Modeling for Maximum Profitability for Small Business Lenders Using Misclassification Costs

Daniel Borrero

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Data Science

Department of Mathematical Statistics

Central Connecticut State University

New Britain, Connecticut

Fall 2020

Thesis Advisor:

Dr. Krishna Saha

Department of Mathematical Sciences

Modeling for Maximum Profitability for Small Business Lenders Using Misclassification Costs

Daniel Borrero

An Abstract of a Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Data Science

Department of Mathematical Statistics

Central Connecticut State University

New Britain, Connecticut

Fall 2020

Thesis Advisor:

Dr. Krishna Saha

Department of Mathematical Sciences

ABSTRACT

With the rapid growth of large companies and online retailers, many small businesses are struggling to keep up with the competition and need financial support to keep their doors open. Luckily, the Small Business Administration (SBA) caters exclusively to entrepreneurs and small businesses providing loans to start, grow, or keep their business afloat. Loans of this type were greatly needed by businesses during the mid to late 2000s with the onset of the Great Recession, which is the most severe economic recession in the United States since the Great Depression.

Many businesses take these loans without the means to pay them back, which increases the risk for lenders to approve future loan applications. As hundreds of thousands of SBA loans are applied for every year, lenders do not have the capability to meet with individual businesses to assess whether they are suited to pay back the loans in full. Even if this were possible, lenders would make their decisions solely on the information provided by the borrower.

Using loan data provided by the SBA website, several classification models were developed to predict whether or not a business would default on their loans. The C5.0 model with misclassification costs was determined to be the best performing model based on overall profitability, as well as other evaluation measures. In addition, Term and Bank State were established as the most important variables in predicting loan default.

Table of Contents

INTRODUCTION

In recent years, small businesses have been dying in America largely due to larger companies taking up more of the market over time. In addition, as of 2011 the U.S. is losing more business than it is creating, which has contributed to the increased costs of starting up a business including healthcare and technology costs [1]. This loss in revenue leads businesses to find ways to acquire capital to stay afloat, and many turn to Small Business Administration (SBA) loans. However, loans may not be enough to keep the businesses running, which will result in the loans going into default. For this reason, it would be beneficial for lenders to utilize a model to minimize the rate of default.

Statement of Purpose

The purpose of this analysis is to test several classification models and determine the model that maximizes revenue per applicant. Various classification models including CART, Random Forest, and Neural Networks were run using *Default* as the target variable, and data driven misclassification costs were used to determine the profitability. While SBA loans are backed by the SBA in case of default, they are not backed completely. The SBA backs up to 85% of loans up to $150,000 and up to 75% of loans greater than $150,000, so lenders may still lose money on defaulted loans.[2] Using classification models can help to predict whether businesses will pay back these loans or not.

[1] https://thecapitalist.com/why-small-business-america-dying/
[2] https://www.sba.gov/partners/lenders/7a-loan-program/types-7a-loans#section-header-0

Stated Research Goals

While there have been studies predicting loan default, they mostly focused on personal loans. In addition, they rarely used misclassification costs as a way to evaluate their models and solely relied on performance measures such as accuracy.

In this analysis, SBA loan data was run through numerous classification models to determine which is optimal in predicting loan default. In the process, Exploratory Data Analysis (EDA) was performed to determine which variables had the strongest relationship with the target variable *Default*, and which variables were not important enough to be included in modeling. Once the models were created, they were run on a test dataset, and the metrics from all models were compared, such as accuracy and specificity, along with the average profitability from the misclassification costs. Once the optimal model was identified, we determined which variables were significant in predicting whether or not loans would go into default.

Literature Review

The Small Business Administration was created in 1953 as an independent agency of the federal government to aid, counsel, assist and protect the interests of small business concerns, which helps Americans start, build, and grow businesses.[3] Since its inception, the SBA has approved hundreds of thousands of loans worth billions of dollars per year. However, many businesses are unable to make the payments and the loans eventually go into default. 1 out of 6 SBA 7(a) loans issued from 2006 through 2015 were not paid back, so creating a model to accurately predict default would be very beneficial to lenders as well as to the SBA.[4]

Few studies have been done using SBA loan data and classification models to predict default. Professors at California State University - Sacramento (Li, M., Mickel, A., & Taylor, S. 2018) developed an assignment for students using SBA loan data that would follow the American Statistical Association's Guidelines for Assessment and Instruction in Statistics Education (GAISE). The case study was designed to teach statistical thinking by focusing on how to use real data to make informed decisions for a particular purpose. For this assignment, students assumed the role of a loan officer who is deciding whether to approve a loan to a small business. The original case study only used Logistic Regression as a classification method, however an adapted version was used for graduate students. The adapted version used Logistic Regression, Neural Networks, and Support Vector Machines (SVM) on a small subset of the data (*State=CA, NAICS= Real Estate Rental and Leasing*). After evaluating, the Neural Network model had the lowest

[3] https://www.sba.gov/about-sba/organization
[4] https://www.nerdwallet.com/blog/small-business/sba-loan-default-know-cant-pay/

misclassification rate and the SVM model had the highest misclassification rate. Nigro, P., & Glennon, D. (2005) performed a study using a discrete-time hazard approach to predict default. SBA 7(a) loans disbursed between 1983-1998 were randomly selected from one of three groups based on loan maturity: short-term (3 years), medium-term (7 years), and long-term (15 years). Results showed that maturity specific models more accurately predict the number of defaults over time relative to the pooled-sample models. The researchers also determined that industry classification is an important determinant of default, and that the success and failure of a small loan is closely tied to both the regional and industrial economic conditions in which the borrower operates. For example, they found long-term loans in agriculture, forestry and fishing industries showed much lower default rates compared to manufacturing and retail trade industries.

While studies using SBA loan data are lacking, there are many using other types of loans, such as peer-to-peer loans. For one recent study the authors (Tariq Aziz, H. I., Sohail, A., Aslam, U., & Batcha, N. K. 2019) applied data mining techniques for predicting and classifying loan default. They used the Sample, Explore, Modify, Model, and Assess methodology (SEMMA). This methodology differs from CRISP-DM as SEMMA requires an understanding of the pre-requisites of the business understanding and databases, which was an advantage for this study. Three models were used to predict default: A Decision Tree algorithm (DT), Logistic Regression (LR) and Neural Networks (NN), and their performances were evaluated on various parameters. The NN model had the highest accuracy, however, sensitivity and specificity were considered most crucial in determining the optimal model. The LR model was considered optimal as it had the

highest sensitivity and lowest specificity. Alomari, Z. (2017) from Concordia University performed a study for default prediction of Peer-to-Peer loans and for learning interesting associations between various attributes of the same loan applications. The first goal was to find a classification model that would be as accurate as possible at predicting whether a peer-to-peer loan would be paid off or default, and the second goal was to find interesting relations and associations among variables that could be valuable to potential investors. Loan data from Lending Club between 2012-2013 was used to create seven classification models, including Random Forest and Artificial Neural Networks. The most effective classification model was determined based on highest accuracy, which was achieved using Random Forest with an accuracy of 71.75%. Another study performed by Bhargava (Sarma, K. S. 2017) utilized Lending Club data to predict default. The variables used for modeling were determined using 1-R Square based on the correlation and significance. Four models were compared: Decision Tree, Logistic Regression, Random Forest and Neural Network. They were evaluated by minimum misclassification rate, and the Random Forest model performed best out of the four models. These previously mentioned studies decided on an optimal model based on maximum accuracy or minimum misclassification rate. Misclassification of false negatives can be more fatal for lending companies and financial institutes as it results in approving a loan that likely cannot be paid back in full and is a more useful way to evaluate models (Amit, R., & Zott, C. 2015). This analysis went a step further by using misclassification costs to determine the optimal model based on profit as opposed to certain measures.

Statement of Need

Of the studies performed on predicting loan default, most focused on individuals and personal loans as opposed to companies and business loans. While they may seem similar, they are fairly different, and datasets for each contain different variables, and many are not applicable to business loans.

In addition to the lack of studies on business loan default, most studies evaluated their models based on a single metric or combination of metrics, such as accuracy, sensitivity, and specificity. The purpose of creating models for these studies is to see which can best predict default, assuming the best model would be most profitable. While these metrics can determine which model is most profitable, the metrics used to determine the optimal model vary from study to study. For this reason, data driven misclassification costs were used, which calculated the average profit per loan.

This may also provide insight into business loan default during a period of economic hardship. The data used for this analysis spanned from the beginning of 2005 to the end of 2009. The Great Recession was a global economic downturn that devastated world financial markets as well as banking and real estate industries from December 2007 to June 2009.[5] During this time small business loans drastically declined, prior to large increase from 2005 to 2007, and businesses struggled greatly to repay them.[6]

[5] https://www.history.com/topics/21st-century/recession
[6] https://files.consumerfinance.gov/f/documents/cfpb_data-point_small-business-lending-great-recession.pdf

EXPERIMENTAL PROCEDURES

This analysis was performed following the Data Science Methodology (DSM), as proposed in *Data Science Using Python and R* (Larose, C. D., & Larose, D. T. 2019). DSM is an adaptation of the Cross Industry Standard Practice for Data Mining (CRISP-DM), which is the most widely used analytics process standard.[7] It is a methodology that helps the analyst keep track of which phase is being performed (Larose, D. T. 2018). The methodology consists of seven phases:

1. Problem Understanding Phase
2. Data Preparation Phase
3. Exploratory Data Analysis (EDA) Phase
4. Setup Phase
5. Modeling Phase
6. Evaluation Phase
7. Deployment Phase

Each phase was an integral part of the analysis and is explained in detail as the phase was performed.

[7] https://www.forbes.com/sites/metabrown/2015/07/29/what-it-needs-to-know-about-the-data-mining-process/#55190c21515f

Phase 1: Problem Solving

The problem-solving phase is essential and can be considered the most important phase of an analysis; it provides the framework of the questions that want to be answered, along with how the analysis will provide the necessary information to provide answers to these questions. It consists of two stages:

1. Clearly Enunciate the Project Objectives
2. Translate these objectives into a Data Science Problem

The first objective was to learn about the loans that go into default and those that do not, and to learn about the factors that may be influential. This was done in the EDA phase using bar charts, plots, heat maps, etc. to identify and possible relationships between the variables and the target variable, *Default*.

The second objective was to develop a method that will identify likely positive responses (*Default*=1). Doing so would allow lenders to identify potential defaults and not approve loans, which would save lenders from losing money. This was done using several classification models to predict *Default*. Misclassification costs were the main way to compare the models, as they produced a matrix that was used to calculate the average amount of profit generated or lost for the lender, and the optimal model was the model with the highest profitability on average per loan. Other measures such as accuracy and sensitivity were also compared to evaluate the models.

Phase 2: Data Preparation Phase

The dataset used for this analysis was the National SBA dataset. It contained data for small business loans that were issued between 1987-2014. In total there were 899,164 records and 27 variables. The variables along with their descriptions can be seen in Table 1 (Min, L., Mickel, A., & Taylor, S. 2018).

Variable	Data Type	Description of variable
LoanNr_ChkDgt	Text	Identifier - Primary key
Name	Text	Borrower name
City	Text	Borrower city
State	Text	Borrower state (Including Washington D.C.)
Zip	Text	Borrower zip code
Bank	Text	Bank name
BankState	Text	Bank state (Including Washington D.C.)
NAICS	Text	North American Industry Classification System code
ApprovalDate	Date/Time	Date SBA commitment issued
ApprovalFY	Text	Fiscal year of commitment
Term	Number	Loan term in months
NoEmp	Number	Number of business employees
NewExist	Text	1 = Existing business, 2 = New business
CreateJob	Number	Number of jobs created
RetainedJob	Number	Number of jobs retained
FranchiseCode	Text	Franchise code, (00000 or 00001) = No franchise
UrbanRural	Text	1 = Urban, 2 = Rural, 0=undefined
RevLineCr	Text	Revolving line of credit: Y = Yes, N = No
LowDoc	Text	LowDoc Loan Program: Y = Yes, N = No
ChgOffDate	Date/Time	The date when a loan is declared to be in default
DisbursementDate	Date/Time	Disbursement date
DisbursementGross	Currency	Amount disbursed
BalanceGross	Currency	Gross amount outstanding
MIS_Status	Text	Loan status charged off = CHGOFF, Paid in full = PIF
ChgOffPrinGr	Currency	Charged-off amount
GrAppv	Currency	Gross amount of loan approved by bank
SBA_Approval	Currency	SBA's guaranteed amount of approved loan

Table 1: Description of the Original 27 Variables

For this analysis, only loans approved between 2005 and 2009 were used. Since the type of SBA loan was not provided, all loans were assumed to be SBA 7(a) loans, which are the most common type of SBA loan. Several variables were reclassified, and new variables were created to be more useful for the analysis.

MIS_Status

The *MIS_Status* variable represents whether a loan is defaulted on or not and is the target variable for this experiment. Either the loan was paid in full, *PIF*, or was charged off, *CHGOFF*. A charge-off is a debt that a creditor has given up trying to collect on after a business missed payments for several months.[8] This was treated as a default, and the variable was modified. MIS_Status was changed to *Default*. Loans paid in full were changed to 0 (No default) and those charged off were changed to 1 (Default).

UrbanRural

The *UrbanRural* variable represents whether the company receiving the loan was located in an urban area (*UrbanRural=1*) or a rural area (*UrbanRural=2)*. Records where this area was not determined have a value of 0. To simplify, a Reclassify node was used to change values of 1 (Urban) to *U* and values of 2 (Rural) to *R*. The undefined records were dealt with in the Data Cleaning section.

[8] https://www.creditkarma.com/advice/i/what-is-a-charge-off/

NewExist

The *NewExist* variable represents whether the company is an existing (*NewExist=1*) or a new business (*NewExist=2*). This variable was reclassified similar to the *UrbanRural* variable. A Reclassify node was used to change values of 1 (Existing Business) to *E* and values of 2 (New Business) to *N*.

New Variables for Misclassification Costs

While the data contained 27 variables, some could not be used as-is to generate the amounts to populate the cost matrix, which would be used to rebalance the data and generate misclassification costs.

The first amount needed was the average loss taken by the lender on a loan that went into default, which is affected by several factors. *ChgOffPrinGr* is the amount that was charged-off, or not paid do the lender. This amount was assumed to only account for principal payments and excludes any interest. For loans in default, subtracting this value from the original loan amount *DisbursementGross* resulted in the amount of the loan that has been paid to the lender before defaulting. This new amount was denoted as *Amount_Paid.* However, these loans were backed by the SBA. In the event a loan is defaulted on, a portion of the loan would be paid back to the lender by the SBA. This amount is represented as *SBA_Approval*. These variables were used to calculate the total amount lost for the lender, *Amount_Lost*. It was calculated by taking the disbursed amount of the loan, *DisbursementGross*, and subtracting both the amount paid on the loan, *Amount_Paid*, and the portion guaranteed by the SBA, *SBA_Approval*. For many

loans in default, the amount paid plus the portion backed by the SBA was more than the original loan amount. For these cases *Amount_Lost* was changed to 0, as it is assumed defaulted loans are not profitable for the lender. This calculation was only needed for records where *Default=1*. For loans paid off (*Default=0*), this value was assumed to be 0.

The second amount needed was the average profit made by the lender. This profit is the amount of interest collected on a loan paid in full. Since no interest rates were included in the dataset, rates for 2005-2009 were calculated using the historical prime rates along with the calculations used by the SBA to calculate loan interest. The interest rate consists of a percentage given by the SBA, along with a prime rate. The prime rate is among the most widely used benchmark in setting home equity lines of credit and credit card rates and is in turn based on the federal funds rate set by the Federal Reserve.[9] It is an important index used by banks to set rates on consumer loan products, including SBA loans.[10] Prime rates from late 2004 to 2015 are listed in Table 2. In December 2008 the prime rate dropped to 3.25% despite the country's economic downturn, and this rate kept until late 2015.[11] These rates are used by all banks and the variable is denoted *Prime_Rate*.

[9] https://www.bankrate.com/rates/interest-rates/prime-rate.aspx
[10] https://www.bankrate.com/rates/interest-rates/wall-street-prime-rate.aspx
[11] https://homeguides.sfgate.com/prime-rate-change-history-3222.html

Historical U.S. Prime Rates	
Effective Date	**Rate**
12/17/2015	3.50%
12/16/2008	3.25%
10/29/2008	4.00%
10/8/2008	4.50%
4/30/2008	5.00%
3/18/2008	5.25%
1/30/2008	6.00%
1/22/2008	6.50%
12/11/2007	7.25%
10/31/2007	7.50%
9/18/2007	7.75%
6/29/2006	8.25%
5/10/2006	8.00%
3/28/2006	7.75%
1/31/2006	7.50%
12/13/2005	7.25%
11/1/2005	7.00%
9/20/2005	6.75%
8/9/2005	6.50%
6/30/2005	6.25%
5/3/2005	6.00%
3/22/2005	5.75%
2/2/2005	5.50%
12/14/2004	5.00%

Table 2: Historical U.S. Prime Rates[12]

The other portion of the interest calculation is the rate set by the SBA. For loans with a term under 84 months, this rate is as low as 2.25% and as high as 4.25% and decreases as the loan amount increases. For loans of 84 months or more, this rate is as low as 2.75% and as high as 4.75%. The amounts and rates can be seen in Table 3. The SBA portion of the rate is denoted *SBA_Interest*. The sum of these two rates is the total interest rate, denoted *Max_Interest*.

[12] https://www.jpmorganchase.com/corporate/About-JPMC/historical-prime-rate.htm

Maximum Interest Rates	
Loans < 84 Months	
Loan Amount	***Rates***
$0-$25,000	*Prime Rate + 4.25%*
$25,001-$50,000	*Prime Rate + 3.25%*
Over $50,000	*Prime Rate + 2.25%*
Loans ≥ 84 Months	
Loan Amount	***Rates***
$0-$25,000	*Prime Rate + 4.75%*
$25,001-$50,000	*Prime Rate + 3.75%*
Over $50,000	*Prime Rate + 2.75%*

Table 3: Maximum Interest Rates[13]

The amount of interest paid on a loan, *Interest_Paid*, was calculated in Excel using the *CUMIPMT* function, which returns the cumulative interest paid on a loan between a starting period and an ending period.[14] The arguments are as follows:

- Rate: Interest Rate, *Max_Interest*
- Nper: The total number of payment periods, *Term*
- Pv: The present value, *DisbursementGross*
- Start_period: The first period in the calculation. Payment periods are numbered beginning with 1
- End_period: The last period in the calculation (Equal to *n_pay*)
- Type: The timing of the payment
    - 0: Payment at the end of the period
    - 1: Payment at the beginning of the period

[13] https://www.sba.gov/sites/default/files/articles/Loan_Chart_Jan_2018_Version_A.pdf
[14] https://support.microsoft.com/en-ie/office/cumipmt-function-61067bb0-9016-427d-b95b-1a752af0e606

The variable *n_pay* was calculated by using the *PMT* function in Excel to find the monthly payment, *monthly_payment*. The monthly payment was divided into *Amount_Paid* to calculate the estimated number of payments, *n_pay*. For loans not in default, *n_pay* is equal to *Term*.

Payments were assumed to be made at the end of a period, which is consistent with several SBA amortization calculators. Calculations were made assuming early/larger payments were not made, which would decrease the amount of interest. The following is an example interest calculation using a loan taken by Optimal Family Dental LLC in January 2019. In the formula the interest rate of 5.50% is divided by 12, as the interest is accrued monthly.

Interest Rate Calculator

Rate	5.50%	
		Amount of Interest
Nper	60	
Pv	$ 75,000.00	$ (10,955.23)
Start_period	1	
End_period	60	
Type	0	

In total, Optimal Family Dental LLC paid $85,955.23 to pay of their loan. This included $10,955.23 of interest which is profit for the lender, The Huntington National Bank.

Data Cleaning

The Data Audit node was used to determine the quality of our data. The Quality tab displays important information regarding both the fields and records. The data was 66.67% complete based on the fields and 35.19% complete based on the records as shown in Figure 1. The fields that were not 100% complete were focused on.

