{

"cells": [

{

"cell\_type": "markdown",

"id": "6016dd3d",

"metadata": {

"id": "6016dd3d"

},

"source": [

"# Assignment 2\n",

"# Data Visualization and Pre-processing"

]

},

{

"cell\_type": "markdown",

"id": "717ccc1b",

"metadata": {

"id": "717ccc1b"

},

"source": [

"## 1. Perform Below Visualizations.\n",

"### Univariate Analysis\n",

"#### 1. Summary Statistics"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "19c48fa6",

"metadata": {

"id": "19c48fa6"

},

"outputs": [],

"source": [

"import pandas as pd\n",

"import numpy as np\n",

"import matplotlib.pyplot as plt\n",

"import seaborn as sns\n",

"import statsmodels.api as sm"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "01105cab",

"metadata": {

"id": "01105cab",

"outputId": "06dbee39-b7b2-4264-f007-73868add91a2"

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>RowNumber</th>\n",

" <th>CustomerId</th>\n",

" <th>Surname</th>\n",

" <th>CreditScore</th>\n",

" <th>Geography</th>\n",

" <th>Gender</th>\n",

" <th>Age</th>\n",

" <th>Tenure</th>\n",

" <th>Balance</th>\n",

" <th>NumOfProducts</th>\n",

" <th>HasCrCard</th>\n",

" <th>IsActiveMember</th>\n",

" <th>EstimatedSalary</th>\n",

" <th>Exited</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1</td>\n",

" <td>15634602</td>\n",

" <td>Hargrave</td>\n",

" <td>619</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>42</td>\n",

" <td>2</td>\n",

" <td>0.00</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>101348.88</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>2</td>\n",

" <td>15647311</td>\n",

" <td>Hill</td>\n",

" <td>608</td>\n",

" <td>Spain</td>\n",

" <td>Female</td>\n",

" <td>41</td>\n",

" <td>1</td>\n",

" <td>83807.86</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>112542.58</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>3</td>\n",

" <td>15619304</td>\n",

" <td>Onio</td>\n",

" <td>502</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>42</td>\n",

" <td>8</td>\n",

" <td>159660.80</td>\n",

" <td>3</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>113931.57</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>4</td>\n",

" <td>15701354</td>\n",

" <td>Boni</td>\n",

" <td>699</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>39</td>\n",

" <td>1</td>\n",

" <td>0.00</td>\n",

" <td>2</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>93826.63</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>5</td>\n",

" <td>15737888</td>\n",

" <td>Mitchell</td>\n",

" <td>850</td>\n",

" <td>Spain</td>\n",

" <td>Female</td>\n",

" <td>43</td>\n",

" <td>2</td>\n",

" <td>125510.82</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>79084.10</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9995</th>\n",

" <td>9996</td>\n",

" <td>15606229</td>\n",

" <td>Obijiaku</td>\n",

" <td>771</td>\n",

" <td>France</td>\n",

" <td>Male</td>\n",

" <td>39</td>\n",

" <td>5</td>\n",

" <td>0.00</td>\n",

" <td>2</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>96270.64</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9996</th>\n",

" <td>9997</td>\n",

" <td>15569892</td>\n",

" <td>Johnstone</td>\n",

" <td>516</td>\n",

" <td>France</td>\n",

" <td>Male</td>\n",

" <td>35</td>\n",

" <td>10</td>\n",

" <td>57369.61</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>101699.77</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9997</th>\n",

" <td>9998</td>\n",

" <td>15584532</td>\n",

" <td>Liu</td>\n",

" <td>709</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>36</td>\n",

" <td>7</td>\n",

" <td>0.00</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>42085.58</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9998</th>\n",

" <td>9999</td>\n",

" <td>15682355</td>\n",

" <td>Sabbatini</td>\n",

" <td>772</td>\n",

" <td>Germany</td>\n",

" <td>Male</td>\n",

" <td>42</td>\n",

" <td>3</td>\n",

" <td>75075.31</td>\n",

" <td>2</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>92888.52</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9999</th>\n",

" <td>10000</td>\n",

" <td>15628319</td>\n",

" <td>Walker</td>\n",

" <td>792</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>28</td>\n",

" <td>4</td>\n",

" <td>130142.79</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>38190.78</td>\n",

" <td>0</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>10000 rows × 14 columns</p>\n",

"</div>"

],

"text/plain": [

" RowNumber CustomerId Surname CreditScore Geography Gender Age \\\n",

"0 1 15634602 Hargrave 619 France Female 42 \n",

"1 2 15647311 Hill 608 Spain Female 41 \n",

"2 3 15619304 Onio 502 France Female 42 \n",

"3 4 15701354 Boni 699 France Female 39 \n",

"4 5 15737888 Mitchell 850 Spain Female 43 \n",

"... ... ... ... ... ... ... ... \n",

"9995 9996 15606229 Obijiaku 771 France Male 39 \n",

"9996 9997 15569892 Johnstone 516 France Male 35 \n",

"9997 9998 15584532 Liu 709 France Female 36 \n",

"9998 9999 15682355 Sabbatini 772 Germany Male 42 \n",

"9999 10000 15628319 Walker 792 France Female 28 \n",

"\n",

" Tenure Balance NumOfProducts HasCrCard IsActiveMember \\\n",

"0 2 0.00 1 1 1 \n",

"1 1 83807.86 1 0 1 \n",

"2 8 159660.80 3 1 0 \n",

"3 1 0.00 2 0 0 \n",

"4 2 125510.82 1 1 1 \n",

"... ... ... ... ... ... \n",

"9995 5 0.00 2 1 0 \n",

"9996 10 57369.61 1 1 1 \n",

"9997 7 0.00 1 0 1 \n",

"9998 3 75075.31 2 1 0 \n",

"9999 4 130142.79 1 1 0 \n",

"\n",

" EstimatedSalary Exited \n",

"0 101348.88 1 \n",

"1 112542.58 0 \n",

"2 113931.57 1 \n",

"3 93826.63 0 \n",

"4 79084.10 0 \n",

"... ... ... \n",

"9995 96270.64 0 \n",

"9996 101699.77 0 \n",

"9997 42085.58 1 \n",

"9998 92888.52 1 \n",

"9999 38190.78 0 \n",

"\n",

"[10000 rows x 14 columns]"

]

},

"execution\_count": 2,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data = pd.read\_csv('C:\\Karthikvignesh\\Churn\_Modelling.csv')\n",

"file\_data"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "cdd604b1",

"metadata": {

"id": "cdd604b1",

"outputId": "ebcc6576-a219-4c17-d494-5c681bb5e0ab"

},

"outputs": [

{

"data": {

"text/plain": [

"76485.88928799961"

]

},

"execution\_count": 3,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data['Balance'].mean()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "1b444cfb",

"metadata": {

"id": "1b444cfb",

"outputId": "ad0317d6-57b1-4f4f-bb16-007014886e8f"

},

"outputs": [

{

"data": {

"text/plain": [

"97198.54000000001"

]

},

"execution\_count": 4,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data['Balance'].median()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "827a51fc",

"metadata": {

"id": "827a51fc",

"outputId": "34afa18c-5340-4016-bef7-1ed7f5e56311"

},

"outputs": [

{

"data": {

"text/plain": [

"62397.40520238623"

]

},

"execution\_count": 5,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data['Balance'].std()"

]

},

{

"cell\_type": "markdown",

"id": "0376c8ac",

"metadata": {

"id": "0376c8ac"

},

"source": [

"#### 2. Frequency Table"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "828e96c3",

"metadata": {

"id": "828e96c3",

"outputId": "144d89ab-0c7b-4c6f-8ad9-139a1f3f3de9"

},

"outputs": [

{

"data": {

"text/plain": [

"Smith 32\n",

"Scott 29\n",

"Martin 29\n",

"Walker 28\n",

"Brown 26\n",

" ..\n",

"Izmailov 1\n",

"Bold 1\n",

"Bonham 1\n",

"Poninski 1\n",

"Burbidge 1\n",

"Name: Surname, Length: 2932, dtype: int64"

]

},

"execution\_count": 6,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data['Surname'].value\_counts()"

]

},

{

"cell\_type": "markdown",

"id": "accc0e74",

"metadata": {

"id": "accc0e74"

},

"source": [

"#### 3. Create Charts"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "a9803069",

"metadata": {

"id": "a9803069",

"outputId": "5eef1c85-1a22-41c3-94cd-a3b62ff3610a"

},

"outputs": [

{

"data": {

"text/plain": [

"<AxesSubplot:>"

]

},

"execution\_count": 7,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"file\_data.boxplot(column=['Balance'], grid=False)\n"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "815daa4c",

"metadata": {

"id": "815daa4c",

"outputId": "f51e1abc-e471-447e-fa13-81c1cac833e0"

},

"outputs": [

{

"data": {

"text/plain": [

"array([[<AxesSubplot:title={'center':'Balance'}>]], dtype=object)"

