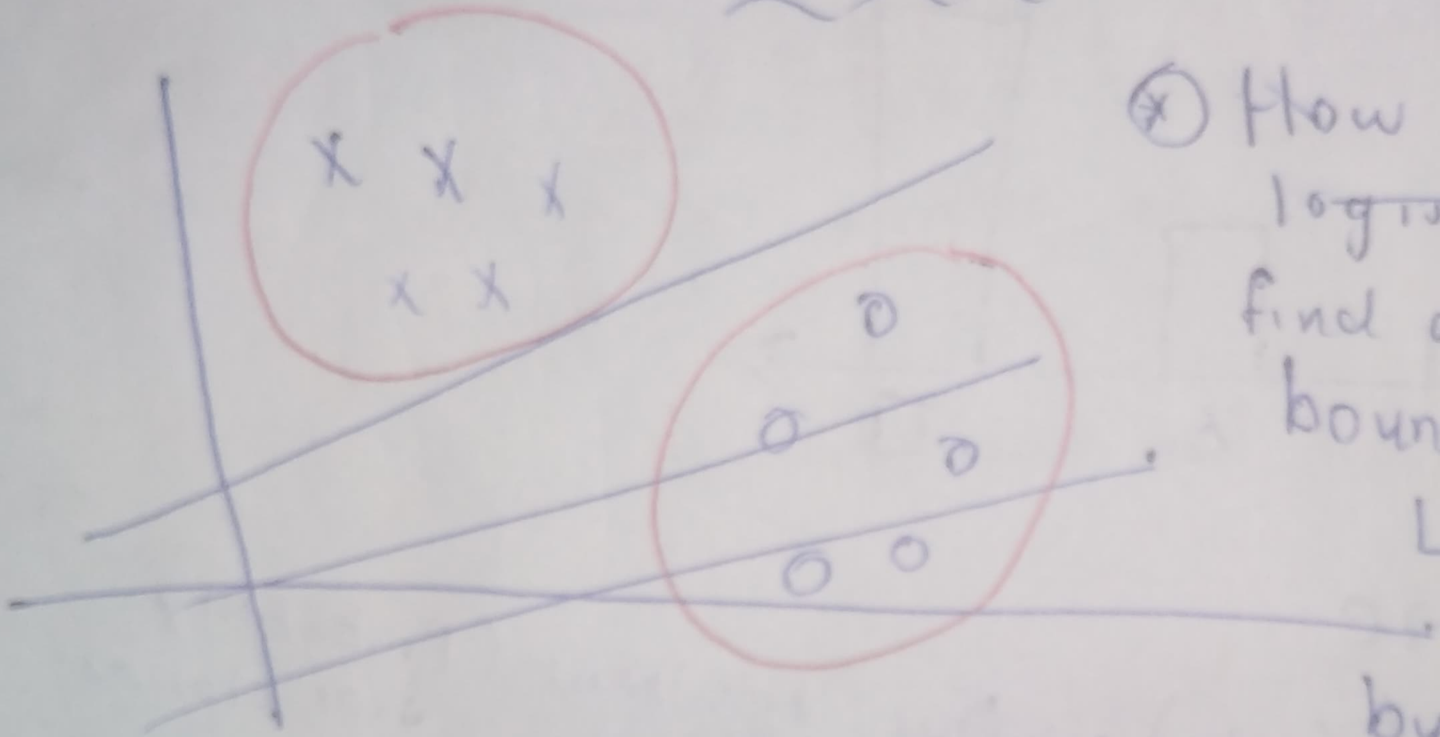
Lecture-5



* How would logisl Discriminati find a decision boundry?
  ↳ $\boxed{\text{Discriminativ}}$ model work by finding a decision boundry.

* The read cluster shows how $\boxed{\text{generative}}$ border works

* Rather than looking simeoullaneously at two Classes to find a decision boundry (Discriminative model); Generative model builds a model of what of the classes look like

Discriminative

Learn $P(Y|X)$

(or $\overset{\text{learn}}{\downarrow} h_\theta(x) = \begin{cases} 0 \\ 1 \end{cases}$ directly).

