

Pemulusan Harga Adjusted Closing IHSG Periode 1 Januari 2017 - 31 Maret 2022

Kelompok 14 - MPDW Genap 21/22

Anggota Kelompok

- G14190007 Dhiffa Fatihah Umami
- G14190026 Nenden Maulidia
- G14190043 Tony Alfian
- G14190047 Alvin Christian
- G14190071 Muhammad Hadyan Rizki

IHSG

Indeks Harga Saham Gabungan (IHSG) merupakan catatan-catatan atas perubahan maupun pergerakan harga saham sejak mulai pertama kali beredar sampai pada suatu saat tertentu. Ada beberapa faktor yang mempengaruhi IHSG, salah satunya adalah faktor makroekonomi seperti tingkat inflasi, nilai tukar rupiah, tingkat suku bunga dan lain-lain (Krisna dan Wirawati 2013).

Indeks Harga Saham Gabungan (IHSG) merupakan suatu nilai untuk mengukur kinerja seluruh saham. IHSG mencerminkan perkembangan pasar secara keseluruhan. Jika IHSG mengalami kenaikan dari hari kemarin maka beberapa saham yang berada pada bursa efek mengalami kenaikan. Oleh karena itu, peramalan harga akan sangat bermanfaat untuk para investor sehingga mereka dapat mengetahui prospek investasi saham di masa yang akan datang (Afittia 2018).

Dataset yang akan digunakan merupakan dataset IHSG dari tanggal 1 Januari 2017 hingga 31 Maret 2022 dengan jumlah 1.916 amatan. Karena terdapat banyak missing value (tanggal merah dan akhir pekan), maka perlu dilakukan imputasi.

Harga saham yang akan diamati adalah Adjusted Closing Price (harga closing yang sudah disesuaikan). Harga closing hanya menyatakan harga terakhir ketika pasar saham ditutup. Sedangkan harga closing yang telah disesuaikan mempertimbangkan faktor lain seperti dividen, stock splits, dan penawaran saham baru sehingga bisa menggambarkan nilai saham dengan lebih baik.

Praproses Data

Import Library dan Dataset

Mengimport libraries yang akan digunakan

```
library(imputeTS)
library(knitr)
library(quantmod)
library(forecast)
```

Import data dan melihat 10 data pertama.

```
start <- as.POSIXct("2017-01-01")
end <- as.POSIXct("2022-03-31")
getSymbols(Symbols = "^JKSE",src = "yahoo", from = start, to = end)

## [1] "^JKSE"

data <- as.data.frame(JKSE)
data$Date <- as.Date(rownames(data))
fulldates <- data.frame(Date=seq(as.Date("2017-01-01"), as.Date("2022-03-31"), by="days"))
ihsg <- merge(fulldates,data,by="Date",all.x=T)

kable(head(ihsg,10), caption="10 baris pertama Harga harian IHSG")
```

Table 1: 10 baris pertama Harga harian IHSG

Date	JKSE.Open	JKSE.High	JKSE.Low	JKSE.Close	JKSE.Volume	JKSE.Adjusted
2017-01-01	NA	NA	NA	NA	NA	NA
2017-01-02	NA	NA	NA	NA	NA	NA
2017-01-03	5290.388	5292.181	5246.852	5275.971	33217000	5275.971
2017-01-04	5270.016	5312.956	5249.234	5301.183	53111300	5301.183
2017-01-05	5302.633	5328.490	5302.633	5325.504	77219400	5325.504
2017-01-06	5321.745	5350.245	5318.385	5347.022	71615300	5347.022
2017-01-07	NA	NA	NA	NA	NA	NA
2017-01-08	NA	NA	NA	NA	NA	NA
2017-01-09	5350.880	5360.061	5307.585	5316.364	97010200	5316.364
2017-01-10	5330.015	5331.133	5292.063	5309.924	70792500	5309.924

Dari 10 data pertama, dapat dilihat bahwa terdapat beberapa missing values. Pasar Saham tidak dibuka pada hari sabtu-minggu dan tanggal merah, sehingga dapat dipastikan akan terdapat missing value tiap minggunya. Maka akan dilakukan interpolasi spline untuk mengatasi missing value tersebut.

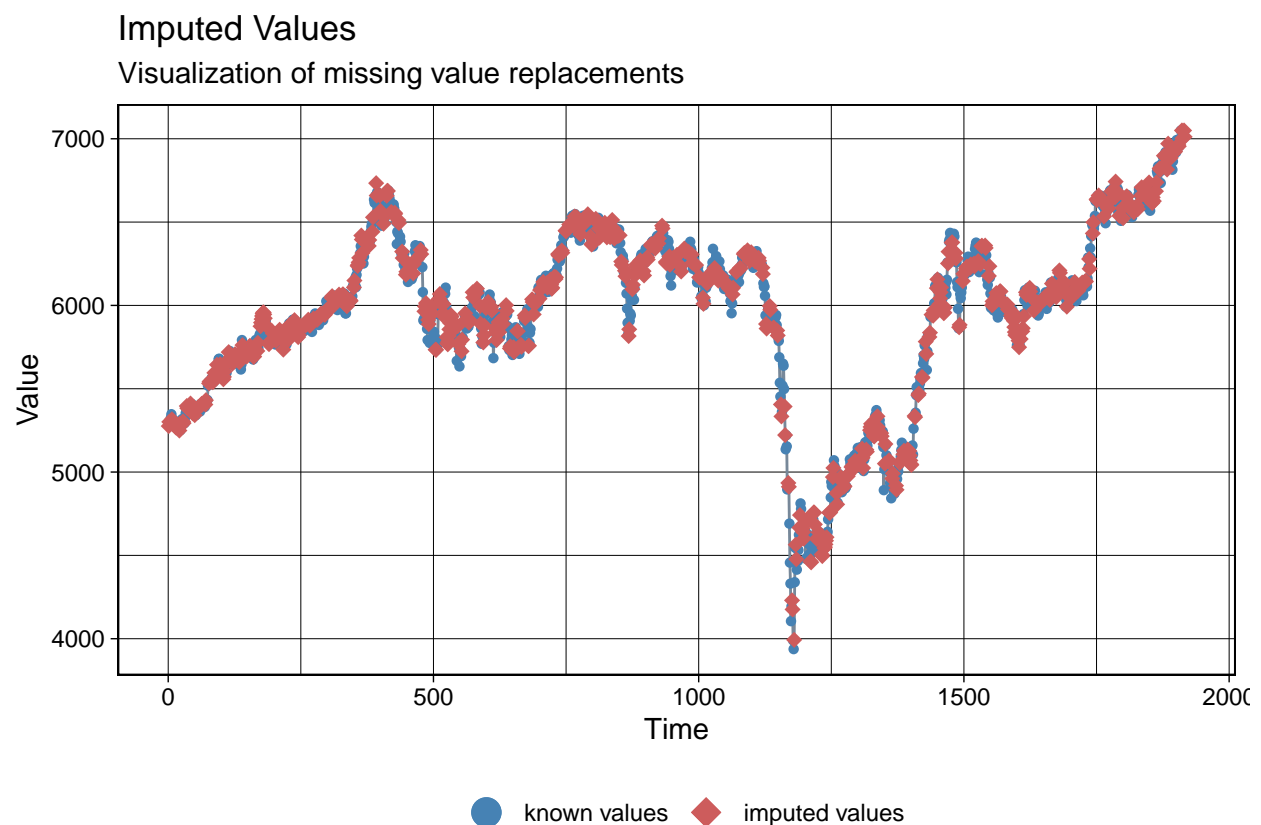
Imputasi Missing Values

Berdasarkan hasil uji modified time-series model, ditemukan bahwa bulan April dan Desember memiliki pengaruh positif dan signifikan terhadap return bulanan pasar, dan hal ini menunjukkan adanya pola seasonal bulanan yang terjadi selama periode Januari 2002 hingga Desember 2011 di Bursa Efek Indonesia (Perwitasari 2012).

