CS722/822: Machine Learning

Instructor: Jiangwen Sun
Computer Science Department

- We just introduced linear discriminant analysis and logistic regression
- Now let us discuss Support Vector Machine

History of SVM

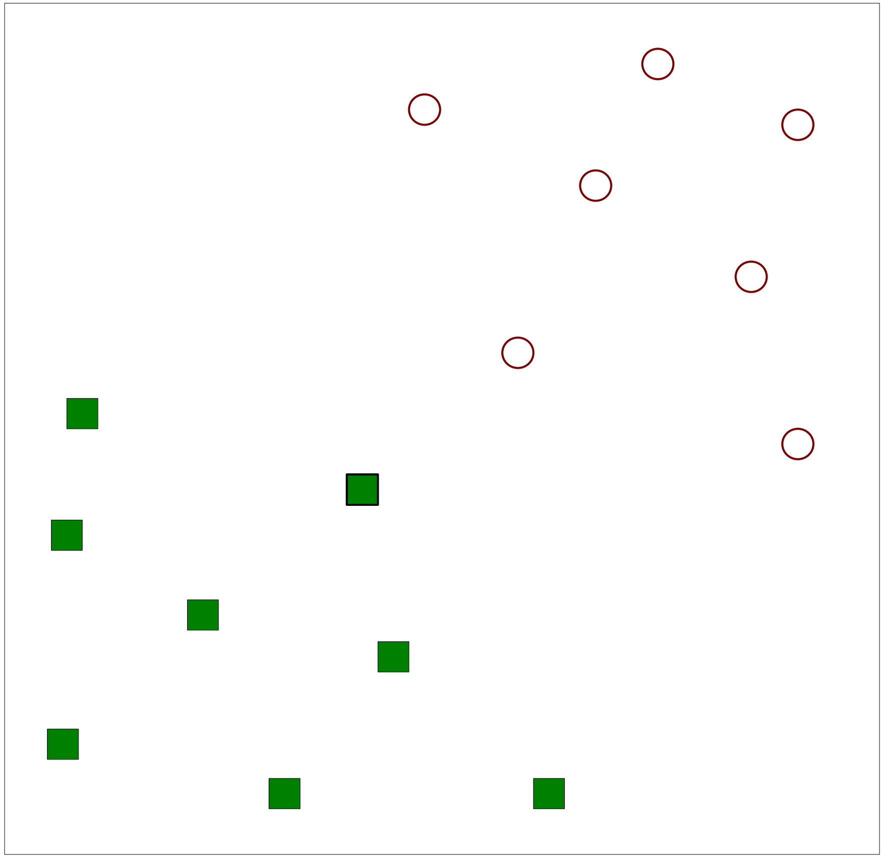
- SVM is inspired from statistical learning theory [3].
- SVM was first introduced in 1992 [1].
- SVM becomes popular because of its success in handwritten digit recognition [2].
- SVM is now regarded as an important example of "kernel methods", one of the important areas in machine learning. http://www.kernel-machines.org/

[1] B.E. Boser *et al*. A Training Algorithm for Optimal Margin Classifiers. Proceedings of the Fifth Annual Workshop on Computational Learning Theory 5 144-152, Pittsburgh, 1992.

[2] L. Bottou *et al*. Comparison of classifier methods: a case study in handwritten digit recognition. Proceedings of the 12th IAPR International Conference on Pattern Recognition, vol. 2, pp. 77-82. 1994.

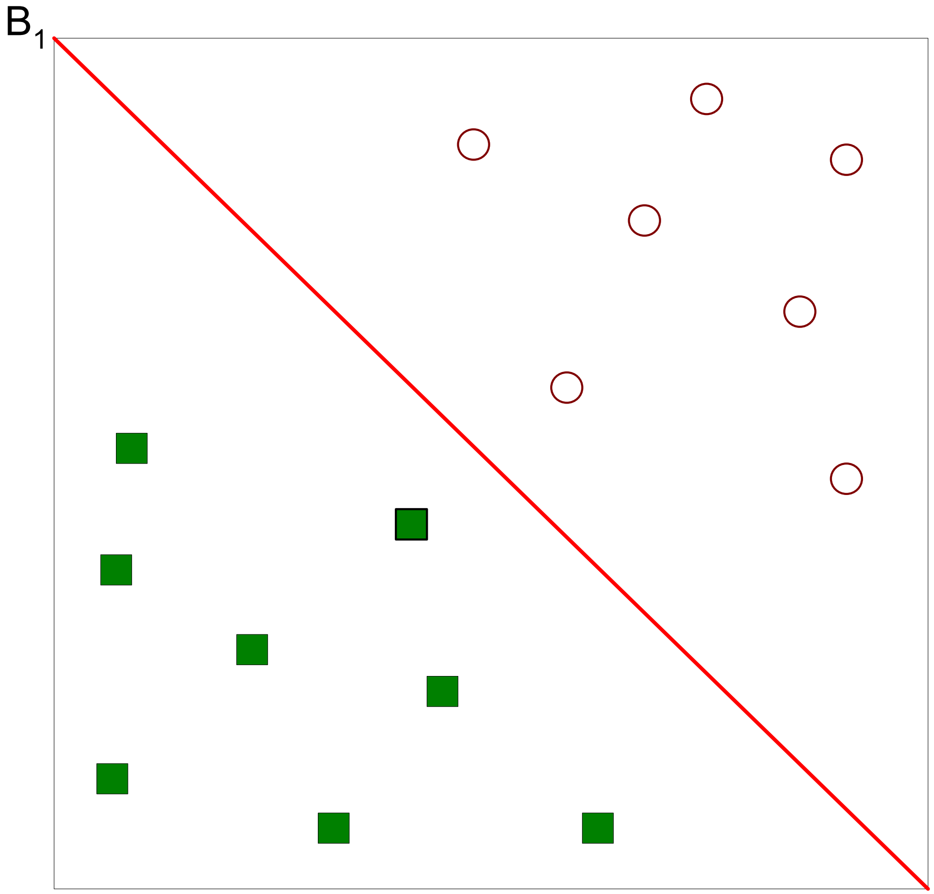
[3] V. Vapnik. The Nature of Statistical Learning Theory. 1nd edition, Springer, 1996.

Support Vector Machines



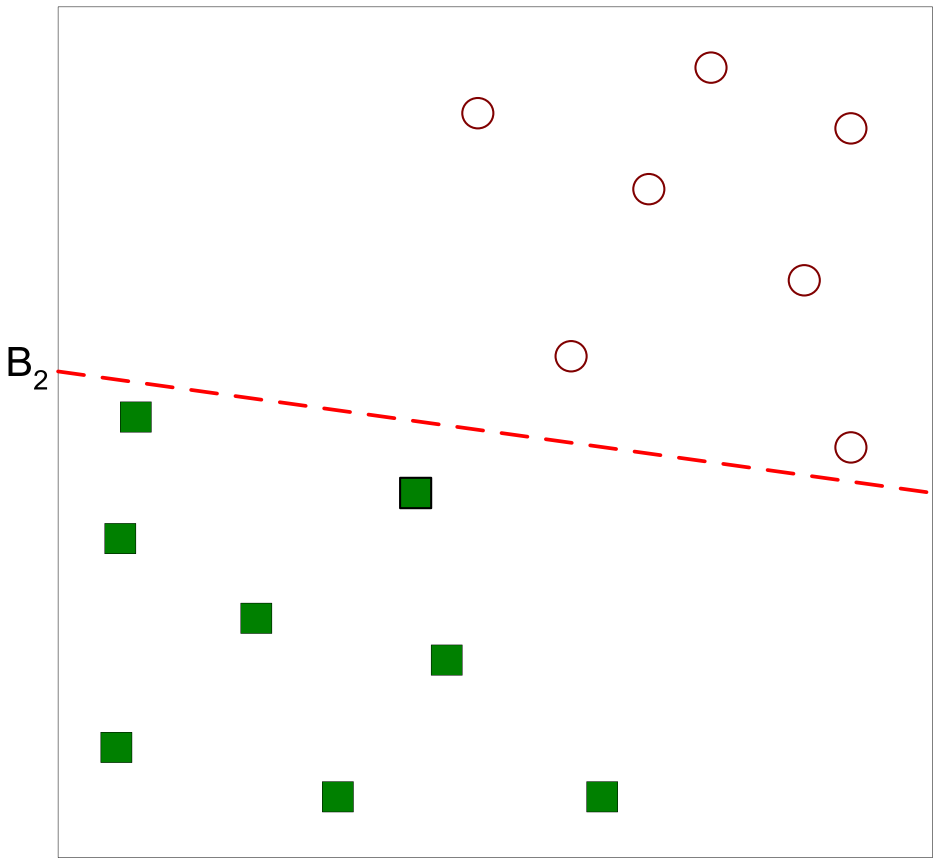
- Find a linear hyperplane (decision boundary) that will separate the data

