

Assignment 1 Solutions

Machine Learning
CS 6375, SO

Instructor:

Rishabh Iyer.

Assignment 1 Solutions

Q1] Which loss functions make sense for regression?

c) $L_i(w, b) = [f(w, b, x_i) - y_i]^3$

Does not make sense because this loss will be minimized when

$$f(w, b, x_i) - y_i \approx -\infty$$

and not when $f(w, b, x_i) - y_i \approx 0$

b) $L_i(w, b) = [f(w, b, x_i) - y_i]^4$

This does make sense because L_i is minimized when $f(w, b, x_i) \approx y_i$

c) $L_i(w, b) = \exp[f(w, b, x_i) - y_i]$

Does not make sense because this will be minimized when

$$f(w, b, x_i) - y_i \approx -\infty$$

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

and not when $f(w, b, x_i) \approx y_i$

Extra (Regression)

Properties of Loss functions (Regression)

① $L(y, \hat{y}) = 0$ iff $y = \hat{y}$

② $L(y, \hat{y}) \geq 0$ + y, \hat{y}

③ Good to have: Easy to minimise
 $L(y, \hat{y})$

Eg $L(y, \hat{y}) = \exp(y - \hat{y})$

satisfies ② but not ①

$$\exp(y - \hat{y}) = 0 \text{ if } y - \hat{y} = -\infty$$

$$d) L_i(w, b) = \max(0, -y_i f(w, b, x_i))$$

This loss fn also does not make sense because it is zero when $y_i f(w, b, x_i) \geq 0$

and not when $y_i f(w, b, x_i) < 0$

For e.g if $y_i = 2000$ & $f(w, b, x_i) = 1$

it will still be zero!

This is a Classification & not regression loss!

Note: Some students mentioned convexity.

It is a good to have but not a requirement.

Gradients

$$\nabla f_w(w, b, x_i) = x_i, \quad \nabla f_b(w, b, x_i) = 1.$$

$$a) \nabla L_i(w, b) = 3[f(w, b, x_i) - y_i]^2 \nabla f_w(w, b, x_i)$$

$$b) \nabla L_i^j(w, b) = 4(f(w, b, x_i) - y_i)^3 \nabla f_w(w, b, x_i)$$

$$c) \nabla L_i^j(w, b) = \exp[f(w, b, x_i) - y_i] \nabla f_w(w, b, x_i)$$

$$d) \nabla L_i^j(w, b) = -\sum_{y_j: f(w, b, x_j) < 0} y_j \nabla f_w(w, b, x_j)$$

Gradients for b are same. Just replace w with b .

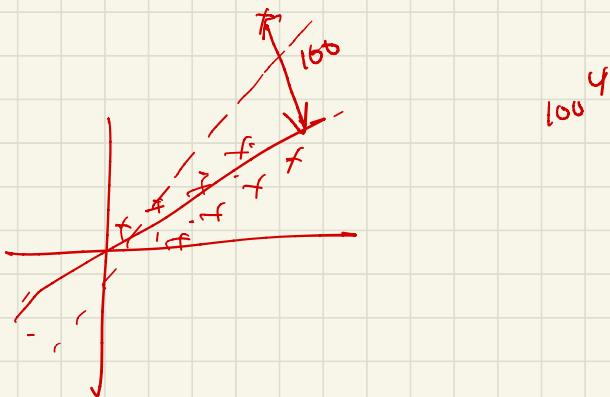
Bonus

$$[f(w, b, n_i) - y_i]^4 \text{ v/s } [f(w, b, n_i) - y_i]^2$$

The 4th power will be more sensitive to outliers. This can be a con for robust learning & a pro for fair learning (e.g. imbalance)

Another bonus:

How will you fix a) & c) to make it into a valid loss function?



Q2]

a) $L_i(w, b) = \max(0, 1 - y_i f(w, b, x_i))^2$

This is squared hinge loss. It is a good choice for classification, because like the hinge loss

$$L_i(w, b) = 0 \quad \text{if} \quad y_i f(w, b, x_i) \geq 1$$



b) $L_i(w, b) = [y_i - f(w, b, x_i)]^4$

Note: this is a regression loss.
but is not good for classification
because we want

$$\hat{y}_i \approx \text{sign}(f(w, b, x_i))$$

and not $y_i \approx f(w, b, x_i)$

Extra (Classfn)

