Birla Institute of Applied Sciences
बिरला इंस्टिट्यूट ऑफ़ अप्लाइड साइंसेस
Bhimtal, Distt: Nainital, Uttarakhand- 263136

Pattern Recognition

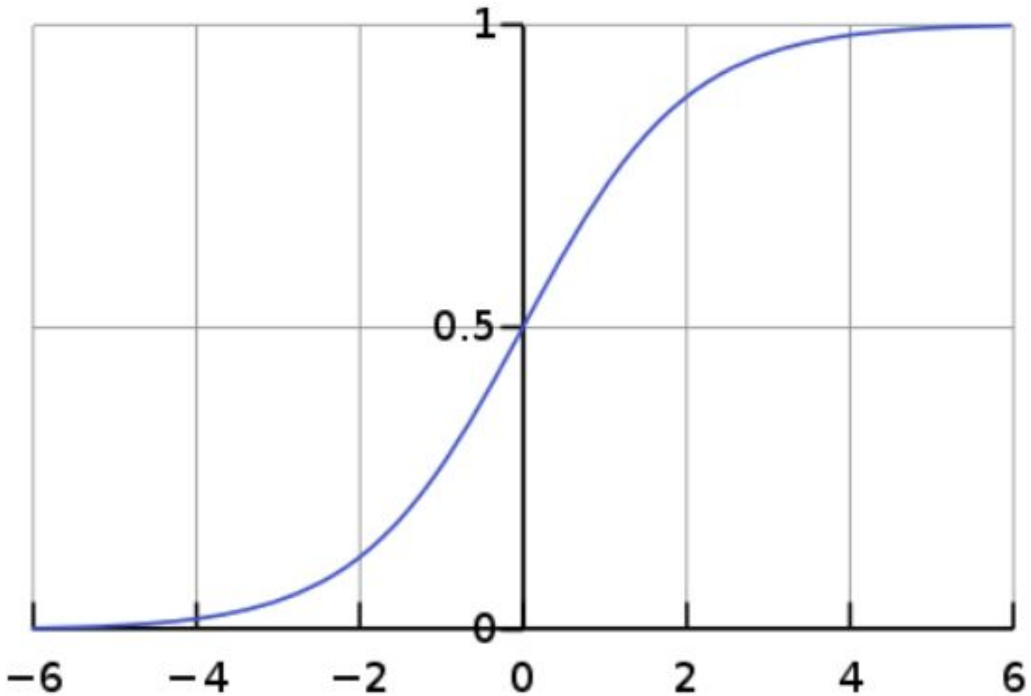
- S. S. Samant

Logistic Regression

The function is called logistic function or sigmoid function.

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

Here, $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots = \beta^T \mathbf{x}$



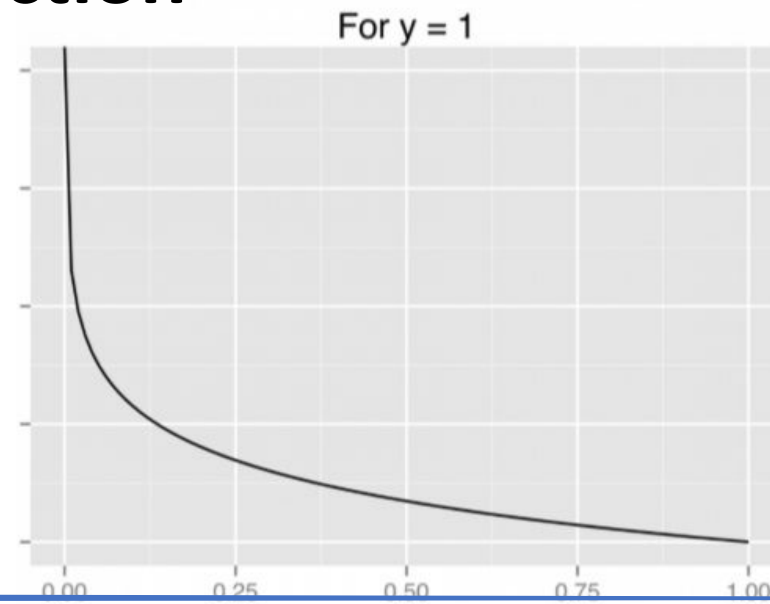
$y=1$ when $\beta^T \mathbf{x} \geq \mathbf{0}$

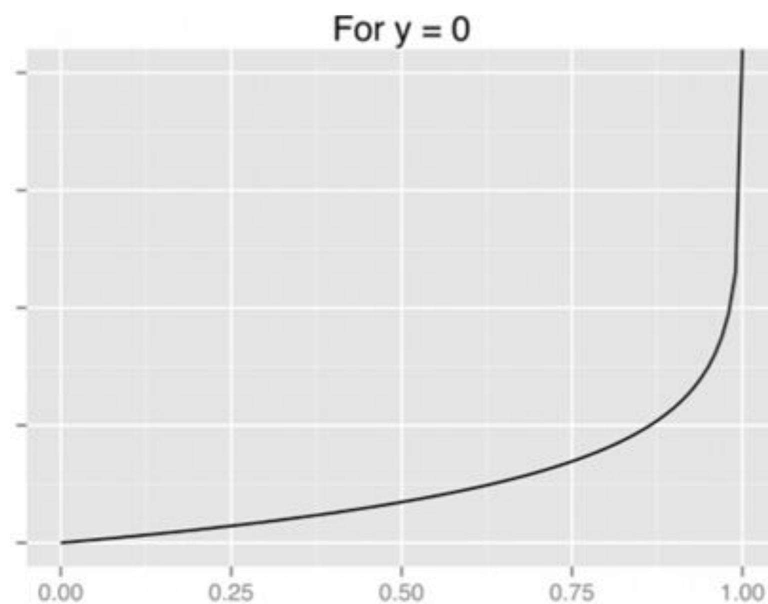
$y=0$ when $\beta^T \mathbf{x} < \mathbf{0}$

