ROE Statistical Analysis and Investment Grade Prediction
MACHINE LEARNING IN FINANCE

Chong Zhao[1], Qiuchen Lu[3], Sixin Ma[2], Yu Chi Chen[4]
[1,2,3,4]Master Candidate in Financial Engineering, University of Illinois at Urbana-Champaign
[1] Student ID: 657063491 E-mail:chongz3@illinois.edu [2] Student ID: 663585699 E-mail:qlu10@illinois.edu
[3] Student ID: 662675907 E-mail:qlu10@illinois.edu [4] Student ID: 672225652 E-mail:yuchicc2@illinois.edu
3252 Digital Computer Lab, 1304 West Springfield Avenue, Champaign, IL 61801 USA

ABSTRACT

This project will start from statistical analysis of important indicators in financial statements and then apply important machine learning classifiers, especially random decision forests classifier to predict the bonds' investment grade. The investment grade are classified into two categories: investment bond and junk bond based on Moody's rating. The investment bond sharing the Moody rating equal or higher than Baa3 while the junk bond sharing the rating lower than Baa3.

I. STATISTICAL ANALYSIS ON ROE

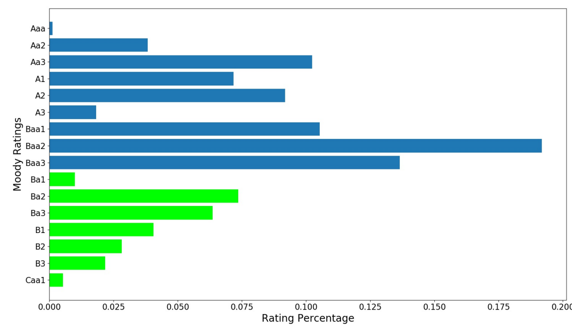
IN this section, the project will discuss the important indicators in the financial statements and especially the contributions to ROE, widely used in decision-making process of investors. The indicators discussed are divided into four categories: Revenue, Debt, Cash Flow and Market Indicators.

Here are classification of indicators.

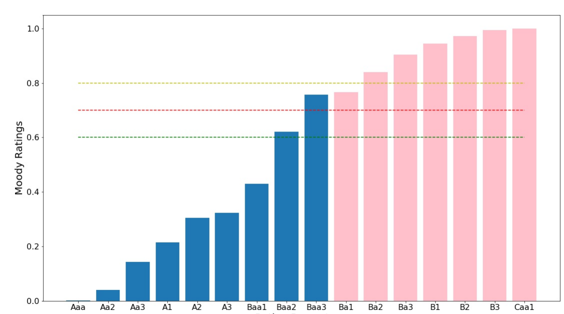
Classification	Indicators
Revenue	Sales, Gross Margin, EBITDA, EBITDA Margin, Net Income
Debt	Total Debt, Net Debt, LT Debt, ST Debt, Net Debt/EBITDA
Cash Flow	Cash, Free Cash Flow, CFO, CFO/Debt, Interest Coverage
Market Value	Total MV, Total Debt/MV,Net Debt/MV,
Liquidity	Total Liquidity, Current Liabilities, Total Liquidity
Analysis Indicators	EPS, PE, ROA, ROE

A. Dataset Visualization

In this part, we briefly show the percentage of differnet rating bonds. The mode and the median of the bonds lie in Baa rating. The green bars represent junk bond while the blue bar represent the investing bond.



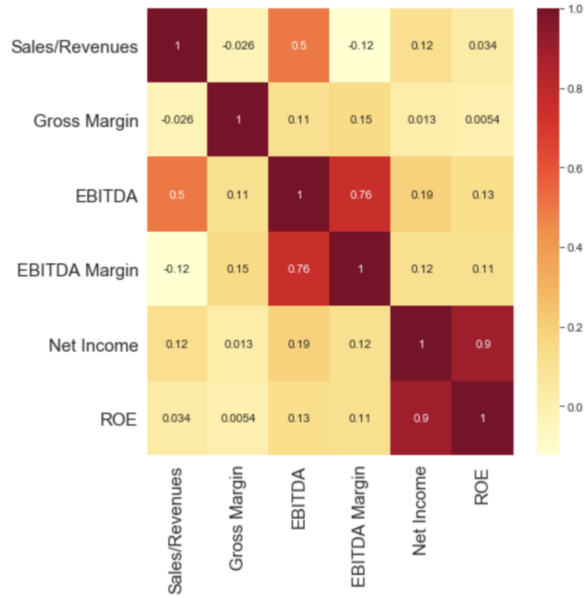
Then the CDF of the bar is plotted to show that the investment bond account for roughly 75% of the total bond. It also becomes clear that the aim of the project is to identify the junk bonds from the total bonds.

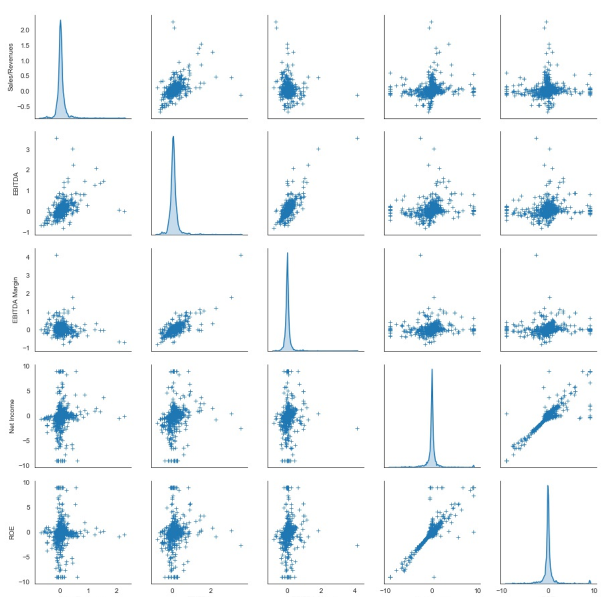


B. Statistical Analysis on Revenue Statements

We will discuss the relationship between revenues and ROE in this part. In order to better test and show the statistical relationship between revenues section and ROE, we standardize the data into the interval [-9,9].

