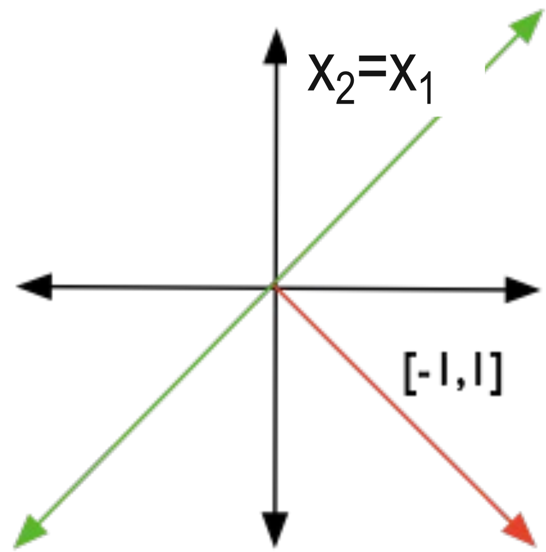
Support Vector Machines (SVMs)

Lyle Ungar

Learning objectives
Review decision boundaries
Hinge loss
SVM as large margin method
SVM – primal and dual

Representing Lines

- How do we represent a line?
  - $x_2 = x_1$
  - $0 = x_1 - x_2$
  - $0 = [1,-1]^T[x_1, x_2]$



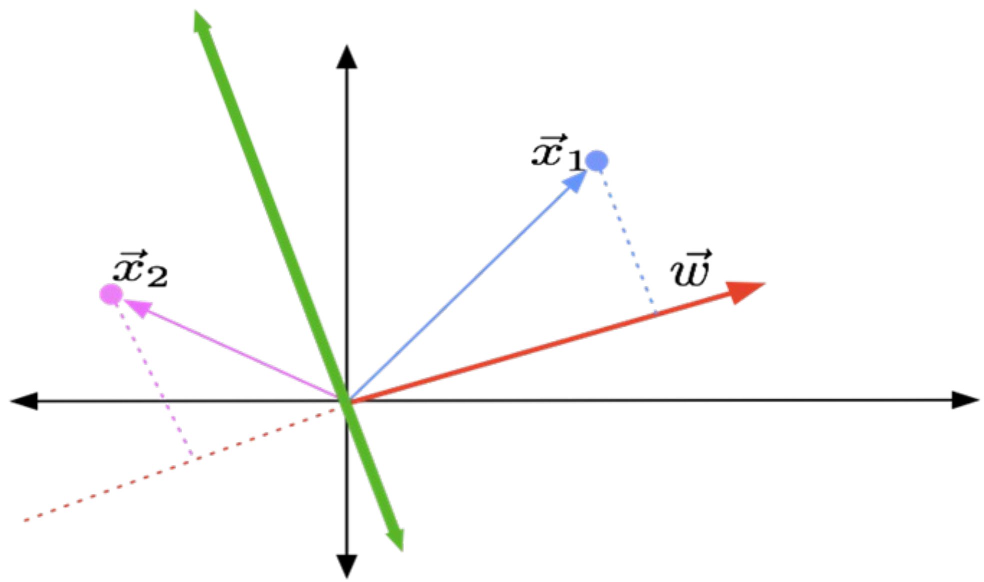
- In general, a hyperplane is defined by

$$0 = w^T x$$

The red vector (w) defines the green plane that is orthogonal to it.

Why bother with this weird representation?

Projections



$(\vec{w} \cdot \vec{x})\vec{w}$ is the projection of $\vec{x}$ onto $\vec{w}$

alternate intuition: the dot product of two vectors is simply the product of their lengths and the cosine of the angle between them

