Final project for DS105:

WILL THE CUSTOMER CHURN?

SCENARIO

As one of the leading telco in the country, the company are proud to provide quality services that have been receiving satisfactory feedback from customers frequently. But from recent quarterly reports, there have been a drop in the customer base.

We are instructed by the management to create a model to predict if a customer will churn so that they are able to come up with solutions to keep hold of them.

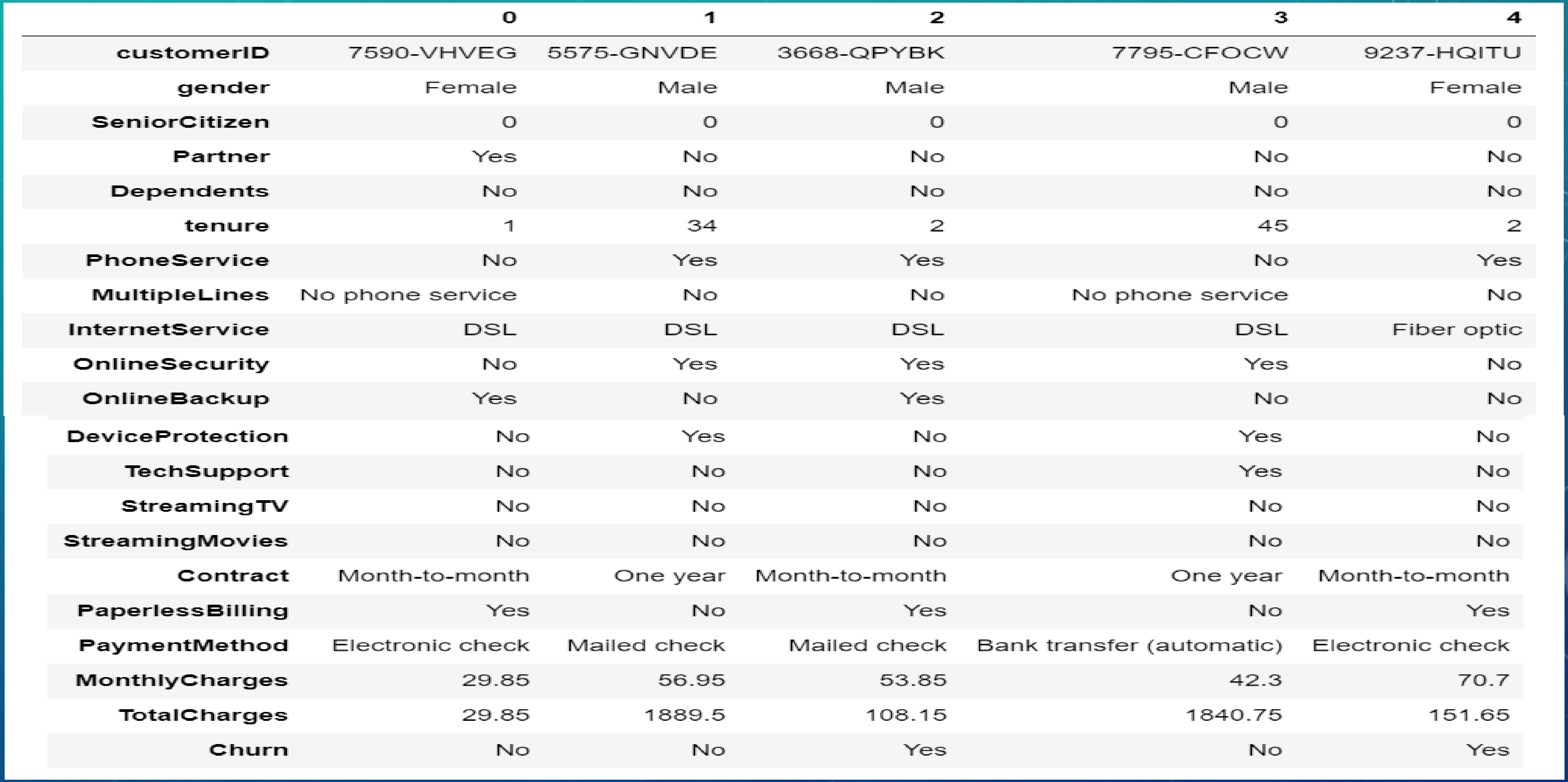
OBJECTIVES

- Understand the dataset
- Perform data cleaning
- Pre-process the dataset for our model
- Find the baseline of some models and choose the suitable model
- Evaluate the final model selected

CHALLENGES

- Check if the dataset have duplicate data and missing values
- To decide on the encoding method for categorical features
- If scaling is needed on the dataset
- What approach should we use if the dataset is imbalanced
- Since “Churn” (“Yes” or “No”) will be the label, our model will be a **Supervised Classification Model**. We will try a few models like Logistic Regression, Decision Trees, Random Forest, Support Vector Machine, Gradient Boost, XGBoost and select the best performed model for our problem
- Tune the hyperparameter of the selected model to improve the performance

UNDERSTAND THE DATA (TELCO CUSTOMER CHURN | KAGGLE)



	0	1	2	3	4
customerID	7590-VHVEG	5575-GNVDE	3668-QPYBK	7795-CFOCW	9237-HQITU
gender	Female	Male	Male	Male	Female
SeniorCitizen	0	0	0	0	0
Partner	Yes	No	No	No	No
Dependents	No	No	No	No	No
tenure	1	34	2	45	2
PhoneService	No	Yes	Yes	No	Yes
MultipleLines	No phone service	No	No	No phone service	No
InternetService	DSL	DSL	DSL	DSL	Fiber optic
OnlineSecurity	No	Yes	Yes	Yes	No
OnlineBackup	Yes	No	Yes	No	No
DeviceProtection	No	Yes	No	Yes	No
TechSupport	No	No	No	Yes	No
StreamingTV	No	No	No	No	No
StreamingMovies	No	No	No	No	No
Contract	Month-to-month	One year	Month-to-month	One year	Month-to-month
PaperlessBilling	Yes	No	Yes	No	Yes
PaymentMethod	Electronic check	Mailed check	Mailed check	Bank transfer (automatic)	Electronic check
MonthlyCharges	29.85	56.95	53.85	42.3	70.7
TotalCharges	29.85	1889.5	108.15	1840.75	151.65
Churn	No	No	Yes	No	Yes

UNDERSTAND THE DATA

- The dataset consists of 7043 rows and 21 features
- The feature “Churn” will be our label – “Yes” or “No”
- Features description:
  - customerID – ID number of customer
  - gender – the sex of the customer (“Male” or “Female”)
  - SeniorCitizen – whether customer is a senior citizen (0 or 1)
  - Partner – whether customer has partner (“Yes” or “No”)
  - Dependents – whether customer has dependents (“Yes” or “No”)

UNDERSTAND THE DATA

➢ Features description:

- Tenure – number of months customer stay with the company
- PhoneService – whether customer has phone service (“Yes” or “No”)
- MultipleLines – whether customer has multiple lines (“Yes”, “No” or “No phone service”)
- InternetService – customer internet service provider (“DSL”, “Fiber optic” or “No”)
- OnlineSecurity – whether customer has online sercurity (“Yes”, “No” or “No internet service”)

UNDERSTAND THE DATA

- ➢ Features description:
  - OnlineBackup – whether customer has online backup (“Yes”, “No” or “No internet service”)
  - DeviceProtection – whether customer has device protection (“Yes”, “No” or “No internet service”)
  - TechSupport – whether customer has tech support (“Yes”, “No” or “No internet service”)
  - StreamingTV – whether customer has streaming TV (“Yes”, “No” or “No internet service”)
  - StreamingMovies – whether customer has streaming movies (“Yes”, “No” or “No internet service”)

