10-701 Introduction to Machine Learning (PhD)
Lecture 10: SVMs

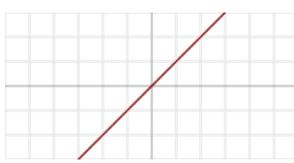
Leila Wehbe
Carnegie Mellon University
Machine Learning Department

Slides based on on Tom Mitchell's
10-701 Spring 2016 material
and Andrew Ng's lecture notes at:
http://cs229.stanford.edu/notes/cs229-notes3.pdf

Neural Networks

Choice of activation gate

Linear

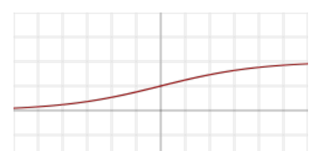


$f(x) = x$ $\qquad$ $f'(x) = 1$

Can be used to predict continuous values at output.
What happens when you stay linear layers?

Choice of activation gate

Sigmoid

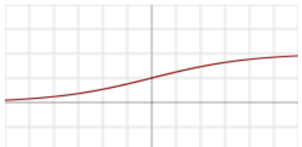


$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$ $\qquad$ $f'(x) = f(x)(1 - f(x))$

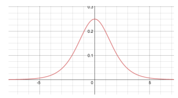
Outputs between 0 and 1, can be used for probability
Can saturate when very low or very high weights
Contributes to vanishing gradient

Choice of activation gate

Sigmoid



f'(x)

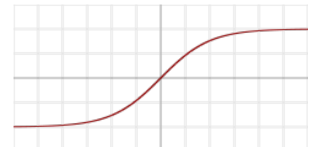


$$f(x) = \sigma(x) = \frac{1}{1+e^{-x}} \qquad f'(x) = f(x)(1 - f(x))$$

Outputs between 0 and 1, can be used for probability
Can saturate
Contributes to vanishing gradient

Choice of activation gate

tanh

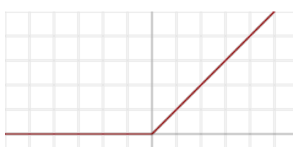


$$f(x) = \tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \qquad f'(x) = 1 - f(x)^2$$

Range -1 to 1

Choice of activation gate

Rectified Linear Unit (ReLu)

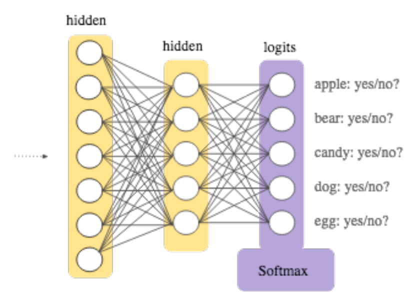


$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases} \qquad f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$$

Solve the vanishing gradient problem, but have a problem that nodes might die when negative value and never update. Can fix with leaky ReLu

Choice of activation gate

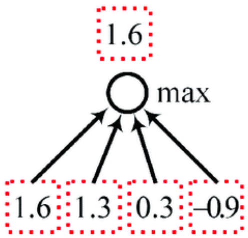
Softmax



$$f_i(\vec{x}) = \frac{e^{x_i}}{\sum_{j=1}^{J} e^{x_j}} \qquad \frac{\partial f_i(\vec{x})}{\partial x_j} = f_i(\vec{x})(\delta_{ij} - f_j(\vec{x}))$$

Choice of activation gate

Maxpool



$$f(\vec{x}) = \max_i x_i$$

$$\frac{\partial f}{\partial x_j} = \begin{cases} 1 & \text{for } j = \operatorname*{argmax}_i x_i \\ 0 & \text{for } j \neq \operatorname*{argmax}_i x_i \end{cases}$$

Choice of loss function

MSE

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2$$

L2

$$\mathcal{L} = \sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2$$

Choice of loss function

Binary cross-entropy

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}\left[y^{(i)}\log(\hat{y}^{(i)}) + (1 - y^{(i)})\log(1 - \hat{y}^{(i)})\right]$$

