

**Physics-Guided Neural Network
for the Prediction of Mass Change from Water Diffusion**

A Thesis Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Michael B. May
May 2025

UNCLASSIFIED

This document has been reviewed by a DC/RO and has been determined to be UNCLASSIFIED, not UCN, and contains no CUI based on current classification guidance. This review does not constitute a review for CUI outside of classification guidance and does not constitute clearance for Public Release.

Name: Vincent Lamberti

Date Reviewed: 4/10/2025

CNS eDC/RO ID: 932026

COPYRIGHT NOTICE

This document has been authored by Consolidated Nuclear Security, LLC, under Contract DE-NA-0001942 with the U.S. Department of Energy/National Nuclear Security Administration, or a subcontractor thereof. The United States Government retains and the publisher, by accepting the document for publication, acknowledges that the United States Government retains a nonexclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this document, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, or allow others to do so, for United States Government purposes.

DISCLAIMER

The information or opinions expressed herein by the author are solely those of the author. Neither the U.S. Government, Consolidated Nuclear Security, LLC, nor any of their employees warrant, or assume any legal liability for, the accuracy, completeness or usefulness of any information contained herein.

ACKNOWLEDGEMENTS

Thank you to my advisor from the Nuclear Engineering Department Dr. Jamie Coble and my mentor from the Y-12 National Security Complex Dr. Vincent Lamberti for their guidance in the making of this thesis. Thank you also to Dr. Derick Weis and Dr. Russell Hallman Jr. from Y-12 for their expertise and experiment. Lastly, I would like to thank my parents and family for their constant support during my graduate studies.

ABSTRACT

Machine learning (ML) processes such as neural networks (NN) have strong capabilities in the prediction of the state of a given problem domain. A drawback of NNs is their inability to generalize predictions outside of the domain of data for which they are trained, a problem exasperated by sparse data sets. For physical processes, this can be remedied by the use of physics-guided neural networks (PGNN). By using physical equations describing the process, approximate outputs of a future state of the process can be generated for inputs outside the domain of measured data as well as optimize coefficients within the equation(s) to match the data via gradient descent.

The PGNN was applied to two polyurethane foam samples contained within a glovebox with controlled pressure, temperature, and varied humidity levels, with the mass of the samples finely measured to 0.01 mg precision over time. The mass of the samples obeys the physics of Fickian water diffusion from the air, which were utilized in the PGNN. Combining the physics approximation with data measured from the glovebox, an algorithm was produced that can approximate the future mass of a sample to within 10% of the measured mass for four differing humidity levels of 0, 5, 30, and 75% as well as optimize coefficients such as the diffusivity per area defined as $\frac{D}{s^2}$ for some diffusivity D and unknown material thickness s to physically valid values of polyurethane foam. While future work is necessary to create a realistically applicable PGNN, this study indicates that physics can help guide future prediction regardless of a lack of measured data and outcomes to train upon.

TABLE OF CONTENTS

Chapter One Introduction and General Information	1
Controlled Glovebox Environment	1
Foam Samples	1
Robotic Measurement System	2
Physics-Guided Neural Networks	3
Chapter Two Literature Review	5
Water Diffusion	5
Physics Utilization in Neural Networks	7
Chapter Three Methodology	11
Data Munging and Analysis	11
Variable Manipulation	11
Model Architecture	14
Development of Parameter Structure	14
Training Hyperparameters	15
Loss Function	17
Data-Incorporated Loss Function	17
Physics-Incorporated Loss Function	18
Additional Loss Function Developments	20
Chapter Four Results and Discussion	24
Beige Polyurethane Foam	24
Data Only Training	24
Training on Increasing Mass Curves	25
Final Model	27
Gray Polyurethane Foam	36
Model Architecture from Beige Samples	37
Chapter Five Conclusions	42
Successful Results	42
Complications	43
Future Work	44

References	46
Appendix	51
Vita.....	73

LIST OF TABLES

Table 3.1 Example of a data table for a sample. The time, temperature, and relative humidity are inputs and the mass is the output.	12
Table 3.2 Architecture of the layers, number of neurons, and added functions along with the typical starting constants for the final model created for the PGNN on all humidity mass curves for both foam types. A ReLU activation was applied to each hidden layer, with a 10% dropout of neurons randomly selected. The number of samples per input is set to 100 in the mini batch.	16
Table 4.1 Values for the constants determined by the model trained on all mass curves. In general, the 0% humidity constants are radically different from the others due to a modified equation. For fundamental constants of the foam, the diffusivity term is very consistent as is the b parameter for the non-zero humidity groups. The a parameter is mostly consistent for 30% and 75%, but the 5% value differs, likely from other diffusion effects at low humidity.	30
Table 4.2 Values for the constants from the model trained on unideal initial guesses. The constants converge to similar values as in Table 4.1, especially in the diffusivity per area and somewhat in the initial mass, with the a and b parameters having different values but overall produce similar values in the humidity term of the physics equation.	34
Table 4.3 Values for the constants from the model trained on gray sample #7. The constants optimize differently than those of the beige samples in Tables 4.1 and 4.2, but match well to the mass change data. In terms of physical parameters, the diffusivity per area is generally lower across all levels compared to the beige samples, indicating a slower rate of diffusion; this matches expectation as the higher density gray foam would experience a lesser degree of diffusion.	41

LIST OF FIGURES

Figure 1.1 Picture of the foam samples in the controlled glovebox. Both the beige and gray foam samples are shown along with a few metal placeholders as a comparative object. Courtesy of Y-12 National Security Complex.	2
Figure 1.2 Picture of the robotic arm and the precision scale measuring a sample. The arm automatically moves the sample from its placeholder to the scale and returns it after measurement. Courtesy of Y-12 National Security Complex.	3
Figure 3.1 Data collected for Sample #4, a beige polyurethane foam, showing the relative mass change and relative humidity over time. The four humidity groups of 0%, 5%, 30%, and 75% show different regimes of mass change, with the loss of water causing the initial decrease and subsequently gaining water after. This follows diffusion mass changes seen in Jansen et al. [18].	12
Figure 3.2 A plot of a training for sample 4, a beige foam. The training is conducted for 5000 epochs over approximately 50 minutes at an initial learning rate of 0.0005. The loss curve indicates typical NN training of an exponential decrease, with the final loss reaching 0.327 for all loss terms, traditional and physics based.	17
Figure 3.3 Visual schematic of the PGNN during training for N epochs.	23
Figure 4.1 Prediction results for a traditional NN architecture on the 75% humidity mass curve for beige sample #4. The NN was given 50% of the starting data to train upon, which it is able to fit to a high fidelity, but is unable to have any future time prediction in the remaining data used in testing as expected for a data-only NN. ...	25
Figure 4.2 The results for the 75% humidity mass curve of beige sample #4 for the model trained on the three curves of increasing mass. All curves were trained using 40% of the data for the traditional NN with the physics loss function retaining a higher influence on the outcome. The predictions closely match the true mass change, with the physics aiding in predicting the untrained data at a small cost of error in the training data predictions.	26
Figure 4.3 Residuals of the 75% humidity mass curve of beige sample #4 for the model trained on three curves of increasing mass. The relative mass change remains within	

approximately 0.02-0.04% with the most deviation being about 0.06%, giving a high-fidelity prediction. 26

Figure 4.4 Results for the 0% humidity mass curve of beige sample #4 for the fully-trained model on all curves using 40% of the data for training. The predictions overall are closely aligned, but the test curve diverges slightly as time increases. This is likely due to the physics equation being inaccurate at later times for this behavior or from conflict in the NN from the close centers of the RBF for 0% and 5%. 28

Figure 4.5 Residuals for the 0% humidity mass curve of beige sample #4 for the fully-trained model on all curves using 40% of the data for training. The predictions are within +/- 0.02% of zero with the test data predictions diverging slightly, but only deviates up to 0.06% from the actual data indicating a decent fit. 28

Figure 4.6 Results for the 75% humidity mass curve of beige sample #4 for the fully-trained model on all curves using 40% of the data for training. The predictions overall are closely aligned, but the test curve separates a small amount from the true mass as time increases. This is likely due to the physics equation providing inaccuracies for the later time data or the NN having conflicting weights and biases formed during training from the decreasing and increasing mass data. 29

Figure 4.7 Residuals for the 75% humidity mass curve of beige sample #4 for the fully-trained model on all curves using 40% of the data for training. The predictions are within +/- 0.03% of zero with the test data predictions separating slightly, but only deviates up to 0.08% from the actual data indicating a decent fit. 29

Figure 4.8 The one-dimensional diffusion equation with optimized parameters against the mass change data for the 0% humidity group of beige sample #4. The equation has good fidelity with the curve, indicating the parameters match well to the physical system and validates the assumption of the one-dimensional solution to a high degree. 32

Figure 4.9 The one-dimensional diffusion equation with optimized parameters against the mass change data for the 75% humidity group of beige sample #4. The equation has good fidelity with the curve, indicating the parameters match well to the physical

system and validates the assumption of the one-dimensional solution to a high degree.	32
Figure 4.10 Results for the 75% humidity mass curve of beige sample #4 with training using constant coefficients in the physics equation. Without updating the constants, the NN trains on an optimized curve as seen in Figure 4.9, giving arguably better results than the NN having to optimize the coefficients during training.	33
Figure 4.11 Residuals for the 75% humidity mass curve of beige sample #4 with training using constant coefficients. The spread of residuals is slightly less than Figure 4.7, but still has some deviation up to approximately 0.08%.	33
Figure 4.12 Results for the 30% humidity mass curve of beige sample #15 using the model trained on data from beige sample #4. As #15 is subjected to near similar conditions as #4, the model matches well to unseen data from a different sample, with robustness against small deviations in the inputs that exist between both samples.	35
Figure 4.13 Residuals for the 30% humidity mass curve of beige sample #15 using the model trained on data from beige sample #4. Most of the prediction is within +/- 0.02% of the data with the most deviation at the beginning only reaching up to 0.05%.	35
Figure 4.14 Results for the 5% humidity mass curve of gray sample #7 for the fully-trained model on all curves using 40% of the data for training. The predictions for this group suffer from a general lack of fidelity in the training data and from unphysical spikes. This is likely due to an unoptimized NN architecture for this foam type being unable to precisely predict the very small mass changes at this humidity level.	38
Figure 4.15 Residuals for the 5% humidity mass curve of gray sample #7. The middle time section differs only by about +/- 0.001%, but suffers from unphysical spikes and a diverging difference at the beginning time up to -0.008.	38
Figure 4.16 Results for the 75% humidity mass curve of gray sample #7 for the fully-trained model on all curves using 40% of the data for training. The predictions for this group are generally of a high fidelity as with the beige sample.	39

Figure 4.17 Residuals for the 75% humidity mass curve of gray sample #7. The results are of high fidelity, within +/- 0.02% for most of the data and only reaching almost 0.05% for the late time prediction.....	39
Figure 4.18 The one-dimensional diffusion equation with optimized parameters against the mass change data for the 5% humidity group of gray sample #7. The equation has good fidelity with the curve, indicating the parameters match well to the physical system regardless of the errors the PGNN predicts in Figure 4.14.	40

LIST OF APPENDIX FIGURES

Appendix Figure 1 Data collected for Sample #4, a beige polyurethane foam, showing the relative mass change and the temperature in Kelvin over time. The relative point for the mass is the minimum value of all masses for the sample.	51
Appendix Figure 2 Data collected for Sample #4, a beige polyurethane foam, showing the relative mass change and the dew point in Kelvin over time. The relative point for the mass is the minimum value of all masses for the sample.	51
Appendix Figure 3 The results for the 5% humidity mass curve for the model trained on the three curves of increasing mass of beige sample #4. All curves were trained using 40% of the data for the traditional NN with the physics loss function retaining a higher influence on the outcome. The predictions are not as closely matched as the other curves likely due to the small mass increase paired with small input changes creating confusion for the model to separate well, but still offers a relatively close prediction.	52
Appendix Figure 4 Residuals of the 5% humidity mass curve for the model trained on three curves of increasing mass of beige sample #4. The relative mass change experiences more deviations than the other humidity mass curves but at most differs by about 0.02% and generally overpredicts the mass. This is likely due to the small mass change causing difficulty for the NN to approximate.....	52
Appendix Figure 5 The results for the 30% humidity mass curve for the model trained on the three curves of increasing mass of beige sample #4. All curves were trained using 40% of the data for the traditional NN with the physics loss function retaining a higher influence on the outcome. The predictions closely match the actual mass curve.....	53
Appendix Figure 6 Residuals of the 30% humidity mass curve of beige sample #4 for the model trained on three curves of increasing mass. The relative mass change differs by approximately 0.02% from zero for most of the time with the most deviation at the beginning up to 0.07% different. Overall, it represents a high-fidelity prediction.	53

Appendix Figure 7 Results for the 5% humidity mass curve of beige sample #4 trained on the full data set. The predictions have similar results to Appendix Figure 3 trained on the increasing mass only, but is also improved on the predictions of the test data.	54
Appendix Figure 8 Residuals for the 5% humidity mass curve of beige sample #4 trained on the full data set. The predictions are within +/- 0.005% of the data with the most deviation at the early time up to -0.02%.	54
Appendix Figure 9 Results for the 30% humidity mass curve of beige sample #4 trained on the full data set. The predictions have similar results to Appendix Figure 5 trained on the increasing mass only.	55
Appendix Figure 10 Residuals for the 30% humidity mass curve of beige sample #4 trained on the full data set. The predictions are within +/- 0.02% of the data with the most deviation at the early time up to -0.05%.	55
Appendix Figure 11 The predictions on the 5% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of beige sample #4. There is overall good fidelity, indicating validity of the equation under complex conditions and the ability of the model to determine physical constants from the data. The values can be seen in Table 4.1.	56
Appendix Figure 12 The predictions on the 30% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of beige sample #4. The fidelity is extremely well, indicating validity of the equation under complex conditions and the ability of the model to determine physical constants from the data. The values can be seen in Table 4.1.	56
Appendix Figure 13 Results for the 0% humidity mass curve for the model trained on all data of beige sample #4 with set constants. The predictions appear to have a higher fidelity than the original predictions in Figure 4.4, indicating that the PGNN had a better training when the equation was optimized and supporting the assumption that the learning of the constants led the PGNN to inaccurately update weights and biases near the beginning of training.	57

Appendix Figure 14 Residuals for the 0% humidity mass curve for the model trained with set constants on beige sample #4. The predictions are within +/- 0.01% of the data for the majority of time up to about 20 days, with the remaining future time predictions deviating up to -0.05%.	57
Appendix Figure 15 Results for the 5% humidity mass curve for the model trained on all data of beige sample #4 with set constants. The predictions appear to have a similar output than the original predictions of Appendix Figure 7, but appears generally smoother except for cases of unrealistic mass spikes likely due to input fluctuations.	58
Appendix Figure 16 Residuals for the 5% humidity mass curve for the model trained with set constants on beige sample #4. The predictions are within +/- 0.005% of the data for the majority of time past around day 38, with the most divergence at the beginning up to -0.025% and several unrealistic mass spikes likely due to input fluctuations.....	58
Appendix Figure 17 Results for the 30% humidity mass curve for the model trained on all data of beige sample #4 with set constants. The predictions appear mostly consistent with the original predictions in Appendix Figure 9, but appears to have better fidelity at the end time and worse fidelity near the start time.....	59
Appendix Figure 18 Residuals for the 30% humidity mass curve for the model trained with set constants on beige sample #4. The predictions are within +/- 0.02% of the data for the majority of time past around day 52, with the most divergence at the beginning up to -0.08% and an unrealistic mass spike likely due to an input fluctuation.	59
Appendix Figure 19 Results for the 0% humidity mass curve for the model trained on all data of beige sample #4 with unideal constants. The predictions have good fidelity and appear to even have better performance than the original predictions in Figure 4.4.....	60
Appendix Figure 20 Residuals for the 0% humidity mass curve for the model trained with unideal constants on beige sample #4. The predictions are within +/- 0.04% of the	

data for almost the full time, with the most divergence at the beginning up to -	
0.06%.	60
Appendix Figure 21 Results for the 5% humidity mass curve for the model trained on all	
data of beige sample #4 with unideal constants. The predictions have a poor	
performance compared to the original predictions in Appendix Figure 7, indicating	
that the model was unable to optimize the weights and biases and constants as seen	
in Table 4.2.	61
Appendix Figure 22 Residuals for the 5% humidity mass curve for the model trained with	
unideal constants on beige sample #4. The predictions deviate more so than other	
5% predictions between -0.02% and 0.03%.	61
Appendix Figure 23 Results for the 30% humidity mass curve for the model trained on all	
data of beige sample #4 with unideal constants. The predictions have decent fidelity	
with the data such as Appendix Figures 9, 17, but appear to have deviation in the	
training data and unreasonable spike at around 63 days.....	62
Appendix Figure 24 Residuals for the 30% humidity mass curve for the model trained	
with unideal constants on beige sample #4. The predictions are within +/- 0.02%	
past day 52 but deviate up to -0.08% at the beginning with an unrealistic spike at	
day 63.....	62
Appendix Figure 25 Results for the 75% humidity mass curve for the model trained on all	
data of beige sample #4 with unideal constants. The predictions have decent fidelity	
with the data such as Figures 4.6, 4.10 indicating the constants optimized close to	
the original values in Table 4.1.....	63
Appendix Figure 26 Residuals for the 75% humidity mass curve for the model trained	
with unideal constants on beige sample #4. The predictions are within +/- 0.04%	
between day 75 and 95, with deviations up to -0.08% at the beginning and 0.08% at	
the end.....	63
Appendix Figure 27 Results for the 0% humidity mass curve of beige sample #15 for the	
model trained on all data of beige sample #4. The predictions have decent fidelity	
but deviate more so than the results for the original predictions in Figure 4.4,	
indicating the varied inputs between the samples can have a significant impact.	64

Appendix Figure 28 Residuals for the 0% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions have decent fidelity between +/- 0.02% up to about day 15 but deviate up to -0.1% at the end..	64
Appendix Figure 29 Results for the 5% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions overall overestimate the true data and has a slightly degraded performance compared to the original predictions in Appendix Figure 3.	65
Appendix Figure 30 Residuals for the 5% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions overshoot the data and are approximately -0.005% different, with the most deviation being -0.02% at the beginning.....	65
Appendix Figure 31 Results for the 75% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions overall have excellent fidelity as with Figure 4.6, likely indicating the inputs are closely aligned between the samples for this group.....	66
Appendix Figure 32 Residuals for the 75% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions are within +/- 0.04% up to about 90 days with the most deviation and +/- 0.06% at the end and beginning respectively.	66
Appendix Figure 33 Results for the 0% humidity mass curve of beige sample #4 for the model trained on only the solo humidity group. The predictions have great fidelity, and arguably better than other 0% predictions of different models as expected for a model not having to process different data with similar inputs together.....	67
Appendix Figure 34 Results for the 5% humidity mass curve of beige sample #4 for the model trained on only the solo humidity group. The predictions have great fidelity, and overall better performance compared to most of the other 5% predictions.	67
Appendix Figure 35 Results for the 30% humidity mass curve of beige sample #4 for the model trained on only the solo humidity group. The predictions have great fidelity similar to other 30% predictions, although has some deviation in the beginning of the training data.....	68

Appendix Figure 36 Results for the 75% humidity mass curve of beige sample #4 for the model trained on only the solo humidity group. The predictions have great fidelity similar to other 75% predictions with generally better performance and the end, although has some deviation in the beginning of the training data.....	68
Appendix Figure 37 Results for the 0% humidity mass curve for the model trained on all data of gray sample #7. The predictions have good fidelity with the data similar to that of the beige sample training	69
Appendix Figure 38 Residuals for the 0% humidity mass curve for the model trained on all data of gray sample #7. The predictions fall within +/- 0.005% up to day 20 before deviating up to -0.02%.....	69
Appendix Figure 39 Results for the 30% humidity mass curve for the model trained on all data of gray sample #7. The predictions have good fidelity with the data similar to that of the beige sample training	70
Appendix Figure 40 Residuals for the 30% humidity mass curve for the model trained on all data of gray sample #7. The predictions fall within +/- 0.01% for almost the whole data with the most deviation at the beginning up to -0.03%.	70
Appendix Figure 41 The predictions on the 0% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of gray sample #7. There is overall good fidelity, indicating validity of the equation under complex conditions and the ability of the model to determine physical constants from the data. The values can be seen in Table 4.3.....	71
Appendix Figure 42 The predictions on the 30% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of gray sample #7. There is overall excellent fidelity, indicating validity of the equation under complex conditions and the ability of the model to determine physical constants from the data. The values can be seen in Table 4.3.....	71
Appendix Figure 43 The predictions on the 75% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of gray sample #7. There is overall good fidelity, indicating validity of the equation under complex	

conditions and the ability of the model to determine physical constants from the data. The values can be seen in Table 4.3.....	72
--	----

LIST OF ABBREVIATIONS

PGNN – Physics-Guided Neural Network

NN – Neural Network

ML – Machine Learning

RMSE – Root Mean Square Error

ReLU – Rectified Linear Unit

CHAPTER ONE

INTRODUCTION AND GENERAL INFORMATION

Within the domain of the development of mechanical systems and manufacturing, the creation of a product undergoes several stages of testing and development. Within such stages of development, manufacturers often use proxy parts in place of real components to save on costs during design phases when assembling products [1]. Such proxy parts are made with cheap materials, with one being polyurethane foam. However, when subjected to different environmental conditions, the foam is more susceptible to change than the real components they represent. One such change is the change in mass from water diffusion into and out of the material due to relative humidity and the subsequent volume change from swelling or contraction. While a typically negligible effect on the order of milligrams, parts that require very precise parameters such as mass and volume as when they were manufactured can fail to act as proxies within the framework of mechanical systems and hinder the overall design process at later times when other parts or the true parts are added.

To address the effect of water diffusion causing mass change in polyurethane foam materials used as proxy parts in mechanical systems on-site, the Y-12 National Security Complex conducted an experiment within a controlled glovebox environment on two types of foam with varied humidity levels, measuring the mass with a precise scale over a period of approximately 100 days. The details of the glovebox and mechanism for measuring the mass as well as for physics-guided neural networks can be seen below.

Controlled Glovebox Environment

Foam Samples

The measurement of mass change of two types of polyurethane foam from General Plastics, FR-3740 (beige) [2] and FR-4550 (gray) [3], was conducted in a controlled glovebox experiment as a result of a change in humidity causing the diffusion of water from the environment. The foam samples have varying density to allow different levels of diffusion, with the beige samples at $40 \text{ lbs/ft}^3 = 0.641 \text{ g/cm}^3$ and the gray samples at $50 \text{ lbs/ft}^3 = 0.801 \text{ g/cm}^3$ and both samples are cylindrical in shape with half the internal volume hollow and the other half solid for a total volume of $6.05 \text{ in}^3 = 99.14$

cm^3 as seen in **Figure 1.1**. There are 24 total samples within the data, with four samples being either a stainless-steel holder or PVC pipe for comparative measurements; the beige and gray cylinders are divided equally at ten samples each.

Robotic Measurement System

The data were collected over the period of approximately 100 days by an automatic system to measure the mass of the samples with varying degrees of relative humidity. The humidity was controlled and measured by a water column and chilled-mirror hygrometers [4] that combine humid air bubbled through the water column with dry air to achieve the desired relative humidity measured by the hygrometer, testing 0%, 5%, 30%, and 75% relative humidity in the glovebox at approximately equal amounts of time in the total experiment. A FANUC robotic arm [5] seen in **Figure 1.2** would pick up each sample individually and place it on a Mettler Toledo XPE 205 high precision balance with a precision of 0.01 mg [6]. The temperature was held approximately constant around 303 K but had some degree of fluctuation as the glovebox was not perfectly sealed due to wiring needs of the robot. The temperature was measured in the glovebox using 16 thermocouples averaged to produce a single value in Celsius before being converted into Kelvin for traditional scientific reporting units.

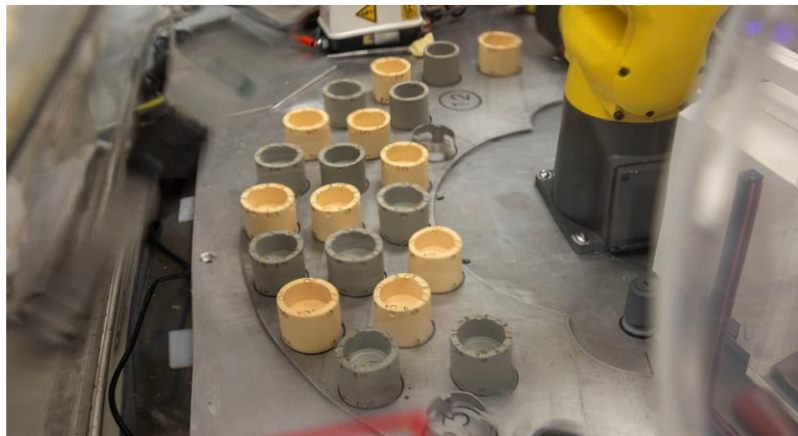


Figure 1.1 Picture of the foam samples in the controlled glovebox. Both the beige and gray foam samples are shown along with a few metal placeholders as a comparative object. Courtesy of Y-12 National Security Complex.

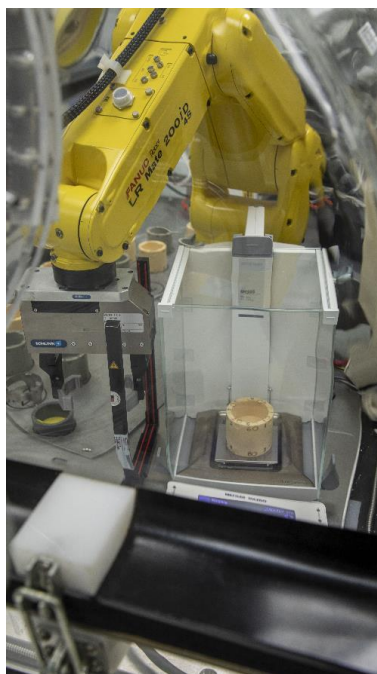


Figure 1.2 Picture of the robotic arm and the precision scale measuring a sample. The arm automatically moves the sample from its placeholder to the scale and returns it after measurement. Courtesy of Y-12 National Security Complex.