Logistic Regression – Cost function

Cost function =

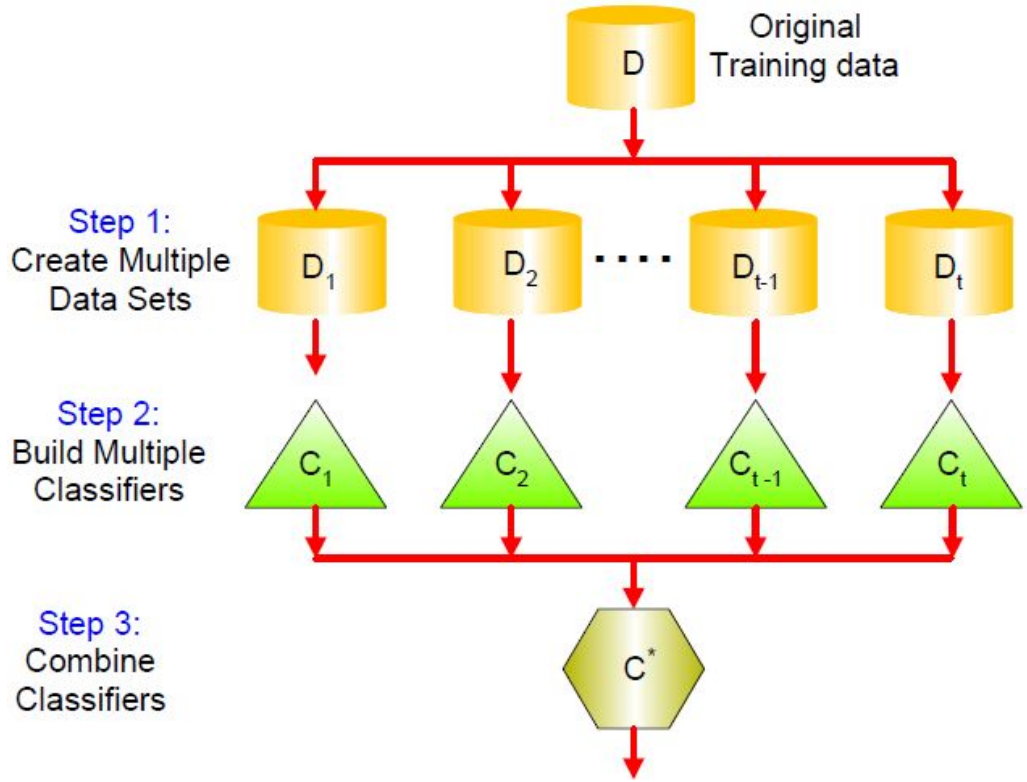
$$
\begin{cases}
-\log(\sigma(t)) & \text{if } y = 1 \\
-\log(1-\sigma(t)) & \text{if } y = 0
\end{cases}
$$





Ensemble methods

- Construct a set of classifiers from the training data
- Predict class label of previously unseen records by aggregating predictions made by multiple classifiers



Ensemble methods – why they work?

- Suppose there are 5 base classifiers
  - Each classifier has error rate, $\varepsilon = 0.35$
  - Assume classifiers are independent

  What is the probability that the ensemble classifier makes a wrong prediction?

Ensemble methods – why they work?

- Suppose there are 5 base classifiers
  - Each classifier has error rate, ε = 0.35
  - Assume classifiers are independent

  What is the probability that the ensemble classifier makes a wrong prediction?

$$5C_3(.35)^3(.65)^2+5C_4(.35)^4(.65)+5C_4\,(.35)^5$$

$$= 0.24$$

Ensemble methods – why they work?

- Suppose there are 25 base classifiers
  - Each classifier has error rate, $\varepsilon = 0.35$
  - Assume classifiers are independent

  What is the probability that the ensemble classifier makes a wrong prediction?

Ensemble methods – why they work?

- Suppose there are 25 base classifiers
  - Each classifier has error rate, ε = 0.35
  - Assume classifiers are independent

  What is the probability that the ensemble classifier makes a wrong prediction?

$$\sum_{i=13}^{25} \binom{25}{i} \varepsilon^i (1-\varepsilon)^{25-i} = 0.06$$

Generating Ensemble of classifiers

- Bagging (**b**ootstrap **agg**regat**ing**)
- Boosting

Generating Ensemble of classifiers

- Bagging (**b**ootstrap **agg**regat**ing**) – sampling with replacement

- Boosting – boosting weight of wrongly classified samples

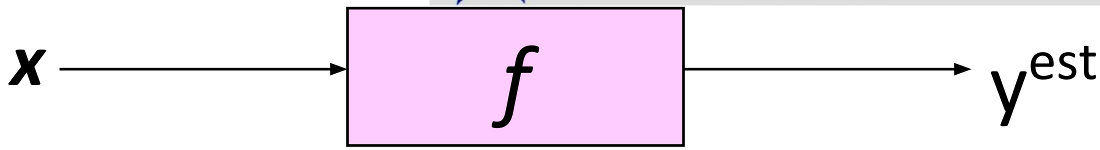
Random Forest classifier

- Bagging is performed - repeatedly select a random sample with replacement of the training set and fits trees to these samples:
- At each candidate split in the learning process, a random subset of the features is selected
- *Combine* results of individual classifiers built on the samples and subset features
  - Combining classifiers? Ex. voting

Support Vector Machine (SVM)

- SVM was first introduced in 1992
- SVM becomes popular because of its success in handwritten digit recognition
- SVM is now regarded as an important example of *kernel methods*, one of the key area in machine learning

