Monetary Flood Damage Prediction Based On Machine Learning Models

JAYA K. BIJOOR

Mentor: Shreyaa Raghavan, Ph.D. candidate, Massachusetts Institute of Technology
Mentor: Jane Woodward, Adjunct Professor, Stanford University

Abstract

Flooding and heavy rains have increased by 50% in the last decade due to climate change. Since 2000, there has been a 24% increase in people directly affected by flooding, and for the past 25 years, the United States has experienced an urban flooding event every 2 to 3 days [1]. Currently, the Federal Emergency Management Agency (FEMA) is the only resource people utilize to determine whether their homes are at risk for flooding. However, FEMA flood maps are costly to produce, outdated, and lack predictive accuracy, leaving homeowners unaware and unprepared for potential dangers [2]. There is a dire need to create additional tools for communities to obtain more accurate estimates of flood damages, but typically, computational methods require rich datasets, which are difficult to obtain for flooding events [3]. In this paper, we aim to create a model that predicts monetary damages caused by floods using various factors related to the flood event, such as duration, location, and cause. I utilize a flood dataset from the National Oceanic and Atmospheric Administration (NOAA) and data pre-processing techniques to remove irrelevant features and handle missing values [4]. I generated flood damage predictions by leveraging regression and classification machine learning methods, such as Linear Regression, Random Forest Regression, XGBoost Regression, and Neural Networks. Our findings show we can effectively use upsampling techniques to combat limited flood data. We also show that floods with higher monetary damages are easier to predict, which is important because these floods inflict greater hardship on communities. The results of this research can improve preparedness for flood-related risks, property value assessments, and accuracy of flood insurance policy underwriting [2]. Ultimately, the proposed model provides a preliminary study on how we can help individuals make better-informed decisions and prepare for the impact of flooding on their homes. We hope this work encourages future research that uses machine learning to prepare citizens for natural disasters.

I. Introduction

Flooding is a huge problem around the world. It leads to the loss of lives and homes and causes financial stress on families. Floods are extremely common and costly. Climate change will only exacerbate their frequency and intensity, particularly in low-lying areas and regions prone to heavy rainfall [5]. According to the United Nations, climate change has contributed to rising sea levels, increasing the risk of storm surges and coastal flooding. In addition, warmer temperatures lead to more precipitation, which can overwhelm existing drainage systems and cause riverine floods [6].

A study published in Environmental Health Perspectives shows that low-income communities in the United States are disproportionately affected by flooding. These communities are more likely to be in flood-prone areas, such as coastal regions or near rivers. Low-income communities may lack the resources and infrastructure to adequately prepare for or recover from flood events [7]. Moreover, lower-income individuals may have limited access to insurance coverage, making it difficult to recover financially after a flood [3].

Furthermore, climate change exacerbates existing inequalities by disproportionately impacting vulnerable populations. According to the World Health Organization, climate change is expected to increase the burden of diseases, displacement, and food insecurity, particularly among low-income communities in developing countries. These communities often lack the resources to adapt to or mitigate the impacts of flooding caused by climate change [8].

The average cost per flood event in the United States is $4.6 billion. In addition, 21.8 million U.S. homes are in flood-prone zones, which is 67% more than the number of at-risk homes reflected on federal FEMA flood maps [1]. This is because FEMA maps are costly, take at least seven years to update, and are not predictive. As a result, these flood maps leave homeowners unaware and unprepared for the potential dangers they face, what type of insurance they should buy, and what types of infrastructure should be restricted in their communities [2]. As people continue to grapple with the challenges of climate change, it is essential to prioritize equitable and sustainable solutions to address the harm caused by flooding and reduce the disproportionate

impacts on vulnerable populations. The goal of this project is to help alleviate these problems. If people are informed about the potential outcomes of flooding in their area, they can prepare in advance and recover economically from the damage faster [3].

As a result, we seek to create a tool to predict monetary damage before flooding occurs. I use data-driven and machine learning approaches to create and analyze multiple models. I also ensure that the variables used in the models are readily available so that everyday individuals can utilize this tool in the future.

II. Background

A. Flood Risk Prediction

Most of the current literature on flood-related prediction has focused on improving flood risk assessments. A study in Norway showed that artificial neural networks and support vector machines were the most successful models for short and long-term flood risk prediction. The study also found that rainfall, although typically used in flood prediction, was an inadequate feature for accurate prediction. The researchers discovered that factors like geography and soil moisture were more effective flood resource variables [9]. Another study by the European Geosciences Union further discussed the importance of machine learning in flood prediction, citing the value of basic regression techniques over more advanced ones like convolutional neural networks [10]. A third study in Romania used classification and regression trees, such as Random Forest, Boosted Regression Trees, and Extreme Gradient Boosting, to create a flood susceptibility map. The results showed that slope was the most important factor triggering flash floods, and the Random Forest model achieved the highest accuracy [11]. While these studies show potential flood variables that could be useful, many of the variables they use are uncommon and not easily accessible, such as surge level, daily flows, and flood peak discharge. Additionally, all of these studies were conducted in Europe, so their data and models may not be well-suited for flood prediction in the United States.

