XI'AN JIAOTONG-LIVERPOOL UNIVERSITY

西 交 利 物 浦 大 学

ACF 313: Risk Management for Business Groupwork project:

A copula-based simulation model for big tech and defensive portfolios risk management

Members：

Pinyi.Wu	1507854
Shiqi.Dong	1507633
Xiaowen.Man	1509289
Yifan.Yang	1508632

December 16, 2018.

Content

1. Introduction *(Pinyi.Wu 70%, Shiqi.Dong 10%, Xiaowen.Man 10%, Yifan.Yang 10%)*

To conduct portfolio risk management, it is important to identify various categories of financial risks which include but not limited to market risk, credit risk, liquidity risk and operational risk (Maverick, 2018). Market risk, known as systematic risk, is the likelihood that the potential losses the investors would endure due to changes in the financial market factors (Chen, 2018a). Examples of market risk could be the recession of economy which would result in the poor performance of the overall market and influence all asset classes. Thus, market risk is undiversifiable and unpredictable (Nickolas, 2018). In this paper, the market risk will be discussed and explored by creating two portfolios incorporating four stocks of big tech companies and four stocks belonging to defensive industries.

In Oct. 2018, the global stock indices suffered their worst monthly performance since the last 9 years and global equity markets were much more volatile than before (J.P. Morgan, 2018). However, under the circumstance of the President Trump's tax reforms, multiple US tech companies have gained advantages to reduce the cost and repatriated billions in overseas (Waters and Johnson, 2018). Apple Inc. significantly benefited from the policy with respect to the earning growth and shareholder capital returns. As Tae indicated in 2018, Apple conducted stock buybacks to return near $25 billion to shareholders per quarter. Additionally, several big tech companies in the United States including Apple, Alphabet, Cisco, Microsoft and Oracle spent over $115 billion buying back the stocks in the first three quarters in 2018 and increased their capital investment by 42 percent compared with the same period in previous year (Kim, 2018). Stock buyback is normally a promising news as repurchasing the outstanding shares could reduce the cost of capital and it may imply a pleasant performance of the issuing companies (White, 2018). Moreover, large amount of stock buyback will increase the earnings per share and a high value of it will indicate potential investing values (Vartan, 1984).

As a consequence, it is reasonable to create a portfolio with these big tech companies as it has strong support by Anastasia Amoroso, the Head of GIO Investment Strategy at J.P.

Morgan Private Bank, who indicated that software and IT services providers will embrace a positive future with significant investment in data privacy (Amoroso, 2018).

In this report. Apple Inc. (AAPL), Alphabet (GOOGL), Microsoft (MSFT) and Cisco (CSCO) will be selected to form big tech portfolio as they are all traded on NASDAQ stock market and the financial data is provided by Yahoo Finance.

As stated above, global economy is undergoing a poor performance as it could be reflected by performance of S&P 500 index which plunged 6.7 percent from Oct. 3rd to Oct.11th in 2018 (Imbert, 2018a). As a consequence, to be more focused on defense, the portfolio strategy will differ from the previous one. As the Jim Paulsen, the Chief investment strategist of the Leuthold Group, indicated that although the recession stage has not arrived, the bull market of U.S. large-capitalization stocks is in its middle period (Imbert, 2018a). Thus, defensive equity sectors such as utilities, consumer staples and real estate should be considered to create a defensive portfolio. In this paper, Physicians Realty Trust (DOC/PRT) will be selected as it is in recession-proof healthcare industry and it will benefit from a growing demographic (Frankel, 2018). With regard to utilities stocks, they gained at least 2% in four continuous weeks and previous record was three successive weeks in 2004 (Intelligent Income,2018). This is a positive signal and consequently, NextEra Energy (NEE) and American Electric Power (AEP) will be chose. Besides, the stock of Johnson & Johnson (JNJ) will be considered as consumer staple stock. These four stocks which diversify across sectors will contribute to a defensive portfolio.

This report aims to evaluate the above two portfolios (big tech and defensive) by multiple methods. Firstly, based on the alpha and beta values, we categorize eight stocks to form two portfolios, then we use the log returns of stocks to fit the normal and t distribution to determine the optimal marginal distribution. Next, normal and t copula are employed to model the dependence structure of the stocks in the same portfolio. According to log-likelihood and AIC values, t distribution and t copula is considered as the more appropriate candidates in the above two stages respectively. Then we compute VaR by using Monte Carlo simulation and compare the values with the actual daily profit and loss to evaluate the calculated VaR, and simply discuss the potential reasons for the violations. Finally, we compare the two portfolios and give a conclusion.

Section 2 will offer the insights into pertinent literatures, and section 3 presents detailed methods adopted in this report. Section 4 explains the details for marginal distribution fitting and copula fitting. Simulating VaR and backtesting for VaR is positioned at section 5 and 6 respectively. Section 7 compares the two portfolios in various perspectives, and a final conclusion will be given at section 8.

2. Literature review *(Pinyi.Wu 10%, Shiqi.Dong 10%, Xiaowen.Man 10%, Yifan.Yang 70%)*

2.1 Financial Risk Management

Financial risk is any types of risks associated with financing which may result in loss and default of companies. Furthermore, financial risk management is using financial instruments to decrease or avoid assets loss in a firm.

Theoretically, it is known that the value of a firm is irrelevant with the risk structure of itself from the Modigliani-Miller theorem (Christoffersen, 2012). However, the reality of capital market is stricter, which means risk financial risk management is necessarily required for a corporate. It is important for a firm to focus on risk management, because good risk management improves the performance of the firm. According to Christoffersen (2012), some researchers have found that less volatile cash flows cause lower costs of capital and increase investment, which proves that a firm using risk management would have better performance than the firm which did not.

There are different types of risks faced by a company. Firstly, market risk is the result of undesired movements in market prices of assets in a portfolio. Moreover, market risk management is aiming to handle the estimation of loss distribution of a portfolio of assets over a fixed time horizon. Secondly, credit risk is defined as the risk arisen from the possibility that a counterparty may not partly or fully fulfill its obligation on time, which causes the transactions default (Christoffersen, 2012). Thirdly, liquidity risk is a specific risk

in low liquidity markets, such as low trading volume and large bid-offer spreads (Christoffersen, 2012). Liquidity refers to the ability of a company to make cash payments in transactions. Lastly, operational risk is the risk because of transaction errors, inadequate controls, and failure of management. Operational risk should be reduced and ideally eliminated though it offers less returns (Christoffersen, 2012).
Overall, financial risk management can effectively reduce risks and improve the performance of firms.

2.2 VaR

Value-at-risk (VaR) is a powerful tool that can assess market risk in real time, which provides important suggestion when making decisions in trading and hedging. Referring to Allen and Powell (2007), it was first introduced by JP Morgan in 1994 and promoted by amendments to the Basel Accord in 1996 which required banks to set aside capital for predicting market risk. VaR is an attractive risk measure based on the variance-covariance parametric model, which makes it a standard tool for financial entities (Allen and Powell, 2007).

VaR can calculate maximum expected losses over a given time period at a given tolerance level (Allen and Powell, 2007). It is a non-convex and discontinuous function of the confidence level ɑ for discrete distributions (Rockaffar and Uryasev, 1999). There are 3 ways to calculate VaR. Firstly, the variance-covariance method estimates VaR when assuming a normal distribution, which is the most widely used method. Secondly, the historical method divides historical losses into best-to-worst categories and calculates VaR based on the assumption of history iteration. Thirdly, the Monte Carlo method, which is defined as simulating multiple random scenarios.

However, VaR has poor mathematical characteristics, including a lack of sub-additivity and convexity (Artzner et al., 1999). Some pervious research has found the weaknesses of VaR. According to Rockafellar and Uryasev (1999), VaR combined with two portfolios may be greater than the sum of risks for each portfolio and it is also difficult to optimize when it is calculated from scenarios. Furthermore, some pervious researchers have showed that VaR may not perform well as a function of portfolio positions and may result in multiple local extrema, which can be a major obstacle to trying to determine an optimal combination of positions or even a specific combination of VaR (Rockafellar and Uryasev, 1999).

As a consequence, although VaR has good performance in market risk management and was adopted as a standard risk analysis in financial area, it is necessary to use another compensatory model to compare the result of VaR for a more reasonable result.

2.3 Normal distribution and Skewed Student's t distribution

According to Kouwenberg et al. (2009), normal VaR model is defined as a standard VaR model with normally distributed error terms, representing normal asset return distributions with zero skewness and no excess kurtosis. However, in real-world cases, it may not be a realistic assumption for alternative asset classes. As a consequence, the skewed Student's t VaR model should be proposed. Comparing normal distribution and Student's t distribution, the student's t distribution can be used for historical time series with a large number of extreme observations far from the mean (Kouwenberg et al., 2009).

2.4 Copula and t-copula

Copula is also called Gaussian copula which can represent the dependence structure of the multivariate normal distribution (Demarta and McNeil, 2005). In the reference of Nelsen (2006), copulas were identified as the functions that join or couple multivariate distribution functions to their uniform one-dimensional marginal distribution functions.

Referring to Wei and Zhang (2004), copula is a new methodology that measures dependence between random variables and also be used in the study of characteristics of financial markets, portfolio aggregation and risk analysis.

According to Demarta and McNeil (2005), the t copula is defined as a model to represent the dependence structure reserved in a multivariate t distribution. The t copula was attractive in the area of financial return data, because of its ability to capture the dependent extreme value (Demarta and McNeil, 2005). Compared to the Gaussian copula, the fit of t copula is often good and is better than that of Gaussian copula.

3. Methodology *(Pinyi.Wu 30%, Shiqi.Dong 30%, Xiaowen.Man 30%, Yifan.Yang 10%)*

3.1 Beta coefficient and Alpha

Beta coefficient is one of the principles of stock selection which aims to form different types of portfolios; Alpha will be employed to verify whether a specific stock outperforms the market index (e.g. S&P500).

Beta coefficient (β), a widely used method to evaluate risk (Boskovska and Svrtinov, 2016), is employed to measure the risk volatility concerning the required return of a specific asset compared to the volatility of the average of the overall market portfolio. Its value can be derived by the following:

$$\beta = \frac{Cov_{i,m}}{\sigma_m^2} = \frac{(Ri - \overline{Ri})(Rm - \overline{Rm})}{(Rm - \overline{Rm})^2}$$

$Cov_{i,m}$: a covariance of the yield of a certain share and yield of market portfolio (or market index like S&P500)

$\sigma^2 m$: variance of yield of the market portfolio (market index).

Ri : daily yield of share i;

$\overline{Ri}$: average yield of share i;

Rm : dailly yield of S&P 500;

$\overline{Rm}$: average yield of share S&P;

Referring to Radović and Vasiljević (2012), the stocks with β values greater than 1 will be categorized into big tech portfolios.

Values	Implication	Category	Example
β > +1	greater volatility concerning the return of this asset compared to the volatility of the average of the overall market portfolio	risky and aggressive; systematic risk is higher than the market as a whole	When β is 2, asset price will increase by 20% if the price of market portfolio increases by 10%
β < +1	less volatility concerning the required return of this asset compared to the volatility of	less risky and defensive;	When β is 0.5, asset price will increase by 5% if the

	the average of the overall market portfolio	systematic risk is lower	price of market portfolio increases by 10%

Table 1 The interpretation of different β values

Alpha is defined as the excess return on an investment relative to the return on a benchmark and it can be calculated by the following formulas. Usually, alpha is expressed as a percentage that illustrates how an investment performed relative to a benchmark index. For example, a positive alpha value, 5 (+5), means that the return of this portfolio outperformed the benchmark index's performance by 5% (CFI Education,2018).

$$r = R_f + \beta\,(R_m - R_f) + \alpha$$

$$\alpha = r - R_f - \beta\,(R_m - R_f)$$

	APPL	MSFT	GOOG	CSCO
BETA	1. 139716	1. 403659	1. 407882	1. 281424
ALPHA	0. 2990228	0. 8617935	0. 8703022	0. 6008469

	DOC	NEE	AEP	JNJ
BETA	0. 494812	0. 1869068	0. 07563511	0. 8025714
ALPHA	−1. 077989	−1. 733722	−1. 971505	−0. 4210983

Table 2 Alpha values of these stocks

Based on the alpha values of these stocks in Table 2, it can be concluded that stocks belonging to big tech portfolio outperformed the market portfolio in varying degrees; and defensive stocks underperformed the market portfolio, so this definitely verifies the assumption that big tech stocks may be worth to buy and hold, but this guess needs further investigation and robust numerical evidence.

	PRT	NEE	AEP	JNJ
PRT	1	0.3434772	0.4304687	0.2017552
NEE	0.3434772	1	0.2017552	0.1975702
AEP	0.4304687	0.2017552	1	0.1761077
JNJ	0.2017552	0.1975702	0.1761077	1

Table 3 Correlation matrix among return for defensive portfolio

	AAPL	MSFT	GOOG	CSCO
AAPL	1	0.5381241	0.5529475	0.4127518
MSFT	0.5381241	1	0.7270103	0.5707302
GOOG	0.5529475	0.7270103	1	0.525555
CSCO	0.4127518	0.5707302	0.525555	1

Table 4 Correlation matrix among return for Big Tech portfolio

3.2 Correlation between T Bill and Portfolios

As it often the case, the stock market would go in opposite direction of bond market and the extend would depend on the correlation between them (Duff, n.d.). Thus, the following figure shows the performance of the 10 Year T bill in the period.

Figure 1. Performance of 10 Year T Bill

Figure 1 generally shows a positive tendency of 10 Year T Bill in this period and the correlation between it and the portfolios can be view from the figures in below.

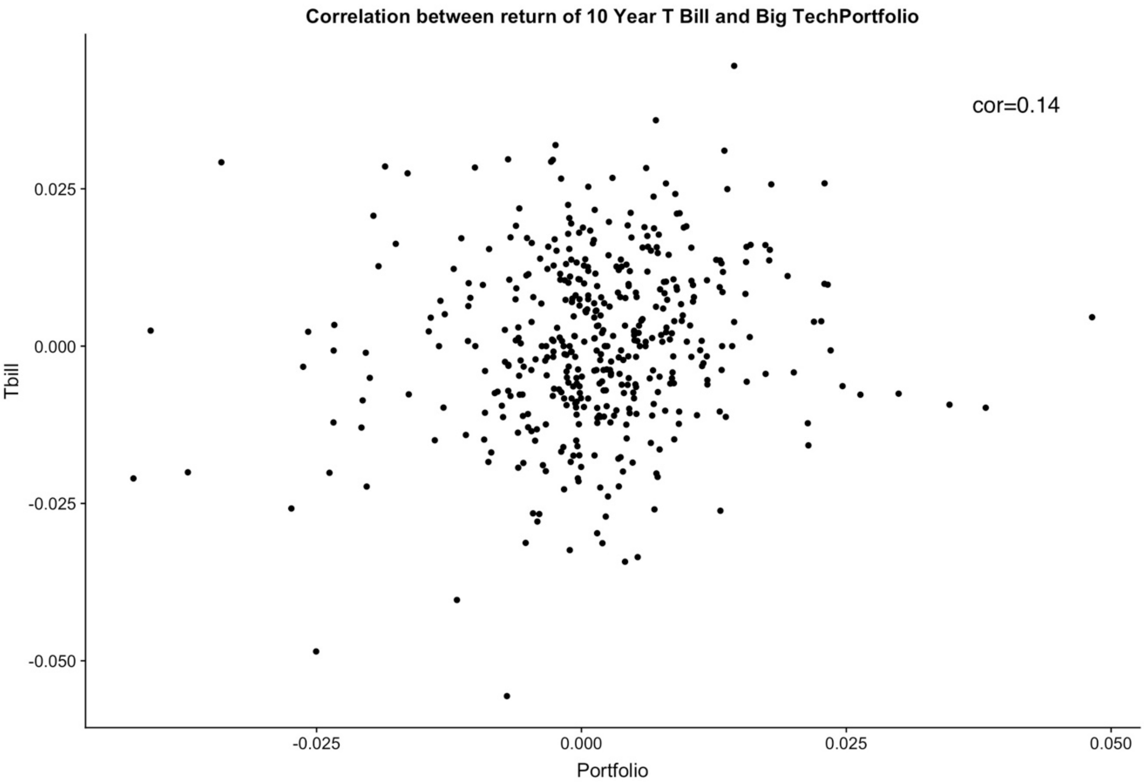


Figure 2. Correlation between 10 Year T Bill and Big Tech Portfolio

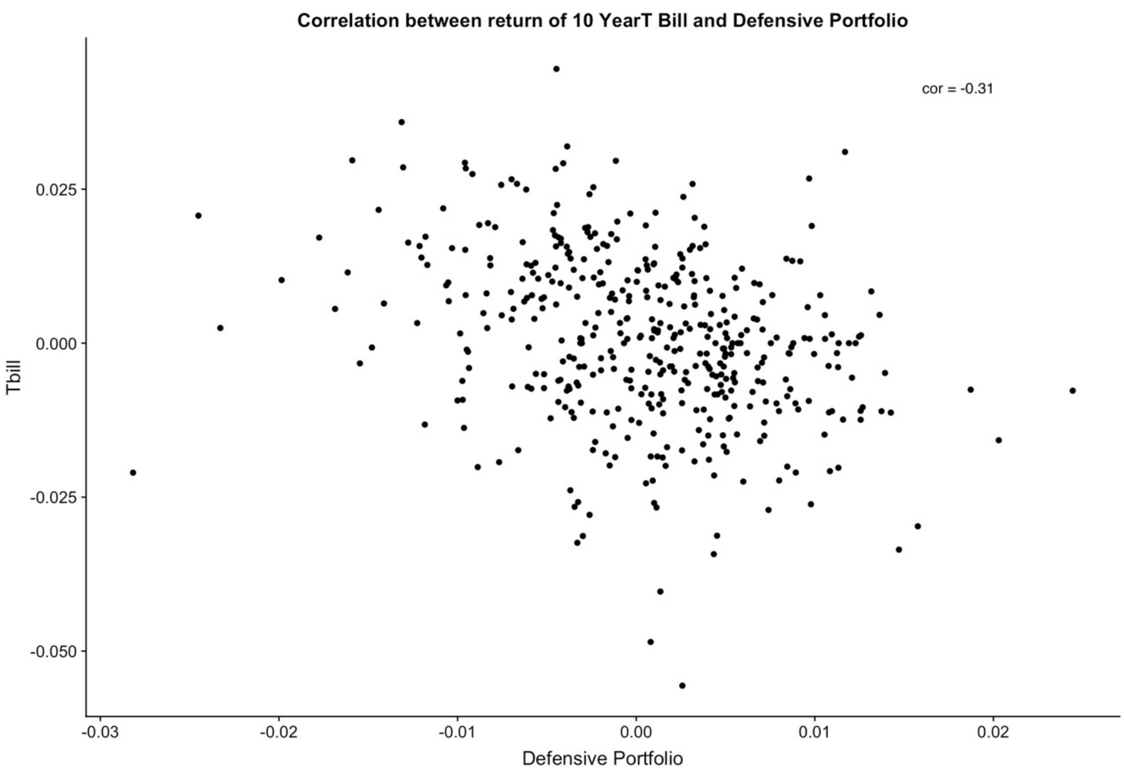


Figure 3. Correlation between 10 Year T Bill and Defensive Portfolio

By comparison, the defensive portfolio contains a higher absolute value of correlation which means it has stronger correlation with 10 Year T Bill than Big Tech Portfolio. In addition, since it is negative correlated with the T Bill, it is assumed that it will go to the opposite direction of the T Bill and its performance will be explained in the following sections. With respect to Big Tech portfolio, it has very small positive correlation with T Bill so it is assumed is basically is not correlated with T Bills to a certain degree.

