(13)

Logistic Regression (LR)

LR for binary classification.

$$\text{I/P: } \vec{x} = \vec{x} \in \mathbb{R}^d$$
$$\text{O/P: } y \in \{0, 1\}$$

We are interested in $P(Y=1 / X=x)$

Model assumption: $Y / X = x \sim \text{Bernoulli}(\theta_x)$, $\theta_x \in [0, 1]$

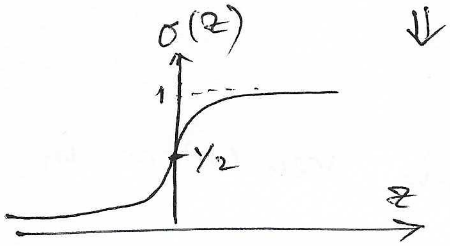
Can we regress from $\vec{x} \to \theta_x$? (ie, modelling directly with a linear function)

i.e. $\hat{\theta} = \vec{w}^T \vec{x}$

$\downarrow$ problem

θ has to be +ve & less than 1. $w^T x$ will produce some value, & we want something bet^n 0 & 1.

$\Downarrow$



$$\sigma(z) = \frac{1}{1+e^{-z}}$$

$Y | X \sim \text{Bernoulli}(\sigma(w^T x))$ ← Logistic regression.

So the posterior probability is,

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

← measure w directly from data.

Some facts.

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

Complementary

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

$$\hat{y}_{MAP} = \underset{y \in \{0,1\}}{\arg\max}\ P(Y=y / X=x)$$

So we're literally choosing bet^n two numbers:

$\left.\begin{array}{l} P(Y=1|X) \\ P(Y=0|X) \end{array}\right\}$ so $\underset{y}{\arg\max}\ P(Y=y|X) = 1$ is same as:

$$P(Y=1|X) \overset{?}{\geq} P(Y=0|X)$$

or, $\dfrac{P(Y=1|X)}{P(Y=0|X)} \overset{?}{\geq} 1$ $\left\{\begin{array}{l}\text{if yes then } \hat{y}=1 \\ \text{else } \hat{y}=0\end{array}\right.$

Also, $\log\left[\frac{P(Y=1|X=x)}{P(Y=0|X=x)}\right] \overset{?}{\geq} 0$

$\Rightarrow \log\left[\frac{(1/1+e^{-w^Tx})}{e^{-w^Tx}/1+e^{-w^Tx}}\right] \geq 0 \Rightarrow \log(e^{w^Tx}) \overset{?}{\geq} 0$

$\Rightarrow \boxed{w^Tx \overset{?}{\geq} 0}.$ ← linear classifier.

So if $w^Tx \geq 0 \Rightarrow \hat{y}=1$

$w^Tx < 0 \Rightarrow \hat{y}=0$

Hence, $w^Tx \equiv$ score of class 1

$\frac{1}{1+e^{-w^Tx}} \equiv$ Probability of class 1.

Estimation of $\vec{w}$

$\boxed{MLE}$ Will be similar to the way we've seen before in coin toss / Bernoulli parameters:

Coin toss	LR
$y \in \{0,1\}$	$y \in \{0,1\}$
Dataset: $D=\{y_1,\dots,y_N\}$	Dataset: $D=\{(x_1,y_1),\dots(x_N,y_N)\}$
$Y \sim Ber(\theta)$	$Y\|X=x \sim Ber(\underbrace{\sigma(w^Tx)}_{\theta_x})$
$P(Y=1)=\theta$	$P(Y=1\|X=x)=\theta_x$
$P(Y=0)=1-\theta$	$P(Y=0\|X=x)=1-\theta_x$
Likelihood for one sample	Likelihood of one sample:
$L(\theta)=\theta^y(1-\theta)^{1-y}$	$L(w)=\theta_x^y(1-\theta_x)^{1-y}$
Likelihood of dataset:	Likelihood of dataset:
$L(\theta)=\prod_{i=1}^N \theta^{y_i}(1-\theta)^{1-y_i}$	$L(w)=\prod_{i=1}^N \theta_{x_i}^{y_i}(1-\theta_{x_i})^{1-y_i}$
$\downarrow$ $\theta^{\alpha_H}(1-\theta)^{\alpha_T}$	$=\prod_{i=1}^N[\sigma(w^Tx_i)]^{y_i}[1-\sigma(w^Tx_i)]^{1-y_i}$

