

# Arima.R

Fiona

Fri Oct 13 16:02:08 2017

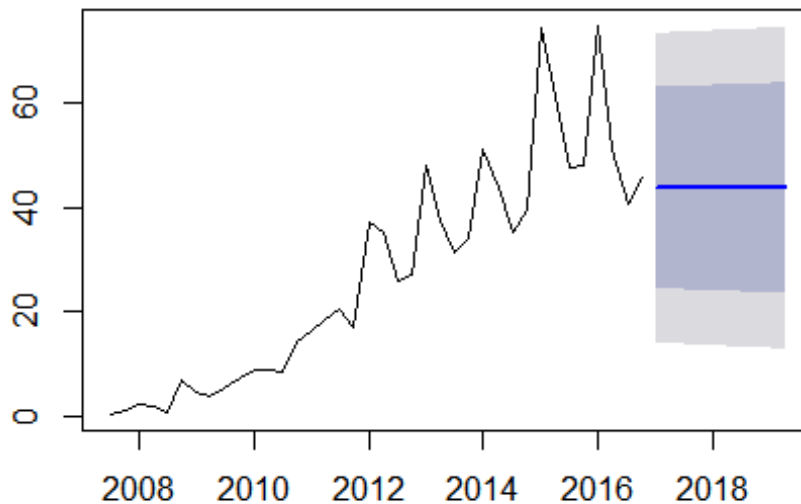
```
# Install packages
#install.packages("forecast")
library("forecast")

## Warning: package 'forecast' was built under R version 3.3.3

# Import data
data <- read.csv("C:/Users/Fiona/Desktop/Business Analytics and Decision
Sciences/Forecasting and Business Analytics/FABA-L3-Notes
(1)/iphonesales.csv", header=TRUE)
# save as time series data
ipts <- ts(data$Sales, start=c(2007, 3), end=c(2016, 4), frequency=4)

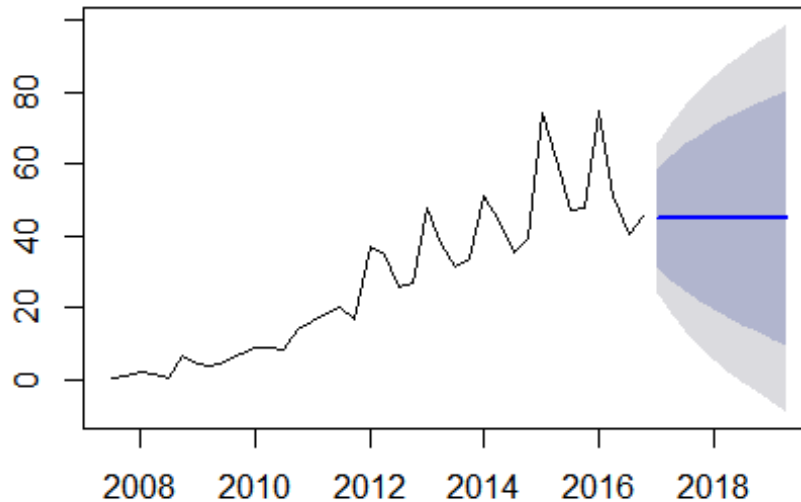
# Simple Exponential Smoothing - Low alpha
SES <- ses(ipts, alpha=0.1)
plot(SES)
```

## Forecasts from Simple exponential smoothing



```
# Simple Exponential Smoothing - high alpha
SES2 <- ses(ipts, alpha=0.8)
plot(SES2)
```