Data yang digunakan adalah data univariate. Biasanya data univariate menggunakan linear interpolation untuk proses imputasi, namun data tersebut merupakan data musiman. linear interpolation hanya bisa digunakan untuk data yang bersifat non-musiman (Hyndman dan Athanasopoulos 2013). Maka dari itu, imputasi data yang cocok digunakan untuk data IHSG adalah spline interpolation, Rumus spline interpolation sebagai berikut.

$$f(x) = \begin{cases} f(x_0) + m_0(x - x_0) & x_0 \leq x \leq x_1 \\ f(x_1) + m_1(x - x_1) & x_1 \leq x \leq x_2 \\ \vdots & \\ f(x_{n-1}) + m_{n-1}(x - x_{n-1}) & x_{n-1} \leq x \leq x_n \end{cases}$$

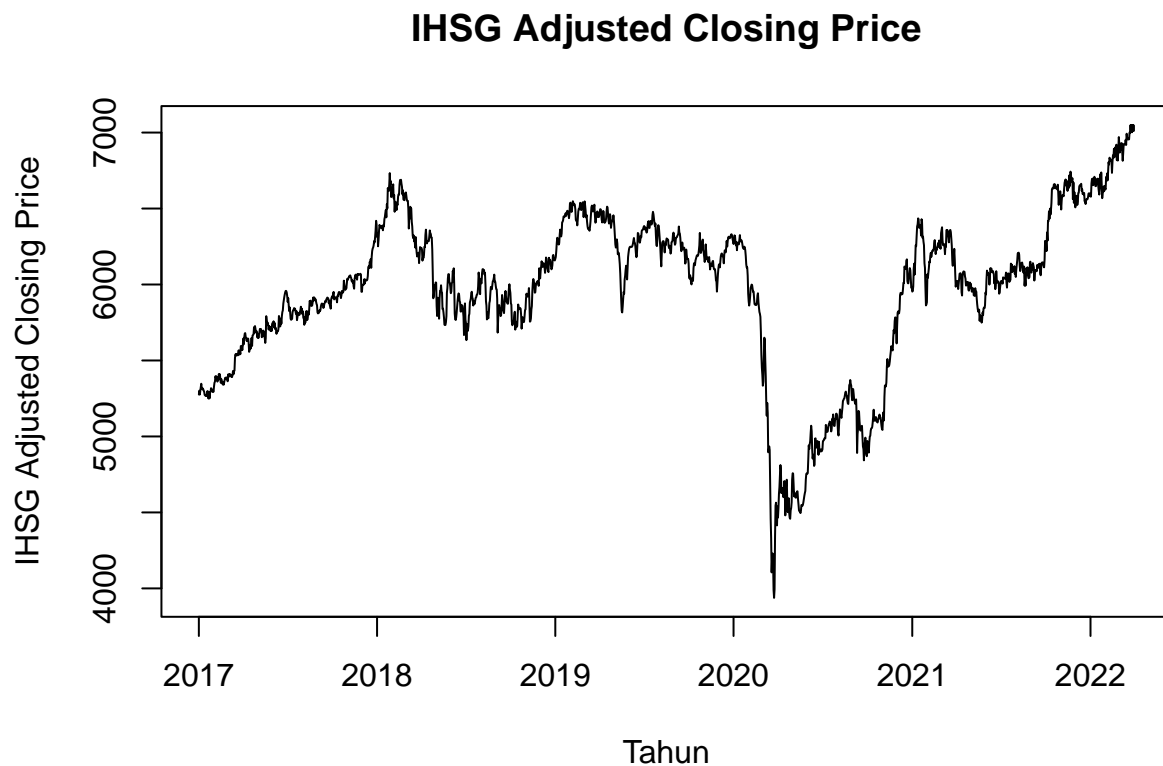
```
ihsg.imputed <- na_interpolation(ihsg, option="spline")
ggplot_na_imputations(ihsg$JKSE.Adjusted, ihsg.imputed$JKSE.Adjusted)
```



Dapat kita lihat bahwa imputasi tidak mengubah tren dari harga Adjusted Closing Price IHSX. Mari kita coba amati plot time series dari Adjusted Closing Price harian IHSX.

Eksplorasi Plot Deret Waktu

```
ihsg.ts <- ts(ihsg.imputed$JKSE.Adjusted)
plot(ihsg.imputed$JKSE.Adjusted~ihsg.imputed$Date,
     type = "l", xlab = "Tahun", main = "IHSG Adjusted Closing Price",
     ylab = "IHSG Adjusted Closing Price")
```



Train-test split data

Membagi dataset menjadi data training untuk melatih model dan data testing untuk validasi performa model.

```
ihsg.ts <- ts(ihsg.imputed$JKSE.Adjusted)
ihsg.seasonal <- ts(ihsg.imputed$JKSE.Adjusted, frequency=28)

train.prop <- floor(nrow(ihsg.imputed)*0.8)
test.prop <- nrow(ihsg.imputed)-train.prop

ts.train <- head(ihsg.ts, train.prop)
ts.test <- tail(ihsg.ts, test.prop)
seasonal.train <- head(ihsg.seasonal, train.prop)
seasonal.test <- tail(ihsg.seasonal, test.prop)
```

Moving Average

Menurut Makridakis, Wheelwright dan McGee (1995) Single Moving Average (SMA) atau rata-rata bergerak tunggal adalah nilai rata-rata yang tidak tertimbang dari n data sebelumnya atau dengan kata lain sebuah teknik yang merata-ratakan sebuah angka dari nilai aktual terbaru, diperbaharui sebagai nilai-nilai baru yang tersedia.

Double Moving Average (DMA) adalah metode dimana satu kelompok rata-rata bergerak dihitung, dan kemudian kelompok kedua dihitung rata-rata bergerak hasil pada kelompok pertama (Hanke et al. 2003).

Rumus perhitungan Moving Average adalah sebagai berikut:

Moving average orde ke-k

$$M_t = \hat{Y}_{t+1} = \frac{Y_t + Y_{t-1} + Y_{t-2} + \dots + Y_{t-k+1}}{k}$$

Moving average kedua

$$M'_t = \frac{M_t + M_{t-1} + M_{t-2} + \dots + M_{t-k+1}}{k}$$

dengan :

Y_t : data aktual pada waktu ke- t

k : banyak orde yang digunakan

p : banyak periode yang akan diramalkan

Berdasarkan penelitian yang dilakukan oleh Hadi et al. (2021) perbandingan parameter N untuk peramalan saham BBRI dengan metode Single Moving Average (SMA) menggunakan $n = 3, 5, 7, 9$ dan 11 . Hasil dari penelitian tersebut menunjukkan bahwa metode SMA dengan $n = 3$ memberikan nilai MAPE paling kecil diantara nilai parameter n yang lain. Oleh karena itu, kita akan menggunakan $n=3$ pada metode SMA

Berdasarkan penelitian yang dilakukan oleh Afifah et al. (2020) proses peramalan IHSG dengan metode Double Moving Average (DMA) menggunakan parameter $n=3$. Oleh karena itu, kita akan menggunakan $n=3$ pada metode DMA.

