An Analysis of the impact of COVID-19 on the US Equity market

Luke Allen [1], Jaxom Dunn[2], and Frazer Birch[3],
School of Computing, Electronics and Mathematics,
University of Plymouth, PL4 8AA
(Dated: 17/05/2021)

Abstract

The COVID-19 pandemic has demonstrated a severe impact on the US equity market and it is easily observable. The impact on the stock market has been due to most industries relying on human interaction, and in a pandemic, human interaction is limited to reduce fatalities and the strain on healthcare services. We wanted to demonstrate that the impact of the pandemic could be negated, or even profited from by utilising the performance similarities (correlations) that the chosen equities had. We used statistical analysis to calculate these correlations throughout the chosen time period and subsequently used them to develop a portfolio which had a positive performance during the pandemic. After retrospectively applying the portfolio to the data set, we discovered that we obtained a significantly positive return, even with minimal risk and higher risk resulted in a linear increase in gain. We wanted to analyse the realistic applications of our testing, and optimise our results to obtain the most realistic results so that our portfolio could be applied now and help to predict movements in future.

[1]Email: luke.allen-7@students.plymouth.ac.uk
[2]Email: jaxom.dunn@students.plymouth.ac.uk
[3]Email: frazer.birch@students.plymouth.ac.uk

Contents

1 Introduction

1.1 Background Information

The Coronavirus (COVID-19) pandemic was first declared a pandemic by the World Health Organisation on March 11th 2020 [1] and is still an ongoing issue throughout the world. In order to reduce the spread of COVID-19 the US government limited human interaction by closing schools and issuing a lockdown; as a result of this a majority of the population had to work from home. Many businesses that couldn't operate from people's homes ended up closing down or downsizing thus leading to mass unemployment; by April 2nd 2020 10 million Americans were out of work [2].

Using the daily quotes of four United States (US) equity market sectors we analysed the portfolio returns during the Coronavirus (COVID-19) pandemic. The four US equity market sectors chosen were Healthcare [3], Consumer Services [4], Financial Services [5] and Real Estate [6]; the data we analysed was obtained using Yahoo finance and was provided by our project supervisor Dr Alessandro Cardinali. The four Exchange Traded Funds (ETF)s were chosen as their wide portfolios allows for a greater representation of the market as a whole rather than selecting individuals assets. It should be noted that these ETFs will never be completely representative of a sector as the shares selected for the index are non-random in nature since they have been selected based on their performance [7]. The portfolios of these ETFs can be found at the iShares website [8].

As of March 31st 2021 the US equity market is worth over $49 Trillion dollars [9] and in 2018 it was the biggest stock market in the world accounting for 43% of the world market value [10] because of this COVID-19 had a huge impact on the value of stocks which made the effects of it easily observed. The four sectors given were diverse which allowed us to show that the pandemic affected the market and not just one sector. Figure (1) shows that following the days after COVID-19 was declared a pandemic the closing price for all four of our sectors fell and all but Healthcare failed to recover to their pre-pandemic peak; after the initial drop Healthcare continued to rise well past it's pre-pandemic peak however this is expected as the demand for Healthcare increases during a pandemic.

The aim of this investigation was to demonstrate the impact of COVID-19 on the US stock market and how it could be negated or even profited from. We did this by utilising the correlation that the four equities had. To calculate the correlation we performed statistical analysis of our data between January 3rd 2017 to February 9th 2021. Using these correlations we developed a portfolio; which had a positive performance during the COVID-19 pandemic. The portfolio that we developed was then retrospectively applied to the dataset; using this we found that we obtained a significantly positive return that showed a linear increase in gain with both minimal and higher risks.

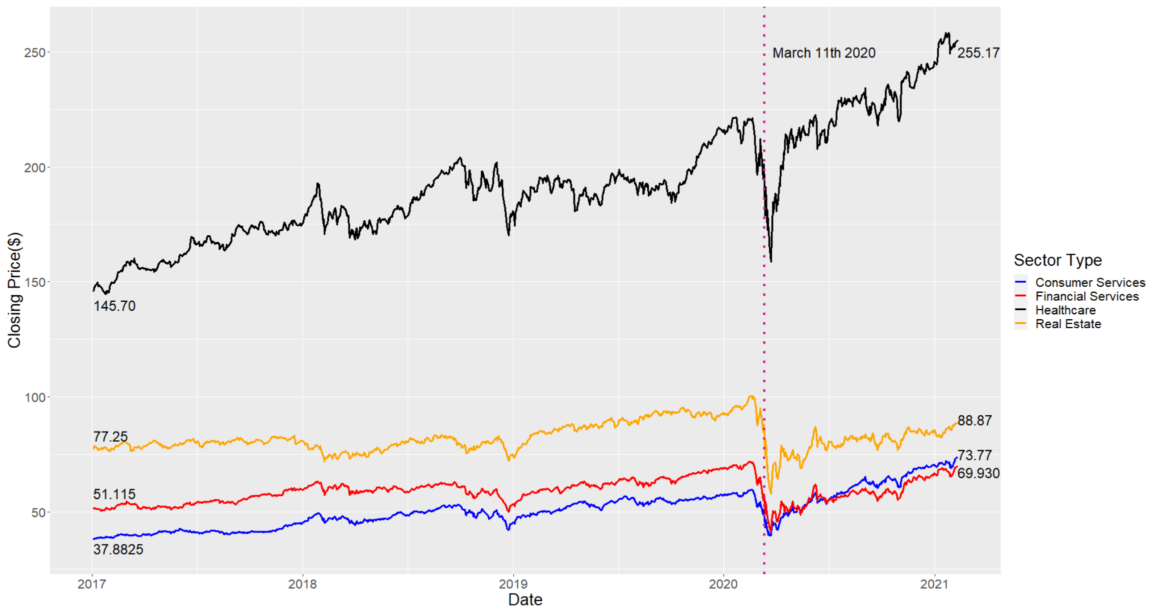

Figure 1: Closing price of our four different US market indices

1.2 Data Set Analysis

Figure (1) displays a plot of the entire data set in which the effect of COVID-19 can clearly be seen during 2020. As previously mentioned, March 11th being the definition of when the pandemic was declared highlights the performance of the four sectors at this time, with all four having great losses for the majority of 2020.

In order to understand how the pandemic as a whole impacted the four equities overall, Figure (2) shows the correlation of the stocks. The top right-hand corner displays the correlation coefficients between pairs of sectors, the diagonal displays the density plots of each sector and the bottom left-hand corner displays a scatter graph with the line of best fit for the pairs of sectors. All of the correlation coefficients and the line plots were positive which tells us that there is positive correlation between all pairs of the sectors, as expected across the equity market, but with different degrees of correlation which is indicative of the different unsystematic risks in each of the four sectors. The highest correlating sectors were that of Consumer Services and Healthcare with a correlation of 95% which indicates that they performed extremely similar over the time period. Conversely, Real Estate and Healthcare were the least correlated with a correlation of 35% which suggests that these two sectors had different responses to events overall. The Figure (3) is the same data though shortened to only include those dates where COVID-19 would have effected the performance of the equities which we have defined as starting January 31st 2020. This is because despite the US being widely unaffected by COVID-19 at this time, the market would reflect growing concerns surrounding the developments across the world especially in China. Shortening the data set allows a better analysis of how these stocks performed as a result of the pandemic. The correlations are shown to have all increased between the pairs which suggests the significance of the pandemic as a systematic risk affecting all four sectors in very similar ways. The density plots in Figure (3) for Healthcare and Consumer Services are skewed to the right which shows they returned to pre-pandemic levels of growth and above. Whilst Real Estate and Financial Services remain similar to their densities in the Figure (2) which would suggest less growth towards recovering.