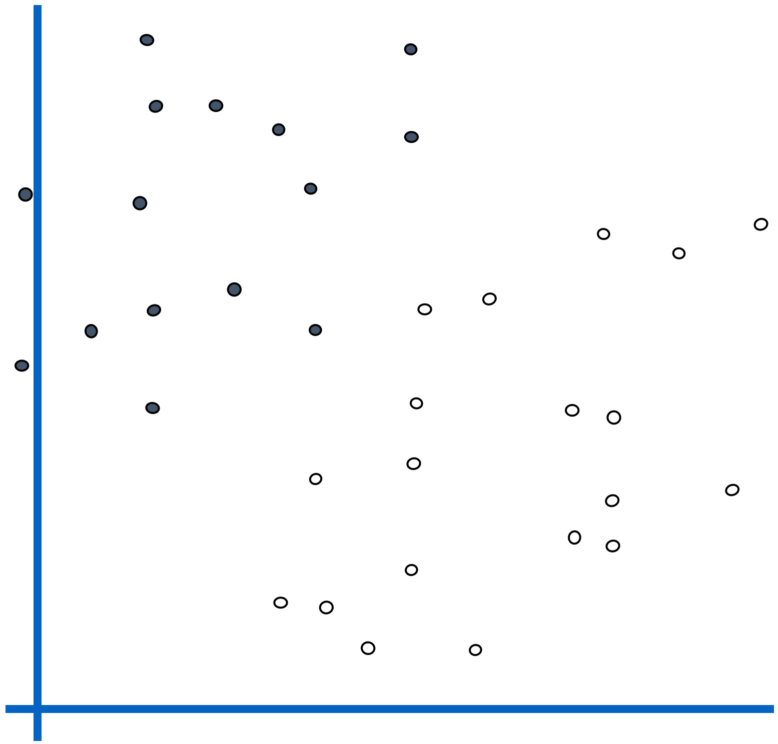
Linear Classifiers



$$f(\mathbf{x},\mathbf{w},b) = \text{sign}(\mathbf{w}\cdot\mathbf{x} - b)$$

w: weight vector
x: data vector

- ● denotes +1
- ○ denotes -1



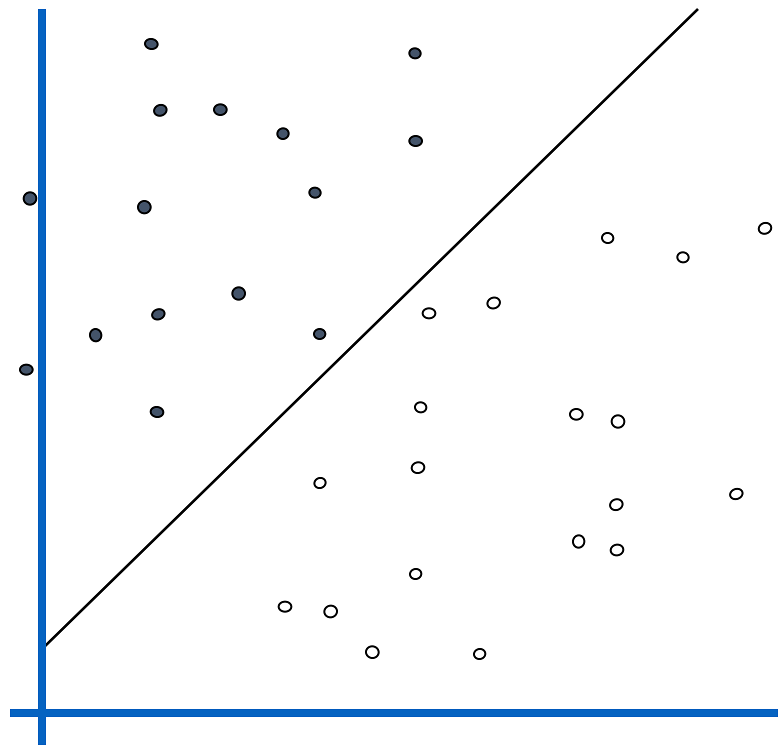
How would you classify this data?

Linear Classifiers

$$x \longrightarrow \boxed{f} \xrightarrow{} y^{est}, \quad \alpha \downarrow f$$

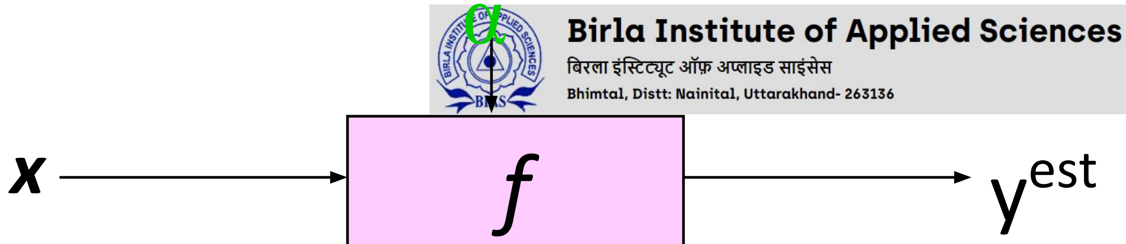
$$f(\mathbf{x}, \mathbf{w}, b) = sign(\mathbf{w} \cdot \mathbf{x} - b)$$

- denotes +1
- denotes -1



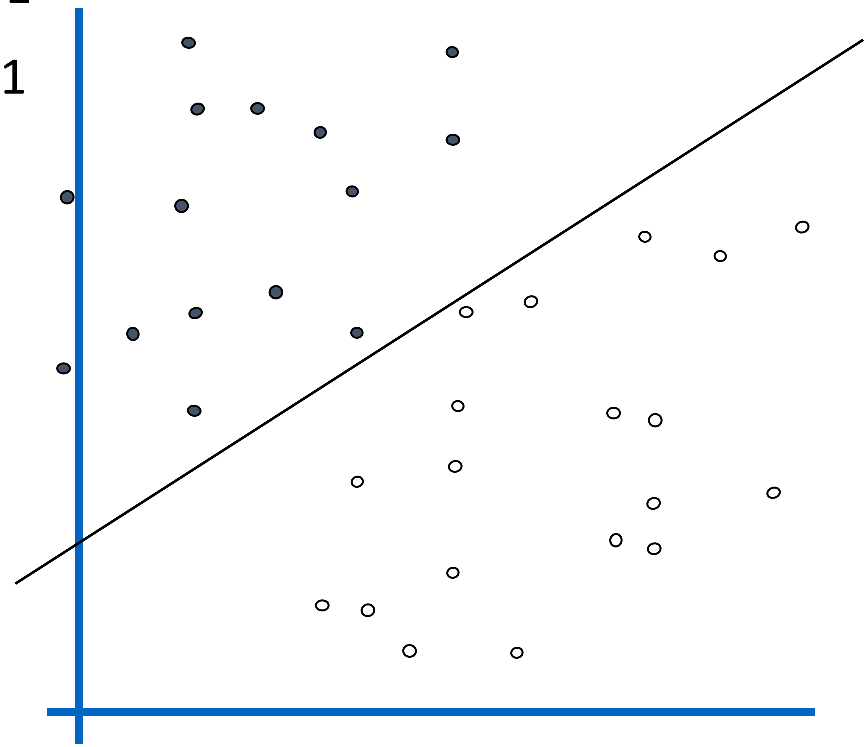
How would you classify this data?

