Latihan Pertemuan 8 : Model Seasonal ARIMA (SARIMA)

2024-11-03

# Packages

library(tidyverse)  
library(forecast)  
library(TSA)  
library(aTSA)  
library(car)  
library(lmtest)  
library(tidyr)

# Impor Data

Data yang digunakan adalah data bulanan food and beverages dari bulan Januari 2005 hingga Desember 2019 di United States. Data diambil dari Kaggle dan berjumlah 180 baris data.

musiman <- read.csv("C:/Users/hp/Documents/KULIAH/SEMESTER 5/MPDW/PRAKTIKUM/PERTEMUAN 8/datamusiman.csv", sep = ";")  
head(musiman)

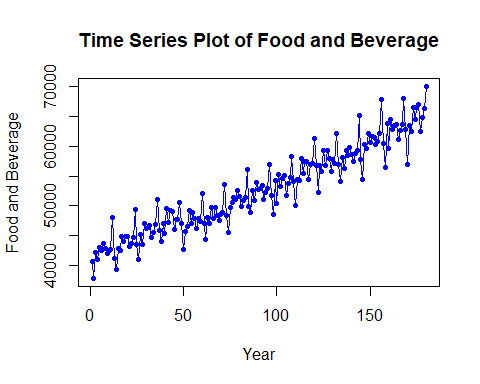
## Food  
## 1 40592  
## 2 37829  
## 3 42159  
## 4 40917  
## 5 42964  
## 6 42445

class(musiman)

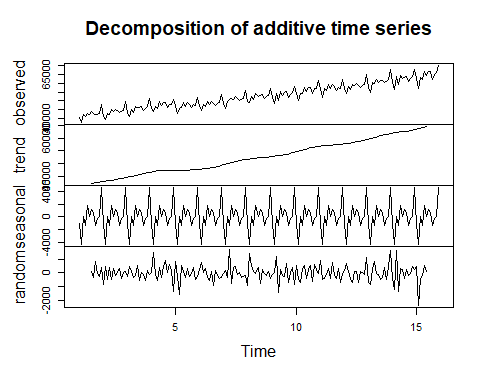
## [1] "data.frame"

# Eksplorasi Data

ts.plot(musiman, type="l", xlab = "Year", ylab="Food and Beverage", col="blue")  
title(main = "Time Series Plot of Food and Beverage", cex.sub = 0.8)  
points(musiman, pch = 20, col = "blue")

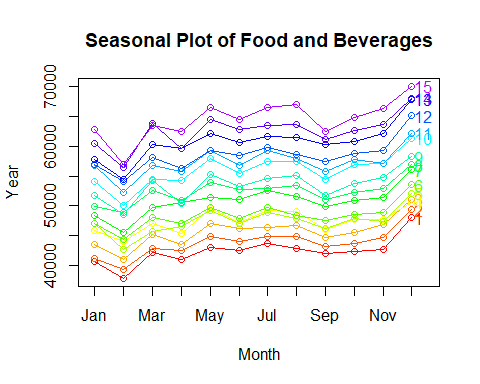


musiman <- ts(musiman, frequency = 12) # Untuk data bulanan  
dec.musiman <- decompose(musiman)  
plot(dec.musiman)



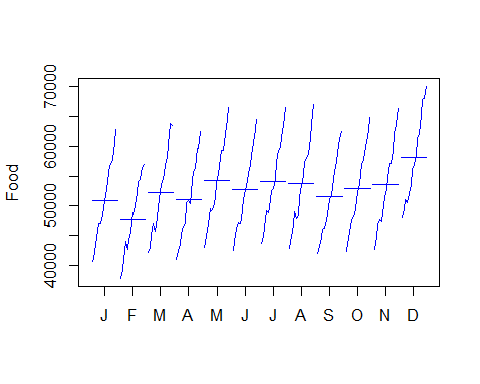
Secara eksplorasi, terlihat adanya kecenderungan data memiliki tren naik dan perilaku berulang kecenderungan musiman dalam deret tersebut. Kecenderungan musiman dapat dilihat dengan lebih jelas dengan menampilkan deret waktu per tahun.

seasonplot(musiman,12,main="Seasonal Plot of Food and Beverages", ylab="Year",  
 year.labels = TRUE, col=rainbow(18))

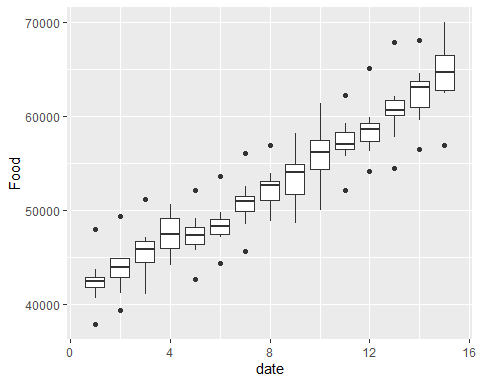


Gambar menunjukkan bahwa food and beverages tinggi pada bulan Januari, Maret, Mei, Agustus, Desember, dan rendah pada bulan Februari, April, Juni, September. Perilaku tersebut terus berulang dari tahun ke tahun.

monthplot(musiman,ylab="Food", col="blue")



frame<-data.frame(Food=as.matrix(musiman), date=lubridate::year(zoo::as.Date(musiman)))  
  
library(ggplot2)  
ggplot(frame,aes(y=Food,x=date,group=date))+  
 geom\_boxplot()



Berdasarkan hasil plot di atas dapat terlihat bahwa data memiliki pola yang hampir sama dari tahun ke tahun sehingga dapat disimpulkan bahwa periode musimannya adalah 12. Selain itu, apabila dilihat dari boxplot, terlihat bahwa data cenderung homogen dari tahun ke tahun meskupun terdapat beberapa pencilan. Untuk memastikan bahwa data homogen akan dilakukan uji homogenitas dengan fligner.test.

