



# VCU

Virginia Commonwealth University  
VCU Scholars Compass

---

Theses and Dissertations

Graduate School

---

2021

## Applied Machine Learning in Extrusion-Based Bioprinting

Shuyu Tian

*Virginia Commonwealth University*

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



Part of the [Artificial Intelligence and Robotics Commons](#), [Biomaterials Commons](#), [Biomedical Devices and Instrumentation Commons](#), [Polymer and Organic Materials Commons](#), and the [Polymer Science Commons](#)

© The Author

---

Downloaded from

<https://scholarscompass.vcu.edu/etd/6850>

This Thesis 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](mailto:libcompass@vcu.edu).



**VCU**

College of Engineering

Chemical and Life Science Engineering

# **Applied Machine Learning in Extrusion-Based Bioprinting**

By

Shuyu Tian

A dissertation submitted in partial fulfillment of the requirements for the degree of Masters of Science in Chemical and Life Science Engineering at Virginia Commonwealth University.

Approved:

Advisor

Nastassja Lewinski, Ph.D.

Associate Professor, Department of Chemical and Life Science Engineering

Advisory Committee

Christina Tang, Ph.D., Department of Chemical and Life Science  
Engineering

Bridget McInnes, Ph.D., Department of Computer Science

Virginia Commonwealth University

Richmond, Virginia

© Shuyu Tian 2021  
All Rights Reserved

## Acknowledgements

I would like to thank my research advisor, Dr. Nastassja Lewinski for her support, patience, and mentorship. Dr. Lewinski has always been a source of inspiration and clarity, and her assistance and suggestions were invaluable towards the completion of not only this work, but other endeavours in our group as well. I want to also thank Dr. Bridget McInnes for her guidance and mentorship in not only the work in this thesis project, but also with navigating graduate school life. I would also like to thank Dr. Christina Tang, Dr. Hong Zhao, Dr. Daeha Joung, and Dr. Jennifer Puetzer for their helpful suggestions for research in my graduate school journey. I would like to acknowledge and thank the Department of Chemical and Life Engineering at Virginia Commonwealth University for their support of my education in chemical engineering and support in conference outreach.

I would also like to thank my labmates Dr. Liza Roger and Andrea Ferrar Vega for their helpful and valuable suggestions throughout the course of this research and other. I want to also thank Nick Rodriguez for his helpful suggestions on the thesis project when I ran into various issues and Rory Stevens for aiding in expanding the dataset used for this research.

I want to provide the deepest gratitude to my mom Wei Wang, my dad Haibin Tian, and my sister Sophie for their encouragement and being my support line whenever I need to talk. I also want to thank many friends I made since I moved to Richmond, including Anne Katherine Brooks, Kaitlin Kay, Adam Luxon, and Taylor Scott.

Thank you all for making my graduate school experience an extraordinary one for me.

# Table of Contents

Acknowledgements .....	i
Table of Contents .....	ii
List of Figures .....	v
List of Tables.....	vii
Abstract .....	1
Contributions .....	2
1. Introduction.....	2
1.1. Extrusion-based bioprinting.....	2
1.2 Common parameters across the EBB field .....	3
1.2.1. Printability .....	3
1.2.2. Extrudability .....	6
1.2.3. Shape Fidelity .....	6
1.2.4. Cellular considerations in printability.....	9
1.3. Fundamentals of machine learning (ML).....	12
1.4. Applications of ML in bioprinting .....	15
1.5 Project Overview .....	16
2. Development of supervised classification and regression models .....	19
2.1. Introduction .....	19
2.2. Materials and methods .....	19
2.2.1. Dataset creation.....	19
2.2.2. Experimental design .....	20
2.2.3. Model evaluation .....	21
2.2.4. Data preprocessing .....	22
2.2.5. Dataset training size variation .....	22
2.3. Results .....	22
2.3.1. Cell viability model performance .....	22
2.3.2. Filament diameter model performance .....	25
2.3.3. Model predictions compared to experimental trends (non-primary cells) .....	27
2.3.4. Effect of training data size on output predictions .....	28
2.3.5. Extrusion pressure recommendation prediction performance .....	31

2.4. Discussion .....	34
3. Experimental evaluation of supervised classification and regression models .....	39
3.1 Introduction.....	39
3.2. Materials and methods .....	40
3.2.1. Biomaterial ink synthesis .....	40
3.2.2. Cell culture maintenance .....	40
3.2.3. Bioink synthesis and construct printing .....	40
3.2.4. Live/dead staining .....	41
3.2.5. Filament diameter measurements .....	41
3.2.6. Extrusion pressure measurements .....	42
3.2.7. Intrastudy model creation and usage .....	42
3.2.8. Statistical analysis .....	43
3.3. Results .....	43
3.3.1. Effect of specified training data on cell viability .....	43
3.3.2. Effect of specified training data on extrusion pressure recommendation predictions .....	48
3.4. Discussion .....	51
3.4.1. Effect of specified training data on cell viability .....	51
3.4.2. Experimental errors and recommendations .....	53
4. Additional considerations for EBB predictive modeling .....	55
4.1. Relationship between extrusion pressure and material concentration in Al/Gel hydrogel .....	55
4.1.1. Introduction .....	55
4.1.2. Materials and methods .....	55
4.1.2.1. Biomaterial ink synthesis .....	55
4.1.2.2. Biomaterial ink printing .....	56
4.1.3. Results and discussion .....	56
4.2. Effect of support boths .....	58
4.2.1. Materials and methods .....	59
4.2.1.1. Synthesis of FRESH gelatin slurry .....	59
4.2.1.2. Alginate hydrogel synthesis .....	59
4.2.1.2. Alginate hydrogel printing.....	60
4.2.2. Results and discussion .....	60
4.2.2.1. Optimization of gelatin slurry synthesis .....	60

4.2.2.2. Optimization of feed rate .....	62
4.2.2.3. Effect of FRESH printing process on shape fidelity.....	64
4.3. Effect of various non-primary cells .....	66
4.3.1. Materials and methods .....	66
4.3.2. Results and discussion .....	67
5. Conclusion and Future Work .....	68
5.1 Conclusion .....	68
5.2 Future Work.....	70
6. Contributions.....	72
7. Vita .....	73
8. References .....	76
9. Appendix A: Figures and Tables .....	87
Appendix B: Python code used for ML model training, evaluation, and prediction of cell viability and extrusion pressure .....	167
Appendix C: Python code used for ML model training, evaluation, and prediction of filament diameter .....	186

## List of Figures

<b>Figure 1: Phase diagram indicating the optimal temperature and crosslinking extent for GelMA at 10, 15, and 20 % weight/volume (% w/v).....</b>	<b>4</b>
<b>Figure 2: Evaluation of printability from alginate concentration (weight %) and oxidation percentage combinations, taking into account material density, kinetic viscosity, and cell viability directly post-printing .....</b>	<b>5</b>
<b>Figure 3: Printability matrix based on filament shrinkage and swelling and filament deposition characteristics: a) Characterization of filament alginate-laponite filament based on surface structure and swelling/shrinkage behavior b) At constant laponite concentration in biomaterial ink, the effects of alginate and the ratio of nozzle movement speed to rate of extrusion (speed ratio) is shown c) Printability matrix based on material concentrations and speed ratio is shown, where filament with consistent cross sectional diameters and smooth surfaces can still be formed despite swelling.....</b>	<b>9</b>
<b>Figure 4: Optimal printing conditions incorporated cell viability and osteogenic differentiation for choosing the optimal concentrations for methacrylate-HA printing .....</b>	<b>11</b>
<b>Figure 5: Confusion matrix of logistic regression and support vector classification of cell viability. 10% of the cell viability dataset was used as testing data while 90% of the dataset was used as training data. CV indicates cell viability.....</b>	<b>24</b>
<b>Figure 6: Feature importance rankings of material, equipment, and experimental parameters based on random forest regression modeling of cell viability.....</b>	<b>24</b>
<b>Figure 7: Feature importance rankings of material, equipment, and experimental parameters based on random forest classification modeling of cell viability.....</b>	<b>25</b>
<b>Figure 8: Confusion matrix of support vector classification of filament diameter. 10% of the filament diameter dataset was used as testing data while 90% of the dataset was used as training data. FD indicates filament diameter.....</b>	<b>26</b>
<b>Figure 9: Feature importance rankings of material, equipment, and experimental parameters based on random forest regression modeling of filament diameter .....</b>	<b>27</b>
<b>Figure 10: Feature importance rankings of material, equipment, and experimental parameters based on random forest regression modeling of filament diameter.....</b>	<b>27</b>
<b>Figure 11: Feature importance rankings of material, equipment, and experimental parameters based on random forest regression modeling of extrusion pressure.....</b>	<b>35</b>
<b>Figure 12: Feature importance rankings of material, equipment, and experimental parameters based on random forest classification modeling of extrusion pressure.....</b>	<b>35</b>
<b>Figure 13: Total/dead confocal images in the cross-sectional view in the X-Z plane of A) 3/4 Alg/Gel and B) 3/7 Alg/Gel immediately after extrusion of A) 3/4 Alg/Gel and b) 3/7 Alg/Gel. The frames of images are 1.4 by 0.5 mm in dimension.....</b>	<b>44</b>
<b>Figure 14: 3/4 Alg/Gel constructs directly after printing onto tissue-culture treated well plate surfaces. The scale bars depict 1000 <math>\mu\text{m}</math>.....</b>	<b>47</b>
<b>Figure 15: 3/7 Alg/Gel constructs directly after printing onto tissue-culture treated well plate surfaces. The scale bars depict 1000 <math>\mu\text{m}</math>.....</b>	<b>47</b>



<b>Figure 16: Brightfield images of a) 3/4 Alg/Gel and b) 3/7 Alg/Gel filaments directly after extrusion. The scale bars depict 1000 <math>\mu\text{m}</math>.....</b>	<b>48</b>
<b>Figure 17: Brightfield images of a) 3/4 Alg/Gel and b) 3/7 Alg/Gel filaments directly after extrusion. The scale bars depict 1000 <math>\mu\text{m}</math>.....</b>	<b>48</b>
<b>Figure 18: Extrusion pressure required to produce filament deposits of two layers extruded at various gelatin concentration extruded from A) a 22G conical nozzle and B) a 25G conical nozzle. Bars with the same color indicate constant alginate concentration. ....</b>	<b>57</b>
<b>Figure 19: Based on original synthesis methods of FRESH gelatin slurry, A) no clear separation of supernatant and compacted gelatin particles can be observed or B) distinct separation of supernatant and gelatin microparticles can be observed. ....</b>	<b>61</b>
<b>Figure 20: 2% (w/v) alginate hydrogel construct printed into gelatin slurry with a <math>\text{CaCl}_2</math> concentration of 11 mM .....</b>	<b>64</b>
<b>Figure 21: 2% (w/v) alginate hydrogel construct after aspiration of melted gelatin slurry with a <math>\text{CaCl}_2</math> concentration of 11 mM.....</b>	<b>64</b>
<b>Figure 22: 2% (w/v) alginate hydrogel printed into gelatin slurry with <math>\text{CaCl}_2</math> concentration of 100 mM. The pink-colored components indicate alginate hydrogel deposits from the extrusion process. ....</b>	<b>65</b>
<b>Figure 23: 2% alginate deposited in gelatin slurry with 20 mM <math>\text{CaCl}_2</math> A) one minute post-printing and B) directly after removal of melted gelatin slurry. ....</b>	<b>66</b>

# List of Tables

<b>Table 1: Cell viability regression model performance based on average values of coefficients of determination (<math>R^2</math>) values and mean squared error (MSE) under 5-fold cross validation</b> .....	23
<b>Table 2: Cell viability classification model performance based on average values of accuracy, precision, and recall under 5-fold cross validation.</b> ....	23
<b>Table 3: Filament diameter regression model performance based on average values of coefficients of determination (<math>R^2</math>) values and mean squared error (MSE) under 5-fold cross validation</b> .....	25
<b>Table 4: Filament diameter classification model performance based on average values of accuracy, precision, and recall under 5-fold cross validation</b> .....	26
<b>Table 5: Cell viability regression model performance based on average values of coefficients of determination (<math>R^2</math>) values and mean squared error (MSE) under 2-fold, 5-fold, and 10-fold cross validation</b> .....	29
<b>Table 6: Random forest classification model performance for cell viability based on average values of coefficients of accuracy, precision, and recall scores under 2-fold, 5-fold, and 10-fold cross validation</b> .....	30
<b>Table 7: Filament diameter regression model performance based on average values of coefficients of determination (<math>R^2</math>) values and mean squared error (MSE) under 2-fold, 5-fold, and 10-fold cross validation</b> .....	31
<b>Table 8: Classification model performance for filament diameter based on average values of accuracy, precision, and recall values under 2-fold, 5-fold, and 10-fold cross validation</b> .....	32
<b>Table 9: Cell viability regression model performance based on average values of coefficients of determination (<math>R^2</math>) values and mean squared error (MSE) under 2-fold, 5-fold, and 10-fold cross validation</b> .....	33
<b>Table 10: Classification model performance for extrusion pressure based on average values of accuracy, precision, and recall values under 2-fold, 5-fold, and 10-fold cross validation</b> .....	34
<b>Table 11: Predicted cell viability values are compared against experimental values for corresponding material concentrations of alginate and gelatin</b> .....	35
<b>Table 12: Predicted filament diameter values are compared against experimental values for corresponding material concentrations of alginate and gelatin</b> .....	46
<b>Table 13: Predicted extrusion pressure required to deposit material are compared against experimental values for corresponding material concentrations of alginate and gelatin</b> .....	50
<b>Table 14: Quantitative shape recovery behavior observation of FRESH gelatin slurry 1 minute after movement of nozzle within the compacted slurry at 4 °C</b> .....	63

# Abstract

Optimization of extrusion-based bioprinting (EBB) parameters have been systematically conducted through experimentation. However, the process is time and resource-intensive and not easily translatable across different laboratories. A machine learning (ML) approach to EBB parameter optimization can accelerate this process for laboratories across the field through training using data collected from published literature. In this work, regression-based and classification-based ML models were investigated for their abilities to predict printing outcomes of cell viability and filament diameter for cell-containing alginate and gelatin composite hydrogels. Regression-based models were investigated for their ability to predict suitable extrusion pressure given desired cell viability when keeping other experimental parameters constant. Also, models trained across data from general literature were compared to models trained across data from one literature source that utilized alginate and gelatin bioinks and experimental conditions closely replicatable with available laboratory resources. The results indicate that models trained on large amounts of generalized data can impart physical trends on cell viability, filament diameter, and extrusion pressure seen in past literature. Regression models trained on the larger dataset also predicted cell viability closer to experimental values for material concentration combinations not seen in training data of the single-paper-based regression models. While the best performing classification models for cell viability can achieve an average prediction accuracy of around 70%, the cell viability predictions remained constant despite altering input parameter combinations. Trained models on bioprinting literature data show the potential usage of applying ML models to bioprinting experimental design. Furthermore, experimental parameters of polymer precursor concentration, support bath presence, and non-primary cell types were empirically explored in their effects on extrusion pressure, filament diameter, and cell viability respectively.

# 1. Introduction: Extrusion-based Bioprinting and Applicability of Machine Assisted Experimentation

Adapted from:

1. S. Tian, H. Zhao, N. Lewinski, Key parameters and applications of extrusion-based bioprinting, *Bioprinting*. (2021) e00156. <https://doi.org/10.1016/j.bprint.2021.e00156>.
2. S. Tian, Machine Assisted Experimentation of Extrusion-based Bioprinting Systems, (2021). <https://www.mdpi.com/2072-666X/12/7/780>.

## 1.1 Extrusion-based bioprinting

In the field of bioprinting, the most frequent modality used is extrusion-based bioprinting (EBB). Extrusion bioprinters contain three main features: 1) cartridge or syringe reservoirs for cell-laden or cell-free material, 2) a dispensing mechanism such as pneumatic pressure, pistons, or ejection screws, 3) and a nozzle to control printing resolution [1]. Extrusion bioprinters can contain multiple printheads to deposit various cell-laden or cell-free material. This holds an advantage of over other forms of bioprinting modalities such laser-assisted bioprinting and electro-spinning as there can be higher control over where biomaterials, different cell types and densities, signaling molecules, and support materials can be deposited. For this work, the term bioink will be defined by the International Society of Biofabrication's definition: 'formulation of cells suitable for processing by an automated biofabrication technology that may also contain biologically active components and biomaterials [2]. In addition, EBB can operate at higher cell densities and compared to additional printing modalities. Many extrusion printer systems offer additional printing processing capabilities, including UV irradiation capabilities for photoinitiation of crosslinkable material and the ability to hold a support bath for low viscosity bioinks. Throughout the field, the characterization of printing construct quality is done with various quantitative and qualitative approaches. Furthermore, printing parameters ensuring high cell viability and appropriate expression of cellular markers have not been characterized across different types of organs.

## 1.2 Common parameters across the EBB field

### 1.2.1. Printability

Printability can be generally defined as the suitability of a bioink to be extruded into the desired construct geometry need. Within this broad term, different research groups have developed specific definitions, quantitative and qualitative, capturing specific material and post-processing parameters such as rheological properties and shape fidelity ratio of printed filament [3–10]. Notable quantitative measurements of printability include a modification of the circularity equation noted as  $Pr$  (**Equation 1**).

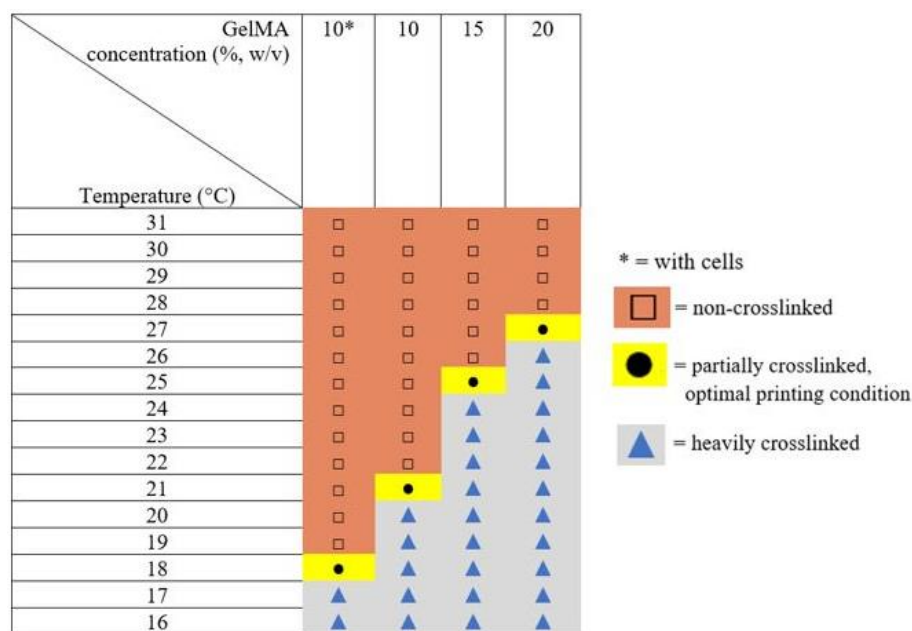
$$Pr = \frac{L^2}{16A} \quad (1)$$

In Equation 1,  $L$  here indicates the perimeter of filament forming an infilled pore given a cross-sectional area at its base and  $A$  is the cross-sectional area of the pore. A  $Pr$  value of one would indicate perfect square shape [8,11]. The equation is modified to quantify deviation from square geometry of pores formed from layered filaments. Deviation from square geometry is used as most of EBB constructs contain successive filament with  $90^\circ$  deviation from the previous layer. Another equation used to quantify printability is the printability optimization index (POI) (**Equation 2**), which accounts for extrusion pressure ( $P$ ), nozzle diameter ( $D$ ), Young's modulus of the extruded material ( $M$ ), and printing accuracy ( $Acc$ ) [12,13].

$$POI = \frac{Acc M}{DP} \quad (2)$$

Different groups in the field have captured printability through several types of phenomena: extrudability, filament continuity and consistency, shape fidelity, and filament spreading [14]. With combinations of extruded biomaterial characteristics, resultant cell viability, and printing settings, ranges of printability can exist to provide a set of optimized parameters to use for printing aforementioned biomaterial.

When conducting printability tests to observe the effects of viscosity, material concentration, cell concentration, and mechanical properties, a visualized matrix or phase diagram of what can be extruded produces succinct yet important information. In general, three sets of printability criteria exist for a printability matrix: printable, unprintable (viscosity of the material being too low, leakage through printing nozzle), and unextrudable (viscosity of the material being too high). The most common type of printability matrices are based on material concentrations, where the printability criteria is correlated to different combinations of one material's concentration to another printing variable, often including temperature (**Fig. 1**), cross-linking duration, concentration of another printing material, or nozzle movement speed [15–23].



**Figure 1.** Phase diagram indicating the optimal temperature and crosslinking extent for GelMA at 10, 15, and 20 % weight/volume (% w/v) (Adapted from Gu et al. [16])

Printer-specific parameters, such as feed rate, have also been used in printability matrices (e.g. correlation of viscosity-based speed parameter and material storage modulus ratio to filament definition [21]). Other forms of printability matrices include print material elastic modulus and maximum loading, material extrusion rate and extrusion pressure, dynamic moduli and extrusion pressure, nozzle

movement speed and extrusion pressure, and material concentration and oxidation percentage (**Fig. 2**) [4,15,24–26].

Oxidized alginate bioink selection						
Ox. \ Conc.	2	5	8	10	15	20
0	-	+	+	-	-	-
1	-	-	+	+	-	-
3	-	-	+	+	-	-
5	-	-	-	+	+	-
10	-	-	-	-	+	+

**Figure 2.** Evaluation of printability from alginate concentration (weight %) and oxidation percentage combinations, taking into account material density, kinetic viscosity, and cell viability directly post-printing (Reproduced with permission from Jia et al. [26])

Printability studies thus far have gathered sufficient data to extract trends and understand the different materials used. Material storage and loss moduli can be linked to viscosity and printability parameters. It is known that when the storage and loss moduli are equivalent, the material is at its gelation point and becomes rigid enough to form discrete layers. However, specific upper and lower limits of these ratios have not been established for the complete process of flow initiation to gel formation [27,28]. Another useful process to supplement printability parameters is modeling a material’s viscosity change over shear stress behavior (shear sweep test) using a non-Newtonian fluid model such as the Ostwald-de Waele (Power Law) or Hershel-Bulkley shear-thinning model [29]. Specifically,  $K$ , the flow consistency index and  $n$ , the flow behavior index or these equations can be obtained by finding the y-intercept and slope of fitted regressions on logarithmic scale shear sweep data respectively. The flow behavior index is of particular interest as it indicates the extent of shear thinning ( $n < 1$ ) or shear thickening ( $n > 1$ ) of a material [30]. In one case, a material with larger  $K$  and lower  $n$  resulted in having

acceptable printability across a larger pressure range [29].

### *1.2.2. Extrudability*

Extrudability is simply the ability for a bioink or biomaterial to be extruded out of a nozzle to form a consistent filament. A bioink's viscosity is the most important indicator to resist against extrusion. With higher bioink viscosity, higher shear stress can result at the interface between the inner nozzle surface and bioink, which can disrupt cell membrane integrity and lead to lowered cell viability [31]. To reduce shear stress-induced cell death, it is standard in the EBB field to use shear thinning hydrogels, meaning their viscosities decrease with increasing shear strain. With large nozzle outlets, less extrusion pressure is needed to extrude filament and subsequently inducing lower shear stress as well. The downside to using larger nozzles is the reduction of filament resolution. Extrudability has been characterized in various formats as well. One group quantified extrudability as pressure needed to attain a specific material flow rate through a nozzle [24] while other groups deemed a bioink of a certain polymer precursor concentration to be unextrudable if it cannot achieve a flow at a maximum operating pressure [32], if gelation occurs before extrusion at too high of a rate [33], or if nozzles attached onto the cartridge becomes loose before material can be extruded out [34].

### *1.2.3. Shape fidelity*

Shape fidelity is the ability for a printed construct to retain as much of the desired size and deposition geometry of the computer aided design model is one of most important printability measures. The degree of shape fidelity can be determined through image analysis by measuring the variation in filament diameter between the computer-aided design and the printed design, i.e. ink spreading, angular and height differences [35–38]. Using the printability ratio for suitable pore size (e.g. mesh designs) and construct geometries can supplement image analysis, although this practice is not widespread. Shape fidelity needs to be defined considering the extruded materials' cumulative interactions with the substrate, multiple filament layers, cells, and biochemical additives. Bioink composition, extrusion pressure, and nozzle or platform movement speed also need optimization to produce suitable shape



fidelity. One method that evaluates these parameters together is examining filament coalescence. Filament coalescence is the merging of discrete filaments into a single filament. Coalescence is tested based on the distance between filament strands printed when bioink composition and printer parameters are fixed. Filament coalescence tests have been considered to further develop appropriate resolution [9]. Based on filament fusion studies, the minimum distance between two filaments that does not result in merging is a key factor to consider for 3D construct designs after viscoelastic and filament diameter behavior. Despite filament diameter being a straightforward method to evaluate shape fidelity, in practice few groups report this parameter. Meanwhile, filament line spacing after deposition is also seldom stated. Fiber diameter, spacing, and pore size, are crucial to determining scaffold integrity and its ability to support cell survival and proliferation.

Methods for assessing fidelity also vary, where one can assess fidelity post-printing by forcing the printed structures to undergo different physical and chemical changes to see if shape fidelity is retained [39]. Current definitions of shape fidelity have focused on retaining the physical structure of printed scaffolds without considering cellular effects. New assessment of shape fidelity should incorporate effects on cell viability, extracellular matrix (ECM) production rates, as well as set limits to expansion and shrinkage of filament post-printing.

Resolution is defined as the diameter of filament extruded. Suitable resolution values for bioprinting depends on the desired intricacies of the construct geometry used. Printing with smaller resolution opens up the possibilities to accurately deposit bioink in geometries at scales similar to the tissue of interest. When single filament strands extruded from a cylindrical nozzle are considered, theoretical filament diameter can be modeled based on the Power Law as:

$$d = D \sqrt{\frac{D}{2v} \left(\frac{\eta}{K}\right)^{1/(n-1)}} \quad (3)$$

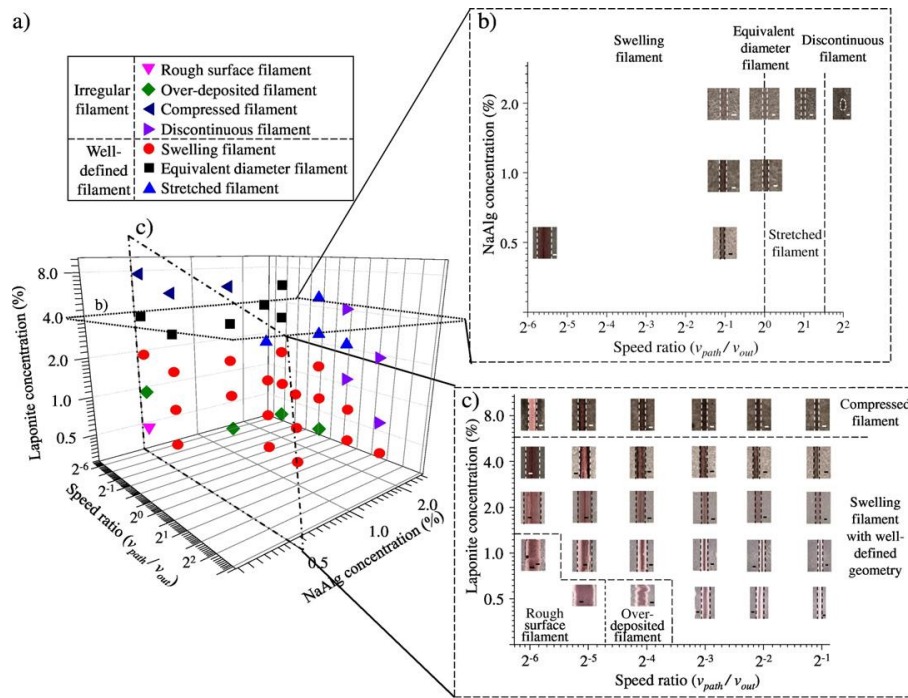
where  $D$  is the nozzle diameter,  $v$  is the average flow rate velocity,  $\eta$  is the viscosity of the material, and  $K$  and  $n$  are the power law coefficients [40]. The average flow rate ( $v$ ) can be found from a

derivation based a non-Newtonian fluid extruding through a cylindrical nozzle, shown in **Equation 3** [29]:

$$v = \left(\frac{-\Delta P}{2LK}\right) \left(\frac{n}{3n+1}\right) R^{\frac{n+1}{n}} \quad (4)$$

Here,  $\Delta P$  is the extrusion pressure,  $L$  is length of the nozzle, and  $R$  is the nozzle inner radius. From **Equation 4**, increasing viscosity for a shear-thinning material ( $n < 1$ ), decreasing nozzle diameter, increasing flow rate velocity, and decreasing  $n$  all would decrease filament diameter printed, which improves printing resolution under the traditional definition that smaller diameters mean increased resolution [40].

Hydrophobic surfaces lead to less spreading of deposited materials, and lead to a smaller increase in filament diameter onto a substrate. Miri et al. [40] demonstrates this in resolution simulations, although surface tension effects are noted to be smaller than nozzle or substrate movement speed and material viscosity due to limitations of modeling viscoelastic shape recovery mechanisms as well as the weight of deposited material. Shrinkage in filament diameter from nozzle diameter is desired for resolution control for a wide range of applications. However, ideal filament diameter, or width, during and after deposition should be the same as the nozzle diameter [41,42]. Noticeable shrinkage and swelling have been observed for alginate and other materials added in hydrogel composites [21,43,44]. Although filament swelling and shrinkage can occur under different printing parameters, printability can still be maintained (**Fig. 3**) [21]. When optimizing bioprinting parameters, it is commonly stated that cell viability ranges from 40-80% for extrusion printing [45–47], and improving this range to 90% or above would be optimal [37]. If printing with lower resolution as opposed to a higher resolution results in higher long-term cell viability, and desired biological functions, thinner filaments may not be necessary.

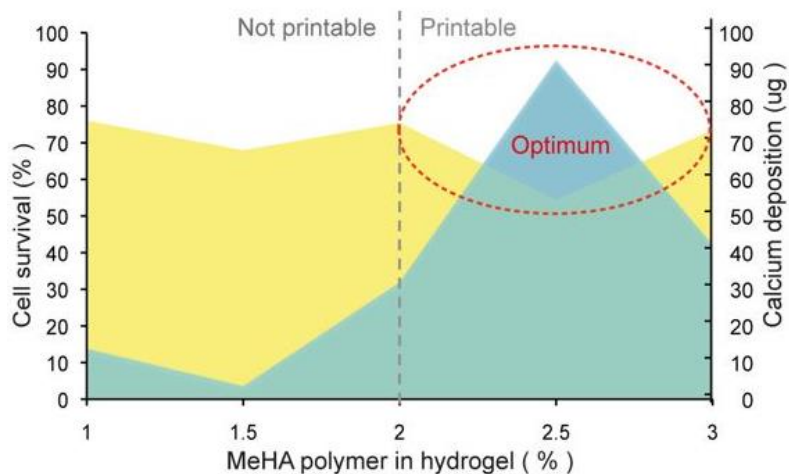


**Figure. 3.** Printability matrix based on filament shrinkage and swelling and filament deposition characteristics: a) Characterization of filament alginate-laponite filament based on surface structure and swelling/shrinkage behavior b) At constant laponite concentration in biomaterial ink, the effects of alginate and the ratio of nozzle movement speed to rate of extrusion (speed ratio) is shown c) Printability matrix based on material concentrations and speed ratio is shown, where filament with consistent cross sectional diameters and smooth surfaces can still be formed despite swelling (reproduced with the permission of Huang et. al. [21])

#### 1.2.4. Cellular considerations in printability

Printability criteria often consider cell viability, especially those related to shear stress and residence time of cells within a nozzle; however, specific ranges of acceptable shear stresses, residence times, and bioink viscosity remain to be established for different types of cells. For example, mice embryonic stem cells can experience drastic cell viability loss under the same range of shear stress where immortalized cell such as mice L929 fibroblasts, human mesenchymal stem cells, or HeLa cells are less affected [8,48,49]. Additionally, shear stress can impart a decrease in proliferative ability in stem and immortalized cells at relatively higher values [48,49] while also improving proliferation and

differentiation in human stem cells between the ranges of 5 to 10 kPa [48]. Additionally, sterilization methods on bioink polymer precursors needs to be chosen carefully to not lower printability. Seen in one study, autoclaving alginate powder led to hydrogels that extruded filaments with higher spreading ratio as compared to hydrogel filaments composed of UV-sterilized, and ethanol-sterilized alginate powders [50]. The study also demonstrated that across different sterilization techniques, relatively high printing pressure with moderate nozzle movement speeds can be used to reduce deposition discontinuities. Since printability is affected by many variables, printing parameters need to be systematically and consistently characterized to clearly understand causes of specific behaviors of printed constructs and the resulting cell viabilities. Surface tension between the bioink and substrates, viscoelastic effects of hydrogel bioinks, thixotropic behavior of bioinks, unique polymer processing steps such as die swelling during extrusion [4], and the effect of cell density and cell type on printability metrics as compared to cell-free bioinks are all areas that would warrant more work to better define windows of printability. Furthermore, it would be beneficial to visualize information that correlates cell growth and metabolic activity, such as viability and protein expression, to printability. Poldervaart et al. [19] demonstrate this by finding an optimal printing range to promote desired cell viability at certain methacrylated hyaluronic acid concentrations while also correlating the printing range to calcium deposition from human bone marrow-derived mesenchymal stem cells (**Fig. 4**).



**Fig. 4.** Optimal printing conditions incorporated cell viability and osteogenic differentiation for choosing the optimal concentrations for methacrylate-HA printing (Reproduced with permission Poldervaart et al. [19])

It has been shown that printing pressure has a far greater impact on cell viability than nozzle diameter [51]. This does not diminish the effect that nozzle size has on cell viability during and after the extrusion process. Subsequent work demonstrated decreasing nozzle size increases shear stress applied to cell-laden bioinks and lowers cell-viability over 7 days [48]. Other groups' works expanded upon Nair et al.'s [51] findings for both conical and cylindrical nozzles through modeling material flow undergoing Poiseuille flow, although cell viability trends differ vastly between the two types of extrusion tips [31,52]. Comparatively, the high shear stress region in the cylindrical nozzles was determined to cover distances up around 16 mm from the nozzle outlet, which implies cells are exposed to a larger area of high shear stress than in a conical nozzle [31]. However, the highest shear stress present at the nozzle tip (<1 mm from the nozzle outlet) can be up to ten times higher than the greatest shear stress on a cylindrical nozzle tip [31]. This gives rise to a tradeoff between nozzle geometry and extrusion pressure. It is suggested that conical nozzles are preferred when lower pressures are used due to cells being exposed to high shear stresses for a lower duration. Meanwhile, cylindrical nozzles are to be used at higher pressure scenarios as conical nozzle tip shear stress becomes significantly higher than in cylindrical nozzles [31]. Beyond overall cell viability, cellular movement in bioink based on the

velocity profile and distance from a cylindrical nozzle centerline was modeled and tested by one group, finding that cell morphology and survival conditions (healthy, apoptotic, and necrotic) became more varied as extrusion pressure increases while enzymatic expression of stem cell differentiation into osteogenic categories increased [53]. Cellular distribution during extrusion as a function of radial distance from the nozzle centerline was also derived.

### 1.3 Fundamentals of machine learning (ML)

Machine learning (ML) is a subset of artificial intelligence aimed at creating predictive systems from existing data and set algorithms. In contrast to models based on explicit physical equations, such as Power Law models in the case of predicting extruded bioink filament diameters, ML approaches use pattern recognition algorithms to discern mathematical relationships between empirical observations of input variables and extrapolate them to predict chemical, biological and physical properties of desired products. Conventionally, ML approaches can be very efficient methods of modeling desired input combinations once a sizable dataset is created. ML approaches also do not require extensive computational power in many cases. As the ML models only require datasets and a framework for training and validation, bioprinting data can be incorporated in dataset form to potentially accelerate formulation of different bioprinting material and printing parameters to output certain biological and physical endpoints.

When constructing ML models, the following criteria must be satisfied:

- 1. Data sourcing:** The quality of data used for ML model training is paramount for the usability of the model. Not only does the model have to contain a sizable number of data instances, but also contains a diverse combination of input and output variables if possible. In general, datasets will contain sources of bias mainly originating from the process of data extraction. Bias can exist from error sources as well as data selection processes.
- 2. Data cleaning and curation:** With dataset creation comes the need to eliminate errors, omitted data,

and controlling for outliers. Common data cleaning procedures include imputing values for missing data based on common values seen for a variable in a dataset, deleting duplicative data instances, deleting data instances with physically unfeasible characteristics. Normalization of quantitative portions of datasets can also be included when data is derived from multiple sources. This is to consider discrepancies in characterization methods on gathering data sources. With dynamic and large datasets, automated workflows and pipelines are needed to provide quick and accurate curation.

- 3. Data representation/encoding:** Data representation pairs hand in hand with data cleaning and curation. How data is represented, or encoded for model training, is critical to determining model performance. Major forms of representation of data includes categorical, numerical, binary, and graphical. Categorical representation of data assigns integer code to each category. For example, surface charge of a nanoparticle in a quantitative structure-activity relationship dataset can be represented as categorical data, with negative charge assigned to integer values of 0, neutral charge assigned to integer values of 1, and positive charge assigned to integer values of 2. A subset of categorical data is binary data, where only two categories exist for represented data. Binary data is commonly used to represent the presence of a variable. For instance, the existence of nanoparticle coating can be represented as binary data with “Yes” represented as 0 and “No” represented as 1. Numerical data is straightforward to explain; variables with continuous numerical distributions are represented as numbers.
- 4. Model choice:** Various ML models exist with varying complexity. From a big-picture view, ML algorithms can be divided into two training styles: supervised and unsupervised. The major delineation between the two styles is the use of data labelling. Supervised learning algorithms use datasets with set labels of input variables for training whereas unsupervised learning algorithms use unlabeled input data [54]. Supervised learners undergo iterations of training where the model makes output predictions and is corrected when predictions made are wrong. The training process continues

until an acceptable threshold level on training accuracy is reached. Supervised learning algorithms are mainly used to predict classification and regression output variables from a labelled dataset. Unsupervised learning models are trained through deducing structures and patterns within input data, where the model attempts to reduce redundancy and organizes data by similarity. Instances of unsupervised learning include data clustering and association rule learning [54].

Complexity of ML algorithms can range from classical linear algorithms such as support vector machines and logistical regression models, to ensemble models such as random forests, and deep learning models that involves usage of artificial neural networks. The complexity of ML algorithms does not relate to whether they are suitable for training with a specific dataset. If complex algorithms were to be chosen, baseline models should be developed with simpler algorithms to evaluate difference in prediction performance to justify the usage of complex algorithms.

**5. Model training and validation:** Training a ML model involves splitting a dataset into three sets: a training set, validation set, and testing set. The training set is the portion of the dataset that is used to develop a modeling using the algorithm of choice. The validation set is used for optimization of hyperparameters of ML algorithms, such as maximum tree depth in random forest models. Lastly, the testing set is the sample of data used for evaluating the predictive accuracy of the trained model. The formation of these subsets can be done by randomly splitting data into set portions or through specified split based on initially clustering data together to ensure diversity in each subset to represent the intended application range. Validation of the trained model (different from validation for algorithm hyperparameter tuning) can then be conducted on desired input parameter combination ranges to output outcomes that can be compared to the trained model's predicted outcome.

## **1.4 Application of ML in Bioprinting**

The inclusion of ML in 3D bioprinting is relatively new, although unique contributions have been made thus far. Shi et al. implemented a multilayer perceptron-based artificial neural network



trained with computational fluid dynamics simulations of droplet formation and flow behavior to predict classification-based droplet behavior based on voltage, nozzle diameter, bioink surface tension, and bioink viscosity input parameters for a drop-on-demand bioprinting system [55]. Experimental validation of six different input parameter combinations associated with different droplet formations confirmed that experimental results matched with predicted droplet formation with each combination. The same group developed a multi-objective optimization design method using a gradient descent-optimized fully connected neural networks to create single droplets based on optimized voltage, nozzle diameter, bioink viscosity, and bioink surface tension in comparison to randomly set voltages, bioinks with arbitrary surface tensions and viscosities, and printer nozzle diameter [56]. Specific to EBB, ML has been used for iterative optimization of printability. Lasso regression was used to optimize printed structures in a support bath using both underlying physical parameters that are not directly manipulated (e.g. bath material recovery time and perturbation growth rate) along with directly manipulated experimental variables (e.g. material flow rate, bath material concentration, and extrusion material concentration) [57]. The benefit of using this model was that a specific combination of construct height, support bath material concentration, and retraction distance was found to retain print fidelity while printing at a faster speed. Another iterative study applied Bayesian Optimization on an initial dataset of printability scores based on material and EBB printing parameters, of which parameter combinations are predicted with new experimental results to improve printability scores until an optimal parameter combination is met with the highest possible printability score [58]. The use of ML resulted in needing 4 to 47 experiments to find optimal parameter combinations compared to using a total possible number of experiments ranging from 6,000 to 10,000 determined by the Bayesian Optimization algorithm. Coney et al. also examined random forest regressor and classifier capabilities in determining printed construct quality using a previous EBB dataset containing systematic examination of poly(propylene fumarate) [59,60]. Results indicated satisfactory labeling performance from both random forest models.

The common theme amongst above studies is that living cells were not used. Incorporating

cellular parameters and predicting cellular performance in bioprinted constructs appear to be the next step in ML incorporation in bioprinting. Lee et al. tested cell viability based on collagen, hyaluronic acid, and fibrin formulations predicted using the relative least general generalization algorithm along with multiple regression modeling for printability [61]. On top of maintaining suitable shape fidelity, cell-laden scaffolds with optimized material concentrations exhibited increasing cell proliferation and migration up to 28 days after printing. Xu et al. developed a model based on ensemble learning for cell viability prediction in stereolithography-based bioprinting [62]. Prediction performance on 10% of the dataset used showed a coefficient of determination ( $R^2$ ) score of 0.953, indicating high goodness-of-fit for viability prediction of new parameter combinations.

## 1.5 Project Overview

Applications of ML in EBB are currently limited in translatability across different experimental conditions, predictability for both biological and physical printability outcomes, and a lack of experimental validation of developed models in most studies. The goal of the project is to develop and comparatively evaluate predictive models through the lens of four objectives:

1. Outcome prediction: How accurate are biological and printability predictions compared to experimental values for specific testing parameters?
2. Condition recommendations: If specific biological and printability outcomes are desired, what values does one or more input variable need to be to achieve those outcomes?
3. Prediction size: How do models created with varied training data size vary in performance metrics?
4. Specified dataset vs. generalized dataset: Training models through a dataset that uses similar or the same experimental conditions as the conditions of experimental validation, how do outcome predictions fare compared to using a larger dataset containing data sourced from varying experimental conditions?

