Assignment 1

SVM Modeling on Universal Bank Dataset: Predicting Personal Loan Clientele

Theodore Fitch

Department of Data Analytics, University of Maryland Global Campus

DATA 640: Predictive Modeling

Dr. Steven Knode

January 30th, Spring 2024

Introduction:

The purpose of this analysis was to analyze the Universal Bank dataset and predict whether customers will accept the bank's personal loan based upon the other variables using a support vector machine (SVM). SVM models are classification prediction models that create a hyperplane in order to separate data (Srivastava, 2014). SVMs are quite powerful and are useful for small datasets (few values) and for datasets with many attributes. However, they can be a bit of a "black box", and they don't work as well for larger datasets as they're quite computationally expensive (all values must be converted to numeric inputs so categorical inputs are converted using dummy variables)(Ray, 2024). But, for the latter point, sampling of the dataset can overcome this barrier. SVMs have been used effectively to help predict a multitude of issues. Examples include credit card fraud (Şahin and Duman, 2011), customer churn (Kim et al., 2005), and bankruptcy (Erdogan, 2013). An SVM would be an ideal model to help predict whether a bank's customer will also accept their personal loan. This is incredibly useful information for any bank to know since they can use the demographic information to then predict : 1. who in their current client base they should be advertising their loans to and 2. which potential customers they should focus their marketing efforts on. Some variables seem easier to predict: for example, one can reasonably presume that customers with higher average total credit card spending; with higher debt; or with higher mortgages might also be more likely to have a personal loan with the bank. Each variable will need to be analyzed using the SVM since there are a plethora of factors which could affect whether someone will choose their bank's loan.

This project thus focused on analyzing the Universal Bank dataset in order to predict whether a customer uses the Universal Bank personal loan using the existing variables in an SVM model. The SAS SEMMA method will be used as a general approach towards this data.

This stands for Sample, Explore, Modify, Model, and Assess (Shafique and Qaiser, 2014). The steps are each somewhat self-explanatory: Sampling focuses on creating a sample dataset appropriately sized to model; Exploring focuses on identifying what pre-processing must take place, and possible underlying relationships or trends in the data; Modifying focuses on pre-processing the data. This includes selection or transformation of variables and changing rows (whether for missing values or looking for outliers); Modeling focuses on creating predictive models for the data; Assessing focuses on determining how each model perform (comparatively and as standalone models) and how useful they are.

Data Information and Exploratory Data Analysis:

This dataset was provided by the classroom for DATA 640 and was entitled "UniversalBank data.csv". It was uploaded to SAS Enterprise Miner as a .sas7bdat filetype (Figure 2). It contained 5,000 rows with 14 variables (Figure 1). As it was a relatively small dataset, it was deemed unnecessary to sample the dataset into a smaller subset (Ray, 2024). There were 7 interval variables (columns A-M, except for the nominal/ordinal attributes), 1 nominal (education), 1 ordinal (family), and 5 binary variables: (columns J-N). The first column was effectively ignored as it was an index. The target variable was column J, "Personal Loan" which was heavily imbalanced (480 entries of 5,000; rate of 9.6%)(Figure 3). Other variables are: age (years), working experience (years), income (in $1,000s), ZIP code, family size, average spending on credit cards per month (in $1,000s), education level (1: undergraduate, 2: graduate, 3: advanced/professional), mortgage (in $1,000s), and what services of the bank the customer uses including securities account, certificate of deposit (CD), online banking, and a credit card (all expressed as binary variables)(Figure 3). There are no missing entries in the whole dataset Figures 4,5). All ZIP codes belonged to California (Figure 6). One value (row 386) in column E

(ZIP Code) was entered incorrectly as it only contained 4 characters (Figure 5). There were 52 negative values found for column C (Experience) as well as 60 values of "0". No significant outliers were observed for the numeric variables (age, income, mortgage, etc.) and thus significant skew was not expected to be an issue for this dataset. When creating graphs to explore the Chi-Square and Worth of each variable, Income, CCAvg, CD_Account, Mortgage, Education, and Family were found to be the 6 most important variables with Chi-Squared values ranging from 1,411-30 (Figure 7,8). The next 6 variables combined Chi-Squared values were <35 showing the significance of the aforementioned variables.

Preprocessing:

As mentioned previously, no missing values existed in this dataset thus no imputations were necessary. The ZIP codes were transformed from their original 5-digit state to only the first 2 digits. This was thought helpful since the first 2 digits of a ZIP code indicate a general location in a state. In this case, all ZIP codes belonged to California locations: 90, 91, and 94 being city dense locations and 92, 93, 95, and 96 being rural (Polly, 2014). While ZIP codes can have significant demographic variation (especially within cities), it was thought that rural locations and city-dense locations would be the best way to categorize the ZIP codes. This was performed using a Transform Variables node. Then, since 1 value had only 4 digits, a correction was needed using a Replacement node. The first Transform Variables node was also used to take the absolute value of all negative values in the Experience variable (as these were deemed to be misinputs). No feature engineering was deemed helpful for this dataset. A Drop node was then used to clear unnecessary variables from the dataset (several were created during the usage of the Transform Variables node, and the ID was also dropped as this was simply an index (Figure 9). The

cleaned/transformed data was then separated into training and validation datasets using 2 separate Data Partition nodes at rates of 55% and 45% and 80% and 20% respectively (Figure 2).

