{

"cells": [

{

"cell\_type": "markdown",

"metadata": {

"id": "uOxsWL4fyQ2J"

},

"source": [

"# Aula 1 - Clasificación: ¿Cómo funciona?"

]

},

{

"cell\_type": "markdown",

"metadata": {

"id": "vuZVqaVCyQ2M"

},

"source": [

"## 1.1 - Importando los datos"

]

},

{

"cell\_type": "code",

"execution\_count": 1,

"metadata": {

"id": "EcFx0fP0yQ2N"

},

"outputs": [],

"source": [

"import pandas as pd"

]

},

{

"cell\_type": "code",

"execution\_count": 2,

"metadata": {

"id": "jMxgxTXnyQ2O"

},

"outputs": [],

"source": [

"datos = pd.read\_csv('Customer.csv')"

]

},

{

"cell\_type": "code",

"execution\_count": 3,

"metadata": {

"id": "neE16a7wyQ2O",

"colab": {

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

},

"outputId": "e92643c8-7c7e-403f-9ec3-12ca7d42ba25"

},

"outputs": [

{

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

"data": {

"text/plain": [

"(7043, 18)"

]

},

"metadata": {},

"execution\_count": 3

}

],

"source": [

"datos.shape"

]

},

{

"cell\_type": "code",

"execution\_count": 4,

"metadata": {

"id": "k\_QTfXOSyQ2P",

"colab": {

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

"height": 270

},

"outputId": "6519a537-5de4-495f-ac59-b70df310ca8b"

},

"outputs": [

{

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

"data": {

"text/plain": [

" Mayor60Años Conyuge Dependientes MesesDeContrato TelefonoFijo \\\n",

"0 0 Si No 1 No \n",

"1 0 No No 34 Si \n",

"2 0 No No 2 Si \n",

"3 0 No No 45 No \n",

"4 0 No No 2 Si \n",

"\n",

" VariasLineasTelefonicas ServicioDeInternet SeguridadOnline BackupOnline \\\n",

"0 SinServicioTelefonico DSL No Si \n",

"1 No DSL Si No \n",

"2 No DSL Si Si \n",

"3 SinServicioTelefonico DSL Si No \n",

"4 No FibraOptica No No \n",

"\n",

" SeguroEnDispositivo SoporteTecnico TVCable Streaming TipoDeContrato \\\n",

"0 No No No No Mensual \n",

"1 Si No No No UnAño \n",

"2 No No No No Mensual \n",

"3 Si Si No No UnAño \n",

"4 No No No No Mensual \n",

"\n",

" PagoOnline FormaDePago CuentaMensual Churn \n",

"0 Si ChequeDigital 29.85 No \n",

"1 No ChequePapel 56.95 No \n",

"2 Si ChequePapel 53.85 Si \n",

"3 No DebitoEnCuenta 42.30 No \n",

"4 Si ChequeDigital 70.70 Si "

],

"text/html": [

"\n",

"\n",

" <div id=\"df-985f9057-6526-47b3-be28-2963cfa1b912\">\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>Mayor60Años</th>\n",

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

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

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

" <td>No</td>\n",

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

" <td>No</td>\n",

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

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

" <td>No</td>\n",

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

" <td>No</td>\n",

" <td>No</td>\n",

" <td>No</td>\n",

" <td>No</td>\n",

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

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

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

" <td>29.85</td>\n",

" <td>No</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>No</td>\n",

" <td>No</td>\n",

" <td>34</td>\n",

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

" <td>No</td>\n",

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

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

" <td>No</td>\n",

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

" <td>No</td>\n",

" <td>No</td>\n",

" <td>No</td>\n",

" <td>UnAño</td>\n",

" <td>No</td>\n",

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

" <td>56.95</td>\n",

" <td>No</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>No</td>\n",

" <td>No</td>\n",

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

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

" <td>No</td>\n",

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

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

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

" <td>No</td>\n",

" <td>No</td>\n",

" <td>No</td>\n",

" <td>No</td>\n",

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

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

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

" <td>53.85</td>\n",

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

" </tr>\n",

" <tr>\n",

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

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

" <td>No</td>\n",

" <td>No</td>\n",

" <td>45</td>\n",

" <td>No</td>\n",

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

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

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

" <td>No</td>\n",

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

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

" <td>No</td>\n",

" <td>No</td>\n",

" <td>UnAño</td>\n",

" <td>No</td>\n",

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

" <td>42.30</td>\n",

" <td>No</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>No</td>\n",

" <td>No</td>\n",

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

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

" <td>No</td>\n",

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

" <td>No</td>\n",

" <td>No</td>\n",

" <td>No</td>\n",

" <td>No</td>\n",

" <td>No</td>\n",

" <td>No</td>\n",

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

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

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

" <td>70.70</td>\n",

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-985f9057-6526-47b3-be28-2963cfa1b912')\"\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",

"\n",

"\n",

" <div id=\"df-7291a366-9390-42ec-ae02-241eeb1223ec\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-7291a366-9390-42ec-ae02-241eeb1223ec')\"\n",

" title=\"Suggest charts.\"\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",

" <g>\n",

" <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",

" </g>\n",

"</svg>\n",

" </button>\n",

" </div>\n",

"\n",

"<style>\n",

" .colab-df-quickchart {\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-quickchart: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-quickchart {\n",

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

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-quickchart: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",

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

" const containerElement = document.querySelector('#' + key);\n",

" const charts = await google.colab.kernel.invokeFunction(\n",

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

" }\n",

" </script>\n",

"\n",

" <script>\n",

"\n",

"function displayQuickchartButton(domScope) {\n",

" let quickchartButtonEl =\n",

" domScope.querySelector('#df-7291a366-9390-42ec-ae02-241eeb1223ec button.colab-df-quickchart');\n",

" quickchartButtonEl.style.display =\n",

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

"}\n",

"\n",

" displayQuickchartButton(document);\n",

" </script>\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-985f9057-6526-47b3-be28-2963cfa1b912 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-985f9057-6526-47b3-be28-2963cfa1b912');\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

}

],

"source": [

"datos.head()"

]

},

{

"cell\_type": "markdown",

"metadata": {

"id": "V8lNCeRnyQ2P"

},

"source": [

"## 1.2 - Analizando las Variables"

]

},

{

"cell\_type": "code",

"execution\_count": 5,

"metadata": {

"id": "uQLvJoZ0yQ2Q",

"colab": {

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

"height": 206

},

"outputId": "6f5f0040-5d68-4084-c403-feb9535ff60a"

},

"outputs": [

{

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

"data": {

"text/plain": [

" Conyuge Dependientes TelefonoFijo PagoOnline Churn\n",

"0 1 0 0 1 0\n",

"1 0 0 1 0 0\n",

"2 0 0 1 1 1\n",

"3 0 0 0 0 0\n",

"4 0 0 1 1 1"

],

"text/html": [

"\n",

"\n",

" <div id=\"df-a97f9251-2dea-42ae-a744-87e7a8456e8e\">\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>Conyuge</th>\n",

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

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

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

" <th>Churn</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",

" <td>1</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",

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

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

" </tr>\n",

" <tr>\n",

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

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

" <td>0</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",

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

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-a97f9251-2dea-42ae-a744-87e7a8456e8e')\"\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",

"\n",

"\n",

" <div id=\"df-8a0f3eec-c32d-4f35-99a2-538f6a7dfc7f\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-8a0f3eec-c32d-4f35-99a2-538f6a7dfc7f')\"\n",

" title=\"Suggest charts.\"\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",

" <g>\n",

" <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",

" </g>\n",

"</svg>\n",

" </button>\n",

" </div>\n",

"\n",

"<style>\n",

" .colab-df-quickchart {\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-quickchart: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-quickchart {\n",

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

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-quickchart: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",

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

" const containerElement = document.querySelector('#' + key);\n",

" const charts = await google.colab.kernel.invokeFunction(\n",

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

" }\n",

" </script>\n",

"\n",

" <script>\n",

"\n",

"function displayQuickchartButton(domScope) {\n",

" let quickchartButtonEl =\n",

" domScope.querySelector('#df-8a0f3eec-c32d-4f35-99a2-538f6a7dfc7f button.colab-df-quickchart');\n",

" quickchartButtonEl.style.display =\n",

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

"}\n",

"\n",

" displayQuickchartButton(document);\n",

" </script>\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-a97f9251-2dea-42ae-a744-87e7a8456e8e 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-a97f9251-2dea-42ae-a744-87e7a8456e8e');\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": 5

}

],

"source": [

"#Modificación de forma manual\n",

"diccionario = {'Si': 1,\n",

" 'No': 0}\n",

"\n",

"datosmodificados = datos[['Conyuge', 'Dependientes', 'TelefonoFijo', 'PagoOnline', 'Churn']].replace(diccionario)\n",

"datosmodificados.head()"

]

},

{

"cell\_type": "code",

"execution\_count": 6,

"metadata": {

"id": "0Dc3-8wayQ2R"

},

"outputs": [],

"source": [

"#Transformación con get\_dummies\n",

"dummie\_datos = pd.get\_dummies(datos.drop(['Conyuge', 'Dependientes', 'TelefonoFijo', 'PagoOnline', 'Churn'],\n",

" axis=1))\n",

"\n",

"#Unión de los datos transformados con los que ya teníamos\n",

"datos\_final = pd.concat([datosmodificados, dummie\_datos], axis=1)"

]

},

{

"cell\_type": "code",

"source": [

"datos\_final.head()"

],

"metadata": {

"id": "UmQrjUnh08vu",

"colab": {

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

"height": 300

},

"outputId": "b2744b0b-d088-42bb-e2a1-3be26d197934"

