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

Pattern Recognition

- S. S. Samant

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

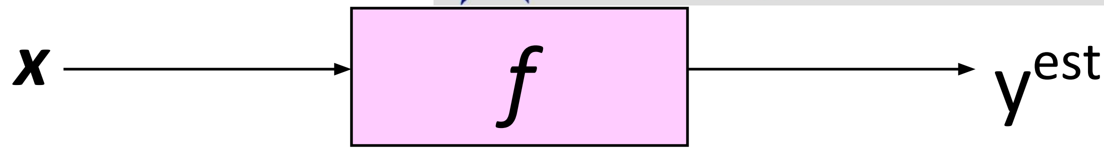
Other Types of Ensembles

- Extremely Randomized Trees Classifier(Extra Trees Classifier)
- Boosting – AdaBoost, Gradient Boosting
- Stacking

Support Vector Machine (SVM)

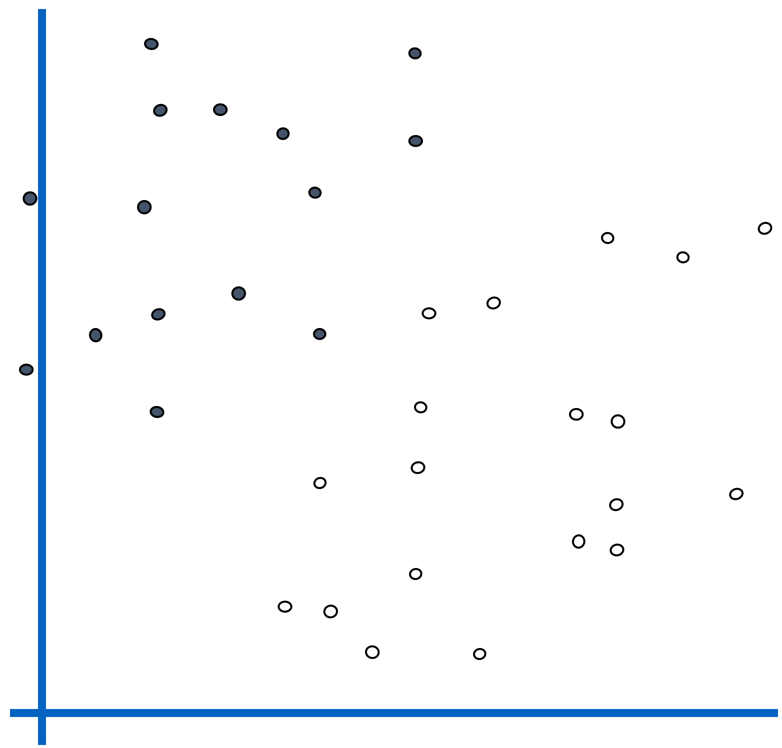
- SVM was first introduced in 1992
- SVM becomes popular because of its success in handwritten digit recognition
- SVM is now regarded as an important example of *kernel methods*, one of the key area in machine learning

Linear Classifiers



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

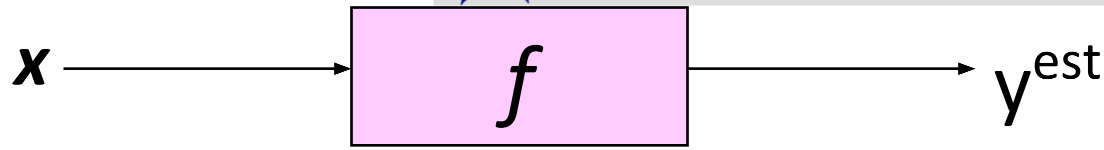
- ● denotes +1
- ○ denotes -1



w: weight vector
x: data vector

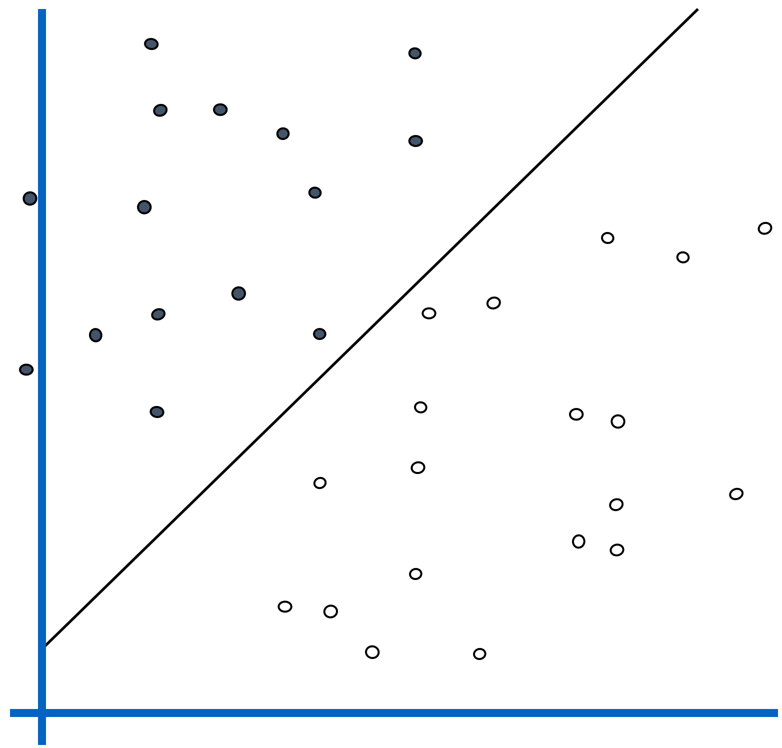
How would you classify this data?