]

},

"execution\_count": 8,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"file\_data.hist(column='Balance', grid=False, edgecolor='black')"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "a698398a",

"metadata": {

"id": "a698398a",

"outputId": "9b25b3fd-7a2e-4f73-eec6-00d139c5e2a0"

},

"outputs": [

{

"data": {

"text/plain": [

"<AxesSubplot:xlabel='Balance', ylabel='Density'>"

]

},

"execution\_count": 9,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"sns.kdeplot(file\_data['Balance'])"

]

},

{

"cell\_type": "markdown",

"id": "333db4b1",

"metadata": {

"id": "333db4b1"

},

"source": [

"### Bi - Variate Analysis"

]

},

{

"cell\_type": "markdown",

"id": "87e3a8d3",

"metadata": {

"id": "87e3a8d3"

},

"source": [

"#### 1. Scatterplots"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "0e09d504",

"metadata": {

"id": "0e09d504",

"outputId": "eb557781-2bc3-4638-8e1a-db413d8cb63c"

},

"outputs": [

{

"data": {

"text/plain": [

"Text(0, 0.5, 'Age')"

]

},

"execution\_count": 10,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"plt.scatter(file\_data.CreditScore.head(100), file\_data.Age.head(100))\n",

"plt.title('Scatter')\n",

"plt.xlabel('CreditScore')\n",

"plt.ylabel('Age')"

]

},

{

"cell\_type": "markdown",

"id": "a3053dcd",

"metadata": {

"id": "a3053dcd"

},

"source": [

"#### 2. Correlation Coefficients"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "6e94ddc0",

"metadata": {

"id": "6e94ddc0",

"outputId": "5b01a183-71c3-490b-9f62-4e6b9207c160"

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>RowNumber</th>\n",

" <th>CustomerId</th>\n",

" <th>CreditScore</th>\n",

" <th>Age</th>\n",

" <th>Tenure</th>\n",

" <th>Balance</th>\n",

" <th>NumOfProducts</th>\n",

" <th>HasCrCard</th>\n",

" <th>IsActiveMember</th>\n",

" <th>EstimatedSalary</th>\n",

" <th>Exited</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>RowNumber</th>\n",

" <td>1.000000</td>\n",

" <td>0.004202</td>\n",

" <td>0.005840</td>\n",

" <td>0.000783</td>\n",

" <td>-0.006495</td>\n",

" <td>-0.009067</td>\n",

" <td>0.007246</td>\n",

" <td>0.000599</td>\n",

" <td>0.012044</td>\n",

" <td>-0.005988</td>\n",

" <td>-0.016571</td>\n",

" </tr>\n",

" <tr>\n",

" <th>CustomerId</th>\n",

" <td>0.004202</td>\n",

" <td>1.000000</td>\n",

" <td>0.005308</td>\n",

" <td>0.009497</td>\n",

" <td>-0.014883</td>\n",

" <td>-0.012419</td>\n",

" <td>0.016972</td>\n",

" <td>-0.014025</td>\n",

" <td>0.001665</td>\n",

" <td>0.015271</td>\n",

" <td>-0.006248</td>\n",

" </tr>\n",

" <tr>\n",

" <th>CreditScore</th>\n",

" <td>0.005840</td>\n",

" <td>0.005308</td>\n",

" <td>1.000000</td>\n",

" <td>-0.003965</td>\n",

" <td>0.000842</td>\n",

" <td>0.006268</td>\n",

" <td>0.012238</td>\n",

" <td>-0.005458</td>\n",

" <td>0.025651</td>\n",

" <td>-0.001384</td>\n",

" <td>-0.027094</td>\n",

" </tr>\n",

" <tr>\n",

" <th>Age</th>\n",

" <td>0.000783</td>\n",

" <td>0.009497</td>\n",

" <td>-0.003965</td>\n",

" <td>1.000000</td>\n",

" <td>-0.009997</td>\n",

" <td>0.028308</td>\n",

" <td>-0.030680</td>\n",

" <td>-0.011721</td>\n",

" <td>0.085472</td>\n",

" <td>-0.007201</td>\n",

" <td>0.285323</td>\n",

" </tr>\n",

" <tr>\n",

" <th>Tenure</th>\n",

" <td>-0.006495</td>\n",

" <td>-0.014883</td>\n",

" <td>0.000842</td>\n",

" <td>-0.009997</td>\n",

" <td>1.000000</td>\n",

" <td>-0.012254</td>\n",

" <td>0.013444</td>\n",

" <td>0.022583</td>\n",

" <td>-0.028362</td>\n",

" <td>0.007784</td>\n",

" <td>-0.014001</td>\n",

" </tr>\n",

" <tr>\n",

" <th>Balance</th>\n",

" <td>-0.009067</td>\n",

" <td>-0.012419</td>\n",

" <td>0.006268</td>\n",

" <td>0.028308</td>\n",

" <td>-0.012254</td>\n",

" <td>1.000000</td>\n",

" <td>-0.304180</td>\n",

" <td>-0.014858</td>\n",

" <td>-0.010084</td>\n",

" <td>0.012797</td>\n",

" <td>0.118533</td>\n",

" </tr>\n",

" <tr>\n",

" <th>NumOfProducts</th>\n",

" <td>0.007246</td>\n",

" <td>0.016972</td>\n",

" <td>0.012238</td>\n",

" <td>-0.030680</td>\n",

" <td>0.013444</td>\n",

" <td>-0.304180</td>\n",

" <td>1.000000</td>\n",

" <td>0.003183</td>\n",

" <td>0.009612</td>\n",

" <td>0.014204</td>\n",

" <td>-0.047820</td>\n",

" </tr>\n",

" <tr>\n",

" <th>HasCrCard</th>\n",

" <td>0.000599</td>\n",

" <td>-0.014025</td>\n",

" <td>-0.005458</td>\n",

" <td>-0.011721</td>\n",

" <td>0.022583</td>\n",

" <td>-0.014858</td>\n",

" <td>0.003183</td>\n",

" <td>1.000000</td>\n",

" <td>-0.011866</td>\n",

" <td>-0.009933</td>\n",

" <td>-0.007138</td>\n",

" </tr>\n",

" <tr>\n",

" <th>IsActiveMember</th>\n",

" <td>0.012044</td>\n",

" <td>0.001665</td>\n",

" <td>0.025651</td>\n",

" <td>0.085472</td>\n",

" <td>-0.028362</td>\n",

" <td>-0.010084</td>\n",

" <td>0.009612</td>\n",

" <td>-0.011866</td>\n",

" <td>1.000000</td>\n",

" <td>-0.011421</td>\n",

" <td>-0.156128</td>\n",

" </tr>\n",

" <tr>\n",

" <th>EstimatedSalary</th>\n",

" <td>-0.005988</td>\n",

" <td>0.015271</td>\n",

" <td>-0.001384</td>\n",

" <td>-0.007201</td>\n",

" <td>0.007784</td>\n",

" <td>0.012797</td>\n",

" <td>0.014204</td>\n",

" <td>-0.009933</td>\n",

" <td>-0.011421</td>\n",

" <td>1.000000</td>\n",

" <td>0.012097</td>\n",

" </tr>\n",

" <tr>\n",

" <th>Exited</th>\n",

" <td>-0.016571</td>\n",

" <td>-0.006248</td>\n",

" <td>-0.027094</td>\n",

" <td>0.285323</td>\n",

" <td>-0.014001</td>\n",

" <td>0.118533</td>\n",

" <td>-0.047820</td>\n",

" <td>-0.007138</td>\n",

" <td>-0.156128</td>\n",

" <td>0.012097</td>\n",

" <td>1.000000</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" RowNumber CustomerId CreditScore Age Tenure \\\n",

