BUSN 41201 Midterm

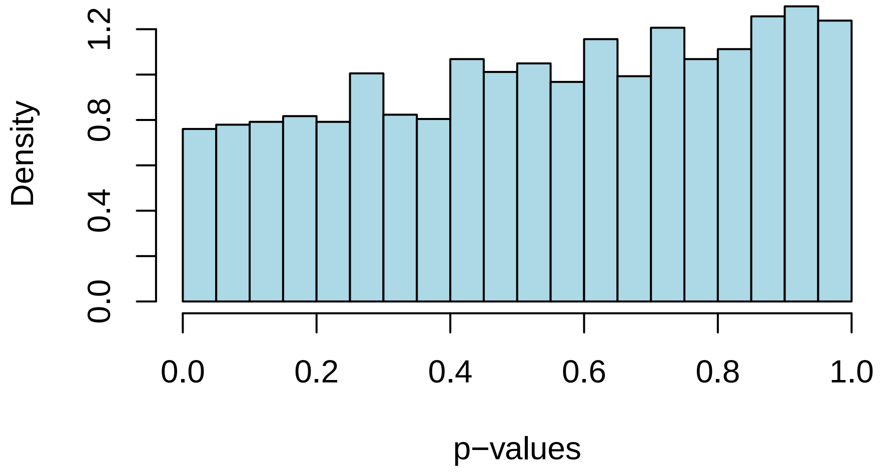
05/01/2022

Q1.1

****** Prices**

```
o <- order(mrgpvals_p)
hist(mrgpvals_p[o], breaks=20, col="lightblue", freq = FALSE,main = 'Figure 1. Histgram of p-values (pr
```

Figure 1. Histgram of p−values (price)



For the price, there isn't clear spike in the plot, although the density increases a bit could p = 0.4 and p=0.8. Thus, we can't claim that it's enough signal to predict price.

****** Repeat for volatility**

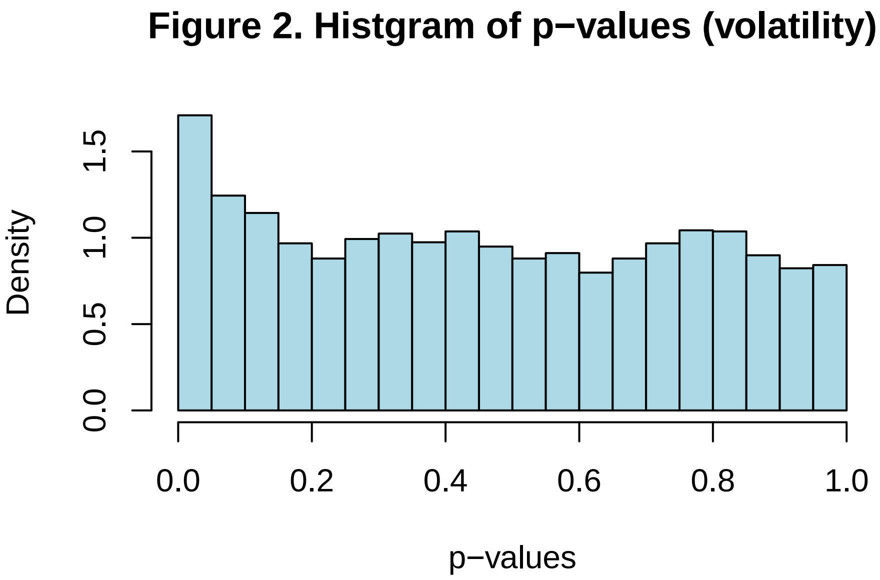
```
Outcome<-V

clusterExport(cl,"Outcome")

# run the regressions in parallel

mrgpvals_v <- unlist(parLapply(cl,P,margreg))
```

```
# plot
hist(as.numeric(mrgpvals_v), breaks=20, col="lightblue", freq = FALSE, main='Figure 2. Histgram of p-val
```



