{

"nbformat": 4,

"nbformat\_minor": 0,

"metadata": {

"colab": {

"provenance": []

},

"kernelspec": {

"name": "python3",

"display\_name": "Python 3"

},

"language\_info": {

"name": "python"

}

},

"cells": [

{

"cell\_type": "code",

"source": [

"# Import necessary libraries\n",

"import pandas as pd\n",

"import numpy as np\n",

"import seaborn as sns\n",

"import matplotlib.pyplot as plt\n",

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

"from sklearn.preprocessing import StandardScaler\n",

"from sklearn.ensemble import RandomForestClassifier\n",

"from sklearn.metrics import classification\_report, confusion\_matrix\n",

"from imblearn.over\_sampling import SMOTE\n",

"from scipy import stats"

],

"metadata": {

"id": "2kdC8Earf8WI"

},

"execution\_count": 32,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"# Load the dataset\n",

"data = pd.read\_csv('/content/datacamp\_workspace\_export\_2024-04-01 19\_32\_28.csv')"

],

"metadata": {

"id": "znL31VjFg6M4"

},

"execution\_count": 34,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"# Identify patterns in the data\n",

"sns.pairplot(data, hue='not.fully.paid')\n",

"plt.show()"

],

"metadata": {

"colab": {

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

"height": 1000

},

"id": "bXiyXsZxgxiN",

"outputId": "d1282062-c23a-4a45-bde0-e0168705c20b"

},

"execution\_count": 36,

"outputs": [

{

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

"data": {

"text/plain": [

"<Figure size 3100.61x3000 with 156 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"cell\_type": "code",

"execution\_count": null,

"metadata": {

"id": "KiwaNAjx3zpe"

},

"outputs": [],

"source": [

"import numpy as np\n",

"import pandas as pd"

]

},

{

"cell\_type": "code",

"source": [

"%matplotlib inline"

],

"metadata": {

"id": "RWKEnZNi4T2Y"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"loans = pd.read\_csv('/content/datacamp\_workspace\_export\_2024-04-01 19\_32\_28.csv')"

],

"metadata": {

"id": "KTHxkuHL4VBZ"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"loans.head()"

],

"metadata": {

"colab": {

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

"height": 226

},

"id": "9AM52kr94t9A",

"outputId": "7fa166d4-fb58-44f9-9cb4-e7e1da844f26"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

" credit.policy purpose int.rate installment log.annual.inc \\\n",

"0 1 debt\_consolidation 0.1189 829.10 11.350407 \n",

"1 1 credit\_card 0.1071 228.22 11.082143 \n",

"2 1 debt\_consolidation 0.1357 366.86 10.373491 \n",

"3 1 debt\_consolidation 0.1008 162.34 11.350407 \n",

"4 1 credit\_card 0.1426 102.92 11.299732 \n",

"\n",

" dti fico days.with.cr.line revol.bal revol.util inq.last.6mths \\\n",

"0 19.48 737 5639.958333 28854 52.1 0 \n",

"1 14.29 707 2760.000000 33623 76.7 0 \n",

"2 11.63 682 4710.000000 3511 25.6 1 \n",

"3 8.10 712 2699.958333 33667 73.2 1 \n",

"4 14.97 667 4066.000000 4740 39.5 0 \n",

"\n",

" delinq.2yrs pub.rec not.fully.paid \n",

"0 0 0 0 \n",

"1 0 0 0 \n",

"2 0 0 0 \n",

"3 0 0 0 \n",

"4 1 0 0 "

],

"text/html": [

"\n",

" <div id=\"df-f4bb167e-01bb-4481-81aa-1db95237d71c\" 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>credit.policy</th>\n",

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

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

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

" <th>log.annual.inc</th>\n",

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

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

" <th>days.with.cr.line</th>\n",

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

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

" <th>inq.last.6mths</th>\n",

" <th>delinq.2yrs</th>\n",

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

" <th>not.fully.paid</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

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

" <td>debt\_consolidation</td>\n",

" <td>0.1189</td>\n",

" <td>829.10</td>\n",

" <td>11.350407</td>\n",

" <td>19.48</td>\n",

" <td>737</td>\n",

" <td>5639.958333</td>\n",

" <td>28854</td>\n",

" <td>52.1</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>1</td>\n",

" <td>credit\_card</td>\n",

" <td>0.1071</td>\n",

" <td>228.22</td>\n",

" <td>11.082143</td>\n",

" <td>14.29</td>\n",

" <td>707</td>\n",

" <td>2760.000000</td>\n",

" <td>33623</td>\n",

" <td>76.7</td>\n",

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

" <td>debt\_consolidation</td>\n",

" <td>0.1357</td>\n",

" <td>366.86</td>\n",

" <td>10.373491</td>\n",

" <td>11.63</td>\n",

" <td>682</td>\n",

" <td>4710.000000</td>\n",

" <td>3511</td>\n",

" <td>25.6</td>\n",

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

" <td>debt\_consolidation</td>\n",

" <td>0.1008</td>\n",

" <td>162.34</td>\n",

" <td>11.350407</td>\n",

" <td>8.10</td>\n",

" <td>712</td>\n",

" <td>2699.958333</td>\n",

" <td>33667</td>\n",

" <td>73.2</td>\n",

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

" <td>credit\_card</td>\n",

" <td>0.1426</td>\n",

" <td>102.92</td>\n",

" <td>11.299732</td>\n",

" <td>14.97</td>\n",

" <td>667</td>\n",

" <td>4066.000000</td>\n",

" <td>4740</td>\n",

" <td>39.5</td>\n",

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

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

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

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

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

"\n",

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

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-f4bb167e-01bb-4481-81aa-1db95237d71c')\"\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 -960 960 960\">\n",

" <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220 220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",

" </svg>\n",

" </button>\n",

"\n",

" <style>\n",

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

" display:flex;\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",

" .colab-df-buttons div {\n",

" margin-bottom: 4px;\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-f4bb167e-01bb-4481-81aa-1db95237d71c 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-f4bb167e-01bb-4481-81aa-1db95237d71c');\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",

"\n",

"\n",

"<div id=\"df-4e6c86b9-df7b-461a-a532-483359501f52\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-4e6c86b9-df7b-461a-a532-483359501f52')\"\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",

"\n",

"<style>\n",

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

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

" --fill-color: #1967D2;\n",

" --hover-bg-color: #E2EBFA;\n",

" --hover-fill-color: #174EA6;\n",

" --disabled-fill-color: #AAA;\n",

" --disabled-bg-color: #DDD;\n",

" }\n",

"\n",

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

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

" --fill-color: #D2E3FC;\n",

" --hover-bg-color: #434B5C;\n",

" --hover-fill-color: #FFFFFF;\n",

" --disabled-bg-color: #3B4455;\n",

" --disabled-fill-color: #666;\n",

" }\n",

"\n",

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

" background-color: var(--bg-color);\n",

" border: none;\n",

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

" cursor: pointer;\n",

" display: none;\n",

" fill: var(--fill-color);\n",

" height: 32px;\n",

" padding: 0;\n",

" width: 32px;\n",

" }\n",

"\n",

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

" background-color: var(--hover-bg-color);\n",

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

" fill: var(--button-hover-fill-color);\n",

" }\n",

"\n",

" .colab-df-quickchart-complete:disabled,\n",

" .colab-df-quickchart-complete:disabled:hover {\n",

" background-color: var(--disabled-bg-color);\n",

" fill: var(--disabled-fill-color);\n",

" box-shadow: none;\n",

" }\n",

"\n",

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

" border: 2px solid var(--fill-color);\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" animation:\n",

" spin 1s steps(1) infinite;\n",

" }\n",

"\n",

" @keyframes spin {\n",

" 0% {\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" border-left-color: var(--fill-color);\n",

" }\n",

" 20% {\n",

" border-color: transparent;\n",

" border-left-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" }\n",

" 30% {\n",

" border-color: transparent;\n",

" border-left-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" border-right-color: var(--fill-color);\n",

" }\n",

" 40% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" }\n",

" 60% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" }\n",

" 80% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" border-bottom-color: var(--fill-color);\n",

" }\n",

" 90% {\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" }\n",

" }\n",

"</style>\n",

"\n",

" <script>\n",

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

" const quickchartButtonEl =\n",

" document.querySelector('#' + key + ' button');\n",

" quickchartButtonEl.disabled = true; // To prevent multiple clicks.\n",

" quickchartButtonEl.classList.add('colab-df-spinner');\n",

" try {\n",

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

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

" } catch (error) {\n",

" console.error('Error during call to suggestCharts:', error);\n",

" }\n",

" quickchartButtonEl.classList.remove('colab-df-spinner');\n",

" quickchartButtonEl.classList.add('colab-df-quickchart-complete');\n",

" }\n",

" (() => {\n",

" let quickchartButtonEl =\n",

" document.querySelector('#df-4e6c86b9-df7b-461a-a532-483359501f52 button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

"\n",

" </div>\n",

" </div>\n"