## Uji Homogenitas

Uji asumsi formal terhadap kehomogenan ragam yang digunakan yaitu *Fligner-Killen test*, dimana:

: Ragam homogen

: Ragam tidak homogen

library(car)  
fligner.test(Food ~ date, data=frame)

##   
## Fligner-Killeen test of homogeneity of variances  
##   
## data: Food by date  
## Fligner-Killeen:med chi-squared = 7.6117, df = 14, p-value = 0.9086

Berdasarkan hasil uji *Fligner-Killeen* dengan menggunakan taraf signifikansi didapatkan *p-value* sebesar 0.9086. sehingga tak tolak atau dengan kata lain **ragam data sudah stasioner**.

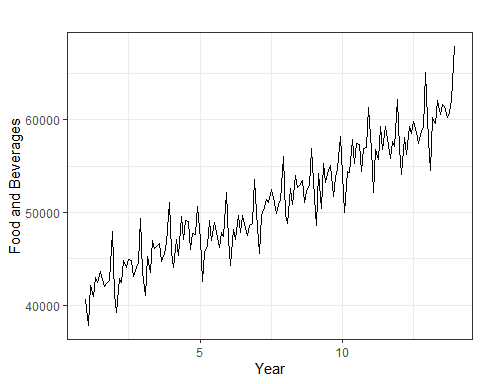
# Pembagian Data

Pembagian data dilakukan dengan mengambil sekitar 86% data awal (156 observasi) sebagai data latih dan 14% sisanya (24 observasi) sebagai data uji.

train.ts <- subset(musiman,start=1,end=156)  
test.ts <- subset(musiman,start=157,end=180)

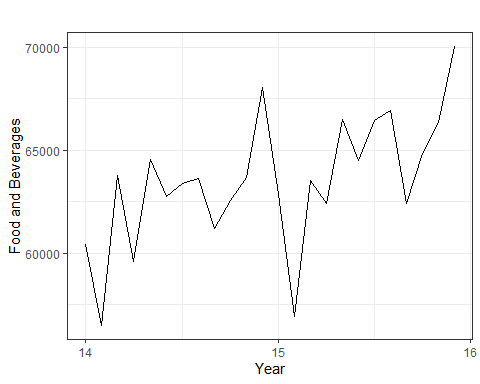
### Plot Data Latih

autoplot(train.ts) + theme\_bw() + xlab("Year") + ylab("Food and Beverages")



### Plot Data Uji

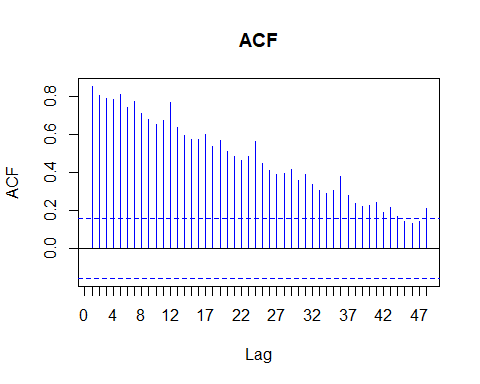
autoplot(test.ts) + theme\_bw() + xlab("Year") + ylab("Food and Beverages")



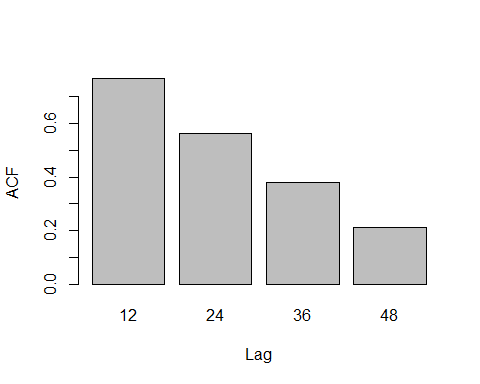
# Non-Seasonal ARIMA

## Kestasioneran Data

acf0 <- acf(train.ts,main="ACF",lag.max=48,xaxt="n", col="blue")  
axis(1, at=0:48/12, labels=0:48)



acf0$lag <- acf0$lag \* 12  
acf0.1 <- as.data.frame(cbind(acf0$acf,acf0$lag))  
acf0.2 <- acf0.1[which(acf0.1$V2%%12==0),]  
barplot(height = acf0.2$V1,   
names.arg=acf0.2$V2, ylab="ACF", xlab="Lag")



Berdasarkan plot deret sebelumnya diketahui bahwa perilaku deret berulang setiap tahun, atau dikatakan bahwa deret memiliki periode musiman bulanan, sehingga . Perhatikan nilai fungsi autokorelasi pada lag-lag musiman (lag 12, 24, 36,...) dalam plot ACF contoh di atas. Tampak bahwa nilai autokorelasi pada lag-lag tersebut memiliki hubungan yang kuat. Bagaimanapun juga, plot ACF contoh meluruh secara perlahan dan membentuk pola gelombang kosinus teredam, yang menandakan ketidakstasioneran (plot deret juga menunjukkan adanya trend naik dalam deret).

tseries::adf.test(train.ts)

## Warning in tseries::adf.test(train.ts): p-value smaller than printed p-value

##   
## Augmented Dickey-Fuller Test  
##   
## data: train.ts  
## Dickey-Fuller = -4.4893, Lag order = 5, p-value = 0.01  
## alternative hypothesis: stationary

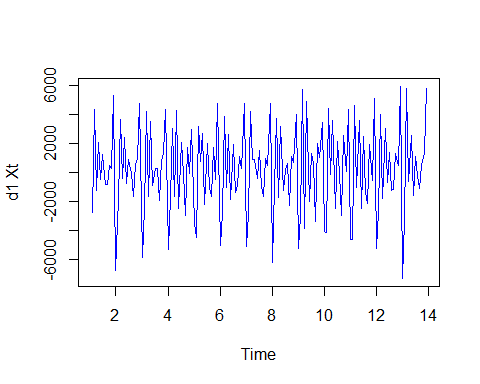
: Data tidak stasioner dalam rataan

: Data stasioner dalam rataan

Berdasarkan uji ADF tersebut, didapat *p-value* sebesar 0.01 yang lebih kecil dari taraf nyata 5% sehingga tolak dan menandakan bahwa data stasioner dalam rataan. Hal ini tidak sesuai dengan hasil eksplorasi menggunakan plot time series dan plot ACF, sehingga ketidakstasioneran model kedepannya harus tetap ditangani.

