MACHINE LEARNING

DATA ANALYSIS

IIT PALAKKAD

"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."

--Tom M Mitchell

It can be categorized under 3 categories:-

1. SUPERVISED
2. UNSUPERVISED
3. REINFORCEMENT

SUPERVISED LEARNING:-

The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.

Further this map will be used to predict the outputs for other inputs.

Eg.,

- Regression
- Classification
- Support Vector Machine

UNSUPERVISED LEARNING:-

No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).

Eg.,

- Clustering
  - Google News
  - Facebook Friend Suggestions
- Anomaly Detection
- Dimensionality Reduction
  - Image compression

REINFORCEMENT LEARNING:-

A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent[5]:3). The program is provided feedback in terms of rewards and punishments as it navigates its problem space.

Eg.,

- Computer Games
- Vehicular Navigation

REGRESSION

Regression is a method that enables you to determine the relationship between a continuous process output (Y) and some factor (X).

Eg:

1. Predicting the price of a house.
2. Given a picture of male/female we have predict whether he/she is a high school,graduate etc

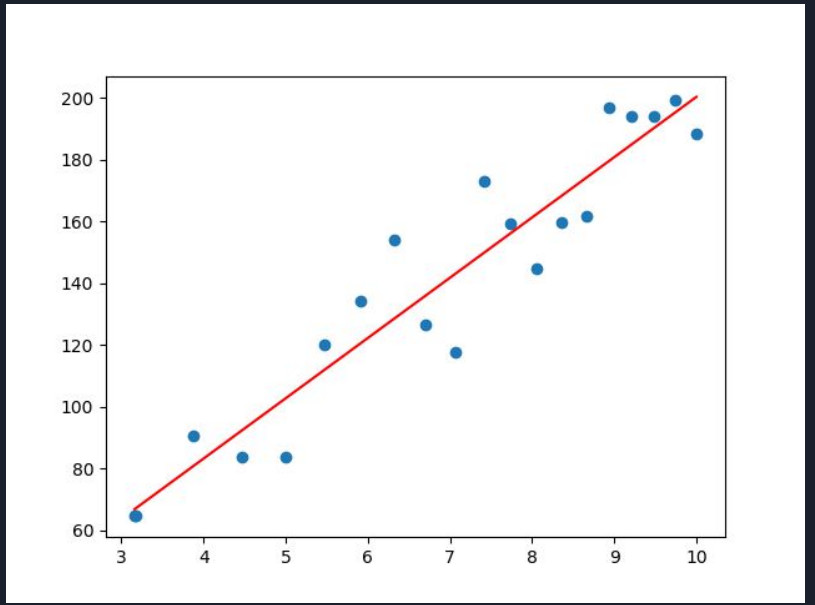
Types:

- Linear Regression
- Polynomial Regression
- Non-Linear Regression
- Generalized Regression

LINEAR REGRESSION

Linear Regression is a type of regression in which output(Y) can be expressed as linear function of input(X).

$Y = \Theta.X + C;$



NOTATIONS:-

$x^{(i)}$ = input variable (feature)

$y^{(i)}$ = output or target variable (label)

$(x^{(i)}, y^{(i)})$ = a training example

Note that the ‘i’ is not the power of the variable. It's just a representation to denote a particular example.

m = number of training examples

$\therefore i = \{1, 2, \ldots, m\}$

n= number of features

NOTATIONS:-

$h_\theta(x) = \theta_0 + \theta_1 x_1 + \ldots\ldots + \theta_n x_n$ = hypothesis function

Where ,

θ_j = weights or parameters

For a line type of hypothesis:-

$h_\theta(x) = \theta_0 + \theta_1 x_1$ (slope-intercept form)

For simplicity let's assume $x_0 = 1$ then,

$h_\theta(x) = \sum \theta_j x_j$ where $j = \{0, 1, 2, \ldots\ldots, n\}$

$\Rightarrow h_\theta(x^{(i)}) = \sum_j \theta_j x^{(i)}_j$ (for a particular training example 'i')

NOTATIONS:-

Representing θ_j and x_j in the form of column vectors.

