

METODE PERAMALAN DERET WAKTU

Pendugaan Model ARIMA

Metode yang digunakan untuk menduga koefisien model ada 4, yaitu:

1. Metode Momen:
Menyamakan momen populasi dan momen contoh
2. Metode Kuadrat Terkecil Bersyarat:
Meminimumkan jumlah kuadrat $S_*(\phi, \mu)$
3. Metode Kuadrat Terkecil Tidak Bersyarat:
Meminimumkan jumlah kuadrat $S(\phi, \mu)$
4. Metode Kemungkinan Maksimum:
Memaksimumkan fungsi kemungkinan $L(\phi, \mu, \sigma_a^2)$

Pada RStudio, di fungsi ARIMA, set method nya menjadi ML (maximum likelihood), CSS (conditional sum of square), atau CSS - ML. Defaultnya adalah menggunakan CSS untuk menghitung nilai inisiasinya, lalu dilanjutkan dengan ML.

Diagnosa dan Peramalan Model ARIMA

Langkah - langkah dalam melakukan peramalan:

1. Membagi data menjadi 2, yaitu data training dan data testing
2. Diagnosa model dilakukan pada data training, lalu dilakukan peramalan dengan jumlahnya sebanyak data testing
3. Hitung errornya dengan cara menghitung selisih antara data testing dan hasil peramalan. Error yang paling kecil merupakan metode dan parameter yang paling baik.
4. Data training dan data testing disatukan kembali. Lalu diterapkan metode dan parameter yang didapatkan dari poin 3.
5. Lakukan peramalan.

Langkah - langkah diagnosa model ARIMA yang dilakukan pada data training:

1. Uji kestasioneran. Jika stasioner → lanjut, jika tidak stasioner → lakukan pembedaan (differencing)
2. Cek ACF dan PACFnya untuk membantu dalam menentukan model tentatif.
3. Lakukan uji signifikansi koefisien model
4. Lakukan analisis sisaan pada model tentatif agar memenuhi asumsi. Asumsi - asumsi tersebut adalah sebagai berikut:
 - a. sisaan menyebar normal (cek dengan QQPlot, uji Jarque-Bera)
 - b. ragam sisaan homogen (cek dengan plot antara fits dan residual)

- c. sisaan tidak saling berkorelasi (cek dengan ACF, PACF, dan plot antara waktu dan residual, uji Ljung-Box)
5. Lakukan overfitting
6. Jika model yang didapatkan dari overfitting lebih dari 1, lakukan perbandingan akurasi peramalan

Langkah - langkah overfitting:

1. Dari model tentatif, naikan ordenya satu per satu. Hindari menaikkan ordenya secara bersamaan. Misal model tentatifnya adalah ARMA(1,1), maka dicobakan model lain yaitu ARMA(2,1), ARMA(1,2), lalu ARMA(2,2)
2. Overfitting juga dapat dilakukan berdasarkan analisis sisaan. Perhatikan ACF dan PACF pada model tentatif.
3. Uji signifikansi koefisien model. Cek apakah koefisien tambahan berpengaruh signifikan atau tidak.

Penerapan di R

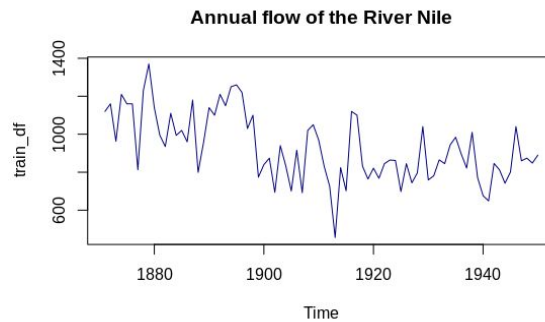
```
#Install library yg dibutuhkan
install.packages(c("lmtest", "forecast", "tseries"))

#Library
library(lmtest)
library(forecast)
library(tseries)

#Baca data
nile <- datasets::Nile

#Bagi data menjadi train sama test
train_df <- nile[1:80]
test_df <- nile[81:100]

#Eksplorasi data train
train_df <- ts(train_df, start = 1871)
plot(train_df,
     col = "navyblue",
     main = "Annual flow of the River Nile")
```

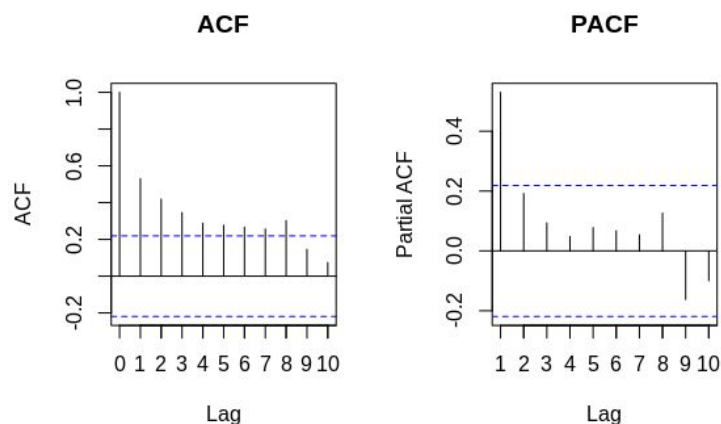


Interpretasi :

