TELECOM CUSTOMER CHURN PREDICTION

REPRESENTED BY: PREETI, CHETNA & PRAKATHI

INTRODUCTION

CHURN PREDICTION - Identifying At-risk Customers Who Are Likely To Cancel Their Subscriptions Or Close/Abandon Their Accounts. A Churn Model Works By Passing Previous Customer Data Through A Machine Learning Model To Identify The Connections Between Features And Targets And Make Predictions About New Customers.

CHURN - is the measure of how many customers stop using a product. This can be measured based on actual usage or failure to renew (when the product is sold using a subscription model). Often evaluated for a specific period of time, there can be a monthly, quarterly, or annual churn rate.

BUSINESS PROBLEM OVERVIEW:-

In The Telecom Industry, Customers Are Able To Choose From Multiple Service Providers And Actively Switch From One Operator To Another. In This Highly Competitive Market, The Telecommunications Industry Experiences An Average Of 15-25% Annual Churn Rate. Given The Fact That It Costs 5-10 Times More To Acquire A New Customer Than To Retain An Existing One, Customer Retention Has Now Become Even More Important Than Customer Acquisition.

For Many Incumbent Operators, Retaining High Profitable Customers Is The Number One Business Goal.

To Reduce Customer Churn, Telecom Companies Need To Predict Which Customers Are At High Risk Of Churn.

DATA SET DESCRIPTION

Importing Libraries.
Source data is in CSV format.
Data set contains
Dependent Target variable: “Churn”
Churn Rate (Baseline) is 26.5%.

