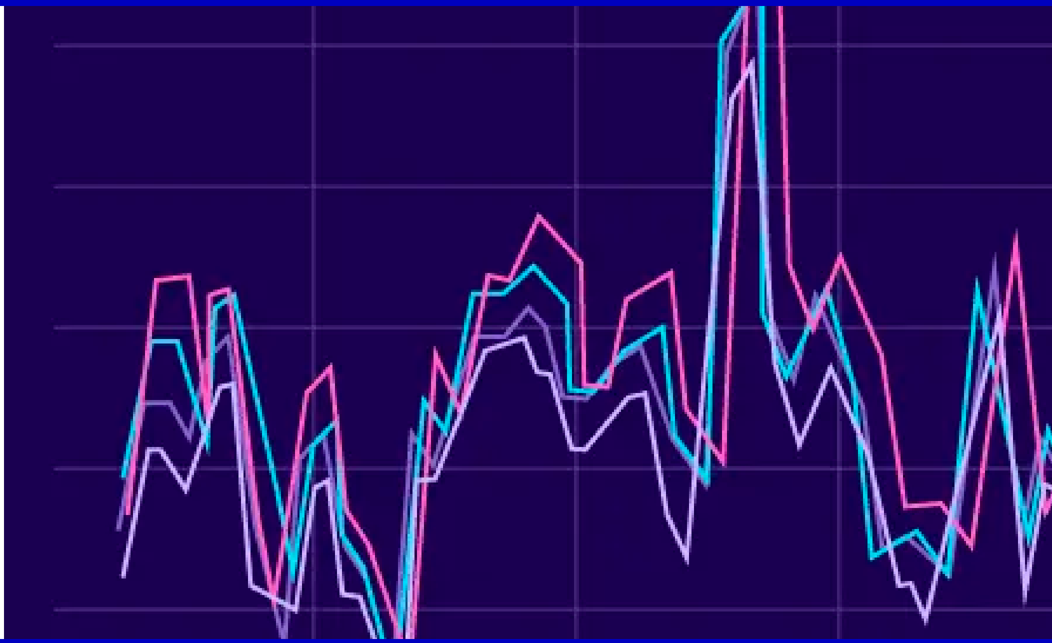
Analisis Deret Waktu
AK2281

MODUL IV
MODEL HETEROSKEDASTIS



Pengajar : Dr. Sandy Vantika, S.Si., M.Si.
Koordinator : Leonardo Valentino Kosasih, S.Aktr.
Kontak : (+62) 823 8599 5958

Modul Praktikum Analisis Deret Waktu - AK2281 terdiri atas 5 buah modul sebagai berikut :

1. Modul 1 : Konsep Dasar Deret Waktu
2. Modul 2 : Model Data Tak Stasioner
3. Modul 3 : Model Musiman
4. Modul 4 : Model Heteroskedastik
5. Modul 5 : Pengayaan

Tim Praktikum AK2281

Koordinator Praktikum : Leonardo V. Kosasih, S.Aktr.

Tim Penyusun Modul Praktikum :

Ang Ditra Alif Pradana	10120046	Matthew Alfarazh	10820021
Feby Yolanda	10819028	Pamella Cathryn	10820033
Ferdinan Gratius Budisatya	10819041	Jeremy	10820034
Jevan Christopher Aryento	10820010	Aloysius Vincent	10820038
Shelly Delfiani	10820014	Kevin Christ Aditya	10820039
Binsar Gunadi Simbolon	10820017	Shafina Aulia Kusuma Putri	10820049

Desain sampul oleh : Matthew Alfarazh - 10820021

Daftar Isi

Modul 4

Tujuan:

1. Memahami model deret waktu heteroskedastik
2. Mengindentifikasi deret waktu heteroskedastik

Pada modul kali ini ada beberapa *library* yang akan digunakan yakni sebagai berikut :

```
# Untuk membaca data dengan format csv
library(readr)

# Untuk membersihkan data
library(tidyr)

# Untuk mengubah data menjadi Time Series,
# membuat plot ACF, plot PACF, Model ARIMA dan ADF Test
library(tseries)

# Untuk melihat signifikansi koefisien dari parameter
library(lmtest)

# Untuk memprediksi data dari model
library(forecast)

# Untuk memeriksa adanya sifat heteroskedastik pada data
library(FinTS)

# Uji Anderson Darling
library(nortest)

# Model GARCH
library(fGarch)
```

Sedangkan data yang akan diolah adalah data harga penutupan Doge Coin dari bulan April tahun 2021 hingga akhir bulan November tahun 2021 yang diperoleh dari Yahoo Finance.

TEKAN UNTUK MENGUNDUH DATA

Alur Permodelan

Secara umum, alur permodelan dapat dibagi menjadi berikut :

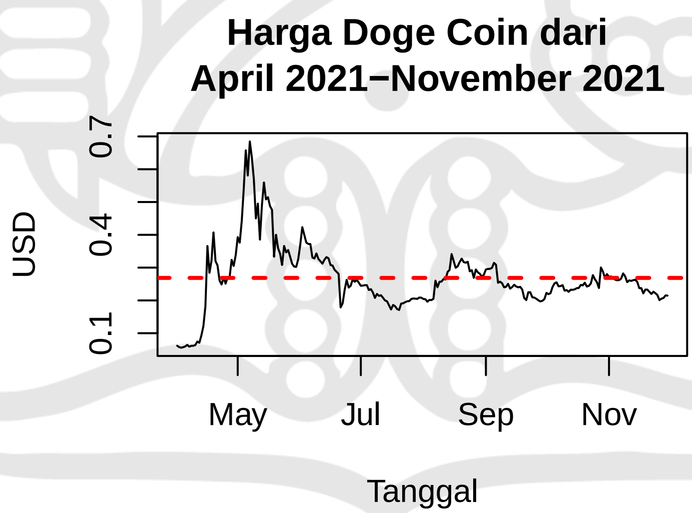
1. EDA (Exploratory Data Analysis) atau Analisis Data
2. Mempersiapkan data
3. Identifikasi Model
4. Estimasi Parameter
5. Uji Diagnostik
6. *Forecasting* / Peramalan

