Final Paper

Professor: Johanna Devaney

Student: Ethan Lo

CUNY Graduate Center

Data Analysis and Visualization

DATA 71200 Advanced Data Analysis Methods (Summer 2022)

Contents

[1) Describe your data set, why you chose it, and what you are trying to predict with it 3](#_Toc108120803)

[2) Detail what you did to clean your data and any changes in the data representation that you applied. Discuss any challenges that arose during this process. 3](#_Toc108120804)

[General Description 3](#_Toc108120805)

[Value Counts 4](#_Toc108120806)

[Data Cleaning 4](#_Toc108120807)

[3) Discuss what you learned by visualizing your data 4](#_Toc108120808)

[DataFrame.hist 5](#_Toc108120809)

[plotting.scatter\_matrix() 6](#_Toc108120810)

[Scatter Plot 7](#_Toc108120811)

[Heatmap 8](#_Toc108120812)

[4) Describe your experiments with the two supervised learning algorithms you chose. This should include a brief description, in your own words, of what the algorithms do and the parameters that you adjusted. You should also report the relative performance of the algorithms on predicting your target attribute, reflecting on the reasons for any differences in performance between models and parameter settings. 8](#_Toc108120813)

[k-Nearest Neigbors 9](#_Toc108120814)

[Support Vector Machines 11](#_Toc108120815)

[5) Describe your experiments using PCA for feature selection, discussing whether it improved any of your results with your best-performing supervised learning algorithm. 13](#_Toc108120816)

[6) Discuss the results of using PCA as a pre-processing step for clustering. This should include a brief description, in your own words, of what the algorithms do. If you used the Wine dataset for this, briefly explain why your original dataset wasn’t appropriate. 14](#_Toc108120817)

[7) Summarize what you learned across the three projects, including what you think worked and what you would do differently if you had to do it over. 15](#_Toc108120818)

[Appendix 16](#_Toc108120819)

## Describe your data set, why you chose it, and what you are trying to predict with it

My data set is “Company Bankruptcy Prediction” from Kaggle.com (<https://www.kaggle.com/datasets/fedesoriano/company-bankruptcy-prediction>). There are three reasons why I have chosen the data set; first, The data were collected from the Taiwan Economic Journal for the years 1999 to 2009. Company bankruptcy was defined based on the business regulations of the Taiwan Stock Exchange. As a Taiwanese I have to honor their hard work. Second, Since the data set has only binary labels, and it will be used for predicting both supervised and unsupervised experiments. It will be more convenient for my processed. Last but not least, it has 95 features and 6819 rows, perfect amounts for both supervised and unsupervised learning; A large enough number of features to training effectively, but there won't be too many rows causing training to take too much time.

The goal of predicting is a company will Bankruptcy or not.

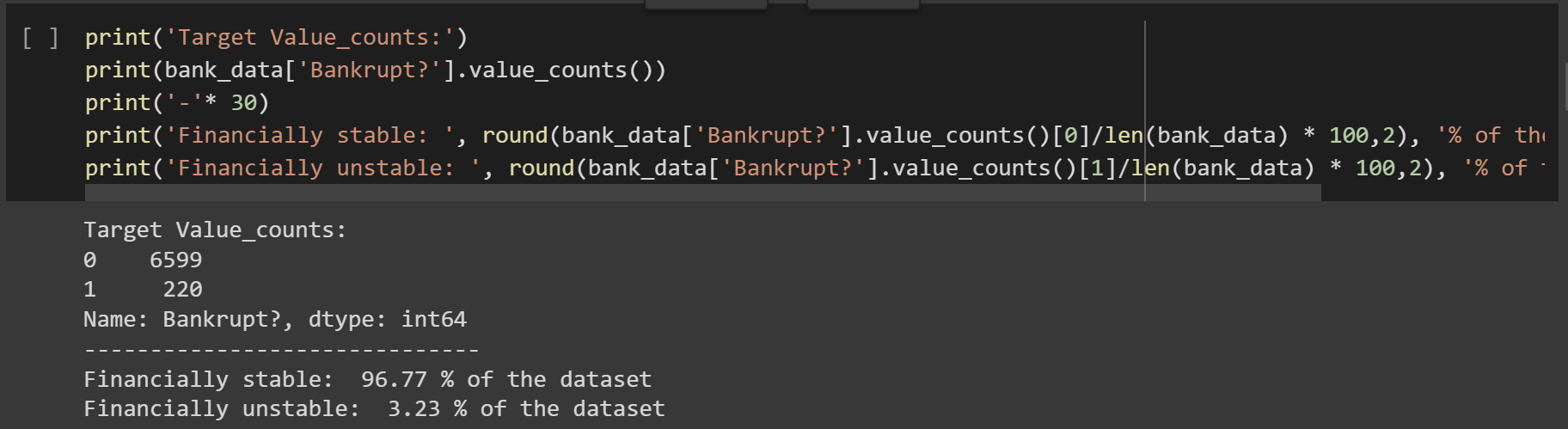
## 2) Detail what you did to clean your data and any changes in the data representation that you applied. Discuss any challenges that arose during this process.

