

Zaman Serileri Analizi

ELİF EKMEKÇİ

2023-05-11

```
library(readxl)
altindata <- read_excel("/Users/elif/Desktop/altindata.xlsx")
```

```
head(altindata)
```

```
## # A tibble: 6 x 4
##   date                xe xau_usd_ounce xau_try_gram
##   <dtm>              <dbl>         <dbl>         <dbl>
## 1 2013-01-01 00:00:00  1.76          1664.          94.5
## 2 2013-02-01 00:00:00  1.80          1580.          91.9
## 3 2013-03-01 00:00:00  1.81          1597.          93.4
## 4 2013-04-01 00:00:00  1.79          1477.          84.7
## 5 2013-05-01 00:00:00  1.88          1387.          84.8
## 6 2013-06-01 00:00:00  1.93          1234.          74.1
```

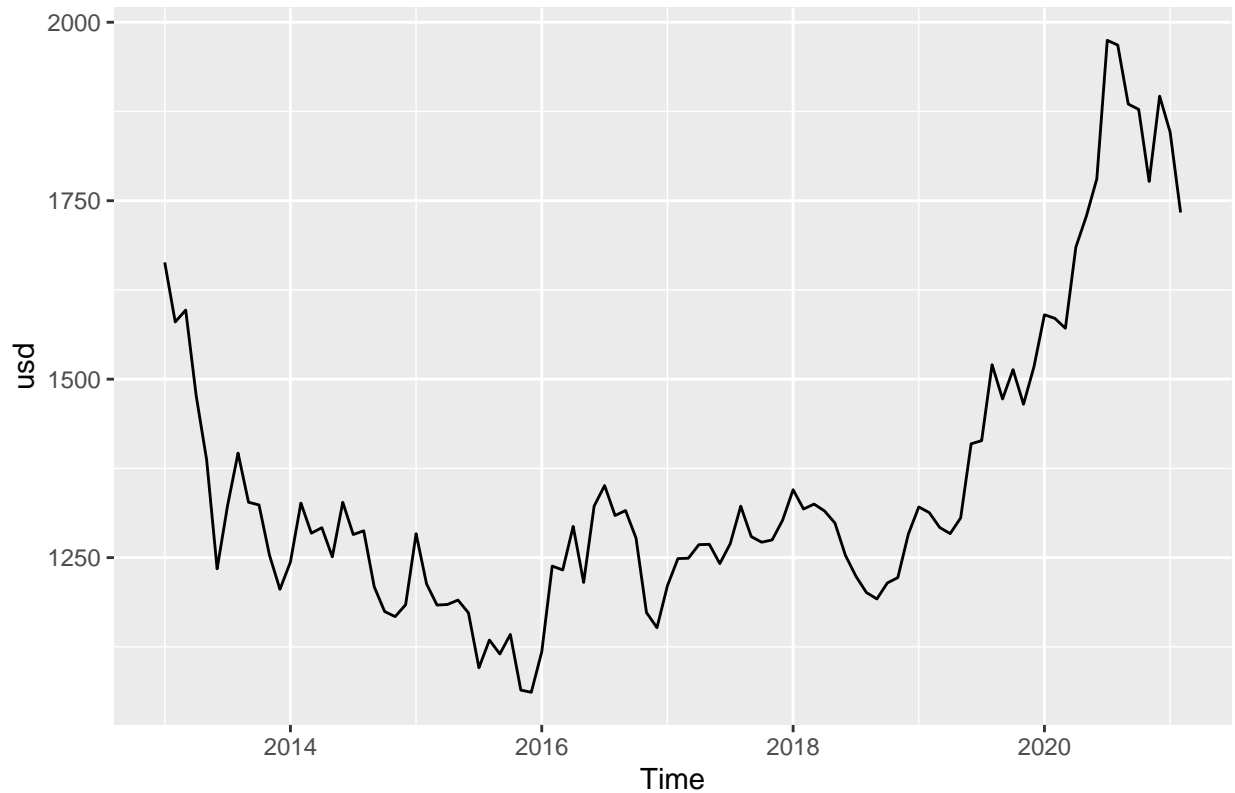
Verimizin ilk 6 gözlemini incelediğimizde aylık bir veri olduğunu ve 2013-01-01 tarihinden başladığını görüyoruz.

Verimizden **xau_usd_ounce** değişkenini modelleyelim. Bunun için `ts()` fonksiyonu ile değişkenimizi seçelim:

```
library(fpp2)
usd <- ts(altindata$xau_usd_ounce, start = c(2013,1), frequency = 12)
# her ayın 1.gununden itibaren aldığı için c(2013,1) şeklinde yazdık
```

xau_usd_ounce değişkenimizin grafiğini çizdirip trend ve mevsimselliğini inceleyelim:

```
autoplot(usd)
```



- Grafik incelendiğinde belirgin net bir trend olmadığını görüyoruz.

