4/24/20

Logistic Regression using Softmax Cost

derive logistic regression using $\underline{\underline{y_p}} = \{-1, +1\}$

- Softmax cost equivalent to the cross entropy cost
- Sew together
  - → regression approach
  - → perceptron "

$N=1$ $N=2$

regression

Perceptron

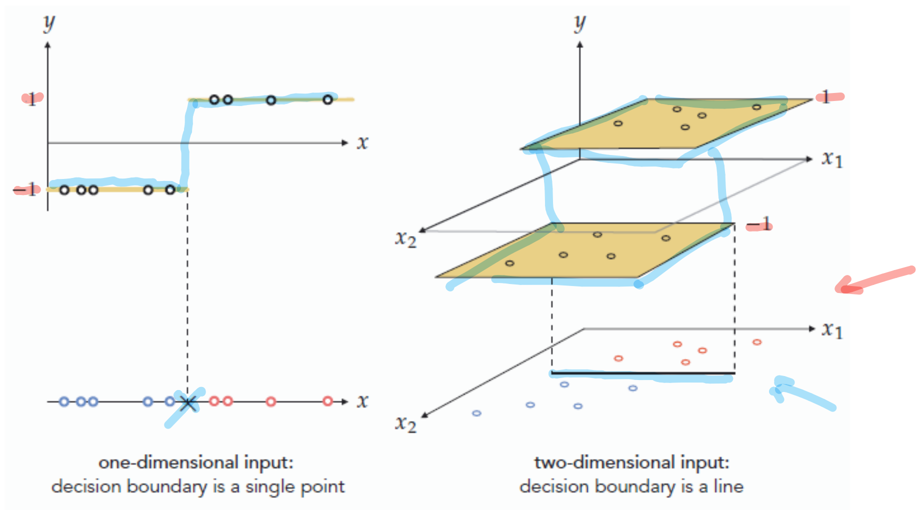


Figure 6.8 The analogous setup to Figure 6.1, only here we use label values $y_p \in \{-1, +1\}$.

linear classification

again, we want to approximate this "step" function

$$\operatorname{sign}(x) = \begin{cases} +1, & x > 0 \\ -1, & x < 0 \end{cases}$$

$$\operatorname{sign}(\tilde{x}^T \tilde{w})$$

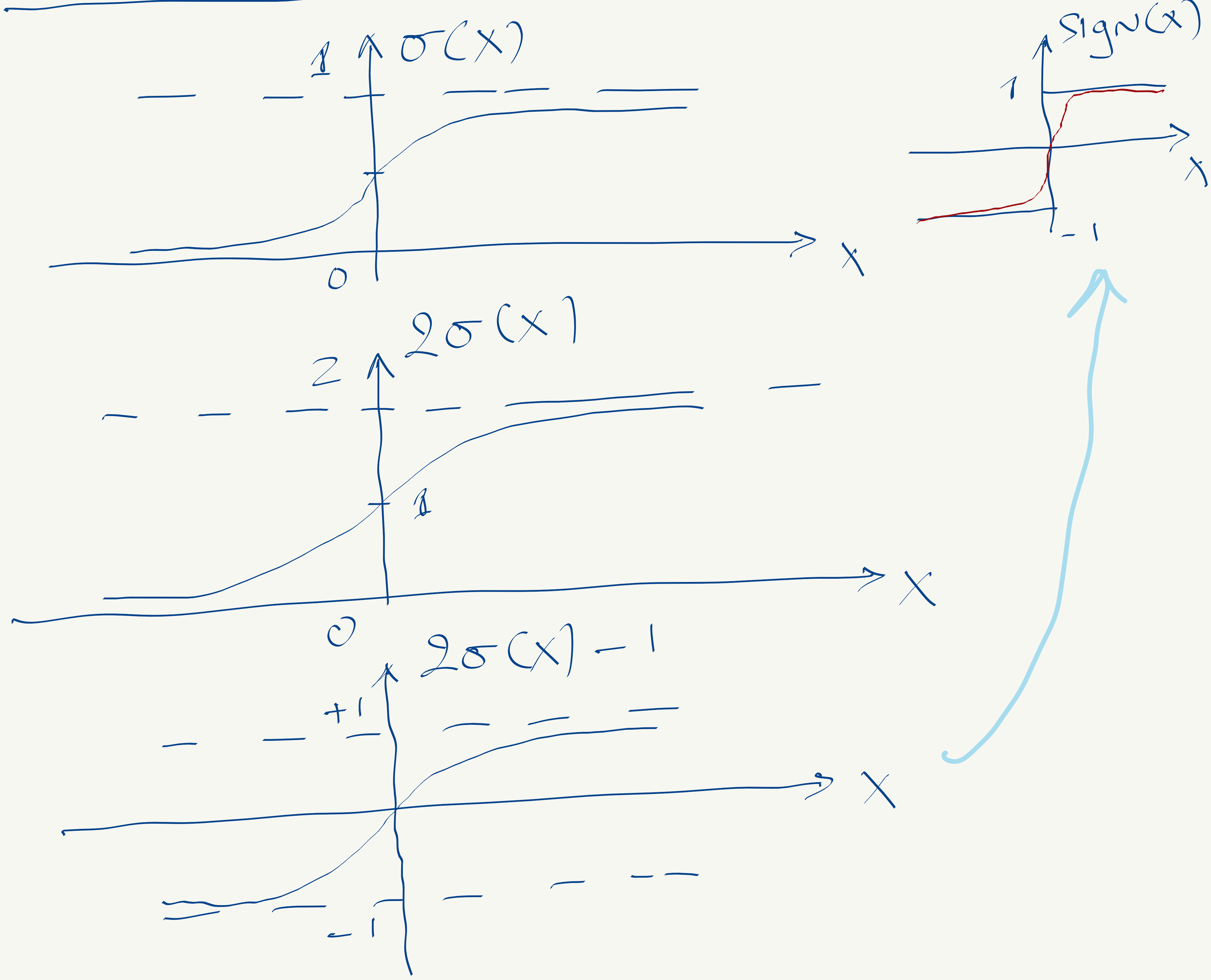
linear classifier: $\underline{\tilde{x}}^T \tilde{w} = 0$

