7 days Machine Learning Algorithms

Purpose ÷ Clear the Interviews

Agenda

① Introduction to ML (AI Vs ML Vs DL Vs DS)
② Supervised ML and Unsupervised ML
③ Linear Regression (Maths & Geometric Intuition)
④ R^2 & Adjusted R^2
⑤ Ridge and Lasso Regression

AI application

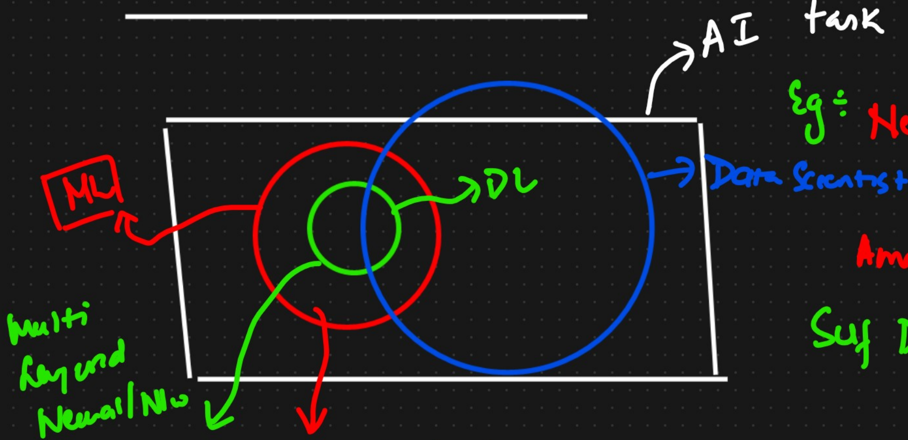
AI application is able to do it own task without any human intervention

Eg ÷ Netflix → Action → Recommendation
→ Comedy → "

Amazon.in → Iphone → Headphones }

Self Driving Cars →

① AI Vs ML Vs DL Vs DS



Mimic Human Brain }

Stats tools to analyze, Visualize, prediction, forecasting

Machine & Deep Learning | Reinforcement

- Supervised ML
  - Regression
  - CLASSIFICATION
- Unsupervised ML → CLUSTERING
  DIMENSIONALITY REDUCTION

Supervised ML

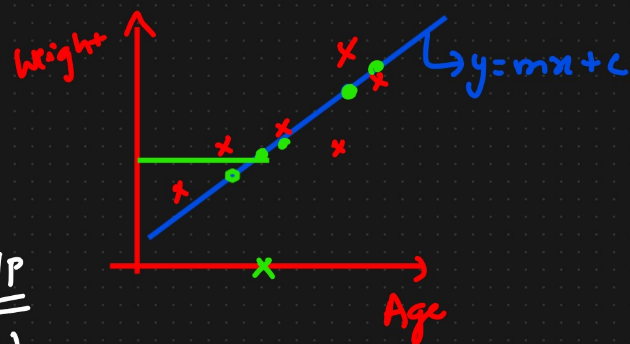
Age	Weight
24	62
25	63
21	72
27	62

Weight → O/P

Age → hypothesis → O/P Weight

Independent features → Age

Dependent feature → Weight



① Regression Problem

Age	Weight
24	72
23	71
25	71.5
—	-

Weight → O/P
↓
Continuous variable
⇓
Regression Problem

② CLASSIFICATION

Binary classification
Multiclass classification

No. of hours	No. of play hours	No. of sleep	O/P P/F
—	—	—	P
—	—	—	F
—	—	—	P
—	—	—	F

