Machine Learning

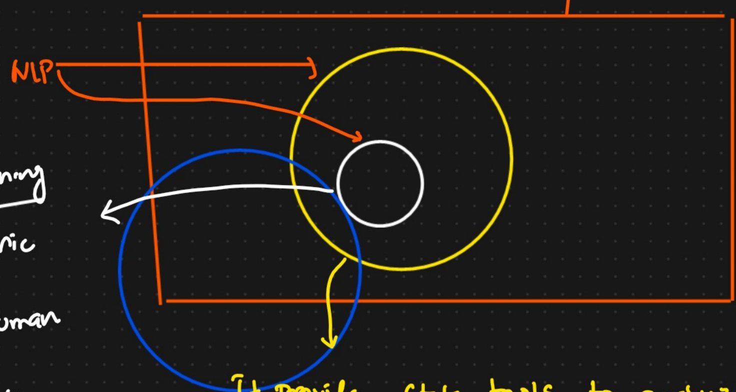
AI Vs ML Vs DL Vs DS

NLP → Alexa (?)
⇓
I/p ⇒ Text data.

Artificial Intelligence

AI — Applications that can perform its own task without any human intervention

Eg: Alexa
Self Driving Cars
Amazon shopping Website
⇓
Recommendation System



Deep Learning
Mimic the Human Brain

{Multilayered Neural N/w}

Eg: Object Detection, Image Recognition, Chatbots, Recommendation System.

It provides stats tools to analyze the data, visualize, predictive models, forecasting

1. Statistics
2. Probabilities
3. Linear Algebra
4. Calculus

Gen AI → Large Language Models

② Supervised, Unsupervised, Semi Supervised, Reinforcement Learning

Types of ML And DL

① Supervised ML → CLASSIFICATION, → Regression

DATASET: O/p feature → Categories, → Continuous

New I/p → Model → O/p {predicted}.
(O/p into Model)

Independent features		Dependent feature
No of hours played	No of study hours	Pass/Fail
8	2	Fail
7	3	Fail
6	4	Fail
5	5	Pass
4	6	Pass

Binary classification.
Multiclass Classification.

House Price Prediction

I/p features

Continuous {Regression}.

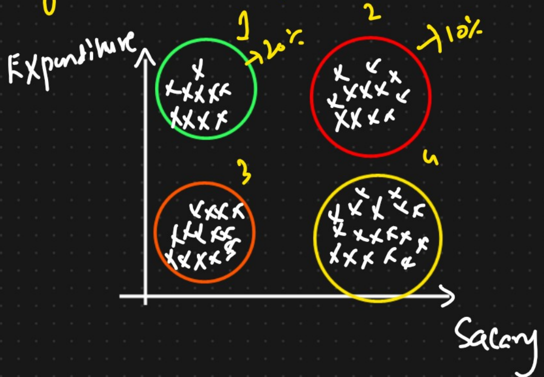
House Size	No. of Bedrooms	No. of Bathrooms	No. of Balcony

$\boxed{\text{Price}}$ → Dependent feature

② Unsupervised ML

⇒ No o/p or dependent features ⇒ clusters ⇒ Similar Groups

Eg: Customer Segmentation.

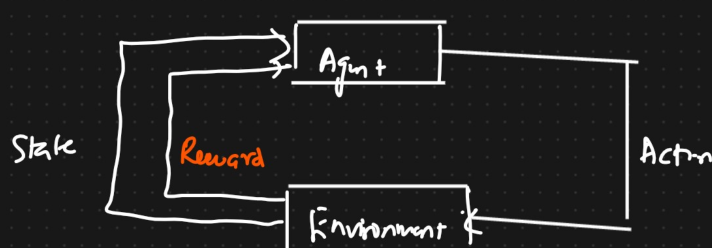


③ Semi Supervised :

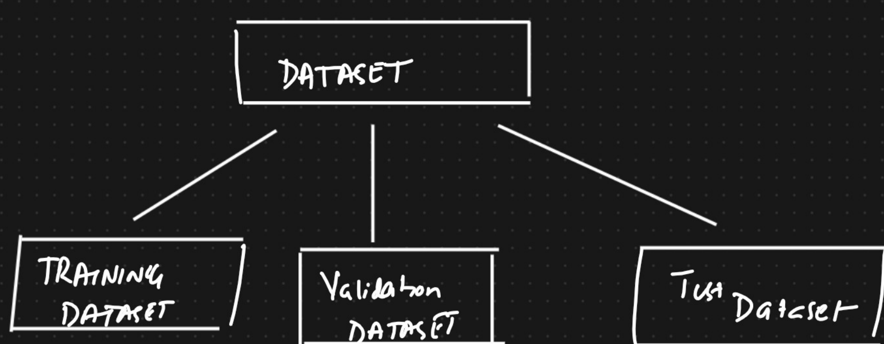
Supervised + Unsupervised

④ Reinforcement Learning : {Research}.