correct classification: $\boxed{\operatorname{sign}(\tilde{x}_p^T \tilde{w}) = y_p}$

$$\Rightarrow \boxed{\begin{array}{l}\underline{\text{LS cost}} \\ g(\tilde{w}) = \frac{1}{P} \sum_{p=1}^{P} \left(\underline{\operatorname{sign}}(\tilde{x}_p^T \tilde{w}) - y_p \right)^2 \end{array}}$$

- flat $\Rightarrow$ difficulty optimizing

Smooth approximations



$$\tanh(x) = 2\sigma(x) - 1$$

LS:

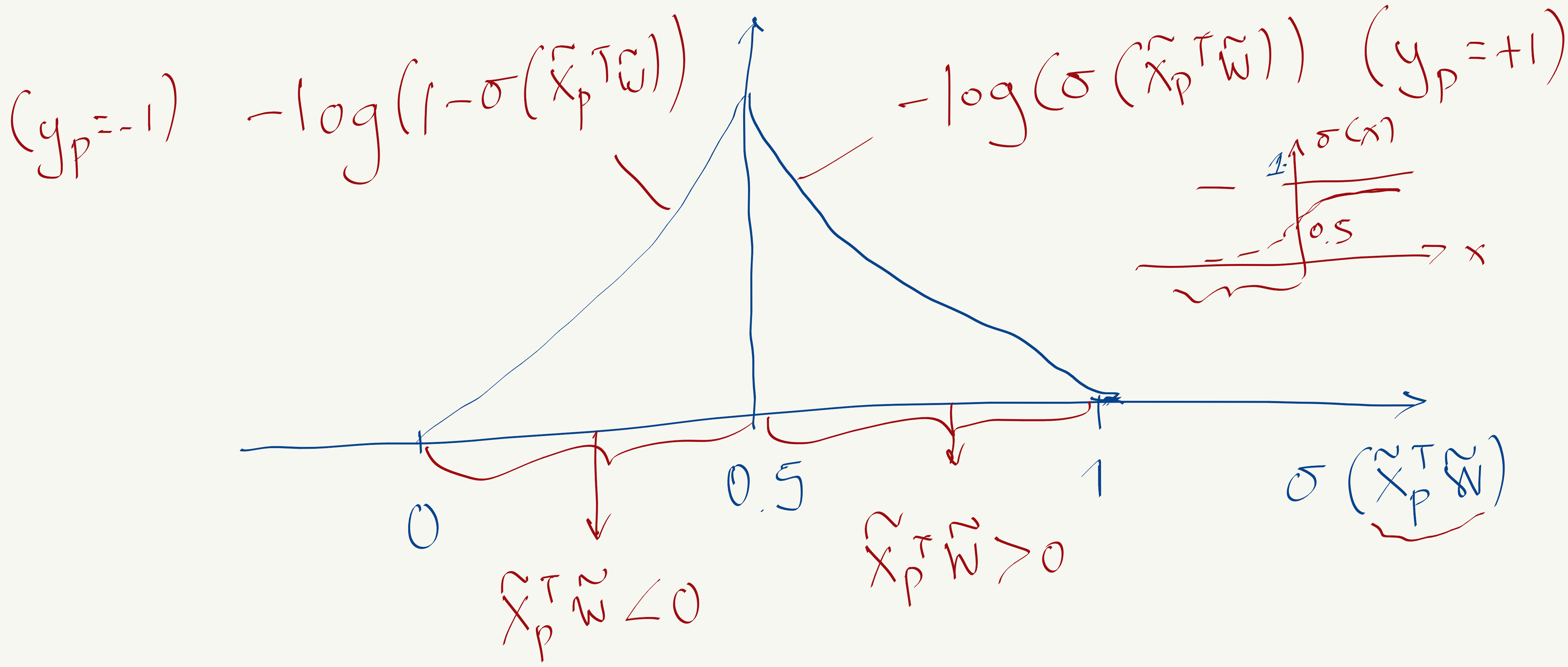
$$\boxed{g(\tilde{w}) = \frac{1}{P}\sum_{p=1}^{P}\left(\tanh\left(\tilde{x}_p^T \tilde{w}\right) - y_p\right)^2}$$

