Problems handle

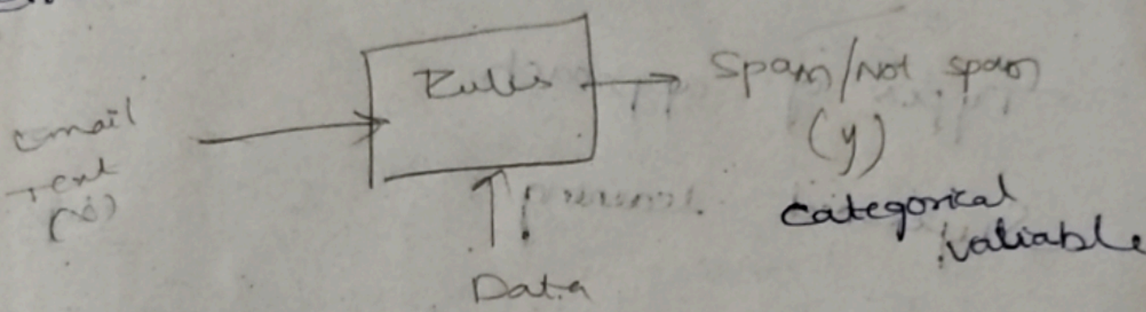
Steps

1. Train
2. test
3. Deploy the model

- classification } Supervised learning
- prediction (Regression)
- clustering } unsupervised
- Dimensionality Reduction

→ classification —— y is categorical

→ Email classification

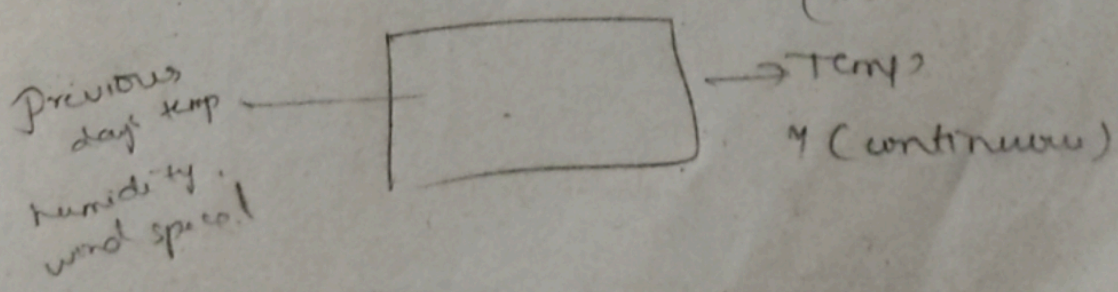


Data

x email	label
Meeting at 10:00am	Not spam
won $1000	spam

→ Regression

(1) Weather prediction



Regression y is continuous

→ Clustering

1) Classification (x, y) → discrete

2) Regression (x, y) → continuous

If only x present

3) Clustering (x)

4) Dimensionality Reduction (Reducing x)

Types of approach

Supervised learning :

1. Learning from solved examples — Training
2. Exercises — validation
3. Exam — Testing

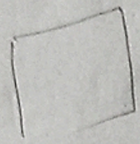
Steps

x → Email; y → Label → Human knowledge (supervision)

Email	Label
meeting at 10	not spam
won lottery	spam

Example:

1) Face Recognition



2) Card fraud Detection

Last 10 transaction, location

3) Disease Prediction

symptoms → $\boxed{}$ → Disease (Y)
X

4) Stock price prediction

previous 10 days price $\boxed{}$ Tomorrow price (Y)
(x)

Dataset:

Split the data into train set, validation set, test set

	X	Y
1		
2		
3		
10		

Train (60%) - 80%

validation (20%) 10%

Test (20%) 10%

Select by Randomization

Train - 5, 10, 1, 6, 2, 7, 4, 3

validation - 9

Test - 8

Regression

Simplest

	No. of hours study	marks
Train	1	50
	2	55
	3	65
	4	70
	5	80
Test	3.5	?