Linear Classifiers



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

- ● denotes +1
- ○ denotes -1



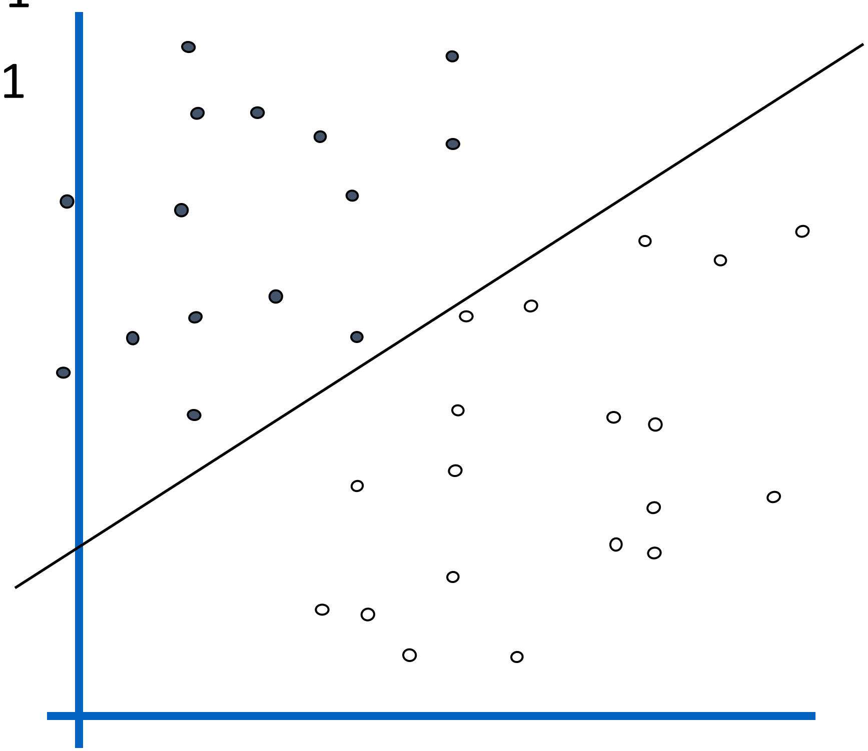
How would you classify this data?

Linear Classifiers

$$\boldsymbol{x} \longrightarrow \boxed{f} \longrightarrow y^{est}$$

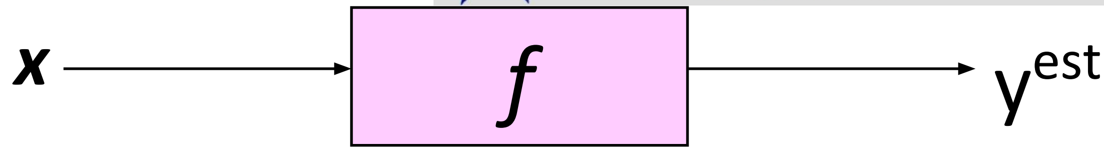
$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}.\ \mathbf{x} + b)$

- ● denotes +1
- ○ denotes -1



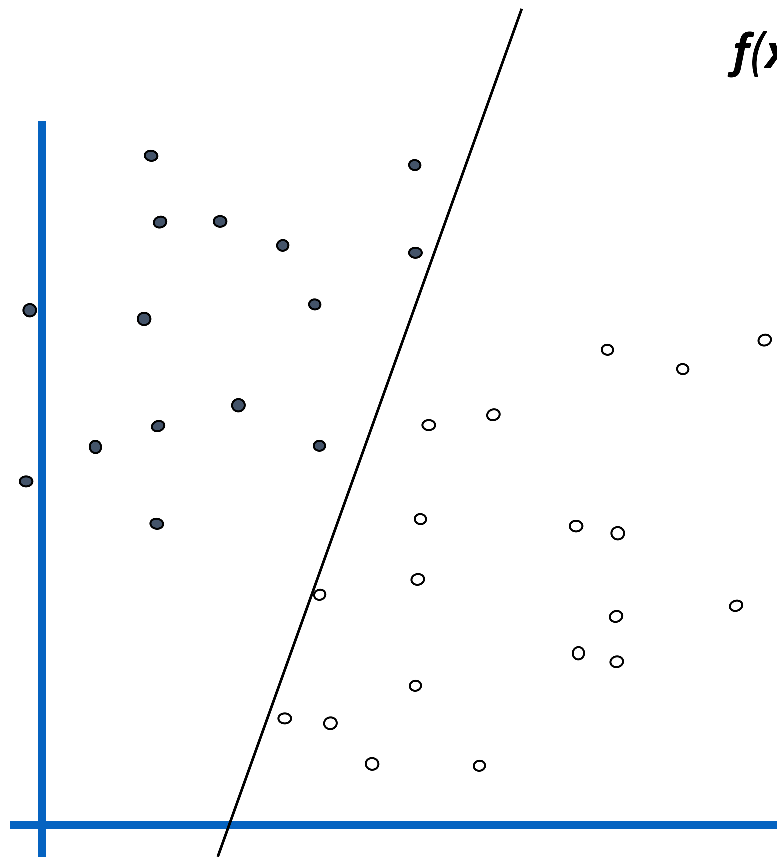
How would you classify this data?

