A Glimpse of Machine Learning in Finances

CS 4120 — Machine Learning, Data Mining
David Thompson and Ali Raisolsadat
University Of Prince Edward Island

March 20, 2020

Abstract

The financial and insurance industries are constructing a large amount of historical data. These data, labeled and not labeled, are hard to interpret by the already existing models in computational finance. This difficulty is due to unexpected shocks to the economy and firms in recent years, unruliness, and fractiousness of the financial markets. The goal of financial institutions by providing this massive amount of data is, of course, creating models that can predict the future position of investment portfolios. The recent discoveries in machine learning and neural networks have given a reason to financial institutions to look past the theoretical models, which fail in extreme instances of market behavior and consider models that can create financial portfolios based on raw data without human behavior, altering the investment decisions, and lowering the amount of risk associated with these portfolios.

Financial ratios are one of the oldest methods that are still in use for investing decisions by investors and financial professionals. They provide figures on how a firm is operating in a period. These operating specifics provide information for the investor resulting in a decrease in the amount of risk associated with the health of a particular company. These ratios are vaguely defined and, in some instances, give false information to the investor, but they impact investor's behavior and decisions significantly. Therefore we decided to look at an alternative way to get this information without the use of ratios. We looked at the raw financial data provided for companies and used features that encapsulated the ratios. We used the clustering method to gain information about the health of all the companies listed in our data and created a portfolio that considered the most healthy companies. We used a sample of the above portfolio to create a multiple univariate regression model to predict the future expected return on the sample portfolio.

Contents

1. Introduction

Investing is one of the most exciting ways to earn money. Investing is a way of expanding our wealth by spending some of the savings we have in financial markets with the expectation of some profit. However, this is not as easy as it sounds. Profits can be affected by many factors that arise from financial markets.

One of the most important and the most famous financial markets that people invest in is the stock market. The difficulty of investing in the stock market is calculating the amount of risk. Risk is the uncertainty on the amount of return (profit or loss) of a stock. Many mathematicians and economists have come up with models that try to enclose the amount of risk in financial portfolios. Some of these models work great to some extent, but they fail to measure the total amount of risk associated with portfolios. That is why financial professionals try to reduce the amount of uncertainty and risk.

Diversification of a portfolio is one of the most important ways to reduce the amount of risk. Diversification is a strategy that mixes a variety of financial instruments in a particular portfolio, intending to reduce the amount of return on a portfolio to zero or a positive number. Another way of diversifying a portfolio is by making sure that an investment in a company can generate a positive return on a portfolio. We need to look at a company and decide if this company can survive or thrive in the given conditions of the market. Financial professionals use financial ratios that can be calculated from the accounting information of a company to decide whether a company is doing well enough for possible future investments. However, the accuracy of these ratios is questionable. Some of these ratios give a definite answer, such as debt to asset ratio, but some are open to interpretation. Professionals need to interpret these ratios and decide on an investment. What if the investor interprets these results wrong? We can see that this increases risk. We can also question the validity of these ratios.

In recent years there been an increase in algorithmic trading and the use of machine learning in computational finances. Financial industries are moving away from human-based decisions regarding investments, which is full of fault and move towards a more stable decision making for their financial portfolios. This way, financial industries can reduce the risk associated with human behavior and let the machine make intermediate investment decisions.

Out motivation was to create an algorithm that uses raw financial data, instead of ratios and put the decision making of creating a financial portfolio on this algorithm. That is an algorithm that can identify "healthy" and "unhealthy" companies without using troublesome ratios. With this algorithm, we can significantly decrease the amount of risk associated with human error and create a semi diversified portfolio, denoted as V_T. We also created another algorithm to let us predict future expected returns on a sample portfolio, denoted as ϕ_T. What follows is a description of our data and our model analysis.