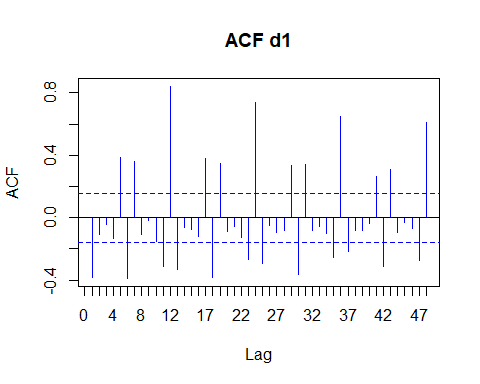
## Pembedaan

d1 <- diff(train.ts)  
ts.plot(d1, type="l", ylab="d1 Xt", col="blue")

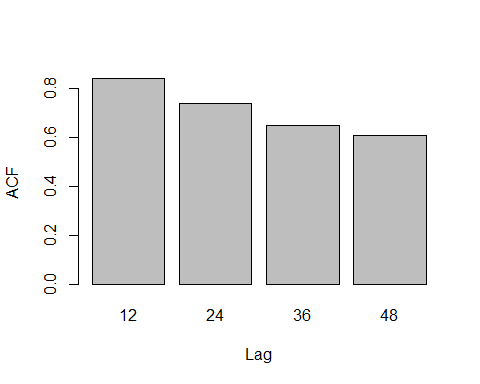


*Differencing non-seasonal* jika dilihat berdasarkan plot di atas berhasil mengatasi ketidakstasioneran dalam rataan untuk komponen *non-seasonal*.

acf1 <- acf(d1,lag.max=48,xaxt="n", main="ACF d1", col="blue")  
axis(1, at=0:48/12, labels=0:48)



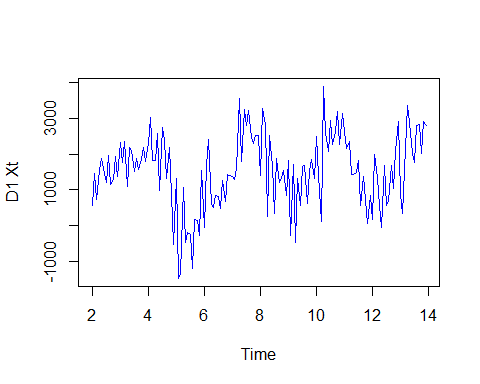
acf2 <- acf1$lag <- acf1$lag \* 12  
acf1.1 <- as.data.frame(cbind(acf1$acf,acf1$lag))  
acf1.2 <- acf1.1[which(acf1.1$V2%%12==0),]  
barplot(height = acf1.2$V1, names.arg=acf1.2$V2, ylab="ACF", xlab="Lag")



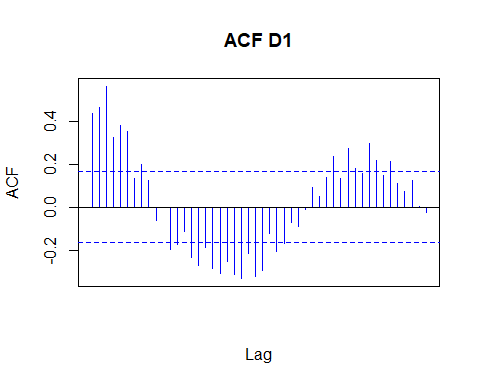
Plot ACF data *non-seasonal differencing* mengkonfirmasi kestasioneran komponen *non-seasonal* (namun perhatikan lag 12,24, dst), pada *series* *seasonal* **belum stasioner**. Hal ini menunjukkan adanya kecenderungan musiman

# Seasonal ARIMA

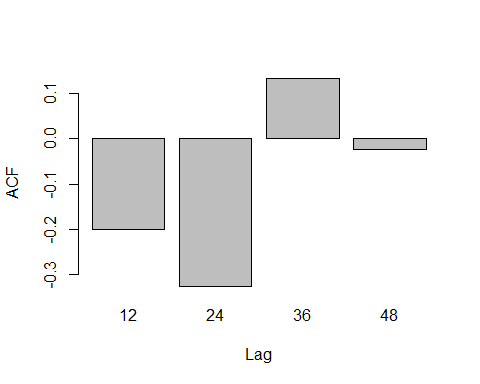
D1 <- diff(train.ts,12)  
ts.plot(D1, type="l", ylab="D1 Xt", col="blue")



acf2<-acf(D1,lag.max=48,xaxt="n", main="ACF D1", col="blue")



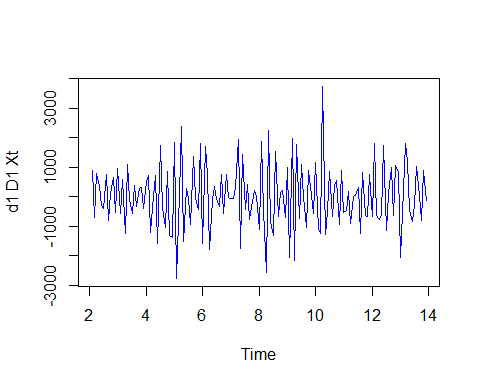
acf2$lag <- acf2$lag \* 12  
acf2.1 <- as.data.frame(cbind(acf2$acf,acf2$lag))  
acf2.2 <- acf2.1[which(acf2.1$V2%%12==0),]  
barplot(height = acf2.2$V1, names.arg=acf2.2$V2, ylab="ACF", xlab="Lag")



*Non-seasonal differencing* D = 12 berhasil mengatasi ketidakstasioneran dalam rataan untuk komponen *seasonal*nya (namun tidak untuk komponen *non-seasonal*nya).

