Notes On Machine Learning

Parameter Estimation

** given a parameter value of a distribution how the density/distribution function looks like.

** Sampling: if x follows a distribution, how can we obtain different value of x with ease.

** Parameter estimation:
Given different values of x, how can we obtain the parameters of the underlying function.

** Given data, statistics/ML aim to learn parameters of the underlying distribution from data.

→ Gaussian Mixture model / probabilistic graphical model / linear regression / logistic regression.

** Sampling is essential in probabilistic machine learning & Bayesian statistics.

→ Latent dirichlet allocation / Gaussian process / probabilistic graphical model.

→ Marcov chain model (MCMC - Marchov chain Monte carlo).

** Any statistic used to estimate the value of an unknown parameter θ is called an estimator of θ.

Ex - μ & σ for Normal distribution
λ for precision.

The strategy is to choose the parameter in such a way that maximiz the likelihood between the prediction & true label or minimize the loss between the prediction & tone label.

once you define loss function only need to optimize the loss function with the parameter.

two ways to defining loss function

① LSE (Least square Error)

This is special form of MLE. we may define the loss function as:

$$\hat{w}_{LSE} = \underset{w}{\operatorname{argmin}} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$

$$= \underset{w}{\operatorname{argmin}} \sum_{i=1}^{N} (x_i w_i - y_i)^2$$

$$= \underset{w}{\operatorname{argmin}} (Xw - y)^2$$

This is closed form of above

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{bmatrix}_{p\times 1}$$

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & & & \\ x_n & \cdots & \cdots & x_{np} \end{bmatrix}_{n\times p}$$

Now we need to choose w that minimize

$$(Xw - y)^2$$

So, Loss function

$$L(w) = \frac{1}{2} \sum_{i=1}^{N} (x_i w_i - y_i)^2$$

$$= \frac{1}{2} (xw - y)^2$$

$$\frac{\partial L(w)}{\partial w} \Rightarrow (xw - y) x^T = 0$$

$$\Rightarrow x^T x w - x^T y = 0$$

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

Thus, $\vec{w} = (x^T x)^{-1} x^T y$ gives least square estimation for our parameter w.

(2) Maximum Likelihood Estimation (MLE)

MLE to find the parameters that maximize the likelihood between y & $\hat{y}$.

We can define the loss function of MLE as

$$\hat{\theta}_{MLE} = \text{argmax} \boxed{P(D|\theta)}$$

- $P(D|\theta)$: this represents probability on the dataset D, calculated by the parameter θ.
- Parameter to estimate which is similar to w
- D is the dataset

This is similar to $P(\vec{x}|y) = P(x.w|y)$

Ex: MLE for Bernoulli parameter!

proof: $P(x) = \begin{cases} 1-p & \text{for } x=0 \\ p & \text{for } x=1 \end{cases}$

It is also written as $p^x(1-p)^{x-1}$ as a closed form.

$$f(x_1, x_2 \cdots x_n | p) = P(x=x_1, x=x_2, x=x_3 \cdots x=x_n | p)$$

$$\Rightarrow P(x=x_1|p)\, P(x=x_2|p) \cdots P(x=x_n|p)$$

$$\Rightarrow p^{x_1}(1-p)^{x_1-1} \cdot p^{x_2}(1-p)^{x_2-1} \cdots p^{x_n}(1-p)^{x_n-1}$$

$$\Rightarrow p^{\sum_{i=1}^{n} x_i} (1-p)^{\sum_{i=1}^{n} x_i - n}$$

taking log both side to find log likelihood.