2. Data Description

The data was taken from the Kaggle website for this project. The data folder contains 5 years of financial data (2014- 2018) in the format of ".csv" files. Each year's data set contains a list of 4500+ NYSE companies and the 200+ financial indicators that can found on the 10K filings of these companies at the end of each year. The indicators in this data are essential to investors, shareholders, and stakeholders because it gives vital information about the health of the company. An investor may incur huge losses if the health of a company is terrible and has a negative economic profit.

The data provided for each year is very comprehensive. There are numerical and categorical features included in our each year's data set. The critical thing to remember is that some of these features are more important to investors than others. Therefore with the use of feature reduction, we only consider numerical features that are considered most relevant to a future investor. With this technique, we can make sure our model decides the healthiness of a company using the essential information.

There are few problems that we encountered with this data. One problem is the missing information. Companies prefer not to share numbers with shareholders, or there is redacted information from accounting statements due to competition in the market. There is also a chance that the individual creating the data sets did not enter the information for a specific company. Since it is crucial to know fully about a company before any investments, we decided to discard companies lacking the features we had in mind. Discarding these companies reduced the number of training examples for our K-Means algorithm, and left

us with 2021 companies as our training examples.

3. Background

The main objective of creating every financial portfolio is to lower risk (market uncertainty) as much as possible. That way, we can be sure of a specific amount of return on our portfolio. Our approach for this project is simple. First, We like to lower the amount of risk associated with human decisions and poorly defined financial ratios, and then try to predict the amount of return on a sample portfolio. With our given data, we like to create a portfolio V_T, which only contains the stocks of healthy companies. Financial professionals use financial ratios to do so. However, since these ratios are questionable and professionals can misinterpret them, we use the exact data to differentiate between healthy and non-healthy companies. We then create a small sample portfolio, ϕ_T, from our V_T, and use a regression model to evaluate the expected return for each year.

In a competitive industry, a company lives on when economic profit is either zero or positive. Economic profit is the difference between revenue and the costs of all production inputs and opportunity costs. Our data only considers the accounting profit. Therefore our K-Means algorithm used to construct V_T only considers the health of a company using the accounting profit, which is troublesome since we won't know for definite if the company is all healthy. However, since accounting profit considers everything but opportunity cost, we can say that this algorithm reduces risk.

Nevertheless, economic profit can affect stock prices. So our regression model can accumulate risk since our data does not have features considering opportunity costs. The following is a brief description of the techniques and algorithms used in this project. The full implementation of these techniques was submitted previously.

3.1 K-means Clustering

Since this data set contains many companies, it follows that we have many training examples. These training examples, however, do not have any labels. Consequently, we decided to utilize the K-means clustering algorithm [1]. We also have seen other clustering methods, such as Density-based spatial clustering of applications with noise (DBSCAN), but since the clusters have the same distribution, DBSCAN won't perform well. A disadvantage of using the K-means clustering method, however, is not knowing the optimum K value. Therefore we implement the "Elbow Method" to find our usable K-values.

3.2 Elbow Method

The "Elbow Method" is one way to validate the number of clusters we can use in a K-Means algorithm [6]. This method iterates through potential K-values. It also uses the sum of squared error (SSE) as a cost function to evaluate the cost value corresponding to the iterating K-values. Once we have both, we graph the K-values against cost values.

Once graphed, we can visually spot the "elbow" of the graph, which represents the point where the shape of the graph begins to decrease at a linear rate. In our analysis, this method used more than once to find an optimal K-value. From these results, we decided to use two K-values for our clustering algorithm.

3.3 Multiple Univariate Regression

After creating a portfolio of companies, it is helpful to create an algorithm that uses historical returns on the stock of these companies and try to predict future returns. We used a multiple linear regression, which takes a sample portfolio, ϕ_T, as training examples, with features used in the K-Means algorithm and historical returns as our labels (target values). Since we had a linear regression with small training examples, we used the normal equation (as seen below) to optimize the parameters of the model instead of using gradient descent:

$$\theta = (X^T X)^{-1} X^T y$$

where θ is our optimized parameters, X is our training examples (financial indicators for each company) and y is our labels (yearly stock returns for that company). This way, we can let the model learn how to predict future returns on our ϕ_T portfolio. We use root mean square error (RMSE), to calculate the error between the real returns and our predictions.