Initially, the heat map and pair plot of revenue indicators and ROE is as following:

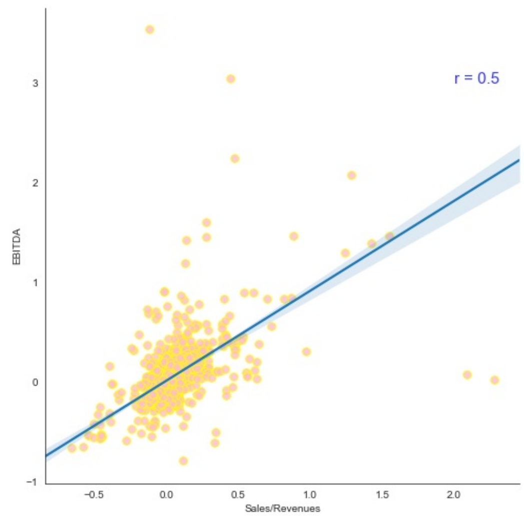




From the heat map and pair plot, several interesting conclusions could be drawn.

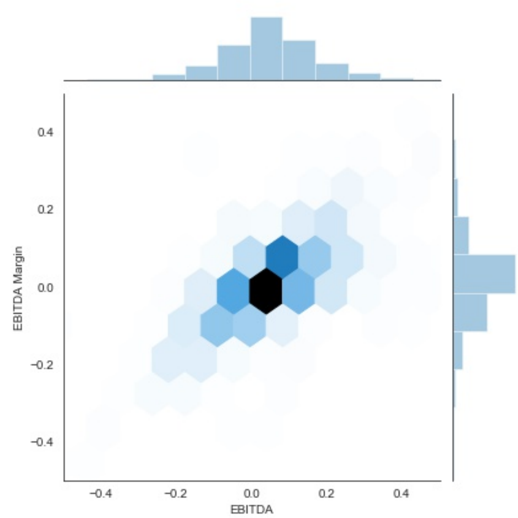
1. It seems that the sales share a close relationship approximately 0.5 pearson correlation with EBITDA(Earnings Before Interest, Tax, Depreciation and Amortization) since the EBITDA equals sales minus operating cost plus depreciation and amortization exactly. However, since sales need to minus operating cost, interest, tax, depreciation and amortization to get net income. The pearson correlation is diluted by the five factors to 0.12.

The following plot shows more details about the relationship between EBITDA and sales. The light blue area shows the confidence interval of the regression line.

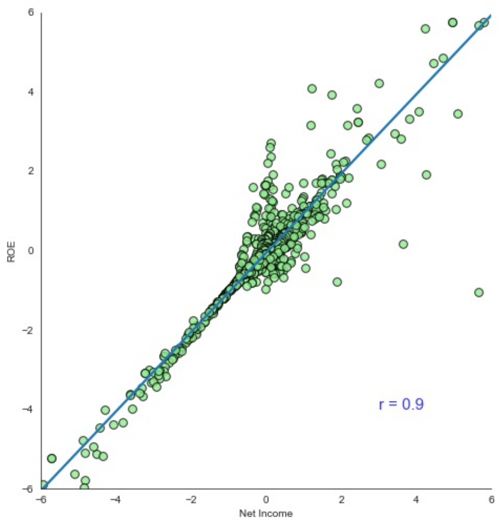


2. It is not astonshing that EBITDA sharing a close relationship as high as 0.76 with EBITDA margin while the sales sharing a negative relationship 0.12 with EBITDA since the EBITDA could be calculated as following formula. The joint plot shows vividly the close relationship between EBITDA and EBITDA margin.

$$EBITDA\ Margin = \frac{EBITDA}{Total\ Revenue}$$



After discussing the indicators within the revenue statements, it is the very time to test the pearson correlation between revenue and ROE. Firstly, we select the net income, which is the most significant factor affecting ROE. We test the explanation extent of the net income by linear regression and R^2 .



The R^2 of net income is:

$$R^2_{Net\ Income} = 0.8034$$

There is only a slight improvement when other factors are added in the linear regression model:

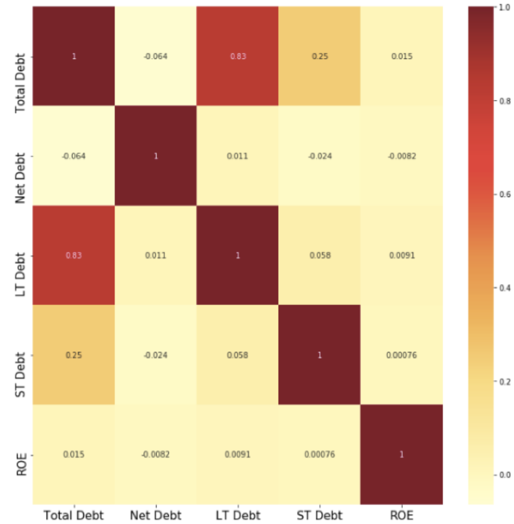
$$R^2_{All\ Revenue\ Factors} = 0.8093$$

Hence, the net income explains the ROE in a very large extent, which could also be drawn from the following fomula:

$$ROE = \frac{Net\ Income}{Equity}$$

C. Statistical Analysis on Balance Sheet and Cash Flow

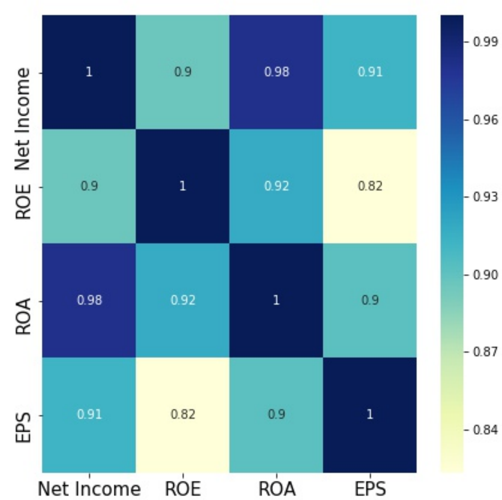
In this part, we would like to explore the relationship between debt and ROE. As shown in the following, we only observe high correlation between long term debt, total debt and short term debt, which is pretty naive. The reason that there is no clear relationship between ROE and debt might be that well-performing firms could borrow to finance project while high debt could also lead to bad results.



Surprisingly, what happens in debt also occurs in cash flow analysis. No clear correlation of cash flow and ROE is detected in the dataset.

D. Statistical Analysis within market indicators

Having done enough statistical analysis in financial statements, we could turn to important corporate finance indicators.

