# -\*- coding: utf-8 -\*-

"""Untitled1.ipynb

Automatically generated by Colaboratory.

Original file is located at

https://colab.research.google.com/drive/1SOoyatCT0ep4QNIhloAz8eqwPg4V5YrZ

"""

{

"cells": [

{

"cell\_type": "markdown",

"metadata": {

"id": "view-in-github",

"colab\_type": "text"

},

"source": [

"<a href=\"https://colab.research.google.com/github/Siddhi026/ybi\_foundation/blob/main/Copy\_of\_Welcome\_To\_Colaboratory.ipynb\" target=\"\_parent\"><img src=\"https://colab.research.google.com/assets/colab-badge.svg\" alt=\"Open In Colab\"/></a>"

]

},

{

"cell\_type": "markdown",

"source": [

"# Mileage Prediction - Regression Analysis\n"

],

"metadata": {

"id": "WnDJoGlQmRio"

}

},

{

"cell\_type": "markdown",

"source": [

"# Source:\n",

"\n",

"This dataset was taken from the StatLib library which is maintained at Carnegie Mellon University. The dataset was used in the 1983 American Statistical Association Exposition.\n",

"\n",

"#Data Set Information:\n",

"\n",

"This dataset is a slightly modified version of the dataset provided in the StatLib library. In line with the use by Ross Quinlan (1993) in predicting the attribute 'mpg\", 8 of the original instances were removed because they had unknown values for the 'mpg\" attribute. The original dataset is available in the file \"auto-mpg.data-original\".\n",

"\n",

"\"The data concerns city-cycle fuel consumption in miles per gallon, to be predicted in terms of 3 multivalued discrete and 5 continuous\n",

"attributes.\" (Quinlan, 1993)\n",

"\n",

"#Attribute Information:\n",

"1. mpg: continuous\n",

"2. cylinders: multi-valued discrete\n",

"3. displacement: continuous\n",

"4. horsepower: continuous\n",

"5. weight: continuous\n",

"6. acceleration: continuous\n",

"7. model year: multi-valued discrete\n",

"8. origin: multi-valued discrete\n",

"9. car name: string (unique for each instance)"

],

"metadata": {

"id": "YMJpbX6xmSK2"

}

},

{

"cell\_type": "markdown",

"source": [

"# Import Data"

],

"metadata": {

"id": "ErQzkclUmTRD"

}

},

{

"cell\_type": "code",

"source": [

"import pandas as pd\n",

"\n",

"import numpy as np\n",

"\n",

"import matplotlib.pyplot as plt\n",

"\n",

"import seaborn as sns\n"

],

"metadata": {

"id": "-qWiF-bcmT2c"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"# Import Data"

],

"metadata": {

"id": "YLrt1j5DmWmd"

}

},

{

"cell\_type": "code",

"source": [

"df = pd.read\_csv('https://github.com/YBI-Foundation/Dataset/raw/main/MPG.csv')\n",

"\n",

"df.head()"

],

"metadata": {

"colab": {

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

"height": 206

},

"id": "cz6bJB-0mY8w",

"outputId": "8c4fdeda-7cb0-4154-ce08-45871d93e559"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

" mpg cylinders displacement horsepower weight acceleration \\\n",

"0 18.0 8 307.0 130.0 3504 12.0 \n",

"1 15.0 8 350.0 165.0 3693 11.5 \n",

"2 18.0 8 318.0 150.0 3436 11.0 \n",

"3 16.0 8 304.0 150.0 3433 12.0 \n",

"4 17.0 8 302.0 140.0 3449 10.5 \n",

"\n",

" model\_year origin name \n",

"0 70 usa chevrolet chevelle malibu \n",

"1 70 usa buick skylark 320 \n",

"2 70 usa plymouth satellite \n",

"3 70 usa amc rebel sst \n",

"4 70 usa ford torino "

],

"text/html": [

"\n",

" <div id=\"df-88a800d0-d5fd-488a-a078-09f7da1c62c7\" 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>mpg</th>\n",

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

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

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

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

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

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

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

" <td>18.0</td>\n",

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

" <td>307.0</td>\n",

" <td>130.0</td>\n",

" <td>3504</td>\n",

" <td>12.0</td>\n",

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

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

" <td>chevrolet chevelle malibu</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>15.0</td>\n",

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

" <td>350.0</td>\n",

" <td>165.0</td>\n",

" <td>3693</td>\n",

" <td>11.5</td>\n",

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

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

" <td>buick skylark 320</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>18.0</td>\n",

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

" <td>318.0</td>\n",

" <td>150.0</td>\n",

" <td>3436</td>\n",

" <td>11.0</td>\n",

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

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

" <td>plymouth satellite</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>16.0</td>\n",

