

```
In [2]: import pandas as pd
import seaborn as sns
import matplotlib as plt
import numpy as np
```

C:\Users\Rajarshi\anaconda3\lib\site-packages\scipy__init__.py:146: UserWarning: A NumPy version >=1.17.3 and <1.25.0 is required for this version of SciPy (detected version 1.26.0
warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}")

```
In [4]: pip install cufflinks
```

Collecting cufflinks
 Downloading cufflinks-0.17.3.tar.gz (81 kB)
Requirement already satisfied: numpy>=1.9.2 in c:\users\rajarshi\anaconda3\lib\site-packages (from cufflinks) (1.26.0)
Requirement already satisfied: pandas>=0.19.2 in c:\users\rajarshi\anaconda3\lib\site-packages (from cufflinks) (2.1.1)
Requirement already satisfied: plotly>=4.1.1 in c:\users\rajarshi\anaconda3\lib\site-packages (from cufflinks) (5.6.0)
Requirement already satisfied: six>=1.9.0 in c:\users\rajarshi\anaconda3\lib\site-packages (from cufflinks) (1.16.0)
Collecting colorlover>=0.2.1
 Downloading colorlover-0.3.0-py3-none-any.whl (8.9 kB)
Requirement already satisfied: setuptools>=34.4.1 in c:\users\rajarshi\anaconda3\lib\site-packages (from cufflinks) (61.2.0)
Requirement already satisfied: ipython>=5.3.0 in c:\users\rajarshi\anaconda3\lib\site-packages (from cufflinks) (8.2.0)
Requirement already satisfied: ipywidgets>=7.0.0 in c:\users\rajarshi\anaconda3\lib\site-packages (from cufflinks) (7.6.5)
Requirement already satisfied: decorator in c:\users\rajarshi\anaconda3\lib\site-packages (from ipython>=5.3.0->cufflinks) (5.1.1)
Requirement already satisfied: stack-data in c:\users\rajarshi\anaconda3\lib\site-packages (from ipython>=5.3.0->cufflinks) (0.6.2)

```
In [6]: import plotly.express as px
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
import cufflinks as cf # Works as a connector between the pandas Library and plotly
cf.go_offline()
from sklearn.model_selection import GridSearchCV # GridSearchCV implements a "fit" and a "score" method.
```

```
In [8]: df = pd.read_csv('telecom_churn.csv')
df.head(5)

#Class - 0 -> customer retained
#Class - 1 -> customer has left
```

Out[8]:

mon_chrg	total_night_minutes	total_night_calls	total_night_charge	total_intl_minutes	total_intl_calls	total_intl_charge	number_customer_service_calls
16.78	244.7	91	11.01	10.0	3	2.70	1
16.62	254.4	103	11.45	13.7	3	3.70	1
10.30	162.6	104	7.32	12.2	5	3.29	0
5.26	196.9	89	8.86	6.6	7	1.78	2
12.61	186.9	121	8.41	10.1	3	2.73	3

```
In [9]: # Summary of Data Frame
df.info()
```

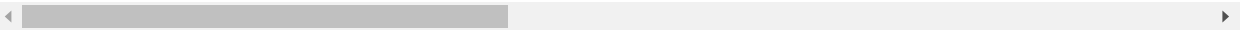
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5000 entries, 0 to 4999
Data columns (total 21 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   state                                5000 non-null   int64
1   account_length                      5000 non-null   int64
2   area_code                           5000 non-null   int64
3   phone_number                        5000 non-null   int64
4   international_plan                  5000 non-null   int64
5   voice_mail_plan                     5000 non-null   int64
6   number_vmail_messages              5000 non-null   int64
7   total_day_minutes                   5000 non-null   float64
8   total_day_calls                     5000 non-null   int64
9   total_day_charge                    5000 non-null   float64
10  total_eve_minutes                   5000 non-null   float64
11  total_eve_calls                     5000 non-null   int64
12  total_eve_charge                    5000 non-null   float64
13  total_night_minutes                 5000 non-null   float64
14  total_night_calls                   5000 non-null   int64
15  total_night_charge                  5000 non-null   float64
16  total_intl_minutes                  5000 non-null   float64
17  total_intl_calls                    5000 non-null   int64
18  total_intl_charge                   5000 non-null   float64
19  number_customer_service_calls       5000 non-null   int64
20  class                               5000 non-null   int64
dtypes: float64(8), int64(13)
memory usage: 820.4 KB
```