Generative learning algorithm:

Learns $\boxed{P(X|Y)}$

Features — class.

Also Learns $\boxed{P(Y)}$

class prior.

Bayes Rules (GLA): → Testing Times:-

$$P(y=1|X) = \frac{\boxed{P(X|y=1)} \cdot \boxed{P[y=1]}}{P(X)}.$$

$$P(x) = P(X|y=1) \cdot P(y=1) + P(X|y=0) \cdot P(y=0).$$

Gaussian discriminant Analysis (GDA):-

- Suppose $x \in \mathbb{R}^n$ (drop $X_0 = 1$ convention).
- Assume $P(X|Y)$ is Gaussian.
-

• What is $\boxed{\text{multi-variate gaussian?}}$

⊛ $z \sim \mathcal{N}(\vec{\mu}, \Sigma)$

⊛ $z \in \mathbb{R}^n, \mu \in \mathbb{R}^n, \Sigma \in \mathbb{R}^{n \times n}$

$$E[z] = \mu$$

$$Cov(z) = E[(z-\mu)(z-\mu)^T]$$

$$= \boxed{E\, zz^T - (Ez)(Ez)^T}$$

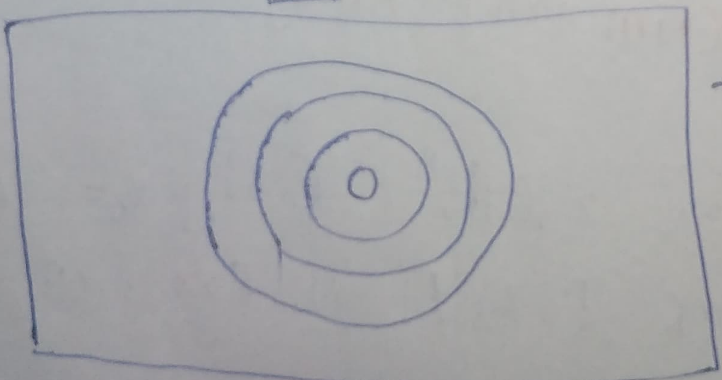
$$p(x) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right)$$

PDF for multi-variate gaussian

→ Contor for Multivariate gaussian

$$\rightarrow \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; \mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

z_1

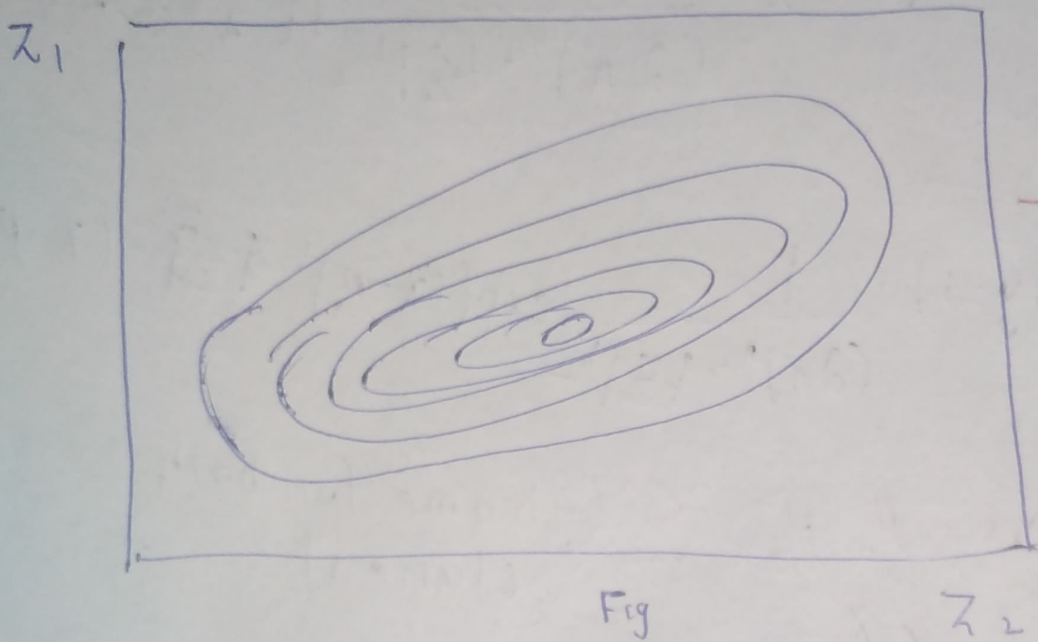


→ Prefectly Rounded Circles.