Complete fields (%): 66.67% Complete records (%): 35.18%

Field	Measurement	Outliers	Extremes	Action	Impute Missi...	Method	% Complete	Valid Records	Null Value	Empty String	White Space	Blank Value
ChgOffDate	Continuous	0	0	None	Never	Fixed	35.663	96239	173621	0	0	173621
LowDoc	Flag	--	--	--	Never	Fixed	99.401	268244	0	1616	1616	1616
Default	Flag	--	--	--	Never	Fixed	99.593	268762	0	1098	1098	1098
UrbanRural	Flag	--	--	--	Never	Fixed	99.618	268830	0	0	0	1030
Disburse...	Continuous	0	0	None	Never	Fixed	99.77	269238	622	0	0	622
NewExist	Flag	--	--	--	Never	Fixed	99.981	269809	1	0	0	50
Bank	Categorical	--	--	--	Never	Fixed	99.996	269849	0	11	11	11
BankState	Nominal	--	--	--	Never	Fixed	99.996	269849	0	11	11	11
RevLineCr	Flag	--	--	--	Never	Fixed	99.997	269852	0	8	8	8

Figure 1: Output of incomplete fields from Data Audit node

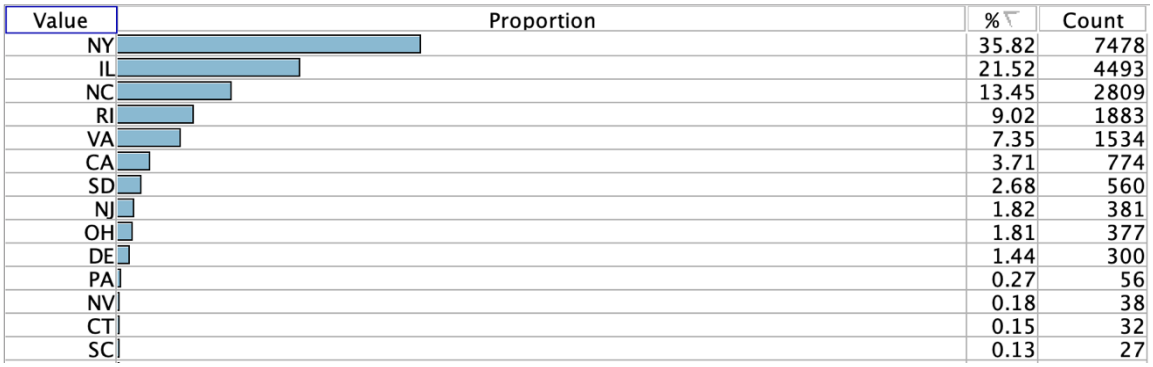
The least complete variable was *ChgOffDate*, which was below 36% complete. While this may seem like a variable lacking in information, it was fairly accurate. This variable lists the dates that loans were defaulted on, however a majority of the loans were not defaulted on, so the majority of loans would not have a charge off date. The percentage of loans in default can later be seen in the EDA section, however it is similar to the percentage of records having charge off dates. Since the analysis is only focused on whether loans went into default or not, this variable was discarded.

Next were the flag variables: *LowDoc, Default*, *NewExist, UrbanRural*, *RevLineCr,* and *UrbanRural*. One effective way of imputing values is to estimate them based on the other variables. This was done in Modeler through the Data Audit node using CART models. This method was used for *Low Doc*, *Default*, and *UrbanRural* as each had over 1,000 missing values. For *NewExist* and *RevLineCr* the blanks and null values were replaced with random values (N or E for *NewExist* and Y or N for *RevLineCr*) as these

fields had a small number of blank and null values. Running these methods created a Supernode that contained the newly imputed values that were predicted from the CART models.

The *DisbursementDate* variable is the date loans was disbursed and had 622 missing values. This variable is different from *ApprovalDate,* which is the date that the SBA backs the loan. However, most of the loans were disbursed the same month they were approved, so the *DisbursementDate* variable was omitted.

The *Bank* and *BankState* variables had missing values for the same 11 records. *BankState* was omitted as there were over 3000 banks and they were not used for modeling. While it may make sense that the company gets a loan from a bank in the same state this is not the case, as more companies than not received their SBA loans from banks in different states. The 11 records all list NY as their *State*, so *BankState* was viewed for all records where the *State* is New York.



Value	Proportion	%	Count
NY		35.82	7478
IL		21.52	4493
NC		13.45	2809
RI		9.02	1883
VA		7.35	1534
CA		3.71	774
SD		2.68	560
NJ		1.82	381
OH		1.81	377
DE		1.44	300
PA		0.27	56
NV		0.18	38
CT		0.15	32
SC		0.13	27

Figure 2: Distribution of State where BankState = "NY"

Over 36% of the loans came from within the state and is the highest percentage of all states, so New York (NY) was used as the *BankState* for these 11 records.

The next step was to identify the variables that could be omitted, the first being *LoanNr_ChkDgt*. This is a 10-digit identification number given by the SBA as is unique to each loan, so it was not useful for modeling.[15] The company name, *Name*, was also omitted as most of these companies only received one loan.

The variables *State*, *City*, and *Zip* are all related. *Zip* is a subset of *City*, as one city can have multiple zip codes. For this reason, *Zip* was omitted. *City* is a subset of *State*, and while it may be useful on a smaller scale, there were too many cities in the data, so this variable was also omitted.

The approval date by the SBA, *ApprovalDate*, is similar to the fiscal year, *ApprovalFY*. The federal government runs on a fiscal year of October 1st to September 30th, so a loan approved in October 2005 will show 2006 for the fiscal year.[16] While the fiscal year may be useful, a majority of the records had the same approval year and fiscal year. The *ApprovalDate* is also more specific, so it can be viewed by month, quarter, or year. For this reason, *ApprovalDate* was used for modeling and *ApprovalFY* was omitted.

It is important to note that this step only focused on fields with missing values. It did not focus on outliers or removing data entry errors. This is addressed in the EDA phase.

[15] https://www.sba.gov/node/6364#:~:text=Column%20S%20(SBA%20GP%20Number,payment%20information%20cannot%20be%20processed

[16] https://www.investopedia.com/terms/f/fiscalyear.asp

Phase 3: Exploratory Data Analysis

The Exploratory Data Analysis (EDA) phase was is necessary to obtain a basic understanding using graphical analysis and descriptive statistics (Larose, D. T. 2018). Doing so provides insight to possible relationships between variables, as well as between variables and the target variable, *Default* using tables, bar plots, heat maps, and other methods. Beginning with *Default*, the distribution is shown in Figure 3.

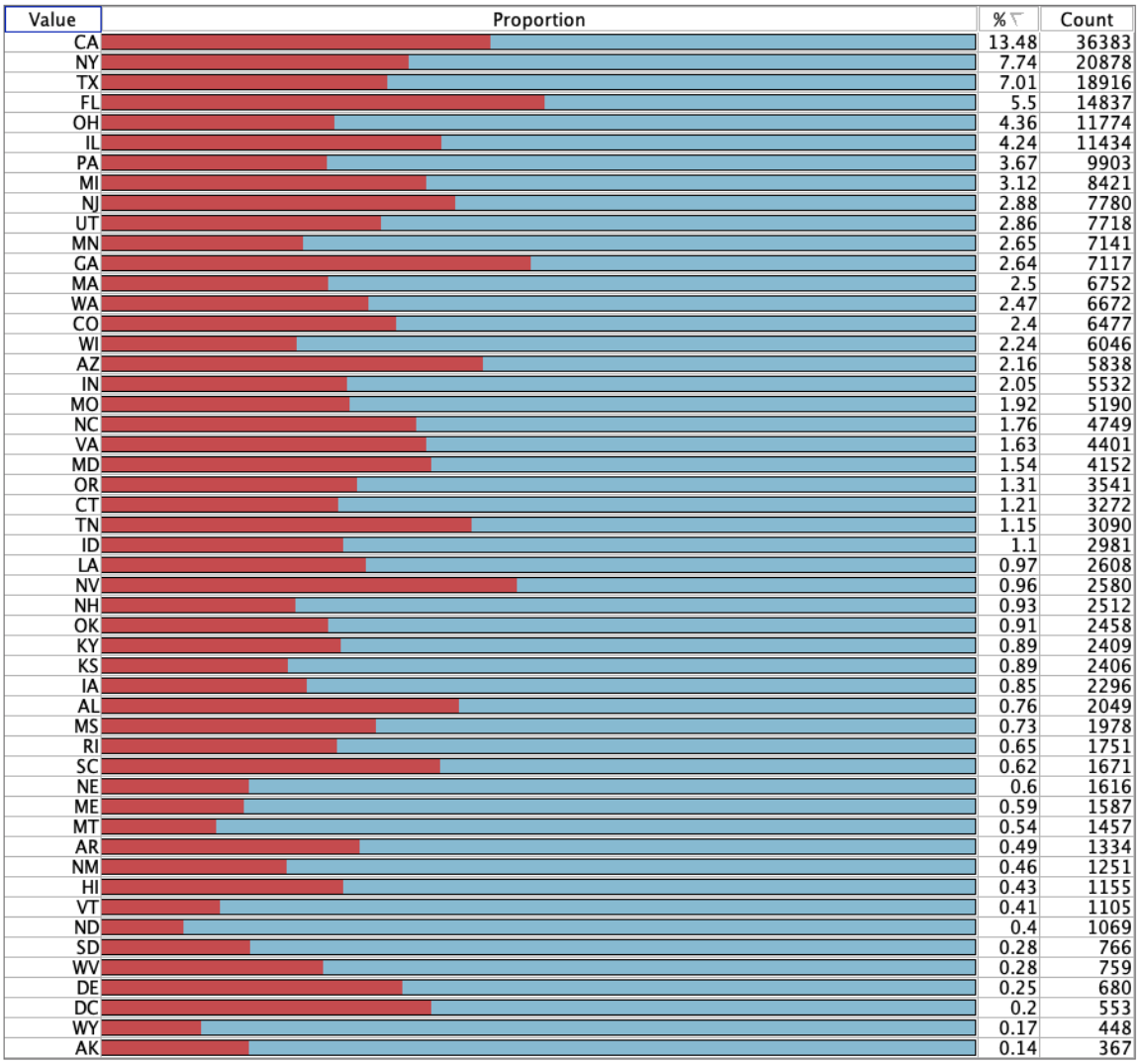
Value	Proportion	%	Count
0		65.36	176378
1		34.64	93482

Figure 3: Distribution of the target variable Default

The distribution shows about 35% of the loans went into default (93,482 records) and about 65% of the loans were paid back in full (176,376).

State

The next variable is *State*. The normalized distribution of records by State is shown in Figure 4. with an overlay of *Default*. The top four states: California, New York, Texas and Florida make up a third of the records, with California having the most (36,383). Viewing these overlays, there is no obvious trend in default based on the number of records.



Value	Proportion	%	Count
CA		13.48	36383
NY		7.74	20878
TX		7.01	18916
FL		5.5	14837
OH		4.36	11774
IL		4.24	11434
PA		3.67	9903
MI		3.12	8421
NJ		2.88	7780
UT		2.86	7718
MN		2.65	7141
GA		2.64	7117
MA		2.5	6752
WA		2.47	6672
CO		2.4	6477
WI		2.24	6046
AZ		2.16	5838
IN		2.05	5532
MO		1.92	5190
NC		1.76	4749
VA		1.63	4401
MD		1.54	4152
OR		1.31	3541
CT		1.21	3272
TN		1.15	3090
ID		1.1	2981
LA		0.97	2608
NV		0.96	2580
NH		0.93	2512
OK		0.91	2458
KY		0.89	2409
KS		0.89	2406
IA		0.85	2296
AL		0.76	2049
MS		0.73	1978
RI		0.65	1751
SC		0.62	1671
NE		0.6	1616
ME		0.59	1587
MT		0.54	1457
AR		0.49	1334
NM		0.46	1251
HI		0.43	1155
VT		0.41	1105
ND		0.4	1069
SD		0.28	766
WV		0.28	759
DE		0.25	680
DC		0.2	553
WY		0.17	448
AK		0.14	367

Figure 4. Normalized Distribution of State with overlay of Default

Figure 5 is a heatmap showing the percentage of defaulted loans by *State* created in Tableau. Viewing the map, there are trends in default based on the region. For example, the West and Southeast have higher default rates, with Florida having the highest at 50.72%. New England and the West North Central have lower default rates, with North Dakota having the lowest at 9.35%. While there are large differences in the number of records between states, the percentage of default changes drastically, meaning the

geographic location may have a role in default. For this reason, *State* was included in the classification models.

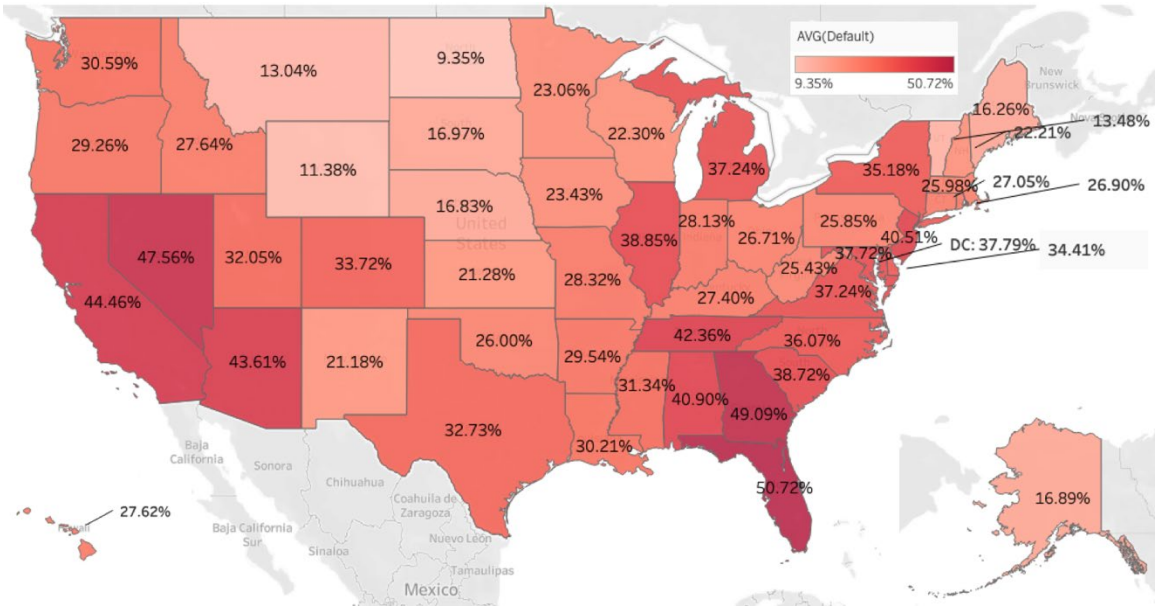


Figure 5: Heatmap of Default Percentage by State

BankState

BankState is the state where the lenders are located, as more of the loans come from out of state rather than within the state. The normalized distribution of records is shown in Figure 6. with an overlay of *Default*. California is the state with the most records (35,914), however the next three states with the most records are different from those in *State*. There is again no obvious trend, however there was one record for Puerto Rico (PR). Since there was only one record for PR, it was removed from the data.

Value	Proportion	%	Count
CA		13.31	35914
NC		12.33	33284
IL		10.24	27623
OH		8.61	23226
RI		6.05	16339
VA		5.67	15300
SD		5.17	13948
NY		4.92	13268
DE		4.31	11619
UT		2.8	7546
TX		2.43	6560
WI		1.58	4253
MN		1.5	4061
FL		1.47	3956
PA		1.34	3617
MO		1.27	3414
OR		1.21	3255
AL		1.11	2986
GA		1.0	2688
MA		0.91	2443
NJ		0.79	2119
CT		0.78	2116
IN		0.75	2032
WA		0.75	2026
CO		0.67	1796
IA		0.65	1759
MI		0.59	1605
KS		0.53	1443
OK		0.53	1433
AR		0.5	1358
NE		0.45	1213
MT		0.43	1169
MS		0.42	1133
ID		0.41	1104
SC		0.4	1086
ND		0.38	1035
NH		0.38	1034
NV		0.38	1030
MD		0.37	985
TN		0.32	875
ME		0.32	862
KY		0.31	844
LA		0.31	840
HI		0.3	813
NM		0.29	784
AZ		0.28	753
VT		0.25	666
WY		0.09	253
WV		0.07	182
DC		0.04	106
AK		0.04	105
PR		0.0	1

Figure 6. Normalized Distribution of BankState with overlay of Default

Figure 7 is heatmap showing the percentage of defaulted loans by *BankState*. There are similar trends from the previous heatmap (Figure 5). The West and Southeast have higher default rates, with Virginia having the highest rate at 51.12%, followed by California and North Carolina at 47.06% and 46.52%, respectively. The West North Central region has lower default rates, with the exception of South Dakota at 39%. Vermont and Colorado are tied for the lowest default rate, 8.41%.

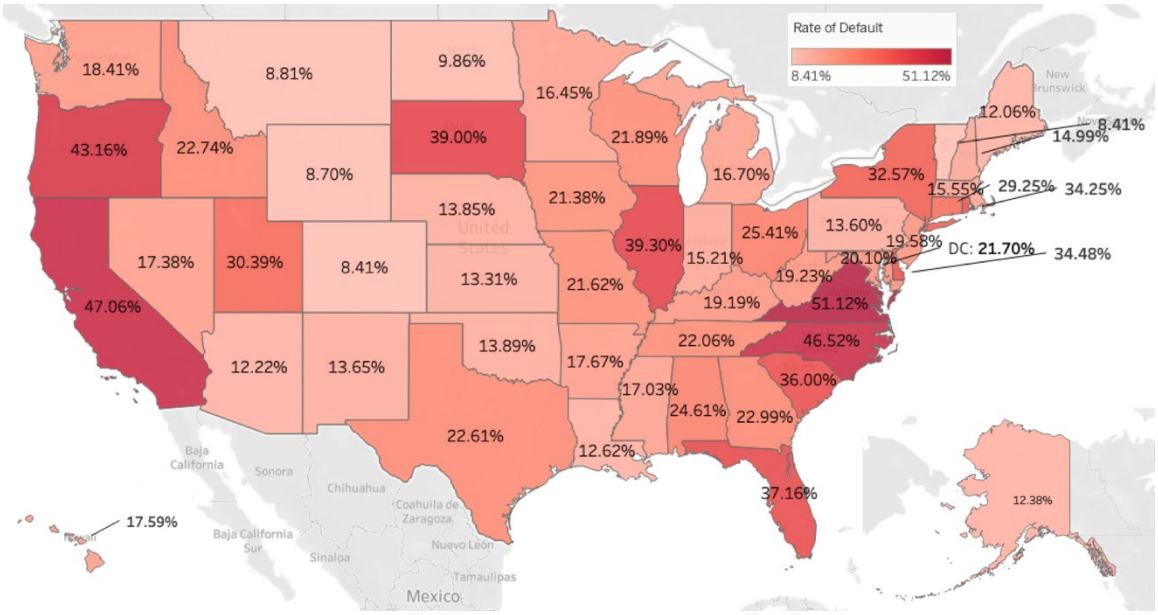


Figure 7: Heatmap of Default Percentage by BankState

NAICS

The North American Industry Classification System (NAICS) is a standard used by Federal statistical agencies in classifying business establishments for the purpose of collecting, analyzing, and publishing statistical data related to the U.S. business economy.[17] It consists of a code with the first 2 digits representing the industry. Using more digits is more specific, for example within Health care (Sector 62), codes starting with 621 are ambulatory health care services and codes starting with 622 are hospitals.[18] The *NAICS* variable lists this code for each record, however only the first two digits representing the industry were used for this analysis. Table 4 shows the first two digits of the codes along with the industry description.

Code	Industry Title	Code	Industry Title
11	Agriculture, Forestry, Fishing & Hunting	53	Real Estate Rental and Leasing
21	Mining	54	Professional, Scientific, and Technical Services
22	Utilities	55	Management of Companies and Enterprises
23	Construction	56	Administrative and Support and Waste Management and Remediation Services
31-33	Manufacturing	61	Educational Services
42	Wholesale Trade	62	Health Care and Social Assistance
44-45	Retail Trade	71	Arts, Entertainment, and Recreation
48-49	Transportation and Warehousing	72	Accommodation and Food Services
51	Information	81	Other Services (except Public Administration)
52	Finance and Insurance	92	Public Administration

Table 4: Description of Industry by first two digits of the NAICS code[19]

[17] https://www.census.gov/eos/www/naics/
[18] https://classcodes.com/lookup/sector-62/
[19] https://www.naics.com/search/

Since only the first two digits were needed to reclassify the codes to their respective industry, the data was exported from Modeler and opened in Excel, where Text to Columns was used to separate the first 2 digits from the rest of the code.

