LAPORAN PRAKTIKUM PERAMALAN VAR

Disusun Untuk Memenuhi Tugas

Mata Kuliah:

Praktikum Pemodelan Statistika Terapan

Wahyu Ikbal Maulana

(3323600056)

PROGRAM STUDI STr SAINS DATA TERAPAN

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

A. Percobaan 3

a. Kode, Output, dan Analisis

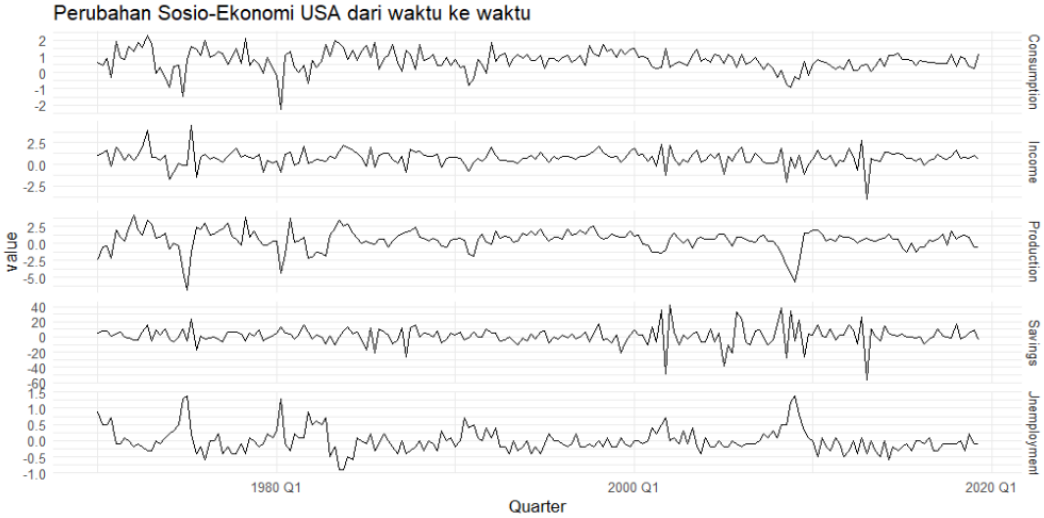
```
#1. Gunakan library fpp3
#Time Series #3
#1. Gunakan library fpp3
library(fpp3)
us_change
```

A tsibble: 198 x 6 [1Q]

Quarter <S3: yearquarter>	Consumption <dbl>	Income <dbl>	Production <dbl>	Savings <dbl>	Unemployment <dbl>
0	0.618566400	1.04480133	-2.45248553	5.29901407	0.9
90	0.451984019	1.22564719	-0.55145947	7.78989382	0.5
181	0.872871780	1.58515384	-0.35865175	7.40398407	0.5
273	-0.271847933	-0.23954487	-2.18569087	1.16989824	0.7
365	1.901344959	1.97592495	1.90976436	3.53566693	-0.1
455	0.914877346	1.44590853	0.90156952	5.87476357	-0.1
546	0.794109641	0.52114911	0.30803071	-0.40623534	0.1
638	1.645633164	1.15917609	2.29136214	-1.48625890	0.0
730	1.311189694	0.45685671	4.15429495	-4.29197220	-0.2
821	1.885777877	1.03338931	1.88867271	-4.69219636	-0.1
912	1.529333612	1.92629490	1.26541145	5.92345399	-0.2
1004	2.319557542	3.92994922	3.43958747	16.07351412	-0.3

1-12 of 198 rows Previous 1 2 3 4 5 6 ... 17 Next

```
#2. EDA
us_change %>%
  pivot_longer(-Quarter, names_to = "variable", values_to = "value") %>%
  ggplot(aes(x = Quarter, y = value)) +
  geom_line() +
  facet_grid(variable ~ ., scales = "free_y") +
  labs(title = "Perubahan Sosio-Ekonomi USA dari waktu ke waktu") +
  theme_minimal()
```



Analisis:

Di atas adalah EDA untuk membuat plot untuk setiap variabel untuk mengetahui secara garis besar data sudah stasioner atau belum

```
library(tseries)
df <- us_change[, -1]
stationary_test <- data.frame("ADF" = double(), "KPSS" = double())
for (i in 1:ncol(df)) {
  stationary_test[i, "ADF"] <- adf.test(pull(df[, i]))$p.value
  stationary_test[i, "KPSS"] <- kpss.test(pull(df[, i]))$p.value
}
stationary_test %>%
  mutate(variable = colnames(df)) %>%
  select(variable, ADF, KPSS)
```

Description: df [5 × 3]

	variable <chr>	ADF <dbl>	KPSS <dbl>
1	Consumption	0.01	0.1
2	Income	0.01	0.1
3	Production	0.01	0.1
4	Savings	0.01	0.1
5	Unemployment	0.01	0.1

5 rows

Analisis:

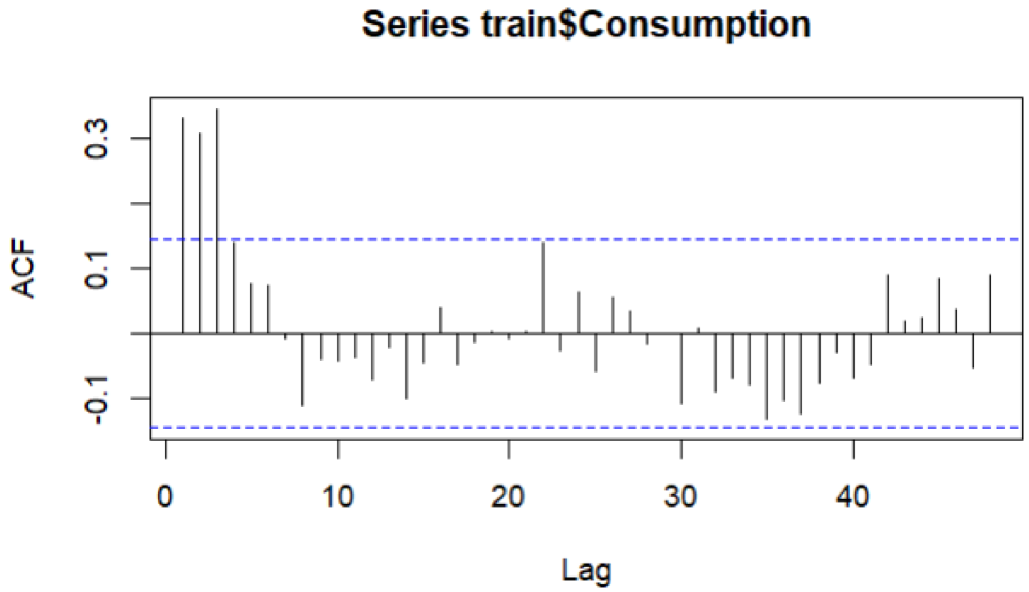
Berdasarkan p-value pada ADF test, seluruh variabel mempunyai p-value 0.01 (Stasioner) dan KPSS test 0.1 (Stasioner), sehingga didapat bahwa kelima variabel tersebut signifikan stasioner.

