Combination VaR Forecasts Using Elastic Net Penalized Quantile Regression

Yuxiang Li

Department of Mathematics

Rutgers, The State University of New Jersey

Abstract

Value at risk (VaR) is a very useful tool to quantify extreme losses. There are many different methodologies to forecast value at risk. However, those methodologies always produce widely different VaR forecasts. In this paper, we try to combine some frequently used VaR forecasts by penalized quantile regression. We randomly choose 30 stocks from SP500 to form a equally weighted portfolio. Then we compare penalized quantile regression with other standalone VaR methods as well as two combination methods(Simple average and unpenalized quantile regression). To evaluate different forecasts methods, we perform backtesting using unconditional coverage test and conditional coverage test. The results show that penalized quantile regressions perform better than standalone methods.

1. Introduction

The value at risk (VaR) is defined as the worst possible loss over a target horizon that will not be exceeded with a given probability (Jorion, 2006). Forecasting VaR has attracted a great deal of attention, since the Basel Committee on Banking Supervision (1996, 2006, 2011) adopted VaR to calculate the minimum capital requirements to cover the market risk for banks.

Many standalone VaR forecasting methods have been introduced. Berkowitz, Christoffersen, and Pelletier (2011), Prignon and Smith (2008, 2010b) and Pritsker (2006) discuss historical simulation (HS), which is the oldest procedure of forecasting VaR. Based on HS, Zikovic and Aktan (2011) proposed forecasting VaR using weighted historical simulation (WHS). An alternative forecast is conditional autoregressive value at risk (CAViaR) introduced by Engle and Manganelli (2004). Since VaR can be seen as the quantile of the distribution of returns, another method tries to model this tail distributions of portfolio returns using extreme value theory (EVT). This method uses scale and shape parameters to model tail distributions without making any assumptions about center distributions. Some other methods pay attention to estimate volatility of returns and calculate VaR forecasts. Among these methods, GARCH(1, 1) of Bollerslev (1986) , EGARCH(1, 1) of Nelson (1991) and the AP-ARCH(1, 1) of Ding et al. (1993) are the most popular models used to estimate volatility.

However, no single VaR forecast outperforms others throughout the existing VaR forecasting comparisons. They all have different advantages and disadvantages. All models are prone to suffer from model misspecification and estimation uncertainty. Thus, people begin to seek a better forecasting by combining predictions from different models. In this paper, we first study some mainly used standalone VaR forecasting methods and combination forecasting methods. In the empirical parts, we first form a portfolio by randomly choosing 30 stocks from SP500 from Jan 1, 2014 to Dec 31, 2017. Then we apply 4 traditional VaR forecasts method, including historical simulation, static normal distribution, extreme value theory and GARCH(1, 1). We combine these 4 forecasts by quantile regression with elastic net penalty of Zou and Hastie (2005), which is a linear combination of the ridge penalty of Hoerl and Kennard (1970a,b) and the lasso of Tibshirani (1996). Finally, by comparing with standalone VaR forecasts and combination forecasts, we find penalized quantile regression is a better combination method for VaR forecasting.

The rest of this paper is organized as follows. Section 2 reviews recent studies of VaR combination methods. Section 3 introduces the methodology and details about penalized quantile regression. Section 4 describes the data we use. Section 5 presents the main results with forms and figures. Section 6 includes some discussion. Section 7 gives a conclusion of this paper.

2. Literature Review

Since single VaR models suffer from different deficiencies, especially during financial crisis period when all financial risk measures failed to provide enough protect for portfolios, people try to make combinations of different VaR models for a better forecast.

Forecasts combination has a very long history. There are several reasons that people believe for using Forecasts combination. First is that different forecasts models might cover different information sets. Thus, combination forecasts would generate a smaller expected loss. The second reason for combining forecasts is that structural breaks might affect forecasts differently. See Bates and Granger (1969), Figlewski and Urich (1983), Diebold and Pauly (1987) and Timmermann (2006). The third reason is emphasized by Clemen (1989), Makridakis (1989), Diebold and Lopez (1996) and Stock and Watson (2001, 2004) that single model would have misspcification bias. Timmermann (2006) systematically analyzes the factors that determine the advantages from combining forecasts. He concludes that combination forecasts are more stable than standalone forecasts in terms of diversified gains, robustness and model misspecification.

Jimenez Martin et al. (2011) uses 12 new strategies based on combinations of different VaR forecasts. For example, lowwer bound, upper bound, average and nine additional strategies based on the 10th, ... 50th, ... 90th percentiles. The methods are very straightforward, but they do not improve standalone VaR forecasts too much. Halbleib and Pohlmeier (2012) develop a new method to combine VaR forecasts by using quantile regression, which is introduced by Koenker and Bassett (1978). Since quantile regression estimator tries to minimize tick loss function, it is reasonable to incorporate the tick loss for the estimation and evaluation of VaR forecasts. However, since different VaR forecasts would always move together, the combination weights might become unstable because of multicollinearity in quantile regression. Based on this point of view, Bayer (2018) suggests using penalized quantile regression for the combination of VaR forecasts. And thorough comparison analysis, the penalized quantile regressions perform better in terms of backtesting and tick losses than the standalone models and several competing forecast combination approaches.

3. Methodology

3.1. Notation

α confidence level, set to be 1% throughout the paper.

h forecast horizon, set to be 1 day throughout the paper.

$Combo\_VaR_{t+1|t}$ combination VaR forecast for day $t+1$ based on information available at day t.

$VaR_{i,t+1|t}$ ith standalone VaR forecast for day $t+1$ based on information available at day t.

$\beta_{i,t+1|t}$ combination weights for ith VaR forecast for day $t+1$ based on information available at day t.

3.2. Quantile regression

We try to estimate combination weights based on quantile regression (Koenker and Basset, 1978) and model α quantile of return as a linear function of VaR_i.

$$R = \beta_{0,t+1|t} + \sum_{i=1}^{n} \beta_{i,t+1|t} VaR_{i,t+1|t} + \epsilon \tag{1}$$

Just as linear regression estimates the mean value of the response variable for given levels of the predictor variables, quantile regression models the relation between a set of predictor variables and specific percentiles (or quantiles) of the response variable. The quantile regression parameter estimates the change in a specified quantile of the response variable produced by a one unit change in the predictor variable.

Thus, in order to get a quantile regression estimates of equation (1), we need to first define a piecewise linear tick loss function

$$\rho(u) = (\alpha - 1_{\{u<0\}})u \tag{2}$$

Then we can estimate β_i by minimize the cost function

$$\boldsymbol{\beta} = \operatorname*{argmin}_{\boldsymbol{\beta}} \sum \rho(R - \beta_{0,t+1|t} - \sum_{i=1}^{n} \beta_{i,t+1|t} VaR_{i,t+1|t}) \tag{3}$$

One advantage of quantile regression, relative to the ordinary least squares regression, is that while OLS can be inefficient if the errors are highly non-normal, QR is more robust to non-normal errors and outliers.