Şimdi `xau_usd_ounce` değişkeninin verilerini üstel düzleştirme modelleri ile modelleyip modeller arasından en uygununu belirleyelim:

Verimizi test ve train olarak ayıralım:

```
test <- window(usd, start = c(2020,10))
train <- window(usd, end = c(2020,9))
h <- length(test)
```

BASİT ÜSTEL DÜZLEŞTİRME

İlk olarak **basit üstel düzleştirme** modelini kullanalım.

-Basit üstel düzleştirme modelinde gözlemlere önceki gözlemlere doğru alfa ağırlıkları veriliyor.

```
se <- ses(train, h = h)
summary(se)
```

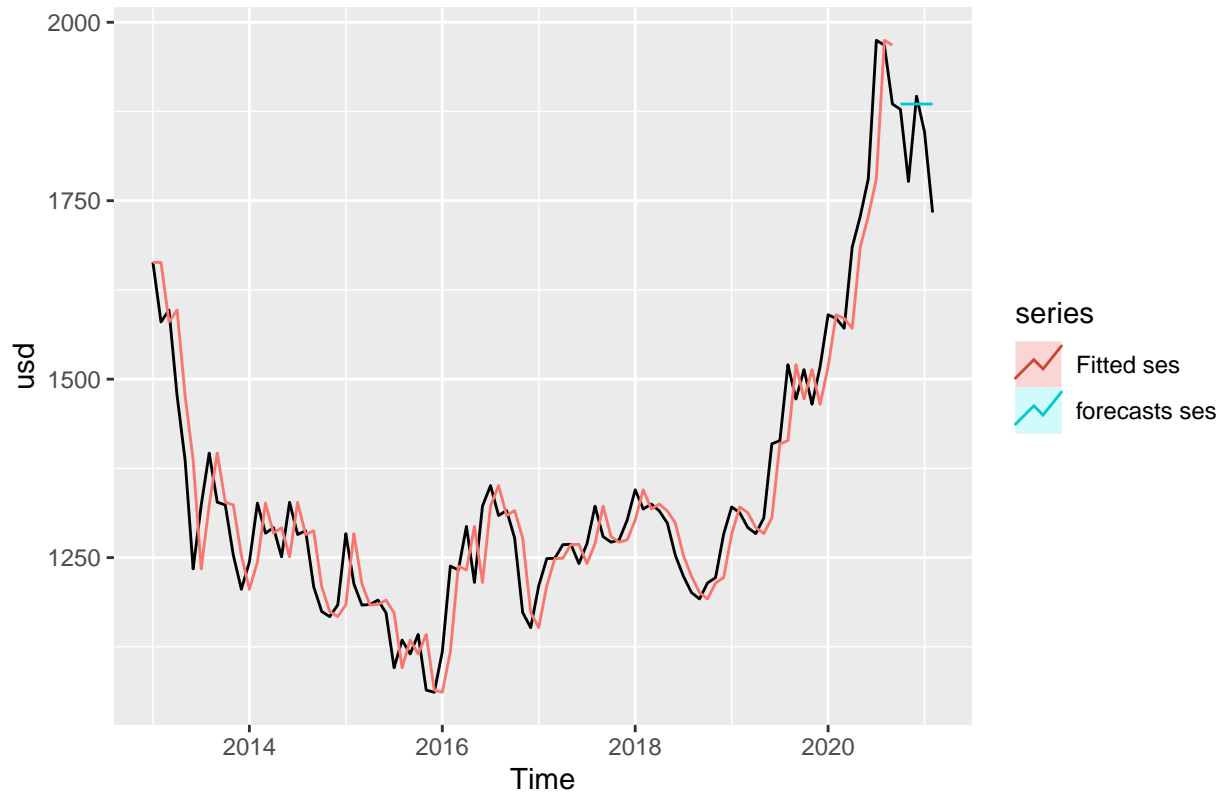
```
##
## Forecast method: Simple exponential smoothing
##
## Model Information:
```

```
## Simple exponential smoothing
##
## Call:
## ses(y = train, h = h)
##
## Smoothing parameters:
##   alpha = 0.9999
##
## Initial states:
##   l = 1663.5497
##
## sigma: 58.8451
##
##      AIC      AICc      BIC
## 1183.443 1183.713 1191.041
##
## Error measures:
##           ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
## Training set 2.386246 58.20892 44.32447 0.04372887 3.30737 0.315972 0.03954407
##
## Forecasts:
##      Point Forecast      Lo 80      Hi 80      Lo 95      Hi 95
## Oct 2020      1885.448 1810.035 1960.861 1770.114 2000.783
## Nov 2020      1885.448 1778.804 1992.093 1722.349 2048.547
## Dec 2020      1885.448 1754.838 2016.059 1685.697 2085.200
## Jan 2021      1885.448 1734.634 2036.263 1654.797 2116.099
## Feb 2021      1885.448 1716.833 2054.063 1627.574 2143.323
```

Çıktı incelendiğinde $\alpha = 0.9999$ olarak hesaplanmıştır. Bu değer 1'e oldukça yakındır bu nedenle modelin ağırlıkları son gözleme verdiğini önceki (ilk) gözlemlere fazla önem (ağırlık) vermediğini anlıyoruz.