Support Vector Machines



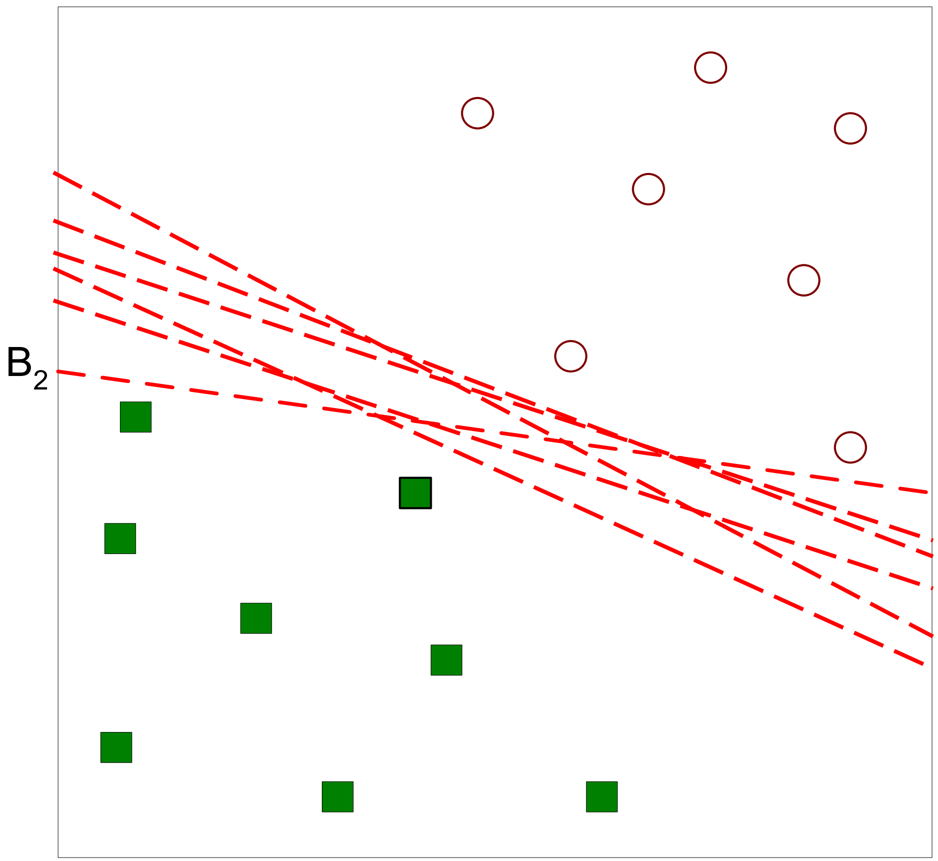
- One Possible Solution

Support Vector Machines



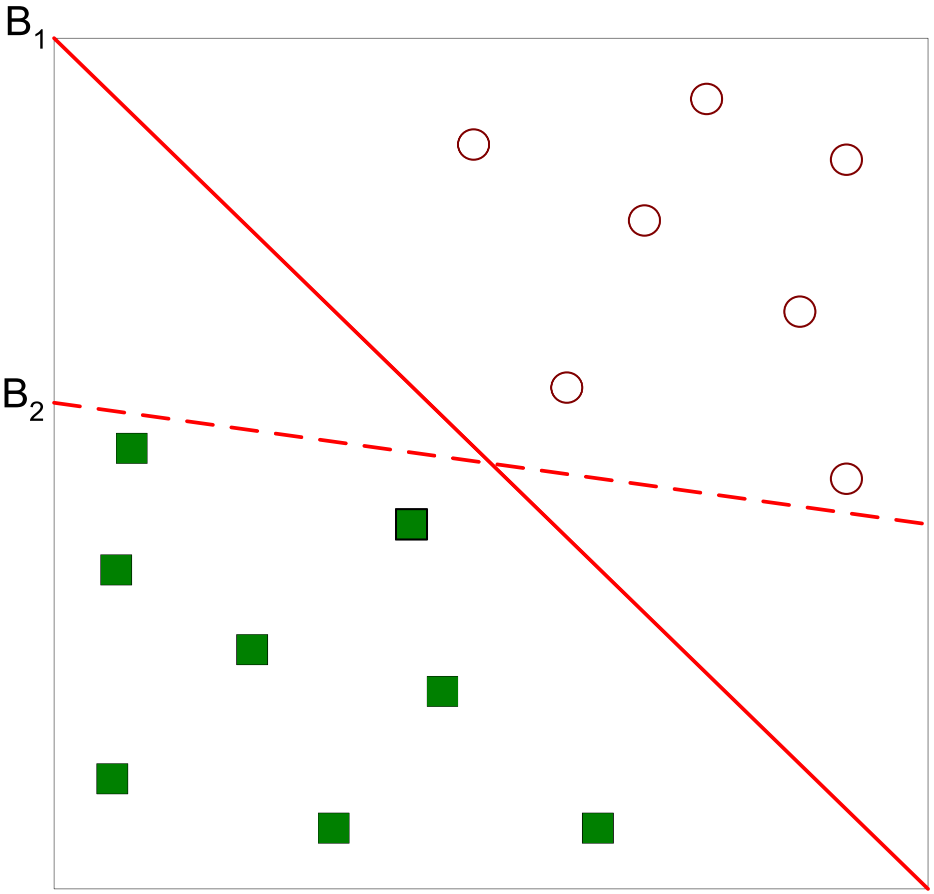
- Another possible solution

Support Vector Machines



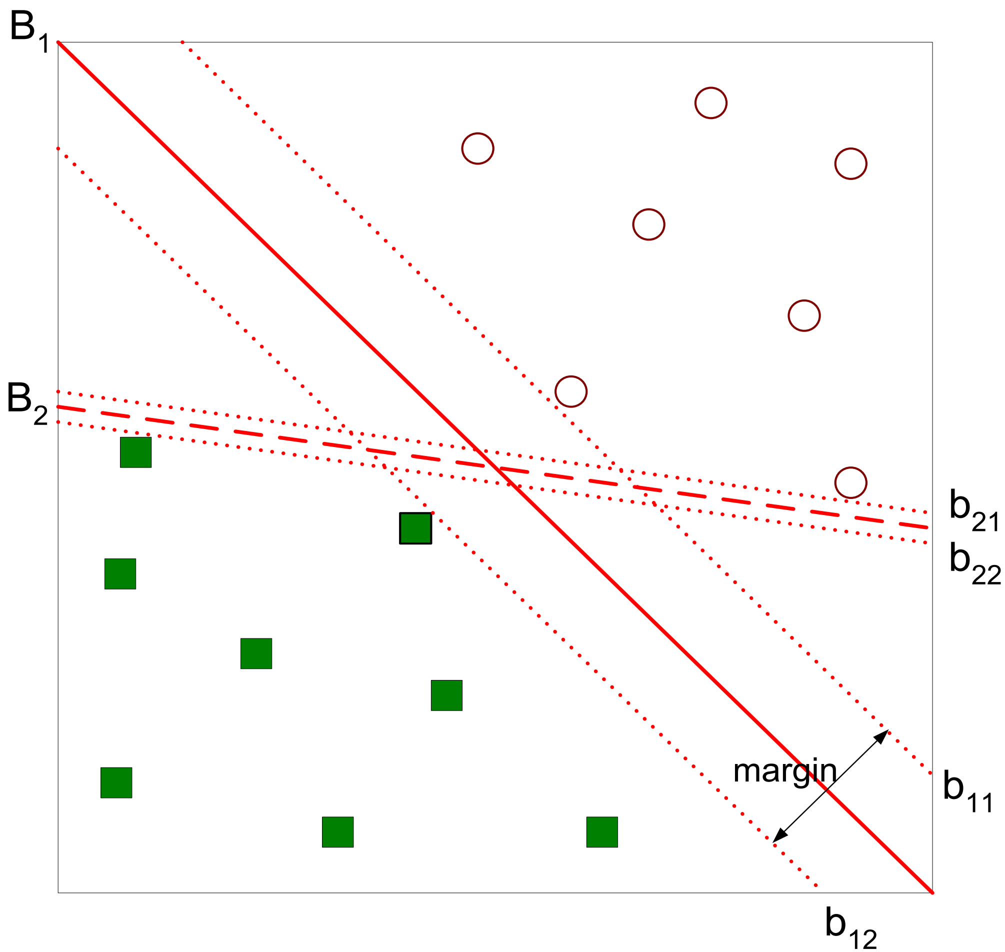
- Other possible solutions

Support Vector Machines



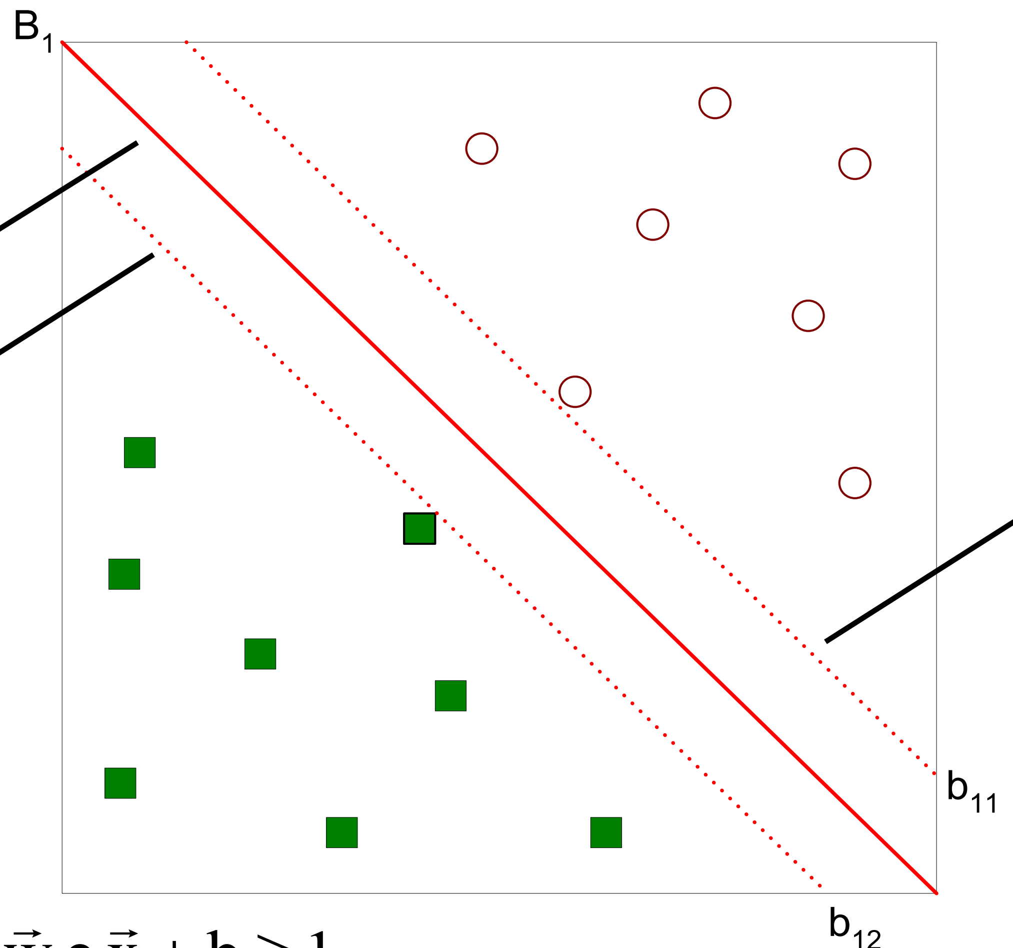
- Which one is better? B1 or B2?
- How do you define better?

