**R:**

* In the previous version of R, we just had only Actual and forecasted graph.
* In new version – The actual dataset got splitted into Train and Test Data. We then trained the model using train data and do the validation using test data.
* Chart 1: Train Data
* Chart 2: Test Data
* Chart 3: Predicting the Test Data
* Chart 4: Forecasting the Future Data
* In addition to that, we have included two parameters – **Percentage Change** & No. of **Years**.

**Python:**

* There is not much change in the python version but in addition to the previous version we have included a new Parameter called **Years**.
* The approach and procedure used in R and Python is almost same.
* Chart 1: Train Data
* Chart 2: Test Data
* Chart 3: Predicting the Test Data
* Chart 4: Forecasting the Future Data

**Native Power BI:**

* This is same as Previous version.
* Power BI has in-built feature/Functionality to forecast future data. This forecasting feature is lost when we make use of Parameter. So, Parameter is removed here.

**Variance:**

* This is the new tab we have included in the report where we displayed the following:
* Chart 1: Python and R (Actual + Forecasted Data)
* Chart 2: Native Power BI
* Chart 3: Only forecasted Data (Python + R)

Note: Same Dataset has been used for Python, R and Native Power BI.

* Initially R and Python were two separate files.
* In the updated version we have integrated Python, R and Native Power BI in one file.
* We have used same Dataset for Python, R & Native Power BI.
* In python, the algorithm we used is Exponential Smoothening and for R the algorithm used is Holt’s Winter.
* Same processing steps has been used both in Python and R
* In the previous version we had only one parameter i.e., Percentage Change. Now, we have included one more Parameter – Years in the updated version.
* Preprocessing steps: Organized all the tables under Edit – Queries.
* Changed the Look and feel of the Visualizations.
* Maintained Standards in naming the objects.
* Maintained same colors for respective language (example: Python – is represented using Green color throughout the project).
* We added a new page called – Variance which displays Python and R in one graph. Native Power bi in a separate graph. It also shows only the forecasted data of Python and R.