3.3. Elastic net penalty

Although quantile regression is an ideal method for VaR forecasts combination, the high correlation among different VaR forecasts could induce unstable estimator and overfitting problem. That is why penalty is necessary for our models.

The elastic net penalty of Zou and Hastie (2005) is a linear combination of lasso penalty of Tibshirani (1996) and ridge penalty of Hoerl and Kennard (1970a,b). The result of the elastic net penalty is a combination of the effects of the lasso and Ridge penalties, which include variable selection and estimator shrinking. The quantile regression cost function under elastic net penalty is given by

$$\boldsymbol{\beta} = \operatorname*{argmin}_{\boldsymbol{\beta}} \sum \rho(R - \beta_{0,t+1|t} - \sum_{i=1}^{n} \beta_{i,t+1|t} VaR_{i,t+1|t}) + \lambda(\delta||\boldsymbol{\beta}||_1 + (1-\delta)||\boldsymbol{\beta}||_2^2) \tag{4}$$

where λ is the regularization parameter and $\delta \in [0, 1]$. And the this can be estimated through the R (R Core Team, 2016) implementation of Yi (2017) in the 'hqreg' library.

3.4. Becktesting and model comparison

Since we use a rolling window to keep calculating VaR forecasts and VaR combination forecasts, we will end up getting several time series of VaR forecasts. We will assess their performance under different situations. For example, over the whole time horizon, during financial crisis and so on. We also apply unconditional coverage (UC) test by Kupiec (1995) and conditional coverage (CC) test by Christoffersen (1998) to see whether VaR models are valid.

4. Data description

The dataset we use are the daily adjusted close price of 30 constituents of SP500 from Jan 1, 2014 to Dec 31, 2017, a total of 3524 days[1]. And our portfolio is formed by equally weighted investing on those assets. Then we use a 500-day rolling window to calculate 1-day ahead VaR forecasts with 4 different methods, which will give us 4 time series of length 3023. We keep using a 500-day rolling window to estimate combination weights by quantile regression to calculate 1-day $Combo\_VaR$. Thus, we will lose 1000 data and end up getting different 1-day VaR forecasts from Dec 24, 2007 to Dec 29, 2017, a total of 2523 days.

4.1. Standalone VaR forecasts

We choose 4 widely used VaR forecasts models to estimate our standalone VaR forecasts. These models cover parametric, semi-parametric and non-parametric methods.

4.1.1. Normal distribution

This method assumes the return of assets is normally distributed with mean μ and variance σ^2. Thus, to calculate VaR is to calculate the quantile

[1] The symbols of the assets are: MCD, XEL, HOG, UNH, PAYX, ADM, APA, CB, COL, COP, PPL, T, FCX, CVS, GT, MCK, GPC, APD, BBY, APC, ORCL, AEE, AFL, NEM, DE, FLR, JWN, EOG, PNW, CAG

of a gaussian distribution.

$$VaR^N_{t+1|t} = \hat{\mu} + \hat{\sigma}\Phi^{-1}(\alpha) \tag{5}$$

where $\Phi^{-1}(\cdot)$ is the inverse of standard normal distribution. $\hat{\mu}$ and $\hat{\sigma}$ are estimators of sample mean and sample standard deviation. In this paper, we use a rolling window of 500 observations to estimate μ and σ.

The normal method is very simple and fast. However, the biggest defect of this method is the normality assumption. As we all know, financial asset returns have a very fat tail distribution, which means we are underestimating our risks by using normal distribution.

4.1.2. Historical simulation

Historical simulation is a classic and popular non-parametric method. Without any assumption about return distribution, it predicts next day's VaR by the empirical α-quantile of a period of past returns.

$$VaR^{HS}_{t+1|t} = R(\omega) \tag{6}$$

where $R(\omega)$ is the ωth-order statistic of returns, and $\omega = T * \alpha$.

Historical simulation is popular due to its simplicity. And it also overcomes the deficiency of wrong distribution assumption. However, VaR^{HS} tends to have big jumps whenever our window moves into or out of a large movement of returns.

4.1.3. Extreme value theory

Since we only focus on the tail part of return distribution for risk management, extreme value theory is a very useful tool that models the tail distribution without any assumption about the center of the distribution. Under extreme value theory, the cumulative distribution function (CDF) of a variable x beyond a cutoff point u is

$$F(x) = 1 - \frac{N_u}{N}(1 + \frac{\xi}{\beta}(x-u))^{-1/\xi} \tag{7}$$

where β is the scale parameter and ξ is the shape parameter. And as ξ tends to zero, This CDF will reduce to normal distribution. When $\xi > 0$, it implies heavy tail distribution. Then, VaR forecasts under extreme value theory are given by

$$VaR^{EVT}_{t+1|t} = u + \frac{\hat{\beta}}{\hat{\xi}}\{[\frac{N}{N_u}(1-\alpha)]^{-\hat{\xi}} - 1\} \tag{8}$$

$\hat{\beta}$ and $\hat{\xi}$ are estimators of scale and shape parameters, which we will estimate using maximum likelihood. And the log likelihood function is

$$(\hat{\beta}, \hat{\xi}) = \underset{\beta,\xi}{\operatorname{argmax}} \sum \ln [\frac{1}{\beta}(1 + \frac{\xi}{\beta}(x-u))^{-1/\xi - 1}] \tag{9}$$

Since the method above models right tail of the distribution, we will first multiply portfolio returns by -1 and then apply above method to estimate VaR^{EVT}.

The EVT approach is very useful for estimating tail probabilities of extreme events, especially when empirical data suffers from a lock of data in tails.

4.1.4. GARCH model

The GARCH method models the conditional variance of return. It assumes that the variance of return follows a predictable pattern. We use the GARCH(1, 1) of Bollerslev (1986).

$$\sigma_t^2 = \omega + \alpha r_{t-1} + \beta\sigma_{t-1}^2 \tag{10}$$

And VaR forecasts under GARCH(1, 1) method are given by

$$ {}_{t+1|t}^{GARCH} = \mu_{t+1|t} + \sigma_{t+1|t} Q_\alpha(Z_t) \quad (11) $$

where $\mu_{t+1|t}$ and $\sigma_{t+1|t}$ are one-step-ahead forecasts of the mean and the volatility, $Q_\alpha(Z_t)$ is the unconditional α-quantile of the innovations. We assume $\mu_{t+1|t}$ equals to previous period sample mean, and the innovation term is a standard normal distribution. And the model estimation can be done through R implementation of Ghalanos (2018) in the 'rugarch' library.

GARCH model can display the volatility cluster of financial series. But, its drawback is the nonlinearity and great calculation.

4.2. Combination VaR forecasts

This section introduces the combination methods we use. There are 3 combinations, including simple mean, unpenalized quantile regression and penalized quantile regression with elastic net penalty.