B. Predicting Economic Loss from Floods

Not a lot of work on flood-related economic loss has been conducted, but some research has been done on this topic in China. A study of flooding in China using machine learning shows that the frequency and intensity of flooding increase as climate change worsens. The research paper presents a prediction model for direct economic losses caused by floods, showing that an increase in flooding is directly proportional to economic losses in farming. The study uses a machine learning approach, specifically Gradient Boosting Regression with MinMax Scaler, to predict economic losses based on various indicators. The researchers conducted a correlation analysis and identified several indicators positively correlated with economic losses, such as reservoir loss, reduced production in the agricultural sector, casualties, and road disruption. While this study illustrates a successful model and useful factors, it focuses on economic loss in farming regions in China, whereas we aim to create a model that can predict damage for both rural and urban areas in the United States. The study conducted in China also uses several niche factors such as railway disruption, length of sewage pipes, and sown areas with different production levels [12]. We aim to apply these findings to communities in the United States by using easily available factors that citizens would have access to.

III. Dataset

I used a United States Flood dataset from NOAA [4]. The initial dataset dimensions were 698,507 samples with 26 features (columns) each. However, after cleaning the dataset, the dimensions were 41,703 samples with 75 features each.

Feature	Description	Feature	Description
DATE_BEGIN*	begin datetime of an event yyyymmddHHMMSS	SOURCE*	flood information source
DATE_END*	end datetime of an event yyyymmddHHMMSS	SOURCE_DB*	source database
DURATION	duration of an event in hours	SOURCE_ID*	original ID in the source database
LON	longitude in degrees	DESCRIPTION*	event description
LAT	latitude in degrees	SLOPE	calculated slope based on SRTM DEM 90m
COUNTRY*	United States of America	DEM	Digital Elevation Model
STATE	US state name	LULC	Land Use Land Cover
AREA	affected areas in km^2	DISTANCE_RIVER	distance to major river network in km
FATALITY	number of fatalities	CONT_AREA	contributing area (km^2), from MERIT Hydro
DAMAGE	economic damages in US dollars	DEPTH	500-yr flood depth
SEVERITY*	event severity, (1/1.5/2) according to DFO	YEAR	year of the event
CAUSE	cause of the flood	GEOMETRY*	hydraulic geometry
ID*	unique ID in the dataset	LOCATION*	town event occured in

Table 1. The table shows every feature in the original dataset with a description. Starred features were removed from the cleaned dataset.

I cleaned the dataset to ensure the best predictions.

1) **Removed irrelevant attributes**: I removed the columns ID, database source, and ID source because they are irrelevant to predicting the damages. I removed the columns for the start and end date because there is already a column for the duration in hours, and I did not want the model to double count the duration of the flood. I removed the country and source columns because every instance had the same value for country, United States, and source, NOAA. I removed the description because it was difficult to classify as the description of most of the samples is different, and there is already a column for cause. I removed severity, geometry, and location because most instances had null values for these two factors.
2) **Imputed incomplete data**: I imputed (replaced the null values in a column with the average of the known values in the rest of the column) for duration, area, fatality, slope,

digital elevation model, land use, land cover, distance to river, contributing area, and 500-year flood depth. Very few of these values were missing, so imputing them would have a very small effect on the model.

3) **Removed samples with incomplete data**: Some samples had null values for many of their features. I removed the samples where damage, state, latitude, longitude, and year were not included because they could not be imputed. I did not impute damage because it is the predicted value. I also dropped the instances where damage was below $\$10^3$ and above $\$10^{10}$ because those values were likely incorrect and could skew the results.
4) **Discretized attributes**: I discretized (categorized) the columns for state and cause because I wanted each value in the dataset to be associated with a number. I used one-hot encoding to discretize by creating new columns for each possible state or cause, adding 1 for yes and 0 for no. I chose one hot encoding over label encoding because label encoding assigns whole numbers starting from 0 and increases. I did not want the model to assume that a larger number has more significance than a smaller number. I changed the null values in the state and cause columns to "unknown" so that these values could be discretized.

A. Visualizing the Dataset

Before creating a model, I visualized the dataset using a bar graph to show the frequencies of different values. As shown in *Fig. 1*, most damage values fall between $\$10^3$ and $\$10^6$.

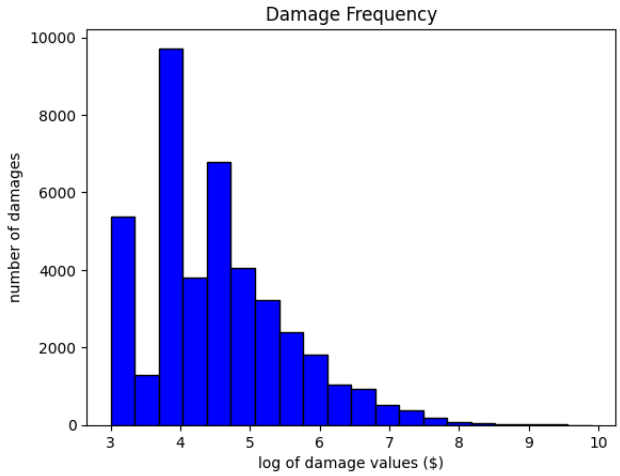

Fig 1. Histogram of the frequency of damage values.

