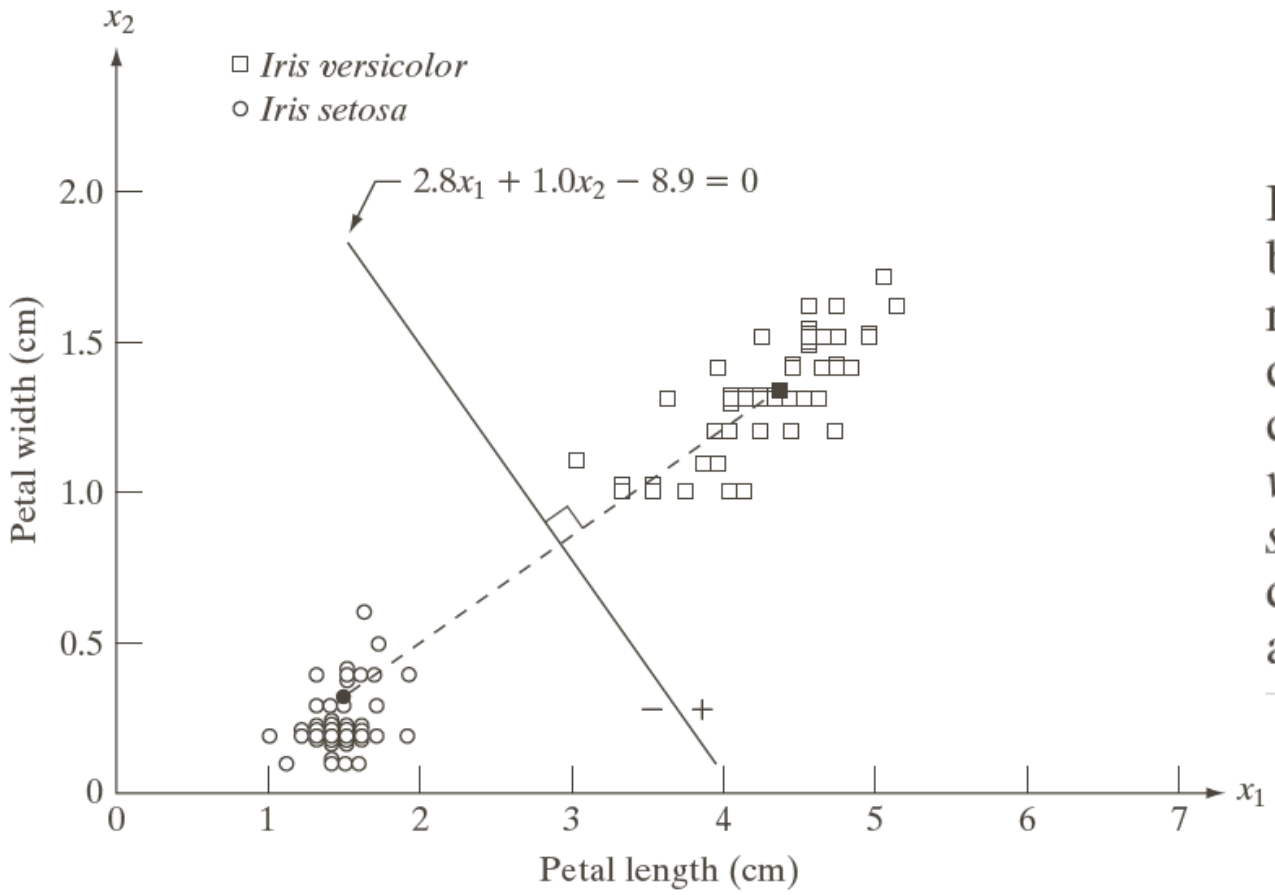
LECT-14
Machine Learning

Minimum Distance

Minimum Distance Classifier



Decision boundary of minimum distance classifier for the classes of *Iris versicolor* and *Iris setosa*. The dark dot and square are the means.

Minimum Distance Classifier

- For a test sample X, compute $D_j(X)$ for each class j
- Assign class with minimum D(x) value

$$D_j(\mathbf{x}) = \left\| \mathbf{x} - \mathbf{m}_j \right\|$$

- Here $m_j = \left[(\mathbf{x} - \mathbf{m}_j)^T (\mathbf{x} - \mathbf{m}_j) \right]^{1/2}$ ก j[th] class

Minimum Distance Classifier

Manipulating $D_j(X)$

$$
\begin{aligned}
D_j^2(\mathbf{x}) &= \left\|\mathbf{x}-\mathbf{m}_j\right\|^2 = (\mathbf{x}-\mathbf{m}_j)^T(\mathbf{x}-\mathbf{m}_j) \\
&= \mathbf{x}^T\mathbf{x}-2\mathbf{x}^T\mathbf{m}_j+\mathbf{m}_j^T\mathbf{m}_j \\
&= \mathbf{x}^T\mathbf{x}-\mathbf{2}\left(\mathbf{x}^T\mathbf{m}_j-\frac{1}{2}\mathbf{m}_j^T\mathbf{m}_j\right).
\end{aligned}
$$

Minimum Distance Classifier

- Now instead of $D_j(X)$, we compute discriminent function $d_j(X)$ for each class
- Assign class with maximum $d_j(X)$ value

