PROJECT METODE PERAMALAN

Analisis Runtun Waktu pada Data Jumlah Penumpang Pesawat

Disusun oleh:

Kelompok B

Diki Wahyudi	**2106709131**
Medifa Puspaningrum	**2106634944**
Rachel Thyffani Margaretha	**2106726900**
Vesya Padmadewi	**2106726926**
Whitney	**2106700946**

PROGRAM STUDI STATISTIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS INDONESIA

JUNI 2023

ABSTRAK

Metode Peramalan atau *Forecasting* merupakan salah satu teknik dalam sistem perencanaan yang berfungsi untuk menentukan aktivitas produksi yang akan terjadi di masa yang akan datang berdasarkan data historis masa lalu guna memperoleh suatu sistem dan kebijakan yang lebih baik dan menguntungkan bagi perusahaan atau organisasi yang terkait. Untuk mendapatkan laba yang besar pada suatu perusahaan adalah dengan menentukan prediksi penjualan pada bulan berikutnya. Prediksi merupakan salah satu kunci dari keberhasilan penjualan karena dengan nilai prediksi penjualan yang bisa dijadikan panduan sebagai acuan untuk menentukan suatu penjualan produk. Dalam hal jumlah penumpang pesawat, maskapai penerbangan penting untuk meramalkan (*forecasting*) jumlah penumpang pesawat di masa depan agar dapat membuat keputusan yang lebih terstruktur mengenai perluasan kapasitas, perencanaan rute, dan strategi pemasaran. Metode ARIMA merupakan metode analisis data runtun waktu yang sering digunakan untuk peramalan. *Seasonal Autoregressive Integrated Moving Average* (SARIMA) merupakan pengembangan dari model *Autoregressive Integrated Moving Average* (ARIMA) pada data runtun waktu yang memiliki pola musiman. Dalam penelitian ini, data jumlah penumpang bulanan maskapai Amerika Serikat dari tahun 1949 hingga 1960 akan dimodelkan dengan analisis runtun waktu. Berdasarkan hasil analisis, model yang sesuai dengan jumlah penumpang pesawat bulanan ($\{Y_t\}$) yaitu ln(Y_t) merupakan ARIMA$(0,1,1)(0,1,1)_{12}$. Dengan model tersebut, didapatkan *forecasting* untuk data `test` menghasilkan nilai MAPE = 1.478059 < 10, yang mengindikasikan bahwa peramalan (*forecasting*) dilakukan dengan sangat akurat. Nilai *forecasting* menunjukkan tren meningkat yang mengindikasikan kemungkinan kenaikan jumlah penumpang pesawat bulanan pada maskapai penerbangan Amerika Serikat di masa depan.

Kata Kunci: Penumpang Pesawat, Peramalan, SARIMA

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan ilmu teknologi telah melakukan banyak perubahan dalam berbagai bidang kehidupan, misalnya dalam bidang transportasi. Perjalanan yang awalnya dilakukan dengan berjalan kaki, secara perlahan mulai berubah dengan kemunculan transportasi darat, laut, hingga udara. Kemudahan yang diberikan oleh transportasi ini dapat dilihat dalam kehidupan sehari hari, seperti membantu dalam perpindahan barang maupun manusia ke berbagai wilayah sehingga berperan pula dalam menunjang perkembangan pembangunan. Transportasi udara, pada khususnya, merupakan transportasi yang memiliki keunggulan lebih, yakni dapat menjangkau berbagai wilayah dalam waktu yang relatif lebih cepat dan bebas hambatan dibanding dua jenis transportasi lain. Hal ini dapat dibuktikan melalui perjalanan yang biasanya harus ditempuh dalam waktu berhari-hari dapat dilakukan hanya dalam hitungan jam saja. Adanya faktor ini merupakan salah satu alasan mengapa banyak dari masyarakat yang lebih memilih untuk bepergian menggunakan transportasi udara, seperti pesawat.

Amerika Serikat, sebagai salah satu negara maju dan merupakan negara tersibuk di dunia, merupakan negara di urutan pertama dengan jumlah penumpang pesawat terbanyak per harinya di dunia. Dilansir dari FAA Air Traffic Organization, terdapat sebanyak 917 juta penumpang maskapai Amerika Serikat pada tahun 2022 atau naik sebesar 55% dari tahun sebelumnya. Hal ini menunjukkan bahwa penggunaan transportasi udara cukup menarik perhatian banyak orang.

Pada penelitian ini, akan dilakukan analisis *forecast* untuk meramalkan jumlah penumpang pesawat Amerika Serikat selama 4 tahun ke depan (dimulai dari akhir waktu data `train`). Data yang digunakan merupakan data jumlah penumpang bulanan maskapai Amerika Serikat dari tahun 1949 hingga 1960 yang dapat diakses melalui *platform* Kaggle.

1.2 Rumusan Masalah

Berdasarkan latar belakang dari penelitian yang dilakukan, maka dapat dibuat beberapa rumusan masalah sebagai berikut.

1. Bagaimana model yang cocok untuk memodelkan data jumlah penumpang pesawat di Amerika Serikat?
2. Bagaimana hasil *forecast* (ramalan) jumlah penumpang pesawat Amerika Serikat selama 4 tahun ke depan menggunakan metode analisis runtun waktu?

3. Bagaimana keakuratan hasil *forecast* menggunakan analisis runtun waktu dalam *forecasting* (peramalan) jumlah penumpang pesawat bulanan di Amerika Serikat?

1.3 Tujuan

Berdasarkan rumusan masalah di atas, maka tujuan yang ingin dicapai dalam penelitian ini adalah

1. Mendapatkan model runtun waktu yang sesuai untuk memodelkan jumlah penumpang pesawat pada maskapai penerbangan di Amerika Serikat.
2. Mengetahui hasil *forecast* (ramalan) jumlah penumpang pesawat pada maskapai penerbangan Amerika Serikat selama 4 tahun kedepan menggunakan analisis runtun waktu.
3. Mengetahui keakuratan hasil *forecast* dalam meramalkan (*forecasting*) jumlah penumpang pesawat bulanan di Amerika Serikat.

1.4 Manfaat

Beberapa manfaat yang dapat diharapkan akan diperoleh dari hasil analisis runtun waktu pada data jumlah penumpang pesawat maskapai penerbangan di Amerika Serikat adalah sebagai berikut.

1. Kita memperoleh model yang dapat meramalkan (*forecasting*) jumlah penumpang di masa depan sehingga dapat mempermudah maskapai penerbangan untuk merencanakan sumber daya, kapasitas, dan operasional penerbangan.
2. Kita dapat mendeteksi anomali yang mengindikasikan kejadian signifikan, seperti cuaca ekstrem, pandemi, atau resesi ekonomi.
3. Kita dapat memahami dan memprediksi nilai-nilai masa depan agar dapat membuat keputusan yang lebih terstruktur mengenai perluasan kapasitas, perencanaan rute, dan strategi pemasaran.

BAB II

METODE PENELITIAN

2.1 Variabel Penelitian

Kami menggunakan data yang diambil dari Kaggle melalui link berikut ini. https://www.kaggle.com/datasets/chirag19/air-passengers. Dataset tersebut berisi jumlah penumpang bulanan maskapai Amerika Serikat dari tahun 1949 hingga 1960. Dataset ini juga tersedia dalam dataset bawaan R dengan nama `AirPassengers`.

Keterangan variabel:

- `month`: bulan dari tahun.
- `passenger`: jumlah penumpang yang melakukan perjalanan pada bulan tersebut (dalam satuan ribuan).

2.2 Data

Import Library

```
library(ggplot2)
library(ggfortify)
library(tseries)
library(forecast)
library(TSA)
library(TSstudio)
library(zoo)
library(lmtest)
library(fpp2)
library(knitr)
```

Import Data

```
data("AirPassengers")
AP <- AirPassengers
passanger <- as.vector(AP)
```

```r
month <- as.yearmon(time(AP))
df <- data.frame(passanger, month)
```

Catatan: `df` merupakan representasi dataframe dari objek time series `AP`. Akan ditampilkan data yang telah di-*import* tersebut.

```r
AP
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1949 112 118 132 129 121 135 148 148 136 119 104 118
## 1950 115 126 141 135 125 149 170 170 158 133 114 140
## 1951 145 150 178 163 172 178 199 199 184 162 146 166
## 1952 171 180 193 181 183 218 230 242 209 191 172 194
## 1953 196 196 236 235 229 243 264 272 237 211 180 201
## 1954 204 188 235 227 234 264 302 293 259 229 203 229
## 1955 242 233 267 269 270 315 364 347 312 274 237 278
## 1956 284 277 317 313 318 374 413 405 355 306 271 306
## 1957 315 301 356 348 355 422 465 467 404 347 305 336
## 1958 340 318 362 348 363 435 491 505 404 359 310 337
## 1959 360 342 406 396 420 472 548 559 463 407 362 405
## 1960 417 391 419 461 472 535 622 606 508 461 390 432
```

Ukuran Data

```r
start(AP)
```

```
## [1] 1949    1
```

```r
end(AP)
```

```
## [1] 1960   12
```

```r
frequency(AP)
```

```
## [1] 12
```

Terlihat bahwa data tercatat dari bulan Januari tahun 1949 sampai bulan Desember 1960 dengan frekuensi per bulan sehingga banyaknya observasi yang tercatat ada $12 \times 12 = 144$ observasi.

2.3 Jenis Data

- `passanger`: numerik, rasio
- `month`: kategorik, nominal

```r
knitr::kable(head(df), align = "cc", caption = "Head dari df", booktabs = TRUE)
```

Tabel 1: Head dari df

passanger	month
112	Jan 1949
118	Feb 1949
132	Mar 1949
129	Apr 1949
121	May 1949
135	Jun 1949

2.4 Metode Analisis Data

Runtun Waktu

Dalam model runtun waktu yang akan kami buat, $Y_t :=$ `passenger` yaitu jumlah penumpang pesawat pada periode t. Runtun waktu $\{Y_t | t = 0, 1, 2, ...\}$ merupakan proses stokastik dengan t dalam satuan bulan.

Dekomposisi Runtun Waktu

Dekomposisi mengasumsikan bahwa data terdiri atas:

$$\begin{aligned}\text{data} &= \text{pola} + \text{eror}\\ &= f(\text{tren-siklus, musiman, eror}).\end{aligned}$$

Jadi, selain komponen pola, terdapat juga komponen dari eror atau keacakan. Eror tersebut dianggap sebagai perbedaan antara efek gabungan dari kedua subpola dari runtun waktu dan data asli.

Representasi matematika dari pendekatan dekomposisi tersebut yaitu

$$Y_t = f(S_t, T_t, E_t)$$

di mana Y_t merupakan nilai runtun waktu (data asli) pada periode t, S_t merupakan komponen musiman (*seasonal*) pada periode t, T_t merupakan komponen tren-siklus (*trend-cycle*) pada periode t, dan E_t merupakan komponen iregular (sisa) pada periode t.

Dekomposisi Aditif: $Y_t = S_t + T_t + E_t$.

Dekomposisi Multiplikatif: $Y_t = S_t \times T_t \times E_t$.

ARMA (*Autoregressive Moving Average*)

Jika Y_t merupakan proses "*mixed autoregressive moving average*" order p dan q, dinotasikan ARMA(p, q), maka bentuk persamaannya adalah

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + ... + \phi_p Y_{t-p} + e_t - \theta_1 e_{t-1} - \theta_2 e_{t-2} - ... - \theta_q e_{t-q}. \tag{1}$$

ARIMA (*Integrated Autoregressive Moving Average*)

Jika $W_t, W_t = \nabla^d Y_t$, merupakan proses ARMA(p, q) yang stasioner, maka Y_t merupakan proses ARIMA(p, d, q).

