Stock Market Analysis

Ankit Jain Rakesh Kumar
University of California Riverside
arake001@ucr.edu

Krupa Hegde
University of California Riverside
khegd001@ucr.edu

Amruta Sawant
University of California Riverside
asawa005@ucr.edu

ABSTRACT

Stock transactions have always been dependent on intuition. However, many prediction algorithms are now trying to emulate the human intuition to get a hold of transactions that should be carried out during purchase of stocks. The stock values are dependent on various external factors that cause fluctuations in the prices over a certain period. Extreme Value Theory (EVT) is a well-established method for computing univariate and multivariate tail distributions that are useful for forecasting stock market risk or modeling the tail dependence of risky assets. We use Extreme Value Theory(EVT) to predict the extreme prices of the stock for the near future. By using EVT we provide an estimate of lower bound and an upper bound to the expected stock price and evaluate if the measured value falls in the range. Furthermore, we use hierarchical clustering and Non-Negative Matrix Factorization(NMF) methods for clustering the stock data. This process helps aggregate similar stock trends and place them into a cluster.

1 INTRODUCTION

Stock market analysis consists of transactions like buying and selling the shares, which constitute to be amongst the largest group of conventions in today's world. A lot of factors affect the entire stock exchange system. The constant fluctuating values of stocks have enabled a lot of research in finding the dependencies between the prices and the parameters on which the values are dependent. The project analyzes the stock market, by using the data collected by crawling share prices from yahoo finance, for over 10 companies from three sectors namely- technology, health care and finance. There are many approaches and data mining algorithms for analyzing stock market. Most approaches used for stock analysis deploy the assumption of data dependency. We try to analyze the stock data with an assumption that the data is independent. In this project we try to analyze the stock data using Extreme Value Theory with our assumption. We use Hierarchical and NMF clustering methods to cluster the stock data.

Figure 1. shows that the stock market is a time series graph, where stock values varies randomly.

2 MOTIVATION

A lot of research has been done in the field of stock analysis. Economists keep researching about the factors affecting the stocks and the changes occurring in their values. Prior works have performed the analysis,based on the concept that the current data of the stock depends upon the previous stock data. However, the idea of price of stock being dependent on its previous value is ambiguous, since, it is based on many factors which contribute to its price. Any base value is calculated only on these factors and hence is not dependent on its antecedent. The percentage increase or decrease can be used to acquire more information,while doing analysis, but,

Figure 1: Time series plot for CITI BANK .

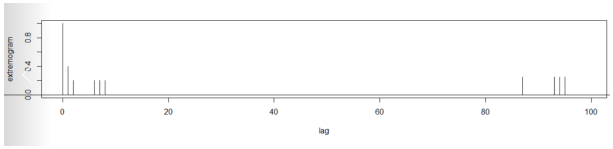

Figure 2: Extremogram of single path time series plot for the AMAZON .

this doesn't account to the dependency between the prior and later values. Our project is hence based on the idea that, though the share prices are dependent on many external constituents,it is not probable for it to be indigent on the prior data.

An extremogram [5] is a measure of extremal dependence for time series measurement. This method contrary to the usual methods for characterizing the dependence of samples, it focuses only on extreme values. Figure.2 shows the extremogram to single case for Amazon. Here the extreme samples of the time series have a limited correlation for 98% confidence intervals. p(h)<0.2. Figure 2 is the main motivation for our project to show that the stock data samples are not dependent on each other.

3 BACKGROUND WORK

This section consists of the literature survey for stock market analysis, and the clustering done on it.

3.1 Literature survey

Many papers published on stock market analysis, depend on the assumption that the stock market values are dependent on the earlier values for that stock. However, we have assumed that the stock prices are dependent on the external factors, which have explicit or implicit effects on the stock, but not on the former value of that stock. Taking this into consideration, we went through many

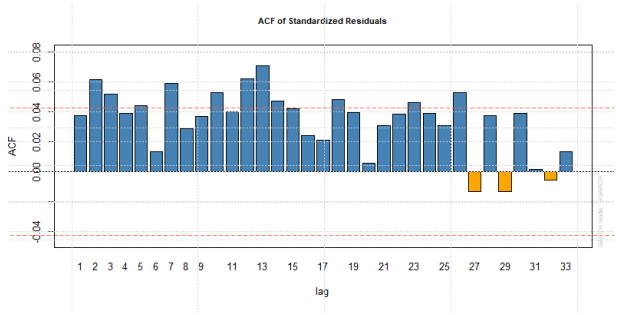

Figure 3: ACF of standardized residuals plot for AMAZON .

papers, which dealt with similar concepts as EVT(for dependent values). In the paper [2, 7] the authors uses a bivariate EVT distribution to show, the dependencies between the Australian stock exchange and the other stock exchange from various other countries. Asymptotic dependencies between the data are calculated and results are given in the form of log return series. X and X'are used to measure the extreme dependencies for measuring the asymptotic dependence of two sets of random variables. They have also used GARCH Filters to work on the dependencies of the two variables, which they term as heteroscedasticity. It is impossible to predict the exact value of stock. There are many approaches but none of them can predict the exact value of stock. There are applications or papers which say if the stock can go high or low. We have not found any papers which give range for the stock value to lie.

Literature survey reveals that k means clustering is not suitable for stock data. Even the density based clustering does not suit financial data set [6]. Hierarchical clustering is one of the clustering algorithm which can be used for analyzing stock data by using efficient metric for measuring the similarity between clusters. With higher DI or with lower DBI, Normalized centroid based clustering gives good number of clusters which is helpful in understanding data classification [6]. Agglomerative hierarchical clustering is suggested in many papers. In one of the paper author suggests GMM and AR(1) clustering are suitable to this field. Most suggested methods for clustering the stock data are NMF clustering and Hierarchical clustering.