② Unsupervised ML

Unsupervised ML → Clustering
→ Dimensionality Reduction

Salary	Age
—	-
—	-

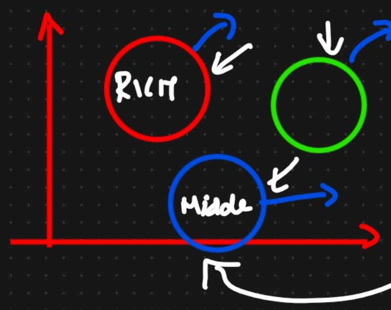
{No Dependent variable}

Clustering → Customer Segmentation

Ad marketing

GROUPING

Customer Segmentation



Eg: Product 1
↳ Rich

Product 2
↳ Middle

② Dimensionality Reduction

1000 → Lower Dimension
↳ 100

⇓

PCA, LDA

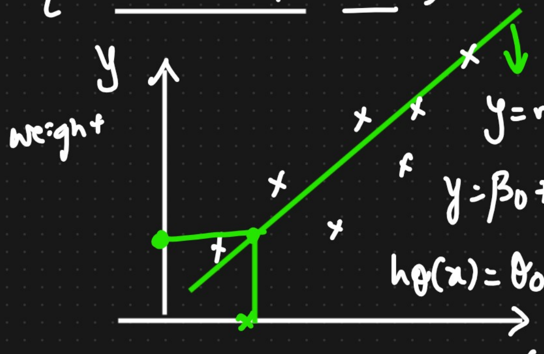
Supervised

① Linear Regression
② Ridge & Lasso
③ Logistic Reg
④ Decision Tree
⑤ AdaBoost
⑥ Random Forest
⑦ Gradient Boosting
⑧ Xgboost
⑨ Naive Bayes
⑩ SVM
⑪ KNN

Unsupervised

① K Means
② DBScan
③ Hierarchical
④ K Nearest Neighbor Cluster
⑤ PCA
⑥ LDA

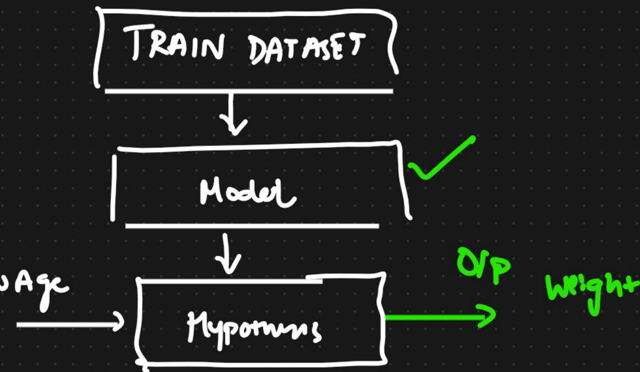
① {Linear Regression}



$y = mx + c$

$y = \beta_0 + \beta_1 x$

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



Credits : Andrew NG

y is a linear function of x

Equation of a straight line

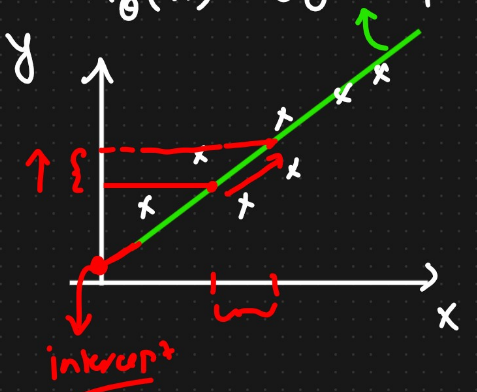
$$h_\theta(x) = \theta_0 + \theta_1 x$$

When $x=0$

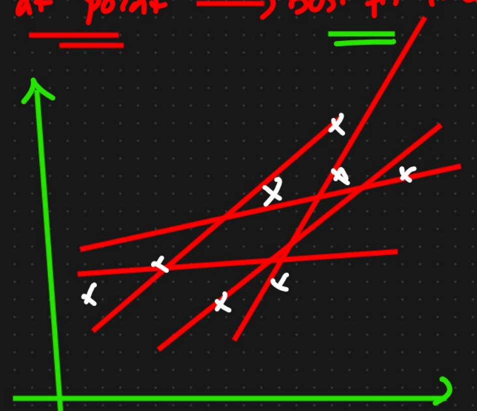
θ_0 = Intercept

θ_1 = Slope or coefficient

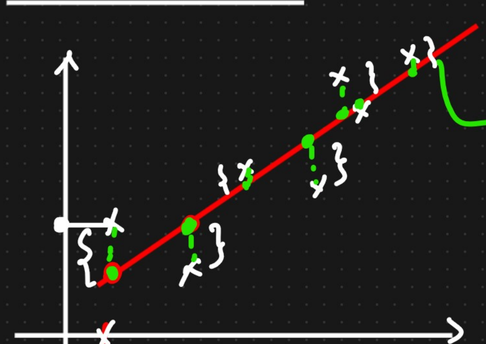
x_i = data points



Start at point ⟶ best fit line



Linear Regression



Minimal ↓↓↓↓

⟶ Best fit line

Hypothesis

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

purpose

{Derivation}

$x^n = nx^{n-1}$

x^2

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

Cost function

$$\boxed{J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2}$$

⟶ Cost function

↳ Squared Error Function

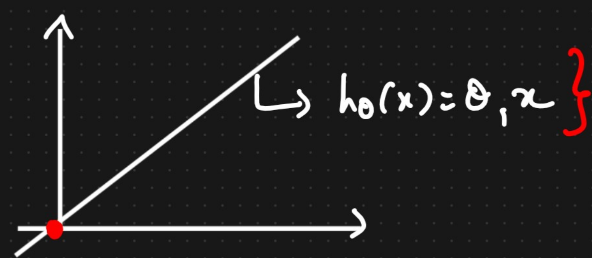
What we need to solve

$$\underset{\theta_0,\theta_1}{\text{minimize}} \ \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)})-y^{(i)}\right)^2$$