Linear Classifiers



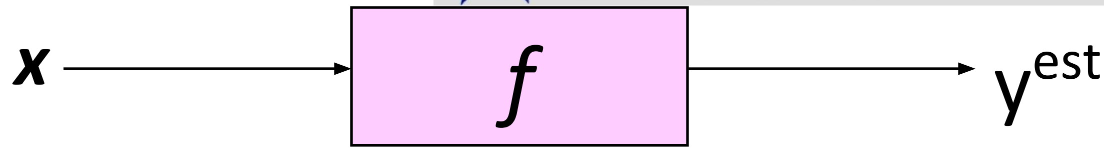
$f(\boldsymbol{x},\boldsymbol{w},b) = sign(\boldsymbol{w}.\ \boldsymbol{x} + b)$

- ● denotes +1
- ○ denotes -1



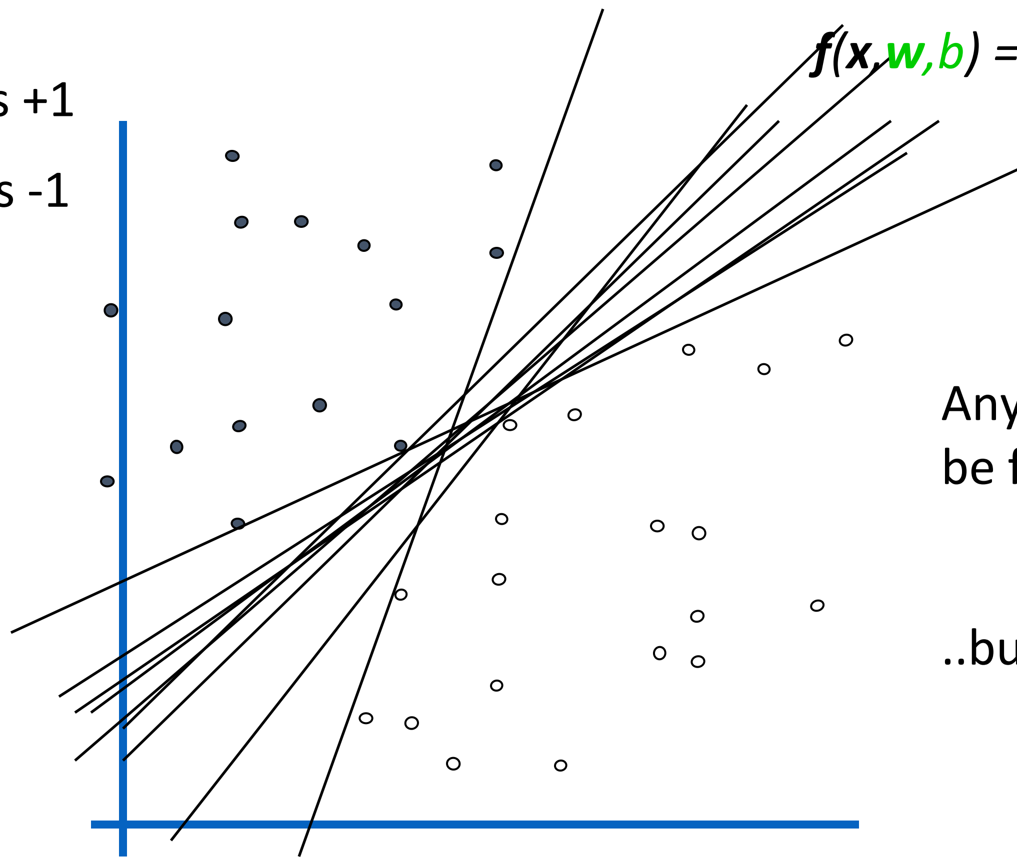
How would you classify this data?

Linear Classifiers



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

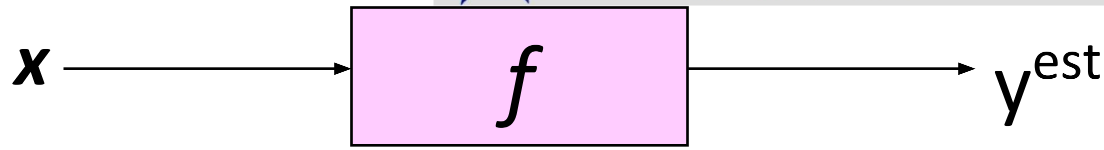
- ● denotes +1
- ○ denotes -1



Any of these would be fine..

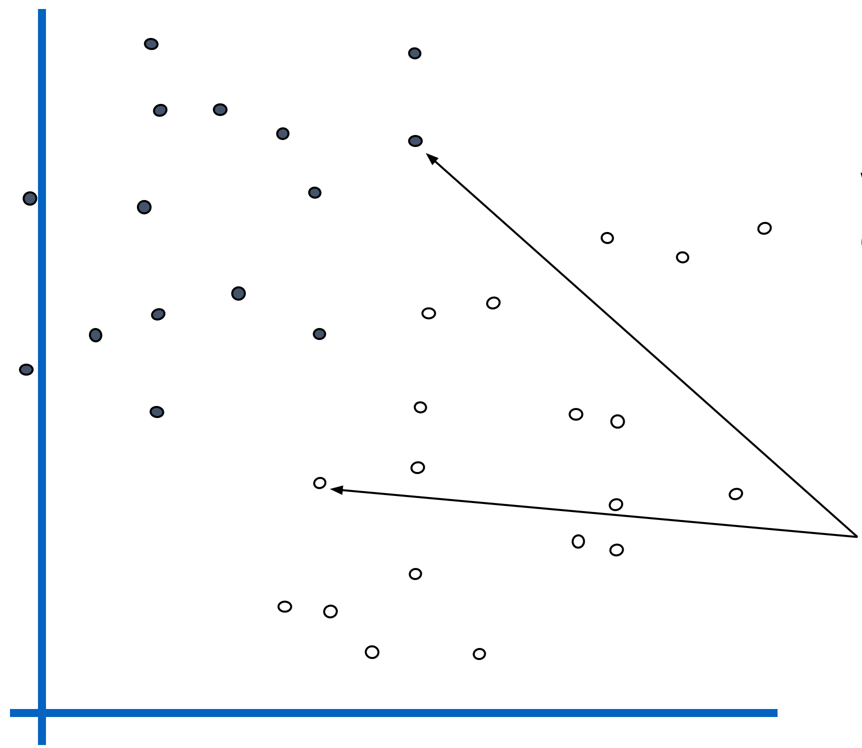
..but which is best?

