Naive Bayes & Logistic Regression

Let's say we want to classify a person into Male/Female based on hair length

Probabilistic Classifier →

$$P(C = C_k \mid x)$$

$$\Downarrow$$

$$P(C = \text{Male/Female given hair length})$$

C/y = Class Label , x = feature vector

We need to find this probability.

If $P(C=M \mid x) > P(C=F \mid x)$; then we will output Male

Bayes Theorem →

$$\overbrace{P(y_i \mid x_i)}^{\text{posterior}} = \frac{\overbrace{P(x_i \mid y_i)}^{\text{liklihood}} \cdot \overbrace{P(y_i)}^{\text{prior knowledge}}}{\underbrace{P(x_i)}_{\text{marginalization}}}$$

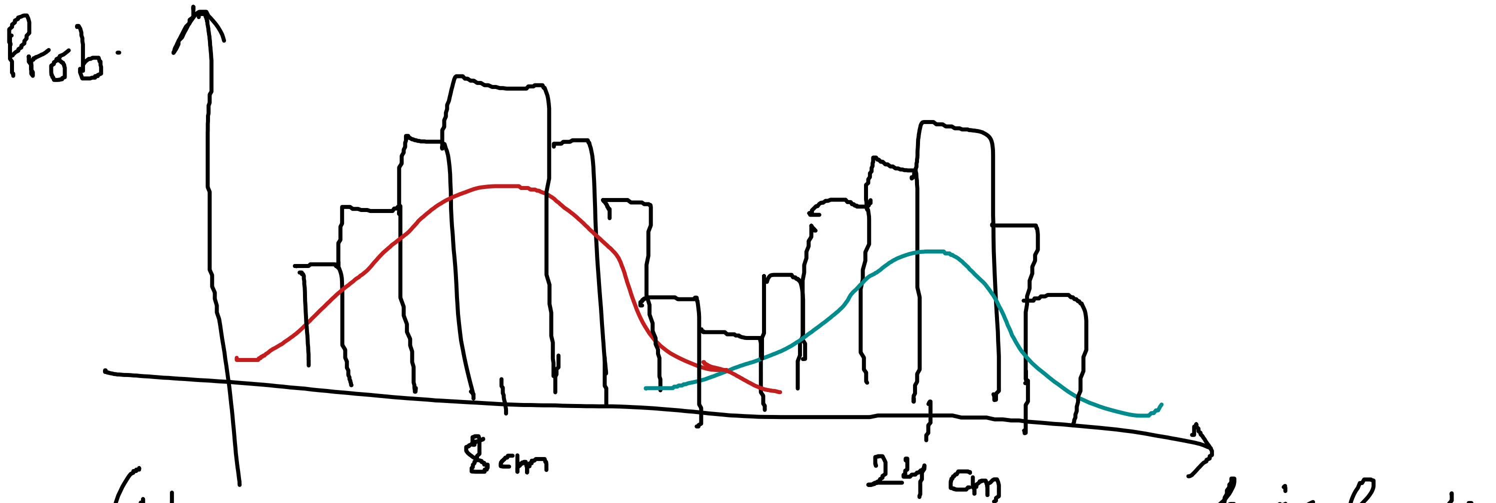
Same for both classes
so we can ignore it

We know this

$$\frac{\text{Male}}{\text{Total}}, \frac{\text{Female}}{\text{Total}}$$

Fitting a gaussian →

$$p(x_i \mid \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \; ; \; \{x_i\}_{i=1:N}$$



(Normalized Histogram = PDF)

To find the best gaussian we need to define a cost function.

Can you think of one?

MLE

MLE → We are maximising the prob. of N samples, while maximising the likelihood of a curve.

$(\theta = \{\mu, \sigma\})$

$$p(X|\theta) = p(x_1 \cdot x_2 \cdot x_3 \cdots x_N | \theta)$$

* iid assumptions → (independently & identically dist.)

$$p(X|\theta) = \prod_{i=1}^{N} p(x_i|\theta)$$

$$L(\theta) = \ln\left(p(X|\theta)\right) = \sum_{i=1}^{N} \ln\left(p(x_i|\theta)\right)$$ (log is monotonic)

$$\underbrace{\hat{\theta} = \operatorname{argmax}(L(\theta))}_{\text{find optimal } \theta}$$

$$\nabla_\theta L = \sum_{i=1}^{N} \ln\left(p(x_i|\theta)\right) = 0$$

Closed form solution →

$$p(x_i \mid \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

$$\underbrace{\ln\left(p(\;)\right)}_{\text{loss function } L(\mu,\theta)} = -N\ln\left(\sigma\sqrt{2\pi}\right) - \sum_{i=1}^{N} \frac{(x_i-\mu)^2}{2\sigma^2}$$

① $$\frac{\partial L}{\partial \mu} = 0 \;; \quad \sum_{i=1}^{N} \frac{(x_i-\mu)}{\sigma^2} = 0$$

$$\sum x_i - N\cdot\mu = 0$$

$$\boxed{\mu = \frac{\sum x_i}{N}} \quad \text{(mean)}$$

② $$\frac{\partial L}{\partial \sigma} = 0 \;; \quad \frac{-N\cdot\sqrt{2\pi}}{\sigma\sqrt{2\pi}} + \frac{\sum(x_i-\mu)^2}{\sigma^3}$$

$$\boxed{\sigma = \sqrt{\frac{\sum(x_i-\mu)^2}{N}}} \quad \text{(std dev)}$$

Naive Bayes →

We can extend the above model to multiple features.