$$\Downarrow$$

$$\underset{\theta_0,\theta_1}{\text{minimize}} \ J(\theta_0,\theta_1)$$

⊛ $h_\theta(x) = \theta_0 + \theta_1 x$ If $\theta_0 = 0$

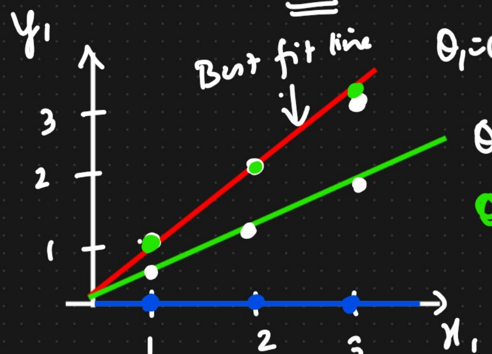


$h_\theta(x) = \theta_1 x$

→ slope

$\theta_1 = 1$ (1, 1) (2, 2) (3, 3)

$\theta_1 = 0.5$

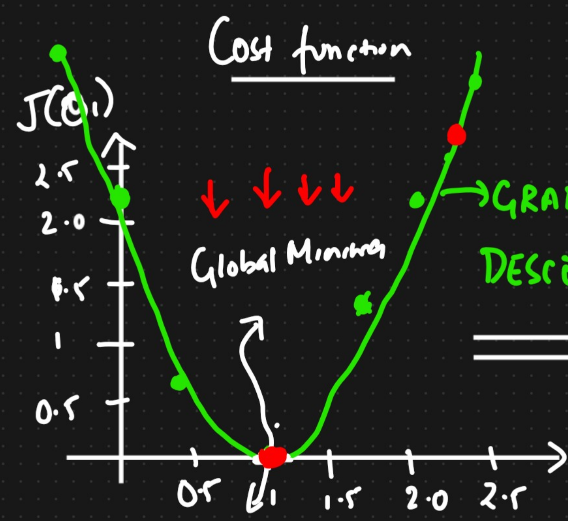


$\theta_1 = 0$

$\theta_1 = 2$

$\theta_1 = 1$

$J(\theta_1) = 0$



GRADIENT DESCENT

$$J(\theta_1) = \frac{1}{2m}\sum_{i=1}^{3}\left(h_\theta(x^{i})-y^{(i)}\right)^2$$

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

$$J(\theta_1) = 0$$

$\theta_1 = 0.5$

$$J(\theta_1) = \frac{1}{2m}\sum_{i=1}^{3}\left(h_\theta(x^{(i)})-y^{(i)}\right)^2$$

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

$$= \frac{1}{2\times 3}\left[0.25+1+2.25\right] \approx 0.58$$

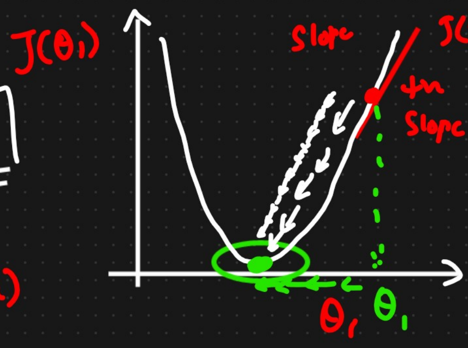
$$J(\theta_1) = \frac{1}{2m}\left[(0-1)^2+(0-2)^2+(0-3)^2\right]$$

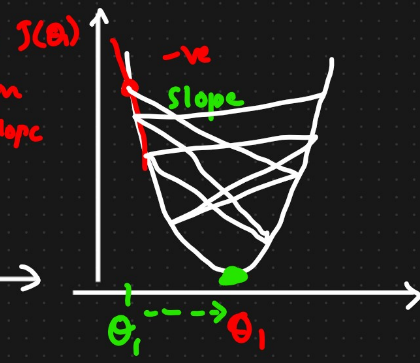
$$= \frac{1}{6}\left[1+4+9\right]$$

$$\approx 2.3$$

$\alpha \Rightarrow$ Huge

$\boxed{\alpha = 0.01}$





Convergence Algorithm

Repeat until convergence

derivative (slope)

{

$$\theta_j := \theta_j - \alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_0,\theta_1)}$$