Untuk menghilangkan kecenderungan musiman dilakukan pembedaan musiman terhadap deret hasil pembedaan pertama.

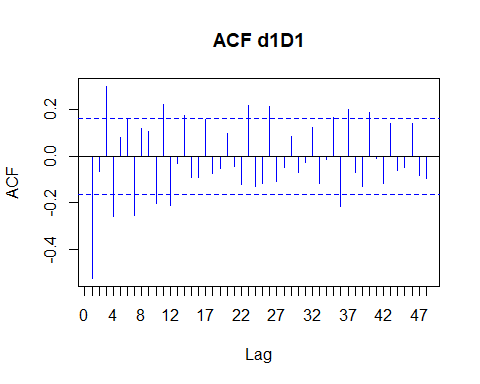
d1D1 <- diff(D1)  
ts.plot(d1D1, type="l", ylab="d1 D1 Xt", col="blue")



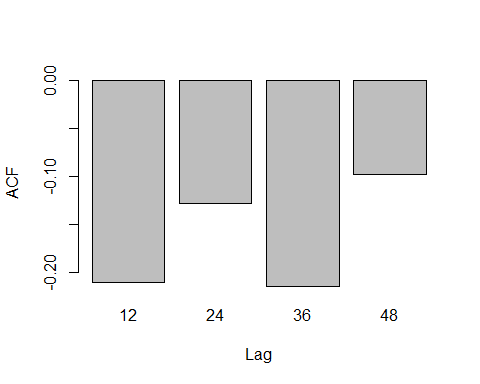
Setelah pembedaan pertama dan pembedaan musiman tampak bahwa deret sudah tidak memiliki kecenderungan apapun. Selanjutnya penentuan ordo *p*, *q* dan *P*, *Q* dapat dilakukan menggunakan plot ACF dan PACF contoh dari deret hasil pembedaan pertama dan pembedaan musiman tersebut.

## Identifikasi Model

acf3 <- acf(d1D1,lag.max=48,xaxt="n", main="ACF d1D1", col="blue")  
axis(1, at=0:48/12, labels=0:48)

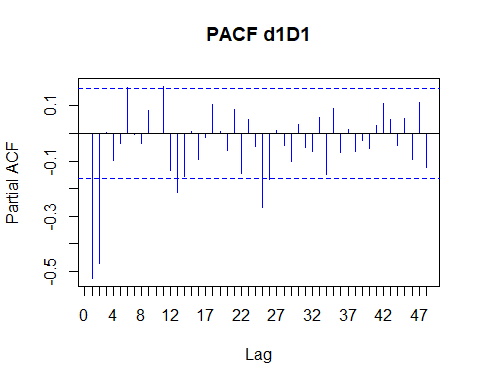


acf3$lag <- acf3$lag \* 12  
acf3.1 <- as.data.frame(cbind(acf3$acf,acf3$lag))  
acf3.2 <- acf3.1[which(acf3.1$V2%%12==0),]  
barplot(height = acf3.2$V1, names.arg=acf3.2$V2, ylab="ACF",   
xlab="Lag")

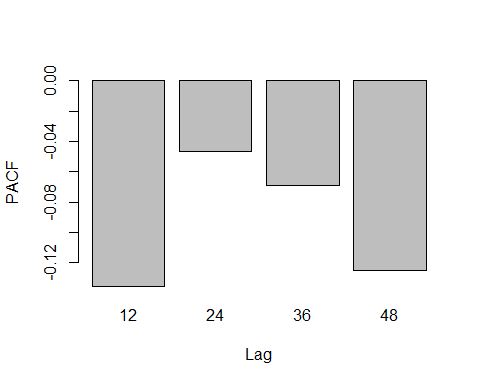


Berdasarkan plot ACF contoh lag 1 signifikan sehingga dipilih ordo *q*=1 , dan lag musiman tidak terlihat model AR yang terbentuk karena cenderung *tails-off* sehingga Q=0.

pacf3 <- pacf(d1D1,lag.max=48,xaxt="n", main="PACF d1D1", col="blue")  
axis(1, at=0:48/12, labels=0:48)



pacf3$lag <- pacf3$lag \* 12  
pacf3.1 <- as.data.frame(cbind(pacf3$acf,pacf3$lag))  
pacf3.2 <- pacf3.1[which(pacf3.1$V2%%12==0),]  
barplot(height = pacf3.2$V1, names.arg=pacf3.2$V2, ylab="PACF", xlab="Lag")



Plot PACF contoh menunjukkan *cuts-off* pada lag-2 sehingga ordo *p*=2, sementara pada pola musimannya lag 12 cenderung cut off sehingga P=1

Model musiman yang dipilih untuk deret konsentrasi karbon dioksida adalah , , . Ingat kembali bahwa model yang digunakan bersifat tentatif dan dapat berubah saat diagnostik model.