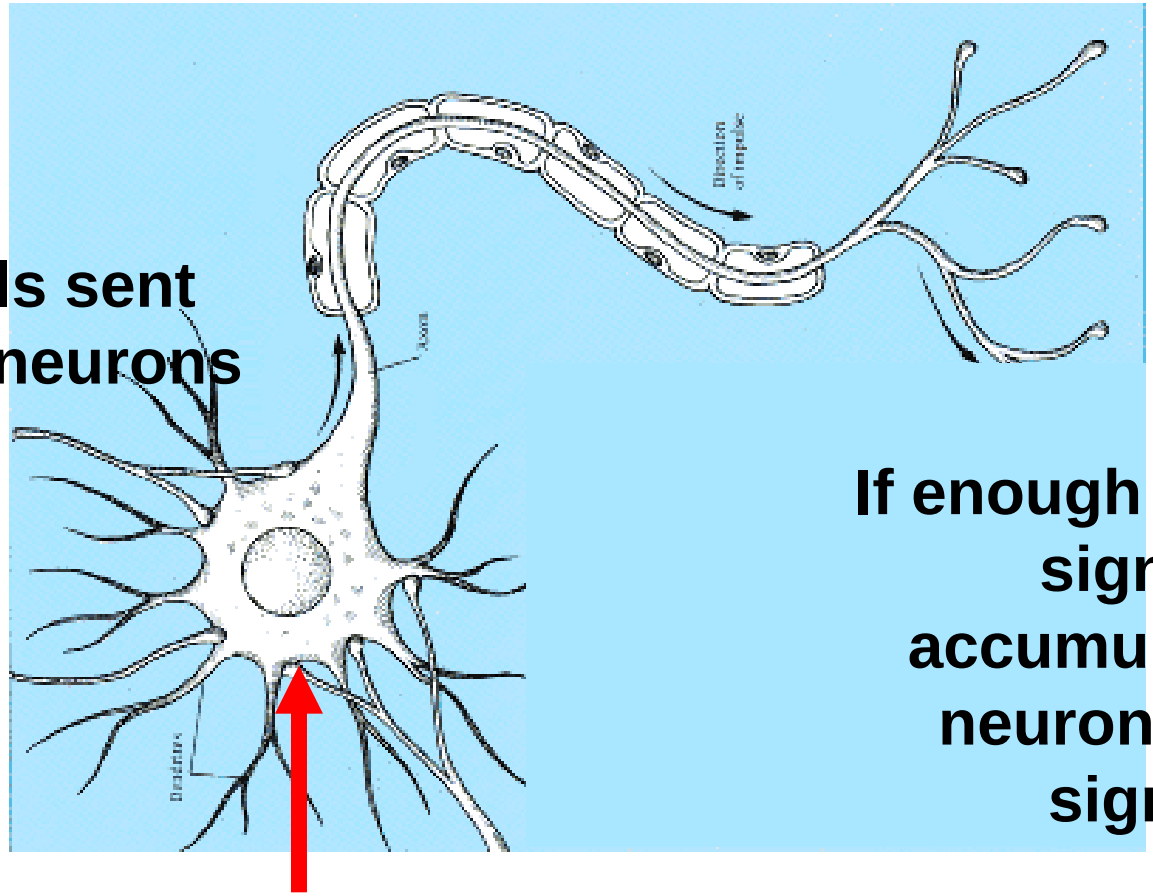
$$d_j(\mathbf{x}) = \mathbf{x}^T\mathbf{m}_j - \frac{1}{2}\mathbf{m}_j^T\mathbf{m}_j \qquad j = 1, 2, \ldots, W$$

- Equation for decision boundary between two classes *i* and *j*

$$d_{ij}(\mathbf{x}) = \mathbf{x}^T(\mathbf{m}_i - \mathbf{m}_j) - \frac{1}{2}(\mathbf{m}_i^T\mathbf{m}_i - \mathbf{m}_j^T\mathbf{m}_j)$$

Artificial Neural Network - Perceptron

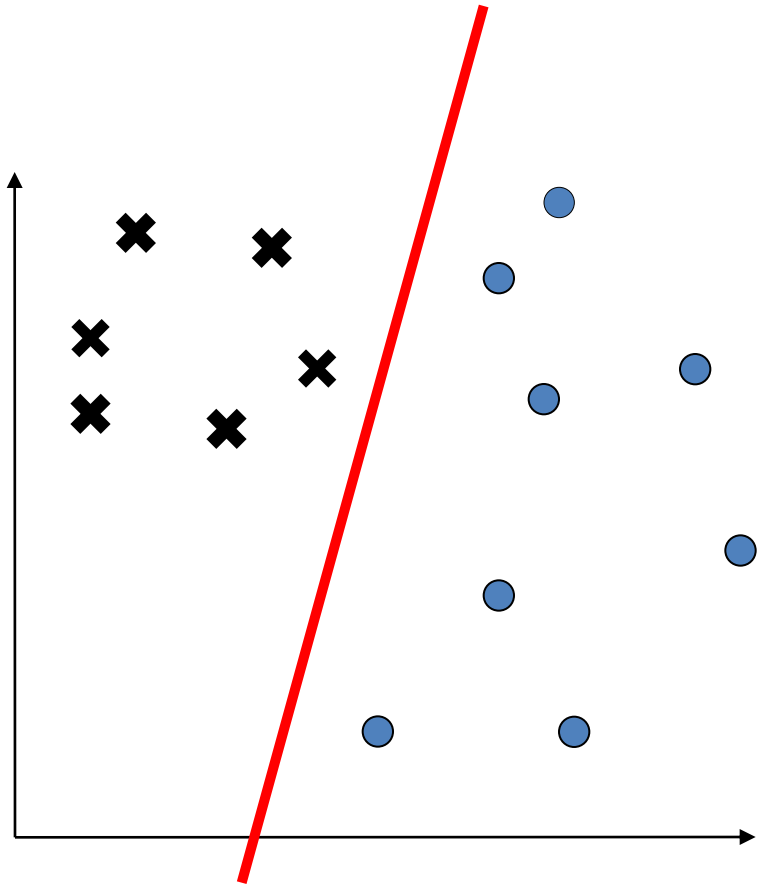
Input signals sent from other neurons



If enough sufficient signals accumulate, the neuron fires a signal.