Multiclass:

$$-\frac{1}{n}\sum_{n}\sum_{k} y_{nk}\log f_k(\boldsymbol{x}_n)$$

Choice of loss function

Binary cross-entropy

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}\left[y^{(i)}\log(\hat{y}^{(i)}) + (1 - y^{(i)})\log(1 - \hat{y}^{(i)})\right]$$

Doesn’t have saturation problem

$$\mathcal{L} = y\log(\sigma(\mathbf{z})) + (1 - y)\log(1 - \sigma(\mathbf{z})),$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}} = (y - \sigma(\mathbf{z}))\cdot \mathbf{x}$$

Deep Networks

Deep networks: informal term for more recent generation of neural nets, with features such as:

- more hidden layers
- built from more heterogenous units
  - sigmoid, rectilinear, max pooling, LSTM, …
- shared weights across units (convolutional)
- with application-specific network architecture
  - time series, computer vision, speech recognition, ...
  - recurrent networks, max-pooled convolutional layers with local receptive fields...
- bi-directional units
- pretrained on unlabelled data (auto-encoders)
- ...

Impact of Deep Learning

- Speech Recognition
- Computer Vision
- Recommender Systems
- Language Understanding
- Drug Discovery and Medical Image Analysis

Microsoft Google IBM eBay NETFLIX MERCK NOVARTIS

[Courtesy of R. Salakhutdinov]

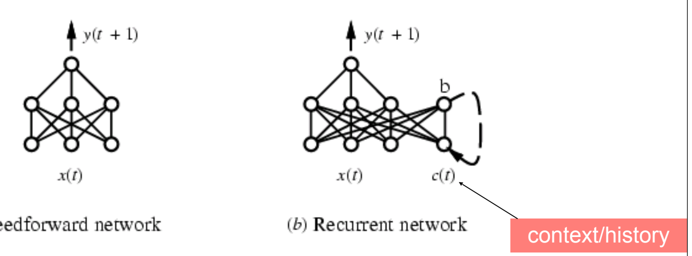
Training Networks on Time Series

- Suppose we want to predict next state of world
  - and it depends on history of unknown length
  - e.g., robot with forward-facing sensors trying to predict next sensor reading as it moves and turns
  - e.g., anticipate the next word in the sentence

Recurrent Networks: Time Series

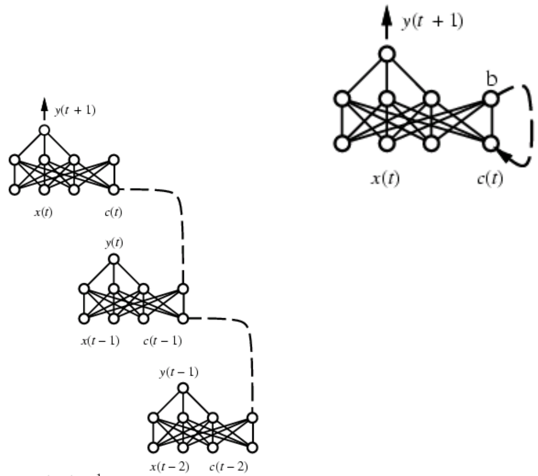
- Suppose we want to predict next state of world
  - and it depends on history of unknown length
  - e.g., robot with forward-facing sensors trying to predict next sensor reading as it moves and turns