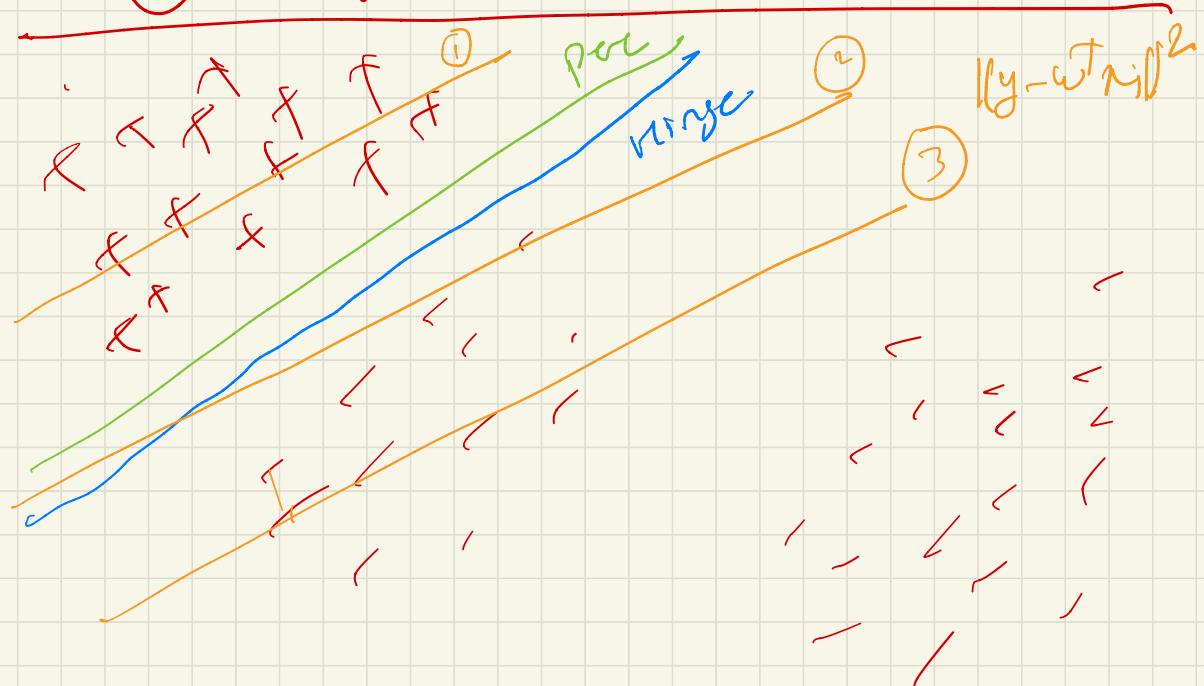
A loss fn $L(y, \hat{y})$ is good

for classfn if

Assume: $y \in \{0, 1\}$, $\hat{y} \in \mathbb{R}$

① $L(y, \hat{y}) = 0$ iff $y = \text{sign}(\hat{y})$

② $L(y, \hat{y}) \geq 0 \forall y, \hat{y}$



$$c) L_i(w, b) = \exp(f(w, b, x_i) - y_i)$$

This is neither good for classification nor regression because

again, it is minimized when

$$f(w, b, x_i) - y_i \approx -\infty$$

and not when

$$y_i = \text{sign}(f(w, b, x_i))$$

$$\text{OR } y_i; f(w, b, x_i) \geq 0$$

$$d) L_i(w, b) = \exp(-y_i f(w, b, x_i))$$

This is the exponential loss which is good for classification.

when $y_i f(w, b, n_i) \geq 0$,

the exp loss is ≤ 1

but is ≈ 0 only when

$y_i f(w, b, n_i) \approx -\infty$

gradives

a) $D_{L_w}(w, b, n_i) = 1_{y_i f(w, b, n_i) < 0} 2 y_i D_{fw}(w, b, n_i)$

[same with b]

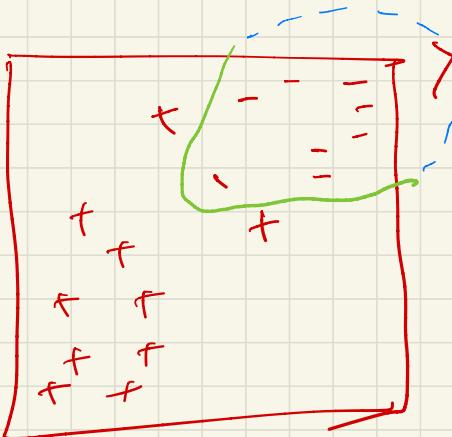
b) same as Q1.b)

c) same as Q1.c)

d) $D_{L_w}(w, b, n_i) = -\exp(-y_i f(w, b, n_i)) y_i D_{fw}(w, b, n_i)$

[same with b]

Q3)



$\times \leftarrow$ Not Ellipse/Circle
but parabola.
ellipse/circle will
overfit.