Arus air sungai nil dari tahun 1871 sampai dengan 1950 mengalami penurunan arus dari yang semula berkisar 1100 menjadi 900

```
#ACF-PACF
par(mfrow = c(1,2))
acf(train_df, lag.max = 10, main = "ACF")
axis(1, at = 1:15, labels = 1:15)

pacf(train_df, lag.max = 10, main = "PACF")
axis(1, at = 1:15, labels = 1:15)
```



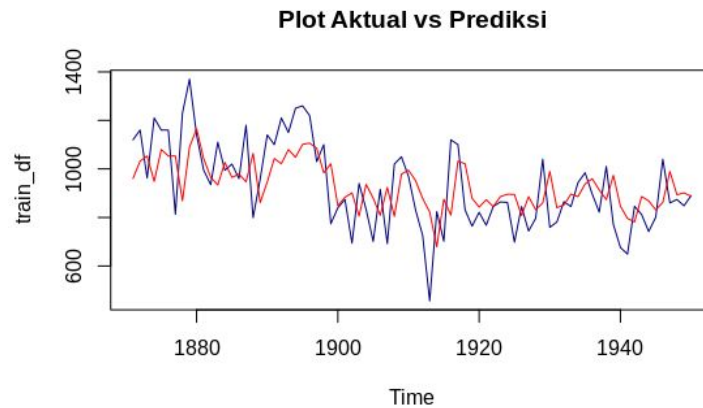
Interpretasi :

Berdasarkan hasil plot ACF dan PACF terlihat bahwa pada PACF membentuk pola terpotong (cuts-off) setelah lag ke-1. Sedangkan pada ACF membentuk pola menurun eksponensial (tails-off). Sehingga model tentatif yang didapatkan yaitu ARIMA(1,0,0)

```
#Fitting Model ARIMA
#AR(1)
arima_model <- Arima(train_df, order = c(1,0,0), method = c("ML"))
arima_pred <- arima_model$fitted

plot(train_df,
```

```
col = "navyblue",
type = "l",
main = "Plot Aktual vs Prediksi")
lines(arima_pred,
col = "red")
```



```
summary(arima_model)
> summary(arima_model)
Series: train_df
ARIMA(1,0,0) with non-zero mean
```

```
Coefficients:
      ar1      mean
    0.5317  931.9649
s.e. 0.0941  35.1241
```

```
sigma^2 estimated as 22822: log likelihood=-514.09
AIC=1034.18 AICc=1034.49 BIC=1041.32
```

```
Training set error measures:
      ME  RMSE  MAE  MPE  MAPE  MASE  ACF1
Training set -1.593991 149.1691 121.7781 -2.97998 13.8443 0.9082767
-0.1064647
```

```
coeftest(arima_model)
> coeftest(arima_model)
```

z test of coefficients:

```
      Estimate Std. Error z value Pr(>|z|)
ar1      0.531672  0.094143  5.6475 1.628e-08 ***
intercept 931.964944  35.124106 26.5335 < 2.2e-16 ***
```

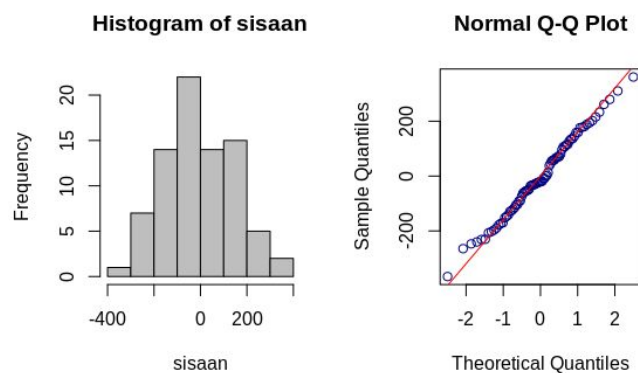
 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Interpretasi :

berdasarkan hasil uji signifikansi koefisien model ARIMA(1,0,0) didapatkan koefisien ar1 signifikan pada taraf nyata 5%

```
#Analisis Sisaan
#1. Sisaan Menyebar Normal
sisaan <- arima_model$residuals
```

```
#Secara Eksploratif
par(mfrow = c(1,2))
hist(sisaan, col = "grey")
qqnorm(sisaan, col = "navyblue")
qqline(sisaan, col = "red")
```



Interpretasi :

Berdasarkan Histogram sisaan menyebar normal. Begitupun pada QQ plot sisaan berada pada sepanjang garis lurus yang menyatakan bahwa sisaan menyebar normal

```
#Uji Formal Jarque Bera
jarque.bera.test(sisaan)
> jarque.bera.test(sisaan)
```

Jarque Bera Test

```
data: sisaan
X-squared = 0.7906, df = 2, p-value = 0.6735
```

Interpretasi :