3.3 Normal copula and t-copula

After fitting marginal distributions, we model the dependence structure using normal and t-copula. Dependence structure is the correlation matrix $\sum$ (a positive definite matrix) among stocks' log returns.

At the stage of fitting normal copula, L satisfies the equitation $LL^T = \sum$, which means that L denotes the (lower-triangular) Cholesky factor of $\sum$. This correlation matrix L will be used for simulating financial returns under copula. After calculating L, we can generate the source of the randomness with independent standard normal variates vector is $Z = (Z_1, Z_2, \dots \dots Z_d)$, which denote the independent stock returns. It can be transformed into the correlated normal vector $\tilde{Z} = LZ$.

Under normal copula, the return of the portfolios is:

$$R(\mathbf{Z}) = \sum_{j=1}^{d} w_j e^{G_j^{-1}(\Phi(\tilde{Z}_j))} \quad \text{with} \quad \tilde{\mathbf{Z}} = \mathbf{L}\,\mathbf{Z}$$

Where Φ is the CDF of standard normal distribution; G_j is the CDF of the marginal distribution of the return of the jth stock (G^{-1} is inverse CDF or quantile); $G^{-1}(\Phi(\tilde{Z}))$,j=1,⋯,d gives log returns under normal copula dependence structure; Thus, $e^{G^{-1}(\Phi(\tilde{Z}j))}$, j = 1,⋯ ,d gives us returns.

Finally, by multiplying returns of stocks with weights in the portfolio, we get return of the portfolio.

For t-copula, we use vector T rather than $\tilde{Z}$ and volatility scaling factor c_i should be taken in to account. T is derived from $T = \tilde{Z}/\sqrt{Y/v}$, the multivariate t-distribution by generating a random variate Y from a chi- squared distribution with v degrees of freedom (χv^2). c_i is a scaling factor concerning the daily volatility σ_i and the variance var_i of the *i* th marginal distribution by the formula:

The portfolio return under this circumstance can be denoted as:

$$R(V) = \sum_{i=1}^{d} w_i \, e^{c_i G_i^{-1}(F(\mathrm{T}))}$$

3.4 VaR

As Kresta (2013) claimed, there were 3 mainstream approaches to conduct VaR estimation, namely, variance-covariance approach, historical simulation and Monte Carlo simulation. In our project, we adopt Monte Carlo simulation to simulate portfolio VaR. Algorithm 1 explicitly demonstrates the rational and process concerning how we conduct it.

Algorithm 1. Computation of VaR employing Monte Carlo Simulation

```
Define the period of daily VaR that need to be calculated, from "dateFrom" to "dateTo";
Define the period of source data used to derive one specified VaR. Initially, "endDate" = "dateFrom" and "maxDate" = a year before "endDate";
for ("dateFrom" to "dateTo") {
Get the companies' adjusted prices from Yahoo over the period from "maxDate" to "endDate";
Calculate the log daily returns of 8 companies;
Fit the log daily returns with t-distribution and get their means(μ) and standard deviations(σ);
do {
    for (1 to 10000) {
    Compute Cholesky factor L of Σ for each portfolio, i.e., LL'=Σ;
    Generate independent standard normal variates Z, then compute ;
      Calculate each company's return R = ;
    Calculate Portfolios' return R_port = w1*R1+ w2*R2+ w3*R3+ w4*R4;
    }
  Count the times (n) of the case that R_port is less than VaR;
  Find VaR where n is equal to 10000 * alpha (confidence level) by using least square method;
  } while (n is equal to 10000 * alpha (confidence level))
"maxDate" = The day after "maxDate";
"endDate" = The day after "endDate";
}
```

4. Marginal distribution and copula fitting to financial data

(Pinyi.Wu 30%, Shiqi.Dong 30%, Xiaowen.Man 30%, Yifan.Yang 10%)

In this section, we attempt to use normal and t-distribution for marginal distribution and determine the optimal model according to the maximize likelihood (MLE) and Akaike Information Criterion (AIC) methods, then we use normal and t-copula to fit the optimal model (t or normal distribution) respectively. The t-distribution is a simple extension of the Gaussian distribution, according to our result, t-distribution shows higher fitting goodness.

4.1 Fitting marginal distribution

Inference functions for margins methods are employed to fit a dependence structure between log returns of stocks. Based on daily log returns of stock *i* during *T* days, the log-likelihood maximization problem for fitting marginal distribution is:

$$max \sum_{t=1}^{T} ln(f_j(x_j^t;\beta_j)), j = 1, \ldots, d, (4.1)$$

Where f_j is the probability density function and β_j is the parameter vector for candidate distribution.

The returns financial products rarely show a normal shape (Ang and Chen, 2002). Before we fitted these 2 distributions, we had tested the normality of the datasets of stock log returns to figure out whether log returns follow normal distribution, using Shapiro–Francia (SF), Anderson–Darling (AD), Cramer–von Mises (CVM), Lilliefors, and Pearson chi-squared tests belonging to the "nortest" package in R.
Given a significance level of 0.05, the null (H_0) stating that the data comes from the normal distribution can be rejected in most tests. As Table 3 and Table 4 shows, log returns do not follow a normal distribution.

Stock	**AD test**	**SF test**	**CVM test**	**Lilliefors test**	**Pearson test**
AAPL	1.832e-13	2.538e-10	3.382e-09	1.89e-08	3.331e-06
MSFT	6.889e-14	1.859e-12	2.452e09	9.369e-08	1.902e-04
GOOG	1.312e-13	4.082e-10	3.854e-09	1.059e-09	2.652e-06
CSCO	< 2.2 e-16	6.653e-14	7.37e-10	6.563e-11	2.136e-07

Table 5 p-values for five normality tests for stock log returns in Big Tech portfolio

Stock	**AD test**	**SF test**	**CVM test**	**Lilliefors test**	**Pearson test**
PRT	5.135e-04	6.885e-08	0.0006375	0.002237	0.06205
NEE	0.002096	6.564e-04	0.006643	0.0567	0.3817
AEP	0.00118	7.819e-04	0.001929	0.03563	0.007487
JNJ	4.479e-12	3.317e-11	1.076e-08	2.273e-10	3.731e-06

Table 6 p-values for five normality tests for stock log returns in defensive portfolio

After deriving the above conclusion (non-normality of the log returns), we fit the t-distribution firstly but also fit the normal distribution for further verification.

At fitting stage, the package 'fitdistrplus' is used to fit the two distributions.

When determine the optimal marginal distribution, the magnitude of log-likelihood values will be considered as the best criteria. Nevertheless, the estimated number of parameters are excluded from log-likelihood values. AIC value, a comprehensive indicator, will also be referred to for a judicious decision in this process. AIC value can be derived in this equitation:

$$AIC = -2 \times LogLik + 2 \times NE$$

LogLik represents the log-likelihood value and NE denotes estimated number of parameters.

For each log return dataset, we fit them into the two marginal distributions respectively, and the distribution with greater log-likelihood and lower AIC will be preferred and selected. According to Table 5 and Table 6, t-distribution is the best-suited for all log return datasets, so this candidate will be picked out and chosen as the marginal distribution.

Apart from the statistical evidence of the 5 tests belonging to 'nortest' package, Table 5 and 6 also reinforce that t-distribution fits these data better, and normal distribution also do not show high goodness of fitting.

	GAUSSIAN		**T-DISTRIBUTION**	
STOCKS	Log-likelihood	AIC	Log-likelihood	AIC
AAPL	1293.009	-2582.018	1328.183	-2650.367
MSFT	1315.659	-2627.317	1359.811	-2713.622
GOOG	1295.032	-2586.063	1325.636	-2645.273
CSCO	1312.224	-2620.447	1371.151	-2736.303

Table 7 Results of Fitting different distributions for Big Tech portfolio (Appendix 1)

	GAUSSIAN		T-DISTRIBUTION	
STOCKS	Log-likelihood	AIC	Log-likelihood	AIC
PRT	1306.203	-2608.406	1318.922	-2631.432
NEE	1462.348	-2920.696	1470.149	-2934.297
AEP	1464.501	-2925.002	1468.863	-2931.726
JNJ	1410.416	-2816.833	1446.424	-2886.849

Table 8 Results of Fitting different distributions for defensive portfolio (Appendix 2)

4.2 Copula fitting

Based on the above process, we conclude that t-distribution is proper for our data concerning stocks' log returns, so we can fit normal copula and t-copula to the log returns of stocks and compare the result.

The log likelihood and AIC will also be referred to when determine which copula is better. The log likelihood function of the copula is:

$$max \sum_{t=1}^{T} lnc(F_j(x_1^t;\beta_1),\ldots,F_d(x_d^t;\beta_d);\alpha), \quad (4.2)$$

Where F_j is the cumulative distribution function (CDF) of marginal distribution for stock $j = 1, \ldots, d, c$ is the density of copula function and α is the parameter vector for the copula. In this part, we only select normal and t-copula and utilized 'copula' package. The criteria of selecting the superior one is the same as the last step: the model who has bigger log-likelihood and lower AIC will be selected. The numerical results concerning fitting performances of the 2 models are displayed below:

	log-likelihood
Normal copula	240.4
T-copula	254.9

Results of copula-fitting for defensive portfolio

	log-likelihood
Normal copula	347.9
T-copula	382

Results of copula-fitting for aggressive portfolio

As Demarta and McNeil (2005) indicated, t-copula is perceived as a superior model when modeling multivariate data especially in financial return context; and based on the empirical analysis of Mashal et al. (2003) and Breymann et al. (2003), t-copula often outperformed Gaussian copula. Our result above presents the same conclusion: t-copula shows better fitting goodness, hence it is selected.

	AAPL	MSFT	GOOG	CSCO
AAPL	1	0.5648	0.5675	0.4818
MSFT	0.5648	1	0.6960	0.5850
GOOG	0.5675	0.6960	1	0.5261
CSCO	0.4818	0.5850	0.5261	1

Table 9 Correlation matrix of the fitted t-copula for stock log returns for Big Tech portfolio

	PRT	NEE	AEP	JNJ
PRT	1	0.3534	0.4349	0.2182
NEE	0.3534	1	0.7625	0.2605
AEP	0.4349	0.7625	1	0.1983
JNJ	0.2182	0.2605	0.1983	1

Table 10 Correlation matrix of the fitted t-copula for stock log returns for defensive portfolio

5.Measure market risk by Monte Carlo simulation and calculate VaR *(Pinyi.Wu 30%, Shiqi.Dong 30%, Xiaowen.Man 30%, Yifan.Yang 10%)*

We calculated VaR spanning from 2017.01.01 to 2018.10.01, which can be detailed interpreted as the following:

The period we concentrate on is 2017-01-01 ~ 2018-10-01 (only the trading days are considered), to simulate VaR, for the date (Y-M-D), we use the data of (Y-1, M, D-1)~(Y, M, D-1) into Monte Carlo simulation . More specifically, when simulate the VaR for date 2018-9-29, the log return data spanning 2017-9-28 to 2018-9-28) is used for Monte Carlo simulation, in other words, the most recent 252 available data at each currently focused time point. Then the 439 simulated VaR can be derived.

We set alpha as 0.05 and Monte Carlo simulation is applied to assess the risk of portfolios.

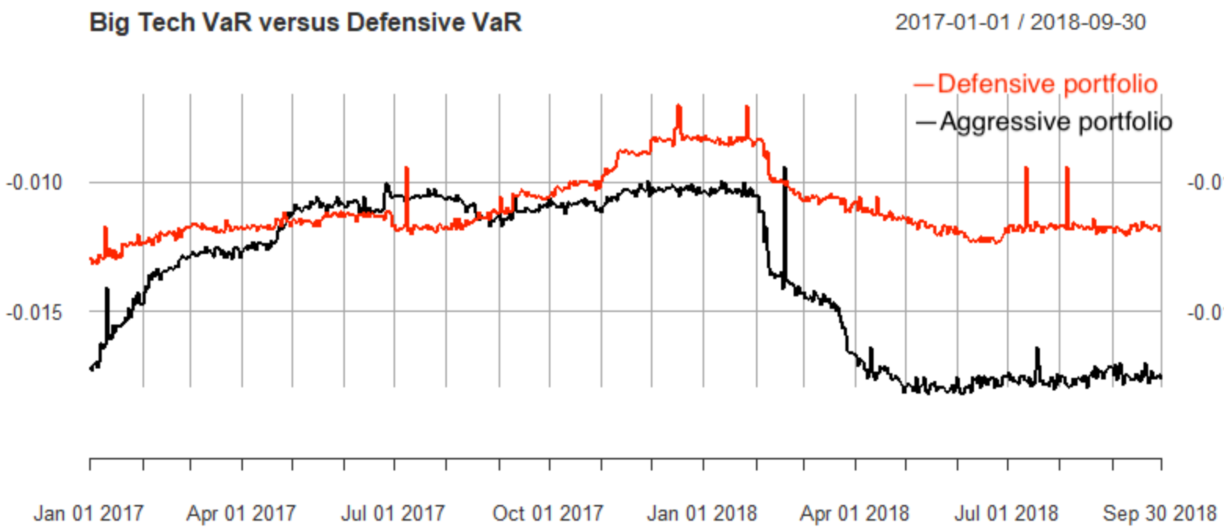


Figure 3. VaR Comparison of VaR between Big Tech and defensive portfolios

6. Backtesting for VaR *(Pinyi.Wu 30%, Shiqi.Dong 30%, Xiaowen.Man 30%, Yifan.Yang 10%)*

The simulated VaR has been derived in section 5, the purpose of this section is to compare the actual profit or loss with the simulated values and evaluate the results.

6.1 Significance of backtesting for VaR

After a series preparation above, the VaR based on Monte Carlo Simulation can be derived finally. But it is not the end of the story, and backtesting is indispensable to verify the accuracy of VaR model (Terzić and Milojević , 2016) and provide a reality check on the calculations (Abbott, 2013), as the VaR we got is based on the historical data and statistical simulation, so it may differs from the realistic situation. In financial domain, 'backtesting' is used in diverse manners but usually, it refers to evaluating the model by utilizing historical data and compare the results with the realized rates of return (Christoffersen, 2009), alternatively, the statistical process where we make comparisons between the actual losses and the forecasts is known as backtesting.

The inaccuracy of the VaR encompasses two sources: firstly, there exists outliers and violations (the realized loss which significantly exceeds the simulated VaR); the other one is the time period when the outliers tend to be in a group (1). Checking the validations is vital in order to verify the reliability of VaR model, alternatively, backtesting actual losses are in accordance with the projected ones.

6.2 Calculating the realized and actual Profit & Loss

To obtain the actual Profit & Loss, we calculate daily return or loss of the portfolio using the following equitation:

$$R_j = w_i * ret_{i,\ j}$$

w_i represents the weight of stock *i* (we assign 0.25 as the weight of each stock), and $ret_{i,\ j}$ denotes the return of stock *i* at time *j*. By aggregating individual stocks' returns at time *j*, we can get the portfolio return at time *j*.

6.3 Comparing the losses in actual situation with the simulated VaR and Possible explanation of violations

As Christoffersen (2009) defined, "hit sequence", which is indicator function of VaR exceptions should be considered to evaluate the performance of VaR model:

$$I_t \begin{cases} 1 \; if \; l_t > VaR_t^p \\ 0 \; if \; l_t \leq VaR_t^p \end{cases}$$

where I(·) is the indicator function, l_t is the real loss observed at time t+1, VaR_t^p is the simulated VaR with probability level p at time t , and when the portfolio has positive returns, we classify these in to value 0. Then we can count the number of violations which breaks through the line of simulated VaR. For big tech and defensive portfolios, there are 29 and 14 violations respectively with the 95% probability level (438 trading days in total).

For visualizing purpose, we draw the line of simulated VaR calculated before, and the actual Profit & Loss also are present in the same graph to easily observe the violations or the exceedance.

