{

"nbformat": 4,

"nbformat\_minor": 0,

"metadata": {

"colab": {

"provenance": []

},

"kernelspec": {

"name": "python3",

"display\_name": "Python 3"

},

"language\_info": {

"name": "python"

}

},

"cells": [

{

"cell\_type": "code",

"source": [

"import cv2\n",

"import csv\n",

"import pandas as pd\n",

"import numpy as np\n",

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

],

"metadata": {

"id": "XQ2xh7HyQ5Nn"

},

"execution\_count": 21,

"outputs": []

},

{

"cell\_type": "code",

"execution\_count": 54,

"metadata": {

"id": "LwutIB3TP8pQ"

},

"outputs": [],

"source": [

"data = pd.read\_csv('/content/drive/MyDrive/real estate.csv')"

]

},

{

"cell\_type": "code",

"source": [

"data.head()"

],

"metadata": {

"colab": {

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

"height": 224

},

"id": "hshrqmIoReFo",

"outputId": "20f3f0a7-9d6a-4299-8645-c119f06cef5a"

},

"execution\_count": 55,

"outputs": [

{

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

"data": {

"text/plain": [

" No X1 transaction date X2 house age \\\n",

"0 1 2012.917 32.0 \n",

"1 2 2012.917 19.5 \n",

"2 3 2013.583 13.3 \n",

"3 4 2013.500 13.3 \n",

"4 5 2012.833 5.0 \n",

"\n",

" X3 distance to the nearest MRT station X4 number of convenience stores \\\n",

"0 84.87882 10 \n",

"1 306.59470 9 \n",

"2 561.98450 5 \n",

"3 561.98450 5 \n",

"4 390.56840 5 \n",

"\n",

" X5 latitude X6 longitude Y house price of unit area \n",

"0 24.98298 121.54024 37.9 \n",

"1 24.98034 121.53951 42.2 \n",

"2 24.98746 121.54391 47.3 \n",

"3 24.98746 121.54391 54.8 \n",

"4 24.97937 121.54245 43.1 "

],

"text/html": [

"\n",

" <div id=\"df-51a8d6e3-2292-414d-a264-7dc7f65d9b2e\" 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>No</th>\n",

" <th>X1 transaction date</th>\n",

" <th>X2 house age</th>\n",

" <th>X3 distance to the nearest MRT station</th>\n",

" <th>X4 number of convenience stores</th>\n",

" <th>X5 latitude</th>\n",

" <th>X6 longitude</th>\n",

" <th>Y house price of unit area</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

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

" <td>2012.917</td>\n",

" <td>32.0</td>\n",

" <td>84.87882</td>\n",

" <td>10</td>\n",

" <td>24.98298</td>\n",

" <td>121.54024</td>\n",

" <td>37.9</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>2012.917</td>\n",

" <td>19.5</td>\n",

" <td>306.59470</td>\n",

" <td>9</td>\n",

" <td>24.98034</td>\n",

" <td>121.53951</td>\n",

" <td>42.2</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>2013.583</td>\n",

" <td>13.3</td>\n",

" <td>561.98450</td>\n",

" <td>5</td>\n",

" <td>24.98746</td>\n",

" <td>121.54391</td>\n",

" <td>47.3</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>2013.500</td>\n",

" <td>13.3</td>\n",

" <td>561.98450</td>\n",

" <td>5</td>\n",

" <td>24.98746</td>\n",

" <td>121.54391</td>\n",

" <td>54.8</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>5</td>\n",

" <td>2012.833</td>\n",

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

" <td>390.56840</td>\n",

" <td>5</td>\n",

" <td>24.97937</td>\n",

" <td>121.54245</td>\n",

" <td>43.1</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-51a8d6e3-2292-414d-a264-7dc7f65d9b2e')\"\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-51a8d6e3-2292-414d-a264-7dc7f65d9b2e 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-51a8d6e3-2292-414d-a264-7dc7f65d9b2e');\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-73d17925-bf91-4156-8e8b-f7b02cdff64e\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-73d17925-bf91-4156-8e8b-f7b02cdff64e')\"\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-73d17925-bf91-4156-8e8b-f7b02cdff64e button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