Models and Methods:

The processed dataset was then run in 12 total models (Table 1). Models 1-5 were created using the default parameters for each possible method/kernel. SVMs have 2 major types of methods for creating and applying kernels: Active Set methods (AS) and Interior Point (IP) methods. The kernels created in SAS Enterprise Miner are linear (IP), polynomial (IP and AS), sigmoid (AS), and radial base function (RBF, a circular pattern)(AS). After running Models 1-5, it was observed polynomial and RBF kernels performed best. Thus, Models 6-9 were created by adjusting the parameters up by 1 from the default (and for Model 9, it was adjusted to 5.0). Comparing Models 1-9 using the Model Comparison node, showed Models 5, 6, & 8 performed best. So, these 3 models were copied and connected to the 80:20 Data Partition node to form Models 10-12. It is critical to note that a cost of c = 1 was applied to every model. This helped to reduce the impact of the heavy imbalance of the target variable where the model was penalized for each false negative it makes. This was necessary so that the predictive models don't overestimate the target variable as being false (since the event is rare) and end up with a higher number of false negatives (Cao et al., 2013). However, there is a tradeoff. Having too high of a cost value can result in overfitting to the model and less generalizability. The heavy imbalance of the target variable could also have been performed by sampling a portion of the dataset such that the sample had less imbalance in the target variable (closer to 50:50). This wasn't done because the dataset was already quite small.

Results and Model Evaluation:

Of the twelve models created, only 2 models had accuracy rates (on the validation dataset) of <93% (Table 1). Considering the target variable was heavily imbalanced at 480/5,000 entries (9.6%), any model with >90.4% accuracy was better than random choice (true random of binary targets is 50% but the data shows the target variable was imbalanced). The sigmoid model (4) performed worst at only 83.4% accuracy. The best models were polynomial models (Models 2, 5, and 10) with the best accuracy at 98.3% (Model 10) on the validation data. Importantly, all of the 80:20 split predictive models' accuracy on the validation dataset was >97%. This demonstrates no overfitting was occurring since the models still performed excellently even with a small validation dataset. The misclassification rate was simply 1-Accuracy but was added for clarity. The Cumulative Lift value shows how far the model "lifted" accuracy from true random (value at random = 1); comparable results can be observed that correlated well with Accuracy where sigmoid performed worst. However, the highest Cumulative Lift was observed for Model 2 at 9.18. Lastly, ROC Index (or Area Under the Curve, AUC), was high at >97% for most models (except sigmoid). All ROC charts showed the models performed very well, especially on the validation data (Figures 10,11). The cost function was adjusted multiple times for multiple models, but no major differences were observed in resultant accuracy. Specifically, cost was increased for Model 10 from 1-10-25-100 but no significant changes in accuracy were observed. To ensure there weren't underlying issues with this, the false negative (FN), false positive (FP), true positive (TP), and true negative (TN) rates of the classification/confusion matrix of Model 10 were observed (Maria Navin & Pankaja, 2016)(Figure 13).

- TN/(TN+FP) = 896/(896+9) = 99% **specificity**
- FN/(FN+TP) = 8/(8+89) = 8% **false negative rate**
- TP/(TP+FN) = 89/89+8) = 92% **sensitivity**

Thus, the cost factor appears to have worked as the best model appears to still have a significantly high specificity and sensitivity rate with a relatively low false negative rate. A

cutoff of 0.1 was also applied to Model 10 to determine if a different cutoff would be appropriate. The node outlined that 0.5 was the appropriate cutoff and this is observed in Figure 11 where the model's accuracy starts to react its peak while the true positive rate begins to nosedive (Figure 14). Lastly, the Mean Squared Error was relatively low for all models meaning there was low deviation in the models' predictions from actual outcomes (Figure 12).

Index	Data Partition (train: val)	Diagram Title	Optimization Method	Method Parameter /Degree	Accuracy (validation)	Misclassification Rate (validation)	ROC Index (validation)	Cumulative Lift
1	55:45	HPSVM 1) linear	Interior Point	N/A	0.9645	0.0355	0.97	8.174
2	55:45	HPSVM 2) IP polynomial	Interior Point	2	0.9818	0.0182	0.99	9.184
3	55:45	HPSVM 3) RBF	Active Set	1.0	0.9689	0.0311	0.98	8.771
4	55:45	HPSVM 4) sigmoid	Active Set	-1.0 to 1.0	0.8344	0.1656	0.61	1.332
5	55:45	HPSVM 5) AS polynomial	Active Set	2	0.9818	0.0182	0.99	9.184
6	55:45	HPSVM 6) IP polynomial 3	Interior Point	3	0.9782	0.0218	0.98	8.954
7	55:45	HPSVM 7) AS polynomial 3	Active Set	3	0.9782	0.0218	0.98	8.954
8	55:45	HPSVM 8) RBF	Active Set	2.0	0.9716	0.0284	0.98	9.046
9	55:45	HPSVM 9) RBF 5	Active Set	5.0	0.9294	0.0706	0.97	8.082
10	80:20	HPSVM 5) AS polynomial	Active Set	2	0.9830	0.0170	0.99	9.103
11	80:20	HPSVM 6) IP polynomial 3	Interior Point	3	0.9760	0.0240	0.99	8.898
12	80:20	HPSVM 8) RBF	Active Set	2.0	0.9780	0.0220	0.98	9.000

Table 1. Overview of created predictive models. Validation Accuracy, Misclassification Rate, and Cumulative Lift were reported to 4 significant figures, while ROC Index was reported to 2 figures. RBF model stands for Radial Base Function. Model 10 highlighted in red performed best.