Fig.

z_2

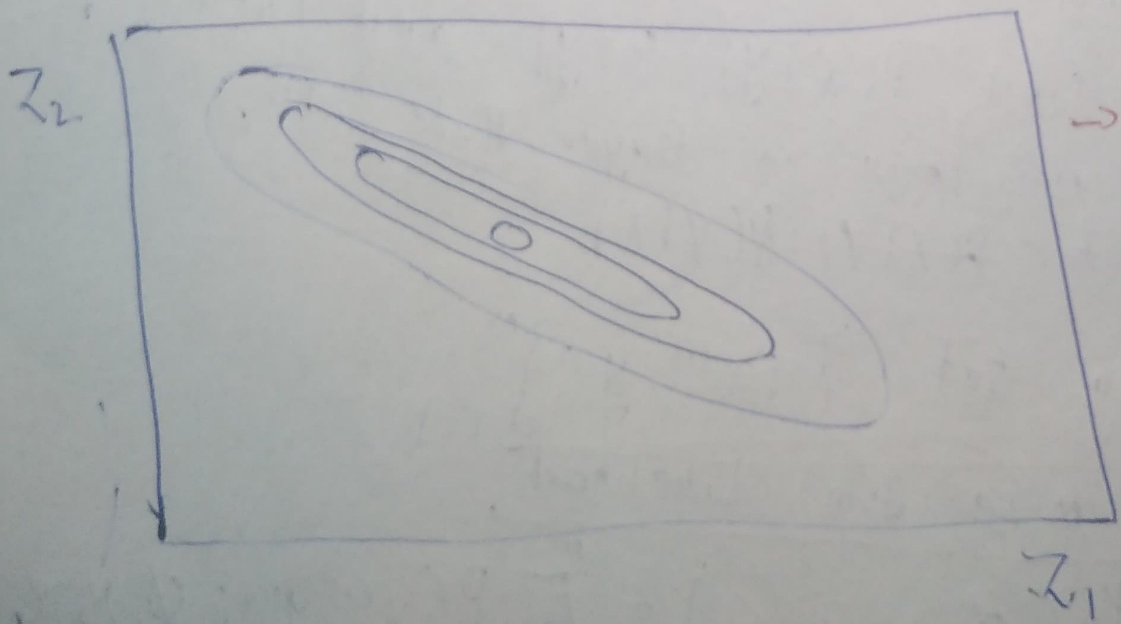
(x) $\Sigma = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$; $\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$



→ z_2 and z_1 being positively correlated.

Fig

(x) $\Sigma = \begin{bmatrix} 1 & -0.5 \\ -0.5 & 1 \end{bmatrix}$; $\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$



→ z_1 and z_2 being negatively correlated.

* GDA model:-

* $$P(x \mid y=0) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \cdot \exp\left(-\frac{1}{2}(x-\mu_0)^T \Sigma^{-1} (x-\mu_0)\right)$$

* $$P(x \mid y=1) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \cdot \exp\left((x-\mu_1)^T \cdot \frac{1}{2} \Sigma^{-1} (x-\mu_1)\right)$$

Need to find these

- Parameters:- μ_0, μ_1, Σ (Same for both classes).

$\emptyset$

* $P(y) = \emptyset^{y} (1-\emptyset)^{1-y}$ (Bernouli Variable)

* We find these parameter and then find $P(x \mid y)$, $P(y)$ and then will use these in Bayes Rule to predict ~~$P(X \mid Y)$~~ $\cdot$ $P(Y \mid X)$.