Increase the Reward As much as possible



⑤



We will Train our

Hyperparameter

Model will be Test

Models	Tuning	
⇓	⇓	⇓
Books	Different Books	Exam ⇒ Question paper → Perform
QA	⇓	⇓ ⇓
⇓	Increasing Accuracy	Brain Test, Marks ↑↑
Train	of the Model	

④ **Machine Learning**

① Model Performance → Accuracy ↑↑ → High

② Overfitting, Underfitting

③ Bias Vs Variance

Overfitting And Underfitting

Variance → Test Accuracy.

Bias → Training Accuracy

DATASET

① Books → TRAIN → Model Is TRAIN → Accuracy ↑↑ → 95%

Exam → Test → Model Is TESTED → Accuracy ↓↓ → 65%

} Overfitting ⇓ { Low Bias, High Variance }

② Train Accuracy ↓↓ → 55%

Test Accuracy ↓↓ → 50%

60%

} Underfitting { High Bias, High Variance }

Aim For ⇓.

Generalized Model

Train Acc ↑↑ = 95%

Test Acc ↑↑ = 90%

{ Low Bias
Low Variance }

Simple Linear Regression

DATASET

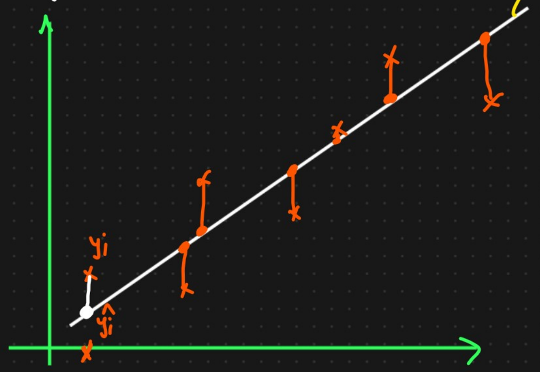
x Weight (Independent feature)	Height (Dependent feature, o/p) y
74	170
80	180
75	175.5
–	–
–	–
–	–

I/p And o/p

TRAIN DATASET ↓ Model

New Weight → Model → prediction Height

TRAIN ACC ↑↑
TEST ACC ↑↑ } Generalized Model



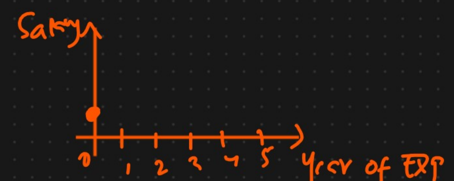
→ best fit line → Predictions

$\sum$ Error $(y_i - \hat{y}_i)$ ↓↓ Minimal

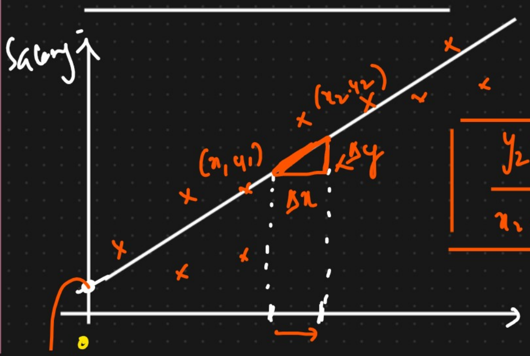
Line 1 ⇒
Error ⇒ 25

Line 2
Error = 23

Geometrical Intuition



Mathematical Intuition



$\boxed{\dfrac{y_2 - y_1}{x_2 - x_1}} = \dfrac{\Delta y}{\Delta x}$

$\hat{y} = mx + c$

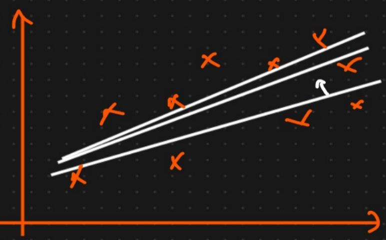
$\hat{y} = \beta_1 x_1 + \beta_0$

$\boxed{h_\theta(x) = \theta_1 x_1 + \theta_0}$

m = slope
c = intercept

θ_0 ⇒ Intercept
θ_1 ⇒ Slope or coefficient
x_i ⇒ DATA POINTS

Intercept



Cost function [Error]

$$J(\theta_0,\theta_1) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - h_\theta(x_i)\right)^2$$