Figure 8 shows the normalized distribution of the *NAICS* variable with an overlay of *Default*. There are 7 records that show as 0.000000 meaning there was no code listed for these records, so these were removed from the dataset. The distribution is seen in Figure 8.

Value	Proportion	%	Count
Retail Trade		17.15	46292
Construction		11.23	30291
Accommodation and Food Ser...		10.68	28808
Professional, Scientific, and T...		10.65	28731
Other Services (except Public ...		9.39	25336
Manufacturing		7.9	21306
Health Care and Social Assista...		6.39	17246
Wholesale Trade		5.99	16171
Administrative and Support a...		5.58	15050
Transportation and Warehous...		4.54	12251
Real Estate Rental and Leasing		2.59	7000
Arts, Entertainment, and Recr...		2.06	5561
Finance and Insurance		1.85	4981
Information		1.82	4922
Educational Services		1.24	3333
Agriculture, Forestry, Fishing ...		0.62	1674
Mining		0.21	564
Utilities		0.07	199
Public Administration		0.03	71
Management of Companies a...		0.02	65

Figure 8. Normalized Distribution of NAICS with overlay of Default

Many of these industries had similar default rates. Real Estate, Rental, and Leasing had the highest, followed by Finance and Insurance. The Mining, Utilities, and Health Care and Social Assistance industries had the lowest default rates.

ApprovalDate

The *ApprovalDate* variable shows the date loans was backed by the SBA. While the date is specific to the day, it was analyzed by Year and Quarter in Tableau.

Year of Approval Date	Quarter of Approval Date	
2005	Q1	19,289
	Q2	19,730
	Q3	19,437
	Q4	16,659
2006	Q1	19,609
	Q2	20,705
	Q3	19,067
	Q4	17,865
2007	Q1	18,575
	Q2	18,675
	Q3	16,759
	Q4	13,006
2008	Q1	11,220
	Q2	8,861
	Q3	6,453
	Q4	4,252
2009	Q1	3,811
	Q2	5,201
	Q3	5,861
	Q4	4,817

Figure 9: Distribution of ApprovalDate

Year 2006 had the most records by year, with Quarter 2 having the most records by Quarter (77,246 and 20,705 respectively). After 2006 the number of records declined by year, with 2009 having the fewest records (19,690).

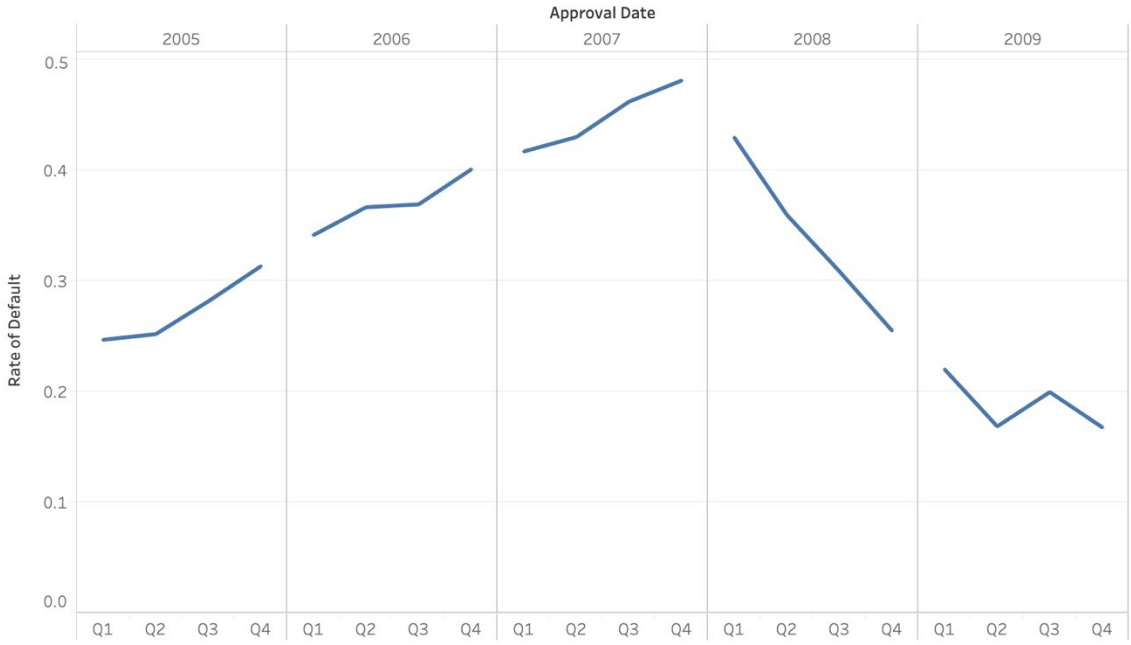


Figure 10: Plot of ApprovalDate against Default

Viewing Figure 10, the rate of Default tends to increase until it peaks in Quarter 4 of 2007. From Quarter 4 of 2007 there was a steep decline in 2008 followed by a slight decrease through 2009.

Term

The *Term* variable represents the term, or maturity date of the loan. Figure 11 shows the statistics.

Term

Statistics

Count	269852
Mean	82.105
Min	0.000
Max	360.000
Range	360.000
Variance	3291.281
Standard Deviation	57.370
Standard Error of Mean	0.110
Median	84.000
Mode	84.000

Figure 11: Output from Statistics Node

The median and most common loan term is 84 months (7 years). All loans are assumed to be SBA 7(a) loans, which have a maximum term of 25 years, or 300 months.[20] Since there were records with terms up to 360 months, records with a term over 300 months were removed. Records where *Term=0* were also removed, as a loan term cannot be zero and are likely errors. Lastly, only loans with a term of at least 6 months were considered, so records with a term between 2 and 5 were also removed. In total this removed 4,474 records. A normalized histogram with an overlay of *Default* can be seen in Figure 14 with a bin width of 12, so each bar represents one year.

[20] https://www.sba.gov/partners/lenders/7a-loan-program/terms-conditions-eligibility#section-header-3

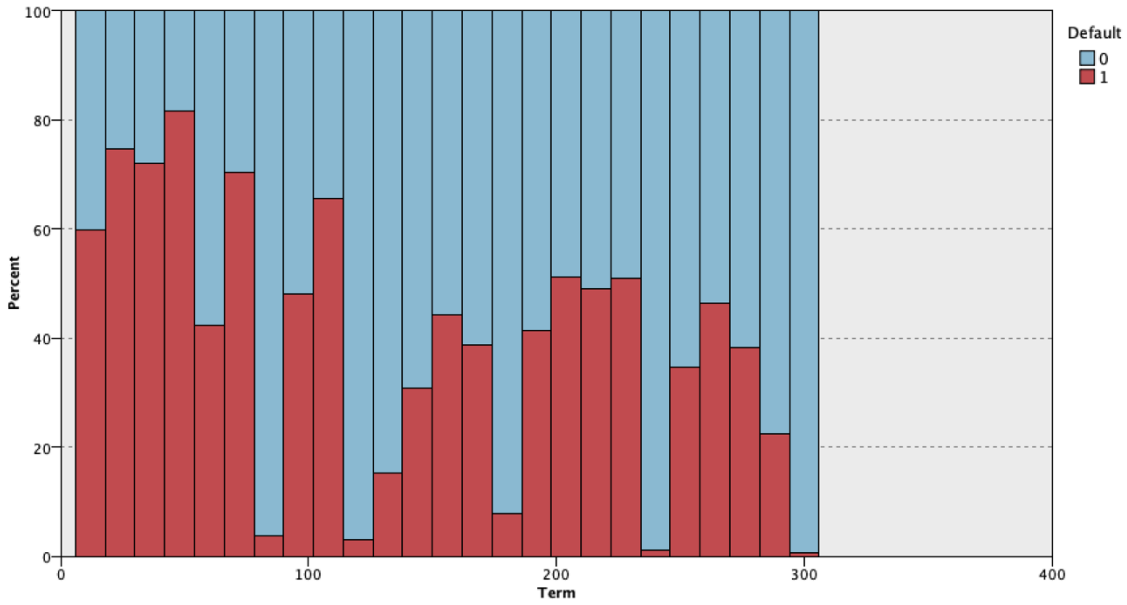


Figure 12: Normalized Histogram of Term with overlay of Default

Viewing the histogram there are very high rates of default for terms up to 72 months, and a general decrease in default rate after 72 months. However, the trend is general and there are clear exceptions.

NoEmp

The *NoEmp* variable represents the number of employees for this business taking out the loan. For this dataset the number of employees range from 0 to 8000. These values were binned into categories by size as defined by the U.S. Bureau of Labor Statistics.[21]

Size Class	Number of Employees
Size Class 1	1-4 employees
Size Class 2	5-9 employees
Size Class 3	10-19 employees
Size Class 4	20-49 employees
Size Class 5	50-99 employees
Size Class 6	100-249 employees
Size Class 7	250-499 employees
Size Class 8	500-999 employees
Size Class 9	1,000 or more employees

Table 5: Size Class by Number of Employees

Since the smallest size class is 1-4 employees, the 4,643 records that show no employees (*NoEmp=0*) were removed. The new variable is named *BusinessSize*.

Viewing the normalized distribution of *BusinessSize*, Size Class 1 has over 60% of the total records and the highest rate of default. The trend is as the number of workers (by

[21] https://www.bls.gov/bdm/bdmfirmsize.htm

Class Size) increases, the rate of default decreases. The exception is Size Class 9, which has the third highest rate of default.

Value	Proportion	%	Count
Size Class 1		60.99	161849
Size Class 2		19.1	50689
Size Class 3		10.62	28179
Size Class 4		5.9	15645
Size Class 9		1.76	4661
Size Class 5		1.2	3196
Size Class 6		0.39	1022
Size Class 7		0.04	109
Size Class 8		0.01	28

Figure 13. Normalized Distribution of BusinessSize with overlay of Default

NewExist

The *NewExist* variable represents whether the business is new (less than 2 years) or existing (more than 2 years. This is a flag variable with values of *N* and *E*, respectively. The normalized distribution is shown in Figure 14.

Value	Proportion	%	Count
E		67.84	180020
N		32.16	85358

Figure 14. Normalized Distribution of NewExist with overlay of Default

Viewing the distribution, about two thirds of the loans were given to existing businesses, and the remaining one third to new businesses, and there was a slightly higher rate of default for loans given to existing businesses as opposed to new businesses.

CreateJob

The *CreateJob* variable lists the number of jobs created by the company. The distribution is shown in Figure 15 using a histogram with 10 bins and an overlay of *Default*.

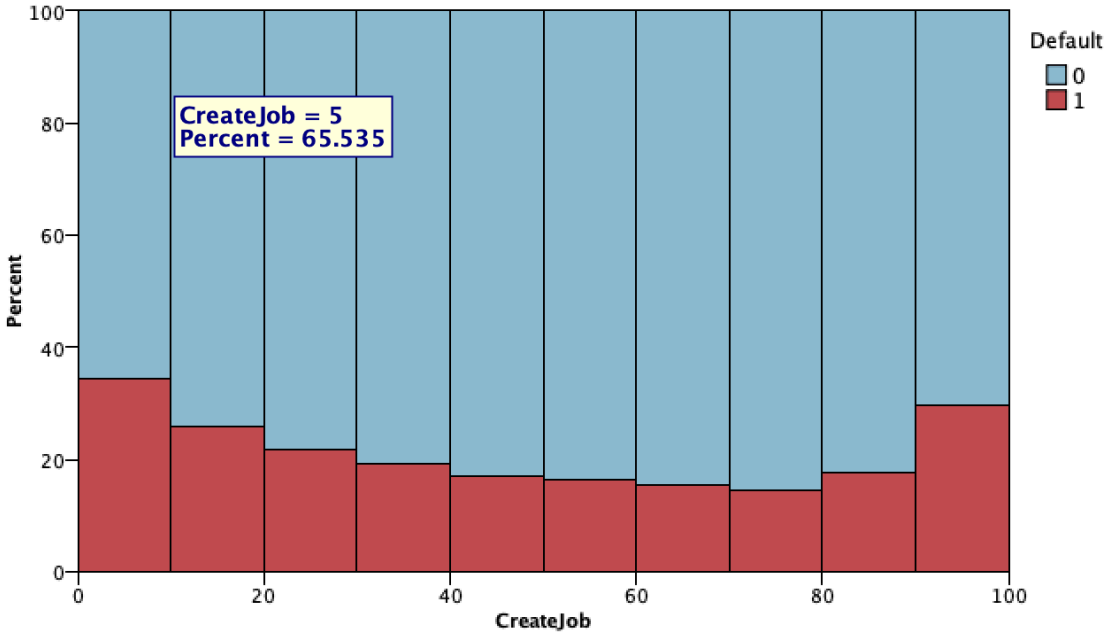


Figure 15: Histogram of CreateJob with overlay of Default

Viewing the histogram, over 65% of the records in the dataset are in the first bin. Running a Statistics Node showed the median and mode for this variable were both 0, so at least half of the records did not create new jobs. For this reason, the variable was reclassified to a flag variable named *CreateJob?*. Values of 0 were changed *N* for no and all other values were changed to *Y* for yes.

CreateJob

Statistics

Count	265378
Mean	2.458
Min	0.000
Max	2020.000
Range	2020.000
Variance	180.197
Standard Deviation	13.424
Standard Error of Mean	0.026
Median	0.000
Mode	0.000

Figure 16: Output from Statistics Node for CreateJob

The normalized distribution does not show much difference for this newly defined variable. About 55% of these records did not create new jobs as opposed to the 45% that did. However, the default rate for companies that did not create new jobs is slightly lower compared to companies that did.

Value	Proportion	%	Count
N		55.79	148062
Y		44.21	117316

Figure 17: Normalized Distribution of CreateJob? with overlay of Default

RetainedJob

The *RetainedJob* variable shows the number of jobs retained by the company. The distribution is shown in Figure 18 using a histogram with 10 bins and an overlay of *Default*.

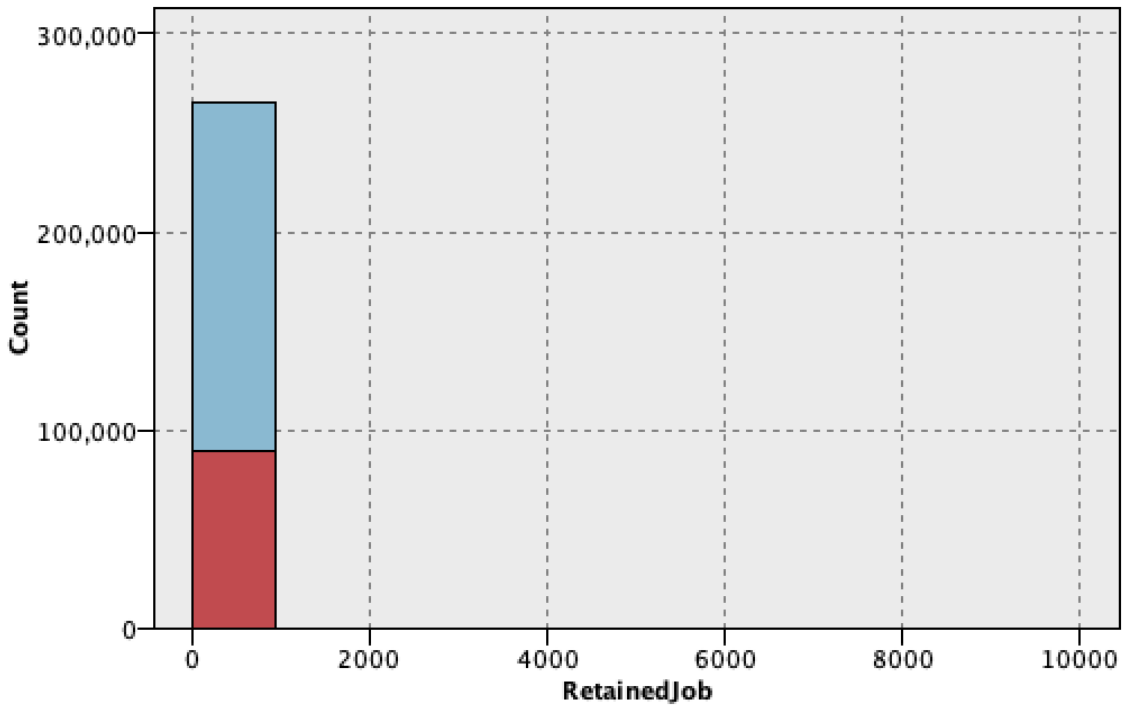


Figure 18: Histogram of RetainedJob with overlay of Default

Similar to *CreateJob*, most of the records in the dataset are in the first bin. Running a Statistics Node showed the mode to be 1. This variable was reclassified into a new variable *RetainedJob?* similar to *CreateJob?*, however this new variable is categorical with 3 categories: No retained Jobs (0), one retained job (1), or multiple retained jobs (2 or more).

RetainedJob

Statistics

Count	265378
Mean	6.000
Min	0.000
Max	9500.000
Range	9500.000
Variance	688.311
Standard Deviation	26.236
Standard Error of Mean	0.051
Median	3.000
Mode	1.000

Figure 19: Output from Statistics Node for RetainedJob

The normalized distribution shows that over two thirds of the records retained multiple jobs, with ne retained job and no retained jobs accounting for about 19% and 13%, respectively. One retained job had the highest default rate, and no retained jobs had the lowest default rate.

Value	Proportion	%	Count
Multiple		67.52	179172
One		18.98	50380
No		13.5	35826

Figure 20: Normalized Distribution of RetainedJob? with overlay of Default

UrbanRural

The normalized distribution of *UrbanRural* is shown in Figure 21.

Value	Proportion	%	Count
U		80.85	214570
R		19.15	50808

Figure 21: Normalized Distribution of UrbanRural with overlay of Default

About 80% of the records are from urban areas as opposed to 20% from rural areas. The default rate for urban areas was also higher compared to rural areas.

FranchiseCode

The Franchise code is a code created by the SBA and assigned to companies. Companies that are listed in the SBA Franchise Directory are eligible for financial assistance such as SBA loans.[22]

Franchise Code	Count of Franchise Code
0	150,052
1	104,324
3	1
395	5
399	1
401	8
404	1
407	33
414	2
416	3
419	1
420	3
426	1
452	22
470	2
485	4
800	7
835	2
900	5
944	1
950	7
1350	71

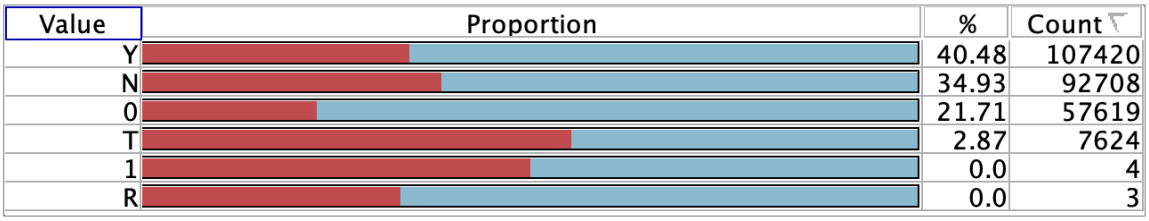
Figure 22: Count of Values from FranchiseCode

[22] https://www.sba.gov/sba-franchise-directory

Franchise codes 00000 and 00001 (shown as 0 and 1 as leading zeroes are removed) indicate no franchise. These values account for over 95% of all records. Since most of these records are not franchises, this variable was not used for modeling.

RevLineCr

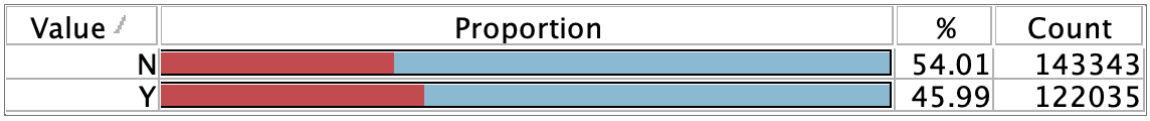
The *RevLineCr* variable is a flag variable with values of yes (Y) or no (N) indicating whether or not the company taking the loan had a revolving line of credit. Figure 23 shows the distribution of records.



Value	Proportion	%	Count
Y		40.48	107420
N		34.93	92708
0		21.71	57619
T		2.87	7624
1		0.0	4
R		0.0	3

Figure 23: Normalized Distribution of RevLineCr with overlay of Default

Viewing the distribution, values of "0", "T", "1", and "R", and are likely to be data entry errors with the exception of 0, which accounts for over 21%. To remove these values, they were reclassified as blank values. These blanks were then replaced with "Y" or "N" values using the CART algorithm through a Data Audit Node, as used in the data cleaning section.