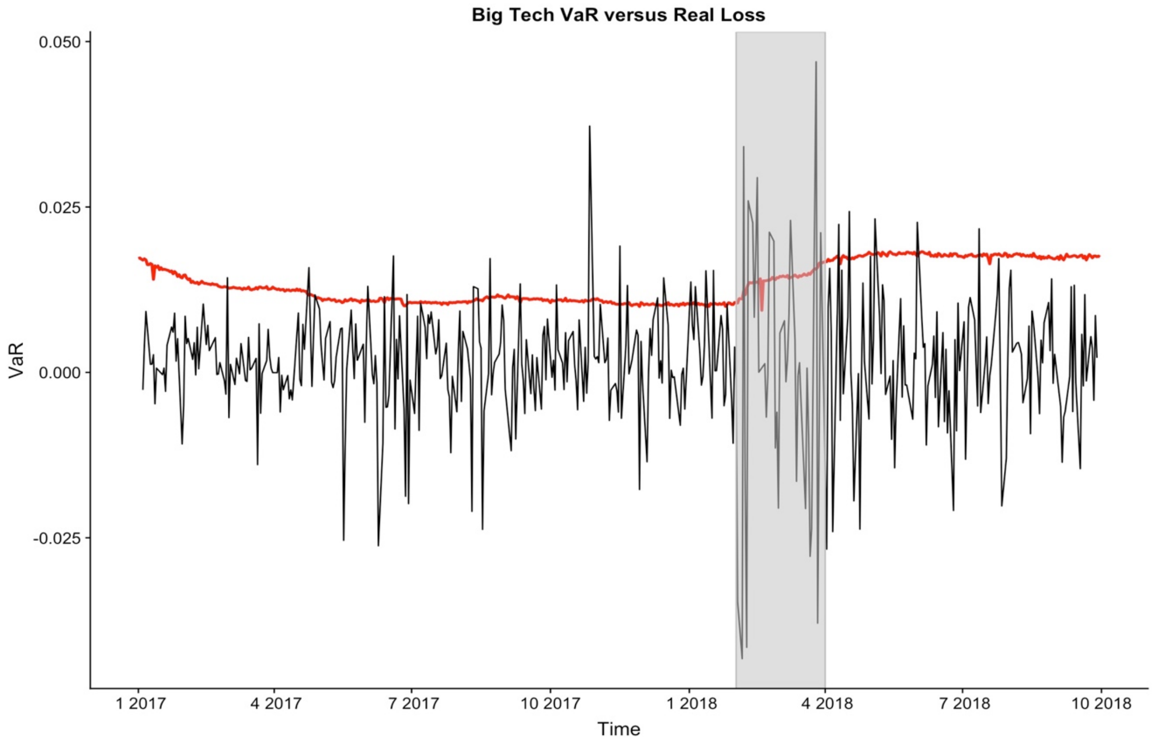


Figure 4. Comparison between the actual profit & loss and simulated VaR of big tech portfolio

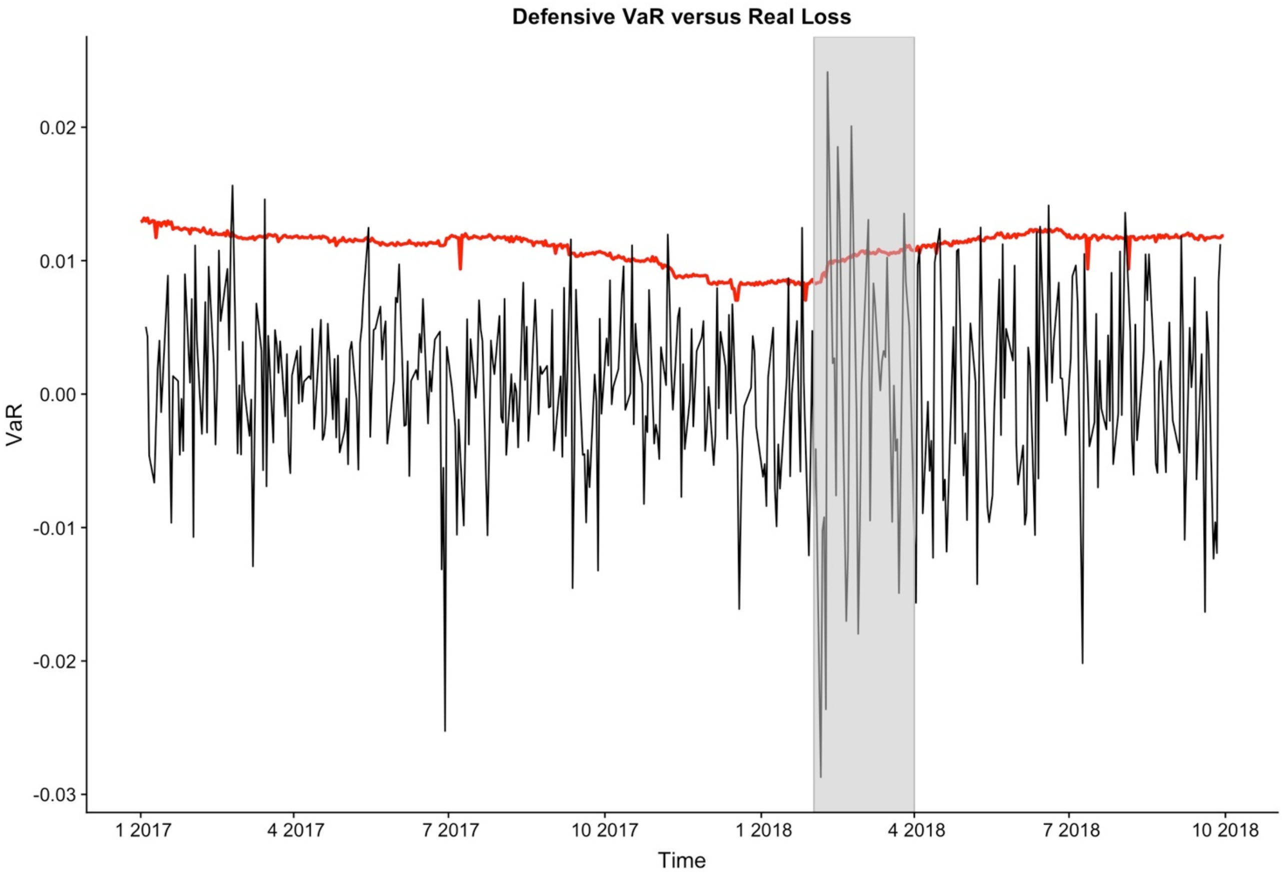


Figure 5. Comparison between the actual profit & loss and simulated VaR of defensive portfolio

During the February and March in 2018, it is manifest to see the returns for both portfolios show high fluctuation during these 2 months, and the actual loss tend to hit and break through the line of simulated VaR (the red line). The reasons for the poor predicting performance of VaR during this period can be intricate, we attempt to capture some events which may exert potential effects on the portfolio return, and the following news are cited from Bloomberg (2018):

In February, 1) Microsoft signed an agreement with Sunseap group and would create the solar energy portfolio in Singapore; Apple was going to hold a press conference for its new product;

During March, 1) Apple faced the third trial in running patent battle with VirnetX , as VirnetX claime that Apple infringed their patents; 2)It was reported that Google may add sales on cloud, artificial intelligence (like open-source machine learning framework) advances , these action would boost revenue.

At the end of February, Trump was said to likely impose stiff steel, aluminum tariffs, which contributed to profound impacts and shock news among various industries, including the non-cyclical and defensive sectors, and NextEra Energy (NEE) and American Electric Power (AEP) are categorized in these industries.

Besides the typical news and vital events mentioned above, many other pieces of information and events that are not mentioned also have influence on the portfolio performance, and this is ultimately reflexed in the actual profit and loss.

The violations are likely to imply that the statistical approach (Monte Carlo simulation) estimating VaR may not efficiently respond to the shocking news sometimes; various factors will impose influence on the portfolio performance.

By dating back the historical events in the U.S. stock market during that time, various events and other factors may lead to high volatile return or loss, which cannot be fully considered in our simulating method, so there exists several violations. But based on the graph and the number of violations, generally, estimating VaR by Monte Carlo simulation offers high accuracy, as the actual losses are below the simulated VaR in the vast majority of cases.

7. Portfolio comparison *(Pinyi.Wu 30%, Shiqi.Dong 30%, Xiaowen.Man 30%, Yifan.Yang 10%)*

There are various methodologies to compare the performance and the risks between portfolios. In this report, the volatility, cumulative return and drawdowns will be selected to analyze the performance among the big tech portfolio, defensive portfolio and the benchmark which is SP500.

Volatility

Garch(p,q) model is applied to calculate the volatility which is defined below and Garch(1,1) (Fryzlewicz, n.d.).

$$y_k = \sigma_k \varepsilon_k$$

$$\sigma_k^2 = \omega_t \sum_{i=1}^{p} \alpha_i y_{k-i}^2 + \sum_{j=1}^{q} \beta_j \sigma_{k-j}^2$$

In the code, the distribution.mode was set to be “std” which is t-distribution rather than normal distribution as the stock price presented above usually have long tails. The daily volatilities are given below.

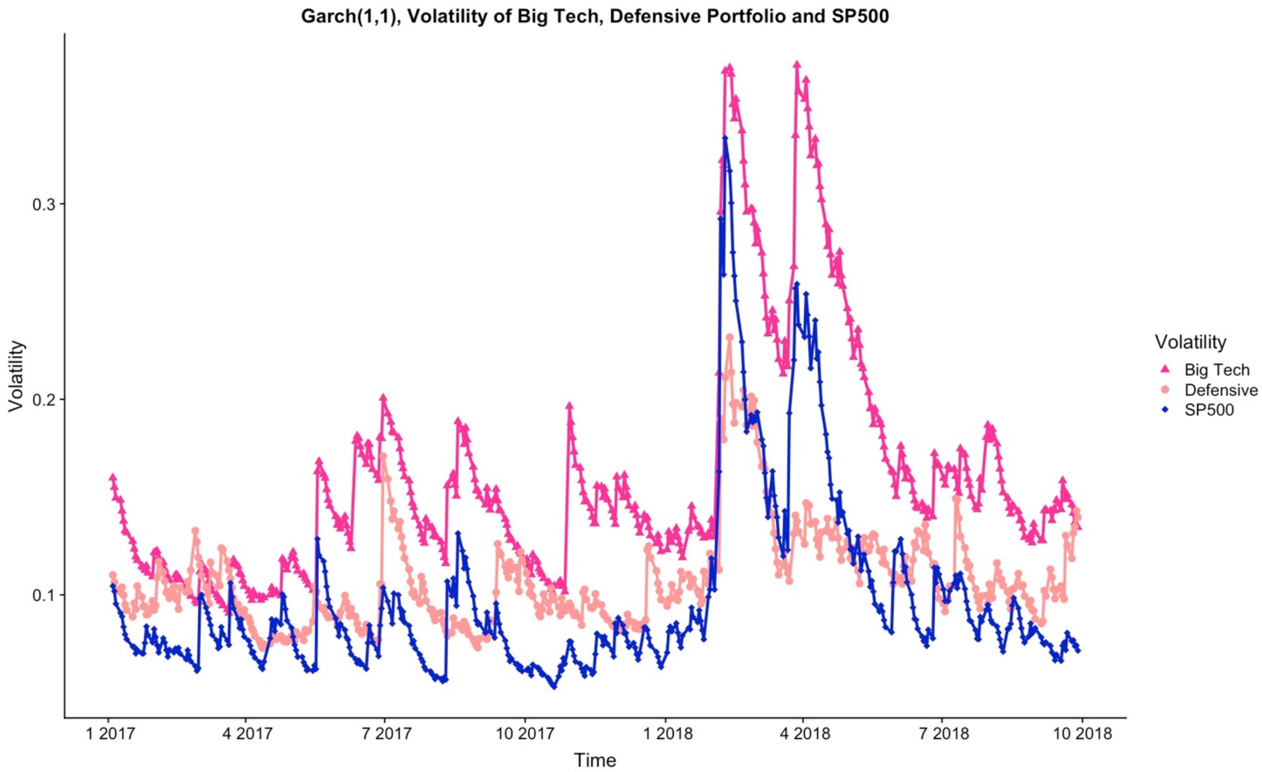


Figure 6. Volatility of Big Tech, defensive and S&P 500

During this period, all of the three experienced a high volatility in February, 2018. As regard to Big Tech portfolio, it shows a sudden increase and a following drop during February, 2018 and April, 2018. As a consequence, according to the definition of volatility, it refers to the fluctuation of a stock price (Chen, 2018b). A higher volatility indicates the price could change significantly in either direction. Together with the Beta coefficient computed before, the volatility of Big Tech portfolio which contains higher value of Beta basically move above the benchmark and conversely, the defensive portfolio moves below the benchmark. Normally, volatility is associated with risk. However, low volatility may be a better indicator of a lower return rather than a lower risk which will be explained in the following sectors.

Recall the section 6 explicitly shows cluster violations of VaR of two months spanning from 2018.02 to 2018.04 in both portfolios, we can also find the similar abnormal trend in volatility: as the figure 6 demonstrates, the volatility graph also illustrates the portfolios have high volatility during 2018.02-2018.04. By taking various of factors like the tax tariff and other shocking news happened that time, these information is absorbed and reflexed in the actual profit and loss. Thus, the VaR violations and some extreme profit and loss during 2018.02-2018.04 can be interpreted. Additionally, the high volatility can match these VaR violations and extreme profit and loss during 2018.02-2018.04.

Cumulative Return

Cumulative return can be defined as the total gain of loss that an investment would generate which is irrespective of time period (Chen, 2018c). Since the data from Yahoo Finance does not contain dividends or corporate actions of companies, the calculation would be much easier and the raw mathematical return will be applied. The equation below explains the calculations.

$$\frac{\text{Current Price of Security} - \text{Original Price of Security}}{\text{Original Price of Security}}$$

Besides the comparison between the big tech and defensive portfolios, the combination of the benchmark return will explicitly show to what extent does the portfolio perform against the benchmark. For example, if a company consistent underperforming the benchmark, it could be the signal that it is not promising and may be experiencing troubles (Martin, 2018).

In this case, the function which is “Charts.CumReturns()” in the package “PerformanceAnalytics” will be applied first to plot the cumulative return and the “ylog” in it is set to FALSE since the normal return is expected. The figure is shown below.

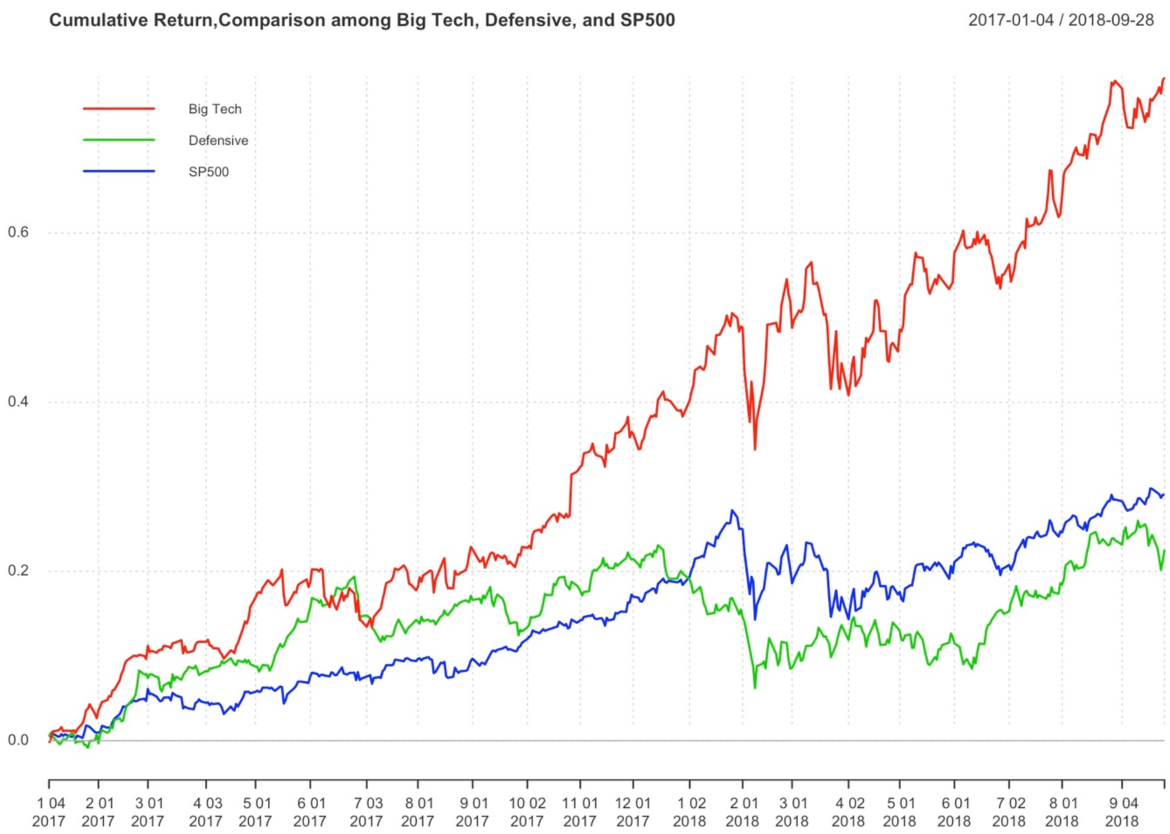


Figure 7. Cumulative Return of portfolios

As shown in the Figure 7, red, green and blue stand for Big Tech portfolio, Defensive portfolio and S&P500 respectively. It is clear that the Big Tech portfolio significantly outperform the S&P500 and the Defensive Portfolio with over 60% cumulative return. In which case, the investor could get over $16000 back with initial investment of $10000. With respect to Defensive portfolio, it generally has similar performance with the benchmark so the Big Tech portfolio has a better performance. To have an overall view of the figure, there is a significant plummet during February, 2018 which affect the cumulative return of them. Thus, referring to the real events during that period, it is discovered that stock market plunged on Feb 5th, 2018. The Dow Jones industrial average dropped more than 1500 points and erased the gains in 2018 (Cox, 2018). In addition, S&P5000 declined 4.1% which results in the biggest one-day drop since August 2011 (Imbert, 2018b). The highlighted area in Figure 8 provides corresponding information.