- Idea: use hidden layer in network to capture/ remember state history



(a) Feedforward network (b) Recurrent network

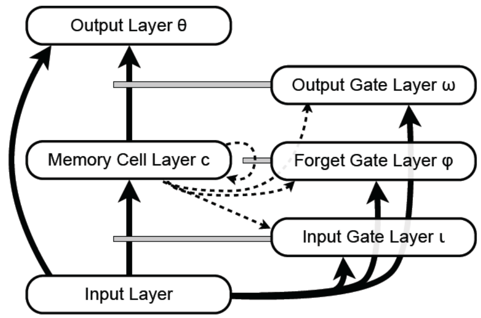
Recurrent Networks on Time Series



(c) Recurrent network unfolded in time

Long-Short Term Memory (LSTM) Units

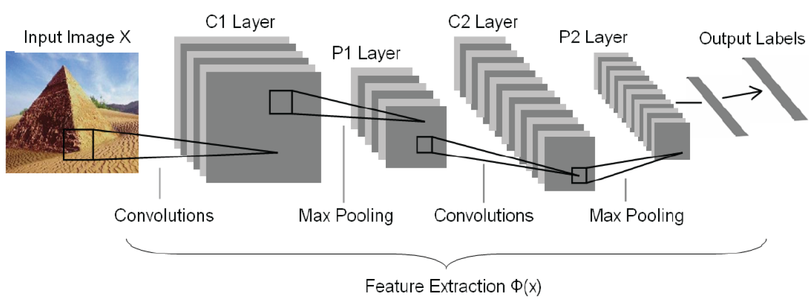
- Threshold unit/subnetwork with memory
  - still trainable with gradient descent



LSTM-g unit from [Monner & Regia, 2013]

Convolutional Neural Nets for Image Recognition

[Le Cun, 1992]



- specialized architecture: mix different types of units, not completely connected, motivated by primate visual cortex
- many shared parameters, stochastic gradient training
- very successful! now many specialized architectures for vision, speech, translation, …

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

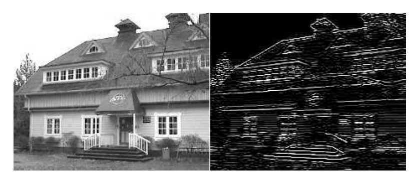
Simple box blur

0	0	0	5	0	0	0
0	5	18	32	18	5	0
0	18	64	100	64	18	0
5	32	100	100	100	32	5
0	18	64	100	64	18	0
0	5	18	32	18	5	0
0	0	0	5	0	0	0

Gaussian blur

-1	-1	-1
2	2	2
-1	-1	-1

Horizontal lines



By Utkarsh Sinha from: http://aishack.in/tutorials/image-convolution-examples/

-1	-1	-1
2	2	2
-1	-1	-1

Horizontal lines

-1	2	-1
-1	2	-1
-1	2	-1

Vertical lines

-1	-1	2
-1	2	-1
2	-1	-1

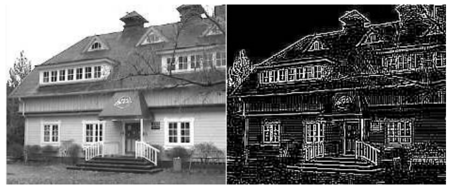
45 degree lines

2	-1	-1
-1	2	-1
-1	-1	2

135 degree lines

-1	-1	-1
-1	8	-1
-1	-1	-1

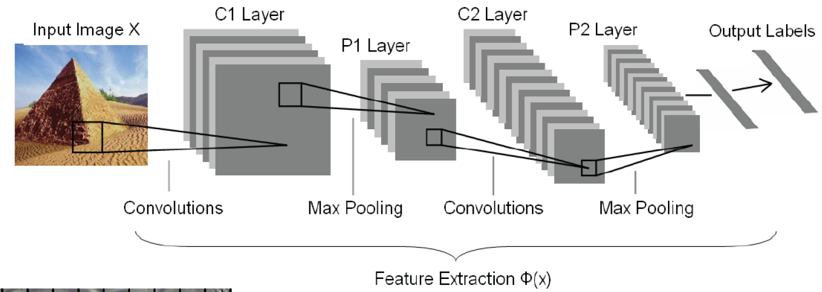
Edge detection

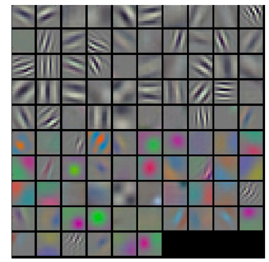


By Utkarsh Sinha from: http://aishack.in/tutorials/image-convolution-examples/

Filters learned from data

[Le Cun, 1992]





See http://cs231n.github.io/understanding-cnn/
And this article on distill.pub:
https://distill.pub/2018/building-blocks/