Value	Proportion	%	Count
N		54.01	143343
Y		45.99	122035

Figure 24: Normalized Distribution of RevLineCr with overlay of Default

The values for *RevLineCr* are fairly split. There are roughly 19,000 more records that do not have revolving lines of credit compared to those that do, and the rate of default was slightly higher for businesses with revolving lines of credit.

LowDoc

LowDoc is another flag variable indicating whether or not the company taking out the loan is part of the LowDoc, or low documentation plan. This is a special plan created by the SBA that promises quick processing for loan amounts under $150,000, making it faster and easier to apply for and receive an SBA 7(a) loan.[23] The distribution is shown in Figure 25.

Value	Proportion	%	Count
N		98.34	260980
Y		1.25	3310
A		0.17	462
S		0.16	431
C		0.05	138
R		0.02	56
1		0.0	1

Figure 25: Normalized Distribution of LowDoc with overlay of Default

The distribution shows more potential data entry errors: “A”, “S”, “C”, “R”, and “1”. However, over 98% of the records are classified as *N* (not part of the program), so this variable was omitted.

[23] https://www.entrepreneur.com/encyclopedia/lowdoc

DisbursementGross

The *DisbursementGross* variable lists the amount of the loan given to the business. The distribution can be seen in Figure 26.

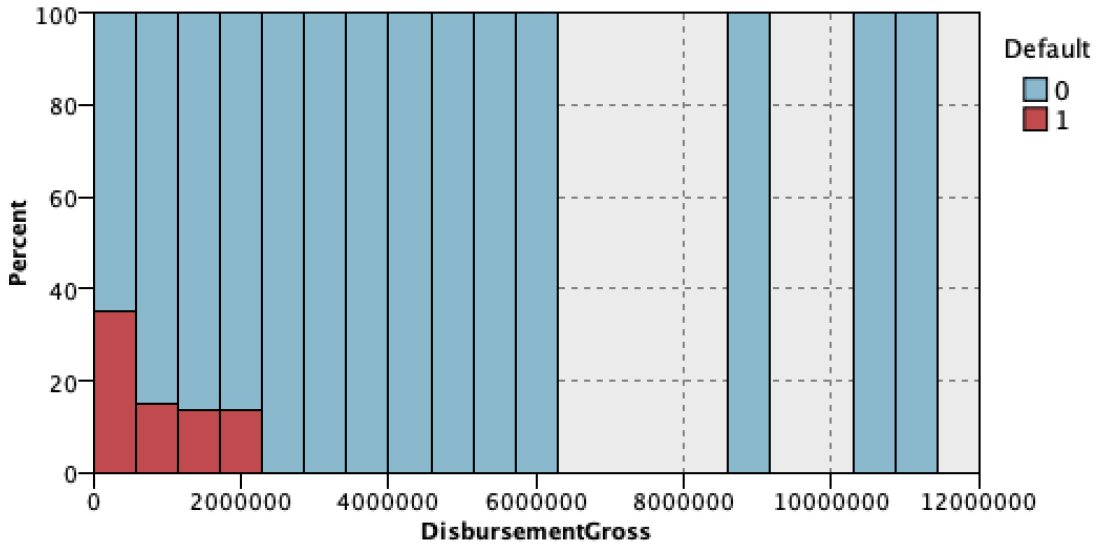


Figure 26: Normalized Histogram of DisbursementGross with overlay of Default

The first bin has the highest rate of default upwards of 40%. This rate is cut in half for the next bin and continues to slowly decrease. Viewing the statistics, a majority of the loans are loans under $1,000,000 as the median is $62,500 and the 90th percentile is $395,000. The statistics also show a minimum of $0, and these $0 values are likely errors in the data. These records along with any records with values under $1000 were not considered, which removes 289 records. Figure 27 shows all loans over $2,000,000.

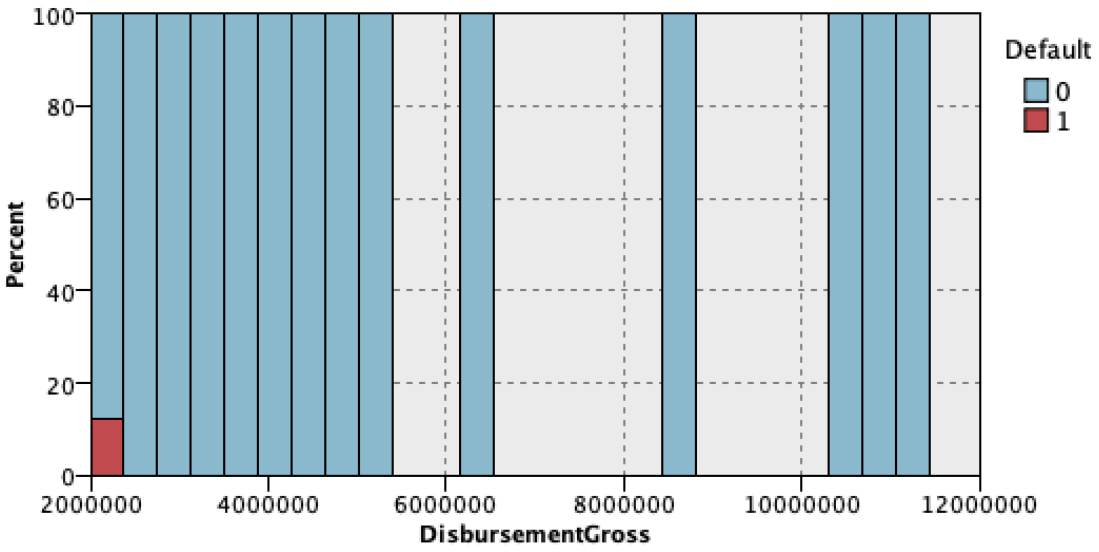


Figure 27: Histogram of DisbursementGross ≥ $2,000,000 with overlay of Default

Most of these loans are between $2,000,000 and $2,200,000, as the bins have a width of $200,000, and only 7 loans over $5,000,000. SBA 7(a) loans have a maximum dispersible amount of $5,000,000, so these 7 records were removed from the dataset. The remaining loans were plotted on a normalized histogram with an overlay of *Default*.

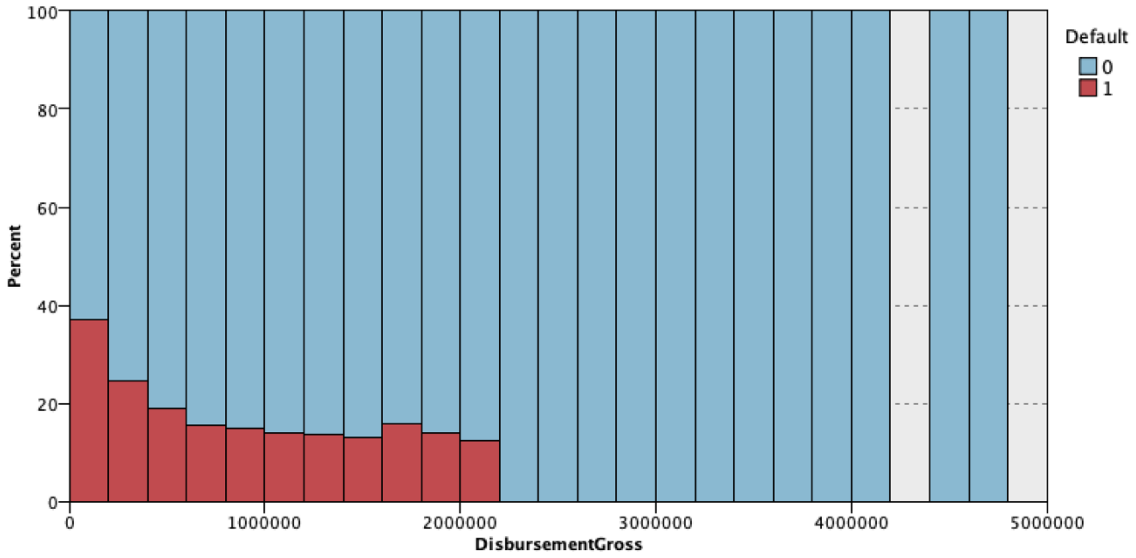


Figure 28: Normalized Histogram of DisbursementGross with overlay of Default

The highest default rate is for loans under $200,000 and is upwards of 40%. Generally, as the loan amount increases, the default rate decreases, and it levels out around the $1,000,000 mark. For all loans over $2,200,000 there were no cases of default.

Phase 4: Setup Phase

After completing the EDA phase 265,082 records and 13 variables remain for modeling: 12 predictors along with our target variable *Default*.

Variable Name	Variable Type
Default	Flag
State	Nominal
BankState	Nominal
NAICS	Nominal
ApprovalDate	Continuous
Term	Continuous
NewExist	Flag
UrbanRural	Flag
RevLineCr	Flag
DisbursementGross	Continuous
BusinessSize	Nominal
CreateJob?	Flag
RetainedJob?	Nominal

Table 6: Table of remaining 13 variables

To ensure there was no multicollinearity, a Multiple Regression model was created and analyzed using the Variance Inflation Factor (VIF). The VIF measures how much variance of a regression coefficient is inflated due to a multicollinearity model. As a rule of thumb, a VIF value that exceeds 5 or 10 indicates a problematic amount of collinearity.[24] Set-to-Flag nodes were used to create indicator variables for all variables

[24] http://www.sthda.com/english/articles/39-regression-model-diagnostics/160-multicollinearity-essentials-and-vif-in-r/

except *Term* and *DisbursementGross*. Viewing the VIF values from our model, none appear to be high enough to indicate a problematic amount of multicollinearity. The highest was for *BankState=MT* (3.713), however a majority of the indicators had VIF values below 1.5.

Prior to modeling it was necessary to partition the data. The data was partitioned into two mutually exclusive datasets: A training dataset and a test dataset. In practice, data is usually split randomly 70-30 or 80-20 into train and test datasets respectively in statistical modeling, in which training data is utilized for building the model and its effectiveness is checked on test data.[25] A 70-30 split was used for this analysis as this is a very common split for large datasets. The training data, *SBA_Training*, contained 70% of the total records. The remaining 30% comprised the test dataset, *SBA_Test*.

These partitioned datasets were validated using statistical tests. The two remaining continuous variables were tested, *Term* and *DisbursementGross*, which was done using the two-sample t-test. For each t-test, $\alpha = 0.05$ was used.

[25] https://dataandbeyond.wordpress.com/2017/08/24/split-of-train-and-test-data/

Validating the Partition

The two-sample t-test is the method used with non-standardized continuous predictors. The hypotheses are defined as the following:

$$\boldsymbol{H_0}: \boldsymbol{\mu_{training} = \mu_{test}}$$

$$\boldsymbol{H_A}: \boldsymbol{\mu_{training} \neq \mu_{test}}$$

The null hypothesis H_0 states that the training and test datasets are not significantly different, whereas the alternative hypothesis H_A states they are significantly different. The test statistic t for this method is calculated using

$$t = \frac{\bar{x}_{training} - \bar{x}_{test}}{\sqrt{\frac{s^2_{training}}{n_{training}} + \frac{s^2_{test}}{n_{test}}}}$$

where $\bar{x}$ is the mean, s is the standard deviation and n is the number of records. These values were calculated using a Statistics Node and are shown in Table 7.

Training	**Test**
$\bar{x}_{training} = 83.016$	$\bar{x}_{test} = 83.089$
$s^2_{training} = 3182.224$	$s^2_{test} = 3176.580$
$n_{training} = 186{,}055$	$n_{test} = 79{,}316$

Table 7: Statistics for Term

Using these values, the test statistic t was calculated to be -0.31. The p-value was calculated with degrees of freedom the smaller of $n_{training} - 1$ and $n_{test} - 1$. Using the smaller number, $n_{test} - 1$, this resulted in a p-value of 0.7566. This value is not less than 0.05, therefore H_0 was not rejected, indicating there was not sufficient evidence that *Term* differed between the training and test datasets.

The same procedure was used for *DisbursementGross*. The statistics are shown in Table 8.

Training	Test
$\bar{x}_{training} = 161712.762$	$\bar{x}_{test} = 163065.191$
$s^2_{training} = 81384718288.461$	$s^2_{test} = 83250172020.377$
$n_{training} = 186{,}055$	$n_{test} = 79{,}316$

Table 8: Statistics for DisbursementGross

Using these values, the test statistic *t* was calculated to be -1.11. The p-value was again calculated with degrees of freedom the smaller of $n_{training} - 1$ and $n_{test} - 1$, which resulted in a p-value of 0.2670. This value is not less than 0.05, therefore H_0 was not rejected, indicating there was not sufficient evidence that *DisbursementGross* differed between the training and test datasets.

With the partition validated, the data was rebalanced as a surrogate for misclassification costs. This is necessary for models such as Neural Networks and Random Forests, as there is no option to rebalance the data within the respective model nodes. This changed the error cost ratio between false negatives and false positives, which are the two classifiers that impact average profit. Many classification models have the option to adjust the cost matrix before running the model. The cost matrix layout is shown in the following table:

Predicted Category			
Actual Category		0	1
	0	**TN**	**FP**
	1	**FN**	**TP**

Layout of the confusion matrix

The binary classifiers are defined as the following:

True Negative (TN): Correctly predicting that a borrower will not default on their loan. The cost for the borrower is the average interest on the given loans. (Negative amount represents cost for the borrower).

False Positive (FP): Incorrectly predicting that a borrower will default on their loan. There are no incurred costs for either side, since no loan is given. Potential revenue is not considered.

False Negative (FN): Incorrectly predicting that a borrower will not default on their loan. This is the worst option for the lender, since the loan would be approved and defaulted on. We assume a portion of the loan (Or possibly all) is paid back for those who default, so the cost for the loan company is *Amount_Lost*.

True Positive (TP): Correctly predicting that a borrower will default on their loan. There are no incurred costs for either side, since no loan is given.

Positive classifications result in no potential loss or gain for the lender, so values of $0 were placed in the cost matrix. A TN response correctly predicts that a borrower will not default on their loan. Borrowers will have the entire loan paid plus interest, which is profit for the lenders. Across all loans, the average interest, *Interest_Paid*, was $94,661. This was expressed as a negative amount in the cost matrix, as this represented a $94,661 loss for the borrower. A FN response incorrectly predicts a business will pay back the loan, so the business will default on the loan. This meant the lender lost the amount on the loan that has not been paid. However, the SBA guaranteed up to 85% of the approved amount on these loans, so lenders did not lose as much money. Across all loans, the average loss on a loan, *Amount_Lost*, was $4,754.58 and is represented as a positive number as it was a loss to the lenders and benefited the businesses.

Predicted Category			
Actual Category		0	1
	0	*-$94,661.53*	*$0*
	1	*$4,754.58*	*$0*

Cost Matrix

From this cost matrix, we manipulated the top row by adding $94,661.53, which removed the negative amount. This is an application of *Decision Invariance Under Row Adjustment*, which states a classification decision is not changed by the addition or subtraction of a constant from the cells in the same row of a cost matrix. Doing so resulted in the adjusted cost matrix.

Predicted Category			
Actual Category		0	1
	0	*0*	*$94,661.53*
	1	*$4,754.58*	*0*

Adjusted Cost Matrix

The adjusted cost matrix was further simplified using *Decision Invariance Under Scaling*, which states a classification decision is not changed by scaling a constant in all cells of the cost matrix (Larose, D. T., Larose, C. D. 2015). Each cell was divided by $4,754.58 to simplify the adjusted cost matrix, reducing the FP value to 1.

Predicted Category			
Actual Category		0	1
	0	*0*	*19.9*
	1	*1*	*0*

Simplified Cost Matrix

This resulted in the simplified cost matrix. This matrix was used for the models where misclassification costs could be defined. However, models such as Neural Networks and Random Forest do not offer this in Modeler, so the data was rebalanced as a surrogate for misclassification costs for the test set prior to running these models.

Phase 5: Modeling Phase

There were 6 different classifications that were created on the training data. For each, 2 models were created. The first was a "naïve" model, which was run on the unbalanced data and without misclassification costs. These naïve models were used as a baseline for comparison to the models with misclassification costs. The second model utilized misclassification costs or rebalancing. These models were then run using the test data, and the results were evaluated and compared in Phase 6. The following are the 6 models that were used:

- CART
- C5.0
- CHAID
- QUEST
- Neural Network (NN)
- Random Forest (RF)

With the Setup Phase complete, the Modeling Phase began with the CART models. Figures 29a and 29b show the outputs from both models. The models are very similar, as they almost solely used *Term* for the splits starting with 81.5, very close to the median *Term* value of 84. The only split not using *Term* was one of the last splits, using *Disbursement_Gross* at $32,070.50. After these models were run through the test data, Matrix Nodes were connected and run to determine the error matrices, which showed

both the correct and incorrect classifications made by the models. The matrices for the CART models are shown in Table 9.

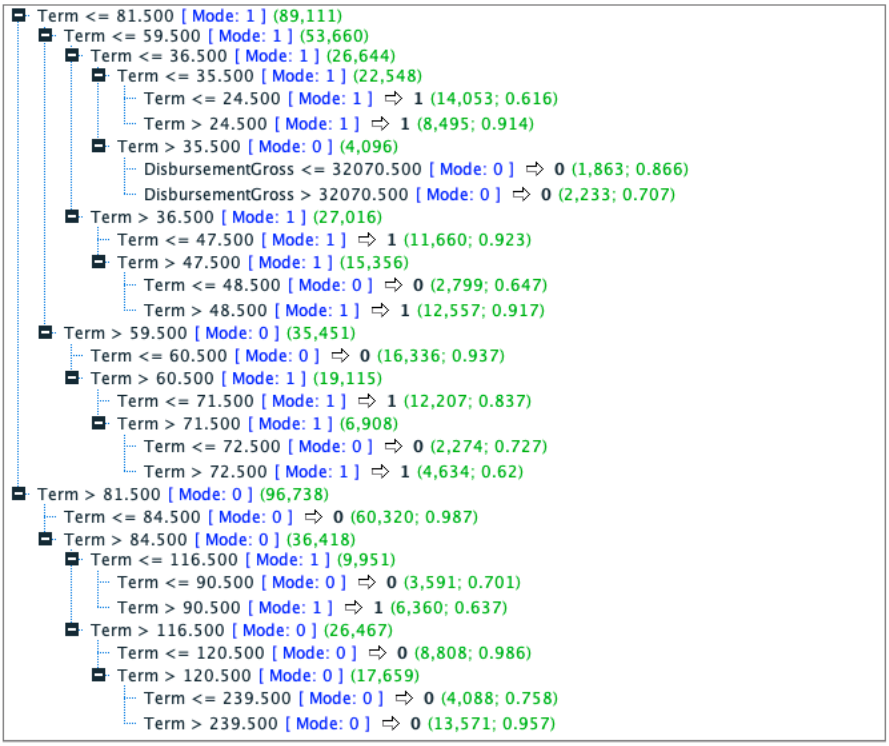


Figure 29a. Output from Naïve CART Model

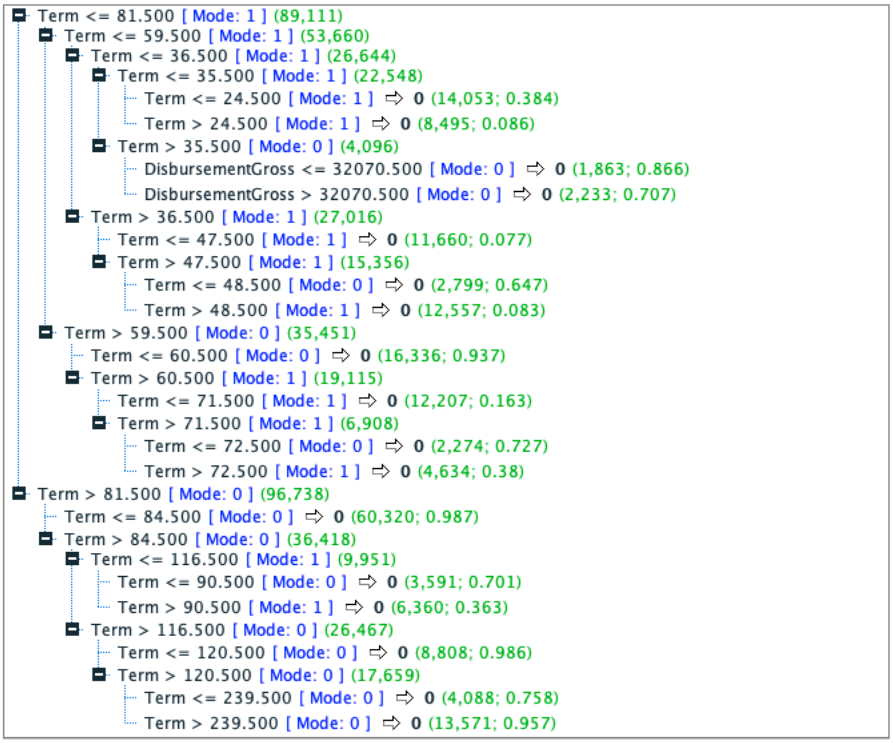


Figure 29b. Output from CART Model with MCC

Naïve CART Model			
Predicted Category			
Actual Category		0	1
	0	**46,390**	**6,009**
	1	**3,177**	**23,657**

CART Model with MCC			
Predicted Category			
Actual Category		0	1
	0	**52,399**	**0**
	1	**26,834**	**0**

Table 9: Matrices for CART Models

Viewing these matrices, the model with MCC had no positive classifications, essentially making it All-Negative model. While this model seems undesirable, it is difficult to make any assumptions solely by the contingency tables. The evaluation costs along with the profits are included in the Evaluation Phase.