1 EDA

Pada bagian ini akan dilihat grafik data, dan juga stastik deskriptif dari data

```
# Memanggil Data
data<- read_csv("DOGE-USD.csv",
    col_types = cols(Date = col_date(format = "%m/%d/%Y")))

plot(data$Date,data$Close, type = 'l',
     main = 'Harga Doge Coin dari \n April 2021-November 2021',
     xlab = 'Tanggal', ylab= 'USD')
# Membuat garis rataan
abline(h=mean(data$Close),lwd=2,
       lty = 2, col ='red')
```




```
# Statistik Deskriptif
summary(data$Close)
```

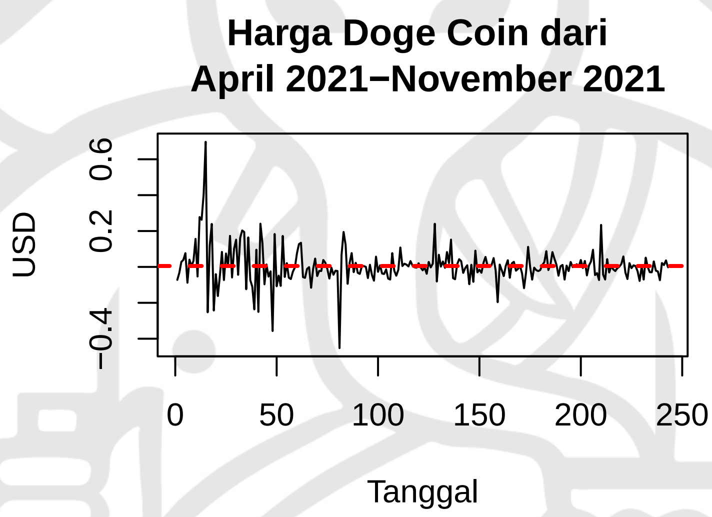
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0558  0.2184  0.2547  0.2686  0.3066  0.6848
```

Dapat dilihat terdapat kenaikkan yang besar pada bulan April 2021 dan lagi pada bulan Mei 2021 yang disusul dengan penurnan harga hingga April 2022. Data yang akan dimodelkan adalah data log-return yang didefinisikan dengan :

$$r_t = \log \frac{y_{t+1}}{y_t} = \log y_{t+1} - \log y_t, \quad t = 1, \ldots 243$$

Sehingga diperoleh

```
ts_data <- ts(diff(log(data$Close)))
plot(ts_data, type = 'l',
     main = 'Harga Doge Coin dari \n April 2021-November 2021',
     xlab = 'Tanggal', ylab= 'USD')
# Membuat garis rataan
abline(h=mean(ts_data),lwd=2,lty = 2, col ='red')
```

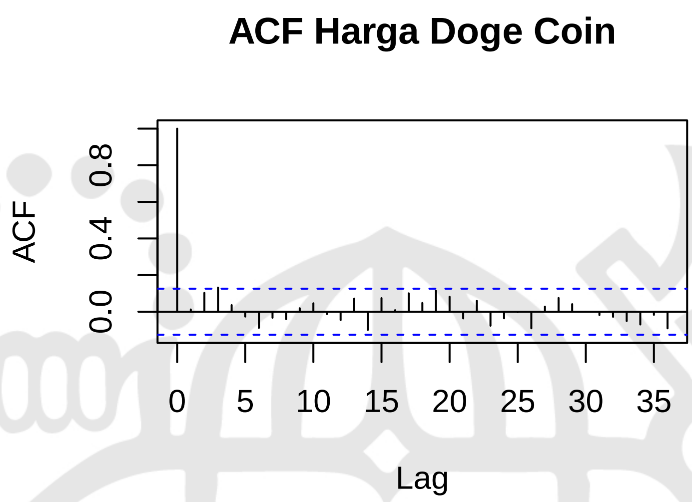



```
# Statistik Deskriptif
summary(ts_data)
```

```
##        Min.     1st Qu.      Median        Mean     3rd Qu.        Max.
## -0.4523407 -0.0391006 -0.0009946  0.0051128  0.0305130  0.6971347
```

Jika dimodelkan dengan model ARIMA maka dapat diperoleh model sebagai berikut :

```
acf(ts_data, main = 'ACF Harga Doge Coin', lag.max = 36)
```

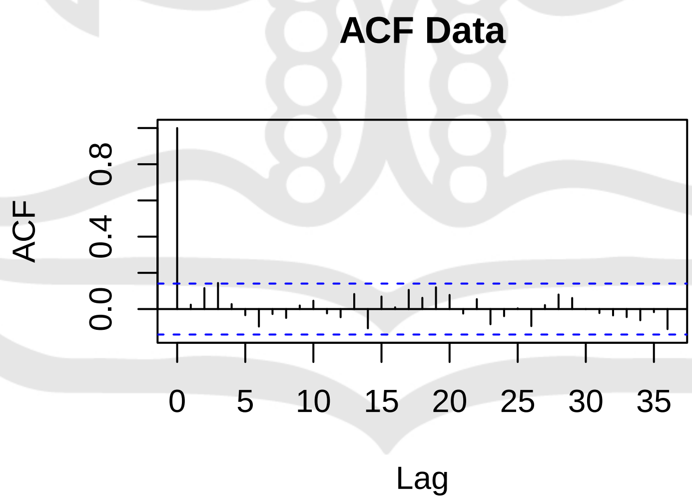



