Group Name： Solo

Name: Vishnu Priya Malchetti

Email: vishnupriyam.sapbasis@gmail.com

Country: Ireland

Specialization: Data Analyst

Problem description:

We are determined to help XYZ Bank improve its cross-selling strategies and enhance customer engagement. The bank offers a wide array of financial products and services, including savings accounts, credit cards, mortgages, loans, and investment options. However, we've observed that many of our customers have limited product adoption and aren't fully utilizing the range of services available to them. To tackle this challenge head-on, we plan to implement customer segmentation techniques to gain deeper insights into our customer base. By dividing our customers into distinct groups based on their demographics, financial behavior, and product usage patterns, we hope to identify specific customer segments that are more likely to use products and services. Armed with this valuable information, we aim to create personalized marketing strategies and tailored cross-selling initiatives to boost customer satisfaction and encourage higher product adoption. As part of our data analysis team, the objective is to thoroughly analyze the extensive customer dataset provided by XYZ Bank and conduct a comprehensive customer segmentation analysis. The dataset includes detailed information about each customer, such as age, gender, income, transaction history, product holdings, and tenure with our bank.

EDA:

Missing values checks

```
[30]: #missing values checking
      df1.isnull().sum()
```

```
[30]: fecha_dato                0
      ncodpers                  0
      ind_empleado          27734
      pais_residencia       27734
      sexo                  27804
      age                       0
      fecha_alta            27734
      ind_nuevo             27734
      antiguedad                0
      indrel                27734
      ult_fec_cli_1t     13622516
      indrel_1mes          149781
      tiprel_1mes          149781
      indresi               27734
      indext                27734
      conyuemp           13645501
      canal_entrada        186126
      indfall               27734
```

```
[34]: df1['indrel']=df1['indrel'].fillna(df1['indrel'].mean())

[35]: df1['tipodom']=df1['tipodom'].fillna(df1['tipodom'].mean())

[36]: df1['cod_prov']=df1['cod_prov'].fillna(df1['cod_prov'].mean())

[37]: df1['ind_actividad_cliente']=df1['ind_actividad_cliente'].fillna(df1['ind_actividad_cliente'].mean())

[38]: df1['renta']=df1['renta'].fillna(df1['renta'].mean())

[39]: df1['ind_nomina_ult1']=df1['ind_nomina_ult1'].fillna(df1['ind_nomina_ult1'].mean())

[40]: df1['ind_nom_pens_ult1']=df1['ind_nom_pens_ult1'].fillna(df1['ind_nom_pens_ult1'].mean())

[41]: df1.isnull().sum()

[41]: fecha_dato                0
      ncodpers                  0
      ind_empleado          27734
```

Fill missing values in object columns with the most frequent value

[45]:
```
object_columns_with_nulls = [
    'ind_empleado', 'pais_residencia', 'sexo', 'fecha_alta',
    'ult_fec_cli_1t', 'indrel_1mes', 'tiprel_1mes', 'indresi',
    'indext', 'conyuemp', 'canal_entrada', 'indfall', 'nomprov', 'segmento'
]

# Fill missing values in object columns with the most frequent value
for col in object_columns_with_nulls:
    most_frequent_value = df1[col].mode()[0]
    df1[col].fillna(most_frequent_value, inplace=True)

# Verify if there are any remaining missing values
remaining_data= df1.isnull().sum()

print("Remaining data:")
print(remaining_data)
```

```
Remaining data:
fecha_dato              0
ncodpers                0
ind_empleado            0
pais_residencia         0
sexo                    0
age                     0
fecha_alta              0
```

[46]:
```
df1.describe().round()
```

[46]:

	ncodpers	ind_nuevo	indrel	tipodom	cod_prov	ind_actividad_cliente	renta	ind_ahor_fin_ult1	ind_aval_fin_ult1	ind_cco_fin
count	13647309.0	13647309.0	13647309.0	13647309.0	13647309.0	13647309.0	13647309.0	13647309.0	13647309.0	13647
mean	834904.0	0.0	1.0	1.0	27.0	0.0	134254.0	0.0	0.0	
std	431565.0	0.0	4.0	0.0	13.0	0.0	205659.0	0.0	0.0	
min	15889.0	0.0	1.0	1.0	1.0	0.0	1203.0	0.0	0.0	
25%	452813.0	0.0	1.0	1.0	15.0	0.0	76437.0	0.0	0.0	
50%	931893.0	0.0	1.0	1.0	28.0	0.0	124680.0	0.0	0.0	
75%	1199286.0	0.0	1.0	1.0	34.0	1.0	137452.0	0.0	0.0	
max	1553689.0	1.0	99.0	1.0	52.0	1.0	28894396.0	1.0	1.0	