## Forecasts from Simple exponential smoothing

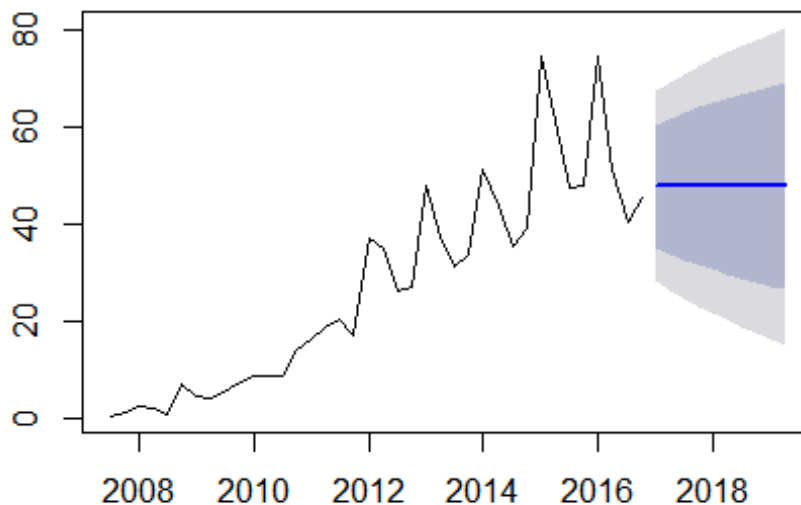


```
# Simple Exponential Smoothing with alpha calculated for you
SES3 <- ses(ipts)
summary(SES3)
```

```
##
## Forecast method: Simple exponential smoothing
##
## Model Information:
## Simple exponential smoothing
##
## Call:
## ses(y = ipts)
##
## Smoothing parameters:
##   alpha = 0.4446
##
## Initial states:
##   l = 1.1887
##
## sigma: 10.0195
##
##      AIC      AICc      BIC
## 319.3729 320.0788 324.2856
```

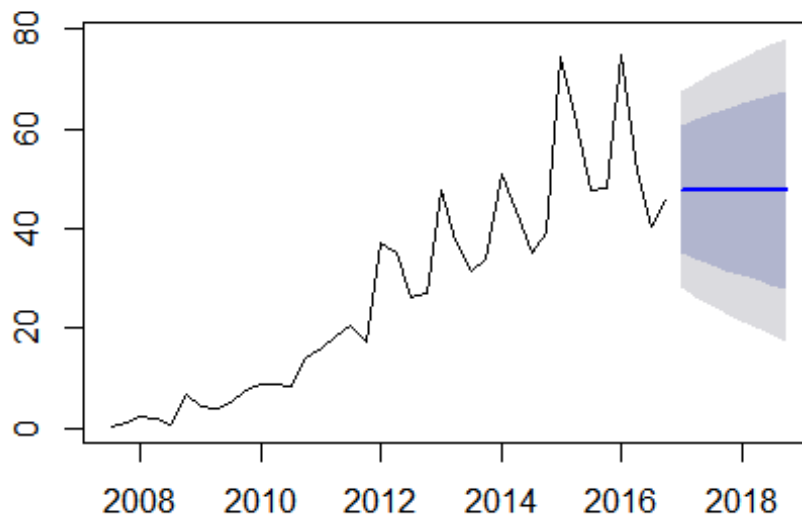
```
##
## Error measures:
##           ME      RMSE      MAE      MPE      MAPE      MASE
## Training set 2.757327 10.01951 6.343764 1.563795 35.17341 0.8776009
##           ACF1
## Training set -0.0001853376
##
## Forecasts:
##           Point Forecast      Lo 80      Hi 80      Lo 95      Hi 95
## 2017 Q1          47.76988 34.92936 60.61040 28.13200 67.40776
## 2017 Q2          47.76988 33.71762 61.82213 26.27881 69.26095
## 2017 Q3          47.76988 32.60239 62.93737 24.57320 70.96655
## 2017 Q4          47.76988 31.56371 63.97604 22.98469 72.55506
## 2018 Q1          47.76988 30.58772 64.95204 21.49203 74.04772
## 2018 Q2          47.76988 29.66426 65.87550 20.07972 75.46004
## 2018 Q3          47.76988 28.78566 66.75409 18.73602 76.80373
## 2018 Q4          47.76988 27.94597 67.59379 17.45183 78.08793
## 2019 Q1          47.76988 27.14043 68.39933 16.21985 79.31990
## 2019 Q2          47.76988 26.36518 69.17458 15.03421 80.50554
plot(SES3)
```

## Forecasts from Simple exponential smoothing



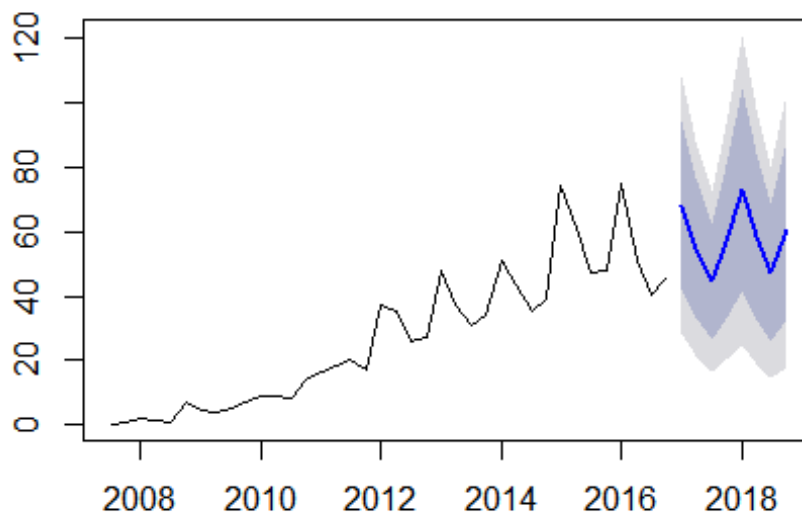
```
# Simple Exponential Smoothing using the ets function
se <- ets(ipts, model="ANN")
plot(forecast(se))
```

