CASE STUDY#2

FORECASTING WALMART'S REVENUE

Alarmelu Pichu Mani – TJ6723

Q1. Plot the data and visualize time series components.

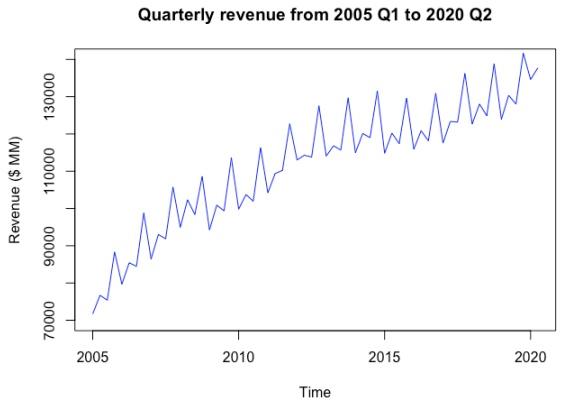
1a - Create time series data set in R using the ts() function.

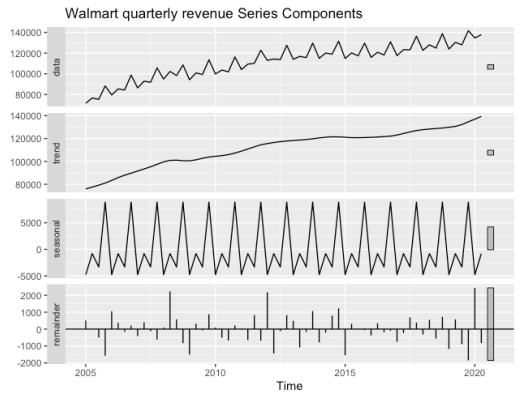
The ts () function in R creates a time series object by taking in our given historical Walmart quarterly revenue dataset (a numeric vector). The code is as follows.

```
revenue.ts <- ts (revenue.data$Revenue, start = c(2005,1), end = c(2020, 2), freq = 4)
```

where start and end are the times of the first and last observation and frequency is the number of observations per unit time (1=annual, 4=quarterly, 12=monthly).

1b- Apply the plot () function to create a data plot with the historical data, provide it in your report, and explain what time series components can be visualized in this plot.





Plotting the time series data generated by the ts function, help us visualize the trend and seasonality. By looking at the Quarterly revenue from 2005 Q1 to 2020 Q2 plot, we can see that the data has an upward trend and seasonality present. The revenue goes up in Q2, sees a dip in Q3 and eventually peaking in Q4. This behavior keeps repeating over the course of the time taken into consideration. In order to understand the components of the time series data better, we can use the autoplot function in R to plot the different components individually. The Walmart quarterly revenue series components plot helps us visualize the trend and seasonality separately. In this plot we can clearly see the seasonal behavior of the revenue data clearly, Q2 and Q4 showing peaks in revenue and a dip in Q3. The overall upward trend is also clearly visible. The remainder component is the remaining variation between the time series and the combination of trend and seasonal component. The remainder is showing periods of random variations across quarters of each year.

Q2- Apply five regression models using data partition.

Consider the following 5 regression-based models:

i. *Regression model with linear trend*
ii. *Regression mode with quadratic trend*
iii. *Regression model with seasonality*
iv. *Regression model with linear trend and seasonality*
v. *Regression model with quadratic trend and seasonality.*

2a- Develop data partition with the validation partition of 16 periods and the rest for the training partition.

A validation data partitioning of 16 periods Is created using the window function in R.

This function extracts subsets of the full revenue.ts dataset and creates training and validation data based on our requirements.

```
nValid <- 16
nTrain <- length(revenue.ts) - nValid
train.ts <- window(revenue.ts, start = c(2005, 1), end = c(2005, nTrain))
valid.ts <- window(revenue.ts, start = c(2005, nTrain + 1),
                   end = c(2005, nTrain + nValid))
```

The training and validation datasets are as follows:

```
> train.ts
       Qtr1   Qtr2   Qtr3   Qtr4
2005  71680  76697  75397  88327
2006  79676  85430  84467  98795
2007  86410  92999  91865 105749
2008  94940 102342  98345 108627
2009  94242 100876  99373 113594
2010  99811 103726 101952 116360
2011 104189 109366 110226 122728
2012 113010 114282 113800 127559
2013 114070 116830 115688 129706
2014 114960 120125 119001 131565
2015 114826 120229 117408 129667
2016 115904 120854
> valid.ts
       Qtr1   Qtr2   Qtr3   Qtr4
2016                118179 130936
2017 117542 123355 123179 136267
2018 122690 128028 124894 138793
2019 123925 130377 127991 141671
2020 134622 137742
```

Figure 1- Training and validation data set

2b- Use the tslm() function for the training partition to develop each of the 5 regression models from the above list. Apply the summary () function to identify the model structure and parameters for each regression model, show them in your report, and also present the respective model equation. Use each model to forecast revenues for the validation period using the forecast() function.

i. Regression model with linear trend:

Linear trend is used to fit a global trend that applies to the data and gets extrapolated to the forecasting. The linear trend implies that the series increase or decrease linearly over time.

The tslm() function in R helps us build a regression with linear trend based upon time and the summary() function help us understand the summary statistics of the linear trend model. The goodness for use of a model is evaluated using a combination of the coefficient of determination – R-squared, the associated probabilities and the F-statistic as given below.

```
> #i.Regression model with linear trend
> train.lin <- tslm(train.ts ~ trend)
> #summary
> summary(train.lin)

Call:
tslm(formula = train.ts ~ trend)

Residuals:
     Min       1Q   Median       3Q      Max
-11944.2  -4520.5   -553.5   2316.8  13030.8

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 81740.51    2009.22   40.68   <2e-16 ***
trend        1024.62      74.44   13.76   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6703 on 44 degrees of freedom
Multiple R-squared:  0.8115,    Adjusted R-squared:  0.8072
F-statistic: 189.5 on 1 and 44 DF,  p-value: < 2.2e-16
```

Figure 2- Summary output from R

Model equation:

$Y_t = 81740.51 + 1024.62t$

Evaluating the model:

The co-efficient of determination R-squared indicates is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model. In our case, about 81% of variation in the historical data can be explained by the independent variables of the regression model. Hence the R-squared value is statistically significant.

As for the statistical significance of the intercept and trend, the associated probability values are significantly lower than 5%. This means that the chances of the intercept and the trend values becoming 0 are extremely low (2.2e-16 actually means 2.2x10^-16, which is a very small number mathematically). This implies that the intercept and trend are statistically significant.

As for the F-statistic, the value is on the higher side along with the associated probability significantly lower than 5%. This implies that the linear model is a good fit for the data.

Forecasting using the linear trend model:

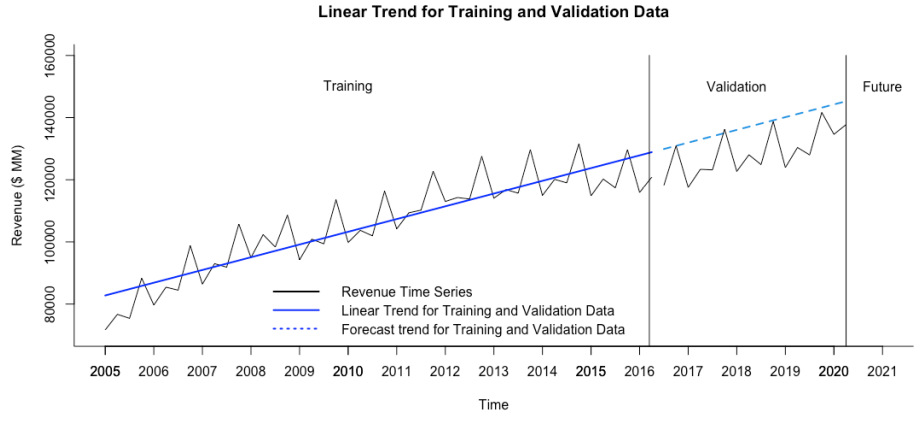
The forecast () function in R helps us forecast the validation data as follows.

train.lin.pred <- forecast(train.lin, h = nValid, level = 0)

formattable(data.frame(train.lin.pred))

	Point.Forecast	Lo.0	Hi.0
2016 Q3	129897.4	129897.4	129897.4
2016 Q4	130922.1	130922.1	130922.1
2017 Q1	131946.7	131946.7	131946.7
2017 Q2	132971.3	132971.3	132971.3
2017 Q3	133995.9	133995.9	133995.9
2017 Q4	135020.5	135020.5	135020.5
2018 Q1	136045.1	136045.1	136045.1
2018 Q2	137069.8	137069.8	137069.8
2018 Q3	138094.4	138094.4	138094.4
2018 Q4	139119.0	139119.0	139119.0
2019 Q1	140143.6	140143.6	140143.6
2019 Q2	141168.2	141168.2	141168.2
2019 Q3	142192.8	142192.8	142192.8
2019 Q4	143217.5	143217.5	143217.5
2020 Q1	144242.1	144242.1	144242.1
2020 Q2	145266.7	145266.7	145266.7

However, plotting the time series data, linear trend and the forecast for validation data help us visually understand the forecasting tendencies of the model.



Here we can see that the linear trend model is doing reasonably well in the training data but still there are places where it is skewed. The model captures the overall upward trend but fails to address the variations in the data. In the validation dataset, the model tends to overestimate the forecast when compared to the historical data in hand.

ii. Regression model with quadratic trend:

The regression model with a quadratic trend is developed as follows. As previously done, the goodness for use will be evaluated using the summary statistics of the model.

Model equation:

$Y_t = 73558.34 + 2047.39t - 21.76t^2$

```
> # ii. Regression mode with quadratic trend
> train.quad <- tslm(train.ts ~ trend + I(trend^2))
> #summary
> summary(train.quad)

Call:
tslm(formula = train.ts ~ trend + I(trend^2))

Residuals:
   Min     1Q Median     3Q    Max
 -7833  -3881  -1569   3575  10929

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 73558.345   2671.671  27.533  < 2e-16 ***
trend        2047.386    262.210   7.808 8.88e-10 ***
I(trend^2)    -21.761      5.409  -4.023 0.000228 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5779 on 43 degrees of freedom
Multiple R-squared:  0.8631,    Adjusted R-squared:  0.8567
F-statistic: 135.5 on 2 and 43 DF,  p-value: < 2.2e-16
```

Figure 3- Summary output from R

Evaluating the model:

The coefficient of determination R-squared value is at 86% and the adjusted R-squared value is at 85%, considering both the values as we have two independent variables (t and t^2). The associated p-values also show that the probabilities of the intercept, trend and trend^2 becoming 0 are less than 5% (almost 0). Hence, they are statistically significant.

As for the F-statistic, it is reasonably on the higher side with a low p-value. This implies that the overall goodness for use of the quadratic trend model for forecasting is high.

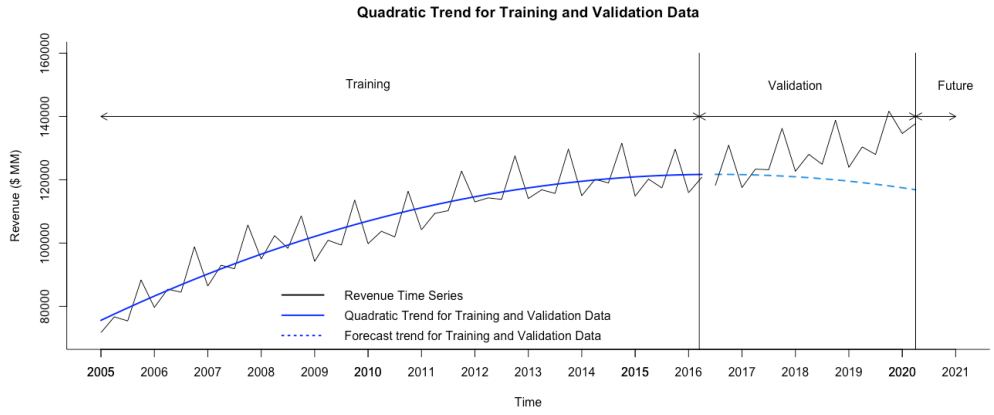
Forecasting using the quadratic trend model:

train.quad.pred <- forecast(train.quad, h = nValid, level = 0)

formattable(data.frame(train.quad.pred)).