Model Musiman Multiplikatif

Model musiman multiplikatif ARMA$(p, d, q)(P, Q)_s$ dengan periode musiman s didefinisikan sebagai model dengan polinomial karakteristik AR $\phi(x)\Phi(x)$ dan polinomial karakteristik MA $\theta(x)\Theta(x)$ dengan

$$\phi(x) = 1 - \phi_1 x - \phi_2 x^2 - ... - \phi_p x^p$$

$$\Phi(x) = 1 - \Phi_1 x^s - \Phi_2 x^{2s} - ... - \Phi_P x^{Ps}$$

dan

$$\theta(x) = 1 - \theta_1 x - \theta_2 x^2 - ... - \theta_q x^q$$

$$\Theta(x) = 1 - \Theta_1 x^s - \Theta_2 x^{2s} - ... - \Theta_Q x^{Qs}.$$

Model ARIMA Musiman Nonstasioner

Difference musiman dengan periode s untuk runtun waktu Y_t dinotasikan dengan $\nabla_s Y_t$ dan didefinisikan sebagai $\nabla_s Y_t = Y_t - Y_{t-s}$. Untuk model musiman multiplikatif ARIMA dengan order *nonseasonal* (regular) p, d, dan q, order *seasonal* P, D, dan Q, serta periode *seasonal* s, jika *difference* dari runtun $W_t = \nabla^d \nabla_s^D Y_t$ memenuhi ARMA$(p, q)(P, Q)_s$, maka runtun Y_t disebut model ARIMA$(p, d, q)(P, D, Q)_s$ dengan periode musiman (*seasonal*) s.

Sampel *Autocorrelation Function* (ACF)

Taksiran untuk ρ_k, autokorelasi pada lag ke-k, yaitu

$$r_k = \frac{\sum_{t=k+1}^{n}(Y_t - \bar{Y})(Y_{t-k} - \bar{Y})}{\sum_{t=1}^{n}(Y_t - \bar{Y})^2}$$

di mana $k = 1, 2, ...$

Ukuran Akurasi *Forecasting*

Misalkan Y_t merupakan observasi sebenarnya pada waktu periode ke-t dan F_t merupakan *forecast* (ramalan) untuk periode yang sama, maka eror didefinsikan sebagai

$$e_t = Y_t - F_t.$$

Jika terdapat n observasi dam *forecast*, maka ukuran akurasi *forecasting* dari suatu model yaitu

$$\text{ME} = \frac{1}{n}\sum_{t=1}^{n} e_t \text{ (Mean Error)}$$

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n} |e_t| \text{ (Mean Absolute Error)}$$

$$\text{MSE} = \frac{1}{n}\sum_{t=1}^{n} e_t^2 \text{ (Mean Squared Error)}$$

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n} \left| \left(\frac{Y_t - F_t}{Y_t}\right) \times 100 \right| \text{ (Mean Absolute Percentage Error).}$$

BAB III

HASIL DAN PEMBAHASAN

3.1 *Exploratory Data Analysis* (EDA)

Preprocessing

Pada bagian ini, akan dicek apakah terdapat *missing value* atau tidak.

```
sapply(df, function(x) sum(is.na(x)))
```

```
## passanger     month
##         0         0
```

Terlihat bahwa data `df` (representasi dataframe dari objek time series `AP`) tidak memiliki *missing value*.

Statistika Deskriptif

```
summary(AP)
```

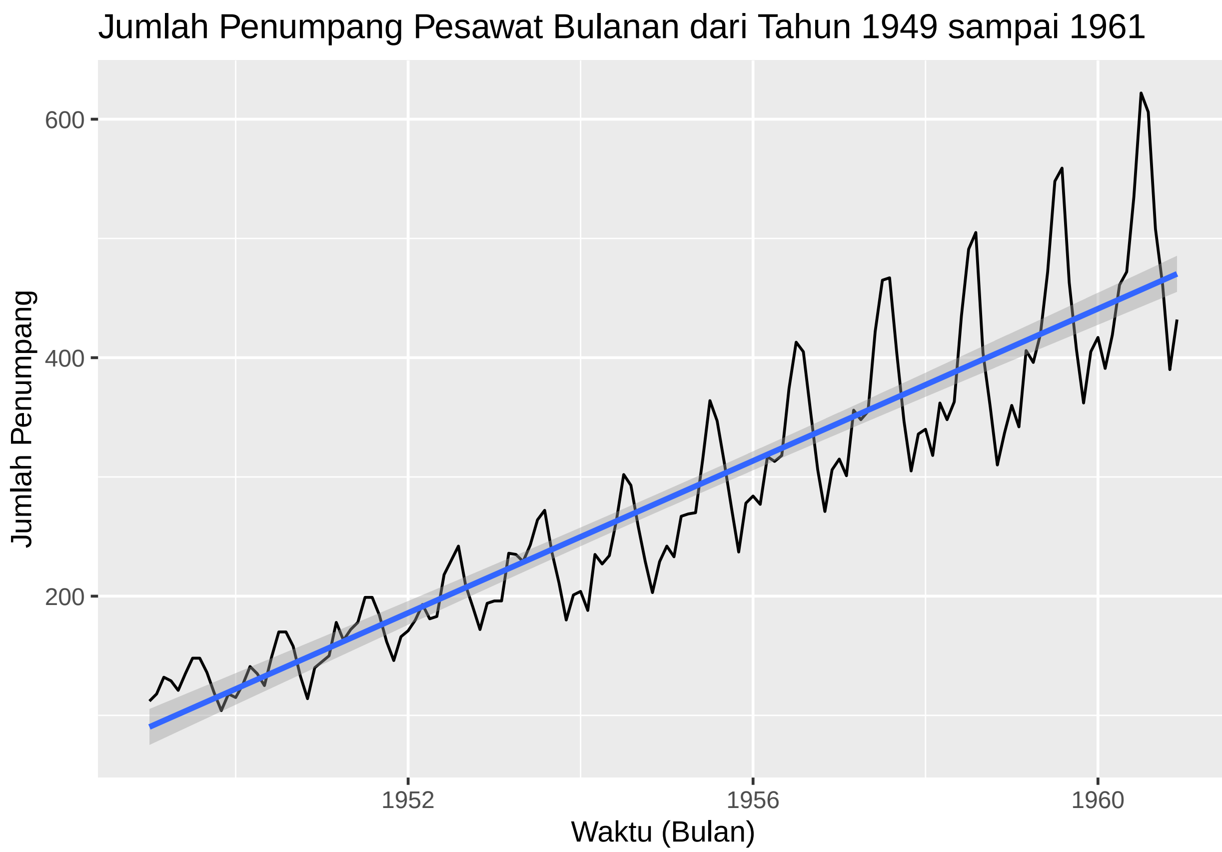
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   104.0   180.0   265.5   280.3   360.5   622.0
```

Terlihat bahwa rata-rata jumlah penumpang pesawat per bulannya yaitu sekitar 265.5 ribu penumpang, di mana jumlah penumpang pesawat per bulan terbanyak yaitu 622 ribu dan jumlah penumpang pesawat per bulan tersedikit yaitu 104 ribu penumpang.

Visualisasi

Berikut ini merupakan *time* plot dari `AP`.

```
ggplot(AP, aes(x = time(AP), y = AP)) + geom_line() +
  geom_smooth(method = "lm") +
  labs(x = "Waktu (Bulan)", y = "Jumlah Penumpang",
       title = "Jumlah Penumpang Pesawat Bulanan dari Tahun 1949 sampai 1961")
```

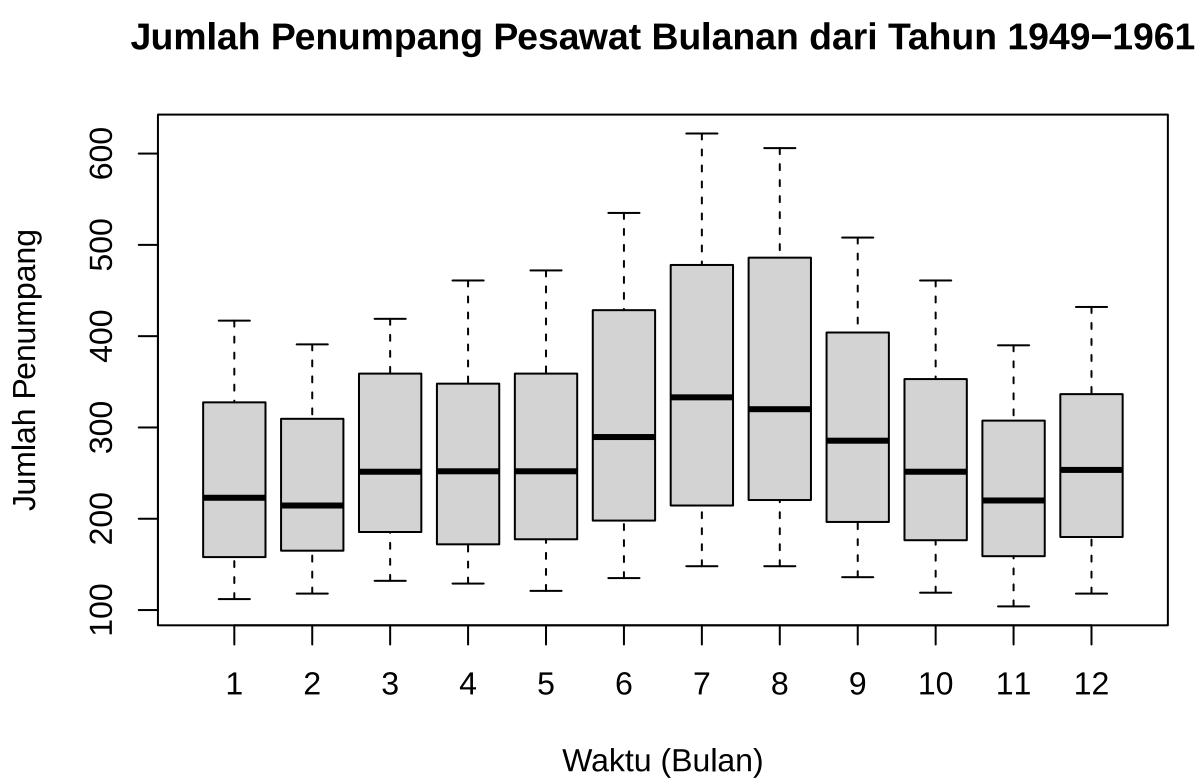


Gambar 1: Line Plot Jumlah Penumpang Pesawat

Jumlah penumpang meningkat dari waktu ke waktu setiap tahun yang menunjukkan tren linier meningkat. Hal tersebut mungkin disebabkan oleh meningkatnya permintaan untuk perjalanan penerbangan dan komersialisasi maskapai penerbangan dalam periode waktu tersebut. Jika data tersebut dimodelkan dengan model linier (garis biru), model tersebut hanya mampu menangkap tren linier meningkatnya saja, tidak menangkap efek musiman dan efek perkalian (variansi yang semakin membesar) dari waktu ke waktu. Oleh karena itu, model linier tidak cocok untuk data `AP`. Dari *time* plot tersebut, terlihat bahwa mean dan variansi dari runtun waktu `AP` berubah seiring waktu yang mengindikasikan ketidakstasioneran.

Selanjutnya, akan dibuat boxplot dari `AP`.