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

" <td>304.0</td>\n",

" <td>150.0</td>\n",

" <td>3433</td>\n",

" <td>12.0</td>\n",

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

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

" <td>amc rebel sst</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>17.0</td>\n",

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

" <td>302.0</td>\n",

" <td>140.0</td>\n",

" <td>3449</td>\n",

" <td>10.5</td>\n",

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

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

" <td>ford torino</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-88a800d0-d5fd-488a-a078-09f7da1c62c7')\"\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-88a800d0-d5fd-488a-a078-09f7da1c62c7 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-88a800d0-d5fd-488a-a078-09f7da1c62c7');\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-83435564-6592-49c6-bdd6-9f57da3aad63\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-83435564-6592-49c6-bdd6-9f57da3aad63')\"\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-83435564-6592-49c6-bdd6-9f57da3aad63 button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

" </div>\n",

" </div>\n"

]

},

"metadata": {},

"execution\_count": 3

}

]

},

{

"cell\_type": "code",

"source": [

"df.nunique()"

],

"metadata": {

"colab": {

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

},

"id": "70MYdavHmZIy",

"outputId": "120997e0-ed5c-4515-84ed-b426347f31e9"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"mpg 129\n",

"cylinders 5\n",

"displacement 82\n",

"horsepower 93\n",

"weight 351\n",

"acceleration 95\n",

"model\_year 13\n",

"origin 3\n",

"name 305\n",

"dtype: int64"

]

},

"metadata": {},

"execution\_count": 4

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Data Preprocessing"

],

"metadata": {

"id": "SsCk4kFFmZte"

}

},

{

"cell\_type": "code",

"source": [

"df.info()"

],

"metadata": {

"colab": {

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

},

"id": "sWUHMvnimZ\_d",

"outputId": "86630ae9-389d-4bd6-8704-fac396ee23e8"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

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

"RangeIndex: 398 entries, 0 to 397\n",

"Data columns (total 9 columns):\n",

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

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

" 0 mpg 398 non-null float64\n",

" 1 cylinders 398 non-null int64 \n",

" 2 displacement 398 non-null float64\n",

" 3 horsepower 392 non-null float64\n",

" 4 weight 398 non-null int64 \n",

" 5 acceleration 398 non-null float64\n",

" 6 model\_year 398 non-null int64 \n",

" 7 origin 398 non-null object \n",

" 8 name 398 non-null object \n",

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

"memory usage: 28.1+ KB\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"df.describe()"

],

"metadata": {

"colab": {

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

"height": 300

},

"id": "XOFh6wK1maJl",

"outputId": "3ddcce31-ba8f-4166-962a-8821c89c7f12"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

" mpg cylinders displacement horsepower weight \\\n",

"count 398.000000 398.000000 398.000000 392.000000 398.000000 \n",

"mean 23.514573 5.454774 193.425879 104.469388 2970.424623 \n",

"std 7.815984 1.701004 104.269838 38.491160 846.841774 \n",

"min 9.000000 3.000000 68.000000 46.000000 1613.000000 \n",

"25% 17.500000 4.000000 104.250000 75.000000 2223.750000 \n",

"50% 23.000000 4.000000 148.500000 93.500000 2803.500000 \n",

"75% 29.000000 8.000000 262.000000 126.000000 3608.000000 \n",

"max 46.600000 8.000000 455.000000 230.000000 5140.000000 \n",

"\n",

" acceleration model\_year \n",

"count 398.000000 398.000000 \n",

"mean 15.568090 76.010050 \n",

"std 2.757689 3.697627 \n",

"min 8.000000 70.000000 \n",

"25% 13.825000 73.000000 \n",

"50% 15.500000 76.000000 \n",

"75% 17.175000 79.000000 \n",

"max 24.800000 82.000000 "

],

"text/html": [

"\n",

" <div id=\"df-a4864fc7-6a0b-453e-867d-84e1999ceb20\" 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>mpg</th>\n",

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

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

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

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

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

" <td>398.000000</td>\n",

" <td>398.000000</td>\n",

" <td>398.000000</td>\n",

" <td>392.000000</td>\n",

" <td>398.000000</td>\n",

" <td>398.000000</td>\n",

" <td>398.000000</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>23.514573</td>\n",

" <td>5.454774</td>\n",

" <td>193.425879</td>\n",

" <td>104.469388</td>\n",

" <td>2970.424623</td>\n",

" <td>15.568090</td>\n",

" <td>76.010050</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>7.815984</td>\n",

" <td>1.701004</td>\n",

" <td>104.269838</td>\n",

" <td>38.491160</td>\n",

" <td>846.841774</td>\n",

" <td>2.757689</td>\n",

" <td>3.697627</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>9.000000</td>\n",

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

" <td>68.000000</td>\n",

" <td>46.000000</td>\n",

" <td>1613.000000</td>\n",

" <td>8.000000</td>\n",

" <td>70.000000</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>17.500000</td>\n",

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

" <td>104.250000</td>\n",

" <td>75.000000</td>\n",

" <td>2223.750000</td>\n",

" <td>13.825000</td>\n",

" <td>73.000000</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>23.000000</td>\n",

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

" <td>148.500000</td>\n",

" <td>93.500000</td>\n",

" <td>2803.500000</td>\n",

" <td>15.500000</td>\n",

" <td>76.000000</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>29.000000</td>\n",

" <td>8.000000</td>\n",

" <td>262.000000</td>\n",

" <td>126.000000</td>\n",

" <td>3608.000000</td>\n",

" <td>17.175000</td>\n",

" <td>79.000000</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>46.600000</td>\n",

" <td>8.000000</td>\n",

" <td>455.000000</td>\n",

" <td>230.000000</td>\n",

" <td>5140.000000</td>\n",

" <td>24.800000</td>\n",

" <td>82.000000</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-a4864fc7-6a0b-453e-867d-84e1999ceb20')\"\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-a4864fc7-6a0b-453e-867d-84e1999ceb20 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-a4864fc7-6a0b-453e-867d-84e1999ceb20');\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-396a3e1a-7bfe-4ae7-bff6-32963adee36a\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-396a3e1a-7bfe-4ae7-bff6-32963adee36a')\"\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-396a3e1a-7bfe-4ae7-bff6-32963adee36a button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

" </div>\n",

" </div>\n"

]

},

"metadata": {},

"execution\_count": 6

}

]

},

{

"cell\_type": "code",

"source": [

"df.corr()"

],

"metadata": {

"colab": {

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

"height": 323

},

"id": "FE6wynEwmcFF",

"outputId": "ef7a5b3e-d7db-4080-9ede-c77e1af63fc3"

},

"execution\_count": null,

"outputs": [

{

"output\_type": "stream",

"name": "stderr",

"text": [

"<ipython-input-7-2f6f6606aa2c>:1: FutureWarning: The default value of numeric\_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric\_only to silence this warning.\n",

" df.corr()\n"

]

},

{

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

"data": {

"text/plain": [

" mpg cylinders displacement horsepower weight \\\n",

"mpg 1.000000 -0.775396 -0.804203 -0.778427 -0.831741 \n",

"cylinders -0.775396 1.000000 0.950721 0.842983 0.896017 \n",

"displacement -0.804203 0.950721 1.000000 0.897257 0.932824 \n",

"horsepower -0.778427 0.842983 0.897257 1.000000 0.864538 \n",

"weight -0.831741 0.896017 0.932824 0.864538 1.000000 \n",

"acceleration 0.420289 -0.505419 -0.543684 -0.689196 -0.417457 \n",

"model\_year 0.579267 -0.348746 -0.370164 -0.416361 -0.306564 \n",

"\n",

" acceleration model\_year \n",

"mpg 0.420289 0.579267 \n",

"cylinders -0.505419 -0.348746 \n",

"displacement -0.543684 -0.370164 \n",

"horsepower -0.689196 -0.416361 \n",

"weight -0.417457 -0.306564 \n",

"acceleration 1.000000 0.288137 \n",

"model\_year 0.288137 1.000000 "

],

"text/html": [

"\n",

" <div id=\"df-9f6a618d-e40a-497a-9988-e4492e102ba5\" 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>mpg</th>\n",

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

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

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

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

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

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

" <td>-0.775396</td>\n",

" <td>-0.804203</td>\n",

" <td>-0.778427</td>\n",

" <td>-0.831741</td>\n",

" <td>0.420289</td>\n",

" <td>0.579267</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>-0.775396</td>\n",

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

" <td>0.950721</td>\n",

" <td>0.842983</td>\n",

" <td>0.896017</td>\n",

" <td>-0.505419</td>\n",

" <td>-0.348746</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>-0.804203</td>\n",

" <td>0.950721</td>\n",

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

" <td>0.897257</td>\n",

" <td>0.932824</td>\n",

" <td>-0.543684</td>\n",

" <td>-0.370164</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>-0.778427</td>\n",

" <td>0.842983</td>\n",

" <td>0.897257</td>\n",

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

" <td>0.864538</td>\n",

" <td>-0.689196</td>\n",

" <td>-0.416361</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>-0.831741</td>\n",

" <td>0.896017</td>\n",

" <td>0.932824</td>\n",

" <td>0.864538</td>\n",

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

" <td>-0.417457</td>\n",

" <td>-0.306564</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>0.420289</td>\n",

" <td>-0.505419</td>\n",

" <td>-0.543684</td>\n",

" <td>-0.689196</td>\n",

" <td>-0.417457</td>\n",

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

" <td>0.288137</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>0.579267</td>\n",

" <td>-0.348746</td>\n",

" <td>-0.370164</td>\n",

" <td>-0.416361</td>\n",

" <td>-0.306564</td>\n",

" <td>0.288137</td>\n",

" <td>1.000000</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-9f6a618d-e40a-497a-9988-e4492e102ba5')\"\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-9f6a618d-e40a-497a-9988-e4492e102ba5 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-9f6a618d-e40a-497a-9988-e4492e102ba5');\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-9b6a5e91-fdcb-4699-8bcc-dcfc0f570b5b\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-9b6a5e91-fdcb-4699-8bcc-dcfc0f570b5b')\"\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-9b6a5e91-fdcb-4699-8bcc-dcfc0f570b5b button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

" </div>\n",

" </div>\n"

]

},

"metadata": {},

"execution\_count": 7

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Remove Missing Values"

],

"metadata": {

"id": "j7bwNUMomcf5"

}

},

{

"cell\_type": "code",

"source": [

"df = df.dropna()"

],

"metadata": {

"id": "C2jRNje\_mfLO"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"df.info"

],

"metadata": {

"colab": {

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

},

"id": "vuVao5b9mfT4",

"outputId": "fa972190-aef3-42ae-86b1-2fd8938185f6"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"<bound method DataFrame.info of mpg cylinders displacement horsepower weight acceleration \\\n",

"0 18.0 8 307.0 130.0 3504 12.0 \n",

"1 15.0 8 350.0 165.0 3693 11.5 \n",

"2 18.0 8 318.0 150.0 3436 11.0 \n",

"3 16.0 8 304.0 150.0 3433 12.0 \n",

"4 17.0 8 302.0 140.0 3449 10.5 \n",

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

"393 27.0 4 140.0 86.0 2790 15.6 \n",

"394 44.0 4 97.0 52.0 2130 24.6 \n",

"395 32.0 4 135.0 84.0 2295 11.6 \n",

"396 28.0 4 120.0 79.0 2625 18.6 \n",

"397 31.0 4 119.0 82.0 2720 19.4 \n",

"\n",

" model\_year origin name \n",

"0 70 usa chevrolet chevelle malibu \n",

"1 70 usa buick skylark 320 \n",

"2 70 usa plymouth satellite \n",

"3 70 usa amc rebel sst \n",

"4 70 usa ford torino \n",

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

"393 82 usa ford mustang gl \n",

"394 82 europe vw pickup \n",

"395 82 usa dodge rampage \n",

"396 82 usa ford ranger \n",

"397 82 usa chevy s-10 \n",

"\n",

"[392 rows x 9 columns]>"

]

},

"metadata": {},

"execution\_count": 9

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Data Visualization"

],

"metadata": {

"id": "GHCOQ5VEomEM"

}

},

{

"cell\_type": "code",

"source": [

"sns.pairplot(df, x\_vars= ['displacement', 'horsepower', 'weight', 'acceleration', 'mpg'], y\_vars=['mpg']);"

],

"metadata": {

"colab": {

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

"height": 268

},

"id": "2U5qOuTamfc4",

"outputId": "6df29e42-09a5-4bd3-c936-1114aabe547d"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"<Figure size 1250x250 with 6 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"cell\_type": "code",

"source": [

"sns.regplot(x = 'displacement', y = 'mpg', data = df);"

],

"metadata": {

"colab": {

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

"height": 449

},

"id": "5iXQT\_F1mfoj",

"outputId": "4ac919be-1a88-4cbc-fe43-fe3ad00a9e37"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"<Figure size 640x480 with 1 Axes>"

],

"image/png": "\n"

},

"metadata": {}

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Define Target Variable y and Feature X"

],

"metadata": {

"id": "016hOlkAot-p"

}

},

{

"cell\_type": "code",

"source": [

"df.columns"

],

"metadata": {

"colab": {

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

},

"id": "tJ-saFysmfy7",

"outputId": "6841eeaa-27db-4b15-ffe3-56820332d052"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"Index(['mpg', 'cylinders', 'displacement', 'horsepower', 'weight',\n",

" 'acceleration', 'model\_year', 'origin', 'name'],\n",

" dtype='object')"

]

},

"metadata": {},

"execution\_count": 12

}

]

},

{

"cell\_type": "code",

"source": [

"y = df['mpg']"

],

"metadata": {

"id": "-AWduuOpmf65"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"y.shape"

],

"metadata": {

"colab": {

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

},

"id": "NNe2Zh7zmjEh",

"outputId": "4041dbfc-b037-472a-8f44-4b9b152a8fd4"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"(392,)"

]

},

"metadata": {},

"execution\_count": 14

}

]

},

{

"cell\_type": "code",

"source": [

"X = df[['displacement', 'horsepower', 'weight', 'acceleration']]"

],

"metadata": {

"id": "KHpL-4tdmjOp"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X.shape"