An additional precaution to reduce error in the measurement is the use of an anti-static device [7] to remove charges from the surface of the samples. The sample is passed through the device, which creates ions in the air to cancel charges on the surface, before being placed on the scale. This was added after Y-12 discovered inconsistent mass readings when testing the scale, attributing the effect to static on the reading output. At the time of each mass measurement, the approximately constant temperature, ambient air pressure from the environment, controlled humidity, and other miscellaneous variables such as the flow rates of the dry and humid air were recorded in the glovebox as additional data points.

Physics-Guided Neural Networks

The capabilities of NNs serve as a powerful tool of prediction, classification, and other useful tasks that do not require complex modeling or deterministic methods. However, a common flaw in a traditional NN is the requirement of a large amount of training data required to predict on the variable domain of interest, which provides no understanding of

the physical system nor understanding of how the computer optimizes to achieve its outputs, only creating a mathematical map of inputs to outputs. While the NN performs efficiently for the domain of the data trained upon, it catastrophically fails to provide confident predictions for new data not within the domain. For many practical systems in which NNs would provide an advantage, there lacks sufficient data collected from the system to allow for a generalized predictive trained model, a problem that often plagues much of supervised machine learning [8].

A variation on the traditional NN that aids in training and prediction outside of domain of collected data is the physics-guided neural network (PGNN). The PGNN is a combination of a traditional neural network that trains on data collected for the physical system and the utilization of physics equations that model the underlying process to provide constraints for the NN during training [9]. While the traditional portion is confined to a limited domain of the data, the physics portion can cover the entirety of the desired domain including both that in which the data resides and outside of which data is not collected. The result is a model that can train to match the progression of the system for real data while approximating the progression of the system outside of the domain, leading to a greater level of prediction that a standard NN would be unable to provide. An added benefit to the use of physics in the training is the allowance for constants of the equations to be determined through a fit to the available data; from a given starting value, the model will train to determine constants that best fit the equation output to the available data. While the level of fidelity of the model is dependent on the accuracy to which the physics equations describe reality, a PGNN imposes constraints that can provide greater accuracy for data outside the domain compared to a NN.

This analysis aims to combine simple physics of diffusion with the data collected on the foam from experimentation into a PGNN for the prediction of future mass change that has fidelity with the true mass change past an arbitrary ending point of data utilized in training. Results portraying a similarity in curve shapes with predictions within a low percentage of the true mass change would indicate the initial success and potential implementation of the PGNN for this problem of diffusion and for other physical systems.

CHAPTER TWO

LITERATURE REVIEW

Water diffusion is a well-studied topic but can be increasingly complex for multi-dimensional analysis with non-uniform shapes, while the use of physics within NNs has many successful implementations for various physical systems, from using physics-informed machine learning (ML) with the utilization of differential equations to ML regularized and/or guided by physics equations as applied in this thesis. The following literature review discusses the physics of water diffusion in various materials including polyurethane foam as well as physics utilized in ML for the analysis of various problems.

Water Diffusion

Classical diffusion was first circumscribed into a theory by Adolf Fick to describe how the macroscopic concentration of the diffusing liquid changes over time in his second law [10]. The second law is a second-order partial differential equation of the form:

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial x^2}$$

where c is the concentration in dimensions of $[\# \text{ length}^{-3}]$, t is the time, D is the coefficient of diffusion in dimensions of $[\text{length}^2 \text{ time}^{-1}]$, and x is the length. This equation relates how the concentration of molecules of the liquid over time diffuses into or out of a given length of the material. The differential equation by itself does not offer much utility for describing a given system, requiring it to be solved for the conditions under which diffusion is occurring such as the geometry, although it has utility for some physics utilization in ML such as physics-informed neural networks that utilize differential equations. For this use, it would require data of the concentration of the liquid, the time, and the distance it travels which are difficult to measure, but have been accomplished to model diffusion [11].

The solution to the one-dimensional form of the differential equation for a slab geometry using generalized bounds and converting the concentration into weight by integrating the concentration over the thickness has the form [12]:

$$G \equiv \frac{m - m_i}{m_m - m_i} = 1 - \frac{8}{\pi^2} \sum_{j=1}^{\infty} \frac{\exp \left[-(2j+1)^2 \pi^2 * \frac{D_x t}{\left(\frac{1}{2}s\right)^2} \right]}{(2j+1)^2}$$

that can be simplified to the approximation via fit [13]:

$$G = 1 - \exp \left[-7.3 \left(\frac{D_x t}{s^2} \right)^{0.75} \right]$$

where D_x is the diffusivity of the material, s is the thickness, and t is the time. This study combines $\frac{D_x}{s^2}$ into a single constant C due to uncertainty in the thickness s . Further rearranging gives a weight percent change [13]:

$$M = G(M_m - M_i) + M_i$$

where M is the percent weight gain, M_i is the initial moisture content in percent, and M_m is the maximum moisture content obtainable for the environmental conditions in percent given by $M_m = a\phi^b$ for relative humidity ϕ in percent and unitless constants a/b that depend on the material. This equation is the simplest form for use in the PGNN to give a model prediction for the weight change. The equation can be extended to more dimensions to provide greater physical insight into diffusion on multiple axes for more complex shapes, although this requires a more involved solution [14] [15]. The typical shape of the weight change as a function of time models that of increasing exponential with an asymptote at the saturation point, which is verified by several studies [13] [15] [16] [17] [18].

The physics of diffusion is heavily controlled by various factors, some of which small changes can have profound effects on the diffusivity. The largest factor in the value of the diffusivity comes from the temperature of the environment, which controls the speed of the diffusing molecules. It has been well studied that the diffusion coefficient increases with increasing temperature: depending on the material, this change can be by a factor of 5 to 6 for changes of 90 K in foams [18] to a factor of 10 for changes of 100 K in graphite materials [13]. These findings are supported by an increase in moisture absorbency with

increased temperature [19]. The increased diffusion coefficient correlates to a shorter time of diffusion per area as the units are in inverse time. The relative humidity and the water content within the material also has a noticeable impact, with higher concentrations of water typically decreasing the coefficient due to effects such as blocking of water uptake and hydrogen bonding between the water and material [20] and higher humidity driving a faster absorption rate [21]. The barometric pressure also provides some degree of influence on the diffusivity with higher pressures driving greater water uptake [15], although this effect is neglected in this study due to no major fluctuation in the pressure over the 100 days. Due to these effects on the parameters of the diffusion equation, each of the four humidity levels analyzed in this thesis are given their own set of constants to achieve the best modeling of the physics for their environmental conditions such as temperature, air pressure, and relative humidity. A single set of values could be used but the overall fit would be worse due to the parameters having environmental influences. A final effect on the material from water uptake and temperature, especially for porous materials such as foam, is the change in volume. With the absorption of water into the foam material, a corresponding increase in volume results that is dependent on the swelling efficiency of the material [22] [16] [21]. With regards to the temperature, materials have a tendency to expand and contract when subjected to higher or lower temperatures due to the increased energy in the structure of the material. This phenomenon is controlled by the thermal expansion coefficient and can lead to several factors of volume change within a 100 K difference in the temperature [22]. While a change in volume affects the density of the material and can influence the rate of diffusion [15], the change to the coefficient is considered negligible for this study due to requiring a substantial density change not seen in this study to affect the diffusion in an apparent manner.

Physics Utilization in Neural Networks

The incorporation of physics to constrain and aid in NN techniques for various applications is a relatively recent area of research. Many uses of physics to inform, regularize, and guide ML have been established to advance the computational accuracy of models.

Previously discussed was the use of a physics-informed NN utilizing the physics of Fickian diffusion; the “informed” portion of the architecture takes advantage of the use of gradients and derivatives in the NN to implement a loss function based on the differential equation of the underlying physics [11]. By taking gradients of the predictions over the variables of the data as they appear in the equation, such as the concentration over distance, the NN can be constrained to predict outside the training data domain by satisfying the differential equation, but requires knowledge of the governing differential equation and associated variables. However, this method can provide greater fidelity to the predictions of the NN as well as be used to solve unknown constants via gradient descent [11]. The method of “informed” NNs has been extended into a limited number of fields due to the complex nature of many problems preventing the understanding of underlying differential equations, but has some success such as: the prediction of the moisture content in plant cells using differential equations of conservation laws of mass and energy [23]; the estimation of hydraulic conductivity in saturated and unsaturated subsurface flows using Darcy’s law [24]; solving the diffusivity equations with sources and sinks in similar subsurface flow problems [25]; and the merging of numerical methods from partial differential equations to aid a NN on diffusion-sorption in clay [26]. A related variation to the informed NN and a close relative to the “guided” NN is the model that is “regularized” by physics. Using a governing equation that models a part of the domain of interest while having data for a different region of the domain allows a coupling of modeling and NN prediction, using the physics to regularize the point of coupling and constrain future NN predictions. This method is implemented to predict the amount of swelling within SiC due to neutron irradiation, a material to be potentially used as cladding for nuclear reactors, using a physics model for the lower temperature domain to dovetail into experimental data in the higher temperature domain of interest [27]. The physics model becomes increasingly invalidated at higher temperatures, but can constrain the NN prediction at the point of overlap.

The final variation, which this paper analyzes, is the use of physics to “guide” the NN. This method similarly uses a governing equation that models behavior as in the “regularized” method, but utilizes the physics to act as a constraint over the whole

domain of interest, including the data, instead of modeling a part of the domain to couple to the domain with data [9]. The predictions of the NN must conform to the governing physics as to prevent unreasonable outcomes for the domain of the training data and for future prediction, on which traditional NNs typically fail. There are a variety of different physical applications for which PGNNs have been used to improve prediction and determine unknown coefficients. A PGNN is able to aid in hybrid process modeling of a nonlinear continuous stirred-tank reactor used in chemistry by enforcing a material balance using conservation laws [28]. Other nonlinear dynamics such as the Lotka-Volterra model, Duffing equation, and Lorenz system can be partially incorporated at an intermediate layer in a NN to improve model accuracy [29]. Power flow equations help determine the best operation of existing systems with data-driven approaches lacking generalizability due to simplified assumptions, but can be improved by incorporating Kirchhoff's laws into the NN [30]. In terms of mechanical systems, structural damage supervised learning methods suffering from sparse damage data are supplemented by physics from the finite element model [31], the deformation of elastic plates predicted by NNs include classical theory from the Föppl-von Kàrmàn equations with set boundary conditions [32], and physics-based models are combined with data-driven modeling to aid in the prediction of wear on machining tools [33].

While the incorporation of physics into data-driven NN models has generally aided in accuracy by providing physical constraints to predictions, imperfections within modeling can also hurt the ability of the PGNN. Physics-based models often have assumptions, simplifications, and imperfections in representing real-world dynamics which can hinder performance, making it imperative to use physics in contexts where fewer assumptions in the modeling exist [34]. For this analysis, the use of one-dimensional slab geometry diffusion physics is assumed to model the foam cylinder samples to first-order to highlight the potential of the PGNN. Higher dimensional diffusion physics and/or accounting for effects such as temperature and pressure, regardless of their minimal fluctuation, on the diffusivity could be implemented to improve the physics modeling at a future point. The cost function utilized in this thesis, described in detail later, utilizes logic to discourage unphysical mass changes between subsequent data points to improve

the fidelity of the model with the realistic expectation for the evolution of the mass change.

CHAPTER THREE

METHODOLOGY

The creation and implementation of the PGNN is divided into several portions of data manipulation, neural network architecture configuration, development of the loss function guiding the training, and adjustment of hyperparameters controlling the training.

Data Munging and Analysis

The data collected by the automated system over the course of approximately 100 days was compiled in a spreadsheet with many measurements of other variables taken at the time of the mass measurement for each sample. The data was sorted by sample number in MATLAB [35], with non-important variables not related to the physics of diffusion, such as the air flow rates, removed from the table to reduce spurious correlations in the input data. The variables included as inputs for training include the relative humidity, temperature, pressure, and the time with the sole output variable being the mass change. An example of a portion of a data table can be seen in **Table 3.1**.

It can be seen in **Figure 3.1** that for an individual sample, the relative mass change can be divided into four groups characterized by humidity levels at approximately 0%, 5%, 30%, and 75%. The physics of diffusion dictates that the mass of each group evolves by the same mechanism semi-independently of each other, with the only interaction being the starting/ending mass percent for each group. Therefore, each group is trained separately in the overall model, with the data from each group added subsequently during training. The evolution of the mass curves follows similar trends of mass changes from diffusion such as Fig. 3 of Jansen et al. [18].

Much of the development of the PGNN was limited to a single beige foam, sample #4, for time management and simplicity, with verification performed with a different beige sample (#15) and a more minor analysis of a gray foam, sample #7.

Variable Manipulation

The time variable is recorded in a datetime format that is subsequently changed to units of seconds. For each humidity group in a sample, the time is shifted to make the initial point at zero seconds in the training of each group; this allows the PGNN to have consistency in the treatment of each group, each evolving in an independent manner. For

Table 3.1 Example of a data table for a sample. The time, temperature, and relative humidity are inputs and the mass is the output.

Time (s.)	Mass (g)	Pressure (torr)	Temp. (K)	Relative Humidity (%)
247	61.1177	742.9008	303.98	0.01
2297	61.1086	742.8984	304.2	0.01
7854	61.0954	742.6374	303.77	0.01
13454	61.0805	741.4899	303.59	0.01
...	...	...	...	...

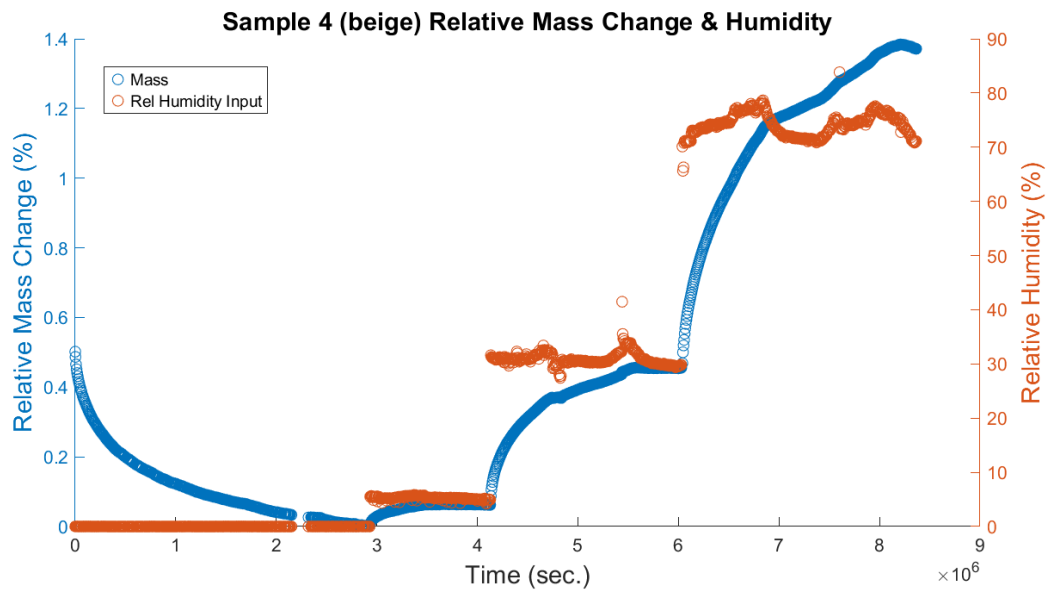


Figure 3.1 Data collected for Sample #4, a beige polyurethane foam, showing the relative mass change and relative humidity over time. The four humidity groups of 0%, 5%, 30%, and 75% show different regimes of mass change, with the loss of water causing the initial decrease and subsequently gaining water after. This follows diffusion mass changes seen in Jansen et al. [18].

training purposes, the time is further converted into units of days to keep the time variable at a lower order of magnitude.

The mass recorded in grams of each sample is converted into a relative percent change to match the output of the physics equations. For the purposes of training, the initial point for the relative comparison is the starting data point for each humidity group to ensure consistency, although such changes cause decreasing mass changes like the 0% humidity group to have a negative relative mass change and initial mass percentage; the negative initial mass percentage is recorded as positive to prevent confusion on an unphysical mass value. It should be noted in **Figure 3.1** that the relative point for the whole domain of data is the minimum mass for all the data. Other plots showing the evolution of the inputs against the mass change can be seen in **Appendix Figures 1-2**.

There were several datapoints that have incorrectly recorded values likely due to computer or instrument error. In some cases, the pressure value was different by a factor of 100 from a decimal point error, which was corrected by multiplying those values by that factor. For relative humidity levels, values that were excessively different from the data points before and after prompted the removal of that observation due to recording error.

An important component in the training of a NN is the standardization of all input variables to prevent any one variable from controlling the overall training [36]. Due to the orders of magnitude difference between many of the inputs, such as time on order 10 in units of days and pressure on order 100 in units of torr, each input variable is standardized using z-score as follows:

$$Z = \frac{X - \mu}{\sigma}$$

where Z is the z-score of the variable, X is the array of values, μ is the mean, and σ is the standard deviation. With this standardization, all input variables range within a few digits of zero and equalizes each within the training scheme. An important distinction in the standardization is how each variable is standardized. As the humidity, temperature, and pressure vary over the whole domain of training data, each of those variables are standardized by the mean and standard deviation of the full range. The time variable is

treated differently as the time is reset for each humidity group; therefore, the time variable is standardized to the mean and standard deviation of each group instead of the full domain.

A final consideration that may affect the reading of the balance scale is phenomenon of air buoyancy on the foam samples. For high precision mass measurements, the fluid of the air will cause a reduction in the true weight of the sample, especially for scales on order 10^{-5} resolution of the mass of the sample or less [37]. The buoyancy force is dependent on the air density and the volume of displacement by the sample; when multiplied, the mass of the displaced air is calculated which can be added to the weight reading to achieve the true weight of the sample. The precise air density is a complex fitted equation dependent on the pressure, temperature, relative humidity and partial pressures of the air [38], and the volume is 99.14 cm^3 for the samples as previously described. While the buoyancy can affect the true weight reading of the scale, it is ignored for this study as the temperature, pressure, and humidity are approximately constant over the course of measurements per humidity group, which constrains the buoyancy to be a near constant term. As such, the shape of the curve and the relative mass change should be consistent across measurements and, along with the extremely small deviations, make any buoyancy correction negligible for training the PGNN.

Model Architecture

Development of Parameter Structure

The architecture of the NN has undergone several iterations to optimize the training and validation. Such changes include the number of layers, the number of nodes per layer, and the starting values for constants used in the physics equations.

The network is a custom-made structure object within MATLAB [35] with substructures of fully connected layers consisting of a weights group holding the values that connect neurons between layers and a bias group holding the offset values of the neurons in the layer. Before training, the weights are initialized with random values using a random number generator from the normal distribution and the biases are initialized to zero; both are converted into learnable parameters through the use of *dlarray* in MATLAB [35]. The final forms of the model consisted of approximately six fully connected layers that

increase in the number of nodes from the input to the middle of the layers before decreasing in size to give a single output, that of the predicted mass change. Each layer uses a rectified linear unit (ReLU) as an activation function with dropout of 10% implemented for each layer. The layout can be seen in entirety in **Table 3.2**.

An additional substructure to the parameters is that of the values for the constants of the physics equations. There are four constants describing the equation, detailed in the following section, that are populated with initial guesses based on values found for other materials undergoing diffusion [13] [20] [39] [40] [41]. Each constant in the structure is an array of four values that account for the four humidity groups trained individually within a sample; during the course of training, the values within each constant will update separately such that each humidity group will obtain its own group of tuned constants describing the data from the curve due to differences in the other factors such as temperature having minor effects on the constants between groups. While the PGNN has the ability to update the constants to learn their physical values, it is limited so far in its ability to make large changes from the initial value given, making it important to seed values that are relatively close to the true values of the constants and at the very least have the same order of magnitude.

Training Hyperparameters

For a standard training event, the hyperparameters are maintained at the following values. A training consists of 5000 epochs on approximately 1400 datapoints for a single foam sample. The data is segmented into the four groups based on the relative humidity, with each group of data placed into a minibatch object that trains on 100 samples per iteration. The initial learning rate is set to 0.0005 with a decay of the learning rate at a value of 0.00001; these values were chosen after trial-and-error of initial training events. The parameters of the NN architecture are updated using the Adam optimizer [42] with the constants of the physics equations manually constrained to be positive by using the absolute value to prevent imaginary results of the equations caused by fraction exponents of negative numbers. An example of a plot showcasing the training of a sample can be seen in **Figure 3.2**.

Table 3.2 Architecture of the layers, number of neurons, and added functions along with the typical starting constants for the final model created for the PGNN on all humidity mass curves for both foam types. A ReLU activation was applied to each hidden layer, with a 10% dropout of neurons randomly selected. The number of samples per input is set to 100 in the mini batch.

Layer	Neurons	Activation	Dropout
Input	7	None	None
Hidden Layer 1	9	ReLU	10%
Hidden Layer 2	25	ReLU	10%
Hidden Layer 3	64	ReLU	10%
Hidden Layer 4	25	ReLU	10%
Hidden Layer 5	9	ReLU	10%
Output	1	None	None

Constant Values	0% Hum.	5% Hum.	30% Hum.	75% Hum.
a	0.0	0.05	0.05	0.05
b	0.8	0.5	0.5	0.5
D_x/s^2	0.01	0.01	0.01	0.01
M_i	0.4	0.001	0.001	0.001

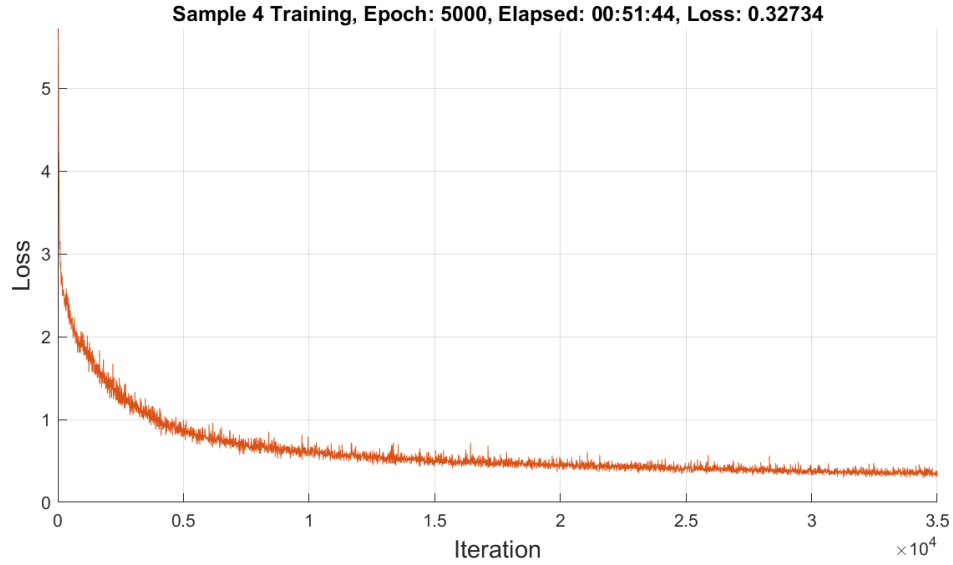


Figure 3.2 A plot of a training for sample 4, a beige foam. The training is conducted for 5000 epochs over approximately 50 minutes at an initial learning rate of 0.0005. The loss curve indicates typical NN training of an exponential decrease, with the final loss reaching 0.327 for all loss terms, traditional and physics based.

Loss Function

Data-Incorporated Loss Function

The portion of the NN driven by the data is considered to be a traditional model that trains on data given to it and would only be confined to that particular domain. The training data, in the standardized format, is fed through the model hidden layers with the ReLU activation functions and dropout incorporation of 10% of the neurons selected randomly to obtain the mass change prediction for the training domain. A root mean square error function (RMSE) is used to calculate the loss between the mass change labels and the predictions. The relevance of the data-driven portion of the NN is dependent on the weight assigned to its loss and the amount of data given to it as training. For each of the four curves for the different humidity groups of a sample, initial tests were performed using 60% of the data for training with the remaining 40% utilized in prediction testing, but after improvements and trials, only 40% of the data was given to the traditional NN for training, leaving the majority of the prediction of the later time evolution to the physics-driven portion. In general, the more data made available for

training correlates to a more accurate matching to the true mass change values, which deemphasizes the importance and use of the physics in the model; the relative importance of either the data or physics is also driven by the weights manually chosen for each loss term. The equation of the data loss function is seen below for N data points, with M_{NN} being the output of the NN and M_{act} being the true mass change:

$$L_{data} = \sqrt{\frac{1}{N} \sum_{i=1}^N (M_{NN,i} - M_{act,i})^2}$$

Physics-Incorporated Loss Function

Within the PGNN framework, the equations of the process studied constrains the NN to make more ideal guesses for untrained data based on the evolution of the physics. From analysis of the evolution of the mass change in the foam samples, it was assumed that the samples followed some form of Fickian diffusion that is best described by a differential equation relating the concentration of the water to its position within the foam. Solving the differential equation for a one-dimensional slab gives the following equation described in the literature review [13]:

$$M = G(M_m - M_i) + M_i$$

where M is the percent weight gain, M_i is the initial moisture content, M_m is the maximum moisture content obtainable for the environmental conditions, and G is a time dependent parameter. G has the approximation:

$$G = 1 - \exp \left[-7.3 \left(\frac{D_x t}{s^2} \right)^{0.75} \right]$$

where D_x is the diffusivity of the material, s is the thickness, and t is the time; it should be noted that the diffusivity constant has inverse time as a unit, and the value of it will be determined by the unit of time that t is. For further simplification and due to the uncertainty around the thickness, D_x and s^2 are combined into a single constant called C . Likewise, the maximum moisture content can be described by:

$$M_m = a\phi^b$$

where ϕ is the relative humidity in percent and a/b are constants dependent on the material that can be determinable from the asymptote of the mass change for a given

humidity. For a material that is losing moisture, G is modified to only include the exponential term. While the samples are complex three-dimensional shapes, it has been determined through testing that the one-dimensional diffusion equation acts as a decent approximation for the behavior of the mass change over time.

The loss term incorporating the mass change equation functions in a similar method to that of the data-driven portion of the NN but requires additional modifications to obtain a physics-based prediction used in the RMSE comparison to the mass labels. The equation produces an output of the mass change that is dependent on the inputs of relative humidity, initial mass change, and time. To obtain the mass prediction, the inputs from the data for those variables must be utilized with the correct units to obtain a physically viable answer; these variables are also taken on the entire domain of the data, including both the portion utilized in the traditional training (usually 40%) and that not used in training, which is important for making decent predictions for those later times. Within the function controlling the loss, the inputs for the physics portion are fed through the model to make a prediction of the mass in which those are inputs are in their z-score standardized form. After the prediction, the variables are transformed back into their physical units by inverting the process of the z-score transformation, which simply requires the multiplication of the standard deviation followed by the addition of the mean for each variable. The standard deviations and means for the variables are saved and used as inputs in the function, with those statistics being per humidity group for the time due to the time being reset when each group is trained and for the whole dataset in the case of the other input variables.