},

"execution\_count": 7,

"outputs": [

{

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

"data": {

"text/plain": [

" Conyuge Dependientes TelefonoFijo PagoOnline Churn Mayor60Años \\\n",

"0 1 0 0 1 0 0 \n",

"1 0 0 1 0 0 0 \n",

"2 0 0 1 1 1 0 \n",

"3 0 0 0 0 0 0 \n",

"4 0 0 1 1 1 0 \n",

"\n",

" MesesDeContrato CuentaMensual VariasLineasTelefonicas\_No \\\n",

"0 1 29.85 0 \n",

"1 34 56.95 1 \n",

"2 2 53.85 1 \n",

"3 45 42.30 0 \n",

"4 2 70.70 1 \n",

"\n",

" VariasLineasTelefonicas\_Si ... Streaming\_No Streaming\_Si \\\n",

"0 0 ... 1 0 \n",

"1 0 ... 1 0 \n",

"2 0 ... 1 0 \n",

"3 0 ... 1 0 \n",

"4 0 ... 1 0 \n",

"\n",

" Streaming\_SinServicioDeInternet TipoDeContrato\_DosAños \\\n",

"0 0 0 \n",

"1 0 0 \n",

"2 0 0 \n",

"3 0 0 \n",

"4 0 0 \n",

"\n",

" TipoDeContrato\_Mensual TipoDeContrato\_UnAño FormaDePago\_ChequeDigital \\\n",

"0 1 0 1 \n",

"1 0 1 0 \n",

"2 1 0 0 \n",

"3 0 1 0 \n",

"4 1 0 1 \n",

"\n",

" FormaDePago\_ChequePapel FormaDePago\_DebitoEnCuenta \\\n",

"0 0 0 \n",

"1 1 0 \n",

"2 1 0 \n",

"3 0 1 \n",

"4 0 0 \n",

"\n",

" FormaDePago\_TarjetaDeCredito \n",

"0 0 \n",

"1 0 \n",

"2 0 \n",

"3 0 \n",

"4 0 \n",

"\n",

"[5 rows x 39 columns]"

],

"text/html": [

"\n",

"\n",

" <div id=\"df-6c3a9d78-6091-49c2-9ad3-357169389d85\">\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>Conyuge</th>\n",

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

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

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

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

" <th>Mayor60Años</th>\n",

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

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

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

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

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

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

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

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

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

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

" <th>TipoDeContrato\_UnAño</th>\n",

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

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

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

" <th>FormaDePago\_TarjetaDeCredito</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",

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

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

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

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

" <td>29.85</td>\n",

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

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

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

" <td>1</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",

" <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",

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

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

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

" <td>34</td>\n",

" <td>56.95</td>\n",

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

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

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

" <td>1</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",

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

" </tr>\n",

" <tr>\n",

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

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

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

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

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

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

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

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

" <td>53.85</td>\n",

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

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

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

" <td>1</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",

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

" </tr>\n",

" <tr>\n",

" <th>3</th>\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>45</td>\n",

" <td>42.30</td>\n",

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

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

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

" <td>1</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>0</td>\n",

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

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

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

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

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

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

" <td>70.70</td>\n",

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

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

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

" <td>1</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",

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

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>5 rows × 39 columns</p>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-6c3a9d78-6091-49c2-9ad3-357169389d85')\"\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",

"\n",

"\n",

" <div id=\"df-02da5e36-00bd-4c36-8f87-d7671271e50d\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-02da5e36-00bd-4c36-8f87-d7671271e50d')\"\n",

" title=\"Suggest charts.\"\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",

" <g>\n",

" <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",

" </g>\n",

"</svg>\n",

" </button>\n",

" </div>\n",

"\n",

"<style>\n",

" .colab-df-quickchart {\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-quickchart: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-quickchart {\n",

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

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-quickchart: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",

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

" const containerElement = document.querySelector('#' + key);\n",

" const charts = await google.colab.kernel.invokeFunction(\n",

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

" }\n",

" </script>\n",

"\n",

" <script>\n",

"\n",

"function displayQuickchartButton(domScope) {\n",

" let quickchartButtonEl =\n",

" domScope.querySelector('#df-02da5e36-00bd-4c36-8f87-d7671271e50d button.colab-df-quickchart');\n",

" quickchartButtonEl.style.display =\n",

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

"}\n",

"\n",

" displayQuickchartButton(document);\n",

" </script>\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-6c3a9d78-6091-49c2-9ad3-357169389d85 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-6c3a9d78-6091-49c2-9ad3-357169389d85');\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": 7

}

]

},

{

"cell\_type": "markdown",

"source": [

"## 1.3 - Definición Formal"

],

"metadata": {

"id": "D1YLHA3h37s1"

}

},

{

"cell\_type": "markdown",

"source": [

"Informaciones para la clasificación:\n",

"\n",

"$X$ = inputs (datos de entrada)\n",

"\n",

"$y$ = outputs (datos de salida)"

],

"metadata": {

"id": "sSgnt7QWM5rm"

}

},

{

"cell\_type": "code",

"source": [

"#TIP\n",

"pd.set\_option('display.max\_columns', 39)"

],

"metadata": {

"id": "k\_o0S4UyOqzd"

},

"execution\_count": 8,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"datos\_final.head()"

],

"metadata": {

"id": "oiHVlSC737NE",

"colab": {

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

"height": 270

},

"outputId": "2e9c757b-095f-45ad-f243-b1d1f92cb35f"

},

"execution\_count": 9,

"outputs": [

{

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

"data": {

"text/plain": [

" Conyuge Dependientes TelefonoFijo PagoOnline Churn Mayor60Años \\\n",

"0 1 0 0 1 0 0 \n",

"1 0 0 1 0 0 0 \n",

"2 0 0 1 1 1 0 \n",

"3 0 0 0 0 0 0 \n",

"4 0 0 1 1 1 0 \n",

"\n",

" MesesDeContrato CuentaMensual VariasLineasTelefonicas\_No \\\n",

"0 1 29.85 0 \n",

"1 34 56.95 1 \n",

"2 2 53.85 1 \n",

"3 45 42.30 0 \n",

"4 2 70.70 1 \n",

"\n",

" VariasLineasTelefonicas\_Si VariasLineasTelefonicas\_SinServicioTelefonico \\\n",

"0 0 1 \n",

"1 0 0 \n",

"2 0 0 \n",

"3 0 1 \n",

"4 0 0 \n",

"\n",

" ServicioDeInternet\_DSL ServicioDeInternet\_FibraOptica \\\n",

"0 1 0 \n",

"1 1 0 \n",

"2 1 0 \n",

"3 1 0 \n",

"4 0 1 \n",

"\n",

" ServicioDeInternet\_No SeguridadOnline\_No SeguridadOnline\_Si \\\n",

"0 0 1 0 \n",

"1 0 0 1 \n",

"2 0 0 1 \n",

"3 0 0 1 \n",

"4 0 1 0 \n",

"\n",

" SeguridadOnline\_SinServicioDeInternet BackupOnline\_No BackupOnline\_Si \\\n",

"0 0 0 1 \n",

"1 0 1 0 \n",

"2 0 0 1 \n",

"3 0 1 0 \n",

"4 0 1 0 \n",

"\n",

" BackupOnline\_SinServicioDeInternet SeguroEnDispositivo\_No \\\n",

"0 0 1 \n",

"1 0 0 \n",

"2 0 1 \n",

"3 0 0 \n",

"4 0 1 \n",

"\n",

" SeguroEnDispositivo\_Si SeguroEnDispositivo\_SinServicioDeInternet \\\n",

"0 0 0 \n",

"1 1 0 \n",

"2 0 0 \n",

"3 1 0 \n",

"4 0 0 \n",

"\n",

" SoporteTecnico\_No SoporteTecnico\_Si SoporteTecnico\_SinServicioDeInternet \\\n",

"0 1 0 0 \n",

"1 1 0 0 \n",

"2 1 0 0 \n",

"3 0 1 0 \n",

"4 1 0 0 \n",

"\n",

" TVCable\_No TVCable\_Si TVCable\_SinServicioDeInternet Streaming\_No \\\n",

"0 1 0 0 1 \n",

"1 1 0 0 1 \n",

"2 1 0 0 1 \n",

"3 1 0 0 1 \n",

"4 1 0 0 1 \n",

"\n",

" Streaming\_Si Streaming\_SinServicioDeInternet TipoDeContrato\_DosAños \\\n",

"0 0 0 0 \n",

"1 0 0 0 \n",

"2 0 0 0 \n",

"3 0 0 0 \n",

"4 0 0 0 \n",

"\n",

" TipoDeContrato\_Mensual TipoDeContrato\_UnAño FormaDePago\_ChequeDigital \\\n",

"0 1 0 1 \n",

"1 0 1 0 \n",

"2 1 0 0 \n",

"3 0 1 0 \n",

"4 1 0 1 \n",

"\n",

" FormaDePago\_ChequePapel FormaDePago\_DebitoEnCuenta \\\n",

"0 0 0 \n",

"1 1 0 \n",

"2 1 0 \n",

"3 0 1 \n",

"4 0 0 \n",

"\n",

" FormaDePago\_TarjetaDeCredito \n",

"0 0 \n",

"1 0 \n",

"2 0 \n",

"3 0 \n",

"4 0 "

],

"text/html": [

"\n",

"\n",

" <div id=\"df-fcb91a5d-9221-4f38-8507-a786e330ce79\">\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>Conyuge</th>\n",

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

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

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

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

" <th>Mayor60Años</th>\n",

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

" <th>TipoDeContrato\_UnAño</th>\n",

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

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

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

" <th>FormaDePago\_TarjetaDeCredito</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",

