FoDA L25

(Linear) Classification

↳ Loss functions

Classification core ML

Input (X, y) $X \subset \mathbb{R}^d$ $y \in \{-1, +1\}$

supervised problem $X = \{x_1, x_2, \dots x_n\}$ two classes

X attributes of data

y outcome

$$g(x) = \operatorname{sign}(f(x))$$
$$= \begin{cases} +1 & \text{if } f(x) \geq 0 \\ -1 & \text{if } f(x) < 0 \end{cases}$$

Goal: function $f : \mathbb{R}^d \to \{-1, +1\}$ $g(x_i) = y_i$

s.t. on data $(x_i, y_i) \in (X, y)$ $\cancel{f(x_i) = y_i}$

on as many as possible

assume $(x_i, y_i) \underset{iid}{\sim} \mu$ unknown dist   build f s.t. on

new data $(x, y) \sim \mu$ $\cancel{f(x) = y}$

$g(x) = y$

OCCAMS RAZOR

Simple model tend to generalize better.

Restrict f is linear function

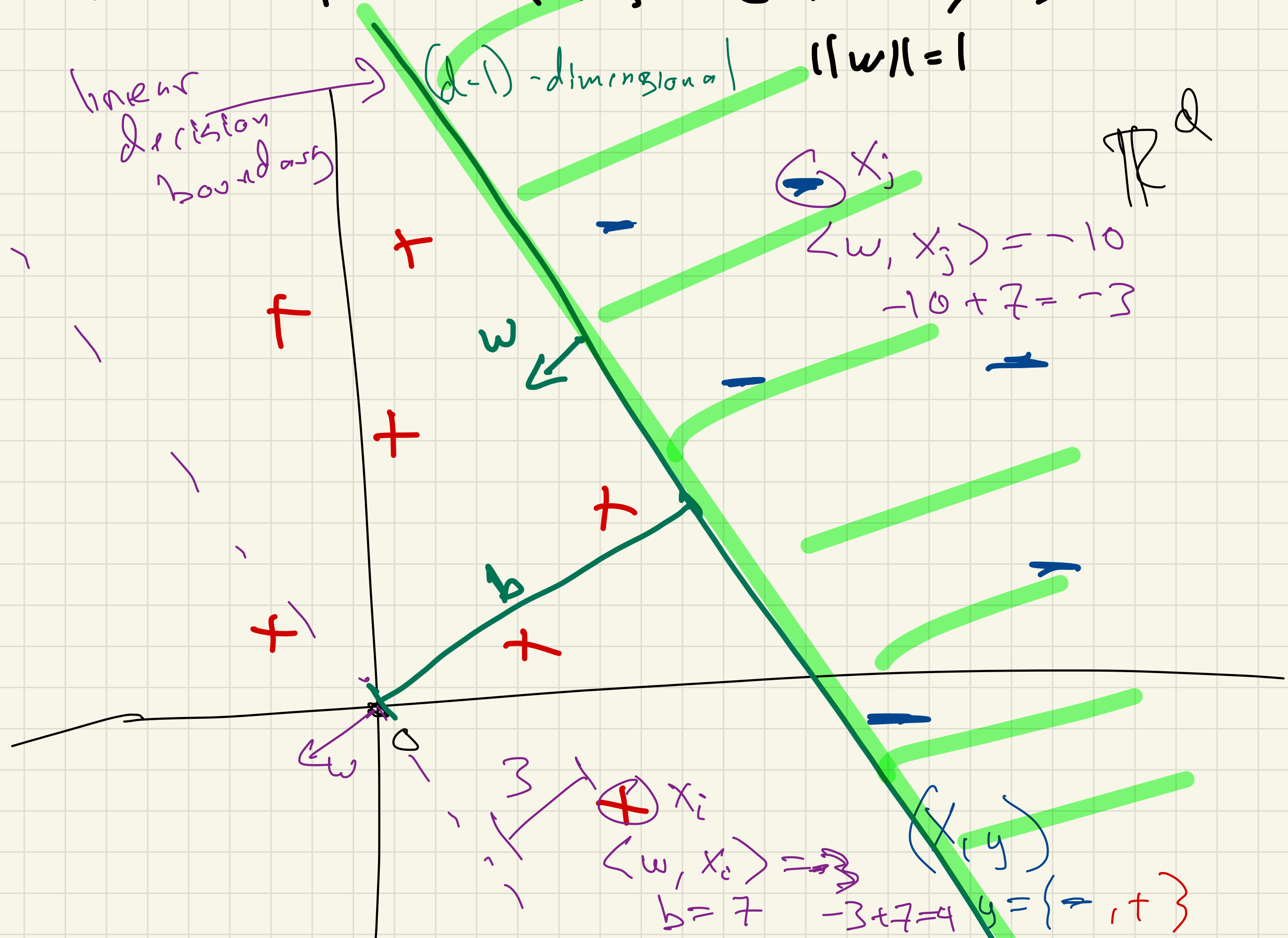
$$f(x) = \overset{b+}{\alpha_0} + \overset{w_1}{\alpha_1} x^{(1)} + \overset{w_2}{\alpha_2} x^{(2)} + \dots + \overset{w_d}{\alpha_d} x^{(d)}$$