Connection strengths determine how the signals are accumulated

A (Linear) Decision Boundary

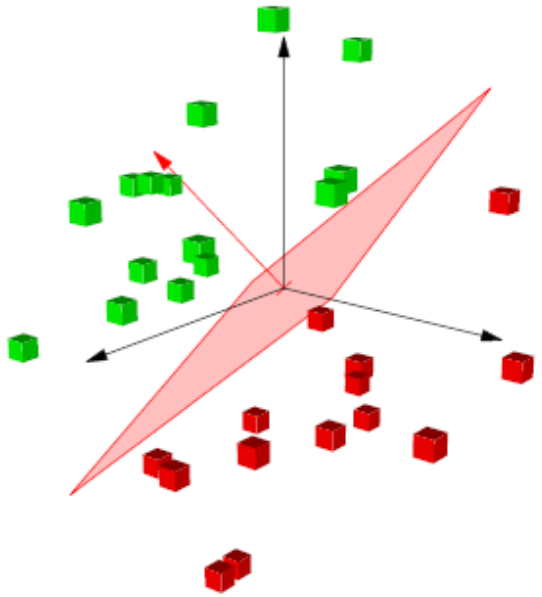


Represented by:
One artificial neuron called a "Perceptron"

Low space complexity

Perceptron

- The perceptron with a step function performs **classification**
- The perceptron can be 'visualised' as a decision boundary in input space



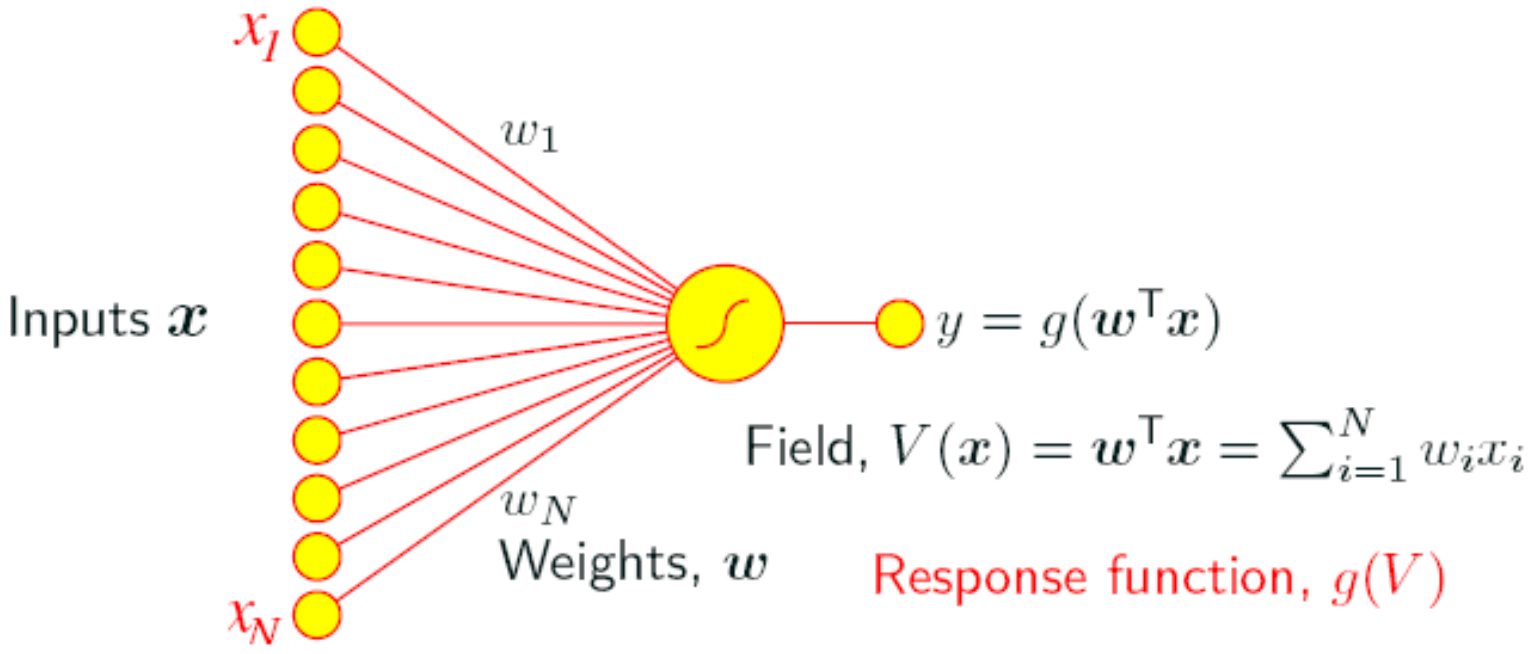
- The perceptron can only separate linear-separable inputs

