www.delhivery.com

Cash on Delivery Return Prediction

11th July 2017

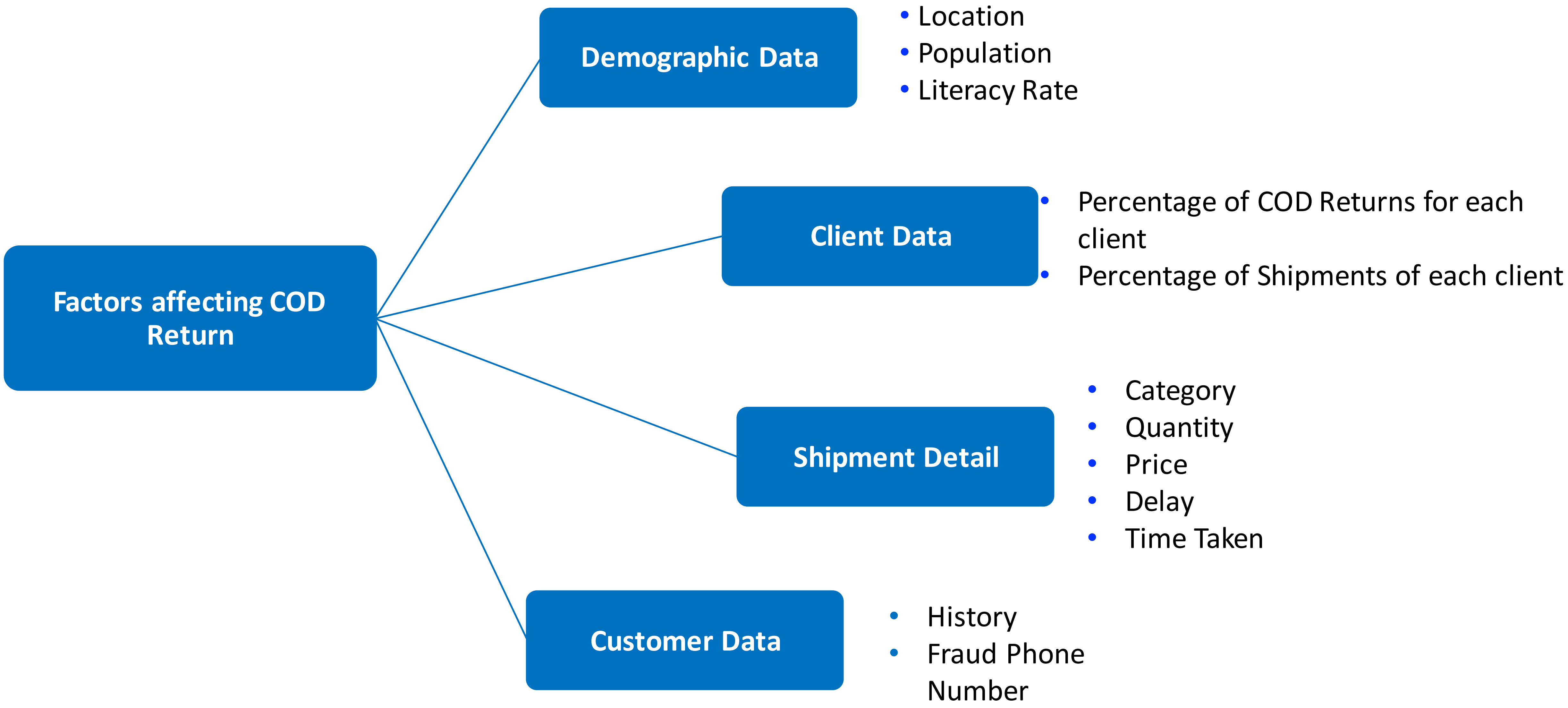
Problem Statement

Business Problem :

- On an average 17 % of the total COD shipments are returned.
- Returned shipments have higher probability of getting lost.
- This leads to additional cost and wastage of time for both the clients and the business.

Objective :

- To reduce the percentage of returned shipments and to avoid unwanted wastage of resources the probability of a COD shipment being returned needs to be predicted.
- ***So the objective here is to predict the probability of each shipment of being returned.***


Factors affecting COD Return
Demographic Data
Location
Population
Literacy Rate
Client Data
Percentage of COD Returns for each client
Percentage of Shipments of each client
Shipment Detail
Category
Quantity
Price
Delay
Time Taken
Customer Data
History
Fraud Phone Number

Hypothesis Generation

Hypothesis Generation

Demographic Data :

- **Location** - Higher return probability of shipments in Metros followed by Tier-I, II , III, IV
- **Population** - Higher the population of the city, more the number of returns
- **Literacy Rate** - More COD return rate expected with area having higher literacy rate

Shipment Details :

- **Category of the Shipment** – Certain categories will have higher return rates compared to others.
- **Quantity of the Shipment in each delivery** – Cod return will be higher on the days when the number of shipments is more compared to the average
- **Price of the Shipment** – More is the price of the shipment, higher will be probability of its return
- **Time taken** – If the Time taken is higher then the shipment is likely to be returned
- **Delay** – If there is delay, then shipment is likely to be returned

Contd.