	Point.Forecast	Lo.0	Hi.0
2016 Q3	121715.3	121715.3	121715.3
2016 Q4	121695.4	121695.4	121695.4
2017 Q1	121631.9	121631.9	121631.9
2017 Q2	121525.0	121525.0	121525.0
2017 Q3	121374.5	121374.5	121374.5
2017 Q4	121180.5	121180.5	121180.5
2018 Q1	120943.0	120943.0	120943.0
2018 Q2	120661.9	120661.9	120661.9
2018 Q3	120337.3	120337.3	120337.3
2018 Q4	119969.3	119969.3	119969.3
2019 Q1	119557.6	119557.6	119557.6
2019 Q2	119102.5	119102.5	119102.5
2019 Q3	118603.8	118603.8	118603.8
2019 Q4	118061.7	118061.7	118061.7
2020 Q1	117476.0	117476.0	117476.0
2020 Q2	116846.7	116846.7	116846.7

However, plotting the time series data, quadratic trend and the forecast for validation data help us visually understand the forecasting tendencies of the model. Here we can see that the quadratic trend model is doing well in the training data but still there are places where it is skewed to the lower side and fails to capture the variations in the data. These variations are typically the seasonality component of the data. In the validation dataset, the model tends to underestimate the forecast in comparison to the historical data.



iii. Regression model with seasonality:

This model fits the series that falls into a seasonal pattern. As we saw in the components plot (question 1b) there is a seasonality component to the time series data we are analyzing. In order to incorporate it using regression-based models, we make use of dummy variables.

When we use the tslm () function to build the regression model with just seasonality,

train. season <- tslm(train.ts ~ season)

we get the summary statistics for the model in terms of the seasons (no trend like in the earlier models).

Model equation:

$Y_t = \beta_0 + \beta_1 D_2 + \beta_3 D_3$,

Where D is an artificially created variable (dummy variable) also if we have k seasons, we use k-1 dummy variables.

D_2 = represent Q2 when 1, 0 otherwise.

D_3 = represent Q3 when 1, 0 otherwise.

D_4 = represent Q4 when 1, 0 otherwise.

When all the dummy variables are 0, it represents Q1.

```
> # iii. Regression model with seasonality
> train.season <- tslm(train.ts ~ season)
> #summary of seasonal model and associated parameters.
> summary(train.season)

Call:
tslm(formula = train.ts ~ season)

Residuals:
     Min       1Q   Median       3Q      Max
-28629.8  -9229.4     81.6  13616.6  16499.0

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   100310       4208  23.835   <2e-16 ***
season2         5003       5952   0.841   0.4053
season3         2192       6086   0.360   0.7205
season4        15388       6086   2.529   0.0153 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14580 on 42 degrees of freedom
Multiple R-squared:  0.1489,    Adjusted R-squared:  0.08812
F-statistic:  2.45 on 3 and 42 DF,  p-value: 0.07681
```

Figure 4- Summary output from R

Evaluating the model:

When we look at the R-squared value, we can see that only 14% of the variation in the time series data can be explained by this model. This is because we are considering only the seasonality in this model, which is only a part of the characteristics of the full data.

As for the probabilities associated with the seasons, we see that season 2 and 3 are not statistically significant (probability > 5%), while the other seasons are seen to be statistically significant. The F-statistic is on the lower side numerically and the associated p-value is close to 8%, which implies that may not be a good model to address all the variations and components in the dataset. This is however in line with our understanding that this model mainly captures the seasonality component alone.

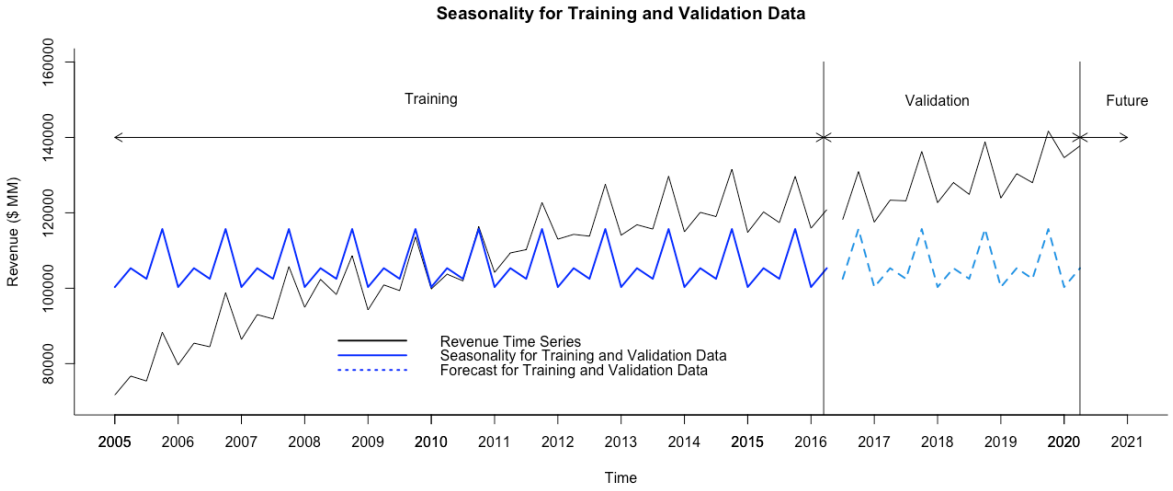
Forecasting using the seasonality model:

train.season.pred <- forecast(train.season, h = nValid, level = 0)

formattable(data.frame(train.season.pred))