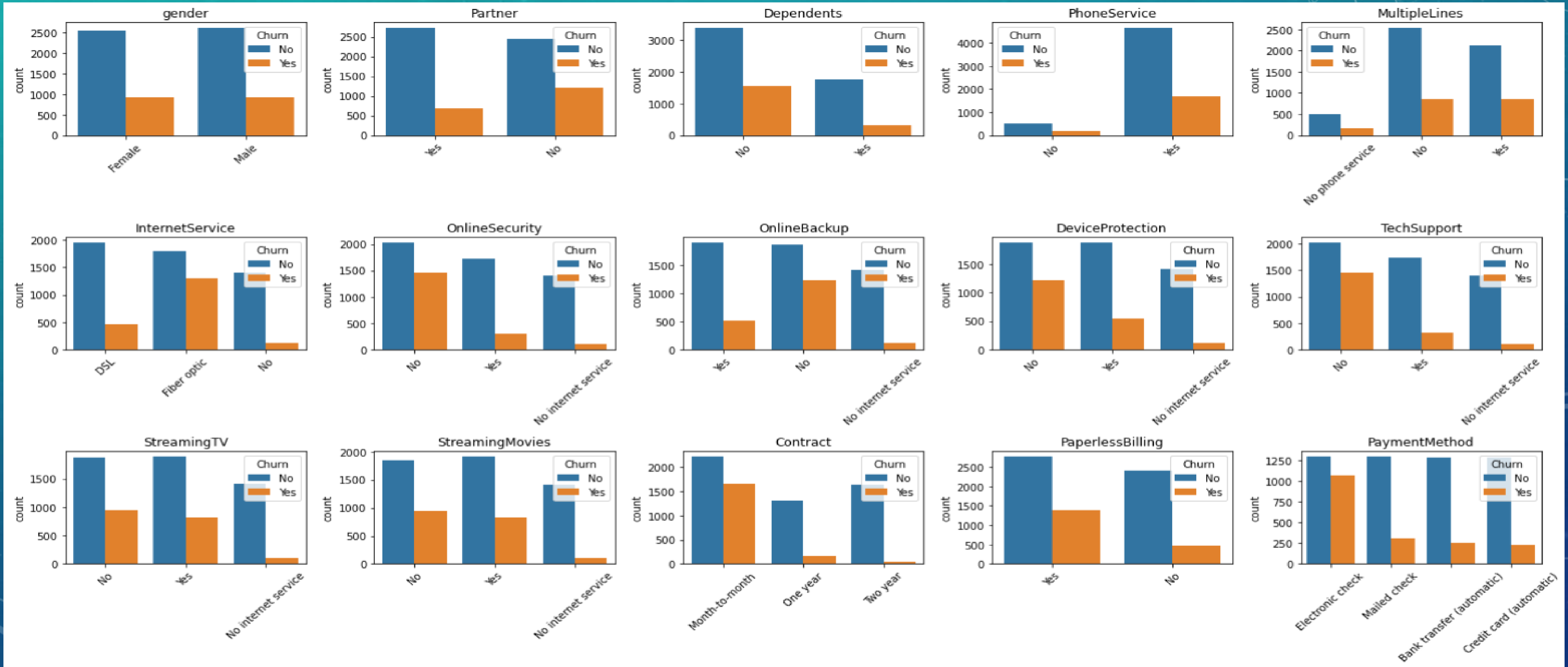
UNDERSTAND THE DATA

➢ Features description:

- Contract – the customer’s contract term (“Month-to-month”, “One year”, “Two year”)
- PaperlessBilling – whether the customer has paperless billing (“Yes” or “No”)
- PaymentMethod – customer’s payment method (“Electronic check”, “Mailed check”, “Bank transfer (automatic)”, “Credit card (automatic)”)
- MonthlyCharges – customer’s monthly charges
- TotalCharges – customer’s total charges over the period he/she stay

UNDERSTAND THE DATA

- number of customers that stay or churn in each categorical feature.



UNDERSTANDING THE DATA

Seems like there are some interesting insight from the count plot above:

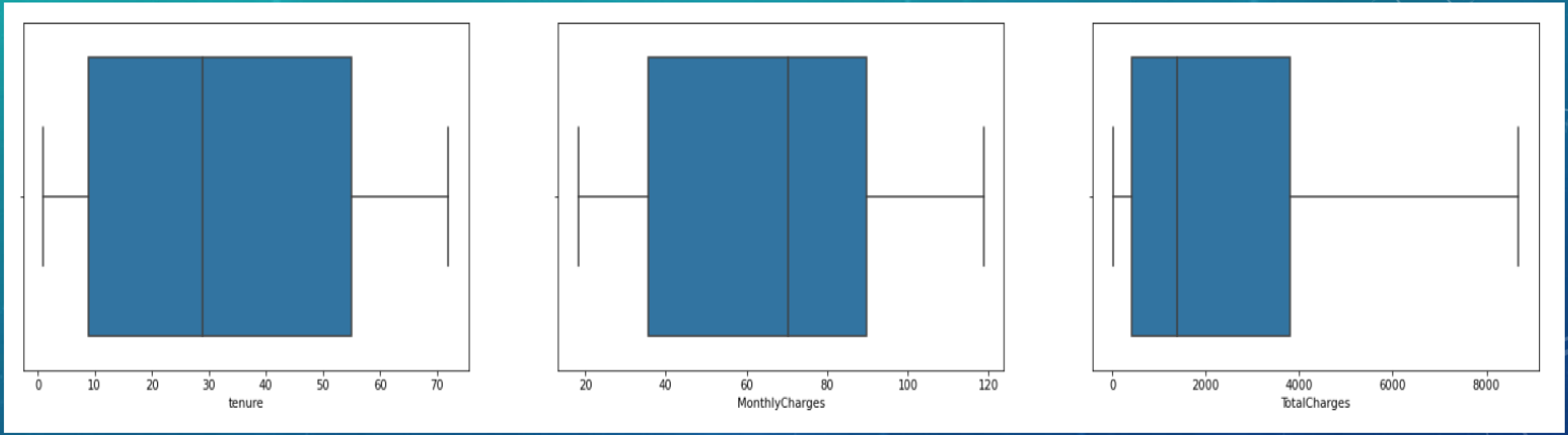
- Gender does not affect whether the customer stay or churn - almost the same.
- Customers with partner and dependents tend to stay with the company - one payment for all?
- Customers with phone service churn more regardless of single or multiple line - prices?
- Customers using fiber optic churn more - again prices?

UNDERSTANDING THE DATA

- Customers without online security, online backup, device protection and tech support churn more - higher premium for these services?
- Having streaming package or not does not affect much to the rate of churn - almost the same.
- Customers without contract churn the most - NEED TO TIE THEM DOWN!!!
- Customers having paperless billing tend to churn more - mostly tech savvy so more easily to compare prices and services online. Younger generation?
- Customers using electronics check payment churn more than those using other payment methods - more troublesome?

UNDERSTAND THE DATA

- Check for any outliers in “tenure”, “MonlyCharges”, “TotalCharges”

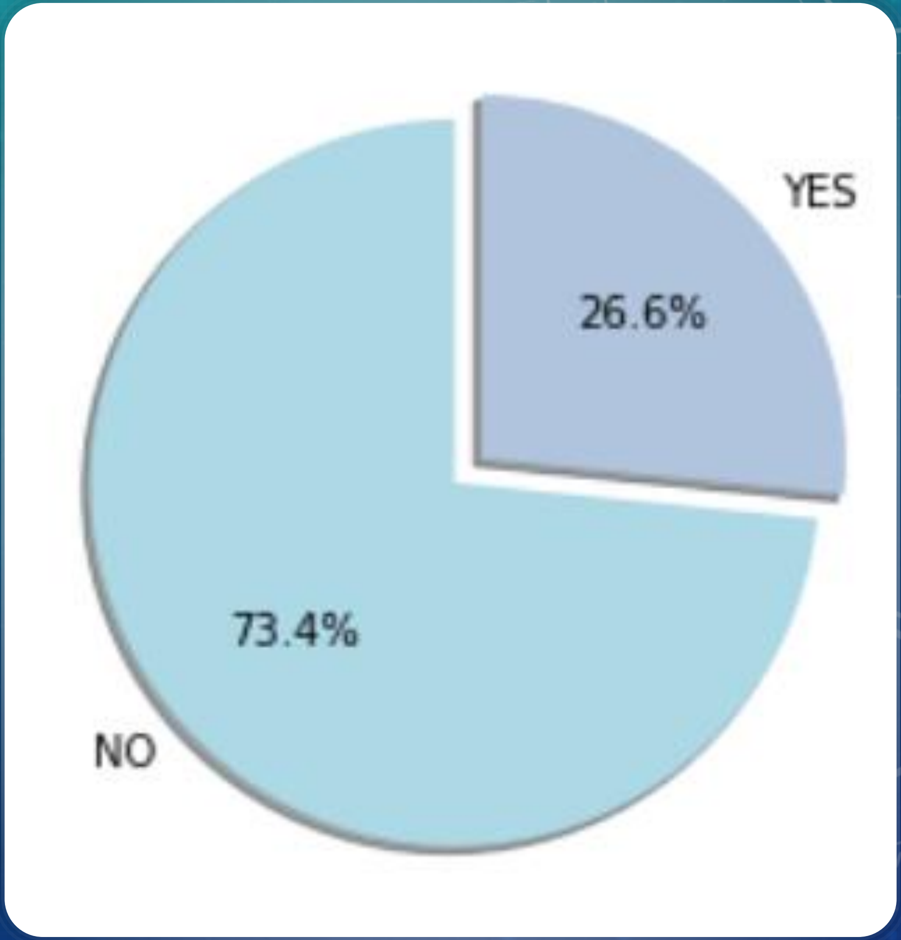


There are no outliers

UNDERSTAND THE DATA

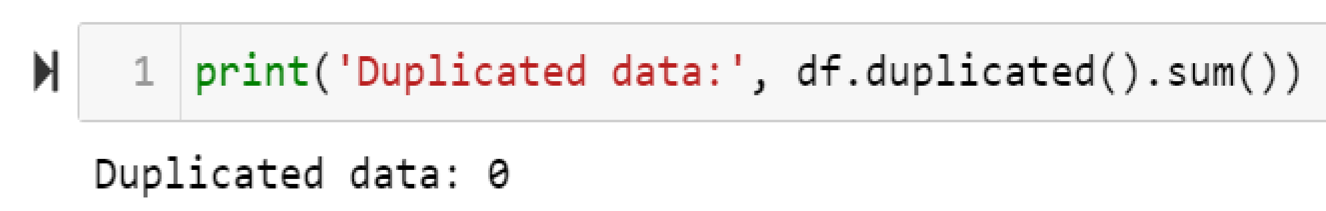
- Distribution of “Churn”

  - we have only about 26.6% of rows that is “YES” (customers who will churn).
  - The dataset is imbalanced and we will have to deal with it later when we are training the models.