$$X \longrightarrow \boxed{f(x)} \longrightarrow \hat{y} \approx y$$

Model

Linear function :

Simplest Linear function ——— (line)

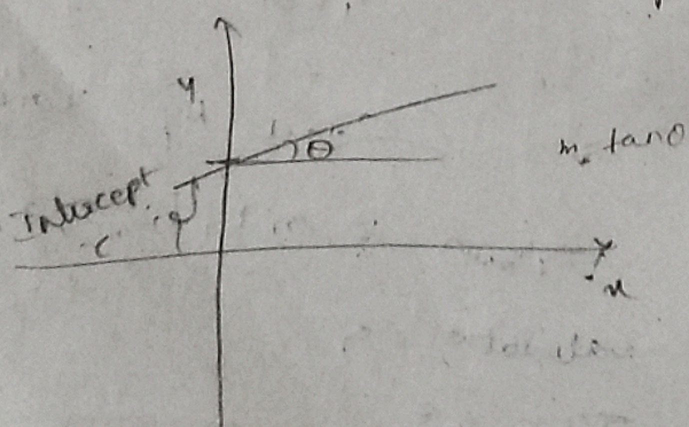
Slope

$$y = mx + c$$

output

Intercept

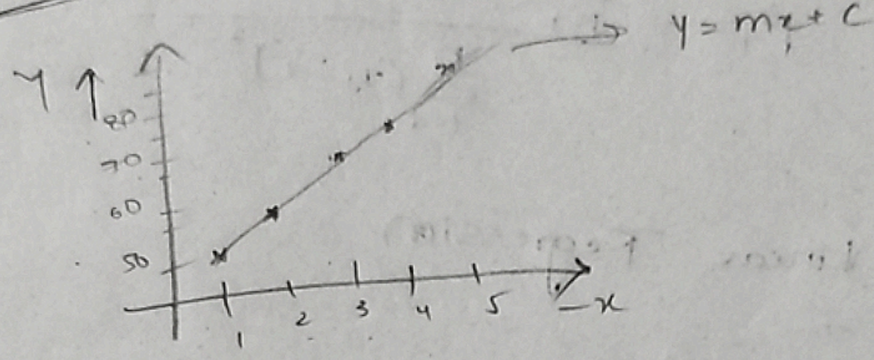
Input



Intercept : tells
where the line is located

* M and c we can control to define the line

Scatter plot



Chatgpt have

175 billion parameter

m & c → parameters

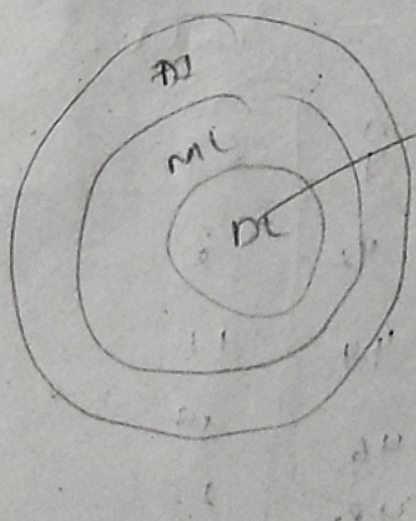
Parameters:

→ Controls the model

$y = mx + c$

m and c are the parameters (2p)

Training = Finding right parameters.