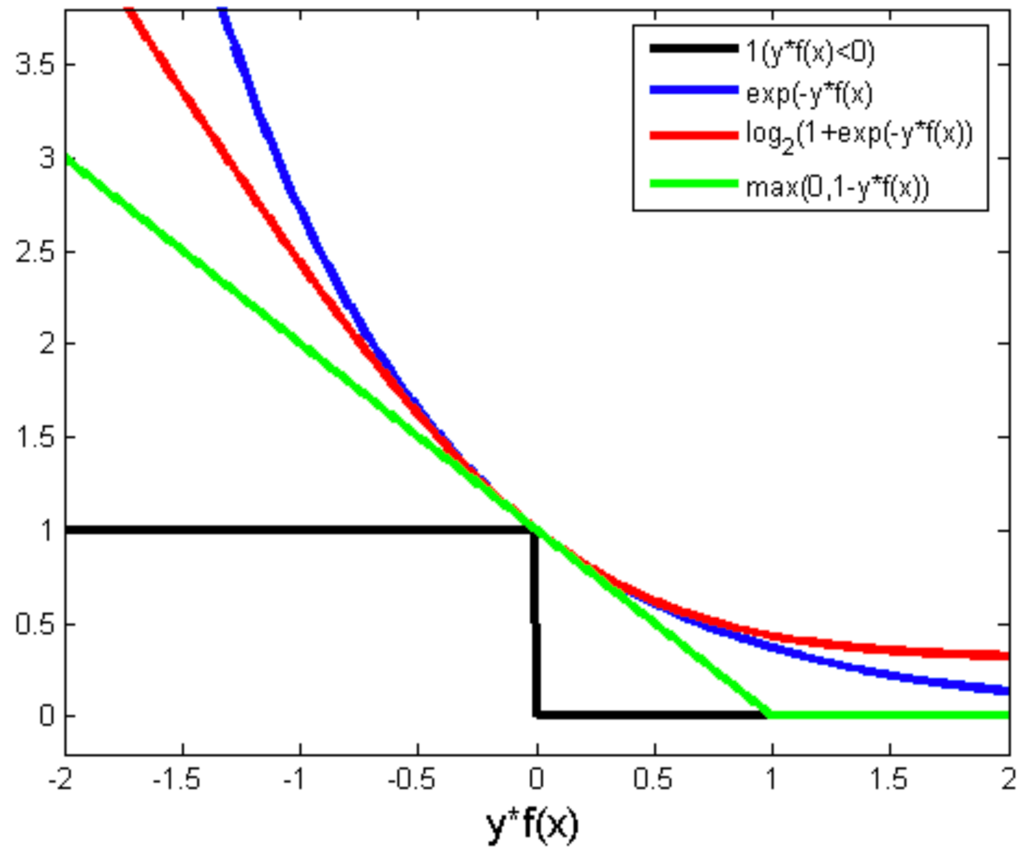
Now classification is easy!

- **Input: x**
- **Model: w**
- **Score:** $\mathbf{w}^T\mathbf{x}$
- **Prediction:** $\mathrm{sgn}(\mathbf{w}^T\mathbf{x})$
- But how do we learn ***w***?

Support Vector Machines (SVMs)

- **Minimize hinge loss**
  - With regularization
- **”Large margin” methods**

Loss functions for classification



Hinge loss

SVM: Hinge loss, Ridge penalty

$h(\boldsymbol{x}) = sign(\boldsymbol{w}^T\boldsymbol{x} + b)$

$$\min_{\mathbf{w},b,\xi\geq 0} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i$$

$$\xi_i = \max(0, 1 - y_i(\mathbf{w}^\top\mathbf{x}_i + b))$$

0 if score is correct by 1 or more
(hinge loss)