```
boxplot(AP ~ cycle(AP), xlab = "Waktu (Bulan)", ylab = "Jumlah Penumpang",
        main = "Jumlah Penumpang Pesawat Bulanan dari Tahun 1949-1961")
```

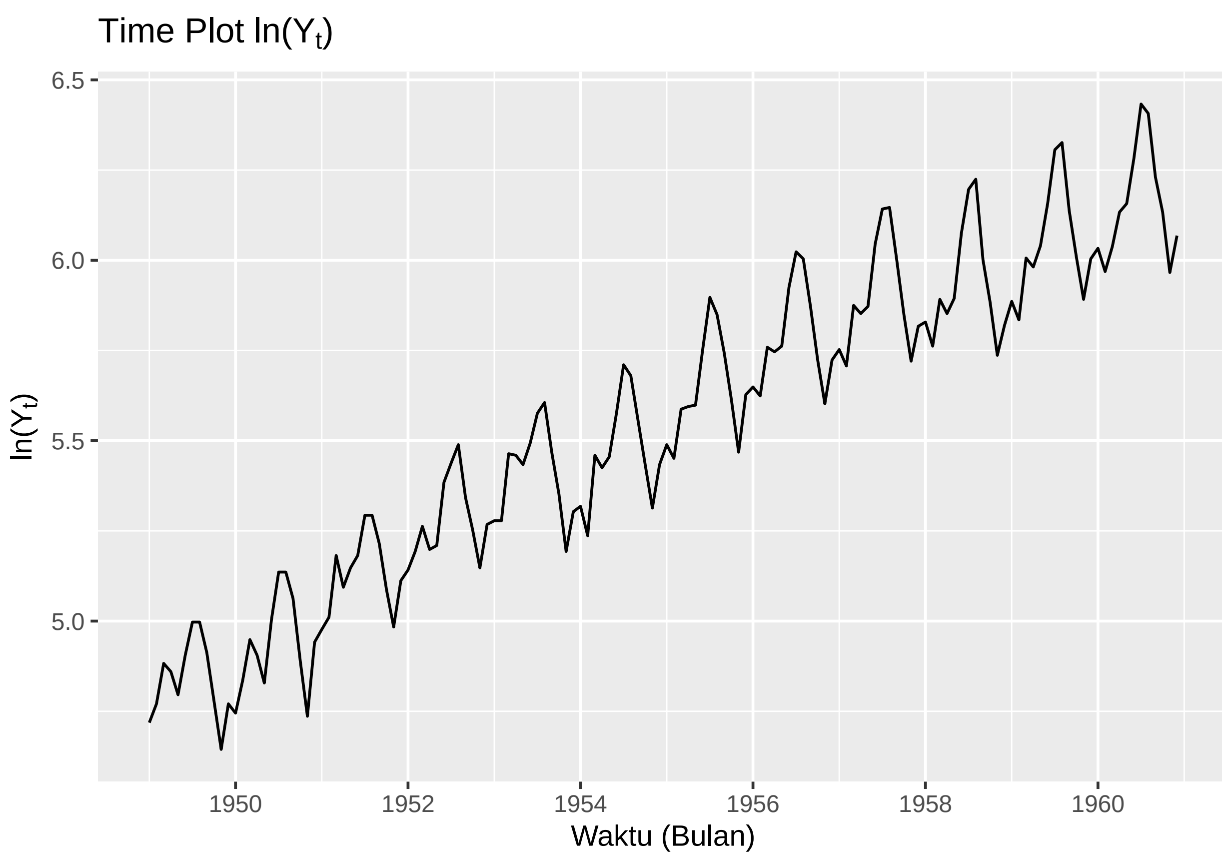


Gambar 2: Boxplot Jumlah Penumpang Pesawat

Dalam boxplot tersebut, terdapat lebih banyak penumpang yang bepergian di bulan 6 sampai 9 dengan rata-rata yang lebih tinggi dan variansi yang lebih tinggi daripada bulan lainnya. Informasi tersebut mengindikasikan terdapatnya pola musiman dengan siklus 12 bulan. Hal tersebut mungkin disebabkan oleh lebih banyak orang yang berlibur dan melakukan penerbangan selama periode musim panas di Amerika Serikat.

Karena runtun waktu Y_t variansinya meningkat seiring bertambahnya waktu, akan dibuat *time* plot dari transformasi Y_t yaitu $Z_t = \ln(Y_t)$ untuk mengecek variansi menjadi lebih konstan atau tidak.

```
autoplot(log(AP)) + labs(x = "Waktu (Bulan)", y = expression("ln(Y"[t]*")"),
       title = expression("Time Plot ln(Y"[t]*")"))
```



Gambar 3: Line Plot $\ln(Y_t)$

Terlihat bahwa variansi dari runtun waktu $\ln(Y_t)$ lebih konstan dibandingkan dengan runtun waktu asli Y_t.

Berdasarkan tiga plot tersebut didapatkan informasi bahwa

1. Model linier tidak cocok untuk data ini karena tidak dapat menangkap efek musiman pada data.
2. `AP` tampaknya merupakan deret waktu multiplikatif karena variasi musimannya (*seasonal*) meningkat seiring bertambahnya waktu. Terdapat pola musiman dengan siklus 12 bulan.
3. Tidak ada outlier.
4. Terdapat indikasi ketidakstasioneran pada runtun waktu Y_t.
5. Runtun waktu $\ln(Y_t)$ terlihat lebih konstan dalam variansi daripada runtun waktu Y_t.

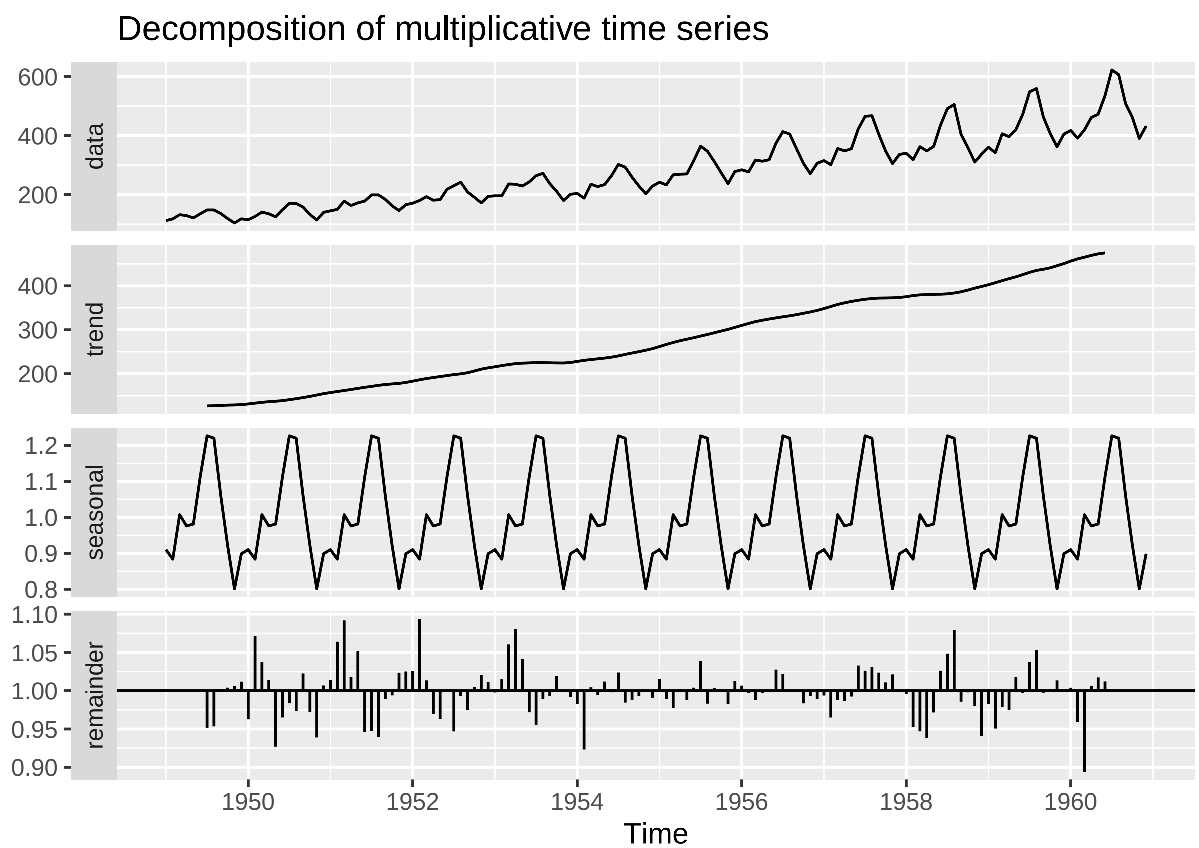
Dekomposisi Runtun Waktu

Berdasarkan bagian visualisasi, didapatkan informasi bahwa data `AP` cenderung merupakan runtun waaktu multiplikatif. Pada bagian ini, runtun waktu tersebut akan diuraikan untuk memperkirakan tren, musiman, dan komponen acak (*eror*) menggunakan *moving average*.

Model multiplikatif dari runtun waktu tersebut yaitu

$$Y_t = S_t \times T_t \times E_t.$$

```
decomposeAP <- decompose(AP, "multiplicative")
autoplot(decomposeAP)
```



Gambar 4: Plot Dekomposisi Runtun Waktu

Dalam plot runtun waktu yang telah terdekomposisi tersebut, terlihat bahwa terdapat pola tren dan musiman yang kuat seperti pada bagian visualisasi sebelumnya.

Data *Splitting*

Data akan dibagi menjadi data *train* dan *test*. Akan diambil data dari tahun 1949 sampai 1958 untuk data `train` dan data sisanya yaitu dari tahun 1959 sampai 1960 untuk data `test`. *Cross validation* dilakukan pada data, dengan data *train* digunakan untuk pemodelan, kemudian data *test* akan digunakan untuk menilai kinerja (*performance*) dari model.

```
train <- window(AP, start = 1949, end = c(1958, 12))
test <- window(AP, start = 1959, end = c(1960, 12))
```

3.2 Uji Stasioneritas Runtun Waktu

Runtun waktu stasioner memiliki syarat bahwa rata-rata, variansi, dan kovariansi bukan merupakan fungsi dari waktu. Agar sesuai dengan model ARIMA, runtun waktu harus stasioner.

Akan digunakan Uji Kwiatkowski-Phillips-Schimdt-Shin (KPSS).

H_0: Tidak ada *unit root* (runtun waktu stasioner)

H_1: Ada *unit root* (runtun waktu tidak stasioner)

```
kpss.test(train)
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  train
## KPSS Level = 2.2762, Truncation lag parameter = 4, p-value = 0.01
```

Tolak H_0 jika $p - value < \alpha$. Dari hasil tersebut, didapatkan bahwa $p - value = 0.01 < \alpha = 0.05$ yang berarti H_0 ditolak. Berarti, runtun waktu Y_t tidak stasioner. Berdasarkan hasil visualisasi pada bagian EDA, didapatkan bahwa ln(Y_t) variansinya cenderung konstan, namun memiliki tren meningkat. Akan dilakukan uji stasioneritas pada runtun waktu dari *differencing* pertama pada ln(Y_t).

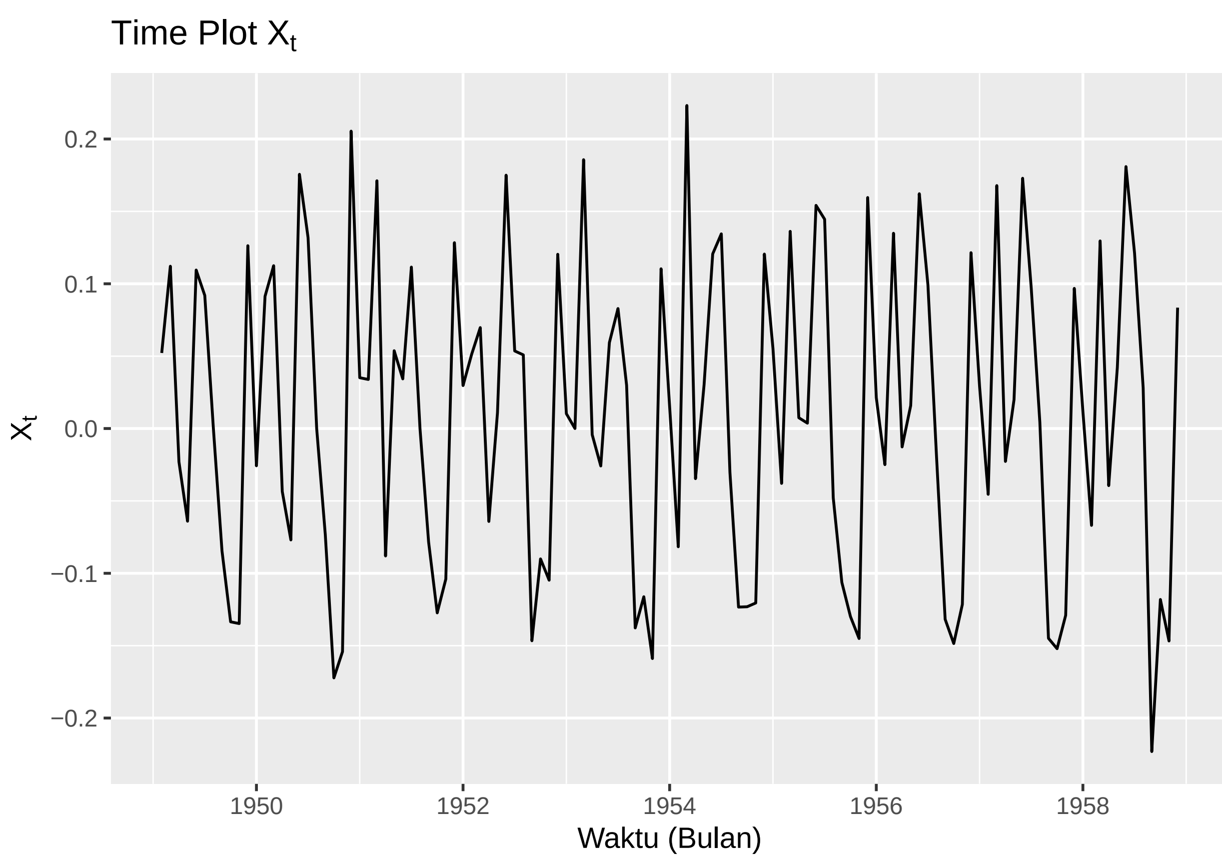
```
kpss.test(diff(log(train), lag = 1))
```

```
##
```

```
##  KPSS Test for Level Stationarity
##
## data:  diff(log(train), lag = 1)
## KPSS Level = 0.039385, Truncation lag parameter = 4, p-value = 0.1
```

Terlihat bahwa $p - value = 0.1 > \alpha = 0.05$ yang berarti H_0 tidak ditolak. Misalkan $Z_t = \ln(Y_t)$, maka runtun waktu $X_t = \nabla Z_t = Z_t - Z_{t-1}$ merupakan runtun waktu yang stasioner.