The inputs are fed into the equation in a series of subsequent actions. The first step of this process seeks to update the constants associated with the equations to create a better fit of the equation to the collected data. The initial guesses for the constants are as follows: for the diffusivity per area constant $\frac{D_x}{s^2}$, a value of 0.01 days^{-1} for both sample types was chosen; the a parameter used 0.05 for the non-zero humidity groups and 0.0 for the 0% humidity; the b parameter used 0.5 for the non-zero humidity groups and 0.8 for 0%; the M_i parameter was 0.001% for the non-zero humidity groups and -0.4% for 0%. The 0%

humidity group uses different starting parameters due to the form change of the equation for decreasing mass. All starting parameters were chosen based on a combination of prior research [13] [20] [39] [40] [41] and curve fitting results used as a separate check. For each humidity group, the equation uses the initial guesses to solve the equation and use a RMSE function to compare the output to the true masses. The update based on this loss allows the model to change the values of the constants to better match the data and theoretically converge to values that physically represent the material. For the second portion of the physics loss function, the constants are frozen to prevent a secondary update, with the frozen constants re-inputted into the equation to obtain a new mass output. With this new output, another RMSE function then calculates the loss between the model prediction output using the standardized inputs and the equation output. In essence, the first portion of the physics loss updates the constants to converge based on the data and is given the subscript “csts” while the second portion updates the model weights and biases based on the calculated equation and is given the subscript “phys”. This system allows for the determination of fundamental constants of the material and the updating of the model to predict on future times based on the equation output of those times. Both terms in the physics loss can be controlled in terms of importance in the overall loss function through the weights applied to each term. The equations of the physics loss functions are seen below for N data points, with M_{NN} being the output of the NN, M_{act} being the true mass change, and M_{eqn} being the output of the equation:

$$L_{csts} = \sqrt{\frac{1}{N} \sum_{i=1}^N (M_{eqn,i} - M_{act,i})^2}; L_{phys} = \sqrt{\frac{1}{N} \sum_{i=1}^N (M_{NN,i} - M_{eqn,i})^2}$$

Additional Loss Function Developments

To further aid in the fidelity of the model, other techniques and constraints were applied to the NN and loss function. These techniques are a combination of standard ML regularizations and application-based regularizations to provide a greater physical consistency and prevent drastic predictions due to small changes in the inputs.

The training of the whole data of a sample requires the separation of the different humidity groups, giving four mass curves that are trained subsequently within the whole model. As the model essentially performs four trainings, the inputs for each of the humidity groups are modified using a radial basis function [43]. This function uses a Gaussian function that measures the similarity between data points and chosen values known as centers for comparison, with the following form:

$$RBF = e^{-\frac{(x-x_0)^2}{\sigma^2}}$$

where x is the data point, x_0 is the center, and σ describes the width between centers. The radial basis function takes the humidity input from the data and compares the value to one of four centers chosen based on the values of the standardized humidity percentages: -1 for 0%, -0.8 for 5%, 0.025 for 30%, and 1.424 for 75% relative humidity with σ^2 chosen as 0.01 to make the distinction between the centers sharp. For a given standardized humidity value, the center that most closely matches it will produce the highest value up to the value of one, with the other centers giving values less than and closer to zero. Therefore, the humidity input is split from one value into four inputs, with one of those inputs having the highest value depending on the humidity percentage. In theory, this will allow the NN to isolate different parts of the architecture for training the data from the four humidity groups, allowing different nodes to update their weights and biases depending on the group without confusing the neural network with potentially conflicting inputs from different groups. This could potentially cause some hindrance, however, if the centers of two groups are closely related such as for 0% and 5% humidity if the width is not sharp enough, but could also theoretically be used to predict untrained intermediate humidity levels by splitting a chosen humidity between two centers to get an average prediction.

A traditional addition to the loss function is the use of a regularization term that is typical of many NN applications. The specific type used in the PGNN is known as ridge regression or L2 regularization, which aims to penalize large weights within the architecture in order to reduce model complexity and prevent overfitting of the data [44]. The implementation of L2 regularization required a custom-made scheme rather than a

built-in method in MATLAB [35] due to the custom nature of the PGNN, with the equation being:

$$L_2 = \lambda \sum_{i=1}^p w_i^2$$

with λ being the weighting factor w_i are the weights, and p is the total number of layers. This regularization was ultimately kept to a low-level of importance due to faults it was creating in the training, such as producing constant value predictions, by making the weighting factor 0.01.

A final addition includes an implementation of a physics-based regularization term to encourage better and more realistic predictions. For a foam absorbing or desorbing water, the mass change from one timestep to the next should follow a small increase or decrease in a smooth transition. In other words, the mass change prediction should not be an erratic jump for a small change in the inputs, but follow physically allowable change. To counter these effects in the PGNN, a custom regularization term was implemented that was based around the difference in the mass from a given timestep to the next one:

$$L_{diff} = \frac{\lambda}{M_{diff}} \sum_{i=1}^N ReLU(M_{diff,i} - \zeta)$$

where λ is a weighting factor as before, M_{diff} is the magnitude of the mass difference in predictions between two timesteps, and ζ is a threshold value for the mass difference. For mass differences that exceed the given threshold, a penalty is imposed on the NN that will reduce future predictions from making large guesses; differences less than the threshold will be zero and not contribute per the ReLU function. The weighting factor was set to a high mark over the course of trials to a value of 30.

The overall loss function is as follows:

$$L = \lambda_{data}L_{data} + \lambda_{phys}L_{phys} + \lambda_{csts}L_{csts} + \lambda_{diff}L_{diff} + \lambda_{L2}L_2$$

and the weighting factors in order are 1.0, 3.0, 0.5, 30.0, and 0.01. A visual schematic of the PGNN can be seen below in **Figure 3.3**.

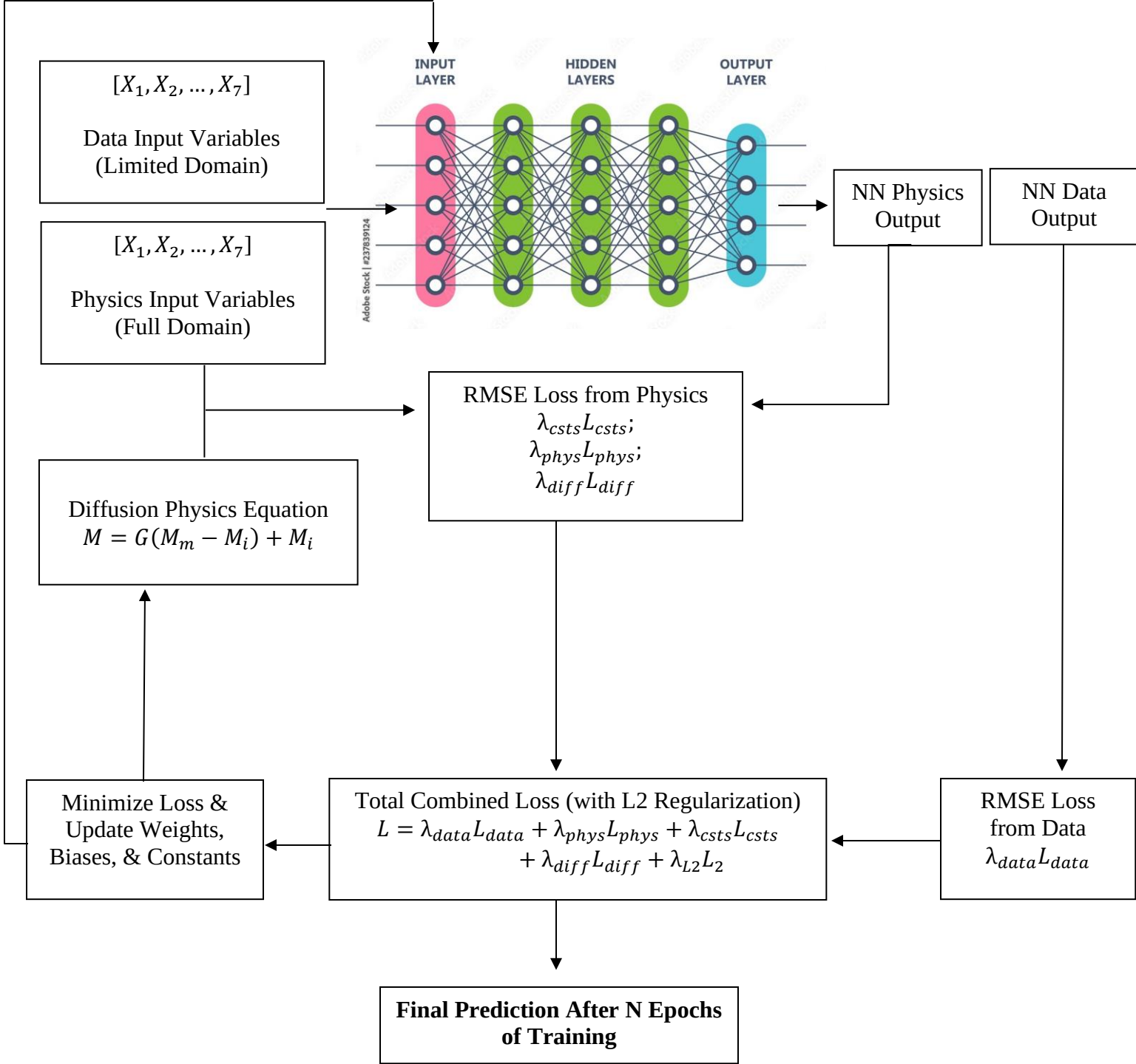


Figure 3.3 Visual schematic of the PGNN during training for N epochs.

CHAPTER FOUR

RESULTS AND DISCUSSION

The result of training features both interesting and promising outcomes as well as problems with the PGNN to be overcome in future work. As a proof of concept, the PGNN exceeds the capabilities of a standard NN, providing greater prediction abilities for a relatively small amount of data available for training in the traditional portion of the model, and could exceed a purely physics-based model when implemented in real environmental conditions. The following results highlights the successes and challenges for future mass change prediction for both foam types.

Beige Polyurethane Foam

The initial trial performances of the PGNN when developing the architecture were performed with the beige polyurethane foam, specifically sample #4 within the 24 total samples used in the 100 days of data collection; cross-sample verification was performed with a different beige foam in sample #15 as seen further below. For the four humidity groups in the total data, each mass curve is isolated from the others, giving one curve of decreasing mass and three of increasing mass. The amount of training data for the traditional NN could be controlled to varying percentages, with the majority of training set to 40% for the finalized model. The following outlines several results of training the PGNN for the beige sample.

Data Only Training

One of the initial tests to highlight the shortcomings of a traditional NN without the use of physics was the implementation of the standard data-driven architecture on 50% of given data. Using the data for the 75% humidity curve from the beige sample #4, training occurred for approximately 1000 epochs with the results seen in **Figure 4.1**. As expected, the NN is able to match the data that it trains on extremely well, with predictions on the remaining data used in the test having little fidelity with the true mass change. This exemplifies the strength of the NN in its ability to predict on data it has trained upon and on inputs that are similar to the training data but also the weakness in predicting solutions to data that evolves in time with inputs that are outside of the training domain.

Training on Increasing Mass Curves

The incorporation of physics into the NN architecture required the implementation of the equation(s) within the loss function controlling how the weights and biases update. During initial trials of the PGNN, only the three mass curves that showcase increasing mass changes were utilized as they all share characteristics of the same controlling physics; the decreasing mass curve has a modification to the physics as mentioned before that was incorporated during later trials. An example of results for the 75% humidity curve of sample #4 can be seen in **Figure 4.2** and **Figure 4.3** with the results for the remaining humidity groups seen in **Appendix Figures 3-6**. The training occurred for the established 5000 epochs on 40% of the data used for the traditional data-driven portion of the PGNN, with the physics loss having a larger weight than the data loss to aid in the prediction of the test data. From the figures, it can be seen that there is a high fidelity between the predictions of the PGNN and the true mass change curve. Based on the residuals between the curves, the predictions remain within 0.02-0.04% of the relative mass change of the data with the largest amount of deviation being approximately 0.06%. The same results occur for the other humidity groups seen in the Appendix, although the 5% humidity predictions appear to be at a lower level of fidelity, but are at most 0.02%

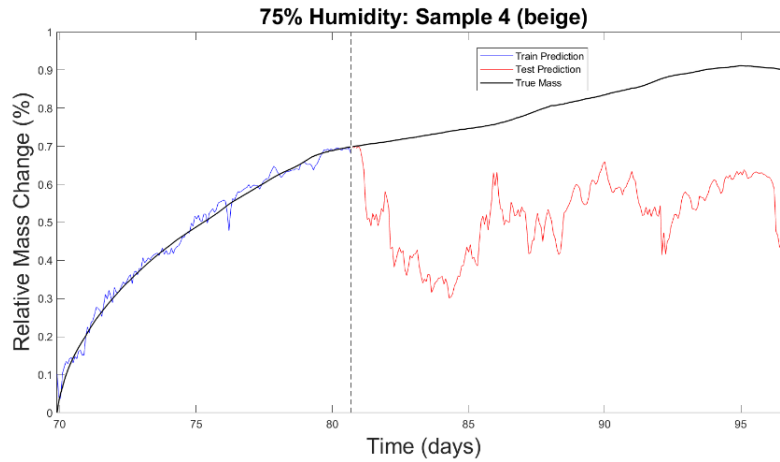


Figure 4.1 Prediction results for a traditional NN architecture on the 75% humidity mass curve for beige sample #4. The NN was given 50% of the starting data to train upon, which it is able to fit to a high fidelity, but is unable to have any future time prediction in the remaining data used in testing as expected for a data-only NN.

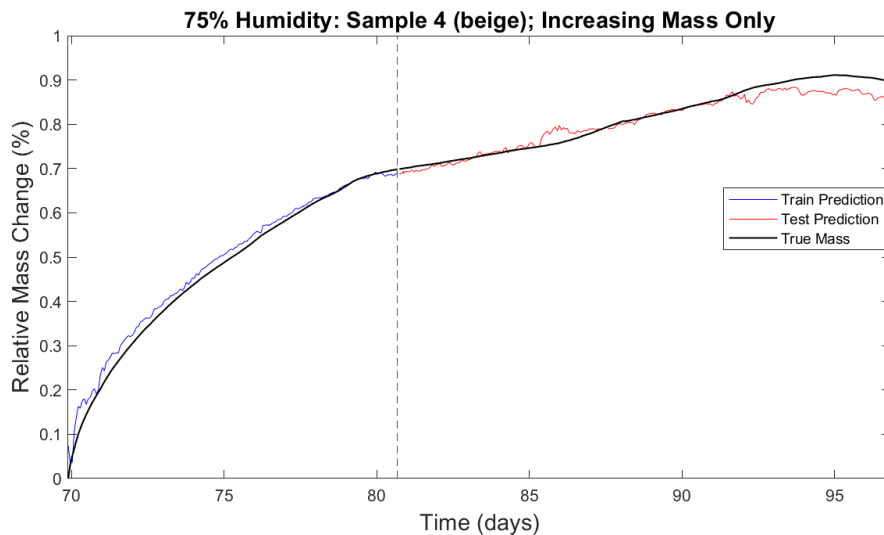


Figure 4.2 The results for the 75% humidity mass curve of beige sample #4 for the model trained on the three curves of increasing mass. All curves were trained using 40% of the data for the traditional NN with the physics loss function retaining a higher influence on the outcome. The predictions closely match the true mass change, with the physics aiding in predicting the untrained data at a small cost of error in the training data predictions.

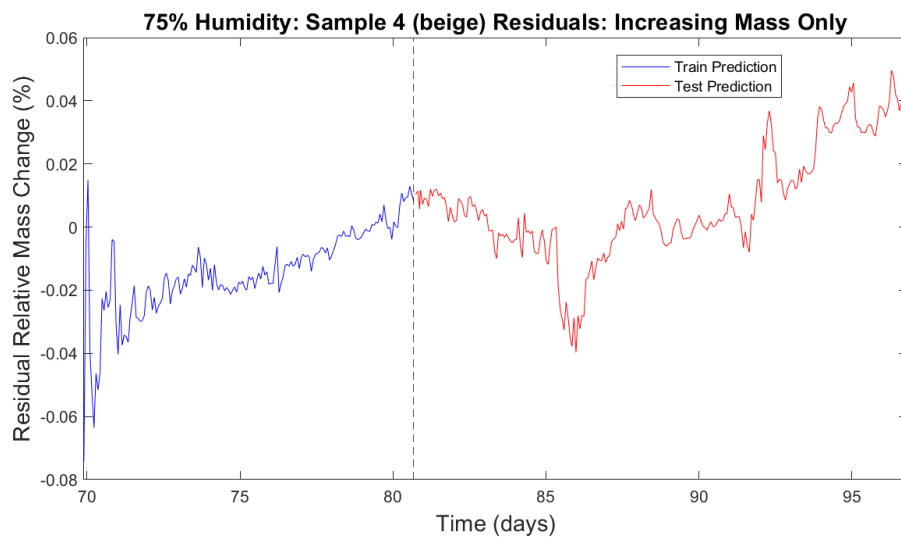


Figure 4.3 Residuals of the 75% humidity mass curve of beige sample #4 for the model trained on three curves of increasing mass. The relative mass change remains within approximately 0.02-0.04% with the most deviation being about 0.06%, giving a high-fidelity prediction.

different from the actual mass change; the model generally predicted a higher mass change. The mismatch of the curves is most likely due to the extremely small mass change that occurs in the foam at low humidity, along with small changes in the inputs between mass outputs creating confusion within the NN.

Final Model

The final model created for the beige samples incorporated both the increasing mass curves as well as the decreasing curve. The decreasing portion is accounted for in the loss function through the use of coding logic to separate the different humidity groups by their respective data; this logic is also applied to keeping the constants for the equations separate for implementation in the loss function. The addition of the decreasing mass represents a more complex problem for the PGNN to solve as the nature, shape of the data, and values of the constants are radically different than the non-zero humidity mass curves. This requires the architecture of the NN to update weights and biases to incorporate different pathways to achieve negative and positive mass predictions, which is the foundation for the use of the radial basis function. The results for the 0% humidity mass curve and the 75% humidity mass curve can be found in **Figures 4.4-4.7**, with the remaining humidity mass curves found in **Appendix Figures 7-10**. It can be seen in the figures that there are some deviations between the predictions and the true mass curves that are not seen for the predictions of the model trained only on the increasing mass curves such as in **Figure 4.2**. The likely cause for such differences between the two models is the introduction of conflicting mass outputs for similar inputs across all variables except for the humidity; the radial basis function serves to aid in separating the NN architecture for different humidity groups, but due to the black box nature of the NN, there could be overlap between weights and biases in the layers conflicting the outputs. The negative mass curve of the 0% humidity requires a varying set of constants compared to the increasing mass curves and has a modification to the physics equation used unilaterally by the other data. These variations could create confusion in the NN to account for the prediction of negative mass, impacting the predictions for all curves to a small degree. While this problem could be rectified by improvements to the architecture, including the addition of more layers and/or neurons, the results as it stands have mostly

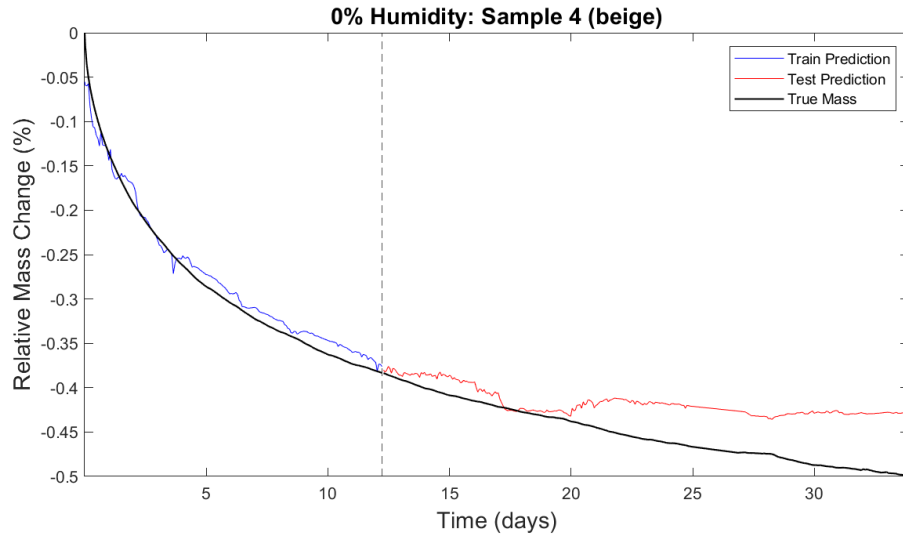


Figure 4.4 Results for the 0% humidity mass curve of beige sample #4 for the fully-trained model on all curves using 40% of the data for training. The predictions overall are closely aligned, but the test curve diverges slightly as time increases. This is likely due to the physics equation being inaccurate at later times for this behavior or from conflict in the NN from the close centers of the RBF for 0% and 5%.

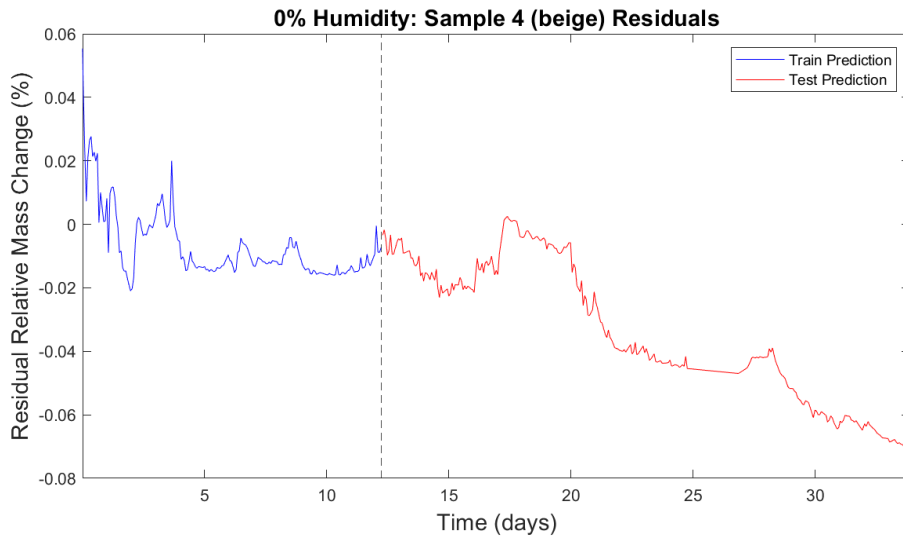


Figure 4.5 Residuals for the 0% humidity mass curve of beige sample #4 for the fully-trained model on all curves using 40% of the data for training. The predictions are within $\pm 0.02\%$ of zero with the test data predictions diverging slightly, but only deviates up to 0.06% from the actual data indicating a decent fit.

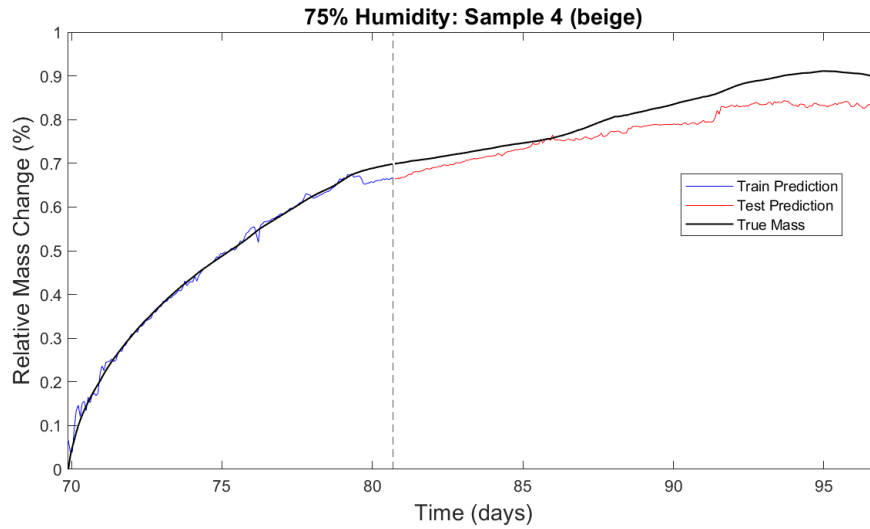


Figure 4.6 Results for the 75% humidity mass curve of beige sample #4 for the fully-trained model on all curves using 40% of the data for training. The predictions overall are closely aligned, but the test curve separates a small amount from the true mass as time increases. This is likely due to the physics equation providing inaccuracies for the later time data or the NN having conflicting weights and biases formed during training from the decreasing and increasing mass data.