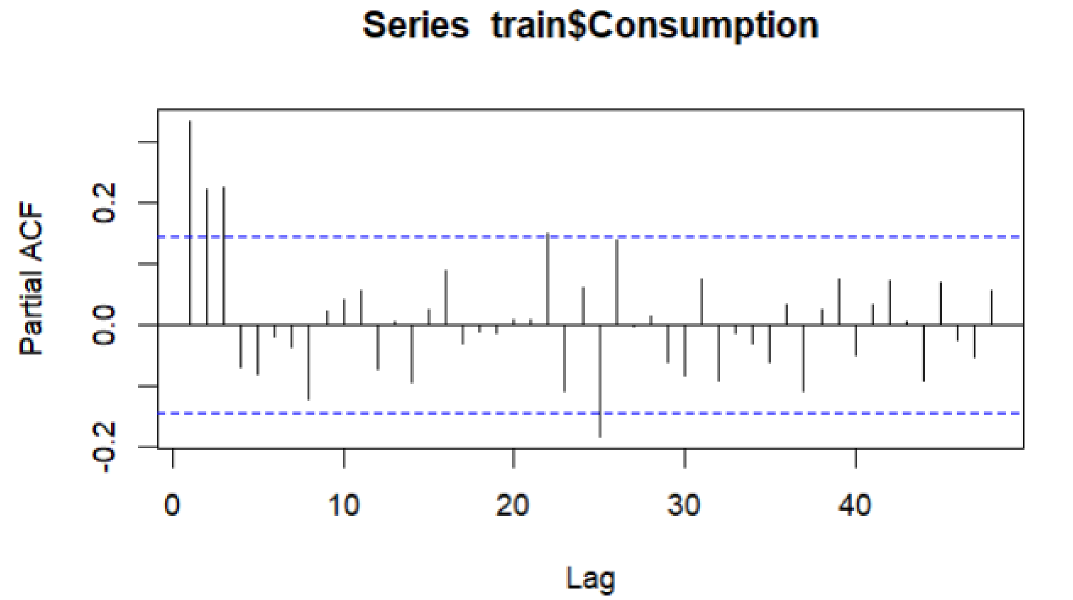
```
test <- us_change %>%
  mutate(year = year(Quarter)) %>%
  filter(year >= 2016)
train <- us_change %>%
  mutate(year = year(Quarter)) %>%
  filter(year < 2016)
```

Analisis:

Membagi dataset menjadi 2 subset data, yakni data train dan data test.

```
#Plot ACF dan PACF
acf(train$Consumption, 48) # menampilkan ACF sampai lag 48
pacf(train$Consumption, 48) # menampilkan PACF sampai lag 48
```





Analisis:

Menampilkan plot ACF dan plot PACF sampai dengan lag 48

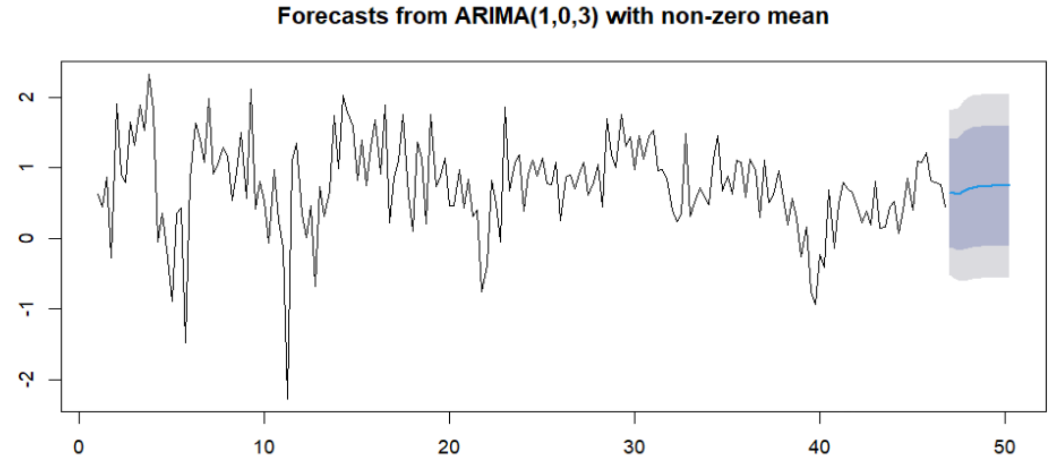
```r
library(forecast)
fit <- auto.arima(ts(train$Consumption, frequency = 4), seasonal = F)
summary(fit)
```

```
Series: ts(train$Consumption, frequency = 4)
ARIMA(1,0,3) with non-zero mean

Coefficients:
         ar1      ma1    ma2     ma3    mean
      0.5747  -0.3581  0.093  0.1946  0.7418
s.e.  0.1526   0.1635  0.081  0.0857  0.0936

sigma^2 = 0.3533:  log likelihood = -163.03
AIC=338.06   AICc=338.53   BIC=357.35

Training set error measures:
                      ME      RMSE       MAE       MPE     MAPE
Training set 0.001107612 0.5862617 0.4378066 -35.61455 161.7278
                 MASE         ACF1
Training set 0.672272 -0.002685885
```

Analisis:

Menampilkan "summary" dari pemodelan ARIMA orde "(1,0,3)" dan menampilkan ukuran error pada model data train ARIMA.

```r
#Prediksi/forcasting
fore <- forecast(object = fit, h = nrow(test))
plot(fore)
```



Analisis:

Di atas adalah hasil plot untuk visualisasi peramalan dari model ARIMA

```
#Model ARIMAX
library(forecast)
library(lmtest)
datay = train[,2]
datax = train[,3:6]
estimasiarmax = arima(datay, order=c(1,0,3), xreg=datax)
estimasiarmax; coeftest(estimasiarmax)
```

```
Call:
arima(x = datay, order = c(1, 0, 3), xreg = datax)

Coefficients:
         ar1      ma1     ma2      ma3  intercept  Income  Production
      0.0508  -0.1519  0.1674  -0.1657     0.2461  0.7628      0.0422
s.e.  0.2762   0.2685  0.0878   0.1048     0.0352  0.0429      0.0236
      Savings  Unemployment
      -0.0541       -0.1648
s.e.   0.0033        0.0995

sigma^2 estimated as 0.09544:  log likelihood = -45.02,  aic = 108.03

z test of coefficients:

               Estimate Std. Error  z value  Pr(>|z|)
ar1           0.0507915  0.2761992   0.1839   0.85410
ma1          -0.1518913  0.2684846  -0.5657   0.57157
ma2           0.1673974  0.0877953   1.9067   0.05656 .
ma3          -0.1656920  0.1047923  -1.5811   0.11384
intercept     0.2460633  0.0352217   6.9861 2.826e-12 ***
Income        0.7627902  0.0428927  17.7837 < 2.2e-16 ***
Production    0.0422120  0.0235944   1.7891   0.07360 .
Savings      -0.0541032  0.0032678 -16.5564 < 2.2e-16 ***
Unemployment -0.1647536  0.0994910  -1.6560   0.09773 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Analisis:

Ringkasan model ARIMA menunjukkan orde ARIMA yang telah disesuaikan, sementara pengujian koefisien menggunakan fungsi coeftest() menghasilkan uji statistik untuk masing-masing koefisien dalam model, yang memungkinkan untuk menentukan signifikansi mereka dalam memprediksi variabel respons.

Dengan menggunakan model ARIMA dengan variabel eksogen, Anda dapat memperhitungkan efek variabel prediktor dalam memprediksi variabel respons. Ini dapat meningkatkan akurasi ramalan Anda dengan memasukkan informasi tambahan yang relevan ke dalam model.

```
#Model ARIMAX
library(forecast)
library(lmtest)
datay = train[,2]
#datax = cbind(train[,3],train[,5:6])
datax = train[,3:6]
estimasiarmax2 = arima(datay, order=c(0,1,1), xreg=datax)
estimasiarmax2; coeftest(estimasiarmax2)
```

```
Call:
arima(x = datay, order = c(0, 1, 1), xreg = datax)

Coefficients:
          ma1  Income  Production  Savings  Unemployment
      -0.9771  0.7288      0.0384  -0.0519       -0.2258
s.e.   0.0165  0.0415      0.0252   0.0030        0.1063

sigma^2 estimated as 0.1003:  log likelihood = -50.82,  aic = 111.64

z test of coefficients:

              Estimate Std. Error  z value Pr(>|z|)
ma1         -0.9770868  0.0164651 -59.3430  < 2e-16 ***
Income       0.7288264  0.0414953  17.5641  < 2e-16 ***
Production   0.0384058  0.0252318   1.5221  0.12798
Savings     -0.0518911  0.0030238 -17.1609  < 2e-16 ***
Unemployment -0.2258255 0.1063100  -2.1242  0.03365 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Analisis:

Dengan menggunakan model ARIMAX, Anda dapat memperhitungkan efek variabel prediktor eksternal dalam memprediksi variabel respons. Ini memungkinkan Anda untuk memperbaiki akurasi ramalan Anda dengan memasukkan informasi tambahan yang relevan ke dalam model.

Nilai estimasi koefisien dari model ARIMAX. Anda memiliki koefisien untuk komponen moving average (ma1) serta untuk masing-masing variabel eksogen (Income, Production, Savings, dan Unemployment). Koefisien positif atau negatif menunjukkan arah dan kekuatan hubungan antara variabel eksogen dan variabel respons.

```
summary(estimasiarmax)
```

```
Call:
arima(x = datay, order = c(1, 0, 3), xreg = datax)

Coefficients:
         ar1      ma1     ma2      ma3  intercept  Income  Production
      0.0508  -0.1519  0.1674  -0.1657     0.2461  0.7628      0.0422
s.e.  0.2762   0.2685  0.0878   0.1048     0.0352  0.0429      0.0236
      Savings  Unemployment
      -0.0541       -0.1648
s.e.   0.0033        0.0995

sigma^2 estimated as 0.09544:  log likelihood = -45.02,  aic = 108.03

Training set error measures:
                     ME      RMSE       MAE        MPE      MAPE
Training set 0.02702775 0.322829 0.2434111 0.07587493 0.7515478
                  MASE        ACF1
Training set 0.5419737 -0.05040848
```

Analisis:

Menampilkan ringkasan dari "estimasiarmax" dengan menampilkan ukuran error pada model train.

```
res_arimax=estimasiarmax2$residuals
Box.test(res_arimax,lag = 20)
```

```
         Box-Pierce test

data:  res_arimax
X-squared = 24.699, df = 20, p-value = 0.2132
```

Analisis:

Uji Box-Pierce yang dijalankan bertujuan untuk menguji apakah ada ketergantungan serial pada residual dari model ARIMAX. Dengan nilai p-value sebesar 0.2132 (lebih besar dari tingkat signifikansi umum 0.05), gagal menolak hipotesis nol. Ini menunjukkan bahwa tidak ada bukti yang cukup untuk menyatakan bahwa ada ketergantungan serial dalam residual pemodelan ARIMAX. Dengan demikian, model mungkin berhasil menangkap pola-pola dalam data dengan baik.

