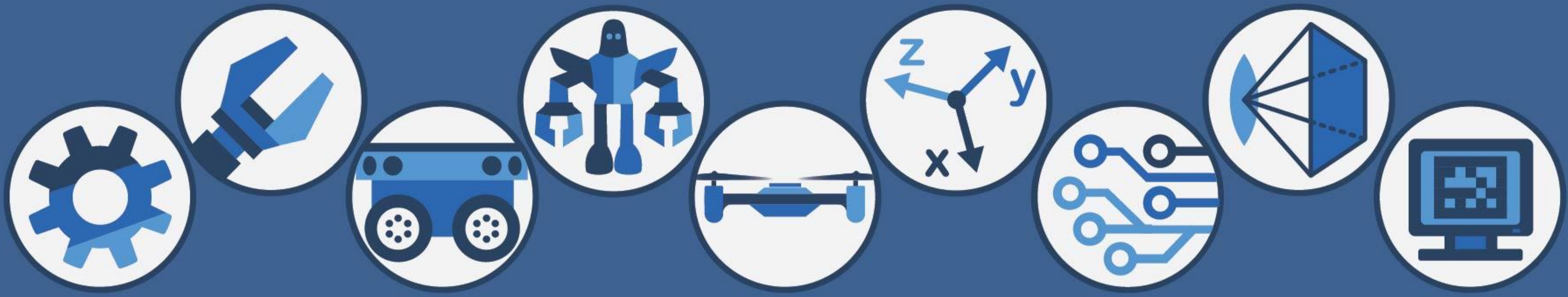
MACHINE LEARNING LAB

Linear Regression

SUNEELA ZAFAR



School of Electrical Engineering and Computer Science (SEECS), NUST, Pakistan

Features and Labels

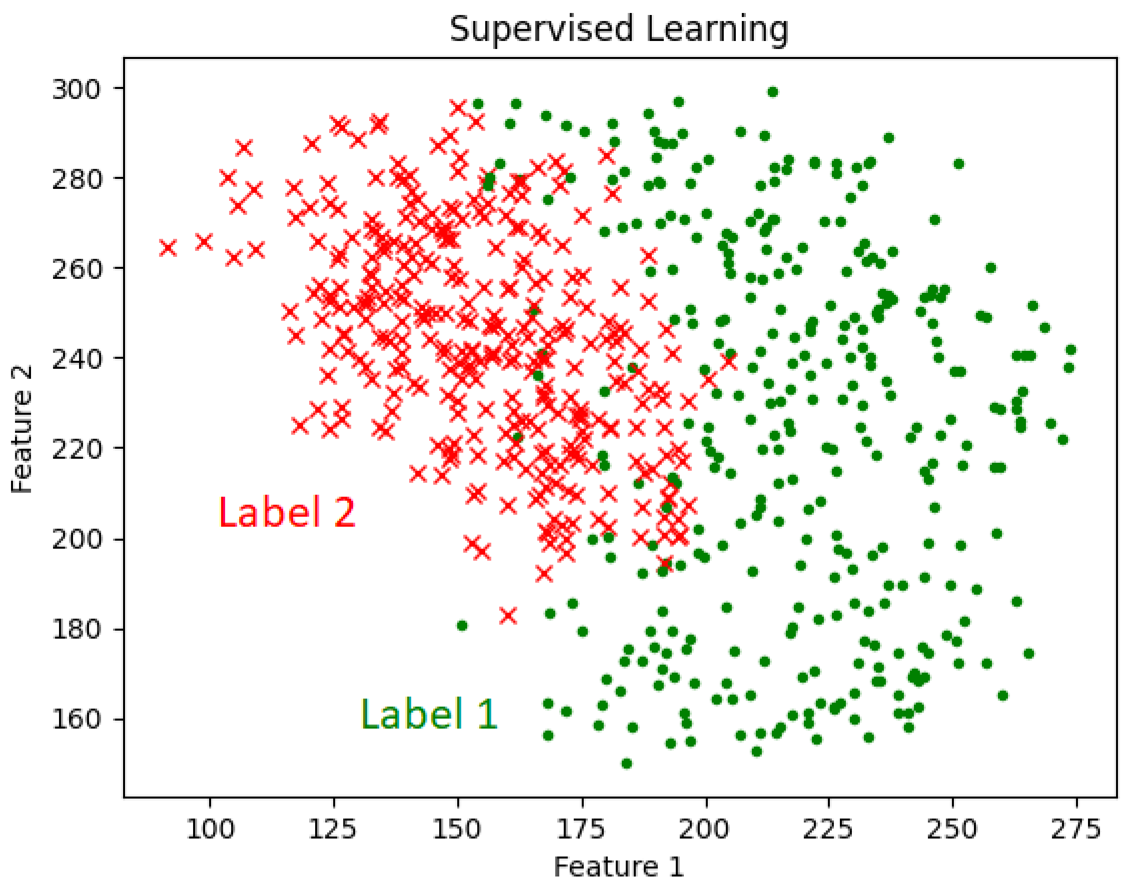
- Machine Learning consists of a wide variety of techniques that can be classified as Supervised, Unsupervised and Reinforcement Learning etc.

