**Predicting Success and Default in SBA Loans: A Data-Driven Approach to Small Business Loans**

****

**Group A5**

**Gavin Campbell, Bella Davies, Brandon Wong, William Li, Eric Gulotty**

**BA305 Spring 2024 - Prof. Karaca**

**21APR2024**

**Table Of Contents**

1. **Introduction ………………………………………………………………………….… 4**
2. **Data Description ………………………………………………………………….….… 4**
3. **Methodology …………………………………………………………….……………… 6**
   1. **Dimensionality Reduction ………………………………………………………6**
   2. **Train/Validation/Test Split………………………………………………………7**
   3. **Baseline Model …………………………………………………………………. 8**
   4. **Logistic Regression ……………………………………………………………...8**
   5. **Decision Trees - Full, Reduced, and Grid Searched ……...…………………...8**
   6. **Random Forests …………………………………………………………………9**
   7. **Gradient Boosting Classifier ……………………………………………………9**
4. **Findings and Implications ……………………………………………………………..10**
5. **Conclusion …………………………………………………………………………..… 12**
6. **Appendix** 
   1. **Appendix 1. Cleaned Dataframe………………………………………………13**
   2. **Appendix 2. Heatmap of Feature Correlations……………………………….13**
   3. **Appendix 3. Principal Component Analysis…………………………….…….14**
   4. **Appendix 4. PCA Scree Test Plot ……………………………………………...14**
   5. **Appendix 5. Results of Models with PCA vs. Without PCA…………………15**
   6. **Appendix 6. Grid Searched Decision Tree Confusion Matrix……………….16**
   7. **Appendix 7. Gradient Boosting Tree Confusion Matrix……………………..16**
   8. **Appendix 8. With PCA - Feature Importances of Gradient Boosting………16**
   9. **Appendix 9. Without PCA - Feature Importances of Gradient Boosting…...17**
   10. **Appendix 10. Without PCA - Feature Importances of Decision Tree……….17**
   11. **Appendix 11. Top Splits of Grid Search (Best) Decision Tree………………..18**
   12. **Appendix 12. Distribution of NAICS Industries……………………………...18**
   13. **Appendix 13. Quantiles and Distribution of Term……………………………19**
   14. **Appendix 14. Quantiles and Distribution of Disbursement………………….19**
   15. **Appendix 15. Results of Testing Ex. Business Loans on Decision Tree ……..20**
   16. **Appendix 16. Tree Prediction Paths of Tested Example Business Loans .….21**
   17. **Appendix 17. Data Dictionary…………………………………………………22**

**Introduction**

Small businesses are the backbone of the American economy and play a vital role in keeping workers employed. For such businesses, securing additional funding from external parties to expand their operations can be the key to achieving lasting success. The U.S. Small Business Administration (SBA), founded in 1953 on the principle of promoting and assisting small enterprises in the U.S. credit market, aims to aid this process through the provision of what are known as SBA loans. The beneficiaries of this program sometimes achieve resounding success and become household names, such as Apple and FedEx, while some less fortunate businesses fell on hard times and defaulted on their SBA-guaranteed loans. Our dataset, hosted on Kaggle, is taken directly from the SBA and contains numerous characteristics of small businesses that have been awarded SBA-guaranteed loans between the years 1987 and 2014. Using this dataset, we hope to extract the most important features of successful small businesses, alongside the leading causes for a default. With these insights, our goal is to provide small business owners with a degree of guidance about particular aspects of their business that can be most impactful and further optimized to ensure the successful repayment of their SBA loan. To perform this analysis, our group used four separate methods: logistic regression, decision trees, random forests, and gradient-boosted trees. We decided on these methods due to the high interpretability of each model’s output, which aligns well with our stated goal of providing small businesses with actionable insights that increase their chances of repaying SBA loans in full.

**Data Description**

Our group decided to work on a data source from the US Small Business Administration (SBA). This data source link can be found [here](https://www.kaggle.com/datasets/mirbektoktogaraev/should-this-loan-be-approved-or-denied). This dataset was gathered to help students learn statistics. Originally, it was supposed to be a case study where a student undertook the role of a bank loan officer and would have to create a model in which they would approve or deny a given company's loan package. The US SBA was founded in 1953 to assist small businesses acquire funding, as they feel that they are the primary means of job description in the US. The SBA assists small business enterprises through a loan-guaranteeing program, taking over some of the risk by providing collateral to the bank should the business default on its original loan. This dataset collected historical data from 1987 to 2014 from their primary source data of helping these companies out with securing their enterprise material.

The original dataset has 899,164 observations with 27 different variables. The majority of these variables are numerical in nature, however are categorical data points. Examples of these categorical data points that are not super intuitive are the NAICS codes. These are codes that are published by US Census data which places the business into large buckets of industry based on the first two digits of the code. Examples of these would be agriculture, utilities, information, etc. Our group decided to dummy code all of these variables in these big buckets of industry. This would allow us during our analysis to delineate whether, within this dataset, any of the individual industries had a significant impact on whether or not the loan would be accepted.

The next dummy variables to be cleaned in this dataset are NewExist and LowDoc. NewExist represents how new the business is, with 1 representing a business that is considered to already be "existing" for two or more years, while 2 represents a business that has existed for 2 or fewer years.

LowDoc was a categorical variable that our group dummy coded into 1 for yes and 0 for no. The LowDoc loan program was implemented where loans under $150k could be processed using a one-page application. This would allow us to distinguish whether or not this short-form loan option had any impact on the acceptance of the loan with all other variables held constant.

Finally, the last variable that was not super intuitive was MIS\_Status. This delineated whether or not the loan had been defaulted on or charged off (MIS\_Status= 1), or if the loan had been successfully paid off in full (MIS\_Status = 0) within the timeframe that was agreed upon at the conception of the loan. This is what our group used as our decision (y) variable to measure the impact of all the other variables to determine which would be the most impactful from the perspective of a small company when applying for a loan in their respective industry.

The further steps that our group took for preprocessing are as follows. We dropped columns that we deemed would not be useful or too specific (ie loan ID, business name, city, date), and dropped all rows with missing values. Next, we had to enumerate the numerical variables that were not strictly integers yet (as in the dataset they were strings with dollar signs or decimal places). Next, as discussed previously we changed all of the binary categorical y/n variables into 0s and 1s to allow us to run various models on them (RevLineCr, Franchise, UrbanRural, NewExist, LowDoc, and MIS\_Status). Next, we created dummy variables for all the buckets delineated by NAICS, as well as we created a new variable which was a binary 0/1 categorical variable, delineating whether or not a certain loan was applied for during the period of the great recession in the United States from December 2007-June 2009, as we felt that this might have some impact on loan acceptance and wanted to take that into account. In summary, our numeric variables remaining are Term, NoEmp, CreateJob, RetainedJob, DisbursementGross, BalanceGross, GrAppv, and SBA\_Appv. The remaining categorical dummy variables are Recession, NewExist, Franchise, UrbanRural, RevLineCr, LowDoc, MIS\_Status, and the NAICS industry codes. The data dictionary with definitions for all variables in our cleaned dataset is shown in *Appendix 17*, the final data frame is shown in *Appendix 1,* including 547,698 observations. The majority of our variables are shown in the top row, and the full cleaned dataset can be found in the ipynb notebook file.

**Methodology**

***Dimensionality Reduction***

Before training the models, we wanted to examine the correlations between features in the dataset.We ran an initial heatmap on the variables to see which ones were over-correlated. We chose to temporarily exclude the NAICS codes along with other categorical variables as dimension reduction works best on numerical metric variables. A breakdown of the heatmap can be found in *Appendix 2*, where it delineates. Upon examining correlations between numeric variables, we observed that there were high correlations between Term and DisbursementGross (0.5), Term and GrAppV (0.5), and Term and SBA\_Appv (0.6).

When running PCA on the numeric variables, a scree test using the point of inflection indicated that we should use the first two principal components, component0 and component1. The results of PCA’s explained variance, cumulative variance, and breakdown of PC’s are shown in *Appendix 3* and the scree test plot is shown in *Appendix 4.* From these results, we have identified using the point of inflection to use two principal components. We have defined those principal components as PC 0 (Propensity for the loan to be larger) which explains 42.1% of the variance, and PC 1 (Propensity for the borrowing business to be larger) which explains 15.5% of the variance. PC 0 represents companies that take loans with longer terms, higher disbursement amount (delivery of money from lender to borrower), and more gross guaranteed as well as SBA amount (a substitute for collateral and will pay out to the lender in the case that the borrower defaults on the loan). PC 1 represents larger businesses that create and retain more jobs. Together, PC 0 and PC 1 explain 57.6% of the cumulative variance.

