{

"nbformat": 4,

"nbformat\_minor": 0,

"metadata": {

"colab": {

"provenance": []

},

"kernelspec": {

"name": "python3",

"display\_name": "Python 3"

},

"language\_info": {

"name": "python"

}

},

"cells": [

{

"cell\_type": "markdown",

"source": [

"\*\*JAYASRI\*\*"

],

"metadata": {

"id": "Bt7j\_1S9PCmU"

}

},

{

"cell\_type": "markdown",

"source": [

"\*\*1. Downloading Dataset:Chrun\_Modelling\*\*"

],

"metadata": {

"id": "gGvrdK\_125yd"

}

},

{

"cell\_type": "markdown",

"source": [

"\*\*2. Load The Dataset\*\*"

],

"metadata": {

"id": "3OwB3\_kB3jn1"

}

},

{

"cell\_type": "code",

"execution\_count": null,

"metadata": {

"id": "kZLcc9eVKhdb"

},

"outputs": [],

"source": [

"import numpy as np\n",

"import pandas as pd\n",

"import seaborn as sns\n",

"import matplotlib.pyplot as plt"

]

},

{

"cell\_type": "code",

"source": [

"df = pd.read\_csv('/content/drive/MyDrive/Churn\_Modelling.csv')\n",

"df.head()\n"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 270

},

"id": "N-veA3-EK1DR",

"outputId": "68175099-1a5b-4208-a7da-b6da213d28fe"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "execute\_result",

"data": {

"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 "

],

"text/html": [

"\n",

" <div id=\"df-70483a8c-3862-4b8f-aff9-05e87bef530a\">\n",

" <div class=\"colab-df-container\">\n",

" <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>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-70483a8c-3862-4b8f-aff9-05e87bef530a')\"\n",

" title=\"Convert this dataframe to an interactive table.\"\n",

" style=\"display:none;\">\n",

" \n",

" <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0 24 24\"\n",

" width=\"24px\">\n",

" <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",

" <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-2.06.94zm10 10l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-1.37c-.4-.4-.92-.59-1.43-.59-.52 0-1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0 2.83L4 21.41c.39.39.9.59 1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-2.86zM5.41 20L4 18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",

" </svg>\n",

" </button>\n",

" \n",

" <style>\n",

" .colab-df-container {\n",

" display:flex;\n",

" flex-wrap:wrap;\n",

" gap: 12px;\n",

" }\n",

"\n",

" .colab-df-convert {\n",

" background-color: #E8F0FE;\n",

" border: none;\n",

" border-radius: 50%;\n",

" cursor: pointer;\n",

" display: none;\n",

" fill: #1967D2;\n",

" height: 32px;\n",

" padding: 0 0 0 0;\n",

" width: 32px;\n",

" }\n",

"\n",

" .colab-df-convert:hover {\n",

" background-color: #E2EBFA;\n",

" box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px rgba(60, 64, 67, 0.15);\n",

" fill: #174EA6;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert {\n",

" background-color: #3B4455;\n",

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert:hover {\n",

" background-color: #434B5C;\n",

" box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",

" filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",

" fill: #FFFFFF;\n",

" }\n",

" </style>\n",

"\n",

" <script>\n",

" const buttonEl =\n",

" document.querySelector('#df-70483a8c-3862-4b8f-aff9-05e87bef530a button.colab-df-convert');\n",

" buttonEl.style.display =\n",

" google.colab.kernel.accessAllowed ? 'block' : 'none';\n",

"\n",

" async function convertToInteractive(key) {\n",

" const element = document.querySelector('#df-70483a8c-3862-4b8f-aff9-05e87bef530a');\n",

" const dataTable =\n",

" await google.colab.kernel.invokeFunction('convertToInteractive',\n",

" [key], {});\n",

" if (!dataTable) return;\n",

"\n",

" const docLinkHtml = 'Like what you see? Visit the ' +\n",

" '<a target=\"\_blank\" href=https://colab.research.google.com/notebooks/data\_table.ipynb>data table notebook</a>'\n",

" + ' to learn more about interactive tables.';\n",

" element.innerHTML = '';\n",

" dataTable['output\_type'] = 'display\_data';\n",

" await google.colab.output.renderOutput(dataTable, element);\n",

" const docLink = document.createElement('div');\n",

" docLink.innerHTML = docLinkHtml;\n",

" element.appendChild(docLink);\n",

" }\n",

" </script>\n",

" </div>\n",

" </div>\n",

" "

