Forcasting Analysis Individual Assignment

Manish Tripathi 12010079

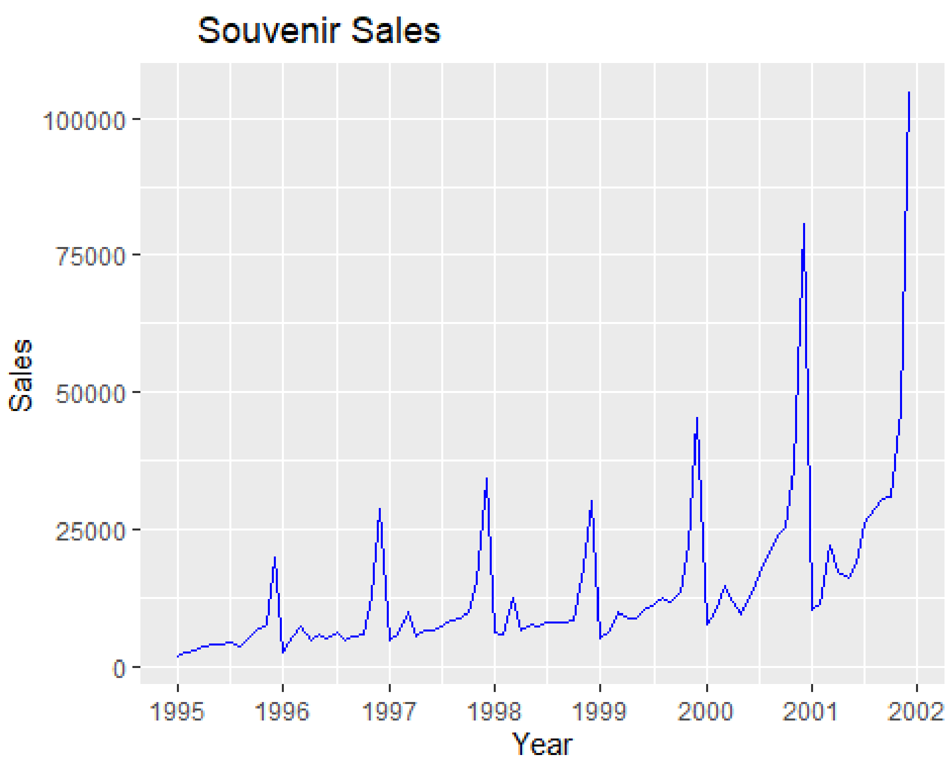
2021-06-08

```r
# Importing the libraries

library(readxl)
library(ggpubr)

## Warning: package 'ggpubr' was built under R version 3.6.3

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.6.3

library(forecast)

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

##
## Attaching package: 'forecast'

## The following object is masked from 'package:ggpubr':
##
##     gghistogram

# Importing the dataset and converting it into a time series

SouvenirSales <-
read_excel("C:/Users/PankhuriManish/Desktop/FA/SouvenirSales.xlsx",
                            col_types = c("date", "numeric"))

SouvenirSales.ts <- ts(SouvenirSales$Sales, start = c(1995,1), frequency =
12)
SouvenirSales.ts

##            Jan      Feb      Mar      Apr      May      Jun      Jul
## 1995   1664.81  2397.53  2840.71  3547.29  3752.96  3714.74  4349.61
## 1996   2499.81  5198.24  7225.14  4806.03  5900.88  4951.34  6179.12
## 1997   4717.02  5702.63  9957.58  5304.78  6492.43  6630.80  7349.62
## 1998   5921.10  5814.58 12421.25  6369.77  7609.12  7224.75  8121.22
## 1999   4826.64  6470.23  9638.77  8821.17  8722.37 10209.48 11276.55
## 2000   7615.03  9849.69 14558.40 11587.33  9332.56 13082.09 16732.78
## 2001  10243.24 11266.88 21826.84 17357.33 15997.79 18601.53 26155.15
##            Aug      Sep      Oct      Nov      Dec
```

```
## 1995   3566.34   5021.82   6423.48   7600.60  19756.21
## 1996   4752.15   5496.43   5835.10  12600.08  28541.72
## 1997   8176.62   8573.17   9690.50  15151.84  34061.01
## 1998   7979.25   8093.06   8476.70  17914.66  30114.41
## 1999  12552.22  11637.39  13606.89  21822.11  45060.69
## 2000  19888.61  23933.38  25391.35  36024.80  80721.71
## 2001  28586.52  30505.41  30821.33  46634.38 104660.67
```

a. Plot the time series of the original data. Which time series components appear from the plot.

```r
# Visualizing the data

autoplot(SouvenirSales.ts, color = "blue") +   ylab("Sales") +
  xlab("Year") + ggtitle("Souvenir Sales") +
  theme(plot.title = element_text(hjust = 0.1)) +
  scale_x_continuous(breaks = seq(1995, 2002))

## Scale for 'x' is already present. Adding another scale for 'x', which will
## replace the existing scale.
```



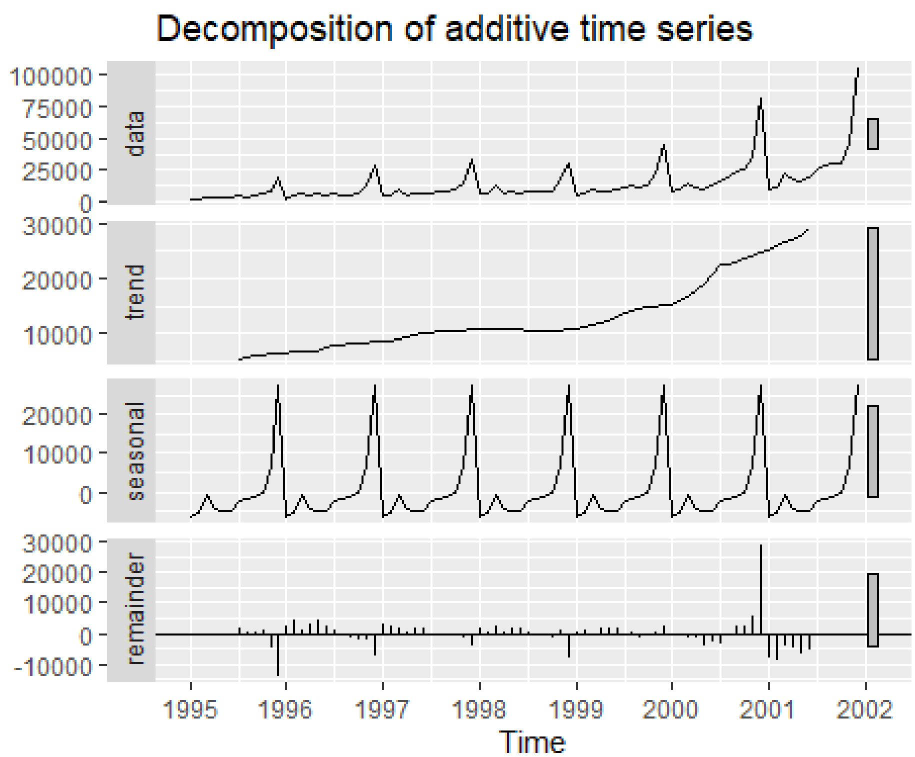
Based on the time plot the Souvenir Sales data seems to have the following components:

i) Level: All time series data have level by default
ii) Trend: There seems to be an increasing trend
iii) Seasonality: the observations towards the end of the year show a repeatitive pattern with sparp spike, suggesting presence of seasonality.

