



VCU

Virginia Commonwealth University
VCU Scholars Compass

Theses and Dissertations

Graduate School

2020

Zero-inflated longitudinal mixture model for stochastic radiographic lung compositional change following radiotherapy of lung cancer

Viviana A. Rodríguez Romero
Virginia Commonwealth University

Follow this and additional works at: <https://scholarscompass.vcu.edu/etd>



Part of the [Applied Statistics Commons](#), [Biostatistics Commons](#), [Longitudinal Data Analysis and Time Series Commons](#), [Multivariate Analysis Commons](#), [Numerical Analysis and Computation Commons](#), [Statistical Methodology Commons](#), and the [Statistical Models Commons](#)

© The Author

Downloaded from

<https://scholarscompass.vcu.edu/etd/6422>

This Dissertation is brought to you for free and open access by the Graduate School at VCU Scholars Compass. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of VCU Scholars Compass. For more information, please contact libcompass@vcu.edu.

©Viviana Alejandra Rodriguez Romero 2020

All Rights Reserved

**Zero-inflated longitudinal mixture model for stochastic radiographic lung
compositional change following radiotherapy of lung cancer**

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor
of Philosophy in Biostatistics at Virginia Commonwealth University.

by

Viviana Alejandra Rodriguez Romero

B.S. Statistics, Universidad Nacional de Colombia, 2006

M.S. Clinical Epidemiology, Pontificia Universidad Javeriana, 2012

Committee Members:

Nitai D. Mukhopadhyay, Ph.D., Department of Biostatistics (Advisor)

Le Kang, Ph.D., Department of Biostatistics

Adam Sima, Ph.D., Department of Biostatistics

Elisabeth Weiss, M.D., Department of Radiation Oncology

Jinxing Yu, M.D., M.S., Department of Radiology

Virginia Commonwealth University

Richmond, Virginia

July, 2020

Acknowledgment

I have always looked forward to writing this section since it would imply that I am done! However, now that I have reached this moment, my first thought is I want to thank so many people, but I am too tired to do it. Nevertheless, in these unique times where we are forced to be far away from family and friends, it would be ungrateful of me to not say how thankful I am to those whose efforts have supported me. So, here we go!

First, I would like to thank my advisor, Dr. Nitai, whose thoughts and ideas have brought me here today. It would have been impossible to get here without your knowledge, but especially without your encouragement when I was about to give up. I would also like to thank my committee members, Dr. Kang, Dr. Sima, Dr. Weiss, and Dr. Yu, for their guidance with my research. I am extremely thankful to Dr. Nicky Mahon for all her help with the CT Scans, and to Helen Wang for her priceless help with the cluster.

I would like to acknowledge Minciencias for giving me the scholarship to pursue this Ph.D., to the people at Fulbright Colombia and their partners in the U.S.A. for making the process easier, to Pontificia Universidad Javeriana for economic support, and to VCU and the Department of Biostatistics for opening the doors that I needed.

I am also thankful for the department's faculty, especially those who prepared their classes and gave it their all during their lectures. Your examples have been extremely inspiring for me as I prepare for what awaits me in my academic career. To Russ Boyle, who was always available to hear and help us, even when heating his hot glass of water. To Dr. Sima for all your patience, support, and push. Also, to my BCL coworkers – Brian Di Pace, Victoria

Okhomina, and Keighly Bradbrook. Working with you all has been a pleasure.

I could write pages and pages full of stories explaining why I am so grateful to the women in my Fall 2015 cohort, but that would make this document even longer, and nobody would read it. To Anna, Erin, Jin, Kate, Keighly, and their significant others, for not only helping me survive graduate school, but also for opening the doors of their homes and making me feel like I had a family in Richmond. Thank you, thank you so much. I hope half of our plans to see each other in the future become real one day. I would also like to acknowledge all of the students in the department, in particular the Fall 2014 and 2016 cohorts. The Fall 2014 cohort was always looking after us, while Christine always had a smile for me, and Jonathan was always available at three in the morning to help me with stats.

I am incredibly grateful to my family and friends back in Colombia, who, despite the distance, have always had words of support. To my mom and aunts, whose love and strength have been the pillar of my life. To my family in the U.S.A., Germany, Rita, and Betty, for the friendship and all the laughter.

To Fedra, Ellie, and Miranda, for giving me a reason to wake up every morning. And, last but not least, to my husband, to whom I am totally grateful for all the coffees at dawn, for drying my tears of despair when the math stat homework was not going well, and for putting up with my frustration and bad mood during this research. In short, for having accepted the role of carrying out this adventure by my side. I just hope I can live out all the experiences of the adventures I have left in life with your hand in mine.

Table of Contents

Acknowledgment	ii
Abbreviations	xiii
Abstract	xv
1. Introduction	1
1.1 Background knowledge	1
1.2 Motivating example	5
2. Approximation to assemble subvolumes as a combination of voxels . . .	11
2.1 Cubic patch definition	11
2.2 Spherical patch definition	27
2.3 Isodose patch definition	29
2.4 Conclusion	34
3. Two-part mixed effects mixture model for zero-inflated longitudinal compositional data	38
3.1 The likelihood	40
3.2 Model estimation	51
4. Model performance	58
4.1 Results	60
4.1.1 Performance of estimated fixed effects	60

4.1.2	Performance for estimation of nuisance parameters	65
4.2	Conclusion	67
5.	Probabilistic model for longitudinal lung tissue changes following RT . .	72
5.1	Assessing model adequacy	76
5.2	Clinical implications	83
5.3	Conclusion	88
6.	Identification of different patterns of tissue deformation	91
6.1	Results	92
6.2	Conclusion	96
7.	Association of tissue change with radiation pneumonitis in NSCLS patients	100
7.1	Results	101
7.2	Conclusion	107
8.	Discussion	109
Appendix	120
A	Clustering results for the reference patient	120
B	Relevant Code	121
B.1	Chapter 2: Patch definitions	121
B.2	Chapter 3: Two-part mixed-effects mixture model for zero-inflated longitudinal compositional data	136
B.3	Chapter 4: Simulation assessing model performance	163
B.4	Chapter 5: Application on reference patienter data	205
B.5	Chapter 6: Cluster analysis on reference patienter data	229
B.6	Chapter 7: Association of tissue change with radiation pneumonitis in NSCLS patients	233

Vita 236

List of Tables

1.1	Characteristics of 18 NSCLC patients	9
2.1	Total number of patches in the cubic-patch definition for the reference patient	13
2.2	Total number of patches in the cubic-patch definition size 8 for the 18 NSCLC patients	27
2.3	Total number of patches in the isodose-patch definition for the reference patient	32
4.1	Simulation results for estimated fixed effects	62
4.2	Type I error for estimated fixed effects	64
4.3	Simulation results for estimated nuisance parameters	69
5.1	Parameter estimates and standard errors of the fixed effects for modeling RILD by patch definition	74
5.2	Additive logistic transform estimates of the model applied to the composition given that there is some RILD presence	75
6.1	Results of clustering with partition around medoids algorithm by RILD presence for the reference patient	93
7.1	Parameter estimates and standard errors of the fixed effects for modeling RILD using isodose lines patch definition	102
7.2	Correlation and mean absolute scaled error (MASE) between observed and predicted data using isodose lines patch definition	104
7.3	Additive logistic transform estimates of the model applied to the composition given that there is some RILD presence using isodose lines patch definition .	104

7.4	Average silhouette coefficient by number of clusters using partition around medoids algorithm with the isodose lines patch definition	105
7.5	Results of clustering with partition around medoids algorithm using isodose lines patch definition	106
7.6	Model discrimination of radiation pneumonitis by number of clusters using partition around medoids algorithm with the isodose lines patch definition .	107

List of Figures

1.1	Stereotactic body radiation therapy (SBRT) treatment planning images . . .	6
1.2	Multiple 2D transverse slices saved as a 3D matrix	7
1.3	CT scan follow-up with lung shrinkage after radiation therapy	8
2.1	Cubic patch	12
2.2	Dose distribution by patch size and RILD presence in the cubic-patch definition for the reference patient	15
2.3	Calinski-Harabasz (CH) index and mean standard deviation (SD) for patches with RILD by patch size in the cubic-patch definition for the reference patient	16
2.4	Distribution of maximum dose for patches without RILD by patch size in the cubic-patch definition	17
2.5	Distribution of maximum dose for patches with RILD by patch size in the cubic-patch definition	18
2.6	Distribution of minimum dose for patches without RILD by patch size in the cubic-patch definition	19
2.7	Distribution of minimum dose for patches with RILD by patch size in the cubic-patch definition	20
2.8	Distribution of standard deviation of dose for patches without RILD by patch size in the cubic-patch definition	21
2.9	Distribution of standard deviation of dose for patches with RILD by patch size in the cubic-patch definition	22

2.10	Mean standard deviation of dose for patches with RILD by patch size in the cubic-patch definition	23
2.11	Calinski-Harabasz (CH) index for patches with RILD by patch size in the cubic-patch definition	24
2.12	Calinski-Harabasz (CH) index, mean SD and distribution of standard deviation for patches with RILD combining data of 18 NSCLC patients by patch size in cubic-patch definition	25
2.13	Percentage of RILD presence, median dose by patch, and density plot of <i>alr</i> -transformation in dense and hazy RILD by time after RT in the cubic-patch size 8 for the reference patient	25
2.14	Ternary plots of dense and hazy composition by time after RT in the cubic-patch size 8 for the reference patient	26
2.15	Scatter plot of <i>alr</i> -transformed dense and hazy vs. median dose per patch in the cubic-patch size 8 for the reference patient	26
2.16	Spherical patch	28
2.17	Percentage of RILD presence, median dose by patch, and density plot of <i>alr</i> -transformation in dense and hazy RILD by time after RT in the spherical-patch for the reference patient	29
2.18	Ternary plots of dense and hazy composition by time after RT in the spherical-patch for the reference patient	30
2.19	Scatter plot of <i>alr</i> -transformed dense and hazy vs. median dose per patch in the spherical-patch for the reference patient	30
2.20	Percentage of RILD presence by time after RT in the spherical-patch for the 18 NSCLC patients	31
2.21	Isodose patch	32
2.22	Calinski-Harabasz (CH) index for patches with RILD by interval length threshold in the isodose-patch definition for the reference patient	33

2.23	Percentage of RILD presence, median dose by patch, and density plot of <i>alr</i> transformation in dense and hazy RILD by time after RT in the isodose-patch of 0.2 Gy for the reference patient	34
2.24	Ternary plots of dense and hazy composition by time after RT in the isodose-patch of 0.2 Gy for the reference patient	35
2.25	Scatter plot of <i>alr</i> -transformed dense and hazy vs. median dose per patch in the isodose-patch of 0.2 Gy for the reference patient	35
4.1	Simulation Results: Percentage of convergence, average number of iteration before convergence, average time for estimation of fixed effects (min), and average time for estimation of random effects (RE)	61
4.2	Simulation Results for estimated fixed effects: average estimate of fixed effects in the occurrence model ± 1.96 average standard error	64
4.3	Simulation Results for estimated fixed effects: average estimate of fixed effects in the intensity model for <i>alr</i> (Dense) ± 1.96 average standard error	65
4.4	Simulation Results for estimated fixed effects: average estimate of fixed effects in the intensity model for <i>alr</i> (Hazy) ± 1.96 average standard error	66
4.5	Simulation Results: Average correlation ² between simulated and predicted data ± 1.96 standard deviation	67
4.6	Simulation Results: Average mean absolute scaled error (MASE) between simulated and predicted data ± 1.96 standard deviation	68
5.1	Normal Q-Q plots for random effects in the occurrence model (c_i) by patch definition	77
5.2	Chi-Square Q-Q plots for bivariate residuals from the intensity model by patch definition	79
5.3	Standardized residuals vs. dose for the reference patient by patch definition .	80
5.4	Standardized residuals vs. time using median dose as predictor, for the reference patient by patch definition	81

5.5	Estimated mean $alr(Dense)$ and $alr(Hazy)$ vs. logit-probability of RILD for the reference patient by patch definition	82
5.6	Observed number of times with RILD vs. expected probability of RILD for the reference patient by patch definition	84
5.7	Observed vs. predicted $alr(Dense)$ and $alr(Hazy)$ for the reference patient by patch definition	85
5.8	Predicted ternary plots of dense and hazy composition by time after RT in the isodose-patch using median dose as predictor for the reference patient . .	87
5.9	Predicted ternary plots of dense and hazy composition by time after RT in the spherical-patch using log of median dose as predictor	88
6.1	Average silhouette coefficient by number of clusters using partition around medoids algorithm for the reference patient	94
6.2	Boxplot of the median dose by the number of resulting clusters using partition around medoids algorithm for the reference patient	95
6.3	Boxplot of predicted composition (Dense,Hazy,Normal) by the number of resulting clusters using partition around medoids algorithm for the reference patient	96
6.4	Boxplot of observed composition (Dense,Hazy,Normal) by the number of resulting clusters using partition around medoids algorithm for the reference patient	98
7.1	Histograms of parameter estimates of the fixed effects for modeling RILD using isodose lines patch definition by radiation pneumonitis (RP) diagnosis	103
A.1	Silhouette analysis by the number of resulting clusters using partition around medoids algorithm for the reference patient	120

Abbreviations

Alr	Additive log-ratio transformation
AUC	Area under the curve
CD	Compositional data
CG	Center of gravity
CH	Calinski-Harabasz index
Clr	Centered log-ratio transformation
CT	Computed tomography
EM	Expectation-maximization algorithm
Est	Estimate
GGO	Ground-glass opacities
Ilr	Isometric log-ratio transformation
LC	Lung cancer
LCT	Lung carcinoid tumor
MAX	Maximum
MASE	Mean absolute scaled error

MLD	Mean dose to the lungs
MIN	Minimum
NSCLC	Non-small cell lung cancer
NTCP	Normal tissue complication probability
PAM	Partitioning around medoids
PTV	Planning target volume
RILD	Radiation-induced lung damage
RF	Radiation fibrosis
RP	Radiation pneumonitis
RT	Radiation therapy
SBRT	Stereotactic body radiotherapy
SCLC	Small cell lung cancer
SB	Standardized bias
SD	Standard deviation
SE	Standard error
VBA	Voxel- based approach

Abstract

ZERO-INFLATED LONGITUDINAL MIXTURE MODEL FOR STOCHASTIC RADIO-
GRAPHIC LUNG COMPOSITIONAL CHANGE FOLLOWING RADIOTHERAPY OF
LUNG CANCER

By Viviana Alejandra Rodriguez Romero, M.S., Ph.D. Student

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Biostatistics at Virginia Commonwealth University.

Virginia Commonwealth University, 2020.

Major Director: Nitai D. Mukhopadhyay, Ph.D., Department of Biostatistics

Compositional data (CD) is mostly analyzed as relative data, using ratios of components, and log-ratio transformations to be able to use known multivariable statistical methods. Therefore, CD where some components equal zero represent a problem. Furthermore, when the data is measured longitudinally, observations are spatially related and appear to come from a mixture population, the analysis becomes highly complex. For this matter, a two-part model was proposed to deal with structural zeros in longitudinal CD using a mixed-

effects model. Furthermore, the model has been extended to the case where the non-zero components of the vector might a two component mixture population. Maximum likelihood estimates for fixed effects and variance components are calculated by an approximate Fisher scoring procedure base on sixth-order Laplace approximation. The EM algorithm is used to estimate the probability of the mixture model.

The proposed model was used to analyze the radiation therapy effect on tissue change in one patient with non-small cell lung cancer (NSCLC). Five CT-scans were obtained during 24 months following RT. Instead of using voxel-level data, voxels were grouped into larger subvolumes called patches. Data in each patch can be represented by a vector in the form of CD with the proportions of tissue classified as dense, hazy, or normal. A statistical model of radiation-induced lung damage (RILD) over time for each patch as a function of time and dose was implemented. The predicted longitudinal compositions were classified to describe tissue change using cluster analysis. Finally, proposed method and cluster analysis were applied to two groups of patients with and without radiation pneumonitis (RP) to characterize tissue changes in RP.

1. Introduction

1.1 Background knowledge

Compositional data are vectors of non-negative real numbers that represent parts of some whole. Therefore, a composition $\mathbf{x} = [x_1, x_2, \dots, x_D]$ is subject to the constraint $\sum_{i=1}^D x_i = k$, with $x_i > 0$. Usually, the parts of a composition are presented as proportions or percentages, then $k=1$ or 100 .¹ Common examples include the proportion of different cell types in a patient's blood, proportion of nutrients in a patient's diet, portions of the chemical elements in the air, the proportion of working time spent on different activities, etc. In all previous examples, there is a total amount, but our interest is on how this total is split among the different components. The sum over the amounts of all parts must add up to the total. Then, the proportion with respect to the total for each component can be calculated and used for analysis.

Compositional data have been commonly analyzed using multivariate data analysis without any transformation. Nonetheless, this approach can lead to paradoxes and misinterpretations of the results. Karl Pearson was the first person to comment about the issues when analyzing proportions using standard statistical methods. Unfortunately, his observations were ignored until Felix Chayes² pointed out the spurious correlation when using multivariate analysis that leads to a negative bias induced by the constant-sum constraint.^{3,4}

In the 1980s, Aitchison proposed the first methodology with a new geometry framework to analyze compositional data as relative data, using ratios of components and log-ratio transformations to be able to use known multivariable statistical methods. Currently, new approaches using orthonormal coordinates have been implemented to be able to use the real

Euclidean space when analyzing compositional data.^{3,4}

The composition $\mathbf{x} = [x_1, x_2, \dots, x_D]$ is a D -part composition, but it is completely specified by its first $D - 1$ parts, since $x_D = k - x_1 - \dots - x_{D-1}$. Consequently, a D -part composition is a $D - 1$ dimensional vector. Therefore, in the analysis of compositional data, the sample space is restricted to the simplex by the constant-sum constraint.^{1,3,4} This singularity invalidates the use of the Euclidian space, which is the reason why statistical measurements such as mean, standard deviation, correlation, and even hypothesis testing may lead to erroneous interpretations when used in compositional data analysis.^{1,4}

Scale invariance, permutation invariance, and subcompositional coherence are the main principles of compositional data analysis. The scale invariance property states that two compositions \mathbf{x} and \mathbf{y} must produce the same results if their parts represent the same percentages. For instance, $\mathbf{x} = [10, 40, 50]$ and $\mathbf{y} = [50, 200, 250]$ might be expressed as the composition $[0.1, 0.4, 0.5]$, so that \mathbf{x} and \mathbf{y} are compositional equivalent and should lead to the same results. This principle reinforces the idea that a composition holds only relative information. The permutation invariance implies that the results do not depend on the order that the parts appear in the composition. Finally, subcompositional coherence states that the inferences obtained when analyzing a subcomposition should be the same as the inferences using the whole composition when analyzing the common parts.^{1,4}

The simplex space, and the principles of compositional data, stated above, prompts the definition of new basic operations to handle this type of data. Aitchison proposed the perturbation and the powering operations.¹ Then, the simplex with these two operations establishes a vector space.⁴

Following the rationale of the logit transformation, Aitchison proposed the additive log-ratio (*alr*) transformation

$$alr(\mathbf{x}) = \left(\ln \frac{x_1}{x_D}, \dots, \frac{x_{D-1}}{x_D} \right) \quad (1.1)$$

for working with compositional data.¹ As mentioned above, the analysis of compositional

data is based on the relative information, which justifies the use of ratios. Since log-ratios are easier to manipulate than the ratios, and have some advantages in interpretability and power when compared with other transformations,⁵ it justifies the use of logarithms.

In Equation (1.1), the last part of the composition is used as the denominator for the ratio, although any part could have been used. Consequently, different *alr* transformations can be obtained from the same composition. Therefore, the *alr* transformation is not symmetric, which has implications in the analysis such as inner products or distances cannot be computed.^{6,7} Despite this limitation, when analyzing compositional data as dependent variable using linear statistical methods, the *alr* transformation is very useful since the additive logistic normal distribution can be used.^{1,4,8}

As a strategy to solve the above-mentioned limitations, Aitchison proposed the centered log-ratio transformation

$$clr(\mathbf{x}) = \left(\ln \frac{x_1}{g(\mathbf{x})}, \dots, \ln \frac{x_D}{g(\mathbf{x})} \right) \quad (1.2)$$

where $g(\mathbf{x}) = (x_1 \times x_2 \times \dots \times x_D)$ is the geometric mean.¹ The *clr* transformation is symmetric, but the sum of its parts adds to zero. Hence, the covariance matrix of the transformation is singular. Furthermore, the *clr* transformation is not subcompositional coherent, since the geometric mean of composition might not be the same as the geometric mean in a subcomposition.^{4,7} The *clr* transformation, however, preserves distances allowing us to use statistical methods that require this measurement as an analysis input.⁹

Finally, the isometric log-ratio transformation (*ilr*) was defined to create an orthogonal coordinate system in the simplex.⁶ This transformation allows the use of geometric elements in the simplex, permitting the use of most complex statistical approaches. The main idea behind this transformation is to split the composition into two different subsets and use them to construct the orthogonal coordinates for the composition.⁶ In addition of this, there are many different ways to divide the composition into two distinct subsets. Despite the mathematical advantages, this approach leads to some difficulties in choosing the best

subset, as well as in the interpretation of the results.^{7,10}

Since the proposed transformations all use ratios and the logarithmic transformation, compositional data where some component equals zero, run into problem. Some methods for handling this problem have been proposed.^{1,11-16} The modified Aitchison replacement is one of the most frequently implemented method in the analysis of compositional data. This replacement method first substitutes the parts of the composition equal zero by a small quantity, and afterwards, each non-zero component is reduced by a factor to preserve the share ratios.¹¹ The above method works properly in the presence of rounding zeros or generated by detection limits in the measurement. Nonetheless, in the presence of structural zeros (true zeros), the idea of substituting them for a small amount does not seem entirely adequate. For this matter, in this research, a two stages modeling approach has been proposed to deal with structural zeros. The first step determines if the composition contains parts equal zero and the second stage models the proportions among the non-zero parts of the composition.^{12,13}

Compositional data as response variable has been analyzed using the transformed composition as the regressor in multivariate analysis of variance (MANOVA) and multivariate regression assuming multivariate normal distributions, logistic-normal distributions or additive-logistic skew-normal distributions;^{1,4,8} but also some non-parametric and Bayesian approaches have been implemented to particular cases.¹⁷⁻²⁰ Some applications of mixed effect models and polynomial smoothing splines have been used to study the longitudinal behavior of compositional data.^{16,21-23} Also, several applications implementing dimension reduction and clustering of compositional data have been performed, and several publications have talked about the interpretation of these results.^{1,7,9,10} Further, approximations to clustering compositional data trajectories over time have been proposed.²¹

1.2 Motivating example

Lung cancer (LC) is the second most common cancer in both men and women, and even though the incidence of LC has declined, it still represents almost 14% of all new cases of cancer.^{24,25} Although there has been an improvement on early diagnosis and treatment for these patients,²⁶ LC is the leading cause of cancer death, representing about 25% of the cancer deaths in both genders, and overall it has a low 5-year survival (18%).^{24,27} There are three main types of LC, non-small cell (NSCLC), small cell (SCLC), and lung carcinoid tumor (LCT); while SCLC is the most deadly type of LC, NSCLC is the most frequent type, accounting for 80 to 90% of LC cases.²⁵ NSCLC covers squamous cell carcinoma, large cell carcinoma, adenocarcinoma, among other less common types of LC. Even though smoking is the main risk factor for NSCLC, other factors such as exposure to asbestos, air pollution and family history of LC have also shown an association with an increase in LC risk.²⁵

Radiation therapy (RT) is the standard treatment for medically inoperable patients in the early stages of NSCLC.²⁸ The idea behind RT is to break the DNA inside cells, keeping cancer cells from growing and causing them to die.²⁹ Stereotactic body radiotherapy (SBRT), one type of external RT, works with a specially designed coordinate-system to accurately target the tumor while limiting the toxic exposure to normal tissue.^{30,31} As Figure 1.1 shows, SBRT delivers treatment dose over several fractions using beams from different angles, which allows performing treatment in fewer fractions with a higher dose per fraction as compared with the conventional RT. Usually, conventional RT is conducted in 20 to 40 fractions (2Gy/fraction), while SBRT schedules between three and five fractions with doses of 6 to 20 Gy/fraction.³⁰⁻³³ These key features, tumor targeting and fewer fractions with a higher dose, enhance the treatment in dose conformity, tumor coverage, therapeutic ratio, and also allows for delivering the treatment in a shorter time.²⁸

Despite the precision of the SBRT and the fact that the total dose of RT to be delivered is limited by the volume of the normal tissue in the lung, healthy tissue near the tumor is exposed to clinically relevant doses of RT causing radiation-induced lung damage (RILD)

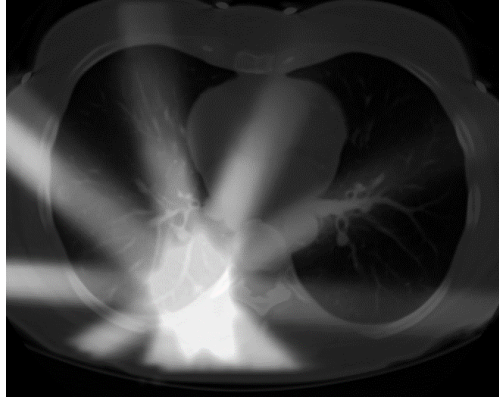
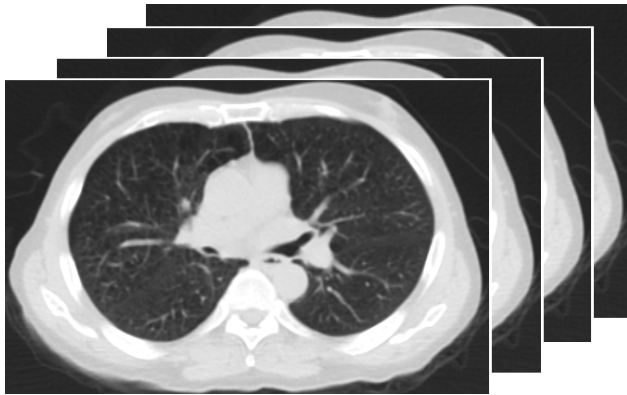


Figure 1.1: Stereotactic body radiation therapy (SBRT) treatment planning images

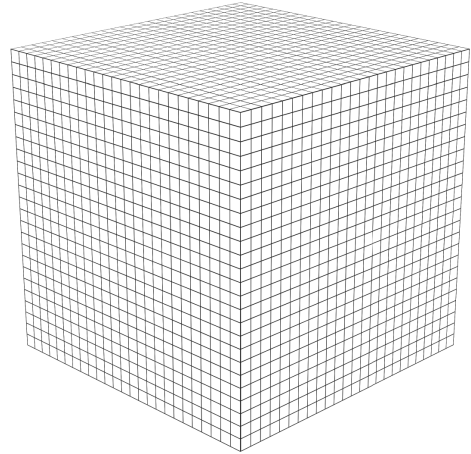
in the vicinity of the tumor.³⁴ The RILD patterns and severity can vary widely among patients and although some damage is observed in almost all patients undergoing RT, it is expected that under a low radiation exposure, healthy tissue's cells do not suffer any permanent damage that may lead to late toxicities.^{28,30} Nonetheless, about 10 to 20% of patients undergoing RT develop clinical symptoms of radiation pneumonitis (RP), a lung inflammation considered one of the significant toxicities following RT,^{31,33} and therefore is one of the main limitations of the dose to be delivered in the treatment.²⁸

Symptoms of RP and radiologic changes in the lung are usually developed between six weeks to six months after RT.^{28,30} Although most patients are asymptomatic or with moderate symptoms, RP has been associated with adverse effects on lung functionality, quality of life, and even mortality.^{28,31,35}

Following RT, computed tomography (CT) scans are obtained at regular time points to assess RP, observe tumor regression, and detect potential tumor recurrence. The CT scan combines numerous x-ray taken from different angles to create multiple 2D transverse slices of the body parts in a gray scale (Figure 1.2a). CT scan data is saved as a three-dimensional array of voxels, where each voxel stores the data (gray scale value) corresponding to a tiny cube of tissue on the CT scan (Figure 1.2b). On the three-dimensional array, each voxel is denoted by ijk , where $i = 1, \dots, I$ and $j = 1, \dots, J$ represents the position of the voxel along the transverse plane; and $k = 1, \dots, K$ being the number of the transverse slices. In



(a) Multiple 2D transverse slices



(b) 3D matrix

Figure 1.2: Multiple 2D transverse slices saved as a 3D matrix

the context of lung CT, the voxel dimensions are approximately 2-3 mm in superior-inferior direction, and less than 1 mm in the other two directions.

Several grading systems that combine clinical, functional and radiographic changes have been implemented for the diagnosis of RP.²⁸ However, each scale uses a different weighting for the above factors, making the comparison between scales particularly challenging.³⁶ Also, the grading systems depend mostly on physicians' interpretations of the CT scan and clinical findings, which might be very subjective and non-reproducible.³⁷ For example, dyspnea, one of the common symptoms, is non-specific and can be caused by other pathologies; or improvement in lung function can be caused by a shrinkage of the tumor, hiding the presence of RP.³⁶

Even though RILD pattern and the recovery process after conventional RT have been widely studied, the hypofractionation feature of the SBRT might result in different injury patterns. Some studies have reported that RILD after SBRT is visible a little later and with different appearance compared to conventional RT.^{30,32} Some approximations to use the CT scan to build normal tissue complication probability (NTCP) models have been developed, most of them looking in the dose/volume relationship to plan the treatment.^{28,36} However, the results have not been entirely satisfactory. Among several reasons, these models use a

single value to summarize all the dose effects, ignoring the local distribution of the dose-response as well as the local variability in composition and functionality of the lung.^{36,38-40}

A second approach, based on voxel data, has been proposed to deal with the underlying dose distribution in the lung.^{39,41} Even though this approach offers a good perspective for analyzing CT images, this represents some issues regarding the analysis over time since the lung tissue deforms after RT (fibrotic distortions), e.g., tumor shrinkage, which allows the expansion of the normal tissue into the vacant space.⁴² Thus, it is not possible to assume that a voxel in the three-months CT scan represents the same tissue as the corresponding voxel at the six-months CT scan (Figure 1.3). Image registration would improve the temporal integrity of the patches by taking care of systemic sources such as patient motion, breathing pattern, etc. However, traditional registration algorithms, including deformable registrations, do not account for fibrotic distortion. Therefore, even with state-of-the-art image registration, integrity of the voxel tissue across time cannot be assured in clinical image pairs without extensive, labor-intensive effort to manually annotate images to a sufficient degree. As a result, following the RILD pattern for each voxel is not meaningful from a medical point of view.

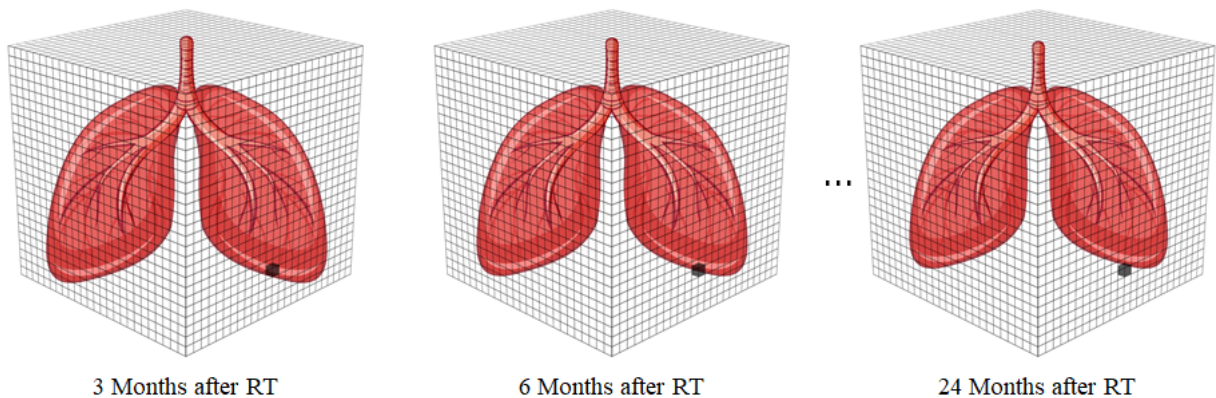


Figure 1.3: CT scan follow-up with lung shrinkage after radiation therapy

All things considered, there is a clear need for objective image interpretation that provides better characterization of RILD, its correlation with RP, and that allows a better differentiation of RP from other lung pathologies after RT. With a much lower level of noise

compared to earlier medical images, statistical models have much to offer in reaching this goal by using minute changes in tissue texture. Hence, it is necessary to develop statistical models for smaller segments of tissue as observed in the current day high-resolution images rather than an aggregate model. The new approach should also be flexible enough to accommodate the spatial dependence between neighboring voxels, and other risk factors when modeling the changes over time of RILD using comprehensive CT scan data.

Our preliminary data consists of five CT scans of one NSCLC female patient (reference patient) obtained at 3, 6, 12, 18 and 24 months following SBRT with 57 Gy dose. The patient’s tumor was located on the right upper lobe, and the planning target volume (PTV) was 23.2 mL. Data of the reference patient has been used to build and test all methodologies in the present work. Additionally, data of 18 NSCLC patients with follow-up CT scans at 3,6 and 12 months post-treatment has been used to assess the reproducibility of the proposed methodologies. Patients were treated from 2007 - 2016 at the Massey Cancer Center at Virginia Commonwealth University. Treatment dose varied between 46 and 70 Gy delivered in several fractions every other day. Nine patients in this group have a positive diagnosis of RP. Frequency matching was used to assure the same distributions by treatment technique and primary tumor volume between patients with and without RP. Table 1.1 shows a description of the patient characteristics, overall and by RP diagnosis

Characteristics	Overall	No RP (n=9)	RN (n=9)
Age, mean (sd)	68.2 (10.1)	64.5 (3.8)	72 (2.6)
Male	9 (50.0%)	3 (33.3%)	6 (66.7%)
African American race	10 (55.6)	5 (55.6%)	5 (55.6)
Current smoker	9 (50.0%)	6 (66.7%)	3 (33.3%)
Diabetes	3 (16.7%)	1 (11.1%)	2 (22.2%)
Tumor located in the upper lobe	14 (77.8%)	6 (66.7%)	7 (77.8%)
Baseline lung density (HU), mean (sd)	-724 (67.0)	-721 (21.6)	-727 (24.2)
Planning target volume, mean (sd)	49 (38.2)	48 (12.4)	50 (13.8)
Prescription dose (Gy), mean (sd)	59 (1.2)	59 (2.1)	60 (1.2)

Table 1.1: Characteristics of 18 NSCLC patients

For this study, one physician manually contoured the area with RILD on the CT scans

using commercial contouring software (MIM Maestro v6.6, MIM Software, Cleveland, OH). Even though, in the CT scan, the extent of radiographic injury is observed in a continuous manner with the RT dose, voxels were classified according to their radiographic findings into three-threshold based ordinal categories of radiographic injury, namely, dense (representing areas of consolidation and fibrosis), hazy (representing ground-glass changes) and normal.^{30,32}

After RILD contouring, deformable image registration was performed using a multi-pass b-spline to account for major changes in lung architecture after RT. Follow-up CT scans were registered to the baseline treatment planning scan using a commercial registration package (Velocity - Varian Medial Systems).

2. Approximation to assemble subvolumes as a combination of voxels

A lot of information regarding disease progression/regression, radiation response, tumor growth is captured in the CT scans of the patients. Statistical analysis of these CT scans using a single measure to summarize the whole RT dose-response effect in the lung is clearly losing a lot of information contained in the entire image. The other extreme, using each voxel level data is impractical, as the spatial integrity of the voxel over time is very difficult to ascertain even after a very modern image registration technique. Both such approach would lead to misspecified model in the longitudinal analysis of RILD. Thus, a more meaningful approach would be to consider combination of the voxels into larger subvolumes (patches), small enough that the treatment dose to the voxels within a patch is approximately homogeneous, but large enough to be trackable over time. Several definitions of such patches, as well as an evaluation of their performance is presented in the sequel for 18 NSCLC patients scanned between 3 to 5 times following their RT.

2.1 Cubic patch definition

The first approach, called cubic patch, consists of p voxels along each direction in the CT scan (i , j , and k). Figure 2.1 (a) shows an example of the cubic patch composition grouping eight voxels along each direction ($p=8$). Each small cube represents a voxel, and each colored group of voxels represents a patch. In this case, each patch consists of 512 voxels. In the following, different definitions of these patches combining p voxels along each direction with $p = 4, \dots, 10$ are explored to define their suitability through some notion of optimality.

Once a patch has been defined, voxel-level data must be combined to obtain data at

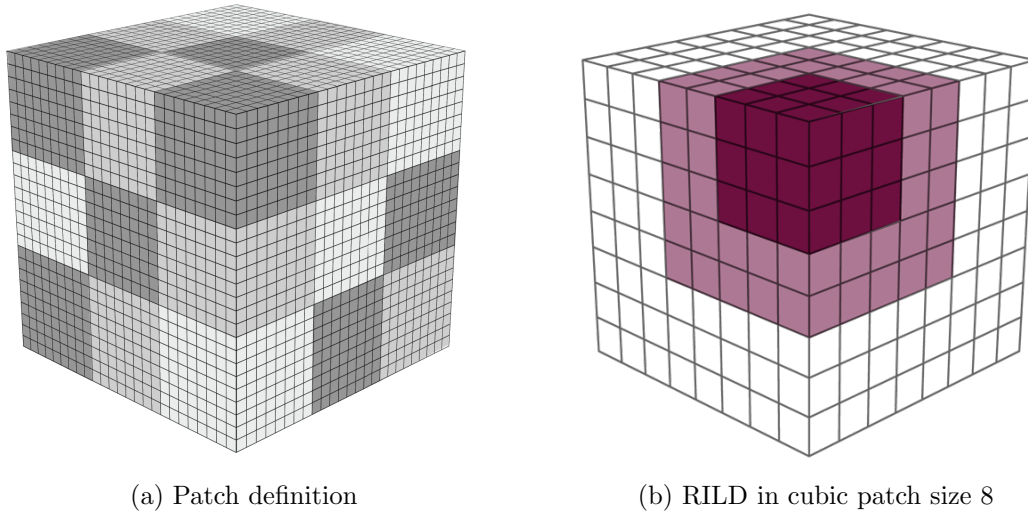


Figure 2.1: Cubic patch

the patch-level. Patch-level dose was obtained through the minimum RT dose, maximum, median, mean, and standard deviation of voxel dose. To obtain a measure of the RILD for each patch, the proportions of voxels falling into the category of 'dense' or 'hazy' was calculated. The Figure 2.1 (b) is a zoom-in of one of the patches of size 8. The voxels in dark purple represent voxels classified as dense regions, light purple are hazy regions, and white voxels represent normal tissue. In this example, there are 27 dense voxels, 98 hazy voxels, and 387 normal tissue voxels. Then, the RILD measure for this patch is 5.3% dense, 19.1% hazy, and 75.6% normal. This calculation was done for all the follow-up CT scans. Therefore, each patch is summarized into a vector $p_{it} = (d_{it}, h_{it}, c_{it})$, where $i = 1, \dots, \text{total number of patches}$, $t = 3, 6, 12, \dots, T_i$; d_{it} and h_{it} represent the proportion of dense and hazy voxels in the patch i on time t , $0 \leq d_{it} \leq 1$ and $0 \leq h_{it} \leq 1$ and $c_{it} = 1 - d_{it} - h_{it}$. This vector is in the form of compositional data where components of the vector are positive and sum to 1. In this case, two vectors p_{it} and $p_{i't}$ with the proportions for two different patches i and i' , are correlated if they are close neighbors. Thus, we have correlated compositional data measured over time, where the unit of observation is a composition within each patch.

The cubic-patch approach was implemented in the CT scans of our reference patient.

Patch size	Total number of patches	Patches without RILD and max dose = 0		Patches for analysis		Patches with RILD	
		n	%	n	%	n	%
$4 \times 4 \times 4$	425,984	311,403	73.10	114,581	26.90	295	0.26
$5 \times 5 \times 5$	222,789	162,515	72.95	60,274	27.05	201	0.33
$6 \times 6 \times 6$	133,128	98,443	73.95	34,685	26.05	115	0.33
$7 \times 7 \times 7$	82,140	59,304	72.20	22,836	27.80	86	0.38
$8 \times 8 \times 8$	53,248	37,753	70.90	15,495	29.10	77	0.50
$9 \times 9 \times 9$	38,988	27,823	71.36	11,169	28.64	56	0.50
$10 \times 10 \times 10$	29,744	21,285	71.56	8,459	28.44	59	0.70

Table 2.1: Total number of patches in the cubic-patch definition for the reference patient

The total number of patches by patch size is presented in Table 2.1. The total number of patches range from 425,984 to 29,744 depending on the size of the patch. However, for all patch sizes, more than 70% of the patches did not receive any radiation (max dose equals zero) and did not show any RILD, then these patches are excluded from further analysis. Out of the patches used for analysis purposes, less than 1% present RILD at some point during the 2-years follow-up for the reference patient.

Since RILD is a direct adverse event of the received radiation dose, our statistical model would use patch dose as a covariate to model the change in tissue damage. Hence, the dose distribution within patch should be homogeneous enough to make any of our summary dose definition to be a good representative measure of patch dose. Figure 2.2 shows the distribution of the maximum, minimum, and standard deviation of the received dose by presence/absence of RILD for the reference patient. For the maximum amount of radiation, it is expected that patches without RILD received a small amount of RT while patches with RILD received a larger amount of RT. The max dose for more than 80% of the patches without RILD was lower than 2 Gy, while it was more than 10 Gy for all patches with RILD. As expected, the minimum dose in patches without RILD was lower than 2 Gy for more than 90% of the patches for all patch sizes. On the other hand, in patches with RILD, more than 50% have a minimum dose larger than 5 Gy. Finally, patches without RILD have a low within patch dose variability (Figure 2.2 (e)), thus indicating a good homogeneity in

the amount of radiation received within those patches. Nevertheless, there is more within patch variability in patches with RILD (Figure 2.2 (f)). For patch sizes 4 to 8, at least 50% of the patches have an SD lower than 5 Gy. However, in patch sizes 9 and 10, almost all the patches have an SD larger than 5 Gy, indicating larger heterogeneity in those patch sizes.

Since the distributions of the minimum and maximum dose for both RILD groups, as well as the distribution of the SD in the group without RILD, meet the expected, a deeper analysis of the heterogeneity of the dose received in the patches with RILD was performed using the Calinski-Harabasz (CH) index.⁴³ The CH index is a method for clustering evaluation, which measures how distinct a patch is from other patches. This index calculates the ratio between the average sum of squares between patches and the average sum of squares within patches, for each patch size. Higher values of the CH index are indicators of better separation between patches. In comparison with other clustering validation methods, the use of the average sum of squares instead of the sum of squares allows for a more fair assessment when comparing clustering setups with different numbers of clusters, in this case, different number of patches. Figure 2.3 (a) shows the CH index for the different patch sizes. Although patch size 10 has the highest value of CH index, the mean SD at size 10 is larger than 10 Gy (Figure 2.3 (b)). Now, the patch size 8 shows the second-largest CH index, 51.0% of the patches in that size have an SD lower than 5 Gy, and the mean SD is 7.5 Gy. Therefore, the cubic-patch of size 8 seems the most appropriate for analysis purposes.

As validation of the obtained results, the same statistics presented in Table 2.1, and Figures 2.2 and 2.3 were calculated for the 18 NSCLC patients. The distribution of the maximum, minimum and SD of the RT dose for patches without RILD is presented on Figures 2.4, 2.6, and 2.8, respectively. As observed, the distribution across patch sizes is very similar within patients, which does not give indications of any issues that should concern us with any of the patch sizes. In addition, the distribution of the maximum dose in patches with RILD gives similar results (Figure 2.5). Summarizing the results of the standard deviation within patches and the CH index (Figures 2.10 and 2.11, respectively),

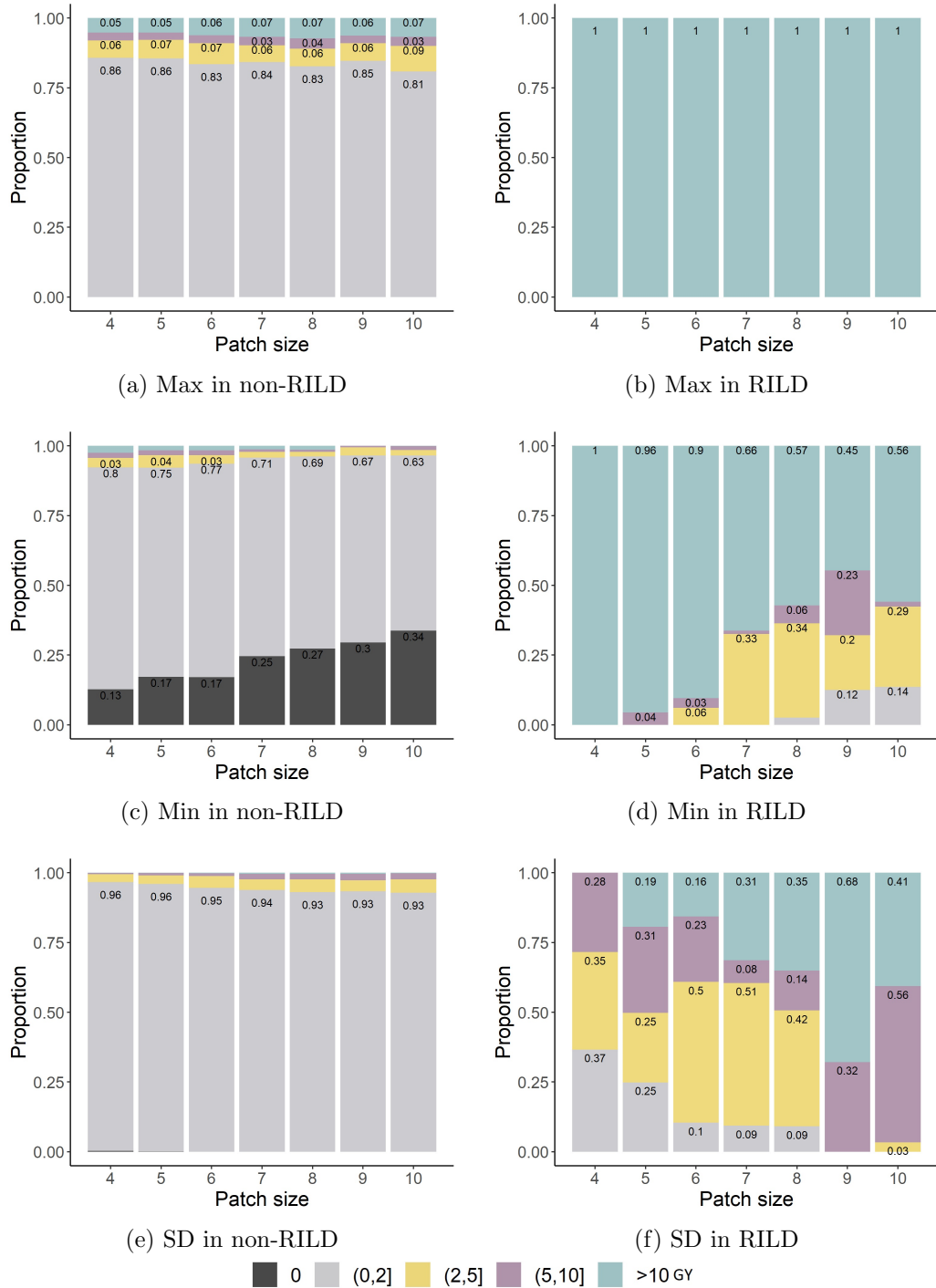


Figure 2.2: Dose distribution by patch size and RILD presence in the cubic-patch definition for the reference patient

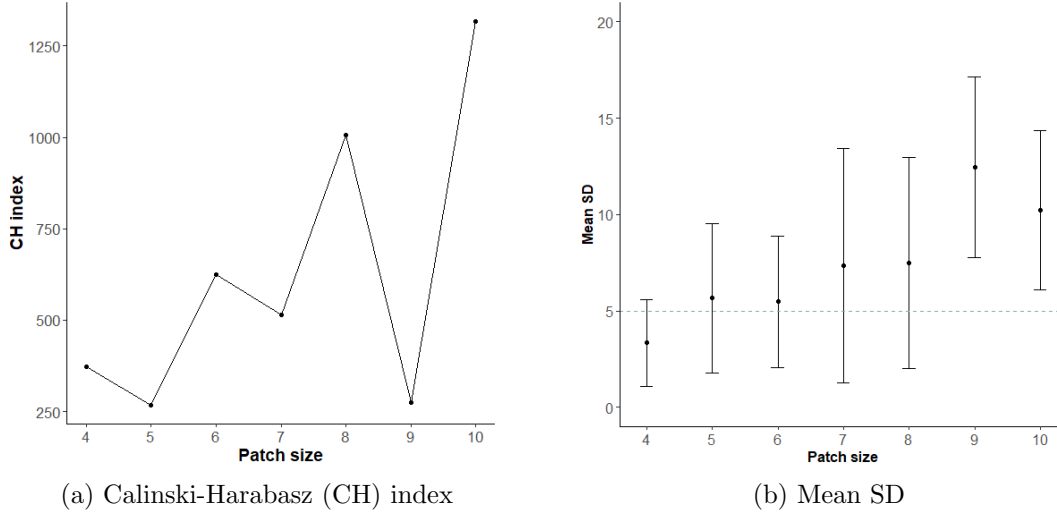


Figure 2.3: Calinski-Harabasz (CH) index and mean standard deviation (SD) for patches with RILD by patch size in the cubic-patch definition for the reference patient

patch size 8 seems a reasonable option for almost all patients but patients 14 and 18 where sizes 10 and 9 seem more suitable. Then, trying to consolidate all patient’s data, overall mean SD and CH index were calculated (Figure 2.12 (a)). Even though sizes 9 and 10 have a larger CH index, they also have a larger mean SD, implying larger variability within patches. Therefore, with the idea of maintaining the lowest heterogeneity in the received dose inside each patch, patch size 8 is selected as the best option for further analysis.

The cubic patch size 8 has 15,495 patches on the reference patient, and 77 of them presented RILD at some point during the first 24 months after RT. Then, less than 1.0% of the patches present some level of RILD over time after RT (Figure 2.13 (a)). The median dose in patches with RILD is at least 10 Gy (Figure 2.13 (b)), while the median dose in patches without RILD range from 0 Gy to 50 Gy. Figure 2.13 (c) shows the distribution of composition $p_{it} = (d_{it}, h_{it}, c_{it})$ after *alr* transformation. As observed, both dense and hazy transformed data show a bimodal distribution.

The ternary plots of the RILD composition overtime for the reference patient are displayed in Figure 2.14. Change from 3 months to 6 months is marked by an increase in dense proportions. Following that, at 12 months, the plot shows a reduction in both dense and

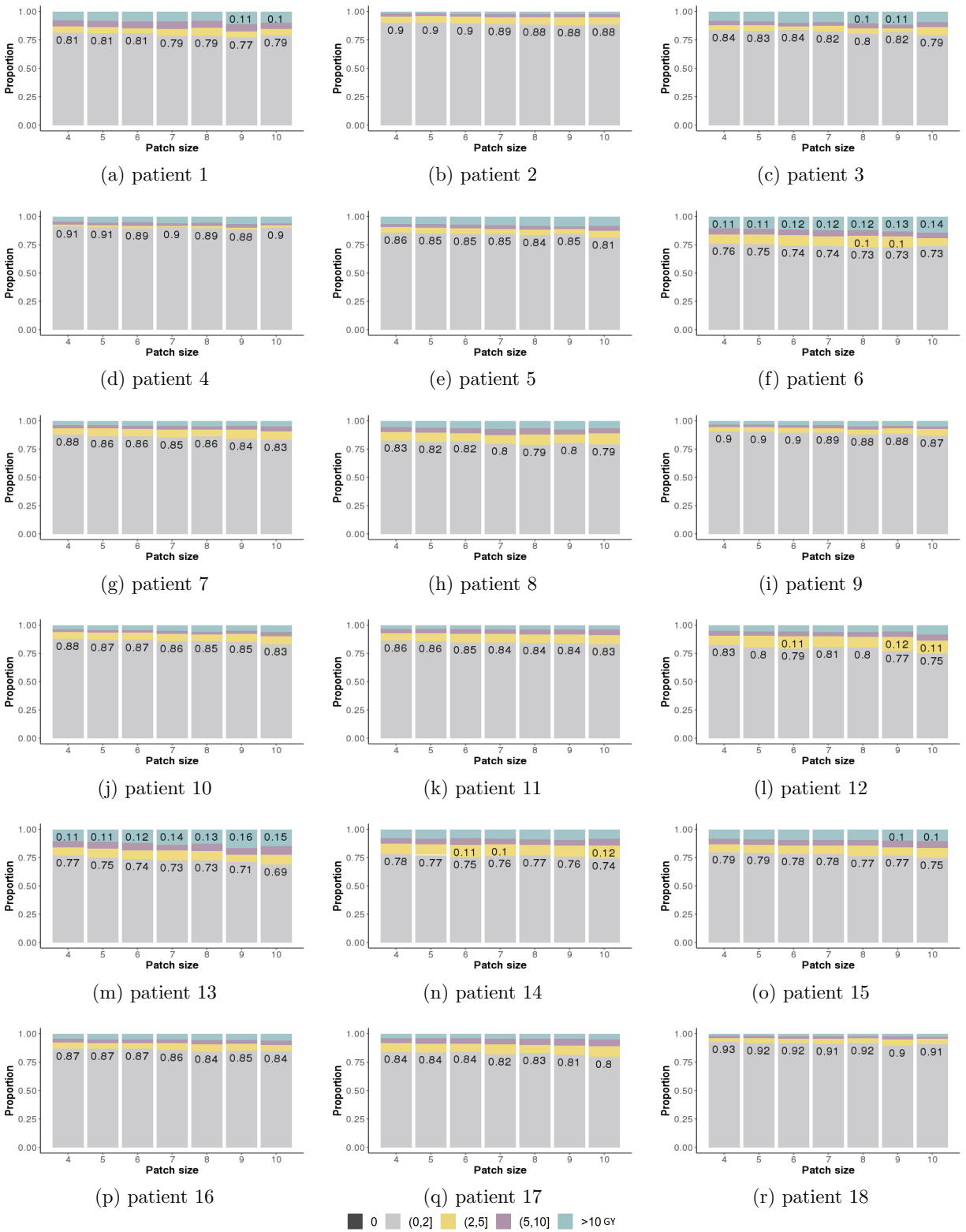


Figure 2.4: Distribution of maximum dose for patches without RILD by patch size in the cubic-patch definition



Figure 2.5: Distribution of maximum dose for patches with RILD by patch size in the cubic-patch definition

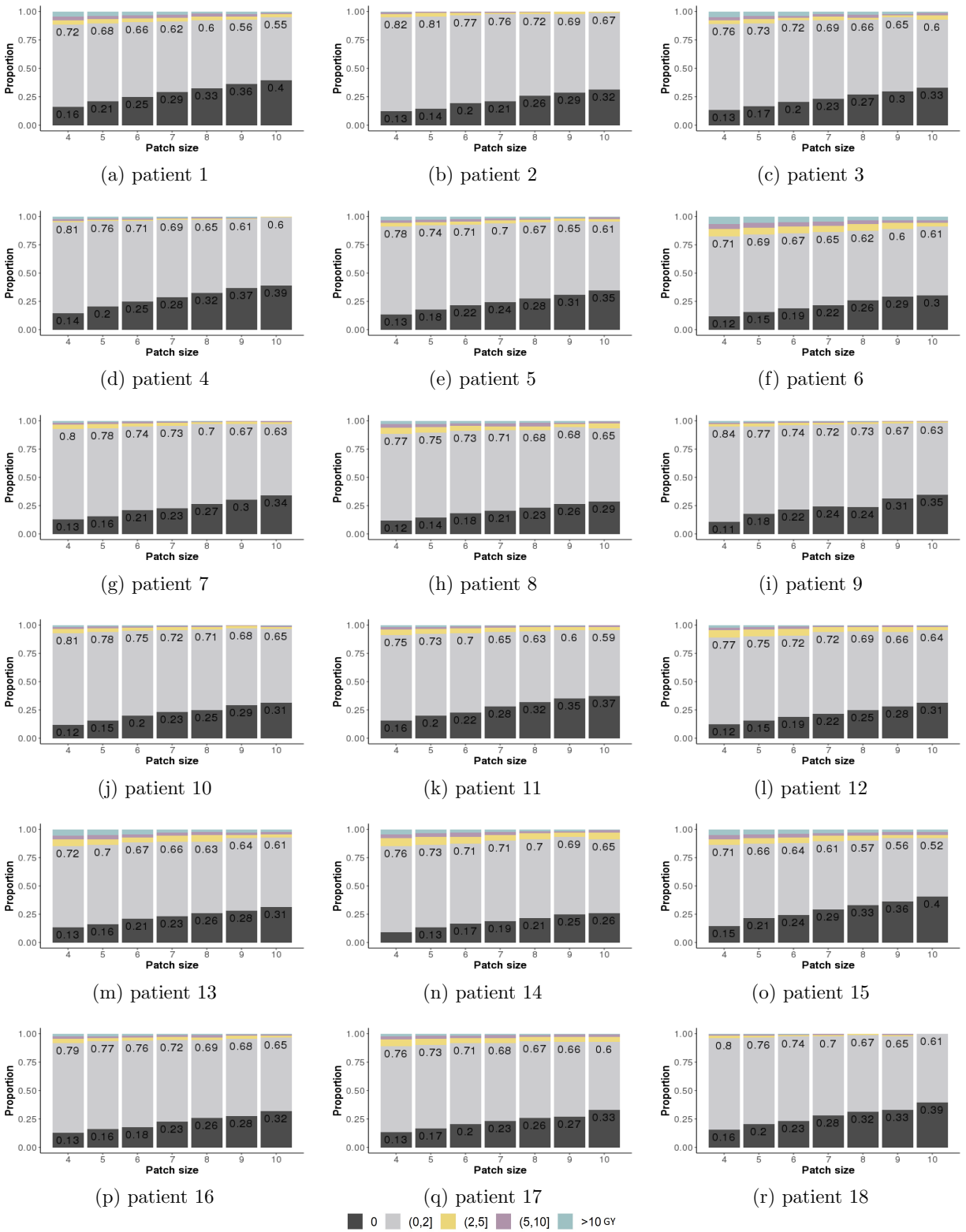


Figure 2.6: Distribution of minimum dose for patches without RILD by patch size in the cubic-patch definition

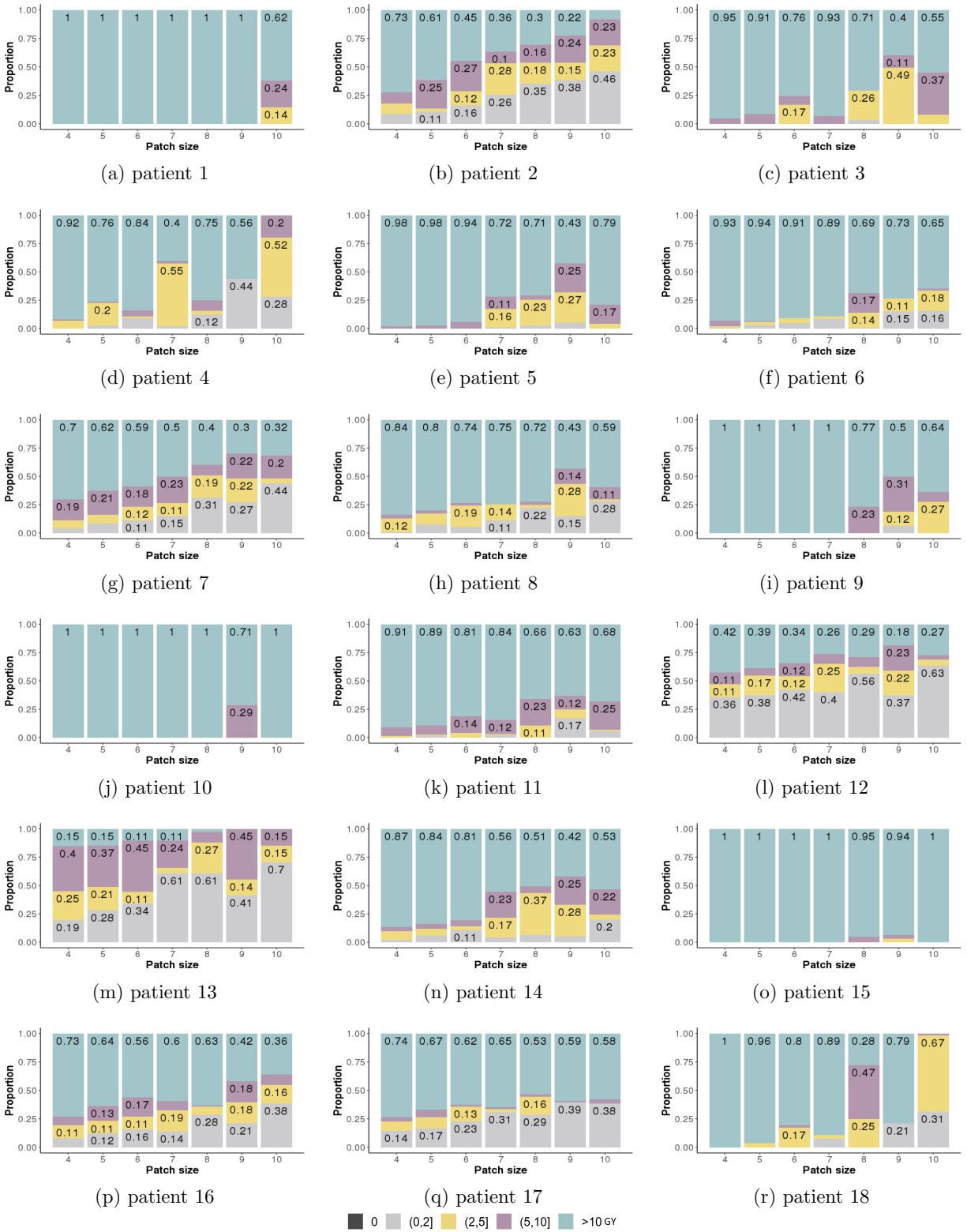


Figure 2.7: Distribution of minimum dose for patches with RILD by patch size in the cubic-patch definition

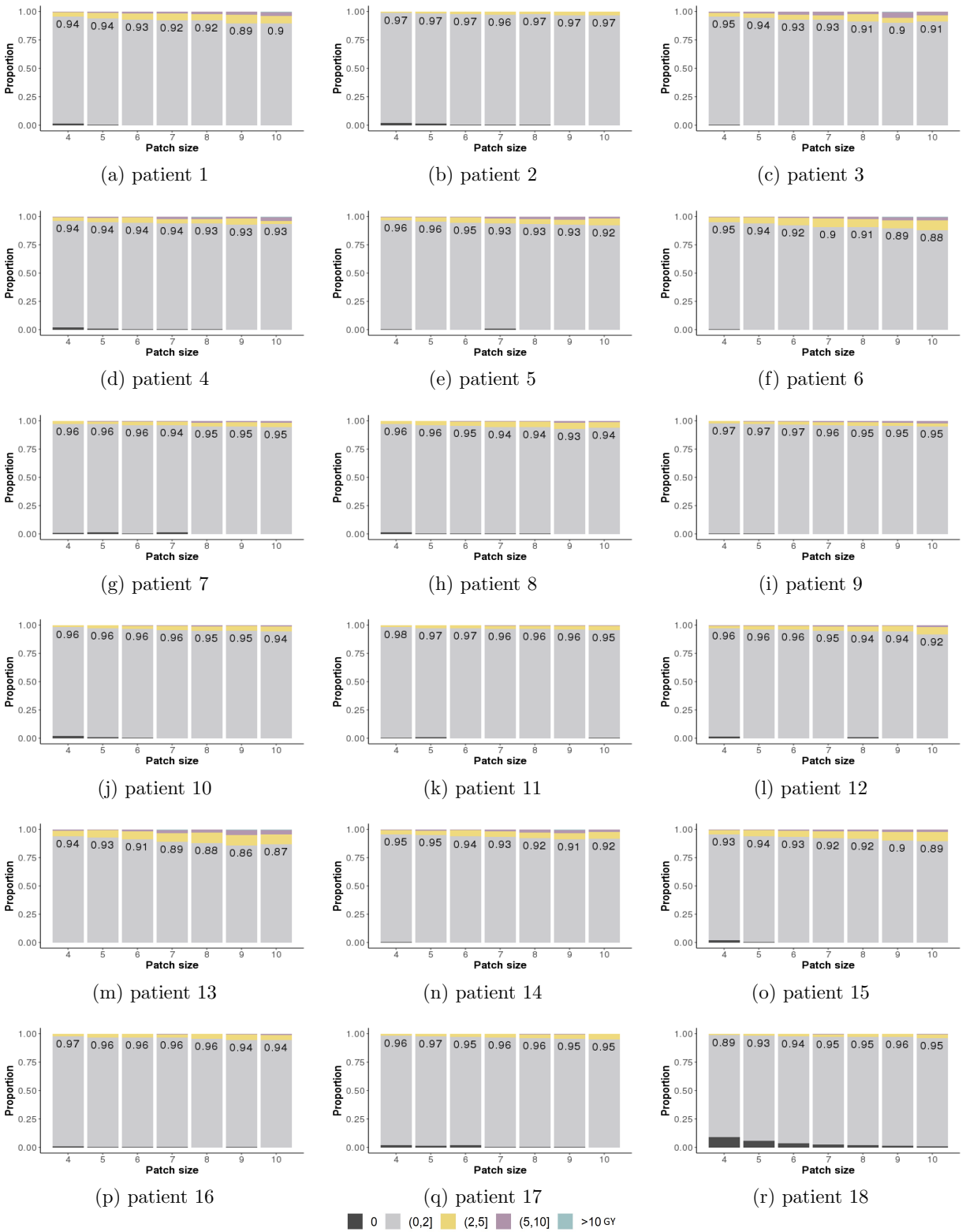


Figure 2.8: Distribution of standard deviation of dose for patches without RILD by patch size in the cubic-patch definition

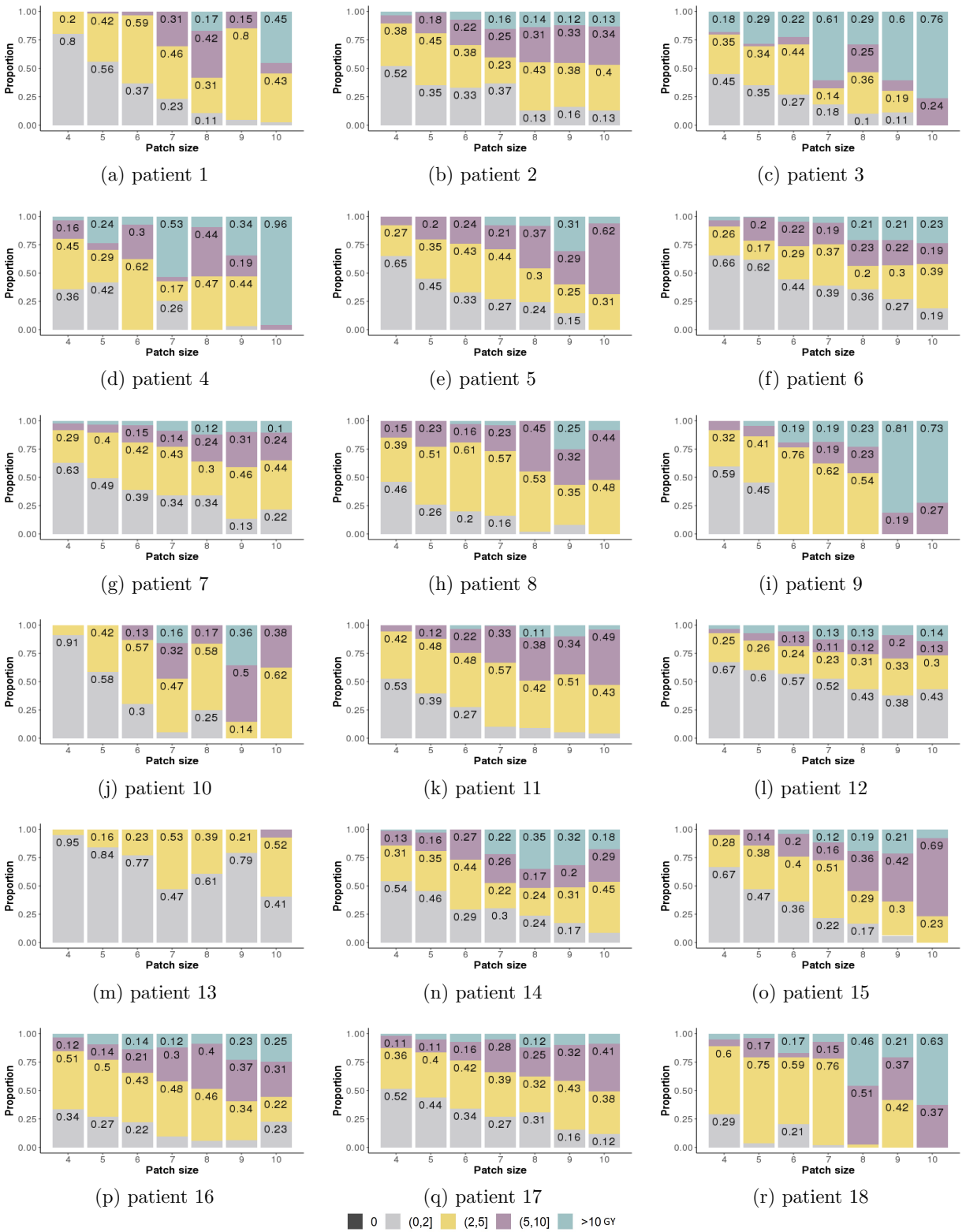


Figure 2.9: Distribution of standard deviation of dose for patches with RILD by patch size in the cubic-patch definition

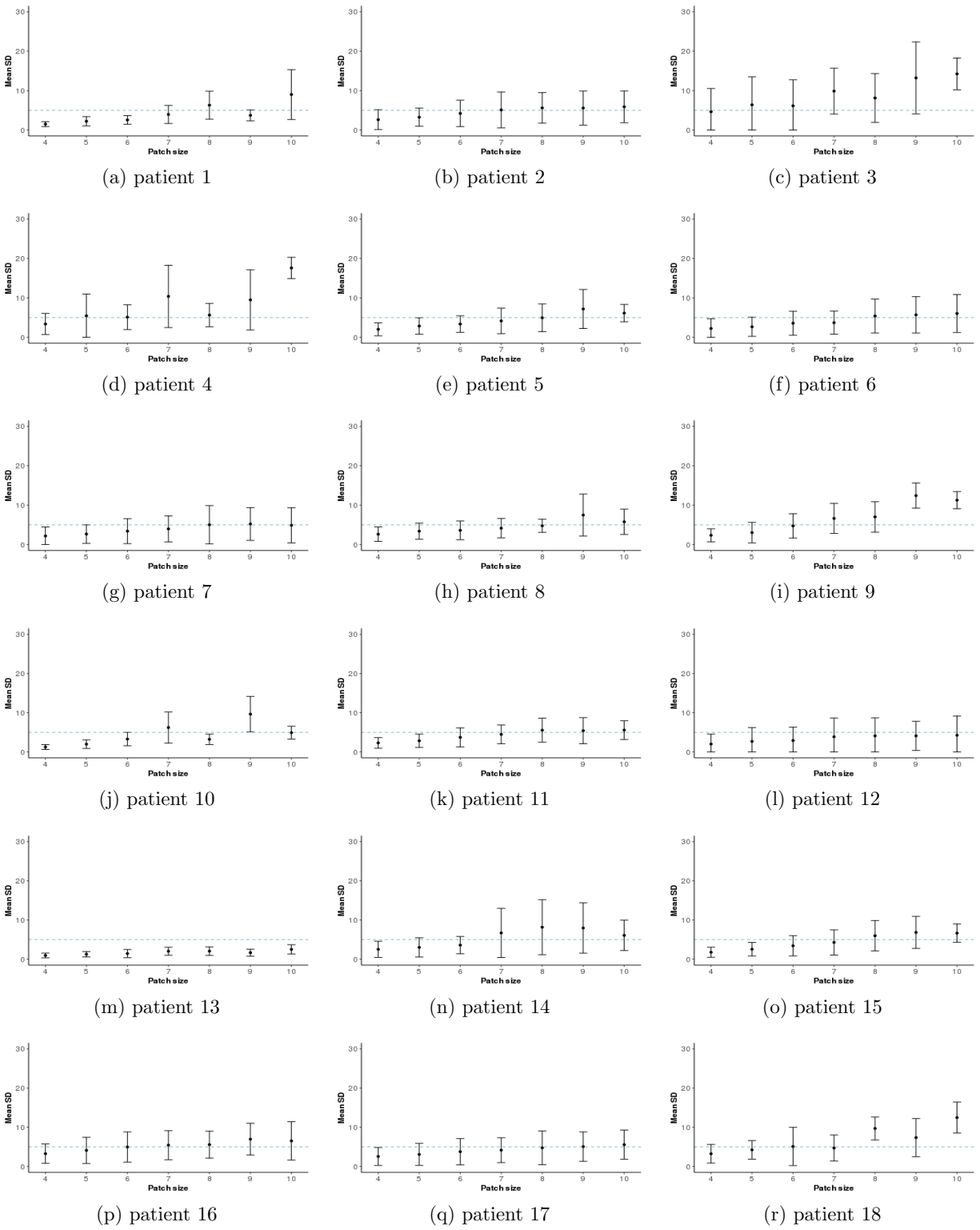


Figure 2.10: Mean standard deviation of dose for patches with RILD by patch size in the cubic-patch definition

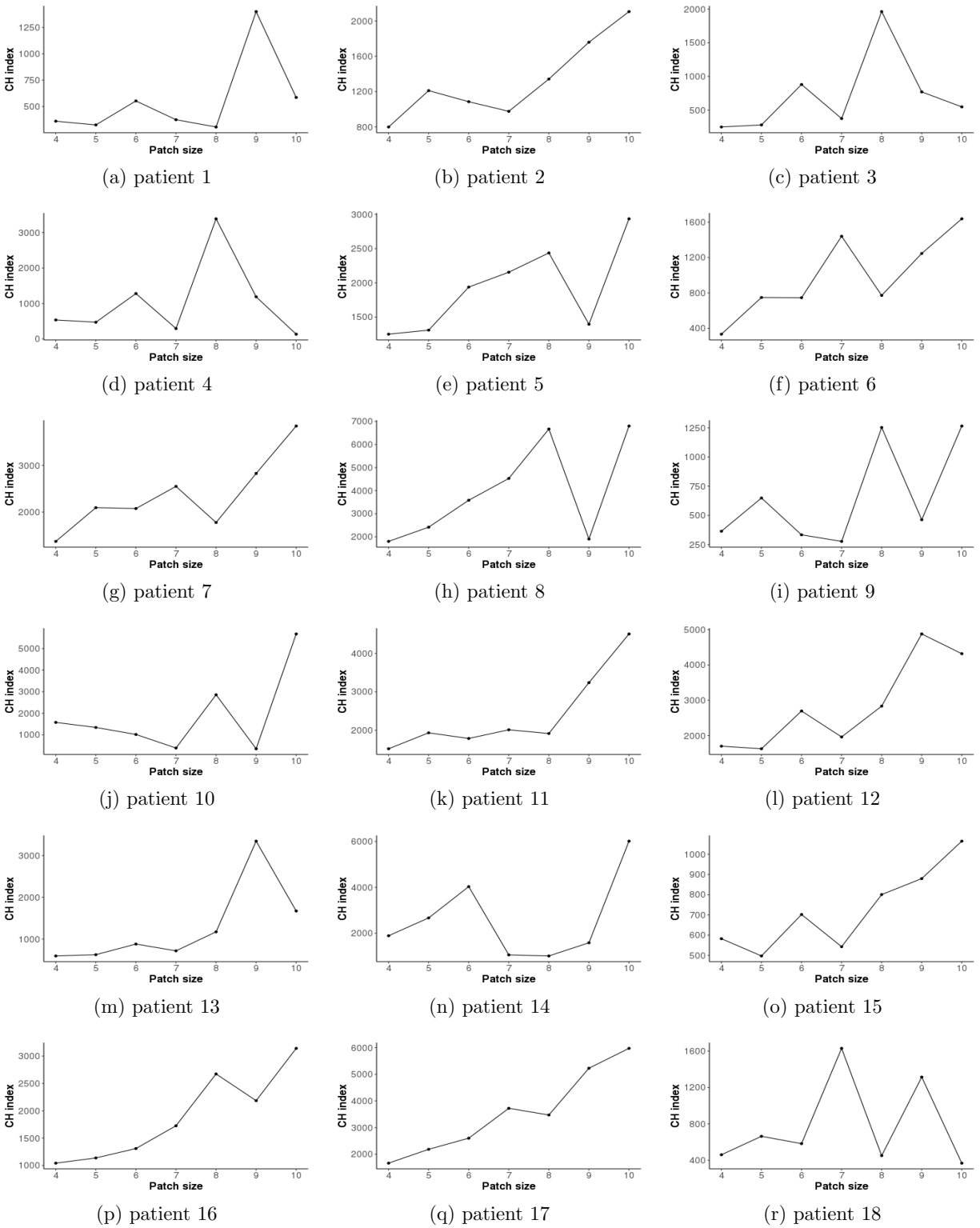


Figure 2.11: Calinski-Harabasz (CH) index for patches with RILD by patch size in the cubic-patch definition

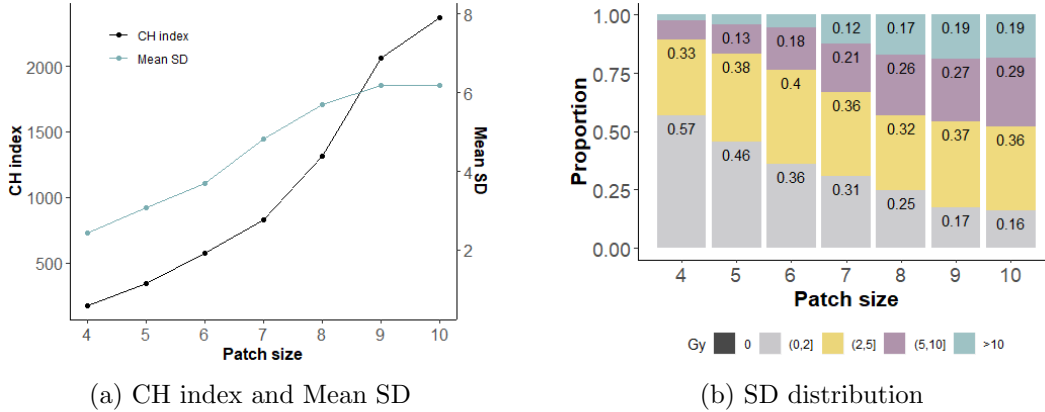


Figure 2.12: Calinski-Harabasz (CH) index, mean SD and distribution of standard deviation for patches with RILD combining data of 18 NSCLC patients by patch size in cubic-patch definition

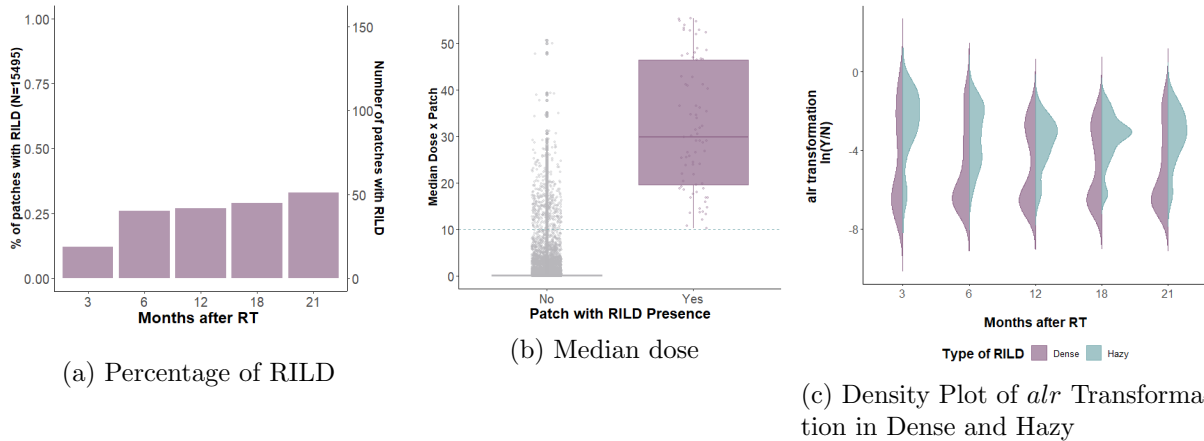


Figure 2.13: Percentage of RILD presence, median dose by patch, and density plot of *alr* transformation in dense and hazy RILD by time after RT in the cubic-patch size 8 for the reference patient

hazy portions. However, at 24 months, the percentages of hazy and dense increase. Finally, Figure 2.15 shows the values of $alr(dense)$ and $alr(hazy)$ vs. the median dose of RT for the reference patient. As observed, at low doses of RT, the percentage of dense and hazy are zero, and even though some damage is observed after a dose of 16 Gy, many of the patches present zero or very low values of dense and hazy with large doses of RT.