"RowNumber 1.000000 0.004202 0.005840 0.000783 -0.006495 \n",

"CustomerId 0.004202 1.000000 0.005308 0.009497 -0.014883 \n",

"CreditScore 0.005840 0.005308 1.000000 -0.003965 0.000842 \n",

"Age 0.000783 0.009497 -0.003965 1.000000 -0.009997 \n",

"Tenure -0.006495 -0.014883 0.000842 -0.009997 1.000000 \n",

"Balance -0.009067 -0.012419 0.006268 0.028308 -0.012254 \n",

"NumOfProducts 0.007246 0.016972 0.012238 -0.030680 0.013444 \n",

"HasCrCard 0.000599 -0.014025 -0.005458 -0.011721 0.022583 \n",

"IsActiveMember 0.012044 0.001665 0.025651 0.085472 -0.028362 \n",

"EstimatedSalary -0.005988 0.015271 -0.001384 -0.007201 0.007784 \n",

"Exited -0.016571 -0.006248 -0.027094 0.285323 -0.014001 \n",

"\n",

" Balance NumOfProducts HasCrCard IsActiveMember \\\n",

"RowNumber -0.009067 0.007246 0.000599 0.012044 \n",

"CustomerId -0.012419 0.016972 -0.014025 0.001665 \n",

"CreditScore 0.006268 0.012238 -0.005458 0.025651 \n",

"Age 0.028308 -0.030680 -0.011721 0.085472 \n",

"Tenure -0.012254 0.013444 0.022583 -0.028362 \n",

"Balance 1.000000 -0.304180 -0.014858 -0.010084 \n",

"NumOfProducts -0.304180 1.000000 0.003183 0.009612 \n",

"HasCrCard -0.014858 0.003183 1.000000 -0.011866 \n",

"IsActiveMember -0.010084 0.009612 -0.011866 1.000000 \n",

"EstimatedSalary 0.012797 0.014204 -0.009933 -0.011421 \n",

"Exited 0.118533 -0.047820 -0.007138 -0.156128 \n",

"\n",

" EstimatedSalary Exited \n",

"RowNumber -0.005988 -0.016571 \n",

"CustomerId 0.015271 -0.006248 \n",

"CreditScore -0.001384 -0.027094 \n",

"Age -0.007201 0.285323 \n",

"Tenure 0.007784 -0.014001 \n",

"Balance 0.012797 0.118533 \n",

"NumOfProducts 0.014204 -0.047820 \n",

"HasCrCard -0.009933 -0.007138 \n",

"IsActiveMember -0.011421 -0.156128 \n",

"EstimatedSalary 1.000000 0.012097 \n",

"Exited 0.012097 1.000000 "

]

},

"execution\_count": 11,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.corr()"

]

},

{

"cell\_type": "markdown",

"id": "4b1557ff",

"metadata": {

"id": "4b1557ff"

},

"source": [

"#### 3. Simple Linear Regression"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "0d981f49",

"metadata": {

"id": "0d981f49",

"outputId": "05ffd897-d33d-4b09-dd56-b76b7226ded9"

},

"outputs": [

{

"data": {

"text/html": [

"<table class=\"simpletable\">\n",

"<caption>OLS Regression Results</caption>\n",

"<tr>\n",

" <th>Dep. Variable:</th> <td>CustomerId</td> <th> R-squared: </th> <td> 0.000</td> \n",

"</tr>\n",

"<tr>\n",

" <th>Model:</th> <td>OLS</td> <th> Adj. R-squared: </th> <td> 0.000</td> \n",

"</tr>\n",

"<tr>\n",

" <th>Method:</th> <td>Least Squares</td> <th> F-statistic: </th> <td> 1.967</td> \n",

"</tr>\n",

"<tr>\n",

" <th>Date:</th> <td>Sun, 25 Sep 2022</td> <th> Prob (F-statistic):</th> <td> 0.161</td> \n",

"</tr>\n",

"<tr>\n",

" <th>Time:</th> <td>15:55:30</td> <th> Log-Likelihood: </th> <td>-1.2602e+05</td>\n",

"</tr>\n",

"<tr>\n",

" <th>No. Observations:</th> <td> 10000</td> <th> AIC: </th> <td>2.521e+05</td> \n",

"</tr>\n",

"<tr>\n",

" <th>Df Residuals:</th> <td> 9998</td> <th> BIC: </th> <td>2.521e+05</td> \n",

"</tr>\n",

"<tr>\n",

" <th>Df Model:</th> <td> 1</td> <th> </th> <td> </td> \n",

"</tr>\n",

"<tr>\n",

" <th>Covariance Type:</th> <td>nonrobust</td> <th> </th> <td> </td> \n",

"</tr>\n",

"</table>\n",

"<table class=\"simpletable\">\n",

"<tr>\n",

" <td></td> <th>coef</th> <th>std err</th> <th>t</th> <th>P>|t|</th> <th>[0.025</th> <th>0.975]</th> \n",

"</tr>\n",

"<tr>\n",

" <th>const</th> <td> 1.569e+07</td> <td> 1325.512</td> <td> 1.18e+04</td> <td> 0.000</td> <td> 1.57e+07</td> <td> 1.57e+07</td>\n",

"</tr>\n",

"<tr>\n",

" <th>HasCrCard</th> <td>-2213.3059</td> <td> 1578.103</td> <td> -1.403</td> <td> 0.161</td> <td>-5306.705</td> <td> 880.093</td>\n",

"</tr>\n",

"</table>\n",

"<table class=\"simpletable\">\n",

"<tr>\n",

" <th>Omnibus:</th> <td>8394.858</td> <th> Durbin-Watson: </th> <td> 2.019</td> \n",

"</tr>\n",

"<tr>\n",

" <th>Prob(Omnibus):</th> <td> 0.000</td> <th> Jarque-Bera (JB): </th> <td> 596.113</td> \n",

"</tr>\n",

"<tr>\n",

" <th>Skew:</th> <td> 0.001</td> <th> Prob(JB): </th> <td>3.60e-130</td>\n",

"</tr>\n",

"<tr>\n",

" <th>Kurtosis:</th> <td> 1.804</td> <th> Cond. No. </th> <td> 3.45</td> \n",

"</tr>\n",

"</table><br/><br/>Notes:<br/>[1] Standard Errors assume that the covariance matrix of the errors is correctly specified."

],

"text/plain": [

"<class 'statsmodels.iolib.summary.Summary'>\n",

"\"\"\"\n",

" OLS Regression Results \n",

"==============================================================================\n",

"Dep. Variable: CustomerId R-squared: 0.000\n",

"Model: OLS Adj. R-squared: 0.000\n",

"Method: Least Squares F-statistic: 1.967\n",

"Date: Sun, 25 Sep 2022 Prob (F-statistic): 0.161\n",

"Time: 15:55:30 Log-Likelihood: -1.2602e+05\n",

"No. Observations: 10000 AIC: 2.521e+05\n",

"Df Residuals: 9998 BIC: 2.521e+05\n",

"Df Model: 1 \n",

"Covariance Type: nonrobust \n",

"==============================================================================\n",

" coef std err t P>|t| [0.025 0.975]\n",

"------------------------------------------------------------------------------\n",

"const 1.569e+07 1325.512 1.18e+04 0.000 1.57e+07 1.57e+07\n",

"HasCrCard -2213.3059 1578.103 -1.403 0.161 -5306.705 880.093\n",

"==============================================================================\n",

"Omnibus: 8394.858 Durbin-Watson: 2.019\n",

"Prob(Omnibus): 0.000 Jarque-Bera (JB): 596.113\n",

"Skew: 0.001 Prob(JB): 3.60e-130\n",

"Kurtosis: 1.804 Cond. No. 3.45\n",

"==============================================================================\n",

"\n",

"Notes:\n",

"[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.\n",

"\"\"\""

]

},

"execution\_count": 12,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"y = file\_data['CustomerId']\n",

"x = file\_data['HasCrCard']\n",

"x = sm.add\_constant(x)\n",

"model = sm.OLS(y,x).fit()\n",

"model.summary()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e73e8b35",

"metadata": {

"id": "e73e8b35",

"outputId": "95c63a8b-1eae-4d29-a331-521057ed97fc"

},

"outputs": [

{

"data": {

"text/plain": [

"Text(0, 0.5, 'CreditScore')"

]

},

"execution\_count": 13,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"plt.plot(file\_data['RowNumber'].head() ,file\_data['CreditScore'].head(), )\n",

"\n",

"plt.title('Line plot')\n",

"plt.xlabel('RowNumber')\n",

"plt.ylabel('CreditScore')"

]

},

{

"cell\_type": "markdown",

"id": "8bcdbe83",

"metadata": {

"id": "8bcdbe83"

},

"source": [

"### Multi - Variate Analysis"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "549b6449",

"metadata": {

"id": "549b6449",

"outputId": "5222610c-dd07-4d0c-8a3b-d91bca91a604"

},

"outputs": [

{

"data": {

"text/plain": [

"<AxesSubplot:>"

]

},

"execution\_count": 14,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 864x720 with 2 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"f = plt.subplots(figsize=(12,10))\n",

"sns.heatmap(file\_data.head().corr(), cmap=\"YlGnBu\")"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e166dd37",

"metadata": {

"id": "e166dd37",

"outputId": "5adfddcc-3f45-4111-af4d-a4f1ca2a424e"

},

"outputs": [

{

"data": {

"text/plain": [

"<seaborn.matrix.ClusterGrid at 0x1e3cd562e20>"

]

},

"execution\_count": 15,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 720x720 with 4 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"corrmat = file\_data.corr(method='spearman')\n",

"cg = sns.clustermap(corrmat, cmap=\"YlGnBu\", linewidths=0.1);\n",

"plt.setp(cg.ax\_heatmap.yaxis.get\_majorticklabels(), rotation=0)\n",

"cg"

]

},

{

"cell\_type": "markdown",

"id": "ef291891",

"metadata": {

"id": "ef291891"

},

"source": [

"## 4. Perform descriptive statistics on the dataset.\n"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e11cacda",