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{bmatrix}$$

$$x^{(i)} = \begin{bmatrix} x_0^{(i)} \\ x_1^{(i)} \\ \vdots \\ x_n^{(i)} \end{bmatrix}$$

$\therefore h_\theta(x^{(i)}) = \sum_j \theta_j x^{(i)}_j = \theta^T x^{(i)}$ (θ^T = transpose of parameters matrix)

NOTATIONS:-

“Cost Function” OR “mean squared error” OR “squared error function” : -

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta \left(x^{(i)} \right) - y^{(i)} \right)^2$$

Sometimes, abbreviated as MSE.

Gradient

The gradient is a vector which points in the direction of greatest rate of increase of the function.

1. f (x, y) increases most rapidly in the direction of its gradient.
2. f (x, y) decreases most rapidly in the opposite direction of its gradient
3. Magnitude of gradient is rate of increase

Magnitude of gradient in two dimension can be considered as the slope of tangent.

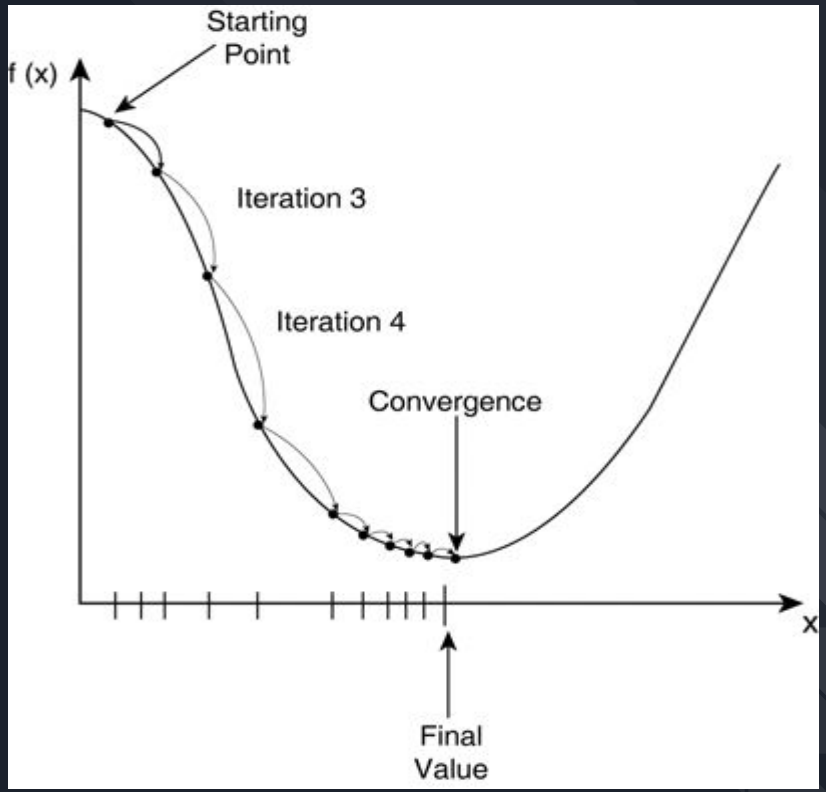
Gradient Descent(2D)

Let f(x) is a convex function.
A convex function has only global minimum.

Aim: To find X at which value of function f(x) is minimum.

Algorithm:-

Repeat till you reach minimum {
    X := X - α*(slope)
}
α is a very small constant and it is called as learning rate.



GRADIENT DESCENT FOR LINEAR REGRESSION (1-FEATURE):

$$Repeat - Until - Convergence\{$$

$$\theta_j := \theta_j - \alpha \left(\frac{\partial J}{\partial \theta_j} \right)$$

$$\}$$

Thanks!

Amit Vikram Singh (111601001)
Kaushal Kishore (111601008)
Sai Suchith Mahajan (121601016)