"\n",

" </div>\n",

" </div>\n"

],

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

"type": "dataframe",

"variable\_name": "data",

"summary": "{\n \"name\": \"data\",\n \"rows\": 414,\n \"fields\": [\n {\n \"column\": \"No\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 119,\n \"min\": 1,\n \"max\": 414,\n \"num\_unique\_values\": 414,\n \"samples\": [\n 359,\n 351,\n 374\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X1 transaction date\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.2819672402630115,\n \"min\": 2012.667,\n \"max\": 2013.583,\n \"num\_unique\_values\": 12,\n \"samples\": [\n 2013.0,\n 2012.75,\n 2012.917\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X2 house age\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 11.39248453324253,\n \"min\": 0.0,\n \"max\": 43.8,\n \"num\_unique\_values\": 236,\n \"samples\": [\n 38.2,\n 3.4,\n 27.5\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X3 distance to the nearest MRT station\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 1262.1095954078512,\n \"min\": 23.38284,\n \"max\": 6488.021,\n \"num\_unique\_values\": 259,\n \"samples\": [\n 552.4371,\n 4605.749,\n 405.2134\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X4 number of convenience stores\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 2,\n \"min\": 0,\n \"max\": 10,\n \"num\_unique\_values\": 11,\n \"samples\": [\n 6,\n 10,\n 8\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X5 latitude\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.012410196590450338,\n \"min\": 24.93207,\n \"max\": 25.01459,\n \"num\_unique\_values\": 234,\n \"samples\": [\n 24.98573,\n 24.96143,\n 24.96696\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X6 longitude\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.015347183004592205,\n \"min\": 121.47353,\n \"max\": 121.56627,\n \"num\_unique\_values\": 232,\n \"samples\": [\n 121.54634,\n 121.50831,\n 121.53372\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"Y house price of unit area\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 13.606487697735314,\n \"min\": 7.6,\n \"max\": 117.5,\n \"num\_unique\_values\": 270,\n \"samples\": [\n 25.0,\n 30.7,\n 18.3\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}"

}

},

"metadata": {},

"execution\_count": 55

}

]

},

{

"cell\_type": "code",

"source": [

"#check for null values\n",

"data.isna().sum()"

],

"metadata": {

"colab": {

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

},

"id": "81M2qTQORrOH",

"outputId": "3feac118-518c-46f8-c8d0-9b2cc8636e3e"

},

"execution\_count": 56,

"outputs": [

{

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

"data": {

"text/plain": [

"No 0\n",

"X1 transaction date 0\n",

"X2 house age 0\n",

"X3 distance to the nearest MRT station 0\n",

"X4 number of convenience stores 0\n",

"X5 latitude 0\n",

"X6 longitude 0\n",

"Y house price of unit area 0\n",

"dtype: int64"

]

},

"metadata": {},

"execution\_count": 56

}

]

},

{

"cell\_type": "code",

"source": [

"data.info()"

],

"metadata": {

"colab": {

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

},

"id": "zZQQ0EBPSLjd",

"outputId": "2a6dd104-d5ff-4578-8130-e70acfe94c2b"

},

"execution\_count": 57,

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

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

"RangeIndex: 414 entries, 0 to 413\n",

"Data columns (total 8 columns):\n",

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

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

" 0 No 414 non-null int64 \n",

" 1 X1 transaction date 414 non-null float64\n",

" 2 X2 house age 414 non-null float64\n",

" 3 X3 distance to the nearest MRT station 414 non-null float64\n",

" 4 X4 number of convenience stores 414 non-null int64 \n",

" 5 X5 latitude 414 non-null float64\n",

" 6 X6 longitude 414 non-null float64\n",

" 7 Y house price of unit area 414 non-null float64\n",

"dtypes: float64(6), int64(2)\n",

"memory usage: 26.0 KB\n"

]

}

]