Support Vector Machines

Andrew W. Moore

School of Computer Science

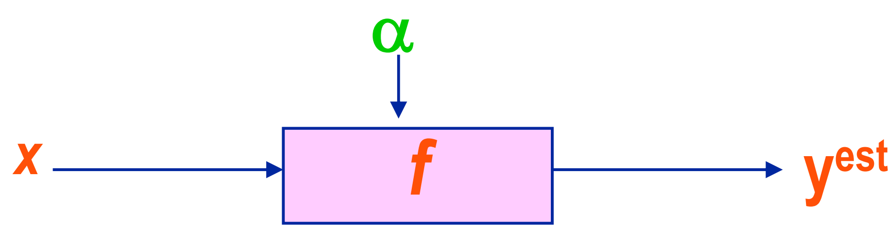
Carnegie Mellon University

www.cs.cmu.edu/~awm

awm@cs.cmu.edu

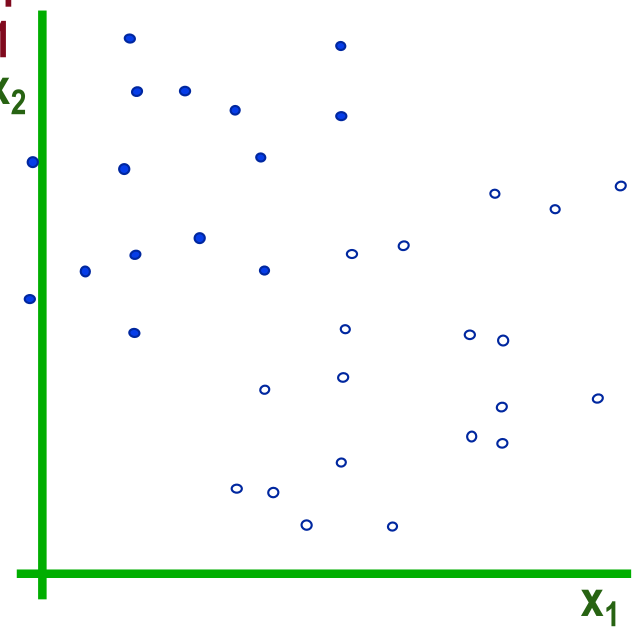
With modifications by Lyle Ungar

Linear Classifiers



$f(x,w,b) = sign(w.\ x - b)$

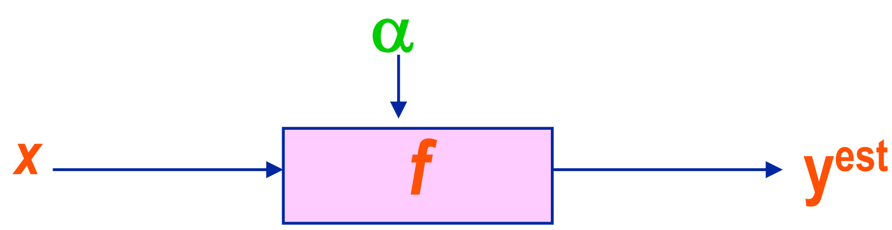
- denotes +1
- denotes -1



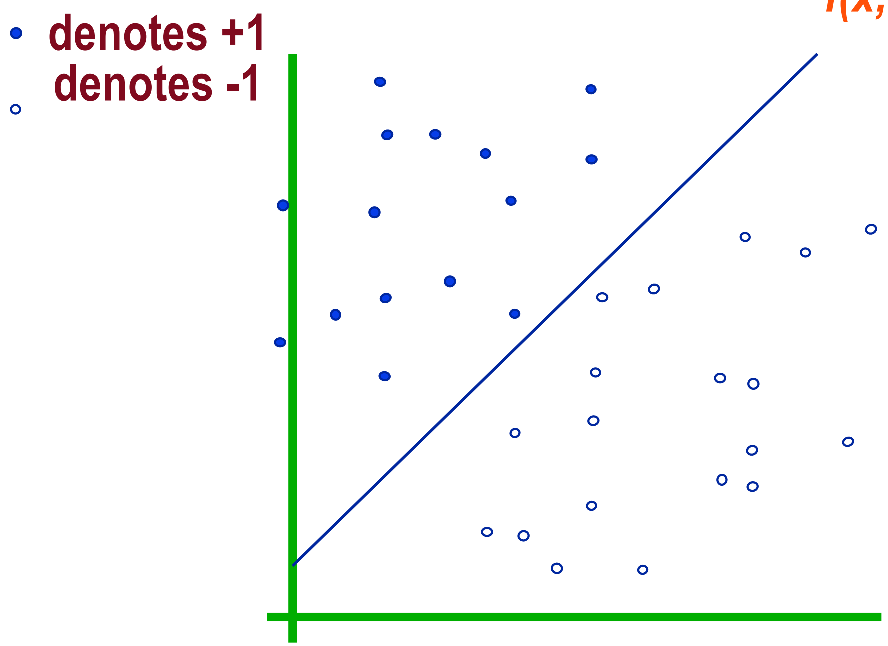
How would you classify this data?

