Assignment: MLSL Individual Assignment

Batch – AMPBA-2021S

ID – 12010066

Name – Debjit Ray

Problem Statement

Build a prediction model to predict if a Fedex shipment will be delivered on/ before time.

Business Objective

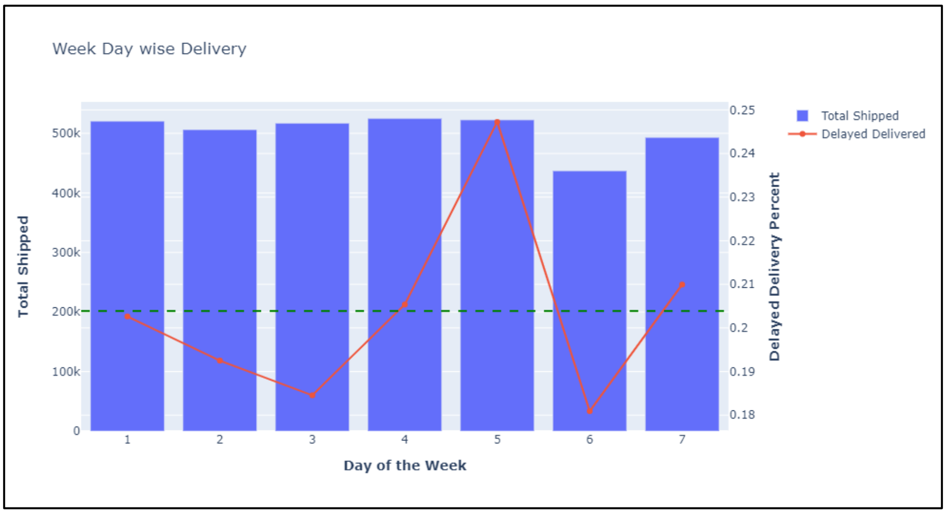
Maximize on time delivery.

Business Constraints

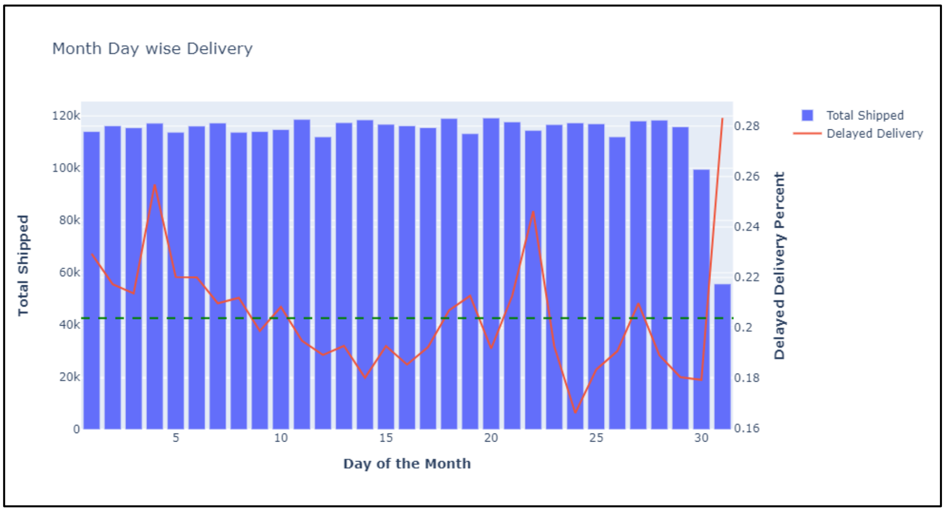
Minimize any delays and unpredictability which will have negative impact on the organization.