3.2 Background work Implementation

The main emphasis of the previous work is been given to the GARCH filters and Conditional EVT, where they calculate the log returns of the stock data for the datasets, which is equal to the log residuals of the stock data. They use GARCH models in the stage one with a view to filtering the return series to obtain nearly iid conditions. In the next stage, the EVT framework is applied to the standardized residuals.

Figure 3. shows that the dependencies of values of the stock residuals (log returns).

Figure 4. shows the log returns for the standardized residuals.

Performing the above GARCH filters on the given stock data gives us few performance analysis metrics shown in the Figure 5.

Once the GARCH filters implementation is done we get the residuals which nearly obey the iid conditions. Furthermore, using

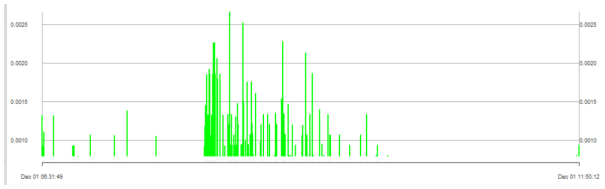

Figure 4: log returns for the standardized residuals plot for AMAZON .

	X1
Observations	2134.0000
NAs	0.0000
Minimum	-0.0027
Quartile 1	-0.0003
Median	0.0001
Arithmetic Mean	0.1532
Geometric Mean	0.1121
Quartile 3	0.0005
Maximum	1.0000
SE Mean	0.0078
LCL Mean (0.95)	0.1379
UCL Mean (0.95)	0.1685
Variance	0.1298
Stdev	0.3603
Skewness	1.9253
Kurtosis	1.7069

Figure 5: Performance metrics for the AMAZON after applying GARCH filters .

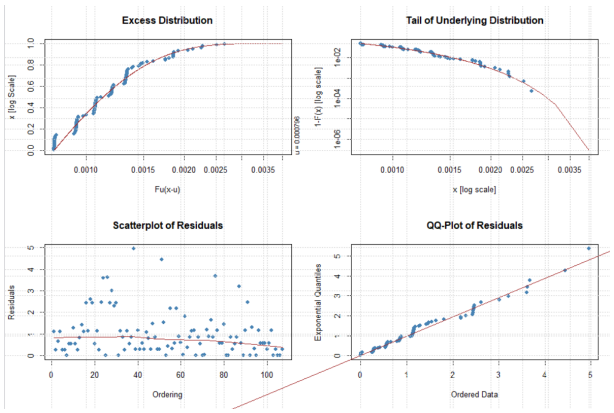

Figure 6: GPD for the stock residuals.

these residuals we apply the GPD (EVT) and the results are not good as seen in the Figure 6.

From the Figure 6. we see that the QQ plot, Excess distribution and the Tail of underlying distributions do not resemble the same distribution.

Hence, the previous work using EVT [1, 4] do not give better results for stock market analysis and also we cannot predict the future values.

4 METHODOLOGY AND IMPLEMENTATION

In this section we explain our methodology and implementation for stock market analysis.

4.1 Data

price	time	change	changePerc	close	oepn	volume	avgVolume	bid	ask	MarketCapp	PERatio	beta	eps
1176.75	12/1/2017 6:29	15.48	(+1.33%)	1161.27	1167.1	4509208	3487830	1176.00x200	1176.30x100	567.043B	299.28	1.42	3.93
1176.75	12/1/2017 6:30	15.48	(+1.33%)	1161.27	1172.05	4509208	3487830	1176.00x200	1176.30x100	567.043B	299.28	1.42	3.93
1176.75	12/1/2017 6:30	15.48	(+1.33%)	1161.27	1172.05	4509208	3487830	1176.00x200	1176.30x100	567.043B	299.28	1.42	3.93
1176.75	12/1/2017 6:30	15.48	(+1.33%)	1161.27	1172.05	4509208	3487830	1176.00x200	1176.30x100	567.043B	299.28	1.42	3.93
1176.75	12/1/2017 6:30	15.48	(+1.33%)	1161.27	1172.05	4509208	3487830	1176.00x200	1176.30x100	567.043B	299.28	1.42	3.93
1174.5	12/1/2017 6:30	-2.25	(-0.19%)	1161.27	1172.05	158014	3487830	1176.00x200	1176.30x100	565.959B	298.7	1.42	3.93
1174.5	12/1/2017 6:31	-2.25	(-0.19%)	1161.27	1172.05	161629	3487830	1176.00x200	1176.30x100	565.959B	298.7	1.42	3.93
1174.5	12/1/2017 6:31	-2.25	(-0.19%)	1161.27	1172.05	166646	3487830	1176.00x200	1176.30x100	565.959B	298.7	1.42	3.93
1173.96	12/1/2017 6:31	-2.79	(-0.24%)	1161.27	1172.05	171833	3487830	1176.00x200	1176.30x100	565.701B	298.57	1.42	3.93
1173.84	12/1/2017 6:31	-2.91	(-0.25%)	1161.27	1172.05	173006	3487830	1176.00x200	1176.30x100	565.641B	298.54	1.42	3.93
1173.08	12/1/2017 6:31	-3.67	(-0.31%)	1161.27	1172.05	176839	3487830	1176.00x200	1176.30x100	565.274B	298.34	1.42	3.93
1173.07	12/1/2017 6:31	-3.68	(-0.31%)	1161.27	1172.05	178919	3487830	1176.00x200	1176.30x100	565.27B	298.34	1.42	3.93
1173.05	12/1/2017 6:32	-3.7	(-0.31%)	1161.27	1172.05	180542	3487830	1176.00x200	1176.50x300	565.26B	298.33	1.42	3.93
1173.88	12/1/2017 6:32	-2.87	(-0.24%)	1161.27	1172.05	183140	3487830	1176.00x200	1176.50x300	565.66B	298.55	1.42	3.93
1175.53	12/1/2017 6:32	-1.22	(-0.10%)	1161.27	1172.05	189473	3487830	1176.00x200	1176.50x300	566.455B	298.96	1.42	3.93
1175.079	12/1/2017 6:32	-1.6707	(-0.1420%)	1161.27	1172.05	192319	3487830	1176.0000x200	1176.5000x300	566.238B	298.85	1.42	3.932
1175.441	12/1/2017 6:32	-1.3087	(-0.1112%)	1161.27	1172.05	198135	3487830	1176.0000x200	1176.5000x300	566.412B	298.94	1.42	3.932
1176.79	12/1/2017 6:32	0.04	(+0.00%)	1161.27	1172.05	202503	3487830	1176.00x200	1176.50x300	567.062B	299.29	1.42	3.93
1176.635	12/1/2017 6:33	-0.115	(-0.010%)	1161.27	1172.05	208609	3487830	1176.000x200	1176.500x300	566.987B	299.25	1.42	3.932
1176.94	12/1/2017 6:33	0.19	(+0.02%)	1176.75	1172.05	210742	3487830	1176.00x200	1176.50x300	567.134B	299.32	1.42	3.93
1177	12/1/2017 6:33	0.25	(+0.02%)	1176.75	1172.05	215739	3487830	1176.00x200	1176.50x300	567.163B	299.34	1.42	3.93
1177.01	12/1/2017 6:33	0.26	(+0.02%)	1176.75	1172.05	218894	3487830	1176.00x200	1176.50x300	567.168B	299.34	1.42	3.93
1177	12/1/2017 6:33	0.25	(+0.02%)	1176.75	1172.05	224604	3487830	1176.00x200	1176.50x300	567.163B	299.34	1.42	3.93
1177	12/1/2017 6:34	0.2501	(+0.0213%)	1176.75	1172.05	228607	3487830	1176.0000x200	1176.5000x300	567.163B	299.21	1.42	3.932
1177.77	12/1/2017 6:34	1.02	(+0.09%)	1176.75	1172.05	232138	3487830	1176.00x200	1176.50x300	567.534B	299.53	1.42	3.93
1179.18	12/1/2017 6:34	2.43	(+0.21%)	1176.75	1172.05	241035	3487830	1176.00x200	1176.50x300	568.214B	299.8	1.42	3.93