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

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

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

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

" <td>29.85</td>\n",

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

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

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

" <td>1</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",

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

" <td>1</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",

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

" <td>1</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",

" <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",

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

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

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

" <td>34</td>\n",

" <td>56.95</td>\n",

" <td>1</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",

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

" <td>1</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",

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

" <td>1</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>0</td>\n",

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

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

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

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

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

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

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

" <td>53.85</td>\n",

" <td>1</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",

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

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

" <td>1</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",

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

" <td>1</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",

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

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

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

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

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

" </tr>\n",

" <tr>\n",

" <th>3</th>\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>45</td>\n",

" <td>42.30</td>\n",

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

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

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

" <td>1</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",

" <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",

" <td>1</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>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>0</td>\n",

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

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

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

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

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

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

" <td>70.70</td>\n",

" <td>1</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",

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

" <td>1</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",

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

" <td>1</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",

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

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

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

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

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-fcb91a5d-9221-4f38-8507-a786e330ce79')\"\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",

"\n",

"\n",

" <div id=\"df-ddce61b4-c0ef-4ef4-aa10-a2ce75679e21\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-ddce61b4-c0ef-4ef4-aa10-a2ce75679e21')\"\n",

" title=\"Suggest charts.\"\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",

" <g>\n",

" <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",

" </g>\n",

"</svg>\n",

" </button>\n",

" </div>\n",

"\n",

"<style>\n",

" .colab-df-quickchart {\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-quickchart: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-quickchart {\n",

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

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-quickchart: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",

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

" const containerElement = document.querySelector('#' + key);\n",

" const charts = await google.colab.kernel.invokeFunction(\n",

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

" }\n",

" </script>\n",

"\n",

" <script>\n",

"\n",

"function displayQuickchartButton(domScope) {\n",

" let quickchartButtonEl =\n",

" domScope.querySelector('#df-ddce61b4-c0ef-4ef4-aa10-a2ce75679e21 button.colab-df-quickchart');\n",

" quickchartButtonEl.style.display =\n",

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

"}\n",

"\n",

" displayQuickchartButton(document);\n",

" </script>\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-fcb91a5d-9221-4f38-8507-a786e330ce79 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-fcb91a5d-9221-4f38-8507-a786e330ce79');\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

}

]

},

{

"cell\_type": "markdown",

"source": [

"\n",

"$y\_i$ = $f(x\_i)$"

],

"metadata": {

"id": "mlxxn4VYTQnV"

}

},

{

"cell\_type": "code",

"source": [

"Xmaria = [[0,0,1,1,0,0,39.90,1,0,0,0,1,0,1,0,0,0,0,1,1,1,0,0,1,0,1,0,0,0,0,1,0,0,1,0,0,0,1]]"

],

"metadata": {

"id": "JVqSH1yp4Cq2"

},

"execution\_count": 10,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#ymaria = ?"

],

"metadata": {

"id": "1Btl3Wct4CoN"

},

"execution\_count": 11,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"Nuevo par de informaciones = ($Xmaria$, $ymaria$)"

],

"metadata": {

"id": "n4nyxwPFTncE"

}

},

{

"cell\_type": "markdown",

"source": [

"## 1.4 - Balanceamiento de los datos"

],

"metadata": {

"id": "zCcuXt0FZfNe"

}

},

{

"cell\_type": "code",

"source": [

"#variable target está desbalanceada\n",

"import seaborn as sns\n",

"%matplotlib inline\n",

"ax = sns.countplot(x='Churn', data=datos\_final)"

],

"metadata": {

"id": "PZ98-642QERQ",

"colab": {

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

"height": 449

},

"outputId": "92a270b9-7bde-45c6-fe91-d8b28d5dd556"

},

"execution\_count": 12,

"outputs": [

{

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

"data": {

"text/plain": [

"<Figure size 640x480 with 1 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"cell\_type": "code",

"source": [

"datos\_final.Churn.value\_counts()"

],

"metadata": {

"id": "tDmNwxUGQt0U",

"colab": {

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

},

"outputId": "77b7e928-6dd4-4225-95b1-4544aa4bb55d"

},

"execution\_count": 13,

"outputs": [

{

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

"data": {

"text/plain": [

"0 5174\n",

"1 1869\n",

"Name: Churn, dtype: int64"

]

},

"metadata": {},

"execution\_count": 13

}

]

},

{

"cell\_type": "code",

"source": [

"#biblioteca para balancear los datos utilizando over\_sampling\n",

"from imblearn.over\_sampling import SMOTE"

],

"metadata": {

"id": "TcFY4SXuZyIO"

},

"execution\_count": 14,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#dividiendo los datos en características y target\n",

"X = datos\_final.drop('Churn', axis = 1)\n",

"y = datos\_final['Churn']"

],

"metadata": {

"id": "0-I2aDVhQUdu"

},

"execution\_count": 15,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"smt = SMOTE(random\_state=123)\n",

"X, y = smt.fit\_resample(X, y)"

],

"metadata": {

"id": "5lIY03boZyBN"

},

"execution\_count": 16,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#unión de los datos balanceados\n",

"datos\_final = pd.concat([X, y], axis=1)"

],

"metadata": {

"id": "bnlqzLVUZl0u"

},

"execution\_count": 17,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#verificación 1 - unión de los datos\n",

"datos\_final.head(2)"

],

"metadata": {

"id": "GQj3bEEOZ42e",

"colab": {

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

"height": 176

},

"outputId": "b5843d2d-3c0d-4cfe-aea9-526a820f1fd1"

},

"execution\_count": 18,

"outputs": [

{

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

"data": {

"text/plain": [

" Conyuge Dependientes TelefonoFijo PagoOnline Mayor60Años \\\n",

"0 1 0 0 1 0 \n",

"1 0 0 1 0 0 \n",

"\n",

" MesesDeContrato CuentaMensual VariasLineasTelefonicas\_No \\\n",

"0 1 29.85 0 \n",

"1 34 56.95 1 \n",

"\n",

" VariasLineasTelefonicas\_Si VariasLineasTelefonicas\_SinServicioTelefonico \\\n",

"0 0 1 \n",

"1 0 0 \n",

"\n",

" ServicioDeInternet\_DSL ServicioDeInternet\_FibraOptica \\\n",

"0 1 0 \n",

"1 1 0 \n",

"\n",

" ServicioDeInternet\_No SeguridadOnline\_No SeguridadOnline\_Si \\\n",

"0 0 1 0 \n",

"1 0 0 1 \n",

"\n",

" SeguridadOnline\_SinServicioDeInternet BackupOnline\_No BackupOnline\_Si \\\n",

"0 0 0 1 \n",

"1 0 1 0 \n",

"\n",

" BackupOnline\_SinServicioDeInternet SeguroEnDispositivo\_No \\\n",

"0 0 1 \n",

"1 0 0 \n",

"\n",

" SeguroEnDispositivo\_Si SeguroEnDispositivo\_SinServicioDeInternet \\\n",

"0 0 0 \n",

"1 1 0 \n",

"\n",

" SoporteTecnico\_No SoporteTecnico\_Si SoporteTecnico\_SinServicioDeInternet \\\n",

"0 1 0 0 \n",

"1 1 0 0 \n",

"\n",

" TVCable\_No TVCable\_Si TVCable\_SinServicioDeInternet Streaming\_No \\\n",

"0 1 0 0 1 \n",

"1 1 0 0 1 \n",

"\n",

" Streaming\_Si Streaming\_SinServicioDeInternet TipoDeContrato\_DosAños \\\n",

"0 0 0 0 \n",

"1 0 0 0 \n",

"\n",

" TipoDeContrato\_Mensual TipoDeContrato\_UnAño FormaDePago\_ChequeDigital \\\n",

"0 1 0 1 \n",

"1 0 1 0 \n",

"\n",

" FormaDePago\_ChequePapel FormaDePago\_DebitoEnCuenta \\\n",

"0 0 0 \n",

"1 1 0 \n",

"\n",

" FormaDePago\_TarjetaDeCredito Churn \n",

"0 0 0 \n",

"1 0 0 "

],

"text/html": [

"\n",

"\n",

" <div id=\"df-2bce39df-f060-40c6-818c-d033862aa0d8\">\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>Conyuge</th>\n",

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

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

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

" <th>Mayor60Años</th>\n",

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

" <th>TipoDeContrato\_UnAño</th>\n",

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

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

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

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

" <th>Churn</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",

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

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

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

" <td>29.85</td>\n",

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

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

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

" <td>1</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",

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

" <td>1</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",

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

" <td>1</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",

" <td>0</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",

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

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

" <td>34</td>\n",

" <td>56.95</td>\n",

" <td>1</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",

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

" <td>1</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",

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

" <td>1</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>0</td>\n",

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

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

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

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

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

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-2bce39df-f060-40c6-818c-d033862aa0d8')\"\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",

"\n",

"\n",

" <div id=\"df-e8940d1b-a431-4033-9f45-8175d66d339a\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-e8940d1b-a431-4033-9f45-8175d66d339a')\"\n",

" title=\"Suggest charts.\"\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",

" <g>\n",

" <path d=\"M19 3H5c-1.1 0-2 .9-2 2v14c0 1.1.9 2 2 2h14c1.1 0 2-.9 2-2V5c0-1.1-.9-2-2-2zM9 17H7v-7h2v7zm4 0h-2V7h2v10zm4 0h-2v-4h2v4z\"/>\n",

" </g>\n",

"</svg>\n",

" </button>\n",

" </div>\n",

"\n",

"<style>\n",

" .colab-df-quickchart {\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-quickchart: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-quickchart {\n",