Conclusion, Limitations, and Improvements:

A total of 12 predictive SVM models were generated in order to predict which customers were most likely to accept a personal loan from the bank. Of those, 11 models performed better than random and 3 performed at >98% accuracy. Polynomial models in general were the best, with Model 10 being the best (a polynomial kernel degree 2). This model should be used for future campaigns to predict which current customers are most likely to accept the personal loan the bank wants to offer. As previously mentioned, the variables with highest worth/Chi-Square

values were Income, how much a customer spends monthly on their credit cards, if they have a CD account with the bank, mortgage, education, and family. These values must continue being gathered for this dataset while the other variables had far less value towards these models.

There are multiple limitations to this study which should be mentioned. Most evidently, the dataset is limited. There are certainly other variables a bank would have on its customers which would be helpful in predicting whether they may need a personal loan. While mortgage and monthly credit card spend were used, debt at the bank, average total saved, average total spent, and monthly save/spend ration would prove invaluable for a study like this. The bank should consider gathering these variables as they would be relatively inexpensive to gather since those amounts should all be calculable internally without need for a survey. Other variables which would require a survey (and thus would be more expensive to obtain but would likely be helpful) would be total debt across all accounts (including not at the bank), car loan amount, and total debt to total savings ratio. More work should also be done to determine exactly how each variable affected the outcome. For example, it would be vital to understand how income (with the highest worth value) correlates with the target variable which would be done using a logistic regression model. Personal loans should be targeted at those with higher incomes (who are more likely to be able to afford them and pay them back without defaulting) and a model should be used to find the ideal income cutoff.

Another improvement which should be made to this study would be to singularly optimize one model further until it reached a 99% threshold. This would be critical before implementation as each percentage point represents more customers accepting the loan and further revenue for the bank. Model 10 should be adjusted with other factors: increasing polynomial degree, adding a penalty factor, or adjusting the data partition ratio further. Since this

dataset was relatively small, Model 10 should also be evaluated using other datasets to ensure further accuracy of the model. In conclusion, the combination of fine-tuning Model 10 with a better dataset using the new, more informative variables mentioned above would create a model with better predictive accuracy resulting in more customers for the bank for personal loans.

References:

Cao, P., Zhao, D., & Zaiane, O. (2013). An optimized cost-sensitive SVM for imbalanced data learning. In *Pacific-Asia conference on knowledge discovery and data mining* (pp. 280-292). Berlin, Heidelberg: Springer Berlin Heidelberg.

Erdogan, B. E. (2013). Prediction of bankruptcy using support vector machines: an application to bank bankruptcy. *Journal of Statistical Computation and Simulation*, *83*(8), 1543-1555.

Kim, S., Shin, K. S., & Park, K. (2005). An application of support vector machines for customer churn analysis: Credit card case. In *International Conference on Natural Computation* (pp. 636-647). Berlin, Heidelberg: Springer Berlin Heidelberg.

Knode, S. (2024). universal bank description.docx. University of Maryland Global Campus DATA 640 Learning Portal. Retrieved January 17th, 2024.

Maria Navin, J. R., & Pankaja, R. (2016). Performance analysis of text classification algorithms using confusion matrix. *International Journal of Engineering and Technical Research (IJETR)*, *6*(4), 75-8.

Polly, G. (2014). *The US grouped by first two zip code digits*. Imgur. https://imgur.com/NJGcg6v

Ray, S. (2024). *Learn how to use support vector machines (SVM) for Data Science*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/

Şahin, Y. G., & Duman, E. (2011). Detecting credit card fraud by decision trees and support vector machines.

Shafique, U., & Qaiser, H. (2014). A comparative study of data mining process models (KDD, CRISP-DM and SEMMA). *International Journal of Innovation and Scientific Research*, *12*(1), 217-222.

Srivastava, T. (2022). *Support Vector Machine - simplified*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2014/10/support-vector-machine-simplified/

Appendix:

Data Description:

ID	Customer ID
Age	Customer's age in completed years
Experience	#years of professional experience
Income	Annual income of the customer ($000)
ZIPCode	Home Address ZIP code.
Family	Family size of the customer
CCAvg	Avg. spending on credit cards per month ($000)
Education	Education Level. 1: Undergrad; 2: Graduate; 3: Advanced/Professional
Mortgage	Value of house mortgage if any. ($000)
Personal Loan	Did this customer accept the personal loan offered in the last campaign?
Securities Account	Does the customer have a securities account with the bank?
CD Account	Does the customer have a certificate of deposit (CD) account with the bank?
Online	Does the customer use internet banking facilities?
CreditCard	Does the customer use a credit card issued by UniversalBank?

