Debt Portfolio Capstone

Author: Mark Yingling

[1 Problem Statement 3](#_Toc527196340)

[2 Gather Data 4](#_Toc527196341)

[3 Exploratory Data Analysis (EDA) 5](#_Toc527196342)

[3.1 Data Dictionary 5](#_Toc527196343)

[3.2 Data interpretation and the workflow 6](#_Toc527196344)

[3.3 Feature Engineering 6](#_Toc527196345)

[4 Model the data 7](#_Toc527196346)

[4.1 Logistic Regression 7](#_Toc527196347)

[**4.1.1** Confusion Matrix Evaluation Metrics 7](#_Toc527196348)

[**4.1.2** ROC Curve 7](#_Toc527196349)

[**4.1.3** IBM Watson Studio Model Evaluation 7](#_Toc527196350)

[4.2 DecisionTreeClassifier 7](#_Toc527196351)

[**4.2.1** ROC Curve 7](#_Toc527196352)

[4.3 RandomForestClassifier 8](#_Toc527196353)

[**4.3.1** ROC Curve 8](#_Toc527196354)

[**4.3.2** IBM Watson Studio Model Evaluation 8](#_Toc527196355)

[4.4 XGBoost 8](#_Toc527196356)

[**4.4.1** ROC Curve 8](#_Toc527196357)

[**4.4.2** IBM Watson Studio Model Evaluation 8](#_Toc527196358)

[4.5 KNeighbors 8](#_Toc527196359)

[**4.5.1** Cross Validation 9](#_Toc527196360)

[5 Evaluate the Model 10](#_Toc527196361)

[6 Answer the Problem 11](#_Toc527196362)

# Problem Statement

Debt is a large problem in the United States. Many people have fallen into large amounts of debt much of which have a staggering interest rates attached to it. A significant number of these people have chosen not to repay this debt. This can be attributed to any number of reasons (loss of employment, other costs taking precedence, with the interest being so high they don't see any reduction in the principle amount, etc…) After a period of time trying to collect on the debt creditors will write this debt off and sell it in hopes of recouping some of what is owed. The process by which debt is sold comes in the form of debt portfolios i.e., a number of different debtors are bound together and sold as a percentage of the total amount owed. This could me pennies on the dollar, but the amount paid varies on the amount of information given to the buyer about the debtors included in the portfolio. For instance, all statements and correspondence each debtor has had with the selling creditor. The more data provided the more the portfolio will cost, however the more data you have the better you will be able to predict if someone will eventually repay what is owed.

Getting people who have a demonstrated a propensity of ignoring their debts to pay is the key to success when purchasing portfolio s of debt. How then do we do this? It is our belief that people in this position would like a way out. Either to just have the load of the debt off their shoulders, or to get their credit back so they my buy a house, car, etc… This was implemented successfully with this portfolio by forgiving the interest and fees currently associated with the loan and reducing the interest rate to prime plus 3 for debtors meeting the following criteria:

* Pass a bankruptcy check
* Lives in the Denver metro local
* Remaining balance < 15k
* Didn’t look at interest rates of 8%
* Married and own their home

Can we accurately identify if the portfolio is good one to purchase by determining the Likelihood of someone repaying their debt?

* What were your findings?
* What risks/limitations/assumptions affect these findings?

# Gather Data

We have a portfolio of debt that was purchased on October 24, 2008. This portfolio is comprised of 42 debtors with a total debt amount of $182,222.36. The cost of the portfolio was $6,000.

The data came in the form of two paper spreadsheets. I used Abbyy FineReader, which is an optical character recognition (OCR) piece of software that allowed me to read the paper documents into a .csv file. From there I needed to separate several columns that the software merged together because of several reasons; some of the cells were highlighted, there were several hand written notes, and the pages were taped together so that the rows could be seen with all the columns. I also needed to fix several of the date, name, and address fields for the same reasons as above.

# Exploratory Data Analysis (EDA)

Upon the initial look at the two spreadsheets it became clear that they were the same group of accounts. When the portfolio was purchased two people were doing the analysis. I’m guessing that the two spreadsheets were each individual’s work product. The first data set had less missing data so we chose to work with it.

## Data Dictionary

There are 42 rows of data with 28 columns

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Description |
| Account Number | int64 | Issuers account number |
| PortID | object | Issuer id |
| Remaining Balance | float64 | The remaining balance owed |
| Interests Fees | float64 | Interest and fees associated with the account |
| Date Opened | datetime64[ns] | Opening date for the card |
| Charge Off Date | datetime64[ns] | When account was written off by the issuer |
| Last Pay Date | datetime64[ns] | Last date a payment was made on the loan |
| Last Pay Amount | float64 | The amount last paid |
| Last Activity Date | datetime64[ns] | The date of the last payment |
| Interest Rate % | float64 | Interest rate of the loan |
| Issuer | object | The issuer of the credit card i.e., Chase, or some other bank |
| Merchant | object | This is the company that offers the card to its customers i.e., United Airlines Mileage Plus or the bank itself |
| FCFRA Date | datetime64[ns] | Fair Credit Reporting Act – This is the last time the issuer reported to the credit bureau. |
| OOS Date | datetime64[ns] | Out of Statute – Six years after the last payment made by the consumer |
| Account Type | object | Type of debt portfolio |
| Last Name | object | Debtors last name |
| First Name | object | Debtors first name |
| Middle Name | object | Debtors middle name |
| Address | object | Debtors address |
| City | object | Debtors city of residence |
| State | object | Debtors home state |
| Zip | object | Debtors zip code |
| County | object | Debtors home county |
| SSN | float64 | Debtors social security number |
| Home Phone | float64 | Debtors home phone number |
| Employer Phone | float64 | Debtors employers phone number |
| Cycle | object | How “fresh the account is i.e., how many times did the issuer send it out to a collection agency prior to sale |
| Paid | int64 | Was the debt paid |

## Data interpretation and the workflow

In doing the manual analysis my SME took the following into account:

* Bankruptcy check
* Denver metro local
* Home owner weighted higher then renters
* Remaining balance < 15k
* Didn’t look at interest rates of 8%
* Look to see if they were married
* Looked at the cycle the loan was in
* Calls were also made to each of the debtors and questions were asked to determine their current financial situation.