$$\log f(x_1, x_2 \cdots x_n | p) = \sum_{i=1}^{n} x_i \log p + \left(\sum_{i=1}^{n} x_i - n\right) \log(1-p)$$

$$\frac{\partial}{\partial p}\left(\log f(x_1, x_2 \cdots x_n | p)\right) = \frac{1}{p}\sum_{i=1}^{n} x_i + \frac{1}{1-p}\left(\sum_{i=1}^{n} x_i - n\right) = 0$$

$$\Rightarrow \frac{1}{p}\sum_{i=1}^{n} x_i + \frac{1}{1-p}\sum_{i=1}^{n} x_i - \frac{n}{1-p} = 0$$

$$\Rightarrow \sum_{i=1}^{n} x_i \left(\frac{1}{p} + \frac{1}{1-p}\right) = \frac{n}{1-p}$$

$$\Rightarrow \sum_{i=1}^{n} x_i \left(\frac{1-p+p}{p(1-p)}\right) = \frac{n}{1-p}$$

$$\Rightarrow \boxed{p = \frac{1}{n}\sum_{i=1}^{n} x_i}$$

MLE for normal Distribution:

$$P(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

take log likelihood,

$$\log P(x|\mu,\sigma^2) = \log\left[\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}\right]$$

$$L = \sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}\right)$$

$$L = \sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi}\sigma}\right) + \sum_{i=1}^{n} \log e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

$$L = \sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi}\sigma}\right) + \sum_{i=1}^{n} -\frac{(x_i-\mu)^2}{2\sigma^2}$$

$$\frac{\partial L}{\partial \mu} = \sum_{i=1}^{n} -\frac{1}{2\sigma^2} 2(x_i-\mu)(-1)$$

$$= \sum_{i=1}^{n} \frac{x_i-\mu}{\sigma^2} = 0 \Rightarrow \sum_{i=1}^{n} x_i - n\mu = 0$$

$$\Rightarrow \boxed{\mu = \frac{1}{n}\sum_{i=1}^{n} x_i}$$

$$\frac{\partial L}{\partial \sigma} = \frac{\partial}{\partial \sigma}\left(n \log\left(2\pi\sigma^2\right)^{-1/2}\right) + -\frac{1}{2}\sum_{i=1}^{n} (x_i-\mu)^2 \frac{2}{\partial\sigma}(\sigma^{-2})$$

$$= -\frac{n}{2}\frac{1}{2\pi\sigma^2} * 2\pi\sigma + -\frac{1}{2}\sum_{i=1}^{n} (x_i-\mu)^2 - 2\sigma^{-3}$$

$$\Rightarrow -\frac{n}{\sigma} + \frac{1}{\sigma^3}\sum_{i=1}^{n} (x_i-\mu)^2 = 0$$

$$\Rightarrow \frac{1}{\sigma^2} \sum_{i=1}^{n} (x_i - \mu)^2 = n$$

$$\Rightarrow \boxed{\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2}}$$

(3) Maximum A-posteriori Estimation (MAP)

$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}}\, P(\theta \mid D) = \underset{\theta}{\operatorname{argmax}}\, \frac{P(D \mid \theta)\, P(\theta)}{P(D)}$$

$$\propto \underset{\theta}{\operatorname{argmax}}\, P(D \mid \theta)\, P(\theta)$$

Since $P(D)$ in the above expression doesn't depend on θ, we can just maximize the numerator $P(D \mid \theta)\, P(\theta)$.

$$\hat{\theta}_{MAP} = \underset{\theta}{\operatorname{argmax}}\, P(D \mid \theta)\, P(\theta)$$

$$= \underset{\theta}{\operatorname{argmax}} \left\{ \log\left(P(D \mid \theta)\right) + \log P(\theta) \right\}$$

$$= \underset{\theta}{\operatorname{argmax}} \left\{ \sum_{i=1}^{n} \log P(D_i \mid \theta) + \log P(\theta) \right\}$$

This is same as MLE with extra log-prior-distribution term.

MAP allows incorporating our prior knowledge about θ in its distribution.

** Prior belief of p.

→ This prior belief of p will be 0 outside [0,1]

→ In most cases we want to choose a distribution for prior beliefs that peaks somewhere in the interval [0,1].

□ Beta distribution:

Parameterized by two shape constants α & β.

$$\text{Prob}(p|\alpha,\beta) = \frac{1}{B(\alpha,\beta)} p^{\alpha-1}(1-p)^{\beta-1}$$

$$B = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$$

where $\Gamma n = (n-1)!$

1.1 Definition

Beta distribution is a type of statistical distribution, which has two free parameters. It is used as a prior distribution in Bayesian inference, due to the fact that it is the conjugate prior distribution for the binomial distribution, which means that the posterior distribution and the prior distribution are in the same family.