4.2.1. Simple mean

Simple mean is a very simple and empirically successful method in combining forecasts. The combination weights of simple mean approach are given by

$$ \beta_{i,t+1|t} = \frac{1}{N},\ for\ all\ i\ in\ 1, 2, ..., N \quad (12) $$

4.2.2. Unpenalized quantile regression

As a comparison between quantile regression methods, we adopt the unpenalized quantile regression by Koenker (2016). The combination weights of unpenalized quantile regression are given by

$$ \boldsymbol{\beta} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum \rho(R - \beta_{0,t+1|t} - \sum_{i=1}^{n} \beta_{i,t+1|t} VaR_{i,t+1|t}) \quad (13) $$

4.2.3. Elastic net penalized quantile regression

Since different VaR forecasts are highly correlated to each other, Bayer (2018) suggests using penalized quantile regression to deal with multicollinearity. Elastic net penalty is a linear combination of lasso penalty and ridge penalty. The combination weights of elastic net penalized quantile regression are given by

$$\begin{aligned}\boldsymbol{\beta} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum \rho(R - \beta_{0,t+1|t} - \sum_{i=1}^{n} \beta_{i,t+1|t} VaR_{i,t+1|t}) \\ +\lambda(\delta||\boldsymbol{\beta}||_1 + (1-\delta)||\boldsymbol{\beta}||_2^2)\end{aligned} \tag{14}$$

4.2.4. Backtesting

We will assess the performance of different approached by the unconditional coverage backtest by Kupiec (1995) and the conditional coverage backtest by Engle and Manganelli (2004).

4.2.5. Unconditional coverage

This method focuses on VaR failures. By defining failure rate as $\frac{N}{T}$, where N is the number of exceptions and T is the number of total observations, we would expect the failure rate not far from the confidence level α. Then we can apply a likelihood ratio test. The test statistic is given by

$$LR_{uc} = -2\ln\left[(1-\alpha)^{T-N}\alpha^N\right] + 2\ln\left\{[1-(N/T)]^{T-N}(N/T)N\right\} \tag{15}$$

where N is the number of exceptions observed, T is the number of total observations. LR_{uc} is chi-square distributed with one degree of freedom under the null hypothesis that α is the true probability. Thus, we would reject null hypothesis at 99% test confidence level if $LR_{uc} > 6.635$.

4.2.6. Conditional coverage

The unconditional coverage model ignores time variation in the data. The VaR forecasts would also be invalid if the exceptions cluster over time. Thus, the deviation should be serially independent if the model were to be true. Christoffersen (1998) proposes the very influential conditional coverage test. The test statistic is given by

$$LR_{cc} = LR_{uc} + LR_{ind} \tag{16}$$

$$\begin{aligned} LR_{ind} = &-2\ln\left[(1-\pi)^{(}T_{00}+T_{10})\pi^{(}T_{01}+T_{11})\right] \\ &+2\ln\left[(1-\pi_0)^{T_{00}}\pi_0^{T_{01}}(1-\pi_1)^{T_{10}}\pi_1^{T_{11}}\right] \end{aligned} \tag{17}$$

where T_{ij} is the number of days in which state j occurs in one day and state i occurred at the previous day. π_i is the probability of an exception conditional on state i the previous day. And here we set state 1 to be an exception. Thus, we have

$$\pi = \frac{T_{01}+T_{11}}{T_{00}+T_{01}+T_{10}+T_{11}} \tag{18}$$

$$\pi_0 = \frac{T_{01}}{T_{00}+T_{01}} \tag{19}$$

$$\pi_1 = \frac{T_{11}}{T_{10}+T_{11}} \tag{20}$$

The test statistic for conditional coverage, LR_{cc} is chi-square distributed with two degree of freedom. Thus, we would reject the null at 99% test confidence level if $LR_{cc} > 9.211$. And we would reject independence alone if $LR_{ind} > 6.635$.

5. Results

5.1. VaR forecasts results

5.1.1. Standalone VaR forecasts

Figure 1-4 show the results of 4 standalone VaR forecasts, together with the portfolio returns and exceptions. As we can see from the plot, all these 4 standalone VaR forecasts have more exceptions than expected, especially during fluctuating period. And there are obvious clustering phenomena for historical simulation, normal and extreme value theory models.

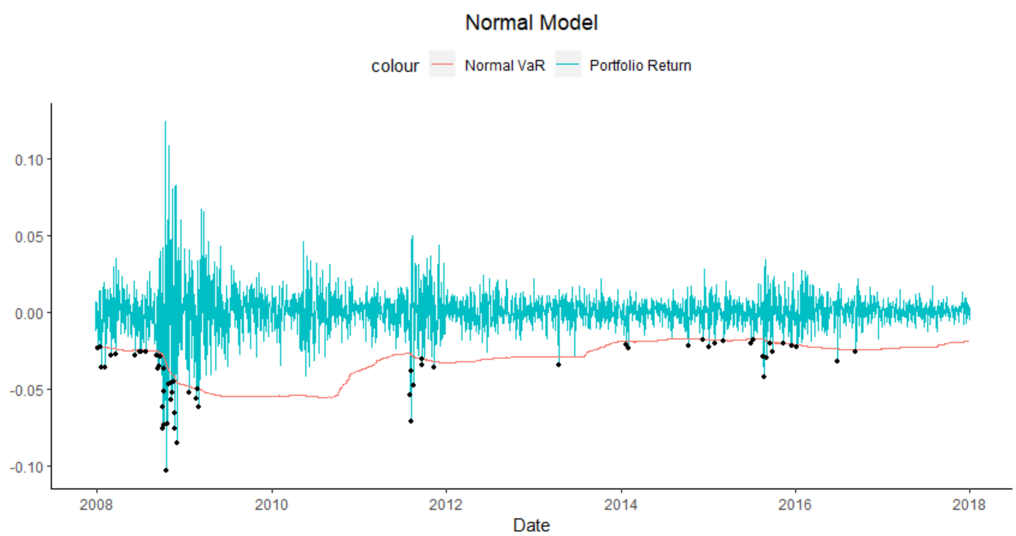

Figure 1: Normal Model VaR Forecast

The figures show the characteristics of each methods as we talk above. Normal model always underestimate risks, thus its VaR forecasts provide less protection than others. Historical simulation has big jumps every time when our window moves in or out of a fluctuating period. GARCH(1, 1) model is very good at estimating volatility movements. And there seems no obvious clustering from the plot.