To examine these research objectives, the development of predictive models that incorporate data of

various experimental settings is of particular interest. Specifically, these models have potential robustness to be used with various materials, cellular, and printer conditions. As stated previously, data collection for ML model implementation have still been sourced from singular research groups. Ideally, development of data scrapping tools for quantitative variable information in-text and in-image can accelerate and automate the dataset creation process. However, no known tools exist for curating information from bioprinting literature, specifically image analysis platforms to aid printability and graphical information extraction. In addition, ample amounts of data are needed to provide an accurate snapshot of experimental conditions used across bioprinting laboratories in the past decade and more. An initial goal of the project is to compile a dataset containing prevalent bioink materials, solvents used for bioink synthesis, crosslinking agents and their quantitative characteristics, printer and printer accessory characteristics, cellular characteristics and outcomes, and printability outcomes.

With this dataset, one can compare different ML models trained on the dataset via classification and regression performance metrics. Common regression performance metrics of coefficient of determination ( $R^2$ ) and mean squared error were used in this project to assess model performance validated on held out data. Accuracy, precision, and recall performance metrics were used to capture and compare classification model performance. From the base dataset, data instances with existing filament diameter values were compiled to create a dataset for filament diameter predictions and the same strategy was applied to create a dataset for extrusion pressure condition recommendation prediction. Performance metric evaluation was conducted with varying training set sizes to determine dataset size importance. A substantially smaller dataset was created via data gathered from literature using almost the same experimental set up available in our lab to train chosen ML models for evaluating the specified dataset effect.

The development and evaluation of these ML models can provide a baseline examination of the efficacy of ML in creating generalized predictive models for EBB. Models evaluated for different parameters can also elicit the relative importance of variables used for predictions, which can offer

insight on future experimental development centered around those variables.

## 2. Development of supervised classification and regression models

### Adapted from:

1. S. Tian, Machine Assisted Experimentation of Extrusion-based Bioprinting Systems, (2021). <https://www.mdpi.com/2072-666X/12/7/780>.

### 2.1 Introduction

The database utilized for ML model training contains bioink material concentration, solvent used, polymer crosslinking information, printing settings, cell viability, and printability results accrued from 75 EBB manuscripts over the past 13 years. Shown through previously mentioned studies, data used for bioprinting ML model training and testing has only been gathered from and applied within group. To our knowledge, our compilation of experimental data and parameters reported from different bioprinting laboratories for ML applications is the first of its kind. The database contains 617 unique instances of cell viability and 339 unique instances of printability. We analyzed the ability of ML regression and classification techniques to accurately and precisely predict cell viability and filament diameter outcomes based on certain combinations of material, biological, and printing parameters. In parallel, extrusion pressure recommendation models were also evaluated comparatively for prediction accuracy and preciseness.

### 2.2 Materials and methods

#### 2.2.1. Dataset creation

Two datasets of 617 instances corresponding to a unique cell viability value and 339 instances corresponding to a unique filament diameter value were collected from 75 EBB papers found through the search terms  $TS = \text{Extrusion AND (Bioprinting OR Bioink)}$  and  $TS = (\text{Extrusion OR Extrud*}) \text{ AND (Bioprint* OR Bioink*) AND (alginate*) AND (gelatin*) AND (viability OR viable* OR surviv* OR death OR proliferat*)}$  in Web of Science. Material concentration, solvent usage, crosslinking mechanism and duration, printer settings, observation duration, cell viability, and filament diameter were recorded for each unique instance of either cell viability and/or filament diameter. Papers used for

data extraction all utilized live/dead staining for comparable cell viability quantification. When cell viability data were presented in graphical form, PlotDigitizer software (<http://plotdigitizer.sourceforge.net>) was used to estimate cell viability values in relation to the viability scales they are presented against. Filament diameter values were extracted (via PlotDigitizer) from images provided in different manuscripts corresponding to different times of observation after printing. The datasets created are available through the Open Science Framework [63].

### 2.2.2. *Experimental design*

We framed the prediction of cell viability, filament diameter, and extrusion pressure as supervised regression-based and classification-based questions. In the regression models, a value of cell viability and filament diameter was predicted based on the training set and compared with the true cell viability and filament diameter values of the test set. For cell viability classification models, a binary class was created from the numerical cell viability data by setting a threshold for acceptable cell viability to be equal to or above 80.0%. The cell viability class was “Acceptable Cell Viability” with values of “Y” for yes and “N” for no. For filament diameter classification models, a binary class was created from the numeric filament diameter data by setting a threshold for tolerable filament diameter equal or above 10.0% error [17,18]. This was determined by calculating the absolute difference between filament diameter and nozzle diameter and dividing by nozzle diameter. The class was named “Acceptable Filament Diameter” with values of “Y” for yes and “N” for no based on above criteria. At hydrostatic pressures above 100 kPa, cell metabolic behavior can become negatively affected [64]. In cell viability instances with stated extrusion pressures, instances with a pressure above 100 kPa were deemed to have unacceptable extrusion pressure, while the rest were deemed acceptable. We evaluated three regression learners in this study: support vector regression, linear regression, and random forest regression; and three classification learners: random forest classification, logistic regression classification, and support vector machines.

### 2.2.3. *Model evaluation*

Metrics used for evaluating regression model performance were the coefficient of determination ( $R^2$ ) and mean squared error (MSE).  $R^2$  is a measure of goodness of fit of the model on provided data. It indicates the proportion of the variance in the dependent variable that is explained by independent variables. A perfectly fit model will have a  $R^2$  value of zero. MSE indicates the average of the squares of errors. Errors are the differences between actual values and predicted values. As MSE values become closer to zero, the lower the overall error becomes for model fit onto data. One regression model was chosen for prediction usage based on the highest coefficient  $R^2$  values and lowest MSE over k-fold cross validation training evaluation up to k=10 relative to other models.

Metrics used for evaluating classification model performance were accuracy, precision, and recall scores. Accuracy represents the percentage of correctly predicted outcomes for a sample. Precision is calculated by the ratio of true positive prediction over the sum of true positive and false positive predictions. Precision represents the proportion of correct predictions over sum of all predictions of the same label. Recall is similar to precision calculations, but the number of false positives is replaced by false negatives in predictions. This represents the proportion of all instances of the same label that are predicted correctly. Overall, a classification model was chosen for prediction usage based on the highest average prediction accuracy uses k-fold cross validation training evaluation methodology.

Chosen models were then utilized to predict acceptable cell viability and filament diameter from material and printing parameter combinations feasible to conduct in our laboratory for experimental verification of the predicted values. In addition, extrudability of low viscosity and high viscosity was also tested using materials and material concentrations within range of the dataset by predicting the extrusion pressure that would produce desired cell viability and filament diameters. Datasets were preprocessed and ML models were created through Python programming language (Python 3.8) via Jupyter notebook files (<https://jupyter.org/>). **Appendix B** and **C** provides the code used to preprocess

data, build models, evaluate models, and output variable predictions.

#### 2.2.4. *Data preprocessing*

Within the dataset, null instances for bioink temperature (i.e. syringe temperature) and printing substrate temperature were set at 22 °C as the majority experiments were conducted or are assumed to be conducted at room temperature. Additional variables with more than 50% null values were removed from the dataset and non-printing in-stances were also removed (instances with cast molded bioink or other methods with cells cultured within non-extruded hydrogel). Variables with only null instances and instances of zero units were removed prior to model usage as available imputation methods of null values would not provide an accurate representation of actual quantitative values of the variables used in respective manuscripts. Additional variables with null values and non-zero instances were imputed through k-nearest-neighbors imputing with a neighbor range of 30. Categorical data was encoded through one-hot-encoding. Feature selection was performed through conducting feature importance analysis on variables within the cell viability and filament diameter datasets respectively using random forest regression. For regression model performance evaluation, continuous variable instances were normalized through the `MinMaxScaler()` function (Sci-kit Learn package, Python).

#### 2.2.5. *Dataset training size variation*

Cross-validation of datasets was used to test training size variation by varying how many folds the training data is divided into. The greater number of folds, the greater the number of instances used for training. For each model, performance metrics were compared by k-fold cross validation with k values of 2, 5, and 10.

## 2.3 Results

### 2.3.1. *Cell viability model performance*

Amongst regression models, random forest regression models for cell viability predictions elicited higher  $R^2$  while minimizing average MSE (**Table 1, Fig. S1-2**).



**Table 1.** Cell viability regression model performance based on average values of coefficients of determination ( $R^2$ ) values and mean squared error (MSE) under 5-fold cross validation.  $R^2$  and MSE values were averaged from validation amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.

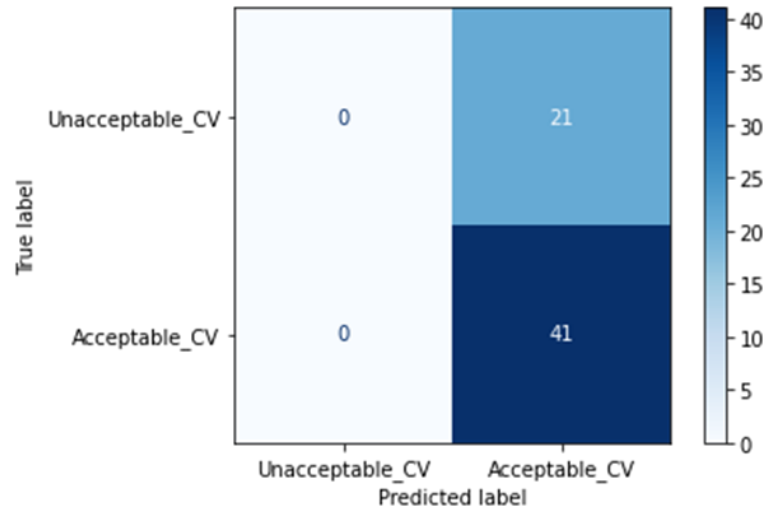
<b>Cell viability regression learner model</b>	<b>Average <math>R^2</math></b>	<b>Average MSE</b>
Random forest regression	0.384	0.019
Linear regression	0.231	0.024
Support vector regression	0.000	0.031

Random forest classification models elicited higher prediction accuracy, precision, and recall than other models tested (**Table 2, Fig. S3-5**).

**Table 2.** Cell viability classification model performance based on average values of accuracy, precision, and recall under 5-fold cross validation. Accuracy, precision, and recall values averaged from validation amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.

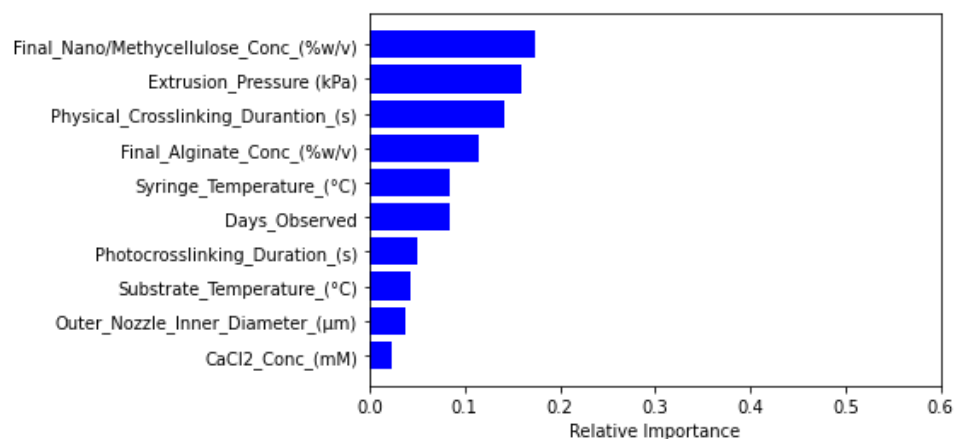
<b>Cell viability classification learner model</b>	<b>Average Accuracy</b>	<b>Average Precision</b>	<b>Average Recall</b>
Random forest classification	0.689	0.678	0.942
Logistic regression	0.616	0.616	1
Support vector classification	0.616	0.616	1

Both logistic regression and support vector classification models elicited the same performance values for accuracy, precision, and recall due to labeling all cell viability classifications as acceptable cell viability during the model fitting process (**Fig. 5**).

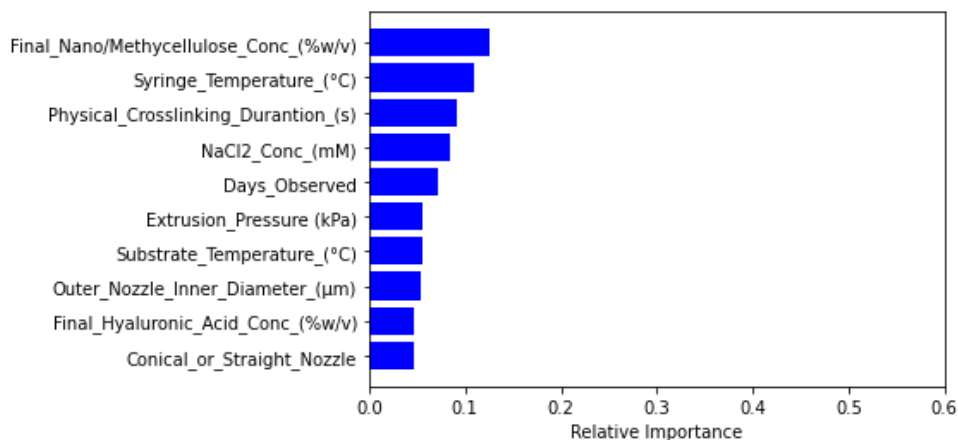


**Figure 5.** Confusion matrix of logistic regression and support vector classification of cell viability. 10% of the cell viability dataset was used as testing data while 90% of the dataset was used as training data. CV indicates cell viability.

Feature importance testing based on decision trees generated from random forest tree models indicated relatively major effects from extrusion pressure, specific material concentration, solvent choice, nozzle diameter, and printing temperatures for cell viability predictions (**Fig. 6-7**).



**Figure 6.** Feature importance rankings of material, equipment, and experimental parameters based on random forest regression modeling of cell viability



**Figure 7.** Feature importance rankings of material, equipment, and experimental parameters based on random forest classification modeling of cell viability

### 2.3.2. *Filament diameter model performance*

Amongst regression models, random forest regression models for filament diameter predictions also produced higher coefficients of determination while minimizing average mean squared error (**Table 3, Fig. S6-7**).

**Table 3.** Filament diameter regression model performance based on average values of coefficients of determination ( $R^2$ ) values and mean squared error (MSE) under 5-fold cross validation.  $R^2$  and MSE values were averaged from validation amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.

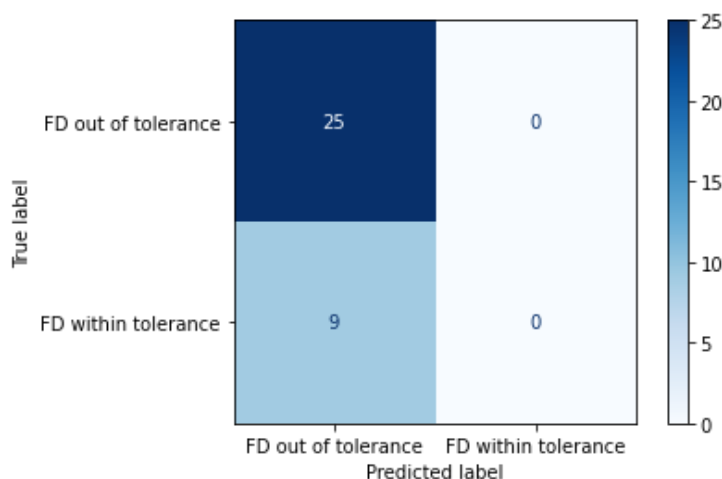
<b>Learner model</b>	<b>Average <math>R^2</math></b>	<b>Average MSE</b>
Random forest regression	0.645	0.007
Linear regression	0.543	0.009
Support vector regression	0.003	0.019

Similar to cell viability classification, when predicting acceptable filament diameter random forest classification models produced higher prediction accuracy, precision, and recall than other models tested (Table 4. Fig. S8-10).

**Table 4.** Filament diameter classification model performance based on average values of accuracy, precision, and recall under 5-fold cross validation. Accuracy, precision, and recall values averaged from validation amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.

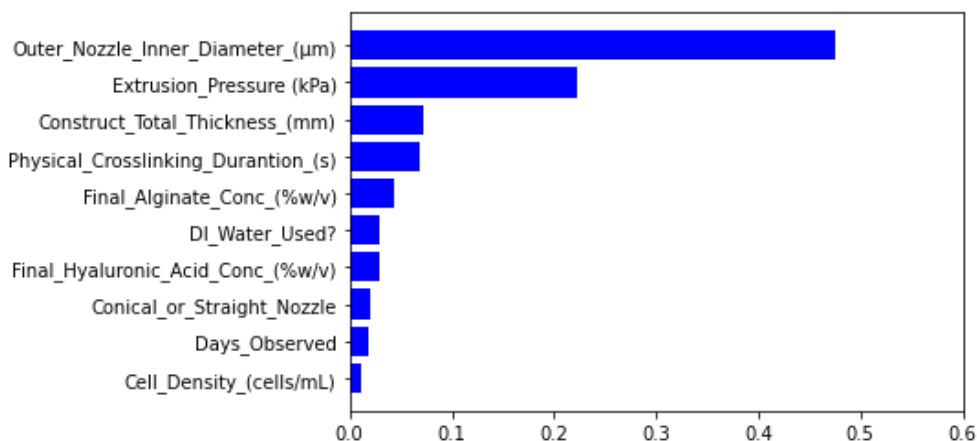
Learner model	Average Accuracy	Average Precision	Average Recall
Random forest classification	0.941	0.952	0.808
Logistic regression	0.846	0.753	0.569
Support vector classification	0.752	0.000	0.000

The support vector classification model generated precision and recall scores of zero due to labeling all filament diameter tolerance classifications as out of tolerance during the model fitting process (Fig. 8).

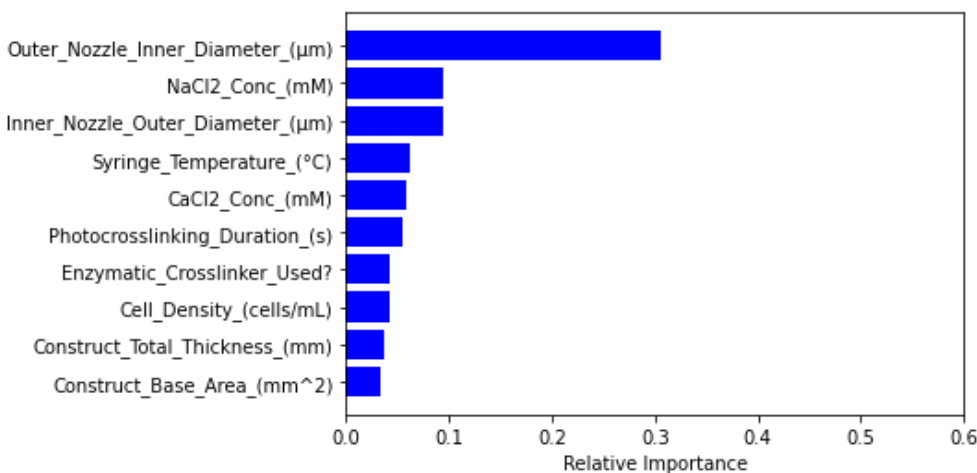


**Figure 8.** Confusion matrix of support vector classification of filament diameter. 10% of the filament diameter dataset was used as testing data while 90% of the dataset was used as training data. FD indicates filament diameter

Through feature importance analysis, nozzle diameter was ranked as the most important feature affecting filament diameter model prediction (**Fig. 9-10**).



**Figure 9.** Feature importance rankings of material, equipment, and experimental parameters based on random forest regression modeling of filament diameter



**Figure 10.** Feature importance rankings of material, equipment, and experimental parameters based on random forest regression modeling of filament diameter

### 2.3.3. Model predictions compared to experimental trends (non-primary cells)

Holding all but one input parameter constant, including the time of observation at zero days, both random forest and linear regression models translated several physical variable's impact onto prediction trends. Regression models predicted decreased cell viability with increasing alginate concentration (**Tables S37-39, 43-45**), increasing syringe temperature above 37 °C (**Table S50-51**) or increasing

extrusion pressure (**Table S42, S58**). The linear regression model also discerned trends reflective of the dataset. Specifically, lower extrusion pressures result in higher cell viabilities (**Table S56**) when increasing syringe temperature (**Tables S50-51**), when increasing cell density (**Tables S61-63**), with decreasing gelatin concentrations (**Tables S59-61**), and with increasing nozzle size (**Tables 47-49**).

When predicting filament diameter, the random forest regression model predicted decreasing filament diameters when ionic crosslinking duration post-extrusion is above 9 minutes (**Table S15**), when extrusion pressure is increased up to 90 kPa (**Table S7**), and when nozzle diameter is decreased (**Table S8-9**).

The linear regression model further predicts smaller filament diameters when syringe temperature, printing substrate temperature, gelatin concentration, CaCl<sub>2</sub> concentration, and ionic crosslinking duration increased individually (**Tables S10-11, S14-16**). Furthermore, filament diameter increased when alginate concentration increased when predicted with linear regression (**Table S13**). Using random forest classification, filament diameters produced were deemed to be within tolerance when using nozzle diameters of 840 μm or larger.

#### *2.3.4. Effect of training data size on output predictions*

Through increasing the number of cross validation folds, R<sup>2</sup> increased while MSE performance saw minimal change for two random forest regression and linear regression on cell viability predictions (**Table 5, Fig. S11-16**).

**Table 5.** Cell viability regression model performance based on average values of coefficients of determination ( $R^2$ ) values and mean squared error (MSE) under 2-fold, 5-fold, and 10-fold cross validation.  $R^2$  and MSE values were averaged from validation amongst all n combinations of one fold being trained and tested on the remaining n-1 folds.

<b>Learner model</b>	<b>Number of folds</b>	<b>Average <math>R^2</math></b>	<b>Average MSE</b>
Random forest regression	2	0.310	0.022
	5	0.384	0.019
	10	0.377	0.019
Linear regression	2	0.109	0.028
	5	0.231	0.024
	10	0.219	0.024
Support vector regression	2	-0.015	0.032
	5	0.000	0.031
	10	-0.007	0.031

For random forest cell viability classification, we can see that accuracy, precision, and recall stayed consistent with increased number of folds and in turn, training set size for cell viability (**Table 6, Fig. S17-19**).

**Table 6.** Random forest classification model performance for cell viability based on average values of coefficients of accuracy, precision, and recall scores under 2-fold, 5-fold, and 10-fold cross validation. Accuracy, precision, and recall values were averaged from validation amongst all n combinations of one fold being trained and tested on the remaining n-1 folds.

<b>Learner model</b>	<b>Number of folds</b>	<b>Average Accuracy Score</b>	<b>Average Precision Score</b>	<b>Average Recall Score</b>
Random forest classification	2	0.708	0.700	0.925
	5	0.689	0.678	0.942
	10	0.703	0.693	0.936
Logistic regression	2	0.616	0.616	1.000
	5	0.616	0.616	1.000
	10	0.616	0.616	1.000
Support vector classification	2	0.616	0.616	1.000
	5	0.616	0.616	1.000
	10	0.616	0.616	1.000

For filament diameter modeling, random forest regression model saw minimal effects due to training data size while linear regression saw large increases in  $R^2$  and decrease in MSE as the number of cross validation folds increased from two to five (**Table 7, Fig. S20-23**).



**Table 7.** Filament diameter regression model performance based on average values of coefficients of determination ( $R^2$ ) values and mean squared error (MSE) under 2-fold, 5-fold, and 10-fold cross validation.  $R^2$  and MSE values were averaged from validation amongst all n combinations of one fold being trained and tested on the remaining n-1 folds.

<b>Learner model</b>	<b>Number of folds</b>	<b>Average <math>R^2</math></b>	<b>Average MSE</b>
Random forest regression	2	0.639	0.007
	5	0.645	0.007
	10	0.604	0.008
Linear regression	2	-5.256	0.115
	5	0.543	0.009
	10	0.491	0.009
Support vector regression	2	0.005	0.019
	5	0.003	0.019
	10	-0.025	0.019

Accuracy, precision, and recall did not see significant changes regardless of increasing training data size (Table 8, Fig. S24-29).

**Table 8.** Classification model performance for filament diameter based on average values of accuracy, precision, and recall values under 2-fold, 5-fold, and 10-fold cross validation. Accuracy, precision, and recall values were averaged from validation amongst all n combinations of one fold being trained and tested on the remaining n-1 folds.

<b>Learner model</b>	<b>Number of folds</b>	<b>Average Accuracy Score</b>	<b>Average Precision Score</b>	<b>Average Recall Score</b>
Random forest classification	2	0.926	0.912	0.785
	5	0.941	0.952	0.808
	10	0.935	0.951	0.799
Logistic regression	2	0.829	0.726	0.514
	5	0.846	0.753	0.569
	10	0.832	0.764	0.517
Support vector classification	2	0.752	0	0
	5	0.752	0	0
	10	0.752	0	0

### 2.3.5. *Extrusion pressure recommendation prediction performance*

Amongst regression models, random forest regression models for extrusion pressure predictions also produced higher coefficients of determination while minimizing average mean squared error (**Table 9. Fig. S30-31**).

**Table 9.** Cell viability regression model performance based on average values of coefficients of determination ( $R^2$ ) values and mean squared error (MSE) under 2-fold, 5-fold, and 10-fold cross validation.  $R^2$  and MSE values were averaged from validation amongst all n combinations of one fold being trained and tested on the remaining n-1 folds.

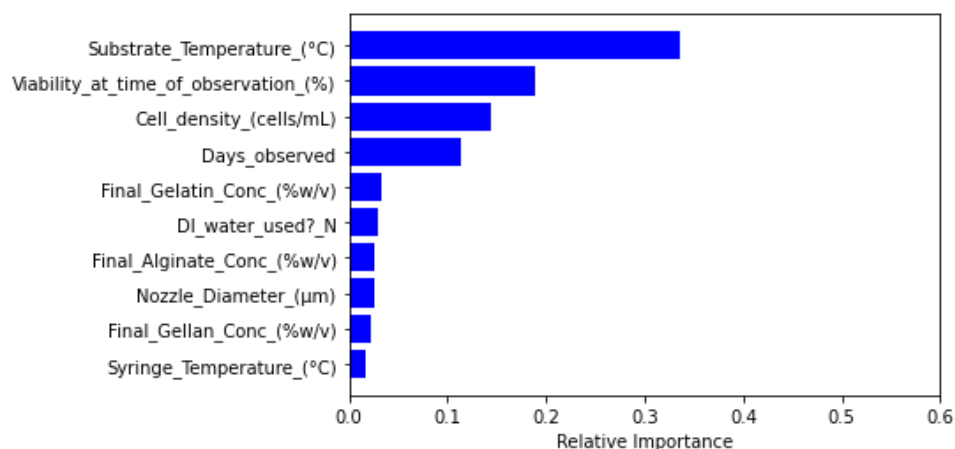
<b>Learner model</b>	<b>Number of folds</b>	<b>Average <math>R^2</math></b>	<b>Average MSE</b>
Random forest regression	2	0.540	0.041
	5	0.636	0.030
	10	0.623	0.030
Linear regression	2	0.498	0.045
	5	0.420	0.047
	10	0.418	0.046
Support vector regression	2	0.567	0.039
	5	0.515	0.040
	10	0.507	0.040

Similar to cell viability classification, when predicting acceptable filament diameter random forest classification models produced higher prediction accuracy, precision, and recall than other models tested (**Table 10. Fig. S32-34**).

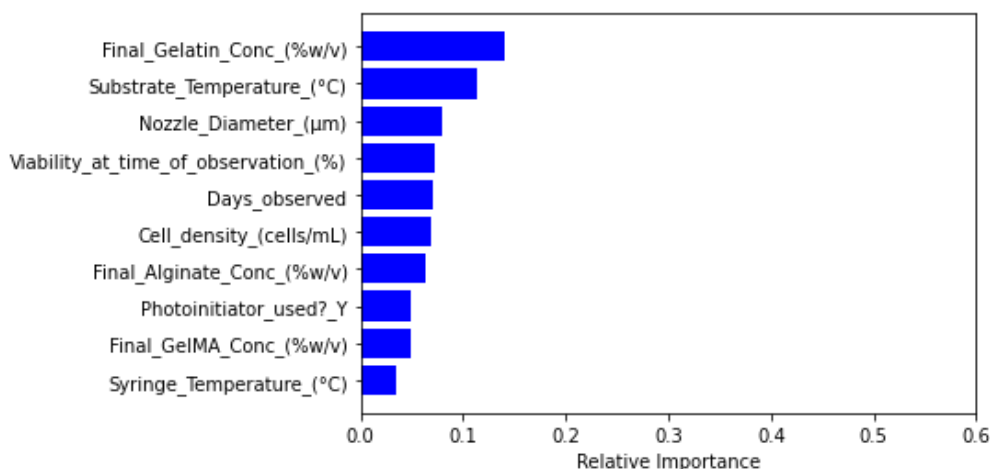
**Table 10.** Classification model performance for extrusion pressure based on average values of accuracy, precision, and recall values under 2-fold, 5-fold, and 10-fold cross validation. Accuracy, precision, and recall values were averaged from validation amongst all n combinations of one fold being trained and tested on the remaining n-1 folds.

<b>Learner model</b>	<b>Number of folds</b>	<b>Average Accuracy Score</b>	<b>Average Precision Score</b>	<b>Average Recall Score</b>
Random forest classification	2	0.763	0.807	0.701
	5	0.760	0.817	0.696
	10	0.768	0.814	0.701
Logistic regression	2	0.720	0.775	0.643
	5	0.729	0.802	0.639
	10	0.726	0.789	0.650
Support vector classification	2	0.709	0.775	0.618
	5	0.729	0.827	0.604
	10	0.723	0.819	0.601

Both random forest regression and linear regression models indicated increased pressure needed with higher alginate and gelatin concentrations, although the random forest regression model predicted a lower range of extrusion pressures while the linear regression model predicted a higher range. Based on feature importance rankings, substrate temperature appears to be the most significant variable impacting extrusion pressure used (**Fig. 11-12**).



**Figure 11.** Feature importance rankings of material, equipment, and experimental parameters based on random forest regression modeling of extrusion pressure



**Figure 12.** Feature importance rankings of material, equipment, and experimental parameters based on random forest classification modeling of extrusion pressure

## 2.4 Discussion

In this study, we approached the application of ML to bioprinting in two ways. First, we applied regression and classification models to data derived from a single study, which is more directly comparable to published studies on ML in bioprinting. In addition, we went further and applied the same ML techniques to a larger dataset encompassing results from 75 different studies to understand whether this data aggregation approach can effectively widen the area of model applicability. The random forest regression, random forest classification, and linear regression models created can be used to an extent in

conjunction with one another for outcome prediction as well as condition recommendation. Both random forest and linear regression models, more so the linear regression models, have shown the ability to represent several physical phenomena that have been documented in previous bioprinting or hydrogel studies. In particular, the general trends of increased extrusion pressure and alginate concentration resulting in decreased cell viability prediction values correlates with findings in previous literature that indicate increasing alginate concentration results in decreased cell viability [51,65,66]. In other cases, trends of predicted values oppose what is seen in literature [67,68]. These trends include decreased cell viability with increasing nozzle diameter (**Tables S45-46**), increased cell viability with increasing gelatin concentration (**Tables S54-56**), and larger filament diameter in DMEM-based bioink compared to saline solution-based bioink (**Tables S12**).

Compared to other ML models created for bioprinting predictions, the regression models created in this study provided lower  $R^2$  values and comparable error with similar proportion of training data to test data while accuracy of classification models were lower as well [59,62]. A major reason for this is the difference in experiment variation for the datasets used to create the models. Input parameters gathered from published studies contained a limited number of independent variables due to the chosen experimental design which focused on answering a specific research question versus parameter optimization. In addition, our dataset is inherently heterogeneous due to being acquired from studies conducted using different testing conditions and printing strategies. Comparatively, past studies contain larger amounts of data collected from controlled experimental settings [59]. Not all of the input conditions used in developing the ML models were reported in every study included in the dataset. Although missing data can be estimated using imputation, this can lead to misrepresentation of the features' weight on the output parameters and consequently lead to worse performance metrics along with prediction values that do not correlate with experimental results.

Feature importance ranking results indicated that cell density as a parameter did not carry as great of a weight in random forest predictive function for cell viability compared other bioink and

equipment parameters. Increasing cell density in bioink has been shown to marginally improve cell viability in the short term (0 to 1 day post-printing) for primary cells and stem cells [69,70]. Increasing cell density may also lead to an increase in cell agglomerates. In cases such as cell densities above  $5.0 \times 10^6$  cells/mL, cell viability decreases drastically the longer printed constructs are cultured (7 to 21 days) [70]. This can be due to the creation of hypoxic conditions for cells in inner areas of cell agglomerates which limits nutrient and waste transport through cell structure. In the cell viability dataset, the correlation of cell density on cell viability does not result in notable trends when cell density increases. Amongst the 617 instances used for cell viability model training, only 196 instances used cell densities above  $5.0 \times 10^6$  cells/mL. Amongst those instances, 65.3% of cell viability are acceptable ( $\geq 80\%$ ). This is a similar distribution to the cell viability value distribution in the overall dataset, where 61.6% of cell viability values are deemed acceptable ( $\geq 80\%$ ). In addition, the majority of unacceptable ( $< 80\%$ ) cell viability amongst instances containing more than  $5.0 \times 10^6$  cells/mL corresponded with cell density values between  $5.0$  to  $10.0 \times 10^6$  cells/mL while instances with higher cell concentrations saw smaller portions of cell viability values being unacceptable.

Compared to other ML models created for bioprinting predictions, the regression models created in this study provided lower  $R^2$  values and comparable error with similar proportion of training data to test data while accuracy of classification models were lower as well [59,62]. A major reason for this is the difference in experiment variation for the datasets used to create the models. Input parameters gathered from published studies contained a limited number of independent variables due to the chosen experimental design which focused on answering a specific research question versus parameter optimization. In addition, our dataset is inherently heterogeneous due to being acquired from studies conducted using different testing conditions and printing strategies. Comparatively, past studies contain larger amounts of data collected from controlled experimental settings [59]. Not all of the input conditions used in developing the ML models were reported in every study included in the dataset. Although missing data can be estimated using imputation, this can lead to misrepresentation of the

features' weight on the output parameters and consequently lead to worse performance metrics along with prediction values that do not correlate with experimental results.



### **3. Experimental evaluation of supervised classification and regression models**

#### **Adapted from:**

1. S. Tian, Machine Assisted Experimentation of Extrusion-based Bioprinting Systems, (2021). <https://www.mdpi.com/2072-666X/12/7/780>.

#### **3.1 Introduction**

Creating ML models that result in relatively high-performance metrics such as high correlation of determination or high prediction accuracy indicate the chosen algorithm's ability to recognize specific patterns well amongst input variables within the information limits of the dataset. However, using trained models to elicit outcomes and condition recommendations based on combinations of input and output variables not present within the training dataset can result in predictions that are out of expected tolerance. For bioprinting, a primary goal of ML models is to be used for determining accurate predicted results for myriads of printing and material settings, of which these settings may not be present in datasets used for training predictive models. Comparing predictions based on these settings to experimental results using the same settings can allow users to evaluate strengths and limitations of models and develop improvement methods to the models such as the use of hyperparameter tuning. The additional experimental data gain can be added for model training as well.

Here, we experimentally evaluate for the first time to our knowledge, cell viability, filament diameter, and extrusion pressure predictions of bioinks of different polymer precursor concentrations to bioprinting regression and classification model predictions. Specifically, polymer precursor concentration was varied through changing gelatin precursor concentration to evaluate cell viability and filament diameter predicted values while alginate and gelatin precursor concentrations were varied to evaluate extrusion pressure predicted values.

## 3.2 Materials and methods

### 3.2.1. Biomaterial ink synthesis

Sodium alginate powder (Sigma W201502) and gelatin (type B, 300 bloom derived from bovine, Sigma G9382) were sterilized under UV radiation for 30 minutes. After-wards, the powders were dissolved in complete cell culture media composed of Dulbecco's Modified Eagle Medium (DMEM, Gibco), 10% fetal bovine serum (FBS, Life Technologies) and 1% penicillin-streptomycin (Gibco). The mixtures were heated to 50°C and magnetically stirred for 4 hours. Complete mixtures were then vortexed for 1 minute and centrifuged at 167 RCF for 3 minutes to eliminate bubbles. Hydrogels were stored at 4°C prior to experimentation. Concentrations of sodium alginate (Alg) and gelatin (Gel) mixtures in complete media are denoted as Alg/Gel in units of %w/v. Extrusion of bioinks and biomaterial inks were conducted at 22.5°C. The 100mM CaCl<sub>2</sub> solution used to crosslink printed constructs was prepared by dissolving CaCl<sub>2</sub> (Sigma-Aldrich) in complete cell culture media and sterile filtering through a 0.22 µm syringe filter (Millipore).

### 3.2.2. Cell culture maintenance

Mouse neuroblastoma cells (N2A, CCL-131 cell line, American Type Culture Collection, ATCC) were cultured at 37 °C in humidified 5% CO<sub>2</sub> atmosphere using complete cell culture media in T75 cell flasks (Falcon™, Corning). Cells were passaged every 4 to 5 days with 0.05% trypsin/EDTA (Gibco) and a portion was split for use to prepare bioinks for printing.

### 3.2.3. Bioink synthesis and construct printing

Alg/Gel hydrogels were heated up to 37 °C prior to mixing with cells. Cell suspensions containing  $1.0 \times 10^6$  trypsinized cells were centrifuged to create cell pellets for mixing. A cell density of  $1.0 \times 10^6$  cells/mL was chosen due to it being the most common cell density used amongst studies used to compile the training dataset. To this cell pellet, 1mL of liquified hydrogel was added using a 10 mL syringe (BD Falcon) and then triturated using a pipet for 30 seconds to mix thoroughly. The mixture was then aspirated into a 10 mL syringe and transferred to a 3 mL cartridge (Nordson EFD) via a female

to female luer lock connection. The bioink was then held at room temperature to allow complete gelation. The duration of complete gelation depends on the concentration of sodium alginate and gelatin used. Once gelation is reached, the 3 mL cartridge is then secured onto an extrusion-based bioprinter (INKREDIBLE, Cellink). For cell viability testing, 80 mm x 80 mm x 0.8 mm models were printed at a feed rate of 10 mm/s into 24 well plates at 22.5°C. 22G conical nozzles (Nordson EFD) were used. For confocal microscopy imaging, models were printed onto sterile cover glass slides. Directly after printing completion, pictures of constructs were taken, and constructs were exposed to 100 mM CaCl<sub>2</sub> crosslinking solution for 1 minute. Afterwards, remaining crosslinking solution was aspirated and constructs were rinsed with Dulbecco's PBS (DPBS, Gibco, 7.4 pH). The constructs were then incubated at 37 °C with 5% CO<sub>2</sub> with complete cell culture medium.

#### 3.2.4. *Live/dead staining:*

N2A cell viability was determined by staining cells with Hoechst 33342 (40.6 μM) and propidium iodide (19.7 μM) dye solutions (Readyprobes, ThermoFisher) following the manufacturer's protocol. Briefly, cell culture media was aspirated and replaced with DPBS containing 1 drop of Hoechst 33342 and 1 drop of propidium iodide then incubated for 15 minutes incubation at 37 °C with no light exposure. Excitation/emission wave-lengths of 358/461 nm and excitation/emission wavelengths of 580/604nm were used to image Hoechst 33342 and propidium iodide stained cells respectively using an imaging plate reader (Cytation 3, BioTek). Z-stack images of stained cells in bioink were taken through confocal microscopy (LSM 710, Zeiss). Cell counting for cell viability was conducted with Cytation 3 Cell Imaging software. Cell viability was determined by dividing the total number of cells (total number of Hoechst 33342 stained cells subtracted by the number of dead cells stained from propidium iodide) by the total number of Hoechst 33342-stained cells.

#### 3.2.5. *Filament diameter measurements:*

Constructs were imaged using an imaging plate reader (Cytation 3, BioTek). Collected images were analyzed using ImageJ (<https://imagej.nih.gov/ij/>) for filament diameter length.

### 3.2.6. *Extrusion pressure measurements:*

Using cell viability dataset instances with available extrusion pressure values (353 instances), random forest regression and linear regression models were created to predict extrusion pressure values needed to extrude specific material concentrations to produce 80% cell viability. Bioinks of 3/4 Alg/Gel, 3/7 Alg/Gel and 8/20 Alg/Gel were used to test extrusion pressure predictability within and near the edge of material concentration bounds of the dataset used. Material preparation procedure for testing extrusion pressure is the same as in section 3.2.1.

### 3.2.7. **Intrastudy model creation and usage**

A comparison of general dataset predictive ability was done with a selected study that used alginate and gelatin multicomponent hydrogel [71]. 16 instances of unique cell viability outcomes from material and equipment parameters were used to create a random forest classification and regression model, as well as a linear regression and support vector regression model for cell viability. Filament diameter trend was also produced from four filament diameter data points corresponding to different material and pressure combinations through multiple regression. Two cell viability values, one based on parameter values within range of the intrastudy dataset, and another based on parameter values out of range of the intrastudy dataset, were predicted for and compared against predicted values of the overall dataset. Filament diameter of constructs printed with an alginate and gelatin multicomponent bioink was compared against the intrastudy regression model as well as with the random forest regression model predictions made for the same material and equipment parameters. Since only 4 filament diameter values were provided in the specific study, a fitted regression model was used. A multiple linear regression was fit to data correlating extrusion pressure and alginate concentration with filament diameter, resulting in a regression equation (**Equation 5**) of:

$$z = Ax + By + C \quad (5)$$

where  $A = 333.26$ ,  $B = -0.245$ , and  $C = -781.4$ . The variable  $x$  represents the alginate concentration (% w/v),  $y$  is the extrusion pressure (kPa), and  $z$  is the filament diameter ( $\mu\text{m}$ ).

### 3.2.8. *Statistical analysis:*

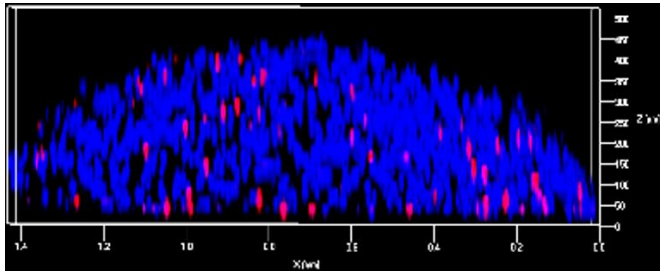
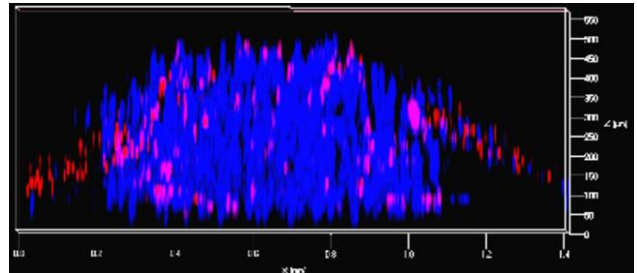
Cell viability, filament diameter, and extrusion pressure measurements were expressed as mean  $\pm$  standard deviation. Statistical significance between any two groups of either cell viability, filament diameter, and extrusion pressure measurements was tested through one-way ANOVA with the significance level set as  $p < 0.05$ . Percent error was calculated for experimental cell viability, filament diameter, and extrusion pressures as compared to predicted values.

