Project2

Yiting Zhang

2/13/2022

```
google =read.csv("GOOG_data.csv")

fb =read.csv("FB_data.csv")

return =read.csv("Return.csv")

google <- google[-2450,]
```

```
ts_google <- ts(google$Adj.Close, start=c(2012,5),frequency=365)

ts_fb <- ts(fb$Adj.Close, start=c(2012,5),frequency=365)
```

j

Compare your forecast from (i) to the 12-steps ahead forecasts from ARIMA, Holt-Winters, and ETS models. Which model performs best in terms of MAPE?

k

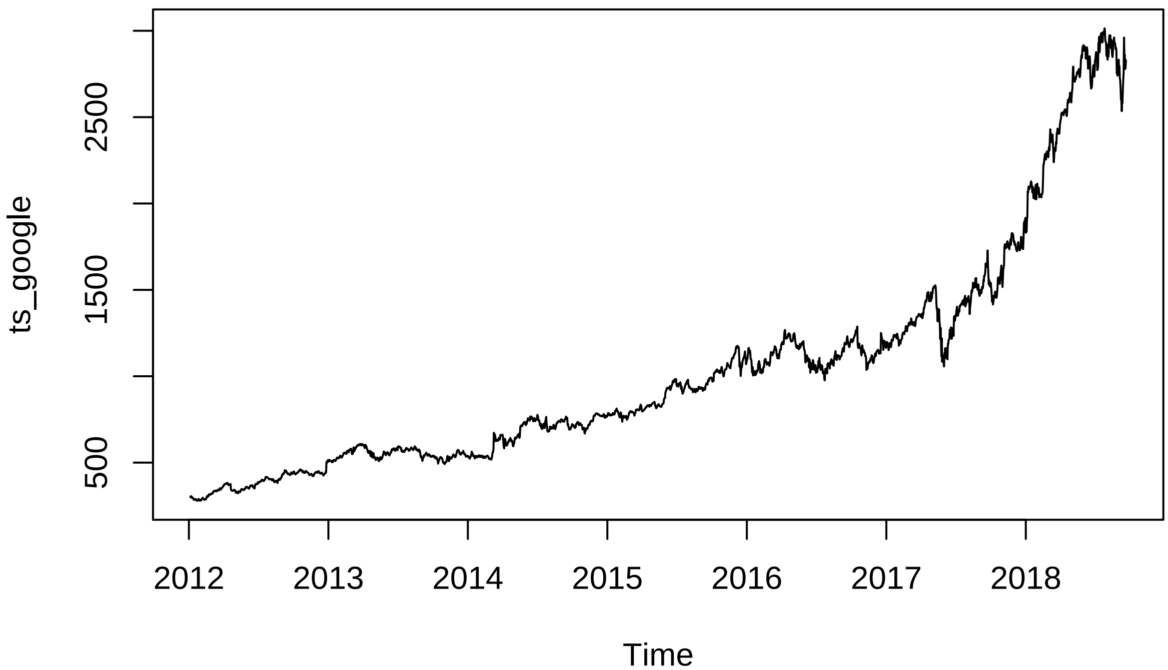
Combine the four forecasts and comment on the MAPE from this forecasts vs., the individual ones.