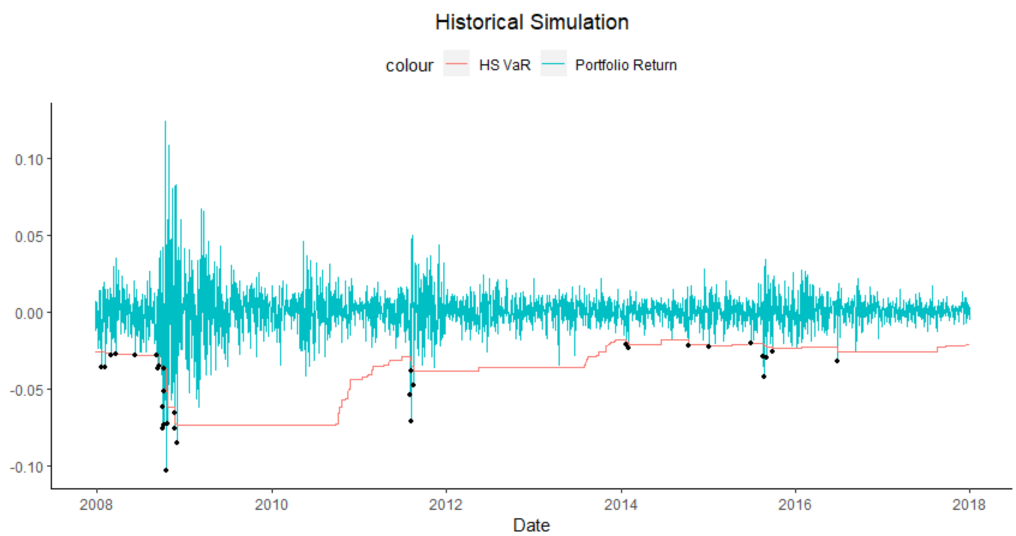

Figure 2: Historical Simulation VaR Forecast

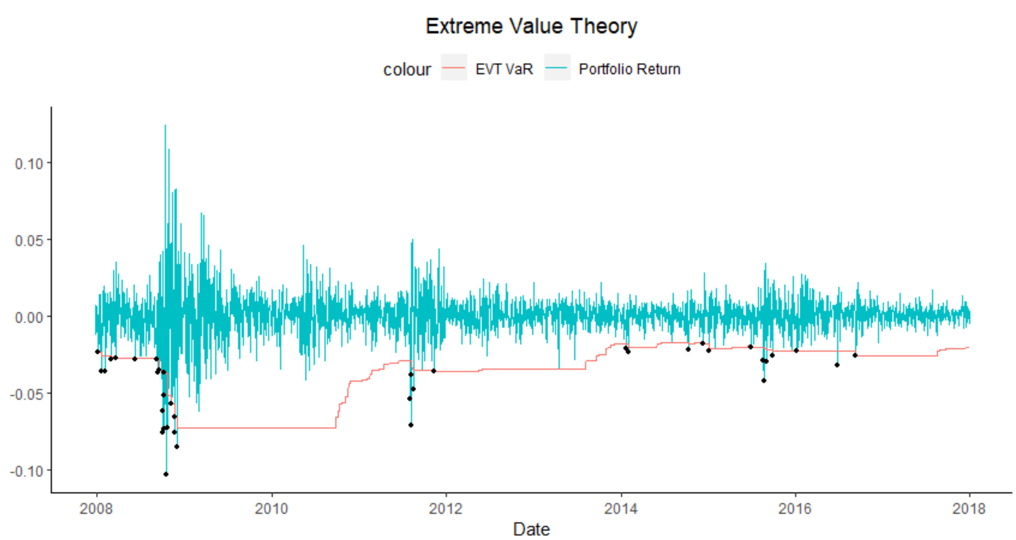

Figure 3: EVT VaR Forecast

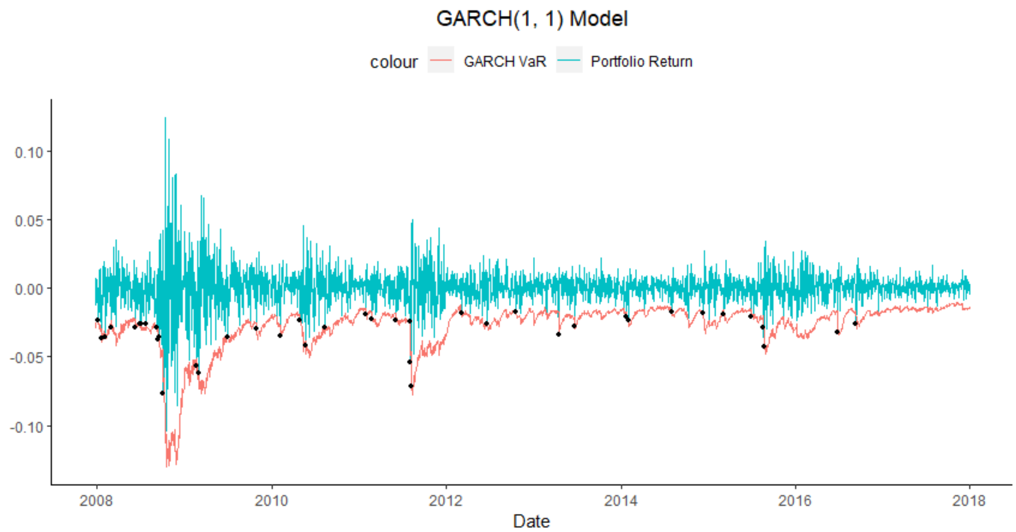

Figure 4: GARCH(1, 1) VaR Forecast

5.1.2. Combination VaR forecasts

Figure 5-7 show the results of 3 combination VaR forecasts, together with the portfolio returns and exceptions. As we can see, the combination forecasts are much better than standalone forecasts. And there seems to be clustering in simple mean and unpenalized quantile regression methods. Simple mean combination compromises different standalone VaR forecasts. Although it is a simple method, its performance is even better than unpenalized QR method. And from the plot, we could feel that penalized quantile regression method might be the best combination of the three. There are fewer exceptions and no sign of clustering. And it also get some features from GARCH(1, 1) model.

In order to evaluate these approached more precisely, we need to conduct statistical test on them.

5.2. Backtesting

In order to test the validation of VaR forecasts, we perform the mostly used method unconditional coverage and conditional coverage tests. We test