}

Learning Rate

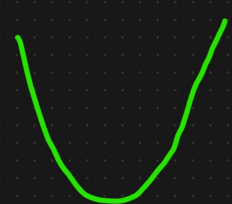
$\theta_1 := \theta_1 - \alpha\,(+ve)$

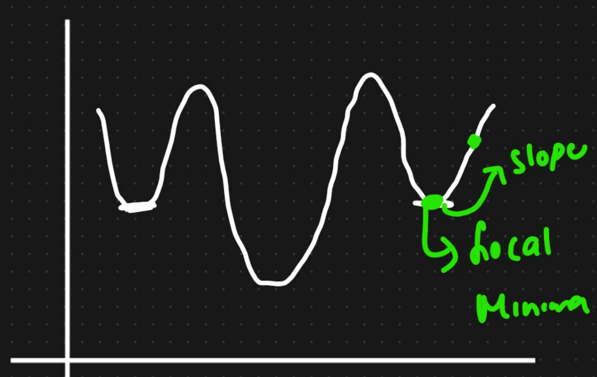
$\theta_1 := \theta_1 - \alpha\,(-ve)$

$\theta_1 := \theta_1 + \alpha\,(ve)$

$\theta_1 := \theta_1 - \alpha\,(0)$

$\theta_1 := \theta_1$





Gradient Descent Algorithm

Repeat until Convergence

{

$$\theta_j := \theta_j - \alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_0,\theta_1)}$$

}

$j = 0$ and 1

$$\frac{\partial}{\partial\theta_j} J(\theta_0,\theta_1) = \frac{\partial}{\partial\theta_j} \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)^2 \qquad \frac{1}{2m}x^2 \checkmark \quad \frac{\cancel{2}}{\cancel{2}m}x$$

Convergence Algorithm.

$$\begin{cases} j=0 \Rightarrow \dfrac{\partial}{\partial\theta_0} J(\theta_0,\theta_1) = \dfrac{1}{m}\displaystyle\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right) \\ j=1 \Rightarrow \dfrac{\partial}{\partial\theta_1} J(\theta_0,\theta_1) = \dfrac{1}{m}\displaystyle\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right) x^{(i)} \end{cases}$$

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

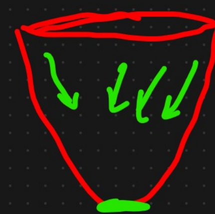
$x^2 \Downarrow 2x$

$\frac{\partial}{\partial\theta_1}(\theta_1 x) = x$

$\{\alpha = 0.001\}$ α = Learning Rate

Repeat until converge

$$\{$$
$$\theta_0 := \theta_0 - \alpha \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)$$
$$\theta_1 := \theta_1 - \alpha \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right) x^{(i)}$$
$$\}$$



Performance Metrics

R^2 and Adjusted R^2

$\boxed{-ve}$???

$h_\theta(x)$

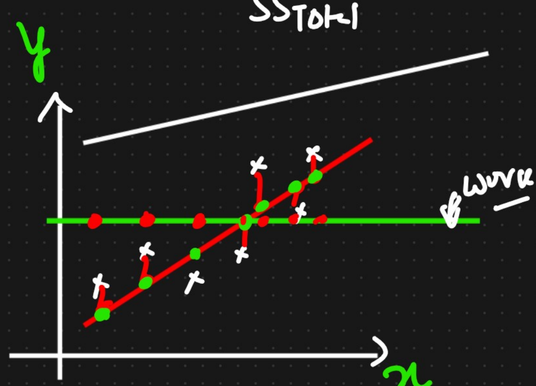
$$R^2 = 1 - \frac{SS_{Res}}{SS_{Total}} = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} = 1 - \frac{Low}{High}$$

small number

= Big number

= 90%

Higher



Gender Bedrooms Location Price

$\times$ correlated

$R^2 = 85\%$

2 $\boxed{R^2 = 90\%}$ ✓✓

3 $\boxed{R^2 = 91\%}$ ✓ → Model

Adjusted R^2

p = features or predictors

✓ ↑↑ Big ✓

$$R^2_{adjusted} = 1 - \frac{(1-R^2)(N-1)}{\boxed{N-p-1}}$$

p=2 > >> p=3

= N = No. of datapoint

P = No. of predictors

p >>>

$\left\{ \begin{array}{l} p=2 \quad = R^2 = 90\% \\ p=3 = \; = R^2 = 91\% \uparrow\uparrow \end{array} \right.$

$R^2_{adjusted} = 86\%$ ↓↓

86%

$R^2_{adjusted} = 82\%$ ↓↓

82%

$R^2 \uparrow\uparrow\uparrow$