Deep Learning

Neural Network : Inspired by Brain Neurons

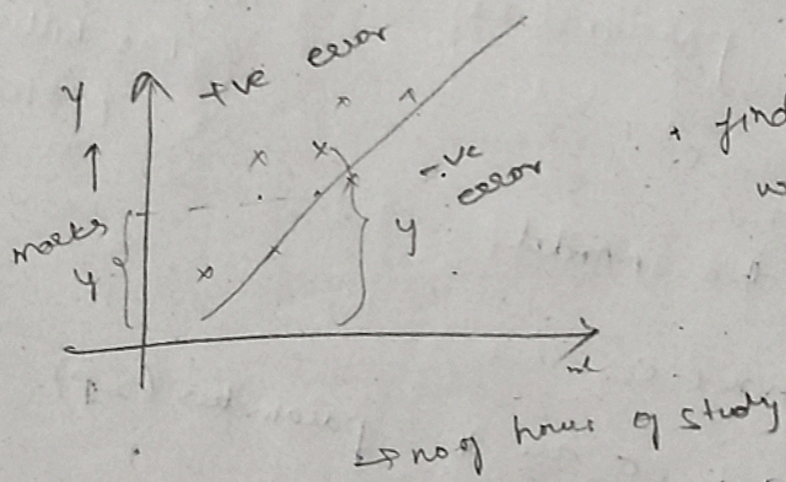
$$y = \beta_0 + \beta_1 x$$

$$\beta_0 = \bar{y} - \beta_1 \bar{x}$$

$$\beta_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

Simple Linear Regression

$$\hat{y} = \theta_0 + \theta_1 x$$



→ find the line which gives minimal error

$$\hat{y} = 4 + 10x$$

$x = 5$, $\left.\begin{array}{l}\hat{y} = 54 \\ y = 60\end{array}\right\}$ error

$$\text{Error} = y - \hat{y}$$

		Model-1		Model-2	
x	y	$\hat{y}$	ε_1	$\hat{y}$	ε_1
1	50	14	36	42	8
2	55	24	31	44	11
3	65	34	31	46	19
4	70	44	26	48	22
5	80	54	26	50	30

Model - 1 : 40 + 10x

Model - 2 : 40 + 2x

Select the model which produce minimal error

Error :

Error = $y - \hat{y}$

$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

= Sum of squared error

(1) All errors will become positive

(2) It magnifies the large errors.

Finding the best parameters :

(Minimum Error)

(1) using Formula .

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

$$\theta_1 = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 + (\sum x_i)^2}$$

① model → param...

② Error

③ Train
↳ Find the parameters for which error is minimum

Regression :

x	y
1	50
2	55
3	65
4	70
5	80

$$\hat{y} = \theta_0 + \theta_1 x$$

Find θ_0 and θ_1 such that the error is minimum

SSE → sum of square errors

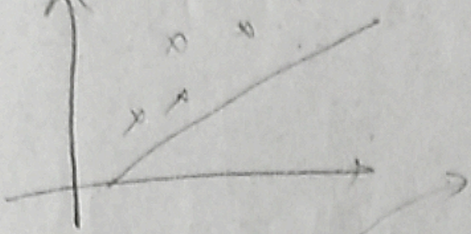
$$y_i - \hat{y}_i$$

$$|y_i - \hat{y}_i|$$

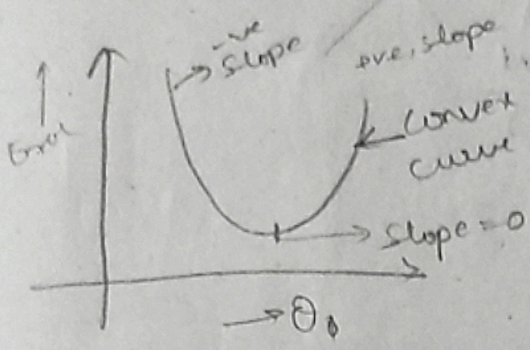
$$SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - \theta_0 - \theta_1 \ldots$$

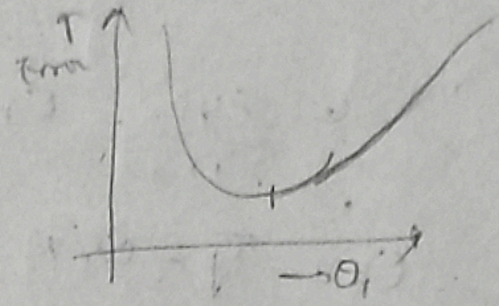
Formula :