},

{

"cell\_type": "code",

"source": [

"data.rename(columns = {'Y house price of unit area':'price'}, inplace = True)\n"

],

"metadata": {

"id": "cO8SyGcZcYwg"

},

"execution\_count": 58,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"data.head()"

],

"metadata": {

"colab": {

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

"height": 224

},

"id": "2UJLMw56c8nH",

"outputId": "c8ed582a-d247-41d9-c09a-038cad473d8d"

},

"execution\_count": 59,

"outputs": [

{

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

"data": {

"text/plain": [

" No X1 transaction date X2 house age \\\n",

"0 1 2012.917 32.0 \n",

"1 2 2012.917 19.5 \n",

"2 3 2013.583 13.3 \n",

"3 4 2013.500 13.3 \n",

"4 5 2012.833 5.0 \n",

"\n",

" X3 distance to the nearest MRT station X4 number of convenience stores \\\n",

"0 84.87882 10 \n",

"1 306.59470 9 \n",

"2 561.98450 5 \n",

"3 561.98450 5 \n",

"4 390.56840 5 \n",

"\n",

" X5 latitude X6 longitude price \n",

"0 24.98298 121.54024 37.9 \n",

"1 24.98034 121.53951 42.2 \n",

"2 24.98746 121.54391 47.3 \n",

"3 24.98746 121.54391 54.8 \n",

"4 24.97937 121.54245 43.1 "

],

"text/html": [

"\n",

" <div id=\"df-1ad089b9-13da-4476-9314-bfe4041a63f2\" 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>No</th>\n",

" <th>X1 transaction date</th>\n",

" <th>X2 house age</th>\n",

" <th>X3 distance to the nearest MRT station</th>\n",

" <th>X4 number of convenience stores</th>\n",

" <th>X5 latitude</th>\n",

" <th>X6 longitude</th>\n",

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

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

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

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

" <td>2012.917</td>\n",

" <td>32.0</td>\n",

" <td>84.87882</td>\n",

" <td>10</td>\n",

" <td>24.98298</td>\n",

" <td>121.54024</td>\n",

" <td>37.9</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>2012.917</td>\n",

" <td>19.5</td>\n",

" <td>306.59470</td>\n",

" <td>9</td>\n",

" <td>24.98034</td>\n",

" <td>121.53951</td>\n",

" <td>42.2</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>2013.583</td>\n",

" <td>13.3</td>\n",

" <td>561.98450</td>\n",

" <td>5</td>\n",

" <td>24.98746</td>\n",

" <td>121.54391</td>\n",

" <td>47.3</td>\n",

" </tr>\n",

" <tr>\n",

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

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

" <td>2013.500</td>\n",

" <td>13.3</td>\n",

" <td>561.98450</td>\n",

" <td>5</td>\n",

" <td>24.98746</td>\n",

" <td>121.54391</td>\n",

" <td>54.8</td>\n",

" </tr>\n",

" <tr>\n",

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

" <td>5</td>\n",

" <td>2012.833</td>\n",

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

" <td>390.56840</td>\n",

" <td>5</td>\n",

" <td>24.97937</td>\n",

" <td>121.54245</td>\n",

" <td>43.1</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-1ad089b9-13da-4476-9314-bfe4041a63f2')\"\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-1ad089b9-13da-4476-9314-bfe4041a63f2 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-1ad089b9-13da-4476-9314-bfe4041a63f2');\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-e7a2f697-b912-49d4-b8b4-fbf720bacd49\">\n",

" <button class=\"colab-df-quickchart\" onclick=\"quickchart('df-e7a2f697-b912-49d4-b8b4-fbf720bacd49')\"\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-e7a2f697-b912-49d4-b8b4-fbf720bacd49 button');\n",

" quickchartButtonEl.style.display =\n",

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

" })();\n",

" </script>\n",

"</div>\n",

"\n",

" </div>\n",

" </div>\n"