Menetapkan parameter $m = 3$ untuk moving average.

```
m = 3
```

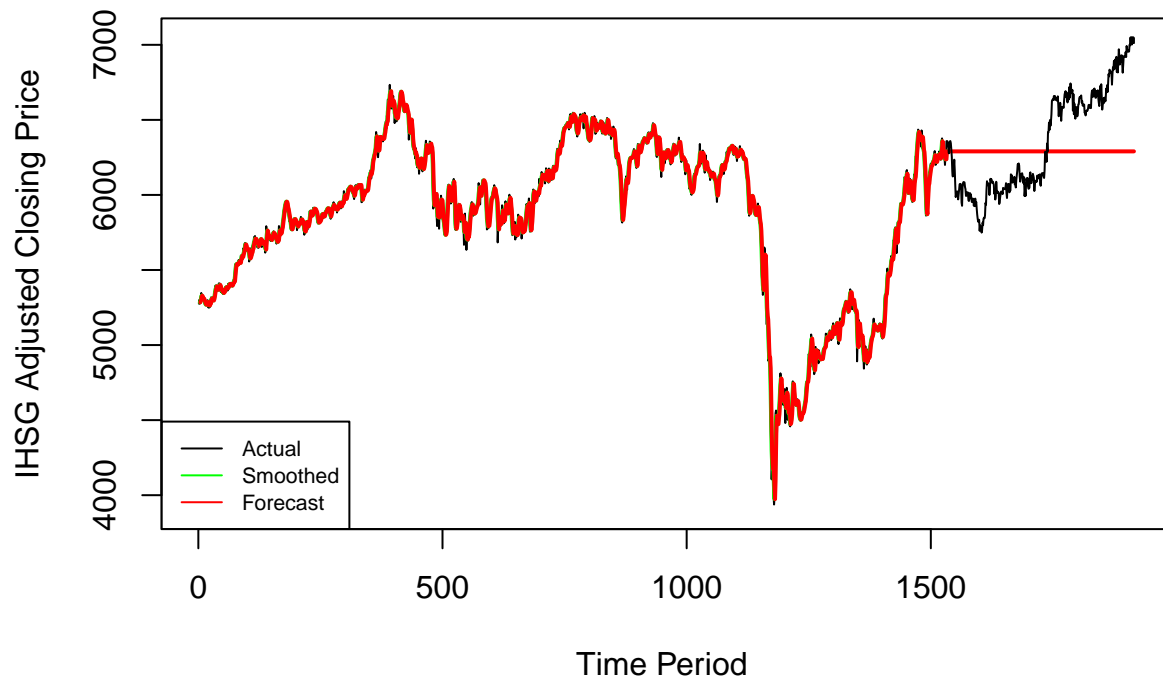
Single Moving Average

```
data.sma <- SMA(ts.train, n=m)
data.fc <- c(NA, data.sma)
data.gab <- data.frame(cbind(actual=c(ts.train, rep(NA, test.prop)), smoothing=c(data.sma, rep(NA, test.prop)),
                           forecast=c(data.fc, rep(data.fc[length(data.fc)], test.prop-1))))

ts.plot(data.gab[,1], xlab="Time Period ", ylab="IHSG Adjusted Closing Price",
        main="SMA of IHSG Adjusted Closing Price, m=3", ylim=c(3900,7050))
lines(data.gab[,2], col="green", lwd=2)
lines(data.gab[,3], col="red", lwd=2)
lines(ts.test)
```

```
legend("bottomleft",c("Actual","Smoothed","Forecast"), lty=1,
      col=c("black","green","red"), cex=0.7)
```

SMA of IHSG Adjusted Closing Price, m=3



Double Moving Average

```
dma <- SMA(data.sma, n = m)
At <- 2*data.sma - dma
Bt <- 2/(m-1)*(data.sma - dma)
data.dma <- At+Bt
data.fc2 <- c(NA, data.dma)

t = 1:test.prop+1
f = c()

for (i in t) {
  f[i] = At[length(At)] + Bt[length(Bt)]*(i)
}

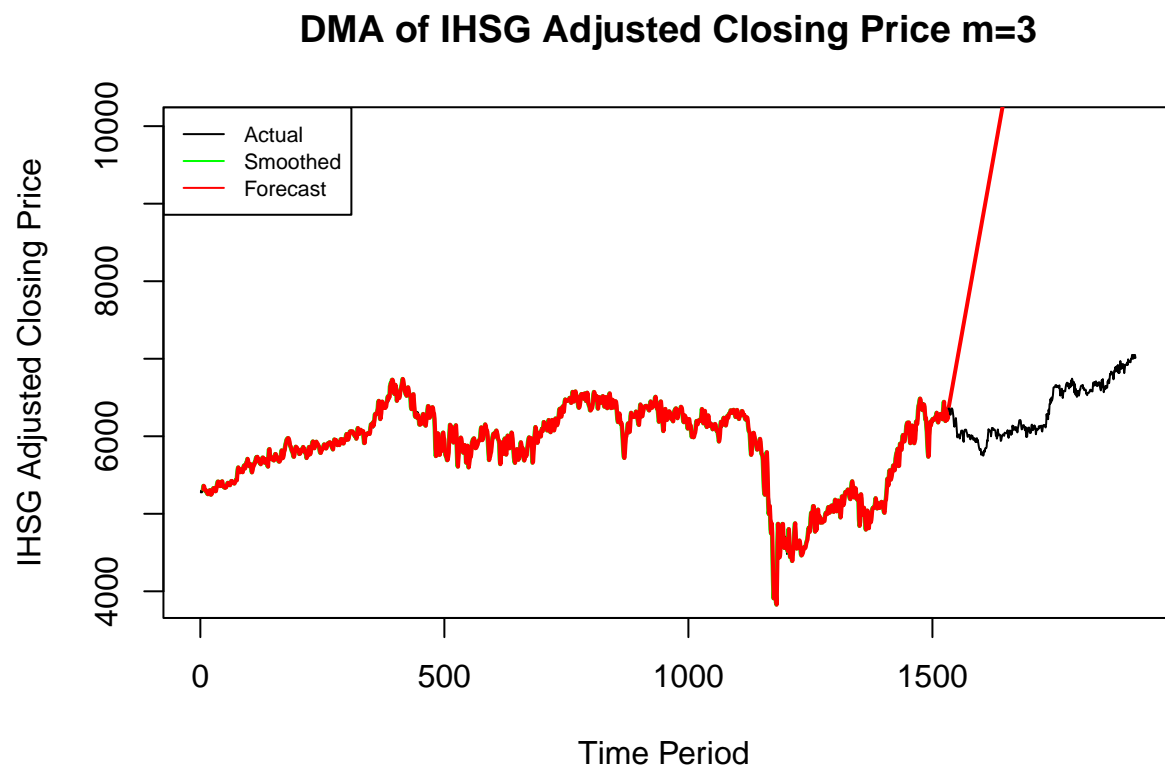
data.gab2 <- data.frame(cbind(actual = c(ts.train,rep(NA,test.prop+1)),
                             pemulusan1 = c(data.sma,rep(NA,test.prop+1)),
                             pemulusan2 = c(data.dma, rep(NA,test.prop+1)),
                             At = c(At, rep(NA,test.prop+1)),
                             Bt = c(Bt,rep(NA,test.prop+1)),
```

```

forecast = c(data.fc2, f[-1]))

ts.plot(data.gab2[,1], xlab="Time Period ", ylab="IHSG Adjusted Closing Price",
        main= "DMA of IHSG Adjusted Closing Price m=3", ylim=c(3900,10000))
lines(data.gab2[,3],col="green",lwd=2)
lines(data.gab2[,6],col="red",lwd=2)
lines(ts.test)
legend("topleft",c("Actual","Smoothed","Forecast"), lty=1,
      col=c("black","green","red"), cex=0.7)

```



Exponential Smoothing

Single Exponential Smoothing (SES) adalah metode peramalan yang digunakan untuk meramalkan masa yang akan datang dengan melakukan proses pemulusan yang menghasilkan data ramalan yang lebih kecil nilai kesalahannya. Dalam pemulusan eksponensial terdapat satu atau lebih parameter pemulusan yang ditentukan secara eksplisit dan hasil pilihan menentukan bobot yang dikenakan pada nilai observasi (Makridakis et al. 1999).

Double Exponential Smoothing (DES) adalah metode yang digunakan ketika data menunjukkan adanya trend. Exponential Smoothing dengan adanya trend seperti pemulusan sederhana kecuali bahwa dua komponen harus diperbarui setiap periode, level, dan trendnya. Dimana level adalah estimasi yang dimuluskan dari nilai data pada akhir masing-masing periode (Agus 2009).

