**Business Analytics Project Report**

June 23, 2019

Subject Matter Expert: Robert Sacco

MSBA Capstone Advisor: Dr. Yasin Ozcelik

***All of my files are on GitHub, which can be found through this link:*** <https://github.com/dsacco22/Capstone_Project.git>

**Project Motivation/Background**

The motivation for this project is to start to bring people away from using Excel based graphs and charts, as Excel is very clunky and outdated compared to other programs that are not being utilized by big companies because they are afraid to make the transition. R Shiny is at the forefront of data visualization, and I was not able to have the opportunity to use it in the Business Analytics Program, but major companies like to see that potential employees have a background in using R, even if it is for visualizations only.

Excel is very slow sometimes when handling hundreds of thousands of rows of data that has a vast number of attributes. The power that R has as a statistical programming language is underutilized by major companies, and I would like to show a team in the mortgage banking sector at JP Morgan Chase the advantages of using R over Excel.

**Problem Statement/Statement of Work**

The project that I intend to create and deliver includes a data warehouse containing years of public use data from Fannie Mae and Freddie Mac that will serve as a backbone for a visualization application using R Shiny. The purpose of the application is to create a simple way for the end user to use an easy-to-use UI, and ultimately generate a report that is formatted in a CSV file that can be used for further analysis.

The data warehouse is going to be hosted on AWS cloud using PostgreSQL as the infrastructure of the data warehouse. I have never used PostgreSQL before, but I am aware that there are forums and discussion boards for help when I am stuck. From my understanding, there are minor differences from using MySQL and Oracle, which I have experience in using SQL. I will be able to troubleshoot any SQL errors that I have through those forums and discussion boards.

I have to also create some sort of analytical model that will be used with the data. I will create two models. One model using Logistic Regression and the other model using XGBoost for Decision Trees.

**Data Sources**

The data source that I used was from a public use file from fhfa.gov. Please follow this link to find the data files that I downloaded:

<https://www.fhfa.gov/DataTools/Downloads/Pages/FHLBank-Public-Use-Database-Previous-Years.aspx>

The two data sources that I used were from the year 2016 and 2017. I also have the ‘Data Dictionary’ in my GitHub Repository.

**Data Preparation/System Design**

I will have the system design at the end of this section. I will have the appropriate file names to refer to in my GitHub repository for viewing.

The data preprocessing model that I would use for the 2016 data would be MinMax, but I did not have to do preprocessing. The features for the data that I was using was multipliers based on values from the original data set, that can be found from the FHFA website. What the multipliers do is essentially bin the values for me, and then I am able to create analytical models from that. I did not bin them using regular values (such as 1,2,3,4,5 etc.) because they are already binned, and they can be changed depending on who is using them. The multipliers are also used beyond binning the records for each attribute.

The design of the system is straight forward. I downloaded the CSV files from the FHFA website, and imported them into a data warehouse from AWS that is stored in the AWS cloud. I used PostgreSQL as the infrastructure for the data warehouse, and there are certain packages that I had to import in the analytical section of my project in order to pull the data from the database. I tried to create a read-only user for the database, as well as the role for the read-only user, but I think that I am missing a procedure for the read-only user. This is something that I would like to incorporate in the future for the data warehouse, so I can only have admin access/privileges.

From there, I pulled the data into my Python notebook, and began building the models for Logistic Regression and XGBoost.

**Modeling/System Walkthrough**

For the modeling, I created two models, Logistic Regression and XGBoost. As I stated in the section before, I did not have to do any preprocessing for the data, as everything was binned into multipliers for the data.

For importing the data into the notebook, I had to use a package called ‘psycopg2’ for connecting and pulling the data from the PostgreSQL data warehouse. There was a bit of troubleshooting with this section, as I had to create a couple of variables to make sure that I could connect, and then run the SQL statement that would pull all of the data from 2016 into the notebook for the models.

For the descriptive statistics, I did the usual descriptive statistics that anyone would do when going through new data for the first time. Again, I did not have any data that needed to be preprocessed. I also did some recoding for the ‘PropType’ column, as it was an object, since it had alphanumeric characters for its records.

I first built the Logistic Regression model, using 25% of the 2016 data as the testing set, with the random seed of 20317. I also used RandomOverSampler for the training data, as I wanted to make sure that the data was truly randomized and not biased in any way.

While creating the Logistic Regression model, my accuracy and precision were above 88% for every single model that I tried creating. I have about 56,000 rows of data with 8 features that I am using for the Logistic Regression model, I tried to make the model as least accurate as I could to make sure that it would run against another data set.