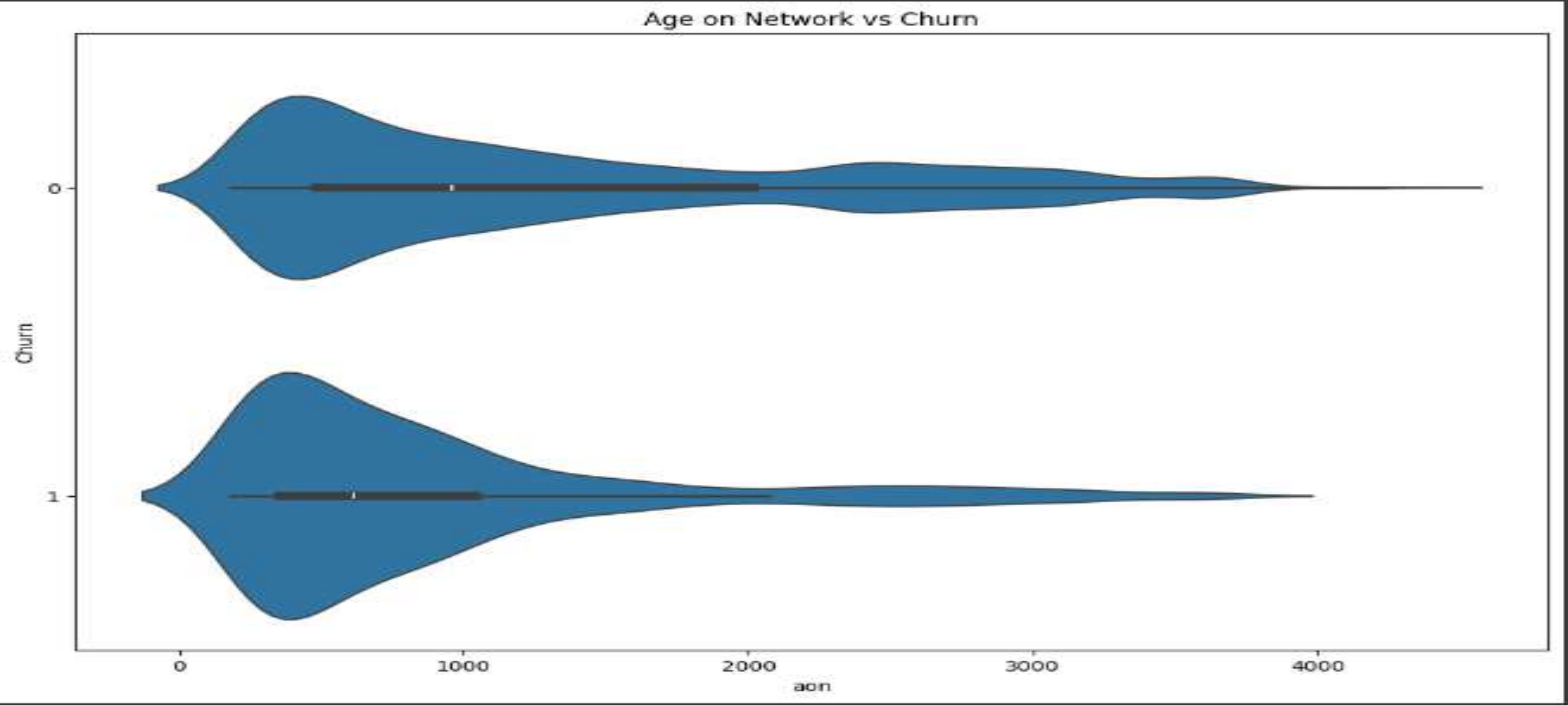
EDA

Data Visualizing Using Sea Born And Matplotlib
Exploratory Data Analysis Is An Approach To Analyze Data Sets & To Summarize Their Main Characteristics, Often With Visual Methods.

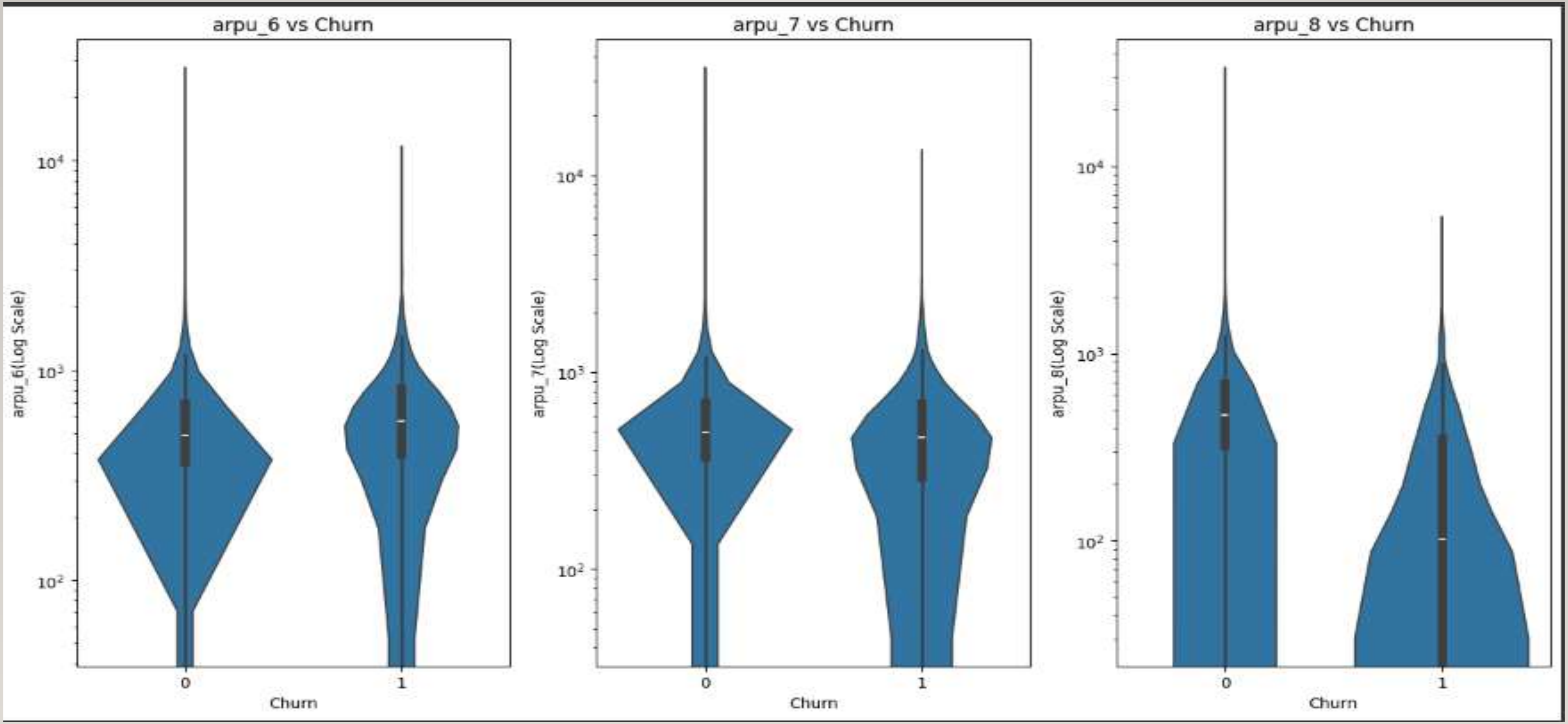
A Statistical Model Can Be Used Or Not, But Primarily EDA Is For Seeing What The Data Can Tell Us Beyond The Formal Modelling Or Hypothesis.