In comparing the selected models’ performance on the validation dataset with and without dimension reduction, we found that the models performed better without PCA. The results of all models with and without PCA can be seen in *Appendix 5*. We also tried PCs up to 6 however to cover more variance however these also performed worse than the training data without PCs. Furthermore, in examining feature importances with and without PCA in *Appendix 8* and *Appendix 9*, we can see that information about each numerical variable is lost when replacing them with PCs, especially with the “Term” variable as this has 0.8 importance while “F1” is the first principal component has 0.6 which is less than 0.8, indicating that PCA is not necessary because the “Term” variable should be kept in the dataset due to its high importance. Moreover, the dataset already only contains 8 numerical metric variables and many categorical dummy variables, so it makes sense that reducing the dimensions of the metric variables further may not improve the model. Due to these reasons, we decided to run all models without principal component analysis.

***Train/Validation/Test Split***

We split the dataset into training, test, and validation sets. We first split the dataset into 80% training data and 20% test data. Then we split 25% of the training data for our validation data. The validation is used to improve the model and tune hyperparameters, while the test set is not used until the end to evaluate the final model's accuracy. Our resulting split is 60% training data, 20% testing data, and 20% validation data. While the training data is only 60%, we have 547,698 total observations in our total dataset allowing for sufficient training data to train the model. The following final models have been tested on the test set after tuning on the validation set. This ensures that the model has not seen the test data and parameters have not been changed from the results of the test dataset.

***Baseline Model***

We assigned the same prediction of 0 (Paid in full) to all instances and evaluated the accuracy of all identical predictions. This is done to provide a baseline accuracy score of 76.12%, as any machine learning model tried should perform better than the naive model since the naive model assumes conditional independence on the training dataset, however, our dataset shows significant correlations and the naive model would not be the best model. Therefore we will explore further models to make better predictions than the naive model, which follows inaccurate assumptions about independence for our data.

***Logistic Regression***

We explored Logistic Regression, a commonly used model for binary classification. For this model, the odds ratio of the coefficients would allow us to see the expected change in odds that the loan is Charged Off (MIS\_Status = 1) for a one-unit increase in each independent variable Xi. Our first logistic regression model was used to adjust penalty types (L1, L2) and maximum iterations which resulted in an accuracy of 76.1%. For our second model, we experimented with the best cutoff score to find the best cutoff by iterating over a range of 0.1 to 0.9. The best cutoff was found to be 0.3, which gave these performance metrics: Accuracy = 80.5%, Precision = 61.2%, Recall = 50.8%, F1 = 55.5%.

***Decision Trees - Full, Reduced, and Grid Searched***

Decision Trees were chosen due to their interpretability, providing a general set of “rules'' or decision paths for why a small business loan may be paid in full or charged off. Furthermore, inputs for decision trees allow for a mixture of categorical and numerical variables, aligning well with our database. Additionally, by listing feature importances, we would also be able to identify the most important variables for determining whether a loan is paid in full or charged off.

To establish a decision tree baseline, we started with a full tree with 56479 nodes, 28240 leaves, a max depth of 52, and an accuracy of 89.8%. This was a great starting point as its accuracy was significantly higher than our naive baseline. But we had to take into account the possibility of overfitting. So we implemented a Reduced Tree and a Grid Search Tree. For the Reduced Tree, we implemented these limits max\_depth=10, min\_samples\_split = 50, min\_samples\_leaf = 50, and min\_impurity\_decrease = 0.01. After testing it on a 5-fold cross-validation test, it resulted in an accuracy score of 84.7%. We then utilized the GridSearchCV function to test an array of parameters. The parameters that resulted from this search were: ‘max\_depth’: 30, ‘min\_impurity\_decrease’: 0, and ‘min\_samples\_leaf’: 30. The 5-fold CV resulted in an accuracy of 92.8%. It also had a high recall of 85.7% and an F1 Score of 85%. This was the best of all the decision tree models we experimented with, and the confusion matrix can be seen in *Appendix 6.*

***Random Forests***

We also developed a Random Forest model, as it provides robustness against overfitting by averaging multiple decision trees, each trained on different parts of the data. This is called bagging or bootstrap aggregation. Similarly to decision trees, random forests can also offer valuable insights into feature importance but are less interpretable since they are made up of many trees. Again we established a baseline Random Forest model with ‘n\_estimators’: 100, which yielded a CV score of 91.2%. To improve our baseline Random Forest model, we also utilized the GridSearchCV function to test for the best parameters. This increased computation cost and time, but successfully returned the best parameters of ‘max\_depth’: 50, ‘min\_impurity\_decrease’: 0, and ‘min\_samples\_leaf’: 30. These values resulted in a 5-fold CV score of 91.3%, which interestingly minimally improved our accuracy.

***Gradient Boosting Classifier***

Lastly, we tried Gradient Boosting, an ensemble method that improves performance iteratively by focusing on errors of prior trees, optimizing performance in our imbalanced datasets where the number of non-defaults significantly exceeds the number of defaults. We experimented with different parameters and resulted in the best parameters of ‘n\_estimators’: 200, ‘learning\_rate’: 0.2, and ‘max\_depth’: 6. These parameters resulted in a CV score of 93.4%, the best of all models we’ve experimented with. Looking into the confusion matrix it results in the following scores, Recall 85.8%, Precision 86.6%, and an F1 score of 86.2% which were also the best of all the models. Gradient boosting allows for better accuracy while still offering us some interpretability through feature importance. The confusion matrix is shown in *Appendix 7*.

**Findings and Implications**

Without using PCA, we found that almost all models performed better and resulted in higher accuracy scores. One model that performed worse than others in general was logistic regression, which may be because logistic regression tends to work better with mostly metric predictor variables like in linear regression, while our dataset contains mostly categorical predictor variables. The models that performed best were Decision Tree with 92.8% accuracy and 84.3% recall on the validation set and Gradient Boosting with 93.4% accuracy and 85.8% recall. We chose to consider recall or sensitivity for our problem since false negatives are more costly. A false negative in this scenario represents a loan that was predicted to be paid in full but was charged off. Our goal is to improve accuracy as well as recall scores, due to our focus on false negatives. Moreover, we also used recall because the class we are predicting, MIS\_Status, is imbalanced (mostly paid in full). We will focus on the Decision Tree as our final model for testing since the decision tree is only 0.6% less accurate and 1.5% less recall than the gradient-boosted tree. When considering the tradeoff, the decision tree provides more interpretability with the prediction paths for business decision purposes than gradient boosting, and the difference between accuracy and recall scores between the two is marginal.

The most important features are almost identical in both top-performing models, as seen in *Appendix 9* and *Appendix 10* The top features in both are Term (loan term in months), SBA\_Appv (SBA’s guaranteed amount of approved loan), DisbursementGross (amount disbursed), GRAppv (gross amount of loan approved by bank), RevLineCr (revolving line of credit = 1, no = 0), and NoEmp (number of business employees). From examining the top splits of the decision tree in *Appendix 11*, we can see that the top few splits are determined by Term, or the loan term length, which is the most important feature highlighted in the feature importance charts in *Appendix 9* and *Appendix 10*. At the root node, the tree splits on ‘Term <= 59.5’ with a Gini impurity of 0.34, dividing the samples into two groups: one with 25,034 and the other with 78,584 samples. It is interesting to note that this Term value of 59.5 is at the first quartile of the distribution of Term values in the dataset, as seen in *Appendix 13.* The following node’s splits are based on further distinctions of the loan term: on the left ‘Term <= 48.5’, and the right ‘Term <= 82.5’, both leading to nodes with lower Gini impurities and a more refined distribution of samples. Again, we note that 82.5 is around the second quartile or median of the distribution of Term values in the dataset from *Appendix 13*. This indicates that ‘Term’ is a critical factor in the early stages of decision-making within the model. As the tree branches out, additional features such as ‘DisbursementGross’ and ‘GrAppv’ are introduced, indicating the model’s transition to focusing on loan disbursement amounts, which aids in finer sample classification.