Client Data :

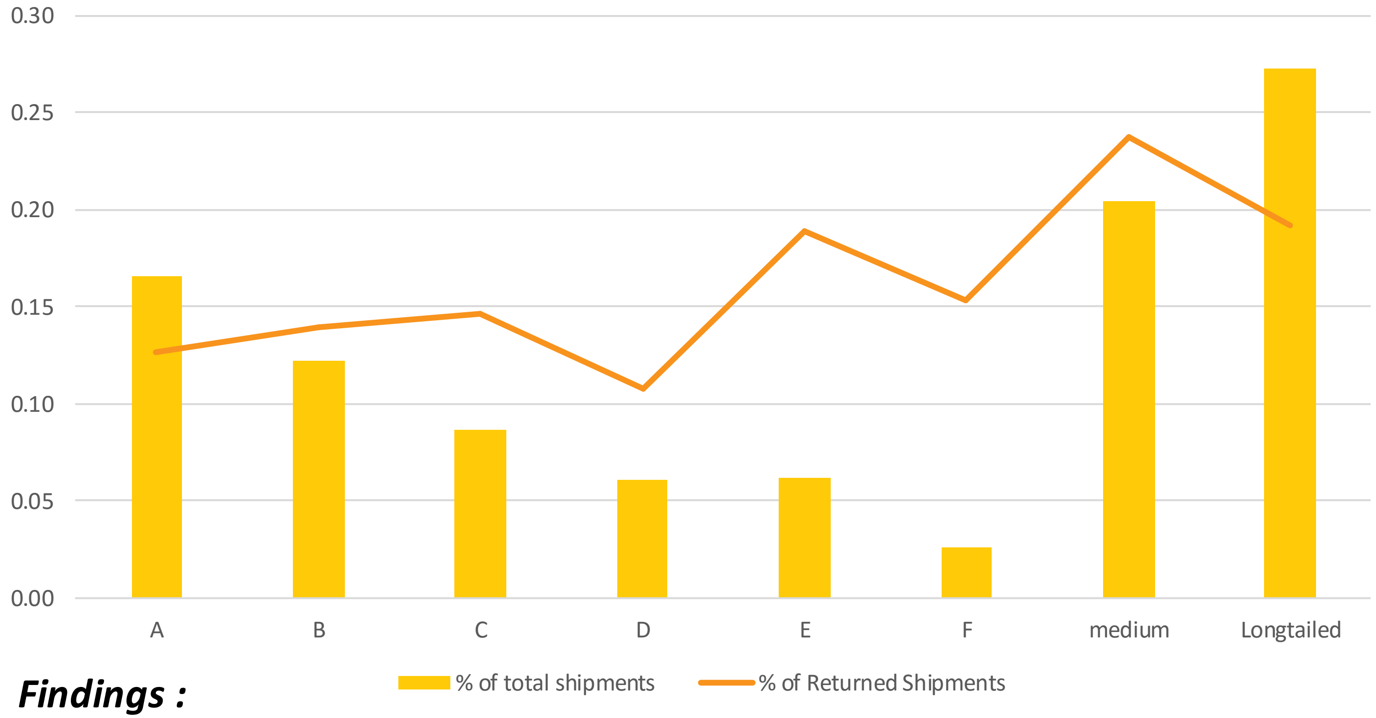
- **Percentage of COD Returns for each client** - Higher the COD return percentage, higher the return probability of its shipment.
- **Percentage of Shipments of each client** – Higher is the Percentage, higher the return probability of its shipment.

Customer Data :

- **History of the Customer** – If the customer has had high returns then return probability of its shipment will be higher
- **Fraud Phone Number** – If a fraud number is given by a customer then return probability of its shipment will be higher.