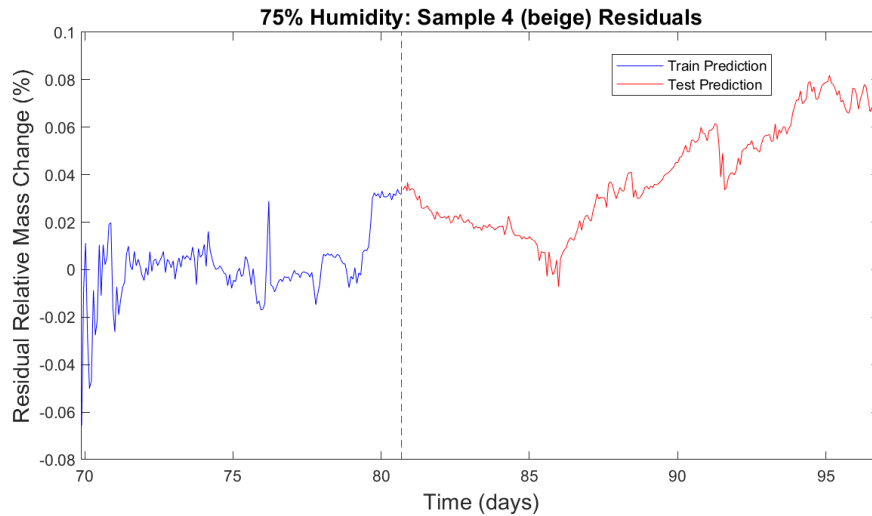


Figure 4.7 Residuals for the 75% humidity mass curve of beige sample #4 for the fully-trained model on all curves using 40% of the data for training. The predictions are within $\pm 0.03\%$ of zero with the test data predictions separating slightly, but only deviates up to 0.08% from the actual data indicating a decent fit.

consistent predictions to the actual mass curves with the deviation being within +/- 0.03% and being never more than 0.08%. It cannot be assumed what the model would predict if given more time past what data currently counts to, the mass change would asymptote to a constant value as the moisture saturates, but the trends of the model predictions seem to indicate an asymptote as well giving a future prediction that has a deviation from the true saturation but would remain constant. The values of the updated constants can be seen in **Table 4.1**. The constants are overall very consistent with each other, though the 0% humidity data overall has differing constants in the parameters a , b , and M_i due to the nature of the modified diffusion equation. For the diffusivity per area, the standard deviation is 0.006 and is centered around 0.0134 days^{-1} . Depending on the type of polyurethane and the conditions such as water concentration and temperature, the coefficient can vary but for some foams have values ranging from 0.00864 - 0.2 days^{-1} [39] [41]; other polyurethane foams have smaller values of 0.001 - 0.003 days^{-1} [40] [20]. It can be assumed that the optimized coefficient from the PGNN is a valid physical value as it remains within known bounds. The output of the equations utilizing the optimized outputs against the true mass curves for both 0% and 75% humidity groups can be seen in **Figures 4.8-4.9**, with the remaining humidity groups in **Appendix Figures 11-12**. It can be seen that the output of the equations is a close match the behavior of the mass change, signifying the ability of the PGNN to optimize the constants and validating

Table 4.1 Values for the constants determined by the model trained on all mass curves. In general, the 0% humidity constants are radically different from the others due to a modified equation. For fundamental constants of the foam, the diffusivity term is very consistent as is the b parameter for the non-zero humidity groups. The a parameter is mostly consistent for 30% and 75%, but the 5% value differs, likely from other diffusion effects at low humidity.

Humidity Curve (%)	D_x/s^2 (days^{-1})	a	b	M_i (%)
0	0.0133	0.000	1.2059	0.4625
5	0.0251	0.0298	0.4363	0.0041
30	0.0128	0.0744	0.4847	0.0197
75	0.0098	0.0960	0.5297	0.0057

the assumption of one-dimensional diffusion in this problem. It also highlights a confusing outcome of the PGNN in that the NN predictions are sometimes worse than the equation prediction, such as later time predictions after 90 days in **Figure 4.6** and **Figure 4.9**, when the NN trains on the equation output at that time. Although no reason is certain, it is most likely believed that poor modeling from the equation during early times in training when the constants are not optimized causes the NN to update weights and biases to unideal values that do not update to better values later in training when the equation is optimized due to a decay in the learning rate reducing gradient changes. To test this theory, a new training was performed with the same architecture on sample #4 with constants set to the optimized values that were determined, with results for the 75% humidity group in **Figures 4.10-4.11** and the remaining results in **Appendix Figures 13-18**. There is a noticeable improvement in the later time predictions compared to **Figure 4.6**, but also has a slightly degraded performance in the training data portion. While it is again uncertain why, it is assumed that the NN is modeling the disparity between the equation and mass change in **Figure 4.10** while focusing instead on the data in **Figure 4.6**; the black box nature prevents true knowledge of how the NN updates during trainings, though parameters such as the initialization and loss function weights can affect it.

A general trend seen in the diffusivity per area is the decrease in value from low humidity to high humidity, with the 75% humidity group typically having the lowest coefficient. This effect is likely due to the saturation of water within the foam from previous absorption in the lower humidity groups, blocking new moisture from absorbing in free volume holes and/or clustering water molecules in the foam reducing the diffusivity such as in Fig. 1 of Zhou & Choi [20].

The initial values for each constant were chosen to be close to the assumed true values based on the polyurethane diffusion coefficient ranges [39] [20] [40] and fits to the data performed separately to ensure the model maintains high precision with a shorter computation time. To test the ability of the PGNN to update the parameters to ideal values from varied initial conditions, the model was retrained using values an order of

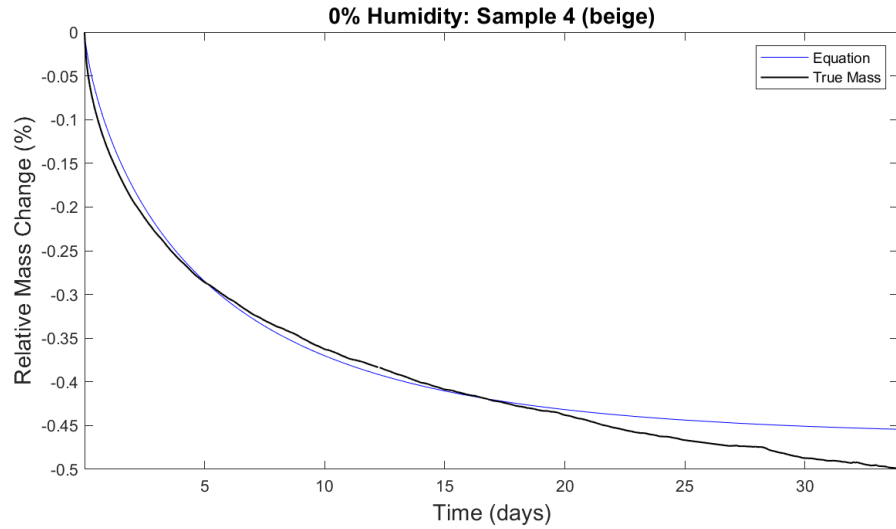


Figure 4.8 The one-dimensional diffusion equation with optimized parameters against the mass change data for the 0% humidity group of beige sample #4. The equation has good fidelity with the curve, indicating the parameters match well to the physical system and validates the assumption of the one-dimensional solution to a high degree.

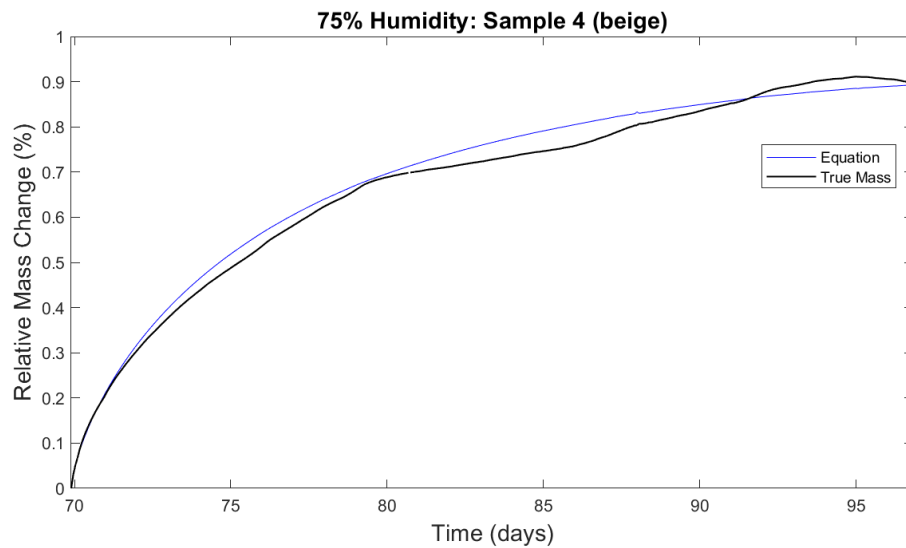


Figure 4.9 The one-dimensional diffusion equation with optimized parameters against the mass change data for the 75% humidity group of beige sample #4. The equation has good fidelity with the curve, indicating the parameters match well to the physical system and validates the assumption of the one-dimensional solution to a high degree.

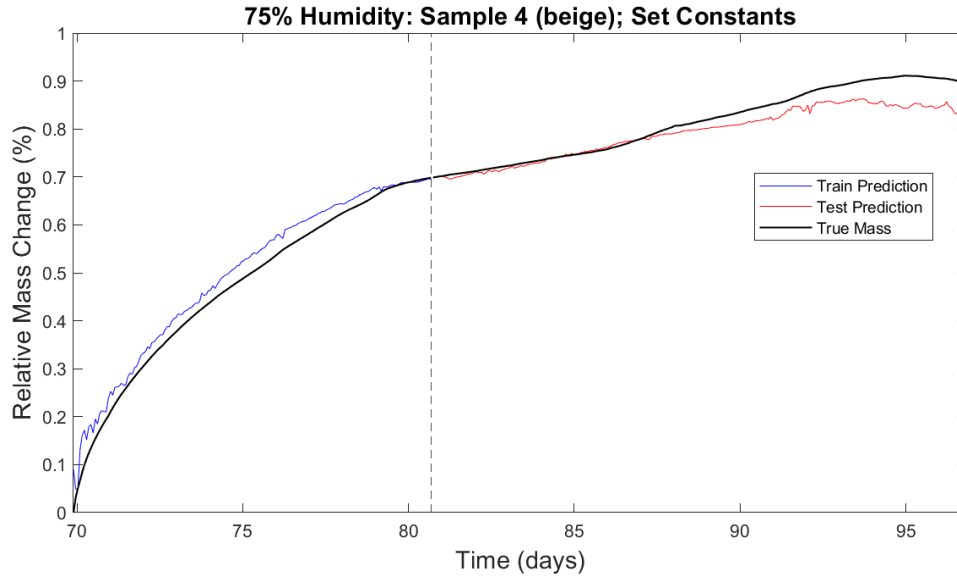


Figure 4.10 Results for the 75% humidity mass curve of beige sample #4 with training using constant coefficients in the physics equation. Without updating the constants, the NN trains on an optimized curve as seen in **Figure 4.9**, giving arguably better results than the NN having to optimize the coefficients during training.

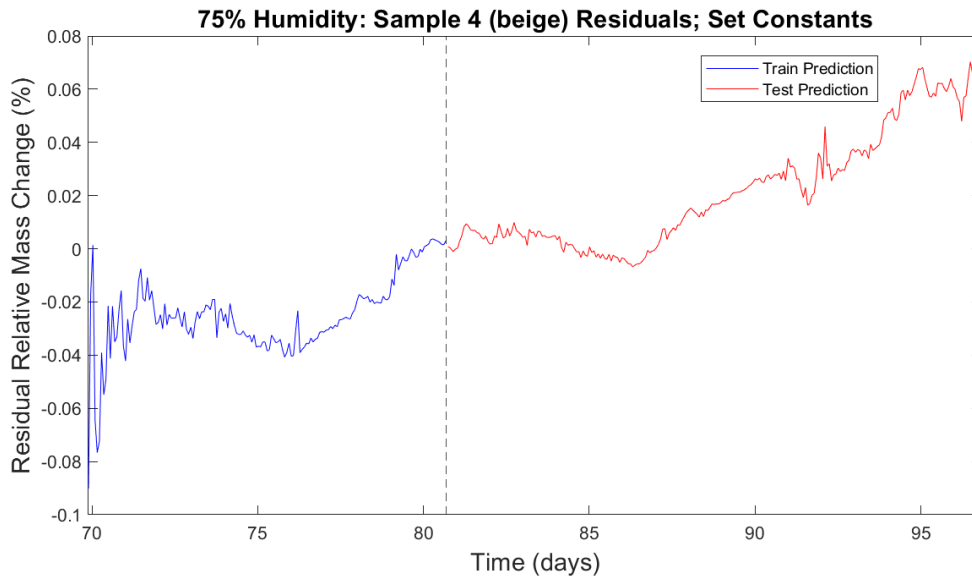


Figure 4.11 Residuals for the 75% humidity mass curve of beige sample #4 with training using constant coefficients. The spread of residuals is slightly less than **Figure 4.7**, but still has some deviation up to approximately 0.08%.

magnitude different from the original initial values used. The initial values used were 0.1 for the diffusivity per area, 0.5 for **a**, 1.5 for **b**, with the initial mass kept the same at 0.001. The updated constants can be seen in **Table 4.2** and the plots of results can be seen in **Appendix Figures 19-26**. The training extended to 6000 epochs to account for the longer time needed to reduce the loss value. The plots indicate that the model was able to converge to a high-fidelity solution similar to the model trained on ideal initial values, although the 5% humidity mass curve solution has larger deviations and less fidelity that is typical for the predictions of the data of that humidity. The constants converge to similar values as in **Table 4.1**, with the diffusivity per area obtaining values within 0.1-0.13% of the other model and the initial mass achieving approximately similar values. The **a** and **b** parameters updated to different values than the original model, but through testing of their use in the equation $a\phi^b$ with the humidity, achieved approximate values such as 0.981 and 0.945 for the 75% humidity value. This suggests that the model converged on another minimum for those constants that worked similarly to the minimum of the original model and could explain the deviations in the results.

To test the model on different data that was not trained upon, the parameters were applied to a different beige sample, #15, in order to determine if the PGNN can generalize to other samples of the same type. An example of a test on the 30% humidity mass curve can be seen in **Figures 4.12-4.13**, with the rest of curves found in **Appendix Figures 27-32**. It can be seen from the figures that the prediction curve matches the true mass to a

Table 4.2 Values for the constants from the model trained on unideal initial guesses. The constants converge to similar values as in **Table 4.1**, especially in the diffusivity per area and somewhat in the initial mass, with the **a** and **b** parameters having different values but overall produce similar values in the humidity term of the physics equation.

Humidity Curve (%)	D_x/s^2 ($days^{-1}$)	a	b	M_i (%)
0	0.0120	0.000	2.298	0.4751
5	0.0219	0.0123	1.0017	0.0047
30	0.0116	0.0124	1.0111	0.0228
75	0.008	0.0117	1.0258	0.0255

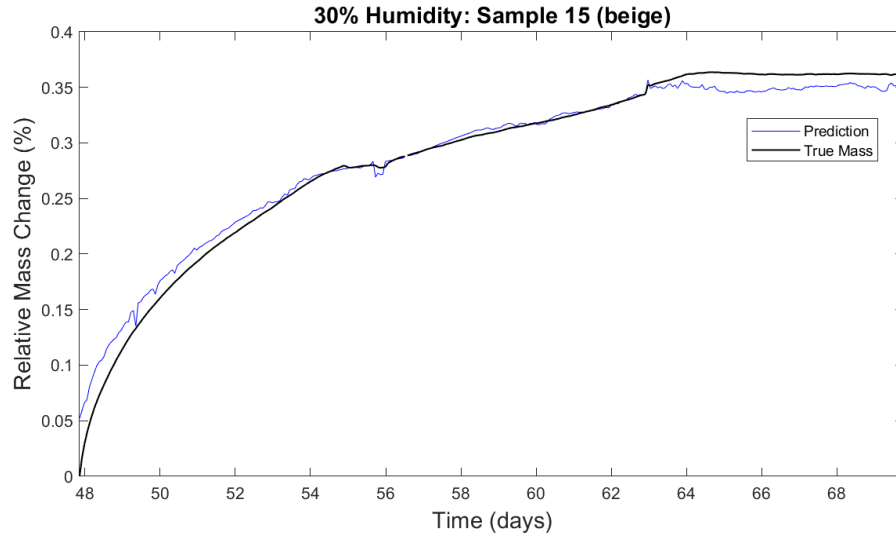


Figure 4.12 Results for the 30% humidity mass curve of beige sample #15 using the model trained on data from beige sample #4. As #15 is subjected to near similar conditions as #4, the model matches well to unseen data from a different sample, with robustness against small deviations in the inputs that exist between both samples.

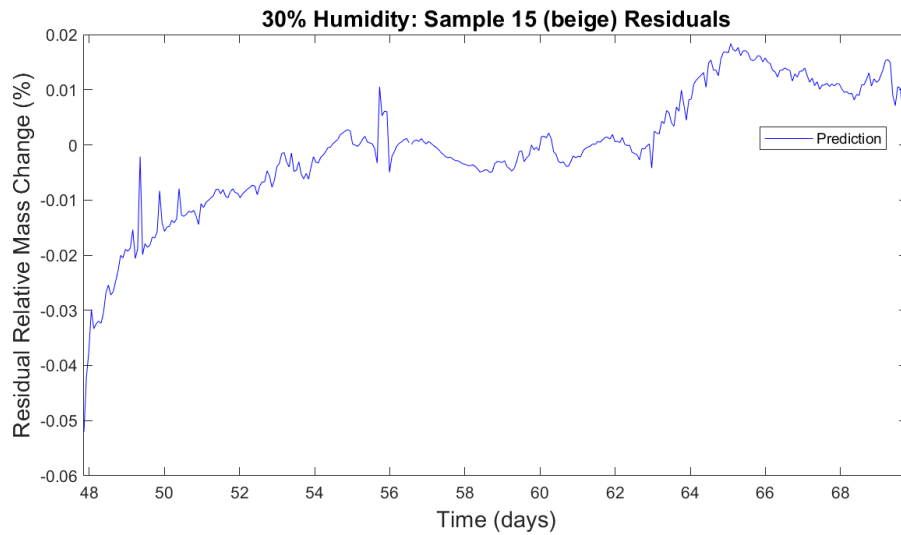


Figure 4.13 Residuals for the 30% humidity mass curve of beige sample #15 using the model trained on data from beige sample #4. Most of the prediction is within $\pm 0.02\%$ of the data with the most deviation at the beginning only reaching up to 0.05%.

similar and arguably better degree as the model with sample #4 from which it was trained. While the conditions for which sample #4 and #15 are subjected to are approximately similar when measurements are taken, as seen with similar evolutions of the mass curve, the model is robust enough against small deviations in the inputs to allow for a consistent and accurate prediction for another sample. This would allow the model to likely make accurate predictions for the beige polyurethane foam type at the trained humidity levels, including other small complex shapes besides the cylinder as the equation fit for a one-dimensional slab describes the mass change of a three-dimensional object surprisingly well. This generalization should be applicable to other humidity levels assuming that additional data is collected for foam samples at a future time to increase the training data.

A final test of training was performed for sample #4 by creating models for each humidity mass curve individually instead of combining all four into one training. When not subject to a large degree of varying data, the models work extremely well albeit not under ideal conditions for creating a usable product. The results of the trainings can be seen in **Appendix Figures 33-36**.

Gray Polyurethane Foam

For the purposes of a proof of concept for the PGNN in mass change from diffusion, there was little experimentation utilizing the gray polyurethane foam as the majority of testing was performed with the beige foam. It was also assumed that once an ideal architecture was finalized for the beige foam, the same architecture would theoretically be a good fit for the gray foam training. Comparisons of the mass change data between the foam samples indicate that the gray foam experienced a smaller degree of mass change than the beige foam under the same conditions due to material differences including the density; the density of the beige foam is 0.641 g/cm^3 [2] while the gray foam is 0.801 g/cm^3 [3], decreasing the rate of diffusion through the material. Due to the differences in mass change as a function of material property, the model trained on the beige foam would predict values significantly different for the gray foam given the same input values. The following details the short experimentation of training a model for the gray foam samples.

Model Architecture from Beige Samples

The successes of the architecture designed for the beige samples through many trials indicated that the same architecture would have a similar degree of success with the gray foam due to their co-evolution of the mass change. The same number of layers and neurons were utilized with the training at the same number of 5000 epochs on the data of gray sample #7. While the constants of the equation should have different values compared to the beige foam due to material differences, the initial values were kept the same as the order of magnitude would be similar and the model ideally converges to the correct value. The results of the training for the 5% and 75% humidity groups can be seen in **Figures 4.14–4.17** with the remaining plots seen in **Appendix Figures 37–40**. Overall, the new model trained for the gray samples has good fidelity with the mass change data similar to the model trained for the beige sample. Some differences are apparent however, especially with the 5% humidity group, where the model performs arguably worse than the model for the beige foam with diverging predictions and unphysical spikes in mass. This is likely due to the NN architecture being unoptimized for the gray foam sample compared to the beige sample. Due to the gray sample being at a higher density, the overall amount of water absorption is less than that of the beige foam samples and correspondingly leads to a lesser degree of mass change. Due to smaller mass changes such as in the 5% humidity group, the NN architecture optimized for the beige foam may be unable to resolve the data for the gray foam without further changes to the layers, weights and biases, or other parameters. As the majority of the work developing the PGNN was performed with the beige samples, there was little time to optimize the model for the gray samples within the scope of this study and would require future research. While the PGNN had difficulties with the 5% humidity group in the gray samples, the optimization of the constants in the physics equations had no issues in learning physical values to match the equation to the mass change to a large degree as seen in **Figure 4.18** For the 5% humidity group, with the remaining groups seen in **Appendix Figures 41–43**. It can be seen that the equation provides a close fit to the mass change data with the optimized constants, indicating the effectiveness of the model to optimize the equation but also highlighting the degraded performance of the PGNN in

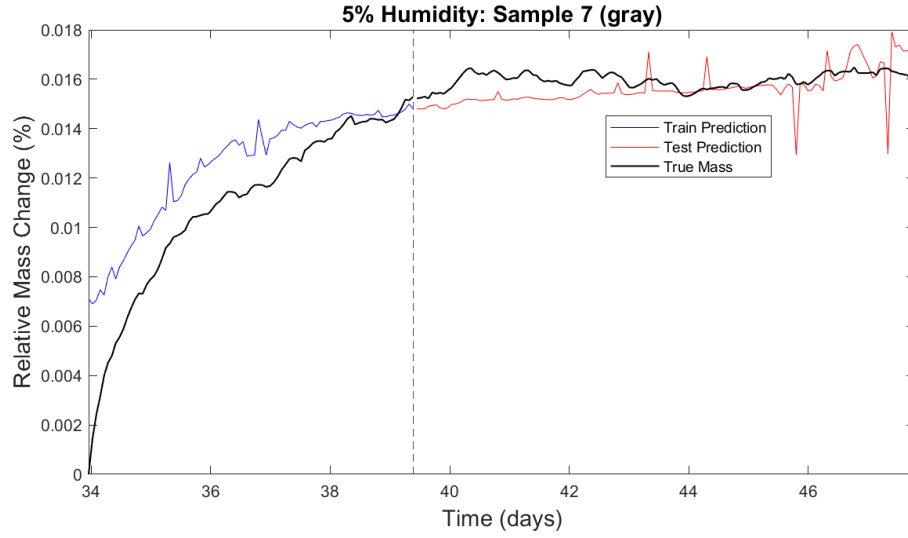


Figure 4.14 Results for the 5% humidity mass curve of gray sample #7 for the fully-trained model on all curves using 40% of the data for training. The predictions for this group suffer from a general lack of fidelity in the training data and from unphysical spikes. This is likely due to an unoptimized NN architecture for this foam type being unable to precisely predict the very small mass changes at this humidity level or confusion between the close RBF centers of the 0% and 5% groups.

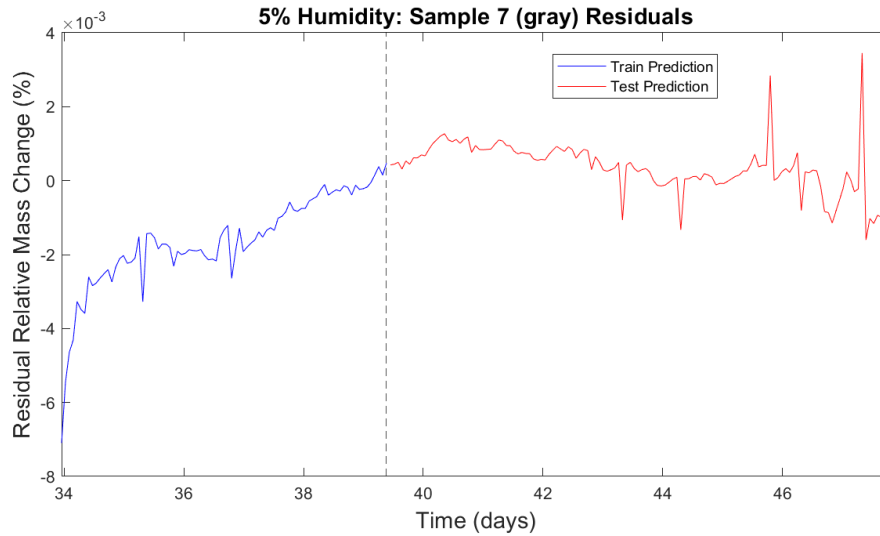


Figure 4.15 Residuals for the 5% humidity mass curve of gray sample #7. The middle time section differs only by about $\pm 0.001\%$, but suffers from unphysical spikes and a diverging difference at the beginning time up to -0.008 .

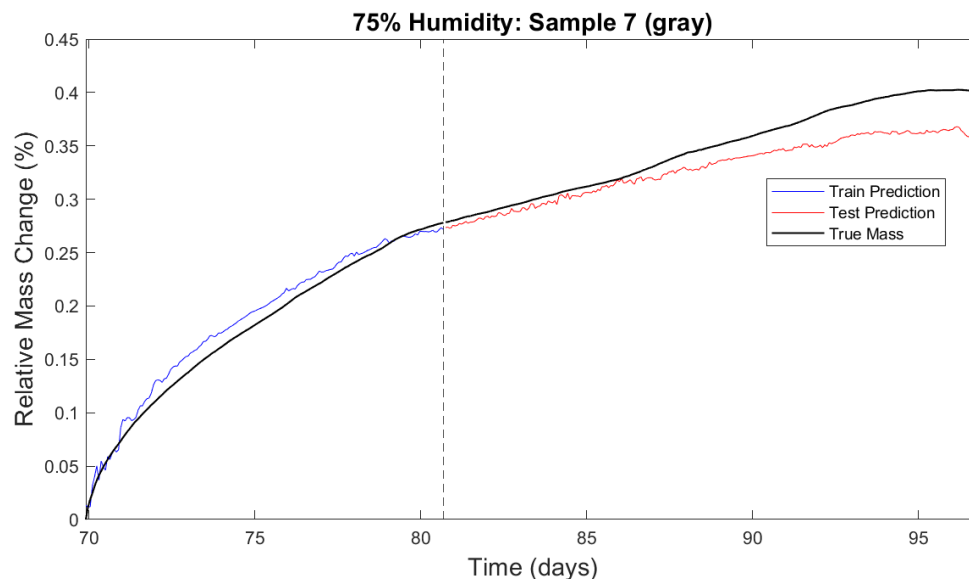


Figure 4.16 Results for the 75% humidity mass curve of gray sample #7 for the fully-trained model on all curves using 40% of the data for training. The predictions for this group are generally of a high fidelity as with the beige sample.