l

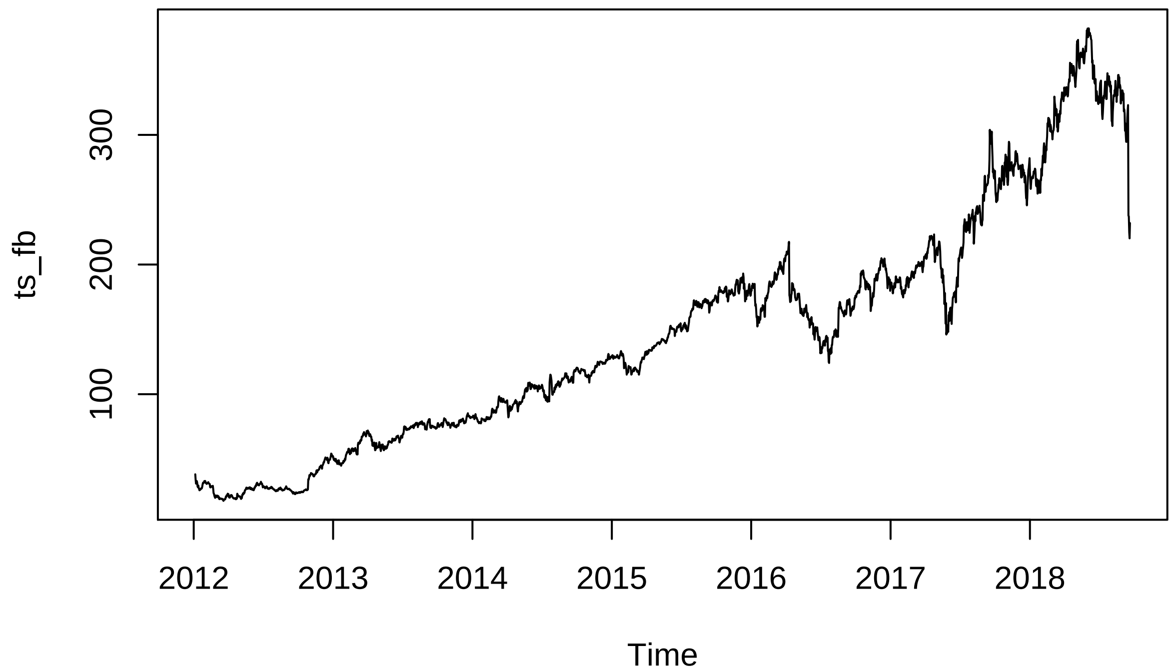
Fit an appropriate VAR model using your two variables. Make sure to show the relevant plots and discuss your results from the fit.

According to the plot, returns of goodle and facebook all show smean reverting nature.

```
plot(ts_google)
```


ts_google
500
1500
2500
2012
2013
2014
2015
2016
2017
2018
Time

```
plot(ts_fb)
```




```
library(vars)
```

```
## Loading required package: urca

## Loading required package: lmtest

##
## Attaching package: 'vars'

## The following object is masked from 'package:fable':
##
##     VAR
```

```
VARselect(data.frame(ts_google, ts_fb))
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      9      9      2      9
##
## $criteria
##                 1           2           3           4           5           6
## AIC(n)   8.276902    8.262392    8.261988    8.259220    8.255113    8.255609
## HQ(n)    8.282088    8.271035    8.274088    8.274778    8.274128    8.278081
```

```
## SC(n)     8.291168    8.286170    8.295276    8.302019    8.307424    8.317430
## FPE(n) 3931.993015 3875.353239 3873.785803 3863.078996 3847.246841 3849.154206
##                  7           8           9          10
## AIC(n)    8.247073    8.238297    8.233818    8.233919
## HQ(n)     8.273003    8.267684    8.266663    8.270221
## SC(n)     8.318406    8.319141    8.324173    8.333785
## FPE(n) 3816.441117 3783.095020 3766.189407 3766.569809
```