Exploratory Data Analysis

Clients – Volume Distribution and Return Rates



Buckets	%Returned	No.of clients
HIGH	0.25	346
MEDIUM	0.14	170
LOW	0.09	1219

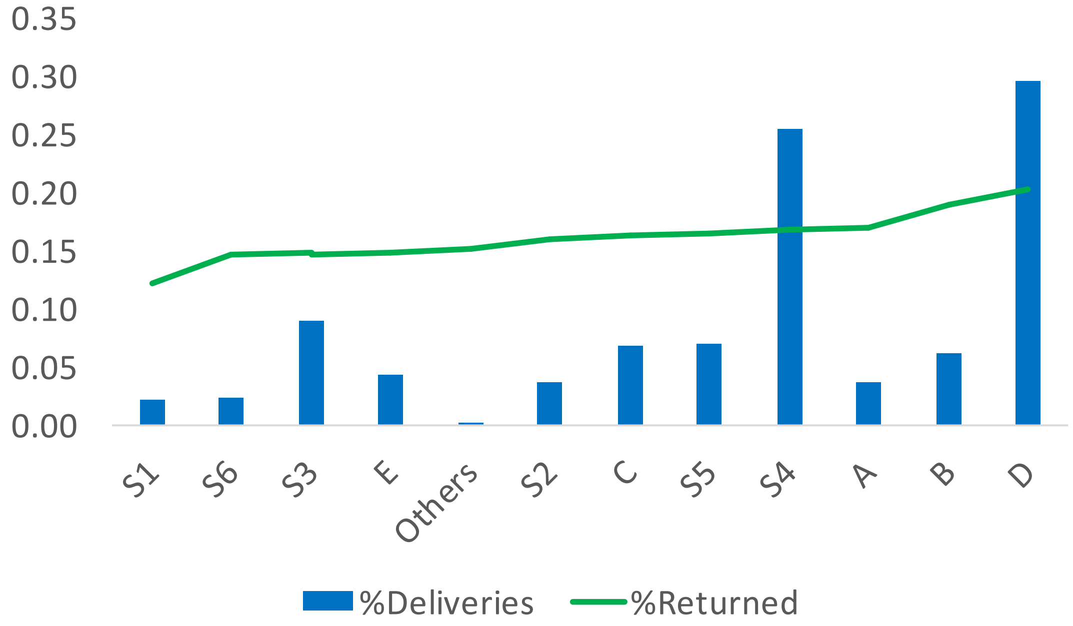
Findings :

- C, E, F, Medium and Lontailed clients have high return rates even though their contribution to the total COD shipments are less.
- Medium contains 15 Clients with contribution between 0.75% to 2.5% and Longtailed contains 1800 Clients with contribution less than 075%

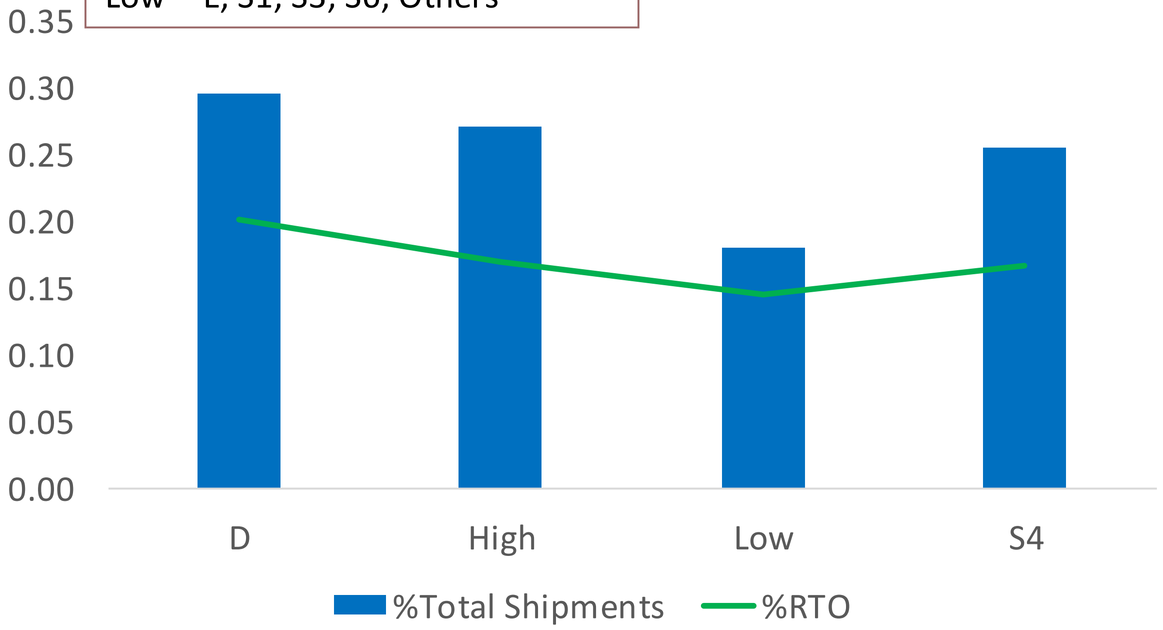
Findings:

- Buckets are divided on the basis of return rates of each client
- Low contains clients with return rate less than 13%, Medium between 13% to 19% and High more than 19%