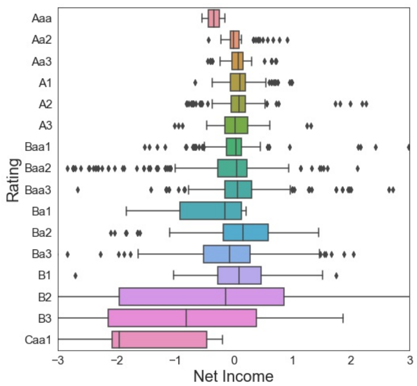


The figure shows that net income plays a significant role to influence the coporate finance indicators: ROE, ROA and EPS. These four indicators exhibit extremely high inter-correlation. The conclusion might be that the net income is the first point to be focused on to perform investment grade prediction.

II. Exploratory Data Analysis on Rating

RATING analysis is the most important focus of this project and in this part we will discuss the relationship between revenue and rating in thorough details.

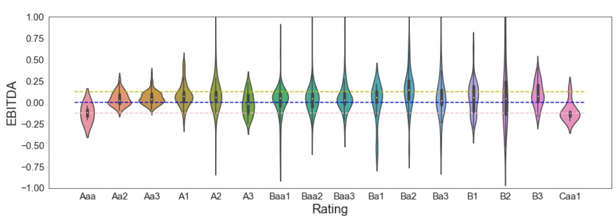
Net income is always a good start point for it influences the ROE and other investing indicators most significant as shown before. So we produce the rating and net income boxplot in this part.

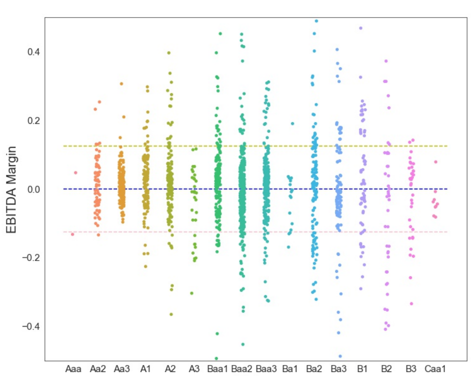


The net income of Aaa is slightly low, but it doesn't indicate meaningful fact because the Aaa data is not enough in this dataset.

In fact, from the graph one could reach the conclusion that as the rating becomes lower, the median of net income tends to become lower and the net income exhibits a higher volatility, which means the net incomes of the companies sharing low ratings are highly instable.

Same things happen to EBITDA and EBITDA margin:





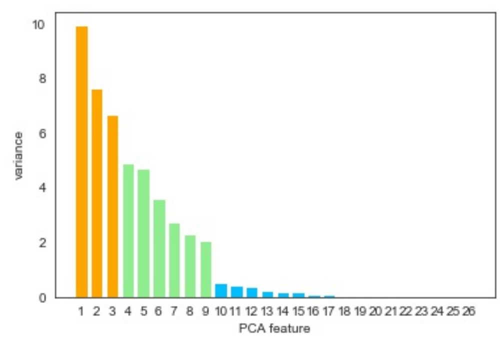
From the two figures, high variance could also be seen as rating is lower and same things also happen to cash flow and EPS.

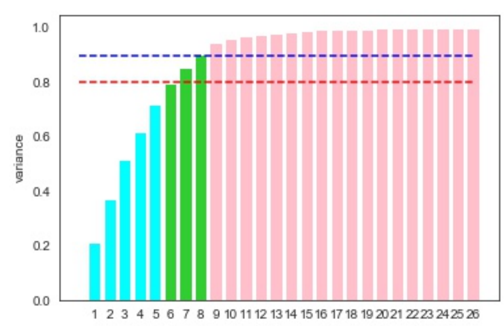
III. Preprocessing and Feature Extraction

THERE is no missing value in the dataset while there are many outliers in each column. While most of the data is standardized well in the interval [-3,3], there are outliers lying too far from the upper bound 3 and lower bound -3. Therefore, we restandardize the data by replacing the data greater than 9 with 9 and less than -9 with -9. In this way, all the factors are restandardized into the interval [-9;9].

Since there are only 26 features, it is suitable that we fit all the features in the model. However, we still decide to apply PCA transformation to identify and extract main features.

The picture in the following shows that even if there are 26 features in total, there are only 9 important features appearing a high variance after PCA linear transformation. Approximately 3 features contributes variance more than 6, 6 features contributes variances 2 to 5, while the other 17 features contributes less than 0.5.





The first 6 features after PCA account for approximately 80% of the total variance while the first 8 features account for approximately 90%. However, considering that the dataset is not so large, we want to keep as much information as we could to improve the model accuracy.

IV. Model Fitting and Evaluation

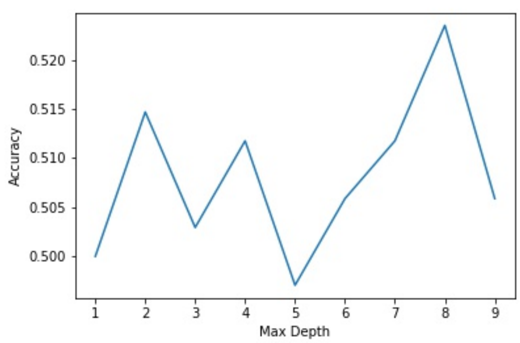
CONSIDERING that this is a typical classification problem: investing bond or junk bond, we decide to try logistic regression, decision tree, support vector machine and KNN to fit our dataset.

A. SMOTE Resampling

After splitting our dataset into training set and testing set, we find that the two categories are pretty imbalanced. As shown before, approximately 75% of the total dataset is investing bond while only a quarter is junk bond. Hence we resample only on training dataset and then fit our model on resampling training set to avoid building connections between training set and testing set.

B. Decision Tree Hyperparameter Selection

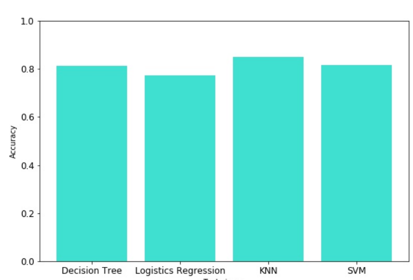
In order to choose the optimal depth parameter and classification criterion. We apply the grid search on the dataset. Here is the fitting result regarding max depth.



