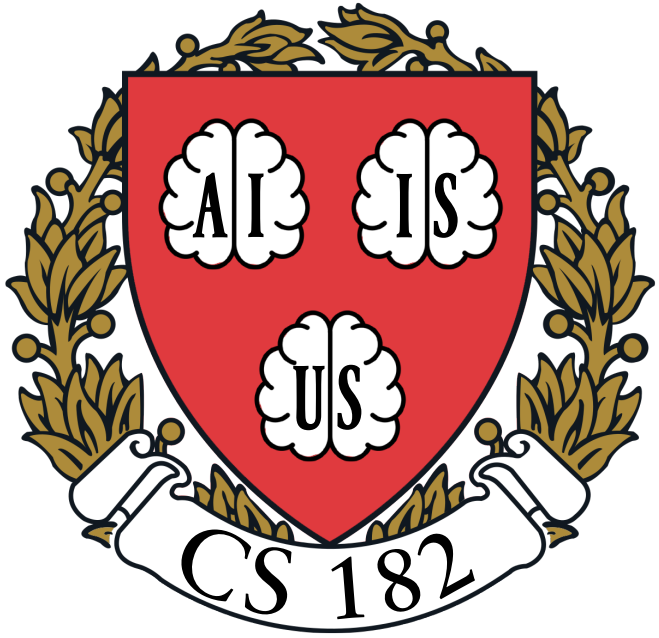
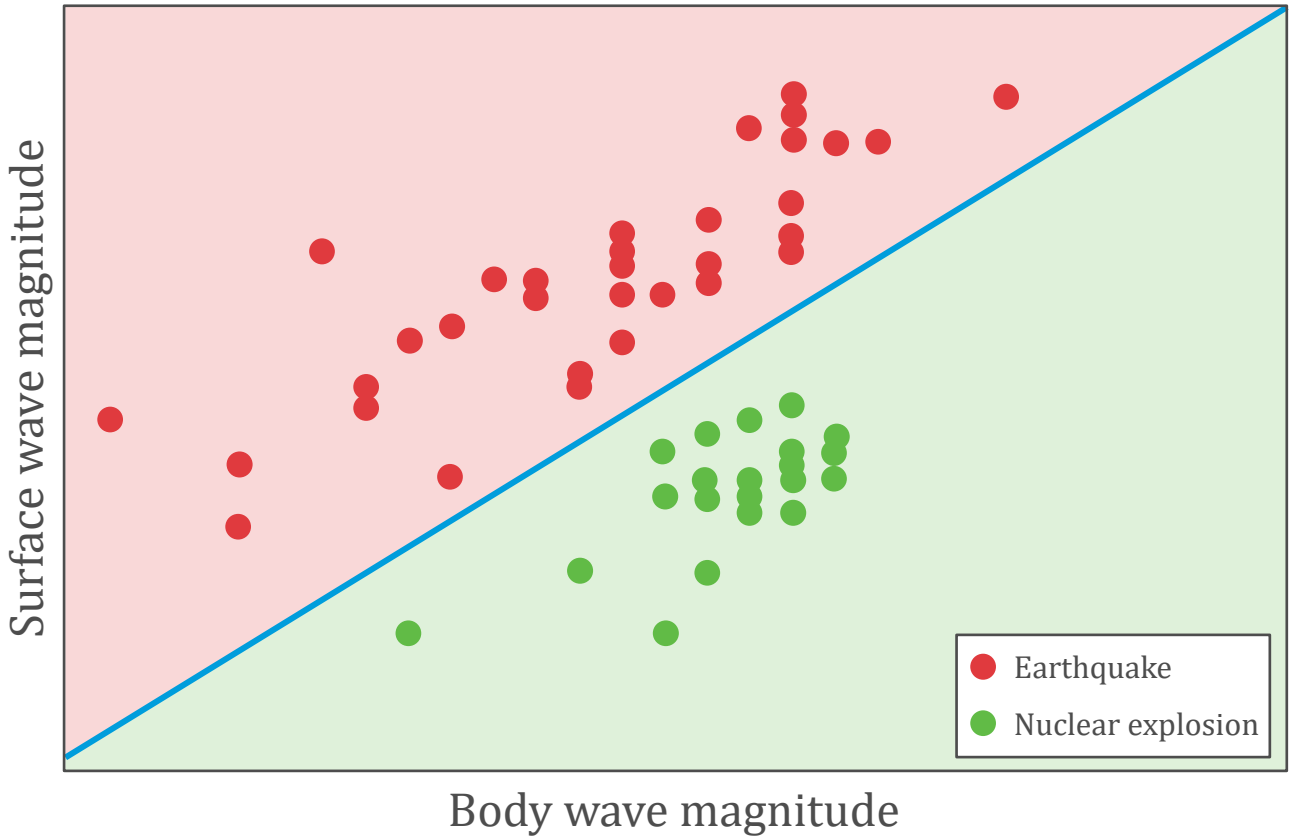