Table 2.2 presents the number of patches for each of the 18 NSCLC for the cubic-patch size 8. The total number of patches goes from 40 thousand to 73 thousand for the different patients. However, between 50% to more than 80%, in some cases, of the patches did not

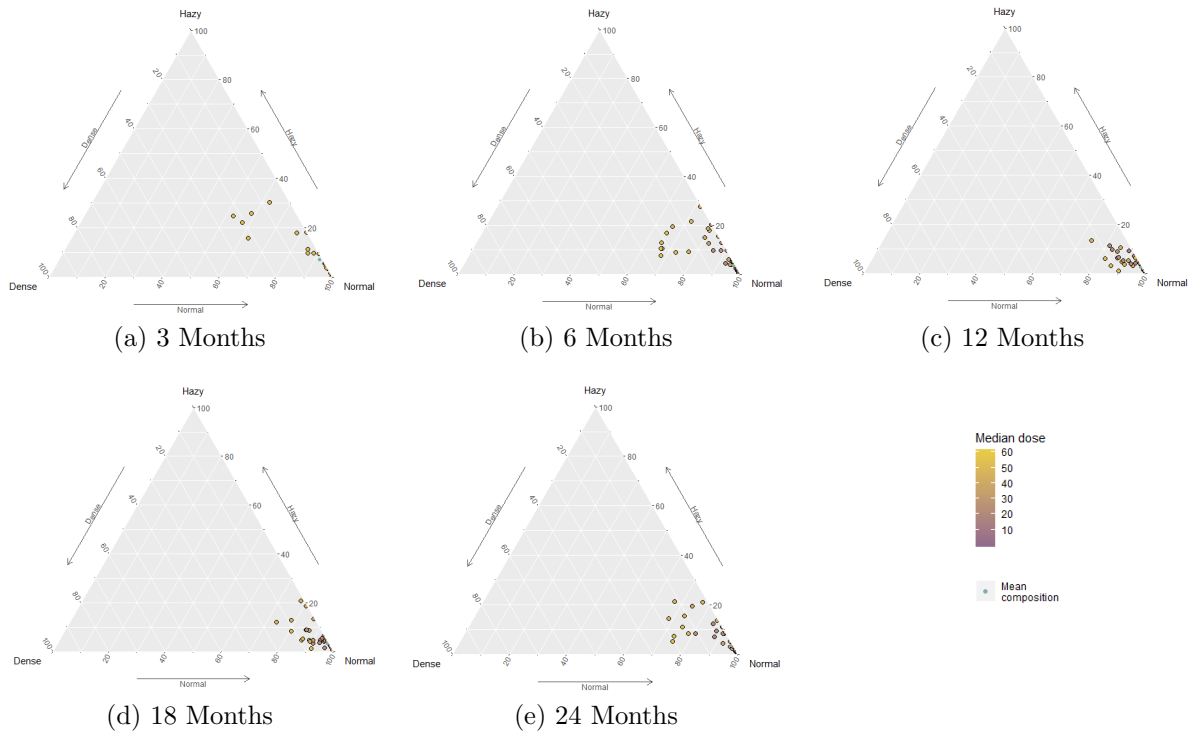


Figure 2.14: Ternary plots of dense and hazy composition by time after RT in the cubic-patch size 8 for the reference patient

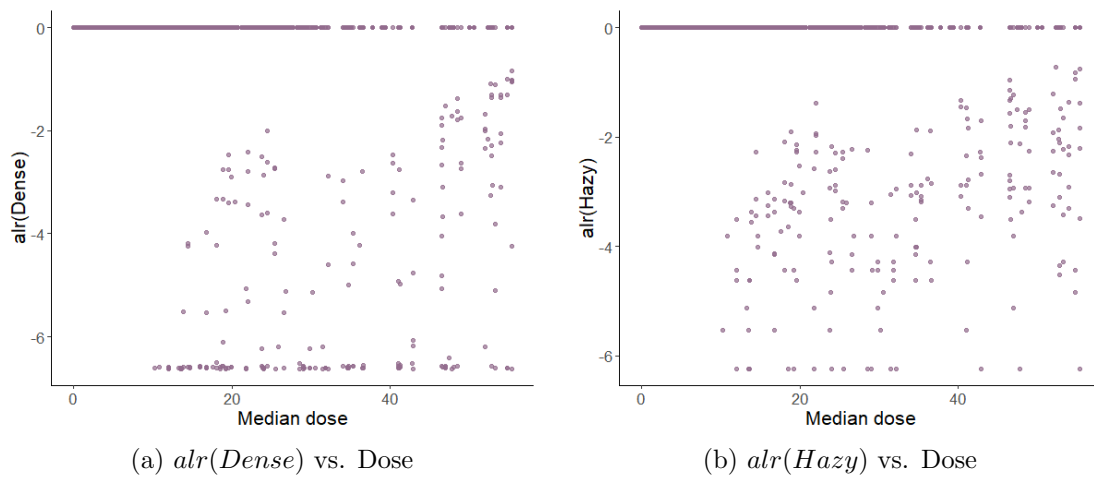


Figure 2.15: Scatter plot of alr -transformed dense and hazy vs. median dose per patch in the cubic-patch size 8 for the reference patient

Patient	Total number of patches	Patches without RILD and max dose = 0		Patches for analysis		Patches with RILD	
		n	%	n	%	n	%
1	61440	45755	74.47	15685	25.53	65	0.41
2	57344	36245	63.21	21099	36.79	247	1.17
3	60632	42049	69.35	18583	30.65	69	0.37
4	61440	45335	73.79	16105	26.21	32	0.20
5	61440	47856	77.89	13584	22.11	83	0.61
6	61440	46785	76.15	14655	23.85	206	1.41
7	57344	39880	69.55	17464	30.45	390	2.23
8	65536	45096	68.81	20440	31.19	109	0.53
9	69632	47444	68.14	22188	31.86	13	0.06
10	69632	52294	75.10	17338	24.90	12	0.07
11	69632	56884	81.69	12748	18.31	133	1.04
12	49152	38614	78.56	10538	21.44	335	3.18
13	40960	26100	63.72	14860	36.28	33	0.22
14	49152	25355	51.58	23797	48.42	409	1.72
15	65536	56779	86.64	8757	13.36	42	0.48
16	65536	52298	79.80	13238	20.20	144	1.09
17	73728	47520	64.45	26208	35.55	326	1.24
18	73728	48976	66.43	24752	33.57	72	0.29

Table 2.2: Total number of patches in the cubic-patch definition size 8 for the 18 NSCLC patients

received any radiation neither present some RILD; therefore, those patches are not considered for further analysis. As observed on the reference patient, the percentage of patches with RILD is small on the 18 NSCLC patients (0.06% to 3.18%).

2.2 Spherical patch definition

A second patch definition approach was defined as the combination of the voxels at the same distance of the voxel with the highest received dose (Reference). Using Figure 2.16 (a) as an example, the Reference is identified on the CT scan (red dot), and the Euclidian distant from the Reference is calculated for all other voxels. Then, the first patch consists of the Reference voxel and the voxels within a voxel distance from it. The second patch consists of those voxels with a Euclidean distance greater than 1 voxel but less or equal to 2 voxels.

The third patch has the voxels with at Euclidian distance between 2 and 3, and so on. Even though the main idea of this patch approach is to build spheres, given the cuboid shape of the voxels, the operationalization of the spherical patch looks more like Figure 2.16 (b), where each color represents a different patch.

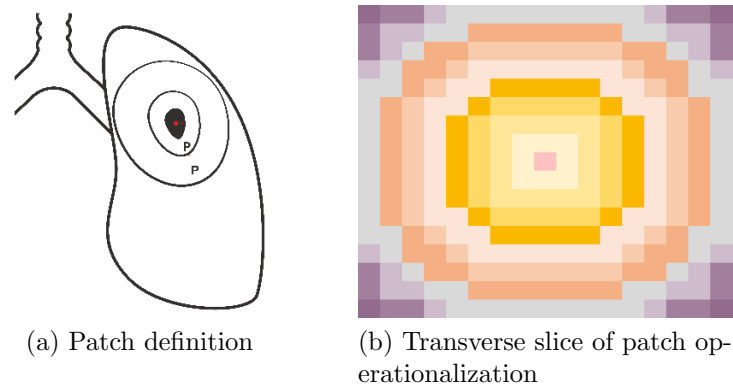


Figure 2.16: Spherical patch

The spherical patch approach on the reference patient has 297 patches, one of them composed of voxels that did not receive any dose and did not present RILD. Then, 296 patches are useful for analysis purposes. Given the way the patches are constructed in this approach, each patch has a different size. In the reference patient, the patch's size varies from 27 voxels for the first patch to 114,640 voxels in the largest patch. Out of the 296 patches, 50 (16.89%) presented RILD at some point during the first 24 months after RT (Figure 2.17 (a)). The median dose in patches with RILD range from 0.27 Gy to 57.08, while the median dose in patches without RILD range from 0 Gy to 0.26 Gy (Figure 2.17 (b)). Figure 2.17 (c) shows the distribution of the composition $p_{it} = (d_{it}, h_{it}, c_{it})$ after alr transformation. As observed, both dense and hazy transformed data show a bimodal distribution.

The ternary plots of the RILD composition overtime for the reference patient are displayed in Figure 2.18. At 3 months, some patches present a high percentage of dense and hazy. Change from 3 months to 6 months is marked by a decrease in dense and hazy proportions. Following that, at 12 months, almost no hazy, either dense, compositions are observed. Finally, Figure 2.19 shows the values of $alr(dense)$ and $alr(hazy)$ vs. the median dose of

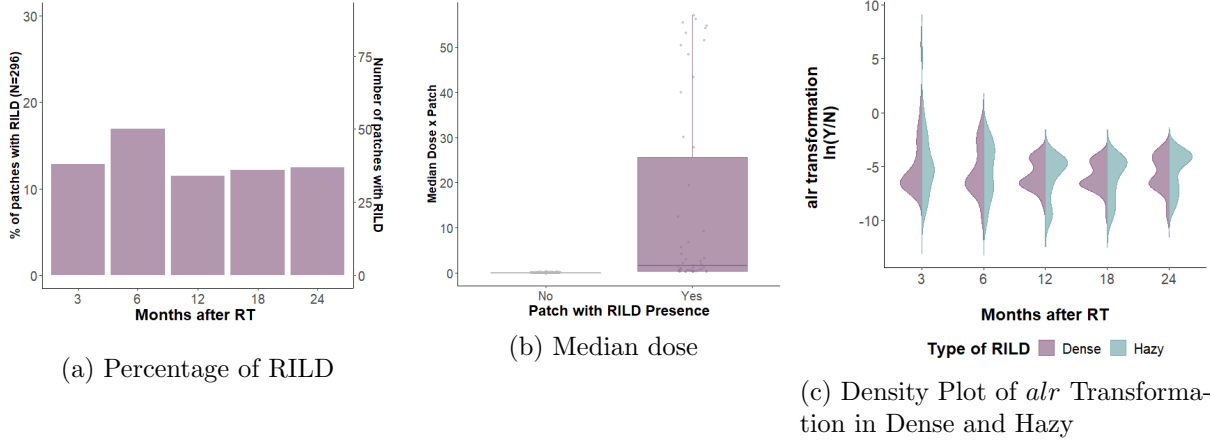


Figure 2.17: Percentage of RILD presence, median dose by patch, and density plot of air transformation in dense and hazy RILD by time after RT in the spherical-patch for the reference patient

RT for the reference patient. As observed, at doses lower than 40 Gy, very small values of dense and hazy are observed. However, in the highest doses, the percentages of both dense and hazy increase.

When analyzing the data of 18 NSCLC patients, the number of patches ranges from 185 to 398 and the percentage of patches with RILD ranges from 2.52% to 45.54% (Figure 2.20). Additionally, the distribution of the transformed data presented also a bimodal shape, as observed for the reference patient.

2.3 Isodose patch definition

Lastly, a third patch approach was defined using the isodose lines, combining voxels that received somewhat a uniform dose (Figure 2.21 (a)). The lines are drawn at regular intervals of length t , and voxels with a received treatment dose within the interval are combined on the same patch. In that way, voxels that received a dose on the interval $(0, t]$ were combined in one patch, and voxels with a received dose on the interval $(t, 2t]$, were combined on a different patch. Voxels that did not receive any radiation therapy (dose equals zero) were combined on the same patch. For the isodose lines approach, intervals of length $t = 0.05, 0.1, 0.2, 0.5, 1$ Gy have been attempted. Figure 2.21 (b) shows a transverse slice of the isodose-patch approach

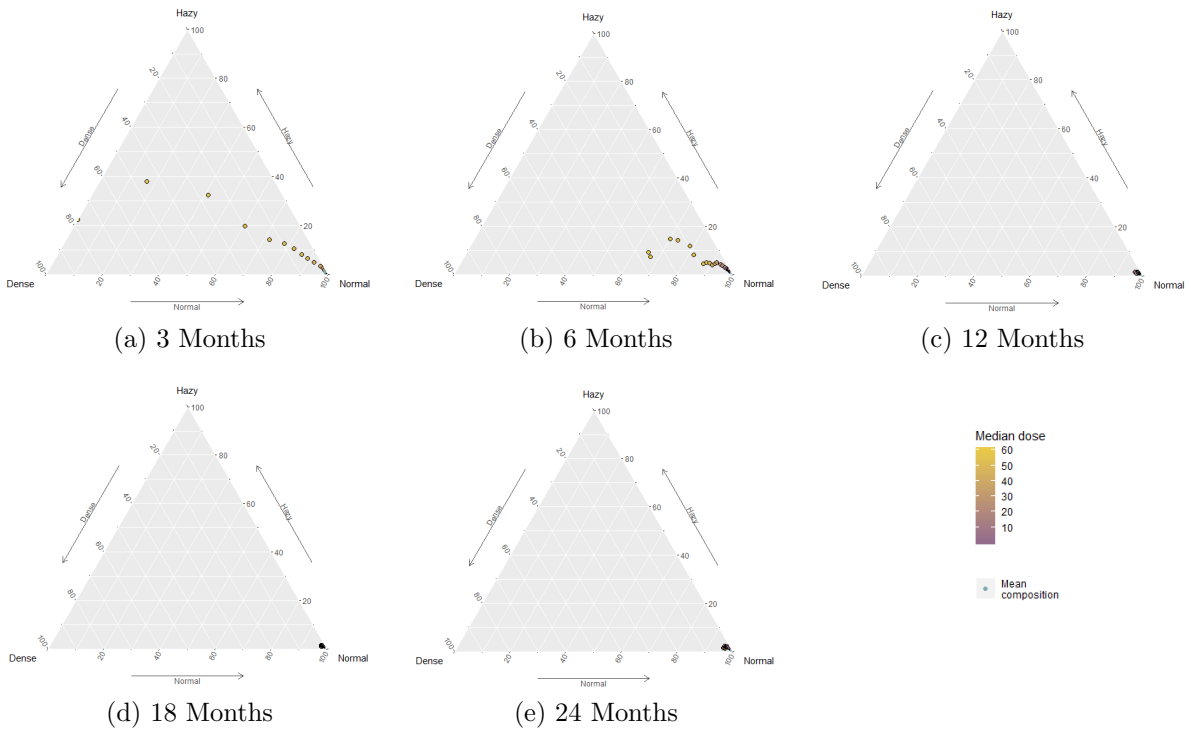


Figure 2.18: Ternary plots of dense and hazy composition by time after RT in the spherical-patch for the reference patient

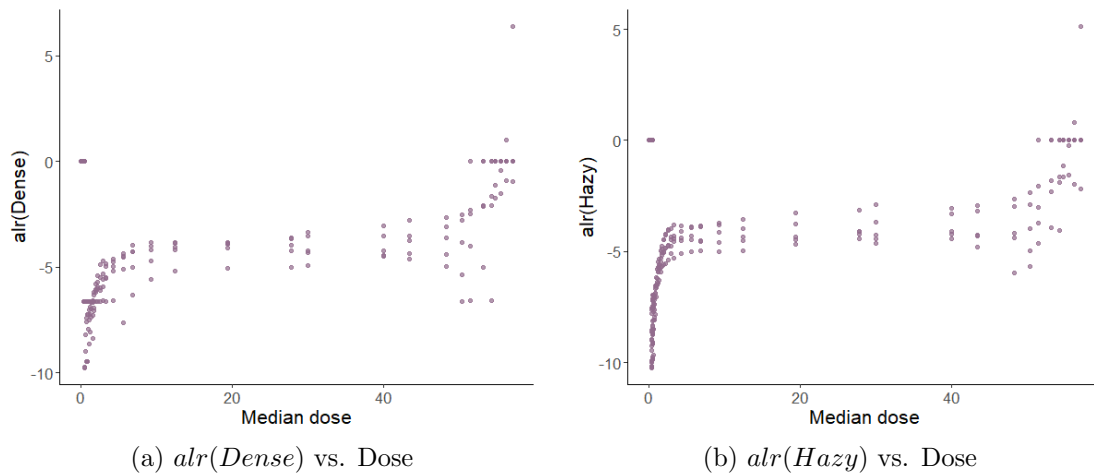


Figure 2.19: Scatter plot of alr -transformed dense and hazy vs. median dose per patch in the spherical-patch for the reference patient

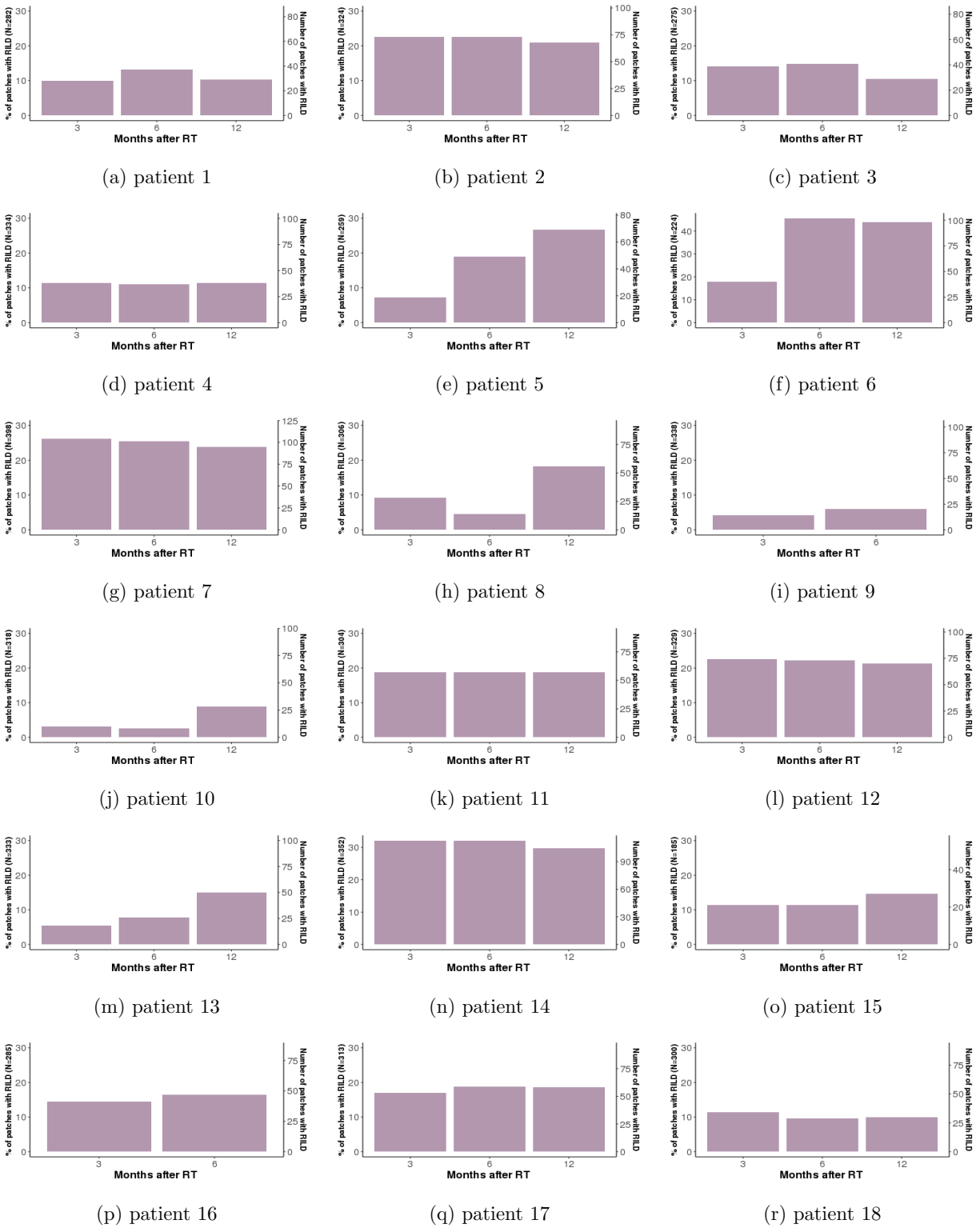


Figure 2.20: Percentage of RILD presence by time after RT in the spherical-patch for the 18 NSCLC patients

applied to the reference patient. The ocean-green region corresponds to voxels that did not receive any RT dose, and the center of the image, where the smallest region is located (dark blue), is the region with the highest received dose.

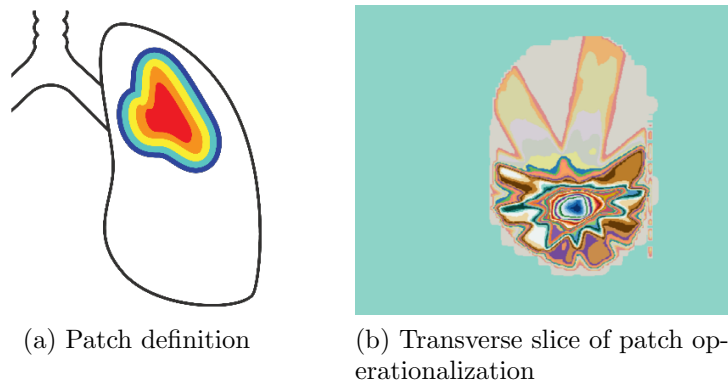


Figure 2.21: Isodose patch

The total number of patches by threshold t is presented in Table 2.3. In the reference patient, the total number of patches goes from 1151 on the interval of length $t = 0.05$ Gy, to 59 patches when $t = 1$ Gy, but there is always one patch completely formed with zero-dose voxels that it is not useful for analysis purposes. As the spherical patch, in the isodose-patch, the number of voxels per patch varies from patch to patch. The patch's size varies from 5 voxels for the first patch to 5,856,399 voxels depending on the patch interval length. In the isodose-patch definition, 50 to 60% of the patches present RILD after the RT.

Interval length	Total number of patches	Voxels x patch	Patches with RILD n	%
0.05	1151	5 - 2,185,368	627	54.47
0.10	576	16 - 3,512,097	326	56.60
0.20	289	16 - 4,698,252	169	58.48
0.50	116	156 - 5,534,109	69	59.48
1.00	59	156 - 5,856,399	36	61.02

Table 2.3: Total number of patches in the isodose-patch definition for the reference patient

Figure 2.22 shows the CH index for the isodose-patch approach. Since the isodose patches are built based on the dose, the CH index monotone decreases as the interval increases, and

its value shows a high homogeneity in the dose within patches. Therefore, all interval-length thresholds would represent a good option to guarantee dose homogeneity. Then, for further analysis, the interval length of 0.2 Gy has been used since it increases the likelihood of tracking the same region over time.

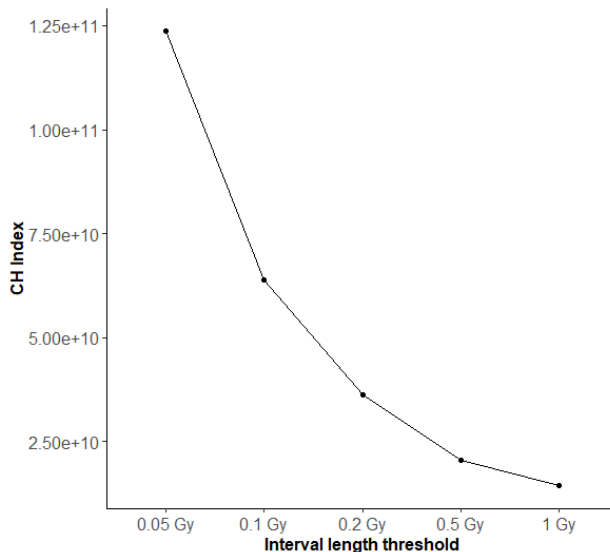


Figure 2.22: Calinski-Harabasz (CH) index for patches with RILD by interval length threshold in the isodose-patch definition for the reference patient

The isodose-patch of interval length of 0.2 Gy has 288 patches for analysis, and the rate of patches with RILD is 58.48% (Figure 2.23 (a)). The median dose in patches with RILD is at least 22 Gy (Figure 2.23 (b)), while the median dose in patches without RILD range from 0 Gy to 24 Gy. Figure 2.23 (c) shows the distribution of the composition $p_{it} = (d_{it}, h_{it}, c_{it})$ after *alr* transformation. As observed, both dense and hazy transformed data show a bimodal distribution.

The ternary plots of the RILD composition overtime for the reference patient are displayed in Figure 2.24. At 3 months, some patches present a high percentage of dense and hazy. Change from 3 months to 6 months is marked by a steep decrease in dense and hazy proportions. Following that, at 12 months, low percentages of hazy and dense are observed. Similar patterns are observed at 18 and 24 months after RT. Figure 2.25 shows the values of $alr(dense)$ and $alr(hazy)$ vs. the median dose of RT for the reference patient. As observed,

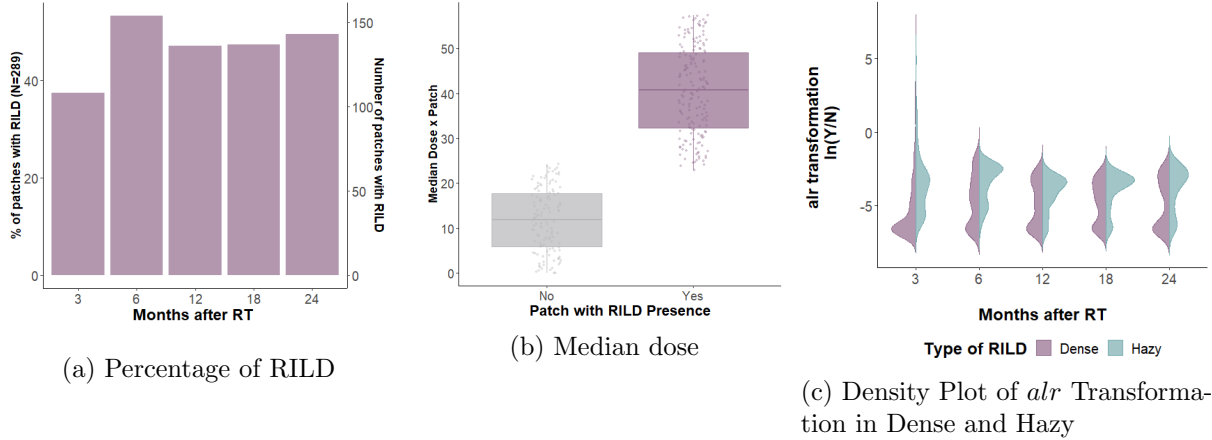


Figure 2.23: Percentage of RILD presence, median dose by patch, and density plot of *alr* transformation in dense and hazy RILD by time after RT in the isodose-patch of 0.2 Gy for the reference patient

at doses lower than 20 Gy, almost not dense neither hazy are observed. However, in the highest doses, the percentages of both dense and hazy increase.

For the 18 NSCLC patients, the number of patches ranges from 234 to 353, and the percentage of patches with RILD goes from 21.59% to 99.34%. Additionally, the distribution of the transformed data presented also a bimodal shape, as observed for the reference patient.

2.4 Conclusion

Cancer patients are usually followed via non-invasive imaging at frequent intervals. From the practitioner point of view, the use of medical imaging has become quite sophisticated and continues to evolve into a very accurate tool for medical diagnosis. Present levels of technological sophistication render very high resolution for almost all medical images of different modalities. As newer modalities and better resolution of existing modalities continue to improve, a much better level of prediction accuracy, automation, and early diagnosis of disease is expected. Therefore, development of image-based analysis that characterize imaging changes and predict the clinical outcomes is currently the focus of medical research.

Nevertheless, the analysis of medical images has been under-developed as the technology of medical imaging progressed exponentially over the past few decades. It is rarely used

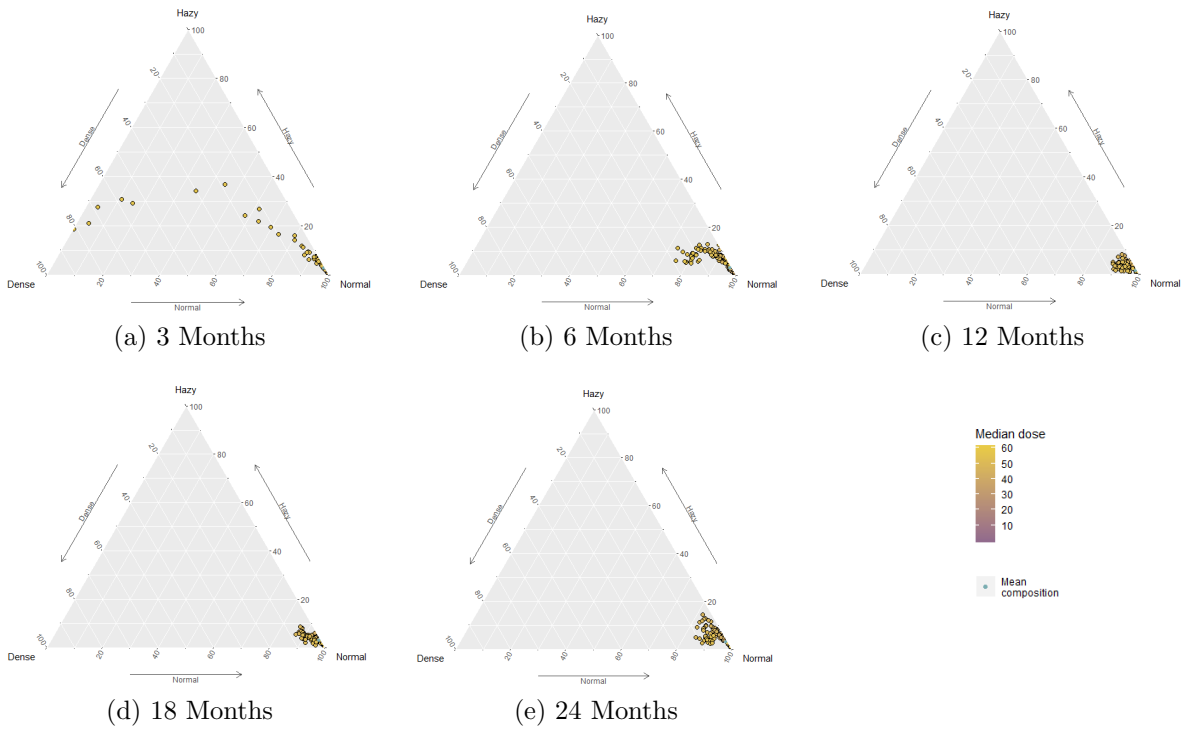


Figure 2.24: Ternary plots of dense and hazy composition by time after RT in the isodose-patch of 0.2 Gy for the reference patient

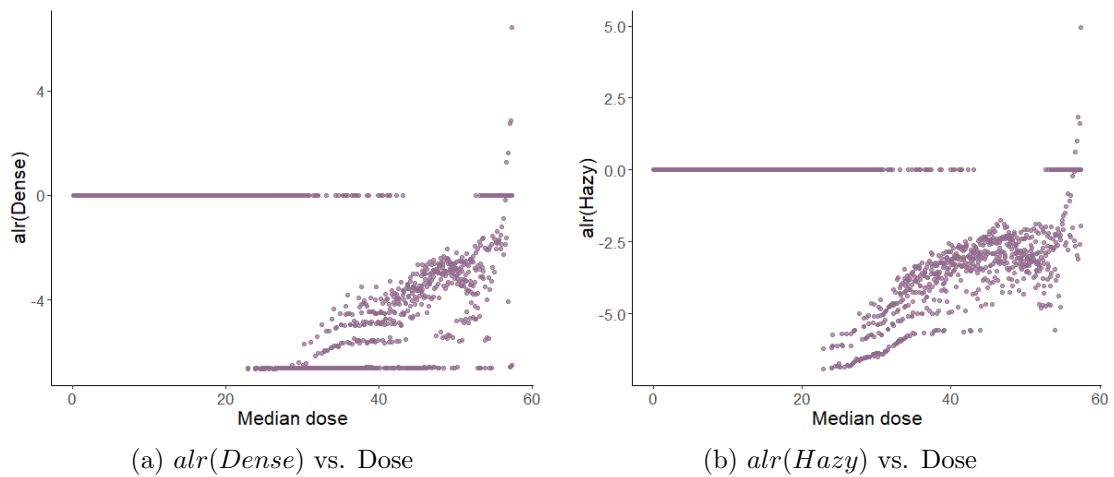


Figure 2.25: Scatter plot of alr -transformed dense and hazy vs. median dose per patch in the isodose-patch of 0.2 Gy for the reference patient

for predicting or determining the future course of action. Generally, two approaches using the CT scan for the study of RILD following RT have been implemented. Still, both methodologies present important limitations to characterize the local changes after radiation therapy.

Consequently, we proposed three different approaches to create small groups of voxels and study the changes in RILD at the patch level. The cubic-patch combines adjacent voxels along each direction in cuboid shape. The spherical-patch finds the voxel with the highest dose and combine equidistant voxels into the same patch. Finally, the isodose-patch merges voxels that received a very similar amount RT.

In the cubic-patch approach, patch size eight was selected for carrying further analysis given a combination between its size and the dose homogeneity within patches. There is only one proposal for the spherical-patch approach. For the isodose-patch approach, five different interval lengths were assessed, all of them with high dose homogeneity. The interval length of 0.2 Gy was selected for further analysis.

In addition to the shape, the three approaches to patch definition present several differences. First, the number of patches per patient. While the cubic-patch results on thousands of patches for analysis, the spherical-patch and the isodose-patch approaches result in 200 to 400 patches. A small number of patches when analyzing the CT scans of one patient might result in a more efficient and faster way to analysis RILD over time. However, the reduction in the number of patches came from an increase in the number of voxels within patches. In the cubic-patch size 8, most of the patches combine 512 voxels, but border patches are composed of fewer voxels. However, this does not represent a problem because the patches located on the border usually receive none or a minimal RT dose and do not present any RILD. On the other hand, the number of voxels within a patch for the two other patch definitions varies from as few as five voxels to as many as millions. The number of voxels within a patch might become an essential issue in the analysis of these two patch definitions since we are going to use the percentage of dense and hazy regions as primary outcomes.

Therefore, patches with a large number of voxels can result in tiny proportions of dense and hazy damage.

Although the cubic-patch approach shows an interesting property giving the consistency on the number of voxels within a patch, giving a large number of patches for analysis in that approach, the percentage of patches with RILD is less than 3.00% for all the analyzed subjects. Even though in the other two patch definition, the rate of RILD is larger as compared with the observed in the cubic-patch, the distribution of RILD still heavily biased towards the no presence of RILD. Therefore, when analyzing data from different approaches, it is crucial to have a statistical model that can handle the heavily-biased distribution of RILD.

The distribution of the proportion of dense and hazy, for all patch definitions, showed a bimodal distribution. This might be suggesting two different types of patches, response to RT dose, or different locations on the lung. Whatever the reason for this finding, the present pattern must be taken into account in the analysis to produce an adequate study of RILD over time.

Finally, it is important to highlight that for all the presented patch approaches, the patches present spatial correlation with each other. For example, in the cubic-patch, neighbors patches are correlated and this correlation should decrease as the distance between the patches increases. Then, this characteristic of the data structure will play an important role when analyzing the CT scan over time.

3. Two-part mixed effects mixture model for zero-inflated longitudinal compositional data

Let \mathbf{Y}_{ij} a semicontinuous bivariate vector for the i -th ($i = 1, \dots, N$) patch at time t_{ij} ($j = 1, \dots, n_i$). Then, this outcome vector can be represented by two processes, an occurrence variable U_{ij} and an intensity vector \mathbf{V}_{ij} . Then,

$$U_{ij} = \begin{cases} 1 & \text{if } \mathbf{Y}_{ij} \neq \mathbf{0} \\ 0 & \text{otherwise} \end{cases}$$

and the intensity variable $\mathbf{V}_{ij} = \log(\mathbf{Y}_{ij})$, when $\mathbf{Y}_{ij} \neq \mathbf{0}$. The log transformation has been used to make \mathbf{V}_{ij} approximately normal with a patch-time-specific mean, and at the same time, handle compositional data.

Now the focus is on the distribution for the occurrence variable U_{ij} , and the conditional distribution of \mathbf{V}_{ij} . Let us assume that U_{ij} follows a random effects logistic regression model:

$$\begin{aligned} \text{logit}\{Pr(U_{ij} = 1|\mathbf{c}_i)\} &= \text{logit}\{\pi_{ij}|\mathbf{c}_i\} \\ &= \eta_{ij} \\ &= \mathbf{X}_{ij}\boldsymbol{\beta} + \mathbf{Z}_{ij}\mathbf{c}_i \end{aligned} \tag{3.3}$$

where \mathbf{X}_{ij} is a $1 \times p$ covariate vector for the fixed effects, $\boldsymbol{\beta}$ is a $p \times 1$ fixed effects regression coefficient vector, \mathbf{Z}_{ij} is a $1 \times q$ covariate vector for the random effects, and \mathbf{c}_i is a $q \times 1$ patch-level vector of random intercept and random effects.

Further, the vector $\mathbf{V}_{ij}|\mathbf{Y}_{ij} \neq \mathbf{0}$ follows a bivariate linear mixed model:

$$\mathbf{V}_{ij} | (\mathbf{d}_i, \mathbf{Y}_{ij} \neq 0) = \mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i + \boldsymbol{\epsilon}_{ij} \quad (3.4)$$

where \mathbf{X}_{ij}^* and \mathbf{Z}_{ij}^* might have the same variables as \mathbf{X}_{ij} and \mathbf{Z}_{ij} in (3.3), but this is not required; $\boldsymbol{\gamma}$ is the regression coefficient matrix ($p^* \times 2$) for the fixed effects; and \mathbf{d}_i is the subject-level matrix with the random effects ($q^* \times 2$). Additionally, the error term

$$\boldsymbol{\epsilon}_{ij} \sim N \left(\mathbf{0}, \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_{Y1}^2 & \rho \sigma_{Y1} \sigma_{Y2} \\ \rho \sigma_{Y1} \sigma_{Y2} & \sigma_{Y2}^2 \end{pmatrix} \right) \quad (3.5)$$

The random effects vector $(\mathbf{c}_i, \tilde{\mathbf{d}}_i)$, where $\tilde{\mathbf{d}}_i = \text{vec}(\mathbf{d}_i)$, is distributed by a mixture of 2 multivariate normal distributions. Correlation between random effects in (3.3) and (3.4) is allowed since the presence or absence of tissue damage at one time point is related with the intensity of the damage, if any, at that and other time points. Then,

$$\mathbf{b}_i = \begin{bmatrix} \mathbf{c}_i \\ \tilde{\mathbf{d}}_i \end{bmatrix} \sim \sum_{R=1}^2 m_R N \left(\boldsymbol{\mu}_R = \begin{pmatrix} \mathbf{0}_{q \times q} \\ \boldsymbol{\mu}_r \end{pmatrix}, \boldsymbol{\Psi} = \begin{pmatrix} \boldsymbol{\Psi}_{cc} & \boldsymbol{\Psi}_{cd} \\ \boldsymbol{\Psi}_{dc} & \boldsymbol{\Psi}_{dd} \end{pmatrix} \right) \quad (3.6)$$

with $\sum_{r=1}^2 m_r = 1$ and $\sum_{r=1}^2 m_r \boldsymbol{\mu}_r = \mathbf{0}$.

Now, we define $\Delta_i = 1$ if \mathbf{b}_i is sampled from the first component in the mixture, and 0 otherwise. Then,

$$\mathbf{c}_i \sim N(\mathbf{0}, \boldsymbol{\Psi}_{cc}) \quad (3.7)$$

and

$$\tilde{\mathbf{d}}_i | \mathbf{c}_i \sim [m_1 N(\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i, \mathbf{H})]^{\Delta_i} [(1 - m_1) N(\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i, \mathbf{H})]^{1 - \Delta_i} \quad (3.8)$$

where $\mathbf{H} = \boldsymbol{\Psi}_{dd} - \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{cd}$, and $\boldsymbol{\mu}_2 = -\frac{m_1}{1 - m_1} \boldsymbol{\mu}_1$.

3.1 The likelihood

The estimation of the parameters in the model defined by (3.3) - (3.8) is based on the maximization of the likelihood

$$\begin{aligned}
L &\propto \prod_{i=1}^N \int_{\mathbf{c}_i} \int_{\tilde{\mathbf{d}}_i} f(\mathbf{y}_i | \boldsymbol{\beta}, \boldsymbol{\gamma}, \mathbf{c}_i, \tilde{\mathbf{d}}_i, \Sigma) f(\mathbf{c}_i, \tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \boldsymbol{\Psi}) d\mathbf{c}_i d\tilde{\mathbf{d}}_i \quad (3.9) \\
&\quad \text{assuming } \mathbf{Y}_i \text{ are } i.i.d: \\
&= \prod_{i=1}^N \int_{\mathbf{c}_i} \int_{\tilde{\mathbf{d}}_i} \prod_{j=1}^{n_i} f(\mathbf{y}_{ij} | \boldsymbol{\beta}, \boldsymbol{\gamma}, \mathbf{c}_i, \tilde{\mathbf{d}}_i, \Sigma) f(\mathbf{c}_i, \tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \boldsymbol{\Psi}) d\mathbf{c}_i d\tilde{\mathbf{d}}_i \\
&\quad \text{with the two-parts model and using conditional distribution for random effects:} \\
&= \prod_{i=1}^N \int_{\mathbf{c}_i} \int_{\tilde{\mathbf{d}}_i} \prod_{j=1}^{n_i} (1 - \pi_{ij})^{(1-U_{ij})} \pi_{ij}^{U_{ij}} [f(\mathbf{V}_{ij} | \boldsymbol{\gamma}, \tilde{\mathbf{d}}_i, \Sigma)]^{U_{ij}} \times f(\mathbf{c}_i | \boldsymbol{\Psi}_{cc}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\mathbf{c}_i d\tilde{\mathbf{d}}_i \\
&\quad \text{re-expressing internal product:} \\
&= \prod_{i=1}^N \int_{\mathbf{c}_i} \int_{\tilde{\mathbf{d}}_i} \prod_{j=1}^{n_i} (1 - \pi_{ij})^{(1-U_{ij})} \pi_{ij}^{U_{ij}} \prod_{j=1}^{n_i} [f(\mathbf{V}_{ij} | \boldsymbol{\gamma}, \tilde{\mathbf{d}}_i, \Sigma)]^{U_{ij}} f(\mathbf{c}_i | \boldsymbol{\Psi}_{cc}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\mathbf{c}_i d\tilde{\mathbf{d}}_i \\
&\quad \text{using that } \exp(\log(x)) = x: \\
&= \prod_{i=1}^N \int_{\mathbf{c}_i} \int_{\tilde{\mathbf{d}}_i} \exp\left(\log\left(\prod_{j=1}^{n_i} (1 - \pi_{ij})^{(1-U_{ij})} \pi_{ij}^{U_{ij}}\right)\right) \times \exp\left(\log\left(\prod_{j=1}^{n_i} [f(\mathbf{V}_{ij} | \boldsymbol{\gamma}, \tilde{\mathbf{d}}_i, \Sigma)]^{U_{ij}}\right)\right) \\
&\quad \times f(\mathbf{c}_i | \boldsymbol{\Psi}_{cc}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\mathbf{c}_i d\tilde{\mathbf{d}}_i \\
&= \prod_{i=1}^N \int_{\mathbf{c}_i} \int_{\tilde{\mathbf{d}}_i} \exp\left(\sum_{j=1}^{n_i} \left((1 - U_{ij}) \log(1 - \pi_{ij}) + U_{ij} \log(\pi_{ij})\right)\right) \\
&\quad \times \exp\left(\sum_{j=1}^{n_i} U_{ij} \left[-\log(2\pi) - \frac{1}{2} \log(|\Sigma|) - \frac{1}{2} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i)) \Sigma^{-1} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i))^T\right]\right) \\
&\quad \times f(\mathbf{c}_i | \boldsymbol{\Psi}_{cc}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\mathbf{c}_i d\tilde{\mathbf{d}}_i \\
&\quad \text{Let } n_i^* \text{ the number of non-zero vectors in } \mathbf{y}_i \\
&= \prod_{i=1}^N \int_{\mathbf{c}_i} \int_{\tilde{\mathbf{d}}_i} \times \exp\left(\sum_{j=1}^{n_i} \left(U_{ij} \log\left(\frac{\pi_{ij}}{(1 - \pi_{ij})}\right) + \log(1 - \pi_{ij})\right)\right)
\end{aligned}$$

$$\begin{aligned}
& \times \exp\left(-n_i^* \log(2\pi) - \frac{n_i^*}{2} \log(|\Sigma|)\right. \\
& \quad \left. - \frac{1}{2} \sum_{j=1}^{n_i} U_{ij} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i)) \Sigma^{-1} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i))^T\right) \\
& \quad \times f(\mathbf{c}_i | \boldsymbol{\Psi}_{cc}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\mathbf{c}_i d\tilde{\mathbf{d}}_i \\
= & \prod_{i=1}^N \int_{\mathbf{c}_i} \int_{\tilde{\mathbf{d}}_i} \exp\left(\sum_{j=1}^{n_i} (U_{ij} \eta_{ij} + \log(1 - \pi_{ij}))\right) \\
& \quad \times \exp\left(-n_i^* \log(2\pi) - \frac{n_i^*}{2} \log(|\Sigma|)\right. \\
& \quad \left. - \frac{1}{2} \sum_{j=1}^{n_i} U_{ij} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i)) \Sigma^{-1} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i))^T\right) \\
& \quad \times f(\mathbf{c}_i | \boldsymbol{\Psi}_{cc}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\mathbf{c}_i d\tilde{\mathbf{d}}_i \\
= & \prod_{i=1}^N \int_{\mathbf{c}_i} \int_{\tilde{\mathbf{d}}_i} \exp(l_{U_{ij}}) \times \exp(l_{V_{ij}}) \times f(\mathbf{c}_i | \boldsymbol{\Psi}_{cc}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\mathbf{c}_i d\tilde{\mathbf{d}}_i \\
= & \prod_{i=1}^N \int_{\mathbf{c}_i} \exp(l_{U_{ij}}) f(\mathbf{c}_i | \boldsymbol{\Psi}_{cc}) \left(\int_{\tilde{\mathbf{d}}_i} \exp(l_{V_{ij}}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\tilde{\mathbf{d}}_i \right) d\mathbf{c}_i \tag{3.10}
\end{aligned}$$

Then, working with the inner integral in (3.10)

$$\begin{aligned}
& \int_{\tilde{\mathbf{d}}_i} \exp(l_{V_{ij}}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\tilde{\mathbf{d}}_i \\
= & \int_{\tilde{\mathbf{d}}_i} \exp(l_{V_{ij}}) [m_1 N(\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i, \mathbf{H})]^{\Delta_i} [(1 - m_1) N(\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i, \mathbf{H})]^{1 - \Delta_i} d\tilde{\mathbf{d}}_i \\
= & m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} \int_{\tilde{\mathbf{d}}_i} \exp(l_{V_{ij}}) N(\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i, \mathbf{H})^{\Delta_i} N(\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i, \mathbf{H})^{1 - \Delta_i} d\tilde{\mathbf{d}}_i \\
= & m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-q^*} |H|^{-\frac{1}{2}} \int_{\tilde{\mathbf{d}}_i} \exp\left(-n_i^* \log(2\pi) - \frac{n_i^*}{2} \log(|\Sigma|)\right. \\
& \quad \left. - \frac{1}{2} \sum_{j=1}^{n_i} U_{ij} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i)) \Sigma^{-1} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i))^T\right) \\
& \quad \times \exp\left(-\Delta_i \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i))^T \mathbf{H}^{-1} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)))\right)\right)
\end{aligned}$$

$$\begin{aligned}
& \times \exp\left(- (1 - \Delta_i) \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i))^T \mathbf{H}^{-1} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)) \right)\right) d\tilde{\mathbf{d}}_i \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-q^*} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} \\
& \times \int_{\tilde{\mathbf{d}}_i} \exp\left(\frac{1}{2} \sum_{j=1}^{n_i} U_{ij} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i)) \Sigma^{-1} (\mathbf{V}_{ij} - (\mathbf{X}_{ij}^* \boldsymbol{\gamma} + \mathbf{Z}_{ij}^* \mathbf{d}_i))^T\right) \\
& \times \exp\left(- \Delta_i \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i))^T \mathbf{H}^{-1} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)) \right)\right) \\
& \times \exp\left(- (1 - \Delta_i) \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i))^T \mathbf{H}^{-1} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)) \right)\right) d\tilde{\mathbf{d}}_i
\end{aligned}$$

Let i^* identify the group of observations $\mathbf{y}_{ij} \neq \mathbf{0}$ in \mathbf{y}_i

$$\begin{aligned}
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-q^* - n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} \\
& \times \int_{\tilde{\mathbf{d}}_i} \exp\left(-\frac{1}{2} (\tilde{\mathbf{V}}_{i^*} - (\tilde{\mathbf{X}}_{i^*}^* \tilde{\boldsymbol{\gamma}} + \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i))^T \tilde{\Sigma}^{-1} (\tilde{\mathbf{V}}_{i^*} - (\tilde{\mathbf{X}}_{i^*}^* \tilde{\boldsymbol{\gamma}} + \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i))\right) \\
& \times \exp\left(- \Delta_i \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i))^T \mathbf{H}^{-1} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)) \right)\right) \\
& \times \exp\left(- (1 - \Delta_i) \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i))^T \mathbf{H}^{-1} (\tilde{\mathbf{d}}_i - (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)) \right)\right) d\tilde{\mathbf{d}}_i
\end{aligned}$$

where $\tilde{\mathbf{V}}_{i^*} = \text{vec}(\mathbf{V}_{i^*})$, $\tilde{\boldsymbol{\gamma}} = \text{vec}(\boldsymbol{\gamma})$, $\tilde{\mathbf{X}}_{i^*}^* = I_{2 \times 2} \otimes \mathbf{X}_{i^*}^*$, $\tilde{\mathbf{Z}}_{i^*}^* = I_{2 \times 2} \otimes \mathbf{Z}_{i^*}^*$, and $\tilde{\Sigma}^{-1} = \Sigma^{-1} \otimes I_{n_{i^*} \times n_{i^*}}$

Let $\mathbf{A}_{i^*} = \tilde{\mathbf{V}}_{i^*} - \tilde{\mathbf{X}}_{i^*}^* \tilde{\boldsymbol{\gamma}}$; $\mathbf{f}_{i1} = \boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i$, and $\mathbf{f}_{i2} = \boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i$. Then,

$$\begin{aligned}
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-q^* - n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} \int_{\tilde{\mathbf{d}}_i} \exp\left(-\frac{1}{2} (\mathbf{A}_{i^*} - \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i)^T \tilde{\Sigma}^{-1} (\mathbf{A}_{i^*} - \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i)\right) \\
& \times \exp\left(- \Delta_i \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i - \mathbf{f}_{i1})^T \mathbf{H}^{-1} (\tilde{\mathbf{d}}_i - \mathbf{f}_{i1}) \right)\right) \\
& \times \exp\left(- (1 - \Delta_i) \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i - \mathbf{f}_{i2})^T \mathbf{H}^{-1} (\tilde{\mathbf{d}}_i - \mathbf{f}_{i2}) \right)\right) d\tilde{\mathbf{d}}_i \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-q^* - n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} \int_{\tilde{\mathbf{d}}_i} \exp\left(-\frac{1}{2} (\mathbf{A}_{i^*}^T - \tilde{\mathbf{d}}_i^T \tilde{\mathbf{Z}}_{i^*}^{*T}) (\tilde{\Sigma}^{-1} \mathbf{A}_{i^*} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i)\right) \\
& \times \exp\left(- \Delta_i \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i^T - \mathbf{f}_{i1}^T) (\mathbf{H}^{-1} \tilde{\mathbf{d}}_i - \mathbf{H}^{-1} \mathbf{f}_{i1}) \right)\right) \\
& \times \exp\left(- (1 - \Delta_i) \left(-\frac{1}{2} (\tilde{\mathbf{d}}_i^T - \mathbf{f}_{i2}^T) (\mathbf{H}^{-1} \tilde{\mathbf{d}}_i - \mathbf{H}^{-1} \mathbf{f}_{i2}) \right)\right) d\tilde{\mathbf{d}}_i \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + q^*)} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}}
\end{aligned}$$

$$\begin{aligned}
& \times \int_{\tilde{\mathbf{d}}_i} \exp \left(-\frac{1}{2} (\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} - \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i - \tilde{\mathbf{d}}_i^T \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} + \tilde{\mathbf{d}}_i^T \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i) \right. \\
& \quad - \frac{1}{2} \Delta_i (\tilde{\mathbf{d}}_i^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i - \tilde{\mathbf{d}}_i^T \mathbf{H}^{-1} \mathbf{f}_{i1} - \mathbf{f}_{i1}^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i + \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{f}_{i1}) \\
& \quad \left. - \frac{1}{2} (1 - \Delta_i) (\tilde{\mathbf{d}}_i^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i - \tilde{\mathbf{d}}_i^T \mathbf{H}^{-1} \mathbf{f}_{i2} - \mathbf{f}_{i2}^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i + \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{f}_{i2}) \right) d\tilde{\mathbf{d}}_i \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + q^*)} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} \\
& \quad \times \int_{\tilde{\mathbf{d}}_i} \exp \left(-\frac{1}{2} (\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} - 2\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i + \tilde{\mathbf{d}}_i^T \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i) \right. \\
& \quad - \frac{1}{2} \Delta_i (\tilde{\mathbf{d}}_i^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i - 2\mathbf{f}_{i1}^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i + \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{f}_{i1}) \\
& \quad \left. - \frac{1}{2} (1 - \Delta_i) (\tilde{\mathbf{d}}_i^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i - 2\mathbf{f}_{i2}^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i + \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{f}_{i2}) \right) d\tilde{\mathbf{d}}_i \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + q^*)} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} \\
& \quad \times \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} - \frac{1}{2} \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{f}_{i1} - \frac{1}{2} (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{f}_{i2} \right) \\
& \quad \times \int_{\tilde{\mathbf{d}}_i} \exp \left(\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i + \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i + (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i \right. \\
& \quad \left. - \frac{1}{2} (1 - \Delta_i) \tilde{\mathbf{d}}_i^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i - \frac{1}{2} \tilde{\mathbf{d}}_i^T \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \tilde{\mathbf{d}}_i - \frac{1}{2} \Delta_i \tilde{\mathbf{d}}_i^T \mathbf{H}^{-1} \tilde{\mathbf{d}}_i \right) d\tilde{\mathbf{d}}_i \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + q^*)} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} \\
& \quad \times \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} - \frac{1}{2} \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{f}_{i1} - \frac{1}{2} (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{f}_{i2} \right) \\
& \quad \times \int_{\tilde{\mathbf{d}}_i} \exp \left((\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* + \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} + (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1}) \tilde{\mathbf{d}}_i - \frac{1}{2} \tilde{\mathbf{d}}_i^T (\mathbf{H}^{-1} + \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^*) \tilde{\mathbf{d}}_i \right) d\tilde{\mathbf{d}}_i \\
\text{Let } \mathbf{B}_i & = \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* + \mathbf{H}^{-1} \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + q^*)} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} \\
& \quad \times \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} - \frac{1}{2} \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{f}_{i1} - \frac{1}{2} (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{f}_{i2} \right) \\
& \quad \times \int_{\tilde{\mathbf{d}}_i} \exp \left((\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* + \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} + (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1}) \tilde{\mathbf{d}}_i - \frac{1}{2} \tilde{\mathbf{d}}_i^T \mathbf{B}_i \tilde{\mathbf{d}}_i \right) d\tilde{\mathbf{d}}_i
\end{aligned}$$

$$\begin{aligned}
&= m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + q^*)} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} (2\pi)^{q^*} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \\
&\quad \times \exp\left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} - \frac{1}{2} \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{f}_{i1} - \frac{1}{2} (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{f}_{i2}\right) \\
&\quad \times \exp\left(\frac{1}{2} (\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* + \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} + (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1}) \mathbf{B}_i^{-1}\right. \\
&\quad \quad \left. \times (\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* + \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} + (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1})^T\right) \\
&= m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \\
&\quad \times \exp\left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} - \frac{1}{2} \Delta_i (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)^T \mathbf{H}^{-1} (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)\right. \\
&\quad \quad \left. - \frac{1}{2} (1 - \Delta_i) (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)^T \mathbf{H}^{-1} (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)\right) \\
&\quad \times \exp\left(\frac{1}{2} (\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* + \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} + (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1})\right. \\
&\quad \quad \left. \times (\mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} + \Delta_i \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i1} + (1 - \Delta_i) \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i2})\right) \\
&= m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \\
&\quad \times \exp\left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} - \frac{1}{2} \Delta_i (\boldsymbol{\mu}_1^T + \mathbf{c}_i^T \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{dc}^T) (\mathbf{H}^{-1} \boldsymbol{\mu}_1 + \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)\right. \\
&\quad \quad \left. - \frac{1}{2} (1 - \Delta_i) (\boldsymbol{\mu}_2^T + \mathbf{c}_i^T \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{dc}^T) (\mathbf{H}^{-1} \boldsymbol{\mu}_2 + \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)\right) \\
&\quad \times \exp\left(\frac{1}{2} (\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} + \Delta_i \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i1}\right. \\
&\quad \quad + (1 - \Delta_i) \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i2} + \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} \\
&\quad \quad + \Delta_i^2 \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i1} + (1 - \Delta_i) \Delta_i \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i2} \\
&\quad \quad + (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} + \Delta_i (1 - \Delta_i) \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i1} \\
&\quad \quad \left. + (1 - \Delta_i)^2 \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i2})\right) \\
&= m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \exp\left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*}\right)
\end{aligned}$$

$$\begin{aligned}
& -\frac{1}{2}\Delta_i(\boldsymbol{\mu}_1^T \mathbf{H}^{-1} \boldsymbol{\mu}_1 + 2\boldsymbol{\mu}_1^T \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i + \mathbf{c}_i^T \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{dc}^T \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i) \\
& -\frac{1}{2}(1 - \Delta_i)(\boldsymbol{\mu}_2^T \mathbf{H}^{-1} \boldsymbol{\mu}_2 + 2\boldsymbol{\mu}_2^T \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i + \mathbf{c}_i^T \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{dc}^T \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i) \Big) \\
& \times \exp \left(\frac{1}{2} (\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} + 2\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} (\Delta_i \mathbf{f}_{i1} + (1 - \Delta_i) \mathbf{f}_{i2}) \right. \\
& \quad \left. + \Delta_i^2 \mathbf{f}_{i1}^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i1} + (1 - \Delta_i)^2 \mathbf{f}_{i2}^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{f}_{i2}) \right) \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} \right. \\
& \quad \left. + \frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} \right) \exp \left(-\frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \boldsymbol{\mu}_1 - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \boldsymbol{\mu}_2 \right. \\
& \quad \left. - (\Delta_i \boldsymbol{\mu}_1^T + (1 - \Delta_i) \boldsymbol{\mu}_2^T) \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i - \frac{1}{2} \mathbf{c}_i^T \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{dc}^T \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i \right) \\
& \times \exp \left(\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} (\Delta_i (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i) + (1 - \Delta_i) (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)) \right. \\
& \quad \left. + \frac{1}{2} \Delta_i^2 (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} (\boldsymbol{\mu}_1 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i) \right. \\
& \quad \left. + \frac{1}{2} (1 - \Delta_i)^2 (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i)^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} (\boldsymbol{\mu}_2 + \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i) \right) \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} \right. \\
& \quad \left. + \frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} \right) \exp \left(\Delta_i \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \boldsymbol{\mu}_1 - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \boldsymbol{\mu}_1 \right. \\
& \quad \left. + \frac{1}{2} \Delta_i^2 \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \boldsymbol{\mu}_1 + (1 - \Delta_i) \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \boldsymbol{\mu}_2 - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \boldsymbol{\mu}_2 \right. \\
& \quad \left. + \frac{1}{2} (1 - \Delta_i)^2 \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \boldsymbol{\mu}_2 + \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i \right. \\
& \quad \left. - (\Delta_i \boldsymbol{\mu}_1^T + (1 - \Delta_i) \boldsymbol{\mu}_2^T) \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i \right. \\
& \quad \left. + (\Delta_i^2 \boldsymbol{\mu}_1^T + (1 - \Delta_i)^2 \boldsymbol{\mu}_2^T) \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i - \frac{1}{2} \mathbf{c}_i^T \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{dc}^T \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i \right. \\
& \quad \left. + \frac{1}{2} (\Delta_i^2 + (1 - \Delta_i)^2) \mathbf{c}_i^T \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{dc}^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i \right) \\
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} \right.
\end{aligned}$$

$$\begin{aligned}
& + \frac{1}{2} \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \mathbf{A}_{i^*} \Big) \exp \left(\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} (\Delta_i \boldsymbol{\mu}_1 + (1 - \Delta_i) \boldsymbol{\mu}_2) \right. \\
& - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T (\mathbf{H}^{-1} - \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1}) \boldsymbol{\mu}_1 - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T (\mathbf{H}^{-1} - \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1}) \boldsymbol{\mu}_2 \\
& + \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i \\
& - (\Delta_i \boldsymbol{\mu}_1^T + (1 - \Delta_i) \boldsymbol{\mu}_2^T) (\mathbf{H}^{-1} - \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1}) \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i \\
& \left. - \frac{1}{2} \mathbf{c}_i^T \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{dc}^T (\mathbf{H}^{-1} - \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1}) \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i \right)
\end{aligned}$$

Note that $\mathbf{H}^{-1} - \mathbf{H}^{-1} \mathbf{B}_i^{-1} \mathbf{H}^{-1} = \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^*$

Let $\mathbf{E}_i^T = \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1}$, $\mathbf{M}_i^T = \Delta_i \boldsymbol{\mu}_1^T + (1 - \Delta_i) \boldsymbol{\mu}_2^T$,

and $\mathbf{D}_i = \boldsymbol{\Psi}_{cc}^{-1} \boldsymbol{\Psi}_{dc}^T (\mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^*) \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1}$. Then,

$$\begin{aligned}
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*} \right) \\
& \times \exp \left(\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1 \right. \\
& - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2 + \mathbf{E}_i^T \mathbf{c}_i \\
& \left. - \mathbf{M}_i^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1} \mathbf{c}_i - \frac{1}{2} \mathbf{c}_i^T \mathbf{D}_i \mathbf{c}_i \right)
\end{aligned}$$

Let $\mathcal{G}_i^T = \mathbf{E}_i^T - \mathbf{M}_i^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\Psi}_{dc} \boldsymbol{\Psi}_{cc}^{-1}$

$$\begin{aligned}
& = m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*} \right) \\
& \times \exp \left(\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1 \right. \\
& \left. - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2 + \mathcal{G}_i^T \mathbf{c}_i - \frac{1}{2} \mathbf{c}_i^T \mathbf{D}_i \mathbf{c}_i \right) \tag{3.11}
\end{aligned}$$

Now, going back to the integral in (3.10):

$$\int_{\mathbf{c}_i} \exp(l_{U_{ij}}) f(\mathbf{c}_i | \boldsymbol{\Psi}_{cc}) \left(\int_{\tilde{\mathbf{d}}_i} \exp(l_{V_{ij}}) f(\tilde{\mathbf{d}}_i | m_1, \boldsymbol{\mu}_1, \mathbf{c}_i, \boldsymbol{\Psi}_{dd}) d\tilde{\mathbf{d}}_i \right) d\mathbf{c}_i$$

$$\begin{aligned}
&= m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-n_i^*} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} \exp\left(-\frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*}\right) \\
&\quad \times \exp\left(\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1\right. \\
&\quad \quad \left. - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2\right) \\
&\quad \times \int_{\mathbf{c}_i} \exp(l_{U_{ij}}) f(\mathbf{c}_i | \Psi_{cc}) \exp\left(\mathcal{G}_i^T \mathbf{c}_i - \frac{1}{2} \mathbf{c}_i^T \mathbf{D}_i \mathbf{c}_i\right) d\mathbf{c}_i \\
&= m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + \frac{q}{2})} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} |\Psi_{cc}|^{-\frac{1}{2}} \\
&\quad \times \exp\left(-\frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*}\right) \\
&\quad \times \exp\left(\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1\right. \\
&\quad \quad \left. - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2\right) \\
&\quad \times \int_{\mathbf{c}_i} \exp\left(l_{U_{ij}} - \frac{1}{2} \mathbf{c}_i^T \Psi_{cc}^{-1} \mathbf{c}_i + \mathcal{G}_i^T \mathbf{c}_i - \frac{1}{2} \mathbf{c}_i^T \mathbf{D}_i \mathbf{c}_i\right) d\mathbf{c}_i \\
&= m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + \frac{q}{2})} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} |\Psi_{cc}|^{-\frac{1}{2}} \\
&\quad \times \exp\left(-\frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*}\right) \\
&\quad \times \exp\left(\mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1\right. \\
&\quad \quad \left. - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2\right) \\
&\quad \times \int_{\mathbf{c}_i} \exp\left(l_{U_{ij}} - \frac{1}{2} \mathbf{c}_i^T (\Psi_{cc}^{-1} + \mathbf{D}_i) \mathbf{c}_i + \mathcal{G}_i^T \mathbf{c}_i\right) d\mathbf{c}_i \\
&= m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + \frac{q}{2})} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} |\Psi_{cc}|^{-\frac{1}{2}} \\
&\quad \times \exp\left(-\frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*} + \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i\right. \\
&\quad \quad - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1 - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2 \\
&\quad \quad \left. + \frac{1}{2} \mathcal{G}_i^T (\Psi_{cc}^{-1} + \mathbf{D}_i)^{-1} \mathcal{G}_i\right)
\end{aligned}$$

$$\times \int_{\mathbf{c}_i} \exp\left(l_{U_{ij}} - \frac{1}{2}(\mathbf{c}_i - (\boldsymbol{\Psi}_{cc}^{-1} + \mathbf{D}_i)^{-1}\mathcal{G}_i)^T (\boldsymbol{\Psi}_{cc}^{-1} + \mathbf{D}_i)(\mathbf{c}_i - (\boldsymbol{\Psi}_{cc}^{-1} + \mathbf{D}_i)^{-1}\mathcal{G}_i)\right) d\mathbf{c}_i \quad (3.12)$$

The integral in (3.12) is intractable, but it is similar to the likelihood in a mixed effects logistic regression where the random effects follow a normal distribution with different location and scale parameters. As discussed for different authors, several strategies have been implemented to evaluate this integral. Using Olsen and Schafer⁴⁴ approach with a sixth-order Laplace approximation, we have that:

$$\begin{aligned} & \int_{\mathbf{c}_i} \exp(f(\mathbf{c}_i)) d\mathbf{c}_i \\ &= \int_{\mathbf{c}_i} \exp\left(l_{U_{ij}} - \frac{1}{2}(\mathbf{c}_i - (\boldsymbol{\Psi}_{cc}^{-1} + \mathbf{D}_i)^{-1}\mathcal{G}_i)^T (\boldsymbol{\Psi}_{cc}^{-1} + \mathbf{D}_i)(\mathbf{c}_i - (\boldsymbol{\Psi}_{cc}^{-1} + \mathbf{D}_i)^{-1}\mathcal{G}_i)\right) d\mathbf{c}_i \\ &\approx \exp(f(\tilde{\mathbf{c}}_i)) | -[f^{(2)}(\tilde{\mathbf{c}}_i)]^{-1}|^{\frac{1}{2}} \left(1 - \frac{1}{8} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(4)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^2 - \frac{1}{48} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(6)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^3 \right. \\ &\quad \left. + \frac{15}{72} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right)^T \mathbf{G}_i^{-1} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right) \right) \quad (3.13) \end{aligned}$$

where:

- $\tilde{\mathbf{c}}_i = (\boldsymbol{\Psi}_{cc}^{-1} + \mathbf{D}_i + \mathbf{Z}_i^T \widetilde{\mathbf{W}}_i \mathbf{Z}_i)^{-1} (\mathbf{Z}_i^T \widetilde{\mathbf{W}}_i (\mathbf{U}_i^* - \mathbf{X}_i \boldsymbol{\beta}) + \mathcal{G}_i)$
- $\mathbf{U}_i^* = \widetilde{\mathbf{W}}_i^{-1} (\mathbf{U}_i - \tilde{\boldsymbol{\pi}}_i) + \tilde{\boldsymbol{\eta}}_i$
- \mathbf{W}_i is a diagonal matrix with elements $\pi_{ij}(1 - \pi_{ij})$
- $\widetilde{\mathbf{W}}_i$, $\widetilde{\mathbf{W}}_i^{-1}$, $\tilde{\boldsymbol{\pi}}_i$, and $\tilde{\boldsymbol{\eta}}_i$ are evaluated at $\mathbf{c}_i = \tilde{\mathbf{c}}_i$
- $-f^{(2)}(\tilde{\mathbf{c}}_i) = \mathbf{Z}_i^T \widetilde{\mathbf{W}}_i \mathbf{Z}_i + \boldsymbol{\Psi}_{cc}^{-1} + \mathbf{D}_i = \mathbf{G}_i$
- $\tilde{m}_{ij}^{(k)}$ is the $(k - 1)$ th derivative of π_{ij} with respect to η_{ij} evaluated at $\tilde{\mathbf{c}}_i$

Even though more terms can be added to the Laplace approximation for greater accuracy, the omitted terms on the sixth-order approximation are $O_p(n_i^{-5/2})$ or smaller.⁴⁴

Now, going back to (3.12),

$$\begin{aligned}
& m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + \frac{q}{2})} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} |\Psi_{cc}|^{-\frac{1}{2}} \\
& \times \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*} + \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i \right. \\
& \quad - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1 - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2 \\
& \quad \left. + \frac{1}{2} \mathcal{G}_i^T (\Psi_{cc}^{-1} + \mathbf{D}_i)^{-1} \mathcal{G}_i \right) \\
& \times \int_{\mathbf{c}_i} \exp \left(l_{U_{ij}} - \frac{1}{2} (\mathbf{c}_i - (\Psi_{cc}^{-1} + \mathbf{D}_i)^{-1} \mathcal{G}_i)^T (\Psi_{cc}^{-1} + \mathbf{D}_i) (\mathbf{c}_i - (\Psi_{cc}^{-1} + \mathbf{D}_i)^{-1} \mathcal{G}_i) \right) d\mathbf{c}_i \\
& \approx m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + \frac{q}{2})} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} |\Psi_{cc}|^{-\frac{1}{2}} \\
& \times \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*} + \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i \right. \\
& \quad - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1 - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2 \\
& \quad \left. + \frac{1}{2} \mathcal{G}_i^T (\Psi_{cc}^{-1} + \mathbf{D}_i)^{-1} \mathcal{G}_i \right) \\
& \times \exp(f(\tilde{\mathbf{c}}_i)) - [f^{(2)}(\tilde{\mathbf{c}}_i)]^{-1} \Big|^{1/2} \left(1 - \frac{1}{8} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(4)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^2 - \frac{1}{48} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(6)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^3 \right. \\
& \quad \left. + \frac{15}{72} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right)^T \mathbf{G}_i^{-1} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right) \right) \quad (3.14)
\end{aligned}$$

Then,

$$\begin{aligned}
L & \approx \prod_{i=1}^N m_1^{\Delta_i} (1 - m_1)^{1 - \Delta_i} (2\pi)^{-(n_i^* + \frac{q}{2})} |\Sigma|^{-\frac{n_i^*}{2}} |\mathbf{H}|^{-\frac{1}{2}} |\mathbf{B}_i^{-1}|^{\frac{1}{2}} |\Psi_{cc}|^{-\frac{1}{2}} \\
& \times \exp \left(-\frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*} + \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i \right. \\
& \quad \left. - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1 - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2 \right)
\end{aligned}$$

$$\begin{aligned}
& + \frac{1}{2} \mathcal{G}_i^T (\Psi_{cc}^{-1} + \mathbf{D}_i)^{-1} \mathcal{G}_i \Big) \\
& \times \exp(f(\tilde{\mathbf{c}}_i)) | - [f^{(2)}(\tilde{\mathbf{c}}_i)]^{-1} |^{\frac{1}{2}} \left(1 - \frac{1}{8} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(4)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^2 - \frac{1}{48} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(6)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^3 \right. \\
& \left. + \frac{15}{72} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right)^T \mathbf{G}_i^{-1} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right) \right) \quad (3.15)
\end{aligned}$$

Finally, the approximate log-likelihood is:

$$\begin{aligned}
l & \approx \sum_{i=1}^N \left[\Delta_i \log(m_1) + (1 - \Delta_i) \log(1 - m_1) - \left(n_i^* + \frac{q}{2} \right) \log(2\pi) - \frac{n_i^*}{2} \log|\Sigma| - \frac{1}{2} \log|\mathbf{H}| + \frac{1}{2} \log|\mathbf{B}_i^{-1}| \right. \\
& - \frac{1}{2} \log|\Psi_{cc}| - \frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*} + \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i \\
& - \frac{1}{2} \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1 - \frac{1}{2} (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2 \\
& + \frac{1}{2} \mathcal{G}_i^T (\Psi_{cc}^{-1} + \mathbf{D}_i)^{-1} \mathcal{G}_i + f(\tilde{\mathbf{c}}_i) + \frac{1}{2} \log| - [f^{(2)}(\tilde{\mathbf{c}}_i)] + \log \left(1 - \frac{1}{8} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(4)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^2 \right. \\
& \left. - \frac{1}{48} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(6)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^3 + \frac{15}{72} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right)^T \mathbf{G}_i^{-1} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right) \right) \Big] \\
& \approx \hat{N} \log(m_1) + (N - \hat{N}) \log(1 - m_1) - \left(N^* + \frac{Nq}{2} \right) \log(2\pi) - \frac{N^*}{2} \log|\Sigma| - \frac{N}{2} \log|\mathbf{H}| - \frac{N}{2} \log|\Psi_{cc}| \\
& + \frac{1}{2} \sum_{i=1}^N \log|\mathbf{B}_i^{-1}| - \frac{1}{2} \sum_{i=1}^N \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*} + \sum_{i=1}^N \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} \mathbf{M}_i \\
& - \frac{1}{2} \sum_{i=1}^N \Delta_i \boldsymbol{\mu}_1^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_1 - \frac{1}{2} \sum_{i=1}^N (1 - \Delta_i) \boldsymbol{\mu}_2^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_2 \\
& + \frac{1}{2} \sum_{i=1}^N \mathcal{G}_i^T (\Psi_{cc}^{-1} + \mathbf{D}_i)^{-1} \mathcal{G}_i + \sum_{i=1}^N f(\tilde{\mathbf{c}}_i) + \frac{1}{2} \sum_{i=1}^N \log| - [f^{(2)}(\tilde{\mathbf{c}}_i)] + \sum_{i=1}^N \log P_i \quad (3.16)
\end{aligned}$$

where

- $\hat{N} = \sum_{i=1}^N \Delta_i$
- $N^* = \sum_{i=1}^N n_i^*$
- $P_i = 1 - \frac{1}{8} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(4)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^2 - \frac{1}{48} \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(6)} (\mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T)^3 + \frac{15}{72} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right)^T \mathbf{G}_i^{-1} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ij}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_i^{-1} \mathbf{Z}_{ij}^T \right)$

3.2 Model estimation

First, the expectation–maximization (EM) algorithm is implemented. In the E-step, the conditional expectation of the likelihood is obtained, and only the posterior probability to belong to the first component of the mixture for each i -th patch has to be calculated as:

$$m_{1i}(\boldsymbol{\theta}^{(t)}) = \frac{\exp(S_{1i})}{\sum_{r=1}^2 \exp(S_{ri})} \quad (3.17)$$

where

$$\begin{aligned} S_{ri} = & -\left(n_i^* + \frac{q}{2}\right) \log(2\pi) - \frac{n_i^*}{2} \log|\Sigma| - \frac{1}{2} \log|\mathbf{H}| + \frac{1}{2} \log|\mathbf{B}_i^{-1}| - \frac{1}{2} \log|\Psi_{cc}| \\ & - \frac{1}{2} \mathbf{A}_{i^*}^T (\tilde{\Sigma}^{-1} - \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1}) \mathbf{A}_{i^*} + \mathbf{A}_{i^*}^T \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \mathbf{B}_i^{-1} \mathbf{H}^{-1} m_r \boldsymbol{\mu}_r^T \\ & - \frac{1}{2} m_r \boldsymbol{\mu}_r^T \mathbf{H}^{-1} \mathbf{B}_i^{-1} \tilde{\mathbf{Z}}_{i^*}^{*T} \tilde{\Sigma}^{-1} \tilde{\mathbf{Z}}_{i^*}^* \boldsymbol{\mu}_r + \frac{1}{2} \mathcal{G}_{ir}^T (\Psi_{cc}^{-1} + \mathbf{D}_i)^{-1} \mathcal{G}_{ir} + f(\tilde{\mathbf{c}}_{ir}) + \frac{1}{2} \log| - [f^{(2)}(\tilde{\mathbf{c}}_{ir})] \\ & + \log\left(1 - \frac{1}{8} \sum_{j=1}^{n_i} \tilde{m}_{ijr}^{(4)} (\mathbf{Z}_{ij} \mathbf{G}_{ir}^{-1} \mathbf{Z}_{ij}^T)^2 - \frac{1}{48} \sum_{j=1}^{n_i} \tilde{m}_{ijr}^{(6)} (\mathbf{Z}_{ij} \mathbf{G}_{ir}^{-1} \mathbf{Z}_{ij}^T)^3 \right. \\ & \left. + \frac{15}{72} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ijr}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_{ir}^{-1} \mathbf{Z}_{ij}^T \right)^T \mathbf{G}_{ir}^{-1} \left(\sum_{j=1}^{n_i} \mathbf{Z}_{ij}^T \tilde{m}_{ijr}^{(3)} \mathbf{Z}_{ij} \mathbf{G}_{ir}^{-1} \mathbf{Z}_{ij}^T \right) \right) \end{aligned} \quad (3.18)$$

In the M-step, the conditional expectation is maximized with respect of m_1 , the probability of belonging to the first component of the mixture, which allow us to obtained an updated estimate $m_1^{(t+1)}$ as:

$$m_1^{(t+1)} = \frac{1}{N} \sum_{i=1}^N m_{1i}(\boldsymbol{\theta}^{(t)}) \quad (3.19)$$