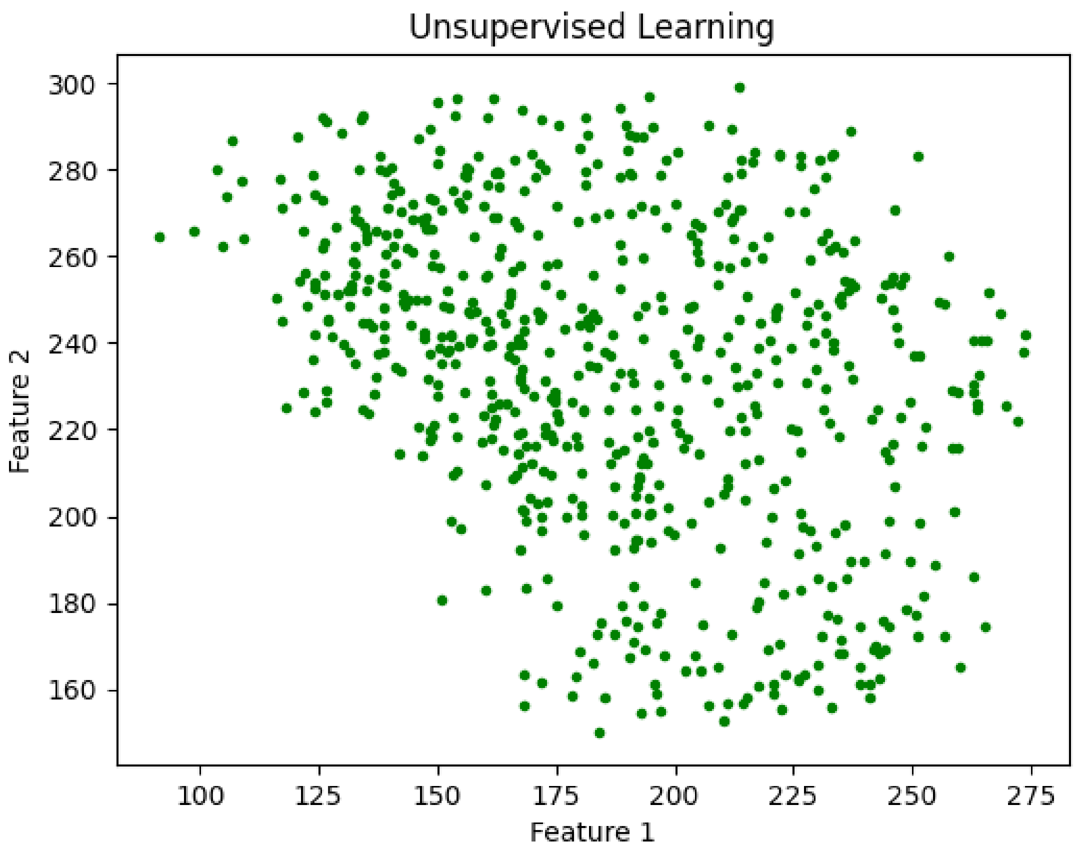
- A machine learning dataset essentially contains:
  - **Features:** the input columns in the dataset
  - **Labels:** the output columns in the dataset