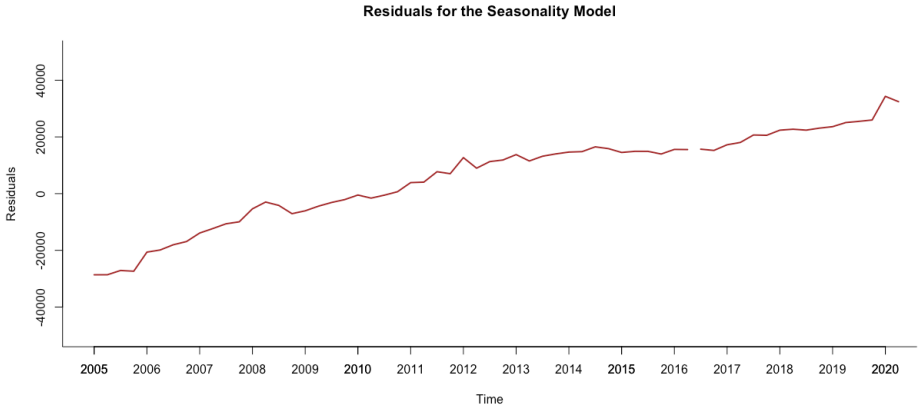
	Point.Forecast	Lo.0	Hi.0
2016 Q3	102502.0	102502.0	102502.0
2016 Q4	115697.9	115697.9	115697.9
2017 Q1	100309.8	100309.8	100309.8
2017 Q2	105313.0	105313.0	105313.0
2017 Q3	102502.0	102502.0	102502.0
2017 Q4	115697.9	115697.9	115697.9
2018 Q1	100309.8	100309.8	100309.8
2018 Q2	105313.0	105313.0	105313.0
2018 Q3	102502.0	102502.0	102502.0
2018 Q4	115697.9	115697.9	115697.9
2019 Q1	100309.8	100309.8	100309.8
2019 Q2	105313.0	105313.0	105313.0
2019 Q3	102502.0	102502.0	102502.0
2019 Q4	115697.9	115697.9	115697.9
2020 Q1	100309.8	100309.8	100309.8
2020 Q2	105313.0	105313.0	105313.0

However, plotting the time series data, seasonality and the forecast for validation data help us visually understand the forecasting tendencies of the model.



Here we can clearly see that the model captures only the seasonality, as the plot for the seasonality for training data is horizontal and does not show any upward or downward trend as seen in the historical data. Hence during the validation, the model misses out on the trend and underestimates the forecast.

When we plot the residuals of the regression model with just seasonality, we can see that the trend is visible.



Though this model efficiently captures the seasonality, it misses out on the trend. This calls for a model that has the ability to capture a combination of trend and seasonality.

iv. Regression model with linear trend and seasonality

We build the regression model with linear trend and seasonality using the tslm () function as follows.

train.Lintrend.season <- tslm(train.ts ~ trend + season)

We get the summary statistics of the model as follows.

summary(train.Lintrend.season)

```
> #iv. Regression model with linear trend and seasonality
> train.Lintrend.season <- tslm(train.ts ~ trend + season)
> # See summary of linear trend and seasonality model and associated parameters.
> summary(train.Lintrend.season)

Call:
tslm(formula = train.ts ~ trend + season)

Residuals:
    Min      1Q  Median      3Q     Max
-7102.9 -1775.9   793.3  2007.1  7163.0

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 77001.66    1489.86  51.684  < 2e-16 ***
trend        1013.40      42.94  23.599  < 2e-16 ***
season2      3989.77    1578.02   2.528   0.0154 *
season3      2192.17    1612.89   1.359   0.1815
season4     14374.68    1613.46   8.909 3.88e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3864 on 41 degrees of freedom
Multiple R-squared:  0.9416,    Adjusted R-squared:  0.9359
F-statistic: 165.4 on 4 and 41 DF,  p-value: < 2.2e-16
```

Figure 5- Summary output from R

Model equation:

$Yt = 77001.66 + 1013.40t + 3989.77D_2 + 2192.17D_3 + 14374.68D_4$

Evaluating the model:

The R-squared and the adjusted R-squared values show that 94% of variation in the historical data can be explained using this regression model with linear trend and seasonality. This is a sign of good fit of the regression model.

The p-values associated with trend and intercepts are also extremely low (almost 0) which implies that the chances of the intercepts and trend becoming 0 are very low. Thus, making them statistically significant.

As for the p-values associated with the seasons, almost all of them are extremely lower than 5% implying that the seasons are statistically significant. Season 3 alone has a 6% chance that the value could be 0. This is the only aspect in the model that can be regarded as statistically insignificant.

F-statistic also reinforces our earlier evaluation of good fit. F-statistic is numerically high with a. low p-value. Hence this model is a good fit for our dataset.

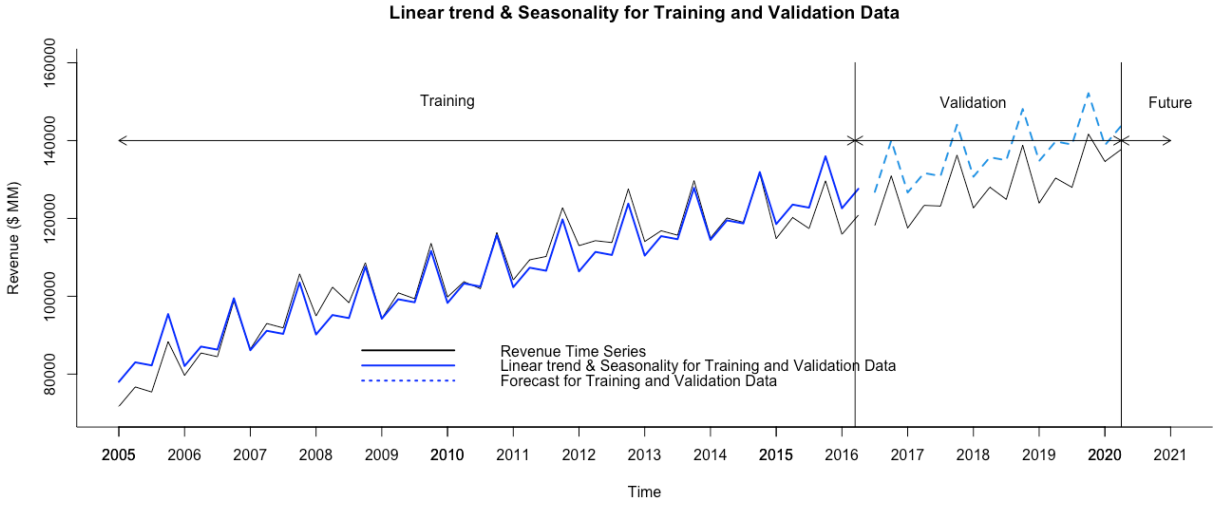
Forecasting using the regression model with linear trend and seasonality:

train.Lintrend.season.pred <- forecast(train.Lintrend.season, h = nValid, level = 0)

formattable(data.frame(train.Lintrend.season.pred))

	Point.Forecast	Lo.0	Hi.0
2016 Q3	126823.6	126823.6	126823.6
2016 Q4	140019.5	140019.5	140019.5
2017 Q1	126658.2	126658.2	126658.2
2017 Q2	131661.4	131661.4	131661.4
2017 Q3	130877.2	130877.2	130877.2
2017 Q4	144073.1	144073.1	144073.1
2018 Q1	130711.8	130711.8	130711.8
2018 Q2	135715.0	135715.0	135715.0
2018 Q3	134930.8	134930.8	134930.8
2018 Q4	148126.7	148126.7	148126.7
2019 Q1	134765.4	134765.4	134765.4
2019 Q2	139768.6	139768.6	139768.6
2019 Q3	138984.4	138984.4	138984.4
2019 Q4	152180.3	152180.3	152180.3
2020 Q1	138819.0	138819.0	138819.0
2020 Q2	143822.2	143822.2	143822.2

However, plotting the time series data, linear trend and seasonality and the forecast for validation data help us visually understand the forecasting tendencies of the model.



When we look at the above plot of the linear trend and seasonality for training and validation in comparison with the original dataset, we see that the model performs very well in the training capturing the trend and seasonality of the data. However, we can see some overestimation in the validation data. This can be improved using some smoothening techniques in addition to regression models.

v. Regression model with quadratic trend and seasonality.

We build the regression model with quadratic trend and seasonality using the tslm() function as follows.

train.Quadtrend.season <- tslm(train.ts ~ trend + I(trend^2) + season)

We get the summary statistics using,

summary(train.Quadtrend.season)

```
> # v. Regression model with quadratic trend and seasonality.
> train.Quadtrend.season <- tslm(train.ts ~ trend + I(trend^2) + season)
> # See summary of quadratic trend and seasonality model and associated parameters.
> summary(train.Quadtrend.season)

Call:
tslm(formula = train.ts ~ trend + I(trend^2) + season)

Residuals:
   Min     1Q Median     3Q    Max
 -3752  -1380     75   1360   5148

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 69770.953   1025.702  68.023  < 2e-16 ***
trend        1953.925     90.956  21.482  < 2e-16 ***
I(trend^2)    -20.011      1.877 -10.662 2.94e-13 ***
season2      3989.768    815.059   4.895 1.65e-05 ***
season3      1578.490    835.055   1.890    0.066 .
season4     13761.001    835.349  16.473  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1996 on 40 degrees of freedom
Multiple R-squared:  0.9848,    Adjusted R-squared:  0.9829
F-statistic: 518.7 on 5 and 40 DF,  p-value: < 2.2e-16
```

Figure 6- Summary output from R

Model equation:

$Y_t = 69770.95 - 1953.92t - 20.11t^2 + 3989.76\ D_2 + 1578.5\ D_3 + 13761.0\ D_4$

Evaluating the model:

The R-squared and the adjusted R-squared values show that 98% of variation in the historical data can be explained using this regression model with quadratic trend and seasonality. This is a sign of good fit of the regression model.

As for the p-values of the intercepts and trend, they are extremely lower than 5%. Hence, we can safely say that the trends and seasonality components are statistically significant.

As for the p-values associated with the seasons, almost all of them are extremely lower than 5% implying that the seasons are statistically significant. Season 3 alone has a 6% chance that the value could be 0. This is the only aspect in the model that can be regarded as statistically insignificant.

F-statistic also reinforces our earlier evaluation of good fit. F-statistic is numerically high with a. low p-value. Hence this model is a good fit for our dataset

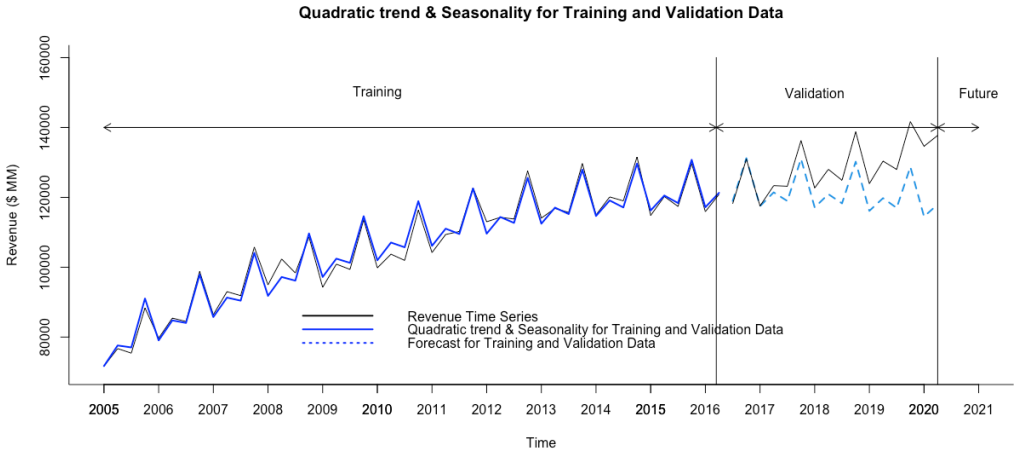
Forecasting using the regression model with quadratic trend and seasonality:

train.Quadtrend.season.pred <- forecast(train.Quadtrend.season, h = nValid, level = 0)

formattable(data.frame(train.Quadtrend.season.pred))