### Forecasts from ETS(A,N,N)

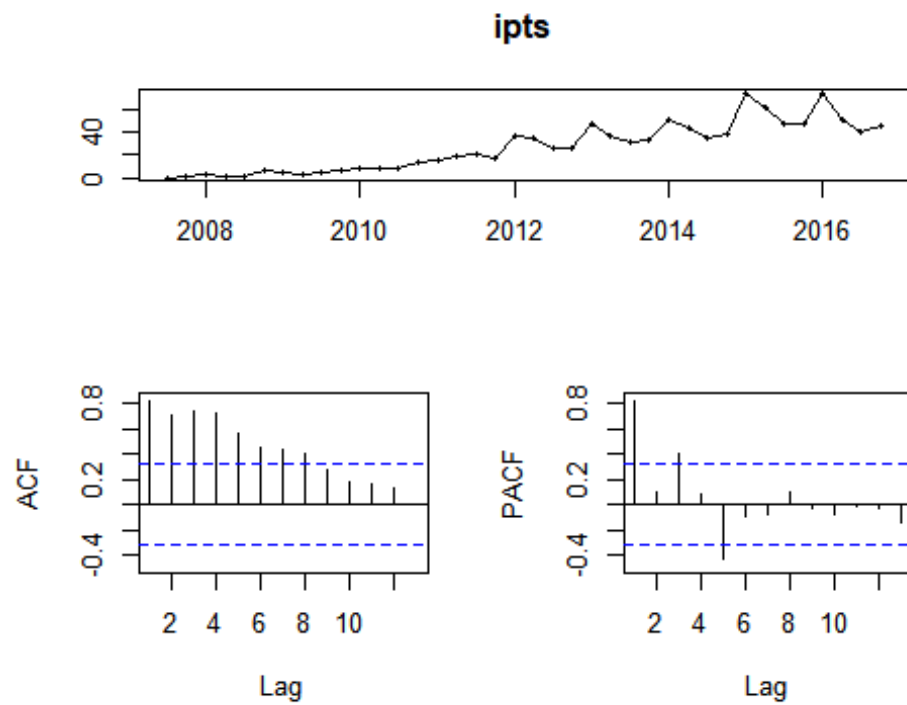


```
# Multiplicative Holt-Winters' method with multiplicative errors  
se <- ets(ipts, model="MAM")  
plot(forecast(se))
```

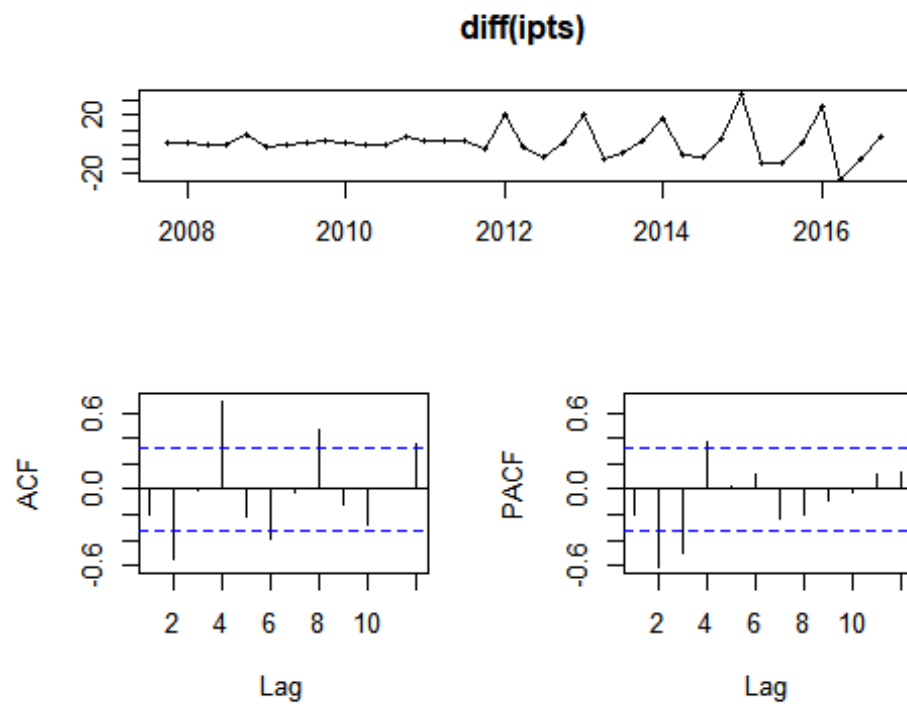
### Forecasts from ETS(M,A,M)



