A Supervised Learning Approach to Predicting Regional Maternal Mortality Risk in Nigeria

Sameer Sundrani (sundrani), **Amy Zhang** (ayzhang), **Cora Wendlandt** (coraw)

Full Code is available at: https://github.com/sameersundrani/CS229_FinalProject_Nigeria

1 Introduction

Everyday in 2017, roughly 810 women died due to complications from pregnancy or childbirth [1]. The vast majority of these deaths can be traced back to social determinants such as socioeconomic status, race/ethnicity and health system infrastructure, making maternal mortality a key focus of the global health community for decades. The World Health Organization defines the global maternal mortality ratio (MMR) as the "number of maternal deaths during a given time period per 100,000 live births during the same period", with maternal deaths being defined as the "annual number of female deaths from any cause related to or aggravated by pregnancy or its management during pregnancy and childbirth or within 42 days of termination of pregnancy, irrespective of the duration and site of the pregnancy" [2]. The Sustainable Development Goals (SDGs) set the goal to reduce the global MMR to less than 70 per 100,000 live births by 2030. Maternal mortality largely occurs in low resourced countries with 94% of all maternal deaths occuring in low and lower middle-income countries and Sub-Saharan Africa. Nigeria has the fourth highest MMR in the world with of 917 deaths per 100,000 live births [3]. Many questions surround how to reach the most vulnerable populations, so identifying these populations would help policy-makers and organizations in providing targeted, anticipatory aid.

Machine learning techniques can potentially identify vulnerable and at-risk populations before life-threatening complications occur. Our project aims to evaluate the predictive capabilities of supervised learning algorithms in the context of maternal mortality. Through doing so, we hope to build models that can ultimately identify high risk regions in Nigeria, where we chose Nigeria due to its high MMR and data availability constraints. Accordingly, we will evaluate our models on their ability to reduce their false positive rate at a threshold true positive rate. We use the Demographic Health Survey (DHS) data which includes surveyed households with information including household size, location, and various pieces of personal information [4]. We input to our algorithms a set of features surrounding maternal health (selected using domain knowledge). Next, we use class balancing techniques and variations of logistic regression, SVMs and Extreme Gradient Boosting (XGBoost) to output a binary classification for each validation example and use these aggregated classifications to predict the maternal mortality risk associated with each DHS region.

2 Related Work

To our knowledge, no one has utilized machine learning techniques on DHS data to better understand maternal mortality. There has, however, been similar work on non-DHS data related to maternal mortality. The vast majority of this work is focused on advancing supervised learning techniques to quantify the risk of specific maternal death related outcomes.

Hoffman et al. apply supervised learning techniques like logistic regression and tree algorithms to predict maternal readmission for hypertensive disorder risk [5]. They applied class weighting, oversampling and undersampling to address class imbalance and used XGBoost as their final model due to its ability to handle missing data, insensitivity to features to monotonic transformations and robustness to collinear features. This generated an AUC of 0.85 and 0.81 on the derivation and validation data, respectively. Given the similarities in desired output and their relatively successful performance, we echoed their ML pipeline. We must note that this research is difficult to generalize to low-resourced populations as it was conducted in a high-resource setting using a specific facility's clinical data and involving experts in data collection.

Rodríguez et al. also apply supervised learning techniques like logistic regression and support vector machines (SVMs) to predict the risk of severe maternal morbidity using data collected from a clinic in Colombia [6]. Their logistic regression obtained a 51.8% positive predictive value (PPV) and a 67.7% F1 score for predicting risk in the first semester of pregnancy, and their SVM model obtained a 27% precision score for predicting risk in the third quarter of pregnancy. Similarly, Pan et al. apply random forests, linear discriminant analysis, penalized logistic regression and naive Bayes to quantify adverse birth risk among pregnant women [7]. Their penalized logistic regression model was the most successful with a mean PPV of 0.319. Finally, Westcott et al. applied logistic regression, random forests, XGBoost, and SVMs to identify patients at risk for postpartum hemorrhage during delivery [8]. XGBoost was the most successful with an accuracy of 98.1% and sensitivity of

0.763. These three studies indicate that with more fine tuning and algorithmic variations like weighted LR and weighted SVMs, we can generalize these results to a less specialized dataset such as the DHS. It should also be noted that the data in each of these studies allowed the researchers to choose relevant features using a mix of domain knowledge and feature selection techniques like information gain and correlation. Such techniques are not feasible with DHS data given how the data is encoded, and we will therefore have to rely purely on domain knowledge.