Fall 2022 | Lecture 18

Linear Classification

Ariel Procaccia | Harvard University

LINEAR CLASSIFICATION



Earthquakes and nuclear explosions in the Middle East and Asia, 1982-1990

LINEAR CLASSIFICATION

- A hypothesis is defined by
$$h_{\boldsymbol{w}}(\boldsymbol{x}) = \text{Threshold}(\boldsymbol{w} \cdot \boldsymbol{x})$$
where $\text{Threshold}(z) = +1$ if $z \geq 0$ and $\text{Threshold}(z) = -1$ if $z < 0$
- A linear separator can be found via a linear feasiblity program

$$\begin{aligned} &\text{find } \boldsymbol{w} \\ &\text{s.t. } \forall i \in \mathcal{D}^+, \boldsymbol{w} \cdot \boldsymbol{x}^{(i)} \geq 0 \\ &\qquad \forall i \in \mathcal{D}^-, \boldsymbol{w} \cdot \boldsymbol{x}^{(i)} \leq -\epsilon \end{aligned}$$

- But we want to learn online

The New York Times

July 13, 1958

Eléctronic 'Brain' Teaches Itself

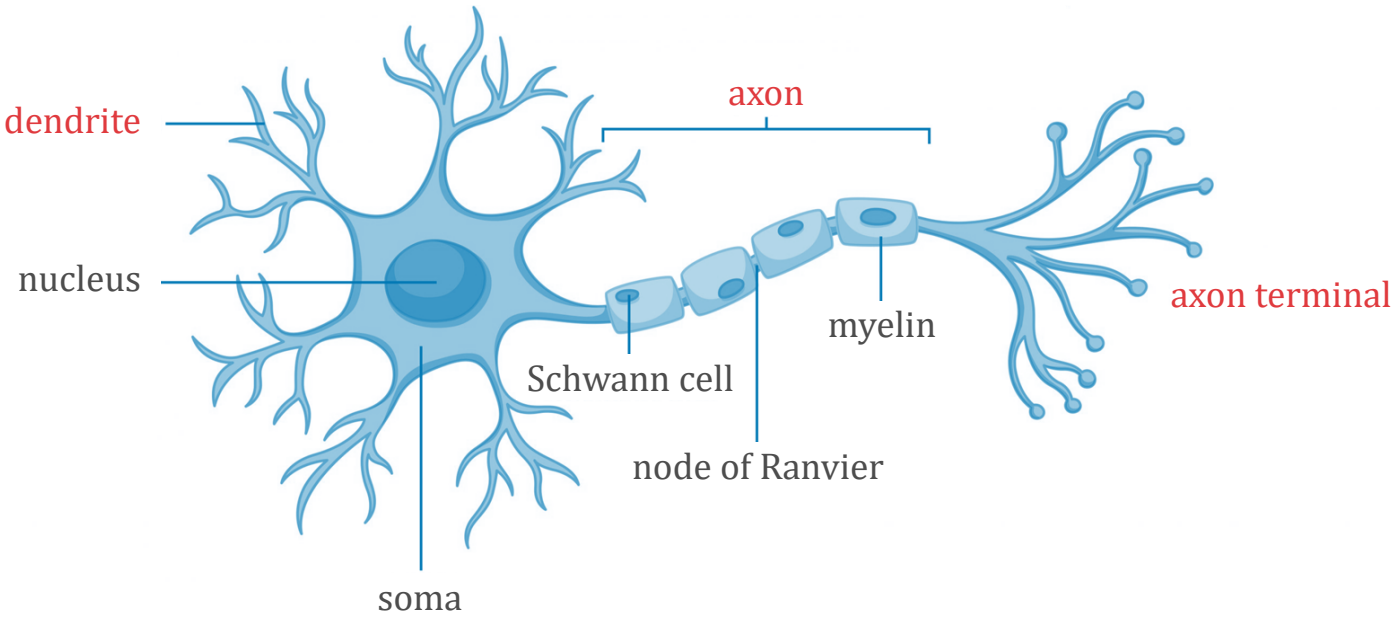
The Navy last week demonstrated the embryo of an electronic computer named the Perceptron which, when completed in about a year, is expected to be the first non-living mechanism able to "perceive, recognize and identify its surroundings without human training or control." Navy officers demonstrating a preliminary form of the device in Washington said they hesitated to call it a machine because it is so much like a "human being without life."