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

" fill: #D2E3FC;\n",

" }\n",

"\n",

" [theme=dark] .colab-df-quickchart: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",

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

" const containerElement = document.querySelector('#' + key);\n",

" const charts = await google.colab.kernel.invokeFunction(\n",

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

" }\n",

" </script>\n",

"\n",

" <script>\n",

"\n",

"function displayQuickchartButton(domScope) {\n",

" let quickchartButtonEl =\n",

" domScope.querySelector('#df-e8940d1b-a431-4033-9f45-8175d66d339a button.colab-df-quickchart');\n",

" quickchartButtonEl.style.display =\n",

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

"}\n",

"\n",

" displayQuickchartButton(document);\n",

" </script>\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-2bce39df-f060-40c6-818c-d033862aa0d8 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-2bce39df-f060-40c6-818c-d033862aa0d8');\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": 18

}

]

},

{

"cell\_type": "code",

"source": [

"#verificación 2 - balanceamiento\n",

"ax = sns.countplot(x='Churn', data=datos\_final)"

],

"metadata": {

"id": "jSzaHOlyZ7r-",

"colab": {

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

"height": 449

},

"outputId": "72b49417-4310-4954-fe10-6c0bbff76f51"

},

"execution\_count": 19,

"outputs": [

{

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

"data": {

"text/plain": [

"<Figure size 640x480 with 1 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"cell\_type": "code",

"source": [

"datos\_final.Churn.value\_counts()"

],

"metadata": {

"id": "Zx0HYHd3RGl7",

"colab": {

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

},

"outputId": "272b6207-9e7a-4a66-cc68-a10a2bc4a7b0"

},

"execution\_count": 20,

"outputs": [

{

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

"data": {

"text/plain": [

"0 5174\n",

"1 5174\n",

"Name: Churn, dtype: int64"

]

},

"metadata": {},

"execution\_count": 20

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Aula 2 - Método baseado en la proximidad"

],

"metadata": {

"id": "\_d4jEXubcPuO"

}

},

{

"cell\_type": "markdown",

"source": [

"## 2.1 - Modelo K-nearest neighbors (KNN)\n",

"\n",

"(PPT)"

],

"metadata": {

"id": "PVGb9U19cW4V"

}

},

{

"cell\_type": "markdown",

"source": [

"## 2.2 - KNN en la práctica"

],

"metadata": {

"id": "MvIfmyE2ckVV"

}

},

{

"cell\_type": "code",

"source": [

"Xmaria"

],

"metadata": {

"id": "blRWRPdXcWJ1",

"colab": {

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

},

"outputId": "fda0e39b-871f-4992-f79d-51333ddf5255"

},

"execution\_count": 21,

"outputs": [

{

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

"data": {

"text/plain": [

"[[0,\n",

" 0,\n",

" 1,\n",

" 1,\n",

" 0,\n",

" 0,\n",

" 39.9,\n",

" 1,\n",

" 0,\n",

" 0,\n",

" 0,\n",

" 1,\n",

" 0,\n",

" 1,\n",

" 0,\n",

" 0,\n",

" 0,\n",

" 0,\n",

" 1,\n",

" 1,\n",

" 1,\n",

" 0,\n",

" 0,\n",

" 1,\n",

" 0,\n",

" 1,\n",

" 0,\n",

" 0,\n",

" 0,\n",

" 0,\n",

" 1,\n",

" 0,\n",

" 0,\n",

" 1,\n",

" 0,\n",

" 0,\n",

" 0,\n",

" 1]]"

]

},

"metadata": {},

"execution\_count": 21

}

]

},

{

"cell\_type": "code",

"source": [

"#ymaria = ?"

],

"metadata": {

"id": "tdwhZOphjbIg"

},

"execution\_count": 22,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#División en inputs y outputs\n",

"X = datos\_final.drop('Churn', axis = 1)\n",

"y = datos\_final['Churn']"

],

"metadata": {

"id": "h0re3ABxjlLW"

},

"execution\_count": 23,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#biblioteca para padronizar los datos\n",

"from sklearn.preprocessing import StandardScaler"

],

"metadata": {

"id": "cw\_-29nlkLMV"

},

"execution\_count": 24,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"norm = StandardScaler()\n",

"X\_normalizado = norm.fit\_transform(X)\n",

"X\_normalizado"

],

"metadata": {

"id": "zGYhtla\_kgpd",

"colab": {

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

},

"outputId": "d156c917-0827-4720-c5a3-57d6de0e3f34"

},

"execution\_count": 25,

"outputs": [

{

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

"data": {

"text/plain": [

"array([[ 1.26542767, -0.52771004, -3.06083934, ..., -0.47912706,\n",

" -0.4360152 , -0.42576817],\n",

" [-0.79024667, -0.52771004, 0.32670777, ..., 2.08712905,\n",

" -0.4360152 , -0.42576817],\n",

" [-0.79024667, -0.52771004, 0.32670777, ..., 2.08712905,\n",

" -0.4360152 , -0.42576817],\n",

" ...,\n",

" [-0.79024667, -0.52771004, 0.32670777, ..., -0.47912706,\n",

" -0.4360152 , -0.42576817],\n",

" [-0.79024667, -0.52771004, 0.32670777, ..., -0.47912706,\n",

" -0.4360152 , 2.34869598],\n",

" [-0.79024667, -0.52771004, 0.32670777, ..., -0.47912706,\n",

" -0.4360152 , 2.34869598]])"

]

},

"metadata": {},

"execution\_count": 25

}

]

},

{

"cell\_type": "code",

"source": [

"X\_normalizado[0]"

],

"metadata": {

"id": "Cs-ivgv7qsKL",

"colab": {

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

},

"outputId": "b61c0120-5b4a-412d-e202-2c243b0778c7"

},

"execution\_count": 26,

"outputs": [

{

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

"data": {

"text/plain": [

"array([ 1.26542767, -0.52771004, -3.06083934, 0.81482091, -0.41318358,\n",

" -1.10771815, -1.3217706 , -0.88447396, -0.83366906, 3.18604176,\n",

" 1.51268338, -1.04386816, -0.44617598, 0.86827317, -0.51721942,\n",

" -0.44617598, -0.95834643, 1.57389661, -0.44617598, 1.04285807,\n",

" -0.64211419, -0.44617598, 0.87979784, -0.52426638, -0.44617598,\n",

" 1.18690873, -0.77707385, -0.44617598, 1.20311799, -0.78976268,\n",

" -0.44617598, -0.44648735, 0.73788726, -0.41989297, 1.30373042,\n",

" -0.47912706, -0.4360152 , -0.42576817])"

]

},

"metadata": {},

"execution\_count": 26

}

]

},

{

"cell\_type": "code",

"source": [

"Xmaria\_normalizado = norm.transform(pd.DataFrame(Xmaria, columns = X.columns))\n",

"Xmaria\_normalizado"

],

"metadata": {

"id": "Ll6FnFO-kosd",

"colab": {

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

},

"outputId": "6a68cf5b-92eb-4e37-d6a6-89058fd58876"

},

"execution\_count": 27,

"outputs": [

{

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

"data": {

"text/plain": [

"array([[-0.79024667, -0.52771004, 0.32670777, 0.81482091, -0.41318358,\n",

" -1.14938255, -0.97236043, 1.13061554, -0.83366906, -0.31386908,\n",

" -0.66107688, 0.95797538, -0.44617598, 0.86827317, -0.51721942,\n",

" -0.44617598, -0.95834643, -0.63536575, 2.24126814, 1.04285807,\n",

" 1.5573554 , -0.44617598, -1.13662476, 1.90742727, -0.44617598,\n",

" 1.18690873, -0.77707385, -0.44617598, -0.83117367, -0.78976268,\n",

" 2.24126814, -0.44648735, -1.35522058, 2.38155929, -0.76702973,\n",

" -0.47912706, -0.4360152 , 2.34869598]])"

]

},

"metadata": {},

"execution\_count": 27

}

]

},

{

"cell\_type": "markdown",

"source": [

"Distancia Euclidiana:\n",

"\n",

"$\\sqrt{\\sum\_{i=1}^k(a\_{i}-b\_{i})^2}$\n"

],

"metadata": {

"id": "pgU6OxezlB\_F"

}

},

{

"cell\_type": "code",

"source": [

"import numpy as np"

],

"metadata": {

"id": "SbxwDdiIk36F"

},

"execution\_count": 28,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"a = Xmaria\_normalizado"

],

"metadata": {

"id": "9oywjidemxXl"

},

"execution\_count": 29,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"b = X\_normalizado[0]"

],

"metadata": {

"id": "JL85j369li4d"

},

"execution\_count": 30,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#1 - comenzamos restando\n",

"a - b"

],

"metadata": {

"id": "mmhEqixanDgF",

"colab": {

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

},

"outputId": "63a374c6-a326-45d4-fd91-e644a3f72366"

},

"execution\_count": 31,

"outputs": [

{

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

"data": {

"text/plain": [

"array([[-2.05567434, 0. , 3.38754711, 0. , 0. ,\n",

" -0.0416644 , 0.34941017, 2.0150895 , 0. , -3.49991084,\n",

" -2.17376025, 2.00184354, 0. , 0. , 0. ,\n",

" 0. , 0. , -2.20926236, 2.68744412, 0. ,\n",

" 2.19946959, 0. , -2.01642259, 2.43169366, 0. ,\n",

" 0. , 0. , 0. , -2.03429166, 0. ,\n",

" 2.68744412, 0. , -2.09310784, 2.80145225, -2.07076016,\n",

" 0. , 0. , 2.77446415]])"

]

},

"metadata": {},

"execution\_count": 31

}

]