* Training set $\{(x^{(i)}, y^{(i)})\}_{i=1}^{m}$

* $\boxed{\text{Maximize Joint Likelihood:-}}$

$$\mathcal{L}(\emptyset, \mu_0, \mu_1, \Sigma) = \prod_{i=1}^{m} P(x^{(i)}, y^{(i)}; \emptyset, \mu_0, \mu_1, \Sigma)$$

$$= \prod_{i=1}^{m} P(x^{(i)} | y^{(i)}) \cdot P(y^{(i)})$$

→ But Remeber for $\boxed{\text{Discriminative model}}$ we will maximizing $\boxed{\text{Conditional likelihood}}$:-

$$L(\theta) = \prod_{i=1}^{m} P(y^{(i)} | x^{(i)}; \theta).$$

→ Maximum likelihood Estimation for GDA:-

$$\max_{\phi, \mu_0, \mu_1, \Sigma} = \underbrace{l(\phi, \mu_0, \mu_1, \Sigma)}$$

$$= \log\left(\prod_{i=1}^{m} P(x^{(i)} | y^{(i)}) \cdot P(y^{(i)})\right)$$

$$\rightarrow \nabla_{\phi, \mu_0, \mu_1, \Sigma} \, l(\phi, \mu_0, \mu_1, \Sigma) = 0$$

↳ You derivative for $l(\cdot\cdot)$ w.r.t to each of the paramiter and set to zero. If you do this then for each of the parameter, the following equation pops-up:-

① $$\phi = \frac{\sum_{i=1}^{m} (y^{(i)})}{m} = \frac{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}}{m}.$$

→ Indicator notation (True if condition holds else False).

2

$$\mu_0 = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=0\} \cdot x^{(i)}}{\cancel{\sum_{i=1}^{m} x^{(i)}\, m} \sum_{i=1}^{m} 1\{y^{(i)}=0\}}$$

Same explanation.

3

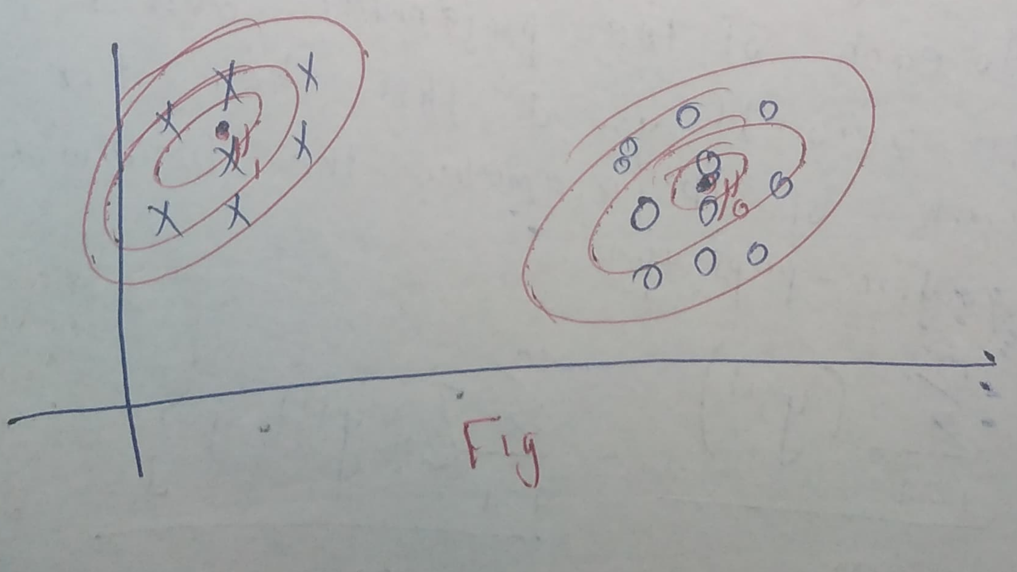
$$\mu_1 = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=1\} \cdot x^{(i)}}{\sum_{i=1}^{m} 1\{y^{(i)}=1\}}$$