Support Vector Machines



- Find hyperplane maximizes the margin => B1 is better than B2

Support Vector Machines



$$\vec{w} \bullet \vec{x} + b = 0$$

$$\vec{w} \bullet \vec{x} + b = -1$$

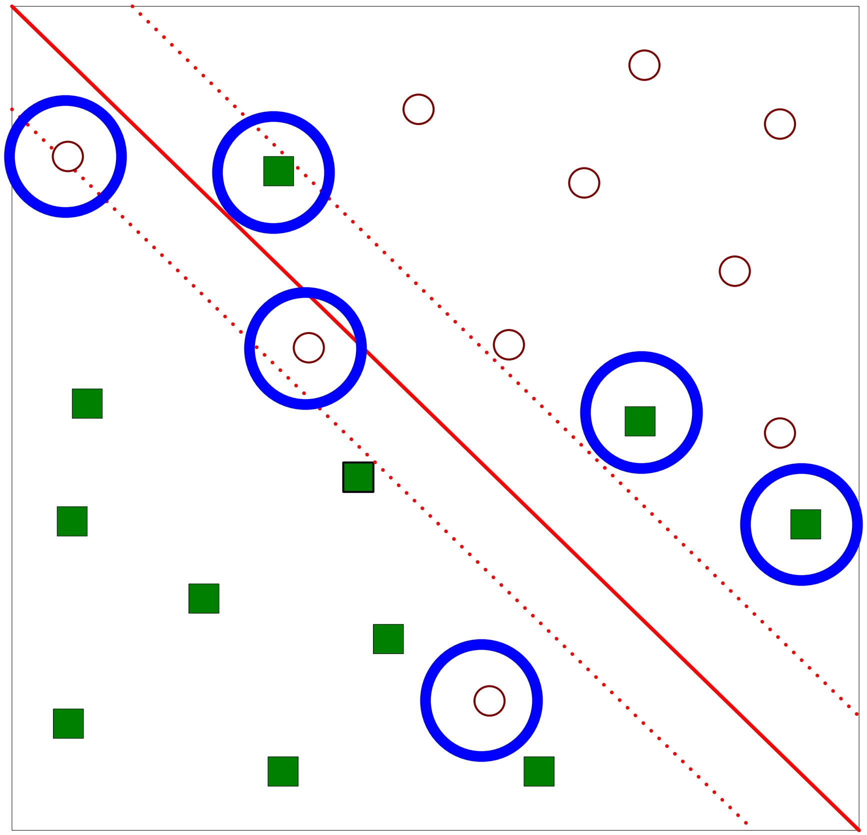
$$\vec{w} \bullet \vec{x} + b = +1$$

$$f(\vec{x}) = \begin{cases} 1 & \text{if } \vec{\mathrm{w}} \bullet \vec{\mathrm{x}} + \mathrm{b} \geq 1 \\ -1 & \text{if } \vec{\mathrm{w}} \bullet \vec{\mathrm{x}} + \mathrm{b} \leq -1 \end{cases}$$

$$\text{Margin} = \frac{2}{\| \vec{w} \|^2}$$

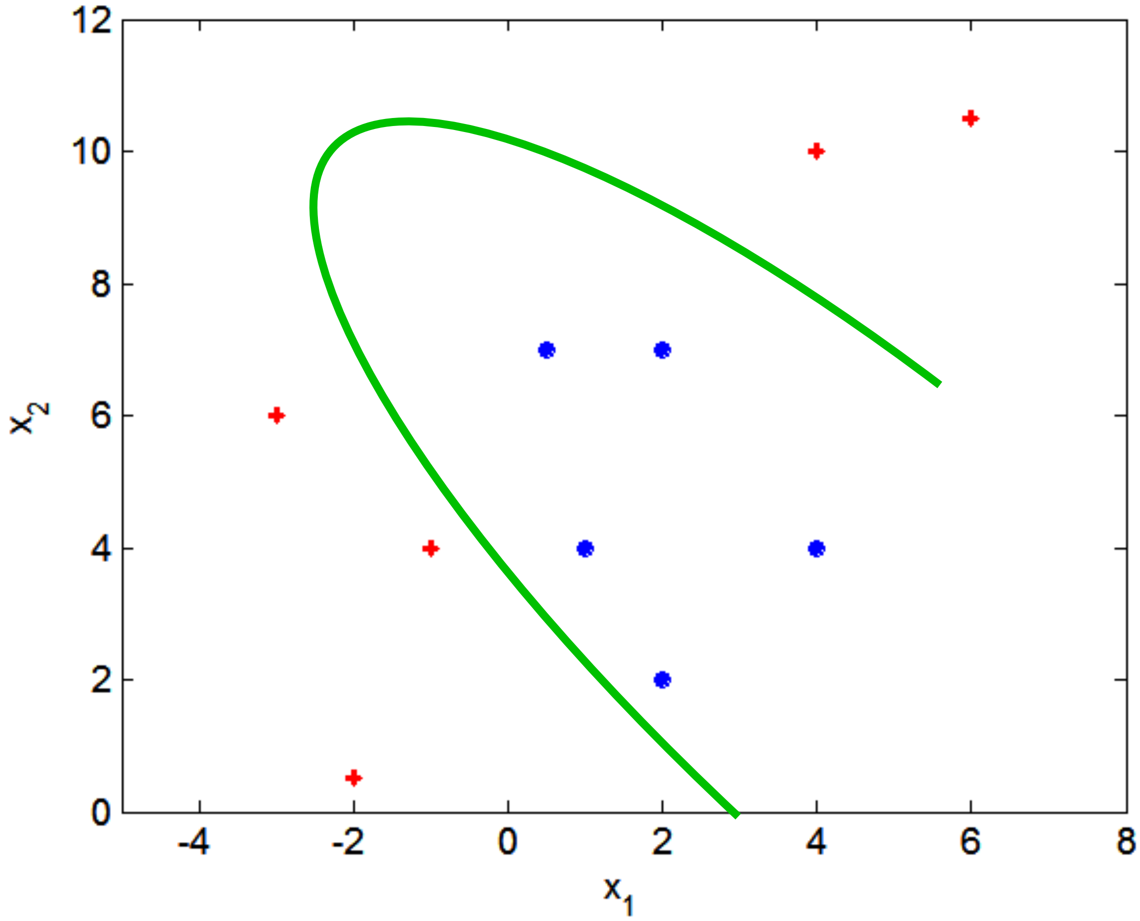
Support Vector Machines

- What if the problem is not linearly separable?



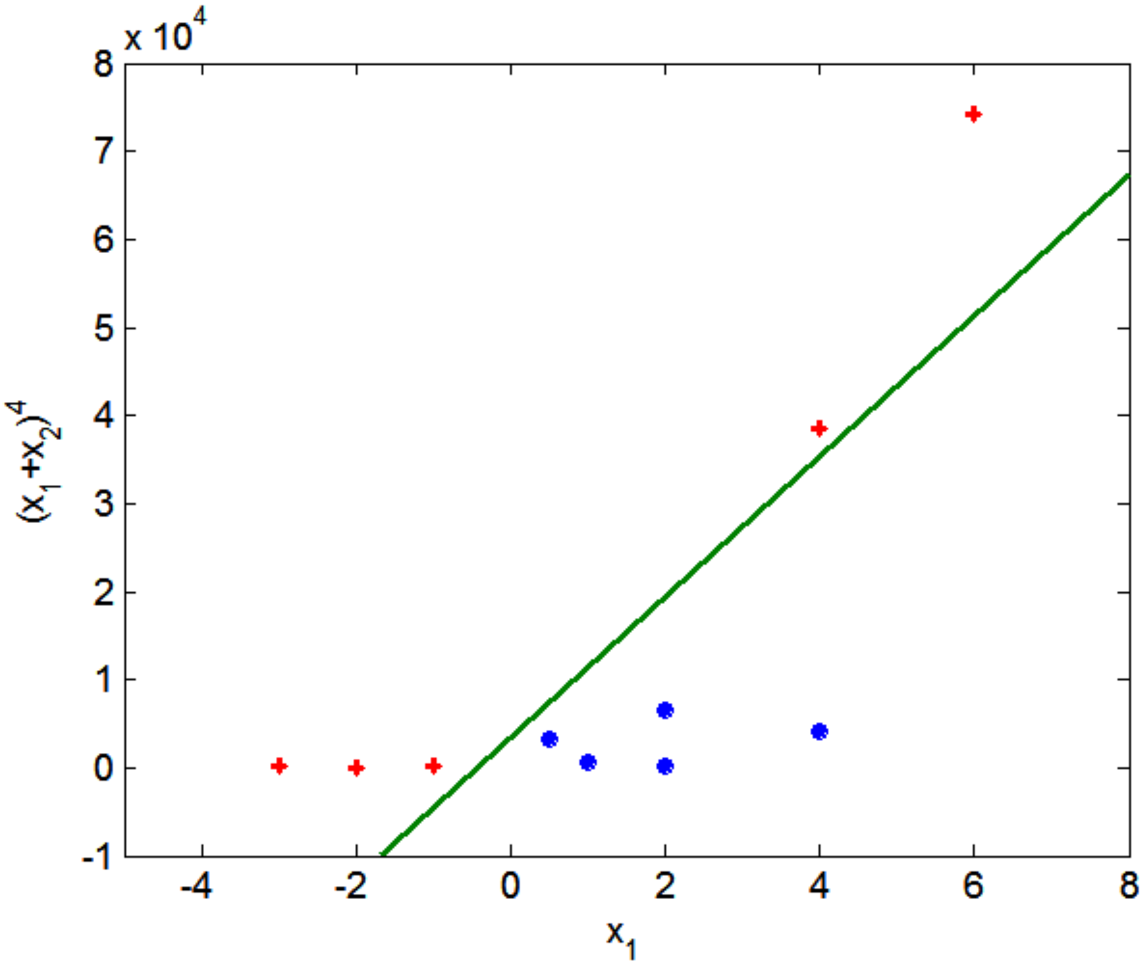
Nonlinear Support Vector Machines

- What if decision boundary is not linear?