**EACF**

TSA::eacf(d1D1)

## AR/MA  
## 0 1 2 3 4 5 6 7 8 9 10 11 12 13  
## 0 x o x x o o x o o x x x o x   
## 1 x o x x o o x o o o o x o x   
## 2 o x o o o o o o o o o x o o   
## 3 o x o x o o o o o o o x o o   
## 4 x x o x o o o o o o o x o o   
## 5 x x x x o o o o o o o x x x   
## 6 o x x o o o o o o o o x o o   
## 7 o o x o o o o o o o o x o o

Karena, kedua komponen telah stasioner. Identifikasi komponen *non-seasonal* adalah ARIMA(2,1,2), ARIMA(2,1,3). Identifikasi komponen *seasonal* adalah , sehingga model tentatif yang diperoleh adalah:

## Pendugaan Parameter

#ARIMA(2,1,0)x(1,1,0)12  
tmodel1 <- Arima(train.ts,order=c(2,1,0),seasonal=c(1,1,0))  
summary(tmodel1)

## Series: train.ts   
## ARIMA(2,1,0)(1,1,0)[12]   
##   
## Coefficients:  
## ar1 ar2 sar1  
## -0.8264 -0.5230 -0.3572  
## s.e. 0.0725 0.0725 0.0835  
##   
## sigma^2 = 587269: log likelihood = -1152.43  
## AIC=2312.86 AICc=2313.15 BIC=2324.71  
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 33.26157 725.9727 542.4426 0.05183726 1.052167 0.3348942  
## ACF1  
## Training set 0.00655883

lmtest::coeftest(tmodel1)

##   
## z test of coefficients:  
##   
## Estimate Std. Error z value Pr(>|z|)   
## ar1 -0.826409 0.072510 -11.3972 < 2.2e-16 \*\*\*  
## ar2 -0.522959 0.072470 -7.2162 5.347e-13 \*\*\*  
## sar1 -0.357207 0.083484 -4.2788 1.879e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#ARIMA(0,1,2)x(1,1,0)12  
tmodel2 <- Arima(train.ts,order=c(0,1,2),seasonal=c(1,1,0))  
summary(tmodel2)

## Series: train.ts   
## ARIMA(0,1,2)(1,1,0)[12]   
##   
## Coefficients:  
## ma1 ma2 sar1  
## -0.9111 0.3211 -0.3195  
## s.e. 0.0856 0.0814 0.0834  
##   
## sigma^2 = 631254: log likelihood = -1157.37  
## AIC=2322.73 AICc=2323.02 BIC=2334.59  
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 34.81516 752.6686 563.7383 0.05373695 1.093274 0.3480418  
## ACF1  
## Training set 0.02536796

lmtest::coeftest(tmodel2)

##   
## z test of coefficients:  
##   
## Estimate Std. Error z value Pr(>|z|)   
## ma1 -0.911053 0.085585 -10.6451 < 2.2e-16 \*\*\*  
## ma2 0.321129 0.081385 3.9458 7.954e-05 \*\*\*  
## sar1 -0.319537 0.083434 -3.8298 0.0001282 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#ARIMA(2,1,2)x(1,1,0)12  
tmodel3 <- Arima(train.ts,order=c(2,1,2),seasonal=c(1,1,0))  
summary(tmodel3)

## Series: train.ts   
## ARIMA(2,1,2)(1,1,0)[12]   
##   
## Coefficients:  
## ar1 ar2 ma1 ma2 sar1  
## -0.8931 -0.5249 0.0742 -0.0681 -0.3532  
## s.e. 0.1407 0.0985 0.1586 0.1230 0.0839  
##   
## sigma^2 = 594114: log likelihood = -1152.22  
## AIC=2316.44 AICc=2317.06 BIC=2334.22  
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 33.68291 724.957 540.2713 0.05260186 1.048458 0.3335537  
## ACF1  
## Training set -0.002099447

lmtest::coeftest(tmodel3)

##   
## z test of coefficients:  
##   
## Estimate Std. Error z value Pr(>|z|)   
## ar1 -0.893115 0.140675 -6.3488 2.170e-10 \*\*\*  
## ar2 -0.524880 0.098494 -5.3290 9.874e-08 \*\*\*  
## ma1 0.074166 0.158594 0.4676 0.6400   
## ma2 -0.068137 0.123033 -0.5538 0.5797   
## sar1 -0.353151 0.083912 -4.2086 2.570e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#ARIMA(2,1,3)x(1,1,0)12  
tmodel4 <- Arima(train.ts,order=c(2,1,3),seasonal=c(1,1,0))  
summary(tmodel4)

## Series: train.ts   
## ARIMA(2,1,3)(1,1,0)[12]   
##   
## Coefficients:  
## ar1 ar2 ma1 ma2 ma3 sar1  
## -1.1561 -0.9971 0.5305 0.3501 -0.5834 -0.4248  
## s.e. 0.0070 0.0042 0.0632 0.0705 0.0608 0.0782  
##   
## sigma^2 = 493772: log likelihood = -1142.26  
## AIC=2298.52 AICc=2299.35 BIC=2319.26  
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 32.25224 658.508 472.8635 0.05036343 0.9237744 0.2919373  
## ACF1  
## Training set -0.07357871

lmtest::coeftest(tmodel4)

##   
## z test of coefficients:  
##   
## Estimate Std. Error z value Pr(>|z|)   
## ar1 -1.1561244 0.0069541 -166.2507 < 2.2e-16 \*\*\*  
## ar2 -0.9970522 0.0041564 -239.8863 < 2.2e-16 \*\*\*  
## ma1 0.5305223 0.0631696 8.3984 < 2.2e-16 \*\*\*  
## ma2 0.3500587 0.0704568 4.9684 6.750e-07 \*\*\*  
## ma3 -0.5834457 0.0608073 -9.5950 < 2.2e-16 \*\*\*  
## sar1 -0.4248081 0.0782237 -5.4307 5.614e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#ARIMA(1,1,1)x(1,1,0)12  
tmodel5 <- Arima(train.ts,order=c(1,1,1),seasonal=c(1,1,0))  
summary(tmodel5)