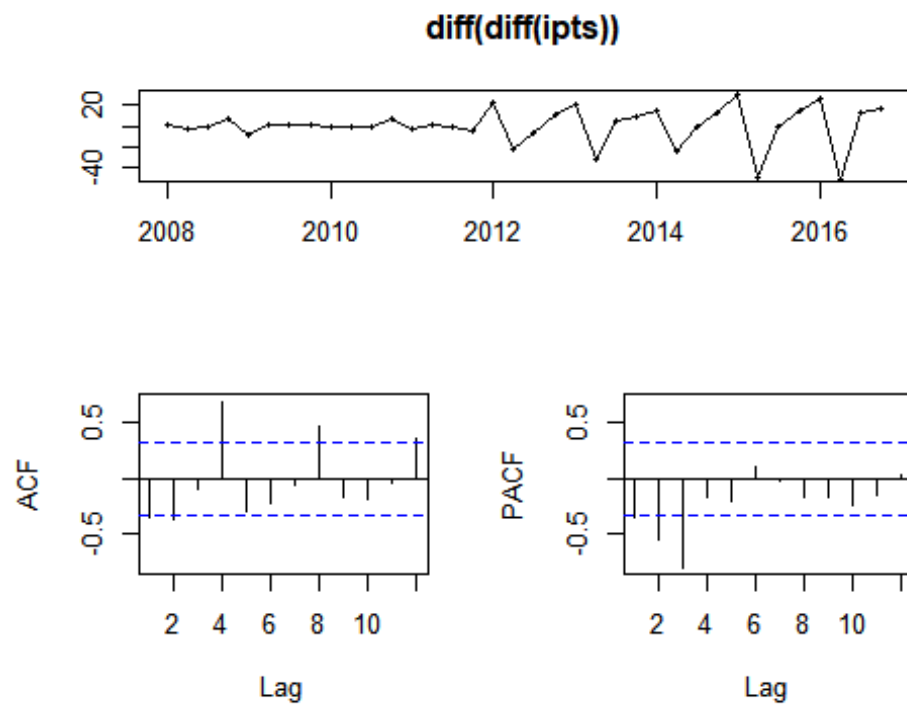
```
# ACF and PACF plots  
tsdisplay(ipts)
```



```
# first differenced plots  
tsdisplay(diff(ipts))
```

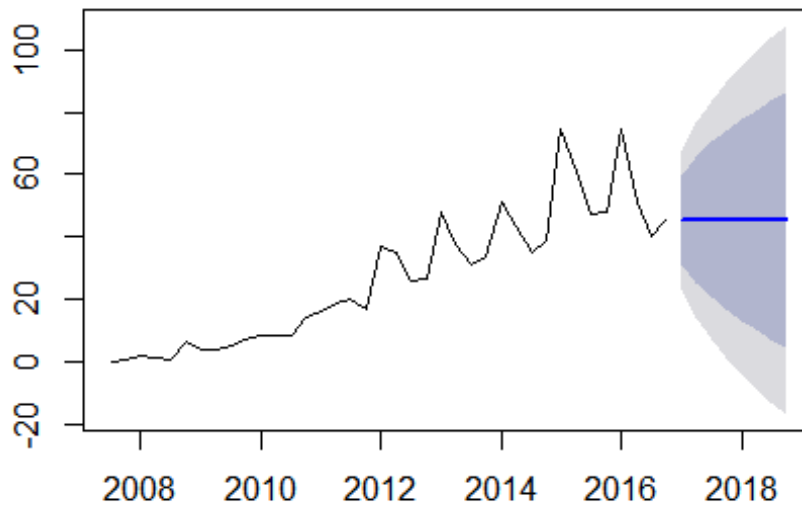


```
# second differenced plots  
tsdisplay(diff(diff(ipts)))
```