Nonlinear Support Vector Machines

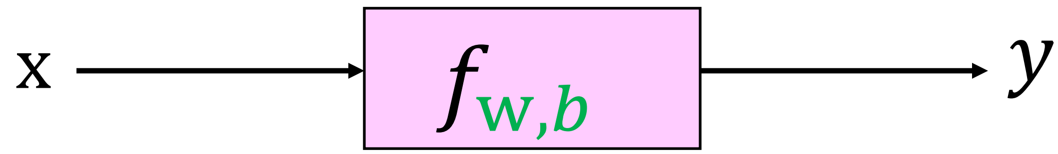
- Transform data into higher dimensional space



Outline of SVM lecture

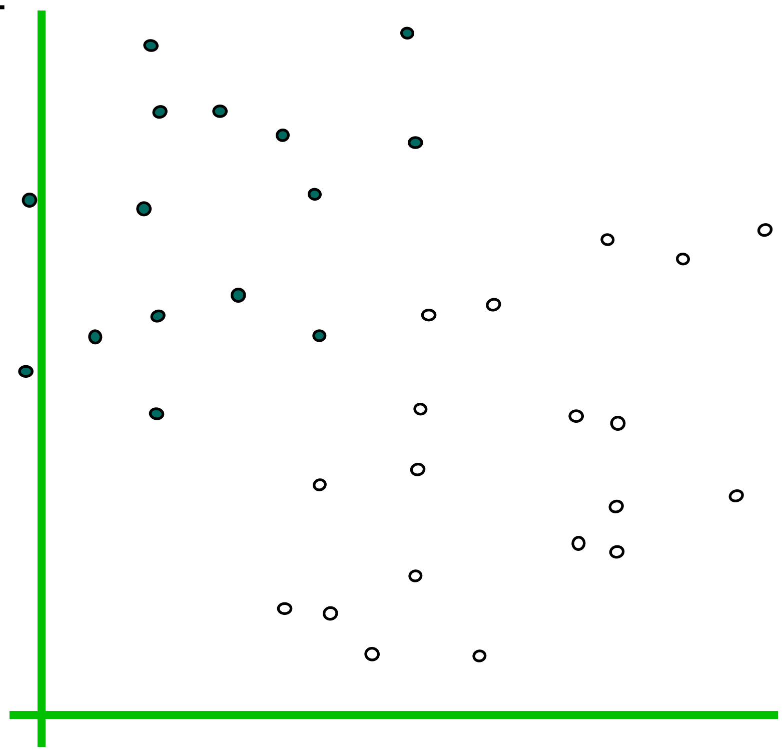
- Linear classifier
- Maximum margin classifier
  - Estimate the margin
- SVM for separable data
- SVM for non-separable data

Linear classifiers



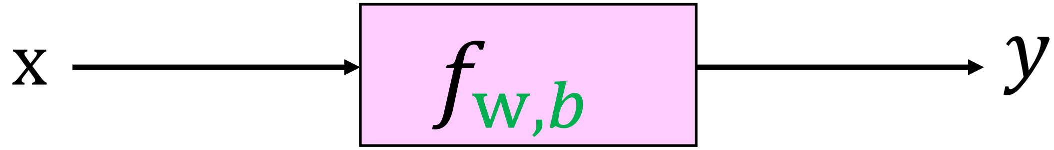
$$f(\mathrm{x}, \mathrm{w}, b) = sign(\mathrm{w} \cdot \mathrm{x} + b)$$

- denotes +1
- denotes -1



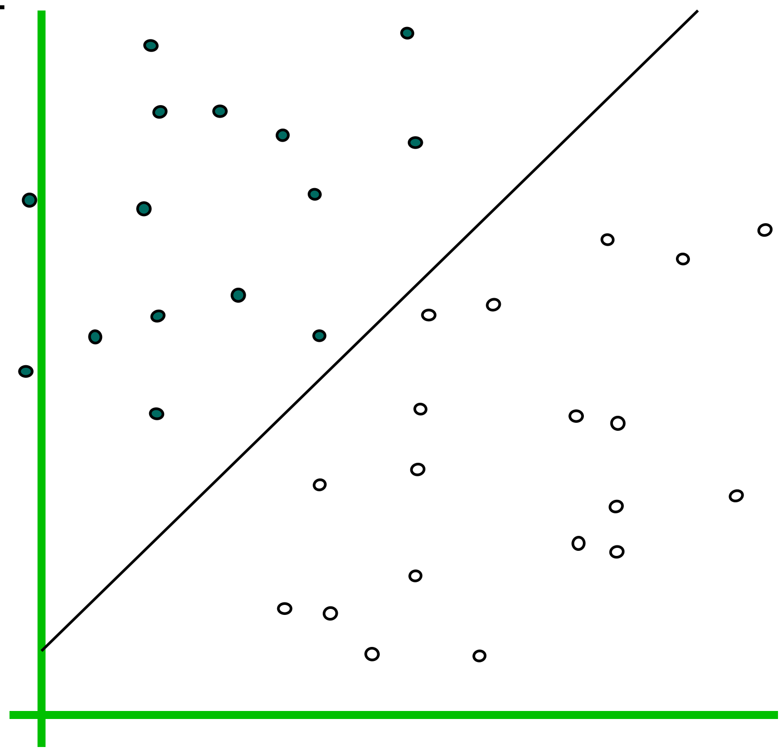
How would you classify this data?

Linear classifiers



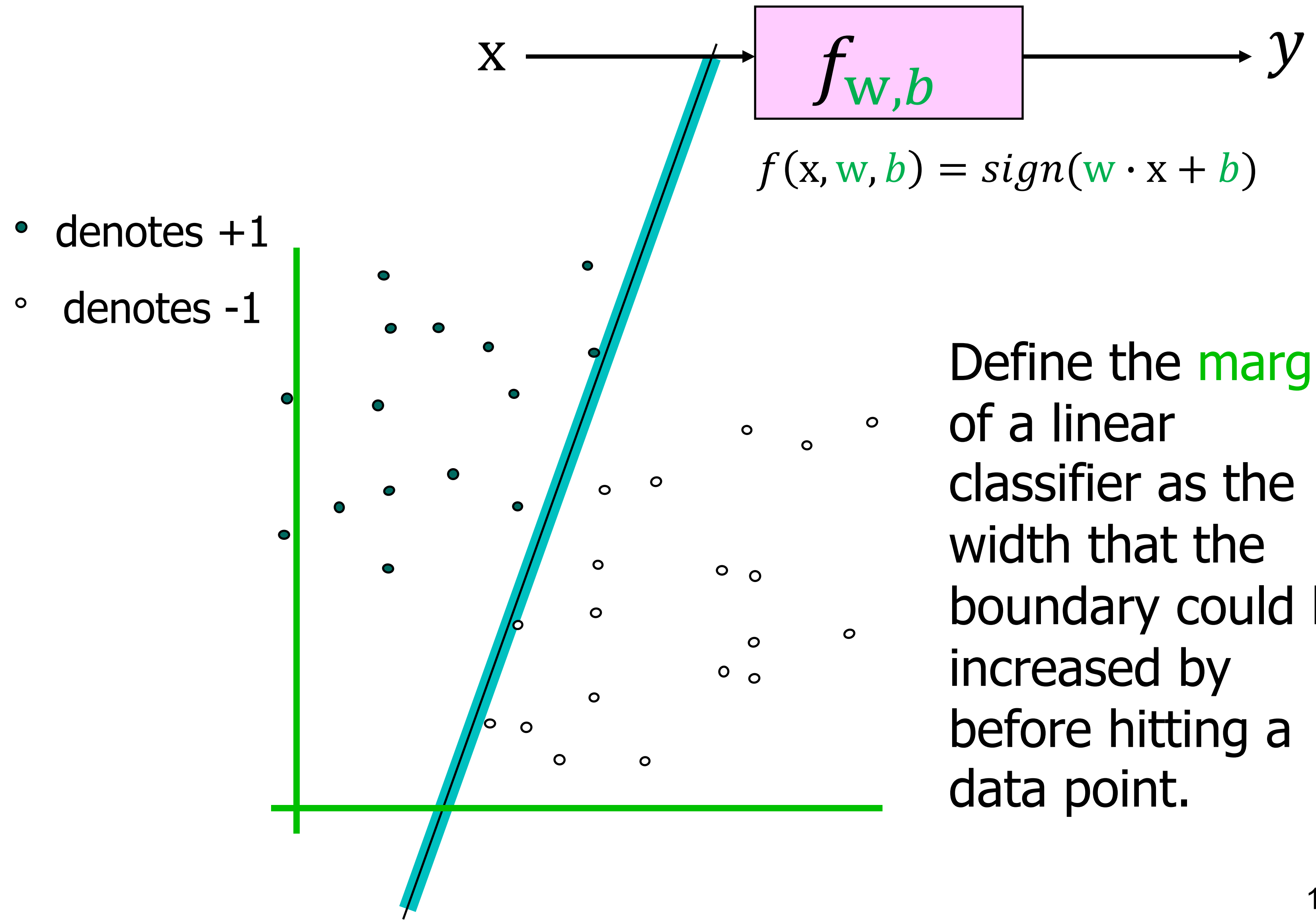
$$f(\mathrm{x}, \mathrm{w}, b) = sign(\mathrm{w} \cdot \mathrm{x} + b)$$

- denotes +1
- denotes -1



How would you classify this data?