"metadata": {

"id": "e11cacda",

"outputId": "9fc2f8ce-07c4-43a9-9b30-380041027ed6"

},

"outputs": [

{

"data": {

"text/plain": [

"(10000, 14)"

]

},

"execution\_count": 16,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.shape"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "15bab369",

"metadata": {

"id": "15bab369",

"outputId": "000486e5-9965-424f-8beb-b008060430c7"

},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"<class 'pandas.core.frame.DataFrame'>\n",

"RangeIndex: 10000 entries, 0 to 9999\n",

"Data columns (total 14 columns):\n",

" # Column Non-Null Count Dtype \n",

"--- ------ -------------- ----- \n",

" 0 RowNumber 10000 non-null int64 \n",

" 1 CustomerId 10000 non-null int64 \n",

" 2 Surname 10000 non-null object \n",

" 3 CreditScore 10000 non-null int64 \n",

" 4 Geography 10000 non-null object \n",

" 5 Gender 10000 non-null object \n",

" 6 Age 10000 non-null int64 \n",

" 7 Tenure 10000 non-null int64 \n",

" 8 Balance 10000 non-null float64\n",

" 9 NumOfProducts 10000 non-null int64 \n",

" 10 HasCrCard 10000 non-null int64 \n",

" 11 IsActiveMember 10000 non-null int64 \n",

" 12 EstimatedSalary 10000 non-null float64\n",

" 13 Exited 10000 non-null int64 \n",

"dtypes: float64(2), int64(9), object(3)\n",

"memory usage: 1.1+ MB\n"

]

}

],

"source": [

"file\_data.info()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "dc49fc58",

"metadata": {

"id": "dc49fc58",

"outputId": "80761e89-d465-4f19-8a0d-7eaf8cd0e9f8"

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>RowNumber</th>\n",

" <th>CustomerId</th>\n",

" <th>CreditScore</th>\n",

" <th>Age</th>\n",

" <th>Tenure</th>\n",

" <th>Balance</th>\n",

" <th>NumOfProducts</th>\n",

" <th>HasCrCard</th>\n",

" <th>IsActiveMember</th>\n",

" <th>EstimatedSalary</th>\n",

" <th>Exited</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>count</th>\n",

" <td>10000.00000</td>\n",

" <td>1.000000e+04</td>\n",

" <td>10000.000000</td>\n",

" <td>10000.000000</td>\n",

" <td>10000.000000</td>\n",

" <td>10000.000000</td>\n",

" <td>10000.000000</td>\n",

" <td>10000.00000</td>\n",

" <td>10000.000000</td>\n",

" <td>10000.000000</td>\n",

" <td>10000.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>mean</th>\n",

" <td>5000.50000</td>\n",

" <td>1.569094e+07</td>\n",

" <td>650.528800</td>\n",

" <td>38.921800</td>\n",

" <td>5.012800</td>\n",

" <td>76485.889288</td>\n",

" <td>1.530200</td>\n",

" <td>0.70550</td>\n",

" <td>0.515100</td>\n",

" <td>100090.239881</td>\n",

" <td>0.203700</td>\n",

" </tr>\n",

" <tr>\n",

" <th>std</th>\n",

" <td>2886.89568</td>\n",

" <td>7.193619e+04</td>\n",

" <td>96.653299</td>\n",

" <td>10.487806</td>\n",

" <td>2.892174</td>\n",

" <td>62397.405202</td>\n",

" <td>0.581654</td>\n",

" <td>0.45584</td>\n",

" <td>0.499797</td>\n",

" <td>57510.492818</td>\n",

" <td>0.402769</td>\n",

" </tr>\n",

" <tr>\n",

" <th>min</th>\n",

" <td>1.00000</td>\n",

" <td>1.556570e+07</td>\n",

" <td>350.000000</td>\n",

" <td>18.000000</td>\n",

" <td>0.000000</td>\n",

" <td>0.000000</td>\n",

" <td>1.000000</td>\n",

" <td>0.00000</td>\n",

" <td>0.000000</td>\n",

" <td>11.580000</td>\n",

" <td>0.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>25%</th>\n",

" <td>2500.75000</td>\n",

" <td>1.562853e+07</td>\n",

" <td>584.000000</td>\n",

" <td>32.000000</td>\n",

" <td>3.000000</td>\n",

" <td>0.000000</td>\n",

" <td>1.000000</td>\n",

" <td>0.00000</td>\n",

" <td>0.000000</td>\n",

" <td>51002.110000</td>\n",

" <td>0.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>50%</th>\n",

" <td>5000.50000</td>\n",

" <td>1.569074e+07</td>\n",

" <td>652.000000</td>\n",

" <td>37.000000</td>\n",

" <td>5.000000</td>\n",

" <td>97198.540000</td>\n",

" <td>1.000000</td>\n",

" <td>1.00000</td>\n",

" <td>1.000000</td>\n",

" <td>100193.915000</td>\n",

" <td>0.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>75%</th>\n",

" <td>7500.25000</td>\n",

" <td>1.575323e+07</td>\n",

" <td>718.000000</td>\n",

" <td>44.000000</td>\n",

" <td>7.000000</td>\n",

" <td>127644.240000</td>\n",

" <td>2.000000</td>\n",

" <td>1.00000</td>\n",

" <td>1.000000</td>\n",

" <td>149388.247500</td>\n",

" <td>0.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>max</th>\n",

" <td>10000.00000</td>\n",

" <td>1.581569e+07</td>\n",

" <td>850.000000</td>\n",

" <td>92.000000</td>\n",

" <td>10.000000</td>\n",

" <td>250898.090000</td>\n",

" <td>4.000000</td>\n",

" <td>1.00000</td>\n",

" <td>1.000000</td>\n",

" <td>199992.480000</td>\n",

" <td>1.000000</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" RowNumber CustomerId CreditScore Age Tenure \\\n",

"count 10000.00000 1.000000e+04 10000.000000 10000.000000 10000.000000 \n",

"mean 5000.50000 1.569094e+07 650.528800 38.921800 5.012800 \n",

"std 2886.89568 7.193619e+04 96.653299 10.487806 2.892174 \n",

"min 1.00000 1.556570e+07 350.000000 18.000000 0.000000 \n",

"25% 2500.75000 1.562853e+07 584.000000 32.000000 3.000000 \n",

"50% 5000.50000 1.569074e+07 652.000000 37.000000 5.000000 \n",

"75% 7500.25000 1.575323e+07 718.000000 44.000000 7.000000 \n",

"max 10000.00000 1.581569e+07 850.000000 92.000000 10.000000 \n",

"\n",

" Balance NumOfProducts HasCrCard IsActiveMember \\\n",

"count 10000.000000 10000.000000 10000.00000 10000.000000 \n",

"mean 76485.889288 1.530200 0.70550 0.515100 \n",

"std 62397.405202 0.581654 0.45584 0.499797 \n",

"min 0.000000 1.000000 0.00000 0.000000 \n",

"25% 0.000000 1.000000 0.00000 0.000000 \n",

"50% 97198.540000 1.000000 1.00000 1.000000 \n",

"75% 127644.240000 2.000000 1.00000 1.000000 \n",

"max 250898.090000 4.000000 1.00000 1.000000 \n",

"\n",

" EstimatedSalary Exited \n",

"count 10000.000000 10000.000000 \n",

"mean 100090.239881 0.203700 \n",

"std 57510.492818 0.402769 \n",

"min 11.580000 0.000000 \n",

"25% 51002.110000 0.000000 \n",

"50% 100193.915000 0.000000 \n",

"75% 149388.247500 0.000000 \n",

"max 199992.480000 1.000000 "

]

},

"execution\_count": 18,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.describe()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "7022ef41",

"metadata": {

"id": "7022ef41",

"outputId": "a4764ab0-3d02-4346-8b8e-b9b6de50a1e9"

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>RowNumber</th>\n",

" <th>CustomerId</th>\n",

" <th>Surname</th>\n",

" <th>CreditScore</th>\n",

" <th>Geography</th>\n",

" <th>Gender</th>\n",

" <th>Age</th>\n",

" <th>Tenure</th>\n",

" <th>Balance</th>\n",

" <th>NumOfProducts</th>\n",

" <th>HasCrCard</th>\n",

" <th>IsActiveMember</th>\n",

" <th>EstimatedSalary</th>\n",

" <th>Exited</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1</td>\n",

" <td>15634602</td>\n",

" <td>Hargrave</td>\n",

" <td>619</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>42</td>\n",

" <td>2</td>\n",

" <td>0.00</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>101348.88</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>2</td>\n",

" <td>15647311</td>\n",

" <td>Hill</td>\n",

" <td>608</td>\n",

" <td>Spain</td>\n",

" <td>Female</td>\n",

" <td>41</td>\n",

" <td>1</td>\n",

" <td>83807.86</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>112542.58</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>3</td>\n",