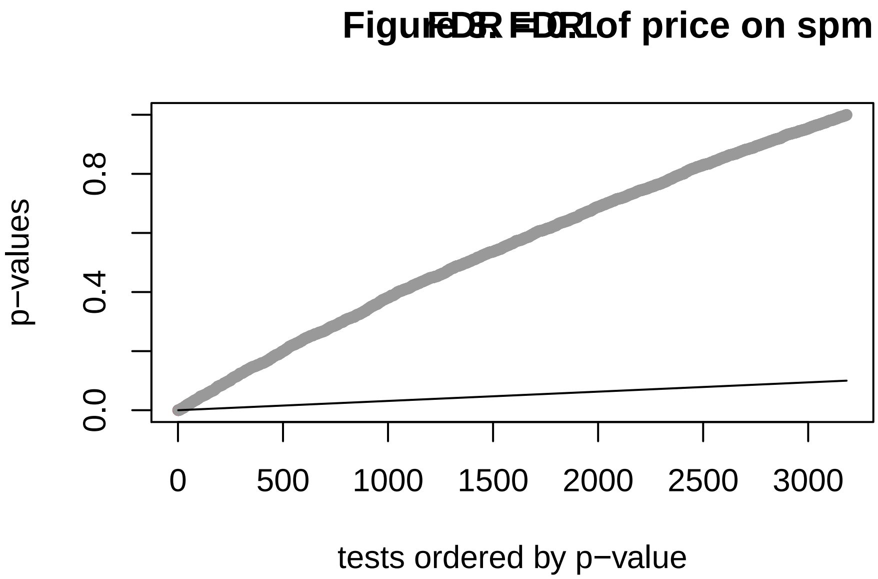
However, for the volatility, we can observe a clear spike near zero, which indicates that the p-value distribution deviates from a uniform distribution. This is an indication for signal. Therefore, the spike near zero helps us know that it's highly likely to be a signal to predict.

Q1.2

**** **Prices**

```
# cutoff
source("fdr.R")

cutoff_p <- fdr_cut(mrgpvals_p,0.1,TRUE)
title("Figure 3. FDR of price on spm", adj = 1)
```



```
print(cutoff_p)
```

```
## [1] 1.026222e-05
```

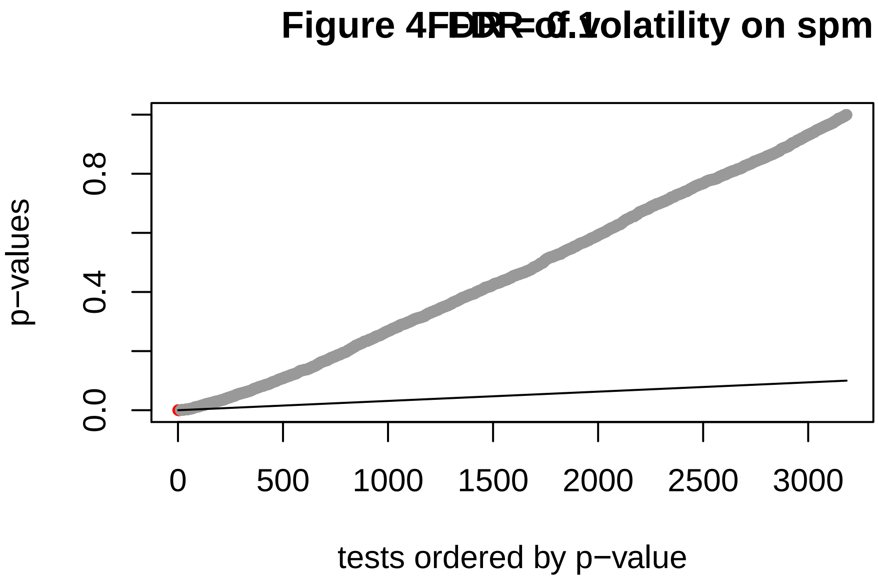
```
# If we did classical statistical testing at alpha level 0.1, we would have 245 discoveries for prices
kable(table(mrgpvals_p<0.1))
```

Var1	Freq
FALSE	2938
TRUE	245

The alpha value is 1.026222e-05 for price, and 245 words are significant.

**** Repeat for volatility

```
# Repeat for volatility
cutoff_v <- fdr_cut(mrgpvals_v,0.1,TRUE)
title("Figure 4. FDR of volatility on spm", adj = 1)
```



```
print(cutoff_v)
```

```
## [1] 0.0003571024
```

```
# If we did classical statistical testing at alpha level 0.1, we would have 470 discoveries for prices
kable(table(mrgpvals_v<0.1))
```

Var1	Freq
FALSE	2713
TRUE	470

The alpha value is 0.0003571024 for volatility, and 470 words are significant.

- Advantage of FDR: It enables us to analyze big data without committing errors which would easily occur with traditional t test. It allows us to parse big data and do that in parallel. We can estimate the effect of each word on price and volatility separately with FDR.
- Disadvantage: However, with FDR, we cannot decide the exact context of the words which occur in the final analysis result. A lot of words should occur at the same time or have different meaning from its literal meaning when used in News headlines. Since we're estimating the effect of each word separately, it's likely we may misunderstand those words in the sentence with FDR method. Therefore, multilinear problems will occur and the independence of FDR p-value results would be affected.