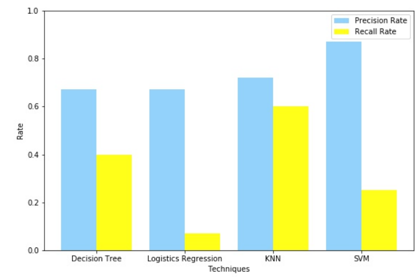
From the figure, we could denote that there is no significant difference as the max depth increases. Hence, it is enough to choose 4 as the max depth. Besides, the Grid Search function suggests that the entropy rule performs slightly better than gini.

C. Fitting and Results

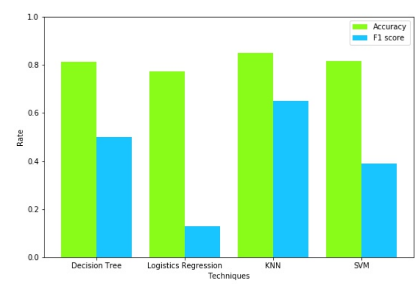
Here comes the accuracy score of the four methods: decision tree, logistics regression, KNN and SVM.



It seems that the result of the four methods are perfect and close. However, when we take a close look on the precision and recall rate, the result is pretty upset.



The precision rate for junk bond is only slightly higher than 60%, suggesting that many investing bonds are classified to junk bond. What is even worse, the recall rate for logistics regression and SVM is so low that most junk bonds are not identified correctly. Rather, they are classified into investing bonds, which is serious and dangerous for a mutual fund. The f1 score basically suggests the fact that there is a large gap between accuracy and f1 score.



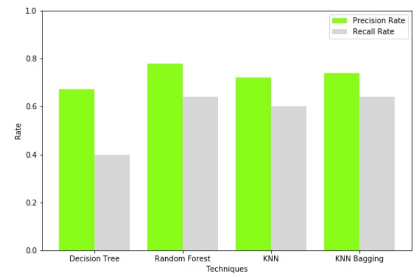
D. Evaluation

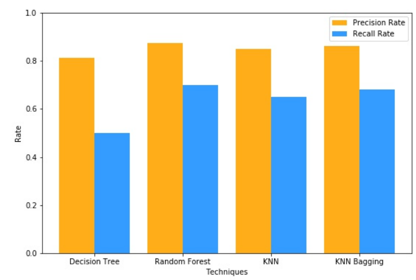
The large gap between the accuracy and f1 score means that the models are trained mostly to indentify the investing bonds while most junk bonds are also classified as the investing bonds. The reason is that the high imbalance in training set, 75% of training set is investing bond. Since there are also many abnormal data, the logistics regression and SVM are especially vulnerable to abnormals since they are more formula-based, while the KNN and decision performs more rubust because they are rule-based methods.

V. Model Ensembling

AFTER testing single classification model, we apply ensembling learning in this section to improve predicting accuracy and f1 score.

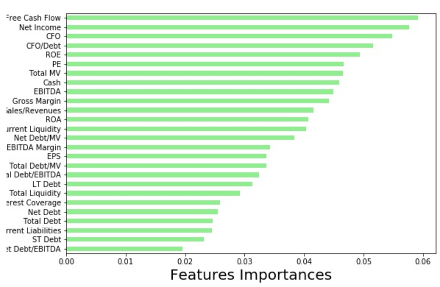
We mainly choose decision tree and KNN model to construst the base of ensembling since the two models perform very robust in the previous section. In this way, we bag the KNN with the nearest number 7 as optimal parameter and apply the random forest algorithm. The result follows:





From the two graphs we could reach the conclusion that the decision tree model is improved a lot after ensembling while the KNN model remains nearly unchanged.

We also apply the feature importance function of random forest and get the following interesting results:



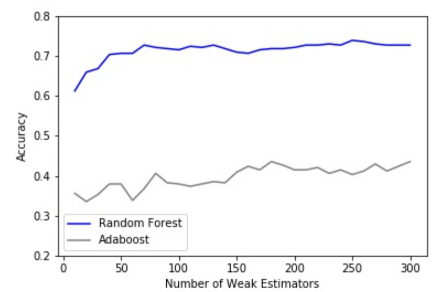
The results show that the most important feature is the free cash flow which could be explained that the more free cash flow, the more stable of the profits and stronger ability to deal with abnormal situation. The figure also suggests that all features contribute while not a single feature contribute too much or too little. That is to say, when it comes to analyze the bond rating of a company, revenue statement, cash flow statement and asset-debt statement are equally important and no one shall be neglected.

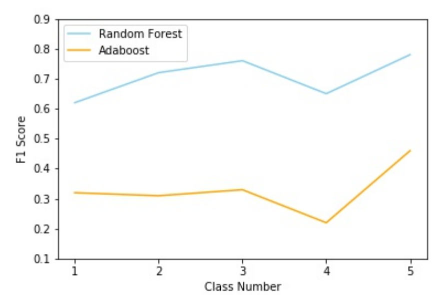
VI. Mutiple Classification

MUTIPLE classificaiton is generally harder than binary classification in the same dataset. So it is reasonable to start simply from random forest and Adaboost classification since we might face more errors than binary classification. Besides, since there are too few tuples in some ratings leading to high volatility in classification, we decided to merge some classes together. The following table shows the merge process in original dataset:

New Rating Label	Original Label
1	B1,B2,B3,Caa1
2	Ba1,Ba2,Ba3
3	Baa1,Baa2,Baa3
4	A1,A2,A3
5	Aaa,Aa2,Aa3

We then try different number of weak estimators by the two algorithms and get the following results:





The first figure shows that the Adaboost performs a pretty low accuracy even if the number of weak estimators increase to hundreds, while the random forest performs a really high accuracy when the estimators increase. Also, the random forest algorithm reaches higher f1 score than Adaboost. The figure suggests that the Adaboost algorithm overfitted the dataset and hence performs poorly on the test dataset.

VII. Conclusions

A. Feature Analysis

The ROE, ROA and EPS is more than 80% determined by net income and as the rating goes down, the net income, EBITDA and EBITDA margin becomes more volatile.

B. Rating Prediction

The SMOTE could rebalance the dataset and therefore increase the accuracy a little.

Even if the 26 features contributes similar importance, it is still possible to extract 9 main features by PCA transformation accounting for more than 90% of the variance.

Although SVM, KNN, Decision Tree and Logistic Regression share a similar 80% accuracy, there is a low recall rate of Logistics Regression and SVM in the imbalanced test set. Considering there are many extreme values in the dataset, the formula based algorithms including Logistic Regression and SVM seems to be pretty vulnerable, while the rule based algorithm including KNN and Decision Tree performs very robustly.