```
adf.test(ts_data)
```

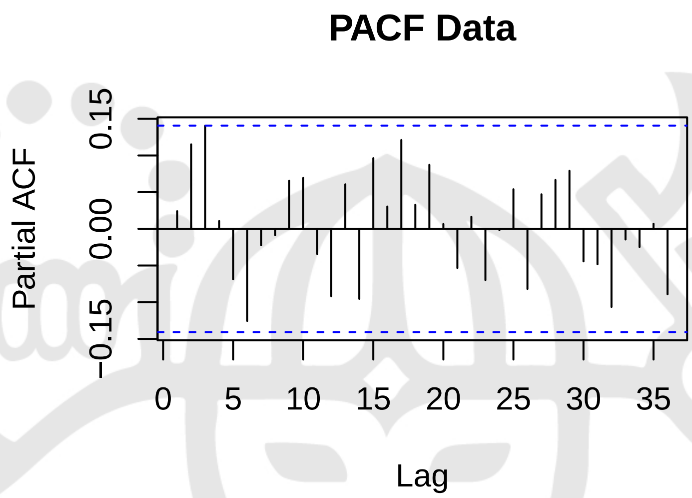
```
## Warning in adf.test(ts_data): p-value smaller than printed p-value
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_data
## Dickey-Fuller = -6.3393, Lag order = 6, p-value = 0.01
## alternative hypothesis: stationary
```

Dapat dilihat bahwa hasil uji ADF diperoleh $p - value < \alpha = 5$. Maka data tidak perlu diferensiasi dan dapat langsung ditentukan modelnya.

```
ts_train <- ts_data[1:(floor(0.8*length(ts_data)))]
ts_validation <- ts_data[ceiling(0.8*length(ts_data)): length(ts_data)]
acf(ts_train, main = 'ACF Data', lag.max = 36)
```

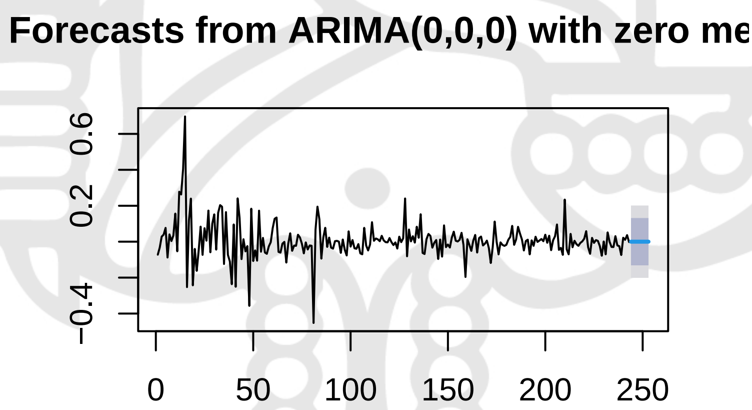


```
pacf(ts_train, main = 'PACF Data', lag.max = 36)
```



apat dilihat bahwa tidak ada model yang cocok untuk memodelkan data. Misalkan akan dimodelkan dengan AIC yang terendah maka dengan fungsi `auto.arima` diperoleh hasil berikut :

```
mod <- auto.arima(ts_data, max.q = 0,allowmean = TRUE, allowdrift = TRUE, method = "ML")
fc <- forecast(ts_data, model=mod, h=10)
plot(fc)
```

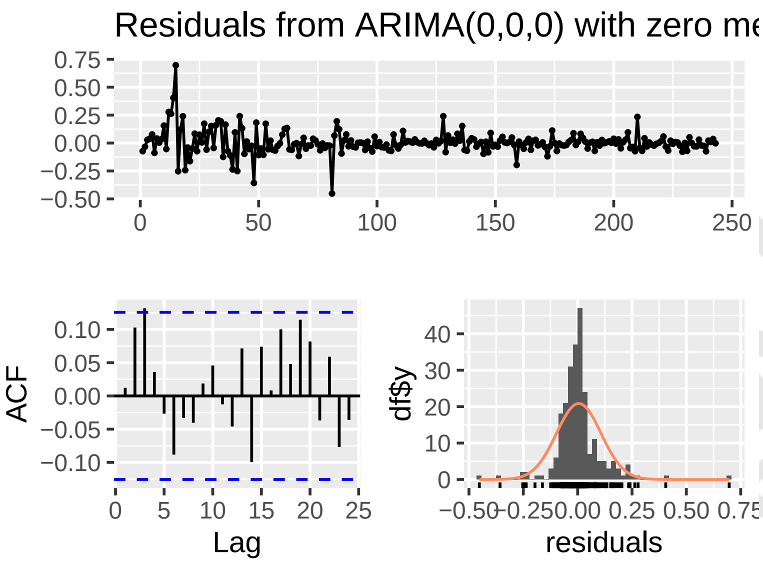


Jurnal Praktikum 1

Apakah data yang mengikuti model ARIMA (0,0,0) pada data return adalah suatu hal yang tidak wajar ? Berikan alasan anda !

Pada grafik **forecast** dapat dilihat bahwa prediksi data seperti garis linear saja yang tidak menangkap efek keacakkan dari data. Tentu saja hal ini bukanlah hal yang diinginkan dari model.

```
checkresiduals(mod)
```



Gambar 1: Grafik Uji Diagnostik

```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,0,0) with zero mean
## Q* = 10.735, df = 10, p-value = 0.3785
##
## Model df: 0.   Total lags used: 10
```

Dapat dilihat dari grafik histogram bahwa data tidak berdistribusi normal namun dari grafik ACF dapat muncul dugaan bahwa galat tidak saling bebas, hal ini didukung uji statistik Ljung-Box yang nilai $p - value > \alpha$ sehingga diperoleh bahwa data tidak saling bebas.

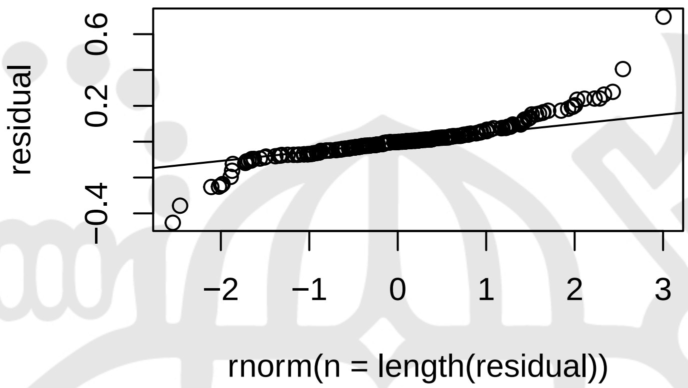
2 Model Heteroskedastis

Berbeda dengan permodelan yang telah dipelajari sebelumnya, model heteroskedastis akan memodelkan variansi bersyarat pada data. Hal ini dapat dilakukan secara **independen** (tanpa perlu menentukan model ARIMA ataupun SARIMA dari data) dan dapat pula **digabung** dengan model yang sudah ada. Model Heteroskedastis ini digunakan untuk mengatasi galat/residual data yang gagal dalam uji diagnostik.

Akan dikaji lebih lanjut apakah residual dari data mengikuti distribusi normal