],

"application/vnd.google.colaboratory.intrinsic+json": {

"type": "dataframe",

"variable\_name": "loans",

"summary": "{\n \"name\": \"loans\",\n \"rows\": 100,\n \"fields\": [\n {\n \"column\": \"credit.policy\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 1,\n \"max\": 1,\n \"num\_unique\_values\": 1,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"purpose\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 7,\n \"samples\": [\n \"debt\_consolidation\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"int.rate\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.023150181970181687,\n \"min\": 0.0714,\n \"max\": 0.1596,\n \"num\_unique\_values\": 46,\n \"samples\": [\n 0.0964\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"installment\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 162.3676786764776,\n \"min\": 32.55,\n \"max\": 829.1,\n \"num\_unique\_values\": 83,\n \"samples\": [\n 78.42\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"log.annual.inc\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.7157951261195352,\n \"min\": 8.987196821,\n \"max\": 12.4292162,\n \"num\_unique\_values\": 59,\n \"samples\": [\n 11.35040654\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"dti\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 5.981154494820301,\n \"min\": 0.0,\n \"max\": 22.09,\n \"num\_unique\_values\": 91,\n \"samples\": [\n 18.64\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"fico\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 44,\n \"min\": 627,\n \"max\": 812,\n \"num\_unique\_values\": 29,\n \"samples\": [\n 627\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"days.with.cr.line\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 2630.856598362081,\n \"min\": 1110.0,\n \"max\": 14008.95833,\n \"num\_unique\_values\": 91,\n \"samples\": [\n 2460.0\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"revol.bal\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 22933,\n \"min\": 0,\n \"max\": 128000,\n \"num\_unique\_values\": 96,\n \"samples\": [\n 8155\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"revol.util\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 26.17236957734109,\n \"min\": 0.0,\n \"max\": 93.4,\n \"num\_unique\_values\": 93,\n \"samples\": [\n 16.9\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"inq.last.6mths\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 1,\n \"min\": 0,\n \"max\": 4,\n \"num\_unique\_values\": 5,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"delinq.2yrs\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 2,\n \"num\_unique\_values\": 3,\n \"samples\": [\n 0\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"pub.rec\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 1,\n \"num\_unique\_values\": 2,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"not.fully.paid\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 1,\n \"num\_unique\_values\": 2,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}"

}

},

"metadata": {},

"execution\_count": 4

}

]

},

{

"cell\_type": "code",

"source": [

"loans.head().info()"

],

"metadata": {

"colab": {

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

},

"id": "wx7npRTe4u9m",

"outputId": "7a8666de-8ea4-4150-b7f5-87165d1e8c83"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

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

"RangeIndex: 5 entries, 0 to 4\n",

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

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

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

" 0 credit.policy 5 non-null int64 \n",

" 1 purpose 5 non-null object \n",

" 2 int.rate 5 non-null float64\n",

" 3 installment 5 non-null float64\n",

" 4 log.annual.inc 5 non-null float64\n",

" 5 dti 5 non-null float64\n",

" 6 fico 5 non-null int64 \n",

" 7 days.with.cr.line 5 non-null float64\n",

" 8 revol.bal 5 non-null int64 \n",

" 9 revol.util 5 non-null float64\n",

" 10 inq.last.6mths 5 non-null int64 \n",

" 11 delinq.2yrs 5 non-null int64 \n",

" 12 pub.rec 5 non-null int64 \n",

" 13 not.fully.paid 5 non-null int64 \n",

"dtypes: float64(6), int64(7), object(1)\n",

"memory usage: 688.0+ bytes\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"loans.head().shape"

],

"metadata": {

"colab": {

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

},

"id": "NjXaXtqR4zBi",

"outputId": "21557ec7-f53e-4916-f3c1-0fd00902a50c"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"(5, 14)"

]

},

"metadata": {},

"execution\_count": 6

}

]

},

{

"cell\_type": "code",

"source": [

"loans.head().describe()"

],

"metadata": {

"colab": {

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

"height": 320

},

"id": "fXi4fbb142F5",

"outputId": "9e7cceba-9b74-472b-af14-b629a75c1e4a"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

" credit.policy int.rate installment log.annual.inc dti \\\n",

"count 5.0 5.000000 5.000000 5.000000 5.000000 \n",

"mean 1.0 0.121020 337.888000 11.091236 13.694000 \n",

"std 0.0 0.017947 291.625932 0.416248 4.213494 \n",

"min 1.0 0.100800 102.920000 10.373491 8.100000 \n",

"25% 1.0 0.107100 162.340000 11.082143 11.630000 \n",

"50% 1.0 0.118900 228.220000 11.299732 14.290000 \n",

"75% 1.0 0.135700 366.860000 11.350407 14.970000 \n",

"max 1.0 0.142600 829.100000 11.350407 19.480000 \n",

"\n",

" fico days.with.cr.line revol.bal revol.util \\\n",

"count 5.000000 5.000000 5.000000 5.000000 \n",

"mean 701.000000 3975.183333 20879.000000 53.420000 \n",

"std 27.248853 1267.137358 15424.476085 21.809791 \n",

"min 667.000000 2699.958333 3511.000000 25.600000 \n",

"25% 682.000000 2760.000000 4740.000000 39.500000 \n",

"50% 707.000000 4066.000000 28854.000000 52.100000 \n",

"75% 712.000000 4710.000000 33623.000000 73.200000 \n",

"max 737.000000 5639.958333 33667.000000 76.700000 \n",

"\n",

" inq.last.6mths delinq.2yrs pub.rec not.fully.paid \n",

"count 5.000000 5.000000 5.0 5.0 \n",

"mean 0.400000 0.200000 0.0 0.0 \n",

"std 0.547723 0.447214 0.0 0.0 \n",

"min 0.000000 0.000000 0.0 0.0 \n",

"25% 0.000000 0.000000 0.0 0.0 \n",

"50% 0.000000 0.000000 0.0 0.0 \n",

"75% 1.000000 0.000000 0.0 0.0 \n",

"max 1.000000 1.000000 0.0 0.0 "

],

"text/html": [

"\n",

" <div id=\"df-8f6f305c-ed7a-49b0-8272-a6169477f88f\" 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>credit.policy</th>\n",

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

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

" <th>log.annual.inc</th>\n",

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

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

" <th>days.with.cr.line</th>\n",

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

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

" <th>inq.last.6mths</th>\n",

" <th>delinq.2yrs</th>\n",

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

" <th>not.fully.paid</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

" <td>5.0</td>\n",

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

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

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

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

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

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

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

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

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

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

" <td>5.0</td>\n",

" <td>5.0</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>0.121020</td>\n",

" <td>337.888000</td>\n",

" <td>11.091236</td>\n",

" <td>13.694000</td>\n",

" <td>701.000000</td>\n",

" <td>3975.183333</td>\n",

" <td>20879.000000</td>\n",

" <td>53.420000</td>\n",

" <td>0.400000</td>\n",

" <td>0.200000</td>\n",

" <td>0.0</td>\n",

" <td>0.0</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>0.0</td>\n",

" <td>0.017947</td>\n",

" <td>291.625932</td>\n",

" <td>0.416248</td>\n",

" <td>4.213494</td>\n",

" <td>27.248853</td>\n",

" <td>1267.137358</td>\n",

" <td>15424.476085</td>\n",

" <td>21.809791</td>\n",

" <td>0.547723</td>\n",

" <td>0.447214</td>\n",

" <td>0.0</td>\n",

" <td>0.0</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>0.100800</td>\n",

" <td>102.920000</td>\n",

" <td>10.373491</td>\n",

" <td>8.100000</td>\n",

" <td>667.000000</td>\n",

" <td>2699.958333</td>\n",

" <td>3511.000000</td>\n",

" <td>25.600000</td>\n",

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

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

" <td>0.0</td>\n",

" <td>0.0</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>0.107100</td>\n",

" <td>162.340000</td>\n",

" <td>11.082143</td>\n",

" <td>11.630000</td>\n",

" <td>682.000000</td>\n",

" <td>2760.000000</td>\n",

" <td>4740.000000</td>\n",

" <td>39.500000</td>\n",

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

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

" <td>0.0</td>\n",

" <td>0.0</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>0.118900</td>\n",

" <td>228.220000</td>\n",

" <td>11.299732</td>\n",

" <td>14.290000</td>\n",

" <td>707.000000</td>\n",

" <td>4066.000000</td>\n",

" <td>28854.000000</td>\n",

" <td>52.100000</td>\n",

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

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

" <td>0.0</td>\n",

" <td>0.0</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>0.135700</td>\n",

" <td>366.860000</td>\n",

" <td>11.350407</td>\n",

" <td>14.970000</td>\n",

" <td>712.000000</td>\n",

" <td>4710.000000</td>\n",

" <td>33623.000000</td>\n",

" <td>73.200000</td>\n",

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

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

" <td>0.0</td>\n",

" <td>0.0</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>0.142600</td>\n",

" <td>829.100000</td>\n",

" <td>11.350407</td>\n",

" <td>19.480000</td>\n",

" <td>737.000000</td>\n",

" <td>5639.958333</td>\n",

" <td>33667.000000</td>\n",

" <td>76.700000</td>\n",

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

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

" <td>0.0</td>\n",

" <td>0.0</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

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

"\n",

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

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-8f6f305c-ed7a-49b0-8272-a6169477f88f')\"\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 -960 960 960\">\n",

" <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220 220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",

" </svg>\n",

" </button>\n",

"\n",

" <style>\n",

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

" display:flex;\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",

" .colab-df-buttons div {\n",

" margin-bottom: 4px;\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-8f6f305c-ed7a-49b0-8272-a6169477f88f 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-8f6f305c-ed7a-49b0-8272-a6169477f88f');\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",

"\n",

"\n",

"<div id=\"df-02f1d32d-89af-4549-9537-ecd3432259b5\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-02f1d32d-89af-4549-9537-ecd3432259b5')\"\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",