There is a significant improvement in accuracy and F1 score after replacing decision tree by random decision forest, while there is little improvement after bagging the KNN.

In multiple classification, the random decision forest also exhibits robust performance compared to the Adaboost, which could easily overfit the training set for it fits the errors too much.

Resources

In order for others' convenience, I have uploaded the code file on my gitub:https://github.com/Chong-code/IE598. I am looking forward to further discuss the code with you!

Acknowledgement

We very appreciate that Professor Matthew Murphy offers us industrial insights to apply effective algorithms and helpful suggestions to build article framework.

Datacamp courses really help us to move forward in both theoretical analysis and practical programming, without which the results would be unattainable.

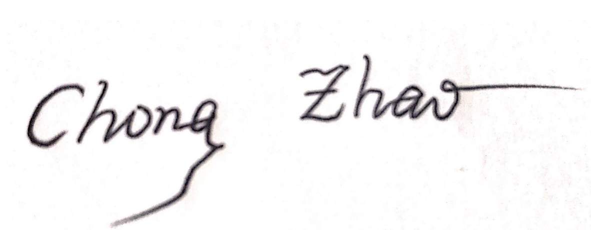
We also appreciate Market Axes who provides us with the valuable opportunity to face real world challenge.

Thank you again!

Academic Integrity

I hereby certify that I have read the University policy on Academic Integrity and I am not in violation of Academic Integrity of University of Illinois at Urbana-Champaign.

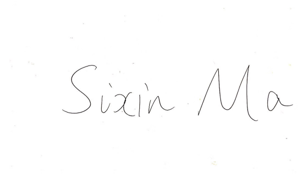
Signature:



Master Candidate in Financial Engineering
University of Illinois at Urbana-Champaign

Master Candidate in Financial Engineering
University of Illinois at Urbana-Champaign

Master Candidate in Financial Engineering
University of Illinois at Urbana-Champaign

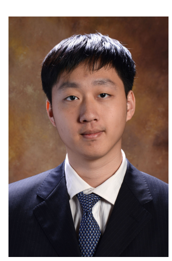


Master Candidate in Financial Engineering
University of Illinois at Urbana-Champaign

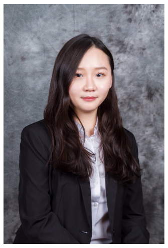
Chong Zhao Master Candidate in Financial Engineering in University of Illinois at Urbana Champaign, IL, USA. Chong Zhao received the B.S. degree in physics from University of Chinese Academy of Sciences, Beijing, PRC, in 2019.

Having interned in various financial and technological companies, Chong is experienced in quantative analysis by python, C++ and R. He is especially passinate about financial derivative pricing and risk management.

Currently, he is looking for 2020 summer internship and full time job after December 2020 in commercial banks, investment banks, insurers and hedge funds. He is open to any global opportunities.



Qiuchen Lu Qiuchen Lu is a graduate student in financial engineering and he completed his bachelor's degree in actuarial science, both at the University of Illinois at Urbana-Champaign.
He is both a market enthusiast and professional engineer. He is a CFA level 2 candidate and he can program in C++, Python, Matlab and R really well. He is actively looking for summer 2020 internship in the investment and trading industry. He is an extremely fast learner and is willing to learn anything to fit for the position.



Yu Chi Chen Yu-Chi Chen is a first-year master's student majoring in Financial Engineering, and she earned her bachelor's degree in Actuarial Science, both at University of Illinois at Urbana-Champaign. She has passed two actuarial exams and is currently pursuing to attain the Associate of the Society of Actuaries. With her academic background and internship experience, she developed her programming skills in Excel, Python, R and C++.
She is actively looking for Summer 2020 intern positions in actuarial science and finance industry. She is excited to learn anything and willing to relocate.

PLACE
PHOTO
HERE

Sixin Ma Sixin Ma, master in Financial Engineering in University of Illinois at Urbana Champaign, received the B.S. degree in mathematics and statistics from University of Illinois at Urbana Champaign in 2019.
Sixin is specialized in R and python programming of data analysis and also familiar with c++ and has interned in some financial companies doing data analysis and industrial research. Besides, she did research by using statistical methods during undergraduate.

US Treasury Bond and Commercial Paper Yield Curve Analysis From 1979 Oil Crisis to 1997 Asian Financial Crisis

Machine Learning in Finance

Chong Zhao[1], Qiuchen Lu[3], Sixin Ma[2], Yu Chi Chen[4]
[1,2,3,4]Master Candidate in Financial Engineering, University of Illinois at Urbana-Champaign
[1] Student ID: 657063491 E-mail:chongz3@illinois.edu [2] Student ID: 663585699 E-mail:qlu10@illinois.edu
[3] Student ID: 662675907 E-mail:qlu10@illinois.edu [4] Student ID: 672225652 E-mail:yuchicc2@illinois.edu
3252 Digital Computer Lab, 1304 West Springfield Avenue, Champaign, IL 61801 USA

Abstract

The report is to analyze data from MLF_GP2_EconCycle, which contains 223 monthly observations of the US Treasury bond and commercial paper yield curves and the USPHCI Economic Activity Index. Our goal is to use the data to predict the percentage change in the USPHCI index 9 month ahead.

I. Exploratory Data Analysis

In order to have a better understanding of the data, we perform the Exploratory Data Analysis. Firstly, we call the *head()* to look into the first five rows of the data. There are 12 features and 3 target variables. Feature variables are the US Treasury Bond yield for 1, 2, 3, 5, 7, and 10 years, and the commercial paper yield for 1, 3, and 6 months. The target variables are the percentage change in 3, 6, and 9 months ahead.