Single Exponential Smoothing

Nilai SES dihitung dengan:

$$\hat{Y}_{t+1} = \alpha Y_t + (1-\alpha) \hat{Y}_t \quad \dots\dots\dots (1)$$

Keterangan:

\hat{Y}_{t+1} = nilai peramalan untuk periode berikutnya

Y_t = permintaan untuk periode t

\hat{Y}_t = nilai peramalan untuk periode t

α = faktor bobot penghalusan ($0 < \alpha < 1$)

Menurut Indah (2021), metode peramalan IHSG dengan menggunakan metode Single Exponential Smoothing (SES) diperoleh tingkat kesalahan yang paling kecil dibanding metode lainnya yakni ketika $\alpha = 0.5$. Sedangkan menurut Siregar (2016), α optimum untuk meramal IHSG menggunakan metode SES yaitu ketika $\alpha = 0.9$.

```
ses.1 <- HoltWinters(ts.train, gamma = F, beta = F, alpha = 0.5)
ses.2 <- HoltWinters(ts.train, gamma = F, beta = F, alpha = 0.9)
ses.opt <- HoltWinters(ts.train, gamma = F, beta = F)

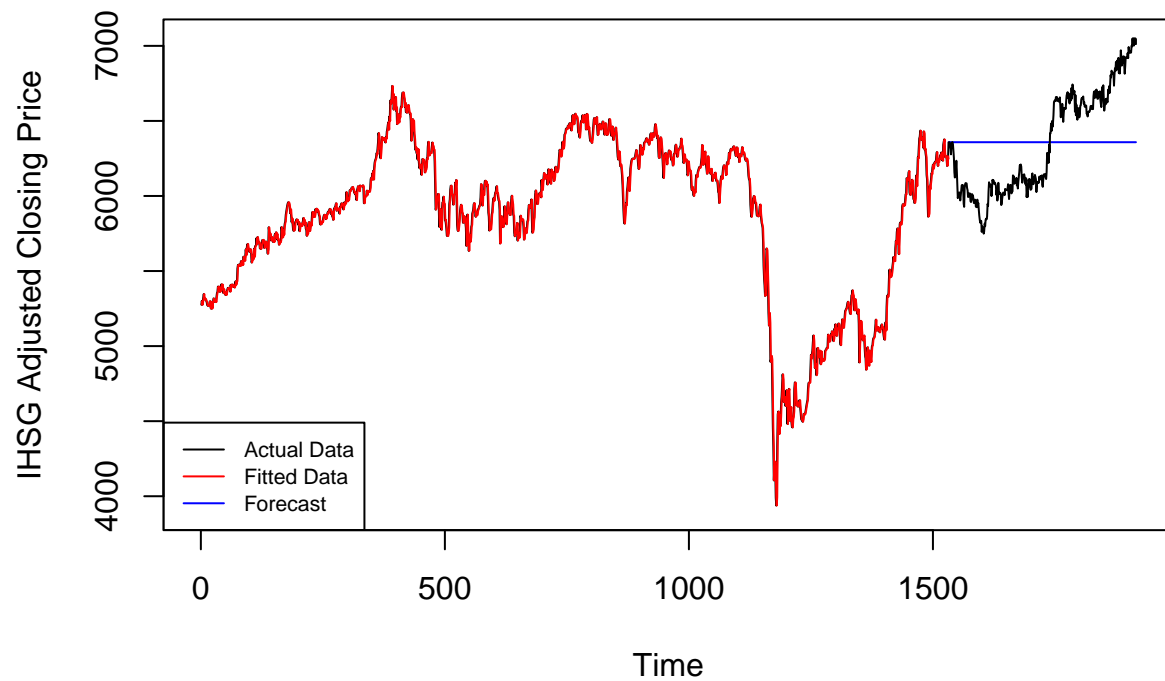
ses.opt #optimum parameter for ses a = 0.9999414

## Holt-Winters exponential smoothing without trend and without seasonal component.
##
## Call:
## HoltWinters(x = ts.train, beta = F, gamma = F)
##
## Smoothing parameters:
##   alpha: 0.9999414
##   beta  : FALSE
##   gamma : FALSE
##
## Coefficients:
##           [,1]
## a 6358.203

fc.ses1 <- predict(ses.1, n.ahead = test.prop)
fc.ses2 <- predict(ses.2, n.ahead = test.prop)
fc.sesopt <- predict(ses.opt, n.ahead = test.prop)

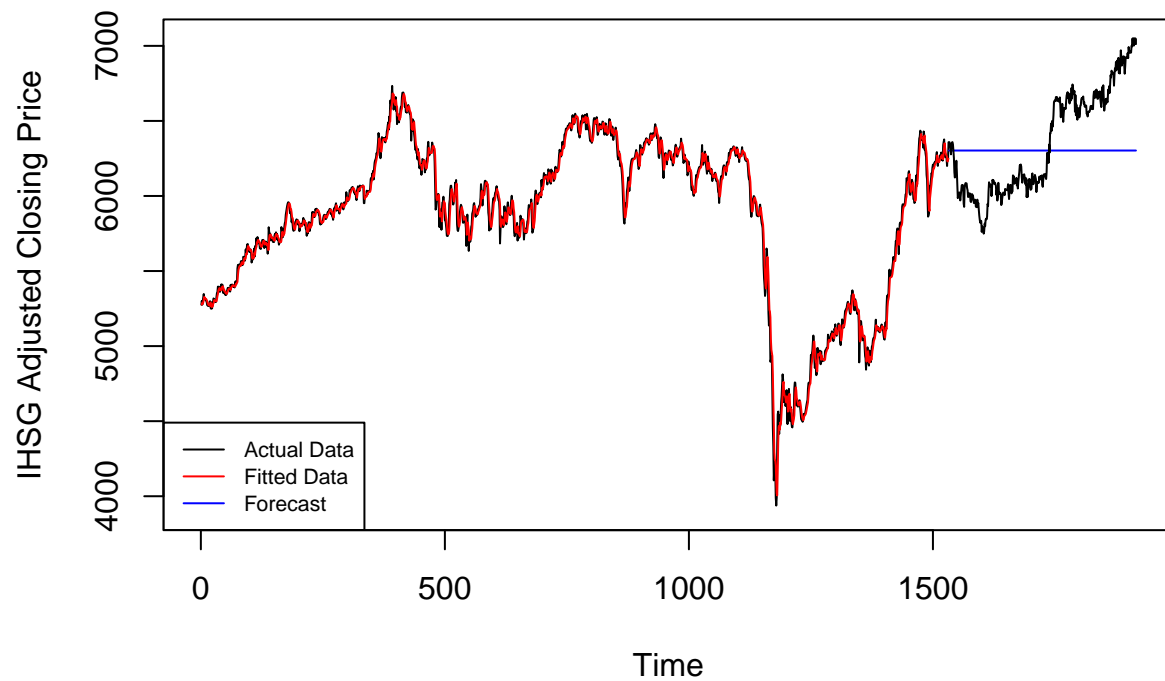
plot(ts.train,main="SES with Optimal parameter alpha=0.9999414", type="l", col="black", pch=12,
      ylab="IHSG Adjusted Closing Price", xlim=c(0,1916), ylim=c(3900,7050))
lines(ses.opt$fitted[,2],type="l",col="red")
lines(fc.sesopt,type="l",col="blue")
lines(ts.test,type="l")
legend("bottomleft",c("Actual Data","Fitted Data","Forecast"),
      col=c("black","red","blue"),lty=1, cex=0.7)
```