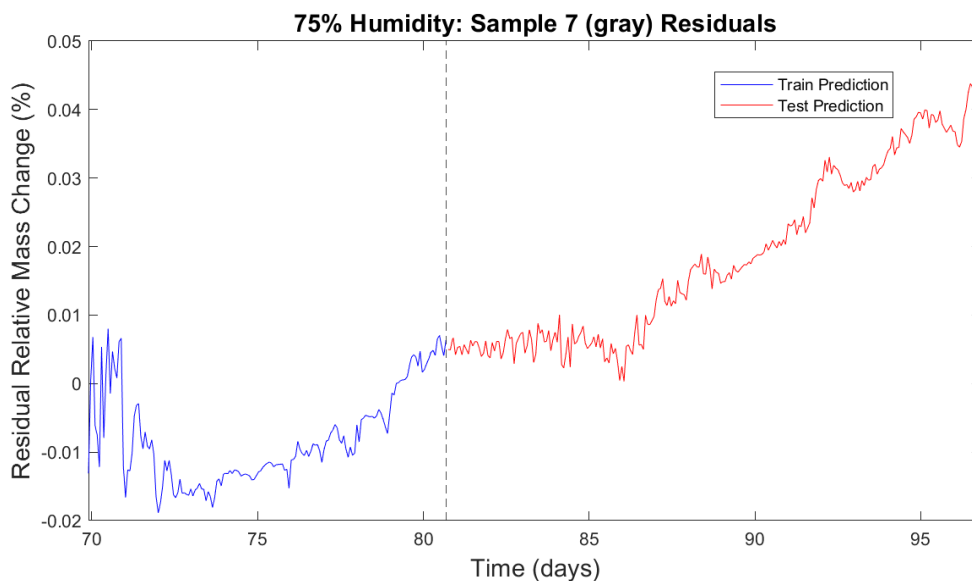


Figure 4.17 Residuals for the 75% humidity mass curve of gray sample #7. The results are of high fidelity, within $\pm 0.02\%$ for most of the data and only reaching almost 0.05% for the late time prediction.

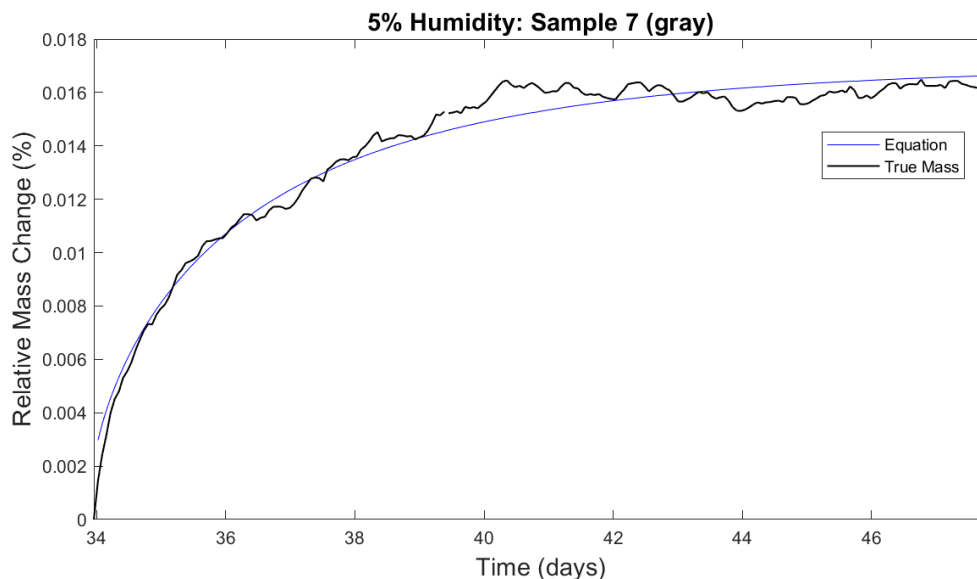


Figure 4.18 The one-dimensional diffusion equation with optimized parameters against the mass change data for the 5% humidity group of gray sample #7. The equation has good fidelity with the curve, indicating the parameters match well to the physical system regardless of the errors the PGNN predicts in **Figure 4.14**.

predictions; the discussion on this dichotomy is the same as above for the beige foam. The optimized constants for the gray foam can be seen in **Table 4.3**. While the parameters are overall different than the beige foam constants due to physical differences, the parameter of interest when comparing the two is that of the diffusivity per area. Overall, the values for this parameter are generally smaller than that of the beige foam, which corresponds to a longer diffusion time due to the inverse time units. This matches expectations that the diffusion for the higher density gray foam would have a lesser degree of diffusion than that of the beige foam, indicating that the PGNN is able to converge onto physically meaningful and potentially correct constants describing the systems.

Table 4.3 Values for the constants from the model trained on gray sample #7. The constants optimize differently than those of the beige samples in **Tables 4.1** and **4.2**, but match well to the mass change data. In terms of physical parameters, the diffusivity per area is generally lower across all levels compared to the beige samples, indicating a slower rate of diffusion; this matches expectation as the higher density gray foam would experience a lesser degree of diffusion.

Humidity Curve (%)	D_x/s^2 (<i>days</i>⁻¹)	a	b	M_i (%)
0	0.0084	0.0001	2.2363	0.2003
5	0.0285	0.0101	0.3249	0.0020
30	0.008	0.0431	0.4213	0.0061
75	0.0064	0.0605	0.460	0.0023

CHAPTER FIVE

CONCLUSIONS

The experimentation and trials of the PGNN has highlighted many promising successes as well as challenges in its use for mass change prediction from water diffusion. The following describes conclusions on the results of training, complications to overcome, and proposed future work to improve the fidelity and usability of the model.

Successful Results

Overall, the PGNN performed admirably in the prediction of future data not utilized as part of training for the traditional NN portion of the model. For many results of the beige sample testing and somewhat for the gray sample testing, the predictions made for the later 60% of data not trained upon remained within 0.1% or less of the true mass change and generally follows the evolution of curve including the asymptote at the saturation point. The fidelity between the model and the data improves to a larger degree when only considering the non-zero humidity groups or each group separately due to the reduction in varying mass data for similar inputs, complications for which techniques such as the radial basis function sought to eliminate by creating additional inputs to distinguish the different humidity levels.

For the determination of constants, the PGNN showcases a strength in converging to optimal solutions that help best fit the equation to the data. From four constants that required optimization due to unknown values a priori, the model gave predictions that produce an equation fit matching well to the data as seen in results described previously, including from initial values that were an order of magnitude different than was determined via curve fitting and research [20] [39] [40] [41]. For the diffusion coefficient per area, the values predicted are physically consistent with approximations of the ranges made for other polyurethane materials around 0.01 days^{-1} assuming unit area, and correlate to the relative strengths of the two foam materials; the gray foam with a higher density and lower mass change appropriately has a smaller predicted diffusion coefficient per area than the beige foam. While there is a weakness in the optimization in that the full data is required to obtain accurate constants, the model cannot optimize on just 40% of

the initial data, it showcases the dual ability of the model to optimize the weights and biases as well as parameters of the equation.

Complications

For its improvements in the prediction of untrained data compared to a standard NN, there are several complications preventing it from achieving the highest level of fidelity on all data. The most significant of these is the apparent struggle of the model to make predictions that follow the data for the 5% humidity group generally compared to the other humidity groups, especially for the gray foam samples. The predictions on this group of data suffers from both a visual divergence from the data curve in both the training near the initial mass and test portions as well as more unphysical spikes in the mass, even though the average deviation was about 0.02%. Likely causes stem from both the computational and physical domains, with the former attributed to the extremely small mass changes over the time period causing the model to react to small changes in the inputs or conflicting data from close RBF centers for 0% and 5% and the later attributed to potential diffusion effects not accounted for in the model creating difficulty for the equation to match to the data for some samples. Deviation at the initial mass could also be from forcing the starting mass to be 0 for all curves, making any $M_i > 0$ in the model lead to a slight divergence at the initial points of the data.

Another complication can be seen in the loss of accuracy of the model for the data on which it was trained. In comparing results for the 30% humidity group such as in **Appendix Figures 9-10**, it can be seen that the addition of the physics loss function creates a deviation from the true mass in the training data that would not be seen in a traditional NN such as for the 75% humidity group example in **Figure 4.1**. This effect is more confusing when comparing the curve of the equation with optimized parameters to the 30% humidity mass curve in **Appendix Figure 12**, where the equation decently fits the mass change and should provide additional fidelity instead of hindering the data portion of the PGNN. It is assumed that during the course of training, the inaccurate constants in the process of updating are used in driving the physics loss of the PGNN, leading to the model training on an inaccurate equation of the data early before it is optimized; the early training may have a memory effect on the weights and biases

causing them to not optimize fully. This could potentially be corrected by ignoring the contribution of the equation to the solution on the training data, instead using equation for constraining the untrained future time portions.

A final complication related to the commercial use of the PGNN in the prediction of the mass of foam is the controlled glovebox experiment being unrepresentative of the natural environment. The glovebox, although having some fluctuation, maintained relatively steady levels of pressure, temperature, and humidity for the value at which it was set. This allowed the foam to maintain Fickian diffusion for a given humidity level evident with curves matching other diffusion studies [13] [15] [16] [17] [18]. The process of diffusion is heavily dependent on the temperature [13] [18], making it so that natural daily changes would likely have a significant impact on the future mass, and on humidity changes that do not remain naturally steady over the approximately two weeks that the glovebox held foam samples at. These factors paired with barometric pressure and other confounding factors provide additional challenges in using a model trained under controlled conditions for natural scenario prediction.

Future Work

The PGNN can be improved in several ways to increase fidelity and make a more generalizable model for practical use. The first improvement stems in the architecture defining the model. To account for issues in the weights and biases, a finer search of the number of layers and neurons per layer that optimize the training can be conducted utilizing a methodical search. The same methodical search can be applied to the hyperparameters controlling the system, including the learning rate, decay rate, number of epochs, number in a minibatch, and the weighting factors in the loss function, but would require many combinations and time without the implementation of an automatic search; perhaps a genetic algorithm would aid in this work.

From the perspective of physics, the mass change from diffusion was modeled from the solution for a one-dimensional slab [13] and provides a decently accurate representation. However, such modeling could potentially be improved by considering a three-dimensional solution that requires additional mathematics but theoretically should match the conditions of diffusion for the half-hollow cylinder studied. Additional physics

including a study of more complex diffusion and other extraneous factors could provide further aid.

From the perspective of data, acquiring more mass measurements at many other humidity levels and varying temperatures can help provide a foundation for building a generalizable model for any humidity level and temperature affecting the rate of diffusion. Such a model would require past memory of the conditions of the foam to predict to future time and additional physics to account for changes to constants over time such as the diffusivity per area, which is heavily dependent on temperature [13] [18]. Additional data could potentially be collected in a more “natural” setting that subjects the foam to environmental humidity and temperature over a long period of time.

Finally, this model could be extended from the realm of mass change from water to volume change from swelling and contraction. For the foam acting as proxy parts in mechanical systems requiring precise volume to fit within the whole of the parts, knowledge of how humidity causing swelling would be important to predict. This extension would require three-dimensional knowledge of the water concentration and properties of the foam that allow swelling to occur.

Although it can be arguably reasoned that the predictions of the equation using optimized constants is preferable for future mass prediction than the PGNN when comparing the differences between the outputs of each against the mass, a PGNN can offer substantial advantages given a careful architecture. With the constants unknown, a PGNN is able to optimize the values during the course of training its predictions instead of relying on a fitted curve, which would be advantageous for real-time data analysis. Another issue with the equation is the reliance on a steady-state humidity over the course of each mass change, an unrealistic outcome in applications that could potentially be corrected through the use of ML with extensive data constrained by equations to make physically consistent predictions and allow greater control over the materials in their respective capacities.

REFERENCES

- [1] Stratasys, "Surrogate Parts for Design, Manufacturing, Training, and Support," 2015. [Online]. Available: https://aerospace.emea.stratasys.com/assets/downloads/part-verification/Surrogate_Parts_Technical_Application_Guide.pdf. [Accessed 24 February 2025].
- [2] General Plastics Manufacturing Company, *Last-a-Foam FR-3700 Rigid Polyurethane Foam*, Tacoma, WA, 2018.
- [3] General Plastics Manufacturing Company, *Last-a-Foam FR-4500 Rigid Polyurethane*, Tacoma, WA, 2019.
- [4] Edgetech Instruments Inc., *DewMaster Chilled Mirror Hygrometer*, Hudson, MA.
- [5] FANUC America, *LR Mate 200iD/4S Robot*, Rochester Hills, MI, 2017.
- [6] Mettler Toledo International Inc., *Balance XPE 205*, Columbus, OH, 2018.
- [7] Mettler Toledo International Inc., *U-Electrode*, Columbus.
- [8] K. Poulinakis, D. Drikakis, I. W. Kokkinakis and S. M. Spottswood, "Machine-learning methods on noisy and sparse data," *Mathematics*, vol. 11, no. 1, p. 236, 2023.
- [9] A. Daw, A. Karpatne, W. Watkins, J. Read and V. Kumar, "Physics-guided Neural Networks (PGNN): An Application in Lake Temperature Modeling," ArXiv, 2021.
- [10] A. Fick, "On liquid diffusion," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 10, no. 63, pp. 30-39, 1855.
- [11] D. Li, B. Yan, T. Gao, G. Li and Y. Wang, "PINN Model of Diffusion Coefficient Identification Problem in Fick's Laws," *ACS Omega*, vol. 9, no. 3, pp. 3846-3857, 2024.
- [12] W. Jost, *Diffusion in Solids, Liquids, Gases*, Academic Press, 1960.
- [13] C.-H. Shen and G. S. Springer, "Moisture Absorption and Desorption of Composite Materials," *Journal of Composite Materials*, vol. 10, p. 2, 1976.

- [14] J. Crank, *The Mathematics of Diffusion*, 2nd ed., Oxford: Clarendon Press, 1975.
- [15] D. Gunn, D. Moores, W. Thomas and A. Wardle, "Sorption of Water Vapour Into Polyurethane Foams," *Chemical Engineering Science*, vol. 29, pp. 549-559, 1974.
- [16] Y. J. Weitsman, "2.11 Effects of Fluids on Polymeric Composites - A Review," in *Comprehensive Composite Materials*, Elsevier, 2000, pp. 369-401.
- [17] Z. Huo, M. Mohamed, J. R. Nicholas, X. Wang and K. Chandrashekhara, "Experimentation and simulation of moisture diffusion in foam-cored polyurethane sandwich structure," *Journal of Sandwich Structures & Materials*, vol. 18, no. 1, pp. 30-49, 2016.
- [18] K. Jansen, M. Zhang, L. Ernst, D.-K. Vu and L. Weiss, "Effect of temperature and humidity on moisture diffusion in an epoxy moulding compound material," *Microelectronics Reliability*, vol. 107, p. 113596, 2020.
- [19] Y. Yi, Y. Sun, L. Wang and P. Xiao, "Impacts of moisture absorption on electrical properties of rigid polyurethane foam for composite post insulator of UHVDC transmission line," *Electrical Power and Energy Systems*, vol. 131, p. 107098, 2021.
- [20] D. Zhou and P. Choi, "Molecular dynamics study of water diffusivity at low concentrations in non-swollen and swollen polyurethanes," *Polymer*, vol. 53, pp. 3253-3260, 2012.
- [21] C. Wang, Z. Xu, Y. Xia, C. Zhang, H. Fang and K. Sun, "Water transport mechanism and performance evaluation in polyurethane materials: A state-of-the-art review," *Polymer Testing*, vol. 138, p. 108554, 2024.
- [22] M. J. Adamson, "Thermal expansion and swelling of cured epoxy resin used in graphite/epoxy composite materials," *Journal of Materials Science*, vol. 15, pp. 1736-1745, 1980.
- [23] C. P. Batuwatta-Gamage, C. M. Rathnayaka, H. C. P. Karunasena, W. D. C. C. Wijerathne, H. Jeong, Z. G. Welsh, M. A. Karim and Y. T. Gu, "A physics-informed neural network-based surrogate framework to predict moisture

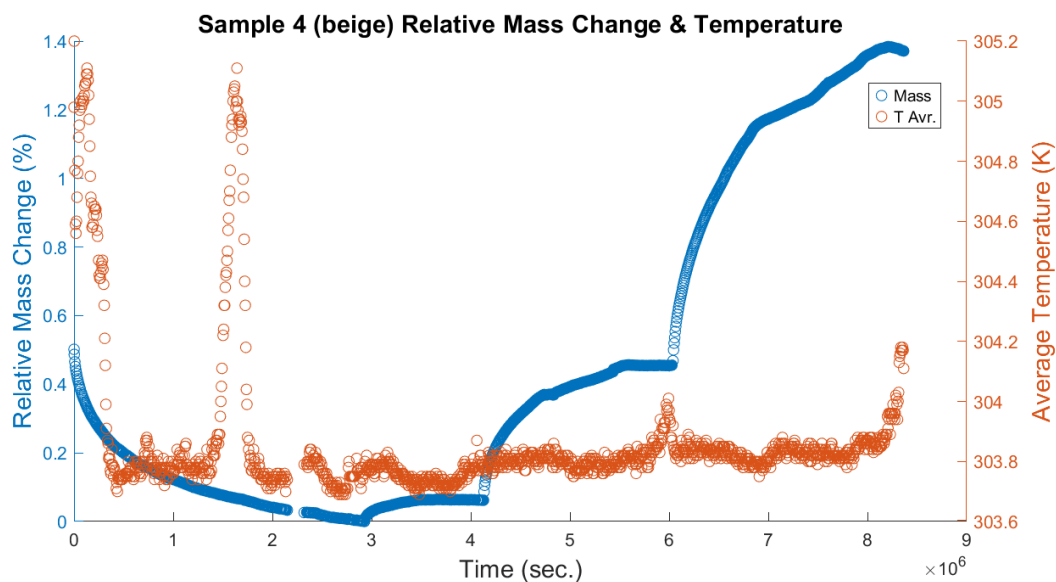
concentration and shrinkage of a plant cell during drying," *Journal of Food Engineering*, vol. 332, p. 111137, 2022.

- [24] A. M. Tartakovsky, C. O. Marrero, P. Perdikaris, G. D. Tartakovsky and D. Barajas-Solano, "Physics-informed deep neural networks for learning parameters and constitutive relationships in subsurface flow problems," *Water Resources Research*, vol. 56, no. 5, 2020.
- [25] R. Xu, N. Wang and D. Zhang, "Solution of diffusivity equations with local sources/sinks and surrogate modeling using weak form theory-guided neural network," *Advances in Water Resources*, vol. 153, p. 103941, 2021.
- [26] T. Praditia, Karlbauer, M., S. Otte, S. Oladyshkin, M. V. Butz and W. Nowak, "Learning groundwater contaminant diffusion-sorption processes with a finite volume neural network," *Water Resources Research*, vol. 58, no. 12, 2022.
- [27] K. Kobayashi and S. B. Alam, "Physics-regularized neural networks for predictive modeling of silicon carbide swelling with limited experimental data," *Scientific Reports*, vol. 14, p. 30666, 2024.
- [28] E. Gallup, T. Gallup and K. Powell, "Physics-guided neural networks with engineering domain knowledge for hybrid process modeling," *Computers & Chemical Engineering*, vol. 170, p. 108111, 2023.
- [29] H. Robinson, S. Pawar, A. Rasheed and O. San, "Physics guided neural networks for modelling of non-linear dynamics," *Neural Networks*, vol. 154, pp. 333-345, 2022.
- [30] X. Hu, H. Hu, S. Verma and Z. L. Zhang, "Physics-guided deep neural networks for power flow analysis," *IEEE Transactions on Power Systems*, vol. 36, no. 3, pp. 2082-2092, 2020.
- [31] Z. Huang, X. Yin and Y. Liu, "Physics-guided deep neural network for structural damage identification," *Ocean Engineering*, vol. 260, p. 112073, 2022.
- [32] W. Li, M. Z. Bazant and J. Zhu, "A physics-guided neural network framework for elastic plates: Comparison of governing equations-based and energy-based

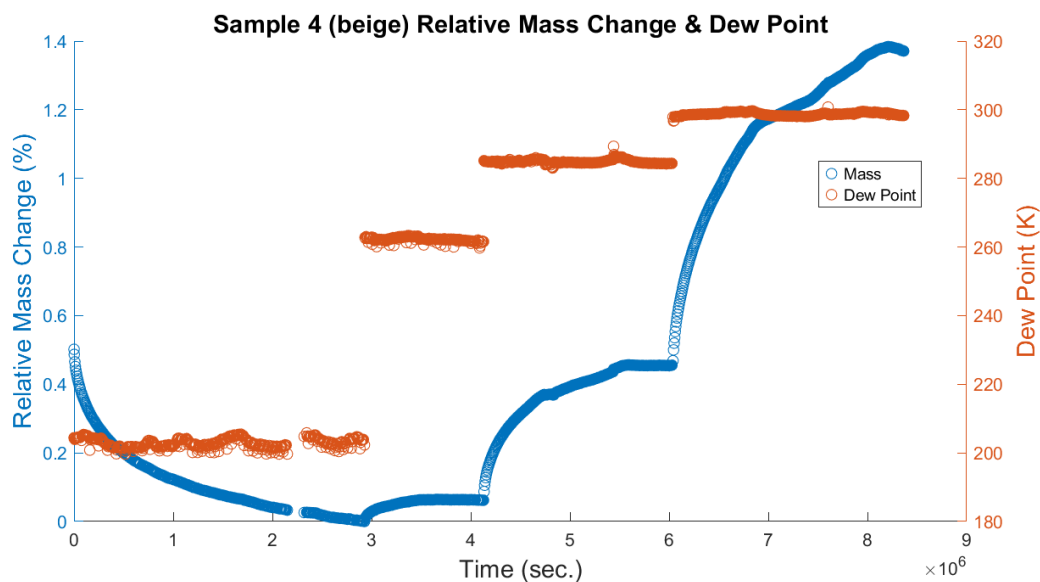
- approaches," *Computer Methods in Applied Mechanics and Engineering*, vol. 383, p. 113933, 2021.
- [33] J. Wang, Y. Li, R. Zhao and R. X. Gao, "Physics guided neural network for machining tool wear prediction," *Journal of Manufacturing Systems*, vol. 57, pp. 298-310, 2020.
- [34] A. Carter, S. Imtiaz and G. Naterer, "Imperfect physics-guided neural networks," *Chemical Engineering Science*, vol. 305, p. 121153, 2025.
- [35] The MathWorks, Inc., *MATLAB version: 24.1.0.2628055 (R2024a)*, Natick, MA, 2024.
- [36] M. Shanker, M. Y. Hu and M. S. Hung, "Effect of data standardization on neural network training," *Omega*, vol. 24, no. 4, pp. 385-397, 1996.
- [37] S. Janas and S. Karpisz, "Buoyancy Force In Mass Measurement," *Radweg Balances and Scales*, 2014.
- [38] G. L. Harris, "Selected Laboratory and Measurement Practices and Procedures to Support Basic Mass Calibrations," National Institute of Standards and Technology, 2019.
- [39] Y. Zhao, J. S. Taylor and S. Chellam, "Predicting RO/NF water quality by modified solution diffusion model and artificial neural networks," *Journal of Membrane Science*, vol. 263, no. 1-2, pp. 38-46, 2005.
- [40] J. Huacuja-Sanchez, K. Muller and W. Possart, "Water diffusion in a crosslinked polyether-based polyurethane adhesive," *International Journal of Adhesion & Adhesive*, vol. 66, pp. 167-175, 2016.
- [41] A. Sabbahi and J. M. Vergnaud, "Absorption of water by polyurethane foam. modelling and experiments," *European Polymer Journal*, vol. 29, no. 9, pp. 1243-1246, 1993.
- [42] D. P. Kingma and J. Ba, "Adam: A Method of Stochastic Optimization," in *3rd International Conference for Learning Representations*, San Diego, 2015.

- [43] D. Lowe and D. Broomhead, "Multivariable functional interpolation and adaptive networks," *Complex Systems*, vol. 2, no. 3, pp. 321-355, 1988.
- [44] C. Cortes, M. Mohri and A. Rostamizadeh, "L2 Regularization for Learning Kernels," in *25th Conference on Uncertainty in Artificial Intelligence*, Montreal, 2009.

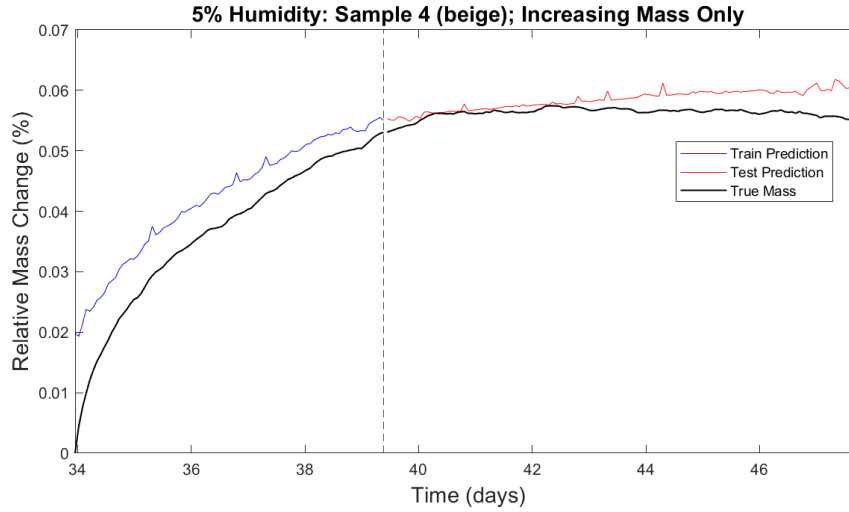
APPENDIX



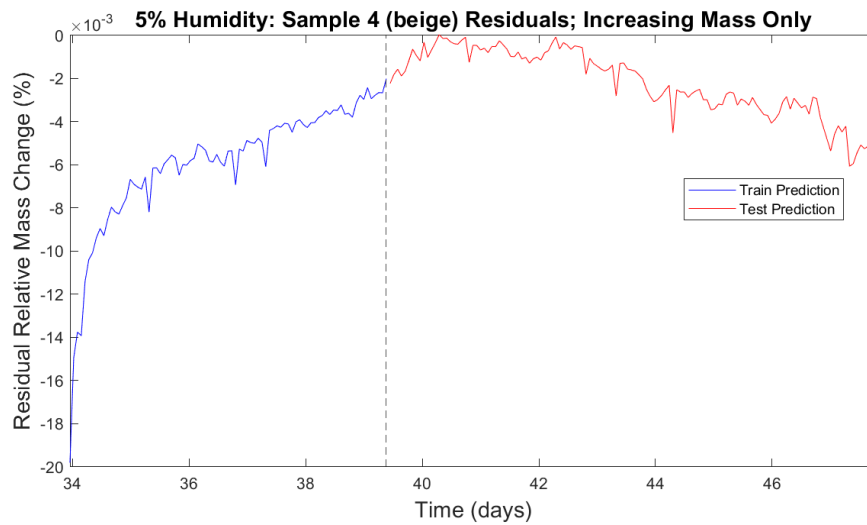
Appendix Figure 1 Data collected for Sample #4, a beige polyurethane foam, showing the relative mass change and the temperature in Kelvin over time. The relative point for the mass is the minimum value of all masses for the sample.



