BDA601-BIG DATA AND ANALYTICS

Visualisation and Model Development

NILAY ALTINAY
A00065797 / TORRENS UNIVERSITY

Table of Contents

THE PROBLEM STATEMENT

Our most important problem is actually the question of why we lost our customers. In this circumstance, we have encountered negativities. Some of them are lost data or data with no value. In this report, all these problems are answered with graphs and ratios.

HANDLING MISISNG VALUES

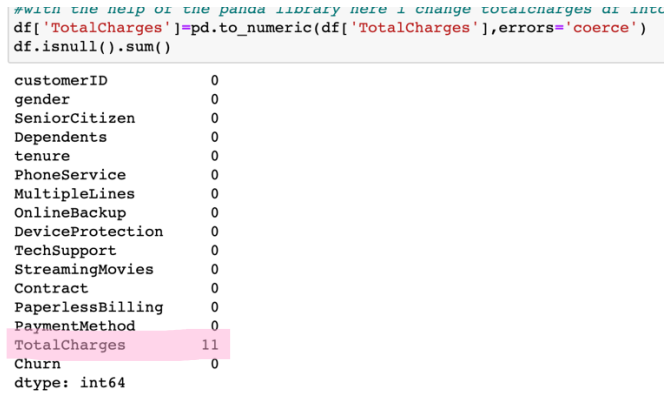
Numerous methods may be employed by data sciences to deal with missing data. On the one side, methods with robust handling of null data include randomized forests and KNN.

On the other side, we could have to handle the absence of data on our own. Remove the rows with empty values as the first typical response to missing value. Normally, every row with a blank value in any cell has that row erased. However, this frequently results in the removal of several rows and the subsequent loss of data. Hence, if there are few information sets, this strategy is often not employed (Insightsoftware, 2022).

One of the below methods could be used to fill in the missing value as the first method:

- Change it out with a constant number. When this strategy is implemented in consultation with the subject-matter expert for the data we are working with, it may be effective.
- Substitute the median or mean instead. When there is little data, this method works okay, but it introduces bias.
- Utilizing data from other cells, fill it out for empty values.
- Making a straightforward regression model is just another method for forecasting null values. Utilizing different columns, rows or cells in the whole dataset such as, tenure and services, we can forecast the null cells in this case. When building the forecasting models, we must take into account the situations where the input columns have missing values. Choosing only the features without missing values or selecting the rows without any cells with missing values is an easy approach to handle this.

According to our case study, the number of row of missing values are as in the figure below.



```
#with the help of the panda library here I change totalcharges dt into
df['TotalCharges']=pd.to_numeric(df['TotalCharges'],errors='coerce')
df.isnull().sum()
```

```
customerID          0
gender              0
SeniorCitizen       0
Dependents          0
tenure              0
PhoneService        0
MultipleLines       0
OnlineBackup        0
DeviceProtection    0
TechSupport         0
StreamingMovies     0
Contract            0
PaperlessBilling    0
PaymentMethod       0
TotalCharges       11
Churn               0
dtype: int64
```

Figure 1. Missing Values

There are missing values only in TotalCharges column in Telco data. In order to handle missing data, I could utilise my intuition to determine why the data is absent. This column is actually connected to the tenure column. In other words, as long as the customer stays in the company, the total amount of the invoices will increase. Accordingly, even mean or median values of TotalCharges attribute for these empty rows can be estimated. If we had not removed the MountlyCharges column in the data constraction batch, we could have calculated this and filled only 11 rows with ease. Additionally, we can substitute the missing value. This may be based purely on data from the column which has null data or may also replace on data from other columns in the whole data.

```
sns.barplot(x="importance", y="Feature", data=plot.sort_values(by="importance", ascending=False),palette = 'BuGn_r')
plt.title('Features Importance ')
plt.tight_layout()
plt.show()
```