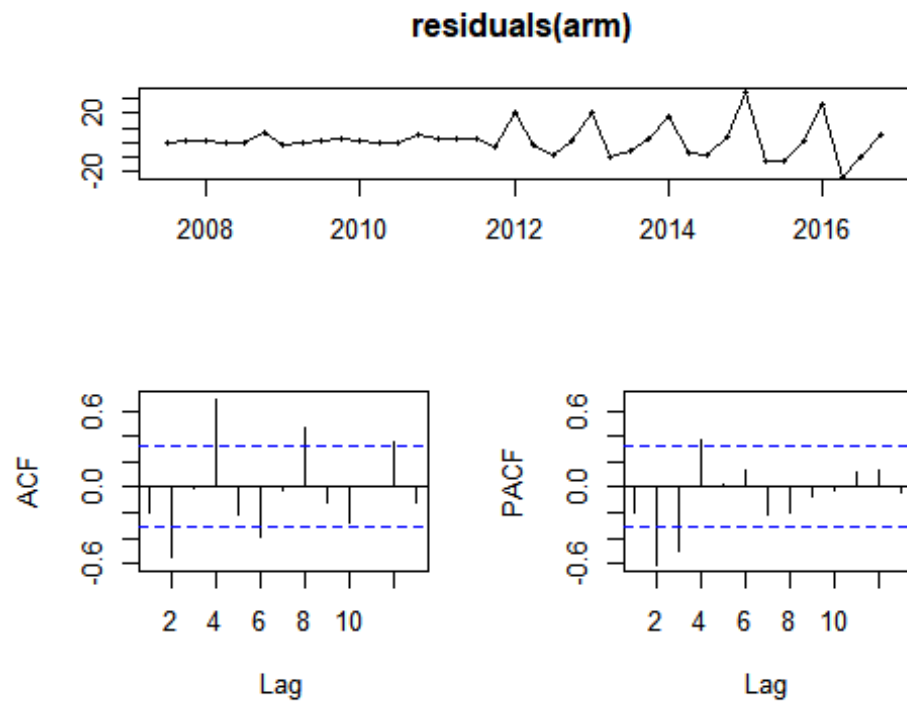


```
# ARIMA (0,1,0)
arm <- arima(ipts,order=c(0,1,0))
plot(forecast(arm))
```

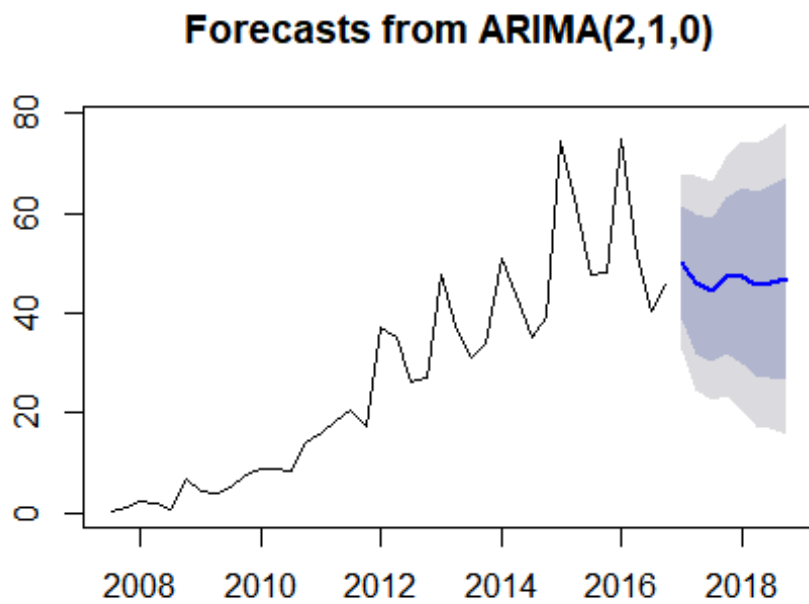
### Forecasts from ARIMA(0,1,0)



```
tsdisplay(residuals(arm))
```