Figure 7: Sample data file for amazon stock.

We have crawled yahoo finance stock data using Jsoup. To follow the number of requests that yahoo allows, we have crawled data alternatively for [8,9,10,11] seconds. Figure7 shows the sample of data crawled for Amazon company. The crawled data from the yahoo finance for the various companies accounts for around 2200 per company in a day. The data which is crawled has many outliers/redundant data due to the delay in updating from the site to yahoo finance. It requires cleaning the data. We have used openRefine to remove some of the redundant data and for some data we have done the cleaning manually. Rows which have more missing attributes are neglected. We found missing values in EPS and PE ratios in fewer places. These values are assumed to be zero. From the crawled data we have considered the following attributes:

- Price: Stock value at that time.
- Time: NYSE website time and pacific time both are taken.
- Change: Change is stock value from the opening value. It shows the amount of increment or decrement in the stock value.
- Percentage Change: Percentage change in stock value from the opening value.
- Close: Previous close value of stock.
- Open: Open price of stock of that day.
- Volume: Number of shares traded on latest trending day.
- Average Volume: Average of volume measured over 30 days.
- Bid: Te highest bid price in the market
- Ask:The ask price is the lowest price a seller of a stock is willing to accept for a share of that given stock.
- Market Capitalization: Total value of a company in the stock market. It is generally calculated by multiplying the shares outstanding by current share price.
- P/E : Price to earning ratio
- EPS- earnings per share- the net income over d last 4 quarters divided by the shares outstanding
- Beta: The measure of a fund 's or a stock's risk in relation to the market or to an alternative benchmark

We have crawled stock data for 12 different stock companies in 3 sectors. 4 companies in Health sector domain, 4 companies in finance domain and 4 in technology sector. Stock companies we have taken are Amazon, Apple, Google, Twitter, Goldman Sachs, Morgan Stanley, Citi Bank, Jp Morgan, Pfizer, Sanofi, Biorad, Novartis

4.2 Extreme Value Theory

Extreme Value Theory(EVT) is a branch of statistics for analyzing the tail behavior of the distribution. Extreme Value Theory helps in finding out the rare events in the distribution (right and left tail), even if we do not have enough samples to analyze them. The main applications of EVT include hydrology, finance and insurance.

4.2.1 ***Generalized Extreme Value Theory***. Definition : Let X_1, X_2, . . ., X_n be a sequence of independent random variables distributed according to a common (usually unknown) function F and define M_n= $max_{i=1}$ ^{n}= (Xi). From the results by Fisher, Tippett and Gnedenko , if there are sequences of normalizing constants a_n > 0, b_n such that (M_n - b_n)/a_n converge in distribution (as n) to a non-degenerate distribution of M_n, namely G(z) = PrM_n ≤ z, G(z) can be only of three types; which can be expressed with parameters being respectively the location, scale, and shape of G, is known as the Generalized Extreme Value (GEV) distribution.

With the GEV distribution, the shape parameter is well within the confidence intervals. The shape parameter decides as to which EV distribution better models the data for the given data sample in the stock.