Exploratory Data Analysis Findings

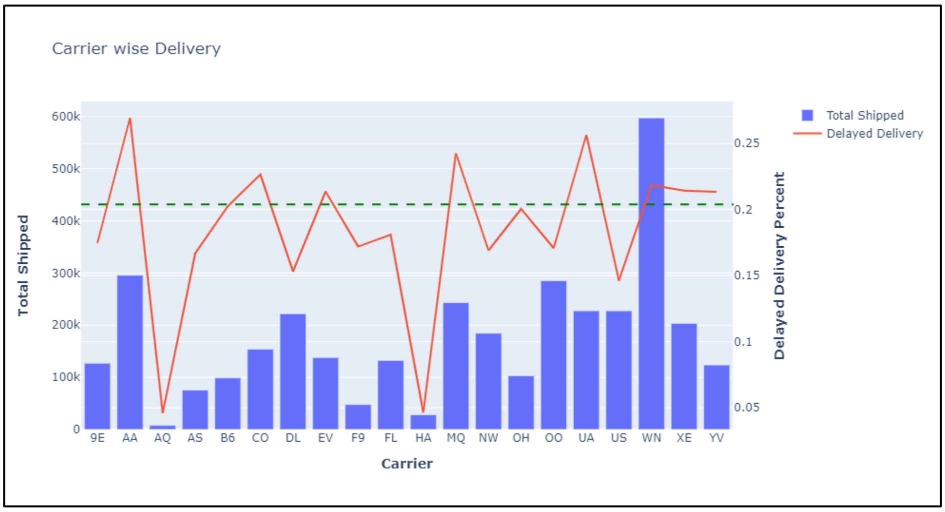
- There are 3,604,175 (3.6 mn.) observations in the dataset.
- All the data are from 2008 Jan to Jun
- Not all records have the delivery status, shipment delay, planned time of travel and actual shipment time details.
- There are 20 unique carriers, 297 unique Sources and 299 unique Destinations
- The data is imbalanced with 79.6% on time delivery and 20.4% late deliveries.
- Even by removing the incomplete records, we do not lose out any of the Carriers or Source or Destination.
- The incomplete records are from the same population as the complete records.



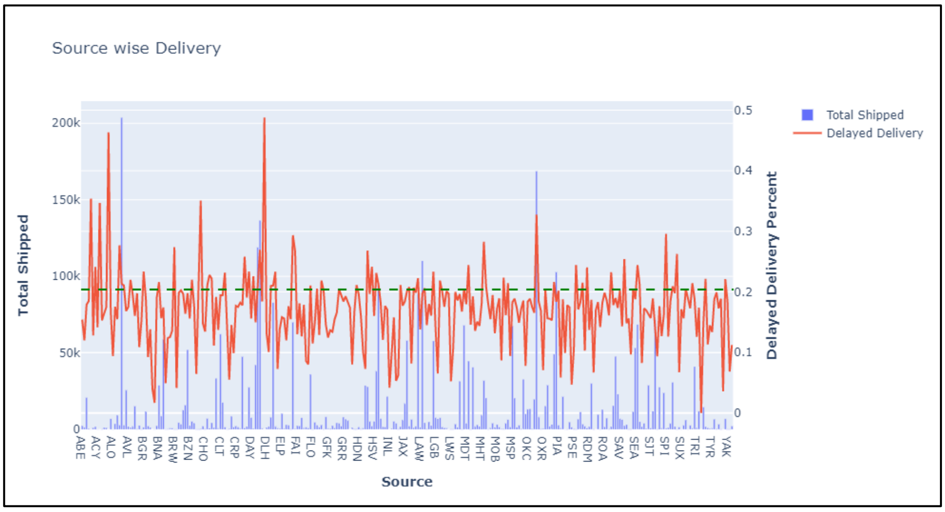
- Volume shipped each day is almost equally distributed except, day 6 and day 7 of the week having slightly less numbers shipped.
- Items shipped on Day 5 of the week have a greater probability of getting delivered late.
- Similarly, the items shipped on Day 6 and Day 3 of the week have the highest probabilities of getting delivered on/ before time.