Basit üstel düzleştirme modelinden hesapladığımız train set ve test seti üzerindeki tahminlerin, veri seti üzerindeki grafiklerini çizdirelim:

```
autoplot(usd) +
  autolayer(fitted(se), series = "Fitted ses") +
  autolayer(se, series = "forecasts ses", PI = FALSE)
```



Test ve training seti RMSE değerlerini hesaplayalım:

```
a<-accuracy(se,test)
a
```

```
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set  2.386246 58.20892 44.32447  0.04372887 3.307370 0.315972
## Test set     -59.240303 85.52806 63.65698 -3.36335429 3.596241 0.453786
##              ACF1 Theil's U
## Training set  0.03954407      NA
## Test set     -0.33887586 0.9454276
```

Training set Cross validated RMSE hesaplayalım:

```
e1 <- tsCV(train,ses)
RMSE1 <- sqrt(mean(e1^2, na.rm = TRUE))
RMSE1
```

```
## [1] 59.26864
```

HOLT'UN LINEER TREND YÖNTEMİ

Şimdi Holt'un Linear Trend Yöntemini uygulayalım

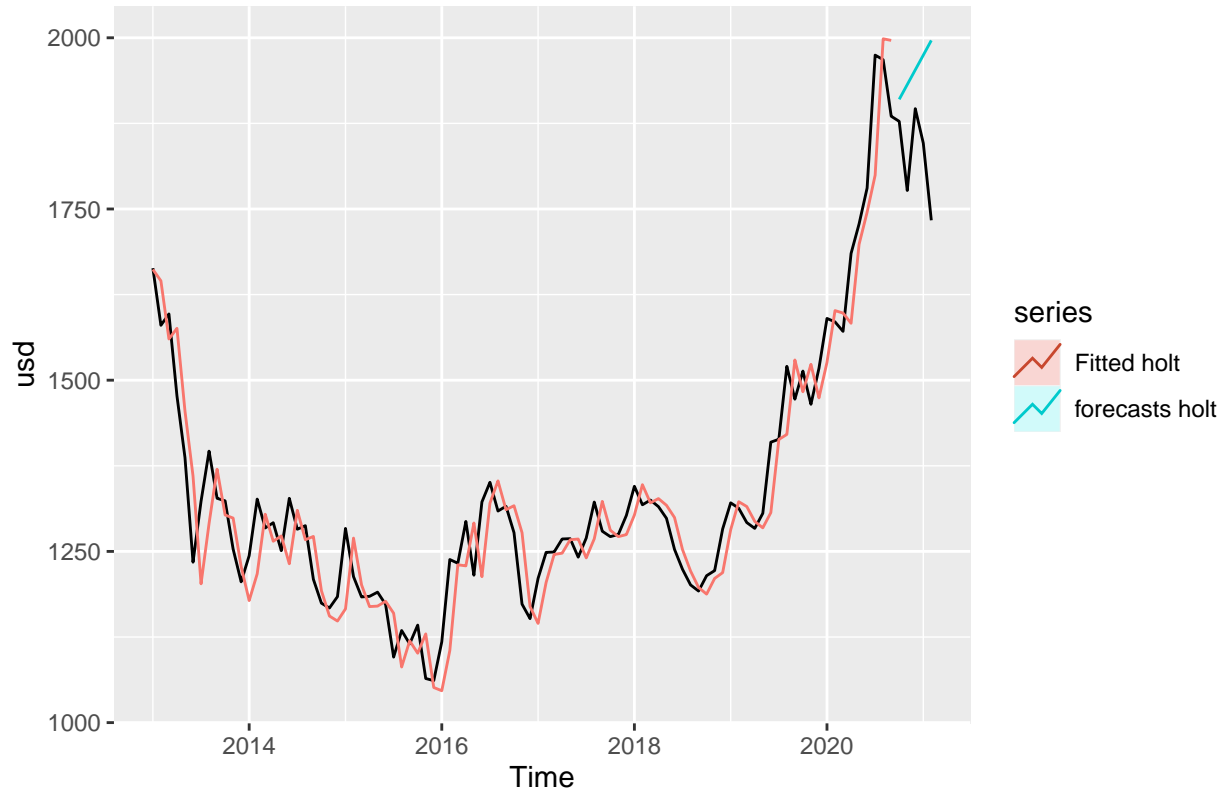
- Bu modelde basit üstel düzeltirme modeline ek olarak trend equation geliyor bu da 2 tane ekstra parametre (b_0 ve β) ekliyor.
- Modelin normal ve sönümlenmeli olarak iki çeşidi var.