Algorithm for the Extreme Value Theory Estimation

- Set the initial block size b to 5.
- If the number of blocks [N/b] is less than 30, then stop
- Segment execution times x_1, x_2.., x_N into blocks of b measurements.
- For each of the [N/b] blocks find the maximum values y_1,...$y_{[N/b]}$ where y_i = max ($x_{(i-1)b+1}$, $x_{(i-1)b+2}$, ..., x_{ib}).
- Estimate the best-fit parameters $\sigma, \mu and \xi$ to the block maximum values y_1, ...$y_{[N/b]}$.
- If the Chi Squared fit between the block maximum values y_1, ...$y_{[N/b]}$ and the GEV parameters μ, σ *and* ξ does not exceed a 0.05 confidence value, then double the value of b and go back to Step 2.
- Finally we plot the probability of exceedance vs the Stock price for the testing analysis.

Sampling the maxima and minima

To estimate the model parameters for GEV ξ, μ *and* σ, one must select the sample of maxima and minima from a set of samples for the given stock. The block maxima/minima are the natural approach of performing it, observed data samples are partitioned into m set of disjunct m data blocks of equal lengths and the maximum/ minimum of each block is selected. Furthermore, we use these maximum/minimum samples for fitting the model. Figure 8. gives

Goldman Sachs	Shape parameter (ξ)	Location parameter (μ)	Scale parameter (σ)
Block Maximum	-0.361843500424505	2.382293741091156e+02	0.3292357591369 42
Block Minimum	-0.436503903491632	2.380764724383187e+02	0.3742347255593 29

Amazon	Shape parameter (ξ)	Location parameter (μ)	Scale parameter (σ)
Block Maximum	0.210873951607306	1.162411620632933e+03	3.165474114781903
Block Minimum	-0.008470880745046	1.161424701452333e+03	3.933939338246518

Figure 8: Parameters for the GEV method for AMAZON and GS stocks .

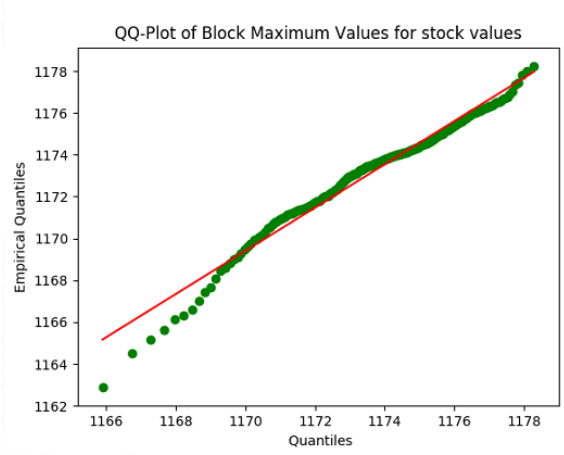

Figure 9: QQ plot of Block Maximum values for stock values .

the parameters for the maximum and minimum of the blocks and their corresponding parameters for Amazon and GS stocks.

Inference for the GEV distribution

Since we do not have any restrictions on the shape parameter ξ, we apply Maximum likelihood estimation for the calculation of all the three parameters of the GEV distribution. We have used Matlab to find out the parameters for the given stock distributions.

Goodness-of-fit and extreme value conditions

Poor fit of the GEV models can be recognized as the violation of iid assumption, if difficulties in obtaining asymptotic convergence properties take place; or when the data simply cannot be analyzed via EVT.

4.3 Evaluation of EVT

For the AMAZON stocks, we have used the 28th November as the training data(finding out the three parameters for the GEV estimation), which helps in obtaining the upper and lower bound.

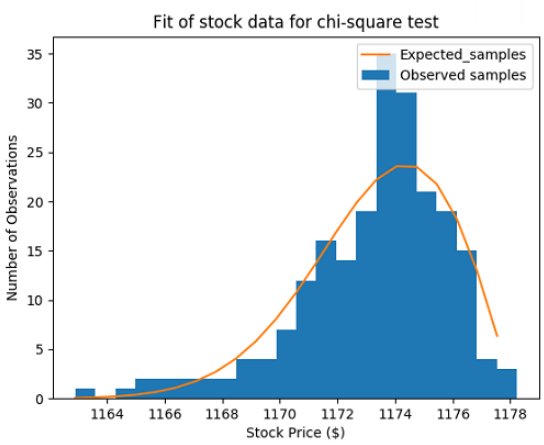

Figure 10: Fit of stock data for chi-square test .

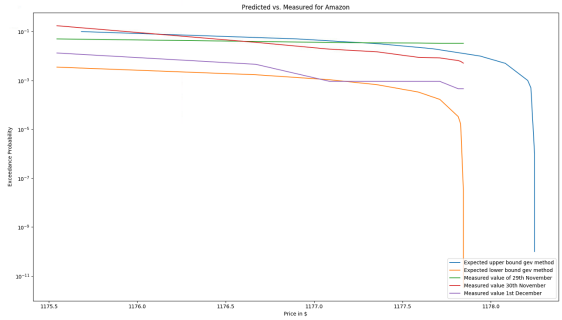

Figure 11: Predicted vs Measured stock data for Amazon.

Furthermore, prediction part has been performed for the near future values of the stock. Here we check what is the probability of exceedance to our prediction.

From the Figure 11. we see that the stock data for Amazon (29th November, 30th November and 1st December) almost lie inside our prediction. But few outliers are present which is seen in the graph, where the stock data is exceeding the upper and lower bound. Furthermore, we see that the probability of exceedance is in the range less than 0.08, which shows that it is very rare and probability of exceeding our prediction is very low.

Similarly for the GOLDMAN SACHS stocks, we have used the 16th November as the training data(finding out the three parameters for the GEV estimation), which helps in obtaining the upper and lower bound. Furthermore, prediction part has been performed for the near future values of the stock. Here we check what is the probability of exceedance to our prediction as per the Figure 12.

Our prediction for the near future values for the Goldman Sachs, rest well inside our prediction which shows that there are very less chance for the data to be exceeding out prediction.