Q1.3

```
significant<- mrgpvals_v<=cutoff_v

# We know the number of discovery is 20
q <- 0.01

q * 20 # this is the expected number of false discoveries
```

```
## [1] 0.2
```

```r
# Names of the most significant words
o<-order(mrgpvals_v, decreasing=FALSE)

top20<-o[1:20]

top_pvals_20<-mrgpvals_v[top20]

kable(names(top_pvals_20))
```

x
tunisia
georgia
terror
fusion
ossetia
iceland
christian
republican
been
zika
barrel
vatican
arrest
august
hebdo
juarez
payout
when
cup
mali

```r
# Test for independence
dj <- dj %>%
  mutate(Volatility = log(High - Low))
reg <- glm(Volatility ~ lag(Volatility), data = dj)
summary(reg)
```

```
##
## Call:
## glm(formula = Volatility ~ lag(Volatility), data = dj)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.33128  -0.33003  -0.00561   0.31995   1.51361
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2.42010    0.09629   25.13   <2e-16 ***
## lag(Volatility)  0.51250    0.01928   26.58   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

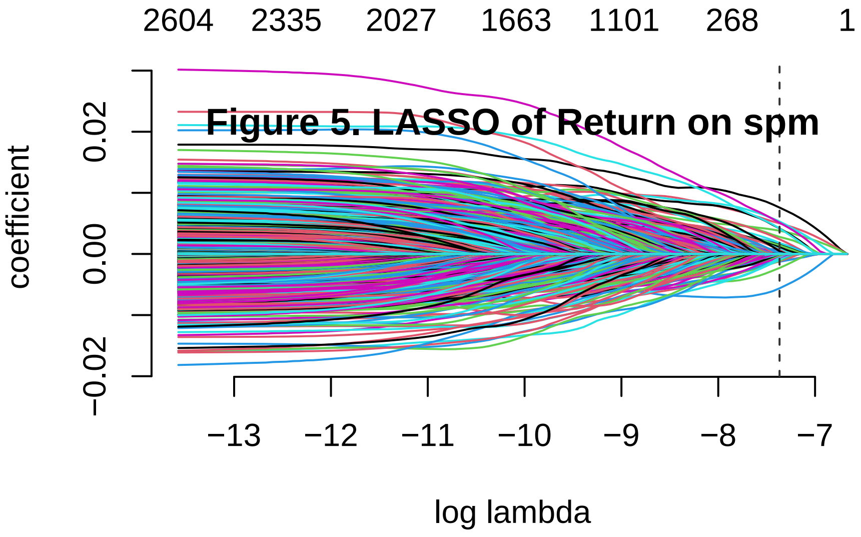
```
##
## (Dispersion parameter for gaussian family taken to be 0.226516)
##
##     Null deviance: 609.88  on 1987  degrees of freedom
## Residual deviance: 449.86  on 1986  degrees of freedom
##   (1 observation deleted due to missingness)
## AIC: 2693.6
##
## Number of Fisher Scoring iterations: 2
```

- If we only mark the 20 smallest p-values as significant, we 'll we only observe 0.2 discoveries to be false. However, we can only count word with integer, it means it's highly likely that we won't find any false discovery.

- p values are not independent in this case. We can see that with the regression above, the p-value is extremely small, which means there is some kind of correlation between the volatility today and volatility yesterday. In other words, the x variable in different tests are correlated with each other, thus, the p-value we get from each separate tests won't be independent any more.

Q2.1

```
lasso1<- gamlr(spm, y=R, lambda.min.ratio=1e-3)
plot(lasso1)
title("Figure 5. LASSO of Return on spm", line = -2)
```



```
# Choose the best lambda with AICc
# AICc selected coef
beta_return <- coef(lasso1)