DATA CLEANING

1. Check for any duplicate data:

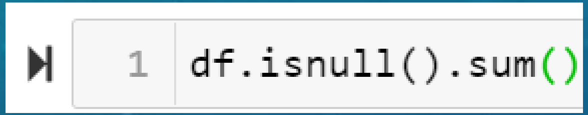


```python
print('Duplicated data:', df.duplicated().sum())
```

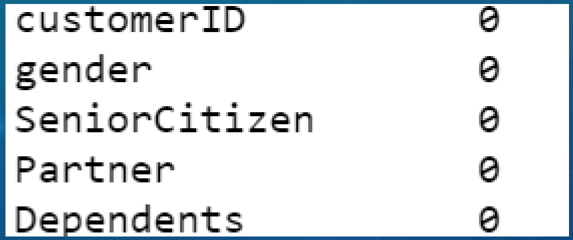
```
Duplicated data: 0
```

2. Drop “customerID” from the dataset

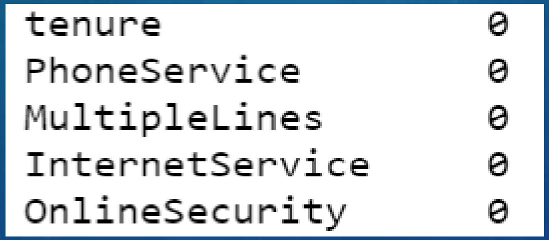
3. Check for any missing values



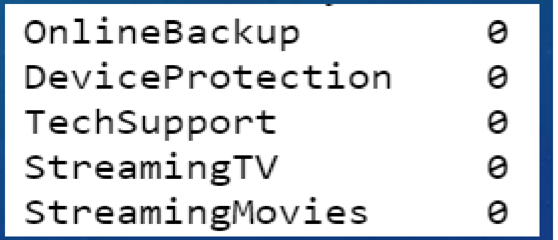
```python
df.isnull().sum()
```



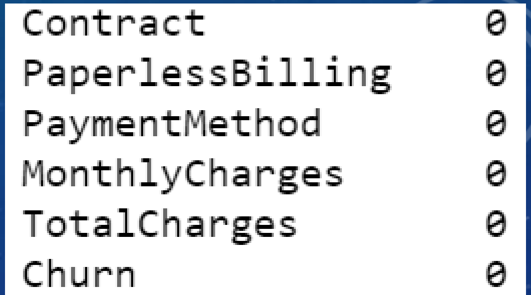
```
customerID          0
gender              0
SeniorCitizen       0
Partner             0
Dependents          0
```



```
tenure              0
PhoneService        0
MultipleLines       0
InternetService     0
OnlineSecurity      0
```



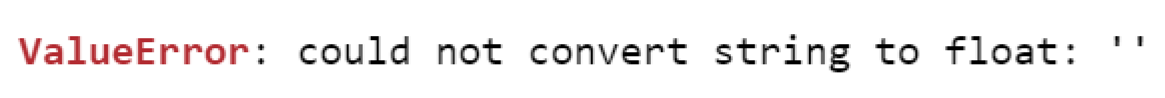
```
OnlineBackup        0
DeviceProtection    0
TechSupport         0
StreamingTV         0
StreamingMovies     0
```



```
Contract            0
PaperlessBilling    0
PaymentMethod       0
MonthlyCharges      0
TotalCharges        0
Churn               0
```

DATA CLEANING

4. Checking on the features data type, we need to convert “TotalCharges” from object to float

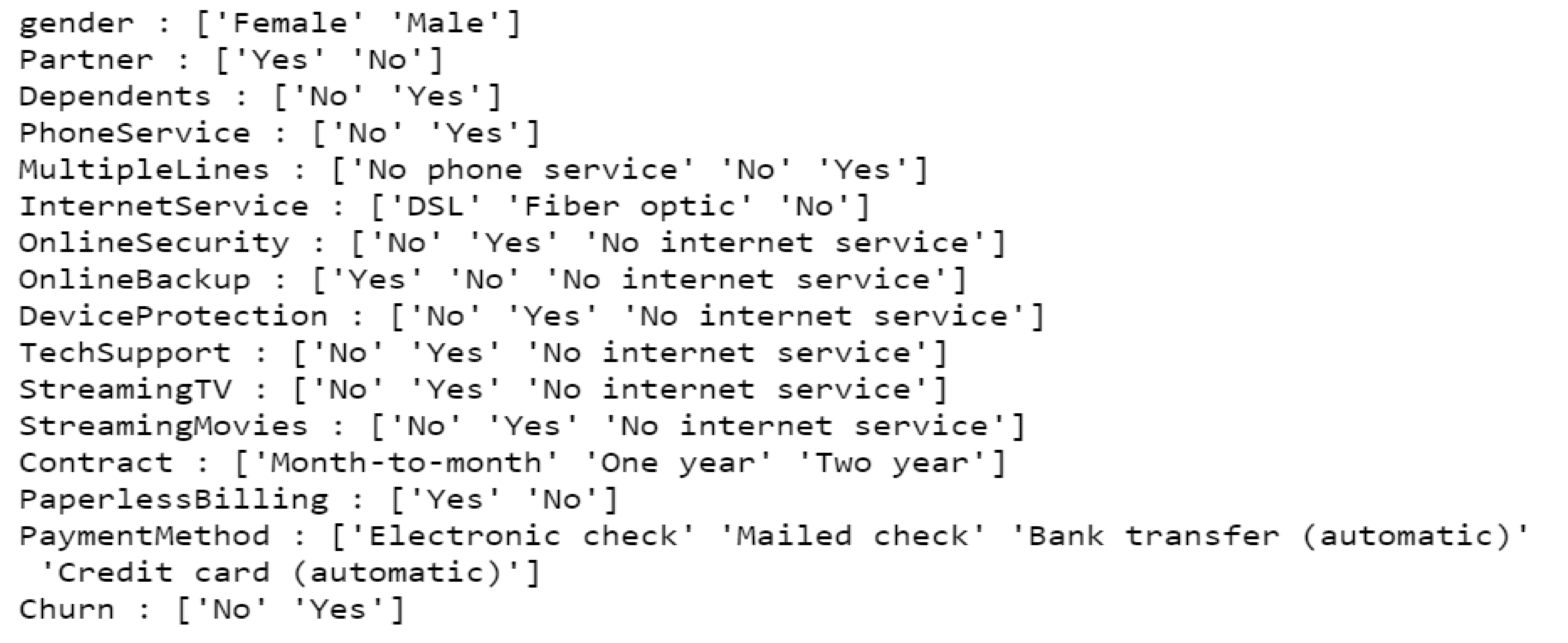


```
ValueError: could not convert string to float: ''
```

WE GOT AN ERROR!!!

From the error it seems like there are unexpected missing values (‘ ‘) in “TotalCharges”. We need to check the dataset again and see if there are other unexpected missing values in other object features