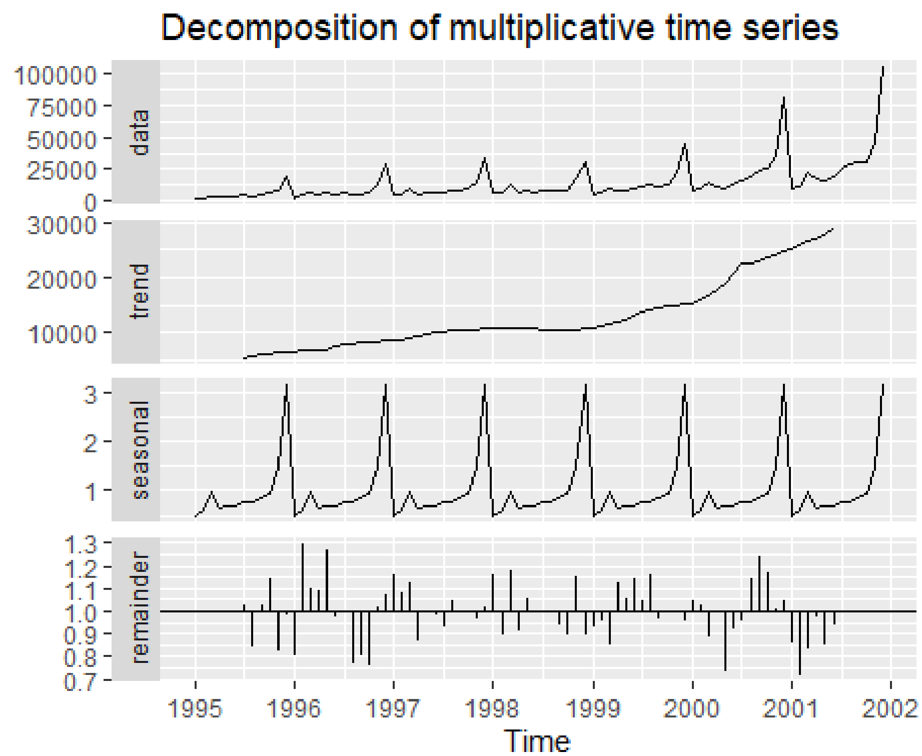
Also the seasonality component seems to be increasing by some factor so seems like a multiplicative time series with trend and seasonality.

To better understand the components present in the time series, decomposing the time series using both Additive and Multiplicative Decomposition methods.

```
# Additive Decomposition of the time series

ss1 <- decompose(SouvenirSales.ts, type= "additive")
autoplot(ss1)
```




```
# Multiplicative Decomposition of the time series
ss2 <- decompose(SouvenirSales.ts, type= "multiplicative")
autoplot(ss2)
```



To see which decomposition fits the data better, we will calculate the Root mean squared errors of both decompositions.

```
# Estimating RMSE of Additivite Decomposition
sqrt(mean(na.omit(ss1$random)^2))

## [1] 4828.972

# Estimating RMSE of Multiplicative Decomposition
sqrt(mean(na.omit(ss2$random)^2))

## [1] 1.001358
```

based on the RMSE, Multiplicative Decomposition of data seems to give better results.

b. Fit a linear trend model with additive seasonality (Model A) and exponential trend model with multiplicative seasonality (Model B). Consider January as the reference group for each model. Produce the regression coefficients and the validation set errors. Remember to fit only the training period.

```
# Splitting the data into train and test

train <- window(SouvenirSales.ts,end=c(2000,12), frequency=12)
test <- window(SouvenirSales.ts,start=c(2001,1), frequency=12)
```

```r
# Building Linear Trend with Additive Seasonality
ModelA <- tslm(train ~ trend + season)
summary(ModelA)
```

```
##
## Call:
## tslm(formula = train ~ trend + season)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -12592  -2359   -411   1940  33651
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3065.55    2640.26  -1.161  0.25029
## trend         245.36      34.08   7.199 1.24e-09 ***
## season2      1119.38    3422.06   0.327  0.74474
## season3      4408.84    3422.56   1.288  0.20272
## season4      1462.57    3423.41   0.427  0.67077
## season5      1446.19    3424.60   0.422  0.67434
## season6      1867.98    3426.13   0.545  0.58766
## season7      2988.56    3427.99   0.872  0.38684
## season8      3227.58    3430.19   0.941  0.35058
## season9      3955.56    3432.73   1.152  0.25384
## season10     4821.66    3435.61   1.403  0.16573
## season11    11524.64    3438.82   3.351  0.00141 **
## season12    32469.55    3442.36   9.432 2.19e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5927 on 59 degrees of freedom
## Multiple R-squared:  0.7903, Adjusted R-squared:  0.7476
## F-statistic: 18.53 on 12 and 59 DF,  p-value: 9.435e-16
```

```r
ModelA
```

```
##
## Call:
## tslm(formula = train ~ trend + season)
##
## Coefficients:
## (Intercept)        trend      season2      season3      season4
season5
##     -3065.6        245.4       1119.4       4408.8       1462.6
1446.2
##     season6      season7      season8      season9     season10
season11
##      1868.0       2988.6       3227.6       3955.6       4821.7
11524.6
```

```
##    season12
##     32469.6
```

Predictions for Model A

```
ModelA.pred <- forecast(ModelA, h=length(test), level =0)
ModelA.pred
```

```
##          Point Forecast     Lo 0     Hi 0
## Jan 2001       14846.03 14846.03 14846.03
## Feb 2001       16210.78 16210.78 16210.78
## Mar 2001       19745.60 19745.60 19745.60
## Apr 2001       17044.69 17044.69 17044.69
## May 2001       17273.68 17273.68 17273.68
## Jun 2001       17940.83 17940.83 17940.83
## Jul 2001       19306.78 19306.78 19306.78
## Aug 2001       19791.16 19791.16 19791.16
## Sep 2001       20764.50 20764.50 20764.50
## Oct 2001       21875.97 21875.97 21875.97
## Nov 2001       28824.31 28824.31 28824.31
## Dec 2001       50014.59 50014.59 50014.59
```

Errors for Linear Trend with Additive Seasonality (Model A)

```
accuracy(ModelA.pred$mean,test)
```

```
##                ME     RMSE      MAE      MPE     MAPE      ACF1 Theil's U
## Test set 8251.513 17451.55 10055.28 10.53397 26.66568 0.3206228 0.9075924
```

Building Exponential Trend with Multiplicative Seasonality

```
# Building Exponential Trend with Multiplicative Seasonality
ModelB <- tslm(train ~ trend + season, lambda = 0)
summary(ModelB)
```

```
##
## Call:
## tslm(formula = train ~ trend + season, lambda = 0)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4529 -0.1163  0.0001  0.1005  0.3438
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 7.646363   0.084120  90.898  < 2e-16 ***
## trend       0.021120   0.001086  19.449  < 2e-16 ***
## season2     0.282015   0.109028   2.587 0.012178 *
## season3     0.694998   0.109044   6.374 3.08e-08 ***
## season4     0.373873   0.109071   3.428 0.001115 **
## season5     0.421710   0.109109   3.865 0.000279 ***
## season6     0.447046   0.109158   4.095 0.000130 ***
```

```
## season7     0.583380   0.109217   5.341 1.55e-06 ***
## season8     0.546897   0.109287   5.004 5.37e-06 ***
## season9     0.635565   0.109368   5.811 2.65e-07 ***
## season10    0.729490   0.109460   6.664 9.98e-09 ***
## season11    1.200954   0.109562  10.961 7.38e-16 ***
## season12    1.952202   0.109675  17.800  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1888 on 59 degrees of freedom
## Multiple R-squared:  0.9424, Adjusted R-squared:  0.9306
## F-statistic:  80.4 on 12 and 59 DF,  p-value: < 2.2e-16
```

```
ModelB
```

```
##
## Call:
## tslm(formula = train ~ trend + season, lambda = 0)
##
## Coefficients:
## (Intercept)        trend      season2      season3      season4
season5
##     7.64636      0.02112      0.28201      0.69500      0.37387
0.42171
##     season6      season7      season8      season9     season10
season11
##     0.44705      0.58338      0.54690      0.63557      0.72949
1.20095
##    season12
##     1.95220
```

Predictions for Model B

```
ModelB.pred <- forecast(ModelB, h=length(test), level =0)
ModelB.pred
```

```
##          Point Forecast      Lo 0      Hi 0
## Jan 2001       9780.022  9780.022  9780.022
## Feb 2001      13243.095 13243.095 13243.095
## Mar 2001      20441.749 20441.749 20441.749
## Apr 2001      15143.541 15143.541 15143.541
## May 2001      16224.628 16224.628 16224.628
## Jun 2001      16996.137 16996.137 16996.137
## Jul 2001      19894.424 19894.424 19894.424
## Aug 2001      19591.112 19591.112 19591.112
## Sep 2001      21864.492 21864.492 21864.492
## Oct 2001      24530.299 24530.299 24530.299
## Nov 2001      40144.775 40144.775 40144.775
## Dec 2001      86908.868 86908.868 86908.868
```

Errors for Exponential Trend with Multiplicative Seasonality (Model B)

```
accuracy(ModelB.pred$mean,test)

##                ME     RMSE      MAE      MPE    MAPE      ACF1 Theil's U
## Test set 4824.494 7101.444 5191.669 12.35943 15.5191 0.4245018 0.4610253
```

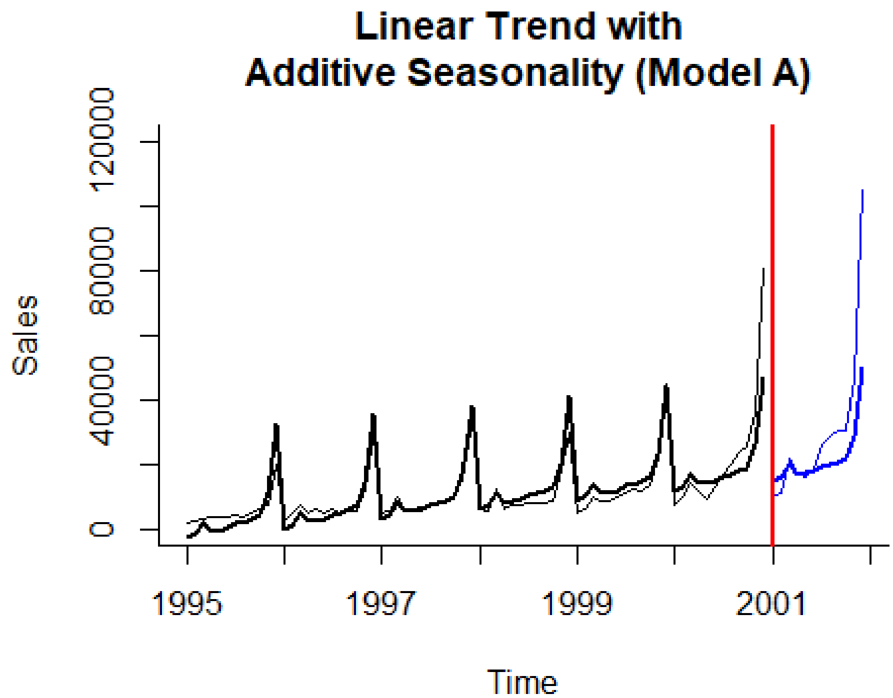
c. Which model is the best model considering RMSE as the metric? Could you have understood this from the line chart? Explain. Produce the plot showing the forecasts from both models along with actual data. In a separate plot, present the residuals from both models (consider only the validation set residuals).

Model B i.e. Exponential Trend with Multiplicative Seasonality is a better model considering the RMSE.

Based on solely the line chart also we could see a multiplicative effect for seasonality as the magnitude of susequent spike was much greate than the previous spike, but could not have commented on the exponential trend.