```
autoplot(diff(log(train), lag = 1)) + labs(x = "Waktu (Bulan)", y = expression("X"[t]),
        title = expression("Time Plot X"[t]))
```



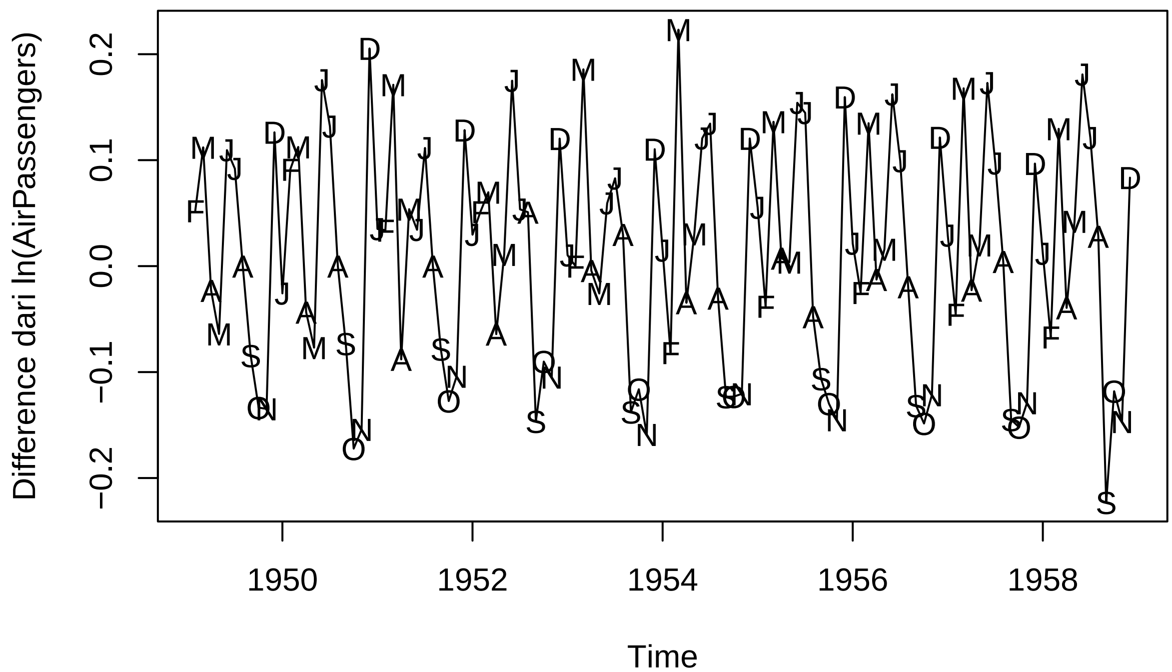
Gambar 5: Line Plot X_t

Dari *time* plot tersebut, terlihat bahwa mean dan variansi X_t relatif stabil, tidak bergantung pada waktu.

3.3 Spesifikasi Model

Akan dibuat *time* plot X_t dengan menambahkan label bulan di plotnya.

```r
plot(diff(log(train), lag = 1), type = 'l', ylab = 'Difference dari ln(AirPassengers)')
points(diff(log(train), lag = 1), x = time(diff(log(train), lag = 1)),
       pch = as.vector(season(diff(log(train), lag = 1))))
```

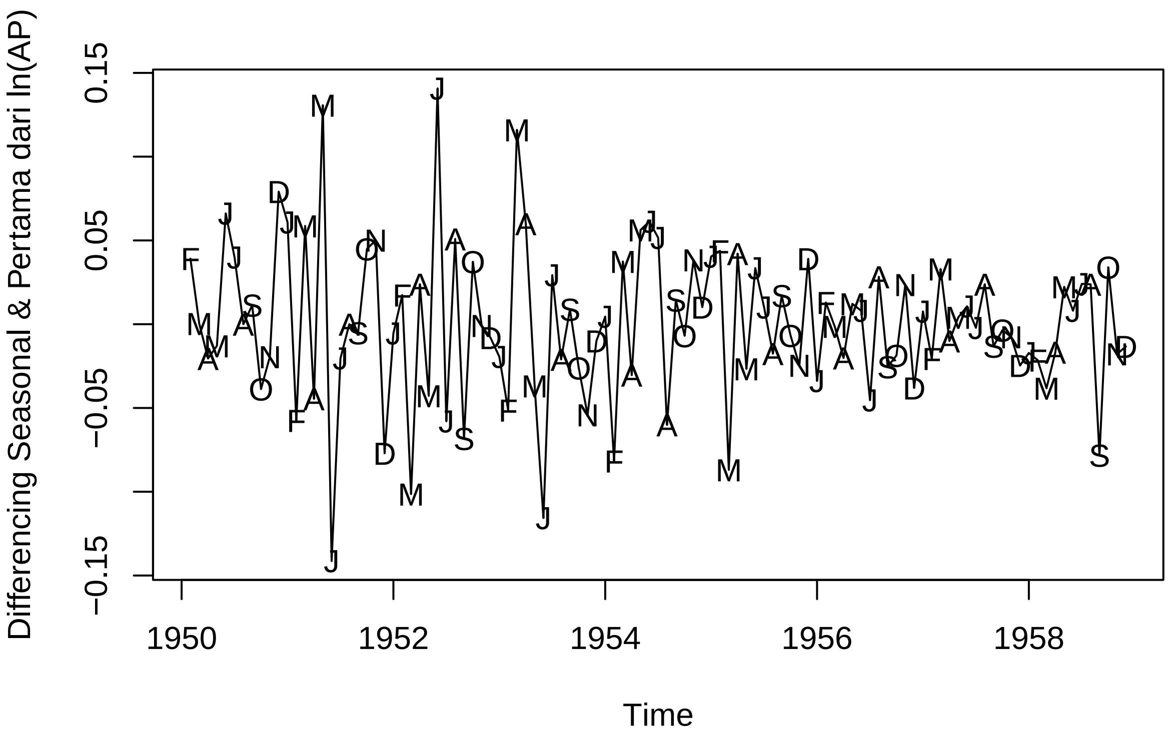


Gambar 6: Deteksi Seasonal dari Differencing

Pola musiman dapat diamati dengan melihat label bulan dari plot tersebut. Bulan September, Oktober, dan November sebagian besar berada di titik terendah dan Desember sebagian besar berada di titik tertinggi. Hal ini sejalan dengan visualisasi pada bagian EDA, terdapatnya pola musiman dengan periode musimannya 12. Oleh karena itu, akan dibuat *time* plot runtun waktu *differencing* musiman dengan periode musiman $s = 12$ dari *differencing* pertama $\ln Y_t$ untuk mengetahui apakah pola musiman masih terdeteksi atau tidak.