```
In [10]: #Description of the data in the DataFrame
df.describe()
```

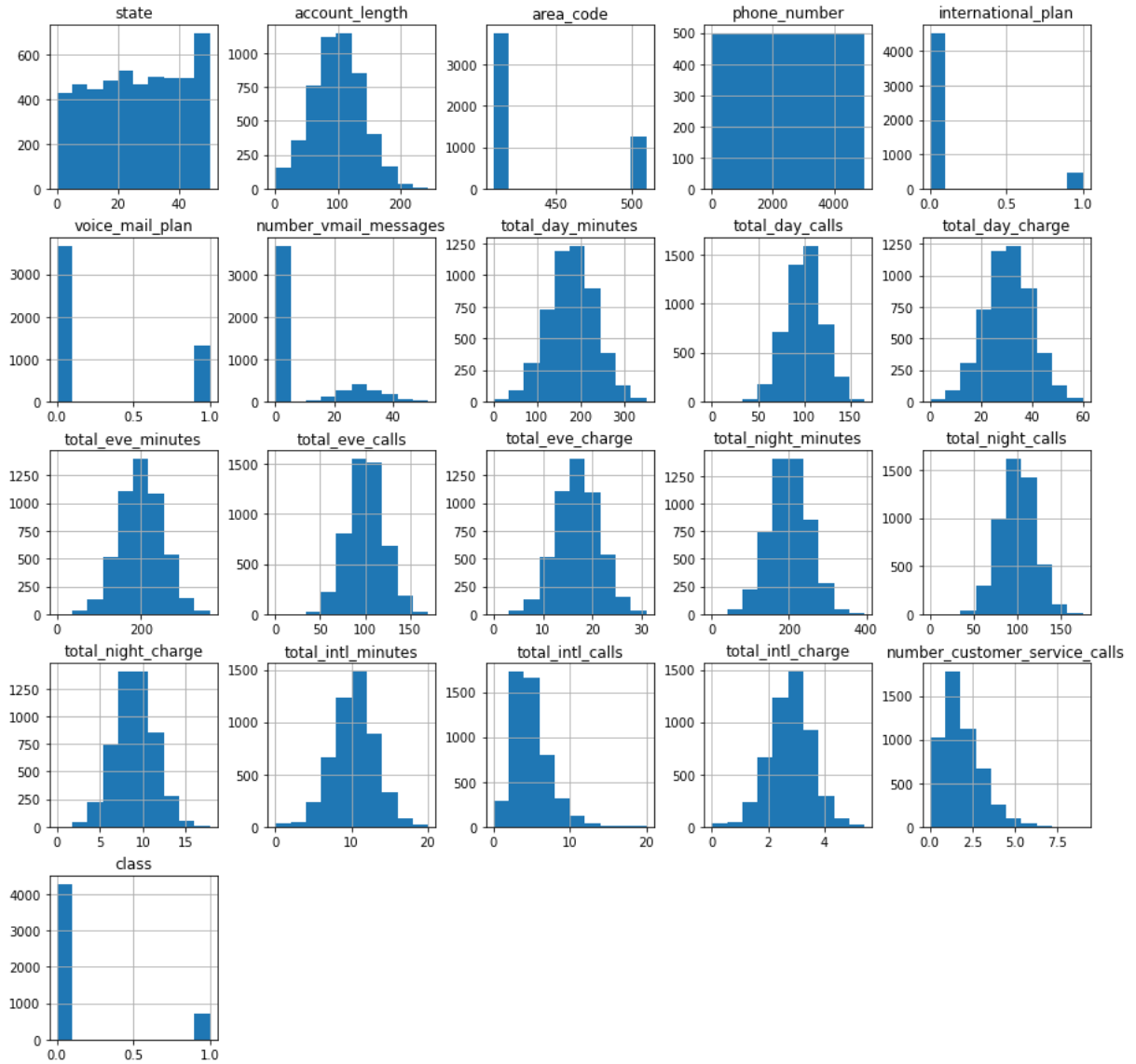
Out[10]:

	state	account_length	area_code	phone_number	international_plan	voice_mail_plan	number_vmail_messages	total_day_minutes	total
count	5000.00000	5000.00000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000
mean	25.99840	100.25860	436.911400	2499.500000	0.094600	0.264600	7.755200	180.288900	180.288900
std	14.80348	39.69456	42.209182	1443.520003	0.292691	0.441164	13.546393	53.894699	53.894699
min	0.00000	1.00000	408.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	13.00000	73.00000	408.000000	1249.750000	0.000000	0.000000	0.000000	143.700000	143.700000
50%	26.00000	100.00000	415.000000	2499.500000	0.000000	0.000000	0.000000	180.100000	180.100000
75%	39.00000	127.00000	415.000000	3749.250000	0.000000	1.000000	17.000000	216.200000	216.200000
max	50.00000	243.00000	510.000000	4999.000000	1.000000	1.000000	52.000000	351.500000	351.500000

8 rows × 21 columns