Linear Classifiers



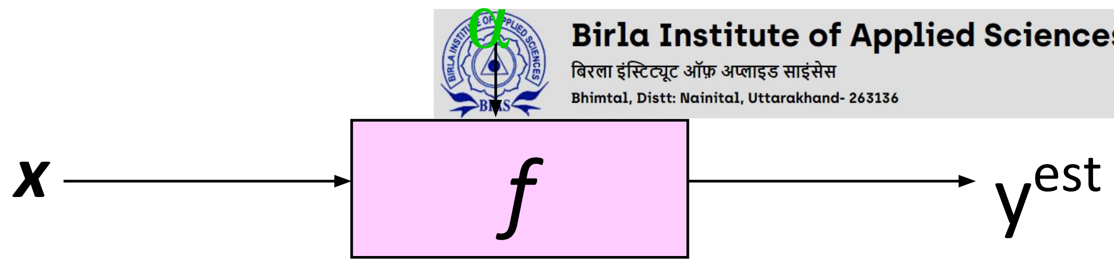
$$f(\mathbf{x}, \mathbf{w}, b) = sign(\mathbf{w} \cdot \mathbf{x} - b)$$

- ● denotes +1
- ○ denotes -1



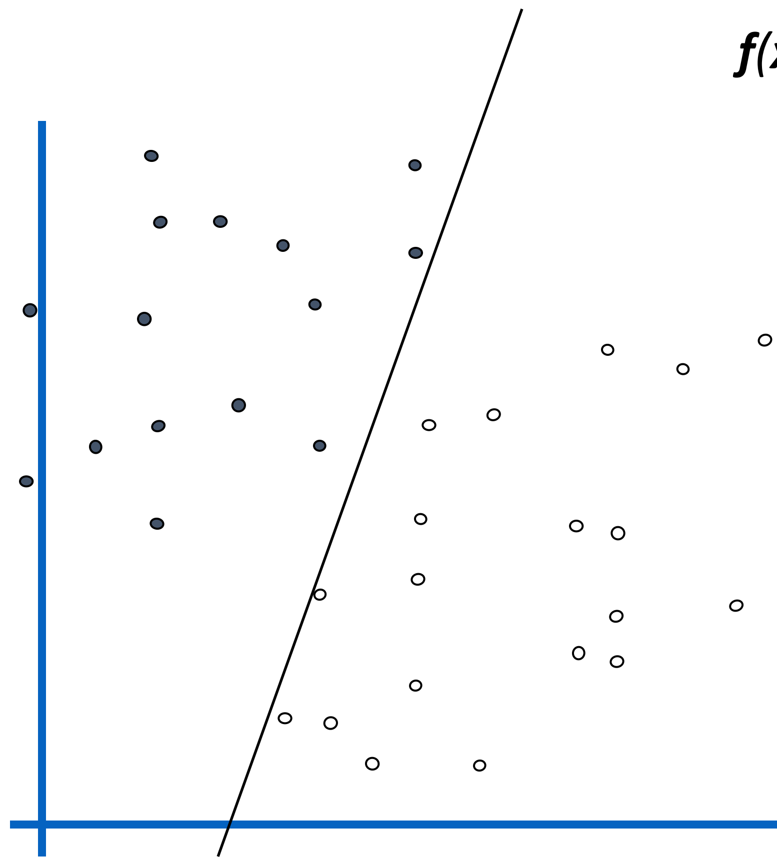
How would you classify this data?

Linear Classifiers



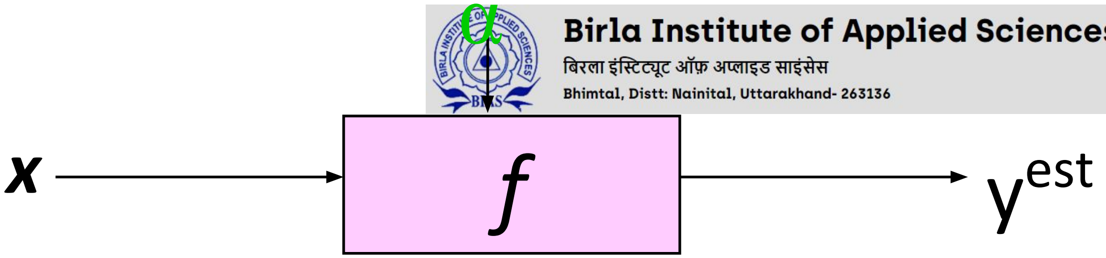
$$f(\boldsymbol{x},\boldsymbol{w},b) = \mathrm{sign}(\boldsymbol{w}.\ \boldsymbol{x} - b)$$