Classifier Margin



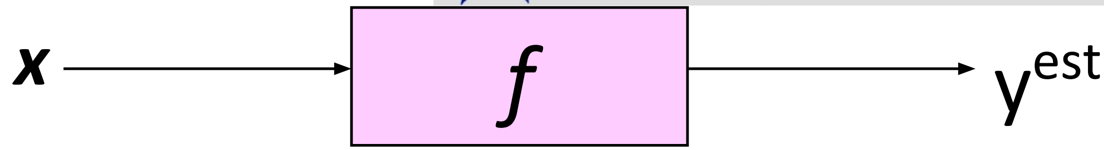
$f(\mathbf{x},\mathbf{w},b) = sign(\mathbf{w}\cdot\mathbf{x} + b)$

- ● denotes +1
- ○ denotes -1



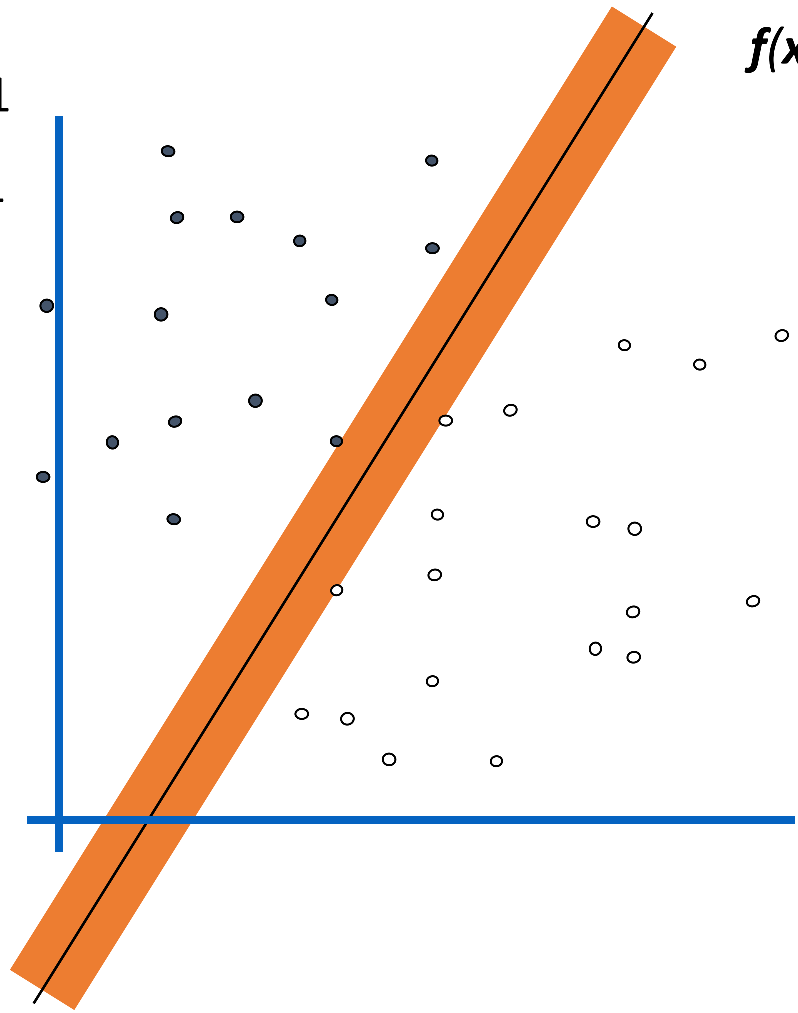
Define the margin of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.