```
In [43]: import matplotlib.pyplot as plt
## specify figure size (width, height)
df.hist(figsize=(15,15))
plt.show()
```

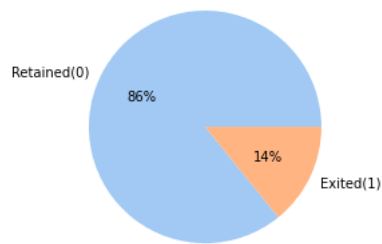


```
In [13]: pip install matplotlib
```

Requirement already satisfied: matplotlib in c:\users\rajarshi\anaconda3\lib\site-packages (3.5.1)
Requirement already satisfied: cyclers>=0.10 in c:\users\rajarshi\anaconda3\lib\site-packages (from matplotlib) (0.11.0)
Requirement already satisfied: pillow>=6.2.0 in c:\users\rajarshi\anaconda3\lib\site-packages (from matplotlib) (9.0.1)
Requirement already satisfied: packaging>=20.0 in c:\users\rajarshi\anaconda3\lib\site-packages (from matplotlib) (21.3)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\rajarshi\anaconda3\lib\site-packages (from matplotlib) (1.3.2)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\rajarshi\anaconda3\lib\site-packages (from matplotlib) (4.25.0)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\rajarshi\anaconda3\lib\site-packages (from matplotlib) (3.0.4)
Requirement already satisfied: numpy>=1.17 in c:\users\rajarshi\anaconda3\lib\site-packages (from matplotlib) (1.26.0)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\rajarshi\anaconda3\lib\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\rajarshi\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
Note: you may need to restart the kernel to use updated packages.