Features: $[1, n_1, n_2, n_1 n_2, n_1^2, n_2^2]$

Dataset is Linearly sep with quadratic features.

Q4)

$$\text{① } \min_w \sum_{i=1}^M [w^T x^i + b - y^i]^2$$

$$\text{s.t. } \|w - w_0\|^2 \leq k^2$$

$$L(w, b, \lambda) = \sum_{i=1}^M [w^T x^i + b - y^i]^2 + \lambda (\|w - w_0\|^2 - k^2)$$

$$\nabla L_w = 2 \sum_{i=1}^M (w^T x^i + b - y^i) x^i + 2\lambda (w - w_0) = 0$$

$$\nabla L_b = 2 \sum_{i=1}^M (w^T x^i + b - y^i) = 0$$

$$\Rightarrow b = \frac{\sum_{i=1}^M (y^i - w^T x^i)}{m}$$

Re (coll.)

$$2 \sum_{i=1}^M (w^T x^i + b - y^i) x^i + 2\lambda(w - w_0) = 0$$

$$\text{sub. } b = \frac{\sum_{i=1}^M (y^i - w^T x^i)}{M}$$

$$\Rightarrow 2 \sum_{i=1}^M (w^T x^i - y^i) x^i + 2 \sum_{i=1}^M \frac{(y^i - w^T x^i)}{M} \sum_{i=1}^M x^i + 2\lambda(w - w_0) = 0$$

Solve for $w = g(\lambda)$ and plug it back into
Dual.

$\beta \in \mathbb{R}^{n+1}$

Advanced approach

Write Least Squares as Matrix operation.

$$L(\beta) = (y - X\beta)^T (y - X\beta) \quad X = \begin{bmatrix} x_1 \\ \vdots \\ x_M \end{bmatrix}$$

$$\text{where } \beta = [w/b]$$

Then, note:

$$\min_{\beta} (y - X\beta)^T (y - X\beta)$$

$$\text{s.t. } \|\beta - \beta_0\|^2 \leq k^2 \quad \|y - X\beta\|^2$$

$$\Rightarrow L(\lambda, \beta) = (y - X\beta)^T (y - X\beta) + \lambda [\|\beta - \beta_0\|^2 - k^2]$$

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix} \approx \begin{bmatrix} w\beta \\ \vdots \\ w\beta \end{bmatrix} = X\beta$$

$$= X\beta$$

$$\frac{\partial L}{\partial \beta} = -2X^T(y - X\beta) + 2\lambda(\beta - \beta_0) = 0$$

$$\Rightarrow 2X^T X \beta - 2X^T y + 2\lambda(\beta - \beta_0) = 0$$

$$\Rightarrow \beta(X^T X + \lambda I) = X^T y + \lambda \beta_0$$

$$\Rightarrow \beta = (X^T X + \lambda I)^{-1}(X^T y + \lambda \beta_0)$$

Substitute β into $L(\beta, \gamma)$ to get

the dual

* Note: Connection to closed form of Linear Regression?

$$LR(\beta) = (y - X\beta)^T(y - X\beta)$$

$$\min_{\beta} LR(\beta), \quad \nabla_{\beta} LR(\beta) = 0$$

$$-2X^T(y - X\beta) = 0$$

$$X^T X \beta = X^T y$$

$$\Rightarrow \beta = \cancel{(X^T X)^{-1}} X^T y$$

Q4]

② $\min_{w, b} \frac{1}{2} \|w\|^2$

s.t. $y_i (w^T x_i + b) \geq 1$
 $\|w - w_0\|^2 \leq k^2$

$$L(w, b, \lambda, M) = \frac{1}{2} \|w\|^2$$

→ ①

$$+ \sum_{i=1}^M (1 - y_i (w^T x_i + b)) \lambda_i + M \left[\|w - w_0\|^2 - k^2 \right]$$

Now, $D L_w = w - \sum_{i=1}^M \lambda_i y_i x_i + 2M(w - w_0) = 0$

$$\Rightarrow w [1 + 2M] = \sum_{i=1}^M \lambda_i y_i x_i + 2M w_0$$

$$\Rightarrow w = \frac{\sum_{i=1}^M \lambda_i y_i x_i + 2M w_0}{1 + 2M}$$

→ ②

Sim, $D L_b = - \sum_{i=1}^M \lambda_i y_i = 0$ [constraint] → ③

Substitute w (②) into ① to get b (④)
with constraint as ③