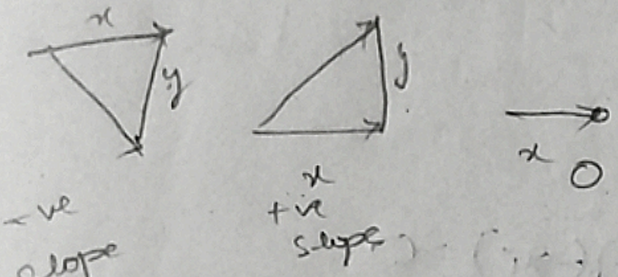
$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$



$\frac{\partial Error}{\partial \theta_0}$ = +ve







change in y with respect to x $\frac{dy}{dx}$ = → derivative

⇒ Find θ_0 and θ_1 such that

$$\frac{\partial Error}{\partial \theta_1} = 0, \quad \frac{\partial Error}{\partial \theta_1} = 0$$

Find θ_0

$$\Rightarrow \frac{\partial SSE}{\partial \theta_0} = 0$$

$$\rightarrow 2\sum(y_i - \theta_0 - \theta_1 x_i)(-1) = 0$$

$$-2\sum(y_i - \theta_0 - \theta_1 x_i) = 0$$

$$\sum(y_i - \theta_0 - \theta_1 x_i) = 0$$

$$\sum y_i - \sum \theta_0 - \sum \theta_1 x_i = 0$$

$$\theta_0 = \frac{\sum y_i - \theta_1 \sum x_i}{n}$$

$$= \frac{\sum y_i}{n} - \theta_1 \frac{\sum x_i}{n}$$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

Find θ_1

$$\frac{\partial SSE}{\partial \theta_1} = 0$$

$$\Rightarrow 2 \sum (y_i - \theta_0 - \theta_1 x_i).(-x_i) = 0$$

$$\Rightarrow -2 \sum (y_i - \theta_0 - \theta_1 x_i)(x_i) = 0$$

$$\Rightarrow \sum (y_i - \theta_0 - \theta_1 x_i) x_i = 0$$

$$\Rightarrow \sum (x_i y_i - \theta_0 x_i - \theta_1 x_i^2) = 0$$

$$\Rightarrow \sum x_i y_i - \theta_0 \sum x_i - \theta_1 \sum x_i^2 = 0$$

$$\Rightarrow \sum x_i y_i - \left(\frac{\sum y_i - \theta_1 \sum x_i}{n}\right) \sum x_i - \theta_1 \sum x_i^2 = 0$$

$$\Rightarrow n \sum x_i y_i - (\sum y_i - \theta_0 \sum x_i) \sum x_i - n\theta_1 \sum x_i^2 = 0$$

$$\Rightarrow n \sum x_i y_i - \sum x_i y_i + \theta_1 (\sum x_i)^2 - n\theta_1 \sum x_i^2 = 0$$

$$\Rightarrow n \sum x_i y_i - \sum x_i \sum y_i = \theta_1 (n \sum x_i^2 - (\sum x_i)^2)$$

$$\theta_1 = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\quad}$$

$m \times n \quad = \quad n \times 1$

Regression:

Estimating θ_0 and θ_1 using formula:

$= 41.5$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

$$\theta_1 = \frac{n\sum xy - \sum x \sum y}{n\sum x^2 - (\sum x)^2} = 7.5$$

- Estimating $\theta = \{\theta_0, \theta_1\}$ using normal form

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

Mean of $x = \bar{x} = \frac{1+2+3+4+5}{5} = \frac{15}{5} = 3$

$= X @ \theta$

$\bar{y} = \frac{50+55+65+70+80}{5} = 64$

→

	x	y	$\hat{y}$
1, 1	1	50	49
	2	55	56.5
1, 2	3	65	64
1, 3	4	70	71.5
1, 4	5	80	79
1, 5			

$$y = \begin{bmatrix} 50 \\ 55 \\ 65 \\ 70 \\ 80 \end{bmatrix}$$

$$\hat{y} = \theta_0 + \theta_1 x = \theta_0 \cdot 1 + \theta_1 \cdot x$$

$$= \underbrace{\begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 1 & 5 \end{bmatrix}}_{X} \times \underbrace{\begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}}_{\theta}$$

$$\theta = (x^T x)^{-1} x^T y$$

→ $\hat{y} = X\theta$

→ $X = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 1 & 5 \end{bmatrix}$ $\quad y = \begin{bmatrix} 50 \\ 55 \\ 65 \\ 70 \end{bmatrix}$

