SpringBoard Data Science Capstone Project I-

Predicting Bankruptcy Possibility of Emerging Business

Dan Wang

May 5th, 2020

Contents

1. Introduction
2. Data Pre-processing

* Get Column Names from Website
* Handle Missing Values
* Identify and Remove Outliers

1. Data Storytelling

* Bankruptcy Rate Over the 5 Year Period
* Data Structure
* Feature Selection

1. Models

* Logistic Regression
* Gaussian Naïve Bayes
* Random Forest

1. Model Comparison
2. Introduction

When people first start a business, the financial performance at the beginning stage may not be well. Loans, notes payable or mortgage will be a good strategy for business to conquer some difficult times. However, not all loan applications will get approved since not all businesses will succeed. Bankers or investors definitely need to separate highly bankruptcy business out from all the others in order to lower their investment risk. Based on the interest of bankers and investors, my project here tries to utilize machine learning algorithms to make predictions on bankruptcy according to current business financial ratios.

I acquired my dataset from UCI Machine Learning Repository online. The dataset is composed of different financial ratios collected from emerging business. The dataset has 5 separate files and each file has one-year financial information. The five files have the same numbers of attributes, but different number of instances indicating information was collected from different companies each year, but utilized the same financial measurement.

The first dataset contains information from the first year of the forecasting period and indicates the bankruptcy status after five years. The second dataset contains information from the second year of the forecasting period and indicates the bankruptcy status after four years. The third dataset contains information from the third year of the forecasting period and indicates the bankruptcy status after three years. Same rules apply to the rest of the two datasets.

1. Data Pre-processing

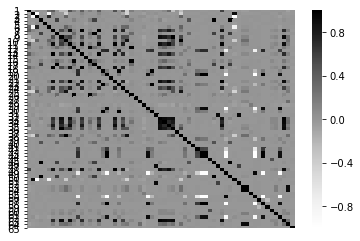
* Get Column Names from Website

From observing the outlet of five datasets, column names are uniform for all the datasets and were placed on the UCI Machine Learning Repository website. Some website scrape libraries need to be utilized in order to extract column names from the website. I utilized BeautifulSoup library and Requests library to finish this task and pass the corresponding column names to all datasets.

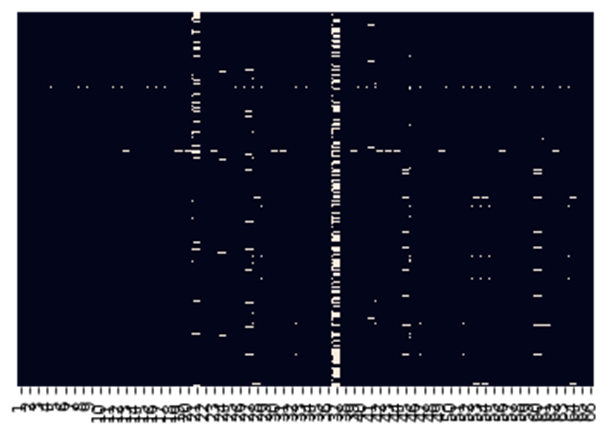
* Handle Missing Values

Since all five datasets have exactly same attributes, I decide to concatenate them into one big data set at current data cleaning stage to make sure all the datasets will still have the same number of attributes later. After the concatenation of all datasets, a simple python code has been launched in order to have a general idea how many data are collected and how many missing values need to be taken care of. The five datasets combined have totally 43,405 entries and 66 attributes (including target attribute).

My first data cleaning process is to extract variables that are independent from each other by implementing Pearson Correlation method on my dataset. Variables have correlation values greater than 0.9 are deleted from dataset. The heat map below is the illustration the correlation of the entire dataset. The darker the area, the higher correlation it is. 14 attributes that have correlation values more than 0.9 are highly dependent from other variables and get deleted from dataset, leaving 42 attributes still in dataset.



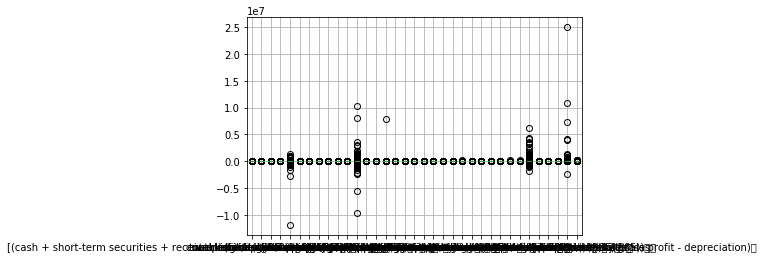
The second step is to handle missing values in dataset. A heat map is created below to help identify these columns and rows. The highlighted cells represent missing values while the dark cells represent normal cells. The more highlighted cells in a row or a column, the higher chances it will be removed. I set my boundary to 2,000 and 10 which means any attribute has more than 2,000 missing values or any row has more than 10 missing values are deemed to lack of meaningful information and get removed from dataset.