Figure 1. Table showing the dataset variable descriptions (Knode, 2024).

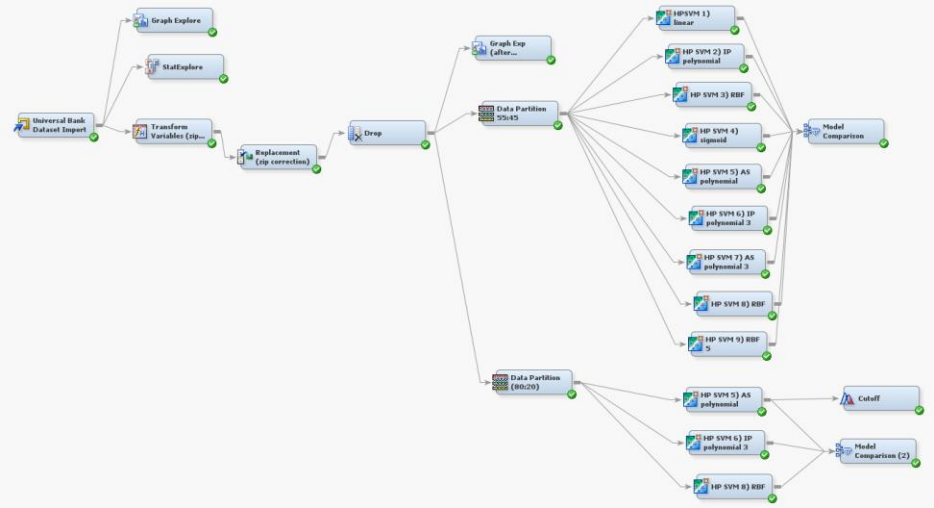
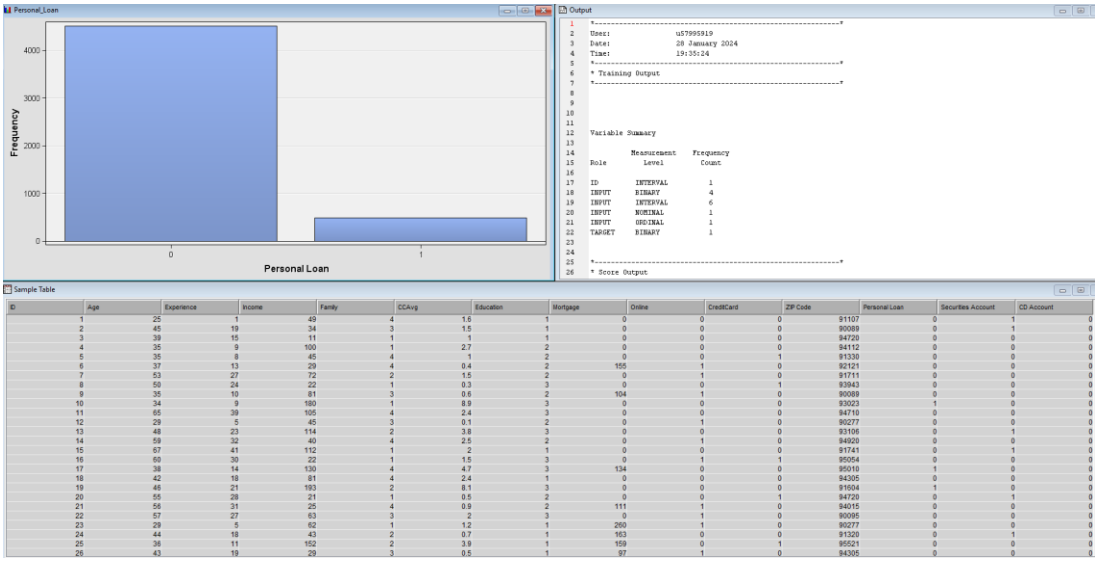


Figure 2. SAS Enterprise Miner Diagram of SVM Modeling.



ID	Age	Experience	Income	Family	CCAvg	Education	Mortgage	Online	CreditCard	ZIP Code	Personal Loan	Securities Account	CD Account
1	25	1	49	4	1.6	1	0	0	0	91107	0	1	0
2	45	19	34	3	1.5	1	0	0	0	90089	0	1	0
3	39	15	11	1	1	1	0	0	0	94720	0	0	0
4	35	9	100	1	2.7	2	0	0	0	94112	0	0	0
5	35	8	45	4	1	2	0	0	1	91330	0	0	0
6	37	13	29	4	0.4	2	155	1	0	92121	0	0	0
7	53	27	72	2	1.5	2	0	1	0	91711	0	0	0
8	50	24	22	1	0.3	3	0	0	1	93943	0	0	0
9	35	10	81	3	0.6	2	104	1	0	90089	0	0	0
10	34	9	180	1	8.9	3	0	0	0	93023	1	0	0
11	65	39	105	4	2.4	3	0	0	0	94710	0	0	0
12	29	5	45	3	0.1	2	0	1	0	90277	0	0	0
13	48	23	114	2	3.8	3	0	0	0	93106	0	1	0
14	59	32	40	4	2.5	2	0	1	0	94928	0	0	0
15	67	41	112	1	2	1	0	0	0	91741	0	1	0
16	60	30	22	1	1.5	3	0	1	1	95054	0	0	0
17	38	14	130	4	4.7	3	134	0	0	95010	1	0	0
18	42	18	81	4	2.4	1	0	0	0	94305	0	0	0
19	46	21	193	2	8.1	3	0	0	0	91604	1	0	0
20	55	28	21	1	0.5	2	0	0	1	94720	0	1	0
21	56	31	25	4	0.9	2	111	1	0	94015	0	0	0
22	57	27	63	3	2	3	0	1	0	90095	0	0	0
23	29	5	62	1	1.2	1	260	1	0	90277	0	0	0
24	44	18	43	2	0.7	1	163	0	0	91320	0	1	0
25	36	11	152	2	3.9	1	159	0	1	95521	0	0	0
26	43	19	29	3	0.5	1	97	1	0	94305	0	0	0