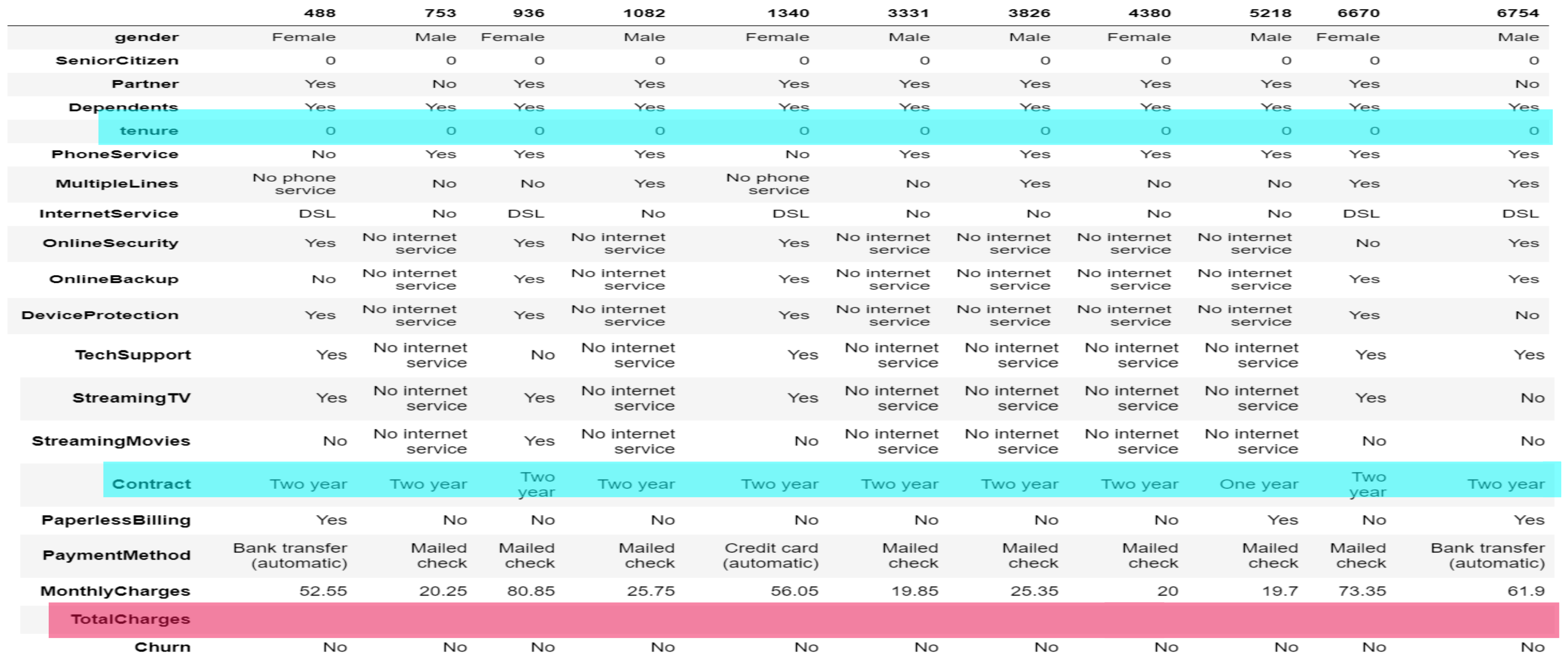
DATA CLEANING



```
gender : ['Female' 'Male']
Partner : ['Yes' 'No']
Dependents : ['No' 'Yes']
PhoneService : ['No' 'Yes']
MultipleLines : ['No phone service' 'No' 'Yes']
InternetService : ['DSL' 'Fiber optic' 'No']
OnlineSecurity : ['No' 'Yes' 'No internet service']
OnlineBackup : ['Yes' 'No' 'No internet service']
DeviceProtection : ['No' 'Yes' 'No internet service']
TechSupport : ['No' 'Yes' 'No internet service']
StreamingTV : ['No' 'Yes' 'No internet service']
StreamingMovies : ['No' 'Yes' 'No internet service']
Contract : ['Month-to-month' 'One year' 'Two year']
PaperlessBilling : ['Yes' 'No']
PaymentMethod : ['Electronic check' 'Mailed check' 'Bank transfer (automatic)'
 'Credit card (automatic)']
Churn : ['No' 'Yes']
```

Only “TotalCharges” has missing values

DATA CLEANING



	488	753	936	1082	1340	3331	3826	4380	5218	6670	6754
gender	Female	Male	Female	Male	Female	Male	Male	Female	Male	Female	Male
SeniorCitizen	0	0	0	0	0	0	0	0	0	0	0
Partner	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Dependents	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
tenure	0	0	0	0	0	0	0	0	0	0	0
PhoneService	No	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes
MultipleLines	No phone service	No	No	Yes	No phone service	No	Yes	No	No	Yes	Yes
InternetService	DSL	No	DSL	No	DSL	No	No	No	No	DSL	DSL
OnlineSecurity	Yes	No internet service	Yes	No internet service	Yes	No internet service	No internet service	No internet service	No internet service	No	Yes
OnlineBackup	No	No internet service	Yes	No internet service	Yes	No internet service	No internet service	No internet service	No internet service	Yes	Yes
DeviceProtection	Yes	No internet service	Yes	No internet service	Yes	No internet service	No internet service	No internet service	No internet service	Yes	No
TechSupport	Yes	No internet service	No	No internet service	Yes	No internet service	No internet service	No internet service	No internet service	Yes	Yes
StreamingTV	Yes	No internet service	Yes	No internet service	Yes	No internet service	No internet service	No internet service	No internet service	Yes	No
StreamingMovies	No	No internet service	Yes	No internet service	No	No internet service	No internet service	No internet service	No internet service	No	No
Contract	Two year	Two year	Two year	Two year	Two year	Two year	Two year	Two year	One year	Two year	Two year
PaperlessBilling	Yes	No	No	No	No	No	No	No	Yes	No	Yes
PaymentMethod	Bank transfer (automatic)	Mailed check	Mailed check	Mailed check	Credit card (automatic)	Mailed check	Mailed check	Mailed check	Mailed check	Mailed check	Bank transfer (automatic)
MonthlyCharges	52.55	20.25	80.85	25.75	56.05	19.85	25.35	20	19.7	73.35	61.9
TotalCharges											
Churn	No	No	No	No	No	No	No	No	No	No	No

There are only 11 rows with missing values

Remove from dataset since they are all newly join with “tenure” less then 1 month and all under contract and convert “TotalCharges” to float

DATA PRE - PROCESSING

1. Spilt the dataset into X (features) and y (labels)
2. Convert features:

- "gender", "Partner", "Dependents", "PhoneService" and "PaperlessBilling“ using Dummy Encoding as there are only 2 classes in these features
- “Contract” using Ordinal Encoding as we ranking to the classes (“Month-to-month” as “2”, “One year” as “1”, “Two years as “0”)
- using Frequency Encoding for rest of the categorical features
- use Label Encoding to convert the label

DATA PRE - PROCESSING

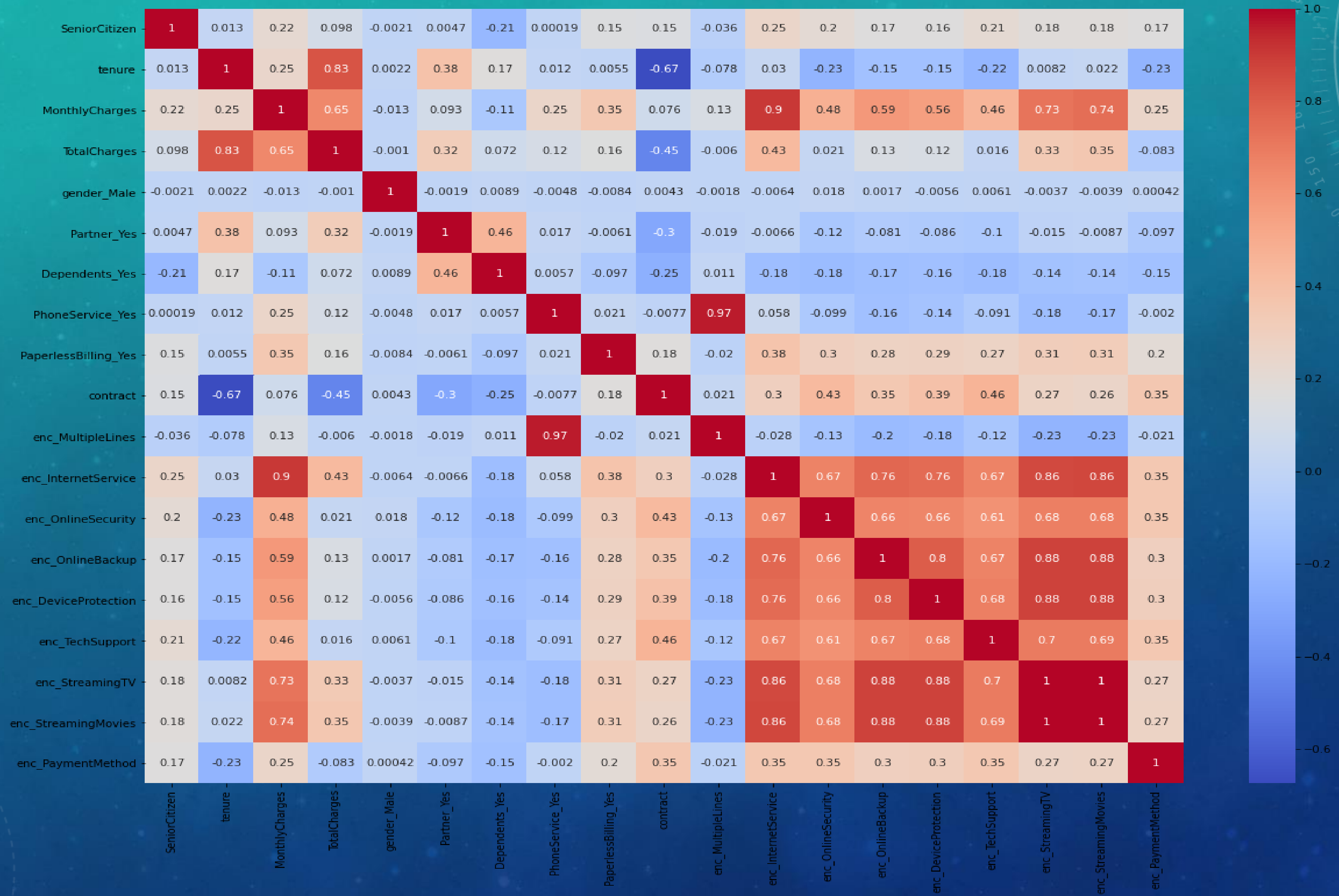
3. Split the dataset into train set and test set

   - Since the dataset is imbalance, we will split the data using stratification so that the percentage of customers who churn will be the same in both train and test dataset.