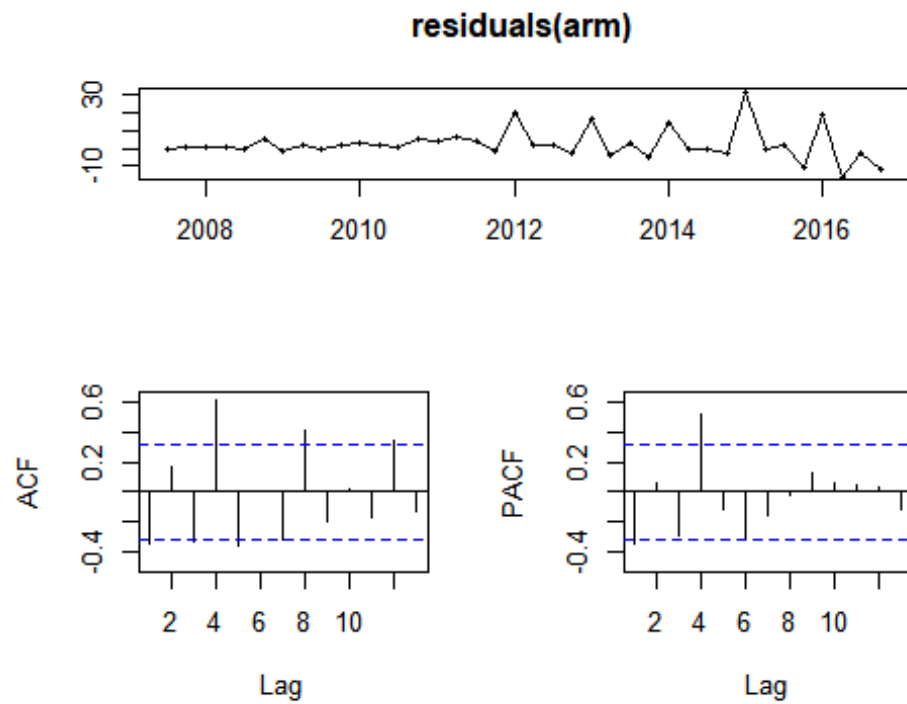


```
# ARIMA (2,1,0)
arm <- arima(ipts,order=c(2,1,0))
plot(forecast(arm))
```



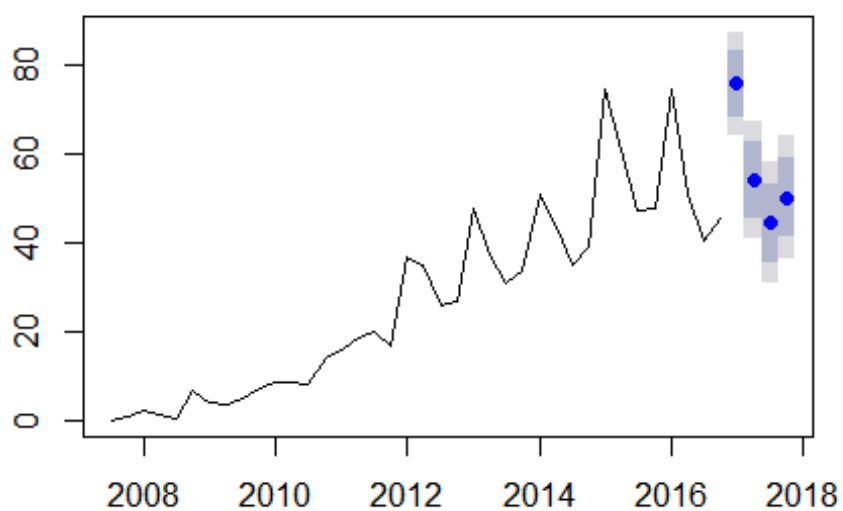


```
tsdisplay(residuals(arm))
```



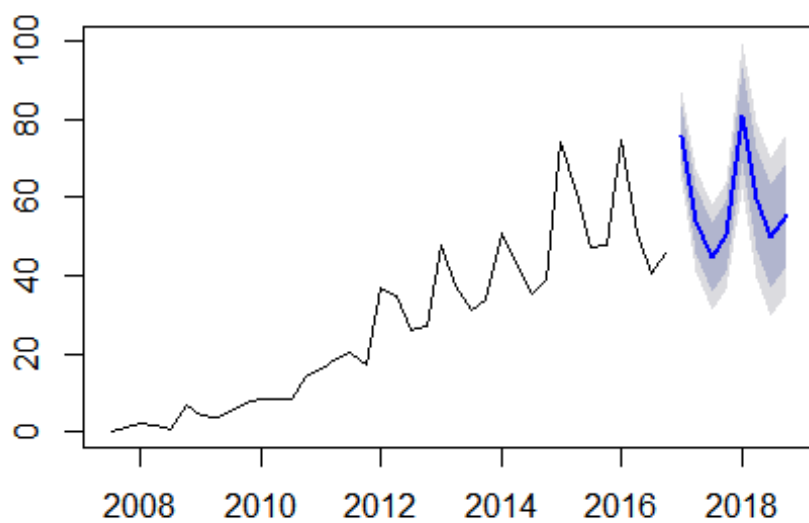
```
# Autofit ARIMA inc seasonal  
afit <- auto.arima(ipts)  
# Plot 1yr ARIMA forecast  
plot(forecast(afit,h=4))
```