However, the other parameters in the model cannot be updated analytically. Then, an

approximate Fisher scoring numerical maximization procedure is implemented. Score vectors were derived using the proposed score vectors by Olsen and Schafer⁴⁴, but ϕ_{cc} , ϕ_H , and ϕ_Σ represent the vectorized upper triangles of Ψ_{cc} , H , and Σ . Additionally, score vectors were extended to accommodate multivariate data. Expressions for the components of the score vector for the i -th patch are:

$$\begin{aligned}
S_{\beta_i} &= X_i^T \tilde{W}_i (U_i^* - X_i \beta - Z_i \tilde{c}_i) + \sum_{j=1}^{n_i} \left(-\frac{1}{2} \tilde{m}_{ij}^{(3)} (Z_{ij} G_i^{-1} Z_{ij}^T) \right) (X_{ij}^T - X_i^T \tilde{W}_i Z_i G_i^{-1} Z_{ij}^T) \\
&\quad + \frac{1}{P_i} \sum_{j=1}^{n_i} R_{ij}^* (X_{ij}^T - X_i^T \tilde{W}_i Z_i G_i^{-1} Z_{ij}^T) \tag{3.20}
\end{aligned}$$

$$\begin{aligned}
S_{\gamma_i} &= \tilde{X}_i^{*T} \tilde{\Sigma}^{-1} \tilde{V}_i - \tilde{X}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{V}_i - \tilde{X}_i^{*T} \tilde{\Sigma}^{-1} \tilde{X}_i^* \tilde{\gamma} + \tilde{X}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{X}_i^* \tilde{\gamma} \\
&\quad - \tilde{X}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc}^{-1} \tilde{c}_i + t_i^T E_{ij} \tag{3.21}
\end{aligned}$$

$$S_{\mu_{1i}} = -\frac{1}{2} \left(m_1 + \frac{m_1^2}{(1-m_1)} \right) (H^{-1} B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* + \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} H^{-1}) \mu_{1i} \tag{3.22}$$

$$\begin{aligned}
S_{\phi_\Sigma} &= \left(\frac{\partial \text{vec}(\Sigma^{-1})}{\partial \phi_\Sigma^T} \right)^T \left(\frac{n_i^*}{2} \text{vec}(\Sigma) \right) \\
&\quad + \left(\frac{\partial \text{vec}(\tilde{\Sigma}^{-1})}{\partial \phi_\Sigma^T} \right)^T \left(\frac{1}{2} \text{vec} \left(-\tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} - A_i A_i^T + 2A_i A_i^T \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \right. \right. \\
&\quad \left. \left. - \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} A_i A_i^T \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \right) - \frac{m_1}{2} \text{vec}(\tilde{Z}_i^* B_i^{-1} H^{-1} \mu_1 \mu_1^T \tilde{Z}_i^{*T} \right. \\
&\quad \left. - \tilde{Z}_i^* B_i^{-1} H^{-1} \mu_1 \mu_1^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \right) - \frac{1-m_1}{2} \text{vec}(\tilde{Z}_i^* B_i^{-1} H^{-1} \mu_2 \mu_2^T \tilde{Z}_i^{*T} \\
&\quad \left. - \tilde{Z}_i^* B_i^{-1} H^{-1} \mu_2 \mu_2^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \right) + P_i^{*T} E_{ij} - \frac{1}{2} [(\tilde{Z}_i^* B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc}^{-1} \otimes \tilde{Z}_i^* \Psi_{dc} \Psi_{cc}^{-1})
\end{aligned}$$

$$\begin{aligned}
& - (\tilde{Z}_i^* B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc_i}^{-1} \otimes \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* \Psi_{dc} \Psi_{cc_i}^{-1})] \text{vec}(G_i^{-1}) \\
& + \frac{1}{2} \text{vec} \left[\tilde{Z}_i^* B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc_i}^{-1} \tilde{c}_i \tilde{c}_i^T (\Psi_{dc} \Psi_{cc_i}^{-1})^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \right. \\
& - \tilde{Z}_i^* B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc_i}^{-1} \tilde{c}_i \tilde{c}_i^T (\Psi_{dc} \Psi_{cc_i}^{-1})^T \tilde{Z}_i^{*T} + 2A_i \tilde{c}_i^T (\Psi_{dc} \Psi_{cc_i}^{-1})^T H^{-1} B_i^{-1} \tilde{Z}_i^{*T} \\
& \left. - 2\tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} A_i \tilde{c}_i^T (\Psi_{dc} \Psi_{cc_i}^{-1})^T H^{-1} B_i^{-1} \tilde{Z}_i^{*T} \right] \\
& + \frac{1}{P_i} \left[\sum_{j=1}^{n_i} \text{vec}(\tilde{Z}_i^* B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc_i}^{-1} G_i^{-1} Z_{ij}^T Z_{ij} G_i^{-1} (\Psi_{dc} \Psi_{cc_i}^{-1})^T (\tilde{Z}_i^{*T} - \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T})) F_{ij}^* \right. \\
& \left. - \frac{15}{72} \text{vec}(\tilde{Z}_i^* B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc_i}^{-1} G_i^{-1} K_i K_i^T G_i^{-1} (\Psi_{dc} \Psi_{cc_i}^{-1})^T (\tilde{Z}_i^{*T} - \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T})) \right] \Bigg) \tag{3.23}
\end{aligned}$$

$$\begin{aligned}
S_{\phi_{cc_i}} & = \left(\frac{\partial \text{vec}(\Psi_{cc_i}^{-1})}{\partial \psi_{cc_i}^T} \right)^T \left(\frac{1}{2} \text{vec}(\Psi_{cc_i} - G_i^{-1} - \tilde{c}_i \tilde{c}_i^T) + D_i^{*T} E_{ij} + \frac{1}{P_i} \left(\sum_{j=1}^{n_i} \text{vec}(G_i^{-1} Z_{ij}^T Z_{ij} G_i^{-1}) F_{ij}^* \right. \right. \\
& \left. \left. - \frac{15}{72} \text{vec}(G_i^{-1} K_i K_i^T G_i^{-1}) \right) \right) \tag{3.24}
\end{aligned}$$

$$\begin{aligned}
S_{\Psi_{dc} \Psi_{cc_i}^{-1}} & = -(I_p \otimes H_i^*) (\text{vec}(G_i^{-1}) + \text{vec}(\tilde{c}_i \tilde{c}_i^T)) + \text{vec}(H^{-1} B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} A_i \tilde{c}_i^T) + S_i^{*T} E_{ij} \\
& + \frac{2}{P_i} \left(\sum_{j=1}^{n_i} \text{vec}(H_i^* G_i^{-1} Z_{ij}^T Z_{ij} G_i^{-1}) F_{ij}^* - \frac{15}{36} \text{vec}(H_i^* G_i^{-1} K_i K_i^T G_i^{-1}) \right) \tag{3.25}
\end{aligned}$$

$$\begin{aligned}
S_{\phi_{H_i}} & = \left(\frac{\partial \text{vec}(H^{-1})}{\partial \phi_H^T} \right)^T \left(-\frac{1}{2} ((B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* \Psi_{dc} \Psi_{cc_i}^{-1} \otimes \Psi_{dc} \Psi_{cc_i}^{-1}) \right. \\
& - (B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc_i}^{-1} \otimes B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* \Psi_{dc} \Psi_{cc_i}^{-1})) \text{vec}(G_i^{-1}) \\
& + \frac{1}{2} \text{vec}[H - B_i^{-1} - B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} A_i A_i^T \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} + m_1 B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* \mu_1 \mu_1^T H^{-1} B_i^{-1} \\
& - m_1 B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* \mu_1 \mu_1^T + (1 - m_1) B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* \mu_2 \mu_2^T H^{-1} B_i^{-1} \\
& \left. - (1 - m_1) B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* \mu_2 \mu_2^T - \Psi_{dc} \Psi_{cc_i}^{-1} \tilde{c}_i \tilde{c}_i^T (\Psi_{dc} \Psi_{cc_i}^{-1})^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \right)
\end{aligned}$$

$$\begin{aligned}
& + 2B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} A_i \tilde{c}_i^T (\Psi_{dc} \Psi_{cc}^{-1})^T + B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc}^{-1} \tilde{c}_i \tilde{c}_i^T (\Psi_{dc} \Psi_{cc}^{-1})^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \\
& - 2B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} A_i \tilde{c}_i^T (\Psi_{dc} \Psi_{cc}^{-1})^T H^{-1} B_i^{-1}] + M_i^{*T} E_{ij} \\
& + \frac{1}{P_i} \left[\sum_{j=1}^{n_i} \text{vec}((I_{2q^*} - B_i^{-1} H^{-1}) \Psi_{dc} \Psi_{cc}^{-1} G_i^{-1} Z_{ij}^T Z_{ij} G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1}) F_{ij}^* \right. \\
& \left. - \frac{15}{72} \text{vec}((I_{2q^*} - B_i^{-1} H^{-1}) \Psi_{dc} \Psi_{cc}^{-1} G_i^{-1} K_i K_i^T G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1}) \right] \Bigg)
\end{aligned} \tag{3.26}$$

where

$$\begin{aligned}
P_i^* & = (\partial \tilde{c}_i) / (\partial \text{vec}^T(\tilde{\Sigma}^{-1})) = (Y_i^{*T} G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \otimes G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T H^{-1} B_i^{-1} \tilde{Z}_i^{*T}) \\
& \quad - (Y_i^{*T} G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T \tilde{Z}_i^{*T} \otimes G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T H^{-1} B_i^{-1} \tilde{Z}_i^{*T}) \\
& \quad - (A_i^T \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \tilde{Z}_i^{*T} \otimes G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T H^{-1} B_i^{-1} \tilde{Z}_i^{*T}) \\
& \quad + (A_i^T \otimes G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T H^{-1} B_i^{-1} \tilde{Z}_i^{*T})
\end{aligned} \tag{3.27}$$

$$\begin{aligned}
M_i^* & = -(Y_i^{*T} G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \otimes G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T) + (A_i^T \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \otimes G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T) \\
& \quad - (A_i^T \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \otimes G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T H^{-1} B_i^{-1}) \\
& \quad + (Y_i^{*T} G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} \otimes G_i^{-1} (\Psi_{dc} \Psi_{cc}^{-1})^T H^{-1} B_i^{-1})
\end{aligned} \tag{3.28}$$

$$D_i^* = -(Y_i^{*T} G_i^{-1} \otimes G_i^{-1}) \tag{3.29}$$

$$t_i = -G_i^{-1}(\Psi_{dc}\Psi_{cc}^{-1})^T H^{-1} B_i^{-1} \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{X}_i^* \quad (3.30)$$

$$S_i^* = ((A_i^T \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} H^{-1} \otimes G_i^{-1})) - (G_i^{-1} \otimes Y_i^{*T} G_i^{-1} H_i^{*T}) - (Y_i^{*T} G_i^{-1} \otimes G_i^{-1} H_i^{*T}) \quad (3.31)$$

$$E_{ij} = E_i - (\Psi_{cc}^{-1} + D_i) \tilde{c}_i + \sum_{j=1}^{n_i} Z_{ij}^T \left(-\frac{1}{2} \tilde{m}_{ij}^{(3)} (Z_{ij} G_i^{-1} Z_{ij}^T) + (U_{ij} - \tilde{\pi}_{ij}) + \frac{1}{P_i} R_{ij}^* \right) \quad (3.32)$$

$$H_i^* = \tilde{Z}_i^{*T} \tilde{\Sigma}^{-1} \tilde{Z}_i^* B_i^{-1} H^{-1} \Psi_{dc} \Psi_{cc}^{-1} \quad (3.33)$$

$$Y_i^* = Z_i^T \tilde{W}_i (U_i^* - X_i \beta) + E_i \quad (3.34)$$

$$F_{ij}^* = \frac{1}{4} \tilde{m}_{ij}^{(4)} (Z_{ij} G_i^{-1} Z_{ij}^T) + \frac{1}{16} \tilde{m}_{ij}^{(6)} (Z_{ij} G_i^{-1} Z_{ij}^T)^2 - \frac{15}{36} \tilde{m}_{ij}^{(3)} (Z_{ij} G_i^{-1} K_i) \quad (3.35)$$

$$\begin{aligned} R_{ij}^* = & -\frac{1}{8} \tilde{m}_{ij}^{(5)} (Z_{ij} G_i^{-1} Z_{ij}^T)^2 + \frac{1}{4} \tilde{m}_{ij}^{(3)} Z_{ij} F_i Z_{ij}^T - \frac{1}{48} \tilde{m}_{ij}^{(7)} (Z_{ij} G_i^{-1} Z_{ij}^T)^3 + \frac{1}{16} \tilde{m}_{ij}^{(3)} Z_{ij} f_i Z_{ij}^T \\ & - \frac{15}{72} \tilde{m}_{ij}^{(3)} (Z_{ij} G_i^{-1} K_i)^2 + \frac{15}{36} \tilde{m}_{ij}^{(4)} (K_i^T G_i^{-1} Z_{ij}^T) (Z_{ij} G_i^{-1} Z_{ij}^T) - \frac{15}{36} \tilde{m}_{ij}^{(3)} Z_{ij} h_i Z_{ij}^T \end{aligned} \quad (3.36)$$

$$F_i = \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(4)} G_i^{-1} Z_{ij}^T (Z_{ij} G_i^{-1} Z_{ij}^T) Z_{ij} G_i^{-1} \quad (3.37)$$

$$f_i = \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(6)} G_i^{-1} Z_{ij}^T (Z_{ij} G_i^{-1} Z_{ij}^T)^2 Z_{ij} G_i^{-1} \quad (3.38)$$

$$h_i = \sum_{j=1}^{n_i} \tilde{m}_{ij}^{(3)} G_i^{-1} Z_{ij}^T Z_{ij} G_i^{-1} (Z_{ij} G_i^{-1} K_i) \quad (3.39)$$

and

$$K_i = \sum_{j=1}^{n_i} Z_{ij}^T \tilde{m}_{ij}^{(3)} Z_{ij} G_i^{-1} Z_{ij}^T \quad (3.40)$$

At each iteration, the values of the new estimates for the variance and correlation parameters are validated to assess whether the estimations are outside the parameter space. When needed, a step-halving procedure is applied to return the estimates within the parameter space. Also, a second step-halving procedure was implemented for all parameters in the model to guarantee that the deviance of the new set of estimates is lower than in the previous iteration.

Initial values for β , γ , Ψ_{cc} , and the diagonal in Ψ_{dd} are generated by logistic regression for the occurrence variable and independent linear mixed model for each variable in the intensity vector. Starting values for the out of diagonal in Ψ_{dd} are set using the diagonal values and a correlation of 0.1. Initial values for Σ are set as the variances and correlations of the variables in the intensity vector. For simplicity, the starting values for m_1 and μ_1 are set as 0.6 and $(-0.1, 0.1)$, respectively.

Approximate empirical Bayes estimates by importance sampling for unnormalized densities are implemented for the random effects. A thousand samples $(c_i^{(1)}, \dots, c_i^{(1000)})$ were taken from a multivariate t distribution with 4 degrees of freedom, centered at the mode \tilde{c}_i with covariance matrix proportional to G_i^{-1} , as proposed by Gelman et al.,⁴⁵ and implemented by Olsen and Shafer.⁴⁴ Unadjusted importance sampling, as well as truncated⁴⁶ and Pareto smoothed importance sampling⁴⁷ are implemented. Using the drawn sample for c_i , estimated means for c_i and d_i were calculated. All estimation procedures have been implemented in R version 3.4.4.⁴⁸

4. Model performance

Performance of the proposed method was evaluated using a simulation study. However, special care was taken to keep the simulation set up to be very similar to real data. Dose-response data were simulated based on the motivating example of RILD in lung cancer patients using the spherical-patch definition, presented in Chapter 2. The composition (%Dense, %Hazy, %Normal) was the outcome of interest (\mathbf{Y}_{ij}), for the i -th ($i = 1, \dots, N$) patch at time t_{ij} ($j = 1, \dots, n_i$). Then, as mentioned in Chapter 3, this outcome vector can be represented by two processes. First, an occurrence variable U_{ij} which equals one if some RILD is present (Normal $\neq 1$), or zero otherwise. The second process is the bivariate intensity vector \mathbf{V}_{ij} , which is the *alr* transformation of the composition, when $\mathbf{Y}_{ij} \neq 0$.

Three different number of patches ($N = 300, 500, 1000$) were used. Also, three different numbers of follow-up measurements after RT were considered ($n_i = 3, 4, 5$). The received dose was simulated following a logarithmic distribution. For both parts of the model (occurrence and intensity), intercept, dose-effect, and time after radiotherapy in years (linear and quadratic) were considered as fixed effects to build the data. As a random effects, only a random intercept was used in both model parts (c_i and \mathbf{d}_i for occurrence and intensity models respectively).

The following are the population parameters used in the simulation:

$$\begin{aligned}
\text{logit}\{Pr(U_{ij} = 1|\mathbf{c}_i)\} &= -10 + 1.5 \text{Dose}_i - 2 \text{time}_{ij} - 3 \text{time}_{ij}^2 + \mathbf{c}_i \\
\text{alr}(\text{Dense})_{ij} | (\mathbf{d}_i, \mathbf{Y}_{ij} \neq 0) &= -7 + 1 \text{Dose}_i + 0.5 \text{time}_{ij} - 1 \text{time}_{ij}^2 + \mathbf{d}_{1i} + \boldsymbol{\epsilon}_{1ij} \\
\text{alr}(\text{Hazy})_{ij} | (\mathbf{d}_i, \mathbf{Y}_{ij} \neq 0) &= -3 + 2 \text{Dose}_i + 0.5 \text{time}_{ij} - 1 \text{time}_{ij}^2 + \mathbf{d}_{2i} + \boldsymbol{\epsilon}_{2ij}
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{\epsilon}_{ij} &\sim N \left(\mathbf{0}, \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.14 \\ 0.14 & 0.5 \end{pmatrix} \right) \\
\mathbf{b}_i = \begin{bmatrix} \mathbf{c}_i \\ \tilde{\mathbf{d}}_{1i} \\ \tilde{\mathbf{d}}_{2i} \end{bmatrix} &\sim 0.6N \left(\begin{pmatrix} 0 \\ -0.1 \\ 0.1 \end{pmatrix}, \boldsymbol{\Psi} \right) + 0.4N \left(\begin{pmatrix} 0 \\ 0.15 \\ -0.15 \end{pmatrix}, \boldsymbol{\Psi} \right) \\
\boldsymbol{\Psi} &= \begin{pmatrix} 1 & 0.07 & 0.20 \\ 0.07 & 0.50 & 0.07 \\ 0.20 & 0.07 & 1 \end{pmatrix}
\end{aligned}$$

A total of 1000 data sets were generated with the above parameters and within each combination of sample size and follow-up time points. Subsequently, the scoring procedure for the estimation of the two-part mixed-effects mixture model for zero-inflated longitudinal compositional data, presented in Chapter 3, was applied to each data set. Estimates for all parameters, regression coefficients and nuisance parameters, were obtained. The performance of the algorithm in the estimation of the regression coefficients was assessed by two statistics, which were computed for each parameter estimate: the standardized bias (SB) (difference between the average estimate and the true value as a percentage of the SD estimate), and the coverage rate of the nominal 95% confidence interval (estimate $\pm 1.96\text{SE}$). As suggested by Collins et al.,⁴⁹ SB larger than 40% was considered as an indicator of poor performance. Although the nominal coverage percentage is 95%, coverage rate lower than 90% was considered troublesome.⁴⁹ Additionally, the square of the correlation between the

observed and predicted data, as well as the mean absolute scaled error (MASE), were calculated for each simulation. As a comparison method, the square of the correlation and MASE were calculated for the "naive" model, using the initial values obtained by a mixed-effects logistic regression for the occurrence variable and independent linear mixed model for each variable in the intensity vector without correlation between the models.

Additionally, to assess the model's false positive rate of the fixed effects, the same simulation scenario described above was performed but with the coefficients of dose, time, and time squared set equal to zero. Parameter estimates and respective standard errors were calculated using the proposed model, and 95% confidence intervals were obtained. Type I error was calculated as the proportion of intervals that did not contain zero.

4.1 Results

For all the simulation scenarios, the procedure failed to converge in less than 5 samples (Figure 4.1 (a)) and for those that the procedure converged, the average number of iterations for convergence was between 11 to 14 steps (Figure 4.1 (b)). The average time estimating fixed effects was less than 10 min with 300 patches ($N = 300$), between 10 and 20 min with $N = 500$, and between 35 min and 50 min with $N = 1000$ (Figure 4.1 (c)). The estimation of the random effects took longer, around 45 min with $N = 300$, 1 to 1.5 hours with $N = 500$, and more than 2.5 hours with $N = 1000$ (Figure 4.1 (d)).

4.1.1 Performance of estimated fixed effects

The results of the simulation for the fixed effects are shown in Table 4.1, and a set of plots showing these results are presented in Figure 4.2. For each scenario, Table 4.1 list the average and the standard deviation (SD) of the point estimates, the average standard error (SE), the standardized bias (SB), and the coverage rate. In the occurrence model, it seems that the coefficients for the intercept and dose are underestimated, particularly with $N = 300$. In the intensity model, for both $alr(Dense)$ and $alr(Hazy)$, the estimation of the parameters

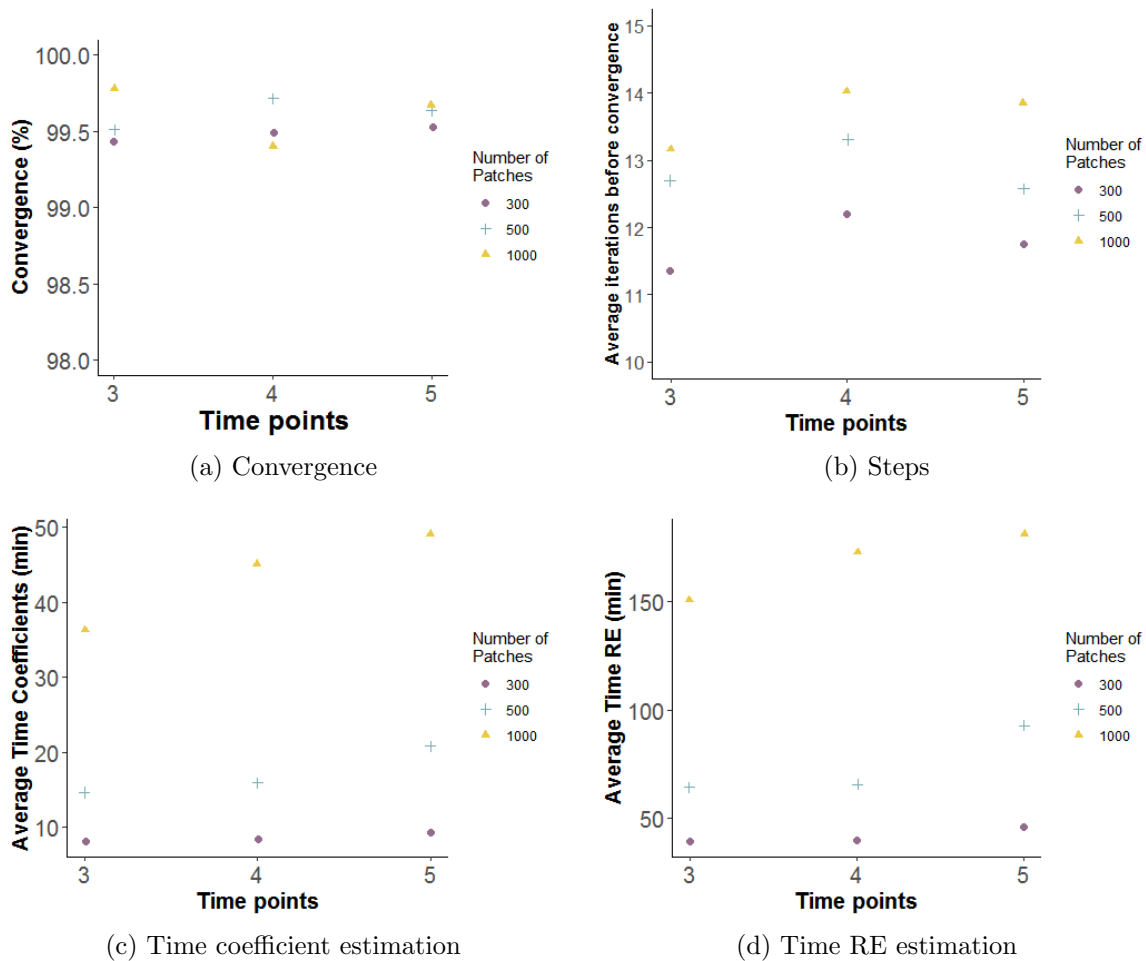


Figure 4.1: Simulation Results: Percentage of convergence, average number of iteration before convergence, average time for estimation of fixed effects (min), and average time for estimation of random effects (RE)

is quite good, even in small sample sizes. The number of time points seems to have a substantial effect on the performance of the estimates in occurrence model, with estimation bias decreasing as the number of time points increases.

Summary of the standardized bias (SB) demonstrates that it is consistently lower than 40% for all parameters, therefore bias impact on the efficiency, coverage and error rate with the proposed model is minimal. However, in some scenarios, the SB is larger than 20%, particularly for the estimation of the intercepts and the dose effect. In most of the cases, the average SE is close to the SD of the estimate; however, with $N = 300$, the average SE and the SD differ substantially. This might indicate that the estimation of the standard

Population	Occurrence				alr(Dense)				alr(Hazy)			
	Int. -10	Dose 1.5	Time -2	Time2 -3	Int. -7	Dose 1	Time 0.5	Time2 -1	Int. -3	Dose 2	Time 0.5	Time2 -1
N=1000, t=5												
Average estimate	-10.38	1.46	-1.93	-3.19	-6.98	1.00	0.49	-0.99	-2.96	1.99	0.48	-0.99
SD estimate	1.32	0.27	1.73	1.34	0.23	0.02	0.52	0.26	0.19	0.02	0.33	0.17
Average SE	1.01	0.07	1.93	0.96	0.22	0.01	0.50	0.30	0.19	0.01	0.33	0.19
Standardized bias (%)	-29.12	-15.86	4.28	-14.50	7.56	-16.86	-2.50	2.24	21.26	-27.98	-6.45	6.20
Coverage (%)	87.26	54.26	97.79	95.69	89.77	61.99	88.97	87.96	91.07	63.89	88.26	87.96
N=1000, t=4												
Average estimate	-10.29	1.44	-2.09	-3.04	-7.00	0.99	0.50	-1.00	-2.97	1.99	0.50	-1.00
SD estimate	1.17	0.24	1.79	1.14	0.26	0.05	0.69	0.42	0.20	0.02	0.46	0.28
Average SE	1.34	0.13	2.65	1.57	0.21	0.01	0.57	0.35	0.18	0.01	0.39	0.24
Standardized bias (%)	-24.29	-25.28	-5.05	-3.59	0.66	-11.38	-0.02	1.14	17.41	-28.00	-0.71	1.27
Coverage (%)	93.76	72.13	98.69	98.09	85.71	63.08	88.13	87.73	89.03	63.58	86.72	86.02
N=1000, t=3												
Average estimate	-10.36	1.44	-1.96	-3.17	-6.97	1.00	0.43	-0.94	-2.95	1.99	0.47	-0.98
SD estimate	1.49	0.26	3.40	2.79	0.37	0.02	1.37	1.08	0.29	0.02	0.96	0.76
Average SE	2.32	0.17	6.10	4.65	0.30	0.01	1.10	0.87	0.23	0.01	0.73	0.58
Standardized bias (%)	-24.01	-23.08	1.10	-6.01	8.00	-13.21	-5.14	5.12	18.28	-21.25	-2.87	3.21
Coverage (%)	99.70	80.06	99.70	99.90	85.77	60.42	87.07	86.37	84.67	60.92	84.27	83.67
N=500, t=5												
Average estimate	-10.52	1.50	-2.12	-3.15	-6.99	1.00	0.49	-1.00	-2.97	1.99	0.52	-1.01
SD estimate	1.69	0.31	1.98	1.10	0.23	0.03	0.56	0.27	0.21	0.03	0.42	0.21
Average SE	1.27	0.08	2.45	1.27	0.25	0.01	0.57	0.31	0.23	0.01	0.39	0.21
Standardized bias (%)	-31.08	-1.12	-6.19	-13.63	3.07	-5.79	-1.34	0.96	12.73	-21.99	3.58	-3.26
Coverage (%)	86.45	51.71	98.19	95.28	91.47	68.17	90.56	92.17	90.56	68.98	86.04	87.65
N=500, t=4												
Average estimate	-10.58	1.50	-2.01	-3.24	-6.98	1.00	0.48	-0.99	-2.96	2.00	0.48	-0.98
SD estimate	1.85	0.30	2.42	1.65	0.29	0.02	0.82	0.49	0.24	0.02	0.58	0.34
Average SE	2.02	0.18	4.13	2.62	0.28	0.01	0.80	0.50	0.24	0.01	0.53	0.34
Standardized bias (%)	-31.51	1.63	-0.34	-14.86	6.89	-13.57	-2.55	2.17	16.12	-17.04	-3.11	4.48
Coverage (%)	94.98	74.22	99.30	98.29	88.67	65.60	90.67	91.37	91.37	69.01	88.37	88.16
N=500, t=3												
Average estimate	-10.70	1.51	-2.13	-3.18	-7.00	1.00	0.48	-0.98	-2.99	2.00	0.53	-1.02
SD estimate	2.82	0.46	4.78	3.69	0.42	0.02	1.58	1.24	0.35	0.02	1.23	0.97
Average SE	3.35	0.26	8.47	6.43	0.39	0.01	1.47	1.17	0.31	0.01	1.01	0.79
Standardized bias (%)	-24.72	2.67	-2.63	-4.96	-0.06	-7.74	-0.97	1.61	3.79	-20.72	2.39	-2.44
Coverage (%)	99.60	86.83	99.30	99.30	90.55	65.03	91.06	91.16	89.75	68.44	87.04	87.04

Table 4.1: Simulation results for estimated fixed effects

errors generated by the proposed method may not adequately represent the variation of the estimator in small sample sizes. Coverage percentages were close to 90% or larger for all parameters but the dose coefficient. Contrary to our common experience, relatively higher coverage rates were observed with the smallest sample size ($N = 300$). This is due to large standard errors resulting from small sample size, not better estimation performance.

The false positive rate of the fixed effects model is presented in Table 4.2. The type I error rate for the coefficient associated with dose was equal or less than 0.05 for both model

Population	Occurrence				alr(Dense)				alr(Hazy)			
	Int. -10	Dose 1.5	Time -2	Time2 -3	Int. -7	Dose 1	Time 0.5	Time2 -1	Int. -3	Dose 2	Time 0.5	Time2 -1
N=300, t=5												
Average estimate	-11.23	1.14	-2.30	-3.70	-6.99	1.00	0.52	-1.01	-2.99	2.00	0.53	-1.01
SD estimate	7.97	9.19	6.74	8.47	0.28	0.03	0.62	0.30	0.24	0.03	0.43	0.20
Average SE	1.61	0.09	3.06	1.49	0.30	0.01	0.69	0.37	0.27	0.02	0.47	0.25
Standardized bias (%)	-15.47	-3.90	-4.51	-8.28	3.22	-1.78	3.21	-3.31	4.65	-9.14	6.66	-6.91
Coverage (%)	88.14	49.05	97.99	95.98	91.56	64.82	92.76	92.96	93.87	70.05	92.56	92.56
N=300, t=4												
Average estimate	-11.41	0.97	-2.74	-3.69	-6.99	1.00	0.50	-1.00	-2.96	2.00	0.51	-1.01
SD estimate	14.11	12.18	12.04	10.56	0.31	0.02	0.88	0.53	0.26	0.02	0.63	0.38
Average SE	2.26	0.20	4.15	2.47	0.33	0.01	0.92	0.57	0.27	0.01	0.62	0.39
Standardized bias (%)	-9.97	-4.32	-6.14	-6.50	4.36	-5.79	-0.31	0.21	15.94	-15.68	2.10	-1.62
Coverage (%)	95.18	74.17	98.99	98.69	93.57	68.54	93.87	93.87	90.85	71.66	89.45	90.65
N=300, t=3												
Average estimate	-13.22	0.34	-3.38	-4.12	-7.01	1.00	0.60	-1.08	-2.97	2.00	0.46	-0.96
SD estimate	25.30	19.33	17.48	11.20	0.51	0.03	1.88	1.48	0.38	0.03	1.38	1.08
Average SE	4.81	0.35	12.08	9.11	0.50	0.02	1.89	1.50	0.38	0.02	1.26	1.00
Standardized bias (%)	-12.71	-6.02	-7.88	-9.97	-2.43	-11.28	5.19	-5.22	8.91	-17.41	-2.99	3.70
Coverage (%)	98.09	90.24	98.09	97.79	92.25	67.10	92.45	92.45	91.95	69.11	89.84	90.74

Table 4.1: Simulation results for estimated fixed effects (cont.)

parts for all sample sizes (N) and the number of follow-up time points (t). The coefficient for time showed the highest type I error among the model coefficients, particularly for the first part of the model (occurrence model). This issue is a result of the underestimation of that coefficient, which gets worse as the sample size (N) increases. For the coefficient associated with Time², the type I error rate is larger than 0.05 for most of the scenarios. Additionally, an increase in the error is observed with larger sample sizes and number of time points; this given to the increase in the precision of the estimates as the number of observations increases.

The average squared correlation between observed and predicted values was at least 0.70 for the proposed two-part model, and it was less than 0.40 in the naive approach (Figure 4.5). Similarly, the Two-part model also showed better performance than the naive model when comparing the MASE, with an average MASE lower than 0.71 and 0.58 for dense and hazy, respectively, in the two-part model, and larger than 0.90 in the naive model (Figure 4.6).

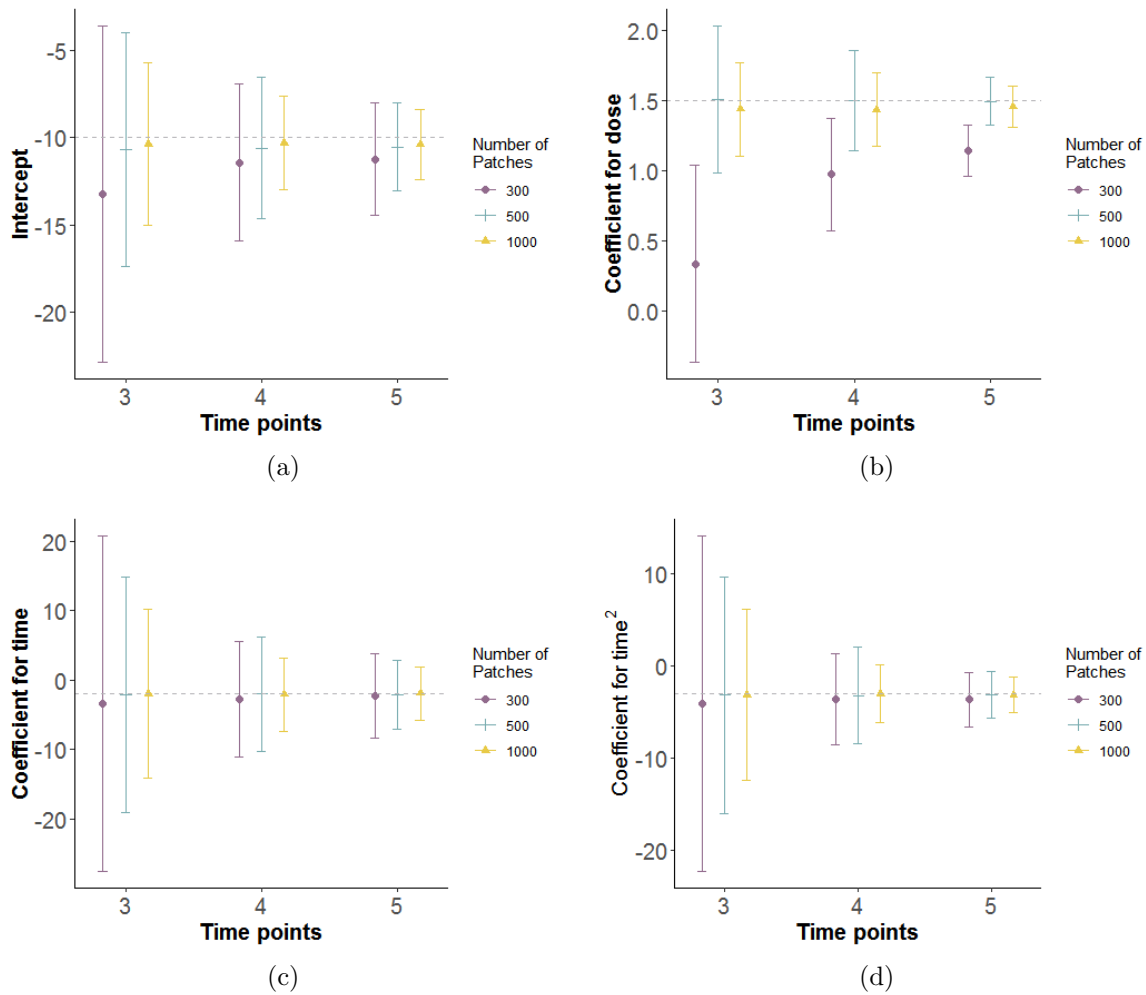


Figure 4.2: Simulation Results for estimated fixed effects: average estimate of fixed effects in the occurrence model ± 1.96 average standard error

N	t	Dose			Time			Time ²		
		Occurrence	Dense	Hazy	Occurrence	Dense	Hazy	Occurrence	Dense	Hazy
1000	5	0.03	0.04	0.05	0.62	0.08	0.11	0.22	0.11	0.12
1000	4	0.02	0.05	0.03	0.42	0.08	0.11	0.03	0.09	0.13
1000	3	0.02	0.05	0.04	0.02	0.11	0.14	0.01	0.11	0.14
500	5	0.02	0.04	0.05	0.87	0.06	0.07	0.13	0.06	0.10
500	4	0.03	0.05	0.04	0.50	0.06	0.11	0.03	0.08	0.11
500	3	0.01	0.03	0.04	0.03	0.08	0.12	0.01	0.09	0.11
300	5	0.02	0.05	0.04	0.84	0.04	0.06	0.09	0.06	0.09
300	4	0.03	0.05	0.05	0.27	0.06	0.06	0.03	0.07	0.09
300	3	0.02	0.04	0.04	0.01	0.06	0.08	0.02	0.08	0.09

Table 4.2: Type I error for estimated fixed effects

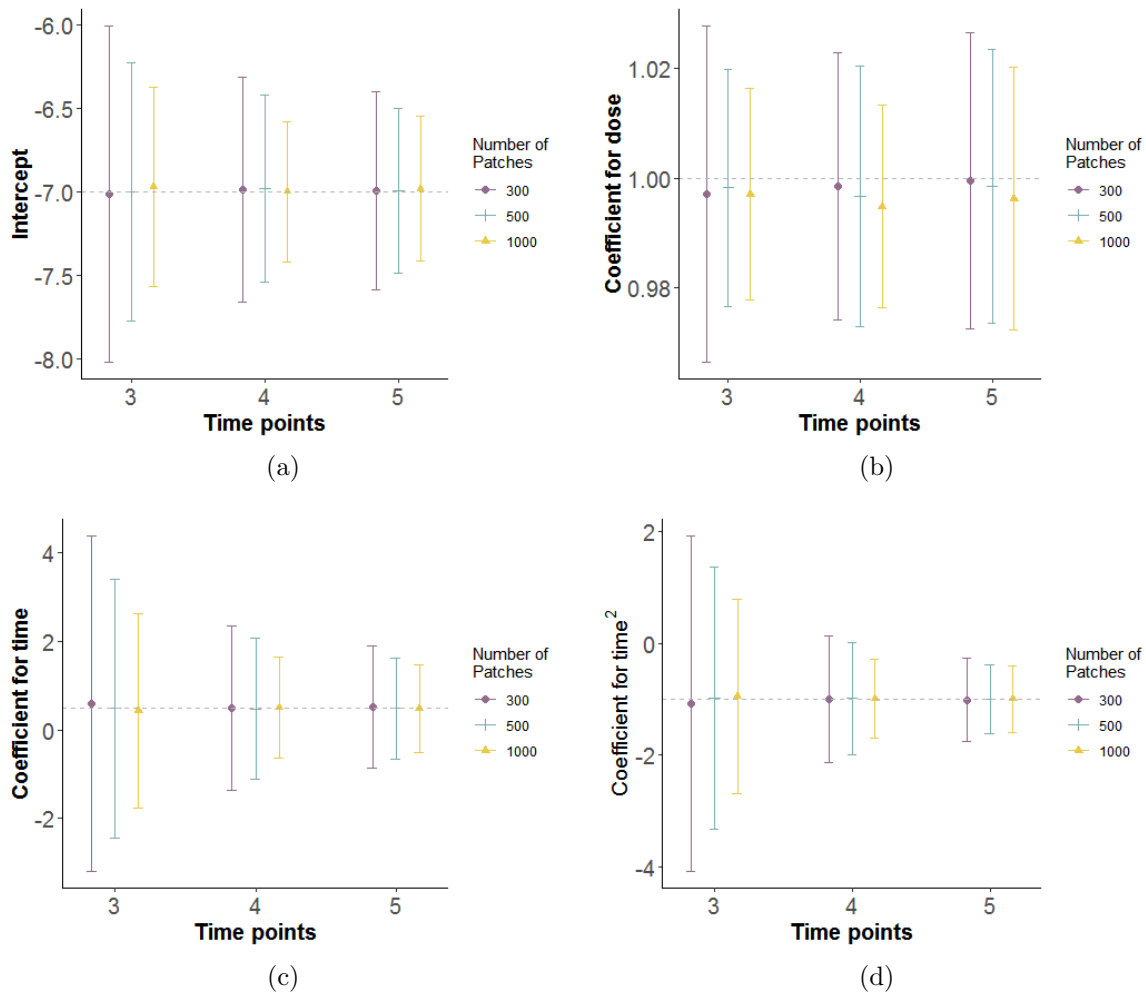


Figure 4.3: Simulation Results for estimated fixed effects: average estimate of fixed effects in the intensity model for $\text{aln}(\text{Dense}) \pm 1.96$ average standard error

4.1.2 Performance for estimation of nuisance parameters

In addition to the estimates of the fixed effects, the proposed model also performs the estimation of the variance components (Σ and Ψ) and the components associated with the mixture model (m and μ). The results obtained in the simulation for these parameters are observed in the Table 4.3. This table lists the average and the SD of the point estimates. All the variance components but Ψ_{12} are underestimated and strongly biased. This bias is basically caused by the inclusion of the step-halving procedure that forces a lower deviance in each iteration, not allowing a large and faster change in some parameters. However,

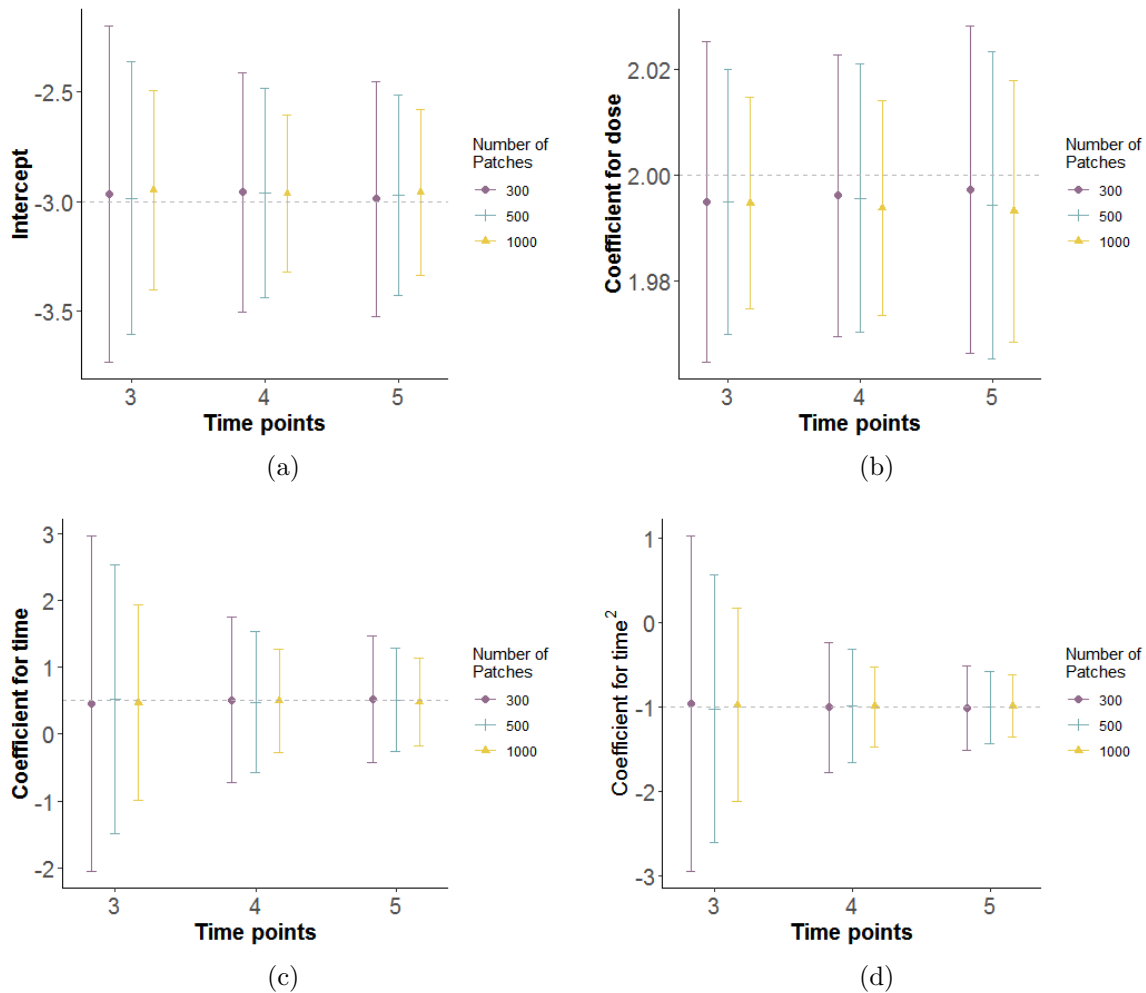


Figure 4.4: Simulation Results for estimated fixed effects: average estimate of fixed effects in the intensity model for $\text{alr}(\text{Hazy}) \pm 1.96$ average standard error

the step-halving procedure is necessary and widely used to deal with common convergence problems when using non-standard link functions, such as the log link binomial model used in the occurrence model.

The model performance on the estimation of the parameters of the mixture model was also not satisfactory. The average percentage of membership to the most frequent part of the mixture (m) has been shrunk towards 0.50, and the mean towards zero, indicating a no mixture model. This has then an effect on the estimation of the mean (μ), that being consistent with $m = 0.5$ has to take values very close to zero.

Now, even though the nuisance parameters are usually not the main interest in research,

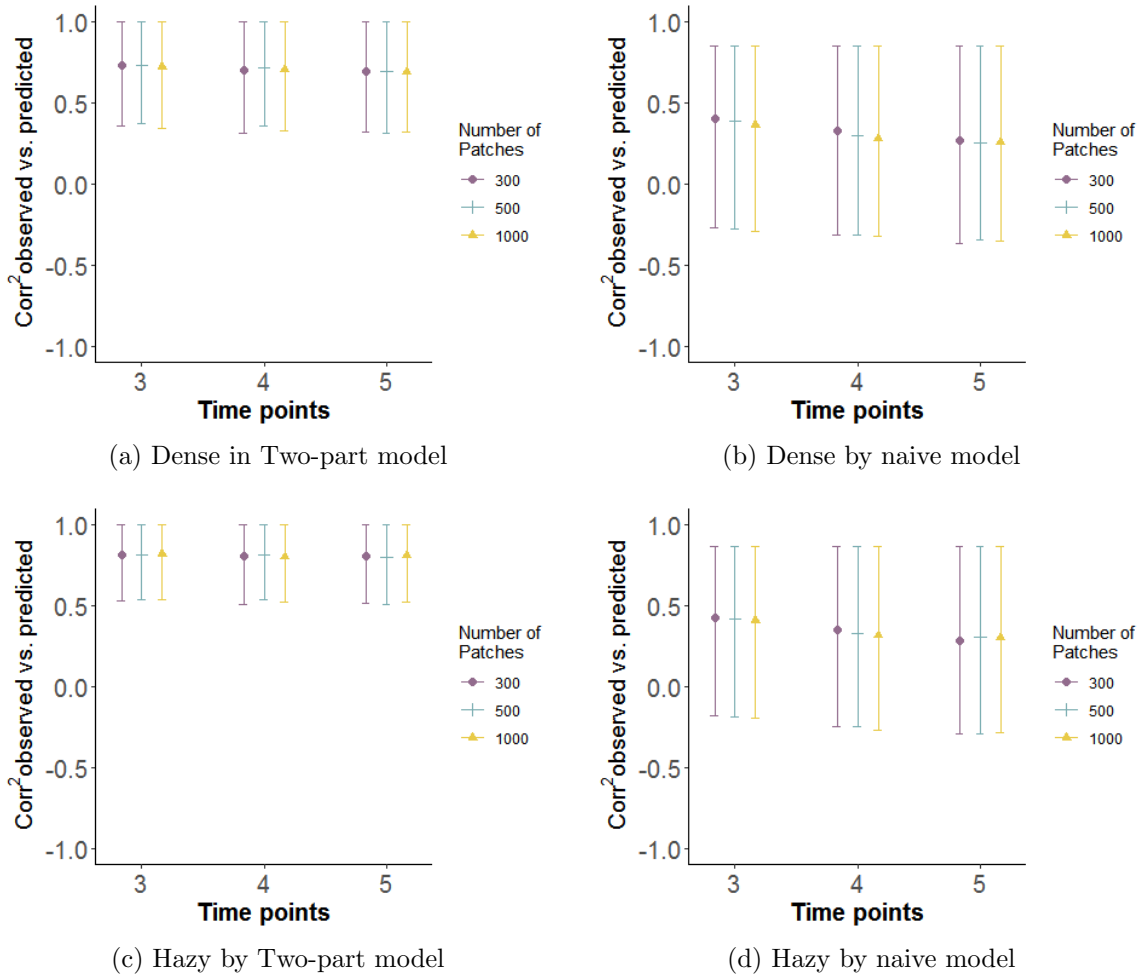


Figure 4.5: Simulation Results: Average correlation² between simulated and predicted data ± 1.96 standard deviation

Two-part model predictions correspond to the proposed method. Naive predictions use predictions from independent mixed-effects models)

they are used on the estimation of the random effects, and therefore will affect the model's predictions, and are also used on the hypothesis testing involving these parameters to find a more parsimonious model.

4.2 Conclusion

Since the scoring procedure proposed in the two-part mixed-effects mixture model for zero-inflated longitudinal compositional data is only a maximization to an approximate log-likelihood, it was important to analyze the performance of the model in estimating the

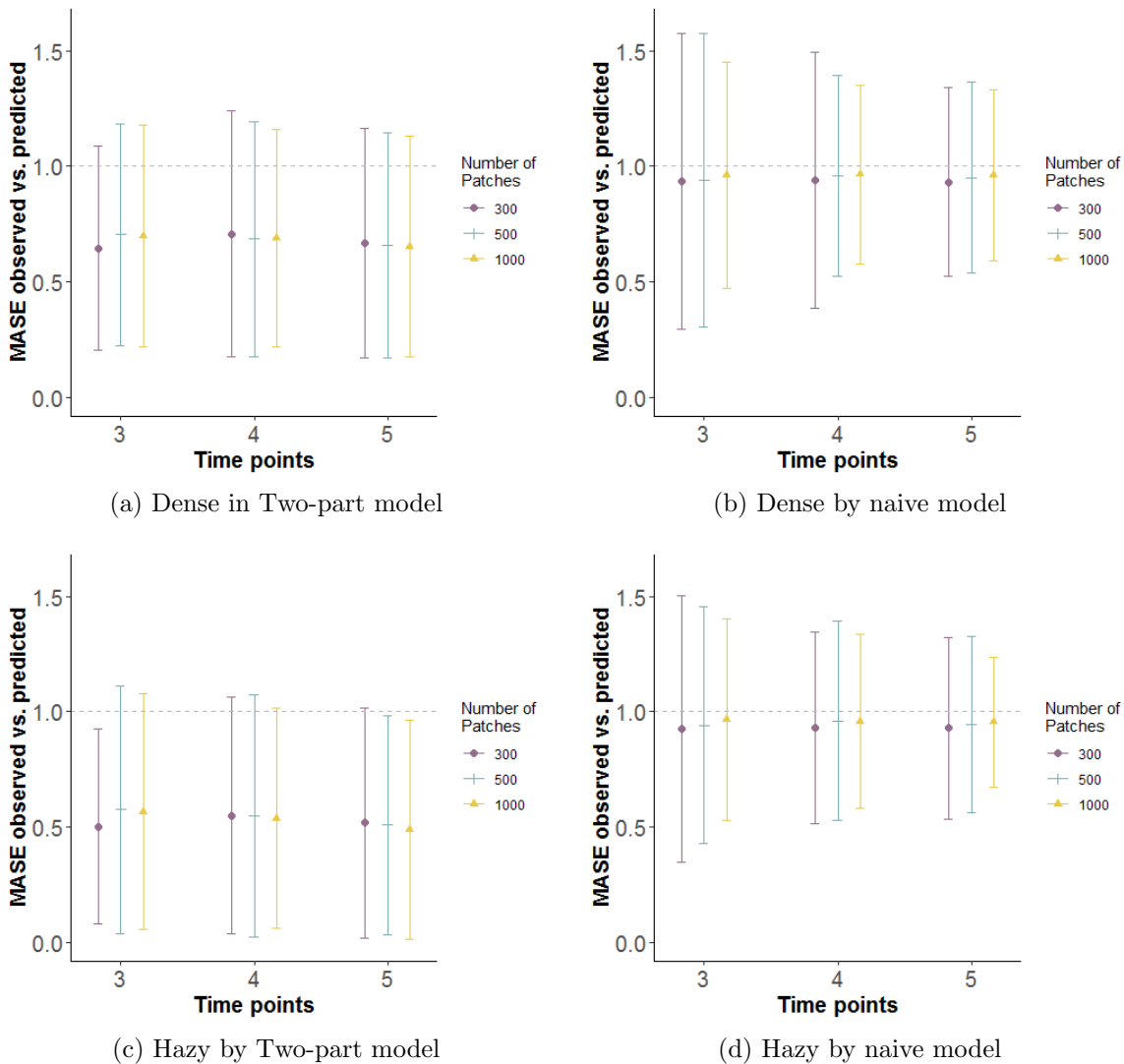


Figure 4.6: Simulation Results: Average mean absolute scaled error (MASE) between simulated and predicted data ± 1.96 standard deviation

Two-part model predictions correspond to the proposed method. Naive predictions use predictions from independent mixed-effects models)

parameters. With this objective, a simulation was performed to assess some of the performance aspects of the model. The results showed mixed performance, with reasonable estimates of the model’s fixed effects, but with highly biased estimates of other parameters.

Although the method seems to have good convergence in different sample sizes, in future studies, it is important to assess whether the implemented step-halving procedure is causing false convergence. The implemented procedure is similar to the implemented “loop3” in the R function *glm*,⁵⁰ which invokes the step-halving until convergence, until the difference in

Population	Σ_{11} 1	Σ_{22} 0.5	Cor ₁₂ 0.2	Ψ_{11} 1	Ψ_{12} 0.07	Ψ_{13} 0.2	Ψ_{22} 0.5	Ψ_{33} 1	Ψ_{23} 0.07	m 0.6	μ_1 -0.1	μ_2 0.1
N=1000, t=5												
Average estimate	0.74	0.35	0.59	0.75	0.07	0.11	0.29	0.55	-0.07	0.51	0.00	0.00
SD estimate	0.19	0.14	0.30	0.72	0.05	0.06	0.12	0.13	0.09	0.01	0.01	0.02
N=1000, t=4												
Average estimate	0.74	0.35	0.62	0.71	0.07	0.11	0.27	0.55	-0.06	0.51	0.00	0.00
SD estimate	0.19	0.13	0.30	0.64	0.04	0.06	0.11	0.12	0.09	0.01	0.01	0.01
N=1000, t=3												
Average estimate	0.74	0.33	0.69	0.75	0.07	0.11	0.27	0.53	-0.05	0.51	0.00	0.00
SD estimate	0.19	0.12	0.27	0.81	0.04	0.06	0.11	0.11	0.08	0.01	0.01	0.01
N=500, t=5												
Average estimate	0.81	0.38	0.57	0.86	0.07	0.11	0.27	0.54	-0.05	0.51	-0.01	0.00
SD estimate	0.19	0.14	0.28	1.62	0.05	0.07	0.12	0.16	0.09	0.01	0.01	0.01
N=500, t=4												
Average estimate	0.80	0.37	0.58	0.83	0.07	0.11	0.27	0.53	-0.04	0.51	-0.01	0.00
SD estimate	0.20	0.14	0.28	1.03	0.06	0.08	0.13	0.14	0.09	0.01	0.01	0.01
N=500, t=3												
Average estimate	0.81	0.37	0.62	0.92	0.07	0.12	0.27	0.53	-0.04	0.51	-0.01	0.00
SD estimate	0.20	0.14	0.26	1.47	0.10	0.10	0.14	0.15	0.09	0.01	0.01	0.01
N=300, t=5												
Average estimate	0.86	0.40	0.61	1.27	0.08	0.11	0.27	0.53	-0.03	0.51	-0.01	0.01
SD estimate	0.20	0.14	0.24	7.02	0.14	0.19	0.13	0.15	0.09	0.01	0.01	0.01
N=300, t=4												
Average estimate	0.86	0.39	0.61	1.49	0.08	0.12	0.27	0.53	-0.03	0.51	-0.01	0.01
SD estimate	0.20	0.13	0.24	9.93	0.27	0.21	0.15	0.15	0.08	0.01	0.01	0.01
N=300, t=3												
Average estimate	0.88	0.40	0.64	2.91	0.08	0.14	0.27	0.53	-0.03	0.51	-0.01	0.01
SD estimate	0.21	0.14	0.24	21.3	0.18	0.32	0.16	0.17	0.08	0.01	0.01	0.01

Table 4.3: Simulation results for estimated nuisance parameters

the relative deviance goes from one side of the convergence region to the other, or until this process has been repeated a fixed number of times. In this last situation, the new estimates can be very close to the previous ones and fall inside the convergence region, which could be causing a false convergence.⁵⁰ In this study, this step-halving procedure has repeated a maximum of 50 times. The effects in the model performance on those cases when convergence is attained after reaching the maximum number of repetitions need to be assessed.

The average time that the procedure takes for the estimation of the fixed effects seems to be in the feasible range for statistical analysis. Even for $N = 1000$, the procedure took on

average almost an hour to converge. The estimation of the random effects required a longer time, and large differences in the average time were observed between sample sizes, with less than one hour for $N = 300$ and more than 2.5 hours for $N = 1000$. The long times for the random effects estimations are a consequence of the empirical Bayes procedure involving the importance sampling method. In our approach, three different sets of random effects were produced (normal, truncated, and Pareto), then some optimization of the program might be done by providing only the Pareto estimates. However, it will only have a small reduction in the estimation time. Therefore, for estimation of the fixed the proposed method seems equally suited even for large sample size. Nevertheless, if the objective is focused on predictions, the estimation of the random effects will be necessary, and the method might be impractical for very large sample sizes.

The method proposed in the present dissertation seems to be making an adequate estimate of the variability of the estimates through the SE. However, for $N = 300$ performance of the estimated SEs was inadequate. Since it is very common to find studies with sample sizes smaller than 300, it is advisable to use the method with caution in such cases. In future research on this method, it is recommended to test the model performance on smaller sample sizes.

Estimation of nuisance parameters was somewhat unsatisfactory. As mentioned above, “loop 3” is the most likely cause of poor model performance in estimating variance components. By performing the step-halving procedure in each iteration, the change of each parameter is reduced more and more with each time this loop runs. Therefore, the changes in the estimates between iterations can be tiny for some parameters. Improvements to the step-halving procedure, perhaps limiting it only to fixed effects, could be implemented to mend this lack of performance.

The estimation of the parameters in the mixture model was also unsatisfactory. In this case, the percentage of membership in the most frequent distribution was reduced to 0.5. This could be an effect of the implemented estimation process, in which all patches, with or

without RILD, were used to update the m in each iteration. However, the proposed model only assumes a mixture of distributions for the random effects of the intensity vector (\mathbf{d}_i). Perhaps, limiting the update of m to observations with non-zero intensity could improve the estimation of m , and therefore the estimation of μ .

In conclusion, the two-part mixed-effects mixture model for zero-inflated longitudinal compositional data has shown reasonable performance in estimating fixed effects. However, the model would benefit from improvements in the estimation of the nuisance parameters. Additionally, future research should study the impact of the inclusion of other random effects that represent clustering, or random slopes, as well as the effect of missing data.

5. Probabilistic model for longitudinal lung tissue changes following RT

The two-part mixed-effects mixture model for zero-inflated longitudinal compositional data presented in Chapter 3 was applied separately to each of the patch definitions in Chapter 2 for the data of the reference patient. As mentioned, the RILD intensity on each patch was consolidated in the compositional vector $p_{it} = (\%Dense, \%Hazy, \%Normal)$. Then, each patch was classified as without RILD presence ($\%Normal=100$), or with RILD presence ($\%Normal \neq 100$).

For applying the proposed model, the outcome variable ($\mathbf{Y}_{ij}; i = 1, \dots, \text{number of patches}; j = 1, \dots, 5$) was the ratio of the vector p_{it} , using the last component (Normal) as denominator ($\mathbf{Y}_{ij}=(\%Dense/\%Normal, \%Hazy/\%Normal)$). When the whole patch presented RILD ($\%Normal=0$), or when only one of the damage components, dense or hazy, equals zero, the zero replacement method proposed by Fry et al.¹¹ with $\delta = 0.003$ was applied before calculating the *alr* transformation. Therefore, $\mathbf{Y}_{ij} = \mathbf{0}$ represents no observed tissue damage after radiation in the patch i at the time point j ($\%Dense=0, \%Hazy=0, \%Normal=100$).

The percentage of patches without RILD varies across patch definitions, as observed in Chapter 2. This percentage ranges between 50% to 99% (Figures 2.13 (a), 2.17 (a), 2.23 (a)). Regardless of the patch definition, there is a large proportion of zeros on these data sets. Hence, the occurrence variable is defined as

$$U_{ij} = \begin{cases} 1 & \text{If RILD is observed in the patch } i \text{ at the time point } j \\ 0 & \text{otherwise} \end{cases}$$

and the intensity variable $\mathbf{V}_{ij} = \log(\mathbf{Y}_{ij})$, when $\mathbf{Y}_{ij} \neq \mathbf{0}$. Then, \mathbf{V}_{ij} is the *alr* trans-

formation of the compositional vector ($\%Dense, \%Hazy, \%Normal$). Median patch dose, time in years, and time squared were used as explanatory variables for modeling the fixed effects for both outcomes (occurrence and intensity). To account for the interdependence over time, a random intercept was used in both models. No other factors were considered as random effects. Iterative estimation was performed until convergence was attained (tolerance = $1e-8$). Table 5.1 shows the estimates of the fixed effects in the models for all three patch definitions.

In the cubic-patch definition of dimension $8 \times 8 \times 8$, the reference patient has 15,495 patches for analysis, and only 0.50% of them present RILD at some point. For this scenario, the scoring procedure converged in 6 iterations, which took 8.3 hours for maximum relative parameter change of $4.9E-14$. Calculation of starting values took 15.5 minutes, and the estimation of the random effects took 41.1 hours. Looking at the parameter estimates, an increment in one Gy showed a 14% increment in the odds of RILD for a particular patch (OR: 1.14; 95%CI:1.11, 1.18) (Table 5.1). On the other hand, the odds of RILD increase during the first 14 months, then decrease over time. The estimated baseline composition, when the patch received zero dose, was ($\%Dense=0, \%Hazy=0.02, \%Normal=0.98$). Table 5.2 shows the shift in the composition from the baseline by a unit change in each covariate when the other covariates are unchanged. Values close to 0.33 imply that changes in the covariate do not have an effect on that particular part of the composition. In the same way, values larger than 0.33 suggest a positive correlation between the covariate and the composition part, and values smaller than 0.33 indicate negative correlations. Therefore, the dose is positively correlated with the percentage of dense, and it is negatively correlated with normal tissue.

Parameter	Cubic			Spherical			Isodose					
	Est	S.E.	<i>exp</i> (Est)	Est	S.E.	<i>exp</i> (Est)	Est	S.E.	<i>exp</i> (Est)	95%CI		
Occurrence model												
Intercept	-15.32	0.91	2.2E-07	(0, 0)	-10.08	0.02	4.20E-05	(0, 0)	-9.89	0.76	5.1E-05	(0, 0)
Median dose	0.13	0.02	1.14	(1.11, 1.18)	0.46	0.17	1.58	(1.14, 2.19)	0.23	0.02	1.26	(1.21, 1.31)
Years	4.16	1.34	64.35	(4.69, 882.57)	-2.87	0.01	0.06	(0.06, 0.06)	2.35	0.90	10.48	(1.78, 61.58)
Years2	-1.76	0.70	0.17	(0.04, 0.67)	0.58	0.01	1.79	(1.76, 1.81)	-0.84	0.44	0.43	(0.18, 1.01)
Intensity model												
Dense/Normal												
Intercept	-8.50	0.85	2.0E-04	(0, 0)	-6.39	0.46	1.68E-03	(0, 0)	-13.02	1.8E-06	2.2E-06	(0, 0)
Median dose	0.07	0.01	1.08	(1.05, 1.11)	0.16	0.02	1.18	(1.14, 1.22)	0.16	3.2E-08	1.18	(1.18, 1.18)
Years	2.35	1.26	10.49	(0.90, 122.84)	-0.44	0.62	0.65	(0.19, 2.18)	2.72	0.39	15.20	(7.11, 32.52)
Years2	-0.96	0.61	0.38	(0.12, 1.27)	0.27	0.29	1.31	(0.75, 2.29)	-0.99	0.19	0.37	(0.26, 0.54)
Hazy/Normal												
Intercept	-4.08	0.53	0.02	(0.01, 0.05)	-6.7	0.57	1.24E-03	(0, 0)	-9.82	0.25	5.4E-05	(0, 0)
Median dose	0.04	0.01	1.04	(1.02, 1.06)	0.20	0.02	1.22	(1.18, 1.26)	0.14	4.3E-03	1.15	(1.14, 1.16)
Years	-1.46	1.04	0.23	(0.03, 1.78)	-1.07	0.66	0.34	(0.09, 1.24)	0.50	0.28	1.65	(0.95, 2.85)
Years2	0.68	0.49	1.98	(0.75, 5.22)	0.55	0.32	1.73	(0.93, 3.22)	-0.16	0.14	0.85	(0.65, 1.11)

Table 5.1: Parameter estimates and standard errors of the fixed effects for modeling RILD by patch definition

Est:Estimate

Patch definition	Dense	Hazy	Normal	Disturbance
Cubic				
Baseline	0.00	0.02	0.98	
Median dose	0.35	0.33	0.32	0.05
Years	0.89	0.02	0.09	2.72
Years2	0.11	0.59	0.30	1.17
Spherical				
Baseline	0.00	0.00	1.00	
Median dose	0.35	0.36	0.29	0.15
Years	0.33	0.17	0.50	0.76
Years2	0.32	0.43	0.25	0.39
Isodose				
Baseline	0.00	0.00	1.00	
Median dose	0.35	0.35	0.30	0.13
Years	0.85	0.09	0.06	2.05
Years2	0.17	0.38	0.45	0.75

Table 5.2: Additive logistic transform estimates of the model applied to the composition given that there is some RILD presence

The reference patient has 296 patches of the spherical type, and less than 20% of them present RILD. In this patch approach, the scoring procedure took 6 iterations to converge, which took 7 minutes to a maximum relative parameter change of 1.3E-13. Additionally, the estimation of the random effects took 33 minutes. The odds of RILD, for a particular patch, increased 58% by each Gy increase in the dose (OR:1.58, 95%CI:1.14, 2.19) (Table 5.1). On the other hand, the odds of RILD decrease over time. Given that there is RILD presence in the patch, the estimated baseline composition was (Dense=0, Hazy=0, Normal=1) (Table 5.2). An increase of 1 Gy in the dose shifts the baseline composition by (0.35, 0.36, 0.29). Therefore, in this scenario, the dose is positively correlated with the percentage of dense and hazy, while it is negatively correlated with the percentage of normal tissue. Moreover, over time a big shift on the composition would be expected as an increment on the normal tissue by month.

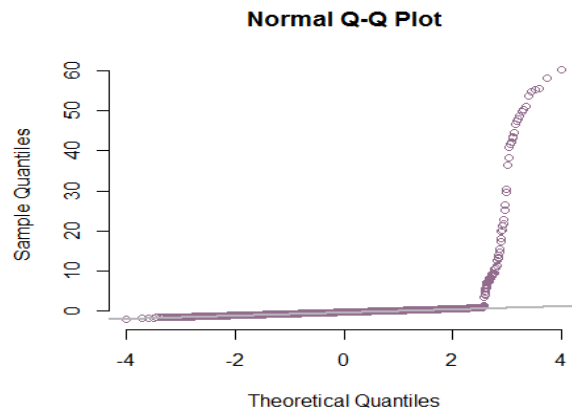
For the isodose definition, the reference patient has 288 patches, of which 58.48% present RILD at some point after RT. The model converged after 6 iterations, which took 8.1 minutes

to a maximum relative parameter change of 1.2E-13. Also, the estimation of the random effects took 30 minutes. In this patch approach, the odds of RILD, for a particular patch, increased 26% for each Gy increase in the dose (Table 5.1). Likewise, the odds of RILD increase until 17 months after RT, then the odds decrease over time. In the results for the intensity vector, the estimated baseline composition was (Dense= 0, Hazy= 0, Normal= 1) (Table 5.2). An increase of 1 Gy in the dose shifts the baseline composition by (0.35, 0.35, 0.30). Consequently, the dose is positively correlated with both dense and hazy. Furthermore, over time the changes in the compositions are characterized by an increment in the proportion of dense.

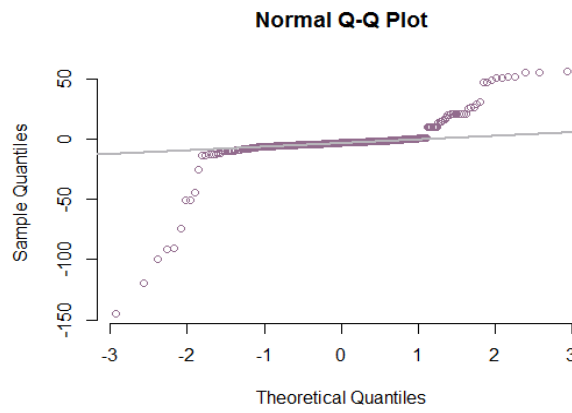
5.1 Assessing model adequacy

Assumptions and goodness of fit of the model were assessed for all patch definitions. First, the normality assumption of the random effects was assessed. In this cases, the random effects come from two different sources, the random effects for the occurrence model (c_i), and the random effects from the intensity model (d_i). In the occurrence model, the random effects are distributed $N(\mathbf{0}, \Psi_{cc})$. The qq-plots of this random effect for all patch definitions are presented on Figure 5.1. Apparent deviations of the normality assumption are observed for almost all patch definitions. In the cubic definition, the tail of the distribution corresponds to patches without RILD, while in the spherical and isodose definitions, the tails correspond to patches with RILD.

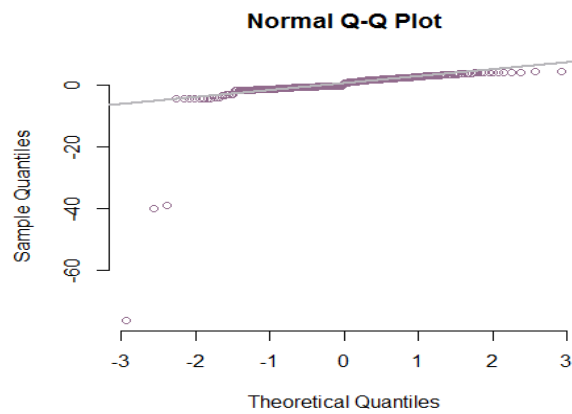
The bivariate vector of random effects in the intensity model is distributed by a mixture of normal distributions. A goodness-of-fit test of a mixture of normal distributions using the Cramér-von Mises⁵¹ can be implemented for this objective. However, these results are not presented in the current work. The lack of agreement with the normality assumption can be a given by the exclusion or misspecification of the variables in the model. However, as shown in Verbeke and Lesaffre⁵², and in Butler and Louis⁵³, the inference of the fixed effects might be robust to the non-normality of random effects, but it can affect the predictions at



(a) Cubic



(b) Spherical



(c) Isodose

Figure 5.1: Normal Q-Q plots for random effects in the occurrence model (c_i) by patch definition

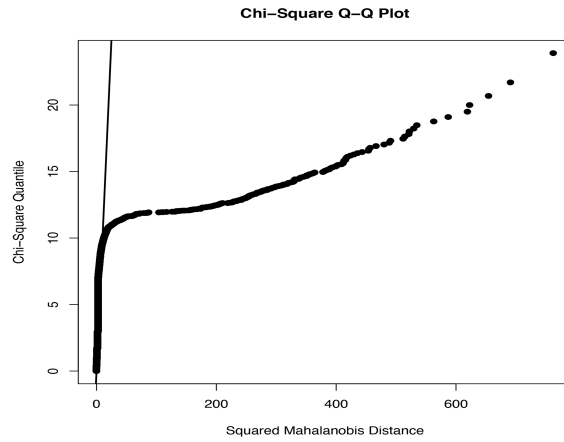
the patch level.

Figure 5.2 shows the q-q plots for the errors of the intensity model. There is also evidence of a lack of normality in the residuals, and mainly, in all patch definitions, patches that move further from the straight line correspond to patches without RILD, which is expected given that the intensity model is conditional on the presence of RILD. However, the number of patches in all the patch definitions is larger than 250. Therefore, Central Limit Theorem should assure some robustness, provided the model is asymptotically unbiased.

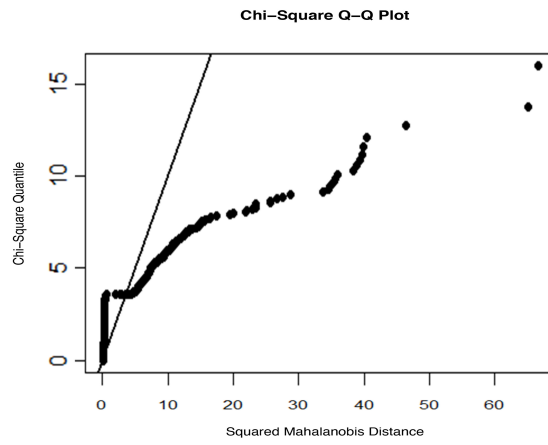
Nevertheless, violation of the homoscedasticity assumption of the residuals might have a negative impact on the variance estimates of the parameters in the model. Figures 5.3 and 5.4 shows the scatter plot of the standardized residuals versus dose and time. The subresiduals come from the comparison between the predicted values in the intensity model and the observed values. The predicted values from the intensity model were multiplied for the estimated probability of RILD, and these values were used on the calculation of the residuals. Also, in these plots, patches with no observed RILD at any occasion cluster along lines, and since the estimates of the random effects (\mathbf{d}_i) are set at the population-level predictions, large standard errors for the random effects in these patches are expected.

For the cubic patch, residuals for both dense and hazy seem to have zero variability in low doses (< 10 Gy), since these patches did not present any RILD at any time point (Figure 5.3 (a)-(d)). In doses larger than 10 Gy, there is not evidence of heteroscedasticity in the cubic-patch approach. In the spherical patch, for both transformed dense and hazy, the residuals seem to have larger variability on the lower and higher ends of dose range (Figure 5.3 (e)-(h)). There is no evidence of a relationship between median dose and the subresiduals (Figure 5.3 (e) and (g)), which contributes some evidence in favor of the linearity assumption.

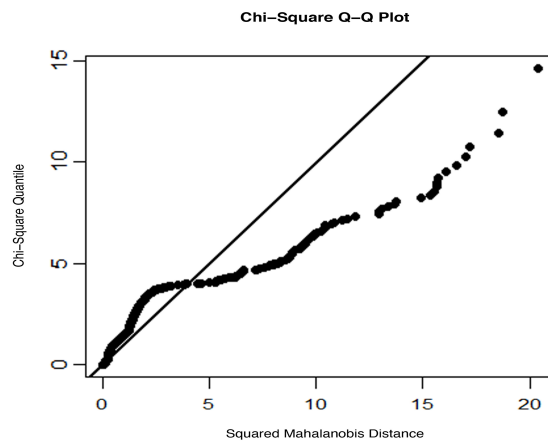
In the isodose patch, residuals in doses less than 20 Gy have zero variability because those patches did not present RILD at any time point. After 20 Gy, the residuals seem to have larger variability at larger doses of RT (Figure 5.3 (i)-(l)). Additionally, these scatter plots suggest some relationship between the standardized residuals and the median dose,



(a) Cubic



(b) Spherical



(c) Isodose

Figure 5.2: Chi-Square Q-Q plots for bivariate residuals from the intensity model by patch definition

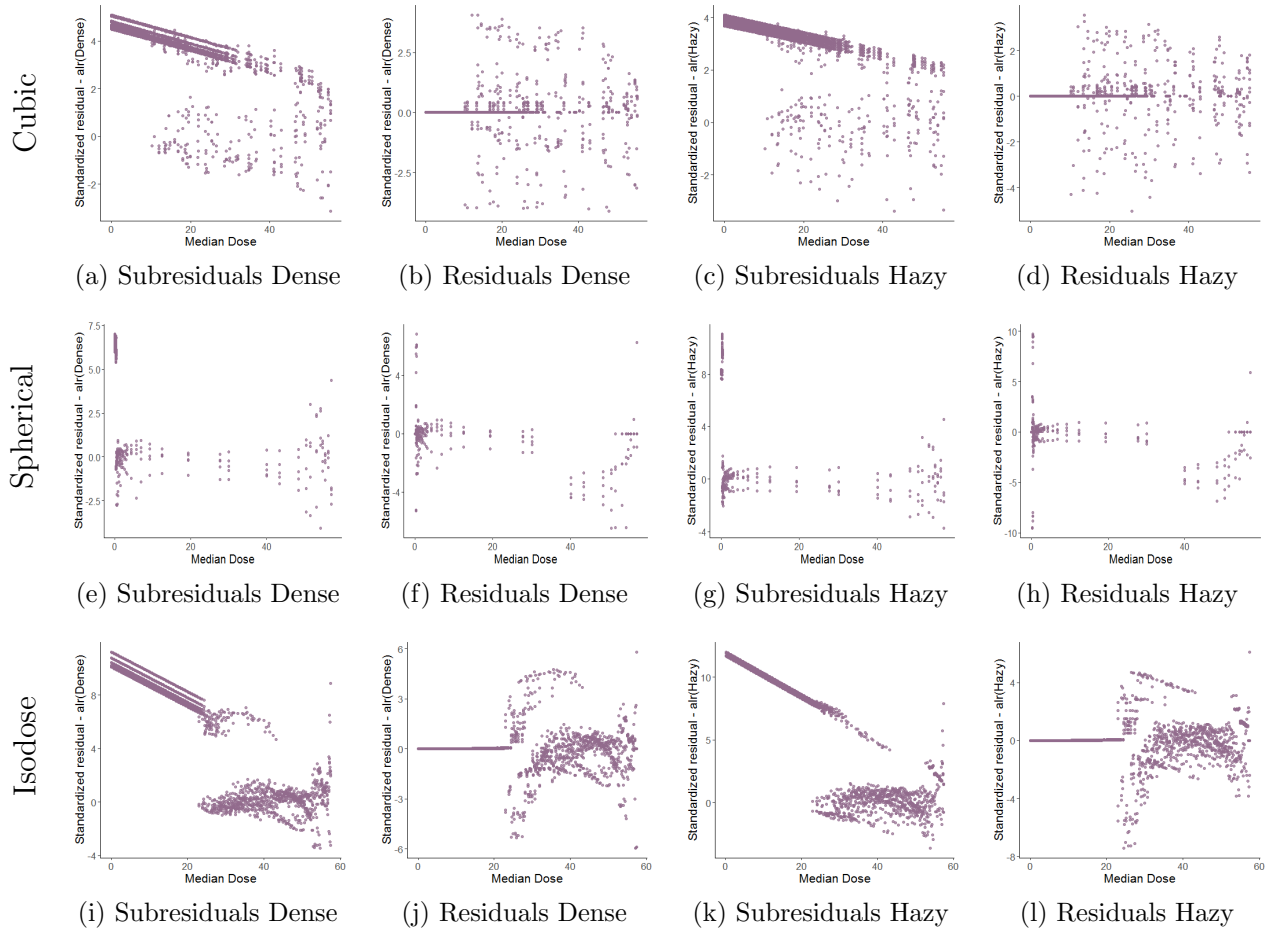


Figure 5.3: Standardized residuals vs. dose for the reference patient by patch definition. Subresiduals are calculated from the intensity model, residuals are calculated from the two-part model (integration between occurrence and intensity models)

which might imply that further transformation of the median dose could improve the model goodness-of-fit in the isodose patch. When looking at the plots between the residuals and time, no particular systematic pattern indicating violations of assumptions are observed (Figure 5.4).