## Series: train.ts   
## ARIMA(1,1,1)(1,1,0)[12]   
##   
## Coefficients:  
## ar1 ma1 sar1  
## -0.2706 -0.5485 -0.2894  
## s.e. 0.1009 0.0753 0.0824  
##   
## sigma^2 = 657622: log likelihood = -1160.1  
## AIC=2328.19 AICc=2328.48 BIC=2340.04  
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 36.77762 768.2275 582.5693 0.05720869 1.129692 0.3596677  
## ACF1  
## Training set -0.04773109

lmtest::coeftest(tmodel5)

##   
## z test of coefficients:  
##   
## Estimate Std. Error z value Pr(>|z|)   
## ar1 -0.270640 0.100898 -2.6823 0.007311 \*\*   
## ma1 -0.548462 0.075294 -7.2843 3.234e-13 \*\*\*  
## sar1 -0.289438 0.082390 -3.5130 0.000443 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

AICKandidatModel <- c(tmodel1$aic, tmodel2$aic, tmodel3$aic,  
 tmodel4$aic, tmodel5$aic)  
AICcKandidatModel <- c(tmodel1$aicc, tmodel2$aicc, tmodel3$aicc,  
 tmodel4$aicc, tmodel5$aicc)  
BICKandidatModel <- c(tmodel1$bic, tmodel2$bic, tmodel3$bic,  
 tmodel4$bic, tmodel5$bic)  
KandidatModelARIMA <- c("ARIMA(2,1,0)(1,1,0)12", "ARIMA(0,1,2)(1,1,0)12",  
 "ARIMA(2,1,2)(1,1,0)12", "ARIMA(2,1,3)(1,1,0)12",  
 "ARIMA(1,1,1)(1,1,0)12")  
compmodelARIMA <- cbind(KandidatModelARIMA, AICKandidatModel,  
 AICcKandidatModel, BICKandidatModel)  
colnames(compmodelARIMA) <- c("Kandidat Model", "Nilai AIC",   
 "Nilai AICc", "Nilai BIC")  
compmodelARIMA <- as.data.frame(compmodelARIMA)  
compmodelARIMA

## Kandidat Model Nilai AIC Nilai AICc Nilai BIC  
## 1 ARIMA(2,1,0)(1,1,0)12 2312.85756456884 2313.14741964131 2324.70894308988  
## 2 ARIMA(0,1,2)(1,1,0)12 2322.73401238392 2323.02386745638 2334.58539090496  
## 3 ARIMA(2,1,2)(1,1,0)12 2316.44184820616 2317.05949526498 2334.21891598771  
## 4 ARIMA(2,1,3)(1,1,0)12 2298.51740642984 2299.34703605947 2319.25731884166  
## 5 ARIMA(1,1,1)(1,1,0)12 2328.19258062385 2328.48243569631 2340.04395914489

Model **terbaik** berdasarkan nilai **AIC dan AICc terkecil** dari kandidat model **yaitu** dengan semua parameter yang signfikan

model.auto.arima <- auto.arima(train.ts)  
summary(model.auto.arima)

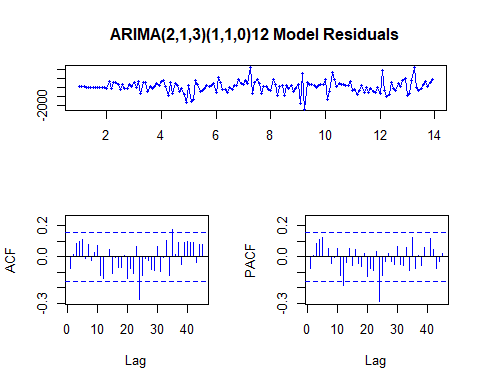
## Series: train.ts   
## ARIMA(3,0,0)(0,1,1)[12] with drift   
##   
## Coefficients:  
## ar1 ar2 ar3 sma1 drift  
## 0.0952 0.2793 0.5062 -0.8841 128.7331  
## s.e. 0.0741 0.0730 0.0753 0.1301 7.9905  
##   
## sigma^2 = 421225: log likelihood = -1143.3  
## AIC=2298.61 AICc=2299.22 BIC=2316.42  
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set -11.06658 612.6358 448.8154 -0.04804293 0.8663892 0.2770905  
## ACF1  
## Training set 0.01270017

lmtest::coeftest(model.auto.arima)

##   
## z test of coefficients:  
##   
## Estimate Std. Error z value Pr(>|z|)   
## ar1 0.095160 0.074111 1.2840 0.1991350   
## ar2 0.279326 0.072984 3.8272 0.0001296 \*\*\*  
## ar3 0.506238 0.075341 6.7193 1.826e-11 \*\*\*  
## sma1 -0.884118 0.130125 -6.7944 1.088e-11 \*\*\*  
## drift 128.733103 7.990462 16.1108 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Diagnostik Model

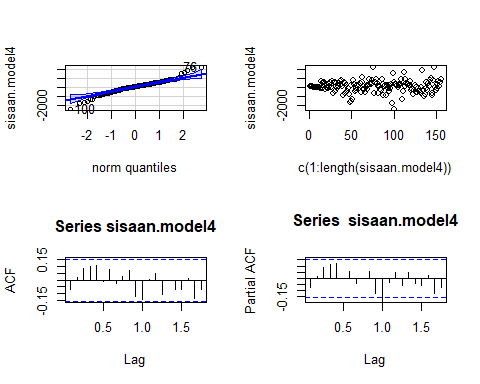
tsdisplay(residuals(tmodel4), lag.max=45,   
 main='ARIMA(2,1,3)(1,1,0)12 Model Residuals', col="blue")



#Eksplorasi  
sisaan.model4 <- tmodel4$residuals  
par(mfrow=c(2,2))  
car::qqPlot(sisaan.model4)

## [1] 100 76

plot(c(1:length(sisaan.model4)),sisaan.model4)  
acf(sisaan.model4)  
pacf(sisaan.model4)



par(mfrow = c(1,1))