4. Perform feature scaling on "tenure“, "MonthlyCharges", "TotalCharges" and "Contract"

5. Perform a multicollinearity check to see any correlations between features.

DATA PRE - PROCESSING



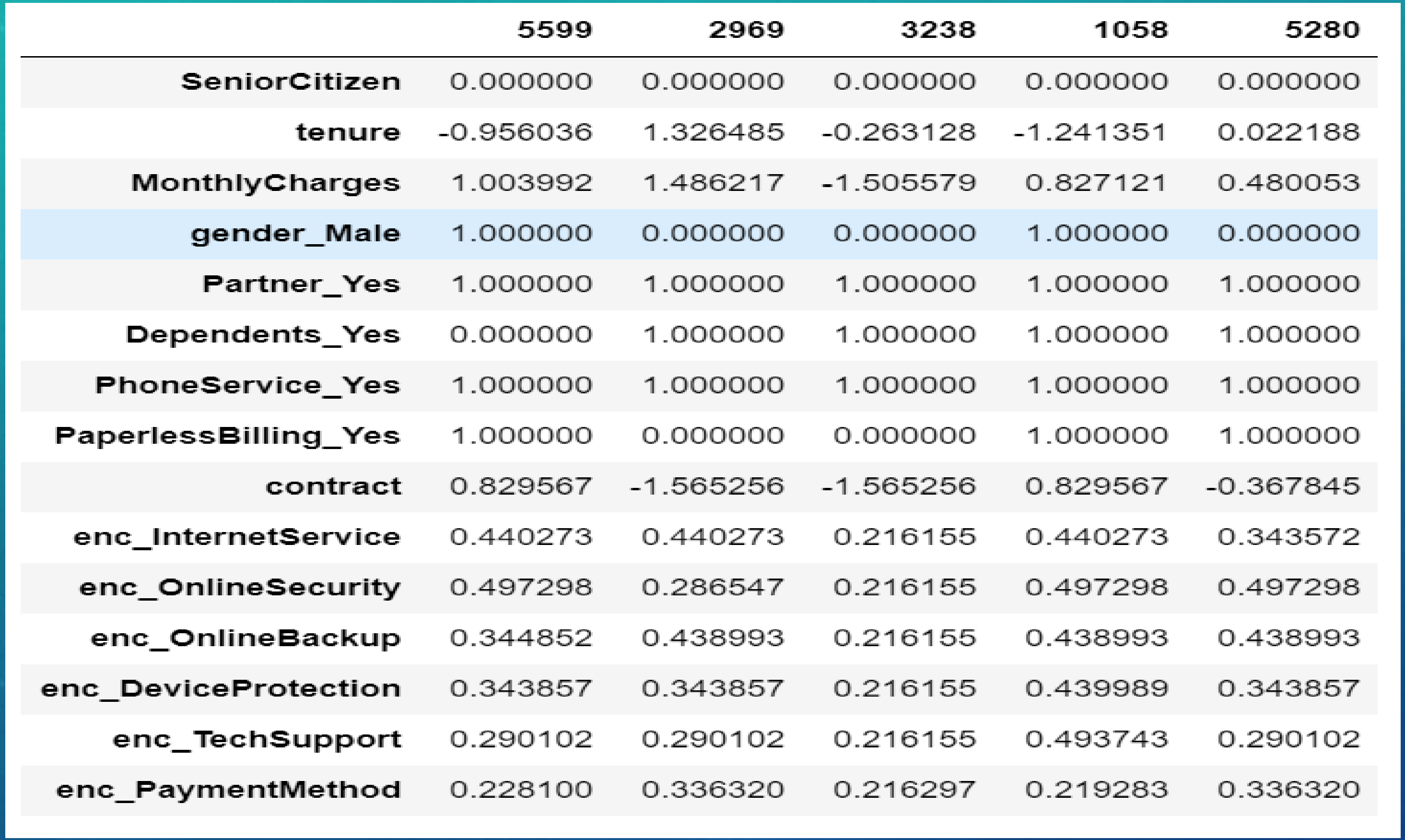
	SeniorCitizen	tenure	MonthlyCharges	TotalCharges	gender_Male	Partner_Yes	Dependents_Yes	PhoneService_Yes	PaperlessBilling_Yes	contract	enc_MultipleLines	enc_InternetService	enc_OnlineSecurity	enc_OnlineBackup	enc_DeviceProtection	enc_TechSupport	enc_StreamingTV	enc_StreamingMovies	enc_PaymentMethod
SeniorCitizen	1	0.013	0.22	0.098	-0.0021	0.0047	-0.21	0.00019	0.15	0.15	-0.036	0.25	0.2	0.17	0.16	0.21	0.18	0.18	0.17
tenure	0.013	1	0.25	0.83	0.0022	0.38	0.17	0.012	0.0055	-0.67	-0.078	0.03	-0.23	-0.15	-0.15	-0.22	0.0082	0.022	-0.23
MonthlyCharges	0.22	0.25	1	0.65	-0.013	0.093	-0.11	0.25	0.35	0.076	0.13	0.9	0.48	0.59	0.56	0.46	0.73	0.74	0.25
TotalCharges	0.098	0.83	0.65	1	-0.001	0.32	0.072	0.12	0.16	-0.45	-0.006	0.43	0.021	0.13	0.12	0.016	0.33	0.35	-0.083
gender_Male	-0.0021	0.0022	-0.013	-0.001	1	-0.0019	0.0089	-0.0048	-0.0084	0.0043	-0.0018	-0.0064	0.018	0.0017	-0.0056	0.0061	-0.0037	-0.0039	0.00042
Partner_Yes	0.0047	0.38	0.093	0.32	-0.0019	1	0.46	0.017	-0.0061	-0.3	-0.019	-0.0066	-0.12	-0.081	-0.086	-0.1	-0.015	-0.0087	-0.097
Dependents_Yes	-0.21	0.17	-0.11	0.072	0.0089	0.46	1	0.0057	-0.097	-0.25	0.011	-0.18	-0.18	-0.17	-0.16	-0.18	-0.14	-0.14	-0.15
PhoneService_Yes	0.00019	0.012	0.25	0.12	-0.0048	0.017	0.0057	1	0.021	-0.0077	0.97	0.058	-0.099	-0.16	-0.14	-0.091	-0.18	-0.17	-0.002
PaperlessBilling_Yes	0.15	0.0055	0.35	0.16	-0.0084	-0.0061	-0.097	0.021	1	0.18	-0.02	0.38	0.3	0.28	0.29	0.27	0.31	0.31	0.2
contract	0.15	-0.67	0.076	-0.45	0.0043	-0.3	-0.25	-0.0077	0.18	1	0.021	0.3	0.43	0.35	0.39	0.46	0.27	0.26	0.35
enc_MultipleLines	-0.036	-0.078	0.13	-0.006	-0.0018	-0.019	0.011	0.97	-0.02	0.021	1	-0.028	-0.13	-0.2	-0.18	-0.12	-0.23	-0.23	-0.021
enc_InternetService	0.25	0.03	0.9	0.43	-0.0064	-0.0066	-0.18	0.058	0.38	0.3	-0.028	1	0.67	0.76	0.76	0.67	0.86	0.86	0.35
enc_OnlineSecurity	0.2	-0.23	0.48	0.021	0.018	-0.12	-0.18	-0.099	0.3	0.43	-0.13	0.67	1	0.66	0.66	0.61	0.68	0.68	0.35
enc_OnlineBackup	0.17	-0.15	0.59	0.13	0.0017	-0.081	-0.17	-0.16	0.28	0.35	-0.2	0.76	0.66	1	0.8	0.67	0.88	0.88	0.3
enc_DeviceProtection	0.16	-0.15	0.56	0.12	-0.0056	-0.086	-0.16	-0.14	0.29	0.39	-0.18	0.76	0.66	0.8	1	0.68	0.88	0.88	0.3
enc_TechSupport	0.21	-0.22	0.46	0.016	0.0061	-0.1	-0.18	-0.091	0.27	0.46	-0.12	0.67	0.61	0.67	0.68	1	0.7	0.69	0.35
enc_StreamingTV	0.18	0.0082	0.73	0.33	-0.0037	-0.015	-0.14	-0.18	0.31	0.27	-0.23	0.86	0.68	0.88	0.88	0.7	1	1	0.27
enc_StreamingMovies	0.18	0.022	0.74	0.35	-0.0039	-0.0087	-0.14	-0.17	0.31	0.26	-0.23	0.86	0.68	0.88	0.88	0.69	1	1	0.27
enc_PaymentMethod	0.17	-0.23	0.25	-0.083	0.00042	-0.097	-0.15	-0.002	0.2	0.35	-0.021	0.35	0.35	0.3	0.3	0.35	0.27	0.27	1

DATA PRE - PROCESSING

- "tenure", "TotalCharges"
- "MonthlyCharges", "InternetService"
- "PhoneService", "MultipleLines"
- "InternetService", "StreamingTV", "StreamingMovies"
- "OnlineBackup", "StreamingTV", "StreamingMovies"
- "DeviceProtection", "StreamingTV", "StreamingMovies"