$x = (x^{(1)}, x^{(2)}, \dots x^{(d)}) \in \mathbb{R}^d$

$\left.\begin{array}{l} b \in \mathbb{R} \\ w \in \mathbb{R}^d \end{array}\right)$ parameters of the model

typically $\|w\| = 1$

$g(x) = \text{sign}(f(x))$ $\qquad$ $f(x) = b + \langle w, x \rangle$



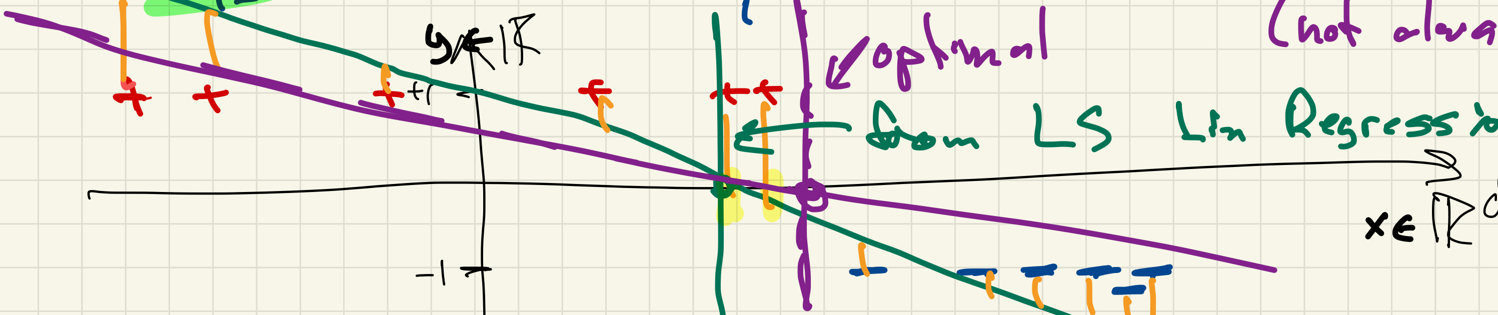
Input $(x_i, y_i) \subset \mathbb{R}^d \times \{-1, +1\}$

Goal: Find linear $f_{w,b} : \mathbb{R}^d \to \mathbb{R}$ so $\text{sign}(f(x_i)) = y_i$

How to solve for w, b? $f(x) = b + \langle w, x \rangle$

Apply least squares Linear Regression Algorithm.

solves for w, b minimize $\underbrace{\sum_i (f(x_i) - y_i)^2}$ ok.... (not always)



optimal

from LS Lin Regression

Can we use GD to minimize?

$g_\alpha = \text{sign}(f_\alpha(\cdot))$

$$\Delta(g_\alpha(X,y)) = \sum_{i=1}^{n} \left(1 - \mathbb{1}\left(\text{sign}(y_i) = \text{sign}(f_\alpha(x_i))\right)\right)$$

$y_i := g_\alpha(x_i)$

= # misclassified points.

identity function

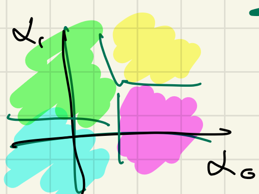
$\mathbb{1} : \{\text{true}, \text{false}\} \to \{0, 1\}$

$$\mathbb{1} = \begin{cases} 1 & \text{if True} \\ 0 & \text{if false} \end{cases}$$

No: can't use GD

- not convex
- no gradient!

$\Delta(g_\alpha(X,y))[\alpha]$ peice-wise constant



Loss Functions (approximate Δ)

$$f(\alpha) = \mathcal{L}(g_{\alpha}, (X, y)) = \sum_{i=1}^{n} \ell(g_{\alpha}, \underbrace{(x_i, y_i)}_{\text{bivariate}})$$

$$= \sum_{i=1}^{n} \ell_{\alpha}(\underbrace{z_i}_{\text{univariate}})$$

$$\boxed{z_i = y_i \, g_{\alpha}(x_i)}$$

$$= \sum_{i=1}^{n} f_i(\alpha)$$

$$\boxed{f_i(\alpha) = \ell_{\alpha}(z_i = y_i g_{\alpha}(x_i))}$$

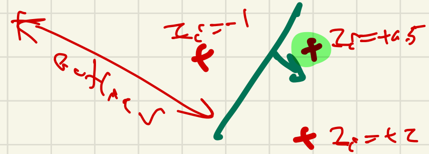
f is decomposible

$$y_i \underbrace{g_{\alpha}}_{\text{linear}}(x_i) = \begin{cases} >0 & \text{if } y_i < 0,\ g_{\alpha}(x_i) < 0 \\ >0 & \text{if } y_i > 0,\ g_{\alpha}(x_i) > 0 \\ <0 & \text{otherwise (bad prediction)} \end{cases}$$