Zones- Volume Distribution and Return Rates



High – A, B, C, S2, S4, S5
Low – E, S1, S3, S6, Others



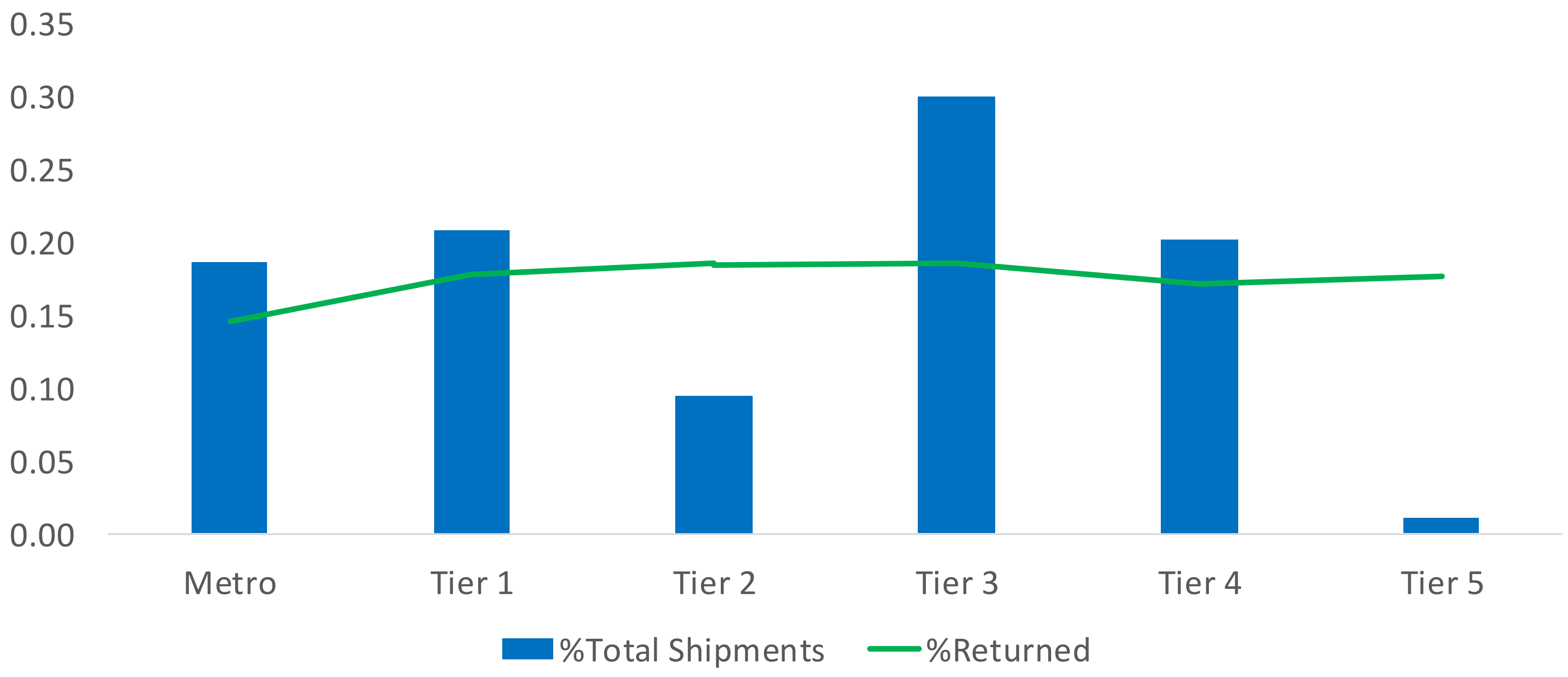
Findings:

- Maximum contribution to the total shipments - D, S4, at the same time they even have high return rates

Findings:

- Buckets are formed on the basis of their return rate and volume distribution.

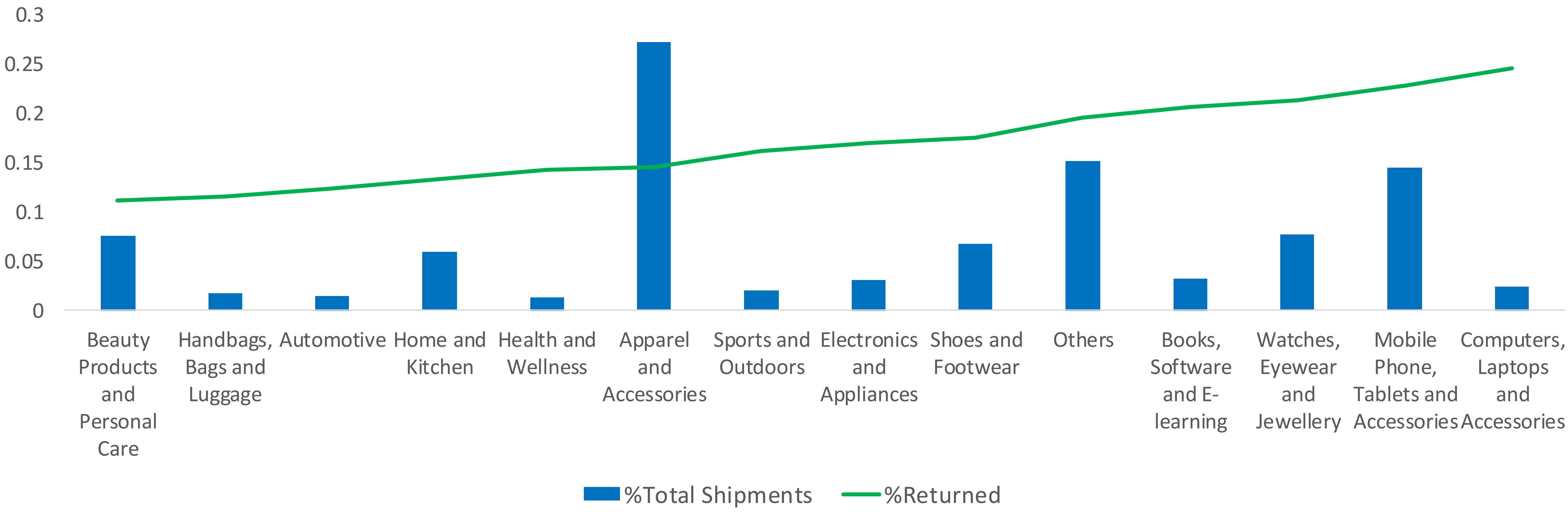
Tier - Volume Distribution and Return Rates



Findings:

- Metros have less return rate than all other tiers.
- Tier 3 has the highest contribution to the total shipments.