The probability distribution function (pdf) of the beta distribution is defined as,

$$\begin{aligned} f(x;\alpha,\beta) &= \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)+\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1} \\ &= \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)} \end{aligned} \quad (1)$$

where $\alpha > 0, \beta > 0, x \in [0, 1]$ and $\Gamma(\cdot)$ denotes the gamma function.

1.2 Application

Considering the classical Bernoulli problem (repeated coin flipping), after n trials, there are s successes (heads) and f failures (tails). Let a random variable x denote the success probability of each trial. The likelihood for parameters s and f given $x = p$ is is the following binomial distribution,

$$L(s,f|x=p) = \binom{n}{s} x^s(1-x)^{n-s} \quad (2)$$

If belief about prior probability information is reasonably well approximated by a beta distribution,

$$P(x=p;\alpha,\beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)} \quad (3)$$

According to Bayes's theorem, the posterior probability is given by the product of the likelihood function and the prior probability normalised by the integral as follows,

$$\begin{aligned} P(x=p|s,f) &= \frac{L(s,f|x=p)P(x=p;\alpha,\beta)}{\int_0^1 L(s,f|x=p)P(x=p;\alpha,\beta)dx} \\ &= \frac{\binom{n}{s} x^s(1-x)^{n-s}\frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)}}{\int_0^1 \binom{n}{s} x^s(1-x)^{n-s}\frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)}dx} \\ &= \frac{x^{s+\alpha-1}(1-x)^{n-s+\beta-1}}{B(s+\alpha,n-s+\beta)} \end{aligned} \quad (4)$$

The posterior probability function is also a beta distribution (conjugate). It is convenient to compute. This is the main reason why we approximate the prior using a beta distribution.

** Derive a-posteriori estimation for the parameter θ.

$$\hat{P}_{MAP} = \left(\prod_{i=1}^{n} P_{Bernoulli}(x_i|\theta)\right) \cdot P_{Beta}(\theta|\alpha,\beta)$$

$$L = \sum_{i=1}^{n} \log P_{Bernoulli}(x_i|\theta) + \log P_{Beta}(\theta|\alpha,\beta)$$

$$= \sum_{i=1}^{n} \log\left[\theta^{x_i} * (1-\theta)^{x_i-1}\right] + \log \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{B(\alpha,\beta)}$$

$$L = \sum_{i=1}^{n} \log\theta^{x_i} + \sum_{i=1}^{n} \log(1-\theta)^{x_i-1} + \log\theta^{\alpha-1} + \log(1-\theta)^{\beta-1} - \log B(\alpha,\beta)$$

$$L = \sum_{i=1}^{n} x_i \log\theta + \sum_{i=1}^{n} (x_i-1)\log(1-\theta) + (\alpha-1)\log\theta + (\beta-1)\log(1-\theta) - \log B(\alpha,\beta)$$

$$\frac{\partial L}{\partial \theta} = \sum_{i=1}^{n} \frac{x_i}{\theta} - \sum_{i=1}^{n} \frac{(x_i-1)}{(1-\theta)} + \frac{\alpha-1}{\theta} - \frac{\beta-1}{1-\theta} = 0$$

$$\Rightarrow \frac{1}{\theta}\sum_{i=1}^{n} x_i + \frac{1}{1-\theta}\sum_{i=1}^{n} x_i - \frac{n}{1-\theta} + \frac{\alpha-1}{\theta} - \frac{\beta-1}{1-\theta} = 0$$

$$\Rightarrow \frac{1}{\theta}\left[\sum_{i=1}^{n} x_i + \alpha - 1\right] + \frac{1}{1-\theta}\left[\sum_{i=1}^{n} x_i - n - \beta + 1\right] = 0$$

$$\Rightarrow (1-\theta)\left[\sum_{i=1}^{n} x_i + \alpha - 1\right] + \theta\sum_{i=1}^{n} x_i - n\theta - \beta\theta + \theta = 0$$