Next, we tested an example business with differing loan terms and disbursement amounts, since these are the most important features of decision tree and gradient boosting, the best-performing models. Our example business is in the most common small business industry of Retail trade (shown in *Appendix 12*), has 8 business employees, created 2 jobs, had 6 jobs retained, is an existing business, not a franchise, and is in an urban location. We tested our model on 9 different business loans with small, medium, and large loan terms (60 months, 85 months, 100 months) and disbursement amounts ($34000, $80000, $200000) using the quantiles from the data distributions of Term and Disbursement as seen in *Appendix 13* and *Appendix 14.* to see which business loans are more likely to be charged off or paid in full. The results of these tests can be seen in *Appendix 15.* These results show that shorter loan terms like 60 months are more likely to be paid in full. As the loan term gets longer, the probability of this business being able to pay in full decreases. Similarly, smaller disbursement amounts, like $34,000, are more likely to be paid in full, and as the disbursement amount grows, it becomes less clear whether the loan will be able to be paid in full. Examining the tree prediction paths in *Appendix 16,* we see predictions are mostly generated on the right side of the tree since we tested terms larger than 60, the first split of the tree. We found that loans that are significantly small in terms and amount tend to be charged off, so we focused on the full distribution of terms and amounts using the quantiles to identify which values are more likely to result in the loan being paid off.

These tests show that small businesses should take caution when asking for larger loans and longer loan terms, as they may be less likely to pay back the loan. If a loan is charged off, this will have negative implications for the small business. To ensure the business has a higher probability of being able to pay back the loan, they should ask for smaller amounts in shorter loan periods. This will help the business keep track of and manage smaller loans easier. Moreover, longer loans over longer periods will also yield more interest payments for the business, which is also a negative side effect for the business in terms of its cash inflow.

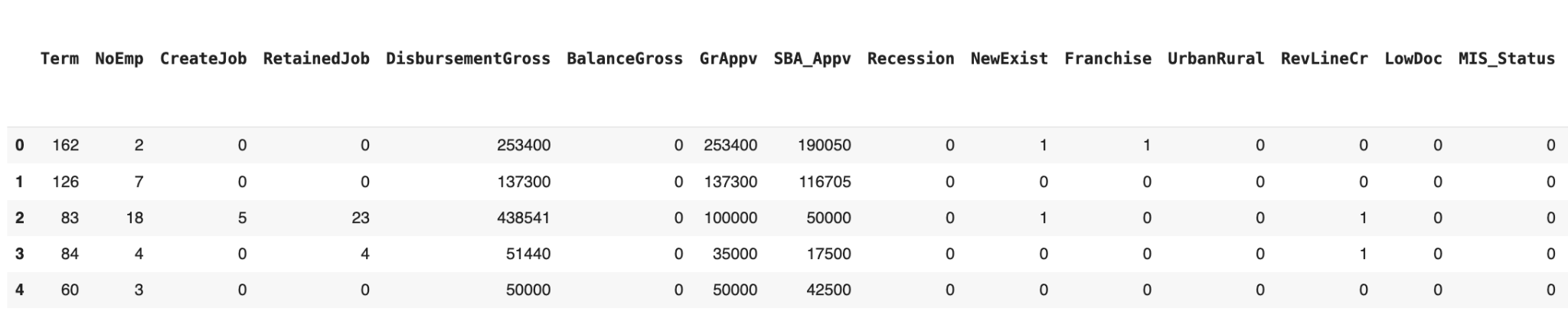
**Conclusion**

In conclusion, we were able to use the loans dataset to create an accurate, interpretable model to predict which kinds of business loans are most likely to be paid in full. We cleaned the dataset by removing uninformative features, creating dummy variables, and dropping missing values. After trying dimension reduction using PCA, we decided to use the original 8 metric variables after the dataset with principal components performed worse with our models. Running the baseline model and comparing it against logistic regression, decision tree, random forest, and gradient boosting, our top-performing models were decision tree and gradient boosting with 92.8% and 93.4% accuracy respectively, and 84.3% and 85.8% recall scores. We looked at recall when considering false negatives and the imbalanced target class. Ultimately we decided on the decision tree model due to the marginal difference in scores. When considering the tradeoff, this model provides more interpretability for business decision-making purposes. In testing our model, we found that smaller amounts and shorter loan terms are more likely to be paid in full. As the loan term increases and the amount distributed increases, it becomes more likely that the loan will be charged off. Therefore our recommendation to small businesses is to consider asking for smaller amounts over smaller loan terms, as the business would be more likely to be able to pay back these types of loans in full and avoid having the loan default.

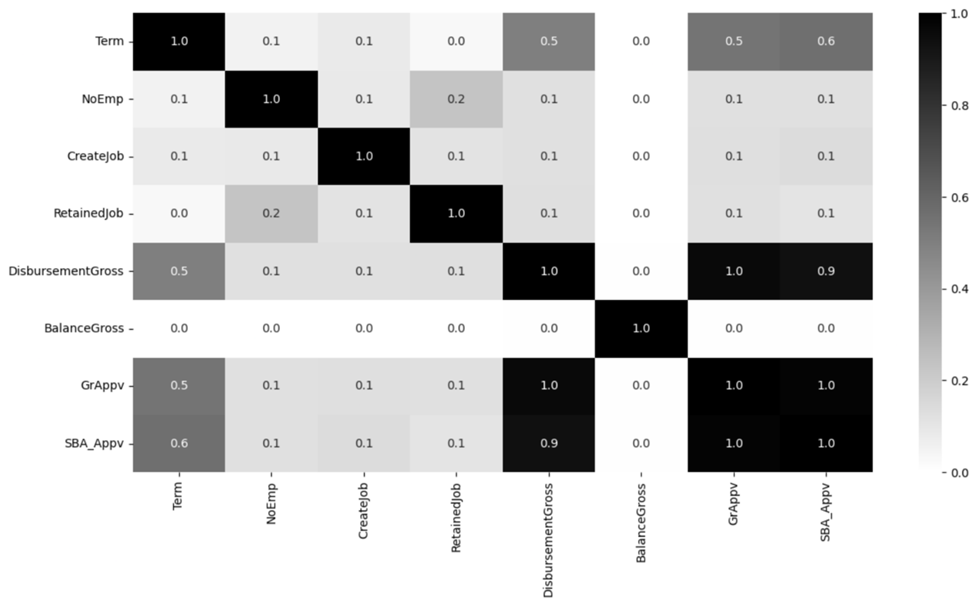
Some ways to improve our models in the future could be to implement decision tree pruning, as our group faced some errors when attempting this. However, we used a grid search to find the optimal parameters for our tree resulting in satisfactory scores. Pruning the tree could offer another way to further limit and reduce the tree to prevent overfitting. Another way to improve our models would be to run a grid search on the gradient-boosted tree. While we ran a grid search on the decision tree and random forest, we were not able to run it on gradient boosting due to limited computer power. With more powerful systems, we would be able to run grid searches on our gradient-boosted tree as well as more powerful models. Another powerful model that could result in higher accuracy and recall scores is neural networks. While improving the recall score further would help businesses to reduce costly false negatives, the model would lose the interpretability of tree paths and feature importances. Our model provides these interpretable aspects and allows businesses to identify how the model makes decisions for a particular small business loan.