****

The first two steps help to take care most of missing values in dataset. The rest missing values are filled in by attributes’ median values since it is the most acceptable methods and avoid the effect of outliers.

* Identify and Remove Outliers

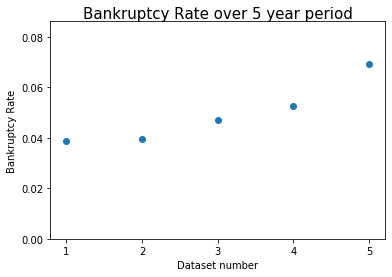
As of current stage, dataset is left with 43,335 rows and 37 attributes. Among the 37 attributes, 35 attributes are financial ratios while the other two variables are target variable and original dataset number. All the 35 attributes are place in a single boxplot to help identify outliers. Figuer below show the distribution of all 35 attributes, the X axis is the name of attributes while the Y axis is the range of all attributes. Obviously, some outliers need to be removed from dataset. Data frame quantile method is used here to identify outliers. Financial ratios less than 5th quantile or greater than 95th quantile are deemed to be outliers and replace with median value of that specific attribute.



1. Data Storytelling:

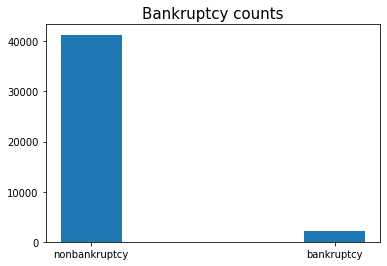
* Bankruptcy Rate Over the 5 Year Period

The dataset is a 5 year dataset. The bankruptcy rate over the 5 year period is ploted on below chart to demenstrate the bankruptcy trend. The first and second year bankruptcy rate is relative stable around 4% and started to increase at third year and finally reached to 7% at fifth year. What the plot shown makes sense to me since it is very common for startup business not perfoming well at the beginning stage, but continue in business for a couple of years, and then finally file bankruptcy if keep perfoming poorly.



* Data Structure

The five datasets has more than 45,000 records in total and the bankruptcy rate is around 4% to 7% which means more than 41,000 records shows non-bankruptcy, the rest of the 4,000 records are bankruptcy. Both the bankruptcy versus non-bankruptcy ratio and the total record numbers of each category imply the potential issue of the dataset, class imbalance. A histgram has been plotted below in order to show the severity of dataset imbalance.



It is important to fix the imbalanced datase before training models since some models may favor majority outcomes and lose accuracy which means the model can simply predict all business not going to bankruptcy and have more than 93% accuracy. In order to avoid that kind of scenario, an upsampling technique has been utlized which creates repetitive bankruptcy records till the ratio of bankruptcy versus nonbankrupcy reaches 1:1. When the quantity of bankrutcy records and the nonbankruptcy records are the same, different models can be trained according to the transformed dataset.

* Feature Selection

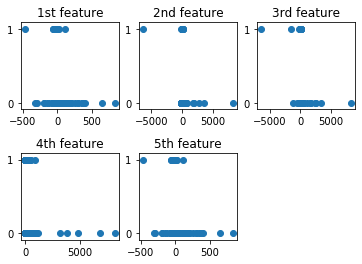
In order to select features that contribute most to the target variable, the SelectKBest method with score function parameter equals to chi-squared test was adopted to help. The chi-squared test is used to test the independence of two variables. When two variables are independent, the chi-square value is low. When two variables are dependent, the chi-square value is high. Since I am looking for features that are most dependent with target variable, features with higher chi-square values are the desired features. However, chi-squared test requres all data to be non-negative which means my dataset need to be rescaled. The normalization method is used here to rescale my dataset to have values between 0 and 1. Passing the normalized dataset in to the SelectKBest method will return a list of scores of all features based on the chi-squaer test results. The most relevant five features with scores greater than 280 are selected to train models.