4.4 Clustering

To study how the stocks are related or to know which stocks behave similarly, we can use clustering. Similarity can be in many ways,

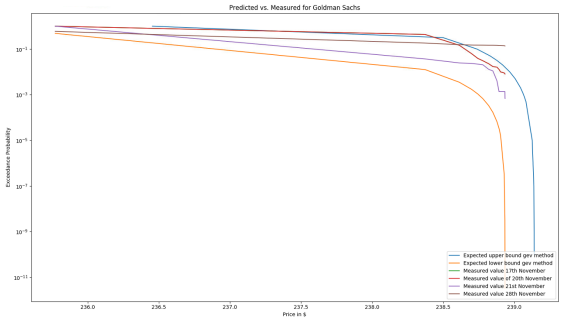

Figure 12: Predicted vs Measured stock data for Goldman Sachs.

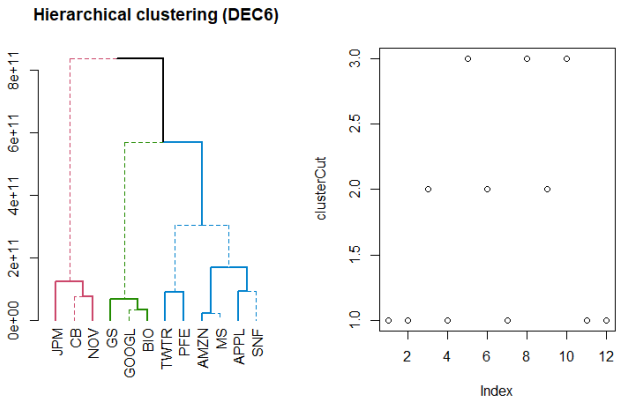

Figure 13: Hierarchical Clustering on December 6th data

grouping can be done based on the price, size of the company, number of employees, products of the company, location of the company or by sector. We can use different clustering algorithms on stock data. Choosing a clustering algorithm mainly depends on the type of data which needs to be clustered. Analysis of stock market, demands many attributes to be considered while acknowledging the similarities between the companies. We have done the clustering using Hierarchical clustering method and NMF method.

4.4.1 ***H****ierarchical clustering.* In hierarchical clustering, we cluster data in hierarchical way where in first step we have N objects in a single cluster, but as the process proceeds we get N clusters each with one object. Agglomerative hierarchical cluster uses the combination of N objects into groups and divisive methods which separate N objects into finer groupings. In this project we have considered agglomerative hierarchical approach with Euclidean distance. Agglomerative clustering is usually displayed in a tree like structure called dendrogram. These begin with each object in a different cluster. At every step, the two clusters that are most similar are joined into a single new cluster. The first task is to form the distances between each objects.

Algorithm : Let the distance between clusters i and j be represented as dij and let cluster i contain ni objects. Let D represent the set of all remaining dij . Suppose there are N objects to cluster.

- Find the smallest element d_{ij} remaining in D.
- Merge clusters i and j into a single new cluster, k.
- Calculate a new set of distances d_{km} using the following distance formula.
  $d_{km} = \alpha_i d_{im} + \alpha_j d_{jm} + \beta d_{ij} + \gamma |d_{im} - d_{jm}|$ Here m represents any cluster other than k. These new distances replace dim and d jm in D. Also let nk = ni + n j . (Eight algorithms available represent eight choices for $\alpha i, j, \beta, and \gamma$).
- Repeat steps 1 - 3 until D contains a single group made up off all objects. This will require N-1 iterations. We will now give brief comments about each of the eight techniques

There are 8 different techniques namely, Single linkage, Complete linkage, Simple average, Centroid, Median, Group average, Ward's minimum variance and Flexible strategy. To know which technique is a good fit, there are measures like cophenetic correlation coefficient or delta. For the sake of simplicity, we have considered complete linkage and average linkage. In complete linkage a cluster is formed when all the dissimilarities between pairs of objects in the cluster are less than a particular level. It defines the distance between two groups as the distance between their two farthest-apart members. It results in well separated and compact clusters. Average linkage method defines the distance between groups as the average distance between each of the members.

To form the clusters, we have selected few attributes from the crawled data. For the hierarchical clustering, we have considered price of stock, volume, average volume, market capital, PE ratio, beta, eps, change in price, bid and ask. First, the clustering is done on daily basis and then on a week full data. For a particular day's data, closing price of stock for that day, number of shares traded that day(volume), average volume, difference in the market capital, change in stock price compared to the opening value of that day along with above mentioned attributes are considered. Figure 13. represents hierarchical clustering with a dendrogram plotted on the data of Dec 6th. This diagram explains which clusters have been joined at each stage of the analysis and the distance between clusters at the time of joining. The large jump in the distance between clusters shows at one point clusters which were similar were joined and at another stage clusters joined were apart. So, we can find the optimum number of clusters by looking at the jumps in distance. From the left side of the Figure 13. it is clear that number of clusters should be either 2 or 3. We have considered 3 clusters.

For clustering on a week data, total volume for that week, total change in value of the stock, average bid and ask for that week, difference in the market capital for that week and other attributes like eps, PE ratio etc are considered. Figure 14. shows the dendrogram for data considered between dec 1st to dec 8th.

Figure 14. shows hierarchical clustering with complete linkage and Figure 15. represents clustering with average linkage. We can see in both the cases, the clustering is same. Usually depending on the distance matrix, there will be a change in both the methods. For the complete linkage, stocks grouped are as in Figure 16. and for average linkage, stocks grouped are as in Figure 17.