Sum of feature vector for example with y=1

# of example with y=1

4

$$\Sigma = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T$$



Fig

Prediction Time :- (GDA) :-

$$\arg\max_y P(y|x) = \frac{\arg\max_y P(x|y) \cdot P(y)}{P(x)}$$

$\min_z (z-5)^2 = 0$, $\arg\min_z (z-5)^2 = 5$

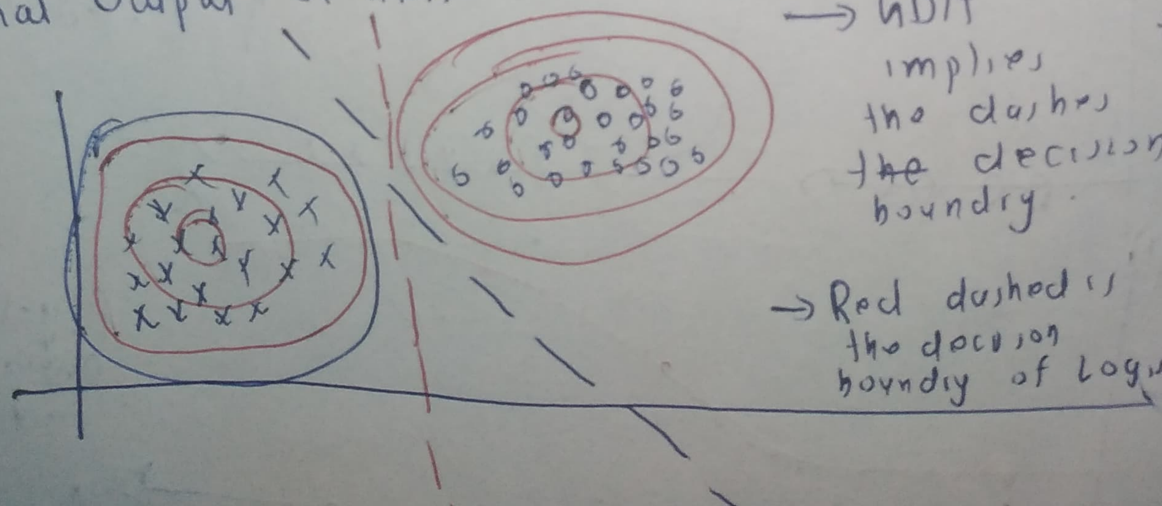
↳ Argmin defines the value you need to plug in to get the minimum.

→ What is argmin/argmax?

$$\arg\max_y (P(y|x)) = \arg\max_y (P(x|y) \cdot P(y))$$

↳ Denominator is not needed in case of argmax (It's a positive value).

Final Output of GDA :-



→ GDA implies the dashes the decision boundry.

Quite same but points is different

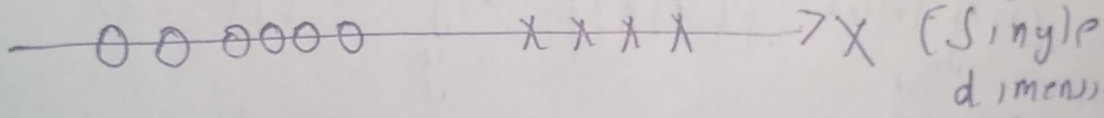
→ Red dashed is the decision boundry of Logistic Regression.

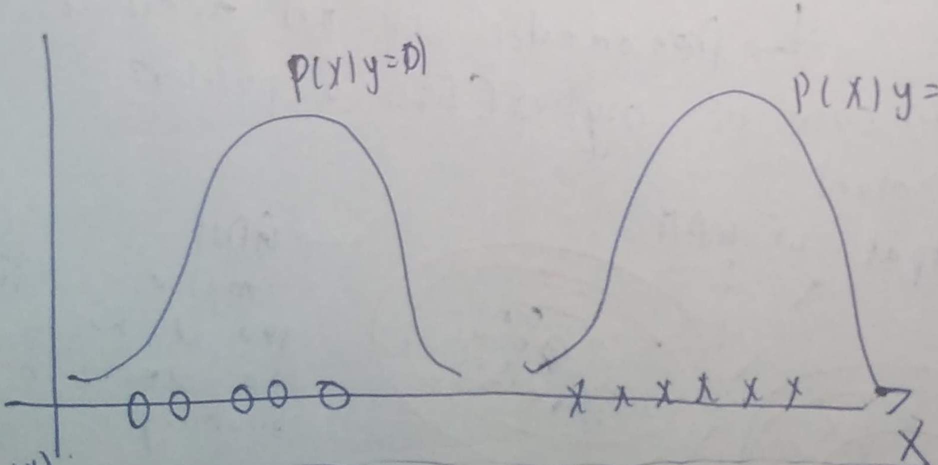
* Compare GDA to logistic regression:

* For fixed $\phi, \mu_0, \mu_1, \Sigma$ lets plot $(P(y=1 \mid x; \phi, \mu_0, \mu_1, \Sigma)$ as a function of x

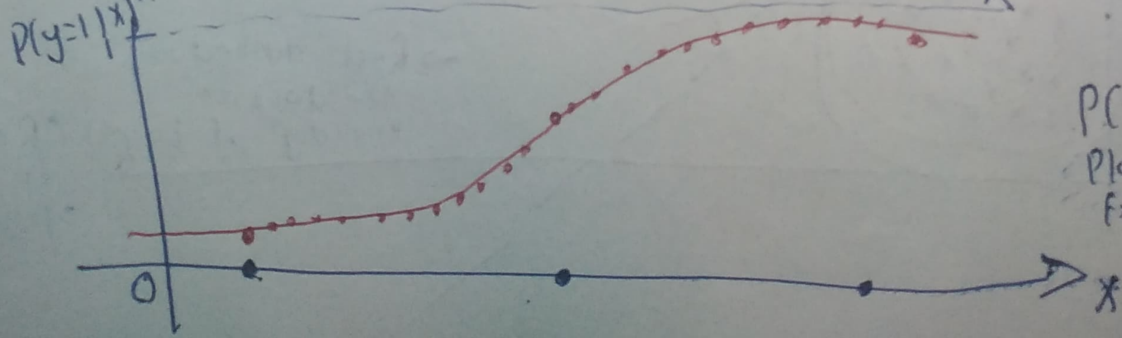
$$= \frac{P(x \mid y=1; \mu, \Sigma)\ \boxed{P(y=1; \phi)}^{\to \phi}}{P(x; \phi, \mu_0, \mu_1, \Sigma)}$$

* For every value of x you can compute the above ratio (with fixed parameters).





P(y=1) = 0.5



P(y=1|x) Plotting for different x.

⑧ If you $P(Y=1|X)$ with different X then a $\boxed{\text{Sigmoid function}}$ appears

⑨ Mechanics of GDA and LR are different but they both end up choosing sigmoid function for $P(Y|X)$

⑨ ~~When is f~~

⑩ GDA assumes:- (Generative model)

$$x|y=0 \sim \mathcal{N}(\mu_0, \Sigma)$$
$$x|y=1 \sim \mathcal{N}(\mu_1, \Sigma)$$
$$y \sim Ber(\phi)$$

Logistic regression (Discriminative model)

$$P(y=1|x) = \frac{1}{1+e^{-\theta^T x}}$$
$$(\text{"}x_0 = 1\text{"}).$$

⊛ As illustrative above that these set of assumes $\boxed{\text{implies} \Rightarrow}$

⊛ However, the opposite is not true so if you know $P(Y|X)$ is governed by sigmoid function you cannot imply $P(X|Y)$ is Gaussian ($\nLeftarrow$)

* This shows that GDA takes $\boxed{\text{stronger assumption}}$ while LR do not since $P(X|Y)$ under (Logistic Regression) LR can be of any distribution.

→ Which works better when?

* $\boxed{P(X|Y) \text{ is gaussian}}$ then GDA would work better because of its assumption or else logistic regression

GDA is more computationally efficent.

* Lets say:-
  - $X|y=1 \sim \text{Possion}(\lambda_1)$
  - $X|y=0 \sim \text{Possion}(\lambda_2)$
  - $y \sim Bi(\phi)$

→ $P(Y=1|X)$ is still logistic regression.