$$\Rightarrow \sum_{i=1}^{n} x_i + \alpha - 1 - \theta\sum_{i=1}^{n} x_i - \alpha\theta + \theta + \theta\sum_{i=1}^{n} x_i - n\theta - \beta\theta + \theta = 0$$

$$\Rightarrow \sum_{i=1}^{n} x_i + \alpha - 1 + \theta[-\alpha - \beta - n + 2] = 0$$

$$\Rightarrow \theta[n + \alpha + \beta - 2] = \sum_{i=1}^{n} x_i + \alpha - 1$$

$$\Rightarrow \theta_{MAP} = \frac{\sum_{i=1}^{n} x_i + \alpha - 1}{n + \alpha + \beta - 2}$$

** Apply MAP on multinomial distribution

The multinomial distribution is a common distribution for characterizing categorical variables. Suppose a random variable Z has k categories, we can code each category as an integer, leading to $Z \in \{1, 2, \cdots, k\}$. Suppose that $P(Z = k) = p_k$. The parameter $\{p_1, \cdots, p_k\}$ describes the entire distribution of k (with the constraint that $\sum_j p_j = 1$). Suppose we generate $Z_1, \cdots, Z_n$ IID from the above distributions and let

$$X_j = \sum_{i=1}^{n} I(Z_i = j) = \#\text{ of observations in the category } j.$$

Then the random vector $X = (X_1, \cdots, X_k)$ is said to be from a multinomial distribution with parameter $(n, p_1, \cdots, p_k)$. We often write

$$X \sim M_k(n; p_1, \cdots, p_k)$$

to denote a multinomial distribution.

Example (pet lovers). The following is a hypothetical dataset about how many students prefer a particular animal as a pet. Each row (except the 'total') can be viewed as a random vector from a multinomial distribution. For instance, the first row $(18, 20, 6, 4, 2)$ can be viewed as a random draw from a multinomial distribution $M_5(n = 50; p_1, \cdots, p_5)$. The second and the third row can be viewed as other random draws from the same distribution.

	cat	dog	rabbit	hamster	fish	total
Class 1	18	20	6	4	2	50
Class 2	15	15	10	5	5	50
Class 3	17	18	8	4	3	50

7.1 Properties of multinomial distribution

The PMF of a multinomial distribution has a simple closed-form. If $X \sim M_k(n; p_1, \cdots, p_k)$, then

$$p(X = x) = p(X_1 = x_1, \cdots, X_k = x_k) = \frac{n!}{x_1! x_2! \cdots x_k!} p_1^{x_1} \cdots p_k^{x_k}.$$

The *multinomial coefficient* $\frac{n!}{x_1! x_2! \cdots x_k!} = \binom{n}{x_1, \cdots, x_n}$ is the number of possible ways to put n balls into k boxes.

The famous *multinomial expansion* is

$$(a_1 + a_2 + \cdots + a_k)^n = \sum_{x_i \geq 0, \sum_i x_i = n} \frac{n!}{x_1! x_2! \cdots x_k!} a_1^{x_1} a_2^{x_2} \cdots a_k^{x_k}.$$

This implies that $\sum_{x_i \geq 0, \sum_i x_i = n} p(X = x) = 1$.

$L(\theta|D) \propto P_{multinomial}(x_1 x_2 \cdots x_n | \theta_1 \theta_2 \cdots \theta_n) \cdot P_{Dirichlet}(\theta_1, \theta_2 \cdots \theta_n | \alpha_1 \alpha_2 \cdots \alpha_n)$

$= \ln \prod_{i=1}^{N} \frac{\theta_i^{x_i}}{[x_i]} \times \left[\frac{\Gamma\left(\sum_{j=1}^{N} \alpha_j\right)}{\prod_{j=1}^{N} \Gamma(\alpha_j)} \right] \prod_{i=1}^{N} \theta_i^{\alpha_i - 1}$

removing all constant

$\prod_{i=1}^{N} \theta_i^{x_i} \prod_{i=1}^{N} \theta_i^{\alpha_i - 1}$

$$= \prod_{i=1}^{n} \theta_i^{x_i + \alpha_i - 1}$$

$$L = \log\left(\prod_{i=1}^{N} \theta_i^{x_i+\alpha_i-1}\right) = \sum_{i=1}^{N} (x_i + \alpha_i - 1) \log \theta_i$$

we have Constraint $\sum_{i=1}^{n} \theta_i = 1$

To impose this constraint we will apply lagrange multiplier λ to our log-likelihood.