## 3.3 Results

### 3.3.1. *Effect of specified training data on cell viability*

Using the complete cell viability dataset for model training, the random forest regression model resulted in a predicted cell viability of 73.1% for a material combination of 3/4 Alg/Gel, 100 mM CaCl<sub>2</sub> crosslinking solution with an exposure duration of 60 seconds, and extrusion through a 22G conical nozzle at room temperature (22.5 °C). For another material combination with 3/7 Alg/Gel, 100 mM CaCl<sub>2</sub> crosslinking solution with an exposure duration of 60 seconds, and extrusion through a 22G conical nozzle at room temperature (22.5 °C), the random forest regression model predicted the same cell viability value of 71.7%.

A specific study was used to create an alginate and gelatin-focused dataset for random forest regression model training [71]. For 3/4 and 3/7 Alg/Gel, a random forest regression model created from this specified dataset resulted in a cell viability prediction of 91.0% both material combinations when extrusion pressure was set constant. Actual cell viability of values gathered from live/dead staining showed a larger number of dead cells present directly after printing in 3/7 Alg/Gel than 3/4 Alg/Gel constructs (**Fig. 13, S35-38**).

**A****B**

**Figure 13.** Total/dead confocal images in the cross-sectional view in the X-Z plane of **A)** 3/4 Alg/Gel and **B)** 3/7 Alg/Gel immediately after extrusion of **A)** 3/4 Alg/Gel and **b)** 3/7 Alg/Gel. The frames of images are 1.4 by 0.5 mm in dimension.

Resultant cell viability values for 3/4 and 3/7 Alg/Gel constructs are  $85.2\% \pm 9.1$  and  $64.2\% \pm 10.6\%$  (**Table 11**). Random forest classification, logistic regression, and support vector regression models predicted acceptable cell viability for both material conditions based on tested material concentration and printing parameters (**Table S1**). All predictions were made keeping extrusion pressure constant at 95.4 kPa.

**Table 11.** Predicted cell viability values are compared against experimental values for corresponding material concentrations of alginate and gelatin. Actual values represent the mean  $\pm$  standard deviation for all samples (n = number of samples) measured from at least 3 batches of Alg/Gel bioink.

<b>Prediction Model</b>	<b>Material concentration (%w/v)</b>	<b>Predicted viability (%)</b>	<b>Actual viability (%)</b>	<b>Error (%)</b>
Random forest regression, complete dataset	3/4 Alg/Gel	73.1	85.2 + 9.1 (n = 8)	16.6
	3/7 Alg/Gel	71.7	64.2 $\pm$ 10.6 (n = 11)	10.5
Linear regression, complete dataset	3/4 Alg/Gel	91.0	85.2 + 9.1 (n = 8)	6.37
	3/7 Alg/Gel	91.0	64.2 $\pm$ 10.6 (n = 11)	29.5
Random forest regression, intrastudy dataset	3/4 Alg/Gel	74.0	85.2 + 9.1 (n = 8)	15.1
	3/7 Alg/Gel	75.3	64.2 $\pm$ 10.6 (n = 11)	14.7
Linear regression, intrastudy dataset	3/4 Alg/Gel	-25.9	85.2 + 9.1 (n = 8)	429
	3/7 Alg/Gel	-25.9	64.2 $\pm$ 10.6 (n = 11)	348

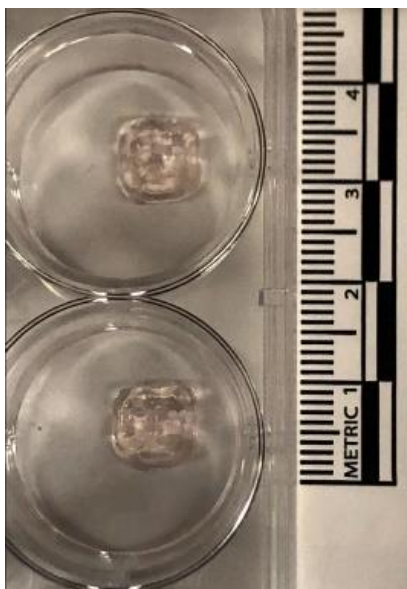
Using the complete filament diameter dataset for model training, the random forest regression model resulted in predicted filament diameters of 1073  $\mu\text{m}$  and 857  $\mu\text{m}$  for 3/4 and 3/7 Alg/Gel respectively (**Table 12**).

**Table 12.** Predicted filament diameter values are compared against experimental values for corresponding material concentrations of alginate and gelatin. Actual values represent the mean  $\pm$  standard deviation for all samples (n = number of samples) measured from at least 3 batches of Alg/Gel bioink.

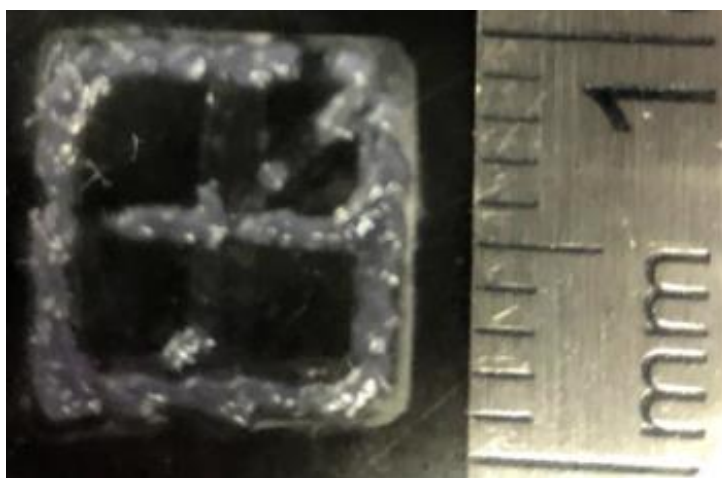
Prediction model	Material concentration (%w/v)	Predicted filament diameter ( $\mu\text{m}$ )	Actual value ( $\mu\text{m}$ )	Error (%)
Random forest regression	3/4 Alg/Gel	1037.3 $\mu\text{m}$ (prediction pressure = 25 kPa)	927.6 $\pm$ 106.0 $\mu\text{m}$ (n=8)	10.6
	3/4 Alg/Gel	752.2 $\mu\text{m}$ (predictive pressure = 103.3 kPa)	927.6 $\pm$ 106.0 $\mu\text{m}$ (n=8)	23.3
	3/7 Alg/Gel	857.3 $\mu\text{m}$ (predictive pressure = 75 kPa)	707.2 $\pm$ 146.1 $\mu\text{m}$ (n=11)	17.5
	3/7 Alg/Gel	752.2 $\mu\text{m}$ (prediction pressure = 103.3 kPa)	707.2 $\pm$ 146.1 $\mu\text{m}$ (n=11)	5.98
Linear regression	3/4 Alg/Gel	1275.8 $\mu\text{m}$ (prediction pressure = 25 kPa)	927.6 $\pm$ 106.0 $\mu\text{m}$ (n=8)	27.3
	3/4 Alg/Gel	1149.0 $\mu\text{m}$ (prediction pressure = 103.3 kPa)	927.6 $\pm$ 106.0 $\mu\text{m}$ (n=8)	19.3
	3/7 Alg/Gel	1187.1 $\mu\text{m}$ (prediction pressure = 75 kPa)	707.2 $\pm$ 146.1 $\mu\text{m}$ (n=11)	40.4
	3/7 Alg/Gel	1141.3 $\mu\text{m}$ (prediction pressure = 103.3 kPa)	707.2 $\pm$ 146.1 $\mu\text{m}$ (n=11)	38.0
Intrastudy linear regression	3/4 Alg/Gel	212.3 $\mu\text{m}$ (prediction pressure = 25 kPa)	927.6 $\pm$ 106.0 $\mu\text{m}$ (n=8)	337
	3/7 Alg/Gel	200.0 $\mu\text{m}$ (prediction pressure = 75 kPa)	707.2 $\pm$ 146.1 $\mu\text{m}$ (n=11)	254



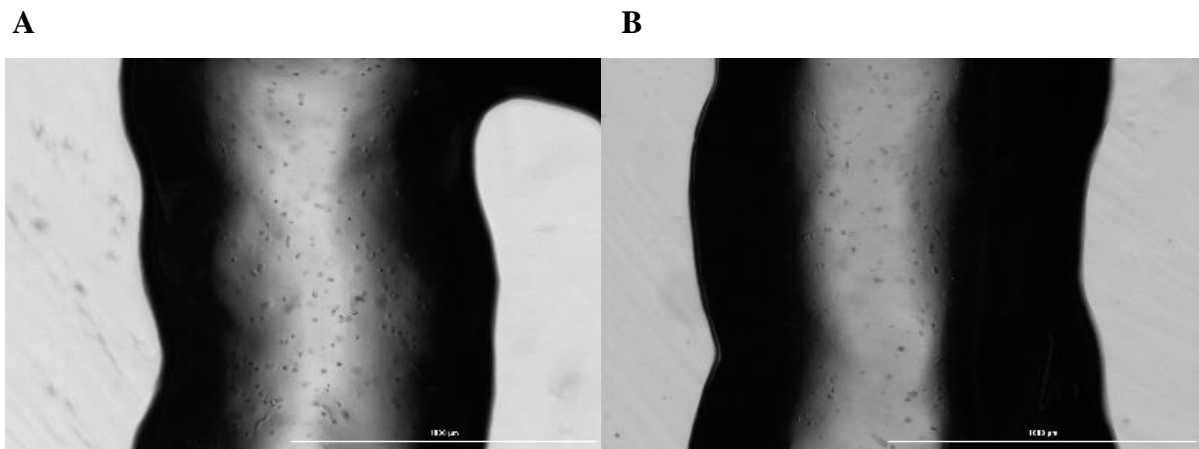
Filaments from constructs printed with 3/4 Alg/Gel resulted in  $1157 \pm 102.2 \mu\text{m}$  pre-crosslinking and  $927.6 \pm 106.0 \mu\text{m}$  after crosslinking. For 3/7 Alg/Gel, filament diameter pre-crosslinking was measured at  $817.0 \pm 107.7 \mu\text{m}$  while measuring at  $707.2 \pm 146.1 \mu\text{m}$  directly after crosslinking (**Fig. 14-17**).



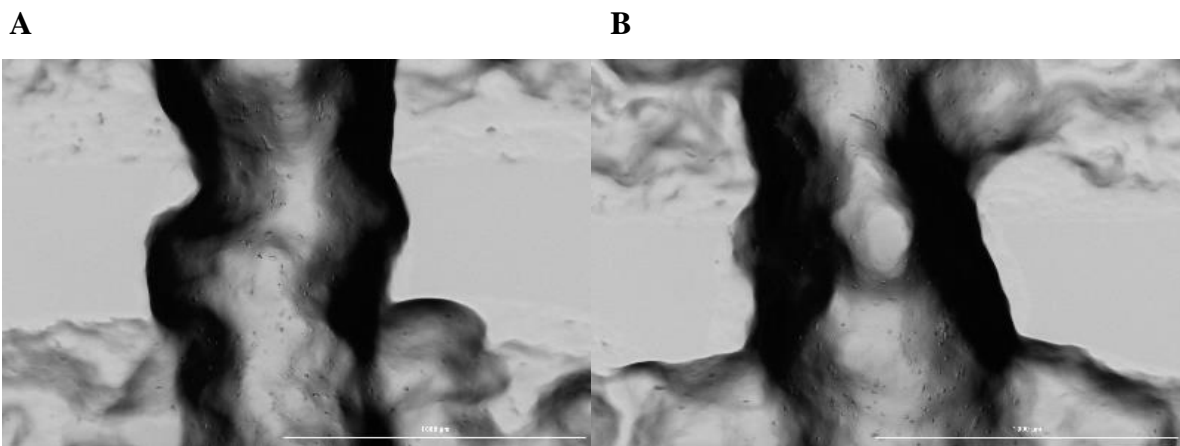
**Figure 14.** 3/4 Alg/Gel constructs directly after printing onto tissue-culture treated well plate surfaces. The scale bars depict  $1000 \mu\text{m}$



**Figure 15.** 3/7 Alg/Gel constructs directly after printing onto tissue-culture treated well plate surfaces. The scale bars depict  $1000 \mu\text{m}$



**Figure 16.** Brightfield images of **A)** 3/4 Alg/Gel and **B)** 3/7 Alg/Gel filaments directly after extrusion. The scale bars depict 1000  $\mu\text{m}$



**Figure 17.** Brightfield images of **A)** 3/4 Alg/Gel and **B)** 3/7 Alg/Gel filaments directly after extrusion. The scale bars depict 1000  $\mu\text{m}$

The percent error of the crosslinked filament diameters with respect to nozzle diameter (410  $\mu\text{m}$ ) is 126% and 72.5% for 3/4 Alg/Gel and 3/7 Alg/Gel constructs respectively, making them out of tolerance. All classification models predicted unacceptable filament diameter tolerance for both 3/4 Alg/Gel and 3/7 Alg/Gel combinations (**Table S2**).

### 3.3.2. *Effect of specified training data on extrusion pressure recommendation predictions*

Both random forest regression and linear regression models indicated increased pressure needed with higher alginate and gelatin concentrations, although the random forest regression model predicted

a lower range of extrusion pressures while the linear regression model predicted a higher range. Constructs printed using 3/7 Alg/Gel required an average extrusion pressure of 71.7 kPa. Bioink with 8/20 Alg/Gel was not able to form printed constructs due to high material viscosity, although over-deposited filament was extruded at an average pressure of 208.3 kPa. As material concentrations increased, prediction accuracy of random forest regression diminished while prediction accuracy of linear regression improved, as noted by percent error calculations (**Table 10**). The random forest classification model was able to predict acceptable extrusion pressure correctly for 3/4 Alg/Gel and 8/20 Alg/Gel, but not for 3/7 Alg/Gel. Meanwhile, logistic regression and support vector classification models predicted that all material concentration combinations printed under the same printing settings can result in using pressure within the acceptable pressure range (**Table S3**). All model predictions were conducted with desired cell viability set to 90% immediately after printing. In the cases of 3/4 Alg/Gel and 3/7 Alg/Gel, the pressure needed for extrusion and construct formation was smaller and resulting cell viabilities were also lower than 90% (**Table 3**).

**Table 13.** Predicted extrusion pressure required to deposit material are compared against experimental values for corresponding material concentrations of alginate and gelatin. Actual values represent the mean  $\pm$  standard deviation for all samples (n = number of batches).

<b>Prediction model</b>	<b>Material concentration (%w/v)</b>	<b>Predicted extrusion pressure (kPa)</b>	<b>Actual value (kPa)</b>	<b>Error (%)</b>
Random forest regression	3/4 Alg/Gel	56.9	37.3 + 8.7 (n = 3)	34.4
Random forest regression	3/7 Alg/Gel	150.6	83.7 $\pm$ 4.2 (n = 3)	44.4
Random forest regression	8/20 Alg/Gel	150.6	208.3 $\pm$ 6.2 (n = 3)	38.3
Linear regression	3/4 Alg/Gel	140.8	37.3 + 8.7 (n = 3)	73.4
Linear regression	3/7 Alg/Gel	162.9	83.7 $\pm$ 4.2 (n = 3)	48.6
Linear regression	8/20 Alg/Gel	240.0	208.3 $\pm$ 6.2 (n = 3)	13.2

## 3.4 Discussion

### 3.4.1. *Influence of cellular parameters has on predictive outcomes*

Previous EBB studies have shown that increasing pressure-induced shear stress on cells can cause decreases in cell viability for both immortalized cell lines and stem cells [8,31,48]. In a case with 10% w/v gelatin printed with HepG2 cells and 27 gauge conical nozzles, cell viability notably decreased from 96% to 84% when pressure increased from 200 to 300 kPa [31]. Blaeser et al. indicated notable decrease in cell viability of L929 fibroblasts encapsulated in alginate hydrogels when average shear stress within the printing orifice reached 5 kPa or above [48]. Specifically, cell viability drops from 96% in cases with less than 5 kPa average shear stress reduced to 91% cell viability within 5 to 10 kPa, further to 76% at higher shear stress values. In terms of pressure, a 5 kPa shear stress value corresponded to a pressure between 100 and 150 kPa when a 300  $\mu$ m cylindrical valve is used along with an alginate concentration of 1.0% w/v. Since N2A cells were used in validation experiments, cell viability behavior under shear stress would be similar to previous studies also using non-primary cell lines. Random forest classification was seen to produce varied prediction results in cases of primary cell usage with conical nozzles used. Testing the predicted effects of extrusion pressure on primary cells printed through conical nozzles, cell viability was found to become unacceptable above 20 kPa for 3/5 Alg/Gel while 3/8 Alg/Gel was found to have acceptable viability across pressures from 0 to 300 kPa. Increasing alginate concentrations, 5/2 Alg/Gel also saw unacceptable cell viability above 20 kPa while 5/4 Alg/Gel saw unacceptable viability only when above 270 kPa. When varying syringe cartridge temperature for printing primary cells, 3/5 Alg/Gel bioink saw unacceptable cell viability at temperatures above 20 °C while 3/8 Alg/Gel usage resulted in unacceptable cell viability at 36 °C or above. Interestingly, a 5/2 Alg/Gel material concentration resulted in unacceptable cell viability specifically at 23 °C as well as temperatures above 36 °C, while all other temperatures from 4 to 40 °C resulted in acceptable cell viability. When gelatin concentration increased to 4% w/v while alginate

concentration remained constant at 5% w/v (5/4 Alg/Gel), all predicted cell viability values up to 40 °C were acceptable. In the case of predicting suitable extrusion pressure, the use of primary cells resulted in a decrease of around 20 kPa less pressure needed for the same material concentration and printing setting as compared to using non-primary cells. Overall, to elucidate more straightforward modeling of primary cell viability behavior, more data gathered from studies using primary cells and straight nozzles is needed to understand if nozzle geometry imparts different biological effects for primary cells compared to non-primary cells.

For cell viability predictions, regression models hold promise for further development. The random forest filament diameter regression model offers greater prediction accuracy compared to the linear regression model based on percent error from actual filament diameter values. If the user knows the extrusion range suitable for their bioink, filament diameter predictions can become even more accurate (**Table 9**). Amongst all models, filament diameter prediction models mapped closest to experimental results if accounting for the prior knowledge of suitable pressure ranges to input for predictions. Unlike other predictive models, the nozzle diameter and extrusion pressure were relatively much more impactful variables to the model (**Figure 9-10**) as compared to the most important variables found through feature importance of other random forest models (**Figures 6-7, 11-12**). For extrusion pressure predictions, the random forest regression model underestimates required extrusion pressure while the linear regression model overestimates required pressure. Correction factors determined from uncertainty factor evaluation can be applied for these models to produce prediction outcomes closer to actual results. For the models in this study to be used effectively, users still need to have baseline knowledge of how material parameters and printing settings affect cell viability, filament diameter, and extrusion pressure needed, such as in the case of filament diameter regression models. Based on trends extracted from tuning different parameters, future experiments could focus on collecting more data for the variables where more data would improve the predictive power of the models.

The nature of how cell viability values are derived and calculated can play a large role in how representative they are of true biological conditions of cells within printed constructs. In studies using live/dead staining to derive cell viability values, how large the area of focus on the construct for cell counting is accounted for. A standard area of observation for a section of construct filament is not provided. In most cases, it is not clear at what focus the transverse plane of a filament is examined. Furthermore, whether specific sections of a construct are used, e.g., outer-boundary filament strands, at an intersection of filament in the middle of a construct, or randomly-selected sections are selected for cell viability measurements is not clear.

### 3.4.2. *Experimental errors and recommendations*

Cell viability values can be compared similarly amongst each other as live/dead staining was used to determine most cell viability values in all studies used for dataset compilation. However, due to the diverse methodology used to calculate cell viability in the field of bioprinting [14], cell viability values derived from different assays may not be simply grouped together for model creation due to the measurements of different biological endpoints. Assays that measure different endpoints than live/dead staining dyes used in this study (Calcein AM, propidium iodide, and ethidium homodimer), such as the MTT (3-(4,5-Dimethylthiazol-2-yl)-2,5-Diphenyltetrazolium Bromide) absorbance assay or the lactate dehydrogenase membrane integrity assay, can provide different relative viability values from colorimetric readings as compared to stained cell counting. Despite cell viability assay variation and disparate measurement procedures for live/dead cell staining assays, the random forest regression model's predicted cell viability values fall within normal experimental ranges. Building upon this study, a future direction can be to compare ML model robustness when trained on data composed of assays that use measure the same cellular endpoints.

Additional future directions of this work can be to apply experimental results to improve quantitative predictions. The use of first principle calculations can be used to estimate missing variables

in the dataset. For example, the Power-law or Herschel-Bulkley fluid behavior modeling can be used to find non-Newtonian index values of selected materials to then convert lengthwise and volumetric extrusion rates to missing extrusion pressures, and vice versa [10,21,29,48]. Additional non-linear learners, such as k-nearest neighbor classification and regression models, can be explored as models that generate higher prediction performance than existing models created.



## 4. Additional considerations for EBB predictive modeling

### 4.1 Relationship between extrusion pressure and material concentration in Alg/Gel hydrogel

#### 4.1.1. Introduction

As evident through **Equation 6**, higher viscosity ( $\eta$ ) hydrogel require larger extrusion to produce flow [72].

$$\Delta P = 32\eta Lv \left(\frac{3n+1}{4n}\right) \left(\frac{d^2}{D^4}\right) \quad (6)$$

Variables in the equation represent the same physical attributes as in **Equations 3 and 4**. Here,  $d$  indicates filament diameter. Incorporation of a relationship between material concentration and extrusion pressure can simplify ML model training through reduction of input variables by only requiring one variable type once the relationship is clearly defined. From literature, relationships between alginate and gelatin composite material concentration and printability have been elucidated for varied polymer concentrations, although a systematic examination of layer deposition ability has not been elucidated yet [24]. In terms of printability, an optimal range exists where filament is defined as suitably extruded when no hydrogel beading occurs due to insufficient extrusion pressure nor warpage of extruded filament due to excessive pressure. A preliminary study on a range of alginate/gelatin composite hydrogels was conducted to determine optimal material concentrations to produce suitable filament deposition ability without exceeding pressure limits that can result in significant shear-stress induced cell death [64].

#### 4.1.2. Materials and methods

##### 4.1.2.1. Biomaterial ink synthesis

Sodium alginate powder (Sigma W201502) and gelatin powder (type B, 300 bloom derived from bovine, Sigma G9382) at different concentrations were dissolved in complete cell culture media composed of Dulbecco's Modified Eagle Medium (DMEM, Gibco), 10% fetal bovine serum (FBS, Life Technologies) and 1% penicillin-streptomycin (Gibco). For instance, to synthesize an alginate-gelatin

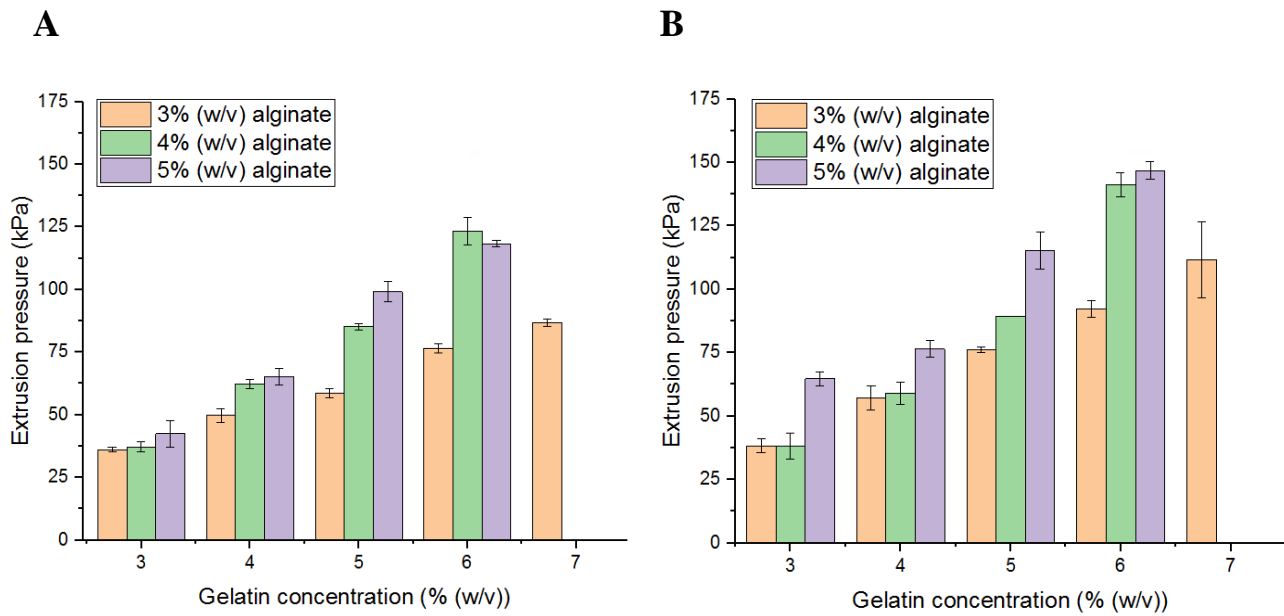
composite hydrogel with 3% and 7% (w/v) alginate and gelatin concentration, 0.150 g of sodium alginate powder and 0.350 g of gelatin powder were weighed and transferred into a scintillation vial and 5 mL of complete cell culture media was added afterwards with a magnetic stirbar. The mixtures were heated to 50°C and magnetically stirred for 2-4 hours. Complete mixtures were then vortexed for 3 minutes and centrifuged at 1,663 RCF for 3 minutes to eliminate bubbles. Hydrogels were subsequently refrigerated at 4°C prior to experimentation and left at room temperature (22.5 °C) for 4 hours prior to printability testing. Similar to biomaterial ink nomenclature in *Chapter 3*, concentrations of Alg and Gel mixtures in complete media are denoted as Alg/Gel in units of % w/v.

#### 4.1.2.2. *Biomaterial ink printing*

Extrusion of bioinks and biomaterial inks were conducted at 22.5 °C with a 22G and a 25G conical nozzle (EFD Nordson). All biomaterial inks were deposited onto glass petri dishes with the layer height (distance between the nozzle and substrate for the first printing layer) of deposition set at the same length as the inner diameter of the nozzle used. For instance, using a 22G inner nozzle diameter corresponded to a 410 µm offset height. 10 by 10 mm constructs with two filament layers were oriented to access printability. The printing feed rate was set to 10 mm/s.

#### 4.1.3. *Results and discussion*

A range of 3 to 5 % w/v alginate and a range of 3 to 6 % w/v gelatin were combined to synthesize Alg/Gel combinations from 3/3 Alg/Gel to 5/6 Alg/Gel. 3/7 Al/Gel was also synthesized and printed. When keeping the alginate concentration constant while increasing gelatin concentration, hydrogel larger extrusion pressure needed to produce printed constructs that exhibit consistent filament deposition (also described as having a lack of filament discontinuity) (**Fig. 18**).



**Figure 18:** Extrusion pressure required to produce filament deposits of two layers extruded at various gelatin concentration extruded from **A**) a 22G conical nozzle and **B**) a 25G conical nozzle. Bars with the same color indicate constant alginate concentration.

At constant gelatin concentrations, the greatest rate of extrusion pressure increase was observed when alginate concentration increased from 3% to 4% when printed with 22G and 25G conical nozzles. As gelatin concentration increases, the rate of increase in extrusion pressures with increased alginate concentration also increases, indicative of increased stiffness with increased alginate concentration [13]. At constant alginate concentrations, rate of extrusion pressure also increases with increased gelatin concentrations. Extrusion pressure required to produce consistent and stacked filaments at alginate and gelatin concentrations at or above 3% and 4% respectively were varied in absolute values compared to literature results [24,71]. Compared to identical Alg/Gel concentrations of 3/4 and 4/4% (w/v), extrusion pressure require to produce high structure fidelity with a 22G nozzle was lower in literature for 3/4 Alg/Gel, with an extrusion pressure of 20 kPa compared to an average extrusion pressure value of 36.1 kPa and higher in literature for 4/4 Alg/Gel at 120 kPa as compared to 49.7 kPa [71]. Potential sources of error causing this deviation in printability results from literature can be a difference in methods of printability evaluation. Printability in the referenced literature was not defined explicitly.

Furthermore, the offset distance in the referenced literature was set at 100  $\mu\text{m}$  as compared to 410  $\mu\text{m}$  and 260  $\mu\text{m}$  in our study, the solvent used for hydrogel synthesis was PBS as opposed to DMEM in our methods, and the feed rate of the 3D extrusion setup in literature was never provided. Compared to a similar study by Gao et al. where DMEM is used as solvent for a polymer concentration of 5/6 Alg/Gel, the extrusion pressure required to produce multilayer filament deposits is higher in our study (125 kPa versus 90 kPa) [24]. This higher pressure required is attributed to the faster feed rate (600 mm/min versus 200 mm/min), allowing less hydrogel material to be deposited with less drag force exerted on deposits from the nozzle. With this consideration, a relationship incorporating feed rate along with polymer precursor concentration and extrusion pressure can be derived to further reduce the number of variables needed for an EBB-based dataset used for ML training. Such relationship can take form as a multiple regression model incorporating the effect of polymer material concentration and extrusion pressure. Also, if possible, adding an input variable of nozzle offset in ML training datasets can be useful in determining the relationship of polymer precursor concentration with extrusion pressure needed to produce consistent filament deposits. This additional variable can also be incorporated with regression-based relationships amongst feed rate, material concentration, and extrusion pressure.

## **4.2 Effect of support baths**

The use of support baths containing media with the ability to rapidly transition from a solid-like state to a fluidic state and vice versa is a common method to allow for printing of low viscosity hydrogels. This self-healing property allows for encapsulation of hydrogel to prevent hydrogel flow and maintain printed structure. In particular, the technique of extruding hydrogel into a Bingham plastic slurry composed of gelatin particles of several hundreds of micrometers in diameter in a solvent containing divalent ions or photocrosslinking agents is particularly popular. This technique is termed FRESH, in short for freeform reversible embedding of suspended hydrogels [73]. Other forms of support baths used include the use of gellan microgels [74], modified hyaluronic acid [75], and alginate

microparticles [76]. Despite the proposed effect of support baths retaining hydrogel filament geometry, the synthesis method of the support bath and the retrieval process of the printed constructs can lead to large variation in printability, specifically in filament diameter size.

#### 4.2.1. *Materials and methods*

##### 4.2.1.1. *Synthesis of FRESH gelatin slurry*

To create the gelatin slurry support bath, 4.5% (w/v) gelatin (Type A, Thermo Fisher Scientific) was mixed in 150 mL of 11 or 100 mM CaCl<sub>2</sub> (Sigma-Aldrich) into a solution and then gelled it for 12 hours at 4°C in a 500-ml mason jar (Ball Inc.). Next, 350 ml of 11 or 100 mM CaCl<sub>2</sub> at 4°C was added to the jar to fill the jar to the brim. This step helps to prevent air bubble formation and to reduce undesired heating and fusing of gelatin particle as much as possible. The contents of the jar were blended at “pulse” speed setting for 120 seconds at 30 second intervals followed by additional 30 seconds of rest. The rest period prevents overheating from prolonged frictional heat generation from the blender blades. After blending, a gelatin slurry was formed and poured into 50 mL conical tubes (Falcon, Corning) and centrifuged at 3260 RCF for 2 minutes to separate excess supernatant with solubilized gelatin and slurry microparticles. Excess supernatant was aspirated and replaced with up to 10 mL of 11 mM or 100 mM CaCl<sub>2</sub> at 4°C. The slurry was then vortexed for 1 minute to create a homogenous suspension and centrifuged again for supernatant separation. The process was repeated until no bubbles or foam were observed at the top of the supernatant, which indicated that most of the soluble gelatin was removed. The remaining slurry was refrigerated at 4 °C until future use. Slurry properties remained stable for four weeks.

##### 4.2.1.2. *Alginate hydrogel synthesis*

0.100 g of sodium alginate powder (Sigma W201502) powder was dissolved in 5 mL of complete cell culture media composed of Dulbecco’s Modified Eagle Medium (DMEM, Gibco), 10% fetal bovine serum (FBS, Life Technologies) and 1% penicillin-streptomycin (Gibco). The mixtures were

heated to 50°C and magnetically stirred for 2-4 hours. Complete mixtures were then vortexed for 3 minutes and centrifuged at 167 RCF for 3 minutes to eliminate bubbles. Hydrogels were subsequently refrigerated at 4°C prior to experimentation and left at room temperature (22.5 °C) for 6 hours prior to printing.

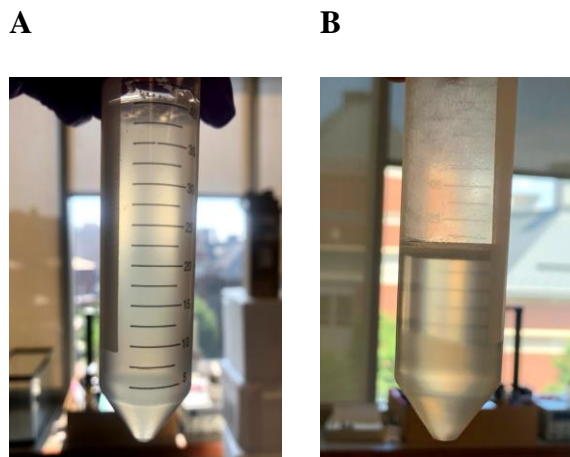
#### 4.2.1.3. *Alginate hydrogel printing*

Extrusion of alginate hydrogel was conducted at 22.5 °C with both a 25 gauge (25G) (inner diameter = 410 µm) cylindrical nozzle (EFD Nordson). Gelatin slurries are transferred to petri dishes or culture plate wells for FRESH printing. When transferred into culture plate wells, the plate was centrifuged at 739 RCF for 2 minutes at 4 °C to eliminate cavities with slurry sample and to compact slurry to produce a consistent density of gelatin microparticles for hydrogels to be embedded in. Prior to printing, excess liquid at the surface of the slurry in petri dishes or cell culture plate wells was wiped off with paper wipes (Kimberly Clark). All alginate hydrogels were deposited into the gelatin slurry. The feed rate was set to 4 mm/s. Extrusion pressure was set at range from 25 to 41 kPa.

#### 4.2.2. *Results and Discussion*

##### 4.2.2.1. *Optimization of gelatin slurry synthesis*

Initial replication attempts of Hinton et al.'s synthesis procedure of gelatin slurry resulted in inconsistent separation of excess supernatant and compacted gelatin slurry after the centrifugation process (**Fig. 19**) [73].



**Figure 19:** Based on original synthesis methods of FRESH gelatin slurry, A) no clear separation of supernatant and compacted gelatin particles can be observed or B) distinct separation of supernatant and gelatin microparticles can be observed.

Comparatively, gelatin slurry batches with no excess supernatant separated required more yield stress to initiate movement of thin and long objects than batches produced that generated supernatant top layers after centrifugation. The initial step taken to resolve synthesis consistency issue was to store gel-phase gelatin and  $\text{CaCl}_2$  solution at  $-20\text{ }^\circ\text{C}$  for 2 hours to initiate ice crystal formation in the  $\text{CaCl}_2$  solution. This is to maintain as low of a temperature as possible for prevention of gelatin microparticle fusion as heat is added through friction between the blender blades and blender contents during the mixing. Implementing this process before blending and centrifuging still resulted in poor repeatability of supernatant separation. Lewicki et al.'s FRESH gelation slurry synthesis method where filling to the brim with  $\text{CaCl}_2$  solution of a blender jar was recommended prior to blending to reduce overheating of gelatin slurry during blending [77]. This step was added to the overall gelatin slurry synthesis procedure prior to setting the blending components at  $-20\text{ }^\circ\text{C}$ . Additional steps to reduce overheating and fusion of gelatin microparticles include limiting durations of blending to limit prolonged periods of heat addition from friction. Instead of blending for a continuous 120 seconds, blending proceeded at 30 second intervals with 30 seconds of rest four times. The addition of these steps resulted in consistent separation of excess supernatant and compact gelatin slurry that allows for cylindrical nozzle movement and shape



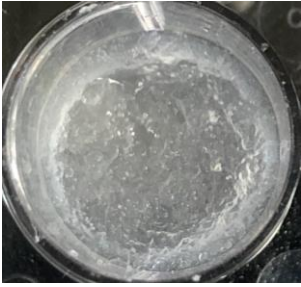







recovery during the printing process.

#### 4.2.2.2. *Optimization of feed rate*

Recovery of gelatin slurry material after being displaced by a moving nozzle is not an instantaneous process and requires time. Additionally, too high of stress exerted on the slurry material can result in larger displacement areas that will not revert back into the undisturbed shape. Therefore, determining an optimized feed rate of a nozzle to avoid excessive disruption of the support bath is critical to take advantage of its Bingham plastic behavior. To determine an optimized feed rate, a range of feed rates were tested from 2 to 10 mm/s, at 2, 3, 4, 5, and 10 mm/s. At a feed rate of 4 mm/s, shape recovery behavior was retained well as gelatin slurry material retained their original position within the slurry after becoming displaced one minute post nozzle movement within the slurry (**Table 14**). This feed rate was chosen for subsequent FRESH bioprinting of alginate hydrogels.

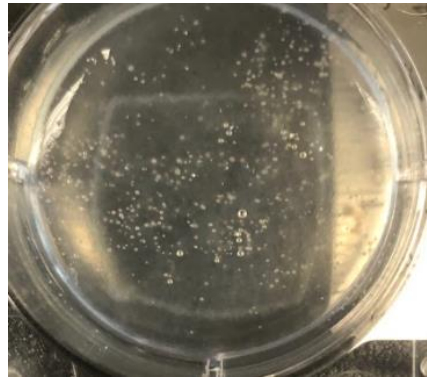


**Table 14.** Quantitative shape recovery behavior observation of FRESH gelatin slurry 1 minute after movement of nozzle within the compacted slurry at 4 °C. Feed rate was adjusted from 2 to 10 mm/s and a 25G cylindrical steel nozzle was actuated within the slurry.

Feed rate (mm/s)	2	3	4	5	10
					
					

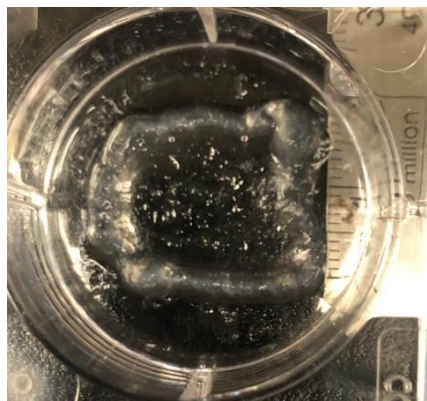
#### 4.2.2.3. *Effect of FRESH printing process on shape fidelity*

The printing of alginate hydrogel in gelatin slurry initially retained shape fidelity of bioprinted constructs directly after printing. Due to diffusion of phenol red coloring, alginate hydrogel quickly changed color from pink-red to a white color. When placed in incubation at 37 °C to melt the gelatin slurry and absorption of water into the alginate hydrogel resulted in swelling, leading to expanded filament diameter shapes (**Fig. 20**).



**Figure 20:** 2% (w/v) alginate hydrogel construct printed into gelatin slurry with a  $\text{CaCl}_2$  concentration of 11 mM

When exposed melted gelatin slurry was aspirated, resultant slurry shape fidelity increased several times greater than the nozzle diameter of which they were printed out of. In addition to swelling behavior, gelatin also attaches to the alginate strands, creating undesired polymeric deposits (**Fig . 20**).



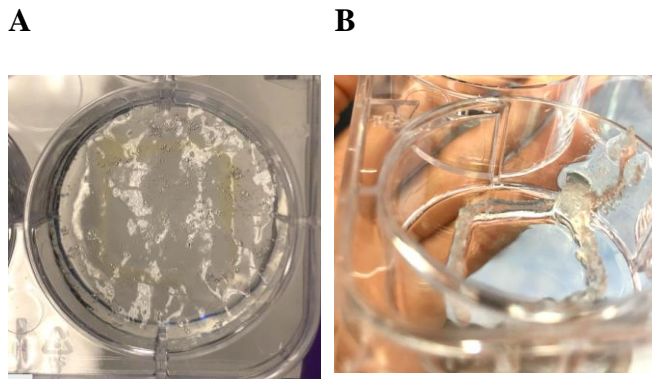
**Figure 21:** 2% (w/v) alginate hydrogel construct after aspiration of melted gelatin slurry with a  $\text{CaCl}_2$  concentration of 11 mM

A method to reduce gelatin fusing to alginate hydrogel can be to increase the ionic crosslinker concentration in the gelatin slurry. When  $\text{CaCl}_2$  concentration was increased to 100 mM from 11 mM commonly used in FRESH bioprinting literature, alginate hydrogel deposited crosslinked rapidly when extruded out of a nozzle. Due to the drastically shorter crosslinking time, hydrogels were dragged through the slurry, leading to no formation of a desired construct (**Fig. 21**).



**Figure 22:** 2% (w/v) alginate hydrogel printed into gelatin slurry with  $\text{CaCl}_2$  concentration of 100 mM. The pink-colored components indicate alginate hydrogel deposits from the extrusion process.

As 100 mM concentration resulted in premature crosslinking, gelatin slurry with  $\text{CaCl}_2$  concentration of 20 mM was synthesized and used for 2% (w/v) alginate printing. Similar to printing into slurry with 11 mM  $\text{CaCl}_2$ , resultant constructs after incubation and removal of melted slurry still contained large deposits of gelatin and swelled filament (**Fig. 23**)



**Figure 23:** 2% alginate deposited in gelatin slurry with 20 mM CaCl<sub>2</sub> **A)** one minute post-printing and **B)** directly after removal of melted gelatin slurry.

Deposition of low viscosity hydrogel that can be crosslinked ionically into FRESH gelatin slurry holds promise as a method to improve the robustness of biomaterials that can be used for EBB. However, optimization of the slurry is critical because the hydrogels used, specifically alginate-based ones, are susceptible to high levels of fluid absorption and structural disintegration of constructs that can make the process infeasible to produce constructs with suitable shape fidelity. A subset of data focused on support-bath based EBB can be created and used for ML model training to examine support bath effects on filament diameter effects.