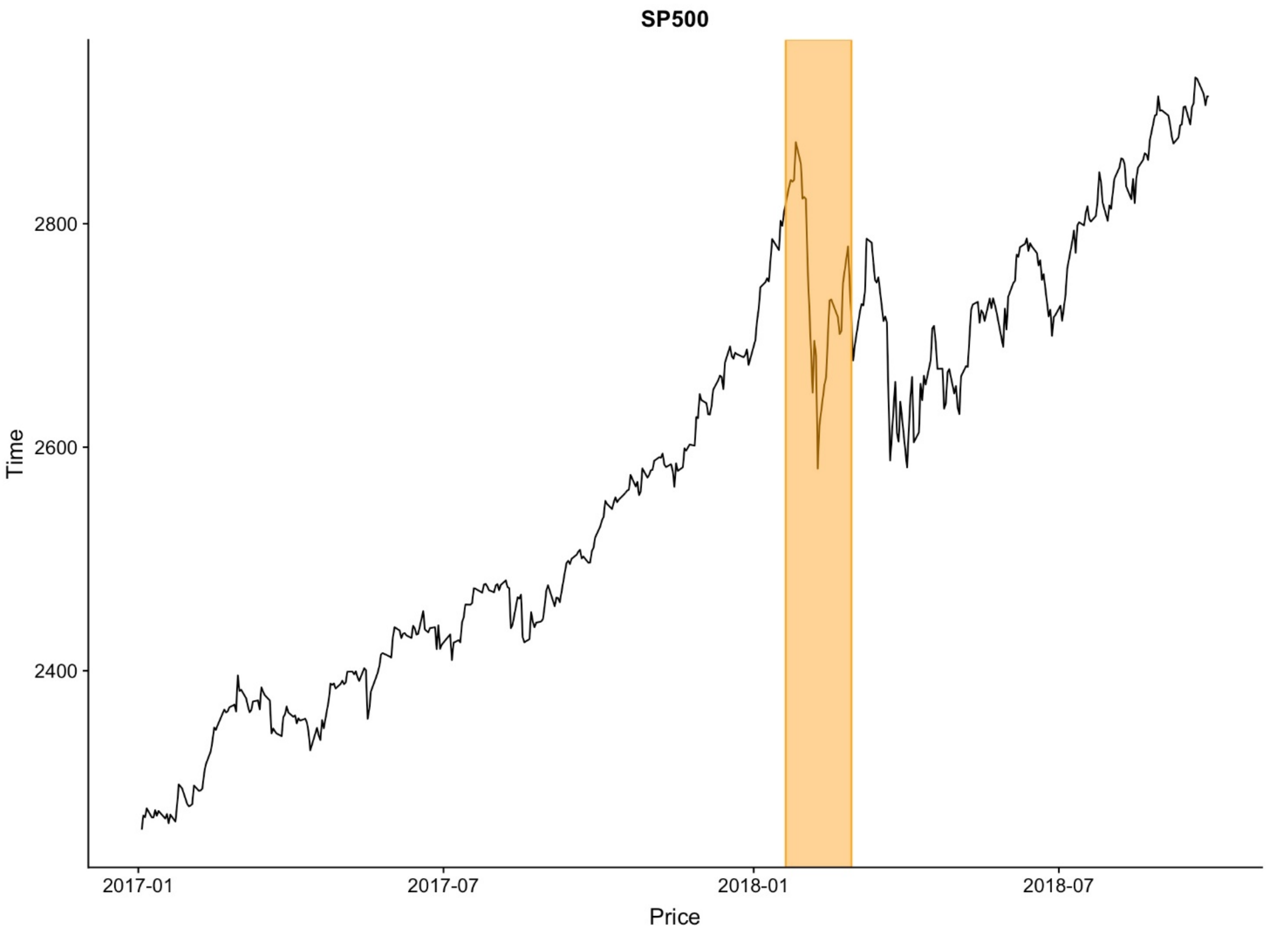


Figure 8. S&P500

The main reason behind this event would be the inflation fears and thus led to the selling phenomenon which is simultaneously reflected in the drawdowns.

However, after the huge plummet, there was a V-shaped recovery of Big Tech Portfolio followed on. A V-shaped recovery will be caused by the increased consumer demand spending in economy activity (Kenton, 2018a). Real events were discovered such as Apple Music is about to overtake Spotify as it has 5% growth of subscribers per month compared to 2% of Spotify (Steele, 2018).

Drawdown

Drawdown is defined as the peak-to-through reduction during a particular time period (Chen, 2018d). It is a vital measurement to measure the financial risk. The worst case which is named maximum drawdown, is an indicator of the maximum loss from a peak to a trough (Kenton, 2018b). It focuses on the capital preservation and most investor would bear 20% drawdown. To mitigate such risk, diversified portfolio is preferred since the market effect on different sectors would vary distinctly (Chen, 2018d). Therefore, defensive portfolio is a good option since the stocks in it cross sectors and it will provide protection against the drawdown. With real data, the drawdown risk is visualized in Figure 9 which is plotted by the self-modified function.

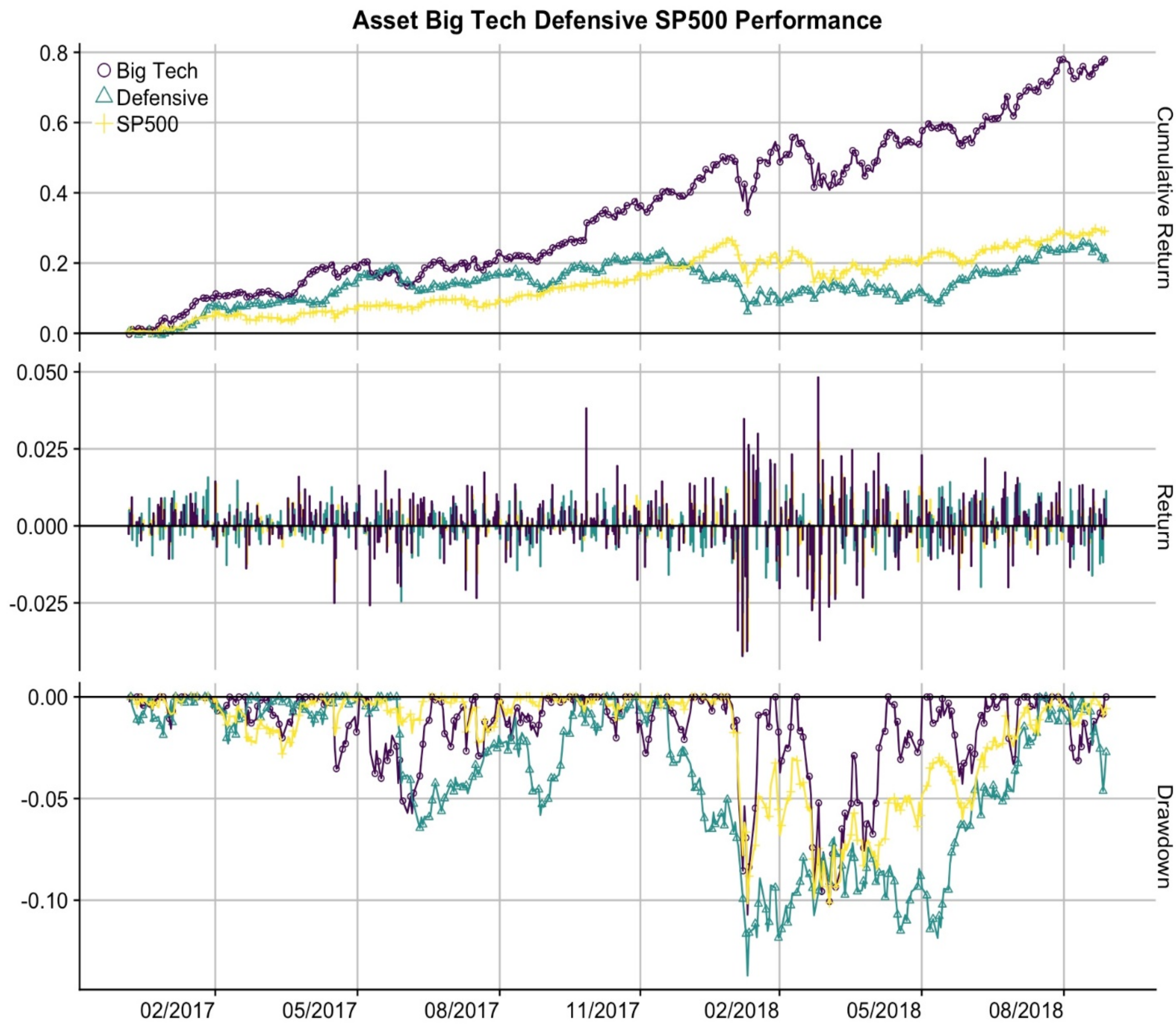


Figure 9. Drawdown

In the drawdown section of Figure 7, green, purple and yellow represent Big Tech Portfolio, Defensive Portfolio and the benchmark which is S&P500 separately. During Feb, 2018, the drawdown risk of defensive portfolio increased and exceeded -10% which was approaching -15%. By comparison, the big tech portfolio and the benchmark still held at around -10%. By analyzing the news during that period, the reasons why the defensive portfolio experienced a worse drawdown risk can be concluded as 1.) energy, financial service and healthcare sectors suffered considerable declining. 2.) only 17 stocks in the S&P500 ended up with a gain and the utilities sector which is considered as defensive also suffered losses, though it is less severe. 3.) since there is limitation on the numbers of stocks in a portfolio, the defensive portfolio in this project is not diversified enough as it only covers three sectors (Ferguson, et.al, 2018). Besides, referring to the Alpha computed above, all the stocks in the defensive portfolio contain the negative value of Alpha which means the defensive portfolio underperformed the benchmark in the period. As a consequence, it is unable to mitigate the drawdown risk in such situation and it is better to avoid or eliminate negative Alpha when investing (West, 2018).

8.Conclusion *(Pinyi.Wu 30%, Shiqi.Dong 30%, Xiaowen.Man 30%, Yifan.Yang 10%)*

This report quantifies the risk of a defensive and big tech portfolio by using copula-based approach and Monte Carlo simulation to calculate VaR. And the validation of simulated VaR, namely, backtesting for VaR, are also implemented. Although there are several violations, the calculated VaR can efficient simulate the real profit & loss of the time period in the majority of the cases. Nevertheless, it can be concluded that the pure statistical approach cannot fully cover the emergent events happening in the current financial market as there are still exceptions that cannot be captured or simulated by the simulation model. Reasonably, in order to efficiently simulate and quantify financial risks, realistic and real time information from the market and more complicated dependence structure or simulation model should be utilized.

By comparing the performance of portfolios, big tech portfolio shows higher profitability with high volatility. However, in contrast, the lower volatility of defensive portfolio does not indicate low risk directly and it may imply lower return. Thus, to measure the performance of the cumulative return, big tech portfolio shows great advantage when compared to the defensive one. Even during the negative economy period such as Feb, 2018, the big tech experienced a strong V-shaped recovery from recession. On the contrary, the benchmark and the defensive portfolio lack the ability to recover immediately. In addition, to measure the drawdown risk, the defensive portfolio which has lower volatility shows larger drawdown risk which is approaching -15%. The reason behind it could be the negative Alpha which results in its underperformance to the benchmark. As regard to big tech portfolio, it held around -10% and decreased after the Feb, 2018 and it is less risky than the defensive portfolio. As a consequence, the big tech portfolio in this case during this period from Jan. 1st,2017 to Oct. 1st, 2018 shows better performance and promising future.

Reference (prepared by using Harvard Style of Referencing, 5th edition, Anglia Ruskin University, January 2015.)

Abbott, K., 2013. Value At Risk (VAR) Models, *18.S096 Topics in Mathematics with Applications in Finance*. [online via internal VLE], Massachusetts Institute of Technology. Available at: <https://ocw.mit.edu/courses/mathematics/18-s096-topics-in-mathematics-with-applications-in-finance-fall-2013/lecture-notes/MIT18_S096F13_lecnote7.pdf> [Accessed Date 16 December 2018].

Allen, D. and Powell, R., 2007. Thoughts on VaR and CVaR. ECU Publications.

Amoroso, A., 2018. Focus: As data privacy concerns heighten, software stands to gain. Available at:< https://ideasandinsights.jpmorgan.com/content/2565/in-focus-as-data-privacy-concerns-heighten-software-stands-to-gain>[Accessed: 16 December 2018].

Ang, A. and Chen, J., 2002. Asymmetric correlations of equity portfolios. *The Review of Financial Studies,* 63, pp.443–494.

Artzner, P., Delbaen, F., Eber, J.M., and Heath, D., 1999. Coherent Measures of Risk. *Mathematical Finance*, 9, 203-228 [online]. Available at:< https://doi.org/10.1111/1467-9965.00068>[Accessed: 16 December 2018].

Boskovska, D. and Svrtinov, V., 2016. Risk of shares on the Macedonian capital market measured by the beta coefficient. *Economic Development,* 2(1), pp.79-90.

Chen, J., 2018a. Market Risk. Available at:< https://www.investopedia.com/terms/m/marketrisk.asp >[Accessed: 16 December 2018].

Chen, J., 2018c. Cumulative Return. Available at:< https://www.investopedia.com/terms/c/cumulativereturn.asp> [Accessed: 16 December 2018].

Chen, J., 2018d. Drawdown. Available at:< https://www.investopedia.com/terms/d/drawdown.asp> [Accessed: 16 December 2018].

Chen, J., 2018b. Volatility. Available at:< https://www.investopedia.com/terms/v/volatility.asp> [Accessed: 16 December 2018].

Christoffersen, P., 2009. Backtesting. *Encyclopedia of Quantitative Finance*, R. Cont (Ed), pp.355-389.

Christoffersen, P., 2012. Elements of Financial Risk Management. *Elsevier Science.*

Corporate Finance Institution, 2018. *What's alpha?.* [online] Available at: < https://corporatefinanceinstitute.com/resources/knowledge/finance/alpha/ > [Accessed 16 December 2018].

Cox, J., 2018. Stock market plunged Monday. Available at:< https://www.cnbc.com/2018/02/05/why-the-stock-market-plunged-today.html> [Accessed: 16 December 2018].

Demarta, S. and McNeil, A., 2005. The t Copula and Related Copulas. *International Statistical Review,* **73 (**1), pp.111–129.

Duff, V., n.d.. *Correlation of Treasuries With Stocks.* [online] Available at: < https://finance.zacks.com/correlation-treasuries-stocks-10871.html#> [Accessed 16 December 2018].

Ferguson, W.G., Katz, H.S. and Rosner, A., 2018. Stock Market Today: February 5, 2018. [Online] Available at:

<http://www.valueline.com/Markets/Daily_Updates/Stock_Market_Today__February_5,_2018.aspx#.XBYdFxMzYWo > [Accessed: 16 December 2018].

Frankel, M., 2018. Is Physicians Realty Trust a Buy?. Available at: <https://www.fool.com/investing/2018/08/15/is-physicians-realty-trust-a-buy.aspx>[Accessed: 16 December 2018].

Fryzlewicz, P., n.d. Lecture notes: Financial time series, ARCH and GARCH models 26.

Imbert, F., 2018a. Dow plunges 1,175 points in wild trading session, S&P 500 goes negative for 2018. Available at:< https://www.cnbc.com/2018/02/04/us-stocks-interest-rates-futures.html >[Accessed: 16 December 2018].

Imbert, F., 2018b. Dow dives nearly 300 points, S&P 500 dips into correction levels in another wild day on Wall Street. Available at: <https://www.cnbc.com/2018/10/26/stock-market-us-futures-show-drop-for-dow.html> [Accessed: 16 December 2018].

Intelligent Income, 2018. High Yield Dividend Stocks — December 2018. Available at: <https://www.simplysafedividends.com/intelligent-income/posts/3-high-yield-dividend-stocks-july-2018 >[Accessed: 16 December 2018].

J.P.Morgan, 2018. The importance of a late-cycle bond strategy. Available at:< https://ideasandinsights.jpmorgan.com/content/2648/the-importance-of-a-late-cycle-bond-strategy- >[Accessed: 16 December 2018].

Kenton, W., 2018a. V-Shaped Recovery. [Online] Available at: <https://www.investopedia.com/terms/v/v-shaped-recovery.asp> [Accessed: 16 December 2018].

Kenton, W., 2018b. Maximum Drawdown (MDD). Available at: < https://www.investopedia.com/terms/m/maximum-drawdown-mdd.asp>([Accessed: 16 December 2018].

Kim, T., 2018. Apple investors should thank Trump’s tax cuts for the stock’s record high. Available at:<https://www.cnbc.com/2018/08/01/apple-investors-should-thank-trumps-tax-cuts-for-the-stocks-record-h.html>[Accessed: 16 December 2018].

Kouwenberg, R., Mentink, A., Schouten, M. and Sonnenberg, R., 2009. Estimating Value at Risk of Institutional Protfolios with Alternative Asset Classes. In: Gregoriou, ed. 2009. *The VaR Modeling Handbook.* New York: McGraw-Hill. Ch.2.

Kresta, A., 2013. Backtesting the filtered historical simulation for the VaR estimation of the Prague stock market index. *Journal of Economics, Management and Business,* 23 (1), pp. 15–26.

La Monica, P.R., 2018. Dow falls 832 points in third-worst day by points ever. Available at:< https://edition.cnn.com/2018/10/10/investing/stock-market-today-techs-falling/index.html> [Accessed: 16 December 2018].

Martin, M.J., 2018. Cumulative Return vs. Annualized Return. Available at: <https://pocketsense.com/cumulative-return-vs-annualized-return-2374.html >([Accessed: 16 December 2018].

Maverick, J.B., 2018. What are the major categories of financial risk for a company?. Available at:< https://www.investopedia.com/ask/answers/062415/what-are-major-categories-financial-risk-company.asp >[Accessed: 16 December 2018].

Musvosvi, C., n.d. *Backtesting Value-at-Risk (VaR): The Basics.* [online] Available at: <https://www.investopedia.com/articles/professionals/081215/backtesting-valueatrisk-var-basics.asp> [Accessed 16 December 2018].

Nelsen, R., 2006. An Introduction to Copulas. 2 ed ed. New York: Springer Science Business Media.