$$L = \sum_{i=1}^{N} (x_i + \alpha_i - 1) \log \theta_i + \lambda \left(\sum_{i=1}^{n} \theta_i - 1\right) = 0$$

taking derivative on single θ_i cancels out other terms

$$\frac{\partial L}{\partial \theta_i} = \frac{x_i + \alpha_i - 1}{\theta_i} - \lambda = 0 \Rightarrow \theta_i = \frac{x_i + \alpha_i - 1}{\lambda}$$

given $\sum_{i=1}^{N} \theta_i = 1$

$$\sum_{i=1}^{N} \theta_i = \frac{1}{\lambda} \sum_{i=1}^{N} x_i + \alpha_i - 1 = 1$$

$$\Rightarrow \lambda = \sum_{i=1}^{N} x_i + \alpha_i - 1$$

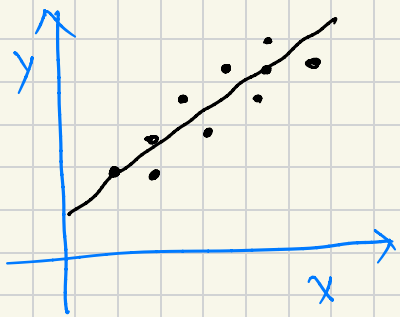
So, replacing λ's value

$$\hat{\theta}_{i\,MAP} = \frac{x_i + \alpha_i - 1}{\sum_{i=1}^{N} x_i + \alpha_i - 1}$$

Linear Regression:

Learn a function which maps input to output

$f: X \to Y$



$$\hat{y}_i = w_0 + w_1 x_i$$
$$= w_0 + w^T x$$

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{bmatrix}_{p \times 1}$$

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & & & \\ x_{n1} & \cdots & \cdots & x_{np} \end{bmatrix}_{n \times p}$$

Loss $L(w) = \frac{1}{2} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$

$$L(w) = \frac{1}{2N} \sum_{i=1}^{N} \{ y_i - (w^T x_i + w_0) \}^2$$

↑ our goal is to minimize this loss function.

We can also write it as

$$\frac{\partial L(w)}{\partial w} = \frac{\partial}{\partial w} \left[\frac{1}{2N} (Xw - y)^2 \right]$$

$$\nabla E(w) \Rightarrow \frac{1}{2N} 2 (Xw - y) X^T = 0$$

$$2) \quad X^T X w = X^T y$$

$$\Rightarrow \quad w = (X^T X)^{-1} X^T y$$

$(X^T X)^{-1}$: $p \times N$, $N \times p$ → $p \times p$

$X^T y$: $p \times N$, $N \times 1$ → $p \times 1$

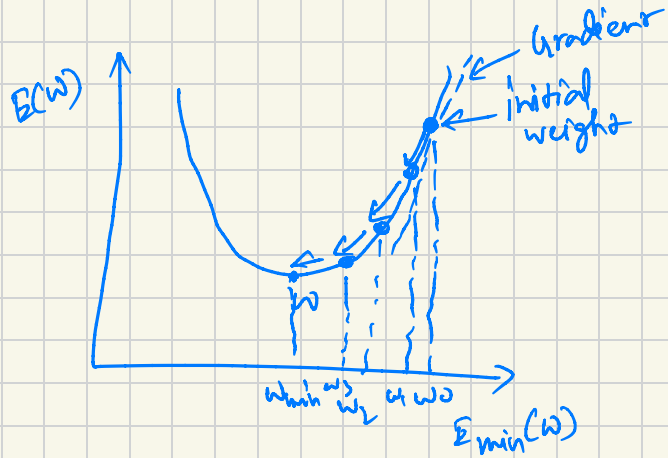
w: $p \times p$, $p \times 1$ → $p \times 1$

