- Regression : ended up with optimization problem :

$$\min_{w} \frac{1}{m} \sum_{i=1}^{m} (w^T x^i - y^i)^2 = \left(\text{square loss function}\right).$$

convex + differentiable. there can use GD!

$$E[Y|X] = w^T x \qquad (*)$$

even if $*$ does not hold, can still find min cost (?).

- Classification :

what is the loss function for classification?

our guess for label $y^i \in \{0, 1\}$ is going to be $\text{sign}(w^T x)$ for some vector w.

  - Penalized if $\text{sign}(w^T x^i) \neq y^i$
  - No penalty if $\text{sign}(w^T x^i) = y^i$.

Examine quantity $y^i \cdot (w^T x^i)$:

if negative $\to$ penalty.

if positive $\to$ no penalty.

$\underbrace{\text{0-1 loss}}_{\text{for classification}}$:
$$\ell_{0\text{-}1}(z) = \begin{cases} 1 & \text{if } z \leq 0 \quad (\text{mistake} \to \text{penalty}) \\ 0 & \text{if } z > 0 \quad (\text{no mistake} \to \text{no penalty}). \end{cases}$$

The optimization problem associated with classification :

$$\boxed{\min_{w} \frac{1}{m} \sum_{i=1}^{m} \ell_{0\text{-}1}(y^i \cdot w^T x^i)}$$

Want to solve this .

When does perceptron find w with small loss?

Recall the perceptron required $\exists\, w$ s.t. $\forall\, x$:

$$y \cdot w^T x > \rho \Rightarrow \text{convergence \# of mistake} < \frac{1}{\rho^2}$$

$y \in \{-1, +1\}$

What if there is no margin?

Also, there might not exist a w s.t. $\text{sign}(w^T x^i) = y^i \ \forall\, i$, e.g. they are not linearly separable, e.g. there is not a half space that correctly classifies all the points.

Intuitively, we can still try to $\min_w \left[\frac{1}{m} \sum_{i=1}^{m} L_{0\text{-}1}(y^i w^T x^i)\right]$

$\Rightarrow$ To find a half-space that maximizes the # of correct labels, even though it cannot fully separate correct from incorrect.

What is the computational complexity of this optimization problem?

Unfortunately, $L_{0\text{-}1}$ is neither convex nor differentiable.

This problem is NP hard, e.g. unlikely to have polynomial time solution.

Also it is "agnostically learning a half-space"

Summary :

Regression $\longrightarrow$ convex loss function.

Classification $\longrightarrow$ non convex loss $(0-1)$. (Bad news).

Idea : Relax the 0-1 loss to a different nicer loss :
"surrogate loss", related to 0-1 loss but convex.
would get solutions that are close or high-quality.

Let's introduce a few losses : logistic, hinge, exponential.

- $\underline{\text{Logistic}(z)} = \log\left(1+e^{-z}\right)$

$$\ell_{\text{logistic}}\left(y \cdot W^T x^i\right) = \log\left(1+e^{-(y^i \cdot W^T x^i)}\right)$$

if $\underbrace{y^i W^T x^i}_{\text{margin}} << 0 \Rightarrow \ell_{\text{logistic}}\left(y^i \cdot W^T x^i\right)$ is large.

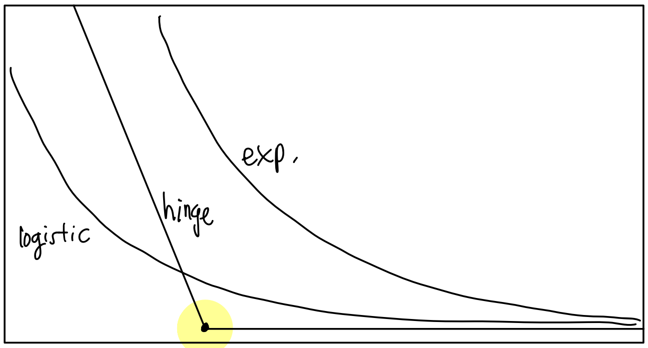
" $>> 0 \Rightarrow \ell_{\text{logistic}}\left(y^i \cdot W^T x^i\right)$ is small. (moves to 0).

- $\underline{\ell_{\text{hinge}}(z)} = \max\{1-z, 0\}$

$\ell_{\text{hinge}}\left(y^i \cdot W^T x^i\right)$

$\Rightarrow$ large when $y^i \cdot W^T x^i$ is negative,
small when $y^i W^T x^i$ is positive.

- $\underline{\ell_{\text{exp}}} = e^{-z}$.