Next are the C5.0 models. The outputs are shown in Figures 30a and 30b. Both models used *Term* for the root node, however the split values differ. The naïve model used 81.5 similar to both CART models, however the model with MCC used a split value of 59.9. The remaining splits for these models varied greatly. The matrices are shown in Table 10.

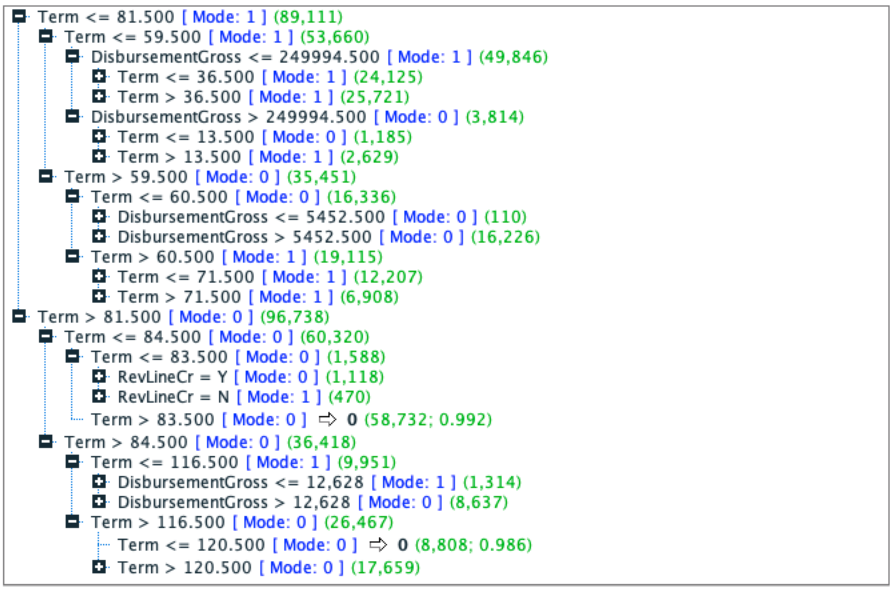


Figure 30a. Output from Naïve C5.0 Model

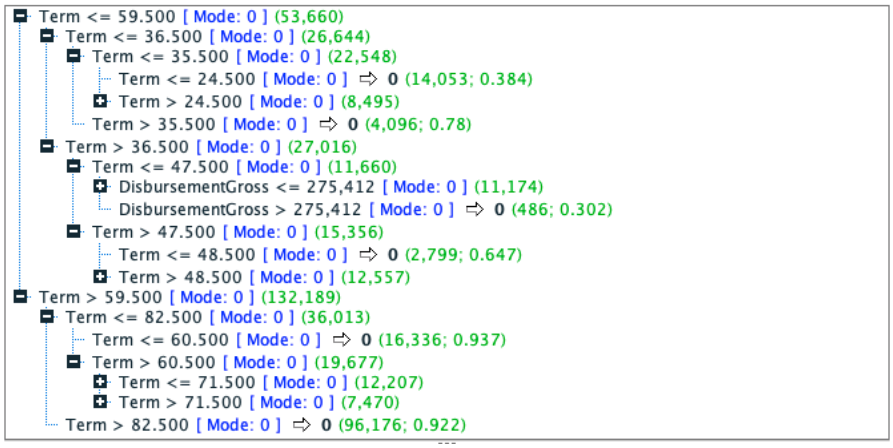


Figure 30b. Output from C5.0 Model with MCC

Naïve C5.0 Model			
Predicted Category			
Actual Category		0	1
	0	**49,159**	**3,240**
	1	**2,555**	**24,279**

C5.0 Model with MCC			
Predicted Category			
Actual Category		0	1
	0	**52,079**	**320**
	1	**17,863**	**8,971**

Table 10: Matrices for C5.0 Models

The matrices show fairly similar TN values, while the rest of the values vary significantly. The naïve model had many more TP classifications. The naïve model also had many more FP classifications while the model with MCC had more FN classifications.

Thirdly the CHAID models. The outputs are shown in Figures 31a and 31b. Each had multiple root nodes, as CHAID models are not restricted to binary splits, however both models used *Term* with the same split values. The remaining splits are similar between both models. Both utilized *Bank_State* for most of the second splits, followed by *State*, *RevLineCr*, *Disbursement_Gross* and again *Bank_State* for the remaining splits. The matrices are shown in Table 11.

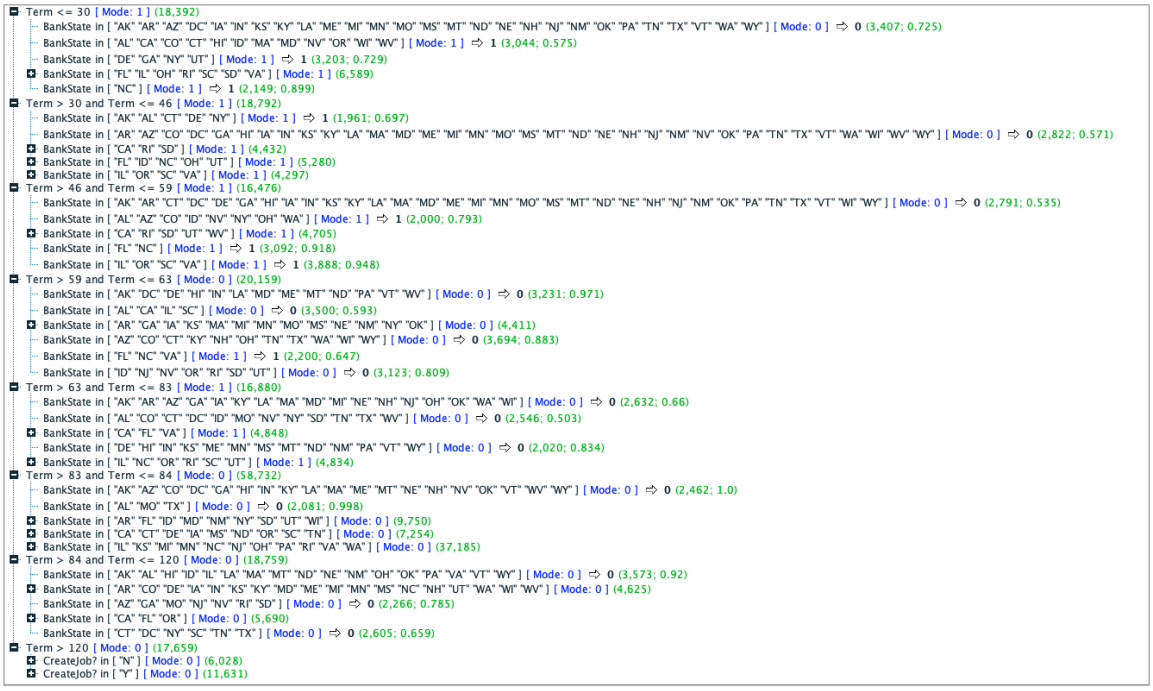


Figure 31a. Output from Naïve CHAID Model

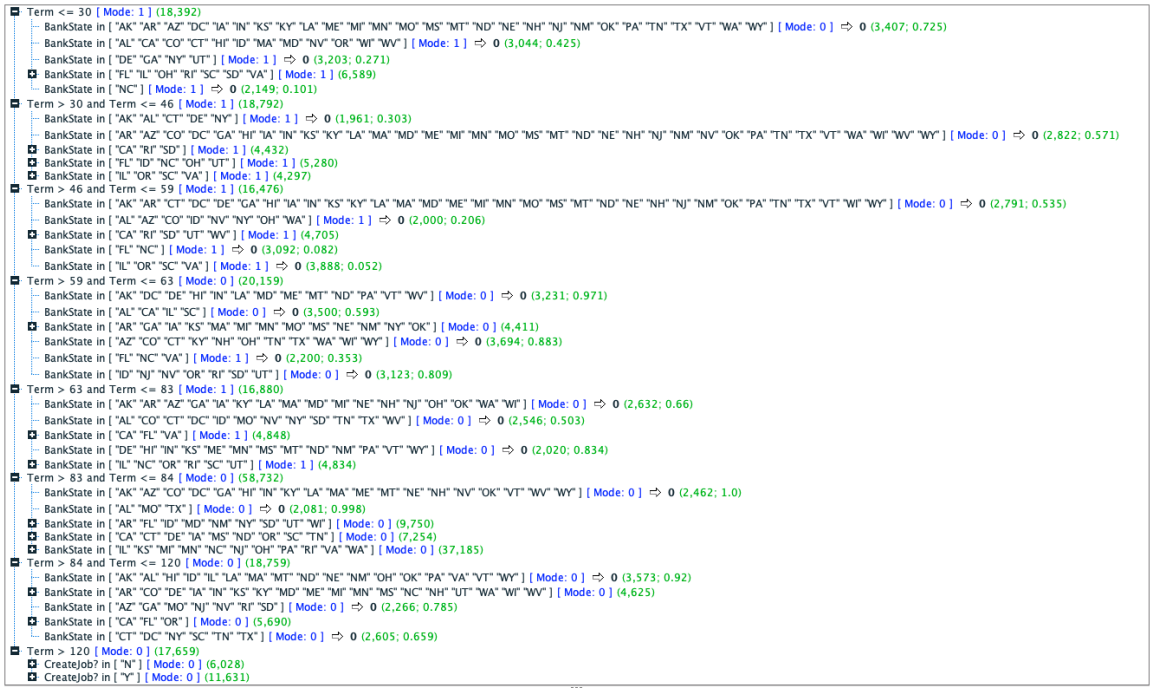


Figure 31b. Output from CHAID Model with MCC

Naïve CHAID Model			
Predicted Category			
Actual Category		0	1
	0	**47,988**	**4,411**
	1	**6,108**	**20,726**

CHAID Model with MCC			
Predicted Category			
Actual Category		0	1
	0	**52,351**	**48**
	1	**25,835**	**999**

Table 11: Matrices for CHAID Models

Again, these models have fairly similar TN classifications while the other are very different. Similar to the C5.0 models, the naïve CHAID model had more FP and many more TP classifications while the model with MCC had more FN classifications.

Next are the QUEST models. The outputs are shown in Figures 32a and 32b. Both models only had one split using *Bank_State*, and both splits predicted *Default* to be 0. The matrices are shown in Table 12.

Figure 32a. Output from Naïve QUEST Model

Figure 32b. Output from QUEST Model with MCC

Naïve QUEST Model			
Predicted Category			
Actual Category		0	1
	0	**52,399**	**0**
	1	**26,834**	**0**

QUEST Model with MCC			
Predicted Category			
Actual Category		0	1
	0	**52,399**	**0**
	1	**26,834**	**0**

Table 12: Matrices for QUEST Models

Since both models only predicted *Default* to be 0, this created two All-Negative models similar to the CART model with MCC. While these models were not so useful, they were still evaluated in the next phase.

The next method was Neural Networks. Neural Networks works quite differently than our other models. Also known as Artificial Neural Networks, this method is designed to simulate the way the human brain analyzes and processes information, with neuron nodes interconnected like a web.[26] The outputs are shown in Figures 33a and 33b.

[26] https://www.investopedia.com/terms/a/artificial-neural-networks-ann.asp

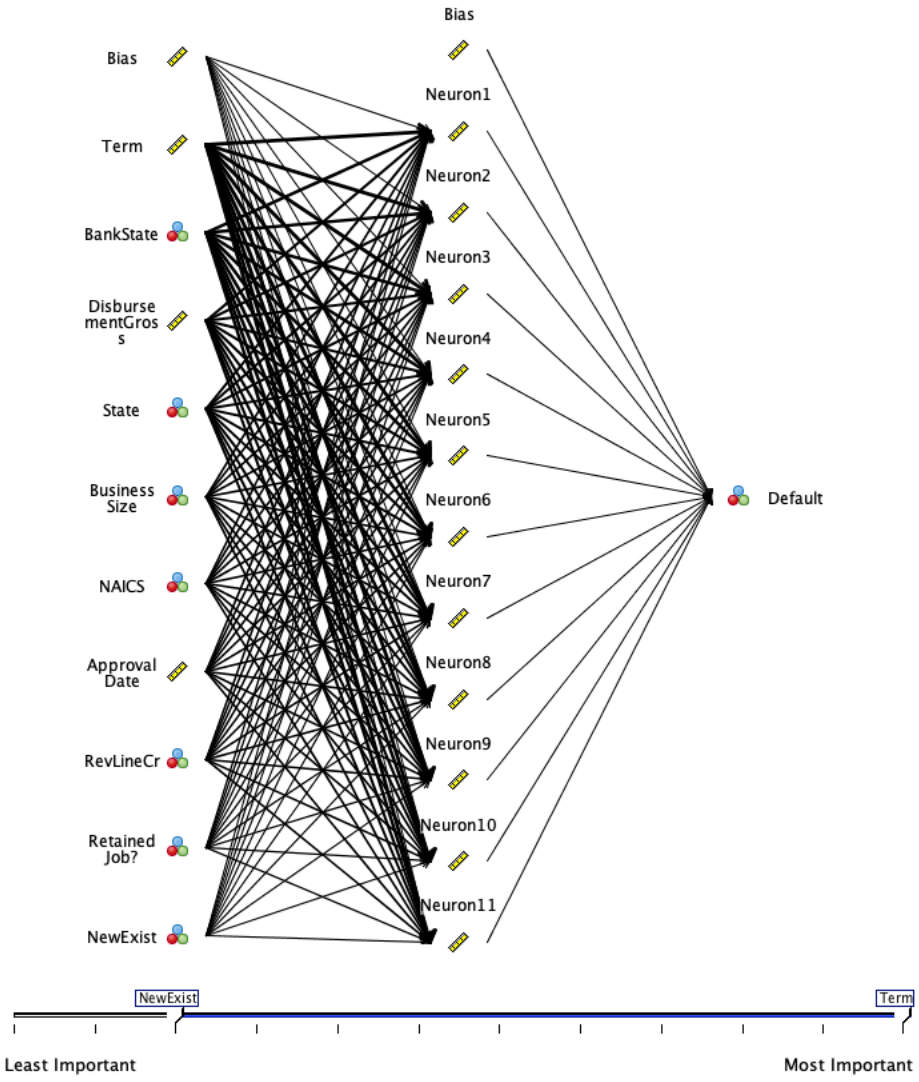


Figure 33a. Output from Naïve NN Model

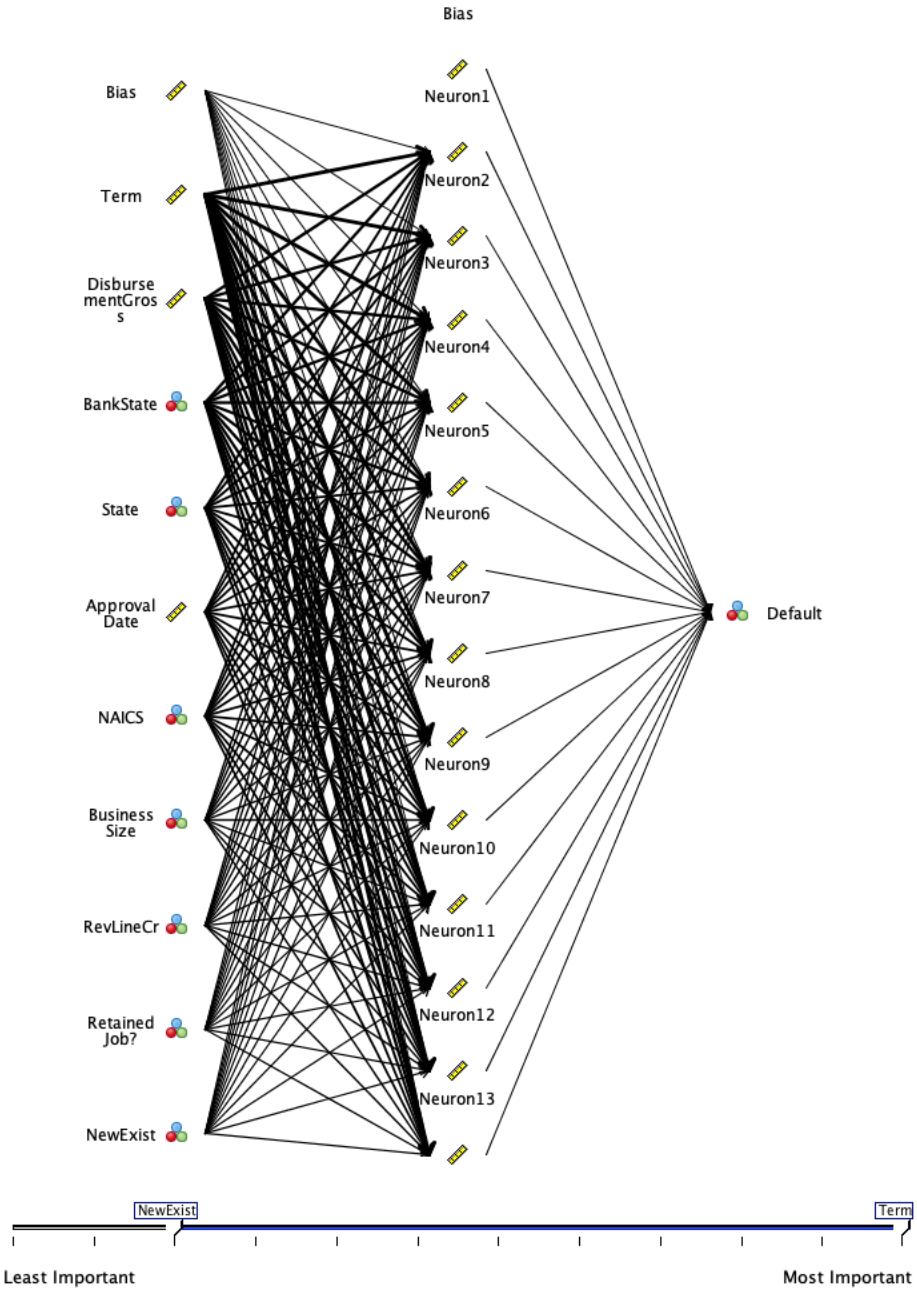


Figure 33b. Output from Rebalanced NN Model

Naïve NN Model			
Predicted Category			
Actual Category		0	1
	0	**48,005**	**4,394**
	1	**5,153**	**21,681**

Rebalanced NN Model			
Predicted Category			
Actual Category		0	1
	0	**52,318**	**81**
	1	**24,979**	**1,855**

Table 13: Matrices for NN Models

These matrices are consistent with those from our previous models. The Rebalanced Model had much fewer positive predictions compared to the naïve model and had many more FN predictions.

The last models tested were Random Forest models. The outputs are shown in Figures 34a and 34b. For RF, the outputs show predictor importance. Both models have *Term* as the most important predictor, and *DisbursementGross* and *ApprovalDate* as the next two, while in different orders between both models. After the top three, the order is very different. For example, both utilize *Bank_State*, however the naïve model uses states *CA* and *NC* with the model with MCC uses *IL* and *VA*. The model with MCC also uses three of the *NAICS* categories while the naïve model does not use any.