**Appendices**

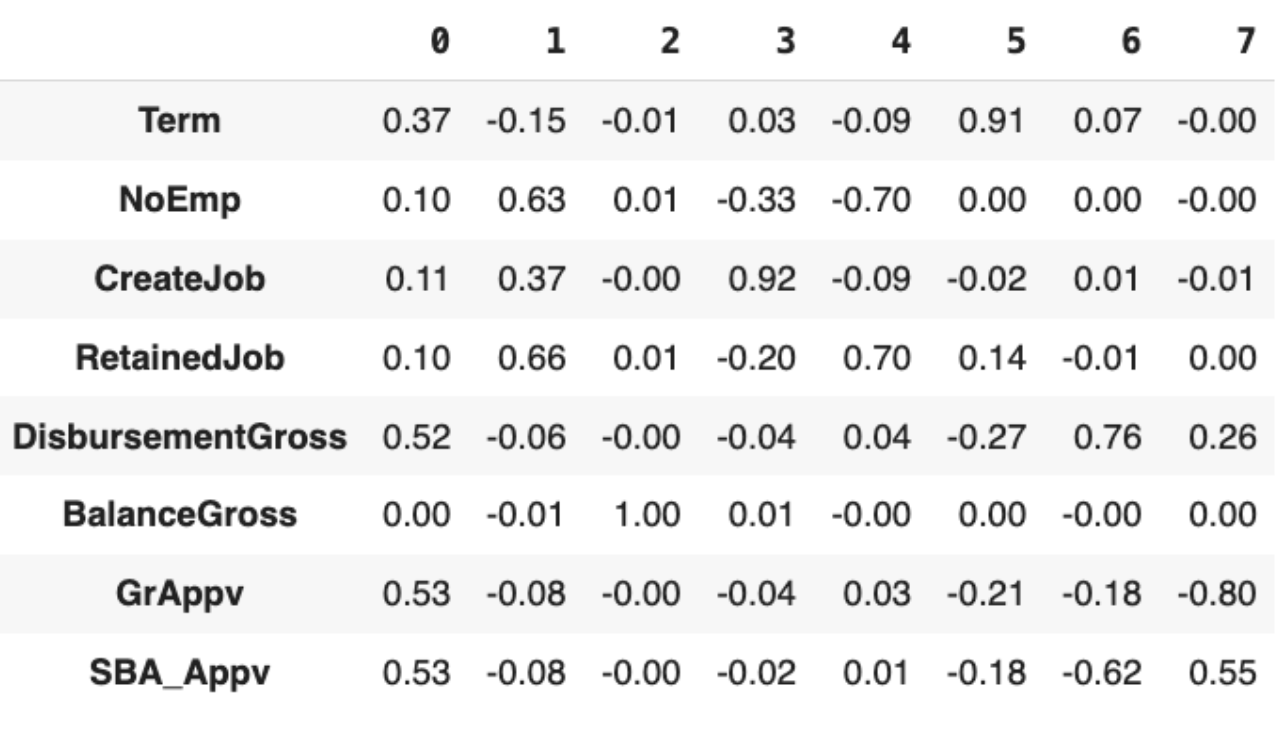
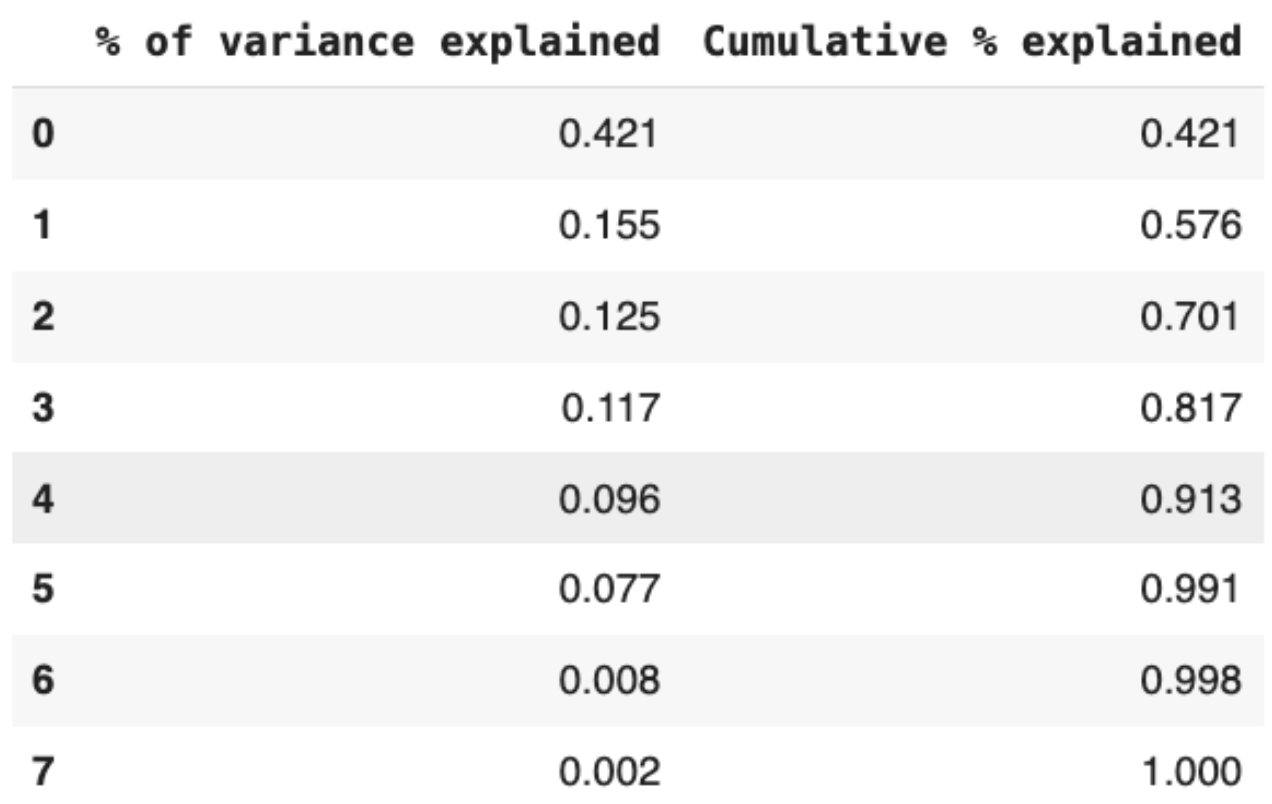
**Appendix 1: Cleaned Dataframe**



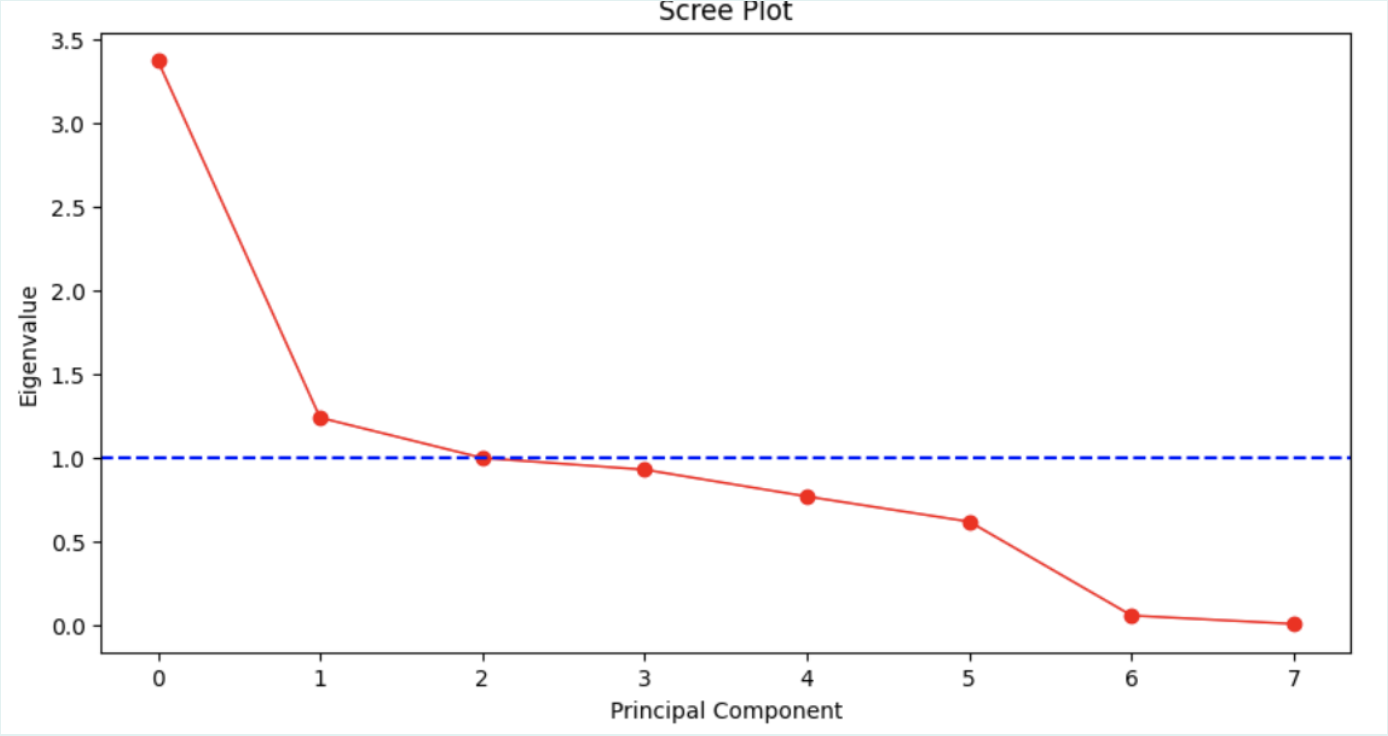
**Appendix 2: Heatmap of Feature Correlations**