- ● denotes +1
- ○ denotes -1

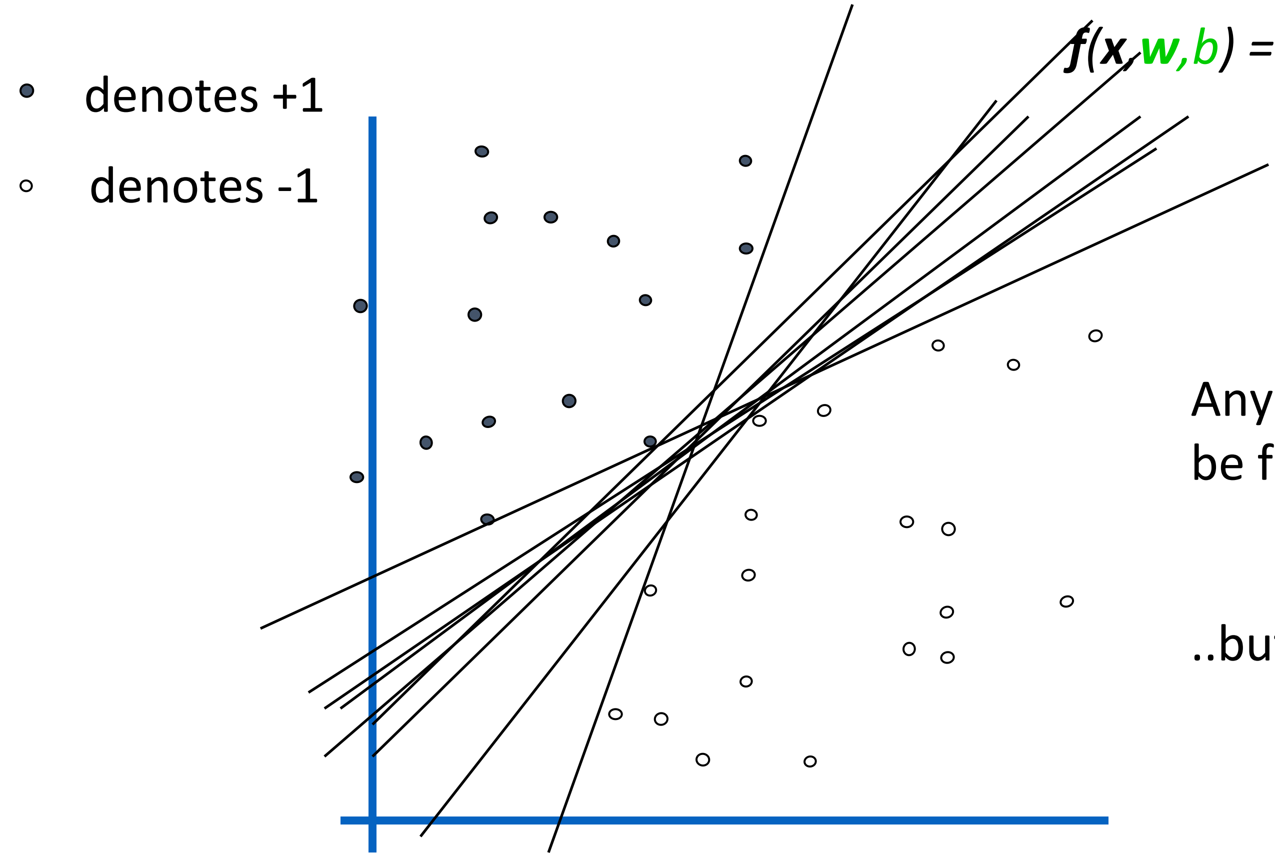


How would you classify this data?

Linear Classifiers



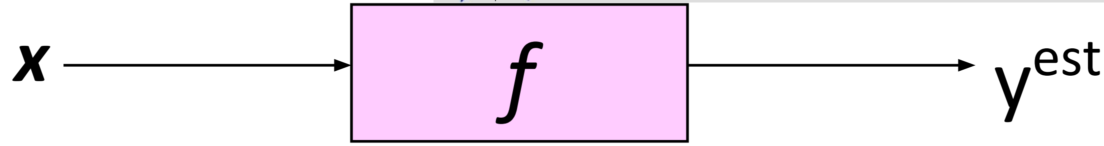
$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w} \cdot \mathbf{x} - b)$



Any of these would be fine..

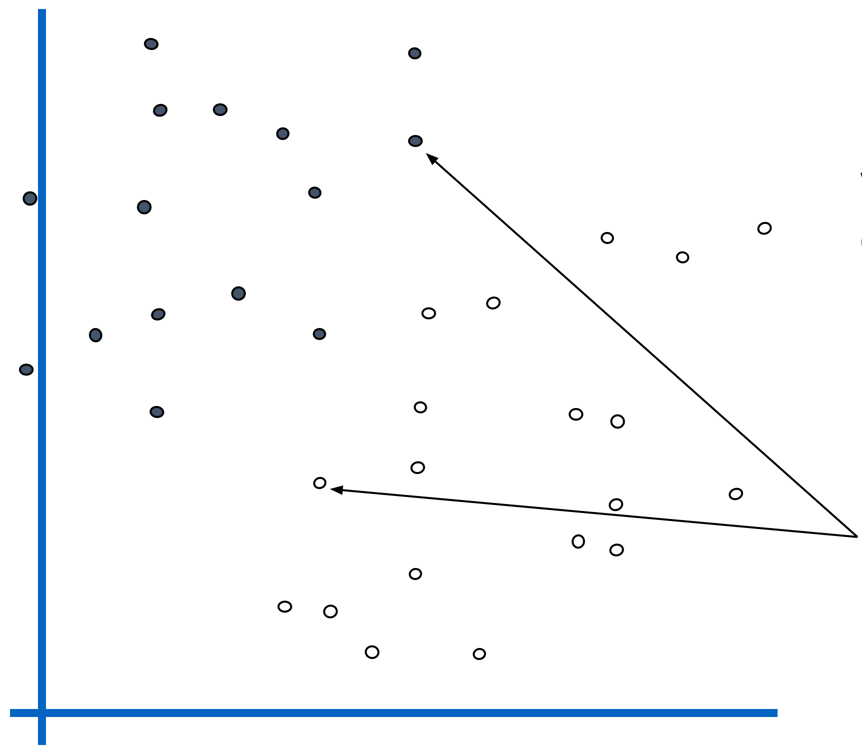
..but which is best?