6. Drop "MultipleLines“, "StreamingTV", "StreamingMovies“, “TotalCharges”

DATA PRE-PROCESSING



	5599	2969	3238	1058	5280
SeniorCitizen	0.000000	0.000000	0.000000	0.000000	0.000000
tenure	-0.956036	1.326485	-0.263128	-1.241351	0.022188
MonthlyCharges	1.003992	1.486217	-1.505579	0.827121	0.480053
gender_Male	1.000000	0.000000	0.000000	1.000000	0.000000
Partner_Yes	1.000000	1.000000	1.000000	1.000000	1.000000
Dependents_Yes	0.000000	1.000000	1.000000	1.000000	1.000000
PhoneService_Yes	1.000000	1.000000	1.000000	1.000000	1.000000
PaperlessBilling_Yes	1.000000	0.000000	0.000000	1.000000	1.000000
contract	0.829567	-1.565256	-1.565256	0.829567	-0.367845
enc_InternetService	0.440273	0.440273	0.216155	0.440273	0.343572
enc_OnlineSecurity	0.497298	0.286547	0.216155	0.497298	0.497298
enc_OnlineBackup	0.344852	0.438993	0.216155	0.438993	0.438993
enc_DeviceProtection	0.343857	0.343857	0.216155	0.439989	0.343857
enc_TechSupport	0.290102	0.290102	0.216155	0.493743	0.290102
enc_PaymentMethod	0.228100	0.336320	0.216297	0.219283	0.336320

7. The dataset is ready for training!!!

TRAINING THE MODELS

1. Train a few model and get the baseline performance of each model

   - will be using parameter “class_weight”, “sample_weight” as the dataset is imbalanced

2. Models we will be training:

   - Logistic Regression

   - Decision Tree

   - Random Forest

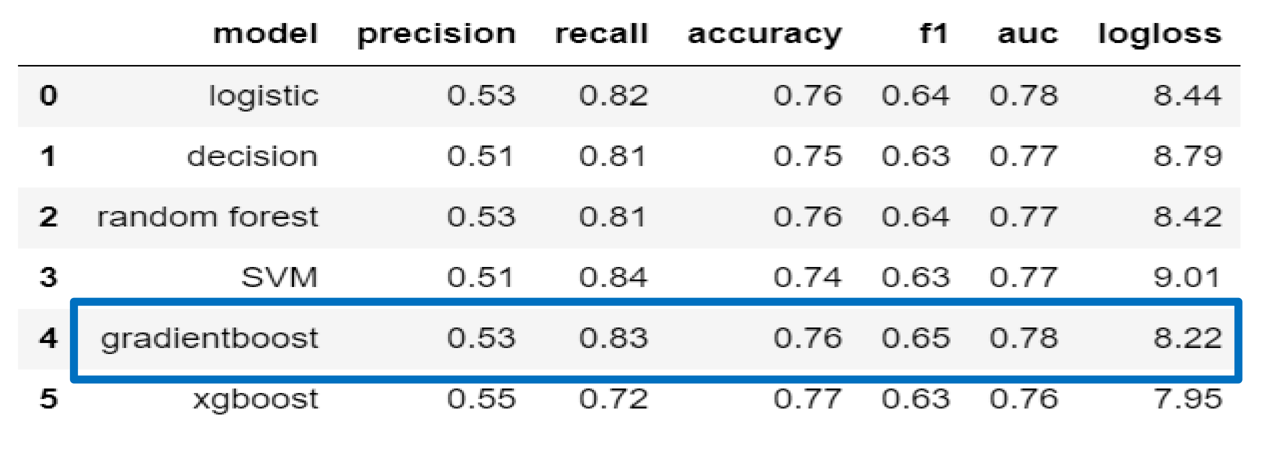
   - Support Vector Machine

   - Gradient Boost

   - XGBoost

MODEL EVALUATION

- We have computed the metrics value for all the model in the table shown:



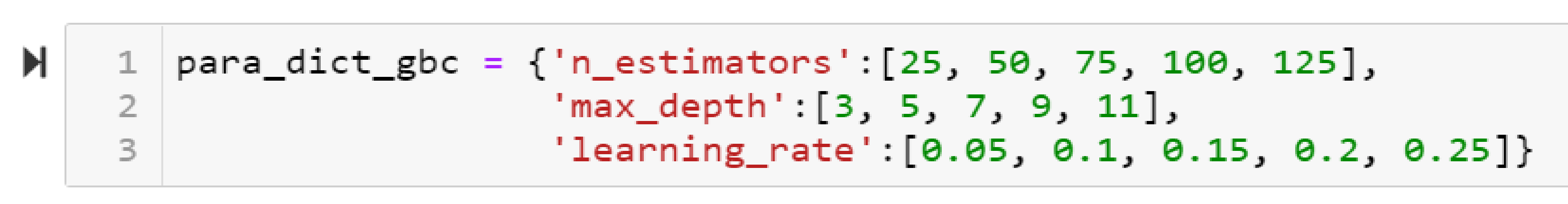
	model	precision	recall	accuracy	f1	auc	logloss
0	logistic	0.53	0.82	0.76	0.64	0.78	8.44
1	decision	0.51	0.81	0.75	0.63	0.77	8.79
2	random forest	0.53	0.81	0.76	0.64	0.77	8.42
3	SVM	0.51	0.84	0.74	0.63	0.77	9.01
4	gradientboost	0.53	0.83	0.76	0.65	0.78	8.22
5	xgboost	0.55	0.72	0.77	0.63	0.76	7.95

- From the table we can see that the scores for all models looks identical.
- We decide to choose our model base on recall (the predicted number of customers that churned against the actual number of customers that churned) and log-loss (measure how close the probability of the predicted class is to the ground truth). Here, we will choose Gradient Boost as our final model.

*SVM has best recall but worst log-loss. XGBOOST has the best log-loss but worst recall. That is why Gradient Boost is our pick.

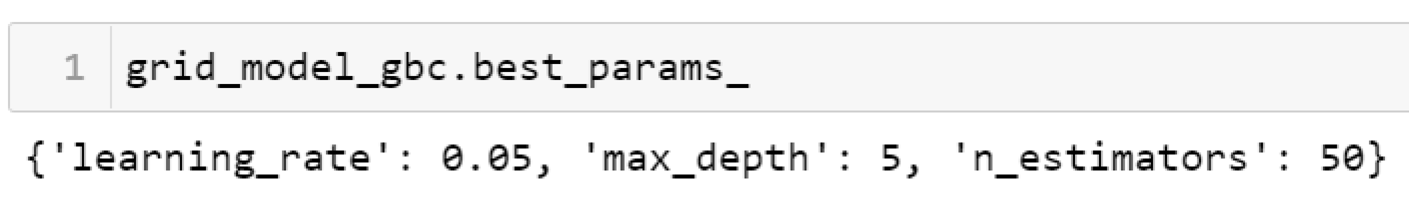
TRAINING THE SELECTED MODEL (GRADIENTBOOST)

1. Find the best parameters value for the model
   - Set a few values for the parameters



```
para_dict_gbc = {'n_estimators':[25, 50, 75, 100, 125],
                 'max_depth':[3, 5, 7, 9, 11],
                 'learning_rate':[0.05, 0.1, 0.15, 0.2, 0.25]}
```

   - Use GridSearchCV to get the best parameters value



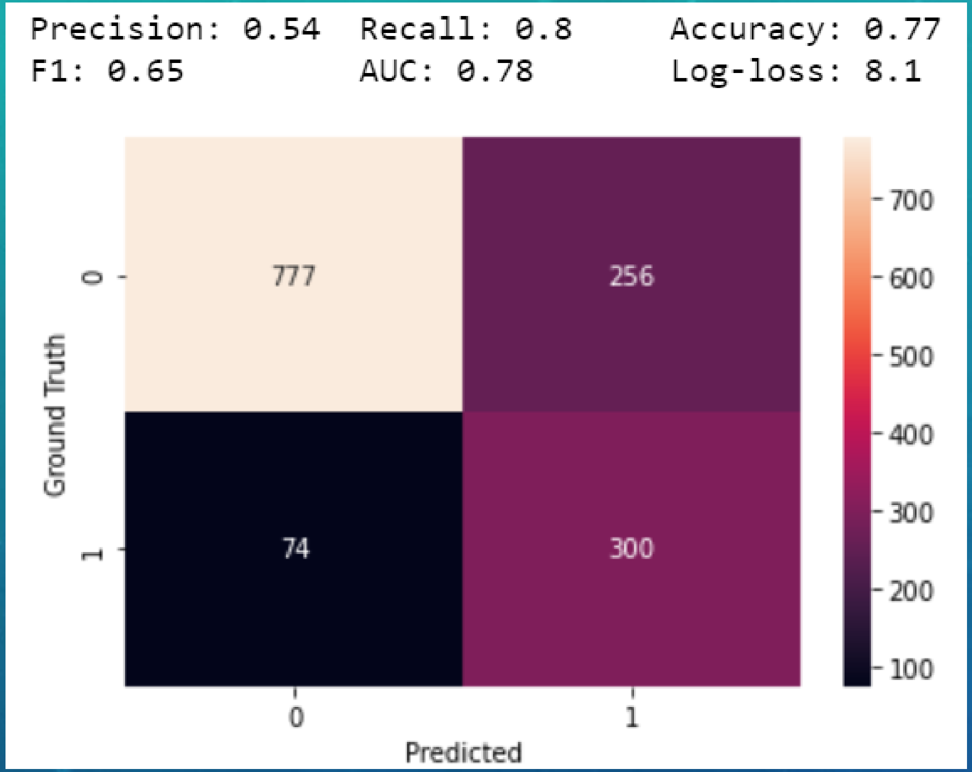
```
grid_model_gbc.best_params_
```

```
{'learning_rate': 0.05, 'max_depth': 5, 'n_estimators': 50}
```

