Supervision: Dr. Aya Saad

Service Cancellation Predictor with GUI

Team members:

1. Ahmed mohamed abdelhady 20201700077
2. khaled mahmoud shaaban 20201700236
3. ahmed mohamed abdelhamed 20201701051
4. mazen hamada badr 20201700641
5. tarek mohamed saad 20201700425
6. Ahmed tamer elsayed 20201700030

AI Project – FCIS - General

**Project objective:**

Service cancellation is simply when customers leave doing business with an entity. It involves determining the possibility of customers stopping doing business with an entity. It is a critical prediction for many businesses because the cost of acquiring a new customer is five times more than that of retaining an existing customer, so for many businesses, the ability to predict that a particular customer is at a high risk of canceling service, while there is still time to do something about it. The main aim of this project is to predict if the customer is at a high risk of canceling service.

**Dataset:**

The dataset we used contains 7042 record and 21 columns (7043 × 21)

**Tools & Libraries:**

We used a bunch of libraries and its modules

1. Pandas
2. Sklearn
3. Tkinter

Text

Description automatically generated

The “Pandas” library is used to load the data from the csv file to make it easier to work with it and to help us in describing the data types of the features and their numbers.

Text

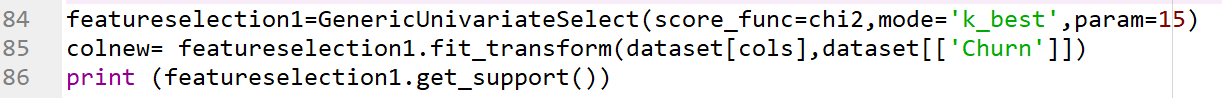
Description automatically generated

After that we used the “Label Encoder” function to transform all the categorical data in all features to numerical data in order to train the models.

Logo

Description automatically generated with medium confidence

After the Encoding step we had to select specific features to get the best accuracy we can get, so we used the module “feature\_selection” then we chose the “GenericUnivariateSelect” function to help us get the best features to keep in order to reach the best accuracy.



Then we usen the “model\_selection” module and the function “train\_test\_split” to split the data (80% for training the model and 20% to test it).



============

**Logistic Regression model**

At that moment the data is ready to be passed to the models to be trained as it is cleaned and the most suitable features are selected, so we began with the “Logistic Regression”

****

We noticed that the “liblinear” has more accuracy than “saga” so we used it.

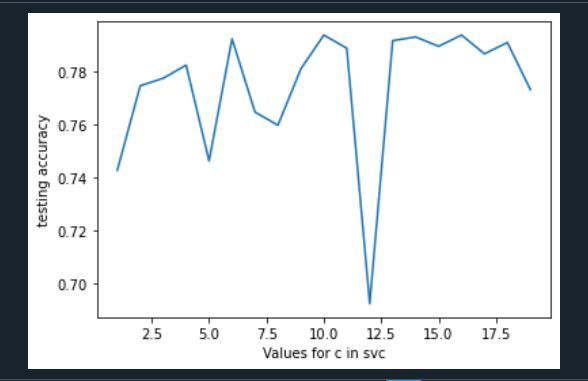
The train accuracy: 0.8030222222222222

The test accuracy: 0.8045486851457001

===============

**SVC model**

We used “metrices” library and the “accuracy\_score” function to know the best “C” we should put in the SVC model



So, we knew that the best value is “10” from the graph and we put in the model.



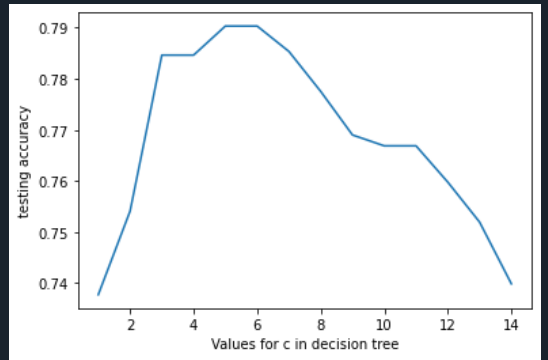
The train accuracy: 0.7884444444444444

The test accuracy: 0.7938877043354655

==============

**Decision Tree model**

We also used the same function to know the “max\_depth” value in the decision tree model.



We knew the value is “7” to get the best accuracy from this model.



The train accuracy: 0.808

The test accuracy: 0.7938877043354655

============

**KNC model**

We used the same function here to know the best value of “k” in the

K-nearest classifier model.

Chart, line chart

Description automatically generated

We knew that the best value is “10” so we used it to get the highest accuracy we can reach.

A picture containing icon

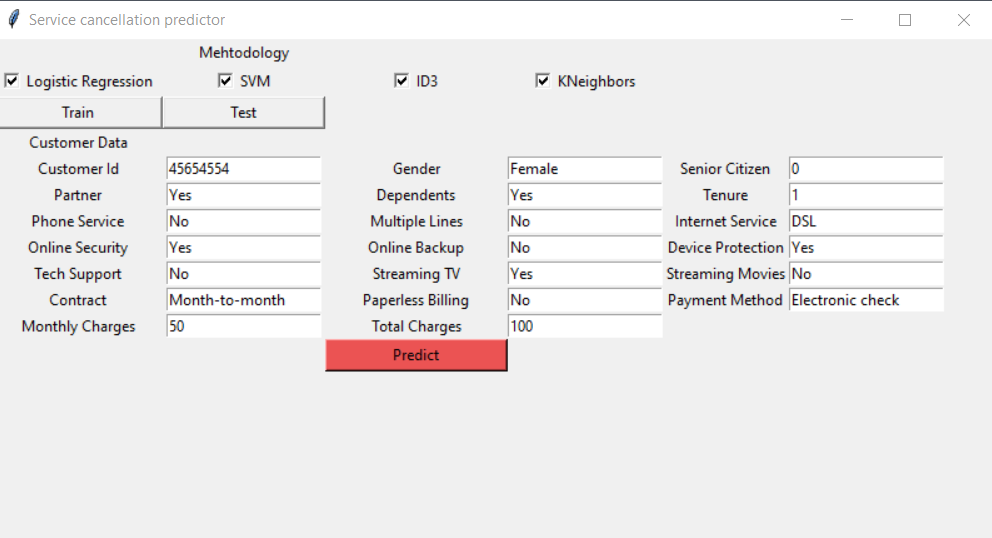
Description automatically generated

The train accuracy: **0.8138666666666666**

The test accuracy: **0.7896233120113717**

**===========**

After that we implemented the GUI interface with the Tkinter library to facilitate the usage of the model to the End-user



So, we can take an overview of the steps to make this prediction model

1. Studying the data and its features
2. Cleaning the data and removing the null values and the duplicates
3. Splitting the data and get it ready for the models
4. Selecting the most suitable features to get the best accuracy
5. Knowing the most suitable attributes for the models
6. Training the models and get the best accuracy
7. Building the GUI interface