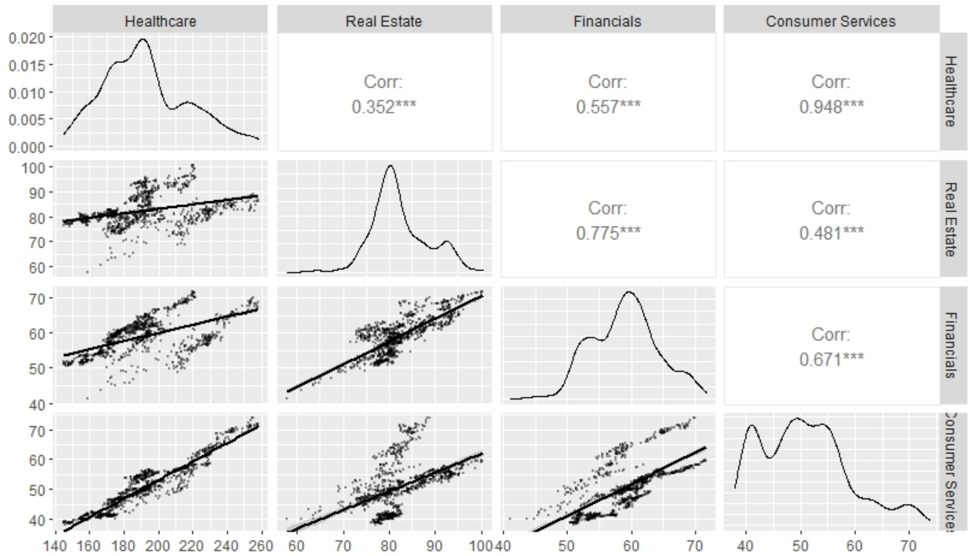

Figure 2: Correlation and density plot between sectors across entire time series

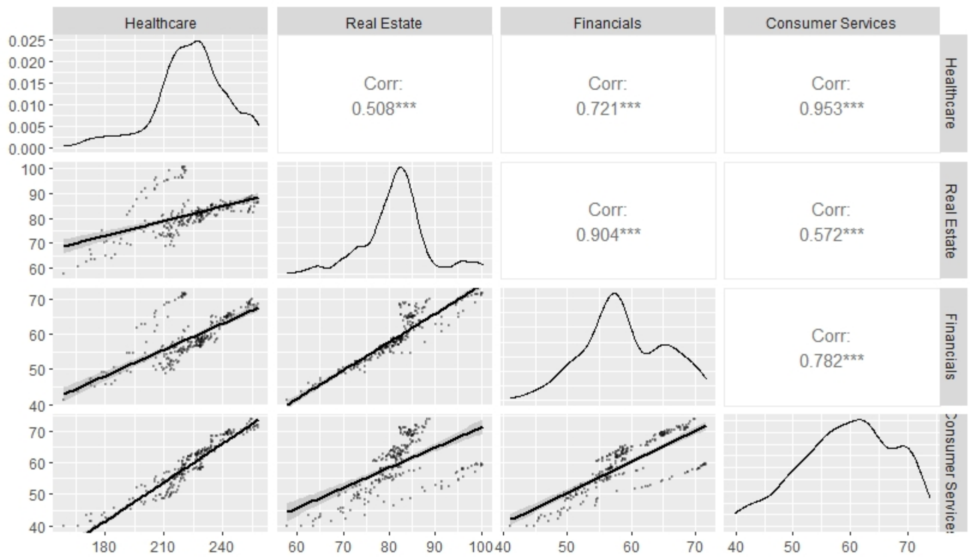

Figure 3: Correlation and density plot between sectors during COVID-19

2 Methods

The data set used was formatted as a time series data set that allowed manipulation using RStudio. We wanted to use conditional Mean-Variance portfolios to determine if we could generate a portfolio to negate the impacts of COVID-19, a method learned in our previous Financial Statistics module [11]. The weights of the portfolio were generated using the following formula:

$$\hat{\mathbf{w}}_{t+1|t}(q) = \frac{\widehat{H}_{t+1}^{-1}\mathbf{1}}{a} + q[\widehat{H}_{t+1}^{-1}(\hat{\boldsymbol{\mu}} - \frac{b}{a})] \tag{1}$$

where

$$a = \mathbf{1}'\widehat{H}_{t+1}^{-1}\mathbf{1} \ , \tag{2}$$

$$b = \mathbf{1}'\widehat{H}_{t+1}^{-1}\hat{\boldsymbol{\mu}} \ . \tag{3}$$

where $\hat{\boldsymbol{\mu}}$ is the one-step-ahead conditional mean, q is a positive number that represents the risk appetite of investors and $\widehat{H}_{t+1}^{-1}$ is the inverse of the one-step-ahead conditional covariance matrix (CCM) for the vector of returns $\boldsymbol{\epsilon_t}$. To obtain the CCM, we deduced that multivariate (MV) GARCH modelling would be the most effective way of generating it, due to the multiple variables present in the dataset. We used RStudio to perform Dynamic Conditional Correlation (DCC) estimation, which estimates an MV GARCH model. We used DCC as it is a far more efficient way of generating an MV GARCH model.

2.1 Dynamic Conditional Correlation

In terms of DCC estimation, the CCM is represented by:

$$H_t = D_t^{1/2} R_t D_t^{1/2} \tag{4}$$

where $D_t = \text{diag}\ (h_{ii,t})$. D_t contains the volatilities and R_t is a matrix which contains the conditional correlations. The $h_{ii,t}$ are treated as univariate GARCH(1,1) models. $\hat{\boldsymbol{\omega}}_{\boldsymbol{t}}$ are the residuals of the GARCH(1,1) model:

$$\hat{\boldsymbol{\omega}}_t = \hat{H}_t^{-1/2}\hat{\boldsymbol{\epsilon}}_{\boldsymbol{t}} \tag{5}$$

$\hat{\rho}$ is an exponential smoothing estimator:

$$\hat{\rho}_{ij,t} = \frac{\sum_{s=t-n-1}^{t-1}(\lambda^s \hat{\omega}_{i,t-s})}{\sqrt{(\sum_{s=t-n-1}^{t-1}\lambda^s \hat{\omega}_{i,t-s})(\sum_{s=t-n-1}^{t-1}\lambda^s \hat{\omega}_{j,t-s})}} \tag{6}$$

where $\hat{\omega}_{ij,t} = h_{ii,t}^{-1/2}\hat{\epsilon}_{i,t}$ for $i = 1, 2, ..., m.$, $\hat{\omega}$ are the individual values in the matrix omega and λ is the smoothing co-efficient. However, if λ is constant, it is too restrictive. Therefore, we introduce $q_{ij,t}$ as a recursive representation of exponential smoothing, releasing that restriction:

$$\rho_{ij,t} = \frac{q_{ij,t}}{\sqrt{q_{jj,t} \cdot q_{ii,t}}} \tag{7}$$

where

$$q_{ij,t} = (1-\lambda)(\hat{\omega}_{i,t-1}\hat{\omega}_{j,t-1}) + \lambda(q_{ij,t-1}) \tag{8}$$

We define $q_{ij,t}$ as the conditional covariance between the sequences $\hat{\omega}_{i,t-1}$ and $\hat{\omega}_{j,t-1}$. However, this is a non-stable univariate GARCH model. Instead we define it such that we have a stable univariate GARCH:

$$q_{ij,t} = \rho_{ij} + \alpha(\hat{\omega}_{i,t-1}\hat{\omega}_{j,t-1} - \rho_{ij}) + \beta(q_{ij,t-1} - \rho_{ij}) \tag{9}$$

where ρ_{ij} are the unconditional correlations between the residuals sequence ω_{it}, and ω_{jt}. The matrix form is given by Q_t:

$$Q_t = S(1-\alpha-\beta) + \alpha(\boldsymbol{\hat{\omega}_{t-1}\hat{\omega}'_{t-1}}) + \beta Q_{t-1} \tag{10}$$

The matrix form is given by Q_t, where S is the unconditional correlation matrix of the residuals ω_t, equivalent to R_t from our starting equation. Now that we have our H_t we can generate our required portfolio weights, and analyse the effectiveness of our model.

Once we obtained our values for our weights, we needed to retroactively apply them to the time scale to determine if our predictions would have been valid. We applied our formulae to the previous 300 days worth of prices, starting from the 300th day, and acted as we would if we applied this in the past. Our portfolios were re-calculated daily, following the new window of data, and estimated returns were generated. The returns were generated from the average log returns on the window times for the weights of each equity within the portfolio at each step. With our backtested data, it should be noted the first 300 values do not exist, because the window size is 300.

2.2 Dynamic Conditional Correlation with EWMA estimation

As we defined in our DCC section, the forecasted returns used for the weights equations was calculated using the one step ahead conditional mean. This is an overly simplistic method of estimating returns, so our backtested returns may not accurately reflect the behaviour and actual return we could have expected. Exponential Weighted Moving Averages (EWMA) are useful for the forecasting of markets because they use the simpler moving average baseline, and expand upon it to give strong accurate estimations. In this case, we selected the Hull Moving Average (HMA) [12] as a method for forecasting expected prices, and taking our weights from that method. The HMA is a useful MA to use because of how it is derived.

$$\mathrm{WMA}_1 = \mathrm{WMA}(\mathrm{prices}, n) \tag{11}$$

$$\mathrm{WMA}_2 = \mathrm{WMA}(\mathrm{prices}, \frac{n}{2}) \tag{12}$$

where prices is the equity prices and n is the number of specified observations

$$\mathrm{HMA}_{raw} = (2 * \mathrm{WMA}_1) - \mathrm{WMA}_2 \tag{13}$$

$$\text{HMA} = \text{WMA}(\text{HMA}_{raw}, \sqrt{n}) \tag{14}$$

This process reduces the lag you would usually get from a regular moving average because the reduced n values in the WMAs give more weight to recent values, while still taking into account the whole window of observations, maintaining accuracy. After generating a forecast based upon the HMA, we implemented this into our weights equation (1) as the new $\hat{\boldsymbol{\mu}}$ from which to derive our risk value. This will give more accurate representations of the implications of the risk value q which is down to investor preference. After using this method, we followed the process as before, where we multiply the weights by the average log returns on the window, to generate estimated returns.