Figure 3. Graph showing the imbalance of the Personal Loan variable (left), the Variable Summary Table (right), and example data from UniversalBank dataset (bottom).

Class Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

Data Role	Variable Name	Role	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
TRAIN	CD_Account	INPUT	2	0	0	93.96	1	6.04
TRAIN	CreditCard	INPUT	2	0	0	70.60	1	29.40
TRAIN	Education	INPUT	3	0	1	41.92	3	30.02
TRAIN	Family	INPUT	4	0	1	29.44	2	25.92
TRAIN	Online	INPUT	2	0	1	59.68	0	40.32
TRAIN	Securities_Account	INPUT	2	0	0	89.56	1	10.44
TRAIN	Personal_Loan	TARGET	2	0	0	90.40	1	9.60

Figure 4. Class variable summary statistics.

Interval Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

Variable	Role	Mean	Standard Deviation	Non Missing	Missing	Minimum	Median	Maximum	Skewness	Kurtosis
Age	INPUT	45.3384	11.46317	5000	0	23	45	67	-0.02934	-1.15307
CCAvg	INPUT	1.937938	1.747659	5000	0	0	1.5	10	1.598443	2.646706
Experience	INPUT	20.1046	11.46795	5000	0	-3	20	43	-0.02632	-1.12152
Income	INPUT	73.7742	46.03373	5000	0	8	64	224	0.841339	-0.04424
Mortgage	INPUT	56.4988	101.7138	5000	0	0	0	635	2.104002	4.756797
ZIP_Code	INPUT	93152.5	2121.852	5000	0	9307	93437	96651	-12.5002	486.2043

Figure 5. Interval variable summary statistics.

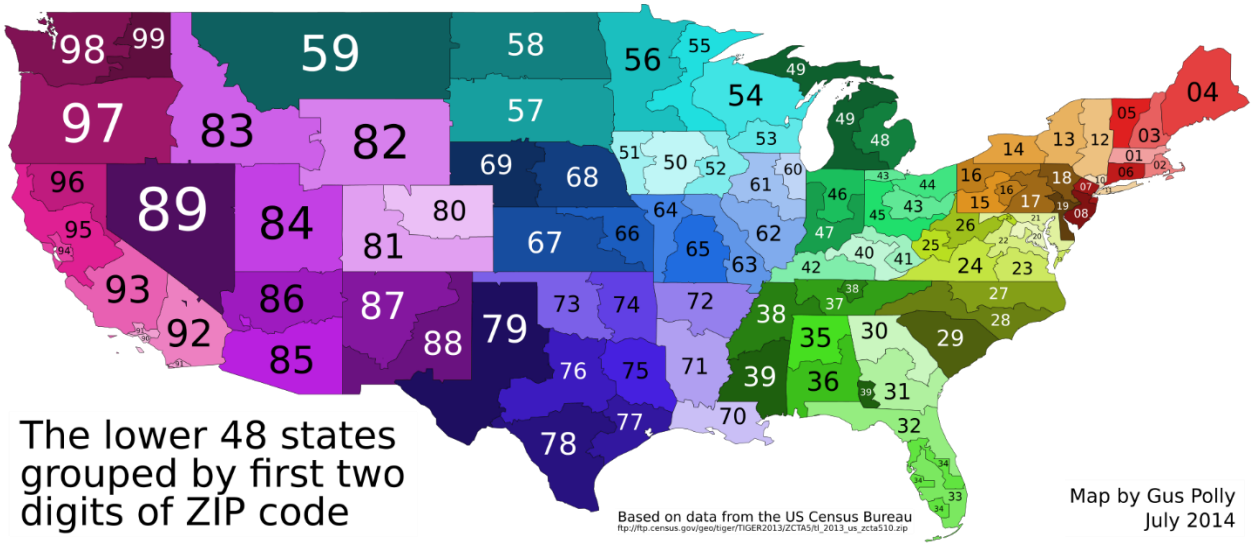


Figure 6. Map showing the lower 48 United States grouped by first 2 digits of ZIP code (All 90-96 ZIP codes can be seen in California)(Polly, 2014).