Nickolas, S., 2018. What are the primary sources of market risk?. Available at: <https://www.investopedia.com/ask/answers/042415/what-are-primary-sources-market-risk.asp> [Accessed: 16 December 2018].

Pflug G.C., 2000. Some Remarks on the Value-at-Risk and the Conditional Value-at-Risk. In: Uryasev S.P. (eds) Probabilistic Constrained Optimization. Nonconvex Optimization and Its Applications, vol 49. Springer, Boston, MA [online]. Available at:< https://doi.org/10.1007/978-1-4757-3150-7_15> [Accessed: 16 December 2018].

Radović, M. and Vasiljević, A., 2012. The stability of the beta coefficient for the most liquid stocks in the capital market in Serbia in the period 2006-2011. *Economic Themes, 50(3),* pp. 423-441.

Rockaffar, R. and Uryasev, S., 1999. Optimization of Conditional Value-at-Risk [pdf]. Available at: <http://www.pacca.info/public/files/docs/public/finance/Active%20Risk%20Management/Uryasev%20Rockafellar-%20Optimization%20CVaR.pdf> [Accessed: 16 December 2018].

Sak, H and Haksöz, Ç., 2011. A copula-based simulation model for supply portfolio risk. T*he Journal of Operational Risk,* 6(3), pp.15-38.

Sergey, S., Gaia, S. and Konstantin, K., n.d. VaR vs CVaR in Risk Management and Optimization [pdf]. Available at:< http://www.ise.ufl.edu/uryasev/files/2011/11/VaR_vs_CVaR_CARISMA_conference_2010.pdf > [Accessed: 16 December 2018].

Steele, A., 2018. Apple Music on Track to Overtake Spotify in U.S. Subscribers. [Online] Available at:< https://www.wsj.com/articles/apple-music-on-track-to-overtake-spotify-in-u-s-subscribers-1517745720 > [Accessed: 16 December 2018].

Terzić, I. and Milojević, M., 2016. Risk model backtesting. *Scientific review article,* 62(1), pp.151-162.

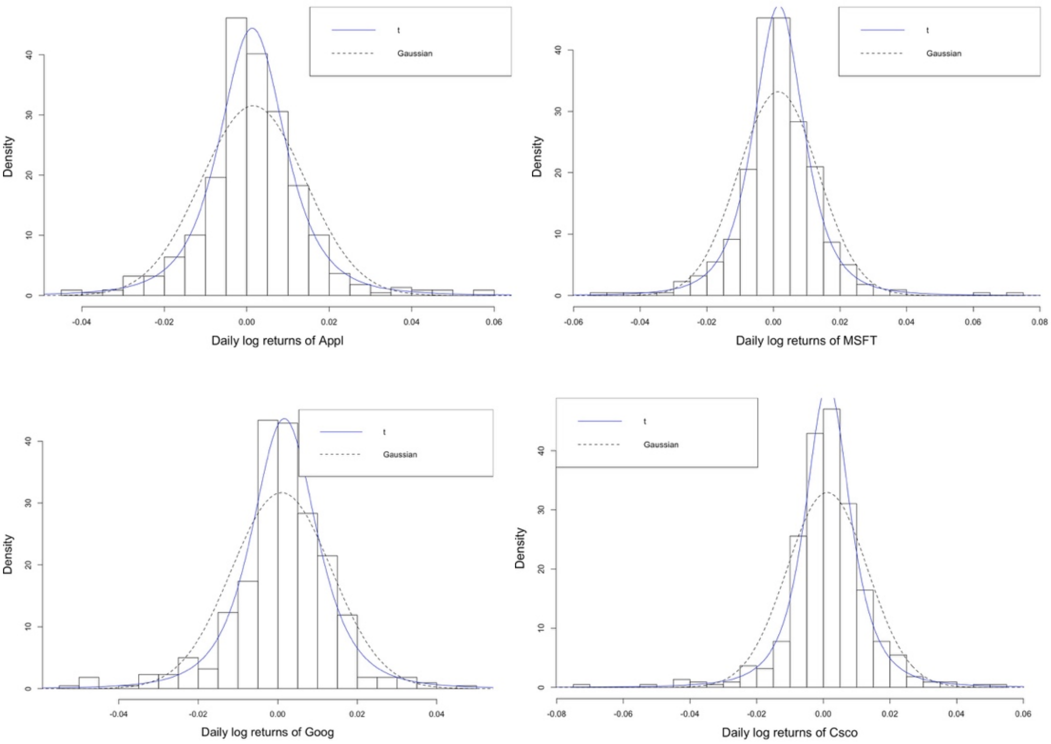
Vartan, V.G., 1984. Market Place; Stock Buyback Advantages. Available at:< https://www.nytimes.com/1984/05/15/business/market-place-stock-buyback-advantages.html> [Accessed: 16 December 2018].

Waters, R. and Johnson, A.E., 2018. Five US tech giants spend combined $115bn on buying back stock. Available at:< https://www.ft.com/content/2017ec30-e711-11e8-8a85-04b8afea6ea3> [Accessed: 16 December 2018].

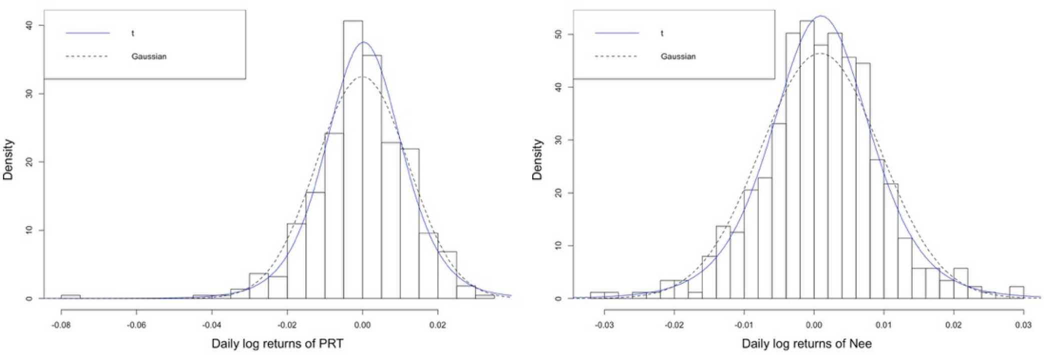
Wei, Y., Zhang, S., 2004. Dependence Analysis of Finance Markets: Copula-GARCH Model and Its Application. *Systems Engineering*, 22 (4) [online]. Available at: <http://en.cnki.com.cn/Article_en/CJFDTOTAL-GCXT200404002.htm> [Accessed: 16 December 2018].

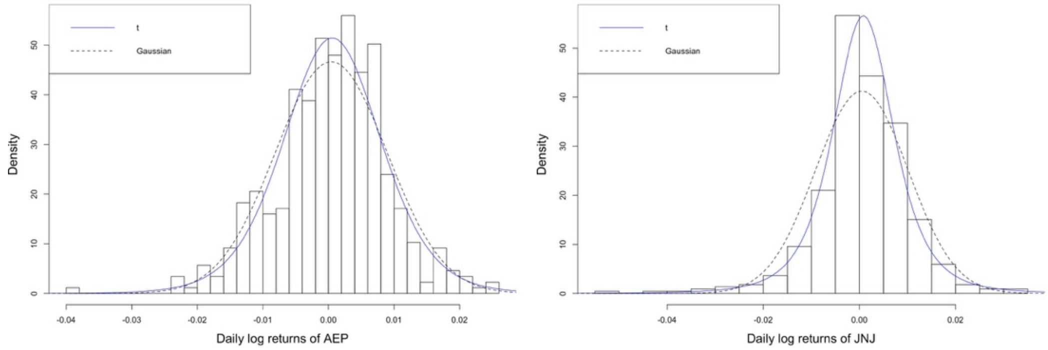
White, C.B., 2018. In what situations does it benefit a company to buy back outstanding shares?. Available at: <https://www.investopedia.com/ask/answers/040815/what-situations-does-it-benefit-company-buy-back-outstanding-shares.asp>[Accessed: 16 December 2018].