Age on Network

```
5]  plt.figure(figsize=(12,8))
    sns.violinplot(x='aon', y='Churn', data=data)
    plt.title('Age on Network vs Churn')
    plt.show()
```



```python
columns = ['arpu_6','arpu_7','arpu_8']
num_univariate_analysis(columns,'log')
```



```
columns = ['monthly_2g_6', 'monthly_2g_7', 'monthly_2g_8']
cat_univariate_analysis(columns)
```

Customers who churned (Churn : 1)

	monthly_2g_6	count	percent	cumulative_count	cumulative_percent
0	0	166	93.2584	166	93.2584
1	1	12	6.74157	178	100

	monthly_2g_7	count	percent	cumulative_count	cumulative_percent
0	0	170	95.5056	170	95.5056
1	1	8	4.49438	178	100

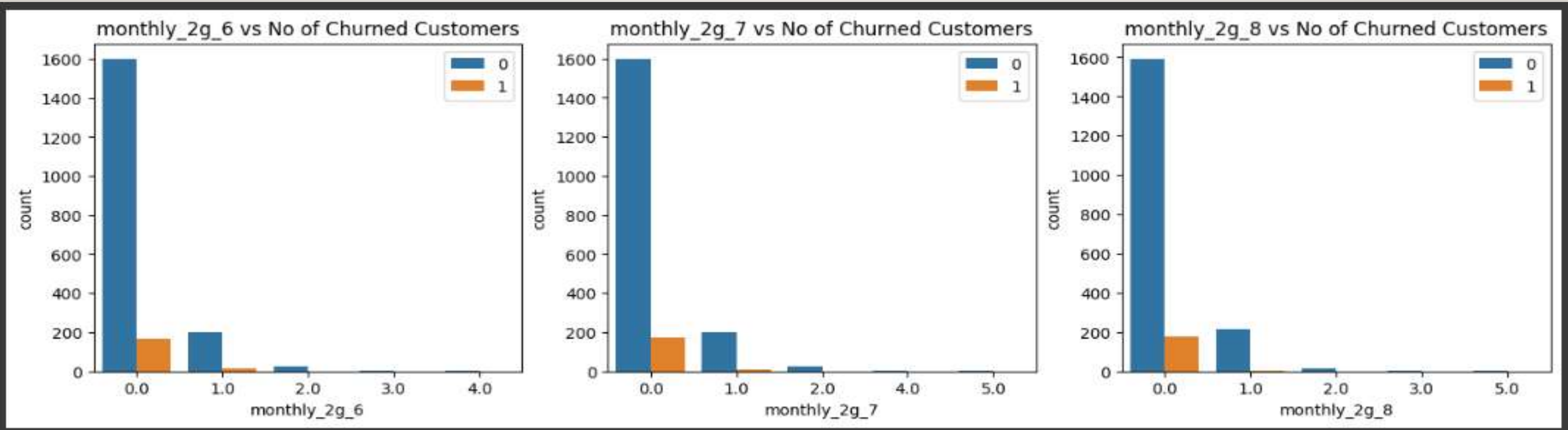
	monthly_2g_8	count	percent	cumulative_count	cumulative_percent
0	0	174	97.7528	174	97.7528
1	1	4	2.24719	178	100