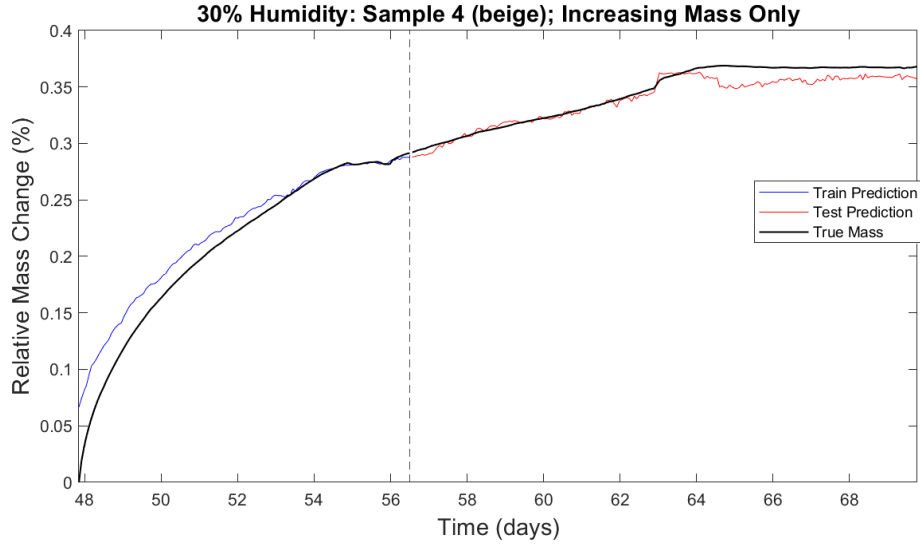
Appendix Figure 2 Data collected for Sample #4, a beige polyurethane foam, showing the relative mass change and the dew point in Kelvin over time. The relative point for the mass is the minimum value of all masses for the sample.



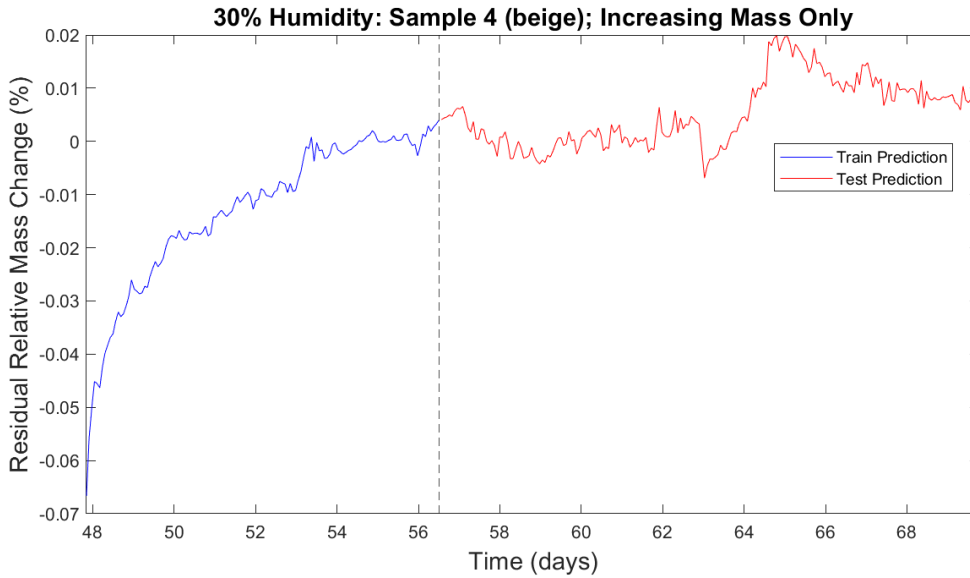
Appendix Figure 3 The results for the 5% humidity mass curve for the model trained on the three curves of increasing mass of beige sample #4. All curves were trained using 40% of the data for the traditional NN with the physics loss function retaining a higher influence on the outcome. The predictions are not as closely matched as the other curves likely due to the small mass increase paired with small input changes creating confusion for the model to separate well, but still offers a relatively close prediction.



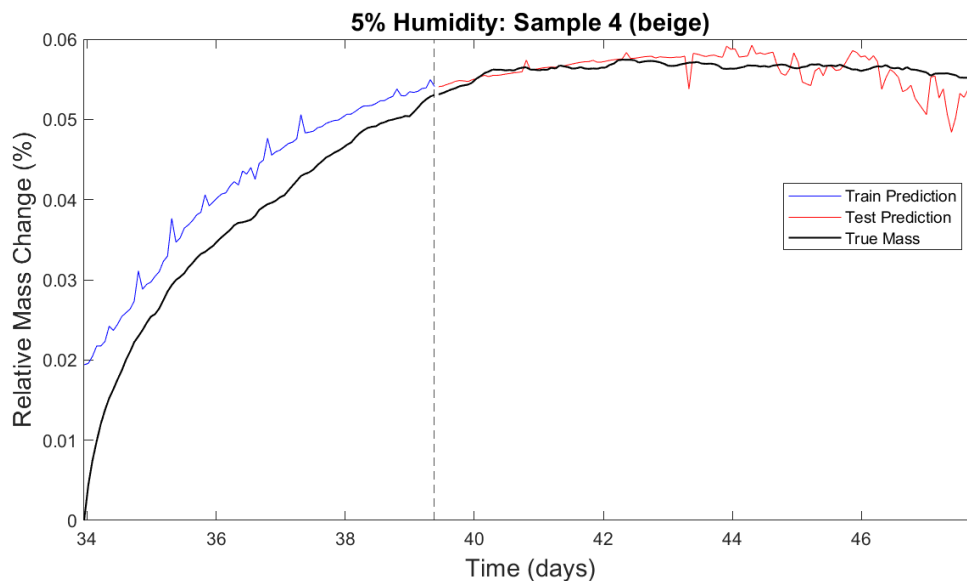
Appendix Figure 4 Residuals of the 5% humidity mass curve for the model trained on three curves of increasing mass of beige sample #4. The relative mass change experiences more deviations than the other humidity mass curves but at most differs by about 0.02% and generally overpredicts the mass. This is likely due to the small mass change causing difficulty for the NN to approximate.



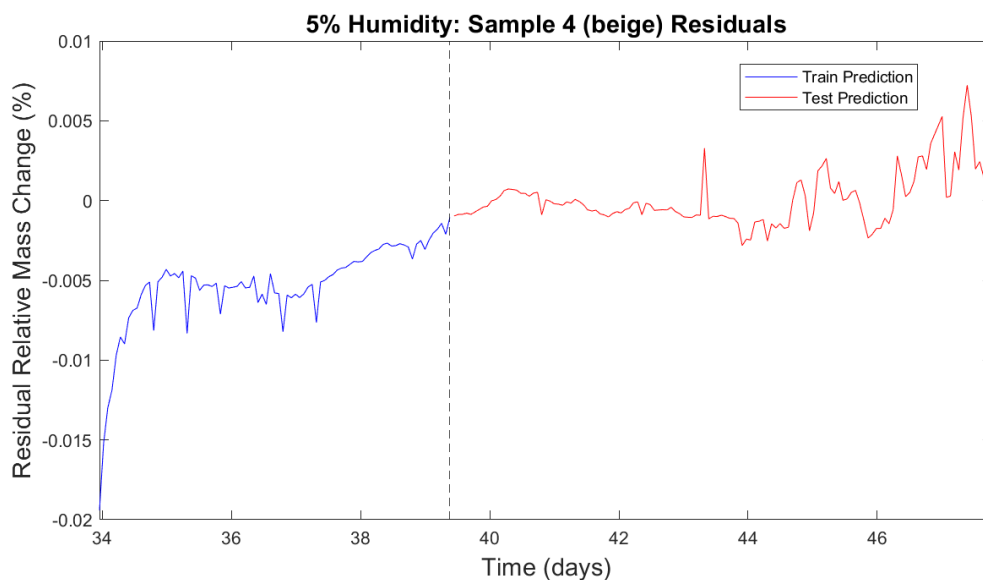
Appendix Figure 5 The results for the 30% humidity mass curve for the model trained on the three curves of increasing mass of beige sample #4. All curves were trained using 40% of the data for the traditional NN with the physics loss function retaining a higher influence on the outcome. The predictions closely match the actual mass curve.



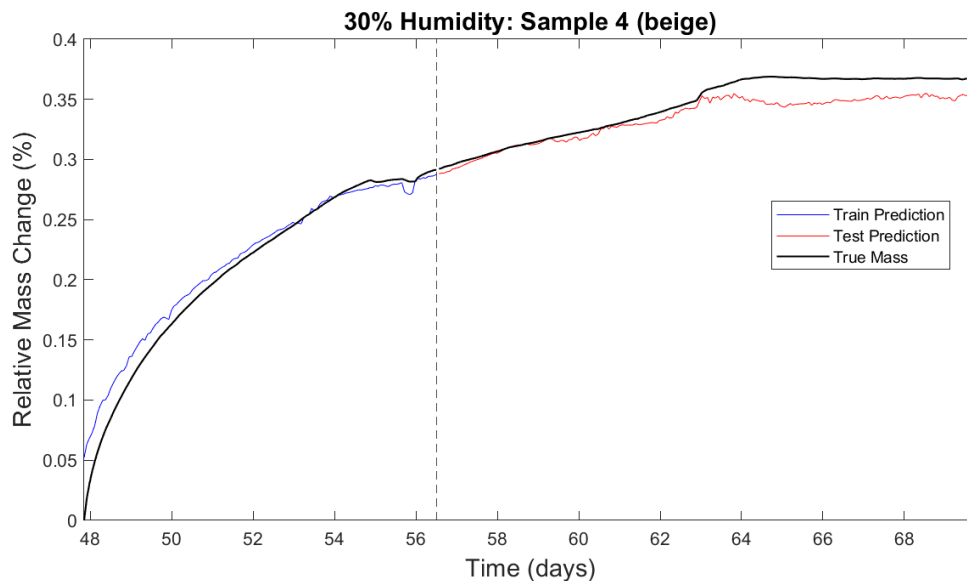
Appendix Figure 6 Residuals of the 30% humidity mass curve of beige sample #4 for the model trained on three curves of increasing mass. The relative mass change differs by approximately 0.02% from zero for most of the time with the most deviation at the beginning up to 0.07% different. Overall, it represents a high-fidelity prediction.



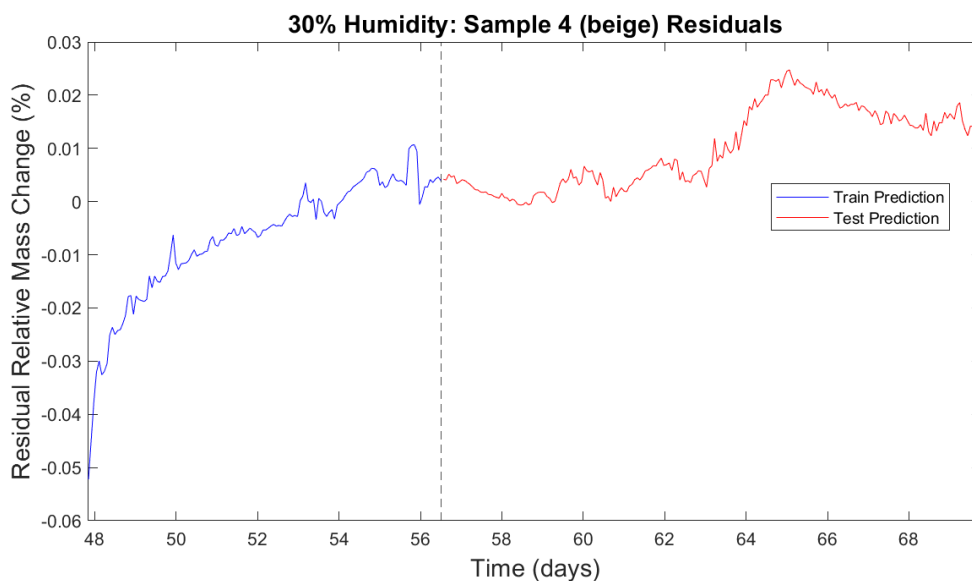
Appendix Figure 7 Results for the 5% humidity mass curve of beige sample #4 trained on the full data set. The predictions have similar results to **Appendix Figure 3** trained on the increasing mass only, but is also improved on the predictions of the test data.



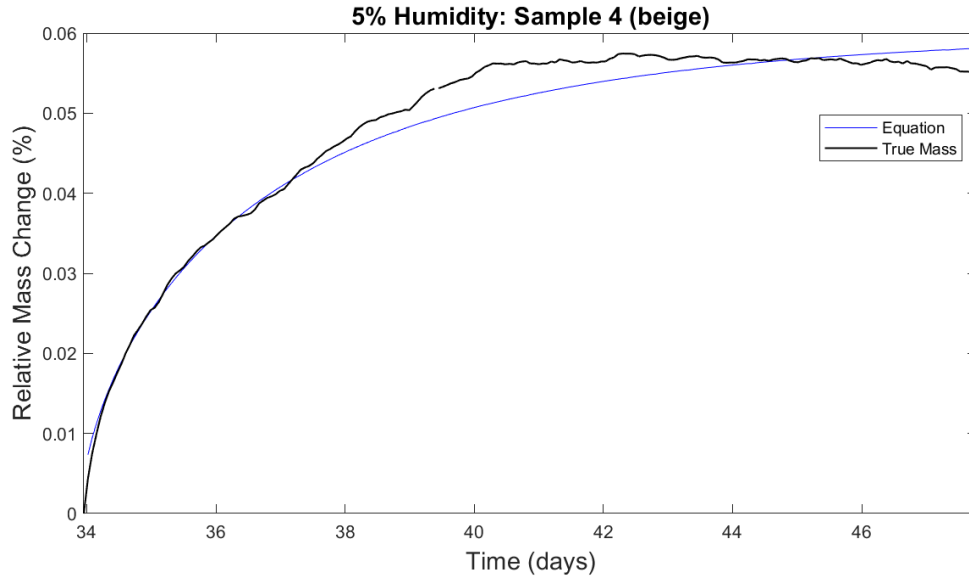
Appendix Figure 8 Residuals for the 5% humidity mass curve of beige sample #4 trained on the full data set. The predictions are within $\pm 0.005\%$ of the data with the most deviation at the early time up to -0.02% .



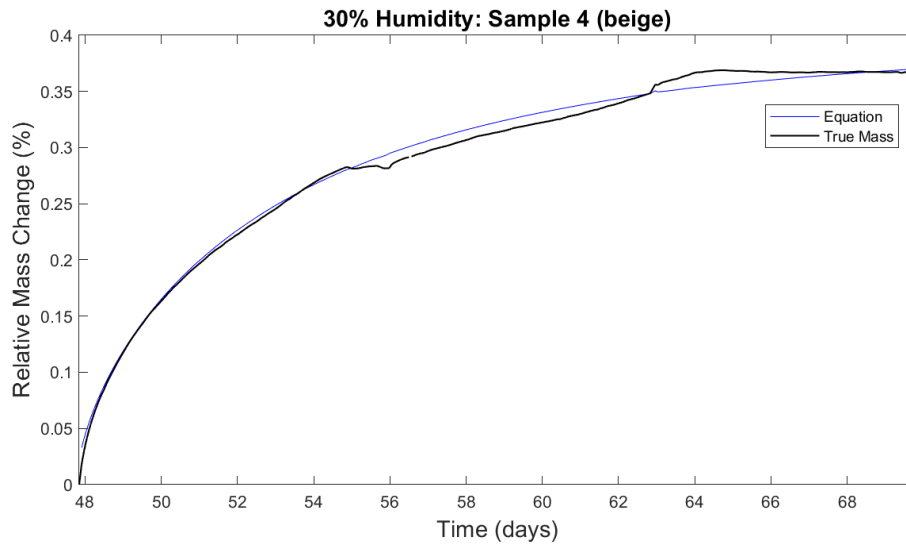
Appendix Figure 9 Results for the 30% humidity mass curve of beige sample #4 trained on the full data set. The predictions have similar results to **Appendix Figure 5** trained on the increasing mass only.



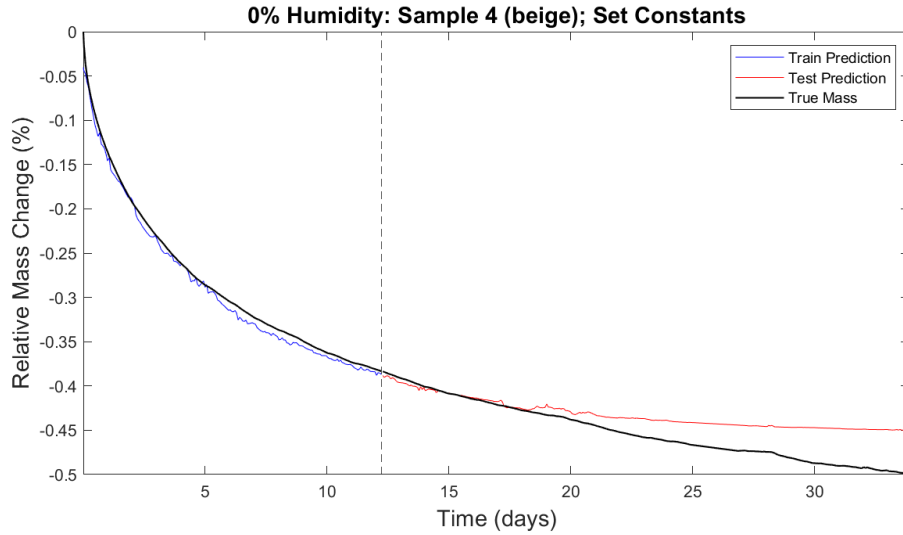
Appendix Figure 10 Residuals for the 30% humidity mass curve of beige sample #4 trained on the full data set. The predictions are within $\pm 0.02\%$ of the data with the most deviation at the early time up to -0.05% .



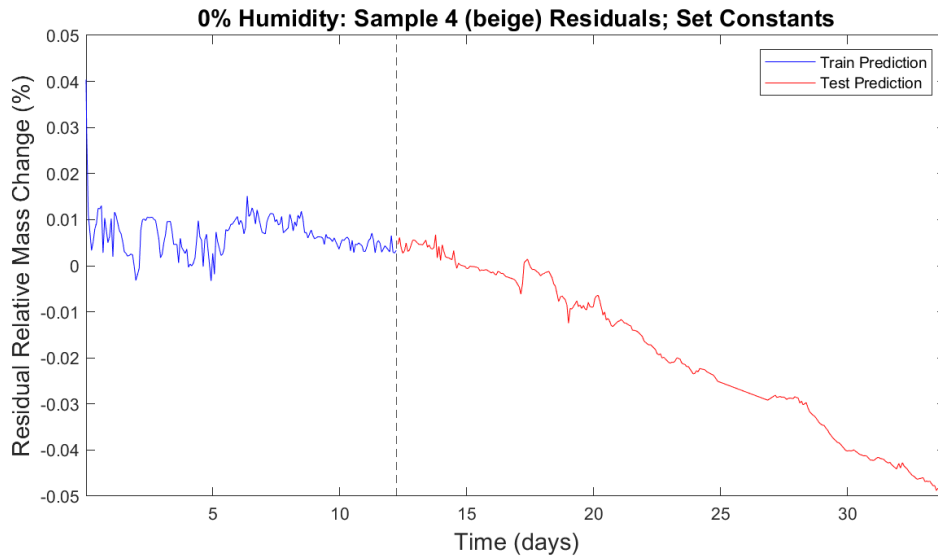
Appendix Figure 11 The predictions on the 5% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of beige sample #4. There is overall good fidelity, indicating validity of the equation under complex conditions and the ability of the model to determine physical constants from the data. The values can be seen in **Table 4.1**.



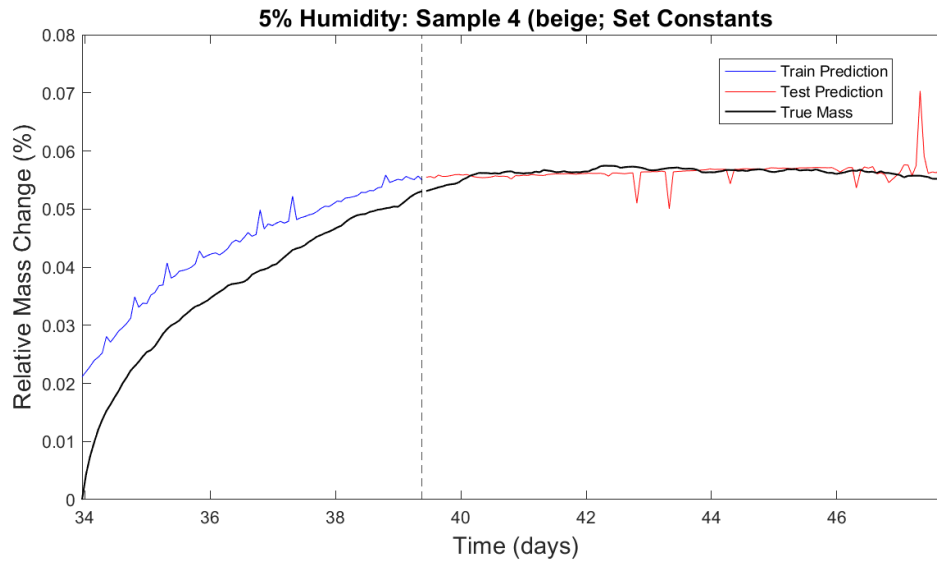
Appendix Figure 12 The predictions on the 30% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of beige sample #4. The fidelity is extremely well, indicating validity of the equation under complex conditions and the ability of the model to determine physical constants from the data. The values can be seen in **Table 4.1**.



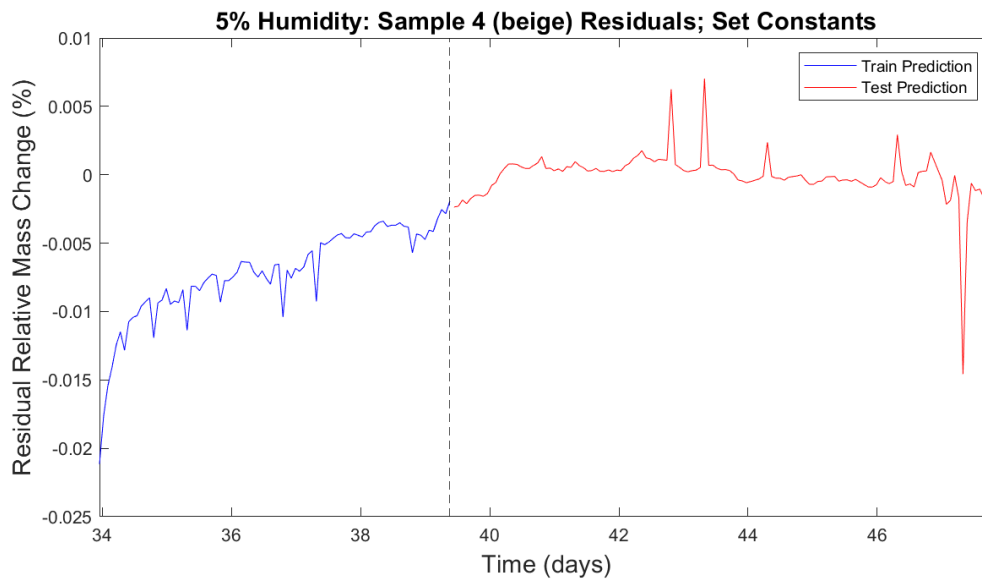
Appendix Figure 13 Results for the 0% humidity mass curve for the model trained on all data of beige sample #4 with set constants. The predictions appear to have a higher fidelity than the original predictions in **Figure 4.4**, indicating that the PGNN had a better training when the equation was optimized and supporting the assumption that the learning of the constants led the PGNN to inaccurately update weights and biases near the beginning of training.



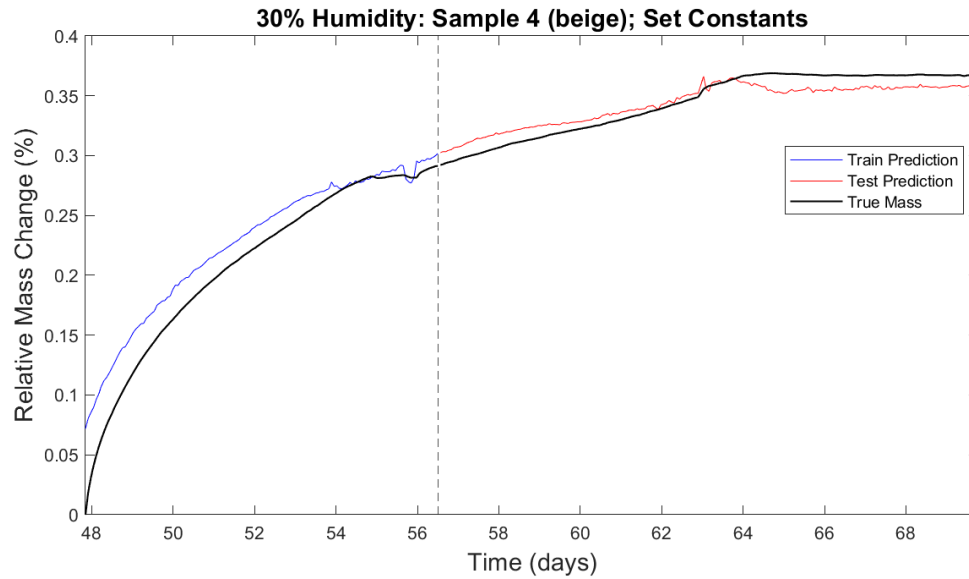
Appendix Figure 14 Residuals for the 0% humidity mass curve for the model trained with set constants on beige sample #4. The predictions are within +/- 0.01% of the data for the majority of time up to about 20 days, with the remaining future time predictions deviating up to -0.05%.