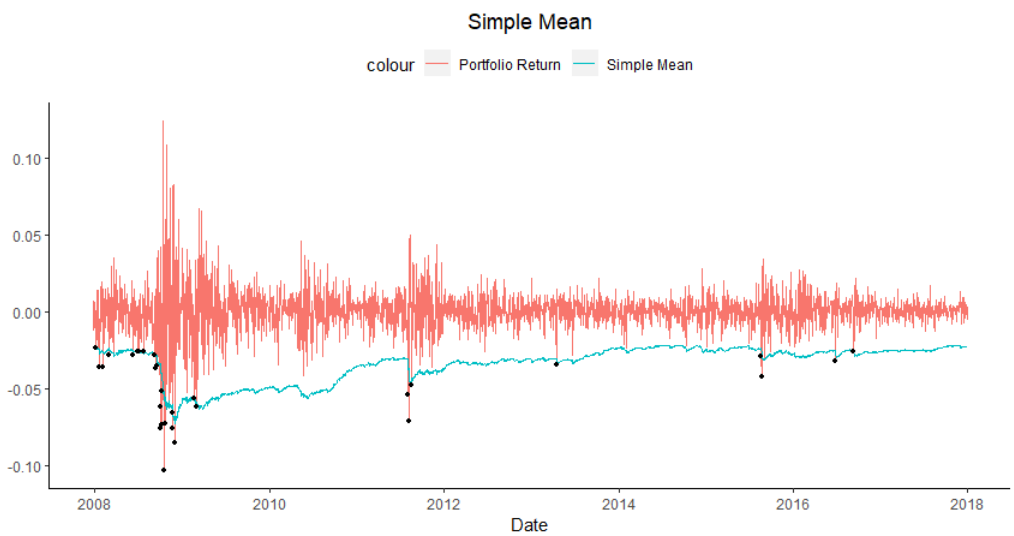

Figure 5: Simple Mean Combination VaR Forecast

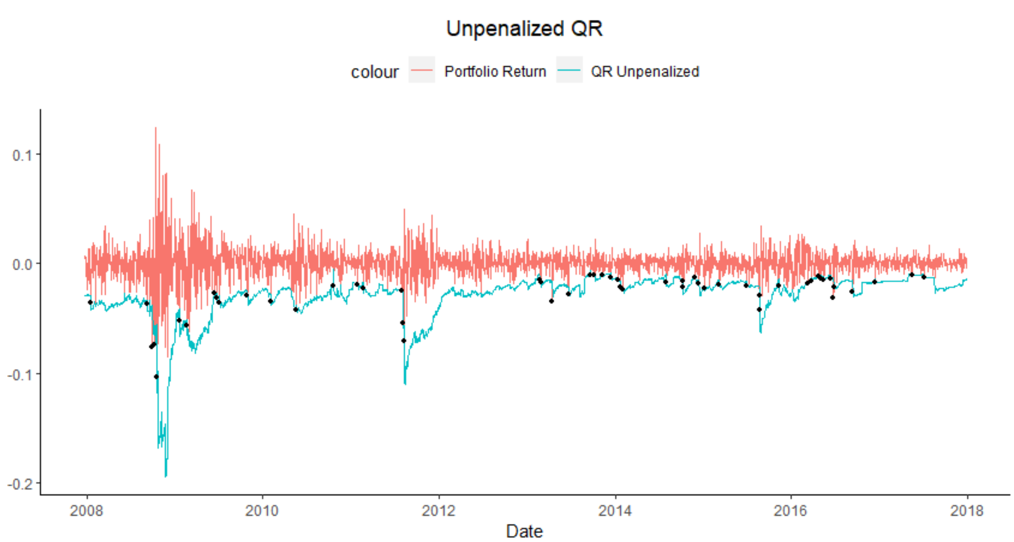

Figure 6: Unpenalized Quantile Regression VaR Forecast

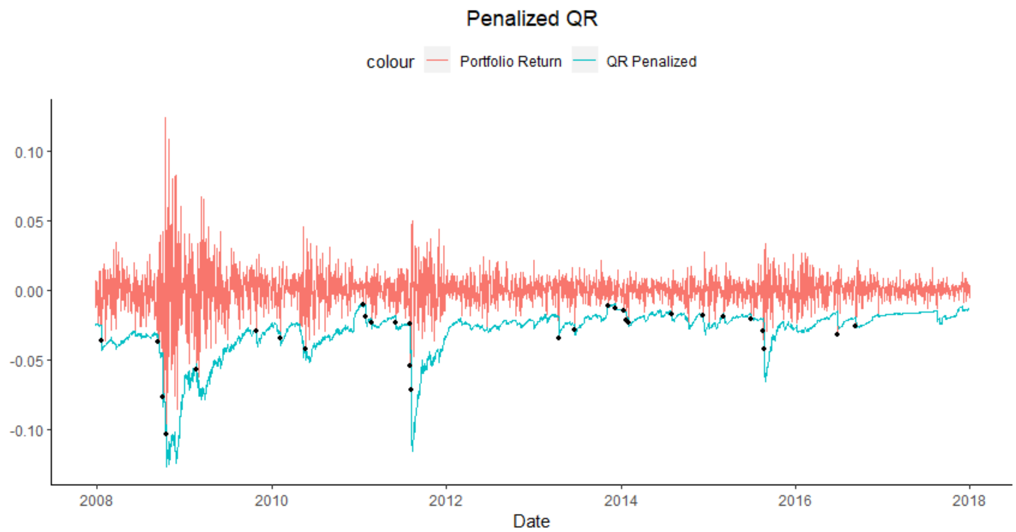

Figure 7: Penalized Quantile Regression VaR Forecast

the performance of VaR forecasts under different situation. We divide our overall period into two parts, the fluctuating period (Dec 24, 2007 - Dec 31, 2012) and the calm period (Jan 1, 2013 - Dec 29, 2017).

Table 1 - 3 show the backtesting results on overall period, fluctuant period and calm period respectively. They report the p-value of the LR_{uc}, LR_{ind} and LR_{cc} tests applied to the VaR forecasts. Models rejected at the 1% significance level are shown in bold.

First, we notice that during calm period, all standalone methods. Besides, most of the standalone VaR forecasts are rejected by unconditional coverage test, which means they underestimate the risk. However, historical simulation is the only standalone forecasts that passes unconditional coverage test. As for conditional coverage test, all standalone methods fail to provide valid VaR forecasts during financial crisis period. But, historical simulation and GARCH(1, 1) methods perform well over calm period period and overall period.

For combination methods, simple mean and penalized QR methods both perform well in terms of unconditional and conditional coverage. And penalized QR method produces less exceptions during fluctuant period and overall

Table 1: Comparison of Backtesting Results, Overall Period (Dec 24, 2007 - Dec 29, 2017)

	Backtesting overall period			
	Exceptions	LR_{uc}	LR_{ind}	LR_{cc}
Normal	62	**0.000**	**0.004**	**0.000**
HS	32	0.193	**0.007**	0.011
EVT	40	**0.006**	**0.003**	**0.000**
GARCH(1, 1)	41	**0.004**	0.698	0.014
Simple Mean	33	0.138	0.453	0.251
Unpenalized QR	53	**0.000**	0.438	**0.000**
Penalized QR	30	0.354	0.370	0.436

period. We also also find unpenalized QR method fails to provide valid VaR forecasts. The poor performance of unpenalized QR method is due to the multicollinearity problem discussed by Bayer (2018), which causes unpenalized quantile regression to generate unstable combination weights.

6. Summary

In this paper, we try to combine VaR forecasts using elastic net penalized quantile regression and to test its performance with other standalone and combination methods. Elastic net penalized quantile regression improves unpenalized quantile regression in forecasts combination with lasso and ridge penalty, which solve the multicollinearity problem and perform variable selection. From the backtesting, we find that (1) Penalized QR method does produce a very good and valid VaR forecast in different situations; (2) Unpenalized QR method fails to provide valid VaR forecasts due to the multicollinearity; (3) Simple mean method performs unexpectedly well in conditional and unconditional coverage tests; (4) Most standalone VaR forecasts underestimate risks, except historical simulation.