Modeli kuralım:

```
modholt <- holt(train, h = h)
```

Hesapladığımız train set ve test seti üzerindeki tahminlerin, veri seti üzerindeki grafiklerini çizdirelim:

```
autoplot(usd) +  
  autolayer(fitted(modholt), series = "Fitted holt") +  
  autolayer(modholt, series = "forecasts holt", PI = FALSE)
```



Test ve training seti RMSE ve Cross validated RMSE değerlerini hesaplayalım:

```
b <- accuracy(modholt, test)  
b
```

```
##           ME      RMSE      MAE      MPE      MAPE      MASE  
## Training set  8.274717 57.82088 44.72772 0.5560221 3.334145 0.3188466  
## Test set     -126.938969 150.78555 126.93897 -7.1038301 7.103830 0.9048987  
##           ACF1 Theil's U  
## Training set -0.02264716      NA  
## Test set     -0.13472659  1.660634
```

```
e2 <- tsCV(train,holt)
RMSE2 <- sqrt(mean(e2^2, na.rm = TRUE))
RMSE2
```

```
## [1] 67.43844
```

HOLT'UN SÖNÜMLEMELİ MODELİ

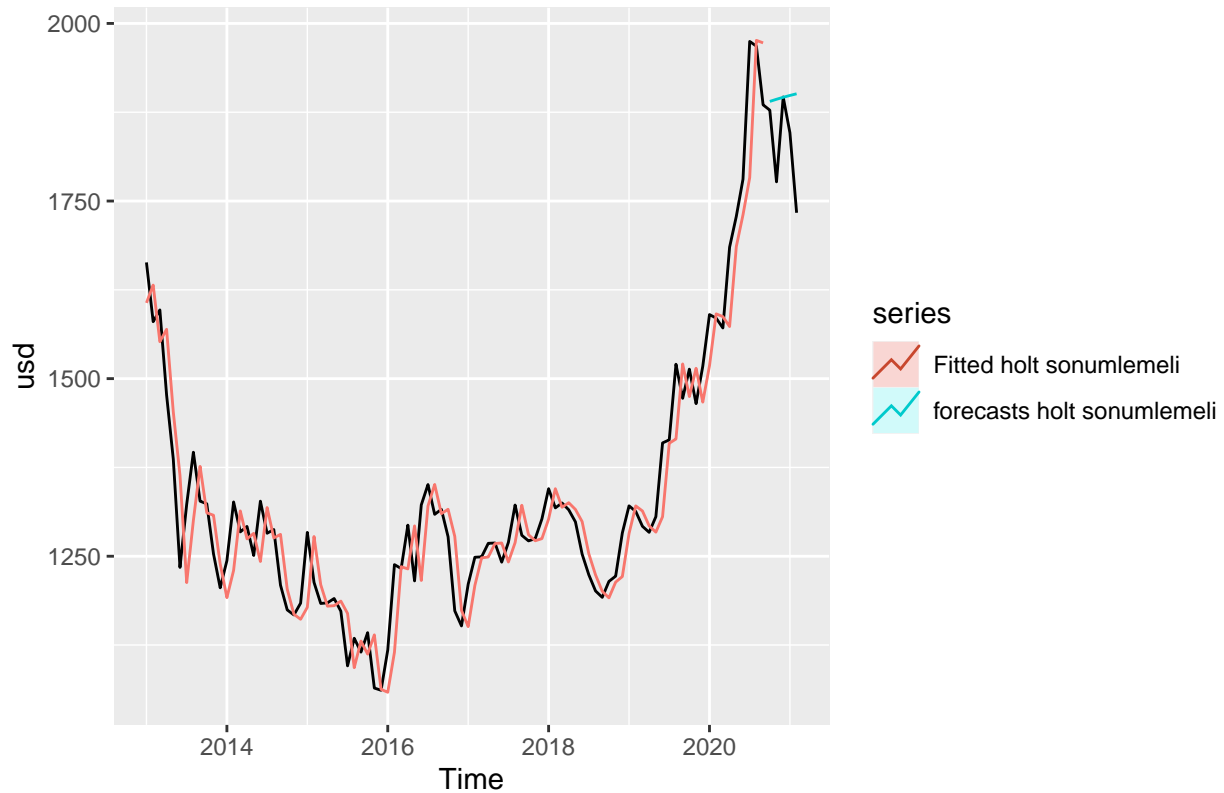
Şimdi Holt'un Sönümlemeli modelinin kullanalım

```
modholtsonumlemeli <- holt(train,damped =TRUE, h =h)
# belirtmedigimiz icin phi degerini kendisi belirleyecek
modholtsonumlemeli$model
```

```
## Damped Holt's method
##
## Call:
## holt(y = train, h = h, damped = TRUE)
##
## Smoothing parameters:
##   alpha = 0.9823
##   beta  = 0.011
##   phi   = 0.9246
##
## Initial states:
##   l = 1640.7147
##   b = -36.8405
##
## sigma: 58.7914
##
##      AIC      AICc      BIC
## 1186.155 1187.132 1201.351
```