```
In [15]: import matplotlib.pyplot as plt

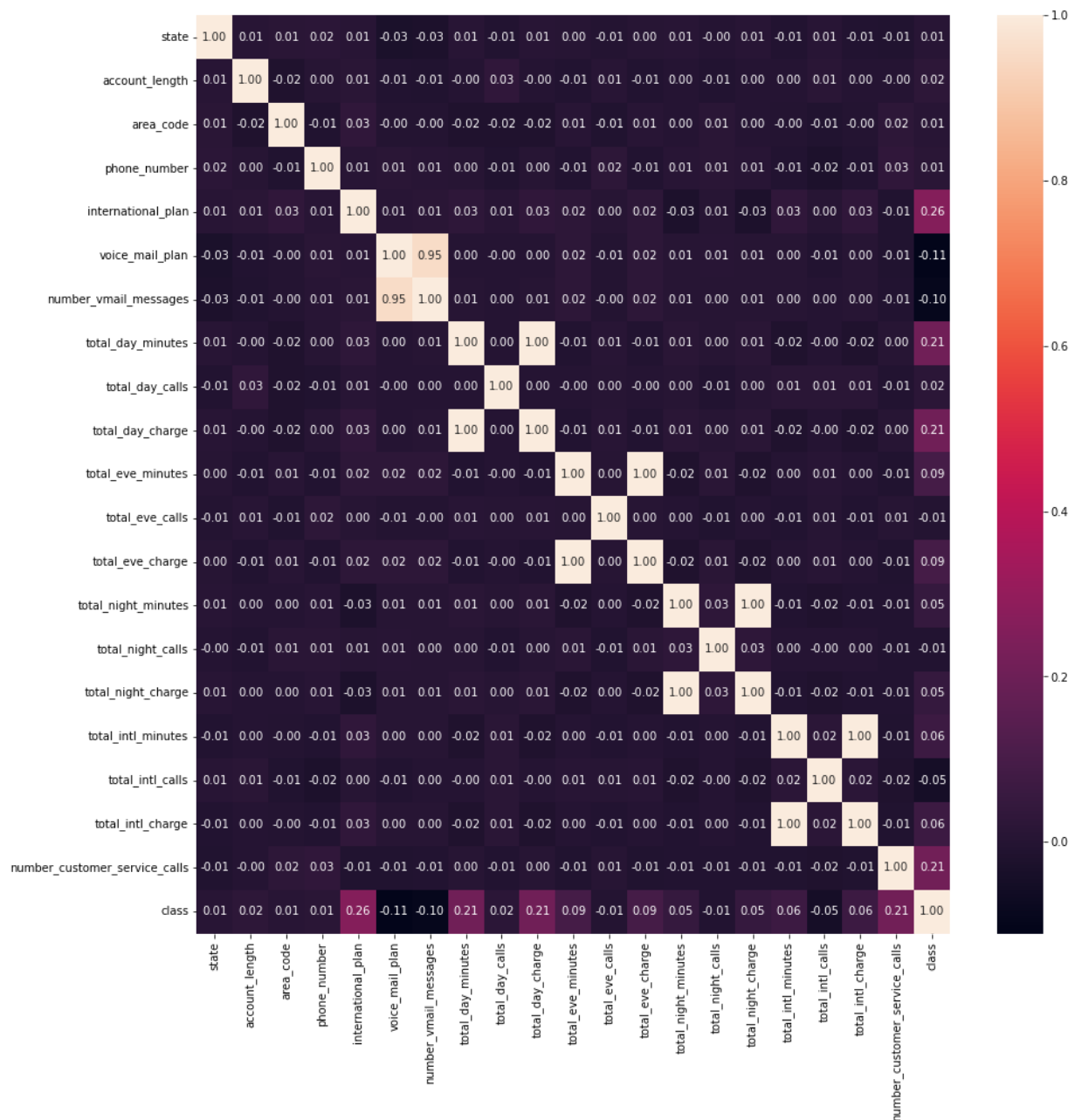
colors = sns.color_palette('pastel')[0:5]
plt.pie(df['class'].value_counts(), labels = ['Retained(0)', 'Exited(1)'], colors = colors, autopct='%0.0f%%')
plt.show()
```



```
In [16]: ## The resulting correlation matrix shows how each column correlates with every other column in the DataFrame.
##The values in the matrix range from -1 to 1, where:

#1 indicates a perfect positive correlation,
#-1 indicates a perfect negative correlation, and
# 0 indicates no correlation.

corr_matrix = df.corr()
plt.figure(figsize = (15,15))
sns.heatmap(corr_matrix, annot = True, fmt = '.0.2f')
plt.show()
```



```
In [17]: # Churn by day charges
ax = sns.kdeplot(df.total_day_charge[df["class"] == 0],
                color = "Red", shade = True) ## Line creates a KDE plot for the "total_day_charge" column where the "class"
ax = sns.kdeplot(df.total_day_charge[df["class"] == 1],
                color = "Blue", shade = True) ## Line adds another KDE plot for the "total_day_charge" column where the "c

ax.legend(["Retain", "Exited"], loc = "upper right") ## adds a Legend to the plot indicating which color corresponds to w
ax.set_ylabel("Density") ## This sets the label for the y-axis as "Density".
ax.set_xlabel("Day Charges") ## This sets the label for the x-axis as "Day Charges".
ax.set_title("Distribution of day charges by churn") ## This sets the title of the plot as "Distribution of day charges b
```

C:\Users\Rajarshi\AppData\Local\Temp\ipykernel_13116\3303476346.py:2: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

C:\Users\Rajarshi\anaconda3\lib\site-packages\seaborn_oldcore.py:1498: FutureWarning:

is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

C:\Users\Rajarshi\anaconda3\lib\site-packages\seaborn_oldcore.py:1119: FutureWarning:

use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

C:\Users\Rajarshi\AppData\Local\Temp\ipykernel_13116\3303476346.py:4: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

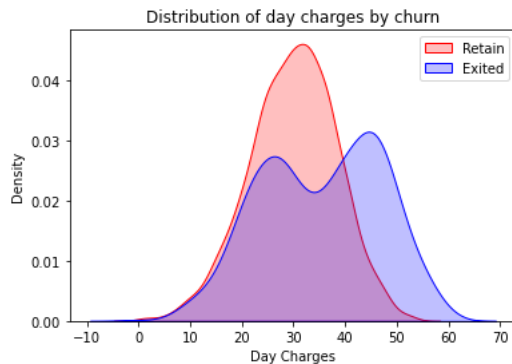
C:\Users\Rajarshi\anaconda3\lib\site-packages\seaborn_oldcore.py:1498: FutureWarning:

is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

C:\Users\Rajarshi\anaconda3\lib\site-packages\seaborn_oldcore.py:1119: FutureWarning:

use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

Out[17]: Text(0.5, 1.0, 'Distribution of day charges by churn')