2.3 Cointegration Relationships

For our second method, we went for a simpler method. We studied the cointegration of the data set to determine if there were any relationships we could use in order to generate some forecasts and predict our expected returns that way. Specifically, we used Johansen's approach [11] to model cointegrated VARs of which we have a model with a restricted trend. The restricted vector error correction model (VECM) representation of this model is:

$$\Delta X_t = \mu_0 + \alpha(\beta' X_{t-1} + \rho_1 t) + \Gamma_1 \Delta X_{t-1} + \cdots + \Gamma_{p-1} \Delta X_{t-p+1} + \epsilon_t \tag{15}$$

where X_t is the series which is not cointegrated with a drift vector μ_0 and the cointegrating relationships $\beta' X_t$ may have non-zero mean.
Johansen's approach involves a series of steps:

- Specify a $VAR(p)$ model for X_t, where X_t is the cointegrated vector
- Perform likelihood tests for the rank of the long-run impact matrix in order to determine the number of cointegrating vectors needed
- Perform normalisation if needed
- Estimate the cointegrated VECM via maximum likelihood

We determined our model has a trend, as we consider the COVID-19 pandemic and its impacts on the market as a whole to be a trend. We used hypothesis testing to determine how many cointegrating vectors exist, if any. The drawbacks of using cointegrated models is that if there are no cointegrating relationships, the forecasts will not be effective. Using the R package 'vars', we can perform Johansen's approach to determine which cointegration relationships exist, if any. After performing the subsequent hypothesis testing to discern if the cointegrating relationships exist, we found that there were none. The packaged function for discerning the VAR model does not compute with a data set with 0 co-integrating relationships,so to complete the process, just in case some degree of modelling could still provide a buffer against the impacts of COVID-19, we ran the modelling as if there was 1 co-integrating relationship. We chose 1 co-integrating relationship as this is the minimum required to complete the test. Running the forecast in this way, and using our backtesting process we developed to reflect our forecasts as if they were run in real time, we found that it was not an efficient method of forecasting returns, and it gave poor results.

3 Results

3.1 Mean-Variance Portfolio

Drawing attention to the weights in our portfolio, with the two risk parameters Q=0 and Q=1, as seen in Figure (4) and (5) respectively, we can see that the patterns are close to being identical, with Financial Services and Consumer Services being oppositely correlated and a crossover point around 0.7. At around the autumn 2020 mark, there is less emphasis in Financial Services with the higher risk value, which leads to a recovery of the previous losses. As expected based on how consumer services performed, it was the highest weighted asset in the portfolio for the majority of the pandemic.

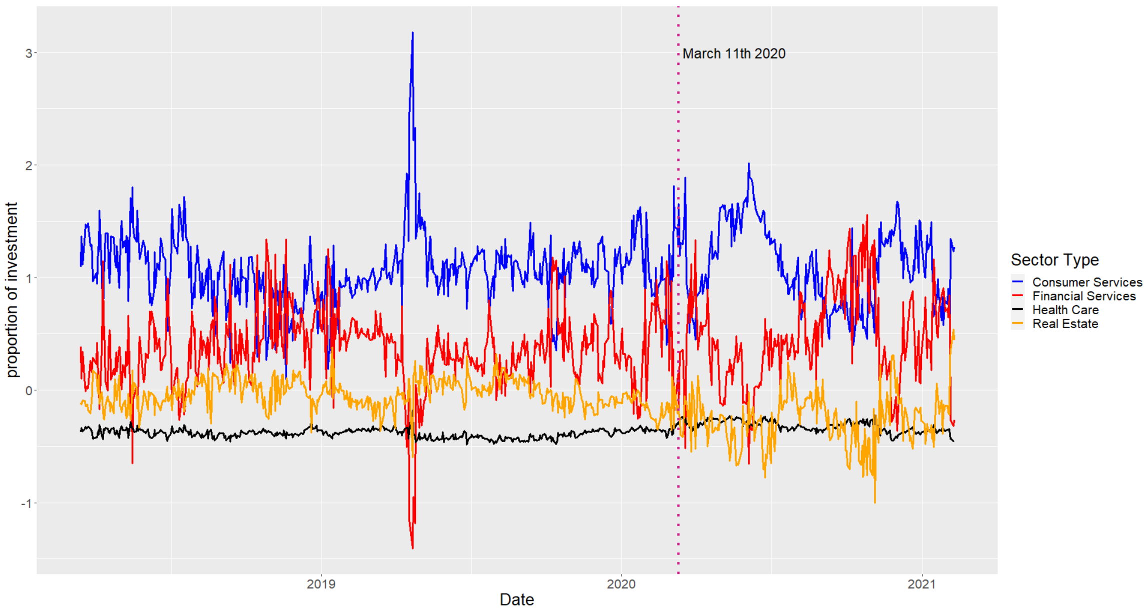

Figure 4: Portfolio weights for Q=0

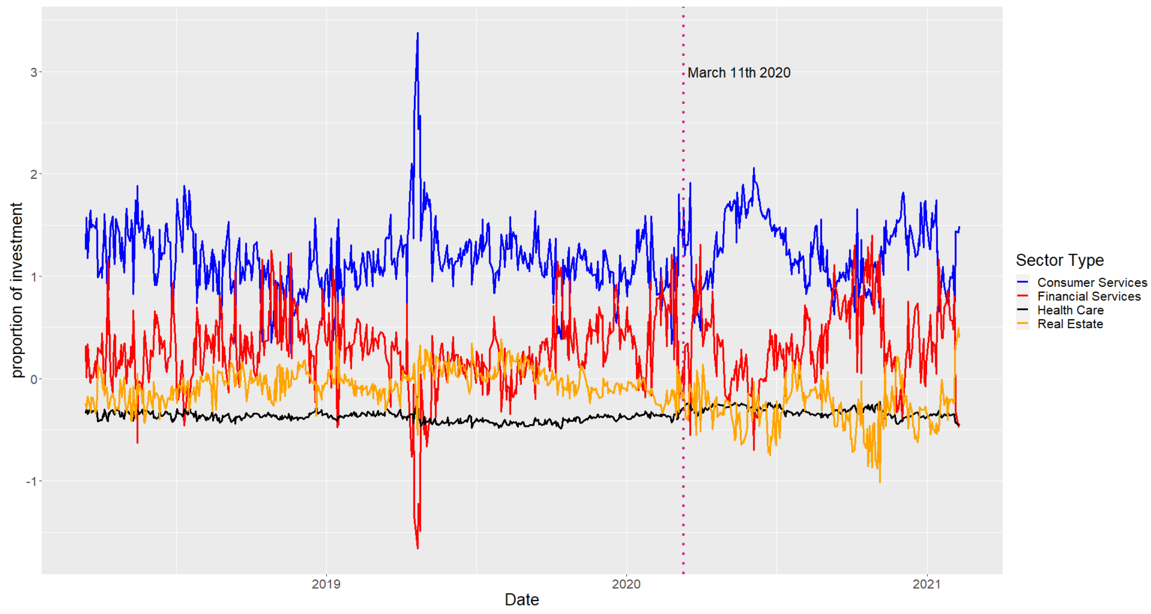

Figure 5: Portfolio weights for Q=1

The portfolio returns for four different risk appetites are displayed in Figure (6) ; where q = 0 is the lowest risk and q = 10 is a higher risk.

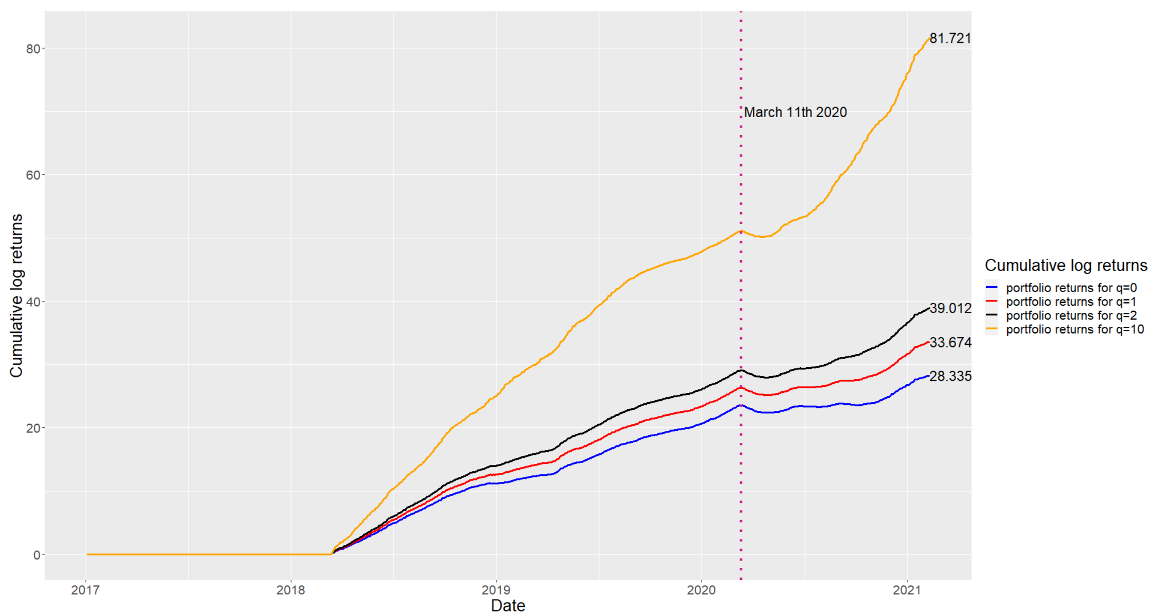

Figure 6: Mean-variance returns for various risk appetites

A lower risk appetite results in lower cumulative log returns and it can be seen that as the risk increases so does the cumulative log returns. At the first peak of the cumulative log returns, March 11th 2020, following the announcement of the pandemic the returns for all risk appetites fell. However, after the initial shock from the stock market all risk appetites including the lowest risk started to increase again and ended up being higher than the previous peak. This would suggest that the announcement had an influential depreciating impact on the stocks. Since q is a scalar from equation (1), we find that the difference in the cumulative log returns to be 5.339 between each risk appetite.