Further explained clustering are with respect average linkage. Another way to represent the clustering is through circular dendrograms as seen in Figure 18. By the dendrograms, we can see the

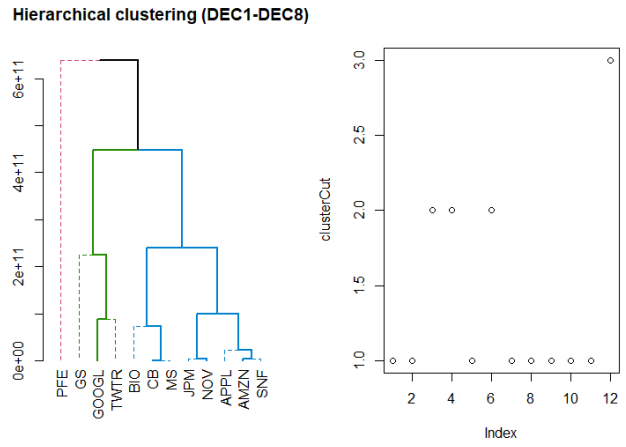

Figure 14: Hierarchical Clustering on Dec1st to 8th data with complete linkage

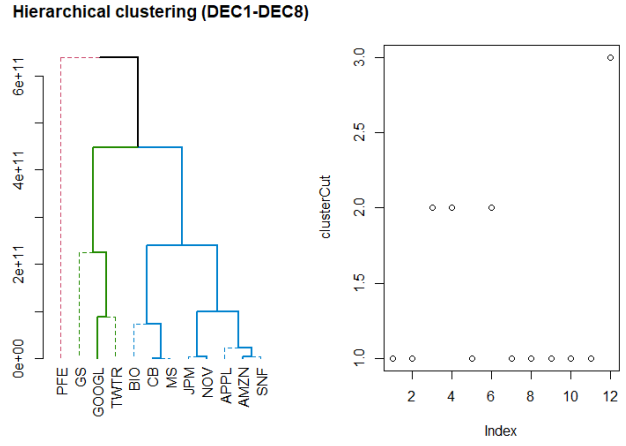

Figure 15: Hierarchical Clustering on Dec1st to 8th data with average linkage

```
> plot(clusterCut)
> print(clusterCut)
 AMZN  APPL GOOGL  TWTR    CB    GS    MS   JPM   BIO   NOV   SNF   PFE
    1     1     2     2     1     2     1     1     1     1     1     3
>
> #-----------------------------------------------------------------
>
```

Figure 16: Hierarchical Clustering with complete linkage

```
> clusterCut <- cutree(hc, 3)
> plot(clusterCut)
> print(clusterCut)
 AMZN  APPL GOOGL  TWTR    CB    GS    MS   JPM   BIO   NOV   SNF   PFE
    1     1     2     2     1     2     1     1     1     1     1     3
>
> #-----------------------------------------------------------------
```

Figure 17: Hierarchical Clustering with average linkage

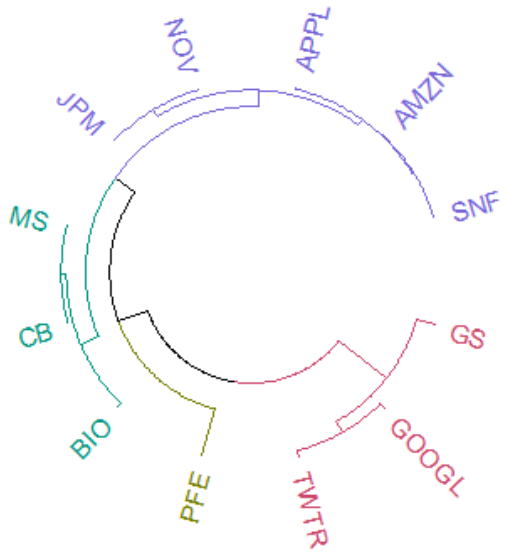

Figure 18: Hierarchical Clustering with average linkage

distance between the stock PFE and other stocks are more. Even though stocks selected are from 3 different sectors, stocks behaving similar within the sector are less. Stocks are grouped differently based on different attributes. By the Figure 13. and Figure 14 we can see that a single day clustering differs from overall clustering done by taking a week data. In Figure 19. represented as a table, rows represents the clustering done either on one week data or on the mentioned day with given attributes and columns show the stocks which we have considered. Stocks represented by 1 are grouped together in the bottom level, representing the distance between them is less, stocks representing 2 and 3 are of higher levels. We can see from the table how each day data results in different clusters. Clustering by the change in value of stock results in similar clusters for Dec4, Dec5, Dec6 and Dec1st to 8th data. But as there are some bigger companies and smaller ones, the volume price and the market capitalization play an important role. By considering those attributes we can see in most of the cases Apple, Amazon, Morgan Stanley, Sanofi, and Jp Morgan stocks behave some what similarly. Similarly Google and Goldman Sachs are clustered.

*4.4.2 **NMF.*** Finding an algorithm which supports the variability in the parameters is the one which is preferred the most for clustering the stock data. Many advanced methods have now been developed for the clustering datasets, which use these varied mathematical parameters have been deduced. One such algorithm is Non-Negative Matrix Factorization. NMF has an inherent clustering property. It uses the basic concept of matrix factorization, wherein the original matrix is split into two different matrices by specifying a rank for the matrix. Rank of a matrix is calculated by checking the dependency amongst the columns of the matrix. The rank of any matrix itself shows the dependency between the data of the