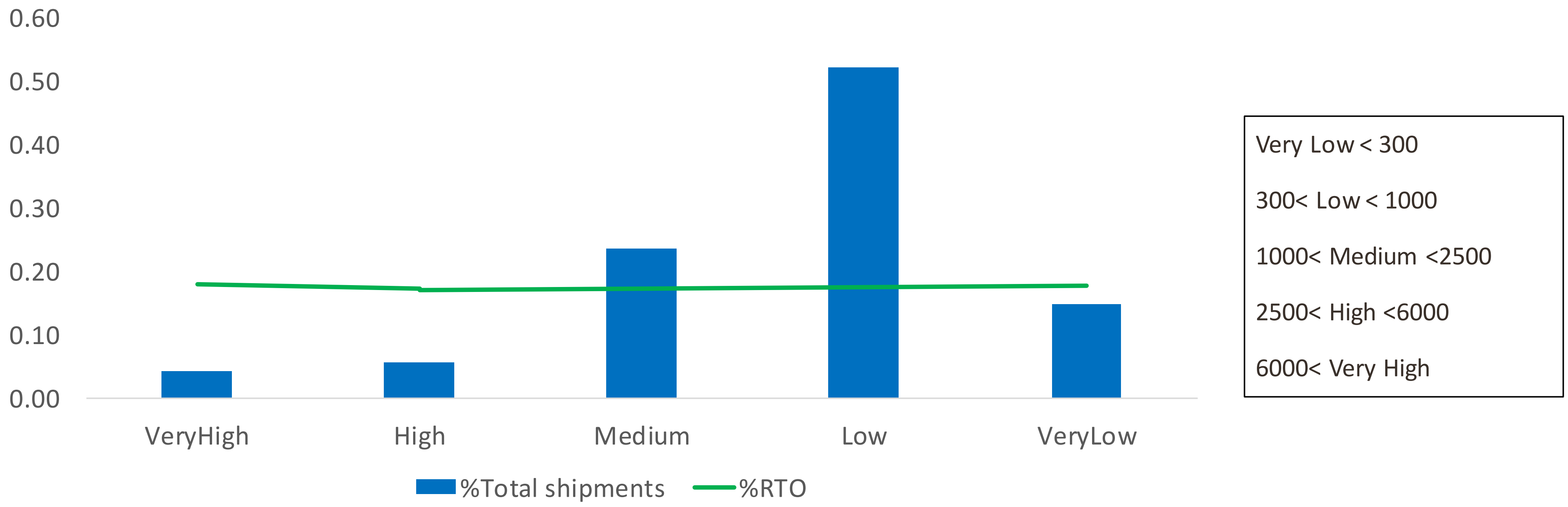
Categories - Volume Distribution and Return Rates



Findings:

- Categories like Computers, Mobile phones, Watches and books have high return rates.
- Apparel and accessories have the highest contribution to the total shipments

COD Amount - Volume Distribution and Return Rates

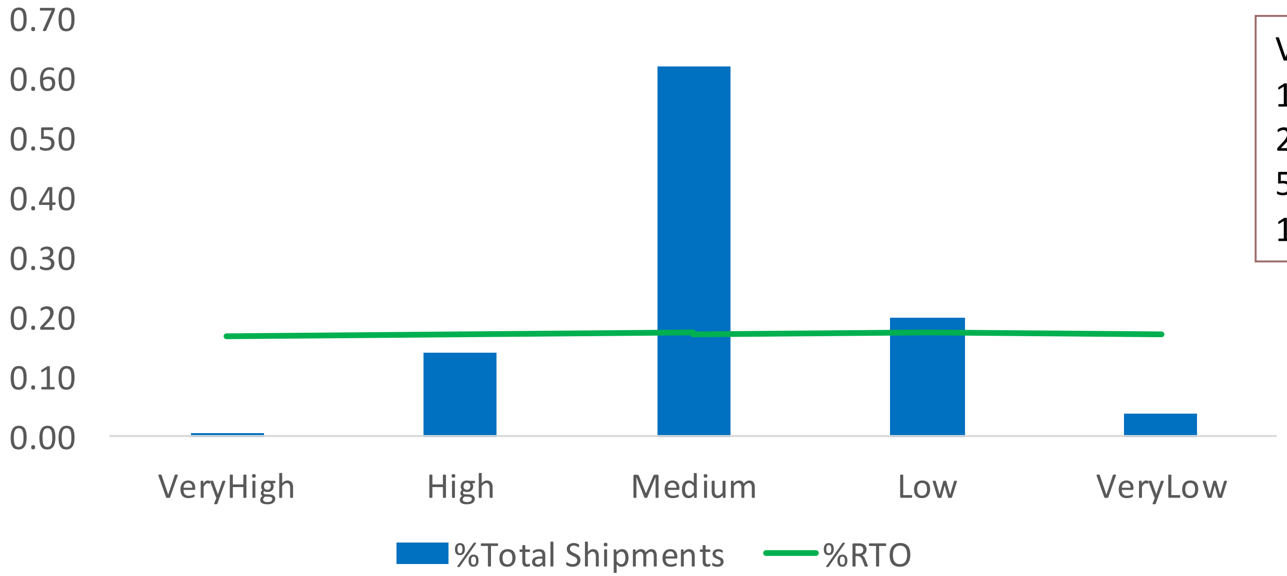


Findings:

COD amount has no effect on the return rate

Time Taken / Delay - Volume Distribution and Return Rates

Time Taken for shipment to reach the customer



Very Low < 1 day
1 day< Low < 2 days
2 days < Medium < 5 days
5 days < High < 10 days
10 days < Very High

Affect of delay on return rate

S.No.	Delay	Total	Returned	%Returned	% Total Shipments
1	FALSE	4824272	830052	0.17	0.87
2	TRUE	699890	129621	0.19	0.13

Findings:

- Time taken has no effect on the return rate.
- Delay has small a impact on the return rate.

Model Building

Approach to model building

- The data set given is an unbalanced data set.
- 25% of the Delivered data was combined with the RTO data in order to have balanced data set (Undersampling).
- Next a random sample of half the data entries was taken from the combined balanced dataset
- It was divided in the ratio 70:30 into training and testing dataset

Variables considered for First iteration

- COD amount
- Tier
- Delay
- Time taken
- Client buckets based on COD return %age (High, Medium, Low)
- Client buckets based on volume distribution
- Category buckets based on COD return %age (High, Medium, Low)
- Zone buckets based on COD return %age (D,High,S4, Low)
- Fraud Phone Numbers

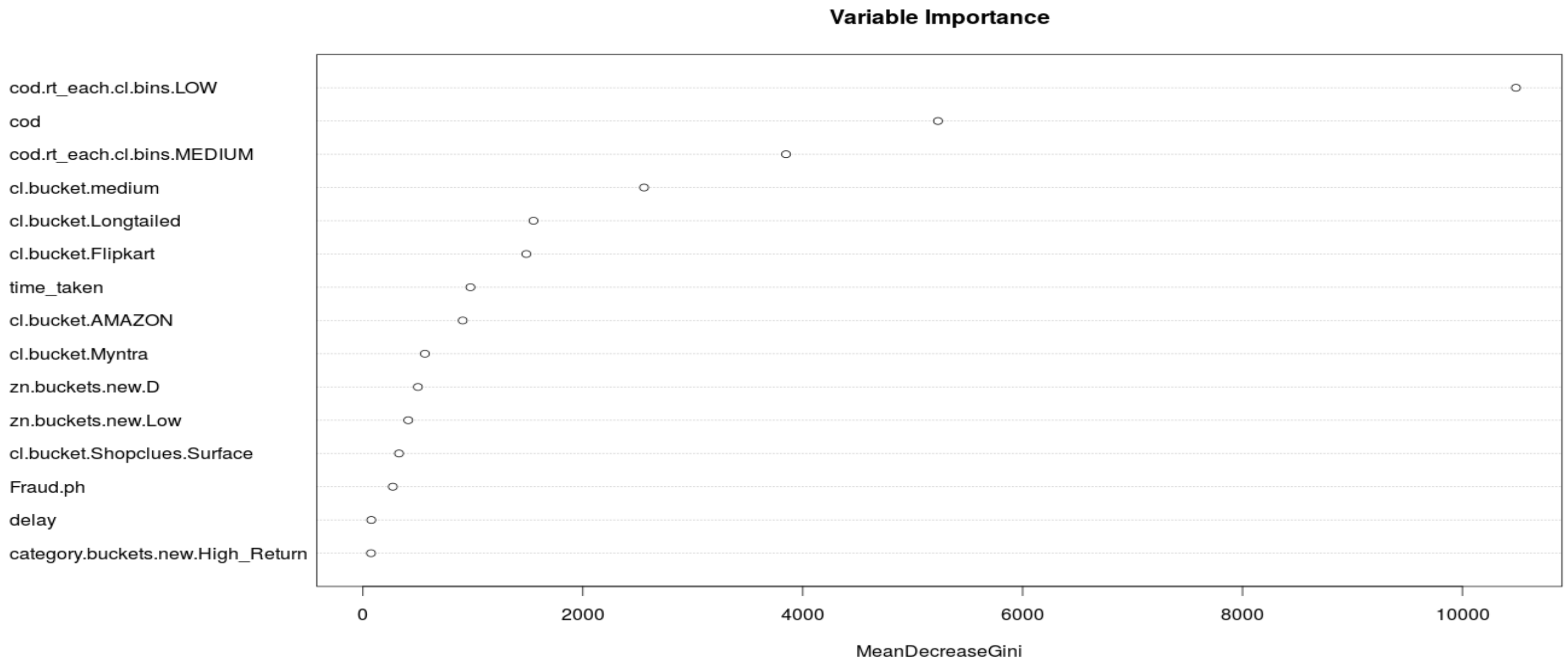
Model (First Iteration) – Logistic Regression

