Lawrence Technological University
Department of Math and Computer Science
MCS 5623 Machine Learning
Assignment # 04
Michael Giuliani
10/20/2024

I. INTRODUCTION & DOMAIN KNOWLEDGE

In this assignment, a dataset from University of California Irvine (UCI), called "Default of Credit Card Clients" [1], is used to write two different methods of predicting if a given client will default on payment of a credit card bill. The characteristics of the data are analyzed, the preprocessing required to make the data useful is determined, insights are drawn about what data points may or may not be useful in achieving the prediction goals, and visualizations of the data are explored to gain a better understanding of what relationships can be found in the data to make better predictions.

To make the predictions, two methods are explored: the linear regression classifier, and the random forest ensemble classifier. A linear regression classifier attempts to identify two classes from each other by calculating a dividing line between them and predicting the class of a sample based on which side of the line it is on [2]. A random forest ensemble classifier builds a group of decision trees based on randomly selected samples and features in the dataset, where the prediction is the averaged result of all trees [3]. These methods are compared and contrasted to see how their performance compares, what they do better or worse, and what can be changed for each to improve performance.

II. DATASET ANALYSIS & UNDERSTANDING

A. Data Characteristics

The "Default of Credit Card Clients" dataset from UCI provides data on thirty thousand individuals, detailing whether they have defaulted or not, six months of credit payment and balance history, as well as other statistics like age, sex, education level, and marital status. All of the data is numerically encoded and was preprocessed by UCI for any missing values [1].

The data is very imbalanced, with 77.87% not defaulted and 22.12% defaulted. Training techniques need to be used to avoid skewing predictions too much toward the more represented class [4]. When making predictions on this dataset, the performance is expected to be in the 70% - 80% accuracy range at the least, since if the model predicts "Not Defaulted" for every sample, it can be 77.87% accurate.

After cleaning and analysis, some columns were found that were more useful for classification, and some that were not needed at all. There were some rows that could be deleted as well, which were not found until several preprocessing steps were taken.

B. Feature Analysis & Selection

To select features for classification, the best relationship to look at is how the features correlate to the class labels. Figure 1 shows a correlation heatmap between every column in the dataset, where red is a strong positive correlation, white is no correlation, and blue is strong negative correlation. Since the goal is to classify whether someone will default, the row correlating the class labels to each column should be the main focus.

It becomes immediately clear from the correlation values that most of the data in this dataset is not useful for classifying if someone will default. Sex, education level, marital status, age, bill amounts, and payment amounts all have correlation values in the hundredths, if not thousandths, while 1.00 or -1.00 represents perfect correlation.

It also becomes clear which columns are useful. The columns giving data on how many months behind or ahead the person was on payments each month are ranging 0.19 to 0.32. Relatively high, but still low. The limit balance column has a slightly larger negative correlation to the class labels than other columns at -0.15, but it is not high enough to be useful. For the best classification results with the given data, the first payment history column was used, labeled "PAY_1". This column was chosen because it has the highest correlation of any other column to the class labels at 0.32.

C. Data Cleaning/Preprocessing

The dataset was already made quite clean and ready for use by UCI, but there was more to be done. Every column was labeled by UCI with an 'X' followed by a number starting from one. There is also a row in the dataset that gives a more readable label for the column. For example, column "X5" has the label "AGE" in its description row. The more readable column names from that description row were used to replace the column names given by UCI. Doing this results in the description row being duplicated, so after renaming the columns that row was deleted.

After doing this it was found that the first payment history column was labeled "PAY_0", while the next was "PAY_2". The payment and balance size history columns begin at one, so "PAY_0" was renamed to "PAY_1". The dataset also contains a column labeled "ID" that simply numbers each row sequentially, which is not useful for this application, so that column was deleted.

Only after taking the steps outlined above, checking for duplicate values revealed thirty-five duplicate rows. Duplicate training samples can lead to those samples having undue influence on the predictions, so we delete them to keep each training sample balanced. Since the dataset is relatively large at thirty-thousand samples, deleting thirty-five of them is not an issue.

Since UCI did most of the preprocessing themselves, techniques like encoding of categorical columns did not need to be taken. Encoding non-numeric categories allows us to use them in classifiers and other AI algorithms since these algorithms rely on numeric data [5]. Encoding replaces the categorical data with a numeric representation of some sort to represent each unique category, with that numbering technique varying depending on the application. UCI already did this for this dataset.

D. Data Visualization

Figure 1 shows a cross-correlation heatmap of the dataset. The correlation to the "default payment next month" row is what matters for this assignment. Most columns have negligible correlation with it, and are therefore not useful for classification. The payment history columns have the best correlation, with "PAY_1" having the highest.

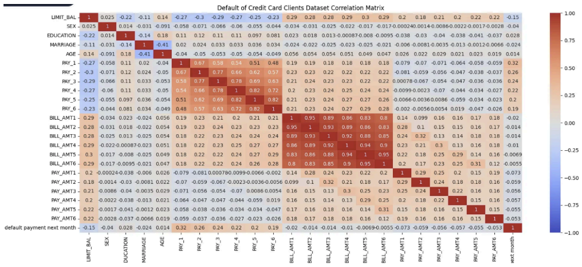

Fig. 1. Cross-Correlation Matrix Heatmap of the "Default of Credit Card Clients" Dataset. [6]

Figure 2 shows the age distribution of the dataset and how many people of each age defaulted or did not default. The ratio of defaulted to not defaulted is consistent and only shrinks a small amount in the older ages. This explains why age had little correlation with whether someone defaulted.