These points are ***linearly separable***

Linear Classifiers

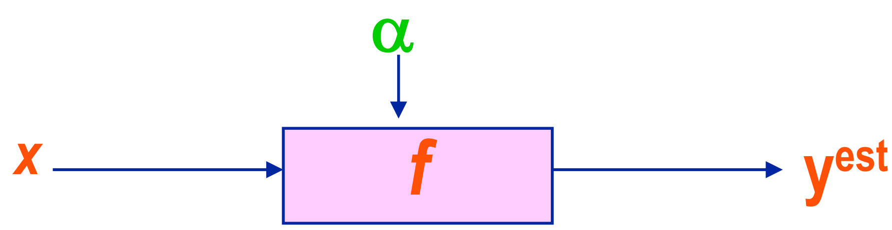


$f(x,w,b) = sign(w. x - b)$

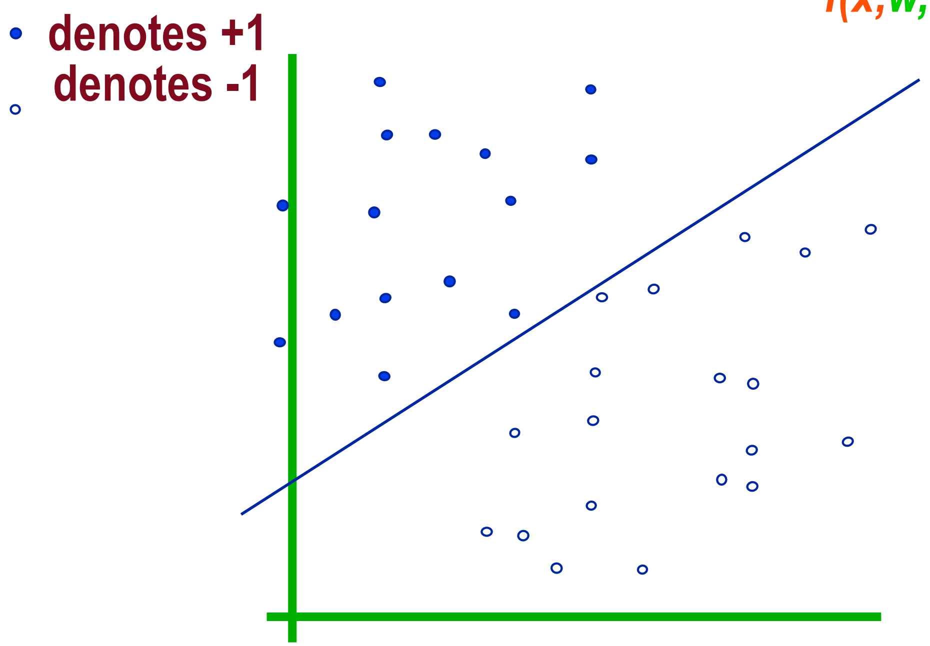


How would you classify this data?

Linear Classifiers

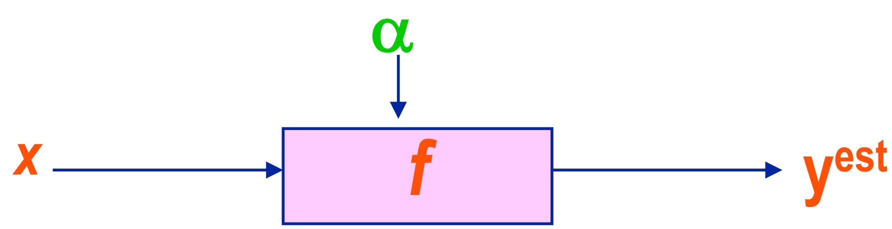


$f(x,w,b) = sign(w. x - b)$

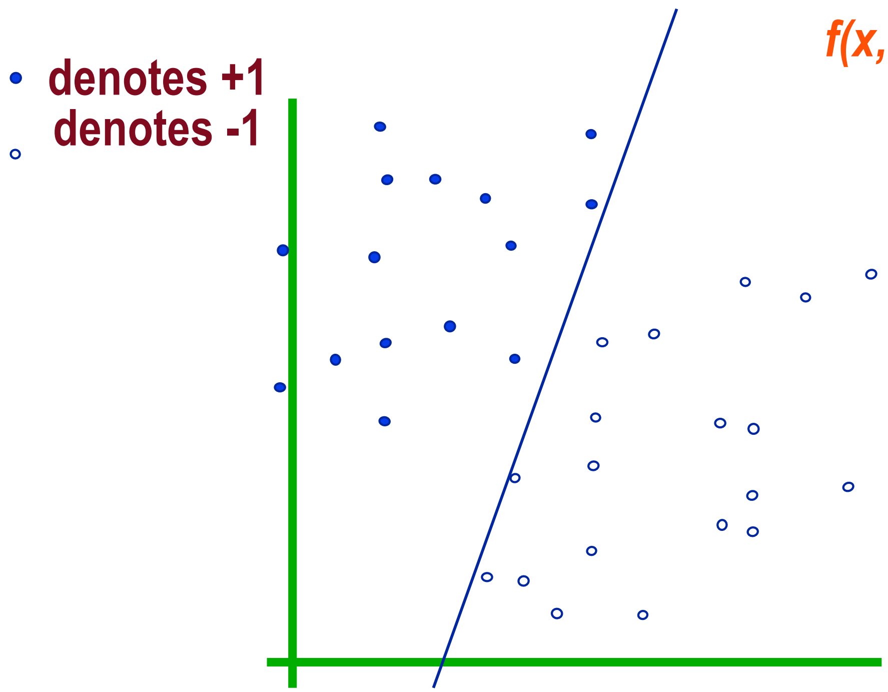


How would you classify this data?