```
In [18]: # Churn by evening charges
ax = sns.kdeplot(df.total_eve_charge[(df["class"] == 0)],
                 color = "Red", shade = True)
ax = sns.kdeplot(df.total_eve_charge[(df["class"] == 1)],
                 color = "Blue", shade = True)

ax.legend(["Retain", "Exited"], loc = "upper right")
ax.set_ylabel("Density")
ax.set_xlabel("Evening Charges")
ax.set_title("Distribution of evening charges by churn")
```

C:\Users\Rajarshi\AppData\Local\Temp\ipykernel_13116\601002929.py:2: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

C:\Users\Rajarshi\anaconda3\lib\site-packages\seaborn_oldcore.py:1498: FutureWarning:

is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

C:\Users\Rajarshi\anaconda3\lib\site-packages\seaborn_oldcore.py:1119: FutureWarning:

use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

C:\Users\Rajarshi\AppData\Local\Temp\ipykernel_13116\601002929.py:4: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

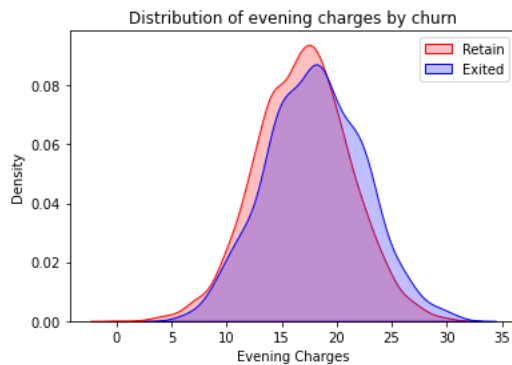
C:\Users\Rajarshi\anaconda3\lib\site-packages\seaborn_oldcore.py:1498: FutureWarning:

is_categorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead

C:\Users\Rajarshi\anaconda3\lib\site-packages\seaborn_oldcore.py:1119: FutureWarning:

use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

Out[18]: Text(0.5, 1.0, 'Distribution of evening charges by churn')



Telecom Churn Prediction : Data Preprocessing

```
In [19]: ## Preparing the data for machine Learning modeling by separating the features (X) from the target variable (Y).

X = df.drop(['class', 'area_code', 'phone_number'], axis='columns')
Y = df['class']
```

```
In [20]: from sklearn.model_selection import train_test_split ## function is commonly used to split datasets into random train and test sets
x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size = 0.2) ## Line splits the features (X) and the target variable (Y)
```

Telecom Churn Prediction : Feature Selection

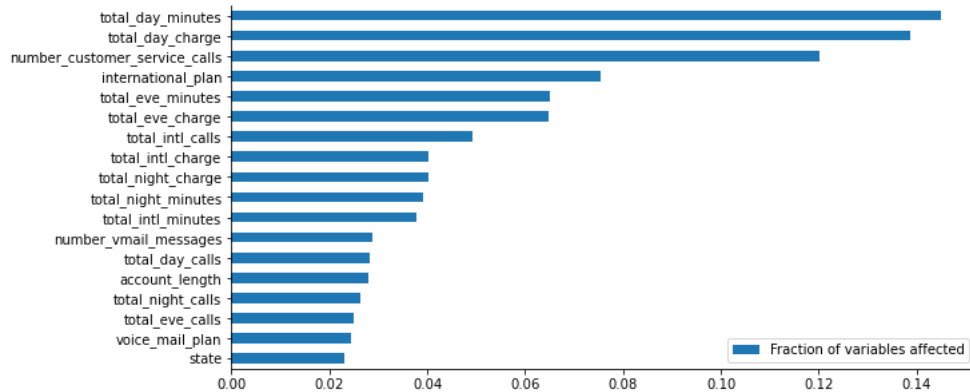
```
In [21]: from sklearn.ensemble import RandomForestClassifier
RF = RandomForestClassifier()
RF.fit(x_train,y_train) ## Line trains the random forest classifier model using the training data. The fit method fits th
```

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

Out[21]: RandomForestClassifier()