Table 2: Comparison of Backtesting Results, Fluctuant Period (Dec 24, 2007 - Dec 31, 2012)

	Flactuant Period			
	Exceptions	LR_{uc}	LR_{ind}	LR_{cc}
Normal	41	**0.000**	0.050	**0.000**
HS	22	0.017	0.057	**0.009**
EVT	26	**0.001**	0.112	**0.001**
GARCH(1, 1)	29	**0.000**	0.243	**0.000**
Simple Mean	20	0.055	0.423	0.115
Unpenalized QR	19	0.094	0.446	0.185
Penalized QR	15	0.517	0.548	0.677

Table 3: Comparison of Backtesting Results, Calm Period (Jan 1, 2013 - Dec 29, 2017)

	Calm Period			
	Exceptions	LR_{uc}	LR_{ind}	LR_{cc}
Normal	21	0.030	0.047	0.013
HS	10	0.447	0.065	0.137
EVT	14	0.695	**0.008**	0.027
GARCH(1, 1)	12	0.866	0.100	0.255
Simple Mean	13	0.908	0.121	0.298
Unpenalized QR	34	**0.000**	0.312	**0.000**
Penalized QR	15	0.508	0.169	0.311

7. References

[1] Abad, P., Benito, S., 2013. A detailed comparison of value at risk estimates. Math. Comput. Simul. 94, 258276.

[2] Bayer, Sebastian. 2018. Combining Value-at-Risk forecasts using penalized quantile regressions. Econometrics and Statistics 8: 5677.

[3] Berkowitz, J., Christoffersen, P., Pelletier, D., 2011. Evaluating value-at-risk models with desk-level data. Manage. Sci. 57 (12), 22132227.

[4] Giacomini, R., Komunjer, I., 2005. Evaluation and combination of conditional quantile forecasts. J. Bus. Econ. Stat. 23 (4), 416431.

[5] Halbleib, R., Pohlmeier, W., 2012. Improving the value at risk forecasts: theory and evidence from the financial crisis. J. Econ. Dyn. Control 36 (8), 12121228.

[6] Hoerl, A.E., Kennard, R.W., 1970a. Ridge regression: applications to nonorthogonal problems. Technometrics 12 (1), 6982.

[7] Hoerl, A.E., Kennard, R.W., 1970b. Ridge regression: biased estimation for nonorthogonal problems. Technometrics 12 (1), 5567.

[8] Hull, John C. (2018): "Risk Management and Financial Institutions", 5th edn

[9] Jeon, J., Taylor, J.W., 2013. Using CAViaR models with implied volatility for value-at-risk estimation. J. Forecasting 32 (1), 6274.

[10] Jorion, P. (2006): "Value at Risk" McGraw-Hill, 3rd edn

[11] Koenker, R., Basset, G., 1978. Regression quantiles. Econometrica 46, 3350.

[12] Kuester, K., Mittnik, S., Paolella, M.S., 2006. Value-at-risk prediction: a comparison of alternative strategies. Journal of Financial Econometrics 4 (1), 5389.

[13] McAleer, M., Jimenez-Martin, J.-A., Teodosio, P.-A., 2013a. GFC-robust risk management strategies under the Basel Accord. Int. Rev. Econ. Finance 27, 97111.

[14] Nieto, M.R., Ruiz, E., 2016. Frontiers in VaR forecasting and backtesting. Int. J. Forecasting 32 (2), 475501.

[15] Timmermann, A., 2006. Forecast combinations. In: Elliott, G., Granger, C.W., Timmermann, A. (Eds.), Handbook of Economic Forecasting, Vol. 1. Elsevier, pp. 135196.

