

# Prior Knowledge

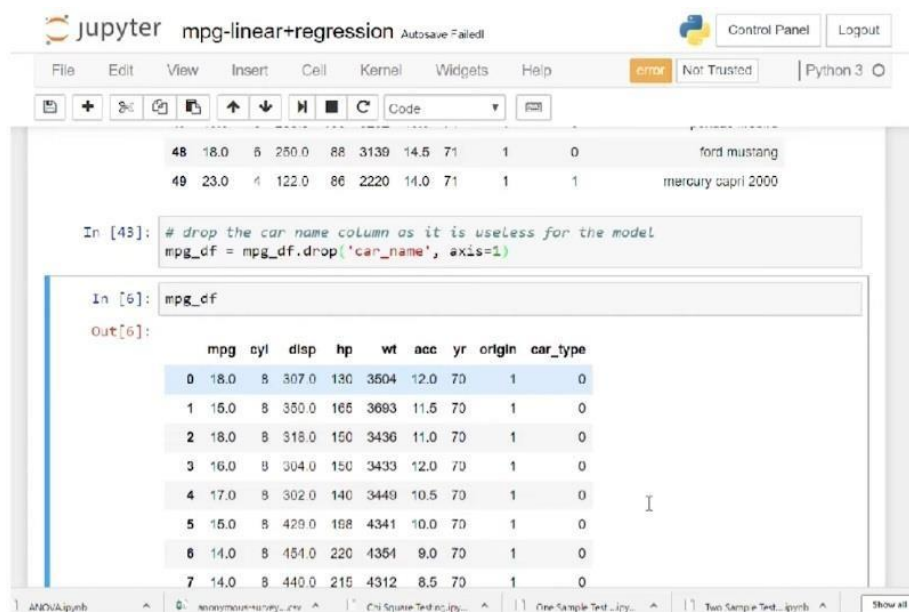
Date	4 October 2022
Team Id	PNT2022TMID21248
Project Name	Smart Lender-Applicant Credibility Prediction for Loan Approval

## Machine Learning Concepts:

Machine Learning is **an application of artificial intelligence where a computer/machine learns from the past experiences (input data) and makes future predictions.**

## Supervised Learning:

Supervised learning, also known as supervised machine learning, is **a subcategory of machine learning and artificial intelligence.** It is defined by its use of labeled datasets to train algorithms that to classify data or predict outcomes accurately.



The screenshot shows a Jupyter Notebook titled "mpg-linear+regression". The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and code execution, and a status bar at the bottom. The code cell shows the following:

```
In [43]: # drop the car name column as it is useless for the model
mpg_df = mpg_df.drop('car_name', axis=1)
```

The output cell displays the resulting DataFrame:

```
In [6]: mpg_df
Out[6]:
```

	mpg	cyl	displ	hp	wt	acc	yr	origin	car_type
0	18.0	8	307.0	130	3504	12.0	70	1	0
1	15.0	8	350.0	165	3693	11.5	70	1	0
2	18.0	8	318.0	150	3436	11.0	70	1	0
3	16.0	8	304.0	150	3433	12.0	70	1	0
4	17.0	8	302.0	140	3449	10.5	70	1	0
5	15.0	8	429.0	198	4341	10.0	70	1	0
6	14.0	8	454.0	220	4351	9.0	70	1	0
7	14.0	8	440.0	215	4312	8.5	70	1	0

```
5 5 116 74 0 0 25.6 0.201 30 0
6 3 78 50 32 88 31.0 0.248 26 1
7 10 115 0 0 0 35.3 0.134 29 0
8 2 197 70 45 543 30.5 0.158 53 1
9 8 125 96 0 0 0.0 0.232 54 1
```