We can examine the performance by comparing it to keeping an equally weighted long position only portfolio. Figure (7) displays the returns of an equally weighted long position portfolio. This means the returns mimic the average returns across all 4 assets. It is greatly influenced by the pandemic, and although it returns to positive returns, most investors would have cut their losses in response to the shock once they dropped negative or at least would have shrank their portfolio.

Figure 7: Equally weighted portfolio long position returns

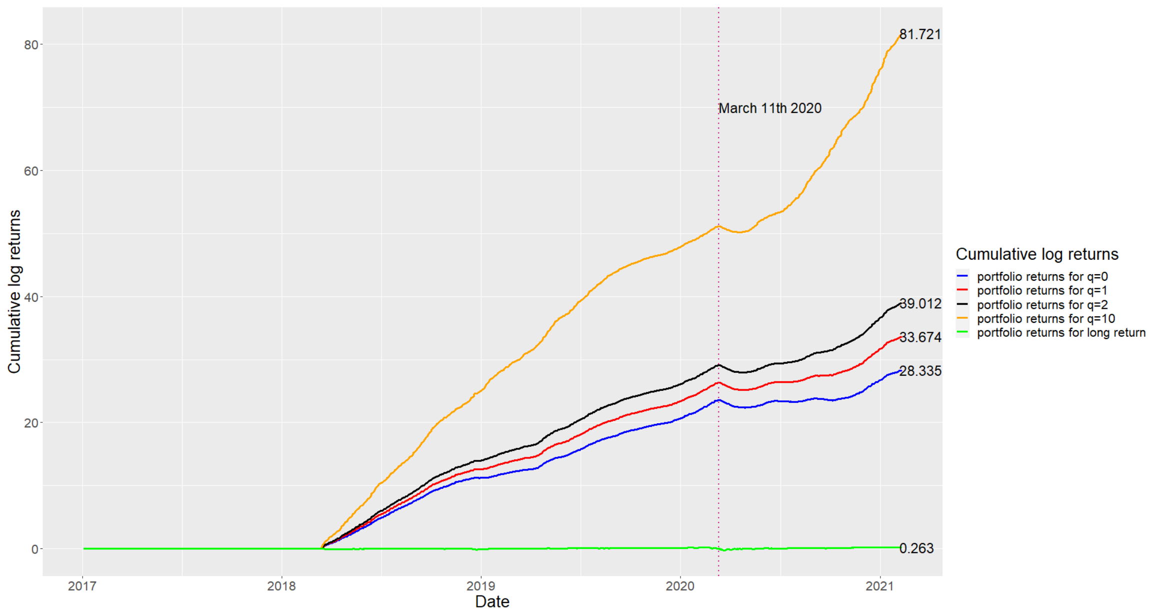

Figure 8: Combined graph of mean-variance portfolios and equally weighted long position

Figure (8) is the two previous figures combined and shows the portfolio returns for a long position as well as various risk appetites.

It can be seen that the long position portfolio is a flat line compared to the returns available from our modelled portfolio. We see over 100 times the return on investment when comparing the cumulative log returns of the minimal risk portfolio to the long position equally weighted portfolio. These are strong results, showing our portfolio will significantly out-perform a long only portfolio. This is from a purely statistical standpoint, as we have not taken into account the transaction costs of changing your portfolio on a daily basis, but we would imagine that the transaction costs in no way exceed the increase in return. We draw from this data to say that this method efficiently negates the impacts of the COVID-19 pandemic, and after a short time, can capitalise on the subsequent recovery.

3.2 EWMA-Variance Portfolio

Our results from an EWMA forecasted variance portfolio showed that for a minimal value of risk, we would obtain a 44.184 value of cumulative log returns. When we increase the risk value, we see that the volatility of the portfolio goes up throughout the trading window. We can see in Figure (9) that the daily return on our different risk values are extremely volatile. Fortunately the portfolio does have more positive days than negative days, showing that the responsiveness of the HMA forecasts help to reflect the weightings and keep the portfolio positive. This is a realistic representation of risk and shows that when you increase the risk value, the portfolio reflects that risk. Increasing the risk value does not significantly impact the overall cumulative return though, as shown by Figure (10) which shows that the risk does not necessarily outweigh the rewards. We always want to see a more realistic representation, to avoid over-simplification, and avoid the danger of overconfidence in the market. All in all, this method is successful in negating, and capitalising on the impacts of COVID-19, while retaining a decent return over the pre-COVID-19 data.

Figure 9: Forecasted returns based on EWMA

Figure 10: Forecasted portfolio returns based on cumulative sum of EWMA

3.3 Cointegrated Relation Portfolio

The portfolio generated by the Cointegration Relationships method shows to be useful but not significantly efficient compared to the other methods we have demonstrated. The fact that there are no cointegrating relationships shows that using this method would have been ruled out in the first instance but our interest in how useful it could be without such relationships has shown that it still has some uses. The process returned 3.339 value of log returns over the entire portfolio. From a realistic standpoint, we have not taken into account the costs of a daily adjustment of a portfolio, therefore these results are of a purely statistical view. Individually, the Financial

Services had the worst performing section of the portfolio, but still managed to hold its own. This shows that the portfolio correctly established the poor performance of that sector and adjusted its weights accordingly. The portfolio overall has performed better than the long only evenly weighted portfolio, so we can consider this a success, but there were still risky parts which would affect the confidence of an investor in a normal circumstance. In using this method as a way of mitigating the COVID-19 pandemic, we would not suggest it, as we used a restricted trend model only because we knew about the COVID-19 pandemic occurring and that lends trend information to our findings. If you were to try to use this method, with no co-integrating relationships to circumvent the impacts of COVID-19, there are better ways to do so.

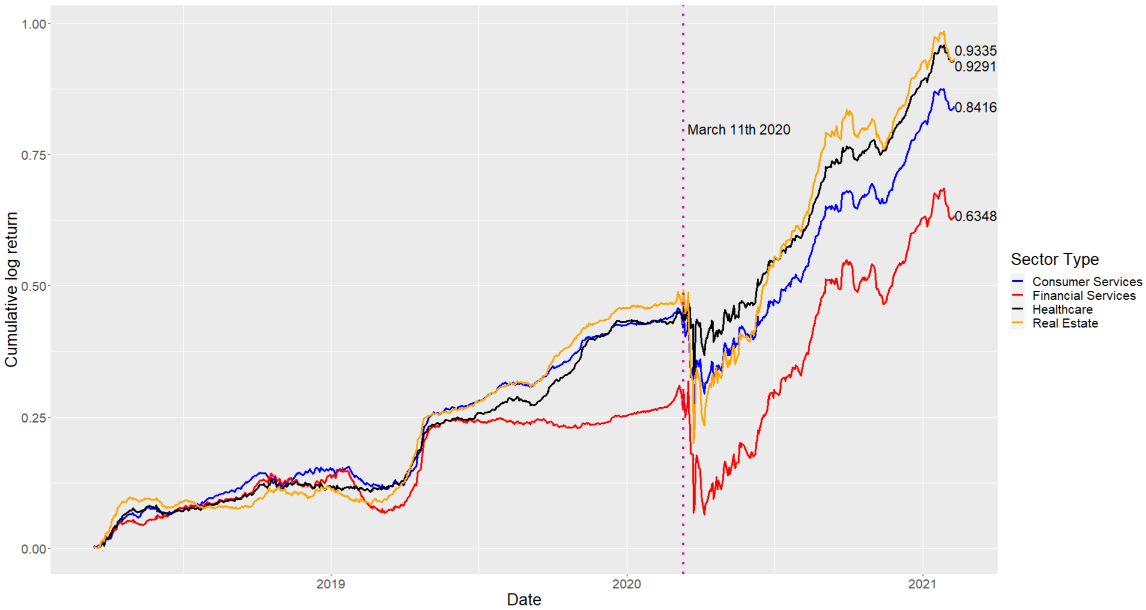

Figure 11: Cointegrated Relation portfolio returns for the four assets

4 Discussion

4.1 Portfolio Performance Comparison

To analyse the performance of the various portfolios, we can compare the results of their cumulative log returns. We first look at the results as displayed in Figure (7) for a long-equally weighted portfolio as a baseline. This type of portfolio produced minimal returns of only 0.263. The cointegrated-variance portfolio produces 3.339 which despite being an improvement on the long only portfolio, the returns are minimal, and could be risky, due to the reliance on cointegrating relationships which don't exist. We need to keep in mind that this could be a lucky performance, and this may not carry over to another data set. Coming to our one-step-ahead conditional mean variance portfolio (Section DCC), we see that we get significant returns throughout the entire trading window, where the risk appetite shows nothing but increased returns and this could be due to the oversimplification of the forecast. When compared to the returns available from the EWMA forecast, we see that for a set increase in risk value, the conditional mean portfolio outperforms the EWMA at a point of q = 10. This is most likely due to the oversimplification of the forecast and the risk not reflecting properly. The risk representation in the EWMA forecast is accurate and good to see. We get a significant return for low risk and the risk in this case does not outweigh the reward, considering that the COVID-19 pandemic was a significant impactor on the US equity market, and it created volatility which is hard to overcome.