exp(log(lasso1$lambda[which.min(AICc(lasso1))]))
```

```
##        seg11
## 0.0006318906
```

```
sum(beta_return !=0)
```

```
## [1] 48
```

```
# in-sample R2
dev <- lasso1$deviance[which.min(AICc(lasso1))] # deviance of the AICc selected model
dev0<- lasso1$deviance[1] # null deviance
1-dev/dev0
```

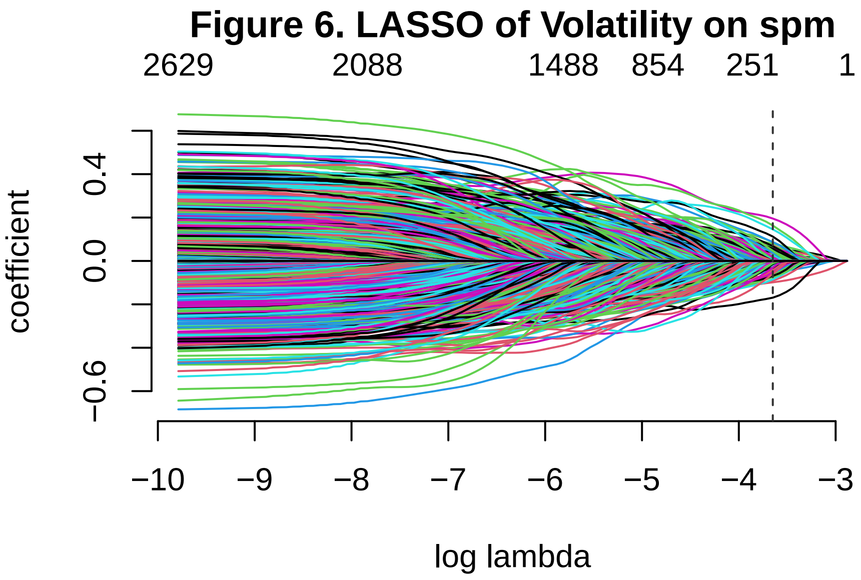
```
##      seg11
## 0.07200122
```

Based on the result from the above LASSO analysis, the best lambda occurs when the condition that log lambda equals -7.367 is met, i.e. $\lambda = 0.00063$. The in-sample R square is pretty low, which is 0.072, meaning that the goodness of fit is poor. Therefore, we fail to conclude that headlines can predict returns with solid evidence.

Q2.2

****** LASSO Analysis of volatility**

```
lasso2<- gamlr(spm, y=V, lambda.min.ratio=1e-3)
plot(lasso2)
title("Figure 6. LASSO of Volatility on spm")
```



```
# Choose the best lambda with AICc
# AICc selected coef
beta_return <- coef(lasso2)

exp(log(lasso2$lambda[which.min(AICc(lasso2))]))
```

```
##      seg12
## 0.02601636
```

```
sum(beta_return !=0)
```

```
## [1] 135
```

```
# in-sample R2
dev_v <- lasso2$deviance[which.min(AICc(lasso2))] # deviance of the AICc selected model
dev0_v<- lasso2$deviance[1] # null deviance
1-dev_v/dev0_v
```

```
##     seg12
## 0.1616116
```

If we run LASSO for volatility instead, the best lambda occurs when the condition that log lambda equals -3.649 is met, i.e., $\lambda = 0.026$. The in-sample R square is better than that of price, which is 0.162, but it's still not good enough to conclude that volatility is correlated with headline words.

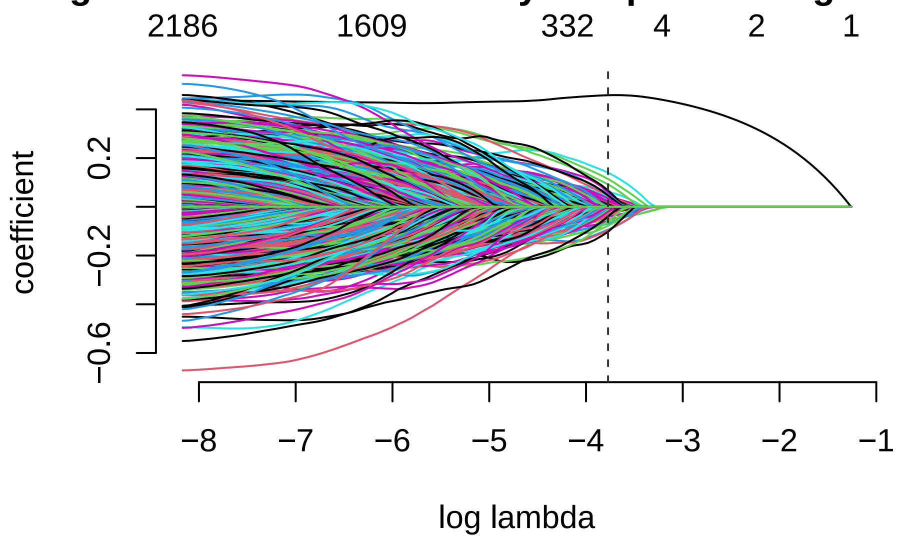
**** Add extra predictor