Appendix

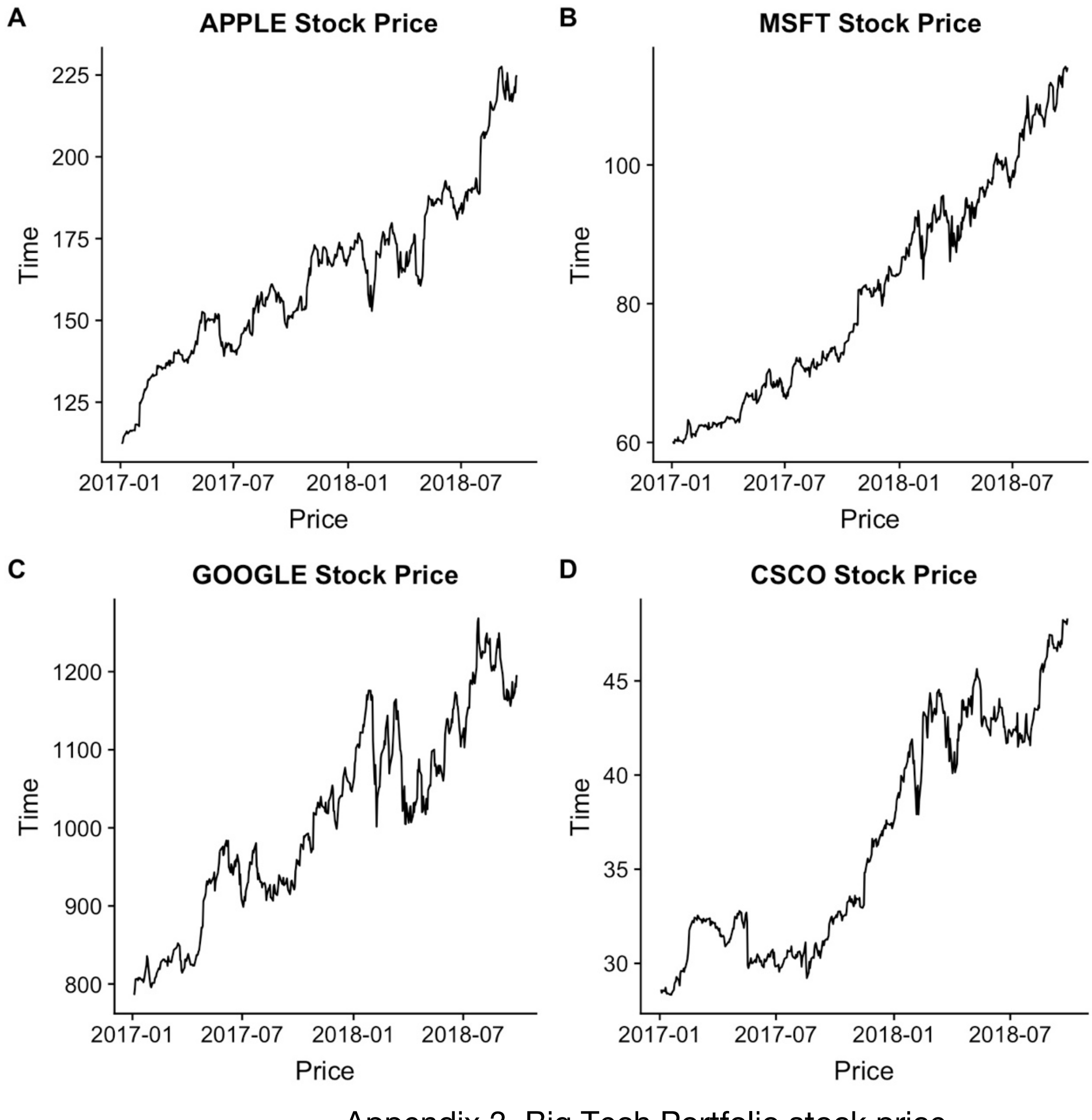


Appendix 1. Marginal Distribution of Stocks in Big Tech Portfolio

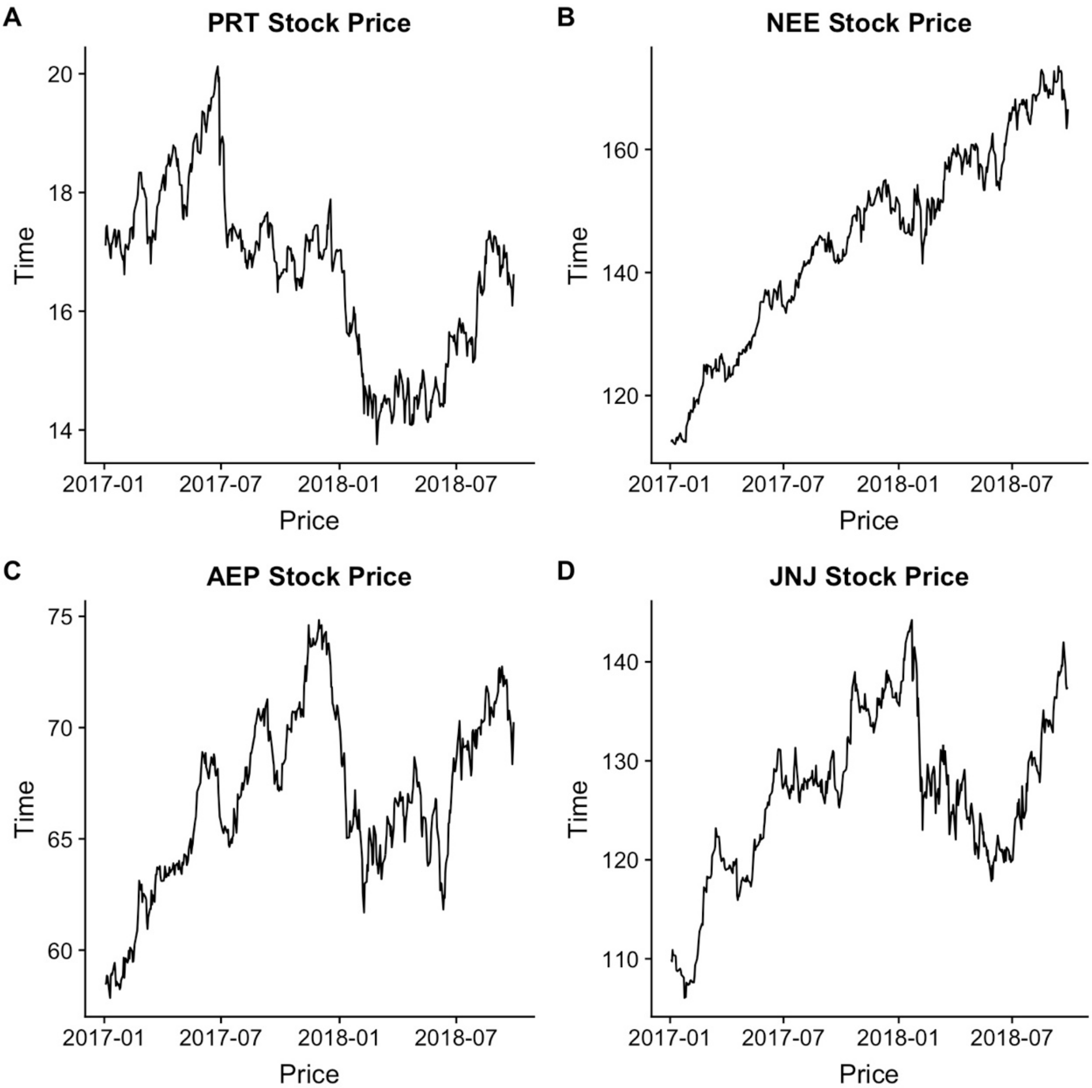




Appendix 2. Marginal Distribution of Stocks in Defensive Portfolio



Appendix 3. Big Tech Portfolio stock price



Appendix 4. Defensive Portfolio stock price

Appendix 5. R code *(Pinyi.Wu 30%, Shiqi.Dong 30%, Xiaowen.Man 30%, Yifan.Yang 10%)*

```
ibrary(quantmod)
library(PortfolioAnalytics)
library(PerformanceAnalytics)
library(xts)
library(rugarch)
library(rmgarch)
library(car)
library(fitdistrplus)
library(cowplot)
library(ggplot2)
library(ellipse)
library(corrplot)
library(scales)
library(expss)
library(copula)
maxDate <- as.POSIXct("2017-01-01", format = "%Y-%m-%d")
endDate <- as.POSIXct("2018-10-01", format = "%Y-%m-%d")
companies <- c("AAPL","MSFT","GOOGL","CSCO")
weight <- c(0.25,0.25,0.25,0.25)

#Get price and calculate the return
```

```r
Appl_adj <- Ad(getSymbols("AAPL", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
time_apple <- as.data.frame(Appl_adj)
time_apple <- cbind(date=as.Date(rownames(time_apple)),time_apple)
Appl_ret <- dailyReturn(Appl_adj)[-1]

MSFT_adj <- Ad(getSymbols("MSFT", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
time_msft <- as.data.frame(MSFT_adj)
time_msft <- cbind(date=as.Date(rownames(time_msft)),time_msft)
MSFT_ret <- dailyReturn(MSFT_adj)[-1]

Goog_adj <- Ad(getSymbols("GOOG", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
time_goog <- as.data.frame(Goog_adj)
time_goog <- cbind(date=as.Date(rownames(time_goog)),time_goog)
Goog_ret <- dailyReturn(Goog_adj)[-1]

Csco_adj <- Ad(getSymbols("CSCO", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
time_csco <- as.data.frame(Csco_adj)
time_csco <- cbind(date=as.Date(rownames(time_csco)),time_csco)
Csco_ret <- dailyReturn(Csco_adj)[-1]

#Correlation
cor_Appl_MSFT <- cor(Appl_ret, MSFT_ret)
cor_Appl_Goog <- cor(Appl_ret, Goog_ret)
cor_Appl_Csco <- cor(Appl_ret, Csco_ret)
cor_MSFT_Goog <- cor(MSFT_ret, Goog_ret)
cor_MSFT_Csco <- cor(MSFT_ret, Csco_ret)
cor_Goog_Csco <- cor(Goog_ret, Csco_ret)
corelation <- c(cor_Appl_MSFT, cor_Appl_Goog, cor_Appl_Csco, cor_MSFT_Goog, cor_MSFT_Csco, cor_Goog_Csco)

num_company <- length(companies)
cor_ret <- matrix(NA, num_company, num_company)
diag(cor_ret) <- 1

k <- 1
for (i in 1 : (num_company - 1)) {
  for (j in (i+1) : num_company) {
    cor_ret[i,j] <- cor_ret[j,i] <- corelation[k]
    k = k + 1
  }
  k = 1
}
#Plot correlation


corrplot(cor_ret, method = "square")
#Corvariance
cov_Appl_MSFT <- cov(Appl_ret, MSFT_ret)
cov_Appl_Goog <- cov(Appl_ret, Goog_ret)
cov_Appl_Csco <- cov(Appl_ret, Csco_ret)
cov_MSFT_Goog <- cov(MSFT_ret, Goog_ret)
cov_MSFT_Csco <- cov(MSFT_ret, Csco_ret)
cov_Goog_Csco <- cov(Goog_ret, Csco_ret)
corvariance <- c(cov_Appl_MSFT, cov_Appl_Goog, cov_Appl_Csco,
                 cov_MSFT_Goog, cov_MSFT_Csco, cov_Goog_Csco)

cov_ret <- matrix(NA, num_company,num_company)
diag(cov_ret) <- c( cov(Appl_ret, Appl_ret), cov(MSFT_ret, MSFT_ret),cov(Goog_ret, Goog_ret), cov(Csco_ret, Csco_ret))

k <- 1
for (i in 1 : (num_company - 1)) {
```

```r
  for (j in (i+1) : num_company) {
    options(scipen = 999)
    cov_ret[i,j] <- cov_ret[j,i] <- corvariance[k]
    k = k + 1
  }
  k = 1
}

#Plot
Apple <- ggplot() +
  geom_line(aes(x = time_apple$date, y =time_apple$AAPL.Adjusted),
            colour = 'black') +
  ggtitle('APPLE Stock Price') +
  xlab('Price') +
  ylab('Time')

MSFT <- ggplot() +
  geom_line(aes(x = time_msft$date, y =time_msft$MSFT.Adjusted),
                  colour = 'black') +
        ggtitle('MSFT Stock Price') +
        xlab('Price') +
        ylab('Time')

      Google <- ggplot() +
        geom_line(aes(x = time_goog$date, y =time_goog$GOOG.Adjusted),
                  colour = 'black') +
        ggtitle('GOOGLE Stock Price') +
        xlab('Price') +
        ylab('Time')

      CSCO <- ggplot() +
        geom_line(aes(x = time_csco$date, y =time_csco$CSCO.Adjusted),
                  colour = 'black') +
        ggtitle('CSCO Stock Price') +
        xlab('Price') +
        ylab('Time')

      plot_grid(Apple, MSFT, Google, CSCO, labels = "AUTO")


      #Compute the Beta coefficients
      SP500_adj <- Ad(getSymbols("^GSPC", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
      time_sp <- as.data.frame(SP500_adj)
      time_sp <- cbind(date=as.Date(rownames(time_sp)),time_sp)
      SP500_ret <- dailyReturn(SP500_adj)[-1]





      beta_Appl <- cov(Appl_ret, SP500_ret)/var(SP500_ret)
      beta_MSFT <- cov(MSFT_ret, SP500_ret)/var(SP500_ret)
      beta_Goog <- cov(Goog_ret, SP500_ret)/var(SP500_ret)
      beta_Csco <- cov(Csco_ret, SP500_ret)/var(SP500_ret)
      beta_Port <- sum(c(beta_Appl,beta_MSFT,beta_Goog,beta_Csco)*weight)


      rects <- data.frame(xmin=as.Date("2018-01-20"), xmax=as.Date("2018-02-28"), ymin=-Inf, ymax=Inf)


      ggplot() +
        geom_line(aes(x = time_sp$date,
                      y = SP500_adj),
                  colour = 'black') +
```

```r
        geom_rect(data=rects, aes(xmin=xmin, xmax=xmax, ymin=ymin, ymax=ymax),
                  color="orange",
                  fill = "orange",
                  alpha=0.5,
                  inherit.aes = FALSE)+
        ggtitle('SP500') +
        xlab('Price') +
        ylab('Time')









      Port_beta <- cbind(beta_Appl, beta_MSFT, beta_Goog, beta_Csco)
      colnames(Port_beta) <- companies
      rownames(Port_beta) <- "Beta"
      #Calculate Alpha
      Tbill <- getSymbols("^TNX", src = "yahoo", start = maxDate, to = endDate)
      Tbill <- TNX['2017-01-01/2018-10-01']

      Tbill_ret <- dailyReturn(na.omit(Tbill$TNX.Adjusted))[-1]
      RF <- mean(na.omit(Tbill))
      Appl_rf <- CAPM.alpha(Appl_ret, SP500_ret, RF)
      Goog_rf <- CAPM.alpha(Goog_ret, SP500_ret, RF)
      MSFT_rf <- CAPM.alpha(MSFT_ret, SP500_ret, RF)
      Csco_rf <- CAPM.alpha(Csco_ret, SP500_ret, RF)
      Port_rf <- cbind(Appl_rf, MSFT_rf, Goog_rf, Csco_rf)
      colnames(Port_rf) <- companies
      rownames(Port_rf) <- "Alpha"


      #Calculate VaR by variacne-corvariance method
      Port_ret <- cbind(Appl_ret,MSFT_ret, Goog_ret, Csco_ret)
      mean_Port <- mean(Port_ret * weight)
      sd_Port = sd(Port_ret * weight)
      prob = qnorm(0.01)
      VaR_cc = mean_Port + prob * sd_Port
      var.portfolio(Port_ret, weights = weight)

      bigT_ret <- (Port_ret*weight)[,1] + (Port_ret*weight)[,2] +(Port_ret*weight)[,3] +
(Port_ret*weight)[,4]


      cor(Tbill_ret,bigT_ret)

      ggplot() +
        geom_point(aes(x = bigT_ret,
                       y =Tbill_ret),
                   colour = 'black') +
        ggtitle('Correlation between return of 10 Year T Bill and Big TechPortfolio') +
        xlab('Portfolio') +
        ylab('Tbill')



      port_price <- cbind(Appl_adj, MSFT_adj, Goog_adj, Csco_adj) * weight


      #Treynor Ratio and Sharpe Ratio
      trey <- TreynorRatio(bigT_ret, SP500_ret, Rf = RF, scale = 252)

      sharpe <- SharpeRatio.annualized(bigT_ret, Rf = RF,
```

```
                                        scale = 252, geometric = TRUE)


      #Fit into normal and t distribution
      #Define PDF, CDF and quantile of t distribution
      dt_G <- function(x, mean, sd, nu){
        dt((x-mean)/sd,nu)/sd
      }

      pt_G <- function(x, mean, sd, nu){
        pt((x-mean)/sd,nu)
      }
      #
      qt_G <- function(x, mean, sd, nu){
        qt(x,nu) * sd + mean
      }

      #Log return
      Appl_log <- as.numeric(na.omit(diff(log(Appl_adj))))
      Appl_G <- fitdist(Appl_log, "norm", start = list(mean = mean(Appl_log), sd = sd(Appl_log)))
      summary(Appl_G)[6:7]
      Appl_t <- fitdist(Appl_log, "t_G", start = list(mean = mean(Appl_log), sd = sd(Appl_log), nu = 5))
      summary(Appl_t)[6:7]

      MSFT_log <- as.numeric(na.omit(diff(log(MSFT_adj))))
      MSFT_G <- fitdist(MSFT_log, "norm", start = list(mean = mean(MSFT_log), sd = sd(MSFT_log)))
      summary(MSFT_G)[6:7]
      MSFT_t <- fitdist(MSFT_log, "t_G", start = list(mean = mean(MSFT_log), sd = sd(MSFT_log), nu = 5))
      summary(MSFT_t)[6:7]

      Goog_log <- as.numeric(na.omit(diff(log(Goog_adj))))
      Goog_G <- fitdist(Goog_log, "norm", start = list(mean = mean(Goog_log), sd = sd(Goog_log)))
      summary(Goog_G)[6:7]
      Goog_t <- fitdist(Goog_log, "t_G", start = list(mean = mean(Goog_log), sd = sd(Goog_log), nu = 5))
      summary(Goog_t)[6:7]

      Csco_log <- as.numeric(na.omit(diff(log(Csco_adj))))
      Csco_G <- fitdist(Csco_log, "norm", start = list(mean = mean(Csco_log), sd = sd(Csco_log)))
      summary(Csco_G)[6:7]
      Csco_t <- fitdist(Csco_log, "t_G", start = list(mean = mean(Csco_log), sd = sd(Csco_log), nu = 5))
      summary(Csco_t)[6:7]




      table_G <- rbind(summary(Appl_G)[6:7],summary(MSFT_G)[6:7],summary(Goog_G)[6:7],summary(Csco_G)[6:7])
      table_t <- rbind(summary(Appl_t)[6:7],summary(MSFT_t)[6:7],summary(Goog_t)[6:7],summary(Csco_t)[6:7])

      table_compare <- cbind(table_G, table_t)
      rownames(table_compare) <- companies



      #histogram plot and density of t and Gaussian
      histt <- function(data, ft, location = "topleft", legend.cex = 1, xlab){
```

```
        # parameters of the fitted t distribution
        mean_t <- as.list(ft$estimate)$mean
        sd_t <- as.list(ft$estimate)$sd
        nu_t <- as.list(ft$estimate)$nu

        # drawing histogram and assigning to h so that we can get the breakpoints between histogram cells (we will use it!)
        h <- hist(data, breaks=30)

        # x sequence for the additional plots on the histogram
        x_seq <- seq(-3,3,length=10000)

        # y sequence: density of fitted t distr. at x_seq
        yhistt <- dt_G(x_seq, mean=mean_t, sd=sd_t, nu=nu_t)

        # y sequence: density of normal distr. with mean and standard deviation of log-returns at x_seq
        # Note. I did not fit the normal distribution as we would get almost the same mean and standard deviation
        yhistNorm <- dnorm(x_seq, mean=mean(data), sd=sd((data)))

        # drawing histogram but this time we draw its density as y axis (freq=FALSE)
        hist(data,freq=FALSE,xlab=xlab, ylab="Density", breaks=h$breaks, main=paste(""),cex.lab=1.5)

        # adding the density of the fitted t distribution at x_seq
        lines(x_seq, yhistt, col=4)
        # adding the density of the normal distribution at x_seq
        lines(x_seq, yhistNorm, lty = "dashed")

        # Legend
        tmp.text <- c("t", "Gaussian")
        legend(location, legend = tmp.text, cex = legend.cex, lty = c(1,2), col=c(4,1))

      }

      histt(Appl_log, Appl_t, xlab="Daily log returns of Appl")
      histt(MSFT_log, MSFT_t, xlab="Daily log returns of MSFT")
      histt(Goog_log, Goog_t, xlab="Daily log returns of Goog")
      histt(Csco_log, Csco_t, xlab="Daily log returns of Csco")

      #Create the matrix of correlation of normal and t, compute the copula
      company_matrix <- matrix(nrow = length(Appl_log) , ncol = 4)
      company_matrix[,1] <- pt_G(Appl_log,
                                 mean = as.list(Appl_t$estimate)$mean,
                                 sd = as.list(Appl_t$estimate)$sd,
                                 nu = as.list(Appl_t$estimate)$nu)
      company_matrix[,2] <- pt_G(MSFT_log,
                                 mean = as.list(MSFT_t$estimate)$mean,
                                 sd = as.list(MSFT_t$estimate)$sd,
                                 nu = as.list(MSFT_t$estimate)$nu)
      company_matrix[,3] <- pt_G(Goog_log,
                                 mean = as.list(Goog_t$estimate)$mean,
                                 sd = as.list(Goog_t$estimate)$sd,
                                 nu = as.list(Goog_t$estimate)$nu)
      company_matrix[,4] <- pt_G(Csco_log,
                                 mean = as.list(Csco_t$estimate)$mean,
                                 sd = as.list(Csco_t$estimate)$sd,
                                 nu = as.list(Csco_t$estimate)$nu)

      norm.cop <- normalCopula(dim = 4,dispstr="un")
      n.cop <- fitCopula(norm.cop,company_matrix,method="ml")
      n.cop

      t.cop <- tCopula(dim = 4, dispstr = "un")
      t.cop <- fitCopula(t.cop, company_matrix, method = "ml")
```

```
      t.cop

      num_company <- length(companies)
      cor_norm <- matrix(NA, num_company, num_company)
      diag(cor_norm) <- 1
      k <- 1
      for (i in 1 : (num_company - 1)) {
        for (j in (i+1) : num_company) {
          cor_norm[i,j] <- cor_norm[j,i] <- as.numeric(coef(n.cop)[k])
          k = k + 1
        }
        k = 1
      }


      cor_t <- matrix(NA, num_company, num_company)
      diag(cor_t) <- 1
      for (i in 1 : (num_company - 1)) {
        for (j in (i+1) : num_company) {
          cor_t[i,j] <- cor_t[j,i] <- as.numeric(coef(t.cop)[k])
          k = k + 1
        }
        k = 1
      }


      #Defensive Portfolio
      #
      companies_defensive <- c("PRT","NEE","AEP","JNJ")
      weight_defensive <- c(0.25,0.25,0.25,0.25)
      #Get price and calculate the return
      #Defensive portfolio
      Prt_adj <- Ad(getSymbols("DOC", auto.assign = F, src = "yahoo", from = maxDate, to
= endDate))
      time_Prt <- as.data.frame(Prt_adj)
      time_Prt <- cbind(date=as.Date(rownames(time_Prt)),time_Prt)
      Prt_ret <- dailyReturn(Prt_adj)[-1]

      Nee_adj <- Ad(getSymbols("NEE", auto.assign = F, src = "yahoo", from = maxDate, to
= endDate))
      time_Nee <- as.data.frame(Nee_adj)
      time_Nee <- cbind(date=as.Date(rownames(time_Nee)),time_Nee)
      Nee_ret <- dailyReturn(Nee_adj)[-1]


      Aep_adj <- Ad(getSymbols("AEP", auto.assign = F, src = "yahoo", from = maxDate, to
= endDate))
      time_Aep <- as.data.frame(Aep_adj)
      time_Aep <- cbind(date=as.Date(rownames(time_Aep)),time_Aep)
      Aep_ret <- dailyReturn(Aep_adj)[-1]

      JNJ_adj <- Ad(getSymbols("JNJ", auto.assign = F, src = "yahoo", from = maxDate, to
= endDate))
      time_JNJ <- as.data.frame(JNJ_adj)
      time_JNJ <- cbind(date=as.Date(rownames(time_JNJ)),time_JNJ)
      JNJ_ret <- dailyReturn(JNJ_adj)[-1]

      #Compute Correlation of defensive stocks
      cor_Prt_Nee <- cor(Prt_ret, Nee_ret)
      cor_Prt_Aep <- cor(Prt_ret, Aep_ret)
      cor_Prt_JNJ <- cor(Prt_ret, JNJ_ret)
      cor_Nee_Aep <- cor(Nee_ret, Aep_ret)
      cor_Nee_JNJ <- cor(Nee_ret, JNJ_ret)
      cor_Aep_JNJ <- cor(Aep_ret, JNJ_ret)
      corelation_def <- c(cor_Prt_Nee,cor_Prt_Aep,cor_Prt_JNJ,cor_Nee_Aep,cor_Nee_JNJ,cor
_Aep_JNJ)
```

```
        num_company_def <- length(companies_defensive)
        cor_ret_def <- matrix(NA, num_company_def, num_company_def)
        diag(cor_ret_def) <- 1

        k <- 1
        for (i in 1 : (num_company_def - 1)) {
          for (j in (i+1) : num_company_def) {
            cor_ret_def[i,j] <- cor_ret_def[j,i] <- corelation_def[k]
            k = k + 1
          }
          k = 1
        }
        #Plot correlation


        # corrplot(cor_ret_def, method = "square")
        #Corvariance
        cov_Prt_Nee <- cov(Prt_ret, Nee_ret)
        cov_Prt_Aep <- cov(Prt_ret, Aep_ret)
        cov_Prt_JNJ <- cov(Prt_ret, JNJ_ret)
        cov_Nee_Aep <- cov(Nee_ret, Aep_ret)
        cov_Nee_JNJ <- cov(Nee_ret, JNJ_ret)
        cov_Aep_JNJ <- cov(Aep_ret, JNJ_ret)
        corvariacne_def <- c(cov_Prt_Nee,cov_Prt_Aep,cov_Prt_JNJ,
                             cov_Nee_Aep,cov_Nee_JNJ,cov_Aep_JNJ)

        cov_ret_def <- matrix(NA, num_company_def,num_company_def)
        diag(cov_ret_def) <- c( cov(Prt_ret, Prt_ret), cov(Nee_ret, Nee_ret),cov(Aep_ret, A
ep_ret), cov(JNJ_ret, JNJ_ret))

        k <- 1
        for (i in 1 : (num_company_def - 1)) {
          for (j in (i+1) : num_company_def) {
            options(scipen = 999)
            cov_ret_def[i,j] <- cov_ret_def[j,i] <- corvariacne_def[k]
            k = k + 1
          }
          k = 1
        }

        #Plot
        PRT <- ggplot() +
          geom_line(aes(x = time_Prt$date, y =time_Prt$DOC.Adjusted),
                    colour = 'black') +
          ggtitle('PRT Stock Price') +
          xlab('Price') +
          ylab('Time')

        NEE <- ggplot() +
          geom_line(aes(x = time_Nee$date, y =time_Nee$NEE.Adjusted),
                    colour = 'black') +
          ggtitle('NEE Stock Price') +
          xlab('Price') +
          ylab('Time')

        AEP <- ggplot() +
          geom_line(aes(x = time_Aep$date, y =time_Aep$AEP.Adjusted),
                    colour = 'black') +
          ggtitle('AEP Stock Price') +
          xlab('Price') +
          ylab('Time')

        JNJ <- ggplot() +
          geom_line(aes(x = time_JNJ$date, y =time_JNJ$JNJ.Adjusted),
                    colour = 'black') +
```

```
  ggtitle('JNJ Stock Price') +
  xlab('Price') +
  ylab('Time')

plot_grid(PRT, NEE, AEP, JNJ, labels = "AUTO")

#Beta of defensive portfolio
beta_Prt <- cov(Prt_ret, SP500_ret)/var(SP500_ret)
beta_Nee <- cov(Nee_ret, SP500_ret)/var(SP500_ret)
beta_Aep <- cov(Aep_ret, SP500_ret)/var(SP500_ret)
beta_JNJ <- cov(JNJ_ret, SP500_ret)/var(SP500_ret)
beta_Port_def <- sum(c(beta_Prt,beta_Nee,beta_Aep,beta_JNJ)*weight_defensive)

Port_beta_def <- cbind(beta_Prt, beta_Nee, beta_Aep, beta_JNJ)
colnames(Port_beta_def) <- companies_defensive
rownames(Port_beta_def) <- "Beta"


#Alpha of defensive portfolio
Prt_rf <- CAPM.alpha(Prt_ret, SP500_ret, RF)
Nee_rf <- CAPM.alpha(Nee_ret, SP500_ret, RF)
Aep_rf <- CAPM.alpha(Aep_ret, SP500_ret, RF)
JNJ_rf <- CAPM.alpha(JNJ_ret, SP500_ret, RF)
Port_rf_def <- cbind(Prt_rf, Nee_rf, Aep_rf, JNJ_rf)
colnames(Port_rf_def) <- companies_defensive
rownames(Port_rf_def) <- "Alpha"