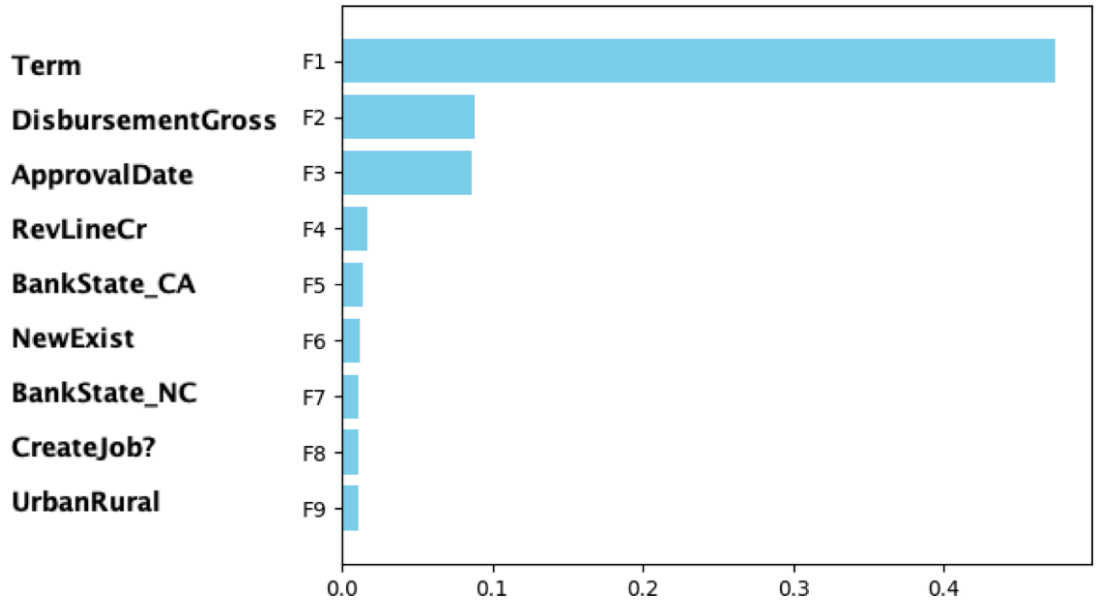


Figure 34a. Output from Naïve RF Model

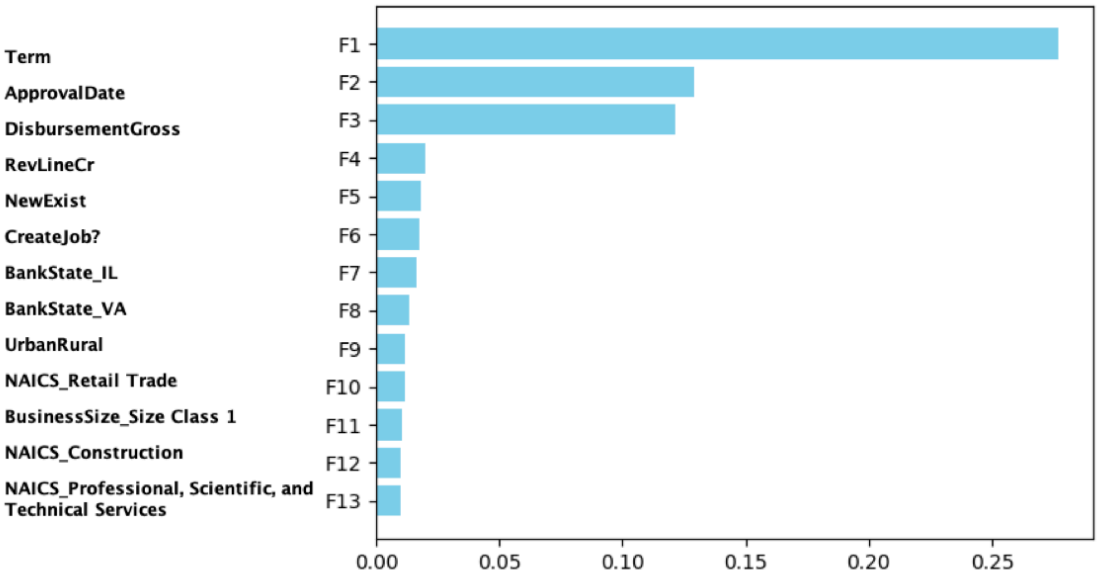


Figure 34b. Output from Rebalanced RF Model

Naïve RF Model			
Predicted Category			
Actual Category		0	1
	0	**50,011**	**2,388**
	1	**5,006**	**21,828**

Rebalanced RF Model			
Predicted Category			
Actual Category		0	1
	0	**50,645**	**1,754**
	1	**6,273**	**20,561**

Table 14: Matrices for RF Models

Viewing these matrices, the Rebalanced RF Model is the only non-naïve model that did not have far fewer positive predictions. While still fewer compared to the naïve model, there were still over 22,000 positive predictions, and the number of negative predictions were fairly similar between the two models.

Phase 6: Evaluation Phase

Once all models were executed using the test data, the results were displayed and compared by the following measures:

Accuracy

Accuracy is an overall measure of the proportion of correct classifications being made by the model.

$$\frac{TN + TP}{TN + FN + FP + TP}$$

Sensitivity

Sensitivity measures the proportion of all positive records predicted by the model. It is a measure of the ability to classify a record positively.

$$\frac{TP}{TP + FN}$$

Specificity

Specificity measures the proportion of all negative records predicted by the model. It is a measure of the ability to classify a record negatively.

$$\frac{TN}{FP + TN}$$

Proportion of True Positives

The Proportion of True Positives measures the proportion of all positive classifications that are actually positive. This is also known as Precision.

$$\frac{TP}{FP + TP}$$

Proportion of True Negatives

The Proportion of True Negatives measures the proportion of all negative classifications that are actually negative.

$$\frac{TN}{FN + TN}$$

Overall Model Profit

The Overall Model Profit is the total made from all records for a given model. This is calculated using the values for MCC listed in the Setup Phase.

$$-\$4{,}754.58 * FN + \$94{,}661.53 * TN$$

Profit per Loan

Profit per Loan is the profit made per loan on average. This is calculated by taking the Overall Model Profit and dividing by the number of records.

$$\frac{-\$4{,}754.58 * FN + \$94{,}661.53 * TN}{79{,}233}$$

The first set of evaluations compared the naïve models to the MCC/Rebalanced models for each classification type. The second section compared all naïve models and then all the MCC/Rebalanced models. Starting with CART the evaluation measures are shown in Table 15.

Measure	**Naïve CART Model**	**CART Model with MCC**
Accuracy	0.8841	0.6613
Sensitivity	0.8816	0
Specificity	0.8853	1
Proportion of True Positives	0.7974	N/A
Proportion of True Negatives	0.9359	0.6613
Overall Model Profit	$4,376,243,076.04	$4,832,585,110.75
Profit per Loan	$55,232,58	$60,992.08

Table 15: Evaluation Measures for CART Models

Beginning with accuracy, the naïve model was much more accurate at 88.41% compared to 66.13% for the model with MCC. The naïve model also had a high sensitivity of 88.16% while the model with MCC has a value of 0, since there were no positive classifications. This also resulted in no value for the proportion of true positives (since there were no positives) and a low proportion of true negatives. While underperforming based on the previous measures, the model with MCC generated over $5,000 more per loan. This is due to the misclassification costs favoring negative responses, specifically TN responses, and is seen with the remaining models.

Next are the C5.0 models. The evaluation measures are shown in Table 16.

Measure	**Naïve C5.0 Model**	**C5.0 Model with MCC**
Accuracy	0.9269	0.7705
Sensitivity	0.9048	0.3343
Specificity	0.9382	0.9939
Proportion of True Positives	0.8822	0.9656
Proportion of True Negatives	0.9506	0.7446
Overall Model Profit	$4,641,318,201.37	$4,844,946,758.33
Profit per Loan	$58,579.10	$61,148.09

Table 16: Evaluation Measures for C5.0 Models

The naïve model had an accuracy of 92.69%, which was the highest accuracy of all models in this analysis. The model with MCC also had a good accuracy at 77.05%. Viewing sensitivity, the model with MCC had a low value of 33.43% compared to the naïve model at 90.48%. For the remaining measures, excluding profit, both models performed well. The naïve model had a slightly higher proportion of true negatives while the model with MCC had a slightly higher proportion of true positives, and both models had specificity values over 93%. The model with MCC generated higher profits, over $61,000, compared to the naïve model at $58,579.10.

The next set of models are the CHAID models. The evaluation measures are shown in Table 17.

Measure	**Naïve CHAID Model**	**CHAID Model with MCC**
Accuracy	0.8672	0.6733
Sensitivity	0.7724	0.0372
Specificity	0.9158	0.9991
Proportion of True Positives	0.8245	0.9542
Proportion of True Negatives	0.8871	0.6696
Overall Model Profit	$4,513,576,527.00	$4,832,791,182,73
Profit per Loan	$56,965.87	$60,994.68

Table 17: Evaluation Measures for CHAID Models

The naïve model had an accuracy of 86.72%, which was the lowest of all the naïve models up to this point, compared to 67.33% for the model with MCC. The naïve model also had a lower sensitivity compared to the previous naïve models at 77.24%, however it performed well in terms of specificity, proportion of true positives, and proportion of true negatives. The model with MCC performed well in terms of specificity and proportion of true positives, but not so well in terms of sensitivity and proportion of true negatives. Viewing the profits, the model with MCC generated over $3,000 more per loan compared to the naïve model.

QUEST is the next set of models that were evaluated. The evaluation measures are shown in Table 18.

Measure	**Naïve QUEST Model**	**QUEST Model with**
Accuracy	0.6613	0.6613
Sensitivity	0	0
Specificity	1	1
Proportion of True Positives	N/A	N/A
Proportion of True Negatives	0.6613	0.6613
Overall Model Profit	$4,832,585,110.75	$4,832,585,110.75
Profit per Loan	$60,992.08	$60,992.08

Table 18: Evaluation Measures for QUEST Models

As noted in the Modeling Phase, both models represent an All-Negative model and were equally as bad. Values for accuracy, sensitivity, proportion of true positives, and proportion of true negatives were all low due to the lack of any positive classifications. However, due to the misclassification costs, these models generated almost $61,000 per loan.

Next are the fifth set of models, Neural Networks. The evaluation measures are shown in Table 19.

Measure	**Naïve NN Model**	**Rebalanced NN Model**
Accuracy	0.8795	0.6837
Sensitivity	0.8080	0.0691
Specificity	0.9161	0.9985
Proportion of True Positives	0.8315	0.9582
Proportion of True Negatives	0.9031	0.6768
Overall Model Profit	$4,519,726,396.91	$4,833,737,272.72
Profit per Loan	$57,043.48	$61,006.62

Table 19: Evaluation Measures for NN Models

The naïve model performed very well, with values for all measures exceeding 80% (excluding profits), and an accuracy of 87.95%. The rebalanced model had an accuracy of 68.37% and performed poorly in sensitivity and proportion of true negatives but performed well in specificity and proportion of true positives, with both exceeding 95%. In terms of profits, the rebalanced model still generated almost $4,000 more per loan compared to the naïve model.

The last set of models are Random Forest models. The evaluation measures are shown in Table 20.

Measure	**Naïve RF Model**	**Rebalanced RF Model**
Accuracy	0.9067	0.8987
Sensitivity	0.8134	0.7662
Specificity	0.9544	0.9665
Proportion of True Positives	0.9014	0.9213
Proportion of True Negatives	0.9090	0.8898
Overall Model Profit	$4,710,316,349.35	$4,764,307,706.51
Profit per Loan	$59,448.92	$60,130.35

Table 20: Evaluation Measures for RF Models

Viewing these measures, both models performed very well. For the naïve model, the accuracy was over 90%, and all other measures (excluding profit) were over 81%. This was also one of the best performing naïve models based on profit, just shy of $60,000 per loan. The rebalanced model also performed very well, considering all other MCC/rebalanced models performed poorly in some areas. This model had an accuracy just shy of 90%, with all measures (excluding profit) over 88% with the exception of sensitivity at 76.62%. This model also generated over $60,000 per loan.

With all naïve and MCC/Rebalanced models compared for each method, all naïve and all MCC/Rebalanced models were compared against one another, starting with the naïve models shown in Table 21. The best performing measure between all models is listed in green, and the worst in red.

Measure	**CART**	**C5.0**	**CHAID**
Accuracy	0.8841	0.9269	0.8672
Sensitivity	0.8816	0.9048	0.7724
Specificity	0.8853	0.9382	0.9158
Proportion of True Positives	0.7974	0.8822	0.8245
Proportion of True Negatives	0.9359	0.9506	0.8871
Overall Model Profit	$4,376,243,076.04	$4,641,318,201.37	$4,513,576,527.00
Profit per Loan	$55,232,58	$58,579.10	$56,965.87

Measure	**QUEST**	**NN**	**RF**
Accuracy	0.6613	0.8795	0.9067
Sensitivity	0	0.8080	0.8134
Specificity	1	0.9161	0.9544
Proportion of True Positives	N/A	0.8315	0.9014
Proportion of True Negatives	0.6613	0.9031	0.9090
Overall Model Profit	$4,832,585,110.75	$4,519,726,396.91	$4,710,316,349.35
Profit per Loan	$60,992.08	$57,043.48	$59,448.92

Table 21: Evaluation Measures for all Naïve Models

Beginning with accuracy, the QUEST model performed the worst by far. The accuracy was only 66.13%, which was again due to no positive classifications. All other models performed very well, with the C5.0 model having the highest accuracy of 92.69%, followed by RF at 90.67%.

QUEST also performed the worst based on sensitivity, and C5.0 performed best at 90.48%, followed by CART and RF. As expected, QUEST performed best based on specificity due to all the predictions being negative. If we excluded QUEST based on this reasoning, RF performs the best at 95.44%, however all other models had values over 91%.

Moving to proportion of true positives the RF model performed best at 90.14% and since QUEST had no positive records, this value is undefined. Excluding QUEST, CART performed the poorest at 79.74%, and the remaining models all had values over 82%. The C5.0 model performed best based on proportion of true negatives at 95.06%, followed by the CART model at 93.59%. For the QUEST model, this value was equal to its accuracy, 66.13%, which was the worst for all models.

Lastly viewing profit, the QUEST model generated $60,992.08 per loan, the only model generating over $60,000. This was due to the high profit gained from a TN response compared to the small lost for a FN misclassification. The RF model came in second at $59,448.92 per loan, and CART came in last at $55,232,58 per loan.

The same evaluation was performed using the rebalanced and MCC models. These are shown in Table 22. Again, the best performing measure between all models is listed in green and the worst in red.

Measure	CART	C5.0	CHAID
Accuracy	0.6613	0.7705	0.6733
Sensitivity	0	0.3343	0.0372
Specificity	1	0.9939	0.9991
Proportion of True Positives	N/A	0.9656	0.9542
Proportion of True Negatives	0.6613	0.7446	0.6696
Overall Model Profit	$4,832,585,110.75	$4,844,946,758.33	$4,832,791,182,73
Profit per Loan	$60,992.08	$61,148.09	$60,994.68

Measure	QUEST	NN	RF
Accuracy	0.6613	0.6837	0.8987
Sensitivity	0	0.0691	0.7662
Specificity	1	0.9985	0.9665
Proportion of True Positives	N/A	0.9582	0.9213
Proportion of True Negatives	0.6613	0.6768	0.8898
Overall Model Profit	$4,832,585,110.75	$4,833,737,272.72	$4,764,307,706.51
Profit per Loan	$60,992.08	$61,006.62	$60,130.35

Table 22: Evaluation Measures for all Rebalanced and MCC Models

The RF model had the best accuracy at 89.87%, with the next highest being 77.05% for the C5.0 model. The remaining models all had accuracies under 70%, with the CART and QUEST models performing worst at 66.13%, as these replicate an All-Negative model similar to the naïve QUEST model.

For sensitivity, the RF outperformed all other models by far, with a value of 76.62%. C5.0 was again number two at 33.43%, and the remaining models had values under 7%. CART and QUEST performed worst as the sensitivity is 0. However, these two models had specificity values of 1. RF was the worst performing at 96.65%, however between all models the specificities were within 4% of one another.

All the models performed well in terms of proportion of true positives. The C5.0 was highest at 96.56% and no models were below 92%, with the exception of CART and QUEST where the values were undefined. The RF model performed best in terms of proportion of true negatives at 88.98%. The next highest was C5.0 at 74.46%, and the worst was again CART and QUEST, equal to their accuracies of 66.13%.

Viewing the profits, all models generated between $60,000 and $62,000 per loan. The C5.0 model generated the most profit per loan at $61,148.09 while RF generated the least profit per loan, $60,130.35.

The last portion of the Evaluation Phase was identifying the most significant variables for predicting *Default*. Table 23 lists the top 5 predictors for each MCC/rebalanced model.

Rank	**CART**	**C5.0**	**CHAID**
1	Term	Term	Term
2	NewExist	DisbursementGross	BankState
3	UrbanRural	BankState	CreateJob?
4	BusinessSize	UrbanRural	RevLineCr
5	RetainedJob?	NewExist	RetainedJob?

Rank	**QUEST**	**NN**	**RF**
1	BankState	Term	Term
2	BusinessSize	BankState	ApprovalDate
3	RetainedJob?	State	DisbursementGross
4	NAICS	DisbursementGross	RevLineCr
5	UrbanRural	ApprovalDate	NewExist

Table 23: Predictor Importance for MCC/Rebalanced Models

The variable *Term* is at the top of the list for every model except for QUEST, which concludes the *Term* is the most important variable for predicting loan default. The next most prevalent variable is *BankState*, which appeared in 4 of the models, and was the most important predictor for the QUEST model. Variables *NewExist*, *UrbanRural*, *RetainedJob?*, and *DisbursementGross* follow, each appearing in 3 of the models. *State* is the only variable not listed in the top 5 for any model.

Phase 7: Deployment Phase

The Deployment Phase was the final phase, where businesses will typically take the best performing models and adapt them for use on real-world datasets. Another example of deployment would be a report explaining the results and the best models. For this analysis, the thesis is the final report which fulfills the deployment phase. While there is no application for this analysis, the same procedures and models can be applied to real world SBA loan datasets.

The same procedure can be used for non-SBA loans as well. The only difference would be the calculations used for misclassification costs. For *Amount_Lost*, one would simply omit the *SBA_Approval* portion (assuming no one is backing the loans). For a simpler approach, one could assume the entire amount is lost on a defaulted loan, so *Disbursement_Gross* would equal *Amount_Lost* for this case.

CONCLUSION

In this analysis, six types of models were for predicting loan default. For each type, a naïve model and a model with MCC was created, for a total of twelve models. The non-naïve models generated the most profits, even though important measures such as accuracy were generally poor. This was due to the MCC, which typically incur more of a loss for FN responses than they do a gain for FN responses. However, for this analysis, a TN response generated almost 20 times more profit than a loss from a FN response, so the MCC favored models with more TN responses, which were the All-Negative models. For this reason, the optimal model was not chosen solely based on profits, but instead chosen based on all performance measures. The C5.0 model with MCC was concluded to be the most optimal model considering all measures. Variables *Term* and *BankState* were concluded to be the most significant variables for predicting loan default.

LIMITATIONS

One of the limitations was not having the interest rate for each loan. While not included in the original dataset, the prime rates from 2005-2009 were used with the maximum interest rate calculations shown in Table 4 to calculate interest.

Payments were assumed to be made at the beginning of the month as opposed to the end of the month, which would change the amount. While there was no information found regarding this, calculations from several SBA rate calculators match the calculations made when payments were made at the end of the month. Any possible late fees issued by lenders were also not considered.

The interest rate format that is used for 7(a) loans was used for all records. While most SBA loans are 7(a) loans, other SBA loans use different interest rate calculations. The SBA National data did not specify the type of SBA loan, so all loans were assumed to be 7(a) loans, as these are the most common SBA loans.

While the term of the loan was given, there was no variables indicating the number of terms used to pay off the loan. While many businesses use the entire term of the loan to pay it back, many choose to make larger payments to pay off the loan early, which results in less interest paid over the life of the loan. While it is clear that some loans were paid off early, the amount of interest was calculated assuming the loan was paid off based on the term of the loan.

Lastly, Logistic Regression was not included as one of the models due to a quasi-complete separation in the data. While not included in this analysis, Logistic Regression is a very useful classification model for binary target variables.