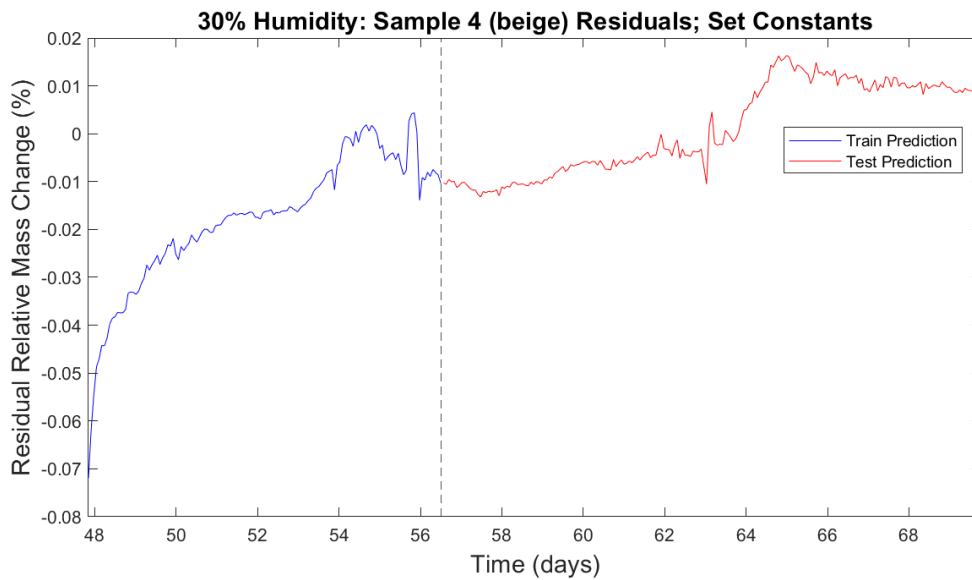
Appendix Figure 15 Results for the 5% humidity mass curve for the model trained on all data of beige sample #4 with set constants. The predictions appear to have a similar output than the original predictions of **Appendix Figure 7**, but appears generally smoother except for cases of unrealistic mass spikes likely due to input fluctuations.



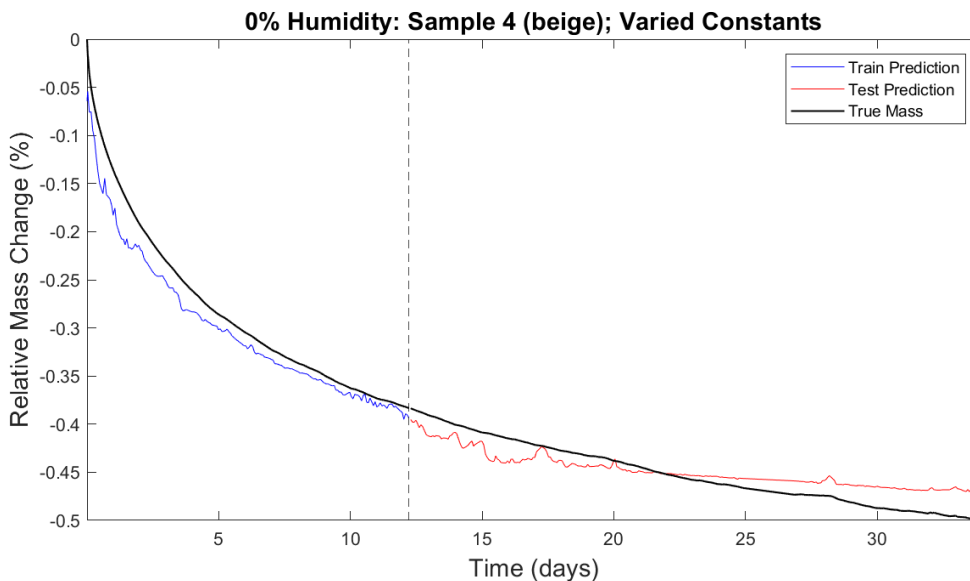
Appendix Figure 16 Residuals for the 5% humidity mass curve for the model trained with set constants on beige sample #4. The predictions are within $\pm 0.005\%$ of the data for the majority of time past around day 38, with the most divergence at the beginning up to -0.025% and several unrealistic mass spikes likely due to input fluctuations.



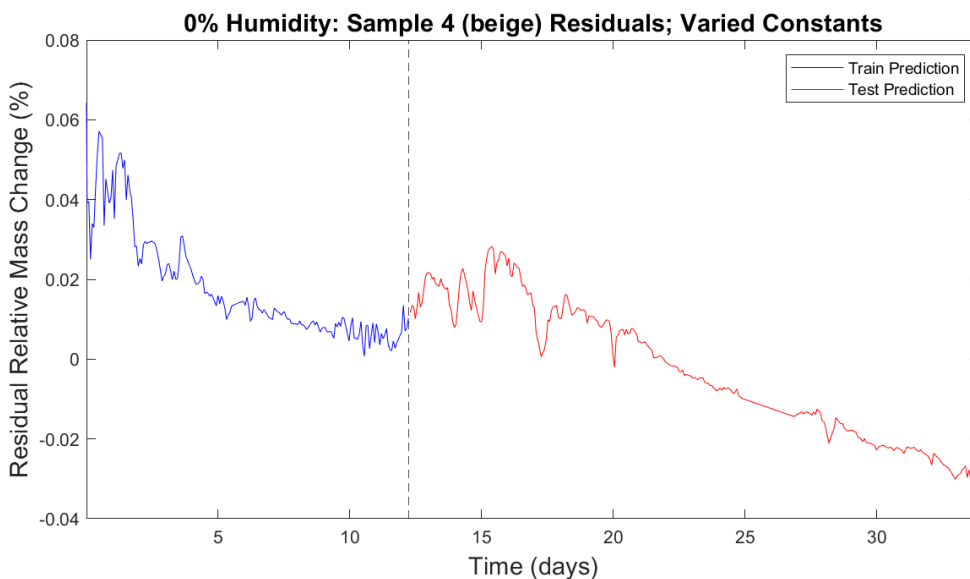
Appendix Figure 17 Results for the 30% humidity mass curve for the model trained on all data of beige sample #4 with set constants. The predictions appear mostly consistent with the original predictions in **Appendix Figure 9**, but appears to have better fidelity at the end time and worse fidelity near the start time.



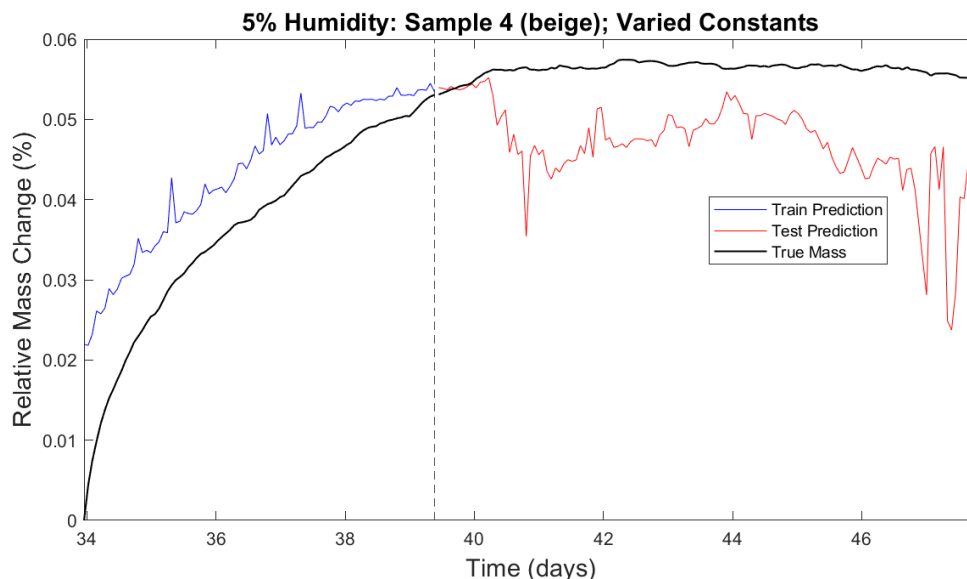
Appendix Figure 18 Residuals for the 30% humidity mass curve for the model trained with set constants on beige sample #4. The predictions are within $\pm 0.02\%$ of the data for the majority of time past around day 52, with the most divergence at the beginning up to -0.08% and an unrealistic mass spike likely due to an input fluctuation.



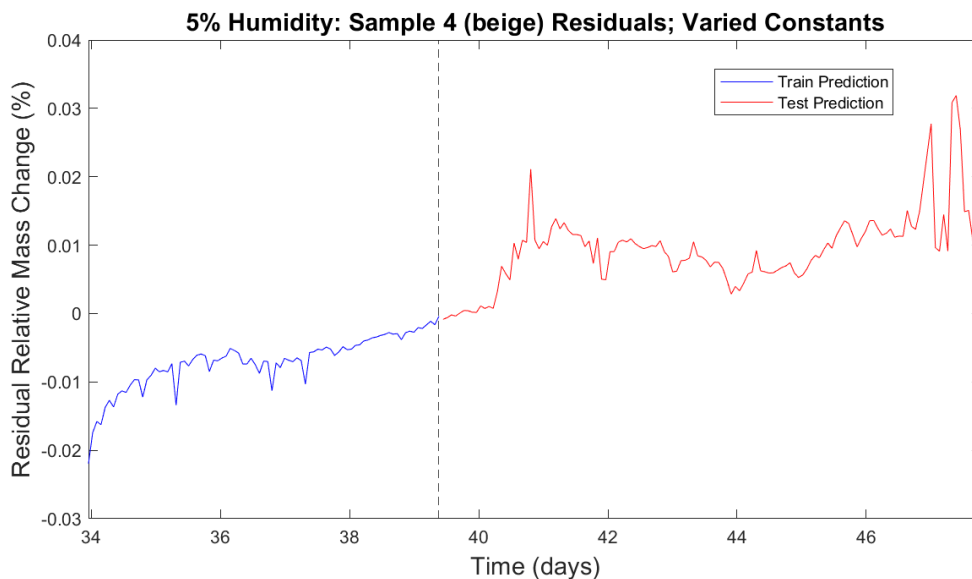
Appendix Figure 19 Results for the 0% humidity mass curve for the model trained on all data of beige sample #4 with unideal constants. The predictions have good fidelity and appear to even have better performance than the original predictions in **Figure 4.4**.



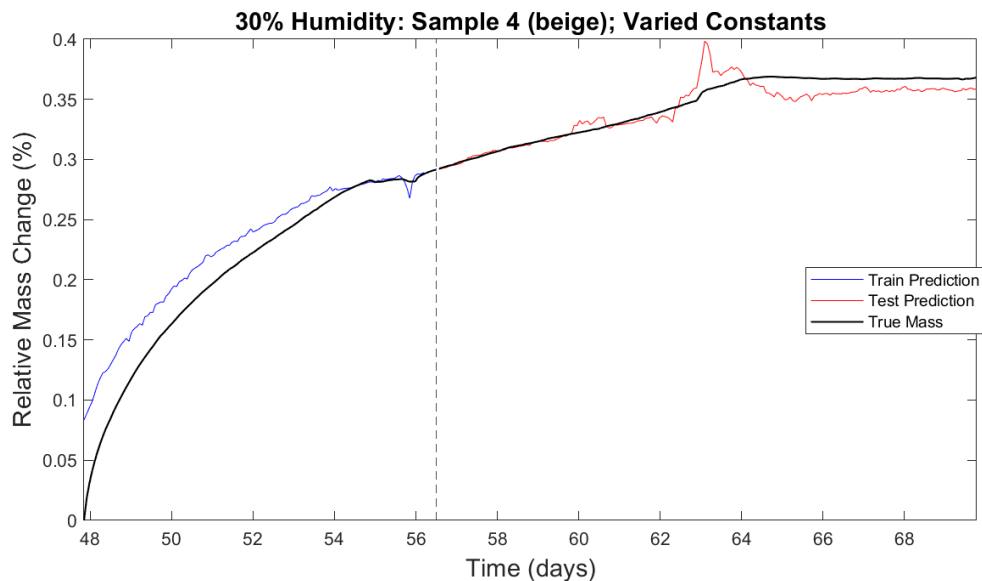
Appendix Figure 20 Residuals for the 0% humidity mass curve for the model trained with unideal constants on beige sample #4. The predictions are within $\pm 0.04\%$ of the data for almost the full time, with the most divergence at the beginning up to -0.06% .



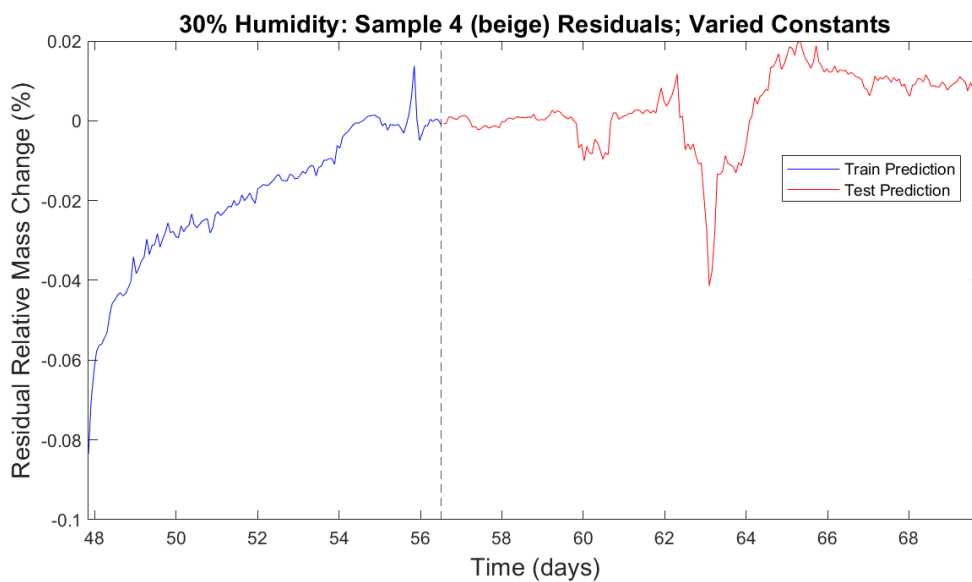
Appendix Figure 21 Results for the 5% humidity mass curve for the model trained on all data of beige sample #4 with unideal constants. The predictions have a poor performance compared to the original predictions in **Appendix Figure 7**, indicating that the model was unable to optimize the weights and biases and constants as seen in **Table 4.2**.



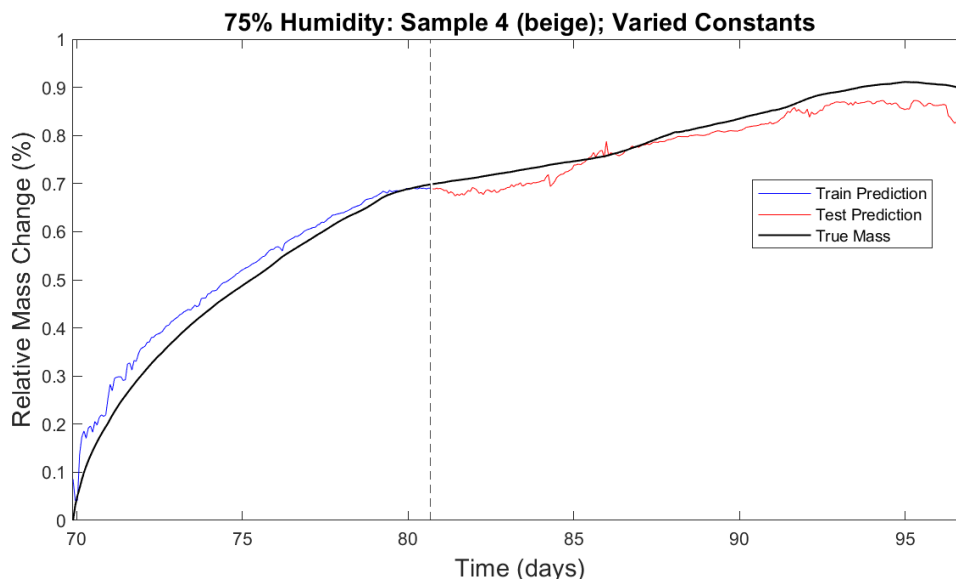
Appendix Figure 22 Residuals for the 5% humidity mass curve for the model trained with unideal constants on beige sample #4. The predictions deviate more so than other 5% predictions between -0.02% and 0.03%.



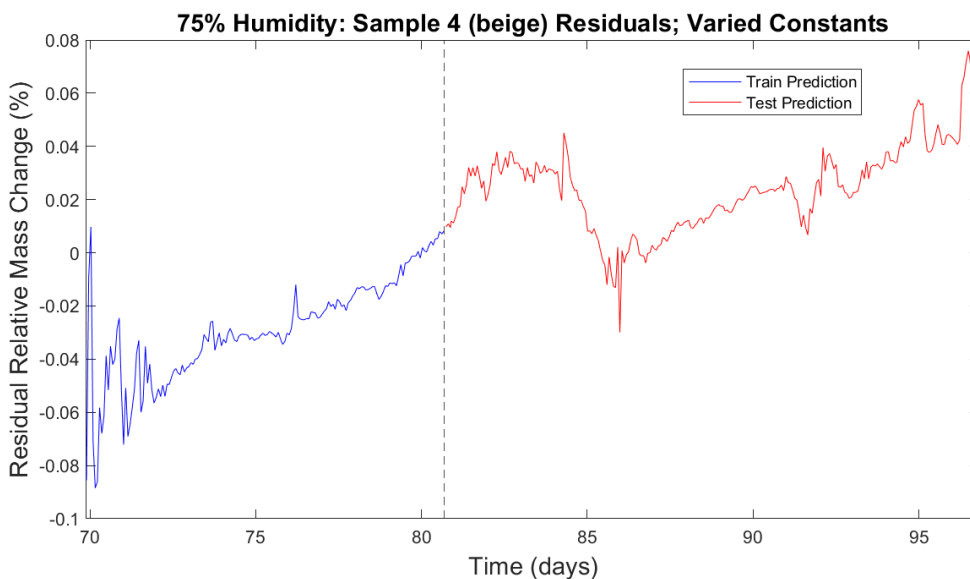
Appendix Figure 23 Results for the 30% humidity mass curve for the model trained on all data of beige sample #4 with unideal constants. The predictions have decent fidelity with the data such as **Appendix Figures 9, 17**, but appear to have deviation in the training data and unreasonable spike at around 63 days.



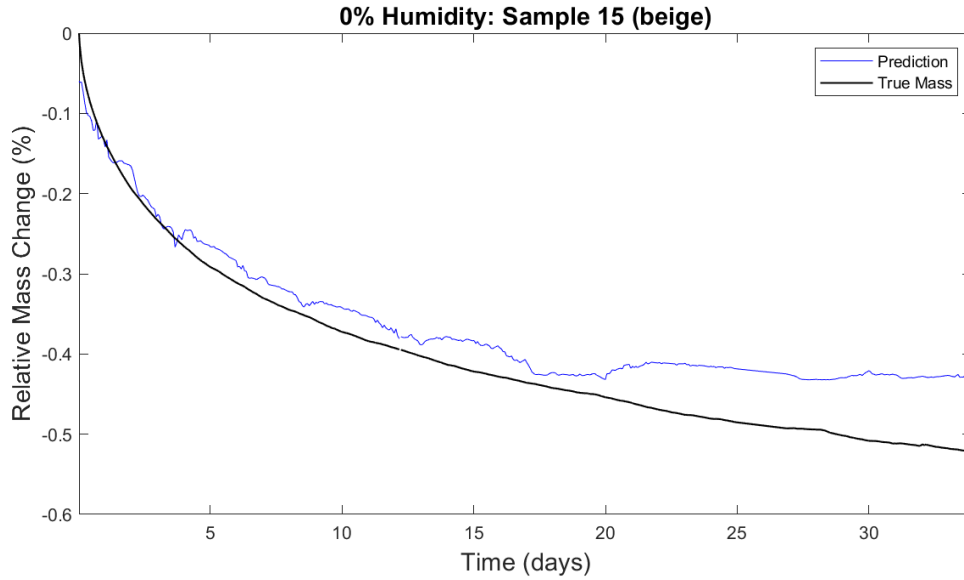
Appendix Figure 24 Residuals for the 30% humidity mass curve for the model trained with unideal constants on beige sample #4. The predictions are within $\pm 0.02\%$ past day 52 but deviate up to -0.08% at the beginning with an unrealistic spike at day 63.



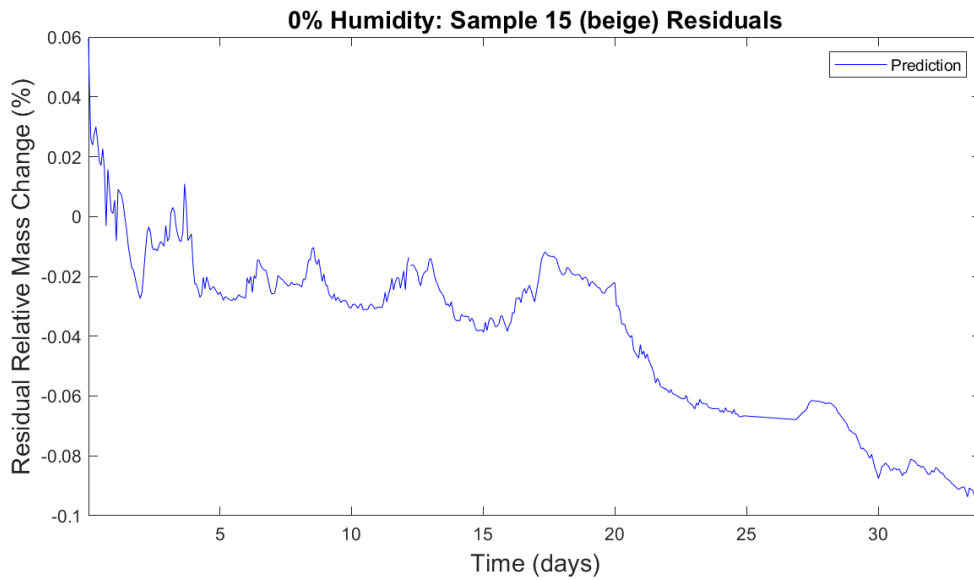
Appendix Figure 25 Results for the 75% humidity mass curve for the model trained on all data of beige sample #4 with unideal constants. The predictions have decent fidelity with the data such as **Figures 4.6, 4.10** indicating the constants optimized close to the original values in **Table 4.1**.



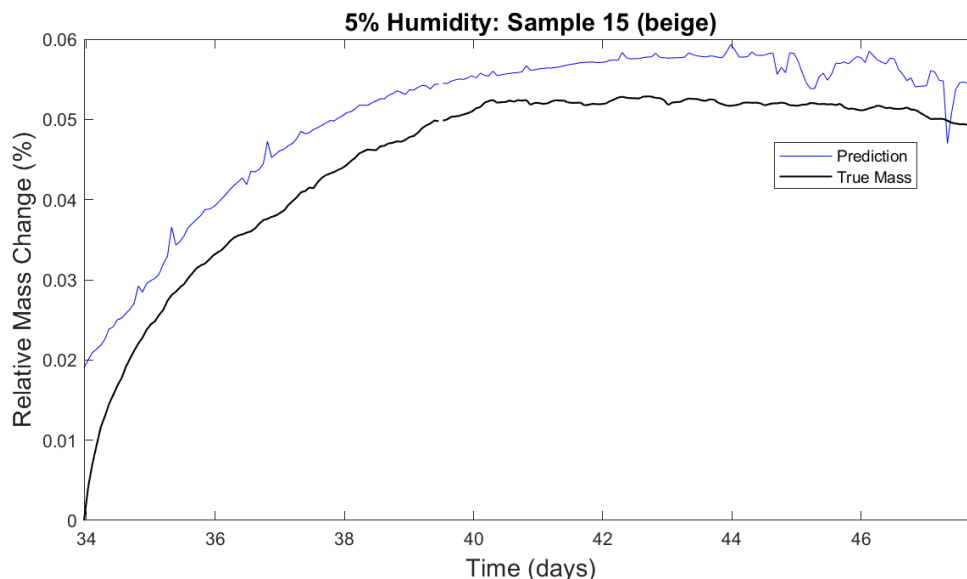
Appendix Figure 26 Residuals for the 75% humidity mass curve for the model trained with unideal constants on beige sample #4. The predictions are within $\pm 0.04\%$ between day 75 and 95, with deviations up to -0.08% at the beginning and 0.08% at the end.



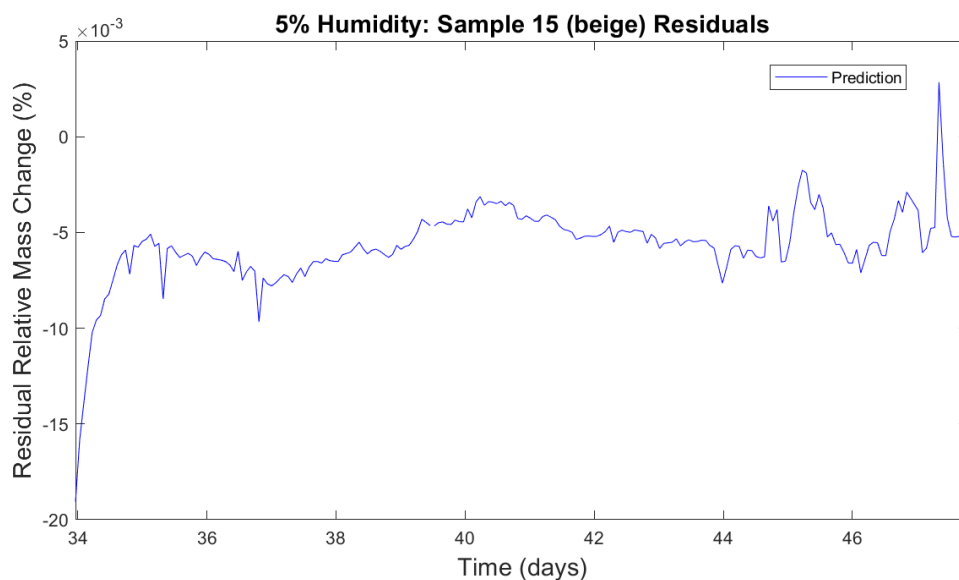
Appendix Figure 27 Results for the 0% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions have decent fidelity but deviate more so than the results for the original predictions in **Figure 4.4**, indicating the varied inputs between the samples can have a significant impact.