```
residual <- residuals(mod,standardize=T)

# QQPlot
set.seed(1602)
qqplot(rnorm(n = length(residual)), residual)
qqline(residual)
```




```
# Saphiro Wilk Test
shapiro.test(residual)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residual
## W = 0.8397, p-value = 3.885e-15
```

```
# Anderson Darling
ad.test(residual)
```

```
##
##  Anderson-Darling normality test
##
## data:  residual
## A = 10.06, p-value < 2.2e-16
```

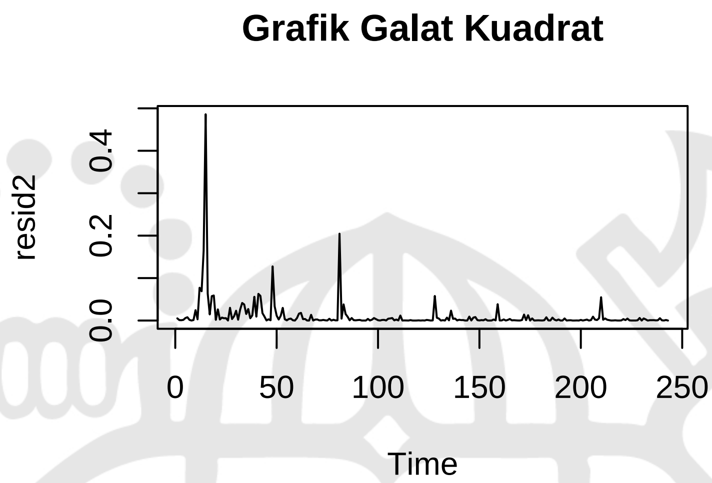
Dapat dilihat dari hasil uji Shapiro-Wilk dan Anderson Darling $p-value < \alpha$ sehingga dapat disimpulkan bahwa data residual **tidak berdistribusi normal**. Hal ini juga didukung dari QQ - plot yang tidak berada pada garis miring.

Berikutnya akan diperiksa adanya heteroskedastis bersyarat atau yang akan disebut dengan uji ARCH pada residu. Statistik uji yang dapat dilakukan adalah uji Ljung-Box dan uji *Lagrangre Multiplier*, data yang akan diuji adalah data residu kuadrat.

Jurnal Praktikum 2

Tulislah H_0 dan H_a dari uji Lagrangre Multiplier

```
resid2 <- residual^2
plot(resid2, main = "Grafik Galat Kuadrat", type = 'l')
```




```
Box.test(resid2, lag = 12)
```

```
##
##  Box-Pierce test
##
## data:  resid2
## X-squared = 54.964, df = 12, p-value = 1.837e-07
```

```
ArchTest(residual)
```

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  residual
## Chi-squared = 35.617, df = 12, p-value = 0.0003732
```

Jurnal Praktikum 3

Mengapa pada uji Ljung Box data harus dikuadratkan sedangkan `ArchTest` *tidak ?*

Dari uji statistik yang dilakukan dilihat bahwa $p - value < \alpha$ sehingga dapat disimpulkan bahwa data residual memiliki efek ARCH.

Berikutnya akan ditentukan model yang cocok pada data residual.

3 Penentuan Model

Secara umum, model GARCH (p, q) dapat ditulis dengan persamaan berikut:

$$\sigma^2_{t|t-1} = \omega + \beta_1\sigma^2_{t-1|t-2} + \cdots + \beta_p\sigma^2_{t-p|t-p-1} + \alpha_1 r^2_{t-1} + \alpha_2 r^2_{t-2} + \cdots + \alpha_q r^2_{t-q} \tag{3.1}$$

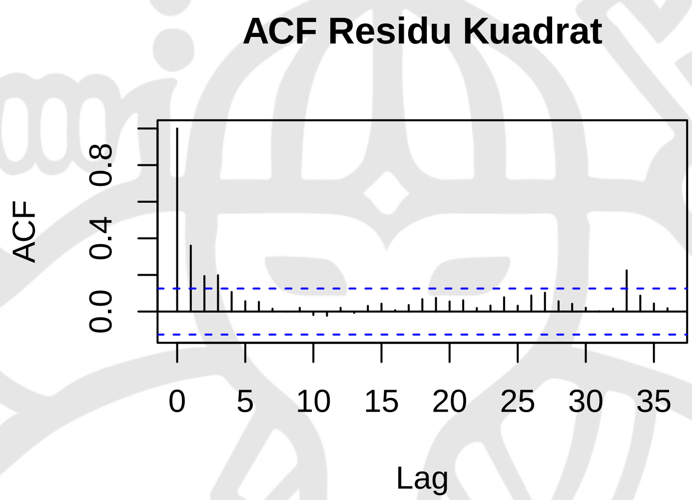
dengan $\sigma^2_{t|t-1}$ adalah variansi bersyarat dari data r_t.
Catatan:

1. Terdapat perbedaan pendifinisian model GARCH pada *library* tertentu. Ada yang menggunakan pendefinisian GARCH(p, q) dan ada juga yang menggunakan GARCH(q, p).

