Rstudio

7 options.

ladder notation

How to minimize missespecifiation Error?
Make H richer, A as well.

How to minimize stimation error?
Increase n (the sample size)

→ we can not know from where the error is comming from.

02/05

$$Y = g(\bar{x}) + \underbrace{h^*(\bar{x}) - g(\bar{x}) +}_{e \text{ (residual)}}$$

For a new observation x^*, $\hat{y} = g(x^*)$

g comes from In supervise learning → historical data

$$g = \underset{\text{algorithm}}{A}(\mathbb{D}, \mathcal{H}) \to \text{model space}$$

Loan Model

$\mathcal{Y} = \{0, 1\}$ → pay back loan (credit)

0 → no pay back loan

↳ Model is called binary classification model if {0,1}.

Null Model: You have no features and you were to create the best model g.

$\mathcal{H} = \mathcal{Y}$ $g =$ ~~[illegible]~~ Model (y)

We know have one X : credit score $x = [300, 850]$

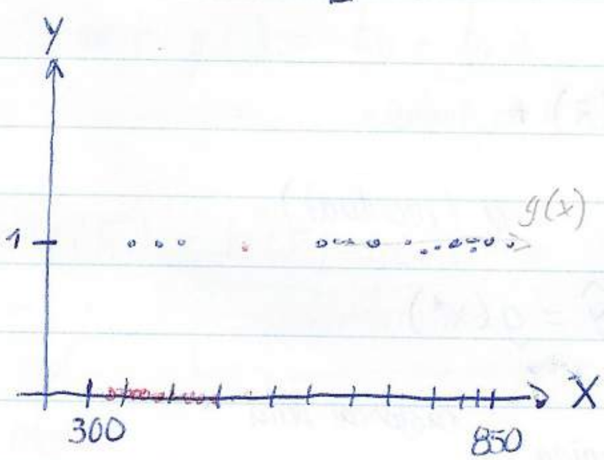
$$\mathbb{D} = (X, y) = \left\langle \begin{bmatrix} 810 \\ 390 \\ 350 \\ \vdots \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \\ \vdots \end{bmatrix} \right\rangle$$

same

Bob

$$\mathbb{D} = [X \mid y] = \begin{bmatrix} 810 & 1 \\ 390 & 0 \\ 250 & 1 \\ \vdots & \vdots \end{bmatrix}$$

row back



Indicator

parameter

$$\mathcal{H} = \{ \mathbb{1}_{x \geq \theta} : \theta \in \Theta \}$$

This is all potential models

parameter space

eg. $h(x) = \mathbb{1}_{x \geq 500}$,

$h(x) = \mathbb{1}_{x \geq 497.3}$

$$e \in \{-1, 0, 1\}$$

$$y = \underbrace{g(x)}_{\hat{y}} + e = e = y - \hat{y}$$

The algorith A produces g, but g is specified by θ

A : finds θ

How about pick θ wich gives the least prediction error in $\mathbb{D}$ (in-sample error estimate).

$$\underbrace{\text{\# errors}}\ = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\ g(x_i) \neq y_i$$

$$ME = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\ g(x_i) \neq y_i$$

all prediction model

↳ $\mathbb{D}$ in-sample

Missclassification Errors

$$\underset{\text{accuracy}}{ACC} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\ g(x_i) = y_i$$

$$A = \theta_g = \underset{\theta \in \mathbb{H}}{\text{argument}}\ \{ME\}$$

↑ objective function, fitness function

[6] is the thing that you max or minimize.

Rewrit objective function using e's (in-sample residuals $\mathbb{D}$)

$$ME = \underbrace{\frac{1}{n}\sum_{i=1}^{n} |e_i|}_{\text{SAE (Sum absolut Error)}} \overset{\text{only in binary Classification}}{=} \underbrace{\frac{1}{n}\sum_{i=1}^{n} e_i^2}_{\text{SSE (Sum Square residuals)}}$$