Berdasarkan plot di atas terlihat bahwa sisaan mengikuti sebaran normal. Selanjutnya, ditinjau dari plot ACF dan PACF terlihat bahwa ada lag yang signifikan. Hal tersebut menunjukkan bahwa kemungkinan ada gejala autokorelasi pada sisaan. Selanjutnya, untuk memastikan kembali akan dilakukan uji asumsi secara formal:

## Uji Formal

#1) Sisaan Menyebar Normal  
ks.test(sisaan.model4,"pnorm")

##   
## Asymptotic one-sample Kolmogorov-Smirnov test  
##   
## data: sisaan.model4  
## D = 0.59164, p-value < 2.2e-16  
## alternative hypothesis: two-sided

#tak tolak H0 > sisaan menyebar normal  
shapiro.test(sisaan.model4)

##   
## Shapiro-Wilk normality test  
##   
## data: sisaan.model4  
## W = 0.95898, p-value = 0.0001432

nortest::ad.test(sisaan.model4)

##   
## Anderson-Darling normality test  
##   
## data: sisaan.model4  
## A = 1.7118, p-value = 0.0002093

Selain dengan eksplorasi, asumsi tersebut dapat diuji menggunakan uji formal. Pada tahapan ini uji formal yang digunakan untuk normalitas adalah uji Kolmogorov-Smirnov (KS), Shapiro-Wilk, dan Anderson-Darling. Hipotesis pada uji kenormalan adalah sebagai berikut.

: Sisaan menyebar normal

: Sisaan tidak menyebar normal

Berdasarkan uji KS, Shapiro-Wilk test, dan Anderson-Darling test tersebut, didapat *p-value* sebesar 0.000 yang lebih kecil dari taraf nyata 5% sehingga tolak dan menandakan bahwa sisaan menyebar normal.

#2) Sisaan saling bebas/tidak ada autokorelasi  
Box.test(sisaan.model4, type = "Ljung")

##   
## Box-Ljung test  
##   
## data: sisaan.model4  
## X-squared = 0.8609, df = 1, p-value = 0.3535

#tak tolak H0 > sisaan saling bebas

Selanjutnya akan dilakukan uji formal untuk kebebasan sisaan menggunakan uji Ljung-Box. Hipotesis yang digunakan adalah sebagai berikut.

: Sisaan saling bebas

: Sisaan tidak tidak saling bebas

Berdasarkan uji Ljung-Box tersebut, didapat *p-value* sebesar 0.3535 yang lebih besar dari taraf nyata 5% sehingga tak tolak dan menandakan bahwa sisaan saling bebas. Hal ini berbeda dengan eksplorasi.

#3) Sisaan homogen   
Box.test((sisaan.model4)^2, type = "Ljung")

##   
## Box-Ljung test  
##   
## data: (sisaan.model4)^2  
## X-squared = 4.1795, df = 1, p-value = 0.04092

#tak tolak H0 > sisaan homogen

Hipotesis yang digunakan untuk uji kehomogenan ragam adalah sebagai berikut.

: Ragam sisaan homogen

: Ragam sisaan tidak homogen

Berdasarkan uji Ljung-Box terhadap sisaan kuadrat tersebut, didapat *p-value* sebesar 0.04092 yang lebih kecil dari taraf nyata 5% sehingga tolak dan menandakan bahwa ragam sisaan tidak homogen.

#4) Nilai tengah sisaan sama dengan nol   
t.test(sisaan.model4, mu = 0, conf.level = 0.95)

##   
## One Sample t-test  
##   
## data: sisaan.model4  
## t = 0.6105, df = 155, p-value = 0.5424  
## alternative hypothesis: true mean is not equal to 0  
## 95 percent confidence interval:  
## -72.10581 136.61028  
## sample estimates:  
## mean of x   
## 32.25224

#tak tolak h0 > nilai tengah sisaan sama dengan 0

Terakhir, dengan uji-t, akan dicek apakah nilai tengah sisaan sama dengan nol. Hipotesis yang diujikan sebagai berikut.

: nilai tengah sisaan sama dengan 0

: nilai tengah sisaan tidak sama dengan 0

Berdasarkan uji-ttersebut, didapat *p-value* sebesar 0.5424 yang lebih besar dari taraf nyata 5% sehingga tak tolak dan menandakan bahwa nilai tengah sisaan sama dengan nol. Hal ini berbeda dengan eksplorasi.

## Overfitting

Pada model musiman, ordo yang dilakukan *overfit* adalah ordo musiman (*P*, *Q*).

#ARIMA(0,1,1)x(1,1,1)12  
tmodel1.ofP <- Arima(train.ts,order=c(2,1,3),seasonal=c(1,1,1))  
summary(tmodel1.ofP)

## Series: train.ts   
## ARIMA(2,1,3)(1,1,1)[12]   
##   
## Coefficients:  
## ar1 ar2 ma1 ma2 ma3 sar1 sma1  
## -1.1578 -0.9988 0.5099 0.2946 -0.6228 0.0628 -0.8987  
## s.e. 0.0055 0.0023 0.0609 0.0699 0.0597 0.1136 0.1481  
##   
## sigma^2 = 362663: log likelihood = -1127.42  
## AIC=2270.85 AICc=2271.92 BIC=2294.55  
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 38.67956 562.2882 401.3872 0.05614976 0.7815085 0.2478092  
## ACF1  
## Training set -0.09323106

lmtest::coeftest(tmodel1.ofP)