8. Appendix

```r
library(extRemes)
library(PerformanceAnalytics)
library(quantmod)
library(ggplot2)
library(reshape)
library(rugarch)
library(dplyr)
library(quantreg)
library(readxl)
library(hqreg)
library(zoo)

p = 0.99
start = as.Date('2004-1-1')
end = as.Date('2017-12-31')

# Read stock names from file Equity_List.csv
# The 30 stocks are randomly chosen from S&P500
# Portfolio is formed by equally weighted among 30 stocks
tickers <- read.csv('Equity_List.csv', header = FALSE)
stock_data <- data.frame(lapply(tickers$V1, function(x) Ad(getSymbols.yahoo(
    x, from = start, to = end, auto.assign = FALSE))))
stock_rets <- na.omit(ROC(stock_data, type = "discrete"))
port_ret <- apply(stock_rets, 1, mean)
port <- data.frame(date = as.Date(row.names(stock_rets)), port_ret, row.
    names = NULL)

# Calculate 1-day VaR forecast with rolling window of 500 days
# Historical Simulation
HS <- function(mydata){
  res <- mydata[order(mydata)][5]
  return(res)
}

HS_VaR <- na.omit(lag(rollapply(port$port_ret, 500, HS)))

# Parametric VaR
# Normal distribution is used
cal_var <- function(x){
  return(-1*qnorm(p, mean(x), sd(x)))
}

para_VaR <- na.omit(lag(rollapply(port$port_ret, 500, cal_var)))

# Garch VaR
# This block might take 5+ minutes to run
garch11.spec <- ugarchspec(mean.model = list(armaOrder=c(0,0)),
                           distribution.model = 'norm')

garchmodel <- function(x){
  garch11.fit <- ugarchfit(data = x,
                           spec = garch11.spec,
                           solver.control = list(tol = 1e-4, delta=1e-9))

  forecast <- ugarchforecast(garch11.fit)
```

```r

  std <- forecast@forecast$sigmaFor[1]

  return(-1*qnorm(p, mean(x), std))
}

garch_var <- NULL
for(i in 500:length(port$port_ret)){
  garch_var <- rbind(garch_var, garchmodel(port$port_ret[1:i]))
}
garch_var <- na.omit(lag(garch_var))

# EVT VaR
EVT_model <- function(mydata){
  ret <- -1 * mydata
  thresh <- quantile(ret, p)

  # Log likelihood fucntion
  likelihood <- function(x){
    mu <- ret[ret > thresh] - thresh
    logL <- -1 * sum(log(((1+(x[2]*mu/x[1]))^(-1/x[2]-1))/x[1]))
    return(logL)
  }

  # Optimization process
  c <- c(0.1,0.1)   # Start point
  coef <- optim(c,likelihood)
  # Optimization results
  scale = coef$par[1]
  shape = coef$par[2]

  # Calculate VaR
  Nu <- length(ret[ret > thresh])
  N <- length(ret)
  EVT <- -1 * ( thresh + ( ((1-p)*N/Nu)^(-1*shape)-1 ) * scale / shape)

  return(EVT)
}

EVT_VaR <- na.omit(lag(rollapply(port$port_ret, 500, EVT_model)))

# Combine the 3 VaR estimates together with portfolio returns
all_VaR <- data.frame(port$port_ret[501:length(port$port_ret)], HS_VaR, para_VaR, EVT_VaR, garch_var)
names(all_VaR) <- c('return', 'HS', 'para', 'evt', 'garch')

# Estimate qr_VaR using quantile regression
# Unpenalized quantile regression
QRmodel <- function(mydata){
  mydata <- data.frame(mydata)
  col <- ncol(mydata)
  row <- nrow(mydata)

  X <- mydata[1:row-1, 1:4]
  y <- mydata[1:row-1, 5]

  fit <- rq(y~X$HS+X$para+X$evt+X$garch, tau = 1-p, data = mydata)
```

```
  b0 <- fit$coeff[1]
  b1 <- fit$coeff[2]
  b2 <- fit$coeff[3]
  b3 <- fit$coeff[4]
  b4 <- fit$coeff[5]

  res <- b0 + b1*mydata[row,1] + b2*mydata[row,2] + b3*mydata[row,3] + b4*
      mydata[row,4]

  return(res)
}

qr_VaR <- rollapply(all_VaR, 501, QRmodel, by.column = FALSE)

# Elastic Net Penalized quantile regression
QRmodel_Penalized <- function(mydata){
  col <- ncol(mydata)
  row <- nrow(mydata)

  X <- mydata[1:row-1, 1:4]
  y <- mydata[1:row-1, 5]

  fit <- hqreg_raw(X,y, tau = 1-p, method = 'quantile')
  b0 <- fit$beta[1,100]
  b1 <- fit$beta[2,100]
  b2 <- fit$beta[3,100]
  b3 <- fit$beta[4,100]
  b4 <- fit$beta[5,100]

  res <- b0 + b1*mydata[row,1] + b2*mydata[row,2] + b3*mydata[row,3] + b4*
      mydata[row,4]

  return(res)
}

qr_VaR_penalized <- rollapply(all_VaR, 501, QRmodel_Penalized, by.column =
    FALSE)

# Get all data together
VaR_data <- data.frame(port$date[501:length(port$date)], all_VaR)
VaR_data <- data.frame(VaR_data[501:length(VaR_data$para), ], qr_VaR, qr_VaR
    _penalized)
names(VaR_data) <- c( 'Date', 'Return', 'HS', 'Para', 'EVT', 'Garch', 'QR',
    'QR_P')
VaR_data <- mutate(VaR_data, Mean=(VaR_data$HS+VaR_data$Para+VaR_data$EVT+
    VaR_data$Garch)/4)

# Backtesting
# We will test forecasts performance over 3 periods:
# 1. Overall Period (Dec 24, 2007 - Dec 29, 2017)
# 2. Fluctuant Period (Dec 24, 2007 - Dec 31, 2012)
# 3. Calm Period (Jan 1, 2013 - Dec 29, 2017)
# Eceptions are defined that the return is less than VaR forecasts
Exceptions <- data.frame(cbind(VaR_data$Return < VaR_data$Para,
                               VaR_data$Return < VaR_data$HS,
                               VaR_data$Return < VaR_data$EVT,
                               VaR_data$Return < VaR_data$Garch,
```

```
                                    VaR_data$Return < VaR_data$Mean,
                                    VaR_data$Return < VaR_data$QR,
                                    VaR_data$Return < VaR_data$QR_P))
Exceptions_F <- data.frame(cbind(VaR_data$Return[1:1264] < VaR_data$Para[1:1264],
                                      VaR_data$Return[1:1264] < VaR_data$HS[1:1264],
                                      VaR_data$Return[1:1264] < VaR_data$EVT[1:1264],
                                      VaR_data$Return[1:1264] < VaR_data$Garch[1:1264],
                                      VaR_data$Return[1:1264] < VaR_data$Mean[1:1264],
                                      VaR_data$Return[1:1264] < VaR_data$QR[1:1264],
                                      VaR_data$Return[1:1264] < VaR_data$QR_P[1:1264]))
Exceptions_C <- data.frame(cbind(VaR_data$Return[1265:2523] < VaR_data$Para[1265:2523],
                                      VaR_data$Return[1265:2523] < VaR_data$HS[1265:2523],
                                      VaR_data$Return[1265:2523] < VaR_data$EVT[1265:2523],
                                      VaR_data$Return[1265:2523] < VaR_data$Garch[1265:2523],
                                      VaR_data$Return[1265:2523] < VaR_data$Mean[1265:2523],
                                      VaR_data$Return[1265:2523] < VaR_data$QR[1265:2523],
                                      VaR_data$Return[1265:2523] < VaR_data$QR_P[1265:2523]))
colnames(Exceptions) <- c('Normal', 'HS', 'EVT', 'GARCH', 'Mean', 'QR', 'QR_P')
colnames(Exceptions_F) <- c('Normal', 'HS', 'EVT', 'GARCH', 'Mean', 'QR', 'QR_P')
colnames(Exceptions_C) <- c('Normal', 'HS', 'EVT', 'GARCH', 'Mean', 'QR', 'QR_P')
Backtest <- data.frame(apply(Exceptions, 2, sum))
Backtest_F <- data.frame(apply(Exceptions_F, 2, sum))
Backtest_C <- data.frame(apply(Exceptions_C, 2, sum))

# Unconditional Coverage
UC_test <- function(mydata, arg){
  T <- arg
  N <- mydata
  P <- N / T
  Q <- 1 - P
  q <- 1 - p
  LR.ration <- log(((Q^(T-N)*P^N)/(p^(T-N)*q^N))^2)
  return(LR.ration)
}

LR_UC <- apply(Backtest, 2, UC_test, arg = 2523)
UC <- (round((1 - pchisq(LR_UC, df = 1)), digits = 3))

LR_UC_F <- apply(Backtest_F, 2, UC_test, arg = 1264)