Next, I depicted the data using a correlation matrix, which shows the correlation between any two factors. Most of the correlation values are close to zero, as indicated in *Fig. 2*. This is one indicator that a non-linear regression model will work better for prediction.

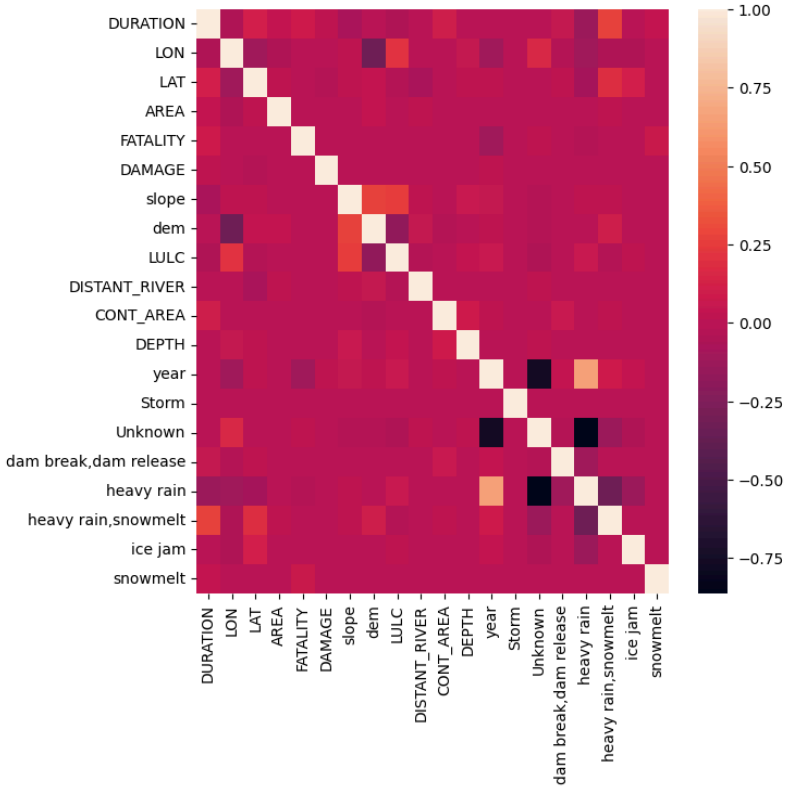

Fig 2. Correlation matrix of attributes excluding all the states to improve readability. Lighter colors show a greater positive correlation and darker colors display a greater negative correlation.

IV. Methodology

A. Regression

Regression is a useful tool for determining the relationship between independent variables (input features) and a dependent variable (monetary damage) [13]. I implemented several regression models to see which ones had the best accuracy. Before each model was tested, the dataset was split into train and test data (x_train, y_train, x_test, and y_test). The size of the test data is 20%

of the entire dataset. Training data helps the model learn, and testing data helps find the model's accuracy. X is all the input variables, and y is the output value (damages).

B. Multiple Linear Regression

The model aims to find the best fit line between the dependent variable (in this setup, damages) and all of the other features (independent variables). The best fit line is modeled by the following equation, $y = b_0 + b_1x_1 + b_2x_2 ... + b_nx_n$, where b_0 is the intercept and b_l is the slope/correlation of the corresponding attribute, x_l [14].

C. Gradient Boosted Regression

This model predicts non-linear relationships between damages and the other features by combining multiple weak models to create a decision tree with better performance. This model is represented by the equation, $F_0(x) = argmin\, y(\sum_{i=1}^{n} L(y_i, y))$, where F_0 is the predicted value, L is the loss function, and argmin searches for the y value that minimizes $\sum_{i=1}^{n} L(y_i, y)$. The loss function we chose was mean squared error (MSE). MSE is calculated by taking the average squared difference between the predicted and true values for damages. It is represented by the equation, $MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y_i})^2$, where N is the number of samples we are testing against. I also chose a max_depth value of 2, which is a relatively low value. Max_depth indicates how deep the decision tree is, so the deeper the tree, the more splits it has. Higher values for max_depth can often lead to overfitting, which occurs when the model accurately predicts training data but does not improve on new data [15].

D. Ridge Regression

Ridge regression is typically useful for models, like ours, with less than 100,000 samples. Ridge regression is similar to linear regression, except it incorporates a small amount of bias (the extent

to which the model deviates from the samples). The added bias is called the Ridge Regression Penalty, represented by the equation $\lambda \times slope^2$, where λ is determined by cross-validation and $slope$ is the weight of each feature [16].

E. Random Forest Regression

Similarly to gradient boosted regression, random forest regression combines multiple weak models, such as models that only predict the mean or use linear regression, to create an additive model with more accurate predictions. It is represented by the equation, $g(x) = f_0(x) + f_1(x) + f_2(x) + ...$, where g is the final model composed of the simple models, f_i [17].