Holt'un Sönümlemeli Modeli'nden hesapladığımız train set ve test seti üzerindeki tahminlerin, veri seti üzerindeki grafiklerini çizdirelim:

```
autoplot(usd) +
  autolayer(fitted(modholtsonumlemeli), series = "Fitted holt sonumlemeli") +
  autolayer(modholtsonumlemeli, series = "forecasts holt sonumlemeli", PI = FALSE)
```



Test ve training seti RMSE ve Cross validated RMSE değerlerini hesaplayalım

```
c <- accuracy(modholtsonumlemeli,test)
fdamped <- function(y,h){holt(y,damped = TRUE,h=h)}
# holt diye yazdigimiz fonksiyonun default'unda sonumleme yok
# bu nedenle biz sonumleme iceren fdamped adinda bir fonksiyon yazdik
e3 <- tsCV(train,fdamped)
# train yanina fonksiyon yaziyoruz karistirma
RMSE3 <- sqrt(mean(e3^2, na.rm = TRUE))
RMSE3
```

```
## [1] 63.89074
```

HOLT WINTERS YÖNTEMİ

Holt Winters modelini kurup özet istatistiklerine bakalım

```
holdwin <-hw(train,h=h)
summary(holdwin)
```

```
##
## Forecast method: Holt-Winters' additive method
##
## Model Information:
```

```

## Holt-Winters' additive method
##
## Call:
## hw(y = train, h = h)
##
## Smoothing parameters:
##   alpha = 0.951
##   beta  = 0.0344
##   gamma = 1e-04
##
## Initial states:
##   l = 1507.3888
##   b = 3.988
##   s = -37.2147 -48.271 -9.8832 -4.6892 29.1468 6.0212
##       15.1147 1.0625 17.1054 -12.8233 22.988 21.4427
##
## sigma: 60.68
##
##      AIC      AICc      BIC
## 1201.618 1209.778 1244.672
##
## Error measures:
##           ME      RMSE      MAE      MPE      MAPE      MASE
## Training set 4.056592 55.21403 42.36356 0.1983762 3.138439 0.3019934
##           ACF1
## Training set -0.005881105
##
## Forecasts:
##      Point Forecast      Lo 80      Hi 80      Lo 95      Hi 95
## Oct 2020      1900.638 1822.873 1978.402 1781.707 2019.568
## Nov 2020      1879.191 1770.018 1988.363 1712.226 2046.156
## Dec 2020      1907.199 1772.267 2042.131 1700.839 2113.559
## Jan 2021      1982.817 1824.940 2140.695 1741.365 2224.270
## Feb 2021      2001.301 1822.167 2180.435 1727.339 2275.263

```

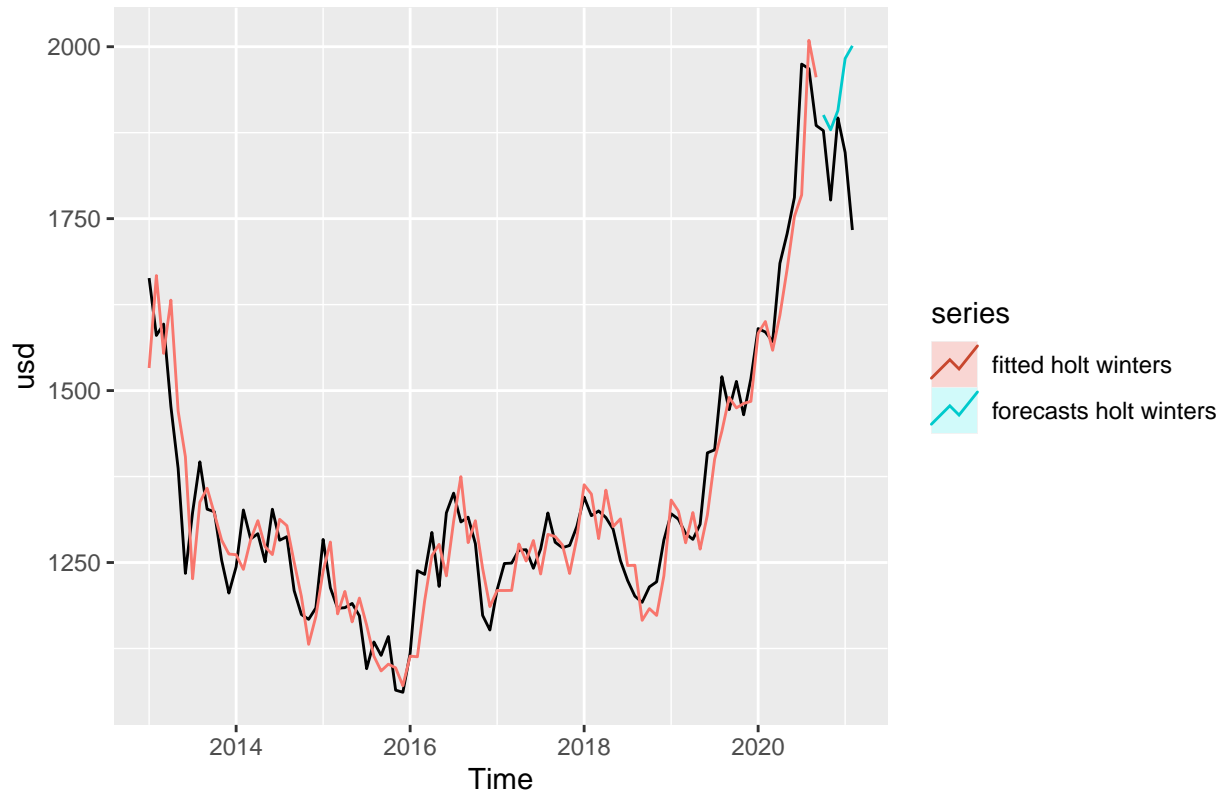
Kasım 2020 için nokta tahminin 1879.191 olarak hesaplandığını görüyoruz.

