{

"nbformat": 4,

"nbformat\_minor": 0,

"metadata": {

"colab": {

"provenance": []

},

"kernelspec": {

"name": "python3",

"display\_name": "Python 3"

},

"language\_info": {

"name": "python"

}

},

"cells": [

{

"cell\_type": "code",

"execution\_count": 1,

"metadata": {

"colab": {

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

},

"id": "nTgu8ZrxF7Fq",

"outputId": "8ddc05aa-7249-4971-e95c-db761dd10804"

},

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/\n",

"Collecting scikit-plot\n",

" Downloading scikit\_plot-0.3.7-py3-none-any.whl (33 kB)\n",

"Requirement already satisfied: matplotlib>=1.4.0 in /usr/local/lib/python3.8/dist-packages (from scikit-plot) (3.2.2)\n",

"Requirement already satisfied: joblib>=0.10 in /usr/local/lib/python3.8/dist-packages (from scikit-plot) (1.2.0)\n",

"Requirement already satisfied: scipy>=0.9 in /usr/local/lib/python3.8/dist-packages (from scikit-plot) (1.7.3)\n",

"Requirement already satisfied: scikit-learn>=0.18 in /usr/local/lib/python3.8/dist-packages (from scikit-plot) (1.0.2)\n",

"Requirement already satisfied: numpy>=1.11 in /usr/local/lib/python3.8/dist-packages (from matplotlib>=1.4.0->scikit-plot) (1.21.6)\n",

"Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.8/dist-packages (from matplotlib>=1.4.0->scikit-plot) (3.0.9)\n",

"Requirement already satisfied: python-dateutil>=2.1 in /usr/local/lib/python3.8/dist-packages (from matplotlib>=1.4.0->scikit-plot) (2.8.2)\n",

"Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.8/dist-packages (from matplotlib>=1.4.0->scikit-plot) (1.4.4)\n",

"Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.8/dist-packages (from matplotlib>=1.4.0->scikit-plot) (0.11.0)\n",

"Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.8/dist-packages (from python-dateutil>=2.1->matplotlib>=1.4.0->scikit-plot) (1.15.0)\n",

"Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.8/dist-packages (from scikit-learn>=0.18->scikit-plot) (3.1.0)\n",

"Installing collected packages: scikit-plot\n",

"Successfully installed scikit-plot-0.3.7\n"

]

}

],

"source": [

"#install scikitplot\n",

"!pip install scikit-plot\n"

]

},

{

"cell\_type": "code",

"source": [

"from sklearn.preprocessing import LabelEncoder\n",

"import pandas as pd\n",

"\n",

"from sklearn.naive\_bayes import GaussianNB, BernoulliNB, MultinomialNB\n",

"from sklearn.svm import SVC, LinearSVC\n",

"from sklearn.linear\_model import LogisticRegression\n",

"from sklearn.tree import DecisionTreeClassifier\n",

"from sklearn.ensemble import RandomForestClassifier\n",

"from sklearn.neural\_network import MLPClassifier\n",

"\n",

"from sklearn.metrics import accuracy\_score, f1\_score\n",

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

"from sklearn import metrics, model\_selection\n",

"\n",

"import scikitplot as skplt"

],

"metadata": {

"id": "DlA\_KypuGEk2"

},

"execution\_count": 7,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"from google.colab import drive\n",

"drive.mount('/content/drive')\n",

"df = pd.read\_csv('/content/drive/My Drive/gender-prediction.csv')"

],

"metadata": {

"colab": {

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

},

"id": "2bUYUJJUGJ1r",

"outputId": "6dfdc451-1202-4713-fdd6-bc363d5ec6d3"

},

"execution\_count": 3,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"Mounted at /content/drive\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"New\_df = df"

],

"metadata": {

"id": "jlRzB6PnGWLc"

},

"execution\_count": 4,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"New\_df = New\_df.drop(['beard','hair\_length','scarf','eye\_color','gender'], axis=1)\n",

"New\_df"

],

"metadata": {

"colab": {

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

"height": 423

},

"id": "QGr1q6q1Gb3o",

"outputId": "09786d6b-3b58-4f66-e95a-dcf9703800d4"

},

"execution\_count": 5,

"outputs": [

{

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

"data": {

"text/plain": [

" height weight shoe\_size\n",

"0 71 176 44\n",

"1 68 165 41\n",

"2 62 132 37\n",

"3 65 138 38\n",

"4 70 197 43\n",

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

"75 65 99 39\n",

"76 61 98 37\n",

"77 67 119 40\n",

"78 70 190 43\n",

"79 62 142 37\n",

"\n",

"[80 rows x 3 columns]"