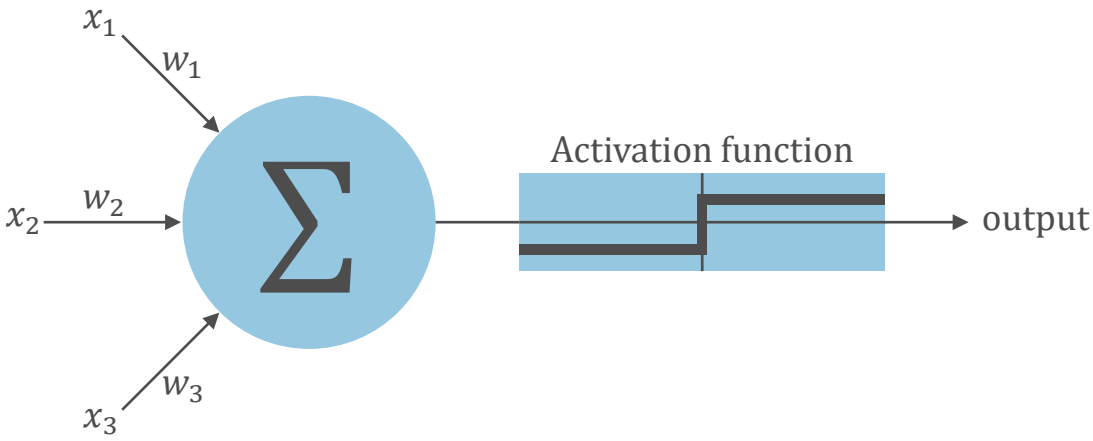
recognize the difference between right and left, almost the way a child learns.

When fully developed, the Perceptron will be designed to remember images and information it has perceived itself, whereas ordinary computers remember only what is fed into them on punch cards or magnetic tape.

Later Perceptrons, Dr. Rosenblatt said, will be able to recognize people and call out their names. Printed pages, longhand letters and even

ELECTRONIC BRAIN

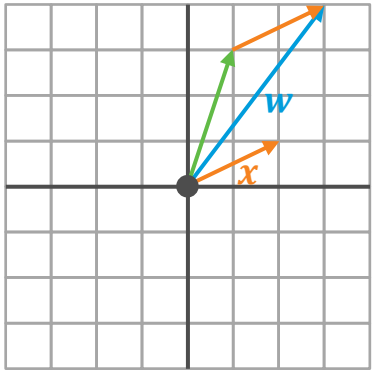




PERCEPTRON

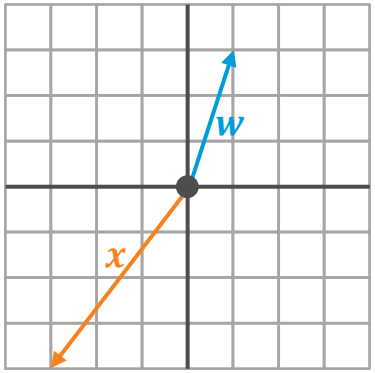
The Perceptron learning rule: For each example $(\boldsymbol{x}, y)$, classify $\hat{y} = \text{Threshold}(\boldsymbol{w} \cdot \boldsymbol{x})$ and, If $\hat{y} \neq y$, update $\boldsymbol{w} = \boldsymbol{w} + y \cdot \boldsymbol{x}$

Stage k



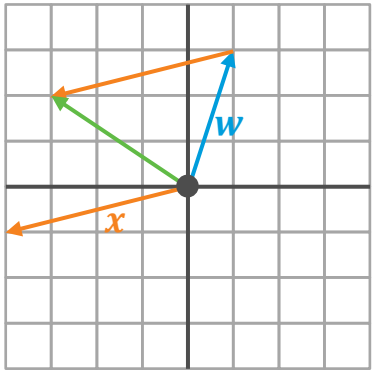
$\boldsymbol{w} = (3,4)$
$(\boldsymbol{x}, y) = \big((2,1), -1\big)$
$\boldsymbol{w} \cdot \boldsymbol{x} = 10$
$\boldsymbol{w} = (3,4) - (2,1) = (1,3)$

Stage $k+1$



$\boldsymbol{w} = (1,3)$
$(\boldsymbol{x}, y) = \big((-3,-4), -1\big)$
$\boldsymbol{w} \cdot \boldsymbol{x} = -15$
$\boldsymbol{w}$ is unchanged