F. XGBoost Regression

XGBoost regression is an extreme implementation of gradient boosted regression. It is often used with datasets with over 1,000 training samples and less than 100 features, like ours. Unlike gradient boosting regression, XGBoost has more advanced regularization, which helps minimize the loss function (MSE) and reduces overfitting and underfitting [18].

G. Regression Neural Network

Neural networks simulate how neurons work in our brains to learn the complex non-linear relationship between the features and the target value, in our case, damages. As shown in *Fig. 3*, neural networks consist of an input layer, hidden layers, and an output layer. Each layer has n neurons and an activation function that introduces non-linearity to the function. A neural network is modeled by the equation, $f: R_n \rightarrow R_p, f = g \circ f_k \circ ... f_2 \circ f_1$, where n is the dimension of the features, p is the dimension of the damages, g is the output function, and each function f_i is a composed multivariate function, $f_i(x) = a(w_i x + b_i)$. The a in the composite function represents the activation function [19].

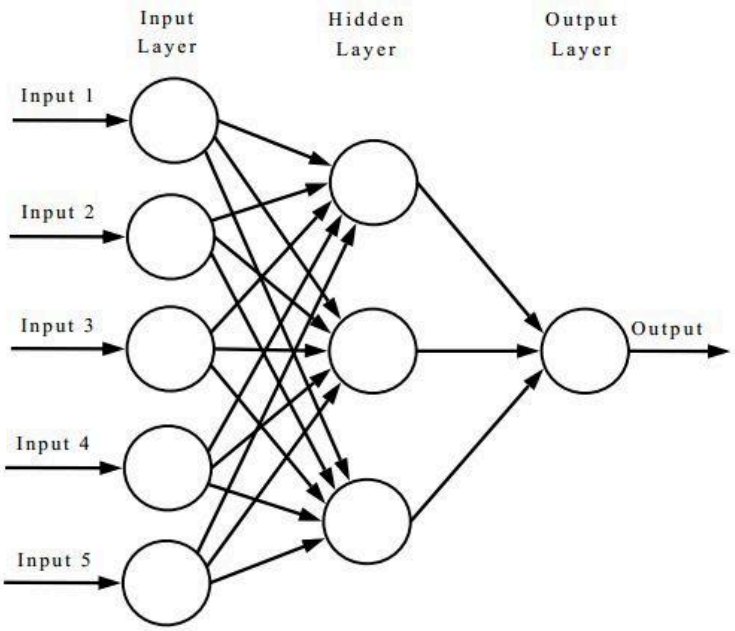

Fig 3. Diagram of a neural network illustrating input layers, hidden layers, output layers, and weights [20].

H. Classification

Classification models differ from regression models because rather than predicting a single numerical value, they predict whether a sample falls within a category [21]. Rather than predicting the exact value of damage, we propose a classification model that predicts the degree of damage. I use 7 categories, where the degrees of monetary damage are 10^3, 10^4, 10^5, 10^6, 10^7, 10^8, 10^9 .

I. Classification Neural Network

To prepare the data for the model, I first took the floor of the log of the damages, $\lfloor log(d_i) \rfloor$, where d_i is the damage value. As a result, the damages ranged from $(3, 10)$. I shifted the scale to $(0, 7)$ by subtracting all the damage values by 3.

Each sample's category is represented through one-hot encoding. One-hot encoding gives each sample an encoded sequence so that the model does not assume that higher values have more importance [22]. For example, if a damage value falls within the 10^5 category, it is not given a value of 2, but a one-hot encoded value of [0, 0, 1, 0, 0, 0, 0]. Likewise, a damage value within the 10^3 category has a value of [1, 0, 0, 0, 0, 0, 0].

As shown in *Fig. 4*, most of the data falls into the category with damage values that have a degree 10^4. To avoid skewing the model, I used random oversampling, which randomly samples data from a category to boost the number of data points in that category [23]. I did this until every category had the same amount of data.

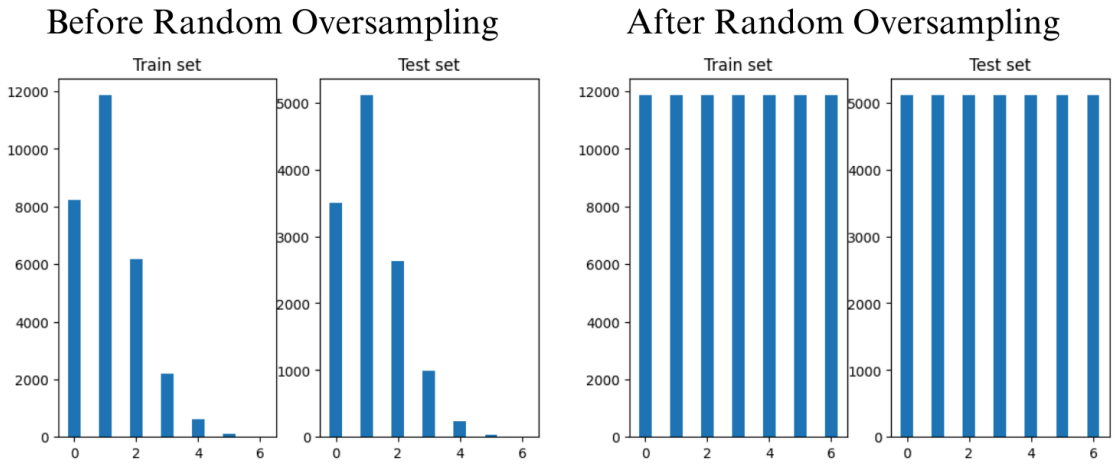

Fig 4. These bar graphs illustrate the frequency of each category for the train and test data before and after random oversampling.