### General Description

Use scikitlearn to divide the data set into training and testing sets. Make sure that the testing and training sets are balanced in terms of target classes. The dataset is composed of a combination of 6819 observations per each of our 96 features. All of the features are numerical (int64 or float64). There are no missing values (Nan) among the data. The Class label is "Bankrupt?", 0 is fine, 1 is bankrupt. The detail of the data set is in the Annexs.

### Value Counts

Looking at the result, it is clear to see how the labels are strongly unbalanced. We got 6599 rows are Financially stable, only 220 rows are Financially unstable.

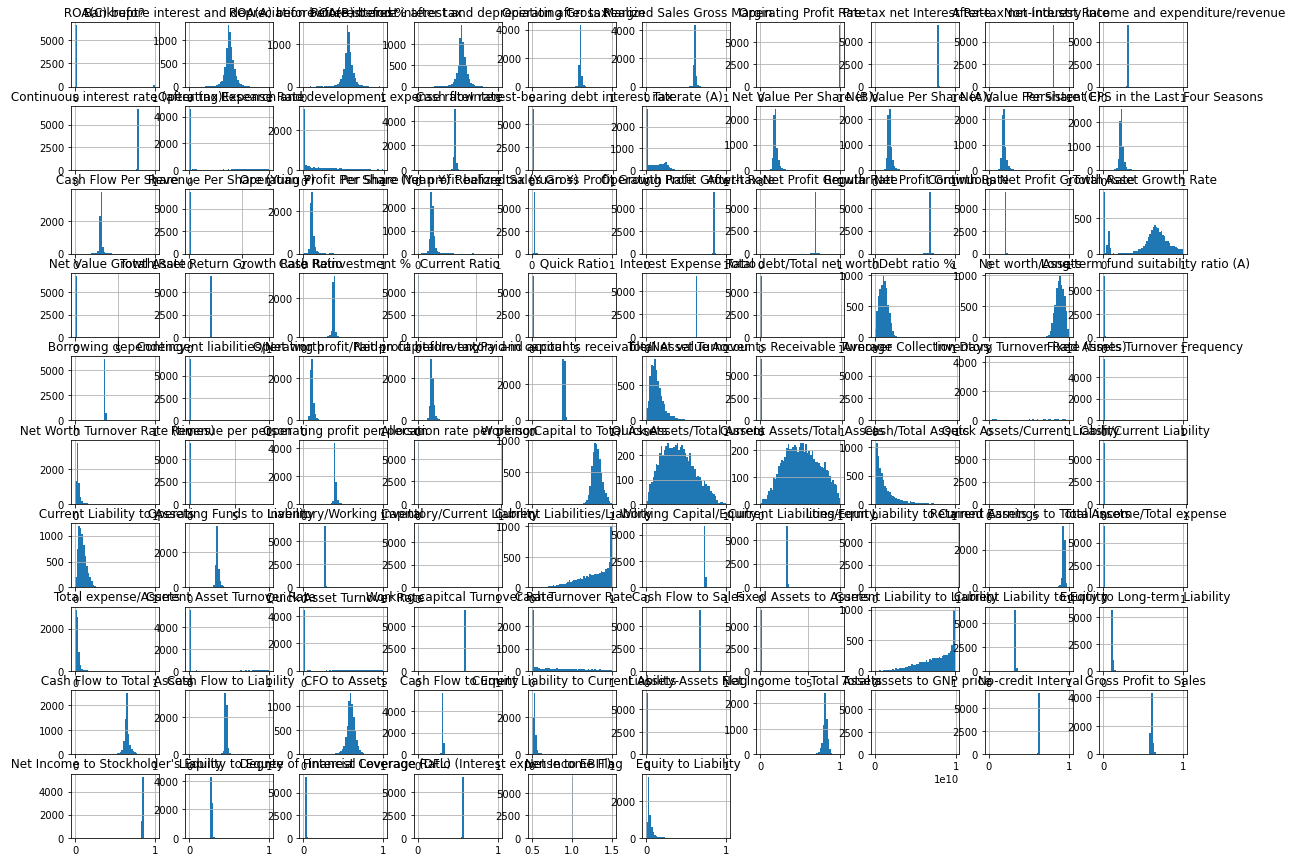


### Data Cleaning

Although we already know that there are no missing values, it is important to computationally check that this is true. I used DataFrame.dropna() function to make sure no missing values in my data set. In the end, used DataFrame.duplicated().sum() to make sure there has no duplicated in the data set.

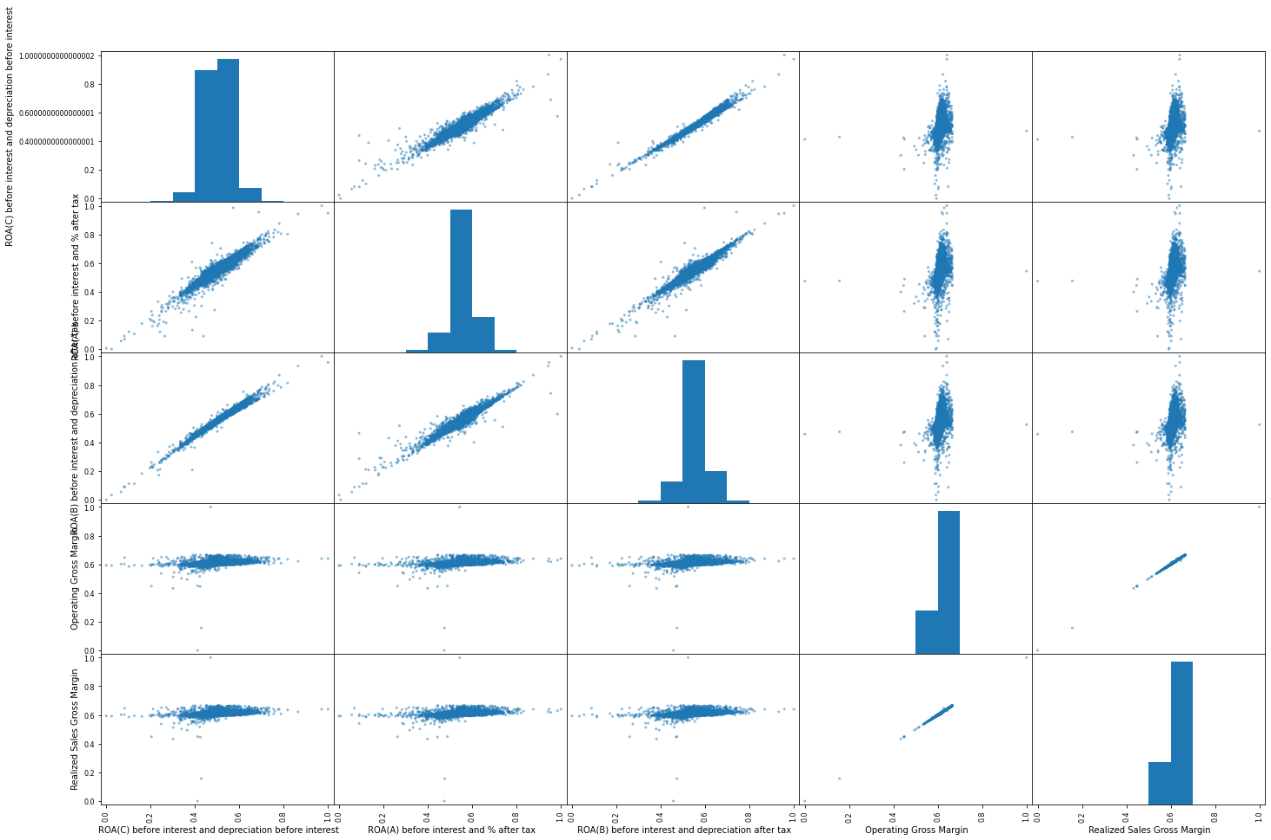
## 3) Discuss what you learned by visualizing your data