"\n",

"<style>\n",

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

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

" --fill-color: #1967D2;\n",

" --hover-bg-color: #E2EBFA;\n",

" --hover-fill-color: #174EA6;\n",

" --disabled-fill-color: #AAA;\n",

" --disabled-bg-color: #DDD;\n",

" }\n",

"\n",

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

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

" --fill-color: #D2E3FC;\n",

" --hover-bg-color: #434B5C;\n",

" --hover-fill-color: #FFFFFF;\n",

" --disabled-bg-color: #3B4455;\n",

" --disabled-fill-color: #666;\n",

" }\n",

"\n",

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

" background-color: var(--bg-color);\n",

" border: none;\n",

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

" cursor: pointer;\n",

" display: none;\n",

" fill: var(--fill-color);\n",

" height: 32px;\n",

" padding: 0;\n",

" width: 32px;\n",

" }\n",

"\n",

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

" background-color: var(--hover-bg-color);\n",

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

" fill: var(--button-hover-fill-color);\n",

" }\n",

"\n",

" .colab-df-quickchart-complete:disabled,\n",

" .colab-df-quickchart-complete:disabled:hover {\n",

" background-color: var(--disabled-bg-color);\n",

" fill: var(--disabled-fill-color);\n",

" box-shadow: none;\n",

" }\n",

"\n",

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

" border: 2px solid var(--fill-color);\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" animation:\n",

" spin 1s steps(1) infinite;\n",

" }\n",

"\n",

" @keyframes spin {\n",

" 0% {\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" border-left-color: var(--fill-color);\n",

" }\n",

" 20% {\n",

" border-color: transparent;\n",

" border-left-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" }\n",

" 30% {\n",

" border-color: transparent;\n",

" border-left-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" border-right-color: var(--fill-color);\n",

" }\n",

" 40% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" }\n",

" 60% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" }\n",

" 80% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" border-bottom-color: var(--fill-color);\n",

" }\n",

" 90% {\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" }\n",

" }\n",

"</style>\n",

"\n",

" <script>\n",

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

" const quickchartButtonEl =\n",

" document.querySelector('#' + key + ' button');\n",

" quickchartButtonEl.disabled = true; // To prevent multiple clicks.\n",

" quickchartButtonEl.classList.add('colab-df-spinner');\n",

" try {\n",

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

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

" } catch (error) {\n",

" console.error('Error during call to suggestCharts:', error);\n",

" }\n",

" quickchartButtonEl.classList.remove('colab-df-spinner');\n",

" quickchartButtonEl.classList.add('colab-df-quickchart-complete');\n",

" }\n",

" (() => {\n",

" let quickchartButtonEl =\n",

" document.querySelector('#df-02f1d32d-89af-4549-9537-ecd3432259b5 button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

"\n",

" </div>\n",

" </div>\n"

],

"application/vnd.google.colaboratory.intrinsic+json": {

"type": "dataframe",

"variable\_name": "loans",

"summary": "{\n \"name\": \"loans\",\n \"rows\": 100,\n \"fields\": [\n {\n \"column\": \"credit.policy\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 1,\n \"max\": 1,\n \"num\_unique\_values\": 1,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"purpose\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 7,\n \"samples\": [\n \"debt\_consolidation\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"int.rate\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.023150181970181687,\n \"min\": 0.0714,\n \"max\": 0.1596,\n \"num\_unique\_values\": 46,\n \"samples\": [\n 0.0964\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"installment\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 162.3676786764776,\n \"min\": 32.55,\n \"max\": 829.1,\n \"num\_unique\_values\": 83,\n \"samples\": [\n 78.42\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"log.annual.inc\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.7157951261195352,\n \"min\": 8.987196821,\n \"max\": 12.4292162,\n \"num\_unique\_values\": 59,\n \"samples\": [\n 11.35040654\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"dti\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 5.981154494820301,\n \"min\": 0.0,\n \"max\": 22.09,\n \"num\_unique\_values\": 91,\n \"samples\": [\n 18.64\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"fico\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 44,\n \"min\": 627,\n \"max\": 812,\n \"num\_unique\_values\": 29,\n \"samples\": [\n 627\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"days.with.cr.line\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 2630.856598362081,\n \"min\": 1110.0,\n \"max\": 14008.95833,\n \"num\_unique\_values\": 91,\n \"samples\": [\n 2460.0\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"revol.bal\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 22933,\n \"min\": 0,\n \"max\": 128000,\n \"num\_unique\_values\": 96,\n \"samples\": [\n 8155\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"revol.util\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 26.17236957734109,\n \"min\": 0.0,\n \"max\": 93.4,\n \"num\_unique\_values\": 93,\n \"samples\": [\n 16.9\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"inq.last.6mths\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 1,\n \"min\": 0,\n \"max\": 4,\n \"num\_unique\_values\": 5,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"delinq.2yrs\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 2,\n \"num\_unique\_values\": 3,\n \"samples\": [\n 0\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"pub.rec\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 1,\n \"num\_unique\_values\": 2,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"not.fully.paid\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 1,\n \"num\_unique\_values\": 2,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}"

}

},

"metadata": {},

"execution\_count": 7

}

]

},

{

"source": [

"import matplotlib.pyplot as plt\n",

"\n",

"plt.figure(figsize=(10,6))\n",

"loans[loans['credit.policy']==1]['fico'].hist(alpha=0.5,color='blue',bins=30,label='Credit Policy=1')\n",

"loans[loans['credit.policy']==0]['fico'].hist(alpha=0.5,color='red',bins=30,label='Credit Policy=0')\n",

"plt.legend()\n",

"plt.xlabel('FICO')"

],

"cell\_type": "code",

"metadata": {

"colab": {

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

"height": 559

},

"id": "PuR1VOlt5RDF",

"outputId": "46644a33-1281-4eff-ea02-1434a0add2df"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

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

]

},

"metadata": {},

"execution\_count": 10

},

{

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

"data": {

"text/plain": [

"<Figure size 1000x600 with 1 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"cell\_type": "code",

"source": [

"plt.figure(figsize=(10,6))\n",

"loans[loans['not.fully.paid']==1]['fico'].hist(alpha=0.5,color='blue',bins=30,label='Not Fully Paid=1')\n",

"loans[loans['not.fully.paid']==0]['fico'].hist(alpha=0.5,color='red',bins=30,label='Not Fully Paid=0')\n",

"plt.legend()\n",

"plt.xlabel('FICO')"

],

"metadata": {

"colab": {

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

"height": 559

},

"id": "iH-IFZ5s5S0-",

"outputId": "df6eea93-dbfa-4315-d684-ae4b6d6c2908"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

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

]

},

"metadata": {},

"execution\_count": 11

},

{

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

"data": {

"text/plain": [

"<Figure size 1000x600 with 1 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"source": [

"import pandas as pd\n",

"import matplotlib.pyplot as plt\n",

"import seaborn as sns"

],

"cell\_type": "code",

"metadata": {

"id": "PiIhXV5u5sBL"

},

"execution\_count": null,

"outputs": []

},

{

"source": [

"loans = pd.read\_csv(\"/content/datacamp\_workspace\_export\_2024-04-01 19\_32\_28.csv\")"

],

"cell\_type": "code",

"metadata": {

"id": "MVkr3Zj15tpI"

},

"execution\_count": null,

"outputs": []

},

{

"source": [

"plt.figure(figsize=(11,7))\n",

"sns.countplot(x=\"purpose\", hue=\"not.fully.paid\", data=loans, palette=\"Set1\")"

],

"cell\_type": "code",

"metadata": {

"colab": {

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

"height": 597

},

"id": "O-3GjgZR50uM",

"outputId": "201305e0-1457-4bb9-e01e-e0b21610da7e"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"<Axes: xlabel='purpose', ylabel='count'>"

]

},

"metadata": {},

"execution\_count": 19

},

{

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

"data": {

"text/plain": [

"<Figure size 1100x700 with 1 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"cell\_type": "code",

"source": [

"sns.jointplot(x='fico',y='int.rate',data=loans,color='purple')"

],

"metadata": {

"colab": {

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

"height": 624

},

"id": "rdICPE\_g52KE",

"outputId": "a7f84dc9-985a-4635-cbde-d948c6165ec4"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"<seaborn.axisgrid.JointGrid at 0x799cd2a387c0>"

]

},

"metadata": {},

"execution\_count": 20

},

{

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

"data": {

"text/plain": [

"<Figure size 600x600 with 3 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"cell\_type": "code",

"source": [

"plt.figure(figsize=(11,7))\n",

"sns.lmplot(x='fico',y='int.rate',data=loans,hue='credit.policy',col='not.fully.paid',palette='Set1')"

],

"metadata": {

"colab": {

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

"height": 439

},

"id": "A-IVWeQI57Hu",

"outputId": "eafa2f5f-f687-4b55-adcb-5208376c4ef4"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"<seaborn.axisgrid.FacetGrid at 0x799cd27b59f0>"

]

},

"metadata": {},

"execution\_count": 21

},

{

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

"data": {

"text/plain": [

"<Figure size 1100x700 with 0 Axes>"

]

},

"metadata": {}

},

{

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

"data": {

"text/plain": [

"<Figure size 1096.36x500 with 2 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"cell\_type": "code",

"source": [

"loans.info()"

