````
---
title: "Regression"
output: html_document
authors: "Ryan Daher, Karl Juhl, Qiji Xiang, Joaquin"
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```

## R Markdown

This is an R Markdown document. Markdown is a simple formatting
syntax for authoring HTML, PDF, and MS Word documents. For more
details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated
that includes both content as well as the output of any embedded
R code chunks within the document. You can embed an R code chunk
like this:

```{r}
library(fpp2)
library(forecast)
library(ggplot2)
library(seasonal)
library(astsa)
library(urca)
library(tseries)
library(tidyverse)
```

Step 1: Data Cleaning / Formatting
```{r}
getwd()
df <- read.csv("Airport_Monthly_Operational_Report.csv")

df

df<- df[seq(dim(df)[1],1),]
dft <- ts(df[,18], start=c(2013,6), end=c(2019,10),frequency =
12)
dft[37] = 1107545
autoplot(dft)+ylab("Total Passengers")+ggtitle("Traffic at ABIA")
```
````

````
flightmu <- decompose(dft, type = c("multiplicative"))
flightmu %>% autoplot()
seasonTtrend <- flightmu$seasonal*flightmu$trend

autoplot(dft)+
  autolayer(seasonTtrend)+
  ggtitle("Multiplcative Seasonality and Trend")+
  ylab("Total Passengers")
```

Split into Train and Test groups for modelling and model
evaluation
```{r}
flightrain <- window(dft, end=c(2017,12))
flightest <- window(dft, start=c(2018,1))

autoplot(dft)+
  autolayer(flightrain)+
  autolayer(flightest)
```


We did not end up using Holt or Holt winers in the final model
but worth checking
```{r}
h1 <- length(flightest)
h1

#holt-winters
hw1 <- hw(flightrain, h = h1)
autoplot(dft)+
  autolayer(hw1,PI=F)
accuracy(hw1, flightest)

#holt-winters damped
hw1d <- hw(flightrain, damped = T, h = h1)
autoplot(dft)+
  autolayer(hw1d,PI=F)
accuracy(hw1d, flightest)
```

ETS
With auto ets model selection we get to a multiplicative trend,
damped additive seasonality and multiplicative residuals. We
````
````

````
etsflight3

etsforecast <- forecast(etsflight,h=length(flightest))
etsforecast2 <- forecast(etsflight2,h=length(flightest))
etsforecast3 <- forecast(etsflight3,h=length(flightest))

accuracy(etsforecast,flightest)
accuracy(etsforecast2,flightest)
accuracy(etsforecast3,flightest)

autoplot(dft)+
  autolayer(flightest)+
  autolayer(etsforecast,PI=F,colour='blue')+
  ggtitle('ETS Models')+
  ylab('Total Passenger')
  #autolayer(etsforecast2,PI=F,colour='green')

```

We can see that the MAA model is the best one with the lowest
errors. There is no lambda transformation or damping. A reason
for not damping may be because total passangers are growing at a
faster rate in recent years and not leveling off.

```{r}
autoplot(dft)+
  autolayer(etsforecast,PI=F, colour = 'red')
```


Based on ljung box it seems that residuals are not correlated.
Although ACF shows no significant lags. But residual plot looks
still correlated. They are white noise however.


```{r}
checkresiduals(etsforecast)
shapiro.test(etsforecast$residuals)
```

Data is not stationary and has a change of variance so let's
apply boxcox transformation.

```{r}
````

```
dftd1 %>% autoplot()

adf.test(dftd1)
summary(ur.kpss(dftd1))

ndiffs(dftd1,alpha=0.05,test=c("kpss"))
nsdiffs(dftd1,alpha=0.05,test=c("seas"))

dftd2 <- diff(dftd1)
adf.test(dftd2)
summary(ur.kpss(dftd2))

dftd2 %>% autoplot()
```

Autoarima -709.04 -

```{r}
ggtsdisplay(dftd2)

auto.arima(dftd2,seasonal=TRUE,nmodels=10000,stepwise=FALSE)

Arima(dftd2,order=c(0,0,1),seasonal=c(0,0,1),include.mean = F)
Arima(dftd2,order=c(0,0,1),seasonal=c(0,0,1),include.mean = F)
Arima(dftd2,order=c(0,0,1),seasonal=c(0,0,1),include.mean = F)

Arima(dftd2,order=c(0,0,2),seasonal=c(1,0,0),include.mean =
F,lambda=l)
Arima(dftd2,order=c(0,0,3),seasonal=c(1,0,0),include.mean =
F,lambda=l)
Arima(dftd2,order=c(1,0,1),seasonal=c(0,0,1),include.mean =
F,lambda=l)
Arima(dftd2,order=c(1,0,1),seasonal=c(0,0,1),include.mean =
F,lambda=l)

a1 <-Arima(flightrain,order=c(0,1,1),seasonal=c(0,1,1),lambda=l)
a2 <- Arima(flightrain,order=c(0,1,1),seasonal=c(0,1,1))
f1<- forecast(a1,h=length(flightest))
f2<- forecast(a2,h=length(flightest))
accuracy(f1,flightest)
accuracy(f2,flightest)
```

~~~~
auto.arima(dftd2,xreg=fourier(dftd2,K=1),seasonal=F)

atest<-Arima(flightrain, order=c(1,1,0),lambda=l,
xreg=fourier(flightrain,K=1))
atest
ftest<-
forecast(atest,h=length(flightest),xreg=fourier(flightest,K=1))

ftest %>% autoplot()

accuracy(ftest,flightest)
autoplot(dft)+
  autolayer(flightfcst,PI=F)+
  autolayer(ftest,PI=F)

accuracy(flightfcst,flightest)
checkresiduals(flightfcst)
shapiro.test(fcast11$residuals)
```
```{r}

flightarima <- Arima(flightrain, order=c(0,1,1),
seasonal=c(0,1,1),lambda=l)
flightaforecast <- forecast(flightarima, h=length(flightest))

autoplot(dft)+
  autolayer(flightaforecast, PI=F)+
  ggtitle("ARIMA Forecast on the Test")+
  ylab("Total Passengers")

accuracy(flightaforecast, flightest)
a2<-flightaforecast$residuals


checkresiduals(flightaforecast$residuals)
shapiro.test(flightaforecast$residuals)
Box.test(flightaforecast$residuals, lag = 34)


```
~~~~

Adjusting window for optimized forecast

~~~~
```{r}
~~~~

````
checkresiduals(flightforecast2)


shapiro.test(flightforecast2$residuals)

```



```{r}

airport2020 <- window(dft, start=c(2014,2))
pred2020 <- Arima(airport2020,order=c(0,1,1), seasonal=c(0,1,1),
lambda = l)
pred2020f <- forecast(pred2020, h = 14)

pred2020f$mean

sum2020 <- 0
sum2020l <- 0
sum2020u <- 0

for (i in 3:14){
  sum2020 <- sum2020 + pred2020f$mean[i]
  sum2020l <- sum2020l + pred2020f$lower[i,2]
  sum2020u <- sum2020u + pred2020f$upper[i,2]
}

sum2020
sum2020l
sum2020u



autoplot(dft)+
  autolayer(airport2020)+
  autolayer(pred2020f, colour='blue',levels("0%"))+
  ggtitle('ABIA 2020 Passenger Prediction with 2040 Cap')+
  ylab('Total Passengers')+
  geom_hline(yintercept=920000, color =
'green',series='Original')+
  geom_hline(yintercept=1250000, color =
'purple',series='Current')+
  geom_hline(yintercept=2500000, color = 'orange',series='2040
Cap')
````