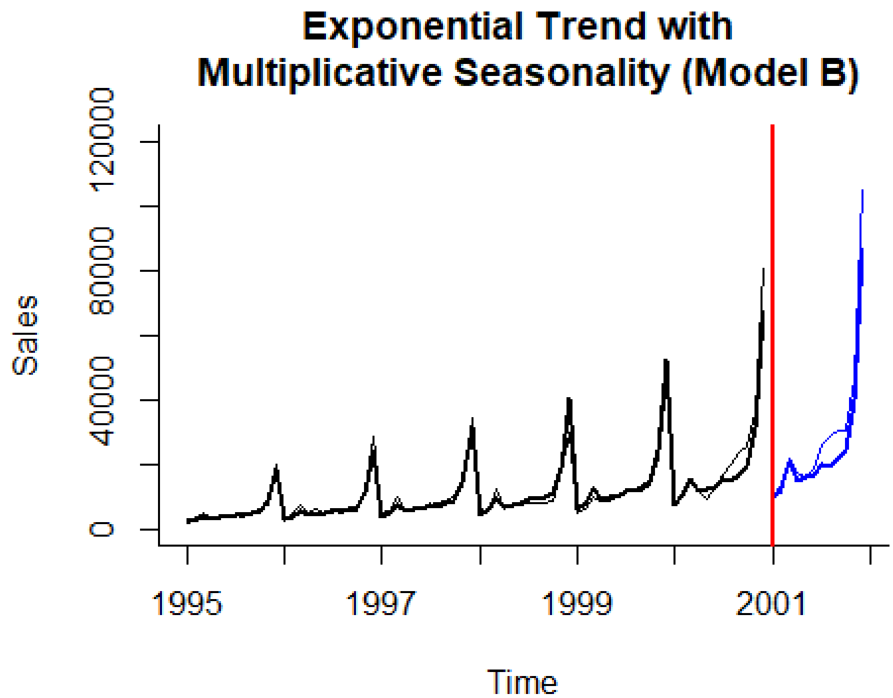
Plot showing the forecast for Linear Trend with Additive Seasonality (Model A)

```
plot(ModelA.pred , xlab ="Time", ylab= "Sales",
     main="Linear Trend with \n Additive Seasonality (Model A)",
     flty=1,  bty="l",ylim= c(0,120000))

lines(ModelA.pred$fitted,lwd=2)
lines(test,col="blue")
abline(v=2001, col="red", lwd=2)
```



Plot showing the forecast for Exponential Trend with Multiplicative Seasonality (Model B)

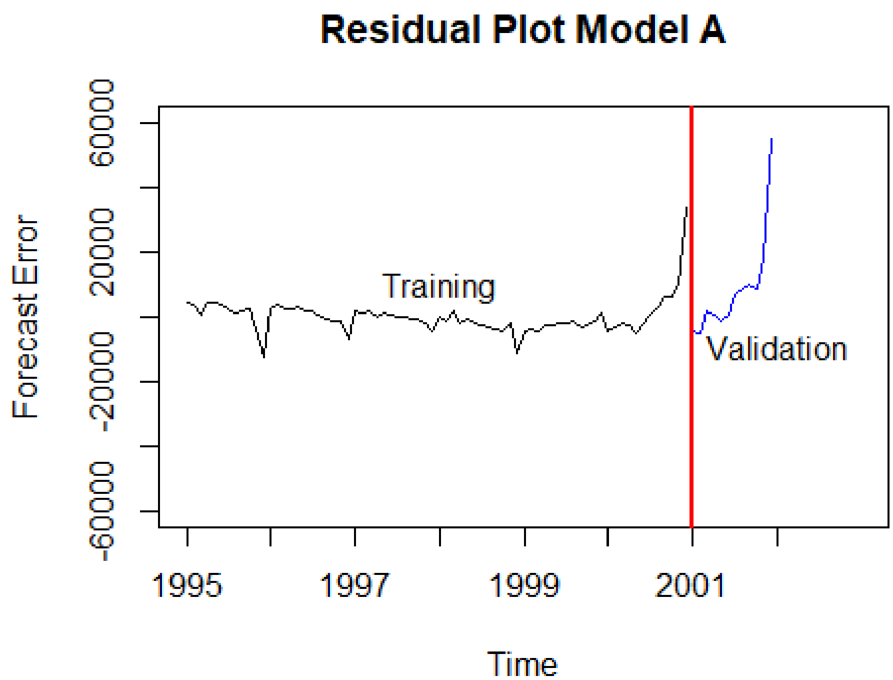
```
plot(ModelB.pred,  xlab ="Time", ylab= "Sales",
     main="Exponential Trend with \n Multiplicative Seasonality (Model B)",
     flty=1,ylim=c(0,120000), bty="l" )
lines(ModelB.pred$fitted,lwd=2)
lines(test,col="blue")
lines(train)
abline(v=2001, col="red", lwd=2)
```



Based on the two plots we can see that the Model B is better at predicting the data.

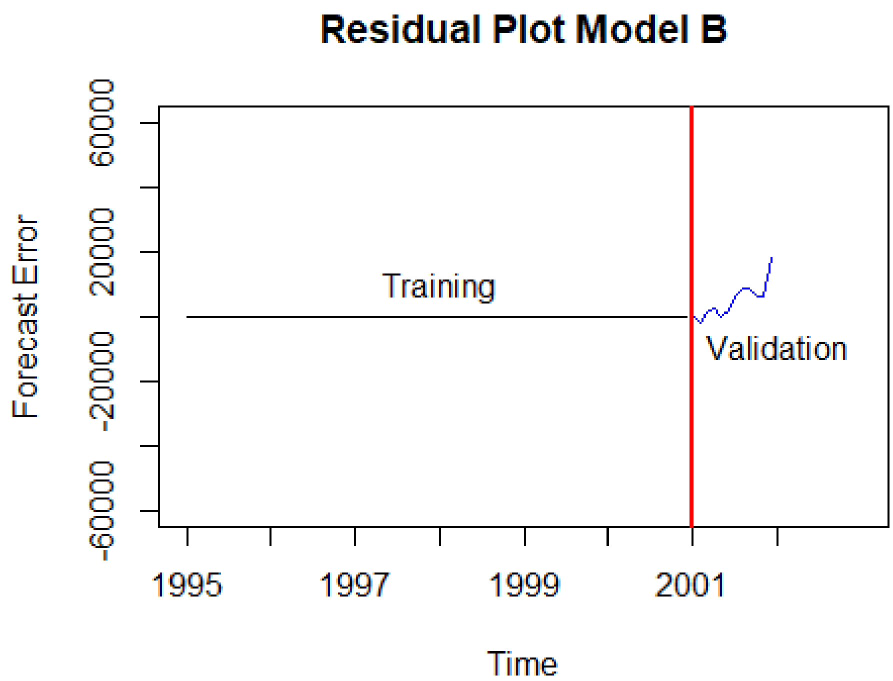
Residuals from Model A (Both Train and Test)

```
plot(ModelA$residuals, ylim= c(-60000,60000),
     main= "Residual Plot Model A", ylab="Forecast Error",
     xlim= c(1995, 2003), xaxp = c(1995, 2002, 2002-1995))
lines(test-ModelA.pred$mean, col="blue")
abline(v=2001, col="red", lwd=2)
text(1998,1, "Training",pos = 3)
text(2002,1, "Validation",pos= 1)
```