** update weight

$$w^{\tau+1} = w^{\tau} - \eta(\nabla E) \rightarrow (xw - y)x^T$$

τ = iteration number

η = learning rate parameter



$w_1 = w_0 - \eta \nabla E$

$w_2 = w_1 - \eta \nabla E$

$w_3 = w_2 - \eta \nabla E$

$w_{min} = w_3 - \eta \nabla E$

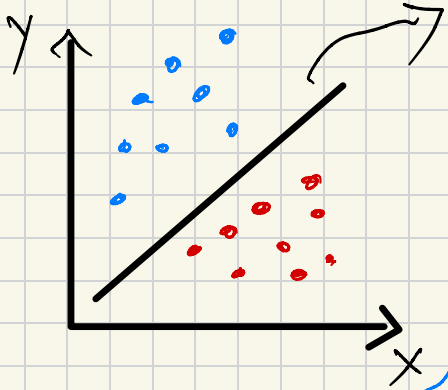
** To apply gradient descent, the function needs to be a convex function.

- This function will have only one global minima.
- It follows a property called Jensen's Inequility.

Logistic Regression:

Classification Problem

Linear Model



this is called discernment function or decision boundary. We need to learn this function.

Why Linear Regression can't be used for classification

→ For classification problem using linear regression, the probability is a Normal distribution. so, output $P(y=0|x) + P(y=1|x) \neq 1$.

→ even if you provide binary output to linear regression, it may provide output in real number like 1, 2, -2 etc which doesn't make sense in case of binary classification.

→ to make proper decision we need probabilistic output which sum upto 1 for all classes.

So, use of linear regression for classification problem is not a good option. We need good model which can distribute our probability value within 0 & 1 and for K-class classification within (1, K).

our task is to,

$$P(Y=1|X) \Rightarrow \mathbb{R}^D \xrightarrow{w^T x} \mathbb{R} \xrightarrow{\text{sigmoid fn}} \{0,1\}$$

linear transformation

also called probit function.

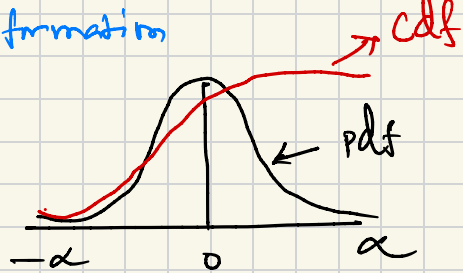
If, $P(x) = N(x|0,1)$

$$CDF(x) = \int_{-\infty}^{\alpha} P(x)\,dx$$

$$CDF(-\infty) = 0$$

$$CDF(0) = \int_{-\infty}^{0} P(x)\,dx = 0.5$$

$$CDF(\infty) = 1.$$



Logistic Sigmoid:

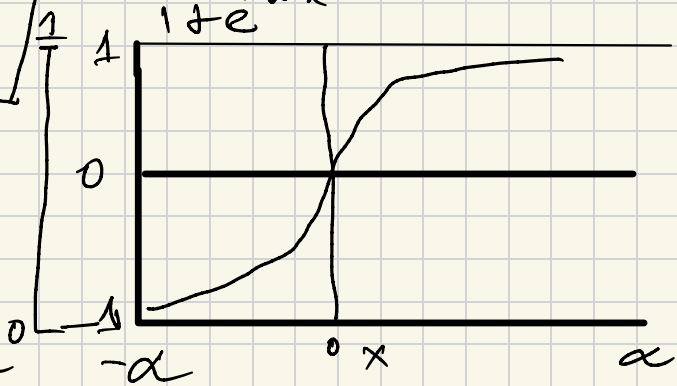
$$P(Y=1|X) = \sigma(w^T x) = \frac{1}{1+e^{-w^T x}}$$