Maximum Margin



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

- ● denotes +1
- ○ denotes -1



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

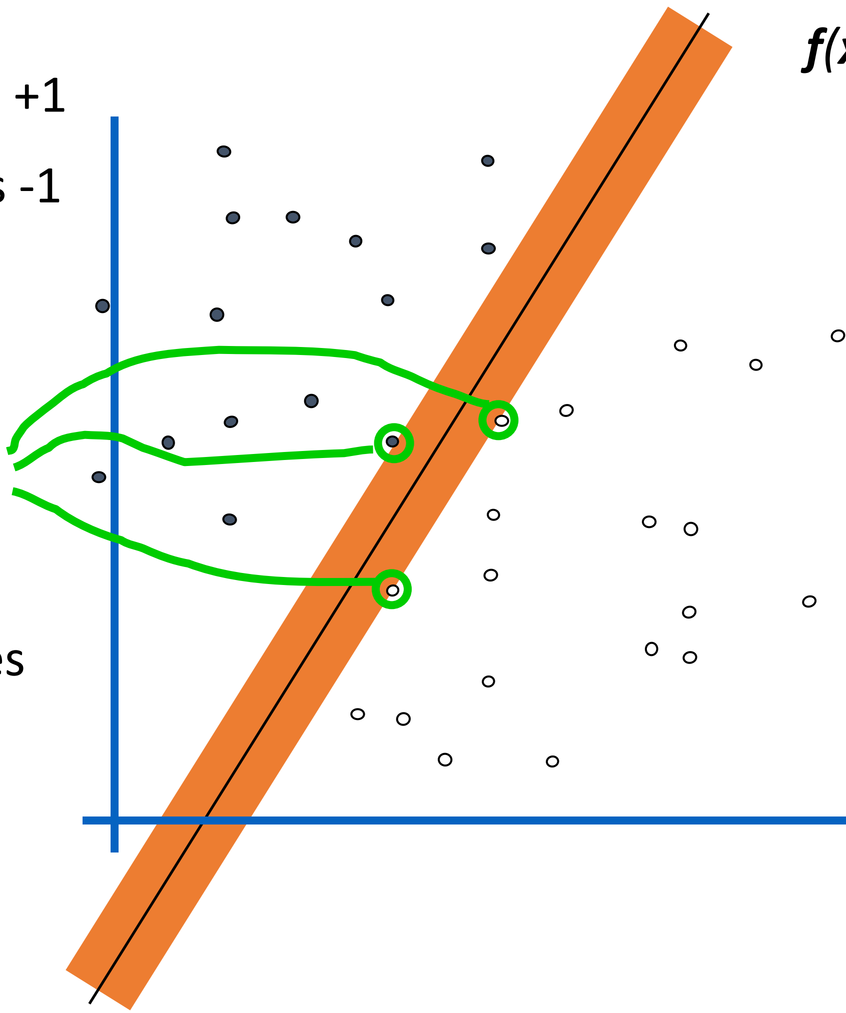
This is the simplest kind of SVM (Called an LSVM)

Linear SVM

Maximum Margin

$$\boldsymbol{x} \longrightarrow \boxed{f} \longrightarrow y^{est}$$

- denotes +1
- ◦ denotes -1



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

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.

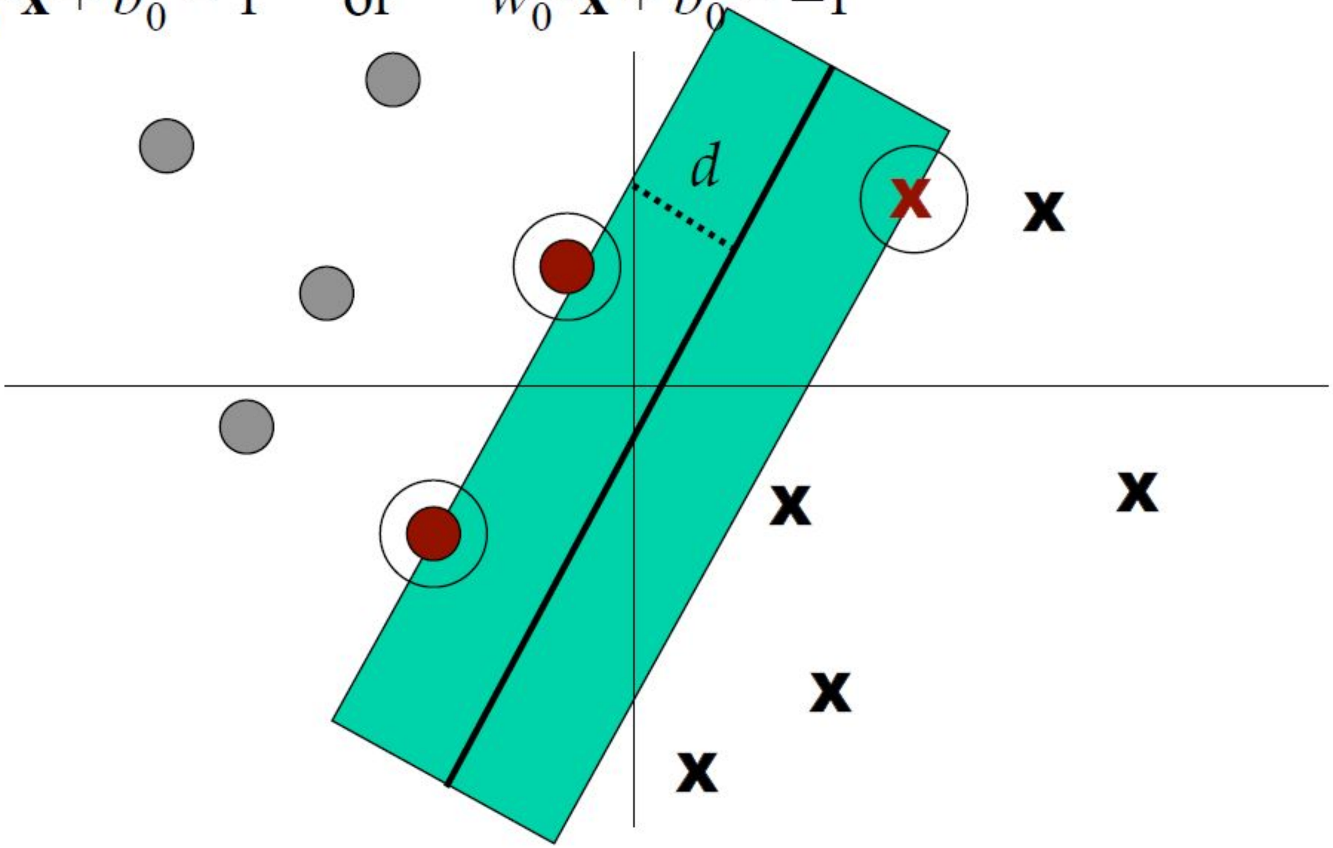
This is the simplest kind of SVM (Called an LSVM)