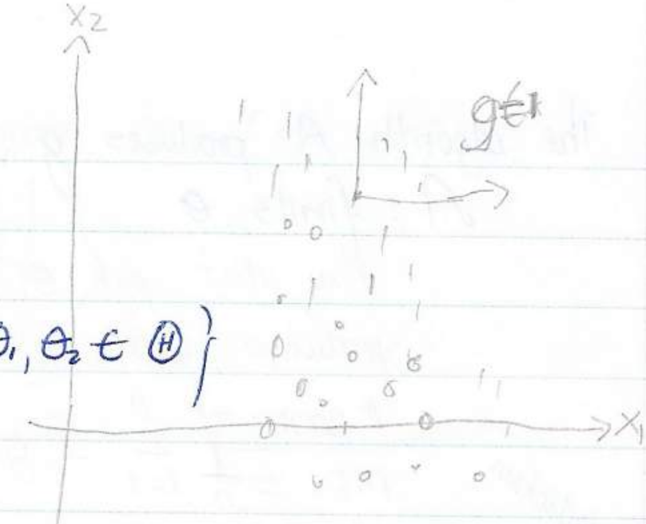
MAE (mean Absolut Error)

" e just can be $(0,1,-1)$

- where $\mathbb{H} \Rightarrow$ unique values of x

X_1 : credit score

X_2 : salary



$H = \{ \mathbb{1}\, x_1 \geq \theta_1 \ \& \ x_2 \geq \theta_2 : \theta_1, \theta_2 \in \Theta \}$

2 dimentiona

$\dim[\Theta] = 2$

very restrictive set.

Threshole

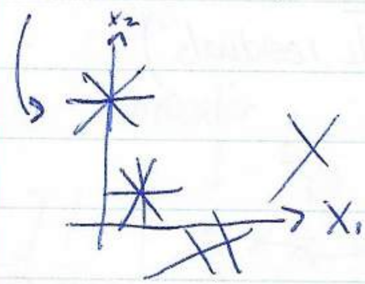
Linear Models

parameters

$\theta_1 \quad \theta_2$

$H = \{ \mathbb{1}\, \underbrace{x_2 \geq a + bx_1}\ : a, b \in \mathbb{R} \} = \{ \mathbb{1}\, \vec{w} \cdot \vec{x} \geq 0, \ \vec{w} \in \mathbb{R}^3 \}$

H is any straigt line



$-a - bx_1 + x_2 \geq 0$

$w_0 + w_1 x_1 + w_2 x_2 \geq 0$

let $\vec{x}_i : [\,1 \ \ x_{i,1} \ \ x_{i,2}\,]$

$$X = \begin{bmatrix} 1 & \uparrow & \uparrow \\ \vdots & \vec{x}_{,1} & \vec{x}_{,2} \\ 1 & \downarrow & \downarrow \end{bmatrix}$$

$\vec{w} \cdot \vec{x} \geq 0$

all $c\vec{w}$ are equivalent solutions, $c \in \mathbb{R}$

divide by paramete and go back to $\mathbb{R}^2$

bega bu ⇒

Perception Learning Algorithm (1957). W, x_i p features...

Step 1: interation $\vec{W}^{t=0} = \vec{0}$ or whatever → lets t denot interation #.

$W_0 + W_1$ → Biasterm / intersept → Input, weight

Step 2: Compute $\hat{y}_i = \mathbb{1}\ \vec{w}^{t=0} \cdot \vec{x}_i$

Step 3: for $j = 0 \ldots p$

$$W_0^{t=1} = W_0^{t=0} + \overbrace{(y_i - \hat{y}_i)}^{e_i} \quad (1)$$

$$W_1^{t=1} = W_1^{t=0} + (y_i - \hat{y}_i) \quad (X_{i,1})$$

$$\vdots$$

$$W_p^{t=1} = W_p^{t=0} + (y_i - \hat{y}_i) \quad (X_{i,p})$$

Step 4: Repeat steps 2,3 for $i = 1 \ldots n$

⇒ to make it smaller or bigger add #s to go out of $\vec{w} \cdot \vec{x} \geq 0$ (↑ Bigger, ↓ Smaller)

Step 5: Repeat steps 2,3,4 until no change or some max itr

This algorithm is prove to converge if the $\mathbb{D}$ is "linearly separable" ie $\exists \vec{w}$ s.t $ME = 0$.

If not it will likely fail

produce a very poor model.

Weakness #1: requires linear separability

Weakness #2: returns any model that separates.