V. Results

A. Tuning the Regression Neural Network

As shown in *Fig. 5*, I modeled the neural network's performance to determine if the model was overfitting. When the model was overfitting, I made adjustments to its hyperparameters. The input layer stayed the same throughout the entire process, as it is standard for the number of neurons in the input layer to equal the number of features in the dataset. I found that the more

hidden layers I added, the more the model was overfitting. As a result, I included two hidden layers with 32 neurons each to reduce the complexity of the model. I determined 0.01 was the best value for the learning rate, as it moderately changed the weights as each epoch progressed. I also tested whether scaling the x values with MinMaxScaler would help improve the predictions as the function scales the data within the range (0,1) to mitigate the effect of outliers. However, using MinMaxScaler caused slight overfitting, so I did not incorporate it into the final model. I chose a kernel initializer (initializes the weights of neurons) of random_normal as it initially samples the weights randomly from a normal distribution. Finally, I used the activation function Rectified Linear Unit (ReLU) because of its simplicity, reducing the training and running time. The ReLU function takes the form $max\ (0.0,\ x)$, when the input is x. It also incorporates non-linearity, which helps us build a more complex model [24]. I also tuned additional hyperparameters such as batch size, epochs, and optimizer.

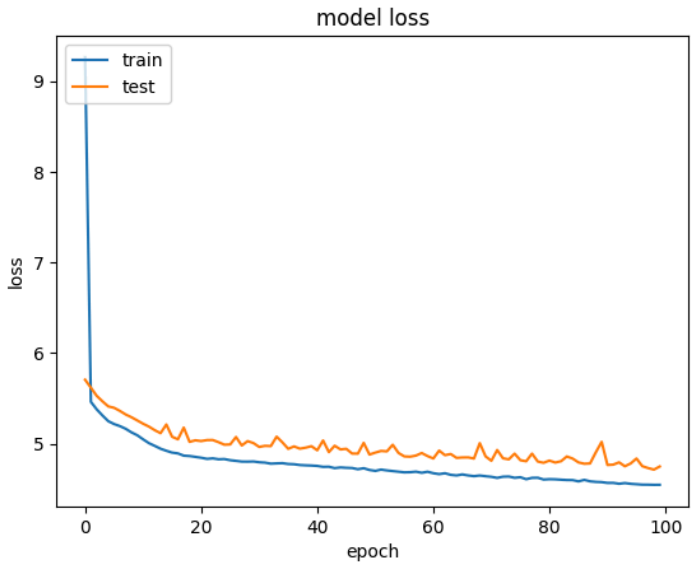

Fig 5. This figure shows the error loss (MSE) as each epoch progresses for the best neural network.

B. Tuning Classification Neural Network

For the classification neural network, I used a slightly different architecture than the regression neural network. I created two hidden layers in the model with 64 neurons each. I also included a dropout layer with a parameter of 0.1 between each layer to prevent overfitting. Dropout randomly drops neurons from the hidden layers and, as a result, reduces bias in the model

because neurons cannot rely on a single input [25]. I chose a sigmoid activation function for the hidden layers because it best predicts probability [26]. I used softmax for the output layer because it transforms the raw outputs of the model into probabilities [27]. Finally, I chose a learning rate of 0.001, so the weights were changed by small amounts as each epoch progressed.

C. Regression Performance Metrics

I used the loss function MSE to quantify the performance of the model. The loss function finds the difference between predicted and actual values [28]. A test of determining whether a model is actually improving compared to simply guessing values is to compare the square root of the mean squared error ($\sqrt{MSE}$) of the predictions to the standard deviation of the test damages (STD). If $\sqrt{MSE} < STD$, the model is doing more than simply predicting the mean of the damages [29]. The standard deviation of the test damages is 5.5468×10^7. As shown in *Table 2*, XGBoost and Gradient Boosting Regression were the only models where $\sqrt{MSE} < STD$, and, therefore, are the best-suited regression models for predicting flood damages.

Model	√MSE	√MSE < STD
XGBoost Regression	5.4932E+07	**Yes**
Gradient Boosting Regression	5.5053E+07	**Yes**
Ridge Regression	5.5480E+07	No
Linear Regression	5.5485E+07	No
Neural Network	5.5546E+07	No
Random Forest Regression	6.1017E+07	No

Table 2. This table shows for which models $\sqrt{MSE} < STD$.

D. Classification Performance Metrics

I graphed the training accuracy and validation accuracy of the model, which both increased steadily. The training accuracy reached 51.14%, and the validation accuracy reached 37.18%. Accuracy is measured by $\frac{correct\ predictions}{total\ predictions} \times 100$. Training accuracy is for the training data, and validation accuracy measures performance on unseen test data [30].