Perceptron

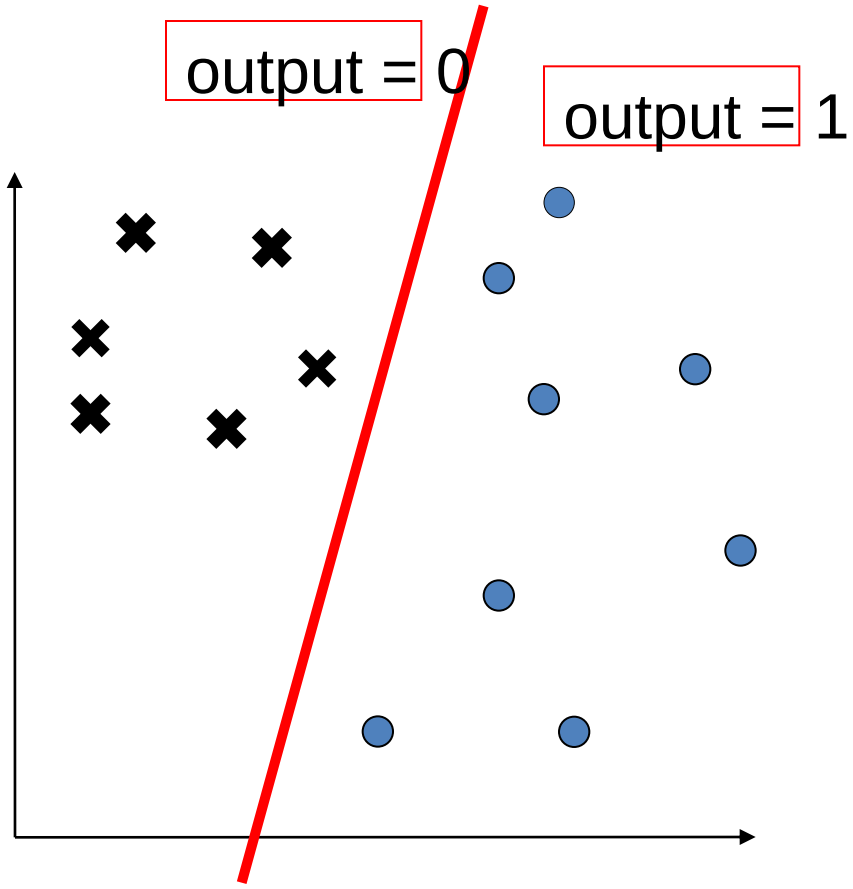
- Given (numeric) input features $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$
- Prediction given by $f(\boldsymbol{x}; \boldsymbol{w})$
- $\boldsymbol{w}$ are parameters or "weights" that we train
- The **perceptron** provides the classic example of a **parametric** learning algorithm

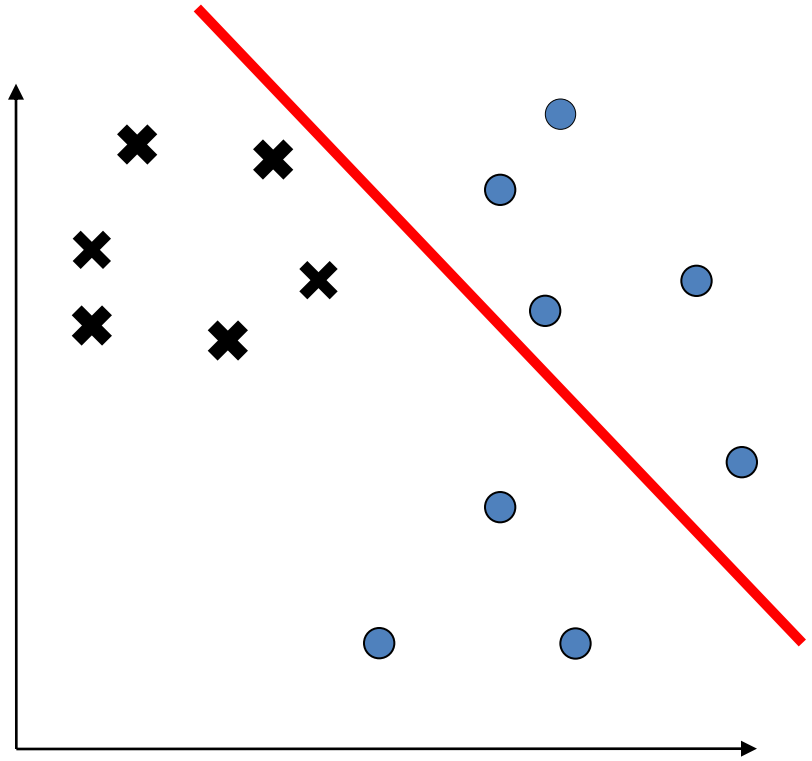
Perceptron

- Proposed by Frank Rosenblatt (1958) (Widrow and Hoff proposed **adaline** at same time)
- Schematic representation