Feature Name Score

* (gross profit + depreciation) / total liability 394.800566
* (net profit + depreciation) / total liabilities 372.687015
* gross profit / short-term liabilities 332.098007
* book value of equity / total liabilities 293.275631
* gross profit (in 3 years) / total assets 288.572538

Below is the scatter plot to visualize the relationship between the selected five features and the target variable. The first feature contributes the most while the fifth feature contribute least

Relationship between features and target variable:



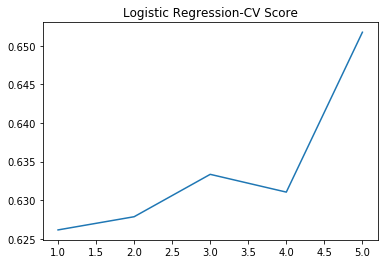
1. Models:

The main idea of my project is to predict if startup businesses going to bankrupt by analyzing their financial performance. It is a kind of classication machine learning projects and three classification machine learning algorithms will be utilized and compared to see which model fit the dataset better and therefore can make better prediction. The three models are logistic regression, Gaussian Naïve Bayes and Random Forests.

Different models may require differnet number of features in order to achieve maximum training effect. In another word, it is another task to find suitable number of features for each model. Since most important five features have been seletcted and ranked by their contribution. I used the same way to select number of features for each model. As a start point, I used only the most important feature to train the model and then, adding the next level important feature in the dataset to train the model. I kept adding the next level important feature in the dataset until all five features was put into dataset to train the models. During this process, a cross validation score was collected before adding in additional feature.

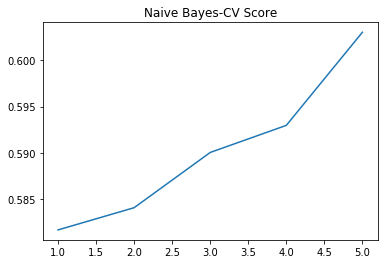
After the data training processing, each model will have five cross validation scores corresponding to total number of features that were put into training dataset. The best number of features for each model will be determined by observing the trend of cross validation scores. The accuracy score will also be calculated after the best number of features are selected. The model has the highest average cross validation score and highest accuray score will get selected.

* Logistic Regression



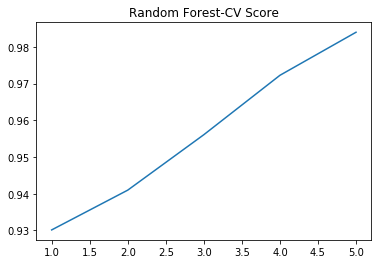
The cross validation scrore keeps increasing from adding features in the dataset and decreasing from adding the fourth feature, but increased again by adding the fifth feature. Even though the highest cross validation scross is presented when all five features are put in the dataset, I think the best number of features for logistic regression is three. The fourth and fifth feature may overfit the model. When passing in the first three features in the dataset to train the model, a cross validation score and an accuracy score are calculated as 0.6333 and 0.438.

* Gaussian Naïve Bayes



The cross validation score of Gaussian Naïve Bayes model keeps increasing from adding additional features in the training dataset. However, it is not a very meaningful increase. The increased portion between one feature and all five features is around 0.15, from 0.581 to 0.603. Even though adding features into the dataset can increase cross validation score, a dramatic score increasing can not be expected. I still picked the five featurs to train Gaussian Naïve Bayes model to get the cross validation score and the accuracy score which are 0.603 and 0.306.

* Random Forest



The Random Forest model looks like the Gaussian Naïve Bayes model, the cross validation scores also keep increasing by adding addional features in the training dataset. However, Random Forest model derives much higher cross validation score than Gaussian Naïve Bayes. Even though the highest score is seen by adding all five features in the dataset, I would select first four features to train the model since the fifth feature only contribute 0.01 score increasing which can not count as a meaningful increase. I even changed the number of n\_estimators from 3 to 4 and also to 6, the cross validation score still changed a little bit. After passing the first four features into Random Forest model, the cross validation score is calculated as 0.972 and the accuracy score reaches 0.900.

1. Model Comparison

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **# of features selected** | **Cross Validation Score** | **Accuracy Score** |
| **Logistic Regression** | **3** | 0.6333 | 0.438 |
| **Gaussian Naïve Bayes** | **5** | 0.603 | 0.306 |
| **Random Forest** | **4** | 0.972 | 0.9 |

All three models’s best number of features have been selected and two indicators have been calcuated from these three models. It is obvious that the Random Forest model did a much better job comparing to the other two models. The Random Forest model will be selected to make predictions.