$(5 \times 2)^T \quad 5 \times 1$

$2 \times 5 \quad 5 \times 1$

$(x^T x)^{-1} \cdot x^T y$

2×1

2×2

Code.

Python: import numpy as np

```
x = np.array([[1, 1], [1, 2], [1, 3], [1, 4], [1, 5]])
y = np.array([50, 55, 65, 70, 80])
theta = np.linalg.inv(X.T @ x) @ X.T @ y
print(theta)
```

theta. [41.5, 7.5]

$$\Rightarrow SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)$$

$$\therefore Y = \begin{bmatrix} 50 \\ 55 \\ 65 \\ 70 \\ 80 \end{bmatrix} \quad \therefore \hat{y} = \begin{bmatrix} 49 \\ 56.5 \\ 64 \\ 71.5 \\ 79 \end{bmatrix}$$

$$= (y - \hat{y})^T (y - \hat{y}) \quad \Rightarrow y - \hat{y} = \begin{bmatrix} 1 \\ -1.5 \\ 1 \\ -1.5 \\ 1 \end{bmatrix}^T \begin{bmatrix} 1 \\ -1.5 \\ 1 \\ -1.5 \\ 1 \end{bmatrix}$$

$$= (Y - X\theta)^T (y - X\theta)$$

$$= y^T y - y^T X\theta - \theta^T X^T y + \theta^T X^T X\theta$$

∂SSE

$1\times5 . 5\times2 . 1\times2$	$1\times2 . 2\times5 . 5\times1$
1×1	1×1
1 term	1 term

$$= y^T X\theta = \theta^T X^T y$$

$$= y^T y - \theta^T X^T y - \theta^T X^T y + \theta^T X^T X\theta$$

ML-1

→ ML is a subset of AI.
→ enables computers to learn from data and make decisions or predictions

* Learn from historical data
* Identify patterns and relationships within the data
* predict outcomes or actions on new data

Key features.

* Data-Driven
* Iterative
* Adaptive

Types of Machine Learning :

* Supervised Learning
  learning from label data.
* Unsupervised learning
  learning from unlabeled data
* Reinforcement learning
  Learning through trial and error

x_i	y_i	$x_i - \bar{x}$	$y_i - \bar{y}$		
1	50	-2	-14	28	4
2	55	-1	-9	9	1
3	65	0	1	0	0
4	70	1	6	6	1
5	80	2	16	32	4

1) Sum of $(x_i - \bar{x})(y_i - \bar{y})$:

$$\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) = 28 + 9 + 0 + 6 + 32 = 75$$

2) sum of $(x_i - \bar{x})^2$

$$\sum_{i=1}^{n} (x_i - \bar{x})^2 = 4 + 1 + 0 + 1 + 4 = 10$$

Slope θ_1 is

$$\theta_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$\theta_1 = \frac{75}{10} = 7.5$$

3) Calculate the Intercept

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

$$\theta_0 = 64 - (7.5)(3)$$

$$\theta_0 = 64 - 22.5 = 41.5$$

4) Final equation:

$$\boxed{\hat{y} = 41.5 + 7.5x}$$

Evaluation metric

What is the performance on test data?

x	y

} train

} test

$\hat{y} = f(x)$

error $\rightarrow$ $SSE = \sum_{i=1}^{n} (y_i - \hat{y}_i)$

$MSE = \frac{SSE}{n}$

$RMSE = \sqrt{mse}$

Evaluation metric

R^2

Adjusted R^2

If variance is less data is located near to mean.