For assessing the assumption that the two models, occurrence and intensity, are correlated, Figure 5.5 shows the estimated log odds for each patch and the expected $alr(Dense)$ and $alr(Hazy)$. As mentioned, the points cluster along lines representing the patches that did not have RILD at any time point after RT. Based on the plots for the cubic patch, patches with low propensities to present RILD also tend to have lower RILD intensity (Figure 5.5 (a) and (b)). Similar trend is observed in the isodose-patch approach (Figure 5.5 (e) and

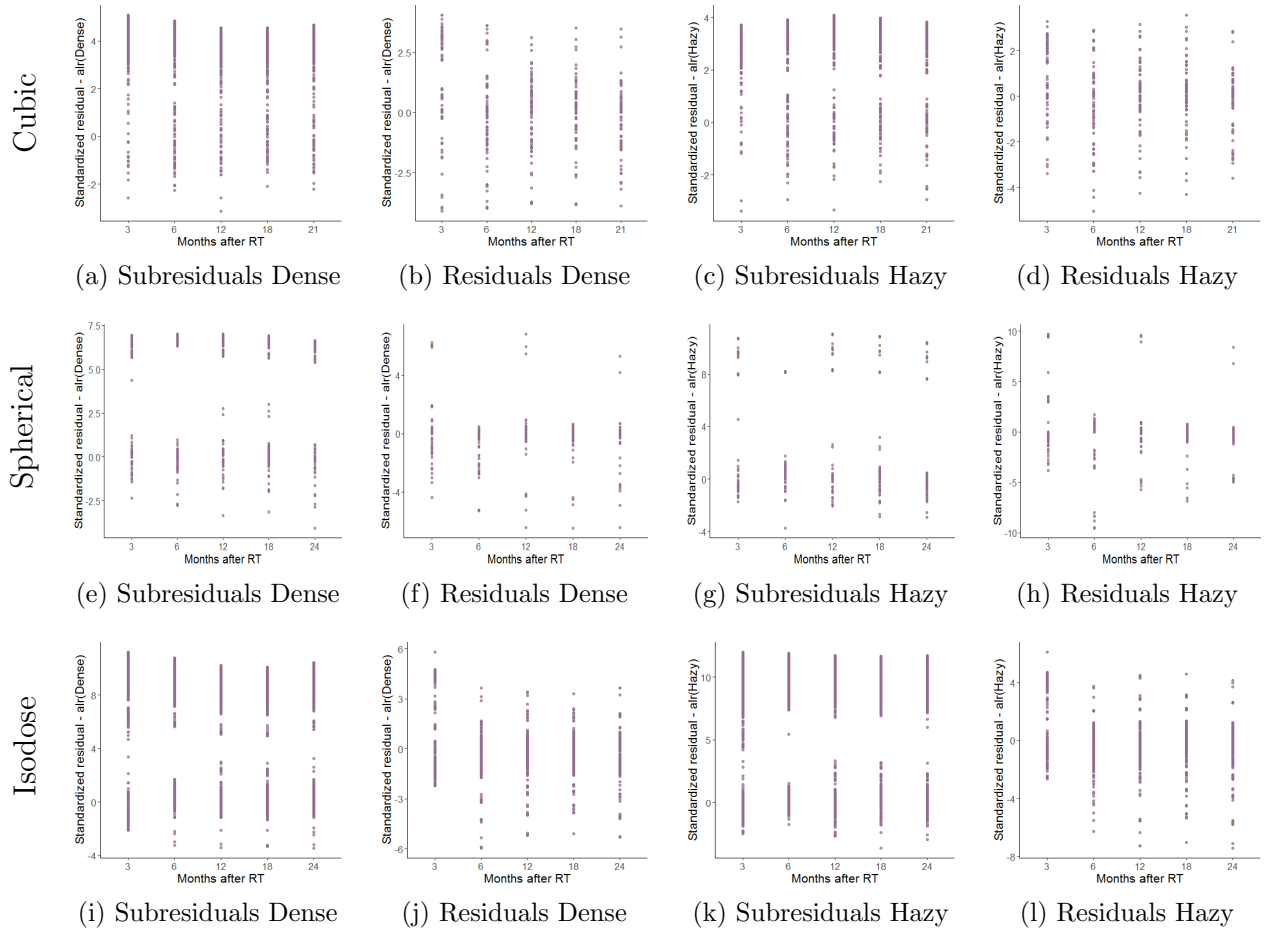


Figure 5.4: Standardized residuals vs. time using median dose as predictor, for the reference patient by patch definition

Subresiduals are calculated from the intensity model, residuals are calculated from the two-part model (integration between occurrence and intensity models)

(f)). However, in the spherical patch, the plots indicate that patches with low propensities to present RILD also tend to have large RILD intensity (Figure 5.5 (c) and (d)).

Figure 5.6 shows the observed number of times that each patch presented RILD, and the sum of the predicted probabilities at each time point ($\sum_{j=1}^{n_i} \hat{U}_{ij}$). As observed, in all patch definitions, the prediction in patches of normal tissue during the whole follow-up (Observed = 0) is very good. However, in the cubic and spherical-patch approaches, the predictions of the occurrence for patches with RILD is either underestimated or overestimated, with large variability. Out of all patch definitions, the isodose patch seems to have the smaller discrepancies in the fit of the occurrence pattern (Figure 5.6 (c)). The scaled residuals

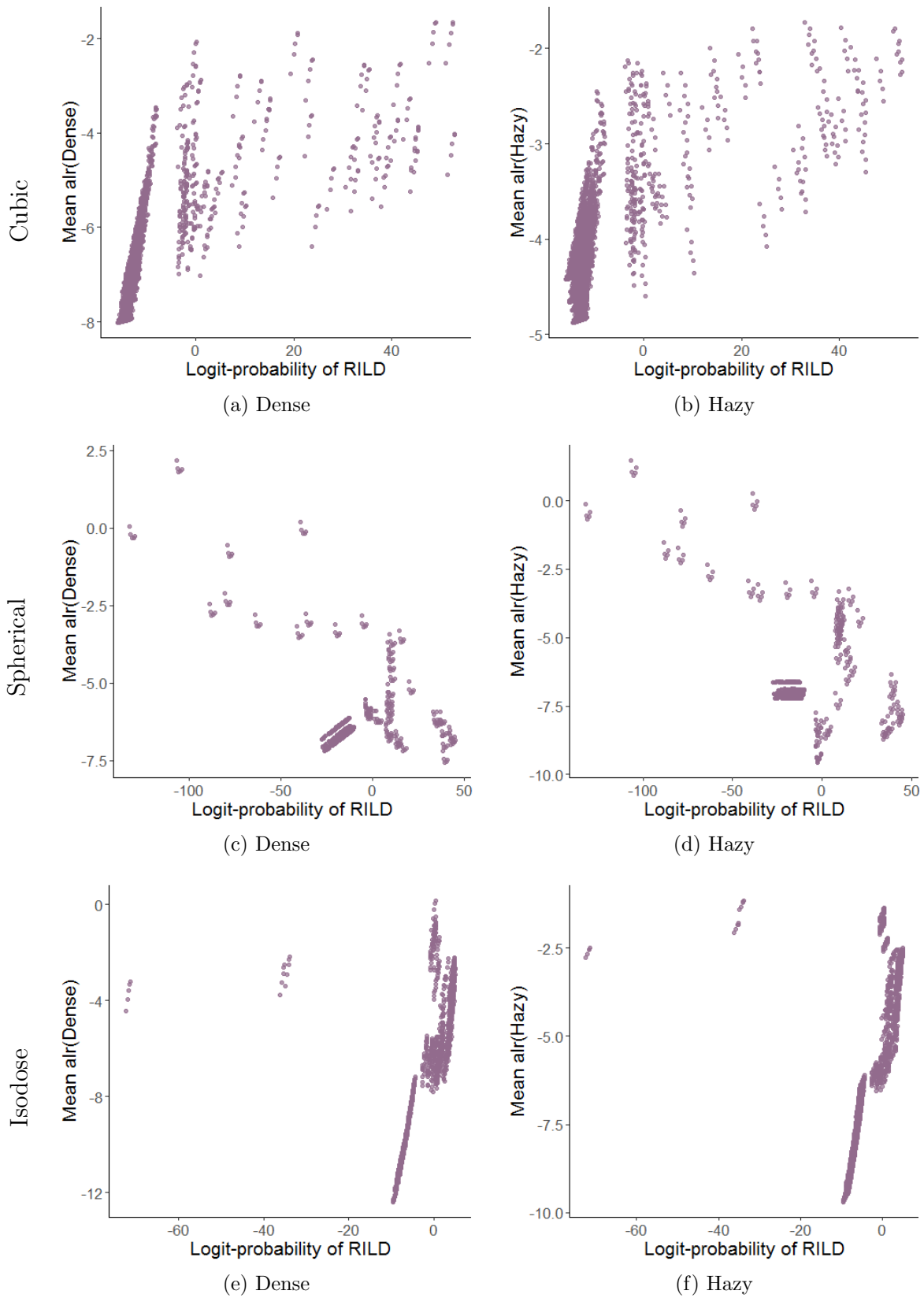


Figure 5.5: Estimated mean $alr(Dense)$ and $alr(Hazy)$ vs. logit-probability of RILD for the reference patient by patch definition

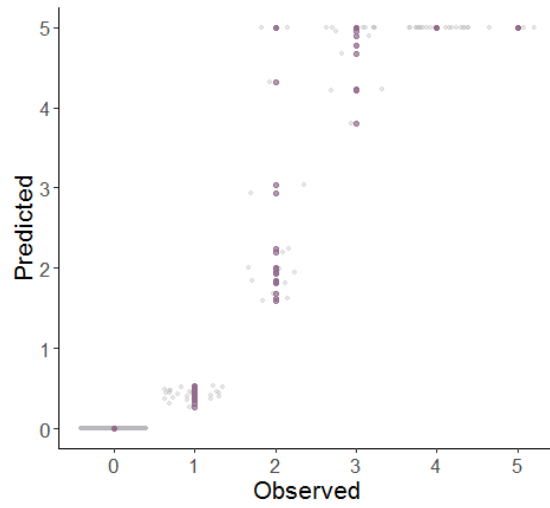
$(U_i - \hat{U}_i) / \sqrt{\hat{U}_i(n_i - \hat{U}_i) / n_i}$ were calculated, and outliers were identified as scaled residuals with magnitude larger than 2.5. In the cubic patch, 0.20 % of the patches were classified as outliers, in the spherical patch this proportion was 5.0%, and it was 1.0% in the isodose patch definition.

Figure 5.7 shows the scatter plots of predicted vs. observed transformed dense and hazy proportions. The predicted values from the intensity model are labeled as “subpredicted” and are in the first column of this set of plots. In the second column are presented the predictions from the first model multiplied by the estimated probability of RILD for each patch. There is an effect of the underestimated probabilities of RILD from the occurrence model on the predicted values from the intensity model adversely resulting in inferior predictions. The model using the isodose patch demonstrates the least deviation even with adjustment with the prediction of the occurrence model (Figure 5.7 (j) and (l)).

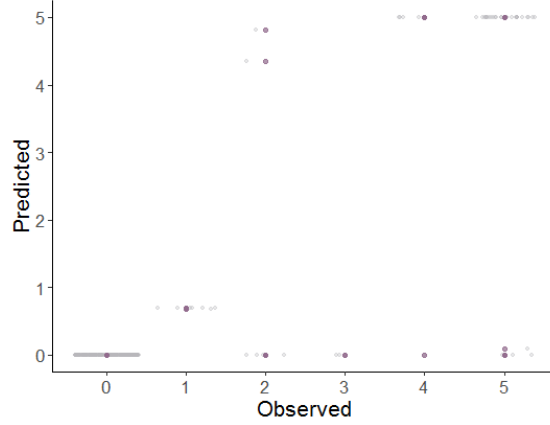
5.2 Clinical implications

In patients with lung cancer, medical images of different modalities are easy to access and a non-invasive tool to monitor radiographic changes following radiotherapy. Although the process of lung tissue changes following radiation is clearly visible, predictive models of such changes are not available. Part of the reason why statistical models for longitudinal image sets are underdeveloped is the complexity of the data. Prior to the models presented in this research, simple and direct longitudinal models for dense and hazy compositions of patches were tested. However, predictions out of those models were unsatisfactory, with strong violations of model assumptions. The two-part model for zero-inflated longitudinal compositional data appears to be a big improvement to address these issues. Additionally, after a descriptive analysis of the patch data, the bimodality of the dense and hazy distribution was observed and the model was adjusted accordingly.

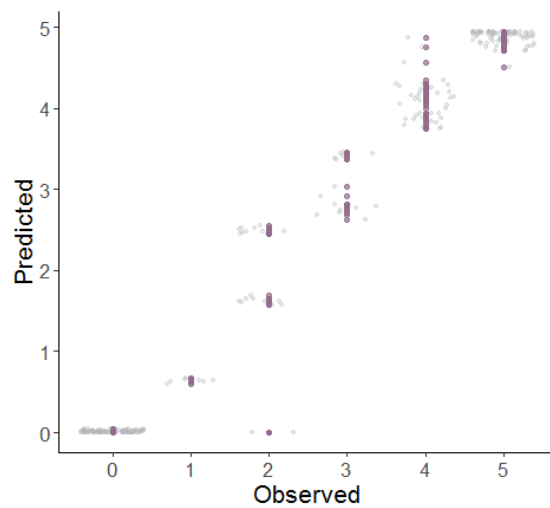
The two-part mixed-effects mixture model for zero-inflated longitudinal compositional data was applied to the three different patch definitions (cubic, spherical, and isodose), and



(a) Cubic



(b) Spherical



(c) Isodose

Figure 5.6: Observed number of times with RILD vs. expected probability of RILD for the reference patient by patch definition

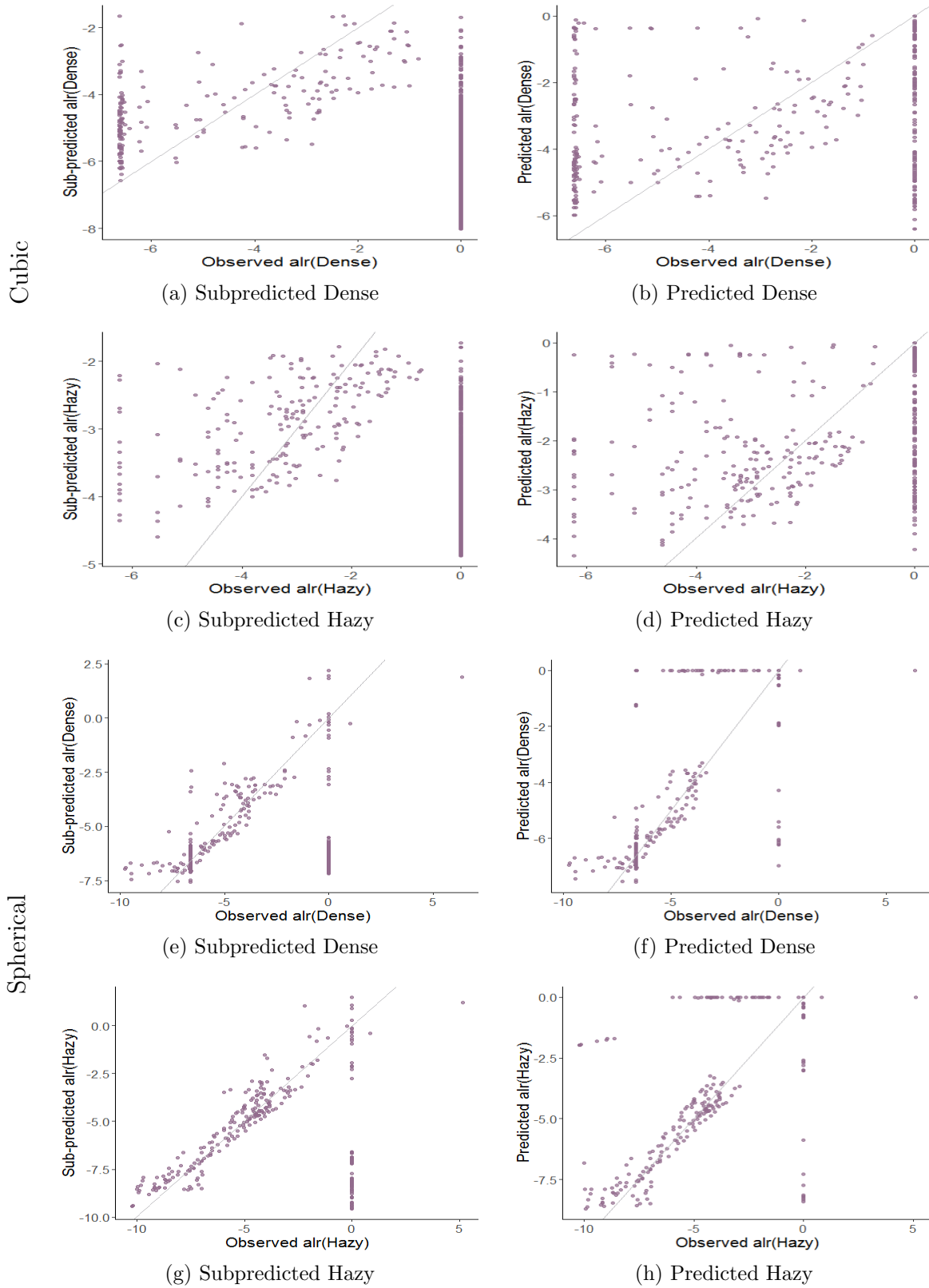


Figure 5.7: Observed vs. predicted $alr(Dense)$ and $alr(Hazy)$ for the reference patient by patch definition

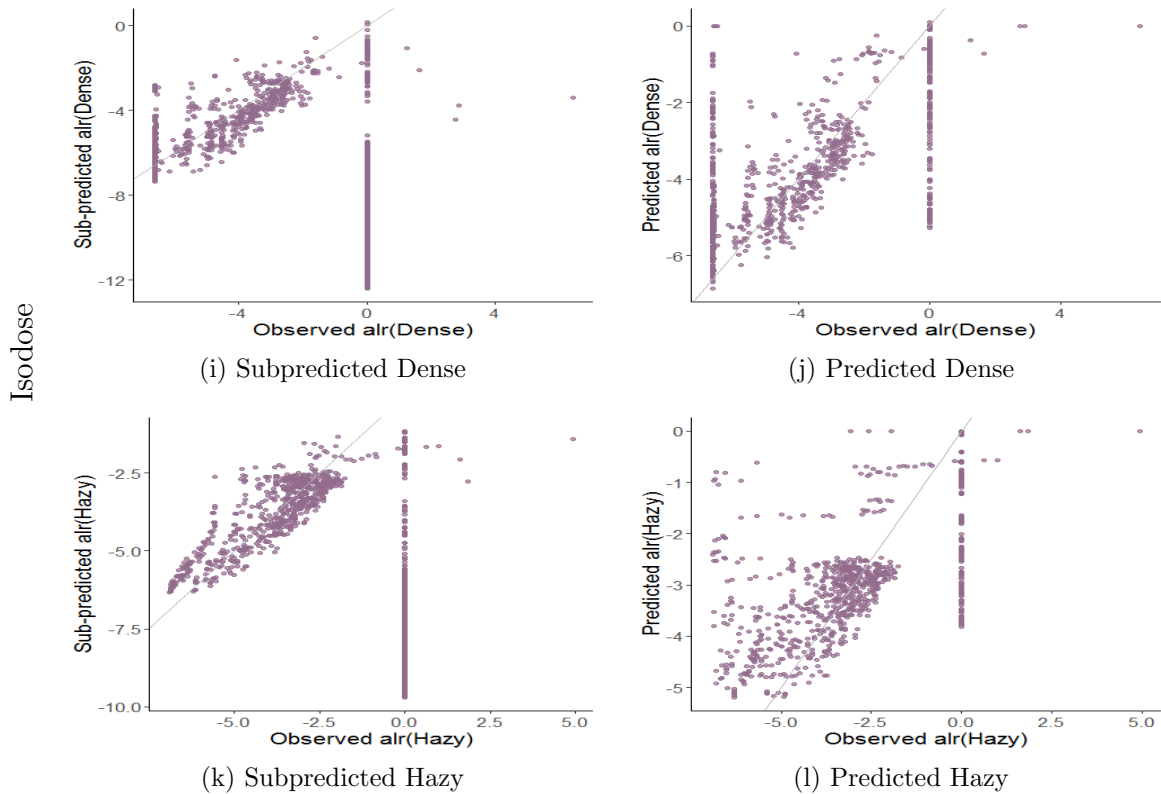


Figure 5.7: Observed vs. predicted $alr(Dense)$ and $alr(Hazy)$ for the reference patient by patch definition (cont.)

Subpredicted values are the predictions of the intensity model, the predicted values are the prediction of the two-part model (integration between occurrence and intensity models)

the most consistent results, with the best goodness-of-fit were obtained in the the isodose patch approach. Based on this model, increments in the RT dose were associated with an increase in the odds of RILD for a particular patch (OR: 1.26, 95%CI:1.21, 1.31) (Table 5.1). Also, the dose was positively associated with both the percentage of dense and hazy (Table 5.2).

The predictions of the two-part model for the isodose patch were transformed back to the composition form. Figure 5.8 shows the predicted composition for the patches of the reference patient. The predictions show a large increment in dense and a small increment in hazy during the first 18 months after RT, and those percentages start decreasing after that time point. When comparing the predictions with the observed data (Figure 2.24), the predictions underestimate the high percentages of dense and hazy at 3 months after RT.

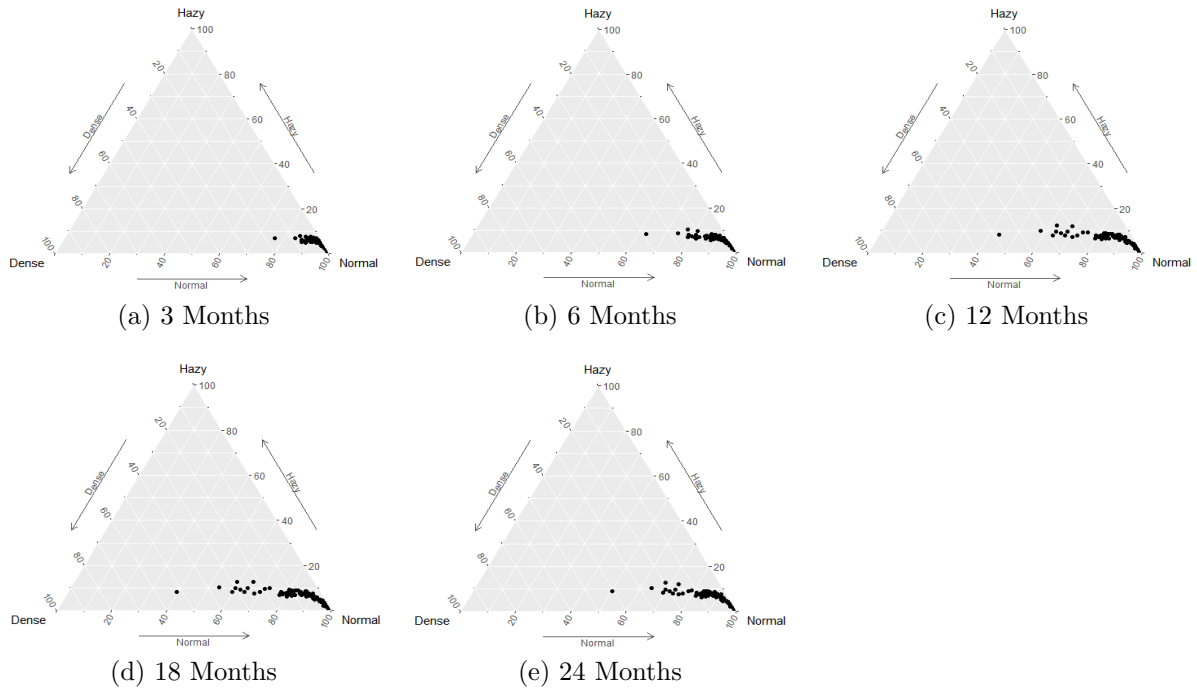


Figure 5.8: Predicted ternary plots of dense and hazy composition by time after RT in the isodose-patch using median dose as predictor for the reference patient

Also, even though the predictions show a decrease in dense and hazy after some time, it is not as substantial and fast as in the observed data. Figure 5.9 show predicted values for doses between 0 to 60 Gy by month. As mentioned before, increments in dose were related with increments in both dense and hazy percentages, and over time, dense percentage increases. Also, similar to the observed data for this patient, only patches with high doses present dense or hazy damage in the predictions.

The violation of the assumptions was mainly related to patches without RILD presence. In these cases, the estimates of the random effects revert to the population-level regression predictors, resulting in very variable estimates, affecting predictions at the patch level. However when comparing the predicted compositions (Figure 5.8) with the observed compositions (Figure 2.24), these prediction problems do not seem to strongly affect the patches without RILD ($\%Dense=0$, $\%Hazy=0$, $\%Normal=1$). Nevertheless, this departures from the assumptions might explain the differences between observed and predicted compositions in

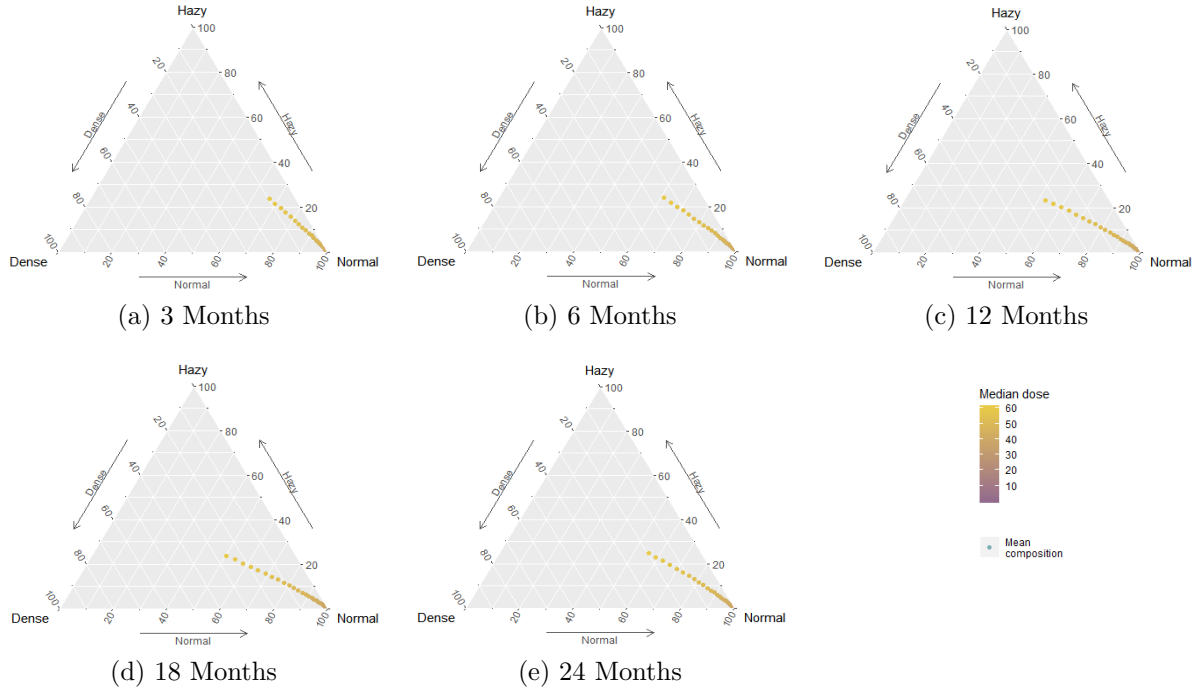


Figure 5.9: Predicted ternary plots of dense and hazy composition by time after RT in the spherical-patch using log of median dose as predictor

patches with RILD.

5.3 Conclusion

The two-part mixed-effects mixture model for zero-inflated longitudinal compositional data was applied to the three different patch definitions (cubic, spherical, and isodose), and mixed results were observed. The isodose patch approach showed the best performing model with respect to goodness-of-fit, by better predictions of both occurrence and intensity, as well as fewer violations to the model assumptions. Based on this model, increments in the RT dose were associated with an increase in the odds of RILD for a particular patch. Also, the dose was positively associated with both the percentage of dense and hazy.

Results suggest a correlation between the random effects of both models (occurrence and intensity). The methodology proposed in this study considers this correlation in the estimation process. Ignoring this relationship, by fitting the logit and the linear models

separately, would introduce substantial bias in the estimates of coefficients, particularly in the coefficients of the intensity model.

It is important to consider that the estimated coefficients in the occurrence model represent the effect of changes in the variables on a particular patch, not a population-average effect. That is, the model responds to the probability of RILD for each patch, not to the percentage of patches with RILD. Population-average estimates tend to maintain the same direction as the mixed model estimates but with smaller magnitude, as well as smaller standard errors. Some approaches to estimate the population-averaged effects based on mixed model estimates have been discussed in the literature,⁵⁴ but were not implemented here since the main interest was the effect of the variables in each patch and not the overall effect.

The lack of agreement with the normality assumption was observed, for both the random effects and the error term in the intensity model. Exclusion or misspecification of the variables in the model might be one of the explanations of this finding. In this particular example, the number of patches in all the patch definitions is larger than 250; Central Limit Theorem should allow asymptotic approximation to a reasonable level. Therefore, the inference of the fixed effects might be robust to the non-normality of random effects,^{52,53} but the predictions at the patch level might be affected. Nevertheless, violation of the homoscedasticity assumption of the residuals might have a negative impact on the variance estimates of the parameters in the model. Notably, the patches with no RILD at all time points represent an issue for both parts of the model. First, in the occurrence model, these patches might have an actual logit of probabilities of RILD of $-\infty$.⁴⁴ Also, even though the intensity model can provide predictions on these patches, these predictions have large standard errors.⁴⁴ Therefore, these patches have a substantial impact on the model assumptions. Furthermore, most of the procedures for diagnosing lack of adequacy in this model are based on the independent results or residuals of each of the parts (occurrence and intensity); however, future research should involve formal and informal methods for model adequacy assessment using the residuals from the two-part model.

Additionally, the model presented in this section is a simplified version of the radiographic lung change following radiotherapy of lung cancer, being able to predict an average trajectory overtime for all patches, which moves higher as the received dose increases, and starts at different points for each patch due to the random intercept. However, this model does not allow us to identify different trajectories over time. Therefore, future applications in this data should include more complex models than better adjust the behavior of different patches over time, e.g. adding interactions between time and received dose.

The current analysis is assuming that the patches are independent; however, the spatial correlation between patches has been ignored. Failing to consider the spatial correlation between patches might result in biased estimates and smaller standard errors.⁵⁵ Adjustments to the proposed model to account for spatial correlation is an important topic for future research. For example, an iterative estimation process can be implemented by estimating the two-part model parameters first, and later, the residuals of that model are used to estimate the spatial correlation. Then, these two steps are repeated until convergence indicated by non-significant changes in the estimated likelihood over consecutive iterations.

6. Identification of different patterns of tissue deformation

In this section, the results from previous sections were used to address a clinical question. The two-part mixed-effects mixture model for zero-inflated longitudinal compositional data from the isodose patch definition provided a predicted time trajectory for the vector Y_{ij} for each patch (Chapter 5). It is expected that patches in distinct anatomic positions or at different distances from the tumor have different patterns of tissue changes. Then, the predicted composition was used to classify the obtained trajectories in different clusters to identify patch groups with similar patterns of tissue changes. The cluster analysis for longitudinal compositional data proposed by Bruno et al.²¹ was implemented to cluster the predicted trajectories of the composition. Since the interest was in mean levels of RILD, the vectors with the predicted trajectories ($\hat{Y}_{ij} = [\hat{Y}_{i3}, \hat{Y}_{i6}, \dots, \hat{Y}_{i24}]^T$) were the data under analysis for this section. Clustering by mean levels groups trajectories with similar predicted means over time even though they have different shapes.

The distance between two compositional trajectories for two different patches i and i' were calculated at each time point using Aitchison distance,⁵⁶ which is an extension of the Euclidean distance for composition data. This distance method has shown to preserve the requirements for compositional data (scale invariance, permutation invariance, perturbation invariance, and subcompositional dominance).⁹ Partitioning around medoids (PAM) algorithm was used for clustering the trajectories. In this method, a representative trajectory for each cluster is selected, and other trajectories are grouped in the cluster with the closest distance to the reference trajectory.

The separation distance between the resulting clusters was assessed using silhouette anal-

ysis.⁵⁷ For each patch, the average distance with patches in the same cluster $a(i)$ is calculated, as well as the average distance of patch i to all patches in the nearest cluster $b(i)$. Then, the silhouette coefficient was calculated for each patch as $s(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}}$. This coefficient ranges from -1 to 1, with 1 being the value that indicates that the patch is far away from the neighboring clusters. Values close to 0 are indicative that the sample is close to the boundary between two neighboring clusters, and negative values are indicators that that particular patch might have been assigned to the wrong cluster. The average silhouette coefficient is presented for the resulting clusters. Additionally, descriptive statistics for the clusters was performed to summarize the different observed patterns.

The idea of creating tissue deformation patterns have important clinical relevance in patient management. Hence the selection of the appropriate number of clusters was guided mainly by the clinical relevance. Final number of clusters was chosen based on which setup provided more interpretable and biologically feasible results.

6.1 Results

The Aitchison distance was calculated for each pair of patches i and i' at each time point, and the average distance across time was used as the distance metric for the clustering analysis. Figure 6.1 shows the average silhouette coefficient against the number of clusters by clustering using PAM algorithm for the reference patient. As observed, two clusters is the clustering configuration with the largest silhouette score, indicating that this configuration showed the best separation between clusters, compared with other configurations. However, when analyzing the characteristics of the patches within these two clusters, cluster 1 has all the patches that did not present RILD at any time point, while cluster 2 has 98.22% of the patches with RILD (Table 6.1). Therefore, this configuration will not give us enough information about the different patterns of tissue change beyond the percentage of lung damage.

The other studied clustering configurations (three to seven clusters) showed similar av-

Number of clusters	No RILD N=119	RILD N=169	Average silhouette coefficient
Two			0.75
Cluster 1	119 (100%)	3 (1.78%)	0.67
Cluster 2	0 (0.00%)	166 (98.22%)	0.81
Three			0.58
Cluster 1	119 (100%)	3 (1.78%)	0.61
Cluster 2	0 (0.00%)	87 (51.48%)	0.46
Cluster 3	0 (0.00%)	79 (46.75%)	0.65
Four			0.52
Cluster 1	56 (47.06%)	3 (1.78%)	0.18
Cluster 2	63(52.94%)	0 (0.00%)	0.75
Cluster 3	0 (0.00%)	87 (51.48%)	0.46
Cluster 4	0 (0.00%)	79 (46.75%)	0.65
Five			0.57
Cluster 1	56 (47.06%)	0 (0.00%)	0.61
Cluster 2	63(52.94%)	0 (0.00%)	0.63
Cluster 3	0 (0.00%)	87 (51.48%)	0.46
Cluster 4	0 (0.00%)	79 (46.75%)	0.65
Cluster 5	0 (0.00%)	3 (1.78%)	0.11
Six			0.52
Cluster 1	33 (27.73%)	2 (1.18%)	0.20
Cluster 2	40 (33.61%)	0 (0.00%)	0.60
Cluster 3	46 (38.66%)	0 (0.00%)	0.62
Cluster 4	0 (0.00%)	87 (51.48%)	0.46
Cluster 5	0 (0.00%)	79 (46.75%)	0.65
Cluster 6	0 (0.00%)	1 (0.60%)	0.00
Seven			0.51
Cluster 1	33 (27.73%)	2 (1.18%)	0.20
Cluster 2	40 (33.61%)	0 (0.00%)	0.60
Cluster 3	46 (38.66%)	0 (0.00%)	0.62
Cluster 4	0 (0.00%)	43 (25.44%)	0.45
Cluster 5	0 (0.00%)	59 (34.91%)	0.49
Cluster 6	0 (0.00%)	64 (37.87%)	0.61
Cluster 7	0 (0.00%)	1 (0.60%)	0.00

Table 6.1: Results of clustering with partition around medoids algorithm by RILD presence for the reference patient

verage silhouette coefficient (Figure 6.1). As observed in Figure 6.2, all these clustering configurations have a strong and positive relationship with the received median dose. Basically the different clusters are generated by different cutoffs of the received median dose with little

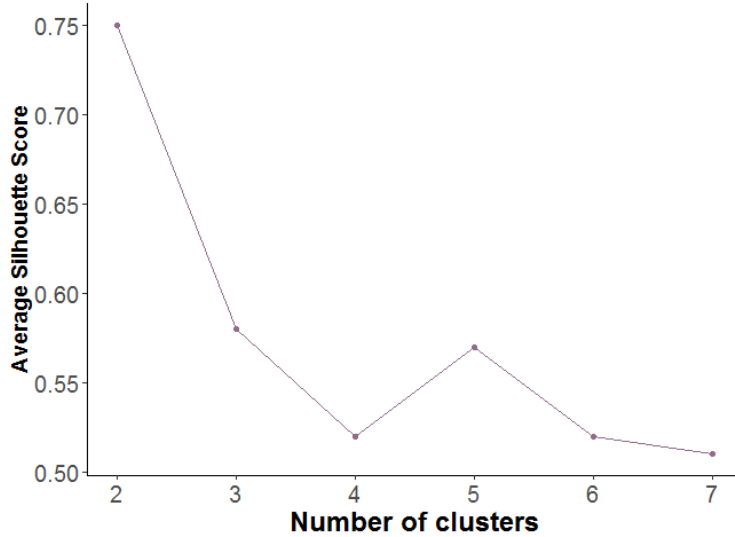


Figure 6.1: Average silhouette coefficient by number of clusters using partition around medoids algorithm for the reference patient

dose overlap between clusters. An important observation here is that in all configurations, except for five clusters, the three patches with the highest dose are classified in the first cluster. These three patches presented very high percentages of dense at 3 months, but this percentage was zero for all observations after 6 months. When looking at the predictions for these three patches, those are close to zero in all time points. Since these patches received the highest dose, they are highly likely to correspond to the tumor. For this reason, the identification of these clusters is important since large tissue changes are expected over time. Additionally, given the construction of the patches with the isodose lines, these patches combine the least number of voxels when compared to other patches.

Figure 6.3 shows the boxplots for the predicted parts of the composition (Dense, Hazy, Normal) by clustering configuration, while Figure 6.4 shows similar plots for the observed compositions. It is observed how all clustering configurations show a pattern to separate patches with and without RILD. The use of 3 and 4 clusters shows a good performance in identifying the magnitude of the composition and dose effect. Even though the clustering algorithm was performed using the predicted compositions, analysis of the distribution of the observed composition within the predicted clusters demonstrates the robustness of the

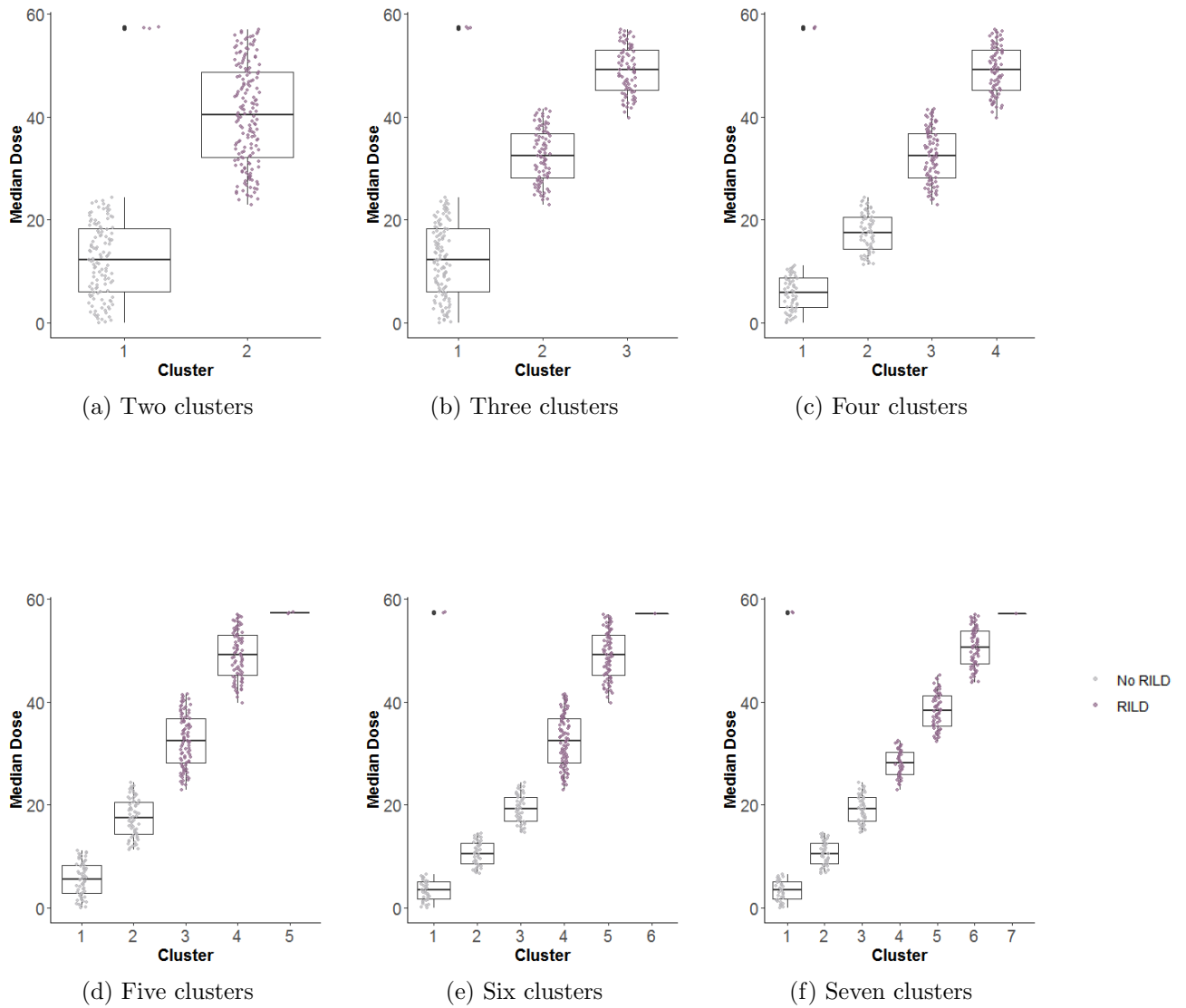


Figure 6.2: Boxplot of the median dose by the number of resulting clusters using partition around medoids algorithm for the reference patient

clustering method to the errors in the prediction. When analyzing the boxplots for the observed compositions (Figure 6.4), even though three clusters present a good separation of the intensity of the damage, the five clustering configuration identifies the patches with very extensive damage at the beginning of the follow-up that show a faster decrease over time, as mentioned before.

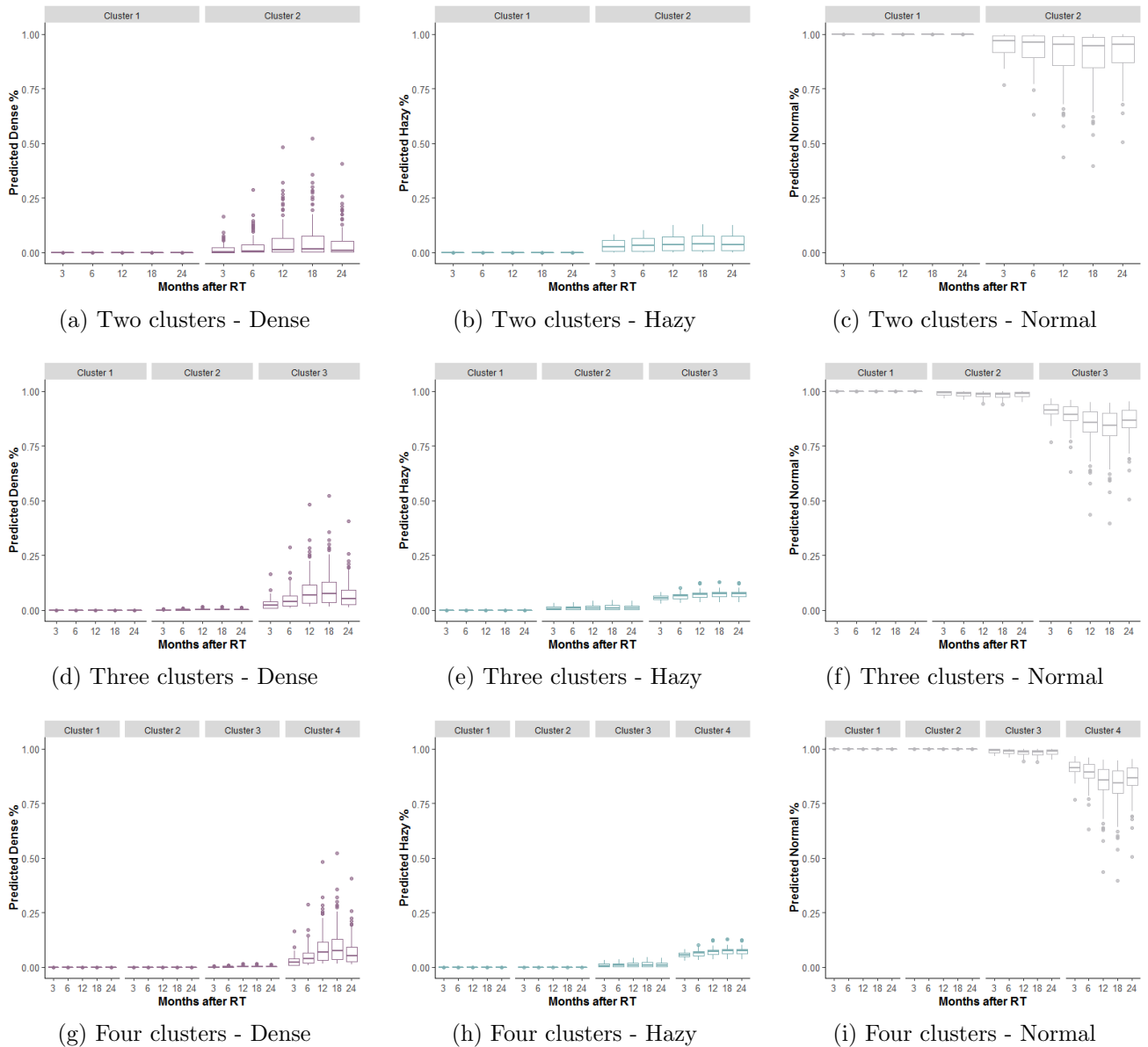


Figure 6.3: Boxplot of predicted composition (Dense,Hazy,Normal) by the number of resulting clusters using partition around medoids algorithm for the reference patient

6.2 Conclusion

This section shows an application of the cluster analysis to compositional data trajectories with a strong presence of zeros on them. Although the clustering was performed using the predicted trajectories, which do not have zeros, they have very small, close to zero, values. Even under this situation, the clustering analysis showed good internal validity

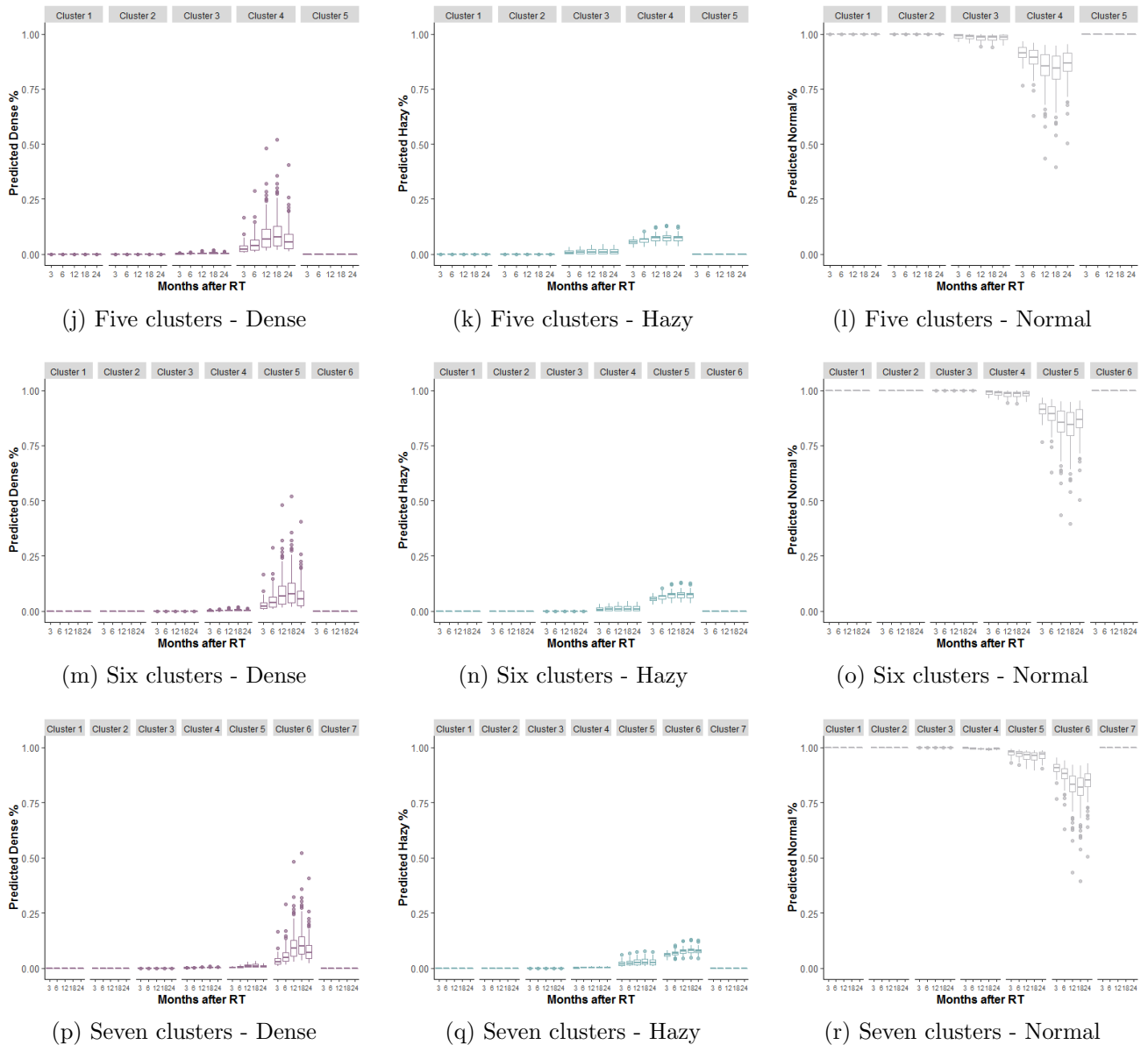


Figure 6.3: Boxplot of predicted composition (Dense,Hazy,Normal) by the number of resulting clusters using partition around medoids algorithm for the reference patient (cont.)

(high silhouette scores), and also showed good separation between cluster with and without RILD, by RILD intensity, and a strong association with radiation dose.

In this application, cluster configurations with two, three, and five clusters showed promise for identifying patch groups with similar trends in radiographic lung change following radiotherapy of lung cancer. However, these results are based on only one patient; therefore, the external validity of these findings is not clear, which implies that it is not

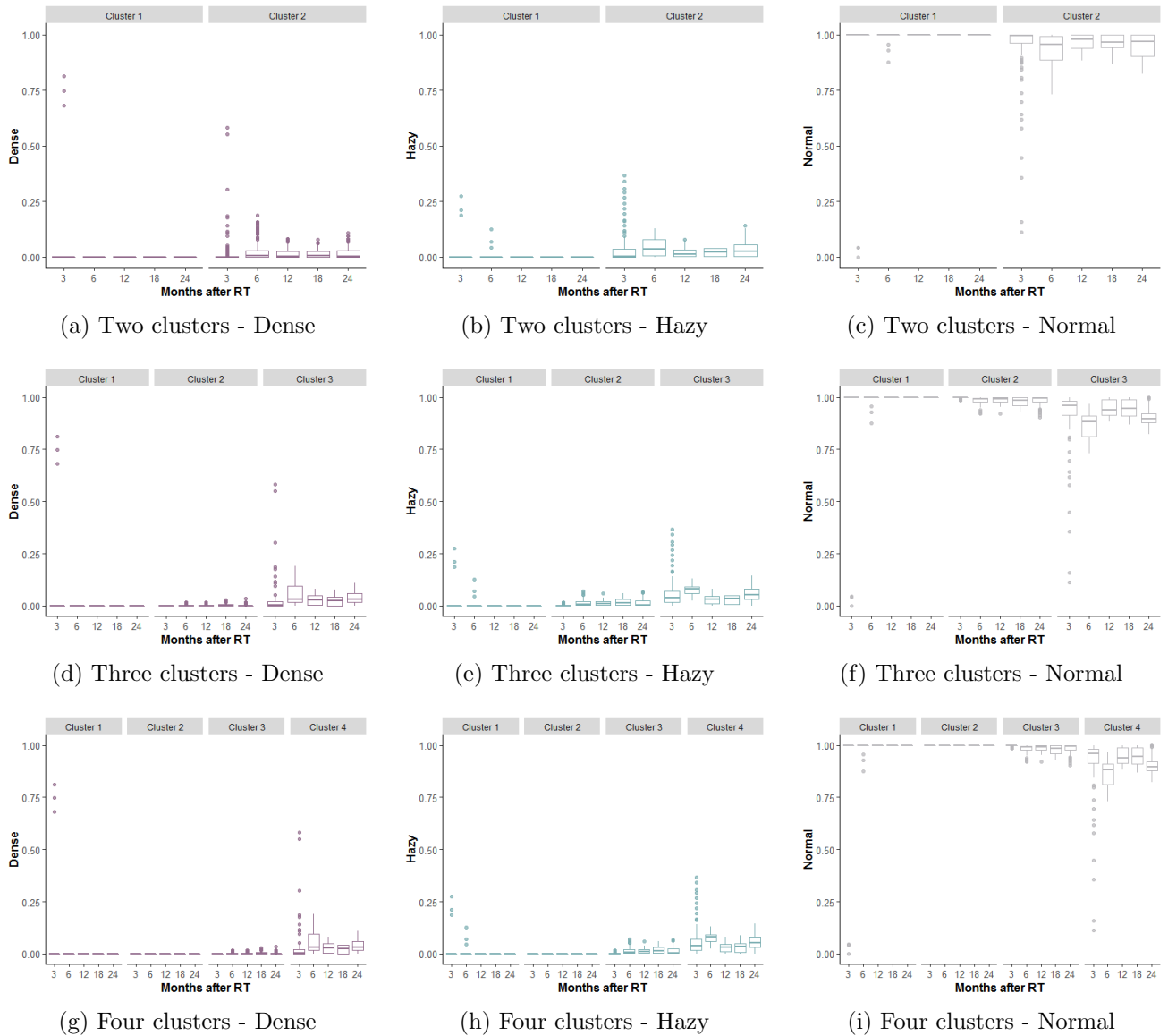


Figure 6.4: Boxplot of observed composition (Dense,Hazy,Normal) by the number of resulting clusters using partition around medoids algorithm for the reference patient

known if similar results will be obtained in other patients. For this reason, and taking into account that the selection of the clustering configuration was proposed to be performed by clinical relevance, it is necessary to study in greater depth the external validity, particularly with cluster settings of two, three, and five clusters.

The results presented in this section were obtained using the predicted values, which groups patches with similar predicted means over time, even though they have different

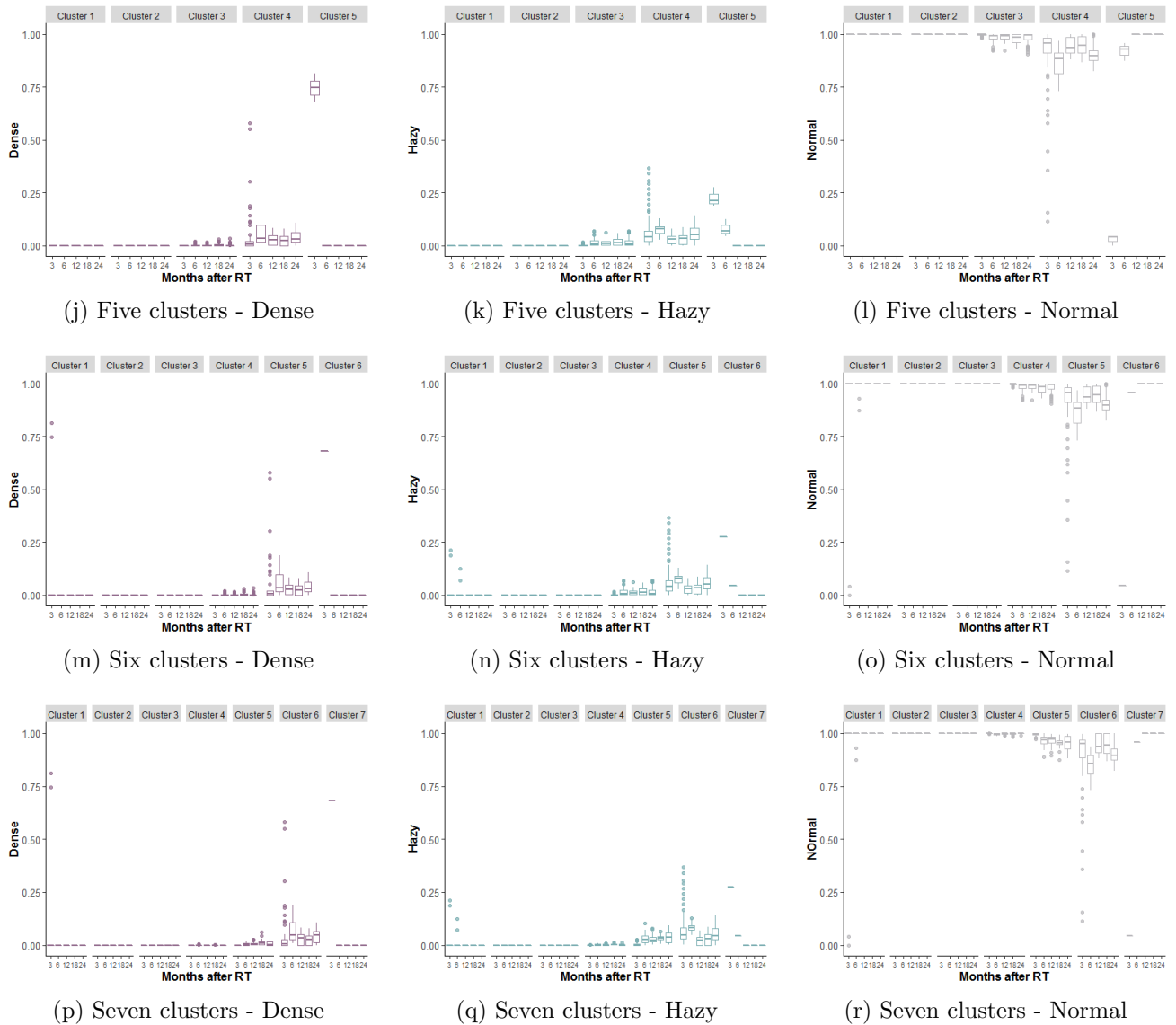


Figure 6.4: Boxplot of observed composition (Dense,Hazy,Normal) by the number of resulting clusters using partition around medoids algorithm for the reference patient (cont.)

shapes. It would also be interesting to perform cluster analysis using the predicted first derivatives, which would allow grouping patches with a similar rate of change over time. This analysis was not performed here since the model presented in Chapter 5 was simple and gave the same rate of change over time for all patches. Therefore, more complex models that better adjust the behavior of different patches over time, adding interactions between time and received dose, might yield better performance.

7. Association of tissue change with radiation pneumonitis in NSCLS patients

Radiation therapy (RT) is the standard treatment for medically inoperable patients in the early stages of NSCLC,²⁸ however, exposure of healthy tissue near the tumor to high doses of RT cause radiation-induced lung damage (RILD) in the vicinity of the tumor.³⁴ An association between RILD, radiation pneumonitis (RP), demographic and clinical factors has been suggested in other studies. However, the presence of differences in tissue changes over time after RT between patients with and without RP is still questionable. Using findings of previous chapters, a characterization of tissue changes in NSCLC patients by RP is studied in this chapter.

Data of two groups of patients, one with RP and another without RP have been collected. Each patient had three follow-up CT scans and otherwise similar tumor and treatment characteristics (Table 1.1). Each CT scan was manually contoured classifying the voxels in dense, hazy, or normal looking tissue based on intensity threshold. Afterward, deformable image registration was performed to ensure temporal correspondence.

Isodose patches were built for each set of CT scans, and patch-specific longitudinal compositional vectors for each patient were calculated. Predicted longitudinal compositions for each patch within patients were obtained using the two-part mixed-effects mixture model for zero-inflated longitudinal compositional data. Next, predicted compositions were classified using the patterns found in Chapter 6 for the reference patient, using two to five clusters. For each patient, the proportion of patches following each type of pattern was calculated. Finally, for each cluster configuration, a logistic regression model with RP diagnosis as outcome was performed with the percentage of patches in each cluster as predictors. Since

the distribution of patches among clusters is a compositional vector, *ilr* transformation was performed before running the model. For each cluster configuration, accuracy, sensitivity, specificity, and area under the curve (AUC) were calculated.

7.1 Results

Out of the 18 patients available for this analysis (Table 1.1), only 16 had more than two follow-up CT Scans, eight patients have RP, and eight did not have such a diagnosis. Table 7.1 shows the estimates of the fixed effects in the models for each patient. Large variability is observed in the estimates of the intercepts in the model of occurrence, as well as in the two outcomes of the intensity model. The estimation of the dose-effect has less variability, with similar direction and magnitude in most of the patients. There is also considerable variability in the estimates of time (Years, Years²). This could be indicating that the trend over time is different for each subject. Histograms of these parameter estimates by RP diagnosis did not show a trend toward differences between groups (Figure 7.1). The squared correlation and MASE between predicted and observed values are shown in Table 7.2. As demonstrated, the prediction performance is different for different patients and might be indicative of the necessity to include other clinical variables to improve the goodness-of-fit of the model.

Id	Occurrence model						Intensity model: Dense/Normal						Intensity model: Hazy/Normal											
	Intercept		Dose		Years		Intercept		Dose		Years		Intercept		Dose		Years							
	Est	SE	Est	SE	Est	SE	Est	SE	Est	SE	Est	SE	Est	SE	Est	SE	Est	SE						
1	-1429.44	1.6E-05	24.95	2.7E-07	1256.73	1.2E-05	-809.51	2.2E-05	-23.31	0.02	0.42	1.7E-03	-1.74	0.04	1.25	0.05	-18.09	0.03	0.32	1.4E-03	-4.97	0.07	4.37	0.03
2	39.28	0.31	0.13	0.01	-47.92	0.20	17.16	0.20	-7.08	0.34	0.06	0.01	4.02	1.09	-2.35	0.80	-4.97	0.27	0.08	7.3E-03	-0.56	0.82	0.03	0.61
3	-5.04	0.99	0.09	0.01	4.50	3.33	-3.69	2.54	-7.08	0.38	0.08	0.01	7.16	1.05	5.31	0.83	-6.48	0.36	0.05	4.6E-03	-5.17	0.87	3.20	0.70
4	-5.79	1.47	0.19	0.03	3.97	4.31	-4.66	3.45	-9.90	0.80	0.10	0.01	0.84	1.59	-0.28	1.22	-10.80	0.37	0.13	0.01	2.33	1.07	-1.57	0.81
5	-08.91	2.1E-04	0.76	1.0E-05	42.57	9.7E-05	9.36	2.0E-04	-8.91	0.23	0.11	0.00	-2.30	0.63	3.43	0.43	-9.59	0.24	0.12	0.24	0.24	0.69	2.20	0.47
6	-6.21	1.99	0.36	0.04	8.72	8.00	-10.12	6.51	-11.89	0.46	0.13	0.01	9.29	1.73	-5.65	1.48	-12.41	0.37	0.16	0.01	9.48	1.32	-6.70	1.17
7	-11.19	0.21	0.21	4.0E-03	98.40	0.30	-79.95	0.30	-3.12	0.22	0.06	3.0E-03	-3.85	0.60	1.90	0.48	-2.12	0.13	0.04	2.0E-03	-9.80	0.33	7.09	0.26
8	-2.16	1.96	0.19	0.02	-38.23	8.00	40.43	6.54	-11.28	0.51	0.12	4.9E-03	-2.96	2.14	6.62	1.68	-9.86	0.52	0.11	4.6E-03	-3.79	2.14	5.86	1.67
10	-11.93	2.71	0.23	0.03	-5.71	6.97	7.13	5.03	-11.37	8.05	0.12	0.15	4.60	3.29	-4.86	2.56	-12.90	4.20	0.13	0.07	13.65	4.18	-10.85	3.16
11	-128.97	0.68	12.17	3.4E-03	44.84	2.14	9.52	1.49	-7.20	0.20	0.17	1.8E-03	-7.09	0.69	6.87	0.51	-8.02	0.19	0.12	3.4E-03	-0.03	0.63	1.52	0.47
12	7.43	1.59	-0.11	0.04	11.35	1.18	-12.91	2.08	-5.52	0.24	0.08	2.0E-03	0.31	0.10	-0.15	0.81	-4.72	0.25	0.07	1.0E-03	-1.73	1.00	-0.07	0.80
13	86.15	5.4E-06	-7.26	4.4E-08	-48.84	3.7E-06	61.07	6.2E-06	-9.09	0.65	0.22	0.04	-1.94	2.29	2.02	1.76	-8.16	0.77	-0.05	0.05	-0.35	2.60	0.74	2.00
14	-0.87	0.38	0.34	0.01	92.15	1.04	-86.95	0.78	-6.49	0.22	0.12	1.9E-03	-5.01	0.69	3.60	0.50	-6.79	0.19	0.09	1.7E-03	0.94	0.63	-1.44	0.45
15	-23.80	2.52	0.36	0.02	-8.19	8.53	17.21	6.19	-12.20	1.56	0.18	0.03	-0.87	1.96	1.49	1.54	-9.81	0.98	0.12	0.02	-0.90	1.51	0.08	1.18
17	-3.72	9.36	3.10	7.59	1.47	10.71	0.09	8.05	-11.28	0.15	0.11	2.1E-03	12.29	0.51	-6.94	0.40	-8.60	0.18	0.12	2.5E-03	2.28	0.62	-1.00	0.45
18	-21.37	22.27	1.35	0.17	-32.60	67.13	20.29	44.65	-11.44	7.6E-06	0.13	1.0E-07	9.73	1.52	-6.79	1.23	-8.54	0.26	0.11	3.5E-03	4.28	0.81	-3.34	0.66

Table 7.1: Parameter estimates and standard errors of the fixed effects for modeling RILD using isodose lines patch definition

Est:Estimate

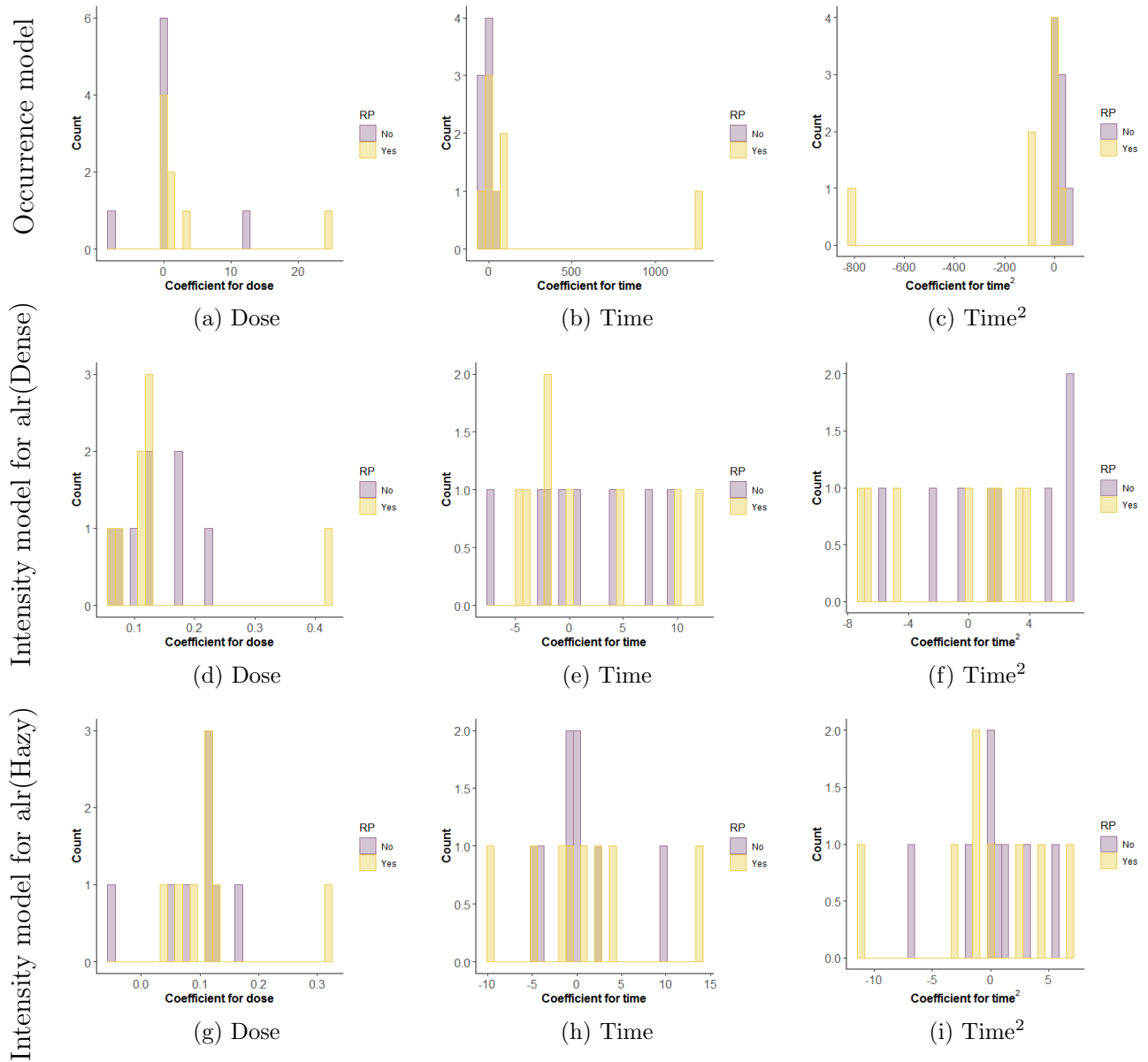


Figure 7.1: Histograms of parameter estimates of the fixed effects for modeling RILD using isodose lines patch definition by radiation pneumonitis (RP) diagnosis

The estimated baseline composition, when the patch received zero dose for each patient is presented in Table 7.3. Also, the shift in the composition from the baseline by a unit change in each covariate, when the other covariates are unchanged, is presented. It is observed how despite the differences in the magnitude and direction of dose estimates in Table 7.1, when transforming the estimates to the impact in the compositional vector, the effect across patients is similar. However, the effect of time does change considerably between patients.

Id	Dense		Hazy	
	Correlation ²	MASE	Correlation ²	MASE
1	0.24	0.98	0.21	0.92
2	0.34	0.90	0.34	1.01
3	0.31	0.90	0.41	0.76
4	0.37	0.99	0.39	0.99
5	0.66	0.33	0.54	0.34
6	0.65	0.58	0.38	0.84
7	0.52	0.35	0.44	0.40
8	0.67	0.37	0.14	0.40
10	0.47	0.94	2.1E-05	0.99
11	0.72	0.51	0.71	0.37
12	0.02	0.96	0.03	0.77
13	0.76	0.69	0.71	0.65
14	0.56	0.82	0.78	0.54
15	0.31	0.94	0.32	0.98
17	0.82	0.33	0.81	0.51
18	2.2E-04	1.00	0.18	0.95

Table 7.2: Correlation and mean absolute scaled error (MASE) between observed and predicted data using isodose lines patch definition

Id	Baseline			Dose			Years			Years ²		
	Dense	Hazy	Normal	Dense	Hazy	Normal	Dense	Hazy	Normal	Dense	Hazy	Normal
1	0.00	0.00	1.00	0.39	0.35	0.26	0.15	0.01	0.85	0.04	0.95	0.01
2	0.00	0.01	0.99	0.34	0.34	0.32	0.97	0.01	0.02	0.04	0.49	0.47
3	0.00	0.00	1.00	0.34	0.34	0.32	0.00	0.01	0.99	0.89	0.11	0.00
4	0.00	0.00	1.00	0.34	0.35	0.31	0.17	0.76	0.07	0.39	0.11	0.51
5	0.00	0.00	1.00	0.34	0.35	0.31	0.06	0.29	0.64	0.75	0.22	0.02
6	0.00	0.00	1.00	0.34	0.35	0.30	0.45	0.55	0.00	0.00	0.00	1.00
7	0.04	0.10	0.86	0.34	0.34	0.32	0.02	0.00	0.98	0.01	0.99	0.00
8	0.00	0.00	1.00	0.35	0.34	0.31	0.05	0.02	0.93	0.68	0.32	0.00
10	0.00	0.00	1.00	0.35	0.35	0.31	0.00	1.00	0.00	0.01	0.00	0.99
11	0.00	0.00	1.00	0.36	0.34	0.30	0.00	0.49	0.51	0.99	0.01	0.00
12	0.00	0.01	0.99	0.34	0.34	0.32	0.54	0.07	0.39	0.31	0.34	0.36
13	0.00	0.00	1.00	0.39	0.30	0.31	0.08	0.38	0.54	0.71	0.20	0.09
14	0.00	0.00	1.00	0.35	0.34	0.31	0.00	0.72	0.28	0.97	0.01	0.03
15	0.00	0.00	1.00	0.36	0.34	0.30	0.23	0.22	0.55	0.68	0.17	0.15
17	0.00	0.00	1.00	0.34	0.35	0.31	1.00	0.00	0.00	0.00	0.27	0.73
18	0.00	0.00	1.00	0.35	0.34	0.31	1.00	0.00	0.00	0.01	0.03	0.96

Table 7.3: Additive logistic transform estimates of the model applied to the composition given that there is some RILD presence using isodose lines patch definition

Predicted compositions were classified using the patterns found in Chapter 6 for the reference patient, using two to five clusters. Table 7.4 shows the average silhouette coefficient for

each patient data set by number of clusters. Similar to the results with the reference patient (Figure 6.1), two clusters was the clustering configuration with the largest silhouette score for almost all patients, indicating that this configuration has the best separation between clusters. The other clustering configurations (three to five clusters) showed a similar average silhouette score.

The proportion of patches following each type of pattern is presented in Table 7.5. Differences between the distribution of cluster membership is observed between patients. These percentages were transformed using the *ilr* transformation and were used as predictors of RP. A summary of performance of the clustering configurations to classify patients with and without RP is shown in Table 7.6. Even though the configuration with two clusters has the highest average silhouette coefficient, this is the configuration with the most inferior performance. Among the other options, four and five clusters seem to have better performance identifying patients with RP.

Id	Number of Clusters			
	Two	Three	Four	Five
1	0.53	0.41	0.38	0.39
2	0.75	0.53	0.57	0.56
3	0.54	0.28	0.39	0.39
4	0.66	0.49	0.49	0.48
5	0.27	0.18	0.34	0.65
6	0.67	0.5	0.53	0.51
7	0.88	0.72	0.68	0.71
8	0.61	0.43	0.4	0.39
10	0.59	0.52	0.54	0.54
11	0.81	0.52	0.52	0.53
12	0.86	0.44	0.44	0.43
13	0.3	0.24	0.23	0.32
14	0.79	0.46	0.49	0.49
15	0.41	0.33	0.47	0.41
17	0.71	0.62	0.61	0.6
18	0.73	0.42	0.41	0.46
Average	0.63	0.44	0.47	0.49

Table 7.4: Average silhouette coefficient by number of clusters using partition around medoids algorithm with the isodose lines patch definition

Id	Two Clusters			Three Cluster			Four Clusters			Five Clusters												
	1	2	1	1	2	3	1	2	3	4	1	2	3	4	5							
1	202	(68.24%)	94	(31.76%)	202	(68.24%)	33	(11.15%)	61	(20.61%)	200	(67.57%)	2	(0.68%)	33	(11.15%)	61	(20.61%)	5	(1.69%)	195	(65.88%)
2	55	(18.46%)	243	(81.54%)	47	(15.77%)	49	(16.44%)	202	(67.79%)	34	(11.41%)	20	(6.71%)	42	(14.09%)	202	(67.79%)	2	(0.67%)	32	(10.74%)
3	77	(21.81%)	276	(78.19%)	57	(16.15%)	235	(66.57%)	61	(17.28%)	61	(17.28%)	87	(24.65%)	205	(58.07%)	61	(17.28%)	0	(0.00%)	0	(0.00%)
4	90	(32.14%)	190	(67.86%)	83	(29.64%)	170	(60.71%)	27	(9.64%)	0	(0.00%)	92	(32.86%)	161	(57.50%)	27	(9.64%)	0	(0.00%)	0	(0.00%)
5	193	(72.28%)	74	(27.72%)	191	(71.54%)	68	(25.47%)	8	(3.00%)	140	(52.43%)	53	(19.85%)	66	(24.72%)	8	(3.00%)	63	(23.60%)	77	(28.84)
6	50	(21.37%)	184	(78.63%)	45	(19.23%)	113	(48.29%)	76	(32.48%)	2	(0.85%)	51	(21.79%)	105	(44.87%)	76	(32.48%)	0	(0.00%)	2	(0.85)
7	30	(9.20%)	296	(90.8%)	30	(9.20%)	31	(9.51%)	265	(81.29%)	30	(9.20%)	1	(0.31%)	30	(9.20%)	265	(81.29%)	16	(4.91%)	14	(4.29)
8	128	(42.11%)	176	(57.89%)	121	(39.80%)	151	(49.67%)	32	(10.53%)	59	(19.41%)	85	(27.96%)	128	(42.11%)	32	(10.53%)	59	(19.41%)	0	(0.00%)
10	171	(59.17%)	118	(40.83%)	162	(56.06%)	112	(38.75%)	15	(5.19%)	92	(31.83%)	83	(28.72%)	99	(34.26%)	15	(5.19%)	92	(31.83%)	0	(0.00%)
11	47	(17.09%)	228	(82.91%)	45	(16.36%)	87	(31.64%)	143	(52.00%)	41	(14.91%)	5	(1.82%)	86	(31.27%)	143	(52.00%)	5	(1.82%)	36	(13.09)
12	1	(0.33%)	302	(99.67%)	1	(0.33%)	55	(18.15%)	247	(81.52%)	0	(0.00%)	247	(81.52%)	55	(18.15%)	247	(81.52%)	0	(0.00%)	0	(0.00%)
13	247	(82.06%)	54	(17.94%)	247	(82.06%)	54	(17.94%)	0	(0.00%)	246	(81.73%)	1	(0.33%)	54	(17.94%)	0	(0.00%)	17	(5.65%)	229	(76.08)
14	42	(12.84%)	285	(87.16%)	40	(12.23%)	85	(25.99%)	202	(61.77%)	27	(8.26%)	14	(4.28%)	84	(25.69%)	202	(61.77%)	3	(0.92%)	24	(7.34)
15	217	(70.45%)	91	(29.55%)	205	(66.56%)	60	(19.48%)	43	(13.96%)	150	(48.70%)	79	(25.65%)	36	(11.69%)	43	(13.96%)	105	(34.09%)	45	(14.61)
17	3	(0.98%)	302	(99.02%)	1	(0.33%)	101	(33.11%)	203	(66.56%)	1	(0.33%)	203	(66.56%)	99	(32.46%)	203	(66.56%)	1	(0.33%)	0	(0.00%)
18	101	(33.33%)	202	(66.67%)	99	(32.67%)	73	(24.09%)	131	(43.23%)	82	(27.06%)	20	(6.60%)	70	(23.10%)	131	(43.23%)	42	(13.86%)	40	(13.20)

Table 7.5: Results of clustering with partition around medoids algorithm using isodose lines patch definition

Number of Clusters	Accuracy	Sensitivity	Specificity	AUC
Two	0.50	0.50	0.50	0.63 (0.31, 0.94)
Three	0.69	0.63	0.75	0.64 (0.33, 0.95)
Four	0.69	0.75	0.63	0.66 (0.34, 0.97)
Five	0.69	0.63	0.75	0.66 (0.34, 0.97)

Table 7.6: Model discrimination of radiation pneumonitis by number of clusters using partition around medoids algorithm with the isodose lines patch definition

7.2 Conclusion

This section shows an application of the isodose-patch definition, the two-part mixed-effects mixture model for zero-inflated longitudinal compositional data, and the cluster analysis to compositional data trajectories in a data set of 16 NSCLC patients. For these patients, the number of patches ranges from 234 to 353, and the percentage of patches with RILD goes from 21.59% to 99.34%.