### **4.3 Effect of various non-primary cells**

Beyond N2A neuroblastoma cells, two other non-primary cell types were printed to observe potential discrepancies in shear-induced cell death mechanisms directly post-extrusion.

#### *4.3.1. Materials and methods*

Alginate-gelatin composite hydrogel was synthesized according to section 3.2.1 with DMEM/F-12 basal cell culture (Gibco) used as the solvent for SH-SY5Y printing and DMEM for HepG2 printing. 6/4 Alg/Gel was used in accordance to previous literature [78] with SH-SY5Y cells while 3/4 and 3/7 Alg/Gel was used with HepG2 as a comparison to N2A cellular printing at the same polymer precursor concentrations. SH-SY5Y neuroblastoma cells were cultured in accordance to procedures in Section

3.2.2 with DMEM/F-12 medium while HepG2 cells were cultured with DMEM. Bioink printing and cell viability quantification were conducted in accordance to procedures in section 3.2.3 and 3.2.4.

#### 4.3.2. *Results and discussion*

Directly after printing at an extrusion pressure of 78 kPa, cell viability of SH-SY5Y cells resulted in an average of  $76.0 \pm 4.18\%$  ( $n = 4$ ) cell viability directly post-printing and crosslinking. For HepG2 cells, printing with 3/4 Alg/Gel bioinks at a pressure of  $48 \text{ kPa} \pm 3.2 \text{ kPa}$  ( $n = 4$ ) resulted in an average cell viability value of  $79.2 \pm 7.14\%$  ( $n = 4$ ) while printing with 3/7 Alg/Gel bioinks at a pressure of  $87.4 \pm 1.2 \text{ kPa}$  ( $n = 5$ ) resulted in an average cell viability value of  $52.5 \pm 4.30\%$  ( $n = 5$ ).

Cell viability values of SH-SY5Y cells printed at a higher pressure with a more viscous bioink (6/4 Alg/Gel) than printing of N2A cells with 3/4 Alg/Gel were expectedly lower. When compared to average cell viability value of N2A cells printed with stiffer 3/7 Alg/Gel, average SH-SY5Y cell viability value was higher by 11.8%. A decrease of 6% in viability was seen with HepG2 cells printed with 3/4 Alg/Gel as compared to N2A cells printed with the same polymer precursor concentration. Similarly, a larger cell viability decrease of 11.7% was seen in HepG2-laden 3/7 Alg/Gel bioink compared to N2A-laden bioink. A major factor for notable decreases in cell viability in bioink with identical polymer precursor concentrations is the higher average pressure required for extrusion. HepG2 cells also aggregate into spheroid-like geometries, leading to more cells near the nozzle-bioink interface during extrusion to experience high shear stress and subsequent cell death due to shear stress. For future statistical models, a correction factor to account for cellular aggregation in HepG2 cells when used can allow for more accurate representation of physical phenomenon within a dataset for machine learning model training. Overall, cell viability trends follow suit amongst different non-primary cells during the extrusion process of retaining similar cell viability values at constant pressures.

## 5. Conclusion and Future Work

### 5.1 Conclusion

Throughout the field of EBB, research groups utilize a wide array of polymer precursor materials, crosslinking strategies, solvents, cells, and biological additives. In addition, a variety of printing settings and strategies are used in accordance to materials used and desired printability outcomes. This research investigated the ability of applied ML models to create accurate predictive models to forecast cellular and printability outcomes as well as experimental input to produce desired cellular outcomes. The effect of dataset training size and specificity was also examined through training, testing, and experimentally validating a set of ML models trained on a large generalized dataset and a set trained on a smaller dataset utilizing experimental parameters that can be replicated in our laboratory (this dataset was termed “intrastudy dataset”). The models trained through experimental parameters and outcomes extracted from literature spanning the past 13 years in EBB were compared in performance metrics as classification and regression models. Amongst both classification and regression models, random forest models resulted in higher performance metrics, seen through cross-validation testing. Compared to previous ML-based EBB literature, performance metrics of coefficient of determination for regression models and accuracy for classification models were lower due to having utilizing different experimental conditions in the training data, although still suitable (0.43 for  $R^2$  and 71.7% for accuracy scores).

Experimental validation of the models at different polymer precursor material concentrations demonstrated wide variability in predictive power. For regression-based cell viability predictions, random forest models trained on the larger generalized dataset were found to produce low variation in predicted cell viability values while linear regression resulted in no variation in cell viability prediction. Regression models trained with the intrastudy dataset resulted in unrealistic cell viability trends with random forest regression predicting increasing cell viability with increased polymer precursor

concentration while linear regression modeling predicting identical and negative cell viability values at different concentrations. All classification models resulted in the same viability predictions of being acceptable with experimental validation input values, indicating a defaulting mechanism when input conditions not present in datasets. Furthermore, low variation and no variation in cell viability regression and classification prediction models indicate lack of strong correlative effects amongst all input parameters used in training datasets to cell viability outcomes as shown through feature importance scoring. Overall, the cell viability range predicted for material concentration ranges tested based on random forest regression models are within experimental cell viability values of different non-primary mammalian cells printed at the same material concentrations.

Printability assessment via resultant filament diameter values indicated that random forest regression demonstrated predictive abilities that minimized error predictions while also demonstrating wider range of predicted results compared to linear regression models. Similar to cell viability experimental validation results, the filament diameter linear regression model trained on the intrastudy dataset resulted in predicted values that are vastly smaller than actual results. This indicated the filament diameter values of the specific literature used for intrastudy dataset creation demonstrated opposite physical trends compared to results of the majority of work used in the larger generalized dataset.

Extrusion pressure condition recommendations demonstrated the need for more data or variable consideration. Random forest regression models demonstrated closer prediction accuracy to actual pressures used at lower polymer precursor concentrations while demonstrating prediction defaulting at higher concentrations. Linear regression models provided more accurate prediction values at higher polymer precursor concentrations as opposed to lower ones. Classification models besides the random forest classifier resulted in acceptable extrusion pressure prediction for all polymer precursor concentrations used, indicating a lack of prediction robustness. At gelatin concentrations above 5%

(w/v), all extrusion pressure predictions were deemed unacceptable, similar to the random forest regression model defaulting to a specific pressure value.

Compared to physical models of filament diameter, machine learning models described in this work takes into consideration post-processing parameters, particularly crosslinking variables, printbed and substrate temperature, and duration of culturing in cell culture media. Whereas physical filament diameter models derive diameter values during the extrusion process, ML models can predict values at extended times post-printing (> 1 hour), providing higher robustness in usage. The robustness of ML models also applies for cell viability estimations. Thus far, major cell viability correlations developed in EBB have focused either on extrusion process-based variables (extrusion pressure, shear stress, residence time, and viscosity) [37,48,51,79] or one aspect of post-printing processing, such as crosslinking duration [80]. ML models developed here encompasses aforementioned printing and post-printing parameters to provide cell viability value predictions of which a singular correlation cannot.

## 5.2 Future Work

As with general cases of improving ML model robustness, increasing training dataset size with diverse input combinations is a straightforward method to greater represent general EBB experimental parameters and results. A method to expedite the manual scrapping process used for dataset creation in this work would be to implement natural language processing tools to scrape quantitative input and output variable values, such as extrusion pressure and cell viability respectively, as well as deploying image processing platforms that can gather quantitative values of graphs and filament diameters of construct images. The development of first principle-based data generation, such as the usage of non-Newtonian index values of specific polymer precursors for extrusion pressure can accelerate dataset expansion. It should be noted that the inclusion of *in silico* generated data may alter the predictive performance of applied ML algorithms, where the base first-principle equation(s) used will heavily influence pattern recognition. Image-processing based ML and deep learning models such as



convoluted neural networks can be used to provide cell proliferation and ECM formation image predictions to supplement quantitative biological outcomes, as well as surface morphology of printed constructs to predict under and over-deposition of materials through surface roughness imaging and filament diameter variation along deposited filament.

Exploring the effect of the amount of missing values for individual variables can clarify the size-based impact of value imputation on outcome prediction. Notably for cell viability model creation, there are several input parameters, including extrusion pressure, syringe temperature, crosslinking duration, and printed temperature that contained large portions (> 15%) of null instances as compared to other variables with null instances within the preprocessed dataset. In the case of this study, tuning the hyperparameter of number of neighbors in k-nearest neighbor imputation may also reveal limitations of using the chosen imputation method. A broader, long-term approach to reducing the frequency of null instances associated with experimental parameters during dataset expansion is to set a reporting guideline for the EBB field to report quantifiable and categorical experimental parameters [14].

Another research article using using alginate-gelatin composite hydrogels can be used as the intrastudy dataset source as a comparison to the selected literature used for this study. In particular, filament shrinkage and reduced filament swelling observed over time post-printing from the intrastudy dataset source counters our empirical observations as well as other alginate-gelatin-based EBB literature [5,81,82], which may provide inaccurate representation of the predictive effectiveness of ML models built through smaller, specified datasets. This potential endeavor can provide greater understanding of the effect of using a smaller, specified dataset.

## 6. Contributions

The work described in this thesis has been published as two peer-reviewed journal articles. Information in the literature review section is covered in a literature review published in *Bioprinting* focusing on optimization and standardization of EBB procedures from material and cellular selection to post-printing analysis [14]. ML-based findings and experimental validation results and discussions are described in a research article published in *Micromachines* [83]. The research described in this thesis is novel due to four major aspects:

1. it provides machine-readable datasets of parameters reported in published studies generated by different research groups as compared to existing approaches which only utilize data generated in-house by one research group;
2. it utilizes both classification and regression-based modeling as compared to existing approaches which focus on regression-based modeling only;
3. it examines cellular outcomes and input optimization of extrusion pressure as compared to existing approaches which mainly focus only on printability metrics;
4. it provides experimental validation of the trained machine learning models as compared to existing approaches that only provide only computational testing outcomes.

## 7. Vita

### Shuyu Tian

716-848-9559 | [tians2@vcu.edu](mailto:tians2@vcu.edu) | LinkedIn/shuyu-tian

#### **EDUCATION:**

**Virginia Commonwealth University**, Richmond, VA  
*M.S. in Chemical and Life Science Engineering*

**GPA: 3.91**  
December 2021

**University of Tulsa**, Tulsa, OK  
*Bachelor of Science in Mechanical Engineering and Economics*

**GPA: 3.72**  
May 2019

#### **RESEARCH EXPERIENCE:**

**Graduate Research– Lewinski Lab (Dr. Nastassja Lewinski)** Richmond, VA 2019-Present

- Developed Python-based machine learning models to predict 3D tissue engineering outcomes, resulting in a first-authored publication
- Conducted literature review writing of the 3D bioprinting field, resulting in a first-authored publication
- Synthesized biomaterials to enhance 3D printing of human cells and tissues for disease modeling
- Developing polymer material substrates for coral cell culturing and engineering and 3D tissue formation
- Evaluated toxicity of nanomaterials, pharmaceuticals, and toxins in human and non-human model systems
- Managed lab equipment, supplies, and experiments of graduate and undergraduate researchers

**Undergraduate Research – Lim Lab (Dr. May Lim, Dr. Kang Liang)** Sydney, Australia 2019

- Created protocol to synthesize new composite metal organic framework nanomaterial based on induction heating
- Optimized scale-up production of magnetic metal organic frameworks by testing heat, agitation, solution concentrations
- Mentored other undergraduate researchers by reviewing protocols and data to ensure proper experimentation
- Presented results at the International Symposium of Clusters and Nanomaterials (November 2019)

**Undergraduate Research– Keller Lab (Dr. Michael Keller, Dr. Joshua Schultz)** Tulsa, OK 2018

- Designed 3D-printed molds for silicone rubber tensile specimen creation using SolidWorks
- Designed 3D-printed wax filament configurations for ferrofluid channels in silicone rubber specimens
- Investigated a new smart-material with active stiffness control via magnetism

**Undergraduate Research– Johannes Lab (Dr. Tyler Johannes)** Tulsa, OK 2018

- Investigated algae pigment variation under different light wavelengths using UV-Vis absorption
- Performed and analyzed phenol sulfuric acid carbohydrate assays, Nile red staining lipid assays, and Bradford protein assays for algae nutrient composition
- Conducted literature reviews and proposed new techniques to analyze algae pigment variation

**SAE Aero Design Team Project Leader and President**, Tulsa, OK 2017-2018

- Lead a design team in the creation of a radio-controlled aircraft that can be reassembled
- Conducted airfoil testing and optimization for radio-controlled aircraft using XFLR5 analysis software
- Utilized laser cutters and SolidWorks modeling to manufacture structurally sound components

## **SKILLS:**

**CAD & Modeling:** SolidWorks, ANSYS, CATIA, AutoCAD, Ultimaker Cura

**Programming:** Python, R, MATLAB, VBA, Mathematica

**Data Analysis and Visualization:** OriginPro, ImageJ

**Material Characterization:** Thermogravimetric Analysis, Transmission-Electron Microscopy, Brunauer Emmett Teller Gas Adsorption, X-ray Powder Diffraction, Rheological Characterization, Soft Material Compression Testing, Dynamic Light Scattering

**Biological:** Aseptic Technique, Mammalian Cell Culture, UV-Vis Spectroscopy, Cytotoxicity Assay Techniques (MTT, LDH, Live/Dead), Microscopy (Confocal, Inverted)

## **PRESENTATIONS & ABSTRACTS:**

**Tian S, Stevens R, McInnes BT, Lewinski NA.** “Machine Assisted Experimentation for Extrusion-Based Bioprinting”, Accepted to **American Institute of Chemical Engineers Annual Meeting**, Boston, MA, November 8, 2021

**Tian S, Stevens R, McInnes BT, Lewinski NA.** “Machine Assisted Experimentation for Extrusion-based Bioprinting” Accepted to **24<sup>th</sup> Annual Graduate Research Symposium**, Richmond, VA, April 20, 2021

**Tian S, Tran R, Xiang J, Kang L, Lim M.** “Exploring the Effect of Magnetic Nanoparticles on Multicomponent Metal Organic Framework Synthesis” Accepted to **International Symposium of Clusters and Nanomaterials**, Richmond, VA November 8, 2019

## **VOLUNTEERING EXPERIENCE:**

**Engineers Without Borders-USA** 2015-Present

**South Central Region Steering Committee, President, Treasurer** 2019-Present

- Moderated monthly networking meetings among 24 chapters across four states
- Prepared annual budget and approves all expenditure for the steering committee
- Develop guidelines and strategies to streamline financial and technical resources for chapters within region

**Student Chapter President, Treasurer, Project Leader** 2015-2018

- Lead implementation of a solar-powered cell phone charger for homeless shelter, resulting in \$200/year savings
- Managed grant applications and fundraisers totaling \$7,000, as well as resource allocation for project funding
- Developed projects through technical design, mentorship outreach, and 3<sup>rd</sup> party sourcing

## **HONORS, AWARDS, SCHOLARSHIPS:**

**NSF Graduate Research Fellowship Program Honorable Mention** Spring 2020

**Sidney Born Medal for Academic Excellence** Spring 2018

**Chevron Scholar** 2015-2018

**Vision Scholar** 2014-2019

## **PROFESSIONAL MEMBERSHIP:**

**American Institute of Chemical Engineers** – Member 2019-Present

**American Society of Mechanical Engineers** - Member 2017-Present,

**American Chemical Society** – Member 2021-Present

**Tau Beta Pi Engineering Honor Society** – Member 2017-Present

**Engineers Without Borders-USA** – Member 2015-Present

## **TEACHING EXPERIENCE:**

### **Unit Operations Graduate Teaching Assistant**

Fall 2021

- Oversaw student performance on laboratory demonstrations
- Graded laboratory notebooks and reports

### **Energy and Material Balances Graduate Teaching Assistant**

Spring 2020-2021

- Instructed undergraduate students on fundamental thermodynamic concepts of chemical engineering
- Provided lectures on using Jupyter Notebook software for Python usage
- Designed and graded assignments, quizzes, and projects

### **Transport Phenomena Graduate Teaching Assistant**

Fall 2019

- Responsible for grading, tutoring, and teaching selected momentum, mass, and heat transfer concepts

### **Professional Aspects of Engineering Undergraduate Teaching Assistant**

Fall 2018

- Responsible for curriculum design and teaching, with specific emphasis on ethics and risk analysis based on industrial experiences

### **Department Tutor, Calculus I-III**

2017-2018

- Provide weekly tutoring sessions to engineering students on multivariate calculus

## **OUTREACH:**

- **EWB-USA South Central Regional Conference** – Texas A&M University, College Station, TX Fall 2021
- **NanoDay** – Science Museum of Virginia, Richmond VA Fall 2019
- **Tulsa Maker Faire** – Oklahoma State Fairgrounds, Tulsa OK Fall 2018
- **Girl Scouts Brownie Day** – University of Tulsa, Tulsa OK 2015-2018
- **NSBE Summer Engineering Education for Kids Program** – New Orleans, LA Summer 2015
- **Tulsa Stem Regional Alliance Outreach** – Memorial Middle School, Tulsa OK Spring 2015

## **PUBLICATIONS:**

**Tian S**, Stevens R, McInnes BT, Lewinski NA. (2021). Machine Assisted Experimentation of Extrusion-Based Bioprinting Systems. Micromachines **12**:780..

**Tian S**, Zhao H, Lewinski NA. (2021). Key parameters and applications of extrusion-based bioprinting, Bioprinting. e00156..

## 8. References

- [1] S. V. Murphy, A. Atala, 3D bioprinting of tissues and organs, *Nat. Biotechnol.* 32 (2014) 773–785. <https://doi.org/10.1038/nbt.2958>.
- [2] J. Groll, J.A. Burdick, D.W. Cho, B. Derby, M. Gelinsky, S.C. Heilshorn, T. Jüngst, J. Malda, V.A. Mironov, K. Nakayama, A. Ovsianikov, W. Sun, S. Takeuchi, J.J. Yoo, T.B.F. Woodfield, A definition of bioinks and their distinction from biomaterial inks, *Biofabrication.* 11 (2019) 013001. <https://doi.org/10.1088/1758-5090/aaec52>.
- [3] K. Christensen, C. Xu, W. Chai, Z. Zhang, J. Fu, Y. Huang, Freeform inkjet printing of cellular structures with bifurcations, *Biotechnol. Bioeng.* 112 (2015) 1047–1055. <https://doi.org/10.1002/bit.25501>.
- [4] G. Gillispie, P. Prim, J. Copus, J. Fisher, A.G. Mikos, J.J. Yoo, A. Atala, S.J. Lee, Assessment methodologies for extrusion-based bioink printability, *Biofabrication.* 12 (2020) 22003. <https://doi.org/10.1088/1758-5090/ab6f0d>.
- [5] Y. He, F. Yang, H. Zhao, Q. Gao, B. Xia, J. Fu, Research on the printability of hydrogels in 3D bioprinting, *Sci. Rep.* 6 (2016) 1–13. <https://doi.org/10.1038/srep29977>.
- [6] S. Kyle, Z.M. Jessop, A. Al-Sabah, I.S. Whitaker, “Printability” of Candidate Biomaterials for Extrusion Based 3D Printing: State-of-the-Art, (2017). <https://doi.org/10.1002/adhm.201700264>.
- [7] S. V. Murphy, A. Skardal, A. Atala, Evaluation of hydrogels for bio-printing applications, *J. Biomed. Mater. Res. - Part A.* 101 A (2013) 272–284. <https://doi.org/10.1002/jbm.a.34326>.
- [8] L. Ouyang, R. Yao, Y. Zhao, W. Sun, Effect of bioink properties on printability and cell viability for 3D bioplotting of embryonic stem cells, *Biofabrication.* 8 (2016). <https://doi.org/10.1088/1758-5090/8/3/035020>.
- [9] A. Ribeiro, M.M. Blokzijl, R. Levato, C.W. Visser, M. Castilho, W.E. Hennink, T. Vermonden, J. Malda, Assessing bioink shape fidelity to aid material development in 3D bioprinting, *Biofabrication.*

10 (2018). <https://doi.org/10.1088/1758-5090/aa90e2>.

[10] Z. Zhang, Y. Jin, J. Yin, C. Xu, R. Xiong, K. Christensen, B.R. Ringeisen, D.B. Chrisey, Y. Huang, Evaluation of bioink printability for bioprinting applications, *Appl. Phys. Rev.* 5 (2018). <https://doi.org/10.1063/1.5053979>.

[11] N. Soltan, L. Ning, F. Mohabatpour, P. Papagerakis, X. Chen, Printability and Cell Viability in Bioprinting Alginate Dialdehyde-Gelatin Scaffolds, *ACS Biomater. Sci. Eng.* 5 (2019) 2976–2987. <https://doi.org/10.1021/acsbiomaterials.9b00167>.

[12] B. Webb, B.J. Doyle, Parameter optimization for 3D bioprinting of hydrogels, *Bioprinting.* 8 (2017) 8–12. <https://doi.org/10.1016/j.bprint.2017.09.001>.

[13] M. Di Giuseppe, N. Law, B. Webb, R. A. Macrae, L.J. Liew, T.B. Sercombe, R.J. Dilley, B.J. Doyle, Mechanical behaviour of alginate-gelatin hydrogels for 3D bioprinting, *J. Mech. Behav. Biomed. Mater.* 79 (2018) 150–157. <https://doi.org/10.1016/j.jmbbm.2017.12.018>.

[14] S. Tian, H. Zhao, N. Lewinski, Key parameters and applications of extrusion-based bioprinting, *Bioprinting.* (2021) e00156. <https://doi.org/10.1016/j.bprint.2021.e00156>.

[15] L.E. Bertassoni, J.C. Cardoso, V. Manoharan, A.L. Cristino, N.S. Bhise, W.A. Araujo, P. Zorlutuna, N.E. Vrana, A.M. Ghaemmaghami, M.R. Dokmeci, A. Khademhosseini, Direct-write bioprinting of cell-laden methacrylated gelatin hydrogels, *Biofabrication.* 6 (2014). <https://doi.org/10.1088/1758-5082/6/2/024105>.

[16] Y. Gu, L. Zhang, X. Du, Z. Fan, L. Wang, W. Sun, Y. Cheng, Y. Zhu, C. Chen, Reversible physical crosslinking strategy with optimal temperature for 3D bioprinting of human chondrocyte-laden gelatin methacryloyl bioink, *J. Biomater. Appl.* 33 (2018) 609–618. <https://doi.org/10.1177/0885328218805864>.

[17] A. Mora-Boza, M.K. Włodarczyk-Biegun, A. Del Campo, B. Vázquez-Lasa, J.S. Román, Glycerylphosphate as an ionic crosslinker for 3D printing of multi-layered scaffolds with improved shape

- fidelity and biological features, *Biomater. Sci.* 8 (2020) 506–516. <https://doi.org/10.1039/c9bm01271k>.
- [18] J. Yin, M. Yan, Y. Wang, J. Fu, H. Suo, 3D Bioprinting of Low-Concentration Cell-Laden Gelatin Methacrylate (GelMA) Bioinks with a Two-Step Cross-linking Strategy, *ACS Appl. Mater. Interfaces.* 10 (2018) 6849–6857. <https://doi.org/10.1021/acsami.7b16059>.
- [19] M.T. Poldervaart, B. Goversen, M. De Ruijter, A. Abbadessa, F.P.W. Melchels, F.C. Öner, W.J.A. Dhert, T. Vermonden, J. Alblas, 3D bioprinting of methacrylated hyaluronic acid (MeHA) hydrogel with intrinsic osteogenicity, *PLoS One.* 12 (2017) 1–15. <https://doi.org/10.1371/journal.pone.0177628>.
- [20] P. Zhuang, W.L. Ng, J. An, C.K. Chua, L.P. Tan, Layer-by-layer ultraviolet assisted extrusion-based (UAE) bioprinting of hydrogel constructs with high aspect ratio for soft tissue engineering applications, *PLoS One.* 14 (2019) 1–21. <https://doi.org/10.1371/journal.pone.0216776>.
- [21] Y. Jin, W. Chai, Y. Huang, Printability study of hydrogel solution extrusion in nanoclay yield-stress bath during printing-then-gelation biofabrication, *Mater. Sci. Eng. C.* 80 (2017) 313–325. <https://doi.org/10.1016/j.msec.2017.05.144>.
- [22] Y. Li, G. Huang, X. Zhang, B. Li, Y. Chen, T. Lu, T.J. Lu, F. Xu, Magnetic hydrogels and their potential biomedical applications, *Adv. Funct. Mater.* 23 (2013) 660–672. <https://doi.org/10.1002/adfm.201201708>.
- [23] N. Chen, K. Zhu, Y.S. Zhang, S. Yan, T. Pan, M. Abudupataer, G. Yu, M.F. Alam, L. Wang, X. Sun, Y. Yu, C. Wang, W. Zhang, Hydrogel Bioink with Multilayered Interfaces Improves Dispersibility of Encapsulated Cells in Extrusion Bioprinting, *ACS Appl. Mater. Interfaces.* 11 (2019) 30585–30595. <https://doi.org/10.1021/acsami.9b09782>.
- [24] T. Gao, G.J. Gillispie, J.S. Copus, A.P.R. Kumar, Y.J. Seol, A. Atala, J.J. Yoo, S.J. Lee, Optimization of gelatin-alginate composite bioink printability using rheological parameters: A systematic approach, *Biofabrication.* 10 (2018). <https://doi.org/10.1088/1758-5090/aacdc7>.



- [25] A. Tirella, A. Orsini, G. Vozzi, A. Ahluwalia, A phase diagram for microfabrication of geometrically controlled hydrogel scaffolds, *Biofabrication*. 1 (2009). <https://doi.org/10.1088/1758-5082/1/4/045002>.
- [26] J. Jia, D.J. Richards, S. Pollard, Y. Tan, J. Rodriguez, R.P. Visconti, T.C. Trusk, M.J. Yost, H. Yao, R.R. Markwald, Y. Mei, Engineering alginate as bioink for bioprinting, *Acta Biomater*. 10 (2014) 4323–4331. <https://doi.org/10.1016/j.actbio.2014.06.034>.
- [27] T. Jiang, J.G. Munguia-Lopez, S. Flores-Torres, J. Kort-Mascort, J.M. Kinsella, Extrusion bioprinting of soft materials: An emerging technique for biological model fabrication, *Appl. Phys. Rev*. 6 (2019). <https://doi.org/10.1063/1.5059393>.
- [28] J.H.Y. Chung, S. Naficy, Z. Yue, R. Kapsa, A. Quigley, S.E. Moulton, G.G. Wallace, Bio-ink properties and printability for extrusion printing living cells, *Biomater. Sci*. 1 (2013) 763–773. <https://doi.org/10.1039/c3bm00012e>.
- [29] N. Paxton, W. Smolan, T. Böck, F. Melchels, J. Groll, T. Jungst, Proposal to assess printability of bioinks for extrusion-based bioprinting and evaluation of rheological properties governing bioprintability, *Biofabrication*. 9 (2017). <https://doi.org/10.1088/1758-5090/aa8dd8>.
- [30] J.M. Townsend, E.C. Beck, S.H. Gehrke, C.J. Berkland, M.S. Detamore, Flow behavior prior to crosslinking: The need for precursor rheology for placement of hydrogels in medical applications and for 3D bioprinting, *Prog. Polym. Sci*. 91 (2019) 126–140. <https://doi.org/10.1016/j.progpolymsci.2019.01.003>.
- [31] T. Billiet, E. Gevaert, T. De Schryver, M. Cornelissen, P. Dubruel, The 3D printing of gelatin methacrylamide cell-laden tissue-engineered constructs with high cell viability, *Biomaterials*. 35 (2014) 49–62. <https://doi.org/10.1016/j.biomaterials.2013.09.078>.
- [32] F. Zhu, L. Cheng, J. Yin, Z.L. Wu, J. Qian, J. Fu, Q. Zheng, 3D Printing of Ultratough Polyion Complex Hydrogels, *ACS Appl. Mater. Interfaces*. 8 (2016) 31304–31310.

<https://doi.org/10.1021/ACSAMI.6B09881>.

[33] S.A. Wilson, L.M. Cross, C.W. Peak, A.K. Gaharwar, Shear-Thinning and Thermo-Reversible Nanoengineered Inks for 3D Bioprinting, *ACS Appl. Mater. Interfaces*. 9 (2017) 43449–43458.

<https://doi.org/10.1021/acsami.7b13602>.

[34] K.D. Roehm, S. V Madihally, Bioprinted chitosan-gelatin thermosensitive hydrogels using an inexpensive 3D printer, *Biofabrication*. 10 (2017) 015002. [https://doi.org/10.1088/1758-](https://doi.org/10.1088/1758-5090/AA96DD)

[5090/AA96DD](https://doi.org/10.1088/1758-5090/AA96DD).

[35] Q. Wang, J. Sun, Q. Yao, C. Ji, J. Liu, Q. Zhu, 3D printing with cellulose materials, *Cellulose*. 25 (2018) 4275–4301. <https://doi.org/10.1007/s10570-018-1888-y>.

[36] M. Kim, G.H. Kim, 3D multi-layered fibrous cellulose structure using an electrohydrodynamic process for tissue engineering, *J. Colloid Interface Sci*. 457 (2015) 180–187.

<https://doi.org/10.1016/j.jcis.2015.07.007>.

[37] Y. Zhao, Y. Li, S. Mao, W. Sun, R. Yao, The influence of printing parameters on cell survival rate and printability in microextrusion-based 3D cell printing technology, *Biofabrication*. 7 (2015).

<https://doi.org/10.1088/1758-5090/7/4/045002>.

[38] W. Schuurman, P.A. Levett, M.W. Pot, P.R. van Weeren, W.J.A. Dhert, D.W. Hutmacher, F.P.W. Melchels, T.J. Klein, J. Malda, Gelatin-methacrylamide hydrogels as potential biomaterials for fabrication of tissue-engineered cartilage constructs, *Macromol. Biosci*. 13 (2013) 551–561.

<https://doi.org/10.1002/mabi.201200471>.

[39] D. Chimene, K.K. Lennox, R.R. Kaunas, A.K. Gaharwar, Advanced Bioinks for 3D Printing: A Materials Science Perspective, *Ann. Biomed. Eng*. 44 (2016) 2090–2102.

<https://doi.org/10.1007/s10439-016-1638-y>.

[40] A.K. Miri, I. Mirzaee, S. Hassan, S. Mesbah Oskui, D. Nieto, A. Khademhosseini, Y.S. Zhang, Effective bioprinting resolution in tissue model fabrication, *Lab Chip*. 19 (2019) 2019–2037.

<https://doi.org/10.1039/c8lc01037d>.

[41] M.H. Kim, Y.W. Lee, W.K. Jung, J. Oh, S.Y. Nam, Enhanced rheological behaviors of alginate hydrogels with carrageenan for extrusion-based bioprinting, *J. Mech. Behav. Biomed. Mater.* 98 (2019) 187–194. <https://doi.org/10.1016/j.jmbbm.2019.06.014>.

[42] A.C. Daly, G.M. Cunniffe, B.N. Sathy, O. Jeon, E. Alsberg, D.J. Kelly, 3D Bioprinting of Developmentally Inspired Templates for Whole Bone Organ Engineering, *Adv. Healthc. Mater.* 5 (2016) 2353–2362. <https://doi.org/10.1002/adhm.201600182>.

[43] X. Yang, Z. Lu, H. Wu, W. Li, L. Zheng, J. Zhao, Collagen-alginate as bioink for three-dimensional (3D) cell printing based cartilage tissue engineering, *Mater. Sci. Eng. C.* 83 (2018) 195–201. <https://doi.org/10.1016/j.msec.2017.09.002>.

[44] S. Naghieh, M.D. Sarker, N.K. Sharma, Z. Barhoumi, X. Chen, Printability of 3D printed hydrogel scaffolds: Influence of hydrogel composition and printing parameters, *Appl. Sci.* 10 (2020). <https://doi.org/10.3390/app10010292>.

[45] I. Matai, G. Kaur, A. Seyedsalehi, A. McClinton, C.T. Laurencin, Progress in 3D bioprinting technology for tissue/organ regenerative engineering, *Biomaterials.* 226 (2020) 119536. <https://doi.org/10.1016/j.biomaterials.2019.119536>.

[46] S. Catros, J.C. Fricain, B. Guillotin, B. Pippenger, R. Bareille, M. Remy, E. Lebraud, B. Desbat, J. Amédée, F. Guillemot, Laser-assisted bioprinting for creating on-demand patterns of human osteoprogenitor cells and nano-hydroxyapatite, *Biofabrication.* 3 (2011). <https://doi.org/10.1088/1758-5082/3/2/025001>.

[47] T. Xu, J. Jin, C. Gregory, J.J. Hickman, T. Boland, Inkjet printing of viable mammalian cells, *Biomaterials.* 26 (2005) 93–99. <https://doi.org/10.1016/j.biomaterials.2004.04.011>.

[48] A. Blaeser, D.F. Duarte Campos, U. Puster, W. Richtering, M.M. Stevens, H. Fischer, Controlling Shear Stress in 3D Bioprinting is a Key Factor to Balance Printing Resolution and Stem

Cell Integrity, *Adv. Healthc. Mater.* 5 (2016) 326–333. <https://doi.org/10.1002/adhm.201500677>.

[49] J. Shi, B. Wu, S. Li, J. Song, B. Song, W.F. Lu, Shear stress analysis and its effects on cell viability and cell proliferation in drop-on-demand bioprinting, *Biomed. Phys. Eng. Express.* 4 (2018) 45028. <https://doi.org/10.1088/2057-1976/aac946>.

[50] P. Chansoria, L.K. Narayanan, M. Wood, C. Alvarado, A. Lin, R.A. Shirwaiker, Effects of Autoclaving, EtOH, and UV Sterilization on the Chemical, Mechanical, Printability, and Biocompatibility Characteristics of Alginate, *ACS Biomater. Sci. Eng.* 6 (2020) 5191–5201. <https://doi.org/10.1021/acsbiomaterials.0c00806>.

[51] K. Nair, M. Gandhi, S. Khalil, K.C. Yan, M. Marcolongo, K. Barbee, W. Sun, Characterization of cell viability during bioprinting processes, *Biotechnol. J.* 4 (2009) 1168–1177. <https://doi.org/10.1002/biot.200900004>.

[52] W. Liu, M.A. Heinrich, Y. Zhou, A. Akpek, N. Hu, X. Liu, X. Guan, Z. Zhong, X. Jin, A. Khademhosseini, Y.S. Zhang, Extrusion Bioprinting of Shear-Thinning Gelatin Methacryloyl Bioinks, *Adv. Healthc. Mater.* 6 (2017) 1–11. <https://doi.org/10.1002/adhm.201601451>.

[53] J. Snyder, A. Rin Son, Q. Hamid, C. Wang, Y. Lui, W. Sun, Mesenchymal stem cell printing and process regulated cell properties, *Biofabrication.* 7 (2015). <https://doi.org/10.1088/1758-5090/7/4/044106>.

[54] N. Artrith, K.T. Butler, F.X. Coudert, S. Han, O. Isayev, A. Jain, A. Walsh, Best practices in machine learning for chemistry, *Nat. Chem.* 2021 136. 13 (2021) 505–508. <https://doi.org/10.1038/s41557-021-00716-z>.

[55] J. Shi, B. Wu, B. Song, J. Song, S. Li, D. Trau, W.F. Lu, Learning-Based Cell Injection Control for Precise Drop-on-Demand Cell Printing, *Ann. Biomed. Eng.* 46 (2018) 1267–1279. <https://doi.org/10.1007/s10439-018-2054-2>.

[56] J. Shi, J. Song, B. Song, W.F. Lu, Multi-Objective Optimization Design through Machine

Learning for Drop-on-Demand Bioprinting, *Engineering*. 5 (2019) 586–593.

<https://doi.org/10.1016/j.eng.2018.12.009>.

[57] A. Menon, B. Póczos, A.W. Feinberg, N.R. Washburn, Optimization of Silicone 3D Printing with Hierarchical Machine Learning, *3D Print. Addit. Manuf.* 6 (2019) 181–189.

<https://doi.org/10.1089/3dp.2018.0088>.

[58] K. Ruberu, M. Senadeera, S. Rana, S. Gupta, J. Chung, Z. Yue, S. Venkatesh, G. Wallace, Coupling machine learning with 3D bioprinting to fast track optimisation of extrusion printing, *Appl. Mater. Today*. 22 (2021) 100914. <https://doi.org/10.1016/j.apmt.2020.100914>.

[59] A. Conev, E.E. Litsa, M.R. Perez, M. Diba, A.G. Mikos, L.E. Kavraki, Machine Learning-Guided Three-Dimensional Printing of Tissue Engineering Scaffolds, *Tissue Eng. Part A*. 26 (2020) 1359–1368. <https://doi.org/10.1089/ten.tea.2020.0191>.

[60] J.E. Trachtenberg, J.K. Placone, B.T. Smith, C.M. Piard, M. Santoro, D.W. Scott, J.P. Fisher, A.G. Mikos, Extrusion-Based 3D Printing of Poly(propylene fumarate) in a Full-Factorial Design, (2016). <https://doi.org/10.1021/acsbiomaterials.6b00026>.

[61] J. Lee, S.J. Oh, S.H. An, W.D. Kim, S.H. Kim, S.H. Kim, Machine learning-based design strategy for 3D printable bioink: Elastic modulus and yield stress determine printability, *Biofabrication*. 12 (2020). <https://doi.org/10.1088/1758-5090/ab8707>.

[62] H. Xu, Q. Liu, J. Casillas, M. Mcanally, N. Mubtasim, L.S. Gollahon, D. Wu, C. Xu, Prediction of cell viability in dynamic optical projection stereolithography-based bioprinting using machine learning, *J. Intell. Manuf.* (2020). <https://doi.org/10.1007/s10845-020-01708-5>.

[63] S. Tian, Machine Assisted Experimentation of Extrusion-based Bioprinting Systems, (2021). <https://doi.org/10.17605/OSF.IO/97DKX>.

[64] F. Abe, Exploration of the Effects of High Hydrostatic Pressure on Microbial Growth, *Physiol. Surviv. Perspect. from Piezophysiology*. 71 (2007) 2347–2357. <https://doi.org/10.1271/bbb.70015>.

- [65] L.D. Loozen, F. Wegman, F.C. Öner, W.J.A. Dhert, J. Alblas, Porous bioprinted constructs in BMP-2 non-viral gene therapy for bone tissue engineering, *J. Mater. Chem. B.* 1 (2013) 6619–6626. <https://doi.org/10.1039/c3tb21093f>.
- [66] J. Zhang, E. Wehrle, J.R. Vetsch, G.R. Paul, M. Rubert, R. Müller, Alginate dependent changes of physical properties in 3D bioprinted cell-laden porous scaffolds affect cell viability and cell morphology Recent citations Alginate dependent changes of physical properties in 3D bioprinted cell-laden porous scaffolds affect cell viability and cell morphology, (2019). <https://doi.org/10.1088/1748-605X/ab3c74>.
- [67] R. Bhuthalingam, P.Q. Lim, S.A. Irvine, S.S. Venkatraman, Automated Robotic Dispensing Technique for Surface Guidance and Bioprinting of Cells, *J. Vis. Exp.* (2016) 54604. <https://doi.org/10.3791/54604>.
- [68] R. Chang, J. Nam, W. Sun, Effects of dispensing pressure and nozzle diameter on cell survival from solid freeform fabrication-based direct cell writing, *Tissue Eng. - Part A.* 14 (2008) 41–48. <https://doi.org/10.1089/ten.a.2007.0004>.
- [69] Y. Yu, Y. Zhang, J.A. Martin, I.T. Ozbolat, Evaluation of cell viability and functionality in vessel-like bioprintable cell-laden tubular channels, *J. Biomech. Eng.* 135 (2013) 1–9. <https://doi.org/10.1115/1.4024575>.
- [70] G. Cidonio, C.R. Alcalá-Orozco, K.S. Lim, M. Glinka, I. Mutreja, Y.-H. Kim, J.I. Dawson, T.B.F. Woodfield, R.O.C. Oreffo, Osteogenic and angiogenic tissue formation in high fidelity nanocomposite Laponite-gelatin bioinks, *Biofabrication.* 11 (2019) 35027. <https://doi.org/10.1088/1758-5090/ab19fd>.
- [71] A. Mondal, A. Gebeyehu, M. Miranda, D. Bahadur, N. Patel, S. Ramakrishnan, A.K. Rishi, M. Singh, Characterization and printability of Sodium alginate -Gelatin hydrogel for bioprinting NSCLC co-culture, *Sci. Rep.* 9 (2019) 1–12. <https://doi.org/10.1038/s41598-019-55034-9>.

- [72] R. Suntornnond, E.Y.S. Tan, J. An, C.K. Chua, A Mathematical Model on the Resolution of Extrusion Bioprinting for the Development of New Bioinks, *Mater.* 2016, Vol. 9, Page 756. 9 (2016) 756. <https://doi.org/10.3390/MA9090756>.
- [73] T.J. Hinton, Q. Jallerat, R.N. Palchesko, J.H. Park, M.S. Grodzicki, H.J. Shue, M.H. Ramadan, A.R. Hudson, A.W. Feinberg, Three-dimensional printing of complex biological structures by freeform reversible embedding of suspended hydrogels, *Sci. Adv.* 1 (2015) 1–10. <https://doi.org/10.1126/sciadv.1500758>.
- [74] A.M. Compaan, K. Song, W. Chai, Y. Huang, Cross-Linkable Microgel Composite Matrix Bath for Embedded Bioprinting of Perfusable Tissue Constructs and Sculpting of Solid Objects, *Cite This ACS Appl. Mater. Interfaces.* 12 (2020) 7868. <https://doi.org/10.1021/acsami.9b15451>.
- [75] C.B. Highley, C.B. Rodell, J.A. Burdick, C.B. Highley, C.B. Rodell, J.A. Burdick, Direct 3D Printing of Shear-Thinning Hydrogels into Self-Healing Hydrogels, *Adv. Mater.* 27 (2015) 5075–5079. <https://doi.org/10.1002/ADMA.201501234>.
- [76] O. Jeon, Y. Bin Lee, H. Jeong, S.J. Lee, D. Wells, E. Alsberg, Individual cell-only bioink and photocurable supporting medium for 3D printing and generation of engineered tissues with complex geometries, *Mater. Horizons.* 6 (2019) 1625–1631. <https://doi.org/10.1039/C9MH00375D>.
- [77] J. Lewicki, J. Bergman, C. Kerins, O. Hermanson, Optimization of 3D bioprinting of human neuroblastoma cells using sodium alginate hydrogel, *Bioprinting.* 16 (2019). <https://doi.org/10.1016/j.bprint.2019.e00053>.
- [78] V. Fantini, M. Bordoni, F. Scocozza, M. Conti, E. Scarian, S. Carelli, A.M. Di Giulio, S. Marconi, O. Pansarasa, F. Auricchio, C. Cereda, Bioink Composition and Printing Parameters for 3D Modeling Neural Tissue, *Cells* 2019, Vol. 8, Page 830. 8 (2019) 830. <https://doi.org/10.3390/CELLS8080830>.
- [79] L. Lucas, A. Aravind, P. Emma, M. Christophe, C. Edwin-Joffrey, Rheology, simulation and

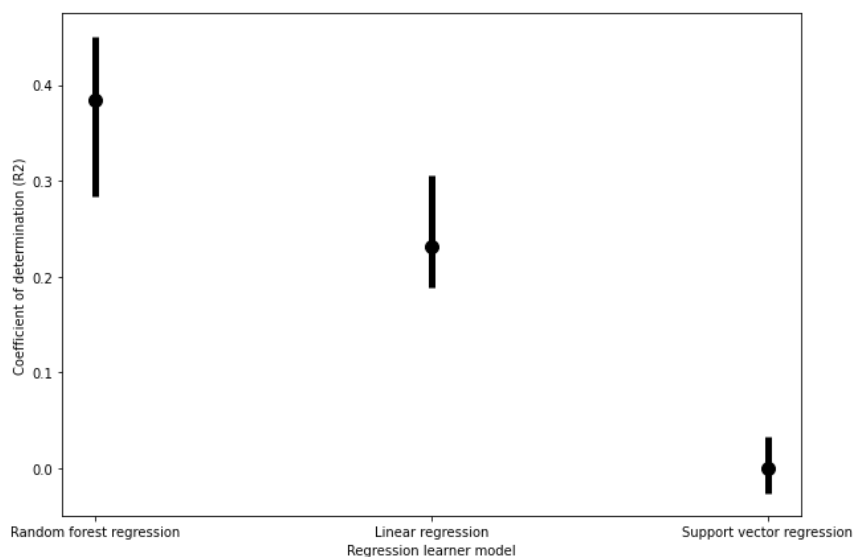
data analysis toward bioprinting cell viability awareness, *Bioprinting*. 21 (2021) e00119.

