Krysten Thompson - w271: Homework 9 (Due: Week 11)

Professor Jeffrey Yau

Develop a Vector Autoregressive Model:

Use series `series01_liveSession_wk10.csv` and build a VAR model. You will have to examine the data, conduct EDA, use `VARselect` to choose a model that minimize `SC` (which is also `BIC`), estimate the chosen model, conduct residual diagnostic, test model assumptions, and make a 3-step ahead forecast.

```
library(knitr)
opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)
```

```
library(car)
library(dplyr)
library(Hmisc)
library(gridExtra)
library(forecast)
library(fpp2)
library(astsa)
library(xts)
library(vars)
```

```
d <- read.csv("series01_liveSession_wk10.csv", header = TRUE, sep=",")
```

```
str(d)
```

```
## 'data.frame':    120 obs. of  2 variables:
##  $ series1: num  -0.172 -0.73 -0.19 -0.275 0.331 ...
##  $ series2: num  -0.1683 0.0061 1.49 -0.0695 -0.13 ...
```

```
head(d,3)
```

```
##      series1      series2
## 1 -0.1721111 -0.168339319
## 2 -0.7301398  0.006102507
## 3 -0.1897081  1.490036660
```

```
tail(d,3)
```

```
##       series1   series2
## 118 -3.367291 -1.857246
## 119 -2.918558 -1.749286
## 120 -2.630061 -1.313685
```

```
describe(d)
```

```
## d
##
##  2  Variables      120  Observations
## ---------------------------------------------------------------------------
## series1
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##      120        0      120        1  -0.0181    1.809  -2.5678  -2.1934
##      .25      .50      .75      .90      .95
##  -1.0350  -0.1822   0.8870   2.2197   2.6019
##
## lowest : -3.367291 -3.088285 -2.918558 -2.802666 -2.630061
## highest:  2.875165  3.098829  3.477134  3.691651  4.313376
## ---------------------------------------------------------------------------
## series2
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##      120        0      120        1    0.174     1.58  -1.8674  -1.7242
##      .25      .50      .75      .90      .95
##  -0.9488   0.2286   0.9687   1.9712   2.8530
##
## lowest : -2.476277 -2.389847 -2.328863 -2.156874 -2.055236
## highest:  2.964379  2.968765  2.984522  3.007582  3.835152
## ---------------------------------------------------------------------------
```

```
values = seq(from = as.Date("2000-01-01"), to = as.Date("2009-12-31"), by = 'month')
```

```
d$date <- values
```

```
head(d)
```

```
##      series1      series2       date
## 1 -0.1721111 -0.168339319 2000-01-01
## 2 -0.7301398  0.006102507 2000-02-01
## 3 -0.1897081  1.490036660 2000-03-01
## 4 -0.2750123 -0.069454043 2000-04-01
## 5  0.3305953 -0.129991203 2000-05-01
## 6 -0.2726928  0.600968225 2000-06-01
```

```
d <- d[,c(3,1,2)]
head(d,3)
```

```
##         date    series1      series2
## 1 2000-01-01 -0.1721111 -0.168339319
## 2 2000-02-01 -0.7301398  0.006102507
## 3 2000-03-01 -0.1897081  1.490036660
```

```
d.ts <- ts(d)
head(d.ts)
```