I also used the loss function, categorical cross-entropy, which returns probabilities that a test damage value is in each category [31]. For example, if a test damage value is 10^3 and the model accurately predicts this value, the true value for damages would be $[1, 0, 0, 0, 0, 0, 0]$, but the predicted could be $[0.5, 0.3, 0.05, 0.03, 0.09, 0.01, 0.02]$. Both are accurate, but $[1, 0, 0, 0, 0, 0, 0]$ has a lower loss because it is the true value. Ideally, the loss should decrease, but as shown in *Fig 6.*, the loss values increased as the epochs progressed.

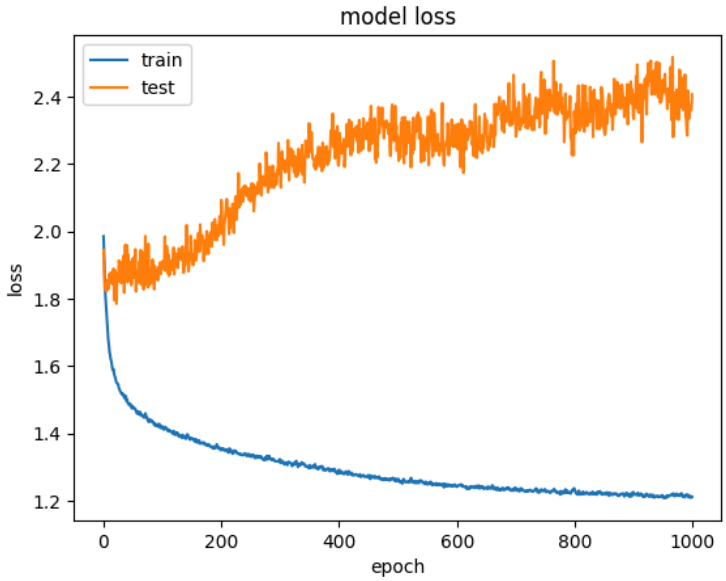

Fig. 6. This figure shows the error loss (categorical cross-entropy) for the model as each epoch progresses.

E. Regression Experimental Results

I plotted the predictions for each model on graphs to qualitatively determine which models had the best results. As shown in *Fig. 7*, Random Forest Regression and Gradient Boosting Regression appear to work best.

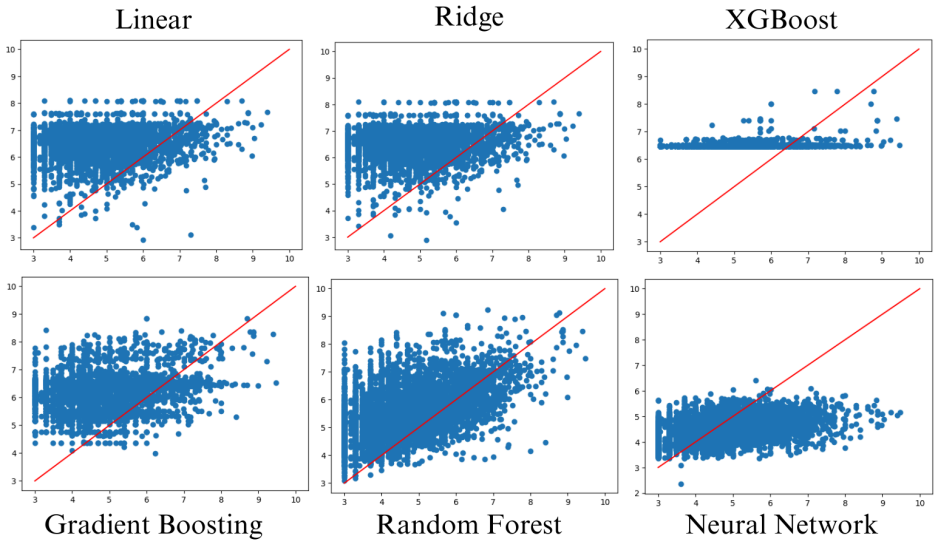

Fig. 7. The x-axes of these graphs are the log of y_test (the accurate values for damages), and the y-axes are the log of the predictions the model outputs given x_test. Each point displays a single sample for the test dataset. In the zero loss scenario, $predictions = y\ test$*, which is displayed by the* $y = x$ *line in red.*

F. Classification Experimental Results

I qualitatively analyzed the results for the classification model using a confusion matrix. By showing what the model predicted within each category, the confusion matrix allows us to determine which degrees of damage are easier to predict. As shown in *Fig. 8*, we found that damages with a degree of 10^3, 10^7, 10^8, and 10^9 were better predicted. We see that there is a more concentrated diagonal for larger values of damage. This shows that classification neural networks have better accuracy for higher monetary damage.