↓

GDA would perform worst

Logistic Regression would still work fine

* Key takeway:-

→ Weak assumption make our model more robust

→ Strong assumption would make our model restricted.

6

* If $P(X|Y)$ is from exponential family with their natural parameter then $P(Y/X)$ would be logistic (sigmoid function).

Naive Bayes

- Email classification problem:-
  - How do you represent a feature vector X?

$$X \in \{0,1\}^n$$

1 → word appears

$$X = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{bmatrix} \begin{matrix} a \\ aardvark \\ aardwolf \\ \vdots \\ buy \\ cs229 \\ zymurgy \end{matrix}$$

↓ An email

10000

* X_j

$X_i = \mathbb{1}\{\text{word } i \text{ appear in email}\}$.

→ In naive bayes want to model $p(x|y)$, $p(y)$

→ 2^{10000} possible outcome of X

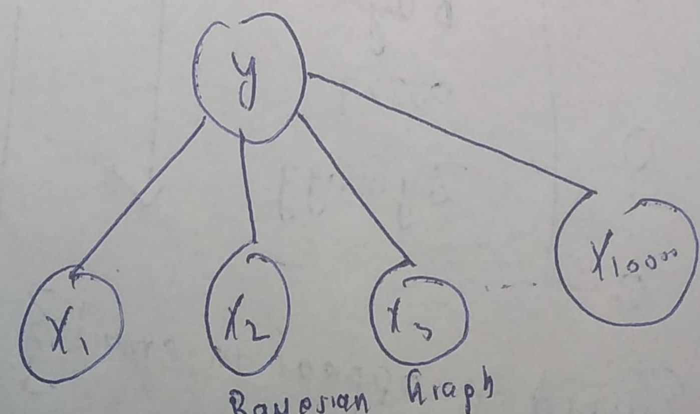
(A) ↳ if we model of $P(X)$ in straightfoward way as a multinomial distribution over 2^{10000} possible out come then you need $2^{10000}-1$ parameters.

(*) In $\boxed{\text{Naive Bayes}}$ we assume X's are conditionally independent given y.

No Assumption ← $P(X_1, \ldots X_{10000}|y) = P(X_1|y)\, P(X_2|y, X_1)\, P(X_3|X_1, X_2) \cdots P(X_{10000}|\ldots)$

Naive Bayes Assumption. (Conditional Independent Assumption). ← $P(X_1, \ldots X_{10000}|y) = P(X_1|y) \cdot P(X_2|y) \cdots P(X_{10000}|y)$

(·)



Bayesian Graph Representation.

(*) $P(x_1, \ldots x_{10000}) = \prod_{i=1}^{n} P(x_n | y)$

(*) Parameter of this model:-

~~$\phi_{j|y=1} = P(x_{j=1} | y=1)$~~

$\phi_{j|y=1} = P(x_{j=1} | y=1)$

$\phi_{j|y=0}\ P(x_{j=1} | y=0)$

$\phi_y = P(y=1)$

→ Joint Likelihood

$\mathcal{L}(\phi_y, \phi_{j|y}) = \prod_{i=1}^{m} P(x^{(i)}, y^{(i)}; \phi_y, \phi_{j|y})$

MLE: $(\nabla_{\phi_y, \phi_{j|y}} \mathcal{L}(\phi_y, \phi_{j|y}) = 0 \rightarrow$ you find outs

$$\phi_y = \frac{\sum_{i=1}^{n} \mathbb{1}\{y^{(i)} = 1\}}{m}$$

$\cancel{\phi}$

(5)

$$\phi_{j|y=1} = \frac{\sum_{i=1}^{m} \mathbb{1}\{x_j^{(i)} = 1, y^{(i)} = 1\}}{\sum_{i=1}^{m} \mathbb{1}\{y^{(i)} = 1\}}$$

→ what fraction of any spam email had word j

→ All spam email

→ Testing time:- (Naive Bayes):-

$$\arg\max_y P(y|x) = \arg\max_y \prod_{j=1}^{p} P(x_j|y) \cdot P(y)$$