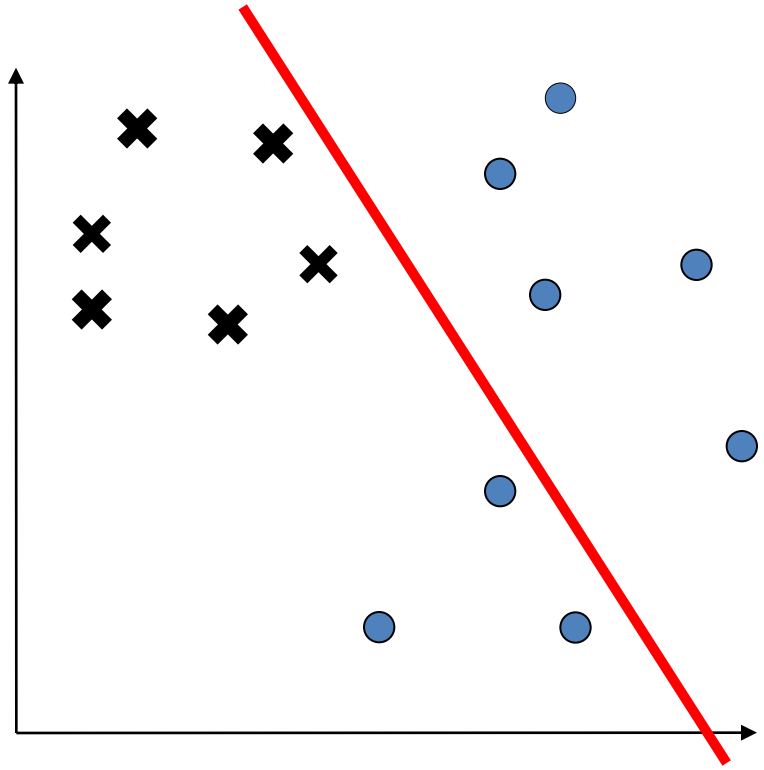
$$\text{if} \quad \left(\sum_{i=1}^{M} x_i w_i \right) \quad > \quad t \qquad \text{then } output = 1, \text{ else } output = 0$$





Is this a good decision boundary?

$$\textit{if} \quad \left(\sum_{i=1}^{M} x_i w_i \right) \quad > \quad t \qquad \text{then } output = 1, \text{ else } output = 0$$

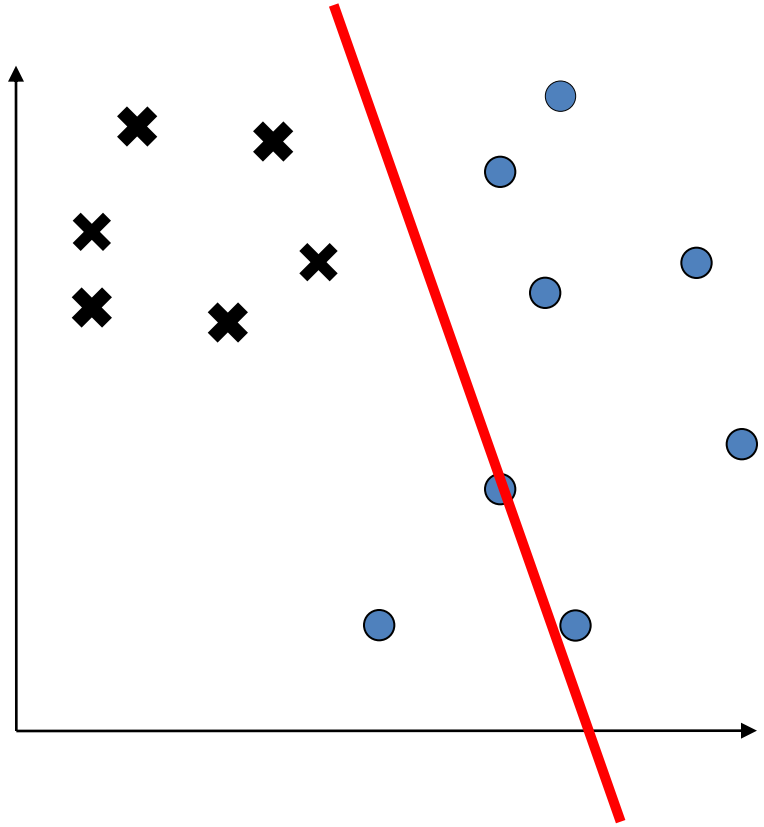


$w_1 = 1.0$

$w_2 = 0.2$

$t = 0.05$

$$\textit{if} \quad \left(\sum_{i=1}^{M} x_i w_i \right) \quad > \quad t \qquad \text{then } output = 1, \text{ else } output = 0$$

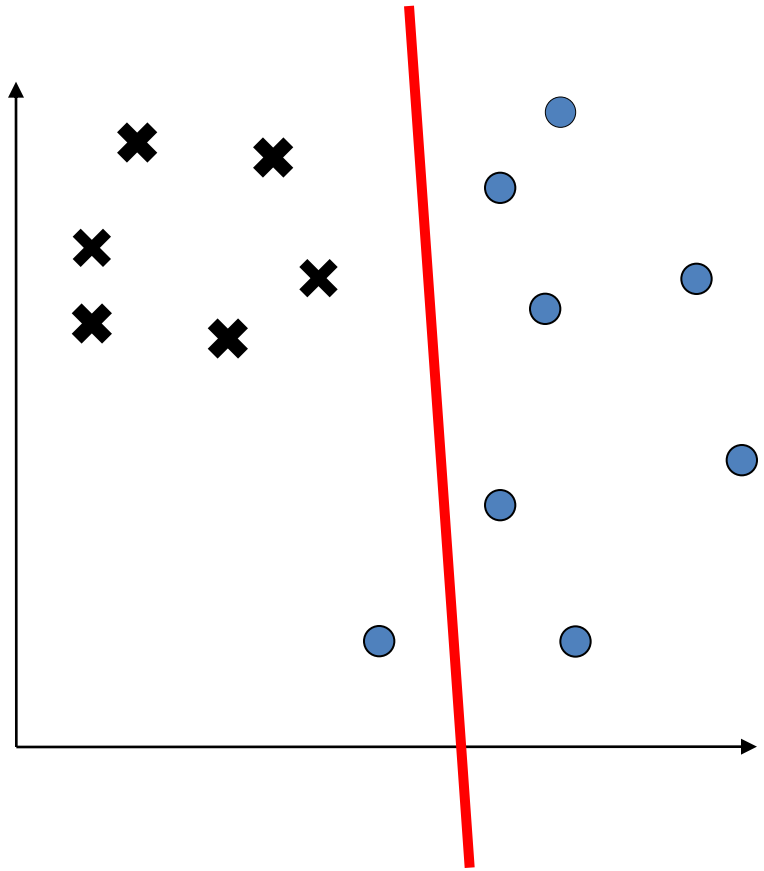


$w_1 = 2.1$

$w_2 = 0.2$

$t = 0.05$

$$\textit{if} \quad \left(\sum_{i=1}^{M} x_i w_i \right) \quad > \quad t \qquad \text{then } output = 1, \text{ else } output = 0$$