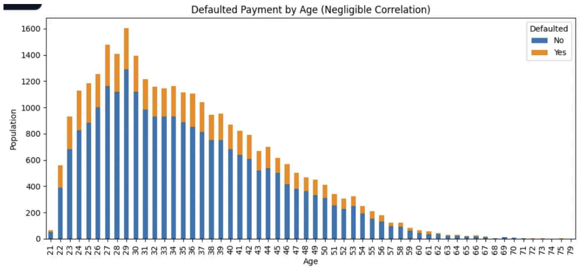

Fig. 2. Population Defaulted (Blue) or Not Defaulted (Orange) by Age. [7]

Figure 3 shows how many months behind or ahead the population was on their payments, and whether the people at each monthly increment defaulted or not. The histories range from two months ahead to eight months behind. As the number of months behind grows, the portion of people that defaulted grows significantly. The columns providing this data had relatively high correlation to whether the person defaulted, so they were most effective for classification training.

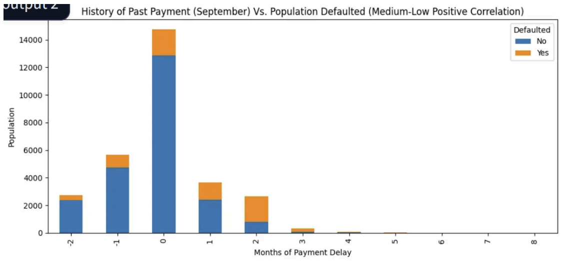

Fig. 3. Population Defaulted (Blue) or Not Defaulted (Orange) Versus Months of Payment Delay. [7]

Figure 4 shows the credit account balance limit distribution of the dataset, and how many people did or did not default at each limit amount. This data had slight negative correlation with whether the person defaulted. It is clear that people with low limits defaulted more often, which explains the correlation.

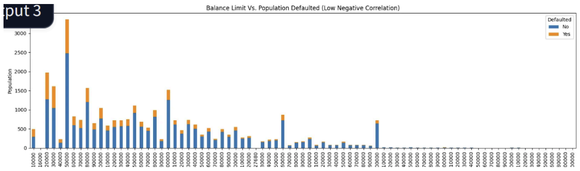

Fig. 4. Population Defaulted (Blue) or Not Defaulted (Orange) Versus Credit Limit Balance. [7]

Figure 5 shows the imbalance of the classes in the dataset. Out of 29,965 samples in the preprocessed dataset, 23,335 samples (77.87%) represent Not Defaulted (0), and 6630 (22.12%) represent Defaulted (1). As a result, the models will be biased towards predicting that someone will not default, since in a worst case, only predicting that someone will not default with this dataset will result in 77.87% accuracy.

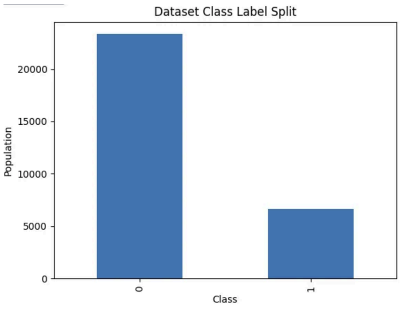

Fig. 5. Population Label Split of Not Defaulted (0) or Defaulted (1). [7]

References

[1] I. Yeh. "Default of Credit Card Clients," UCI Machine Learning Repository, 2009. [Online]. Available: https://doi.org/10.24432/C55S3H.

[2] Wikipedia Contributors, “Linear classifier,” Wikipedia, Aug.29, 2019. https://en.wikipedia.org/wiki/Linear_classifier

[3] Wikipedia Contributors, “Random forest,” *Wikipedia*, Apr. 09, 2019. https://en.wikipedia.org/wiki/Random_forest

[4] J. Brownlee, “A Gentle Introduction to Imbalanced Classification,” *Machine Learning Mastery*, Dec. 22, 2019. https://machinelearningmastery.com/what-is-imbalanced-classification/

[5] K. N. Jarapala, “Categorical Data Encoding Techniques,” *AI Skunks*, Mar. 27, 2023. https://medium.com/aiskunks/categorical-data-encoding-techniques-d6296697a40f

[6] M. Waskom, “seaborn: statistical data visualization,” *Journal of Open Source Software*, vol. 6, no. 60, p. 3021, Apr. 2021, Available:

[7] J. D. Hunter, "Matplotlib: A 2D Graphics Environment", Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007.https://joss.theoj.org/papers/10.21105/joss.03021

[8] “3.8.20 Documentation,” *Python.org*, 2024. https://docs.python.org/3.8/index.html (accessed Oct. 13, 2024).

[9] Harris, C.R., Millman, K.J., van der Walt, S.J. et al. *Array programming with NumPy*. Nature 585, 357–362 (2020). DOI: 10.1038/s41586-020-2649-2.

[10] “API reference — pandas 1.3.3 documentation,” pandas.pydata.org. https://pandas.pydata.org/docs/reference/index.html#api

[11] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” *J. Mach. Learn. Res.*, vol. 12, 2011, doi: https://doi.org/10.5555/1953048.2078195.

[12] G. Team, “Gradio Documentation,” *www.gradio.app*. https://www.gradio.app/docs