},

{

"cell\_type": "code",

"source": [

"#2 - después realizamos la exponenciación\n",

"np.square(a-b)"

],

"metadata": {

"id": "\_aBlq3wPnF49",

"colab": {

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

},

"outputId": "dac6a262-b4ac-4259-b34b-70e4efc7f56d"

},

"execution\_count": 32,

"outputs": [

{

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

"data": {

"text/plain": [

"array([[4.22579699e+00, 0.00000000e+00, 1.14754754e+01, 0.00000000e+00,\n",

" 0.00000000e+00, 1.73592202e-03, 1.22087466e-01, 4.06058568e+00,\n",

" 0.00000000e+00, 1.22493759e+01, 4.72523363e+00, 4.00737757e+00,\n",

" 0.00000000e+00, 0.00000000e+00, 0.00000000e+00, 0.00000000e+00,\n",

" 0.00000000e+00, 4.88084017e+00, 7.22235589e+00, 0.00000000e+00,\n",

" 4.83766647e+00, 0.00000000e+00, 4.06596007e+00, 5.91313404e+00,\n",

" 0.00000000e+00, 0.00000000e+00, 0.00000000e+00, 0.00000000e+00,\n",

" 4.13834258e+00, 0.00000000e+00, 7.22235589e+00, 0.00000000e+00,\n",

" 4.38110042e+00, 7.84813473e+00, 4.28804763e+00, 0.00000000e+00,\n",

" 0.00000000e+00, 7.69765133e+00]])"

]

},

"metadata": {},

"execution\_count": 32

}

]

},

{

"cell\_type": "code",

"source": [

"#3 - realizamos la suma\n",

"np.sum(np.square(a-b))"

],

"metadata": {

"id": "\_bDEeU\_znIxH",

"colab": {

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

},

"outputId": "b540c6de-d5c5-45d9-90ef-ae74239604ac"

},

"execution\_count": 33,

"outputs": [

{

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

"data": {

"text/plain": [

"103.36325779671671"

]

},

"metadata": {},

"execution\_count": 33

}

]

},

{

"cell\_type": "code",

"source": [

"#4 - finalmente obtenemos la raiz cuadrada y tenemos nuestra distancia\n",

"np.sqrt(103.36325779671671)"

],

"metadata": {

"id": "o9NJQSP-nMUm",

"colab": {

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

},

"outputId": "76f74eda-6bb6-4512-a5c9-3a5db3211362"

},

"execution\_count": 34,

"outputs": [

{

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

"data": {

"text/plain": [

"10.166772240820423"

]

},

"metadata": {},

"execution\_count": 34

}

]

},

{

"cell\_type": "markdown",

"source": [

"## 2.3 - Implementando el modelo"

],

"metadata": {

"id": "\_h0pFQfxwYW-"

}

},

{

"cell\_type": "code",

"source": [

"#biblioteca para división de los datos\n",

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

],

"metadata": {

"id": "XtvZ-tMiwjN2"

},

"execution\_count": 35,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X\_train, X\_test, y\_train, y\_test = train\_test\_split(X\_normalizado, y, test\_size=0.3, random\_state=123)"

],

"metadata": {

"id": "lWaKqJrTwxE-"

},

"execution\_count": 36,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"### Entrenamiento y prueba"

],

"metadata": {

"id": "LOKzwv4Uw0u7"

}

},

{

"cell\_type": "code",

"source": [

"#biblioteca para crear el modelo de machine learning\n",

"from sklearn.neighbors import KNeighborsClassifier"

],

"metadata": {

"id": "g6-t-BpEwy3G"

},

"execution\_count": 37,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#iniciar el modelo (creamos el modelo) - por default son 5 vecinos\n",

"knn = KNeighborsClassifier(metric='euclidean')"

],

"metadata": {

"id": "qFs4VHnhw4-O"

},

"execution\_count": 38,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#entrenando el modelo con los datos de entrenamiento\n",

"knn.fit(X\_train, y\_train)"

],

"metadata": {

"id": "H05zbtgIw6gt",

"colab": {

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

"height": 75

},

"outputId": "b4b83d04-2202-4978-c1c8-4cf6df42e3bc"

},

"execution\_count": 39,

"outputs": [

{

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

"data": {

"text/plain": [

"KNeighborsClassifier(metric='euclidean')"

],

"text/html": [

"<style>#sk-container-id-1 {color: black;background-color: white;}#sk-container-id-1 pre{padding: 0;}#sk-container-id-1 div.sk-toggleable {background-color: white;}#sk-container-id-1 label.sk-toggleable\_\_label {cursor: pointer;display: block;width: 100%;margin-bottom: 0;padding: 0.3em;box-sizing: border-box;text-align: center;}#sk-container-id-1 label.sk-toggleable\_\_label-arrow:before {content: \"▸\";float: left;margin-right: 0.25em;color: #696969;}#sk-container-id-1 label.sk-toggleable\_\_label-arrow:hover:before {color: black;}#sk-container-id-1 div.sk-estimator:hover label.sk-toggleable\_\_label-arrow:before {color: black;}#sk-container-id-1 div.sk-toggleable\_\_content {max-height: 0;max-width: 0;overflow: hidden;text-align: left;background-color: #f0f8ff;}#sk-container-id-1 div.sk-toggleable\_\_content pre {margin: 0.2em;color: black;border-radius: 0.25em;background-color: #f0f8ff;}#sk-container-id-1 input.sk-toggleable\_\_control:checked~div.sk-toggleable\_\_content {max-height: 200px;max-width: 100%;overflow: auto;}#sk-container-id-1 input.sk-toggleable\_\_control:checked~label.sk-toggleable\_\_label-arrow:before {content: \"▾\";}#sk-container-id-1 div.sk-estimator input.sk-toggleable\_\_control:checked~label.sk-toggleable\_\_label {background-color: #d4ebff;}#sk-container-id-1 div.sk-label input.sk-toggleable\_\_control:checked~label.sk-toggleable\_\_label {background-color: #d4ebff;}#sk-container-id-1 input.sk-hidden--visually {border: 0;clip: rect(1px 1px 1px 1px);clip: rect(1px, 1px, 1px, 1px);height: 1px;margin: -1px;overflow: hidden;padding: 0;position: absolute;width: 1px;}#sk-container-id-1 div.sk-estimator {font-family: monospace;background-color: #f0f8ff;border: 1px dotted black;border-radius: 0.25em;box-sizing: border-box;margin-bottom: 0.5em;}#sk-container-id-1 div.sk-estimator:hover {background-color: #d4ebff;}#sk-container-id-1 div.sk-parallel-item::after {content: \"\";width: 100%;border-bottom: 1px solid gray;flex-grow: 1;}#sk-container-id-1 div.sk-label:hover label.sk-toggleable\_\_label {background-color: #d4ebff;}#sk-container-id-1 div.sk-serial::before {content: \"\";position: absolute;border-left: 1px solid gray;box-sizing: border-box;top: 0;bottom: 0;left: 50%;z-index: 0;}#sk-container-id-1 div.sk-serial {display: flex;flex-direction: column;align-items: center;background-color: white;padding-right: 0.2em;padding-left: 0.2em;position: relative;}#sk-container-id-1 div.sk-item {position: relative;z-index: 1;}#sk-container-id-1 div.sk-parallel {display: flex;align-items: stretch;justify-content: center;background-color: white;position: relative;}#sk-container-id-1 div.sk-item::before, #sk-container-id-1 div.sk-parallel-item::before {content: \"\";position: absolute;border-left: 1px solid gray;box-sizing: border-box;top: 0;bottom: 0;left: 50%;z-index: -1;}#sk-container-id-1 div.sk-parallel-item {display: flex;flex-direction: column;z-index: 1;position: relative;background-color: white;}#sk-container-id-1 div.sk-parallel-item:first-child::after {align-self: flex-end;width: 50%;}#sk-container-id-1 div.sk-parallel-item:last-child::after {align-self: flex-start;width: 50%;}#sk-container-id-1 div.sk-parallel-item:only-child::after {width: 0;}#sk-container-id-1 div.sk-dashed-wrapped {border: 1px dashed gray;margin: 0 0.4em 0.5em 0.4em;box-sizing: border-box;padding-bottom: 0.4em;background-color: white;}#sk-container-id-1 div.sk-label label {font-family: monospace;font-weight: bold;display: inline-block;line-height: 1.2em;}#sk-container-id-1 div.sk-label-container {text-align: center;}#sk-container-id-1 div.sk-container {/\* jupyter's `normalize.less` sets `[hidden] { display: none; }` but bootstrap.min.css set `[hidden] { display: none !important; }` so we also need the `!important` here to be able to override the default hidden behavior on the sphinx rendered scikit-learn.org. See: https://github.com/scikit-learn/scikit-learn/issues/21755 \*/display: inline-block !important;position: relative;}#sk-container-id-1 div.sk-text-repr-fallback {display: none;}</style><div id=\"sk-container-id-1\" class=\"sk-top-container\"><div class=\"sk-text-repr-fallback\"><pre>KNeighborsClassifier(metric=&#x27;euclidean&#x27;)</pre><b>In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook. <br />On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.</b></div><div class=\"sk-container\" hidden><div class=\"sk-item\"><div class=\"sk-estimator sk-toggleable\"><input class=\"sk-toggleable\_\_control sk-hidden--visually\" id=\"sk-estimator-id-1\" type=\"checkbox\" checked><label for=\"sk-estimator-id-1\" class=\"sk-toggleable\_\_label sk-toggleable\_\_label-arrow\">KNeighborsClassifier</label><div class=\"sk-toggleable\_\_content\"><pre>KNeighborsClassifier(metric=&#x27;euclidean&#x27;)</pre></div></div></div></div></div>"