]

},

"metadata": {},

"execution\_count": 4

}

]

},

{

"cell\_type": "code",

"source": [

"df = df.drop(columns=['RowNumber', 'CustomerId', 'Surname'])\n",

"df.head()"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 270

},

"id": "zSmwYH\_0LMTy",

"outputId": "79a905e3-4f6e-46c3-8d3b-16b092187108"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "execute\_result",

"data": {

"text/plain": [

" CreditScore Geography Gender Age Tenure Balance NumOfProducts \\\n",

"0 619 France Female 42 2 0.00 1 \n",

"1 608 Spain Female 41 1 83807.86 1 \n",

"2 502 France Female 42 8 159660.80 3 \n",

"3 699 France Female 39 1 0.00 2 \n",

"4 850 Spain Female 43 2 125510.82 1 \n",

"\n",

" HasCrCard IsActiveMember EstimatedSalary Exited \n",

"0 1 1 101348.88 1 \n",

"1 0 1 112542.58 0 \n",

"2 1 0 113931.57 1 \n",

"3 0 0 93826.63 0 \n",

"4 1 1 79084.10 0 "

],

"text/html": [

"\n",

" <div id=\"df-6287b8bb-9432-428d-902b-4db9763fb19e\">\n",

" <div class=\"colab-df-container\">\n",

" <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>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>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>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>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>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>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>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-6287b8bb-9432-428d-902b-4db9763fb19e')\"\n",

" title=\"Convert this dataframe to an interactive table.\"\n",

" style=\"display:none;\">\n",

" \n",

" <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0 24 24\"\n",

" width=\"24px\">\n",

" <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",

" <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-2.06.94zm10 10l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-1.37c-.4-.4-.92-.59-1.43-.59-.52 0-1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0 2.83L4 21.41c.39.39.9.59 1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-2.86zM5.41 20L4 18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",

" </svg>\n",

" </button>\n",

" \n",

" <style>\n",

" .colab-df-container {\n",

" display:flex;\n",

" flex-wrap:wrap;\n",

" gap: 12px;\n",

" }\n",

"\n",

" .colab-df-convert {\n",

" background-color: #E8F0FE;\n",

" border: none;\n",

" border-radius: 50%;\n",

" cursor: pointer;\n",

" display: none;\n",

" fill: #1967D2;\n",

" height: 32px;\n",

" padding: 0 0 0 0;\n",

" width: 32px;\n",

" }\n",

"\n",

" .colab-df-convert:hover {\n",

" background-color: #E2EBFA;\n",

" box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px rgba(60, 64, 67, 0.15);\n",

" fill: #174EA6;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert {\n",

" background-color: #3B4455;\n",

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert:hover {\n",

" background-color: #434B5C;\n",

" box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",

" filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",

" fill: #FFFFFF;\n",

" }\n",

" </style>\n",

"\n",

" <script>\n",

" const buttonEl =\n",

" document.querySelector('#df-6287b8bb-9432-428d-902b-4db9763fb19e button.colab-df-convert');\n",

" buttonEl.style.display =\n",

" google.colab.kernel.accessAllowed ? 'block' : 'none';\n",

"\n",

" async function convertToInteractive(key) {\n",

" const element = document.querySelector('#df-6287b8bb-9432-428d-902b-4db9763fb19e');\n",

" const dataTable =\n",

" await google.colab.kernel.invokeFunction('convertToInteractive',\n",

" [key], {});\n",

" if (!dataTable) return;\n",

"\n",

" const docLinkHtml = 'Like what you see? Visit the ' +\n",

" '<a target=\"\_blank\" href=https://colab.research.google.com/notebooks/data\_table.ipynb>data table notebook</a>'\n",

" + ' to learn more about interactive tables.';\n",

" element.innerHTML = '';\n",

" dataTable['output\_type'] = 'display\_data';\n",

" await google.colab.output.renderOutput(dataTable, element);\n",

" const docLink = document.createElement('div');\n",

" docLink.innerHTML = docLinkHtml;\n",

" element.appendChild(docLink);\n",

" }\n",

" </script>\n",

" </div>\n",

" </div>\n",

" "

]

},

"metadata": {},

"execution\_count": 4

}

]

},

{

"cell\_type": "code",

"source": [

"df['IsActiveMember'] = df['IsActiveMember'].astype('category')\n",

"df['Exited'] = df['Exited'].astype('category')\n",

"df['HasCrCard'] = df['HasCrCard'].astype('category')"

],

"metadata": {

"id": "bzVo5D\_dLNoR"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"\*\*3. Perform\*\*"

],

"metadata": {

"id": "ENHluAwfN27V"

}

},

{

"cell\_type": "markdown",

"source": [

"\*\*Univariate Analysis\*\*"

],

"metadata": {

"id": "eV1xtQxdOJZK"

}

},

{

"cell\_type": "markdown",

"source": [

"\*\*Bi - Variate Analysis\*\*"

],

"metadata": {

"id": "ZNLbiUwPOEX6"

}

},

{

"cell\_type": "markdown",

"source": [

"\*\*Multi - Variate Analysis\*\*"

],

"metadata": {

"id": "o\_Pgf0E\_OMnX"

}

},

{

"cell\_type": "code",

"source": [

"sns.kdeplot(x='CreditScore', data = df , hue = 'Exited')\n",

"plt.show()"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 279

},

"id": "ML\_pXsZtLa2d",

"outputId": "06e61a54-7fdc-4065-c6cd-fee8147ec787"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "code",

"source": [

"density = df['Exited'].value\_counts(normalize=True).reset\_index()\n",

"sns.barplot(data=density, x='index', y='Exited', );\n",

"density"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 374

},

"id": "l8WxH13lLpDa",

"outputId": "4162f1dd-4d1b-473b-8e86-876c2a3434f1"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "execute\_result",

"data": {

"text/plain": [

" index Exited\n",

"0 0 0.7963\n",

"1 1 0.2037"

],

"text/html": [

"\n",

" <div id=\"df-fb705e3a-c14b-489a-8aa8-1217fbc731ca\">\n",

" <div class=\"colab-df-container\">\n",

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

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

" <td>0.7963</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>0.2037</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-fb705e3a-c14b-489a-8aa8-1217fbc731ca')\"\n",

" title=\"Convert this dataframe to an interactive table.\"\n",

" style=\"display:none;\">\n",

" \n",

" <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0 24 24\"\n",

" width=\"24px\">\n",

" <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",

" <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-2.06.94zm10 10l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-1.37c-.4-.4-.92-.59-1.43-.59-.52 0-1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0 2.83L4 21.41c.39.39.9.59 1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-2.86zM5.41 20L4 18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",

" </svg>\n",

" </button>\n",

" \n",

" <style>\n",

" .colab-df-container {\n",

" display:flex;\n",

" flex-wrap:wrap;\n",

" gap: 12px;\n",

" }\n",

"\n",

" .colab-df-convert {\n",

" background-color: #E8F0FE;\n",

" border: none;\n",

" border-radius: 50%;\n",

" cursor: pointer;\n",

" display: none;\n",

" fill: #1967D2;\n",

" height: 32px;\n",

" padding: 0 0 0 0;\n",

" width: 32px;\n",

" }\n",

"\n",

" .colab-df-convert:hover {\n",

" background-color: #E2EBFA;\n",

" box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px rgba(60, 64, 67, 0.15);\n",

" fill: #174EA6;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert {\n",

" background-color: #3B4455;\n",

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert:hover {\n",

" background-color: #434B5C;\n",

" box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",

" filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",

" fill: #FFFFFF;\n",

" }\n",

" </style>\n",

"\n",

" <script>\n",

" const buttonEl =\n",

" document.querySelector('#df-fb705e3a-c14b-489a-8aa8-1217fbc731ca button.colab-df-convert');\n",

" buttonEl.style.display =\n",

" google.colab.kernel.accessAllowed ? 'block' : 'none';\n",

"\n",

" async function convertToInteractive(key) {\n",

" const element = document.querySelector('#df-fb705e3a-c14b-489a-8aa8-1217fbc731ca');\n",

" const dataTable =\n",

" await google.colab.kernel.invokeFunction('convertToInteractive',\n",

" [key], {});\n",

" if (!dataTable) return;\n",

"\n",

" const docLinkHtml = 'Like what you see? Visit the ' +\n",

" '<a target=\"\_blank\" href=https://colab.research.google.com/notebooks/data\_table.ipynb>data table notebook</a>'\n",

" + ' to learn more about interactive tables.';\n",

" element.innerHTML = '';\n",

" dataTable['output\_type'] = 'display\_data';\n",

" await google.colab.output.renderOutput(dataTable, element);\n",

" const docLink = document.createElement('div');\n",

" docLink.innerHTML = docLinkHtml;\n",

" element.appendChild(docLink);\n",

" }\n",

" </script>\n",

" </div>\n",

" </div>\n",

" "

]

},

"metadata": {},

"execution\_count": 9

},

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 432x288 with 1 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "code",

"source": [

"categorical = df.drop(columns=['CreditScore', 'Age', 'Tenure', 'Balance', 'EstimatedSalary'])\n",

"rows = int(np.ceil(categorical.shape[1] / 2)) - 1\n",

"fig, axes = plt.subplots(nrows=rows, ncols=2, figsize=(10,6))\n",

"axes = axes.flatten()\n",

"\n",

"for row in range(rows):\n",

" cols = min(2, categorical.shape[1] - row\*2)\n",

" for col in range(cols):\n",

" col\_name = categorical.columns[2 \* row + col]\n",

" ax = axes[row\*2 + col] \n",

"\n",

" sns.countplot(data=categorical, x=col\_name, hue=\"Exited\", ax=ax);\n",

" \n",

"plt.tight\_layout()"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 441

},

"id": "Jkx2G5gwLsCi",

"outputId": "f231d79f-e445-4a59-c6fb-a3c84d7595ed"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 720x432 with 4 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "markdown",

"source": [

"\*\*4. Descriptive statistics bold text\*\*"

],

"metadata": {

"id": "cZOrCZ1lOTB6"

}

},

{

"cell\_type": "code",

"source": [

"df.info()"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "jZc8O6cSL3Mq",

"outputId": "bae5c2ea-5545-4ace-9423-e642b0cd0d19"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

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

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

"Data columns (total 11 columns):\n",

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

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

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

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

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

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

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

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

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

" 7 HasCrCard 10000 non-null category\n",

" 8 IsActiveMember 10000 non-null category\n",

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

" 10 Exited 10000 non-null category\n",

"dtypes: category(3), float64(2), int64(4), object(2)\n",

"memory usage: 654.8+ KB\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"df.describe()"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 300

},

"id": "nbwS2Qe5L5\_h",

"outputId": "b8fd61eb-4cfc-4269-cd6d-37e0bc7f9c4c"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "execute\_result",

"data": {

"text/plain": [

" CreditScore Age Tenure Balance NumOfProducts \\\n",

"count 10000.000000 10000.000000 10000.000000 10000.000000 10000.000000 \n",

"mean 650.528800 38.921800 5.012800 76485.889288 1.530200 \n",

"std 96.653299 10.487806 2.892174 62397.405202 0.581654 \n",

"min 350.000000 18.000000 0.000000 0.000000 1.000000 \n",

"25% 584.000000 32.000000 3.000000 0.000000 1.000000 \n",

"50% 652.000000 37.000000 5.000000 97198.540000 1.000000 \n",

"75% 718.000000 44.000000 7.000000 127644.240000 2.000000 \n",

"max 850.000000 92.000000 10.000000 250898.090000 4.000000 \n",

"\n",

" EstimatedSalary \n",

"count 10000.000000 \n",

"mean 100090.239881 \n",

"std 57510.492818 \n",

"min 11.580000 \n",

"25% 51002.110000 \n",

"50% 100193.915000 \n",

"75% 149388.247500 \n",

"max 199992.480000 "

],

"text/html": [

"\n",

" <div id=\"df-60c550ef-2ba1-4bcb-9306-0711e436e5f8\">\n",

" <div class=\"colab-df-container\">\n",

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

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

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

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

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

" </tr>\n",

" <tr>\n",

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

" </tr>\n",

" <tr>\n",

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

" </tr>\n",

" <tr>\n",

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

" </tr>\n",

" <tr>\n",

" <th>25%</th>\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>51002.110000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>50%</th>\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>100193.915000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>75%</th>\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>149388.247500</td>\n",

" </tr>\n",

" <tr>\n",

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-60c550ef-2ba1-4bcb-9306-0711e436e5f8')\"\n",

" title=\"Convert this dataframe to an interactive table.\"\n",

" style=\"display:none;\">\n",

" \n",

" <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0 24 24\"\n",

" width=\"24px\">\n",

" <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",

" <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-2.06.94zm10 10l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-1.37c-.4-.4-.92-.59-1.43-.59-.52 0-1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0 2.83L4 21.41c.39.39.9.59 1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-2.86zM5.41 20L4 18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",

" </svg>\n",

" </button>\n",

" \n",

" <style>\n",

" .colab-df-container {\n",

" display:flex;\n",

" flex-wrap:wrap;\n",

" gap: 12px;\n",

" }\n",

"\n",

" .colab-df-convert {\n",

" background-color: #E8F0FE;\n",

" border: none;\n",

" border-radius: 50%;\n",

" cursor: pointer;\n",

" display: none;\n",

" fill: #1967D2;\n",

" height: 32px;\n",

" padding: 0 0 0 0;\n",

" width: 32px;\n",

" }\n",

"\n",

" .colab-df-convert:hover {\n",

" background-color: #E2EBFA;\n",

" box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px rgba(60, 64, 67, 0.15);\n",

" fill: #174EA6;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert {\n",

" background-color: #3B4455;\n",

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert:hover {\n",

" background-color: #434B5C;\n",

" box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",

" filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",

" fill: #FFFFFF;\n",

" }\n",

" </style>\n",

"\n",

" <script>\n",

" const buttonEl =\n",

" document.querySelector('#df-60c550ef-2ba1-4bcb-9306-0711e436e5f8 button.colab-df-convert');\n",

" buttonEl.style.display =\n",

" google.colab.kernel.accessAllowed ? 'block' : 'none';\n",

"\n",

" async function convertToInteractive(key) {\n",

" const element = document.querySelector('#df-60c550ef-2ba1-4bcb-9306-0711e436e5f8');\n",

" const dataTable =\n",

" await google.colab.kernel.invokeFunction('convertToInteractive',\n",

" [key], {});\n",

" if (!dataTable) return;\n",

"\n",

" const docLinkHtml = 'Like what you see? Visit the ' +\n",

" '<a target=\"\_blank\" href=https://colab.research.google.com/notebooks/data\_table.ipynb>data table notebook</a>'\n",

" + ' to learn more about interactive tables.';\n",

" element.innerHTML = '';\n",

" dataTable['output\_type'] = 'display\_data';\n",

" await google.colab.output.renderOutput(dataTable, element);\n",

" const docLink = document.createElement('div');\n",

" docLink.innerHTML = docLinkHtml;\n",

" element.appendChild(docLink);\n",

" }\n",

" </script>\n",

" </div>\n",

" </div>\n",

" "

]

},

"metadata": {},

"execution\_count": 13

}

]

},

{

"cell\_type": "markdown",

"source": [

"\*\*5. Handle Missing Values\*\*"

],

"metadata": {

"id": "AbIYwD5YOYPD"

}

},

{

"cell\_type": "code",

"source": [

"df.isna().sum()"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "oI8JJHOHMMaS",

"outputId": "50465b16-f0ed-4132-aac6-52cca5709d41"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "execute\_result",

"data": {

"text/plain": [

"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"

]

},

"metadata": {},

"execution\_count": 15

}

]

},

{

"cell\_type": "markdown",

"source": [

"\*\*In this dataset there is no missing values\*\*"

],

"metadata": {

"id": "t5iIjnqcOdf2"

}

},

{

"cell\_type": "markdown",

"source": [

"\*\*6. Find the outliers and replace the outliers\*\*"

],

"metadata": {

"id": "BYFHAxgbOiki"

}

},

{

"cell\_type": "markdown",

"source": [

"\*\*Finding Outliers\*\*"

],

"metadata": {

"id": "j9\_usMGXOmOK"

}

},

{

"cell\_type": "code",

"source": [

"def box\_scatter(data, x, y): \n",

" fig, (ax1, ax2) = plt.subplots(nrows=2, ncols=1, figsize=(16,6))\n",

" sns.boxplot(data=data, x=x, ax=ax1)\n",

" sns.scatterplot(data=data, x=x,y=y,ax=ax2)"

],

"metadata": {

"id": "gz3bT7ocMbfh"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"box\_scatter(df,'CreditScore','Exited');\n",

"plt.tight\_layout()\n",

"print(f\"# of Bivariate Outliers: {len(df.loc[df['CreditScore'] < 400])}\")"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 384