The coefficients for some parameters vary considerably (intercept and time), therefore in future analysis of this data set, it is suggested to consider random effects for these parameters in the subject level to accommodate such variability. It is important to highlight that estimations on the intensity model (second part) seem more stable among patients than in the occurrence model. This result is consistent with what was observed in Chapter 4.

The method for estimating the parameters ignores the spatial correlation between patches. Augmenting this method by spatial correlation might result in more consistent estimates. Moreover, as shown in Chapters 4 and 5, the two-part model has some other limitations in small samples (around 300 patches) with few follow-ups. All the patients in this data set present this situation and thus suffering from the same limitations. Despite these limitations, dose estimates appear consistent when transforming the data to the original units in the compositional data.

In this application, cluster configurations with four and five clusters showed promise for identifying patients with RP. However, these results are based on a small sample of patients,

and it was not adjusted by other covariates. A larger data set and inclusion of other covariates should benefit the study of the association between tissue change and RP. As a first attempt to associate the differences in tissue change between patients with RP, this shows a lot of promise.

8. Discussion

Longitudinal compositional data, with a peak at zero, occur in many applications in health sciences. Different sources of correlation in the data are observed between subjects over time and between parts of the compositional vector. Additionally, the presence of singular zero represents a particular challenge in the analysis of such data.

Several approximations have been proposed to analyze zero-inflated longitudinal data, from Aitchison's proposal in 1955,⁵⁸ going through various applications of likelihood-based models,⁵⁹ to proposals using Bayesian methods that do not require making asymptotic approximations.⁶⁰ However, only a few approaches have been applied to analyzing zero-inflated longitudinal compositional data.^{16,61}

A new approach for the analysis of zero-inflated longitudinal compositional data, which additionally shows bimodal distributions of the components, is proposed in this work. Here a two-part mixed-effects mixture model was proposed. This model is fully parametric and uses asymptotic approximations to estimate the parameters. The likelihood was approximated using a sixth-order multivariate Laplace method, and even though a higher-order could have been used, empirical evidence shows that it would not improve the accuracy.⁶² The EM algorithm was used to estimate the probability of the mixture model, and the approximate Fisher scoring procedure was implemented to estimate other parameters in the model. Three different sets of empirical Bayes estimates of the random effects were implemented, standard, truncated, and Pareto. No big differences were observed in the estimations between the three methods, however, the predictions with the Pareto estimates were slightly better than with the other methods.

A simulation study showed that the proposed method performs reasonably for estimation of the fixed effects, and their variability. Also, the average time that the procedure takes for the estimation seems feasible for statistical analysis. However, the model produced biased estimates of the nuisance parameters in the model. As discussed in Chapter 4, these results might be caused by the implementation of the step-halving procedure that forces a decrease in the deviance at each iteration. This procedure can be limiting the changes in the estimates and causing false convergence. Changes in this procedure, like using it only on the fixed effects or other approximations to improve convergence, should be studied to improve model performance.

Currently, the implemented code for running the proposed method considers random intercepts for both parts of the model. Nevertheless, future expansion of the model should accommodate random slopes for important factors, as well as other random effects that can handle cluster or multilevel data. Also, the estimated coefficients in this method do not represent the population-average effect. Therefore, future extensions of this work can attempt the estimation of such effects based on the current estimates.

The proposed model was used to address a clinical question. The aim was to build a statistical model for stochastic radiographic lung change following radiotherapy of lung cancer, and to explore how the results could be used to characterize patients with radiation pneumonitis. From the practitioner's point of view, the use of medical imaging has become quite sophisticated and continues to evolve into a very accurate, non-invasive tool for medical diagnosis. In patients with lung cancer, medical images of different modalities are easy to access and a non-invasive tool to monitor radiographic changes following radiotherapy. RT of the lung shows a clear footprint of the dose distribution in post-treatment scans. Although the process of lung tissue changes following radiation is clearly visible, predictive models of such changes are not available. Part of the reason why statistical models for longitudinal image sets are underdeveloped is the complexity of the data and difficulty of accurate image registration. First, image data are often extremely large data sets that

require extensive computational resources to analyze. Additionally, in the case of following LC patients after RT, the temporal integrity of the data is questionable, given the tissue changes in the lung after RT. For this reason, the approaches presented in this study, grouping voxels into larger subvolumes called patches, represents an improvement over aggregative approach that combines the entire image into few summary measures or voxel-wise model.

The two-part mixed-effects mixture model for zero-inflated longitudinal compositional data was applied to the data of one NSCLC patient for the purpose of modeling the intensity of RILD overtime after RT. The proposed method performed well in simulation and was applied successfully to the clinical data, and when analyzing the results in the compositional data framework, estimates seemed to be robust to the model limitations. However, the presented model did not adjust by the spatial correlation between patches, neither included random slopes for time effects, both important features for future studies in these data.

Bibliography

- [1] Aitchison J. The statistical analysis of compositional data. London: Chapman and Hall Ltd; 1986.
- [2] Chayes F. On correlation between variables of constant sum. *Journal of Geophysical Research* (1896-1977). 1960;65(12):4185–4193. Available from: <https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/JZ065i012p04185>.
- [3] Aitchison J, J Egozcue J. Compositional Data Analysis: Where Are We and Where Should We Be Heading? *Mathematical Geology*. 2005;37(7):829–850.
- [4] Pawlowsky-Glahn V, Egozcue JJ, Tolosana-Delgado R. Modeling and analysis of compositional data. 1st ed. Chichester, United Kingdom: John Wiley & Sons Inc; 2015.
- [5] Warton DI, Hui FKC. The arcsine is asinine: the analysis of proportions in ecology. *Ecology*. 2011;92(1):3–10. Available from: <https://esajournals.onlinelibrary.wiley.com/doi/abs/10.1890/10-0340.1>.
- [6] Egozcue JJ, Pawlowsky-Glahn V, Mateu-Figueras G, Barceló-Vidal C. Isometric Logratio Transformations for Compositional Data Analysis. *Mathematical Geology*. 2003;35(3):279–300.
- [7] Pawlowsky-Glahn V, Buccianti A. Compositional Data Analysis: Theory and Applications. 1st ed. Chichester: John Wiley & Sons, Ltd; 2011.
- [8] Mateu-Figueras G, Pawlowsky-Glahn V, Barceló-Vidal C. The additive logistic skew-

- normal distribution on the simplex. *Stochastic Environmental Research and Risk Assessment*. 2005 aug;19(3):205–214.
- [9] Martín-Fernández JA, Barceló-Vidal C, Pawłowsky-Glahn V. Measures of difference for compositional data and hierarchical clustering methods. *Proceedings of IAMG*. 1998;98(1):526–531.
- [10] Greenacre MJ. *Compositional data analysis in practice*. Boca Raton, FL: Taylor & Francis Group; 2018.
- [11] Fry JM, Fry TRL, McLaren KR. Compositional Data Analysis and Zeros in Micro Data. *Applied Economics*. 2000 jun;32(8):953–959.
- [12] Aitchison J, Kay JW. Possible solution of some essential zero problems in compositional data analysis. In: Thió-Henestrosa S, Martín-Fernández JA, editors. *CoDaWork'03*. Girona, Spain: Universitat de Girona; 2003. p. 1–6.
- [13] Bacon-Shone J. Modelling structural zeros in compositional data. In: Thió-Henestrosa S, Martín-Fernández JA, editors. *CoDaWork'03*. Girona, Spain: Universitat de Girona; 2003. p. 1–4.
- [14] Martín-Fernández JA, Barceló-Vidal C, Pawłowsky-Glahn V. Dealing with Zeros and Missing Values in Compositional Data Sets Using Nonparametric Imputation. *Mathematical Geology*. 2003;35(3):253–278.
- [15] Martín-Fernández JA, Hron K, Templ M, Filzmoser P, Palarea-Albaladejo J. Model-based replacement of rounded zeros in compositional data: Classical and robust approaches. *Computational Statistics & Data Analysis*. 2012 sep;56(9):2688–2704.
- [16] Chen EZ, Li H. A two-part mixed-effects model for analyzing longitudinal microbiome compositional data. *Bioinformatics*. 2016;32(17):2611–2617.

- [17] Shimizu TKO, Achcar JA, Tarumoto MH. A Bayesian analysis for compositional data in a repeated measures experiment : a case study. In: 58^a Reunião Anual da Rbras e 15^o Simpósio de Estatística Aplicada à Experimentação Agronômica. Campina Grande; 2013. p. 1–5. Available from: <https://etd.library.emory.edu/concern/etds/2514nm23c?locale=en>.
- [18] Venieri C. Local polynomial regression for compositional data. D’Annunzio of Chieti–Pescara; 2013. Available from: <https://www.researchgate.net/publication/271826420>.
- [19] Di Marzio M, Panzera A, Venieri C. Non-parametric regression for compositional data. *Statistical Modelling: An International Journal*. 2015 apr;15(2):113–133.
- [20] Van der Merwe S. A method for Bayesian regression modelling of composition data [Journal Article]. *South African Statistical Journal*. 2019;53(1):55–64. Available from: <https://journals.co.za/content/journal/10520/EJC-14af74f1cc>.
- [21] Bruno F, Cocchi D, Greco F. Clustering compositional data trajectories: The case of particulate matter in the lower troposphere. *Environmetrics*. 2011;22(8):975–984.
- [22] Tait P, McNicholas P. Probabilistic Partial Least Squares Regression Applied to Longitudinal and Cross-Sectional Compositional Data. In: *Joint Statistical Meetings - American Statistical Association*. Vancouver, Canada; 2018. .
- [23] Mueller H, Dai X. Non- and Semiparametric Methods to Accommodate Dependency and Heterogeneity in Complex Data. In: *Joint Statistical Meetings - American Statistical Association*. Vancouver, Canada; 2018. .
- [24] Siegel RL, Miller KD, Jemal A. Cancer statistics, 2018. *CA: A Cancer Journal for Clinicians*. 2018 jan;68(1):7–30.

- [25] Cancer org. Lung Cancer; 2018. Available from: <https://www.cancer.org/cancer/lung-cancer.html>.
- [26] Wakelee H, Kelly K, Edelman MJ. 50 Years of progress in the systemic therapy of non-small cell lung cancer. American Society of Clinical Oncology educational book American Society of Clinical Oncology Annual Meeting. 2014:177–89.
- [27] Grove O, Berglund AE, Schabath MB, Aerts HJWL, Dekker A, Wang H, et al. Quantitative Computed Tomographic Descriptors Associate Tumor Shape Complexity and Intratumor Heterogeneity with Prognosis in Lung Adenocarcinoma. PLOS ONE. 2015 mar;10(3):e0118261.
- [28] Jain V, Berman AT. Radiation pneumonitis: Old problem, new tricks. Cancers. 2018 jul;10(7):222.
- [29] Baskar R, Dai J, Wenlong N, Yeo R, Yeoh KW. Biological response of cancer cells to radiation treatment. Frontiers in molecular biosciences. 2014;1:24.
- [30] Linda A, Trovo M, Bradleyc JD. Radiation injury of the lung after stereotactic body radiation therapy (SBRT) for lung cancer: A timeline and pattern of CT changes. European Journal of Radiology. 2011;79(1):147–154.
- [31] Matsuo Y, Shibuya K, Nakamura M, Narabayashi M, Sakanaka K, Ueki N, et al. Dose–Volume Metrics Associated With Radiation Pneumonitis After Stereotactic Body Radiation Therapy for Lung Cancer. International Journal of Radiation Oncology, Biology, Physics. 2012 jul;83(4):e545–e549.
- [32] Trovo M, Linda A, El Naqa I, Javidan-Nejad C, Bradley J. Early and late lung radiographic injury following stereotactic body radiation therapy (SBRT). Lung Cancer. 2010 jul;69(1):77–85.

- [33] Baker R, Han G, Sarangkasiri S, DeMarco M, Turke C, Stevens CW, et al. Clinical and Dosimetric Predictors of Radiation Pneumonitis in a Large Series of Patients Treated With Stereotactic Body Radiation Therapy to the Lung. *International Journal of Radiation Oncology*Biology*Physics*. 2013 jan;85(1):190–195.
- [34] Dunne EM, Fraser IM, Liu M. Stereotactic body radiation therapy for lung, spine and oligometastatic disease: current evidence and future directions. *Annals of translational medicine*. 2018 jul;6(14):283.
- [35] Veiga C, Landau D, McClelland JR, Ledermann JA, Hawkes D, Janes SM, et al. Long term radiological features of radiation-induced lung damage. *Radiotherapy and Oncology*. 2018 feb;126(2):300–306.
- [36] Marks LB, Bentzen SM, Deasy JO, Kong FMS, Bradley JD, Vogelius IS, et al. Radiation dose-volume effects in the lung. *International journal of radiation oncology, biology, physics*. 2010 mar;76(3 Suppl):S70–6.
- [37] Kalman NS, Hugo GD, Kahn JM, Zhao SS, Jan N, Mahon RN, et al. Interobserver reliability in describing radiographic lung changes after stereotactic body radiation therapy. *Advances in Radiation Oncology*. 2018 may;0(0).
- [38] Palma DA, van Sörnsen de Koste J, Verbakel WFAR, Vincent A, Senan S. Lung Density Changes After Stereotactic Radiotherapy: A Quantitative Analysis in 50 Patients. *International Journal of Radiation Oncology*Biology*Physics*. 2011 nov;81(4):974–978.
- [39] Ghobadi G, Wiegman EM, Langendijk JA, Widder J, Coppes RP, van Luijk P. A new CT-based method to quantify radiation-induced lung damage in patients. *Radiotherapy and Oncology*. 2015 oct;117(1):4–8.
- [40] Defraene G, van Elmpt W, Crijs W, De Ruyscher D. Regional variability in radiation-induced lung damage can be predicted by baseline CT numbers. *Radiotherapy and Oncology*. 2017 feb;122(2):300–306.

- [41] Monti S, Pacelli R, Cella L, Palma G. Inter-patient image registration algorithms to disentangle regional dose bioeffects. *Scientific Reports*. 2018 dec;8(1):4915.
- [42] Diot Q, Kavanagh B, Scheffer T, Gaspar L, Stuhr K, Miften M. Regional normal lung tissue density changes in patients treated with stereotactic body radiation therapy for lung tumors. *International Journal of Radiation Oncology Biology Physics*. 2012 nov;84(4):1024–1030.
- [43] Calinski T, Harabasz J. A dendrite method for cluster analysis. *Communications in Statistics*. 1974;3(1):1–27.
- [44] Olsen MK, Schafer JL. A Two-Part Random-Effects Model for Semicontinuous Longitudinal Data. *Journal of the American Statistical Association*. 2001;96(454):730–745. Available from: <https://doi.org/10.1198/016214501753168389>.
- [45] Gelman A, Carlin JB, Stern HS, Rubin DB. *Bayesian Data Analysis*. Chapman and Hall/CRC; 1995.
- [46] Ionides EL. Truncated Importance Sampling. *Journal of Computational and Graphical Statistics*. 2008;17(2):295–311. Available from: <http://www.jstor.org/stable/27594308>.
- [47] Vehtari A, Simpson D, Gelman A, Yao Y, Gabry J. *Pareto Smoothed Importance Sampling*; 2015.
- [48] R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria; 2019. Available from: <https://www.R-project.org/>.
- [49] Collins LM, Schafer JL, Kam CM. A comparison of inclusive and restrictive strategies in modern missing data procedures. *Psychological Methods*. 2001;6(4):330–351.
- [50] Marschner IC. *glm2: Fitting Generalized Linear Models with Convergence Problems*. *The R Journal*. 2011;3(2):12–15.

- [51] Garcia Portugues E. Goodness-of-fit tests for distribution models; 2020. Available from: <https://bookdown.org/egarpor/NP-UC3M/nptest.html>.
- [52] Verbeke G, Lesaffre E. The effect of misspecifying the random-effects distribution in linear mixed models for longitudinal data. *Computational Statistics & Data Analysis*. 1997;23(4):541 – 556. Available from: <http://www.sciencedirect.com/science/article/pii/S0167947396000473>.
- [53] Butler SM, Louis TA. Random effects models with non-parametric priors. *Statistics in Medicine*. 1992;11(14-15):1981–2000. Available from: <https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.4780111416>.
- [54] Neuhaus JM, Kalbfleisch JD, Hauck WW. A Comparison of Cluster-Specific and Population-Averaged Approaches for Analyzing Correlated Binary Data. *International Statistical Review / Revue Internationale de Statistique*. 1991;59(1):25–35. Available from: <http://www.jstor.org/stable/1403572>.
- [55] Hodges JS, Reich BJ. Adding Spatially-Correlated Errors Can Mess Up the Fixed Effect You Love. *The American Statistician*. 2010;64(4):325–334. Available from: <https://doi.org/10.1198/tast.2010.10052>.
- [56] Aitchison J. On criteria for measures of compositional difference. *Mathematical Geology*. 1992 may;24(4):365–379.
- [57] Rousseeuw PJ. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*. 1987 nov;20(C):53–65.
- [58] Aitchison J. On the Distribution of a Positive Random Variable Having a Discrete Probability Mass at the Origin*. *Journal of the American Statistical Association*. 1955;50(271):901–908. Available from: <https://doi.org/10.1080/01621459.1955.10501976>.

- [59] Farewell VT, Long DL, Tom BDM, Yiu S, Su L. Two-Part and Related Regression Models for Longitudinal Data. *Annual Review of Statistics and Its Application*. 2017 mar;4(1):283–315.
- [60] Neelon BH, O’Malley AJ, Normand SLT. A Bayesian model for repeated measures zero-inflated count data with application to outpatient psychiatric service use. *Statistical Modelling*. 2010 dec;10(4):421–439.
- [61] Chai H, Jiang H, Lin L, Liu L. A marginalized two-part Beta regression model for microbiome compositional data. *PLOS Computational Biology*. 2018 jul;14(7):e1006329. Available from: <https://dx.plos.org/10.1371/journal.pcbi.1006329>.
- [62] Raudenbush SW, Yang ML, Yosef M. Maximum Likelihood for Generalized Linear Models with Nested Random Effects via High-Order, Multivariate Laplace Approximation. *Journal of Computational and Graphical Statistics*. 2000 mar;9(1):141.

Appendix

A Clustering results for the reference patient

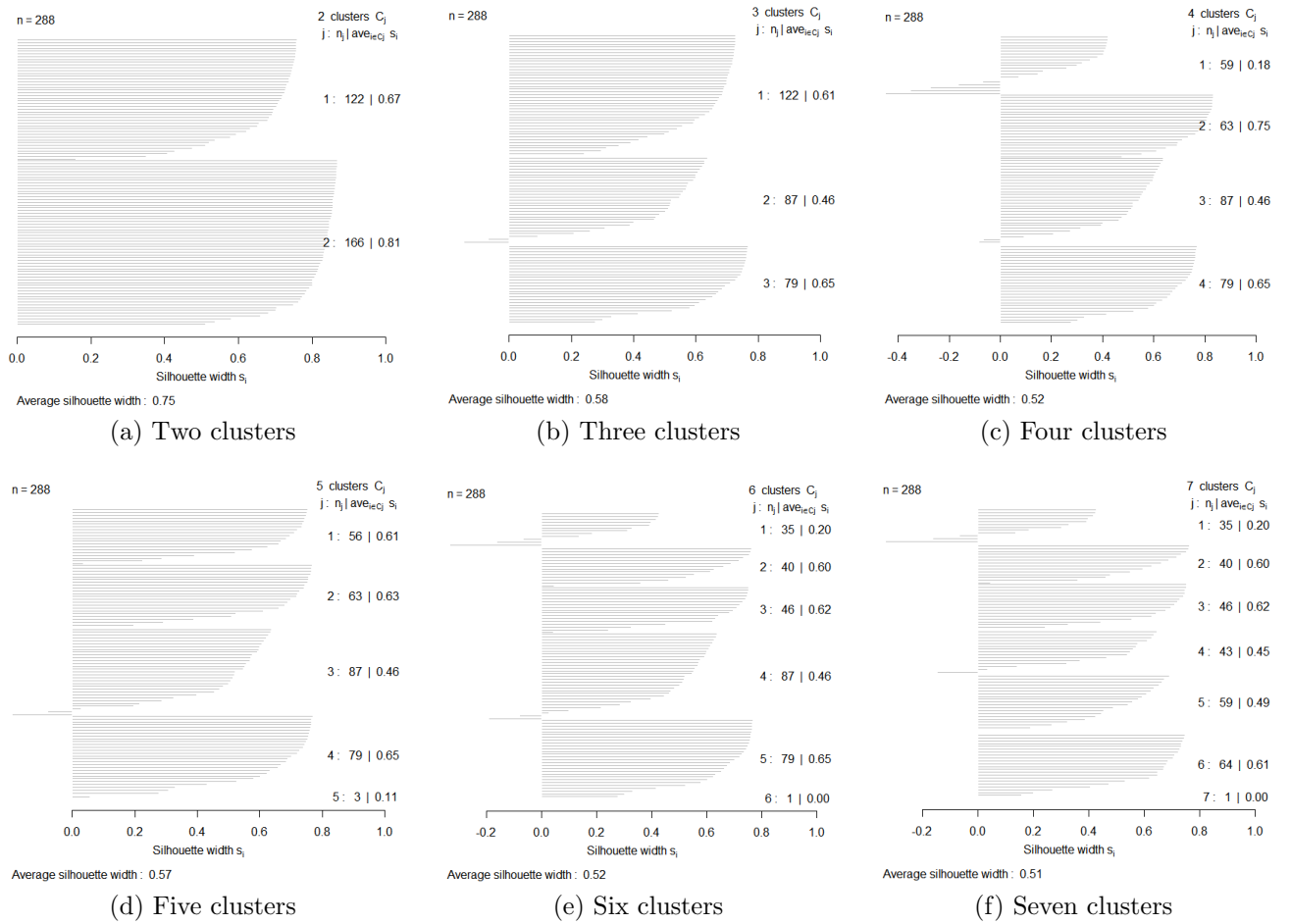


Figure A.1: Silhouette analysis by the number of resulting clusters using partition around medoids algorithm for the reference patient

B Relevant Code

B.1 Chapter 2: Patch definitions

Also available at: <https://github.com/vivifj03/TumorModel/blob/master/Code%20chapter2%20-%20patch%20definition.R>

```
#First, export CT scan data from Slicer to Matlab, and from Matlab  
#to R using Kingston Kang's code.
```

```
#This code is for data of one patient (c1)
```

```
# Packages -----  
library(plyr)  
library(tidyr)
```

```
# Read data -----
```

```
files<-list.files(pattern = "*.rda")  
dta<- new.env()  
lapply(files,load,dta)
```

```
#Have to change the names  
dta$c7_011.rda<-dta$c7_01_3.rda  
dta$c7_012.rda<-dta$c7_01_6.rda  
dta$c7_013.rda<-dta$c7_01_12.rda  
dta$c7_014.rda<-dta$c7_01_18.rda  
dta$c7_015.rda<-dta$c7_01_24.rda  
dta$c7_101.rda<-dta$c7_10_3.rda  
dta$c7_102.rda<-dta$c7_10_6.rda  
dta$c7_103.rda<-dta$c7_10_12.rda  
dta$c7_104.rda<-dta$c7_10_18.rda  
dta$c7_105.rda<-dta$c7_10_24.rda  
dta$c7_111.rda<-dta$c7_11_0.rda  
remove("c7_00_0.rda","c7_01_3.rda","c7_00_18.rda","c7_01_24.rda",  
       "c7_00_12.rda","c7_10_24.rda","c7_00_6.rda","c7_00_3.rda",  
       "c7_01_18.rda","c7_01_12.rda","c7_10_18.rda","c7_00_24.rda",  
       "c7_10_12.rda","c7_10_6.rda","c7_10_3.rda","c7_01_6.rda",  
       "c7_11_0.rda", envir = dta)
```

```
# There is overlap in the images. Hazy has dense data, so next lines  
# remove dense from hazy  
dta$c7_011.rda$dvf<-ifelse(dta$c7_011.rda$dvf==dta$c7_101.rda$dvf,  
                          0,dta$c7_011.rda$dvf)  
dta$c7_012.rda$dvf<-ifelse(dta$c7_012.rda$dvf==dta$c7_102.rda$dvf,  
                          0,dta$c7_012.rda$dvf)  
dta$c7_013.rda$dvf<-ifelse(dta$c7_013.rda$dvf==dta$c7_103.rda$dvf,
```

```

                                0,dta$c7_013.rda$dvf)
dta$c7_014.rda$dvf<-ifelse(dta$c7_014.rda$dvf==dta$c7_104.rda$dvf ,
                                0,dta$c7_014.rda$dvf)
dta$c7_015.rda$dvf<-ifelse(dta$c7_015.rda$dvf==dta$c7_105.rda$dvf ,
                                0,dta$c7_015.rda$dvf)

#I have to fix dose to be positive. because there are some negative
#values
#The min dose value in the original data (small size) is 0.0035
dta$c7_111.rda$dvf<-ifelse(dta$c7_111.rda$dvf<0,0,dta$c7_111.rda$dvf)

#Make sure all images have the same size
# c7_111 id the dose file

#If there are more files in dat than the images we need, then delete
remove(c7_000.rda, c7_000ptv.rda, c7_001.rda, c7_002.rda, c7_003.rda,
        c7_004.rda, c7_005.rda, envir = dta)

# Cubic Patch -----
in_avg_s<-function(x, s){
  #Calculates de average of the values in the cube
  d1<-dim(x$dvf)[1]
  d3<-dim(x$dvf)[3]
  l<-seq(from = 1, to = d1, by = s)
  w<-seq(from = 1, to = d3, by = s)
  l2<-seq(from = 1, to = length(l), by = 1)
  w2<-seq(from = 1, to = length(w), by = 1)
  res<-data.frame(matrix(nrow = length(l)*length(l)*length(w),
                          ncol = 13))
  colnames(res)<-c("patch_w", "patch_l", "patch_d", "patch_w2",
                  "patch_l2", "patch_d2", "patch", "avg", "min", "max",
                  "sd", "med", "riq")
  res[,1]<-rep(1,times=length(l)*length(w))
  res[,2]<-rep(sapply(1, function(x) rep(x,length(l))), length(w))
  res[,3]<-unlist(lapply(w, function(x) rep(x,length(l)^2)))
  res[,4]<-rep(l2,times=length(l)*length(w))
  res[,5]<-rep(sapply(l2, function(x) rep(x,length(l))), length(w))
  res[,6]<-unlist(lapply(w2, function(x) rep(x,length(l)^2)))
  res[,7]<-paste(res[,4],res[,5], res[,6], sep="_" )
  for(i in 1:nrow(res)){
    a<-as.numeric(res[i,1])
    b<-as.numeric(min(a+s-1,d1))
    c<-as.numeric(res[i,2])
    d<-as.numeric(min(c+s-1,d1))
    e<-as.numeric(res[i,3])
    f<-as.numeric(min(e+s-1,d3))
    res[i,8]<-mean(x$dvf[a:b,c:d,e:f], na.rm=TRUE)
  }
}

```



```

    res[i,9]<-min(x$dvf[a:b,c:d,e:f], na.rm = TRUE)
    res[i,10]<-max(x$dvf[a:b,c:d,e:f], na.rm = TRUE)
    res[i,11]<-sd(x$dvf[a:b,c:d,e:f], na.rm=TRUE)
    res[i,12]<-median(x$dvf[a:b,c:d,e:f], na.rm=TRUE)
    res[i,13]<-quantile(x$dvf[a:b,c:d,e:f], 0.75, na.rm=TRUE)-
      quantile(x$dvf[a:b,c:d,e:f], 0.25, na.rm=TRUE)
  }
  return(res)
}

numb<-function(x,s){
  d1<-dim(x$dvf)[1]
  d3<-dim(x$dvf)[3]

  matl<-array(NA,c(d1,d1,d3))
  matw<-array(NA,c(d1,d1,d3))
  matd<-array(NA,c(d1,d1,d3))
  for(l in 1:d1){
    matl[l, ,]<-ceiling(l/s)
  }
  for(w in 1:d1){
    matw[,w ,]<-ceiling(w/s)
  }
  for(d in 1:d3){
    matd[, ,d]<-ceiling(d/s)
  }
  mat<-data.frame(matrix(NA, nrow=d1*d1*d3, ncol=4))
  colnames(mat)<-c("patch_l", "patch_w", "patch_d", "patch")
  mat[,1]<-as.vector(matl)
  mat[,2]<-as.vector(matw)
  mat[,3]<-as.vector(matd)
  mat[,4]<-paste(mat[,1], mat[,2], mat[,3], sep = "_", collapse = NULL)
  return(mat)
}

in_avg_s2<-function(x, s){
  #this fuction is summarizing dose in the patch
  #(max, min ...)
  d1<-dim(x$dvf)[1]
  d3<-dim(x$dvf)[3]

  subx<-data.frame(matrix(NA, nrow=d1*d1*d3, ncol=5))
  colnames(subx)<-c("val", "patch_l", "patch_w", "patch_d", "patch")
  subx[,1]<-as.vector(x$dvf[1:d1,1:d1,1:d3])
  subx[,2:5]<-mat[,1:4]
  n_t<-aggregate(list(n=subx$val), by=list(patch=subx$patch),
    function(x) length(which(!is.na(x))))
}

```

```

mean_t<-aggregate(list(avg=subx$val), by=list(patch=subx$patch),
  FUN=mean, na.rm=TRUE)
min_t<-aggregate(list(min=subx$val), by=list(patch=subx$patch),
  FUN=min, na.rm=TRUE)
max_t<-aggregate(list(max=subx$val), by=list(patch=subx$patch),
  FUN=max, na.rm=TRUE)
sd_t<-aggregate(list(sd=subx$val), by=list(patch=subx$patch),
  FUN=sd, na.rm=TRUE)
median_t<-aggregate(list(med=subx$val), by=list(patch=subx$patch),
  FUN=median, na.rm=TRUE)
iqr_t<-aggregate(list(riq=subx$val), by=list(patch=subx$patch),
  function(x) quantile(x,probs=0.75, na.rm=TRUE)-
  quantile(x,probs=0.25, na.rm=TRUE))

res<-Reduce(function(x, y) merge(x, y, by="patch"),
  list(n_t, mean_t, min_t, max_t, sd_t, median_t, iqr_t))
return(res)
}

P_num<-function(x, s){
  #Build the cubic patch
  d1<-dim(x$dvf)[1]
  d3<-dim(x$dvf)[3]
  dosel_111<-data.frame(matrix(NA, nrow=d1*d1*d3, ncol=8))
  colnames(dosel_111)<-c("dose", "patch_l", "patch_w", "patch_d",
    "patch", "row", "col", "layer")
  dosel_111[,1]<-as.vector(x$dvf[1:d1,1:d1,1:d3])

  dosel_111[,2:5]<-mat[,1:4]
  dosel_111[,6]<-rep(1:d1, by=1, time=d1*d3)
  dosel_111[,7]<-rep(1:d1, by=1, each=d1, time=d3)
  dosel_111[,8]<-rep(1:d3, by=1, each=d1*d1)
  return(dosel_111)
}

k<-8 #This is patch size, if want different size then change
mat<-numb(dta$c7_111.rda,s=k)
dosel<-P_num(dta$c7_111.rda,s=k)
Resumen<-eapply(dta, function(x)in_avg_s2(x,s=k), USE.NAMES = TRUE)

v <- in_avg_s2(dta$c7_011.rda,s=k)

Resumen_l<-ldply(Resumen)
Resumen_l$Id<-sub(".*_", "", Resumen_l$.id)
Resumen_l$file<-substr(sub(".*_", "", Resumen_l$.id) ,1,2)
Resumen_l$date<-substr(sub(".*_", "", Resumen_l$.id) ,3,3)
Resumen_l<-Resumen_l[,c(".id", "Id", "file", "date", "patch", "n",

```

```

      "avg", "min", "max", "sd", "med", "riq"]

Resumen_w<-reshape(Resumen_l[,-1], idvar = c("Id", "date", "patch"),
  timevar = "file", direction = "wide")
Resumen_w<-Resumen_w[,c("Id", "date", "patch", "n.01", "avg.01",
  "avg.10", "avg.11", "min.11", "max.11", "sd.11",
  "med.11", "riq.11")]
colnames(Resumen_w)<-c("Id", "date", "patch", "n", "Hazy", "Dense",
  "Dose_avg", "Dose_min", "Dose_max", "Dose_sd",
  "Dose_med", "Dose_riq")

Resumen_w$Com<-(1-(Resumen_w$Hazy+Resumen_w$Dense))

Resumen_w2<-reshape(Resumen_w, idvar = c("Id","patch"), timevar =
  "date",direction = "wide")
Resumen_w2<-Resumen_w2[,c("Id", "patch", "n.1", "Dose_avg.1",
  "Dose_min.1", "Dose_max.1", "Dose_sd.1",
  "Dose_med.1", "Dose_riq.1", "Dense.1", "Hazy.1",
  "Com.1", "Dense.2", "Hazy.2", "Com.2",
  "Dense.3", "Hazy.3", "Com.3", "Dense.4",
  "Hazy.4", "Com.4", "Dense.5", "Hazy.5",
  "Com.5")]

Resumen_w$date2<-ifelse(Resumen_w$date==1,3,ifelse(Resumen_w$date==2,6,
  ifelse(Resumen_w$date==3,12,
  ifelse(Resumen_w$date==4,18,24))))
#ZeroChange are patches that were always 100% normal
list_ZeroChange <- Resumen_w2[which(Resumen_w2$Com.1==1 &
  Resumen_w2$Com.2==1 &
  Resumen_w2$Com.3==1 &
  Resumen_w2$Com.4==1 &
  Resumen_w2$Com.5==1), 2]

length(list_ZeroChange)
list_maxDoseZero<-Resumen_w[which(Resumen_w$Dose_max<=0),3]

Resumen_w$chageID<-ifelse(Resumen_w$patch %in% list_ZeroChange,0, 1)
Resumen_w$someDose<-ifelse(Resumen_w$patch %in% list_maxDoseZero,0, 1)
dosel$chageID<-ifelse(dosel$patch %in% list_ZeroChange,0,1)
dosel$someDose<-ifelse(dosel$patch %in% list_maxDoseZero,0, 1)
dosel<-dosel[!is.na(dosel$dose),]

rm(dta,files, mat, matl, matd, matw)
rm(end, inicio, in_avg_s, in_avg_s2, numb, P_num)

Resumen_w$keep<-ifelse(Resumen_w$chageID==1,1,0)
Resumen_w$keep<-ifelse(Resumen_w$someDose==1,1,Resumen_w$keep)
dataset<-Resumen_w[which(Resumen_w$keep==1),]
dataset$ZeroHD<-ifelse(dataset$Com==1,1,0) #Both hazy and dense are zero

```

```

datadose<-dataset [dataset$date2==3,c("patch","Dose_avg","Dose_min",
    "Dose_max","Dose_sd","Dose_med","Dose_riq")]
dataset<-dataset [,c("Id","patch","date","date2","Hazy","Dense","Com",
    "chageID", "someDose","keep","ZeroHD")]
dataset<-merge(dataset,datadose,by=c("patch"), all.x=T)
dataset_w<-Resumen_w2[Resumen_w2$patch %in% dataset$patch,]

save.image(file = paste("cubic_patch", k, ".RData", sep = ""))

# Spherical patch -----
max.val<-which(dta$c7_111.rda$dvf == max(dta$c7_111.rda$dvf, na.rm=T),
    arr.ind = TRUE)
numb.centered<-function(x,s,v){
  d1<-dim(x$dvf)[1]
  d3<-dim(x$dvf)[3]

  matl<-array(NA,c(d1,d1,d3))
  matw<-array(NA,c(d1,d1,d3))
  matd<-array(NA,c(d1,d1,d3))
  for(l in 1:d1){
    matl[l, ,]<-ceiling(l/s)
  }
  for(w in 1:d1){
    matw[,w ,]<-ceiling(w/s)
  }
  for(d in 1:d3){
    matd[, ,d]<-ceiling(d/s)
  }
  mat<-data.frame(matrix(NA, nrow=d1*d1*d3, ncol=7))
  colnames(mat)<-c("patch_l","patch_w","patch_d","dif_l","dif_w",
    "dif_d","patch")
  mat[,1]<-as.vector(matl)
  mat[,2]<-as.vector(matw)
  mat[,3]<-as.vector(matd)
  mat[,4]<-mat[,1]-v[1]
  mat[,5]<-mat[,2]-v[2]
  mat[,6]<-mat[,3]-v[3]
  mat[,7]<-floor(sqrt(mat[,4]^2 + mat[,5]^2 + mat[,6]^2))
  mat[,7]<-ifelse(mat[,7]==0,1,mat[,7])
  return(mat)
}

D_sum<-function(x, s,name="value"){
  d1<-dim(x$dvf)[1]
  d3<-dim(x$dvf)[3]
  unwrap<-data.frame(matrix(NA, nrow=d1*d1*d3, ncol=8))
  colnames(unwrap)<-c(name, "patch_l", "patch_w", "patch_d",

```

```

        "patch", "row", "col", "layer")
unwrap[,1] <- as.vector(x$dvf[1:d1,1:d1,1:d3])

unwrap[,2:5] <- mat[,c(1:3,7)]
unwrap[,6] <- rep(1:d1, by=1, time=d1*d3)
unwrap[,7] <- rep(1:d1, by=1, each=d1, time=d3)
unwrap[,8] <- rep(1:d3, by=1, each=d1*d1)

Resumen <- aggregate(list(n=unwrap[name]), by=list(patch = unwrap$patch),
                     FUN=length)
colnames(Resumen) <- c("patch", "n")

Resumen <- merge(Resumen, aggregate(list(Dose_min=unwrap[name]),
                                   by=list(patch = unwrap$patch),
                                   FUN=min), by="patch")
colnames(Resumen) <- c("patch", "n", "Dose_min")
Resumen <- merge(Resumen, aggregate(list(Dose_max=unwrap[name]),
                                   by=list(patch = unwrap$patch),
                                   FUN=max), by="patch")
colnames(Resumen) <- c("patch", "n", "Dose_min", "Dose_max")
Resumen <- merge(Resumen, aggregate(list(Dose_med=unwrap[name]),
                                   by=list(patch = unwrap$patch),
                                   FUN=median), by="patch")
colnames(Resumen) <- c("patch", "n", "Dose_min", "Dose_max", "Dose_med")
Resumen <- merge(Resumen, aggregate(list(Dose_avg=unwrap[name]),
                                   by=list(patch = unwrap$patch),
                                   FUN=mean), by="patch")
colnames(Resumen) <- c("patch", "n", "Dose_min", "Dose_max", "Dose_med",
                      "Dose_avg")
Resumen <- merge(Resumen, aggregate(list(Dose_sd=unwrap[name]),
                                   by=list(patch = unwrap$patch),
                                   FUN=sd), by="patch")
colnames(Resumen) <- c("patch", "n", "Dose_min", "Dose_max", "Dose_med",
                      "Dose_avg", "Dose_sd")

dosel <- list(Resumen=Resumen, dose_raw=unwrap)
return(dosel)
}

DH_sum <- function(x){
d1 <- dim(x$dvf)[1]
d3 <- dim(x$dvf)[3]
unwrap <- data.frame(matrix(NA, nrow=d1*d1*d3, ncol=5))
colnames(unwrap) <- c("value", "patch_l", "patch_w", "patch_d", "patch")
unwrap[,1] <- as.vector(x$dvf[1:d1,1:d1,1:d3])

unwrap[,2:5] <- mat[,c(1:3,7)]

```

```

Resumen<-aggregate(list(count=unwrap$value), by=list(patch =
                    unwrap$patch), FUN=sum)
return(Resumen)
}

max.d<-max.val[1,]
mat<-numb.centered(dta$c7_111.rda,s=1,v=max.d)
dose1.f<-D_sum(dta$c7_111.rda,s=1,name="dose")
dose1<-dose1.f$Resumen

Resumen<-eapply(dta, function(x) DH_sum(x), USE.NAMES = TRUE)

Resumen_1<-ldply(Resumen)
Resumen_1<-separate(Resumen_1, .id, c("Id","file"), "_")
Resumen_1$date<-as.numeric(as.character(substr(Resumen_1$file, 3, 3)))
Resumen_1$file<-as.character(substr(Resumen_1$file, 1, 2))
Resumen_1<-Resumen_1[,c(1,2,5,3,4)]
Resumen_1<-Resumen_1[Resumen_1$date!=0,]
Resumen_1<-Resumen_1[order(Resumen_1$date),]

Resumen_w<-reshape(Resumen_1, idvar = c("Id","date", "patch"),
                    timevar = "file", direction = "wide")
Resumen_w<-Resumen_w[,c("Id","patch","date","count.01","count.10")]
names(Resumen_w)<- c("Id", "patch", "date", "Hazy.n", "Dense.n")
Resumen_w<-merge(Resumen_w,dose1,by="patch",all.x =T)
Resumen_w$Hazy<-Resumen_w$Hazy.n/Resumen_w$n
Resumen_w$Dense<-Resumen_w$Dense.n/Resumen_w$n
Resumen_w$Com<-(1-(Resumen_w$Hazy+Resumen_w$Dense))

Resumen_w<-Resumen_w[order(Resumen_w$patch, Resumen_w$date),]
Resumen_w<-Resumen_w[Resumen_w$date!=0,]

Resumen_w2<-reshape(Resumen_w, idvar = c("Id","patch"),
                    timevar = "date", direction = "wide")

list_ZeroChange <- Resumen_w2[ which(Resumen_w2$Com.3==1 &
                                   Resumen_w2$Com.6==1 &
                                   Resumen_w2$Com.12==1 &
                                   Resumen_w2$Com.18==1 &
                                   Resumen_w2$Com.24==1), 1]

length(list_ZeroChange)

list_maxDoseZero<-Resumen_w[which(Resumen_w$Dose_max<=0),1]

Resumen_w$chageID<-ifelse(Resumen_w$patch %in% list_ZeroChange,0, 1)
Resumen_w$someDose<-ifelse(Resumen_w$patch %in% list_maxDoseZero,0, 1)

```

```

dosel<-dosel.f$dose_raw
save.image(file = paste("spheric_patch.RData", sep = ""))

# Isodose patch -----
m<-ceiling(max(dta$c7_111.rda$dvf,na.rm = T))

#Categorize dose for isodose lines
s1<-c(-Inf,seq(0,m,by=1)) #0,
s2<-c(-Inf,seq(0,m,by=0.5)) #0, ..
s3<-c(-Inf,seq(0,m,by=0.2)) #0, .
s4<-c(-Inf,seq(0,m,by=0.1)) #0, .
s5<-c(-Inf,seq(0,m,by=0.05)) #0, .

d1<-dim(dta$c7_111.rda$dvf)[1]
d3<-dim(dta$c7_111.rda$dvf)[3]
dosel<-data.frame(matrix(NA, nrow=d1*d1*d3, ncol=4))
colnames(dosel)<-c("dose", "row", "col", "layer")
dosel[,1]<-as.vector(dta$c7_111.rda$dvf[1:d1,1:d1,1:d3])
dosel[,2]<-rep(1:d1, by=1, time=d1*d3)
dosel[,3]<-rep(1:d1, by=1, each=d1, time=d3)
dosel[,4]<-rep(1:d3, by=1, each=d1*d1)
dosel$dosecat_s1<-cut(dosel$dose, breaks=s1)
dosel$dosecat_s2<-cut(dosel$dose, breaks=s2)
dosel$dosecat_s3<-cut(dosel$dose, breaks=s3)
dosel$dosecat_s4<-cut(dosel$dose, breaks=s4)
dosel$dosecat_s5<-cut(dosel$dose, breaks=s5)

length(unique(dosel$dosecat_s1))
length(unique(dosel$dosecat_s2))
length(unique(dosel$dosecat_s3))
length(unique(dosel$dosecat_s4))
length(unique(dosel$dosecat_s5))

max.val<-which(dta$c7_111.rda$dvf == max(dta$c7_111.rda$dvf, na.rm=T),
arr.ind = TRUE)

for (i in 1:5) {
  nam <- paste("Resumen_s", i, sep = "")
  v<-aggregate(list(n=dosel$dose), by=list(patch = dosel[,4+i]),
FUN=length)
  v<-merge(v, aggregate(list(Dose_min=dosel$dose), by=list(patch =
dosel[,4+i]), FUN=min), by="patch")
  v<-merge(v, aggregate(list(Dose_max=dosel$dose), by=list(patch =
dosel[,4+i]), FUN=max), by="patch")
  v<-merge(v, aggregate(list(Dose_med=dosel$dose), by=list(patch =
dosel[,4+i]), FUN=median), by="patch")
  v<-merge(v, aggregate(list(Dose_avg=dosel$dose), by=list(patch =

```

```

        dose1[,4+i]), FUN=mean), by="patch")
v<-merge(v, aggregate(list(Dose_sd=dose1$dose), by=list(patch =
        dose1[,4+i]), FUN=sd), by="patch")
assign(nam, v)
}
rm(v, nam)
rm(s1, s2, s3, s4, s5)

DH_sum<-function(x, dose1=dose1, cat=1){
  d1<-dim(x$dvf)[1]
  d3<-dim(x$dvf)[3]
  unwrap<-data.frame(matrix(NA, nrow=d1*d1*d3, ncol=5))
  colnames(unwrap)<-c("value", "patch_l", "patch_w", "patch_d", "patch")
  unwrap[,1]<-as.vector(x$dvf[1:d1,1:d1,1:d3])
  unwrap[,2:5]<-dose1[,c(2:4,4+cat)]
  Resumen<-aggregate(list(count=unwrap$value), by=list(patch =
        unwrap$patch), FUN=sum)

  return(Resumen)
}

Resumen_1<-eapply(dta, function(x) DH_sum(x, dose1=dose1, cat=1),
  USE.NAMES = TRUE)
Resumen_2<-eapply(dta, function(x) DH_sum(x, dose1=dose1, cat=2),
  USE.NAMES = TRUE)
Resumen_3<-eapply(dta, function(x) DH_sum(x, dose1=dose1, cat=3),
  USE.NAMES = TRUE)
Resumen_4<-eapply(dta, function(x) DH_sum(x, dose1=dose1, cat=4),
  USE.NAMES = TRUE)
Resumen_5<-eapply(dta, function(x) DH_sum(x, dose1=dose1, cat=5),
  USE.NAMES = TRUE)
# Code for isodose patch of 1Gy interval.
# Repeat this code for other sized changing Resumen_1
Resumen_1<-ldply(Resumen_1)

Resumen_1<-separate(Resumen_1, .id, c("Id", "file"), "_")
Resumen_1$date<-as.numeric(as.character(substr(Resumen_1$file, 3, 3)))
Resumen_1$file<-as.character(substr(Resumen_1$file, 1, 2))
Resumen_1<-Resumen_1[,c(1,2,5,3,4)]
Resumen_1<-Resumen_1[Resumen_1$date!=0,]
Resumen_1<-Resumen_1[order(Resumen_1$date),]

Resumen_w<-reshape(Resumen_1, idvar = c("Id", "date", "patch"),
  timevar = "file", direction = "wide")
Resumen_w<-Resumen_w[,c("Id", "patch", "date", "count.01", "count.10")]
names(Resumen_w)<- c("Id", "patch", "date", "Hazy.n", "Dense.n")
Resumen_w<-Resumen_w[order(Resumen_w$date, Resumen_w$patch),]

```



```

Resumen_w<-merge(Resumen_w,Resumen_s1,by="patch",all.x =T)
Resumen_w<-Resumen_w[order(Resumen_w$date,Resumen_w$patch),]

Resumen_w$Hazy<-Resumen_w$Hazy.n/Resumen_w$n
Resumen_w$Dense<-Resumen_w$Dense.n/Resumen_w$n
Resumen_w$Com<-(1-(Resumen_w$Hazy+Resumen_w$Dense))

Resumen_w<-Resumen_w[order(Resumen_w$patch,Resumen_w$date),]
Resumen_w<-Resumen_w[Resumen_w$date!=0,]

Resumen_w2<-reshape(Resumen_w, idvar = c("Id","patch"),
                    timevar = "date", direction = "wide")

list_ZeroChange <- Resumen_w2[ which(Resumen_w2$Com.3==1 &
                                   Resumen_w2$Com.6==1 & Resumen_w2$Com.12==1 &
                                   Resumen_w2$Com.18==1 & Resumen_w2$Com.24==1), 1]
length(list_ZeroChange)
list_maxDoseZero<-Resumen_w[which(Resumen_w$Dose_max<=0),1]

Resumen_w$chageID<-ifelse(Resumen_w$patch %in% list_ZeroChange,0, 1)
Resumen_w$someDose<-ifelse(Resumen_w$patch %in% list_maxDoseZero,0, 1)
# After creating Resumen_1_w to Resumen_5_w then save data
save.image(file = paste("isodose_patch", k, ".RData", sep = ""))

# Compositional data -----
# After creating the data sets, some adjustments to the composition
# were done (e.g. Fry zero transformation)

Resumen_w$keep<-ifelse(Resumen_w$chageID==1,1,0)
Resumen_w$keep<-ifelse(Resumen_w$someDose==1,1,Resumen_w$keep)
Resumen_w$ZeroHD<-ifelse(Resumen_w$Com==1,1,0) #Both hazy&dense are zero
dataset<-Resumen_w[which(Resumen_w$keep==1),]
dataset_w<-Resumen_w2[Resumen_w2$patch %in% dataset$patch,]

#Transform data
d<-0.003
N<-3
dataset$M<-apply(dataset[,c("Hazy","Dense","Com")],1,function(x)
                 sum(x==0))
dataset$M<-ifelse(dataset$ZeroHD==1, NA,dataset$M)
dataset$ta<-ifelse(dataset$ZeroHD==1,
                  NA,d*(dataset$M+1)*(N-dataset$M)/(N*N))
dataset$ts<-dataset$ta*dataset$M
dataset$Dense.03<-ifelse(dataset$Dense==0,dataset$ta,ifelse(dataset$M==0,
                  dataset$Dense,dataset$Dense-(dataset$Dense*dataset$ts)))
dataset$Hazy.03<-ifelse(dataset$Hazy==0,dataset$ta,ifelse(dataset$M==0,
                  dataset$Hazy,dataset$Hazy-(dataset$Hazy*dataset$ts)))

```

```

dataset$Com.03<-ifelse(dataset$Com==0,dataset$ta,ifelse(dataset$M==0,
              dataset$Com,dataset$Com-(dataset$Com*dataset$ts)))
dataset$Hazy.t<-dataset$Hazy.03/dataset$Com.03
dataset$Dense.t<-dataset$Dense.03/dataset$Com.03
dataset$Hazy.lt<-log(dataset$Hazy.t)
dataset$Dense.lt<-log(dataset$Dense.t)

dataset$Hazy.t0<-ifelse(is.na(dataset$Hazy.t)==T,0,dataset$Hazy.t)
dataset$Dense.t0<-ifelse(is.na(dataset$Dense.t)==T,0,dataset$Dense.t)
dataset$Hazy.Inon0<-ifelse(dataset$Hazy.t0==0,0,1)
dataset$Dense.Inon0<-ifelse(dataset$Dense.t0==0,0,1)
dataset$Hazy.I0<-1-dataset$Hazy.Inon0
dataset$Dense.I0<-1-dataset$Dense.Inon0

dataset$Hazy.lt0<-ifelse(is.na(dataset$Hazy.lt)==T,0,dataset$Hazy.lt)
dataset$Dense.lt0<-ifelse(is.na(dataset$Dense.lt)==T,0,dataset$Dense.lt)

dataset$date2<-ifelse(dataset$date==1,3,
              ifelse(dataset$date==2,6,
                    ifelse(dataset$date==3,12,
                          ifelse(dataset$date==4,18,24))))

dataset$years<-dataset$date2/12
dataset$years2<-as.numeric(as.character((dataset$years)))**2
dataset$years3<-as.numeric(as.character((dataset$years)))**3

dataset<-dataset[,c("Id","patch","n","date","date2","years",
              "years2","years3","Dose_min",
              "Dose_max","Dose_med","Dose_avg","Dose_sd",
              "Hazy","Dense","Com","chageID","someDose","keep",
              "ZeroHD","M","ta","ts","Dense.03","Hazy.03",
              "Com.03","Hazy.t","Dense.t","Hazy.lt","Dense.lt",
              "Hazy.t0","Dense.t0","Hazy.Inon0","Dense.Inon0",
              "Hazy.I0","Dense.I0","Hazy.lt0","Dense.lt0")]
rm(list=setdiff(ls(), c("dataset", "dose1", "list_maxDoseZero",
              "list_ZeroChange"))))

# Calinski-Harabasz (CH) index
library(nlme)
library(reghelper)
ctrl <- lmeControl(maxIter = 100, msMaxIter = 100, singular.ok=TRUE,
              returnObject=TRUE, opt='optim')

#For cubic
mod <- lme( dose ~ 1, data=dose1, random = ~ 1|patch, control=ctrl)
#For spherical
mod <- lme( dose ~ 1, data=dose1, random = ~ 1|patch, control=ctrl)
#For isodose
mod <- lme( dose ~ 1, data=dose1, random = ~ 1|dosecat_s1, control=ctrl)

```

```

icc<-ICC(mod)
sse<-mod$sigma^2
ssm<-diag(sqrt(getVarCov(mod)))^2
icc.h<-ssm/(ssm+sse)
ch<-(ssm/(mod$dims$ngrps[1]-1))/(sse/(mod$dims$N - mod$dims$ngrps[1]))

# Figures -----
library(ggplot2)
library(reshape2)
GeomSplitViolin <- ggproto("GeomSplitViolin", GeomViolin,
  draw_group = function(self, data, ...,
  draw_quantiles = NULL) {
    data <- transform(data, xminv = x -
      violinwidth * (x - xmin), xmaxv = x +
      violinwidth * (xmax - x))
    grp <- data[1, "group"]
    newdata <- plyr::arrange(transform(data, x =
      if (grp %% 2 == 1) xminv else xmaxv), if
      (grp %% 2 == 1) y else -y)
    newdata <- rbind(newdata[1, ], newdata,
      newdata[nrow(newdata), ], newdata[1, ])
    newdata[c(1, nrow(newdata) - 1,
      nrow(newdata)), "x"] <- round(newdata[1,
      "x"])

    if (length(draw_quantiles) > 0 &
      !scales::zero_range(range(data$y))) {
      stopifnot(all(draw_quantiles >= 0),
        all(draw_quantiles <=1))
      quantiles <-
      ggplot2::create_quantile_segment_frame(data,
        draw_quantiles)
      aesthetics <- data[rep(1, nrow(quantiles)),
        setdiff(names(data), c("x", "y")), drop =
        FALSE]
      aesthetics$alpha <- rep(1, nrow(quantiles))
      both <- cbind(quantiles, aesthetics)
      quantile_grob <- GeomPath$draw_panel(both,
        ...)
      ggplot2:::ggname("geom_split_violin",
        grid::grobTree(GeomPolygon$draw_panel(
          newdata, ...), quantile_grob))
    }
  }
  else {
    ggplot2:::ggname("geom_split_violin",
      GeomPolygon$draw_panel(newdata, ...))
  }
}

```

```

    }
  })

geom_split_violin <- function(mapping = NULL, data = NULL, stat =
  "ydensity", position = "identity", ...,
  draw_quantiles = NULL, trim = TRUE, scale =
  "area", na.rm = FALSE,
  show.legend = NA, inherit.aes = TRUE) {
  layer(data = data, mapping = mapping, stat = stat, geom =
  GeomSplitViolin,
    position = position, show.legend = show.legend, inherit.aes =
    inherit.aes,
    params = list(trim = trim, scale = scale, draw_quantiles =
    draw_quantiles, na.rm = na.rm, ...))
}

# Figure 2.13 (a) [2.17, 2.23]
c<-round(prop.table(table(dataset$ZeroHD, dataset$date2), 2)*100, 2)
c<- as.data.frame(cbind(Months=unique(dataset$date2), P=c[1,]))
len<-nrow(dataset)/length(unique(dataset$date2))
ggplot(data=c, aes(x=factor(Months), y=P)) +
  geom_bar(stat="identity", fill=alpha(c("#926B8D"), 0.7)) +
  xlab("Months after RT\n") +
  ylab(paste("% of patches with RILD (N=", len, ")", sep = "")) +
  scale_y_continuous(sec.axis = sec_axis(~ . * len/100,
    name="Number of patches with RILD"), limits=c(0, 1))

# Figure 2.13 (b) [2.17, 2.23]
boxplot<-as.data.frame(dataset[dataset$date==1,])
ggplot(data = boxplot, aes(x=factor(chageID), y=Dose_med)) +
  geom_boxplot(aes(color=factor(chageID), fill = factor(chageID)),
    alpha=0.7) +
  xlab("Patch with RILD Presence") + ylab("Median Dose x Patch") +
  geom_point(pch = 19, position = position_jitterdodge(),
    alpha=0.3, size=1, aes(colour = factor(chageID)))+
  scale_fill_manual(values = c("#B7B6BA", "#926B8D")) +
  scale_color_manual(values = c("#B7B6BA", "#926B8D")) +
  scale_x_discrete(labels=c("0"= "No", "1"="Yes")) +
  scale_y_continuous(breaks=seq(0, max(boxplot$Dose_med), by=10))

# Figure 2.13 (c) [2.17, 2.23]
my_dat2<-dataset[,c("date2", "Dense.lt", "Hazy.lt")]
my_dat2 <- melt(my_dat2, id.vars=c("date2"))

ggplot(data = my_dat2, aes(x=as.factor(date2), y=value,
  fill =as.factor(variable),
  color=as.factor(variable))) +

```

```

geom_split_violin(trim=FALSE) +
xlab("\nMonths after RT") +
ylab("alr transformation\nln(Y/N)\n") +
scale_color_manual(name = "Type of RILD",
                    values=c("#926B8D", "#7aadb1"), labels=c("Dense",
                    "Hazy")) +
scale_fill_manual(name = "Type of RILD",
                  values=alpha(c("#926B8D", "#7aadb1"),0.7),
                  labels=c("Dense", "Hazy"))

#Figure 2.14 [2.18, 2.24]
library(compositions)
library(ggtern)
library(png)
a<-c(3,6,12,18,21)
k<-1 #Change k for different time points
ggtern(data=dataset[dataset$date2==a[[k]],c("Hazy","Dense","Com")],
        aes(x=Dense, y=Hazy, z=Com)) + geom_point() +
  theme_showarrows() + ggtitle(paste(a[[k]], "Months", sep = "")) +
  theme(plot.title = element_text(face="bold", hjust = 0.5, vjust=8),
        plot.margin = unit(c(0,0,0,0), "cm"))

#Figure 2.15 [2.19, 2.25]
ggplot(dataset, aes(x=Dose_med, y=Dense.lt0)) +
  geom_point(alpha=0.7, col="#926b8d") +
  xlab("Median dose") + ylab("alr(Dense)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
  element_blank(), panel.grid.minor = element_blank(),
  panel.background = element_blank(), axis.line = element_line(colour =
  "black"), text = element_text(size=16),
  legend.position="bottom", legend.title = element_blank())

ggplot(dataset, aes(x=Dose_med, y=Hazy.lt0)) +
  geom_point(alpha=0.7, col="#926b8d") +
  xlab("Median dose") + ylab("alr(Hazy)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
  element_blank(), panel.grid.minor = element_blank(),
  panel.background = element_blank(), axis.line = element_line(colour =
  "black"), text = element_text(size=16),
  legend.position="bottom", legend.title = element_blank())

```

B.2 Chapter 3: Two-part mixed-effects mixture model for zero-inflated longitudinal compositional data

Also available at: <https://github.com/vivifj03/TumorModel/blob/master/Code%20chapter3%20-%20tpmemmzl.R>

```
p.inf.l<-function(x){
  #Some propi<-exp(eta)/(1+exp(eta)) can be inf, too small or large,
  #then we are truncating the values
  x<-ifelse(is.finite(x),x,0.999999)
  x<-ifelse(x<0.000001,0.000001,x)
  x<-ifelse(x>0.999999,0.999999,x)
}

p.inf<-function(x){
  #Some propi<-exp(eta)/(1+exp(eta)) can be inf, too small or large,
  #then we are truncating the values
  x<-ifelse(is.finite(x),x,0.99)
  x<-ifelse(x<1e-6,1e-6,x)
  x<-ifelse(x>0.98,0.98,x)
}

laplace2<-function (logpost, mode, ...)
{
  options(warn = -1)
  fit = optim(mode, logpost, gr = NULL, ..., hessian = FALSE,
             control = list(fnscale = -1))
  options(warn = 0)
  mode = fit$par
  stuff = list(mode = mode, converge = fit$convergence == 0)
  return(stuff)
}

lap.for.c<-function(ram,other){ #find mode for random effects
  ram<-as.matrix(ram)
  U<-other$U
  X<-as.matrix(other$X)
  Z<-as.matrix(other$Z)
  Beta<-as.matrix(other$Beta)
  Psi_cc.i<-as.matrix(other$Psi_cc.i)
  D<-as.matrix(other$D)
  g.To<-as.matrix(other$g.To)
  g.o<-t(g.To)
  eta<- (X %*% Beta) + (Z %*% ram)
  propi<-exp(eta)/(1+exp(eta))
  propi<-apply(propi, 1, p.inf.l)
  eta<- log(propi/(1-propi))
}
```

```

lu<-sum((U * eta) + log(1-propi))
mat<-Psi_cc.i+D
mat.i<-solve(Psi_cc.i+D)
qua<-t(ram-(mat.i %*% g.o)) %*% mat %*% (ram-(mat.i %*% g.o))
fun<-lu-((1/2)*qua)
return(fun)
}

lklhd.em<-function(id,datafile,initial,X.var=X.var,Z.var=Z.var,
                  X_a.var=X_a.var,Z_a.var=Z_a.var,U.var=U.var,
                  V1=V1,V2=V2){
  #Estimation of mixed model for random effects (m)
  Beta=initial$Beta
  Gamma=initial$Gamma
  Gamma<-matrix(Gamma,ncol=1)
  Psi_cc=initial$Psi_cc
  Psi_dd=initial$Psi_dd
  corr<-initial$corr
  Var.V1<-initial$Var.V1
  Var.V2<-initial$Var.V2
  Psi_dc=initial$Psi_dc
  m=initial$m
  mu_1=initial$mu_1

  p<-length(Beta)
  q<-ifelse(length(Psi_cc)>1,dim(Psi_cc)[1],length(Psi_cc))
  p_a<-length(Gamma)
  q_a<-ifelse(length(Psi_dd)>1,dim(Psi_dd)[1],length(Psi_dd))
  Sigma<-matrix(c(Var.V1,rep(corr*sqrt(Var.V1)*sqrt(Var.V2),2),Var.V2),
                nrow = 2,ncol = 2)

  #Data
  base<-datafile[datafile$patch==id,]
  base$int <- rep(1,nrow(base))
  U<-base[,U.var]
  X.var<-c("int",X.var)
  X<-base[,X.var]
  X<-as.matrix(X)
  Z.var<-c("int",Z.var)
  Z<-base[,Z.var]
  Z<-as.matrix(Z)
  ni<-nrow(Z)

  X_a.var<-c("int",X_a.var)
  X_a<-base[,X_a.var]
  X_a<-as.matrix(X_a)
  X_a<-as.matrix(U * X_a)

```

```

Z_a.var<-c("int",Z_a.var)
Z_a<-base[,Z_a.var]
Z_a<-as.matrix(Z_a)
Z_a<-as.matrix(U * Z_a)
n_a<- sum(Z_a[,1])
V_a<-c(base[,V1],base[,V2])

#Calculations
c<-as.matrix(rep(0,length(Z.var)),ncol=1)
mu_2<- -(m/(1-m))*mu_1
Psi_cc.i<-solve(Psi_cc)
Psi_cd<-t(Psi_dc)
H<-Psi_dd-(Psi_dc %*% Psi_cc.i %*% Psi_cd )
H.i<-ginv(H)
X_at<-diag(2) %x% as.matrix(X_a)
Z_at<-diag(2) %x% as.matrix(Z_a)
Sigma.i<-ginv(Sigma)
Sigma.t<-Sigma %x% diag(nrow(X_a))
Sigma.it<-Sigma.i %x% diag(nrow(X_a))
Sigma.vech<-vech(Sigma)

Psi_cc.vech<-vech(Psi_cc)
Psi_ddcci<-Psi_dc %*% solve(Psi_cc) #Check this in the future
#(when more random effects, if still a vector)
H.vech<-vech(H)

A_a<-matrix(V_a-X_at %*% Gamma)

tempcheck<-ifelse(sum(V_a==0)==length(V_a),0,1) #if sum is zero,
#then all observations are zero
if(tempcheck==1){
  B<-t(Z_at) %*% Sigma.it %*% Z_at) + H.i
  B.i<-ginv(B)
} else {
  B<- H.i
  B.i<-H
}

E.T<-t(A_a) %*% Sigma.it %*% Z_at %*% B.i %*% H.i %*%
Psi_dc %*% Psi_cc.i
M_1.T<-m*t(mu_1)
M_2.T<-(1-m)*t(mu_2)
M.T<-M_1.T+M_2.T
D<- t(Psi_dc %*% Psi_cc.i) %*% H.i %*% B.i %*% t(Z_at) %*%
Sigma.it %*% Z_at %*% Psi_dc %*% Psi_cc.i

```



```

g_1.T<-E.T+(M_1.T %**% H.i %**% B.i %**% t(Z_at) %**% Sigma.it %**%
  Z_at %**% Psi_dc %**% Psi_cc.i)
g_2.T<-E.T+(M_2.T %**% H.i %**% B.i %**% t(Z_at) %**% Sigma.it %**%
  Z_at %**% Psi_dc %**% Psi_cc.i)
g.T <- E.T+(M.T %**% H.i %**% B.i %**% t(Z_at) %**% Sigma.it %**%
  Z_at %**% Psi_dc %**% Psi_cc.i)

# Max part for m
other<-list(U=U, X=X, Z=Z, Beta=Beta, Psi_cc.i = Psi_cc.i, D=D,
  g.To=g_1.T)
c_1t<- laplace2(lap.for.c,c,other)$mode
other<-list(U=U, X=X, Z=Z, Beta=Beta, Psi_cc.i = Psi_cc.i, D=D,
  g.To=g_2.T)
c_2t<- laplace2(lap.for.c,c,other)$mode

eta.1<- (X %**% Beta) + (Z %**% c_1t)
propi.1<-exp(eta.1)/(1+exp(eta.1))
propi.1<-apply(propi.1, 1, p.inf)
eta.1<- log(propi.1/(1-propi.1))
W_t.1<- diag(c(propi.1),ncol=length(propi.1)) %**%
  diag(c(1-propi.1),ncol=length(propi.1))
G.1<-(t(Z) %**% W_t.1 %**% Z ) + (Psi_cc.i+D)
G.1.i<-solve(G.1)

eta.2<- (X %**% Beta) + (Z %**% c_2t)
propi.2<-exp(eta.2)/(1+exp(eta.2))
propi.2<-apply(propi.2, 1, p.inf)
eta.2<- log(propi.2/(1-propi.2))
W_t.2<- diag(c(propi.2),ncol=length(propi.2)) %**%
  diag(c(1-propi.2),ncol=length(propi.2))
G.2<-(t(Z) %**% W_t.2 %**% Z ) + (Psi_cc.i+D)
G.2.i<-solve(G.2)

lu<-sum((U * eta.1) + log(1-propi.1))
qua<- t(c_1t-(solve(Psi_cc.i+D)%**%t(g_1.T))) %**% (Psi_cc.i+D) %**%
  (c_1t-(solve(Psi_cc.i+D)%**%t(g_1.T)))
f_1<-lu-((1/2)*qua)
lu<-sum((U * eta.2) + log(1-propi.2))
qua<-t(c_2t-(solve(Psi_cc.i+D)%**%t(g_2.T))) %**% (Psi_cc.i+D) %**%
  (c_2t-(solve(Psi_cc.i+D)%**%t(g_2.T)))
f_2<-lu-((1/2)*qua)

m3_t.1<- diag(W_t.1) * (1-2*propi.1)
m4_t.1<- diag(W_t.1 %**% (diag(dim(W_t.1)[1])-6*W_t.1))
m6_t.1<- m4_t.1*diag(diag(dim(W_t.1)[1])-12*W_t.1)-(12*(m3_t.1^2))
P1.1<- (1/8)* m4_t.1 %**% diag(Z %**% G.1.i %**% t(Z))^2
P2.1<- (1/48)*t(m6_t.1) %**% diag(Z %**% G.1.i %**% t(Z))^3

```

```

P3.1<- (15/72)* t(colSums(Z * m3_t.1 * diag(Z %%% G.1.i %%%
t(Z)))) %%% G.1.i %%% colSums(Z * m3_t.1 * diag(Z %%%
G.1.i %%% t(Z))) #Check that this runs well with more
# random effects
P.1<- 1 - P1.1 -P2.1 + P3.1
P.1<- ifelse(P.1>=1e-25,P.1,0.5)
lf1<- -(n_a+(q/2))*log(2*pi) - (n_a/2)*log(det(Sigma)) -
(1/2)*log(det(H)) + (1/2)*log(det(B.i)) -
(1/2)*log(det(Psi_cc)) - ((1/2)* t(A_a) %%% (Sigma.it -
Sigma.it %%% Z_at %%% B.i %%% t(Z_at) %%% Sigma.it) %%% A_a) +
(t(A_a) %%% Sigma.it %%% Z_at %%% B.i %%% H.i %%% t(M_1.T)) -
((1/2)*m*t(mu_1) %%% H.i %%% B.i %%% t(Z_at) %%% Sigma.it %%%
Z_at %%% mu_1) + ((1/2)* g_1.T %%% solve(Psi_cc.i+D) %%%
t(g_1.T)) + f_1 + (1/2)*log(det(G.1)) + log(P.1)
f1<-exp(lf1)
m3_t.2<- diag(W_t.2) * (1-2*propi.2)
m4_t.2<- diag(W_t.2 %%% (diag(dim(W_t.2)[1])-6*W_t.2))
m6_t.2<- m4_t.2*diag(diag(dim(W_t.2)[1])-12*W_t.2)-(12*m3_t.2^2)
P1.2<- (1/8)* m4_t.2 %%% diag(Z %%% G.2.i %%% t(Z))^2
P2.2<- (1/48)*t(m6_t.2) %%% diag(Z %%% G.2.i %%% t(Z))^3
P3.2<- (15/72)* t(colSums(Z * m3_t.2 * diag(Z %%% G.2.i %%%
t(Z)))) %%% G.2.i %%% colSums(Z * m3_t.2 * diag(Z %%%
G.2.i %%% t(Z))) #Check that this runs well with more random
#effects
P.2<- 1 - P1.2 -P2.2 + P3.2
P.2<- ifelse(P.2>=1e-25,P.1,0.5)
lf2<- -(n_a+(q/2))*log(2*pi) - (n_a/2)*log(det(Sigma)) -
(1/2)*log(det(H)) + (1/2)*log(det(B.i)) -
(1/2)*log(det(Psi_cc)) - ((1/2)* t(A_a) %%% (Sigma.it -
Sigma.it %%% Z_at %%% B.i %%% t(Z_at) %%% Sigma.it) %%% A_a) +
(t(A_a) %%% Sigma.it %%% Z_at %%% B.i %%% H.i %%% t(M_2.T)) -
((1/2)*(1-m)*t(mu_2) %%% H.i %%% B.i %%% t(Z_at) %%%
Sigma.it %%% Z_at %%% mu_2) + ((1/2)* g_2.T %%%
solve(Psi_cc.i+D) %%% t(g_2.T)) + f_2 + (1/2)*log(det(G.2))+
log(P.2)
f2<-exp(lf2)

m.new<-f1/(f1+f2)
m<-ifelse(is.na(m.new),0,m.new)
m<-ifelse(m<0.01,0.01,m)
m<-ifelse(m>0.99,0.99,m)

#max part for other par
other<-list(U=U, X=X, Z=Z, Beta=Beta, Psi_cc.i = Psi_cc.i, D=D,
g.To=E.T)
c_t<- laplace2(lap.for.c,c,other)$mode

```

```

eta<- (X %%% Beta) + (Z %%% c_t)
propi<-exp(eta)/(1+exp(eta))
propi<-apply(propi, 1, p.inf)
eta<- log(propi/(1-propi))
lu<-sum((U * eta) + log(1-propi))
qua<-t(c_t-(solve(Psi_cc.i+D)%%t(g.T))) %%% (Psi_cc.i+D) %%%
      (c_t-(solve(Psi_cc.i+D)%%t(g.T)))
f_v2<-lu-((1/2)*qua)
W_t<- diag(c(propi),ncol=length(propi)) %%%
      diag(c(1-propi),ncol=length(propi))
U_a<- (diag(1/diag(W_t)) %%% (U - propi)) + eta
G<-(t(Z) %%% W_t %%% Z ) + (Psi_cc.i+D)
G.i<-solve(G)

m3_t<-diag(W_t) * (1-2*propi)
m4_t<-diag(W_t %%% (diag(dim(W_t)[1])-6*W_t))
m5_t<-m3_t*diag(diag(dim(W_t)[1])-12*W_t)
m6_t<-m4_t*diag(diag(dim(W_t)[1])-12*W_t)-(12*m3_t^2)
m7_t<-m5_t*diag(diag(dim(W_t)[1])-12*W_t)-(36*m3_t*m4_t)

P1<- (1/8)* m4_t %%% diag(Z %%% G.i %%% t(Z))^2
P2<- (1/48)*t(m6_t) %%% diag(Z %%% G.i %%% t(Z))^3
P3<- (15/72) * t(colSums(Z * m3_t * diag(Z %%% G.i %%% t(Z)))) %%%
      G.i %%% colSums(Z * m3_t * diag(Z %%% G.i %%% t(Z))) #Check
      # that this runs well with more random efects
P<- (1 - P1 - P2 + P3)
P<- ifelse(P>1e-25,P,0.5)

#Scoring calculations
Ki<-matrix(0,nrow=ncol(Z) ,ncol=ncol(Z) )
for (k in 1:nrow(Z)) {
  Ki.temp<-t(Z[k,]) %%% m3_t[k] %%% Z[k,] %%% G.i %%% t(Z[k,])
  Ki=Ki+Ki.temp }
hi<-matrix(0,nrow=ncol(Z),ncol =ncol(Z) )
for (k in 1:nrow(Z)) {
  h.temp<-m3_t[k] %%% G.i %%%t(Z[k,]) %%% (Z[k,] %%% G.i) %%%
      (Z[k,] %%% G.i %%% Ki)
  hi=hi+h.temp }
fi<-matrix(0,nrow=ncol(Z),ncol =ncol(Z) )
for (k in 1:nrow(Z)) {
  fi.temp<-m6_t[k] %%% G.i %%%t(Z[k,]) %%% ((Z[k,] %%% G.i %%%
      t(Z[k,]))^2) %%% (Z[k,] %%% G.i)
  fi=fi+fi.temp }
Fi<-matrix(0,nrow=ncol(Z),ncol =ncol(Z) )
for (k in 1:nrow(Z)) {
  Fi.temp<-m4_t[k] %%% G.i %%% t(Z[k,]) %%% (Z[k,] %%% G.i %%%
      t(Z[k,])) %%% (Z[k,] %%% G.i)

```

```

    Fi=Fi+Fi.temp }
Rij<-matrix(0,nrow=nrow(Z),ncol=1)
for (k in 1:nrow(Z)) {
  Rij[k,1] <- -((1/8)*m5_t[k] %**% (Z[k,] %**% G.i %**% t(Z[k,]))^2) +
    ((1/4)* m3_t[k] %**% Z[k,] %**% Fi %**% t(Z[k,])) -
    ((1/48)*m7_t[k] %**% (Z[k,] %**% G.i %**% t(Z[k,]))^3) +
    ((1/16)* m3_t[k] %**% Z[k,] %**% fi %**% t(Z[k,])) -
    ((15/72)*m3_t[k] %**% (Z[k,] %**% G.i %**% Ki)^2) +
    ((15/36)* m4_t[k] %**% (t(Ki) %**% G.i %**% t(Z[k,])) %**%
    (Z[k,] %**% G.i %**% t(Z[k,]))) - ((15/36)*m3_t[k] %**%
    Z[k,] %**% hi %**% t(Z[k,]))
}
Fij<-matrix(0,nrow=nrow(Z),ncol=1)
for (k in 1: nrow(Z)) {
  Fij[k,1]<- ((1/4)*m4_t[k] %**% (Z[k,] %**% G.i %**% t(Z[k,]))) +
    ((1/16)*m6_t[k] %**% (Z[k,] %**% G.i %**% t(Z[k,]))^2) -
    ((15/36)*m3_t[k] %**% (Z[k,] %**% G.i %**% Ki))      }

Yi<- t(Z) %**% W_t %**% (U_a - (eta-(Z %**% c_t))) + t(E.T)
Hi<- t(Z_at) %**% Sigma.it %**% Z_at %**% B.i %**% H.i %**% Psi_dc %**%
  Psi_cc.i

Eij.int<-matrix(0,nrow=nrow(Z),ncol=1)
for (k in 1:nrow(Z)) {
  Eij.int[k,1]<- t(Z[k,]) %**% (((-1/2)*m3_t[k] %**% (Z[k,] %**% G.i %**%
    t(Z[k,]))) + (U[k] - propi[k]) + (Rij[k]/P) )}
Eij<- t(E.T)-(Psi_cc.i+D)%**%c_t +colSums(Eij.int)

Si<-((t(A_a) %**% Sigma.it %**% Z_at %**% B.i %**% H.i) %x% G.i)-(G.i
  %x% t(Yi) %**% G.i %**% t(Hi))-(t(Yi) %**% G.i %x% G.i %**% t(Hi))
ti<- -G.i %**% t(Psi_dc %**% Psi_cc.i) %**% H.i %**% B.i %**% t(Z_at) %**%
  Sigma.it %**% X_at
Di<- -(t(Yi) %**% G.i %x% G.i)

Mi<- -(t(Yi) %**% G.i %**% t(Psi_dc %**% Psi_cc.i) %**% t(Z_at) %**%
  Sigma.it %**% Z_at %**% B.i %x% (G.i %**% t(Psi_dc %**% Psi_cc.i)))+
  (t(A_a) %**% Sigma.it %**% Z_at %**% B.i %x% (G.i %**% t(Psi_dc %**%
  Psi_cc.i)))- (t(A_a) %**% Sigma.it %**% Z_at %**% B.i %x% (G.i %**%
  t(Psi_dc %**% Psi_cc.i) %**% H.i %**% B.i))+ (t(Yi) %**% G.i %**%
  t(Psi_dc %**% Psi_cc.i) %**% t(Z_at) %**% Sigma.it %**% Z_at %**% B.i
  %x% (G.i %**% t(Psi_dc %**% Psi_cc.i) %**% H.i %**% B.i))

Pi<- (t(Yi) %**% G.i %**% t(Psi_dc %**% Psi_cc.i) %**% t(Z_at) %**%
  Sigma.it %**% Z_at %**% B.i %**% t(Z_at) %x% (G.i %**% t(Psi_dc %**%
  Psi_cc.i) %**% H.i %**% B.i %**% t(Z_at)))- (t(Yi) %**% G.i %**%
  t(Psi_dc %**% Psi_cc.i) %**% t(Z_at) %x% (G.i %**% t(Psi_dc %**%
  Psi_cc.i) %**% H.i %**% B.i %**% t(Z_at)))- (t(A_a) %**% Sigma.it %**%