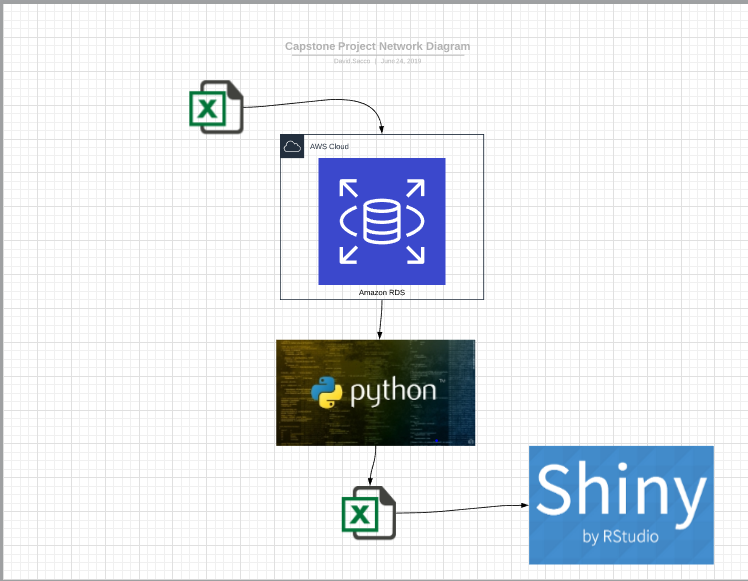
The F1 score that I had for the model was 91. At the end of the script for the Logistic Regression model, I had previously imported the ‘pickle’ package, which allows me to take the model that I created, and use it against another data source, the 2017 data.

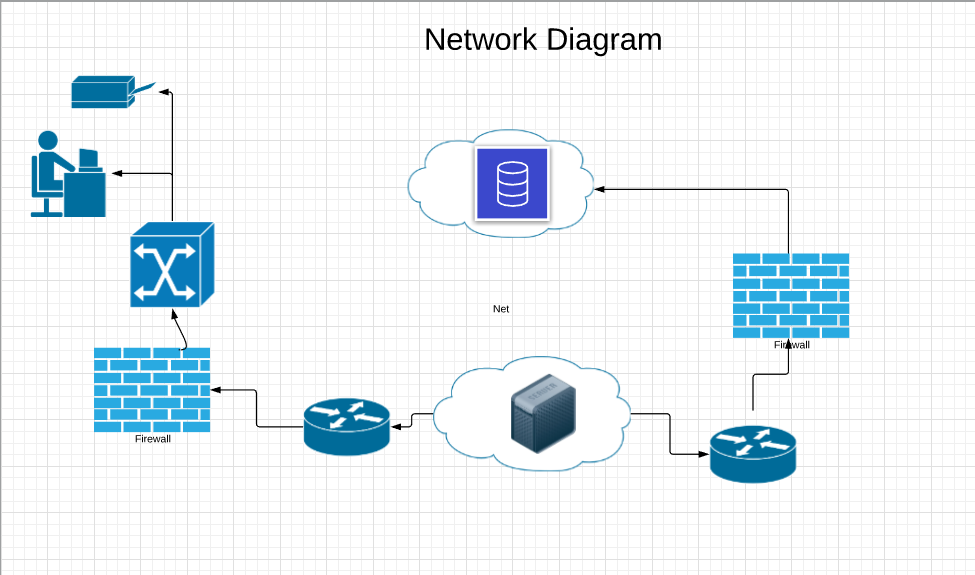
I used the 2017 data against the model and the results were not very good. Almost all of the predicted values for the ‘Portfolio\_Worthy’ column were ‘1’, meaning that they were able to be kept by the team who does the loan evaluations. This was however not true in a lot of cases. It seems that the model was overfit using the 2016 data. I would think that I can use 2015 and 2016 data together and see if that would help create a less overfit model.

After this, I created an XGBoost model that had an RMSE of 0.117355. This is also very good, and it occurred to me at this point that this model was going to be overfit as well. I used the 2016 data to create the model, so using more than one year’s worth of data would be the best thing to avoid overfitting. I then exported the XGBoost model using the ‘pickle’ package, and attempted to use it on the 2017 data. I am still working on that, as I am running into a few errors that I can fix after searching for them on forums/community boards.

After the modeling, I then exported the Excel file so it can be overlooked by the team that would look at it.

**Visuals**





**Interpretation/Application**

Interpreting the data is essentially straight forward, if the loan has a ‘1’ for the ‘Portfolio\_Worthy’ column, then the loan can be either held in the portfolio, or sold and not kept in the portfolio. These teams that work with these loans look over each loan individually and rate them, and only keep a select number that they are allocated per quarter or bi-annually. Being told which ones are able to be kept is about 40% of the work, and then a team goes into the loans and examines them and then based on a set of factors, they will hold them or sell them.

I do not have access to the criteria that they use for the loans that they will keep, or how many they keep at one time. The process is not very quick to determine which loans will be kept or sold off. From my understanding, there is a team in each bank location that reviews all of the loans that are considered worthy to be kept, and then they submit a list to the people that are higher up, and then they make a decision which ones they are going to keep in their portfolio.

**Next Steps**

My first next step is to incorporate the 2015 data into the logistic regression model and see if that will help the Logistic Regression and XGBoost model. After, I would like to develop an application that does this seamlessly for the people that do this, as it will eliminate the time that it takes to generate these Excel sheets. I would preferably like to do it through R Shiny, but I did not have enough time to create an R Shiny app that would be able to sustain what is being asked of me for this project. I feel that it would be a very easy and straight forward GUI, so anyone is able to adjust the parameters that they desire to be in there, and it will generate an Excel sheet that can be easy to use. I would like to learn how to create one of these applications, but it is out of the scope of this class. I would like to roll it out to a select team first, and then get feed back from it, and then roll it out for more teams, based on needs.

**Final Sign-Off**

Dr. Yasin Ozcelik:

Robert Sacco: