**Odds of Bankruptcy**

**“Company Bankruptcy Prediction”**

Hayden Horn, [HHorn@bellarmine.edu](mailto:HHorn@bellarmine.edu)

Jacob Schelonka, [JSchelonka@bellarmine.edu](mailto:JSchelonka@bellarmine.edu)

In our project, we will be creating predictors using multiple machine learning models to predict whether or not a company will go bankrupt. In order to get these experiments to predict if a company will go bankrupt, we had to remove some variables that had little to no correlation to a company going bankrupt. Due to this we only used 15 variables for most experiments, and then 20 for some of them. The data had no null values, so there were no issues of missing values. There are two experiments of every model, one split 1/3rd with 15 categories, and the other having 20 categories and a 1/4th split. We think it is best to run similar perimeters using multiple models, as this will show us a variety of accuracy and make the experiments easier to compare. With the there being more models, there should be one predicter that is very accurate at predicting whether or not a company will go bankrupt, which is our main goal.

In the end, the most accurate experiment are the Support Vector Regression and Random Forest experiments. All experiments were extremely accurate though, some (DT) seemed to be over fitted and extremely accurate, but still flawed. The most inaccurate experiment was done by linear regression, even though it had only a 0.0495 R-squared value. In the end, Random Forest and the second SVR models were obviously the best. The random Forest was the best, because of the reduction of overfitting and averaging the data, rather than just adding it to a continuous equation.

1. **INTRODUCTION**

We will be using the “Company Bankruptcy Prediction” dataset from Kaggle.com. We will be creating an easy way to predict which companies will file bankruptcy. We will be experimenting with all the machine learning model’s we’ve used in class this semester. This includes Support Vector Regression, Linear Regression, Random Forrest, and Decision Tree. Due to the size of the database, we won’t be using the variables with little to no correlation to bankruptcy. We will be doing two experiments with SVR, two experiments with DT, and one experiment for the other models.

Our goal is to create a predicter as close to perfect, while not having to use 95 independent variables. We chose this dataset because it is more difficult to work with than the last project. There are more independent variables, and more entries. With more entries, the predictions should have less error. However, not using all 95 variables may make the prediction less accurate. Attempting to balance out which columns have the biggest impact will also be difficult, as filing for bankruptcy is always different from company to company. One company may file bankruptcy due to access borrowing, whereas other companies could be due to taxation and higher cost of productions. Predicting all bankruptcies with no error is virtually impossible, and since our data is based on reality, reducing the error of the predictor will be difficult.

1. **BACKGROUND**
   1. *Data Set Description*

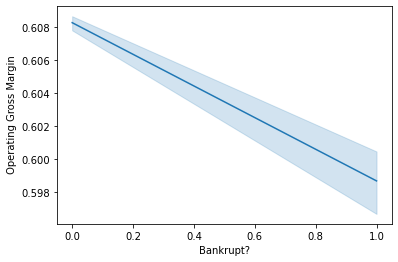
This data set is accessible through Kaggle. The spreadsheet is made up of 6,820 rows and 95 columns. All of the data is integer, or float point data. There is no missing data, which makes it easier to work with. The biggest issue we may face with this data is the size, and number of columns. We chose this data, because it will be more difficult than the last, but there is a better chance at creating a more accurate predictor. The data was collected from the Taiwan Economic Journal for the years 1999 to 2009. This data was collected for the purpose of predicting the chance a company has of filing bankruptcy. This data set is perfect for understanding correlations in company finances and predicting the outcome for each company.

* 1. *Machine Learning Model*

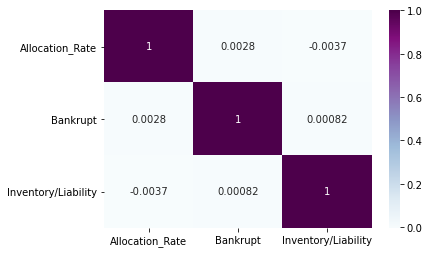
We will be using Multiple Linear Regression, Support Vector Regression, Decision Tree, and Random Forest. Multiple Linear Regression is basically just linear regression, but with more variables to the linear regression equation. SVR tries to find a line/hyperplane that connects variables to each other, attempting to create a perfect prediction using independent variables. Decision Tree views independent variable through branching, and connections. Random Forest does this, but in more complex ways, basically acting as multiple data trees.

1. **EXPLORATORY ANALYSIS**

This *data set contains 6,819 samples with 95 columns with mostly float 64 data types*. All the data used in the experiments are float type data, as the integer data showed no real correlation to whether the company will go bankrupt.



This independent variable has an obvious correlation to whether or not a company goes bankrupt.



This graph shows how little correlation some of the variables have to whether or not a company goes bankrupt.

**Table 1: Data Types**

1. **METHODS**

In this section, describe how you prepared the data for your model and performed multiple experiments using different parameters for the model.