SES with Optimal parameter $\alpha=0.9999414$



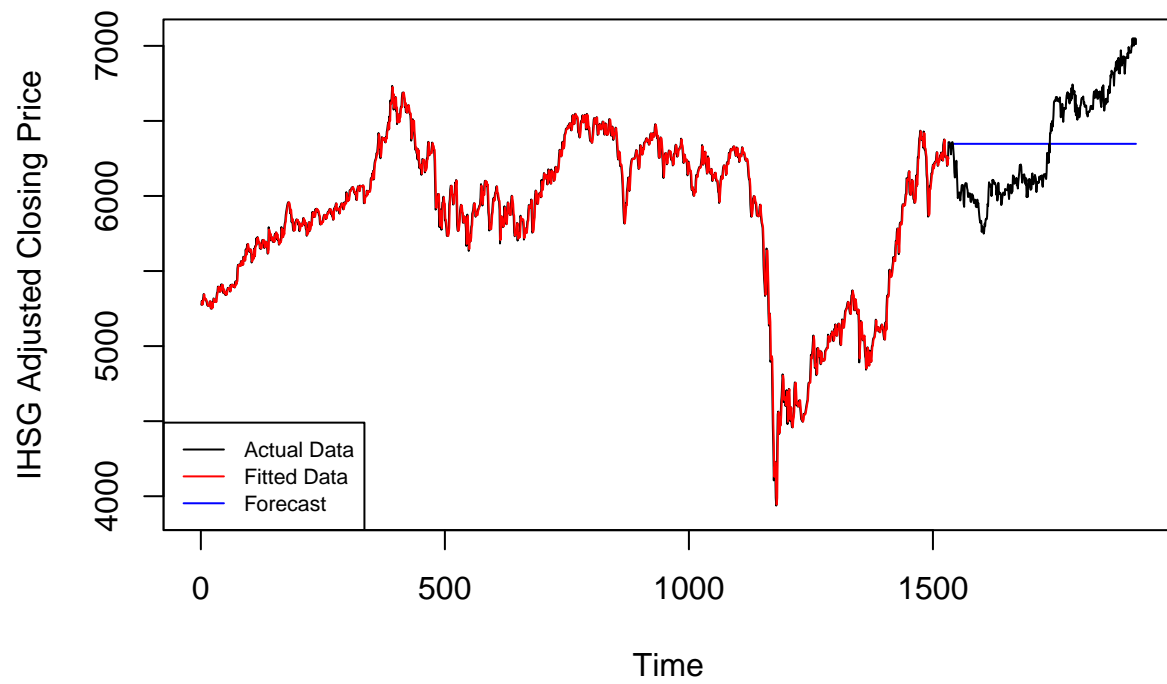
```
plot(ts.train,main="SES with alpha=0.5",type="l",col="black",pch=12,  
     ylab="IHSJ Adjusted Closing Price", xlim=c(0,1916), ylim=c(3900,7050))  
lines(ses.1$fitted[,2],type="l",col="red")  
lines(fc.ses1,type="l",col="blue")  
lines(ts.test,type="l")  
legend("bottomleft",c("Actual Data","Fitted Data","Forecast"),  
      col=c("black","red","blue"),lty=1, cex=0.7)
```

SES with alpha=0.5



```
plot(ts.train,main="SES with alpha=0.9",type="l",col="black",pch=12,
     ylab="IHSJ Adjusted Closing Price",
     xlim=c(0,1916),ylim=c(3900,7050))
lines(ses.2$fitted[,2],type="l",col="red")
lines(fc.ses2,type="l",col="blue")
lines(ts.test,type="l")
legend("bottomleft",c("Actual Data","Fitted Data","Forecast"),
     col=c("black","red","blue"),lty=1, cex=0.7)
```

SES with alpha=0.9



Double Exponential Smoothing

DES dihitung dengan:

$$S'_t = \alpha X_t + (1 - \alpha)(S'_{t-1} + t_{t-1}) \quad (1)$$

$$t_t = \beta(S'_t - S'_{t-1}) + (1 - \beta)t_{t-1} \quad (2)$$

$$F_{t+m} = S'_t + t_t m \quad (3)$$

Inialisasi :

$$S'_1 = X_1 \quad (4)$$

$$t_1 = \frac{(X_2 - X_1) + (X_4 - X_3)}{2} \quad (5)$$

Dimana:

X_t	=	Data <i>demand</i> pada periode t
S'_t	=	nilai <i>single exponential smoothing</i>
t_t	=	Nilai <i>trend</i> pada periode ke - t
α, β	=	parameter pemulusan antara 0-1
F_{t+m}	=	ramalan m periode yang akan diramalkan
m	=	jumlah periode ke muka yang akan diramalkan

Pada metode Double Exponential Smoothing (DES) diperoleh nilai $\alpha = 1$ dan $\beta = 0.02380527$ sebagai nilai konstanta smoothing untuk estimasi level dan trend yang optimal sehingga dapat menghasilkan plot data model yang berhimpitan dengan data observasi (Slamet, Subanti, dan Susanto 2020). Menurut Purnamasari dan Sudirman (2019) model peramalan IHSG dengan menggunakan metode DES untuk $\alpha = 0.86$ dan $\beta = 0.01$ menunjukkan adanya kecenderungan trend naik, dengan nilai MAPE di bawah 3%.

```
des.1 <- HoltWinters(ts.train, alpha = 1, beta=0.024, gamma=F)
des.2 <- HoltWinters(ts.train, alpha = 0.86, beta=0.01, gamma=F)
des.opt <- HoltWinters(ts.train, gamma=F)

des.opt #optimum parameter for des a=1, b=0.01499698
```

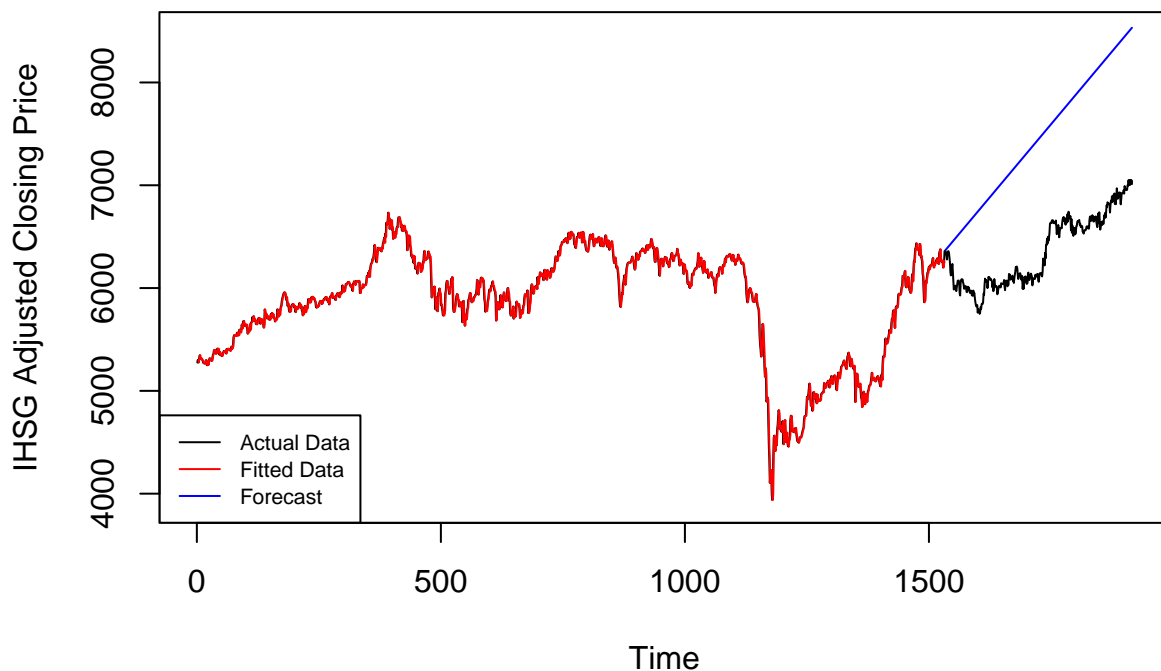
```
## Holt-Winters exponential smoothing with trend and without seasonal component.
##
## Call:
```

```
## HoltWinters(x = ts.train, gamma = F)
##
## Smoothing parameters:
##   alpha: 1
##   beta : 0.01499698
##   gamma: FALSE
##
## Coefficients:
##           [,1]
## a 6358.208984
## b    5.661422

fc.des1 <- predict(des.1, n.ahead = test.prop)
fc.des2 <- predict(des.2, n.ahead = test.prop)
fc.desopt <- predict(des.opt, n.ahead = test.prop)

plot(ts.train,main="DES with Optimal parameter alpha=1 beta=0.01499698",
     type="l",col="black",pch=12, ylab="IHSG Adjusted Closing Price",
     xlim=c(0,1916),ylim=c(3900,8500))
lines(des.opt$fitted[,2],type="l",col="red")
lines(fc.desopt,type="l",col="blue")
lines(ts.test,type="l")
legend("bottomleft",c("Actual Data","Fitted Data","Forecast"),
     col=c("black","red","blue"),lty=1, cex=0.7)
```