Holt Winters Yöntemi kullanarak elde ettiğimiz train ve test seti tahminlerinin grafiğini çizdirelim

```

autoplot(usd)+
  autolayer(fitted(holdwin),series = "fitted holt winters")+
  autolayer(holdwin,series ="forecasts holt winters",PI=FALSE )

```

Test ve training seti RMSE ve Cross validated RMSE değerlerini hesaplayalım

```
##test ve training set RMSE değerleri
d <-accuracy(holdwin,test)
## training set cross validated RMSE değerleri
e4 <-tsCV(train,hw)
# hw fonksiyonun adi
RMSE4 <-sqrt(mean(e4^2,na.rm=TRUE))
RMSE4
```

```
## [1] 62.65128
```

HOLT WINTERS YÖNTEMİ (ÇARPIMSAL MEVSİMSSEL MODEL)

Holt Winters'ın çarpımsal mevsimsellik içeren modelini kurup özet istatistiklerine bakalım

```
holdwinmultip <-hw(train,h=h,seasonal = "multiplicative")
summary(holdwinmultip)
```

```
##
## Forecast method: Holt-Winters' multiplicative method
##
## Model Information:
## Holt-Winters' multiplicative method
```

```
##
## Call:
## hw(y = train, h = h, seasonal = "multiplicative")
##
## Smoothing parameters:
##   alpha = 0.6207
##   beta  = 0.0308
##   gamma = 1e-04
##
## Initial states:
##   l = 1497.3449
##   b = -5.5031
##   s = 0.9675 0.9629 0.9914 0.9836 1.0269 1.0217
##       1.0026 0.9835 1.0053 1.0045 1.0237 1.0263
##
## sigma: 0.0461
##
##      AIC      AICc      BIC
## 1200.784 1208.944 1243.838
##
## Error measures:
##           ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
## Training set 9.975358 56.75553 42.93298 0.5937581 3.215816 0.3060526 0.3079967
##
## Forecasts:
##      Point Forecast      Lo 80      Hi 80      Lo 95      Hi 95
## Oct 2020      1895.781 1762.424 2029.139 1691.829 2099.734
## Nov 2020      1927.254 1771.798 2082.709 1689.504 2165.003
## Dec 2020      2067.966 1881.026 2254.906 1782.066 2353.866
## Jan 2021      2086.295 1878.226 2294.363 1768.082 2404.508
```

Kasım 2020 için nokta tahminin 1895.781 olarak hesaplandığını görüyoruz.

Holt Winters'ın çarpımsal mevsimsellik içeren yöntemini kullanarak elde ettiğimiz train ve test seti tahminlerinin grafiğini çizdirelim

```
autoplot(usd)+
  autolayer(fitted(holdwinmultip),series = "fitted holt winters multiplicative")+
  autolayer(holdwinmultip,series = "forecasts holt winters multiplicative",PI=FALSE )
```



Test ve training seti RMSE ve Cross validated RMSE değerlerini hesaplayalım

```
##test ve training set RMSE değerleri
e<-accuracy(holdwinmultip,test)
## training set cross validated RMSE değerleri
fhw<-function(y,h){hw(y,seasonal = "multiplicative",h=h)}
e5<-tsCV(train,fhw,h=1)
RMSE5<-sqrt(mean(e5^2,na.rm=TRUE))
RMSE5
```

```
## [1] 76.13876
```