```
# remove the last return
Previous<-log(dj[-1,3]-dj[-1,4])
# add the previous return to the model matrix
spm2<-cbind(Previous,spm)
# the first column is the previous volatility
colnames(spm2)[1]<-"previous"

lasso3<- gamlr(spm2, y=V, lambda.min.ratio=1e-3)

plot(lasso3)
title("Figure 7. LASSO of Volatility on spm and Lag Volatility")
```

Figure 7. LASSO of Volatility on spm and Lag Volatili



```
# Choose the best lambda with AICc
# AICc selected coef
beta_return <- coef(lasso3)

exp(log(lasso3$lambda[which.min(AICc(lasso3))]))
```

```
##      seg37
## 0.02300037
```

```
# Choose 10 most strongest words in this model
beta_return <- beta_return %>%
  mutate(strong = abs(seg37)) %>%
  arrange(desc(strong)) %>%
  select(strong, words,everything())
kable(beta_return[1:10,1:2])
```

	strong	words
intercept	2.7164605	intercept
previous	0.4580944	previous
shed	0.1401080	shed
fusion	0.1141525	fusion
unleash	0.1028618	unleash
shake	0.0966033	shake
joe	0.0939041	joe
republican	0.0920512	republican
payout	0.0779229	payout
direct	0.0686633	direct

```
# in-sample R2
dev_lag <- lasso3$deviance[which.min(AICc(lasso3))] # deviance of the AICc selected model
```

```
dev0_lag<- lasso3$deviance[1] # null deviance
1-dev_lag/dev0_lag
```

```
##    seg37
## 0.331585
```

```
# Interpretation of terror coefficient
beta_return$seg37[beta_return$words =="terror"]
```

```
## [1] 0.01151404
```

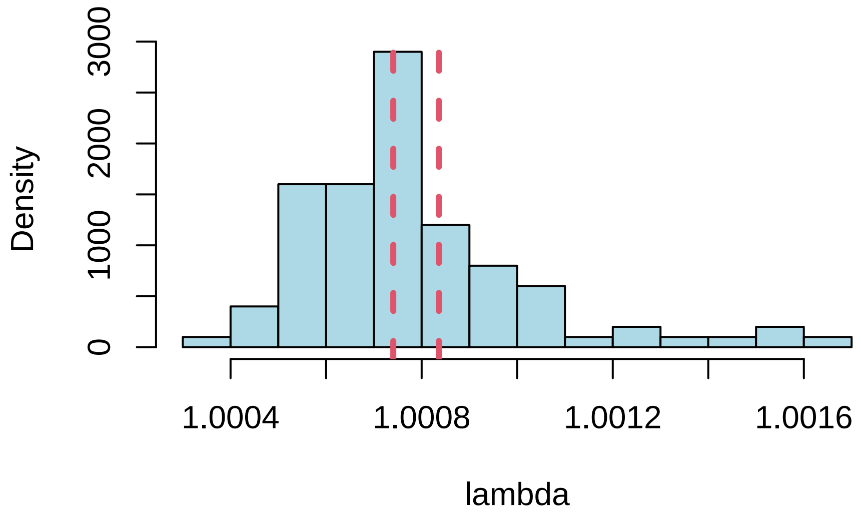
- If we run LASSO for volatility instead, the best lambda occurs when $\lambda = 0.023$. The in-sample R square is 0.33 now, which is better than before.
- My understanding is that *strong* means *significant correaltion* and it does not limited to *positive relation*, therefore, I selected the top10 words which have the largest absolute value of coefficient.
- The interpretation of the coefficient of *terror* is that each 1 more time the word *terror* occur, it will increase the volatility today by 0.012 unit, given the volatility yesterday is constant.

Q2.3

```
# Plot and calculation
results <- t.test(exp(lambda_samp),conf.level = 0.95)
low <- results$conf.int[1]
high <- results$conf.int[2]

hist(exp(lambda_samp), col="lightblue",freq = FALSE,
     xlab="lambda", main="Figure 8. Sampling Distribution of Selected lambda",
     breaks = 15)
abline(v= low, col=2, lty = 2, lwd = 3)
abline(v= high, col=2, lty = 2, lwd = 3)
```

Figure 8. Sampling Distribution of Selected lambda