UC_F <- (round((1 - pchisq(LR_UC_F, df = 1)), digits = 3))
```

```r
LR_UC_C <- apply(Backtest_C, 2, UC_test, arg = 2523-1264)
UC_C <- (round((1 - pchisq(LR_UC_C, df = 1)), digits = 3))

# Conditional Coverage
# Test independence
# Overall Period
temp <- apply(apply(apply(Exceptions, 2, as.numeric), 2, as.character), 2,
    paste, collapse = '')

T_11 <- NULL
for (i in temp) {
  z <- sapply(sapply(strsplit(i,'0'), strsplit, split = ''), length)
  T_11 <- rbind(T_11, sum(z)-length(z[z!=0]))
}
T_01 <- Backtest - T_11
T_10 <- T_01
T_00 <- 2523 - Backtest - T_10

CC_data <- data.frame(T_00, T_10, T_11, T_01)
names(CC_data) <- c('T_00', 'T_10', 'T_11', 'T_01')

CC_data <- CC_data %>%
  mutate(P = (T_01+T_11) / (T_00+T_01+T_10+T_11),
         P_01 = T_01 / (T_00 + T_01),
         P_11 = T_11 / (T_10 + T_11),
         LR.ration = -2 * log((1-P)^(T_00+T_10)*P^(T_01+T_11))
         + 2 * log((1-P_01)^T_00 * P_01^T_01 * (1-P_11)^T_10 * P_11^T_11))
LR_IND <- CC_data$LR.ration
IND <- (round((1 - pchisq(LR_IND, df = 1)), digits = 3))

#Fluctuant Period
temp <- apply(apply(apply(Exceptions_F, 2, as.numeric), 2, as.character), 2,
     paste, collapse = '')

T_11_F <- NULL
for (i in temp) {
  z <- sapply(sapply(strsplit(i,'0'), strsplit, split = ''), length)
  T_11_F <- rbind(T_11_F, sum(z)-length(z[z!=0]))
}
T_01_F <- Backtest_F - T_11_F
T_10_F <- T_01_F
T_00_F <- 1264 - Backtest_F - T_10_F

CC_data_F <- data.frame(T_00_F, T_10_F, T_11_F, T_01_F)
names(CC_data_F) <- c('T_00', 'T_10', 'T_11', 'T_01')

CC_data_F <- CC_data_F %>%
  mutate(P = (T_01+T_11) / (T_00+T_01+T_10+T_11),
         P_01 = T_01 / (T_00 + T_01),
         P_11 = T_11 / (T_10 + T_11),
         LR.ration = -2 * log((1-P)^(T_00+T_10)*P^(T_01+T_11))
         + 2 * log((1-P_01)^T_00 * P_01^T_01 * (1-P_11)^T_10 * P_11^T_11))
LR_IND_F <- CC_data_F$LR.ration
IND_F <- (round((1 - pchisq(LR_IND_F, df = 1)), digits = 3))

# Calm Period
```

```r
temp <- apply(apply(apply(Exceptions_C, 2, as.numeric), 2, as.character), 2,
    paste, collapse = '')

T_11_C <- NULL
for (i in temp) {
  z <- sapply(sapply(strsplit(i,'0'), strsplit, split = ''), length)
  T_11_C <- rbind(T_11_C, sum(z)-length(z[z!=0]))
}
T_01_C <- Backtest_C - T_11_C
T_10_C <- T_01_C
T_00_C <- 1259 - Backtest_C - T_10_C

CC_data_C <- data.frame(T_00_C, T_10_C, T_11_C, T_01_C)
names(CC_data_C) <- c('T_00', 'T_10', 'T_11', 'T_01')

CC_data_C <- CC_data_C %>%
  mutate(P = (T_01+T_11) / (T_00+T_01+T_10+T_11),
         P_01 = T_01 / (T_00 + T_01),
         P_11 = T_11 / (T_10 + T_11),
         LR.ration = -2 * log((1-P)^(T_00+T_10)*P^(T_01+T_11))
         + 2 * log((1-P_01)^T_00 * P_01^T_01 * (1-P_11)^T_10 * P_11^T_11))
LR_IND_C <- CC_data_C$LR.ration
IND_C <- (round((1 - pchisq(LR_IND_C, df = 1)), digits = 3))

# LR_CC
LR_CC <- LR_UC + LR_IND
CC <- round(1 - pchisq(LR_CC, df = 2), digits = 3)

LR_CC_F <- LR_UC_F + LR_IND_F
CC_F <- round(1 - pchisq(LR_CC_F, df = 2), digits = 3)

LR_CC_C <- LR_UC_C + LR_IND_C
CC_C <- round(1 - pchisq(LR_CC_C, df = 2), digits = 3)

# Plot
f1 <- ggplot(VaR_data, aes(x = Date, y = Return, col = 'Portfolio Return'))
   +
  geom_line() +
  geom_line(aes(y = Para, col = 'Normal VaR')) +
  geom_point(aes(y=Return, col='Exceptions'),
             color = ifelse(VaR_data$Return < VaR_data$Para,'black','blue'),
             size = ifelse(VaR_data$Return < VaR_data$Para,1,-1)) +
  labs(title = "Normal Model") +
  theme(plot.title=element_text(hjust=0.5),
        panel.grid.major =element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = 'top',
        axis.title.y = element_blank())

f2 <- ggplot(VaR_data, aes(x = Date, y = Return, col = 'Portfolio Return'))
   +
  geom_line() +
  geom_line(aes(y = HS, col = 'HS VaR')) +
  geom_point(aes(y=Return, col='Exceptions'),
             color = ifelse(VaR_data$Return < VaR_data$HS,'black','blue'),
```

```
                size = ifelse(VaR_data$Return < VaR_data$HS,1,-1)) +
  labs(title = "Historical Simulation") +
  theme(plot.title=element_text(hjust=0.5),
        panel.grid.major =element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = 'top',
        axis.title.y = element_blank())

f3 <- ggplot(VaR_data, aes(x = Date, y = Return, col = 'Portfolio Return')) +
  geom_line() +
  geom_line(aes(y = EVT, col = 'EVT VaR')) +
  geom_point(aes(y=Return, col='Exceptions'),
                color = ifelse(VaR_data$Return < VaR_data$EVT,'black','blue'),
                size = ifelse(VaR_data$Return < VaR_data$EVT,1,-1)) +
  labs(title = "Extreme Value Theory") +
  theme(plot.title=element_text(hjust=0.5),
        panel.grid.major =element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = 'top',
        axis.title.y = element_blank())

f4 <- ggplot(VaR_data, aes(x = Date, y = Return, col = 'Portfolio Return')) +
  geom_line() +
  geom_line(aes(y = Garch, col = 'GARCH VaR')) +
  geom_point(aes(y=Return, col='Exceptions'),
                color = ifelse(VaR_data$Return < VaR_data$Garch,'black','blue'),
                size = ifelse(VaR_data$Return < VaR_data$Garch,1,-1)) +
  labs(title = "GARCH(1, 1) Model") +
  theme(plot.title=element_text(hjust=0.5),
        panel.grid.major =element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = 'top',
        axis.title.y = element_blank())

f5 <- ggplot(VaR_data, aes(x = Date, y = Return, col = 'Portfolio Return')) +
  geom_line() +
  geom_line(aes(y = Mean, col = 'Simple Mean')) +
  geom_point(aes(y=Return, col='Exceptions'),
                color = ifelse(VaR_data$Return < VaR_data$Mean,'black','blue'),
                size = ifelse(VaR_data$Return < VaR_data$Mean,1,-1)) +
  labs(title = "Simple Mean") +
  theme(plot.title=element_text(hjust=0.5),
        panel.grid.major =element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = 'top',
```

```
        axis.title.y = element_blank())

f6 <- ggplot(VaR_data, aes(x = Date, y = Return, col = 'Portfolio Return')) +
  geom_line() +
  geom_line(aes(y = QR, col = 'QR Unpenalized')) +
  geom_point(aes(y=Return, col='Exceptions'),
             color = ifelse(VaR_data$Return < VaR_data$QR,'black','blue'),
             size = ifelse(VaR_data$Return < VaR_data$QR,1,-1)) +
  labs(title = "Unpenalized QR") +
  theme(plot.title=element_text(hjust=0.5),
        panel.grid.major =element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = 'top',
        axis.title.y = element_blank())

f7 <- ggplot(VaR_data, aes(x = Date, y = Return, col = 'Portfolio Return')) +
  geom_line() +
  geom_line(aes(y = QR_P, col = 'QR Penalized')) +
  geom_point(aes(y=Return, col='Exceptions'),
             color = ifelse(VaR_data$Return < VaR_data$QR_P,'black','blue'),
             size = ifelse(VaR_data$Return < VaR_data$QR_P,1,-1)) +
  labs(title = "Penalized QR") +
  theme(plot.title=element_text(hjust=0.5),
        panel.grid.major =element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank(),
        axis.line = element_line(colour = "black"),
        legend.position = 'top',
        axis.title.y = element_blank())
```