],

"metadata": {

"colab": {

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

},

"id": "NVV82VaQmjXZ",

"outputId": "c6564e62-9d13-426d-8016-ca410c7246c5"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"(392, 4)"

]

},

"metadata": {},

"execution\_count": 16

}

]

},

{

"cell\_type": "code",

"source": [

"X"

],

"metadata": {

"colab": {

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

"height": 423

},

"id": "YYyIlz8Pmjhq",

"outputId": "709cced3-1721-4bf5-d694-3a961897054f"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

" displacement horsepower weight acceleration\n",

"0 307.0 130.0 3504 12.0\n",

"1 350.0 165.0 3693 11.5\n",

"2 318.0 150.0 3436 11.0\n",

"3 304.0 150.0 3433 12.0\n",

"4 302.0 140.0 3449 10.5\n",

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

"393 140.0 86.0 2790 15.6\n",

"394 97.0 52.0 2130 24.6\n",

"395 135.0 84.0 2295 11.6\n",

"396 120.0 79.0 2625 18.6\n",

"397 119.0 82.0 2720 19.4\n",

"\n",

"[392 rows x 4 columns]"

],

"text/html": [

"\n",

" <div id=\"df-9c290ef3-51d7-40e7-962c-55c6b3894d7d\" 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>displacement</th>\n",

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

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

" <td>307.0</td>\n",

" <td>130.0</td>\n",

" <td>3504</td>\n",

" <td>12.0</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>350.0</td>\n",

" <td>165.0</td>\n",

" <td>3693</td>\n",

" <td>11.5</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>318.0</td>\n",

" <td>150.0</td>\n",

" <td>3436</td>\n",

" <td>11.0</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>304.0</td>\n",

" <td>150.0</td>\n",

" <td>3433</td>\n",

" <td>12.0</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>302.0</td>\n",

" <td>140.0</td>\n",

" <td>3449</td>\n",

" <td>10.5</td>\n",

" </tr>\n",

" <tr>\n",

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

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

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

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

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

" </tr>\n",

" <tr>\n",

" <th>393</th>\n",

" <td>140.0</td>\n",

" <td>86.0</td>\n",

" <td>2790</td>\n",

" <td>15.6</td>\n",

" </tr>\n",

" <tr>\n",

" <th>394</th>\n",

" <td>97.0</td>\n",

" <td>52.0</td>\n",

" <td>2130</td>\n",

" <td>24.6</td>\n",

" </tr>\n",

" <tr>\n",

" <th>395</th>\n",

" <td>135.0</td>\n",

" <td>84.0</td>\n",

" <td>2295</td>\n",

" <td>11.6</td>\n",

" </tr>\n",

" <tr>\n",

" <th>396</th>\n",

" <td>120.0</td>\n",

" <td>79.0</td>\n",

" <td>2625</td>\n",

" <td>18.6</td>\n",

" </tr>\n",

" <tr>\n",

" <th>397</th>\n",

" <td>119.0</td>\n",

" <td>82.0</td>\n",

" <td>2720</td>\n",

" <td>19.4</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"<p>392 rows × 4 columns</p>\n",

"</div>\n",

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

"\n",

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

" <button class=\"colab-df-convert\" onclick=\"convertToInteractive('df-9c290ef3-51d7-40e7-962c-55c6b3894d7d')\"\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-9c290ef3-51d7-40e7-962c-55c6b3894d7d 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-9c290ef3-51d7-40e7-962c-55c6b3894d7d');\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-c4544cf4-6f0c-48e1-a9f8-f75f447dc759\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-c4544cf4-6f0c-48e1-a9f8-f75f447dc759')\"\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-c4544cf4-6f0c-48e1-a9f8-f75f447dc759 button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

" </div>\n",

" </div>\n"

]

},

"metadata": {},

"execution\_count": 17

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Scalling Data"

],

"metadata": {

"id": "zx9CZzIImkhN"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.preprocessing import StandardScaler"

],

"metadata": {

"id": "eZV5Up9\_mmah"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"ss = StandardScaler()"

],

"metadata": {

"id": "94jYLAzCmmjU"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X = ss.fit\_transform(X)"

],

"metadata": {

"id": "fb3LRlrUmmvh"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X"

],

"metadata": {

"colab": {

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

},

"id": "uxQ-Ot8omp3F",

"outputId": "e344044e-95e3-4cb9-e779-afbf45a26560"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"array([[ 1.07728956, 0.66413273, 0.62054034, -1.285258 ],\n",

" [ 1.48873169, 1.57459447, 0.84333403, -1.46672362],\n",

" [ 1.1825422 , 1.18439658, 0.54038176, -1.64818924],\n",

" ...,\n",

" [-0.56847897, -0.53247413, -0.80463202, -1.4304305 ],\n",

" [-0.7120053 , -0.66254009, -0.41562716, 1.11008813],\n",

" [-0.72157372, -0.58450051, -0.30364091, 1.40043312]])"