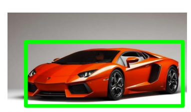
Feature Representations: Traditionally

Data → Feature extraction → Learning algorithm

Object detection

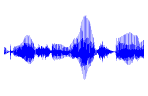
Image

vision features

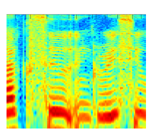


Recognition

Audio classification



Audio

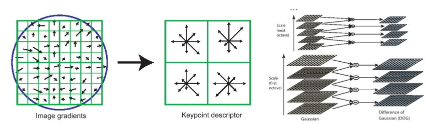


audio features

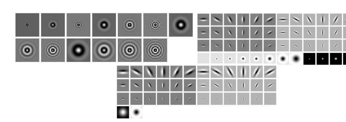
Speaker identification

[Courtesy of R. Salakhutdinov]

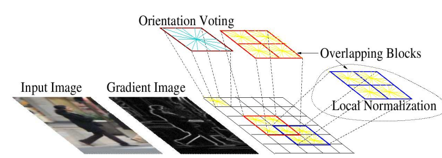
Computer Vision Features



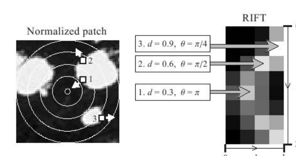
SIFT



Textons

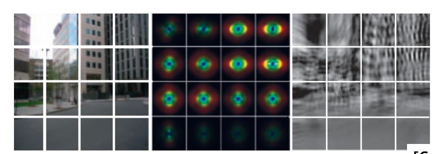


HoG



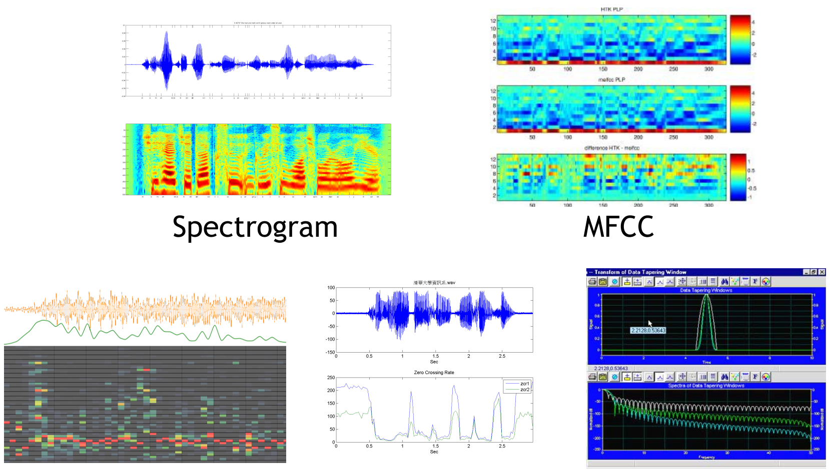
RIFT

GIST



[Courtesy, R. Salakhutdinov]

Audio Features

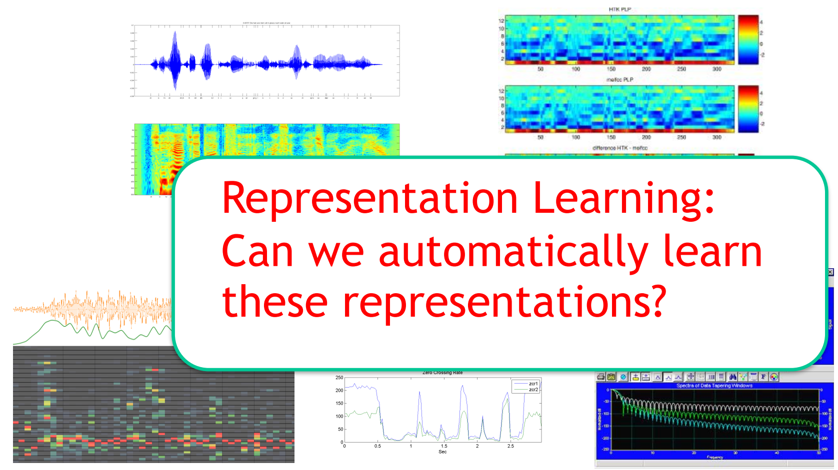


Spectrogram MFCC

Flux ZCR Rolloff

[Courtesy, R. Salakhutdinov]