##   
## z test of coefficients:  
##   
## Estimate Std. Error z value Pr(>|z|)   
## ar1 -1.1578034 0.0054788 -211.3260 < 2.2e-16 \*\*\*  
## ar2 -0.9987509 0.0022852 -437.0491 < 2.2e-16 \*\*\*  
## ma1 0.5099240 0.0609297 8.3691 < 2.2e-16 \*\*\*  
## ma2 0.2945631 0.0698542 4.2168 2.478e-05 \*\*\*  
## ma3 -0.6227520 0.0596981 -10.4317 < 2.2e-16 \*\*\*  
## sar1 0.0627751 0.1136458 0.5524 0.5807   
## sma1 -0.8987371 0.1481070 -6.0682 1.294e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#ARIMA(2,1,3)x(2,1,0)12  
tmodel4.ofQ <- Arima(train.ts,order=c(2,1,3),seasonal=c(2,1,0))  
summary(tmodel4.ofQ)

## Series: train.ts   
## ARIMA(2,1,3)(2,1,0)[12]   
##   
## Coefficients:  
## ar1 ar2 ma1 ma2 ma3 sar1 sar2  
## -1.1577 -0.9977 0.4701 0.2808 -0.6450 -0.5824 -0.3635  
## s.e. 0.0064 0.0033 0.0636 0.0709 0.0619 0.0824 0.0838  
##   
## sigma^2 = 429026: log likelihood = -1133.93  
## AIC=2283.86 AICc=2284.94 BIC=2307.56  
##   
## Training set error measures:  
## ME RMSE MAE MPE MAPE MASE  
## Training set 29.49256 611.5742 437.7662 0.04411717 0.8547936 0.2702689  
## ACF1  
## Training set -0.1059484

lmtest::coeftest(tmodel4.ofQ)

##   
## z test of coefficients:  
##   
## Estimate Std. Error z value Pr(>|z|)   
## ar1 -1.1577479 0.0063664 -181.8515 < 2.2e-16 \*\*\*  
## ar2 -0.9976845 0.0033223 -300.3005 < 2.2e-16 \*\*\*  
## ma1 0.4700847 0.0636301 7.3878 1.493e-13 \*\*\*  
## ma2 0.2808371 0.0709007 3.9610 7.464e-05 \*\*\*  
## ma3 -0.6449593 0.0618988 -10.4196 < 2.2e-16 \*\*\*  
## sar1 -0.5823965 0.0823643 -7.0710 1.538e-12 \*\*\*  
## sar2 -0.3634593 0.0838434 -4.3350 1.458e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

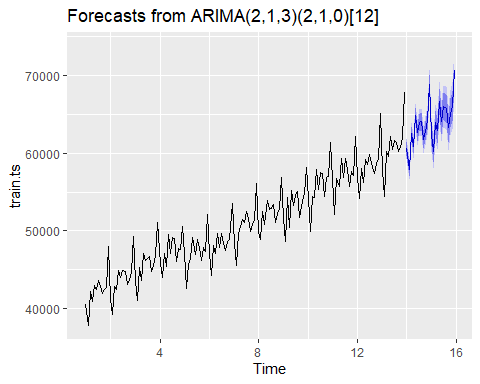
Model *overfitting* yang dicobakan menghasilkan nilai AIC dan signifikansi parameter yang lebih baik dari model awal. Oleh karena itu, model yang digunakan adalah model dengan semua parameter yang signfikan dan nilai AIC yang lebih kecil.

### Peramalan

ramalan\_sarima = forecast::forecast(tmodel4.ofQ, 24)  
ramalan\_sarima

## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  
## Jan 14 60621.42 59775.75 61467.09 59328.08 61914.76  
## Feb 14 57920.09 57036.19 58803.99 56568.28 59271.90  
## Mar 14 62549.36 61607.88 63490.84 61109.49 63989.23  
## Apr 14 60898.90 59911.92 61885.87 59389.45 62408.34  
## May 14 64910.53 63889.45 65931.60 63348.93 66472.12  
## Jun 14 62644.37 61571.41 63717.34 61003.41 64285.33  
## Jul 14 63916.61 62806.34 65026.87 62218.60 65614.62  
## Aug 14 64052.09 62909.76 65194.42 62305.05 65799.14  
## Sep 14 61763.57 60573.91 62953.22 59944.14 63582.99  
## Oct 14 62860.37 61639.26 64081.48 60992.84 64727.89  
## Nov 14 64160.25 62907.94 65412.55 62245.01 66075.49  
## Dec 14 68808.46 67512.96 70103.95 66827.16 70789.75  
## Jan 15 62464.30 61028.04 63900.57 60267.72 64660.88  
## Feb 15 60280.29 58790.57 61770.01 58001.96 62558.63  
## Mar 15 63917.73 62357.53 65477.94 61531.60 66303.87  
## Apr 15 62890.77 61283.90 64497.63 60433.28 65348.25  
## May 15 66645.85 64987.97 68303.72 64110.35 69181.35  
## Jun 15 64113.25 62392.82 65833.68 61482.07 66744.43  
## Jul 15 66048.86 64287.74 67809.99 63355.45 68742.28  
## Aug 15 65799.63 63988.81 67610.46 63030.22 68569.05  
## Sep 15 63270.92 61404.44 65137.40 60416.39 66125.45  
## Oct 15 65161.45 63258.33 67064.58 62250.87 68072.03  
## Nov 15 66063.79 64111.78 68015.81 63078.44 69049.14  
## Dec 15 70677.02 68675.53 72678.51 67616.01 73738.04

autoplot(ramalan\_sarima, col="blue")



accuracy(ramalan\_sarima,test.ts)

## ME RMSE MAE MPE MAPE MASE  
## Training set 29.49256 611.5742 437.7662 0.04411717 0.8547936 0.2702689  
## Test set -366.36993 961.7325 692.0721 -0.62175209 1.1223421 0.4272728  
## ACF1 Theil's U  
## Training set -0.1059484 NA  
## Test set -0.2566952 0.2586747

Hasil akurasi menunjukkan nilai MAPE di bawah 10% yang menyatakan bahwa prediksi sangat akurat.