Cross selling Recommendation Report

Name: Shreya Ramachandra

Email: ramacha4@uwindsor.ca

Country: Canada

Company: Data Glacier

Specialization: Data Analytics

Problem Description

Despite having a strong track record of selling various banking products, such as credit cards, deposit accounts, and retirement accounts, the XYZ credit union in Latin America is struggling to persuade its current customers to purchase additional products or services. This lack of cross-selling is impacting the credit union's overall performance in terms of revenue, profits, and customer retention.

Data Understanding

1. **Used Pandas and other libraries to load the data into a Pandas DataFrame**: This step involves loading the raw data into a structure called a DataFrame, which is a 2-dimensional labeled data structure with columns of potentially different types. Pandas is a popular library for data analysis and manipulation, and it provides convenient methods for loading data from various sources into a data frame.
2. **Checked for and handled duplicates**: Duplicate records can occur in data sets due to various reasons such as human error, system errors, or data import issues. To handle duplicates in the data, a unique identifier such as **customer code** is used to identify and remove duplicate records.
3. **Ensured data types are correct and consistent**: It is important to ensure that data types are correct and consistent in every column. This can be done by checking the data types of each column and converting them to the correct type.
4. **Translated data from Spanish to English**: The translation of data from Spanish to English can improve the understanding of the data.
5. **Updated the customer leave date to the correct format**: Dates are commonly stored in different formats in data sets, so it is important to convert them to a consistent format to ensure that the data can be easily analyzed and processed.
6. **Corrected the data type of gender**: The gender column may contain various text representations for males and females, so it is important to standardize these values to two consistent categories (e.g. male, female).
7. **Check for missing values**: Missing values in data can impact the accuracy of the analysis, so it is important to check for missing values and decide on a strategy for handling them. One common strategy is to drop the missing values.
8. **Save the cleaned and transformed data for future use**: After cleaning and transforming the data, it is a good practice to save the processed data for future use to avoid repeating the process each time the data is needed.

Github:
https://github.com/ShreyaRamachandra/Portfolio_Project_Cross_selling_Recommendation-Bank-

jupyter **Cross Selling** Last Checkpoint: a few seconds ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

In [1]:
```
#Importing necessery libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import datetime

import csv
import warnings
from scipy import stats
from pandas.api.types import is_numeric_dtype
import matplotlib.pyplot as plt
import matplotlib as mpl

warnings.filterwarnings('ignore')
```

In [2]:
```
#importing data

df_test = pd.read_csv("D:/Internship/Data Glacier/Cross Selling/Test.csv")
```

In [3]:
```
#importing data

file= "D:/Internship/Data Glacier/Cross Selling/Train.csv"
df_train = pd.read_csv(file)
```

In [4]:
```
df_train.head(5)
```

Out[4]:

	fecha_dato	ncodpers	ind_empleado	pais_residencia	sexo	age	fecha_alta	ind_nuevo	antiguedad	indrel	...	ind_hip_fin_ult1	ind_plan_fin_ult1	ind_pres_
0	2015-01-28	1375586	N	ES	H	35	2015-01-12	0.0	6	1.0	...	0	0	
1	2015-01-28	1050611	N	ES	V	23	2012-08-10	0.0	35	1.0	...	0	0	

jupyter **Cross Selling** Last Checkpoint: a minute ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

Out[4]:

	fecha_dato	ncodpers	ind_empleado	pais_residencia	sexo	age	fecha_alta	ind_nuevo	antiguedad	indrel	...	ind_hip_fin_ult1	ind_plan_fin_ult1	ind_pres_
0	2015-01-28	1375586	N	ES	H	35	2015-01-12	0.0	6	1.0	...	0	0	
1	2015-01-28	1050611	N	ES	V	23	2012-08-10	0.0	35	1.0	...	0	0	
2	2015-01-28	1050612	N	ES	V	23	2012-08-10	0.0	35	1.0	...	0	0	
3	2015-01-28	1050613	N	ES	H	22	2012-08-10	0.0	35	1.0	...	0	0	
4	2015-01-28	1050614	N	ES	V	23	2012-08-10	0.0	35	1.0	...	0	0	

5 rows × 48 columns

In [5]:
```
df_train.shape
```

Out[5]: (13647309, 48)

In [6]:
```
df_test.shape
```

Out[6]: (929615, 24)

In [7]:
```
#translating from spanish to english

dict = {'fecha_dato' : 'Date',
        'ncodpers' : 'Customer_Code',
        'ind_empleado' : 'Employee_Index',
        'pais_residencia' : 'Country',
        'sexo' : 'Gender',
        'age' : 'Age',
        'fecha_alta' : 'Customer_Join_Date',
        'ind_nuevo' : 'Customer_Index',
        'antiguedad' : 'Customer_Seniority',
```

jupyter Cross Selling Last Checkpoint: 2 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

```
        'ind_ctpp_fin_ult1' : 'Private_Plus_Account',
        'ind_deco_fin_ult1' : 'Short_Term_Deposits',
        'ind_deme_fin_ult1' : 'Medium_Term_Deposits',
        'ind_dela_fin_ult1' : 'Long_Term_Deposits',
        'ind_ecue_fin_ult1' : 'E_Account',
        'ind_fond_fin_ult1' : 'Funds',
        'ind_hip_fin_ult1' : 'Mortgage',
        'ind_plan_fin_ult1' : 'Pensions',
        'ind_pres_fin_ult1' : 'Loans',
        'ind_reca_fin_ult1' : 'Taxes',
        'ind_tjcr_fin_ult1' : 'Credit_Card',
        'ind_valo_fin_ult1' : 'Securities',
        'ind_viv_fin_ult1' : 'Home_Account',
        'ind_nomina_ult1' : 'Payroll',
        'ind_nom_pens_ult1' : 'Pensions2',
        'ind_recibo_ult1' : 'Direct_Debit'
       }
df_train = df_train.rename(columns = dict)
```