},

"id": "sDMI8HQYMd29",

"outputId": "c3cd9ac2-9eb9-42fd-8cd3-473e4b5a9483"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"# of Bivariate Outliers: 19\n"

]

},

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 1152x432 with 2 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "code",

"source": [

"box\_scatter(df,'Age','Exited');\n",

"plt.tight\_layout()\n",

"print(f\"# of Bivariate Outliers: {len(df.loc[df['Age'] > 87])}\")"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 384

},

"id": "DFXYkLRCMgBB",

"outputId": "02677042-73bf-417d-bc13-0437b3dfe77d"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"# of Bivariate Outliers: 3\n"

]

},

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 1152x432 with 2 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "code",

"source": [

"box\_scatter(df,'Balance','Exited');\n",

"plt.tight\_layout()\n",

"print(f\"# of Bivariate Outliers: {len(df.loc[df['Balance'] > 220000])}\")"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 384

},

"id": "FSpMgR9YMjZy",

"outputId": "79620e9a-c242-4058-ee21-2e651c7c0801"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"# of Bivariate Outliers: 4\n"

]

},

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 1152x432 with 2 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "code",

"source": [

"box\_scatter(df,'EstimatedSalary','Exited');\n",

"plt.tight\_layout()"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 367

},

"id": "wjGk6ljJMmqB",

"outputId": "f7170c54-1c34-473c-fad9-7e6cc3ef1f6a"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 1152x432 with 2 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "markdown",

"source": [

"\*\*Removing The Outliers\*\*"

],

"metadata": {

"id": "UasbRhrnOsT7"

}

},

{

"cell\_type": "code",

"source": [

"for i in df:\n",

" if df[i].dtype=='int64' or df[i].dtypes=='float64':\n",

" q1=df[i].quantile(0.25)\n",

" q3=df[i].quantile(0.75)\n",

" iqr=q3-q1\n",

" upper=q3+1.5\*iqr\n",

" lower=q1-1.5\*iqr\n",

" df[i]=np.where(df[i] >upper, upper, df[i])\n",

" df[i]=np.where(df[i] <lower, lower, df[i])"

],

"metadata": {

"id": "nzomQbi\_Mtts"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"box\_scatter(df,'CreditScore','Exited');\n",

"plt.tight\_layout()\n",

"print(f\"# of Bivariate Outliers: {len(df.loc[df['CreditScore'] < 400])}\")"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 384

},

"id": "NDRvCwNuMu1p",

"outputId": "fe025e9f-b86c-4d80-ea6f-d989ab80647b"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"# of Bivariate Outliers: 19\n"

]

},

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 1152x432 with 2 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "code",

"source": [

"box\_scatter(df,'Age','Exited');\n",

"plt.tight\_layout()\n",

"print(f\"# of Bivariate Outliers: {len(df.loc[df['Age'] > 87])}\")"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 384

},

"id": "YzJfK2UaMxqB",

"outputId": "735c850e-6cfe-48f6-86e2-5166c50ea9d7"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"# of Bivariate Outliers: 0\n"

]

},

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 1152x432 with 2 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "code",

"source": [

"box\_scatter(df,'Balance','Exited');\n",

"plt.tight\_layout()\n",

"print(f\"# of Bivariate Outliers: {len(df.loc[df['Balance'] > 220000])}\")"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 384

},

"id": "7VzWjrE9M0jS",

"outputId": "8f5339b3-8b26-4458-bdc6-d2eef19962ae"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"# of Bivariate Outliers: 4\n"

]

},

{

"output\_type": "display\_data",

"data": {

"text/plain": [

"<Figure size 1152x432 with 2 Axes>"

],

"image/png": "\n"

},

"metadata": {

"needs\_background": "light"

}

}

]

},

{

"cell\_type": "markdown",

"source": [

"\*\*7. Check for Categorical columns and perform encoding.\*\*"

],

"metadata": {

"id": "ilY5uQuYOzpb"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.preprocessing import LabelEncoder\n",

"encoder=LabelEncoder()\n",

"for i in df:\n",

" if df[i].dtype=='object' or df[i].dtype=='category':\n",

" df[i]=encoder.fit\_transform(df[i])"

],

"metadata": {

"id": "ABQFdeCIM7F5"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"\*\*8. Split the data into dependent and independent variables.\*\*"

],

"metadata": {

"id": "8SPWpwyQO6Qa"