Linear SVM

Margin

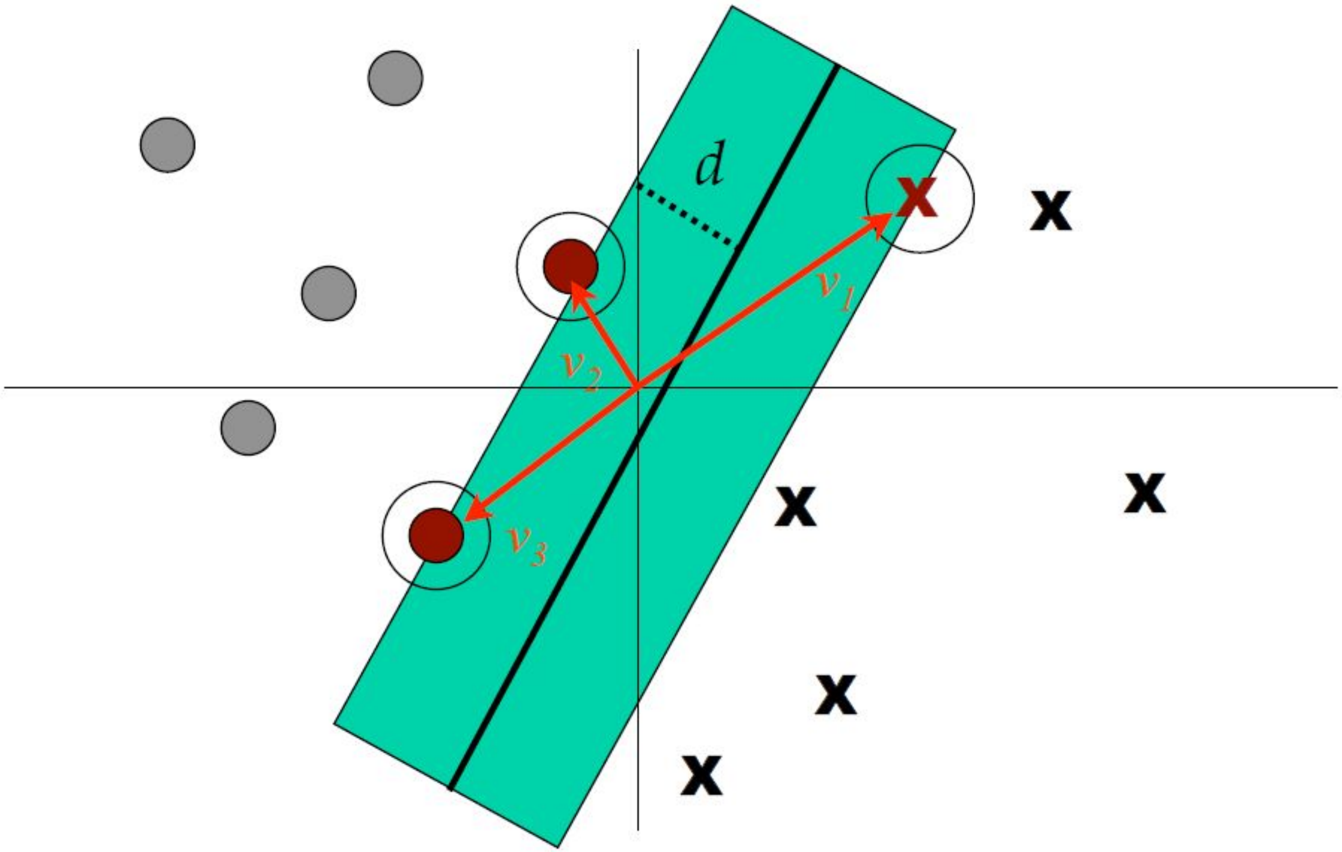
Support Vectors: Input vectors that just touch the boundary of the margin (street) – circled below, there are 3 of them (or, rather, the 'tips' of the vectors

$w_0{}^{\mathrm{T}}\mathbf{x} + b_0 = 1$ or $w_0{}^{\mathrm{T}}\mathbf{x} + b_0 = -1$



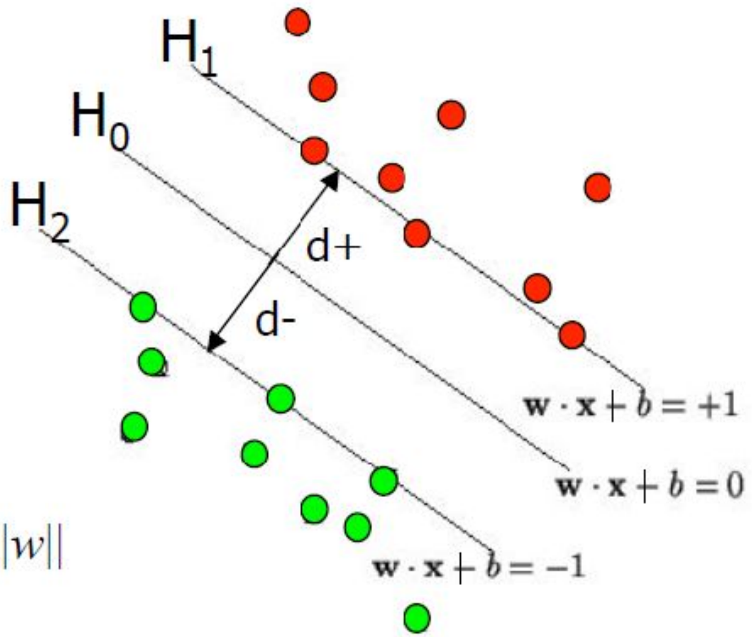
Margin

Here, we have shown the actual support vectors, v_1, v_2, v_3, instead of just the 3 circled points at the tail ends of the support vectors. d denotes 1/2 of the street 'width'



Maximum Margin

We want a classifier (linear separator) with as big a margin as possible.