### DataFrame.hist



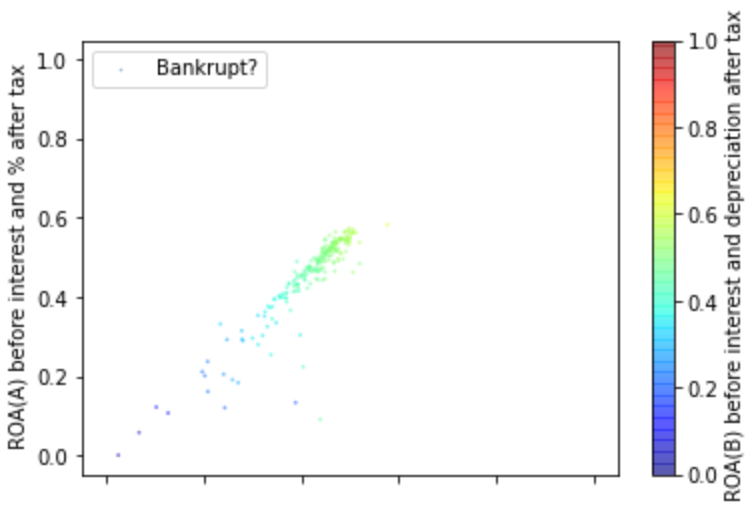
The features in the dataset have significant similarities, probably related to me not standardizing beforehand.

### plotting.scatter\_matrix()



Only do plotting.scatter\_matrix() in the first 5 columns, because use all 96 columns will take forever. Most of the data are normally distributed, so using standardization would be a good preprocessing method.

### Scatter Plot



The features be selected are x=' ROA(C) before interest and depreciation before interest', y=' ROA(A) before interest and % after tax' There is a strong positive correlation between the two. But for the prediction of the model, it doesn't seem to be very helpful.

### Heatmap

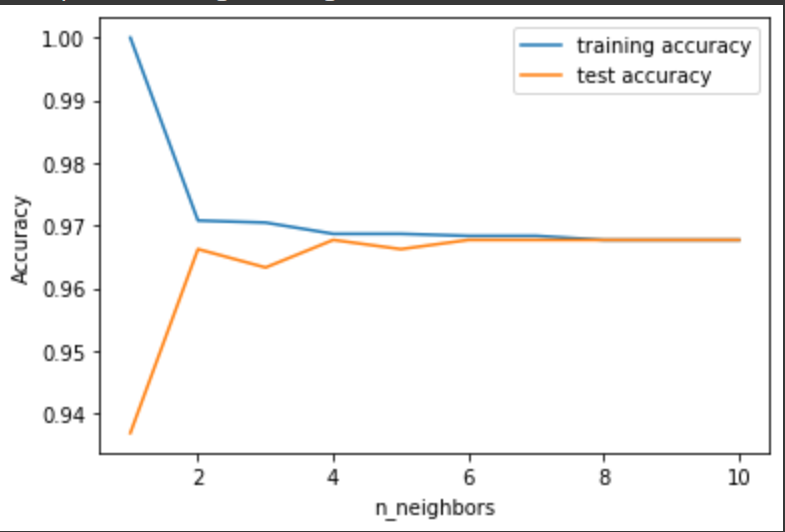


According to the heat map, it can be intuitively found that about half of the data does not have a strong correlation for prediction. The evidence is we only requered 53 features in PCA () to acchive 95% of Variance Explained.