DES with Optimal parameter alpha=1 beta=0.01499698

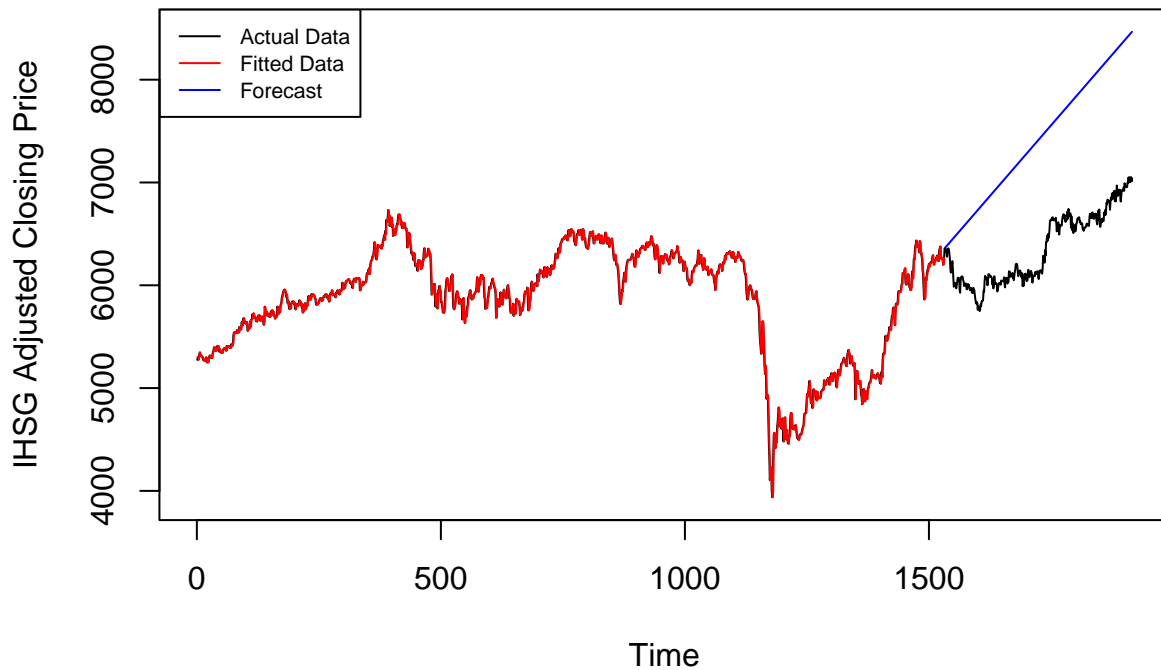


```

plot(ts.train,main="DES with alpha=1 beta=0.024",
     type="l",col="black",pch=12, ylab="IHSG Adjusted Closing Price",
     xlim=c(0,1916),ylim=c(3900,8500))
lines(des.1$fitted[,2],type="l",col="red")
lines(fc.des1,type="l",col="blue")
lines(ts.test,type="l")
legend("topleft",c("Actual Data","Fitted Data","Forecast"),
      col=c("black","red","blue"),lty=1, cex=0.7)

```

DES with alpha=1 beta=0.024

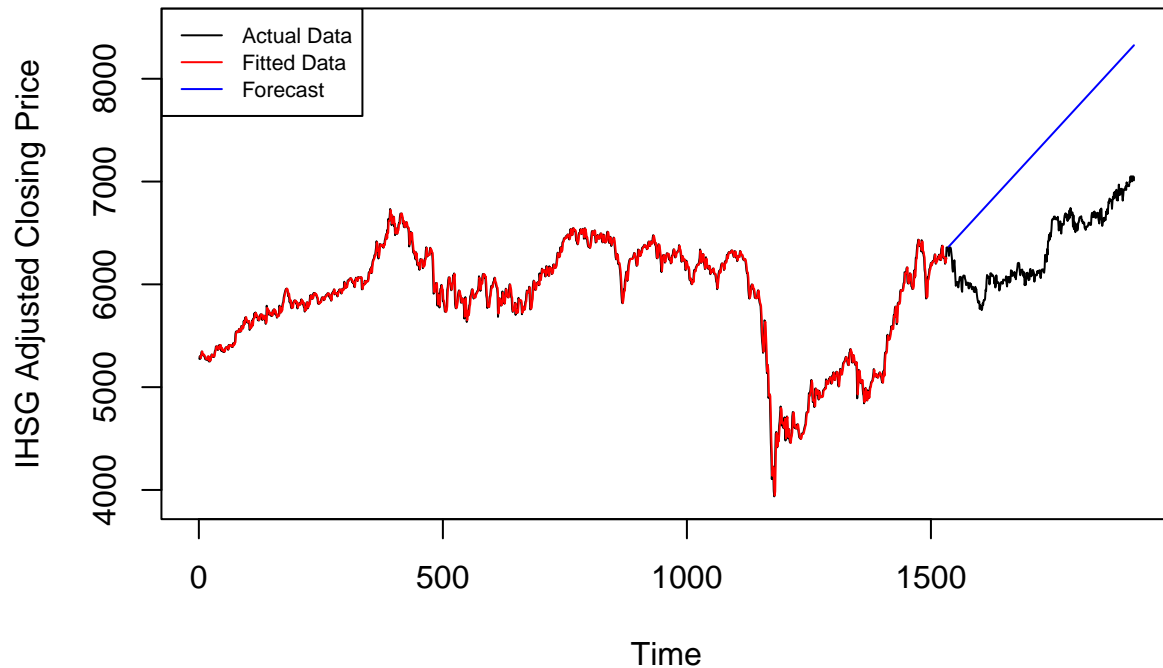


```

plot(ts.train,main="DES with alpha=0.86 beta=0.01",
     type="l",col="black",pch=12, ylab="IHSG Adjusted Closing Price",
     xlim=c(0,1916),ylim=c(3900,8500))
lines(des.2$fitted[,2],type="l",col="red")
lines(fc.des2,type="l",col="blue")
lines(ts.test,type="l")
legend("topleft",c("Actual Data","Fitted Data","Forecast"),
      col=c("black","red","blue"),lty=1, cex=0.7)

```

DES with $\alpha=0.86$ $\beta=0.01$



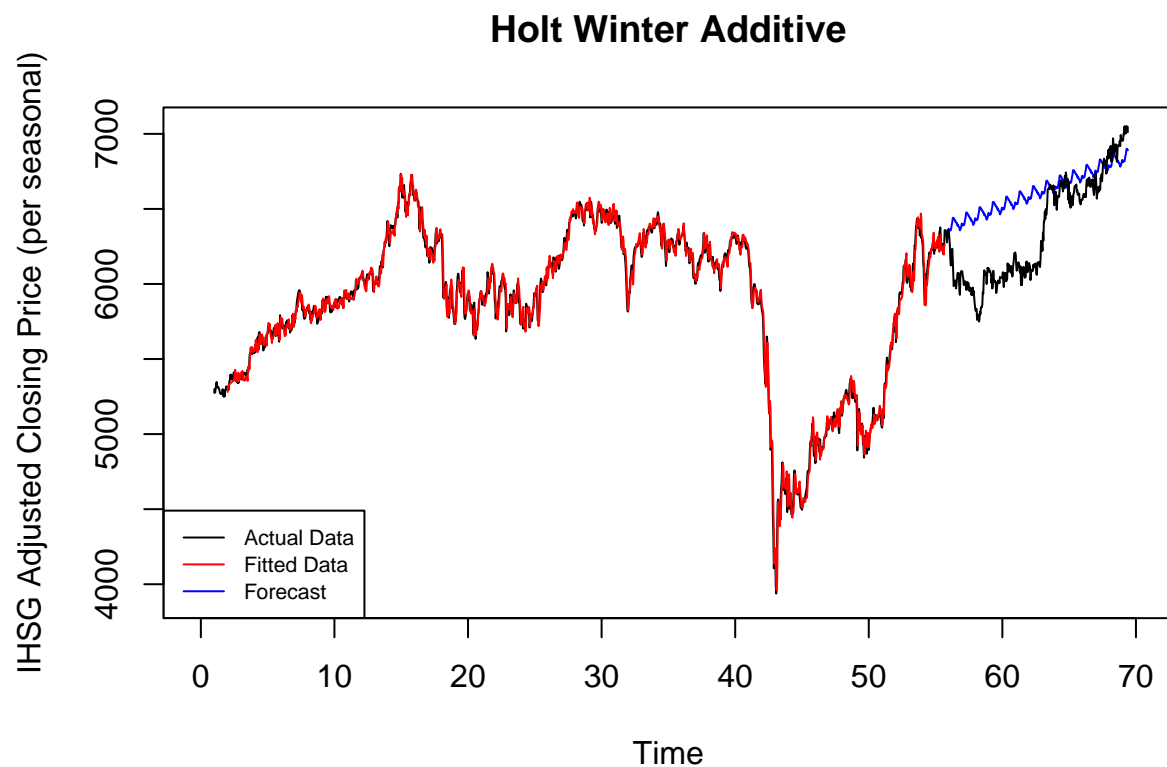
Holt-Winters Seasonal

Holt-Winters atau Triple Exponential Smoothing adalah metode yang digunakan untuk data yang menunjukkan trend dan pengaruh musiman (Kalekar 2004). Mengatasi pengaruh musiman diperlukan parameter ketiga (γ). Berdasarkan tipe pengaruh musiman, metode Holt-Winters dapat dikelompokkan menjadi 2, yaitu: - Multiplicative seasonal model - Additive seasonal model