Audio Features



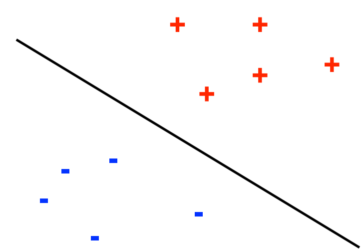
Representation Learning:
Can we automatically learn these representations?

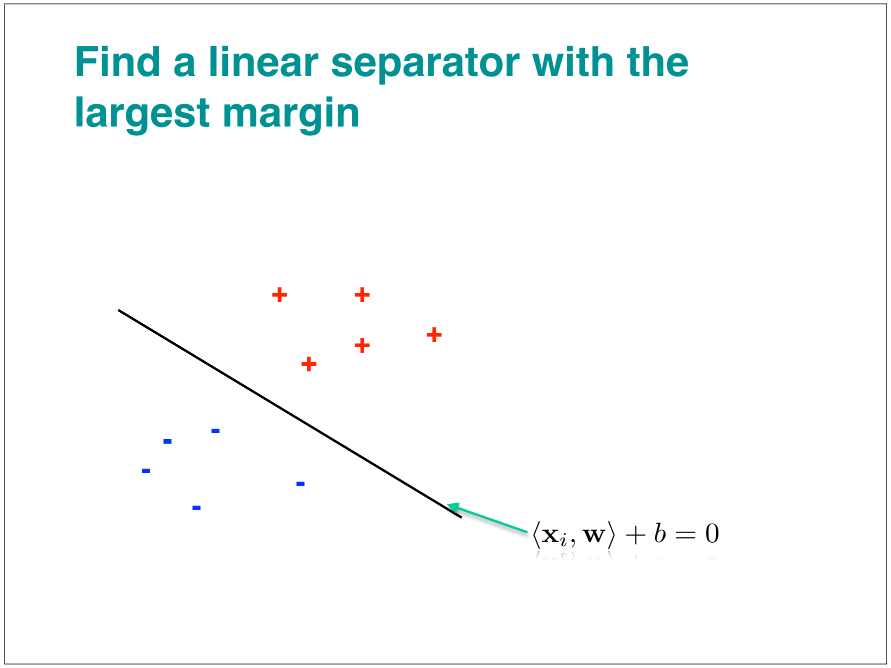
Flux ZCR Rolloff

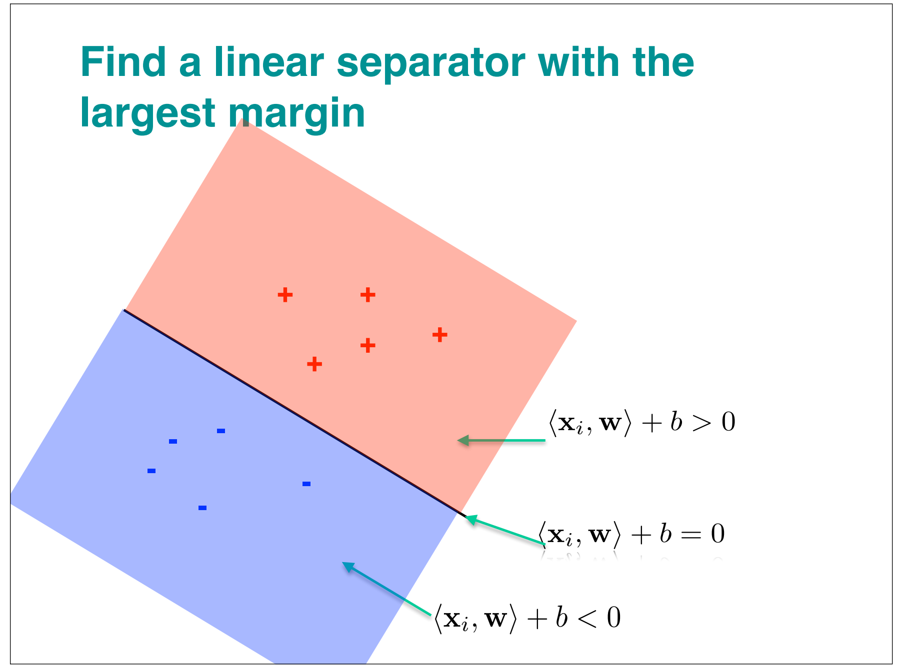
[Courtesy, R. Salakhutdinov]