## 4) Describe your experiments with the two supervised learning algorithms you chose. This should include a brief description, in your own words, of what the algorithms do and the parameters that you adjusted. You should also report the relative performance of the algorithms on predicting your target attribute, reflecting on the reasons for any differences in performance between models and parameter settings.

the two supervised learning algorithms has chosen were k-Nearest Neigbors Classification and Support Vector Machines.

### k-Nearest Neigbors

The reseon why I chosen k-Nearest Neigbors was because that the data set is base on binery claissfattion. As far I know, k-Nearest Neigbors usesuly has well performance in those kind of subject. First, I compared different values for k. As the result, after k >= 4, it has not too much impovement for my model.

So I used Cross Validation to test the comment. When “n\_neighbors=4” in 5-fold of Cross Validation, the mean accuracy is 0.9675 which is closs with my 假說. The second think I did is to optimize my model; Used GridSearchCV to find the best params. As the result, when {'metric': 'chebyshev', 'n\_neighbors': 9, 'weights': 'distance'} the model has the bast performance. In the test data set, the model got accuracy of best performing params 0.9679.

For Evaluation Metrics:

In KNN model:

Test set R^2: 0.96

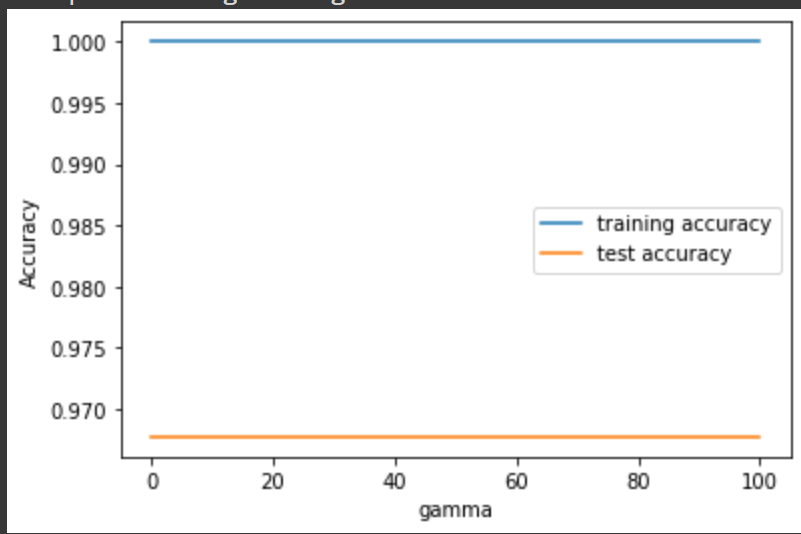