2. Fit and train the model with the parameters best value

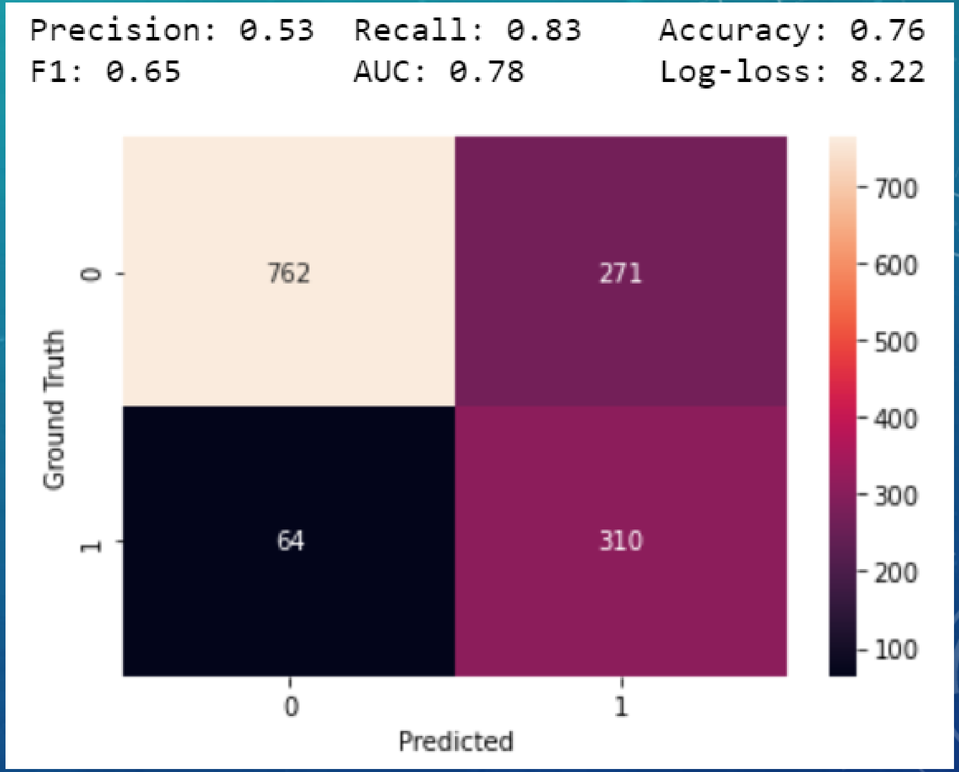
EVALUATE THE MODEL

Scores after tuning



```
Precision: 0.54    Recall: 0.8      Accuracy: 0.77
F1: 0.65           AUC: 0.78        Log-loss: 8.1
```

Scores before tuning

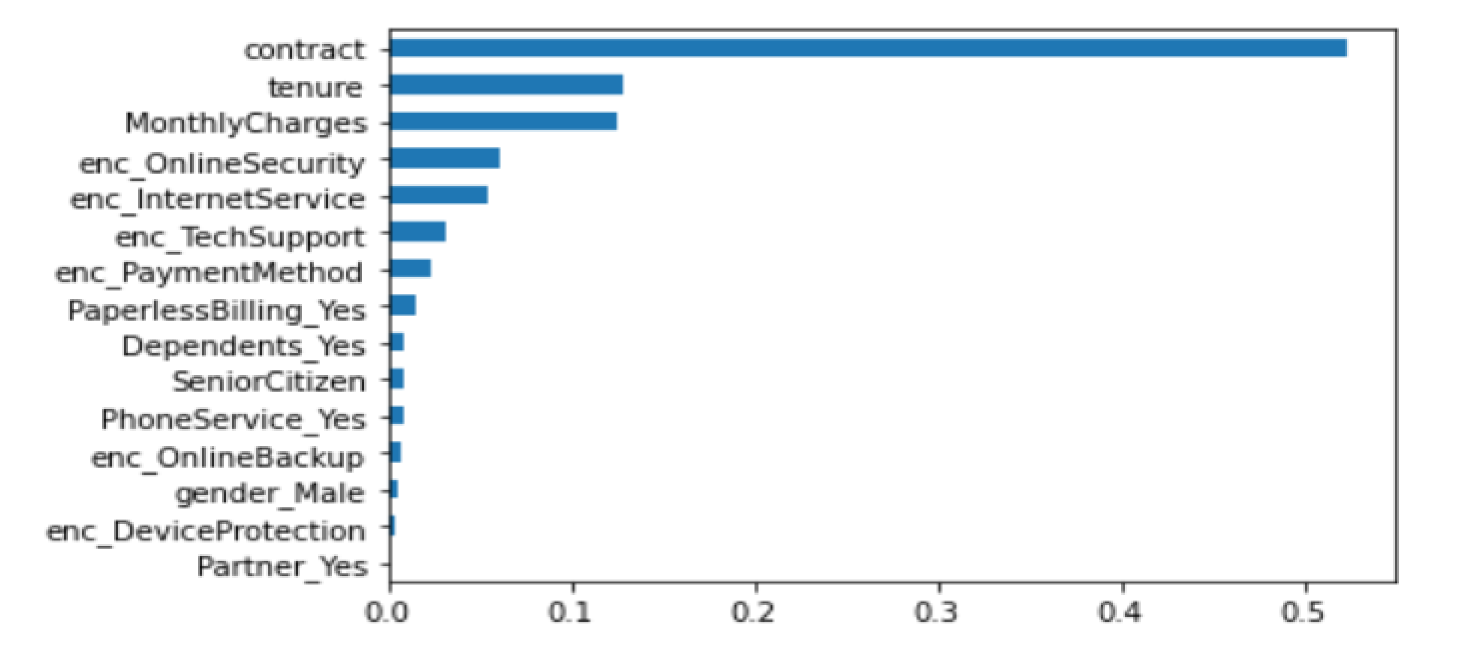


```
Precision: 0.53    Recall: 0.83     Accuracy: 0.76
F1: 0.65           AUC: 0.78        Log-loss: 8.22
```

- The log-loss and accuracy have improved but the recall score has dropped.

EVALUATE THE MODEL

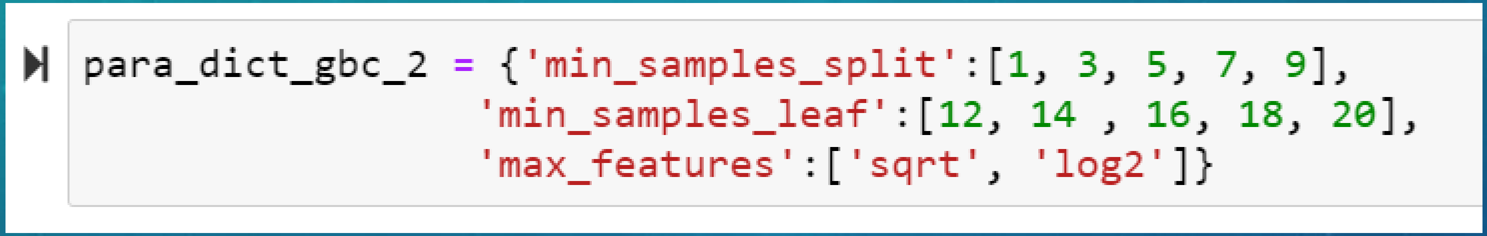
- How each features contributes to the model prediction



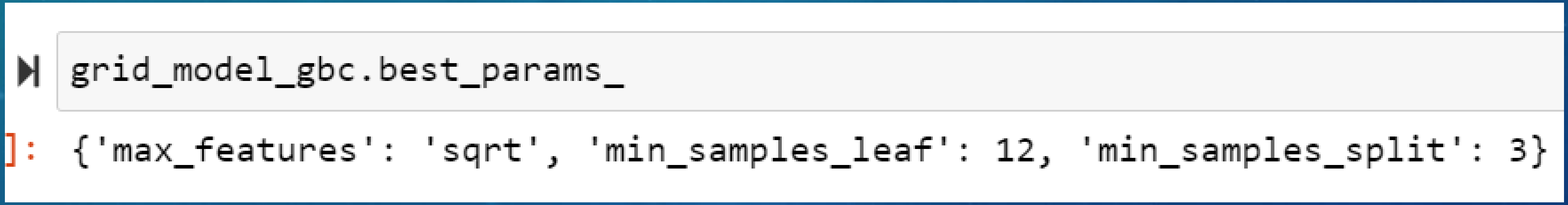
- As we can see, "contract" contributes the most compare to other features (more than 50%).