}

},

{

"cell\_type": "code",

"source": [

"x=df.iloc[:,:-1]\n",

"x.head()"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/",

"height": 206

},

"id": "30sqAt7uNBHa",

"outputId": "e11b2c82-1526-4d47-d8f9-713c17a4b2a8"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "execute\_result",

"data": {

"text/plain": [

" CreditScore Geography Gender Age Tenure Balance NumOfProducts \\\n",

"0 619.0 0 0 42.0 2.0 0.00 1.0 \n",

"1 608.0 2 0 41.0 1.0 83807.86 1.0 \n",

"2 502.0 0 0 42.0 8.0 159660.80 3.0 \n",

"3 699.0 0 0 39.0 1.0 0.00 2.0 \n",

"4 850.0 2 0 43.0 2.0 125510.82 1.0 \n",

"\n",

" HasCrCard IsActiveMember EstimatedSalary \n",

"0 1 1 101348.88 \n",

"1 0 1 112542.58 \n",

"2 1 0 113931.57 \n",

"3 0 0 93826.63 \n",

"4 1 1 79084.10 "

],

"text/html": [

"\n",

" <div id=\"df-543e9849-9122-4e9a-9e9b-b31aea24093b\">\n",

" <div class=\"colab-df-container\">\n",

" <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>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",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

" <td>619.0</td>\n",

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

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

" <td>42.0</td>\n",

" <td>2.0</td>\n",

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

" <td>1.0</td>\n",

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>608.0</td>\n",

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

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

" <td>41.0</td>\n",

" <td>1.0</td>\n",

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

" <td>1.0</td>\n",

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>502.0</td>\n",

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

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

" <td>42.0</td>\n",

" <td>8.0</td>\n",

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

" <td>3.0</td>\n",

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>699.0</td>\n",

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

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

" <td>39.0</td>\n",

" <td>1.0</td>\n",

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

" <td>2.0</td>\n",

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>850.0</td>\n",

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

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

" <td>43.0</td>\n",

" <td>2.0</td>\n",

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

" <td>1.0</td>\n",

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

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

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-543e9849-9122-4e9a-9e9b-b31aea24093b')\"\n",

" title=\"Convert this dataframe to an interactive table.\"\n",

" style=\"display:none;\">\n",

" \n",

" <svg xmlns=\"http://www.w3.org/2000/svg\" height=\"24px\"viewBox=\"0 0 24 24\"\n",

" width=\"24px\">\n",

" <path d=\"M0 0h24v24H0V0z\" fill=\"none\"/>\n",

" <path d=\"M18.56 5.44l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94zm-11 1L8.5 8.5l.94-2.06 2.06-.94-2.06-.94L8.5 2.5l-.94 2.06-2.06.94zm10 10l.94 2.06.94-2.06 2.06-.94-2.06-.94-.94-2.06-.94 2.06-2.06.94z\"/><path d=\"M17.41 7.96l-1.37-1.37c-.4-.4-.92-.59-1.43-.59-.52 0-1.04.2-1.43.59L10.3 9.45l-7.72 7.72c-.78.78-.78 2.05 0 2.83L4 21.41c.39.39.9.59 1.41.59.51 0 1.02-.2 1.41-.59l7.78-7.78 2.81-2.81c.8-.78.8-2.07 0-2.86zM5.41 20L4 18.59l7.72-7.72 1.47 1.35L5.41 20z\"/>\n",

" </svg>\n",

" </button>\n",

" \n",

" <style>\n",

" .colab-df-container {\n",

" display:flex;\n",

" flex-wrap:wrap;\n",

" gap: 12px;\n",

" }\n",

"\n",

" .colab-df-convert {\n",

" background-color: #E8F0FE;\n",

" border: none;\n",

" border-radius: 50%;\n",

" cursor: pointer;\n",

" display: none;\n",

" fill: #1967D2;\n",

" height: 32px;\n",

" padding: 0 0 0 0;\n",

" width: 32px;\n",

" }\n",

"\n",

" .colab-df-convert:hover {\n",

" background-color: #E2EBFA;\n",

" box-shadow: 0px 1px 2px rgba(60, 64, 67, 0.3), 0px 1px 3px 1px rgba(60, 64, 67, 0.15);\n",