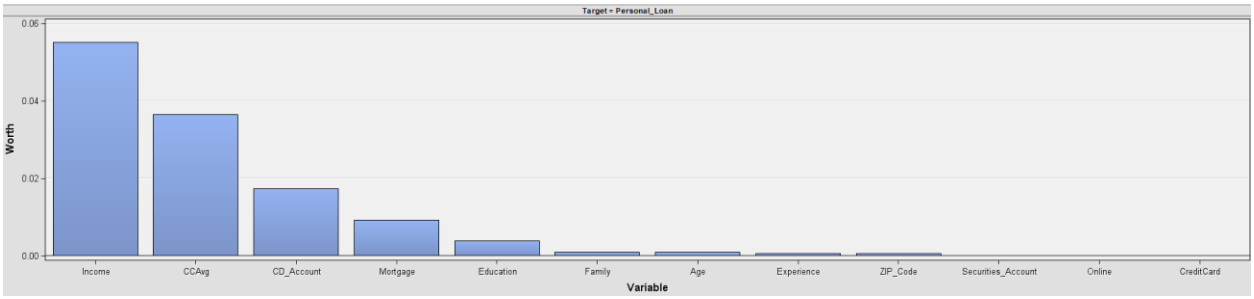
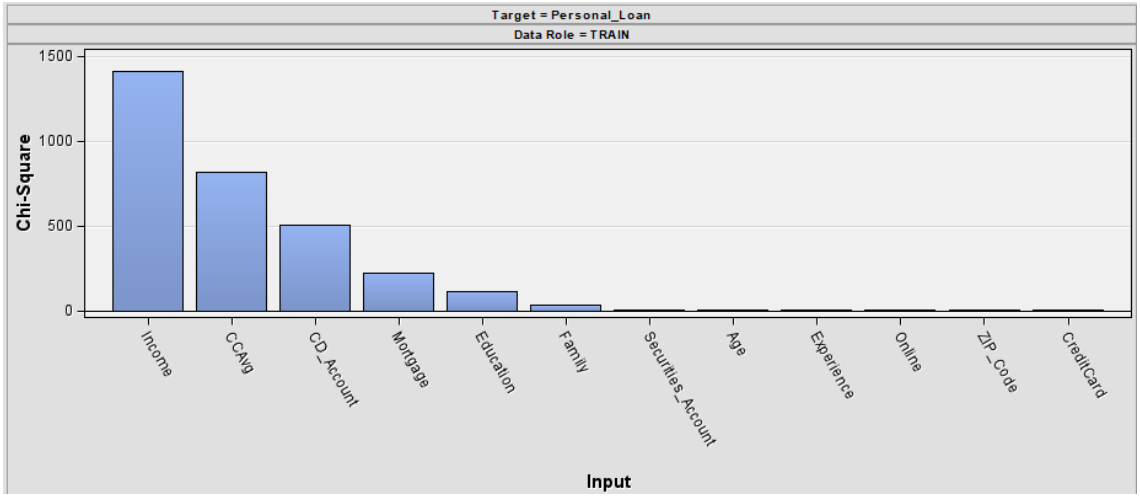


Figure 7. Variable worth for each variable in the UniversalBank dataset.



Chi-Square Statistics
(maximum 500 observations printed)

Data Role=TRAIN Target=Personal_Loan

Input	Chi-Square	Df	Prob
Income	1410.6154	4	<.0001
CCAvg	817.4473	4	<.0001
CD_Account	500.4019	1	<.0001
Mortgage	219.3955	4	<.0001
Education	111.2399	2	<.0001
Family	29.6761	3	<.0001
Securities_Account	2.4099	1	0.1206
Age	0.6125	4	0.9617
Experience	0.4612	4	0.9772
Online	0.1971	1	0.6571
ZIP_Code	0.1062	1	0.7445
CreditCard	0.0392	1	0.8430

Figure 8. Chi-Square values for each variable in the UniversalBank dataset (in chart and table form).

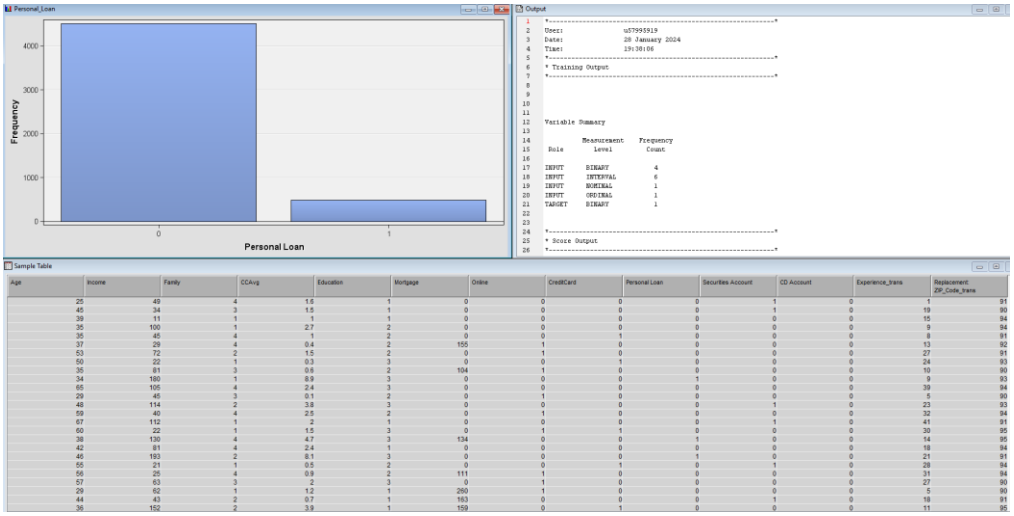


Figure 9. The dataset after preprocessing occurred. The Variable Summary Table (right), and example data from UniversalBank dataset (bottom) can be seen with new variables introduced.