3.4 RMSE

The root mean square error is the standard deviation of residuals (prediction errors). These prediction errors tell us how far the linear regression lies from the data points. So, RMSE measures how spread out these errors are in the model. The formula for the RMSE [4]:

$$\text{RMSE} = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

4. Results and Interpretations

4.1 Feature Table

As mentioned previously, we have around 200+ financial indicators available to use as potential features. This amount of features is too high of a number and could, therefore, lead to over-fitting the data. Hence, we reduce the number to 17 significant features. After dropping NaN data, our number of training examples decreases by roughly 50%.

	Symbol	Revenue	Revenue Growth	Gross Profit	Operating Income	Earnings before Tax	Free Cash Flow	Net Income	Total current assets	Operating Expenses	Net Debt
0	CMCSA	9.450700e+10	0.1115	9.450700e+10	1.900900e+10	1.511100e+10	1.198500e+10	1.173100e+10	2.184800e+10	7.549800e+10	1.080000e+11
1	KMI	1.414400e+10	0.0320	6.856000e+09	3.794000e+09	2.196000e+09	2.119000e+09	1.609000e+09	5.722000e+09	3.062000e+09	3.404400e+10
2	INTC	7.084800e+10	0.1289	4.373700e+10	2.331600e+10	2.331700e+10	1.425100e+10	2.105300e+10	2.878700e+10	2.042100e+10	1.470900e+10
3	MU	3.039100e+10	0.4955	1.789100e+10	1.499400e+10	1.430300e+10	8.521000e+09	1.413500e+10	1.603900e+10	2.897000e+09	-2.163000e+09
5	BAC	9.124700e+10	0.0446	9.124700e+10	3.786600e+10	3.458400e+10	3.952000e+10	2.814700e+10	7.790000e+11	5.338100e+10	-4.440000e+11
...	...	...	...	...	...	...	...	...	...	...	...
4375	WINA	7.251110e+07	0.0395	6.784070e+07	4.180240e+07	3.928310e+07	1.113870e+07	3.012550e+07	2.425130e+07	2.603830e+07	7.717500e+06
4376	WINS	1.519838e+07	-0.3527	1.496750e+07	9.460981e+06	9.460981e+06	6.499370e+06	1.049988e+07	1.296404e+08	5.506514e+06	-5.380500e+06
4382	WVVI	2.307974e+07	0.1068	1.478150e+07	4.182715e+06	3.939586e+06	-2.959752e+06	2.858580e+06	2.863433e+07	1.059878e+07	-1.383677e+06
4385	XELB	3.546600e+07	0.1186	3.276400e+07	3.930000e+06	2.919000e+06	5.117000e+06	1.088000e+06	2.387500e+07	2.883400e+07	1.073800e+07
4388	YTEN	5.560000e+05	-0.4110	5.560000e+05	-9.274000e+06	-9.170000e+06	-8.796000e+06	-9.170000e+06	6.377000e+06	9.830000e+06	3.640000e+05

2201 rows × 18 columns

Figure 4.1: This table contains a list of 2021 companies with 17 specific features to be used for clustering.

4.2 Elbow graphs

Since many of our features were large dollar values with variance, we chose to normalize the data before clustering. Therefore it is better to normalize the data to reduce variance before the elbow method.

For our first elbow method, we graphed the cost function with a range of 20 clusters using the normalized data. Although this graph gave us the ability to observe K-values versus the cost values, it was tough to locate an elbow in this graph. By looking closely, cluster numbers $K = 3$ and $K = 4$ could be considered as elbows but were not ideal.