* 1. *Data Preparation*

Firstly, we graphed a majority of the columns in this dataset and checked for correlations, while not focusing on the columns we knew wouldn’t impact the company filing for bankruptcy. We had to drop majority of the columns in this dataset, in order to make the most accurate predictions. For this large of a dataset, the first conflict derives from the number of columns. To overcome this issue, we will only be using only the best 15 columns. The graphs above show ways in which we looked for correlating variables.

|  |  |
| --- | --- |
| *Variable Name* | *Data Type* |
| Fixed Assets Turnover Frequency | Float64 |
| Borrowing dependency | Float64 |
| Equity to Long-term Liability | Float64 |
| Equity to Liability | Float64 |
| Current Liabilities/Liability | Float64 |
| Total expense/Assets | Float64 |
| No-credit Interval | Float64 |
| Operating Gross Margin | Float64 |
| Cash Flow to Liability | Float64 |
| After-tax Net Profit Growth Rate | Float64 |
| Cash Flow Per Share | Float64 |
| CFO to Assets | Float64 |
| Gross Profit to Sales | Float64 |
| ROA(C) before interest and depreciation before interest | Float64 |
| Operating profit per person | Float64 |

IV. METHODS

We didn’t have to fill in any null values or attempt to add to the dataset. This is because there were no null values, and the data is real data, meaning changing any data would remove the realistic portion of the experiments. However, we did choose to not use majority of the categories, as they had little to no correlation. We decided to run all experiments in a similar manner, in order to see which model was the most accurate. In doing so, we decided to use all models twice. The first experiment being a 1/3rd split, and only using 15 independent variables. The next being split by a fourth and using 5 additional independent variables.

A. Data Preparation

We went through majority of the columns in this dataset, not including variables that obviously don’t affect a company going bankrupt. Using only the columns with obvious correlation to the probability of a company going bankrupt, we ended up with only 15 variables that truly matter. Of these 15, all had a major impact on whether a company will need help paying back liability or not. The following independent variables that matter are shown in the table above, labeled “Data Preparation.”

*Experimental Design*

Table X: Experiment Parameters

|  |  |
| --- | --- |
| **Experiment Number** | **Parameters** |
| 1 | MLR: split = 1/3rd: 15 Independent Variables |
| 2 | MLR: split= 1/4th: 20 Independent Variables |
| 3 | SVR: split = 1/3rd: 15 Independent Variables |
| 4 | SVR: split= 1/4th: 20 Independent Variables |
| 5 | DT: split = 1/3rd: 15 Independent Variables |
| 6 | DT: split= 1/4th: 20 Independent Variables |
| 7 | RF: split = 1/3rd: 15 Independent Variables |
| 8 | RF: split= 1/4th: 20 Independent Variables |

* 1. *Tools Used*

The following tools were used for this analysis: Python v3.5.2 running the Anaconda 4.3.22 environment on both a mac and windows laptop for all analysis and implementation. In addition to base Python, the following libraries were also used: Pandas 0.18.1, NumPy 1.11.3, Matplotlib 1.5.3, Seaborn 0.7.1, SKLearn 0.18.1. We needed Python to actual do the experiments, pandas were needed to be able to read and explain the database. Seaborn was important in finding correlations for the categories. We needed SKLearn to finish the experiments and predict bankruptcy. MLR, SVR, DT, and RF were all used to create a predictor.

1. **RESULTS**
   1. *Classification Measures*

1. MSE = 0.161

2. MSE = 0.164

3. MSE = 0.161

4. MSE = 0.169

5. MSE = 0.220

6. MSE = 0.227

7. MSE = 0.104

8. MSE = 0.097

To our surprise, the Random Forest experiment that has more data, had the lowest error. Although not all the data had an amazing correlation, it showed to help with Random Forest. This is shocking, as in all other experiments with 20 independent variables did worse than the one’s with 15.

* 1. *Problems Encountered*

The biggest issue came from how big the dataset was. With 95 independent variables, it became hard to find data that didn’t matter, or made the prediction less accurate. To do so, we had to graph and compare a lot of data. This took up just as much time as setting up the experiments, if not more.

* 1. *Limitations of Implementation*

The model uses all the independent variables in unison, which won’t always work in the real world. One example would be what we’ve seen recently in America: A company owner could be sued for millions of dollars, which usually results in bankruptcy. There is no real way to objectify this, even though in democracy’s it is a big causing agent in bankruptcy. Our model is also off sometimes, even though it is rare. This means, that none of the predictions are actually 100%, which would be good for something like investing, but not as a 100% objective ay of determining if a company will go bankrupt.

* 1. *Improvements/Future Work*

We would like to create a better SVR experiment, with more manipulated data. One reason this experiment was odd, was because all of the independent variables had the same influence on the y value.

1. **CONCLUSION**

Overall, all experiments worked well in finding out whether or not a company will go bankrupt. The most accurate experiment was the 2nd Random Forest experiment, as it had an MSE of 0.09 and predicted all the test results we inputted correctly. The worst experiments were both of the Decision Tree experiments, because both the MSE, and the R2 scores were the worst. We believe this was due to us over training the data.

None of the experiments had an R2 score over a 0.088, or a 0.227. This tells us our experiments worked well, but none of them were perfect. Our best R2 score was a 0.0333 and the best MSE score was 0.097. However, all of our experiments mean square error were in the range of 0.09 and 0.169, which we are happy with. Decision Tree is the obvious outlier, as it is way higher than the other 6 experiments.

**REFERENCES**

<https://towardsdatascience.com/understanding-multiple-regression-249b16bde83e#:~:text=Multiple%20regression%20is%20an%20extension,of%20a%20different%20physical%20parameter>.

<https://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/>