The results from the VAR model shows the most significance for the lower lags and lags from 5-8. We need impulse response to have further analysis.

```
model <- VAR(data.frame(ts_google, ts_fb), 9)
summary(model)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: ts_google, ts_fb
## Deterministic variables: const
## Sample size: 2440
## Log Likelihood: -16930.941
## Roots of the characteristic polynomial:
## 1.001 1.001 0.812 0.812 0.761 0.761 0.7447 0.7447 0.7416 0.7348 0.7348 0.7347 0.7347 0.7039 0.7039 0
## Call:
## VAR(y = data.frame(ts_google, ts_fb), p = 9)
##
##
## Estimation results for equation ts_google:
## ==========================================
## ts_google = ts_google.l1 + ts_fb.l1 + ts_google.l2 + ts_fb.l2 + ts_google.l3 + ts_fb.l3 + ts_google.l
##
##               Estimate Std. Error t value Pr(>|t|)
## ts_google.l1  0.920428   0.025197  36.529  < 2e-16 ***
## ts_fb.l1      0.081793   0.130138   0.629 0.529732
## ts_google.l2  0.101776   0.035937   2.832 0.004663 **
## ts_fb.l2      0.096584   0.187642   0.515 0.606794
## ts_google.l3 -0.042739   0.036080  -1.185 0.236299
## ts_fb.l3     -0.389529   0.186824  -2.085 0.037173 *
## ts_google.l4 -0.007459   0.036052  -0.207 0.836104
## ts_fb.l4     -0.023400   0.186472  -0.125 0.900147
## ts_google.l5 -0.016351   0.036565  -0.447 0.654797
## ts_fb.l5      0.152745   0.198437   0.770 0.441529
## ts_google.l6 -0.006042   0.036573  -0.165 0.868801
## ts_fb.l6      0.069624   0.203458   0.342 0.732227
## ts_google.l7  0.140932   0.036915   3.818 0.000138 ***
## ts_fb.l7     -0.078059   0.204477  -0.382 0.702682
## ts_google.l8 -0.123422   0.037180  -3.320 0.000915 ***
## ts_fb.l8     -0.235612   0.205048  -1.149 0.250645
## ts_google.l9  0.030224   0.026894   1.124 0.261204
## ts_fb.l9      0.353824   0.148202   2.387 0.017043 *
## const        -0.089147   0.769173  -0.116 0.907741
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

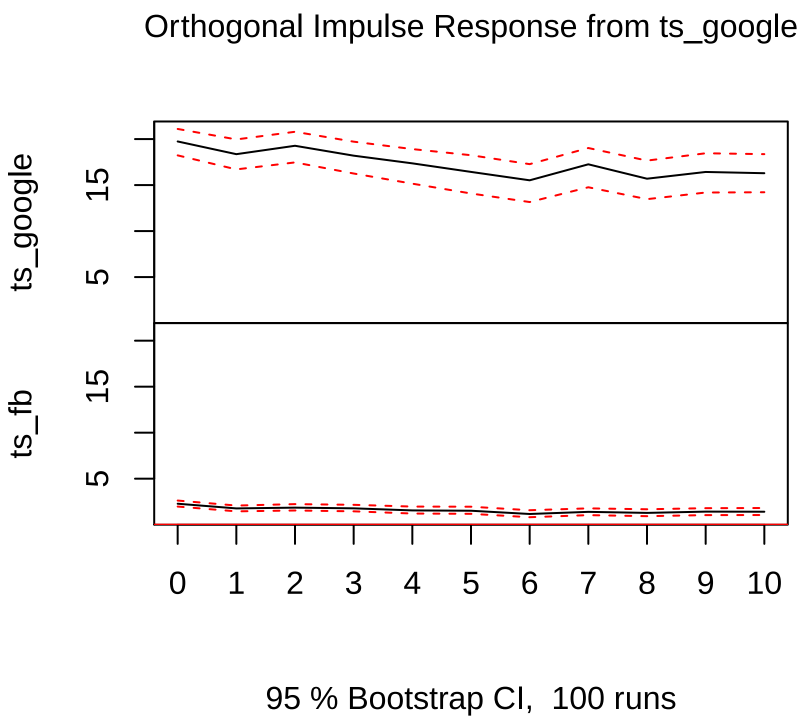
```