	T1Y Index	T2Y Index	T3Y Index	T5Y Index	T7Y Index	T10Y Index	CP1M	CP3M	CP6M	CP1M_T1Y	CP3M_T1Y	CP6M_T1Y	PCT 3MO FWD	PCT 6MO FWD	PCT 9MO FWD
0	10.41	9.86	9.50	9.20	9.14	9.10	9.75	9.95	10.01	0.936599	0.955812	0.961575	0.011470	0.018060	0.024406
1	10.24	9.72	9.29	9.13	9.11	9.10	9.74	9.90	9.96	0.951172	0.966797	0.972656	0.009298	0.014866	0.020612
2	10.25	9.79	9.38	9.20	9.15	9.12	9.72	9.85	9.87	0.948293	0.960976	0.962927	0.010340	0.015455	0.020154
3	10.12	9.78	9.43	9.25	9.21	9.18	9.86	9.95	9.98	0.974308	0.983202	0.986166	0.006720	0.013141	0.017409
4	10.12	9.78	9.42	9.24	9.23	9.25	9.77	9.76	9.71	0.965415	0.964427	0.959486	0.005653	0.011451	0.016353

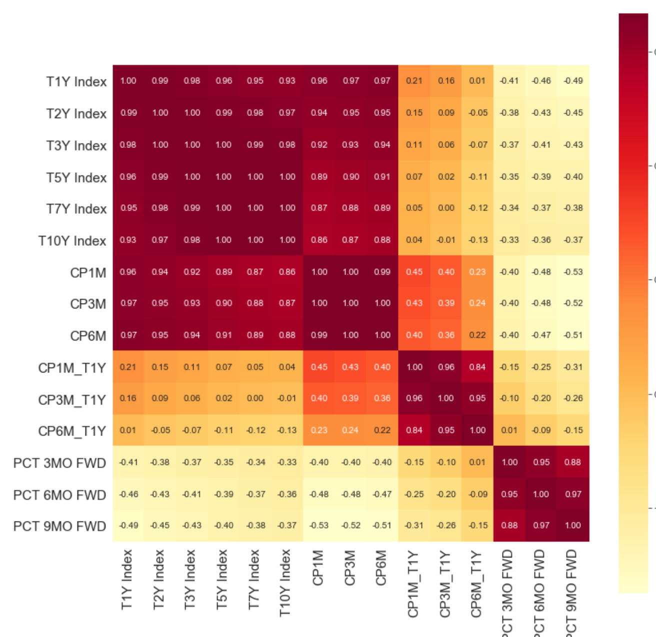
We call the *describe()* to see statistical information about the data. We can see that there are 223 observations in the data. Also, the table has shown the mean, standard deviation, minimum, maximum, and the 25%, 50% 75% quantile for each variable.

	T1Y Index	T2Y Index	T3Y Index	T5Y Index	T7Y Index	T10Y Index	CP1M	CP3M	CP6M	CP1M_T1Y	CP3M_T1Y	CP6M_T1Y	PCT 3MO FWD	PCT 6MO FWD	PCT 9MO FWD
count	223.000	223.000	223.000	223.000	223.000	223.000	223.000	223.000	223.000	223.000	223.000	223.000	223.000	223.000	223.000
mean	8.031	8.411	8.564	8.809	8.980	9.073	7.942	7.937	7.893	0.982	0.984	0.983	0.007	0.014	0.021
std	3.159	2.954	2.820	2.648	2.543	2.448	3.405	3.329	3.181	0.086	0.077	0.067	0.005	0.009	0.013
min	3.180	3.840	4.170	4.710	5.050	5.330	3.110	3.140	3.190	0.718	0.714	0.698	-0.007	-0.010	-0.012
25%	5.735	6.180	6.410	6.695	6.965	7.175	5.605	5.645	5.635	0.934	0.940	0.945	0.006	0.011	0.014
50%	7.670	8.000	8.130	8.330	8.520	8.610	7.730	7.720	7.620	0.973	0.978	0.980	0.008	0.016	0.024
75%	9.840	10.075	10.375	10.525	10.640	10.685	9.345	9.345	9.300	1.033	1.026	1.016	0.010	0.020	0.029
max	16.720	16.460	16.220	15.930	15.650	15.320	18.950	18.070	16.660	1.339	1.277	1.220	0.020	0.037	0.050