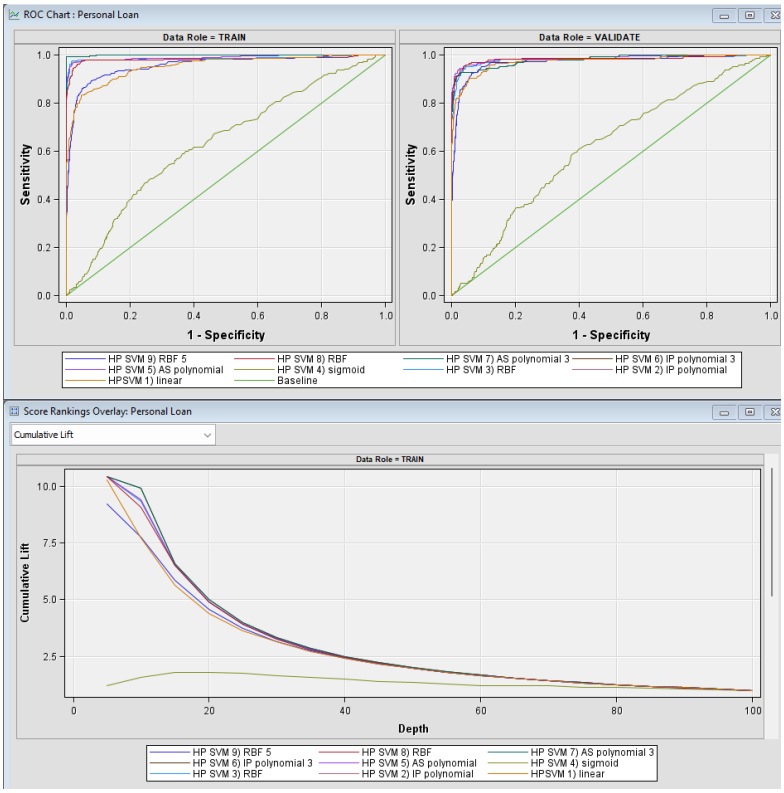


Figure 10. ROC Chart and Cumulative Lift Chart of Models 1-9.

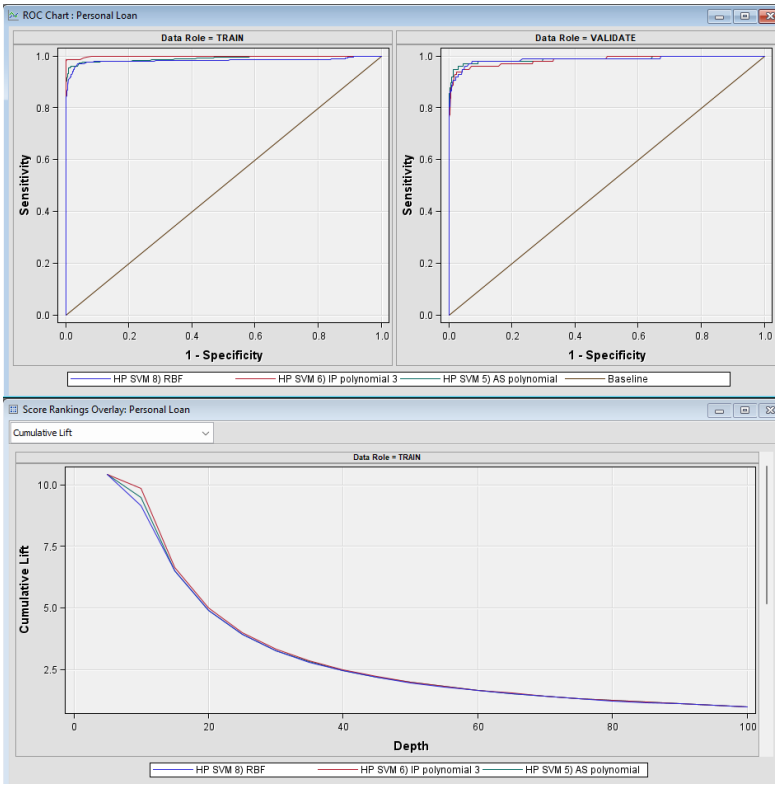


Figure 11. ROC Chart and Cumulative Lift Chart of Models 10-12.