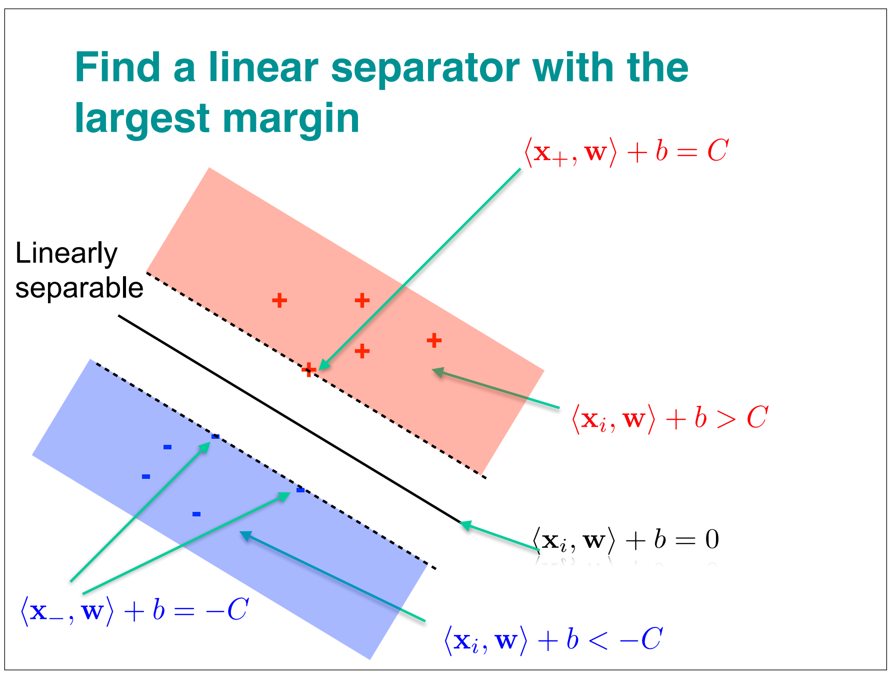
SVMs

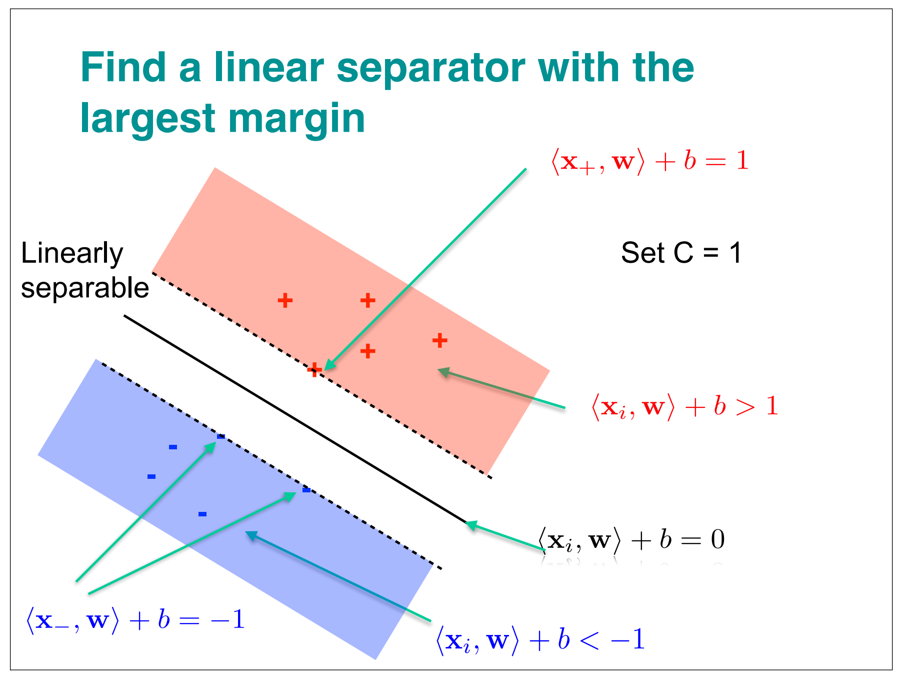
Find a linear separator with the largest margin