	Point.Forecast	Lo.0	Hi.0
2016 Q3	118979.2	118979.2	118979.2
2016 Q4	131214.6	131214.6	131214.6
2017 Q1	117466.4	117466.4	117466.4
2017 Q2	121429.0	121429.0	121429.0
2017 Q3	118950.5	118950.5	118950.5
2017 Q4	131025.8	131025.8	131025.8
2018 Q1	117117.5	117117.5	117117.5
2018 Q2	120920.0	120920.0	120920.0
2018 Q3	118281.4	118281.4	118281.4
2018 Q4	130196.6	130196.6	130196.6
2019 Q1	116128.3	116128.3	116128.3
2019 Q2	119770.7	119770.7	119770.7
2019 Q3	116972.0	116972.0	116972.0
2019 Q4	128727.1	128727.1	128727.1
2020 Q1	114498.7	114498.7	114498.7
2020 Q2	117981.0	117981.0	117981.0

However, plotting the time series data, quadratic trend and seasonality and the forecast for validation data help us visually understand the forecasting tendencies of the model.



When we look at the above plot of the quadratic trend and seasonality for training and validation in comparison with the original dataset, we see that the model performs very well in the training capturing the trend and seasonality of the data. However, we can see some underestimation in the validation data. This can be improved using some smoothening techniques in addition to regression models.

2c- Apply the accuracy() function to compare performance measure of the 5 forecasts you developed in 2b. Present the accuracy measures in your report, compare them, and, using MAPE and RMSE, identify the two most accurate regression models for forecasting.

When we apply the accuracy function to all the 5 forecasts from 2b, we get the following error profiles.

```
> ##COMPARING THE REGRESSION MODELS BASED ON ACCURACY
> # accuracy for Regression model with linear trend
> round(accuracy(train.lin.pred, valid.ts), 3)
                   ME      RMSE      MAE    MPE  MAPE  MASE   ACF1 Theil's U
Training set     0.00  6555.452 5127.986 -0.451 4.931 1.175 -0.084        NA
Test set     -8820.13 10377.449 8977.681 -7.046 7.162 2.057 -0.530      1.02
>
> # accuracy for Regression mode with quadratic trend
> round(accuracy(train.quad.pred, valid.ts), 3)
                   ME      RMSE      MAE    MPE  MAPE  MASE   ACF1 Theil's U
Training set    0.000  5587.708 4577.867 -0.278 4.297 1.049 -0.425        NA
Test set     8719.292 11998.816 9672.569  6.442 7.251 2.216  0.177     1.239
>
> # accuracy for Regression model with seasonality
> round(accuracy(train.season.pred, valid.ts), 3)
                   ME     RMSE      MAE    MPE   MAPE  MASE  ACF1 Theil's U
Training set     0.00 13930.36 11822.55 -1.975 11.931 2.709 0.929        NA
Test set     22806.25 23374.42 22806.25 17.636 17.636 5.225 0.720     2.375
>
> # accuracy for Regression model with linear trend and seasonality
> round(accuracy(train.Lintrend.season.pred, valid.ts), 3)
                    ME     RMSE      MAE    MPE  MAPE  MASE  ACF1 Theil's U
Training set     0.000 3647.876 2878.373 -0.181 2.844 0.659 0.827        NA
Test set     -8609.108 8776.477 8609.108 -6.720 6.720 1.973 0.309     0.893
>
> # accuracy for Regression model with quadratic trend and seasonality.
> round(accuracy(train.Quadtrend.season.pred, valid.ts), 3)
                   ME     RMSE      MAE    MPE  MAPE  MASE  ACF1 Theil's U
Training set    0.000 1861.038 1517.708 -0.031 1.463 0.348 0.669        NA
Test set     7533.251 9703.954 7668.093  5.691 5.803 1.757 0.750     0.975
>
```

Figure 7 - Accuracy output from R

The error profile for Regression model with linear trend and seasonality and the regression model with quadratic trend and seasonality are so much better than what we see with models capturing either just the trends or just seasonality. Considering the MAPE and RMSE for all the models in both the training and validation, we can see that the Regression model with linear trend and seasonality and the regression model with quadratic trend and seasonality have the lowest. Between these 2 models, regression model with quadratic trend and seasonality shows more accuracy in terms of MAPE, MAPE being 5.803. However, when we compare the RMSE, linear trend and seasonality model shows better performance with 8776.477.

Hence it is safe to say that models - Regression model with linear trend and seasonality and the regression model with quadratic trend and seasonality are the most accurate in this mix.

Q3 - Employ the entire data set to make time series forecast.

3a - Apply the two most accurate regression models identified in question to make the forecast for the last two quarters of 2020 and first two quarters of 2021. For that, use the entire data set to develop the regression model using the tslm() function. Apply the summary() function to identify the model structure and parameters, show them in your report, and also present the respective model equation. Use each model to forecast Walmart's revenue in the 4 quarters of 2020 and 2021 using the forecast() function, and present this forecast in your report.

iv. Regression model with linear trend and seasonality – forecast using full dataset:

Building the regression model with tslm() function.

revenue.Lintrend.season <- tslm(revenue.ts ~ trend + season)

summary(revenue.Lintrend.season)

```
> #iv. Regression model with linear trend and seasonality - full dataset
> revenue.Lintrend.season <- tslm(revenue.ts ~ trend + season)
> # See summary of linear trend and seasonality model and associated parameters.
> summary(revenue.Lintrend.season)

Call:
tslm(formula = revenue.ts ~ trend + season)

Residuals:
    Min      1Q  Median      3Q     Max
-9124.2 -3125.8   499.2  3413.9  8312.8

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 79919.20    1418.21  56.352  < 2e-16 ***
trend         854.41      30.23  28.264  < 2e-16 ***
season2      4193.15    1505.72   2.785  0.00726 **
season3      1711.60    1530.30   1.118  0.26806
season4     14095.79    1530.60   9.209 7.04e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4258 on 57 degrees of freedom
Multiple R-squared:  0.9411,    Adjusted R-squared:  0.937
F-statistic: 227.9 on 4 and 57 DF,  p-value: < 2.2e-16
```

Figure 8- Summary output from R

Model equation:

$Y_t = 79919.20 + 854.41t + 4193.15\ D_2 + 1711.60\ D_3 + 14095.79\ D_4$

Evaluating the model:

The R-squared and the adjusted R-squared values show that 94% of variation in the historical data can be explained using this regression model with linear trend and seasonality. This is a sign of good fit of the regression model.