ETS YÖNTEMİ

ETS modelini kuralım

```
etsmod <- ets(train)
etsmod
```

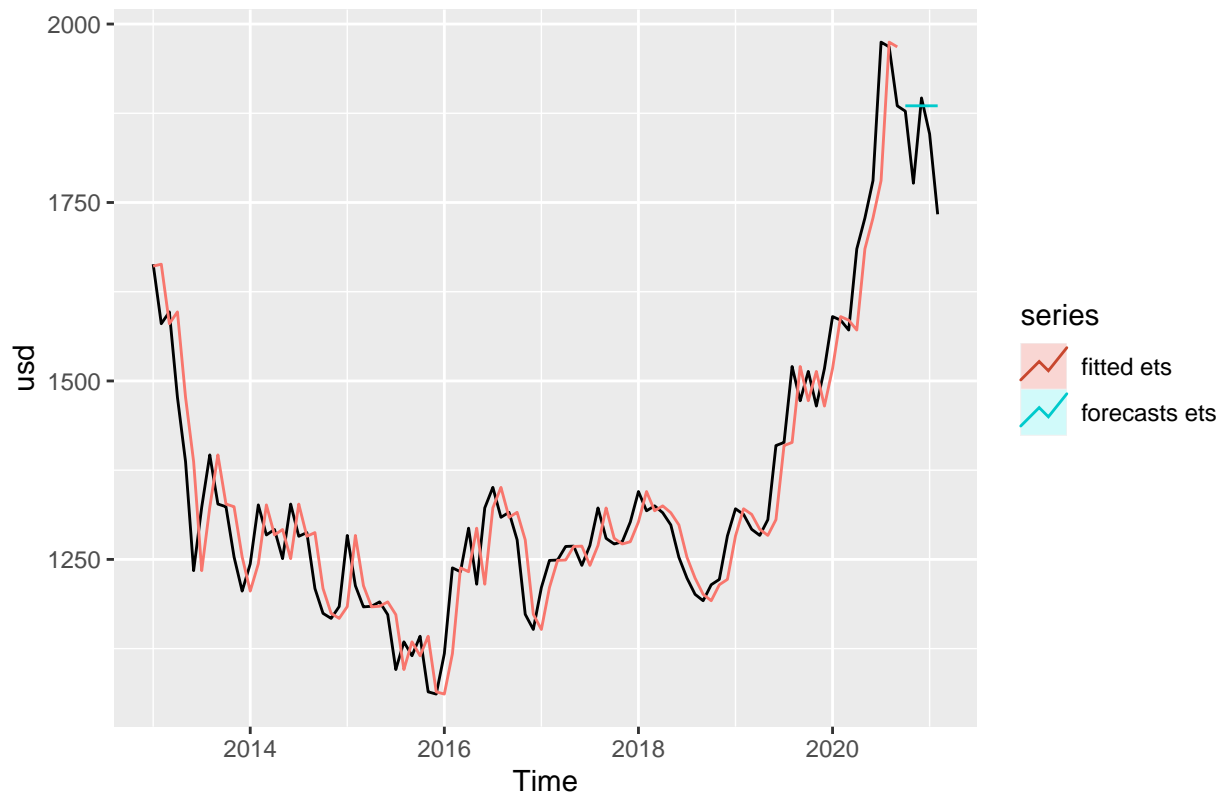
```
## ETS(M,N,N)
##
## Call:
## ets(y = train)
##
## Smoothing parameters:
```

```
##      alpha = 0.9999
##
##      Initial states:
##      l = 1660.831
##
##      sigma: 0.0434
##
##      AIC      AICc      BIC
## 1177.817 1178.087 1185.415
```

Çıktıyı incelediğimizde ETS(M,N,N) modelinin seçildiğini görüyoruz. Bu modelde multiplicative error, trend ve mevsimsellik yok.

ETS yöntemini kullanarak elde ettiğimiz train ve test seti tahminlerinin grafiğini çizdirelim

```
autoplot(usd)+
  autolayer(fitted(etsmod),series="fitted ets") +
  autolayer(forecast(etsmod,h=h),series="forecasts ets",PI=FALSE)
```