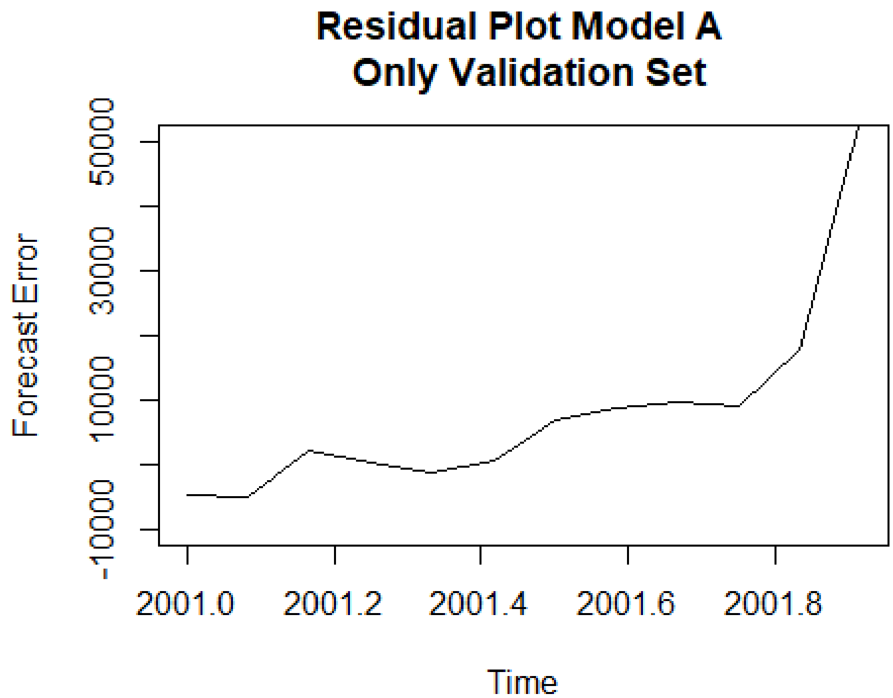
Residuals from Model B (Both Train and Test)

```
plot(ModelB$residuals, ylim= c(-60000,60000),
     main= "Residual Plot Model B", ylab="Forecast Error",
     xlim= c(1995, 2003), xaxp = c(1995, 2002, 2002-1995))
lines(test-ModelB.pred$mean, col="blue")
abline(v=2001, col="red", lwd=2)
text(1998,1, "Training",pos = 3)
text(2002,12, "Validation",pos= 1)
```



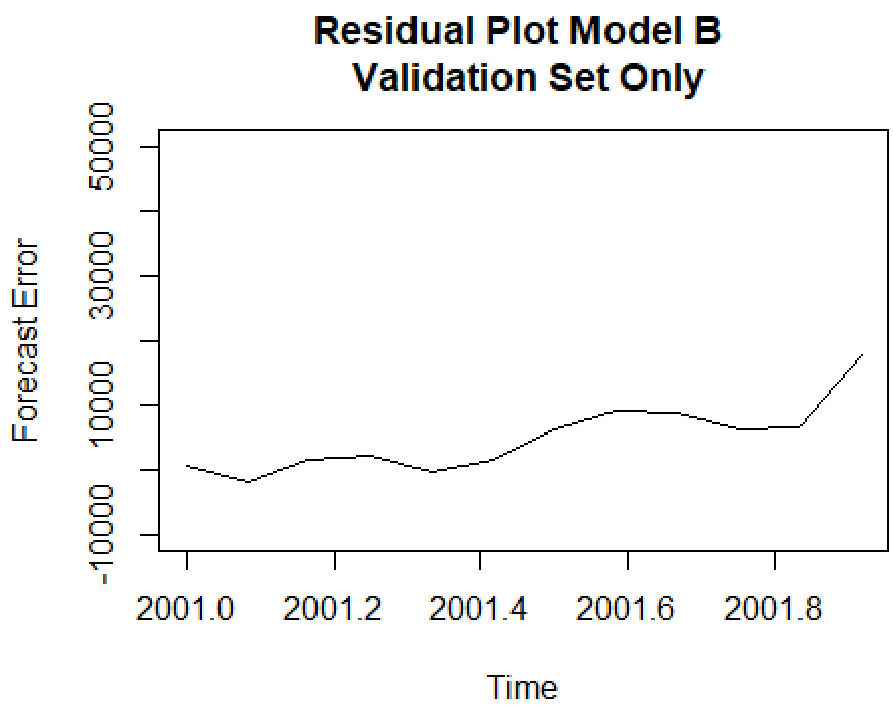
Residuals from Model A (Only Test/Validation Set)

```
plot(test-ModelA.pred$mean, ylim= c(-10000,50000),
     main= "Residual Plot Model A \n Only Validation Set",
     ylab="Forecast Error")
```



Residuals from Model B (Only Test/Validation Set)

```
plot(test-ModelB.pred$mean, ylim= c(-10000,50000),
     main= "Residual Plot Model B \n Validation Set Only",
ylab="Forecast Error")
```



d. Examine the additive model. Which month has the highest average sales during the year. What does the estimated trend coefficient in the model A mean?

December has the highest average sales during the year. The estimated trend coefficient in the model A means that for each unit increase in month, the sales increase by an amount of USD 245.4.

e. Examine the multiplicative model. What does the coefficient of October mean? What does the estimated trend coefficient in the model B mean?

The coefficient of October means that sales in October of any year are 72.9% higher than the sales in January of that particular year, as the base month here is January. The estimated trend coefficient in the model B means that for each unit increase in month, the sales increase by 2.1%.

f. Use the best model type from part (c) to forecast the sales in January 2002. Think carefully which data to use for model fitting in this case.