Besides supervised methods, Kitson et al. show that causal inference can be performed on DHS data [9]. They apply Causal Bayesian Networks (CBNs) to better understand childhood mortality with the goal of correctly identifying clinical interventions.This study showed through TABU and FGES scores that their generated networks demonstrated desirable qualities such as replicability, insensitivity to missing values, and well-defined behavior with knowledge-based constraints. A major limitation, though, is that this model does not scale well as the number of variables increases, as this drastically increases the search space of potential graphs. This makes modeling complex health outcomes like maternal mortality difficult as these outcomes rely on a large number of features; however, given the high interpretability of CBNs and importance of explainable results, this is an approach we would like to explore in the future.

3 Dataset and Features

Feature Name	Description	Feature Name	Description
region	Region of Residence	numDaysGivenIron	Given iron tablets during pregnancy
dv_weight	Domestic violence weight	numDaysTakenIron	Number of days took iron tablets
place	Type of place of residence	drugs_parasites	Drugs for intenstinal parasites
age	Current age of respondent	drugs_SP_Fansidar	During pregnancy took SP/Fansidar for malaria
numSons	Sons at home	child_avg_size	Size of child at birth
numDaughters	Daughters at home	child_avg_weight	Weight at birth
numSonsDied	Boys who died	time_spent	Time spent at place of delivery
numDaughtersDied	Girls who died	num_c_sections	Delivery by caesarean section
numChildren	Total children ever born	post_check	Postnatal check before discharge
curPregnant	Currently pregnant	discharge_check	Checkup after discharge from place of delivery
terminated_before	Miscarriage, abortion, or stillbirth	avg_amenorrhea	Period Returned
amount_antenatalcare	Received antenatal care for pregnancy	numSTI	Had an STD in last 12 months
months_antenatalcare	Months pregnant at first antenatal visit	smokerStatus	Currently smokes cigarettes
numVisitsPregnancy	Antenatal visits during pregnancy	distanceToFacility	Distance, to nearby health facility
blood_pressure	Antenatal care: Blood Pressure	notGoAlone	Not wanting to go alone
urine	Antenatal care: Urine sample	hasHealthcare	Covered by any health insurance
blood	Antenatal care: Blood Sample	sexualViolence	Experienced any sexual violence
numTetanusDuring	Tetanus injection during pregnancy	forcedSex	Physically force you to have sexual intercourses
numTetanusBefore	Tetanus injections before pregnancy	isCircumsized	Respondent circumcised

Table 1: Feature Names and Descriptions

Maternal Mortality Labels: The DHS dataset that we use represents data at the household level in Nigeria, where DHS personnel recorded in depth survey information from 41821 women aged 15-49. Each observation is from a woman in a single household, where we have access to 5394 unprocessed features ranging from maternal health information and information about each family member to whether or not their house has a microwave present. To code maternal mortality as a binary feature $y_i \in \{0, 1\}$, we utilize the DHS suggested maternal mortality protocol, where we sum over the number of sisters a particular female respondent has who have died due to pregnancy complications or related reasons, and label our positive class as those with at least one sister meeting these requirements. Mathematically, our label, y_i, for the i'th example is defined as follows:

$$y_i = \mathbf{1}\{numSisterDeaths_i \geq 1\} \tag{1}$$

where we see a total of 647 positive class labels, totalling 1.54$\%$ of the total labels.

Feature Selection: Due to the vast number of unprocessed survey variables, we chose to select relevant possible predictive features ourselves based on the recorded literature on maternal mortality discussed above. From the DHS survey recode, a file consisting of all possible recorded information [10], we selected and then processed the raw survey data to encode 37 features of interest such as the number of sons a respondent has and the birth weight, if any, of their previous children. All data was encoded as either discrete or real valued numbers, with "NA" encoded as a -1. Due to the discontinuous nature of many of our features, imputing the mean or median of a value was infeasible. If a particular feature applied to multiple possible entries per respondent (e.g. child weight across all children), we include only the mean value of that feature. A full list of chosen features shown in Table 1, and our code for processing the data is included in the final report.

Train/Test Splits: For each model described in our analysis, we implemented a 80$\%$ training and 20$\%$ testing split, where our training set consisted of 33456 examples with 526 positive labels and our testing set consisted of 8365 examples with 121 positive labels.