ℓ_{α} is good if

1. convex
2. has gradient
3. approximate Δ

$$\Delta(z) = \begin{cases} 0 & \text{if } z \geq 0 \\ 1 & \text{if } z < 0 \end{cases}$$



$z_i = y_i \, g_\alpha(x_i)$

good if > 0, otw bad

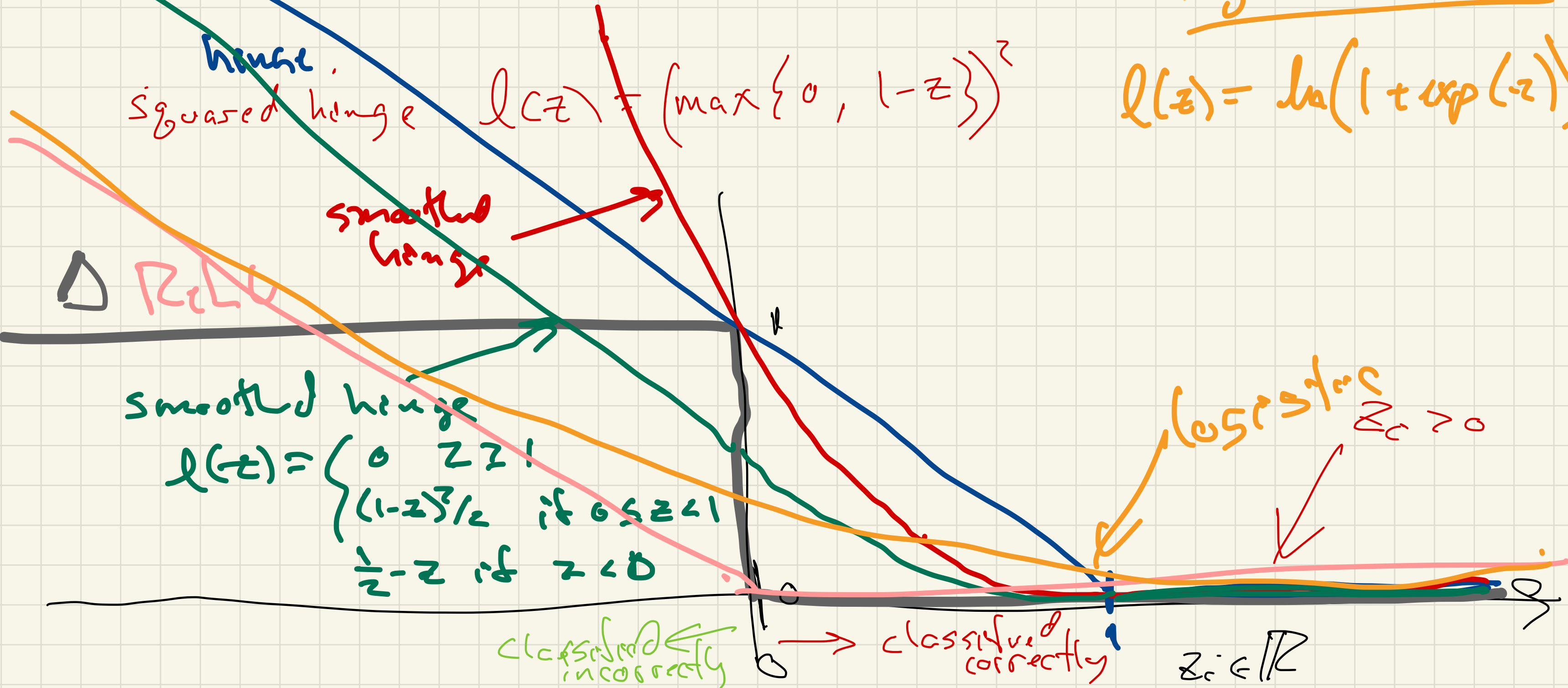
loss functions

hinge loss $\ell(z) = \max\{0, 1-z\}$

squared hinge $\ell(z) = (\max\{0, 1-z\})^2$

logistic loss

$\ell(z) = \ln(1+\exp(-z))$



smoothed hinge

$$\ell(z) = \begin{cases} 0 & z \geq 1 \\ (1-z)^2/2 & \text{if } 0 \leq z < 1 \\ \frac{1}{2} - z & \text{if } z < 0 \end{cases}$$