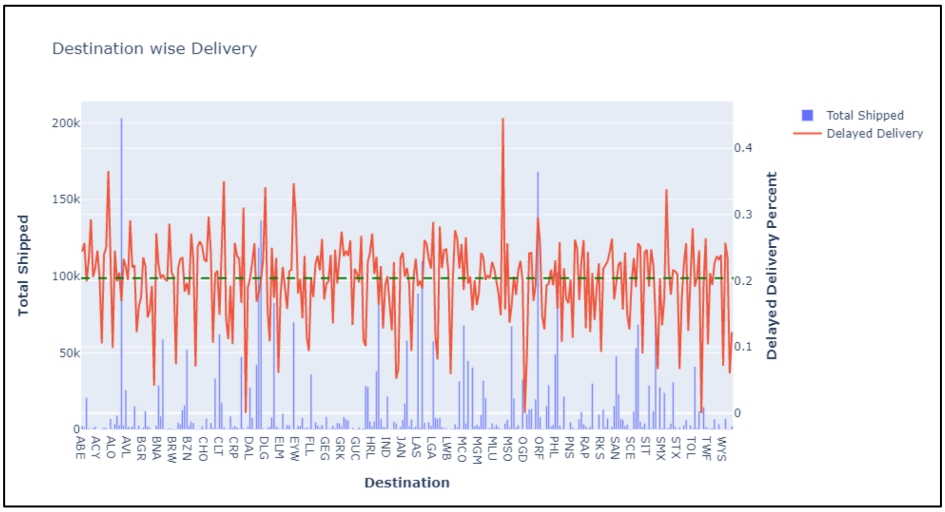


- Except day 30 and 31, most of the other days of the month have almost equal number of items shipped. Day 31 total counts is less because in the first 6 months of the year only 3 months have 31 days. Also, Feb has only 29 days, hence, the total cunt for Day 30 is slightly less.
- This plot reveals that items shipped on Day 24 of the month has the highest probability of getting delivered on time.
- Similarly, the items shipped on Day 31, Day 4 and Day 22 have the least probabilities of getting delivered on time.

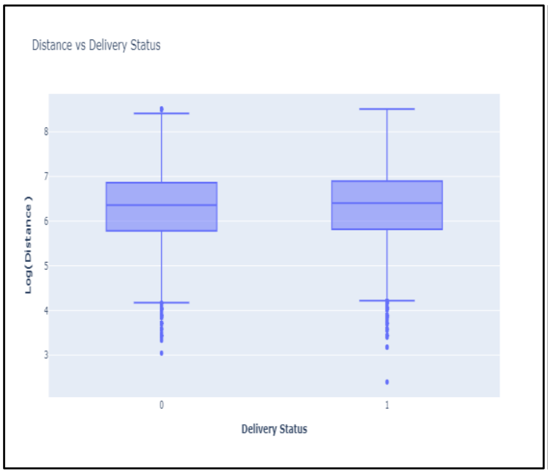


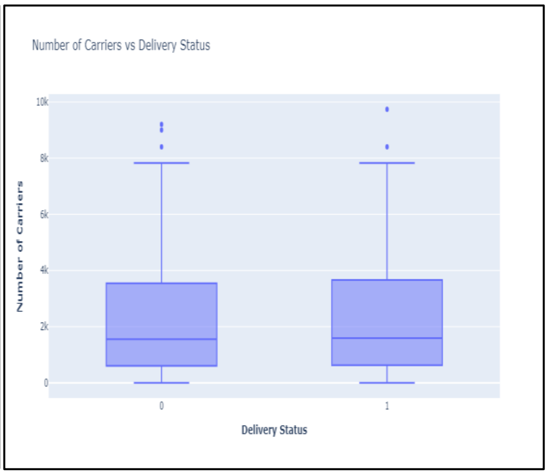
- This plot reveals that carrier "WN" has shipped the highest number of shipments.
- Carriers ike "AQ" and "HA" have the lowest number of shipments but best on/ before time delivery percent.
- "AA" ships a decent number of shipments but has the highest probability of delayed delivery amongst all carriers.