### Forecasts from ARIMA(1,0,0)(0,1,0)[4] with drift



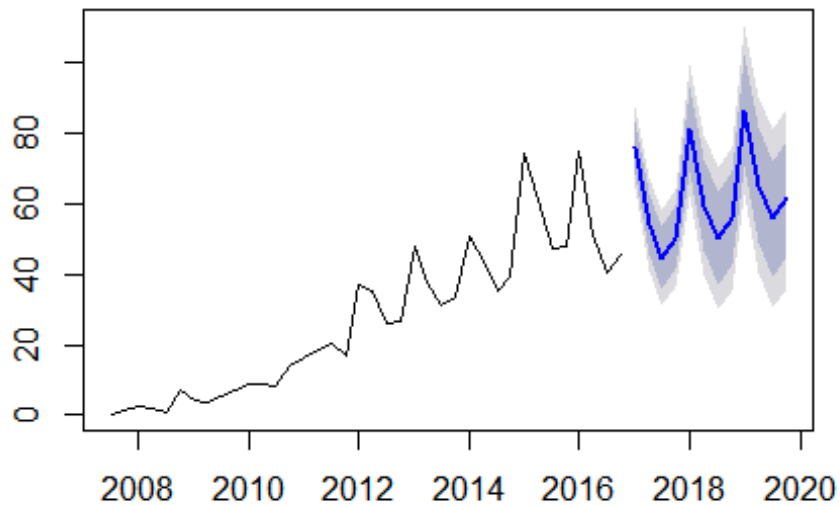
```
# Plot 2yr ARIMA forecast  
plot(forecast(afit,h=8))
```