EVALUATE THE MODEL

- Although the accuracy and log-loss improved, but since we want to catch as much as possible of those customers that churned, we will want the recall score to be as high as possible (but it has dropped).
- Tune more hyperparameters and see how the tuned model perform.



```
para_dict_gbc_2 = {'min_samples_split':[1, 3, 5, 7, 9],
                   'min_samples_leaf':[12, 14 , 16, 18, 20],
                   'max_features':['sqrt', 'log2']}
```



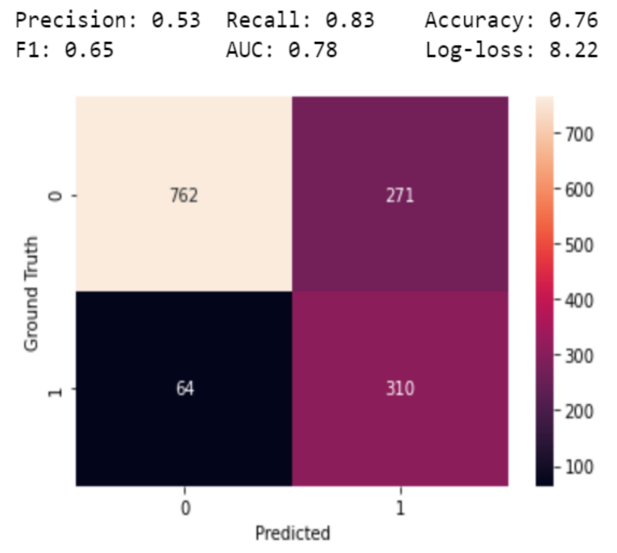
```
grid_model_gbc.best_params_
```

```
]: {'max_features': 'sqrt', 'min_samples_leaf': 12, 'min_samples_split': 3}
```

- Fit and train the model using all hyperparameters we tuned with the best value.

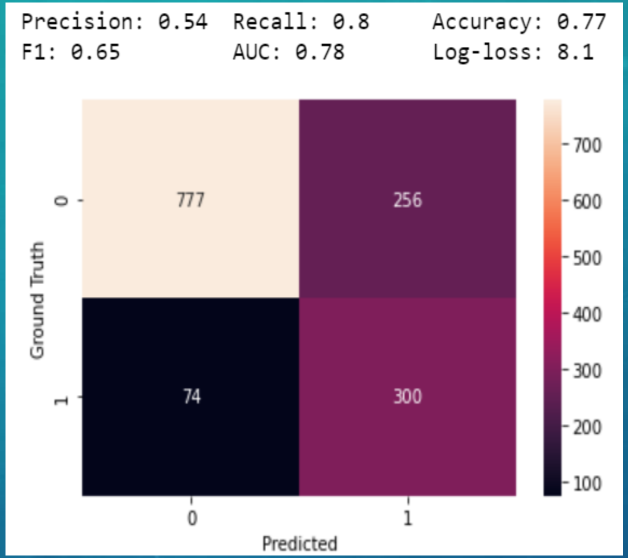
EVALUATE THE MODEL

Base Model



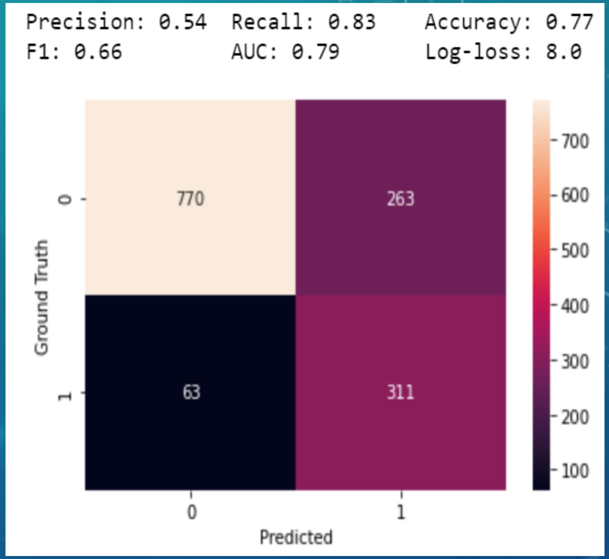
```
Precision: 0.53   Recall: 0.83    Accuracy: 0.76
F1: 0.65          AUC: 0.78       Log-loss: 8.22
```

1st Tuned Model



```
Precision: 0.54   Recall: 0.8     Accuracy: 0.77
F1: 0.65          AUC: 0.78       Log-loss: 8.1
```

2nd Tuned Model

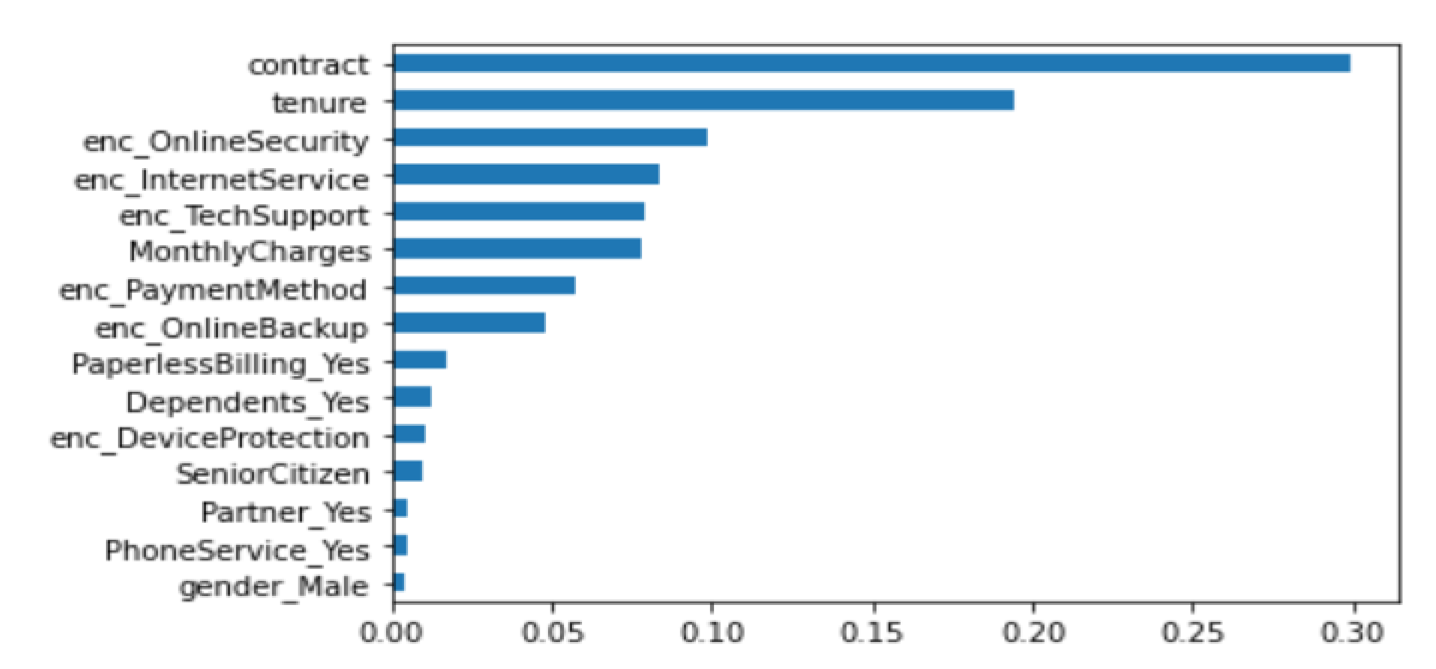


```
Precision: 0.54   Recall: 0.83    Accuracy: 0.77
F1: 0.66          AUC: 0.79       Log-loss: 8.0
```

- Compare to the 1st tuned model, recall has improved from 80% to 83% (same as the base model). Log-loss has also improved by 0.1 (from 8.1 to 8.0). Although the accuracy has dropped from 77% to 76% but it is still same as the base model. The AUC score (which is also a good indicator that show how the model is performing) has also improved from 78% to 79%. Overall we are satisfied with the 2nd tuned model performance so this will be our final model.

EVALUATE THE MODEL

- Take a look at the features importance.



- From the 2nd tuned model, we can see that although "contract" is still the most importance feature but the weightage has reduced. The importance ranking of the other features have also changed.

CONCLUSION

- The count plot give us an indication on which area the company should work on to keep the customers.
- Together with features importance, the company priority now is to tie down customers who have no contract with the company.
- Depend on the management feedback, we might change our approach on the task in a few ways:
  1. Focusing on those customers that are under contract to predict if they will churned when their contract expired – remove customers without contract from the dataset since the company will be working on them.
  2. Change the threshold of the model to reduce the number of customers that actually “churn” but wrongly predicted as “stay” (improve the recall score).

ADDITIONAL INFORMATION

- https://github.com/andychew8015/ML-Final-Project-.git

Thank You!