- After the model is trained on the dataset, it is used to make a prediction (inference)
- To make a prediction, the user provides some new values for the features. These values are not from the dataset and are used by the trained model to give the output (prediction)

Supervised Learning

- Supervised Learning involves datasets that have both features and labels. Examples include Linear Regression, Logistic Regression, Neural Networks etc.
- There are only features in Unsupervised Learning





Dataset

- A dataset in tabular form is shown

i	x1	x2	x3	y
1	90	6040	4	82000
2	20	8658	6	160000
3	20	16905	5	170000
4	85	9180	5	144000
5	20	9200	5	130250
6	20	7945	5	141000
7	120	7658	9	319900
8	50	12822	7	239686
9	20	11096	8	249700
10	190	4456	4	113000

Dataset

- A dataset in tabular form is shown
- Each serial number corresponds to the i-th training example
- There are 10 total training examples; m = 10

i	x1	x2	x3	y
1	90	6040	4	82000
2	20	8658	6	160000
3	20	16905	5	170000
4	85	9180	5	144000
5	20	9200	5	130250
6	20	7945	5	141000
7	120	7658	9	319900
8	50	12822	7	239686
9	20	11096	8	249700
10	190	4456	4	113000

Dataset

- A dataset in tabular form is shown

- Each serial number corresponds to the i-th training example
- There are 10 total training examples; m = 10

- There are 3 features x1, x2, x3
- Each feature column has entries for each training example

i	x1	x2	x3	y
1	90	6040	4	82000
2	20	8658	6	160000
3	20	16905	5	170000
4	85	9180	5	144000
5	20	9200	5	130250
6	20	7945	5	141000
7	120	7658	9	319900
8	50	12822	7	239686
9	20	11096	8	249700
10	190	4456	4	113000

Dataset

- A dataset in tabular form is shown

- Each serial number corresponds to the i-th training example
- There are 10 total training examples; m = 10

- There are 3 features x1, x2, x3
- Each feature column has entries for each training example

- There is 1 label column y
- The label has continuous values; it is a regression problem

i	x1	x2	x3	y
1	90	6040	4	82000
2	20	8658	6	160000
3	20	16905	5	170000
4	85	9180	5	144000
5	20	9200	5	130250
6	20	7945	5	141000
7	120	7658	9	319900
8	50	12822	7	239686
9	20	11096	8	249700
10	190	4456	4	113000

Dataset

- Suppose we are given the following feature values:
  - x1 = 70
  - x2 = 8860
  - x3 = 5
- What will be the corresponding value of y?
  - y = ?
- To predict the value of y, we can train a model on the dataset

i	x1	x2	x3	y
1	90	6040	4	82000
2	20	8658	6	160000
3	20	16905	5	170000
4	85	9180	5	144000
5	20	9200	5	130250
6	20	7945	5	141000
7	120	7658	9	319900
8	50	12822	7	239686
9	20	11096	8	249700
10	190	4456	4	113000

Linear Regression

- This lab will focus on implementing linear regression to train a simple machine learning model on a dataset

- Topics covered in the lab include

  - Feature Scaling
  - Hypothesis $h(x)$
  - Cost Function $J(w, b)$
  - Gradient Descent Algorithm

- These topics form the core of machine learning concepts which will also be central to upcoming labs