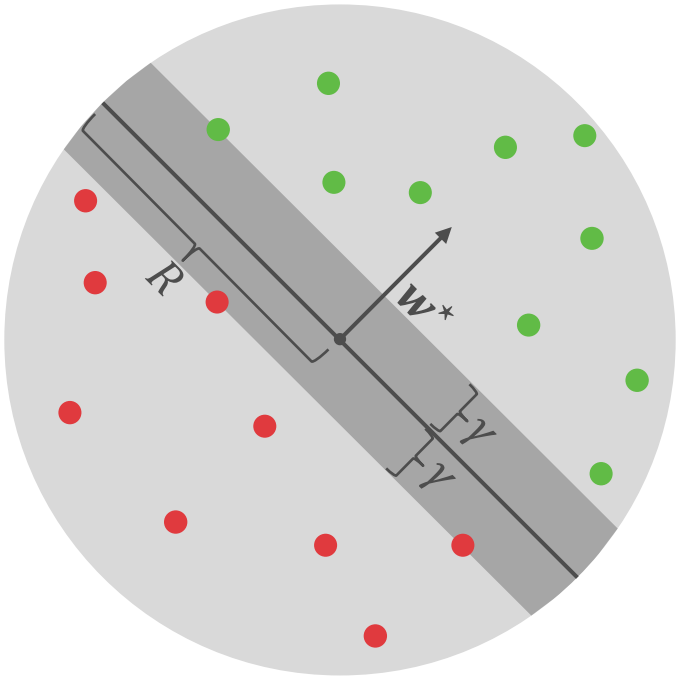
Stage $k+2$



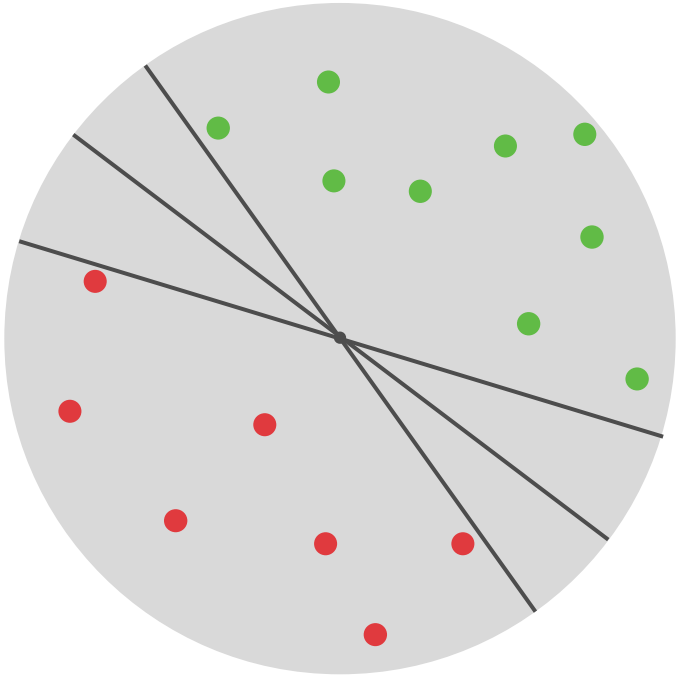
$\boldsymbol{w} = (1,3)$
$(\boldsymbol{x}, y) = \big((-4,-1), 1\big)$
$\boldsymbol{w} \cdot \boldsymbol{x} = -7$
$\boldsymbol{w} = (1,3) + (-4,-1) = (-3,2)$

PERCEPTRON MISTAKE BOUND

Theorem: Given a dataset $\left\{\left(\boldsymbol{x}^{(i)}, y^{(i)}\right)\right\}_{i=1}^{m}$, if $\left\|\boldsymbol{x}^{(i)}\right\| \leq R$ for all i, and there exists $\boldsymbol{w}^{\star}$ such that $\|\boldsymbol{w}^{\star}\| = 1$ and $y^{(i)}\left(\boldsymbol{w}^{\star} \cdot \boldsymbol{x}^{(i)}\right) \geq \gamma$ for all i, then the number of mistakes made by the Perceptron is at most $(R/\gamma)^2$