]

},

"metadata": {},

"execution\_count": 21

}

]

},

{

"cell\_type": "code",

"source": [

"pd.DataFrame(X).describe()"

],

"metadata": {

"colab": {

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

"height": 300

},

"id": "AufpCNivmqCA",

"outputId": "88fcf804-e583-4a03-8591-89a6cf7fdf48"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

" 0 1 2 3\n",

"count 3.920000e+02 3.920000e+02 3.920000e+02 3.920000e+02\n",

"mean -7.250436e-17 -1.812609e-16 -1.812609e-17 4.350262e-16\n",

"std 1.001278e+00 1.001278e+00 1.001278e+00 1.001278e+00\n",

"min -1.209563e+00 -1.520975e+00 -1.608575e+00 -2.736983e+00\n",

"25% -8.555316e-01 -7.665929e-01 -8.868535e-01 -6.410551e-01\n",

"50% -4.153842e-01 -2.853488e-01 -2.052109e-01 -1.499869e-02\n",

"75% 7.782764e-01 5.600800e-01 7.510927e-01 5.384714e-01\n",

"max 2.493416e+00 3.265452e+00 2.549061e+00 3.360262e+00"

],

"text/html": [

"\n",

" <div id=\"df-f0c76863-81f1-427e-858e-83a11150d422\" 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>0</th>\n",

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

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

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

" <td>3.920000e+02</td>\n",

" <td>3.920000e+02</td>\n",

" <td>3.920000e+02</td>\n",

" <td>3.920000e+02</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>-7.250436e-17</td>\n",

" <td>-1.812609e-16</td>\n",

" <td>-1.812609e-17</td>\n",

" <td>4.350262e-16</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>1.001278e+00</td>\n",

" <td>1.001278e+00</td>\n",

" <td>1.001278e+00</td>\n",

" <td>1.001278e+00</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>-1.209563e+00</td>\n",

" <td>-1.520975e+00</td>\n",

" <td>-1.608575e+00</td>\n",

" <td>-2.736983e+00</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>-8.555316e-01</td>\n",

" <td>-7.665929e-01</td>\n",

" <td>-8.868535e-01</td>\n",

" <td>-6.410551e-01</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>-4.153842e-01</td>\n",

" <td>-2.853488e-01</td>\n",

" <td>-2.052109e-01</td>\n",

" <td>-1.499869e-02</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>7.782764e-01</td>\n",

" <td>5.600800e-01</td>\n",

" <td>7.510927e-01</td>\n",

" <td>5.384714e-01</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>2.493416e+00</td>\n",

" <td>3.265452e+00</td>\n",

" <td>2.549061e+00</td>\n",

" <td>3.360262e+00</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-f0c76863-81f1-427e-858e-83a11150d422')\"\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-f0c76863-81f1-427e-858e-83a11150d422 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-f0c76863-81f1-427e-858e-83a11150d422');\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-4c7012ef-c63e-4962-98b2-8e21c2357963\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-4c7012ef-c63e-4962-98b2-8e21c2357963')\"\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-4c7012ef-c63e-4962-98b2-8e21c2357963 button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

" </div>\n",

" </div>\n"

]

},

"metadata": {},

"execution\_count": 22

}

]

},

{

"cell\_type": "markdown",

"source": [

"\*\*After Standardization Mean is Zero and Standard Deviation is One\*\*"

],

"metadata": {

"id": "ynOnTP0Qmq9d"

}

},

{

"cell\_type": "markdown",

"source": [

"# Train Test Split Data"

],

"metadata": {

"id": "ajNyvAfFmsfF"

}

},

{

"cell\_type": "code",

"source": [

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

],

"metadata": {

"id": "04lYsw1vmtby"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, train\_size = 0.7, random\_state = 2529)"

],

"metadata": {

"id": "whz-5TH9mtlx"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X\_train.shape, X\_test.shape, y\_train.shape, y\_test.shape"

],

"metadata": {

"colab": {

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

},

"id": "hOmyNsKkpk6i",

"outputId": "86e747bd-0faa-4483-9ae5-58d043f1a882"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"((274, 4), (118, 4), (274,), (118,))"

]

},

"metadata": {},

"execution\_count": 25

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Linear Regression Model"

],

"metadata": {

"id": "RK8-ZzqamuuA"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.linear\_model import LinearRegression"

],

"metadata": {

"id": "qgaTopiSmv7h"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"lr = LinearRegression()"

],

"metadata": {

"id": "UWUe7SGmmwEL"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

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

],

"metadata": {

"colab": {

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

"height": 74

},

"id": "WvCpi6u2mwOh",

"outputId": "d58deccb-0a42-4d98-f1a2-c28a25521c42"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"LinearRegression()"