##
##
## Residual standard error: 19.74 on 2421 degrees of freedom
## Multiple R-Squared: 0.9991,  Adjusted R-squared: 0.9991
## F-statistic: 1.46e+05 on 18 and 2421 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation ts_fb:
## ======================================
## ts_fb = ts_google.l1 + ts_fb.l1 + ts_google.l2 + ts_fb.l2 + ts_google.l3 + ts_fb.l3 + ts_google.l4 +
##
##               Estimate Std. Error t value Pr(>|t|)
## ts_google.l1 -0.031228   0.004894  -6.381 2.10e-10 ***
## ts_fb.l1      1.046102   0.025276  41.387  < 2e-16 ***
## ts_google.l2  0.032958   0.006980   4.722 2.47e-06 ***
## ts_fb.l2     -0.032593   0.036445  -0.894 0.371251
## ts_google.l3  0.004889   0.007008   0.698 0.485451
## ts_fb.l3     -0.088769   0.036286  -2.446 0.014501 *
## ts_google.l4 -0.014558   0.007002  -2.079 0.037725 *
## ts_fb.l4      0.018158   0.036218   0.501 0.616161
## ts_google.l5  0.007811   0.007102   1.100 0.271507
## ts_fb.l5      0.018667   0.038542   0.484 0.628187
## ts_google.l6 -0.024455   0.007103  -3.443 0.000586 ***
## ts_fb.l6      0.083916   0.039517   2.124 0.033810 *
## ts_google.l7  0.044495   0.007170   6.206 6.38e-10 ***
## ts_fb.l7     -0.135048   0.039715  -3.400 0.000684 ***
## ts_google.l8 -0.018912   0.007221  -2.619 0.008878 **
## ts_fb.l8      0.045391   0.039826   1.140 0.254510
## ts_google.l9 -0.001792   0.005224  -0.343 0.731578
## ts_fb.l9      0.049101   0.028785   1.706 0.088174 .
## const         0.234915   0.149394   1.572 0.115977
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 3.835 on 2421 degrees of freedom
## Multiple R-Squared: 0.9982,  Adjusted R-squared: 0.9982
## F-statistic: 7.438e+04 on 18 and 2421 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##           ts_google ts_fb
## ts_google     389.8  45.0
## ts_fb          45.0  14.7
##
## Correlation matrix of residuals:
##           ts_google  ts_fb
## ts_google    1.0000 0.5944
## ts_fb        0.5944 1.0000
```

m

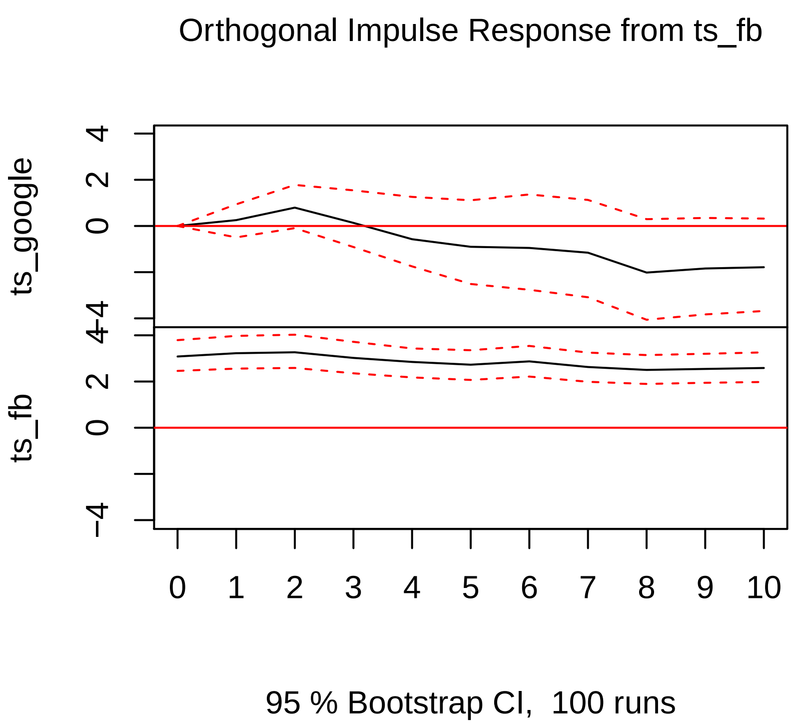
Compute, plot, and interpret the respective impulse response functions.

The impulse response from google shows a correlation of facebook with respect to the fluctuation of google, while opposite does not show a significant response of google to facebook.