]

},

"metadata": {},

"execution\_count": 39

}

]

},

{

"cell\_type": "code",

"source": [

"#probando el modelo con los datos de prueba\n",

"prediccion\_knn = knn.predict(X\_test)"

],

"metadata": {

"id": "2LrZiJbvw-Mf"

},

"execution\_count": 40,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"prediccion\_knn"

],

"metadata": {

"id": "dirCpdIO1MZ9",

"colab": {

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

},

"outputId": "79bab3f0-14c8-4912-c230-e1bd986215bc"

},

"execution\_count": 41,

"outputs": [

{

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

"data": {

"text/plain": [

"array([1, 0, 0, ..., 0, 1, 1])"

]

},

"metadata": {},

"execution\_count": 41

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Aula 3 - Método probabilístico"

],

"metadata": {

"id": "EKHFh1Pr7G40"

}

},

{

"cell\_type": "markdown",

"source": [

"## 3.1 - Teorema de Naive Bayes\n",

"\n",

"(PPT)"

],

"metadata": {

"id": "b\_EAQ1nf7VUu"

}

},

{

"cell\_type": "markdown",

"source": [

"## 3.2 - Modelo Bernoulli Naive Bayes\n",

"\n",

"(PPT)"

],

"metadata": {

"id": "02cUbMWf7dEN"

}

},

{

"cell\_type": "markdown",

"source": [

"## 3.3 - Entrenamiento y prueba"

],

"metadata": {

"id": "zIhV625O7lf9"

}

},

{

"cell\_type": "code",

"source": [

"#biblioteca para crear el modelo de machine learning\n",

"from sklearn.naive\_bayes import BernoulliNB"

],

"metadata": {

"id": "vV4i1JC37xJN"

},

"execution\_count": 42,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#escojo utilizar mediana, porque es el valor central de nuestros datos ordenados\n",

"mediana = np.median(X\_train)\n",

"mediana"

],

"metadata": {

"id": "AEzm1GSZ7tZZ",

"colab": {

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

},

"outputId": "a510df34-4e11-4127-fa72-5af65045126a"

},

"execution\_count": 43,

"outputs": [

{

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

"data": {

"text/plain": [

"-0.4461759755508453"

]

},

"metadata": {},

"execution\_count": 43

}

]

},

{

"cell\_type": "code",

"source": [

"#Binarizando los recursos usando la mediana\n",

"X\_train\_binarizado = np.where(X\_train > mediana, 1, 0)"

],

"metadata": {

"id": "X0HEgH1UXG1l"

},

"execution\_count": 44,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X\_train\_binarizado"

],

"metadata": {

"id": "e5HdbyA27cUJ",

"colab": {

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

},

"outputId": "66c18348-3dd2-4a67-9a3c-c7363ff01499"

},

"execution\_count": 45,

"outputs": [

{

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

"data": {

"text/plain": [

"array([[0, 0, 0, ..., 1, 1, 1],\n",

" [1, 1, 1, ..., 1, 1, 1],\n",

" [0, 0, 1, ..., 0, 1, 1],\n",

" ...,\n",

" [1, 1, 1, ..., 1, 1, 1],\n",

" [1, 1, 1, ..., 0, 1, 1],\n",

" [0, 0, 1, ..., 0, 1, 1]])"

]

},

"metadata": {},

"execution\_count": 45

}

]

},

{

"cell\_type": "code",

"source": [

"y\_train"

],

"metadata": {

"id": "HevdnXNV7rPg",

"colab": {

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

},

"outputId": "4e292eda-fc6a-4995-f434-599e6d6f23d0"

},

"execution\_count": 46,

"outputs": [

{

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

"data": {

"text/plain": [

"4831 0\n",

"5077 0\n",

"9023 1\n",

"4424 1\n",

"5236 0\n",

" ..\n",

"9785 1\n",

"7763 1\n",

"5218 0\n",

"1346 1\n",

"3582 0\n",

"Name: Churn, Length: 7243, dtype: int64"

]

},

"metadata": {},

"execution\_count": 46

}

]

},

{

"cell\_type": "code",

"source": [

"#creamos el modelo\n",

"bnb = BernoulliNB()"

],

"metadata": {

"id": "KnTnDve07ylb"

},

"execution\_count": 47,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#entrenando el modelo\n",

"bnb.fit(X\_train\_binarizado, y\_train)"

],

"metadata": {

"id": "qIIT0uX273SN",

"colab": {

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

"height": 75

},

"outputId": "9649f65b-8f6b-448c-c756-98bedd5d2563"

},

"execution\_count": 48,

"outputs": [

{

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

"data": {

"text/plain": [

"BernoulliNB()"

],

"text/html": [

"<style>#sk-container-id-2 {color: black;background-color: white;}#sk-container-id-2 pre{padding: 0;}#sk-container-id-2 div.sk-toggleable {background-color: white;}#sk-container-id-2 label.sk-toggleable\_\_label {cursor: pointer;display: block;width: 100%;margin-bottom: 0;padding: 0.3em;box-sizing: border-box;text-align: center;}#sk-container-id-2 label.sk-toggleable\_\_label-arrow:before {content: \"▸\";float: left;margin-right: 0.25em;color: #696969;}#sk-container-id-2 label.sk-toggleable\_\_label-arrow:hover:before {color: black;}#sk-container-id-2 div.sk-estimator:hover label.sk-toggleable\_\_label-arrow:before {color: black;}#sk-container-id-2 div.sk-toggleable\_\_content {max-height: 0;max-width: 0;overflow: hidden;text-align: left;background-color: #f0f8ff;}#sk-container-id-2 div.sk-toggleable\_\_content pre {margin: 0.2em;color: black;border-radius: 0.25em;background-color: #f0f8ff;}#sk-container-id-2 input.sk-toggleable\_\_control:checked~div.sk-toggleable\_\_content {max-height: 200px;max-width: 100%;overflow: auto;}#sk-container-id-2 input.sk-toggleable\_\_control:checked~label.sk-toggleable\_\_label-arrow:before {content: \"▾\";}#sk-container-id-2 div.sk-estimator input.sk-toggleable\_\_control:checked~label.sk-toggleable\_\_label {background-color: #d4ebff;}#sk-container-id-2 div.sk-label input.sk-toggleable\_\_control:checked~label.sk-toggleable\_\_label {background-color: #d4ebff;}#sk-container-id-2 input.sk-hidden--visually {border: 0;clip: rect(1px 1px 1px 1px);clip: rect(1px, 1px, 1px, 1px);height: 1px;margin: -1px;overflow: hidden;padding: 0;position: absolute;width: 1px;}#sk-container-id-2 div.sk-estimator {font-family: monospace;background-color: #f0f8ff;border: 1px dotted black;border-radius: 0.25em;box-sizing: border-box;margin-bottom: 0.5em;}#sk-container-id-2 div.sk-estimator:hover {background-color: #d4ebff;}#sk-container-id-2 div.sk-parallel-item::after {content: \"\";width: 100%;border-bottom: 1px solid gray;flex-grow: 1;}#sk-container-id-2 div.sk-label:hover label.sk-toggleable\_\_label {background-color: #d4ebff;}#sk-container-id-2 div.sk-serial::before {content: \"\";position: absolute;border-left: 1px solid gray;box-sizing: border-box;top: 0;bottom: 0;left: 50%;z-index: 0;}#sk-container-id-2 div.sk-serial {display: flex;flex-direction: column;align-items: center;background-color: white;padding-right: 0.2em;padding-left: 0.2em;position: relative;}#sk-container-id-2 div.sk-item {position: relative;z-index: 1;}#sk-container-id-2 div.sk-parallel {display: flex;align-items: stretch;justify-content: center;background-color: white;position: relative;}#sk-container-id-2 div.sk-item::before, #sk-container-id-2 div.sk-parallel-item::before {content: \"\";position: absolute;border-left: 1px solid gray;box-sizing: border-box;top: 0;bottom: 0;left: 50%;z-index: -1;}#sk-container-id-2 div.sk-parallel-item {display: flex;flex-direction: column;z-index: 1;position: relative;background-color: white;}#sk-container-id-2 div.sk-parallel-item:first-child::after {align-self: flex-end;width: 50%;}#sk-container-id-2 div.sk-parallel-item:last-child::after {align-self: flex-start;width: 50%;}#sk-container-id-2 div.sk-parallel-item:only-child::after {width: 0;}#sk-container-id-2 div.sk-dashed-wrapped {border: 1px dashed gray;margin: 0 0.4em 0.5em 0.4em;box-sizing: border-box;padding-bottom: 0.4em;background-color: white;}#sk-container-id-2 div.sk-label label {font-family: monospace;font-weight: bold;display: inline-block;line-height: 1.2em;}#sk-container-id-2 div.sk-label-container {text-align: center;}#sk-container-id-2 div.sk-container {/\* jupyter's `normalize.less` sets `[hidden] { display: none; }` but bootstrap.min.css set `[hidden] { display: none !important; }` so we also need the `!important` here to be able to override the default hidden behavior on the sphinx rendered scikit-learn.org. See: https://github.com/scikit-learn/scikit-learn/issues/21755 \*/display: inline-block !important;position: relative;}#sk-container-id-2 div.sk-text-repr-fallback {display: none;}</style><div id=\"sk-container-id-2\" class=\"sk-top-container\"><div class=\"sk-text-repr-fallback\"><pre>BernoulliNB()</pre><b>In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook. <br />On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.</b></div><div class=\"sk-container\" hidden><div class=\"sk-item\"><div class=\"sk-estimator sk-toggleable\"><input class=\"sk-toggleable\_\_control sk-hidden--visually\" id=\"sk-estimator-id-2\" type=\"checkbox\" checked><label for=\"sk-estimator-id-2\" class=\"sk-toggleable\_\_label sk-toggleable\_\_label-arrow\">BernoulliNB</label><div class=\"sk-toggleable\_\_content\"><pre>BernoulliNB()</pre></div></div></div></div></div>"