```r
plot(diff(diff(log(train)), lag = 12), type = 'l',
     ylab = 'Differencing Seasonal & Pertama dari ln(AP)')
points(diff(diff(log(train)), lag = 12), x = time(diff(diff(log(train)), lag = 12)),
       pch = as.vector(season(diff(diff(log(train)), lag = 12))))
```

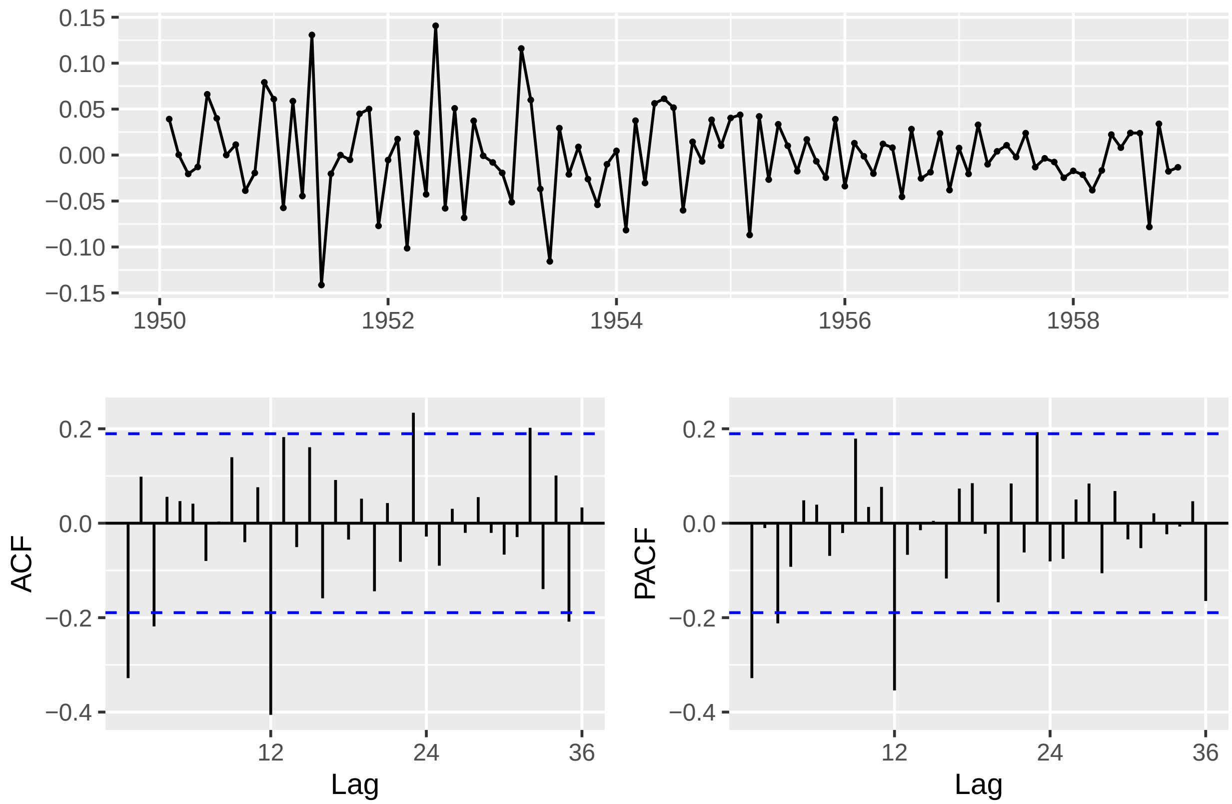
Sekarang, pola musiman jauh lebih tidak jelas. Beberapa titik di bulan Desember tinggi dan beberapa titik yang



Gambar 7: Deteksi Seasonal dari Differencing

lainnya rendah. Demikian juga beberapa titik di bulan Oktober tinggi dan beberapa titik yang lainnya rendah. Oleh karena itu, akan digabungkan *differencing* regular (*nonseasonal*) dan *differencing seasonal* untuk spesifikasi model selanjutnya. Akan dibuat plot ACF dan PACF dari $W_t = \nabla^1 \nabla^1_{12} \ln(Y_t)$ di mana $d = 1, D = 1$, dan $s = 12$.

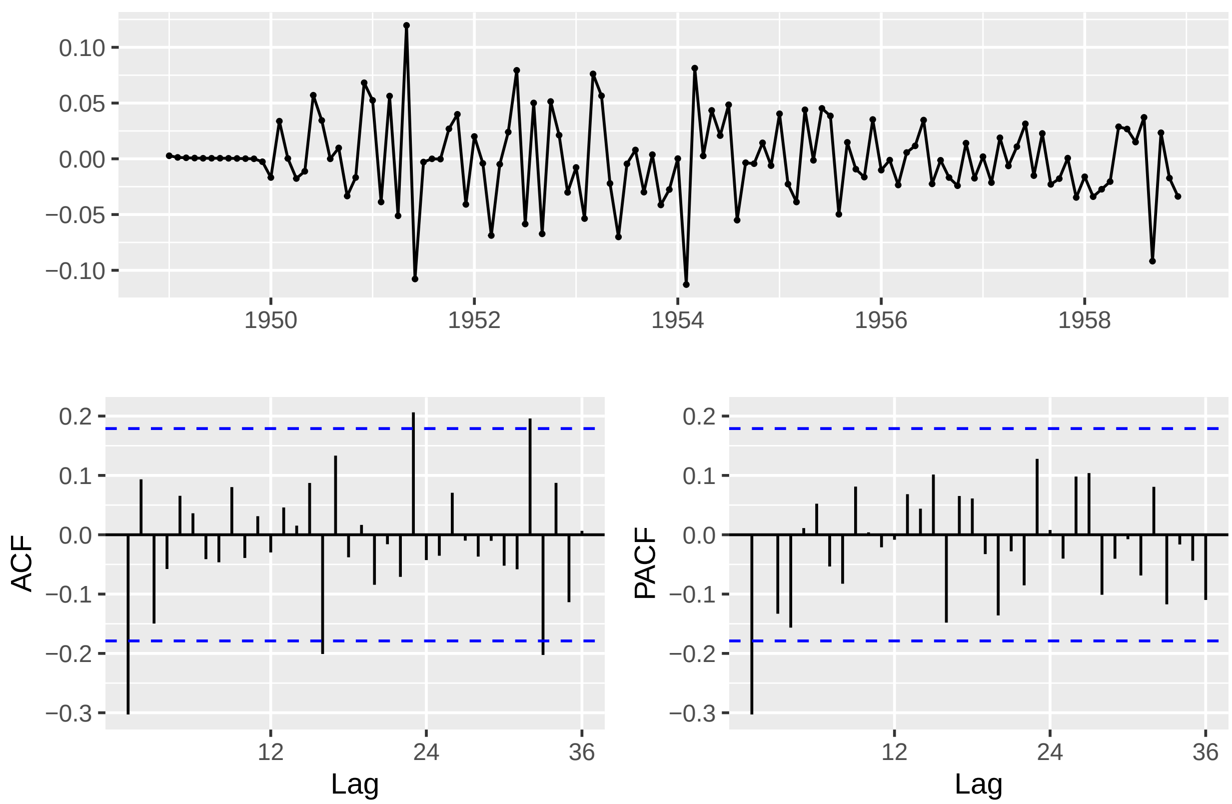
```
ggtsdisplay(diff(diff(log(train), lag = 12)))
```



Gambar 8: Summary Plot

Terlihat bahwa terdapat garis yang signifikan di Lag = 12 pada plot ACF (berhubungan dengan order *seasonal* MA) maupun PACF (berhubungan dengan order *seasonal* AR). Oleh karena itu, akan dilakukan *fitting* model pertama yaitu ARIMA$(0, 1, 0)(1, 1, 1)_{12}$ = `fit1` pada $\ln(Y_t)$.

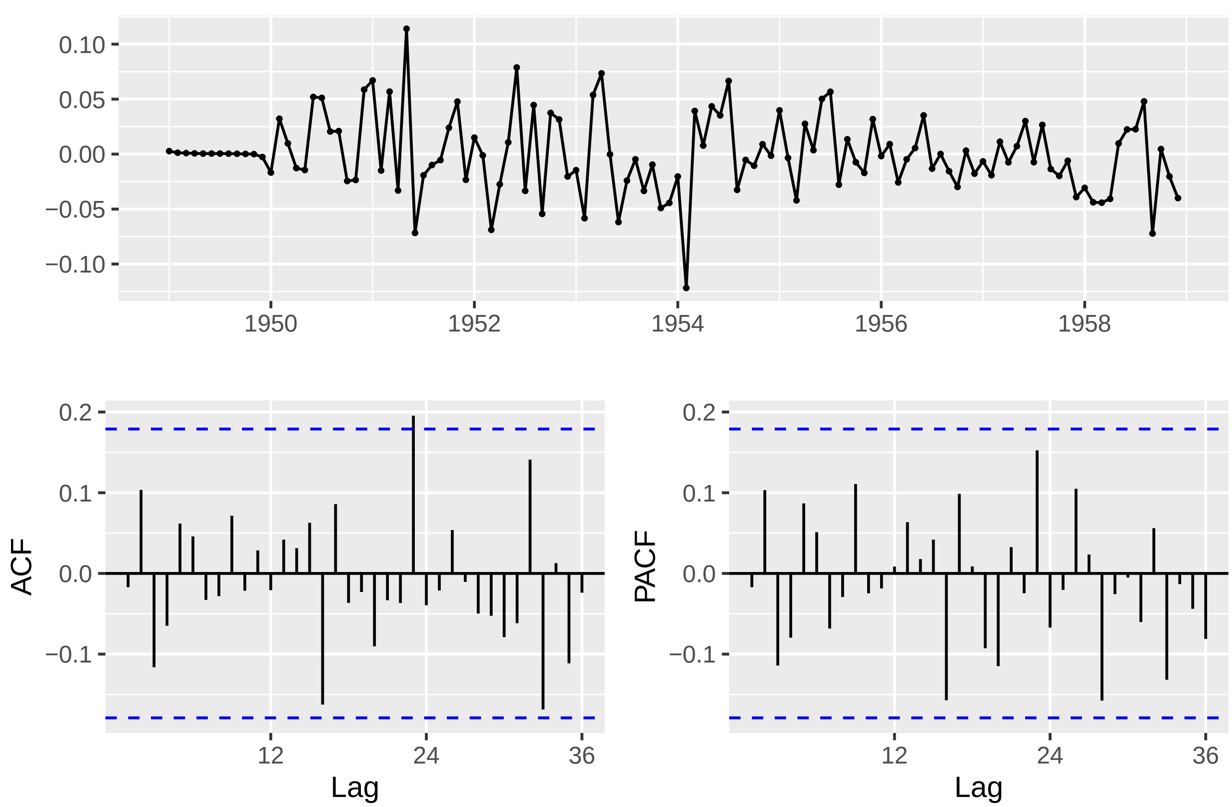
```
fit1 <- Arima(train, order = c(0, 1, 0),
              seasonal = list(order = c(1, 1, 1), period = 12),
              lambda = 0, include.constant = TRUE)
# lambda = 0 -> transformasi ln
# include.constant = TRUE -> karena data tidak mulai dari 0
ggtsdisplay(fit1$residuals)
```



Gambar 9: Summary Plot

Dari plot residual `fit1`, terlihat bahwa terdapat garis yang signifikan di Lag = 1 pada plot ACF maupun PACF. Berdasarkan hal tersebut, akan dilakukan *fitting* model kedua yaitu ARIMA$(1, 1, 1)(1, 1, 1)_{12}$ = `fit2` pada ln(Y_t), di mana melibatkan order reguler dari AR dan MA.

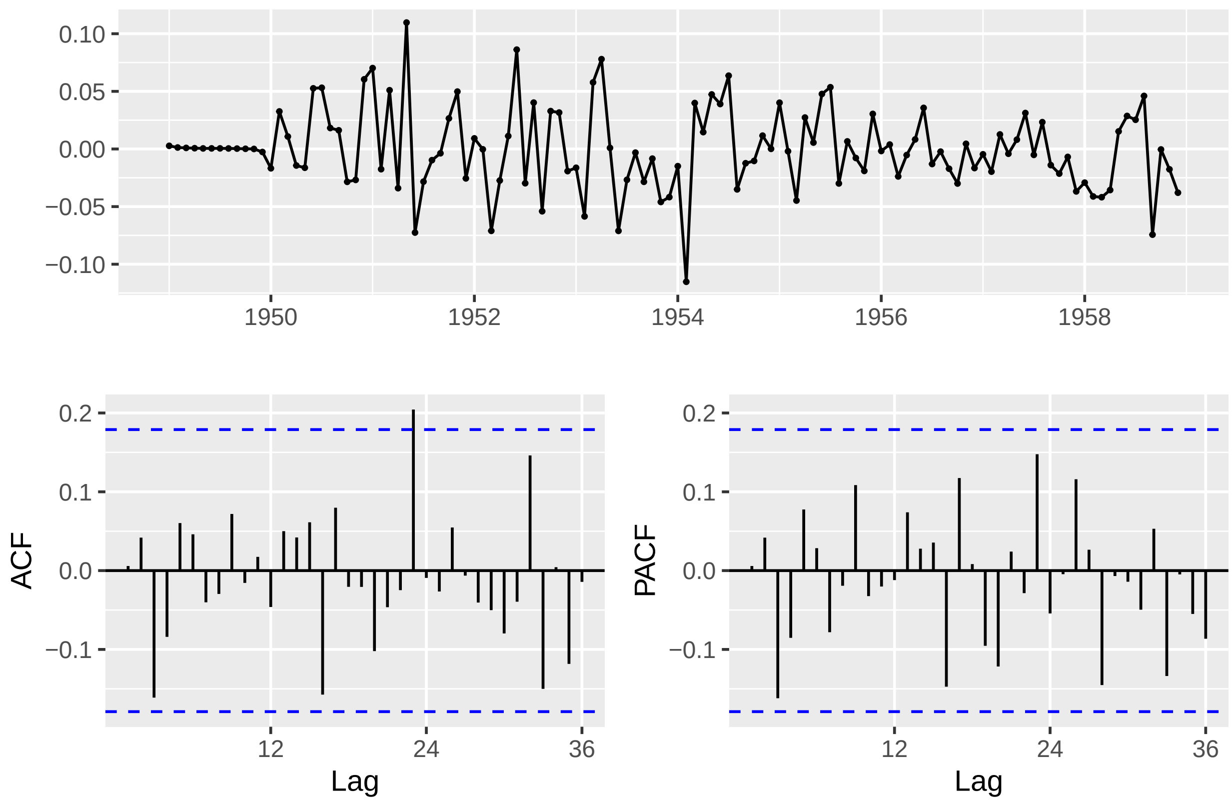
```
fit2 <- Arima(train, order = c(1, 1, 1), seasonal = c(1, 1, 1),
              lambda = 0, include.constant = TRUE)
# lambda = 0 -> transformasi ln
# include.constant = TRUE -> karena data tidak mulai dari 0
# Note: default dari periode seasonal = 12
ggtsdisplay(fit2$residuals)
```



Gambar 10: Summary Plot

Terlihat bahwa untuk PACF sudah tidak signifikan berbeda dengan 0, namun terdapat garis yang signifikan pada Lag = 23 pada ACF. Selanjutnya, akan dilakukan *fitting* model ketiga yaitu ARIMA$(0, 1, 1)(0, 1, 1)_{12}$ = `fit3` pada ln(Y_t) untuk mengecek apakah pengurangan parameter memberikan hasil yang mirip atau tidak dengan model sebelumnya.

```
fit3 <- Arima(train, order = c(0, 1, 1), seasonal = c(0, 1, 1),
              lambda = 0, include.constant = TRUE)
# lambda = 0 -> transformasi ln
# include.constant = TRUE -> karena data tidak mulai dari 0
ggtsdisplay(fit3$residuals)
```



Gambar 11: Summary Plot

Terlihat bahwa pada model ketiga garisnya sudah tidak signifikan lagi untuk PACF, namun terdapat garis yang signifikan pada plot ACF di Lag = 23. Hal tersebut sama dengan model kedua.

Perbandingan Model

Pada bagian ini, akan dilakukan pemilihan model berdasarkan nilai AIC dan BIC-nya.