$\sigma(-\infty) = 0$

$\sigma(0) = 0.5$

$\sigma(\infty) = 1$

so, sigmoid fun we can use to convert $w^T x$ to $\{0,1\}$

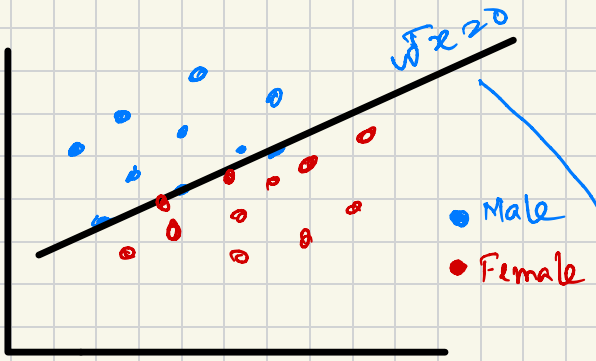


$$P(y=1|x) = \sigma(w^T x) = \frac{1}{1+e^{-w^T x}}$$

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

$$= \frac{e^{-w^T x}}{1+e^{-w^T x}}$$

$$= \frac{1}{1+e^{w^T x}}$$



→ this is decision boundary with equation $w^T x = 0$

$P(y=1|x) = P(y=0|x) = 0.5$

Points on lies on the line $w^T x = 0$ have same probability value = 0.5.

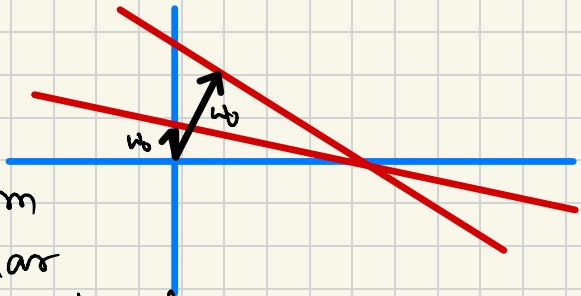
points above $w^T x = 0$ line have high probability towards class 1. or Male class & points lies below $w^T x = 0$ have high probability towards class 0 or Female class.

This probability distribution over the output is good for decision making. But sometime we don't need probability to find confidence towards a positive class. In such scenario we can go with approaches which just provide decision boundary. Ex {LDA, FLD, SVM.}

→ All they learn the decision boundary which is called Discriminant function.

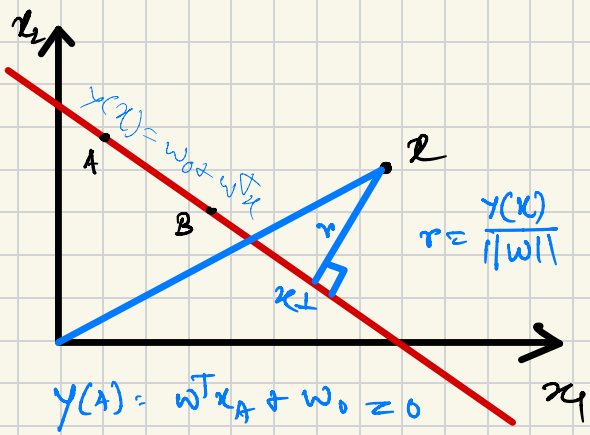
Discriminant Function:

$y(x) = w_0 + w^T x$



w_0 denotes how far your decision boundary from origin. It's a perpendicular direction towards decision boundary from origin.

w is orthogonal to every vector lying within the decision surface, so w determines the orientation of the decision surface.



$r = \frac{y(x)}{\|w\|}$

$y(A) = w^T x_A + w_0 = 0$

$y(B) = w^T x_B + w_0 = 0$

Both are on the line.

So,

$w^T (x_A - x_B) = 0$

Product of two vectors is 0 so they are perpendicular to each other. So, w is orthogonal to every vector lying within the decision surface.

Notes:

Simplistic discriminant function $y = w_0 + w^T x$ represent a hyperplane in $(m-1)$ dimensional space. where m is # of features.