For our second elbow method, we made one change. Instead of using only the normalized data, we combined it with six financial ratios calculated from some of our features. When graphed, we were able to spot an elbow at $K = 3$. We used values $K = 4$ and $K = 5$ to make sure that our algorithm is appealing to risk-averse investors. We could have used $K = 3$, but we want the lowest possible amount of risk to ensure we find strictly healthy companies. Hence for our clustering model, When $K = 4$, the clusters will

present companies that are healthy, semi-healthy, not healthy, and unhealthy, and when $K = 5$, we add additional cluster that represents very unhealthy companies.

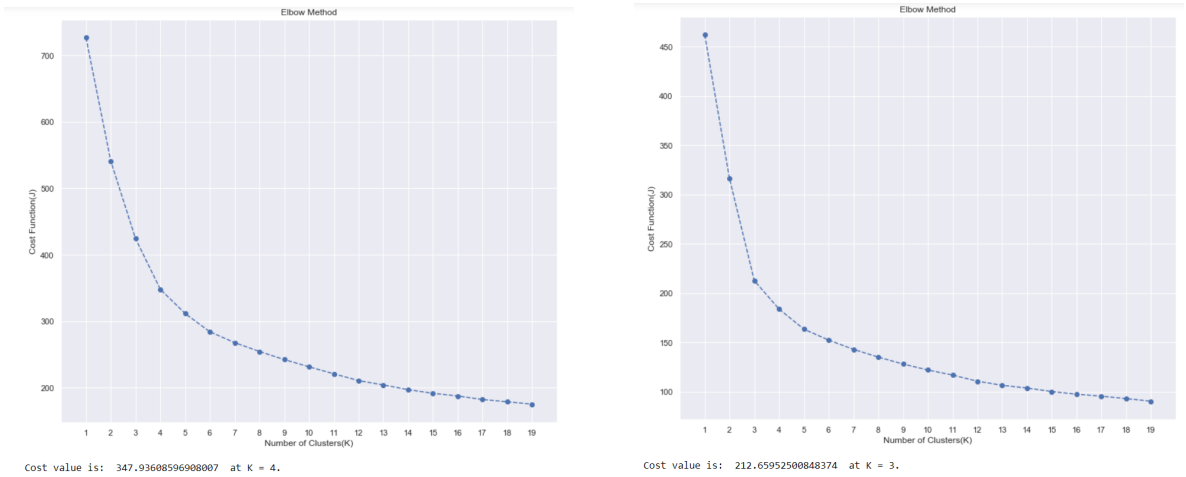

Figure 4.2: The difference between the two elbow graphs with the corresponding cost value.

4.3 Master portfolio

Since we have high dimension data, we cannot examine the clusters visually to choose the best cluster from the K values. Therefore, we need to use a "control" feature, x^v, that represents a good company as an indicator of the best cluster. The selection of the control feature is up to the investor and the industry. For example, monopolies and large corporate companies tend to have extremely high revenues but also a high debt ratio versus their assets. Banks tend to have a high net income because they are not as much as in debt as monopolies are.

Since we like to invest in technology and large companies, we decided to use revenues as our control feature.

It is important to note that high revenue does not necessarily mean a company is healthy. It merely implies that a company's stock would increase during the year. High revenue indicates that a company is managing its wealth very well, has an excellent marketing department, is selling its products or services to costumers, and can pay its outstanding debt at the end of the year. The companies then deduct the cost of goods sold, expenses, and losses to calculate earnings. The calculated earnings used to measure its current price shares relative to its per-share earnings. Therefore when earnings are high, the price of a share (stock) increases.

After running the clustering algorithm, we were able to generate 35+ stocks for our final portfolio V_T, depending on the random state and $x^{(v)}$ control feature. This number is low, but we can say that these companies are healthier than the others. It does not mean that we have a positive return on our investments, which we see later that some of them did not, but there is a higher chance that we will see a positive return with respect to other companies.