As the RMSE for Exponential Trend with Multiplicative Seasonality (Model B) is lesser than that of Linear Trend with Additive Seasonality (Model A). We will select Model B for Prediction. As we have selected our model we will retrain the model on the entire dataset.

```
# Building Exponential Trend with Multiplicative Seasonality
ModelB.retrained <- tslm(SouvenirSales.ts ~ trend + season, lambda = 0)
summary(ModelB.retrained)

##
## Call:
## tslm(formula = SouvenirSales.ts ~ trend + season, lambda = 0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.41644 -0.12619  0.00608  0.11389  0.38567
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 7.6058604  0.0768740  98.939  < 2e-16 ***
## trend       0.0223930  0.0008448  26.508  < 2e-16 ***
## season2     0.2510437  0.0993278   2.527 0.013718 *
## season3     0.6952066  0.0993386   6.998 1.18e-09 ***
## season4     0.3829341  0.0993565   3.854 0.000252 ***
## season5     0.4079944  0.0993817   4.105 0.000106 ***
## season6     0.4469625  0.0994140   4.496 2.63e-05 ***
## season7     0.6082156  0.0994534   6.116 4.69e-08 ***
## season8     0.5853524  0.0995001   5.883 1.21e-07 ***
## season9     0.6663446  0.0995538   6.693 4.27e-09 ***
## season10    0.7440336  0.0996148   7.469 1.61e-10 ***
## season11    1.2030164  0.0996828  12.068  < 2e-16 ***
## season12    1.9581366  0.0997579  19.629  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1858 on 71 degrees of freedom
## Multiple R-squared:  0.9527, Adjusted R-squared:  0.9447
## F-statistic: 119.1 on 12 and 71 DF,  p-value: < 2.2e-16

ModelB.retrained

##
## Call:
## tslm(formula = SouvenirSales.ts ~ trend + season, lambda = 0)
##
## Coefficients:
```

```
## (Intercept)       trend     season2     season3     season4
season5
##     7.60586     0.02239     0.25104     0.69521     0.38293
0.40799
##     season6     season7     season8     season9    season10
season11
##     0.44696     0.60822     0.58535     0.66634     0.74403
1.20302
##    season12
##     1.95814
```

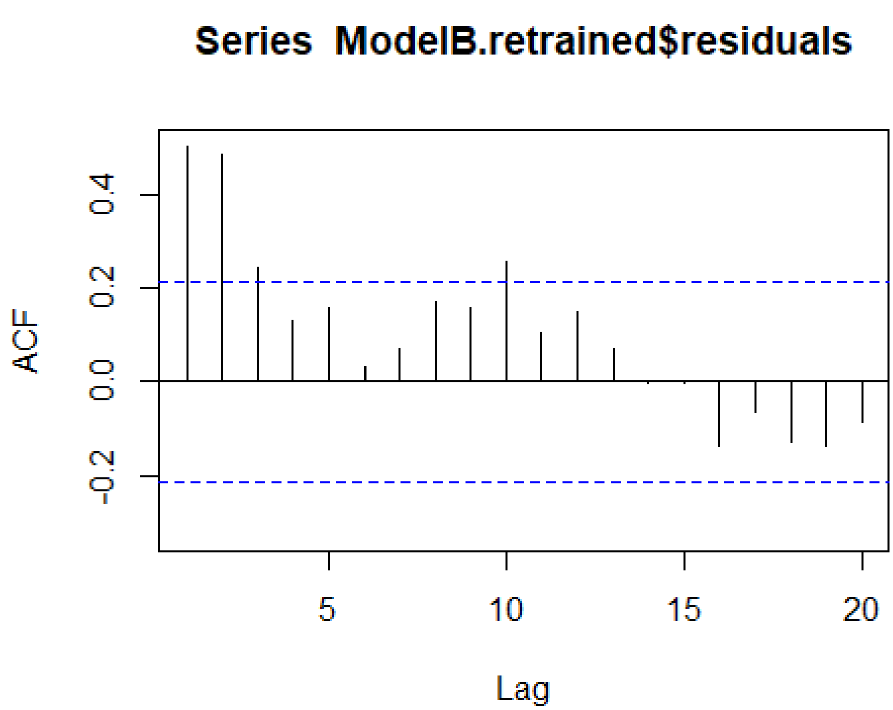
Forcasting for January 2002

```
ModelB.retrained.pred <- forecast(ModelB.retrained, h=1, level =95)
ModelB.retrained.pred

##          Point Forecast    Lo 95    Hi 95
## Jan 2002       13484.06 9000.202 20201.76
```

g. Plot the ACF and PACF plot until lag 20 of the residuals obtained from training set of the best model chosen. Comment on these plots and think what AR(p) model could be a good choice?

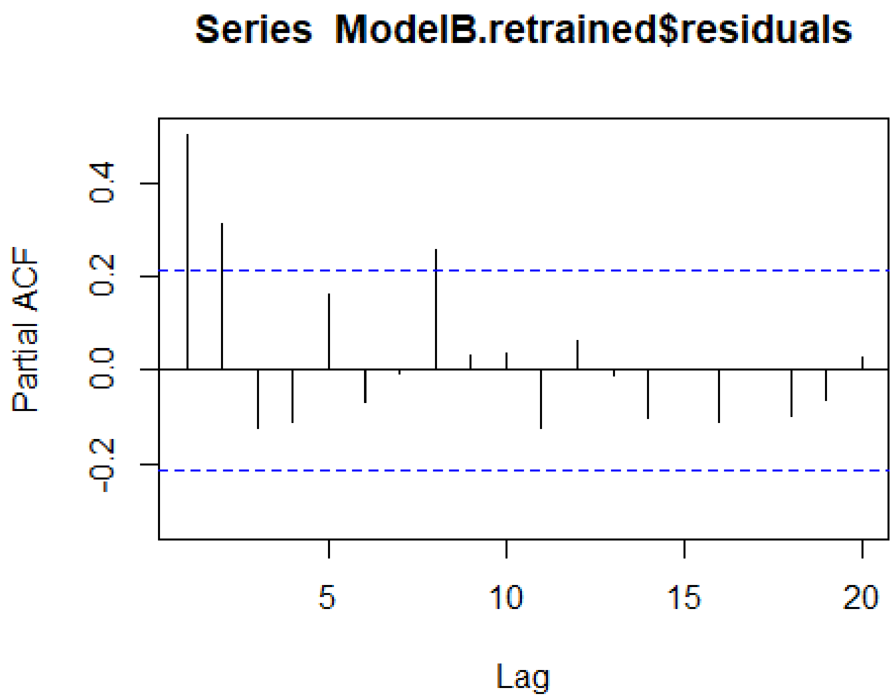
ACF Plot

```
Acf(ModelB.retrained$residuals,lag.max = 20)
```



PACF Plot

```
Pacf(ModelB.retrained$residuals,lag.max = 20)
```



Based on the ACF and the PACF plots the AR (2) model could be a good choice as the ACF plot has very significant lag 1 and lag 2 and significant lag 3 bar. Also we have a decreasing pattern which is sinosuidal.