#Calculate defensive portfolio variance
Port_ret_def <- cbind(Prt_ret, Nee_ret, Aep_ret, JNJ_ret)
mean_Port_def <- mean(Port_ret_def * weight_defensive)
sd_Port_def = sd(Port_ret_def * weight)
prob = qnorm(0.01)

var.portfolio(Port_ret_def, weights = weight_defensive)

def_ret <- (Port_ret_def*weight_defensive)[,1] + (Port_ret_def*weight_defensive)[,2] + (Port_ret_def*weight_defensive)[,3] + (Port_ret_def*weight_defensive)[,4]

cor(def_ret, Tbill_ret)


ggplot() +
  geom_point(aes(x = def_ret,
                 y =Tbill_ret),
             colour = 'black') +
  ggtitle('Correlation between return of TBill and Defensive Portfolio') +
  xlab('Defensive Portfolio') +
  ylab('Tbill')



port_price_def <- cbind(Prt_adj, Nee_adj, Aep_adj, JNJ_adj) * weight_defensive

rescale <- function(x) (x-min(x))/(max(x) - min(x)) * 100

# Log return of Defensive portfolio
Prt_log <- as.numeric(na.omit(diff(log(Prt_adj))))
Prt_G <- fitdist(Prt_log, "norm", start = list(mean = mean(Prt_log), sd = sd(Prt_log)))
summary(Prt_G)[6:7]
Prt_t <- fitdist(Prt_log, "t_G", start = list(mean = mean(Prt_log), sd = sd(Prt_log), nu = 5))
summary(Appl_t)[6:7]

Nee_log <- as.numeric(na.omit(diff(log(Nee_adj))))
```

```
    Nee_G <- fitdist(Nee_log, "norm", start = list(mean = mean(Nee_log), sd = sd(Nee_log)))
    summary(Nee_G)[6:7]
    Nee_t <- fitdist(Nee_log, "t_G", start = list(mean = mean(Nee_log), sd = sd(Nee_log), nu = 5))
    summary(Nee_t)[6:7]

    Aep_log <- as.numeric(na.omit(diff(log(Aep_adj))))
    Aep_G <- fitdist(Aep_log, "norm", start = list(mean = mean(Aep_log), sd = sd(Aep_log)))
    summary(Aep_G)[6:7]
    Aep_t <- fitdist(Aep_log, "t_G", start = list(mean = mean(Aep_log), sd = sd(Aep_log), nu = 5))
    summary(Aep_t)[6:7]

    JNJ_log <- as.numeric(na.omit(diff(log(JNJ_adj))))
    JNJ_G <- fitdist(JNJ_log, "norm", start = list(mean = mean(JNJ_log), sd = sd(JNJ_log)))
    summary(JNJ_G)[6:7]
    JNJ_t <- fitdist(JNJ_log, "t_G", start = list(mean = mean(JNJ_log), sd = sd(JNJ_log), nu = 5))
    summary(JNJ_t)[6:7]

    #plot normal, t distribution
    histt(Prt_log, Prt_t, xlab="Daily log returns of PRT")
    histt(Nee_log, Nee_t, xlab="Daily log returns of Nee")
    histt(Aep_log, Aep_t, xlab="Daily log returns of AEP")
    histt(JNJ_log, JNJ_t, xlab="Daily log returns of JNJ")


    company_matrix_def <- matrix(nrow = length(Prt_log) , ncol = 4)
    company_matrix_def[,1] <- pt_G(Prt_log,
                                   mean = as.list(Prt_t$estimate)$mean,
                                   sd = as.list(Prt_t$estimate)$sd,
                                   nu = as.list(Prt_t$estimate)$nu)
    company_matrix_def[,2] <- pt_G(Nee_log,
                                   mean = as.list(Nee_t$estimate)$mean,
                                   sd = as.list(Nee_t$estimate)$sd,
                                   nu = as.list(Nee_t$estimate)$nu)
    company_matrix_def[,3] <- pt_G(Aep_log,
                                   mean = as.list(Aep_t$estimate)$mean,
                                   sd = as.list(Aep_t$estimate)$sd,
                                   nu = as.list(Aep_t$estimate)$nu)
    company_matrix_def[,4] <- pt_G(JNJ_log,
                                   mean = as.list(JNJ_t$estimate)$mean,
                                   sd = as.list(JNJ_t$estimate)$sd,
                                   nu = as.list(JNJ_t$estimate)$nu)

    norm.cop.def <- normalCopula(dim = 4,dispstr="un")
    n.cop.def <- fitCopula(norm.cop.def,company_matrix_def,method="ml")
    n.cop.def

    t.cop.def <- tCopula(dim = 4, dispstr = "un")
    t.cop.def <- fitCopula(t.cop.def, company_matrix_def, method = "ml")
    t.cop.def

    num_company_def <- length(companies_defensive)
    cor_norm_def <- matrix(NA, num_company_def, num_company_def)
    diag(cor_norm_def) <- 1
    k <- 1
    for (i in 1 : (num_company_def - 1)) {
      for (j in (i+1) : num_company_def) {
        cor_norm_def[i,j] <- cor_norm_def[j,i] <- as.numeric(coef(n.cop.def)[k])
        k = k + 1
      }
      k = 1
```

```r
}


cor_t_def <- matrix(NA, num_company_def, num_company_def)
diag(cor_t_def) <- 1
for (i in 1 : (num_company_def - 1)) {
  for (j in (i+1) : num_company_def) {
    cor_t_def[i,j] <- cor_t_def[j,i] <- as.numeric(coef(t.cop.def)[k])
    k = k + 1
  }
  k = 1
}



#Garch
date_big = as.Date(rownames(time_apple))[-1]
blog <- cbind(Appl_log, MSFT_log, Goog_log, Csco_log) * weight
bigT_log <-xts(blog[,1] + blog[,2] + blog[,3] + blog[,4], date_big)




date_def = as.Date(rownames(time_Prt))[-1]
dlog <- cbind(Prt_log, Nee_log, Aep_log, JNJ_log) * weight_defensive
def_log <- xts(dlog[,1] + dlog[,2] + dlog[,3] +dlog[,4], date_def)

sp_log <- xts(SP500_ret, date_big)

gspec.ru <- ugarchspec(mean.model=list(armaOrder=c(0,0)),
                       variance.model=list(model="sGARCH", garchOrder=c(1,1)),
                       distribution.model="std")
bigTgar <- ugarchfit(gspec.ru, bigT_log)
defgar <- ugarchfit(gspec.ru, def_log)
spgar <- ugarchfit(gspec.ru, sp_log)





coef(bigTgar)
coef(defgar)
# plot volatility estimates


ggplot() +
  geom_line(aes(x= date_big,
                y= sqrt(252) * bigTgar@fit$sigma),
            colour = "#FF3399",
            size = 1) +
  geom_point(aes(x= date_big,
                 y= sqrt(252) * bigTgar@fit$sigma,colour = "colbig",shape = "colbig"),

             size = 2
             ) +
  geom_line(aes(x= date_def,
                y= sqrt(252) * defgar@fit$sigma),
            colour = "#FF9999",
            size = 1) +
  geom_point(aes(x= date_def,
                 y= sqrt(252) * defgar@fit$sigma,colour = "coldef",shape = "coldef"),
             size = 2
             )+
  geom_line(aes(x= date_big,
```

```
                        y= sqrt(252) * spgar@fit$sigma),
                    colour = "#0000CC",
                    size = 1) +
          geom_point(aes(x= date_big,
                         y= sqrt(252) * spgar@fit$sigma,colour = "colsp",shape = "colsp"),

                     size = 2)+
          scale_x_date(breaks = pretty_breaks(15))+
          scale_colour_manual(name="Volatility", values=c("colbig" = "#FF3399", "coldef"="#FF9999", "colsp" = "#0000CC"),
                              labels=c("colbig"="Big Tech", "coldef"="Defensive", "colsp" = "SP500"))+
          scale_shape_manual(name="Volatility", values=c("colbig" = 17, "coldef"=19, "colsp" = 18),
                             labels=c("colbig"="Big Tech", "coldef"="Defensive","colsp" = "SP500")
                             )+
          ggtitle('Garch(1,1), Volatility of Big Tech, Defensive Portfolio and SP500') +
          xlab('Time') +
          ylab('Volatility')










        #Treynor Ratio and Sharpe Ratio


        trey_def <- TreynorRatio(def_ret,
                                 SP500_ret, Rf = RF, scale = 252)

        sharpe_def <- SharpeRatio.annualized(def_ret,
                                             Rf = RF,
                                             scale = 252, geometric = TRUE)


        #Treynor Ratio and Sharpe Ratio Comparison

        tsR <- cbind(trey, sharpe)
        tsR_def <- cbind(trey_def, sharpe_def)
        tsR_comp <- rbind(tsR, tsR_def)
        colnames(tsR_comp) <- c("Treynor Ratio","Sharpe Ratio")
        rownames(tsR_comp) <- c("Big Tech Portfolio", "Defensive Portfolio")

        #Compare Drawdowns

        com_dd <- merge(bigT_ret,def_ret, SP500_ret)
        colnames(com_dd) <- c("Big Tech", "Defensive", "SP500")
        drawdowns <- table.Drawdowns(bigT_ret)
        drawdowns.dates <- cbind(format(drawdowns$From),format(drawdowns$To))
        drawdowns.dates[is.na(drawdowns.dates)] <- format(index(bigT_ret)[NROW(bigT_ret)])

        drawdowns.dates <- lapply(seq_len(nrow(drawdowns.dates)), function(i) drawdowns.dates[i,])


        charts.PerformanceSummary(com_dd,ylog=FALSE,
                                  period.areas = drawdowns.dates,
                                  colorset = c(2,3,4),
                                  legend.loc = "topleft",
```

```
                                 main = "Log Comparison among Big Tech, Defensive, and SP500" )






       chart.CumReturns(com_dd,ylog=FALSE,
                        period.areas = drawdowns.dates,
                        colorset = c(2,3,4),
                        legend.loc = "topleft",
                        main = "Cmulative")




       # advanced charts.PerforanceSummary based on ggplot
       gg.charts.PerformanceSummary <- function(rtn.obj, geometric = TRUE, main = "", plot = TRUE)
       {

         # load libraries
         suppressPackageStartupMessages(require(ggplot2))
         suppressPackageStartupMessages(require(scales))
         suppressPackageStartupMessages(require(reshape))
         suppressPackageStartupMessages(require(PerformanceAnalytics))

         # create function to clean returns if having NAs in data
         clean.rtn.xts <- function(univ.rtn.xts.obj,na.replace=0){
           univ.rtn.xts.obj[is.na(univ.rtn.xts.obj)]<- na.replace
           univ.rtn.xts.obj
         }

         # Create cumulative return function
         cum.rtn <- function(clean.xts.obj, g = TRUE)
         {
           x <- clean.xts.obj
           if(g == TRUE){y <- cumprod(x+1)-1} else {y <- cumsum(x)}
           y
         }

         # Create function to calculate drawdowns
         dd.xts <- function(clean.xts.obj, g = TRUE)
         {
           x <- clean.xts.obj
           if(g == TRUE){y <- PerformanceAnalytics::Drawdowns(x)} else {y <- PerformanceAnalytics:::Drawdowns(x,geometric = FALSE)}
           y
         }

         # create a function to create a dataframe to be usable in ggplot to replicate charts.PerformanceSummary
         cps.df <- function(xts.obj,geometric)
         {
           x <- clean.rtn.xts(xts.obj)
           series.name <- colnames(xts.obj)[1]
           tmp <- cum.rtn(x,geometric)
           tmp$rtn <- x
           tmp$dd <- dd.xts(x,geometric)
           colnames(tmp) <- c("Cumulative Return","Return","Drawdown") # names with space
           tmp.df <- as.data.frame(coredata(tmp))
           tmp.df$Date <- as.POSIXct(index(tmp))
           tmp.df.long <- melt(tmp.df,id.var="Date")
```

```
      tmp.df.long$asset <- rep(series.name,nrow(tmp.df.long))
      tmp.df.long
    }

    # A conditional statement altering the plot according to the number of assets
    if(ncol(rtn.obj)==1)
    {
      # using the cps.df function
      df <- cps.df(rtn.obj,geometric)
      # adding in a title string if need be
      if(main == ""){
        title.string <- paste("Asset Performance")
      } else {
        title.string <- main
      }

      gg.xts <- ggplot(df, aes_string( x = "Date", y = "value", group = "variable" )) +
        facet_grid(variable ~ ., scales = "free_y", space = "fixed") +
        geom_line(data = subset(df, variable == "Cumulative Return")) +
        geom_bar(data = subset(df, variable == "Return"), stat = "identity") +
        geom_line(data = subset(df, variable == "Drawdown")) +
        geom_hline(yintercept = 0, size = 0.5, colour = "black") +
        ggtitle(title.string) +
        theme(axis.text.x = element_text(angle = 0, hjust = 1)) +
        scale_x_datetime(breaks = date_breaks("6 months"), labels = date_format("%m/%Y")) +
        ylab("") +
        xlab("")

    }
    else
    {
      # a few extra bits to deal with the added rtn columns
      no.of.assets <- ncol(rtn.obj)
      asset.names <- colnames(rtn.obj)
      df <- do.call(rbind,lapply(1:no.of.assets, function(x){cps.df(rtn.obj[,x],geometric)}))
      df$asset <- ordered(df$asset, levels=asset.names)
      if(main == ""){
        title.string <- paste("Asset",asset.names[1],asset.names[2],asset.names[3],"Performance")
      } else {
        title.string <- main
      }

      if(no.of.assets>5){legend.rows <- 5} else {legend.rows <- no.of.assets}

      gg.xts <- ggplot(df, aes_string(x = "Date", y = "value" )) +

        # panel layout
        facet_grid(variable~., scales = "free_y", space = "fixed", shrink = TRUE, drop = TRUE, margin =
                           , labeller = label_value) + # label_value is default

        # display points for Cumulative Return and Drawdown, but not for Return
        geom_point(data = subset(df, variable == c("Cumulative Return","Drawdown"))
                   , aes(colour = factor(asset), shape = factor(asset)), size = 1.2, show.legend = TRUE) +

        # manually select shape of geom_point
        scale_shape_manual(values = c(1,2,3)) +

        # line colours for the Index
        geom_line(data = subset(df, variable == "Cumulative Return"), aes(colour = factor(asset)), show.legend = FALSE) +
```

```
            # bar colours for the Return
            geom_bar(data = subset(df,variable == "Return"), stat = "identity"
                     , aes(fill = factor(asset), colour = factor(asset)), position = "dodge", show.legend = FALSE) +
            # line colours for the Drawdown
            geom_line(data = subset(df, variable == "Drawdown"), aes(colour = factor(asset)), show.legend = FALSE) +
            # horizontal line to indicate zero values
            geom_hline(yintercept = 0, size = 0.5, colour = "black") +
            # horizontal ticks
            scale_x_datetime(breaks = date_breaks("3 months"), labels = date_format("%m/%Y")) +
            # main y-axis title
            ylab("") +
            # main x-axis title
            xlab("") +
            # main chart title
            ggtitle(title.string)
          # legend
          gglegend <- guide_legend(override.aes = list(size = 3))
          gg.xts <- gg.xts + guides(colour = gglegend, size = "none") +
            # gglegend <- guide_legend(override.aes = list(size = 3), direction = "horizontal") # direction overwritten by legend.box?
            # gg.xts <- gg.xts + guides(colour = gglegend, size = "none", shape = gglegend) + # Warning: "Duplicated override.aes is ignored"
            theme( legend.title = element_blank()
                   , legend.position = c(0,1)
                   , legend.justification = c(0,1)
                   , legend.background = element_rect(colour = 'grey')
                   , legend.key = element_rect(fill = "white", colour = "white")
                   , axis.text.x = element_text(angle = 0, hjust = 1)
                   , strip.background = element_rect(fill = "white")
                   , panel.background = element_rect(fill = "white", colour = "white")
                   , panel.grid.major = element_line(colour = "grey", size = 0.5)
                   , panel.grid.minor = element_line(colour = NA, size = 0.0)
            )
        }
        assign("gg.xts", gg.xts,envir=.GlobalEnv)
        if(plot == TRUE){
          plot(gg.xts)
        } else {}
      }
      # display chart
      gg.charts.PerformanceSummary(com_dd, geometric = TRUE)
      #EWMA Function
```

```
EWMA<-function(x,lambda)
{
  returns<-Delt(x,type='log') ##calculate log returns
  return_sq<-returns^2 ##square log returns
  y<-as.matrix(x) ##convert from xts to matrix
  n=(1:nrow(y)-1) ##this will be used for the weights
  z<-as.matrix(n) ##convert from numeric to matrix
  weights<-(1 - lambda)*lambda^z ##calculating the weights
  weights<-sort(weights,decreasing=FALSE) #arrange weights from least to greatest as return data is arranged from oldest to newest
  product<-weights*return_sq
  ##multiply weights times squared log returns
  product<-as.matrix(product) ##convert to matrix
  product<-na.omit(product) ##remove all Nas in data
  Variance<-colSums(product) ##sum the product
  Volatility<-sqrt(Variance)
  final<-cbind(Variance,Volatility)
  ##combine columns of Variance and Volatility
}
a <- EWMA(port_price,.94) #test the function

start <- as.Date("2016-01-01")
end <- as.Date("2017-01-01")
diff <- as.numeric(endDate - maxDate)
a <- matrix(NA, diff, 2)

for (i in 1 : diff) {
  Appl_adj <- Ad(getSymbols("AAPL", auto.assign = F, src = "yahoo", from = start, to = end))
  MSFT_adj <- Ad(getSymbols("MSFT", auto.assign = F, src = "yahoo", from = start, to = end))
  Goog_adj <- Ad(getSymbols("GOOG", auto.assign = F, src = "yahoo", from = start, to = end))
  Csco_adj <- Ad(getSymbols("CSCO", auto.assign = F, src = "yahoo", from = start, to = end))
  port_price <- Appl_adj*0.25 + MSFT_adj*0.25 + Goog_adj*0.25 + Csco_adj*0.25

  a[i,1] <- EWMA(port_price,.94)[,1]
  a[i,2] <- EWMA(port_price,.94)[,2]
  start = start + 1
  end = end + 1
```

```
        print(paste(i,"Volatility",a[i,2]))
      }
      b <- matrix(NA, diff, 2)
      for (i in 1 : diff) {
        Prt_adj <- Ad(getSymbols("DOC", auto.assign = F, src = "yahoo", from = start, to = end))
        Nee_adj <- Ad(getSymbols("NEE", auto.assign = F, src = "yahoo", from = start, to = end))
        Aep_adj <- Ad(getSymbols("AEP", auto.assign = F, src = "yahoo", from = start, to = end))
        JNJ_adj <- Ad(getSymbols("JNJ", auto.assign = F, src = "yahoo", from = start, to = end))
        port_price_def <- Prt_adj*0.25 + Nee_adj*0.25 + Aep_adj*0.25 +JNJ_adj*0.25
        b[i,1] <- EWMA(port_price_def,.94)[,1]
        b[i,2] <- EWMA(port_price_def,.94)[,2]
        start = start + 1
        end = end + 1
        print(paste(i,"Volatility",b[i,2]))
      }
      c <- matrix(NA, diff, 2)
      for (i in 1 : diff) {
        SP500_adj <- Ad(getSymbols("^GSPC", auto.assign = F, src = "yahoo", from = start, to = end))
        ew
        c[i,1] <- EWMA(SP500_adj,.94)[,1]
        c[i,2] <- EWMA(SP500_adj,.94)[,2]
        start = start + 1
        end = end + 1
        print(paste(i,"Volatility",c[i,2]))
      }
      write.csv(c, file = "sp500.csv")
```