Linear Classifiers

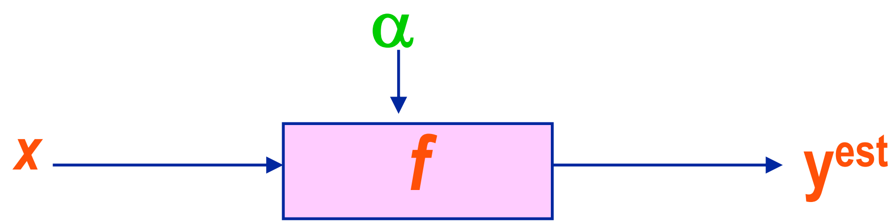


$f(x,w,b) = sign(w. x - b)$

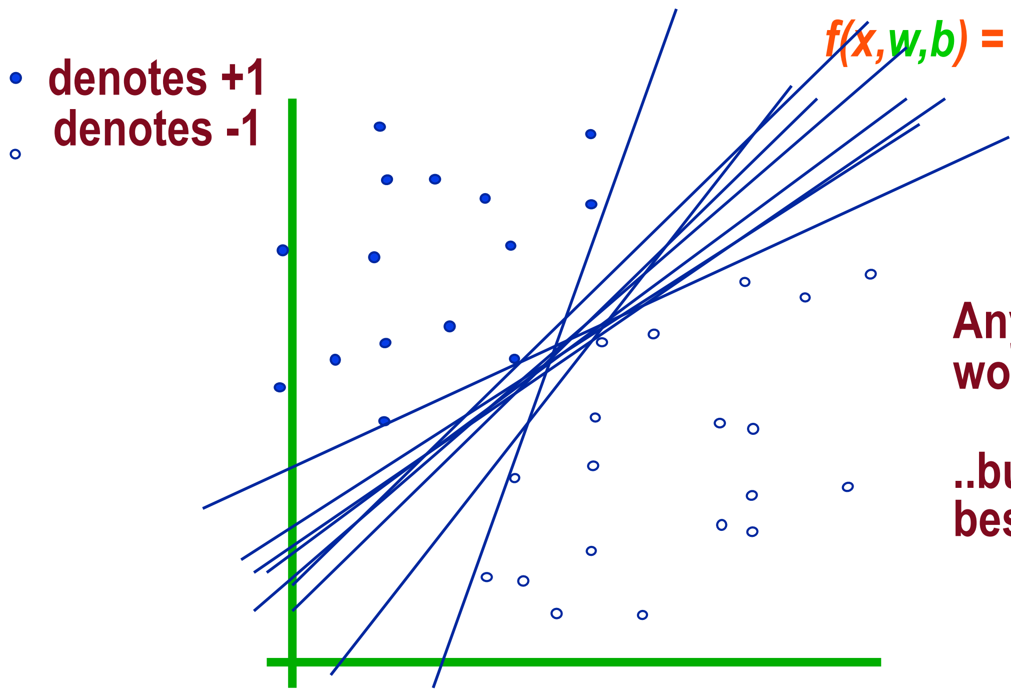


How would you classify this data?

Linear Classifiers



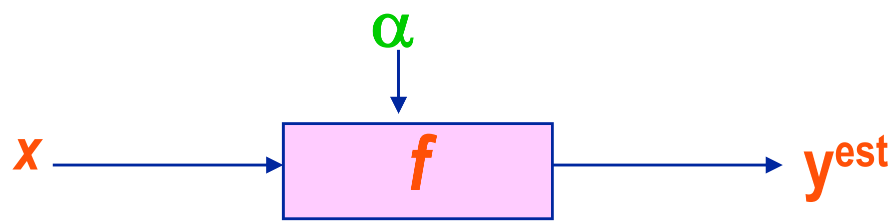
$f(x,w,b) = sign(w.\ x - b)$



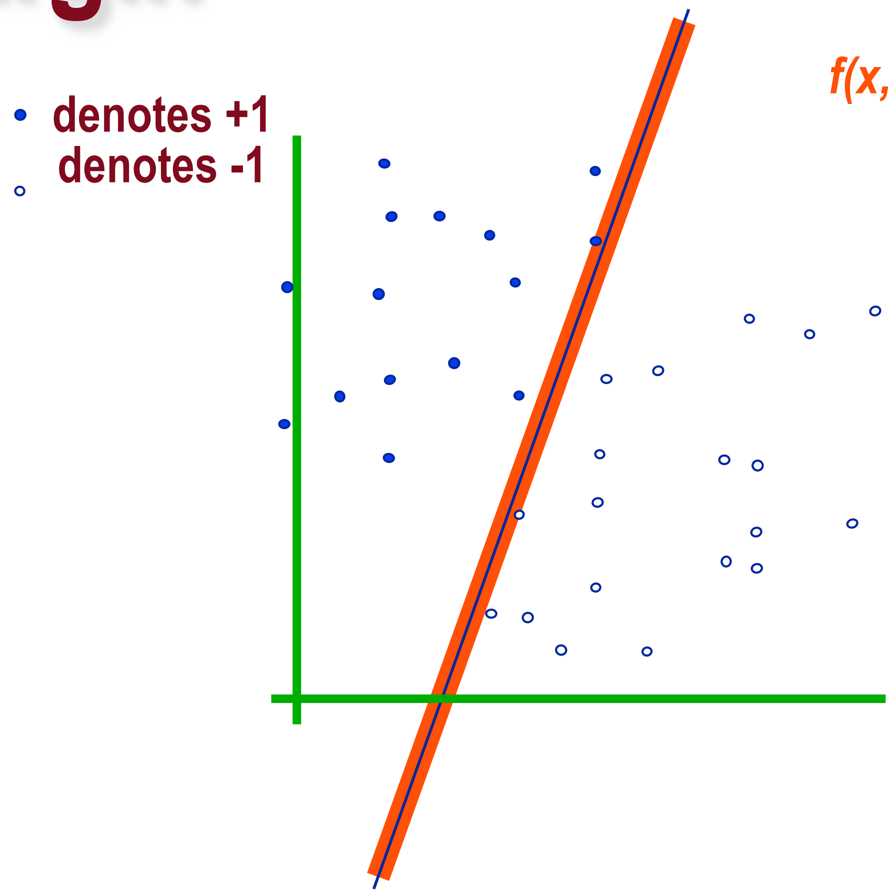
Any of these would be fine..