```
summary(fit1)
```

```
## Series: train
```

```
## ARIMA(0,1,0)(1,1,1)[12]
## Box Cox transformation: lambda= 0
##
## Coefficients:
##         sar1     sma1
##       0.0113  -0.5988
## s.e.  0.1625   0.1452
##
## sigma^2 = 0.001574:  log likelihood = 192
## AIC=-378   AICc=-377.77   BIC=-369.98
##
## Training set error measures:
##                     ME     RMSE      MAE         MPE     MAPE      MASE
## Training set -0.25062 9.193938 6.688437 -0.08301879 2.699198 0.2340736
##                     ACF1
## Training set -0.2725758
```

```
summary(fit2)
```

```
## Series: train
## ARIMA(1,1,1)(1,1,1)[12]
## Box Cox transformation: lambda= 0
##
## Coefficients:
##          ar1      ma1     sar1     sma1
##       0.2001  -0.5378  -0.0721  -0.4927
## s.e.  0.3494   0.3095   0.1778   0.1660
##
## sigma^2 = 0.001452:  log likelihood = 197.72
## AIC=-385.45   AICc=-384.85   BIC=-372.08
##
## Training set error measures:
##                       ME     RMSE      MAE         MPE     MAPE      MASE
## Training set -0.2673434 8.837846 6.508219 -0.07860439 2.627682 0.2277666
```

```
##                     ACF1
## Training set 0.01347027
```

```
summary(fit3)
```

```
## Series: train
## ARIMA(0,1,1)(0,1,1)[12]
## Box Cox transformation: lambda= 0
##
## Coefficients:
##           ma1     sma1
##       -0.3424  -0.5405
## s.e.   0.1009   0.0877
##
## sigma^2 = 0.001432:  log likelihood = 197.51
## AIC=-389.02   AICc=-388.78   BIC=-381
##
## Training set error measures:
##                      ME     RMSE     MAE         MPE     MAPE      MASE
## Training set -0.2372088 8.835339 6.51704 -0.07508532 2.637955 0.2280753
##                     ACF1
## Training set 0.04249699
```

Dari ketiga model tersebut, model `fit3` memiliki nilai AIC dan BIC yang paling kecil yaitu AIC = -389.02 dan BIC = -381. Jadi, model final yang akan dipilih yaitu model `fit3` = ARIMA$(0, 1, 1)(0, 1, 1)_{12}$, yang berarti ln(Y_t) merupakan ARIMA$(0, 1, 1)(0, 1, 1)_{12}$.

3.4 Estimasi Parameter

Akan dilakukan estimasi parameter dari model `fit3`.

```
summary(fit3)
```

```
## Series: train
```

```
## ARIMA(0,1,1)(0,1,1)[12]
## Box Cox transformation: lambda= 0
##
## Coefficients:
##           ma1     sma1
##       -0.3424  -0.5405
## s.e.   0.1009   0.0877
##
## sigma^2 = 0.001432:  log likelihood = 197.51
## AIC=-389.02   AICc=-388.78   BIC=-381
##
## Training set error measures:
##                      ME     RMSE     MAE         MPE     MAPE      MASE
## Training set -0.2372088 8.835339 6.51704 -0.07508532 2.637955 0.2280753
##                     ACF1
## Training set 0.04249699
```

Dari hasil tersebut didapatkan:

$$\begin{aligned}\theta(x)\Theta(x)e_t &= (1+\hat{\theta}x)(1+\hat{\Theta}x^{12})e_t \\ &= e_t + \hat{\theta}xe_t + \hat{\Theta}x^{12}e_t + \hat{\theta}\hat{\Theta}x^{13}e_t \\ &= e_t + \hat{\theta}e_{t-1} + \hat{\Theta}e_{t-12} + \hat{\theta}\hat{\Theta}e_{t-13}\end{aligned}$$

Catatan: tanda + karena *default* persamaan MA pada R menggunakan tanda +. Misalkan $Z_t = \ln(Y_t)$, model ARIMA dari `fit3` yaitu

$$W_t = \nabla^1\nabla^1_{12}Z_t = e_t + \hat{\theta}e_{t-1} + \hat{\Theta}e_{t-12} + \hat{\theta}\hat{\Theta}e_{t-13}$$

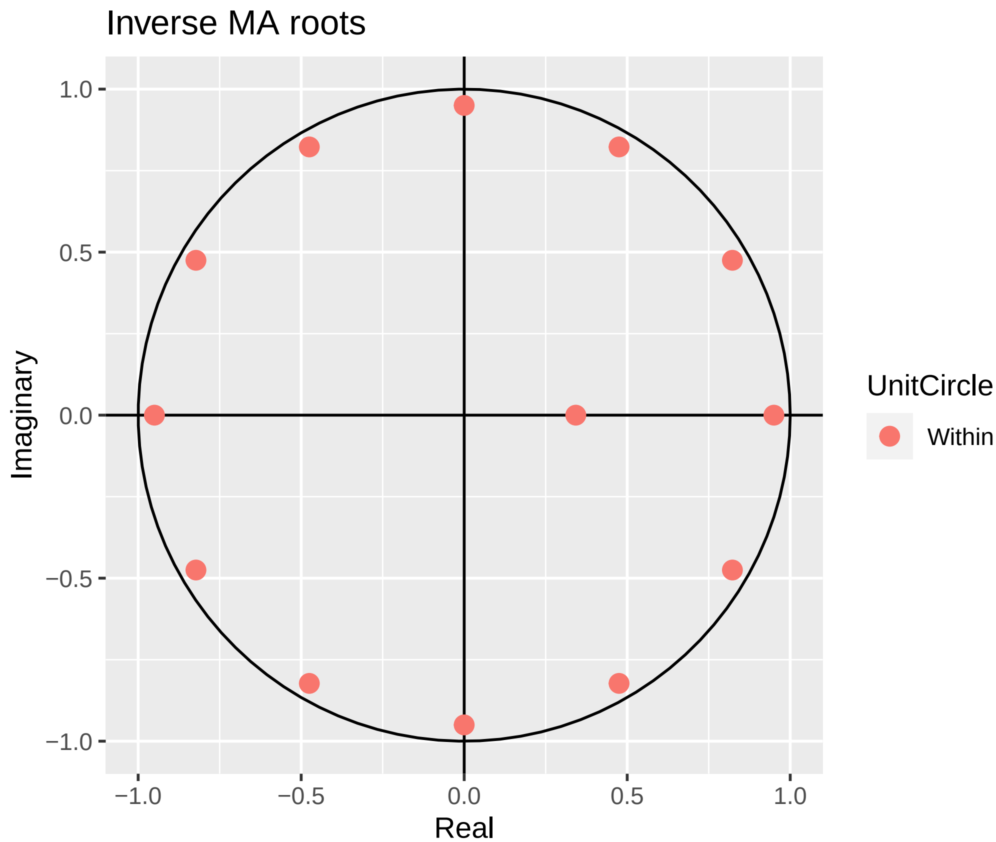
$$Z_t - Z_{t-1} - (Z_{t-12} - Z_{t-13}) = e_t + \hat{\theta}e_{t-1} + \hat{\Theta}e_{t-12} + \hat{\theta}\hat{\Theta}e_{t-13}$$

$$\ln(Y_t) - \ln(Y_{t-1}) - [\ln(Y_{t-12}) - \ln(Y_{t-13})] = e_t + \hat{\theta}e_{t-1} + \hat{\Theta}e_{t-12} + \hat{\theta}\hat{\Theta}e_{t-13}$$

$$\ln(Y_t) - \ln(Y_{t-1}) - [\ln(Y_{t-12}) - \ln(Y_{t-13})] = e_t - 0.3424e_{t-1} - 0.5405e_{t-12} + 0.3424 \times 0.5405e_{t-13}$$

$$\ln(Y_t) - \ln(Y_{t-1}) - [\ln(Y_{t-12}) - \ln(Y_{t-13})] = e_t - 0.3424e_{t-1} - 0.5405e_{t-12} + 0.1850672e_{t-13} \quad (2)$$

```
autoplot(fit3)
```



Gambar 12: Plot Inverse MA Roots

Terlihat bahwa model tersebut *invertible* karena semua *inverse* MA *root* berada di dalam *unit circle* pada bidang kompleks.

Uji Signifikansi Estimasi Parameter

```
coeftest(fit3)
```

```
##
## z test of coefficients:
##
##       Estimate Std. Error z value  Pr(>|z|)
## ma1  -0.342360   0.100917 -3.3925 0.0006926 ***
## sma1 -0.540532   0.087711 -6.1626 7.155e-10 ***
```

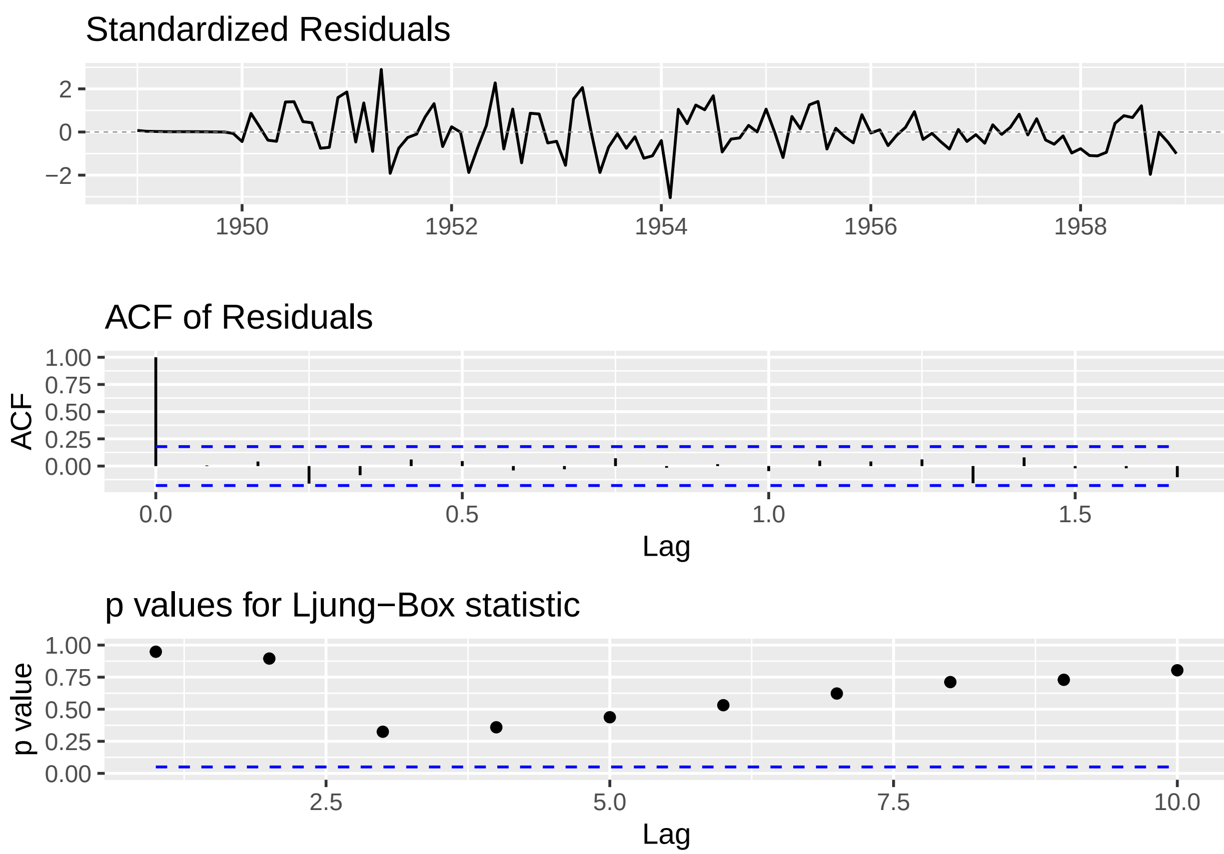
```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Terlihat bahwa kedua $p - value$ untuk masing-masing parameter $< \alpha = 0.05$, yang berarti bahwa kedua koefisien signifikan berbeda dengan 0.

3.5 Model Diagnostik

Akan dilakukan uji model diagnostik pada model di persamaan (1).

```
ggtsdiag(fit3)
```



Gambar 13: Summary Plot

Karena pada bagian spesifikasi model, plot ACF pada residual model `fit3` terdapat garis yang signifikan pada Lag = 23, akan dilakukan uji signifikansi menggunakan uji Ljung-Box.