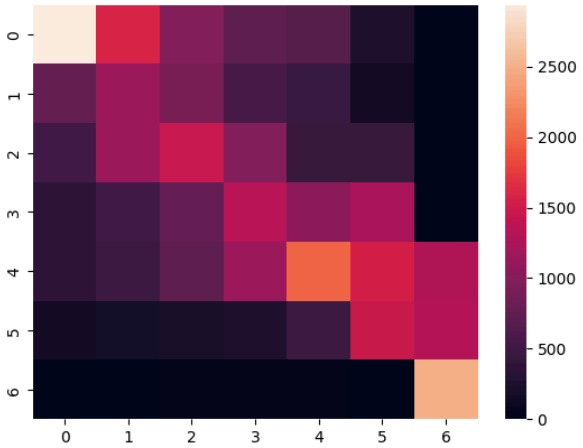

Fig. 8. This correlation matrix shows the frequency of predictions that fell into each category, along with the true value of the predictions. Lighter colors indicate a greater number of samples, and darker colors indicate fewer samples.

VI. Discussion

We found that the Gradient Boosting Regression and XGBoost models worked best. Both models performed better than simply predicting the mean of the damages, and the prediction generally fell along the y=x line. This tells us that decision trees are useful tools for monetary flood prediction and are better predictors than neural networks and other regression models. The regression neural network had an error rate higher than the error for mean prediction. The classification model had a very high loss, but it did predict large values of damages very well. Additionally, random oversampling improved the model's accuracy, which is a good sign because flood data is already very difficult to collect. If we supplement the data with additional datasets that contain a wider range of variables, we could further improve the Gradient Boosting Regression and XGBoost models.

VII. Conclusion

In this paper, we present regression and classification techniques to find viable models that predict monetary damages caused by flooding. Prior work in the field has primarily focused on flood risk instead of flood-related monetary damage. The primary work on economic loss due to flooding was conducted in rural China [12]. Our work seeks to find alternate techniques, features, and models that are useful in predicting economic damage in rural and urban locations in the United States. We find that Gradient Boosting Regression and XGBoost Regression have the best results, upsampling is an effective tool to combat limited flood data, and floods with higher monetary damages are easier to predict with classification neural networks. As climate change progresses, the frequency and impact of flooding will only worsen [32]. There is still much more work to do to create additional tools for communities to obtain more accurate estimates of flood damages [33]. Our work provides a baseline for further flood damage prediction and broader natural disaster research. Future directions could include using richer datasets, variations of Gradient Boosting algorithms, and a more focused range of damages. With these models, citizens can enter information about their homes to discover their risk of flood

damage, insurance companies can improve the accuracy of flood policy underwriting, and governments can determine how best to allocate resources ahead of major flooding events. Further exploration in this field will enable the development of additional tools that can better inform citizens, insurance companies, and governments. This work is a promising step toward better and equitable disaster management decisions.

VIII. Acknowledgments

I want to thank Shreyaa Raghavan for her unwavering dedication to mentoring me. Her expertise in machine learning has been instrumental in shaping the direction of this research. She provided insightful suggestions, constructive feedback, and encouragement, which was critical to successfully completing this project. I am truly grateful for Shreyaa's patience, enthusiasm, and mentorship, which greatly enhanced my research skills and contributed to my personal growth. I would also like to thank Professor Jane Woodward and Ms. Irene Mouzakitis for their support, guidance, and willingness always to say "yes" whenever I needed help. Their commitment to my academic growth and success is so appreciated, and I am immensely grateful for the opportunity to have worked with them.

Bibliography

[1] Flooding is America's most frequent and expensive disaster. (n.d.). https://www.flooddefenders.org/problem

[2] Dow Jones & Company. (2019, June 14). *Is your home at risk of flooding? The data is hard to find*. The Wall Street Journal. https://www.wsj.com/articles/is-your-home-at-risk-of-flooding-the-data-is-hard-to-find-11560418204

[3] Wing, O. (2022, February 1). *New Maps show U.S. flood damage rising 26 percent in next 30 years*. Scientific American. https://www.scientificamerican.com/article/new-maps-show-us-flood-damage-rising-26-percent-in-next-30-years/

[4] Li, Z. (2020). United States Flood Database (v1.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.4355693

[5] Climate change indicators: Coastal flooding | US EPA. (n.d.). https://www.epa.gov/climate-indicators/climate-change-indicators-coastal-flooding

[6] Guterres, A. (2020, July 31). *Climate action and disaster risk reduction*. UNDRR. https://www.undrr.org/climate-action-and-disaster-risk-reduction

[7] Paerl, H. W., Hall, N. S., Hounshell, A. G., Luettich, R. A., Rossignol, K. L., Osburn, C. L., & Bales, J. (2019, July 23). *Recent increase in catastrophic tropical cyclone flooding in coastal North Carolina, USA: Long-term observations suggest a regime shift*. Scientific reports. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6650462/

[8] World Health Organization. (n.d.). *Climate change*. World Health Organization. https://www.who.int/news-room/fact-sheets/detail/climate-change-and-health

[9] Mosavi, A., Ozturk, P., & Chau, K. (2018, October 27). *Flood prediction using Machine Learning Models: Literature review*. MDPI. https://www.mdpi.com/2073-4441/10/11/1536