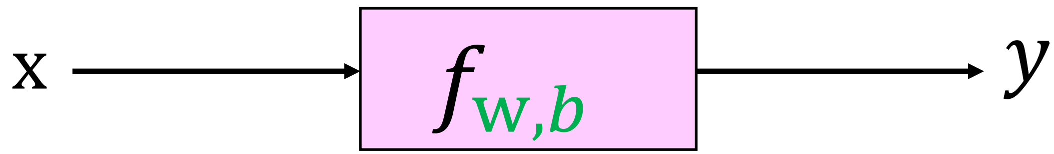
Classifier Margin



$$f(x, w, b) = sign(w \cdot x + b)$$

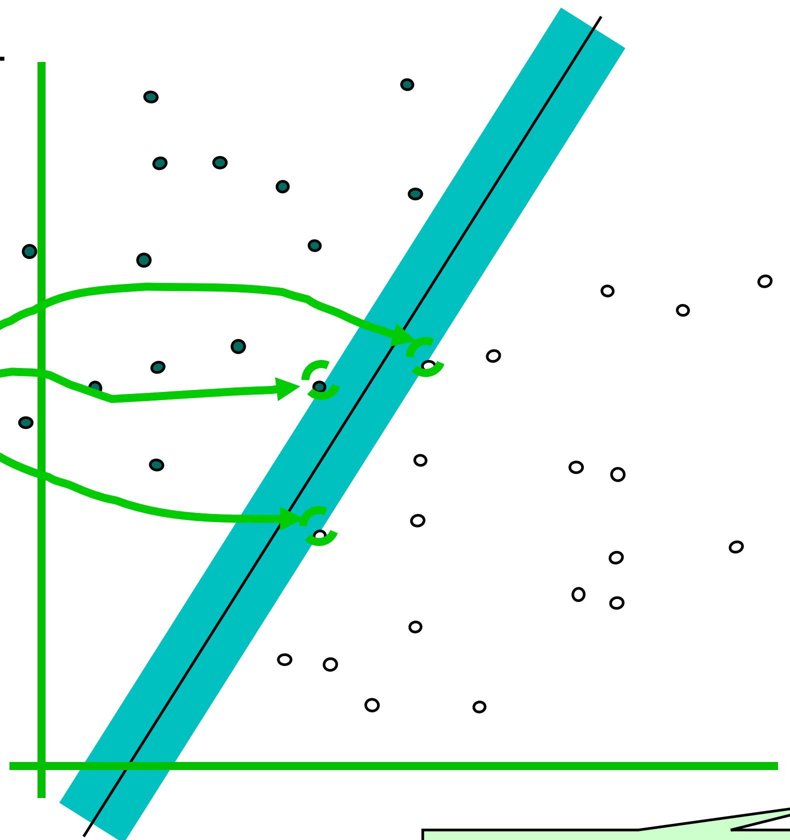
Define the margin of a linear classifier as the width that the boundary could be increased by before hitting a data point.

Maximum Margin



$$f(x, w, b) = sign(w \cdot x + b)$$

- denotes +1
- denotes -1



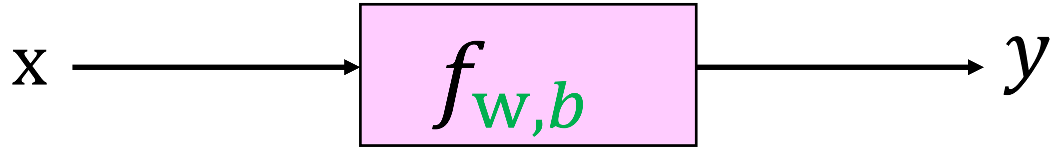
Support Vectors are those data points that the margin pushes up against

The maximum margin linear classifier is the linear classifier with the maximum margin.

This is the simplest kind of SVM (Called an LSVM)

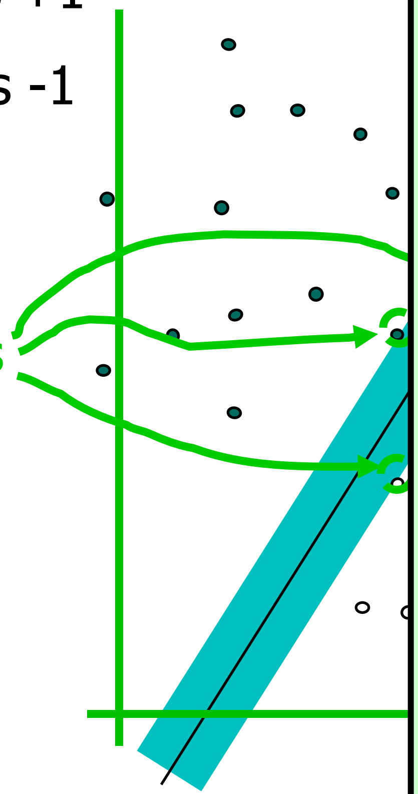
Linear SVM

Why Maximum Margin?



$f(x,w,b) = sign(w \cdot x + b)$

- denotes +1
- ° denotes -1



Support Vectors are those data points that the margin pushes up against

1. Intuitively this feels safest.

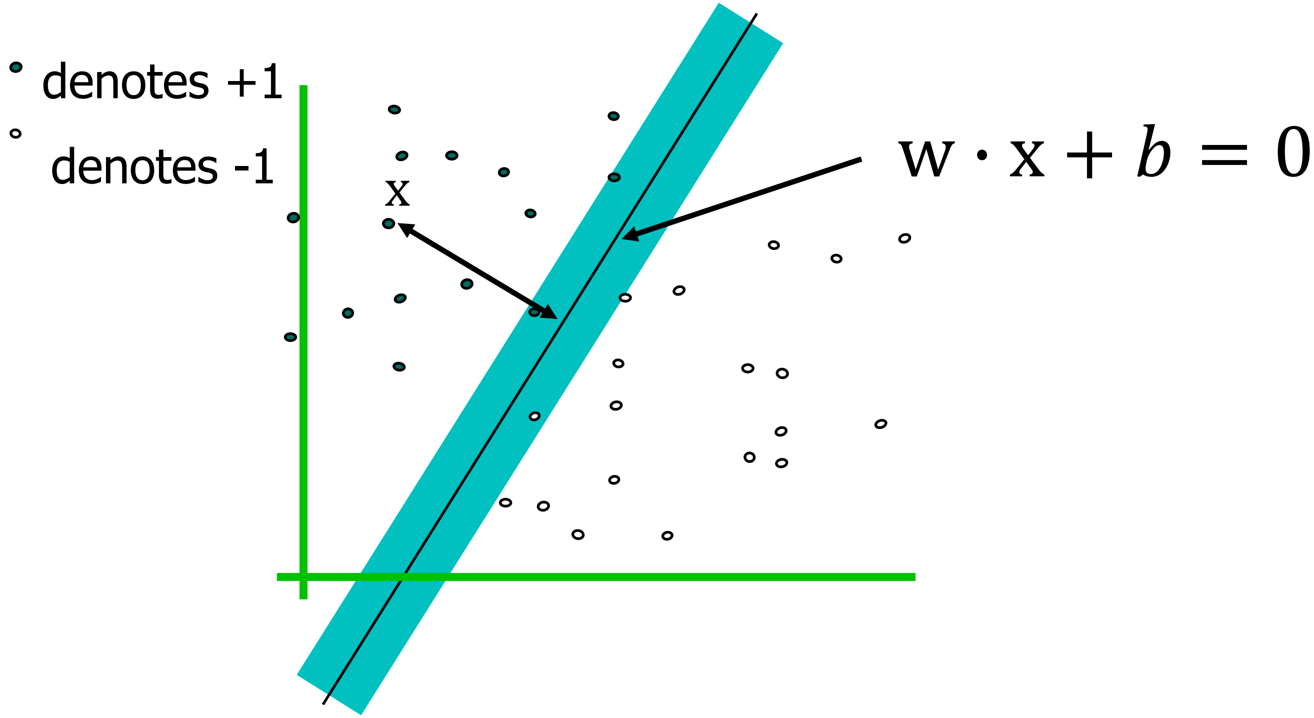
   If we've made a small error in the location of the boundary this gives us least chance of causing a misclassification.

2. The model is immune to removal of any non-support-vector data points.

3. There's some theory (using VC dimension) that is related to (but not the same as) the proposition that this is a good thing.

4. Empirically it works very very well.

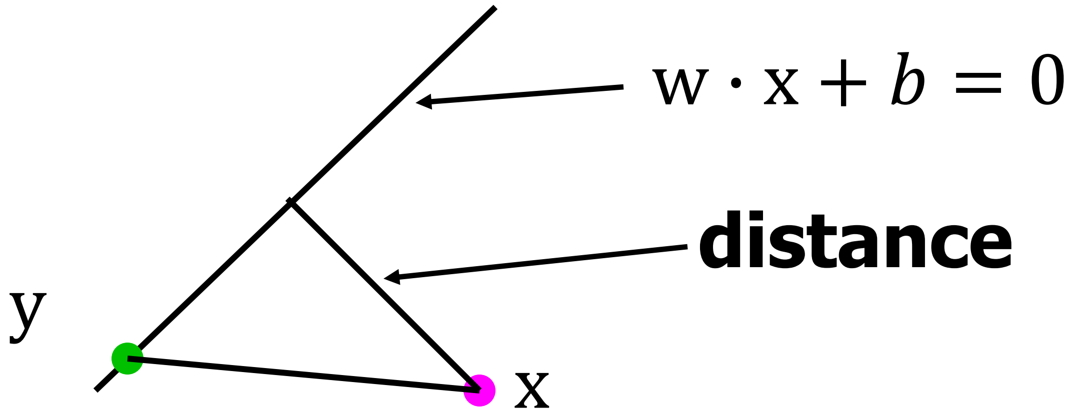
Estimate the Margin



- What is the distance expression for a point x to a hyperplane $\mathrm{w} \cdot \mathrm{x} + b = 0$?

$$d(\mathbf{x}) = \frac{\left|\mathbf{x} \cdot \mathbf{w} + b\right|}{\sqrt{\|\mathbf{w}\|_2^2}} = \frac{\left|\mathbf{x} \cdot \mathbf{w} + b\right|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