2. Persamaan GARCH(p, q) dapat ditulis menjadi suatu bentuk yang serupa dengan model ARMA$(\max\{p, q\}, p)$.

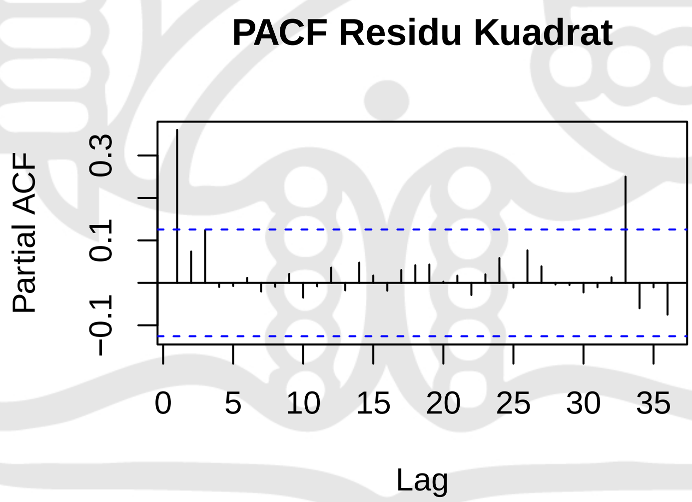
Dalam menentukan model yang cocok, grafik ACF akan menentukan parameter q dari model GARCH (p, q). Sedangkan grafik PACF akan menentukan parameter p dari GARCH (p, q).

```
acf(resid2, main = 'ACF Residu Kuadrat', lag.max = 36)
```



Gambar 2: ACF Residu

```
pacf(resid2, main = 'PACF Residu Kuadrat', lag.max = 36)
```



Gambar 3: ACF Residu

Dari grafik diperoleh model yang cocok adalah :

1. GARCH(3,1)
2. GARCH(3,0)
3. GARCH(1,1) (Dipilih karena parsimoni)

```r
mod_h1 <- garchFit(residual~garch(3,1), data = residual,
                   trace=F, include.mean = F)
summary(mod_h1)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = residual ~ garch(3, 1), data = residual, include.mean = F,
##     trace = F)
##
## Mean and Variance Equation:
##  data ~ garch(3, 1)
## <environment: 0x000001dfd992df20>
##  [data = residual]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##      omega      alpha1      alpha2      alpha3       beta1
## 0.00166008  0.23556180  0.00000001  0.00000001  0.58009101
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## omega  1.660e-03         NaN      NaN      NaN
## alpha1 2.356e-01         NaN      NaN      NaN
## alpha2 1.000e-08         NaN      NaN      NaN
## alpha3 1.000e-08         NaN      NaN      NaN
## beta1  5.801e-01         NaN      NaN      NaN
##
## Log Likelihood:
##  255.3331    normalized:  1.050754
##
## Description:
##  Mon May 15 18:06:49 2023 by user: leona
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  1161.548  0
##  Shapiro-Wilk Test  R    W      0.8679432 1.231483e-13
##  Ljung-Box Test     R    Q(10)  6.650948  0.7579353
```

```
##  Ljung-Box Test     R    Q(15)  8.840947  0.8856858
##  Ljung-Box Test     R    Q(20)  15.40214  0.7529533
##  Ljung-Box Test     R^2  Q(10)  0.8633232 0.9999127
##  Ljung-Box Test     R^2  Q(15)  1.802345  0.9999852
##  Ljung-Box Test     R^2  Q(20)  2.324849  0.9999996
##  LM Arch Test       R    TR^2   0.9625955 0.9999886
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -2.060355 -1.988481 -2.061179 -2.031405
```

```
mod_h2 <- garchFit(residual~garch(3,0), data = residual,
                   trace=F, include.mean = F)
summary(mod_h2)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = residual ~ garch(3, 0), data = residual, include.mean = F,
##     trace = F)
##
## Mean and Variance Equation:
##  data ~ garch(3, 0)
## <environment: 0x000001dfdc8c6158>
##  [data = residual]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##     omega     alpha1     alpha2     alpha3
## 0.0044384  0.2696740  0.2445647  0.0075717
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## omega   0.0044384   0.0005636    7.876 3.33e-15 ***
## alpha1  0.2696740   0.1101895    2.447   0.0144 *
## alpha2  0.2445647   0.1234292    1.981   0.0475 *
## alpha3  0.0075717   0.0463946    0.163   0.8704
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
```

```
##  254.9702    normalized:  1.04926
##
## Description:
##  Mon May 15 18:06:49 2023 by user: leona
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  1065.813  0
##  Shapiro-Wilk Test  R    W      0.8704871 1.722943e-13
##  Ljung-Box Test     R    Q(10)  7.455983  0.6818092
##  Ljung-Box Test     R    Q(15)  10.6772   0.7751238
##  Ljung-Box Test     R    Q(20)  17.02623  0.6512701
##  Ljung-Box Test     R^2  Q(10)  0.8969944 0.9998957
##  Ljung-Box Test     R^2  Q(15)  2.11651   0.9999569
##  Ljung-Box Test     R^2  Q(20)  2.574736  0.9999989
##  LM Arch Test       R    TR^2   1.034349  0.9999829
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -2.065598 -2.008099 -2.066129 -2.042438
```

```
mod_h3 <- garchFit(residual~garch(1,1), data = residual,
                   trace=F, include.mean = F)
summary(mod_h3)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = residual ~ garch(1, 1), data = residual, include.mean = F,
##     trace = F)
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x000001dfd98ca0d8>
##  [data = residual]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##     omega     alpha1      beta1
## 0.0016872  0.2396254  0.5734003
##
## Std. Errors:
##  based on Hessian
```

```
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## omega  0.0016872   0.0005791    2.913  0.00358 **
## alpha1 0.2396254   0.0735329    3.259  0.00112 **
## beta1  0.5734003   0.1077911    5.320 1.04e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  255.6761    normalized:  1.052165
##
## Description:
##  Mon May 15 18:06:49 2023 by user: leona
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  1154.141  0
##  Shapiro-Wilk Test  R    W      0.8685548 1.334475e-13
##  Ljung-Box Test     R    Q(10)  6.705809  0.7528957
##  Ljung-Box Test     R    Q(15)  8.913776  0.8819825
##  Ljung-Box Test     R    Q(20)  15.44708  0.7502798
##  Ljung-Box Test     R^2  Q(10)  0.8316636 0.9999266
##  Ljung-Box Test     R^2  Q(15)  1.769761  0.9999869
##  Ljung-Box Test     R^2  Q(20)  2.283284  0.9999996
##  LM Arch Test       R    TR^2   0.9397625 0.99999
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -2.079639 -2.036515 -2.079939 -2.062269
```