****

**Appendix 3: Principal Component Analysis**

****

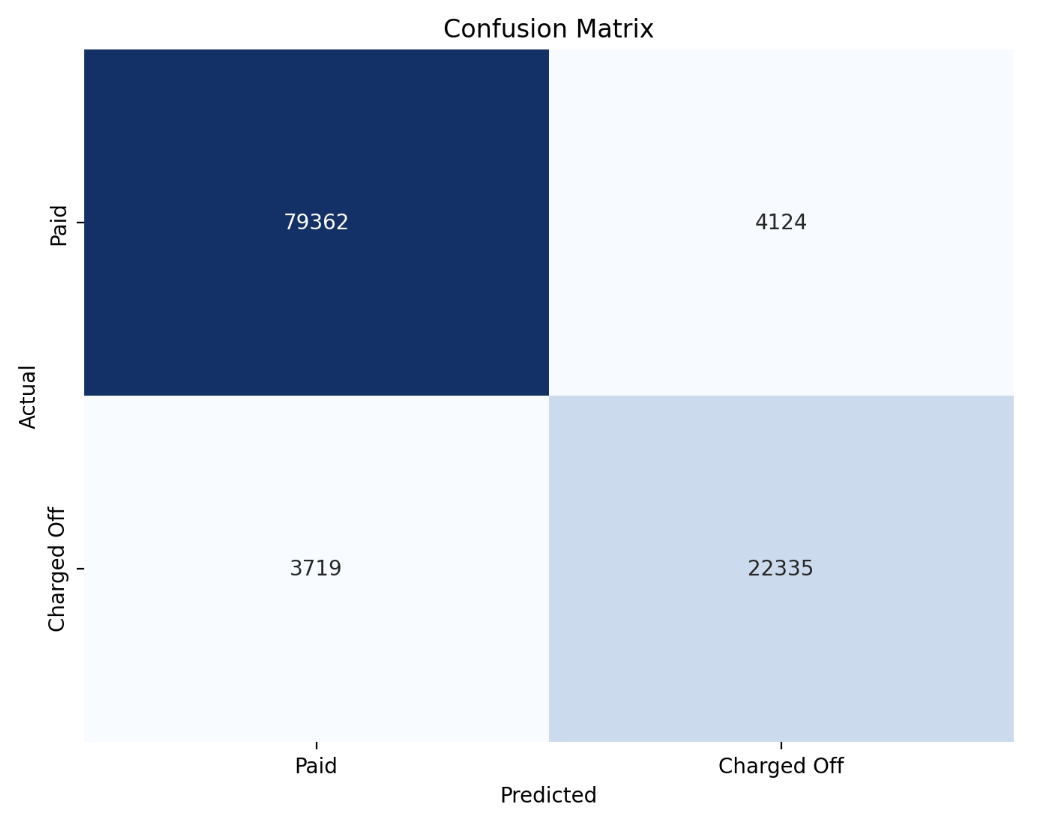
**Appendix 4: Scree Plot**

****

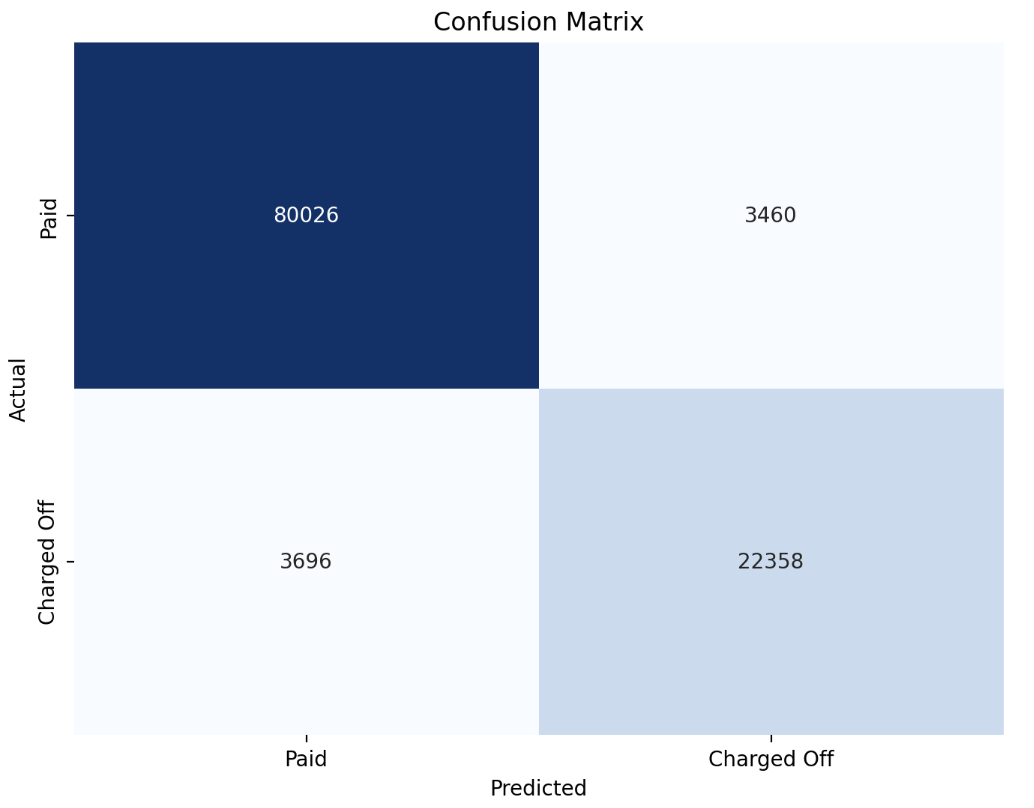
**Appendix 5: Results of Models with PCA vs. Without PCA**

| **Model** | **With PCA** | **Without PCA** |
| --- | --- | --- |
| Logistic Regression | 76.1% | 76.1% |
| Logistic Regression w/ Adjusted Cutoff | 77.2% | 80.5% |
| Full Decision Tree | 74.4% | 89.8% |
| Reduced Tree | 79.1% | 84.7% |
| Decision Tree w/ Grid Search | 80.4%  max\_depth = 20  min\_impurity\_decrease = 0  min\_samples\_leaf = 100 | **92.8%**  max\_depth = 20  min\_impurity\_decrease = 0  min\_samples\_leaf = 30 |
| Random Forest | 77.6% | 91.2% |
| Random Forest w/ Grid Search | 80.7%  max\_depth = 30  min\_impurity\_decrease = 0  min\_samples\_leaf = 10 | 91.3%  max\_depth = 30  min\_impurity\_decrease =  min\_samples\_leaf = |
| Gradient Boosting | 80.8% | **93.4%** |