```r
# Standard Deviation
sd(exp(lambda_samp))
```

```
## [1] 0.0002406706
```

```r
# 95% CI
mean(exp(lambda_samp)) + 1.96*sd(exp(lambda_samp))/10
```

```
## [1] 1.000836
```

```r
mean(exp(lambda_samp)) - 1.96*sd(exp(lambda_samp))/10
```

```
## [1] 1.000741
```

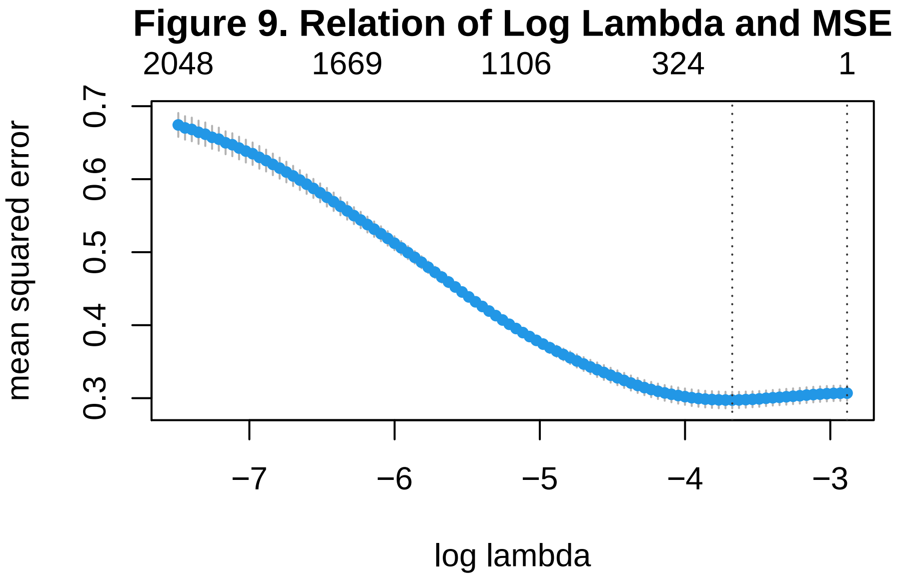
- The 95% CI for λ is (1.000741, 1.000836)
- The standard error for the selected λ is 0.000241

Q3.1

```r
# Marginal Reg
d <- Previous
marginal <- glm(V ~ d)
coef(marginal)["d"]
```

```
##         d
## 0.5119538
```

```
# Predict d from x
cv.treat <- cv.gamlr(spm,d)
dhat <- predict(cv.treat$gamlr, spm)
dhat <- drop(dhat)
plot(cv.treat)
title("Figure 9. Relation of Log Lambda and MSE")
```



```
# R square
1-min(cv.treat$cvm)/cv.treat$cvm[1] # Out of Sample
```

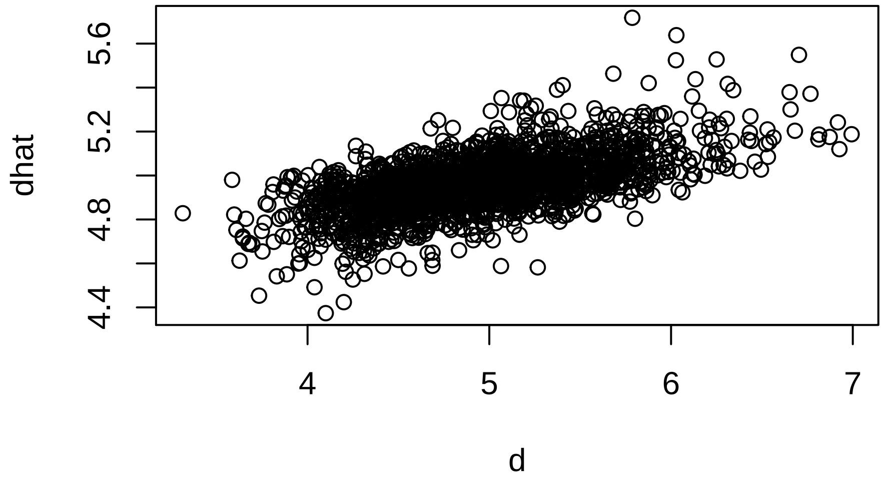
```
## [1] 0.03095948
```