" <td>15619304</td>\n",

" <td>Onio</td>\n",

" <td>502</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>42</td>\n",

" <td>8</td>\n",

" <td>159660.80</td>\n",

" <td>3</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>113931.57</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>4</td>\n",

" <td>15701354</td>\n",

" <td>Boni</td>\n",

" <td>699</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>39</td>\n",

" <td>1</td>\n",

" <td>0.00</td>\n",

" <td>2</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>93826.63</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>5</td>\n",

" <td>15737888</td>\n",

" <td>Mitchell</td>\n",

" <td>850</td>\n",

" <td>Spain</td>\n",

" <td>Female</td>\n",

" <td>43</td>\n",

" <td>2</td>\n",

" <td>125510.82</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>79084.10</td>\n",

" <td>0</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" RowNumber CustomerId Surname CreditScore Geography Gender Age \\\n",

"0 1 15634602 Hargrave 619 France Female 42 \n",

"1 2 15647311 Hill 608 Spain Female 41 \n",

"2 3 15619304 Onio 502 France Female 42 \n",

"3 4 15701354 Boni 699 France Female 39 \n",

"4 5 15737888 Mitchell 850 Spain Female 43 \n",

"\n",

" Tenure Balance NumOfProducts HasCrCard IsActiveMember \\\n",

"0 2 0.00 1 1 1 \n",

"1 1 83807.86 1 0 1 \n",

"2 8 159660.80 3 1 0 \n",

"3 1 0.00 2 0 0 \n",

"4 2 125510.82 1 1 1 \n",

"\n",

" EstimatedSalary Exited \n",

"0 101348.88 1 \n",

"1 112542.58 0 \n",

"2 113931.57 1 \n",

"3 93826.63 0 \n",

"4 79084.10 0 "

]

},

"execution\_count": 19,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.head()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "fceb68f1",

"metadata": {

"id": "fceb68f1",

"outputId": "8a0edb24-dd21-4f18-9790-9d55f6801e6d"

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>RowNumber</th>\n",

" <th>CustomerId</th>\n",

" <th>Surname</th>\n",

" <th>CreditScore</th>\n",

" <th>Geography</th>\n",

" <th>Gender</th>\n",

" <th>Age</th>\n",

" <th>Tenure</th>\n",

" <th>Balance</th>\n",

" <th>NumOfProducts</th>\n",

" <th>HasCrCard</th>\n",

" <th>IsActiveMember</th>\n",

" <th>EstimatedSalary</th>\n",

" <th>Exited</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>9995</th>\n",

" <td>9996</td>\n",

" <td>15606229</td>\n",

" <td>Obijiaku</td>\n",

" <td>771</td>\n",

" <td>France</td>\n",

" <td>Male</td>\n",

" <td>39</td>\n",

" <td>5</td>\n",

" <td>0.00</td>\n",

" <td>2</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>96270.64</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9996</th>\n",

" <td>9997</td>\n",

" <td>15569892</td>\n",

" <td>Johnstone</td>\n",

" <td>516</td>\n",

" <td>France</td>\n",

" <td>Male</td>\n",

" <td>35</td>\n",

" <td>10</td>\n",

" <td>57369.61</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>101699.77</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9997</th>\n",

" <td>9998</td>\n",

" <td>15584532</td>\n",

" <td>Liu</td>\n",

" <td>709</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>36</td>\n",

" <td>7</td>\n",

" <td>0.00</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>42085.58</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9998</th>\n",

" <td>9999</td>\n",

" <td>15682355</td>\n",

" <td>Sabbatini</td>\n",

" <td>772</td>\n",

" <td>Germany</td>\n",

" <td>Male</td>\n",

" <td>42</td>\n",

" <td>3</td>\n",

" <td>75075.31</td>\n",

" <td>2</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>92888.52</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9999</th>\n",

" <td>10000</td>\n",

" <td>15628319</td>\n",

" <td>Walker</td>\n",

" <td>792</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>28</td>\n",

" <td>4</td>\n",

" <td>130142.79</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>38190.78</td>\n",

" <td>0</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" RowNumber CustomerId Surname CreditScore Geography Gender Age \\\n",

"9995 9996 15606229 Obijiaku 771 France Male 39 \n",

"9996 9997 15569892 Johnstone 516 France Male 35 \n",

"9997 9998 15584532 Liu 709 France Female 36 \n",

"9998 9999 15682355 Sabbatini 772 Germany Male 42 \n",

"9999 10000 15628319 Walker 792 France Female 28 \n",

"\n",

" Tenure Balance NumOfProducts HasCrCard IsActiveMember \\\n",

"9995 5 0.00 2 1 0 \n",

"9996 10 57369.61 1 1 1 \n",

"9997 7 0.00 1 0 1 \n",

"9998 3 75075.31 2 1 0 \n",

"9999 4 130142.79 1 1 0 \n",

"\n",

" EstimatedSalary Exited \n",

"9995 96270.64 0 \n",

"9996 101699.77 0 \n",

"9997 42085.58 1 \n",

"9998 92888.52 1 \n",

"9999 38190.78 0 "

]

},

"execution\_count": 20,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.tail()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "38e6f826",

"metadata": {

"id": "38e6f826",

"outputId": "a6b0e55b-d346-4055-d0d7-00955012fd3b"

},

"outputs": [

{

"data": {

"text/plain": [

"RowNumber 5.000500e+03\n",

"CustomerId 1.569094e+07\n",

"CreditScore 6.505288e+02\n",

"Age 3.892180e+01\n",

"Tenure 5.012800e+00\n",

"Balance 7.648589e+04\n",

"NumOfProducts 1.530200e+00\n",

"HasCrCard 7.055000e-01\n",

"IsActiveMember 5.151000e-01\n",

"EstimatedSalary 1.000902e+05\n",

"Exited 2.037000e-01\n",

"dtype: float64"

]

},

"execution\_count": 21,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.mean(numeric\_only=True)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "3aeca667",

"metadata": {

"id": "3aeca667",

"outputId": "a6f22476-7c19-40bd-e7c5-d68e419e650d"

},

"outputs": [

{

"data": {

"text/plain": [

"RowNumber 5.000500e+03\n",

"CustomerId 1.569074e+07\n",

"CreditScore 6.520000e+02\n",

"Age 3.700000e+01\n",

"Tenure 5.000000e+00\n",

"Balance 9.719854e+04\n",

"NumOfProducts 1.000000e+00\n",

"HasCrCard 1.000000e+00\n",

"IsActiveMember 1.000000e+00\n",

"EstimatedSalary 1.001939e+05\n",

"Exited 0.000000e+00\n",

"dtype: float64"

]

},

"execution\_count": 22,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.median(numeric\_only=True)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e6ecaca7",

"metadata": {

"id": "e6ecaca7",

"outputId": "45cca2cd-9f50-4dfc-ef72-87c939e50561"

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>RowNumber</th>\n",

" <th>CustomerId</th>\n",

" <th>Surname</th>\n",

" <th>CreditScore</th>\n",

" <th>Geography</th>\n",

" <th>Gender</th>\n",

" <th>Age</th>\n",

" <th>Tenure</th>\n",

" <th>Balance</th>\n",

" <th>NumOfProducts</th>\n",

" <th>HasCrCard</th>\n",

" <th>IsActiveMember</th>\n",

" <th>EstimatedSalary</th>\n",

" <th>Exited</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1</td>\n",

" <td>15565701</td>\n",

" <td>Smith</td>\n",

" <td>850.0</td>\n",

" <td>France</td>\n",

" <td>Male</td>\n",

" <td>37.0</td>\n",

" <td>2.0</td>\n",

" <td>0.0</td>\n",

" <td>1.0</td>\n",

" <td>1.0</td>\n",

" <td>1.0</td>\n",

" <td>24924.92</td>\n",

" <td>0.0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>2</td>\n",

" <td>15565706</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>3</td>\n",

" <td>15565714</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>4</td>\n",

" <td>15565779</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>5</td>\n",

" <td>15565796</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9995</th>\n",

" <td>9996</td>\n",

" <td>15815628</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9996</th>\n",

" <td>9997</td>\n",

" <td>15815645</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9997</th>\n",

" <td>9998</td>\n",

" <td>15815656</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9998</th>\n",

" <td>9999</td>\n",

" <td>15815660</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9999</th>\n",

" <td>10000</td>\n",

" <td>15815690</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" <td>NaN</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>10000 rows × 14 columns</p>\n",

"</div>"