```

```

      Z_at %**% B.i %**% t(Z_at) %x% (G.i %**% t(Psi_dc %**% Psi_cc.i) %**%
      H.i %**% B.i %**% t(Z_at)))+ (t(A_a) %x% (G.i %**% t(Psi_dc %**%
      Psi_cc.i)%**% H.i %**% B.i %**% t(Z_at)))

#Score vectors
S_b.int1<-matrix(0,nrow=nrow(Z),ncol=ncol(X))
for (k in 1:nrow(Z)) {
  S_b.int1[k,]<-c((( -1/2)*m3_t[k] %**% (Z[k,] %**% G.i %**% t(Z[k,]))) *
    (t(X[k,]) - t(t(X) %**% W_t %**% Z %**% G.i %**% t(Z[k,]))))}
S_b.int1<-t(S_b.int1)
S_b.int1<-rowSums(S_b.int1)
S_b.int2<-matrix(0,nrow=nrow(Z),ncol=ncol(X))
for (k in 1:nrow(Z)) {
  S_b.int2[k,]<-Rij[k]*(t(X[k,]) - t(t(X) %**% W_t %**% Z %**% G.i %**%
    t(Z[k,])))}
S_b.int2<-t(S_b.int2)
S_b.int2<-rowSums(S_b.int2)
S_b<- t(X) %**% W_t %**% (U_a - eta) + S_b.int1 + (c((1/P)) * S_b.int2)
rule1<-sum(abs(S_b)<1e-1)
if(rule1>0){
  S_b<-matrix(0,nrow=ncol(X),1)
}

S_g<-(t(X_at) %**% Sigma.it %**% V_a) - (t(X_at) %**% Sigma.it %**%
  Z_at %**% B.i %**% t(Z_at) %**% Sigma.it %**% V_a) - (t(X_at) %**%
  Sigma.it %**% X_at %**% Gamma) + (t(X_at) %**% Sigma.it %**%
  Z_at %**% B.i %**% t(Z_at) %**% Sigma.it %**% X_at %**% Gamma) -
  (t(X_at) %**% Sigma.it %**% Z_at %**% B.i %**% H.i %**% Psi_dc %**%
  Psi_cc.i %**% c_t) + (t(ti) %**% Eij)

mu_1<-matrix(mu_1,ncol=1)
S_mu.int1<- (H.i %**% B.i %**% t(Z_at) %**% Sigma.it %**% Z_at)
S_mu.int2<- (t(Z_at) %**% Sigma.it %**% Z_at %**% B.i %**% H.i)
S_mu.int3<- (S_mu.int1 + S_mu.int2) %**% mu_1
S_mu.int4<- -(1/2)*(m+((m^2)/(1-m)))
S_mu<- c(S_mu.int4) * S_mu.int3

S_pcc.int1<-ifelse(q==1,t((Psi_cc.i %x% Psi_cc.i)),t((Psi_cc.i %x%
Psi_cc.i) %**% duplication.matrix(n=q)))
S_pcc.int2<-matrix(0,nrow=ncol(Z)*ncol(Z),ncol=ni)
for (k in 1:nrow(Z)) {
  S_pcc.int2[,k]<- vec(G.i %**% t(Z[k,]) %**% Z[k,] %**% G.i) * Fij[k]}
S_pcc.int2<-rowSums(S_pcc.int2)
S_pcc<- -S_pcc.int1 %**% ( (1/2)*vec(Psi_cc - G.i - c_t %**% t(c_t)) +
  t(Di) %**% Eij + (1/P) %**% ( S_pcc.int2 - (15/72)*vec(G.i %**%
  Ki %**% t(Ki) %**% G.i ) ) )
S_pcc<- as.matrix(apply(S_pcc, 1, function(x) {ifelse(abs(x)<1e-1 &

```

```

x!=0, 0, x)}),ncol=1)

S_pdc.int1<-matrix(0,nrow=ncol(Z_at)*ncol(Z),ncol=ni)
for (k in 1:nrow(Z)) {
  S_pdc.int1[,k]<- vec(Hi %**% G.i %**% t(Z[k,]) %**% Z[k,] %**% G.i) *
  Fij[k]}
S_pdc.int1<-rowSums(S_pdc.int1)
S_pdc<- -(diag(x = 1, ncol(Z)) %x% Hi) %**% (vec(G.i) + vec(c_t %**%
  t(c_t))) + vec(H.i %**% B.i %**% t(Z_at) %**% Sigma.it %**%
  A_a %**% t(c_t)) + t(Si) %**% Eij + c(2/P) * (S_pdc.int1 -
  (15/36) * vec(Hi %**% G.i %**% Ki %**% t(Ki) %**% G.i))

S_h.int1<-matrix(0,nrow=ncol(Z_at)*2,ncol=ni)
for (k in 1:nrow(Z)) {
  S_h.int1[,k]<- vec((diag(x = 1, ncol(Z_at)) - B.i %**% H.i) %**%
  Psi_dc %**% Psi_cc.i %**% G.i %**% t(Z[k,]) %**%
  Z[k,] %**% G.i %**% t(Psi_dc %**% Psi_cc.i) %**%
  t(Z_at) %**% Sigma.it %**% Z_at %**% B.i) * Fij[k]}
S_h.int1<-rowSums(S_h.int1)
S_h<-t((-H.i %x% H.i) %**% duplication.matrix(n=ncol(Z_at))) %**%
  (-1/2)*((B.i %**% t(Z_at) %**% Sigma.it %**% Z_at %**% Psi_dc %**%
  Psi_cc.i %x% (Psi_dc %**% Psi_cc.i))-(B.i %**% H.i %**% Psi_dc %**%
  Psi_cc.i %x%(B.i %**% t(Z_at) %**% Sigma.it %**% Z_at %**% Psi_dc %**%
  Psi_cc.i))) %**% vec(G.i) + (1/2)*vec(H - B.i - B.i %**%
  t(Z_at) %**% Sigma.it %**% A_a %**% t(A_a) %**% Sigma.it %**% Z_at %**%
  B.i + c(m) * B.i %**% t(Z_at) %**% Sigma.it %**% Z_at %**% mu_1 %**%
  t(mu_1) %**% H.i %**% B.i - c(m)*B.i %**% t(Z_at) %**% Sigma.it %**%
  Z_at %**% mu_1 %**% t(mu_1)+(1-c(m)) * B.i %**% t(Z_at) %**%
  Sigma.it %**% Z_at %**% mu_2 %**% t(mu_2) %**% H.i %**% B.i -
  (1-c(m))*B.i %**% t(Z_at) %**% Sigma.it %**% Z_at %**% mu_2**%
  t(mu_2)-Psi_dc %**% Psi_cc.i %**% c_t %**% t(c_t) %**% t(Psi_dc %**%
  Psi_cc.i) %**% t(Z_at) %**% Sigma.it %**% Z_at %**% B.i + 2*B.i %**%
  t(Z_at) %**% Sigma.it %**% A_a %**% t(c_t) %**% t(Psi_dc %**%
  Psi_cc.i) + (B.i %**% H.i %**% Psi_dc %**% Psi_cc.i %**% c_t %**%
  t(c_t) %**% t(Psi_dc %**% Psi_cc.i) %**% t(Z_at) %**% Sigma.it %**%
  Z_at %**% B.i) - 2*B.i %**% t(Z_at) %**% Sigma.it %**% A_a %**%
  t(c_t) %**% t(Psi_dc %**% Psi_cc.i) %**% H.i %**% B.i) + t(Mi)**%
  Eij + c(1/P)*(S_h.int1 - (15/72)*vec((diag(x = 1, ncol(Z_at))
  -B.i %**% H.i) %**% Psi_dc %**% Psi_cc.i %**% G.i %**% Ki %**%
  t(Ki) %**% G.i %**% t(Psi_dc %**% Psi_cc.i) %**% t(Z_at) %**%
  Sigma.it %**% Z_at %**% B.i) ) )

S_sigma.int1<-matrix(0,nrow=(dim(Sigma.it)[1]*dim(Sigma.it)[2]),
  ncol=nrow(Z))
for (k in 1:nrow(Z)) {
  S_sigma.int1[,k]<- vec(Z_at %**% B.i %**% H.i %**% (Psi_dc %**%
  Psi_cc.i) %**% G.i %**% t(Z[k,]) %**% Z[k,] %**%

```

```

      G.i %%% t(Psi_dc %%% Psi_cc.i) %%%
      (t(Z_at)-t(Z_at) %%% Sigma.it %%% Z_at %%%
      B.i %%% t(Z_at))) * Fij[k]}
S_sigma.int1<-rowSums(S_sigma.int1)
S_sigma.int2<- ((1/2)*vec(- Z_at %%% B.i %%% t(Z_at) - A_a %%%
t(A_a) + (2*A_a %%% t(A_a) %%% Sigma.it %%% Z_at %%%
B.i %%% t(Z_at)) - Z_at %%% B.i %%% t(Z_at) %%%
Sigma.it %%% A_a %%% t(A_a) %%% Sigma.it %%% Z_at %%%
B.i %%% t(Z_at))- (c(m)/2)*vec(Z_at %%% B.i %%% H.i %%%
mu_1 %%% t(mu_1) %%% t(Z_at)- Z_at %%% B.i %%% H.i %%%
mu_1 %%% t(mu_1) %%% t(Z_at) %%% Sigma.it %%% Z_at %%%
B.i %%% t(Z_at)))-((1-c(m))/2)*vec(Z_at %%% B.i %%%
H.i %%% mu_2 %%% t(mu_2) %%% t(Z_at)- Z_at %%% B.i %%%
H.i %%% mu_2 %%% t(mu_2) %%% t(Z_at) %%% Sigma.it %%%
Z_at %%% B.i %%% t(Z_at))+t(Pi) %%% Eij - (1/2)*
(((Z_at %%% B.i %%% H.i %%% Psi_dc %%% Psi_cc.i) %x%
(Z_at %%% Psi_dc %%% Psi_cc.i))-((Z_at %%% B.i %%%
H.i %%% Psi_dc %%% Psi_cc.i) %x% (Z_at %%% B.i %%%
t(Z_at) %%% Sigma.it %%% Z_at %%% Psi_dc %%%
Psi_cc.i))) %%% vec(G.i)+(1/2)*vec(-(Z_at %%% B.i %%%
H.i %%% Psi_dc %%% Psi_cc.i %%% c_t %%% t(c_t) %%%
t(Psi_dc %%% Psi_cc.i) %%% t(Z_at))+ (Z_at %%% B.i %%%
H.i %%% Psi_dc %%% Psi_cc.i %%% c_t %%% t(c_t) %%%
t(Psi_dc %%% Psi_cc.i) %%% t(Z_at) %%% Sigma.it %%%
Z_at %%% B.i %%% t(Z_at))+2*(A_a %%% t(c_t) %%%
t(Psi_dc %%% Psi_cc.i) %%% H.i %%% B.i %%% t(Z_at))-
2*(Z_at %%% B.i %%% t(Z_at) %%% Sigma.it %%% A_a %%%
t(c_t) %%% t(Psi_dc %%% Psi_cc.i) %%% H.i %%% B.i %%%
t(Z_at)))+(1/c(P))* (S_sigma.int1 - (15/72)*vec(Z_at %%%
B.i %%% H.i %%% (Psi_dc %%% Psi_cc.i) %%% G.i %%%
Ki %%% t(Ki) %%% G.i %%% t(Psi_dc %%% Psi_cc.i) %%%
(t(Z_at)-t(Z_at) %%% Sigma.it %%% Z_at %%% B.i %%%
t(Z_at)))) )
Q_sigma<-(diag(2) %x% commutation.matrix(ni, 2) %x% diag(ni)) %%%
(diag(2) %x% diag(2) %x% vec(diag(ni)))
S_sigma.int3<- t((-Sigma.i %x% Sigma.i) %%%
duplication.matrix(n=ncol(Z_at))) %%%
((n_a/2)*vec(Sigma) )
S_sigma<- S_sigma.int3 - t(duplication.matrix(n=2)) %%% t(Q_sigma) %%%
(t(Sigma.it) %x% Sigma.it) %%% S_sigma.int2
S<-as.matrix(c(S_b=S_b,S_g=S_g,S_mu=S_mu,S_sigma=S_sigma,S_pcc=S_pcc,
S_pdc=S_pdc,S_h=S_h),nrow=23,ncol=1)
S_0<-matrix(0,nrow=23,ncol=1) #the future change this to nrow related
#with S
for(l in 1:23){
  S_0[l,1]<-ifelse(is.na(S[l,1])==T,0,S[l,1])
  #S_0[l,1]<-ifelse(abs(S[l,1])<1e-4,0,S[l,1])

```

```

}

S_0m<-S_0 %*% t(S_0)
S_0m[is.infinite(S_0m)] <- 0
scorei<-t(ginv(S_0m) %*% S_0)
max.vec<-list(m=m,scorei=scorei)
return(max.vec)
}

lklhd.dev<-function(id,datafile,initial,X.var=X.var,Z.var=Z.var,
                    X_a.var=X_a.var,Z_a.var=Z_a.var,U.var=U.var,V1=V1,V2=V2){
  #This function calculates the Deviance
  Beta=initial$Beta
  Gamma=initial$Gamma
  Gamma<-matrix(Gamma,ncol=1)
  Psi_cc=initial$Psi_cc
  Psi_dd=initial$Psi_dd
  corr<-initial$corr
  Var.V1<-initial$Var.V1
  Var.V2<-initial$Var.V2
  Psi_dc=initial$Psi_dc
  m=initial$m
  mu_1=initial$mu_1

  p<-length(Beta)
  q<-ifelse(length(Psi_cc)>1,dim(Psi_cc)[1],length(Psi_cc))
  p_a<-length(Gamma)
  q_a<-ifelse(length(Psi_dd)>1,dim(Psi_dd)[1],length(Psi_dd))
  Sigma<-matrix(c(Var.V1,rep(corr*sqrt(Var.V1)*sqrt(Var.V2),2),Var.V2),
                nrow = 2,ncol = 2)

  #Data
  base<-datafile[datafile$patch==id,]
  base$int <- rep(1,nrow(base))
  U<-base[,U.var]
  X.var<-c("int",X.var)
  X<-base[,X.var]
  X<-as.matrix(X)
  Z.var<-c("int",Z.var)
  Z<-base[,Z.var]
  Z<-as.matrix(Z)
  ni<-nrow(Z)

  X_a.var<-c("int",X_a.var)
  X_a<-base[,X_a.var]
  X_a<-as.matrix(X_a)
  X_a<-as.matrix(U * X_a)

```



```

Z_a.var<-c("int",Z_a.var)
Z_a<-base[,Z_a.var]
Z_a<-as.matrix(Z_a)
Z_a<-as.matrix(U * Z_a)
n_a<- sum(Z_a[,1])
V_a<-c(base[,V1],base[,V2])

#Calculations
c<-as.matrix(rep(0,length(Z.var)),ncol=1)
mu_2<- -(m/(1-m))*mu_1
Psi_cc.i<-solve(Psi_cc)
Psi_cd<-t(Psi_dc)
H<-Psi_dd-(Psi_dc %*% Psi_cc.i %*% Psi_cd )
H.i<-ginv(H)
X_at<-diag(2) %x% as.matrix(X_a)
Z_at<-diag(2) %x% as.matrix(Z_a)
Sigma.i<-ginv(Sigma)
Sigma.t<-Sigma %x% diag(nrow(X_a))
Sigma.it<-Sigma.i %x% diag(nrow(X_a))
Sigma.vech<-vech(Sigma)

Psi_cc.vech<-vech(Psi_cc)
Psi_ddcci<-Psi_dc %*% solve(Psi_cc) #Check this in the future
#(when more random effects, if still a vector)
H.vech<-vech(H)

A_a<-matrix(V_a-X_at %*% Gamma)

tempcheck<-ifelse(sum(V_a==0)==length(V_a),0,1) #if sum is zero,
# the all observations are zero
if(tempcheck==1){
  B<-t(Z_at) %*% Sigma.it %*% Z_at) + H.i
  B.i<-ginv(B)
} else {
  B<- H.i
  B.i<-H
}

E.T<-t(A_a) %*% Sigma.it %*% Z_at %*% B.i %*% H.i %*% Psi_dc %*%
Psi_cc.i
M_1.T<-m*t(mu_1)
M_2.T<-(1-m)*t(mu_2)
M.T<-M_1.T+M_2.T
D<- t(Psi_dc %*% Psi_cc.i) %*% H.i %*% B.i %*% t(Z_at) %*%
Sigma.it %*% Z_at %*% Psi_dc %*% Psi_cc.i
g_1.T<-E.T+(M_1.T %*% H.i %*% B.i %*% t(Z_at) %*% Sigma.it %*%

```

```

      Z_at %%% Psi_dc %%% Psi_cc.i)
g_2.T<-E.T+(M_2.T %%% H.i %%% B.i %%% t(Z_at) %%% Sigma.it %%%
      Z_at %%% Psi_dc %%% Psi_cc.i)
g.T <- E.T+(M.T %%% H.i %%% B.i %%% t(Z_at) %%% Sigma.it %%%
      Z_at %%% Psi_dc %%% Psi_cc.i)

#max part for other par
other<-list(U=U, X=X, Z=Z, Beta=Beta, Psi_cc.i = Psi_cc.i, D=D,
           g.To=E.T)
c_t<- laplace2(lap.for.c,c,other)$mode

eta<- (X %%% Beta) + (Z %%% c_t)
propi<-exp(eta)/(1+exp(eta))
propi<-apply(propi, 1, p.inf)
eta<- log(propi/(1-propi))
lu<-sum((U * eta) + log(1-propi))
qua<-t(c_t-(solve(Psi_cc.i+D)%%t(g.T))) %%% (Psi_cc.i+D) %%%
      (c_t-(solve(Psi_cc.i+D)%%t(g.T)))
f_v2<-lu-((1/2)*qua)
W_t<- diag(c(propi),ncol=length(propi)) %%%
      diag(c(1-propi),ncol=length(propi))
U_a<- (diag(1/diag(W_t)) %%% (U - propi)) + eta
G<-t(Z) %%% W_t %%% Z ) + (Psi_cc.i+D)
G.i<-solve(G)

m3_t<-diag(W_t) * (1-2*propi)
m4_t<-diag(W_t %%% (diag(dim(W_t)[1])-6*W_t))
m5_t<-m3_t*diag(diag(dim(W_t)[1])-12*W_t)
m6_t<-m4_t*diag(diag(dim(W_t)[1])-12*W_t)-(12*m3_t^2)
m7_t<-m5_t*diag(diag(dim(W_t)[1])-12*W_t)-(36*m3_t*m4_t)

P1<- (1/8)* m4_t %%% diag(Z %%% G.i %%% t(Z))^2
P2<- (1/48)*t(m6_t) %%% diag(Z %%% G.i %%% t(Z))^3
P3<- (15/72) * t(colSums(Z * m3_t * diag(Z %%% G.i %%% t(Z)))) %%%
      G.i %%% colSums(Z * m3_t * diag(Z %%% G.i %%% t(Z))) #Check
      #that this runs well with more random effects
P<- (1 - P1 - P2 + P3)
P<- ifelse(P>1e-25,P,0.5)

#loglike
lf<- -(n_a+(q/2))*log(2*pi) - (n_a/2)*log(det(Sigma)) -
      (1/2)*log(det(H)) + (1/2)*log(det(B.i)) - (1/2)*log(det(Psi_cc)) -
      ((1/2)* t(A_a) %%% (Sigma.it - Sigma.it %%% Z_at %%% B.i %%%
      t(Z_at) %%% Sigma.it) %%% A_a) + (t(A_a) %%% Sigma.it %%% Z_at %%%
      B.i %%% H.i %%% t(M.T)) - ((1/2)*(m)*t(mu_1) %%% H.i %%% B.i %%%
      t(Z_at) %%% Sigma.it %%% Z_at %%% mu_1) - ((1/2)*(1-m)*t(mu_2) %%%
      H.i %%% B.i %%% t(Z_at) %%% Sigma.it %%% Z_at %%% mu_2) +

```

```

      ((1/2)* g.T %*% solve(Psi_cc.i+D) %*% t(g.T)) + f_v2 +
      (1/2)*log(det(G))+log(P)

  return(dev=lf)
}

lklhd.var<-function(id,datafile,est,X.var=X.var,Z.var=Z.var,X_a.var=
  X_a.var,Z_a.var=Z_a.var,U.var=U.var,V1=V1,V2=V2){
  # Calculates SE of estimates
  initial<-est
  Beta=initial$Beta
  Gamma=initial$Gamma
  Gamma<-matrix(Gamma,ncol=1)
  Psi_cc=initial$Psi_cc
  Psi_dd=initial$Psi_dd
  corr<-initial$corr
  Var.V1<-initial$Var.V1
  Var.V2<-initial$Var.V2
  Psi_dc=initial$Psi_dc
  m=initial$m
  mu_1=initial$mu_1

  p<-length(Beta)
  q<-ifelse(length(Psi_cc)>1,dim(Psi_cc)[1],length(Psi_cc))
  p_a<-length(Gamma)
  q_a<-ifelse(length(Psi_dd)>1,dim(Psi_dd)[1],length(Psi_dd))
  Sigma<-matrix(c(Var.V1,rep(corr*sqrt(Var.V1)*sqrt(Var.V2),2),Var.V2),
    nrow = 2,ncol = 2)

  #Data
  base<-datafile[datafile$patch==id,]
  base$int <- rep(1,nrow(base))
  U<-base[,U.var]
  X.var<-c("int",X.var)
  X<-base[,X.var]
  X<-as.matrix(X)
  Z.var<-c("int",Z.var)
  Z<-base[,Z.var]
  Z<-as.matrix(Z)
  ni<-nrow(Z)

  X_a.var<-c("int",X_a.var)
  X_a<-base[,X_a.var]
  X_a<-as.matrix(X_a)
  X_a<-as.matrix(U * X_a)

  Z_a.var<-c("int",Z_a.var)

```

```

Z_a<-base[,Z_a.var]
Z_a<-as.matrix(Z_a)
Z_a<-as.matrix(U * Z_a)
n_a<- sum(Z_a[,1])
V_a<-c(base[,V1],base[,V2])

#Calculations
c<-as.matrix(rep(0,length(Z.var)),ncol=1)
mu_2<- -(m/(1-m))*mu_1
Psi_cc.i<-solve(Psi_cc)
Psi_cd<-t(Psi_dc)
H<-Psi_dd-(Psi_dc %*% Psi_cc.i %*% Psi_cd )
H.i<-ginv(H)
X_at<-diag(2) %x% as.matrix(X_a)
Z_at<-diag(2) %x% as.matrix(Z_a)
Sigma.i<-ginv(Sigma)
Sigma.t<-Sigma %x% diag(nrow(X_a))
Sigma.it<-Sigma.i %x% diag(nrow(X_a))
Sigma.vech<-vech(Sigma)

Psi_cc.vech<-vech(Psi_cc)
Psi_ddcci<-Psi_dc %*% solve(Psi_cc) #Check this in the future (when
#more random effects, if still a vector)
H.vech<-vech(H)

A_a<-matrix(V_a-X_at %*% Gamma)

tempcheck<-ifelse(sum(V_a==0)==length(V_a),0,1) #if sum is zero, the
#all observations are zero
if(tempcheck==1){
  B<-(t(Z_at) %*% Sigma.it %*% Z_at) + H.i
  B.i<-ginv(B)
} else {
  B<- H.i
  B.i<-H
}

E.T<-t(A_a) %*% Sigma.it %*% Z_at %*% B.i %*% H.i %*% Psi_dc %*%
Psi_cc.i
M_1.T<-m*t(mu_1)
M_2.T<-(1-m)*t(mu_2)
M.T<-M_1.T+M_2.T
D<- t(Psi_dc %*% Psi_cc.i) %*% H.i %*% B.i %*% t(Z_at) %*%
Sigma.it %*% Z_at %*% Psi_dc %*% Psi_cc.i
g_1.T<-E.T+(M_1.T %*% H.i %*% B.i %*% t(Z_at) %*% Sigma.it %*%
Z_at %*% Psi_dc %*% Psi_cc.i)
g_2.T<-E.T+(M_2.T %*% H.i %*% B.i %*% t(Z_at) %*% Sigma.it %*%

```

```

      Z_at %%% Psi_dc %%% Psi_cc.i)
g.T <- E.T+(M.T %%% H.i %%% B.i %%% t(Z_at) %%% Sigma.it %%%
      Z_at %%% Psi_dc %%% Psi_cc.i)

#max part for other par
other<-list(U=U, X=X, Z=Z, Beta=Beta, Psi_cc.i = Psi_cc.i, D=D,
            g.To=E.T)
c_t<- laplace2(lap.for.c,c,other)$mode

eta<- (X %%% Beta) + (Z %%% c_t)
propi<-exp(eta)/(1+exp(eta))
propi<-apply(propi, 1, p.inf)
eta<- log(propi/(1-propi))
lu<-sum((U * eta) + log(1-propi))
qua<-t(c_t-(solve(Psi_cc.i+D)%%t(g.T))) %%% (Psi_cc.i+D) %%%
      (c_t-(solve(Psi_cc.i+D)%%t(g.T)))
f_v2<-lu-((1/2)*qua)
W_t<- diag(c(propi),ncol=length(propi)) %%%
      diag(c(1-propi),ncol=length(propi))
U_a<- (diag(1/diag(W_t)) %%% (U - propi)) + eta
G<-t(Z) %%% W_t %%% Z ) + (Psi_cc.i+D)
G.i<-solve(G)

m3_t<-diag(W_t) * (1-2*propi)
m4_t<-diag(W_t %%% (diag(dim(W_t)[1])-6*W_t))
m5_t<-m3_t*diag(diag(dim(W_t)[1])-12*W_t)
m6_t<-m4_t*diag(diag(dim(W_t)[1])-12*W_t)-(12*m3_t^2)
m7_t<-m5_t*diag(diag(dim(W_t)[1])-12*W_t)-(36*m3_t*m4_t)

P1<- (1/8)* m4_t %%% diag(Z %%% G.i %%% t(Z))^2
P2<- (1/48)*t(m6_t) %%% diag(Z %%% G.i %%% t(Z))^3
P3<- (15/72) * t(colSums(Z * m3_t * diag(Z %%% G.i %%% t(Z)))) %%%
      G.i %%% colSums(Z * m3_t * diag(Z %%% G.i %%% t(Z))) #Check
      #that this runs well with more random efects
P<- (1 - P1 - P2 + P3)
P<- ifelse(P>1e-25,P,0.5)

#Scoring calculations
Ki<-matrix(0,nrow=ncol(Z) ,ncol=ncol(Z) )
for (k in 1:nrow(Z)) {
  Ki.temp<-t(Z[k,]) %%% m3_t[k] %%% Z[k,] %%% G.i %%% t(Z[k,])
  Ki=Ki+Ki.temp }
hi<-matrix(0,nrow=ncol(Z),ncol =ncol(Z) )
for (k in 1:nrow(Z)) {
  h.temp<-m3_t[k] %%% G.i %%%t(Z[k,]) %%% (Z[k,] %%% G.i) %%%
      (Z[k,] %%% G.i %%% Ki)
  hi=hi+h.temp }

```

```

fi<-matrix(0,nrow=ncol(Z),ncol =ncol(Z) )
for (k in 1:nrow(Z)) {
  fi.temp<-m6_t[k] %*% G.i %*%t(Z[k,]) %*% ((Z[k,] %*% G.i %*%
    t(Z[k,]))^2) %*% (Z[k,] %*% G.i)
  fi=fi+fi.temp }
Fi<-matrix(0,nrow=ncol(Z),ncol =ncol(Z) )
for (k in 1:nrow(Z)) {
  Fi.temp<-m4_t[k] %*% G.i %*% t(Z[k,]) %*% (Z[k,] %*% G.i %*%
    t(Z[k,])) %*% (Z[k,] %*% G.i)
  Fi=Fi+Fi.temp }
Rij<-matrix(0,nrow=nrow(Z),ncol=1)
for (k in 1:nrow(Z)) {
  Rij[k,1] <- -((1/8)*m5_t[k] %*% (Z[k,] %*% G.i %*% t(Z[k,]))^2) +
    ((1/4)* m3_t[k] %*% Z[k,] %*% Fi %*% t(Z[k,])) -
    ((1/48)*m7_t[k] %*% (Z[k,] %*% G.i %*% t(Z[k,]))^3) +
    ((1/16)* m3_t[k] %*% Z[k,] %*% fi %*% t(Z[k,])) -
    ((15/72)*m3_t[k] %*% (Z[k,] %*% G.i %*% Ki)^2) +
    ((15/36)* m4_t[k] %*% (t(Ki) %*% G.i %*% t(Z[k,])) %*%
    (Z[k,] %*% G.i %*% t(Z[k,]))) -
    ((15/36)*m3_t[k] %*% Z[k,] %*% hi %*% t(Z[k,]))
}
Fij<-matrix(0,nrow=nrow(Z),ncol=1)
for (k in 1:nrow(Z)) {
  Fij[k,1]<- ((1/4)*m4_t[k] %*% (Z[k,] %*% G.i %*% t(Z[k,]))) +
    ((1/16)*m6_t[k] %*% (Z[k,] %*% G.i %*% t(Z[k,]))^2) -
    ((15/36)*m3_t[k] %*% (Z[k,] %*% G.i %*% Ki)) }

Yi<- t(Z) %*% W_t %*% (U_a - (eta-(Z %*% c_t))) + t(E.T)
Hi<- t(Z_at) %*% Sigma.it %*% Z_at %*% B.i %*% H.i %*% Psi_dc %*%
  Psi_cc.i

Eij.int <- matrix(0,nrow=nrow(Z),ncol=1)
for (k in 1:nrow(Z)) {
  Eij.int[k,1]<- t(Z[k,]) %*% (((-1/2)*m3_t[k] %*% (Z[k,] %*% G.i %*%
    t(Z[k,]))) + (U[k] - propi[k]) + (Rij[k]/P) )}
Eij<- t(E.T)-(Psi_cc.i+D)%*%c_t +colSums(Eij.int)

Si<-((t(A_a) %*% Sigma.it %*% Z_at %*% B.i %*% H.i) %x% G.i)-(G.i %x%
  t(Yi) %*% G.i %*% t(Hi))-(t(Yi) %*% G.i %x% G.i %*% t(Hi))
ti<- -G.i %*% t(Psi_dc %*% Psi_cc.i) %*% H.i %*% B.i %*% t(Z_at) %*%
  Sigma.it %*% X_at
Di<- -(t(Yi) %*% G.i %x% G.i)

Mi<- -(t(Yi) %*% G.i %*% t(Psi_dc %*% Psi_cc.i) %*% t(Z_at) %*%
  Sigma.it %*% Z_at %*% B.i %x% (G.i %*% t(Psi_dc %*% Psi_cc.i)))+
  (t(A_a) %*% Sigma.it %*% Z_at %*% B.i %x% (G.i %*% t(Psi_dc %*%
  Psi_cc.i)))-(t(A_a) %*% Sigma.it %*% Z_at %*% B.i %x% (G.i %*%

```

```

t(Psi_dc %** Psi_cc.i) %** H.i %** B.i))+t(Yi) %** G.i %**
t(Psi_dc %** Psi_cc.i) %** t(Z_at) %** Sigma.it %** Z_at %**
B.i %x% (G.i %** t(Psi_dc %** Psi_cc.i) %** H.i %** B.i))

Pi<- (t(Yi) %** G.i %** t(Psi_dc %** Psi_cc.i) %** t(Z_at) %**
Sigma.it %** Z_at %** B.i %** t(Z_at) %x% (G.i %** t(Psi_dc %**
Psi_cc.i) %** H.i %** B.i %** t(Z_at)))-(t(Yi) %** G.i %**
t(Psi_dc %** Psi_cc.i) %** t(Z_at) %x% (G.i %** t(Psi_dc %**
Psi_cc.i) %** H.i %** B.i %** t(Z_at)))-(t(A_a) %** Sigma.it %**
Z_at %** B.i %** t(Z_at) %x% (G.i %** t(Psi_dc %** Psi_cc.i) %**
H.i %** B.i %** t(Z_at)))+(t(A_a) %x% (G.i %** t(Psi_dc %**
Psi_cc.i)%** H.i %** B.i %** t(Z_at)))

#Score vectors
S_b.int1<-matrix(0,nrow=nrow(Z),ncol=ncol(X))
for (k in 1:nrow(Z)) {
  S_b.int1[k,]<-c(((1/2)*m3_t[k] %** (Z[k,] %** G.i %** t(Z[k,]))) *
(t(X[k,]) - t(t(X) %** W_t %** Z %** G.i %** t(Z[k,]))))}
S_b.int1<-t(S_b.int1)
S_b.int1<-rowSums(S_b.int1)
S_b.int2<-matrix(0,nrow=nrow(Z),ncol=ncol(X))
for (k in 1:nrow(Z)) {
  S_b.int2[k,]<-Rij[k]*(t(X[k,]) - t(t(X) %** W_t %** Z %** G.i %**
t(Z[k,])))}
S_b.int2<-t(S_b.int2)
S_b.int2<-rowSums(S_b.int2)
S_b<- t(X) %** W_t %** (U_a - eta) + S_b.int1 + (c((1/P)) * S_b.int2)
rule1<-sum(abs(S_b)<1e-1)
if(rule1>0){
  S_b<-matrix(0,nrow=ncol(X),1)
}

S_g<-(t(X_at) %** Sigma.it %** V_a) - (t(X_at) %** Sigma.it %**
Z_at %** B.i %** t(Z_at) %** Sigma.it %** V_a) - (t(X_at) %**
Sigma.it %** X_at %** Gamma) + (t(X_at) %** Sigma.it %**
Z_at %** B.i %** t(Z_at) %** Sigma.it %** X_at %** Gamma) -
(t(X_at) %** Sigma.it %** Z_at %** B.i %** H.i %** Psi_dc %**
Psi_cc.i %** c_t) + (t(ti) %** Eij)

mu_1<-matrix(mu_1,ncol=1)
S_mu.int1<- (H.i %** B.i %** t(Z_at) %** Sigma.it %** Z_at)
S_mu.int2<- (t(Z_at) %** Sigma.it %** Z_at %** B.i %** H.i)
S_mu.int3<- (S_mu.int1 + S_mu.int2) %** mu_1
S_mu.int4<- -(1/2)*(m+((m^2)/(1-m)))
S_mu<- c(S_mu.int4) * S_mu.int3

S_pcc.int1<-ifelse(q==1,t((Psi_cc.i %x% Psi_cc.i)),t((Psi_cc.i %x%

```

```

Psi_cc.i) %%% duplication.matrix(n=q)))
S_pcc.int2<-matrix(0,nrow=ncol(Z)*ncol(Z),ncol=ni)
for (k in 1:nrow(Z)) {
  S_pcc.int2[,k]<- vec(G.i %%% t(Z[k,]) %%% Z[k,] %%% G.i) * Fij[k]}
  S_pcc.int2<-rowSums(S_pcc.int2)
  S_pcc<- -S_pcc.int1 %%% ( (1/2)*vec(Psi_cc - G.i - c_t %%% t(c_t)) +
    t(Di) %%% Eij + (1/P) %%% ( S_pcc.int2 - (15/72)*vec(G.i %%%
    Ki %%% t(Ki) %%% G.i ) ) )
S_pcc<- as.matrix(apply(S_pcc, 1, function(x) {ifelse(abs(x)<1e-1 &
  x!=0, 0, x)}),ncol=1)

S_pdc.int1<-matrix(0,nrow=ncol(Z_at)*ncol(Z),ncol=ni)
for (k in 1:nrow(Z)) {
  S_pdc.int1[,k]<- vec(Hi %%% G.i %%% t(Z[k,]) %%% Z[k,] %%% G.i) *
    Fij[k]}
S_pdc.int1<-rowSums(S_pdc.int1)
S_pdc<- -(diag(x = 1, ncol(Z)) %x% Hi) %%% (vec(G.i) + vec(c_t %%%
  t(c_t))) + vec(H.i %%% B.i %%% t(Z_at) %%% Sigma.it %%%
  A_a %%% t(c_t)) + t(Si) %%% Eij + c(2/P) * (S_pdc.int1 -
  (15/36) * vec(Hi %%% G.i %%% Ki %%% t(Ki) %%% G.i))

S_h.int1<-matrix(0,nrow=ncol(Z_at)*2,ncol=ni)
for (k in 1:nrow(Z)) {
  S_h.int1[,k]<- vec((diag(x = 1, ncol(Z_at)) - B.i %%% H.i) %%%
    Psi_dc %%% Psi_cc.i %%% G.i %%% t(Z[k,]) %%%
    Z[k,] %%% G.i %%% t(Psi_dc %%% Psi_cc.i) %%%
    t(Z_at) %%% Sigma.it %%% Z_at %%% B.i) *
    Fij[k]}
S_h.int1<-rowSums(S_h.int1)
S_h<-t((-H.i %x% H.i) %%% duplication.matrix(n=ncol(Z_at))) %%%
  ( -(1/2)*((B.i %%% t(Z_at) %%% Sigma.it %%% Z_at %%% Psi_dc %%%
  Psi_cc.i %x% (Psi_dc %%% Psi_cc.i))- (B.i %%% H.i %%% Psi_dc %%%
  Psi_cc.i %x%(B.i %%% t(Z_at) %%% Sigma.it %%% Z_at %%% Psi_dc %%%
  Psi_cc.i))) %%% vec(G.i) + (1/2)*vec(H - B.i - B.i %%%
  t(Z_at) %%% Sigma.it %%% A_a %%% t(A_a) %%% Sigma.it %%% Z_at %%%
  B.i + c(m) * B.i %%% t(Z_at) %%% Sigma.it %%% Z_at %%% mu_1 %%%
  t(mu_1) %%% H.i %%% B.i - c(m)*B.i %%% t(Z_at) %%% Sigma.it %%%
  Z_at %%% mu_1 %%% t(mu_1)+(1-c(m)) * B.i %%% t(Z_at) %%%
  Sigma.it %%% Z_at %%% mu_2 %%% t(mu_2) %%% H.i %%% B.i -
  (1-c(m))*B.i %%% t(Z_at) %%% Sigma.it %%% Z_at %%% mu_2 %%%
  t(mu_2)-Psi_dc %%% Psi_cc.i %%% c_t %%% t(c_t) %%% t(Psi_dc %%%
  Psi_cc.i) %%% t(Z_at) %%% Sigma.it %%% Z_at %%% B.i + 2*B.i %%%
  t(Z_at) %%% Sigma.it %%% A_a %%% t(c_t) %%% t(Psi_dc %%%
  Psi_cc.i) + (B.i %%% H.i %%% Psi_dc %%% Psi_cc.i %%% c_t %%%
  t(c_t) %%% t(Psi_dc %%% Psi_cc.i) %%% t(Z_at) %%% Sigma.it %%%
  Z_at %%% B.i) - 2*B.i %%% t(Z_at) %%% Sigma.it %%% A_a %%%
  t(c_t) %%% t(Psi_dc %%% Psi_cc.i) %%% H.i %%% B.i) + t(Mi))%%

```



```

Eij + c(1/P)*(S_h.int1 - (15/72)*vec((diag(x = 1, ncol(Z_at))-
B.i %**% H.i) %**% Psi_dc %**% Psi_cc.i %**% G.i %**% Ki %**% t(Ki) %**%
G.i %**% t(Psi_dc %**% Psi_cc.i) %**% t(Z_at) %**% Sigma.it %**%
Z_at %**% B.i) ) )

S_sigma.int1<-matrix(0,nrow=(dim(Sigma.it)[1]*dim(Sigma.it)[2]),
ncol=nrow(Z))
for (k in 1:nrow(Z)) {
  S_sigma.int1[,k]<- vec(Z_at %**% B.i %**% H.i %**% (Psi_dc %**%
Psi_cc.i) %**% G.i %**% t(Z[k,]) %**% Z[k,] %**%
G.i %**% t(Psi_dc %**% Psi_cc.i) %**%
(t(Z_at)-t(Z_at) %**% Sigma.it %**% Z_at %**%
B.i %**% t(Z_at))) * Fij[k]}
S_sigma.int1<-rowSums(S_sigma.int1)
S_sigma.int2<- ((1/2)*vec(- Z_at %**% B.i %**% t(Z_at) -A_a %**% t(A_a)+
(2*A_a %**% t(A_a) %**% Sigma.it %**% Z_at %**% B.i %**%
t(Z_at)) -Z_at %**% B.i %**% t(Z_at) %**% Sigma.it %**%
A_a %**% t(A_a) %**% Sigma.it %**% Z_at %**% B.i %**%
t(Z_at))-(c(m)/2)*vec(Z_at %**% B.i %**% H.i %**% mu_1 %**%
t(mu_1) %**% t(Z_at)- Z_at %**% B.i %**% H.i %**% mu_1 %**%
t(mu_1) %**% t(Z_at) %**% Sigma.it %**% Z_at %**% B.i %**%
t(Z_at))-((1-c(m))/2)*vec(Z_at %**% B.i %**% H.i %**%
mu_2 %**% t(mu_2) %**% t(Z_at)- Z_at %**% B.i %**% H.i %**%
mu_2 %**% t(mu_2) %**% t(Z_at) %**% Sigma.it %**% Z_at %**%
B.i %**% t(Z_at))+t(Pi) %**% Eij - (1/2)* (((Z_at %**%
B.i %**% H.i %**% Psi_dc %**% Psi_cc.i) %x% (Z_at %**%
Psi_dc %**% Psi_cc.i))-((Z_at %**% B.i %**% H.i %**%
Psi_dc %**% Psi_cc.i) %x% (Z_at %**% B.i %**% t(Z_at) %**%
Sigma.it %**% Z_at %**% Psi_dc %**% Psi_cc.i))) %**%
vec(G.i)+(1/2)*vec(-(Z_at %**% B.i %**% H.i %**% Psi_dc %**%
Psi_cc.i %**% c_t %**% t(c_t) %**% t(Psi_dc %**%
Psi_cc.i) %**% t(Z_at)))+(Z_at %**% B.i %**% H.i %**%
Psi_dc %**% Psi_cc.i %**% c_t %**% t(c_t) %**% t(Psi_dc %**%
Psi_cc.i) %**% t(Z_at) %**% Sigma.it %**% Z_at %**% B.i %**%
t(Z_at))+2*(A_a %**% t(c_t) %**% t(Psi_dc %**% Psi_cc.i) %**%
H.i %**% B.i %**% t(Z_at))-2*(Z_at %**% B.i %**% t(Z_at) %**%
Sigma.it %**% A_a %**% t(c_t) %**% t(Psi_dc %**%
Psi_cc.i) %**% H.i %**% B.i %**%t(Z_at))) + (1/c(P))*
(S_sigma.int1 - (15/72)*vec(Z_at %**% B.i %**% H.i %**%
(Psi_dc %**% Psi_cc.i) %**% G.i %**% Ki %**% t(Ki) %**%
G.i %**% t(Psi_dc %**% Psi_cc.i) %**% (t(Z_at)-t(Z_at) %**%
Sigma.it %**% Z_at %**% B.i %**% t(Z_at)))) )
Q_sigma<- (diag(2) %x% commutation.matrix(ni, 2) %x% diag(ni)) %**%
(diag(2) %x% diag(2) %x% vec(diag(ni)))
S_sigma.int3<- t((-Sigma.i %x% Sigma.i) %**%
duplication.matrix(n=ncol(Z_at))) %**%
((n_a/2)*vec(Sigma) )

```

```

S_sigma<- S_sigma.int3 - t(duplication.matrix(n=2)) %*% t(Q_sigma) %*%
          (t(Sigma.it) %x% Sigma.it) %*% S_sigma.int2

for(l in 1:length(S_b)){
  S_b[l]<-ifelse(is.na(S_b[l])==T,0,S_b[l])
}
for(l in 1:length(S_g)){
  S_g[l]<-ifelse(is.na(S_g[l])==T,0,S_g[l])
}
for(l in 1:length(S_mu)){
  S_mu[l]<-ifelse(is.na(S_mu[l])==T,0,S_mu[l])
}

S_0b <- S_b %*% t(S_b)
S_0g <- S_g %*% t(S_g)
S_0mu <- S_mu %*% t(S_mu)

S_0b[is.infinite(S_0b)] <- 0
S_0g[is.infinite(S_0g)] <- 0
S_0mu[is.infinite(S_0mu)] <- 0

FI<- list(S_0b=S_0b,S_0g=S_0g,S_0mu=S_0mu)
return(FI)
}

lklhd.ram<-function(id,datafile,est,X.var=X.var,Z.var=Z.var,
  X_a.var=X_a.var,Z_a.var=Z_a.var,U.var=U.var,V1=V1,V2=V2,
  sim=10000){
  #Estimation of random effects
  initial<-est
  Beta=initial$Beta
  Gamma=initial$Gamma
  Gamma<-matrix(Gamma,ncol=1)
  Psi_cc=initial$Psi_cc
  Psi_dd=initial$Psi_dd
  corr<-initial$corr
  Var.V1<-initial$Var.V1
  Var.V2<-initial$Var.V2
  Psi_dc=initial$Psi_dc
  m=initial$m
  mu_1=initial$mu_1

  p<-length(Beta)
  q<-ifelse(length(Psi_cc)>1,dim(Psi_cc)[1],length(Psi_cc))
  p_a<-length(Gamma)
  q_a<-ifelse(length(Psi_dd)>1,dim(Psi_dd)[1],length(Psi_dd))
  Sigma<-matrix(c(Var.V1,rep(corr*sqrt(Var.V1)*sqrt(Var.V2),2),Var.V2),

```

```

        nrow = 2, ncol = 2)

#Data
base<-datafile[datafile$patch==id,]
base$int <- rep(1,nrow(base))
U<-base[,U.var]
X.var<-c("int",X.var)
X<-base[,X.var]
X<-as.matrix(X)
Z.var<-c("int",Z.var)
Z<-base[,Z.var]
Z<-as.matrix(Z)
ni<-nrow(Z)

X_a.var<-c("int",X_a.var)
X_a<-base[,X_a.var]
X_a<-as.matrix(X_a)
X_a<-as.matrix(U * X_a)

Z_a.var<-c("int",Z_a.var)
Z_a<-base[,Z_a.var]
Z_a<-as.matrix(Z_a)
Z_a<-as.matrix(U * Z_a)
n_a<- sum(Z_a[,1])
V_a<-c(base[,V1],base[,V2])

#Calculations
mu_2<- -(m/(1-m))*mu_1
Psi_cc.i<-solve(Psi_cc)
Psi_cd<-t(Psi_dc)
H<-Psi_dd-Psi_dc %*% Psi_cc.i %*% Psi_cd
H.i<-ginv(H)
X_at<-diag(2) %x% as.matrix(X_a)
Z_at<-diag(2) %x% as.matrix(Z_a)
Sigma.i<-ginv(Sigma)
Sigma.t<-Sigma %x% diag(nrow(X_a))
Sigma.it<-Sigma.i %x% diag(nrow(X_a))
Sigma.vch<-vech(Sigma)

Psi_cc.vch<-vech(Psi_cc)
Psi_ddcci<-Psi_dc %*% solve(Psi_cc) #Check this in the future (when
#more random effects, if still a vector)
H.vch<-vech(H)

A_a<-matrix(V_a-X_at %*% Gamma)

tempcheck<-ifelse(sum(V_a==0)==length(V_a),0,1) #if sum is zero,

```

```

#the all observations are zero
if(tempcheck==1){
  B<-t(Z_at) %*% Sigma.it %*% Z_at + H.i
  B.i<-ginv(B)
} else {
  B<- H.i
  B.i<-H
}

E.T<-t(A_a) %*% Sigma.it %*% Z_at %*% B.i %*% H.i %*% Psi_dc %*%
Psi_cc.i
D<- Psi_cc.i %*% Psi_cd %*% H.i %*% B.i %*% t(Z_at) %*% Sigma.it %*%
Z_at %*% Psi_dc %*% Psi_cc.i
g_1.T<-E.T+(t(mu_1) %*% H.i %*% B.i %*% t(Z_at) %*% Sigma.it %*%
Z_at %*% Psi_dc %*% Psi_cc.i)
g_2.T<-E.T+(t(mu_2) %*% H.i %*% B.i %*% t(Z_at) %*% Sigma.it %*%
Z_at %*% Psi_dc %*% Psi_cc.i)

c<-as.matrix(rep(0,length(Z.var)),ncol=1)
other<-list(U=U, X=X, Z=Z, Beta=Beta, Psi_cc.i = Psi_cc.i, D=D,
g.To=E.T)
c_t<- laplace2(lap.for.c,c,other)$mode

#Here I had to truncate c to get better predictions but I will have to
# check later
c_t <- ifelse(c_t>60,60,c_t)
eta<- (X %*% Beta) + (Z %*% c_t)
propi<-exp(eta)/(1+exp(eta))
propi<-apply(propi, 1, p.inf)
eta<- log(propi/(1-propi))
W_t<- diag(c(propi),ncol=length(propi)) %*%
diag(c(1-propi),ncol=length(propi))
G<-(t(Z) %*% W_t %*% Z ) + (Psi_cc.i+D)
G.i.ram<-solve(G)

random<-matrix(data=NA,nrow=sim,ncol=15)
colnames(random)<-c("c","cc^t","g","q","w","cw","cc^tw","d1.1","d1.2",
"d2.1","d2.2","d1.1w","d1.2w","d2.1w","d2.2w")
#random[,1]<-rmvt(sim, sigma = G.i.ram, df = 4, delta = c_t,type =
c("shifted")) #With more RE in c, the multivariate f
#unction must be used
random[,1]<-rep(101,sim)
for(e in 1:sim){
  while (random[e,1]>100) {
    random[e,1]<-rt(1, df=4, ncp=c_t)
  }
  #random[e,1]<-rt(1, df=4, ncp=c_t)
}

```

```

random[e,2]<-random[e,1]*random[e,1]
random[e,8:9]<- (mu_1 + (Psi_dc %** Psi_cc.i * random[e,1])) +
  (B.i %** ((t(Z_at) %** Sigma.it %** A_a) -
  (t(Z_at) %** Sigma.it %** Z_at %** Psi_dc %**
  Psi_cc.i * random[e,1])) )
random[e,10:11]<- (mu_2 + (Psi_dc %** Psi_cc.i * random[e,1])) +
  (B.i %** ((t(Z_at) %** Sigma.it %** A_a) -
  (t(Z_at) %** Sigma.it %** Z_at %** Psi_dc %**
  Psi_cc.i * random[e,1])) )
}

for(e in 1:sim){
  #random[e,3]<-dmvt(random[e,1],sigma = G.i.ram, df = 4, delta = c_t,
  #type = c("shifted")) #With more RE in c, the
  #multivariate function must be used
random[e,3]<-dt(random[e,1], df=4, ncp=c_t)
random[e,4]<-dmvnorm(random[e,1], mean = rep(0, q), sigma = Psi_cc,
  log = FALSE)

eta<- (X %** Beta) + (Z %** random[e,1])
propi<-exp(eta)/(1+exp(eta))
propi<-apply(propi, 1, p.inf)
eta<- log(propi/(1-propi))
lu<-sum((U * eta) + log(1-propi))

lf1<- m*exp( (t(A_a) %** Sigma.it %** Z_at %** B.i %** H.i %**
mu_1) - (1/2)*(t(mu_1) %** H.i %** B.i %** t(Z_at) %**
Sigma.it %** Z_at %** mu_1))* exp((E.T %** random[e,1]) -
(t(mu_1) %** H.i %** B.i %** t(Z_at) %** Sigma.it %** Z_at %**
Psi_dc %** Psi_cc.i %** random[e,1]) - (1/2)*
(t(random[e,1]) %** D %** random[e,1]) )

lf2<- (1-m)*exp( (t(A_a) %** Sigma.it %** Z_at %** B.i %** H.i %**
mu_2) - (1/2)*(t(mu_2) %** H.i %** B.i %** t(Z_at) %**
Sigma.it %** Z_at %** mu_2))*exp((E.T %** random[e,1]) -
(t(mu_2) %** H.i %** B.i %** t(Z_at) %** Sigma.it %** Z_at %**
Psi_dc %** Psi_cc.i %** random[e,1]) - (1/2)*
(t(random[e,1])%** D %** random[e,1]) )

lf<- -(n_a+(q/2))*log(2*pi) - (n_a/2)*log(det(Sigma)) -(1/2)*
log(det(H)) + (1/2)*log(det(B.i)) - ((1/2)* t(A_a) %**
(Sigma.it - Sigma.it %** Z_at %** B.i %** t(Z_at) %**
Sigma.it) %** A_a) + log(lf1 + lf2)
qu<- exp(lu) * exp(lf)

random[e,4]<- random[e,4]*qu
random[e,4]<-ifelse(is.finite(random[e,4]),random[e,4],0)

```

```

}

if (sum(random[,4])!=0) {
  minn<-max(min(random[random[,4]!=0,4]),1e-10)
  for(v in 1:sim){
    random[v,4]<-ifelse(random[v,4]<1e-10,minn,random[v,4])
  }
} else {
  for(v in 1:sim){
    random[v,4]<-dmvnorm(random[v,1], mean = rep(0, q), sigma =
      Psi_cc, log = FALSE)
    random[v,4]<-random[v,4]*0.01
  }
  minn<-max(min(random[random[,4]!=0,4]),1e-10)
  for(v in 1:sim){
    random[v,4]<-ifelse(random[v,4]<1e-10,minn,random[v,4])
  }
}

random[,5]<-random[,4]/random[,3]

sumsum<-ifelse(is.infinite(sum(random[,5]*random[,5])) &
  sum(random[,5]*random[,5])<0 ,0,1)
while (sumsum<1e-10) {
  random[,4]<-random[,4]*(1e+10)
  random[,5]<-random[,4]/random[,3]
  sumsum<-ifelse(is.infinite(sum(random[,5]*random[,5])) &
    sum(random[,5]*random[,5])<0 ,0,1)
}

sumsum<-ifelse(is.infinite(sum(random[,5]*random[,5])) &
  sum(random[,5]*random[,5])>0 ,0,1)
while (sumsum<1e-10) {
  random[,4]<-random[,4]/(1e+10)
  random[,5]<-random[,4]/random[,3]
  sumsum<-ifelse(is.infinite(sum(random[,5]*random[,5])) &
    sum(random[,5]*random[,5])>0 ,0,1)
  sumsum
}

random.t<-random
random.p<-random

#### estimates without adjustments on weights
random[,6]<-random[,1]*random[,5]
random[,7]<-random[,2]*random[,5]
random[,12]<-random[,8]*random[,5]

```

```

random[,13]<-random[,9]*random[,5]
random[,14]<-random[,10]*random[,5]
random[,15]<-random[,11]*random[,5]

random <- random[complete.cases(random), ]
c.est <- mean(random[,6])/ mean(random[,5])
#c.est <- ifelse(c.est < -100,-100,c.est)
#c.est <- ifelse(c.est > 100,100,c.est)
cc.t.est <- mean(random[,7])/ mean(random[,5])
d1.est <- (m*mean(random[,12])/ mean(random[,5])) +
          ((1-m)*mean(random[,14])/ mean(random[,5]))
d2.est <- (m*mean(random[,13])/ mean(random[,5])) +
          ((1-m)*mean(random[,15])/ mean(random[,5]))

mean.w<-mean(random[,5])
ess1<-((random[,5]/mean.w)-1)^2
ess1<-sum(ess1)
ess1<- sqrt(ess1/nrow(random))
ess2<- (sum(random[,5])^2)/sum(random[,5]*random[,5])
ram.est<-list(c_t=c_t,c.est=c.est,cc.t.est=cc.t.est,d1.est=d1.est,
             d2.est=d2.est,ess1=ess1, ess2=ess2)

##### Truncated weights
def<-sqrt(sim*mean(random.t[,5]))
for(rt in 1:sim){
  random.t[rt,5]<-min(random.t[rt,5],def)
}

sumsum<-ifelse(is.infinite(sum(random.t[,5]*random.t[,5])) &
              sum(random.t[,5]*random.t[,5])<0 ,0,1)
while (sumsum<1e-10) {
  random.t[,4]<-random.t[,4]*(1e+10)
  random.t[,5]<-random.t[,4]/random.t[,3]
  sumsum<-ifelse(is.infinite(sum(random.t[,5]*random.t[,5])) &
                sum(random.t[,5]*random.t[,5])<0 ,0,1)
}

sumsum<-ifelse(is.infinite(sum(random.t[,5]*random.t[,5])) &
              sum(random.t[,5]*random.t[,5])>0 ,0,1)
while (sumsum<1e-10) {
  random.t[,4]<-random.t[,4]/(1e+10)
  random.t[,5]<-random.t[,4]/random.t[,3]
  sumsum<-ifelse(is.infinite(sum(random.t[,5]*random.t[,5])) &
                sum(random.t[,5]*random.t[,5])>0 ,0,1)
  sumsum
}

```

```

random.t[,6]<-random.t[,1]*random.t[,5]
random.t[,7]<-random.t[,2]*random.t[,5]
random.t[,12]<-random.t[,8]*random.t[,5]
random.t[,13]<-random.t[,9]*random.t[,5]
random.t[,14]<-random.t[,10]*random.t[,5]
random.t[,15]<-random.t[,11]*random.t[,5]

random.t <- random.t[complete.cases(random.t), ]
c.est.t <- mean(random.t[,6])/ mean(random.t[,5])
cc.t.est.t <- mean(random.t[,7])/ mean(random.t[,5])
d1.est.t <- (m*mean(random.t[,12])/ mean(random.t[,5])) +
            ((1-m)*mean(random.t[,14])/ mean(random.t[,5]))
d2.est.t <- (m*mean(random.t[,13])/ mean(random.t[,5])) +
            ((1-m)*mean(random.t[,15])/ mean(random.t[,5]))

mean.w<-mean(random.t[,5])
ess1.t<-((random.t[,5]/mean.w)-1)^2
ess1.t<-sum(ess1.t)
ess1.t<- sqrt(ess1.t/nrow(random.t))
ess2.t<-(sum(random.t[,5])^2)/sum(random.t[,5]*random.t[,5])
ram.est.t<-list(c.est.t=c.est.t,cc.t.est.t=cc.t.est.t,d1.est.t=
                d1.est.t,d2.est.t=d2.est.t,ess1.t=ess1.t,ess2.t=ess2.t)

##### Pareto weights
log_ratios <- log(random.p[,5])
r_eff <- relative_eff(random.p[,5], chain_id = rep(1,sim))
psis_result <- psis(log_ratios, r_eff = r_eff)
random.p[,5]<-weights(psis_result, log=FALSE, normalize = FALSE)

sumsum<-ifelse(is.infinite(sum(random.p[,5]*random.p[,5])) &
               sum(random.p[,5]*random.p[,5])<0 ,0,1)
while (sumsum<1e-10) {
  random.p[,4]<-random.p[,4]*(1e+10)
  random.p[,5]<-random.p[,4]/random.p[,3]
  sumsum<-ifelse(is.infinite(sum(random.p[,5]*random.p[,5])) &
                 sum(random.p[,5]*random.p[,5])<0 ,0,1)
}

sumsum<-ifelse(is.infinite(sum(random.p[,5]*random.p[,5])) &
               sum(random.p[,5]*random.p[,5])>0 ,0,1)
while (sumsum<1e-10) {
  random.p[,4]<-random.p[,4]/(1e+10)
  random.p[,5]<-random.p[,4]/random.p[,3]
  sumsum<-ifelse(is.infinite(sum(random.p[,5]*random.p[,5])) &
                 sum(random.p[,5]*random.p[,5])>0 ,0,1)
  sumsum
}

```



```

}

random.p[,6]<-random.p[,1]*random.p[,5]
random.p[,7]<-random.p[,2]*random.p[,5]
random.p[,12]<-random.p[,8]*random.p[,5]
random.p[,13]<-random.p[,9]*random.p[,5]
random.p[,14]<-random.p[,10]*random.p[,5]
random.p[,15]<-random.p[,11]*random.p[,5]

random.p <- random.p[complete.cases(random.p), ]
c.est.p <- mean(random.p[,6])/ mean(random.p[,5])
cc.t.est.p <- mean(random.p[,7])/ mean(random.p[,5])
d1.est.p <- (m*mean(random.p[,12])/ mean(random.p[,5])) +
            ((1-m)*mean(random.p[,14])/ mean(random.p[,5]))
d2.est.p <- (m*mean(random.p[,13])/ mean(random.p[,5])) +
            ((1-m)*mean(random.p[,15])/ mean(random.p[,5]))

mean.w<-mean(random.p[,5])
ess1.p<-((random.p[,5]/mean.w)-1)^2
ess1.p<-sum(ess1.p)
ess1.p<- sqrt(ess1.p/nrow(random))
ess2.p<-(sum(random.p[,5])^2)/sum(random.p[,5]*random.p[,5])

ram.est.p<-list(c.est.p=c.est.p,cc.t.est.p=cc.t.est.p,d1.est.p=
              d1.est.p,d2.est.p=d2.est.p,ess1.p=ess1.p,ess2.p=ess2.p)
ram.est.f<-list(ram.est=ram.est, ram.est.t=ram.est.t,
              ram.est.p=ram.est.p)
return(ram.est.f)
}

```

B.3 Chapter 4: Simulation assessing model performance

Also available at: <https://github.com/vivifj03/TumorModel/blob/master/Code%20chapter4%20-%20simulation.R>

```

# Simulation -----
library(MASS)

## Simulation base on spherical data
set.seed(124)
for (n in c(300,500,1000)) {
  for (ni in c(3,4,5)) {
    for (rp in 1:1000) {

      # Dose
      Dose<-rlnorm(n, meanlog = 0, sdlog =1)

```

```

Dose<-Dose*(60/max(Dose))
#plot(density(Dose))
Dose <- rep(Dose,times=ni)

id <- rep(1:n,times=ni)
t <- c(3,6,12,18,24)/12
t <- t[1:ni]
time <- rep(t, each=n)
data <- as.data.frame(cbind(id,Dose,time))
data$time2 <- (data$time)^2
data$int <- 1
data <- data[,c("id","int","Dose","time","time2")]
#dataU <- data[,c("id","int","Dose","time")]
dataU<-data

# Parameters
m <- 0.6
mu_1 <- c(-0.1,0.1)
mu_2 <- -(m/(1-m))*mu_1
Psi_diag<-diag(sqrt(c(1,0.5,1)))
R <- matrix(c(1,0.1,0.2,0.1,1,0.1,0.2,0.1,1),nrow = 3)
Psi<-Psi_diag %*% R %*% Psi_diag
Psi_cc <- as.matrix(Psi[1,1])
Psi_dd <- as.matrix(Psi[2:3,2:3])
Psi_dc <- as.matrix(Psi[2:3,1])
Psi_cd <- t(Psi_dc)
#is.positive.definite(Psi_cc)
#is.positive.definite(Psi_dd)
#is.positive.definite(Psi)
mu1 <- matrix(c(0,mu_1),ncol = 1)
mu2 <- matrix(c(0,mu_2),ncol = 1)

corr <- 0.2
Var.V1 <- 1
Var.V2 <- 0.5
Sigma <- matrix(c(Var.V1,rep(corr*sqrt(Var.V1)*sqrt(Var.V2),2),
                    Var.V2),nrow = 2,ncol = 2)
Beta <- as.matrix(c(-10,1.5,-2,-3),ncol=1)
Gamma <- as.matrix(c(-7,1,0.5,-1,
                    -3,2,0.5,-1),ncol=1)
b <- (m*mvrnorm(n = n, mu1, Psi))+((1-m)*mvrnorm(n = n, mu2,
                    Psi))
d <- data.frame(b[,2:3],i=rep(1:ni,ea=n))
d <- d[,1:2]
c <- data.frame(b[,1],i=rep(1:ni,ea=n))
c <- c[,1]
e <- mvrnorm(n=ni*n,c(0,0),Sigma)

```

```

X <- as.matrix(dataU[,-1])
Z <- as.matrix(dataU[,c("int")])
Zc <- (Z * c)
logit_U <- X %*% Beta + Zc
pU <- exp(logit_U)/(1+exp(logit_U))
U <-rbinom(length(pU),1,pU)

ps<-prop.table(table(U))[2]

Zd1 <- Z * d[,1]
Zd2 <- Z * d[,2]
X <- as.matrix(data[,-1])
Z <- as.matrix(data[,c("int")])
V1 <- X %*% Gamma[1:4,] + Zd1 + e[,1]
V2 <- X %*% Gamma[5:8,] + Zd2 + e[,2]

data <- cbind(data,pU,U,V1,V2)
data$V1u <- ifelse(data$U==0,0,data$V1)
data$V2u <- ifelse(data$U==0,0,data$V2)
data$nonZeroHD <- data$U

sim.val<-c(Beta=Beta,Gamma=Gamma, Psi_cc=Psi_cc,Psi_dd=Psi_dd,
          Var.V1=Var.V1,Var.V2=Var.V2,corr=corr,
          Psi_dc=Psi_dc,m=m,mu_1=mu_1)

#plot(data$Dose,data$V1u)
#plot(data$Dose,data$V2u)
#boxplot(Dose~nonZeroHD, data=data)

#####
rm(list=setdiff(ls(), c("data","maxit","n","ni","rp","ps",
                        "sim.val")))

dataset<-data[,-2]

rm(data)
names(dataset)<-c("patch","Dose","time","time2","pU","U","V1",
                "V2","V1u","V2u","nonZeroHD")
filename<-paste("s1/",n,"/",ni,"/data/","ds1","_",n,"_",ni,"_",
                rp,".RData",sep = "")
save.image(file = filename)
}
}
}

```

```

# Estimates for each data set -----
# Packages
library(lme4)
library(nlme)
library(LearnBayes)
library(plyr)
library(pracma)
library(ks)
library(matrixcalc)
library(MASS)
library(mvtnorm)
library(loo)
library(scorer)

n<- NUMBER #300, 500, 1000
rp<-REP #1-1000
prop<-PROP # 3,4,5

filename<-paste("s1/",n,"/",ni,"/data/","ds1","_",n,"_",prop,"_",
               rp,".RData",sep = "")
load(filename)

total.inicio<-Sys.time()
##### Mixed effects logistic regression - gets initial values
#This models is using time in years
data<-dataset
inicio.naive<-Sys.time()
md1 <- glmer(nonZeroHD ~ Dose + time + time2 + (1 | patch),
            data = data, family = binomial,
            control = glmerControl(optimizer = "bobyqa",
            optCtrl=list(maxfun=2e8)),
            nAGQ = 10)

#summary(md1)

temp<-data
temp$V1u<- ifelse(temp$V1u==0,NA,temp$V1u)
temp$V2u<- ifelse(temp$V2u==0,NA,temp$V2u)
temp<- temp[is.na(temp$V1u)==F | is.na(temp$V2u)==F,]
md2d2 <- lmer(V1u ~ Dose + time + time2 + (1| patch),
            data = temp, control = lmerControl(optCtrl=
            list(maxfun=2e10)))
md2d2.r<-ranef(md2d2)

md2h2 <- lmer(V2u ~ Dose + time + time2 + (1| patch),

```

```

                data = temp, control = lmerControl(optCtrl=
                list(maxfun=2e10))
md2h2.r<-ranef(md2h2)

# Predictions in naive model =====
coefp <- md1@beta

temp2<-unlist(unique(temp$patch))
temp2<-cbind(temp2,md2d2.r$patch[,1])
temp2<-cbind(temp2,md2h2.r$patch[,1])
temp2<-as.data.frame(temp2)
colnames(temp2)<-c("patch","d1","d2")
temp3<-setdiff(seq(1:n),temp2$patch)
temp3<-as.data.frame(temp3)
colnames(temp3)<-c("patch")
temp3$d1 <- mean(temp2$d1)
temp3$d2 <- mean(temp2$d2)
temp2<-rbind(temp2,temp3)
temp2<-temp2[order(temp2$patch),]
rm(temp3)

data<-merge(data,temp2,by="patch", all.x = T)
coefd <- md2d2@beta
coefh <- md2h2@beta

data$naive.p1 <- predict(md1, type = "response")
data$naive.d<- coefd[1] + coefd[2]*data$Dose + coefd[3]*data$time +
                coefd[4]*data$time2 + data$d1
data$naive.h<- coefh[1] + coefh[2]*data$Dose + coefh[3]*data$time +
                coefh[4]*data$time2 + data$d2
data$naive.dr<- data$V1u - data$naive.d
data$naive.hr<- data$V2u - data$naive.h
data$V1u.e<-ifelse(data$V1u==0,0,exp(data$V1u))
data$V2u.e<-ifelse(data$V2u==0,0,exp(data$V2u))

data$naive.de<- data$naive.p1 * exp(data$naive.d) *
                mean(exp(md2d2@resp$wtres))
data$naive.he<- data$naive.p1 * exp(data$naive.h) *
                mean(exp(md2h2@resp$wtres))

data$naive.res.de<- data$V1u.e - data$naive.de
data$naive.res.he<- data$V2u.e - data$naive.he

naive.m_Vu.V1<-mean(data$V1u.e)
naive.TSS.V1<-sum((data$V1u.e-naive.m_Vu.V1)^2)
naive.sse.V.V1<-sum((data$V1u.e-data$naive.de)^2)

```

```

naive.ssr.V.V1<-sum((data$naive.de-naive.m_Vu.V1)^2)
naive.tss.V1<-naive.sse.V.V1+naive.ssr.V.V1
naive.corr2.V1<-cor(data$V1u.e,data$naive.de)^2
naive.R2.V1<-1-(var(data$naive.res.de)/var(data$V1u.e))
naive.mse.V1<-mean_squared_error(data$V1u.e,data$naive.de)
naive.mae.V1<-mean_absolute_error(data$V1u.e,data$naive.de)
naive.mase.V1<-mean_absolute_scaled_error(data$V1u.e,data$naive.de)
naive.rmse.V1<-rmse(data$V1u.e,data$naive.de)
naive.tvs.V1<-total_variance_score(data$V1u.e,data$naive.de)
naive.uvs.V1<-unexplained_variance_score(data$V1u.e,data$naive.de)
naive.evs.V1<-explained_variance_score(data$V1u.e,data$naive.de)
      #Best possible score is 1.0

naive.m_Vu.V2<-mean(data$V2u.e)
naive.TSS.V2<-sum((data$V2u.e-naive.m_Vu.V2)^2)
naive.sse.V.V2<-sum((data$V2u.e-data$naive.he)^2)
naive.ssr.V.V2<-sum((data$naive.he-naive.m_Vu.V2)^2)
naive.tss.V2<-naive.sse.V.V2+naive.ssr.V.V2
naive.corr2.V2<-cor(data$V2u.e,data$naive.he)^2
naive.R2.V2<-1-(var(data$naive.res.he)/var(data$V2u.e))
naive.mse.V2<-mean_squared_error(data$V2u.e,data$naive.he)
naive.mae.V2<-mean_absolute_error(data$V2u.e,data$naive.he)
naive.mase.V2<-mean_absolute_scaled_error(data$V2u.e,data$naive.he)
naive.rmse.V2<-rmse(data$V2u.e,data$naive.he)
naive.tvs.V2<-total_variance_score(data$V2u.e,data$naive.he)
naive.uvs.V2<-unexplained_variance_score(data$V2u.e,data$naive.he)
naive.evs.V2<-explained_variance_score(data$V2u.e,data$naive.he)

end.naive<-Sys.time()
end.naive-inicio.naive

# Likelihood (tpmemmz1) =====

#Initial values
#Occurrence variable
Beta<- as.matrix(md1@beta,ncol=1)
if (((md1@theta)^2)<0.1) {
  #In case the initial value of Psi_cc was zero then put a small value
  Psi_cc<-matrix(0.1,1,1)
} else {
  Psi_cc<-matrix(md1@theta)^2
}

#Intensity variable
Gamma<-c(md2d2@beta,md2h2@beta)
Psi_dd<-matrix(c(as.numeric(VarCorr(md2d)[1]),
                 sqrt(as.numeric(VarCorr(md2d)[1])))*

```

```

        sqrt(as.numeric(VarCorr(md2h)[1]))*0.1,
        sqrt(as.numeric(VarCorr(md2d)[1]))*
        sqrt(as.numeric(VarCorr(md2h)[1]))*0.1,
        as.numeric(VarCorr(md2h)[1])),
        nrow = length(md2d2@theta)+length(md2h2@theta),
        ncol = length(md2d2@theta)+length(md2h2@theta))
corr<-cor(temp$V1u, temp$V2u)
Var.V1<-attr(VarCorr(md2d2), "sc")^2
Var.V2<-attr(VarCorr(md2h2), "sc")^2

Psi_dc<-matrix(c(sqrt(diag(Psi_dd))%*%sqrt(Psi_cc)*0.2),
               nrow=ncol(Psi_dd),ncol=ncol(Psi_cc)) #Corr between
               #random effects between models
m<-0.55 #set probability for normal mixture distribution for
        #random effects
mu_1<-c(-0.05,0.05) #Set mean first normal dristribution for
        #random effects

par.l<-list(Beta=Beta, Gamma=Gamma, Psi_cc=Psi_cc, Psi_dd=Psi_dd,
            Var.V1=Var.V1, Var.V2=Var.V2, corr=corr,
            Psi_dc=Psi_dc, m=m, mu_1=mu_1)
par<-c(Beta=Beta, Gamma=Gamma, Psi_cc=Psi_cc, Psi_dd=Psi_dd,
       Var.V1=Var.V1, Var.V2=Var.V2, corr=corr,
       Psi_dc=Psi_dc, m=m, mu_1=mu_1)

patch<-data$patch
patch<-unique(patch)

model.data<-list(datafile=data, id.var = patch,
                 X.var=c("Dose", "time", "time2"), Z.var=NULL,
                 X_a.var=c("Dose", "time", "time2"), Z_a.var=NULL,
                 U.var="nonZeroHD", V1="V1u", V2="V2u")

# Model =====

#Functions
source("tpmemmz1.R") #code in Chapter 3

# Deviance with the sim values
sim<-par.l
sim$Beta<-as.matrix(sim.val[1:4], ncol=1)
sim$Gamma<-as.matrix(sim.val[5:12], ncol=1)
sim$Psi_cc<-as.matrix(sim.val[13], ncol=1)
sim$Psi_dd<-matrix(c(sim.val[14:17]), nrow=2, ncol=2)
sim$Var.V1<-sim.val[18]

```

```

sim$Var.V2<-sim.val[19]
sim$corr<-sim.val[20]
sim$Psi_dc<-matrix(c(sim.val[21:22]),nrow=2,ncol=1)
sim$m<-sim.val[23]
sim$mu_1<-matrix(c(sim.val[24:25]),nrow=2,ncol=1)
in.dev<-lapply(patch,function(x) lklhd.dev(id=x,
      datafile=dataset,initial=sim,
      X.var=c("Dose","time","time2"),Z.var=NULL,
      X_a.var=c("Dose","time","time2"),Z_a.var=NULL,
      U.var="nonZeroHD",V1="V1u",V2="V2u"))
t.dev<-matrix(0,nrow=length(in.dev), ncol = 1)
for (i in 1:length(in.dev)) {
  t.dev[i,1]<-in.dev[[i]]
}
dev0 <- -2*colSums(t.dev)

### Estimation of fixed effects #####

inicio.coef<-Sys.time()
maxit=1000 #For empirical bayes estimates
step=1
criterion=0
estimate<- vector("list", maxit+1)
estimate[[1]]<-par.l
dif.theta<- vector("list", maxit)
rel.dif.theta<- vector("list", maxit)
dev<-vector("numeric", maxit)
reldev<-vector("numeric", maxit)
eu.dis<- vector("numeric", maxit)

theta<-c(Beta=par.l$Beta,
  Gamma=par.l$Gamma,
  mu=par.l$mu_1,
  Sigma.vech=vech(matrix(c(par.l$Var.V1,rep(par.l$corr*
    sqrt(par.l$Var.V1)*sqrt(par.l$Var.V2),2),
    par.l$Var.V2),nrow = 2,ncol = 2)),
  Psi_cc.vech=vech(par.l$Psi_cc),
  Psi_dc.i=par.l$Psi_dc %*% solve(par.l$Psi_cc),
  H.vech=vech(par.l$Psi_dd-(par.l$Psi_dc %*%
    solve(par.l$Psi_cc) %*% t(par.l$Psi_dc)) ))
in.dev<-lapply(patch,function(x) lklhd.dev(id=x,datafile=dataset,
  initial=par.l,
  X.var=c("Dose","time","time2"),Z.var=NULL,
  X_a.var=c("Dose","time","time2"),Z_a.var=NULL,
  U.var="nonZeroHD",V1="V1u",V2="V2u"))
t.dev<-matrix(0,nrow=length(in.dev), ncol = 1)
for (i in 1:length(in.dev)) {

```



```

    t.dev[i,1]<-in.dev[[i]]
}
dev[[1]]<- -2*colSums(t.dev)
print(dev[[1]])

while (criterion<=0) {
  print(step)
  inicio2<-Sys.time()

  par.l0<-par.l
  theta<-c(Beta=par.l$Beta,
           Gamma=par.l$Gamma,
           mu=par.l$mu_1,
           Sigma.vech=vech(matrix(c(par.l$Var.V1,rep(par.l$corr*
           sqrt(par.l$Var.V1)*sqrt(par.l$Var.V2),2),par.l$Var.V2),
           nrow = 2,ncol = 2)),
           Psi_cc.vech=vech(par.l$Psi_cc),
           Psi_dc.i=par.l$Psi_dc %*% solve(par.l$Psi_cc),
           H.vech=vech(par.l$Psi_dd-(par.l$Psi_dc %*%
           solve(par.l$Psi_cc) %*% t(par.l$Psi_dc)) ))

  theta0<-theta

  t<-lapply(patch,function(x) lklhd.em(id=x,datafile=dataset,
           initial=par.l,
           X.var=c("Dose","time","time2"),Z.var=NULL,
           X_a.var=c("Dose","time","time2"),Z_a.var=NULL,
           U.var="nonZeroHD",V1="V1u",V2="V2u"))

  print("Done with lklhd.em")
  subm<-matrix(0,nrow=length(t), ncol = 1)
  for (i in 1:length(t)) {
    subm[i,1]<-t[[i]]$m
  }
  score<-matrix(0,nrow=length(t), ncol = length(t[[1]]$scorei))
  for (i in 1:length(t)) {
    score[i,]<-t[[i]]$scorei
  }

  trunk<- c(18:23)
  for (i in trunk) {
    cuts<-0+3*sd(score[,i])
    for (j in 1:nrow(score)) {
      score[j,i]<-ifelse(abs(score[j,i])>cuts,0,score[j,i])
    }
  }

  score<-colSums(score,na.rm = T)
  theta<-theta+score