Table (1) contains the results of both the mean-variance portfolio and the EWMA-variance portfolio for four risk appetites.

Risk Appetite	Mean-Variance Portfolio	EWMA Portfolio
q = 0	28.335	44.185
q = 1	33.674	44.860
q = 2	39.012	45.535
q = 10	81.721	50.936

Table 1: Cumulative Log Returns of Portfolios

Both of these portfolios completely outdo both the long-equally weighted and the cointegrated variance portfolios. It can be seen that for a lower risk appetite, the EWMA portfolio outperforms the mean-variance portfolio. However, the differences between various appetites for the EWMA are smaller than that for the mean variance. Therefore at a higher risk appetite the mean-variance portfolio will produce greater returns, as seen with the high risk appetite of q = 10. We estimate that a risk appetite of q = 4 would see the mean-variance portfolio being the better strategy. This does however benefit from the oversimplification of the forecast, so in another circumstance, it may not perform as well as expected.

4.2 Conclusion

The COVID-19 pandemic has undeniably affected and still continues to affect the world today in all aspects of life. Its global nature has meant that all countries have had a share in this. The US in particular has suffered socially and economically. In terms of the equity market, we have demonstrated that although it has had a noticeable effect during the start and peak of the pandemic, the market has recovered though with added volatility as new information concerning the pandemic continues to influence.

Although this event caused a market-wide crash, it was possible to survive and even profit from the pandemic's effect as we have shown. Comparing between our various portfolios created, our investigation has found our mean-variance portfolio to produce the most profitable returns with a higher risk appetite. However, given the uncertain and risky nature of the pandemic it would be logical to utilise the EWMA-variance portfolio as it returns better results for a lower risk requirement. For further study, we would take different equities that may have correlation relationships, and analyse them with Johansen's approach, to decide if we could accurately negate the impacts of the pandemic using that method. We could then use covariance portfolios on that data to see if they are still useful on correlated relationships as well.

Acknowledgments

We would like to acknowledge Dr Alessandro Cardinali for his guidance on the project and Dr Julian Stander for his assistance with our R coding graphical queries.

5 Appendix

5.1 References

References

[1] https://www.who.int/director-general/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19---11-march-2020
(accessed 30-04-2021)

[2] https://www.nytimes.com/article/coronavirus-timeline.html
(accessed 30-04-2021)

[3] https://uk.finance.yahoo.com/quote/IYH?p=IYH
(accessed 30-04-2021)

[4] https://uk.finance.yahoo.com/quote/IYC?p=IYC&.tsrc=fin-srch
(accessed 30-04-2021)

[5] https://uk.finance.yahoo.com/quote/IYG?p=IYG&.tsrc=fin-srch
(accessed 30-04-2021)

[6] https://uk.finance.yahoo.com/quote/IYR?p=IYR&.tsrc=fin-srch
(accessed 30-04-2021)

[7] M. Levinson, Guide to Financial Markets, The Economist Newspaper, London, 2018.

[8] https://www.ishares.com/us
(accessed 30-04-21)

[9] https://siblisresearch.com/data/us-stock-market-value/
(accessed 30-04-2021)

[10] https://www.nasdaq.com/articles/us-stock-market-biggest-most-expensive-world-us-economy-not-most-productive-2018-04-02
(accessed 30-04-2021)

[11] A. Cardinali (2018). Financial Statistics, MATH3623 Module Book, University of Plymouth.

[12] https://alanhull.com/hull-moving-average
(accessed 07/05/2020)