Classifier Margin



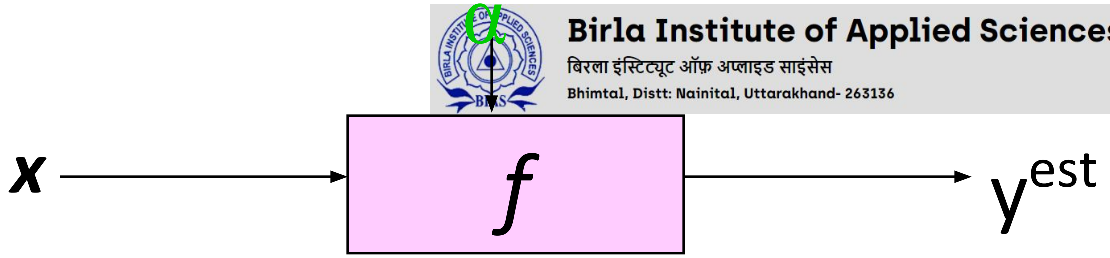
$$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}\cdot \mathbf{x} - b)$$

- ● denotes +1
- ○ denotes -1



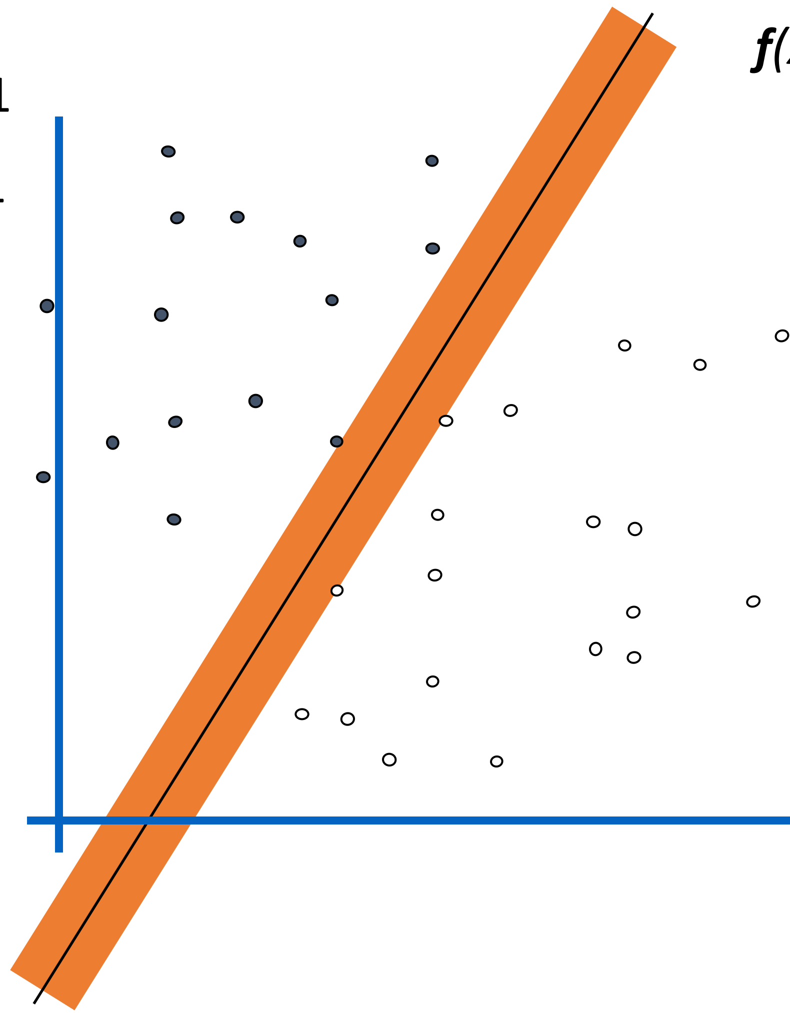
Define the margin of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.

Maximum Margin



• denotes +1

◦ denotes -1



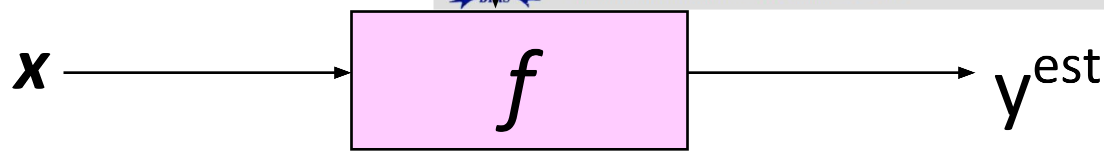
$$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} - b)$$