```

```

#Check values of Sigma, Psi_cc
while (theta[15]<=0)
{
  test0<-theta0[15]
  #test1<-theta[15]
  test1<-0.001
  theta[15]<-(test0+test1)/2
}
while (theta[17]<=0)
{
  test0<-theta0[17]
  #test1<-theta[17]
  test1<-0.001
  theta[17]<-(test0+test1)/2
}
corr.test<-theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
while (corr.test<= (-0.95)) {
  corr.test0<- theta0[16]/(sqrt(theta0[15])*sqrt(theta0[17]))
  #corr.test1<- theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
  corr.test1<- -0.90
  corr.new<- (corr.test0+corr.test1)/2
  theta[16]<-corr.new*sqrt(theta[15])*sqrt(theta[17])
  corr.test<-theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
}
while (corr.test >= 0.95) {
  corr.test0<- theta0[16]/(sqrt(theta0[15])*sqrt(theta0[17]))
  #corr.test1<- theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
  corr.test1<- 0.90
  corr.new<- (corr.test0+corr.test1)/2
  theta[16]<-corr.new*sqrt(theta[15])*sqrt(theta[17])
  corr.test<-theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
}
while (theta[18]<=0)
{
  test0<-theta0[18]
  #test1<-theta[18]
  test1<-0.001
  theta[18]<-(test0+test1)/2
}

#New estimates
m<-mean(subm)
m<-ifelse(m<0.5,(1-m),m)
while (m>=0.95)
{
  m0<-par.l$m
}

```

```

    m<-(0.94+m0)/2
}
Beta=matrix(theta[1:4],ncol=1)
Gamma=matrix(theta[5:12],ncol=1)
mu_1=matrix(theta[13:14],ncol=1)
Var.V1=theta[15]
Var.V2=theta[17]
corr=theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
Psi_cc=matrix(theta[18],ncol=1)
Psi_dc.i=matrix(theta[19:20],ncol=1)
Psi_dc=Psi_dc.i%%Psi_cc
Psi_cd=t(Psi_dc)
H<-matrix(c(theta[21],rep(theta[22],2),theta[23]),ncol=2)

# Check H is positive define
while (det(H) <= 0 )
{
  theta[21] <- (theta0[21]+theta[21])/2
  theta[22] <- (theta0[22]+theta[22])/2
  theta[23] <- (theta0[23]+theta[23])/2
  H<-matrix(c(theta[21],rep(theta[22],2),theta[23]),ncol=2)
}

while (theta[21]<=0)
{
  test0<-theta0[21]
  test1<-0.001
  theta[21]<-(test0+test1)/2
}

while (theta[23]<=0)
{
  test0<-theta0[23]
  test1<-0.001
  theta[23]<-(test0+test1)/2
}
H<-matrix(c(theta[21],theta[22],theta[22],theta[23]),ncol=2)

while (det(H) <= 0 )
{
  theta[21] <- (theta0[21]+theta[21])/2
  theta[22] <- (theta0[22]+theta[22])/2
  theta[23] <- (theta0[23]+theta[23])/2
  H<-matrix(c(theta[21],rep(theta[22],2),theta[23]),ncol=2)
}
Psi_dd=H+(Psi_dc.i %% Psi_cd)

```

```

par.l<-list(Beta=Beta, Gamma=Gamma, Psi_cc=Psi_cc, Psi_dd=Psi_dd,
           Var.V1=Var.V1, Var.V2=Var.V2, corr=corr,
           Psi_dc=Psi_dc, m=m, mu_1=mu_1)

step=step+1

#Deviance

in.dev<-lapply(patch, function(x) lklhd.dev(id=x, datafile=dataset,
      initial=par.l,
      X.var=c("Dose", "time", "time2"), Z.var=NULL,
      X_a.var=c("Dose", "time", "time2"), Z_a.var=NULL,
      U.var="nonZeroHD", V1="V1u", V2="V2u"))
t.dev<-matrix(0, nrow=length(in.dev), ncol = 1)
for (i in 1:length(in.dev)) {
  t.dev[i,1]<-in.dev[[i]]
}
dev[[step]]<- -2*colSums(t.dev)

print( paste("new deviance", dev[[step]]))
#Step-halving in case deviance is larger than previous iteration
reldev[[step]]<-((dev[[step]] - dev[[step-1]])/(0.1 +
              abs(dev[[step]])) )

loop3<-0
if (reldev[[step]]>=1e-8){
  ii<-1
  while (reldev[[step]] >-1e-8) {
    if (ii>50) break
    ii<-ii+1
    loop3<-ifelse(ii==50, 1, loop3)
    par.l$Beta<-(par.l$Beta+par.l0$Beta)/2
    par.l$Gamma<-(par.l$Gamma+par.l0$Gamma)/2
    par.l$Psi_cc<-(par.l$Psi_cc+par.l0$Psi_cc)/2
    par.l$Psi_dd<-(par.l$Psi_dd+par.l0$Psi_dd)/2
    par.l$Var.V1<-(par.l$Var.V1+par.l0$Var.V1)/2
    par.l$Var.V2<-(par.l$Var.V2+par.l0$Var.V2)/2
    par.l$corr<-(par.l$corr+par.l0$corr)/2
    par.l$Psi_dc<-(par.l$Psi_dc+par.l0$Psi_dc)/2
    par.l$m<-(par.l$m+par.l0$m)/2
    par.l$mu_1<-(par.l$mu_1+par.l0$mu_1)/2

    in.dev<-lapply(patch, function(x) lklhd.dev(id=x,
      datafile=dataset, initial=par.l,
      X.var=c("Dose", "time", "time2"), Z.var=NULL,
      X_a.var=c("Dose", "time", "time2"), Z_a.var=NULL,

```

```

        U.var="nonZeroHD",V1="V1u",V2="V2u"))
t.dev<-matrix(0,nrow=length(in.dev), ncol = 1)
for (i in 1:length(in.dev)) {
  t.dev[i,1]<-in.dev[[i]]
}
dev[[step]]<- -2*colSums(t.dev)
reldev[[step]]<-((dev[[step]] - dev[[step-1]])/(0.1 +
  abs(dev[[step]])) )
print( paste("base deviance",dev[[step-1]], "new deviance",
  dev[[step]]))
}
}

reldev[[step]]<-abs((dev[[step]] - dev[[step-1]])/(0.1 +
  abs(dev[[step]])) )
theta<-c(Beta=par.l$Beta,
  Gamma=par.l$Gamma,
  mu=par.l$mu_1,
  Sigma.vech=vech(matrix(c(par.l$Var.V1,rep(par.l$corr*
  sqrt(par.l$Var.V1)*sqrt(par.l$Var.V2),2),
  par.l$Var.V2),nrow = 2,ncol = 2)),
  Psi_cc.vech=vech(par.l$Psi_cc),
  Psi_dc.i=par.l$Psi_dc %% solve(par.l$Psi_cc),
  H.vech=vech(par.l$Psi_dd-(par.l$Psi_dc %%
  solve(par.l$Psi_cc) %% t(par.l$Psi_dc)) ))

estimate[[step]]<-par.l
eu.dis[step] <- (dist(rbind(theta0,theta), method =
  "euclidean")^2)/length(theta)
dif.theta[[step]] <- (theta0-theta)
rel.dif.theta[[step]] <- (theta-theta0)/theta
absdev<-ifelse(step==2,1,abs(dev[[step]] - dev[[step-1]]))

criterion.step<-ifelse(step<=maxit,0,1)
criterion.distance<-ifelse(max(abs(dif.theta[[step]]))>1e-8,0,1)
criterion.max<-ifelse(max(abs(rel.dif.theta[[step]]))>1e-8,0,1)
criterion.dev<-ifelse(absdev>1e-8,0,1)
criterion.reldev<-ifelse(reldev[[step]]>1e-8,0,1)

criterion<-criterion.step+criterion.distance+criterion.max+
  criterion.dev+criterion.reldev

criterion<-ifelse(loop3==1 & step<5,0,criterion)

print(c(criterion.step=criterion.step,criterion.distance=
  criterion.distance,criterion.max=criterion.max,
  criterion=criterion))

```

```

print(c(eu.dis=eu.dis[step],max.abs.dif=max(abs(dif.theta[[step]])),
      max.rel=max(abs(rel.dif.theta[[step]])),
      dev=dev[step],absdev.dif=absdev,reldev=reldev[step]))

end2<-Sys.time()
timeit<- end2-inicio2
print(theta)
print(timeit)
}

end.coef<-Sys.time()
end.coef-inicio.coef

### Standard errors #####
se<-lapply(patch,function(x) lklhd.var(id=x,datafile=dataset,
      est=par.l,
      X.var=c("Dose","time","time2"),Z.var=NULL,
      X_a.var=c("Dose","time","time2"),Z_a.var=NULL,
      U.var="nonZeroHD",V1="V1u",V2="V2u"))

S_0b.total<-matrix(0,nrow=4, ncol = 4)
S_0g.total<-matrix(0,nrow=8, ncol = 8)
S_0mu.total<-matrix(0,nrow=2, ncol = 2)

for (i in 1:length(se)) {
  S_0b.total<-S_0b.total+se[[i]]$S_0b
  S_0g.total<-S_0g.total+se[[i]]$S_0g
  S_0mu.total<-S_0mu.total+se[[i]]$S_0mu
}
S_0b.total.i<-ginv(S_0b.total)
S_0g.total.i<-ginv(S_0g.total)
S_0mu.total.i<-ginv(S_0mu.total)

par.final<-c(unlist(par.l)) #Coefficients
bias<-par.final-sim.val #Bias
rel.bias<-bias*100/sim.val #Relative Bias
sqrt(diag(S_0b.total.i)) #SE for occurence model
sqrt(diag(S_0g.total.i)) #SE for intensity model
se.final1<-c(sqrt(diag(S_0b.total.i)),sqrt(diag(S_0g.total.i)))
lci1<-par.final[1:12]-(1.96*se.final1)
uci1<-par.final[1:12]+(1.96*se.final1)
coverage1<-rep(0,12)
for(e in 1:12) {
  coverage1[e]<-ifelse(sim.val[e]>lci1[e] & sim.val[e]<uci1[e],
    1,coverage1[e])
}

```

```

### Random effects #####

try(if(criterion.step==1) stop("estimation did not converge"))

iniciop<-Sys.time()
random.est <- lapply(patch,function(x) lklhd.ram(id=x,
          datafile=dataset,est=par.l,
          X.var=c("Dose","time","time2"),Z.var=NULL,
          X_a.var=c("Dose","time","time2"),Z_a.var=NULL,
          U.var="nonZeroHD",V1="V1u",V2="V2u"))

endp<-Sys.time()
endp-iniciop

### Predicted values #####
#Predicted using basic EB RE
ran.est <- as.data.frame(matrix(NA,length(patch),7))
ran.est[,1] <- patch
for(i in 1:length(patch)){
  ran.est[i,2]<-(random.est[[i]]$ram.est$c.est)
  ran.est[i,3]<-(random.est[[i]]$ram.est$cc.t.est)
  ran.est[i,4]<-(random.est[[i]]$ram.est$d1.est)
  ran.est[i,5]<-(random.est[[i]]$ram.est$d2.est)
  ran.est[i,6]<-(random.est[[i]]$ram.est$ess1)
  ran.est[i,7]<-(random.est[[i]]$ram.est$ess2)
}
ran.est <- as.data.frame(ran.est)
colnames(ran.est) <- c("patch","c.est","cc.t.est","d1.est",
          "d2.est","ess","ess2")

dataset<-dataset[,1:11]
dataset$V1u.e<-ifelse(dataset$nonZeroHD==0,0,exp(dataset$V1))
dataset$V2u.e<-ifelse(dataset$nonZeroHD==0,0,exp(dataset$V2))
dataset<-merge(dataset,ran.est,by="patch",all.x = T)

dataset$logp.1 <- par.l$Beta[1] + par.l$Beta[2] * dataset$Dose +
  par.l$Beta[3] * dataset$time + par.l$Beta[4] * dataset$time2 +
  dataset$c.est
dataset$odds.1 <- exp(dataset$logp.1)
dataset$p.1 <- dataset$odds.1 / (1 + dataset$odds.1)

dataset$subpred1 <- par.l$Gamma[1] + par.l$Gamma[2] * dataset$Dose +
  par.l$Gamma[3] * dataset$time + par.l$Gamma[4] * dataset$time2 +
  dataset$d1.est
dataset$subpred2 <- par.l$Gamma[5] + par.l$Gamma[6] * dataset$Dose +
  par.l$Gamma[7] * dataset$time + par.l$Gamma[8] * dataset$time2 +
  dataset$d2.est

```

```

dataset$subres1 <- dataset$V1u-dataset$subpred1
dataset$subres2 <- dataset$V2u-dataset$subpred2

duan1<- mean(exp(dataset[dataset$nonZeroHD==1,"subres1"]))
duan2<- mean(exp(dataset[dataset$nonZeroHD==1,"subres2"]))

dataset$pred1<- dataset$p.1*exp(dataset$subpred1)*duan1
dataset$pred2<- dataset$p.1*exp(dataset$subpred2)*duan2
dataset$residuals1<-(dataset$V1u.e-dataset$pred1)
dataset$residuals2<-(dataset$V2u.e-dataset$pred2)

ram.m_Vu.V1<-mean(dataset$V1u.e)
ram.TSS.V1<-sum((dataset$V1u.e-ram.m_Vu.V1)^2)
ram.sse.V.V1<-sum((dataset$V1u.e-dataset$pred1)^2)
ram.ssr.V.V1<-sum((dataset$pred1-ram.m_Vu.V1)^2)
ram.tss.V1<-ram.sse.V.V1+ram.ssr.V.V1
ram.corr2.V1<-cor(dataset$V1u.e,dataset$pred1)^2
ram.R2.V1<-1-(var(dataset$residuals1)/var(dataset$V1u.e))
ram.mse.V1<-mean_squared_error(dataset$V1u.e,dataset$pred1)
ram.mae.V1<-mean_absolute_error(dataset$V1u.e,dataset$pred1)
ram.mase.V1<-mean_absolute_scaled_error(dataset$V1u.e,dataset$pred1)
ram.rmse.V1<-rmse(dataset$V1u.e,dataset$pred1)
ram.tvs.V1<-total_variance_score(dataset$V1u.e,dataset$pred1)
ram.uvs.V1<-unexplained_variance_score(dataset$V1u.e,dataset$pred1)
ram.evs.V1<-explained_variance_score(dataset$V1u.e,dataset$pred1)

ram.m_Vu.V2<-mean(dataset$V2u.e)
ram.TSS.V2<-sum((dataset$V2u.e-ram.m_Vu.V2)^2)
ram.sse.V.V2<-sum((dataset$V2u.e-dataset$pred2)^2)
ram.ssr.V.V2<-sum((dataset$pred2-ram.m_Vu.V2)^2)
ram.tss.V2<-ram.sse.V.V2+ram.ssr.V.V2
ram.corr2.V2<-cor(dataset$V2u.e,dataset$pred2)^2
#https://wiki.bcs.rochester.edu/HlpLab/StatsCourses?action=
#AttachFile&do=get&target=Groningen11.pdf
ram.R2.V2<-1-(var(dataset$residuals2)/var(dataset$V2u.e))
ram.mse.V2<-mean_squared_error(dataset$V2u.e,dataset$pred2)
ram.mae.V2<-mean_absolute_error(dataset$V2u.e,dataset$pred2)
ram.mase.V2<-mean_absolute_scaled_error(dataset$V2u.e,dataset$pred2)
ram.rmse.V2<-rmse(dataset$V2u.e,dataset$pred2)
ram.tvs.V2<-total_variance_score(dataset$V2u.e,dataset$pred2)
ram.uvs.V2<-unexplained_variance_score(dataset$V2u.e,dataset$pred2)
ram.evs.V2<-explained_variance_score(dataset$V2u.e,dataset$pred2)
#Best possible score is 1.0

#Predicted using truncated EB RE
dataset.t<-dataset[,1:13]
ran.est <- as.data.frame(matrix(NA,length(patch),7))

```



```

ran.est[,1] <- patch
for(i in 1:length(patch)){
  ran.est[i,2] <- (random.est[[i]]$ram.est.t$c.est)
  ran.est[i,3] <- (random.est[[i]]$ram.est.t$cc.t.est)
  ran.est[i,4] <- (random.est[[i]]$ram.est.t$d1.est)
  ran.est[i,5] <- (random.est[[i]]$ram.est.t$d2.est)
  ran.est[i,6] <- (random.est[[i]]$ram.est.t$ess1)
  ran.est[i,7] <- (random.est[[i]]$ram.est.t$ess2)
}
ran.est <- as.data.frame(ran.est)
colnames(ran.est) <- c("patch", "c.est", "cc.t.est", "d1.est", "d2.est",
  "ess", "ess2")
dataset.t <- merge(dataset.t, ran.est, by="patch", all.x = T)
dataset.t$logp.1 <- par.l$Beta[1] + par.l$Beta[2] * dataset.t$Dose +
  par.l$Beta[3] * dataset.t$time + par.l$Beta[4] * dataset.t$time2 +
  dataset.t$c.est
dataset.t$odds.1 <- exp(dataset.t$logp.1)
dataset.t$p.1 <- dataset.t$odds.1 / (1 + dataset.t$odds.1)

dataset.t$subpred1 <- par.l$Gamma[1] + par.l$Gamma[2] * dataset.t$Dose +
  par.l$Gamma[3] * dataset.t$time + par.l$Gamma[4] * dataset.t$time2 +
  dataset.t$d1.est
dataset.t$subpred2 <- par.l$Gamma[5] + par.l$Gamma[6] * dataset.t$Dose +
  par.l$Gamma[7] * dataset.t$time + par.l$Gamma[8] * dataset.t$time2 +
  dataset.t$d2.est

dataset.t$subres1 <- dataset.t$V1u - dataset.t$subpred1
dataset.t$subres2 <- dataset.t$V2u - dataset.t$subpred2

duan1 <- mean(exp(dataset.t[dataset.t$nonZeroHD==1, "subres1"]))
duan2 <- mean(exp(dataset.t[dataset.t$nonZeroHD==1, "subres2"]))

dataset.t$pred1 <- dataset.t$p.1 * exp(dataset.t$subpred1) * duan1
dataset.t$pred2 <- dataset.t$p.1 * exp(dataset.t$subpred2) * duan2
dataset.t$residuals1 <- (dataset.t$V1u.e - dataset.t$pred1)
dataset.t$residuals2 <- (dataset.t$V2u.e - dataset.t$pred2)

ram.t.m_Vu.V1 <- mean(dataset.t$V1u.e)
ram.t.TSS.V1 <- sum((dataset.t$V1u.e - ram.t.m_Vu.V1)^2)
ram.t.sse.V.V1 <- sum((dataset.t$V1u.e - dataset.t$pred1)^2)
ram.t.ssr.V.V1 <- sum((dataset.t$pred1 - ram.t.m_Vu.V1)^2)
ram.t.tss.V1 <- ram.t.sse.V.V1 + ram.t.ssr.V.V1
ram.t.corr2.V1 <- cor(dataset.t$V1u.e, dataset.t$pred1)^2
ram.t.mse.V1 <- mean_squared_error(dataset.t$V1u.e, dataset.t$pred1)
ram.t.mae.V1 <- mean_absolute_error(dataset.t$V1u.e, dataset.t$pred1)
ram.t.mase.V1 <- mean_absolute_scaled_error(dataset.t$V1u.e, dataset.t$pred1)
ram.t.rmse.V1 <- rmse(dataset.t$V1u.e, dataset.t$pred1)

```

```

ram.t.tv.s.V1<-total_variance_score(dataset.t$V1u.e,dataset.t$pred1)
ram.t.uvs.V1<-unexplained_variance_score(dataset.t$V1u.e,dataset.t$pred1)
ram.t.evs.V1<-explained_variance_score(dataset.t$V1u.e,dataset.t$pred1)

ram.t.m_Vu.V2<-mean(dataset.t$V2u.e)
ram.t.TSS.V2<-sum((dataset.t$V2u.e-ram.t.m_Vu.V2)^2)
ram.t.sse.V.V2<-sum((dataset.t$V2u.e-dataset.t$pred2)^2)
ram.t.ssr.V.V2<-sum((dataset.t$pred2-ram.t.m_Vu.V2)^2)
ram.t.tss.V2<-ram.t.sse.V.V2+ram.t.ssr.V.V2
ram.t.corr2.V2<-cor(dataset.t$V2u.e,dataset.t$pred2)^2
ram.t.R2.V2<-1-(var(dataset.t$residuals2)/var(dataset.t$V2u.e))
ram.t.mse.V2<-mean_squared_error(dataset.t$V2u.e,dataset.t$pred2)
ram.t.t.mae.V2<-mean_absolute_error(dataset.t$V2u.e,dataset.t$pred2)
ram.t.mase.V2<-mean_absolute_scaled_error(dataset.t$V2u.e,dataset.t$pred2)
ram.t.rmse.V2<-rmse(dataset.t$V2u.e,dataset.t$pred2)
ram.t.tv.s.V2<-total_variance_score(dataset.t$V2u.e,dataset.t$pred2)
ram.t.uvs.V2<-unexplained_variance_score(dataset.t$V2u.e,dataset.t$pred2)
ram.t.evs.V2<-explained_variance_score(dataset.t$V2u.e,dataset.t$pred2)

#Predicted using pareto EB RE
dataset.p<-dataset[,1:13]
ran.est <- as.data.frame(matrix(NA,length(patch),7))
ran.est[,1] <- patch
for(i in 1:length(patch)){
  ran.est[i,2]<-(random.est[[i]]$ram.est$c.est)
  ran.est[i,3]<-(random.est[[i]]$ram.est$cc.t.est)
  ran.est[i,4]<-(random.est[[i]]$ram.est$d1.est)
  ran.est[i,5]<-(random.est[[i]]$ram.est$d2.est)
  ran.est[i,6]<-(random.est[[i]]$ram.est$ess1)
  ran.est[i,7]<-(random.est[[i]]$ram.est$ess2)
}
ran.est <- as.data.frame(ran.est)
colnames(ran.est) <- c("patch","c.est","cc.t.est","d1.est","d2.est",
"ess","ess2")
dataset.p<-merge(dataset.p,ran.est,by="patch",all.x = T)

dataset.p$logp.1 <- par.l$Beta[1]+ par.l$Beta[2]*dataset.p$Dose +
  par.l$Beta[3] * dataset.p$time + par.l$Beta[4]*dataset.p$time2 +
  dataset.p$c.est
dataset.p$odds.1 <- exp(dataset.p$logp.1)
dataset.p$p.1 <- dataset.p$odds.1 / (1 + dataset.p$odds.1)

dataset.p$subpred1 <- par.l$Gamma[1]+par.l$Gamma[2]*dataset.p$Dose +
  par.l$Gamma[3] * dataset.p$time + par.l$Gamma[4]*dataset.p$time2 +
  dataset.p$d1.est
dataset.p$subpred2 <- par.l$Gamma[5]+ par.l$Gamma[6]*dataset.p$Dose +
  par.l$Gamma[7] * dataset.p$time + par.l$Gamma[8] * dataset.p$time2 +

```

```

dataset.p$d2.est

dataset.p$subres1 <- dataset.p$V1u-dataset.p$subpred1
dataset.p$subres2 <- dataset.p$V2u-dataset.p$subpred2

duan1<- mean(exp(dataset.p[dataset.p$nonZeroHD==1,"subres1"]))
duan2<- mean(exp(dataset.p[dataset.p$nonZeroHD==1,"subres2"]))

dataset.p$pred1<- dataset.p$p.1*exp(dataset.p$subpred1)*duan1
dataset.p$pred2<- dataset.p$p.1*exp(dataset.p$subpred2)*duan2
dataset.p$residuals1<-(dataset.p$V1u.e-dataset.p$pred1)
dataset.p$residuals2<-(dataset.p$V2u.e-dataset.p$pred2)

ram.p.m_Vu.V1<-mean(dataset.p$V1u.e)
ram.p.pSS.V1<-sum((dataset.p$V1u.e-ram.p.m_Vu.V1)^2)
ram.p.sse.V.V1<-sum((dataset.p$V1u.e-dataset.p$pred1)^2)
ram.p.ssr.V.V1<-sum((dataset.p$pred1-ram.p.m_Vu.V1)^2)
ram.p.pss.V1<-ram.p.sse.V.V1+ram.p.ssr.V.V1
ram.p.corr2.V1<-cor(dataset.p$V1u.e,dataset.p$pred1)^2
ram.p.R2.V1<-1-(var(dataset.p$residuals1)/var(dataset.p$V1u.e))
ram.p.mse.V1<-mean_squared_error(dataset.p$V1u.e,dataset.p$pred1)
ram.p.mae.V1<-mean_absolute_error(dataset.p$V1u.e,dataset.p$pred1)
ram.p.mase.V1<-mean_absolute_scaled_error(dataset.p$V1u.e,dataset.p$pred1)
ram.p.rmse.V1<-rmse(dataset.p$V1u.e,dataset.p$pred1)
ram.p.pvs.V1<-total_variance_score(dataset.p$V1u.e,dataset.p$pred1)
ram.p.uvs.V1<-unexplained_variance_score(dataset.p$V1u.e,dataset.p$pred1)
ram.p.evs.V1<-explained_variance_score(dataset.p$V1u.e,dataset.p$pred1)

ram.p.m_Vu.V2<-mean(dataset.p$V2u.e)
ram.p.pSS.V2<-sum((dataset.p$V2u.e-ram.p.m_Vu.V2)^2)
ram.p.sse.V.V2<-sum((dataset.p$V2u.e-dataset.p$pred2)^2)
ram.p.ssr.V.V2<-sum((dataset.p$pred2-ram.p.m_Vu.V2)^2)
ram.p.pss.V2<-ram.p.sse.V.V2+ram.p.ssr.V.V2
ram.p.corr2.V2<-cor(dataset.p$V2u.e,dataset.p$pred2)^2
ram.p.R2.V2<-1-(var(dataset.p$residuals2)/var(dataset.p$V2u.e))
ram.p.mse.V2<-mean_squared_error(dataset.p$V2u.e,dataset.p$pred2)
ram.p.p.mae.V2<-mean_absolute_error(dataset.p$V2u.e,dataset.p$pred2)
ram.p.mase.V2<-mean_absolute_scaled_error(dataset.p$V2u.e,dataset.p$pred2)
ram.p.rmse.V2<-rmse(dataset.p$V2u.e,dataset.p$pred2)
ram.p.pvs.V2<-total_variance_score(dataset.p$V2u.e,dataset.p$pred2)
ram.p.uvs.V2<-unexplained_variance_score(dataset.p$V2u.e,dataset.p$pred2)
ram.p.evs.V2<-explained_variance_score(dataset.p$V2u.e,dataset.p$pred2)

save.image(file=filenamep)
total.end<-Sys.time()
total.end-total.inicio

```

```

# Combine Simulation Results =====
files1<-list.files(path = "s1/300/3/pred/",pattern = "*.RData")
files2<-list.files(path = "s1/300/4/pred/",pattern = "*.RData")
files3<-list.files(path = "s1/300/5/pred/",pattern = "*.RData")
files4<-list.files(path = "s1/500/3/pred/",pattern = "*.RData")
files5<-list.files(path = "s1/500/4/pred/",pattern = "*.RData")
files6<-list.files(path = "s1/500/5/pred/",pattern = "*.RData")
files7<-list.files(path = "s1/1000/3/pred/",pattern = "*.RData")
files8<-list.files(path = "s1/1000/4/pred/",pattern = "*.RData")
files9<-list.files(path = "s1/1000/5/pred/",pattern = "*.RData")
files<-c(files1,files2,files3,files4,files5,files6,
         files7,files8,files9)
file_name<-"syymmdd_combine_sim.RData"

rp<-c(1:1000)
NN<- c(300,500,1000)
prop <- c(3,4,5)

tot<-length(rp)*length(NN)*length(prop)
results<-as.data.frame(matrix(0,nrow =tot,ncol = 280))
line<-1

for (i in 1:3) {
  for (k in 1:1000) {
    for (g in 1:3) {
      results[line,1] <- NN[i]
      results[line,2] <- prop[g]
      results[line,3] <- rp[k]
      results[line,4]<-paste("s1/",NN[i],"/",prop[g],"/pred24c/",
                           "s1","_",NN[i],"_",prop[g],"_",rp[k],
                           ".RData",sep = "")
      results[line,5]<-paste("s1","_",NN[i],"_",prop[g],"_",rp[k],
                           ".RData",sep = "")

      line<-line+1
    }
  }
}

save.image(file=file_name)

(tot2<-nrow(results))

results$rdata<-results[,5] %in% files
resultsold<-results
results<-results[results$rdata==TRUE,]

(tot2<-nrow(results))

```

```

naive <- as.data.frame(matrix(0,nrow =tot2 ,ncol = 57))
naive[,1:5] <- results[,1:5]

V1 <- as.data.frame(matrix(0,nrow =tot2 ,ncol = 19))
V1[,1:5] <- results[,1:5]
V2 <- V1
V1_t <- V1
V2_t <- V1
V1_p <- V1
V2_p <- V1

# Predicted
for (l in 1:tot2) {
  load(file = results[l,4])

  naive[l,6:9] <- coefp
  naive[l,10:13] <- coefd
  naive[l,14:17] <- coefh
  naive[l,18:21] <- sqrt(diag(vcov(md1)))
  naive[l,22:25] <- sqrt(diag(vcov(md2d2)))
  naive[l,26:29] <- sqrt(diag(vcov(md2h2)))
  naive[l,30] <- naive.m_Vu.V1
  naive[l,31] <- naive.TSS.V1
  naive[l,32] <- naive.sse.V.V1
  naive[l,33] <- naive.ssr.V.V1
  naive[l,34] <- naive.tss.V1
  naive[l,35] <- naive.corr2.V1
  naive[l,36] <- naive.R2.V1
  naive[l,37] <- naive.mse.V1
  naive[l,38] <- naive.mae.V1
  naive[l,39] <- naive.mase.V1
  naive[l,40] <- naive.rmse.V1
  naive[l,41] <- naive.tvs.V1
  naive[l,42] <- naive.uvs.V1
  naive[l,43] <- naive.evs.V1
  naive[l,44] <- naive.m_Vu.V2
  naive[l,45] <- naive.TSS.V2
  naive[l,46] <- naive.sse.V.V2
  naive[l,47] <- naive.ssr.V.V2
  naive[l,48] <- naive.tss.V2
  naive[l,49] <- naive.corr2.V2
  naive[l,50] <- naive.R2.V2
  naive[l,51] <- naive.mse.V2
  naive[l,52] <- naive.mae.V2
  naive[l,53] <- naive.mase.V2
  naive[l,54] <- naive.rmse.V2

```

```

naive[1,55] <- naive.tvsv.V2
naive[1,56] <- naive.uvs.V2
naive[1,57] <- naive.evs.V2

results[1,6:30] <- par.final
results[1,31:42] <- se.final1
#results[1,43:54] <- se.final2
results[1,55:79] <- bias
results[1,80:104] <- rel.bias
results[1,105:116] <- coverage1
#results[1,117:128] <- coverage2
initial.val <- unlist(estimate[[1]])
results[1,129:153] <- initial.val
results[1,154:178] <- par.final-initial.val
results[1,179:203] <- (par.final-initial.val)*100/initial.val
coverage3 <- rep(0,12)
for(e in 1:12) {
  coverage3[e] <- ifelse(initial.val[e]>lci1[e] & initial.val[e]<uci1[e],
    1, coverage3[e])
}
results[1,204:215] <- coverage3
results[1,216] <- step
results[1,217] <- criterion
results[1,218] <- criterion.distance
results[1,219] <- criterion.max
results[1,220] <- criterion.step
results[1,221] <- criterion.dev
results[1,222] <- criterion.reldev
results[1,223] <- ii
results[1,224] <- end.coef-inicio.coef
results[1,225] <- ps
results[1,226] <- max(abs(dif.theta[[step]]))
results[1,227] <- max(abs(rel.dif.theta[[step]]))
results[1,228] <- absdev
results[1,229] <- reldev[[step]]
results[1,230:241] <- lci1
results[1,242:253] <- uci1
results[1,254:265] <- lci2
results[1,266:277] <- uci2
results[1,278] <- dev[[step]]
results[1,279] <- dev0
results[1,280] <- loop3

V1[1,6] <- ram.m_Vu.V1
V1[1,7] <- ram.TSS.V1
V1[1,8] <- ram.sse.V.V1
V1[1,9] <- ram.ssr.V.V1

```

```

V1[1,10] <-ram.tss.V1
V1[1,11] <-ram.corr2.V1
V1[1,12] <-ram.R2.V1
V1[1,13] <-ram.mse.V1
V1[1,14] <-ram.mae.V1
V1[1,15] <-ram.mase.V1
V1[1,16] <-ram.rmse.V1
V1[1,17] <-ram.tvsv.V1
V1[1,18] <-ram.uvs.V1
V1[1,19] <-ram.evs.V1

V2[1,6] <-ram.m_Vu.V2
V2[1,7] <-ram.TSS.V2
V2[1,8] <-ram.sse.V.V2
V2[1,9] <-ram.ssr.V.V2
V2[1,10] <-ram.tss.V2
V2[1,11] <-ram.corr2.V2
V2[1,12] <-ram.R2.V2
V2[1,13] <-ram.mse.V2
V2[1,14] <-ram.mae.V2
V2[1,15] <-ram.mase.V2
V2[1,16] <-ram.rmse.V2
V2[1,17] <-ram.tvsv.V2
V2[1,18] <-ram.uvs.V2
V2[1,19] <-ram.evs.V2

V1_t[1,6] <-ram.t.m_Vu.V1
V1_t[1,7] <-ram.t.TSS.V1
V1_t[1,8] <-ram.t.sse.V.V1
V1_t[1,9] <-ram.t.ssr.V.V1
V1_t[1,10] <-ram.t.tss.V1
V1_t[1,11] <-ram.t.corr2.V1
V1_t[1,12] <-ram.t.R2.V1
V1_t[1,13] <-ram.t.mse.V1
V1_t[1,14] <-ram.t.mae.V1
V1_t[1,15] <-ram.t.mase.V1
V1_t[1,16] <-ram.t.rmse.V1
V1_t[1,17] <-ram.t.tvsv.V1
V1_t[1,18] <-ram.t.uvs.V1
V1_t[1,19] <-ram.t.evs.V1

V2_t[1,6] <-ram.t.m_Vu.V2
V2_t[1,7] <-ram.t.TSS.V2
V2_t[1,8] <-ram.t.sse.V.V2
V2_t[1,9] <-ram.t.ssr.V.V2
V2_t[1,10] <-ram.t.tss.V2
V2_t[1,11] <-ram.t.corr2.V2

```

```

V2_t[1,12] <-ram.t.R2.V2
V2_t[1,13] <-ram.t.mse.V2
V2_t[1,14] <-ram.t.t.mae.V2
V2_t[1,15] <-ram.t.mase.V2
V2_t[1,16] <-ram.t.rmse.V2
V2_t[1,17] <-ram.t.tvsv.V2
V2_t[1,18] <-ram.t.uvs.V2
V2_t[1,19] <-ram.t.evs.V2

V1_p[1,6] <-ram.p.m_Vu.V1
V1_p[1,7] <-ram.p.pSS.V1
V1_p[1,8] <-ram.p.sse.V.V1
V1_p[1,9] <-ram.p.ssr.V.V1
V1_p[1,10] <-ram.p.pss.V1
V1_p[1,11] <-ram.p.corr2.V1
V1_p[1,12] <-ram.p.R2.V1
V1_p[1,13] <-ram.p.mse.V1
V1_p[1,14] <-ram.p.mae.V1
V1_p[1,15] <-ram.p.mase.V1
V1_p[1,16] <-ram.p.rmse.V1
V1_p[1,17] <-ram.p.pvs.V1
V1_p[1,18] <-ram.p.uvs.V1
V1_p[1,19] <-ram.p.evs.V1

V2_p[1,6] <-ram.p.m_Vu.V2
V2_p[1,7] <-ram.p.pSS.V2
V2_p[1,8] <-ram.p.sse.V.V2
V2_p[1,9] <-ram.p.ssr.V.V2
V2_p[1,10] <-ram.p.pss.V2
V2_p[1,11] <-ram.p.corr2.V2
V2_p[1,12] <-ram.p.R2.V2
V2_p[1,13] <-ram.p.mse.V2
V2_p[1,14] <-ram.p.p.mae.V2
V2_p[1,15] <-ram.p.mase.V2
V2_p[1,16] <-ram.p.rmse.V2
V2_p[1,17] <-ram.p.pvs.V2
V2_p[1,18] <-ram.p.uvs.V2
V2_p[1,19] <-ram.p.evs.V2
}

colnames(naive) <- c("N","ni","rep","file","file2","naive.beta1",
"naive.beta2","naive.beta3","naive.beta4", "naive.gamma1",
"naive.gamma2","naive.gamma3","naive.gamma4","naive.gamma5",
"naive.gamma6","naive.gamma7","naive.gamma8","naive.se.Beta1",
"naive.se.Beta2","naive.se.Beta3","naive.se.Beta4",
"naive.se.Gamma1","naive.se.Gamma2","naive.se.Gamma3",
"naive.se.Gamma4","naive.se.Gamma5","naive.se.Gamma6",

```



```
"naive.se.Gamma7","naive.se.Gamma8","naive.m_Vu.V1","naive.TSS.V1",
"naive.sse.V.V1","naive.ssr.V.V1","naive.tss.V1","naive.corr2.V1",
"naive.R2.V1","naive.mse.V1","naive.mae.V1","naive.mase.V1",
"naive.rmse.V1","naive.tvs.V1","naive.uvs.V1","naive.evs.V1",
"naive.m_Vu.V2","naive.TSS.V2","naive.sse.V.V2","naive.ssr.V.V2",
"naive.tss.V2","naive.corr2.V2","naive.R2.V2","naive.mse.V2",
"naive.mae.V2","naive.mase.V2","naive.rmse.V2","naive.tvs.V2",
"naive.uvs.V2","naive.evs.V2")
```

```
colnames(resultsold)<-c("N","ni","rep","file","file2",
"beta1","beta2","beta3","beta4","gamma1","gamma2","gamma3",
"gamma4","gamma5","gamma6","gamma7","gamma8","Psi_cc","Psi_dd1",
"Psi_dd2","Psi_dd3","Psi_dd4","Var.V1","Var.V2","corr","Psi_dc1",
"Psi_dc2","m","mu_1_1","mu_1_2","se.Beta1","se.Beta2","se.Beta3",
"se.Beta4","se.Gamma1","se.Gamma2","se.Gamma3","se.Gamma4",
"se.Gamma5","se.Gamma6","se.Gamma7","se.Gamma8","se2.Beta1",
"se2.Beta2","se2.Beta3","se2.Beta4","se2.Gamma1","se2.Gamma2",
"se2.Gamma3","se2.Gamma4","se2.Gamma5","se2.Gamma6","se2.Gamma7",
"se2.Gamma8","b.Beta1","b.Beta2","b.Beta3","b.Beta4","b.Gamma1",
"b.Gamma2","b.Gamma3","b.Gamma4","b.Gamma5","b.Gamma6","b.Gamma7",
"b.Gamma8","b.Psi_cc","b.Psi_dd1","b.Psi_dd2","b.Psi_dd3",
"b.Psi_dd4","b.Var.V1","b.Var.V2","b.corr","b.Psi_dc1","b.Psi_dc2",
"b.m","b.mu_11","b.mu_12","rb.Beta1","rb.Beta2","rb.Beta3",
"rb.Beta4","rb.Gamma1","rb.Gamma2","rb.Gamma3","rb.Gamma4",
"rb.Gamma5","rb.Gamma6","rb.Gamma7","rb.Gamma8","rb.Psi_cc",
"rb.Psi_dd1","rb.Psi_dd2","rb.Psi_dd3","rb.Psi_dd4","rb.Var.V1",
"rb.Var.V2","rb.corr","rb.Psi_dc1","rb.Psi_dc2","rb.m","rb.mu_11",
"rb.mu_12","cv.Beta1","cv.Beta2","cv.Beta3","cv.Beta4","cv.Gamma1",
"cv.Gamma2","cv.Gamma3","cv.Gamma4","cv.Gamma5","cv.Gamma6",
"cv.Gamma7","cv.Gamma8","cv2.Beta1","cv2.Beta2","cv2.Beta3",
"cv2.Beta4","cv2.Gamma1","cv2.Gamma2","cv2.Gamma3","cv2.Gamma4",
"cv2.Gamma5","cv2.Gamma6","cv2.Gamma7","cv2.Gamma8","beta1.i",
"beta2.i","beta3.i","beta4.i","gamma1.i","gamma2.i","gamma3.i",
"gamma4.i","gamma5.i","gamma6.i","gamma7.i","gamma8.i","Psi_cc.i",
"Psi_dd1.i","Psi_dd2.i","Psi_dd3.i","Psi_dd4.i","Var.V1.i",
"Var.V2.i","corr.i","Psi_dc1.i","Psi_dc2.i","m.i","mu_1_1.i",
"mu_1_2.i","b.Beta1.i","b.Beta2.i","b.Beta3.i","b.Beta4.i",
"b.Gamma1.i","b.Gamma2.i","b.Gamma3.i","b.Gamma4.i","b.Gamma5.i",
"b.Gamma6.i","b.Gamma7.i","b.Gamma8.i","b.Psi_cc.i","b.Psi_dd1.i",
"b.Psi_dd2.i","b.Psi_dd3.i","b.Psi_dd4.i","b.Var.V1.i",
"b.Var.V2.i","b.corr.i","b.Psi_dc1.i","b.Psi_dc2.i","b.m.i",
"b.mu_11.i","b.mu_12.i","rb.Beta1.i","rb.Beta2.i","rb.Beta3.i",
"rb.Beta4.i","rb.Gamma1.i","rb.Gamma2.i","rb.Gamma3.i",
"rb.Gamma4.i","rb.Gamma5.i","rb.Gamma6.i","rb.Gamma7.i",
"rb.Gamma8.i","rb.Psi_cc.i","rb.Psi_dd1.i","rb.Psi_dd2.i",
"rb.Psi_dd3.i","rb.Psi_dd4.i","rb.Var.V1.i","rb.Var.V2.i",
"rb.corr.i","rb.Psi_dc1.i","rb.Psi_dc2.i","rb.m.i","rb.mu_11.i",
```

```

"rb.mu_12.i", "cv.Beta1.i", "cv.Beta2.i", "cv.Beta3.i", "cv.Beta4.i",
"cv.Gamma1.i", "cv.Gamma2.i", "cv.Gamma3.i", "cv.Gamma4.i",
"cv.Gamma5.i", "cv.Gamma6.i", "cv.Gamma7.i", "cv.Gamma8.i",
"step", "criterion", "criterion.distance", "criterion.max",
"criterion.step", "criterion.dev", "criterion.reldev", "ii",
"time", "ps", "max.abs.dif.theta", "max.abs.rel.dif.theta", "absdev",
"reldev", "lci1.beta1", "lci1.beta2", "lci1.beta3", "lci1.beta4",
"lci1.gamma1", "lci1.gamma2", "lci1.gamma3", "lci1.gamma4",
"lci1.gamma5", "lci1.gamma6", "lci1.gamma7", "lci1.gamma8",
"uci1.beta1", "uci1.beta2", "uci1.beta3", "uci1.beta4",
"uci1.gamma1", "uci1.gamma2", "uci1.gamma3", "uci1.gamma4",
"uci1.gamma5", "uci1.gamma6", "uci1.gamma7", "uci1.gamma8",
"lci2.beta1", "lci2.beta2", "lci2.beta3", "lci2.beta4",
"lci2.gamma1", "lci2.gamma2", "lci2.gamma3", "lci2.gamma4",
"lci2.gamma5", "lci2.gamma6", "lci2.gamma7", "lci2.gamma8",
"uci2.beta1", "uci2.beta2", "uci2.beta3", "uci2.beta4",
"uci2.gamma1", "uci2.gamma2", "uci2.gamma3", "uci2.gamma4",
"uci2.gamma5", "uci2.gamma6", "uci2.gamma7", "uci2.gamma8",
"dev", "dev0", "loop3")
colnames(results) <- colnames(resultsold)
results$conv <- ifelse(results$criterion.distance == 1 |
                      results$criterion.max == 1 |
                      results$criterion.dev == 1 |
                      results$criterion.reldev == 1, 1, 0)
results$step <- results$step - 1
results$time <- results$time / 60

colnames(V1) <- c("N", "ni", "rep", "file", "file2", "m_Vu", "TSS", "sse.V",
                 "ssr.V", "tss", "corr2", "R2", "mse", "mae", "mase",
                 "rmse", "tvs", "uvs", "evs")

colnames(V2) <- colnames(V1)
colnames(V1_t) <- colnames(V1)
colnames(V2_t) <- colnames(V1)
colnames(V1_p) <- colnames(V1)
colnames(V2_p) <- colnames(V1)

save.image(file=file_name)

# Summary =====

# This code uses the compile simulation
# (scenario|sample size|RILD proportion)
# and produce the summary

#Data

```

```

load(file=file_name)

#First, delete objects we do not need
rm(list=setdiff(ls(), c("results", "resultsold", "naive", "V1", "V2",
                        "V1.m", "V2.m", "V1_t", "V2_t", "V1.m_t",
                        "V2.m_t", "V1_p", "V2_p", "V1.m_p", "V2.m_p")))

# We need to trim the outliers.
# We did 1% (0.5% at each end)
# Next lines, find the cutoff for trimming
resultsorig<-results
results.p05<-aggregate(results[,c(1:2,6:30)],
                        by = list(results$N,results$ni),
                        function(x) quantile(x, c(.005), na.rm=T))
results.p995<-aggregate(results[,c(1:2,6:30)],
                        by = list(results$N,results$ni),
                        function(x) quantile(x, c(.995), na.rm=T))
# Next lines are very inefficient and take long time but they
# check if the simulation values are part of the outliers (1%)
results$beta1trim<-0
results$beta2trim<-0
results$beta3trim<-0
results$beta4trim<-0
results$gamma1trim<-0
results$gamma2trim<-0
results$gamma3trim<-0
results$gamma4trim<-0
results$gamma5trim<-0
results$gamma6trim<-0
results$gamma7trim<-0
results$gamma8trim<-0
results$Psi_cctrim<-0
results$Psi_dd1trim<-0
results$Psi_dd2trim<-0
results$Psi_dd3trim<-0
results$Psi_dd4trim<-0
results$Var.V1trim<-0
results$Var.V2trim<-0
results$corrtrim<-0
results$Psi_dc1trim<-0
results$Psi_dc2trim<-0
results$mtrim<-0
results$mu_1_1trim<-0
results$mu_1_2trim<-0

for(j in 1:nrow(results)) {
  N<-results[j,]$N

```

```

ni<-results[j,]$ni
beta1a<-results.p05[results.p05$N==N & results.p05$ni==ni,"beta1"]
beta2a<-results.p05[results.p05$N==N & results.p05$ni==ni,"beta2"]
beta3a<-results.p05[results.p05$N==N & results.p05$ni==ni,"beta3"]
beta4a<-results.p05[results.p05$N==N & results.p05$ni==ni,"beta4"]
gamma1a<-results.p05[results.p05$N==N & results.p05$ni==ni,"gamma1"]
gamma2a<-results.p05[results.p05$N==N & results.p05$ni==ni,"gamma2"]
gamma3a<-results.p05[results.p05$N==N & results.p05$ni==ni,"gamma3"]
gamma4a<-results.p05[results.p05$N==N & results.p05$ni==ni,"gamma4"]
gamma5a<-results.p05[results.p05$N==N & results.p05$ni==ni,"gamma5"]
gamma6a<-results.p05[results.p05$N==N & results.p05$ni==ni,"gamma6"]
gamma7a<-results.p05[results.p05$N==N & results.p05$ni==ni,"gamma7"]
gamma8a<-results.p05[results.p05$N==N & results.p05$ni==ni,"gamma8"]
Psi_cca<-results.p05[results.p05$N==N & results.p05$ni==ni,"Psi_cc"]
Psi_dd1a<-results.p05[results.p05$N==N & results.p05$ni==ni,"Psi_dd1"]
Psi_dd2a<-results.p05[results.p05$N==N & results.p05$ni==ni,"Psi_dd2"]
Psi_dd3a<-results.p05[results.p05$N==N & results.p05$ni==ni,"Psi_dd3"]
Psi_dd4a<-results.p05[results.p05$N==N & results.p05$ni==ni,"Psi_dd4"]
Var.V1a<-results.p05[results.p05$N==N & results.p05$ni==ni,"Var.V1"]
Var.V2a<-results.p05[results.p05$N==N & results.p05$ni==ni,"Var.V2"]
corra<-results.p05[results.p05$N==N & results.p05$ni==ni,"corr"]
Psi_dc1a<-results.p05[results.p05$N==N & results.p05$ni==ni,"Psi_dc1"]
Psi_dc2a<-results.p05[results.p05$N==N & results.p05$ni==ni,"Psi_dc2"]
ma<-results.p05[results.p05$N==N & results.p05$ni==ni,"m"]
mu_1_1a<-results.p05[results.p05$N==N & results.p05$ni==ni,"mu_1_1"]
mu_1_2a<-results.p05[results.p05$N==N & results.p05$ni==ni,"mu_1_2"]

beta1b<-results.p995[results.p995$N==N & results.p995$ni==ni,"beta1"]
beta2b<-results.p995[results.p995$N==N & results.p995$ni==ni,"beta2"]
beta3b<-results.p995[results.p995$N==N & results.p995$ni==ni,"beta3"]
beta4b<-results.p995[results.p995$N==N & results.p995$ni==ni,"beta4"]
gamma1b<-results.p995[results.p995$N==N & results.p995$ni==ni,"gamma1"]
gamma2b<-results.p995[results.p995$N==N & results.p995$ni==ni,"gamma2"]
gamma3b<-results.p995[results.p995$N==N & results.p995$ni==ni,"gamma3"]
gamma4b<-results.p995[results.p995$N==N & results.p995$ni==ni,"gamma4"]
gamma5b<-results.p995[results.p995$N==N & results.p995$ni==ni,"gamma5"]
gamma6b<-results.p995[results.p995$N==N & results.p995$ni==ni,"gamma6"]
gamma7b<-results.p995[results.p995$N==N & results.p995$ni==ni,"gamma7"]
gamma8b<-results.p995[results.p995$N==N & results.p995$ni==ni,"gamma8"]
Psi_ccb<-results.p995[results.p995$N==N & results.p995$ni==ni,"Psi_cc"]
Psi_dd1b<-results.p995[results.p995$N==N & results.p995$ni==ni,"Psi_dd1"]
Psi_dd2b<-results.p995[results.p995$N==N & results.p995$ni==ni,"Psi_dd2"]
Psi_dd3b<-results.p995[results.p995$N==N & results.p995$ni==ni,"Psi_dd3"]
Psi_dd4b<-results.p995[results.p995$N==N & results.p995$ni==ni,"Psi_dd4"]
Var.V1b<-results.p995[results.p995$N==N & results.p995$ni==ni,"Var.V1"]
Var.V2b<-results.p995[results.p995$N==N & results.p995$ni==ni,"Var.V2"]
corrb<-results.p995[results.p995$N==N & results.p995$ni==ni,"corr"]

```

```

Psi_dc1b<-results.p995[results.p995$N==N & results.p995$ni==ni,"Psi_dc1"]
Psi_dc2b<-results.p995[results.p995$N==N & results.p995$ni==ni,"Psi_dc2"]
mb<-results.p995[results.p995$N==N & results.p995$ni==ni,"m"]
mu_1_1b<-results.p995[results.p995$N==N & results.p995$ni==ni,"mu_1_1"]
mu_1_2b<-results.p995[results.p995$N==N & results.p995$ni==ni,"mu_1_2"]

results[j,]$beta1trim<-ifelse(results[j,]$beta1<beta1a |
                             results[j,]$beta1>beta1b,1,0)
results[j,]$beta2trim<-ifelse(results[j,]$beta2<beta2a |
                             results[j,]$beta2>beta2b,1,0)
results[j,]$beta3trim<-ifelse(results[j,]$beta3<beta3a |
                             results[j,]$beta3>beta3b,1,0)
results[j,]$beta4trim<-ifelse(results[j,]$beta4<beta4a |
                             results[j,]$beta4>beta4b,1,0)
results[j,]$gamma1trim<-ifelse(results[j,]$gamma1<gamma1a |
                              results[j,]$gamma1>gamma1b,1,0)
results[j,]$gamma2trim<-ifelse(results[j,]$gamma2<gamma2a |
                              results[j,]$gamma2>gamma2b,1,0)
results[j,]$gamma3trim<-ifelse(results[j,]$gamma3<gamma3a |
                              results[j,]$gamma3>gamma3b,1,0)
results[j,]$gamma4trim<-ifelse(results[j,]$gamma4<gamma4a |
                              results[j,]$gamma4>gamma4b,1,0)
results[j,]$gamma5trim<-ifelse(results[j,]$gamma5<gamma5a |
                              results[j,]$gamma5>gamma5b,1,0)
results[j,]$gamma6trim<-ifelse(results[j,]$gamma6<gamma6a |
                              results[j,]$gamma6>gamma6b,1,0)
results[j,]$gamma7trim<-ifelse(results[j,]$gamma7<gamma7a |
                              results[j,]$gamma7>gamma7b,1,0)
results[j,]$gamma8trim<-ifelse(results[j,]$gamma8<gamma8a |
                              results[j,]$gamma8>gamma8b,1,0)
results[j,]$Psi_cctrim<-ifelse(results[j,]$Psi_cc<Psi_cca |
                              results[j,]$Psi_cc>Psi_ccb,1,0)
results[j,]$Psi_dd1trim<-ifelse(results[j,]$Psi_dd1<Psi_dd1a |
                              results[j,]$Psi_dd1>Psi_dd1b,1,0)
results[j,]$Psi_dd2trim<-ifelse(results[j,]$Psi_dd2<Psi_dd2a |
                              results[j,]$Psi_dd2>Psi_dd2b,1,0)
results[j,]$Psi_dd3trim<-ifelse(results[j,]$Psi_dd3<Psi_dd3a |
                              results[j,]$Psi_dd3>Psi_dd3b,1,0)
results[j,]$Psi_dd4trim<-ifelse(results[j,]$Psi_dd4<Psi_dd4a |
                              results[j,]$Psi_dd4>Psi_dd4b,1,0)
results[j,]$Var.V1trim<-ifelse(results[j,]$Var.V1<Var.V1a |
                              results[j,]$Var.V1>Var.V1b,1,0)
results[j,]$Var.V2trim<-ifelse(results[j,]$Var.V2<Var.V2a |
                              results[j,]$Var.V2>Var.V2b,1,0)
results[j,]$corrtrim<-ifelse(results[j,]$corr<corra |
                              results[j,]$corr>corrb,1,0)
results[j,]$Psi_dc1trim<-ifelse(results[j,]$Psi_dc1<Psi_dc1a |

```

```

                                results[j,]$Psi_dc1>Psi_dc1b,1,0)
results[j,]$Psi_dc2trim<-ifelse(results[j,]$Psi_dc2<Psi_dc2a |
                                results[j,]$Psi_dc2>Psi_dc2b,1,0)
results[j,]$mtrim<-ifelse(results[j,]$m<ma | results[j,]$m>mb,1,0)
results[j,]$mu_1_1trim<-ifelse(results[j,]$mu_1_1<mu_1_1a |
                                results[j,]$mu_1_1>mu_1_1b,1,0)
results[j,]$mu_1_2trim<-ifelse(results[j,]$mu_1_2<mu_1_2a |
                                results[j,]$mu_1_2>mu_1_2b,1,0)
}
results$trim<-ifelse(results$beta1trim==1 & results$beta2trim==1 &
                    results$beta3trim==1 & results$beta4trim==1,1,0)
resultstrim<-results[resultstrim==0,] #this data set deletes the
#simulations where the betas are outliers (all of them)

#summary
#Predictions - Estimates convergence
conv.summary<-aggregate(resultstrim$criteria.step,
                        by = list(resultstrim$N,resultstrim$ni), length)
ps.summary<-aggregate(resultstrim$ps,
                      by = list(resultstrim$N,resultstrim$ni), mean)
#Next line creates time for estimation of the coef in min
resultstrim$time.coef<-(resultstrim$end.coef-
                        resultstrim$inicio.coef)/60
#Next line creates time for estimation of the RE in min
resultstrim$time.RE<-(resultstrim$endp-resultstrim$iniop)/60
time.summary<-aggregate(resultstrim$time.coef,
                        by = list(resultstrim$N,resultstrim$ni), mean)
timep.summary<-aggregate(resultstrim$time.RE,
                         by = list(resultstrim$N,resultstrim$ni), mean)
step.summary<-aggregate(resultstrim$step,
                        by = list(resultstrim$N,resultstrim$ni), mean)

mean_result<-aggregate(resultstrim, by = list(resultstrim$N,
                                             resultstrim$ni), function(x) mean(x, na.rm=TRUE))
write.csv(mean_result, "mean_result.csv")
sd_result<-aggregate(resultstrim, by = list(resultstrim$N,
                                             resultstrim$ni), function(x) sd(x, na.rm=TRUE))
write.csv(sd_result, "sd_result.csv")

#Subset other datasets based on resultstrim
resultstrim$reps<-paste(resultstrim$N,"_",resultstrim$ni,"_",
                        resultstrim$rep,sep = "")
V1$reps<-paste(V1$N,"_",V1$ni,"_",V1$rep,sep = "")
V1.trim<-subset(V1, reps %in% resultstrim$reps)
V1_p$reps<-paste(V1_p$N,"_",V1_p$ni,"_",V1_p$rep,sep = "")
V1_p.trim<-subset(V1_p, reps %in% resultstrim$reps)
V1_t$reps<-paste(V1_t$N,"_",V1_t$ni,"_",V1_t$rep,sep = "")

```

```

V1_t.trim<-subset(V1_t, reps %in% resultstrim$reps)

V2$reps<-paste(V2$N,"_",V2$ni,"_",V2$rep,sep = "")
V2.trim<-subset(V2, reps %in% resultstrim$reps)
V2_p$reps<-paste(V2_p$N,"_",V2_p$ni,"_",V2_p$rep,sep = "")
V2_p.trim<-subset(V2_p, reps %in% resultstrim$reps)
V2_t$reps<-paste(V2_t$N,"_",V2_t$ni,"_",V2_t$rep,sep = "")
V2_t.trim<-subset(V2_t, reps %in% resultstrim$reps)

t.V1<-aggregate(V1.trim, by = list(V1.trim$N,V1.trim$ni),
                function(x) mean(x, na.rm=TRUE))
t.V1_p<-aggregate(V1_p.trim, by = list(V1_p.trim$N,V1_p.trim$ni),
                  function(x) mean(x, na.rm=TRUE))
t.V1_t<-aggregate(V1_t.trim, by = list(V1_t.trim$N,V1_t.trim$ni),
                  function(x) mean(x, na.rm=TRUE))
t.V2<-aggregate(V2.trim, by = list(V2.trim$N,V2.trim$ni),
                function(x) mean(x, na.rm=TRUE))
t.V2_p<-aggregate(V2_p.trim, by = list(V2_p.trim$N,V2_p.trim$ni),
                  function(x) mean(x, na.rm=TRUE))
t.V2_t<-aggregate(V2_t.trim, by = list(V2_t.trim$N,V2_t.trim$ni),
                  function(x) mean(x, na.rm=TRUE))

s.V1<-aggregate(V1.trim, by = list(V1.trim$N,V1.trim$ni),
                function(x) sd(x, na.rm=TRUE))
s.V1_p<-aggregate(V1_p.trim, by = list(V1_p.trim$N,V1_p.trim$ni),
                  function(x) sd(x, na.rm=TRUE))
s.V1_t<-aggregate(V1_t.trim, by = list(V1_t.trim$N,V1_t.trim$ni),
                  function(x) sd(x, na.rm=TRUE))
s.V2<-aggregate(V2.trim, by = list(V2.trim$N,V2.trim$ni),
                function(x) sd(x, na.rm=TRUE))
s.V2_p<-aggregate(V2_p.trim, by = list(V2_p.trim$N,V2_p.trim$ni),
                  function(x) sd(x, na.rm=TRUE))
s.V2_t<-aggregate(V2_t.trim, by = list(V2_t.trim$N,V2_t.trim$ni),
                  function(x) sd(x, na.rm=TRUE))

#Subset other datasets based on resultstrim
naive$reps<-paste(naive$N,"_",naive$ni,"_",naive$rep,sep = "")
naive.trim<-subset(naive, reps %in% resultstrim$reps)
t.naive<-aggregate(naive.trim, by = list(naive.trim$N,naive.trim$ni),
                  function(x) mean(x, na.rm=TRUE))
s.naive<-aggregate(naive.trim, by = list(naive.trim$N,naive.trim$ni),
                  function(x) sd(x, na.rm=TRUE))

### Plots #####
library(xlsx)
library(ggplot2)
# File Simulations 2020 05 10.xlsx has the organized results

```



```

# of the simulations
pred<-read.xlsx("Simulations_2020_05_10.xlsx", sheetIndex=2, header=TRUE)
coef<-read.xlsx("Simulations_2020_05_10.xlsx", sheetIndex=6, header=TRUE)
bias<-read.xlsx("Simulations_2020_05_10.xlsx", sheetIndex=7, header=TRUE)
se_sim<-read.xlsx("Simulations_2020_05_10.xlsx", sheetIndex=8, header=TRUE)
sd_sim<-read.xlsx("Simulations_2020_05_10.xlsx", sheetIndex=9, header=TRUE)
cov_se<-read.xlsx("Simulations_2020_05_10.xlsx", sheetIndex=10, header=TRUE)

pred<-pred[complete.cases(pred),]
coef<-coef[complete.cases(coef),]
bias<-bias[complete.cases(bias),]
cov_se<-cov_se[complete.cases(cov_se),]
sd_sim<-sd_sim[complete.cases(sd_sim),]
se_sim<-se_sim[complete.cases(se_sim),]

pred<-pred[order(pred$Patches,pred$ni),]
coef<-coef[order(coef$Patches,coef$ni),]
bias<-bias[order(bias$Patches,bias$ni),]
cov_se<-cov_se[order(cov_se$Patches,cov_se$ni),]
sd_sim<-sd_sim[order(sd_sim$Patches,sd_sim$ni),]
se_sim<-se_sim[order(se_sim$Patches,se_sim$ni),]

#Convergence
ggplot(data=pred,aes(x = ni,y = Estimates..Convergence,
                    col=factor(Patches))) +
  geom_jitter(width = 0.005,aes(shape=factor(Patches)),size = 2) +
  scale_shape_manual(name="Number_of\nPatches",values=c(16, 3, 17)) +
  scale_color_manual(name="Number_of\nPatches",
                    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
  scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
  scale_y_continuous(name="Convergence_%",
                    breaks=seq(98,100,0.5),limits = c(98,100)) +

ggplot(data=pred,aes(x = ni,y = Average..time.RE..m.,
                    col=factor(Patches))) +
  geom_jitter(width = 0.005,aes(shape=factor(Patches)),size = 2) +
  scale_shape_manual(name="Number_of\nPatches",values=c(16, 3, 17)) +
  scale_color_manual(name="Number_of\nPatches",
                    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
  scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
  scale_y_continuous(name="Average_Time_RE_(min)") +

ggplot(data=pred,aes(x = ni,y = Average..time.coef..m.,
                    col=factor(Patches))) +
  geom_jitter(width = 0.005,aes(shape=factor(Patches)),size = 2) +
  scale_shape_manual(name="Number_of\nPatches",values=c(16, 3, 17)) +
  scale_color_manual(name="Number_of\nPatches",

```



```

        values=c("#926b8d", "#7aadb1", "#E8CA47")) +
scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
scale_y_continuous(name="Average_Time_Coefficients_(min)") +

ggplot(data=pred, aes(x = ni, y = Average..steps, col=factor(Patches)))+
  geom_jitter(width = 0.005, aes(shape=factor(Patches)), size = 2) +
  scale_shape_manual(name="Number_of\nPatches", values=c(16, 3, 17)) +
  scale_color_manual(name="Number_of\nPatches",
    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
  scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
  scale_y_continuous(name="Average_ iterations_ before_ convergence",
    limits = c(10,15)) +

### Coefficients #####
coeff<-cbind(coef, sd_sim)
coeff2<-cbind(coef, se_sim)
colnames(coeff)<-c("Patches", "ni", "Scenario", "Beta1", "Beta2", "Beta3",
  "Beta4", "Gamma1", "Gamma2", "Gamma3", "Gamma4",
  "Gamma5", "Gamma6", "Gamma7", "Gamma8", "Patches.s",
  "ni.s", "Scenario.s", "Beta1.s", "Beta2.s", "Beta3.s",
  "Beta4.s", "Gamma1.s", "Gamma2.s", "Gamma3.s",
  "Gamma4.s", "Gamma5.s", "Gamma6.s", "Gamma7.s",
  "Gamma8.s")
colnames(coeff2)<-c("Patches", "ni", "Scenario", "Beta1", "Beta2", "Beta3",
  "Beta4", "Gamma1", "Gamma2", "Gamma3", "Gamma4",
  "Gamma5", "Gamma6", "Gamma7", "Gamma8", "Patches.s",
  "ni.s", "Scenario.s", "Beta1.s", "Beta2.s", "Beta3.s",
  "Beta4.s", "Gamma1.s", "Gamma2.s", "Gamma3.s",
  "Gamma4.s", "Gamma5.s", "Gamma6.s", "Gamma7.s",
  "Gamma8.s")

#plot with no bars
ggplot(data=coeff, aes(x = ni, y = Beta1, col=factor(Patches))) +
  geom_jitter(width = 0.1, aes(shape=factor(Patches)), size = 2) +
  geom_hline(yintercept=-10, color="#B7B6BA", linetype="dashed")+
  scale_shape_manual(name="Number_of\nPatches", values=c(16, 3, 17))+
  scale_color_manual(name="Number_of\nPatches",
    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
  scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
  scale_y_continuous(name="Intercept") +

#plot with sd of the estimates
ggplot(coeff, aes(x=ni, y=Beta1, color=factor(Patches))) +
  geom_point(position=position_dodge(0.5), aes(shape=factor(Patches)),
    size = 2)+
  geom_errorbar(aes(ymin=Beta1-(2*Beta1.s), ymax=Beta1+(2*Beta1.s)),
    width=.2, position=position_dodge(0.5))+
  geom_hline(yintercept=-10, color="#B7B6BA", linetype="dashed")+

```

```

scale_shape_manual(name="Number_of\nPatches",values=c(16, 3, 17)) +
scale_color_manual(name="Number_of\nPatches",
                    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
scale_y_continuous(name="Intercept") +

#plot with average se
ggplot(coeff2, aes(x=ni, y=Beta1,color=factor(Patches))) +
  geom_point(position=position_dodge(0.5),aes(shape=factor(Patches)),
            size = 2)+
  geom_errorbar(aes(ymin=Beta1-(1.96*Beta1.s), ymax=Beta1+
                    (1.96*Beta1.s)), width=.2,
               position=position_dodge(0.5))+
  geom_hline(yintercept=-10,color="#B7B6BA",linetype="dashed")+
scale_shape_manual(name="Number_of\nPatches",values=c(16, 3, 17)) +
scale_color_manual(name="Number_of\nPatches",
                    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
scale_y_continuous(name="Intercept") +

# Same plots for Beta2, Beta3, Beta4, Gamma1, Gamma2, Gamma3, Gamma4,
# Gamma5, Gamma6, Gamma7, Gamma8

### Bias #####
ggplot(data=bias,aes(x = ni,y = Beta1,col=factor(Patches))) +
  geom_jitter(width = 0.1,aes(shape=factor(Patches)),size = 2) +
  scale_shape_manual(name="Number_of\nPatches",values=c(16, 3, 17)) +
  scale_color_manual(name="Number_of\nPatches",
                    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
  scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
  scale_y_continuous(name="Bias_in_Intercept") +

# Same plots for Beta2, Beta3, Beta4, Gamma1, Gamma2, Gamma3, Gamma4,
# Gamma5, Gamma6, Gamma7, Gamma8

### Plots predictions #####

t.V1<- merge(t.V1,s.V1,by=c("Group.1","Group.2"),all=T)
t.V1_p<-merge(t.V1_p,s.V1_p,by=c("Group.1","Group.2"),all=T)
t.V1_t<-merge(t.V1_t,s.V1_t,by=c("Group.1","Group.2"),all=T)
t.V2<- merge(t.V2,s.V2,by=c("Group.1","Group.2"),all=T)
t.V2_p<-merge(t.V2_p,s.V2_p,by=c("Group.1","Group.2"),all=T)
t.V2_t<-merge(t.V2_t,s.V2_t,by=c("Group.1","Group.2"),all=T)

colnames(t.V1)<- c("Group.1","Group.2","N","ni","rep","file.x",
                 "file2.x","m_Vu","TSS","sse.V","ssr.V","tss",
                 "corr2","R2","mse","mae","mase","rmse","tvs",

```

```

      "uvs", "evs", "reps", "N.s", "ni.s", "rep.s", "file.s",
      "file2.s", "m_Vu.s", "TSS.s", "sse.V.s", "ssr.V.s",
      "tss.s", "corr2.s", "R2.s", "mse.s", "mae.s", "mase.s",
      "rmse.s", "tvs.s", "uvs.s", "evs.s", "reps.y")
colnames(t.V1_p) <- colnames(t.V1)
colnames(t.V1_t) <- colnames(t.V1)
colnames(t.V2) <- colnames(t.V1)
colnames(t.V2_p) <- colnames(t.V1)
colnames(t.V2_t) <- colnames(t.V1)

t.V1 <- t.V1[order(t.V1$N, t.V1$ni),]
t.V1_p <- t.V1_p[order(t.V1_p$N, t.V1_p$ni),]
t.V1_t <- t.V1_t[order(t.V1_t$N, t.V1_t$ni),]
t.V2 <- t.V2[order(t.V2$N, t.V2$ni),]
t.V2_p <- t.V2_p[order(t.V2_p$N, t.V2_p$ni),]
t.V2_t <- t.V2_t[order(t.V2_t$N, t.V2_t$ni),]

ggplot(t.V1_p, aes(x=ni, y=corr2, color=factor(N))) +
  geom_point(position=position_dodge(0.5), aes(shape=factor(N)),
            size = 2) +
  geom_errorbar(aes(ymin=corr2-(1.96*corr2.s),
                  ymax=min(1, corr2+(1.96*corr2.s))), width=.2,
              position=position_dodge(0.5)) +
  scale_shape_manual(name="Number_of\nPatches", values=c(16, 3, 17)) +
  scale_color_manual(name="Number_of\nPatches",
                    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
  scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
  scale_y_continuous(name=expression(paste("Corr"^2,
                                          "observed_vs.\npredicted")), limits = c(-1,1)) +

ggplot(t.V1_p, aes(x=ni, y=mase, color=factor(N))) +
  geom_point(position=position_dodge(0.5), aes(shape=factor(N)),
            size = 2) +
  geom_errorbar(aes(ymin=mase-(1.96*mase.s), ymax=mase+1.96*mase.s),
              width=.2, position=position_dodge(0.5)) +
  geom_hline(yintercept=1, color="#B7B6BA", linetype="dashed") +
  scale_shape_manual(name="Number_of\nPatches", values=c(16, 3, 17)) +
  scale_color_manual(name="Number_of\nPatches",
                    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
  scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
  scale_y_continuous(name="MASE_observed_vs.\npredicted",
                    limits = c(0,1.6)) +

# Same plots for t.V2_p

### Plots predictions with naive #####
t.naive <- merge(t.naive, s.naive, by=c("Group.1", "Group.2"), all=T)

```

```

t.naive<-t.naive[order(t.naive$N.x,t.naive$ni.x),]

ggplot(t.naive, aes(x=ni.x, y=naive.corr2.V1.x,color=factor(N.x))) +
  geom_point(position=position_dodge(0.5),aes(shape=factor(N.x)),
            size = 2)+
  geom_errorbar(aes(ymin=naive.corr2.V1.x-(1.96*naive.corr2.V1.y),
                  ymax=min(1,naive.corr2.V1.x+
                          (1.96*naive.corr2.V1.y))), width=.2,
              position=position_dodge(0.5))+
  scale_shape_manual(name="Number_of\nPatches",values=c(16, 3, 17)) +
  scale_color_manual(name="Number_of\nPatches",
                    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
  scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
  scale_y_continuous(name=expression(paste("Corr"^2,
                                          "observed_vs.\npredicted")),limits = c(-1,1)) +

ggplot(t.naive, aes(x=ni.x, y=naive.mase.V1.x,color=factor(N.x))) +
  geom_point(position=position_dodge(0.5),aes(shape=factor(N.x)),
            size = 2)+
  geom_errorbar(aes(ymin=naive.mase.V1.x-(1.96*naive.mase.V1.y),
                  ymax=naive.mase.V1.x+(1.96*naive.mase.V1.y)),
              width=.2,position=position_dodge(0.5))+
  geom_hline(yintercept=1,color="#B7B6BA",linetype="dashed")+
  scale_shape_manual(name="Number_of\nPatches",values=c(16, 3, 17)) +
  scale_color_manual(name="Number_of\nPatches",
                    values=c("#926b8d", "#7aadb1", "#E8CA47")) +
  scale_x_continuous(name="Time_points", breaks=c(3,4,5)) +
  scale_y_continuous(name="MASE_observed_vs.\npredicted",
                    limits = c(0,1.6)) +

# Same plots for naive.corr2.V2, naive.mase.V1.x

#####
# Previous code was repeated for Type I error simulations using the
# following data set

# Data with zero dose effect #####
# Dose coefficients were set to zero. Also, the intercept was changed to
# get higher probabilities to get 0s and 1s
set.seed(124)
for (n in c(300,500,1000)) {
  for (ni in c(3,4,5)) {
    for (rp in 1:1000) {

      # Dose
      Dose<-rlnorm(n, meanlog = 0, sdlog =1)
      Dose<-Dose*(60/max(Dose))
    }
  }
}

```

```

#plot(density(Dose))
Dose <- rep(Dose,times=ni)

id <- rep(1:n,times=ni)
t <- c(3,6,12,18,24)/12
t <- t[1:ni]
time <- rep(t, each=n)
data <- as.data.frame(cbind(id,Dose,time))
data$time2 <- (data$time)^2
data$int <- 1
data <- data[,c("id","int","Dose","time","time2")]
#dataU <- data[,c("id","int","Dose","time")]
dataU<-data

# Parameter
m <- 0.6
mu_1 <- c(-0.1,0.1)
mu_2 <- -(m/(1-m))*mu_1
Psi_diag<-diag(sqrt(c(1,0.5,1)))
R <- matrix(c(1,0.1,0.2,0.1,1,0.1,0.2,0.1,1),nrow = 3)
Psi<-Psi_diag %*% R %*% Psi_diag
Psi_cc <- as.matrix(Psi[1,1])
Psi_dd <- as.matrix(Psi[2:3,2:3])
Psi_dc <- as.matrix(Psi[2:3,1])
Psi_cd <- t(Psi_dc)
#is.positive.definite(Psi_cc)
#is.positive.definite(Psi_dd)
#is.positive.definite(Psi)
mu1 <- matrix(c(0,mu_1),ncol = 1)
mu2 <- matrix(c(0,mu_2),ncol = 1)

corr <- 0.2
Var.V1 <- 1
Var.V2 <- 0.5
Sigma <- matrix(c(Var.V1,rep(corr*sqrt(Var.V1)*sqrt(Var.V2),2),
Var.V2),nrow = 2,ncol = 2)
Beta <- as.matrix(c(1,0,-2,-3),ncol=1)
Gamma <- as.matrix(c(-5,0,0.5,-1,
-2,0,0.5,-1),ncol=1)
b <- (m*mvrnorm(n = n, mu1, Psi))+((1-m)*mvrnorm(n = n, mu2, Psi))
d <- data.frame(b[,2:3],i=rep(1:ni,ea=n))
d <- d[,1:2]
c <- data.frame(b[,1],i=rep(1:ni,ea=n))
c <- c[,1]
e <- mvrnorm(n=ni*n,c(0,0),Sigma)

X <- as.matrix(dataU[, -1])

```

```

Z <- as.matrix(dataU[,c("int")])
Zc <- (Z * c)
logit_U <- X %*% Beta + Zc
pU <- exp(logit_U)/(1+exp(logit_U))
U <-rbinom(length(pU),1,pU)

ps<-prop.table(table(U))[2]

Zd1 <- Z * d[,1]
Zd2 <- Z * d[,2]
X <- as.matrix(data[,-1])
Z <- as.matrix(data[,c("int")])
V1 <- X %*% Gamma[1:4,] + Zd1 + e[,1]
V2 <- X %*% Gamma[5:8,] + Zd2 + e[,2]

data <- cbind(data,pU,U,V1,V2)
data$V1u <- ifelse(data$U==0,0,data$V1)
data$V2u <- ifelse(data$U==0,0,data$V2)
data$nonZeroHD <- data$U

sim.val<-c(Beta=Beta,Gamma=Gamma, Psi_cc=Psi_cc,Psi_dd=Psi_dd,
          Var.V1=Var.V1,Var.V2=Var.V2,corr=corr,
          Psi_dc=Psi_dc,m=m,mu_1=mu_1)

#plot(data$Dose,data$V1u)
#plot(data$Dose,data$V2u)
#boxplot(Dose~nonZeroHD, data=data)

#####
rm(list=setdiff(ls(), c("data","maxit","n","ni","rp","ps","sim.val")))

dataset<-data[,-2]

rm(data)
names(dataset)<-c("patch","Dose","time","time2","pU","U","V1","V2",
               "V1u","V2u","nonZeroHD")
filename<-paste("/home/rodriguezva/myR/Dissertation/simd0/",
               n,"/",ni,"/data/","dsd0","_",n,"_",ni,"_",rp,
               ".RData",sep = "")
save.image(file = filename)

}
}
}

# Data with zero time effect #####

```

```

set.seed(124)
for (n in c(300,500,1000)) {
  for (ni in c(3,4,5)) {
    for (rp in 1:1000) {

      # Dose
      Dose<-rlnorm(n, meanlog = 0, sdlog =1)
      Dose<-Dose*(60/max(Dose))
      #plot(density(Dose))
      Dose <- rep(Dose,times=ni)

      id <- rep(1:n,times=ni)
      t <- c(3,6,12,18,24)/12
      t <- t[1:ni]
      time <- rep(t, each=n)
      data <- as.data.frame(cbind(id,Dose,time))
      data$time2 <- (data$time)^2
      data$int <- 1
      data <- data[,c("id","int","Dose","time","time2")]
      #dataU <- data[,c("id","int","Dose","time")]
      dataU<-data

      # Parameter
      m <- 0.6
      mu_1 <- c(-0.1,0.1)
      mu_2 <- -(m/(1-m))*mu_1
      Psi_diag<-diag(sqrt(c(1,0.5,1)))
      R <- matrix(c(1,0.1,0.2,0.1,1,0.1,0.2,0.1,1),nrow = 3)
      Psi<-Psi_diag %*% R %*% Psi_diag
      Psi_cc <- as.matrix(Psi[1,1])
      Psi_dd <- as.matrix(Psi[2:3,2:3])
      Psi_dc <- as.matrix(Psi[2:3,1])
      Psi_cd <- t(Psi_dc)
      #is.positive.definite(Psi_cc)
      #is.positive.definite(Psi_dd)
      #is.positive.definite(Psi)
      mu1 <- matrix(c(0,mu_1),ncol = 1)
      mu2 <- matrix(c(0,mu_2),ncol = 1)

      corr <- 0.2
      Var.V1 <- 1
      Var.V2 <- 0.5
      Sigma <- matrix(c(Var.V1,rep(corr*sqrt(Var.V1)*sqrt(Var.V2),2),
                          Var.V2),nrow = 2,ncol = 2)
      Beta <- as.matrix(c(-10,1.5,0,0),ncol=1)
      Gamma <- as.matrix(c(-7,1,0,0,
                          -3,2,0,0),ncol=1)
    }
  }
}