RTO.Log = glm(cs.ss ~ time_taken + cod + cod.rt_each.cl.bins.LOW + cod.rt_each.cl.bins.MEDIUM + category.buckets.new.High_Return + zn.buckets.new.D + zn.buckets.new.Low + cl.bucket.Flipkart + cl.bucket.AMAZON + +cl.bucket.Longtailed + cl.bucket.medium + cl.bucket.Myntra + cl.bucket.Shopclues.Surface + Fraud.ph + delay, data = df_sample.train, family = binomial)

```
Coefficients:
                                  Estimate Std. Error z value Pr(>|z|)    
(Intercept)                      4.979e-01  4.640e-02  10.730  < 2e-16 ***
time_taken                       2.815e-02  5.094e-03   5.527 3.25e-08 ***
cod                              2.993e-05  3.292e-06   9.092  < 2e-16 ***
cod.rt_each.cl.bins.LOW         -1.719e+00  2.879e-02 -59.693  < 2e-16 ***
cod.rt_each.cl.bins.MEDIUM      -9.247e-01  2.837e-02 -32.597  < 2e-16 ***
category.buckets.new.High_Return -1.855e-02  1.584e-02  -1.171  0.24159    
zn.buckets.new.D                -5.940e-02  2.336e-02  -2.543  0.01099 *  
zn.buckets.new.Low              -1.278e-01  2.492e-02  -5.129 2.91e-07 ***
cl.bucket.Flipkart              -9.926e-01  5.253e-02 -18.895  < 2e-16 ***
cl.bucket.AMAZON                -2.268e-01  3.433e-02  -6.606 3.96e-11 ***
cl.bucket.Longtailed             1.067e-01  3.865e-02   2.760  0.00577 ** 
cl.bucket.medium                 7.480e-02  3.973e-02   1.883  0.05975 .  
cl.bucket.Myntra                 2.995e-01  5.732e-02   5.225 1.74e-07 ***
cl.bucket.Shopclues.Surface     -7.221e-01  5.090e-02 -14.187  < 2e-16 ***
Fraud.phTRUE                    -2.735e-02  3.718e-02  -0.736  0.46190    
delayTRUE                       -1.726e-03  2.494e-02  -0.069  0.94482    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model(First Iteration) – Random Forest

RTO.rf = randomForest(cs.ss ~ time_taken + cod + cod.rt_each.cl.bins.LOW + cod.rt_each.cl.bins.MEDIUM + category.buckets.new.High_Return + zn.buckets.new.D + zn.buckets.new.Low + cl.bucket.Flipkart + cl.bucket.AMAZON + cl.bucket.Myntra + cl.bucket.Longtailed + cl.bucket.medium+cl.bucket.Shopclues.Surface + delay + Fraud.ph, data = df_sample.train, nodesize = 20, ntree = 200,trControl = control)



Final Variables Considered

- Time taken
- COD amount
- Client bucket (having low return rate)
- Client bucket (having medium return rate)
- High return categories
- Zn(D)
- Zn(Low)
- Client (Flipkart, Amazon, Myntra, Shopclues, Longtailed)
- Delay
- Fraud phone no.

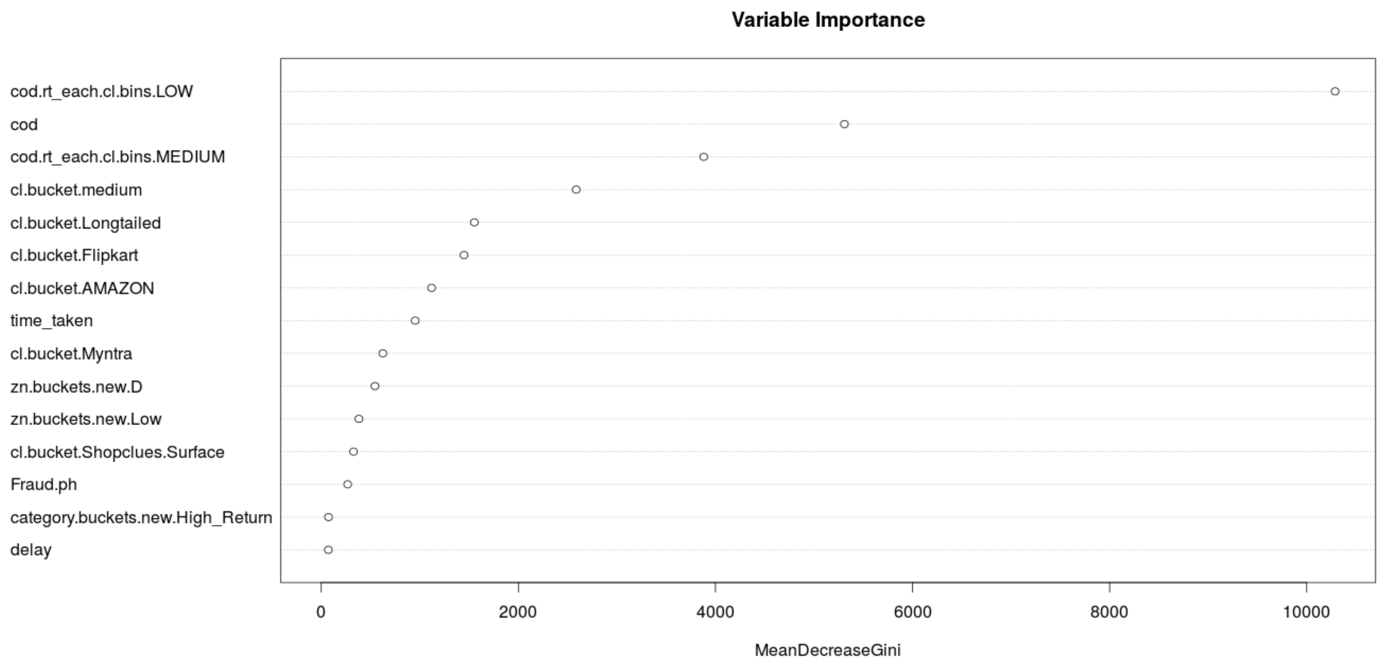
Model (Final Iteration) – Logistic Regression

RTO.Log = glm(cs.ss ~ time_taken + cod + cod.rt_each.cl.bins.LOW + cod.rt_each.cl.bins.MEDIUM + category.buckets.new.High_Return + zn.buckets.new.D + zn.buckets.new.Low + cl.bucket.Flipkart + cl.bucket.AMAZON +cl.bucket.Myntra+cl.bucket.Longtailed + cl.bucket.Shopclues.Surface + delay + Fraud.ph , data = df_sample.train, family = binomial)