```
summary(cv.treat$gamlr)[which.min(AICc(cv.treat$gamlr)),] # In sample
```

```
##
## gaussian gamlr with 3183 inputs and 100 segments.
```

```
plot(d,dhat)
title("Figure 10. Relation of lag and predicted volatility")
```

Figure 10. Relation of lag and predicted volatility

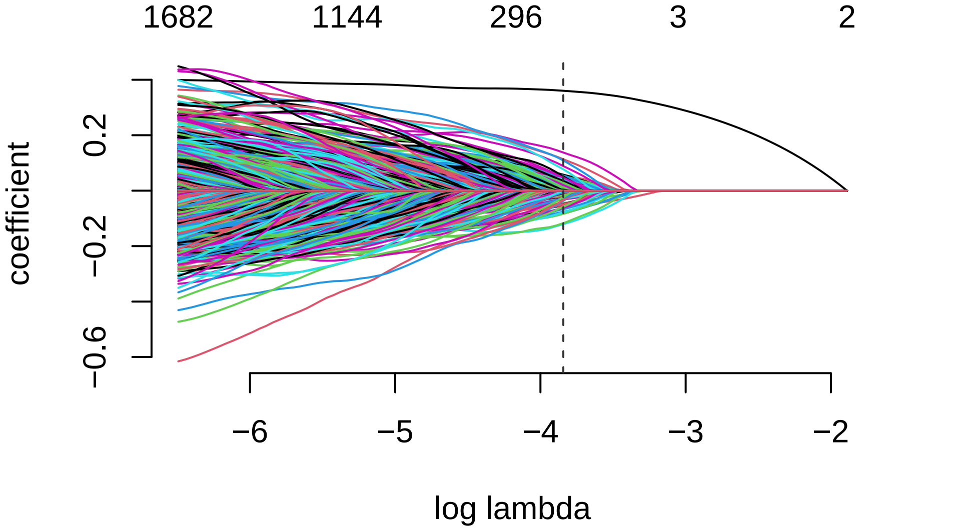


- The marginal regression of volatility on the treatment variable gives us a coefficient at 0.512, which means there is probably some correlation between V_t *and* V_{t-1}.
- We can also notice obvious positive relationship between the treatment and the predicted treatment with spm in the plot, which means the treatment and spm(control variables) is also likely to be correlated. In other words, we could expect confounding effect caused by *spm* if we simply run regression of volatility on V_{t-1}.
- The out of sample R square is 0.0176 and the in sample R square is 0.229. On the one hand, it indicates that there is some independent variation upon which we could measure treatment effect since the OOS is pretty low while we expect an ideal model to have relatively high OOS. Also, it also has some in-sample variations, meaning that treatment may kind of independent of x variables as well. On the other hand, however,the in-sample R square gives us strong signal that we need to measure the effect of treatment given x variables controlled.

Q3.2

```
# DOUBLE LASSO
causal <- gamlr(cbind(d,dhat,spm), V, free = 2)
plot(causal)
title("Figure 11. DOUBLE LASSO of volatility contrlling for spm")
```

Figure 11. DOUBLE LASSO of volatility contrlling for s



```
coef(causal)["d",]
```

```
## [1] 0.3602822
```

```
# Naive LASSO
naive <- gamlr(cbind(d,spm),V)
coef(naive)["d",]
```

```
## [1] 0.4574218
```

- The coefficient we get from DOUBLE LASSO is 0.3603. It means 1 unit increase in yesterday's Volatility will increase today's Volatility by 0.3602 units on average, given new headlines(the control varaibles) are constant.
- However, the coefficient we get from Naive LASSO is 0.457. The effect is much greater than what we get with DOUBLE LASSO. By adding controls, we get decrease in magnitude of effect. It's caused by the fact that spm has a pretty obvious confounding effect on our treatment as shown in Q3.1. Since the controlling variables are positively correlted with both the treatment and the outcome variable, adding controls will isolate the effect of treatment and decrease the magnitude of slope coefficient we get.

Q3.3

```
# Define variable name
x <- spm
y <- V
# Bootstrap
n <- 1988
B <- 100
```

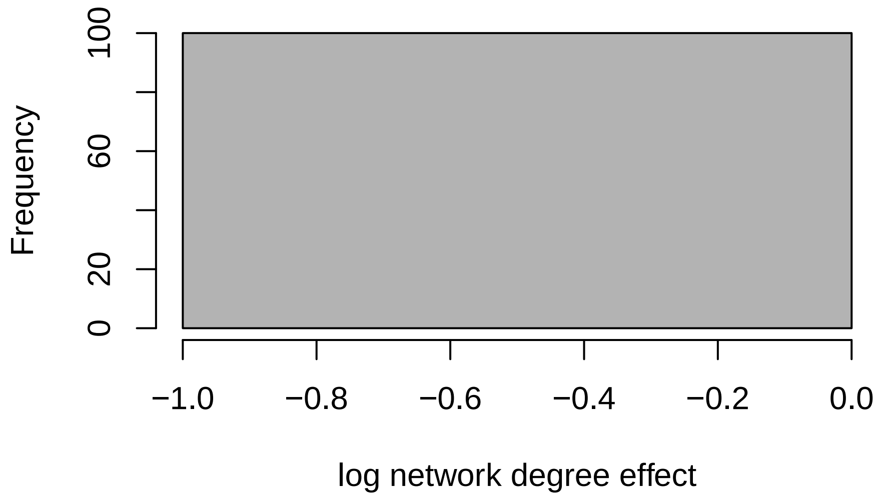
```r
cl = makeCluster(detectCores())