```
In [22]: # Plot the feature importance
feature_scores= pd.DataFrame({"Fraction of variables affected" : RF.feature_importances_},index = X.columns)
feature_scores= feature_scores.sort_values(by = "Fraction of variables affected")
feature_scores.plot(kind = "barh", figsize = (10, 5))
sns.despine()
```



Telecom Churn Prediction : Model Evaluation

- Logistic Regression

```
In [23]: from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix

Grid={ "C":np.logspace(-3,3,7), "penalty":["l1","l2"]}# L1 Lasso L2 ridge
LR_Model = LogisticRegression()
LR_Model_CV=GridSearchCV(LR_Model,Grid,cv=10) ## Line creates a GridSearchCV object named LR_Model_CV. GridSearchCV is a
LR_Model_CV.fit(x_train, y_train) ## Line fits the GridSearchCV object to the training data. It performs an exhaustive se
print("tuned hyperparameters :(best parameters) ",LR_Model_CV.best_params_)
```

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\linear_model_logistic.py:814: ConvergenceWarning:

lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html> (<https://scikit-learn.org/stable/modules/preprocessing.html>)

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression (https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

In [24]: `strength), and the penalty parameter specifies the type of regularization, where "l2" corresponds to Ridge regularization`

`each class. The classification_report function compares the actual target values (y_test) with the predicted values (y_pr`

	precision	recall	f1-score	support
0	0.87	1.00	0.93	862
1	0.73	0.06	0.11	138
accuracy			0.87	1000
macro avg	0.80	0.53	0.52	1000
weighted avg	0.85	0.87	0.81	1000

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\linear_model_logistic.py:814: ConvergenceWarning:

lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html> (<https://scikit-learn.org/stable/modules/preprocessing.html>)

Please also refer to the documentation for alternative solver options:

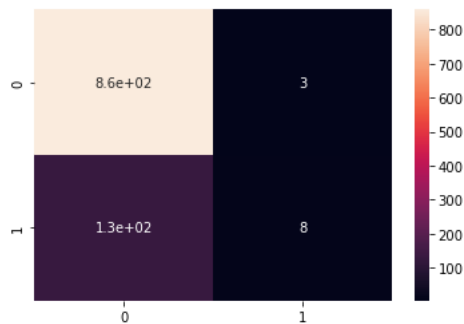
https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression (https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

In [25]: `cm = confusion_matrix(y_test, y_pred) ## This line computes the confusion matrix for the classifier's predictions on the
sns.heatmap(cm, annot = True) ## This line creates a heatmap using seaborn (sns) library, visualizing the confusion matrix`

Out[25]: <AxesSubplot:>



Telecom Churn Prediction : Model Evaluation

- Support Vector Machine


```
In [26]: from sklearn.calibration import CalibratedClassifierCV
from sklearn.svm import LinearSVC

SVM_Model = LinearSVC(max_iter = 10000)
SVM_Model = CalibratedClassifierCV()
SVM_Model.fit(x_train, y_train)
```

```
C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\svm\_base.py:1206: ConvergenceWarning:
Liblinear failed to converge, increase the number of iterations.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\svm\_base.py:1206: ConvergenceWarning:
Liblinear failed to converge, increase the number of iterations.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\svm\_base.py:1206: ConvergenceWarning:
Liblinear failed to converge, increase the number of iterations.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\svm\_base.py:1206: ConvergenceWarning:
Liblinear failed to converge, increase the number of iterations.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.
```

```
Out[26]: CalibratedClassifierCV()
```

```
In [27]: y_pred = SVM_Model.predict(x_test) ## This line uses the trained SVM model (SVM_Model) to predict the target variable (y_test)
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.87	0.99	0.93	862
1	0.50	0.07	0.12	138
accuracy			0.86	1000
macro avg	0.68	0.53	0.52	1000
weighted avg	0.82	0.86	0.81	1000