* Support Vector machine.

i	X_1	X_2	y
1	-1	-1	0
2	1	1	1

$\vec{W}^{t=0} = \vec{0}$

$t=1,\ i=1$

$\hat{y}_1^{t=0} = \mathbb{1}\ \vec{w}\cdot\vec{x} \geq 0 = 1$

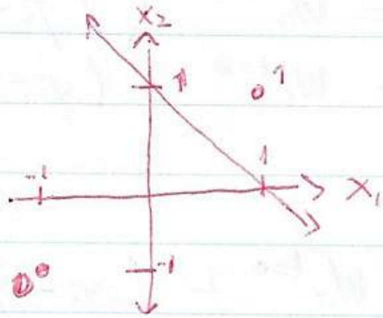
$W_0^{t=1} = (0) + (-1)(1) = -1$

$W_1^{t=1} = (0) + (-1)(-1) = +1$

$W_2^{t=1} = (0) + (1)(1) = +1$

$-1 + X_1 + X_2 \geq 0$

$\Rightarrow X_2 \geq -X_1 + 1$



$t=1 \quad i=2$

$\hat{y}_2 = \mathbb{1} \begin{bmatrix} -1 \\ +1 \\ +1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \geq 0 = 1$

$W_0^{t=1} = (-1) + (0)(1)$

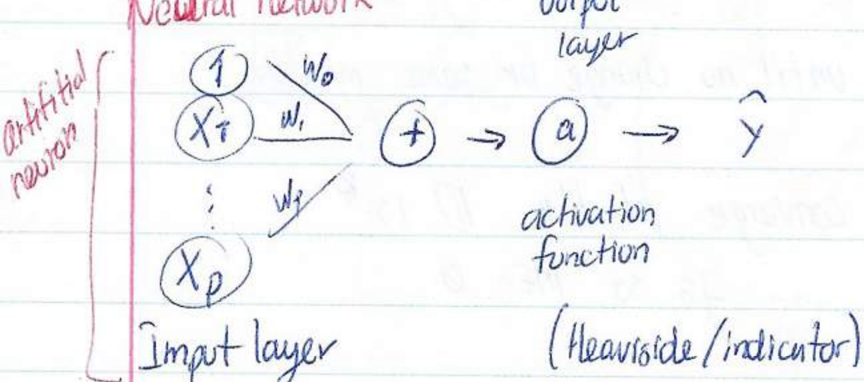
$W_1^{t=1} = (+1) + (0)(1) \Rightarrow \begin{bmatrix} -1 \\ +1 \\ +1 \end{bmatrix}$

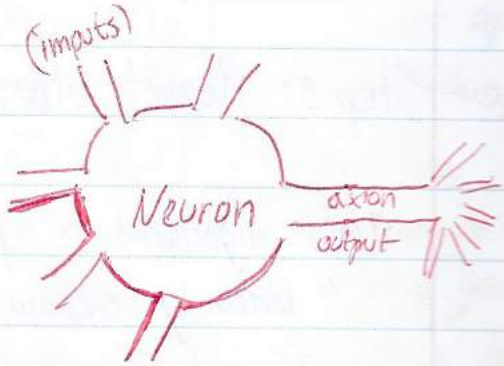
$W_2^{t=1} = (+1) + (0)(1)$

$t=2,\ i=1$

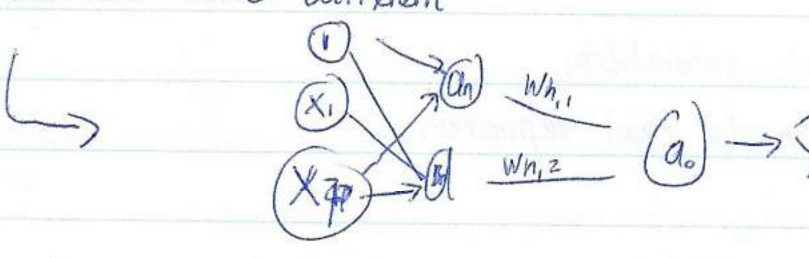
$\hat{y}_1 = \mathbb{1} \begin{bmatrix} -1 \\ +1 \\ +1 \end{bmatrix} \cdot \begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix} \geq 0 = 0$

Neural network





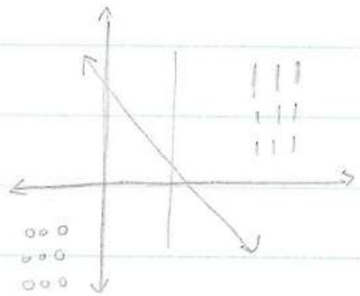
We can have different



Correct the weeknesses is the perseptron algorithm

The first weekness : assumes linear pers

but find the "best" model



R MARKDOWN.

we need } else { → same line

paste → two print outs

0 → TRUE, false   1 → true , 1 → false

switch

Loop

this has to be a vector

sequence.

```
for (i in 1:10) {
    j = i        i by=2
    print(j).
}
```

WE STILL HAVE i & j

still does.