- Non-convex
- Undesirable flat region

  → Normalized GD to optimize it

Pairwise log error

(1) $$g_p(\tilde{w}) = \begin{cases} -\log\left(\sigma\left(\tilde{x}_p^T \tilde{w}\right)\right), & y_p = 1 \\ -\log\left(1-\sigma\left(\tilde{x}_p^T \tilde{w}\right)\right), & y_p = -1 \end{cases}$$



$$g(\tilde{w}) = \frac{1}{P}\sum_{p=1}^{P} g_p(\tilde{w})$$

Can we make it more compact?

(2) $$1-\sigma(x) = 1 - \frac{1}{1+e^{-x}} = \frac{1+e^{-x}-1}{1+e^{-x}} = \frac{e^{-x}}{1+e^{-x}} = \frac{1}{e^{x}+1} = \sigma(-x)$$

(1), (2) :

$$g_p(\tilde{w}) = \begin{cases} -\log(\sigma(\tilde{x}_p^T \tilde{w})) , & y_p = +1 \\ -\log(\sigma(-\tilde{x}_p^T \tilde{w})), & y_p = -1 \end{cases}$$

$$\Rightarrow \boxed{g_p(\tilde{w}) = -\log(\sigma(y_p \tilde{x}_p^T \tilde{w}))}$$

if $y_p = 1 \rightarrow g_p(\tilde{w}) = -\log(\sigma(\tilde{x}_p^T \tilde{w}))$

if $y_p = -1 \rightarrow g_p(\tilde{w}) = -\log(\sigma(-\tilde{x}_p^T \tilde{w}))$

- $\log \frac{1}{x} = -\log x$

$$g_p(\tilde{w}) = \log\left(\frac{1}{\sigma(y_p \tilde{x}_p^T \tilde{w})}\right)$$

$$\Rightarrow \boxed{g_p(\tilde{w}) = \log\left(1 + e^{-y_p \tilde{x}_p^T \tilde{w}}\right)}$$

<u>Softmax logistic regression</u>

* $$g(\tilde{w}) = \frac{1}{P} \sum_{p=1}^{P} \log\left(1 + e^{-y_p \tilde{x}_p^T \tilde{w}}\right)$$

- <u>convex</u>

$$\nabla g(\tilde{w}) = \left(-\frac{1}{P} \sum_{p=1}^{P} \frac{e^{-y_p \underline{\tilde{x}_p^T \tilde{w}}}}{1 + e^{-y_p \underline{\tilde{x}_p^T \tilde{w}}}} y_p \right) \underline{\underline{\tilde{x}_p}} \quad (N+1) \times 1$$