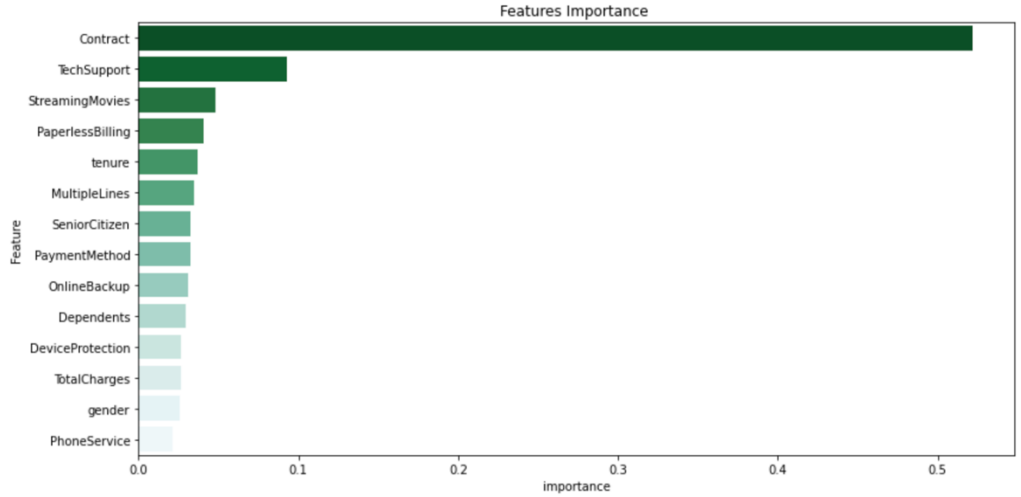


Figure 2. Feature Importance

In my decision tree, the most important attribute is Contract. It is alco can be seen in Figure 2. The TotalCharges attribute has the third to last importance. In addition to this TotalCharges attributes is Missing only 11 columns out of 7043 means that there is only %0.15. However, in this case, the strategy of filling 11 columns with a mean value in order to get rid of missing values may be the best method according to this scenario.

```
telco_data.select('TotalCharges', 'tenure').describe().show()
```

```
+-------+------------------+------------------+
|summary|      TotalCharges|            tenure|
+-------+------------------+------------------+
|  count|              7043|              7043|
|   mean|2283.3004408418697| 32.37114865824223|
| stddev| 2266.771361883145|24.559481023094442|
|    min|                 |                 0|
|    max|             999.9|                72|
+-------+------------------+------------------+
```

Figure 3. Summary of TotalCharges and tenure attribute.

INTERPRETATION OF CHURN ANALYSIS

I first showed it on the notebook to get a general idea of the data. For this purpose, I examined the values found in the attributes. Then I got the average idea by looking at the summaries.

The most important Feature is Contract. The **contract** variable indicates how long a client has been a member; there are three types of contracts: month-to-month (3875 consumers), two years (1695), and one year (1473). It is clear that customers with month-to-month contracts are more likely to leave their service. When we look at the graph which show us by Churn rate, we can easily forecast that Consumers with month-to-month agreements are more prone to churn. Therefore, one strategy to lower the churn rate would be to promote other contract types or gradually discontinue the monthly contract.

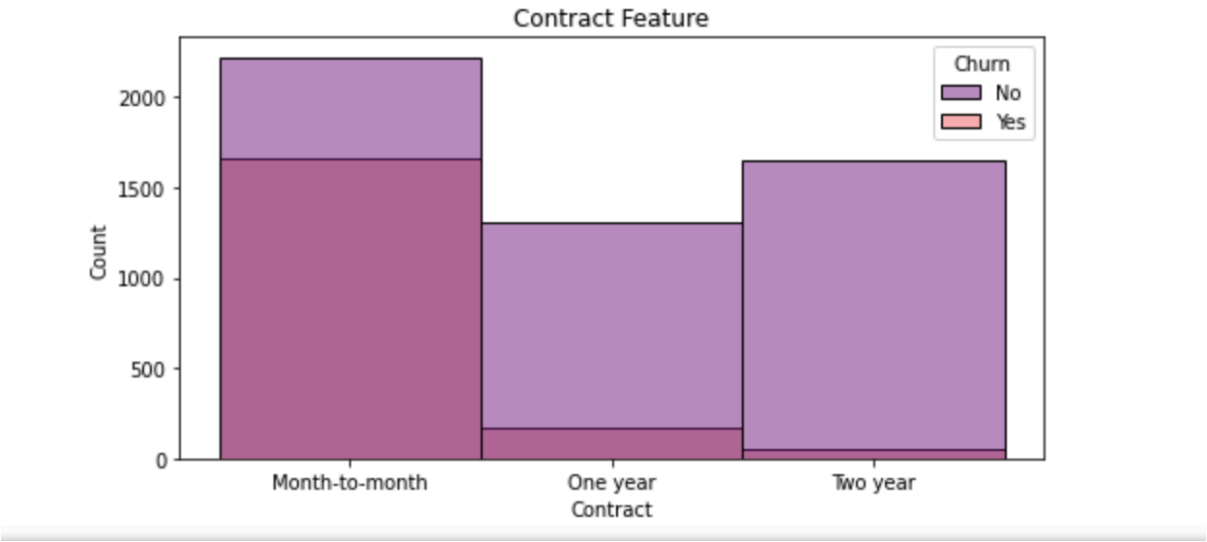


Figure 4. Contract attribute

On the other hand, we can see on the tenure by churn graph that we can see that lower tenure rate customers more likely to churn.