$(z = yW^T x)$ " $x = 1$

Logistic Loss Optimization:

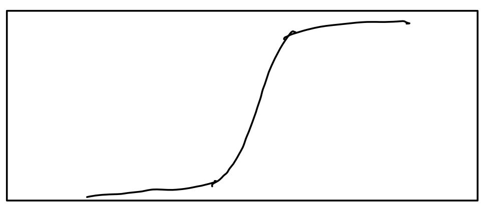
average logistic loss

$$L_{logistic}(w) = \frac{1}{m}\sum_{i=1}^{m} \underbrace{\log\left(1 + \exp(-y^i \cdot w^T x^i)\right)}_{\ell_{logistic}}$$

Goal: $\min\limits_{w} \left[L(w) \right]$

Enter the sigmoid function:

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



As $z \to$ large $\Rightarrow g(z) \to 1$

As $z \to$ small $\Rightarrow g(z) \to 0$

Fact: $g(z) + g(-z) = 1$. Proof:

$$\frac{1}{1+e^{-z}} + \frac{1}{1+e^{z}} \Rightarrow \frac{e^z}{e^z}\left(\frac{1}{1+e^{-z}}\right) + \frac{1}{1+e^{z}} \Rightarrow \frac{e^z}{e^z+1} + \frac{1}{1+e^z} = 1$$

■

$E[Y|X] = g(y \cdot w^T X)$ for some w. $\quad Y \in \{-1, +1\}$

$\Rightarrow Pr[Y=1 \mid X] = g(1 w^T x) = g(w^T x)$

Given X:

- if $w^T x$ is large : $\Rightarrow p[Y=1 \mid X] = \text{large}$
- if $w^T x$ is small or neg : $\Rightarrow p[Y=1 \mid X] = \text{small}$

soft relaxation of half-space.

$$\boxed{Pr[Y = y^i \mid x^i ; w] = g(y^i \cdot w^T x^i), \text{ where } g(x) = \frac{1}{1+e^{-x}}}$$

"Model for logistic regression"

Given a training set $S = \{(x^1, y^1), \cdots, (x^m, y^m)\}$, what is the most likely w given the training set? Use MLE.

$$\text{Likelihood}(w) = \prod_{i=1}^{m} p(Y = y^i \mid x^i, w) = \prod_{i=1}^{m} g(y^i \cdot w^T x^i)$$

max w

$$\text{log-likelihood} = \sum_{i=1}^{m} \log g(y^i \cdot w^T x^i) = \sum_{i=1}^{m} \log\left(\frac{1}{1 + \exp(-y^i w^T x^i)}\right)$$

$$= -\sum_{i=1}^{m} \underbrace{\log\left(1 + \exp(-y^i \cdot w^T x^i)\right)}_{\text{logistic loss}} = -m \cdot L(w)$$

average logistic loss $= \frac{1}{m} \sum \ell_{\text{logistic}}$.

Log-likelihood is maximized when $L(w)$ is minimized, due to $-m$.

Now our goal is to minimize the logistic loss $L(w)$.

Idea : run gradient descent on logistic loss.

This is the algorithm for performing logistic regression.

Note : There is no closed form, hence need for GD.

Let's compute the gradient of $\ell(w)$:

$$\ell_{logistic}(z) \equiv \log(1+e^{-z})$$

1) $$\ell'_{logistic}(z) = \frac{1}{1+e^{-z}} \cdot -e^{-z} = \frac{-1}{1+e^{+z}} = -g(-z)$$

2) Compute $$\frac{\partial \ell_{logistic}(y \cdot w^T x)}{\partial w_k} = \underbrace{-g(-y \cdot w^T x)}_{-g(-z)} \cdot \underbrace{y \cdot x^k}_{\text{chain rule } \frac{\partial z}{\partial w_k}}$$

↖ sigmoid function.

with this formula we can directly apply gradient descent; this precisely tells us how to find max-likelihood w.

What happens if we have multiple labels for y?

What if $y \in \{1, \dots, k\}$?

use multinomial logistic regression: $w^1, \dots, w^k$ weight vectors

or $k-1$

$$P[y=i \mid x] = \frac{e^{w^i \cdot x}}{\sum_{i=1}^{k} e^{(w^i)^T x}}$$

$$P[y=1 \mid x] \propto e^{(w^1)^T x}$$

$$P[y=i \mid x] \propto e^{(w^i)^T x}$$

$$P[y=k] = 1 - \sum_{i=1}^{k-1} P[y=i]$$

What is the associated loss? cross-entropy loss, Generalization of logistic loss.

$\underbrace{\text{(Imagine } y \text{ is a vector of length } k \text{ with a } 1 \text{ in the } j^{th} \text{ position if correct label is } j\text{).}}_{\text{one-hot encoding of labels}}$

Let's say our guess for the probability y has label i is p_i: $-\sum_{i=1}^{k} y_i \log(p_i)$ (this is cross-entropy loss).

softmax $\longrightarrow$ turns real-values into probabilities.

Example:

$W^T x$ via sigmoid$(W^T x) \xrightarrow{map} [0,1]$ probability space.

$$\underbrace{(z_1, \cdots, z_k)}_{k \text{ coordinates}} \longrightarrow \underbrace{\left(\frac{e^{z_1}}{Z}, \frac{e^{z_2}}{Z}, \cdots, \frac{e^{z_k}}{Z}\right)}_{\text{probability space; sums to } 1,}$$

$$\Rightarrow Z = \sum_{i=1}^{k} e^{z_i}$$