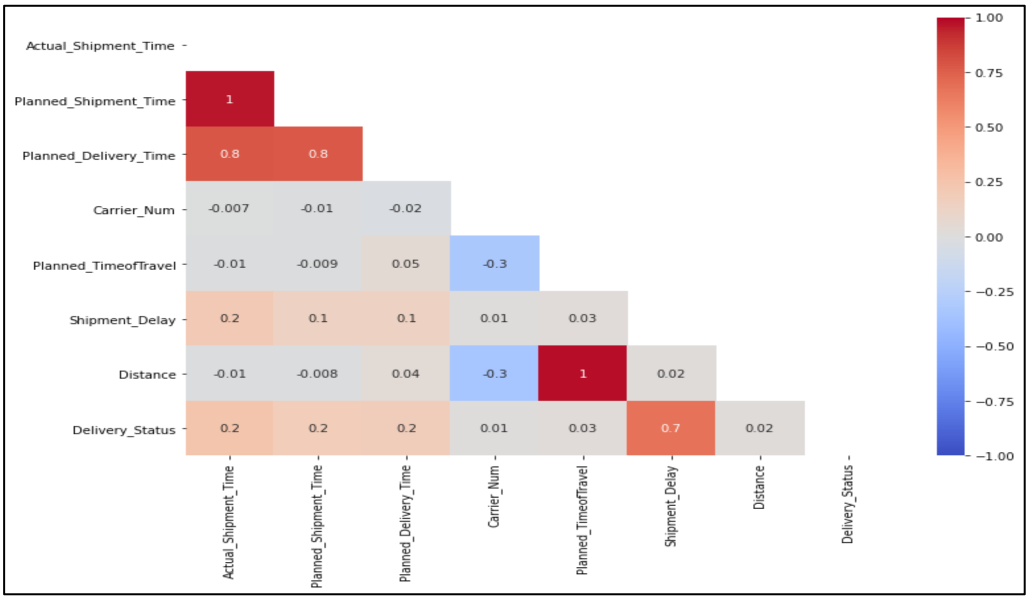


- The source/ destination wise plots show quite a lot of variations in both count of items shipped and delivery percent.



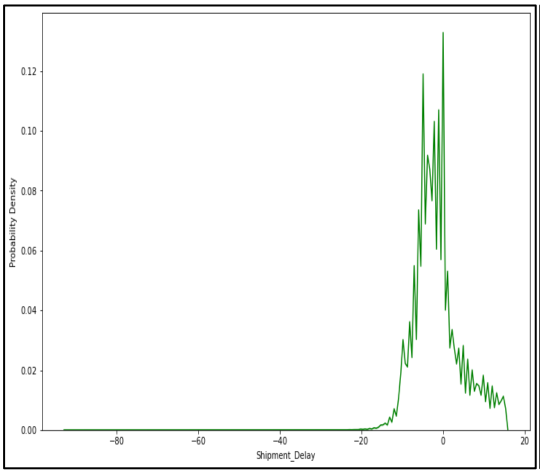


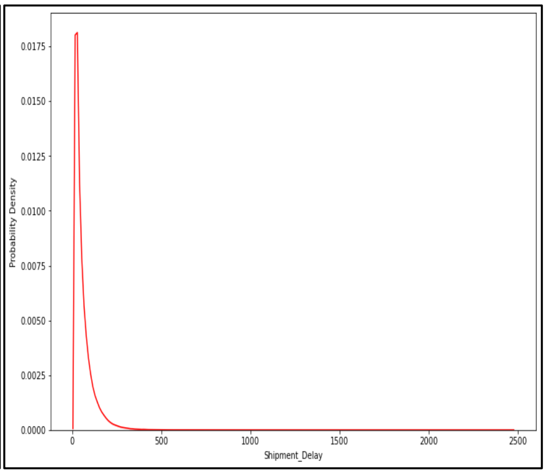
- There is very little to distinguish whether a Delivery is delayed or not based on Distance or Number of Carriers.



- Perfect correlation between *"Distance"* and *"Planned_TimeofTravel"*. This is obviously expected as larger distance is expected to take more time to deliver.
- *"Actual_Shipment_Time"* and *"Planned_Shipment_Time"* are perfectly collinear. However, as checked in most of the cases the *"Planned_Shipment_Time"* is not equal to "*Actual_Shipment_Time*".

- "*Planned_Shipment_Time*" and "*Planned_Delivery_Time*" have high correlation of 0.8. This is expected as "*Planned_Delivery_Time*" most probably is calculated as some function of "*Planned_TimeofTravel*" and "*Planned_Shipment_Time*".
- *"Actual_Shipment_Time"* and "*Planned_Delivery_Time"* have high correlation of 0.8.
- "*Shipment_Delay*" has a high correlation of 0.7 with "*Delivery_Status*".