Customers who did not churn (Churn : 0)

	monthly_2g_6	count	percent	cumulative_count	cumulative_percent
0	0	1598	87.6096	1598	87.6096
1	1	198	10.8553	1796	98.4649
2	2	26	1.42544	1822	99.8904
3	4	1	0.0548246	1823	99.9452
4	3	1	0.0548246	1824	100

	monthly_2g_7	count	percent	cumulative_count	cumulative_percent
0	0	1596	87.5	1596	87.5
1	1	199	10.9101	1795	98.4101
2	2	27	1.48026	1822	99.8904
3	5	1	0.0548246	1823	99.9452
4	4	1	0.0548246	1824	100

	monthly_2g_8	count	percent	cumulative_count	cumulative_percent
0	0	1590	87.1711	1590	87.1711
1	1	216	11.8421	1806	99.0132
2	2	15	0.822368	1821	99.8355
3	3	2	0.109649	1823	99.9452
4	5	1	0.0548246	1824	100



```python
columns = ['monthly_3g_6', 'monthly_3g_7', 'monthly_3g_8']
cat_univariate_analysis(columns)
```

Customers who churned (Churn : 1)

	monthly_3g_6	count	percent	cumulative_count	cumulative_percent
0	0	159	89.3258	159	89.3258
1	1	15	8.42697	174	97.7528
2	2	3	1.68539	177	99.4382
3	3	1	0.561798	178	100

	monthly_3g_7	count	percent	cumulative_count	cumulative_percent
0	0	166	93.2584	166	93.2584
1	1	8	4.49438	174	97.7528
2	2	4	2.24719	178	100

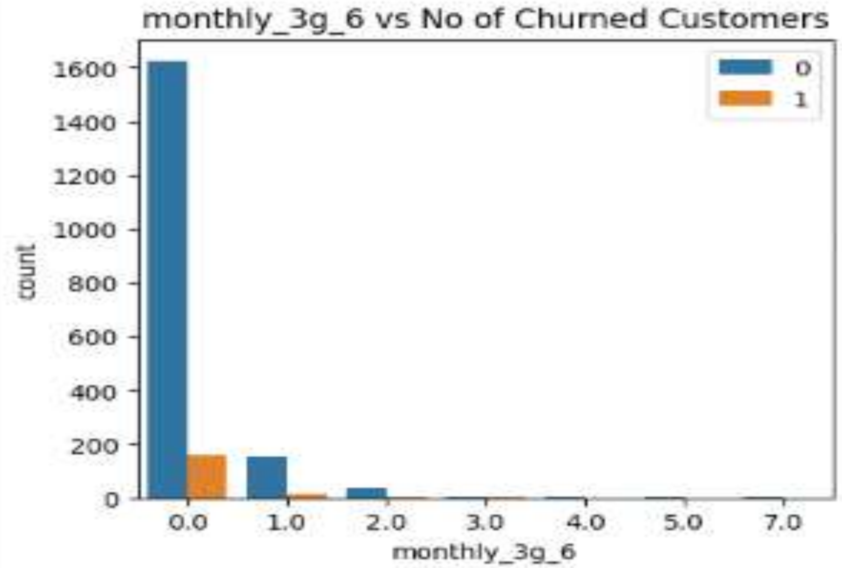
	monthly_3g_8	count	percent	cumulative_count	cumulative_percent
0	0	173	97.191	173	97.191
1	1	4	2.24719	177	99.4382
2	3	1	0.561798	178	100

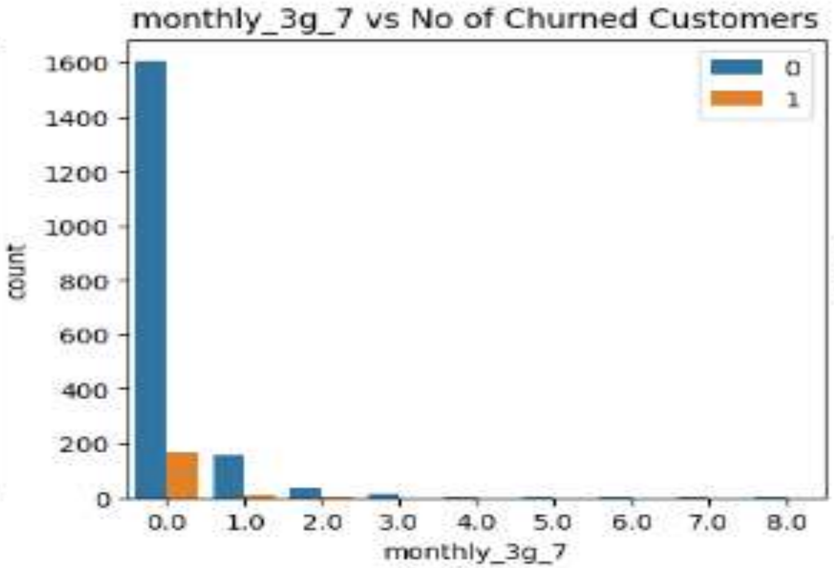
Customers who did not churn (Churn : 0)