]

},

"metadata": {},

"execution\_count": 48

}

]

},

{

"cell\_type": "code",

"source": [

"#Binarizando la base de prueba\n",

"X\_test\_binarizado = np.where(X\_test > np.median(X\_test), 1, 0)"

],

"metadata": {

"id": "2v7NF1BipX6w"

},

"execution\_count": 49,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#probando el modelo\n",

"prediccion\_BNb = bnb.predict(X\_test\_binarizado)"

],

"metadata": {

"id": "BpCOWhx674G6"

},

"execution\_count": 50,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"prediccion\_BNb"

],

"metadata": {

"id": "KIS6Syvf79ou",

"colab": {

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

},

"outputId": "c3c22e9b-a571-4ea3-c650-f259f39f7046"

},

"execution\_count": 51,

"outputs": [

{

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

"data": {

"text/plain": [

"array([1, 0, 0, ..., 1, 1, 1])"

]

},

"metadata": {},

"execution\_count": 51

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Aula 4 - Método Simbólico"

],

"metadata": {

"id": "I0QZ6\_rY00NG"

}

},

{

"cell\_type": "markdown",

"source": [

"## 4.1 - ¿Qué es un árbol de decisión?\n",

"\n",

"(PPT)"

],

"metadata": {

"id": "VglEIUuC2jkO"

}

},

{

"cell\_type": "markdown",

"source": [

"## 4.2 - ¿Cómo funciona un árbol de decisión?\n",

"\n",

"(PPT)"

],

"metadata": {

"id": "jdtLgDk3sAQZ"

}

},

{

"cell\_type": "markdown",

"source": [

"## 4.3 - Implementando el modelo"

],

"metadata": {

"id": "cYFuWWnTsJLP"

}

},

{

"cell\_type": "code",

"source": [

"#biblioteca para crear el modelo de machine learning\n",

"from sklearn.tree import DecisionTreeClassifier"

],

"metadata": {

"id": "VZBVqKxE2i10"

},

"execution\_count": 52,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#iniciando el modelo\n",

"dtc = DecisionTreeClassifier(criterion='entropy', random\_state=42)"

],

"metadata": {

"id": "AR0fPU6qsPmg"

},

"execution\_count": 53,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#entrenando el modelo\n",

"dtc.fit(X\_train, y\_train)"

],

"metadata": {

"id": "zG5vDatzsV7D",

"colab": {

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

"height": 75

},

"outputId": "b64ccccd-81ff-48e2-c1d8-2521c3aa3861"

},

"execution\_count": 54,

"outputs": [

{

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

"data": {

"text/plain": [

"DecisionTreeClassifier(criterion='entropy', random\_state=42)"

],

"text/html": [

"<style>#sk-container-id-3 {color: black;background-color: white;}#sk-container-id-3 pre{padding: 0;}#sk-container-id-3 div.sk-toggleable {background-color: white;}#sk-container-id-3 label.sk-toggleable\_\_label {cursor: pointer;display: block;width: 100%;margin-bottom: 0;padding: 0.3em;box-sizing: border-box;text-align: center;}#sk-container-id-3 label.sk-toggleable\_\_label-arrow:before {content: \"▸\";float: left;margin-right: 0.25em;color: #696969;}#sk-container-id-3 label.sk-toggleable\_\_label-arrow:hover:before {color: black;}#sk-container-id-3 div.sk-estimator:hover label.sk-toggleable\_\_label-arrow:before {color: black;}#sk-container-id-3 div.sk-toggleable\_\_content {max-height: 0;max-width: 0;overflow: hidden;text-align: left;background-color: #f0f8ff;}#sk-container-id-3 div.sk-toggleable\_\_content pre {margin: 0.2em;color: black;border-radius: 0.25em;background-color: #f0f8ff;}#sk-container-id-3 input.sk-toggleable\_\_control:checked~div.sk-toggleable\_\_content {max-height: 200px;max-width: 100%;overflow: auto;}#sk-container-id-3 input.sk-toggleable\_\_control:checked~label.sk-toggleable\_\_label-arrow:before {content: \"▾\";}#sk-container-id-3 div.sk-estimator input.sk-toggleable\_\_control:checked~label.sk-toggleable\_\_label {background-color: #d4ebff;}#sk-container-id-3 div.sk-label input.sk-toggleable\_\_control:checked~label.sk-toggleable\_\_label {background-color: #d4ebff;}#sk-container-id-3 input.sk-hidden--visually {border: 0;clip: rect(1px 1px 1px 1px);clip: rect(1px, 1px, 1px, 1px);height: 1px;margin: -1px;overflow: hidden;padding: 0;position: absolute;width: 1px;}#sk-container-id-3 div.sk-estimator {font-family: monospace;background-color: #f0f8ff;border: 1px dotted black;border-radius: 0.25em;box-sizing: border-box;margin-bottom: 0.5em;}#sk-container-id-3 div.sk-estimator:hover {background-color: #d4ebff;}#sk-container-id-3 div.sk-parallel-item::after {content: \"\";width: 100%;border-bottom: 1px solid gray;flex-grow: 1;}#sk-container-id-3 div.sk-label:hover label.sk-toggleable\_\_label {background-color: #d4ebff;}#sk-container-id-3 div.sk-serial::before {content: \"\";position: absolute;border-left: 1px solid gray;box-sizing: border-box;top: 0;bottom: 0;left: 50%;z-index: 0;}#sk-container-id-3 div.sk-serial {display: flex;flex-direction: column;align-items: center;background-color: white;padding-right: 0.2em;padding-left: 0.2em;position: relative;}#sk-container-id-3 div.sk-item {position: relative;z-index: 1;}#sk-container-id-3 div.sk-parallel {display: flex;align-items: stretch;justify-content: center;background-color: white;position: relative;}#sk-container-id-3 div.sk-item::before, #sk-container-id-3 div.sk-parallel-item::before {content: \"\";position: absolute;border-left: 1px solid gray;box-sizing: border-box;top: 0;bottom: 0;left: 50%;z-index: -1;}#sk-container-id-3 div.sk-parallel-item {display: flex;flex-direction: column;z-index: 1;position: relative;background-color: white;}#sk-container-id-3 div.sk-parallel-item:first-child::after {align-self: flex-end;width: 50%;}#sk-container-id-3 div.sk-parallel-item:last-child::after {align-self: flex-start;width: 50%;}#sk-container-id-3 div.sk-parallel-item:only-child::after {width: 0;}#sk-container-id-3 div.sk-dashed-wrapped {border: 1px dashed gray;margin: 0 0.4em 0.5em 0.4em;box-sizing: border-box;padding-bottom: 0.4em;background-color: white;}#sk-container-id-3 div.sk-label label {font-family: monospace;font-weight: bold;display: inline-block;line-height: 1.2em;}#sk-container-id-3 div.sk-label-container {text-align: center;}#sk-container-id-3 div.sk-container {/\* jupyter's `normalize.less` sets `[hidden] { display: none; }` but bootstrap.min.css set `[hidden] { display: none !important; }` so we also need the `!important` here to be able to override the default hidden behavior on the sphinx rendered scikit-learn.org. See: https://github.com/scikit-learn/scikit-learn/issues/21755 \*/display: inline-block !important;position: relative;}#sk-container-id-3 div.sk-text-repr-fallback {display: none;}</style><div id=\"sk-container-id-3\" class=\"sk-top-container\"><div class=\"sk-text-repr-fallback\"><pre>DecisionTreeClassifier(criterion=&#x27;entropy&#x27;, random\_state=42)</pre><b>In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook. <br />On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.</b></div><div class=\"sk-container\" hidden><div class=\"sk-item\"><div class=\"sk-estimator sk-toggleable\"><input class=\"sk-toggleable\_\_control sk-hidden--visually\" id=\"sk-estimator-id-3\" type=\"checkbox\" checked><label for=\"sk-estimator-id-3\" class=\"sk-toggleable\_\_label sk-toggleable\_\_label-arrow\">DecisionTreeClassifier</label><div class=\"sk-toggleable\_\_content\"><pre>DecisionTreeClassifier(criterion=&#x27;entropy&#x27;, random\_state=42)</pre></div></div></div></div></div>"

]

},

"metadata": {},

"execution\_count": 54

}

]

},

{

"cell\_type": "code",

"source": [

"#verificando la importancia de cada atributo\n",

"dtc.feature\_importances\_"

],

"metadata": {

"id": "Gx2S564TiNqI",

"colab": {

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

},

"outputId": "8255b69f-3492-4524-b131-fbff6fc9693d"

},

"execution\_count": 55,

"outputs": [

{

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

"data": {

"text/plain": [

"array([0.02243428, 0.01223771, 0.00575433, 0.01818431, 0.01693893,\n",

" 0.16491182, 0.22384714, 0.01029507, 0.00861237, 0.00429131,\n",

" 0.0019815 , 0.03128648, 0. , 0.01153661, 0.01516212,\n",

" 0. , 0.00378651, 0.01208236, 0. , 0.01143488,\n",

" 0.00586885, 0. , 0.01332184, 0.02079021, 0.00027769,\n",

" 0.00298793, 0.00351301, 0.00433452, 0.00745511, 0.01266344,\n",

" 0. , 0.03622159, 0.16358635, 0.07445054, 0.02315574,\n",

" 0.02006142, 0.01865312, 0.01788091])"