5.2 R code

The code for the DCC estimation is presented here:

```
load('USequity.R')

#Installing packages
#as ccgarch is no longer available
#we had to use an archive copy
install.packages("ccgarch")
install.packages("quantmod")
install.packages("tseries")
install.packages("ggplot2")
install.packages("tidyverse")
install.packages("GGally")
install.packages("urca")
install.packages("vars")
install.packages('pracma')
install.packages('PerformanceAnalytics')

library(ccgarch)
library(quantmod)
library(tseries)
library(MASS)
library(ggplot2)
library(tidyverse)
library(GGally)
library(urca)
library(pracma)
library(TTR)
library(PerformanceAnalytics)
library(vars)

###Plotting the starting data###
us.sect_plot<-function(){
  #Making us.sect into a data frame
  us_sect_df <- fortify(us.sect)

  # sorts the data into a 3 columns table;
  #date,sector and price
  us_sect_long_df <- us_sect_df %>%
    pivot_longer(cols = 2:5,
                 names_to = "Sector",
                 values_to = "Price")

   #Improving names of the sectors by creating a new
  #column(Sector_new) and renaming the sectors
  us_sect_long_df <- us_sect_long_df %>%
    mutate(Sector_new = factor(Sector,
                               levels = c("ConsumerServ",
```

```
                                                "Financials",
                                                "Healtcare",
                                                "RealEstate"),
                                   labels = c("Consumer Services",
                                                "Financial Services",
                                                "Healthcare",
                                                "Real Estate")))
  # Plotting the graph using ggplot
  us_sect_plot<-ggplot(us_sect_long_df,
                       aes(x = Index,
                           y = Price,
                           colour = Sector_new)) +
    geom_line(size=1.25) +
    theme(text = element_text(size=20))+
    scale_color_manual(values=c("Blue","Red","black","orange"))+
    labs(x = "Date",
         y = "Closing Price($)",
         colour = "Sector Type")+
    #Adding the start of Covid line
    geom_vline(xintercept = as.numeric(
                            us_sect_long_df$Index[3205]),
               linetype="dotted",color = "violetred", size=1.5)+
    annotate("text",x=c(us_sect_long_df$Index[3250]),y=250,
             label=c("March 11th 2020"),size=6,hjust=0)+
    #Adding First and last closing price for each Sector
    annotate("text",x=c(us_sect_long_df$Index[1]),y=140,
             label=c("145.70"),size=6,hjust=0)+
    annotate("text",x=c(us_sect_long_df$Index[2]),y=83,
             label=c("77.25"),size=6,hjust=0)+
    annotate("text",x=c(us_sect_long_df$Index[3]),y=58,
             label=c("51.115"),size=6,hjust=0)+
    annotate("text",x=c(us_sect_long_df$Index[4]),y=34,
             label=c("37.8825"),size=6,hjust=0)+

    annotate("text",x=c(us_sect_long_df$Index[4129]),y=250,
             label=c("255.17"),size=6,hjust=0)+
    annotate("text",x=c(us_sect_long_df$Index[4130]),y=90,
             label=c("88.87"),size=6,hjust=0)+
    annotate("text",x=c(us_sect_long_df$Index[4131]),y=67,
             label=c("69.930"),size=6,hjust=0)+
    annotate("text",x=c(us_sect_long_df$Index[4132]),y=75,
             label=c("73.77"),size=6,hjust=0)
  return(us_sect_plot)
}
us.sect_plot()

###DCC###
#A function that calculates the DCC of the data set
DCC<-function(){
```

```r
#Log return vector step 1

muhat<-function(xts){
  logts<-log(xts)
  logtsdiff<-diff(logts)*100
  logtsdiff[is.na(logtsdiff)]<-0
  mu_hat <- apply(logtsdiff, 2, mean)
  return(mu_hat)
}

#Function 1 complete

mhut<-muhat(us.sect)

#Conditional Covariance Matrix equation step 2

h_return<-function(xts){
  a<-c(0.004,0.005,0.006,0.007)
  A<-diag(0.2,4)
  B<-diag(0.35,4)
  inip<-c(0.15,0.8)
  rtnval<-dcc.estimation(inia=a,iniA=A,iniB=B,ini.dcc=inip,
                         dvar=xts,model='diagonal')
  hmtx<-rtnval$h
  DCCmtx<-rtnval$DCC
  htail<-sqrt(tail(hmtx,n=1))
  tDCCmtx<-tail(DCCmtx,1)
  R=matrix(tDCCmtx,nrow=4,byrow=T)
  DCCM <- (t(htail)%*%htail)*R
  return(solve(DCCM))

}
Health<-  us.sect$Healtcare

Cosumer<-us.sect$ConsumerServ

h_return(us.sect)



#Weights equation Step 3

weights<-function(xts,q){
  mut<-muhat(xts)
  Hinv <-h_return(xts)
  onea <- c(1,1,1,1)
  a<- t(onea) %*% Hinv %*% onea
  b<- t(onea) %*% Hinv %*% mut
  (Hinv %*% onea)/(c(a))+q*(Hinv%*%(mut-c((b/a)))
```

```
}

#function 3 complete

weights(us.sect,1)

#portfolio Returns Function Step 4

Portret<- function(xts,q){
  wi<- weights(xts,q)
  ei<- muhat(xts)
  ret<- c(ei%*%wi)
  return(ret)

}
Portret(us.sect,1)



################
#===============# Section for long evenly weighted returns.
#take the first 300 values to zero
################## Take values of max and min returns,
#against the weighted

logret<-diff(log(us.sect))
logret$Healtcare[1:300]<-0
logret$RealEstate[1:300]<-0
logret$Financials[1:300]<-0
logret$ConsumerServ[1:300]<-0
w<-c(0.25,0.25,0.25,0.25)
ret.long<-cumsum(logret%*%w)
plot.ts(ret.long)

###############################
#=============================# Section for q value 0
###############################

portrets.q0<- rep(0,1033)

window=300


for(t in(window+1):1033){
  winxts<- us.sect[(t-window):t]

  portrets.q0[t]<- Portret(winxts,0)
}
portrets.q0
```

```
port.retq0<-cumsum(portrets.q0)
plot.ts(port.retq0)
weight.zero<-weights(us.sect,0)
################################
#================================# Section for q value 1
################################

portrets<- rep(0,1033)

window=300


for(t in(window+1):1033){
  winxts<- us.sect[(t-window):t]

  portrets[t]<- Portret(winxts,1)
}

weight.one<-weights(us.sect,1)

port.ret<-cumsum(portrets)




################################
#==========================#    section for q value 2
##############################
portrets.q2<- rep(0,1033)
window=300
for(t in(window+1):1033){
  winxts<- us.sect[(t-window):t]

  portrets.q2[t]<- Portret(winxts,2)
}
weight.two<-weights(us.sect,2)
port.retq2<-cumsum(portrets.q2)


################################
#==============================# q value of 3.
################################

portrets.q10<- rep(0,1033)
window=300
for(t in(window+1):1033){
  winxts<- us.sect[(t-window):t]

  portrets.q10[t]<- Portret(winxts,10)
```

```
}

weight.three<-weights(us.sect,10)

port.retq10<-cumsum(portrets.q10)


##############################
#================================#  weights given for various
#risk appetite parameters
################################ q=0

us.sect.weights<-us.sect

window=300
for (t in 1:1033){

  us.sect.weights$Healtcare[t]<-0
  us.sect.weights$RealEstate[t]<-0
  us.sect.weights$Financials[t]<-0
  us.sect.weights$ConsumerServ[t]<-0


}
us.sect.weights.q0<-us.sect.weights

for (t in (window+1):1033){

  winxts<-us.sect[(t-window):t]
  weights.test<-weights(winxts,0)
  us.sect.weights.q0$Healtcare[t]<-weights.test[1]
  us.sect.weights.q0$RealEstate[t]<-weights.test[2]
  us.sect.weights.q0$Financials[t]<-weights.test[3]
  us.sect.weights.q0$ConsumerServ[t]<-weights.test[4]


}

##############################
#================================#  weights given for various
#risk appetite parameters
################################ q=1
us.sect.weights.q1<-us.sect.weights

for (t in (window+1):1033){

  winxts<-us.sect[(t-window):t]
  weights.test<-weights(winxts,1)
  us.sect.weights.q1$Healtcare[t]<-weights.test[1]
```

```
  us.sect.weights.q1$RealEstate[t]<-weights.test[2]
  us.sect.weights.q1$Financials[t]<-weights.test[3]
  us.sect.weights.q1$ConsumerServ[t]<-weights.test[4]


}

###Plotting portfolio returns for various risk appetites###
#Putting data into data frames
port.retq0_df<-as.data.frame(port.retq0)
port.ret_df<-as.data.frame(port.ret)
port.retq2_df<-as.data.frame(port.retq2)
port.retq10_df<-as.data.frame(port.retq10)
us_sect_df <- fortify(us.sect)
#Deleting the healthcare,RealEstate,Financials and
#ConsumerServ columns from us.sect
# We did this as the dates we had was not everyday
#between 03-01-2017 and 09-02-2021
# as there is no trading on the weekends
date<-select(us_sect_df,-Healtcare,-RealEstate,-Financials,
             -ConsumerServ)

#Putting all of the portfolio returns into a data frame
q_df<-data.frame(date,port.retq0_df,port.ret_df,
                 port.retq2_df,port.retq10_df)

#Creates 3 columns; date,portfolio_returns
#and cumulative_log_returns
q_long_df <- q_df %>%
  pivot_longer(cols = 2:5,
               names_to = "portfolio_returns",
               values_to = "cumulative_log_returns")

#Improving names of the portfolio returns by creating
#a new column(portfolio_returns_2) and renaming the sectors
q_long_df_2<- q_long_df %>%
  mutate(portfolio_returns_2 = factor(portfolio_returns,
                    levels = c("port.retq0",
                               "port.ret",
                               "port.retq2",
                               "port.retq10"),
                    labels = c("portfolio returns for q=0",
                               "portfolio returns for q=1",
                               "portfolio returns for q=2",
                               "portfolio returns for q=10")))

#Creating the graph
q_plot<-ggplot(q_long_df_2,
               aes(x = Index,
```

```
                     y = cumulative_log_returns,
                     colour = portfolio_returns_2)) +
  geom_line(size=1.25) +
  theme(text = element_text(size=20))+
  scale_color_manual(values=c("Blue","Red","black","orange"))+
  labs(x = "Date",
       y = "Cumulative log returns",
       colour = "Cumulative log returns")+
  #Adding the start of Covid line
  geom_vline(xintercept = as.numeric(q_long_df_2$Index[3205]),
             linetype="dotted",color = "violetred", size=1.5)+
  annotate("text",x=c(q_long_df_2$Index[3220]),y=70,
           label=c("March 11th 2020"),size=6,hjust=0)+
  #Adding last value for each Sector
  annotate("text",x=c(q_long_df_2$Index[4129]),y=28.33528,
           label=c("28.335"),size=6,hjust=0)+
  annotate("text",x=c(q_long_df_2$Index[4130]),y=33.67388,
           label=c("33.674"),size=6,hjust=0)+
  annotate("text",x=c(q_long_df_2$Index[4131]),y=39.01248,
           label=c("39.012"),size=6,hjust=0)+
  annotate("text",x=c(q_long_df_2$Index[4132]),y=81.72132,
           label=c("81.721"),size=6,hjust=0)

###Plotting portfolio returns for various risk appetites
###and the long position###

#Putting long returns into a data frame
#we have data frames already for the
#different risk appetites cause of the previous plot

port.ret_long<-as.data.frame(ret.long)
#Deleting the healthcare,RealEstate,Financials
#and ConsumerServ columns from us.sect
# We did this as the dates we had was not everyday
#between 03-01-2017 and 09-02-2021
#as there is no trading on the weekends
date<-select(us_sect_df,-Healtcare,-RealEstate,-Financials,
             -ConsumerServ)

#Putting all of the portfolio returns into one data frame
q_all_df<-data.frame(date,port.retq0,port.ret,
                     port.retq2,port.retq10,port.ret_long)

#Creates 3 columns; date,portfolio_returns
#and cumulative_log_returns
q_all_long_df <- q_all_df %>%
  pivot_longer(cols = 2:6,
               names_to = "portfolio_returns",
               values_to = "cumulative_log_returns")
```

```
#Improving names of the portfolio returns by
#creating a new column(portfolio_returns_2)
#and renaming the sectors
q_all_long_df_2<- q_all_long_df %>%
  mutate(portfolio_returns_2 = factor(portfolio_returns,
           levels = c("port.retq0",
                      "port.ret",
                      "port.retq2",
                      "port.retq10",
                      "ret.long"),
           labels = c("portfolio returns for q=0",
                      "portfolio returns for q=1",
                      "portfolio returns for q=2",
                      "portfolio returns for q=10",
                      "portfolio returns for long return")))

#Creating the graph
q_all_plot<-ggplot(q_all_long_df_2,
                   aes(x = Index,
                       y = cumulative_log_returns,
                       colour = portfolio_returns_2)) +
  geom_line(size=1.25) +
  theme(text = element_text(size=20))+
  scale_color_manual(values=c("Blue","Red","black",
                              "orange","green"))+
  labs(x = "Date",
       y = "Cumulative log returns",
       colour = "Cumulative log returns")+
  #Adding the start of Covid line
  geom_vline(xintercept = as.numeric(q_all_long_df_2$
                                      Index[4009]),
             linetype="dotted",color = "violetred", size=1)+
  annotate("text",x=c(q_all_long_df_2$Index[4009]),y=70,
           label=c("March 11th 2020"),size=6,hjust=0)+
  #Adding last value for each Sector
  annotate("text",x=c(q_all_long_df_2$Index[5151]),y=28.33528,
           label=c("28.335"),size=6,hjust=0)+
  annotate("text",x=c(q_all_long_df_2$Index[5152]),y=33.67388,
           label=c("33.674"),size=6,hjust=0)+
  annotate("text",x=c(q_all_long_df_2$Index[5153]),y=39.01248,
           label=c("39.012"),size=6,hjust=0)+
  annotate("text",x=c(q_all_long_df_2$Index[5154]),y=81.721,
           label=c("81.721"),size=6,hjust=0)+
  annotate("text",x=c(q_all_long_df_2$Index[5154]),y=0.263376,
           label=c("0.263"),size=6,hjust=0)
q_all_plot
###Plotting portfolio weights for Q=0###

#Making us.sect.weights.q0_df into a data frame
```

```
us.sect.weights.q0_df <- fortify(us.sect.weights.q0)

# sorts the data into a 3 columns table; date,sector
#and Proportion
us.sect.q0_long_df <- us.sect.weights.q0_df %>%
  pivot_longer(cols = 2:5,
               names_to = "Sector",
               values_to = "Proportion")
#Improving names of the sectors by creating
#a new column(Sector_new) and renaming the sectors
us.sect.q0_long_df <- us.sect.q0_long_df %>%
  mutate(Sector_new = factor(Sector,
                             levels = c("ConsumerServ",
                                        "Financials",
                                        "Healtcare",
                                        "RealEstate"),
                             labels = c("Consumer Services",
                                        "Financial Services",
                                        "Health Care",
                                        "Real Estate")))
#Deleting the first 300 values of each sector
#as these are 0 due to backtesting
us.sect.q0_long_df2<-us.sect.q0_long_df[-c(1:1200),]

#Plotting the graph using ggplot
us.sect.q0_long_plot<-ggplot(us.sect.q0_long_df2,
                                  aes(x = Index,
                                      y = Proportion,
                                      colour = Sector_new)) +
  geom_line(size=1.25) +
  theme(text = element_text(size=20))+
  scale_color_manual(values=c("Blue",
                              "Red",
                              "black",
                              "orange"))+
  labs(x = "Date",
       y = "proportion of investment",
       colour = "Sector Type")+
  #Adding the start of Covid line
  geom_vline(xintercept = as.numeric(
                          us.sect.q0_long_df$Index[3205]),
             linetype="dotted",color = "violetred", size=1.5)+
  annotate("text",x=c(us.sect.q0_long_df$Index[3220]),y=3,
           label=c("March 11th 2020"),size=6,hjust=0)

###Plotting portfolio weights for Q=1###

#Making us.sect.weights.q1_df into a data frame
us.sect.q1_df <- fortify(us.sect.weights.q1)
```

```
# sorts the data into a 3 columns table; date,sector
#and Proportion
us.sect.q1_long_df <- us.sect.q1_df %>%
  pivot_longer(cols = 2:5,
               names_to = "Sector",
               values_to = "Proportion")

#Improving names of the sectors by creating
#a new column(Sector_new) and renaming the sectors
us.sect.q1_long_df <- us.sect.q1_long_df %>%
  mutate(Sector_new = factor(Sector,
                             levels = c("ConsumerServ",
                                        "Financials",
                                        "Healtcare",
                                        "RealEstate"),
                             labels = c("Consumer Services",
                                        "Financial Services",
                                        "Health Care",
                                        "Real Estate")))
#Deleting the first 300 values of each sector as these are 0
#due to backtesting
us.sect.q1_long_df2<-us.sect.q1_long_df[-c(1:1200),]


#Plotting the graph using ggplot
us.sect.q1_long_plot<-ggplot(us.sect.q1_long_df2,
                                aes(x = Index,
                                    y = Proportion,
                                    colour = Sector_new)) +
  geom_line(size=1.25) +
  theme(text = element_text(size=20))+
  scale_color_manual(values=c("Blue",
                              "Red",
                              "black",
                              "orange"))+
  labs(x = "Date",
       y = "proportion of investment",
       colour = "Sector Type")+
  geom_vline(xintercept = as.numeric(
                          us.sect.q1_long_df$Index[3205]),
             linetype="dotted",color = "violetred", size=1.5)+
  annotate("text",x=c(us.sect.q1_long_df$Index[3220]),y=3,
           label=c("March 11th 2020"),size=6,hjust=0)


###Correlation plots###
#Correlation plot for 31/01/2020 to 09/02/2021
pairsdata <- data.frame(us.sect$Healtcare[775:1033],
                        us.sect$RealEstate[775:1033],
```

```
                          us.sect$Financials[775:1033],
                          us.sect$ConsumerServ[775:1033])
  #Correlation plot for 03/01/2017 to 09/02/2021
  pairsdata1 <- data.frame(us.sect$Healtcare,
                           us.sect$RealEstate,
                           us.sect$Financials,
                           us.sect$ConsumerServ)
  #plotting pairsdata1
  pairsdata_plot1<-ggpairs(pairsdata1,
                           lower = list(continuous =
                                          wrap("smooth",
                                               alpha = 0.3,
                                               size=0.3)),
                          columnLabels = c("Healthcare",
                                           "Real Estate",
                                           "Financials",
                                           "Consumer Services"),
                        title = 'Pairs Correlation for US.Sect')

  #plotting pairsdata
  pairsdata_plot<-ggpairs(pairsdata,
                          lower = list(continuous =
                                         wrap("smooth",
                                              alpha = 0.3,
                                              size=0.3)),
                          columnLabels = c("Healthcare",
                                           "Real Estate",
                                           "Financials",
                                           "Consumer Services"),
  title = 'Pairs Correlation for US.Sect During COVID Pandemic')

  plot_list<-list(q_plot,
                  q_all_plot,
                  us.sect.q0_long_plot,
                  us.sect.q1_long_plot,
                  pairsdata_plot,
                  pairsdata_plot1)
  return(plot_list)
}
DCC()
```