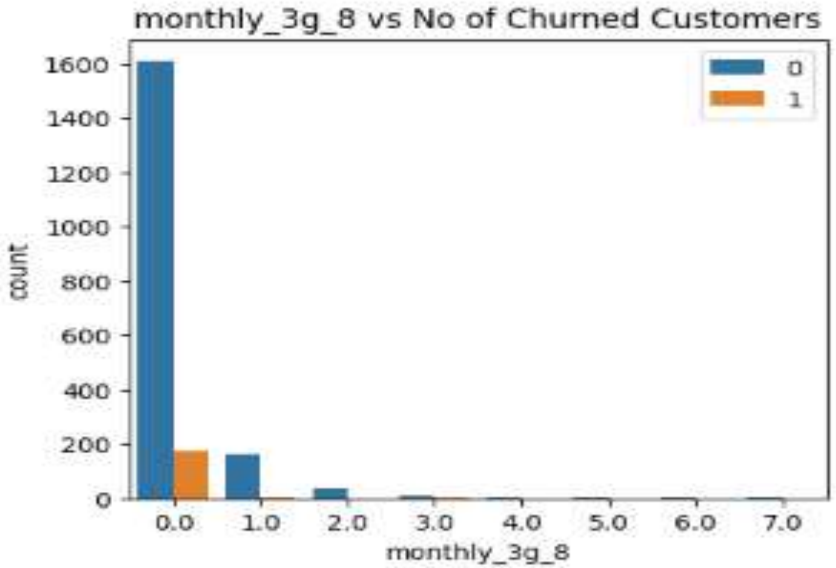
	monthly_3g_6	count	percent	cumulative_count	cumulative_percent
0	0	1623	88.9803	1623	88.9803
1	1	155	8.49781	1778	97.4781
2	2	36	1.97368	1814	99.4518
3	3	5	0.274123	1819	99.7259
4	5	2	0.109649	1821	99.8355
5	4	2	0.109649	1823	99.9452
6	7	1	0.0548246	1824	100

	monthly_3g_7	count	percent	cumulative_count	cumulative_percent
0	0	1605	87.9934	1605	87.9934
1	1	156	8.55263	1761	96.5461
2	2	38	2.08333	1799	98.6294
3	3	13	0.712719	1812	99.3421
4	5	4	0.219298	1816	99.5614
5	4	4	0.219298	1820	99.7807
6	6	2	0.109649	1822	99.8904
7	8	1	0.0548246	1823	99.9452
8	7	1	0.0548246	1824	100