],

"metadata": {

"colab": {

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

},

"id": "mWqByabQ6H6M",

"outputId": "26590234-029b-446c-e822-5f4eb3b9459d"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

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

"RangeIndex: 100 entries, 0 to 99\n",

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

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

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

" 0 credit.policy 100 non-null int64 \n",

" 1 purpose 100 non-null object \n",

" 2 int.rate 100 non-null float64\n",

" 3 installment 100 non-null float64\n",

" 4 log.annual.inc 100 non-null float64\n",

" 5 dti 100 non-null float64\n",

" 6 fico 100 non-null int64 \n",

" 7 days.with.cr.line 100 non-null float64\n",

" 8 revol.bal 100 non-null int64 \n",

" 9 revol.util 100 non-null float64\n",

" 10 inq.last.6mths 100 non-null int64 \n",

" 11 delinq.2yrs 100 non-null int64 \n",

" 12 pub.rec 100 non-null int64 \n",

" 13 not.fully.paid 100 non-null int64 \n",

"dtypes: float64(6), int64(7), object(1)\n",

"memory usage: 11.1+ KB\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"loan\_purpose=['purpose']"

],

"metadata": {

"id": "deb3o53A6Beg"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"final\_data=pd.get\_dummies(loans,columns=loan\_purpose,drop\_first=True)"

],

"metadata": {

"id": "o5o3j-it6PXX"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"# In the above code, drop\_first is done to avoid multi-colinearity\n",

"final\_data.info()"

],

"metadata": {

"colab": {

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

},

"id": "vSHq-\_Mx6UpZ",

"outputId": "cb3778f7-90d9-4b16-beaf-e058ec569533"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

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

"RangeIndex: 100 entries, 0 to 99\n",

"Data columns (total 19 columns):\n",

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

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

" 0 credit.policy 100 non-null int64 \n",

" 1 int.rate 100 non-null float64\n",

" 2 installment 100 non-null float64\n",

" 3 log.annual.inc 100 non-null float64\n",

" 4 dti 100 non-null float64\n",

" 5 fico 100 non-null int64 \n",

" 6 days.with.cr.line 100 non-null float64\n",

" 7 revol.bal 100 non-null int64 \n",

" 8 revol.util 100 non-null float64\n",

" 9 inq.last.6mths 100 non-null int64 \n",

" 10 delinq.2yrs 100 non-null int64 \n",

" 11 pub.rec 100 non-null int64 \n",

" 12 not.fully.paid 100 non-null int64 \n",

" 13 purpose\_credit\_card 100 non-null bool \n",

" 14 purpose\_debt\_consolidation 100 non-null bool \n",

" 15 purpose\_educational 100 non-null bool \n",

" 16 purpose\_home\_improvement 100 non-null bool \n",

" 17 purpose\_major\_purchase 100 non-null bool \n",

" 18 purpose\_small\_business 100 non-null bool \n",

"dtypes: bool(6), float64(6), int64(7)\n",

"memory usage: 10.9 KB\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"final\_data.head()"

],

"metadata": {

"colab": {

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

"height": 226

},

"id": "hsWAv0EX6ZJ9",

"outputId": "eebe1a40-0c4a-4e09-cac5-93411a4a497b"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

" credit.policy int.rate installment log.annual.inc dti fico \\\n",

"0 1 0.1189 829.10 11.350407 19.48 737 \n",

"1 1 0.1071 228.22 11.082143 14.29 707 \n",

"2 1 0.1357 366.86 10.373491 11.63 682 \n",

"3 1 0.1008 162.34 11.350407 8.10 712 \n",

"4 1 0.1426 102.92 11.299732 14.97 667 \n",

"\n",

" days.with.cr.line revol.bal revol.util inq.last.6mths delinq.2yrs \\\n",

"0 5639.958333 28854 52.1 0 0 \n",

"1 2760.000000 33623 76.7 0 0 \n",

"2 4710.000000 3511 25.6 1 0 \n",

"3 2699.958333 33667 73.2 1 0 \n",

"4 4066.000000 4740 39.5 0 1 \n",

"\n",

" pub.rec not.fully.paid purpose\_credit\_card purpose\_debt\_consolidation \\\n",

"0 0 0 False True \n",

"1 0 0 True False \n",

"2 0 0 False True \n",

"3 0 0 False True \n",

"4 0 0 True False \n",

"\n",

" purpose\_educational purpose\_home\_improvement purpose\_major\_purchase \\\n",

"0 False False False \n",

"1 False False False \n",

"2 False False False \n",

"3 False False False \n",

"4 False False False \n",

"\n",

" purpose\_small\_business \n",

"0 False \n",

"1 False \n",

"2 False \n",

"3 False \n",

"4 False "

],

"text/html": [

"\n",

" <div id=\"df-ec738050-e396-42bd-b32b-4707695a82cd\" 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>credit.policy</th>\n",

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

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

" <th>log.annual.inc</th>\n",

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

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

" <th>days.with.cr.line</th>\n",

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

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

" <th>inq.last.6mths</th>\n",

" <th>delinq.2yrs</th>\n",

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

" <th>not.fully.paid</th>\n",

" <th>purpose\_credit\_card</th>\n",

" <th>purpose\_debt\_consolidation</th>\n",

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

" <th>purpose\_home\_improvement</th>\n",

" <th>purpose\_major\_purchase</th>\n",

" <th>purpose\_small\_business</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

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

" <td>0.1189</td>\n",

" <td>829.10</td>\n",

" <td>11.350407</td>\n",

" <td>19.48</td>\n",

" <td>737</td>\n",

" <td>5639.958333</td>\n",

" <td>28854</td>\n",

" <td>52.1</td>\n",

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

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

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

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

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

" <td>0.1071</td>\n",

" <td>228.22</td>\n",

" <td>11.082143</td>\n",

" <td>14.29</td>\n",

" <td>707</td>\n",

" <td>2760.000000</td>\n",

" <td>33623</td>\n",

" <td>76.7</td>\n",

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

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

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

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

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

" <td>0.1357</td>\n",

" <td>366.86</td>\n",

" <td>10.373491</td>\n",

" <td>11.63</td>\n",

" <td>682</td>\n",

" <td>4710.000000</td>\n",

" <td>3511</td>\n",

" <td>25.6</td>\n",

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

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

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

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

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

" <td>0.1008</td>\n",

" <td>162.34</td>\n",

" <td>11.350407</td>\n",

" <td>8.10</td>\n",

" <td>712</td>\n",

" <td>2699.958333</td>\n",

" <td>33667</td>\n",

" <td>73.2</td>\n",

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

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

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

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

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

" <td>0.1426</td>\n",

" <td>102.92</td>\n",

" <td>11.299732</td>\n",

" <td>14.97</td>\n",

" <td>667</td>\n",

" <td>4066.000000</td>\n",

" <td>4740</td>\n",

" <td>39.5</td>\n",

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

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

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

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

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

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

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

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

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

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

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

"\n",

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

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-ec738050-e396-42bd-b32b-4707695a82cd')\"\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 -960 960 960\">\n",

" <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220 220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",

" </svg>\n",

" </button>\n",

"\n",

" <style>\n",

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

" display:flex;\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",

" .colab-df-buttons div {\n",

" margin-bottom: 4px;\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-ec738050-e396-42bd-b32b-4707695a82cd 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-ec738050-e396-42bd-b32b-4707695a82cd');\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",

"\n",

"\n",

"<div id=\"df-11eced67-2c8a-496e-b70e-0c1f829598ca\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-11eced67-2c8a-496e-b70e-0c1f829598ca')\"\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",

"\n",

"<style>\n",

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

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

" --fill-color: #1967D2;\n",

" --hover-bg-color: #E2EBFA;\n",

" --hover-fill-color: #174EA6;\n",

" --disabled-fill-color: #AAA;\n",

" --disabled-bg-color: #DDD;\n",

" }\n",

"\n",

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

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

" --fill-color: #D2E3FC;\n",

" --hover-bg-color: #434B5C;\n",

" --hover-fill-color: #FFFFFF;\n",

" --disabled-bg-color: #3B4455;\n",

" --disabled-fill-color: #666;\n",

" }\n",

"\n",

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

" background-color: var(--bg-color);\n",

" border: none;\n",

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

" cursor: pointer;\n",

" display: none;\n",

" fill: var(--fill-color);\n",

" height: 32px;\n",

" padding: 0;\n",

" width: 32px;\n",

" }\n",

"\n",

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

" background-color: var(--hover-bg-color);\n",

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

" fill: var(--button-hover-fill-color);\n",

" }\n",

"\n",

" .colab-df-quickchart-complete:disabled,\n",

" .colab-df-quickchart-complete:disabled:hover {\n",

" background-color: var(--disabled-bg-color);\n",

" fill: var(--disabled-fill-color);\n",

" box-shadow: none;\n",

" }\n",

"\n",

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

" border: 2px solid var(--fill-color);\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" animation:\n",

" spin 1s steps(1) infinite;\n",

" }\n",

"\n",

" @keyframes spin {\n",

" 0% {\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" border-left-color: var(--fill-color);\n",

" }\n",

" 20% {\n",

" border-color: transparent;\n",

" border-left-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" }\n",

" 30% {\n",

" border-color: transparent;\n",

" border-left-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" border-right-color: var(--fill-color);\n",

" }\n",

" 40% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" }\n",

" 60% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" }\n",

" 80% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" border-bottom-color: var(--fill-color);\n",