" fill: #174EA6;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert {\n",

" background-color: #3B4455;\n",

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-convert:hover {\n",

" background-color: #434B5C;\n",

" box-shadow: 0px 1px 3px 1px rgba(0, 0, 0, 0.15);\n",

" filter: drop-shadow(0px 1px 2px rgba(0, 0, 0, 0.3));\n",

" fill: #FFFFFF;\n",

" }\n",

" </style>\n",

"\n",

" <script>\n",

" const buttonEl =\n",

" document.querySelector('#df-543e9849-9122-4e9a-9e9b-b31aea24093b button.colab-df-convert');\n",

" buttonEl.style.display =\n",

" google.colab.kernel.accessAllowed ? 'block' : 'none';\n",

"\n",

" async function convertToInteractive(key) {\n",

" const element = document.querySelector('#df-543e9849-9122-4e9a-9e9b-b31aea24093b');\n",

" const dataTable =\n",

" await google.colab.kernel.invokeFunction('convertToInteractive',\n",

" [key], {});\n",

" if (!dataTable) return;\n",

"\n",

" const docLinkHtml = 'Like what you see? Visit the ' +\n",

" '<a target=\"\_blank\" href=https://colab.research.google.com/notebooks/data\_table.ipynb>data table notebook</a>'\n",

" + ' to learn more about interactive tables.';\n",

" element.innerHTML = '';\n",

" dataTable['output\_type'] = 'display\_data';\n",

" await google.colab.output.renderOutput(dataTable, element);\n",

" const docLink = document.createElement('div');\n",

" docLink.innerHTML = docLinkHtml;\n",

" element.appendChild(docLink);\n",

" }\n",

" </script>\n",

" </div>\n",

" </div>\n",

" "

]

},

"metadata": {},

"execution\_count": 34

}

]

},

{

"cell\_type": "code",

"source": [

"y=df.iloc[:,-1]\n",

"y.head()"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "QG-ZeLk0NDlK",

"outputId": "9d42ea54-f67b-44fd-c48f-672c412fc2ef"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "execute\_result",

"data": {

"text/plain": [

"0 1\n",

"1 0\n",

"2 1\n",

"3 0\n",

"4 0\n",

"Name: Exited, dtype: int64"

]

},

"metadata": {},

"execution\_count": 35

}

]

},

{

"cell\_type": "markdown",

"source": [

"\*\*9. Scale the independent variables\*\*"

],

"metadata": {

"id": "DM1ZPu30NO-q"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.preprocessing import StandardScaler\n",

"scaler=StandardScaler()\n",

"x=scaler.fit\_transform(x)\n",

"print(x)"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "CXCoGp6INUsJ",

"outputId": "971adcbb-2ec5-4676-9741-4c7a5cb15bd1"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"[[-0.32687761 -0.90188624 -1.09598752 ... 0.64609167 0.97024255\n",

" 0.02188649]\n",

" [-0.44080365 1.51506738 -1.09598752 ... -1.54776799 0.97024255\n",

" 0.21653375]\n",

" [-1.53863634 -0.90188624 -1.09598752 ... 0.64609167 -1.03067011\n",

" 0.2406869 ]\n",

" ...\n",

" [ 0.60524449 -0.90188624 -1.09598752 ... -1.54776799 0.97024255\n",

" -1.00864308]\n",

" [ 1.25772996 0.30659057 0.91241915 ... 0.64609167 -1.03067011\n",

" -0.12523071]\n",

" [ 1.4648682 -0.90188624 -1.09598752 ... 0.64609167 -1.03067011\n",

" -1.07636976]]\n"

]

}

]

},

{

"cell\_type": "markdown",

"source": [

"\*\*10. Split the data into training and testing.\*\*"

],

"metadata": {

"id": "4wSmcKhWNekE"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.model\_selection import train\_test\_split\n",

"x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.20)"

],

"metadata": {

"id": "DotC67DmNg5a"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"print(x\_train.shape)\n",

"print(x\_test.shape)"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "HhXU6ID7NlCS",

"outputId": "85e43a22-1ab7-4bc8-d8c1-8ceb43778022"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"(8000, 10)\n",

"(2000, 10)\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"print(y\_train.shape)\n",

"print(y\_test.shape)"

],

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "dci8h2ABNnc7",

"outputId": "8c54cdf7-6f69-4863-ec15-b9df0c65858e"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"(8000,)\n",

"(2000,)\n"

]

}

]

}

]

}