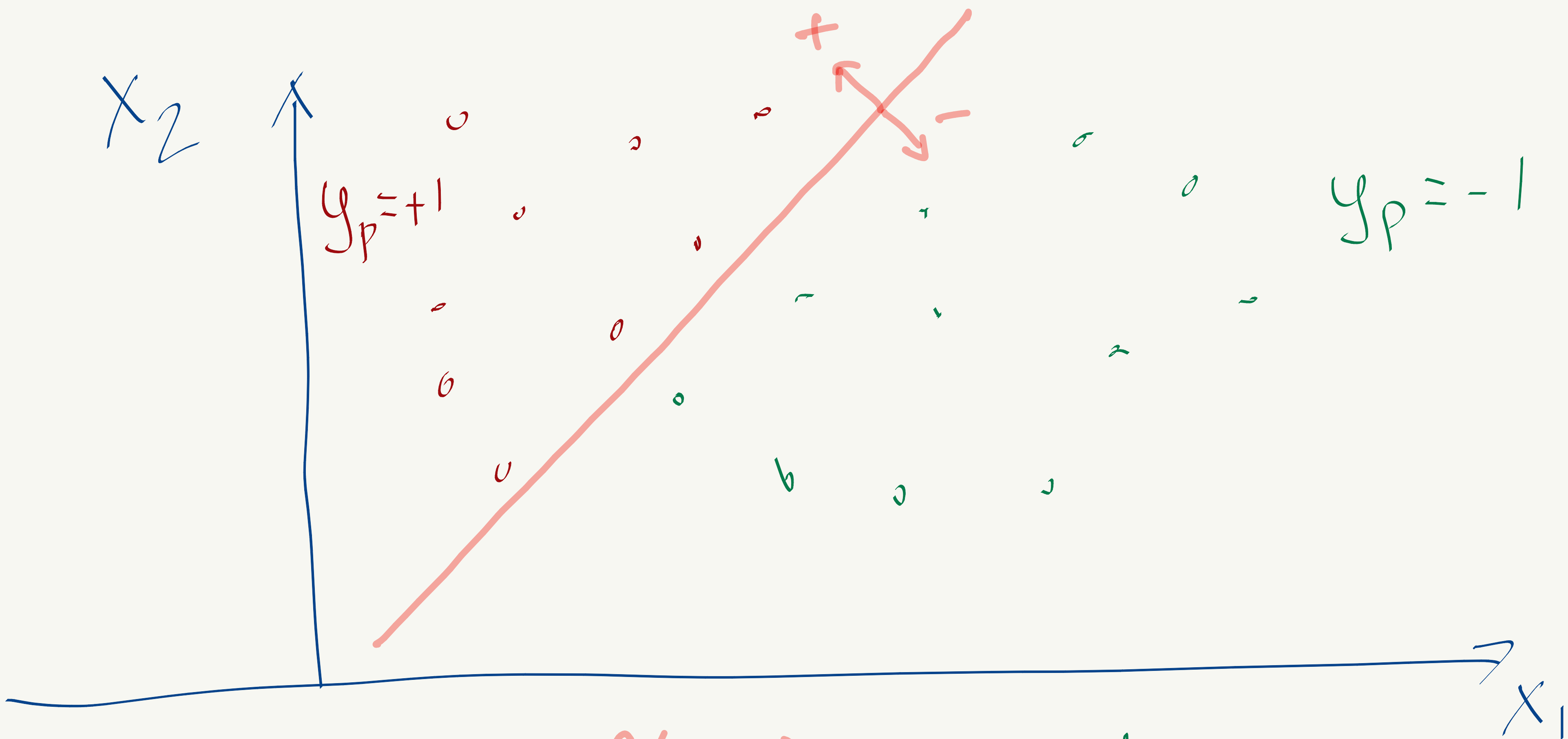
$$\nabla^2 g(\tilde{w}) = \begin{bmatrix} \nabla^T (\nabla g(\tilde{w}) |_{w_0}) \\ \nabla^T (\nabla g(\tilde{w}) |_{w_1}) \\ \vdots \\ \nabla^T (\nabla g(\tilde{w}) |_{w_N}) \end{bmatrix} \qquad \tilde{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_N \end{bmatrix}$$

$$\Rightarrow \boxed{\nabla^2 g(\tilde{w}) = \frac{1}{P} \sum_{p=1}^{P} (\text{scalar}_1)(\text{scalar}_2) \underbrace{\tilde{x}_p \tilde{x}_p^T}_{\text{outer product}}}$$

$$z^T \underline{\underline{A}} z \geq 0, \quad \forall z$$

Perceptron

$\{(\bar{x}_p, y_p)\}_{p=1}^{P}$, $y_p \in \{-1, +1\}$



$h(\bar{x}) = \tilde{x}^T\tilde{w} = 0$ hyperplane

$$\begin{cases} \text{if } \tilde{x}_p^T\tilde{w} > 0, & y_p = 1 \\ \text{if } \tilde{x}_p^T\tilde{w} < 0, & y_p = -1 \end{cases}$$

$$(\tilde{x}_p^T\tilde{w})\,y_p > 0 \Rightarrow \boxed{-y_p(\tilde{x}_p^T\tilde{w}) < 0}$$

should be satisfied for correct classification

$$\Rightarrow g_p(\tilde{w}) = \underline{\max}\Big(0, \underbrace{-y_p(\tilde{x}_p^T\tilde{w})}_{\text{negative}}\Big) = 0$$