Estimate the Margin

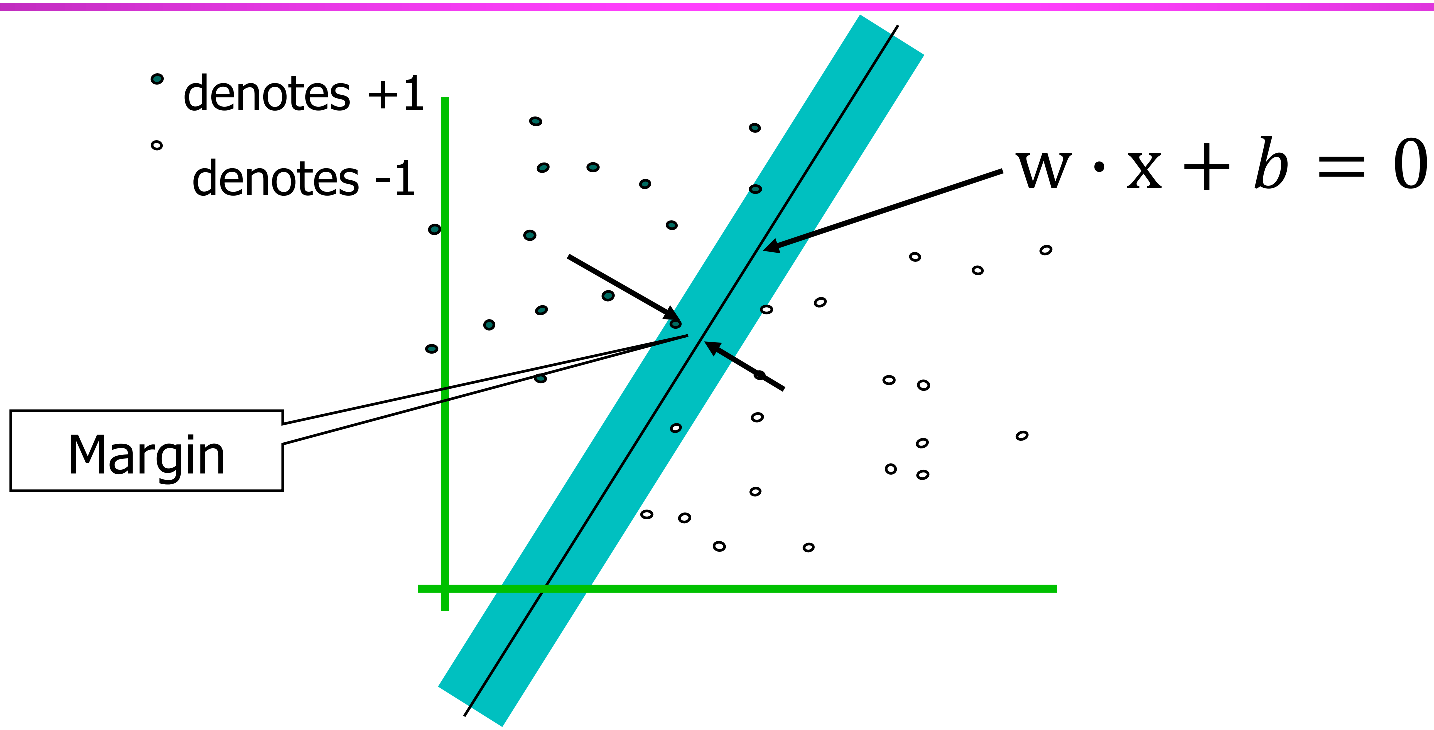


$$\left\| \left(y - x, \frac{w}{\|w\|} \right) \right\| = \frac{|(y-x)w|}{\|w\|} = \frac{|yw - xw|}{\|w\|}$$

Using $yw + b = 0$, we have

$$d = \frac{|-b - xw|}{\|w\|} = \frac{|b + xw|}{\sqrt{\|w\|^2}} = \frac{|b + xw|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

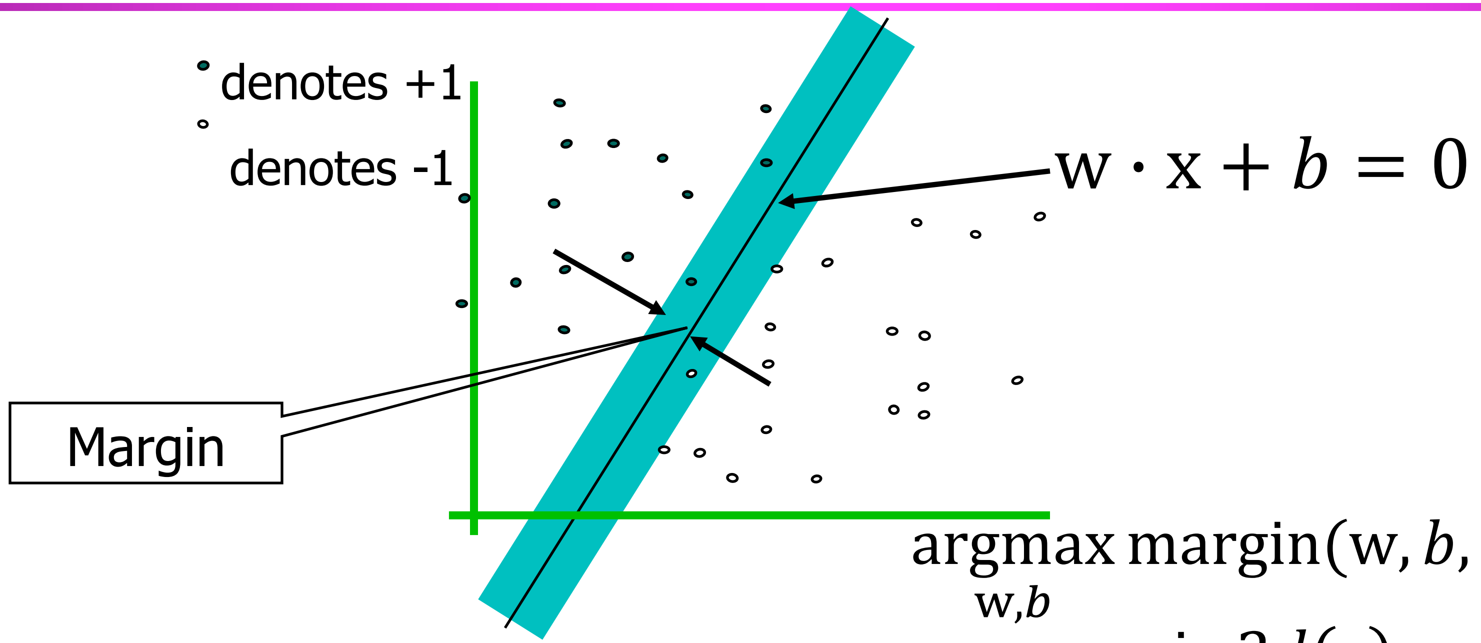
Estimate the Margin



- What is the expression for margin?

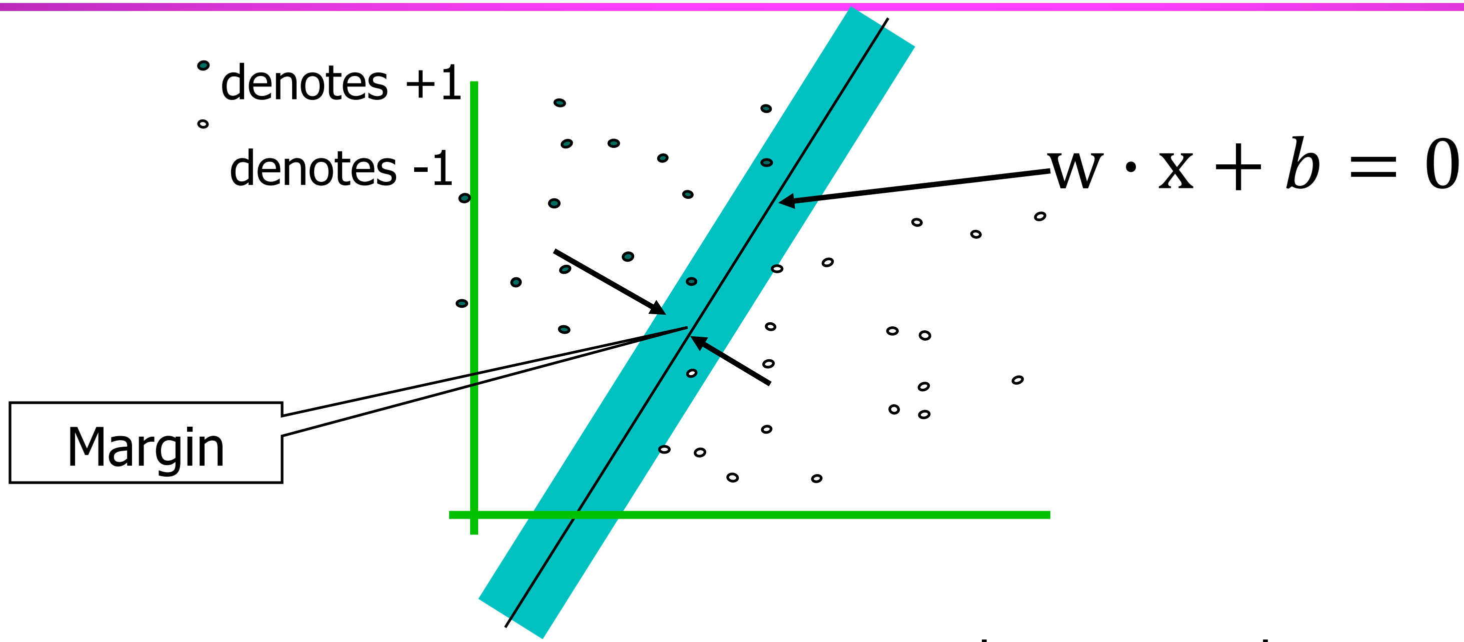
$$\text{margin} \equiv \min_{\mathrm{x} \in D} 2d(\mathrm{x}) = \min_{\mathrm{x} \in D} 2 \frac{|\mathrm{w} \cdot \mathrm{x} + b|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

Maximize Margin



$$
\begin{aligned}
&\operatorname*{argmax}_{\mathrm{w},b}\ \mathrm{margin}(\mathrm{w},b,D)\\
&=\operatorname*{argmax}_{\mathrm{w},b}\ \min_{\mathrm{x}\in D}\ 2d(\mathrm{x})\\
&=\operatorname*{argmax}_{\mathrm{w},b}\ \min_{\mathrm{x}\in D}\ 2\frac{|\mathrm{w}\cdot\mathrm{x}+b|}{\sqrt{\sum_{i=1}^{d} w_i^2}}\\
&=\operatorname*{argmax}_{\mathrm{w},b}\ \min_{\mathrm{x}\in D}\frac{|\mathrm{w}\cdot\mathrm{x}+b|}{\sqrt{\sum_{i=1}^{d} w_i^2}}
\end{aligned}
$$