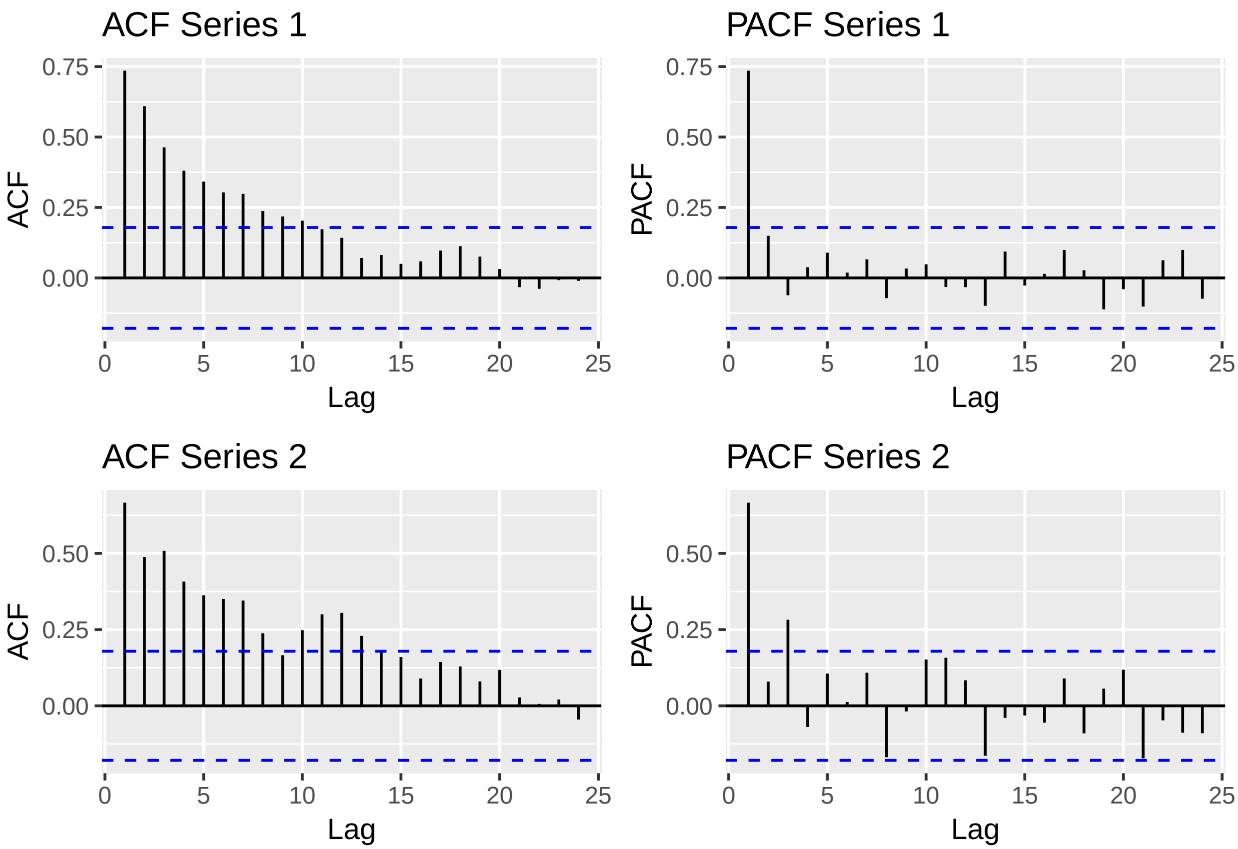
```
## Time Series:
## Start = 1
## End = 6
```

```
## Frequency = 1
##    date    series1      series2
## 1 10957 -0.1721111 -0.168339319
## 2 10988 -0.7301398  0.006102507
## 3 11017 -0.1897081  1.490036660
## 4 11048 -0.2750123 -0.069454043
## 5 11078  0.3305953 -0.129991203
## 6 11109 -0.2726928  0.600968225
```