4 Methods

Logistic Regression: We first employed logistic regression with L2 normalization as a baseline model given its success in related works and because it is relatively easy to train and generates interpretable, linear predictions. Logistic regression is a supervised learning technique used for classification. The hypothesis function is defined as follows:

$$h_\theta(x) = \frac{1}{1 + \exp(-\theta^T x)} \tag{2}$$

The cost function is defined as follows in terms of n (the number of training examples), m (the number of features) and λ (a hyperparameter that controls regularization effects):

$$J(\theta) = \frac{1}{n}\sum_{i=1}^{n} Cost(h(x^{(i)}), y^{(i)}) + \frac{\lambda}{2n}\sum_{j=1}^{m}\theta_j^2 \tag{3}$$

$$Cost(h(x), y) = -y\log(h(x)) + -(1-y)\log(1-h(x)) \tag{4}$$

L2 regularization is added to ensure that the model does not overfit to the training data by arbitrarily scaling θ. We chose the regularization scaling constant through testing grid values chosen from a logarithmic scale between 1e-4 and 1e4. The algorithm then uses gradient descent to learn the optimal θ through minimizing this cost function.

Class Balancing Techniques: We also implemented various data balancing techniques to address our dataset's severe class imbalance between the minority class of maternal deaths and majority class of healthy cases. More specifically, we utilized random oversampling and undersampling, synthetic minority oversampling technique (SMOTE) and borderline SMOTE to balance the training dataset. Random oversampling randomly duplicates minority class datapoints while random undersampling randomly deletes majority class datapoints. We combined these two techniques to form a training set, as relying fully on undersampling can remove a large quantity of potentially important information and relying fully on oversampling can result in overfitting the model. Through manual testing, we found that a 0.1 minority to majority class ratio for oversampling combined 0.5 minority to majority class ratio for undersampling performed optimally as a training set. SMOTE randomly selects a minority class point and randomly selects one of its k-nearest neighbors and generates a new minority point that is the convex combination of these two points, repeating until the classes are balanced and resulting in many new synthetically generated minority class points that are close in the feature space to existing minority class points. Borderline SMOTE, while similar, focuses on generating minority class points from other minority class points that are more likely to be misclassified by the k-nearest neighbor classification model (i.e. the points closest to the decision boundary). This results in higher resolution in these ambiguous class overlapping regions. These three balanced training datasets were then used to train their own logistic regression models similar to the first approach and each model was then tested on the normal (unbalanced) test set.

Weighted Logistic Regression: Since logistic regression does not generally take into account class imbalances, we implemented a weighted logistic regression model with 5-fold cross validation (optimizing for ROC-AUC). This model works like logistic regression but penalizes the model for incorrectly classified points according to their respective class weights. We implemented two approaches for determining class weights. The first was using the heuristic of defining weights as the inverse of class frequency. 5-fold cross validation was then used to train this model. Cross validation is a technique used to reduce the variance of models through splitting the training data into k-folds and generally training on k-1 folds and validating on the remaining fold. This process is repeated through rotating the validation fold through the rest of the k-1 folds to generate k models. The model with the best performance according to the determined evaluation metric is selected as the best model. The second approach for determining class weights was using a grid search with 5-fold cross validation to find the most optimal weight with respect to the ROC-AUC score.

Cost Sensitive SVMs: To address the likelihood that our data is not linearly separable, we also implemented a SVM using the radial basis function kernel. SVMs are also supervised learning techniques that can be used for classification. The objective function in terms of s_i (the slack variable for each example) and C (the soft-margin parameter) is as follows:

$$min\frac{1}{2}||w||^2 + C\sum_{i=1}^{n} s_i \tag{5}$$

subject to the constraint:

$$y_i(w \cdot x_i + b) \geq 1 - s_i \;\; \forall x_i,\; s_i \geq 0 \tag{6}$$

The slack variables corresponding to each training example and the soft-margin parameter are necessary if the dataset is not perfectly linearly separable as they allow for some examples to be misclassified with a penalty. Through utilizing the kernel trick, SVMs are able to translate training examples to much higher (in the case of the radial basis function, infinitely higher) dimensions and calculate the decision boundary that maximizes the margins between the boundary and the closest examples. As SVMs do not generally take into account class imbalance, we implemented a cost sensitive SVM through weighting the soft-margin parameter (which controls "the trade-off between maximizing the separation margin between classes and minimizing the number of classified instances") according to class weights proportional to class frequency [11].