The code for EWMA DCC estimation is presented here:

```
#A function that calculates
#Exponential Weighted Moving Average of the data set
EWMA<-function(){
  window<-300
  ret<-rep(0,1033)
  EWMA.DCC<-function(xts,q){
    for(t in(window+1):1033){
      winxts<- xts[(t-window):t]
      #Calculating the inverse conditional covariance matrix
      a<-c(0.004,0.005,0.006,0.007)
      A<-diag(0.2,4)
      B<-diag(0.35,4)
      inip<-c(0.15,0.8)
      rtnval<-dcc.estimation(inia=a,iniA=A,iniB=B,ini.dcc=inip,
                             dvar=winxts,model='diagonal')
      hmtx<-rtnval$h
      DCCmtx<-rtnval$DCC
      htail<-sqrt(tail(hmtx,n=1))
      tDCCmtx<-tail(DCCmtx,1)
      R=matrix(tDCCmtx,nrow=4,byrow=T)
      DCCM <- (t(htail)%*%htail)*R
      Hinv<-solve(DCCM)
      #Calculating the forecasts for the data set
      Health.Hma<-HMA(winxts$Healtcare,7)
      Real.Hma<-HMA(winxts$RealEstate,7)
      Finance.Hma<-HMA(winxts$Financials,7)
      Consumer.Hma<-HMA(winxts$ConsumerServ,7)
      us.HMA<-cbind(Health.Hma,Real.Hma,Finance.Hma,
                    Consumer.Hma)
      #Locating the most recent forecast in this case
      forecast<-as.numeric(tail(us.HMA,1))

      #Applying the weights equation to the data

      onea <- c(1,1,1,1)
      a<- t(onea) %*% Hinv %*% onea
      b<- t(onea) %*% Hinv %*% forecast
      wi<-(Hinv %*% onea)/(c(a))+q*(Hinv%*%(forecast-c((b/a)))

      #Calculating average returns and
      #applying the weights to them

      logts<-log(xts)
      logtsdiff<-diff(logts)*100
      logtsdiff[is.na(logtsdiff)]<-0
      ei<- apply(logtsdiff, 2, mean)
      ei
      ei%*%wi
```

```
    ret[t]<- c(ei%*%wi)

  }
  return(ret)
}
ewma.ret.q0<-EWMA.DCC(us.sect,0)
ewma.ret.q1<-EWMA.DCC(us.sect,1)
ewma.ret.q2<-EWMA.DCC(us.sect,2)
ewma.ret.q10<-EWMA.DCC(us.sect,10)


#Creating data frames
ewma.ret.q0_df<-as.data.frame(ewma.ret.q0)
ewma.ret.q1_df<-as.data.frame(ewma.ret.q1)
ewma.ret.q2_df<-as.data.frame(ewma.ret.q2)
ewma.ret.q10_df<-as.data.frame(ewma.ret.q10)
us_sect_df <- fortify(us.sect)

#Deleting the healthcare,RealEstate,Financials
#and ConsumerServ columns from us.sect
# We did this as the dates we had was not
#everyday between 03-01-2017 and 09-02-2021
# as there is no trading on the weekends
date<-select(us_sect_df,
             -Healtcare,
             -RealEstate,
             -Financials,
             -ConsumerServ)

#Putting all of the portfolio returns into a data frame
#along with date
ewma_df<-data.frame(date,
                    ewma.ret.q0,
                    ewma.ret.q1,
                    ewma.ret.q2,
                    ewma.ret.q10_df)

#Creates 3 columns; date,portfolio_returns
#and cumulative_log_return
ewma_long_df <- ewma_df %>%
  pivot_longer(cols = 2:5,
               names_to = "portfolio_returns",
               values_to = "log_returns")

#Improving names of the portfolio returns by
#creating a new column(portfolio_returns_2)
#and renaming the returns
ewma_long_df_2<- ewma_long_df %>%
  mutate(portfolio_returns_2 = factor(portfolio_returns,
```

```r
            levels = c("ewma.ret.q0",
                       "ewma.ret.q1",
                       "ewma.ret.q2",
                       "ewma.ret.q10"),
            labels = c("EWMA portfolio returns for q=0",
                       "EWMA portfolio returns for q=1",
                       "EWMA portfolio returns for q=2",
                       "EWMA portfolio returns for q=10")))

#Creating the graph
ewma_plot<-ggplot(ewma_long_df_2,
                  aes(x = Index,
                      y = log_returns,
                      colour = portfolio_returns_2)) +
  geom_line(size=1.25) +
  theme(text = element_text(size=20))+
  scale_color_manual(values=c("black",
                              "blue",
                              "green",
                              "orange"))+
  labs(x = "Date",
       y = "log returns",
       colour = "Risk appetite")+

  #Adding the start of COVID-19 line
  geom_vline(xintercept = as.numeric(
                 ewma_long_df_2$Index[3205]),
             linetype="dotted",color = "violetred", size=1.5)+
  annotate("text",x=c(ewma_long_df_2$Index[3205]),y=2,
           label=c("March 11th 2020"),size=6,hjust=0)


###Calculating the cumulative sum
ewma.ret.q0_cumsum_df<-as.data.frame(cumsum(ewma.ret.q0))
ewma.ret.q1_cumsum_df<-as.data.frame(cumsum(ewma.ret.q1))
ewma.ret.q2_cumsum_df<-as.data.frame(cumsum(ewma.ret.q2))
ewma.ret.q10_cumsum_df<-as.data.frame(cumsum(ewma.ret.q10))

ewma_cumsum_df<-data.frame(date,
                           ewma.ret.q0_cumsum_df,
                           ewma.ret.q1_cumsum_df,
                           ewma.ret.q2_cumsum_df,
                           ewma.ret.q10_cumsum_df)

#Creates 3 columns; date,portfolio_returns
#and cumulative_log_return
ewma_cumsum_long_df <- ewma_cumsum_df %>%
  pivot_longer(cols = 2:5,
               names_to = "portfolio_returns",
```

```
                  values_to = "cumulative_log_returns")

  #Improving names of the portfolio returns by creating
  #a new column(portfolio_returns_2)
  #and renaming the returns
  ewma_cumsum_long_df_2<- ewma_cumsum_long_df %>%
    mutate(portfolio_returns_2 = factor(portfolio_returns,
                  levels = c("cumsum.ewma.ret.q0.",
                             "cumsum.ewma.ret.q1.",
                             "cumsum.ewma.ret.q2.",
                             "cumsum.ewma.ret.q10."),
                  labels = c("EWMA portfolio returns for q=0",
                             "EWMA portfolio returns for q=1",
                             "EWMA portfolio returns for q=2",
                             "EWMA portfolio returns for q=10")))

  #Creating the graph
  ewma_cumsum_plot<-ggplot(ewma_cumsum_long_df_2,
                            aes(x = Index,
                                y = cumulative_log_returns,
                                colour = portfolio_returns_2)) +
    geom_line(size=1.25) +
    theme(text = element_text(size=20))+
    scale_color_manual(values=c("black",
                                "blue",
                                "green",
                                "orange"))+
    labs(x = "Date",
         y = "cumulative log returns",
         colour = "Risk appetite")+
    #Adding the start of COVID-19 line
    geom_vline(xintercept = as.numeric(
          ewma_cumsum_long_df_2$Index[3205]),
          linetype="dotted",color = "violetred", size=1.5)+
    annotate("text",x=c(ewma_cumsum_long_df_2$Index[3205]),
             y=47,label=c("March 11th 2020"),size=6,hjust=0)+
    #Adding last value for each Sector
    annotate("text",x=c(ewma_cumsum_long_df_2$Index[4129]),
             y=43,label=c("44.18484"),size=6,hjust=0)+
    annotate("text",x=c(ewma_cumsum_long_df_2$Index[4130]),
             y=44.85993,label=c("44.85993"),size=6,hjust=0)+
    annotate("text",x=c(ewma_cumsum_long_df_2$Index[4131]),
             y=46.5,label=c("45.53501"),size=6,hjust=0)+
    annotate("text",x=c(ewma_cumsum_long_df_2$Index[4132]),
             y=50.93566,label=c("50.93566"),size=6,hjust=0)
  EWMA_list<-list(ewma_plot,ewma_cumsum_plot)
  return(EWMA_list)
}
EWMA()
```