REFERENCES

Alomari, Z. (2017). *Loan Default Prediction and Identification of Interesting Relations between Attributes of Peer-to-Peer Loan Applications*. ResearchGate. Retrieved from https://www.researchgate.net/publication/322603744_Loan_Default_Prediction_and_Identification_of_Interesting_Relations_between_Attributes_of_Peer-to-Peer_Loan_Applications

Amit, R., & Zott, C. (2015). *Crafting Business Architecture: The Antecedents of Business Model Design*. ResearchGate. Retrieved from https://www.researchgate.net/publication/275773150_Crafting_Business_Architecture_The_Antecedents_Of_Business_Model_Design

Brown, M. S. (2015). What IT Needs To Know About The Data Mining Process. Retrieved from https://www.forbes.com/sites/metabrown/2015/07/29/what-it-needs-to-know-about-the-data-mining-process/#15416d08515f

CUMIPMT function. (n.d.). Retrieved from https://support.microsoft.com/en-ie/office/cumipmt-function-61067bb0-9016-427d-b95b-1a752af0e606

Frankenfield, J. (2020, August 28). Artificial Neural Network (ANN). Retrieved from https://www.investopedia.com/terms/a/artificial-neural-networks-ann.asp

Hathaway, I., & Litan, R. (2016, July 28). Declining Business Dynamism in the United States: A Look at States and Metros. Retrieved August 13, 2020, from https://www.brookings.edu/research/declining-business-dynamism-in-the-united-states-a-look-at-states-and-metros/?utm_source=link_newsv9

Historical Prime Rate. (2020). Retrieved from https://www.jpmorganchase.com/corporate/About-JPMC/historical-prime-rate.htm

Instructions for 1502 Lender Reporting. (n.d.). Retrieved from https://www.sba.gov/node/6364

Introduction to NAICS. (2017). Retrieved from https://www.census.gov/eos/www/naics/

Kassambara, A. (2018, March 11). Multicollinearity Essentials and VIF in R. Retrieved from http://www.sthda.com/english/articles/39-regression-model-diagnostics/160-multicollinearity-essentials-and-vif-in-r/

Larose, C. D., & Larose, D. T. (2019). *Data Science using Python and R*. Hoboken: Wiley.

Larose, D. T. (2018). DATA 511 Notes.

Larose, D. T., & Larose, C. D. (2015). *Data Mining and Predictive Analytics*. Wiley.

Li, M., Mickel, A., & Taylor, S. (2018). *"Should This Loan be Approved or Denied?": A Large Dataset with Class Assignment Guidelines*. American Statistical Association. Retrieved from https://amstat.tandfonline.com/doi/full/10.1080/10691898.2018.1434342#.Xp7etchKjs3

Links to charts and tables for a firm size class. (2020, July 29). Retrieved from https://www.bls.gov/bdm/bdmfirmsize.htm

LowDoc. (n.d.). Retrieved from https://www.entrepreneur.com/encyclopedia/lowdoc

Min, L., Mickel, A., & Taylor, S. (2018). Loan Documentation _National SBA.

NAICS & SIC Identification Tools. (2018). Retrieved from https://www.naics.com/search/

NAICS Sector 62: Health Care and Social Assistance. (2018). Retrieved from https://classcodes.com/lookup/sector-62/

Nicastro, S. (2017, October 3). SBA Loan Default: What to Know If You Can't Pay. Retrieved from https://www.nerdwallet.com/blog/small-business/sba-loan-default-know-cant-pay/

Nigro, P., & Glennon, D. (2005). *An Analysis of Sba Loan Defaults by Maturity Structure*. ResearchGate. Retrieved from https://www.researchgate.net/publication/5150889_An_Analysis_of_SBA_Loan_Defaults_by_Maturity_Structure

Onion, A., Sullivan, M., & Mullen, M. (2017). Great Recession. Retrieved from https://www.history.com/topics/21st-century/recession

Porter, K., Brozic, J., Chorpenning, A., Cothern, L., & Malm, L. (2020). What is a charged-off account? Retrieved from https://www.creditkarma.com/advice/i/what-is-a-charge-off/

Postins, M. (2017, November 21). Prime Rate Change History. Retrieved from https://homeguides.sfgate.com/prime-rate-change-history-3222.html

Prime Rate: Federal Funds Rates Discount Rate Fed Fund Reserve Lending COFI. (2020). Retrieved from https://www.bankrate.com/rates/interest-rates/prime-rate.aspx

Quick Overview of SBA Loan Guaranty Programs. (2017). Retrieved from https://www.sba.gov/sites/default/files/articles/Loan_Chart_Jan_2018_Version_A.pdf

Sarma, K. S. (2017). *Predictive Modeling with SAS Enterprise Miner: Practical Solutions for Business Applications*. Cary, NC: SAS Institute Inc.

SBA Franchise Directory. (n.d.). Retrieved from https://www.sba.gov/sba-franchise-directory

SBA: Organization. (n.d.). Retrieved from https://www.sba.gov/about-sba/organization

Schneider, K., Stocks, C., & Dietrich, J. (2020). Data Point: Small Business Lending and the Great Recession . Retrieved from https://files.consumerfinance.gov/f/documents/cfpb_data-point_small-business-lending-great-recession.pdf

Showmethebell. (2017, August 24). Split of Train and Test Data. Retrieved from https://dataandbeyond.wordpress.com/2017/08/24/split-of-train-and-test-data/

Tariq Aziz, H. I., Sohail, A., Aslam, U., & Batcha, N. K. (2019). *Loan Default Prediction Model Using Sample, Explore, Modify, Model, and Assess (SEMMA)*. ResearchGate. Retrieved from

https://www.researchgate.net/publication/335966813_Loan_Default_Prediction_Model_Using_Sample_Explore_Modify_Model_and_Assess_SEMMA

Terms, conditions, and eligibility. (n.d.). Retrieved from https://www.sba.gov/partners/lenders/7a-loan-program/terms-conditions-eligibility

Tuovila, A. (2020). Fiscal Year (FY) Definition. Retrieved from https://www.investopedia.com/terms/f/fiscalyear.asp

Types of 7(a) loans. (n.d.). Retrieved from https://www.sba.gov/partners/lenders/7a-loan-program/types-7a-loans

Wall Street Prime Rate: WSJ Current Prime Rate Index. (2020). Retrieved from https://www.bankrate.com/rates/interest-rates/wall-street-prime-rate.aspx

APPENDIX

Figure 1: Filler node for reclassifying blanks in BankState

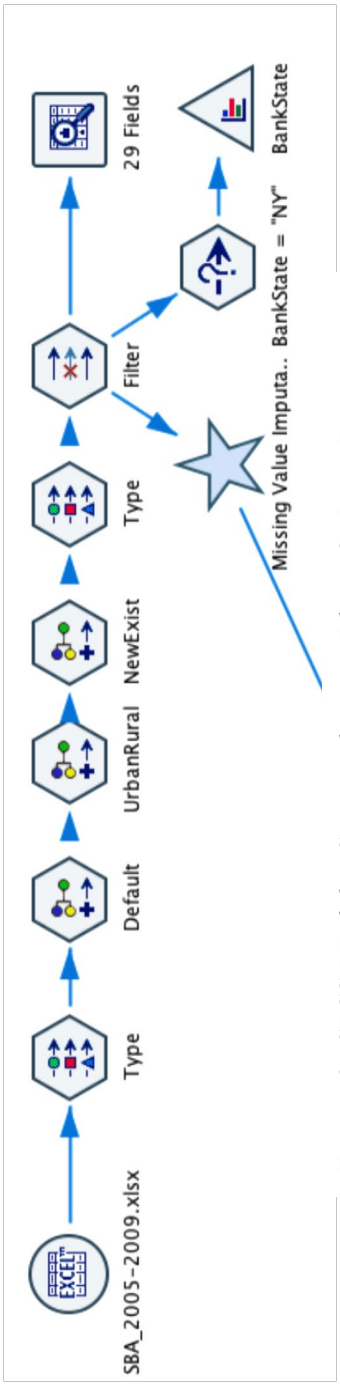


Figure 2: SPSS Modeler Stream showing Phase 2: Data Preparation

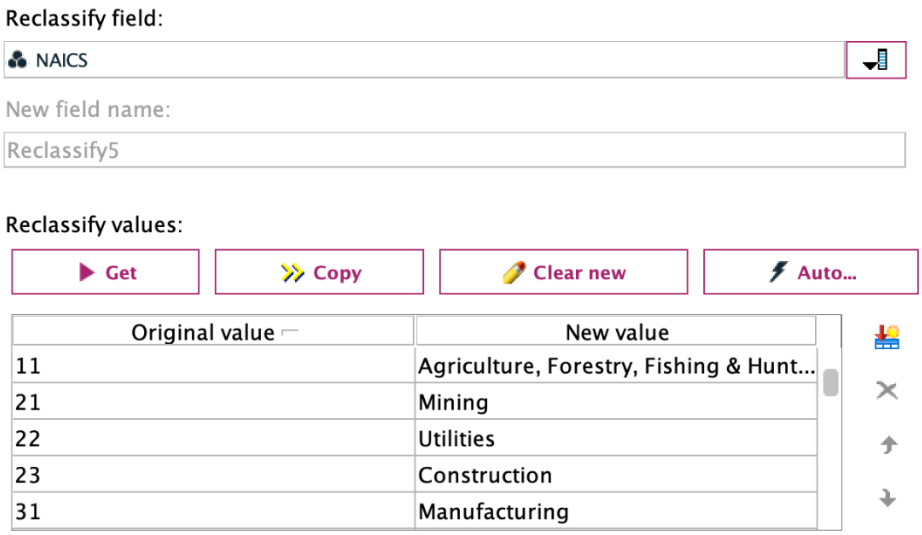


Figure 3: Reclassify Node for NAICS

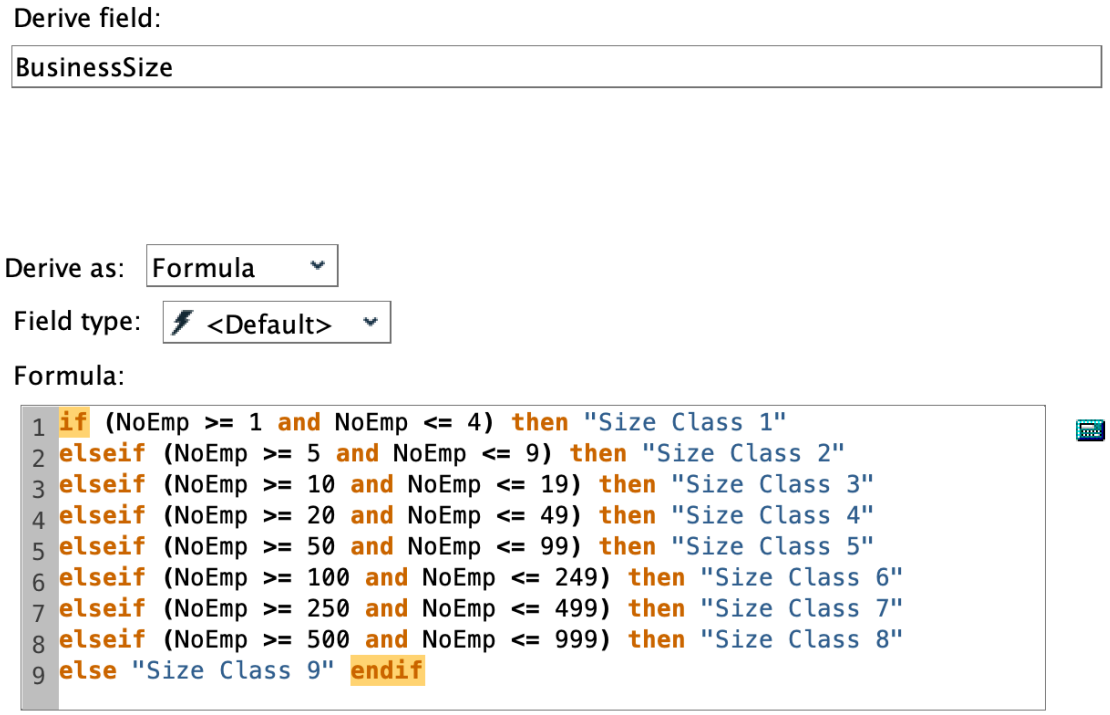


Figure 4: Derive node for BusinessSize

Derive field:

CreateJob?

Derive as: Formula

Field type: Flag

Formula:

```
1 if (CreateJob > 0 ) then "Y" else "N" endif
```

Figure 5: Derive Node for CreateJob?

Derive field:

RetainedJob?

Derive as: Formula

Field type: Categorical

Formula:

```
1 if (RetainedJob = 0 ) then "No"
2 elseif (RetainedJob = 1) then "One"
3 else "Multiple" endif
```

Figure 6: Derive Node for RetainedJob?

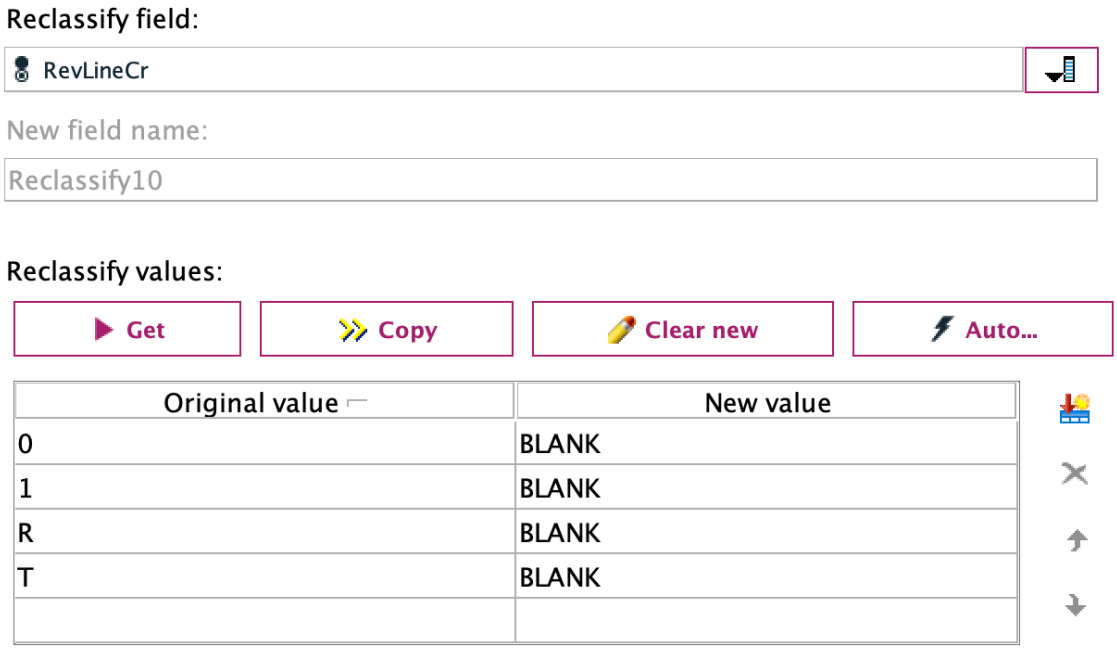


Figure 7: Reclassify Node for RevLineCr

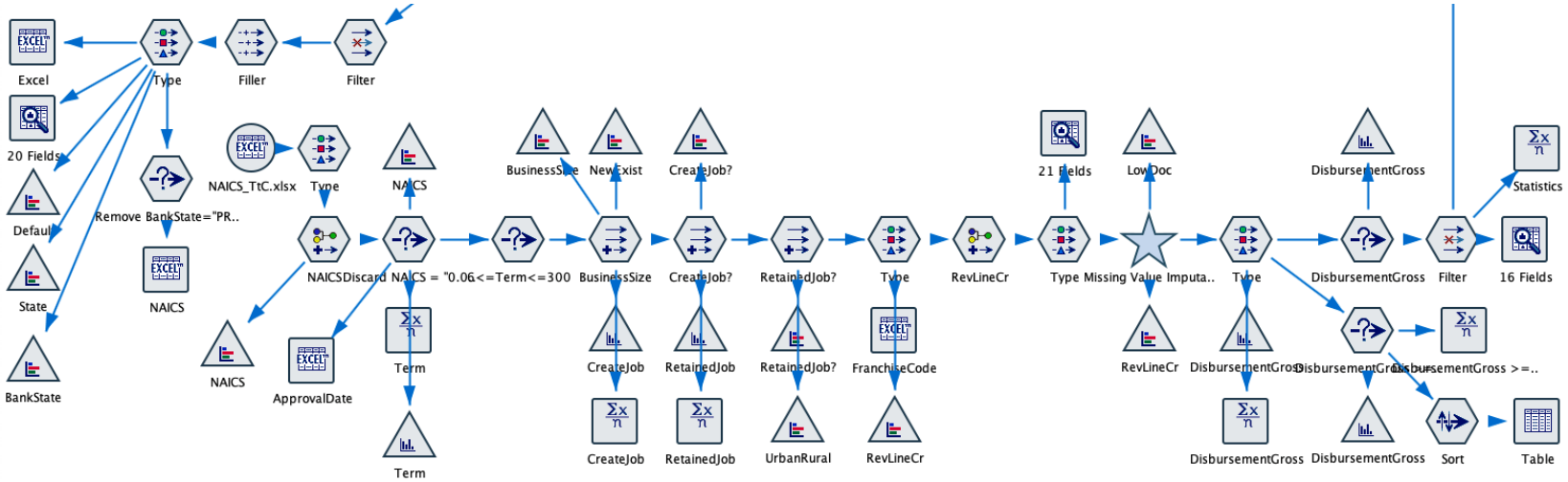


Figure 8: SPSS Modeler Stream showing Phase 3: Exploratory Data Analysis (EDA)