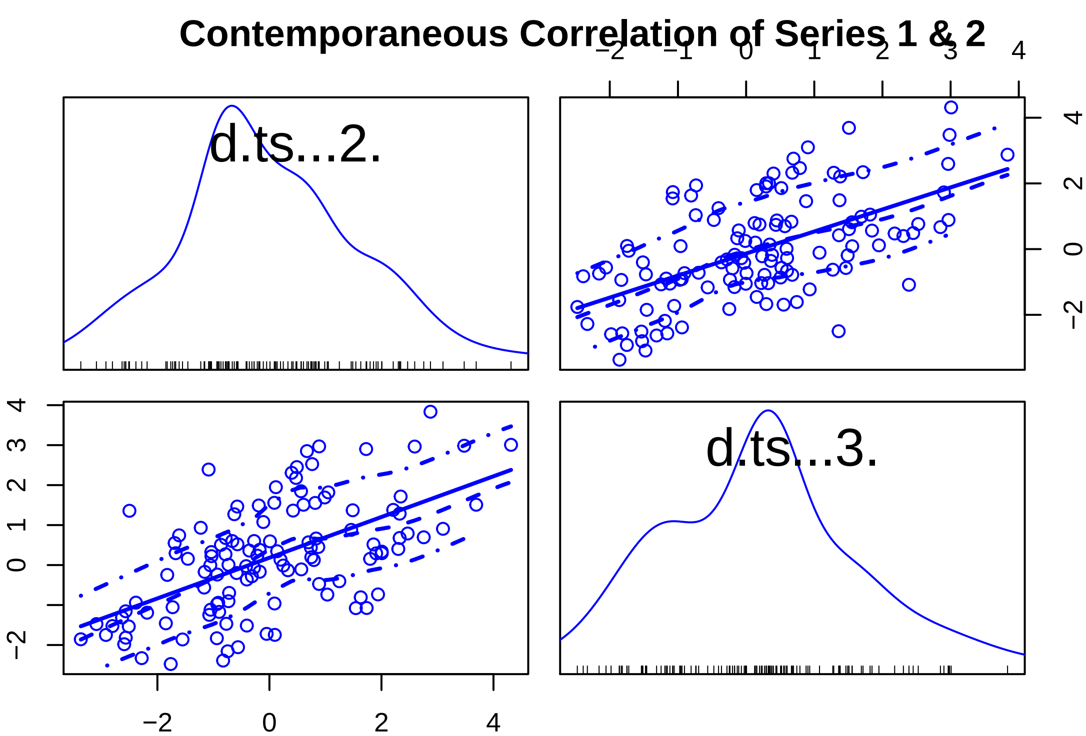
```
#Plot series data
plot.ts(d.ts, main="Plot of Series 1 and Series 2")
```


```
idx <- seq(from = as.Date("2019-01-01"), to = as.Date("2019-04-30"), by = 'day')
```

```
# check autocorrelation and partial autocorrelation for each series
s1 <- ggAcf(d.ts[,2], lag=24) + ggtitle("ACF Series 1")

s1p <- ggPacf(d.ts[,2], lag=24) + ggtitle("PACF Series 1")

s2 <- ggAcf(d.ts[,3], lag=24) + ggtitle("ACF Series 2")

s2p <- ggPacf(d.ts[,3], lag=24) + ggtitle("PACF Series 2")

grid.arrange(s1, s1p, s2, s2p, ncol=2)
```




```r
scatterplotMatrix(~ d.ts[,2] + d.ts[,3])
title("Contemporaneous Correlation of Series 1 & 2")
```



Select optimal number of lags:

```r
#I tested many lag.max (all the way up to 24) and it kept coming back to 1
VARselect(d.ts, lag.max = 5, type = "both")
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      1      1      1      1
##
## $criteria
##                 1          2          3          4           5
## AIC(n) -0.7320088 -0.6878208 -0.6356255 -0.6069818 -0.52978215
## HQ(n)  -0.5866843 -0.4553016 -0.3159117 -0.2000733 -0.03567896
## SC(n)  -0.3739741 -0.1149654  0.1520507  0.3955151  0.68753557
## FPE(n)  0.4810211  0.5030099  0.5305370  0.5469811  0.59260018
```