<https://doi.org/10.1016/J.BPRINT.2020.E00119>.

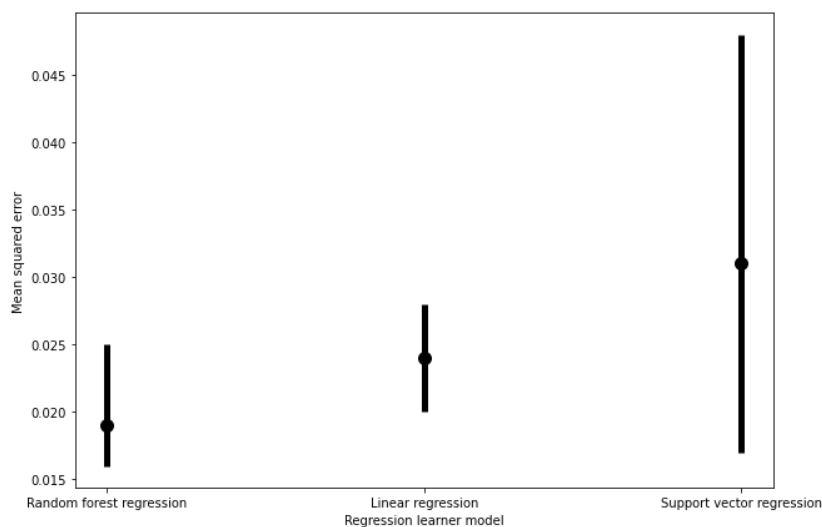
- [80] W. Jia, P.S. Gungor-Ozkerim, Y.S. Zhang, K. Yue, K. Zhu, W. Liu, Q. Pi, B. Byambaa, M.R. Dokmeci, S.R. Shin, A. Khademhosseini, Direct 3D bioprinting of perfusable vascular constructs using a blend bioink, *Biomaterials*. 106 (2016) 58–68. <https://doi.org/10.1016/j.biomaterials.2016.07.038>.
- [81] D. Bociaga, M. Bartniak, J. Grabarczyk, K. Przybyszewska, Sodium alginate/gelatine hydrogels for direct bioprinting-the effect of composition selection and applied solvents on the bioink properties, *Materials (Basel)*. 12 (2019) 1–19. <https://doi.org/10.3390/ma12172669>.
- [82] A. Sanz-Garcia, E. Sodupe-Ortega, A. Pernía-Espinoza, T. Shimizu, C. Escobedo-Lucea, A versatile open-source printhead for low-cost 3d microextrusion-based bioprinting, *Polymers (Basel)*. 12 (2020) 1–18. <https://doi.org/10.3390/polym12102346>.
- [83] S. Tian, R. Stevens, B.T. McInnes, N.A. Lewinski, Machine Assisted Experimentation of Extrusion-Based Bioprinting Systems, *Micromachines* 2021, Vol. 12, Page 780. 12 (2021) 780. <https://doi.org/10.3390/MI12070780>.



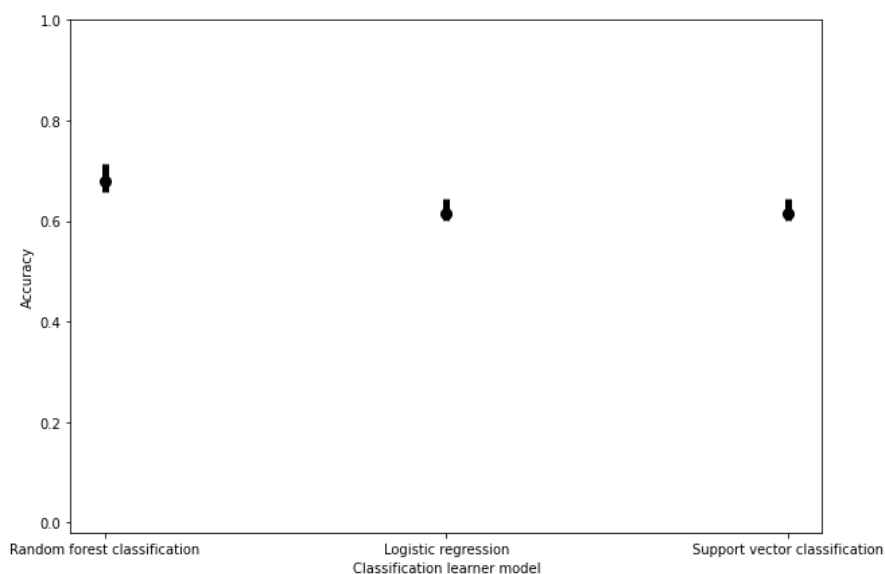
## 9. Appendix A: Supplemental Figures and Tables



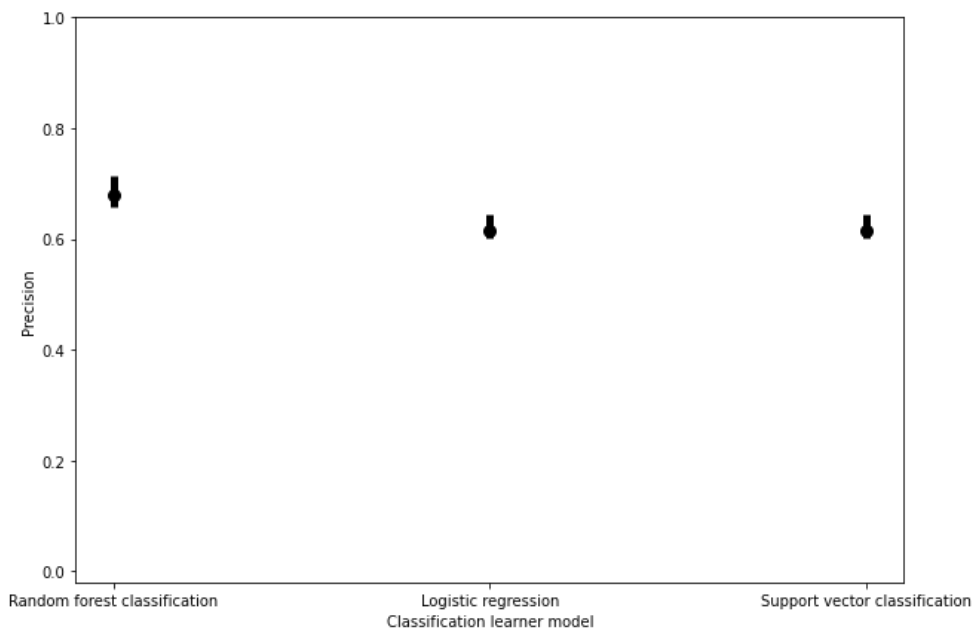
**Figure S1.** Cell viability regression model performance based on coefficients of determination ( $R^2$ ) values under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  values produced amongst the five testing and training combinations.



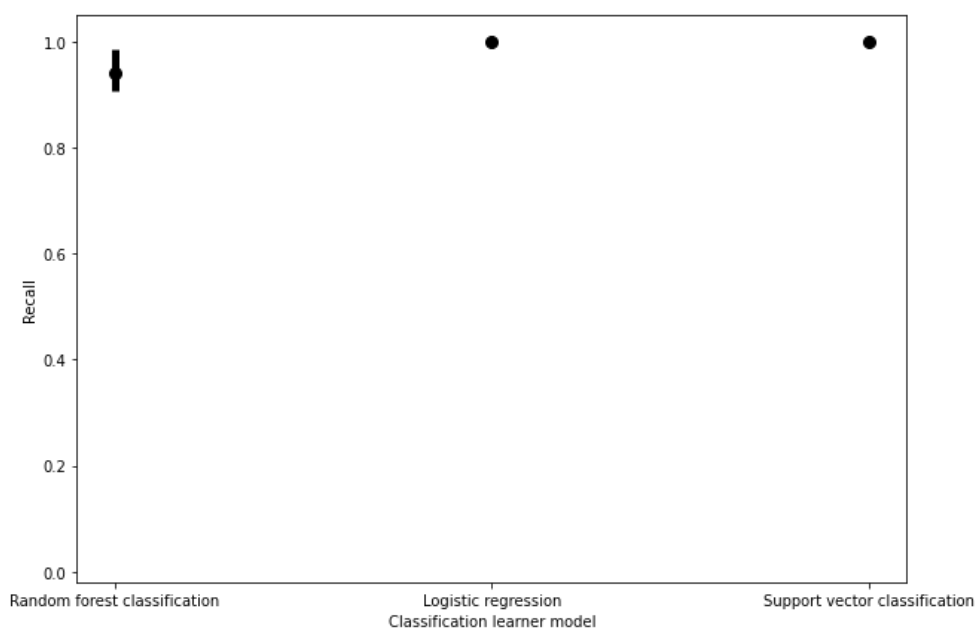
**Figure S2.** Cell viability regression model performance based on mean squared error (MSE) values under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum MSE values produced amongst the five testing and training combinations.



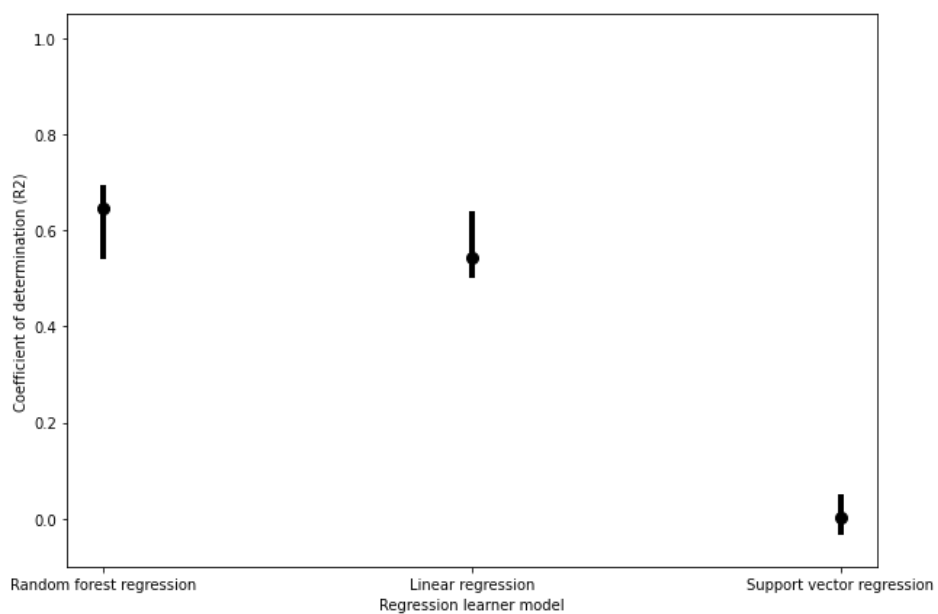
**Figure S3.** Cell viability classification model performance based on accuracy scores under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum scores produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



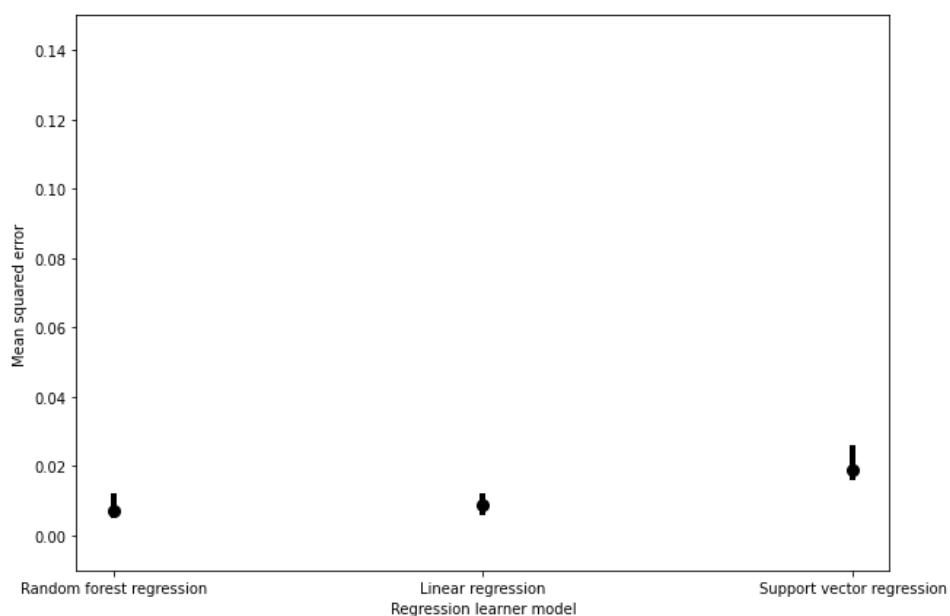
**Figure S4.** Cell viability classification model performance based on precision scores under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum scores produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



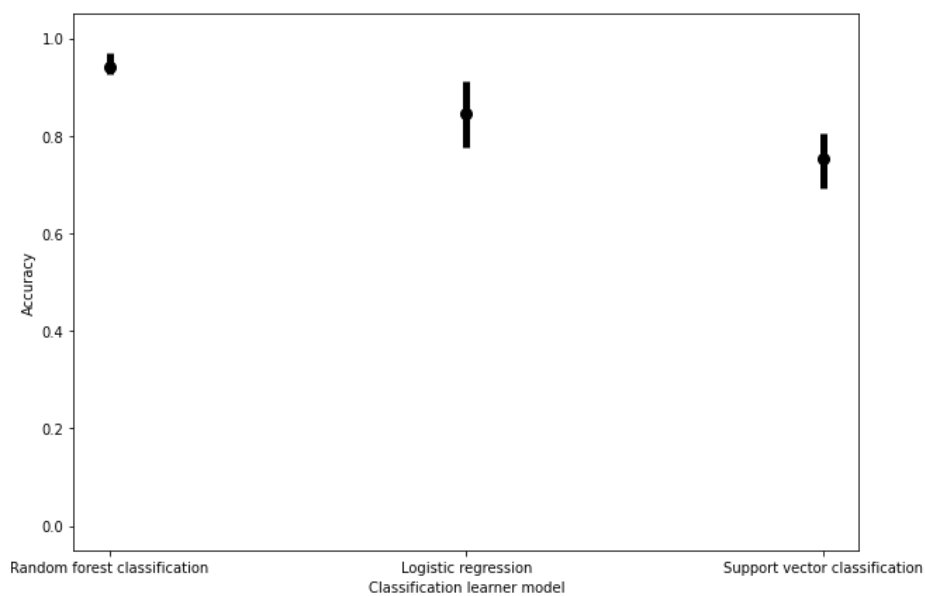
**Figure S5.** Cell viability classification model performance based on recall scores under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum scores produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



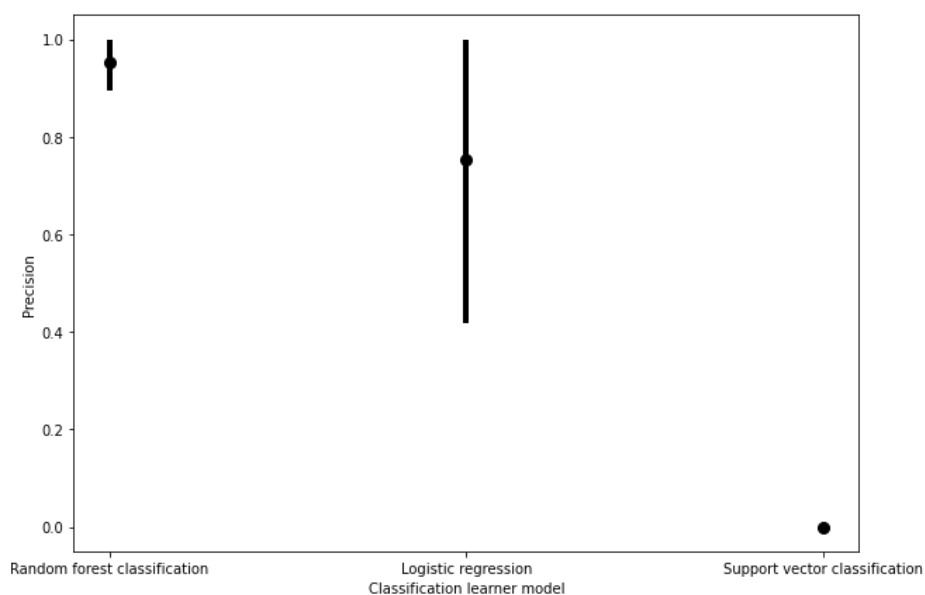
**Figure S6.** Filament diameter regression model performance based on coefficient of determination ( $R^2$ ) values under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  and MSE values produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



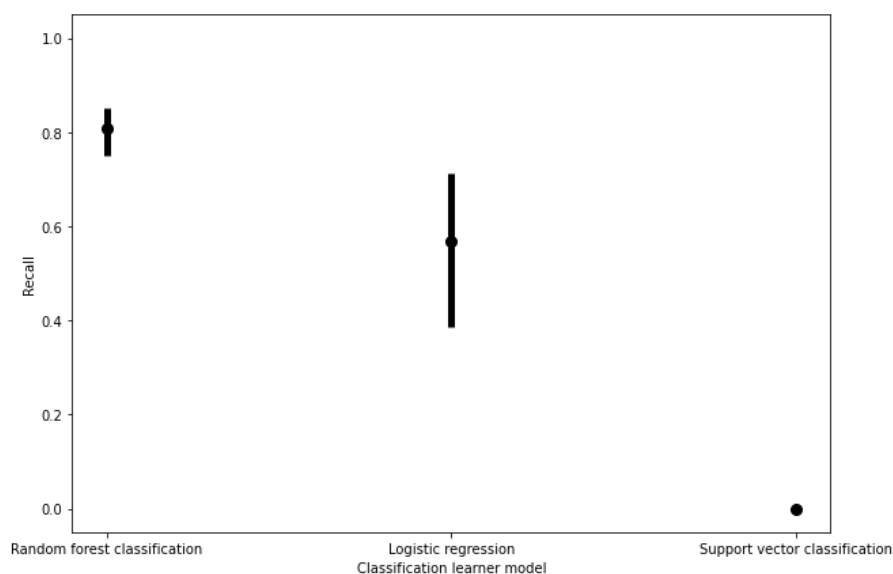
**Figure S7.** Filament diameter regression model performance based on mean squared error (MSE) values under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  and MSE values produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



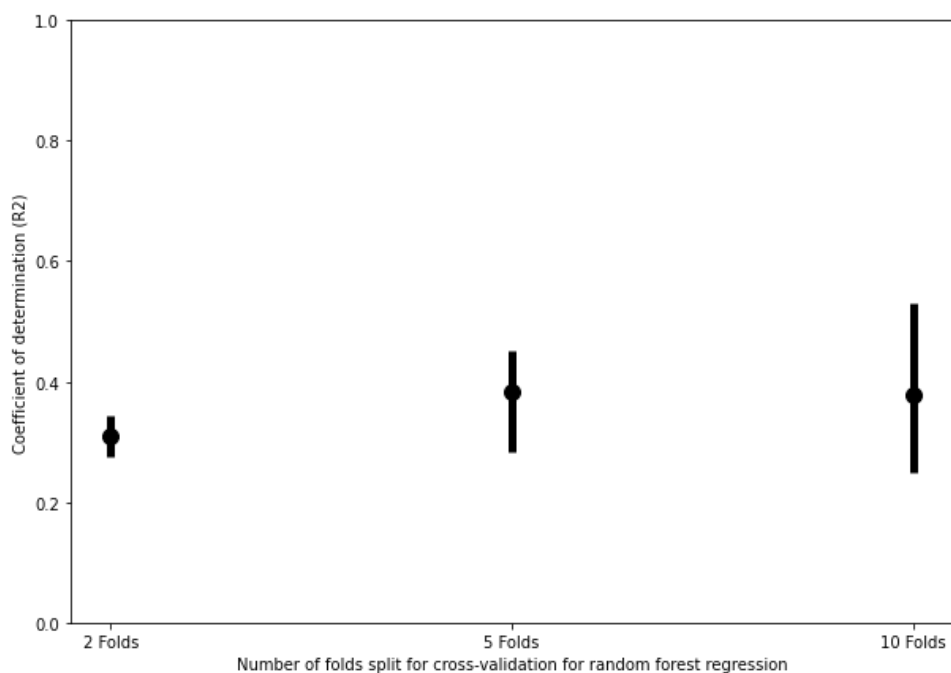
**Figure S8.** Filament diameter classification model performance based on accuracy scores under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum scores produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



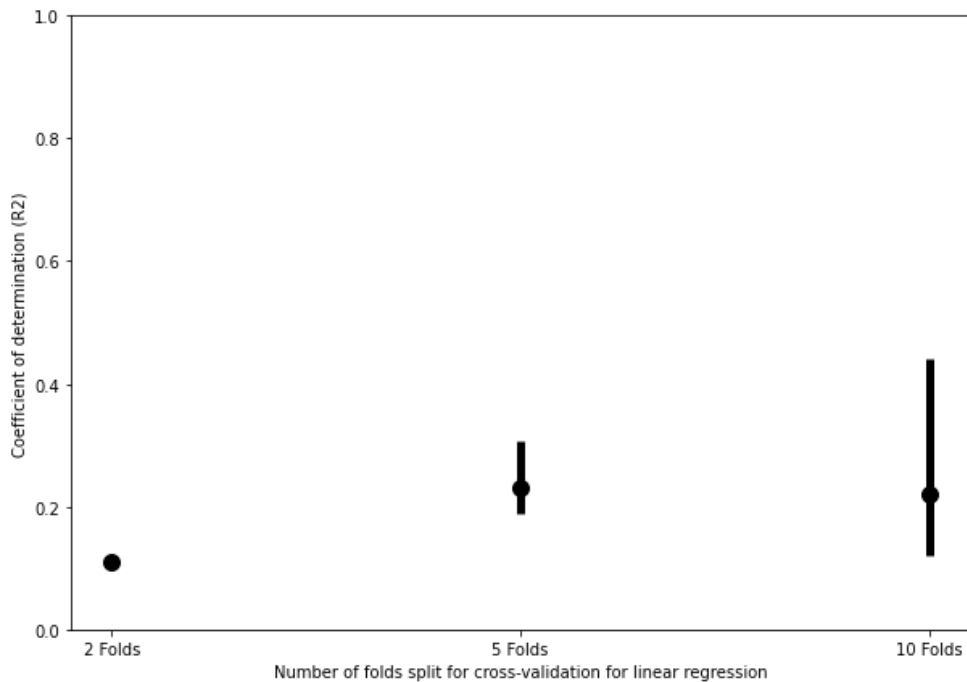
**Figure S9.** Filament diameter classification model performance based on precision scores under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum scores produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



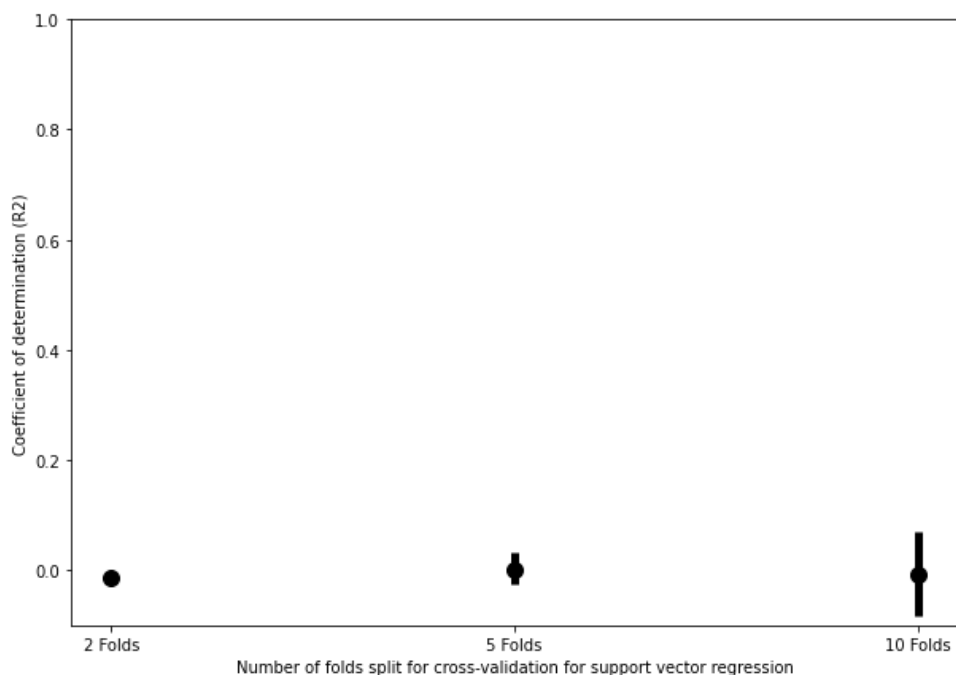
**Figure S10.** Filament diameter classification model performance based on recall scores under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum scores produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



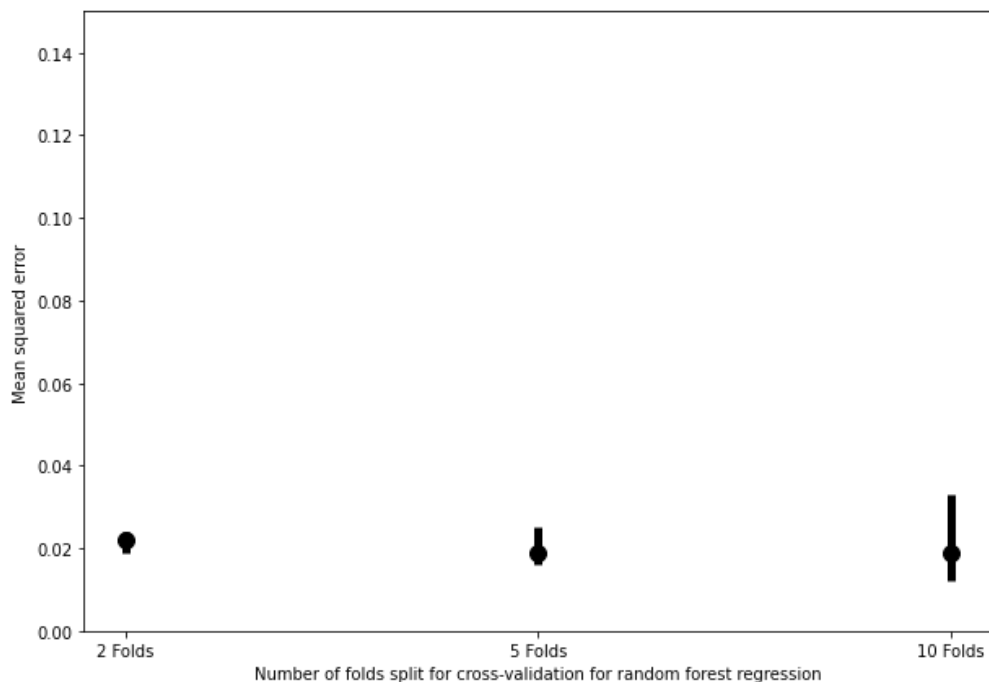
**Figure S11.** Coefficient of determination ( $R^2$ ) values of cell viability regression models based on the number of folds tested for random forest regression. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  values produced for each fold division.



**Figure S12.** Coefficient of determination ( $R^2$ ) values of cell viability regression models based on the number of folds tested for **linear regression**. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  values produced for each fold division.

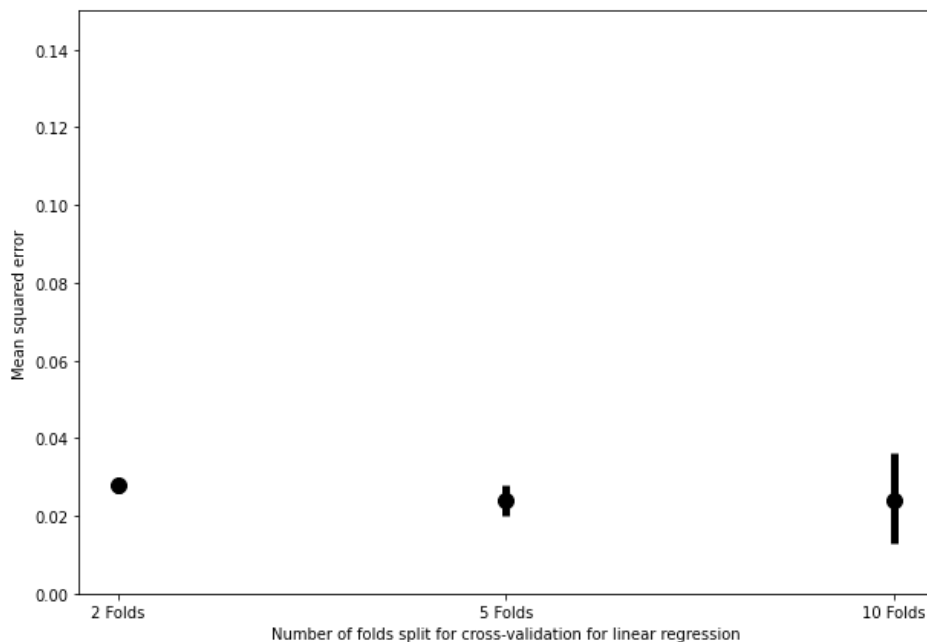


**Figure S13.** Coefficient of determination ( $R^2$ ) values of cell viability regression models based on the number of folds tested for support vector regression. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  values produced for each fold division.

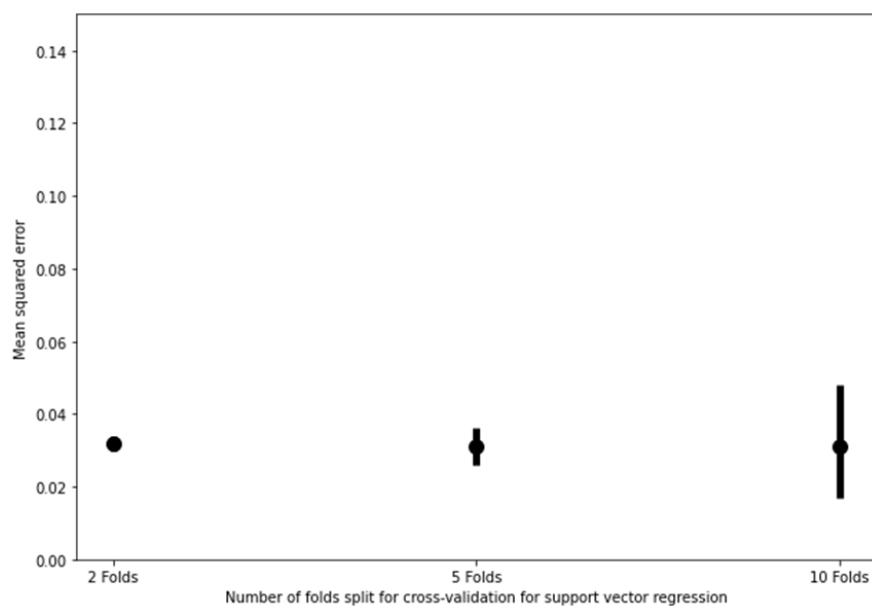


**Figure S14.** Mean squared error (MSE) values of cell viability regression models based on the number of folds tested for **random forest regression**. The upper and lower bounds of the error plots represent the maximum and minimum mean square error values produced for each fold

division.

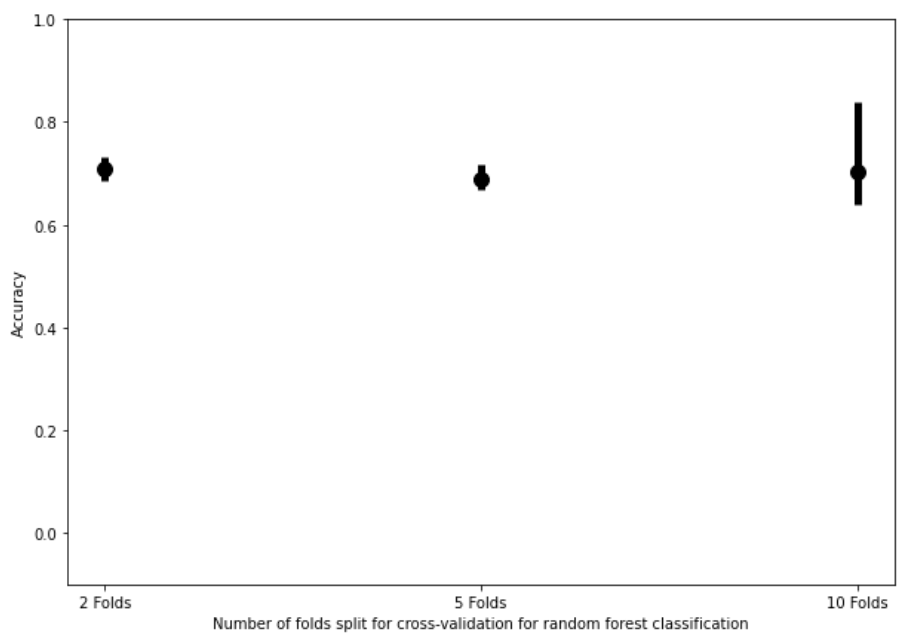


**Figure S15.** Mean squared error values of cell viability regression models based on the number of folds tested for **linear regression**. The upper and lower bounds of the error plots represent the maximum and minimum mean square error values produced for each fold division.

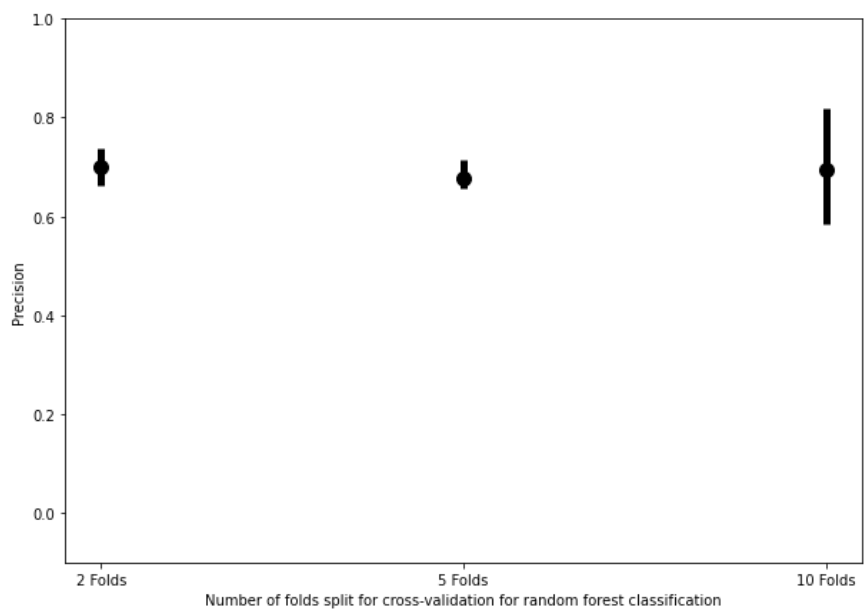


**Figure S16.** Mean squared error values of cell viability regression models based on the number of folds tested for **support vector regression**. The upper and lower bounds of the error plots represent the maximum and minimum mean square error values produced for each fold division.





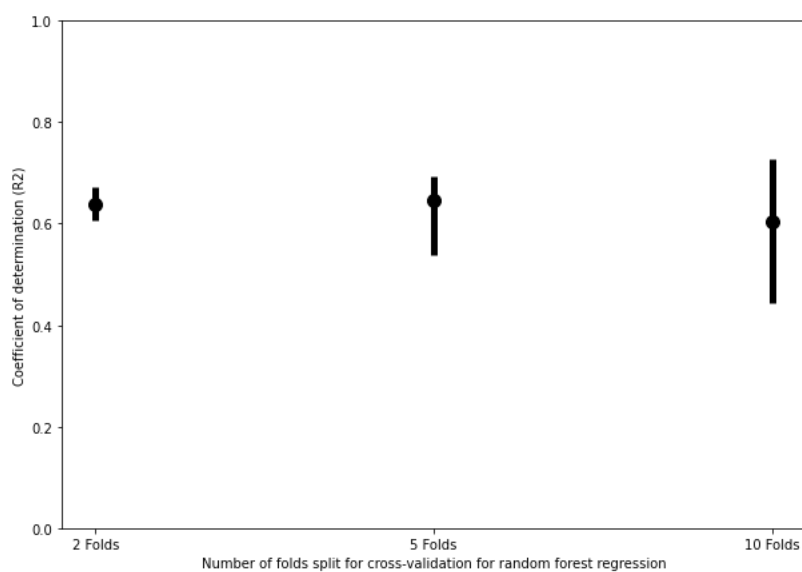
**Figure S17.** Accuracy performance of the random forest classification cell viability model on different k-fold cross validation tests. The upper and lower bounds of the error plots represent the maximum and minimum metric values produced for each fold division.



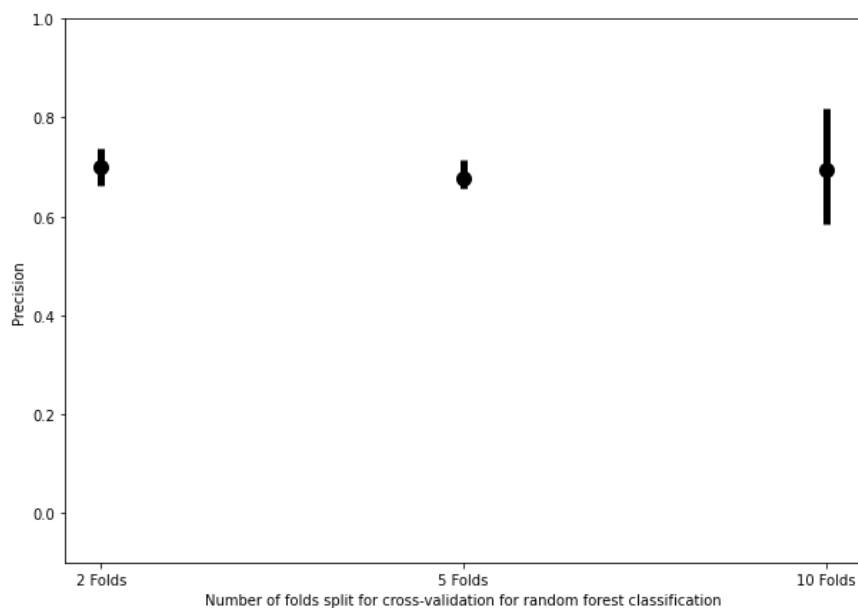
**Figure S18.** Precision performance of the random forest classification cell viability model on different k-fold cross validation tests. The upper and lower bounds of the error plots represent the maximum and minimum metric values produced for each fold division.



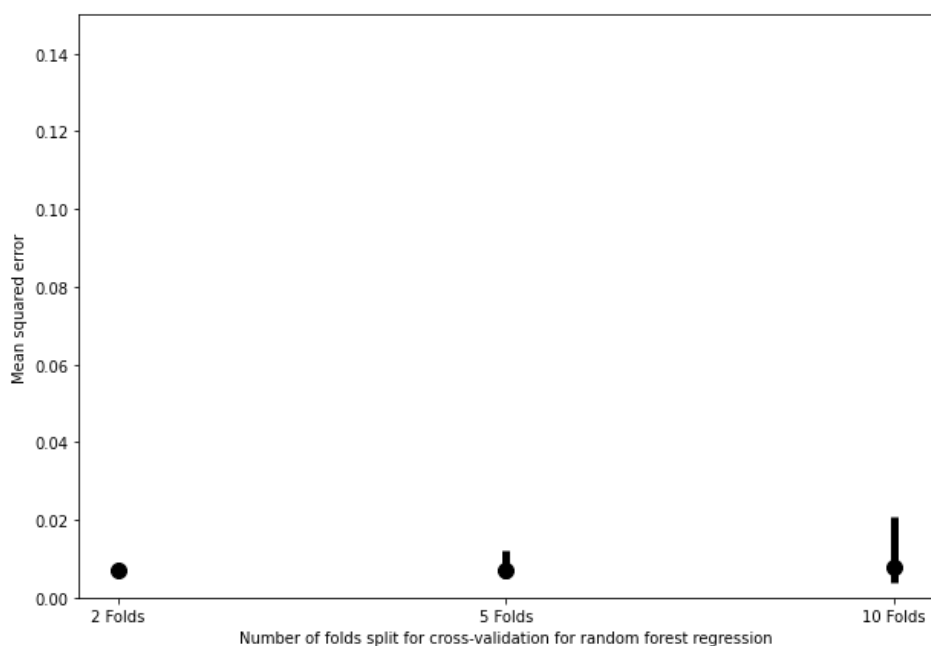
**Figure S19.** Recall performance of the random forest classification cell viability model on different k-fold cross validation tests. The upper and lower bounds of the error plots represent the maximum and minimum metric values produced for each fold division.