Test set RMSE: 0.04

Test set MSE: 0.04

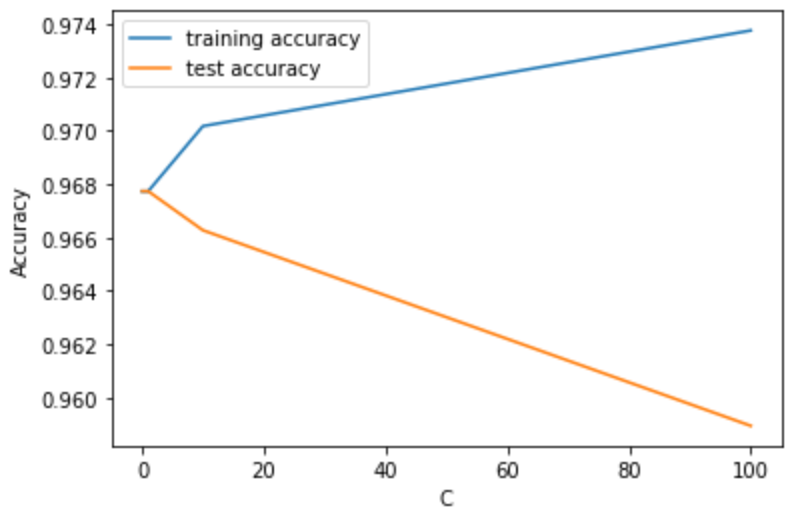
Test set F1: 0.00

### Support Vector Machines

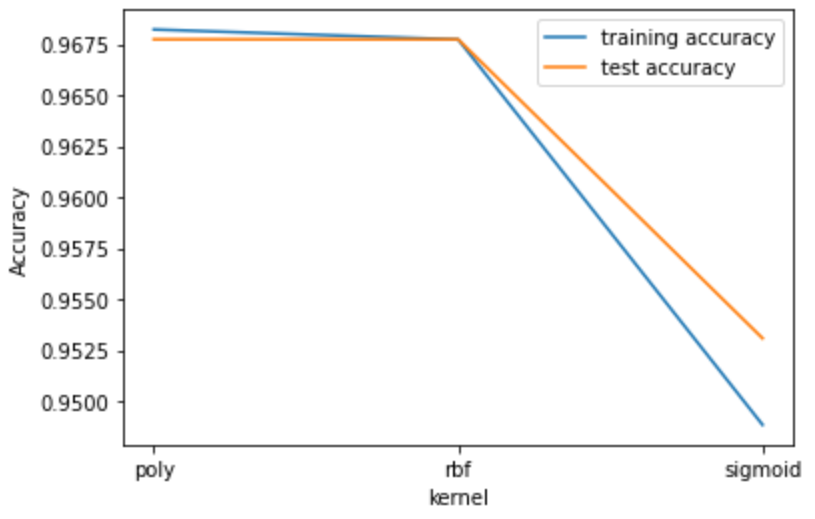
The reseon why I chosen Support Vector Machines was because that the data set has 95 features whick is a high dimaintion data set. In the high dimaintion維度 situation, Support Vector Machines is the perfect way to solve the problem. Fisrt, I compared different values for gamma, I tried gamma = [0.001, 0.01, 0.1, 1, 10, 100], not much difference in this case.



Second I compared different values for C, C\_settings = [0.001, 0.01, 0.1, 1, 10, 100], By the result, when C was growing, the accuracy getting worse.



Finely, I compared different values for kernel, In the kernel, if used linear couldn't run. In the plot, it's easy to see “poly” and “rbf” have better performance than sigmoid in this data set.



For optimized my model; Used GridSearchCV to find the best params. As the result, when {'C': 0.01, 'gamma': 0.01, 'kernel': 'rbf'} the model has the bast performance, The best accuracy is 0.968 in the test data set. In evaluation metrics:

In SVC model:

Test set R^2: 0.97

Test set RMSE: 0.03

Test set MSE: 0.03

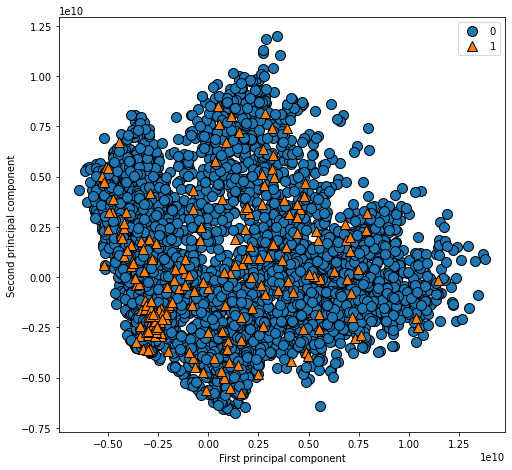
Test set F1: 0.00

Since the data set has the 95 dimestion(維度), Support Vector Machines has better performance than k-Nearest Neigbors Classification was totally be excepted.