],

"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>LinearRegression()</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\">LinearRegression</label><div class=\"sk-toggleable\_\_content\"><pre>LinearRegression()</pre></div></div></div></div></div>"

]

},

"metadata": {},

"execution\_count": 28

}

]

},

{

"cell\_type": "code",

"source": [

"lr.intercept\_"

],

"metadata": {

"colab": {

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

},

"id": "CsFhIZFQmwZb",

"outputId": "d2a08e20-86a1-485a-a2dd-0a3413548b45"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"23.485738559737584"

]

},

"metadata": {},

"execution\_count": 29

}

]

},

{

"cell\_type": "code",

"source": [

"lr.coef\_"

],

"metadata": {

"colab": {

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

},

"id": "ebCeTjoAp7-U",

"outputId": "67573a35-6933-4bdf-b056-68348dfa15c1"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"array([-1.05767743, -1.68734727, -4.10787617, -0.11495177])"

]

},

"metadata": {},

"execution\_count": 30

}

]

},

{

"cell\_type": "markdown",

"source": [

"\*\*Mileage = 23.4 - 1.05Displacemet - 1.68Horsepower - 4.10Weight - 0.115Acceleration + error\*\*"

],

"metadata": {

"id": "5y\_pEHDVmxG9"

}

},

{

"cell\_type": "markdown",

"source": [

"# Predict Test Data"

],

"metadata": {

"id": "hMD7NBKNqGb0"

}

},

{

"cell\_type": "code",

"source": [

"y\_pred = lr.predict(X\_test)"

],

"metadata": {

"id": "XQQakTkxmzSE"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"y\_pred"

],

"metadata": {

"colab": {

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

},

"id": "luBwUXqUm0dr",

"outputId": "31b5c82f-dddd-4634-8114-8b2864d58256"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"array([18.51865637, 15.09305675, 14.30128789, 23.6753321 , 29.7546115 ,\n",

" 23.68796629, 26.61066644, 24.56692437, 15.06260986, 11.94312046,\n",

" 24.08050053, 27.96518468, 31.66130278, 31.01309132, 18.32428976,\n",

" 19.32795009, 28.08847536, 32.1506879 , 31.15859692, 27.15792144,\n",

" 18.82433097, 22.54580176, 26.15598115, 32.36393869, 20.74377679,\n",

" 8.78027518, 22.19699435, 18.20614294, 25.00052718, 15.26421552,\n",

" 23.13441082, 17.10542257, 9.87180062, 30.00790415, 20.41204655,\n",

" 29.11860245, 24.4305187 , 21.72601835, 10.51174626, 13.12426391,\n",

" 21.41938406, 19.96113872, 6.19146626, 17.79025345, 22.5493033 ,\n",

" 29.34765021, 13.4861847 , 25.88852083, 29.40406946, 22.41841964,\n",

" 22.07684766, 16.46575802, 24.06290693, 30.12890046, 10.11318121,\n",

" 9.85011438, 28.07543852, 23.41426617, 20.08501128, 30.68234133,\n",

" 20.92026393, 26.78370281, 22.9078744 , 14.15936872, 24.6439883 ,\n",

" 26.95515832, 15.25709393, 24.11272087, 30.80980589, 14.9770217 ,\n",

" 27.67836372, 24.2372919 , 10.92177228, 30.22858779, 30.88687365,\n",

" 27.33992044, 31.18447082, 10.8873597 , 27.63510608, 16.49231363,\n",

" 25.63229888, 29.49776285, 14.90393439, 32.78670687, 30.37325244,\n",

" 30.9262743 , 14.71702373, 27.09633246, 26.69933806, 29.06424799,\n",

" 32.45810182, 29.44846898, 31.61239999, 31.57891837, 21.46542321,\n",

" 31.76739191, 26.28605476, 28.96419915, 31.09628395, 24.80549594,\n",

" 18.76490961, 23.28043777, 23.04466919, 22.14143162, 15.95854367,\n",

" 28.62870918, 25.58809869, 11.4040908 , 25.73334842, 30.83500051,\n",

" 21.94176255, 15.34532941, 30.37399213, 28.7620624 , 29.3639931 ,\n",

" 29.10476703, 20.44662365, 28.11466839])"

]

},

"metadata": {},

"execution\_count": 34

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Model Accuracy"

],

"metadata": {

"id": "WdM8rsr2qOnC"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.metrics import mean\_absolute\_error, mean\_absolute\_percentage\_error, r2\_score"

],

"metadata": {

"id": "b8wnfPO1m0oo"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"mean\_absolute\_error(y\_test, y\_pred)"

],

"metadata": {

"colab": {

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

},

"id": "KmrmA9A-m0wS",

"outputId": "2483cec0-fbc9-43aa-a6a2-fdc1bd68149d"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"3.3286968643244106"

]

},

"metadata": {},

"execution\_count": 36

}

]