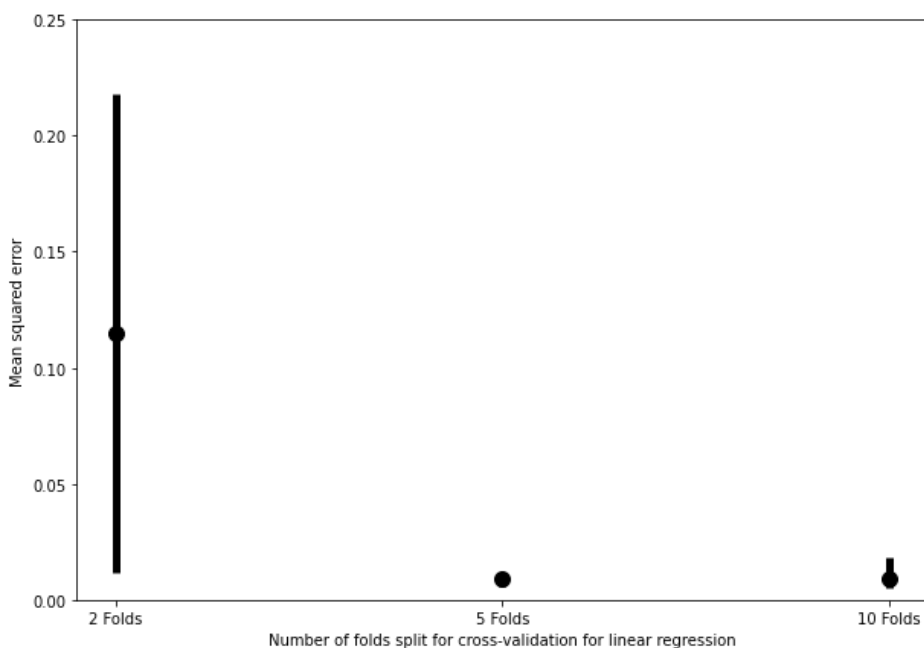
**Figure S20.** Coefficient of determination ( $R^2$ ) scores of filament diameter regression models based on the number of folds tested for random forest regression. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  produced for each fold division.



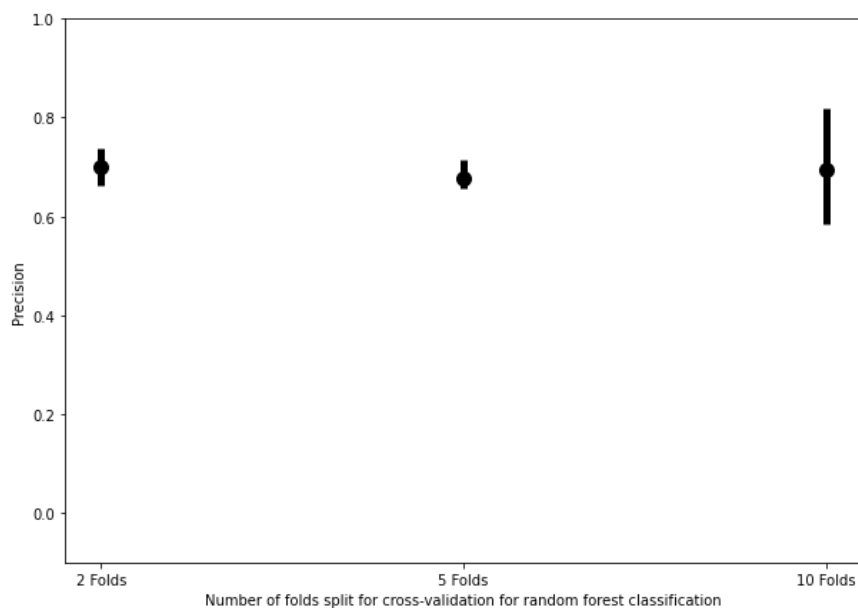
**Figure S21.** Coefficient of determination ( $R^2$ ) scores of filament diameter regression models based on the number of folds tested for linear regression. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  produced for each fold division.



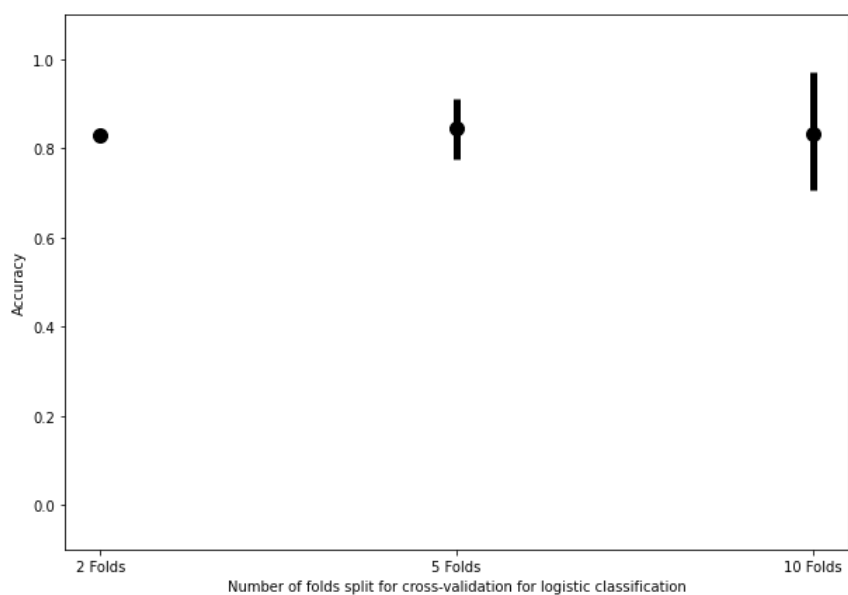
**Figure S22.** Mean squared error (MSE) scores of filament diameter regression models based on the number of folds tested for random forest regression. The upper and lower bounds of the error plots represent the maximum and minimum mean squared error produced for each fold division.



**Figure S23.** Mean squared error scores (MSE) of filament diameter regression models based on the number of folds tested for linear regression. The upper and lower bounds of the error plots represent the maximum and minimum mean squared error produced for each fold division.



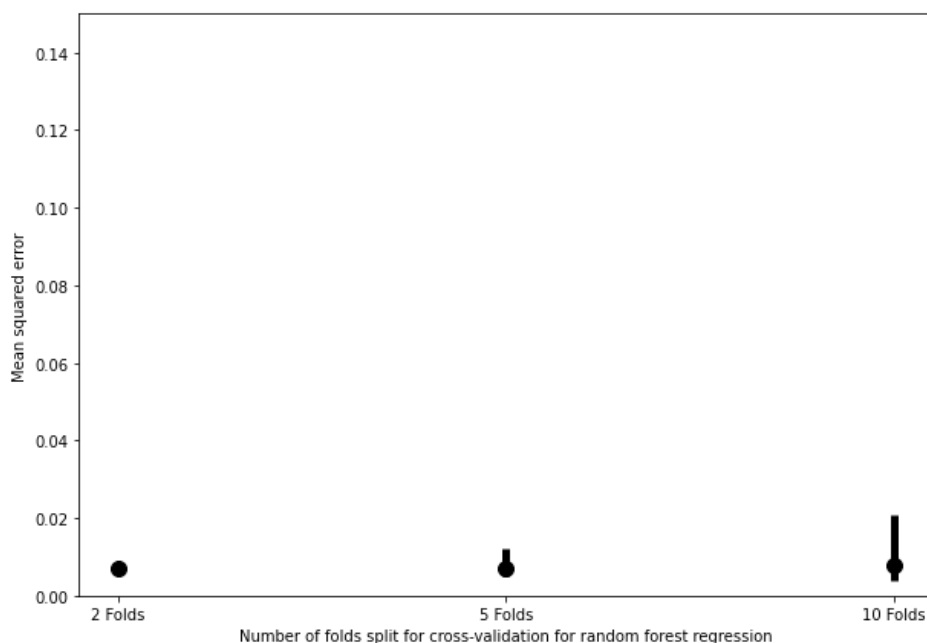
**Figure S24.** Accuracy scores of filament diameter classification models based on the number of folds tested for random forest regression models. The upper and lower bounds of the error plots represent the maximum and minimum accuracy produced for each fold division.



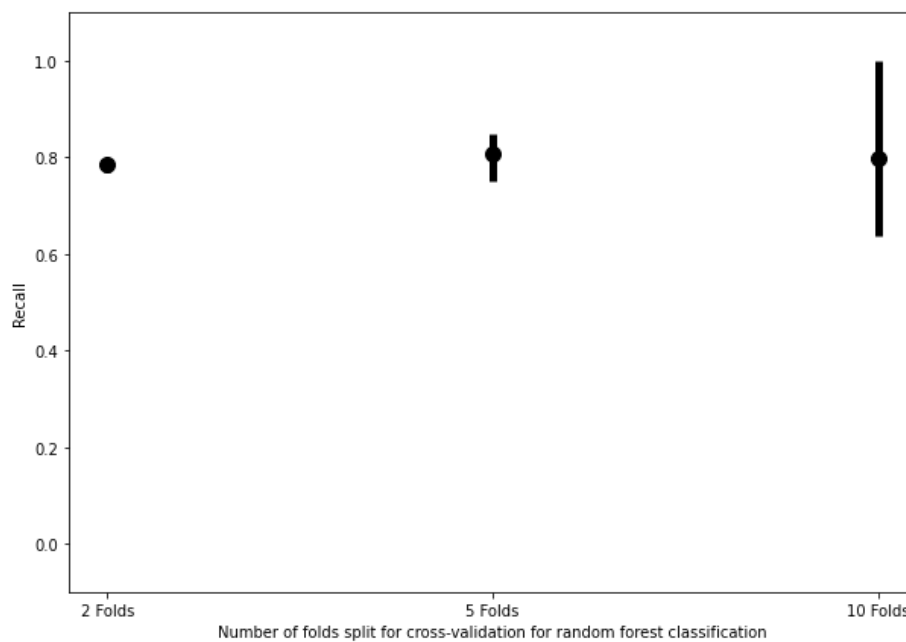
**Figure S25.** Accuracy scores of filament diameter classification models based on the number of folds tested for logistic regression models. The upper and lower bounds of the error plots represent the maximum and minimum accuracy produced for each fold



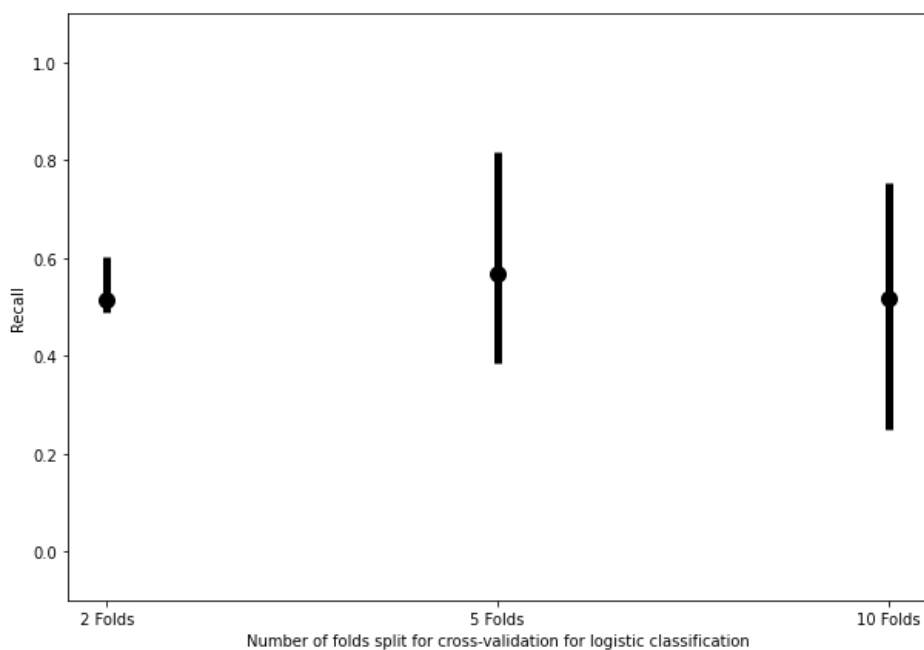
**Figure S26.** Precision scores of filament diameter classification models based on the number of folds tested for random forest regression models. The upper and lower bounds of the error plots represent the maximum and minimum precision produced for each fold division.



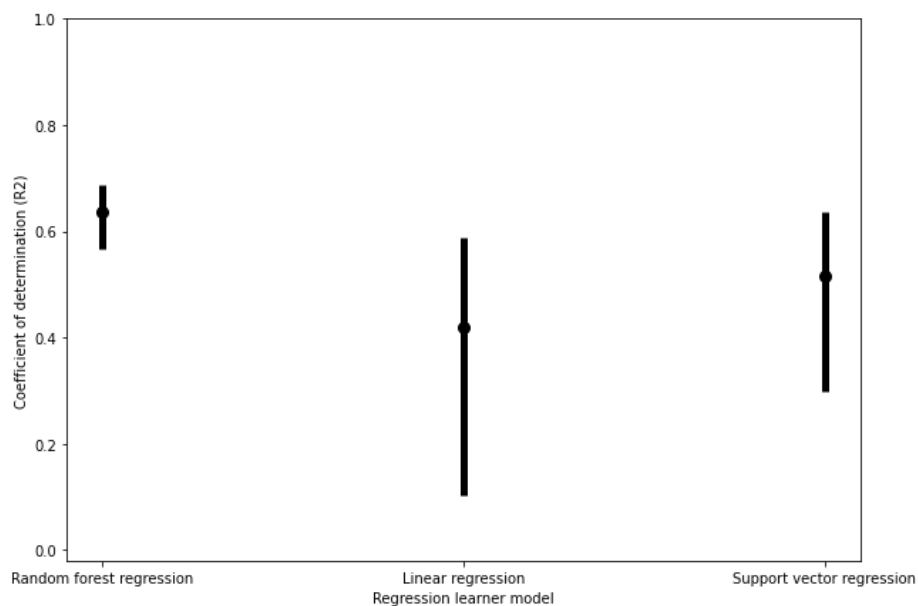
**Figure S27.** Precision scores of filament diameter classification models based on the number of folds tested for logistic regression models. The upper and lower bounds of the error plots represent the maximum and minimum precision produced for each fold division.



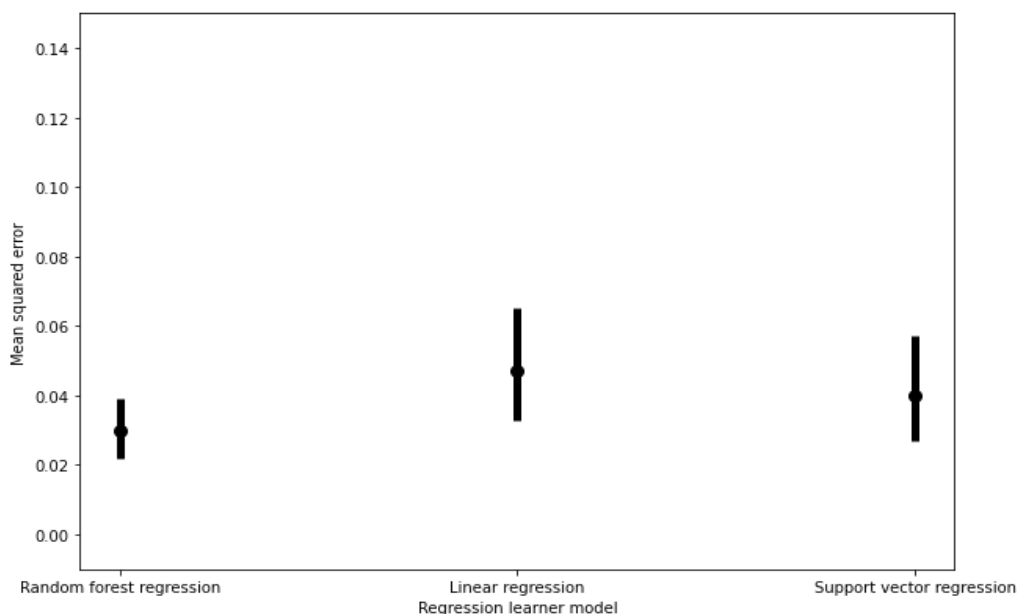
**Figure S28.** Recall scores of filament diameter classification models based on the number of folds tested for random forest regression models. The upper and lower bounds of the error plots represent the maximum and minimum recall produced for each fold division.



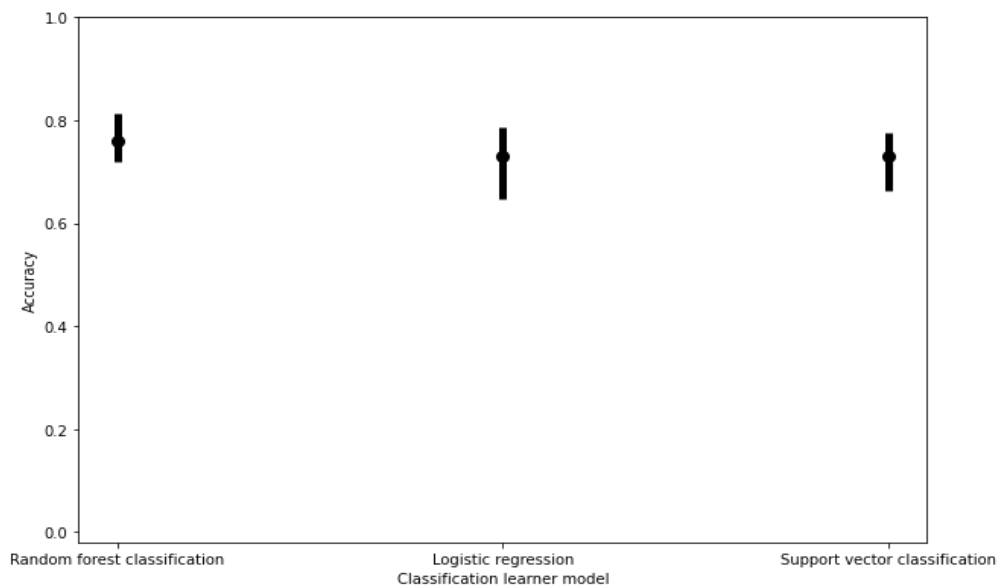
**Figure S29.** Recall scores of filament diameter classification models based on the number of folds tested for logistic regression models. The upper and lower bounds of the error plots represent the maximum and minimum recall produced for each fold division.



**Figure S30.** Extrusion pressure regression model performance based on coefficient of determination ( $R^2$ ) values under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  and MSE values produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.

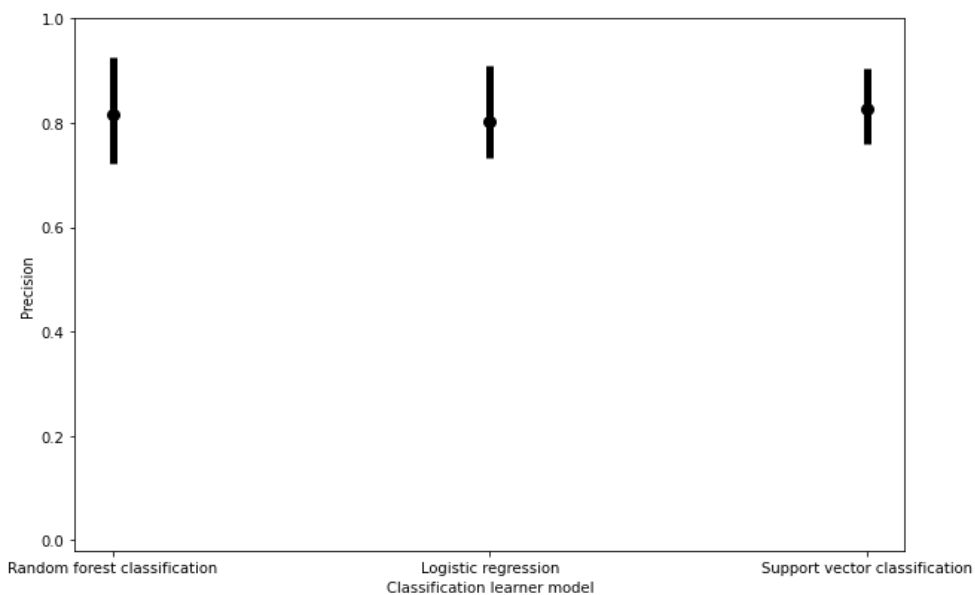


**Figure S31.** Extrusion pressure regression model performance based on coefficient of determination ( $R^2$ ) values under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum  $R^2$  and MSE values produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.

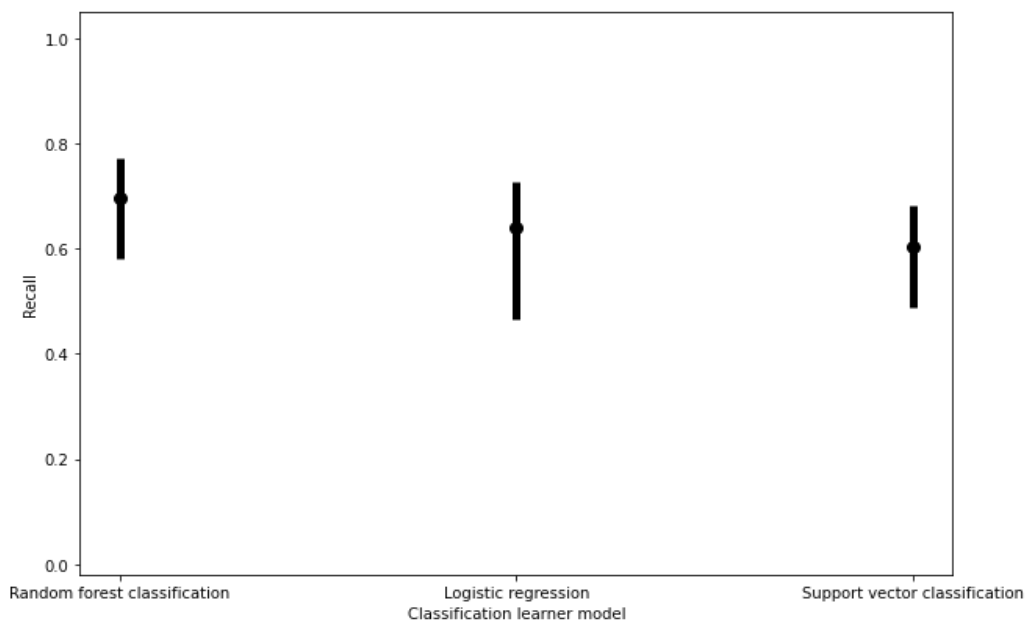


**Figure S32.** Extrusion pressure classification model performance based on accuracy scores under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum scores produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.

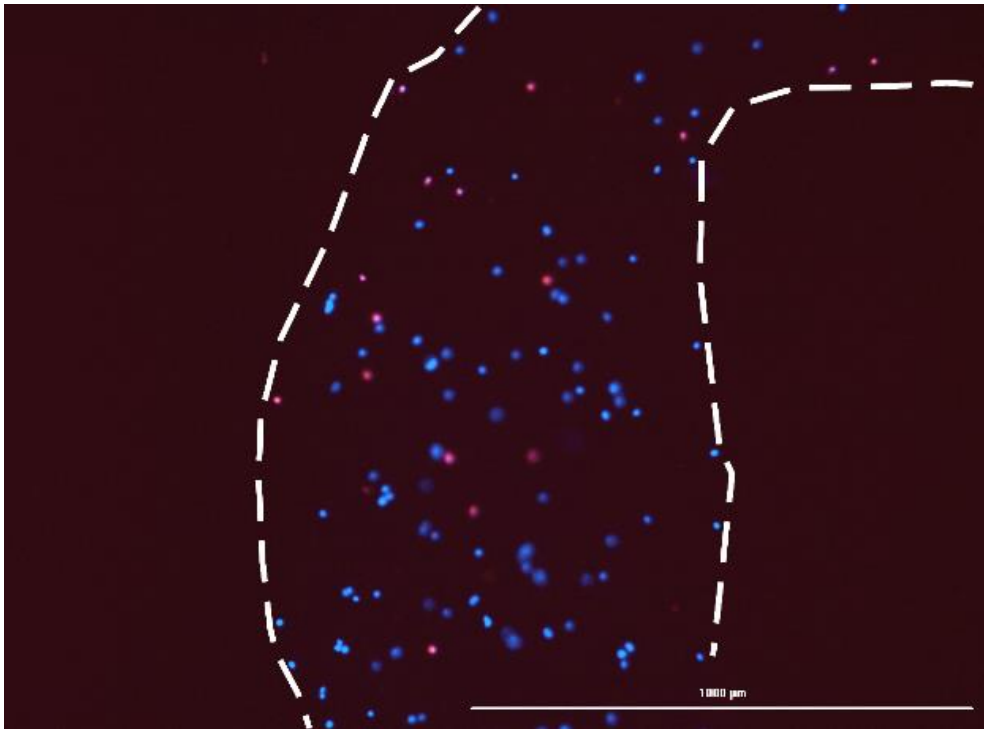




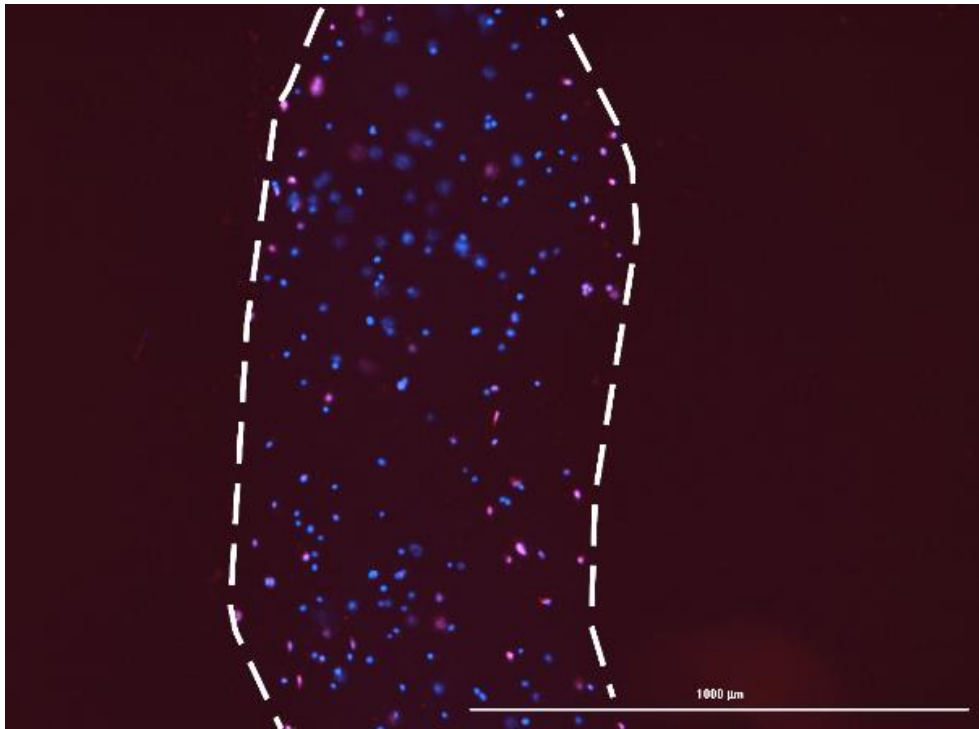
**Figure S33.** Extrusion pressure classification model performance based on precision scores under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum scores produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



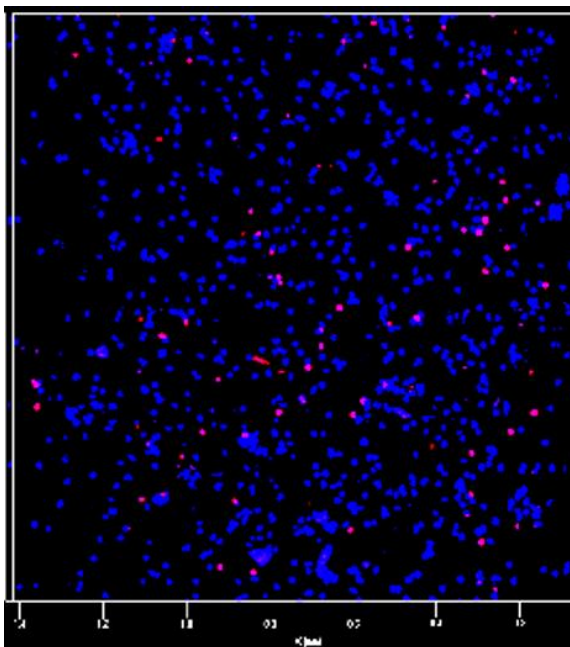
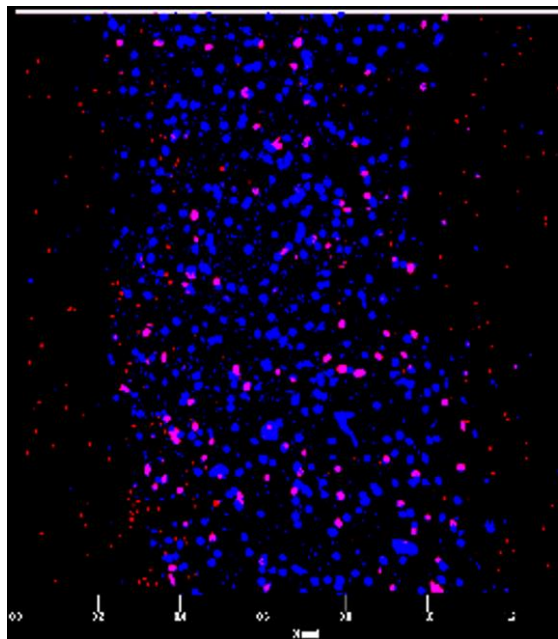
**Figure S34.** Cell viability classification model performance based on recall scores under 5-fold cross validation. The upper and lower bounds of the error plots represent the maximum and minimum scores produced amongst all 5 combinations of one fold being trained and tested on the remaining 4 folds.



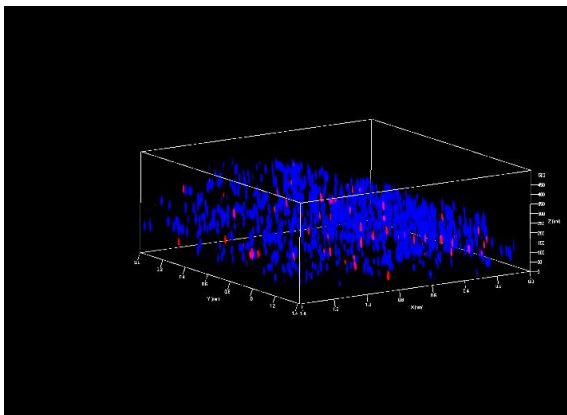
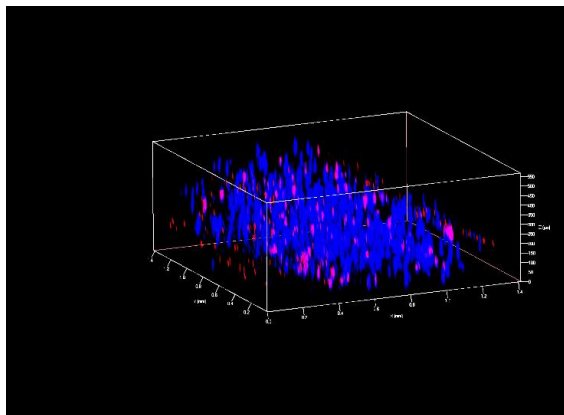
**Figure S35.** Live/dead images taken on through the imaging plate reader immediately after extrusion of 3/4 Alg/Gel. White borders indicate boundaries of the filament.



**Figure S36.** Live/dead images taken on through the imaging plate reader immediately after extrusion of 3/7 Alg/Gel. White borders indicate boundaries of the filament.

**A****B**

**Figure S37.** Total/dead top-view confocal imaging immediately after extrusion of **A)** 3/4 Alg/Gel and **B)** 3/7 Alg/Gel. The frames of images are 1.4 by 1.4 mm in dimension.

**A****B**

**Figure S38.** Isometric view of total/dead confocal imaging for **A)** 3/4 Alg/Gel and **B)** 3/7 Alg/Gel immediately after extrusion (nozzle geometry = conical, nozzle diameter = 410  $\mu\text{m}$ ). The frames of images are 1.4 by 1.4 mm by 0.5 mm in dimension.

**Table S1.** Predicted cell viability and actual cell viability comparison of 3/4 and 3/7 Alg/Gel constructs printed (nozzle geometry = conical, nozzle diameter = 410  $\mu\text{m}$ ).

<b>Cell viability prediction model</b>	<b>Material concentration (%w/v)</b>	<b>Predicted cell viability acceptability (Yes/No)</b>	<b>Actual cell viability acceptability (Yes/No)</b>
Random forest classification	3/4 Alg/Gel	Yes	Yes
	3/7 Alg/Gel	Yes	No
Logistic regression	3/4 Alg/Gel	Yes	Yes
	3/7 Alg/Gel	Yes	No
Support vector classification	3/4 Alg/Gel	Yes	Yes
	3/7 Alg/Gel	Yes	No

**Table S2.** Predicted tolerance and actual tolerance comparison of 3/4 and 3/7 Alg/Gel constructs printed (nozzle geometry = conical, nozzle diameter = 410  $\mu\text{m}$ ).

<b>Cell viability prediction model</b>	<b>Material concentration (%w/v)</b>	<b>Predicted tolerance condition</b>	<b>Percent error from nozzle diameter (410 <math>\mu\text{m}</math>) (%)</b>	<b>Actual tolerance condition</b>
Random forest classification	3/4 Alg/Gel	Not within tolerance	126	Not within tolerance
	3/7 Alg/Gel	Not within tolerances	72.5	Not within tolerance
Logistic regression	3/4 Alg/Gel	Yes	126	Yes
	3/7 Alg/Gel	Yes	72.5	No
Support vector classification	3/4 Alg/Gel	Yes	126	Yes
	3/7 Alg/Gel	Yes	72.5	No

**Table S3.** Predicted extrusion pressure classifications compared against experimental outcomes for corresponding material concentrations of Alg/Gel. Actual values represent the mean  $\pm$  standard deviation for all samples (n = number of batches).

<b>Extrusion pressure prediction model</b>	<b>Material and material concentration (%w/v)</b>	<b>Acceptable extrusion pressure predicted (Yes/No)</b>	<b>Actual extrusion pressure acceptability (Yes/No)</b>
Random forest classification	3/4 Alg/Gel	Yes	Yes
	3/7 Alg/Gel	No	Yes
	8/20 Alg/Gel	No	No
Logistic regression	3/4 Alg/Gel	Yes	Yes
	3/7 Alg/Gel	Yes	Yes
	8/20 Alg/Gel	Yes	No
Support vector classification	3/4 Alg/Gel	Yes	Yes
	3/7 Alg/Gel	Yes	Yes
	8/20 Alg/Gel	Yes	No



Tables S4-9 indicate results of random forest regression-based filament diameter predictions. The following parameters for outcome prediction are set constant to the following values and conditions unless the specific variable is varied to examine its effect on filament diameter prediction:

1. Physical crosslinking duration = 60 seconds
2. Photocrosslinking duration = 0 seconds
3. Inner nozzle outer diameter = 0  $\mu\text{m}$ ,
4. Outer nozzle inner diameter = 410  $\mu\text{m}$
5. Cell density =  $10^6$  cells/mL,
6. Syringe temperature = 22.5  $^{\circ}\text{C}$ .
7. Printing temperature = 22.5  $^{\circ}\text{C}$
8. Days observed = 0 days,
9. Solvent = cell culture medium only
10. Extrusion pressure = 101.3 kPa

**Table S4.** Effect of alginate concentration on filament diameter predictions through random forest regression modeling using the generalized dataset. Gelatin concentration was set to a constant value of 3% (w/v).

<b>Alginate (%w/v)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
0	734.4677
1	734.4677
2	734.4677
3	734.4677
4	734.4677
5	734.4677
6	734.4677
7	734.4677
8	734.4677
9	734.4677
10	734.4677
11	734.4677
12	734.4677
13	734.4677
14	734.4677
15	734.4677
16	734.4677
17	734.4677
18	734.4677
19	734.4677
20	734.4677

**Table S5.** Effect of gelatin concentration on filament diameter predictions through random forest regression modeling using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v).

<b>Gelatin (%w/v)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
0	734.4677
1	734.4677
2	734.4677
3	734.4677
4	734.4677
5	734.4677
6	734.4677
7	734.4677
8	734.4677
9	734.4677
10	734.4677
11	734.4677
12	734.4677
13	734.4677
14	734.4677
15	734.4677
16	734.4677
17	734.4677
18	734.4677
19	734.4677
20	734.4677

**Table S6.** Effect of physical crosslinking duration on filament diameter predictions through random forest regression modeling using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Physical Crosslinking Duration (s)</b> <b>(3/3 % w/v Alg/Gel)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
60	734.4677
120	734.4677
180	734.4677
240	734.4677
300	734.4677
360	734.4677
420	734.4677
480	734.4677
540	<b>734.4677</b>
600	<b>640.1937</b>
660	640.1937
720	640.1937
780	640.1937
840	640.1937
900	640.1937
960	640.1937
1020	640.1937
1080	640.1937
1140	640.1937
1200	640.1937
1260	640.1937
1320	640.1937
1380	640.1937
1440	640.1937
1500	640.1937
1560	640.1937
1620	640.1937
1680	640.1937
1740	640.1937
1800	640.1937

**Table S7.** Effect of extrusion pressure on filament diameter predictions through random forest regression modeling using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Extrusion Pressure (kPa)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
0	1132.115
10	1132.115
20	<b>1132.115</b>
30	<b>885.5386</b>
40	885.5386
50	885.5386
60	885.5386
70	885.5386
80	<b>847.3622</b>
90	<b>734.4677</b>
100	734.4677
110	734.4677
120	734.4677
130	734.4677
140	734.4677
150	734.4677
160	734.4677
170	734.4677
180	734.4677
190	734.4677
200	734.4677
210	734.4677
220	734.4677
230	734.4677
240	734.4677
250	734.4677
260	734.4677
270	734.4677
280	734.4677
290	734.4677
300	734.4677

**Table S8.** Effect of nozzle diameter on filament diameter predictions through random forest regression modeling using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Outer Nozzle Inner Diameter (<math>\mu\text{m}</math>)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
0	419.1145
100	419.1145
200	419.1145
300	460.7804
400	734.4677
500	1114.694
600	1114.694
700	1151.751
800	1151.751
900	1262.503
1000	1262.503
1100	1262.503
1200	1262.503
1300	1262.503
1400	1262.503
1500	1262.503
1600	1262.503
1700	1262.503

**Table S9.** Effect of standard nozzle diameters correlating to standard on filament diameter predictions through random forest regression modeling using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Outer Nozzle Inner Diameter (<math>\mu\text{m}</math>)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
250	419.1145
260	460.7804
410	734.4677
413	734.4677
510	1114.694
610	1151.751
690	1151.751
840	1151.751
860	1151.751

**The following variables do not result in filament diameter prediction differences:**

1. Syringe temperature.
2. Substrate temperature
3. The use of cell culture medium or saline solution
4.  $\text{CaCl}_2$  concentration

For above variables at a constant polymer precursor concentration of 3/3 Alg/Gel, a filament diameter of 734.4677  $\mu\text{m}$  is always predicted.

Tables S10-15 indicate results of linear regression-based filament diameter predictions. The following parameters for outcome prediction are set constant to the following values and conditions unless the specific variable is varied to examine its effect on filament diameter prediction:

1. Physical crosslinking duration = 60 seconds
2. Photocrosslinking duration = 0 seconds
3. Inner nozzle outer diameter = 0  $\mu\text{m}$ ,
4. Outer nozzle inner diameter = 410  $\mu\text{m}$
5. Cell density =  $10^6$  cells/mL,
6. Syringe temperature = 22.5  $^\circ\text{C}$ .
7. Printing temperature = 22.5  $^\circ\text{C}$
8. Days observed = 0 days,
9. Solvent = cell culture medium only
10. Extrusion pressure = 101.3 kPa

**Table S10.** Effect of syringe temperature on filament diameter predictions through linear regression modeling trained using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Syringe Temp (°C)</b>	<b>Filament Diameter (μm)</b>
4	1129.376
5	1120.131
6	1110.886
7	1101.641
8	1092.396
9	1083.151
10	1073.907
11	1064.662
12	1055.417
13	1046.172
14	1036.927
15	1027.682
16	1018.437
17	1009.192
18	999.947
19	990.7021
20	981.4572
21	972.2122
22	962.9673
23	953.7223
24	944.4774
25	935.2325
26	925.9875
27	916.7426
28	907.4976
29	898.2527
30	889.0078
31	879.7628
32	870.5179
33	861.2729
34	852.028
35	842.7831
36	833.5381
37	824.2932
38	815.0483
39	805.8033



**Table S11.** Effect of substrate temperature on filament diameter predictions through linear regression modeling trained using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Substrate Temp (°C)</b>	<b>Filament Diameter (µm)</b>
4	1187.459
5	1185.518
6	1183.576
7	1181.634
8	1179.692
9	1177.751
10	1175.809
11	1173.867
12	1171.925
13	1169.984
14	1168.042
15	1166.1
16	1164.158
17	1162.217
18	1160.275
19	1158.333
20	1156.391
21	1154.45
22	1152.508
23	1150.566
24	1148.624
25	1146.683
26	1144.741
27	1142.799
28	1140.857
29	1138.916
30	1136.974
31	1135.032
32	1133.09
33	1131.149
34	1129.207
35	1127.265
36	1125.323
37	1123.382
38	1121.44
39	1119.498
40	1117.556

**Table S12.** Effect of solvent used on filament diameter predictions through linear regression modeling trained using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Solvent Used</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
DMEM	1151.537
Saline Solution	958.3448

**Table S13.** Effect of alginate concentration on filament diameter predictions through linear regression modeling trained using the generalized dataset. Gelatin concentrations were set to a constant value of 3% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
0	1064.045
1	1093.209
2	1122.373
3	1151.537
4	1180.701
5	1209.865
6	1239.029
7	1268.193
8	1297.357
9	1326.521
10	1355.685
11	1384.849
12	1414.013
13	1443.177
14	1472.341
15	1501.505
16	1530.67
17	1559.834
18	1588.998
19	1618.162
20	1647.326

**Table S14.** Effect of gelatin concentration on filament diameter predictions through linear regression modeling trained using the generalized dataset. Alginate concentrations were set to a constant value of 3% (w/v).