Holt-Winters Additive

```
HWA <- HoltWinters(seasonal.train, seasonal = "additive")
fc.HWA <- forecast(HWA, h=test.prop)
predictHWA <- predict(HWA, n.ahead=test.prop)

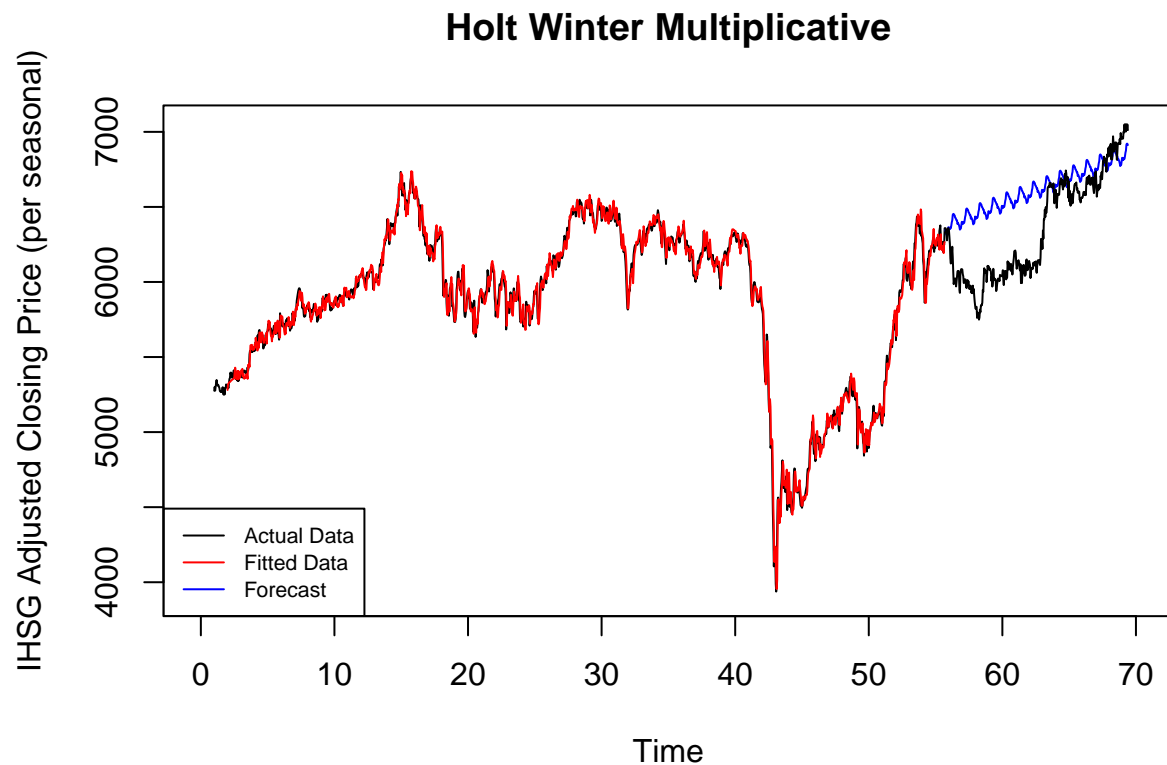
plot(seasonal.train, main="Holt Winter Additive", type="l", col="black", pch=12,
     ylim=c(3900, 7050), xlim=c(0, 70),
     ylab="IHSJ Adjusted Closing Price (per seasonal)")
lines(HWA$fitted[, 2], type="l", col="red")
lines(predictHWA, type="l", col="blue")
lines(seasonal.test, type="l")
legend("bottomleft", c("Actual Data", "Fitted Data", "Forecast"),
     col=c("black", "red", "blue"), lty=1, cex=0.7)
```



Holt-Winters Multiplicative

```
HWM <- HoltWinters(seasonal.train, seasonal = "multiplicative")
fc.HWM <- forecast(HWM, h=test.prop)
predictHWM <- predict(HWM, n.ahead=test.prop)

plot(seasonal.train,main="Holt Winter Multiplicative",type="l",col="black",pch=12,
      ylim=c(3900,7050),xlim=c(0,70),
      ylab="IHSG Adjusted Closing Price (per seasonal)")
lines(HWM$fitted[,2],type="l",col="red")
lines(predictHWM,type="l",col="blue")
lines(seasonal.test,type="l")
legend("bottomleft",c("Actual Data","Fitted Data","Forecast"),
      col=c("black","red","blue"),lty=1,cex=0.7)
```

Ukuran kesalahan untuk mengukur akurasi peramalan

MAPE (Mean Absolute Percentage Error) untuk ukuran kesalahan relatif yang menyatakan persentase kesalahan hasil peramalan terhadap permintaan aktual selama periode tertentu yang akan memberikan informasi persentase kesalahan terlalu tinggi atau rendah (Gaspersz 1998). MSE (Mean Square Error) dihitung dengan menjumlahkan kuadrat semua kesalahan peramalan pada setiap periode dan membaginya dengan jumlah periode peramalan (Vincent 2008).

RMSE dihitung dengan formula:

$$RMSE = \sqrt{\frac{\sum e_i^2}{n}}$$

MAPE dihitung dengan formula:

$$MAPE = \sum_{i=1}^n \frac{|PE_i|}{n}$$

Membandingkan nilai RMSE dari data training

```
error.sma <- ts.train - data.fc[1:length(ts.train)]
RMSE.sma <- sqrt(mean(error.sma[test.prop+1:length(ts.test)]^2))
error.dma = ts.train-data.fc2[1:length(ts.train)]
RMSE.dma = sqrt(mean(error.dma[(m*2):length(ts.train)]^2))

RMSE.ses1 <- sqrt(ses.1$SSE/length(ts.train))
RMSE.ses2 <- sqrt(ses.2$SSE/length(ts.train))
RMSE.sesopt <- sqrt(ses.opt$SSE/length(ts.train))
RMSE.des1 <- sqrt(des.1$SSE/length(ts.train))
RMSE.des2 <- sqrt(des.2$SSE/length(ts.train))
RMSE.desopt <- sqrt(des.opt$SSE/length(ts.train))

RMSE.HWA <- sqrt(HWA$SSE/length(seasonal.train))
RMSE.HWM <- sqrt(HWM$SSE/length(seasonal.train))

err <- data.frame(metode=c("SMA","DMA","SES 1","SES 2","SES opt",
                          "DES 1", "DES 2", "DES opt",
                          "HW Additive", "HW Multiplicative"),
                  RMSE=c(RMSE.sma, RMSE.dma, RMSE.ses1, RMSE.ses2, RMSE.sesopt,
                         RMSE.des1, RMSE.des2, RMSE.desopt, RMSE.HWA, RMSE.HWM))

kable(err)
```

metode	RMSE
SMA	64.89666
DMA	66.31930
SES 1	57.28156
SES 2	47.65417
SES opt	46.61853
DES 1	46.87520
DES 2	48.47003
DES opt	46.84327
HW Additive	48.82866
HW Multiplicative	49.24107

