*Probabilistic Approach and PCA

$$D = \{ (x_1, y_1), (x_2, y_2), \dots \dots (x_n, y_n) \}$$

↳ labels

Train a new function

$y = f(x)$
actual

Predicted → $\hat{f}$: algorithm training : best

↙ close to f

If x near D → less error
If x not near D → high error

$\hat{f}(x; D)$
↳ training data

$$E_D[(y - \hat{f}(x;D))^2]$$ } average error / expected error D

↳ Various Datasets
↳ check error over x
↓
take average
then subtract from y
So we get average ERROR

Variance:-
$E(\text{Prediction} - \text{Expected})^2$
Prediction = made by model for specific training set
expected Prediction – avg prediction made by model across many training sets.

{ $f(x)$: True Label }

{ $\hat{f}$ → predicted
E → Expected }

* Bias Variance Decomposition:-

perfect model we learnt
fixed point
training set

Variance – variance of $\hat{f}(x; D)$ can be defined as follows.

$$\mathrm{Var}_D[\hat{f}(x;D)] = E_D[(\underbrace{E_D[\hat{f}(x;D)] - \hat{f}(x;D)}_{y})^2]$$

y^2

$y = E_D[\hat{f}(x;D)] - \hat{f}(x;D)$

$\hat{f}(x;D)$ → specific

$E_D[\hat{f}(x;D)]$ → range : expected outputs

difference in expected and specific output

High Variance → Sensitivity to D (data points)
High Variance → overfitting → too complex
Low Variance → Stable across all

Overfit → model learns noise and details
→ performs good on training data poor on unseen

Training error : Low
Test Error : High

$\mathbb{E}$ → denote expected value or mean

Bias: We also define Bias

$$\text{Bias}_D[\hat{f}(x;D)] = \underbrace{E_D[\hat{f}(x;D)]}_{\text{Expected}} - \underbrace{f(x)}_{\text{True}}$$

High Bias → High Inductive Bias → Underfitting

High Bias → too simplistic → underfits (does not capture all details)
Low Bias → Close to true value

Bias and Variance Decomposition

$$\underbrace{E_D[(y-\hat{f}(x;D))^2]}_{\text{Average MSE}} = \underbrace{(\text{Bias}_D[\hat{f}(x;D)])^2}_{(\text{Bias})^2} + \underbrace{\text{Var}_D[\hat{f}(x;D)]}_{\text{Variance}}$$

Average MSE
mean Squared error

{ Can be Hyperparameter }

Two Sources of error

In reality,

$$E_{D,\epsilon}[(y-\hat{f}(x;D))^2] = (\text{Bias}_D[\hat{f}(x;D)])^2 + \text{Var}_D[\hat{f}(x;D)] + \sigma^2$$

$E_{D,\epsilon}$ → Data and noise

→ mean Squared Error

$y = f(x) + \underbrace{\epsilon}_{\text{noise}}$

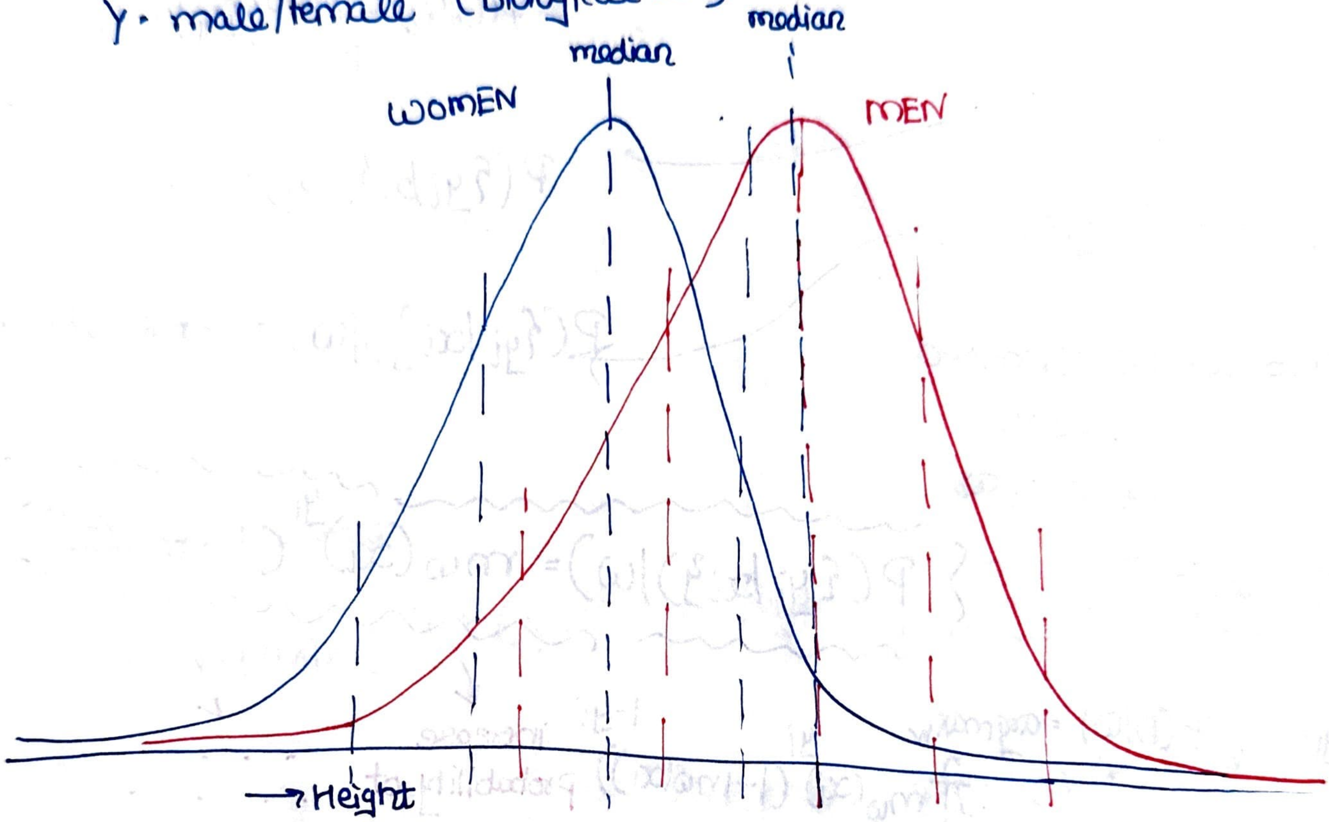
→ average
$E(\epsilon) = 0$
$\text{Var}(\epsilon) = \sigma^2$ → sigma square