Maximize Margin

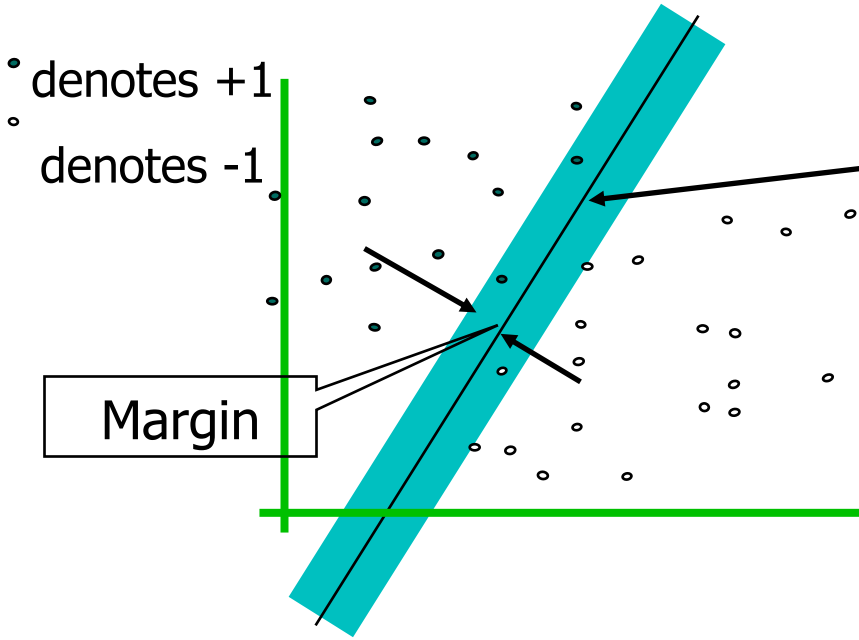


$$\underset{\mathrm{w},b}{\operatorname{argmax}} \min_{\mathrm{x}_i \in D} \frac{|\mathrm{w} \cdot \mathrm{x}_i + b|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

$$\text{subject to } \forall \mathrm{x}_i \in D\colon\ y_i(\mathrm{w} \cdot \mathrm{x}_i + b) > 0$$

- Min-max problem

Maximize Margin



$$\underset{\mathrm{w},b}{\operatorname{argmax}}\ \min_{\mathrm{x}_i \in D} \frac{|\mathrm{w}\cdot\mathrm{x}_i + b|}{\sqrt{\sum_{i=1}^{d} w_i^2}}$$

subject to $\forall \mathrm{x}_i \in D:\ y_i(\mathrm{w}\cdot\mathrm{x}_i + b) > 0$

Strategy:

$$\forall \mathrm{x}_i \in D:\ |\mathrm{w}\cdot\mathrm{x}_i + b| \ \geq 1$$

$$\underset{\mathrm{w},b}{\operatorname{argmin}} \sum_{i=1}^{d} w_i^2$$

subject to $\forall \mathrm{x}_i \in D:\ y_i(\mathrm{w}\cdot\mathrm{x}_i + b) \geq 1$

Maximum Margin Linear Classifier

$$\{\mathrm{w}^*, b^*\} = \underset{\mathrm{w},b}{\operatorname{argmin}} \sum_{i=1}^{d} w_i^2$$

$$\text{subject to } y_1(\mathrm{w} \cdot \mathrm{x}_1 + b) \geq 1$$

$$y_2(\mathrm{w} \cdot \mathrm{x}_2 + b) \geq 1$$

$$\ldots$$

$$y_N(\mathrm{w} \cdot \mathrm{x}_N + b) \geq 1$$

- How to solve it?

Learning via Quadratic Programming

- QP is a well-studied class of optimization algorithms to maximize a quadratic function of some real-valued variables subject to linear constraints.
- Available open-source solvers
  - SVMLight http://svmlight.joachims.org/
  - LibSVM http://www.csie.ntu.edu.tw/~cjlin/libsvm/
  - Matlab optimization toolbox

Quadratic Programming

$$\underset{\mathrm{u}}{\operatorname{argmin}}\, c + \mathrm{q}^T \mathrm{u} + \frac{\mathrm{u}^T \mathrm{R} \mathrm{u}}{2}$$

Quadratic objective

$$\text{subject to } \mathrm{A}\mathrm{u} \leq \mathrm{b}$$

Linear inequality constraints

$$\mathrm{C}\mathrm{u} = \mathrm{d}$$

Linear equality constraints

- R, A and C are pre-given matrices
- q, b and d are pre-given vectors
- c is a pre-given scalar

Quadratic Programming of SVM

$$\{\mathrm{w}^*, b^*\} = \underset{\mathrm{w},b}{\operatorname{argmin}} \sum_{i=1}^{d} w_i^2$$

$$\text{subject to } y_1(\mathrm{w}\cdot\mathrm{x}_1 + b) \geq 1$$

$$y_2(\mathrm{w}\cdot\mathrm{x}_2 + b) \geq 1$$

$$\ldots$$

$$y_N(\mathrm{w}\cdot\mathrm{x}_N + b) \geq 1$$

See white board

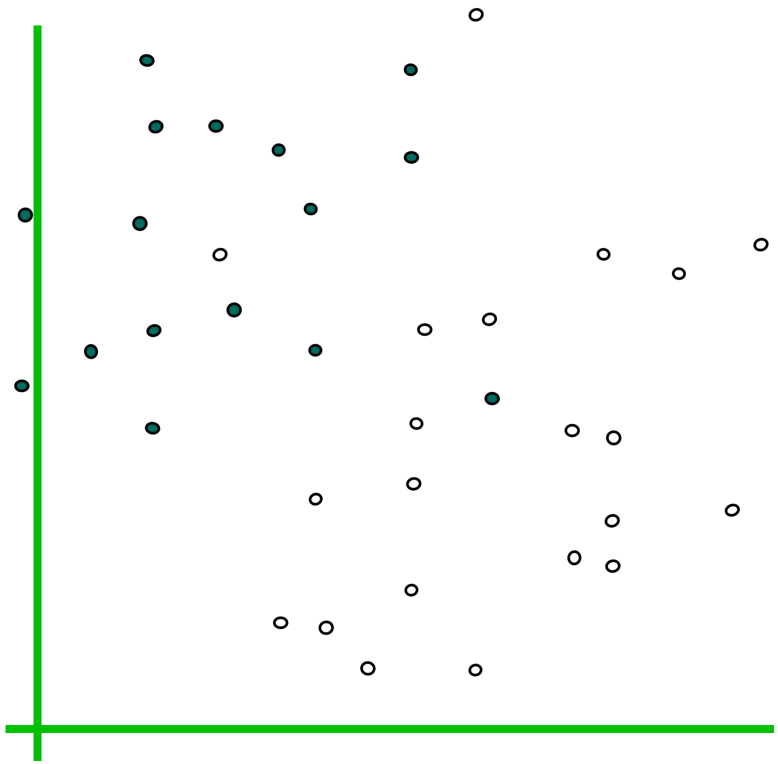
$$\underset{\mathrm{u}}{\operatorname{argmin}}\, c + \mathrm{q}^T\mathrm{u} + \frac{\mathrm{u}^T\mathrm{R}\mathrm{u}}{2}$$

$$\text{subject to } \mathrm{A}\mathrm{u} \leq \mathrm{b}$$

$$\mathrm{C}\mathrm{u} = \mathrm{d}$$

Non-separable

- denotes +1
- ◦ denotes -1

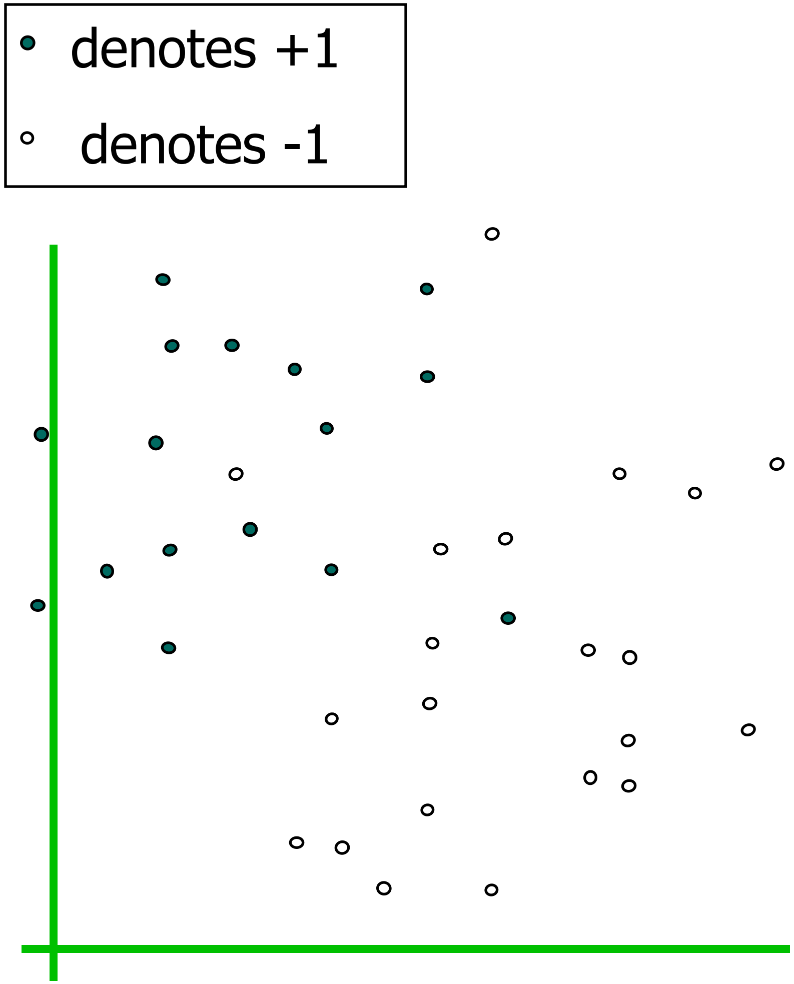


$$\underset{\mathrm{w},b}{\operatorname{argmin}} \sum_{i=1}^{d} w_i^2$$

$$\text{subject to } \forall \mathrm{x}_i \in D:\ y_i(\mathrm{w} \cdot \mathrm{x}_i + b) \geq 1$$

No such (w, b) that can satisfy the constraint on all x_i's

What should we do?