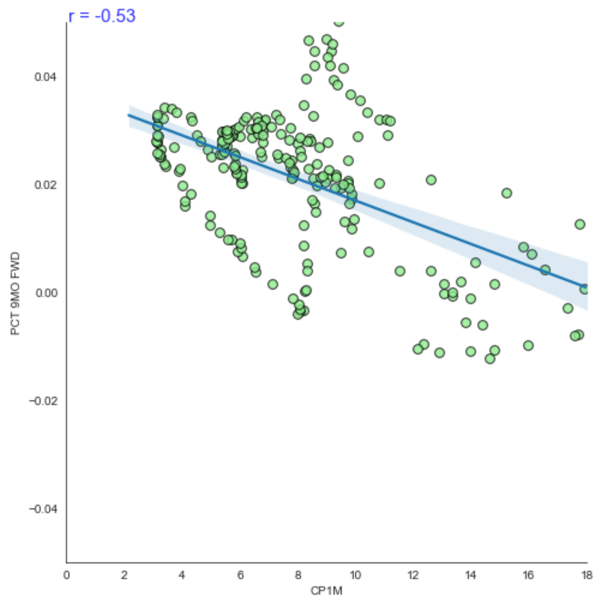
We call the *info()* to look into the data type of each variable. From the following table, we can conclude that the data type for all variable is float64.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 223 entries, 0 to 222
Data columns (total 15 columns):
T1Y Index      223 non-null float64
T2Y Index      223 non-null float64
T3Y Index      223 non-null float64
T5Y Index      223 non-null float64
T7Y Index      223 non-null float64
T10Y Index     223 non-null float64
CP1M           223 non-null float64
CP3M           223 non-null float64
CP6M           223 non-null float64
CP1M_T1Y       223 non-null float64
CP3M_T1Y       223 non-null float64
CP6M_T1Y       223 non-null float64
PCT 3MO FWD    223 non-null float64
PCT 6MO FWD    223 non-null float64
PCT 9MO FWD    223 non-null float64
dtypes: float64(15)
memory usage: 26.2 KB
```

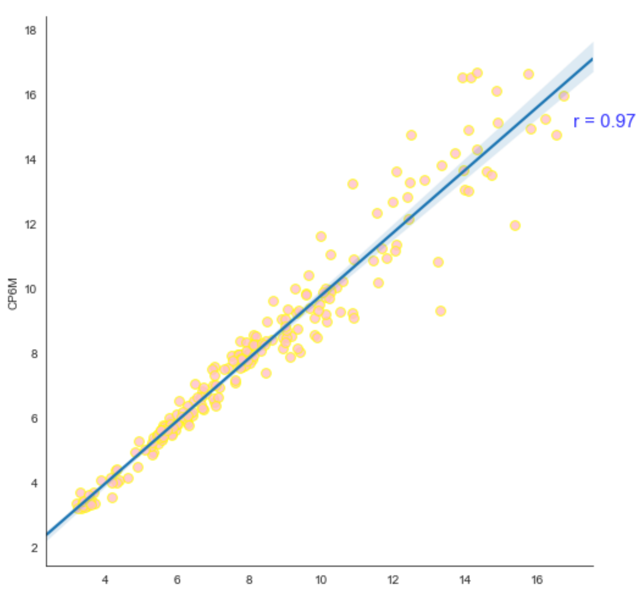
After learning the basic statistical information for each individual variable, we would like to learn more about relationships between the variables. The following heatmap is displayed to show the correlation between each variable.



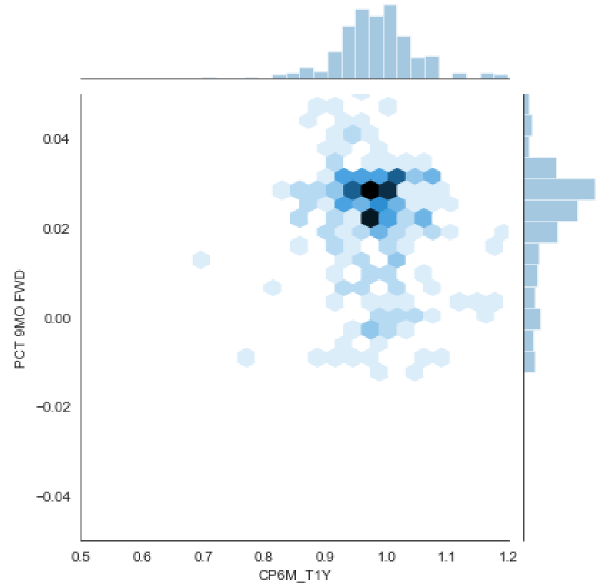
From the previous heatmap, we can see that the feature "CP1M" has the strongest negative correlation with our target variable, "PCT 9MO FWD". Therefore, we construct the scatter plot for these two variables, with x-axis showing the data for CP1M, and y-axis showing the data for PCT 9MO FWD.



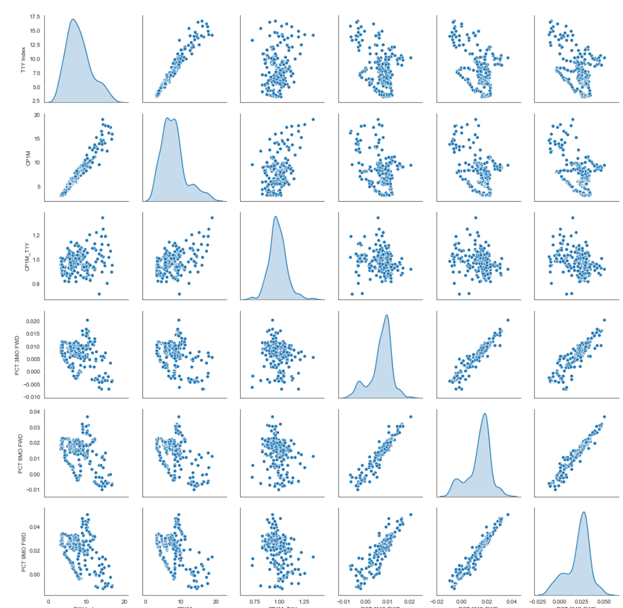
Also, we would like to see the correlation between the US Treasury bond and the commercial paper. Therefore, we find the one with the highest correlation to plot on the scatter plot, with x-axis showing the data for "T1Y Index", and y-axis showing the data for "CP6M".



Since CP6M_T1Y has the weakest negative correlation with our target variable, we construct the hex joint plot to have a closer look for the relationship between "CP6M_T1Y" and "PCT 9MO FWD".

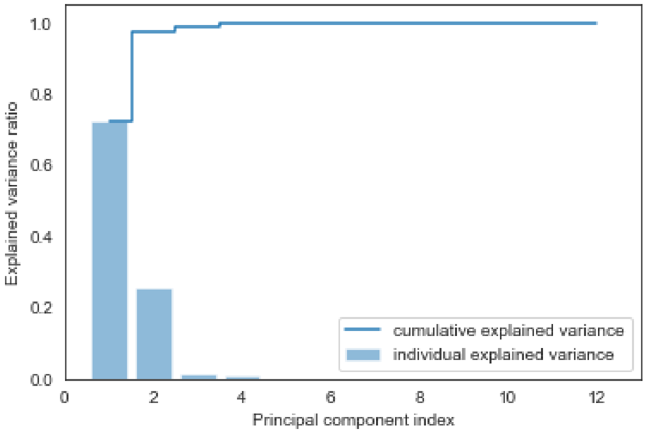


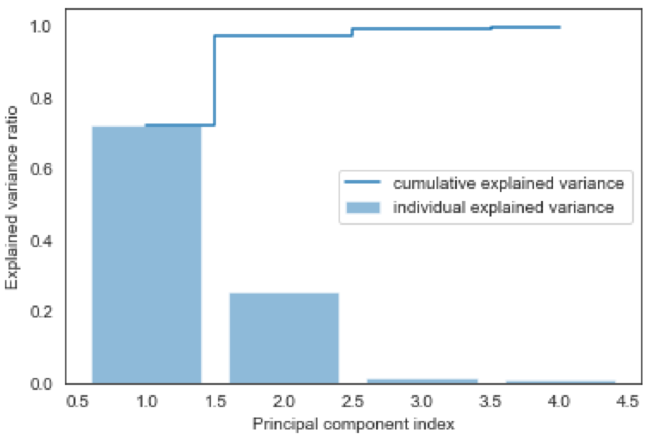
The following is the pair plot of the Treasury Bond and commercial paper with the shortest maturity time, and the percentage changes in USHPCI index. We can see that percentage changes have strong positive correlation between themselves. However, others have shown negatively correlated or no strong correlations.



II. Preprocessing

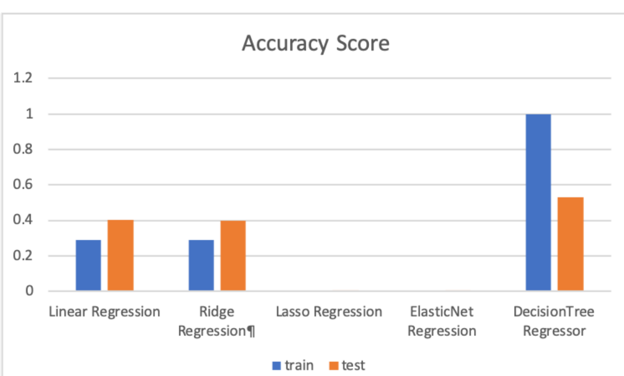
We first did the test train split, and the test size is 0.15. We used a stratified split, so that we can ensure that test set and train set have the same proportional negative y values. We then used the standard scaler to standardize the independent variables. After that, we did the visualization of explained variance and decided that we should use 4 components in pca. We can see that the cumulative explained variance of 4 components is almost 100%.





III. Model Fitting and Evaluation

LINEAR regression, penalized linear regressions and decision tree regression are tried to model our data and we can clearly see that decision did a better job. What is worth to notice is that linear and ridge outperformed lasso and elasticnet because the independent variables have already been preprocessed with pca.

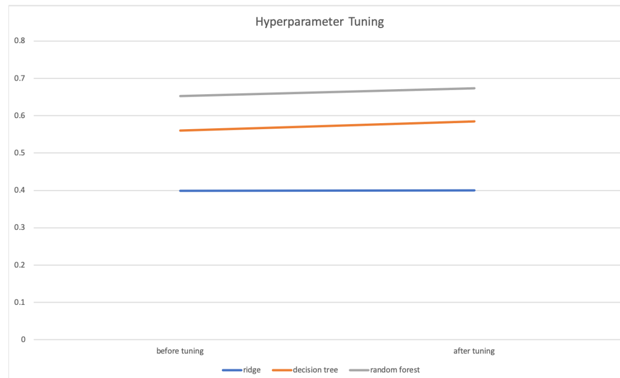


IV. Ensembling

RANDOM forest regressor was tried and it did a much better job than other models without any surprise. The in-sample accuracy is 91.84% and the out-sample accuracy is 65.23%. We have to admit that it is hard to overcome the problem of overfitting when the sample size is small.

V. Hyperparameter Tuning

HYPERPARAMETER tuning was performed for ridge, decision tree and random forest. We can see that the hyperparameter tuning of these three models can improve the model performance. The best alpha for the ridge regression is 0.99; the best max depth of decision tree model is 30; the best number of estimators for regression tree model is 150.



Conclusion