H_0: Residual merupakan *white noise* atau independen dan distribusi identik (i.i.d)

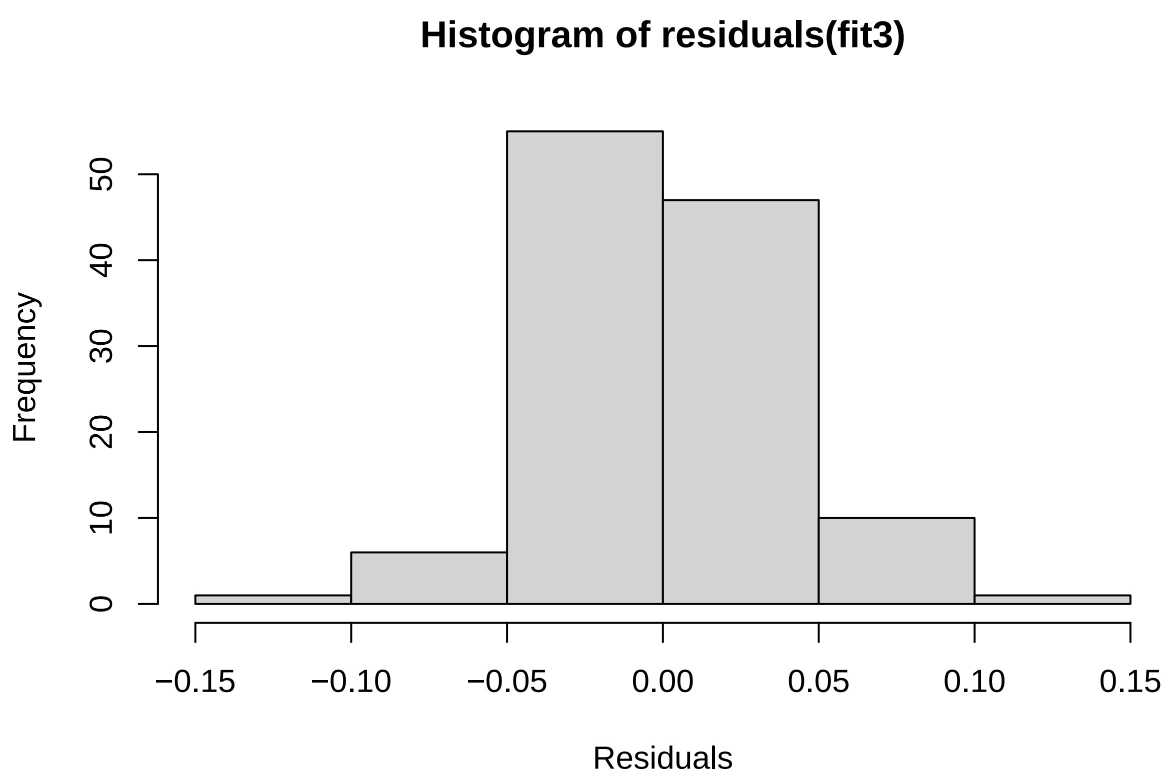
H_1: Residual menunjukkan hubungan korelasi (non-*random*)

```r
Box.test(fit3$residuals, lag = 30, type = "Ljung")
```

```
##
##  Box-Ljung test
##
## data:  fit3$residuals
## X-squared = 22.621, df = 30, p-value = 0.8305
```

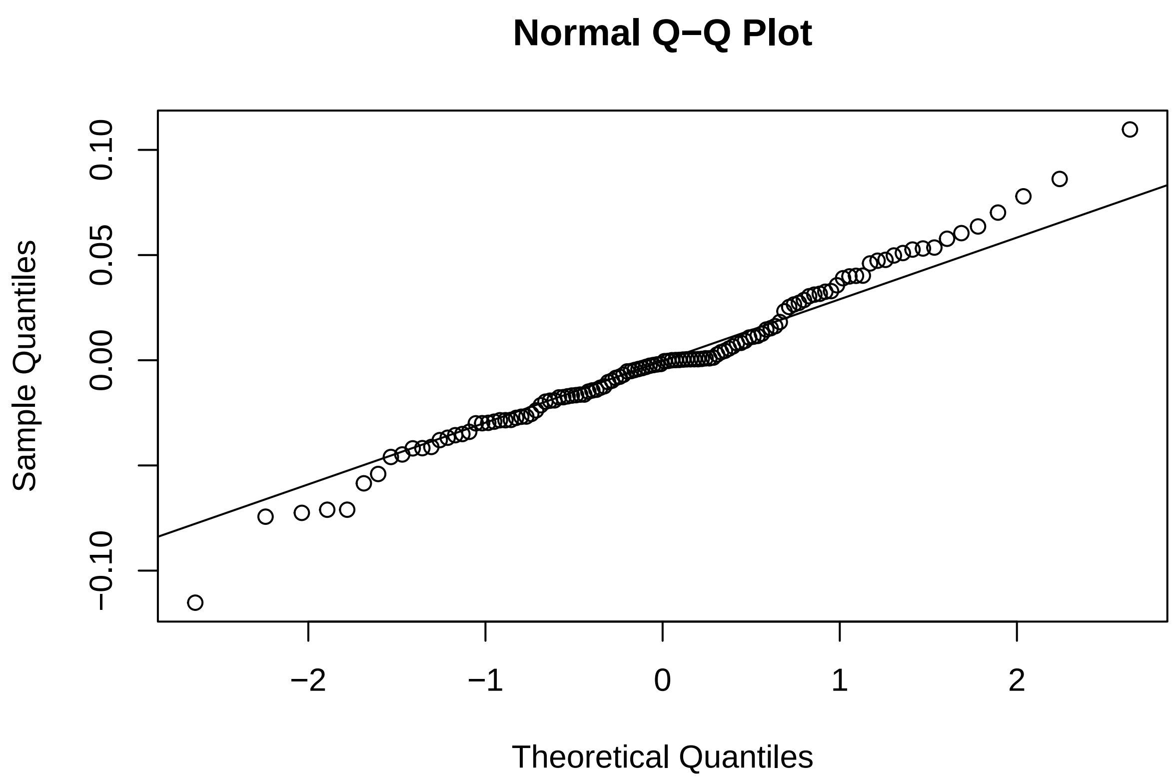
Dilakukan uji Ljung-Box sampai lag ke-30 agar melibatkan lag ke-27. Didapatkan $p - value = 0.8305 > \alpha = 0.05$ yang berarti H_0 tida ditolak. Jadi, residual dari model `fit3` merupakan *white noise*. Plot dari residual tersebut tampak berpusat di sekitar 0 sebagai tanpa pola, yang mana juga sesuai dengan karakteristik *white noise*.

```r
hist(residuals(fit3), xlab = 'Residuals')
```



Gambar 14: Histogram dari Residual

```
qqnorm(residuals(fit3))
qqline(residuals(fit3))
```



Gambar 15: Q-Q Plot dari Residual

Dari kedua plot tersebut, distribusi dari residual relatif tidak memenuhi distribusi normal. Akan dilakukan uji Shapiro-Wilk untuk memastikan hal tersebut.

H_0: Residual berdistribusi normal

H_1: Residual tidak berdistribusi normal

```
shapiro.test(residuals(fit3))
```

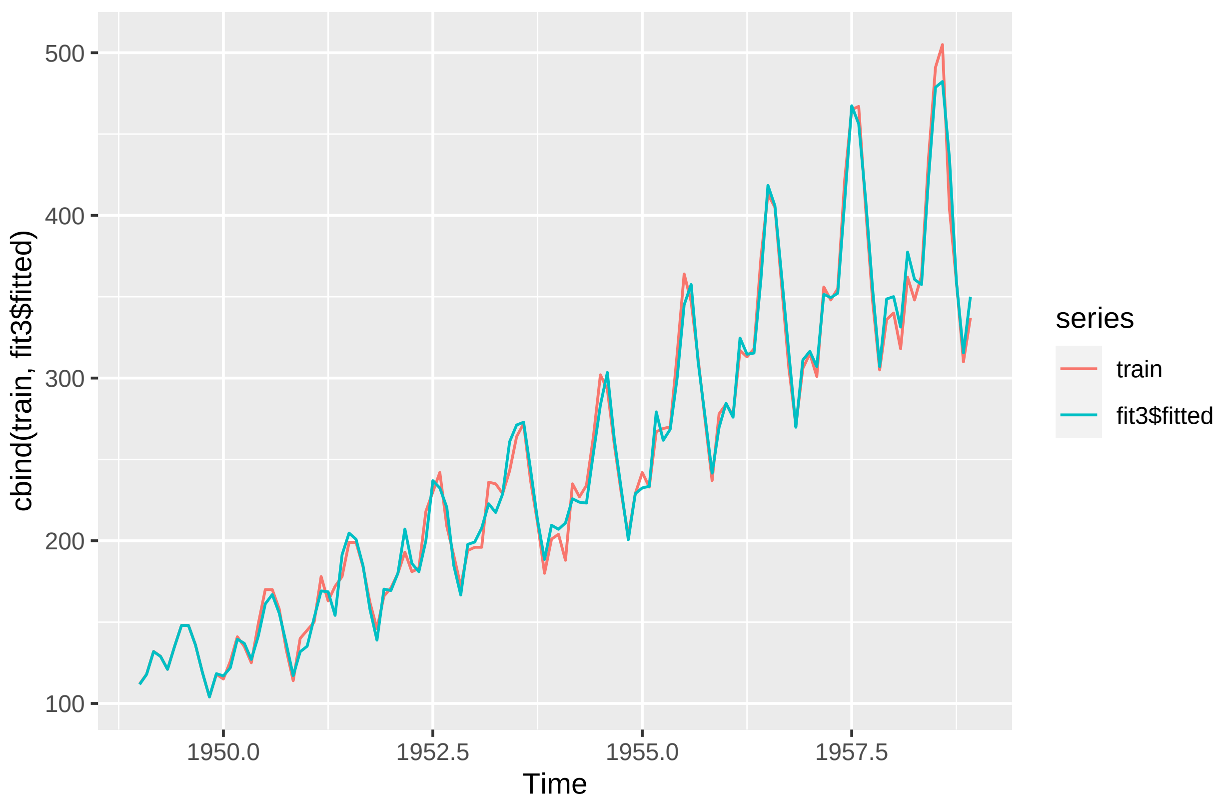
```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(fit3)
## W = 0.98288, p-value = 0.1316
```

Karena $p - value = 0.1316 > \alpha = 0.05$, maka H_0 tidak ditolak. Jadi, residual berdistribusi normal, walaupun secara visual tidak sangat "mirip" dengan distribusi normal. Hal tersebut mungkin dikarenakan jumlah sampel yang kurang banyak. Berdasarkan model diagnostik tersebut, model ARIMA$(0,1,1)(0,1,1)_{12}$ cukup sesuai. Selanjutnya, akan dilakukan *forecasting* dengan menggunakan model tersebut.

3.6 *Forecasting*

Akan dicek visual dari *fitted values* dari model ARIMA$(0,1,1)(0,1,1)_{12}$.

```
autoplot(cbind(train, fit3$fitted))
```



Gambar 16: Forecasting Plot

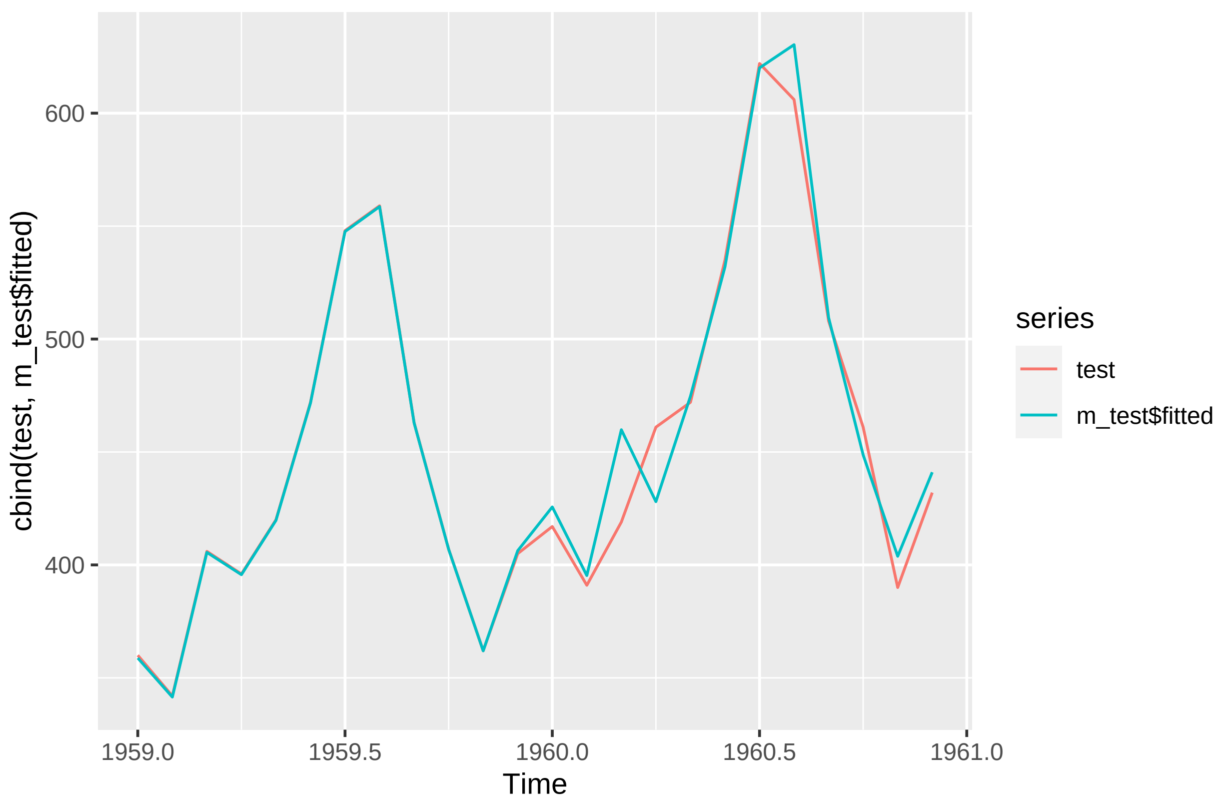
Terlihat bahwa dua grafik cukup berimpit yang menandakan model `fit3` cukup bagus dalam memodelkan data `train`. Akan dilakukan *forecasting* untuk data `test` menggunakan model yang telah dilatih pada data `train`.

```
m_test <- Arima(test, model = fit3)
accuracy(m_test)
```

```
##                      ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -2.16772 12.73486 6.701504 -0.5012175 1.478059 0.1401011
##                    ACF1
## Training set -0.3669897
```