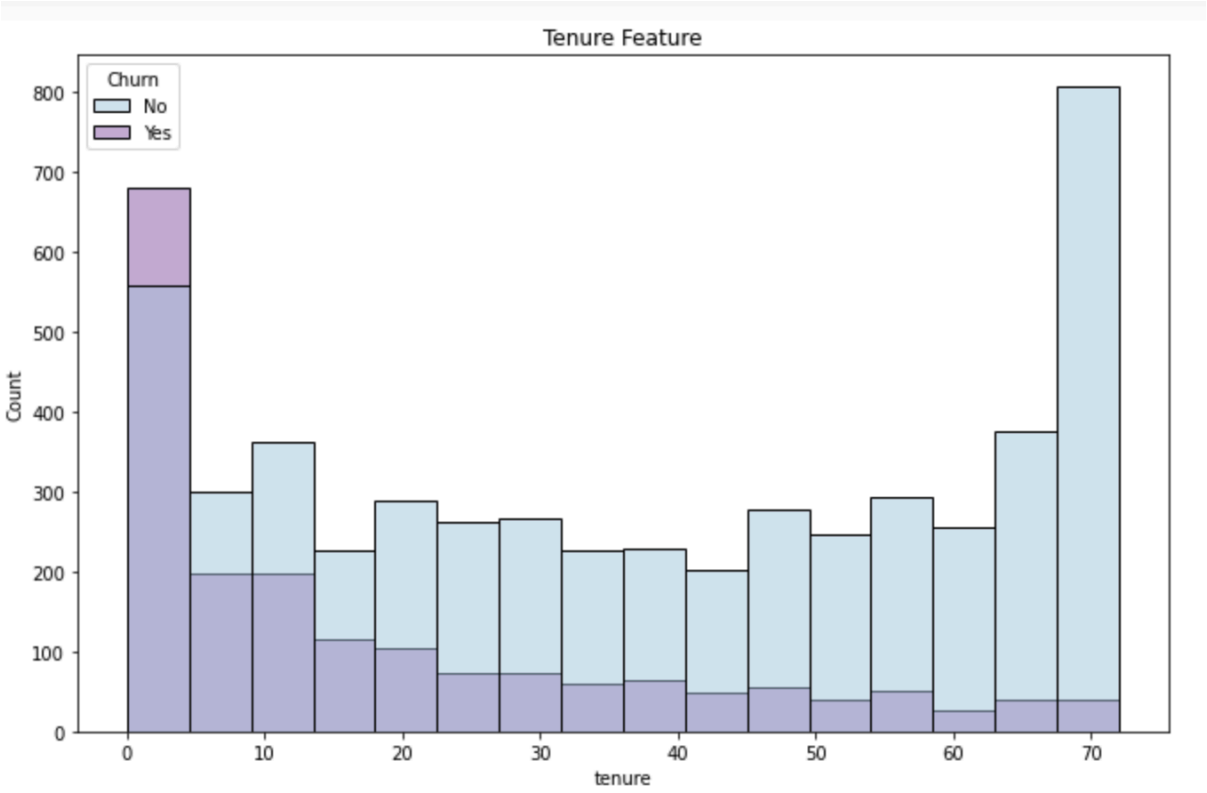


Figure 5. Tenure data

We also can see how many customers based on the tenure attribute as below figure;

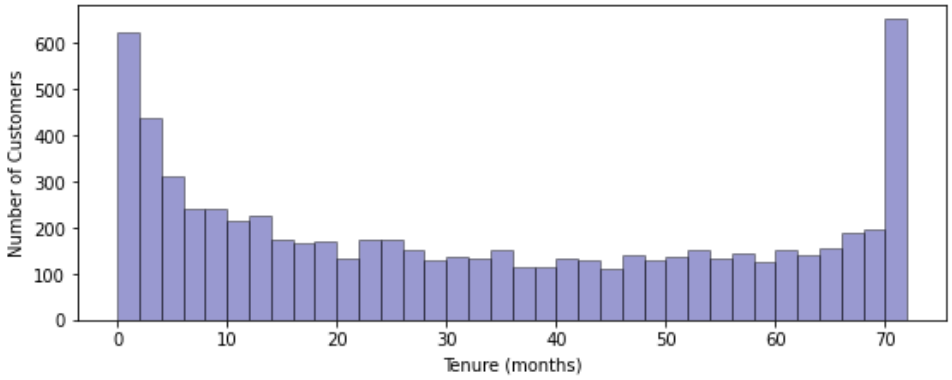


Figure 6. Tenure by number of Customer

By extracting graphs and linking them with churn, I got a deeper perspective. At this stage, I realized the importance of contracting and TechSupport attributes. I have seen that the customers who are on the contract stay with the company longer, and the customers who use the TechSupport service stay with the company longer.