..but which is best?

Classifier Margin

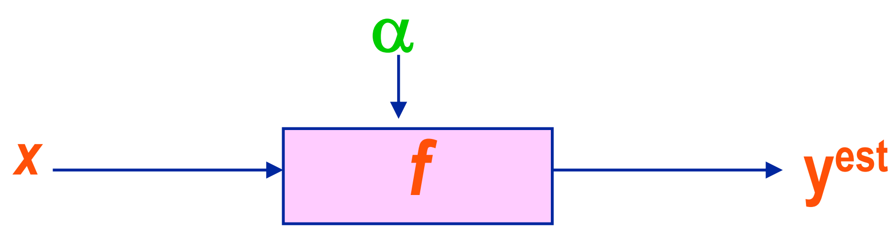


$f(x,w,b) = sign(w. x - b)$

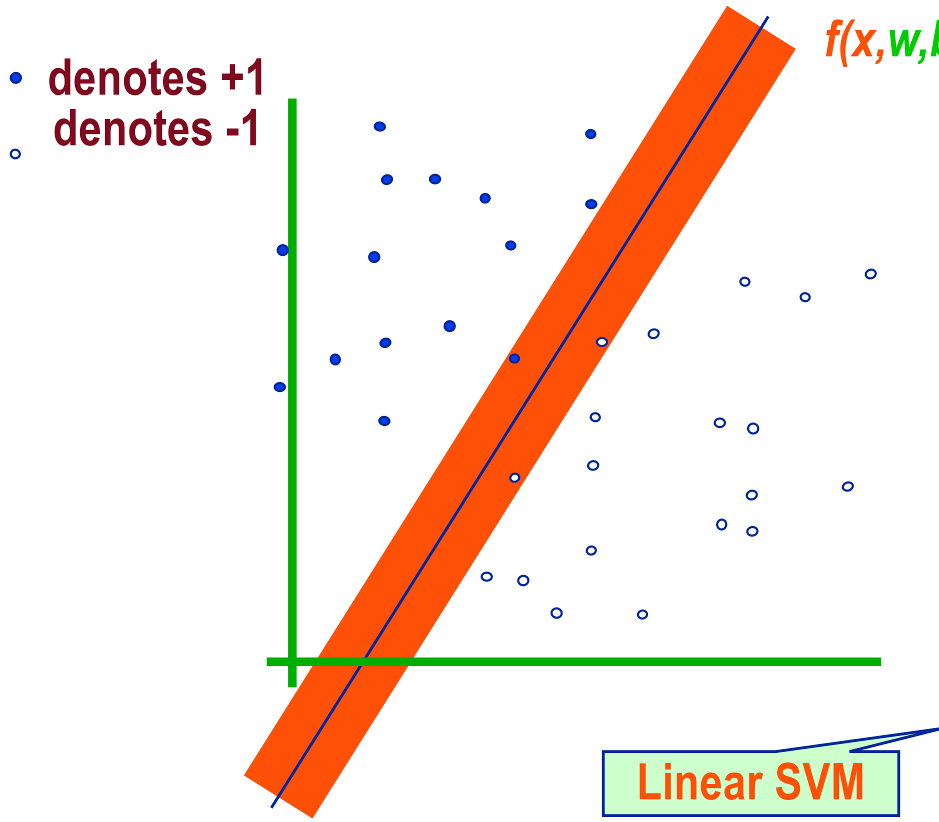


Define the **margin** of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.