Hasil uji jarque bera menyatakan bahwa sisaan menyebar normal

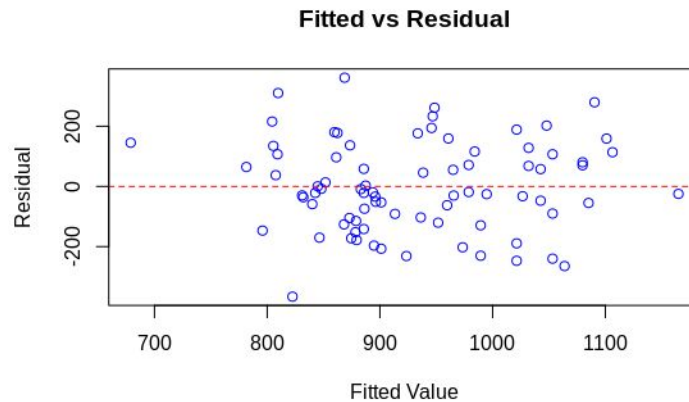
```
#2. Ragam Sisaan Homogen
plot(x = as.numeric(arima_pred), y = as.numeric(sisaan),
     col = "blue",
```

```

main = "Fitted vs Residual",
xlab = "Fitted Value",
ylab = "Residual")

abline(h = 0,
       col = "red",
       lty = 2)

```



Interpretasi :

Secara eksploratif, lebar pita terlihat sama yang menandakan ragam dari sisaan homogen

```

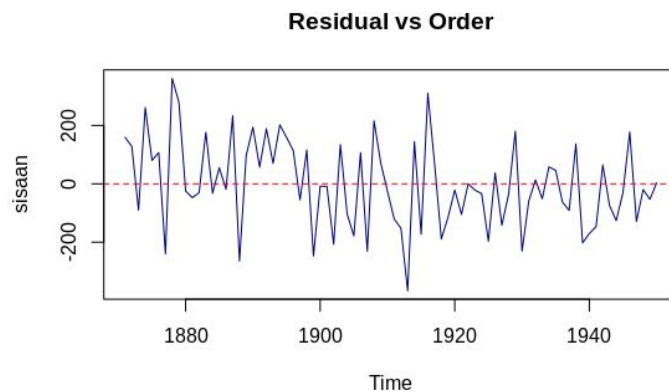
#3. Antar Sisaan Saling Bebas
plot(sisaan,
     col = "navyblue",
     main = "Residual vs Order")

```

```

abline(h = 0,
       col = "red",
       lty = 2)

```



Interpretasi :

Secara eksploratif, sisaan tidak membentuk pola yang sistematis. Hal ini menandakan bahwa antar sisaan saling bebas

#Overfitting

#AR(2)

```
arima_model_2 <- Arima(train_df, order = c(2,0,0), method = "ML")
```

```
summary(arima_model_2)
```

```
> summary(arima_model_2)
```

Series: train_df

ARIMA(2,0,0) with non-zero mean

Coefficients:

	ar1	ar2	mean
	0.4257	0.2016	933.8080
s.e.	0.1084	0.1094	42.6739

sigma² estimated as 22157: log likelihood=-512.43

AIC=1032.86 AICc=1033.4 BIC=1042.39

Training set error measures:

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	-2.688478	146.0338	119.8855	-2.985051	13.6043	0.8941608	-0.02052225

```
coeftest(arima_model_2)
```

```
> coeftest(arima_model_2)
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(> z)
ar1	0.42573	0.10841	3.9272	8.595e-05 ***
ar2	0.20155	0.10942	1.8421	0.06547 .
intercept	933.80801	42.67390	21.8824	< 2.2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Interpretasi :

Terlihat bahwa koefisien ar2 tidak signifikan pada taraf nyata 5%

#Forecasting

```
forecast_arima <- forecast(arima_model, 20)
```

```
accuracy(forecast_arima, test_df)
```

```
> accuracy(forecast_arima, test_df)
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	-1.593991	149.1691	121.7781	-2.979980	13.84430	0.9082767	-0.1064647
Test set	-52.532909	131.7822	107.0994	-7.980067	13.03465	0.7987967	NA

```
forecast_arima_2 <- forecast(arima_model_2, 20)
```

```
accuracy(forecast_arima_2, test_df)
```

```
> accuracy(forecast_arima_2, test_df)
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	-2.688478	146.0338	119.8855	-2.985051	13.60430	0.8941608	-0.02052225
Test set	-50.757720	130.8451	106.1825	-7.766733	12.90656	0.7919581	NA

Interpretasi :

Berdasarkan hasil akurasi peramalan kedua model. Performa model AR(2) memiliki RMSE sebesar 130.84 sedikit lebih baik dibandingkan model AR(1) memiliki RMSE sebesar 131.78. Kemudian, model AR(1) lebih sederhana dibandingkan AR(2) serta koefisien ar2 tidak signifikan. Sehingga model yang dipilih untuk digunakan yaitu AR(1). Diterapkan AR(1) pada keseluruhan data, yaitu gabungan antara data training dan data tes.

#Penerapan pada Semua Data ARIMA(1,0,0)

```
nile_arima <- Arima(nile, order = c(1,0,0), method = "ML")
forecast_nile <- forecast(nile_arima, 20)
plot(forecast_nile)
```