Karena nilai MAPE = 1.478059 < 10 mengindikasikan bahwa peramalan (*forecasting*) yang dilakukan sangat akurat.

```
autoplot(cbind(test, m_test$fitted))
```

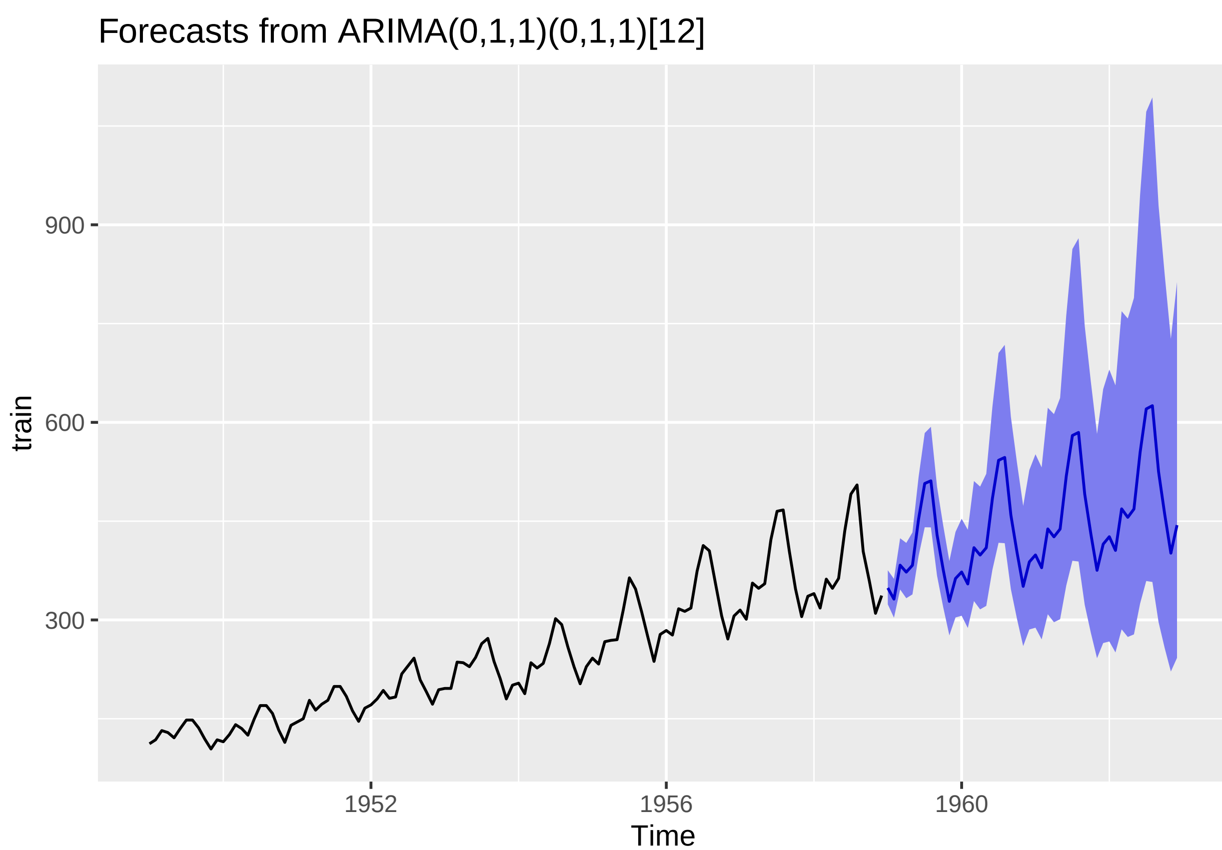


Gambar 17: Forecasting pada Data Test

Terlihat bahwa kedua plot berimpit yang menandakan kecocokan model yang tinggi. Selanjutnya, akan dilakukan *forecasting* untuk 4 tahun ke depan (*forecast origin* pada akhir waktu data `train`) dengan interval kepercayaan 95%, di mana h adalah periode horizon *forecasting* dalam bulan.

```
forecastAP <- forecast(fit3, level = c(95), h = 48)
autoplot(forecastAP)
```

Terlihat bahwa batas perkiraan (*forecast limit*) semakin melebar seiring bertambahnya waktu.



Gambar 18: Forecasting 4 Tahun ke Depan

PENUTUP

Kesimpulan

Berdasarkan hasil analisis yang telah dilakukan menggunakan data jumlah penumpang maskapai Amerika Serikat bulanan dari tahun 1949 hingga 1960 (`AirPassengers`) didapatkan kesimpulan sebagai berikut.

1. Model linier tidak cocok untuk memodelkan data ini karena tidak dapat menangkap efek musiman pada data. Model yang sesuai untuk data tersebut yaitu ln(`AirPassangers`) merupakan ARIMA$(0,1,1)(0,1,1)_{12}$, dengan taksiran model

$$\ln(Y_t) - \ln(Y_{t-1}) - [\ln(Y_{t-12}) - \ln(Y_{t-13})] = e_t - 0.3424e_{t-1} - 0.5405e_{t-12} + 0.1850672e_{t-13}.$$

2. Residual dari model pada persamaan (2) sudah memenuhi asumsi model diagnostik meskipun secara visual kurang menunjukkan kemiripan dengan distribusi normal.

3. *Forecasting* untuk data `test` menghasilkan nilai MAPE = 1.478059 < 10, yang mengindikasikan bahwa peramalan (*forecasting*) dilakukan dengan sangat akurat. Terlihat bahwa data asli dan `fitted values` berimpit pada *time plot*. Untuk *forecasting* 4 tahun ke depan, dimulai dari *forecast origin* yaitu akhir waktu data `train`, batas perkiraan (*forecast limit*) semakin melebar seiring bertambahnya waktu. Nilai *forecasting* menunjukkan tren meningkat yang mengindikasikan kemungkinan kenaikan jumlah penumpang pesawat bulanan pada maskapai penerbangan Amerika Serikat di masa depan.

DAFTAR PUSTAKA

[1] Cryer, Jonathan D., & Chan, Kung-Sik. (2008). *Time Series Analysis with Application in R*. Springer Text in Statistics.

[2] Federal Aviation Administration. (2023). *FAA Contributors to ATO by the Numbers*. Diakses dari https://www.faa.gov/air_traffic/by_the_numbers/media/Air_Traffic_by_the_Numbers_2023.pdf

[3] Makridakis, S. G., Wheelwright, S. C., & Hyndman, R. J. (1997). *Forecasting Methods and Application*. Wiley.

[4] Wulandari, Rosita Ayu, & Gernowo, Rahmat. (2019). *Metode Autoregressive Integrated Moving Average (ARIMA) dan Metode Adaptive Neuro Fuzzy Inference System (ANFIS) dalam Analisis Curah Hujan*. Berkala Fisika: Volume 1 No. 1.

Link **File**

- kaggle.com. Air Passengers. Data diambil dari *link* berikut. https://www.kaggle.com/datasets/chirag19/air-passengers?resource=download&select=AirPassengers.csv
- File *project*, R, dan lain-lain dapat diakses pada *link* berikut. https://drive.google.com/drive/folders/1-2GksgEIB5PltxUG6P7uGq2-BlBVsJeG?usp=drive_link

LAMPIRAN

Lampiran I. Data Penelitian

passanger	month	passanger	month	passanger	month	passanger	month
112	Jan 1949	171	Jan 1952	242	Jan 1955	340	Jan 1958
118	Feb 1949	180	Feb 1952	233	Feb 1955	318	Feb 1958
132	Mar 1949	193	Mar 1952	267	Mar 1955	362	Mar 1958
129	Apr 1949	181	Apr 1952	269	Apr 1955	348	Apr 1958
121	May 1949	183	May 1952	270	May 1955	363	May 1958
135	Jun 1949	218	Jun 1952	315	Jun 1955	435	Jun 1958
148	Jul 1949	230	Jul 1952	364	Jul 1955	491	Jul 1958
148	Aug 1949	242	Aug 1952	347	Aug 1955	505	Aug 1958
136	Sep 1949	209	Sep 1952	312	Sep 1955	404	Sep 1958
119	Oct 1949	191	Oct 1952	274	Oct 1955	359	Oct 1958
104	Nov 1949	172	Nov 1952	237	Nov 1955	310	Nov 1958
118	Dec 1949	194	Dec 1952	278	Dec 1955	337	Dec 1958
115	Jan 1950	196	Jan 1953	284	Jan 1956	360	Jan 1959
126	Feb 1950	196	Feb 1953	277	Feb 1956	342	Feb 1959
141	Mar 1950	236	Mar 1953	317	Mar 1956	406	Mar 1959
135	Apr 1950	235	Apr 1953	313	Apr 1956	396	Apr 1959
125	May 1950	229	May 1953	318	May 1956	420	May 1959
149	Jun 1950	243	Jun 1953	374	Jun 1956	472	Jun 1959
170	Jul 1950	264	Jul 1953	413	Jul 1956	548	Jul 1959
170	Aug 1950	272	Aug 1953	405	Aug 1956	559	Aug 1959
158	Sep 1950	237	Sep 1953	355	Sep 1956	463	Sep 1959
133	Oct 1950	211	Oct 1953	306	Oct 1956	407	Oct 1959
114	Nov 1950	180	Nov 1953	271	Nov 1956	362	Nov 1959
140	Dec 1950	201	Dec 1953	306	Dec 1956	405	Dec 1959
145	Jan 1951	204	Jan 1954	315	Jan 1957	417	Jan 1960
150	Feb 1951	188	Feb 1954	301	Feb 1957	391	Feb 1960
178	Mar 1951	235	Mar 1954	356	Mar 1957	419	Mar 1960
163	Apr 1951	227	Apr 1954	348	Apr 1957	461	Apr 1960

passanger	month	passanger	month	passanger	month	passanger	month
172	May 1951	234	May 1954	355	May 1957	472	May 1960
178	Jun 1951	264	Jun 1954	422	Jun 1957	535	Jun 1960
199	Jul 1951	302	Jul 1954	465	Jul 1957	622	Jul 1960
199	Aug 1951	293	Aug 1954	467	Aug 1957	606	Aug 1960
184	Sep 1951	259	Sep 1954	404	Sep 1957	508	Sep 1960
162	Oct 1951	229	Oct 1954	347	Oct 1957	461	Oct 1960
146	Nov 1951	203	Nov 1954	305	Nov 1957	390	Nov 1960
166	Dec 1951	229	Dec 1954	336	Dec 1957	432	Dec 1960

PERSENTASE KONTRIBUSI KELOMPOK

No.	Nama	NPM	Kontribusi	Tingkat Kontribusi
1.	Diki Wahyudi	2106709131	Mengerjakan *project*	100%
2.	Medifa Puspaningrum	2106634944	Mengerjakan *project*	100%
3.	Rachel Thyffani Margaretha	2106726900	Mengerjakan *project*	100%
4.	Vesya Padmadewi	2106726926	Mengerjakan *project*	100%
5.	Whitney	2106700946	Mengerjakan *project*	100%