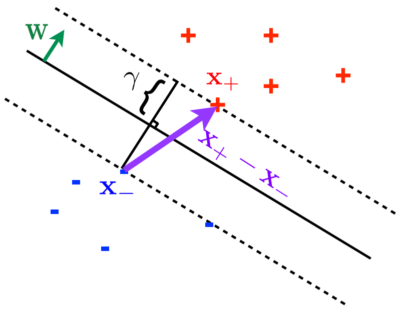

Find a linear separator with the largest margin
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b = 0$


Find a linear separator with the largest margin
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b > 0$
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b = 0$
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b < 0$


Find a linear separator with the largest margin
$\langle \mathbf{x}_+, \mathbf{w} \rangle + b = C$
Linearly separable
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b > C$
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b = 0$
$\langle \mathbf{x}_-, \mathbf{w} \rangle + b = -C$
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b < -C$


Find a linear separator with the largest margin
$\langle \mathbf{x}_+, \mathbf{w} \rangle + b = 1$
Set C = 1
Linearly separable
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b > 1$
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b = 0$
$\langle \mathbf{x}_-, \mathbf{w} \rangle + b = -1$
$\langle \mathbf{x}_i, \mathbf{w} \rangle + b < -1$

Find a linear separator with the largest margin

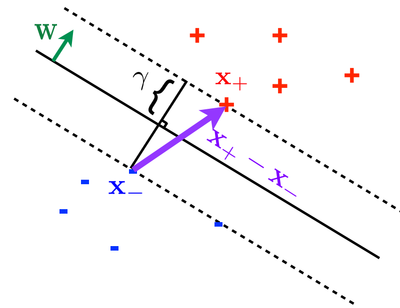


$$2\gamma = \frac{\langle \mathbf{x}_+ - \mathbf{x}_-, \mathbf{w}\rangle}{||\mathbf{w}||}$$

$$\gamma = \frac{1}{2}\frac{\langle \mathbf{x}_+ - \mathbf{x}_-, \mathbf{w}\rangle}{||\mathbf{w}||}$$

Find a linear separator with the largest margin

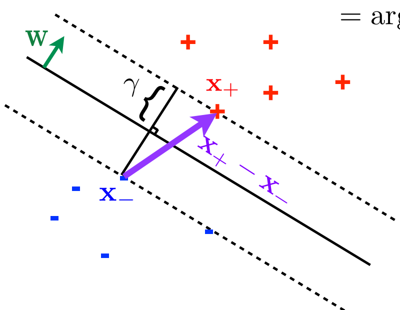
$$\arg\max_{\mathbf{w},b} \gamma = \arg\max_{\mathbf{w},b} \frac{1}{2}\frac{\langle \mathbf{x}_+ - \mathbf{x}_-, \mathbf{w}\rangle}{||\mathbf{w}||}$$