B. Percobaan 4

a. Kode, Output, dan Analisis

```
library(vars)
library(mFilter)
library(tseries)
library(TSstudio)
library(forecast)
library(tidyverse)
```

```
mp <- read_csv("E:/Peramalan VAR/SampleVAR.csv")
head(mp)
```

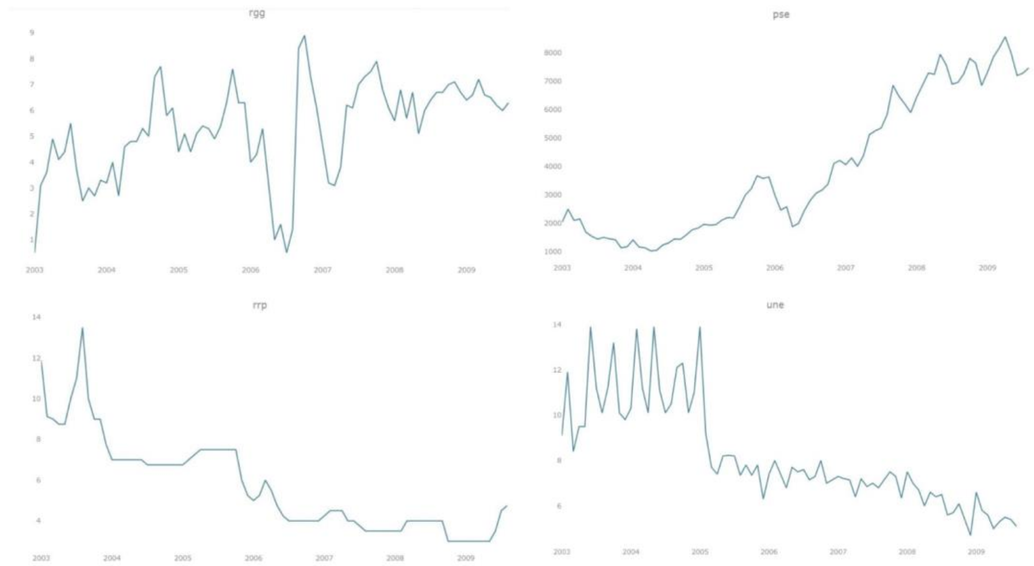
A tibble: 6 × 5

date <chr>	real_gdp_growth <dbl>	psei <dbl>	bsp_rrp <dbl>	unem <dbl>
3/31/1999	0.5	2028.21	11.875	9.1
6/30/1999	3.1	2486.96	9.125	11.9
9/30/1999	3.6	2096.20	9.000	8.4
12/31/1999	4.9	2142.97	8.750	9.5
3/31/2000	4.1	1681.72	8.750	9.5
6/30/2000	4.4	1533.99	10.000	13.9

6 rows

```
rgg <- ts(mp$real_gdp_growth, start = c(2003,1,1), frequency = 12)
pse <- ts(mp$psei, start = c(2003,1,1), frequency = 12)
rrp <- ts(mp$bsp_rrp, start = c(2003,1,1), frequency = 12)
une <- ts(mp$unem, start = c(2003,1,1), frequency = 12)
```

```
ts_plot(rgg)
ts_plot(pse)
ts_plot(rrp)
ts_plot(une)
```



Analisis:

Menampilkan plot time series dari masing-masing variabel dataset.

```
#Uji Stasioneritas
pp.test(rgg)
pp.test(pse)
pp.test(rrp)
pp.test(une)
```

```
Warning: p-value smaller than printed p-value
        Phillips-Perron Unit Root Test

data:  rgg
Dickey-Fuller Z(alpha) = -34.082, Truncation lag parameter =
3, p-value = 0.01
alternative hypothesis: stationary


        Phillips-Perron Unit Root Test

data:  pse
Dickey-Fuller Z(alpha) = -9.2626, Truncation lag parameter =
3, p-value = 0.5715
alternative hypothesis: stationary


        Phillips-Perron Unit Root Test

data:  rrp
Dickey-Fuller Z(alpha) = -22.911, Truncation lag parameter =
3, p-value = 0.02667
alternative hypothesis: stationary

Warning: p-value smaller than printed p-value
        Phillips-Perron Unit Root Test

data:  une
Dickey-Fuller Z(alpha) = -51.958, Truncation lag parameter =
3, p-value = 0.01
alternative hypothesis: stationary
```

Analisis:

Menguji stasioneritas data, dapat disimpulkan bahwa semua data dari setiap variabel bersifat stasioner.

```
#Estimasi menggunakan VAR tanpa PSE
v1 <- cbind(rgg, rrp, une)
colnames(v1) <- cbind("RGG","RRP","UNE")
v2 <- cbind(rgg, pse, rrp)
colnames(v2) <- cbind("RGG","PSE","RRP")
lagselect <- VARselect(v1, lag.max = 15, type = "const")
lagselect
lagselect$selection
```

```
$selection
AIC(n)  HQ(n)  SC(n) FPE(n)
    15      3      1      4

$criteria
                1          2          3          4          5
AIC(n) -1.3346640 -1.3908480 -1.6355168 -1.6504343 -1.4206943
HQ(n)  -1.1762760 -1.1136690 -1.2395467 -1.1356732 -0.7871421
SC(n)  -0.9332387 -0.6883537 -0.6319534 -0.3458019  0.1850071
FPE(n)  0.2633696  0.2494910  0.1962965  0.1951398  0.2491424
                6          7          8          9         10
AIC(n) -1.4418180 -1.5687384 -1.4773771 -1.4577291 -1.3984550
HQ(n)  -0.6894748 -0.6976042 -0.4874519 -0.3490128 -0.1709478
SC(n)   0.4649524  0.6391010  1.0315313  1.3522483  1.7125914
FPE(n)  0.2492901  0.2264155  0.2586502  0.2787891  0.3178962
                11          12          13          14          15
AIC(n) -1.30221158 -1.2050768 -1.2467267 -1.6504750 -1.89072155
HQ(n)   0.04408668  0.2600124  0.3371536  0.0521963 -0.06925919
SC(n)   2.10990382  2.5081076  2.7675267  2.6648474  2.72566988
FPE(n)  0.38393762  0.4760861  0.5305820  0.4289395  0.43101959

AIC(n)  HQ(n)  SC(n) FPE(n)
    15      3      1      4
```

Analisis:

Melakukan estimasi VAR tanpa menggunakan variabel "PSE". Dapat diketahui, bahwa kriteria yang digunakan untuk memilih jumlah lag terbaik adalah AIC (Akaike Information Criterion). Jumlah lag terbaik yang dipilih berdasarkan AIC adalah 15. Namun, mungkin penting untuk melihat juga hasil kriteria informasi lainnya untuk memastikan konsistensi dalam pemilihan jumlah lag terbaik.

```r
#Model Diagnostik
Model1 <- VAR(v1, p = 2, type = "const", season = NULL, exog = NULL)
Model2 <- VAR(v2, p = 2, type = "const", season = NULL, exog = NULL)
summary(Model1)
```

```
VAR Estimation Results:
=========================
Endogenous variables: RGG, RRP, UNE
Deterministic variables: const
Sample size: 78
Log Likelihood: -319.319
Roots of the characteristic polynomial:
0.955 0.5928 0.4971 0.4971 0.2507 0.09807
Call:
VAR(y = v1, p = 2, type = "const", exogen = NULL)


Estimation results for equation RGG:
====================================
RGG = RGG.l1 + RRP.l1 + UNE.l1 + RGG.l2 + RRP.l2 + UNE.l2 + const

       Estimate Std. Error t value Pr(>|t|)
RGG.l1  0.79677    0.11464   6.950 1.45e-09 ***
RRP.l1 -0.03069    0.21170  -0.145 0.885146
UNE.l1  0.01273    0.10383   0.123 0.902794
RGG.l2 -0.20417    0.11082  -1.842 0.069600 .
RRP.l2 -0.11575    0.20076  -0.577 0.566050
UNE.l2 -0.01829    0.10585  -0.173 0.863307
const   3.06626    0.87349   3.510 0.000782 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 1.181 on 71 degrees of freedom
Multiple R-Squared: 0.5814,     Adjusted R-squared: 0.546
F-statistic: 16.43 on 6 and 71 DF,  p-value: 9.028e-12


Estimation results for equation RRP:
====================================
RRP = RGG.l1 + RRP.l1 + UNE.l1 + RGG.l2 + RRP.l2 + UNE.l2 + const

       Estimate Std. Error t value Pr(>|t|)
RGG.l1  0.06435    0.06079   1.059   0.2934
RRP.l1  0.96465    0.11227   8.592 1.33e-12 ***
UNE.l1  0.09761    0.05506   1.773   0.0806 .
RGG.l2 -0.05779    0.05877  -0.983   0.3288
RRP.l2 -0.10184    0.10646  -0.957   0.3420
UNE.l2  0.01425    0.05613   0.254   0.8003
const  -0.22843    0.46322  -0.493   0.6234
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 0.6264 on 71 degrees of freedom
Multiple R-Squared: 0.9272,     Adjusted R-squared: 0.921
F-statistic: 150.6 on 6 and 71 DF,  p-value: < 2.2e-16
```

```
Estimation results for equation UNE:
====================================
UNE = RGG.l1 + RRP.l1 + UNE.l1 + RGG.l2 + RRP.l2 + UNE.l2 + const

        Estimate Std. Error t value Pr(>|t|)
RGG.l1 -0.001015   0.128496  -0.008    0.994
RRP.l1  0.038416   0.237300   0.162    0.872
UNE.l1  0.481141   0.116384   4.134 9.63e-05 ***
RGG.l2 -0.021819   0.124220  -0.176    0.861
RRP.l2  0.187530   0.225032   0.833    0.407
UNE.l2  0.185723   0.118650   1.565    0.122
const   1.464491   0.979099   1.496    0.139
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 1.324 on 71 degrees of freedom
Multiple R-Squared: 0.6968,     Adjusted R-squared: 0.6712
F-statistic:  27.2 on 6 and 71 DF,  p-value: < 2.2e-16



Covariance matrix of residuals:
        RGG      RRP      UNE
RGG 1.39523  0.02849  0.04877
RRP 0.02849  0.39237 -0.01243
UNE 0.04877 -0.01243  1.75300

Correlation matrix of residuals:
        RGG      RRP      UNE
RGG 1.00000  0.03851  0.03118
RRP 0.03851  1.00000 -0.01498
UNE 0.03118 -0.01498  1.00000
```

Analisis:

Menampilkan ringkasan statistik dari variabel model1.

```
Arch1 <- arch.test(Model1, lags.multi = 15, multivariate.only = TRUE)
Arch1
```

```
        ARCH (multivariate)

data:  Residuals of VAR object Model1
Chi-squared = 378, df = 540, p-value = 1
```

Analisis:

Hasil uji statistik ARCH (Autoregressive Conditional Heteroskedasticity) menunjukkan hasil uji keberadaan heteroskedastisitas kondisional dalam residu model VAR. berdasarkan hasil uji ini, tidak ada bukti yang cukup untuk menolak hipotesis nol bahwa tidak ada heteroskedastisitas kondisional dalam residu model VAR Anda. Ini mengindikasikan bahwa residu tersebut mungkin homoskedastis dan tidak menunjukkan pola heteroskedastisitas kondisional yang signifikan.