### Forecasts from ARIMA(1,0,0)(0,1,0)[4] with drift



```
# Plot 3yr ARIMA forecast
plot(forecast(afit,h=12))
```

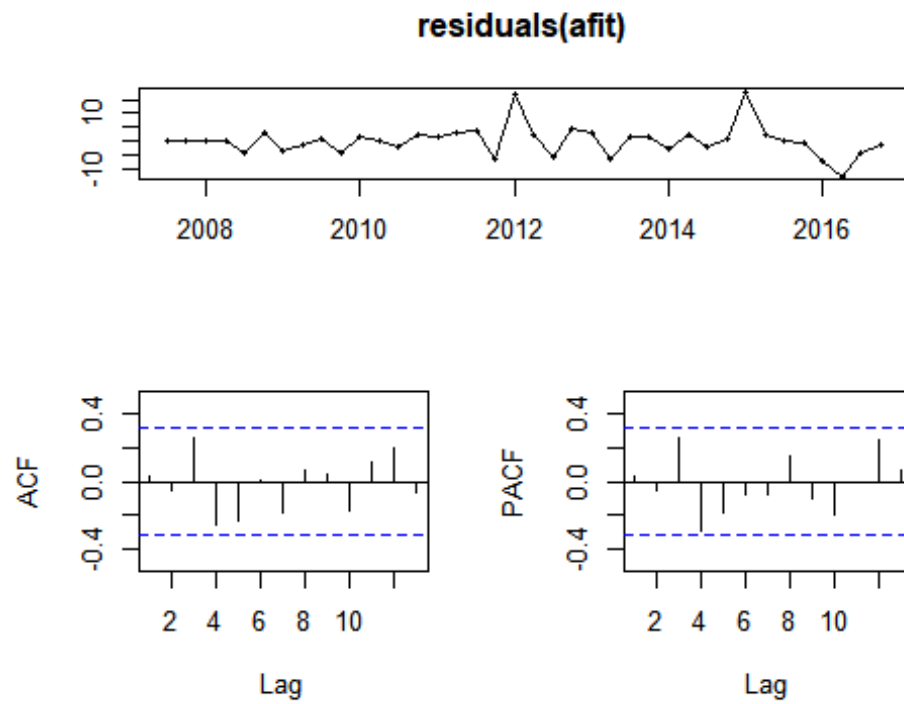
### Forecasts from ARIMA(1,0,0)(0,1,0)[4] with drift



```
afit

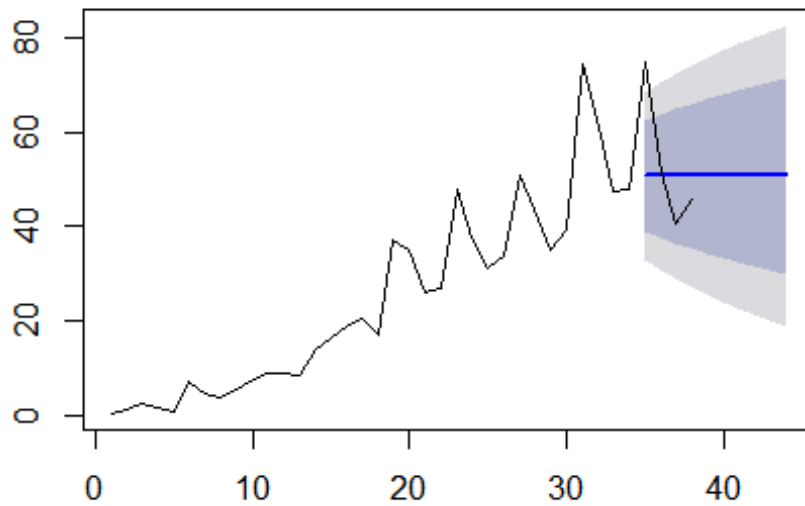
## Series: ipts
## ARIMA(1,0,0)(0,1,0)[4] with drift
##
## Coefficients:
##          ar1    drift
##          0.5537  1.3971
## s.e.  0.1441  0.5312
##
## sigma^2 estimated as 34.32:  log likelihood=-107.5
## AIC=221.01   AICc=221.81   BIC=225.59

# ACF and PACF plots
tsdisplay(residuals(afit))
```



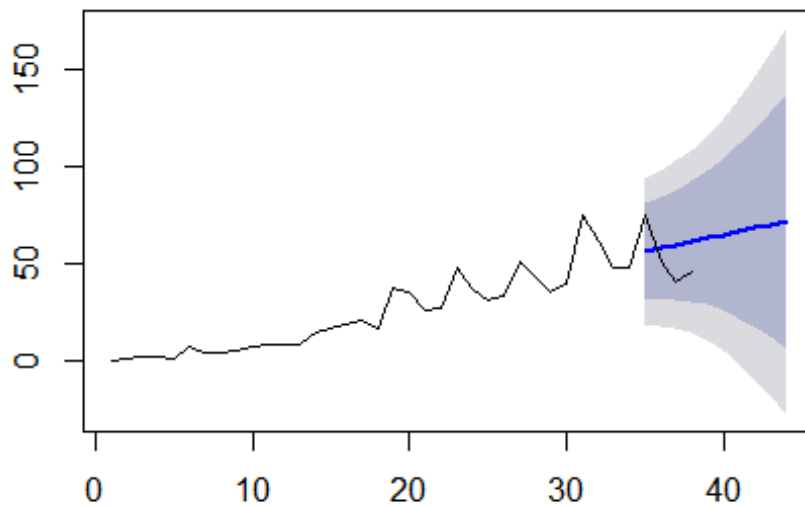
```
fit1 <- ets(ipts[1:34], model="ANN", n=4)
fit2 <- ets(ipts[1:34], model="ZZZ", n=4)
plot(forecast(fit1))
lines(ipts[1:38])
```

### Forecasts from ETS(A,N,N)



```
plot(forecast(fit2))  
lines(ipts[1:38])
```

### Forecasts from ETS(M,A,N)



```
accuracy(forecast(fit1), ipts[35:38])
```

```
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set 2.960779  9.133071  5.402976  3.2517428 34.55871 0.869408
## Test set    2.274466 13.352844 10.015000 -0.9263074 17.51283 1.611542
##              ACF1
## Training set -0.0002953298
## Test set      NA
```

```
accuracy(forecast(fit2), ipts[35:38])
```

```
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set 0.2432179  8.546806  5.960392 -14.47083 35.68646 0.9591035
## Test set    -5.8608090 15.947244 15.144664 -17.79915 30.21404 2.4369706
##              ACF1
## Training set 0.1919996
## Test set      NA
```