Even The PACF plot has very significant lag 1 and lag 2 significantly outside the white noise boundary, then the rest of lags are insignificant.

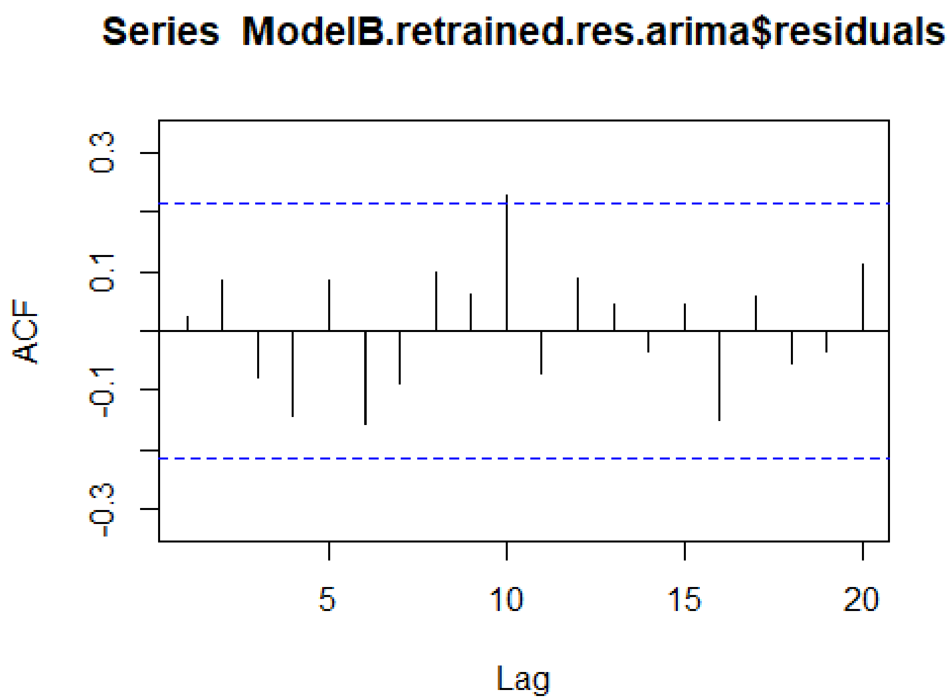
THe ACF and the PACF plots together suggest a AR(2) model.

h. Fit an AR(p) model as you think appropriate from part (h) to the training set residuals and produce the regression coefficients. Was your intuition at part (h) correct?

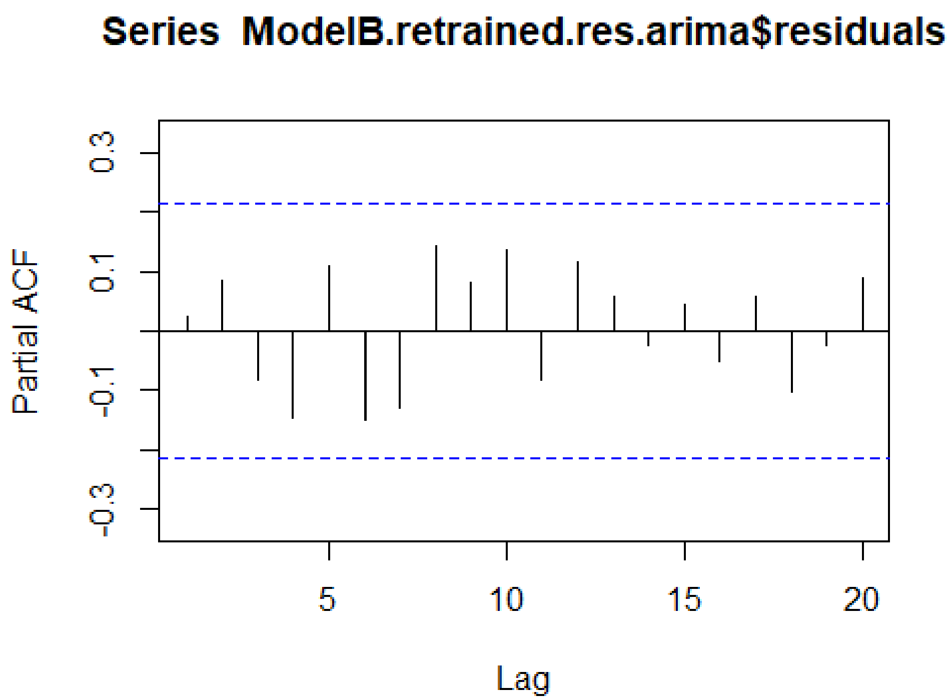
```
errors = ModelB.retrained$residuals
ModelB.retrained.res.arima <- Arima(errors, order = c(2,0,0))
summary(ModelB.retrained.res.arima)

## Series: errors
## ARIMA(2,0,0) with non-zero mean
##
## Coefficients:
##          ar1     ar2     mean
##       0.3488  0.3182  -0.0026
```

```
## s.e.  0.1028  0.1030   0.0441
##
## sigma^2 estimated as 0.02005:  log likelihood=46.28
## AIC=-84.56   AICc=-84.05   BIC=-74.83
##
## Training set error measures:
##                      ME      RMSE       MAE      MPE     MAPE      MASE
## Training set 0.00296632 0.1390468 0.1129372 83.92507 157.7718 0.6464461
##                    ACF1
## Training set 0.02499441
```

```
Acf(ModelB.retrained.res.arima$residuals, lag.max=20)
```




```
Pacf(ModelB.retrained.res.arima$residuals, lag.max=20)
```



Following the AR(2) model the auto correlations at lag 1 and lag 2 have become insignificant. The intution seems to be correct.

i. Now, using the best regression model and AR(p) model, forecast the sales in January 2002. Think carefully which data to use for model fitting in this case.

```
ModelB.retrained.res.arima.pred <- forecast(ModelB.retrained.res.arima, h=1,
level =95)
summary(ModelB.retrained.res.arima.pred)

##
## Forecast method: ARIMA(2,0,0) with non-zero mean
##
## Model Information:
## Series: errors
## ARIMA(2,0,0) with non-zero mean
##
## Coefficients:
##          ar1     ar2     mean
##       0.3488  0.3182  -0.0026
## s.e.  0.1028  0.1030   0.0441
##
## sigma^2 estimated as 0.02005:  log likelihood=46.28
## AIC=-84.56   AICc=-84.05   BIC=-74.83
##
```

```
## Error measures:
##                      ME      RMSE       MAE      MPE     MAPE      MASE
## Training set 0.00296632 0.1390468 0.1129372 83.92507 157.7718 0.6464461
##                    ACF1
## Training set 0.02499441
##
## Forecasts:
##          Point Forecast      Lo 95     Hi 95
## Jan 2002     0.06501293 -0.2125146 0.3425405
```

Forcast for January 2002 based on Model B and AR(2) model will be:

```
Forecast_Jan2002 = ModelB.retrained.res.arima.pred$mean +
  ModelB.retrained.pred$mean
Forecast_Jan2002

##           Jan
## 2002 13484.13
```

2. Short answer type questions:

a. Explain the key difference between cross sectional and time series data.