resamp <- as.data.frame(
  matrix(sample(1:n, B*n, replace=TRUE),ncol=B))

bootfit <- function(ib){
  require(gamlr)
  xb <- x[ib,]
  db <- d[ib]
  yb <- y[ib]
  treatb <- gamlr(xb,db)
  dhatb <- predict(treatb,xb)
  fitb <- gamlr(cbind(db,dhatb,xb),
              yb,free=2)
  return(coef(fitb)["db",])
  }

clusterExport(cl, c("x","d","y"))
parbootgamma <- unlist(parLapply(cl,resamp,bootfit))
stopCluster(cl)

hist(parbootgamma, col="grey70", xlab="log network degree effect", main="")
abline(v=coef(causal)["d",],lwd=2, col="red")
```



Based on the plot, we know that the estimate we get from full sample AICc lies in the right side of the beta distribution given different sampling. Therefore, the effect of treatment *yesterday's volatility may be overestimated. Also, we didn't take other potential relevant factors like the volume of stock exchange on that day, changes in people's trading preferences with time etc, therefore, we need to be careful to draw the conclusion that there si casual relationship between yesterday's volatility and today's volatility controlling News headlines' words.

Bonus

Although we add control variables to the data, we do not know whether the effect would vary based on year, season, region and also volume etc. The ideal setting would be to run a randomized control Trial Experiment. However, it's not feasible here. Therefore, we may analyze weather there is different effect of yesterday's volatility among different groups. There are multiple ways to do this:

Create categorical observable varaible

```
dj <- dj %>%
  separate("Date", c("year", "month", "day"), sep = "-", convert = TRUE) %>%
  select(year, month, day,Open, Volume, Adj.Close, everything())
```

Method 1: set fixed effects unpenalized

```
x_mc <- cbind(x, dj$year, dj$month, dj$day, dj$Adj.Close, dj$Open, dj$Volume)
causal_vc <- gamlr(cbind(d,dhat,x_mc),y,free=c(2,1:3189))
coef(causal_vc)["d",]
```

```
## [1] 0.4880257
```

We can set all 'fixed effects' unpenalized, and the magnitude of coefficient increased significantly to 0.488. It shows the importance of adding penalty in this case.

Method 2: add controls

```
x_mc <- cbind(x, dj$year, dj$month, dj$day, dj$Adj.Close, dj$Open, dj$Volume)
causal_mc <- gamlr(cbind(d,dhat,x_mc),y,free = 2)
coef(causal_mc)["d",]
```

```
## [1] 0.281447
```

```
x_date <- cbind(x,dj$year, dj$month, dj$day)
causal_mc_date <- gamlr(cbind(d,dhat,x_date), y, free= 2)
coef(causal_mc_date)["d",]
```

```
## [1] 0.3602822
```

```
x_stock <- cbind(x,dj$Adj.Close, dj$Open, dj$Volume)
causal_mc_stock <- gamlr(cbind(d,dhat,x_stock), y, free= 2)
coef(causal_mc_stock)["d",]
```

```
## [1] 0.2892585
```

```r
x_volume <- cbind(x,dj$Volume)
causal_mc_volume <- gamlr(cbind(d,dhat,x_volume), y, free= 2)
coef(causal_mc_volume )["d",]
```

```
## [1] 0.2939159
```

We add different control variables to the LASSO. Overall, though the magnitude of treatment coefficient varies, it provides signal that it has some correlation with the outcome.

- When we add all other potentially relevant factors in the dataset to the LASSO, we get a coefficient of 0.2814, which is smaller than the DOUBLE LASSO result we get in Q3.2. It shows that we may neglect some confounding factors which are positively correlated with both today and yesterday's volatility.
- However, adding time-related characteristics as control does not change the result too much. We get a slightly higher coefficient at 0.3603. Time period may not have confounding effect on the LASSO.
- We observe significant decline in the coefficient when adding other stock market factors as control variables. The coefficient is 0.289 now, which also cross validated our previous observation that the decline in coefficient magnitude is caused mainly by stock market relevant factors. More specifically, we can add those control variables separately, and the result shows that the *Volume* of stock trading has significant confounding effect on the treatment, pushing the coefficient decrease to 0.294. It also verified our previous assumption in Q3.3 that we may overestimate the effect of treatment.