Selected Model	Model Node	Model Description	Valid: Misclassification Rate	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error
Y	HPSVM5	HP SVM 5) AS polynomial	0.01821	0.11473	0.01346	0.11268
	HPSVM2	HP SVM 2) IP polynomial	0.01821	0.11475	0.01346	0.11270
	HPSVM6	HP SVM 6) IP polynomial 3	0.02176	0.13659	0.00400	0.13452
	HPSVM7	HP SVM 7) AS polynomial 3	0.02176	0.13664	0.00400	0.13457
	HPSVM8	HP SVM 8) RBF	0.02842	0.08364	0.03348	0.08186
	HPSVM3	HP SVM 3) RBF	0.03108	0.06479	0.02475	0.06791
	HPSVM	HPSVM 1) linear	0.03552	0.10017	0.04658	0.09501
	HPSVM9	HP SVM 9) RBF 5	0.07060	0.07689	0.07132	0.07433
	HPSVM4	HP SVM 4) sigmoid	0.16563	0.15866	0.15757	0.16080

Fit Statistics
Model Selection based on Valid: Misclassification Rate (_VMISC_)

Selected Model	Model Node	Model Description	Valid: Misclassification Rate	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error
Y	HPSVM10	HP SVM 5) AS polynomial	0.016966	0.11165	0.012006	0.10941
	HPSVM12	HP SVM 8) RBF	0.021956	0.08631	0.028514	0.08548
	HPSVM11	HP SVM 6) IP polynomial 3	0.023952	0.14165	0.005753	0.13927

Figure 12. Fit statistics of the model comparison node for Models 1-12.

Event Classification Table
Model Selection based on Valid: Misclassification Rate (_VMISC_)

Model Node	Model Description	Data Role	Target	Target Label	False Negative	True Negative	False Positive	True Positive
HPSVM10	HP SVM 5) AS polynomial	TRAIN	Personal_Loan	Personal Loan	39	3606	9	344
HPSVM10	HP SVM 5) AS polynomial	VALIDATE	Personal_Loan	Personal Loan	8	896	9	89
HPSVM11	HP SVM 6) IP polynomial 3	TRAIN	Personal_Loan	Personal Loan	20	3612	3	363
HPSVM11	HP SVM 6) IP polynomial 3	VALIDATE	Personal_Loan	Personal Loan	8	889	16	89
HPSVM12	HP SVM 8) RBF	TRAIN	Personal_Loan	Personal Loan	112	3613	2	271
HPSVM12	HP SVM 8) RBF	VALIDATE	Personal_Loan	Personal Loan	21	904	1	76

Figure 13. The Classification Matrix for Models 10-12

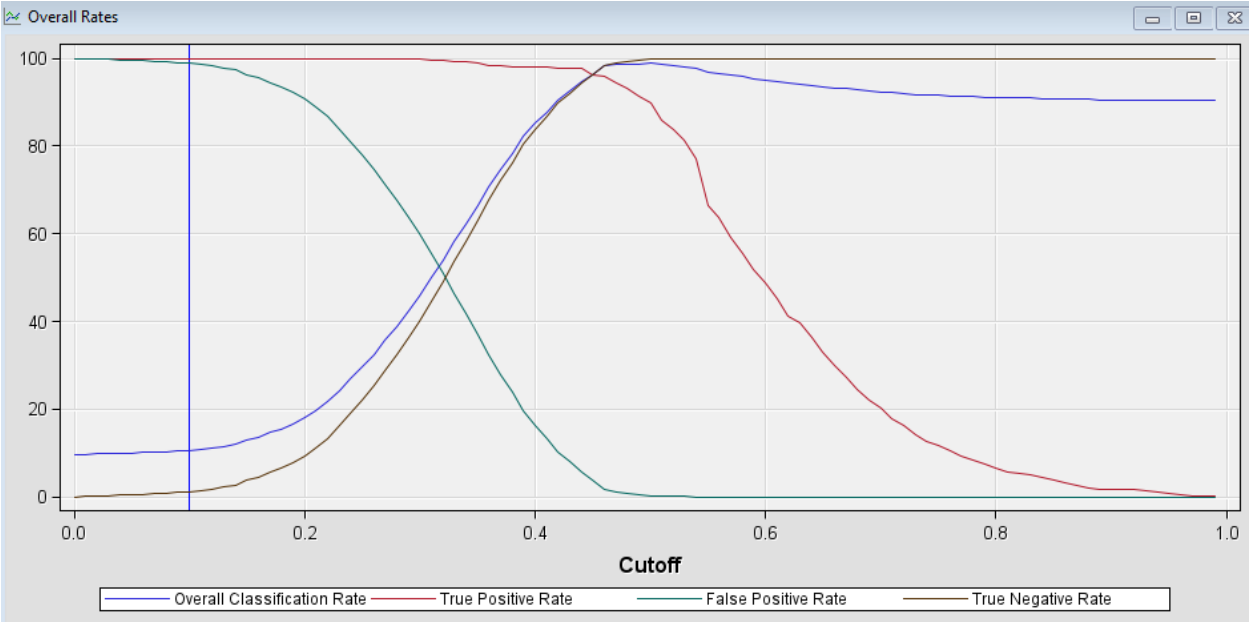


Figure 14. The Overall rates of classification, TP, FP, and TN using a cutoff of 0.1.