	AMZN	APPL	GOOGL	TWTR	CB	GS	MS	JPM	BIO	NOV	SNF	PFE
Clustered based on all atributes on dec 1st to 8th	1	2	3	3	3	1	1	1	1	1	1	1
Clustered based on all the sttributes on Dec 4th	1	2	3	3	3	1	1	1	1	1	1	1
Clustered based on all the sttributes on Dec 5th	1	1	2	2	3	2	2	1	2	3	3	3
Clustered based on all the sttributes on Dec 6th	1	1	2	1	3	2	1	3	2	3	1	1
Clustered based on total change in the price during 1st to 8th dec	1	2	3	2	2	2	2	2	1	2	2	2
Clustering based on total change in price on Dec 4th	1	2	3	2	2	2	2	2	3	2	2	2
based on total change in price on Dec 5th	1	2	1	2	2	3	2	2	3	2	2	2
Clustering based on total change in price on Dec 6th	1	2	3	2	2	2	2	2	2	2	2	2
Clustering based on Price, Volume and MarketCapital during dec 1st to 8th	1	1	2	2	1	2	1	1	1	1	1	3
Clustering based on Price, Volume and MarketCapital on Dec4th	1	2	2	3	3	3	1	1	1	1	1	1
Clustering based on Price, Volume and MarketCapital on Dec5th	1	1	2	2	3	2	2	1	2	3	3	3
Clustering based on Price, Volume and MarketCapital on Dec6th	1	1	2	1	3	2	1	3	2	3	1	1

Figure 19: Comparison of Hierarchical Clustering with different attributes on different dataset

matrix. Non-negative Matrix factorization is based on the idea of using the matrix factorization with no negative values included in all the three matrices. This methodology was first proposed by Lee and Seung in 1999[[3]]. Since then NMF has been used in myriad applications like text mining and Image classification. The main idea in NMF is that a matrix V of nxm dimensions, is split into W, which is of the dimension nxr and into H, which is rxm, where n is the number of rows, m is the number of columns, and r is the rank of the matrix, which is usually chosen to be less than, n or m. These matrices are then updated by multiplicative update rules in the algorithm. The update is done by using Euclidean distance between the two matrices or the divergence in them. We have used

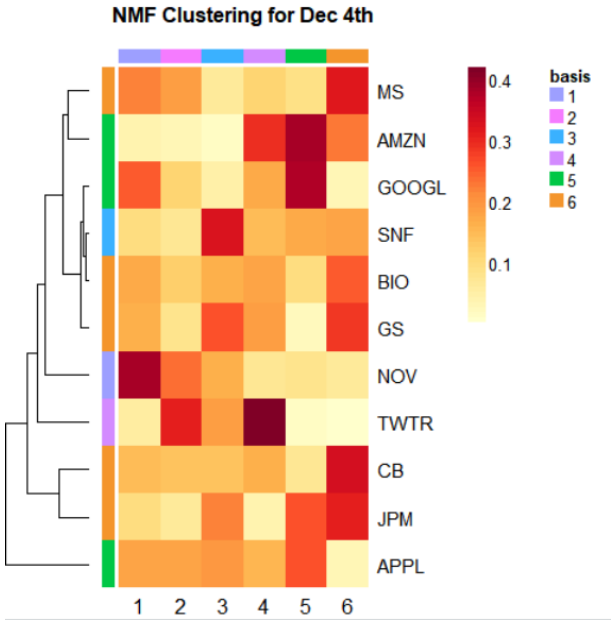

Figure 20: Clustering heat map obtained for 4th December

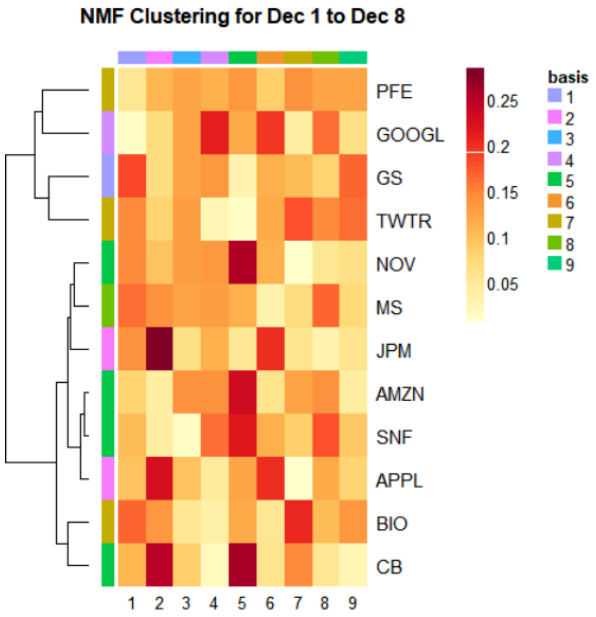

Figure 21: Clustering heat map obtained from December 1st to 8th

the Euclidean distance, to measure the difference between the matrices. The dataset consists of 12 rows and 12 columns wherein, the columns are the features of the data and the rows are the names of the companies.The rank of the matrix is given as six. Deciding the rank is one of the crucial factors. This has been reached upon after trying a set of different numbers for the rank and finalized when appropriate results were obtained. From each sector, four companies are considered.The clustering is done for the data on a daily and weekly basis. The data for December 4th and December 5th is clustered individually. Also, the data from December 1st to December 8th is considered as whole to show the clustering for an entire week. Many of the attributes have negative values, which the algorithm doesn't consider. To overcome this, a small change is made in the dataset. Boolean columns are added which depict 1 for negative and 0 for positive are considered. This is done for attributes consisting of negative values.The clustering obtained is visualized as a heat map.

Figure 20. shows the generated heatmap which, clearly depicts that the companies clustered together mostly belong to the same sector. Like JPMorgan(JPM) and Citibank(CB) belong to the finance sector, Apple(APPL) and Twitter(TWTR) are from technology sector and BioRad laboratories(BIO) and Sanofi(SNF) belong to the health sector. Also, many companies from different sectors are combined either at earlier stage or at a later stage. It is visible that since the rank of the matrix is six, there are six different columns formed in the heatmap. The basis in the heat map is obtained from the rank of the matrix.