```
plot(irf(model), plot.type = "multiple", names = "ts_google")
```




```
plot(irf(model), plot.type = "multiple", names = "ts_fb")
```



n

Perform a Granger-Causality test on your variables and discuss your results from the test.

The Granger test shows that facebook stock price is dependent on google while the converse is not true.

```
# Picks order 9
grangertest(ts_fb ~ ts_google, order = 9)
```

```
## Granger causality test
##
## Model 1: ts_fb ~ Lags(ts_fb, 1:9) + Lags(ts_google, 1:9)
## Model 2: ts_fb ~ Lags(ts_fb, 1:9)
##   Res.Df Df      F    Pr(>F)
## 1   2421
## 2   2430 -9 9.3487 4.423e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
grangertest(ts_google ~ ts_fb, order = 9)
```

```
## Granger causality test
##
## Model 1: ts_google ~ Lags(ts_google, 1:9) + Lags(ts_fb, 1:9)
```

```
## Model 2: ts_google ~ Lags(ts_google, 1:9)
##   Res.Df Df      F  Pr(>F)
## 1   2421
## 2   2430 -9 1.8777 0.05087 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

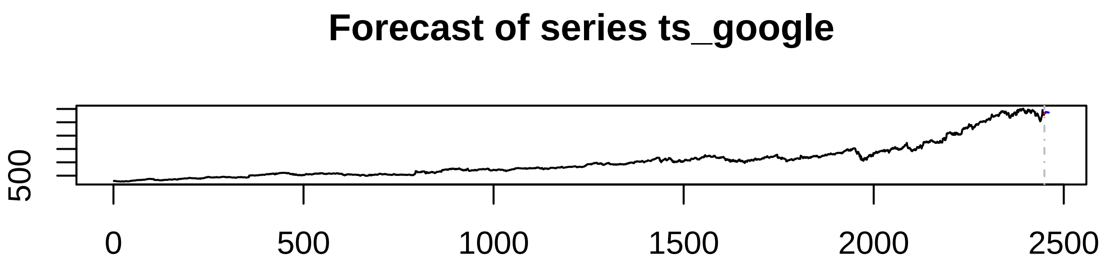
O

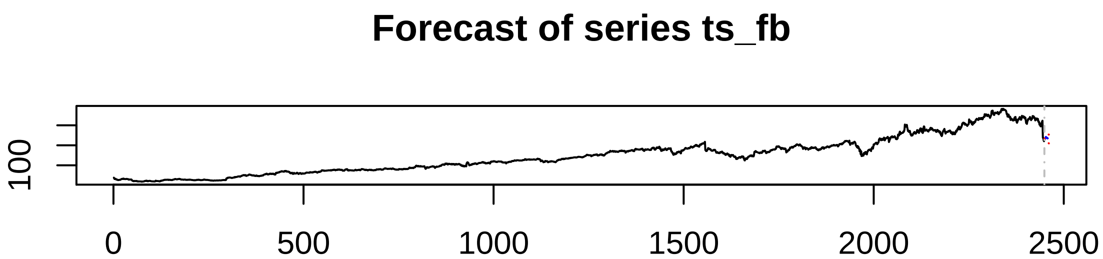
Use your VAR model to forecast 12-steps ahead. Your forecast should include the respective error bands. Comment on the differences between the VAR forecast and the other ones obtained using the different methods.

```r
prd <- predict(model, n.ahead = 12, ci = 0.95, dumvar = NULL)
print(prd)
```