[10] Wagenaar, D., Curran, A., Balbi, M., Bhardwaj, A., Soden, R., Hartato, E., Mestav Sarica, G., Ruangpan, L., Molinario, G., & Lallemant, D. (2020, April 29). *Invited perspectives: How machine learning will change flood risk and impact assessment*. Natural Hazards and Earth System Sciences. https://nhess.copernicus.org/articles/20/1149/2020/

[11] Rahebeh Abedi, R., Bao Pham, Q., Shafizadeh-Moghadam, H., & Costache, R. (n.d.). Flash-flood susceptibility mapping based on XGBoost, Random Forest and Boosted Regression Trees. https://doi.org/10.1080/10106049.2021.1920636

[12] Chen, A., You, S., Li, J., & Liu, H. (2021, November 2). *The economic loss prediction of flooding based on machine learning and the input-output model*. MDPI. https://www.mdpi.com/2073-4433/12/11/1448

[13] Anwar, A. (2021, June 7). *A Beginner's Guide to Regression Analysis in machine learning*. Medium. https://towardsdatascience.com/a-beginners-guide-to-regression-analysis-in-machine-learning-8a828b491bbf

[14] Deepanshi. (2023, July 20). *All you need to know about your first machine learning model - linear regression*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/05/all-you-need-to-know-about-your-first-machine-learning-model-linear-regression/#:~:text=In%20the%20most%20simple%20words,the%20dependent%20and%20independent%20variable

[15] Friedman, J. H. (2001, April 19). *Greedy Function Approximation: A Gradient Boosting Machine*. https://jerryfriedman.su.domains/ftp/trebst.pdf

[16] *1.1. Linear Models*. scikit. (n.d.). https://scikit-learn.org/stable/modules/linear_model.html#ridge-regression-and-classification

[17] *Random Forest regression*. Random Forest Regression GitBook. (n.d.). https://apple.github.io/turicreate/docs/userguide/supervised-learning/random_forest_regression.html

[18] Chen, T., Guestrin, C. (2016, June 10). *XGBoost: A scalable tree boosting system*. arXiv.org. https://arxiv.org/abs/1603.02754

[19] AL-Ma'amari, M. (2018, October 25). *Deep neural networks for regression problems*. Medium. https://towardsdatascience.com/deep-neural-networks-for-regression-problems-81321897ca33

[20] (1959, November 1). *Diagram of an artificial neural network*. TeX. https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network

[21] Zatout, C. (2023, February 6). *A brief introduction to neural networks : A classification problem*. Medium. https://towardsdatascience.com/a-brief-introduction-to-neural-networks-a-classification-problem-43e68c770081

[22] Trotta, F. (2023, April 8). *How and why performing one-hot encoding in your data science project*. Medium. https://towardsdatascience.com/how-and-why-performing-one-hot-encoding-in-your-data-science-project-a1500ec72d85

[23] Kim, M., & Hwang, K.-B. (2022, July 28). *An empirical evaluation of sampling methods for the classification of Imbalanced Data*. PloS one. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9333262/

[24] Agarap, A. F. (2018, March). Deep learning using rectified linear units (ReLU) - researchgate. https://www.researchgate.net/publication/323956667_Deep_Learning_using_Rectified_Linear_Units_ReLU

[25] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014, June). *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*. https://jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf

[26] Werbos, P. J., Rumelhart, D. E., Hush, D. R., Chiang, C., & Lapedes, A. (1998, May 19). *The generalized sigmoid activation function: Competitive supervised learning*. Information Sciences. https://www.sciencedirect.com/science/article/abs/pii/S0020025596002009

[27] S., R., Bharadwaj, A. S., K, D. S., Khadabadi, M. S., & Jayaprakash, A. (2023, March 7). *Digital Implementation of the Softmax Activation Function and the Inverse Softmax Function*. https://ieeexplore.ieee.org/document/10057747

[28] Seif, G. (2022, February 11). *Understanding the 3 most common loss functions for machine learning regression*. Medium. https://towardsdatascience.com/understanding-the-3-most-common-loss-functions-for-machine-learning-regression-23e0ef3e14d3

[29] Meyer, T. H. (2012, September). Root mean square error compared to, and contrasted with, standard deviation. https://www.researchgate.net/publication/263726816_Root_Mean_Square_Error_Compared_to_and_Contrasted_with_Standard_Deviation

[30] Sharma, P. (2022, July 21). *4 proven tricks to improve your deep learning model's performance*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2019/11/4-tricks-improve-deep-learning-model-performance/

[31] Neuralthreads. (2021, December 26). *Categorical cross-entropy loss - the most important loss function*. Medium. https://neuralthreads.medium.com/categorical-cross-entropy-loss-the-most-important-loss-function-d3792151d05b

[32] *Why are floods hitting more places and people?*. Environmental Defense Fund. (n.d.). https://www.edf.org/why-are-floods-hitting-more-places-and-people#:~:text=According%20to%20new%20research%20from,levels%20and%20more%20intense%20hurricanes

[33] Hersher, R., & Kellman, R. (2020, October 20). *Living in harm's way: Why most flood risk is not disclosed*. NPR. https://www.npr.org/2020/10/20/921132721/living-in-harms-way-why-most-flood-risk-is-not-disclosed