Given: 2D feature vector (hair length, Voice pitch)

So, we will fit 4 gaussians

Male hair, Female hair, Male Voice, Female Voice

Note that we still have 2 classes only, but our x is now a matrix instead of vector.

$$P(C=C_k \mid x) = \frac{p(x_1, x_2 \mid C=C_k) \cdot P(C=C_k)}{p(x_1, x_2)}$$

$$= \frac{p(x_1 \mid C=C_k) \cdot p(x_2 \mid C=C_k) \cdot P(C=C_k)}{p(x_1, x_2)}$$

Naive Assumption → All features are independent

(Without Naive, we will have to fit multivariate gaussian)

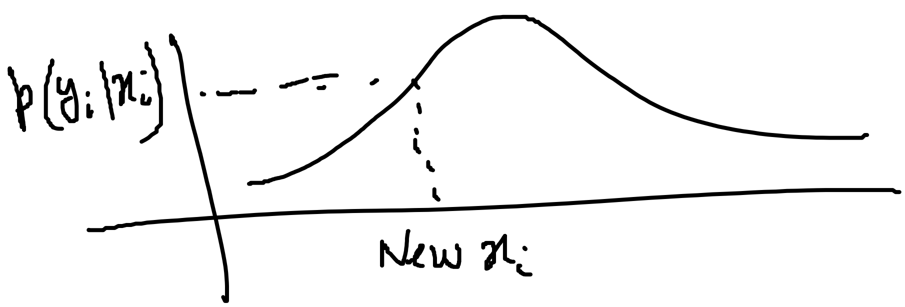
So, for M features →

$$X = N\left\{ \underbrace{\begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}}_{M} \right._{N\times M}$$

We can fit $2 \cdot M$ gaussians assuming all M features are mutually independent

Naive Bayes belongs to a class of models known as $\boxed{\text{Generative models.}}$

Once we know the parameters of optimal gaussian we can generate synthetic data points



We also have $\boxed{\text{discriminative models}}$ which cannot be used for generating data, because they do not store information about distribution of data.

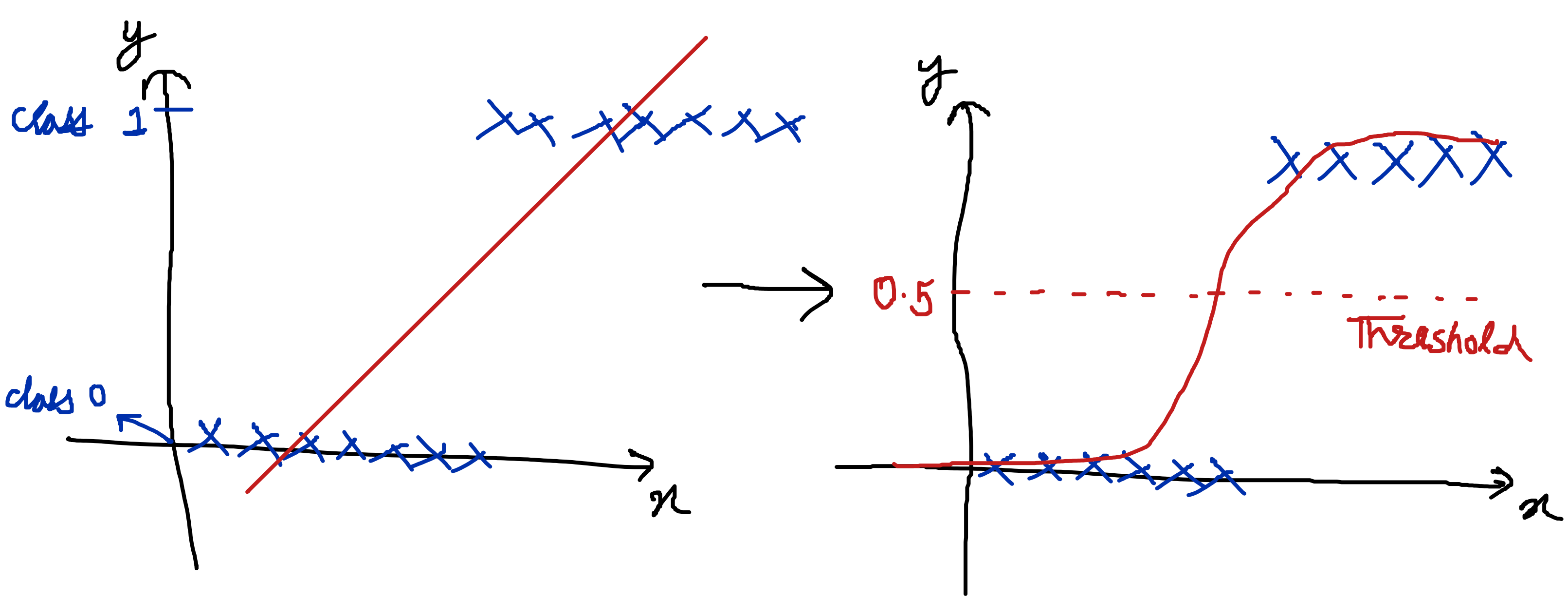
What if our data is not gaussian?