8 rows × 31 columns

Summary statistics of the data

```
#  Exploratory Data Analysis
# Summary statistics
print("\nSummary statistics:")
print(df1.describe())
```

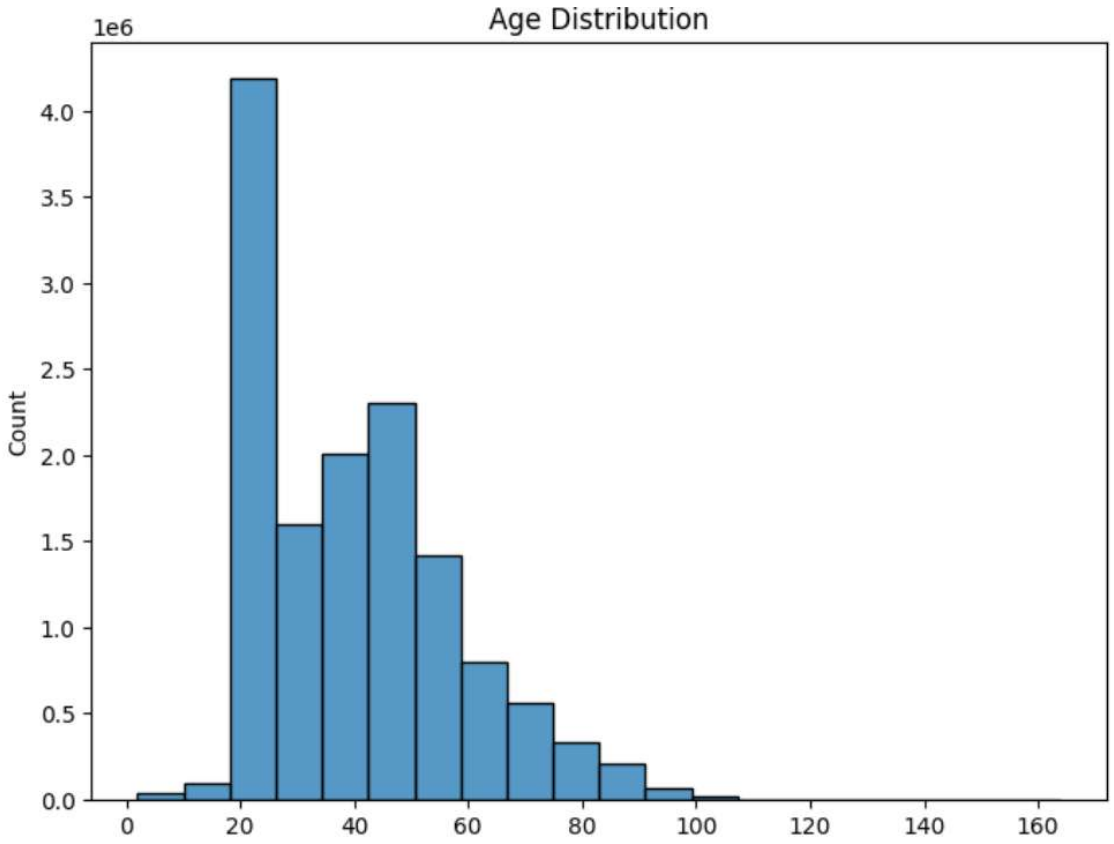
```
Summary statistics:
            ncodpers     ind_nuevo        indrel     tipodom      cod_prov  \
count  1.364731e+07  1.364731e+07  1.364731e+07  13647309.0  1.364731e+07
mean   8.349042e+05  5.956184e-02  1.178399e+00         1.0  2.657147e+01
std    4.315650e+05  2.364327e-01  4.173222e+00         0.0  1.274011e+01
min    1.588900e+04  0.000000e+00  1.000000e+00         1.0  1.000000e+00
25%    4.528130e+05  0.000000e+00  1.000000e+00         1.0  1.500000e+01
50%    9.318930e+05  0.000000e+00  1.000000e+00         1.0  2.800000e+01
75%    1.199286e+06  0.000000e+00  1.000000e+00         1.0  3.400000e+01
max    1.553689e+06  1.000000e+00  9.900000e+01         1.0  5.200000e+01

       ind_actividad_cliente         renta  ind_ahor_fin_ult1  \
count           1.364731e+07  1.364731e+07       1.364731e+07
mean            4.578105e-01  1.342543e+05       1.022912e-04
std             4.977104e-01  2.056589e+05       1.011340e-02
min             0.000000e+00  1.202730e+03       0.000000e+00
25%             0.000000e+00  7.643715e+04       0.000000e+00
50%             0.000000e+00  1.246800e+05       0.000000e+00
75%             1.000000e+00  1.374521e+05       0.000000e+00
```

```