],

"text/html": [

"\n",

" <div id=\"df-dbf14c15-d806-4f25-963c-3ab887328e52\">\n",

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

" <div>\n",

"<style scoped>\n",

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

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

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

" <thead>\n",

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

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

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

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

" <td>71</td>\n",

" <td>176</td>\n",

" <td>44</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>68</td>\n",

" <td>165</td>\n",

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

" </tr>\n",

" <tr>\n",

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

" <td>62</td>\n",

" <td>132</td>\n",

" <td>37</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>65</td>\n",

" <td>138</td>\n",

" <td>38</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>70</td>\n",

" <td>197</td>\n",

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

" </tr>\n",

" <tr>\n",

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

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>65</td>\n",

" <td>99</td>\n",

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

" </tr>\n",

" <tr>\n",

" <th>76</th>\n",

" <td>61</td>\n",

" <td>98</td>\n",

" <td>37</td>\n",

" </tr>\n",

" <tr>\n",

" <th>77</th>\n",

" <td>67</td>\n",

" <td>119</td>\n",

" <td>40</td>\n",

" </tr>\n",

" <tr>\n",

" <th>78</th>\n",

" <td>70</td>\n",

" <td>190</td>\n",

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

" </tr>\n",

" <tr>\n",

" <th>79</th>\n",

" <td>62</td>\n",

" <td>142</td>\n",

" <td>37</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>80 rows × 3 columns</p>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-dbf14c15-d806-4f25-963c-3ab887328e52')\"\n",

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

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

" \n",

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

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

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

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

" </svg>\n",

" </button>\n",

" \n",

" <style>\n",

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

" display:flex;\n",

" flex-wrap:wrap;\n",

" gap: 12px;\n",

" }\n",

"\n",

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

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

" border: none;\n",

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

" cursor: pointer;\n",

" display: none;\n",

" fill: #1967D2;\n",

" height: 32px;\n",

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

" width: 32px;\n",

" }\n",

"\n",

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

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

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

" fill: #174EA6;\n",

" }\n",

"\n",

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

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

" fill: #D2E3FC;\n",

" }\n",

"\n",

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

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

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

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

" fill: #FFFFFF;\n",

" }\n",

" </style>\n",

"\n",

" <script>\n",

" const buttonEl =\n",

" document.querySelector('#df-dbf14c15-d806-4f25-963c-3ab887328e52 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-dbf14c15-d806-4f25-963c-3ab887328e52');\n",

" const dataTable =\n",

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

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

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

"\n",

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

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

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

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

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

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

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

" docLink.innerHTML = docLinkHtml;\n",

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

" }\n",

" </script>\n",

" </div>\n",

" </div>\n",

" "

]

},

"metadata": {},

"execution\_count": 5

}

]

},

{

"cell\_type": "code",

"source": [

"labels = LabelEncoder()\n",

"beard\_encoded = labels.fit\_transform(df['beard'])\n",

"hair\_length\_encoded = labels.fit\_transform(df['hair\_length'])\n",

"scarf\_encoded = labels.fit\_transform(df['scarf'])\n",

"eye\_color\_encoded = labels.fit\_transform(df['eye\_color'])\n",

"gender\_encoded = labels.fit\_transform(df['gender'])\n",

"\n",

"New\_df['beard'] = beard\_encoded\n",

"New\_df['hair\_length'] = hair\_length\_encoded\n",

"New\_df['scarf'] = scarf\_encoded\n",

"New\_df['eye\_color'] = eye\_color\_encoded\n",

"New\_df['gender'] = gender\_encoded\n",

"New\_df"

],

"metadata": {

"colab": {

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

"height": 423

},

"id": "WhTpxIv7Gisx",

"outputId": "a42cc697-7f4c-4c58-c2c6-8073555d70a4"

},

"execution\_count": 8,

"outputs": [

{

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

"data": {

"text/plain": [

" height weight shoe\_size beard hair\_length scarf eye\_color gender\n",

"0 71 176 44 1 3 0 0 1\n",

"1 68 165 41 0 0 0 0 1\n",

"2 62 132 37 0 2 1 1 0\n",

"3 65 138 38 0 1 0 3 0\n",

"4 70 197 43 1 2 0 3 1\n",

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

"75 65 99 39 0 3 1 4 0\n",

"76 61 98 37 0 3 0 2 0\n",

"77 67 119 40 1 3 0 0 1\n",

"78 70 190 43 1 2 0 3 1\n",

"79 62 142 37 1 1 0 1 0\n",

"\n",

"[80 rows x 8 columns]"