[Mean Squared Error] ⇓ MSE

n = no. of datapoints

y = Actual value

$h_\theta(x)$ = Predicted value

Final fun → [In order to get the best fit line]

$$\underset{\theta_0,\theta_1}{\text{Minimize}}\; J(\theta_0,\theta_1) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - h_\theta(x)_i\right)^2$$

$h_\theta(x) = \theta_0 + \theta_1 x_1$

Let consider $\theta_0 = 0$

$$\boxed{h_\theta(x) = \theta_1 x_1}$$

⇓

x	y
1	1
2	2
3	3

	Let $\theta_1 = 1$	Let $\theta_1 = 0.5$	Let $\theta_1 = 0$
Predicted $x_1 = 1$	$h_\theta(x) = 1$	$h_\theta(x) = 0.5$	$h_\theta(x) = 0$
$x_1 = 2$	$h_\theta(x) = 2$	$h_\theta(x) = 1$	$h_\theta(x) = 0$
$x_1 = 3$	$h_\theta(x) = 3$	$h_\theta(x) = 1.5$	$h_\theta(x) = 0$

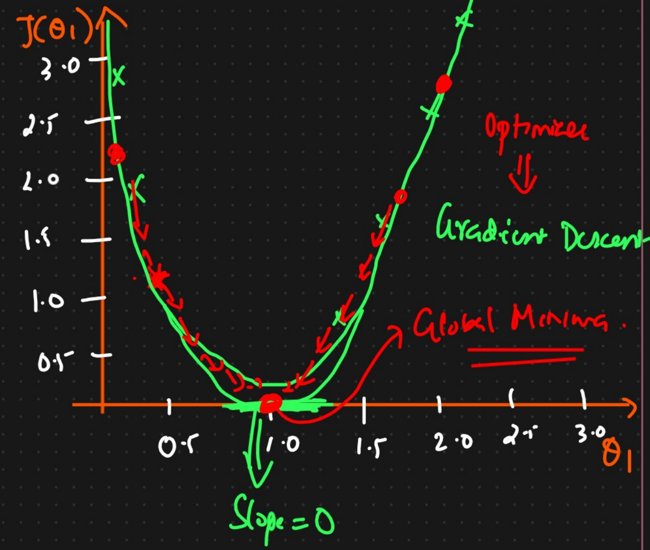
Cost fn $\theta_1 = 1$

$n = 3$

$$J(\theta_0,\theta_1) = \frac{1}{n}\sum_{i=1}^{n}(y_i - h_\theta(x))^2$$

$$= \frac{1}{3}\left[(1-1)^2 + (2-2)^2 + (3-3)^2\right]$$

$$= 0$$



$\theta_1 = 0.5$

$$J(\theta_0,\theta_1) = \frac{1}{n}\sum_{i=1}^{n}(y_i - h_\theta(x))^2$$

$$= \frac{1}{3}\left[(1-0.5)^2 + (2-1)^2 + (3-1.5)^2\right].$$

$$= 1.16.$$

$$J(\theta_0,\theta_1) = \frac{1}{n}\sum_{i=1}^{n}(y_i - h_\theta(x))^2$$

$$= \frac{1}{3}\left[(1-0)^2 + (2-0)^2 + (3-0)^2\right]$$

$$= 4.66$$

Convergence Algorithm {Optimize the change of θ_0, θ_1 to Global Minima}

Repeat until Convergence

{

Learning Rate

$$\theta_j := \theta_j - \alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_j)} \Rightarrow \text{Derivative or Slope}$$

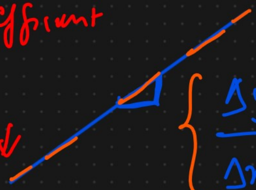
}

$$\frac{\partial}{\partial x}(x^2) = 2x$$

$\theta_1 \Rightarrow$ Coefficient

$\Downarrow$

Error fold



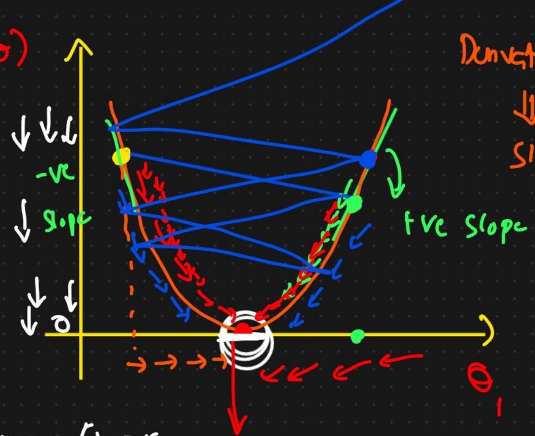
$\left\{ \frac{\Delta y}{\Delta x} \right\}$