In a next step, the VAR model is estimated with the function VAR() and as deterministic regressors a constant is included.

```r
var.fit <- VAR(d.ts, p = 1, type = "both")
summary(var.fit)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: date, series1, series2
## Deterministic variables: both
## Sample size: 119
## Log Likelihood: -445.182
## Roots of the characteristic polynomial:
## 0.8372 0.3397 0.1534
## Call:
## VAR(y = d.ts, p = 1, type = "both")
##
##
## Estimation results for equation date:
## =====================================
## date = date.l1 + series1.l1 + series2.l1 + const + trend
##
##             Estimate Std. Error t value Pr(>|t|)
## date.l1     2.859e-01  8.997e-02   3.178  0.00191 **
## series1.l1 -4.312e-02  4.775e-02  -0.903  0.36838
## series2.l1  2.418e-03  5.393e-02   0.045  0.96432
## const       7.811e+03  9.804e+02   7.967 1.35e-12 ***
## trend       2.173e+01  2.739e+00   7.936 1.59e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.6607 on 114 degrees of freedom
## Multiple R-Squared:     1,   Adjusted R-squared:     1
```

```
## F-statistic: 7.451e+07 on 4 and 114 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation series1:
## ========================================
## series1 = date.l1 + series1.l1 + series2.l1 + const + trend
##
##              Estimate Std. Error t value Pr(>|t|)
## date.l1      -0.06628    0.13458  -0.492    0.623
## series1.l1    0.54617    0.07142   7.647 7.09e-12 ***
## series2.l1    0.38155    0.08066   4.731 6.47e-06 ***
## const       722.17954 1466.33410   0.493    0.623
## trend         2.01494    4.09605   0.492    0.624
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.9881 on 114 degrees of freedom
## Multiple R-Squared: 0.6312,  Adjusted R-squared: 0.6183
## F-statistic: 48.78 on 4 and 114 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation series2:
## ========================================
## series2 = date.l1 + series1.l1 + series2.l1 + const + trend
##
##              Estimate Std. Error t value Pr(>|t|)
## date.l1     1.918e-01  1.341e-01   1.430 0.155377
## series1.l1  2.686e-01  7.116e-02   3.774 0.000257 ***
## series2.l1  4.981e-01  8.035e-02   6.199 9.34e-09 ***
## const      -2.089e+03  1.461e+03  -1.430 0.155416
## trend      -5.838e+00  4.081e+00  -1.431 0.155245
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.9844 on 114 degrees of freedom
## Multiple R-Squared: 0.5175,  Adjusted R-squared: 0.5006
## F-statistic: 30.57 on 4 and 114 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##             date  series1  series2
## date     0.43647 -0.05136 -0.04452
## series1 -0.05136  0.97644  0.08982
## series2 -0.04452  0.08982  0.96914
##
```

```
## Correlation matrix of residuals:
##             date  series1  series2
## date     1.00000 -0.07868 -0.06845
## series1 -0.07868  1.00000  0.09233
## series2 -0.06845  0.09233  1.00000
```

The moduli of the eigenvalues of the companion matrix are all less than one.

```r
roots(var.fit)
```

```
## [1] 0.8371662 0.3396539 0.1533878
```

Diagnostic Testing:

```r
# Test of normality:
normality.test(var.fit, multivariate.only = TRUE)
```

```
## $JB
##
##  JB-Test (multivariate)
##
## data:  Residuals of VAR object var.fit
## Chi-squared = 4.0472, df = 6, p-value = 0.6703
##
##
## $Skewness
##
##  Skewness only (multivariate)
##
## data:  Residuals of VAR object var.fit
## Chi-squared = 3.4472, df = 3, p-value = 0.3277
##
##
## $Kurtosis
##
##  Kurtosis only (multivariate)
##
## data:  Residuals of VAR object var.fit
## Chi-squared = 0.60001, df = 3, p-value = 0.8964
```

```r
# Test of no serial correlation:
serial.test(var.fit, lags.pt = 12, type = "PT.asymptotic")
```

```
##
##  Portmanteau Test (asymptotic)
##
## data:  Residuals of VAR object var.fit
## Chi-squared = 197.99, df = 99, p-value = 1.384e-08
```

```r
#plot(var.fit.ptasy)

# Test of the absence of ARCH effect:
arch.test(var.fit)
```

```
##
##  ARCH (multivariate)
##
## data:  Residuals of VAR object var.fit
## Chi-squared = 204.6, df = 180, p-value = 0.1009
```

Forecast:

```r
forecast(var.fit) %>% autoplot() +
  ggtitle("Var Model Forecast") + xlab("Year")  + ylab("Value")
```