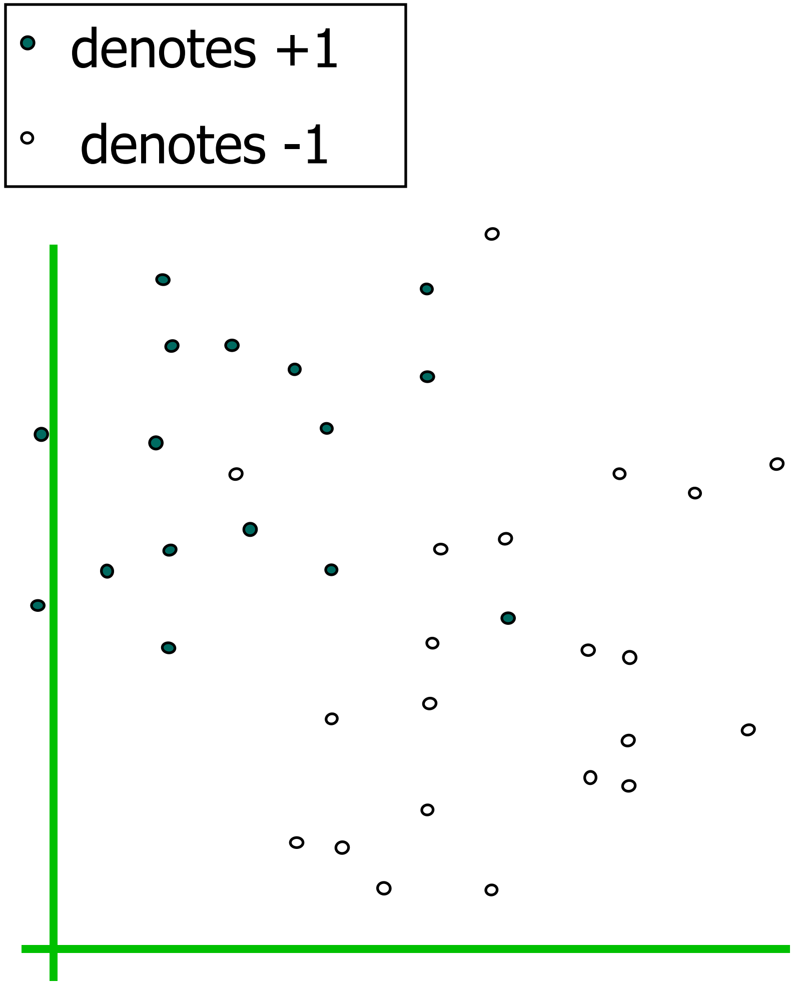


- Relax constraint to allow training error
- Find w that minimizes $||w||^2$, and the same time the number of training set errors

Problem

Two things to minimize makes for an ill-define optimization

What should we do?



- Minimize

$$\|\mathrm{w}\|^2 + C(\#train\ errors)$$

Tradeoff parameter

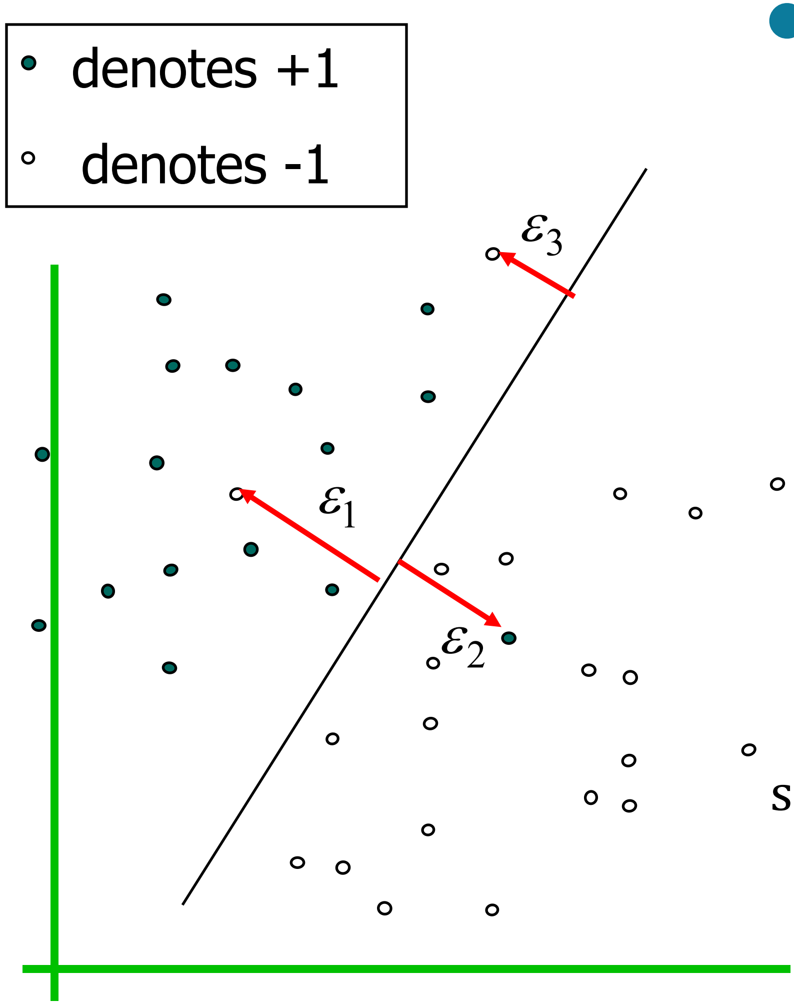
Some points will violate

$$y_i(\mathrm{w} \cdot \mathrm{x}_i + b) \geq 1$$

We allow errors to occur

$$y_i(\mathrm{w} \cdot \mathrm{x}_i + b) \geq 1 - \varepsilon_i, \quad \varepsilon_i \geq 0$$

What should we do?



- Minimize

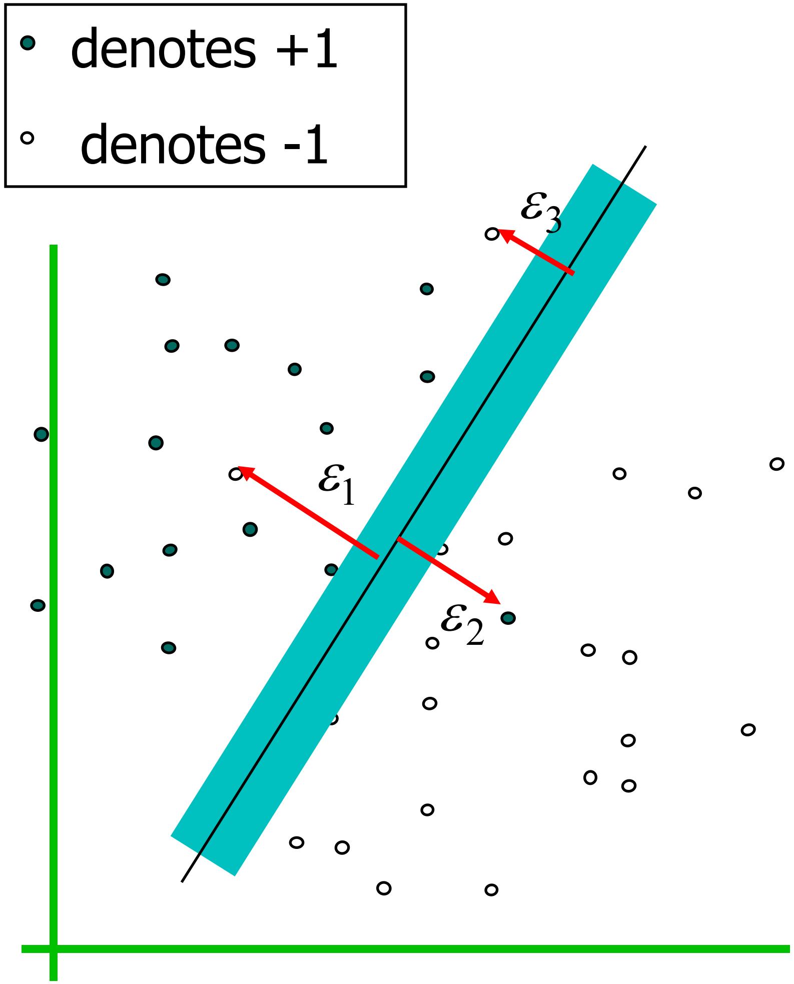
$$\|\mathrm{w}\|^2 + C(\mathit{distance\ of\ error\ points\ to\ their\ correct\ place})$$

$$\sum_{i=1}^{N} \varepsilon_i$$

subject to $\forall \mathrm{x}_i \in D: y_i(\mathrm{w} \cdot \mathrm{x}_i + b) \geq 1 - \varepsilon_i, \varepsilon_i \geq 0$

Hinge loss

Linear inseparable case



$$\underset{\mathrm{w},b,\varepsilon}{\operatorname{argmin}} \sum_{i=1}^{d} w_i^2 + C \sum_{j=1}^{N} \varepsilon_j$$

$$\begin{aligned} \text{subject to } & y_1(\mathrm{w} \cdot \mathrm{x}_1 + b) \geq 1 - \varepsilon_1, \varepsilon_1 \geq 0 \\ & y_2(\mathrm{w} \cdot \mathrm{x}_2 + b) \geq 1 - \varepsilon_2, \varepsilon_2 \geq 0 \\ & \dots \\ & y_N(\mathrm{w} \cdot \mathrm{x}_N + b) \geq 1 - \varepsilon_N, \varepsilon_N \geq 0 \end{aligned}$$

C balances the trade off between margin and classification errors

Determining value for c

- How do we determine the appropriate value for *c* ?
- Cross-validation on training data
  - Take possible choices for c
  - For each choice,
    - Run a cross validation procedure
    - Calculate the error metric (chosen properly)
  - Find the choice that achieves the best metric
  - Use the best choice on all training data