In [8]:
```
df_train.head()
```

Out[8]:

	Date	Customer_Code	Employee_Index	Country	Gender	Age	Customer_Join_Date	Customer_Index	Customer_Seniority	Primary_Customer	...	Mortgage
0	2015-01-28	1375586	N	ES	H	35	2015-01-12	0.0	6	1.0	...	0
1	2015-01-28	1050611	N	ES	V	23	2012-08-10	0.0	35	1.0	...	0
2	2015-01-28	1050612	N	ES	V	23	2012-08-10	0.0	35	1.0	...	0
3	2015-01-28	1050613	N	ES	H	22	2012-08-10	0.0	35	1.0	...	0
4	2015-01-28	1050614	N	ES	V	23	2012-08-10	0.0	35	1.0	...	0

jupyter Cross Selling Last Checkpoint: a few seconds ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

Run Code

In [9]:
```
#checking for null values
df_train.isnull().sum()
```

Out[9]:
```
Date                        0
Customer_Code               0
Employee_Index          27734
Country                 27734
Gender                  27804
Age                         0
Customer_Join_Date      27734
Customer_Index          27734
Customer_Seniority          0
Primary_Customer        27734
Customer_Leave_Date     13622516
Customer_Type           149781
Customer_Relation       149781
Residence_Index         27734
Foriegner_Index         27734
Spouse_Index            13645501
Channel_Used            186126
Deceased_Index          27734
Primary_Address         27735
Customer_Address        93591
Province                93591
Activity_Index          27734
Gross_Income            2794375
Segmentation            189368
Saving_Account              0
Guarantees                  0
Current_Accounts            0
Derivative_Account          0
Payroll_Account             0
Junior_Account              0
More_Private_Account        0
Private_Account             0
Private_Plus_Account        0
Short_Term_Deposits         0
```

jupyter Cross Selling Last Checkpoint: a few seconds ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

In [10]:
```
#Removing all the rows with NA values
df_train = df_train.dropna()
df_train.isna().any()
```

Out[10]:
```
Date                    False
Customer_Code           False
Employee_Index          False
Country                 False
Gender                  False
Age                     False
Customer_Join_Date      False
Customer_Index          False
Customer_Seniority      False
Primary_Customer        False
Customer_Leave_Date     False
Customer_Type           False
Customer_Relation       False
Residence_Index         False
Foriegner_Index         False
Spouse_Index            False
Channel_Used            False
Deceased_Index          False
Primary_Address         False
Customer_Address        False
Province                False
Activity_Index          False
Gross_Income            False
Segmentation            False
Saving_Account          False
Guarantees              False
Current_Accounts        False
Derivative_Account      False
Payroll_Account         False
Junior_Account          False
More_Private_Account    False
Private_Account         False
Private_Plus_Account    False
```

jupyter Cross Selling Last Checkpoint: a few seconds ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

In [11]:
```
df_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 0 entries
Data columns (total 48 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   Date                  0 non-null      object
 1   Customer_Code         0 non-null      int64
 2   Employee_Index        0 non-null      object
 3   Country               0 non-null      object
 4   Gender                0 non-null      object
 5   Age                   0 non-null      object
 6   Customer_Join_Date    0 non-null      object
 7   Customer_Index        0 non-null      float64
 8   Customer_Seniority    0 non-null      object
 9   Primary_Customer      0 non-null      float64
 10  Customer_Leave_Date   0 non-null      object
 11  Customer_Type         0 non-null      object
 12  Customer_Relation     0 non-null      object
 13  Residence_Index       0 non-null      object
 14  Foriegner_Index       0 non-null      object
 15  Spouse_Index          0 non-null      object
 16  Channel_Used          0 non-null      object
 17  Deceased_Index        0 non-null      object
 18  Primary_Address       0 non-null      float64
 19  Customer_Address      0 non-null      float64
 20  Province              0 non-null      object
 21  Activity_Index        0 non-null      float64
 22  Gross_Income          0 non-null      float64
 23  Segmentation          0 non-null      object
 24  Saving_Account        0 non-null      int64
 25  Guarantees            0 non-null      int64
 26  Current_Accounts      0 non-null      int64
 27  Derivative_Account    0 non-null      int64
 28  Payroll_Account       0 non-null      int64
```

```
 37  Funds                0 non-null      int64
 38  Mortgage             0 non-null      int64
 39  Pensions             0 non-null      int64
 40  Loans                0 non-null      int64
 41  Taxes                0 non-null      int64
 42  Credit_Card          0 non-null      int64
 43  Securities           0 non-null      int64
 44  Home_Account         0 non-null      int64
 45  Payroll              0 non-null      float64
 46  Pensions2            0 non-null      float64
 47  Direct_Debit         0 non-null      int64
dtypes: float64(8), int64(23), object(17)
memory usage: 0.0+ bytes
```

In [12]:
```
#changing datatypes

for column in ["Employee_Index", "Country", "Gender"]:
    df_train[column] = df_train[column].astype('category')
```

In [13]:
```
for column in ["Date", "Customer_Join_Date", "Customer_Leave_Date"]:
    df_train[column] = df_train[column].astype('datetime64[ns]')
```

In [14]:
```
df = df_train.copy()
```

In [15]:
```
pd.set_option('display.max_columns', None)
```

In [16]:
```
df.head()
```

Out[16]:

Date	Customer_Code	Employee_Index	Country	Gender	Age	Customer_Join_Date	Customer_Index	Customer_Seniority	Primary_Customer	Customer_Leav

In [17]:
```
#dropping duplicates
df = df.drop_duplicates(subset='Customer_Code', keep="last")
```