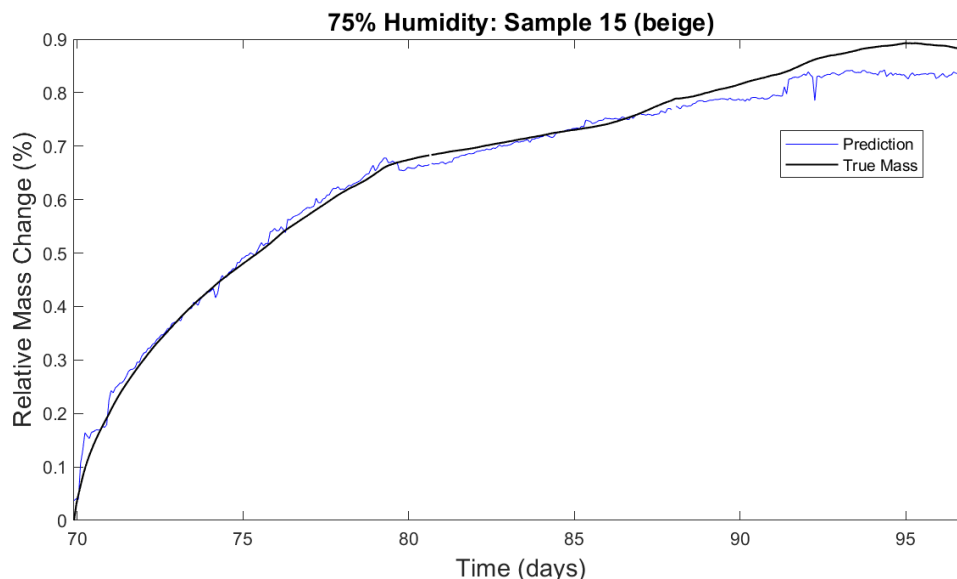
Appendix Figure 28 Residuals for the 0% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions have decent fidelity between $\pm 0.02\%$ up to about day 15 but deviate up to -0.1% at the end.



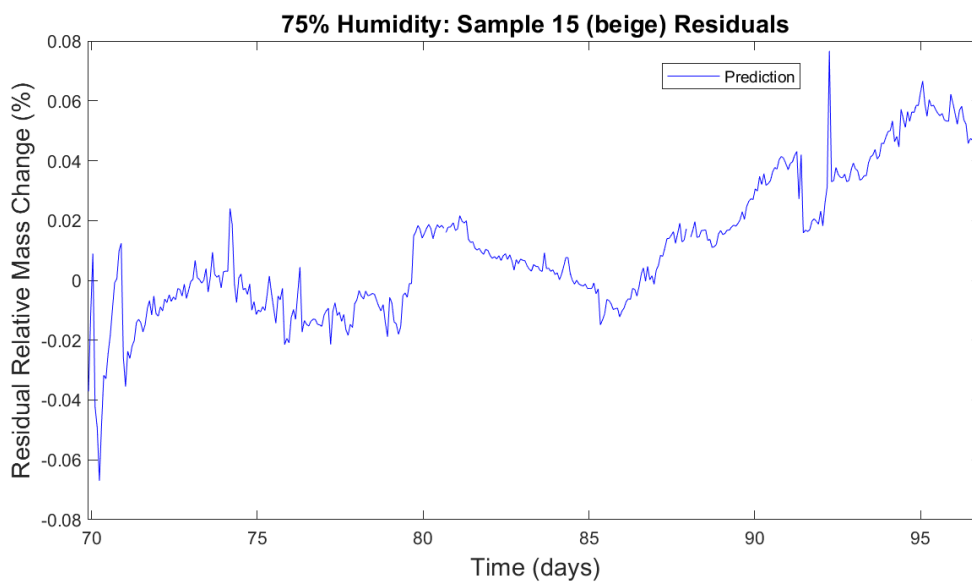
Appendix Figure 29 Results for the 5% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions overall overestimate the true data and has a slightly degraded performance compared to the original predictions in **Appendix Figure 3**.



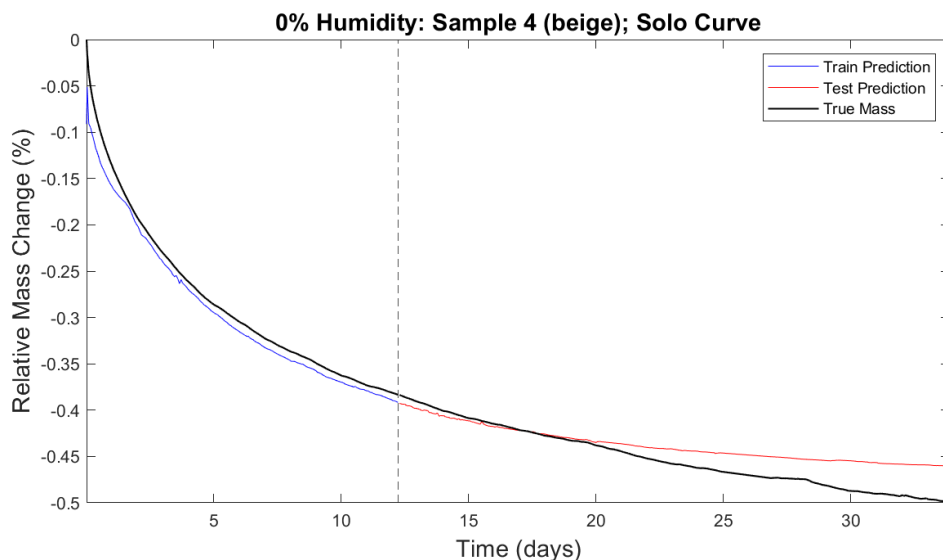
Appendix Figure 30 Residuals for the 5% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions overshoot the data and are approximately -0.005% different, with the most deviation being -0.02% at the beginning.



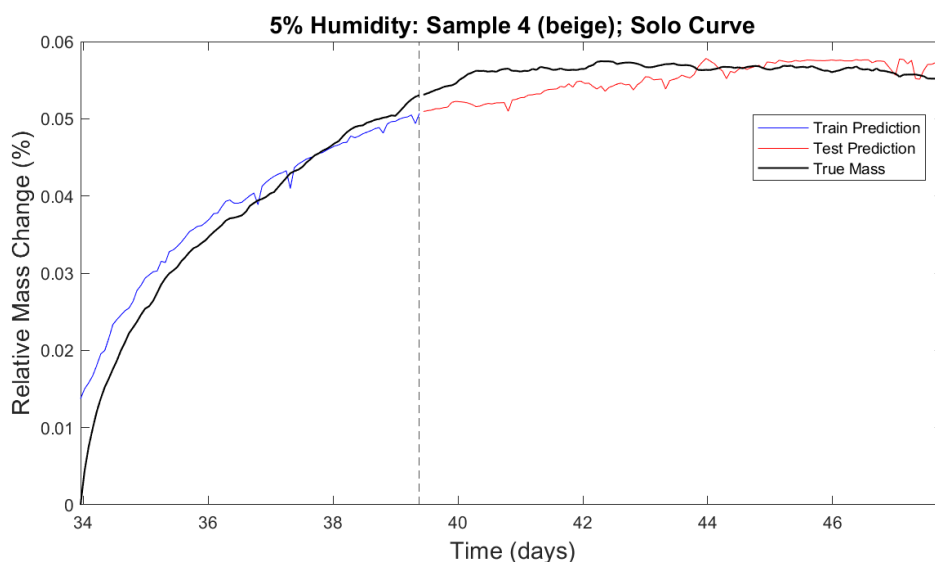
Appendix Figure 31 Results for the 75% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions overall have excellent fidelity as with **Figure 4.6**, likely indicating the inputs are closely aligned between the samples for this group.



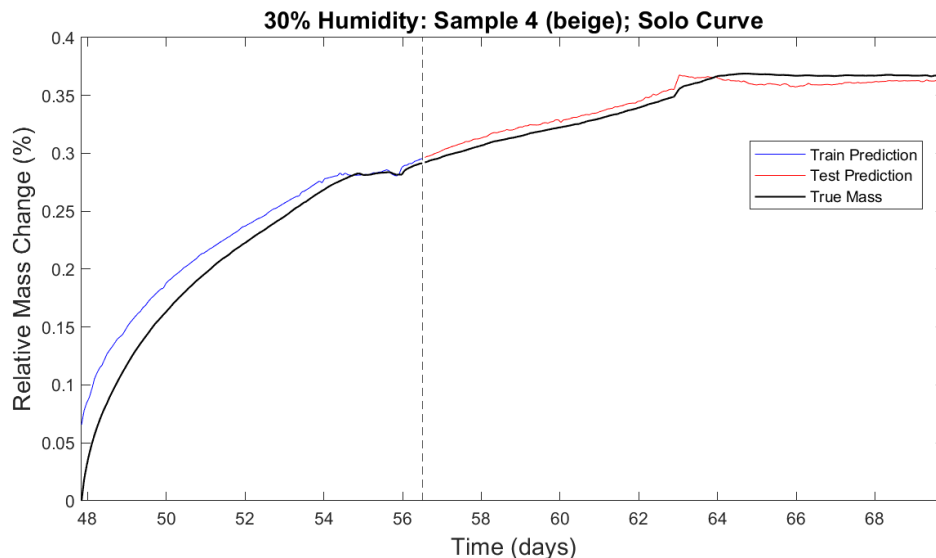
Appendix Figure 32 Residuals for the 75% humidity mass curve of beige sample #15 for the model trained on all data of beige sample #4. The predictions are within $\pm 0.04\%$ up to about 90 days with the most deviation and $\pm 0.06\%$ at the end and beginning respectively.



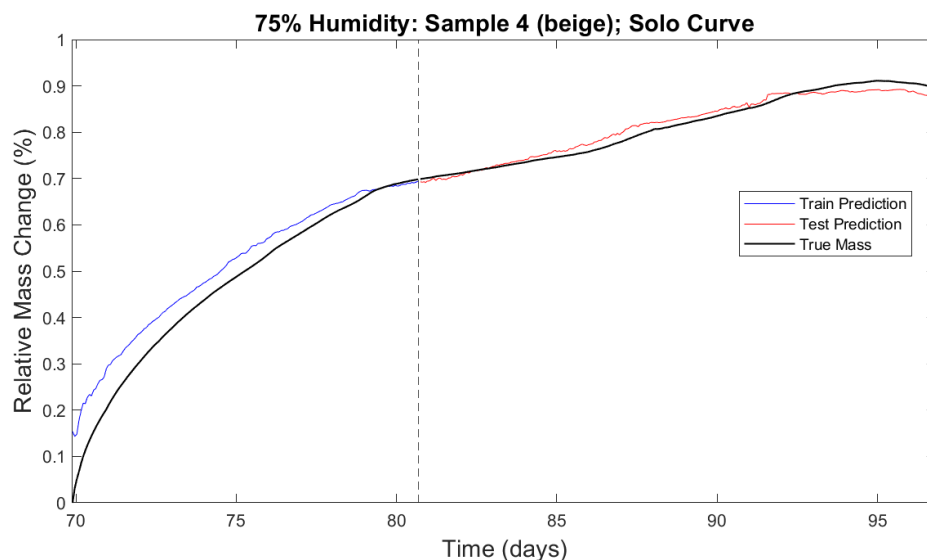
Appendix Figure 33 Results for the 0% humidity mass curve of beige sample #4 for the model trained on only the solo humidity group. The predictions have great fidelity, and arguably better than other 0% predictions of different models as expected for a model not having to process different data with similar inputs together.



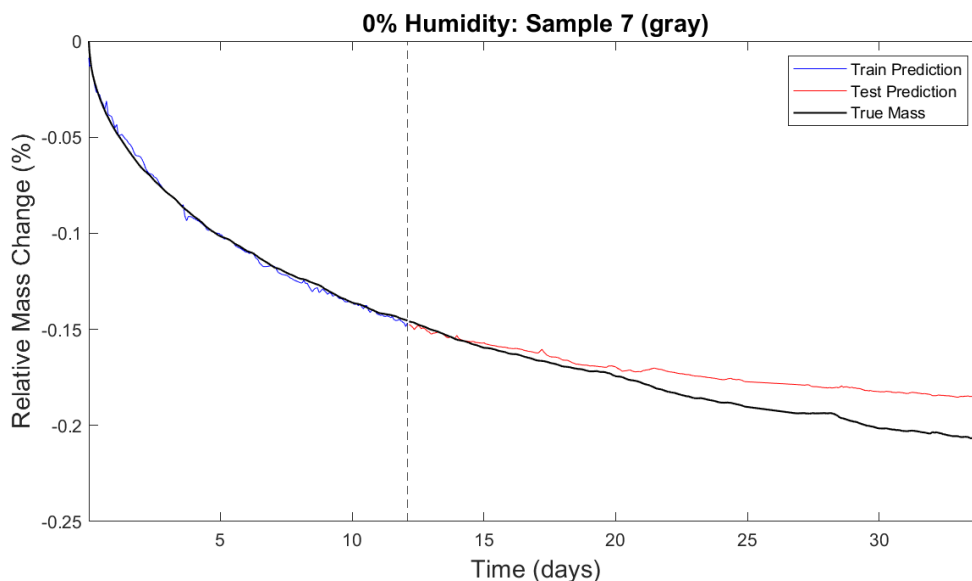
Appendix Figure 34 Results for the 5% humidity mass curve of beige sample #4 for the model trained on only the solo humidity group. The predictions have great fidelity, and overall better performance compared to most of the other 5% predictions.



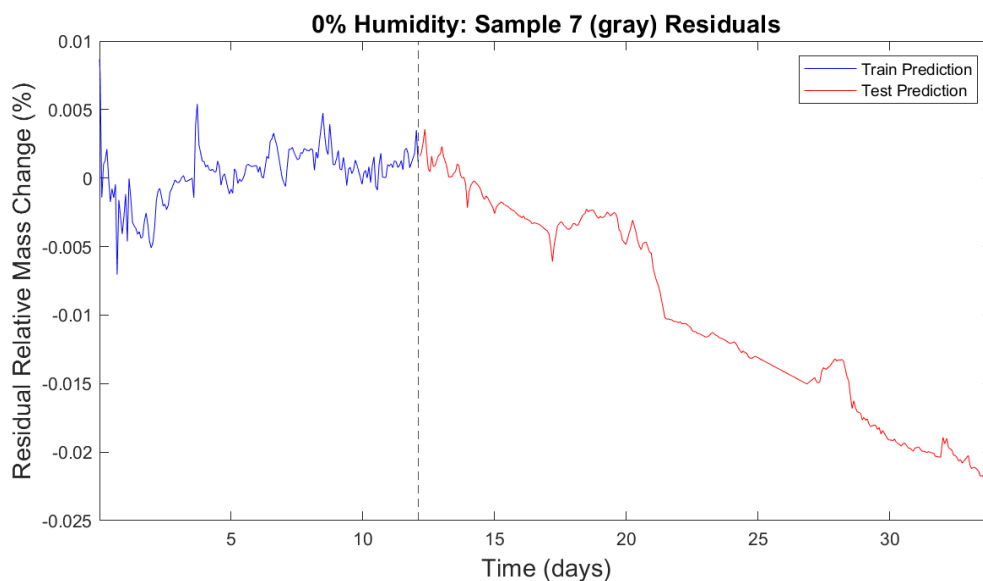
Appendix Figure 35 Results for the 30% humidity mass curve of beige sample #4 for the model trained on only the solo humidity group. The predictions have great fidelity similar to other 30% predictions, although has some deviation in the beginning of the training data.



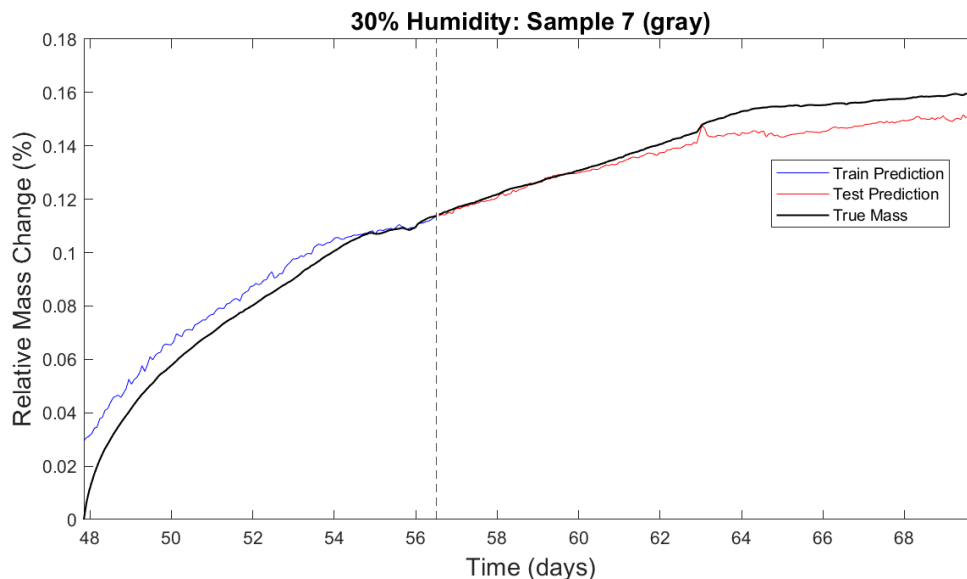
Appendix Figure 36 Results for the 75% humidity mass curve of beige sample #4 for the model trained on only the solo humidity group. The predictions have great fidelity similar to other 75% predictions with generally better performance and the end, although has some deviation in the beginning of the training data.



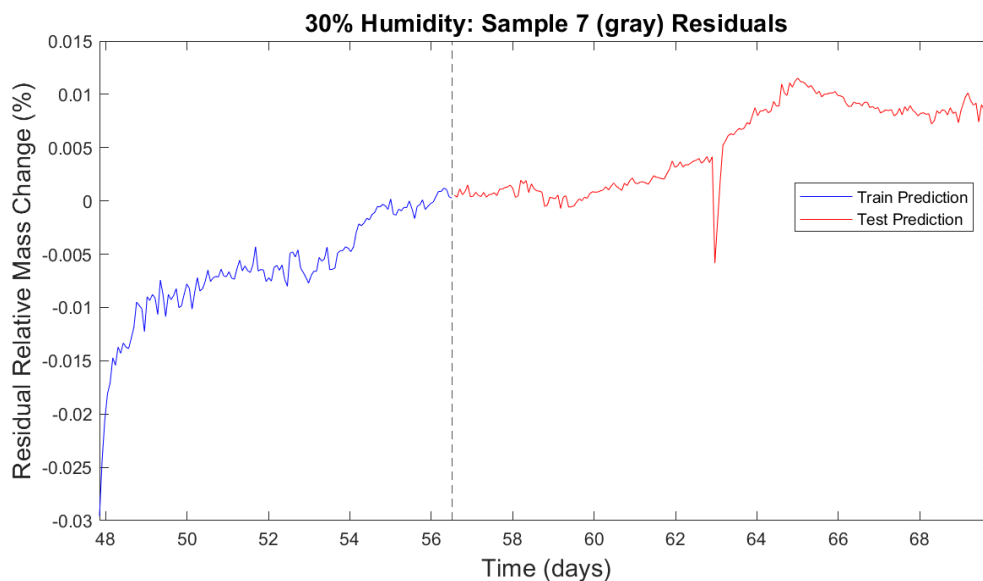
Appendix Figure 37 Results for the 0% humidity mass curve for the model trained on all data of gray sample #7. The predictions have good fidelity with the data similar to that of the beige sample training.



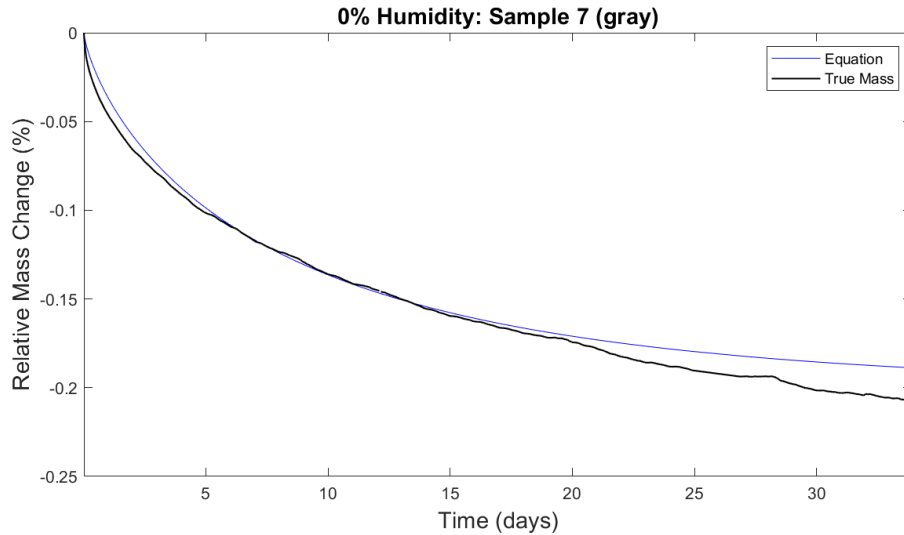
Appendix Figure 38 Residuals for the 0% humidity mass curve for the model trained on all data of gray sample #7. The predictions fall within +/- 0.005% up to day 20 before deviating up to -0.02%.



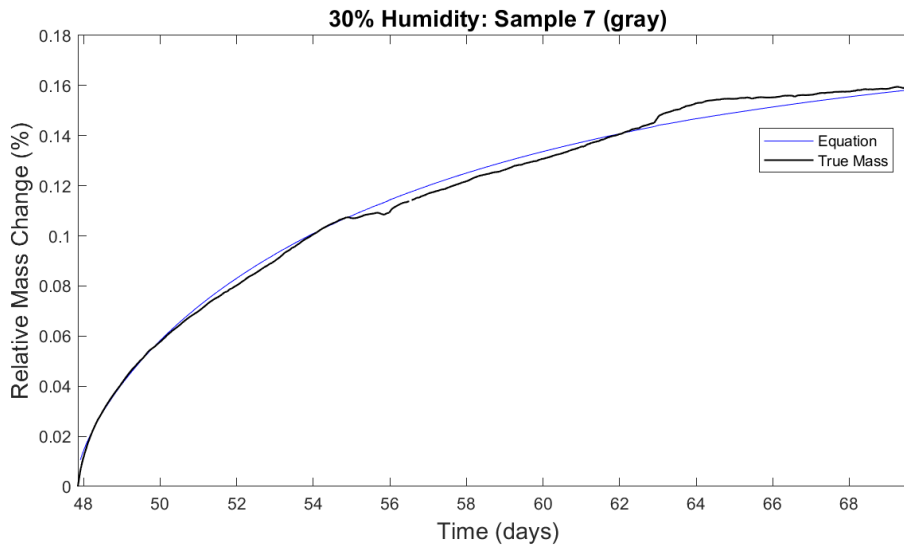
Appendix Figure 39 Results for the 30% humidity mass curve for the model trained on all data of gray sample #7. The predictions have good fidelity with the data similar to that of the beige sample training.



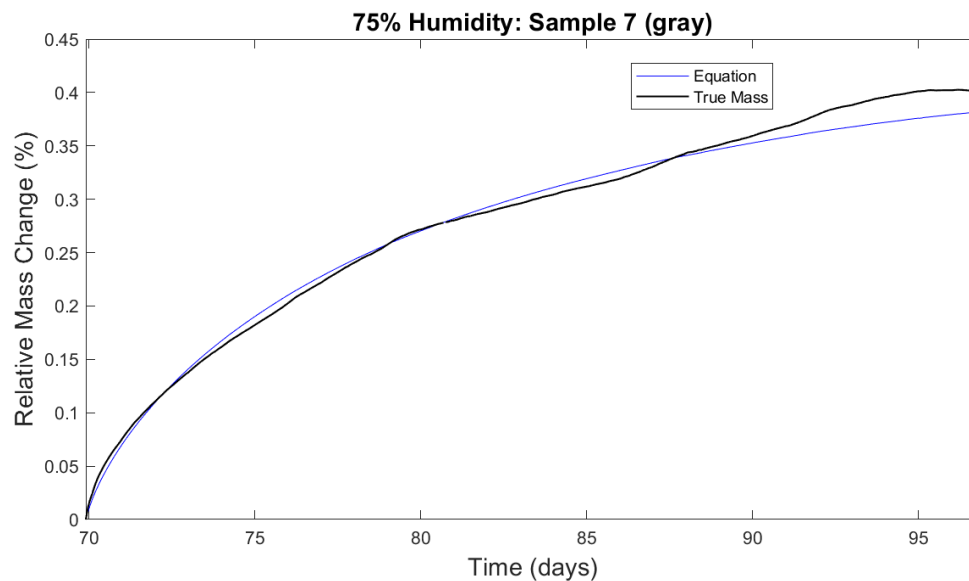
Appendix Figure 40 Residuals for the 30% humidity mass curve for the model trained on all data of gray sample #7. The predictions fall within $\pm 0.01\%$ for almost the whole data with the most deviation at the beginning up to -0.03% .



Appendix Figure 41 The predictions on the 0% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of gray sample #7. There is overall good fidelity, indicating validity of the equation under complex conditions and the ability of the model to determine physical constants from the data. The values can be seen in **Table 4.3**.



Appendix Figure 42 The predictions on the 30% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of gray sample #7. There is overall excellent fidelity, indicating validity of the equation under complex conditions and the ability of the model to determine physical constants from the data. The values can be seen in **Table 4.3**.



Appendix Figure 43 The predictions on the 75% mass curve of the one-dimensional slab equation using the optimized parameters of the full data model of gray sample #7. There is overall good fidelity, indicating validity of the equation under complex conditions and the ability of the model to determine physical constants from the data. The values can be seen in **Table 4.3**.

VITA

Michael B. May was born and raised in southern California where he graduated from Temecula Valley High School. He attended the University of California, Berkeley, earning a BA in Physics and Astrophysics with Distinction. During the course of his studies, Michael aided in supernovae research through observational astronomy utilizing the Nickel Telescope at the Lick Observatory as well as astrobiology research during a summer internship. Following an illuminating internship at Lawrence Livermore National Laboratory working on hydrodynamic simulations, he has focused his research on machine learning applications for his MS degree and later PhD degree in the Nuclear Engineering Department at the University of Tennessee, Knoxville. Michael enjoys playing the cello, running, pickleball, and trivia.