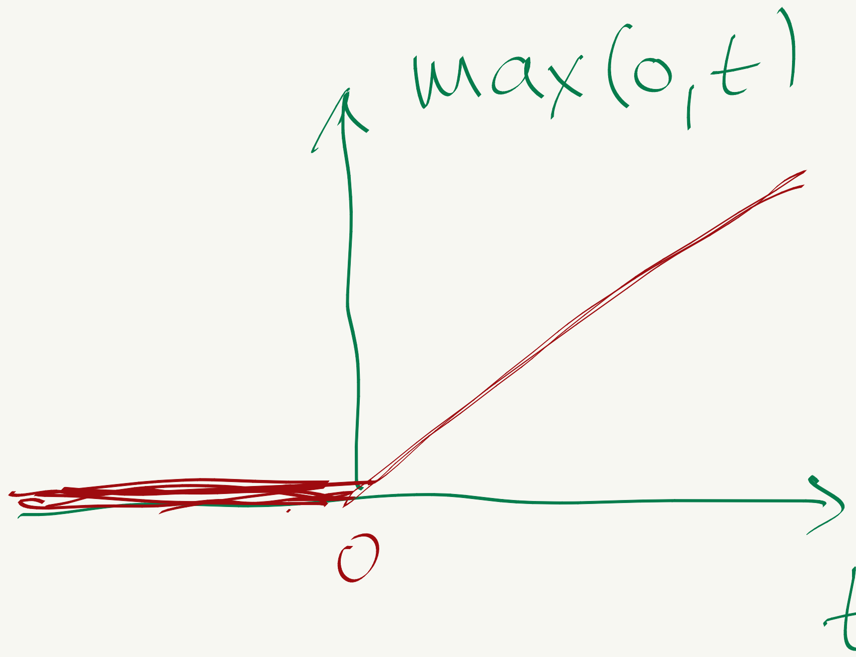
Loss Function

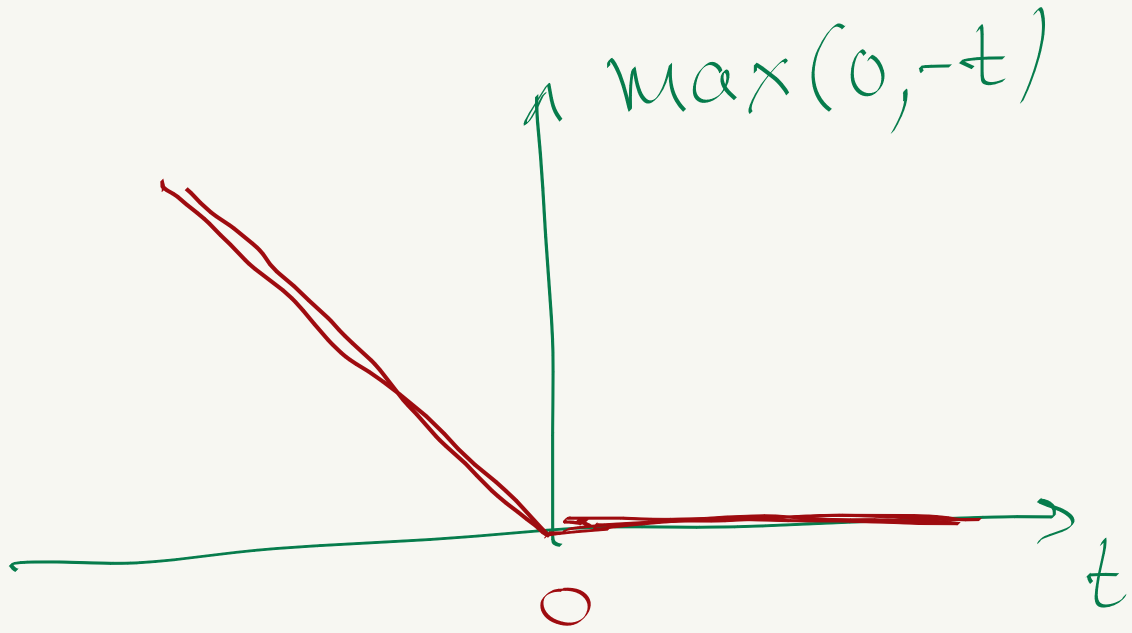
$$\boxed{g(\tilde{w}) = \frac{1}{P} \sum_{p=1}^{P} \max\left(0, -y_p \tilde{x}_p^T \tilde{w}\right)}$$