" }\n",

" 90% {\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" }\n",

" }\n",

"</style>\n",

"\n",

" <script>\n",

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

" const quickchartButtonEl =\n",

" document.querySelector('#' + key + ' button');\n",

" quickchartButtonEl.disabled = true; // To prevent multiple clicks.\n",

" quickchartButtonEl.classList.add('colab-df-spinner');\n",

" try {\n",

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

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

" } catch (error) {\n",

" console.error('Error during call to suggestCharts:', error);\n",

" }\n",

" quickchartButtonEl.classList.remove('colab-df-spinner');\n",

" quickchartButtonEl.classList.add('colab-df-quickchart-complete');\n",

" }\n",

" (() => {\n",

" let quickchartButtonEl =\n",

" document.querySelector('#df-11eced67-2c8a-496e-b70e-0c1f829598ca button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

"\n",

" </div>\n",

" </div>\n"

],

"application/vnd.google.colaboratory.intrinsic+json": {

"type": "dataframe",

"variable\_name": "final\_data",

"summary": "{\n \"name\": \"final\_data\",\n \"rows\": 100,\n \"fields\": [\n {\n \"column\": \"credit.policy\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 1,\n \"max\": 1,\n \"num\_unique\_values\": 1,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"int.rate\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.023150181970181687,\n \"min\": 0.0714,\n \"max\": 0.1596,\n \"num\_unique\_values\": 46,\n \"samples\": [\n 0.0964\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"installment\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 162.3676786764776,\n \"min\": 32.55,\n \"max\": 829.1,\n \"num\_unique\_values\": 83,\n \"samples\": [\n 78.42\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"log.annual.inc\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.7157951261195352,\n \"min\": 8.987196821,\n \"max\": 12.4292162,\n \"num\_unique\_values\": 59,\n \"samples\": [\n 11.35040654\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"dti\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 5.981154494820301,\n \"min\": 0.0,\n \"max\": 22.09,\n \"num\_unique\_values\": 91,\n \"samples\": [\n 18.64\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"fico\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 44,\n \"min\": 627,\n \"max\": 812,\n \"num\_unique\_values\": 29,\n \"samples\": [\n 627\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"days.with.cr.line\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 2630.856598362081,\n \"min\": 1110.0,\n \"max\": 14008.95833,\n \"num\_unique\_values\": 91,\n \"samples\": [\n 2460.0\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"revol.bal\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 22933,\n \"min\": 0,\n \"max\": 128000,\n \"num\_unique\_values\": 96,\n \"samples\": [\n 8155\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"revol.util\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 26.17236957734109,\n \"min\": 0.0,\n \"max\": 93.4,\n \"num\_unique\_values\": 93,\n \"samples\": [\n 16.9\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"inq.last.6mths\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 1,\n \"min\": 0,\n \"max\": 4,\n \"num\_unique\_values\": 5,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"delinq.2yrs\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 2,\n \"num\_unique\_values\": 3,\n \"samples\": [\n 0\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"pub.rec\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 1,\n \"num\_unique\_values\": 2,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"not.fully.paid\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 1,\n \"num\_unique\_values\": 2,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"purpose\_credit\_card\",\n \"properties\": {\n \"dtype\": \"boolean\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n true\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"purpose\_debt\_consolidation\",\n \"properties\": {\n \"dtype\": \"boolean\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n false\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"purpose\_educational\",\n \"properties\": {\n \"dtype\": \"boolean\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n true\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"purpose\_home\_improvement\",\n \"properties\": {\n \"dtype\": \"boolean\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n true\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"purpose\_major\_purchase\",\n \"properties\": {\n \"dtype\": \"boolean\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n true\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"purpose\_small\_business\",\n \"properties\": {\n \"dtype\": \"boolean\",\n \"num\_unique\_values\": 2,\n \"samples\": [\n true\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}"

}

},

"metadata": {},

"execution\_count": 26

}

]

},

{

"source": [

"import pandas as pd"

],

"cell\_type": "code",

"metadata": {

"id": "wpj8rY2MZ2Gh"

},

"execution\_count": null,

"outputs": []

},

{

"source": [

"data = pd.read\_csv('/content/datacamp\_workspace\_export\_2024-04-01 19\_32\_28.csv')"

],

"cell\_type": "code",

"metadata": {

"id": "hC6f6lhhZ2ii"

},

"execution\_count": null,

"outputs": []

},

{

"source": [

"data['new\_feature'] = data['dti'] \* data['fico']"

],

"cell\_type": "code",

"metadata": {

"id": "rldp9LbQZ230"

},

"execution\_count": null,

"outputs": []

},

{

"source": [

"data.head()"

],

"cell\_type": "code",

"metadata": {

"colab": {

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

"height": 226

},

"id": "saZhTe6jZ3Wk",

"outputId": "1a7620de-ef4c-4863-b3f8-b8503f96b63d"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

" credit.policy purpose int.rate installment log.annual.inc \\\n",

"0 1 debt\_consolidation 0.1189 829.10 11.350407 \n",

"1 1 credit\_card 0.1071 228.22 11.082143 \n",

"2 1 debt\_consolidation 0.1357 366.86 10.373491 \n",

"3 1 debt\_consolidation 0.1008 162.34 11.350407 \n",

"4 1 credit\_card 0.1426 102.92 11.299732 \n",

"\n",

" dti fico days.with.cr.line revol.bal revol.util inq.last.6mths \\\n",

"0 19.48 737 5639.958333 28854 52.1 0 \n",

"1 14.29 707 2760.000000 33623 76.7 0 \n",

"2 11.63 682 4710.000000 3511 25.6 1 \n",

"3 8.10 712 2699.958333 33667 73.2 1 \n",

"4 14.97 667 4066.000000 4740 39.5 0 \n",

"\n",

" delinq.2yrs pub.rec not.fully.paid new\_feature \n",

"0 0 0 0 14356.76 \n",

"1 0 0 0 10103.03 \n",

"2 0 0 0 7931.66 \n",

"3 0 0 0 5767.20 \n",

"4 1 0 0 9984.99 "

],

"text/html": [

"\n",

" <div id=\"df-b35951b7-1718-4d02-8ea1-f81c16729e0c\" 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>credit.policy</th>\n",

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

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

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

" <th>log.annual.inc</th>\n",

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

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

" <th>days.with.cr.line</th>\n",

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

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

" <th>inq.last.6mths</th>\n",

" <th>delinq.2yrs</th>\n",

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

" <th>not.fully.paid</th>\n",

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

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

" <td>debt\_consolidation</td>\n",

" <td>0.1189</td>\n",

" <td>829.10</td>\n",

" <td>11.350407</td>\n",

" <td>19.48</td>\n",

" <td>737</td>\n",

" <td>5639.958333</td>\n",

" <td>28854</td>\n",

" <td>52.1</td>\n",

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

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

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

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

" <td>14356.76</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>credit\_card</td>\n",

" <td>0.1071</td>\n",

" <td>228.22</td>\n",

" <td>11.082143</td>\n",

" <td>14.29</td>\n",

" <td>707</td>\n",

" <td>2760.000000</td>\n",

" <td>33623</td>\n",

" <td>76.7</td>\n",

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

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

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

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

" <td>10103.03</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>debt\_consolidation</td>\n",

" <td>0.1357</td>\n",

" <td>366.86</td>\n",

" <td>10.373491</td>\n",

" <td>11.63</td>\n",

" <td>682</td>\n",

" <td>4710.000000</td>\n",

" <td>3511</td>\n",

" <td>25.6</td>\n",

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

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

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

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

" <td>7931.66</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>debt\_consolidation</td>\n",

" <td>0.1008</td>\n",

" <td>162.34</td>\n",

" <td>11.350407</td>\n",

" <td>8.10</td>\n",

" <td>712</td>\n",

" <td>2699.958333</td>\n",

" <td>33667</td>\n",

" <td>73.2</td>\n",

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

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

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

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

" <td>5767.20</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>credit\_card</td>\n",

" <td>0.1426</td>\n",

" <td>102.92</td>\n",

" <td>11.299732</td>\n",

" <td>14.97</td>\n",

" <td>667</td>\n",

" <td>4066.000000</td>\n",

" <td>4740</td>\n",

" <td>39.5</td>\n",

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

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

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

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

" <td>9984.99</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>\n",

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

"\n",

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

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-b35951b7-1718-4d02-8ea1-f81c16729e0c')\"\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 -960 960 960\">\n",

" <path d=\"M120-120v-720h720v720H120Zm60-500h600v-160H180v160Zm220 220h160v-160H400v160Zm0 220h160v-160H400v160ZM180-400h160v-160H180v160Zm440 0h160v-160H620v160ZM180-180h160v-160H180v160Zm440 0h160v-160H620v160Z\"/>\n",

" </svg>\n",

" </button>\n",

"\n",

" <style>\n",

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

" display:flex;\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",

" .colab-df-buttons div {\n",

" margin-bottom: 4px;\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-b35951b7-1718-4d02-8ea1-f81c16729e0c 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-b35951b7-1718-4d02-8ea1-f81c16729e0c');\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",

"\n",

"\n",

"<div id=\"df-eeefacd8-374a-4559-9471-8748780a801d\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-eeefacd8-374a-4559-9471-8748780a801d')\"\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",