<b>Gelatin Concentration (%w/v)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
0	1159.236
1	1156.67
2	1154.103
3	1151.537
4	1148.971
5	1146.404
6	1143.838
7	1141.272
8	1138.705
9	1136.139
10	1133.573
11	1131.006
12	1128.44
13	1125.874
14	1123.307
15	1120.741
16	1118.175
17	1115.608
18	1113.042
19	1110.476
20	1107.909

**Table S15.** Effect of  $\text{CaCl}_2$  concentration on filament diameter predictions through linear regression modeling trained using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b><math>\text{CaCl}_2</math> Concentration (mM)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
0	1204.1
100	1151.537
200	1098.974
300	1046.41
400	993.8469
500	941.2836
600	888.7202
700	836.1568
800	783.5935
900	731.0301
1000	678.4667

**Table S16.** Effect of physical crosslinking duration on filament diameter predictions through linear regression modeling trained using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Physical Crosslinking Duration (s)</b>	<b>Filament Diameter (<math>\mu\text{m}</math>)</b>
30	1151.537
60	1140.062
90	1128.588
120	1117.113
150	1105.638
180	1094.164
210	1082.689
240	1071.215
270	1059.74
300	1048.265
330	1036.791
360	1025.316
390	1013.841
420	1002.367
450	990.8921
480	979.4174
510	967.9428
540	956.4681
570	944.9935
600	933.5188
630	922.0442
660	910.5696
690	899.0949
720	887.6203
750	876.1456
780	864.671
810	853.1964
840	841.7217
870	830.2471
900	818.7724
930	1151.537
960	1140.062
990	1128.588
1020	1117.113
1050	1105.638
1080	1094.164
1110	1082.689
1140	1071.215
1170	1059.74
1200	1048.265
1230	1036.791
1260	1025.316
1290	1013.841

1320	1002.367
1350	990.8921
1380	979.4174
1410	967.9428
1440	956.4681
1470	944.9935
1500	933.5188
1530	922.0442
1560	910.5696
1590	899.0949
1620	887.6203
1650	876.1456
1680	864.671
1710	853.1964
1740	841.7217
1770	830.2471
1800	818.7724

Tables S17-27 indicate results of linear regression-based extrusion pressure predictions. The following parameters for outcome prediction are set constant to the following values and conditions unless the specific variable is varied to examine its effect on extrusion pressure prediction:

1. Physical crosslinking duration = 60 seconds
2. Photocrosslinking duration = 0 seconds
3. Inner nozzle outer diameter = 0  $\mu\text{m}$ ,
4. Outer nozzle inner diameter = 410  $\mu\text{m}$
5. Cell density =  $10^6$  cells/mL,
6. Syringe temperature = 22.5  $^{\circ}\text{C}$ .
7. Printing temperature = 22.5  $^{\circ}\text{C}$
8. Days observed = 0 days,
9. Solvent = cell culture medium only
10. Cell viability = 90%

**Table S17.** Effect of cell viability on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Cell Viability (%)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	261.4062
10	247.866
20	234.3258
30	220.7855
40	207.2453
50	193.7051
60	180.1649
70	166.6247
80	153.0845
90	139.5443
100	126.0041

**Table S18.** Effect of cell viability on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

<b>Cell Viability (%)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	277.0516
10	263.5114
20	249.9712
30	236.431
40	222.8908
50	209.3506
60	195.8104
70	182.2702
80	168.73
90	155.1898
100	141.6496

**Table S19.** Effect of cell viability on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 7% (w/v).

<b>Cell Viability (%)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	292.6971
10	279.1569
20	265.6166
30	252.0764
40	238.5362
50	224.996
60	211.4558
70	197.9156
80	184.3754
90	170.8352
100	157.295

**Table S20.** Effect of syringe temperature on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Syringe Temp (°C)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
4	183.2356
5	181.7196
6	180.2036
7	178.6876
8	177.1716
9	175.6557
10	174.1397
11	172.6237
12	171.1077
13	169.5917
14	168.0757
15	166.5597
16	165.0437
17	163.5277
18	162.0117
19	160.4957
20	158.9797
21	157.4637
22	155.9478
23	154.4318
24	152.9158
25	151.3998
26	149.8838
27	148.3678
28	146.8518
29	145.3358
30	143.8198
31	142.3038
32	140.7878
33	139.2718
34	137.7559
35	136.2399
36	134.7239
37	133.2079
38	131.6919
39	130.1759
40	128.6599



**Table S21.** Effect of cell density on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

<b>Cell Density (10<sup>6</sup> cells/mL)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0.5	155.9832
1	155.1898
1.5	154.3963
2	153.6028
2.5	152.8094
3	152.0159
3.5	151.2225
4	150.429
4.5	149.6355
5	148.8421
5.5	148.0486
6	147.2551
6.5	146.4617
7	145.6682
7.5	144.8748
8	144.0813
8.5	143.2878
9	142.4944
9.5	141.7009
10	140.9075
10.5	140.114
11	139.3205
11.5	138.5271
12	137.7336
12.5	136.9402
13	136.1467
13.5	135.3532
14	134.5598
14.5	133.7663
15	132.9728
15.5	132.1794
16	131.3859
16.5	130.5925
17	129.799
17.5	129.0055
18	128.2121
18.5	127.4186
19	126.6252
19.5	125.8317
20	125.0382

**Table S22.** Effect of substrate temperature on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

<b>Substrate Temperature (°C)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
4	216.8359
5	213.5037
6	210.1715
7	206.8392
8	203.507
9	200.1748
10	196.8426
11	193.5103
12	190.1781
13	186.8459
14	183.5137
15	180.1814
16	176.8492
17	173.517
18	170.1848
19	166.8525
20	163.5203
21	160.1881
22	156.8559
23	153.5236
24	150.1914
25	146.8592
26	143.527
27	140.1948
28	136.8625
29	133.5303
30	130.1981
31	126.8659
32	123.5336
33	120.2014
34	116.8692
35	113.537
36	110.2047
37	106.8725
38	103.5403
39	100.2081
40	96.87584

**Table S23.** Effect of alginate concentration on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 3% (w/v).

<b>Alginate Concentration (% w/v)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	149.9042
1	146.4509
2	142.9976
3	139.5443
4	136.091
5	132.6377
6	129.1845
7	125.7312
8	122.2779
9	118.8246
10	115.3713
11	111.918
12	108.4647
13	105.0115
14	101.5582
15	98.10488
16	94.6516
17	91.19831
18	87.74503
19	84.29174
20	80.83845

**Table S24.** Effect of alginate concentration on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 5% (w/v).

<b>Alginate Concentration (% w/v)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
1	165.5496
2	162.0963
3	158.643
4	155.1898
5	151.7365
6	148.2832
7	144.8299
8	141.3766
9	137.9233
10	134.47
11	131.0168
12	127.5635
13	124.1102
14	120.6569
15	117.2036
16	113.7503
17	110.297
18	106.8438
19	103.3905
20	99.93719

**Table S25.** Effect of gelatin concentration on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v).

<b>Gelatin Concentration (% w/v)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	116.0761
1	123.8989
2	131.7216
3	139.5443
4	147.367
5	155.1898
6	163.0125
7	170.8352
8	178.6579
9	186.4807
10	194.3034
11	202.1261
12	209.9488
13	217.7715
14	225.5943
15	233.417
16	241.2397
17	249.0624
18	256.8852
19	264.7079
20	272.5306

**Table S26.** Effect of gelatin concentration on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 5% (w/v).

<b>Gelatin Concentration (% w/v)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	109.1696
1	116.9923
2	124.815
3	132.6377
4	140.4605
5	148.2832
6	156.1059
7	163.9286
8	171.7514
9	179.5741
10	187.3968
11	195.2195
12	203.0423
13	210.865
14	218.6877
15	226.5104
16	234.3331
17	242.1559
18	249.9786
19	257.8013
20	265.624

**Table S27.** Effect of nozzle size on extrusion pressure predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

<b>Outer Nozzle Inner Diameter (<math>\mu\text{m}</math>)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	146.6149
100	148.7063
200	150.7977
300	152.8892
400	154.9806
500	157.0721
600	159.1635
700	161.2549
800	163.3464
900	165.4378
1000	167.5293

Tables S28-35 indicate results of random forest regression-based extrusion pressure predictions. The following parameters for outcome prediction are set constant to the following values and conditions unless the specific variable is varied to examine its effect on extrusion pressure prediction:

1. Physical crosslinking duration = 60 seconds
2. Photocrosslinking duration = 0 seconds
3. Inner nozzle outer diameter = 0  $\mu\text{m}$ ,
4. Outer nozzle inner diameter = 410  $\mu\text{m}$
5. Cell density =  $10^6$  cells/mL,
6. Syringe temperature = 22.5  $^{\circ}\text{C}$ .
7. Printing temperature = 22.5  $^{\circ}\text{C}$
8. Days observed = 0 days,
9. Solvent = cell culture medium only
10. Cell viability = 90%

**Table S28.** Effect of cell viability on extrusion pressure predictions through random forest regression modeling trained using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

Cell Viability (%)	Extrusion Pressure Predicted (kPa)
0	253.9683
10	253.9683
20	253.9683
30	253.9683
40	256.9041
50	233.0358
60	188.4042
70	63.83094
80	63.44139
90	68.82862
100	68.82862

**Table S29.** Effect of syringe temperature on extrusion pressure predictions through random forest regression modeling trained using the generalized dataset. Alginate and gelatin concentrations were set to a constant value of 3% (w/v).

<b>Syringe Temp (°C)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
4	72.06411
5	72.06411
6	72.06411
7	72.06411
8	68.82862
9	68.82862
10	68.82862
11	68.82862
12	68.82862
13	68.82862
14	68.82862
15	68.82862
16	68.82862
17	68.82862
18	68.82862
19	68.82862
20	68.82862
21	68.82862
22	68.82862
23	68.82862
24	68.82862
25	68.28807
26	68.28807
27	68.28807
28	68.28807
29	68.28807
30	68.28807
31	68.28807
32	68.28807
33	68.28807
34	68.28807
35	68.28807
36	68.28807
37	69.17102
38	69.17102
39	69.17102
40	69.17102



**Table S30.** Effect of substrate temperature on extrusion pressure predictions through random forest regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set a constant value of 5% (w/v).

<b>Substrate Temperature</b>	<b>Extrusion Pressure Predicted (kPa)</b>
4	217.715
5	217.715
6	217.715
7	217.715
8	220.0268
9	220.0268
10	220.0268
11	220.0268
12	220.0268
13	219.9848
14	219.9848
15	219.9848
16	219.9848
17	218.7992
18	216.9612
19	216.9612
20	122.1777
21	102.3297
22	102.3297
23	102.9544
24	102.8891
25	102.8891
26	102.8891
27	102.8891
28	102.8891
29	102.8891
30	102.8891
31	102.8891
32	102.8891
33	102.8891
34	102.8891
35	102.8891
36	102.8891
37	102.8891
38	102.8891
39	102.8891
40	102.8891

**Table S31.** Effect of cell density on extrusion pressure predictions through random forest regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set a constant value of 5% (w/v).

<b>Cell Density (10<sup>6</sup> cells/mL)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0.5	124.4639
1	102.3297
1.5	101.6715
2	101.1057
2.5	101.344
3	101.344
3.5	101.344
4	101.344
4.5	101.344
5	102.034
5.5	98.96097
6	98.05187
6.5	98.05187
7	98.05187
7.5	98.05187
8	98.05187
8.5	98.05187
9	98.05187
9.5	98.05187
10	98.05187
10.5	98.05187
11	98.05187
11.5	98.05187
12	98.05187
12.5	98.05187
13	98.05187
13.5	98.05187
14	98.05187
14.5	98.05187
15	98.05187
15.5	101.2649
16	101.2649
16.5	101.2649
17	101.2649
17.5	101.2649
18	122.8052
18.5	122.8052
19	122.8052
19.5	122.8052
20	122.8052

**Table S32.** Effect of gelatin concentration on extrusion pressure predictions through random forest regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v).

<b>Gelatin Concentration (%w/v)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	68.82862
1	68.82862
2	68.82862
3	68.82862
4	71.93072
5	102.3297
6	104.8197
7	104.8197
8	109.2507
9	109.2507
10	109.2507
11	109.2507
12	109.2507
13	111.6162
14	111.6162
15	111.6162
16	111.6162
17	111.6162
18	111.6162
19	111.6162
20	111.6162

**Table S33.** Effect of gelatin concentration on extrusion pressure predictions through random forest regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 5% (w/v).

<b>Gelatin Concentration (%w/v)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	68.82862
1	68.82862
2	68.82862
3	68.82862
4	71.93072
5	102.8997
6	105.3897
7	105.3897
8	110.5807
9	110.5807
10	110.5807
11	110.5807
12	110.5807
13	112.9462
14	112.9462
15	112.9462
16	112.9462
17	112.9462
18	112.9462
19	112.9462
20	112.9462

**Table S34.** Effect of alginate concentration on extrusion pressure predictions through random forest regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 3% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	70.49031
1	69.30904
2	67.35239
3	68.82862
4	68.82862
5	68.82862
6	68.82862
7	70.31568
8	70.31568
9	70.31568
10	70.31568
11	70.31568
12	70.31568
13	70.31568
14	70.31568
15	70.31568
16	70.31568
17	70.31568
18	70.31568
19	70.31568
20	70.31568

**Table S35.** Effect of alginate concentration on extrusion pressure predictions through random forest regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 5% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	103.5224
1	102.3412
2	100.9006
3	102.3297
4	102.8997
5	102.8997
6	102.8997
7	108.8321
8	108.8321
9	108.8321
10	108.8321
11	108.8321
12	108.8321
13	108.8321
14	108.8321
15	108.8321
16	108.8321
17	108.8321
18	108.8321
19	108.8321
20	108.8321

**Table S36.** Effect of nozzle size on extrusion pressure predictions through random forest regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set a constant value of 5% (w/v).

<b>Outer Nozzle Inner Diameter (μm)</b>	<b>Extrusion Pressure Predicted (kPa)</b>
0	87.70909
100	87.70909
200	86.75659
300	92.39011
400	101.9994
500	102.6386
600	102.6386
700	102.6386
800	102.6386
900	102.6386
1000	102.6386

Tables S37-42 indicate results of random forest regression-based cell viability predictions. The following parameters for outcome prediction are set constant to the following values and conditions unless the specific variable is varied to examine its effect on cell viability prediction:

1. Physical crosslinking duration = 60 seconds
2. Photocrosslinking duration = 0 seconds
3. Inner nozzle outer diameter = 0  $\mu\text{m}$ ,
4. Outer nozzle inner diameter = 410  $\mu\text{m}$
5. Cell density =  $10^6$  cells/mL,
6. Syringe temperature = 22.5  $^{\circ}\text{C}$ .
7. Printing temperature = 22.5  $^{\circ}\text{C}$
8. Days observed = 0 days,
9. Solvent = cell culture medium only
10. Extrusion pressure = 95.2 kPa
11.  $\text{CaCl}_2$  concentration = 100 mM

**Table S37.** Effect of alginate concentration on cell viability predictions through random forest regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 0% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Cell Viability (%)</b>
0	82.42586
1	81.59861
2	81.59861
3	81.59861
4	81.59861
5	81.59861
6	81.59861
7	81.59861
8	81.59861
9	81.59861
10	81.59861
11	81.59861
12	81.59861
13	81.59861
14	81.59861
15	81.59861
16	81.59861
17	81.59861
18	63.85928
19	63.85928
20	63.85928



**Table S38.** Effect of alginate concentration on cell viability predictions through random forest regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 3% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Cell Viability (%)</b>
0	82.42586
1	81.59861
2	81.59861
3	81.59861
4	81.59861
5	81.59861
6	81.59861
7	81.59861
8	81.59861
9	81.59861
10	81.59861
11	81.59861
12	81.59861
13	81.59861
14	81.59861
15	81.59861
16	81.59861
17	81.59861
18	63.85928
19	63.85928
20	63.85928

**Table S39.** Effect of alginate concentration on cell viability predictions through random forest regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 5% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Cell Viability (%)</b>
0	82.42586
1	81.59861
2	81.59861
3	81.59861
4	81.59861
5	81.59861
6	81.59861
7	81.59861
8	81.59861
9	81.59861
10	81.59861
11	81.59861
12	81.59861
13	81.59861
14	81.59861
15	81.59861
16	81.59861
17	81.59861
18	63.85928
19	63.85928
20	63.85928

**Table S40.** Effect of nozzle size on cell viability predictions through random forest regression modeling trained using the generalized dataset. Alginate concentration was set to 2% (w/v) and gelatin concentration was set to a constant value of 0% (w/v).

<b>Outer Nozzle Inner Diameter (µm)</b>	<b>Cell Viability (%)</b>
0	80.07774
100	80.07774
200	81.59861
300	81.59861
400	81.59861
500	81.59861
600	81.59861
700	81.59861
800	81.59861
900	81.59861
1000	81.59861

**Table S41.** Effect of standard nozzle sizes used in the EBB field on cell viability predictions through random forest regression modeling trained using the generalized dataset. Alginate concentration was set to 2% (w/v) and gelatin concentration was set to a constant value of 0% (w/v).

<b>Outer Nozzle Inner Diameter (<math>\mu\text{m}</math>)</b>	<b>Cell Viability (%)</b>
250	81.59861
260	81.59861
410	81.59861
413	81.59861
510	81.59861
610	81.59861
690	81.59861
840	81.59861
860	81.59861

**Table S42.** Effect of standard nozzle sizes used in the EBB field on cell viability predictions through random forest regression modeling trained using the generalized dataset. Alginate concentration was set to 2% (w/v) and gelatin concentration was set to a constant value of 3% (w/v).

<b>Extrusion Pressure (kPa)</b>	<b>Cell Viability (%)</b>
0	79.58217
10	79.58217
20	79.58217
30	79.58217
40	79.58217
50	79.58217
60	79.58217
70	79.58217
80	79.58217
90	79.58217
100	79.58217
110	79.58217
120	79.58217
130	79.58217
140	79.58217
150	80.61875
160	81.68637
170	81.68637
180	81.68637
190	81.68637
200	81.68637
210	81.68637
220	81.68637
230	81.68637
240	81.68637
250	81.68637
260	81.68637
270	81.68637
280	74.03073
290	74.03073
300	74.03073

It is clear from these approaches that random forest regression provides a very discretized way to predict values, which does not reflect actual cell prediction values.

Tables S43-60 indicate results of linear regression-based cell viability predictions. The following parameters for outcome prediction are set constant to the following values and conditions unless the specific variable is varied to examine its effect on extrusion pressure prediction:

1. Physical crosslinking duration = 60 seconds
2. Photocrosslinking duration = 0 seconds
3. Inner nozzle outer diameter = 0  $\mu\text{m}$ ,
4. Outer nozzle inner diameter = 410  $\mu\text{m}$
5. Cell density =  $10^6$  cells/mL,
6. Syringe temperature = 22.5  $^{\circ}\text{C}$ .
7. Printing temperature = 22.5  $^{\circ}\text{C}$
8. Days observed = 0 days,
9. Solvent = cell culture medium only
10. Extrusion pressure = 95.2 kPa
11.  $\text{CaCl}_2$  concentration = 100 mM

**Table S43.** Effect of alginate concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 0% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Cell Viability (%)</b>
0	73.55403
1	73.23163
2	72.90922
3	72.58682
4	72.26441
5	71.942
6	71.6196
7	71.29719
8	70.97479
9	70.65238
10	70.32998
11	70.00757
12	69.68517
13	69.36276
14	69.04036
15	68.71795
16	68.39555
17	68.07314
18	67.75074
19	67.42833
20	67.10593

**Table S44.** Effect of alginate concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 3% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Cell Viability (%)</b>
0	74.20701
1	73.88461
2	73.5622
3	73.2398
4	72.91739
5	72.59499
6	72.27258
7	71.95018
8	71.62777
9	71.30537
10	70.98296
11	70.66056
12	70.33815
13	70.01575
14	69.69334
15	69.37094
16	69.04853
17	68.72613
18	68.40372
19	68.08132
20	67.75891

**Table S45.** Effect of alginate concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 5% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Cell Viability (%)</b>
0	74.64234
1	74.31993
2	73.99753
3	73.67512
4	73.35272
5	73.03031
6	72.70791
7	72.3855
8	72.0631
9	71.74069
10	71.41829
11	71.09588
12	70.77347
13	70.45107
14	70.12866
15	69.80626
16	69.48385
17	69.16145
18	68.83904
19	68.51664
20	68.19423

**Table S46.** Effect of alginate concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Gelatin concentration was set to a constant value of 7% (w/v).

<b>Alginate Concentration (%w/v)</b>	<b>Cell Viability (%)</b>
0	75.07766
1	74.75526
2	74.43285
3	74.11044
4	73.78804
5	73.46563
6	73.14323
7	72.82082
8	72.49842
9	72.17601
10	71.85361
11	71.5312
12	71.2088
13	70.88639
14	70.56399
15	70.24158
16	69.91918
17	69.59677
18	69.27437
19	68.95196
20	68.62956



**Table S47.** Effect of nozzle size on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 2% (w/v) and gelatin concentration was set to a constant value of 0% (w/v).

<b>Outer Nozzle Inner Diameter (<math>\mu\text{m}</math>)</b>	<b>Cell Viability (%)</b>
0	75.64393
100	74.97693
200	74.30992
300	73.64292
400	72.97592
500	72.30892
600	71.64192
700	70.97491
800	70.30791
900	69.64091
1000	68.97391

**Table S48.** Effect of nozzle size on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 2% (w/v) and gelatin concentration was set to a constant value of 3% (w/v).

<b>Outer Nozzle Inner Diameter (<math>\mu\text{m}</math>)</b>	<b>Cell Viability (%)</b>
0	76.29691
100	75.62991
200	74.96291
300	74.29591
400	73.6289
500	72.9619
600	72.2949
700	71.6279
800	70.9609
900	70.29389
1000	69.62689

**Table S49.** Effect of nozzle size on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 2% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

<b>Outer Nozzle Inner Diameter (<math>\mu\text{m}</math>)</b>	<b>Cell Viability (%)</b>
0	76.73224
100	76.06523
200	75.39823
300	74.73123
400	74.06423
500	73.39723
600	72.73022
700	72.06322
800	71.39622
900	70.72922
1000	70.06222

**Table S50.** Effect of syringe temperature on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

Syringe Temp (°C)	Cell Viability (%)
4	77.42513
5	77.22242
6	77.01972
7	76.81702
8	76.61432
9	76.41161
10	76.20891
11	76.00621
12	75.8035
13	75.6008
14	75.3981
15	75.19539
16	74.99269
17	74.78999
18	74.58729
19	74.38458
20	74.18188
21	73.97918
22	73.77647
23	73.57377
24	73.37107
25	73.16836
26	72.96566
27	72.76296
28	72.56026
29	72.35755
30	72.15485
31	71.95215
32	71.74944
33	71.54674
34	71.34404
35	71.14133
36	70.93863
37	70.73593
38	70.53323
39	70.33052
40	70.12782

**Table S51.** Effect of syringe temperature on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 7% (w/v).

Syringe Temperature (°C)	Cell Viability (%)
4	77.86045
5	77.65775
6	77.45504
7	77.25234
8	77.04964
9	76.84694
10	76.64423
11	76.44153
12	76.23883
13	76.03612
14	75.83342
15	75.63072
16	75.42801
17	75.22531
18	75.02261
19	74.81991
20	74.6172
21	74.4145
22	74.2118
23	74.00909
24	73.80639
25	73.60369
26	73.40098
27	73.19828
28	72.99558
29	72.79288
30	72.59017
31	72.38747
32	72.18477
33	71.98206
34	71.77936
35	71.57666
36	71.37395
37	71.17125
38	70.96855
39	70.76585
40	70.56314

**Table S52.** Effect of substrate temperature on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

<b>Substrate Temperature (°C)</b>	<b>Cell Viability (%)</b>
4	86.93854
5	86.2216
6	85.50466
7	84.78771
8	84.07077
9	83.35383
10	82.63689
11	81.91995
12	81.20301
13	80.48607
14	79.76912
15	79.05218
16	78.33524
17	77.6183
18	76.90136
19	76.18442
20	75.46748
21	74.75053
22	74.03359
23	73.31665
24	72.59971
25	71.88277
26	71.16583
27	70.44889
28	69.73194
29	69.015
30	68.29806
31	67.58112
32	66.86418
33	66.14724
34	65.4303
35	64.71335
36	63.99641
37	63.27947
38	62.56253
39	61.84559
40	61.12865

**Table S53.** Effect of substrate temperature on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 7% (w/v).

<b>Substrate Temperature (°C)</b>	<b>Cell Viability (%)</b>
4	87.37386
5	86.65692
6	85.93998
7	85.22304
8	84.5061
9	83.78915
10	83.07221
11	82.35527
12	81.63833
13	80.92139
14	80.20445
15	79.48751
16	78.77056
17	78.05362
18	77.33668
19	76.61974
20	75.9028
21	75.18586
22	74.46892
23	73.75197
24	73.03503
25	72.31809
26	71.60115
27	70.88421
28	70.16727
29	69.45033
30	68.73338
31	68.01644
32	67.2995
33	66.58256
34	65.86562
35	65.14868
36	64.43174
37	63.71479
38	62.99785
39	62.28091
40	61.56397

**Table S54.** Effect of CaCl<sub>2</sub> concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 3% (w/v).

CaCl <sub>2</sub> Concentration (mM)	Cell Viability (%)
0	73.53324
100	73.2398
200	72.94636
300	72.65293
400	72.35949
500	72.06605
600	71.77262
700	71.47918
800	71.18574
900	70.89231
1000	70.59887

**Table S55.** Effect of CaCl<sub>2</sub> concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

CaCl <sub>2</sub> Concentration (mM)	Cell Viability (%)
0	73.96856
100	73.67512
200	73.38169
300	73.08825
400	72.79481
500	72.50138
600	72.20794
700	71.9145
800	71.62107
900	71.32763
1000	71.0342

**Table S56.** Effect of CaCl<sub>2</sub> concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 7% (w/v).

<b>CaCl<sub>2</sub> Concentration (mM)</b>	<b>Cell Viability (%)</b>
0	74.40388
100	74.11044
200	73.81701
300	73.52357
400	73.23014
500	72.9367
600	72.64326
700	72.34983
800	72.05639
900	71.76295
1000	71.46952

**Table S57.** Effect of physical crosslinking duration on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

<b>Physical Crosslinking Duration (s)</b>	<b>Cell Viability (%)</b>
30	73.68288
60	73.67512
90	73.66736
120	73.6596
150	73.65185
180	73.64409
210	73.63633
240	73.62857
270	73.62081
300	73.61305
330	73.60529
360	73.59753
390	73.58977
420	73.58202
450	73.57426
480	73.5665
510	73.55874
540	73.55098
570	73.54322
600	73.53546
630	73.5277
660	73.51994
690	73.51219
720	73.50443



<b>Physical Crosslinking Duration (s)</b>	<b>Cell Viability (%)</b>
750	73.49667
780	73.48891
810	73.48115
840	73.47339
870	73.46563
900	73.45787
930	73.45011
960	73.44235
990	73.4346
1020	73.42684
1050	73.41908
1080	73.41132
1110	73.40356
1140	73.3958
1170	73.38804
1200	73.38028
1230	73.37252
1260	73.36477
1290	73.35701
1320	73.34925
1350	73.34149
1380	73.33373
1410	73.32597
1440	73.31821
1470	73.31045
1500	73.30269
1530	73.29494
1560	73.28718
1590	73.27942
1620	73.27166
1650	73.2639
1680	73.25614
1710	73.24838
1740	73.24062
1770	73.23286
1800	73.22511

**Table S58.** Effect of extrusion pressure on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set to a constant value of 5% (w/v).

<b>Extrusion Pressure (kPa)</b>	<b>Cell Viability (%)</b>
0	76.86298
5	76.66374
10	76.4645
15	76.26526
20	76.06602
25	75.86678
30	75.66754
35	75.46829
40	75.26905
45	75.06981
50	74.87057
55	74.67133
60	74.47209
65	74.27285
70	74.0736
75	73.87436
80	73.67512
85	73.47588
90	73.27664
95	73.0774
100	72.87816
105	72.67892
110	72.47967
115	72.28043
120	72.08119
125	71.88195
130	71.68271
135	71.48347
140	71.28423
145	71.08498
150	70.88574
155	70.6865
160	70.48726
165	70.28802
170	70.08878
175	69.88954
180	69.6903
185	69.49105
190	69.29181
195	69.09257
200	68.89333
205	68.69409
210	68.49485

<b>Extrusion Pressure (kPa)</b>	<b>Cell Viability (%)</b>
215	68.29561
220	68.09636
225	67.89712
230	67.69788
235	67.49864
240	67.2994
245	67.10016
250	66.90092
255	66.70167
260	66.50243
265	66.30319
270	66.10395
275	65.90471
280	65.70547
285	65.50623
290	65.30699
295	65.10774
300	64.9085

**Table S59.** Effect of gelatin concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 0% (w/v).

<b>Increasing gelatin concentration saw increased cell viability.</b>	
<b>Gelatin Concentration (% w/v)</b>	<b>Cell Viability (%)</b>
0	73.55403
1	73.77169
2	73.98935
3	74.20701
4	74.42468
5	74.64234
6	74.86
7	75.07766
8	75.29532
9	75.51298
10	75.73064
11	75.94831
12	76.16597
13	76.38363
14	76.60129
15	76.81895
16	77.03661
17	77.25427
18	77.47194
19	77.6896
20	77.90726

**Table S60.** Effect of gelatin concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v).

<b>Gelatin Concentration (% w/v)</b>	<b>Cell Viability (%)</b>
0	72.58682
1	72.80448
2	73.02214
3	73.2398
4	73.45746
5	73.67512
6	73.89278
7	74.11044
8	74.32811
9	74.54577
10	74.76343
11	74.98109
12	75.19875
13	75.41641
14	75.63407
15	75.85174
16	76.0694
17	76.28706
18	76.50472
19	76.72238
20	76.94004

**Table S61.** Effect of gelatin concentration on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 5% (w/v).

<b>Gelatin Concentration (% w/v)</b>	<b>Cell Viability (%)</b>
0	71.942
1	72.15967
2	72.37733
3	72.59499
4	72.81265
5	73.03031
6	73.24797
7	73.46563
8	73.6833
9	73.90096
10	74.11862
11	74.33628
12	74.55394
13	74.7716
14	74.98926
15	75.20693
16	75.42459
17	75.64225
18	75.85991
19	76.07757
20	76.29523

**Table S62.** Effect of cell density on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set a constant value of 3% (w/v).

<b>Cell Density (<math>10^6/\text{mL}</math>)</b>	<b>Cell Viability (%)</b>
0.5	72.74688
1	72.87102
1.5	72.99516
2	73.1193
2.5	73.24344
3	73.36758
3.5	73.49172
4	73.61586
4.5	73.74
5	73.86414
5.5	73.98828
6	74.11242
6.5	74.23657
7	74.36071
7.5	74.48485
8	74.60899
8.5	74.73313
9	74.85727
9.5	74.98141
10	75.10555

**Table S63.** Effect of cell density on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set a constant value of 5% (w/v).

<b>Cell Density (<math>10^6</math>/mL)</b>	<b>Cell Viability (%)</b>
0.5	73.21686
1	73.341
1.5	73.46514
2	73.58928
2.5	73.71342
3	73.83756
3.5	73.9617
4	74.08584
4.5	74.20998
5	74.33412
5.5	74.45826
6	74.5824
6.5	74.70654
7	74.83068
7.5	74.95482
8	75.07896
8.5	75.2031
9	75.32724
9.5	75.45138
10	75.57552

**Table S64.** Effect of cell density on cell viability predictions through linear regression modeling trained using the generalized dataset. Alginate concentration was set to a constant value of 3% (w/v) and gelatin concentration was set a constant value of 7% (w/v).

<b>Cell Density (<math>10^6/\text{mL}</math>)</b>	<b>Cell Viability (%)</b>
0.5	73.68684
1	73.81098
1.5	73.93512
2	74.05926
2.5	74.1834
3	74.30754
3.5	74.43168
4	74.55582
4.5	74.67996
5	74.8041
5.5	74.92824
6	75.05238
6.5	75.17652
7	75.30066
7.5	75.4248
8	75.54894
8.5	75.67308
9	75.79722
9.5	75.92136
10	76.0455



# Appendix B: Python code used for ML model training, evaluation, and prediction of cell viability and extrusion pressure

## Importing Packages and Functions

```
import pandas as pd
import numpy as np
from numpy import mean
import matplotlib as mp
import matplotlib.pyplot as plt
import os
from sklearn.compose import make_column_transformer
from sklearn.preprocessing import MinMaxScaler, OneHotEncoder
from sklearn.ensemble import RandomForestRegressor, RandomForestClassifier
from sklearn.svm import SVC, SVR
from sklearn.metrics import accuracy_score, precision_score, recall_score,
confusion_matrix, classification_report, f1_score, mean_absolute_error,
mean_squared_error, roc_auc_score, plot_confusion_matrix
from sklearn.impute import SimpleImputer, KNNImputer
from sklearn.model_selection import train_test_split, KFold, LeaveOneOut,
cross_validate, cross_val_score, GridSearchCV, RepeatedStratifiedKFold
from sklearn.linear_model import LinearRegression, LogisticRegression,
Ridge, Lasso, SGDRegressor, SGDClassifier #LogisticRegression is a
classification model
from sklearn.neighbors import KNeighborsRegressor, KNeighborsClassifier
from sklearn import tree
from sklearn.tree import export_graphviz, DecisionTreeClassifier,
DecisionTreeRegressor
from sklearn.pipeline import Pipeline

%matplotlib inline
```

## Loading Dataset

Load the dataset for analysis and training in the code below. Change the file path if needed

```
bioprint_df = pd.read_csv('C:/Users/Shuyu/Desktop/20201229 Bioink
Database/20210406/Final Database/20210429/Classification and Regression
Database (617 instances) 20210429.csv') #change the .csv file path to the
current location of the dataset
```

```
#Setting references column in bioprint_df as the row indices
bioprint_df = bioprint_df.set_index(bioprint_df['Reference'])
bioprint_df = bioprint_df.drop(['Reference'], axis = 1)
```

Print the first 5 instances of data as well as general dataset array information and how many blank values there are per variable

```
#bioprint_df.head(5)
#bioprint_df.shape
bioprint_df.isna().sum()
```

## Data Preprocessing and analysis

### Imputting mode temperatures

```
imputer_mode = SimpleImputer(missing_values = np.nan, strategy =
'most_frequent') #imputing mode value into missing values for temperatures
bioprint_df.loc[:,['Syringe_Temperature_(°C)', 'Substrate_Temperature_(°C)']
] =
imputer_mode.fit_transform(bioprint_df.loc[:,['Syringe_Temperature_(°C)', 'S
ubstrate_Temperature_(°C)']])
```

### Analyzing Numerical (Continuous) Data

#### Dropping Variables and Instances

```
#Drop certain material concentration as no concentration values exist in
papers
```

```
bioprint_df = bioprint_df.drop(['Fiber_Diameter_(µm)'], axis = 1) #drop for
extrusion pressure dataset creation
```

```
bioprint_df =
```

```
bioprint_df.drop(['CaCl2_Conc_(mM)', 'NaCl2_Conc_(mM)', 'BaCl2_Conc_(mM)', 'Sr
Cl2_Conc_(mM)', 'Physical_Crosslinking_Durantion_(s)', 'Photocrosslinking_Dur
ation_(s)'], axis = 1) #drop these variables to create the extrusion
pressure dataset from the cell viability dataset
```

```
#Variables where more than 50% of all instances have null values are
dropped
```

```
#This amounts to variables with 309 or more null instances for the cell
viability dataset with 617 instances
```

```
#bioprint_df = bioprint_df.dropna(axis = 1, thresh=309)
```

```
#Variables where more than 50% of all instances have null values are
dropped
```

```
#This amounts to variables with 177 or more null instances #Extrusion
Pressure with 354 instances
```

```
bioprint_df = bioprint_df.dropna(axis = 1, thresh=177)
```

```
#Drop redundant variables for Mondal Intrastudy dataset creation
```

```
#bioprint_df =
```

```
bioprint_df.drop(['Cell_culture_medium_used?', 'DI_water_used?', 'Precrosslin
king_solution_used?', 'Saline_solution_used?', 'EtOH_solution_used?', 'Photoin
itiator_used?', 'Enzymatic_Crosslinker_used?', 'Matrigel_used?', 'Conical_or_S
traight_None', 'Primary/Not_Primary'], axis = 1)
```

```
#Drop instances without cell viability values
```

```

bioprint_df =
bioprint_df[bioprint_df['Viability_at_time_of_observation_(%)'].notna()]

#Drop nonprinting instances (instances where extrusion pressure is zero)
bioprint_df = bioprint_df.drop(bioprint_df[bioprint_df['Extrusion_Pressure
(kPa)'] == 0 ].index)
bioprint_df = bioprint_df[bioprint_df['Extrusion_Pressure (kPa)'].notna()]
#used to create extrusion pressure dataset

    bioprint_df.head(10)

bioprint_df.shape
#bioprint_df.isna().sum()

    (354, 45)

```

## Feature Selection Through Correlation

This does not require imputing null values yet

```

corr = bioprint_df.corr()

display(corr)

fig, ax = plt.subplots(figsize = (20, 16))
sns.heatmap(corr, xticklabels = corr.columns, yticklabels = corr.columns,
linewidths=0.1)

abs(bioprint_df.corr()["Viability_at_time_of_observation_(%)"])

```

Create the independent variables (x) set and the dependent variable (y) set from the training dataset. The cell viability or other variable name for y need to be change to the full variable name in the dataset used.

```

#Drop for cell viability, extrusion pressure, and Intrastudy dataset
creation
bioprint_df =
bioprint_df.drop(['Final_PEGTA_Conc_(%w/v)', 'Final_PEGMA_Conc_(%w/v)'],
axis = 1)

```

## Imputing Values

```

    bioprint_df.isna().sum() #produces a list of each variable's number of
null values =

```

## Imputation of numerical/continuous values databases

```

imputer_knn = KNNImputer(n_neighbors = 30, weights = "uniform") #imputing
mode value into missing values

#bioprint_df.iloc[:,0:28] =
imputer_knn.fit_transform(bioprint_df.iloc[:,0:28]) #used for cell
viability dataset preprocessing

```

```
bioprint_df.iloc[:,0:22] =  
imputer_knn.fit_transform(bioprint_df.iloc[:,0:22]) #used for extrusion  
pressure dataset preprocessing
```

## Imputation of categorical values in databases

Missing categorical data is replaced with the most common value

```
bioprint_df =  
bioprint_df.fillna(bioprint_df['Conical_or_Straight_Nozzle'].value_counts()  
.index[0])
```

```
#bioprint_df.shape
```

```
bioprint_df.isna().sum() #check if the imputation code works by generating  
a list of the number of null values for each variable
```

```
#Drop categorical or numerical cell viability column depending on which  
type of prediction model is desired (regression versus classification)
```

```
bioprint_df = bioprint_df.drop(['Viability_at_time_of_observation_(%)'],  
axis = 1)
```

```
#bioprint_df = bioprint_df.drop(['Acceptable_Viability_(Yes/No)'], axis =  
1)
```

```
bioprint_df = bioprint_df.drop(['Extrusion_Pressure (kPa)'], axis = 1)
```

```
#bioprint_df = bioprint_df.drop(['Acceptable_Pressure_(Yes/No)'], axis = 1)
```

## Normalizing/Scalarizing and Encoding Continuous and Categorical Data

```
#x = bioprint_df.drop("Viability_at_time_of_observation_(%)", axis = 1)  
#y = bioprint_df["Viability_at_time_of_observation_(%)"].values
```

```
#x = bioprint_df.drop("Acceptable_Viability_(Yes/No)", axis = 1)  
#y = bioprint_df["Acceptable_Viability_(Yes/No)"].values
```

```
#x = bioprint_df.drop("Acceptable_Viability_(Y/N)", axis = 1)  
#y = bioprint_df["Acceptable_Viability_(Y/N)"].values
```

```
#x = bioprint_df.drop("Extrusion_Pressure (kPa)", axis = 1)  
#y = bioprint_df["Extrusion_Pressure (kPa)"].values
```

```
x = bioprint_df.drop("Acceptable_Pressure_(Yes/No)", axis = 1)  
y = bioprint_df["Acceptable_Pressure_(Yes/No)"].values
```

```
#Use MinMaxScaler() function to normalize input values for performance  
metric evaluation. DO NOT USE for value prediction for cell viability and  
extrusion pressure
```

```
#x.iloc[:,0:28] = MinMaxScaler().fit_transform(x.iloc[:,0:28]) # Used for  
cell viability generalized dataset
```

```

#x.iloc[:,0:29] = MinMaxScaler().fit_transform(x.iloc[:,0:29]) #Used for
intrastudy dataset
x.iloc[:,0:22] = MinMaxScaler().fit_transform(x.iloc[:,0:22]) #Used for
extrusion pressure dataset
#y = y.reshape(-1,1) #used for extracting mean squared error
#y = MinMaxScaler().fit_transform(y) #used for extracting mean squared
error

#x = column_trans.fit_transform(x)

x_ohencoded = pd.get_dummies(x, columns
=['Cell_Culture_Medium_Used?', 'DI_Water_Used?', 'Precrosslinking_Solution_Us
ed?', 'Saline_Solution_Used?', 'EtOH_Solution_Used?', 'Photoinitiator_Used?', '
Enzymatic_Crosslinker_Used?', 'Matrigel_Used?', 'Conical_or_Straight_Nozzle',
'Primary/Not_Primary']) #one-hot encoding is used to encode
binary/categorical data in datasets

#x_ohencoded.head()
#x.head()
#x.shape

#x.isna().sum()
y_ohencoded = pd.get_dummies(y)
y_ohencoded.isna().sum()

x_ohencoded.shape
#x_ohencoded.isna().sum()

#x_ohencoded.to_csv('C:/Users/Shuyu/Desktop/export_dataframe.csv', index =
False, header=True) #exports a .csv dataset file with one hot encoded
variables

```

## Machine Learning Algorithms for Regression Modeling

### 1. Random Forest Regressor

```
def rfr_model_optimization(x, y):
```

```
# Perform Grid-Search to find the optimal hyperparameters of random forest
regression models
```

```

    gsc = GridSearchCV(
        estimator=RandomForestRegressor(random_state=42),
        param_grid={
            'max_depth': range(3,7),
            'n_estimators': (10, 50, 100),
        },
        cv=10, scoring='r2', verbose=0,
n_jobs=-1) #verbose controls how many messages are returned

```

```

grid_result = gsc.fit(x, y)
best_params = grid_result.best_params_

#rfr
=DecisionTreeRegressor(max_depth=best_params["max_depth"],random_state=42)
rfr =RandomForestRegressor(max_depth=best_params["max_depth"],
n_estimators=best_params["n_estimators"], random_state=42, verbose=False)
# Perform K-Fold CV
scores = cross_val_score(rfr, x, y, cv=10, scoring='r2')
return best_params,scores

rfr_model(x_ohencoded,y)

y = np.ravel(y)
#x_train, x_test, y_train, y_test =
x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42)

#x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.1,
random_state = 42) #Used for intrastudy dataset since there are no
categorical variables in the intrastudy dataset to one hot encode

#rfr = RandomForestRegressor(max_depth=3,random_state = 42,
n_estimators=100) #Use for intrastudy dataset
#rfr = RandomForestRegressor(max_depth=5,random_state = 42,
n_estimators=10) #Use for cell viability dataset
rfr = RandomForestRegressor(max_depth=6,random_state = 42, n_estimators=10)
#Use for extrusion pressure dataset

rfr.fit(x_train,y_train)

pred_rfr = rfr.predict(x_test) #runs label prediction on the test set
rfr_score = rfr.score(x_tets, y_test) #returns the coefficient of
determination of the model

#aur = roc_auc_score(y_test,pred_rfr)
#mae = mean_absolute_error(y_test,pred_rfr)
#mse = mean_squared_error(y_test,pred_rfr)
#print(mae)
#print(mse)
print(rfr_score) #coefficient of determination scoring

# Used to create random forest based decision tree
#plt.rcParams['figure.figsize'] = [20,10]
#plt.rcParams['font.size']= 10