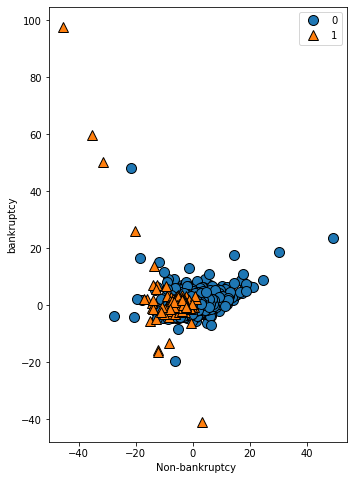
## 5) Describe your experiments using PCA for feature selection, discussing whether it improved any of your results with your best-performing supervised learning algorithm.

PCA() doing the great job to save my time. To retain to capture 95% of the variance, only required 53 features which are almost half of the whole 95 features. This can be seen in my k-Nearest Neigbors model. We can intuitively see that after passing through PCA(), when using the same hyperparameters, the accuracy of the model has indeed been improved, from 0.9675 to 0.9677. However, in Support Vector Machines, PCA() has no effect, the reason is because Support Vector Machines it already contains the function of dimensionality reduction.

## 6) Discuss the results of using PCA as a pre-processing step for clustering. This should include a brief description, in your own words, of what the algorithms do. If you used the Wine dataset for this, briefly explain why your original dataset wasn’t appropriate.

PCA effectively reduces the complexity of the model, and taking half of the features in my dataset can explain almost 95% of the dataset. In addition to improving the speed of training, this also reduces the interference of noise. And scaled data or unscaled data has very significant difference. This graph is unscaled data and is basically completely unclassifiable.

This graph is scaled data and it looks good on classifiable.



## 7) Summarize what you learned across the three projects, including what you think worked and what you would do differently if you had to do it over.

I wasn't fine-tuned enough in the data processing this time, although the data it has been processed, but next time I hope to remove the deviation from the mean. Then I want to standardize the data and then visualize it, maybe I can observe more interesting results. This time, I only used k-Nearest Neighbors and Support Vector Machines. Next time, I will want to add random forest to see if it can be better than the current model. And since I didn't expect PCA to have no effect on Support Vector Machines, I'll try to replace Support Vector Machines with another model next time.