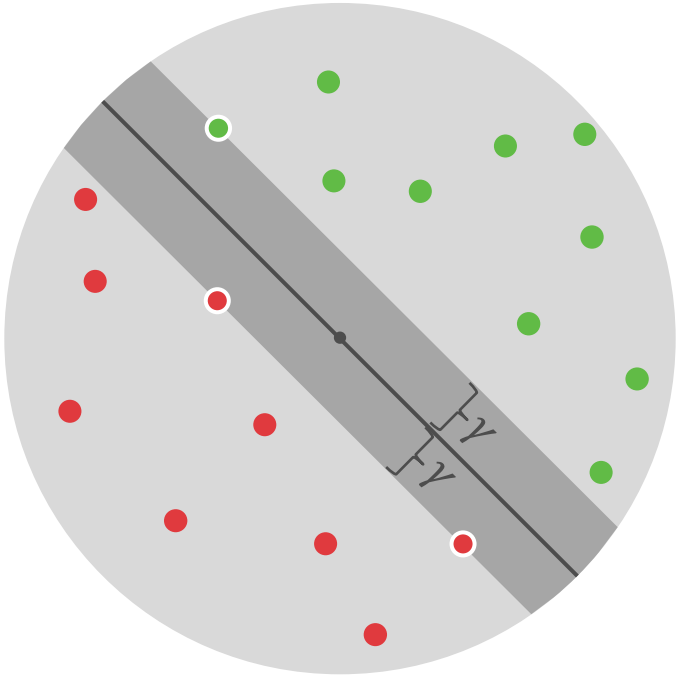


GENERALIZATION



Which separator would you expect to generalize better?

SUPPORT VECTOR MACHINES



Find a maximum margin separator, where the margin is defined by support vectors

SUPPORT VECTOR MACHINES

- Through tedious derivations one can find that, to maximize the margin, it suffices to solve the quadratic program

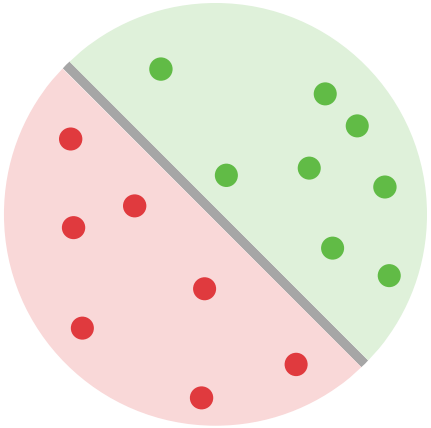
$$\max_{\boldsymbol{\alpha}} \sum_j \alpha_j - \frac{1}{2} \sum_{j,k} \alpha_j \alpha_k y^{(j)} y^{(k)} \left(\boldsymbol{x}^{(j)} \cdot \boldsymbol{x}^{(k)} \right)$$

$$\text{s.t.} \quad \sum_j y^{(j)} \alpha_j = 0$$

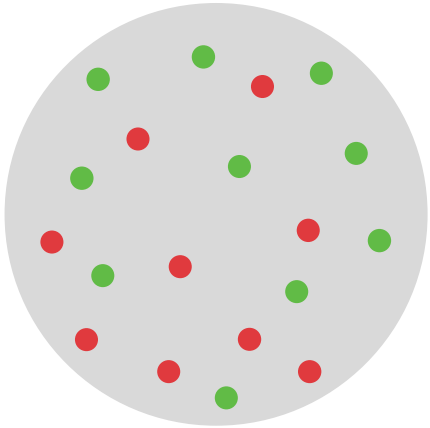
$$\forall j, \alpha_j \geq 0$$

- We can recover the separator from this program's solution
- It holds that $\alpha_j \neq 0$ only for support vectors $\boldsymbol{x}^{(j)}$

NON-SEPARABLE DATA



Linearly separable

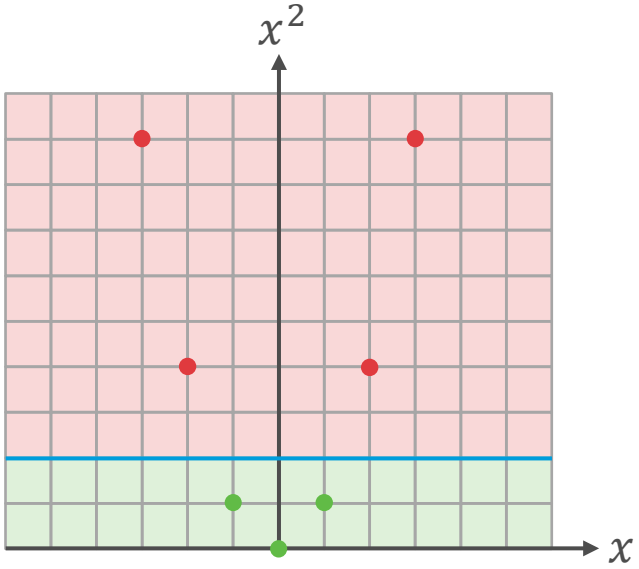


Not linearly separable

Poll 1: Would the Perceptron algorithm converge on non-separable data?