The p-values associated with trend and intercepts are also extremely low (almost 0) which implies that the chances of the intercepts and trend becoming 0 are very low. Thus making them statistically significant.

As for the p-values associated with the seasons, almost all of them are extremely lower than 5% implying that the seasons are statistically significant. Season 3 alone has a 26% chance that the value could be 0. This is the only aspect in the model that can be regarded as statistically insignificant.

F-statistic also reinforces our earlier evaluation of good fit. F-statistic is numerically high with a. low p-value. Hence this model is a good fit for our dataset.

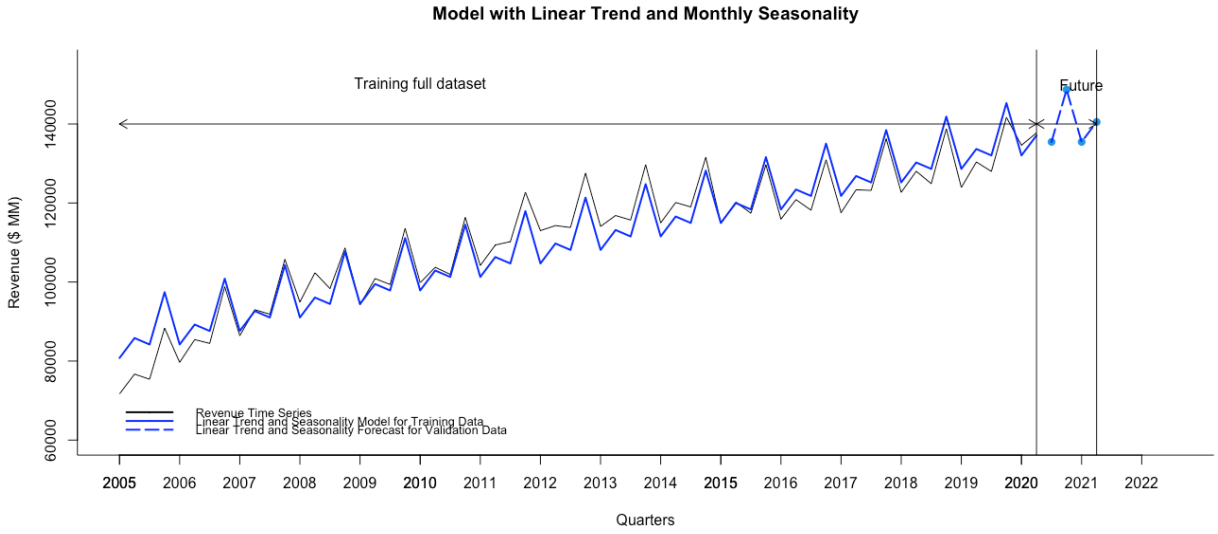
Forecasting using the complete dataset:

revenue.Lintrend.season.pred <- forecast(revenue.Lintrend.season, h = 4, level = 0)

formattable(data.frame(revenue.Lintrend.season.pred))

	Point.Forecast	Lo.0	Hi.0
2020 Q3	135458.9	135458.9	135458.9
2020 Q4	148697.5	148697.5	148697.5
2021 Q1	135456.2	135456.2	135456.2
2021 Q2	140503.7	140503.7	140503.7

Plotting the data to visualize the future forecast.



While forecasting using the full dataset, it is not possible to comment about the quality of the future forecast as we do not have historical data as benchmark. As far as the training is concerned, we can see that the model is able to predict close to the historical data. Hence we can get a closely accurate estimation of the future data using this model.

v. Regression model with quadratic trend and seasonality- forecast using full dataset:

Building the regression model with tslm() function.

revenue.Quadtrend.season <- tslm(revenue.ts ~ trend + I(trend^2) + season)

summary(revenue.Quadtrend.season)

```
> # v. Regression model with quadratic trend and seasonality.- full data set
> revenue.Quadtrend.season <- tslm(revenue.ts ~ trend + I(trend^2) + season)
> # See summary of quadratic trend and seasonality model and associated parameters.
> summary(revenue.Quadtrend.season)

Call:
tslm(formula = revenue.ts ~ trend + I(trend^2) + season)

Residuals:
    Min      1Q  Median      3Q     Max
-4605.0 -1701.3    25.3  1596.7  8218.6

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 72987.016   1268.134  57.555  < 2e-16 ***
trend        1524.243     82.998  18.365  < 2e-16 ***
I(trend^2)    -10.632      1.277  -8.325 2.26e-11 ***
season2      4193.148   1015.506   4.129 0.000122 ***
season3      1272.140   1033.433   1.231 0.223474
season4     13656.326   1033.634  13.212  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2872 on 56 degrees of freedom
Multiple R-squared:  0.9737,    Adjusted R-squared:  0.9713
F-statistic: 414.6 on 5 and 56 DF,  p-value: < 2.2e-16
```

Figure 9- Summary output from R

Model equation:

$$Y_t = 72987.01 + 1524.243\ t - 10.63t^2 + 4193.148\ D_2 + 1272.140\ D_3 + 13656.326\ D_4$$

Evaluating the model:

The R-squared and the adjusted R-squared values show that 97% of variation in the historical data can be explained using this regression model with quadratic trend and seasonality. This is a sign of good fit of the regression model.