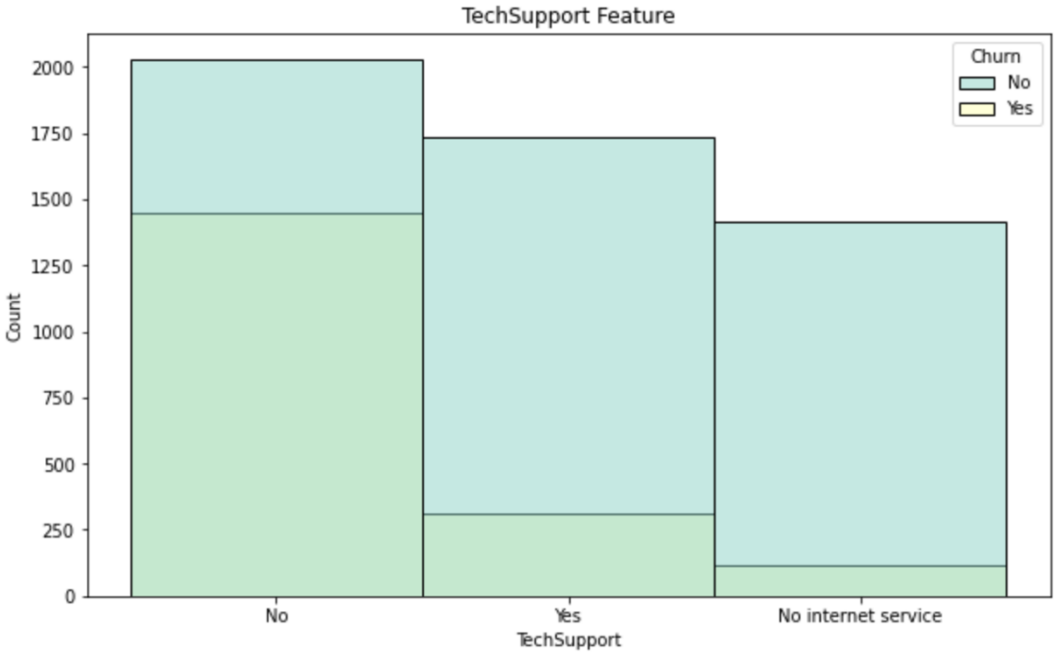


Figure 7. TechSupport Attribute

Based on the Dependencies attribute, whether a customer is a dependent or not is indicated by the binary variable in the Dependents feature.

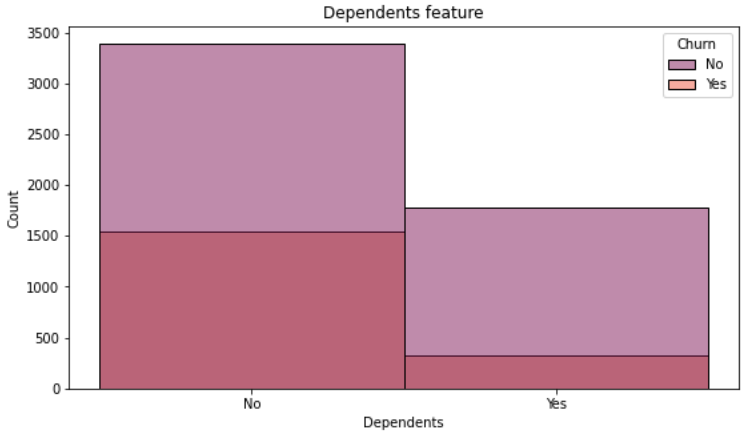


Figure 8. Dependent Attribute

We can see the correlation Matrix figure below and also can insight many commands based on it. Few of them are as following;

- There is strong relation between Contract and tenure as I also mention above.
- There is negative relation between tenure and Payment method attributes. The explanation is obvious.
- I can also see strong correlation between TechSupport and Contract attributes.