Similar attributes are considered for the data for the entire week of December 1st to 8th. Figure 21 shows the heatmap for the same. Here the matrix has been assigned rank nine, for getting a proper clustering of the data.

Many companies are clustered from different sectors here, which is quite different from what we where getting for a single day's data. This shows us the difference of how the same attributes can impact the data, only by changing the time period for which they are considered. To obtain better clustering results, a few attributes where removed and the clustering was performed. The data of December 5 was initially clustered, with all the attributes and later a few changes were made in the selection of these attributes. The features close, open, average volume and change where skipped in the dataset to discover a new set of clusters formed. The results of clustering for both the cases was achieved. The rank of the matrix considered here is 6.

Comparing Figure 22. and Figure 23. we can easily say that the clustering results which we obtain are quite different from one another. This is when a few attributes are skipped and then the analysis is performed.

5 CONCLUSIONS

We are able to predict the range for stock prices using EVT which obeys the iid conditions. Measured values of the stocks are lying in th expected range most of the times. Similarity and dissimilarity between the stocks are visualized by clustering them.

REFERENCES

[1] Jeffery P. Hansen, Scott A. Hissam, and Gabriel A. Moreno. 2009. Statistical-Based WCET Estimation and Validation. In *9th Intl. Workshop on Worst-Case Execution Time Analysis, WCET 2009, Dublin, Ireland, July 1-3, 2009.* http://drops.dagstuhl.de/opus/volltexte/2009/2291

[2] Madhusudan Karmakar and Samit Paul. 2016. Intraday risk management in International stock markets: A conditional EVT approach. *International Review of Financial Analysis* 44, Supplement C (2016), 34 – 55. https://doi.org/10.1016/j.irfa.2015.11.008

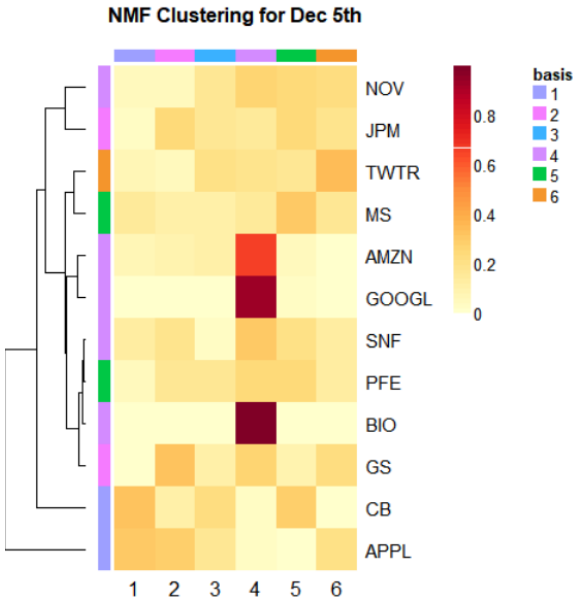

Figure 22: Clustering heat map obtained for the changed 5th December

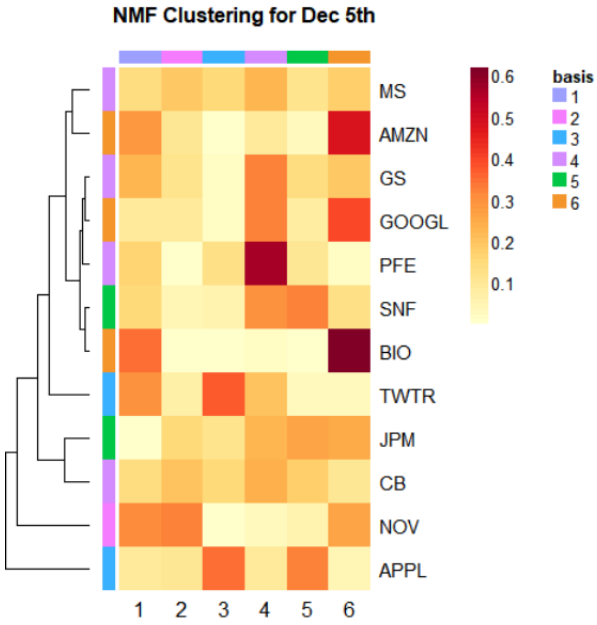

Figure 23: Clustering heat map obtained for 5th December

[3] Daniel D. Lee and H. Sebastian Seung. 1999/10/21/online. Learning the parts of objects by non-negative matrix factorization. *Nature* 401 (1999/10/21/online). http://dx.doi.org/10.1038/44565

[4] G. Lima, D. Dias, and E. Barros. 2016. Extreme Value Theory for Estimating Task Execution Time Bounds: A Careful Look. In *2016 28th Euromicro Conference on Real-Time Systems (ECRTS)*. 200–211. https://doi.org/10.1109/ECRTS.2016.20

[5] Luca Santinelli, Jérôme Morio, Guillaume Dufour, and Damien Jacquemart. 2014. On the Sustainability of the Extreme Value Theory for WCET Estimation. 39 (2014), 21–30. https://doi.org/10.4230/OASIcs.WCET.2014.21

[6] Elena Claudia Serban, Alexandru Bogeanu, and Eugeniu Tudor. 2013. Clustering Techniques In Financial Data Analysis Applications On The U.S. Financial Market. *Annals - Economy Series* 4 (August 2013), 176–194. https://ideas.repec.org/a/cbu/jrnlec/y2013v4p176-194.html

[7] Abhay K. Singh, David E. Allen, and Robert J. Powell. 2017. Tail dependence analysis of stock markets using extreme value theory. *Applied Economics* 49, 45 (2017), 4588–4599. https://doi.org/10.1080/00036846.2017.1287858 arXiv:https://doi.org/10.1080/00036846.2017.1287858