Find a linear separator with the largest margin

$$\arg\max_{\mathbf{w},b} \gamma = \arg\max_{\mathbf{w},b} \frac{1}{2}\frac{\langle \mathbf{x}_+ - \mathbf{x}_-, \mathbf{w}\rangle}{||\mathbf{w}||}$$

$$= \arg\max_{\mathbf{w},b} \frac{1}{2}\frac{\langle \mathbf{x}_+, \mathbf{w}\rangle - \langle \mathbf{x}_-, \mathbf{w}\rangle}{||\mathbf{w}||}$$



$$\langle \mathbf{x}_+, \mathbf{w}\rangle + b = 1$$

$$\langle \mathbf{x}_-, \mathbf{w}\rangle + b = -1$$

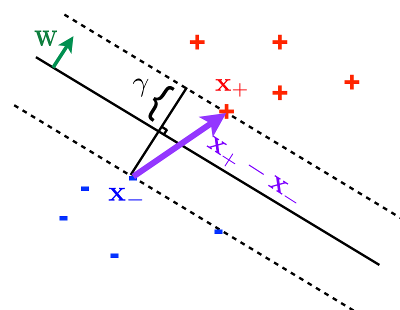
Find a linear separator with the largest margin

$$\arg\max_{\mathbf{w},b} \gamma = \arg\max_{\mathbf{w},b} \frac{1}{2}\frac{1 - b - (-1 - b)}{||\mathbf{w}||}$$

$$= \arg\max_{\mathbf{w},b} \frac{1}{2}\frac{2}{||\mathbf{w}||} = \arg\max_{\mathbf{w},b} \frac{1}{||\mathbf{w}||}$$

$$= \arg\min_{\mathbf{w},b} ||\mathbf{w}||$$

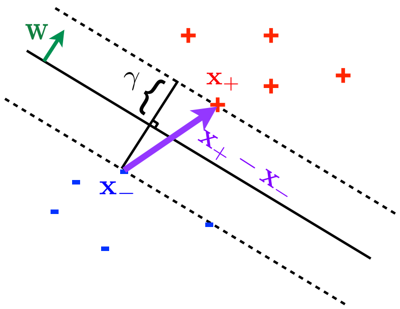
$$= \arg\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2$$



The (primal) optimization problem is:

$$\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2$$

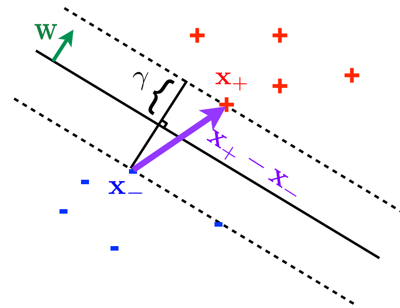
$$\text{s.t.} \quad y_i(\langle \mathbf{x}_i, \mathbf{w} \rangle + b) \geq 1, \quad i = 1, ..., m$$



The (primal) optimization problem is:

$$\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2$$

$$\text{s.t.} \quad y_i(\langle \mathbf{x}_i, \mathbf{w} \rangle + b) \geq 1, \quad i = 1, ..., m$$



This can be written as:

$$\min_{\mathbf{w},b} \; f(\mathbf{w}, b)$$

$$\text{s.t.} \quad g(\mathbf{w}, b) \leq 0, \quad i = 1, ..., m$$

Where $f(\mathbf{w}, b) = ||\mathbf{w}||^2$
and $g(\mathbf{w}, b) = 1 - y_i(\langle \mathbf{x}_i, \mathbf{w} \rangle + b)$
are both convex functions of our parameters **w** and b

Lagrangian

We can write the Lagrangian of:

$$\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2$$

$$\text{s.t.} \quad y_i(\langle \mathbf{x}_i, \mathbf{w} \rangle + b) \geq 1, \quad i = 1, ..., m$$

as:

$$\mathcal{L}(\mathbf{w}, b, \alpha) = f(\mathbf{w}, b) + \sum_{i=1}^{m} \alpha_i g_i(\mathbf{w}, b)$$

$$= \frac{1}{2}||\mathbf{w}||^2 + \sum_{i} \alpha_i \left(1 - y_i(\langle \mathbf{x_i}, \mathbf{w} \rangle + b)\right)$$

$$g_i(\mathbf{w}, b) = 1 - y_i(\langle \mathbf{x}_i, \mathbf{w} \rangle + b)$$