Logistic Regression →

Given $\vec{x}$, LR does the following →

① fit a linear classifier ($w^T x$) similar to Linear Regression

② Apply Sigmoid function to output of ①



$z = w^T x$
$(w_1 x_1 + w_0)$

$$y = \sigma(z) = \frac{1}{1 + e^{-w^T x}}$$

Sigmoid converts $(-\infty, \infty) \to (0,1)$ probabilities.

MLE (Bernaulli) →

Unlike Naive Bayes, here we will directly estimate the posterior probability.

$\{M, F, M, M, M, F, F \_ \_ \_\}$: N observations

$$p = P(\text{Female} / 1)$$
$$1 - p = P(\text{Male} / 0)$$

$$P(Y; p) = \prod_{i=1}^{N} P(y_i ; p)$$ (assuming independent bernaulli trials)

$$L(p) = \ln(P(Y;p)) = \sum \ln(P(y_i ; p))$$

From Bernaulli = $P(y_i ; p) = p^{y_i} \cdot (1-p)^{(1-y_i)}$

$$L(p) = \sum_{i=1}^{N} y_i \ln(p) + (1-y_i) \ln(1-p)$$

For our case, $p = \sigma(x) = \dfrac{1}{1+e^{-w^T x}}$

Some Basic Results →

$$P(y=1 \mid x; w) = \frac{1}{1+e^{-w^T x}} = \frac{e^{w^T x}}{1+e^{w^T x}}$$

$$P(y=0 \mid x; w) = \frac{e^{-w^T x}}{1+e^{-w^T x}} = \frac{1}{1+e^{w^T x}}$$

Back to our loss function →

$$-l(W) \Rightarrow -\sum \left(y_i \ln\left(P(y=1 \mid x_i)\right) + (1-y_i) \ln\left(1-P(y_i=1 \mid x)\right)\right)$$

$$\Rightarrow -\sum \left(y_i \ln\left(\frac{1}{1+e^{-w^T x}}\right) + (1-y_i) \ln\left(\frac{1}{1+e^{w^T x}}\right)\right)$$

$$\Rightarrow \sum \left(y_i \ln\left(1+e^{-w^T x}\right) + (1-y_i) \ln\left(1+e^{w^T x}\right)\right)$$

$$\frac{\partial l}{\partial w} = \sum \left(y_i \cdot \frac{-x \cdot e^{-w^T x}}{1+e^{-w^T x}} + (1-y_i) \frac{x \cdot e^{w^T x}}{1+e^{w^T x}} \right)$$

$$= \sum \left(y_i \cdot \frac{-x \cdot e^{-w^T x}}{1+e^{-w^T x}} + (1-y_i) \frac{x}{1+e^{-w^T x}} \right)$$

$$= \sum \left(\frac{x_i - y_i x_i - x_i y_i e^{-W^T x_i}}{1 + e^{-w^T x}} \right)$$

$$= \sum \left(\frac{x_i - y_i x_i \left(1 + e^{-w^T x_i}\right)}{1 + e^{-w^T x_i}} \right)$$

$$= \sum \left(x_i \left(\sigma(x_i) - y_i \right) \right)$$

In matrix form $\Rightarrow$ $\boxed{\nabla_w L = X^T \left(\sigma(X) - \vec{y} \right)}$

Apply gradient descent to get optimal w.

$$X = \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & x_3 & 1 \end{array} \\ \begin{bmatrix} - & - & - & - \\ - & - & - & - \\ - & - & - & - \end{bmatrix} \end{array}_{N \times 4} ; \quad \vec{y} = \begin{bmatrix} - \\ - \\ - \\ - \\ - \end{bmatrix}_{N \times 1} ; \quad \vec{w} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_0 \end{bmatrix}_{4 \times 1}$$

$w \longrightarrow$ random

$$\underbrace{w}_{4 \times 1} \longrightarrow \underbrace{w}_{4 \times 1} - \alpha \cdot \underbrace{\overset{4 \times N}{X^T} \big(\overset{N \times 1}{\sigma(X)} - \overset{N \times 1}{y} \big)}_{4 \times 1}$$

$\sigma(w^T X)$

$\approx X w$

$N \times 4 \cdot 4 \times 1$

$= N \times 1$

Note → L.R. can also be derived from Cross Entropy Loss,

$$\boxed{L_{CE} = -\sum y_i \log(p_i)}$$

This is equivalent to Log loss or Logistic loss we derived earlier.

What if there are more than 2 classes?

$$\boxed{\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}}$$

Converts vector of K numbers into PDF of K outcomes