`Cross Sectional Data:` Observations of data collected at a given point of time. For example: a) Name of Employees, Salary credited, Tax Deducted in the month of May, b) Marks obtained by students of a particular school in the Class XII boards. The underlying assumption about the data is that it is Independently and Identically Distributed or Randomly Distributed.

`Time Series Data:` Observations are recorded over a period of time, for example, rainfall recorded over past 10 years, monthly number of tourists who visited India in past 12 months. The data in time series shows auto correlation or seriel correlation where the data in time "t" may be correlated to data from time "t-1" or previous time periods. A time series data is comprised of one or more of the components described below:

i) Level
ii) Trend
iii) Sesonality
iv) Cyclicality, and
v) Noise

b. Explain the difference between seasonality and cyclicality.

`Seasonality:` is a short term variation in time series data due to seasonal factors. The distances between the two seasonal cycles should be equal i.e the up and down pattern should repeat at regular intervals. It can be caused due to seasonal factors during certain times in year, month, week, day or hour. For example, Heavy rush of customers in mall during weekends or during certain days of year such a Christmas or New Year. There can be multiple seasonal cycles can coexist in the time series data.

There are two types of seasonality:

`i) Additive Seasonality:` is when the the values increase or decrease by a constant amount

`ii) Multiplicative Sesonality:` is when the values change by a constant degree

`Cyclicality:`Irregular pattrens in the time series data with medium term repetition. For example, The GDP data of country for past 200 years will show impacvt of multiple recessions but the recessions do not repeat after same number of years. This is Cyclicality.

c. Explain why centered moving average is not-considered suitable for forecasting.

In the centered moving average, the trend line will loose some observations at the beginning and some at the end, while in the trailing moving average all the observations lost are at the beginning. The forcasting horizon will be larger if we use the centered moving average as compared to the trailing moving average. For example if the window size is 5 and we want to forcast one period ahead, the forcasting horizon with centered moving average will be 3, whereas with trailing moving average it will be 1. We will always prefer a shorter forcasting horizon as that reduces the chances of error in forcast.

d. Explain stationarity and why is it important for some time series forecasting methods?

A time series data is called stationary if its mean, varience and covarience do not change with time. A time series with trend or seasonality is not stationary as with trend or sesanality the mean and varience may increase or decrease over a period of time. A time series which only has noise is stationary, as the observations in such a time series is randomly distributed.

If the mean, varience or covarience change over time it becomes difficult to forcast future values. Assumption of stationarity implies data is not dependant on time, for example if the sample mean and varience decrease over time we will always be over forcasting based on current values of mean and varience or vice versa. Also most forcasting models work on the assumption that the time series data is stationary. Hence making stationarity important for forcasting mentods.

e. How does an ACF plot help to identify whether a time series is stationary or not?

If the data is not stationary then the ACF plot drops to zero slowly. ACF plot for stationary data drops to zero rapidly. Also if the data is not stationary, the r1 value of the ACF plot will be usually large and positive.

f. Why partitioning time series data into training, validation, and test set is not recommended? Briefly describe two considerations for choosing the width of validation period.

In time series data the most recent observations are the most relavant and most informative as they tell us most about the current scenario, if we divide our data into train, validation and test set, we will be training our model on fairly old data, the scenario could have changed quite a lot in present. Also our forcasting horizon will be fairly large. So in time series data, instead of splitting our data into train, validation and test set, we split our data into train and validation set only. We train our model on the training set and use the validation set for model selection. Once the model is selected, we train our model again on the entire time series data to estimate the parameters of the model used for forcasting. This way we are able incorporate the effect of most recent data in our model.

The width of validation period depends upon:

`Forcasting Horizon:` The validation set width should be similar to the forcasing horizon, for example, if we are looking to forecast next twelve months of sales, the validation set period should be equal to 12, else the the forcasting horizon will fail to mimic the actual scenario. Also if the validation set width is longer, recent information will not be incorporated in our training set and our model will be deficient.

`Seasonality:` The validation set should also be equal to the seasonal cycle, else we will fail to see if the model is corretly forecasting the seasonal variations. If their mare multiple seasonal cycles, the width should be so selected so as to incorporate all the seasonal cycles.

Other things that determine the width of validation period are Forecasting goal, Data frequency and Length of the series.

g. Both smoothing and ARIMA method of forecasting can handle time series data with missing value. True/False. Explain

False. Both soothening and ARIMA can not be used if certain observations in the time series are missing. Both the methods see if there is a correlation between the present value and the past value to forcast. If the to be forcasted value is dependant of the missing value then we will not have any forcast. In such a scenario, we either impute the missing value or use models like linear or logistic regression. Kalman Filter is used as one of the ways to impute the missing values in time series data.

h. Additive and multiplicative decomposition differ in the way the trend is computed. True /False. Explain.

False. The trend is calculated the same way for both additive as well as multiplicative decompositions. Moving average with appropriate window is used to compute the trend. The moving average helps supress sesonality and noise and leaves us with trend.

However, the detrending of series is done differently in Additive and multiplicative decomposition. While in Additive decomposition the moving average is subtracted from the observations to get the detrended series. In multiplicative decomposition the observations are divided by the moving average to get the detrended series.

i. After accounting for trend and seasonality in a time series data, the analyst observes that there is still correlation left amongst the residuals of the time series. Is that a good or a bad news for the analyst? Explain.

This can be considered as good news for the analyst. If there was no correlation in the residuals, there is nothing more the analyst can do with the residuals as they will be completely random, but on the other hand if there is still correlation left in the residuals, the analyst can derieve further information from the residuals and improve the model. If the data is not autocorrelated, we can not improve the model beyond a naive forcast.