$\boxed{\alpha = 0.1}$

$\alpha = 0.001$ $\alpha = 10000$

$\alpha = 1$

Error $J(\theta)$



Derivative

$\Downarrow$

Slope

Minimum Error

α = Learning Rate

$\Downarrow$

Speed of Convergence

① $\theta_1 = \theta_1 - \alpha(+ve)$

$= \theta_1 - (+ve)$

$\boxed{\theta_{1new} < \theta_{1old}}$

② $\theta_1 = \theta_1 - \alpha(-ve)$

$= \theta_1 +$ +ve value

$\theta_{1new} > \theta_{1old}$

Conclusion

Repeat until convergence

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

$$J(\theta_0, \theta_1) = \frac{1}{n} \sum_{i=1}^{n} (y_i - h_\theta(x)_i^2)$$

$\Downarrow$

Mean Square Error

$\Downarrow$

Minimize.

$\theta_1 = 0.3$

$h_\theta(x) = \theta_0 + \theta_1 x_1$

$= \theta_0 + (0.3) * — \Rightarrow$ O/P

1 Independent 1 dependent

② Multiple Linear Regression

Multiple Independent 1 dependent

House Size	No. of Bathroom	No. of Bedrooms	House Price

$h_\theta(x) = \theta_0 + \theta_1 x_1$

$$\boxed{h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3}$$

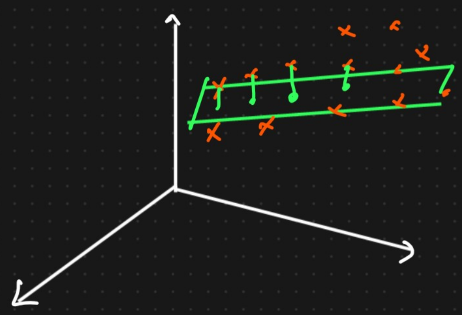
$\boxed{\theta_0, \theta_1, \theta_2, \theta_3}$

$\theta_1 = 0.5$ $\theta_2 = 0.6$ $\theta_3 = 1.25$

$\theta_1 = -0.5$ $\theta_2 = -0.7$ $\theta_3 = 1.00$

unit movement in O/p +ve

Gradient Descent → optimizer → Mean Squared Error ⇒ Derivative



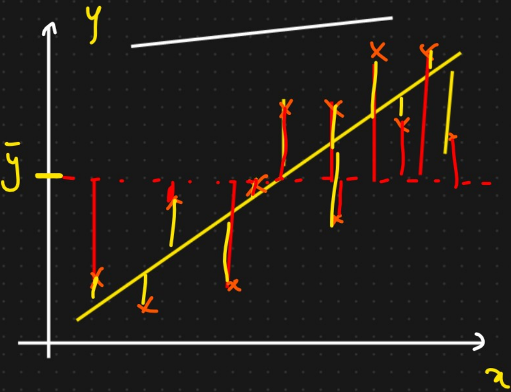
2 Indepent 1 dependent

Performance Metrics And Cost Function

1. R Squared
2. Adjusted R Squared

① R Squared

$$R\ Squared = 1 - \frac{SS_{Res}\ \{Error\}\ (y_i - \hat{y}_i)}{SS_{Total}\ (y_i - \bar{y})}$$



$\bar{y}$: Average of y.

SS_{Res} = Sum of Square Residual

SS_{Total} = Sum of Square Total

$$R\ Squared = 1 - \frac{\sum_{i=1}^{n} (y_i - h_\theta(x))^2 \Rightarrow Small}{\sum_{i=1}^{n} (y_i - \bar{y})^2 \Rightarrow Big} \Bigg\} Small$$

R squared value ranges between $0 \rightarrow 1$

$f_0\ f_1\ f_2\ f_3 \rightarrow$ o/p $\Rightarrow$ overfitting

R Squared = 0.75 ⇒ 75%

R Squared = 80%

R Squared = 85%

R Squared = 87%

② Adjusted R Squared

R Squared = 85%

R Squared = 90%

" = 95%

R Squared = 97%

Gender	Size of house	No. of Rooms	Location	Price

Adjusted R squared = $1 - \frac{(1-R^2)(N-1)}{N-p-1}$

N = No. of datapoints.

R^2 = R squared

P = No. of Independent features.

$\boxed{R^2 = 0.8}$ N = 11 p = 2 p = 3

Adjusted R square = $1 - \frac{[(0.2)(10)]}{11-2-1} = \boxed{0.75}$

$R^2 = 80\%$ Adjusted $R^2 = 75\%$

p = 3 $R^2 = 85\%$ " " = 78%

p = 4 $R^2 = 87\%$ " " = 76%

↳ I/p Not Important ⇒ Adjusted R^2 will decrease.