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

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

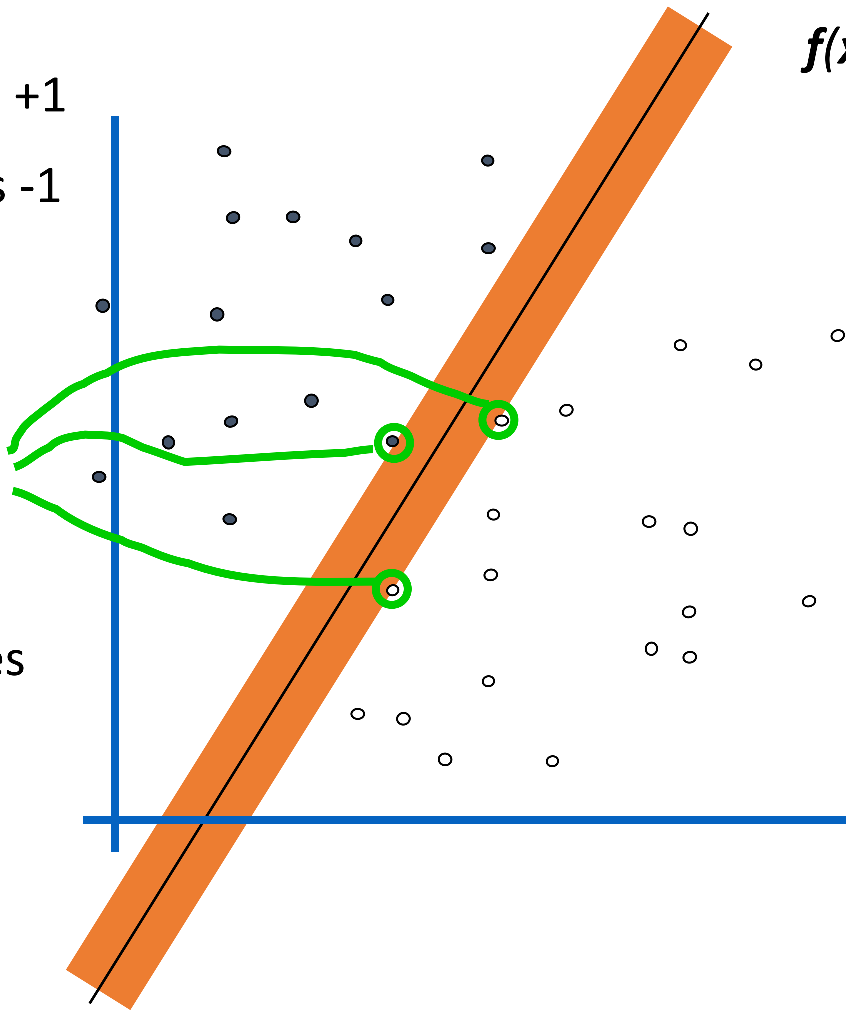
Linear SVM

Maximum Margin



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

- • denotes +1
- ◦ denotes -1



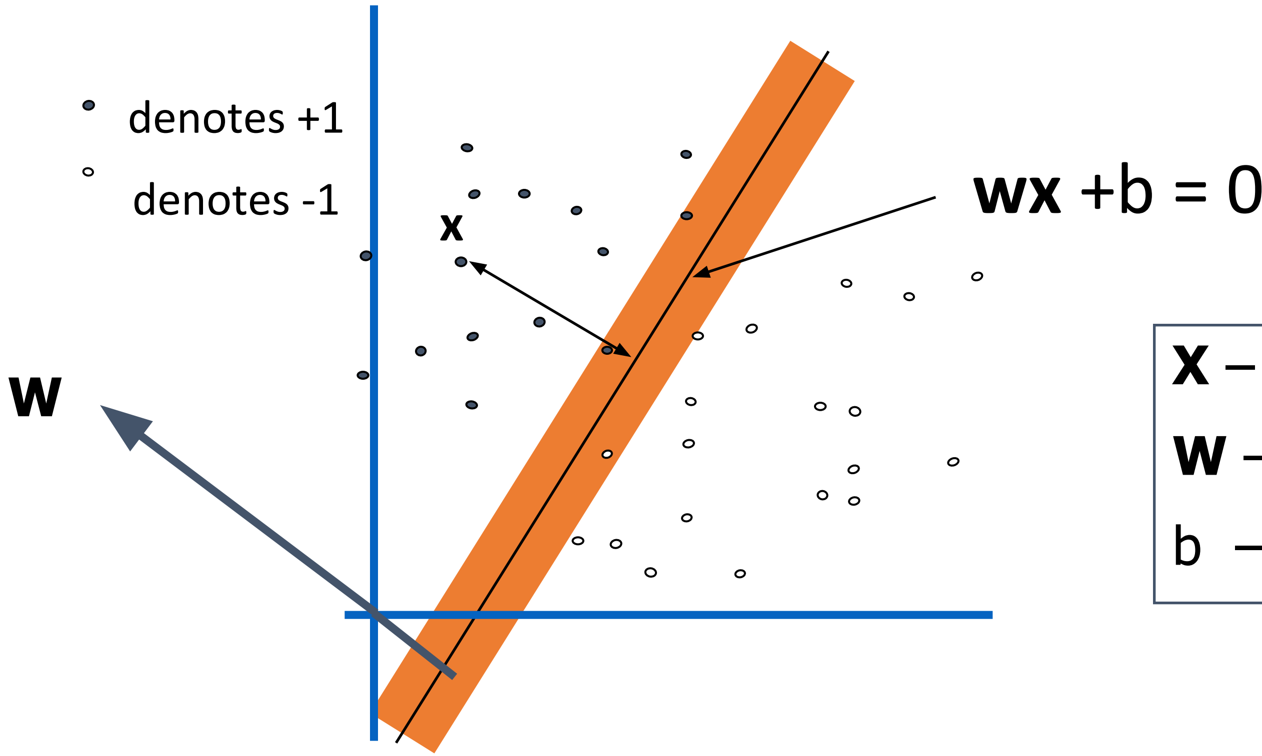
Support Vectors are those datapoints that the margin pushes up against

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

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

Linear SVM

How to calculate the distance from a point to a line?