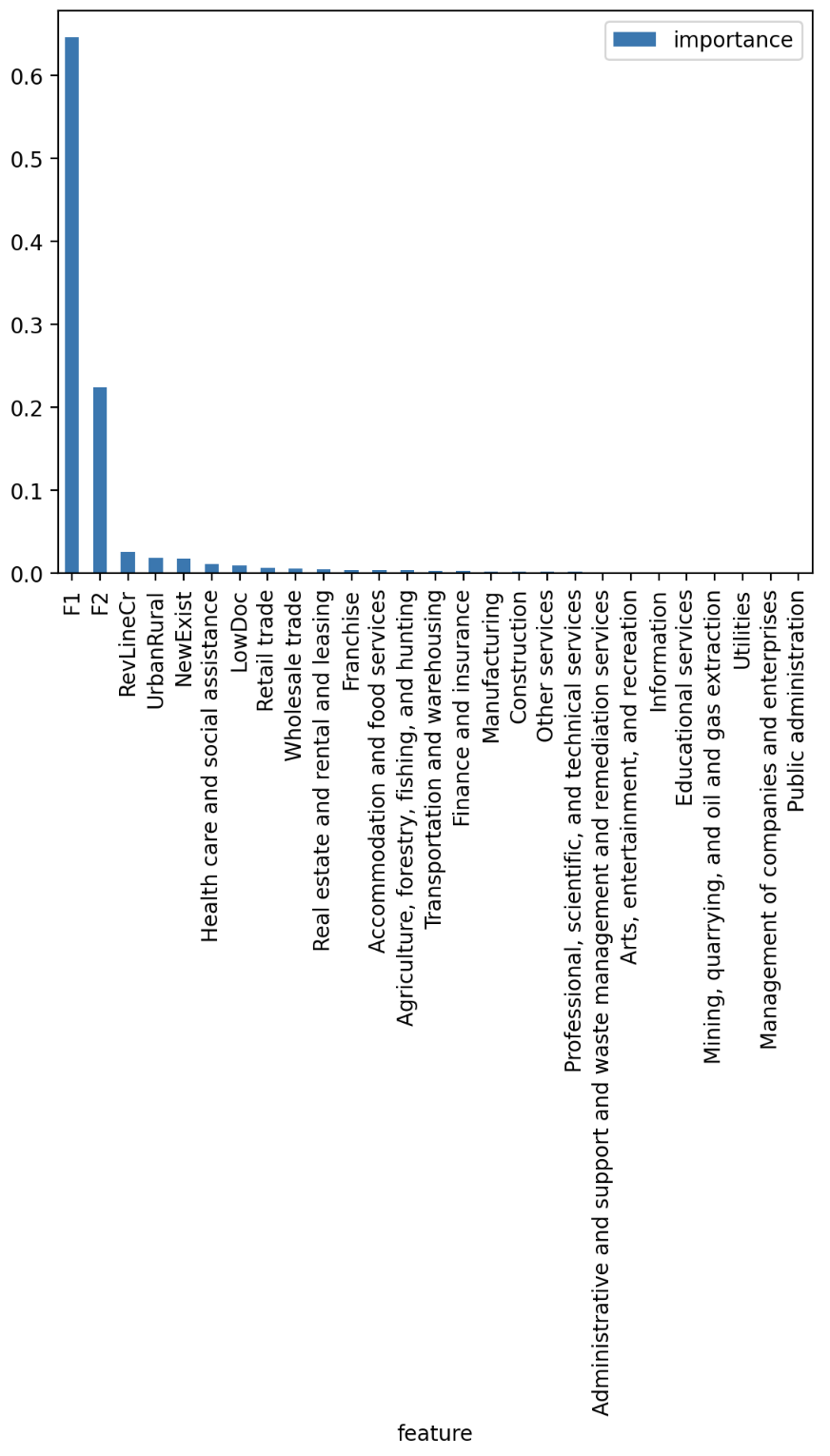
**Appendix 6: Grid Searched Decision Tree Confusion Matrix**

****

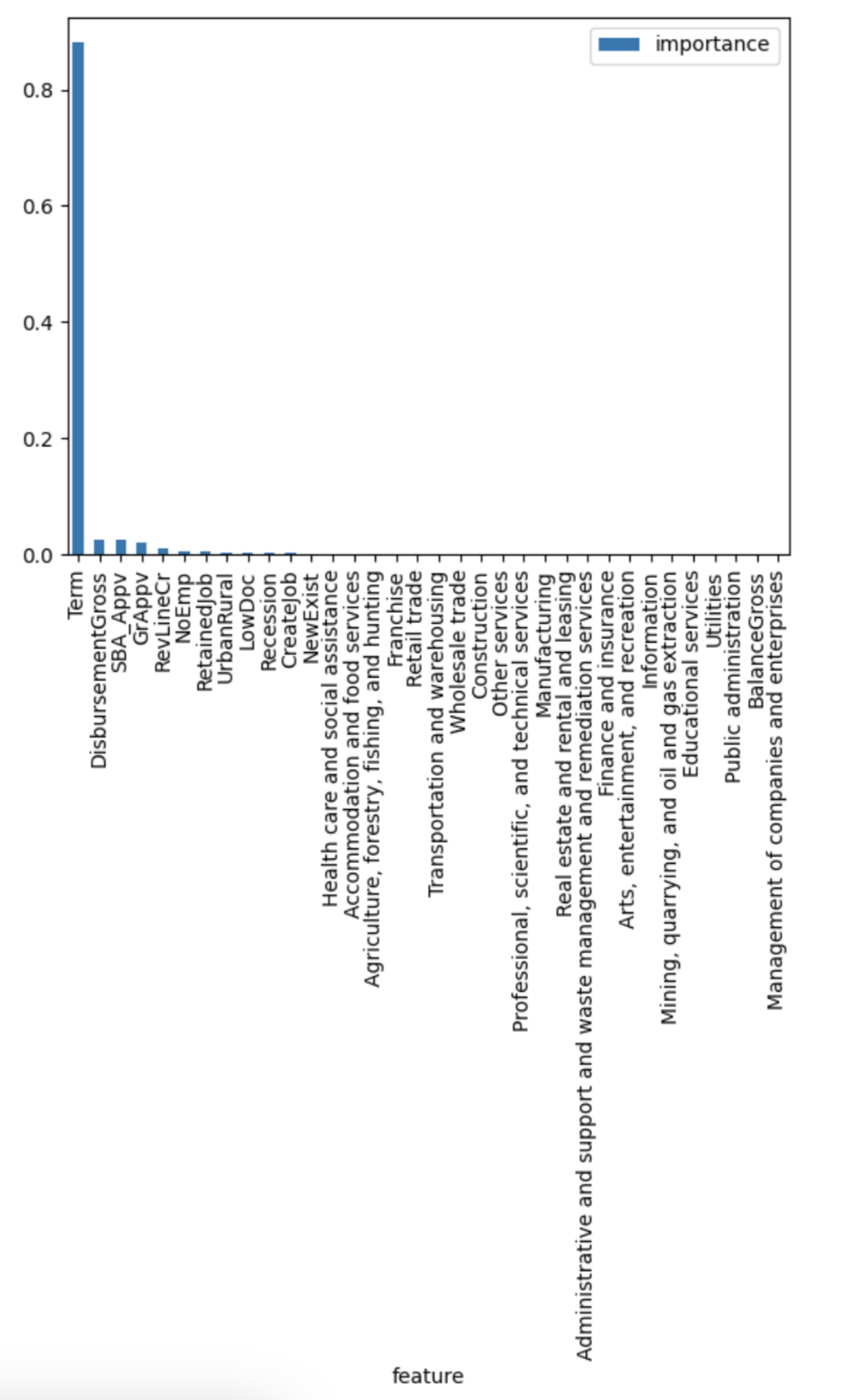
**Appendix 7: Gradient Boosted Tree Confusion Matrix**

****

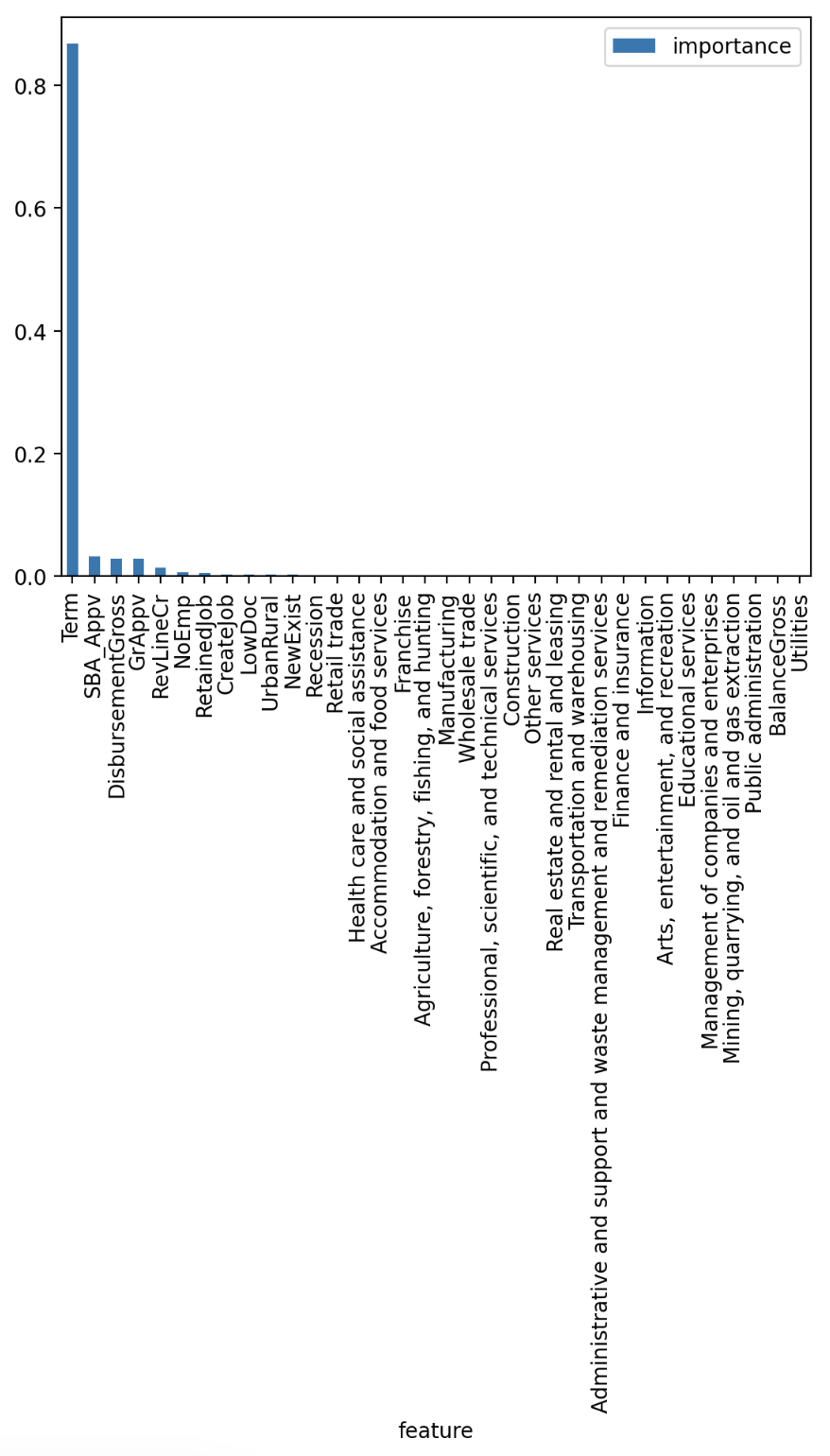
**Appendix 8: *With PCA* - Feature Importances of Gradient Boosting (Best Model)**