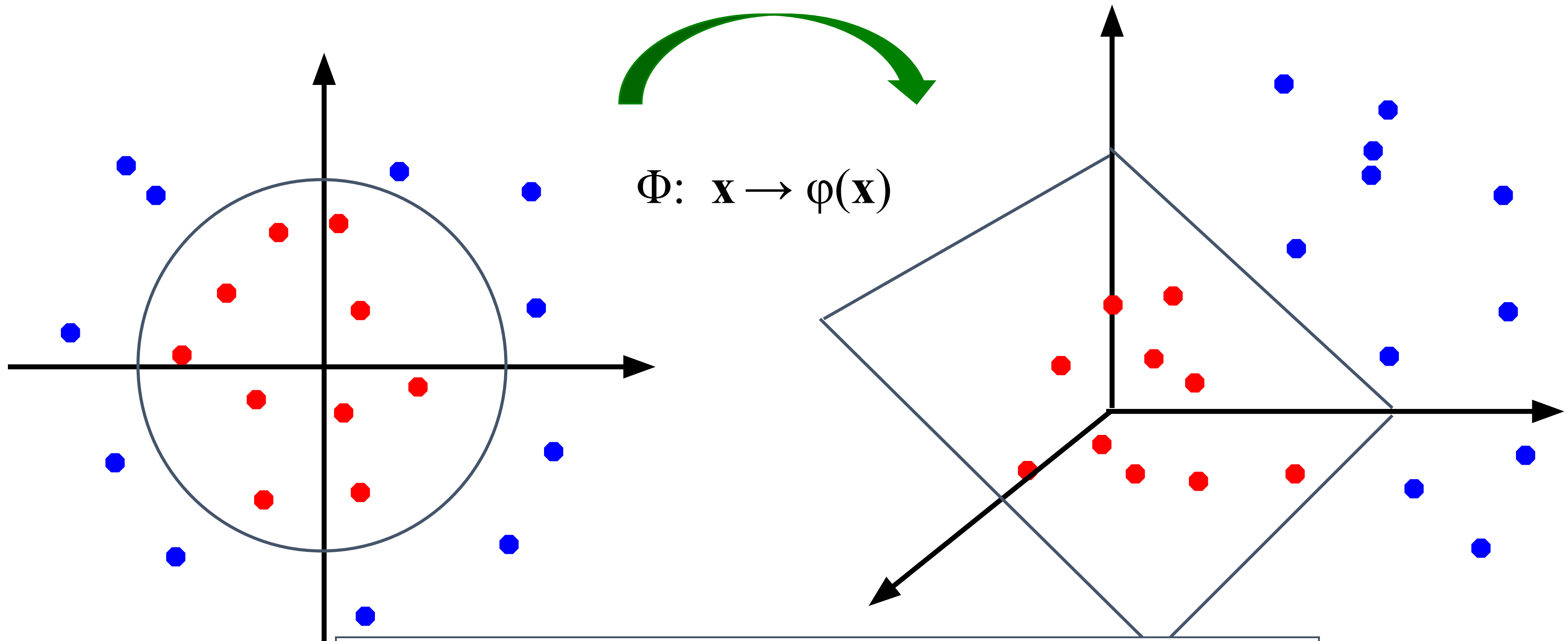
Recall the distance from a point(x_0,y_0) to a line:
$Ax+By+c = 0$ is: $|Ax_0 + By_0 + c|/\text{sqrt}(A^2+B^2)$, so,
The distance between H_0 and H_1 is then:
$|w \bullet x+b|/||w||=1/||w||$, so

The total distance between H_1 and H_2 is thus: $2/||w||$

In order to <u>maximize</u> the margin, we thus need to <u>minimize</u> $||w||$. <u>With the condition that there are no datapoints between II_1 and II_2:</u>

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:

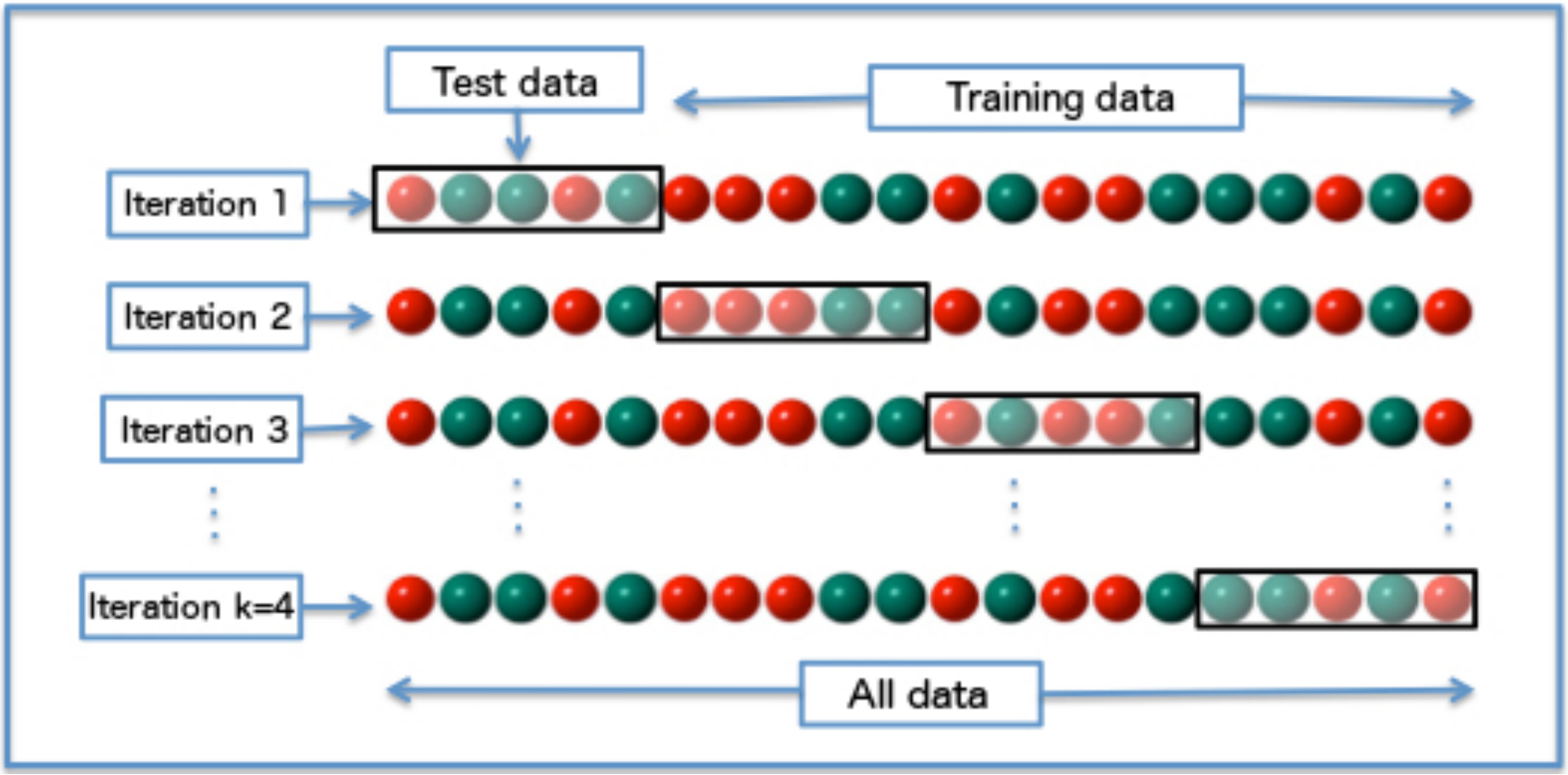


$$\Phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$$

$$(x_1, x_2) \mapsto (z_1, z_2, z_3) := (x_1^2, \sqrt{2}\, x_1 x_2, x_2^2)$$

Important Topics for Implementation

K-fold cross validation



Averaging multiple folds/categories

Micro-averaging : average using total TP/FP etc.

Macro-averaging: average of all fold’s/category’s Precision/Recall/F1-score

Average Precision

$$PRE = \frac{TP}{TP + FP}$$

Micro-averaging : average using total TP/FP etc.

$$PRE_{micro} = \frac{TP_1 + \dots + TP_k}{TP_1 + \dots + TP_k + FP_1 + \dots + FP_k}$$

Macro-averaging: average of all fold’s/category’s Precision/Recall/F1-score

$$PRE_{macro} = \frac{PRE_1 + \dots + PRE_k}{k}$$

Example : Microaverage vs. Macroaverage

Classification on first category

True positive (TP1) = 20
False positive (FP1) = 10
False negative (FN1) = 10

Classification on second category

True positive (TP2) = 40
False positive (FP2) = 20
False negative (FN2) = 10

$$PRE = \frac{TP}{TP + FP}$$

$$REC = TPR = \frac{TP}{P} = \frac{TP}{FN + TP}$$

$$F_1 = 2 \cdot \frac{PRE \cdot REC}{PRE + REC}$$

Example : Microaverage vs. Macroaverage

Classification on first category

True positive (TP1) = 20
False positive (FP1) = 10
False negative (FN1) = 10

Find precision and recall for each category

Classification on second category

True positive (TP2) = 40
False positive (FP2) = 20
False negative (FN2) = 10

$$PRE = \frac{TP}{TP + FP}$$

$$REC = TPR = \frac{TP}{P} = \frac{TP}{FN + TP}$$

$$F_1 = 2 \cdot \frac{PRE \cdot REC}{PRE + REC}$$

Example : Microaverage vs. Macroaverage

Classification on first category

True positive (TP1) = 20
False positive (FP1) = 10
False negative (FN1) = 10

Find F1-score for each category

Classification on second category

True positive (TP2) = 40
False positive (FP2) = 20
False negative (FN2) = 10

$$PRE = \frac{TP}{TP + FP}$$

$$REC = TPR = \frac{TP}{P} = \frac{TP}{FN + TP}$$

$$F_1 = 2 \cdot \frac{PRE \cdot REC}{PRE + REC}$$

Example : Microaverage vs. Macroaverage

Classification on first category

True positive (TP1) = 20
False positive (FP1) = 10
False negative (FN1) = 10

Classification on second category

True positive (TP2) = 40
False positive (FP2) = 20
False negative (FN2) = 10

Find micro/macro averaged F1-score of the two categories

Thank You!