The code for the Cointegrated Relationships is presented here:

```
#A function that calculates the co-integration
#of  the  data  set
cajo<-function(){
  lsect<-log(us.sect) #Take log of data
  #Locates the lag length for the VAR
  var.fit<-VAR(lsect,lag.max=6,ic='AIC')
  var.fit$p
  #Lag length is 5
  us.coint<-ca.jo(lsect, type='eigen',ecdet="trend",K=5)
  # The data can be considered to follow a restricted trend.
  #This is due to the falls and steady recovery
  #due to the pandemic.
  summary(us.coint)
  #This shows there are no co-integrations. In order to
  #continue with this line of forecasting, we will use r=1,
  #as it has the smallest value between the test statistic and
  #the critical values.
  us.var<-vec2var(us.coint, r=1)
  # We use one step ahead for these forecasts for accuracy.
  pred<-predict(us.var, n.ahead=1)
  tai.sect<-tail(lsect,n=1)
  pred.mat<-c(pred$fcst$Healtcare[1],pred$fcst$RealEstate[1],
          pred$fcst$Financials[1],pred$fcst$ConsumerServ[1])
  ans<-pred.mat-tai.sect
  #Algorithmic window test for our forecast


  fore.ret<-NULL
  window<-300
  us.ret<-us.sect


  for(t in(window+1):1033){
    winxts<-lsect[(t-window):t]
    us.coint<-ca.jo(winxts, type='eigen',ecdet="trend",K=5)
    us.var<-vec2var(us.coint, r=1)
    pred<-predict(us.var, n.ahead=1)
    tai.sect<-tail(winxts,n=1)
    pred.mat<-c(pred$fcst$Healtcare[1],
                pred$fcst$RealEstate[1],
                pred$fcst$Financials[1],
                pred$fcst$ConsumerServ[1])
    val<-pred.mat-tai.sect

    fore.ret<-rbind(fore.ret,val)
  }
```

```
#Here we see from our data that we aren't able to correctly
#predict and circumvent the impactsof the pandemic.
#This means that our original idea of GARCH modelling
#with an adaptive portfolio gets better results.

###Plotting Cajo graph###

#Making the cumulative sum of fore.ret into a data frame
fore.ret_df<-fortify(cumsum(fore.ret))
# sorts the data into a 3 columns table; date,sector
#and Cumulative_log_return
fore.ret_long_df <- fore.ret_df %>%
  pivot_longer(cols = 2:5,
               names_to = "Sector",
               values_to = "Cumulative_log_return")
#Improving names of the sectors by creating
#a new column(Sector_new) and renaming the sectors
fore.ret_long_df <- fore.ret_long_df %>%
  mutate(Sector_new = factor(Sector,
                             levels = c("ConsumerServ",
                                        "Financials",
                                        "Healtcare",
                                        "RealEstate"),
                             labels = c("Consumer Services",
                                        "Financial Services",
                                        "Healthcare",
                                        "Real Estate")))
# Plotting the graph using ggplot
fore.ret_plot<-ggplot(fore.ret_long_df,
                      aes(x = Index,
                          y = Cumulative_log_return,
                          colour = Sector_new)) +
  geom_line(size=1.25) +
  theme(text = element_text(size=20))+
  scale_color_manual(values=c("Blue","Red","black","orange"))+
  labs(x = "Date",
       y = "Cumulative log return",
       colour = "Sector Type")+
  #Adding the start of Covid line
  geom_vline(xintercept = as.numeric(
              fore.ret_long_df$Index[2005]),
             linetype="dotted",color = "violetred", size=1.5)+
  annotate("text",x=c(fore.ret_long_df$Index[2020]),y=0.8,
           label=c("March 11th 2020"),size=6,hjust=0)+
  #Adding First and last closing price for each Sector
  annotate("text",x=c(fore.ret_long_df$Index[2929]),y=0.92,
           label=c("0.9291"),size=6,hjust=0)+
  annotate("text",x=c(fore.ret_long_df$Index[2930]),y=0.95,
           label=c("0.9335"),size=6,hjust=0)+
```

```
    annotate("text",x=c(fore.ret_long_df$Index[2931]),y=0.6348,
             label=c("0.6348"),size=6,hjust=0)+
    annotate("text",x=c(fore.ret_long_df$Index[2932]),y=0.8416,
             label=c("0.8416"),size=6,hjust=0)
  return(fore.ret_plot)
}
#Runs the cajo function
cajo()
```