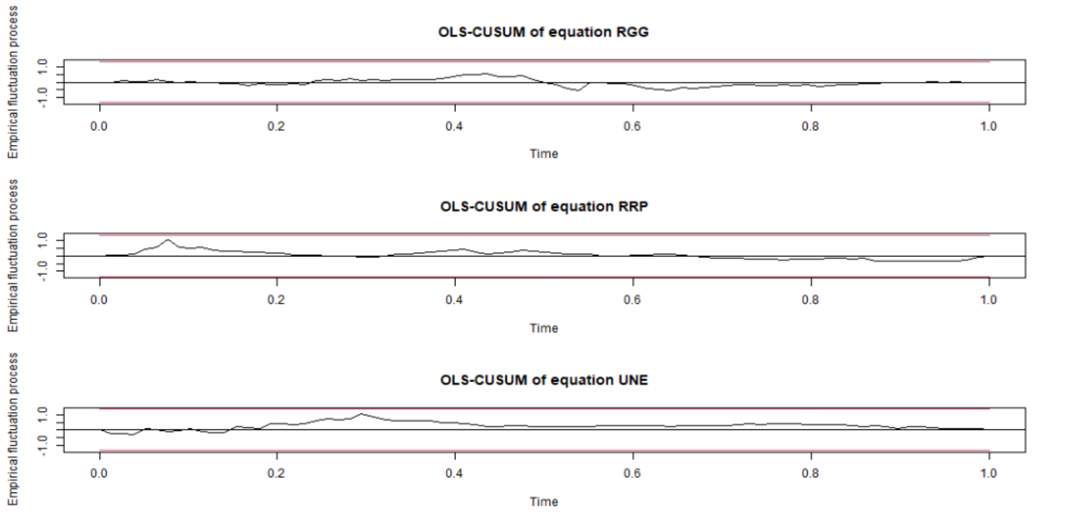
```
#Uji Normalitas
Norm1 <- normality.test(Model1, multivariate.only = TRUE)
Norm1
```

```
$JB

        JB-Test (multivariate)

data:  Residuals of VAR object Model1
Chi-squared = 431.77, df = 6, p-value < 2.2e-16


$Skewness

        Skewness only (multivariate)

data:  Residuals of VAR object Model1
Chi-squared = 28.637, df = 3, p-value = 2.669e-06


$Kurtosis

        Kurtosis only (multivariate)

data:  Residuals of VAR object Model1
Chi-squared = 403.14, df = 3, p-value < 2.2e-16
```

Analisis:

Berdasarkan hasil ini, dapat disimpulkan bahwa residu dari model VAR tidak mengikuti distribusi normal multivariat. Hal ini ditunjukkan oleh nilai p-value yang sangat kecil untuk JB-Test, skewness multivariat, dan kurtosis multivariat.

```
Stability1 <- stability(Model1, type = "OLS-CUSUM")
plot(Stability1)
```



Analisis:

Menampilkan visualisasi untuk menguji stabilitas koefisien model VAR terhadap perkembangan struktural menggunakan metode OLS-CUSUM.

```r
Granger_rgg<- causality(Model2, cause = "RGG")
Granger_rgg
Granger_pse <- causality(Model2, cause = "PSE")
Granger_pse
Granger_rrp <- causality(Model2, cause = "RRP")
Granger_rrp
```

```
$Granger

	Granger causality H0: RGG do not Granger-cause PSE RRP

data:  VAR object Model2
F-Test = 0.69645, df1 = 4, df2 = 213, p-value = 0.5952


$Instant

	H0: No instantaneous causality between: RGG and PSE RRP

data:  VAR object Model2
Chi-squared = 1.5007, df = 2, p-value = 0.4722


$Granger

	Granger causality H0: PSE do not Granger-cause RGG RRP

data:  VAR object Model2
F-Test = 1.9295, df1 = 4, df2 = 213, p-value = 0.1067
```

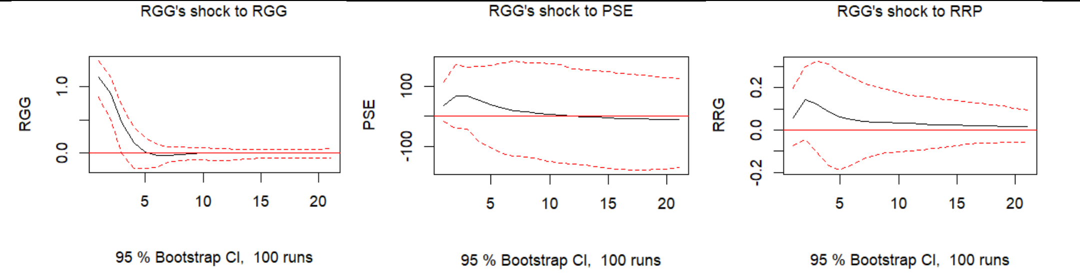
```
$Instant

	H0: No instantaneous causality between: PSE and RGG RRP

data:  VAR object Model2
Chi-squared = 1.4044, df = 2, p-value = 0.4955


$Granger

	Granger causality H0: RRP do not Granger-cause RGG PSE

data:  VAR object Model2
F-Test = 1.2118, df1 = 4, df2 = 213, p-value = 0.3068


$Instant

	H0: No instantaneous causality between: RRP and RGG PSE

data:  VAR object Model2
Chi-squared = 1.2202, df = 2, p-value = 0.5433
```

Analisis:

Hasil output memberikan insight dari tentang hubungan kausalitas antarvariabel dalam model VAR.

```r
#Mohon cek ulang title dari masing-masing gambar
RGGirf <- irf(Model2, impulse = "RGG", response = "RGG",
              n.ahead = 20, boot = TRUE)
plot(RGGirf, ylab = "RGG", main = "RGG's shock to RGG")
PSEirf <- irf(Model2, impulse = "RGG", response = "PSE",
```

```
                n.ahead = 20, boot = TRUE)
plot(PSEirf, ylab = "PSE", main = "RGG's shock to PSE")
RRPirf <- irf(Model2, impulse = "RGG", response = "RRP",
                n.ahead = 20, boot = TRUE)
plot(RRPirf, ylab = "RRG", main = "RGG's shock to RRP")
```

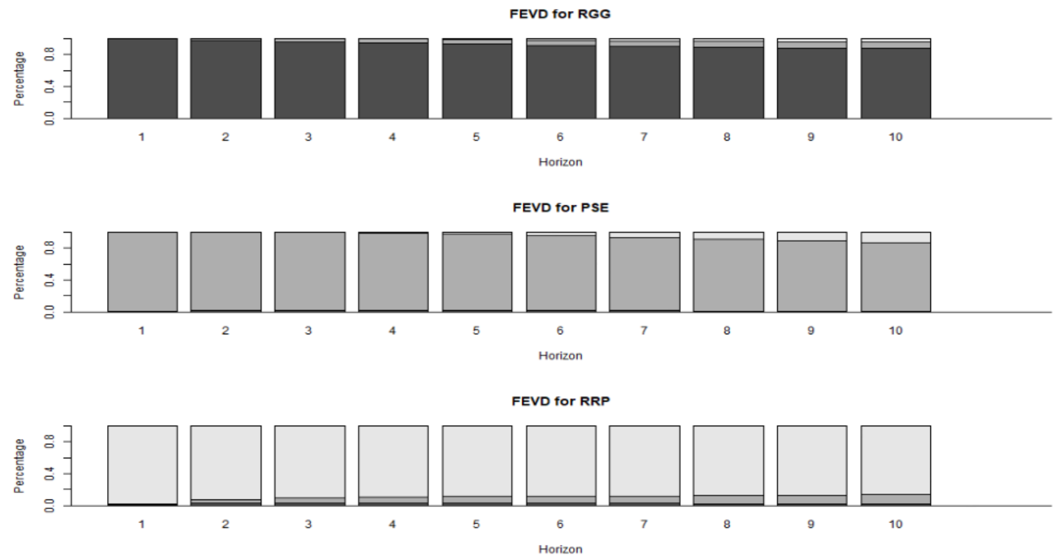


Analisis:

Membuat visualisasi dengan menampilkan plot dari setiap respons impuls yang dihasilkan.

```
FEVD1 <- fevd(Model2, n.ahead = 10)
FEVD1
plot(FEVD1)
```

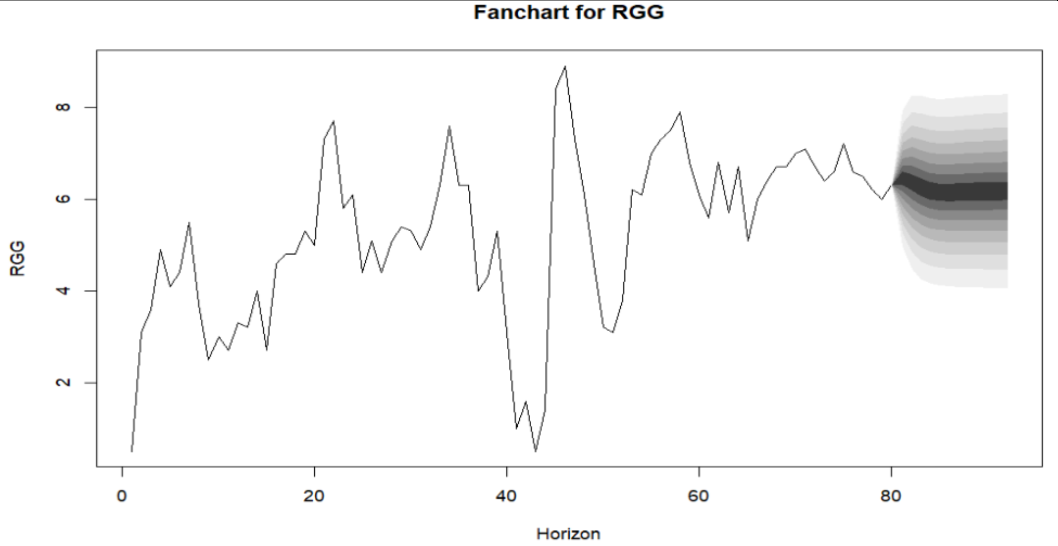
```
$RGG
            RGG        PSE          RRP
 [1,] 1.0000000 0.00000000 0.0000000000
 [2,] 0.9769583 0.02254835 0.0004933950
 [3,] 0.9580639 0.04128374 0.0006523491
 [4,] 0.9432537 0.05269181 0.0040545063
 [5,] 0.9296536 0.05913309 0.0112132924
 [6,] 0.9168601 0.06319506 0.0199448042
 [7,] 0.9051100 0.06638454 0.0285054313
 [8,] 0.8944453 0.06930271 0.0362519884
 [9,] 0.8847428 0.07211039 0.0431468155
[10,] 0.8758523 0.07481153 0.0493361978

$PSE
             RGG       PSE          RRP
 [1,] 0.01025606 0.9897439 0.000000e+00
 [2,] 0.01954674 0.9804137 3.959345e-05
 [3,] 0.02265997 0.9738961 3.443942e-03
 [4,] 0.02144627 0.9657441 1.280960e-02
 [5,] 0.01894557 0.9535505 2.750395e-02
 [6,] 0.01648376 0.9375440 4.597227e-02
 [7,] 0.01439471 0.9189449 6.666042e-02
 [8,] 0.01268410 0.8989683 8.834760e-02
 [9,] 0.01128645 0.8785306 1.101829e-01
[10,] 0.01013709 0.8582538 1.316091e-01

$RRP
              RGG        PSE       RRP
 [1,] 0.007792876 0.00809898 0.9841081
 [2,] 0.029368565 0.04833738 0.9222941
 [3,] 0.033719821 0.06531960 0.9009606
 [4,] 0.032107868 0.07563750 0.8922546
 [5,] 0.029580874 0.08392510 0.8864940
 [6,] 0.027397570 0.09105198 0.8815504
 [7,] 0.025705052 0.09724144 0.8770535
 [8,] 0.024412357 0.10265682 0.8729308
 [9,] 0.023407278 0.10745358 0.8691391
[10,] 0.022602188 0.11176318 0.8656346
```