"\n",

"<style>\n",

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

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

" --fill-color: #1967D2;\n",

" --hover-bg-color: #E2EBFA;\n",

" --hover-fill-color: #174EA6;\n",

" --disabled-fill-color: #AAA;\n",

" --disabled-bg-color: #DDD;\n",

" }\n",

"\n",

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

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

" --fill-color: #D2E3FC;\n",

" --hover-bg-color: #434B5C;\n",

" --hover-fill-color: #FFFFFF;\n",

" --disabled-bg-color: #3B4455;\n",

" --disabled-fill-color: #666;\n",

" }\n",

"\n",

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

" background-color: var(--bg-color);\n",

" border: none;\n",

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

" cursor: pointer;\n",

" display: none;\n",

" fill: var(--fill-color);\n",

" height: 32px;\n",

" padding: 0;\n",

" width: 32px;\n",

" }\n",

"\n",

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

" background-color: var(--hover-bg-color);\n",

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

" fill: var(--button-hover-fill-color);\n",

" }\n",

"\n",

" .colab-df-quickchart-complete:disabled,\n",

" .colab-df-quickchart-complete:disabled:hover {\n",

" background-color: var(--disabled-bg-color);\n",

" fill: var(--disabled-fill-color);\n",

" box-shadow: none;\n",

" }\n",

"\n",

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

" border: 2px solid var(--fill-color);\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" animation:\n",

" spin 1s steps(1) infinite;\n",

" }\n",

"\n",

" @keyframes spin {\n",

" 0% {\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" border-left-color: var(--fill-color);\n",

" }\n",

" 20% {\n",

" border-color: transparent;\n",

" border-left-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" }\n",

" 30% {\n",

" border-color: transparent;\n",

" border-left-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" border-right-color: var(--fill-color);\n",

" }\n",

" 40% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" border-top-color: var(--fill-color);\n",

" }\n",

" 60% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" }\n",

" 80% {\n",

" border-color: transparent;\n",

" border-right-color: var(--fill-color);\n",

" border-bottom-color: var(--fill-color);\n",

" }\n",

" 90% {\n",

" border-color: transparent;\n",

" border-bottom-color: var(--fill-color);\n",

" }\n",

" }\n",

"</style>\n",

"\n",

" <script>\n",

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

" const quickchartButtonEl =\n",

" document.querySelector('#' + key + ' button');\n",

" quickchartButtonEl.disabled = true; // To prevent multiple clicks.\n",

" quickchartButtonEl.classList.add('colab-df-spinner');\n",

" try {\n",

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

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

" } catch (error) {\n",

" console.error('Error during call to suggestCharts:', error);\n",

" }\n",

" quickchartButtonEl.classList.remove('colab-df-spinner');\n",

" quickchartButtonEl.classList.add('colab-df-quickchart-complete');\n",

" }\n",

" (() => {\n",

" let quickchartButtonEl =\n",

" document.querySelector('#df-eeefacd8-374a-4559-9471-8748780a801d button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

"\n",

" </div>\n",

" </div>\n"

],

"application/vnd.google.colaboratory.intrinsic+json": {

"type": "dataframe",

"variable\_name": "data",

"summary": "{\n \"name\": \"data\",\n \"rows\": 100,\n \"fields\": [\n {\n \"column\": \"credit.policy\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 1,\n \"max\": 1,\n \"num\_unique\_values\": 1,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"purpose\",\n \"properties\": {\n \"dtype\": \"category\",\n \"num\_unique\_values\": 7,\n \"samples\": [\n \"debt\_consolidation\"\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"int.rate\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.023150181970181687,\n \"min\": 0.0714,\n \"max\": 0.1596,\n \"num\_unique\_values\": 46,\n \"samples\": [\n 0.0964\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"installment\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 162.3676786764776,\n \"min\": 32.55,\n \"max\": 829.1,\n \"num\_unique\_values\": 83,\n \"samples\": [\n 78.42\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"log.annual.inc\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.7157951261195352,\n \"min\": 8.987196821,\n \"max\": 12.4292162,\n \"num\_unique\_values\": 59,\n \"samples\": [\n 11.35040654\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"dti\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 5.981154494820301,\n \"min\": 0.0,\n \"max\": 22.09,\n \"num\_unique\_values\": 91,\n \"samples\": [\n 18.64\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"fico\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 44,\n \"min\": 627,\n \"max\": 812,\n \"num\_unique\_values\": 29,\n \"samples\": [\n 627\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"days.with.cr.line\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 2630.856598362081,\n \"min\": 1110.0,\n \"max\": 14008.95833,\n \"num\_unique\_values\": 91,\n \"samples\": [\n 2460.0\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"revol.bal\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 22933,\n \"min\": 0,\n \"max\": 128000,\n \"num\_unique\_values\": 96,\n \"samples\": [\n 8155\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"revol.util\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 26.17236957734109,\n \"min\": 0.0,\n \"max\": 93.4,\n \"num\_unique\_values\": 93,\n \"samples\": [\n 16.9\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"inq.last.6mths\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 1,\n \"min\": 0,\n \"max\": 4,\n \"num\_unique\_values\": 5,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"delinq.2yrs\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 2,\n \"num\_unique\_values\": 3,\n \"samples\": [\n 0\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"pub.rec\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 1,\n \"num\_unique\_values\": 2,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"not.fully.paid\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0,\n \"min\": 0,\n \"max\": 1,\n \"num\_unique\_values\": 2,\n \"samples\": [\n 1\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"new\_feature\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 4264.836865386245,\n \"min\": 0.0,\n \"max\": 15440.0,\n \"num\_unique\_values\": 99,\n \"samples\": [\n 4982.49\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}"

}

},

"metadata": {},

"execution\_count": 6

}

]

},

{

"cell\_type": "code",

"source": [

"X = final\_data.drop('not.fully.paid',axis=1)\n",

"y=final\_data['not.fully.paid']"

],

"metadata": {

"id": "sGK9BbqN6dcQ"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

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

],

"metadata": {

"id": "-PRGrRz46iuv"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=101)"

],

"metadata": {

"id": "9QYfsFc76msG"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"from sklearn.tree import DecisionTreeClassifier\n",

"# Instantiating Decision Tree model (basically creating a decision tree object)\n",

"dtree = DecisionTreeClassifier()\n",

"# Training or fitting the model on training data\n",

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

],

"metadata": {

"colab": {

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

"height": 74

},

"id": "AM-hhehq6q77",

"outputId": "51deacdb-ba2e-4ffa-dd7c-a85cef3401fa"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"DecisionTreeClassifier()"

],

"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>DecisionTreeClassifier()</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\">DecisionTreeClassifier</label><div class=\"sk-toggleable\_\_content\"><pre>DecisionTreeClassifier()</pre></div></div></div></div></div>"

]

},

"metadata": {},

"execution\_count": 30

}

]

},

{

"cell\_type": "code",

"source": [

"dtree\_predictions = dtree.predict(X\_test)"

],

"metadata": {

"id": "N0xBuyJy6yal"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"from sklearn.metrics import classification\_report, confusion\_matrix\n",

"print(classification\_report(y\_test,dtree\_predictions))"

],

"metadata": {

"colab": {

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

},

"id": "k2FAmdqh63Kk",

"outputId": "355e8eea-fd5b-49bf-c384-5d5ae3c3f899"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

" precision recall f1-score support\n",

"\n",

" 0 0.89 0.93 0.91 27\n",

" 1 0.00 0.00 0.00 3\n",

"\n",

" accuracy 0.83 30\n",

" macro avg 0.45 0.46 0.45 30\n",

"weighted avg 0.80 0.83 0.82 30\n",

"\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"print(confusion\_matrix(y\_test,dtree\_predictions))"

],

"metadata": {

"colab": {

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

},

"id": "NZ7lMkMf67QP",

"outputId": "926760b1-7406-45fc-b2d7-d2ed13e4f5cd"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"[[25 2]\n",

" [ 3 0]]\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"from sklearn.ensemble import RandomForestClassifier\n",

"# Instantiating Random Forest model (basically creating a random forest object)\n",

"rfc = RandomForestClassifier(n\_estimators=300)\n",

"# Training or fitting the model on training data\n",

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

],

"metadata": {

"colab": {

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

"height": 74

},

"id": "C35skdCo6\_2S",

"outputId": "2be3e6cc-34f3-4f26-b9a2-f99c2445dff8"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"RandomForestClassifier(n\_estimators=300)"

],

"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>RandomForestClassifier(n\_estimators=300)</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\">RandomForestClassifier</label><div class=\"sk-toggleable\_\_content\"><pre>RandomForestClassifier(n\_estimators=300)</pre></div></div></div></div></div>"

]

},

"metadata": {},

"execution\_count": 34

}

]

},

{

"cell\_type": "code",

"source": [

"rfc\_predictions = rfc.predict(X\_test)"

],

"metadata": {

"id": "BRBYx02p7Hhm"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"print(classification\_report(y\_test,rfc\_predictions))"

],

"metadata": {

"colab": {

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

},

"id": "rK2CLnax7Izr",

"outputId": "c99e6a22-456b-421a-f20a-cd1e41109f84"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

" precision recall f1-score support\n",

"\n",

" 0 0.90 1.00 0.95 27\n",

" 1 0.00 0.00 0.00 3\n",

"\n",

" accuracy 0.90 30\n",

" macro avg 0.45 0.50 0.47 30\n",

"weighted avg 0.81 0.90 0.85 30\n",

"\n"

]