We proceeded to create a sample portfolio, ϕ_T. To do this, we picked five random stocks from V_T. We chose the following companies to represent our sample portfolio.

Symbol	Name of the Company	Type
CMCSA	Comcast Corporation	Telecommunications company
T	AT&T. Inc	Telecommunications company
AAPL	Apple. Inc	Technology company
MSFT	Microsoft Company	Technology company
WMT	Walmart Company	Retail company

Table 4.1: Sample portfolio ϕ_T

4.4 PREDICTED Returns vs HISTORICAL Returns

After creating our sample portfolio, we accessed **Yahoo Finances** database for historical information about the stocks in ϕ_T. We used adjusted close prices at the end of the year and opening prices at the beginning of the year to evaluate total return on a stock. Below we can see the adjusted close prices for the our sample portfolio.

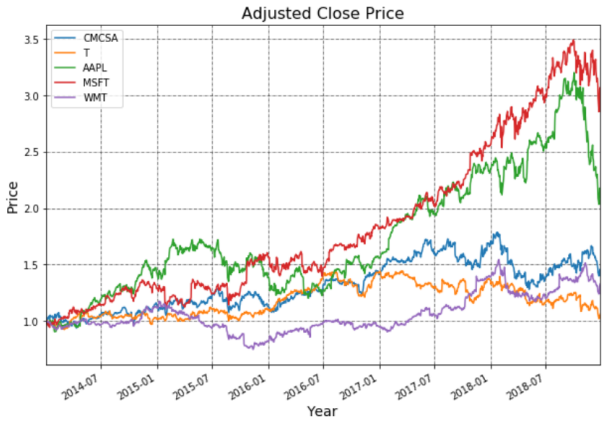

Figure 4.3: Price (in hundreds) vs. Years (2014-2018)

We then proceeded to use this historical information and a regression model. We approached the model in two ways. At first, we used raw feature values. We then normalized our training examples and we ran the regression model for different results. We used the normal equation to optimize our parameters, and we calculated RMSE for the model's predicted results. Please keep in mind that the RMSE values are in the same unit as the predicted values, i.e., if the RMSE value is 30.806, then the expected return on the stock has a standard deviation of almost $\%31$.

We can see the results in figure 4.4 and 4.5. In these figures, we can see the actual historical returns on our sample portfolio versus the predictions of our model. The only difference is that figure 4.5 shows RMSE values for normalized training examples. We can see that some of the RMSE values reduced, but some are now higher than before.

The RMSE values are large due to our reduced amount of training examples. Consequently, this leads to our model being under fitted. If we had access to more training examples, that is, if we had access to daily or monthly company information, then there was a higher chance of creating a more accurate model. Regrettably, this model does not give enough information to make investing decisions.

PREDICTED Returns

	CMCSA	T	AAPL	MSFT	WMT	year	RMSE
0	11.531250	-5.093750	25.562500	16.156250	-12.375000	2014	7.109598
1	-59.556285	-60.224160	-34.707178	-41.399590	-41.148818	2015	30.806531
2	77.994347	11.976519	-8.720534	51.756857	18.154820	2016	29.072128
3	58.820363	15.020561	-47.501074	32.965628	27.529715	2017	33.209450
4	-16.030975	-47.928816	-5.483320	24.351925	-21.545435	2018	12.504246

The predicted return on our portfolio for years 2014-2018: -5.978840368550079

HISTORICAL Returns

	CMCSA	T	AAPL	MSFT	WMT	year
0	1.705207	-28.388810	27.456902	11.344808	-3.731348	2014
1	-10.146865	-18.620589	-11.904417	9.353252	-35.554956	2015
2	18.108101	4.171407	7.569124	8.095585	6.627371	2016
3	9.813918	-20.169875	41.511218	31.603032	36.547809	2017
4	-18.278692	-32.165617	-8.887455	15.891587	-8.055473	2018

The historical return on our portfolio for years 2014-2018: 6.779045025750397

Figure 4.4: Predicted Returns vs. Historical returns for portfolio ϕ_t — Raw features