],

"text/html": [

"\n",

" <div id=\"df-063aedaa-1c0c-4843-bbb2-d8f813d1d6ec\">\n",

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

" <div>\n",

"<style scoped>\n",

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

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

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

" <thead>\n",

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

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

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

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

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

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

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

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

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

" <td>71</td>\n",

" <td>176</td>\n",

" <td>44</td>\n",

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>68</td>\n",

" <td>165</td>\n",

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>62</td>\n",

" <td>132</td>\n",

" <td>37</td>\n",

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>65</td>\n",

" <td>138</td>\n",

" <td>38</td>\n",

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>70</td>\n",

" <td>197</td>\n",

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

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

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

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

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

" <td>65</td>\n",

" <td>99</td>\n",

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

" <th>76</th>\n",

" <td>61</td>\n",

" <td>98</td>\n",

" <td>37</td>\n",

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

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

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

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

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

" </tr>\n",

" <tr>\n",

" <th>77</th>\n",

" <td>67</td>\n",

" <td>119</td>\n",

" <td>40</td>\n",

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

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

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

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

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

" </tr>\n",

" <tr>\n",

" <th>78</th>\n",

" <td>70</td>\n",

" <td>190</td>\n",

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

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

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

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

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

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

" </tr>\n",

" <tr>\n",

" <th>79</th>\n",

" <td>62</td>\n",

" <td>142</td>\n",

" <td>37</td>\n",

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

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

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

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

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

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>80 rows × 8 columns</p>\n",

"</div>\n",

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-063aedaa-1c0c-4843-bbb2-d8f813d1d6ec')\"\n",

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

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

" \n",

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

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

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

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

" </svg>\n",

" </button>\n",

" \n",

" <style>\n",

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

" display:flex;\n",

" flex-wrap:wrap;\n",

" gap: 12px;\n",

" }\n",

"\n",

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

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

" border: none;\n",

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

" cursor: pointer;\n",

" display: none;\n",

" fill: #1967D2;\n",

" height: 32px;\n",

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

" width: 32px;\n",

" }\n",

"\n",

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

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

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

" fill: #174EA6;\n",

" }\n",

"\n",

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

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

" fill: #D2E3FC;\n",

" }\n",

"\n",

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

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

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

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

" fill: #FFFFFF;\n",

" }\n",

" </style>\n",

"\n",

" <script>\n",

" const buttonEl =\n",

" document.querySelector('#df-063aedaa-1c0c-4843-bbb2-d8f813d1d6ec 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-063aedaa-1c0c-4843-bbb2-d8f813d1d6ec');\n",

" const dataTable =\n",

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

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

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

"\n",

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

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

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

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

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

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

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

" docLink.innerHTML = docLinkHtml;\n",

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

" }\n",

" </script>\n",

" </div>\n",

" </div>\n",

" "

]

},

"metadata": {},

"execution\_count": 8

}

]

},

{

"cell\_type": "code",

"source": [

"x = New\_df.drop('gender', axis=1)\n",

"y = New\_df['gender']"

],

"metadata": {

"id": "loS2GZP7G2TA"

},

"execution\_count": 9,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"from sklearn.model\_selection import ShuffleSplit,cross\_val\_score\n",

"\n",

"model = DecisionTreeClassifier()\n",

"\n",

"shuffle\_split=ShuffleSplit(test\_size=0.3,train\_size=0.5,n\_splits=10)\n",

"\n",

"scores=cross\_val\_score(model,x,y,cv=shuffle\_split)\n",

"\n",

"print(\"cross Validation scores:n {}\".format(scores))\n",

"print(\"Average Cross Validation score :{}\".format(scores.mean()))"

],

"metadata": {

"colab": {

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

},

"id": "KhhQufqeG78O",

"outputId": "8acad36a-20b2-4182-e355-9110f5918563"

},

"execution\_count": 10,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"cross Validation scores:n [0.95833333 1. 1. 0.91666667 1. 0.95833333\n",

" 0.91666667 0.875 0.875 0.95833333]\n",

"Average Cross Validation score :0.9458333333333334\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"from sklearn.model\_selection import LeavePOut,cross\_val\_score\n",

"\n",

"model = DecisionTreeClassifier()\n",

"\n",

"lpo = LeavePOut(p=1)\n",

"lpo.get\_n\_splits(x)\n",

"\n",

"scores=cross\_val\_score(model,x,y,cv=lpo)\n",

"\n",

"print(\"cross Validation scores:n {}\".format(scores))\n",

"print(\"Average Cross Validation score :{}\".format(scores.mean()))"

],

"metadata": {

"colab": {

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

},

"id": "n2x8lzL9HDMK",

"outputId": "4c8fcc7c-1e0a-4e8d-e552-4685fa512968"

},

"execution\_count": 11,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"cross Validation scores:n [1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 0. 1. 1. 1.\n",

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

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

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

"Average Cross Validation score :0.9375\n"

]

}

]

}

]

}