],

"text/plain": [

" RowNumber CustomerId Surname CreditScore Geography Gender Age \\\n",

"0 1 15565701 Smith 850.0 France Male 37.0 \n",

"1 2 15565706 NaN NaN NaN NaN NaN \n",

"2 3 15565714 NaN NaN NaN NaN NaN \n",

"3 4 15565779 NaN NaN NaN NaN NaN \n",

"4 5 15565796 NaN NaN NaN NaN NaN \n",

"... ... ... ... ... ... ... ... \n",

"9995 9996 15815628 NaN NaN NaN NaN NaN \n",

"9996 9997 15815645 NaN NaN NaN NaN NaN \n",

"9997 9998 15815656 NaN NaN NaN NaN NaN \n",

"9998 9999 15815660 NaN NaN NaN NaN NaN \n",

"9999 10000 15815690 NaN NaN NaN NaN NaN \n",

"\n",

" Tenure Balance NumOfProducts HasCrCard IsActiveMember \\\n",

"0 2.0 0.0 1.0 1.0 1.0 \n",

"1 NaN NaN NaN NaN NaN \n",

"2 NaN NaN NaN NaN NaN \n",

"3 NaN NaN NaN NaN NaN \n",

"4 NaN NaN NaN NaN NaN \n",

"... ... ... ... ... ... \n",

"9995 NaN NaN NaN NaN NaN \n",

"9996 NaN NaN NaN NaN NaN \n",

"9997 NaN NaN NaN NaN NaN \n",

"9998 NaN NaN NaN NaN NaN \n",

"9999 NaN NaN NaN NaN NaN \n",

"\n",

" EstimatedSalary Exited \n",

"0 24924.92 0.0 \n",

"1 NaN NaN \n",

"2 NaN NaN \n",

"3 NaN NaN \n",

"4 NaN NaN \n",

"... ... ... \n",

"9995 NaN NaN \n",

"9996 NaN NaN \n",

"9997 NaN NaN \n",

"9998 NaN NaN \n",

"9999 NaN NaN \n",

"\n",

"[10000 rows x 14 columns]"

]

},

"execution\_count": 23,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.mode()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "0f3aab82",

"metadata": {

"id": "0f3aab82",

"outputId": "5a8f2b87-8e92-41d6-9e99-9dbffe42f570"

},

"outputs": [

{

"data": {

"text/plain": [

"RowNumber 8.334167e+06\n",

"CustomerId 5.174815e+09\n",

"CreditScore 9.341860e+03\n",

"Age 1.099941e+02\n",

"Tenure 8.364673e+00\n",

"Balance 3.893436e+09\n",

"NumOfProducts 3.383218e-01\n",

"HasCrCard 2.077905e-01\n",

"IsActiveMember 2.497970e-01\n",

"EstimatedSalary 3.307457e+09\n",

"Exited 1.622225e-01\n",

"dtype: float64"

]

},

"execution\_count": 24,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.var(numeric\_only=True)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "4b5c7709",

"metadata": {

"id": "4b5c7709",

"outputId": "aa29f6e3-024e-4c0f-8f13-d8cdc955491f"

},

"outputs": [

{

"data": {

"text/plain": [

"RowNumber 2886.895680\n",

"CustomerId 71936.186123\n",

"CreditScore 96.653299\n",

"Age 10.487806\n",

"Tenure 2.892174\n",

"Balance 62397.405202\n",

"NumOfProducts 0.581654\n",

"HasCrCard 0.455840\n",

"IsActiveMember 0.499797\n",

"EstimatedSalary 57510.492818\n",

"Exited 0.402769\n",

"dtype: float64"

]

},

"execution\_count": 25,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.std(numeric\_only=True)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "231c37dc",

"metadata": {

"id": "231c37dc",

"outputId": "183eac70-dfff-4f35-aaee-41168bd99d66"

},

"outputs": [

{

"data": {

"text/plain": [

"RowNumber 0.000000\n",

"CustomerId 0.001149\n",

"CreditScore -0.071607\n",

"Age 1.011320\n",

"Tenure 0.010991\n",

"Balance -0.141109\n",

"NumOfProducts 0.745568\n",

"HasCrCard -0.901812\n",

"IsActiveMember -0.060437\n",

"EstimatedSalary 0.002085\n",

"Exited 1.471611\n",

"dtype: float64"

]

},

"execution\_count": 26,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.skew(numeric\_only=True)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "3d4e6b74",

"metadata": {

"id": "3d4e6b74",

"outputId": "ca423ea6-57df-4971-b763-4ebe2cab90fb"

},

"outputs": [

{

"data": {

"text/plain": [

"RowNumber -1.200000\n",

"CustomerId -1.196113\n",

"CreditScore -0.425726\n",

"Age 1.395347\n",

"Tenure -1.165225\n",

"Balance -1.489412\n",

"NumOfProducts 0.582981\n",

"HasCrCard -1.186973\n",

"IsActiveMember -1.996747\n",

"EstimatedSalary -1.181518\n",

"Exited 0.165671\n",

"dtype: float64"

]

},

"execution\_count": 27,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.kurt(numeric\_only=True)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "f8fd62af",

"metadata": {

"id": "f8fd62af",

"outputId": "579c5b60-4fec-452a-cb1a-572f04daa781"

},

"outputs": [

{

"data": {

"text/plain": [

"0.75 127644.24\n",

"0.25 0.00\n",

"Name: Balance, dtype: float64"

]

},

"execution\_count": 28,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"quantile = file\_data['Balance'].quantile(q=[0.75, 0.25])\n",

"quantile"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "50737276",

"metadata": {

"id": "50737276",

"outputId": "9d3c6181-e33d-42e7-93f6-2159b9ff7034"

},

"outputs": [

{

"data": {

"text/plain": [

"<AxesSubplot:xlabel='Balance'>"

]

},

"execution\_count": 29,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"x = file\_data.Balance\n",

"sns.boxplot(x=x)"

]

},

{

"cell\_type": "markdown",

"id": "325fb611",

"metadata": {

"id": "325fb611"

},

"source": [

"## 5. Handle the Missing values."

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "1cd6afe9",

"metadata": {

"id": "1cd6afe9",

"outputId": "67e8fe2a-f5cc-4ef5-ac9f-4d0fd6a7415c"

},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

" RowNumber CustomerId Surname CreditScore Geography Gender Age \\\n",

"0 False False False False False False False \n",

"1 False False False False False False False \n",

"2 False False False False False False False \n",

"3 False False False False False False False \n",

"4 False False False False False False False \n",

"... ... ... ... ... ... ... ... \n",

"9995 False False False False False False False \n",

"9996 False False False False False False False \n",

"9997 False False False False False False False \n",

"9998 False False False False False False False \n",

"9999 False False False False False False False \n",

"\n",

" Tenure Balance NumOfProducts HasCrCard IsActiveMember \\\n",

"0 False False False False False \n",

"1 False False False False False \n",

"2 False False False False False \n",

"3 False False False False False \n",

"4 False False False False False \n",

"... ... ... ... ... ... \n",

"9995 False False False False False \n",

"9996 False False False False False \n",

"9997 False False False False False \n",

"9998 False False False False False \n",

"9999 False False False False False \n",

"\n",

" EstimatedSalary Exited \n",

"0 False False \n",

"1 False False \n",

"2 False False \n",

"3 False False \n",

"4 False False \n",

"... ... ... \n",

"9995 False False \n",

"9996 False False \n",

"9997 False False \n",

"9998 False False \n",

"9999 False False \n",

"\n",

"[10000 rows x 14 columns]\n"

]

}

],

"source": [

"print(file\_data.isnull())"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "c237717d",

"metadata": {

"id": "c237717d",

"outputId": "ee41d50b-0ba6-401f-ad98-e96732dba7a4"

},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"RowNumber 0\n",

"CustomerId 0\n",

"Surname 0\n",

"CreditScore 0\n",

"Geography 0\n",

"Gender 0\n",

"Age 0\n",

"Tenure 0\n",

"Balance 0\n",

"NumOfProducts 0\n",

"HasCrCard 0\n",

"IsActiveMember 0\n",

"EstimatedSalary 0\n",

"Exited 0\n",

"dtype: int64\n"

]

}

],

"source": [

"print(file\_data.isnull().sum())"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "bced7bf4",

"metadata": {

"id": "bced7bf4",

"outputId": "a2574375-7619-4a6c-fb8d-451e093a603f"

},

"outputs": [

{

"data": {

"text/plain": [

"RowNumber False\n",

"CustomerId False\n",

"Surname False\n",

"CreditScore False\n",

"Geography False\n",

"Gender False\n",

"Age False\n",

"Tenure False\n",

"Balance False\n",

"NumOfProducts False\n",

"HasCrCard False\n",

"IsActiveMember False\n",

"EstimatedSalary False\n",

"Exited False\n",

"dtype: bool"

]

},

"execution\_count": 32,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"file\_data.isna().any()"

]

},

{

"cell\_type": "markdown",

"id": "1a96a3a9",

"metadata": {

"id": "1a96a3a9"

},

"source": [

"## 6. Find the outliers and replace the outliers"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "9571cda2",

"metadata": {

"id": "9571cda2",

"outputId": "d3565090-22b6-4b5e-d6eb-1ccc1edddb20"

},

"outputs": [

{

"data": {

"text/plain": [

"<AxesSubplot:xlabel='Age'>"

]

},

"execution\_count": 33,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"x = sns.boxplot(x=file\_data[\"Age\"])\n",

"x"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "2d86ac50",

"metadata": {

"id": "2d86ac50",

"outputId": "cfcdee9f-389c-4a10-c848-7e4b5d055443"

},

"outputs": [

{

"data": {

"text/plain": [

"<AxesSubplot:xlabel='Age'>"

]

},

"execution\_count": 34,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"x = file\_data.Age\n",

"sns.boxplot(x=x)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "062e459e",

"metadata": {

"id": "062e459e"

},

"outputs": [],

"source": [

"x = np.where(file\_data['Age']>57,39, file\_data['Age'])"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "ef007603",

"metadata": {

"id": "ef007603",

"outputId": "dc45c362-739e-45bc-84b2-8ddab913fd52"

},

"outputs": [

{

"data": {

"text/plain": [

"<AxesSubplot:>"

]

},

"execution\_count": 36,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"sns.boxplot(x=x)"

]

},

{

"cell\_type": "markdown",

"id": "7d80676d",

"metadata": {

"id": "7d80676d"

},

"source": [

"## 7. Check for Categorical columns and perform encoding."