$\rightarrow$ If variance is high data is located away from mean

$$\Rightarrow R^2 = \frac{\overbrace{\sum (y_i - \hat{y})^2}^{\text{Explained variance}}}{\underbrace{\sum (y_i - \bar{y})^2}_{\text{variance in data}}}$$

ex.:

x	y	$\hat{y}$	$y - \hat{y}$	$(y - \hat{y})^2$	$(y - \bar{y})$	$(y - \bar{y})$
	50	49.5	1	1	-14	196
	55	56.5	-1.5	2.25	-9	81
	65	64	1	1	1	1
	70	71.5	-1.5	2.25	6	36
	80	79	1	1	16	256
				7.5		570

$R^2 =$

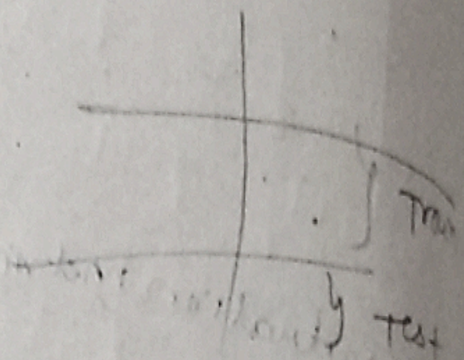
$\bar{Y} = 64$.

$R^2 = \frac{7.5}{570} =$

0.98

Classification

Evaluation Metric :



x	y	$\hat{y}$

2) → Binary classification

→ only 2 labels (spam, not spam)
1, 0

→ Confusion matrix → predicted label

	1	0
1	TP	FN
0	FP	TN

Spam and marked as spam ← TP

FN → More dangerous

3) actual label

FP: Not spam but predicted as spam

TN: not spam and marked as not spam

Cancer Prediction

Sympton → [ ] → Cancer,

Classification:

- → Accuracy
- → Precision
- → Recall
- → F1 score

Y	$\hat{Y}$	
1	1	✓
0	1	×
1	0	×
1	1	✓
0	1	×
1	1	✓
0	1	×
0	0	✓

$$\text{Accuracy} = \frac{\text{Total correct predictions}}{\text{Total predictions}}$$

$$= \frac{TP + TN}{TP + TN + FP + FN}$$

$$= \frac{4}{8} = 0.5 = 50\%$$

To measure uncertainty $\{P_1, P_2\}$

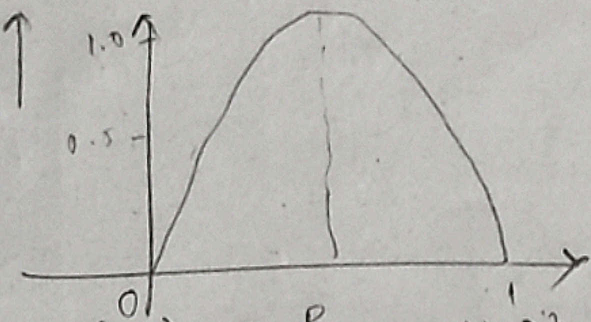
$\{50\%, 50\%\} \rightarrow$ more uncertain[ty]