XGBoost: Finally, we employ XGBoost, a popular model based on decision tree ensembles, or a set of classification trees where leaf values are summed to give a final prediction for a particular respondent [12]. The objective function for XGBoost is as follows:

$$obj = \sum_{i}^{n} l(y_i, \hat{y}_i) + \sum_{k}^{K} \Omega(f_k) \tag{7}$$

where $\hat{y}_i$ is our model output, l is our loss, f is a function over the functional space of possible sets of trees, $\Omega(f_k)$ represents the complexity of a tree, and model training is performed by learning one tree, f_k, at a time.

We tune our model's hyper-parameters using 5-fold class stratified cross-validation and a randomized grid search over central parameters to XGBoost such as the maximum depth of each tree, the positive weight applied to our imbalanced class distribution, and the number of estimators we allow in the model. After tuning, we then manually increased the model's regularization parameters α and λ and lowered the learning rate, η, after noticing possible over-fitting on the training data.

5 Experiments, Results, and Discussion

Evaluation Metric: Before discussing our results we must quantify our primary evaluation metric. We are trying to predict maternal mortality in Nigeria. We see that the SDG for the global MMR is 70 deaths per $100,000$ live births and, in 2018, Nigeria had 917 deaths per $100,000$ live births. Nigeria must reduce its MMR by 92% in order to meet the SDG. Thus, our model will need to have a true positive rate (TPR) threshold of 92% to capture enough cases such that intervention is possible and hopefully successful. We will thus evaluate our models based on the false positive rate (FPR) at this given threshold for TPR.

$$TPR = \frac{TP}{TP + FN} \tag{8}$$

$$FPR = \frac{FP}{FP + TN} \tag{9}$$

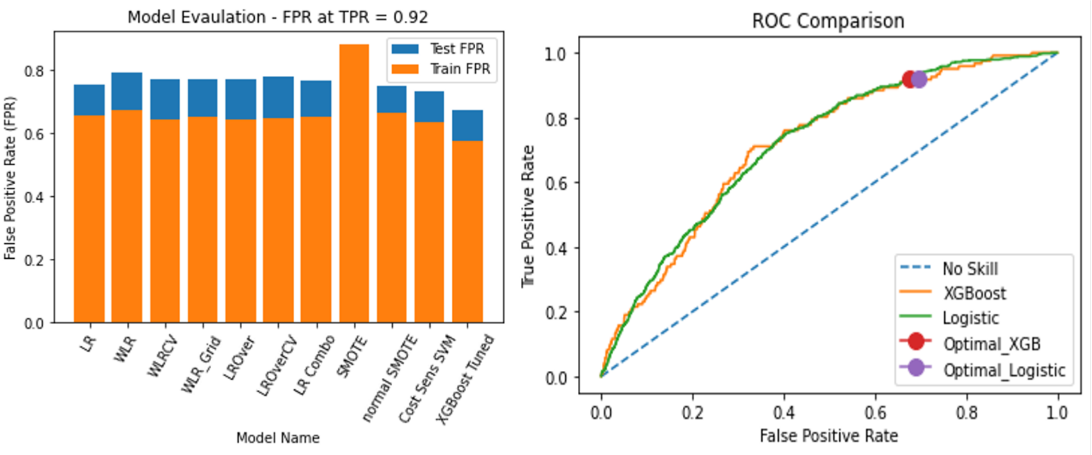

Figure 1: (left) Train/test FPR of each model evaluated at 92% TPR and (right) ROC comparison between baseline LR and final tuned XGBoost model with optimal thresholds at 92% TPR (AUROC's were 0.718 and 0.691 for XGBoost and Logistic Regression, respectively)

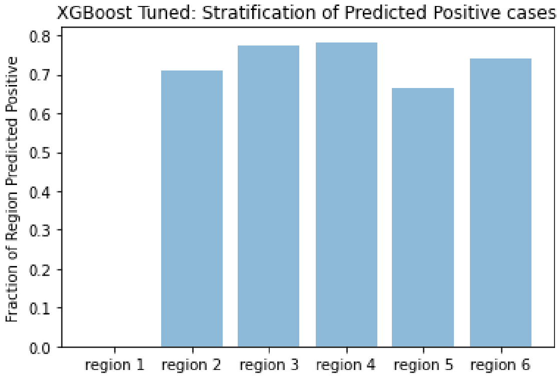

Figure 2: Regional stratification of positively-predicted risk of maternal mortality