```

```

b <- (m*mvrnorm(n = n, mu1, Psi))+((1-m)*mvrnorm(n = n, mu2, Psi))
d <- data.frame(b[,2:3],i=rep(1:ni,ea=n))
d <- d[,1:2]
c <- data.frame(b[,1],i=rep(1:ni,ea=n))
c <- c[,1]
e <- mvrnorm(n=ni*n,c(0,0),Sigma)

X <- as.matrix(dataU[,-1])
Z <- as.matrix(dataU[,c("int")])
Zc <- (Z * c)
logit_U <- X %*% Beta + Zc
pU <- exp(logit_U)/(1+exp(logit_U))
U <-rbinom(length(pU),1,pU)

ps<-prop.table(table(U))[2]

Zd1 <- Z * d[,1]
Zd2 <- Z * d[,2]
X <- as.matrix(data[,,-1])
Z <- as.matrix(data[,c("int")])
V1 <- X %*% Gamma[1:4,] + Zd1 + e[,1]
V2 <- X %*% Gamma[5:8,] + Zd2 + e[,2]

data <- cbind(data,pU,U,V1,V2)
data$V1u <- ifelse(data$U==0,0,data$V1)
data$V2u <- ifelse(data$U==0,0,data$V2)
data$nonZeroHD <- data$U

sim.val<-c(Beta=Beta,Gamma=Gamma, Psi_cc=Psi_cc,Psi_dd=Psi_dd,
          Var.V1=Var.V1,Var.V2=Var.V2,corr=corr,
          Psi_dc=Psi_dc,m=m,mu_1=mu_1)

#plot(data$Dose,data$V1u)
#plot(data$Dose,data$V2u)
#boxplot(Dose~nonZeroHD, data=data)

#####
rm(list=setdiff(ls(), c("data","maxit","n","ni","rp","ps","sim.val")))

dataset<-data[,-2]

rm(data)
names(dataset)<-c("patch","Dose","time","time2","pU","U","V1","V2",
               "V1u", "V2u","nonZeroHD")
filename<-paste("/home/rodriguezva/myR/Dissertation/simt0/",
               n,"/",ni,"/data/","dst0","_",n,"_",ni,"_",rp,

```



```

                                ".RData",sep = "")
    save.image(file = filename)
  }
}
}

# Data with zero time^2 effect #####
set.seed(124)
for (n in c(300,500,1000)) {
  for (ni in c(3,4,5)) {
    for (rp in 1:1000) {

      # Dose
      Dose<-rlnorm(n, meanlog = 0, sdlog =1)
      Dose<-Dose*(60/max(Dose))
      #plot(density(Dose))
      Dose <- rep(Dose,times=ni)

      id <- rep(1:n,times=ni)
      t <- c(3,6,12,18,24)/12
      t <- t[1:ni]
      time <- rep(t, each=n)
      data <- as.data.frame(cbind(id,Dose,time))
      data$time2 <- (data$time)^2
      data$int <- 1
      data <- data[,c("id","int","Dose","time","time2")]
      #dataU <- data[,c("id","int","Dose","time")]
      dataU<-data

      # Parameter
      m <- 0.6
      mu_1 <- c(-0.1,0.1)
      mu_2 <- -(m/(1-m))*mu_1
      Psi_diag<-diag(sqrt(c(1,0.5,1)))
      R <- matrix(c(1,0.1,0.2,0.1,1,0.1,0.2,0.1,1),nrow = 3)
      Psi<-Psi_diag %*% R %*% Psi_diag
      Psi_cc <- as.matrix(Psi[1,1])
      Psi_dd <- as.matrix(Psi[2:3,2:3])
      Psi_dc <- as.matrix(Psi[2:3,1])
      Psi_cd <- t(Psi_dc)
      #is.positive.definite(Psi_cc)
      #is.positive.definite(Psi_dd)
      #is.positive.definite(Psi)
      mu1 <- matrix(c(0,mu_1),ncol = 1)
      mu2 <- matrix(c(0,mu_2),ncol = 1)
    }
  }
}

```

```

corr <- 0.2
Var.V1 <- 1
Var.V2 <- 0.5
Sigma <- matrix(c(Var.V1,rep(corr*sqrt(Var.V1)*sqrt(Var.V2),2),
                  Var.V2),nrow = 2,ncol = 2)
Beta <- as.matrix(c(-10,1.5,-2,0),ncol=1)
Gamma <- as.matrix(c(-7,1,0.5,0,
                    -3,2,0.5,0),ncol=1)
b <- (m*mvrnorm(n = n, mu1, Psi))+((1-m)*mvrnorm(n = n, mu2, Psi))
d <- data.frame(b[,2:3],i=rep(1:ni,ea=n))
d <- d[,1:2]
c <- data.frame(b[,1],i=rep(1:ni,ea=n))
c <- c[,1]
e <- mvrnorm(n=ni*n,c(0,0),Sigma)

X <- as.matrix(dataU[,-1])
Z <- as.matrix(dataU[,c("int")])
Zc <- (Z * c)
logit_U <- X %*% Beta + Zc
pU <- exp(logit_U)/(1+exp(logit_U))
U <-rbinom(length(pU),1,pU)

ps<-prop.table(table(U))[2]

Zd1 <- Z * d[,1]
Zd2 <- Z * d[,2]
X <- as.matrix(data[, -1])
Z <- as.matrix(data[,c("int")])
V1 <- X %*% Gamma[1:4,] + Zd1 + e[,1]
V2 <- X %*% Gamma[5:8,] + Zd2 + e[,2]

data <- cbind(data,pU,U,V1,V2)
data$V1u <- ifelse(data$U==0,0,data$V1)
data$V2u <- ifelse(data$U==0,0,data$V2)
data$nonZeroHD <- data$U

sim.val<-c(Beta=Beta,Gamma=Gamma, Psi_cc=Psi_cc,Psi_dd=Psi_dd,
          Var.V1=Var.V1,Var.V2=Var.V2,corr=corr,
          Psi_dc=Psi_dc,m=m,mu_1=mu_1)

#plot(data$Dose,data$V1u)
#plot(data$Dose,data$V2u)
#boxplot(Dose~nonZeroHD, data=data)

#####
rm(list=setdiff(ls(), c("data","maxit","n","ni","rp","ps","sim.val")))

```

```

dataset<-data[,-2]

rm(data)
names(dataset)<-c("patch","Dose","time","time2","pU","U","V1","V2",
                 "V1u","V2u","nonZeroHD")
filename<-paste("/home/rodriguezva/myR/Dissertation/simt20/",
                n,"/",ni,"/data/","dst20","_",n,"_",ni,"_",rp,
                ".RData",sep = "")
save.image(file = filename)

}
}
}
#warnings()

```

B.4 Chapter 5: Application on reference patienter data

Also available at: <https://github.com/vivifj03/TumorModel/blob/master/Code%20chapter5%20-%20ref%20patient.R>

```

#Following code runs for all patch definitions
# Packages -----
library(lme4)
library(nlme)
library(LearnBayes)
library(plyr)
library(pracma)
library(ks)
library(matrixcalc)
library(MASS)
library(mvtnorm)
library(loo)
library(scorer)

setwd("")
load("") #RData file with patch definition

filelklhd<-"dataset_for_lklhd200425.RData"
fileestimates<-"estimates200425.RData"
filepred<-"predictions200425.RData"

# Initial values -----
dataset<-dataset[order(dataset$patch,dataset$date),]

```

```

total.inicio<-Sys.time()
##### Mixed effects logistic regression - gets initial values
#This models is using time in years
inicio.naive<-Sys.time()
md1 <- glmer(nonZeroHD ~ Dose_med + years + years2 + (1 | patch),
            data = dataset, family = binomial)
temp<-dataset[is.na(dataset$Dense.lt)==F | is.na(dataset$Hazy.lt)==F,]
###Dense
md2d <- lme(Dense.lt ~ Dose_med + years + years2, random = ~ 1| patch,
            data = temp)
md2d2 <- lmer(Dense.lt ~ Dose_med + years + years2 + (1| patch),
            data = temp, control = lmerControl(optCtrl=
            list(maxfun=2e10)))
md2d2.r<-ranef(md2d2)
save.image(filelklhd)
###Hazy
md2h <- lme(Hazy.lt ~ Dose_med + years + years2, random= ~ 1| patch,
            data = temp)
md2h2 <- lmer(Hazy.lt ~ Dose_med + years + years2 + (1| patch),
            data = temp, control = lmerControl(optCtrl=
            list(maxfun=2e10)))
save.image(filelklhd)

#Initial values
#Occurrence variable
Beta<- as.matrix(md1@beta,ncol=1)
if (((md1@theta)^2)<0.1) {
  Psi_cc<-matrix(0.1,1,1)
} else {
  Psi_cc<-matrix(md1@theta)^2
}

#Intensity variable
Gamma<-c(md2d2@beta,md2h2@beta)
Psi_dd<-matrix(c(as.numeric(VarCorr(md2d)[1]),
                sqrt(as.numeric(VarCorr(md2d)[1]))*
                sqrt(as.numeric(VarCorr(md2h)[1]))*0.1,
                sqrt(as.numeric(VarCorr(md2d)[1]))*
                sqrt(as.numeric(VarCorr(md2h)[1]))*0.1,
                as.numeric(VarCorr(md2h)[1])),
              nrow = length(md2d2@theta)+length(md2h2@theta),
              ncol = length(md2d2@theta)+length(md2h2@theta))
corr<-cor(temp$Dense.lt, temp$Hazy.lt)
Var.V1<-attr(VarCorr(md2d2), "sc")^2
Var.V2<-attr(VarCorr(md2h2), "sc")^2

#Corr between random effects between models

```

```

Psi_dc<-matrix(c(sqrt(diag(Psi_dd))%*%sqrt(Psi_cc)*0.2),
              nrow=ncol(Psi_dd),ncol=ncol(Psi_cc))

m<-0.55 #set probability for normal mixture distribution for
        #random effects
mu_1<-c(-0.05,0.05) #Set mean first normal distribution for
        #random effects

par.l<-list(Beta=Beta,Gamma=Gamma, Psi_cc=Psi_cc,Psi_dd=Psi_dd,
            Var.V1=Var.V1,Var.V2=Var.V2,corr=corr,
            Psi_dc=Psi_dc,m=m,mu_1=mu_1)
par<-c(Beta=Beta,Gamma=Gamma, Psi_cc=Psi_cc,Psi_dd=Psi_dd,
       Var.V1=Var.V1,Var.V2=Var.V2,corr=corr,
       Psi_dc=Psi_dc,m=m,mu_1=mu_1)

patch<-dataset$patch
patch<-unique(patch)

patch.ch<-dataset[dataset$chageID==1,]$patch
patch.ch<-unique(patch.ch)

patch.nch<-dataset[dataset$chageID==0,]$patch
patch.nch<-unique(patch.nch)

model.data<-list(datafile=dataset, id.var = patch,
                 X.var=c("Dose_med","years","years2"),Z.var=NULL,
                 X_a.var=c("Dose_med","years","years2"),Z_a.var=NULL,
                 U.var="nonZeroHD",V1="Dense.lt0",V2="Hazy.lt0")

save.image(filelklhd)

# Likelihood - Model -----
source("tpmemmzl.R")

# Estimation of fixed effects =====
inicio.coef<-Sys.time()
maxit=1000
step=1
criterion=0
estimate<- vector("list", maxit+1)
estimate[[1]]<-par.l
dif.theta<- vector("list", maxit)
rel.dif.theta<- vector("list", maxit)
dev<-vector("numeric", maxit)
reldev<-vector("numeric", maxit)
eu.dis<- vector("numeric", maxit)

```

```

theta<-c(Beta=par.l$Beta,
         Gamma=par.l$Gamma,
         mu=par.l$mu_1,
         Sigma.vech=vech(matrix(c(par.l$Var.V1,rep(par.l$corr*
            sqrt(par.l$Var.V1)*sqrt(par.l$Var.V2),2),
            par.l$Var.V2),nrow = 2,ncol = 2)),
         Psi_cc.vech=vech(par.l$Psi_cc),
         Psi_dc.i=par.l$Psi_dc %% ginv(par.l$Psi_cc),
         H.vech=vech(par.l$Psi_dd-(par.l$Psi_dc %%
            ginv(par.l$Psi_cc) %% t(par.l$Psi_dc)) ))

in.dev<-lapply(patch,function(x) lklhd.dev(id=x,datafile=dataset,
            initial=par.l,
            X.var=c("Dose_med","years","years2"),Z.var=NULL,
            X_a.var=c("Dose_med","years","years2"),Z_a.var=NULL,
            U.var="nonZeroHD",V1="Dense.lt0",V2="Hazy.lt0"))

t.dev<-matrix(0,nrow=length(in.dev), ncol = 1)
for (i in 1:length(in.dev)) {
  t.dev[i,1]<-in.dev[[i]]
}
dev[[1]]<- -2*colSums(t.dev)
print(dev[[1]])

while (criterion<=0) {
  print(step)
  inicio2<-Sys.time()

  par.l0<-par.l
  theta<-c(Beta=par.l$Beta,
           Gamma=par.l$Gamma,
           mu=par.l$mu_1,
           Sigma.vech=vech(matrix(c(par.l$Var.V1,rep(par.l$corr*
              sqrt(par.l$Var.V1)*sqrt(par.l$Var.V2),2),
              par.l$Var.V2),nrow = 2,ncol = 2)),
           Psi_cc.vech=vech(par.l$Psi_cc),
           Psi_dc.i=par.l$Psi_dc %% ginv(par.l$Psi_cc),
           H.vech=vech(par.l$Psi_dd-(par.l$Psi_dc %%
              ginv(par.l$Psi_cc) %% t(par.l$Psi_dc)) ))

  theta0<-theta

  t<-lapply(patch,function(x) lklhd.em(id=x,datafile=dataset,
            initial=par.l,
            X.var=c("Dose_med","years","years2"),Z.var=NULL,
            X_a.var=c("Dose_med","years","years2"),Z_a.var=NULL,
            U.var="nonZeroHD",V1="Dense.lt0",V2="Hazy.lt0"))

```

```

print("Done with lklhd.em")
subm<-matrix(0,nrow=length(t), ncol = 1)
for (i in 1:length(t)) {
  subm[i,1]<-t[[i]]$m
}
score<-matrix(0,nrow=length(t), ncol = length(t[[1]]$scorei))
for (i in 1:length(t)) {
  score[i,]<-t[[i]]$scorei
}

trunk<- c(18:23)
for (i in trunk) {
  cuts<-0+3*sd(score[,i])
  for (j in 1:nrow(score)) {
    score[j,i]<-ifelse(abs(score[j,i])>cuts,0,score[j,i])
  }
}

score<-colSums(score,na.rm = T)
theta<-theta+score

#Check values of Sigma, Psi_cc
while (theta[15]<=0)
{
  test0<-theta0[15]
  #test1<-theta[15]
  test1<-0.001
  theta[15]<-(test0+test1)/2
}
while (theta[17]<=0)
{
  test0<-theta0[17]
  #test1<-theta[17]
  test1<-0.001
  theta[17]<-(test0+test1)/2
}
corr.test<-theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
while (corr.test<= (-0.95)) {
  corr.test0<- theta0[16]/(sqrt(theta0[15])*sqrt(theta0[17]))
  #corr.test1<- theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
  corr.test1<- -0.90
  corr.new<- (corr.test0+corr.test1)/2
  theta[16]<-corr.new*sqrt(theta[15])*sqrt(theta[17])
  corr.test<-theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
}
while (corr.test >= 0.95) {
  corr.test0<- theta0[16]/(sqrt(theta0[15])*sqrt(theta0[17]))

```

```

#corr.test1<- theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
corr.test1<- 0.90
corr.new<- (corr.test0+corr.test1)/2
theta[16]<-corr.new*sqrt(theta[15])*sqrt(theta[17])
corr.test<-theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
}
while (theta[18]<=0)
{
  test0<-theta0[18]
  #test1<-theta[18]
  test1<-0.001
  theta[18]<-(test0+test1)/2
}

#New estimates
m<-mean(subm)
m<-ifelse(m<0.5,(1-m),m)
while (m>=0.95)
{
  m0<-par.l$m
  m<-(0.94+m0)/2
}
Beta=matrix(theta[1:4],ncol=1)
Gamma=matrix(theta[5:12],ncol=1)
mu_1=matrix(theta[13:14],ncol=1)
Var.V1=theta[15]
Var.V2=theta[17]
corr=theta[16]/(sqrt(theta[15])*sqrt(theta[17]))
Psi_cc=matrix(theta[18],ncol=1)
Psi_dc.i=matrix(theta[19:20],ncol=1)
Psi_dc=Psi_dc.i%*%Psi_cc
Psi_cd=t(Psi_dc)
H<-matrix(c(theta[21],rep(theta[22],2),theta[23]),ncol=2)

while (det(H) <= 0 )
{
  theta[21] <- (theta0[21]+theta[21])/2
  theta[22] <- (theta0[22]+theta[22])/2
  theta[23] <- (theta0[23]+theta[23])/2
  H<-matrix(c(theta[21],rep(theta[22],2),theta[23]),ncol=2)
}

while (theta[21]<=0)
{
  test0<-theta0[21]
  test1<-0.001
  theta[21]<-(test0+test1)/2
}

```



```

}

while (theta[23] <= 0)
{
  test0 <- theta0[23]
  test1 <- 0.001
  theta[23] <- (test0 + test1) / 2
}
H <- matrix(c(theta[21], theta[22], theta[22], theta[23]), ncol=2)

while (det(H) <= 0)
{
  theta[21] <- (theta0[21] + theta[21]) / 2
  theta[22] <- (theta0[22] + theta[22]) / 2
  theta[23] <- (theta0[23] + theta[23]) / 2
  H <- matrix(c(theta[21], rep(theta[22], 2), theta[23]), ncol=2)
}
Psi_dd = H + (Psi_dc.i %*% Psi_cd)

par.l <- list(Beta = Beta, Gamma = Gamma, Psi_cc = Psi_cc, Psi_dd = Psi_dd,
             Var.V1 = Var.V1, Var.V2 = Var.V2, corr = corr,
             Psi_dc = Psi_dc, m = m, mu_1 = mu_1)

step = step + 1

#Deviance
in.dev <- lapply(patch, function(x) lklhd.dev(id=x, datafile=dataset,
      initial=par.l,
      X.var=c("Dose_med", "years", "years2"), Z.var=NULL,
      X_a.var=c("Dose_med", "years", "years2"), Z_a.var=NULL,
      U.var="nonZeroHD", V1="Dense.lt0", V2="Hazy.lt0"))

t.dev <- matrix(0, nrow=length(in.dev), ncol = 1)
for (i in 1:length(in.dev)) {
  t.dev[i,1] <- in.dev[[i]]
}
dev[[step]] <- -2 * colSums(t.dev)

print(paste("new deviance", dev[[step]]))
#Step-halving in case deviance is larger than previous iteration
reldev[[step]] <- ((dev[[step]] - dev[[step-1]]) / (0.1 +
  abs(dev[[step]])) )

loop3 <- 0
if (reldev[[step]] >= 1e-8) {
  ii <- 1
  while (reldev[[step]] > -1e-8) {

```

```

if (ii>50) break
ii<-ii+1
loop3<-ifelse(ii==50,1,loop3)
par.l$Beta<-(par.l$Beta+par.l0$Beta)/2
par.l$Gamma<-(par.l$Gamma+par.l0$Gamma)/2
par.l$Psi_cc<-(par.l$Psi_cc+par.l0$Psi_cc)/2
par.l$Psi_dd<-(par.l$Psi_dd+par.l0$Psi_dd)/2
par.l$Var.V1<-(par.l$Var.V1+par.l0$Var.V1)/2
par.l$Var.V2<-(par.l$Var.V2+par.l0$Var.V2)/2
par.l$corr<-(par.l$corr+par.l0$corr)/2
par.l$Psi_dc<-(par.l$Psi_dc+par.l0$Psi_dc)/2
par.l$m<-(par.l$m+par.l0$m)/2
par.l$mu_1<-(par.l$mu_1+par.l0$mu_1)/2

in.dev<-lapply(patch,function(x) lklhd.dev(id=x,
      datafile=dataset,initial=par.l,
      X.var=c("Dose_med","years","years2"),
      Z.var=NULL,
      X_a.var=c("Dose_med","years","years2"),
      Z_a.var=NULL,
      U.var="nonZeroHD",V1="Dense.lt0",
      V2="Hazy.lt0"))
t.dev<-matrix(0,nrow=length(in.dev), ncol = 1)
for (i in 1:length(in.dev)) {
  t.dev[i,1]<-in.dev[[i]]
}
dev[[step]]<- -2*colSums(t.dev)
reldev[[step]]<-((dev[[step]] - dev[[step-1]])/(0.1 +
  abs(dev[[step]])) )
print( paste("base deviance",dev[[step-1]], "new deviance",
  dev[[step]]))
}
}

reldev[[step]]<-abs((dev[[step]] - dev[[step-1]])/(0.1 +
  abs(dev[[step]])) )
theta<-c(Beta=par.l$Beta,
  Gamma=par.l$Gamma,
  mu=par.l$mu_1,
  Sigma.vech=vech(matrix(c(par.l$Var.V1,rep(par.l$corr*
  sqrt(par.l$Var.V1)*sqrt(par.l$Var.V2),2),
  par.l$Var.V2),nrow = 2,ncol = 2)),
  Psi_cc.vech=vech(par.l$Psi_cc),
  Psi_dc.i=par.l$Psi_dc %*% solve(par.l$Psi_cc),
  H.vech=vech(par.l$Psi_dd-(par.l$Psi_dc %*%
  solve(par.l$Psi_cc) %*% t(par.l$Psi_dc)) ))

```

```

estimate[[step]]<-par.1
eu.dis[step] <-(dist(rbind(theta0,theta), method = "euclidean")^2)/
      length(theta)
dif.theta[[step]] <- (theta0-theta)
rel.dif.theta[[step]] <- (theta-theta0)/theta
absdev<-ifelse(step==2,1,abs(dev[[step]] - dev[[step-1]]))

criterion.step<-ifelse(step<=maxit,0,1)
criterion.distance<-ifelse(max(abs(dif.theta[[step]]))>1e-8,0,1)
criterion.max<-ifelse(max(abs(rel.dif.theta[[step]]))>1e-8,0,1)
criterion.dev<-ifelse(absdev>1e-8,0,1)
criterion.reldev<-ifelse(reldev[[step]]>1e-8,0,1)

criterion<-criterion.step+criterion.distance+criterion.max+
      criterion.dev+criterion.reldev

criterion<-ifelse(loop3==1 & step<5,0,criterion)

print(c(criterion.step=criterion.step,criterion.distance=
      criterion.distance,criterion.max=criterion.max,
      criterion=criterion))
print(c(eu.dis=eu.dis[step],max.abs.dif=max(abs(dif.theta[[step]])),
      max.rel=max(abs(rel.dif.theta[[step]])),
      dev=dev[step],absdev.dif=absdev,reldev=reldev[step]))

end2<-Sys.time()
timeit<- end2-inicio2
print(theta)
print(timeit)
}
end.coef<-Sys.time()
end.coef-inicio.coef

# For figures of coefficients at each iteration
Beta<-matrix(NA,step,4)
for(i in 1:step){
  Beta[i,]<-(estimate[[i]]$Beta)
}

Gamma<-matrix(NA,step,8)
for(i in 1:step){
  Gamma[i,]<-(estimate[[i]]$Gamma)
}

Psi_cc<-matrix(NA,step,1)

```

```

for(i in 1:step){
  Psi_cc[i,]<-(estimate[[i]]$Psi_cc)
}

Psi_dd<-matrix(NA,step,4)
for(i in 1:step){
  Psi_dd[i,]<-(estimate[[i]]$Psi_dd)
}

Var.V1<-matrix(NA,step,1)
for(i in 1:step){
  Var.V1[i,]<-(estimate[[i]]$Var.V1)
}
Var.V2<-matrix(NA,step,1)
for(i in 1:step){
  Var.V2[i,]<-(estimate[[i]]$Var.V2)
}
corr<-matrix(NA,step,1)
for(i in 1:step){
  corr[i,]<-(estimate[[i]]$corr)
}

m<-matrix(NA,step,1)
for(i in 1:step){
  m[i,]<-(estimate[[i]]$m)
}
mu_1<-matrix(NA,step,2)
for(i in 1:step){
  mu_1[i,]<-(estimate[[i]]$mu_1)
}

save.image(fileestimates)

### Standard errors #####
se<-lapply(patch,function(x) lklhd.var(id=x,datafile=dataset,
  est=par.l,
  X.var=c("Dose_med","years","years2"),Z.var=NULL,
  X_a.var=c("Dose_med","years","years2"),Z_a.var=NULL,
  U.var="nonZeroHD",V1="Dense.lt0",V2="Hazy.lt0"))

S_0b.total<-matrix(0,nrow=4, ncol = 4)
S_0g.total<-matrix(0,nrow=8, ncol = 8)
S_0mu.total<-matrix(0,nrow=2, ncol = 2)

for (i in 1:length(se)) {
  S_0b.total<-S_0b.total+se[[i]]$S_0b
  S_0g.total<-S_0g.total+se[[i]]$S_0g
}

```

```

    S_0mu.total<-S_0mu.total+se[[i]]$S_0mu
  }
  S_0b.total.i<-ginv(S_0b.total)
  S_0g.total.i<-ginv(S_0g.total)
  S_0mu.total.i<-ginv(S_0mu.total)

  sqrt(diag(S_0b.total.i))
  sqrt(diag(S_0g.total.i))

  par.final<-c(unlist(par.l))
  se.final1<-c(sqrt(diag(S_0b.total.i)),sqrt(diag(S_0g.total.i)))
  lci1<-par.final[1:12]-(1.96*se.final1)
  uci1<-par.final[1:12]+(1.96*se.final1)
  save.image(fileestimates)

### Random effects #####
try(if(criterion.step==1) stop("estimation did not converge"))

iniciop<-Sys.time()
random.est <- lapply(patch,function(x) lklhd.ram(id=x,
          datafile=dataset,est=par.l,
          X.var=c("Dose_med","years","years2"),
          Z.var=NULL,
          X_a.var=c("Dose_med","years","years2"),
          Z_a.var=NULL,
          U.var="nonZeroHD",V1="Dense.lt0",
          V2="Hazy.lt0"))

endp<-Sys.time()
endp-iniciop
save.image(filepred)

### Predicted values #####
ran.est <- as.data.frame(matrix(NA,length(patch),7))
ran.est[,1] <- patch
for(i in 1:length(patch)){
  ran.est[i,2]<-(random.est[[i]]$ram.est$c.est)
  ran.est[i,3]<-(random.est[[i]]$ram.est$cc.t.est)
  ran.est[i,4]<-(random.est[[i]]$ram.est$d1.est)
  ran.est[i,5]<-(random.est[[i]]$ram.est$d2.est)
  ran.est[i,6]<-(random.est[[i]]$ram.est$ess1)
  ran.est[i,7]<-(random.est[[i]]$ram.est$ess2)
}
ran.est <- as.data.frame(ran.est)
colnames(ran.est) <- c("patch","c.est","cc.t.est","d1.est","d2.est",
  "ess","ess2")
dataset<-dataset[,1:38]

```

```

dataset<-merge(dataset,ran.est,by="patch",all.x = T)
dataset$logp.1 <- par.l$Beta[1] + par.l$Beta[2]*dataset$Dose_med +
  par.l$Beta[3] * dataset$years + par.l$Beta[4] * dataset$years2 +
  dataset$c.est
dataset$odds.1 <- exp(dataset$logp.1)
dataset$p.1 <- dataset$odds.1 / (1 + dataset$odds.1)
dataset$subpred1 <- par.l$Gamma[1] + par.l$Gamma[2]*dataset$Dose_med +
  par.l$Gamma[3] * dataset$years + par.l$Gamma[4] * dataset$years2 +
  dataset$d1.est
dataset$subpred2 <- par.l$Gamma[5]+par.l$Gamma[6]*dataset$Dose_med +
  par.l$Gamma[7] * dataset$years + par.l$Gamma[8] * dataset$years2 +
  dataset$d2.est
dataset$subres1 <- dataset$Dense.lt0-dataset$subpred1
dataset$subres2 <- dataset$Hazy.lt0-dataset$subpred2
dataset$pred1<- dataset$p.1*dataset$subpred1
dataset$pred2<- dataset$p.1*dataset$subpred2
dataset$residuals1<-(dataset$Dense.lt0-dataset$pred1)
dataset$residuals2<-(dataset$Hazy.lt0-dataset$pred2)
dataset$sresiduals1<-dataset$residuals1^2
dataset$sresiduals2<-dataset$residuals2^2

duan1<- mean(exp(dataset[dataset$nonZeroHD==1,"subres1"]))
duan2<- mean(exp(dataset[dataset$nonZeroHD==1,"subres2"]))

dataset$pred1.e<- dataset$p.1*exp(dataset$subpred1)*duan1
dataset$pred2.e<- dataset$p.1*exp(dataset$subpred2)*duan2
dataset$residuals1.e<-(dataset$Dense.t0-dataset$pred1.e)
dataset$residuals2.e<-(dataset$Hazy.t0-dataset$pred2.e)

ram.m_Vu.V1<-mean(dataset$Dense.t0)
ram.TSS.V1<-sum((dataset$Dense.t0-ram.m_Vu.V1)^2)
ram.sse.V.V1<-sum((dataset$Dense.t0-dataset$pred1.e)^2)
ram.ssr.V.V1<-sum((dataset$pred1.e-ram.m_Vu.V1)^2)
ram.tss.V1<-ram.sse.V.V1+ram.ssr.V.V1
ram.corr2.V1<-cor(dataset$Dense.t0,dataset$pred1.e)^2
ram.R2.V1<-1-(var(dataset$residuals1.e)/var(dataset$Dense.t0))
ram.mse.V1<-mean_squared_error(dataset$Dense.t0,dataset$pred1.e)
ram.mae.V1<-mean_absolute_error(dataset$Dense.t0,dataset$pred1.e)
ram.mase.V1<-mean_absolute_scaled_error(dataset$Dense.t0,dataset$pred1.e)
ram.rmse.V1<-rmse(dataset$Dense.t0,dataset$pred1.e)
ram.tvs.V1<-total_variance_score(dataset$Dense.t0,dataset$pred1.e)
ram.uvs.V1<-unexplained_variance_score(dataset$Dense.t0,dataset$pred1.e)
ram.evs.V1<-explained_variance_score(dataset$Dense.t0,dataset$pred1.e)

ram.m_Vu.V2<-mean(dataset$Hazy.t0)
ram.TSS.V2<-sum((dataset$Hazy.t0-ram.m_Vu.V2)^2)
ram.sse.V.V2<-sum((dataset$Hazy.t0-dataset$pred2.e)^2)

```

```

ram.ssr.V.V2<-sum((dataset$pred2.e-ram.m_Vu.V2)^2)
ram.tss.V2<-ram.sse.V.V2+ram.ssr.V.V2
ram.corr2.V2<-cor(dataset$Hazy.t0,dataset$pred2.e)^2
ram.R2.V2<-1-(var(dataset$residuals2.e)/var(dataset$Hazy.t0))
ram.mse.V2<-mean_squared_error(dataset$Hazy.t0,dataset$pred2.e)
ram.mae.V2<-mean_absolute_error(dataset$Hazy.t0,dataset$pred2.e)
ram.mase.V2<-mean_absolute_scaled_error(dataset$Hazy.t0,dataset$pred2.e)
ram.rmse.V2<-rmse(dataset$Hazy.t0,dataset$pred2.e)
ram.tvs.V2<-total_variance_score(dataset$Hazy.t0,dataset$pred2.e)
ram.uvs.V2<-unexplained_variance_score(dataset$Hazy.t0,dataset$pred2.e)
ram.evs.V2<-explained_variance_score(dataset$Hazy.t0,dataset$pred2.e)

dataset.t<-dataset[,1:38]
ran.est <- as.data.frame(matrix(NA,length(patch),7))
ran.est[,1] <- patch
for(i in 1:length(patch)){
  ran.est[i,2]<-(random.est[[i]]$ram.est.t$c.est)
  ran.est[i,3]<-(random.est[[i]]$ram.est.t$cc.t.est)
  ran.est[i,4]<-(random.est[[i]]$ram.est.t$d1.est)
  ran.est[i,5]<-(random.est[[i]]$ram.est.t$d2.est)
  ran.est[i,6]<-(random.est[[i]]$ram.est.t$ess1)
  ran.est[i,7]<-(random.est[[i]]$ram.est.t$ess2)
}
ran.est <- as.data.frame(ran.est)
colnames(ran.est) <- c("patch","c.est","cc.t.est","d1.est","d2.est",
"ess","ess2")
dataset.t<-merge(dataset.t,ran.est,by="patch",all.x = T)
dataset.t$logp.1 <- par.l$Beta[1]+ par.l$Beta[2]*dataset.t$Dose_med +
par.l$Beta[3] * dataset.t$years + par.l$Beta[4] * dataset.t$years2 +
dataset.t$c.est
dataset.t$odds.1 <- exp(dataset.t$logp.1)
dataset.t$p.1 <- dataset.t$odds.1 / (1 + dataset.t$odds.1)
dataset.t$subpred1 <- par.l$Gamma[1]+par.l$Gamma[2]*dataset.t$Dose_med +
par.l$Gamma[3] * dataset.t$years + par.l$Gamma[4] * dataset.t$years2 +
dataset.t$d1.est
dataset.t$subpred2 <- par.l$Gamma[5]+par.l$Gamma[6]*dataset.t$Dose_med +
par.l$Gamma[7] * dataset.t$years + par.l$Gamma[8] * dataset.t$years2 +
dataset.t$d2.est

dataset.t$subres1 <- dataset.t$Dense.lt0-dataset.t$subpred1
dataset.t$subres2 <- dataset.t$Hazy.lt0-dataset.t$subpred2
dataset.t$pred1<- dataset.t$p.1*dataset.t$subpred1
dataset.t$pred2<- dataset.t$p.1*dataset.t$subpred2
dataset.t$residuals1<-(dataset.t$Dense.lt0-dataset.t$pred1)
dataset.t$residuals2<-(dataset.t$Hazy.lt0-dataset.t$pred2)
dataset.t$sresiduals1<-dataset.t$residuals1^2
dataset.t$sresiduals2<-dataset.t$residuals2^2

```

```

duan1<- mean(exp(dataset.t[dataset.t$nonZeroHD==1,"subres1"]))
duan2<- mean(exp(dataset.t[dataset.t$nonZeroHD==1,"subres2"]))
dataset.t$pred1.e<- dataset.t$p.1*exp(dataset.t$subpred1)*duan1
dataset.t$pred2.e<- dataset.t$p.1*exp(dataset.t$subpred2)*duan2
dataset.t$residuals1.e<-(dataset.t$Dense.t0-dataset.t$pred1.e)
dataset.t$residuals2.e<-(dataset.t$Hazy.t0-dataset.t$pred2.e)

ram.t.m_Vu.V1<-mean(dataset.t$Dense.t0)
ram.t.TSS.V1<-sum((dataset.t$Dense.t0-ram.t.m_Vu.V1)^2)
ram.t.sse.V.V1<-sum((dataset.t$Dense.t0-dataset.t$pred1.e)^2)
ram.t.ssr.V.V1<-sum((dataset.t$pred1.e-ram.t.m_Vu.V1)^2)
ram.t.tss.V1<-ram.t.sse.V.V1+ram.t.ssr.V.V1
ram.t.corr2.V1<-cor(dataset.t$Dense.t0,dataset.t$pred1.e)^2
ram.t.R2.V1<-1-(var(dataset.t$residuals1.e)/var(dataset.t$Dense.t0))
ram.t.mse.V1<-mean_squared_error(dataset.t$Dense.t0,dataset.t$pred1.e)
ram.t.mae.V1<-mean_absolute_error(dataset.t$Dense.t0,dataset.t$pred1.e)
ram.t.mase.V1<-mean_absolute_scaled_error(dataset.t$Dense.t0,
      dataset.t$pred1.e)
ram.t.rmse.V1<-rmse(dataset.t$Dense.t0,dataset.t$pred1.e)
ram.t.tvs.V1<-total_variance_score(dataset.t$Dense.t0,dataset.t$pred1.e)
ram.t.uvs.V1<-unexplained_variance_score(dataset.t$Dense.t0,
      dataset.t$pred1.e)
ram.t.evs.V1<-explained_variance_score(dataset.t$Dense.t0,
      dataset.t$pred1.e)

ram.t.m_Vu.V2<-mean(dataset.t$Hazy.t0)
ram.t.TSS.V2<-sum((dataset.t$Hazy.t0-ram.t.m_Vu.V2)^2)
ram.t.sse.V.V2<-sum((dataset.t$Hazy.t0-dataset.t$pred2.e)^2)
ram.t.ssr.V.V2<-sum((dataset.t$pred2.e-ram.t.m_Vu.V2)^2)
ram.t.tss.V2<-ram.t.sse.V.V2+ram.t.ssr.V.V2
ram.t.corr2.V2<-cor(dataset.t$Hazy.t0,dataset.t$pred2.e)^2
ram.t.R2.V2<-1-(var(dataset.t$residuals2.e)/var(dataset.t$Hazy.t0))
ram.t.mse.V2<-mean_squared_error(dataset.t$Hazy.t0,dataset.t$pred2.e)
ram.t.mae.V2<-mean_absolute_error(dataset.t$Hazy.t0,dataset.t$pred2.e)
ram.t.mase.V2<-mean_absolute_scaled_error(dataset.t$Hazy.t0,
      dataset.t$pred2.e)
ram.t.rmse.V2<-rmse(dataset.t$Hazy.t0,dataset.t$pred2.e)
ram.t.tvs.V2<-total_variance_score(dataset.t$Hazy.t0,dataset.t$pred2.e)
ram.t.uvs.V2<-unexplained_variance_score(dataset.t$Hazy.t0,
      dataset.t$pred2.e)
ram.t.evs.V2<-explained_variance_score(dataset.t$Hazy.t0,dataset.t$pred2.e)

dataset.p<-dataset[,1:38]
ran.est <- as.data.frame(matrix(NA,length(patch),7))
ran.est[,1] <- patch
for(i in 1:length(patch)){

```



```

  ran.est[i,2]<-(random.est[[i]]$ram.est.p$c.est)
  ran.est[i,3]<-(random.est[[i]]$ram.est.p$cc.t.est)
  ran.est[i,4]<-(random.est[[i]]$ram.est.p$d1.est)
  ran.est[i,5]<-(random.est[[i]]$ram.est.p$d2.est)
  ran.est[i,6]<-(random.est[[i]]$ram.est.p$ess1)
  ran.est[i,7]<-(random.est[[i]]$ram.est.p$ess2)
}
ran.est <- as.data.frame(ran.est)
colnames(ran.est) <- c("patch","c.est","cc.t.est","d1.est","d2.est",
                      "ess","ess2")
dataset.p<-merge(dataset.p,ran.est,by="patch",all.x = T)

dataset.p$logp.1 <- par.l$Beta[1]+par.l$Beta[2]*dataset.p$Dose_med +
  par.l$Beta[3] * dataset.p$years + par.l$Beta[4] * dataset.p$years2 +
  dataset.p$c.est
dataset.p$odds.1 <- exp(dataset.p$logp.1)
dataset.p$p.1 <- dataset.p$odds.1 / (1 + dataset.p$odds.1)

dataset.p$subpred1 <- par.l$Gamma[1]+par.l$Gamma[2]*dataset.p$Dose_med +
  par.l$Gamma[3] * dataset.p$years + par.l$Gamma[4] * dataset.p$years2 +
  dataset.p$d1.est
dataset.p$subpred2 <- par.l$Gamma[5]+par.l$Gamma[6]*dataset.p$Dose_med +
  par.l$Gamma[7] * dataset.p$years + par.l$Gamma[8] * dataset.p$years2 +
  dataset.p$d2.est
dataset.p$subres1 <- dataset.p$Dense.lt0-dataset.p$subpred1
dataset.p$subres2 <- dataset.p$Hazy.lt0-dataset.p$subpred2
dataset.p$pred1<- dataset.p$p.1*dataset.p$subpred1
dataset.p$pred2<- dataset.p$p.1*dataset.p$subpred2
dataset.p$residuals1<-(dataset.p$Dense.lt0-dataset.p$pred1)
dataset.p$residuals2<-(dataset.p$Hazy.lt0-dataset.p$pred2)
dataset.p$sresiduals1<-dataset.p$residuals1^2
dataset.p$sresiduals2<-dataset.p$residuals2^2

duan1<- mean(exp(dataset.p[dataset.p$nonZeroHD==1,"subres1"]))
duan2<- mean(exp(dataset.p[dataset.p$nonZeroHD==1,"subres2"]))

dataset.p$pred1.e<- dataset.p$p.1*exp(dataset.p$subpred1)*duan1
dataset.p$pred2.e<- dataset.p$p.1*exp(dataset.p$subpred2)*duan2
dataset.p$residuals1.e<-(dataset.p$Dense.t0-dataset.p$pred1.e)
dataset.p$residuals2.e<-(dataset.p$Hazy.t0-dataset.p$pred2.e)

ram.p.m_Vu.V1<-mean(dataset.p$Dense.t0)
ram.p.TSS.V1<-sum((dataset.p$Dense.t0-ram.p.m_Vu.V1)^2)
ram.p.sse.V.V1<-sum((dataset.p$Dense.t0-dataset.p$pred1.e)^2)
ram.p.ssr.V.V1<-sum((dataset.p$pred1.e-ram.p.m_Vu.V1)^2)
ram.p.tss.V1<-ram.p.sse.V.V1+ram.p.ssr.V.V1
ram.p.corr2.V1<-cor(dataset.p$Dense.t0,dataset.p$pred1.e)^2

```

```

ram.p.R2.V1<-1-(var(dataset.p$residuals1.e)/var(dataset.p$Dense.t0))
ram.p.mse.V1<-mean_squared_error(dataset.p$Dense.t0,dataset.p$pred1.e)
ram.p.mae.V1<-mean_absolute_error(dataset.p$Dense.t0,dataset.p$pred1.e)
ram.p.mase.V1<-mean_absolute_scaled_error(dataset.p$Dense.t0,
      dataset.p$pred1.e)
ram.p.rmse.V1<-rmse(dataset.p$Dense.t0,dataset.p$pred1.e)
ram.p.tvs.V1<-total_variance_score(dataset.p$Dense.t0,dataset.p$pred1.e)
ram.p.uvs.V1<-unexplained_variance_score(dataset.p$Dense.t0,
      dataset.p$pred1.e)
ram.p.evs.V1<-explained_variance_score(dataset.p$Dense.t0,
      dataset.p$pred1.e)

ram.p.m_Vu.V2<-mean(dataset.p$Hazy.t0)
ram.p.TSS.V2<-sum((dataset.p$Hazy.t0-ram.p.m_Vu.V2)^2)
ram.p.sse.V.V2<-sum((dataset.p$Hazy.t0-dataset.p$pred2.e)^2)
ram.p.ssr.V.V2<-sum((dataset.p$pred2.e-ram.p.m_Vu.V2)^2)
ram.p.tss.V2<-ram.p.sse.V.V2+ram.p.ssr.V.V2
ram.p.corr2.V2<-cor(dataset.p$Hazy.t0,dataset.p$pred2.e)^2
ram.p.R2.V2<-1-(var(dataset.p$residuals2.e)/var(dataset.p$Hazy.t0))
ram.p.mse.V2<-mean_squared_error(dataset.p$Hazy.t0,dataset.p$pred2.e)
ram.p.mae.V2<-mean_absolute_error(dataset.p$Hazy.t0,dataset.p$pred2.e)
ram.p.mase.V2<-mean_absolute_scaled_error(dataset.p$Hazy.t0,
      dataset.p$pred2.e)
ram.p.rmse.V2<-rmse(dataset.p$Hazy.t0,dataset.p$pred2.e)
ram.p.tvs.V2<-total_variance_score(dataset.p$Hazy.t0,dataset.p$pred2.e)
ram.p.uvs.V2<-unexplained_variance_score(dataset.p$Hazy.t0,
      dataset.p$pred2.e)
ram.p.evs.V2<-explained_variance_score(dataset.p$Hazy.t0,dataset.p$pred2.e)

total.end<-Sys.time()
total.end-total.inicio
save.image(filepred)

# Results -----
library(compositions)
library(ggtern)
library(png)
library(dplyr)
library(mvnTest)

### Scaled residuals for U #####
agg<-dataset.p[,c("nonZeroHD","Dense.lt","Hazy.lt","Dense.lt0",
      "Hazy.lt0","p.1","subpred1","subpred2","pred1",
      "pred2","patch")]

agg2<- agg %>%

```

```

    group_by(patch) %>%
    summarise_each(sum)
agg2$scaled.u.res<-(agg2$nonZeroHD-agg2$p.1)/sqrt(agg2$p.1 *
          (5-agg2$p.1) /5)
agg2$scaled.u.res.ct<-ifelse(abs(agg2$scaled.u.res)<2.5,0,1)
summary(agg2$scaled.u.res)
table(agg2$scaled.u.res.ct)
prop.table(table(agg2$scaled.u.res.ct))
table(agg2$scaled.u.res.ct,agg2$nonZeroHD)

### Scaled residuals for v for those with at least one nonzero #####
agg3<-agg2[agg2$nonZeroHD!=0,]
agg3$scaled.v1.res<-(agg3$Dense.lt0 - agg3$subpred1)/
          sqrt(agg3$nonZeroHD*par.l$Var.V1)
agg3$scaled.v2.res<-(agg3$Hazy.lt0 - agg3$subpred2)/
          sqrt(agg3$nonZeroHD*par.l$Var.V2 )
agg3$scaled.v1.res.ct<-ifelse(abs(agg3$scaled.v1.res)<2.5,0,1)
agg3$scaled.v2.res.ct<-ifelse(abs(agg3$scaled.v2.res)<2.5,0,1)
table(agg3$scaled.v1.res.ct)
prop.table(table(agg3$scaled.v1.res.ct))
table(agg3$scaled.v1.res.ct,agg3$nonZeroHD)
table(agg3$scaled.v2.res.ct)
prop.table(table(agg3$scaled.v2.res.ct))
table(agg3$scaled.v2.res.ct,agg3$nonZeroHD)

### Residuals #####
v <- dataset.p[order(dataset.p$patch, dataset.p$date),]
v <- v[!duplicated(v$patch),]
test<-qqnorm(v$c.est, pch = 1, frame = FALSE,col = "#926b8d")
x <- c(quantile(test$x,0.25),quantile(test$x,0.75))
y <- c(quantile(log(test$y+(abs(min(test$y))+1)),0.25),
       quantile(log(test$y+(abs(min(test$y))+1)),0.75))
slope <- diff(y)/diff(x)
intercept <- y[1]-slope*x[1]
plot(test$x,log(test$y+(abs(min(test$y))+1)),col = "#926b8d")
la<-c(0,1,2,3,4) #cubic
la<-c(0,1,2,3,4,5) #spherical
la<-c(0,1,2,3,4) #iso
la2<-round(exp(la)-1-abs(min(test$y)),0)
plot(test$x,log(test$y+(abs(min(test$y))+1)),col = "#926b8d",
      xlab="Theoretical Q-Quantiles", ylab="Sample Quantiles", yaxt="n",
      main="Normal Q-Q Plot")
abline(intercept, slope, col="#B7B6BA", lwd = 2)
axis(2,at=la,labels=la2)

x <- c(quantile(test$x,0.25),quantile(test$x,0.75))
y <- c(quantile(sqrt(test$y+(abs(min(test$y))))),0.25),

```

```

        quantile(sqrt(test$y+(abs(min(test$y)))),0.75))
slope <- diff(y)/diff(x)
intercept <- y[1]-slope*x[1]
plot(test$x,sqrt(test$y+(abs(min(test$y)))),col = "#926b8d",
      xlab="Theoretical Quantiles", ylab="Sample Quantiles",
      main="Normal Q-Q Plot")
la<-c(0,2,4,6,8)
la<-c(0,2,4,6,8,10,12,14)
la<-c(0,2,4,6,8)
la2<-round((la^2)-1-abs(min(test$y)),0)
plot(test$x,sqrt(test$y+(abs(min(test$y)))),col = "#926b8d",
      xlab="Theoretical Quantiles", ylab="Sample Quantiles", yaxt="n",
      main="Normal Q-Q Plot")
abline(intercept, slope, col="#B7B6BA", lwd = 2)
axis(2,at=la,labels=la2)

x <- c(quantile(test$x,0.25),quantile(test$x,0.75))
y <- c(quantile(sign(test$y)*(abs(test$y)^(1/3)),0.25),
      quantile(sign(test$y)*(abs(test$y)^(1/3)),0.75))
slope <- diff(y)/diff(x)
intercept <- y[1]-slope*x[1]
plot(test$x,sign(test$y)*(abs(test$y)^(1/3)),col = "#926b8d",
      xlab="Theoretical Quantiles", ylab="Sample Quantiles",
      main="Normal Q-Q Plot")
la<-c(-1,0,1,2,3,4)
la<-c(-4,-2,0,2,4)
la<- c(-4,-3,-2,-1,0,1)
la2<-round((la^3),0)
plot(test$x,sign(test$y)*(abs(test$y)^(1/3)),col = "#926b8d",
      xlab="Theoretical Quantiles", ylab="Sample Quantiles", yaxt="n",
      main="Normal Q-Q Plot")
abline(intercept, slope, col="#B7B6BA", lwd = 2)
axis(2,at=la,labels=la2)

x <- c(quantile(test$x,0.25),quantile(test$x,0.75))
y <- c(quantile(sign(test$y)*(abs(test$y)^(1/5)),0.25),
      quantile(sign(test$y)*(abs(test$y)^(1/5)),0.75))
slope <- diff(y)/diff(x)
intercept <- y[1]-slope*x[1]
plot(test$x,sign(test$y)*(abs(test$y)^(1/5)),col = "#926b8d",
      xlab="Theoretical Quantiles", ylab="Sample Quantiles",
      main="Normal Q-Q Plot")
la<-c(-1,0,1,2)
la<-c(-2,-1,0,1,2)
la<-c(-2,-1,0,1)
la2<-round((la^5),0)
plot(test$x,sign(test$y)*(abs(test$y)^(1/5)),col = "#926b8d",

```

```

        xlab="Theoretical Quantiles", ylab="Sample Quantiles", yaxt="n",
        main="Normal Q-Q Plot")
abline(intercept, slope, col="#B7B6BA", lwd = 2)
axis(2, at=1a, labels=1a2)

qqnorm(v[v$Com!=1,]$c.est, pch = 1, frame = FALSE, col = "#926b8d")
qqline(v[v$Com!=1,]$c.est, col = "#B7B6BA", lwd = 2)

qqnorm(v[v$Com==1,]$c.est, pch = 1, frame = FALSE, col = "#926b8d")
qqline(v[v$Com==1,]$c.est, col = "#B7B6BA", lwd = 2)
boxplot(c.est~nonZeroHD, data=v)

qqnorm(dataset.t$subres1, pch = 1, frame = FALSE)
qqline(dataset.t$subres1, col = "steelblue", lwd = 2)
qqnorm(dataset.t$subres2, pch = 1, frame = FALSE)
qqline(dataset.t$subres2, col = "steelblue", lwd = 2)

qqnorm(dataset.t$residuals1, pch = 1, frame = FALSE)
qqline(dataset.t$residuals1, col = "steelblue", lwd = 2)
qqnorm(dataset.t$residuals2, pch = 1, frame = FALSE)
qqline(dataset.t$residuals2, col = "steelblue", lwd = 2)

AD.test(dataset.t[,c("subres1","subres2")], qqplot=T)
CM.test(dataset.t[,c("subres1","subres2")], qqplot=F)
R.test(dataset.t[,c("subres1","subres2")], qqplot=F)

AD.test(dataset.t[,c("residuals1","residuals2")], qqplot=T)
CM.test(dataset.t[,c("residuals1","residuals2")], qqplot=F)
R.test(dataset.t[,c("residuals1","residuals2")], qqplot=F)

### Additive logistic transform #####
alrInv(c(par.l$Gamma[[1]],par.l$Gamma[[5]]))
ef.dose<-c(par.l$Gamma[[2]],par.l$Gamma[[6]])
alrInv(ef.dose)
sqrt(ef.dose %*% solve(diag(2) + c(1,1) %*% t(c(1,1))) %*% (ef.dose))
ef.time<-c(par.l$Gamma[[3]],par.l$Gamma[[7]])
alrInv(ef.time)
sqrt(ef.time %*% solve(diag(2) + c(1,1) %*% t(c(1,1))) %*% (ef.time))
ef.time2<-c(par.l$Gamma[[4]],par.l$Gamma[[8]])
alrInv(ef.time2)
sqrt(ef.time2 %*% solve(diag(2) + c(1,1) %*% t(c(1,1))) %*% (ef.time2))

### Goodness of fit #####
v2<-dataset.p
v2$e.subpred.f1.v2<-exp(v2$subpred1)*exp(0.5*par.l$Var.V1)
v2$e.subpred.f2.v2<-exp(v2$subpred2)*exp(0.5*par.l$Var.V2)
v2$e.pred.f1<-v2$p.1*v2$e.subpred.f1

```

```

v2$e.pred.f2<-v2$p.1*v2$e.subpred.f2
v2$dense.pred.f<-v2$e.pred.f1/(1+v2$e.pred.f1+v2$e.pred.f2)
v2$hazy.pred.f<-v2$e.pred.f2/(1+v2$e.pred.f1+v2$e.pred.f2)
v2$com.pred.f<-1/(1+v2$e.pred.f1+v2$e.pred.f2)
v2[,c("dense.pred.f","hazy.pred.f","com.pred.f")]<-clo(v2[,
      c("dense.pred.f","hazy.pred.f","com.pred.f")])

v2[,c("dense.res.f")]<-v2$Dense-v2$dense.pred.f
v2[,c("hazy.res.f")]<-v2$Hazy-v2$hazy.pred.f
v2[,c("com.res.f")]<-v2$Com-v2$com.pred.f

sum(v2$dense.res.f^2 )
sum(v2$hazy.res.f^2 )
sum(v2$com.res.f^2 )
sum(v2$dense.res.f^2 ) + sum(v2$hazy.res.f^2 ) +
  sum(v2$com.res.f^2 )
sqrt(sum(v2$dense.res.f^2 ) + sum(v2$hazy.res.f^2 ) +
      sum(v2$com.res.f^2 ))

v2[,c("dense.res.p1")]<-v2$Dense-v2$dense.pred
v2[,c("hazy.res.p1")]<-v2$Hazy-v2$hazy.pred
v2[,c("com.res.p1")]<-v2$Com-v2$com.pred

sum(v2$dense.res.p1^2 )
sum(v2$hazy.res.p1^2 )
sum(v2$com.res.p1^2 )
sum(v2$dense.res.p1^2 ) + sum(v2$hazy.res.p1^2 ) +
  sum(v2$com.res.p1^2 )
sqrt(sum(v2$dense.res.p1^2 ) + sum(v2$hazy.res.p1^2 ) +
      sum(v2$com.res.p1^2 ))

v2[,c("dense.res.p2")]<-v2$Dense-v2$dense.pred2
v2[,c("hazy.res.p2")]<-v2$Hazy-v2$hazy.pred2
v2[,c("com.res.p2")]<-v2$Com-v2$com.pred2

sum(v2$dense.res.p2^2 )
sum(v2$hazy.res.p2^2 )
sum(v2$com.res.p2^2 )
sum(v2$dense.res.p2^2 ) + sum(v2$hazy.res.p2^2 ) +
  sum(v2$com.res.p2^2 )
sqrt(sum(v2$dense.res.p2^2 ) + sum(v2$hazy.res.p2^2 ) +
      sum(v2$com.res.p2^2 ))

# Figures =====
#Figures 5.1 and 5.2 are in results/residuals

#Figures 5.3

```

```

#Standardized residuals
dataset.p$subres1.st<-dataset.p$subres1/sqrt(par.l$Var.V1)
dataset.p$subres2.st<-dataset.p$subres2/sqrt(par.l$Var.V2)
dataset.p$residuals1.st<-dataset.p$residuals1/sqrt(par.l$Var.V1)
dataset.p$residuals2.st<-dataset.p$residuals2/sqrt(par.l$Var.V2)

ggplot(dataset.p, aes(x=Dose_med, y=subres1.st)) +
  geom_point(alpha=0.7,col="#926b8d") +
  xlab("Median_Dose") + ylab("Standardized_residual_alr(Dense)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =
    element_line(colour = "black"),text = element_text(size=16),
    legend.position="bottom",legend.title = element_blank())

ggplot(dataset.p, aes(x=Dose_med, y=residuals1.st)) +
  geom_point(alpha=0.7,col="#926b8d") +
  xlab("Median_Dose") + ylab("Standardized_residual_alr(Dense)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =
    element_line(colour = "black"),text = element_text(size=16),
    legend.position="bottom",legend.title = element_blank())

ggplot(dataset.p, aes(x=Dose_med, y=subres2.st)) +
  geom_point(alpha=0.7,col="#926b8d") +
  xlab("Median_Dose") + ylab("Standardized_residual_alr(Hazy)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =
    element_line(colour = "black"),text = element_text(size=16),
    legend.position="bottom",legend.title = element_blank())

ggplot(dataset.p, aes(x=Dose_med, y=residuals2.st)) +
  geom_point(alpha=0.7,col="#926b8d") +
  xlab("Median_Dose") + ylab("Standardized_residual_alr(Hazy)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =
    element_line(colour = "black"),text = element_text(size=16),
    legend.position="bottom",legend.title = element_blank())

ggplot(dataset.p, aes(x=factor(date2), y=subres1.st)) +
  geom_point(alpha=0.7,col="#926b8d") +
  xlab("Months_after_RT") + ylab("Standardized_residual_alr(Dense)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),

```

```

    panel.background = element_blank(), axis.line =
      element_line(colour = "black"), text = element_text(size=16),
    legend.position="bottom", legend.title = element_blank())

ggplot(dataset.p, aes(x=factor(date2), y=residuals1.st)) +
  geom_point(alpha=0.7, col="#926b8d") +
  xlab("Months after RT") + ylab("Standardized residual - alr(Dense)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =
      element_line(colour = "black"), text = element_text(size=16),
    legend.position="bottom", legend.title = element_blank())

ggplot(dataset.p, aes(x=factor(date2), y=subres2.st)) +
  geom_point(alpha=0.7, col="#926b8d") +
  xlab("Months after RT") + ylab("Standardized residual - alr(Hazy)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =
      element_line(colour = "black"), text = element_text(size=16),
    legend.position="bottom", legend.title = element_blank())

ggplot(dataset.p, aes(x=factor(date2), y=residuals2.st)) +
  geom_point(alpha=0.7, col="#926b8d") +
  xlab("Months after RT") + ylab("Standardized residual - alr(Hazy)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =
      element_line(colour = "black"), text = element_text(size=16),
    legend.position="bottom", legend.title = element_blank())

#Figure 5.5
ggplot(dataset.p, aes(x=logp.1, y=subpred1)) +
  geom_point(alpha=0.7, col="#926b8d") +
  xlab("Logit-probability of RILD") + ylab("Mean alr(Dense)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =
      element_line(colour = "black"), text = element_text(size=16),
    legend.position="bottom", legend.title = element_blank())

ggplot(dataset.p, aes(x=logp.1, y=subpred2)) +
  geom_point(alpha=0.7, col="#926b8d") +
  xlab("Logit-probability of RILD") + ylab("Mean alr(Hazy)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =

```



```

    element_line(colour = "black"),text = element_text(size=16),
    legend.position="bottom",legend.title = element_blank())

```

#Figure 5.6

```

ggplot(agg2, aes(x=nonZeroHD, y=p.1)) +
  geom_point(pch = 19, position = position_jitter(),alpha=0.3,
            size=1,col="#B7B6BA") +
  geom_point(alpha=0.7,col="#926b8d") +
  xlab("Observed") + ylab("Predicted")+
  scale_x_continuous(breaks=c(0,1,2,3,4,5),labels=c(0,1,2,3,4,5)) +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
        element_blank(), panel.grid.minor = element_blank(),
        panel.background = element_blank(), axis.line =
        element_line(colour = "black"),text = element_text(size=16),
        legend.position="bottom",legend.title = element_blank())

```

#Figure 5.7

```

ggplot(dataset.p, aes(x=Dense.lt0, y=subpred1)) +
  geom_point(alpha=0.7,col="#926b8d") +
  geom_abline(slope=1, intercept=0,col="#B7B6BA") +
  xlab("Observed_alr(Dense)") + ylab("Sub-predicted_alr(Dense)")+
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
        element_blank(), panel.grid.minor = element_blank(),
        panel.background = element_blank(), axis.line =
        element_line(colour = "black"),text = element_text(size=16),
        legend.position="bottom",legend.title = element_blank())

```

```

ggplot(dataset.p, aes(x=Hazy.lt0, y=subpred2)) +
  geom_point(alpha=0.7,col="#926b8d") +
  geom_abline(slope=1, intercept=0,col="#B7B6BA") +
  xlab("Observed_alr(Hazy)") + ylab("Sub-predicted_alr(Hazy)")+
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
        element_blank(), panel.grid.minor = element_blank(),
        panel.background = element_blank(), axis.line =
        element_line(colour = "black"),text = element_text(size=16),
        legend.position="bottom",legend.title = element_blank())

```

```

ggplot(dataset.p, aes(x=Dense.lt0, y=pred1)) +
  geom_point(alpha=0.7,col="#926b8d") +
  geom_abline(slope=1, intercept=0,col="#B7B6BA") +
  xlab("Observed_alr(Dense)") + ylab("Predicted_alr(Dense)")+
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
        element_blank(), panel.grid.minor = element_blank(),
        panel.background = element_blank(), axis.line =
        element_line(colour = "black"),text = element_text(size=16),
        legend.position="bottom",legend.title = element_blank())

```

```

ggplot(dataset.p, aes(x=Hazy.lt0, y=pred2)) +
  geom_point(alpha=0.7, col="#926b8d") +
  geom_abline(slope=1, intercept=0, col="#B7B6BA") +
  xlab("Observed alr(Hazy)") + ylab("Predicted alr(Hazy)") +
  theme(plot.title = element_text(hjust = 0.5), panel.grid.major =
    element_blank(), panel.grid.minor = element_blank(),
    panel.background = element_blank(), axis.line =
    element_line(colour = "black"), text = element_text(size=16),
    legend.position="bottom", legend.title = element_blank())

```

#Figure 5.8

```

v<-dataset.p
v$e.pred.f1<-exp(v$pred1)
v$e.pred.f2<-exp(v$pred2)
v$dense.pred.f<-v$e.pred.f1/(1+v$e.pred.f1+v$e.pred.f2)
v$hazy.pred.f<-v$e.pred.f2/(1+v$e.pred.f1+v$e.pred.f2)
v$com.pred.f<-1/(1+v$e.pred.f1+v$e.pred.f2)
v[,c("dense.pred.f", "hazy.pred.f", "com.pred.f")]<-c1o(v[,
  c("dense.pred.f", "hazy.pred.f", "com.pred.f")])

a<-c(3,6,12,18,24)
k<-a[1]
x<-acomp(v[v$date==k, c("dense.pred.f", "hazy.pred.f", "com.pred.f")])
xm<-(mean(x))
ggtern(data=v[v$date==k, c("dense.pred.f", "hazy.pred.f", "com.pred.f")],
  aes(x=dense.pred.f, y=hazy.pred.f, z=com.pred.f)) +
  geom_point() +
  geom_point(aes(x=xm[1], y=xm[2], z=xm[3], colour="Mean")) +
  scale_colour_manual(name = element_blank(), labels =
    "Mean\ncomposition", values=c("#7aadb1")) +
  xlab("Dense") + ylab("Hazy") + zlab("Normal") +
  theme_showarrows() + ggtitle(paste(k, "\u00A0Months", sep = "")) +
  theme(plot.title = element_text(face="bold", hjust = 0.5, vjust=8),
    plot.margin = unit(c(0,0,0,0), "cm"))

```

#Figure 5.9

```

ef.dose.data<-as.data.frame(rep(1:60,5))
colnames(ef.dose.data)<-c("Dose")
ef.dose.data$lDose<-log(ef.dose.data$Dose)
ef.dose.data$month<-rep(c(3,6,12,18,24), each=60)
ef.dose.data$time<-ef.dose.data$month/12
ef.dose.data$log<-par.l$Beta[[1]] + (ef.dose.data$lDose *
  par.l$Beta[[2]]) + (ef.dose.data$time * par.l$Beta[[3]]) +
  ((ef.dose.data$time^2) * par.l$Beta[[4]])
ef.dose.data$p.1<-exp(ef.dose.data$log)/(1+exp(ef.dose.data$log))
ef.dose.data$subpred1<-par.l$Gamma[[1]] + (ef.dose.data$lDose *
  par.l$Gamma[[2]]) + (ef.dose.data$time * par.l$Gamma[[3]])+

```

```

      ((ef.dose.data$time^2) * par.l$Gamma[[4]])
ef.dose.data$subpred2<-par.l$Gamma[[5]] + (ef.dose.data$lDose *
      par.l$Gamma[[6]]) + (ef.dose.data$time * par.l$Gamma[[7]]) +
      ((ef.dose.data$time^2) * par.l$Gamma[[8]])
ef.dose.data$e.subpred.f1<-exp(ef.dose.data$subpred1)*
      exp(0.5*par.l$Var.V1)
ef.dose.data$e.subpred.f2<-exp(ef.dose.data$subpred2)*
      exp(0.5*par.l$Var.V2)
ef.dose.data$e.pred.f1<-ef.dose.data$p.1*ef.dose.data$e.subpred.f1
ef.dose.data$e.pred.f2<-ef.dose.data$p.1*ef.dose.data$e.subpred.f2
ef.dose.data$dense.pred.f<-ef.dose.data$e.pred.f1/
      (1+ef.dose.data$e.pred.f1+ef.dose.data$e.pred.f2)
ef.dose.data$hazy.pred.f<-ef.dose.data$e.pred.f2/
      (1+ef.dose.data$e.pred.f1+ef.dose.data$e.pred.f2)
ef.dose.data$com.pred.f<-1/(1+ef.dose.data$e.pred.f1+
      ef.dose.data$e.pred.f2)
ef.dose.data[,c("dense.pred.f", "hazy.pred.f", "com.pred.f")]<-
      clo(ef.dose.data[,c("dense.pred.f", "hazy.pred.f", "com.pred.f")])
ef.dose.data$t<-ef.dose.data$dense.pred.f+ef.dose.data$hazy.pred.f+
      ef.dose.data$com.pred.f

ggtern(data=ef.dose.data[ef.dose.data$month==24,],
      aes(x=dense.pred.f, y=hazy.pred.f, z=com.pred.f, color=Dose)) +
  geom_point()+
  geom_point(aes(x=b[1], y=b[2], z=b[3]), colour = "#7aadb1") +
  scale_color_gradient(breaks= c(10,20,30,40,50,60), labels=
      c(10,20,30,40,50,60), limits=c(0,60),
      low = "#926b8d" , high = "#E8CA47") +
  labs(fill = "Median_dose") + xlab("Dense") + ylab("Hazy") +
  zlab("Normal") +
  theme_showarrows() + ggtitle(paste(k, "_Months", sep = "")) +
  theme(plot.title = element_text(face="bold", hjust = 0.5,vjust=8),
      plot.margin = unit(c(0,0,0,0), "cm"))

```

B.5 Chapter 6: Cluster analysis on reference patienter data

Also available at: <https://github.com/vivifj03/TumorModel/blob/master/Code%20chapter6%20-%20Clusters.R>

```

# Packages -----
library(compositions)
library(cluster)
library(tidyr)
library(ggplot2)
library(gridExtra)
library(pspline)

```

```

# Data -----
setwd("")
load(".RData") #Data set with prediction (v2 in code Chapter 5)
#v2 have the final predicted compositions

dcluster<-v2[,c("patch","Id","date","Hazy","Dense","Com","chageID",
              "someDose" ,"nonZeroHD","Dose_avg","Dose_min", "Dose_max",
              "Dose_sd","Dose_med","dense.pred.f",
              "hazy.pred.f","com.pred.f")]

patch2<-v2[!duplicated(v2$patch),c("patch","chageID") ]
rm(list=setdiff(ls(),c("v2","dcluster","patch2")))

dcluster[c("dense.p","hazy.p","com.p")]<-clo(dcluster[c("dense.pred.f",
              "hazy.pred.f","com.pred.f")])
dcluster[,c("dense.p.clr","hazy.p.clr","com.p.clr")]<-
              clr(dcluster[c("dense.p","hazy.p","com.p")])

# Clustering -----
ni<-nrow(dcluster)/length(patch2)
df_list <- split(dcluster, as.factor(dcluster$date))
df_list$`3`<-df_list$`3`[,c("patch","dense.p","hazy.p","com.p",
                          "dense.p.clr","hazy.p.clr","com.p.clr")]
df_list$`6`<-df_list$`6`[,c("patch","dense.p","hazy.p","com.p",
                          "dense.p.clr","hazy.p.clr","com.p.clr")]
df_list$`12`<-df_list$`12`[,c("patch","dense.p","hazy.p","com.p",
                              "dense.p.clr","hazy.p.clr","com.p.clr")]
df_list$`18`<-df_list$`18`[,c("patch","dense.p","hazy.p","com.p",
                              "dense.p.clr","hazy.p.clr","com.p.clr")]
df_list$`24`<-df_list$`24`[,c("patch","dense.p","hazy.p","com.p",
                              "dense.p.clr","hazy.p.clr","com.p.clr")]

# Euclidian distance =====
rownames(df_list$`3`)<-df_list$`3`$patch
rownames(df_list$`6`)<-df_list$`6`$patch
rownames(df_list$`12`)<-df_list$`12`$patch
rownames(df_list$`18`)<-df_list$`18`$patch
rownames(df_list$`24`)<-df_list$`24`$patch
d3<-dist(df_list$`3`[,5:7], method = "euclidean")
d6<-dist(df_list$`6`[,5:7], method = "euclidean")
d12<-dist(df_list$`12`[,5:7], method = "euclidean")
d18<-dist(df_list$`18`[,5:7], method = "euclidean")
d24<-dist(df_list$`24`[,5:7], method = "euclidean")
dis<-(d3+d6+d12+d18+d24)/5

c2<-pam(dis,2,diss=T)

```

```

c3<-pam(dis ,3,diss=T)
c4<-pam(dis ,4,diss=T)
c5<-pam(dis ,5,diss=T)
c6<-pam(dis ,6,diss=T)
c7<-pam(dis ,7,diss=T)

patch_c<-cbind(patch2,c2$clustering)
patch_c<-cbind(patch_c,c3$clustering)
patch_c<-cbind(patch_c,c4$clustering)
patch_c<-cbind(patch_c,c5$clustering)
patch_c<-cbind(patch_c,c6$clustering)
patch_c<-cbind(patch_c,c7$clustering)
patch_c<-merge(patch_c,v2[v2$date==3,c("patch","Dose_med")],by="patch")
colnames(patch_c)<-c("patch","changeID" ,"c2","c3","c4","c5","c6","c7",
"Dose")
patch_c<-as.data.frame(patch_c)

# Table 6.1
table(patch_c$changeID,patch_c$c2)
table(patch_c$changeID,patch_c$c3)
table(patch_c$changeID,patch_c$c4)
table(patch_c$changeID,patch_c$c5)
prop.table(table(patch_c$changeID,patch_c$c5),1)
table(patch_c$changeID,patch_c$c6)
prop.table(table(patch_c$changeID,patch_c$c6),1)
table(patch_c$changeID,patch_c$c7)
prop.table(table(patch_c$changeID,patch_c$c7),1)
dcluster<-merge(dcluster,patch_c,by="patch",all.x = T)

#Average silhouette coefficient
plot(c2,main="□")
plot(c3)
plot(c4)
plot(c5)
plot(c6)
plot(c7)

# Figure 6.1
#Peter J. Rousseeuw (1987). "Silhouettes: a Graphical Aid to the
#Interpretation and Validation of Cluster Analysis". Computational
#and Applied Mathematics. 20: 53-65. doi:10.1016/0377-0427(87)90125-7.
x<-c(2,3,4,5,6,7)
y<-c(0.75,0.58,0.52,0.57,0.52,0.51)
dxy<-as.data.frame(cbind(x,y))
ggplot(dxy,aes(x = x,y = y)) +
  geom_line(col="#926b8d") + geom_point(col="#926b8d")+
  xlab("Number□of□clusters") + ylab("Average□Silhouette□Score")+

```

```

#Figure 6.2
#replace c2 for c3-c7
ggplot(data = patch_c, aes(x=factor(c2), y=Dose)) +
  geom_boxplot(alpha=0.7) +
  geom_point(pch = 19, position = position_jitterdodge(),
            alpha=0.7, size=1, aes(colour = factor(changeID)))+
  xlab("Cluster") + ylab("Median_Dose") +
  scale_color_manual(values = c("#B7B6BA", "#926B8D"),
                    labels = c("No_RILD", "RILD"), name="")

#Figure 6.3
# for Figure 6.4 replace dense.pred.f, hazy.pred.f, com.pred.f, for
# Dense, Hazy, Com
cluster.labs <- c("Cluster_1", "Cluster_2")
names(cluster.labs) <- c("1", "2")
ggplot(dcluster, aes(x=factor(date), y=dense.pred.f)) +
  geom_boxplot(alpha=0.7, color="#926B8D")+
  facet_wrap(~c2, labeller = labeller(c2 = cluster.labs)) +
  xlab("Months_after_RT") + ylab("Predicted_Dense_%") + ylim(0,1)

ggplot(dcluster, aes(x=factor(date), y=hazy.pred.f)) +
  geom_boxplot(alpha=0.7, color="#7aadb1")+
  facet_wrap(~c2, labeller = labeller(c2 = cluster.labs)) +
  xlab("Months_after_RT") + ylab("Predicted_Hazy_%") + ylim(0,1)

ggplot(dcluster, aes(x=factor(date), y=com.pred.f)) +
  geom_boxplot(alpha=0.7, color="#B7B6BA")+
  facet_wrap(~c2, labeller = labeller(c2 = cluster.labs)) +
  xlab("Months_after_RT") + ylab("Predicted_Normal_%") + ylim(0,1)

cluster.labs <- c("Cluster_1", "Cluster_2", "Cluster_3")
names(cluster.labs) <- c("1", "2", "3")
ggplot(dcluster, aes(x=factor(date), y=dense.pred.f)) +
  geom_boxplot(alpha=0.7, color="#926B8D")+
  facet_wrap(~c3, labeller = labeller(c3 = cluster.labs)) +
  xlab("Months_after_RT") + ylab("Predicted_Dense_%") + ylim(0,1)

ggplot(dcluster, aes(x=factor(date), y=hazy.pred.f)) +
  geom_boxplot(alpha=0.7, color="#7aadb1")+
  facet_wrap(~c3, labeller = labeller(c3 = cluster.labs)) +
  xlab("Months_after_RT") + ylab("Predicted_Hazy_%") + ylim(0,1)

ggplot(dcluster, aes(x=factor(date), y=com.pred.f)) +
  geom_boxplot(alpha=0.7, color="#B7B6BA")+
  facet_wrap(~c3, labeller = labeller(c3 = cluster.labs)) +
  xlab("Months_after_RT") + ylab("Predicted_Normal_%") + ylim(0,1)

```

```

cluster.labs <- c("Cluster_1", "Cluster_2", "Cluster_3", "Cluster_4")
names(cluster.labs) <- c("1", "2", "3", "4")
ggplot(dcluster, aes(x=factor(date), y=dense.pred.f)) +
  geom_boxplot(alpha=0.7, color="#926B8D")+
  facet_wrap(~c4, labeller = labeller(c4 = cluster.labs), nrow = 1) +
  xlab("Months_after_RT") + ylab("Predicted_Dense_%") + ylim(0,1)

ggplot(dcluster, aes(x=factor(date), y=hazy.pred.f)) +
  geom_boxplot(alpha=0.7, color="#7aadb1")+
  facet_wrap(~c4, labeller = labeller(c4 = cluster.labs), nrow = 1) +
  xlab("Months_after_RT") + ylab("Predicted_Hazy_%") + ylim(0,1)

ggplot(dcluster, aes(x=factor(date), y=com.pred.f)) +
  geom_boxplot(alpha=0.7, color="#B7B6BA")+
  facet_wrap(~c4, labeller = labeller(c4 = cluster.labs), nrow = 1)+
  xlab("Months_after_RT") + ylab("Predicted_Normal_%") + ylim(0,1)

cluster.labs <- c("Cluster_1", "Cluster_2", "Cluster_3", "Cluster_4",
                 "Cluster_5")
names(cluster.labs) <- c("1", "2", "3", "4", "5")
ggplot(dcluster, aes(x=factor(date), y=dense.pred.f)) +
  geom_boxplot(alpha=0.7, color="#926B8D")+
  facet_wrap(~c5, labeller = labeller(c5 = cluster.labs), nrow = 1) +
  xlab("Months_after_RT") + ylab("Predicted_Dense_%") + ylim(0,1)

ggplot(dcluster, aes(x=factor(date), y=hazy.pred.f)) +
  geom_boxplot(alpha=0.7, color="#7aadb1")+
  facet_wrap(~c5, labeller = labeller(c5 = cluster.labs), nrow = 1) +
  xlab("Months_after_RT") + ylab("Predicted_Hazy_%") + ylim(0,1)

ggplot(dcluster, aes(x=factor(date), y=com.pred.f)) +
  geom_boxplot(alpha=0.7, color="#B7B6BA")+
  facet_wrap(~c5, labeller = labeller(c5 = cluster.labs), nrow = 1)+
  xlab("Months_after_RT") + ylab("Predicted_Normal_%") + ylim(0,1)

```

B.6 Chapter 7: Association of tissue change with radiation pneumonitis in NSCLS patients

Also available at: <https://github.com/vivifj03/TumorModel/blob/master/Code%20chapter7%20-%2018patients.R>

```

# Packages -----
library(xlsx)
library(ggplot2)

```

```

library(compositions)
library(e1071)
library(caret)
library(pROC)

# Data -----
# 1. Run model with data set of each patient, save model's coefficients
#    and transform them back to compositional data.
# 2. Use the predictions for each patient and cluster them using the
#    medoids of the reference patient.

coef<-read.xlsx("File_with_coefficients_for_each_patient's_model")
coef_com<-read.xlsx("File_with_coefficients_in_compositional_form_for
#####each_patient's_model")
cluster<-read.xlsx("File_with_distribution_(in_percentage)_of_patches_in
#####each_clustering_configuration_for_each_patient")

# Figures =====
# Figure 7.1
# replace beta2 for beta3-beta4, gamma2-gamma8
ggplot(data=coef, aes(x = beta2)) +
  geom_histogram(aes(color = factor(rp), fill = factor(rp)),
                 position = "identity", bins = 30, alpha = 0.4) +
  xlab("Coefficient_for_dose") + ylab("Count") +
  scale_color_manual(name="RP", values = c("#926b8d", "#E8CA47"),
                    labels = c("No", "Yes")) +
  scale_fill_manual(name="RP", values = c("#926b8d", "#E8CA47"),
                   labels = c("No", "Yes"))

### Regression #####
cluster$rpf<-ifelse(cluster$rp==0, "No", "Yes")

X = acomp( cluster[,13:17] )
X = zeroreplace(X,0.001)
mylogit <- glm(rp ~ ilr(X), data = cluster, family = "binomial")
(a = coef(mylogit)[1])
(b = ilrInv(coef(mylogit)[-1],orig=X))
summary(mylogit)
predict(mylogit)
predpr<-predict(mylogit, type = "response")
mylogit_pred = ifelse(predict(mylogit, type = "link") > 0, "Yes", "No")

calc_class_err = function(actual, predicted) {
  mean(actual != predicted)
}
calc_class_err(actual = cluster$rpf, predicted = mylogit_pred)

```



```
train_tab = table(predicted = mylogit_pred, actual = cluster$rpf)
train_con_mat = confusionMatrix(train_tab, positive = "Yes")
c(train_con_mat$overall["Accuracy"],
  train_con_mat$byClass["Sensitivity"],
  train_con_mat$byClass["Specificity"])

### ROC #####
roccurve <- roc(cluster$rp ~ predpr)
roccurve
plot(roccurve)
```

Vita

Viviana Alejandra Rodriguez Romero was born on October 27, 1983, in Bogota, Colombia. She received her Bachelor of Science in Statistics from Universidad Nacional de Colombia in 2006. Since then, she has been faculty at the School of Medicine in Pontificia Universidad Javeriana. Viviana attended Pontificia Universidad Javeriana, where she graduated in 2012 with an MS in Clinical Epidemiology, and her thesis was evaluated as meritorious. In 2014, Viviana was awarded a scholarship to pursue a doctoral degree in the U.S. She started her PhD studies at Virginia Commonwealth University (VCU) in 2015. During her graduate career at VCU, she has served as Associate in the Consulting Laboratory in the VCU Department of Biostatistics.