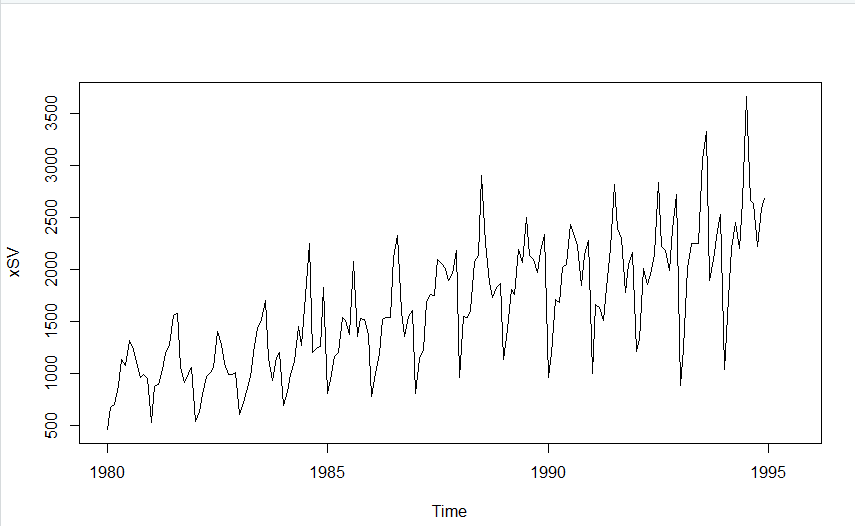
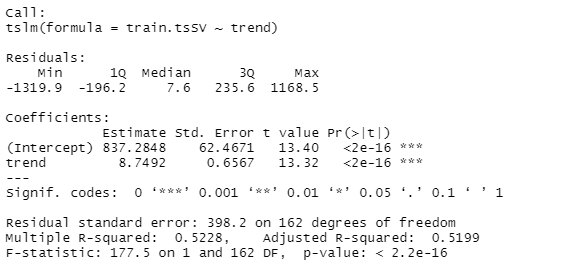
**Time\_Series\_Analytics**

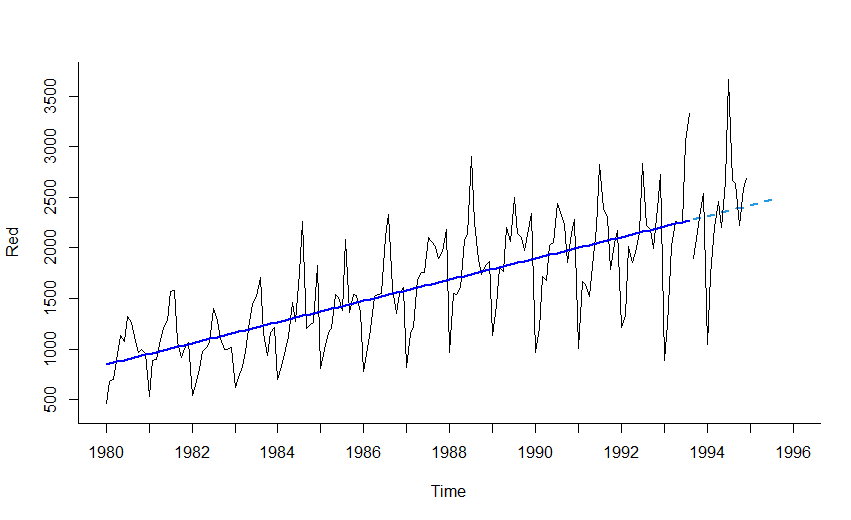
1. Please show the plot of the entire dataset.



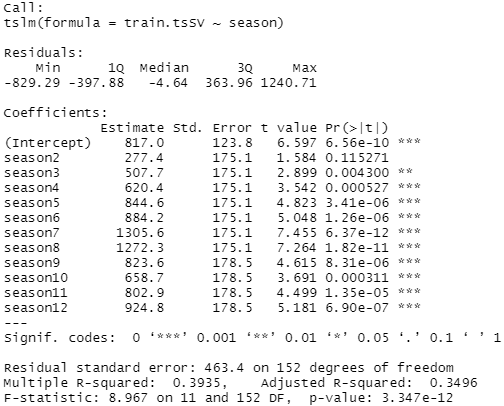
1. Please provide results for each model estimation
   * Linear trend



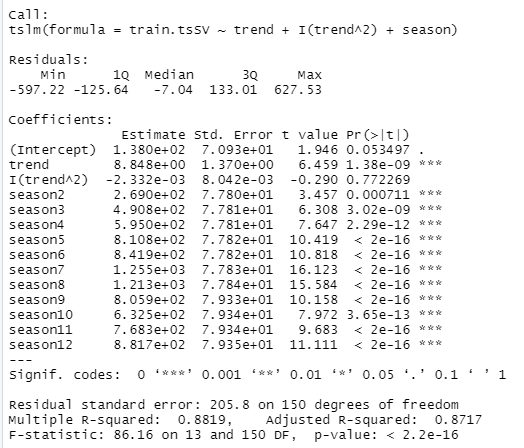
Linear Trend model:



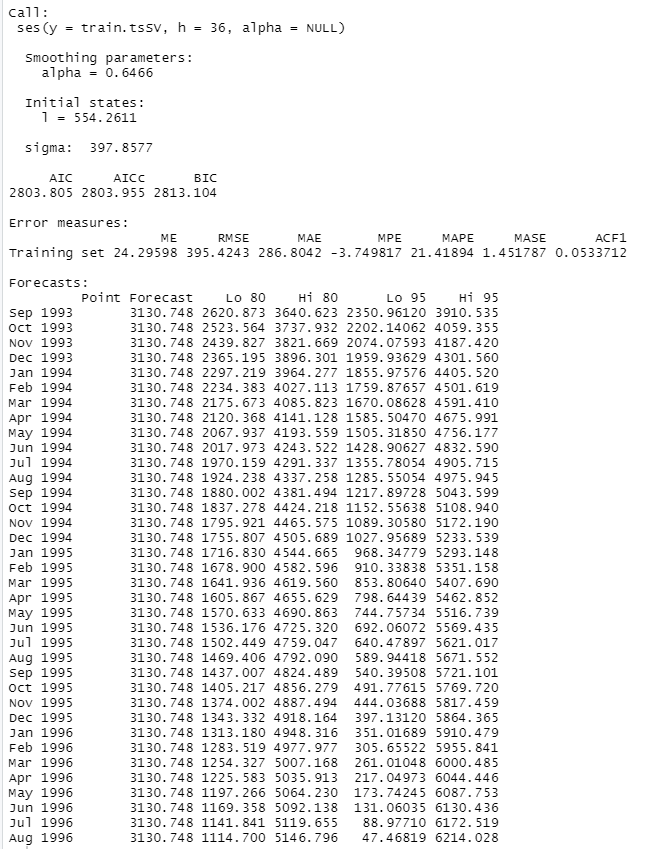
* + Seasonality



* + Linear trend and seasonality



* + Simple exponential smoothing model



1. Which model is the best forecasting model? Please provide your evidence.
   * Linear trend



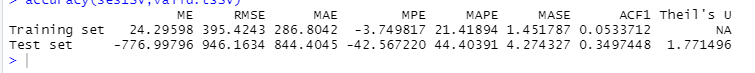
* + Seasonality



* + Linear trend and seasonality



* + Simple exponential smoothing model



From the accuracy values of the models we can infer that, Linear trend and seasonality model is better when compared to others as it has less MAPE values.

1. R-Code:

library(forecast)

#Read Data

dataSV <- read.csv("AustralianWines.csv")

str(dataSV)

head(dataSV)

# start: the time of the first observation

# frequency: number of times per year

xSV <- ts(dataSV$Red, start=c(1980,1),frequency = 12)

xSV

plot(xSV)

# Model 1: Linear Trend Model

AWines.lmSV <- tslm(xSV~trend)

summary(AWines.lmSV)

# Data partition for time series data

# Use the last 36 months data as the training dataset

nValidSV <- 24

nTrainSV <- length(xSV)-nValidSV

train.tsSV <- window(xSV,start=c(1980,1),end=c(1980,nTrainSV))

valid.tsSV <- window(xSV,start=c(1980,nTrainSV+1),end=c(1980,nTrainSV+nValidSV))

train.lmSV <- tslm(train.tsSV~trend)

summary(train.lmSV)

train.lm.predSV <- forecast(train.lmSV,h=nValidSV,level=0)

# Visualize the linear trend model

par(mfrow = c(1, 1))

plot(train.lm.predSV, ylim = c(400, 3700), ylab = "Red", xlab = "Time",

bty = "l", xaxt = "n", xlim = c(1980,1996),main = "", flty = 2)

axis(1, at = seq(1980,1996, 1), labels = format(seq(1980,1996)))

lines(train.lm.predSV$fitted, lwd = 2, col = "blue")

lines(valid.tsSV)

# Evaluate model performance

accuracy(train.lm.predSV,valid.tsSV)

# A model with seasonality

# In R, function tslm() uses ts() which automatically creates the categorical Season column (called season)

# and converts it into dummy variables.

train.lm.seasonSV <- tslm(train.tsSV ~ season)

summary(train.lm.seasonSV)

train.lm.season.predSV <- forecast(train.lm.seasonSV, h = nValidSV, level = 0)

accuracy(train.lm.season.predSV,valid.tsSV)

# A model with trend and seasonality

train.lm.trend.seasonSV <- tslm(train.tsSV ~ trend + I(trend^2)+ season)

summary(train.lm.trend.seasonSV)

train.lm.trend.season.predSV <- forecast(train.lm.trend.seasonSV, h = nValidSV, level = 0)

accuracy(train.lm.trend.season.predSV,valid.tsSV)

# run simple exponential smoothing

# and alpha = 0.2 to fit simple exponential smoothing.

sesSV <- ses(train.tsSV, alpha = 0.2, h=24)

autoplot(sesSV)

accuracy(sesSV,valid.tsSV)

# Use ses function to estimate alpha

ses1SV <- ses(train.tsSV, alpha =NULL, h=24)

summary(ses1SV)

accuracy(ses1SV,valid.tsSV)