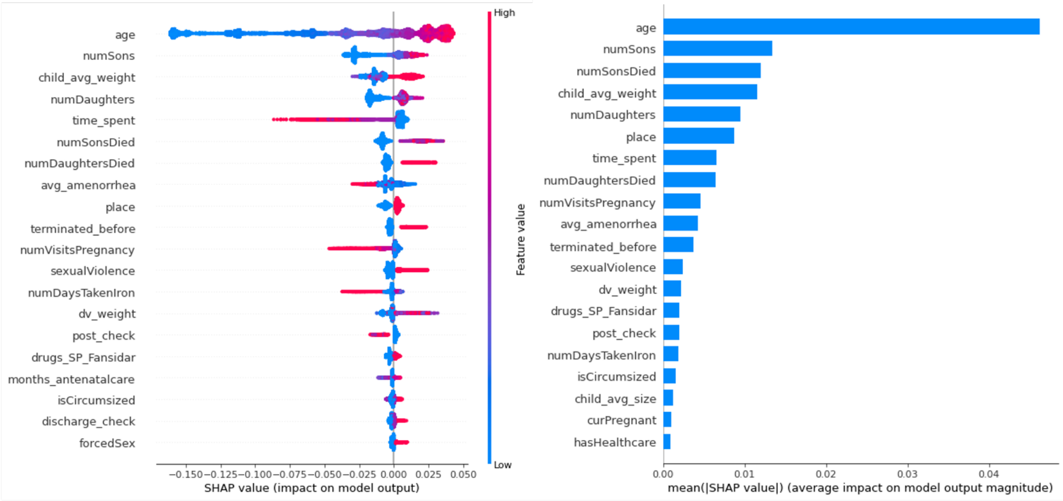

Figure 3: (left) SHAP value model impact plot and (right) XGBoost feature importance ranked by mean absolute value

Regional Stratification of Maternal Mortality: Given our best performing model, XGBoost, we see that it predicted no positive cases for region 1 and the region with the largest fraction of predicted positive cases is region 4, with region 3 as a close second. The lack of positive predicted cases for region 1 may likely be a data issue correlated with accessibility into the region and survey methods. We plan to look into this in the future. Furthermore, we would have liked to conduct a qualitative evaluation of these results through identifying geographic, cultural and/or economic patterns correlated with maternal deaths in these reportedly high risk regions from additional DHS data, but we were not approved for access early enough.

Model Interpretation: From the SHAP values and feature importance rankings in Figure 3, we see that many of the resulting importance rankings are in line with the current domain knowledge, with age ranking the highest on model impact, followed by child-based indicators, both of which hold high importance for maternal health in general. Interestingly, the hasHealthCare feature ranks low, which may be due to traditional healthcare coverage in Nigeria being less important for pregnant women given the strong culture around traditional birthing practices with midwives. Given these, however, we should note that our model has many limitations around bucketing risk and a potential lack of potential important features.

Bias-Variance Tradeoff: We now evaluate our models in terms of the bias-variance tradeoff. Our baseline model of logistic regression achieves a FPR of 65.9% on the training set and 75.3% on the testing set. We see that, although the model performs 10% better on the training versus testing set, both FPR's are poor indicating that our baseline model suffers from high bias. As a result, we decided to implement variations on logistic regression, a support vector machine and XGBoost as shown above. Our best results came from the XGBoost model which yielded a FPR of 57.5% on the training set and 67.6% on the testing set at our TPR threshold. XGBoost allows for non-linearity with the increase of decision trees and their decision boundaries rather than a series of weights, so the elevated complexity reduces the bias within our model. However, we are still in a high bias situation, given that XGBoost still has a FPR of greater than 50% on both the training and testing sets.

Error Analysis: Next, we evaluate the error in our best performing model: XGBoost. XGBoost's final FPR at a 92% TPR was 67.6% on the test data, with the threshold probability for binary classification at 0.50474846. We found that our model misclassified 5577/8365 examples in our test set at this threshold. To understand our model's error performance, we compared the top 10 most confident output probabilities for both the positive and negative class. It should be noted that while we examine the most confident misclassifications, our misclassification probabilities span a narrow range from 0.464 to 0.556. In spite of a small range, there were significant differences between the top 10 false positives and false negatives. We see that the mean age for our false positives was 40.9 years while for the false negatives it was only 18.3 years. This discrepancy makes sense though, as age has both the largest spectrum for impact on the model output and generally younger women will have lower risk for maternal mortality, which is confirmed when inspecting some of the decision trees generated where "age < 19" is the first decision boundary. In addition, we see that the mean number of sons for women in the false positive class was 2.9 while in the false negative class it was 0, with only 1 woman in the false negative class experiencing childbirth herself. XGBoost may therefore be associating higher probability to those with greater number of births in a particular household, and our ranked SHAP value analysis confirms this finding.