},

{

"output\_type": "stream",

"name": "stderr",

"text": [

"/usr/local/lib/python3.10/dist-packages/sklearn/metrics/\_classification.py:1344: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero\_division` parameter to control this behavior.\n",

" \_warn\_prf(average, modifier, msg\_start, len(result))\n",

"/usr/local/lib/python3.10/dist-packages/sklearn/metrics/\_classification.py:1344: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero\_division` parameter to control this behavior.\n",

" \_warn\_prf(average, modifier, msg\_start, len(result))\n",

"/usr/local/lib/python3.10/dist-packages/sklearn/metrics/\_classification.py:1344: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero\_division` parameter to control this behavior.\n",

" \_warn\_prf(average, modifier, msg\_start, len(result))\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"print(confusion\_matrix(y\_test,rfc\_predictions))"

],

"metadata": {

"colab": {

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

},

"id": "DfUMeDt47Mzu",

"outputId": "3684b39c-8f28-4876-9276-f77167608849"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"[[27 0]\n",

" [ 3 0]]\n"

]

}

]

},

{

"source": [

"!pip install scikit-learn"

],

"cell\_type": "code",

"metadata": {

"colab": {

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

},

"id": "5iimAz74cwni",

"outputId": "71bf46e5-2c22-45ad-9085-c468d893f864"

},

"execution\_count": 7,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (1.2.2)\n",

"Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.25.2)\n",

"Requirement already satisfied: scipy>=1.3.2 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.11.4)\n",

"Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.4.0)\n",

"Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (3.4.0)\n"

]

}

]

},

{

"source": [

"from sklearn.model\_selection import GridSearchCV"

],

"cell\_type": "code",

"metadata": {

"id": "-np8KNWwcxDi"

},

"execution\_count": 8,

"outputs": []

},

{

"source": [

"# Define the best\_model variable (replace this with your actual model)\n",

"best\_model = ...\n",

"\n",

"# Save the best model to disk\n",

"joblib.dump(best\_model, 'loan\_classifier.joblib')"

],

"cell\_type": "code",

"metadata": {

"colab": {

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

},

"id": "f9BA85xndFjJ",

"outputId": "548c1fcb-80af-47d6-de24-0fc0f74f2980"

},

"execution\_count": 13,

"outputs": [

{

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

"data": {

"text/plain": [

"['loan\_classifier.joblib']"

]

},

"metadata": {},

"execution\_count": 13

}

]

},

{

"cell\_type": "code",

"source": [

"!pip install gradio"

],

"metadata": {

"colab": {

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

},

"id": "KriET79rdWiu",

"outputId": "bb99849d-ee0e-4e43-a033-fa1d522dcd07"

},

"execution\_count": 17,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"Collecting gradio\n",

" Downloading gradio-4.26.0-py3-none-any.whl (17.1 MB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m17.1/17.1 MB\u001b[0m \u001b[31m32.1 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hCollecting aiofiles<24.0,>=22.0 (from gradio)\n",

" Downloading aiofiles-23.2.1-py3-none-any.whl (15 kB)\n",

"Requirement already satisfied: altair<6.0,>=4.2.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (4.2.2)\n",

"Collecting fastapi (from gradio)\n",

" Downloading fastapi-0.110.1-py3-none-any.whl (91 kB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m91.9/91.9 kB\u001b[0m \u001b[31m7.4 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hCollecting ffmpy (from gradio)\n",

" Downloading ffmpy-0.3.2.tar.gz (5.5 kB)\n",

" Preparing metadata (setup.py) ... \u001b[?25l\u001b[?25hdone\n",

"Collecting gradio-client==0.15.1 (from gradio)\n",

" Downloading gradio\_client-0.15.1-py3-none-any.whl (313 kB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m313.6/313.6 kB\u001b[0m \u001b[31m17.8 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hCollecting httpx>=0.24.1 (from gradio)\n",

" Downloading httpx-0.27.0-py3-none-any.whl (75 kB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m75.6/75.6 kB\u001b[0m \u001b[31m7.8 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hRequirement already satisfied: huggingface-hub>=0.19.3 in /usr/local/lib/python3.10/dist-packages (from gradio) (0.20.3)\n",

"Requirement already satisfied: importlib-resources<7.0,>=1.3 in /usr/local/lib/python3.10/dist-packages (from gradio) (6.4.0)\n",

"Requirement already satisfied: jinja2<4.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (3.1.3)\n",

"Requirement already satisfied: markupsafe~=2.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (2.1.5)\n",

"Requirement already satisfied: matplotlib~=3.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (3.7.1)\n",

"Requirement already satisfied: numpy~=1.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (1.25.2)\n",

"Collecting orjson~=3.0 (from gradio)\n",

" Downloading orjson-3.10.0-cp310-cp310-manylinux\_2\_17\_x86\_64.manylinux2014\_x86\_64.whl (144 kB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m144.8/144.8 kB\u001b[0m \u001b[31m14.0 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hRequirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from gradio) (24.0)\n",

"Requirement already satisfied: pandas<3.0,>=1.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (2.0.3)\n",

"Requirement already satisfied: pillow<11.0,>=8.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (9.4.0)\n",

"Requirement already satisfied: pydantic>=2.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (2.6.4)\n",

"Collecting pydub (from gradio)\n",

" Downloading pydub-0.25.1-py2.py3-none-any.whl (32 kB)\n",

"Collecting python-multipart>=0.0.9 (from gradio)\n",

" Downloading python\_multipart-0.0.9-py3-none-any.whl (22 kB)\n",

"Requirement already satisfied: pyyaml<7.0,>=5.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (6.0.1)\n",

"Collecting ruff>=0.2.2 (from gradio)\n",

" Downloading ruff-0.3.7-py3-none-manylinux\_2\_17\_x86\_64.manylinux2014\_x86\_64.whl (8.9 MB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m8.9/8.9 MB\u001b[0m \u001b[31m33.8 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hCollecting semantic-version~=2.0 (from gradio)\n",

" Downloading semantic\_version-2.10.0-py2.py3-none-any.whl (15 kB)\n",

"Collecting tomlkit==0.12.0 (from gradio)\n",

" Downloading tomlkit-0.12.0-py3-none-any.whl (37 kB)\n",

"Requirement already satisfied: typer[all]<1.0,>=0.9 in /usr/local/lib/python3.10/dist-packages (from gradio) (0.9.4)\n",

"Requirement already satisfied: typing-extensions~=4.0 in /usr/local/lib/python3.10/dist-packages (from gradio) (4.11.0)\n",

"Collecting uvicorn>=0.14.0 (from gradio)\n",

" Downloading uvicorn-0.29.0-py3-none-any.whl (60 kB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m60.8/60.8 kB\u001b[0m \u001b[31m6.4 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hRequirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-packages (from gradio-client==0.15.1->gradio) (2023.6.0)\n",

"Collecting websockets<12.0,>=10.0 (from gradio-client==0.15.1->gradio)\n",

" Downloading websockets-11.0.3-cp310-cp310-manylinux\_2\_5\_x86\_64.manylinux1\_x86\_64.manylinux\_2\_17\_x86\_64.manylinux2014\_x86\_64.whl (129 kB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m129.9/129.9 kB\u001b[0m \u001b[31m8.4 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hRequirement already satisfied: entrypoints in /usr/local/lib/python3.10/dist-packages (from altair<6.0,>=4.2.0->gradio) (0.4)\n",

"Requirement already satisfied: jsonschema>=3.0 in /usr/local/lib/python3.10/dist-packages (from altair<6.0,>=4.2.0->gradio) (4.19.2)\n",

"Requirement already satisfied: toolz in /usr/local/lib/python3.10/dist-packages (from altair<6.0,>=4.2.0->gradio) (0.12.1)\n",

"Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages (from httpx>=0.24.1->gradio) (3.7.1)\n",

"Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-packages (from httpx>=0.24.1->gradio) (2024.2.2)\n",

"Collecting httpcore==1.\* (from httpx>=0.24.1->gradio)\n",

" Downloading httpcore-1.0.5-py3-none-any.whl (77 kB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m77.9/77.9 kB\u001b[0m \u001b[31m8.3 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hRequirement already satisfied: idna in /usr/local/lib/python3.10/dist-packages (from httpx>=0.24.1->gradio) (3.6)\n",

"Requirement already satisfied: sniffio in /usr/local/lib/python3.10/dist-packages (from httpx>=0.24.1->gradio) (1.3.1)\n",

"Collecting h11<0.15,>=0.13 (from httpcore==1.\*->httpx>=0.24.1->gradio)\n",

" Downloading h11-0.14.0-py3-none-any.whl (58 kB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m58.3/58.3 kB\u001b[0m \u001b[31m6.9 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hRequirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.19.3->gradio) (3.13.4)\n",

"Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.19.3->gradio) (2.31.0)\n",

"Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.19.3->gradio) (4.66.2)\n",

"Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (1.2.1)\n",

"Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (0.12.1)\n",

"Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (4.51.0)\n",

"Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (1.4.5)\n",

"Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (3.1.2)\n",

"Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio) (2.8.2)\n",

"Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas<3.0,>=1.0->gradio) (2023.4)\n",

"Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas<3.0,>=1.0->gradio) (2024.1)\n",

"Requirement already satisfied: annotated-types>=0.4.0 in /usr/local/lib/python3.10/dist-packages (from pydantic>=2.0->gradio) (0.6.0)\n",

"Requirement already satisfied: pydantic-core==2.16.3 in /usr/local/lib/python3.10/dist-packages (from pydantic>=2.0->gradio) (2.16.3)\n",