PREDICTED Returns

	CMCSA	T	AAPL	MSFT	WMT	year	RMSE
0	-1.847168	-3.128906	-2.425781	-2.864258	-3.250977	2014	13.191511
1	-19.199116	-17.207754	-11.906142	-19.368432	-24.416308	2015	9.006978
2	-0.989380	-0.273993	1.926569	0.247624	-1.165041	2016	23.403487
3	-0.325107	-0.119609	0.118516	0.326399	-0.400075	2017	17.511149
4	-0.349382	-0.635419	-0.229981	0.100661	-0.329293	2018	23.002587

The predicted return on our portfolio for years 2014-2018: -21.542470943198403

HISTORICAL Returns

	CMCSA	T	AAPL	MSFT	WMT	year
0	1.705207	-28.388810	27.456902	11.344808	-4.155442	2014
1	-10.146865	-18.620589	-11.904417	9.353252	-35.838850	2015
2	18.108101	4.171407	7.569124	8.095585	6.157665	2016
3	9.813918	-20.169875	41.511218	31.603032	35.946284	2017
4	-18.278692	-32.165617	-8.887455	15.891587	-8.460507	2018

The historical return on our portfolio for years 2014-2018: 6.342194352245596

Figure 4.5: Predicted Returns vs. Historical returns for portfolio ϕ_t — Normalized features

5. Further Work and Conclusion

We saw a glimpse of what we can achieve using machine learning algorithms in finance. These simple models were able to create a diversified portfolio using a large amount of financial data, and use this portfolio to try to predict the future expected returns on these stocks.

With more time and more complicated algorithms, we can create more accurate models. Let us look at some possible future work that can improve this project. Starting at our clustering model, instead of using the elbow method to find suitable K-values, we can use the Silhouette method[5]. This method, which gives a value between +1 and -1, measures the relevance or how similar an object is in its cluster. If the measurement is close to +1, then an object is well grouped. Unlike the elbow method, we can use any distance metrics in the Silhouette method.

After finding our clusters, and portfolio V_T, it is possible to diversify it using the historical returns mathematically. This diversification requires more analysis on portfolio optimization and time.

There is an essential problem with this project, which is the lack of training examples for the regression model. We have a substantial amount of stocks in V_T, but each of these stocks only has five financial instances, which is five years of data from 2014 to 2018. If we had access to more instances, such as weekly or monthly data, then we would be able to decide on whether our predictions indicate possible expected return on our sample portfolio.

Another possible way is to try to figure out whether a stock increases or decreases in the future. A logistic model could be beneficial in finding which stocks increase or decrease. A more rigorous approach would be to create a Convolutional Neural Network. However, none of these models can result in an accurate model without the right amount of training examples.

Bibliography

[1] Bolufe-Rohler, Antonio. (2020). *Unsupervised Learning and K-meansFile* [PowerPoint slides]. Retrieved from http://moodle.upei.ca.

[2] Dabbura, I. (2019, September 3). *K-means Clustering: Algorithm*, Applications, Evaluation Methods, and Drawbacks. Retrieved from http://towardsdatascience.com.

[3] Goodfellow, I., Bengio, Y., Courville, A. (2017). *Deep learning*. Chapter 5: Machine Learning Basics. Cambridge, MA: The MIT Press.

[4] James, G., Witten, D., Hastie, T., Tibshirani, R. (2017). *An introduction to statistical learning: with applications in R*. New York: Springer.

[5] Mahbobi, M., Tiemann, T. K., Tiemann, K., T. (n.d.). Chapter 8. *Regression Basics*.

[6] Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20, 53–65. doi: 10.1016/0377-0427(87)90125-7

[7] Yuan, Chunhui, Yang, Haitao. (2019, June 18). Research on K-Value Selection Method of K-Means Clustering Algorithm. Retrieved from https://www.mdpi.com