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

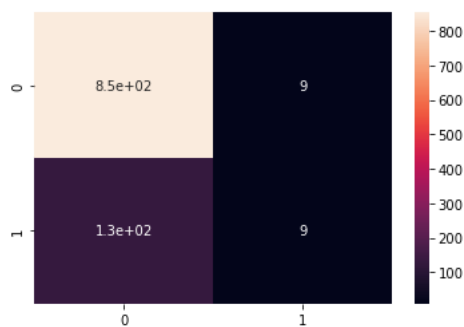
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

```
In [28]: cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot = True)
```

Out[28]: <AxesSubplot:>



Telecom Churn Prediction : Model Evaluation

- Random Forest Classifier

```
In [29]: from sklearn.ensemble import RandomForestClassifier
```

```
RF_Model = RandomForestClassifier()
RF_Model.fit(x_train, y_train)
```

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

Out[29]: RandomForestClassifier()

```
In [30]: y_pred = RF_Model.predict(x_test)
print(classification_report(y_test, y_pred))
```

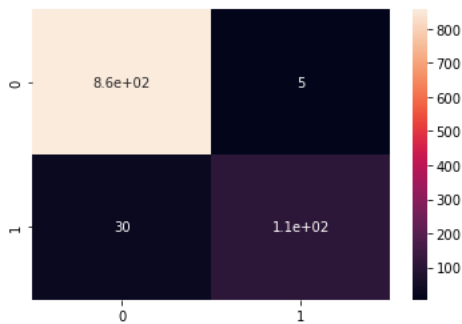
	precision	recall	f1-score	support
0	0.97	0.99	0.98	862
1	0.96	0.78	0.86	138
accuracy			0.96	1000
macro avg	0.96	0.89	0.92	1000
weighted avg	0.96	0.96	0.96	1000

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

```
In [31]: cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot = True)
```

Out[31]: <AxesSubplot:>



Telecom Churn Prediction : Model Evaluation

- K-Nearest Neighbour

```
In [32]: from sklearn.neighbors import KNeighborsClassifier

KNN = KNeighborsClassifier()
k_range = list(range(1, 31))
param_grid = dict(n_neighbors=k_range)

# defining parameter range
grid = GridSearchCV(KNN, param_grid, cv=10, scoring='accuracy', return_train_score=False, verbose=1)

# fitting the model for grid search
grid_search=grid.fit(x_train, y_train)

print(grid_search.best_params_)
```

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

```
In [34]: KNN_Model = KNeighborsClassifier(n_neighbors=9)
KNN_Model.fit(x_train, y_train)
```

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

Out[34]: KNeighborsClassifier(n_neighbors=9)

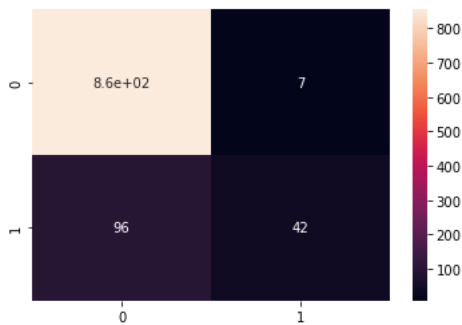
```
In [35]: y_pred = KNN_Model.predict(x_test)
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.90	0.99	0.94	862
1	0.86	0.30	0.45	138
accuracy			0.90	1000
macro avg	0.88	0.65	0.70	1000
weighted avg	0.89	0.90	0.88	1000

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

```
In [36]: cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot = True)
```

Out[36]: <AxesSubplot:>



Telecom Churn Prediction : Model Evaluation

- Naive Bayes Classifier

```
In [37]: from sklearn.naive_bayes import GaussianNB
GNB_Model = GaussianNB()
GNB_Model.fit(x_train, y_train)
```

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

Out[37]: GaussianNB()

```
In [38]: y_pred = GNB_Model.predict(x_test)
print(classification_report(y_test, y_pred))
```

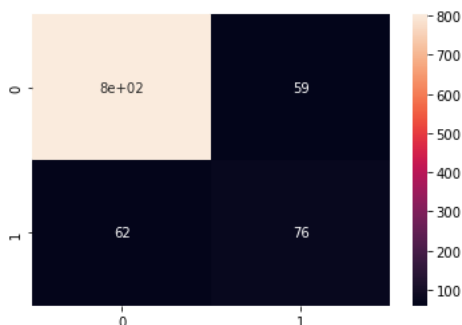
	precision	recall	f1-score	support
0	0.93	0.93	0.93	862
1	0.56	0.55	0.56	138
accuracy			0.88	1000
macro avg	0.75	0.74	0.74	1000
weighted avg	0.88	0.88	0.88	1000

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