$$\tilde{w}^* = \min_{\tilde{w}} g(\tilde{w})$$

Perceptron, hinge cost, max cost function

rectified linear unit (ReLU)





- convex ✓
- non-dif'able at $t=0$
- trivial solution $t=0$

$$t \to -y_p \tilde{x}_p^T \tilde{w} = 0 \Rightarrow \tilde{w} = 0$$

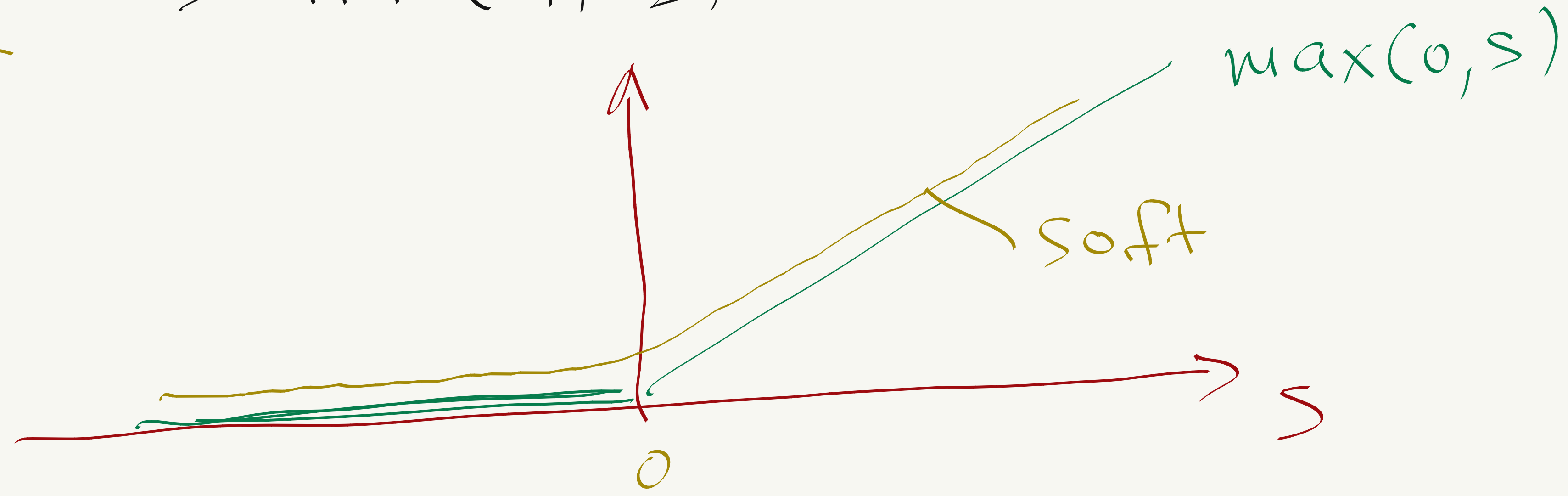
Approximations of max(0,t) function

definition of softmax function

$$\mathrm{soft}(s_1, s_2) = \log\left(e^{s_1} + e^{s_2}\right)$$

∴ we can show that

$$\mathrm{soft}(s_1, s_2) \simeq \max(s_1, s_2)$$



if $s_1 >> s_2 \Rightarrow e^{s_1} >> e^{s_2}$

$\Rightarrow \log\left(e^{s_1} + e^{s_2}\right) \simeq \log\left(e^{s_1}\right) = s_1$

- $\max(s_1, s_2) = s_1$

$$g_p(\tilde{w}) = \max(0, -y_p \tilde{x}_p^T \tilde{w})$$

$$\approx \operatorname{soft}(0, -y_p \tilde{x}_p^T \tilde{w})$$

$$= \log(e^0 + e^{-y_p \tilde{x}_p^T \tilde{w}})$$

$$= \log(1 + e^{-y_p \tilde{x}_p^T \tilde{w}})$$

<u>Softmax logistic regression</u>