VaR versus real loss Plot:

```
library(quantmod)
library(xts)
```

```
library(copula)
library(fitdistrplus)

setwd("/Users/Administrator/Desktop/1/ACF313/Group")
BTVaR <- read.csv("BTport_VaR.csv")
DefVaR <- read.csv("Defport_VaR.csv")

maxDate <- as.Date("2017-01-01")
endDate <- as.Date("2018-10-01")

time <- seq.Date(from = maxDate, to = endDate-1, by = "day")


AAPL_adj <- Ad(getSymbols("AAPL", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
MSFT_adj <- Ad(getSymbols("MSFT", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
Goog_adj <- Ad(getSymbols("GOOG", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
Csco_adj <- Ad(getSymbols("CSCO", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))

DOC_adj <- Ad(getSymbols("DOC", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
NEE_adj <- Ad(getSymbols("NEE", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
AEP_adj <- Ad(getSymbols("AEP", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
JNJ_adj <- Ad(getSymbols("JNJ", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))


retAAPL<-diff(log(AAPL_adj))[-1]
retMSFT<-diff(log(MSFT_adj))[-1]
retGoog<-diff(log(Goog_adj))[-1]
retCsco<-diff(log(Csco_adj))[-1]

retDOC<-diff(log(DOC_adj))[-1]
retNEE<-diff(log(NEE_adj))[-1]
retAEP<-diff(log(AEP_adj))[-1]
retJNJ<-diff(log(JNJ_adj))[-1]

BTportRet<-(retAAPL+retMSFT+retGoog+retCsco)/4
DefportRet<-(retDOC+retNEE+retAEP+retJNJ)/4

BTVaR <-xts(BTVaR,time)
DefVaR <-xts(DefVaR,time)

# BTVaR <- -BTVaR
# BTportRet <- -BTportRet
#
# DefVaR <- -DefVaR
# DefportRet<- -DefportRet
#
# Sys.setlocale("LC_TIME", "English")
#
# plot(BTVaR,main = "Big Tech VaR versus Real Loss",ylim = c(-0.04,0.04),col="#FF4500",lwd = 1)
# lines(BTportRet,col = "black",lwd = 1)
#
# plot(DefVaR,main = "Defensive VaR versus Real Loss",ylim = c(-0.03,0.03),col="#FF4500",lwd = 3)
# lines(DefportRet,col = "black",lwd = 2)
```

Monte Carlo simulated VaR:

```
library("copula")
library("fitdistrplus")
library(quantmod)

dateFrom <- as.Date("2017-01-01")
dateTo <- as.Date("2018-10-01")

diff <- as.numeric(dateTo - dateFrom)
BTport_VaR <-matrix(NA,diff)
Defport_VaR <-matrix(NA,diff)
time <- seq.Date(from = as.Date("2017/01/01",format = "%Y/%m/%d"), by = "day", length.out =
 diff)

dt_G <- function(x, mean, sd, nu){#density
  dt((x-mean)/sd,nu)/sd
}
pt_G <- function(x, mean, sd, nu){#CDF
  pt((x-mean)/sd,nu)
}
qt_G <- function(x, mean, sd, nu){#quantile
  qt(x,nu)*sd+mean
}

resFunc_BT<- function(x){
  for(k in 1:N){
    z <- rnorm(d)
    z_tilde <- L_BT%*%z

    R_AAPL <- exp(qt_G(pnorm(z_tilde[1]), mean=mean_AAPL,
                          sd=sd_AAPL, nu=nu_AAPL))

    R_MSFT <- exp(qt_G(pnorm(z_tilde[2]), mean=mean_MSFT,
                          sd=sd_MSFT, nu=nu_MSFT))

    R_Goog <- exp(qt_G(pnorm(z_tilde[3]), mean=mean_Goog,
                          sd=sd_Goog, nu=nu_Goog))

    R_Csco <- exp(qt_G(pnorm(z_tilde[4]), mean=mean_Csco,
                          sd=sd_Csco, nu=nu_Csco))

    res[k] <- 1*(0.25*R_AAPL+0.25*R_Goog+0.25*R_Csco+0.25*R_MSFT<x)
  }
  mean(res)-alpha
}

resFunc_Def<- function(x){
  for(k in 1:N){
    z <- rnorm(d)
    z_tilde <- L_Def%*%z

    R_DOC <- exp(qt_G(pnorm(z_tilde[1]), mean=mean_DOC,
                          sd=sd_DOC, nu=nu_DOC))

    R_NEE <- exp(qt_G(pnorm(z_tilde[2]), mean=mean_NEE,
                          sd=sd_NEE, nu=nu_NEE))

    R_AEP <- exp(qt_G(pnorm(z_tilde[3]), mean=mean_AEP,
                          sd=sd_AEP, nu=nu_AEP))

    R_JNJ <- exp(qt_G(pnorm(z_tilde[4]), mean=mean_JNJ,
                          sd=sd_JNJ, nu=nu_JNJ))

    res[k] <- 1*(0.25*R_DOC+0.25*R_NEE+0.25*R_AEP+0.25*R_JNJ<x)
```

```
  }
  mean(res)-alpha
}


search_x_BT <- function(a=0.7, b=1, err.tol=0.0001){
  fa <- resFunc_BT(a)
  fb <- resFunc_BT(b)
  while(abs(b-a) > err.tol){
    ab <- (a+b)/2
    fab <- resFunc_BT(ab)
    if((fab*fa)<0){
      b <- ab
      fb <- fab
    }else{
      a <- ab
      fa <- fab
    }
  }
  (a+b)/2
}

search_x_Def <- function(a=0.7, b=1, err.tol=0.0001){
  fa <- resFunc_Def(a)
  fb <- resFunc_Def(b)
  while(abs(b-a) > err.tol){
    ab <- (a+b)/2
    fab <- resFunc_Def(ab)
    if((fab*fa)<0){
      b <- ab
      fb <- fab
    }else{
      a <- ab
      fa <- fab
    }
  }
  (a+b)/2
}

d <- 4
alpha <- 0.05
N<-10000
res<-array(NA,dim=N)

norm.cop <- normalCopula(dim=4,dispstr="un")
maxDate <- as.Date("2016-01-01")
endDate <- as.Date("2017-01-01")

for(i in 1: diff){
  AAPL_adj <- Ad(getSymbols("AAPL", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
  MSFT_adj <- Ad(getSymbols("MSFT", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
  Goog_adj <- Ad(getSymbols("GOOG", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
  Csco_adj <- Ad(getSymbols("CSCO", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))

  DOC_adj <- Ad(getSymbols("DOC", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
  NEE_adj <- Ad(getSymbols("NEE", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
  AEP_adj <- Ad(getSymbols("AEP", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
  JNJ_adj <- Ad(getSymbols("JNJ", auto.assign = F, src = "yahoo", from = maxDate, to = endDate))
```

```
    retAAPL<-as.numeric(diff(log(AAPL_adj))[-1])
    retMSFT<-as.numeric(diff(log(MSFT_adj))[-1])
    retGoog<-as.numeric(diff(log(Goog_adj))[-1])
    retCsco<-as.numeric(diff(log(Csco_adj))[-1])

    retDOC<-as.numeric(diff(log(DOC_adj))[-1])
    retNEE<-as.numeric(diff(log(NEE_adj))[-1])
    retAEP<-as.numeric(diff(log(AEP_adj))[-1])
    retJNJ<-as.numeric(diff(log(JNJ_adj))[-1])

    ft_AAPL <- fitdist(retAAPL,"t_G",
                 start=list(mean=mean(retAAPL),
                       sd=sd(retAAPL),
                       nu=5))
    ft_MSFT <- fitdist(retMSFT,"t_G",
                 start=list(mean=mean(retAAPL),
                       sd=sd(retAAPL),
                       nu=5))
    ft_Goog <- fitdist(retGoog,"t_G",
                 start=list(mean=mean(retAAPL),
                       sd=sd(retAAPL),
                       nu=5))
    ft_Csco <- fitdist(retCsco,"t_G",
                 start=list(mean=mean(retAAPL),
                       sd=sd(retAAPL),
                       nu=5))

    ft_DOC <- fitdist(retDOC,"t_G",
                  start=list(mean=mean(retDOC),
                        sd=sd(retDOC),
                        nu=5))
    ft_NEE <- fitdist(retNEE,"t_G",
                  start=list(mean=mean(retNEE),
                        sd=sd(retNEE),
                        nu=5))
    ft_AEP <- fitdist(retAEP,"t_G",
                  start=list(mean=mean(retAEP),
                        sd=sd(retAEP),
                        nu=5))
    ft_JNJ <- fitdist(retJNJ,"t_G",
                  start=list(mean=mean(retJNJ),
                        sd=sd(retJNJ),
                        nu=5))



    u_BT<-matrix(nrow=length(retAAPL),ncol=4)
    u_BT[,1]<-pt_G(retAAPL,
             mean=as.list(ft_AAPL$estimate)$mean,
             sd=as.list(ft_AAPL$estimate)$sd,
             nu=as.list(ft_AAPL$estimate)$nu)

    u_BT[,2]<-pt_G(retMSFT,
             mean=as.list(ft_MSFT$estimate)$mean,
             sd=as.list(ft_MSFT$estimate)$sd,
             nu=as.list(ft_MSFT$estimate)$nu)

    u_BT[,3]<-pt_G(retGoog,
             mean=as.list(ft_Goog$estimate)$mean,
             sd=as.list(ft_Goog$estimate)$sd,
             nu=as.list(ft_Goog$estimate)$nu)

    u_BT[,4]<-pt_G(retCsco,
             mean=as.list(ft_Csco$estimate)$mean,
```

```
                          sd=as.list(ft_Csco$estimate)$sd,
                          nu=as.list(ft_Csco$estimate)$nu)

             u_Def<-matrix(nrow=length(retDOC),ncol=4)
             u_Def[,1]<-pt_G(retDOC,
                               mean=as.list(ft_DOC$estimate)$mean,
                               sd=as.list(ft_DOC$estimate)$sd,
                               nu=as.list(ft_DOC$estimate)$nu)

             u_Def[,2]<-pt_G(retNEE,
                               mean=as.list(ft_NEE$estimate)$mean,
                               sd=as.list(ft_NEE$estimate)$sd,
                               nu=as.list(ft_NEE$estimate)$nu)

             u_Def[,3]<-pt_G(retAEP,
                               mean=as.list(ft_AEP$estimate)$mean,
                               sd=as.list(ft_AEP$estimate)$sd,
                               nu=as.list(ft_AEP$estimate)$nu)

             u_Def[,4]<-pt_G(retJNJ,
                               mean=as.list(ft_JNJ$estimate)$mean,
                               sd=as.list(ft_JNJ$estimate)$sd,
                               nu=as.list(ft_JNJ$estimate)$nu)


             n.cop_BT <- fitCopula(norm.cop,u_BT,method="ml")

             n.cop_Def <- fitCopula(norm.cop,u_Def,method="ml")

             matrixdata_BT<-as.numeric(coef(n.cop_BT))
             matrixdata_Def<-as.numeric(coef(n.cop_Def))

             mean_AAPL <- as.list(ft_AAPL$estimate)$mean
             sd_AAPL <- as.list(ft_AAPL$estimate)$sd
             nu_AAPL <- as.list(ft_AAPL$estimate)$nu

             mean_MSFT <- as.list(ft_MSFT$estimate)$mean
             sd_MSFT <- as.list(ft_MSFT$estimate)$sd
             nu_MSFT <- as.list(ft_MSFT$estimate)$nu

             mean_Goog <- as.list(ft_Goog$estimate)$mean
             sd_Goog <- as.list(ft_Goog$estimate)$sd
             nu_Goog <- as.list(ft_Goog$estimate)$nu

             mean_Csco <- as.list(ft_Csco$estimate)$mean
             sd_Csco <- as.list(ft_Csco$estimate)$sd
             nu_Csco <- as.list(ft_Csco$estimate)$nu


             mean_DOC <- as.list(ft_DOC$estimate)$mean
             sd_DOC <- as.list(ft_DOC$estimate)$sd
             nu_DOC <- as.list(ft_DOC$estimate)$nu

             mean_NEE <- as.list(ft_NEE$estimate)$mean
             sd_NEE <- as.list(ft_NEE$estimate)$sd
             nu_NEE <- as.list(ft_NEE$estimate)$nu

             mean_AEP <- as.list(ft_AEP$estimate)$mean
             sd_AEP <- as.list(ft_AEP$estimate)$sd
             nu_AEP <- as.list(ft_AEP$estimate)$nu

             mean_JNJ <- as.list(ft_JNJ$estimate)$mean
             sd_JNJ <- as.list(ft_JNJ$estimate)$sd
             nu_JNJ <- as.list(ft_JNJ$estimate)$nu

             cor_BT<-matrix(data=c(1,matrixdata_BT[1],matrixdata_BT[2],matrixdata_BT[3],
```

```
                                   matrixdata_BT[1],1,matrixdata_BT[4],matrixdata_BT[5],
                                   matrixdata_BT[3],matrixdata_BT[4],1,matrixdata_BT[6],
                                   matrixdata_BT[3],matrixdata_BT[5],matrixdata_BT[6],1),nrow=4,ncol=4)

         cor_Def<-matrix(data=c(1,matrixdata_Def[1],matrixdata_Def[2],matrixdata_Def[3],
                                   matrixdata_Def[1],1,matrixdata_Def[4],matrixdata_Def[5],
                                   matrixdata_Def[3],matrixdata_Def[4],1,matrixdata_Def[6],
                                   matrixdata_Def[3],matrixdata_Def[5],matrixdata_Def[6],1),nrow=4,ncol=4)


         L_BT <- t(chol(cor_BT))
         L_Def <- t(chol(cor_Def))

         seed <- sample(1000:5000,1)
         set.seed(seed)

         BTport_VaR[i] <- search_x_BT()-1
         Defport_VaR[i] <- search_x_Def()-1

         maxDate = maxDate + 1
         endDate = endDate + 1

         print(paste(i,"95%:","Big Tech:", BTport_VaR[i],
                                  "Defensive:", Defport_VaR[i]))

       }

       time <- seq.Date(from = as.Date("2017/01/01",format = "%Y/%m/%d"), by = "day", length.out = diff)

       BTVaR <-xts(BTport_VaR,time)
       DefVaR <-xts(Defport_VaR,time)

       Sys.setlocale("LC_TIME", "English")

       plot(BTVaR,main = "Big Tech VaR")
       plot(DefVaR,main = "Defensive VaR")

       plot(BTVaR,main = "Big Tech VaR versus Defensive VaR",ylim = c(-0.020,-0.005))
       lines(DefVaR,lwd = 2,col = "red")

       write.table (BTport_VaR,file ="BTport_VaR.csv", sep =",")
       write.table (Defport_VaR,file ="Defport_VaR.csv", sep =",")
```