**Time Series Analysis-Trend and Seasonality**

Time series analysis captures the pattern of change in data. Traditionally, these patterns are divided into four categories.

1. Trend
2. Seasonality
3. Cyclical pattern
4. Random fluctuation

**Trend:** It is a long-term increase and decrease of data in a continues patter.

**Seasonality:** It represents changes in data in regular pattern.

**Cyclical pattern:** This pattern is defined when a sustained period of high values follows by a series of low values.

**Random fluctuation:** In random fluctuation, the data does not follow any desirable pattern. There is now correlation between successive changes and hence it is called random or random walk.

**Trend Projections**

1. Graphical curve fitting
2. Statistical curve fitting

Statistical curve fitting uses OLS technique to estimate parameters and fit a linear model. There are two basic types of approaches such as

1. Constant rate of change
2. Constant percentage rate of change

**Example:**

***Constant rate of change***

Suppose we have quarterly sales figure for three years where we notice that Q4 sales of every year is relatively high as compare to other quarter sales. Since sales data is a time series, we can relate a time frame to each quarter sales data. Using statistical approach let’s regress sales (S) over time (T)

St = α + βT

Where, St  = sales at time t period

α = Intercept or constant

β = Slop coefficient of time T

After regressing the given sales data, we fit the following model.

St = 281.39 + (12.81×T)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Date** | **Sales (S)** | **Time (T)** |
| Q1 | 30-06-2017 | 300 | 1 |
| Q2 | 30-09-2017 | 305 | 2 |
| Q3 | 31-12-2017 | 315 | 3 |
| Q4 | **31-03-2018** | **340** | 4 |
| Q1 | 30-06-2018 | 346 | 5 |
| Q2 | 30-09-2018 | 352 | 6 |
| Q3 | 31-12-2018 | 364 | 7 |
| Q4 | **31-03-2019** | **390** | 8 |
| Q1 | 30-06-2019 | 397 | 9 |
| Q2 | 30-09-2019 | 404 | 10 |
| Q3 | 31-12-2019 | 418 | 11 |
| Q4 | **31-03-2020** | **445** | 12 |
| Q1 | 30-06-2020 | ? | 13 |

Now using the regression model, we can predict Q1 sales of 2020 as

SQ1, 2020 = 281.39 + (12.81× 13) = 447.9

***Constant percentage rate of change***

In this approach, we can use rate of growth between quarters as significant too to forecast.

St = St-1 (1+g)

Where, g = rate of growth parameter

For Q1, g =

Q2, g =

Then in our case rate of growth of Q1:

Q12018 = = 0.153

Q12019 = = 0.147

Hence Average growth rate of Q1 sales is around 15%. Then Q1 2020 sales would be

Q12020 = 397 × (1.15) = 456.7

**Seasonal Adjustments of Time series data**

Let’s use the above time series model with a deterministic time trend to predict sales.

St = 281.39 + (12.81×T)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Date** | **Sales (S)** | **Time (T)** | **sales-Forecast** |
| Q1 | 30-06-2017 | 300 | 1 | 294.2 |
| Q2 | 30-09-2017 | 305 | 2 | 307.0 |
| Q3 | 31-12-2017 | 315 | 3 | 319.8 |
| Q4 | **31-03-2018** | **340** | 4 | **332.6** |
| Q1 | 30-06-2018 | 346 | 5 | 345.4 |
| Q2 | 30-09-2018 | 352 | 6 | 358.3 |
| Q3 | 31-12-2018 | 364 | 7 | 371.1 |
| Q4 | **31-03-2019** | **390** | 8 | **383.9** |
| Q1 | 30-06-2019 | 397 | 9 | 396.7 |
| Q2 | 30-09-2019 | 404 | 10 | 409.5 |
| Q3 | 31-12-2019 | 418 | 11 | 422.3 |
| Q4 | **31-03-2020** | **445** | 12 | **435.1** |
| Q1 | 30-06-2020 | ? | 13 | **447.9** |

In the above table, we can see that the time series model could not capture the seasonality components in the sales data. Because the regression model has no scope to capture seasonality component separately. Hence, we need to adjust it separately.

***Ratio to Trent approach:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Seasonal Adjustment of sales data | | | | |
| **date** | **Sales (S)** | **sales-forecast (SF)** | **Ratio (S/SF)** | **Seasonal-Adj-Sales** |
| **31-03-2018** | **340** | **332.6** | 1.022 | (332.6 × 1.02) = 339.3 |
| **31-03-2019** | **390** | **383.9** | 1.016 | (383.9 × 1.02) = 391.6 |
| **31-03-2020** | **445** | **435.1** | 1.023 | (435.1 × 1.02) = 443.8 |
|  |  | Average | **1.02** |  |

Similarly, the Q4 projected sales figure will also be adjusted with the seasonal adjustment coefficient (1.02 in the present case) to have more accuracy.

**Exponential Smoothing**

Forecasting using time trend through regression models, uses time as an independent variable. One of the key features of this type of model is that it assigns equal weight to all the observations. But in some cases, recent observation carries more importance than distance data points. If we wish to assign asymmetric weight to time frame and assume that recent observations are more important than distance past, then exponential smoothing is one of the best time series models to follow.

Fn+1 = αXn + (1- α) Fn

Where, Fn+1 = Forecast for n+1 time

Xn = Actual data at time n

Fn = Forecasted data at time n.

Αlpha (α) is the smoothing parameter that lies between 0 < α > 1.

If, α = 0, it assigns entire 100% weight to forecasted value. α closer to zero implies model is smoother. As α moves closer to 1, the model starts assign higher weight to recent observation.

**Optimum α:**

An optimum α i.e. exponential smoothing parameter is decided on the basis of overall accuracy of the model, that can be decided using RMSE or Sum of the square deviation of the error distribution.

Let’s analyze the following sales data and predict using exponsential smoothing model. Also identify an appropriate smoothing parameter **α.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **α = 0.4** |  | **α = 0.6** |  | **α = 0.8** |  |
| **date** | **sales(X)** | **XF1** | **Error (X-XF1)2** | **XF2** | **Error (X-XF2)2** | **XF3** | **Error (X-XF3)2** |
| 1 | 400 | 400 | 0.0 | 400 | 0 | 400 | 0 |
| 2 | 430 | 400 | 900.0 | 400 | 900 | 400 | 900 |
| 3 | 420 | 412 | 64.0 | 418 | 4 | 424 | 16 |
| 4 | 440 | 415 | 615.0 | 419 | 433 | 421 | 369 |
| 5 | 460 | 425 | 1216.6 | 432 | 802 | 436 | 568 |
| 6 | 440 | 439 | 0.9 | 449 | 75 | 455 | 232 |
| 7 | 470 | 439 | 933.7 | 443 | 704 | 443 | 726 |
| 8 | 430 | 452 | 469.4 | 459 | 864 | 465 | 1198 |
| 9 | 440 | 443 | 9.0 | 442 | 3 | 437 | 9 |
| 10 | 420 | 442 | 475.2 | 441 | 429 | 439 | 376 |
| 11 |  | 433 |  | 428 |  | 424 |  |
| Sum of Squared Error | | | 4683.9 |  | **4213.0** |  | 4394.5 |