As for the p-values of the intercepts and trend, they are extremely lower than 5%. Hence, we can safely say that the trends and seasonality components are statistically significant.

As for the p-values associated with the seasons, almost all of them are extremely lower than 5% implying that the seasons are statistically significant. Season 3 alone has a 22% chance that the value could be 0. This is the only aspect in the model that can be regarded as statistically insignificant.

F-statistic also reinforces our earlier evaluation of good fit. F-statistic is numerically high with a. low p-value. Hence this model is a good fit for our dataset

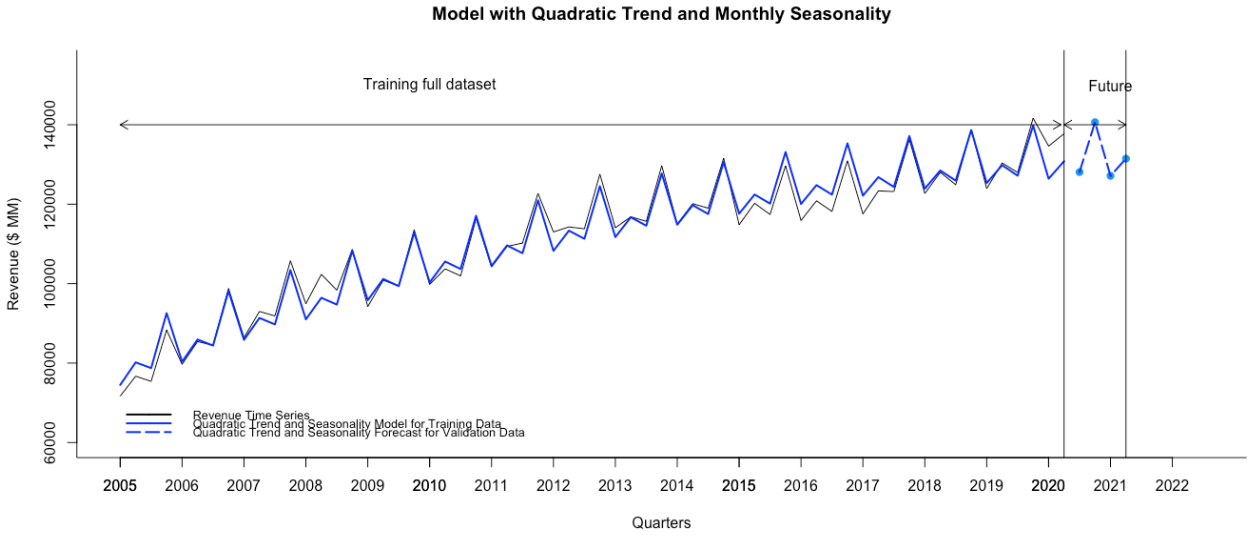
Forecasting using the complete dataset:

revenue.Quadtrend.season.pred <- forecast(revenue.Quadtrend.season, h = 4, level = 0)

formattable(data.frame(revenue.Quadtrend.season.pred))

	Point.Forecast	Lo.0	Hi.0
2020 Q3	128087.3	128087.3	128087.3
2020 Q4	140645.4	140645.4	140645.4
2021 Q1	127141.8	127141.8	127141.8
2021 Q2	131466.4	131466.4	131466.4

Plotting the data to visualize the future forecast:



While forecasting using the full dataset, it is not possible to comment about the quality of the future forecast as we do not have historical data as benchmark. As far as the training is concerned, we can see that the model is able to predict close to the historical data. Hence, we can get a closely accurate estimation of the future data using this model.

3b- Apply the accuracy() function to compare the performance measures of the regression models developed in 3a with those for naïve and seasonal naïve forecasts. Present the accuracy measures in your report, compare them, and identify, using MAPE and RMSE, which forecast is most accurate to forecast Walmart's quarterly revenue in 2020 and 2021.

Comparing the error profiles of the regression models with linear trend and seasonality, quadratic trend and seasonality with the naïve and seasonal naïve forecasts.

```
> #COMPARING ACCURACIES
> # accuracy for Regression model with linear trend and seasonality
> round(accuracy(revenue.Lintrend.season.pred$fitted, revenue.ts), 3)
         ME     RMSE     MAE    MPE  MAPE  ACF1 Theil's U
Test set  0 4082.663 3416.38 -0.224 3.247 0.875     0.463
> # accuracy for Regression model with quadratic trend and seasonality.
> round(accuracy(revenue.Quadtrend.season.pred$fitted, revenue.ts), 3)
         ME     RMSE      MAE    MPE  MAPE  ACF1 Theil's U
Test set  0 2729.216 2077.481 -0.076 1.888 0.755     0.282
> round(accuracy((naive(revenue.ts))$fitted, revenue.ts), 3)
               ME    RMSE      MAE MPE  MAPE   ACF1 Theil's U
Test set 1082.984 9614.53 8103.902 0.7 7.247 -0.716         1
> round(accuracy((snaive(revenue.ts))$fitted, revenue.ts), 3)
               ME     RMSE      MAE   MPE  MAPE  ACF1 Theil's U
Test set 3964.224 5074.236 4163.845 3.696 3.873 0.753     0.558
>
```

Figure 10-Accuracy output from R

RMSE and MAPE easy comparison:

Model	RMSE	MAPE
Regression model with linear trend and seasonality	4082.663	3.247
Regression model with quadratic trend and seasonality	2729.216	1.888
Naïve	9614.53	7.247
Seasonal Naïve	5074.236	3.873

From the above table we can clearly see that the Regression model with quadratic trend and seasonality has the lowest RMSE and MAPE while forecasting for the future using out complete dataset. We see that Naïve model has at least 3 times the RMSE of the quadratic trend model, while seasonal Naïve has at least double the RMSE. The overall MAPE is significantly lower for the quadratic trend model when compared to all the other models.

Hence the most accurate model in terms of RMSE and MAPE for forecasting Walmart's quarterly revenue in 2020 and 2021 would be the **Regression model with quadratic trend and seasonality.**