]

},

"metadata": {},

"execution\_count": 55

}

]

},

{

"cell\_type": "code",

"source": [

"prediccion\_ArbolDecision = dtc.predict(X\_test)"

],

"metadata": {

"id": "1N3ogMwXsYLJ"

},

"execution\_count": 56,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"prediccion\_ArbolDecision"

],

"metadata": {

"id": "8yWcv1rCsbIT",

"colab": {

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

},

"outputId": "be130559-3fa4-4e6e-e421-a4f9f959db93"

},

"execution\_count": 57,

"outputs": [

{

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

"data": {

"text/plain": [

"array([0, 0, 0, ..., 1, 1, 1])"

]

},

"metadata": {},

"execution\_count": 57

}

]

},

{

"cell\_type": "markdown",

"source": [

"# 5 - Validación de los modelos"

],

"metadata": {

"id": "lfmj\_l6ztF5J"

}

},

{

"cell\_type": "markdown",

"source": [

"## 5.1 - Matriz de confusión\n",

"\n"

],

"metadata": {

"id": "5LcpGgohtOM5"

}

},

{

"cell\_type": "markdown",

"source": [

"![image.png](https://raw.githubusercontent.com/ElProfeAlejo/machine\_learning\_clasificacion/main/imagen\_5\_1.png)"

],

"metadata": {

"id": "BbHrkrAnxhmv"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.metrics import confusion\_matrix"

],

"metadata": {

"id": "dIhkghN1tM3d"

},

"execution\_count": 58,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"print(confusion\_matrix(y\_test, prediccion\_knn))"

],

"metadata": {

"id": "vy5vfPQ7mnmP",

"colab": {

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

},

"outputId": "cea311b0-92f9-4e10-e095-400e15e99a81"

},

"execution\_count": 59,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"[[1242 327]\n",

" [ 247 1289]]\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"print(confusion\_matrix(y\_test, prediccion\_BNb))"

],

"metadata": {

"id": "Y\_TkgT9pmnj3",

"colab": {

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

},

"outputId": "7ecd0df9-ff40-4125-fd2f-e31f5bb84687"

},

"execution\_count": 60,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"[[1048 521]\n",

" [ 241 1295]]\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"print(confusion\_matrix(y\_test, prediccion\_ArbolDecision))"

],

"metadata": {

"id": "hNAbk\_Pimnhw",

"colab": {

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

},

"outputId": "6237d7b6-4996-4d70-aa44-150248bfd756"

},

"execution\_count": 61,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"[[1242 327]\n",

" [ 271 1265]]\n"

]

}

]

},

{

"cell\_type": "markdown",

"source": [

"## 5.2 - Accuracy\n",

"A partir del cálculo de la matriz de confusión, podemos inferir otras métricas, como el accuracy.\n",

"\n",

"\n",

"\n",

"\n",

"$ACC$ = ${TP + TN \\over TP + FP + TN + FN}$"

],

"metadata": {

"id": "sQb7vnC12XUf"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.metrics import accuracy\_score"

],

"metadata": {

"id": "Ng67mpWk2af1"

},

"execution\_count": 62,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#modelo KNN\n",

"print(accuracy\_score(y\_test, prediccion\_knn))"

],

"metadata": {

"id": "gp4Purjf2kZe",

"colab": {

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

},

"outputId": "be7be43e-8b3e-4af6-f071-1ccfc34894ba"

},

"execution\_count": 63,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"0.8151368760064412\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"#modelo Bernoulli de Naive Bayes\n",

"print(accuracy\_score(y\_test, prediccion\_BNb))"

],

"metadata": {

"id": "7JrJXLJr2loH",

"colab": {

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

},

"outputId": "f13c3654-2949-4b1a-e791-d86806342b2f"

},

"execution\_count": 64,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"0.7545893719806763\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"#modelo Arbol de Decisión\n",

"print(accuracy\_score(y\_test, prediccion\_ArbolDecision))"

],

"metadata": {

"id": "BkvmkoN92nBX",

"colab": {

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

},

"outputId": "0bb59c4d-298f-4820-b154-45350b1a8a4d"

},

"execution\_count": 65,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"0.8074074074074075\n"

]

}

]

},

{

"cell\_type": "markdown",

"source": [

"## 5.3 - Precisión\n",

"\n",

"Otra métrica importante es la precisión, que calcula cuántos se clasificaron correctamente como positivos ($TP$).\n",

"\n",

"$PS$ = ${TP \\over TP + FP}$"

],

"metadata": {

"id": "kfzcKhw42tmX"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.metrics import precision\_score"

],

"metadata": {

"id": "BM4AbIm62wP6"

},

"execution\_count": 66,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#modelo KNN\n",

"print(precision\_score(y\_test, prediccion\_knn))"

],

"metadata": {

"id": "hBD4K9r523g4",

"colab": {

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

},

"outputId": "b5421b0b-5a2f-4806-b965-c44bee355612"

},

"execution\_count": 67,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"0.7976485148514851\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"#modelo Bernoulli de Naive Bayes\n",

"print(precision\_score(y\_test, prediccion\_BNb))"

],

"metadata": {

"id": "mUJumEYp24rf",

"colab": {

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

},

"outputId": "7f650e9a-3370-45cd-e9d7-325a37200fe3"

},

"execution\_count": 68,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"0.7131057268722467\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"#modelo Arbol de Decisión\n",

"print(precision\_score(y\_test, prediccion\_ArbolDecision))"

],

"metadata": {

"id": "ST2dcOPJ25IS",

"colab": {

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

},

"outputId": "5153b4ee-b4df-410c-edd1-cb4ee52dc1dc"

},

"execution\_count": 69,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"0.7945979899497487\n"

]

}

]

},

{

"cell\_type": "markdown",

"source": [

"## 5.4 - Recall\n",

"\n",

"Otra métrica es el Recall o sensibilidad, calcula qué tan bueno es el modelo para clasificar correctamente un resultado positivo ($TP$).\n",

"\n",

"$RC$ = ${TP \\over TP + FN}$"

],

"metadata": {

"id": "TW59AoyU3DvW"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.metrics import recall\_score"

],

"metadata": {

"id": "SGX6E4Pi3LCt"

},

"execution\_count": 70,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#modelo KNN\n",

"print(recall\_score(y\_test, prediccion\_knn))"

],

"metadata": {

"id": "CVEe2W\_B3OTc",

"colab": {

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

},

"outputId": "6852e4a0-c50b-482d-d481-46e54068b072"

},

"execution\_count": 71,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"0.8391927083333334\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"#modelo Bernoulli de Naive Bayes\n",

"print(recall\_score(y\_test, prediccion\_BNb))"

],

"metadata": {

"id": "edPMCLA93Pw\_",

"colab": {

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

},

"outputId": "39479d4d-738d-4bdd-d203-3b90eb81bc70"

},

"execution\_count": 72,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"0.8430989583333334\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"#modelo Arbol de Decisión\n",

"print(recall\_score(y\_test, prediccion\_ArbolDecision))"

],

"metadata": {

"id": "4Wk5-4ej3Q\_3",

"colab": {

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

},

"outputId": "7a980ec6-1991-45b2-b8ba-c9d4771ea9e1"

},

"execution\_count": 73,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"0.8235677083333334\n"

]

}

]

},

{

"cell\_type": "markdown",

"source": [

"## 5.5 - Escogiendo el mejor modelo\n"

],

"metadata": {

"id": "4O7N26Fg3XwI"

}

},

{

"cell\_type": "code",

"source": [

"#Ejemplo - análisis de las precisiones previamente calculadas\n",

"print('Modelo KNN: ', precision\_score(y\_test, prediccion\_knn))\n",

"print('Modelo Bernoulli de Naive Bayes: ', precision\_score(y\_test, prediccion\_BNb))\n",

"print('Modelo Arbol de Decisión: ', precision\_score(y\_test, prediccion\_ArbolDecision))"

],

"metadata": {

"id": "RJ5Ls9lx3pfI",

"colab": {

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

},

"outputId": "67a59410-1b0d-47ef-a56c-60749db5f122"

},

"execution\_count": 74,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"Modelo KNN: 0.7976485148514851\n",

"Modelo Bernoulli de Naive Bayes: 0.7131057268722467\n",

"Modelo Arbol de Decisión: 0.7945979899497487\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"#Probando el mejor modelo para Maria\n",

"prediccion\_maria = knn.predict(Xmaria\_normalizado)\n",

"diccionario = {'Si': 1, 'No': 0}\n",

"\n",

"clave\_encontrada = next((clave for clave, valor in diccionario.items() if valor == prediccion\_maria[0]), None)\n",

"print(f\"La probabilidad de que Maria se convierta en Churn es: {clave\_encontrada}\")"

],

"metadata": {

"id": "BmhZXFOFganM",

"colab": {

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

},

"outputId": "c852d425-aaf3-4b68-b279-025c3dd9f764"

},

"execution\_count": 75,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"La probabilidad de que Maria se convierta en Churn es: No\n"

]

}

]

}

],

"metadata": {

"interpreter": {

"hash": "df87b94f50c6b8081a9a425f6ade537f06441a824b7d68371eb29c5a5eb2838b"

},

"kernelspec": {

"display\_name": "Python 3.10.4 64-bit",

"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": "3.10.4"

},

"orig\_nbformat": 4,

"colab": {

"provenance": [],

"collapsed\_sections": [

"uOxsWL4fyQ2J",

"\_d4jEXubcPuO",

"EKHFh1Pr7G40",

"I0QZ6\_rY00NG",

"lfmj\_l6ztF5J"

]

}

},

"nbformat": 4,

"nbformat\_minor": 0

}cla