X – Vector

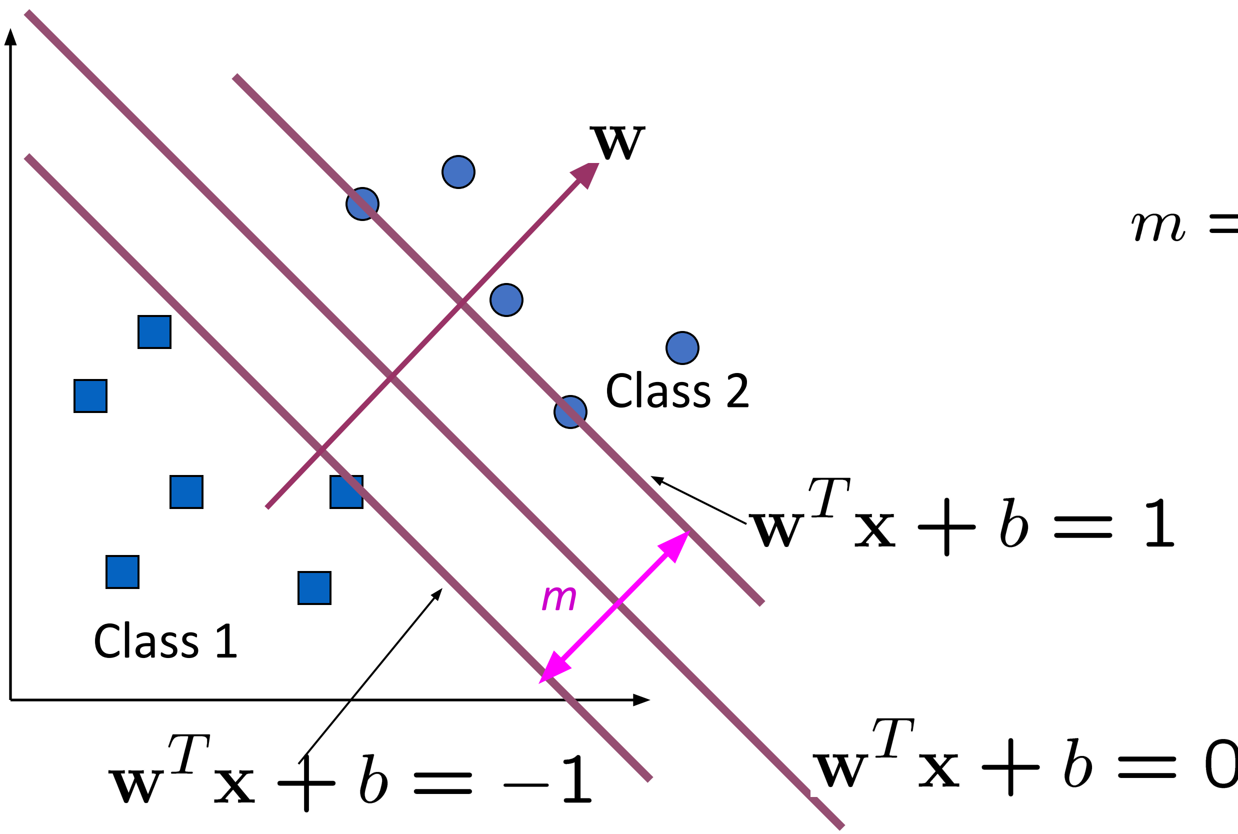
W – Normal Vector

b – Scale Value

- In our case, $w_1*x_1+w_2*x_2+b=0$,
- thus, $\mathbf{w}=(w_1,w_2)$, $\mathbf{x}=(x_1,x_2)$

Large-margin Decision Boundary

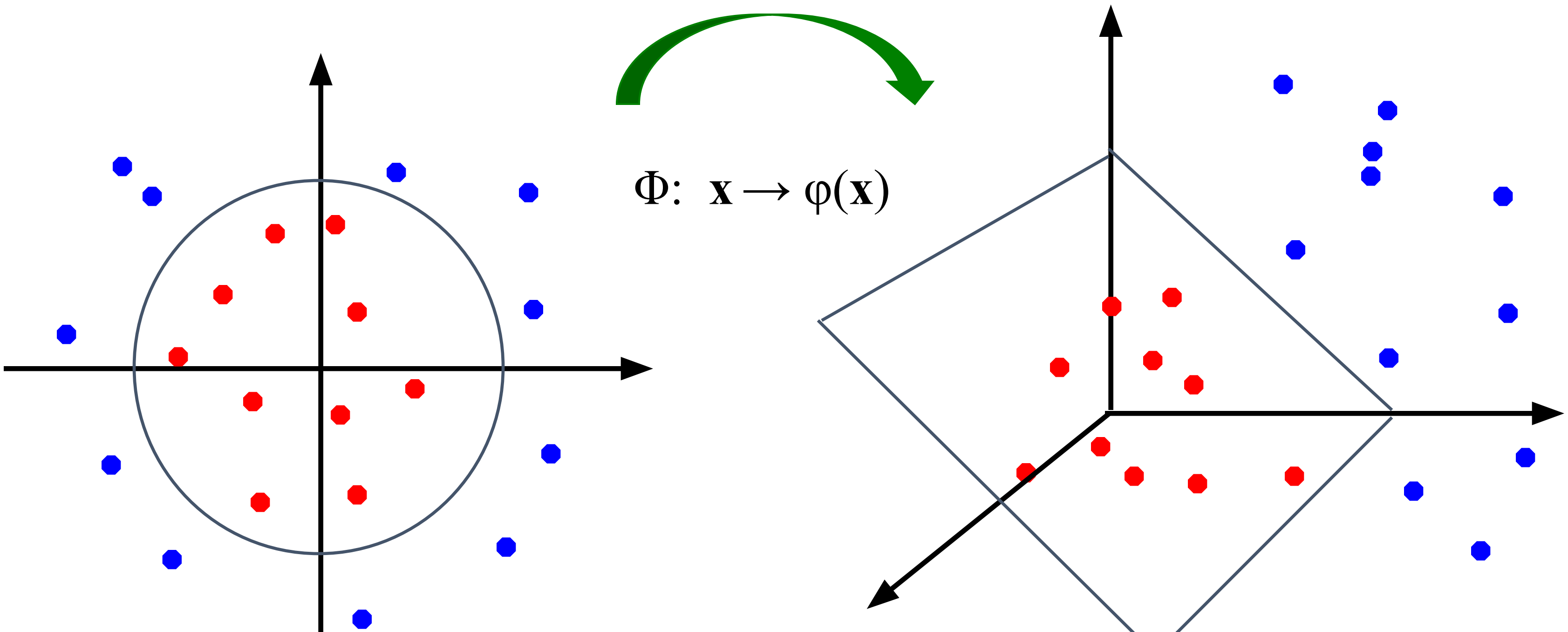
- The decision boundary should be as far away from the data of both classes as possible
  - We should maximize the margin, *m*
  - Distance between the origin and the line $\mathbf{w}^t\mathbf{x}=-b$ is $b/||\mathbf{w}||$



$$m = \frac{2}{||\mathbf{w}||}$$

Non-linear SVMs: Feature spaces

- General idea: the original input space can always be mapped to some higher-dimensional feature space where the training set is separable:



$$\Phi : \mathbb{R}^2 \to \mathbb{R}^3$$
$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}\, x_1 x_2, x_2^2)$$

Thank You!