****

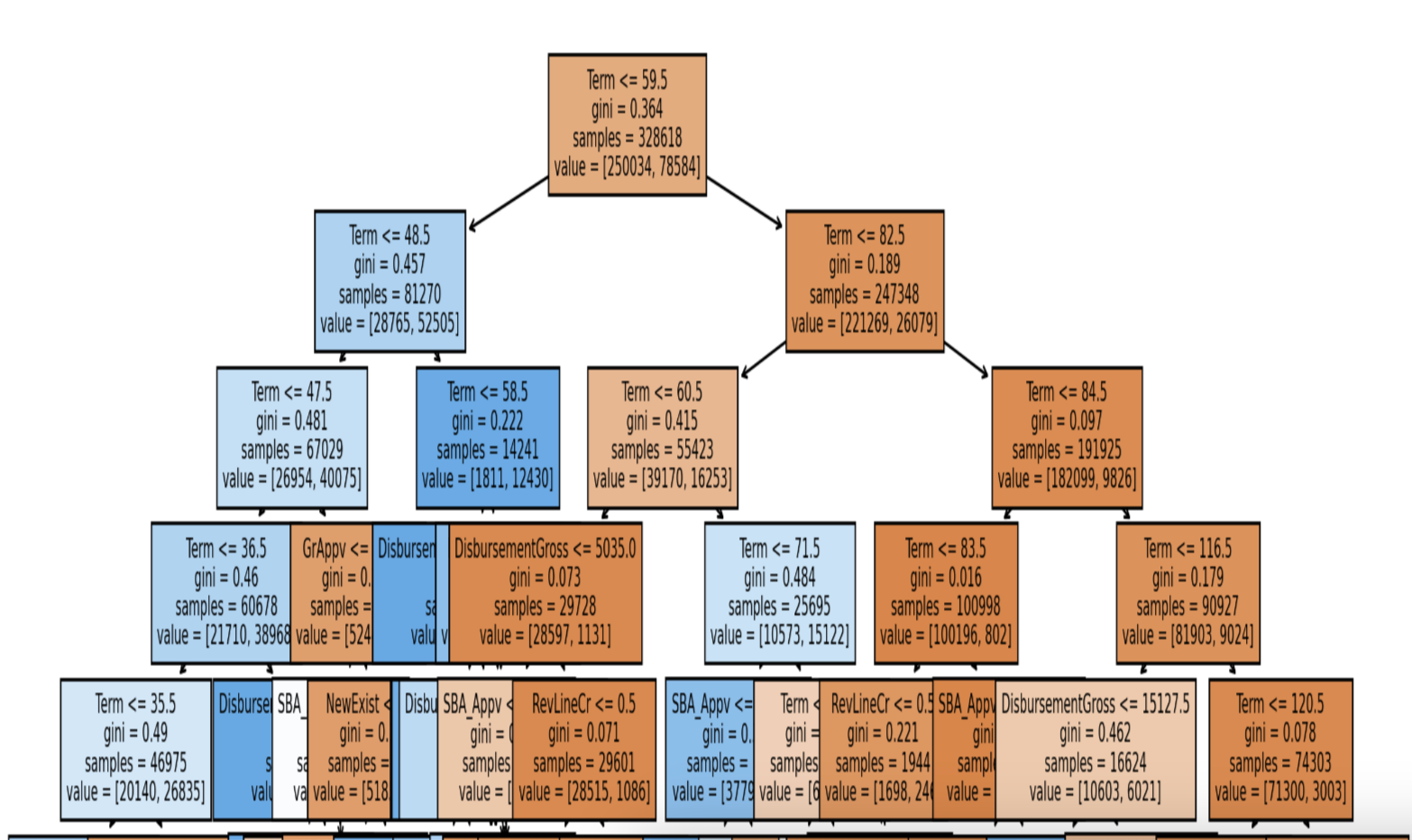
**Appendix 9: *Without PCA -* Feature Importances of Gradient Boosting (Best Model)**

****

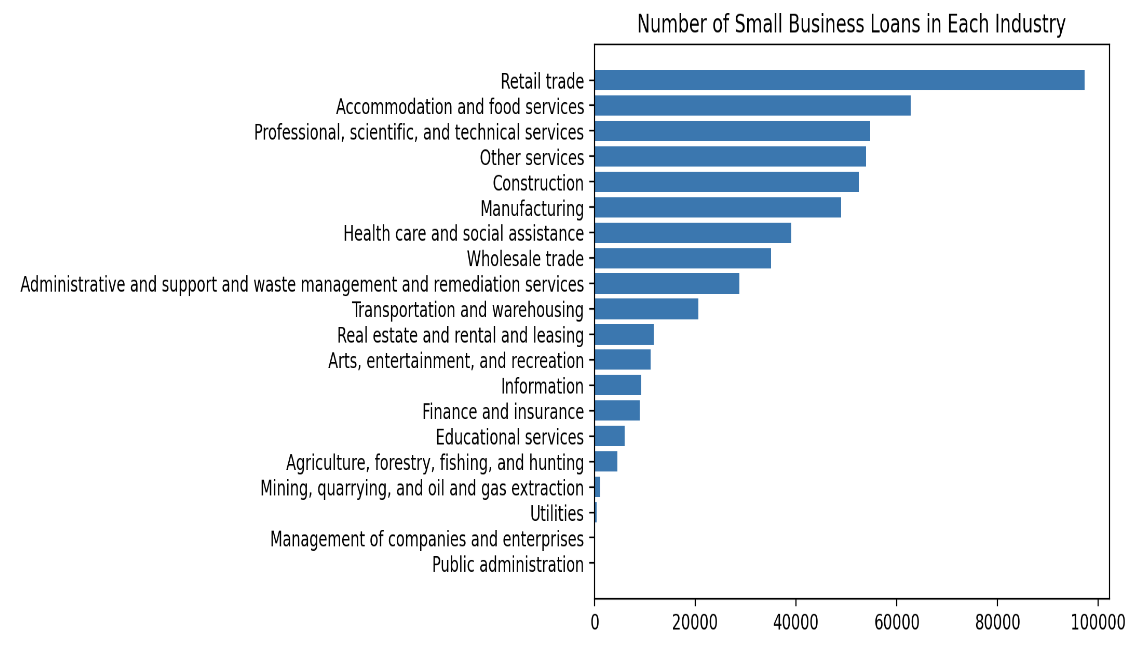
**Appendix 10: *Without PCA -* Feature Importances of Decision Tree (Most Interpretable)**