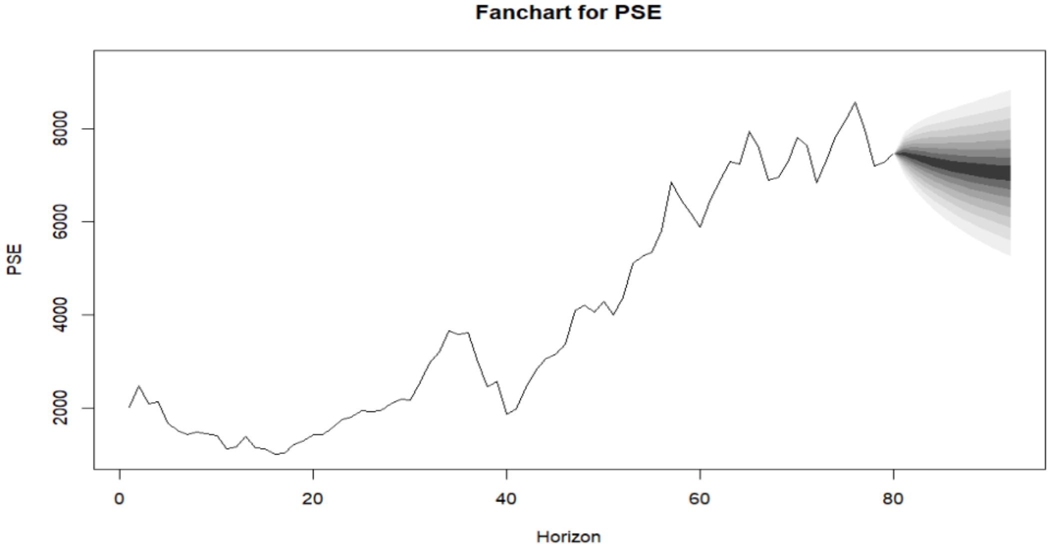


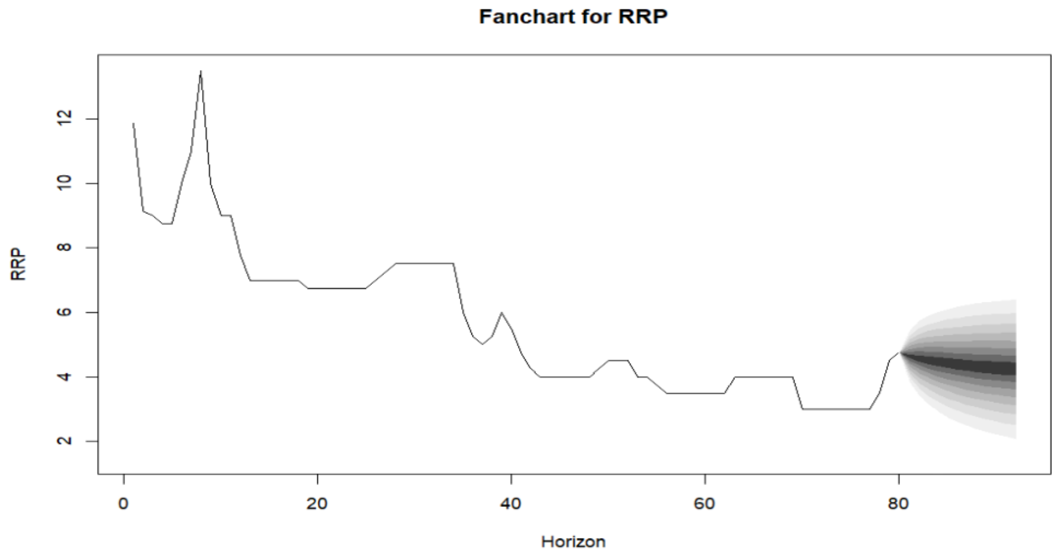
Analisis:

Menghitung dekomposisi varians kesalahan ramalan (FEVD) dari model VAR Anda yang disimpan dalam variabel Model2. Setelah menjalankan fungsi fevd(), menampilkan tabel dekomposisi varians kesalahan ramalan untuk setiap variabel dalam model VAR. Kemudian, membuat visualisasi plot dari hasil FEVD.

```
forecast <- predict(Model2, n.ahead = 12, ci = 0.95)
fanchart(forecast, names = "RGG", main = "Fanchart for RGG",
         xlab = "Horizon", ylab = "RGG")
fanchart(forecast, names = "PSE", main = "Fanchart for PSE",
         xlab = "Horizon", ylab = "PSE")
fanchart(forecast, names = "RRP", main = "Fanchart for RRP",
         xlab = "Horizon", ylab = "RRP")
forecast
```







$RGG

	fcst	lower	upper	CI
[1,]	6.458540	4.195892	8.721188	2.262648
[2,]	6.372503	3.474756	9.270250	2.897747
[3,]	6.254937	3.189548	9.320327	3.065389
[4,]	6.181682	3.077149	9.286216	3.104534
[5,]	6.152116	3.024933	9.279298	3.127183
[6,]	6.147756	2.997813	9.297699	3.149943
[7,]	6.153399	2.982051	9.324746	3.171347
[8,]	6.160898	2.970207	9.351590	3.190692
[9,]	6.167281	2.958954	9.375608	3.208327
[10,]	6.172146	2.947498	9.396794	3.224648
[11,]	6.175941	2.936047	9.415834	3.239893
[12,]	6.179157	2.924958	9.433357	3.254199

$PSE

	fcst	lower	upper	CI
[1,]	7454.190	6743.115	8165.265	711.0751
[2,]	7396.218	6325.662	8466.773	1070.5551
[3,]	7335.615	6001.446	8669.784	1334.1692
[4,]	7279.879	5729.914	8829.845	1549.9654
[5,]	7231.027	5493.176	8968.878	1737.8511
[6,]	7189.111	5281.291	9096.931	1907.8197
[7,]	7153.291	5088.051	9218.531	2065.2400
[8,]	7122.590	4909.538	9335.642	2213.0517
[9,]	7096.193	4743.278	9449.108	2352.9150
[10,]	7073.487	4587.673	9559.302	2485.8144
[11,]	7054.005	4441.634	9666.376	2612.3709
[12,]	7037.369	4304.365	9770.374	2733.0044

$RRP

	fcst	lower	upper	CI
[1,]	4.636844	3.3947461	5.878941	1.242097
[2,]	4.573716	2.7897351	6.357696	1.783981
[3,]	4.518559	2.3821392	6.654980	2.136420
[4,]	4.467933	2.0762599	6.859605	2.391673
[5,]	4.424737	1.8350941	7.014379	2.589643
[6,]	4.388361	1.6384256	7.138295	2.749935
[7,]	4.357049	1.4736315	7.240466	2.883417
[8,]	4.329379	1.3327045	7.326054	2.996675
[9,]	4.304492	1.2104616	7.398522	3.094030
[10,]	4.281918	1.1033836	7.460453	3.178535
[11,]	4.261392	1.0089216	7.513863	3.252471
[12,]	4.242724	0.9251155	7.560332	3.317608

Analisis:

Membuat ramalan dalam 12 bulan ke depan menggunakan model VAR lalu kemudian forecasting ini kemudian digunakan untuk membuat grafik untuk setiap variabel. Setiap grafik memiliki sumbu X yang menunjukkan jangka waktu masa depan sedangkan sumbu Y yang menunjukkan nilai ramalan untuk variabel yang sesuai.