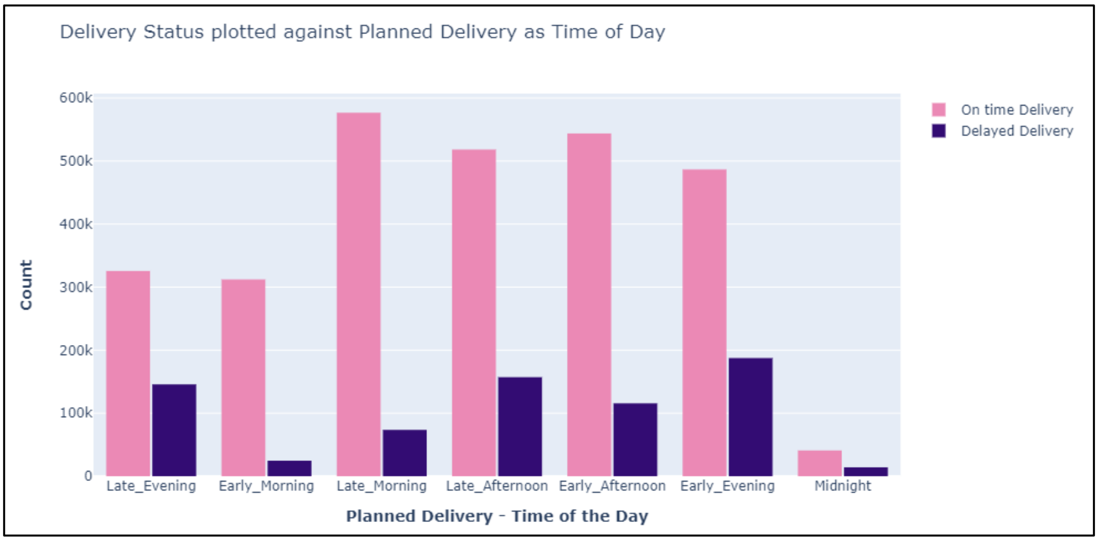


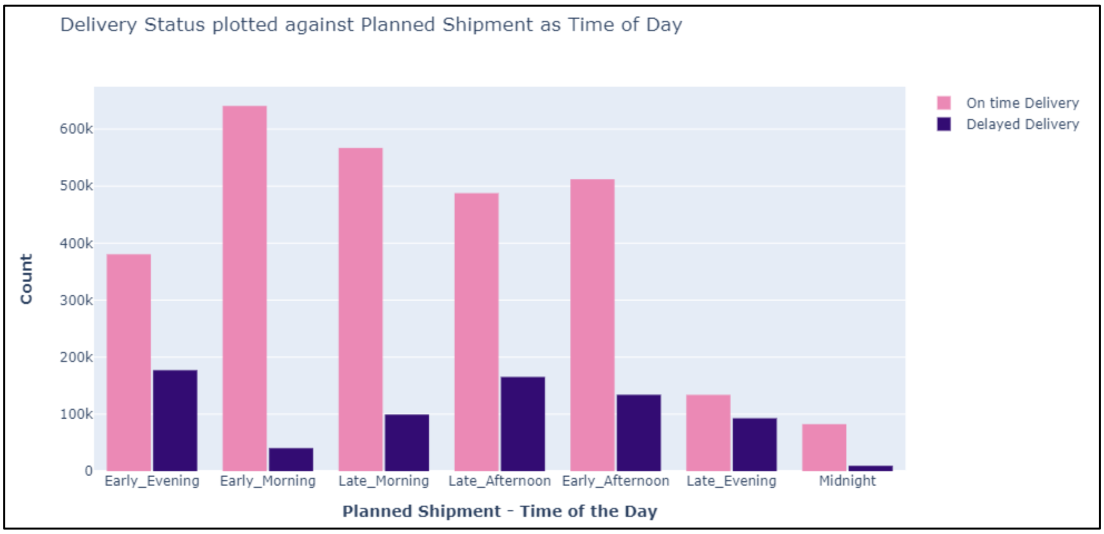
- **A package shipped early i.e. negative Shipment Delay always ensure timely Delivery ("*Delivery_Status*" = 0)**
- **Packages shipped beyond 15 mins delay never make it on time whereas, packages shipped early/ or upto 15 minutes delay always make it on/before time.**