****

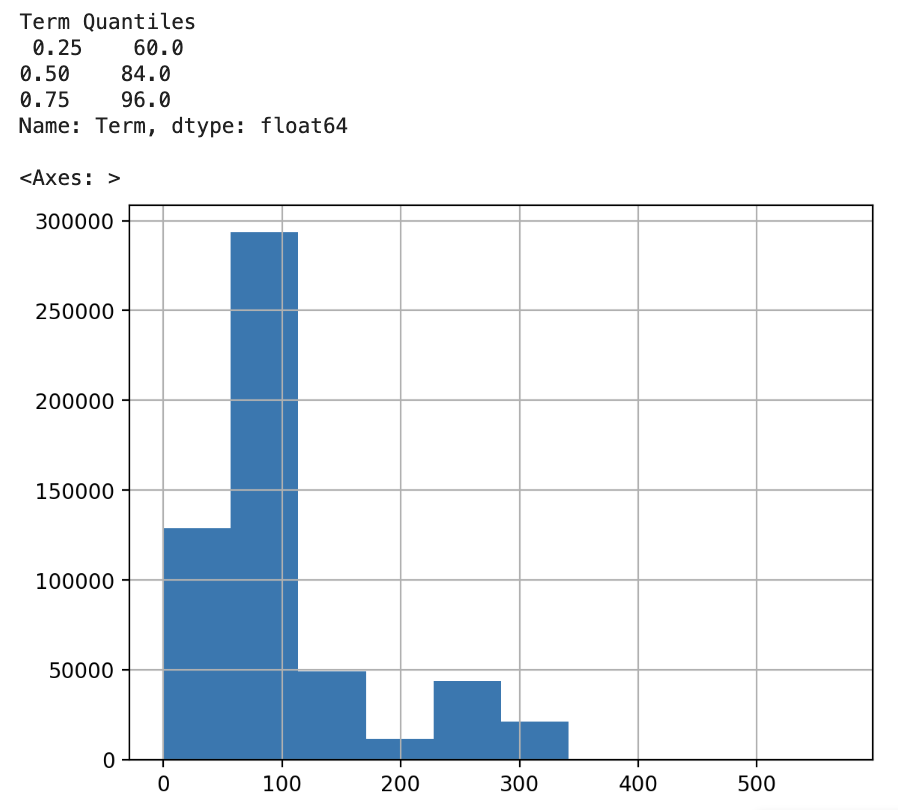
**Appendix 11: Top Splits of Grid Search (Best) Decision Tree**

****

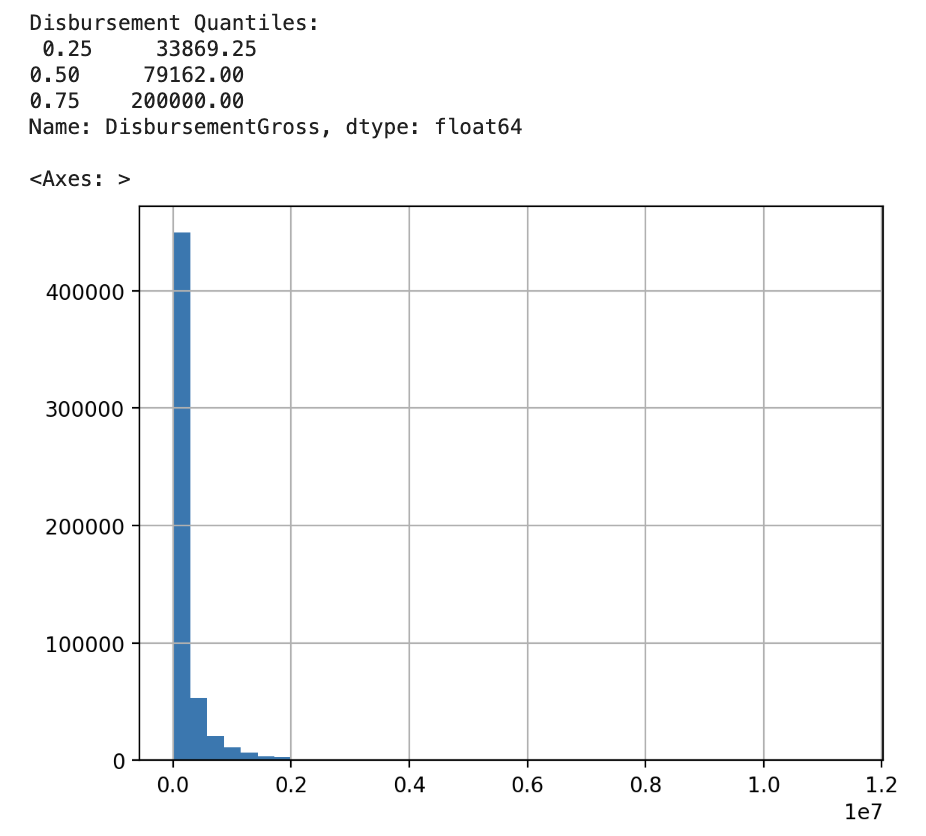
**Appendix 12: Distribution of NAICS Industries**

****

**Appendix 13: Quantiles and Distribution of Term**

****

**Appendix 14: Quantiles and Distribution of Disbursement**

****

**Appendix 15: Results of Testing Ex. Business Loans on Our Decision Tree Model**

| **Loan Term** | **Disbursement** | **MIS\_Status** | **Probability** |
| --- | --- | --- | --- |
| 60 months | $50,000 | **Paid in Full** | **1.00** |
| 60 months | $100,000 | **Paid in Full** | **0.97** |
| 60 months | $250,000 | Paid in Full | 0.81 |
| 85 months | $50,000 | Paid in Full | 0.81 |
| 85 months | $100,000 | Paid in Full | 0.75 |
| 85 months | $250,000 | Charged Off | 0.76 |
| 100 months | $50,000 | Charged Off | 0.53 |
| 100 months | $100,000 | Charged Off | 0.54 |
| 100 months | $250,000 | Charged Off | 0.86 |

**Appendix 16: Tree Prediction Paths of Tested Example Business Loans**

| 60 months, $34,000 | 60 months, $80,000 | 60 months, $200,000 |
| --- | --- | --- |
| 85 months, $34.000 | 85 months, $80,000 | 85 months, $200,000 |
| 100 months, $34,000 | 100 months, $80,000 | 100 months, $200,000 |

**Appendix 17: Data Dictionary**

| **Column** | **Definition** |
| --- | --- |
| Term | Loan term in months |
| NoEmp | Number of business employees |
| CreateJob | Number of jobs created |
| RetainedJob | Number of jobs retained |
| DisbursementGross | Amount disbursed |
| BalanceGross | Gross amount outstanding |
| GrAppv | Gross amount of loan approved by bank |
| SBA\_Appv | SBA’s guaranteed amount of approved loan |
| Recession | 1 = During recession, 0 =Not during recession |
| NewExist | 1 = New business, 0 = Existing business |
| Franchise | 1 = Franchise, 0 = Not a franchise |
| UrbanRural | 1 = Rural, 0 = Urban |
| RevLineCr | 1 = Revolving line of credit, 0 = Not |
| LowDoc | 1 = LowDoc loan program, 0 = Not |
| MIS\_Status | 1 = Charged off, 0 = Paid in full **(Target variable)** |
| Accomodation and food services | 1 = Business in this industry, 0 = Not |
| Administrative and support and waste  management and remediation services | 1 = Business in this industry, 0 = Not |
| Agriculture, forestry, fishing, and hunting | 1 = Business in this industry, 0 = Not |
| Arts, entertainment, and recreation | 1 = Business in this industry, 0 = Not |
| Construction | 1 = Business in this industry, 0 = Not |
| Educational services | 1 = Business in this industry, 0 = Not |
| Finance and insurance | 1 = Business in this industry, 0 = Not |
| Health care and social assistance | 1 = Business in this industry, 0 = Not |
| Information | 1 = Business in this industry, 0 = Not |
| Management of companies and enterprises | 1 = Business in this industry, 0 = Not |
| Manufacturing | 1 = Business in this industry, 0 = Not |
| Mining, quarrying, and oil and gas extraction | 1 = Business in this industry, 0 = Not |
| Other services | 1 = Business in this industry, 0 = Not |
| Professional, scientific, and technical services | 1 = Business in this industry, 0 = Not |
| Public administration | 1 = Business in this industry, 0 = Not |
| Real estate and rental and leasing | 1 = Business in this industry, 0 = Not |
| Retail trade | 1 = Business in this industry, 0 = Not |
| Transportation and warehousing | 1 = Business in this industry, 0 = Not |
| Utilities | 1 = Business in this industry, 0 = Not |
| Wholesale trade | 1 = Business in this industry, 0 = Not |