Dapat dilihat bahwa model GARCH(1,1) memiliki AIC yang paling kecil dengan parameter yang paling sedikit sehingga akan digunakan model GARCH(1,1) untuk memodelkan data. Sehingga diperoleh persamaan model sebagai berikut:

$$\begin{aligned}\sigma^2_{t|t-1} &:= \omega + \beta_1\sigma^2_{t-1|t-2} + \alpha_1 r^2_{t-1} \\ &= 0.0016872 + 0.5734003\sigma^2_{t-1|t-2} + 0.2396254 r^2_{t-1}\end{aligned} \tag{3.2}$$

Sehingga diperoleh model gabungan ARMA(0,0) dan GARCH(1,1) sebagai berikut:

$$\begin{aligned}r_t &:= \varepsilon_t = \sigma_{t|t-1} z_t (\text{Mengapa?}) \\ &= \sqrt{0.0016872 + 0.5734003\sigma^2_{t-1|t-2} + 0.2396254 r^2_{t-1}}\, z_t\end{aligned} \tag{3.3}$$

dengan $z_t \sim N(0,1)$. (Pelajari juga modul `rugarch` untuk memodelkan model GARCH).

4 *Forecast*

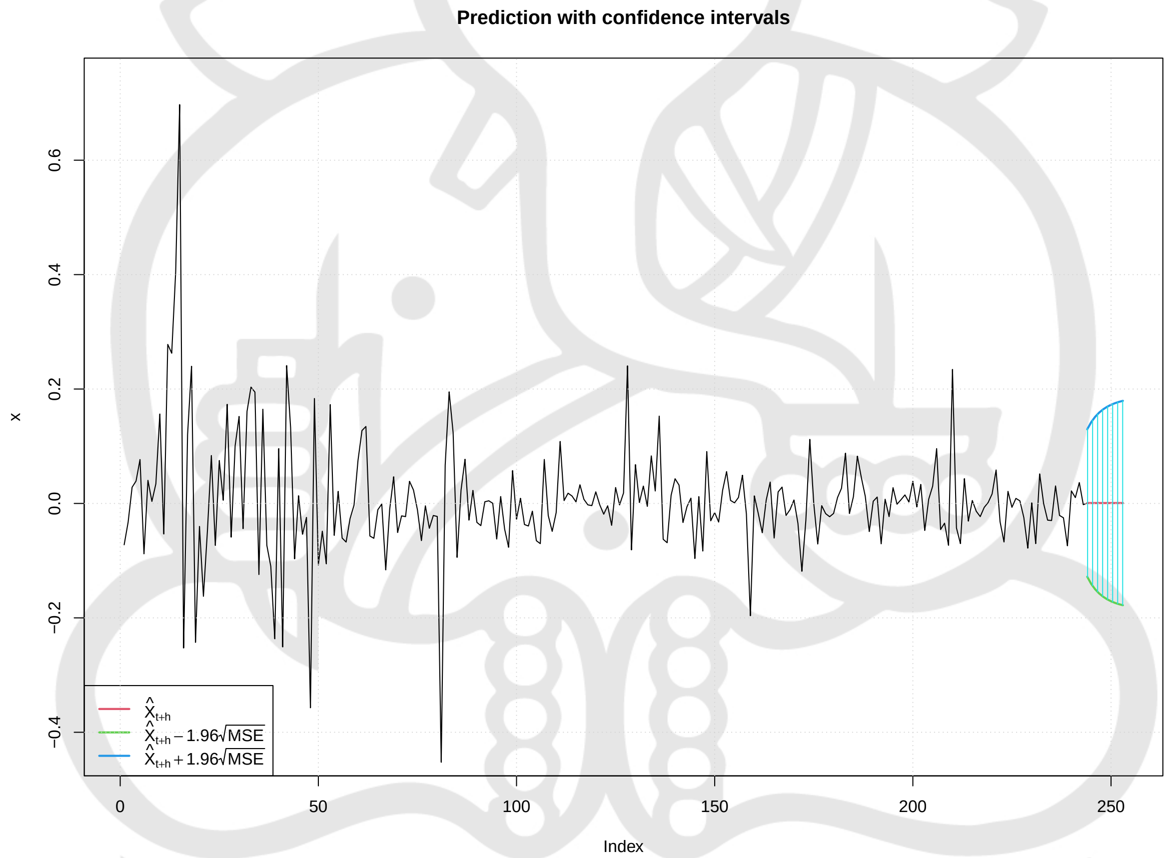
Pada bagian ini akan dilakukan *forecasting* dengan model gabungan ARMA(0,0) dengan GARCH(1,1)

```
m1 <- garchFit(ts_data ~ arma(0,0)+garch(1,1),data=ts_data,trace=F)
summary(m1)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ts_data ~ arma(0, 0) + garch(1, 1), data = ts_data,
##     trace = F)
##
## Mean and Variance Equation:
##  data ~ arma(0, 0) + garch(1, 1)
## <environment: 0x000001dfdc0e7f88>
##  [data = ts_data]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1       beta1
## 0.0007337  0.0016864  0.2409106  0.5726586
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     0.0007337   0.0050465    0.145  0.88440
## omega  0.0016864   0.0005795    2.910  0.00362 **
## alpha1 0.2409106   0.0745684    3.231  0.00123 **
## beta1  0.5726586   0.1081081    5.297 1.18e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  255.6867    normalized:  1.052209
##
## Description:
##  Mon May 15 18:06:49 2023 by user: leona
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
```

```
##  Jarque-Bera Test   R    Chi^2  1145.196   0
##  Shapiro-Wilk Test  R    W      0.8687121 1.362394e-13
##  Ljung-Box Test     R    Q(10)  6.7552    0.7483357
##  Ljung-Box Test     R    Q(15)  8.953567  0.8799327
##  Ljung-Box Test     R    Q(20)  15.49536  0.7473964
##  Ljung-Box Test     R^2  Q(10)  0.828094  0.9999281
##  Ljung-Box Test     R^2  Q(15)  1.790955  0.9999858
##  Ljung-Box Test     R^2  Q(20)  2.298968  0.9999996
##  LM Arch Test       R    TR^2   0.9423357 0.9999898
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -2.071495 -2.013996 -2.072026 -2.048335
```

```
fc_h <- predict(m1, n.ahead = 10, plot=TRUE, conf=.95, nx = length(ts_data))
```




```
fc_h
```

```
##     meanForecast  meanError standardDeviation lowerInterval upperInterval
## 1  0.0007336983 0.06574589        0.06574589    -0.1281259     0.1295933
## 2  0.0007336983 0.07213217        0.07213217    -0.1406428     0.1421102
## 3  0.0007336983 0.07693778        0.07693778    -0.1500616     0.1515290
## 4  0.0007336983 0.08063646        0.08063646    -0.1573109     0.1587783
## 5  0.0007336983 0.08352486        0.08352486    -0.1629720     0.1644394
## 6  0.0007336983 0.08580306        0.08580306    -0.1674372     0.1689046
```

```
## 7  0.0007336983 0.08761285         0.08761285    -0.1709843     0.1724517
## 8  0.0007336983 0.08905811         0.08905811    -0.1738170     0.1752844
## 9  0.0007336983 0.09021684         0.09021684    -0.1760881     0.1775555
## 10 0.0007336983 0.09114869         0.09114869    -0.1779145     0.1793819
```