Feature Scaling

- A preprocessing step that brings feature values in a 0-1 range

$$x_{j_{\text{scaled}}}[\mathrm{i}] = \frac{x_j[\mathrm{i}] - x_{j\_}\mathrm{min}}{x_{j\_}\mathrm{max} - x_{j\_}\mathrm{min}}$$

i	x1	x2	x3	y
1	90	6040	4	82000
2	20	8658	6	160000
3	20	16905	5	170000
4	85	9180	5	144000
5	20	9200	5	130250
6	20	7945	5	141000
7	120	7658	9	319900
8	50	12822	7	239686
9	20	11096	8	249700
10	190	4456	4	113000

i	x1	x2	x3	y
1	0.41	0.13	0.00	82000
2	0.00	0.34	0.40	160000
3	0.00	1.00	0.20	170000
4	0.38	0.38	0.20	144000
5	0.00	0.38	0.20	130250
6	0.00	0.28	0.20	141000
7	0.59	0.26	1.00	319900
8	0.18	0.67	0.60	239686
9	0.00	0.53	0.80	249700
10	1.00	0.00	0.00	113000

Hypothesis h(x)

- The "hypothesis" h(x) gives the predicted value of the model:

$$h(x) = b + w_1 x_1 + w_2 x_2 + w_3 x_3$$

x_j is the j-th feature
w_j is the j-th weight
b is the bias

- Both the weights and bias are the trainable parameters
- At the start, the weights and bias are set to some random values
- During training, the weights and bias are updated (via gradient descent)

- The difference between the actual label y and the predicted label h(x) gives the loss

Cost Function J(w,b)

- The cost function J(w,b) gives the sum of losses over the entire dataset

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)})^2$$

$h(x^i)$ is the predicted label of the i-th training example
y^i is the actual label of the i-th training example
J is the cost (the accumulation of losses)

- At each i-th training example, the loss $(h(x)-y)^2$ is calculated
- The losses from all the training examples are summed to give the cost

- The cost is to be computed at every epoch to check whether it is decreasing or not as the weights and bias are being updated

Gradient Descent

- The gradient descent algorithm for linear regression is given as follows:

$$dw_j = \frac{\partial \mathrm{J}}{\partial w_j} = \frac{1}{m}\sum_{i=1}^{m}(\mathrm{h}(\mathrm{x}^{(i)}) - \mathrm{y}^{(i)})x_j{}^{(i)}$$

$$db = \frac{\partial \mathrm{J}}{\partial b} = \frac{1}{m}\sum_{i=1}^{m}(\mathrm{h}(\mathrm{x}^{(i)}) - \mathrm{y}^{(i)})$$

$$w_j := w_j - \alpha \frac{\partial \mathrm{J}}{\partial w_j}$$

$$b := b - \alpha \frac{\partial \mathrm{J}}{\partial b}$$

w_j, b are the weights and bias

dw_j, db are the derivatives of the cost w.r.t. weights and bias

Common Notations

m	Training examples
n	Number of features
y	Output label
w	Weights
b	Bias
dw	Derivative w.r.t weights
db	Derivative w.r.t bias

Gradient Descent

- The gradient descent algorithm for linear regression is given as follows:

$$w_j := w_j - \alpha \frac{\partial \mathrm{J}}{\partial w_j} \qquad b := b - \alpha \frac{\partial \mathrm{J}}{\partial b}$$

- The weight update given above can be implemented as:

```
temp1 = w1 - alpha *dJdw1
temp2 = w2 - alpha *dJdw2
temp3 = w3 - alpha *dJdw3
temp0 = b - alpha *dJdb
w1 = temp1
w2 = temp2
w3 = temp3
b = temp0
```

```
temp1 = w1 - alpha *dJdw1
w1 = temp1
temp2 = w2 - alpha *dJdw2
w2 = temp2
temp3 = w3 - alpha *dJdw3
w3 = temp3
temp0 = b - alpha *dJdb
b = temp0
```

Lab Tasks

- Download the Lab Material
- Perform the Lab Tasks given in the manual
- Submit the saved manual in PDF form
- Remember to delete your manual after completing the lab session