"Requirement already satisfied: click<9.0.0,>=7.1.1 in /usr/local/lib/python3.10/dist-packages (from typer[all]<1.0,>=0.9->gradio) (8.1.7)\n",

"Collecting colorama<0.5.0,>=0.4.3 (from typer[all]<1.0,>=0.9->gradio)\n",

" Downloading colorama-0.4.6-py2.py3-none-any.whl (25 kB)\n",

"Collecting shellingham<2.0.0,>=1.3.0 (from typer[all]<1.0,>=0.9->gradio)\n",

" Downloading shellingham-1.5.4-py2.py3-none-any.whl (9.8 kB)\n",

"Requirement already satisfied: rich<14.0.0,>=10.11.0 in /usr/local/lib/python3.10/dist-packages (from typer[all]<1.0,>=0.9->gradio) (13.7.1)\n",

"Collecting starlette<0.38.0,>=0.37.2 (from fastapi->gradio)\n",

" Downloading starlette-0.37.2-py3-none-any.whl (71 kB)\n",

"\u001b[2K \u001b[90m━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\u001b[0m \u001b[32m71.9/71.9 kB\u001b[0m \u001b[31m7.4 MB/s\u001b[0m eta \u001b[36m0:00:00\u001b[0m\n",

"\u001b[?25hRequirement already satisfied: attrs>=22.2.0 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=3.0->altair<6.0,>=4.2.0->gradio) (23.2.0)\n",

"Requirement already satisfied: jsonschema-specifications>=2023.03.6 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=3.0->altair<6.0,>=4.2.0->gradio) (2023.12.1)\n",

"Requirement already satisfied: referencing>=0.28.4 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=3.0->altair<6.0,>=4.2.0->gradio) (0.34.0)\n",

"Requirement already satisfied: rpds-py>=0.7.1 in /usr/local/lib/python3.10/dist-packages (from jsonschema>=3.0->altair<6.0,>=4.2.0->gradio) (0.18.0)\n",

"Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib~=3.0->gradio) (1.16.0)\n",

"Requirement already satisfied: markdown-it-py>=2.2.0 in /usr/local/lib/python3.10/dist-packages (from rich<14.0.0,>=10.11.0->typer[all]<1.0,>=0.9->gradio) (3.0.0)\n",

"Requirement already satisfied: pygments<3.0.0,>=2.13.0 in /usr/local/lib/python3.10/dist-packages (from rich<14.0.0,>=10.11.0->typer[all]<1.0,>=0.9->gradio) (2.16.1)\n",

"Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-packages (from anyio->httpx>=0.24.1->gradio) (1.2.0)\n",

"Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.19.3->gradio) (3.3.2)\n",

"Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.19.3->gradio) (2.0.7)\n",

"Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.10/dist-packages (from markdown-it-py>=2.2.0->rich<14.0.0,>=10.11.0->typer[all]<1.0,>=0.9->gradio) (0.1.2)\n",

"Building wheels for collected packages: ffmpy\n",

" Building wheel for ffmpy (setup.py) ... \u001b[?25l\u001b[?25hdone\n",

" Created wheel for ffmpy: filename=ffmpy-0.3.2-py3-none-any.whl size=5584 sha256=1d0db98c923beb65cdbf47d266ee6cfe90284105bd7f10b56adea6df5d0d2805\n",

" Stored in directory: /root/.cache/pip/wheels/bd/65/9a/671fc6dcde07d4418df0c592f8df512b26d7a0029c2a23dd81\n",

"Successfully built ffmpy\n",

"Installing collected packages: pydub, ffmpy, websockets, tomlkit, shellingham, semantic-version, ruff, python-multipart, orjson, h11, colorama, aiofiles, uvicorn, starlette, httpcore, httpx, fastapi, gradio-client, gradio\n",

"Successfully installed aiofiles-23.2.1 colorama-0.4.6 fastapi-0.110.1 ffmpy-0.3.2 gradio-4.26.0 gradio-client-0.15.1 h11-0.14.0 httpcore-1.0.5 httpx-0.27.0 orjson-3.10.0 pydub-0.25.1 python-multipart-0.0.9 ruff-0.3.7 semantic-version-2.10.0 shellingham-1.5.4 starlette-0.37.2 tomlkit-0.12.0 uvicorn-0.29.0 websockets-11.0.3\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"import gradio as gr\n",

"import joblib\n",

"# Load the trained model\n",

"model = joblib.load(\"loan\_classifier.joblib\")\n",

"\n",

"\n",

"def predict\_loan\_status(\n",

" int\_rate,\n",

" installment,\n",

" log\_annual\_inc,\n",

" dti,\n",

" fico,\n",

" revol\_bal,\n",

" revol\_util,\n",

" inq\_last\_6mths,\n",

" delinq\_2yrs,\n",

" pub\_rec,\n",

"\n",

"):\n",

" input\_dict = {\n",

" \"int.rate\": int\_rate,\n",

" \"installment\": installment,\n",

" \"log.annual.inc\": log\_annual\_inc,\n",

" \"dti\": dti,\n",

" \"fico\": fico,\n",

" \"revol.bal\": revol\_bal,\n",

" \"revol.util\": revol\_util,\n",

" \"inq.last.6mths\": inq\_last\_6mths,\n",

" \"delinq.2yrs\": delinq\_2yrs,\n",

" \"pub.rec\": pub\_rec,\n",

"\n",

" }\n",

" # Convert the dictionary to a 2D array\n",

" input\_array = [list(input\_dict.values())]\n",

" prediction = model.predict(input\_array)[0]\n",

"\n",

" if prediction == 0:\n",

" return \"Loan fully paid\"\n",

" else:\n",

" return \"Loan not fully paid\"\n",

"\n",

"\n",

"inputs = [\n",

" gr.Slider(0.06, 0.23, step=0.01, label=\"Interest Rate\"),\n",

" gr.Slider(100, 950, step=10, label=\"Installment\"),\n",

" gr.Slider(7, 15, step=0.1, label=\"Log Annual Income\"),\n",

" gr.Slider(0, 40, step=1, label=\"DTI Ratio\"),\n",

" gr.Slider(600, 850, step=1, label=\"FICO Score\"),\n",

" gr.Slider(0, 120000, step=1000, label=\"Revolving Balance\"),\n",

" gr.Slider(0, 120, step=1, label=\"Revolving Utilization\"),\n",

" gr.Slider(0, 10, step=1, label=\"Inquiries in Last 6 Months\"),\n",

" gr.Slider(0, 20, step=1, label=\"Delinquencies in Last 2 Years\"),\n",

" gr.Slider(0, 10, step=1, label=\"Public Records\"),\n",

"\n",

"]\n",

"outputs = [gr.Label(num\_top\_classes=2)]\n",

"\n",

"title = \"Loan Approval Classifier\"\n",

"description = (\n",

" \"Enter the details of the loan applicant to check if the loan is approved or not.\"\n",

")\n",

"gr.Interface(\n",

" fn=predict\_loan\_status,\n",

" inputs=inputs,\n",

" outputs=outputs,\n",

" title=title,\n",

" description=description,\n",

").launch()"

],

"metadata": {

"colab": {

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

"height": 646

},

"id": "YzHQntIqdXMm",

"outputId": "eddda13c-b65b-4271-896d-864e19db4320"

},

"execution\_count": 18,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"Setting queue=True in a Colab notebook requires sharing enabled. Setting `share=True` (you can turn this off by setting `share=False` in `launch()` explicitly).\n",

"\n",

"Colab notebook detected. To show errors in colab notebook, set debug=True in launch()\n",

"Running on public URL: https://1ff2cd0ec7a04f5a83.gradio.live\n",

"\n",

"This share link expires in 72 hours. For free permanent hosting and GPU upgrades, run `gradio deploy` from Terminal to deploy to Spaces (https://huggingface.co/spaces)\n"

]

},

{

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

"data": {

"text/plain": [

"<IPython.core.display.HTML object>"

],

"text/html": [

"<div><iframe src=\"https://1ff2cd0ec7a04f5a83.gradio.live\" width=\"100%\" height=\"500\" allow=\"autoplay; camera; microphone; clipboard-read; clipboard-write;\" frameborder=\"0\" allowfullscreen></iframe></div>"

]

},

"metadata": {}

},

{

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

"data": {

"text/plain": []

},

"metadata": {},

"execution\_count": 18

}

]

},

{

"source": [

"import pandas as pd"

],

"cell\_type": "code",

"metadata": {

"id": "aH980o7idz4D"

},

"execution\_count": 24,

"outputs": []

},

{

"source": [

"loan\_df = pd.read\_csv(\"/content/datacamp\_workspace\_export\_2024-04-01 19\_32\_28.csv\")"

],

"cell\_type": "code",

"metadata": {

"id": "ksMD1tEqd0Lk"

},

"execution\_count": 25,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"# Perform feature engineering\n",

"loan\_df[\"installment\_to\_income\_ratio\"] = (\n",

" loan\_df[\"installment\"] / loan\_df[\"log.annual.inc\"]\n",

")\n",

"loan\_df[\"credit\_history\"] = (loan\_df[\"delinq.2yrs\"] + loan\_df[\"pub.rec\"]) / loan\_df[\n",

" \"fico\"\n",

"]"

],

"metadata": {

"id": "cG\_0K17SeH2\_"

},

"execution\_count": 23,

"outputs": []

}

]

}