},

{

"cell\_type": "code",

"source": [

"mean\_absolute\_percentage\_error(y\_test, y\_pred)"

],

"metadata": {

"colab": {

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

},

"id": "qiBh2A3Zm07V",

"outputId": "9a0c2e8e-e0e0-43cf-c5b9-d3dd7ad261bd"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"0.14713035779536746"

]

},

"metadata": {},

"execution\_count": 37

}

]

},

{

"cell\_type": "code",

"source": [

"r2\_score(y\_test, y\_pred)"

],

"metadata": {

"colab": {

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

},

"id": "w00nMbesm1Ey",

"outputId": "ce9ad344-286c-49d1-ef6b-0dce8ba35c86"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"0.7031250746717691"

]

},

"metadata": {},

"execution\_count": 38

}

]

},

{

"cell\_type": "markdown",

"source": [

"# Polynomial Regression"

],

"metadata": {

"id": "o\_jQZNZtqsTY"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.preprocessing import PolynomialFeatures"

],

"metadata": {

"id": "GIObIx\_gqsxQ"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"poly = PolynomialFeatures (degree=2, interaction\_only=True, include\_bias=False)"

],

"metadata": {

"id": "pMpeGI-zqtEN"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X\_train2 = poly.fit\_transform(X\_train)"

],

"metadata": {

"id": "rhCzgCMMqtZt"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"X\_test2 = poly.fit\_transform(X\_test)"

],

"metadata": {

"id": "i841ljXwqtsn"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"lr.fit(X\_train2, y\_train)"

],

"metadata": {

"colab": {

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

"height": 74

},

"id": "yLE3f-9sqt4l",

"outputId": "ef533371-f2c7-4ca4-ccc7-877dc96fd467"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"LinearRegression()"

],

"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>LinearRegression()</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\">LinearRegression</label><div class=\"sk-toggleable\_\_content\"><pre>LinearRegression()</pre></div></div></div></div></div>"

]

},

"metadata": {},

"execution\_count": 44

}

]

},

{

"cell\_type": "code",

"source": [

"lr.intercept\_"

],

"metadata": {

"colab": {

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

},

"id": "lCKf-mrrqu1B",

"outputId": "08c5cfab-5442-4867-d392-de8ca57a42e7"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"21.27336450063766"

]

},

"metadata": {},

"execution\_count": 45

}

]

},

{

"cell\_type": "code",

"source": [

"lr.coef\_"

],

"metadata": {

"colab": {

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

},

"id": "QgTG9ceIqvAb",

"outputId": "a5092cc0-f12f-4f6f-b323-072aefdc0963"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"array([-2.76070596, -5.00559628, -1.36884133, -0.81225214, 1.24596571,\n",

" -0.12475017, -0.90542822, 1.35064048, -0.17337823, 1.41680398])"

]

},

"metadata": {},

"execution\_count": 46

}

]

},

{

"cell\_type": "code",

"source": [

"y\_pred\_poly = lr.predict(X\_test2)"

],

"metadata": {

"id": "bFO7mWLHqvKg"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"# Model Accuracy"

],

"metadata": {

"id": "hYYaAGJ3qwKz"

}

},

{

"cell\_type": "code",

"source": [

"from sklearn.metrics import mean\_absolute\_error, mean\_absolute\_percentage\_error, r2\_score"

],

"metadata": {

"id": "RvX7q3B4qw1\_"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"mean\_absolute\_error(y\_test, y\_pred\_poly)"

],

"metadata": {

"colab": {

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

},

"id": "b5RD246Uqw9F",

"outputId": "81724f64-57bf-4f18-caec-5e14d741d7e3"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"2.7887147720295977"

]

},

"metadata": {},

"execution\_count": 49

}

]

},

{

"cell\_type": "code",

"source": [

"mean\_absolute\_percentage\_error(y\_test, y\_pred\_poly)"

],

"metadata": {

"colab": {

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

},

"id": "DEtGOG54qxIE",

"outputId": "95d1608c-99cb-4700-9146-55abc41f04fb"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"0.12074018342938687"

]

},

"metadata": {},

"execution\_count": 50

}

]

},

{

"cell\_type": "code",

"source": [

"r2\_score(y\_test, y\_pred\_poly)"

],

"metadata": {

"colab": {

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

},

"id": "Sv2-rcYfqxVO",

"outputId": "ce05966c-7c92-46cb-ca8a-9f39a785693c"

},

"execution\_count": null,

"outputs": [

{

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

"data": {

"text/plain": [

"0.7461731314563803"

]

},

"metadata": {},

"execution\_count": 51

}

]

}

],

"metadata": {

"colab": {

"provenance": [],

"include\_colab\_link": true

},

"kernelspec": {

"display\_name": "Python 3",

"name": "python3"

}

},

"nbformat": 4,

"nbformat\_minor": 0

}