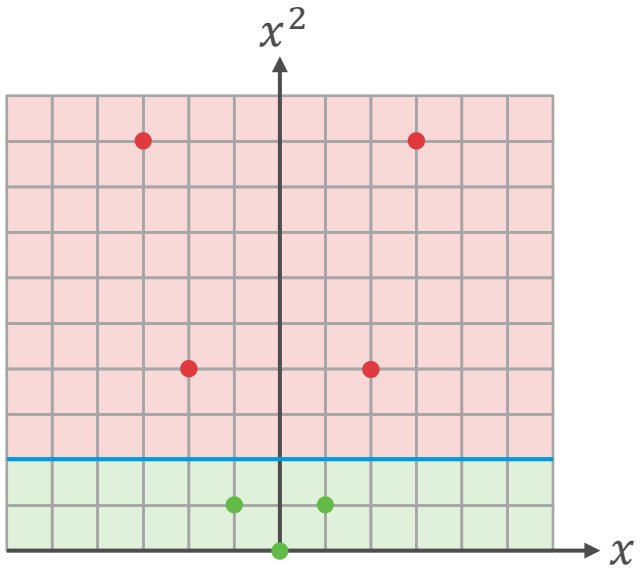
APPROACH 1: HIGHER DIMENSIONS

Not linearly separable



Linearly separable

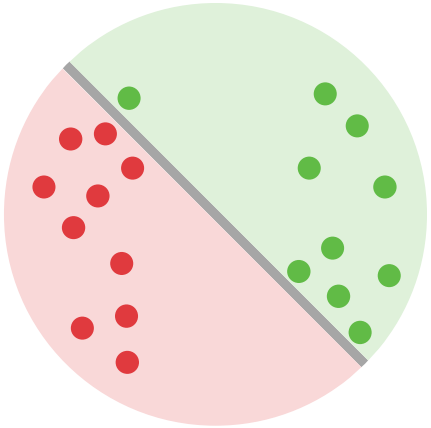
APPROACH 1: HIGHER DIMENSIONS



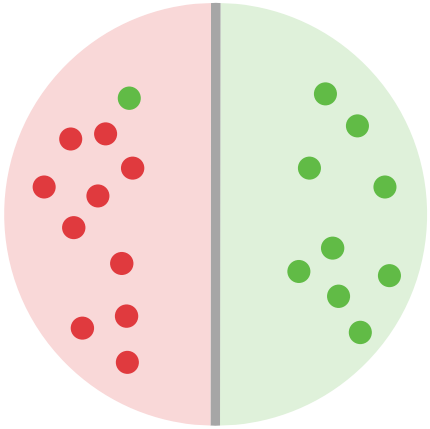
Poll 2: What is the set of positively labeled points on the line that the above separator corresponds to?

- $[0, \sqrt{2}]$
- $[\sqrt{2}, \infty]$
- $[-\sqrt{2}, \sqrt{2}]$
- $[-\infty, -\sqrt{2}] \cup [\sqrt{2}, \infty]$