Data Preparation

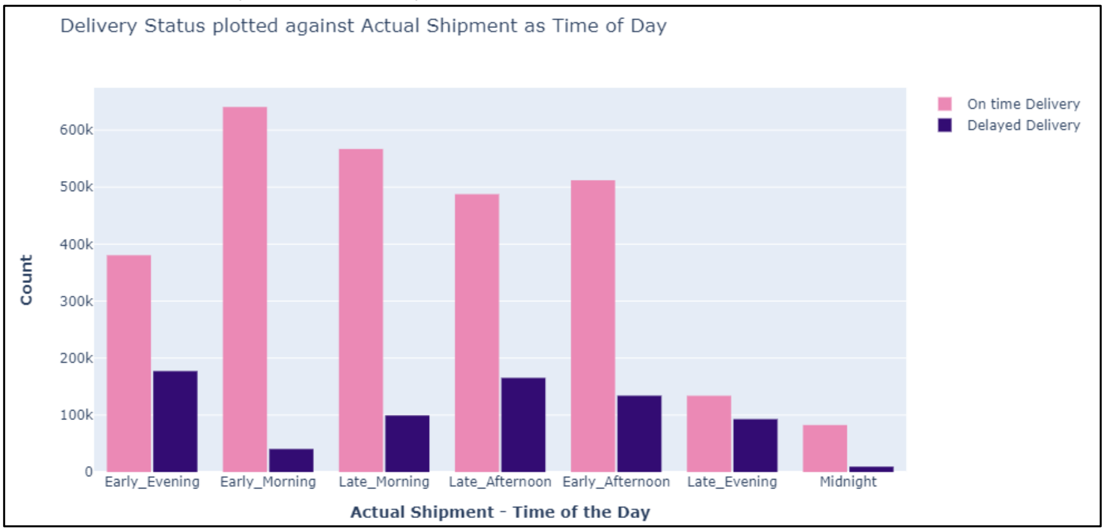
As part of the data preparation, we do the following:

- We drop the *'Year'* field as it is constant through all the records.
- We convert the planned delivery time and planned shipment time into bins, we find that the Delivery Status does differ for different timeslots of the day.

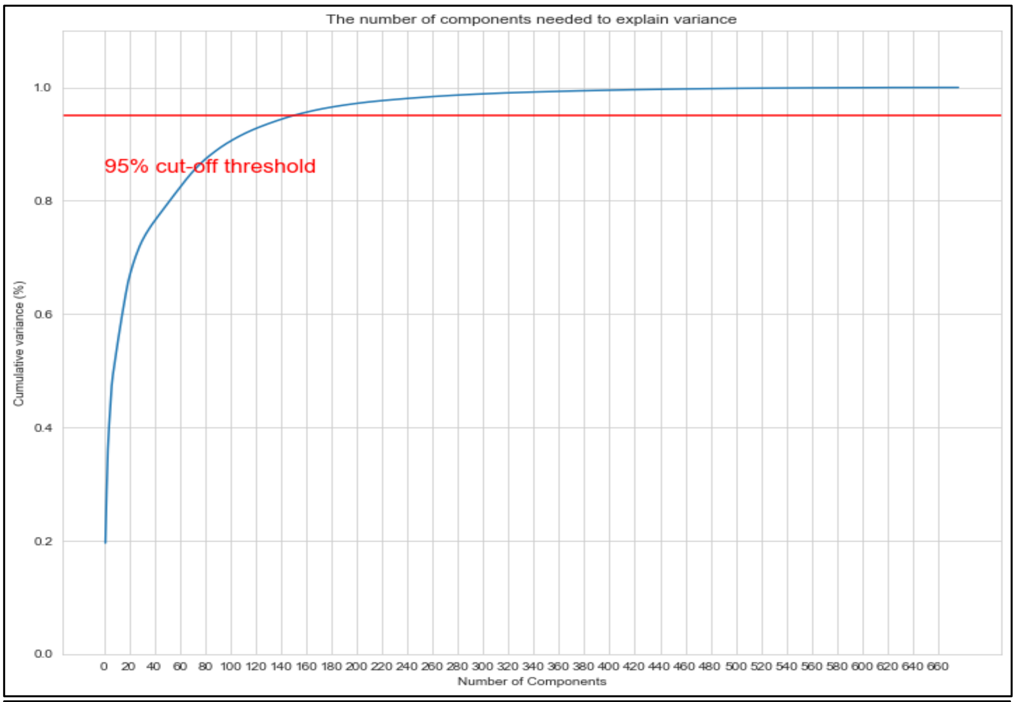




- We convert the actual shipment time, too into bins, and here too we find that the actual time differs for different parts of the day.