In [5]: dataframe.describe()

```
Out[5]:
```

	preg	plas	pres	skin	test	mass	pedi
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243750
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.372500

```
In [1]: # The data lives in the following URL.
#url = "https://archive.ics.uci.edu/ml/machine-learning-databases/pima-indians-diabetes/pima-indians-diabetes.data"

In [2]: # Since it is a data file with no header, we will supply the column names which have been obtained from
# Create a python list of column names called "names"
colnames = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']

# Load the file from local directory using pd.read_csv which is a special form of read_table
# While reading the data, supply the "colnames" list
pima_df = pd.read_csv("pima-indians-diabetes.data", names=colnames)

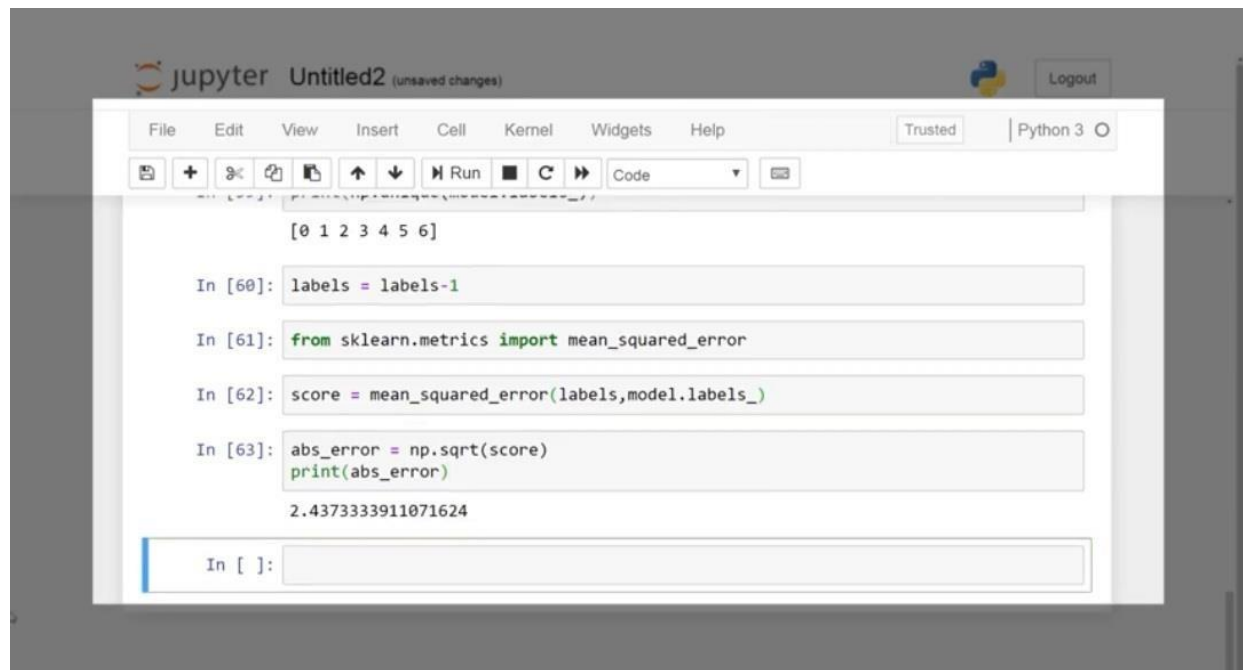
In [4]: pima_df.head(30)
```

```
Out[4]:
```

	preg	plas	pres	skin	test	mass	pedi	age	class
0	5	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	14	26.1	0.157	21	0
4	0	137	40	35	160	43.1	2.288	33	1
5	5	115	74	0	0	25.6	0.201	30	0
6	3	78	50	32	88	31.0	0.248	26	1
7	10	115	0	0	0	35.3	0.134	29	0
8	2	197	70	45	543	30.5	0.158	53	1
9	8	125	96	0	0	0.0	0.232	54	1
10	4	110	92	0	0	37.6	0.191	30	0
11	10	161	74	0	0	33.0	0.517	34	1

# UnSupervised Learning:

**unsupervised learning**, also known as **unsupervised machine learning**, uses machine learning algorithms to analyze and cluster unlabeled datasets. **These algorithms discover hidden patterns or data groupings without the need for human intervention.**



The image shows a Jupyter Notebook window titled "Untitled2 (unsaved changes)". The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help), a toolbar with icons for file operations and execution, and a code editor. The code editor contains the following Python code:

```
[0 1 2 3 4 5 6]

In [60]: labels = labels-1

In [61]: from sklearn.metrics import mean_squared_error

In [62]: score = mean_squared_error(labels,model.labels_)

In [63]: abs_error = np.sqrt(score)
          print(abs_error)

2.4373333911071624

In [ ]:
```

```
Out[71]: array([[ 50.04763437,  8.82875   ],
                [180.017075 , 18.29    ]])
```

```
In [72]: print(kmeans.labels_)
print(len(kmeans.labels_))

[0 0 0 ... 1 1 1]
4000
```

```
In [73]: print(type(kmeans.labels_))
unique, counts = np.unique(kmeans.labels_, return_counts=True)
print(dict(zip(unique, counts)))

<class 'numpy.ndarray'>
{0: 3200, 1: 800}
```

```
In [ ]: df_analyze['cluster'] = kmeans.labels_
sns.set_style('whitegrid')
sns.lmplot('mean_dist_day', 'mean_ove|')
```