       ind_plan_fin_ult1  ind_pres_fin_ult1  ind_reca_fin_ult1  \
count       1.364731e+07       1.364731e+07       1.364731e+07
mean        9.170965e-03       2.627404e-03       5.253636e-02
std         9.532502e-02       5.119083e-02       2.231060e-01
min         0.000000e+00       0.000000e+00       0.000000e+00
25%         0.000000e+00       0.000000e+00       0.000000e+00
50%         0.000000e+00       0.000000e+00       0.000000e+00
75%         0.000000e+00       0.000000e+00       0.000000e+00
max         1.000000e+00       1.000000e+00       1.000000e+00

       ind_tjcr_fin_ult1  ind_valo_fin_ult1  ind_viv_fin_ult1  \
count       1.364731e+07       1.364731e+07      1.364731e+07
mean        4.438868e-02       2.560761e-02      3.847718e-03
std         2.059571e-01       1.579616e-01      6.191053e-02
min         0.000000e+00       0.000000e+00      0.000000e+00
25%         0.000000e+00       0.000000e+00      0.000000e+00
50%         0.000000e+00       0.000000e+00      0.000000e+00
75%         0.000000e+00       0.000000e+00      0.000000e+00
max         1.000000e+00       1.000000e+00      1.000000e+00

       ind_nomina_ult1  ind_nom_pens_ult1  ind_recibo_ult1
count     1.364731e+07       1.364731e+07     1.364731e+07
mean      5.472434e-02       5.942854e-02     1.279162e-01
std       2.273075e-01       2.362858e-01     3.339965e-01
min       0.000000e+00       0.000000e+00     0.000000e+00
```