Maximum Margin

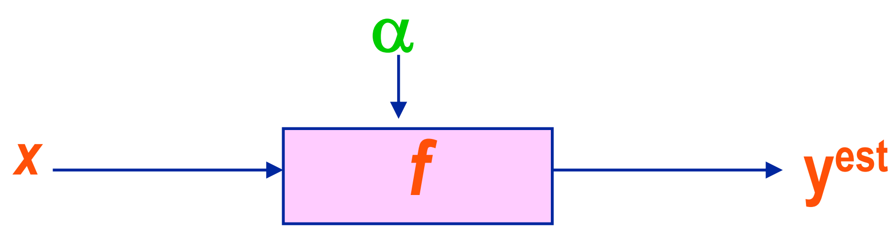


$f(x,w,b) = sign(w. x - b)$

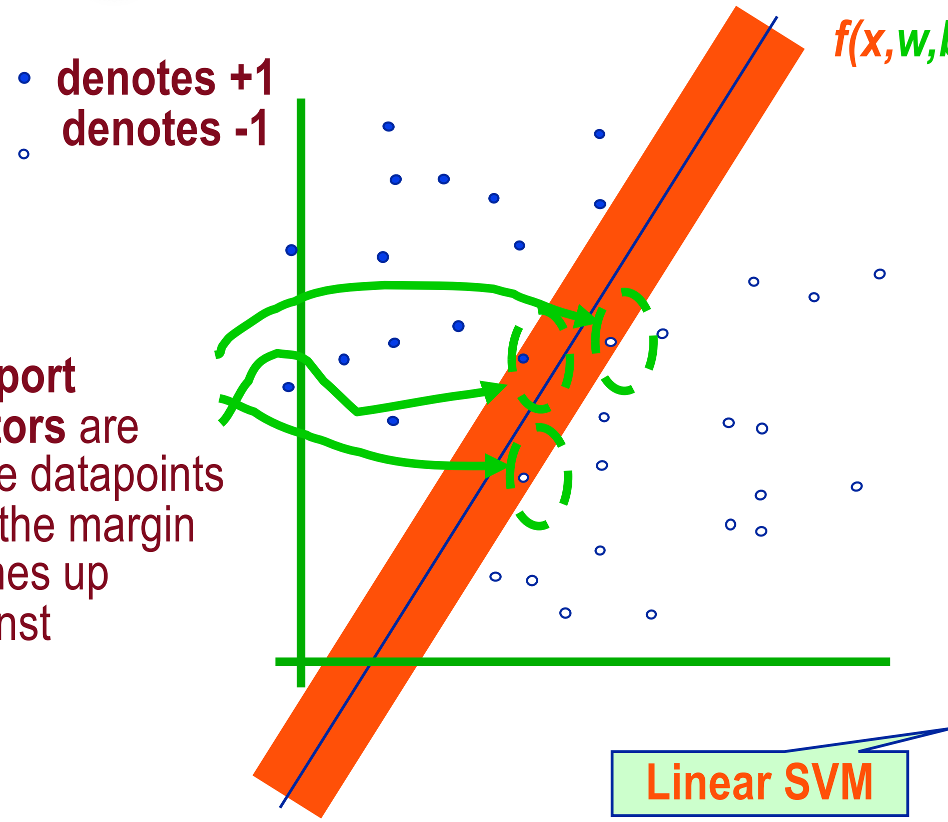


The **maximum margin** linear classifier is the linear classifier with the, um, maximum margin. This is the simplest kind of SVM