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e4f5696d",

"metadata": {

"id": "e4f5696d",

"outputId": "695fad6b-4e14-4a1a-925f-976c9a51665f"

},

"outputs": [

{

"data": {

"text/plain": [

"['France', 'Spain', 'France', 'France', 'Spain', ..., 'France', 'France', 'France', 'Germany', 'France']\n",

"Length: 10000\n",

"Categories (3, object): ['France', 'Germany', 'Spain']"

]

},

"execution\_count": 37,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"pd.Categorical(file\_data[\"Geography\"])"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "8dc9c53c",

"metadata": {

"scrolled": true,

"id": "8dc9c53c",

"outputId": "bee5e47c-c0d8-4128-eb80-6facb647d4a0"

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>France</th>\n",

" <th>Germany</th>\n",

" <th>Spain</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>5</th>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>6</th>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>7</th>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>8</th>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9</th>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" France Germany Spain\n",

"0 1 0 0\n",

"1 0 0 1\n",

"2 1 0 0\n",

"3 1 0 0\n",

"4 0 0 1\n",

"5 0 0 1\n",

"6 1 0 0\n",

"7 0 1 0\n",

"8 1 0 0\n",

"9 1 0 0"

]

},

"execution\_count": 38,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"# One Hot Encoding\n",

"\n",

"pd.get\_dummies(file\_data[\"Geography\"]).head(10)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "dddabe5d",

"metadata": {

"id": "dddabe5d",

"outputId": "a232bb67-bbd6-410a-855b-89d076f307b2"

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>RowNumber</th>\n",

" <th>CustomerId</th>\n",

" <th>CreditScore</th>\n",

" <th>Age</th>\n",

" <th>Tenure</th>\n",

" <th>Balance</th>\n",

" <th>NumOfProducts</th>\n",

" <th>HasCrCard</th>\n",

" <th>IsActiveMember</th>\n",

" <th>EstimatedSalary</th>\n",

" <th>...</th>\n",

" <th>Surname\_Zubarev</th>\n",

" <th>Surname\_Zubareva</th>\n",

" <th>Surname\_Zuev</th>\n",

" <th>Surname\_Zuyev</th>\n",

" <th>Surname\_Zuyeva</th>\n",

" <th>Geography\_France</th>\n",

" <th>Geography\_Germany</th>\n",

" <th>Geography\_Spain</th>\n",

" <th>Gender\_Female</th>\n",

" <th>Gender\_Male</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1</td>\n",

" <td>15634602</td>\n",

" <td>619</td>\n",

" <td>42</td>\n",

" <td>2</td>\n",

" <td>0.00</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>101348.88</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>2</td>\n",

" <td>15647311</td>\n",

" <td>608</td>\n",

" <td>41</td>\n",

" <td>1</td>\n",

" <td>83807.86</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>112542.58</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>3</td>\n",

" <td>15619304</td>\n",

" <td>502</td>\n",

" <td>42</td>\n",

" <td>8</td>\n",

" <td>159660.80</td>\n",

" <td>3</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>113931.57</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>4</td>\n",

" <td>15701354</td>\n",

" <td>699</td>\n",

" <td>39</td>\n",

" <td>1</td>\n",

" <td>0.00</td>\n",

" <td>2</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>93826.63</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>5</td>\n",

" <td>15737888</td>\n",

" <td>850</td>\n",

" <td>43</td>\n",

" <td>2</td>\n",

" <td>125510.82</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>79084.10</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>5</th>\n",

" <td>6</td>\n",

" <td>15574012</td>\n",

" <td>645</td>\n",

" <td>44</td>\n",

" <td>8</td>\n",

" <td>113755.78</td>\n",

" <td>2</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>149756.71</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>6</th>\n",

" <td>7</td>\n",

" <td>15592531</td>\n",

" <td>822</td>\n",

" <td>50</td>\n",

" <td>7</td>\n",

" <td>0.00</td>\n",

" <td>2</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>10062.80</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>7</th>\n",

" <td>8</td>\n",

" <td>15656148</td>\n",

" <td>376</td>\n",

" <td>29</td>\n",

" <td>4</td>\n",

" <td>115046.74</td>\n",

" <td>4</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>119346.88</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>8</th>\n",

" <td>9</td>\n",

" <td>15792365</td>\n",

" <td>501</td>\n",

" <td>44</td>\n",

" <td>4</td>\n",

" <td>142051.07</td>\n",

" <td>2</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>74940.50</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9</th>\n",

" <td>10</td>\n",

" <td>15592389</td>\n",

" <td>684</td>\n",

" <td>27</td>\n",

" <td>2</td>\n",

" <td>134603.88</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>71725.73</td>\n",

" <td>...</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>10 rows × 2948 columns</p>\n",

"</div>"

],

"text/plain": [

" RowNumber CustomerId CreditScore Age Tenure Balance NumOfProducts \\\n",

"0 1 15634602 619 42 2 0.00 1 \n",

"1 2 15647311 608 41 1 83807.86 1 \n",

"2 3 15619304 502 42 8 159660.80 3 \n",

"3 4 15701354 699 39 1 0.00 2 \n",

"4 5 15737888 850 43 2 125510.82 1 \n",

"5 6 15574012 645 44 8 113755.78 2 \n",

"6 7 15592531 822 50 7 0.00 2 \n",

"7 8 15656148 376 29 4 115046.74 4 \n",

"8 9 15792365 501 44 4 142051.07 2 \n",

"9 10 15592389 684 27 2 134603.88 1 \n",

"\n",

" HasCrCard IsActiveMember EstimatedSalary ... Surname\_Zubarev \\\n",

"0 1 1 101348.88 ... 0 \n",

"1 0 1 112542.58 ... 0 \n",

"2 1 0 113931.57 ... 0 \n",

"3 0 0 93826.63 ... 0 \n",

"4 1 1 79084.10 ... 0 \n",

"5 1 0 149756.71 ... 0 \n",

"6 1 1 10062.80 ... 0 \n",

"7 1 0 119346.88 ... 0 \n",

"8 0 1 74940.50 ... 0 \n",

"9 1 1 71725.73 ... 0 \n",

"\n",

" Surname\_Zubareva Surname\_Zuev Surname\_Zuyev Surname\_Zuyeva \\\n",

"0 0 0 0 0 \n",

"1 0 0 0 0 \n",

"2 0 0 0 0 \n",

"3 0 0 0 0 \n",

"4 0 0 0 0 \n",

"5 0 0 0 0 \n",

"6 0 0 0 0 \n",

"7 0 0 0 0 \n",

"8 0 0 0 0 \n",

"9 0 0 0 0 \n",

"\n",

" Geography\_France Geography\_Germany Geography\_Spain Gender\_Female \\\n",

"0 1 0 0 1 \n",

"1 0 0 1 1 \n",

"2 1 0 0 1 \n",

"3 1 0 0 1 \n",

"4 0 0 1 1 \n",

"5 0 0 1 0 \n",

"6 1 0 0 0 \n",

"7 0 1 0 1 \n",

"8 1 0 0 0 \n",

"9 1 0 0 0 \n",

"\n",

" Gender\_Male \n",

"0 0 \n",

"1 0 \n",

"2 0 \n",

"3 0 \n",

"4 0 \n",

"5 1 \n",

"6 1 \n",

"7 0 \n",

"8 1 \n",

"9 1 \n",

"\n",

"[10 rows x 2948 columns]"

]

},

"execution\_count": 39,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"pd.get\_dummies(file\_data).head(10)"

]

},

{

"cell\_type": "markdown",

"id": "d26d116f",

"metadata": {

"id": "d26d116f"

},

"source": [

"## 8. Split the data into dependent and independent variables."