$\{10, 10\} \rightarrow$ more impurity

$\{90\%, 10\%\} \rightarrow$ less uncertainty

$\{90, 10\}$ less impurity

Entropy $= -P_1 \log_2 P_1$
$= -P_2 \log_2 P_2$



$\{0,1\} \rightarrow P_1$ $\{0.5, 0.5\}$ $\{1, 0\}$

$\{P_1, P_2\}$

x_1	x_2	x_3	x_4	label

$E(x_1)$

$E(x_2)$

$E(x_3)$

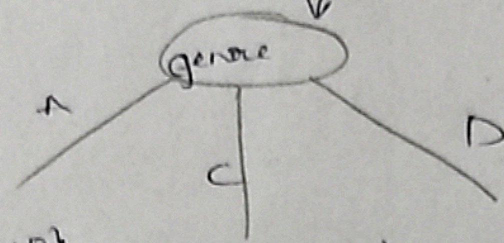
$E(x_4)$

Gentile	Time	watch movie
A	E	Y

Information Gain

{Y : 2, N : 3}

E(root) = 0.97



{Y : 2, N : 0} {Y : 0, N : 2} {Y : 0, N : 1} splits

weight → 2/5 2/5 1/5

$E(Genre_A)$ $E(Genre)_C$ $E(Genre_D)$

= 0 = 0 = 0

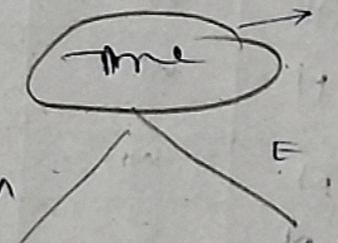
Average Entropy $= \frac{2}{5} \times 0 + \frac{2}{5} \times 0 + \frac{1}{5} \times 0$

$= 0$

IG (Entropy) : Reduction in uncertainity / impurity

$= 0.97 - 0 = 0.97$

{Y : 2, N : 3}



E(root = 0.97

{Y : 1, N : 1} {Y : 1, N : 2}

2/5 3/5

E(Time m) E(Time E)

= 1 = 0.91

Average Entropy (of children) $= \frac{2}{5} \times 1 + \frac{3}{5} \times 0.91$

$= 0.946$

IG (Time) $= 0.97 - 0.946 = 0.024$

Will you play tennis?

Day	Outlook	Temp	Humidity	Wind	Play tennis?
1	S	H	H	N	N
2	S	H	H	S	N
3	O	H	H	W	Y
4	R	M	H	W	Y
5	R	C	N	W	Y
6	R	C	N	S	N
7	O	C	N	S	Y
8	S	M	H	W	N
9	S	C	N	W	Y
10	R	M	N	W	Y
11	S	M	N	S	Y
12	O	H	H	S	Y
13	O	~~M~~ H	N	W	Y

S→Sunny

O→Overcast   M→Mild   N→Normal   W→Weak   N→NO

R→Rainy   C→Cool

Here, Y : 9 , N = 5

outlook

S   O   R

Pivot node

{Y:2, N:3}   {Y:4, N:0}   {Y:3, N:2}

5/14   4/14   5/14

E(outlook$_S$) = 0.97   E(outlook$_O$) = 0

E(outlook$_R$) 0.97

Average entropy $= \frac{5}{14} \times 0.97 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.97$

$= 0.69$

I_G (outlook) = 0.94 − 0.69 = 0.24

temp

H   M   C

{Y: 3, N: 2}   {Y:3, N: 2}   {Y: 2, N: 2}

= 0.97   = 0.97   = 0.81

Average Entropy $= \frac{5}{14} \times 0.97 + \frac{5}{14} \times 0.97 + \frac{4}{14} \times 0.81$