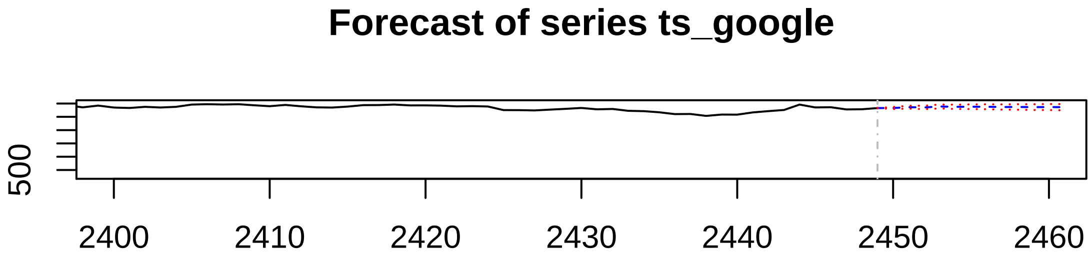
```
## $ts_google
##           fcst    lower    upper        CI
##  [1,] 2830.764 2792.069 2869.458  38.69442
##  [2,] 2861.946 2809.106 2914.787  52.84053
##  [3,] 2852.613 2787.642 2917.584  64.97057
##  [4,] 2889.065 2814.949 2963.181  74.11636
##  [5,] 2875.740 2794.178 2957.301  81.56150
##  [6,] 2880.036 2792.329 2967.744  87.70749
##  [7,] 2877.769 2784.920 2970.619  92.84950
##  [8,] 2868.743 2769.899 2967.587  98.84383
##  [9,] 2870.659 2767.065 2974.253 103.59398
## [10,] 2864.675 2756.134 2973.215 108.54048
## [11,] 2867.415 2754.222 2980.607 113.19236
## [12,] 2862.684 2744.971 2980.397 117.71300
##
## $ts_fb
##           fcst    lower    upper       CI
##  [1,] 229.4275 221.9120 236.9430  7.51549
##  [2,] 233.1294 222.7150 243.5438 10.41438
##  [3,] 237.3343 224.5821 250.0864 12.75219
##  [4,] 242.0739 227.5887 256.5592 14.48527
##  [5,] 239.0955 223.2771 254.9139 15.81844
##  [6,] 238.3258 221.3657 255.2859 16.96009
##  [7,] 236.6957 218.6812 254.7103 18.01454
##  [8,] 234.4066 215.4721 253.3412 18.93453
##  [9,] 234.6945 214.9749 254.4141 19.71958
## [10,] 232.5824 212.0520 253.1129 20.53044
## [11,] 232.3463 211.0202 253.6725 21.32616
## [12,] 231.2995 209.1829 253.4161 22.11657
```

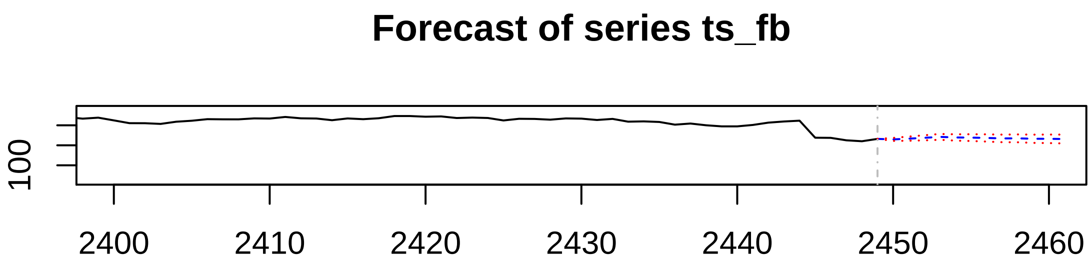
```r
plot(prd)
```






```
plot(prd, xlim=c(2400,2460))
```


Forecast of series ts_google
500
2400
2410
2420
2430
2440
2450
2460


Forecast of series ts_fb
100
2400
2410
2420
2430
2440
2450
2460