APPROACH 2: SOFT MARGIN



No mistakes
small margins

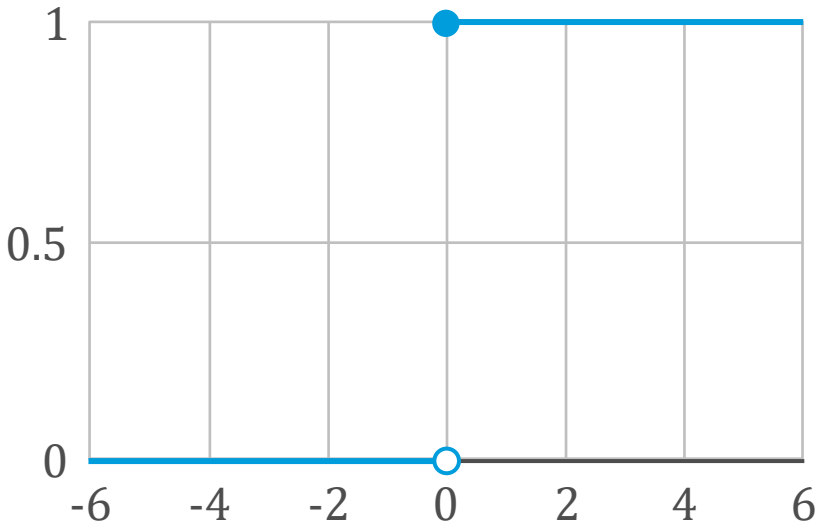


One mistake
Large margins

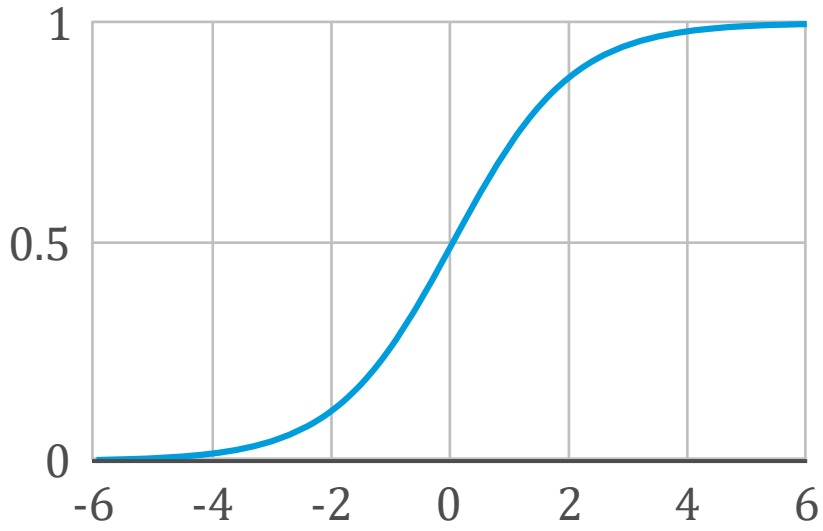
It's possible to enable soft margins by introducing slack variables into the SVM objective and constraints

LOGISTIC ACTIVATION FUNCTION

Another approach for enabling soft margins:



Threshold (step) function
(0/1 version)



Logistic (sigmoid) function
$f(z) = 1/(1 + e^{-z})$

LOGISTIC REGRESSION

- True labels are $y \in \{0,1\}$
- Denote the logistic function by

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

- A hypothesis is defined by

$$h_{\boldsymbol{w}}(\boldsymbol{x}) = \sigma(\boldsymbol{w} \cdot \boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{w} \cdot \boldsymbol{x}}}$$

- Interpreted as the probability that the label of $\boldsymbol{x}$ is 1

LOGISTIC REGRESSION

- Probability of observing $\{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{n}$ under $\boldsymbol{w}$ is

$$\prod_i \left[\sigma\left(\boldsymbol{w}\cdot\boldsymbol{x}^{(i)}\right)^{y^{(i)}} \cdot \left(1-\sigma\left(\boldsymbol{w}\cdot\boldsymbol{x}^{(i)}\right)\right)^{1-y^{(i)}}\right]$$