6 Conclusion and Future Work

In conclusion, we see that our final XGBoost model does suffer from high bias, but was able to reduce that bias from the baseline model with its increased complexity. Although we had hoped that our final model would aid in informing governments and aid organizations of regions with high MMR, the performance of our models show that there is still necessary progress to be made before the predictive capabilities of these techniques can be trusted for clinical or aid interventions. For future work, we would like to consider various feature mappings (such as polynomial mapping) or adding more features from the DHS survey data. Additionally, we would like to expand on the regional bar graphs to obtain more specific geographical locations to see if certain areas are experiencing more maternal mortality than others and should receive more support.

Contributions

We collaborated equally across the project, consistently meeting to discuss progress and future steps.

References

[1] “Maternal Mortality.” World Health Organization, World Health Organization, www.who.int/news-room/fact-sheets/detail/maternal-mortality.

[2] “Indicator Metadata Registry Details.” World Health Organization, World Health Organization, www.who.int/data/gho/indicator-metadata-registry/imr-details/26.

[3] “Maternal Morality Rate Country Comparison.” Central Intelligence Agency, Central Intelligence Agency, www.cia.gov/the-world-factbook/field/maternal-mortality-rate/country-comparison.

[4] “The DHS Program.” The DHS Program - Nigeria: Standard DHS, 2018, www.dhsprogram.com/methodology/survey/survey-display-528.cfm.

[5] Hoffman, Matthew K., et al. “A Machine Learning Algorithm for Predicting Maternal Readmission for Hypertensive Disorders of Pregnancy.” American Journal of Obstetrics amp; Gynecology MFM, vol. 3, no. 1, 2021, p. 100250., doi:10.1016/j.ajogmf.2020.100250.

[6] Arrieta Rodríguez, Eugenia, et al. “A Machine Learning Approach for Severe Maternal Morbidity Prediction at Rafael Calvo Clinic in Cartagena-Colombia.” Computer Information Systems and Industrial Management, 2020, pp. 208–219., doi:10.1007/978-3-030-47679-3_18.

[7] Pan, Ian et al. “Machine Learning for Social Services: A Study of Prenatal Case Management in Illinois.” American journal of public health vol. 107,6 (2017): 938-944. doi:10.2105/AJPH.2017.303711

[8] Westcott, Jill M., et al. “Prediction of Maternal Hemorrhage: Using Machine Learning to Identify Patients at Risk.” 2020, doi:10.1101/2020.06.04.20122663.

[9] Kitson, Neville Kenneth, and Anthony C. Constantinou. “Learning Bayesian Networks from Demographic and Health Survey Data.” Journal of Biomedical Informatics, vol. 113, 2021, p. 103588., doi:10.1016/j.jbi.2020.103588.

[10] "The DHS Program.”, https://www.dhsprogram.com/pubs/pdf/DHSG4/Recode7_DHS_10Sep2018_DHSG4.pdf

[11] Brownlee, Jason. “Cost-Sensitive SVM for Imbalanced Classification.” Machine Learning Mastery, 20 Aug. 2020, machinelearningmastery.com/cost-sensitive-svm-for-imbalanced-classification/.

Python Packages:

[12] Chen, T., Guestrin, C. (2016). "XGBoost: A Scalable Tree Boosting System." In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). New York, NY, USA: ACM. https://doi.org/10.1145/2939672.2939785

[13] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011

[14] Lundberg, S., Lee, S.-I., 2017. "A Unified Approach to Interpreting Model Predictions." arXiv:1705.07874 [cs, stat].

[15] John D. Hunter. Matplotlib: "A 2D Graphics Environment, Computing in Science Engineering," 9, 90-95 (2007), DOI:10.1109/MCSE.2007.55

[16] Wes McKinney. "Data Structures for Statistical Computing in Python," Proceedings of the 9th Python in Science Conference, 51-56 (2010)

[17] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke Travis E. Oliphant. "Array programming with NumPy," Nature, 585, 357–362 (2020), DOI:10.1038/s41586-020-2649-2