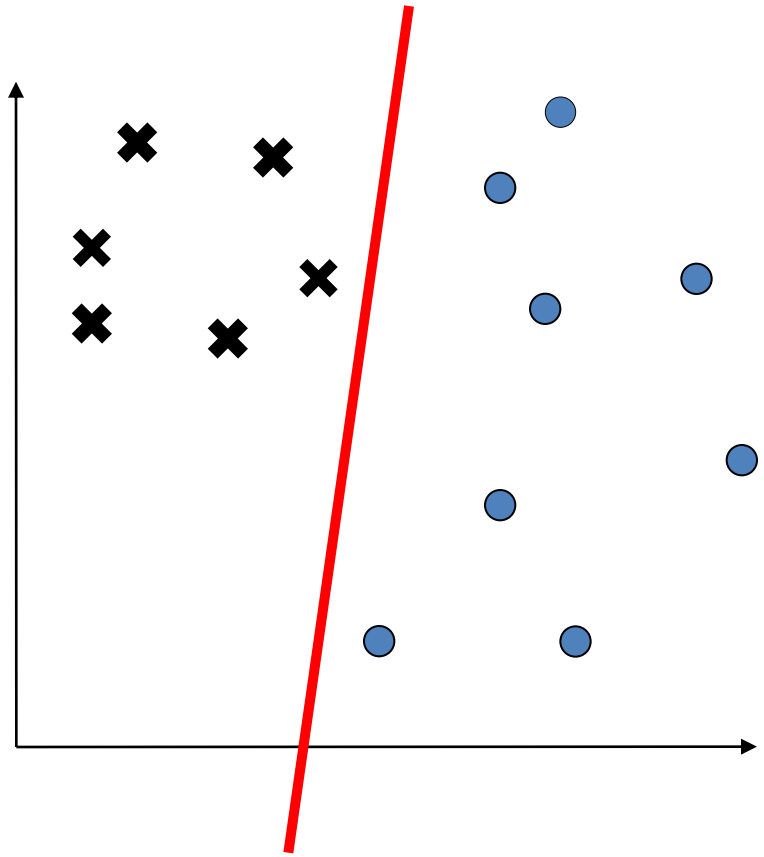


$w_1 = 1.9$

$w_2 = 0.02$

$t = 0.05$

$$\textit{if} \quad \left(\sum_{i=1}^{M} x_i w_i \right) \quad > \quad t \qquad \text{then } output = 1, \text{ else } output = 0$$



$w_1 = -0.8$

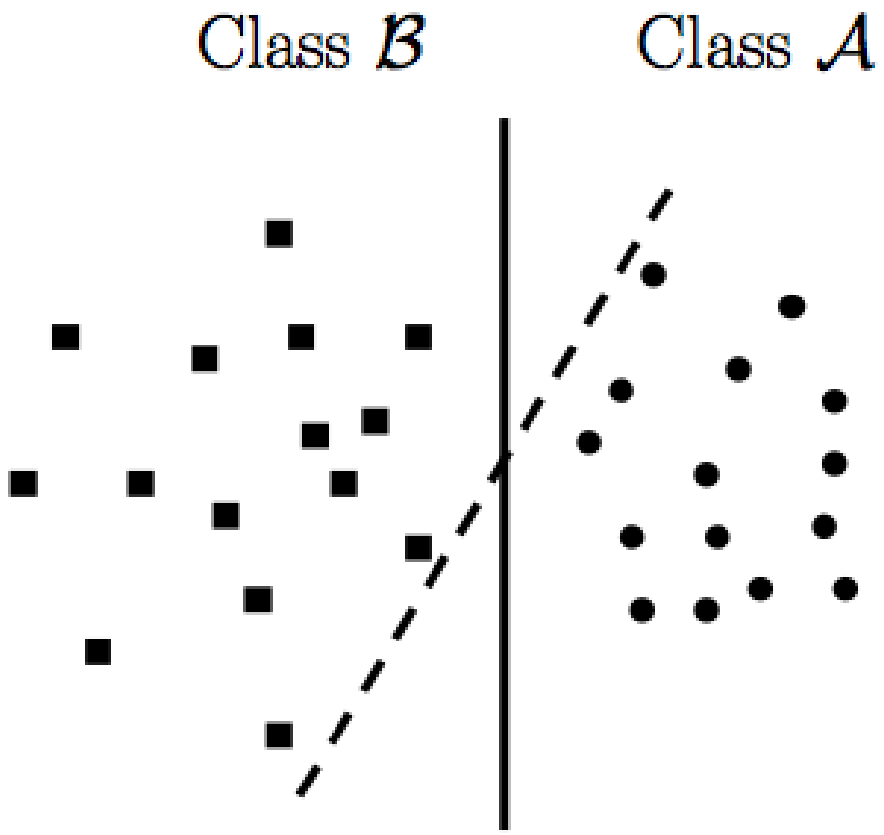
$w_2 = 0.03$

$t = 0.05$

Changing the weights/threshold makes the decision boundary move.