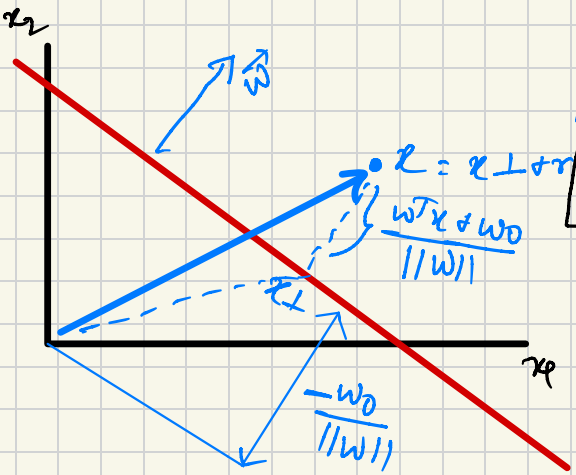
# features	Discriminant function
1	print
2	Line
3	plane
4	Hyperplane in 3D space
...	...
m	Hyperplane in $(m-1)$ D space

We have $w_0 + w^T x = 0$

$$w^T x = -w_0$$

Normalizing both side with the length of the vector $\|w\|$, we get normal distance from the origin to the decision surface.

$$d = \frac{w^T x}{\|w\|} = -\frac{w_0}{\|w\|}$$



So, w_0 shifts the boundary away from origin.

$\frac{w}{\|w\|}$ this is unit vector for direction.

$$y(x) = w^T x + w_0$$

replacing x with $x_\perp + r\frac{w}{\|w\|}$

$$y(x) = w^T x_\perp + r\frac{w^T w}{\|w\|} + w_0$$

both addition is 0 bcz they are on same line/plane.

$$y(x) = r\frac{w^T w}{\|w\|} = r\frac{\|w\|^2}{\|w\|}$$

$$y(x) = r\|w\|$$

So, $$r = \frac{y(x)}{\|w\|}$$

$y(x)$ determines the distance of a point to surface.

So, distance of x from origin =

$$\frac{w^T x + w_0}{\|w\|} + \frac{-w_0}{\|w\|}$$

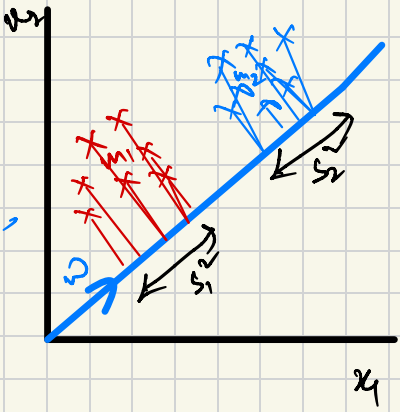
$$\Rightarrow \frac{w^T x}{\|w\|}$$

Supervised Linear Discriminant Analysis (LDA)

→ w as the direction to project x.

→ Find w such that when x is projected, classes are well separated.

→ Simplest measure of the separation of the classes, when projected onto w, is the separation of the projected class means.



$$m_1 = \frac{1}{N_1} \sum_{n \in C_1} x_n \qquad m_2 = \frac{1}{N_2} \sum_{n \in C_2} x_n$$

we need to maximize $m_2 - m_1 = w^T (m_2 - m_1)$

this expression can be made arbitrarily large simply by increasing the magnitude of w. To solve this problem we could constrain w to have unit length, so that $\sum_i w_i^2 = 1$

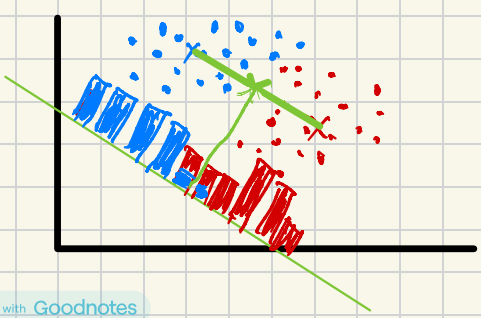
$$L = \underset{w}{\operatorname{argmax}}\; w^T (m_2 - m_1) - \lambda \|w_i\|^2$$

$$\nabla L = (m_2 - m_1) - 2\lambda w = 0$$

$$\Rightarrow \boxed{w = \frac{1}{2\lambda}(m_1 - m_2)}$$

So, $w \propto (m_1 - m_2)$

value of w is still depending on mean of two classes.



Therefore, we can see there are still considerable overlap when projected onto the line joining their means. To eliminate this, Fisher proposes a function which also consider variance of both class points.