```
Coefficients:
                                 Estimate Std. Error  z value Pr(>|z|)    
(Intercept)                     6.039e-01  8.427e-03   71.670  < 2e-16 ***
time_taken                      1.331e-02  1.582e-03    8.415  < 2e-16 ***
cod                             3.292e-05  1.034e-06   31.827  < 2e-16 ***
cod.rt_each.cl.bins.LOW        -1.687e+00  9.044e-03 -186.591  < 2e-16 ***
cod.rt_each.cl.bins.MEDIUM     -9.361e-01  7.097e-03 -131.897  < 2e-16 ***
category.buckets.new.High_Return -2.824e-02  5.001e-03   -5.646 1.64e-08 ***
zn.buckets.new.D               -4.695e-02  7.203e-03   -6.518 7.14e-11 ***
zn.buckets.new.Low             -1.410e-01  7.510e-03  -18.778  < 2e-16 ***
cl.bucket.Flipkart             -1.089e+00  1.027e-02 -106.045  < 2e-16 ***
cl.bucket.AMAZON               -2.885e-01  9.012e-03  -32.016  < 2e-16 ***
cl.bucket.Myntra                2.759e-01  1.441e-02   19.148  < 2e-16 ***
cl.bucket.Longtailed            3.739e-02  6.747e-03    5.542 2.99e-08 ***
cl.bucket.Shopclues.Surface    -7.459e-01  1.152e-02  -64.721  < 2e-16 ***
delayTRUE                       1.628e-02  7.880e-03    2.066 0.038837 *  
Fraud.phTRUE                   -4.226e-02  1.171e-02   -3.609 0.000307 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Model(Final Iteration) – Random Forest

RTO.rf = randomForest(cs.ss ~ time_taken + cod + cod.rt_each.cl.bins.LOW + cod.rt_each.cl.bins.MEDIUM + category.buckets.new.High_Return + zn.buckets.new.D + zn.buckets.new.Low + cl.bucket.Flipkart + cl.bucket.AMAZON + cl.bucket.Longtailed + cl.bucket.medium + cl.bucket.Myntra + cl.bucket.Shopclues.Surface + Fraud.ph + delay, data = df_sample.train, nodesize = 20, ntree = 200,trControl = control)



Confusion matrix –

- Test Dataset

TRUE - RTO
FALSE - Delivered

Logistic Regression

	Predicted	
Actual	FALSE	TRUE
FALSE	139345	31824
TRUE	83342	60609

Accuracy = 64%
Precision = 67%
Sensitivity = 42%
Specificity = 18%

Random Forest

	Predicted	
Actual	FALSE	TRUE
FALSE	142163	29006
TRUE	84637	59314

Accuracy - 64%
Precision - 67%
Sensitivity - 41%
Specificity - 16%

- Training Data

TRUE - RTO
FALSE - Delivered

Logistic Regression

	Predicted	
Actual	FALSE	TRUE
FALSE	93811	77358
TRUE	49069	94882

Accuracy = 60%
Precision = 55%
Sensitivity = 66%
Specificity = 45%

Random Forest

	Predicted	
Actual	FALSE	TRUE
FALSE	331227	68166
TRUE	197900	137985

Accuracy – 64%
Precision – 67%
Sensitivity – 41%
Specificity – 17%

THANK YOU