- Convert all the categorical variables using One Hot encoding. The variables converted are *'Month', 'DayOfWeek', 'DayofMonth', 'Carrier_Name', 'Source', 'Destination', 'Actual_Shipment_Bin', 'Planned_Shipment_Bin', 'Planned_Delivery_Bin'*
- After splitting the dataset into Training, Validation and Test datasets, we apply standard Scaler to scale all the numeric features and then apply PCA to choose the principal components.




```
In [19]: np.cumsum(preProcessing_pipeline.named_steps.reduceDimensions.ipca.explained_variance_ratio_)

Out[19]: array([0.19583992, 0.28786082, 0.36212547, 0.40463084, 0.43965202,
                0.47345372, 0.49329052, 0.5086754 , 0.52385638, 0.53890989,
                0.55319998, 0.56744994, 0.58130458, 0.59511283, 0.60841275,
                0.62123005, 0.63380122, 0.64542195, 0.65695557, 0.66590801,
                0.67396737, 0.68185956, 0.68882913, 0.6955943 , 0.7020234 ,
                0.70796016, 0.71378463, 0.71921386, 0.72428704, 0.72888997,
                0.73317635, 0.73740958, 0.74119518, 0.74483339, 0.74832453,
                0.75179674, 0.75521176, 0.75840642, 0.76152677, 0.76456797,
                0.76757793, 0.77058543, 0.77358165, 0.77656323, 0.77954054,
                0.78251411, 0.78547843, 0.78844037, 0.79139694, 0.79434578,
                0.7972915 , 0.80023494, 0.80316801, 0.80610003, 0.80901725,
                0.81193022, 0.81483795, 0.81773849, 0.82063263, 0.82351492,
                0.82638057, 0.82924317, 0.83208063, 0.83490631, 0.83772834,
                0.84047393, 0.84316037, 0.84582747, 0.84841335, 0.85094644,
                0.85340557, 0.85581991, 0.85822724, 0.86060126, 0.86286789,
                0.86500471, 0.8671297 , 0.86922863, 0.87126585, 0.8732235 ,
                0.87512449, 0.87698896, 0.87880782, 0.88058971, 0.88234616,
                0.88408642, 0.88577335, 0.88744946, 0.88909122, 0.89071706,
                0.89229898, 0.89386159, 0.8953872 , 0.89689854, 0.89838433,
```

- We choose to retain just 50% of the variance by using the first 8 principal components for our model build process.
- The scaled and reduced dimensionality dataset will be used to train, validate and test our models.

Modelling

Considering, our target variable – Delivery Status has just 2 values – 0 and 1, we have a classification problem in hand. Hence, we will build the below models:

- Linear Discriminant Analysis
- Decision Tree
- Random Forest
- Gaussian Naïve Bayes
- K Nearest Neighbours

We train each of these models using our transformed[1] training dataset and test using the transformed validation dataset. On checking the Accuracy, Precision and Recall values, we find the best performing model is Random Forest.

---

[1] By transformed, we mean scaling the original data and then extracting the Principal Components using PCA.

Hyperparameter Tuning

To tune our models to perform even better, we will use GridSearchCV to search for the best parameter combination.

The hyperparameters, we choose to tune for Random Forest are:

```
# Hyper parameters for Random Forest
depth = [20, 40]
min_sampl_leaf = [1, 20]
min_sampl_split = [2, 20, 40]
class_wt = [None, {"0": 0.796, "1":0.204}] #Original distribution of data
```

Retraining & Testing the Final Model

Finally, we define the final model – Random Forest using the hyper parameters identified by GridSearchCV. We test the final model and plot the ROC curve and confusion matrix.

Conclusion

Considering the predictor variable, Shipment Delay can directly help attribute whether the shipment will arrive delayed or on time, we do not need a model. Any shipment delays greater than 15 mins, can directly be predicted to be Delayed and all others can be listed as on time.

However, if we really need to build a model; Random Forest would be the best fit as seen above.