	monthly_3g_8	count	percent	cumulative_count	cumulative_percent



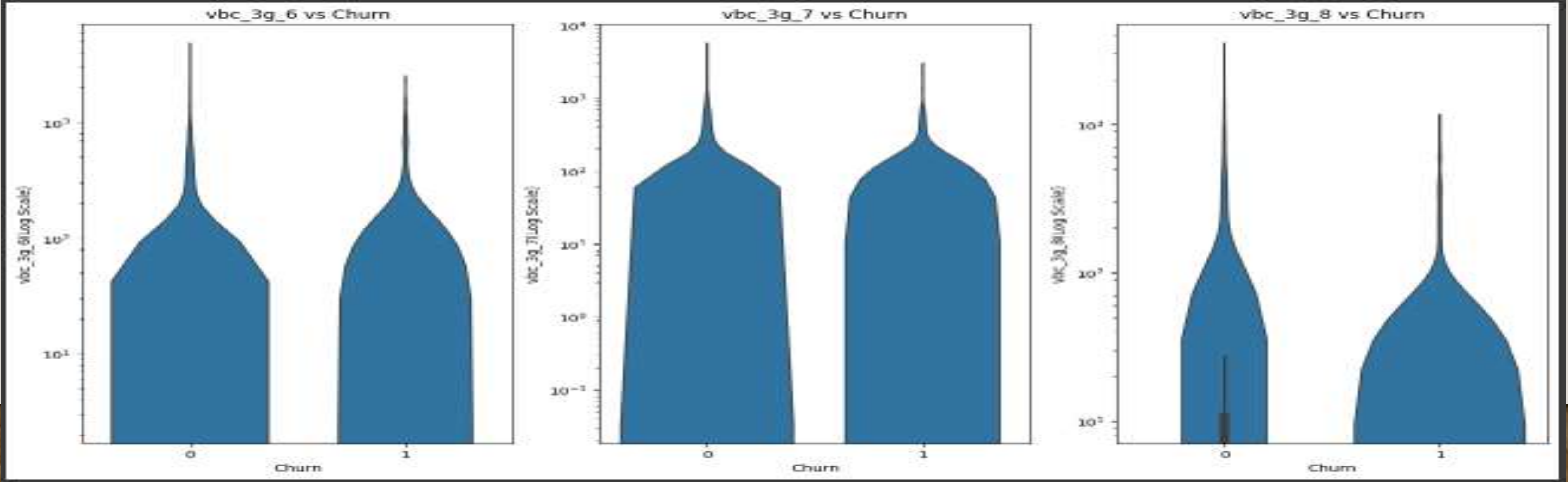




```
columns = [ 'vbc_3g_6', 'vbc_3g_7','vbc_3g_8']
num_univariate_analysis(columns, 'log')
```

```
Customers who churned (Churn : 1)
          vbc_3g_6     vbc_3g_7     vbc_3g_8
count   178.000000   178.000000   178.000000
mean     96.901348    74.042697    31.212697
std     309.296398   306.618233   130.736714
min       0.000000     0.000000     0.000000
25%       0.000000     0.000000     0.000000
50%       0.000000     0.000000     0.000000
75%       0.000000     0.000000     0.000000
max    2307.100000  2788.070000  1079.790000

Customers who did not churn (Churn : 0)
          vbc_3g_6     vbc_3g_7     vbc_3g_8
count  1824.000000  1824.00000  1824.000000
mean    113.048953   125.01199   123.956491
std     360.252809   397.23242   338.913780
min       0.000000     0.00000     0.000000
25%       0.000000     0.00000     0.000000
50%       0.000000     0.00000     0.000000
75%       0.000000     0.00000    10.887500
max    4688.590000  5486.80000  3384.520000
```



DATA EXPLORATION

Variables, “Tenure” and “MonthlyCharges”, both are positively corelated to “TotalCharges” and can be identified approximately as “TotalCharges = Tenure x MonthlyCharges”.
In the scatterplot matrix, red dots represent the records which have churn as “no” and blue dots represent records with churn as “yes”.

Missing Values: “Total Charges” has 11 missing values.

Outliers: There are no outliers in the dataset.

CONCLUSION :

The Best Model To Predict The Churn Is Observed To Be Random Forest Based On The Accuracy As Performance Measure.

The Incoming Calls (With Local Same Operator Mobile/Other Operator Mobile/Fixed Lines, STD Or Special) Plays A Vital Role In Understanding The Possibility Of Churn. Hence, The Operator Should Focus On Incoming Calls Data And Has To Provide Some Kind Of Special Offers To The Customers Whose Incoming Calls Turning Lower.

DETAILS:

After Cleaning The Data, We Broadly Employed Three Models As Mentioned Below Including Some Variations Within These Models In Order To Arrive At The Best Model In Each Of The Cases.

LOGISTIC REGRESSION :

Logistic Regression with RFE Logistic regression with PCA Random Forest For each of these models, the summary of performance measures are as

FOLLOWS:

Logistic Regression

. Train Accuracy : ~79%

. Test Accuracy : ~80%

Logistic regression with PCA
. Train Accuracy : ~91%
. Test Accuracy : ~92%

Decision Tree with PCA:
. Train Accuracy : ~93%
. Test Accuracy : ~92%

Random Forest with PCA:
. Train Accuracy :~ 91%
. Test Accuracy :~ 92%


THANK YOU