Squared error "loss" can be problematic

Classifier design $\min_w \ell(w; A, d) + \lambda r(w)$ — regularized

Squared error loss $\ell(w; A, d) = \|Aw - d\|_2^2$ — loss function

Example: dwarf planet vs. planet

	object	Ceres	Eris	Pluto	Mercury	Earth	Jupiter
x_i	radius ($\times 10^6$)	1.0	2.3	2.4	4.9	12.8	143.0
d_i	label	-1	-1	-1	1	1	1

$$A = \begin{bmatrix} 1 & 1 \\ 2.3 & 1 \\ 2.4 & 1 \\ 4.9 & 1 \\ 12.8 & 1 \\ 143 & 1 \end{bmatrix}, \quad d = \begin{bmatrix} -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \qquad w_{LS} = (A^TA)^{-1}A^Td \approx 0.01\begin{bmatrix} 1 \\ -28 \end{bmatrix}$$

dwarf: $x_i < 28$ (earth!)

planet: $x_i > 28$

squared error → poor classification

Avoid loss due to easy to classify data

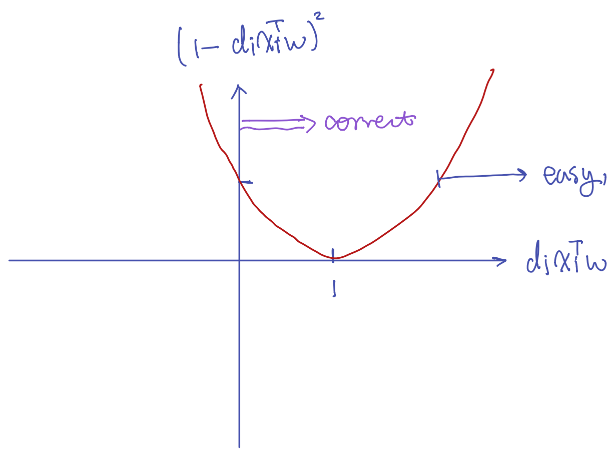
max margin classifier: midpoint between pluto and Mercury: $(4.9-2.4)\div 2+2.4=3.65$

$$\hat{d} = \text{sign}(x - 3.65)$$

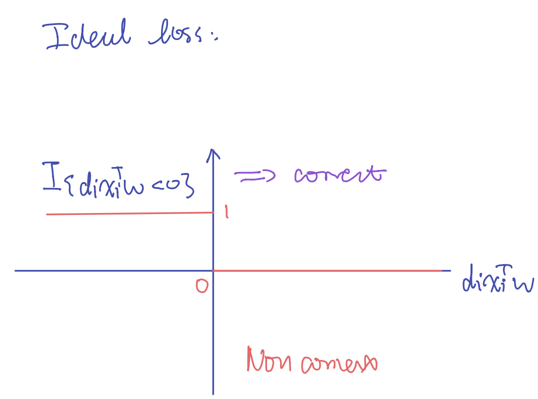
margin: $4.9 - 2.4 = 2.5$ (class separation)

Squared error loss: $\|Aw-d\|_2^2 = \sum_{i=1}^{N}(d_i - x_i^Tw)^2 = \sum_i (1 - d_i x_i^T w)^2$ $(d_i = \pm 1)$

correct classification: $d_i x_i^T w > 0$

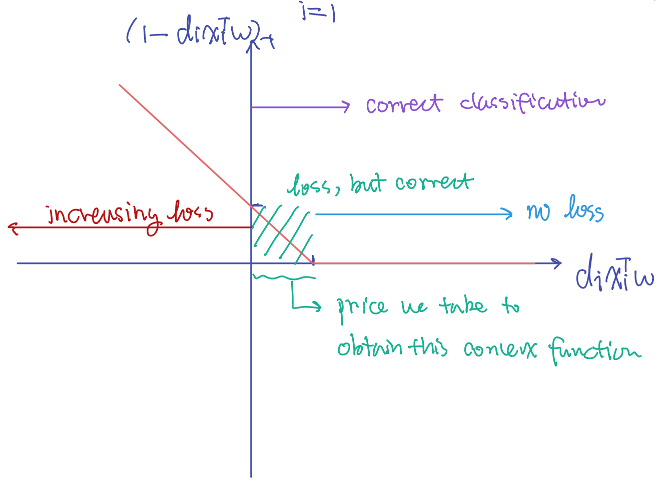


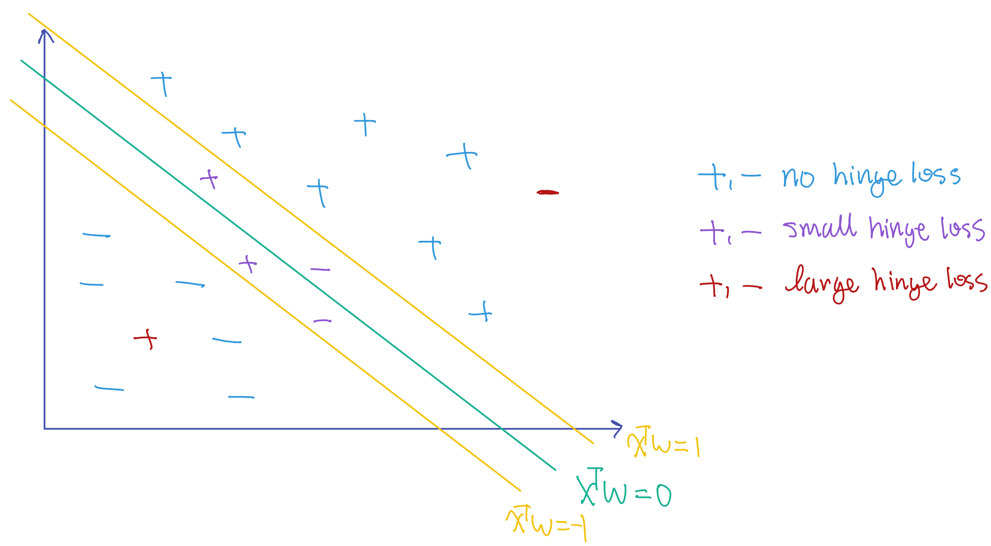
Ideal loss:



Hinge loss is convex and has no loss for easy to classify data

$$\ell(w; A, d) = \sum_{i=1}^{N} (1 - d_i x_i^T w)_+ \qquad (\alpha)_+ = \begin{cases} \alpha & \alpha > 0 \\ 0 & \alpha \le 0 \end{cases}$$





Hinge loss better approximates ideal: number of misclassification

Iterative algorithms required for finding minimum hinge loss classifier