Maximum Margin

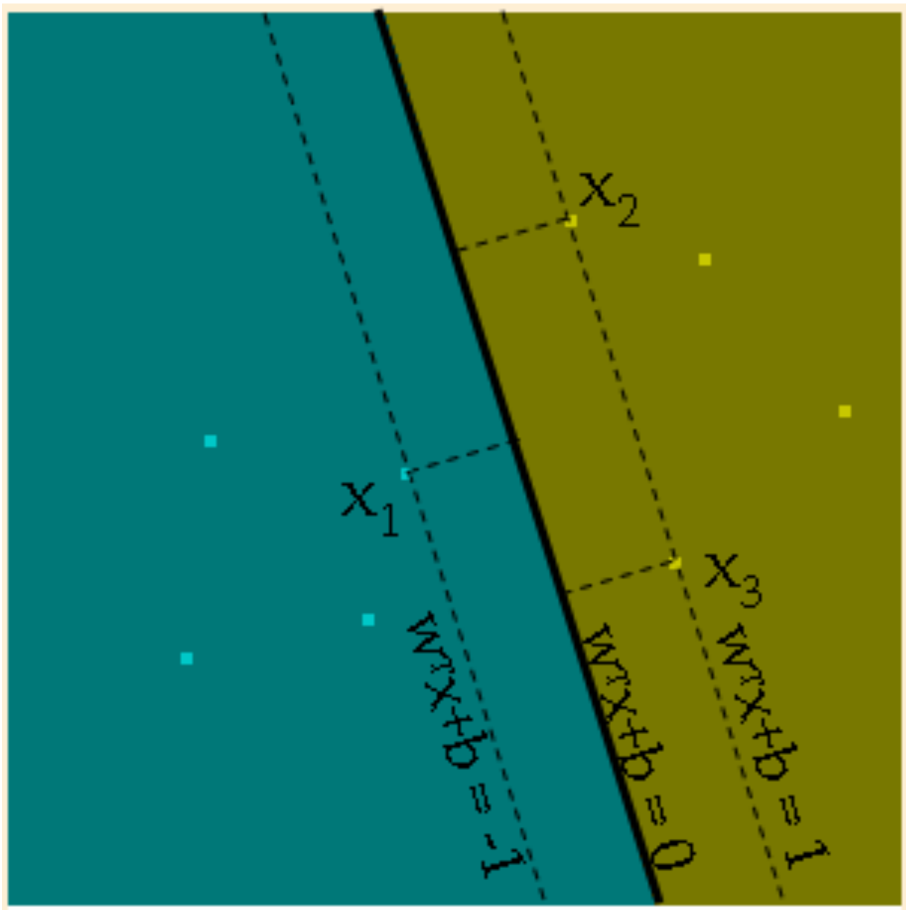


$f(x,w,b) = sign(w.\ x - b)$



Support Vectors are those datapoints that the margin pushes up against

The **maximum margin** linear classifier is the linear classifier with the, um, maximum margin. This is the simplest kind of SVM



$$h(\mathbf{x}) = sign(\mathbf{w}^\top \mathbf{x} + b)$$

Arbitrarily normalize

$$\mathbf{w}^\top \mathbf{x} + b = \pm 1$$

x_1, x_2 and x_3 are support vectors

Compute margin

$$\mathbf{w}^\top \mathbf{x}_1 + b = -1 \quad \text{and} \quad \mathbf{w}^\top \mathbf{x}_2 + b = 1$$

Maximize margin

$$\mathbf{w}^\top (\mathbf{x}_2 - \mathbf{x}_1) = 2 \quad \rightarrow \quad \frac{\mathbf{w}^\top}{2||\mathbf{w}||_2}(\mathbf{x}_2 - \mathbf{x}_1) = \frac{1}{||\mathbf{w}||_2}$$

Max margin interp. of SVM

Separable SVM primal:

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

$$\text{s.t.} \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad i = 1, \ldots, n$$

Note: *min* $w^T w$ maximizes the margin

SVM: Hinge loss, ridge penalty

$h(\boldsymbol{x}) = sign(\boldsymbol{w}^T\boldsymbol{x} + b)$

$$\min_{\mathbf{w},b,\xi\geq 0} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i$$

$$\xi_i = \max(0, 1 - y_i(\mathbf{w}^\top\mathbf{x}_i + b))$$

0 if score is correct by 1 or more
(hinge loss)

Primal and Dual

- **Primal**: the feature space
  - $\boldsymbol{X}^T\mathbf{X}$ is $p*p$
- **Dual:** the observation space
  - $\boldsymbol{XX}^T$ is $n*n$
- Note that $\boldsymbol{XX}^T$ is a *kernel matrix*

Use Lagrange Multiplier magic

Separable SVM dual:

$$\max_{\alpha \geq 0} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j$$

$$\text{s.t.} \quad \sum_i \alpha_i y_i = 0$$

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$

Most constraints are non-binding so most α_i are zero.

The $\boldsymbol{x}_i$ with nonzero α_i are support vectors.

Intuition of Non-binding

- **Points that are on the correct side of the margin don't contribute the hinge loss function**
  - So the loss is entirely in terms of the points on that are inside the margin or on the wrong side of it.

Kernelized separable dual:

$$\max_{\alpha \geq 0} \quad \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$$

$$\text{s.t.} \quad \sum_i \alpha_i y_i = 0$$

$$\mathbf{w}^\top \mathbf{x} + b = \textstyle\sum_i \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b$$

Max margin interp. of SVM

Separable SVM primal:

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

$$\text{s.t.} \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad i = 1, \ldots, n$$

The non-separable case

Hinge primal:

$$\min_{\mathbf{w},b,\xi\geq 0} \quad \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i$$

$$\text{s.t.} \quad y_i(\mathbf{w}^\top\mathbf{x}_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n$$

$$\xi_i = \max(0, 1 - y_i(\mathbf{w}^\top\mathbf{x}_i + b))$$

"Slack variable" – hinge loss from the margin

Generalize it!

Hinge primal:

$$\min_{\mathbf{w},b,\xi\geq 0} \quad \frac{1}{2}\|\mathbf{w}\|_p^p + C\|\xi\|_q^q$$

$$\text{s.t.} \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n$$

The non-separable dual

Hinge dual:

$$\max_{\alpha \geq 0} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j$$

$$\text{s.t.} \quad \sum_i \alpha_i y_i = 0, \quad \alpha_i \leq C, \quad i = 1, \ldots, n$$

$x_i^T x_j$ is a kernel matrix

C controls regularization

What you should know

- **Hinge loss**
  - Slack variable
- **Margin**
- **Support vector**
- **Primal/dual**

Note: we did not cover Lagrange multipliers this year, you are not responsible for them!

Questions?

Top