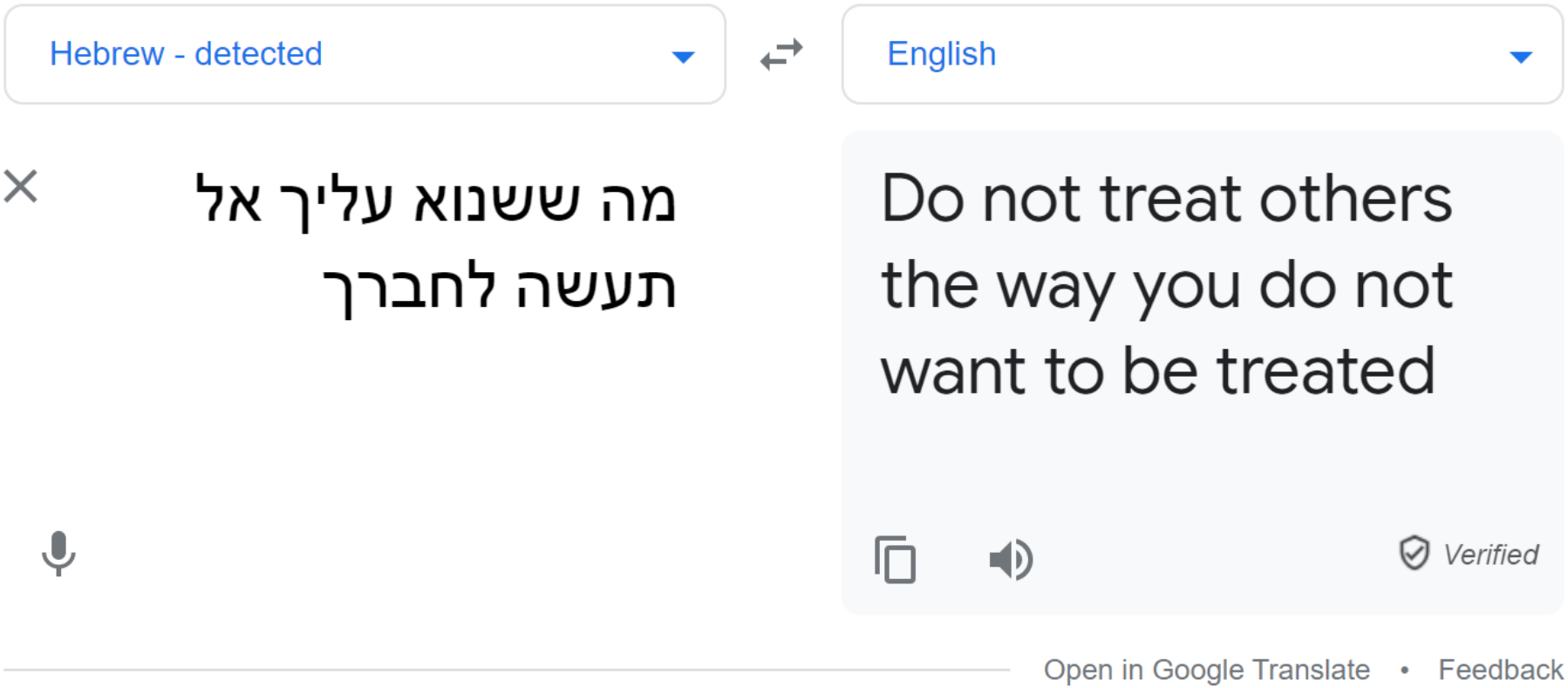
- The log-likelihood function is

$$LL(\boldsymbol{w}) = \log\left(\prod_i \left[\sigma\left(\boldsymbol{w}\cdot\boldsymbol{x}^{(i)}\right)^{y^{(i)}} \cdot \left(1-\sigma\left(\boldsymbol{w}\cdot\boldsymbol{x}^{(i)}\right)\right)^{1-y^{(i)}}\right]\right)$$

$$= \sum_i \left[y^{(i)} \log\left(\sigma\left(\boldsymbol{w}\cdot\boldsymbol{x}^{(i)}\right)\right) + \left(1-y^{(i)}\right)\log\left(1-\sigma\left(\boldsymbol{w}\cdot\boldsymbol{x}^{(i)}\right)\right)\right]$$

- Our goal is to maximize the concave function $LL(\boldsymbol{w})$

WORDS OF WISDOM



GRADIENT ASCENT, IN DETAIL

- If we take steps in the direction of the gradient, we eventually reach a local maximum
- Since our problem is convex, a local maximum is a global maximum
- The gradient ascent update step is

$$w_j^{(t+1)} = w_j^{(t)} + \alpha_t \frac{\partial LL\left(\boldsymbol{w}^{(t)}\right)}{\partial w_j^{(t)}}$$

where α_t is the learning rate

GRADIENT ASCENT, IN DETAIL

- One can verify that $\frac{\partial}{\partial z}\sigma(z) = \sigma(z)\big(1-\sigma(z)\big)$
- For a single example $(\boldsymbol{x}, y)$, we have

$$\frac{\partial LL(\boldsymbol{w})}{\partial w_j} = \frac{\partial}{\partial w_j} y \log \sigma(\boldsymbol{w}\cdot\boldsymbol{x}) + \frac{\partial}{\partial w_j}(1-y)\log\big(1-\sigma(\boldsymbol{w}\cdot\boldsymbol{x})\big)$$

$$= \left[\frac{y}{\sigma(\boldsymbol{w}\cdot\boldsymbol{x})} - \frac{1-y}{1-\sigma(\boldsymbol{w}\cdot\boldsymbol{x})}\right]\frac{\partial}{\partial w_j}\sigma(\boldsymbol{w}\cdot\boldsymbol{x})$$

$$= \left[\frac{y}{\sigma(\boldsymbol{w}\cdot\boldsymbol{x})} - \frac{1-y}{1-\sigma(\boldsymbol{w}\cdot\boldsymbol{x})}\right]\sigma(\boldsymbol{w}\cdot\boldsymbol{x})\big(1-\sigma(\boldsymbol{w}\cdot\boldsymbol{x})\big)x_j$$

$$= \left[\frac{y-\sigma(\boldsymbol{w}\cdot\boldsymbol{x})}{\sigma(\boldsymbol{w}\cdot\boldsymbol{x})\big(1-\sigma(\boldsymbol{w}\cdot\boldsymbol{x})\big)}\right]\sigma(\boldsymbol{w}\cdot\boldsymbol{x})\big(1-\sigma(\boldsymbol{w}\cdot\boldsymbol{x})\big)x_j$$

$$= [y-\sigma(\boldsymbol{w}\cdot\boldsymbol{x})]x_j$$

- We conclude that

$$\frac{\partial LL(\boldsymbol{w})}{\partial w_j} = \sum_{i=1}^{n}\big[y^{(i)} - \sigma\big(\boldsymbol{w}\cdot\boldsymbol{x}^{(i)}\big)\big]x_j^{(i)}$$