Membandingkan nilai RMSE dari data testing

```
test.RMSE.SMA <- sqrt(mean((tail(data.gab$forecast, test.prop)-ts.test)^2))
test.RMSE.DMA <- sqrt(mean((tail(data.gab2$forecast, test.prop)-ts.test)^2))
```

```

test.RMSE.SES1 <- sqrt(mean((fc.ses1-ts.test)^2))
test.RMSE.SES2 <- sqrt(mean((fc.ses2-ts.test)^2))
test.RMSE.SESopt <- sqrt(mean((fc.sesopt-ts.test)^2))
test.RMSE.DES1 <- sqrt(mean((fc.des1-ts.test)^2))
test.RMSE.DES2 <- sqrt(mean((fc.des2-ts.test)^2))
test.RMSE.DESopt <- sqrt(mean((fc.desopt-ts.test)^2))

test.RMSE.HWA <- sqrt(mean((fc.HWA$mean[1:test.prop]-ts.test)^2))
test.RMSE.HWM <- sqrt(mean((fc.HWM$mean[1:test.prop]-ts.test)^2))

test.err <- data.frame(metode=c("SMA","DMA","SES 1","SES 2","SES opt",
                                "DES 1", "DES 2", "DES opt",
                                "HW Additive","HW Multiplicative"),
                      RMSE=c(test.RMSE.SMA, test.RMSE.DMA,
                              test.RMSE.SES1, test.RMSE.SES2, test.RMSE.SESopt,
                              test.RMSE.DES1, test.RMSE.DES2, test.RMSE.DESopt,
                              test.RMSE.HWA, test.RMSE.HWM))

kable(test.err)

```

metode	RMSE
SMA	354.2710
DMA	7660.7545
SES 1	352.0963
SES 2	347.9871
SES opt	347.8415
DES 1	1109.8895
DES 2	1026.5869
DES opt	1146.8849
HW Additive	342.6330
HW Multiplicative	345.4335

Membandingkan nilai MAPE dari data testing

```

MAPE.sma <- mean(abs((ts.test-tail(data.gab$forecast, test.prop))/ts.test))*100
MAPE.dma <- mean(abs((ts.test-tail(data.gab2$forecast, test.prop))/ts.test))*100

MAPE.ses1 <- mean(abs((fc.ses1 - ts.test)/ts.test)) * 100
MAPE.ses2 <- mean(abs((fc.ses2 - ts.test)/ts.test)) * 100
MAPE.sesopt <- mean(abs((fc.sesopt - ts.test)/ts.test)) * 100

MAPE.des1 <- mean(abs((fc.des1 - ts.test)/ts.test)) * 100
MAPE.des2 <- mean(abs((fc.des2 - ts.test)/ts.test)) * 100
MAPE.desopt <- mean(abs((fc.desopt - ts.test)/ts.test)) * 100

MAPE.HWA <- mean(abs((fc.HWA$mean - seasonal.test)/seasonal.test)) * 100
MAPE.HWM <- mean(abs((fc.HWM$mean - seasonal.test)/seasonal.test)) * 100

MAPE <- data.frame(metode=c("SMA","DMA","SES 1","SES 2","SES opt",
                            "DES 1", "DES 2", "DES opt",
                            "HW Additive","HW Multiplicative"),
                  MAPE=c(MAPE.sma, MAPE.dma,
                          MAPE.ses1, MAPE.ses2, MAPE.sesopt,
                          MAPE.des1, MAPE.des2, MAPE.desopt,
                          MAPE.HWA, MAPE.HWM))

```

```

                                "HW Additive", "HW Multiplicative"),
MAPE=c(MAPE.sma, MAPE.dma,
        MAPE.ses1, MAPE.ses2, MAPE.sesopt,
        MAPE.des1, MAPE.des2, MAPE.desopt,
        MAPE.HWA, MAPE.HWM))
kable(MAPE)

```

metode	MAPE
SMA	4.928133
DMA	104.119740
SES 1	4.943376
SES 2	5.005881
SES opt	5.024747
DES 1	16.516330
DES 2	15.323370
DES opt	17.028077
HW Additive	4.547781
HW Multiplicative	4.589989

Kesimpulan

Berdasarkan nilai RMSE dan MAPE dari data testing, terlihat bahwa metode pemulusan Holt-Winters Seasonal (terutama Additive) merupakan metode pemulusan yang paling baik dalam melakukan peramalan Adjusted Closing Price IHSG periode 1 Januari 2017 hingga 31 Maret 2022.

Daftar Pustaka

- Afittia W. 2018. Penerapan Metode Eksponensial Smoothing untuk Meramalkan Indeks Harga Saham Gabungan pada Tahun 2018-2020. Medan (ID) : Universitas Sumatera Utara.
- Agus, Akbar S. 2009. Penerapan Metode Single Moving Average dan Exponential Smoothing dalam Peramalan Permintaan Produk Meubel Jenis Coffe Table pada Java Furniture Klaten. Jurnal Ilmiah (Online), tersedia : <http://core.ac.uk/download/file/478/12349901.pdf> , diunduh : 23 Februari 2022.
- Febriani A.S, Subanti S, Slamet I. 2020. Perbandingan Exponential Smothing Holt's Method dan Double Moving Averages terhadap Peramalan IHSG. Prosiding Pendidikan Matematika dan Matematika. 2(1):1-9.
- Vincent G. 2008. Total Quality Control. Jakarta: PT. Gramedia Pustaka Utama.
- Hadi R, Budiman I, Abadi F, Farmandi A, Muliadi. 2021. OPTIMIZATION OF N VALUE ON SINGLE MOVING AVERAGE (SMA) WITH PARTICLE SWARM OPTIMIZATION (PSO) CASE STUDY OF BRI STOCK. Journal of Data Science and Software Engineering. 2(3): 156-170.
- Hanke, J.E., Wichern, D.W., Reitsch, A.G. 2003. Peramalan Bisnis. Edisi ke-7. penerjemah; Anantanur, Devy. Jakarta: PT Prenhallindo. Terjemahan dari: Business Forecasting 7th Edition.
- Hyndman, R. J., & Athanasopoulos, G. (2013). Forecasting: principles and practice. Otexts.
- Indah N. 2021. Aplikasi metode moving average dan smoothing exponential pada proyeksi harga saham (studi kasus pada harga saham kaef). Jurnal MABIS. 12(2) : 186-196.
- Kalekar SP. 2004. Time series Forecasting using Holt-Winters Exponential Smoothing. Mumbai (IN): Kanwal Rekhi School of Information Technology.

- Krisna AAGD, Wirawati NGP. 2013. Pengaruh inflasi, nilai tukar rupiah, suku bunga sbi pada indeks harga saham gabungan di bei. *E-Jurnal Akuntansi*. 3(2) : 421-435.
- Makridakis, Spyros, Wheelwright SC, McGee VE, McGee. 1999. *Metode dan Aplikasi Peramalan* (2th ed) Volume 1. Jakarta (ID) : Erlangga.
- Perwitasari AD. 2012. *Investigasi Pola Return Seasonal Pada Bursa Efek Indonesia (Studi Pada IHSG, Indeks Saham Sektoral dan Indeks Saham Unggulan (LQ45))*. [skripsi]. Malang (ID) : Universitas Brawijaya.
- Purnamasari I, Sudirman. 2019. Peramalan indeks harga saham gabungan (IHSG) menggunakan metode pemulusan eksponensial ganda dari holt. *Prosiding Seminar Nasional Matematika, Statistika, dan Aplikasinya*; 2019 Okt 5; Samarinda. Samarinda (ID) : Universitas Mulawarman. hlm 150-157.
- Siregar RA. 2016. *Pembangunan aplikasi berbasis WEB untuk peramalan harga saham dengan metode Moving Average, Exponential Smoothing, dan Artificial Neural Network*. [skripsi]. Surabaya (ID) : Institut Teknologi Sepuluh November.
- Slamet I, Subanti S, Susanto AF. 2020. Perbandingan Exponential Smoothing Holt's Method dan Double Moving Averages terhadap peramalan IHSG. *Prosiding Pendidikan Matematika dan Matematika Volume 2*. Surakarta (ID) : Universitas Sebelas Maret. hlm 1-9.
- Subagyo. 2008. *Forecasting Konsep dan Aplikasi*. Yogyakarta: BPFE.
- V. Gaspersz. 1998. *Production Planning and Inventory Control*. Jakarta: Gramedia Pustaka Utama.