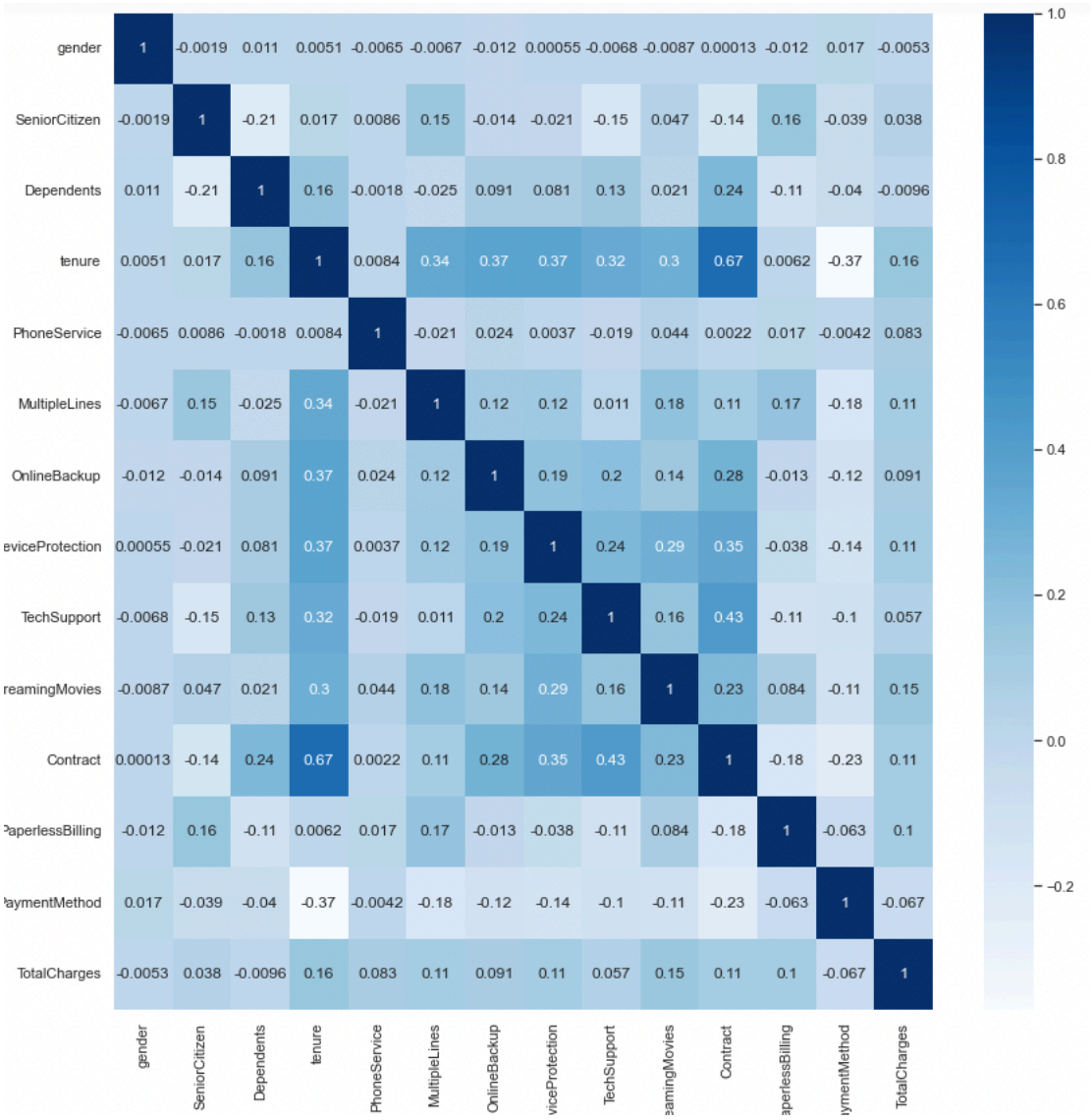


Figure 9. Correlation Matrix

The data analyse accuracy is 0.79. If I had done it according to feature importance, maybe I could have gotten a higher accuracy in my analysis. On the other hand, if I had filled the missing values using mean, this rate would have increased.

```
In [47]: preds=np.around(np.around(preds, decimals=0)) ## round values to 0 and 1
         print(classification_report(y_test,preds))
```

```
              precision    recall  f1-score   support

           0       0.83      0.90      0.86      1036
           1       0.64      0.49      0.55       373

    accuracy                           0.79      1409
   macro avg       0.73      0.70      0.71      1409
weighted avg       0.78      0.79      0.78      1409
```

Figure 10. Classification Report

REFERENCES

Insightsoftware. (2022). *How to Handle Missing Data Values While Data Cleaning*. Insightsoftware. https://insightsoftware.com/blog/how-to-handle-missing-data-values-while-data-cleaning/

Kaggle. (2018). *Telco Customer Churn*. Www.kaggle.com. https://www.kaggle.com/datasets/blastchar/telco-customer-churn