```
In [39]: cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot = True)
```

Out[39]: <AxesSubplot:>



Comparing Models

```
In [40]: # ROC curve
from sklearn.metrics import roc_curve

fpr1, tpr1, thresh1 = roc_curve(y_test, LR_Model.predict_proba(x_test)[: , 1], pos_label = 1)
fpr2, tpr2, thresh2 = roc_curve(y_test, SVM_Model.predict_proba(x_test)[: , 1], pos_label = 1)
fpr3, tpr3, thresh3 = roc_curve(y_test, RF_Model.predict_proba(x_test)[: , 1], pos_label = 1)
fpr4, tpr4, thresh4 = roc_curve(y_test, KNN_Model.predict_proba(x_test)[: , 1], pos_label = 1)
fpr5, tpr5, thresh5 = roc_curve(y_test, GNB_Model.predict_proba(x_test)[: , 1], pos_label = 1)
```

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:

is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

```
In [41]: # AUC score
from sklearn.metrics import roc_auc_score

auc_score1 = roc_auc_score(y_test, LR_Model.predict_proba(x_test)[: , 1])
auc_score2 = roc_auc_score(y_test, SVM_Model.predict_proba(x_test)[: , 1])
auc_score3 = roc_auc_score(y_test, RF_Model.predict_proba(x_test)[: , 1])
auc_score4 = roc_auc_score(y_test, KNN_Model.predict_proba(x_test)[: , 1])
auc_score5 = roc_auc_score(y_test, GNB_Model.predict_proba(x_test)[: , 1])

print("Logistic Regression: ", auc_score1) # Logistic Regression
print("Support Vector Machine: ", auc_score2) # Support Vector Machine
print("Random Forest: ", auc_score3) # Random Forest
print("K-Nearest Neighbors: ", auc_score4) # K-Nearest Neighbors
print("Naive Bayes: ", auc_score5) # Naive Bayes
```

```
Logistic Regression: 0.6805877803557618
Support Vector Machine: 0.7119943508524159
Random Forest: 0.9080668482464104
K-Nearest Neighbors: 0.706626820006726
Naive Bayes: 0.8394868690944551
```

```
C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.

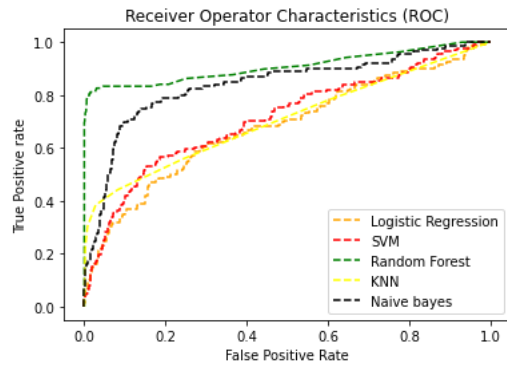
C:\Users\Rajarshi\anaconda3\lib\site-packages\sklearn\utils\validation.py:623: FutureWarning:
is_sparse is deprecated and will be removed in a future version. Check `isinstance(dtype, pd.SparseDtype)` instead.
```

AUC ranges in value from 0 to 1. A model whose predictions are 100% wrong has an AUC of 0.0; one whose predictions are 100% correct has an AUC of 1.0.
Random Forest: 0.9080668482464104 is a better model for prediction.

```
In [42]: ▶ plt.plot(fpr1, tpr1, linestyle = "--", color = "orange", label = "Logistic Regression")
plt.plot(fpr2, tpr2, linestyle = "--", color = "red", label = "SVM")
plt.plot(fpr3, tpr3, linestyle = "--", color = "green", label = "Random Forest")
plt.plot(fpr4, tpr4, linestyle = "--", color = "yellow", label = "KNN")
plt.plot(fpr5, tpr5, linestyle = "--", color = "black", label = "Naive bayes")

plt.title('Receiver Operator Characteristics (ROC)')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive rate')

plt.legend(loc = 'best')
plt.savefig('ROC', dpi = 300)
plt.show()
```



In []: ▶