σ^2 → little random noise

* The maximun Likelihood View :-

x :- height

y · male/female (biological sex)



given individual with height x is what male/female?

→ probability

$$X \sim P(y/x)$$

→ y or x

$$m_w(x) \approx P(y|x)$$

↙ model

Logistic Neuron

$$x \longrightarrow \text{LN} \longrightarrow \hat{P}(y/x)$$

$m_w(x)$: W: best possible W's

D: data

which w has highest Prob

D: Data

$$\hat{w} = \text{argmax}_W \, P(W/D)$$

$$P(W/D) = \frac{P(D|W)\,P(W)}{P(D)}$$

$P(D)$ → independent of set of weights

$$\text{argmax}_W \, P(D|W)\,P(W)$$

*Deriving the nll loss

IID → identically distributed
~~independent~~ distribution. assumption

$$P(D|\omega) = \prod_{j=1}^{n} P(\{y_i|x_i\}|\omega)\}$$

independence

If $y_i = 1$, then

$$P(\{y_i|x_i\})|\omega) = m\omega(x_i)$$

otherwise, if $y_i = 0$

$$P(\{y_i|x_i\})|\omega) = 1 - m\omega(x_i).$$

This can be expressed in one equation as :

IMPORTANT

$$\left\{ P(\{y_i|x_i\})|\omega) = m\omega(x_i)^{y_i} (1 - m\omega(x_i))^{1-y_i} \right.$$

$$\text{argmax}_w P(D|\omega) = \text{argmax}_w \prod_{j=1}^{n} m_\omega(x_i)^{y_i} (1 - m_\omega(x_i))^{1-y_i}$$

increase probability of true label

$y_i = 1$ $y_i = 0$

Equivalent → $\text{argmax}_w \log P(D|W)$

Example:- Find Email Spam/Not Spam

$y = 1$ → Spam

$y = 0$ → Not Spam

Email ID	Features	Label (y)	Pred Probability
1	[5,10,3]	1-Spam	0.9
2	[2,4,1]	0	0.2
3	[3,6,2]	1-Spam	0.7

Email 1 → $y = 1$, $m\omega(x) = 0.9$

$$P(1|x_1, \omega) = m\omega(x_1)^1 (1 - m\omega(x_1))^0$$

$$= 0.9^1 \cdot (1 - 0.9)^0 = 0.9$$

Maximize

arg max_ω

$$\sum_{j=1}^{n} (y_i (m w(x_i) + (1 - y_i)(1 - m_y) x_i)$$

Maximize → minimize its negation

$\text{argmax}_w P(W|D)$

*Naive Bayes Classifier

3	5	3	6	1	7
9	4	0	9	1	2

X: 28x28 Pixels.

[0:255] greyscale

[0,1] → 784 pixels

$y_i \{0, \dots 9\}$

labels → numbers 0,1,2...9

Probability Distribution

$P(Y/X)$

↳ input

$$\hat{y} = \arg\max_y (P(y/x)) = \arg\max_y \frac{P(X/y)\,P(y)}{P(X)}$$

$$= \arg\max_y P(X/y) \cdot P(y)$$

Estimate

$$P(x/y) = \prod_{t=1}^{784} P(X_t/y)$$

$n_{1,5,t}$ → label 5 → if its t ↓ current pixel

$$P\left(X_t = 1 \,\middle|\, \begin{matrix} y=0 \\ y=1 \\ \vdots \\ y=9 \end{matrix}\right) \quad t = 1 \dots 784$$

Given that pixel is 1 for label 5

How many such pixels

$P(X_t = 1 / y = 5)$

Pixel t

n_5 · # of images with label 5

$n_{0,5,t}$ → how may pixels 0

$n_{1,5,t}$ → how may pixels 1

$$\frac{n_{1,5,t}}{n_5}$$