After we use standard scaler and pca to transform the independent variables, it is not necessary to use penalized regression. We can clearly see that the random forest is a much more powerful tool than other simple models and it can dramatically improve the performance. The data size is so small that it is hard to eliminate the overfitting problem although we use both bootstrap and cross validation.

Resources

In order for others' convenience, I have uploaded the code file on my gitub:https://github.com/Qlu10/IE598. I am looking forward to further discuss the code with you!

Acknowledgement

We very appreciate that Professor Matthew Murphy offers us industrial insights to apply effective algorithms and helpful suggestions to build article framework.

Datacamp courses really help us to move forward in both theoretical analysis and practical programming, without which the results would be unattainable.

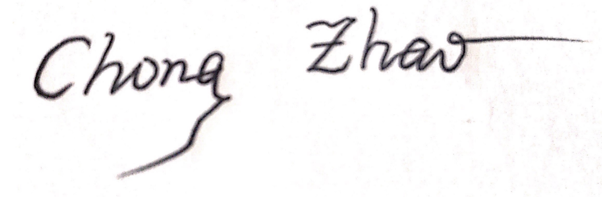
We also appreciate Market Axes who provides us with the valuable opportunity to face real world challenge.

Thank you again!

Academic Integrity

I hereby certify that I have read the University policy on Academic Integrity and I am not in violation of Academic Integrity of University of Illinois at Urbana-Champaign.

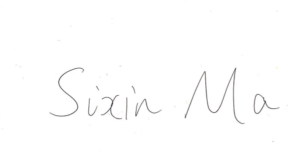
Signature:



Master Candidate in Financial Engineering
University of Illinois at Urbana-Champaign

Master Candidate in Financial Engineering
University of Illinois at Urbana-Champaign

Master Candidate in Financial Engineering
University of Illinois at Urbana-Champaign

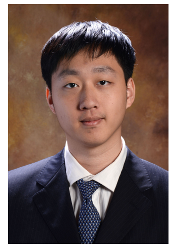


Master Candidate in Financial Engineering
University of Illinois at Urbana-Champaign

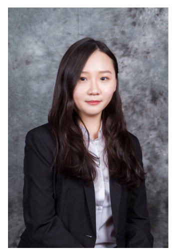
Chong Zhao Master Candidate in Financial Engineering in University of Illinois at Urbana Champaign, IL, USA. Chong Zhao received the B.S. degree in physics from University of Chinese Academy of Sciences, Beijing, PRC, in 2019.

Having interned in various financial and technological companies, Chong is experienced in quantative analysis by python, C++ and R. He is especially passinate about financial derivative pricing and risk management.

Currently, he is looking for 2020 summer internship and full time job after December 2020 in commercial banks, investment banks, insurers and hedge funds. He is open to any global opportunities.



Qiuchen Lu Qiuchen Lu is a graduate student in financial engineering and he completed his bachelor's degree in actuarial science, both at the University of Illinois at Urbana-Champaign.
He is both a market enthusiast and professional engineer. He is a CFA level 2 candidate and he can program in C++, Python, Matlab and R really well. He is actively looking for summer 2020 internship in the investment and trading industry. He is an extremely fast learner and is willing to learn anything to fit for the position.



Yu Chi Chen Yu-Chi Chen is a first-year master's student majoring in Financial Engineering, and she earned her bachelor's degree in Actuarial Science, both at University of Illinois at Urbana-Champaign. She has passed two actuarial exams and is currently pursuing to attain the Associate of the Society of Actuaries. With her academic background and internship experience, she developed her programming skills in Excel, Python, R and C++.
She is actively looking for Summer 2020 intern positions in actuarial science and finance industry. She is excited to learn anything and willing to relocate.

PLACE
PHOTO
HERE

Sixin Ma Sixin Ma, master in Financial Engineering in University of Illinois at Urbana Champaign, received the B.S. degree in mathematics and statistics from University of Illinois at Urbana Champaign in 2019.
Sixin is specialized in R and python programming of data analysis and also familiar with c++ and has interned in some financial companies doing data analysis and industrial research. Besides, she did research by using statistical methods during undergraduate.