],

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

"type": "dataframe",

"variable\_name": "data",

"summary": "{\n \"name\": \"data\",\n \"rows\": 414,\n \"fields\": [\n {\n \"column\": \"No\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 119,\n \"min\": 1,\n \"max\": 414,\n \"num\_unique\_values\": 414,\n \"samples\": [\n 359,\n 351,\n 374\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X1 transaction date\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.2819672402630115,\n \"min\": 2012.667,\n \"max\": 2013.583,\n \"num\_unique\_values\": 12,\n \"samples\": [\n 2013.0,\n 2012.75,\n 2012.917\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X2 house age\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 11.39248453324253,\n \"min\": 0.0,\n \"max\": 43.8,\n \"num\_unique\_values\": 236,\n \"samples\": [\n 38.2,\n 3.4,\n 27.5\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X3 distance to the nearest MRT station\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 1262.1095954078512,\n \"min\": 23.38284,\n \"max\": 6488.021,\n \"num\_unique\_values\": 259,\n \"samples\": [\n 552.4371,\n 4605.749,\n 405.2134\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X4 number of convenience stores\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 2,\n \"min\": 0,\n \"max\": 10,\n \"num\_unique\_values\": 11,\n \"samples\": [\n 6,\n 10,\n 8\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X5 latitude\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.012410196590450338,\n \"min\": 24.93207,\n \"max\": 25.01459,\n \"num\_unique\_values\": 234,\n \"samples\": [\n 24.98573,\n 24.96143,\n 24.96696\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"X6 longitude\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 0.015347183004592205,\n \"min\": 121.47353,\n \"max\": 121.56627,\n \"num\_unique\_values\": 232,\n \"samples\": [\n 121.54634,\n 121.50831,\n 121.53372\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n },\n {\n \"column\": \"price\",\n \"properties\": {\n \"dtype\": \"number\",\n \"std\": 13.606487697735314,\n \"min\": 7.6,\n \"max\": 117.5,\n \"num\_unique\_values\": 270,\n \"samples\": [\n 25.0,\n 30.7,\n 18.3\n ],\n \"semantic\_type\": \"\",\n \"description\": \"\"\n }\n }\n ]\n}"

}

},

"metadata": {},

"execution\_count": 59

}

]

},

{

"cell\_type": "code",

"source": [

"# Separate target from predictors\n",

"#data.dropna(axis=0, subset=['Y house price of unit area'], inplace=True)\n",

"y = data.price\n",

"data.drop(['price'], axis=1, inplace=True)\n",

"y\n"

],

"metadata": {

"colab": {

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

},

"id": "9qL29kJlSeOg",

"outputId": "a5e2e6e2-76a9-4367-93a6-18c17cfb3d51"

},

"execution\_count": 60,

"outputs": [

{

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

"data": {

"text/plain": [

"0 37.9\n",

"1 42.2\n",

"2 47.3\n",

"3 54.8\n",

"4 43.1\n",

" ... \n",

"409 15.4\n",

"410 50.0\n",

"411 40.6\n",

"412 52.5\n",

"413 63.9\n",

"Name: price, Length: 414, dtype: float64"

]

},

"metadata": {},

"execution\_count": 60

}

]

},

{

"cell\_type": "code",

"source": [

"# Break off validation set from training data\n",

"X\_train, X\_test, y\_train, y\_test = train\_test\_split(data, y, train\_size=0.8, test\_size=0.2,\n",

" random\_state=0)"

],

"metadata": {

"id": "Uc-OgrVxaJvZ"

},

"execution\_count": 80,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#train model\n",

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

"model = LinearRegression()"

],

"metadata": {

"id": "pjfgfWo5eW7k"

},

"execution\_count": 81,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"# fit model\n",

"model.fit(X\_train,y\_train)\n"

],

"metadata": {

"id": "tSWgE9NjfH3S"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#predict model\n",

"y\_pred = model.predict(X\_test)\n",

"y\_pred"

],

"metadata": {

"colab": {

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

},

"id": "XE6NdQ4Qfgvw",

"outputId": "d9a4efa5-2399-41a1-cd3c-936e5ec7fec4"