```

```
#fig, axes = plt.subplots(nrows = 1,ncols = 1,figsize = (10,10), dpi=1000)
#_ = tree.plot_tree(rfr.estimators_[9],feature_names = x_ohencoded.columns,
class_names = ['Cell viability (%)'], filled=True, fontsize = 12)
```

```
# Used to create random forest based feature importance ranking graph
```

```
features = x_train.columns
importances = rfr.feature_importances_
indices = np.argsort(importances)
```

```
# customized number of the most important features
```

```
num_features = 10
```

```
#plt.figure(figsize=(10,10))
```

```
#plt.title('Random Forest Regression Feature Importances')
```

```
# only plot the customized number of features
```

```
#Plots a bar graph of the relative feature importance values of the most
importance features
```

```
plt.barh(range(num_features), importances[indices[-num_features:]],
color='b', align='center')
```

```
plt.yticks(range(num_features), [features[i] for i in indices[-
num_features:]])
```

```
plt.xlabel('Relative Importance')
```

```
plt.xlim(0,0.6)
```

```
plt.show()
```

```
#Calculates for coefficient of determination (r2) and mean squared error
values based on the number of cross-validation folds
```

```
def rfr_model():
```

```
    #model = RandomForestRegressor(max_depth = 3, random_state = 42,
n_estimators=100)
```

```
    #model = RandomForestRegressor(max_depth = 5, random_state = 42,
n_estimators=10) #cell viability 617 instances
```

```
    model = RandomForestRegressor(max_depth = 6, random_state = 42,
n_estimators=10) #extrusion pressure 354 instances
```

```
    return model #model already defined
```

```
def rfr_model_performance(cv): #cv is the cross-validation type ex: 10
fold, loocv, stratified, etc
```

```
    model = rfr_model()
```

```
    # evaluate the model, scoring can change from 'r2' to
'neg_mean_squared_error'
```

```
    scores = cross_val_score(model, x_ohencoded, y, scoring='r2', cv=cv,
n_jobs=-1)
```

```

    # return scores
    return mean(scores), scores.min(), scores.max()

# define folds to test
rfr_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

# evaluate each k value
for k in rfr_folds:
    # define the folding configuration to test
    cv = KFold(n_splits=k, shuffle=True, random_state=42)
    #cv is the # of folds

    # evaluate k value
    k_mean, k_min, k_max = rfr_model_performance(cv) # report performance
    print('> folds=%d, r2=%.3f (%.3f,%.3f)' % (k, k_mean, k_min, k_max))
    # store mean score
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(rfr_folds, means, yerr=[mins, maxs], fmt='o', markersize = 5,
color = 'black', linewidth = 3)
# plot the ideal case in a separate color
#plt.plot(rfr_folds, [ideal for _ in range(len(rfr_folds))], color='r',
Label = 'Ideal Mean Squared Error')
plt.title("Number of Cross Validation Folds vs R2", fontsize = 20)
plt.xlabel('Folds tested on', fontsize = 20)
plt.ylabel('R2', fontsize = 20)
#plt.rcParams.update({'font.size': 10})
#plt.legend(loc='upper left')
plt.rcParams["figure.figsize"] = (10,7)
# show the plot
plt.show()

```

## 2. Linear Regression

```

x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42)
#x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.1,
random_state = 42) #Used for intrastudy dataset

lr = LinearRegression()
lr.fit(x_train,y_train)

```



```

pred_lr = lr.predict(x_test) #runs label prediction on the test set
lr_score = lr.score(x_test,y_test) #returns the coefficient of
determination of the model
#aur = roc_auc_score(y_test,pred_rfr)
#mae = mean_absolute_error(y_test,pred_rfr)
#mse = mean_squared_error(y_test,pred_rfr)
#print(mae)
#print(mse)
print(lr_score) #prints the coefficient of determination of the model

def lr_model():
    model = LinearRegression()
    return model

def lr_model_performance(cv): #crossval is the cross-validation type ex: 10
fold, loocv, stratified, etc
    model = lr_model()
    # evaluate the model, scoring can change from 'r2' to
'neg_mean_squared_error'
    scores = cross_val_score(model, x_ohencoded, y, scoring='r2', cv=cv,
n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()

# define folds to test
lr_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

# evaluate each k value
for k in lr_folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=42) cv is the number
of folds
    # evaluate k value
    k_mean, k_min, k_max = lr_model_performance(cv)
    # report performance
    print('> folds=%d, mse=%.3f (%.3f,%.3f)' % (k, k_mean, k_min, k_max))
    # store mean score
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(lr_folds, means, yerr=[mins, maxs], fmt='o', markersize = 5,
color = 'black', linewidth = 3)
# plot the ideal case in a separate color
plt.plot(lr_folds, [ideal for _ in range(len(lr_folds))], color='b', label

```

```

= 'Ideal Mean Squared Error')
plt.title("Folds vs R2")
plt.xlabel('Folds tested on')
plt.ylabel('R2')
#plt.legend()
plt.rcParams.update({'font.size': 10})
# show the plot
plt.show()

```

### 3. Support Vector Regression

```

x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42)
#x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.1,
random_state = 42) #Used for intrastudy dataset

```

```

svr = SVR(kernel = 'rbf')
svr.fit(x_train,y_train)

```

```

pred_svr = svr.predict(x_test) #runs label prediction on the test set
svr_score = svr.score(x_test,y_test) #returns the coefficient of
determination of the model
#aur = roc_auc_score(y_test,pred_svr)
#mae = mean_absolute_error(y_test,pred_svr)
#mse = mean_squared_error(y_test,pred_svr)
#print(mae)
#print(mse)
print(svr_score) #prints the coefficient of determination of the model

```

```

def svr_model():
    model = SVR(kernel = 'rbf')
    return model #model already defined

```

```

def svr_model_performance(cv): #crossval is the cross-validation type ex:
10 fold, loocv, stratified, etc
    model = svr_model()
    # evaluate the model, scoring can change from 'r2' to
'neg_mean_squared_error'
    scores = cross_val_score(model, x_ohencoded, y, scoring='r2', cv=cv,
n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()

```

```

# define folds to test
#svr_folds = [2,5,10]

```

```

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

```

```

# evaluate each k value
for k in svr_folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=42) # cv is the
number of folds

    # evaluate k value
    k_mean, k_min, k_max = svr_model_performance(cv)
    # report performance
    print('> folds=%d, r2=%.3f (%.3f,%.3f)' % (k, k_mean, k_min, k_max))
    # store mean score
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(svr_folds, means, yerr=[mins, maxs], fmt='o', markersize = 5,
color = 'black', linewidth = 3)
# plot the ideal case in a separate color
plt.plot(svr_folds, [ideal for _ in range(len(svr_folds))], color='b',
label = 'Ideal accuracy')
plt.title("R2 vs Folds tested on")
plt.xlabel('Folds tested on')
plt.ylabel('R2')
#plt.legend()
plt.rcParams.update({'font.size': 10})
# show the plot
plt.show()

```

## Classification Models

### 1. Random Forest Classifier

```

def rfc_model(x, y):

# Perform Grid-Search to find the optimal hyperparameters of a random
forest classification model
    gsc = GridSearchCV(
        estimator=RandomForestClassifier(random_state=42),
        param_grid={
            'max_depth': range(3,7),
            'n_estimators': (10, 50, 100, 1000),
        },
        cv=3, scoring='accuracy', verbose=0,
n_jobs=-1) #verbose controls how many messages are returned

    grid_result = gsc.fit(x, y)
    best_params = grid_result.best_params_

```

```

    rfc = RandomForestClassifier(max_depth=best_params["max_depth"],
n_estimators=best_params["n_estimators"], random_state=42, verbose=False)
# Perform K-Fold cross validation
    scores = cross_val_score(rfc, x, y, cv=10, scoring='accuracy')
    #plt.rcParams['figure.figsize'] = [30,20]
    #plt.rcParams['font.size']= 20
    #_ = tree.plot_tree(rfr.estimators_[9],feature_names =
x.columns,filled=True) # plots a decision tree from the random forest model

    return best_params,scores

rfc_model(x_ohencoded,y_ohencoded)
#rfc_model(x,y) #Used for intrastudy dataset

x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42)
#x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.1,
random_state = 42) #Used for intrastudy dataset

#rfc = RandomForestClassifier(max_depth = 3, random_state = 42,
n_estimators=100) #Used for cell viability dataset
#rfc = RandomForestClassifier(max_depth = 3, random_state = 42,
n_estimators=10) #Used for intrastudy dataset
rfc = RandomForestClassifier(max_depth = 6, random_state = 42,
n_estimators=50) #Used for extrusion pressure dataset

rfc.fit(x_train,y_train)

pred_rfc = rfc.predict(x_test)
rfc_score = rfc.score(x_test,y_test)
print(rfc_score)

#Confusion matrix generation from trained model
disp = plot_confusion_matrix(rfc, x_test, y_test,

display_labels=['Unacceptable_CV','Acceptable_CV'],
                cmap=plt.cm.Blues)
print(disp.confusion_matrix)

#plt.rcParams['figure.figsize'] = [30,20]
#plt.rcParams['font.size']= 20
#_ = tree.plot_tree(rfc.estimators_[9],feature_names = x.columns,
class_names = y, filled=True, fontsize = 12)

features = x_train.columns
importances = rfc.feature_importances_

```

```

indices = np.argsort(importances)

# customized number of the most important features
num_features = 10

#plt.figure(figsize=(10,100))
#plt.title('Random Forest Classification Feature Importances')
# only plot the customized number of features

#Plots a bar graph of the relative feature importance values of the most
importance features
plt.barh(range(num_features), importances[indices[-num_features:]],
color='b', align='center')
plt.yticks(range(num_features), [features[i] for i in indices[-
num_features:]])
plt.xlabel('Relative Importance')
plt.xlim(0, 0.6)
plt.show()

def rfc_model():
    #model = RandomForestClassifier(max_depth = 3, random_state = 42,
n_estimators=50) #Intrastudy dataset
    #model = RandomForestClassifier(max_depth = 3, random_state = 42,
n_estimators=100) #617 cell viability instances
    model = RandomForestClassifier(max_depth = 6, random_state = 42,
n_estimators=100) #354 extrusion pressure instances
    return model #model already defined

#scoring = ['accuracy', 'precision', 'recall']
def rfc_model_performance(cv): #crossval is the cross-validation type ex:
10 fold, loocv, stratified, etc
    # get the model
    model = rfc_model()
    #scoring = 'accuracy', 'f1', 'precision', 'recall', 'roc_auc'
    # evaluate the model
    scores = cross_val_score(model, x_ohencoded, y, scoring='recall',
cv=cv, n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()

# define folds to test
#rfc_folds = range(2,11)
rfc_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

# evaluate each k value
for k in rfc_folds:
    # define the test condition

```

```

    cv = KFold(n_splits=k, shuffle=True, random_state=42) # cv is the
number of folds
    # evaluate k value
    k_mean, k_min, k_max = evaluate_rfc_model(cv)

    # report performance
    print('> folds=%d, recall=%.3f (%.3f,%.3f)' % (k, k_mean, k_min,
k_max))
    # store mean score
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(rfc_folds, means, yerr=[mins, maxs], fmt='o', markersize = 5,
color = 'black', linewidth = 3)
# plot the ideal case in a separate color
plt.plot(rfc_folds, [ideal for _ in range(len(rfc_folds))], color='b',
label = 'Ideal accuracy')
plt.title("Number of Cross Validation Folds vs Recall")
plt.xlabel('Folds tested on')
plt.ylabel('Recall')
plt.legend(loc='lower left')
plt.rcParams.update({'font.size': 10})
# show the plot
plt.show()

scoring = {'acc': 'accuracy',
           'prec_macro': 'precision_macro',
           'rec_micro': 'recall_macro',
           'f1': 'f1'}
scores = cross_validate(get_rfc_model(), x, y, scoring=scoring,
                        cv=10, return_train_score=True)

print(scores.keys())
print(scores['test_acc'])

#Displays a decision tree from the random forest classification model

#rfc = RandomForestClassifier(max_depth = 3, n_estimators = 100,
random_state=42, verbose=False)
#plt.rcParams['figure.figsize'] = [30,20]
#plt.rcParams['font.size']= 20
#_ = tree.plot_tree(rfr.estimators_[9],feature_names =
x.columns,filled=True)

```

## 2. Logistic Regression

```

x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42)

```

```

#x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.1,
random_state = 42)
#x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.3,
random_state = 42) #Mondal

logr = LogisticRegression()
logr.fit(x_train,y_train)

#Confusion matrix generation from trained model

#pred_logr = logr.predict(x_test)
#logr_score = logr.score(x_test,y_test)
#print(classification_report(y_test,pred_Logr))
#print(confusion_matrix(y_test,pred_Logr))

disp = plot_confusion_matrix(logr, x_test, y_test,
display_labels=['Unacceptable_CV','Acceptable_CV'],
cmap=plt.cm.Blues)
#disp.ax_.set_title(title)

#print(title)
print(disp.confusion_matrix)

plt.show()

def lr_model():
    model = LogisticRegression(solver='liblinear')
    return model

def lr_model_performance(cv): #crossval is the cross-validation type ex: 10
fold, loocv, stratified, etc
    model = lr_model()

    #scoring = 'accuracy', 'f1', 'precision', 'recall', 'roc_auc'
    # evaluate the model
    scores = cross_val_score(model, x_ohencoded, y, scoring='recall',
cv=cv, n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()

# define folds to test
lr_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

# evaluate each k value
for k in lr_folds:
    # define the test condition

```

```

cv = KFold(n_splits=k, shuffle=True, random_state=42) # cv is the
number of folds
# evaluate k value
k_mean, k_min, k_max = lr_model_performance(cv)

# report performance
print('> folds=%d, recall=%.3f (%.3f,%.3f)' % (k, k_mean, k_min,
k_max))
# store mean score
means.append(k_mean)
# store min and max relative to the mean
mins.append(k_mean - k_min)
maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(lr_folds, means, yerr=[mins, maxs], fmt='o', markersize = 5,
color = 'black', linewidth = 3)
# plot the ideal case in a separate color
plt.plot(lr_folds, [ideal for _ in range(len(lr_folds))], color='b', label
= 'Ideal accuracy')
plt.title("Number of Cross Validation Folds vs Recall")
plt.xlabel('Folds tested on', fontsize = 15)
plt.ylabel('Recall', fontsize = 15)
plt.legend(loc = "upper left")
#plt.rcParams.update({'font.size': 20})
# show the plot
plt.show()

```

### 3. Support Vector Classification

```

x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42)
#x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.3,
random_state = 42) #Mondal
svc = SVC(kernel = 'rbf')
svc.fit(x_train,y_train)

```

```

pred_svc = svc.predict(x_test) #runs Label prediction on the test set
svc_score = svc.score(x_test,y_test) #returns coefficient of determination
of the model
print(svc_score)
#print(classification_report(y_test,pred_svc))
print(confusion_matrix(y_test,pred_svc)) #displays the confusion matrix of
the model

```

```

def svc_model():
    model = SVC(kernel = 'rbf')
    return model #model already defined

```



```

def svc_model_performance(cv): #crossval is the cross-validation type ex:
10 fold, loocv, stratified, etc
    model = svc_model()
    #scoring = 'accuracy', 'f1', 'precision', 'recall', 'roc_auc'

    # evaluate the model
    svc_scores = cross_val_score(model, x_ohencoded, y, scoring='recall',
cv=cv, n_jobs=-1)
    # return scores
    return mean(svc_scores), svc_scores.min(), svc_scores.max()

# define folds to test
svc_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

# evaluate each k value
for k in svc_folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=42) # cv is the
number of folds

    # evaluate k value
    k_mean, k_min, k_max = svc_model_performance(cv)

    # report performance
    print('> folds=%d, recall=%.3f (%.3f,%.3f)' % (k, k_mean, k_min,
k_max))
    # store mean score
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# line plot of k mean values with min/max error bars
plt.errorbar(svc_folds, means, yerr=[mins, maxs], fmt='o', markersize = 5,
color = 'black', linewidth = 3)
# plot the ideal case in a separate color
plt.plot(svc_folds, [ideal for _ in range(len(svc_folds))], color='b',
label = 'Ideal accuracy')
plt.title("Number of Cross Validation Folds vs Recall")
plt.xlabel('Folds tested on')
plt.ylabel('Recall')
plt.legend(loc = "upper left")
plt.rcParams.update({'font.size': 10})
#plt.rcParams["figure.figsize"] = (15,10)
# show the plot
plt.show()

```

## Generating Value Predictions

```
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.1,
random_state = 42)
#rfc.fit(x_train,y_train)
#pred_rfc = rfc.predict(x_test)

print(classification_report(y_test,pred_rfc))

#The prediction .csv file can be set to any variable combinations that uses
the same variables

predict_df = pd.read_csv('C:/Users/Shuyu/Desktop/20201229 Bioink
Database/20210406/Final Database/Cell Viability Prediction Set.csv') #This
is the predicting dataset
#predict_df = pd.read_csv('C:/Users/Shuyu/Desktop/20201229 Bioink
Database/20210406/Final Database/Mondal Intrastudy Dataset Test Set.csv')
#This is the predicting set
#predict_df = pd.read_csv('C:/Users/Shuyu/Desktop/20201229 Bioink
Database/20210406/Final Database/Extrusion Pressure Prediction Set.csv')
#This is the predicting set
#predict_df = pd.read_csv('C:/Users/Shuyu/Desktop/20201229 Bioink
Database/20210406/Final Database/Extrusion Pressure Prediction Set No
Viability No Sub Temp.csv')

#Dropping the predicted variable from the prediction dataset

#predict_df = predict_df.drop(['Viability_at_time_of_observation_(%)'],
axis = 1)
#predict_df =
predict_df.drop(['Acceptable_Viability_(N)', 'Acceptable_Viability_(Y)'],
axis = 1)
predict_df = predict_df.drop(['Acceptable_Viability_(Y/N)'], axis = 1)
#predict_df = predict_df.drop(['Condition', 'Acceptable_Viability_(Y/N)'],
axis = 1) #Mondal
#predict_df =
predict_df.drop(['Condition', 'Viability_at_time_of_observation_(%)'], axis
= 1) #Mondal

#train the desired model used for predictions

#rfr.fit(x_ohencoded,y)
#rfc.fit(x_ohencoded,y)
#lr.fit(x_ohencoded,y)
logr.fit(x_ohencoded,y)
#svr.fit(x_ohencoded,y)
#svc.fit(x_ohencoded,y)

#rfr.fit(x,y) #for Intrastudy dataset
```

```

#lr.fit(x,y) #for Intrastudy dataset
#svr.fit(x,y) #for Intrastudy dataset

#rfc.fit(x,y) #for Intrastudy dataset
#logr.fit(x,y) #for Intrastudy dataset
#svc.fit(x,y) #for Intrastudy dataset

#xnew returns the independent variables of the output .csv with predicted
values

#xnew = predict_df.drop(['Acceptable_Viability_(Y/N)'], axis = 1)
xnew = predict_df.drop(['Viability_at_time_of_observation_(%)'],axis = 1)
#xnew = predict_df.drop(['Extrusion_Pressure (kPa)'],axis = 1)

#xnew =
predict_df.drop(['Acceptable_Viability_(N)', 'Acceptable_Viability_(Y)'],axi
s = 1)
#xnew.head()

#ynew returns the predicted output variable of the output .csv

#ynew = rfr.predict(xnew)
#ynew = rfc.predict(xnew)
#ynew = lr.predict(xnew)
ynew = logr.predict(xnew)
#ynew = svr.predict(xnew)
#ynew = svc.predict(xnew)
#xnew_ohencoded.head()

xnew['Acceptable_Viability_(Y/N)'] = ynew #Appends the output variable onto
the .csv file with prediction values
#xnew['Viability_at_time_of_observation_(%)']=ynew
#xnew['Extrusion_Pressure (kPa)']=ynew

export_df = pd.DataFrame(xnew) #creates the dataframe with corresponding
input and predicted variable values
export_df.to_csv(r'C:/Users/Shuyu/Desktop/export_dataframe.csv', index =
False, header=True) #exports the dataframe as a .csv file to a location on
the computer

```

# Appendix C: Python code used for ML model training, evaluation, and prediction of filament diameter

## Importing Packages and Functions

```
import pandas as pd
import numpy as np
from numpy import mean
import matplotlib as mpl
import matplotlib.pyplot as plt
import os
from sklearn.compose import make_column_transformer
from sklearn.preprocessing import MinMaxScaler, OneHotEncoder
from sklearn.ensemble import RandomForestRegressor, RandomForestClassifier
from sklearn.svm import SVC, SVR
from sklearn import svm, tree
from sklearn.metrics import accuracy_score, precision_score, recall_score,
confusion_matrix, classification_report, f1_score, mean_absolute_error,
mean_squared_error, roc_auc_score, plot_confusion_matrix
from sklearn.impute import SimpleImputer, KNNImputer
from sklearn.model_selection import train_test_split, KFold, LeaveOneOut,
cross_validate, cross_val_score, GridSearchCV
from sklearn.linear_model import LinearRegression, LogisticRegression
#LogisticRegression is a classification model
from sklearn import tree

%matplotlib inline
```

## Loading Dataset

Load the dataset for analysis and training in the code below. Change the file path if needed

```
bioprint_df = pd.read_csv('C:/Users/Shuyu/Desktop/20201229 Bioink
Database/20210406/Final Database/20210429/Filament Diameter with Existing
Instances (340 instances) 20210503.csv') #This is the training dataset. The
.csv file path can be changed to the current location of the dataset

#Setting references column as the row indices
bioprint_df = bioprint_df.set_index(bioprint_df['Reference'])
bioprint_df = bioprint_df.drop(['Reference'], axis = 1)
```

Print the first 5 instances of data as well as general dataset array information and how many blank values there are per variable

```
#bioprint_df.head(5)
#bioprint_df.info()
bioprint_df.isna().sum()
```

## Data Preprocessing and analysis

## Imputing mode temperatures

```
imputer_mode = SimpleImputer(missing_values = np.nan, strategy =  
'most_frequent') #imputing mode value into missing values  
bioprint_df.loc[:,['Syringe_Temperature_(°C)', 'Substrate_Temperature_(°C)']  
] =  
imputer_mode.fit_transform(bioprint_df.loc[:,['Syringe_Temperature_(°C)', 'S  
ubstrate_Temperature_(°C)']])
```

## Analyzing Numerical (Continuous) Data

### Dropping Variables and Instances

```
#Variables where more than 50% of all instances have null values are  
dropped
```

```
#This amounts to variables with 170 or more null instances
```

```
bioprint_df = bioprint_df.dropna(axis = 1, thresh=170)
```

```
bioprint_df.head(5)
```

```
bioprint_df.shape
```

### Imputing Values

```
bioprint_df.isna().sum()
```

### Imputing of numerical/continuous values in the database

```
imputer_knn = KNNImputer(n_neighbors = 30, weights = "uniform") #imputing  
mode value into missing values
```

```
bioprint_df.iloc[:,0:30] =
```

```
imputer_knn.fit_transform(bioprint_df.iloc[:,0:30])
```

### Imputing of categorical values in the database

Missing categorical data is replaced with the most common value

```
bioprint_df =
```

```
bioprint_df.fillna(bioprint_df['Conical_or_Straight_Nozzle'].value_counts()  
.index[0])
```

```
bioprint_df.head(10)
```

```
bioprint_df.isna().sum()
```

```
bioprint_df = bioprint_df.drop("Filament_Diameter_(µm)", axis = 1)
```

```
#bioprint_df = bioprint_df.drop("Acceptable_Filament_Diameter_(Yes/No)",  
axis = 1)
```

## Normalizing/Scalarizing and Encoding Continuous and Categorical Data

```

#x = bioprint_df.drop("Filament_Diameter_(µm)", axis = 1)
#y = bioprint_df["Filament_Diameter_(µm)"].values

x = bioprint_df.drop("Acceptable_Filament_Diameter_(Yes/No)", axis = 1)
y = bioprint_df["Acceptable_Filament_Diameter_(Yes/No)"].values

#Use MinMaxScaler() function to normalize input values for performance
metric evaluation. DO NOT USE for value prediction for filament diameter
#x.iloc[:,0:31] = MinMaxScaler().fit_transform(x.iloc[:,0:31]) #249
instances
#x.iloc[:,0:29] = MinMaxScaler().fit_transform(x.iloc[:,0:29]) #339
instances
#y = y.reshape(-1,1)
#y = MinMaxScaler().fit_transform(y)

x_ohencoded = pd.get_dummies(x, columns
=['Cell_Culture_Medium_Used?', 'DI_Water_Used?', 'Precrosslinking_Solution_Us
ed?', 'Saline_Solution_Used?', 'EtOH_Solution_Used?', 'Photoinitiator_Used?', '
Enzymatic_Crosslinker_Used?', 'Conical_or_Straight_Nozzle'])

#x_ohencoded.head(5)

#x_ohencoded.isna().sum()
#y_ohencoded = pd.get_dummies(y)

print(x)

```

## Normalizing/Scalarizing and Encoding Categorical Data

Our nominal data can be binary encoded through one hot encoding and continuous data can be scalarized

### 1. Random Forest Regressor

```
def rfr_model_optimization(x, y):
```

```
# Perform Grid-Search to find the optimal hyperparameters of random forest
regression models
```

```

    gsc = GridSearchCV(
        estimator=RandomForestRegressor(random_state=42),
        param_grid={
            'max_depth': range(3,7),
            'n_estimators': (10, 50, 100, 1000),
        },
        cv=10, scoring=r2', verbose=0,
n_jobs=-1) #verbose controls how many messages are returned

```

```

    grid_result = gsc.fit(x, y)
    best_params = grid_result.best_params_

```

```

    rfr =RandomForestRegressor(max_depth=best_params["max_depth"],
n_estimators=best_params["n_estimators"], random_state=42, verbose=False)

```

```

# Perform K-Fold CV
scores = cross_val_score(rfr, x, y, cv=10, scoring='r2')

return best_params,scores

rfr_model_optimization(x_ohencoded,y)

x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.1,
random_state = 42)
rfr = RandomForestRegressor(max_depth=3,random_state = 42, n_estimators=10)
#340 instances
rfr.fit(x_train,y_train)

pred_rfr = rfr.predict(x_test) #runs Label prediction on the test set
rfr_score = rfr.score(x_test,y_test) #returns the coefficient of
determination of the model
#aur = roc_auc_score(y_test,pred_rfr)

#mae = mean_absolute_error(y_test,pred_rfr)
#mse = mean_squared_error(y_test,pred_rfr)
#print(mae)
#print(mse)
print(rfr_score)

# Displays a decision tree used in the random forest regression model
#plt.rcParams['figure.figsize'] = [20,10]
#plt.rcParams['font.size']= 10
#fig, axes = plt.subplots(nrows = 1,ncols = 1,figsize = (10,10), dpi=1000)
#_ = tree.plot_tree(rfr.estimators_[9],feature_names = x_ohencoded.columns,
class_names = ['Filament_Diameter_(μm)'], filled=True, fontsize = 13.5)

# Used to create random forest based feature importance ranking graph

features = x_train.columns
importances = rfr.feature_importances_
indices = np.argsort(importances)

# customized number of the most important features
num_features = 10

#plt.figure(figsize=(10,100))
#plt.title('Random Forest Regression Feature Importances')
# only plot the customized number of features

#Plots a bar graph of the relative feature importance values of the most
importance features
plt.barh(range(num_features), importances[indices[-num_features:]],
color='b', align='center')
plt.yticks(range(num_features), [features[i] for i in indices[-

```

```

num_features:]]
plt.xlabel('Relative Importance')
plt.xlim(0,0.6)
plt.show()

def rfr_model():
    model = RandomForestRegressor(max_depth = 3, random_state = 42,
n_estimators=10)
    return model #model already defined

def rfr_model_performance(cv): #cv is the cross-validation type ex: 10
fold, loocv, stratified, etc
    model = get_rfr_model()
    # evaluate the model, scoring can change from 'r2' to 'mse'
    scores = cross_val_score(model, x_ohencoded, y, scoring='r2', cv=cv,
n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()

# define folds to test
rfr_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

# evaluate each k value
for k in rfr_folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=42)
    # evaluate k value
    k_mean, k_min, k_max = rfr_model_performance(cv) #cv is the number of
folds
    # report performance
    print('> folds=%d, r2=%.3f (%.3f,%.3f)' % (k, k_mean, k_min, k_max))
    # store mean score
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(rfr_folds, means, yerr=[mins, maxs], fmt='o', markersize = 5,
color = 'black', linewidth = 3)
# plot the ideal case in a separate color
#plt.plot(rfr_folds, [ideal for _ in range(len(rfr_folds))], color='r',
Label = 'LeaveOneOut Mean Squared Error')
plt.title("Number of Cross Validation Folds vs R2")
plt.xlabel('Folds tested on')
plt.ylabel('R2')
#plt.rcParams.update({'font.size': 10})

```



```
plt.rcParams["figure.figsize"] = (10,7)
#plt.legend()
# show the plot
plt.show()
```

## 2. Linear Regression

```
x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42)
lr = LinearRegression()
lr.fit(x_train,y_train)
```

```
pred_lr = lr.predict(x_test) #runs label prediction on the test set
lr_score = lr.score(x_test,y_test) #returns the coefficient of
determination of the model
#aur = roc_auc_score(y_test,pred_rfr)
```

```
#mae = mean_absolute_error(y_test,pred_lr)
#mse = mean_squared_error(y_test,pred_lr)
#print(mae)
#print(mse)
```

```
print(lr_score) #returns coefficient of determination (r2)
```

```
def lr_model():
    model = LinearRegression()
    return model
```

```
def lr_model_performance(cv): #crossval is the cross-validation type ex: 10
fold, loocv, stratified, etc
    model = lr_model()
    # evaluate the model, scoring can change from 'r2' to
'neg_mean_squared_error'
    scores = cross_val_score(model, x_ohencoded, y, scoring='r2', cv=cv,
n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()
```

```
# define folds to test
lr_folds = [2,5,10]
```

```
# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()
```

```
# evaluate each k value
```

```
for k in lr_folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=42) #cv is the number
of folds
    # evaluate k value
    k_mean, k_min, k_max = lr_model_performance(cv)
```

```

# report performance
print('> folds=%d, r2=%.3f (%.3f,%.3f)' % (k, k_mean, k_min, k_max))
# store mean score
means.append(k_mean)
# store min and max relative to the mean
mins.append(k_mean - k_min)
maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(lr_folds, means, yerr=[mins, maxs], fmt='o', markersize = 5,
color = 'black', linewidth = 3)
# plot the ideal case in a separate color
#plt.plot(lr_folds, [ideal for _ in range(len(lr_folds))], color='r', label
= 'Ideal accuracy')
plt.title("Number of Cross Validation Folds vs R2")
plt.xlabel('Folds tested on')
plt.ylabel('R2')
#plt.legend()
plt.rcParams["figure.figsize"] = (10,7)
# show the plot
plt.show()

```

### 3. Support Vector Regression

```

x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 1)
svr = SVR(kernel='poly')
svr.fit(x_train,y_train)

```

```

pred_svr = svr.predict(x_test) #runs label prediction on the test set
svr_score = svr.score(x_test,y_test) #returns the coefficient of
determination of the model
#aur = roc_auc_score(y_test,pred_rfr)

```

```

#mae = mean_absolute_error(y_test,pred_lr)
#mse = mean_squared_error(y_test,pred_lr)
#print(mae)
#print(mse)
print(svr_score) #coefficient of determination scoring

```

```

def svr_model():
    model = SVR(kernel='rbf')
    return model #model already defined

```

```

def evaluate_svr_model(cv): #crossval is the cross-validation type ex: 10
fold, loocv, stratified, etc
    model = svr_model()
    # evaluate the model
    scores = cross_val_score(model, x_ohencoded, y,

```

```

scoring='neg_mean_squared_error', cv=cv, n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()

# define folds to test
svr_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

# evaluate each k value
for k in svr_folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=42) #cv is the number
of folds

    # evaluate k value
    k_mean, k_min, k_max = svr_model_performance(cv)
    # report performance
    print('> folds=%d, mse=%.3f (%.3f,%.3f)' % (k, k_mean, k_min, k_max))
    # store mean accuracy
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(svr_folds, means, yerr=[mins, maxs], fmt='o', markersize = 5,
color = 'black', linewidth = 3)
# plot the ideal case in a separate color
#plt.plot(svr_folds, [ideal for _ in range(len(svr_folds))], color='r',
Label = 'Ideal accuracy')
plt.title("Number of Cross Validation Folds vs Mean Squared Error")
plt.xlabel('Folds tested on')
plt.ylabel('Mean Squared Error')
#plt.legend()
# show the plot
plt.rcParams["figure.figsize"] = (10,7)
plt.show()

```

## Classification Models

### 1. Random Forest Classifier

```

def rfc_model(x, y):

# Perform Grid-Search to find the optimal hyperparameters of a random
forest classification model
    gsc = GridSearchCV(

```

```

    estimator=RandomForestClassifier(random_state=42),
    param_grid={
        'max_depth': range(3,7),
        'n_estimators': (10, 50, 100, 1000),
    },
    cv=10, scoring='accuracy', verbose=0,
n_jobs=-1) #verbose controls how many messages are returned

    grid_result = gsc.fit(x, y)
    best_params = grid_result.best_params_

    rfr = RandomForestClassifier(max_depth=best_params["max_depth"],
n_estimators=best_params["n_estimators"], random_state=42, verbose=False)
# Perform K-Fold CV
    scores = cross_val_score(rfr, x, y, cv=10, scoring='accuracy')
    return best_params,scores

rfc_model(x_ohencoded,y)

x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42)
rfc = RandomForestClassifier(max_depth = 6, random_state = 42,
n_estimators=10) #339 instances
rfc.fit(x_train,y_train)

pred_rfc = rfc.predict(x_test)
rfc_score = rfc.score(x_test,y_test)
print(classification_report(y_test,pred_rfc))
print(confusion_matrix(y_test,pred_rfc))
#aur = roc_auc_score(y_test,pred_rfr)

#mae = mean_absolute_error(y_test,pred_rfr)
#mse = mean_squared_error(y_test,pred_rfr)
##print(mae)
#print(mse)
#print(rfr_score)

#Displays a decision tree used in the random forest classification model

#plt.rcParams['figure.figsize'] = [30,20]
#plt.rcParams['font.size']= 20
#_ = tree.plot_tree(rfc.estimators_[9],feature_names = x_ohencoded.columns,
class_names = y, filled=True, fontsize = 12)

importances = rfc.feature_importances_
std = np.std([rfc.feature_importances_ for tree in rfc.estimators_],
axis=0)
indices = np.argsort(importances)[::-1]

```

```

# Print the feature ranking
features = x_train.columns
importances = rfc.feature_importances_
indices = np.argsort(importances)

# customized number of the most important features

num_features = 10

#plt.title('Random Forest Classification Feature Importances')
# only plot the customized number of features

#Plots a bar graph of the relative feature importance values of the most
importance features
plt.barh(range(num_features), importances[indices[-num_features:]],
color='b', align='center')
plt.yticks(range(num_features), [features[i] for i in indices[-
num_features:]])
plt.xlabel('Relative Importance')
plt.xlim(0,0.6)
plt.show()

def rfc_model():
    model = RandomForestClassifier(max_depth = 6, random_state = 42,
n_estimators=10)
    return model #model already defined

def rfc_model_performance(cv): #crossval is the cross-validation type ex:
10 fold, loocv, stratified, etc
    model = rfc_model()
    # evaluate the model
#scoring = ['accuracy', 'precision', 'recall']
    scores = cross_val_score(model, x_ohencoded, y, scoring='accuracy',
cv=cv, n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()

# define folds to test
rfc_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

# evaluate each k value
for k in rfc_folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=42) # cv is the

```

```

number of folds
    # evaluate k value
    k_mean, k_min, k_max = evaluate_rfc_model(cv)
    # report performance
    print('> folds=%d, accuracy=%.3f (%.3f,%.3f)' % (k, k_mean, k_min,
k_max))
    # store mean accuracy
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(rfc_folds, means, yerr=[mins, maxs], fmt='o', markersize = 20,
color = 'black', linewidth = 10)
# plot the ideal case in a separate color
plt.plot(rfc_folds, [ideal for _ in range(len(rfc_folds))], color='r',
label = 'Ideal accuracy')
plt.title("Fold vs Accuracy")
plt.xlabel('Fold tested on')
plt.ylabel('Accuracy score')
plt.legend(loc='upper left')
plt.rcParams["figure.figsize"] = (20,20)
# show the plot
plt.show()

scoring = {'acc': 'accuracy',
           'prec_macro': 'precision_macro',
           'rec_micro': 'recall_macro',
           'f1': 'f1'}
scores = cross_validate(get_rfc_model(), x, y, scoring=scoring,
                        cv=10, return_train_score=True)

print(scores.keys())
print(scores['test_acc'])

```

## 2. Logistic Regression

```

x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42)

```

```

logr = LogisticRegression()
logr.fit(x_train,y_train)

```

```

pred_logr = logr.predict(x_test)
logr_score = logr.score(x_test,y_test)
print(classification_report(y_test,pred_logr))
print(confusion_matrix(y_test,pred_logr))
#aur = roc_auc_score(y_test,pred_rfr)

```

```

def lr_model():
    model = LogisticRegression()
    return model

def lr_model_performance(cv): #crossval is the cross-validation type ex: 10
fold, loocv, stratified, etc
    model = lr_model()
    # evaluate the model

#scoring = ['accuracy', 'precision', 'recall']
    scores = cross_val_score(model, x_ohencoded, y, scoring='recall',
cv=cv, n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()

# define folds to test
lr_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()

# evaluate each k value
for k in lr_folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=42) #cv is the number
of folds
    # evaluate k value
    k_mean, k_min, k_max = lr_model_performance(cv)
    # report performance
    print('> folds=%d, recall=%.3f (%.3f,%.3f)' % (k, k_mean, k_min,
k_max))
    # store mean accuracy
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(lr_folds, means, yerr=[mins, maxs], fmt='o',color = 'black')
# plot the ideal case in a separate color
plt.plot(lr_folds, [ideal for _ in range(len(lr_folds))], color='r', label
= 'Ideal accuracy')
plt.title("Fold vs Recall")
plt.xlabel('Fold tested on')
plt.ylabel('Accuracy score')
plt.legend(loc = "upper left")
plt.rcParams["figure.figsize"] = (15,10)
# show the plot
plt.show()

```

### 3. Support Vector Classification

```
x_train, x_test, y_train, y_test = train_test_split(x_ohencoded,y,test_size
= 0.1, random_state = 42, shuffle='false')
svc = SVC(kernel = 'rbf')
svc.fit(x_train,y_train)

pred_svc = svc.predict(x_test)
svc_score = svc.score(x_test,y_test)
print(classification_report(y_test,pred_svc))
print(confusion_matrix(y_test,pred_svc))
#aur = roc_auc_score(y_test,pred_rfr)

# Plots a confusion matrix of support vector classifier performance

disp = plot_confusion_matrix(svc, x_test, y_test,
                             display_labels=['FD out of tolerance','FD
within tolerance'],
                             cmap=plt.cm.Blues)
#disp.ax_.set_title(title)

print(title)
print(disp.confusion_matrix)

plt.show()

def svc_model():
    model = SVC(kernel = 'rbf')
    return model #model already defined

def svc_model_performance(cv): #crossval is the cross-validation type ex:
10 fold, loocv, stratified, etc
    model = svc_model()
    # evaluate the model

#scoring = ['accuracy','precision', 'recall']
    scores = cross_val_score(model, x_ohencoded, y, scoring='recall',
cv=cv, n_jobs=-1)
    # return scores
    return mean(scores), scores.min(), scores.max()

# calculate the ideal test condition
ideal, _, _ = evaluate_svc_model(LeaveOneOut())
print('Ideal: %.3f' % ideal)

# define folds to test
svc_folds = [2,5,10]

# record mean and min/max of each set of results
means, mins, maxs = list(),list(),list()
```



```

# evaluate each k value
for k in svc_folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=42) #cv is the number
of folds
    # evaluate k value
    k_mean, k_min, k_max = evaluate_svc_model(cv)
    # report performance
    print('> folds=%d, recall=%.3f (%.3f,%.3f)' % (k, k_mean, k_min,
k_max))
    # store mean score
    means.append(k_mean)
    # store min and max relative to the mean
    mins.append(k_mean - k_min)
    maxs.append(k_max - k_mean)

# Line plot of k mean values with min/max error bars
plt.errorbar(svc_folds, means, yerr=[mins, maxs], fmt='o',color = 'black')
# plot the ideal case in a separate color
plt.plot(svc_folds, [ideal for _ in range(len(svc_folds))], color='r',
label = 'Ideal accuracy')
plt.title("Fold vs Accuracy")
plt.xlabel('Fold tested on')
plt.ylabel('Accuracy score')
plt.legend(loc = "upper left")
plt.rcParams["figure.figsize"] = (15,10)
# show the plot
plt.show()

```

## Generating Value Predictions

*#The prediction .csv file can be set to any variable combinations that uses the same variables*

```

predict_df = pd.read_csv('C:/Users/Shuyu/Desktop/20201229 Bioink
Database/20210406/Final Database/Filament Diameter Prediction Set 340
Instances.csv') #This is the predicting set

```

```

#rfr.fit(x_ohencoded,y)
#Lr.fit(x_ohencoded,y)
#svr.fit(x_ohencoded,y)

```

```

#rfc.fit(x_ohencoded,y)
#Logr.fit(x_ohencoded,y)
svc.fit(x_ohencoded,y)

```

```

#Dropping filament diameter variable from the prediction dataset
x = bioprint_df.drop("Acceptable_Filament_Diameter_(Yes/No)", axis = 1)
y = bioprint_df["Acceptable_Filament_Diameter_(Yes/No)"].values

```

*#xnew returns the independent variables of the output .csv with predicted values*

```
xnew = predict_df.drop(['Filament_Diameter_(µm)'],axis = 1)
#xnew = predict_df.drop(['Acceptable_Filament_Diameter_(Yes/No)'], axis =1)
```

*#ynew returns the predicted output variable of the output .csv*

```
#ynew = rfr.predict(xnew)
#ynew = lr.predict(xnew)
#ynew = svr.predict(xnew)
```

```
#ynew = rfc.predict(xnew)
#ynew = Logr.predict(xnew)
ynew = svc.predict(xnew)
#xnew_ohencoded.head()
```

```
xnew['Filament_Diameter_(µm)'] = ynew
```

```
export_df = pd.DataFrame(xnew) #creates the dataframe with corresponding input and predicted variable values
```

```
export_df.to_csv(r'C:/Users/Shuyu/Desktop/export_dataframe_FD.csv', index = False, header=True) #exports the dataframe as a .csv file to a location on the computer
```