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "17b84fb5",

"metadata": {

"id": "17b84fb5",

"outputId": "e02b64f2-83fa-48d3-c3b6-5c44f32fc6df"

},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"[[1 15634602 'Hargrave' ... 1 1 101348.88]\n",

" [2 15647311 'Hill' ... 0 1 112542.58]\n",

" [3 15619304 'Onio' ... 1 0 113931.57]\n",

" ...\n",

" [9998 15584532 'Liu' ... 0 1 42085.58]\n",

" [9999 15682355 'Sabbatini' ... 1 0 92888.52]\n",

" [10000 15628319 'Walker' ... 1 0 38190.78]]\n"

]

}

],

"source": [

"# Splitting the Dataset into the Independent\n",

"\n",

"X = file\_data.iloc[:, :-1].values\n",

"print(X)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e3b32cdb",

"metadata": {

"id": "e3b32cdb",

"outputId": "23d54792-279d-49cd-d17b-dffe4caf56e6"

},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"[1 0 1 ... 1 1 0]\n"

]

}

],

"source": [

"# Extracting the Dataset to Get the Dependent\n",

"\n",

"Y = file\_data.iloc[:, -1].values\n",

"print(Y)"

]

},

{

"cell\_type": "markdown",

"id": "99d60f24",

"metadata": {

"id": "99d60f24"

},

"source": [

"## 9. Scale the independent variables"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "1e13083b",

"metadata": {

"id": "1e13083b"

},

"outputs": [],

"source": [

"from sklearn.preprocessing import scale"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e0c45a50",

"metadata": {

"id": "e0c45a50",

"outputId": "19ca154e-2392-44d0-8982-d5bbf1e81cca"

},

"outputs": [

{

"data": {

"text/plain": [

"array([ 0.02188649, 0.21653375, 0.2406869 , ..., -1.00864308,\n",

" -0.12523071, -1.07636976])"

]

},

"execution\_count": 43,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"x = scale(file\_data[\"EstimatedSalary\"])\n",

"x"

]

},

{

"cell\_type": "markdown",

"id": "44b694ca",

"metadata": {

"id": "44b694ca"

},

"source": [

"## 10. Split the data into training and testing"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "b2554367",

"metadata": {

"id": "b2554367"

},

"outputs": [],

"source": [

"from sklearn.model\_selection import train\_test\_split"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "734ea3e9",

"metadata": {

"id": "734ea3e9",

"outputId": "262cc22b-7075-4c06-9dc6-57358ea7c7c6"

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>RowNumber</th>\n",

" <th>CustomerId</th>\n",

" <th>Surname</th>\n",

" <th>CreditScore</th>\n",

" <th>Geography</th>\n",

" <th>Gender</th>\n",

" <th>Age</th>\n",

" <th>Tenure</th>\n",

" <th>Balance</th>\n",

" <th>NumOfProducts</th>\n",

" <th>HasCrCard</th>\n",

" <th>IsActiveMember</th>\n",

" <th>Exited</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1</td>\n",

" <td>15634602</td>\n",

" <td>Hargrave</td>\n",

" <td>619</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>42</td>\n",

" <td>2</td>\n",

" <td>0.00</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>2</td>\n",

" <td>15647311</td>\n",

" <td>Hill</td>\n",

" <td>608</td>\n",

" <td>Spain</td>\n",

" <td>Female</td>\n",

" <td>41</td>\n",

" <td>1</td>\n",

" <td>83807.86</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>3</td>\n",

" <td>15619304</td>\n",

" <td>Onio</td>\n",

" <td>502</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>42</td>\n",

" <td>8</td>\n",

" <td>159660.80</td>\n",

" <td>3</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>4</td>\n",

" <td>15701354</td>\n",

" <td>Boni</td>\n",

" <td>699</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>39</td>\n",

" <td>1</td>\n",

" <td>0.00</td>\n",

" <td>2</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>5</td>\n",

" <td>15737888</td>\n",

" <td>Mitchell</td>\n",

" <td>850</td>\n",

" <td>Spain</td>\n",

" <td>Female</td>\n",

" <td>43</td>\n",

" <td>2</td>\n",

" <td>125510.82</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>...</th>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" <td>...</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9995</th>\n",

" <td>9996</td>\n",

" <td>15606229</td>\n",

" <td>Obijiaku</td>\n",

" <td>771</td>\n",

" <td>France</td>\n",

" <td>Male</td>\n",

" <td>39</td>\n",

" <td>5</td>\n",

" <td>0.00</td>\n",

" <td>2</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9996</th>\n",

" <td>9997</td>\n",

" <td>15569892</td>\n",

" <td>Johnstone</td>\n",

" <td>516</td>\n",

" <td>France</td>\n",

" <td>Male</td>\n",

" <td>35</td>\n",

" <td>10</td>\n",

" <td>57369.61</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9997</th>\n",

" <td>9998</td>\n",

" <td>15584532</td>\n",

" <td>Liu</td>\n",

" <td>709</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>36</td>\n",

" <td>7</td>\n",

" <td>0.00</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9998</th>\n",

" <td>9999</td>\n",

" <td>15682355</td>\n",

" <td>Sabbatini</td>\n",

" <td>772</td>\n",

" <td>Germany</td>\n",

" <td>Male</td>\n",

" <td>42</td>\n",

" <td>3</td>\n",

" <td>75075.31</td>\n",

" <td>2</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>9999</th>\n",

" <td>10000</td>\n",

" <td>15628319</td>\n",

" <td>Walker</td>\n",

" <td>792</td>\n",

" <td>France</td>\n",

" <td>Female</td>\n",

" <td>28</td>\n",

" <td>4</td>\n",

" <td>130142.79</td>\n",

" <td>1</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>10000 rows × 13 columns</p>\n",

"</div>"

],

"text/plain": [

" RowNumber CustomerId Surname CreditScore Geography Gender Age \\\n",

"0 1 15634602 Hargrave 619 France Female 42 \n",

"1 2 15647311 Hill 608 Spain Female 41 \n",

"2 3 15619304 Onio 502 France Female 42 \n",

"3 4 15701354 Boni 699 France Female 39 \n",

"4 5 15737888 Mitchell 850 Spain Female 43 \n",

"... ... ... ... ... ... ... ... \n",

"9995 9996 15606229 Obijiaku 771 France Male 39 \n",

"9996 9997 15569892 Johnstone 516 France Male 35 \n",

"9997 9998 15584532 Liu 709 France Female 36 \n",

"9998 9999 15682355 Sabbatini 772 Germany Male 42 \n",

"9999 10000 15628319 Walker 792 France Female 28 \n",

"\n",

" Tenure Balance NumOfProducts HasCrCard IsActiveMember Exited \n",

"0 2 0.00 1 1 1 1 \n",

"1 1 83807.86 1 0 1 0 \n",

"2 8 159660.80 3 1 0 1 \n",

"3 1 0.00 2 0 0 0 \n",

"4 2 125510.82 1 1 1 0 \n",

"... ... ... ... ... ... ... \n",

"9995 5 0.00 2 1 0 0 \n",

"9996 10 57369.61 1 1 1 0 \n",

"9997 7 0.00 1 0 1 1 \n",

"9998 3 75075.31 2 1 0 1 \n",

"9999 4 130142.79 1 1 0 0 \n",

"\n",

"[10000 rows x 13 columns]"

]

},

"execution\_count": 45,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"x = file\_data.drop(\"EstimatedSalary\", axis=1)\n",

"x"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "07fa8239",

"metadata": {

"id": "07fa8239",

"outputId": "b62c9df3-1e95-49cc-f104-cb09f5f89f33"

},

"outputs": [

{

"data": {

"text/plain": [

"0 101348.88\n",

"1 112542.58\n",

"2 113931.57\n",

"3 93826.63\n",

"4 79084.10\n",

" ... \n",

"9995 96270.64\n",

"9996 101699.77\n",

"9997 42085.58\n",

"9998 92888.52\n",

"9999 38190.78\n",

"Name: EstimatedSalary, Length: 10000, dtype: float64"

]

},

"execution\_count": 46,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"y = file\_data.EstimatedSalary\n",

"y"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "53e2077a",

"metadata": {

"id": "53e2077a"

},

"outputs": [],

"source": [

"x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.2)"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "931e2b49",

"metadata": {

"id": "931e2b49",

"outputId": "9ade3379-dc58-4075-bfc3-aa0d9c69b192"

},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"(8000, 13) (2000, 13)\n"

]

}

],

"source": [

"print(x\_train.shape, x\_test.shape)"

]

}

],

"metadata": {

"kernelspec": {

"display\_name": "Python 3 (ipykernel)",

"language": "python",

"name": "python3"

},

"language\_info": {

"codemirror\_mode": {

"name": "ipython",

"version": 3

},

"file\_extension": ".py",

"mimetype": "text/x-python",

"name": "python",

"nbconvert\_exporter": "python",

"pygments\_lexer": "ipython3",

"version": "3.9.12"

},

"colab": {

"provenance": []

}

},

"nbformat": 4,

"nbformat\_minor": 5

}