},

"execution\_count": 83,

"outputs": [

{

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

"data": {

"text/plain": [

"array([39.58239402, 12.95168324, 41.92356069, 12.0452379 , 42.05946963,\n",

" 39.28628163, 40.79999655, 34.84049471, 50.15018352, 45.57596321,\n",

" 45.35285689, 41.34572498, 38.79094654, 40.10572538, 46.53393646,\n",

" 38.70847921, 39.80702168, 41.47127632, 41.7610089 , 42.49361355,\n",

" 49.1468402 , 31.81302255, 39.16184157, 46.24063756, 46.49219583,\n",

" 43.10198071, 46.32151807, 26.1278043 , 47.3900123 , 20.98052304,\n",

" 44.03894485, 34.7726307 , 43.32974226, 39.2050795 , 43.69491539,\n",

" 31.56451658, 47.1976703 , 40.08972049, 53.15820449, 7.25942379,\n",

" 51.78890909, 33.81267781, 33.77960594, 48.83111829, 14.91325118,\n",

" 43.59055022, 42.40970069, 12.24052675, 34.64954306, 47.93022598,\n",

" 52.67947976, 39.70392356, 48.67788875, 25.00512038, 30.9525337 ,\n",

" 33.92637587, 48.24575219, 40.97590854, 43.38260132, 32.75625508,\n",

" 43.87176365, 44.2214297 , 39.78920416, 47.41154954, 37.25210908,\n",

" 30.12726205, 10.41052944, 35.6113375 , 47.05630793, 40.53553101,\n",

" 31.44424444, 47.11366261, 44.84013415, 30.06525305, 41.0352289 ,\n",

" 36.10880642, 32.12998874, 13.23083234, 46.65630982, 45.17224709,\n",

" 47.71106763, 29.95204886, 43.85222711])"

]

},

"metadata": {},

"execution\_count": 83

}

]

},

{

"cell\_type": "code",

"source": [

"from sklearn.metrics import mean\_absolute\_error, mean\_absolute\_percentage\_error, mean\_squared\_error\n"

],

"metadata": {

"id": "mBvPX8gufo1y"

},

"execution\_count": 84,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"mae =mean\_absolute\_error(y\_test,y\_pred)\n",

"mae"

],

"metadata": {

"colab": {

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

},

"id": "gEkiGmhegDOD",

"outputId": "4a764a97-18b2-4e87-80f7-43bcc8df4d8d"

},

"execution\_count": 85,

"outputs": [

{

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

"data": {

"text/plain": [

"5.776541798115822"

]

},

"metadata": {},

"execution\_count": 85

}

]

},

{

"cell\_type": "code",

"source": [

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

"mape"

],

"metadata": {

"colab": {

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

},

"id": "Ebu0sAWogH9L",

"outputId": "1b0c35b3-00cf-4b92-d6d6-ed061851ff06"

},

"execution\_count": 86,

"outputs": [

{

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

"data": {

"text/plain": [

"0.1704700854507169"

]

},

"metadata": {},

"execution\_count": 86

}

]

},

{

"cell\_type": "code",

"source": [

"mse = mean\_squared\_error(y\_test,y\_pred)\n",

"mse"

],

"metadata": {

"colab": {

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

},

"id": "qwl437x0gPcb",

"outputId": "4a9b2348-c0ed-437e-ecc5-9f482dcac4e9"

},

"execution\_count": 87,

"outputs": [

{

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

"data": {

"text/plain": [

"59.79820550973279"

]

},

"metadata": {},

"execution\_count": 87

}

]

},

{

"cell\_type": "code",

"source": [

"rmse = np.sqrt(mean\_squared\_error(y\_test,y\_pred))\n",

"rmse"

],

"metadata": {

"colab": {

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

},

"id": "ucd3WDdPgVBH",

"outputId": "a3373fb6-a91a-4177-d3e4-f0dff962f4d1"

},

"execution\_count": 88,

"outputs": [

{

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

"data": {

"text/plain": [

"7.7329299434129615"

]

},

"metadata": {},

"execution\_count": 88

}

]

}

]

}