So the log likelihood for LR is,

$$LL(\vec{w}) = \sum_{i=1}^{N} \left[y_i \log \sigma(w^T x_i) + (1-y_i) \log (1 - \sigma(w^T x_i)) \right]$$

$$= \sum_{i=1}^{N} \left[y_i \log\left(\frac{1}{1+e^{-w^T x_i}}\right) + (1-y_i) \log\left(\frac{1}{1+e^{w^T x_i}}\right) \right]$$

$$= \sum_{i=1}^{N} \left[y_i \log\left(\frac{e^{w^T x_i}}{1+e^{w^T x_i}}\right) + (1-y_i) \log\left(\frac{1}{1+e^{w^T x_i}}\right) \right]$$

$$= \sum_{i=1}^{N} \left[y_i \log(e^{w^T x_i}) - y_i \log(1+e^{w^T x_i}) + \log(1) - \log(1+e^{w^T x_i}) - y_i \log(1) + y_i \log(1+e^{w^T x_i}) \right]$$

$$= \sum_{i=1}^{N} \left[y_i \log(e^{w^T x_i}) - \log(1+e^{w^T x_i}) \right]$$

$$= \sum_{i=1}^{N} \left[\underbrace{y_i w^T x_i}_{\text{linear in } w} - \underbrace{\log(1+e^{w^T x_i})}_{\text{not linear in } w} \right]$$

not linear in w → no closed from solution

Vanilla Gradient Descent

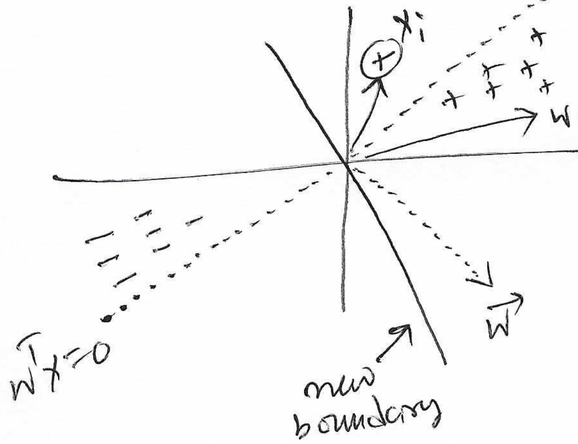
Initialize $w^{(0)}$ & then use the update rule: ← Actually this is gradient ascent

$$w^{t+1} = w^t + \underbrace{\eta}_{\text{step size/ learning rate}} \underbrace{\frac{\partial LL(w)}{\partial w}}_{\text{Gradient (direction of steepest increase)}}$$

$$w^{t+1} = w^t + \eta \sum_{i=1}^{N} \left[y_i x_i^T - \underbrace{\frac{1}{1+e^{w^T x_i}} e^{w^T x_i}}_{p(y_i=1 \mid x_i, w)} x_i^T \right]$$

$$= w^t + \eta \sum_{i=1}^{n} \underbrace{\left[\underbrace{y_i}_{\text{truth } \{0,1\}} - \underbrace{p(y_i = 1 \mid x_i, \vec{w})}_{\text{what our model believes}} \right]}_{\text{error}} \underbrace{x_i^T}_{\text{feature vector/ data}}$$

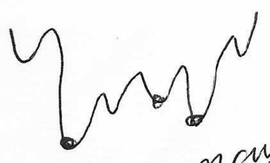
14.5



Geometric interperation of the above equation

Batch gradient descent $\Rightarrow$ moves the entire dataset towards the direction of the gradient

Stochastic " " $\Rightarrow$ perform on sampled points.



GD may or may not work.

Local vs Global minima

GD works always.

$\boxed{\text{MAP estimation of } \vec{w}}$ Model: $Y|X \sim Ber(\sigma(w^T x))$

$w_j \sim \mathcal{N}(0, t^2)$ (i.i.d.)

$$\hat{W}_{MAP} = \underset{W}{\arg\max} \log P(W|D) = \underset{W}{\arg\max} \left(\log P(D|W) + \log P(W)\right)$$

Now as usual, we'll take derivative w.r.t. W & set to 0. For the first term, we have already done it. For the second term:

$$P(\vec{w}) = \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi t^2}} e^{-\frac{w^2}{2t^2}} \quad \text{(i.i.d.)}$$

$$\log P(\vec{w}) = -\frac{w^2}{2t^2} - \frac{d}{2}\log(2\pi t^2)$$

$$\frac{\partial}{\partial w}\left[\log P(\vec{w})\right] = -\frac{2\vec{w}}{2t^2} = -\lambda \vec{w}$$

So the total expression becomes,

$$\frac{\partial}{\partial w}[\cdots] = \underbrace{\sum_{i=1}^{N} [y_i - P(Y=1|x_i)] x_i^T}_{\text{Focus on mistakes}} - \underbrace{\lambda \vec{w}}_{\text{But try to keep norm of } \vec{w} \text{ small}}$$

explain the labels

don't get too confient unless you must.

data wants to increase $||w||$ if it helps in classification

prior resists large weights

$\underbrace{\qquad\qquad\qquad}_{\text{MAP balances the two forces.}}$