$= 0.924$

I_G (Temp) = 0.94 − 0.924 = 0.016

Humidity

H N

{Y:3, N:4} {Y:6, N:1}

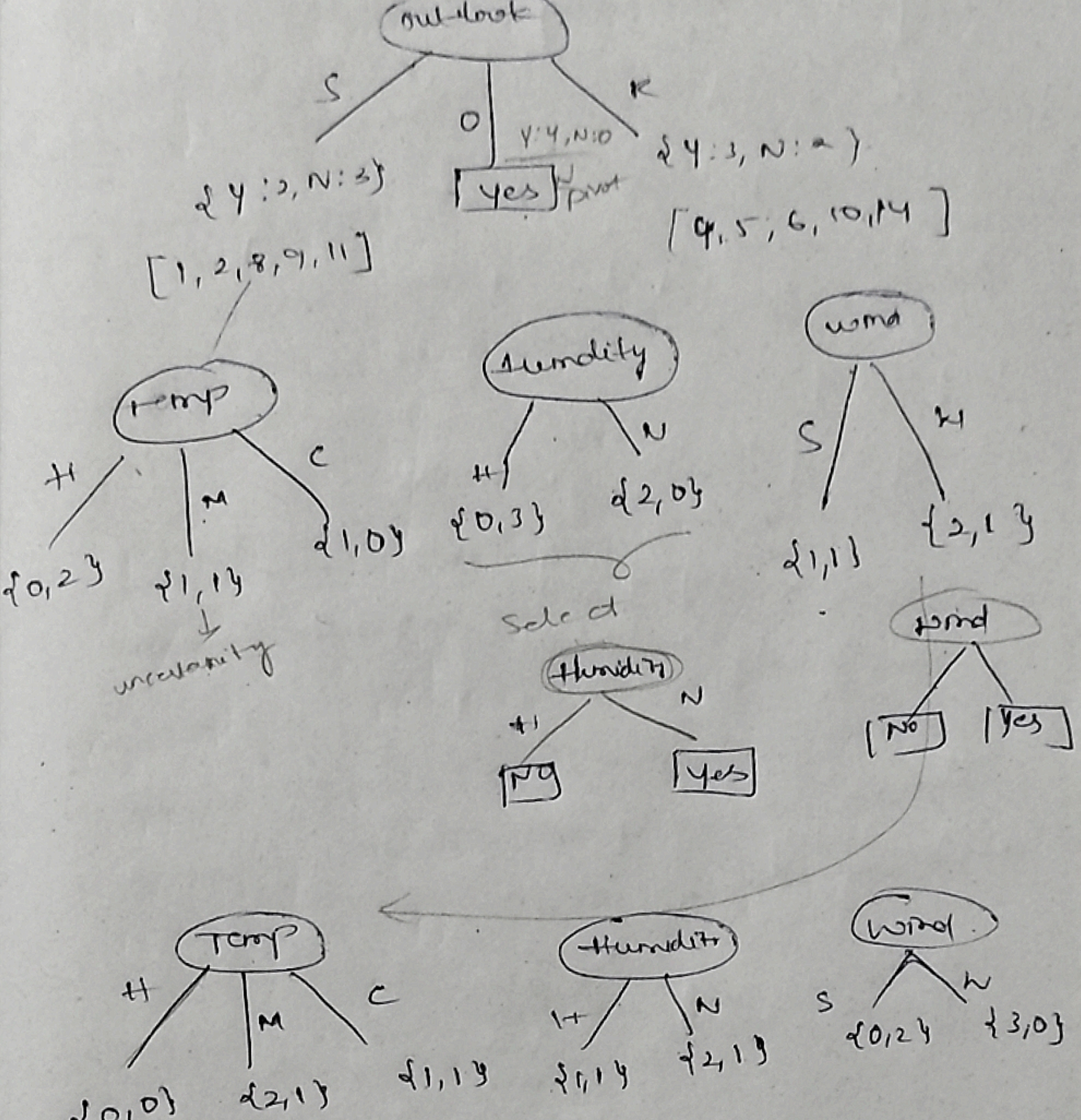
I.G (Humidity) = 0.16

Wind

S N

{Y: , N: } {Y: , N: }

I.G (Wind) = 0.06



Outlook
S
O
R
Y:4, N:0
Yes
{Y:2, N:3}
{Y:3, N:2}
[4, 5, 6, 10, 14]
[1, 2, 8, 9, 11]
Temp
H
M
C
{0,2}
{1,1}
{1,0}
uncertainity
Humidity
H
N
{0,3}
{2,0}
Wind
S
W
{1,1}
{2,1}
Select
Humidity
H
N
No
Yes
Wind
No
Yes
Temp
H
M
C
{0,0}
{2,1}
{1,1}
Humidity
H
N
{1,1}
{2,1}
Wind
S
W
{0,2}
{3,0}