Test ve training seti RMSE ve Cross validated RMSE değerlerini hesaplayalım

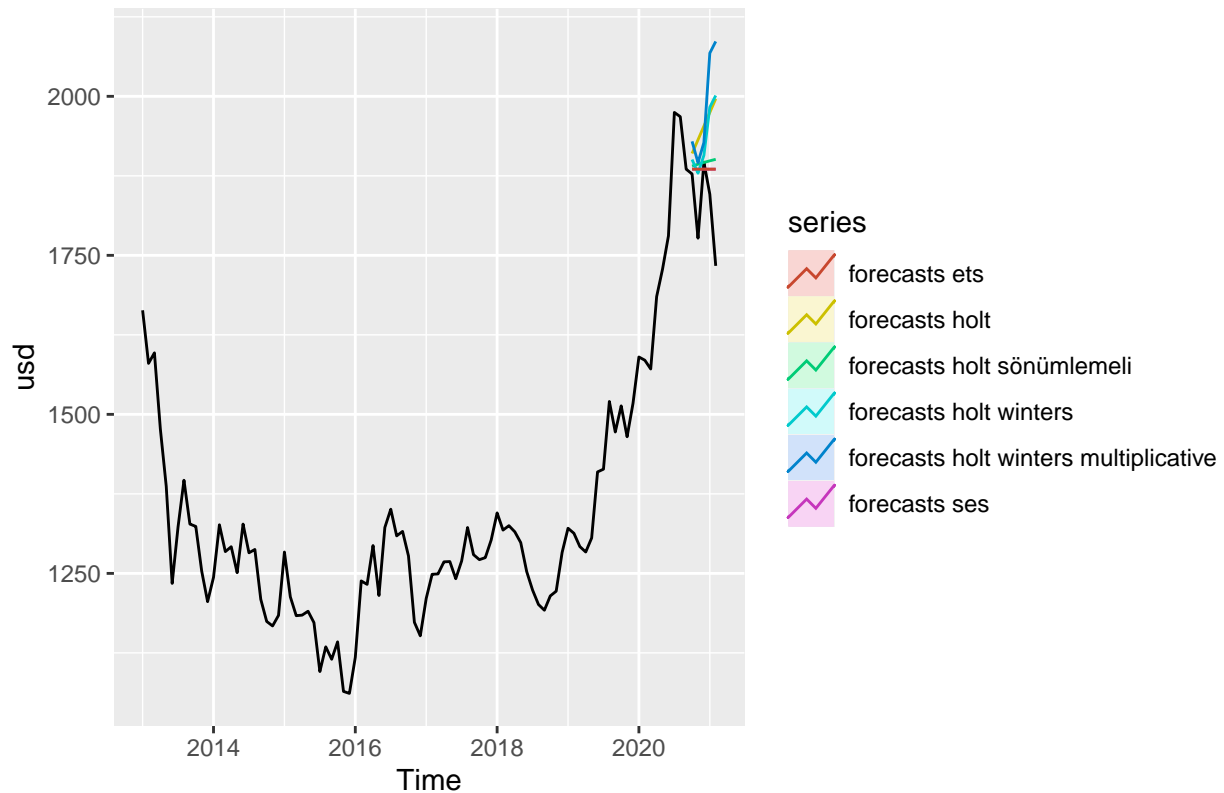
```
##test ve training set RMSE değerleri
f <-accuracy(forecast(etsmod,h=h),test)
## training set cross validated RMSE değerleri
fets <-function(y,h){forecast(ets(y,model="MNN"),h=h)}
# etsmod çıktısından MNN model seçtiğine bakmistik bu yuzden modele MNN yazdik
```

```
e6<-tsCV(train,fets,h=1)
RMSE6<-sqrt(mean(e6^2,na.rm=TRUE))
RMSE6
```

```
## [1] 59.37798
```

Şimdi kurduğumuz tüm modellerin forecastlerini aynı grafik üzerinde gösterelim.

```
autoplot(usd)+autolayer(se,series ="forecasts ses",PI=FALSE )+
  autolayer(modholt,series ="forecasts holt",PI=FALSE )+
  autolayer(modholtsonumlemeli,series ="forecasts holt sönümlemeli",PI=FALSE )+
  autolayer(holdwin,series ="forecasts holt winters",PI=FALSE )+
  autolayer(holdwinmultip,series ="forecasts holt winters multiplicative",PI=FALSE )+
  autolayer(forecast(etsmod,h=h),series ="forecasts ets",PI=FALSE )
```



```
TABLO <- matrix(c(RMSE1,a[2,2],RMSE2,b[2,2],RMSE3,c[2,2],
  RMSE4,d[2,2],RMSE5,e[2,2],RMSE6,f[2,2]),
  nrow = 6, ncol = 2,byrow = T)

rownames(TABLO)<- c("ses","holt","holt_damped",
  "holtwinters_additive",
  "holtwinters_multiplicative","ets")

colnames(TABLO)<- c("CVrmse","testRMSE")
TABLO
```

| ## | CVrmse | testRMSE |
|-------------------------------|----------|-----------|
| ## ses | 59.26864 | 85.52806 |
| ## holt | 67.43844 | 150.78555 |
| ## holt_damped | 63.89074 | 94.31004 |
| ## holtwinters_additive | 62.65128 | 142.46776 |
| ## holtwinters_multiplicative | 76.13876 | 195.64119 |
| ## ets | 59.37798 | 85.52826 |

Sonuçlar karşılaştırıldığında en iyi modelin basit üstel düzeltirme modeli(ses) olduğu görülmektedir.