## Appendix

|  |  |  |
| --- | --- | --- |
|  | Column | Dtype |
| 0 | Bankrupt? | int64 |
| 1 | ROA(C) before interest and depreciation before interest | float64 |
| 2 | ROA(A) before interest and % after tax | float64 |
| 3 | ROA(B) before interest and depreciation after tax | float64 |
| 4 | Operating Gross Margin | float64 |
| 5 | Realized Sales Gross Margin | float64 |
| 6 | Operating Profit Rate | float64 |
| 7 | Pre-tax net Interest Rate | float64 |
| 8 | After-tax net Interest Rate | float64 |
| 9 | Non-industry income and expenditure/revenue | float64 |
| 10 | Continuous interest rate (after tax) | float64 |
| 11 | Operating Expense Rate | float64 |
| 12 | Research and development expense rate | float64 |
| 13 | Cash flow rate | float64 |
| 14 | Interest-bearing debt interest rate | float64 |
| 15 | Tax rate (A) | float64 |
| 16 | Net Value Per Share (B) | float64 |
| 17 | Net Value Per Share (A) | float64 |
| 18 | Net Value Per Share (C) | float64 |
| 19 | Persistent EPS in the Last Four Seasons | float64 |
| 20 | Cash Flow Per Share | float64 |
| 21 | Revenue Per Share (Yuan ¥) | float64 |
| 22 | Operating Profit Per Share (Yuan ¥) | float64 |
| 23 | Per Share Net profit before tax (Yuan ¥) | float64 |
| 24 | Realized Sales Gross Profit Growth Rate | float64 |
| 25 | Operating Profit Growth Rate | float64 |
| 26 | After-tax Net Profit Growth Rate | float64 |
| 27 | Regular Net Profit Growth Rate | float64 |
| 28 | Continuous Net Profit Growth Rate | float64 |
| 29 | Total Asset Growth Rate | float64 |
| 30 | Net Value Growth Rate | float64 |
| 31 | Total Asset Return Growth Rate Ratio | float64 |
| 32 | Cash Reinvestment % | float64 |
| 33 | Current Ratio | float64 |
| 34 | Quick Ratio | float64 |
| 35 | Interest Expense Ratio | float64 |
| 36 | Total debt/Total net worth | float64 |
| 37 | Debt ratio % | float64 |
| 38 | Net worth/Assets | float64 |
| 39 | Long-term fund suitability ratio (A) | float64 |
| 40 | Borrowing dependency | float64 |
| 41 | Contingent liabilities/Net worth | float64 |
| 42 | Operating profit/Paid-in capital | float64 |
| 43 | Net profit before tax/Paid-in capital | float64 |
| 44 | Inventory and accounts receivable/Net value | float64 |
| 45 | Total Asset Turnover | float64 |
| 46 | Accounts Receivable Turnover | float64 |
| 47 | Average Collection Days | float64 |
| 48 | Inventory Turnover Rate (times) | float64 |
| 49 | Fixed Assets Turnover Frequency | float64 |
| 50 | Net Worth Turnover Rate (times) | float64 |
| 51 | Revenue per person | float64 |
| 52 | Operating profit per person | float64 |
| 53 | Allocation rate per person | float64 |
| 54 | Working Capital to Total Assets | float64 |
| 55 | Quick Assets/Total Assets | float64 |
| 56 | Current Assets/Total Assets | float64 |
| 57 | Cash/Total Assets | float64 |
| 58 | Quick Assets/Current Liability | float64 |
| 59 | Cash/Current Liability | float64 |
| 60 | Current Liability to Assets | float64 |
| 61 | Operating Funds to Liability | float64 |
| 62 | Inventory/Working Capital | float64 |
| 63 | Inventory/Current Liability | float64 |
| 64 | Current Liabilities/Liability | float64 |
| 65 | Working Capital/Equity | float64 |
| 66 | Current Liabilities/Equity | float64 |
| 67 | Long-term Liability to Current Assets | float64 |
| 68 | Retained Earnings to Total Assets | float64 |
| 69 | Total income/Total expense | float64 |
| 70 | Total expense/Assets | float64 |
| 71 | Current Asset Turnover Rate | float64 |
| 72 | Quick Asset Turnover Rate | float64 |
| 73 | Working capitcal Turnover Rate | float64 |
| 74 | Cash Turnover Rate | float64 |
| 75 | Cash Flow to Sales | float64 |
| 76 | Fixed Assets to Assets | float64 |
| 77 | Current Liability to Liability | float64 |
| 78 | Current Liability to Equity | float64 |
| 79 | Equity to Long-term Liability | float64 |
| 80 | Cash Flow to Total Assets | float64 |
| 81 | Cash Flow to Liability | float64 |
| 82 | CFO to Assets | float64 |
| 83 | Cash Flow to Equity | float64 |
| 84 | Current Liability to Current Assets | float64 |
| 85 | Liability-Assets Flag | int64 |
| 86 | Net Income to Total Assets | float64 |
| 87 | Total assets to GNP price | float64 |
| 88 | No-credit Interval | float64 |
| 89 | Gross Profit to Sales | float64 |
| 90 | Net Income to Stockholder's Equity | float64 |
| 91 | Liability to Equity | float64 |
| 92 | Degree of Financial Leverage (DFL) | float64 |
| 93 | Interest Coverage Ratio (Interest expense to EBIT) | float64 |
| 94 | Net Income Flag | int64 |
| 95 | Equity to Liability | float64 |
| dtypes: float64(93), int64(3) | | |