## Feature Engineering

* Because the Out of Statute date is an important date I subtracted the date the portfolio was purchased from it to determine how many days remained until the loan was out of statute, and then added a column to the dataframe.
* The Cycle is also an important column in determining if someone will pay back the debt. I created dummy values for this column.

# Model the data

## Logistic Regression

The first model I tried was a logistic regression. This model didn’t perform very well.

### Confusion Matrix Evaluation Metrics

I evaluated the model using model evaluation metrics such as accuracy, precision, and recall.

* The model produced classification rate of 82%, considered as ok accuracy.
* Precision: Precision is about being precise, i.e., how accurate your model is. In other words, you can say, when a model makes a prediction, how often it is correct. In your prediction case, when your Logistic Regression model predicted debtors will pay back 0%, of the time.
* Recall: If there are debtors who pay back in the test set and your Logistic Regression model can identify it 0% of the time.

### ROC Curve

Receiver Operating Characteristic (ROC) curve is a plot of the true positive rate against the false positive rate. It shows the tradeoff between sensitivity and specificity.

* AUC score for the case is 0.4. AUC score 1 represents perfect classifier, and 0.4 represents a worthless classifier. This was a consistent score for each run.

### IBM Watson Studio Model Evaluation

I wanted to see how Watson Studio’s automated model builder would do with this data. It’s version of a Logistic Regression model gave a AUC score of .7, however the overall evaluation was “poor”.

## DecisionTreeClassifier

The next model we tried was a single decision tree classifier. We found that there were four features that were important to the model; “Interest Fees” and “Last Pay Amount

* Accuracy determines how often the model is correct. We had an accuracy score of 76.92%

### ROC Curve

Receiver Operating Characteristic (ROC) curve is a plot of the true positive rate against the false positive rate. It shows the tradeoff between sensitivity and specificity.

* AUC score for the case is 0.4. AUC score 1 represents perfect classifier, and 0.4 represents a worthless classifier. This was a consistent score.

## RandomForestClassifier

Our next model was a random forest. We found that there were eight important features for this model; “Interest Fees”, “Last Pay Amount”, “OOS Days Left”, “Remaining Balance”, “Cycle\_Quanternary”, “Cycle\_Tertiary”, “Interest Rate”, and “Cycle\_Secondary”

* Accuracy determines how often the model is correct. We had an accuracy score of 84.6%

### ROC Curve

Receiver Operating Characteristic (ROC) curve is a plot of the true positive rate against the false positive rate. It shows the tradeoff between sensitivity and specificity.

* AUC score for the case is 0.81. AUC score 1 represents perfect classifier, and 0.81 represents a good classifier. I have received vastly different scores with each run.

### IBM Watson Studio Model Evaluation

I wanted to see how Watson Studio’s automated model builder would do with this data. It’s version of a Random Forest model gave a AUC score of .5, however the overall evaluation was “poor”

## XGBoost

XGBoost is an implementation of a gradient boosted decision tree designed for speed and performance.

* Accuracy determines how often the model is correct. We had an accuracy score of 84.62%

### ROC Curve

XGBoost is an implementation of a gradient boosted decision tree designed for speed and performance.

* AUC score for the case is 0.27. AUC score 1 represents perfect classifier, and 0.27 represents a worthless classifier. This was a consistent score.

### IBM Watson Studio Model Evaluation

I wanted to see how Watson Studio’s automated model builder would do with this data. It’s version of a Random Forest model gave a AUC score of .833, also the overall evaluation was “good”

## KNeighbors

The final model used was K Nearest Neighbors. I used the following as features; “Cycle\_Quanternary”, “Cycle\_Tertiary”, and “Cycle\_Secondary

* Accuracy determines how often the model is correct. We had an accuracy score of 93.3%

### Cross Validation

The mean accuracy we can expect from our KNN model we consistently got was ~86%

# Evaluate the Model

It is a bit difficult to evaluate the performance because each time we ran the models we inconsistent scores for each model, however As we increased the model complexity the accuracy increased. XGBoost, Random Forest and K Nearest Neighbors were comparable in their performance, however the XGBoost and Random Forest model used more features. Using more features gives the model more information to make a better prediction for this reason and I also saw similar performance when a Gradient Boosted model was built using IBM Watson Studio I would deploy the XGBoost model.

|  |  |
| --- | --- |
| Model | Accuracy |
| Logistic Regression | 76.9 |
| Decision Tree | 69.2% |
| Random Forest | 84.6% |
| XGBoost | 84.62% |
| K Nearest Neighbors | 86.5% |

# Answer the Problem

Going back to the original problem statement; Can we accurately identify if the portfolio is good one to purchase? I believe that the model can give a very good indication as to whether a portfolio of debt will have enough debtors that will repay the debt to produce a good return on the investment.

End of document