Berbeda dengan model yang sebelumnya dapat dilihat dari selang kepercayaannya bahwa model GARCH jauh dapat memodelkan variansi dari data.

5 Kesimpulan

Seringkali model yang digunakan tidak dapat memodelkan variansi / risiko di masa depan. Sehingga diperlukan model baru yang dapat membantu memprediksi variansi dari prakiraan di masa depan.

6 Pengayaan: Valuasi Harga Opsi Dengan Simulasi Monte Carlo

Pada bagian ini akan dilakukan simulasi harga *cryptocurrency* yang akan digunakan untuk melakukan valuasi harga opsi di suatu waktu di masa depan dengan melakukan simulasi monte carlo. Harga sekuritas akan dimodelkan dengan model yang telah dibuat sebelumnya dan harga opsi akan divaluasi menggunakan persamaan *Black-Scholes*. Tujuan dari bagian ini hanya untuk memperkenalkan secara singkat dan praktis mengenai aplikasi dari suatu model deret waktu. Pembaca sangat disarankan untuk mempelajari lebih lanjut bagian-bagian yang ditulis pada bagian ini.

6.1 Simulasi Monte Carlo

Simulasi monte carlo adalah suatu simulasi yang menggunakan bilangan acak. Ide dari metode ini adalah dengan membangkitkan bilangan acak dalam jumlah besar untuk menaksir secara empiris nilai statistik yang diinginkan seperti, rataan, standar deviasi, kuartil atas ataupun nilai statistik lainnya. Pada bagian ini akan dilakukan 1 simulasi dengan ukuran simulasi sebesar 1000. Untuk menaksir nilai harga *cryptocurrency* untuk r_t dengan $t = 244, 245, \ldots, 253$.

```
# Inisialisasi variabel untuk simulasi
data_return <- diff(log(data$Close))
sd_return <- sd(data_return)
omega <- mod_h3@fit$matcoef[1]
alpha_1 <- mod_h3@fit$matcoef[2]
beta_1 <- mod_h3@fit$matcoef[3]
garch_11 <- function(sd_data_prev, data_prev){
  sigma_t <- sqrt(omega + alpha_1 * data_prev^2 + beta_1 * sd_data_prev^2)
  data_t <- sigma_t*rnorm(n = 1, mean = 0, sd = 1)
  return (list(sigma_new = sigma_t, data_new = data_t))
}
```

```
hasil_simulasi <- matrix(0, 1000, 10) # Menyimpan data pada matriks
for (ukuran_simulasi in 1:1000){
  data_sim <- tail(data_return, n = 1)
  sd_sim <- sd(data_return)
  for (t in 244:253){
```

```
    garch_sim <- garch_11(sd_sim, tail(data_sim, n =1))
    sd_sim <- garch_sim$sigma_new
    data_sim <- c(data_sim, garch_sim$data_new)
  }
  dummy <- tail(data_sim, n = 10)
  hasil_simulasi[ukuran_simulasi, 1:10] <- dummy
  }
```

6.2 Analisis Simulasi Monte Carlo

Pada bagian ini akan dihitung beberapa nilai statistik yang dipandang penting pada simulasi, di antaranya adalah:

1. Rataan
2. Standar Deviasi
3. Median
4. Persentil ke-90
5. Persentil ke-95

```
statistik <- data.frame(row.names = c('Rataan', 'Stdev', 'Median',
                                      'Persentil90', 'Persentil95'))
for (i in 1:10){
  statistik[1, i] <- mean(hasil_simulasi[1:1000, i])
  statistik[2, i] <- sd(hasil_simulasi[1:1000, i])
  statistik[3, i] <- median(hasil_simulasi[1:1000, i])
  statistik[4, i] <- quantile(hasil_simulasi[1:1000, i], probs = 0.9)
  statistik[5, i] <- quantile(hasil_simulasi[1:1000, i], probs = 0.95)
}
statistik
```

```
##                       V1           V2            V3            V4           V5
## Rataan      0.0020050772 -0.002953175  0.0007963533  1.202505e-03 -0.003845682
## Stdev       0.0897921421  0.088038633  0.0865077545  9.038229e-02  0.092419711
## Median      0.0007004282 -0.001211945 -0.0015595796 -1.711865e-05 -0.004651828
## Persentil90 0.1168847628  0.109742247  0.1104319476  1.125334e-01  0.104864005
## Persentil95 0.1459208938  0.138545898  0.1427410810  1.471652e-01  0.134582299
##                      V6           V7          V8          V9          V10
## Rataan      0.002686294 -0.006780777 0.005802096 0.005677739 0.002045073
## Stdev       0.098818356  0.097557928 0.095785995 0.092807908 0.090504527
## Median      0.003062452 -0.005730442 0.004125164 0.005096664 0.004470458
## Persentil90 0.119442910  0.101705281 0.118646803 0.117818796 0.110737068
## Persentil95 0.154463949  0.150568049 0.154492878 0.149855300 0.153118269
```

Nantinya akan digunakan nilai-nilai pada tabel di atas untuk melakukan valuasi opsi.

6.3 Persamaan *Black-Scholes*

Opsi adalah suatu instrumen keuangan yang nilainya bergantung pada suatu produk sekuritas lainnya, contohnya adalah saham atau pada pemodelan kali ini adalah suatu *cryptocurrency* yaitu Doge Coin. Pada pembahasan ini, akan dibahas mengenai opsi saham khususnya *European Option.* Terdapat 2 jenis opsi yaitu opsi **call** dan opsi **put**. Opsi call adalah suatu hak untuk menjual suatu saham dengan harga tertentu yang disebut dengan *strike price* yang dinotasikan dengan K.
Dalam penilaian suatu opsi, digunakan suatu persamaan yang dikenal dengan *Black-Scholes Formula* yang didefinisikan sebagai berikut:

$$C(S_t, t) = N(d_1)S_t - N(d_2)K\exp(-r(T-t)) \tag{6.1}$$

dengan

1. S_t: harga Doge Coin pada saat t;
2. $N(\cdot)$: fungsi distribusi kumulatif normal baku;
3. $d_1 = \frac{\ln\left(\frac{S_t}{K}\right)+\left(r+\frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}$;
4. $d_2 = d1 - \sigma\sqrt{T-t}$;
5. σ: volatilitas atau standar deviasi dari harga Doge Coin, pada data diperoleh standar deviasi sebesar 0.09835049;
6. K: *Strike Price*, diasumsikan $K = 0.03$;
7. T: waktu tenor, diasumsikan $T = 253$;
8. r: *risk-free interest rate* atau suku bunga tak berisiko, dan diasumsikan $r = 1\%$

Maka dapat diperoleh nilai harga opsi call untuk $t = 1$ adalah sebagai berikut:

$$\begin{aligned}
d_1 &= \frac{\ln\left(\frac{0.0620}{0.03}\right) + \left(0.01 + \frac{0.09835049^2}{2}\right)\left(\frac{253-1}{365.25}\right)}{0.09835049\sqrt{\frac{253-1}{365.25}}} = 9.052372\\
d_2 &= 1.452339 - 0.09835049\sqrt{\frac{253-1}{365.25}} = 8.97068\\
C(0.0620, 1) &= N(9.052372)0.0620 - N(8.97068)(0.03)\exp\left(-0.01\frac{(253-1)}{365.25}\right)\\
&\approx 0.062 - 0.03(0.9931244) = 0.03220627
\end{aligned}$$

Berikutnya akan dilakukan valuasi dari harga opsi untuk $t = 244, 245, \dots, 253$.
Perhatikan bahwa data yang disimulasikan berupa selisih antara nilai logaritma natural dari harga Doge Coin pada waktu t dengan nilai logaritma natural dari harga Doge Coin pada waktu $t-1$. Karena dalam valuasi opsi nanti diperlukan harga dari Doge Coin, maka akan dilakukan transformasi terlebih dahulu. Perhatikan bahwa

$$\begin{aligned}
\hat{r}_{244} &= \log\hat{y}_{245} - \log y_{244}\\
\Leftrightarrow \hat{y}_{245} &= \exp(\hat{r}_{244} + \log y_{244})\\
&= \exp(0.0020050772 + \log 0.214715) = 0.2151459
\end{aligned}$$

Sehingga dengan menggunakan cara yang sama dengan sebelumnya dapat diperoleh

$$d_1 = \frac{\ln\left(\frac{0.2151459}{0.03}\right) + \left(0.01 + \frac{0.09835049^2}{2}\right)\left(\frac{253-244}{365.25}\right)}{0.09835049\sqrt{\frac{253-244}{365.25}}} = \text{diserahkan kepada pembaca sebagai latihan}$$

$$d_2 = \text{diserahkan kepada pembaca sebagai latihan}$$

$$C(0.2151459, 244) = \text{diserahkan kepada pembaca sebagai latihan}$$

Catatan: Tentu saja anda dianjurkan untuk dapat membuat kode untuk menghitung opsi tersebut agar tidak menyulitkan anda dalam mengerjakan jurnal praktikum berikut.

Jurnal Praktikum 4

Hitunglah harga Opsi untuk $t = 245, 246, \ldots, 253$*! Hint: anda cukup mengganti nilai* S_t *dan* t *untuk tiap valuasinya.*

7 Daftar Pustaka

Cryer, J., & Chan, K. (2011). Time Series Analysis. New York: Springer.
Tsay, R. S. (2002). Analysis of Financial Time Series. New York: Wiley
Ross, S. M. (1999). An Introduction to Mathematical Finance. Cambridge University Press

—- Selesai —-