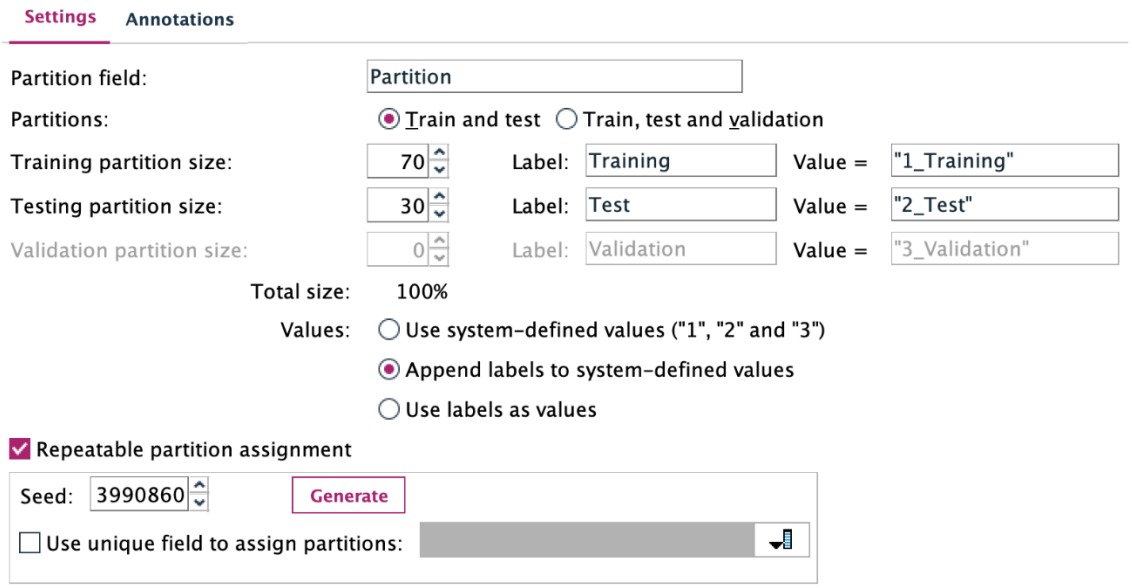


Figure 9: Output from Partition Node

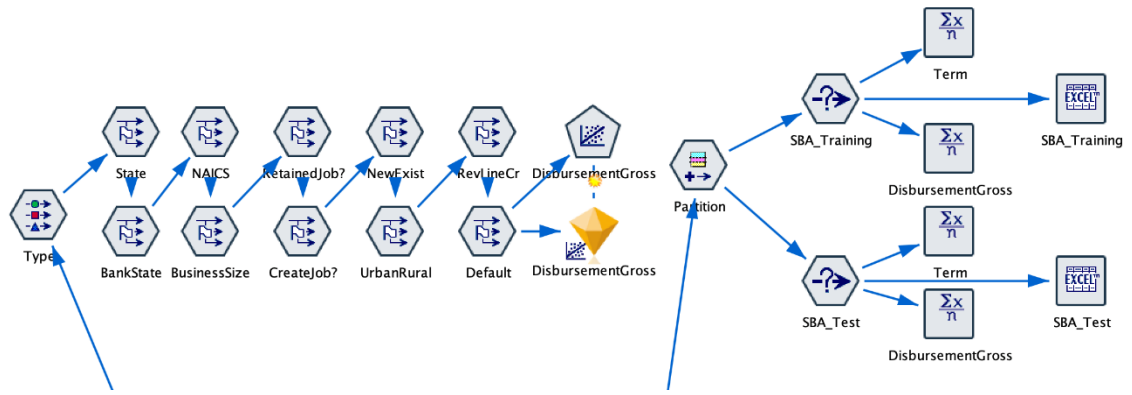


Figure 10: SPSS Modeler Stream showing Phase 4: Setup

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	Collinearity Statistics	
		B	Std. Error	Beta			Tolerance	VIF
1	(Constant)	-110554.04	2455.440		-45.024	.000		
	Term	2403.908	9.684	.474	248.240	.000	.649	1.541
	State_AK	-20276.277	14168.157	-.003	-1.431	.152	.711	1.406
	State_AL	-31301.122	5545.051	-.010	-5.645	.000	.833	1.200
	State_AR	-53391.813	8180.757	-.013	-6.527	.000	.589	1.698
	State_AZ	-2835.008	3470.900	-.001	-.817	.414	.760	1.316
	State_CO	-13615.606	3224.523	-.007	-4.223	.000	.792	1.263
	State_CT	-47941.492	4710.451	-.018	-10.178	.000	.741	1.350
	State_DC	-26116.714	9815.801	-.004	-2.661	.008	.971	1.030
	State_DE	-18695.653	9056.281	-.003	-2.064	.039	.951	1.052
	State_FL	-4374.642	2394.761	-.003	-1.827	.068	.650	1.538
	State_GA	-56.530	3187.430	.000	-.018	.986	.739	1.353
	State_HI	-74699.146	12352.909	-.017	-6.047	.000	.299	3.341
	State_IA	-20638.494	8005.743	-.007	-2.578	.010	.355	2.816
	State_ID	-28037.726	5511.579	-.010	-5.087	.000	.581	1.721
	State_IL	-11502.712	2639.670	-.008	-4.358	.000	.684	1.462
	State_IN	-25514.343	3941.308	-.013	-6.474	.000	.622	1.608
	State_KS	-52295.906	6758.951	-.017	-7.737	.000	.479	2.086
	State_KY	-34788.788	5797.312	-.011	-6.001	.000	.652	1.534
	State_LA	-62259.832	5510.467	-.021	-11.298	.000	.661	1.513
	State_MA	-18900.588	3851.847	-.010	-4.907	.000	.541	1.849
	State_MD	-14860.575	4064.892	-.006	-3.656	.000	.780	1.283
	State_ME	-62512.493	8529.299	-.017	-7.329	.000	.457	2.189
	State_MI	-12696.256	3204.099	-.008	-3.963	.000	.624	1.602
	State_MN	-24178.618	4057.142	-.014	-5.960	.000	.455	2.198
	State_MO	-47927.735	4259.616	-.023	-11.252	.000	.562	1.779
	State_MS	-68274.336	6766.539	-.020	-10.090	.000	.580	1.724
	State_MT	-20666.260	11500.717	-.005	-1.797	.072	.270	3.700
	State_NC	-11345.914	3597.423	-.005	-3.154	.002	.863	1.159
	State_ND	-14380.209	13488.953	-.003	-1.066	.286	.269	3.717
	State_NE	-36931.867	8989.456	-.010	-4.108	.000	.400	2.497
	State_NH	-17253.425	5824.555	-.006	-2.962	.003	.628	1.592
	State_NJ	-14058.061	3201.318	-.008	-4.391	.000	.682	1.467
	State_NM	-25207.395	8254.069	-.006	-3.054	.002	.618	1.619
	State_NV	-12861.994	4890.070	-.004	-2.630	.009	.853	1.172
	State_NY	-32857.108	2277.495	-.031	-14.427	.000	.525	1.904
	State_OH	-22952.787	2755.861	-.016	-8.329	.000	.611	1.637
	State_OK	-47227.417	6844.802	-.016	-6.900	.000	.457	2.187
	State_OR	-12309.052	4218.163	-.005	-2.918	.004	.836	1.196
	State_PA	-32458.953	3219.708	-.021	-10.081	.000	.535	1.868
	State_RI	5784.429	5996.106	.002	.965	.335	.851	1.175
	State_SC	-28016.863	5893.633	-.008	-4.754	.000	.900	1.111
	State_SD	-55747.344	8493.057	-.010	-6.564	.000	.939	1.065
	State_TN	-61854.863	4538.634	-.023	-13.629	.000	.822	1.217
	State_TX	-32025.201	2341.858	-.029	-13.675	.000	.540	1.851
	State_UT	-34995.018	4677.709	-.020	-7.481	.000	.315	3.175
	State_VA	5559.862	3733.157	.002	1.489	.136	.867	1.154
	State_VT	-37669.040	8949.687	-.008	-4.209	.000	.595	1.681
	State_WA	-8185.756	3486.520	-.004	-2.348	.019	.657	1.521
	State_WI	-34775.431	4543.671	-.018	-7.654	.000	.427	2.339
	State_WV	-36486.255	9384.110	-.007	-3.888	.000	.783	1.278
	State_WY	-8450.588	15813.003	-.001	-.534	.593	.464	2.154
	BankState_AK	121901.426	26377.397	.008	4.621	.000	.717	1.395
	BankState_AL	6577.556	4739.646	.002	1.388	.165	.808	1.237
	BankState_AR	67809.621	8550.596	.017	7.930	.000	.529	1.892
	BankState_AZ	72342.965	8899.241	.013	8.129	.000	.868	1.152
	BankState_CO	54709.643	5734.581	.016	9.540	.000	.881	1.135
	BankState_CT	142129.915	5734.141	.044	24.787	.000	.756	1.322
	BankState_DC	165854.522	22326.802	.012	7.428	.000	.981	1.019
	BankState_DE	38256.095	2836.348	.027	13.488	.000	.595	1.680
	BankState_FL	-17894.226	4017.607	-.008	-4.454	.000	.825	1.212

BankState_GA	85154.587	4906.412	.030	17.356	.000	.814	1.229
BankState_HI	89725.347	14670.250	.017	6.116	.000	.301	3.318
BankState_IA	30187.001	9139.102	.009	3.303	.001	.356	2.812
BankState_ID	-10182.037	8464.763	-.002	-1.203	.229	.664	1.507
BankState_IL	2938.847	2073.685	.003	1.417	.156	.488	2.051
BankState_IN	43897.998	6191.615	.013	7.090	.000	.679	1.473
BankState_KS	56754.669	8594.719	.014	6.603	.000	.496	2.016
BankState_KY	14465.936	9495.416	.003	1.523	.128	.687	1.455
BankState_LA	74235.624	9424.247	.015	7.877	.000	.694	1.442
BankState_MA	35965.094	5877.878	.012	6.119	.000	.626	1.598
BankState_MD	79063.342	8005.821	.016	9.876	.000	.846	1.182
BankState_ME	38012.207	11433.625	.007	3.325	.001	.470	2.130
BankState_MI	33285.689	6520.944	.009	5.104	.000	.773	1.293
BankState_MN	53562.484	5212.037	.023	10.277	.000	.478	2.093
BankState_MO	48942.530	5214.167	.019	9.386	.000	.566	1.767
BankState_MS	73477.743	8479.485	.017	8.665	.000	.644	1.554
BankState_MT	41432.951	12875.864	.010	3.218	.001	.269	3.713
BankState_NC	-38061.946	1919.536	-.044	-19.829	.000	.485	2.064
BankState_ND	46274.025	13737.513	.010	3.368	.001	.268	3.731
BankState_NE	42002.074	10343.612	.010	4.061	.000	.401	2.491
BankState_NH	-30424.479	8674.998	-.007	-3.507	.000	.675	1.481
BankState_NJ	52843.276	5551.208	.016	9.519	.000	.803	1.246
BankState_NM	28716.143	10355.100	.005	2.773	.006	.620	1.614
BankState_NV	-4128.998	7591.315	-.001	-.544	.587	.883	1.133
BankState_NY	36388.859	2685.247	.027	13.551	.000	.588	1.701
BankState_OH	-12075.225	2285.820	-.012	-5.283	.000	.469	2.134
BankState_OK	60599.237	8868.538	.015	6.833	.000	.467	2.139
BankState_OR	-14089.096	4348.722	-.005	-3.240	.001	.861	1.162
BankState_PA	50748.018	4709.368	.020	10.776	.000	.661	1.513
BankState_RI	-31850.611	2687.889	-.026	-11.850	.000	.480	2.084
BankState_SC	106056.998	7276.963	.023	14.574	.000	.912	1.096
BankState_SD	26235.684	2429.186	.020	10.800	.000	.675	1.482
BankState_TN	102018.399	8293.012	.020	12.302	.000	.869	1.151
BankState_TX	115188.785	3493.575	.062	32.972	.000	.667	1.500
BankState_UT	2827.880	4861.796	.002	.582	.561	.298	3.356
BankState_VA	-40251.084	2317.755	-.033	-17.366	.000	.670	1.492
BankState_VT	17953.235	11510.183	.003	1.560	.119	.599	1.671
BankState_WA	48611.457	5976.096	.015	8.134	.000	.722	1.385
BankState_WI	43497.665	5342.741	.019	8.141	.000	.436	2.294
BankState_WV	17956.203	18806.412	.002	.955	.340	.803	1.246
BankState_WY	33565.721	20816.258	.004	1.612	.107	.473	2.116
NAICS_Accommodation and Food Services	-11775.815	1753.479	-.013	-6.716	.000	.657	1.522
NAICS_Administrative and Support and Waste Management and Remediation Services	-29324.268	2159.725	-.024	-13.578	.000	.788	1.269
NAICS_Agriculture, Forestry, Fishing & Hunting	54412.186	6081.637	.015	8.947	.000	.849	1.178
NAICS_Arts, Entertainment, and Recreation	-16464.248	3247.259	-.008	-5.070	.000	.908	1.101
NAICS_Construction	-4884.212	1722.183	-.005	-2.836	.005	.658	1.519

NAICS_Educational Services	-35893.390	4100.871	-.014	-8.753	.000	.942	1.062
NAICS_Finance and Insurance	-21467.323	3417.620	-.010	-6.281	.000	.912	1.096
NAICS_Health Care and Social Assistance	-6036.098	2056.998	-.005	-2.934	.003	.767	1.304
NAICS_Information	-5821.062	3435.444	-.003	-1.694	.090	.916	1.092
NAICS_Management of Companies and Enterprises	5703.864	29044.226	.000	.196	.844	.998	1.002
NAICS_Manufacturing	48813.782	1918.461	.046	25.444	.000	.723	1.383
NAICS_Mining	96840.288	9738.745	.015	9.944	.000	.974	1.027
NAICS_Other Services (except Public Administration)	-26088.492	1790.115	-.027	-14.574	.000	.710	1.408
NAICS_Professional, Scientific, and Technical Services	-2883.442	1733.570	-.003	-1.663	.096	.678	1.475
NAICS_Public Administration	-34571.945	27308.454	-.002	-1.266	.206	.998	1.002
NAICS_Real Estate Rental and Leasing	-8093.201	2942.214	-.004	-2.751	.006	.885	1.130
NAICS_Transportation and Warehousing	-13418.030	2341.364	-.010	-5.731	.000	.815	1.227
NAICS_Utilities	16308.182	16226.493	.002	1.005	.315	.996	1.004
NAICS_Wholesale Trade	65071.254	2105.470	.054	30.906	.000	.778	1.286
BusinessSize_Size Class 2	36460.665	1222.782	.050	29.818	.000	.838	1.193
BusinessSize_Size Class 3	106120.050	1541.780	.114	68.830	.000	.858	1.165
BusinessSize_Size Class 4	219590.594	1986.307	.181	110.552	.000	.885	1.130
BusinessSize_Size Class 5	365862.548	4107.624	.139	89.069	.000	.964	1.037
BusinessSize_Size Class 6	468884.223	7152.148	.101	65.559	.000	.986	1.014
BusinessSize_Size Class 7	680393.492	21738.395	.048	31.299	.000	.997	1.003
BusinessSize_Size Class 8	465131.718	42850.492	.017	10.855	.000	.999	1.001
BusinessSize_Size Class 9	2994.374	3563.683	.001	.840	.401	.883	1.133
RetainedJob?_Multiple	17817.443	1239.586	.029	14.374	.000	.575	1.740
RetainedJob?_No	48962.824	1770.697	.058	27.652	.000	.529	1.892
CreateJob?_Y	6128.196	932.032	.011	6.575	.000	.904	1.106
NewExist_E	5702.379	1010.852	.009	5.641	.000	.869	1.151
UrbanRural_R	-5649.476	1218.777	-.008	-4.635	.000	.842	1.187
RevLineCr_Y	13741.920	1042.074	.024	13.187	.000	.718	1.393
Default_1	52176.671	1052.137	.086	49.591	.000	.781	1.280

Figure 11: Multiple Regression Model Output

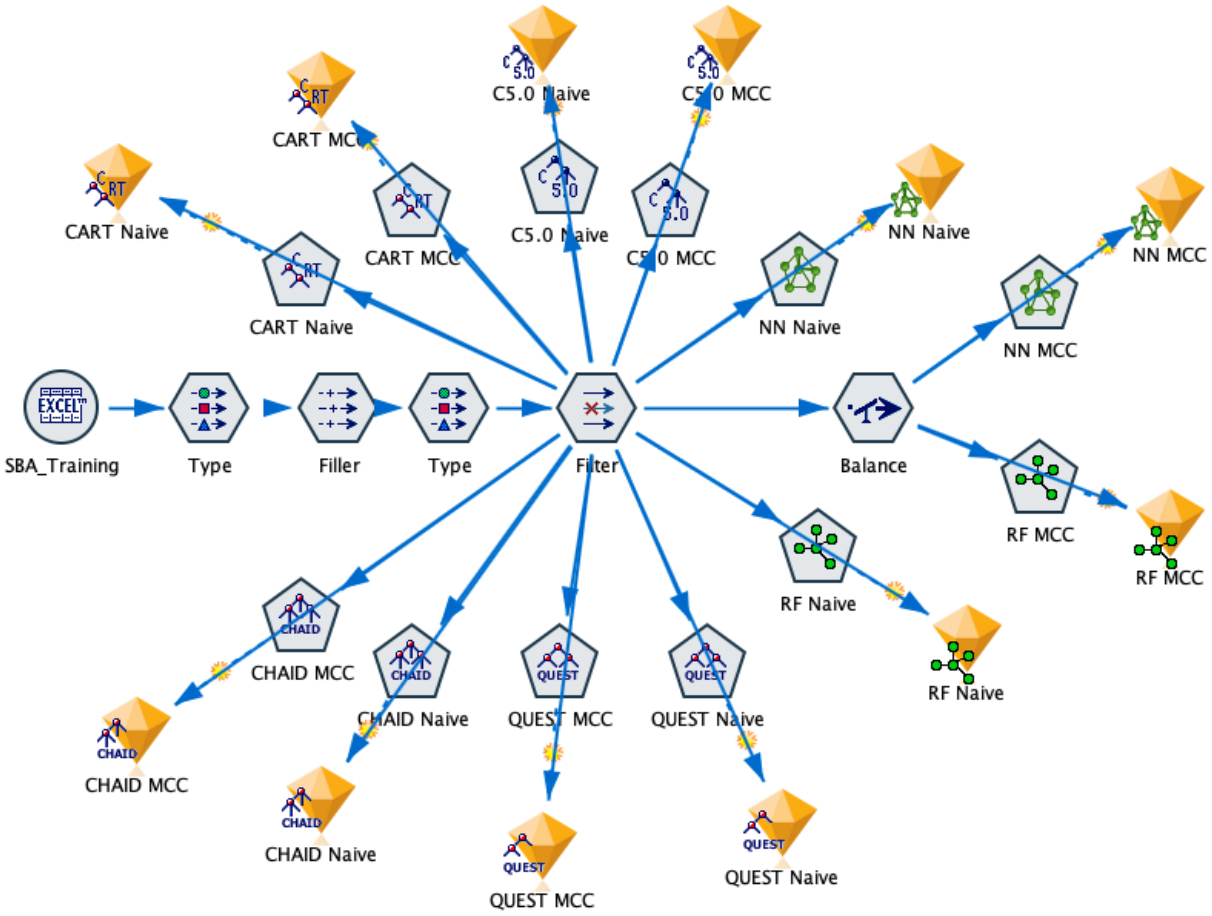


Figure 12: SPSS Modeler Stream showing Phase 5: Modeling

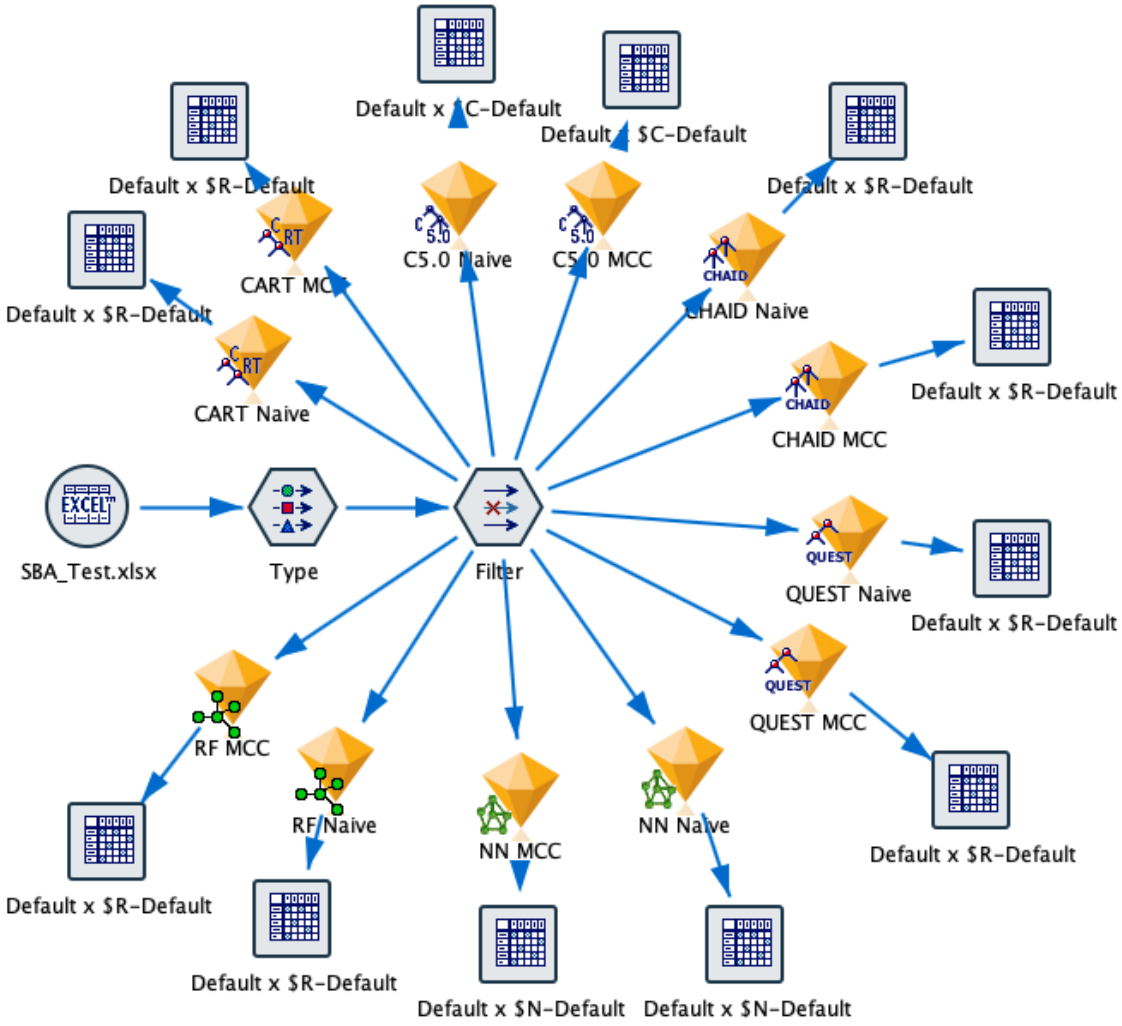


Figure 13: SPSS Modeler Stream showing Phase 6: Evaluation