Age distribution in the data

[49]:
```
df1['age'] = pd.to_numeric(df1['age'], errors='coerce')
```

[50]:
```
# Age distribution
plt.figure(figsize=(8, 6))
sns.histplot(df1['age'],bins=20)
plt.title("Age Distribution")
plt.xlabel("Age")
plt.ylabel("Count")
plt.show()
```



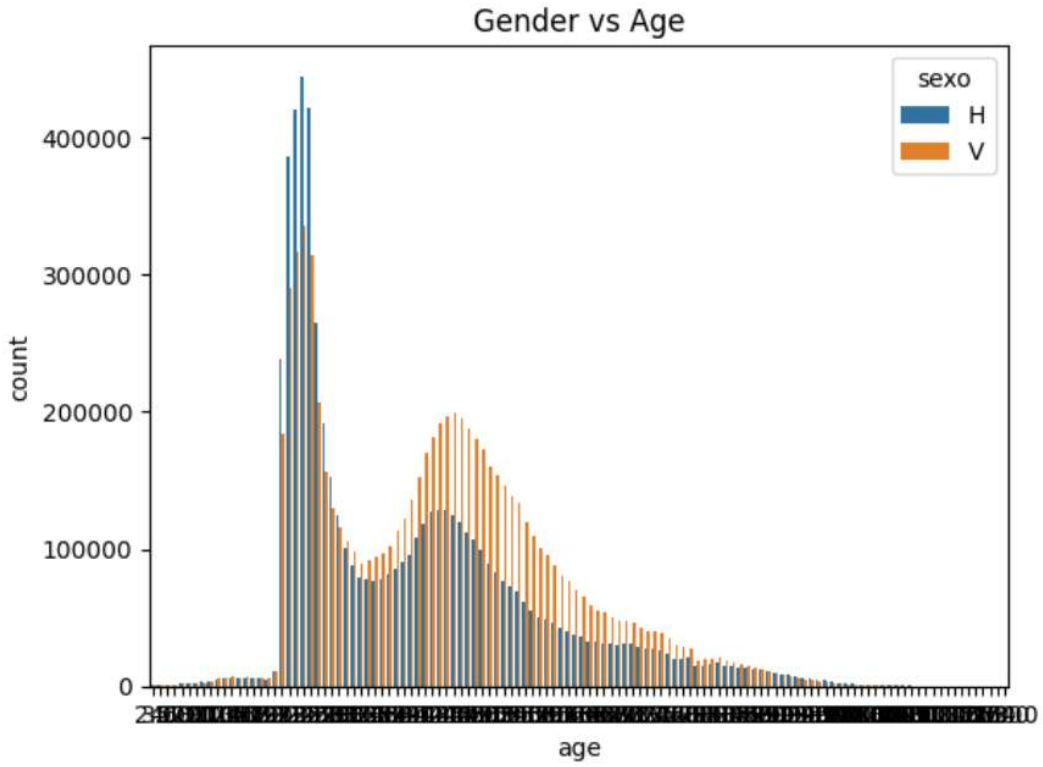
Recommendations For Future Modeling

1. We can see from the Age distribution plot that, majority of customers are between ages 20 and 50 meaning the products are pertronized by the working force.
2. In the future, company would want to make changes to their advertizement to be able to retain most of their young customer because, they tend to want to explore other options.
3. cod_prov has an average of 27.0 with a standard deviation of 13 and might be a variable to consider in this cross-selling recommendation system.
4. 4.ncodpers and renta variables might be significant in our model building

Gender VS Age analysis from the data

```
[56]: sns.countplot(data=df1,x=df1['age'],hue=df1['sexo']).set_title("Gender vs Age")
```

```
[56]: Text(0.5, 1.0, 'Gender vs Age')
```

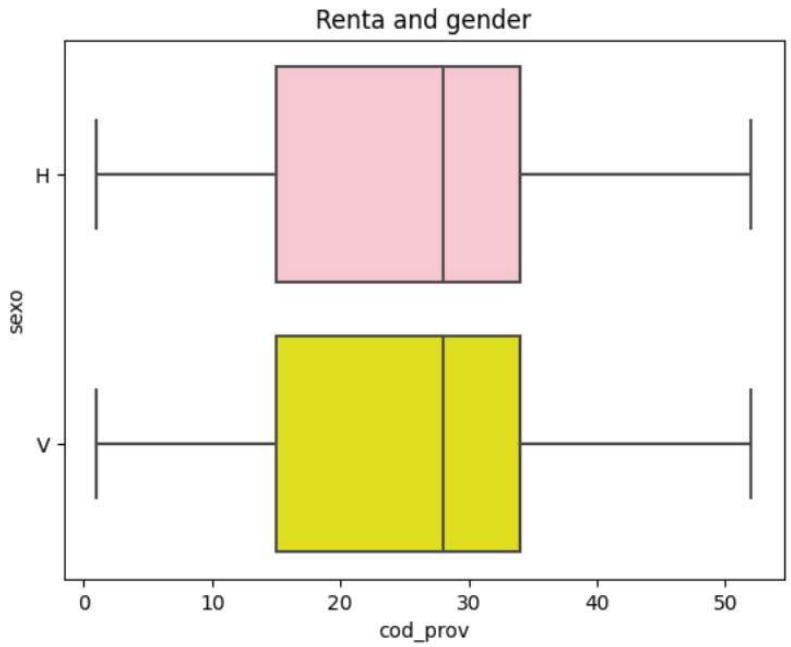


Here is the plot which showing gender and age of customers.

Box plot for the rent and gender

[78]:
```
sns.boxplot(data=df1, x='cod_prov', y='sexo', palette=['pink','yellow']).set_title("cod_prov and gender")
```

[78]:
```
Text(0.5, 1.0, 'Renta and gender')
```



For different gender, Province code (customer's address) are spreading nearly the same.