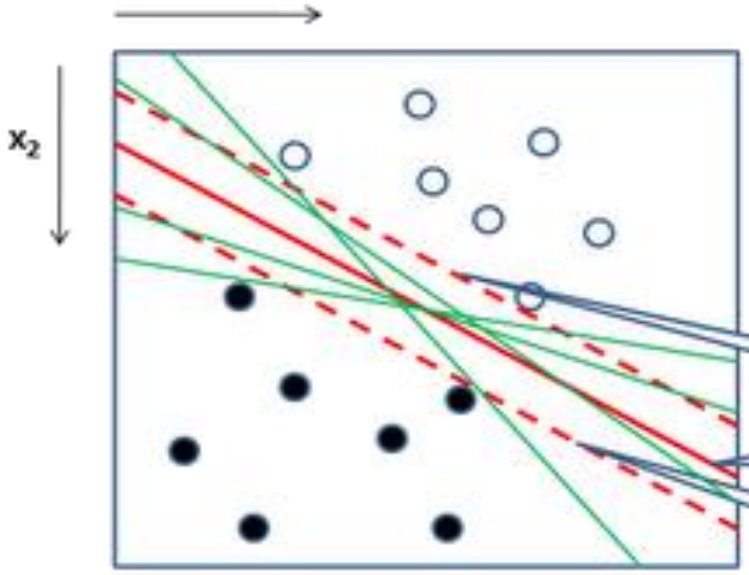
Overview of SVM w.r.t. Perceptron

Perceptron



Which plane is best?

Perceptron VS SVM



All green lines can separate the points
Which one is the best ?
The red line has biggest margin

All +ve samples lies above this line
$y_i = 1$ if $\vec{w}_i . \vec{x} + b > 1$

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

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

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

All -ve samples lies below this line
$y_i = -1$ if $\vec{w}_i . \vec{x} + b < -1$

Margin = $(point_{upperline} - point_{lowerline}) . \vec{w} / |w|$
since $(point_{upperline}) . \vec{w} + b = 1$
And $(point_{lowerline}) . \vec{w} + b = -1$
So margin = $(1 - b + 1 + b) / |w| = 2 / |w|$
Maximize margin is max $2 / |w|^2$ is min $|w|^2 / 2$

Problem:

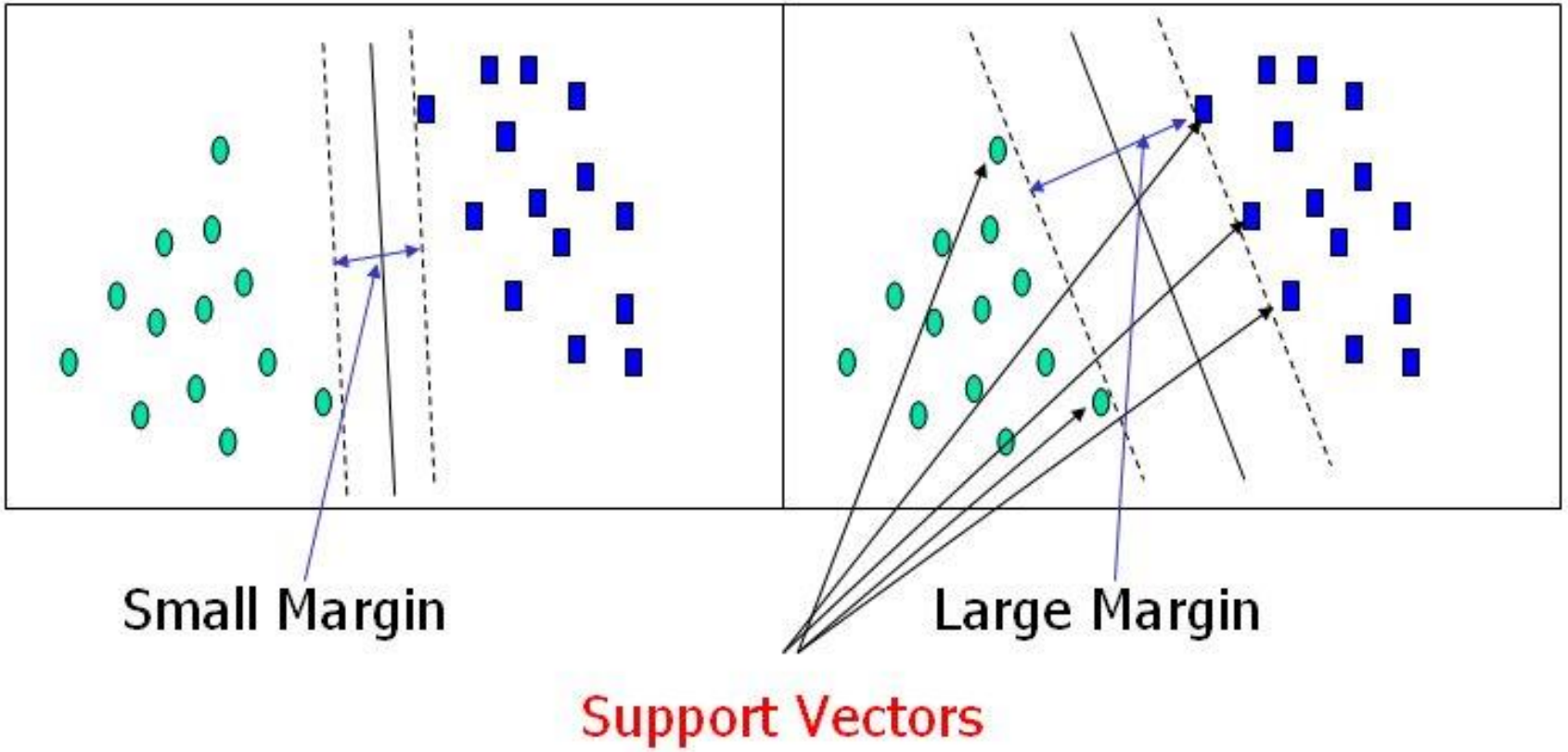
Minimize $(½) \vec{w} . \vec{w}$

With constraint: $y_i(\vec{w}_i . \vec{x} + b) > 1$

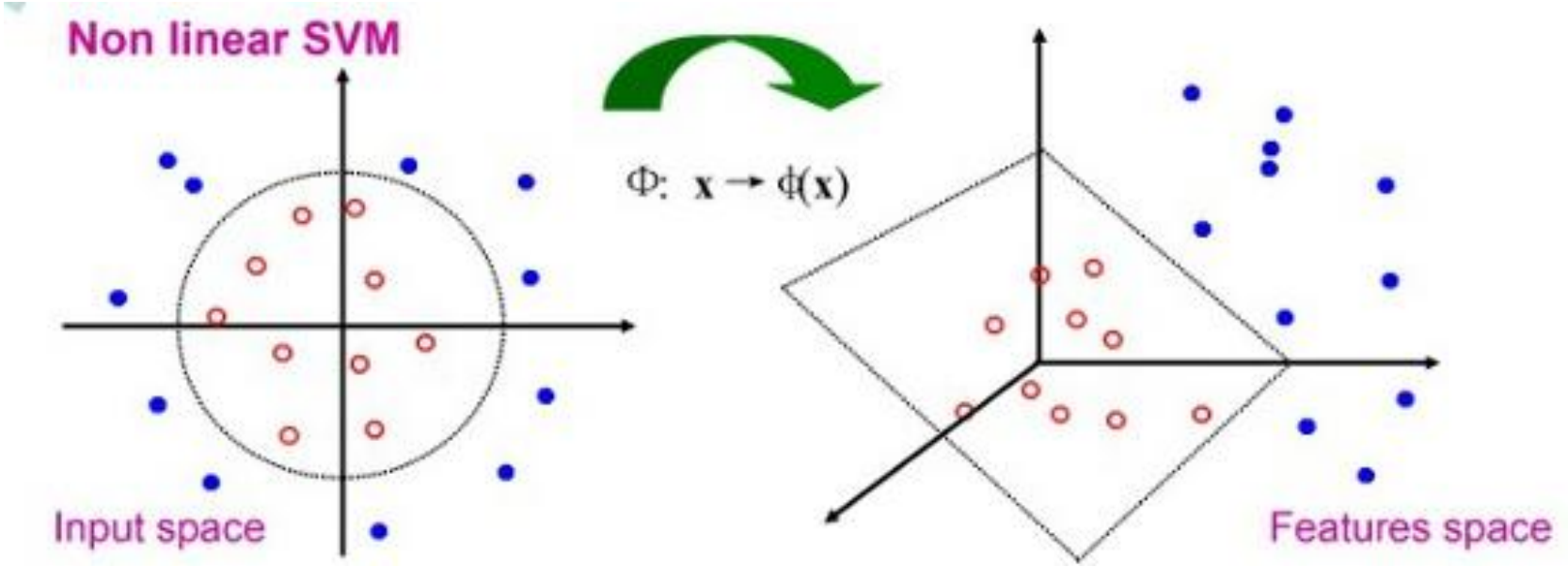
Perceptron VS SVM

- The Perceptron does not try to optimize the separation "distance". As long as it finds a hyperplane that separates the two sets, it is good. SVM on the other hand tries to maximize the "support vector", i.e., the distance between two closest opposite sample points.
- The SVM typically tries to use a "kernel function" to project the sample points to high dimension space to make them linearly separable, while the perceptron assumes the sample points are linearly separable.
- SVM Requires more parameters as compared to
  - choice of kernel
  - selection of kernel parameters
  - selection of the value of the margin parameter

SVM and Margins



SVM for Nonlinear Data



Resources: Datasets

- UCI Repository: http://www.ics.uci.edu/~mlearn/MLRepository.html
- UCI KDD Archive: http://kdd.ics.uci.edu/summary.data.application.html
- Statlib: http://lib.stat.cmu.edu/
- Delve: http://www.cs.utoronto.ca/~delve/

Resources: Journals

- Journal of Machine Learning Research www.jmlr.org
- Machine Learning
- IEEE Transactions on Neural Networks
- IEEE Transactions on Pattern Analysis and Machine Intelligence
- Annals of Statistics
- Journal of the American Statistical Association
- …

Resources: Conferences

- International Conference on Machine Learning (ICML)
- European Conference on Machine Learning (ECML)
- Neural Information Processing Systems (NIPS)
- Computational Learning
- International Joint Conference on Artificial Intelligence (IJCAI)
- ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)
- IEEE Int. Conf. on Data Mining (ICDM)

Acknowledgements

- Emily Fox & Carlos Guestrin, Machine Learning Courses, University of Washington, Coursera
- Introduction to Machine Learning, Alphaydin
- Statistical Pattern Recognition: A Review – A.K Jain et al., PAMI (22) 2000
- Pattern Recognition and Analysis Course – A.K. Jain, MSU
- *Pattern Classification"* by Duda et al., John Wiley & Sons.

Material in these slides has been taken from, the following resources

THANK YOU