KNN

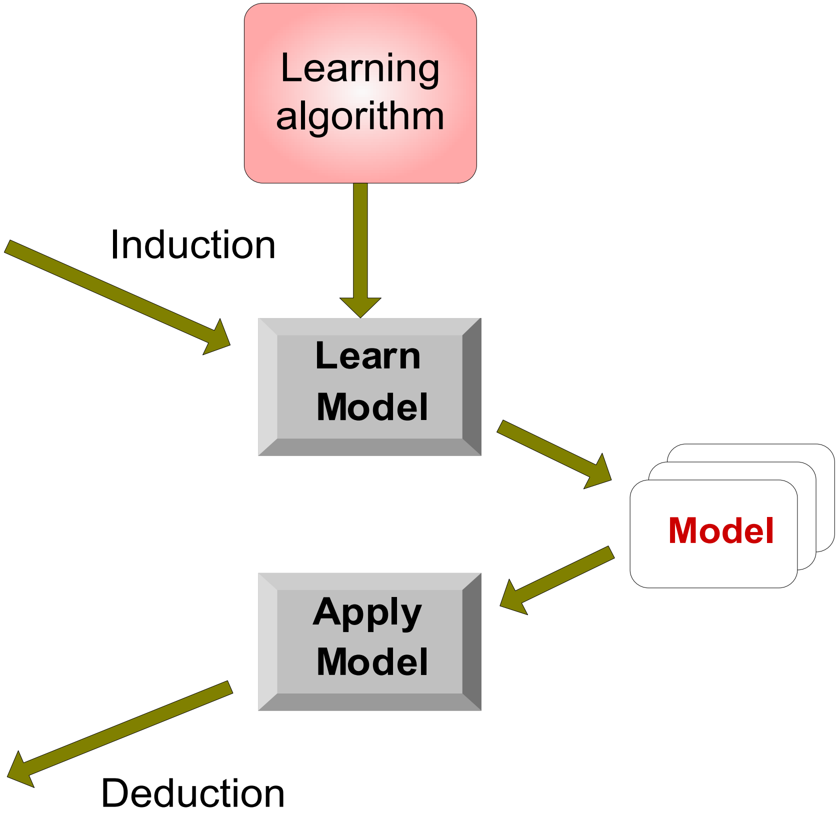
SYRACUSE UNIVERSITY
School of Information Studies

REVIEW DISTANCE MEASURE

CLASSIFICATION PROCESS

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

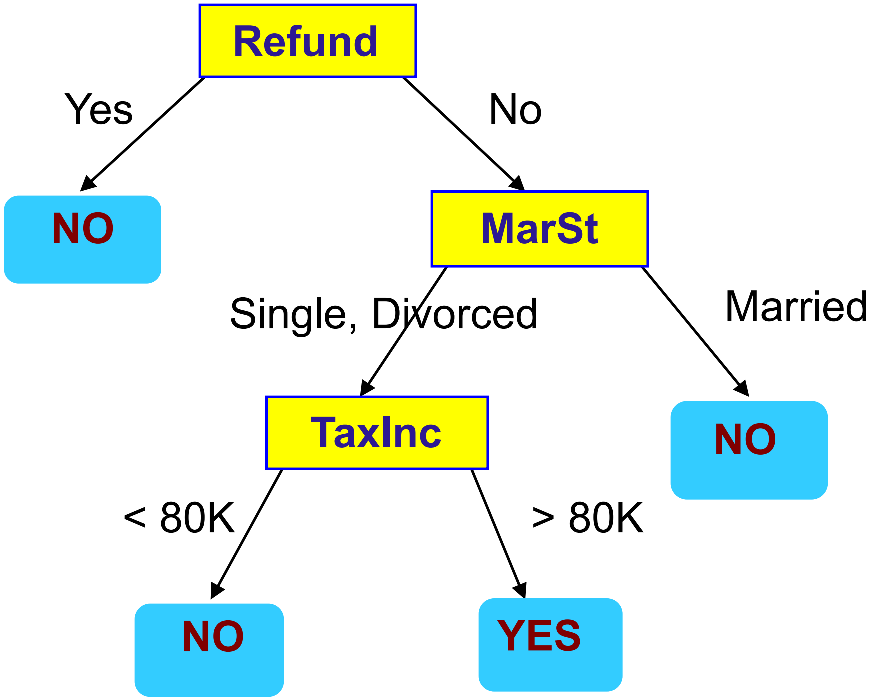


Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

MACHINE LEARNING ALGORITHMS

Algorithms like decision tree and naive Bayes will construct a learning model from training examples and then apply the model for prediction on new test examples.



naive Bayes Classifier:

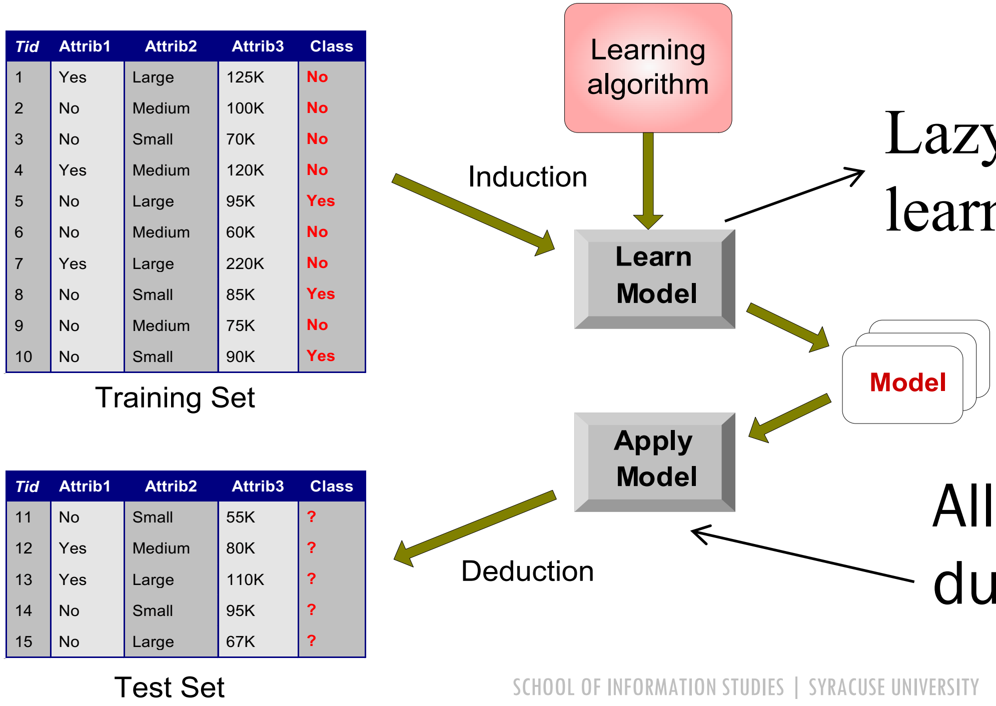
P(Refund=Yes|No) = 3/7
P(Refund=No|No) = 4/7
P(Refund=Yes|Yes) = 0
P(Refund=No|Yes) = 1
P(Marital Status=Single|No) = 2/7
P(Marital Status=Divorced|No)=1/7
P(Marital Status=Married|No) = 4/7
P(Marital Status=Single|Yes) = 2/3
P(Marital Status=Divorced|Yes)=1/3
P(Marital Status=Married|Yes) = 0

For taxable income:

If class=No:	sample mean=110
	sample variance=2975
If class=Yes:	sample mean=90
	sample variance=25

INSTANCE-BASED LEARNING

In contrast, instance-based learning methods simply store the training examples without doing any calculations during training process, and classification and prediction are delayed until new examples are given.



Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

Lazy learner

All calculations happen during prediction.

K-NEAREST NEIGHBOR (K-NN)

Training process:

Read in all training examples.

Classification process:

Given a test example, compare the similarity between the test example and all training examples. Choose the majority-voted category label in the k-nearest training examples.

NEAREST NEIGHBOR CLASSIFICATION

Choosing the value of k:

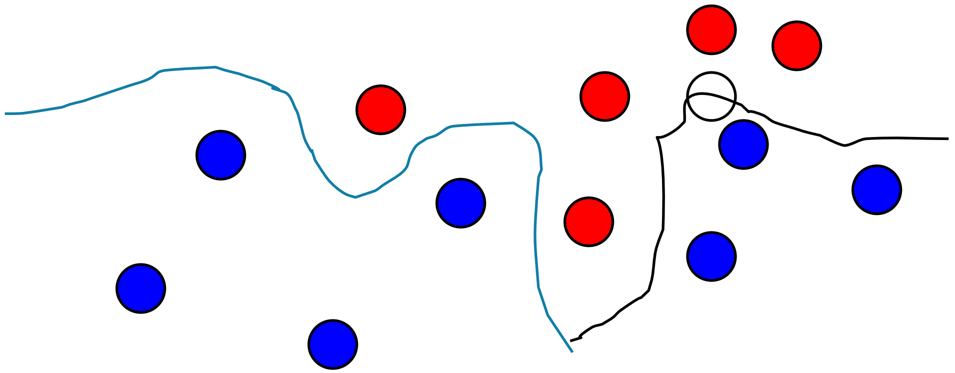
- If k is too small, sensitive to noise points.
- If k is too large, neighborhood may include points from other classes.

ADVANTAGES OF K-NN

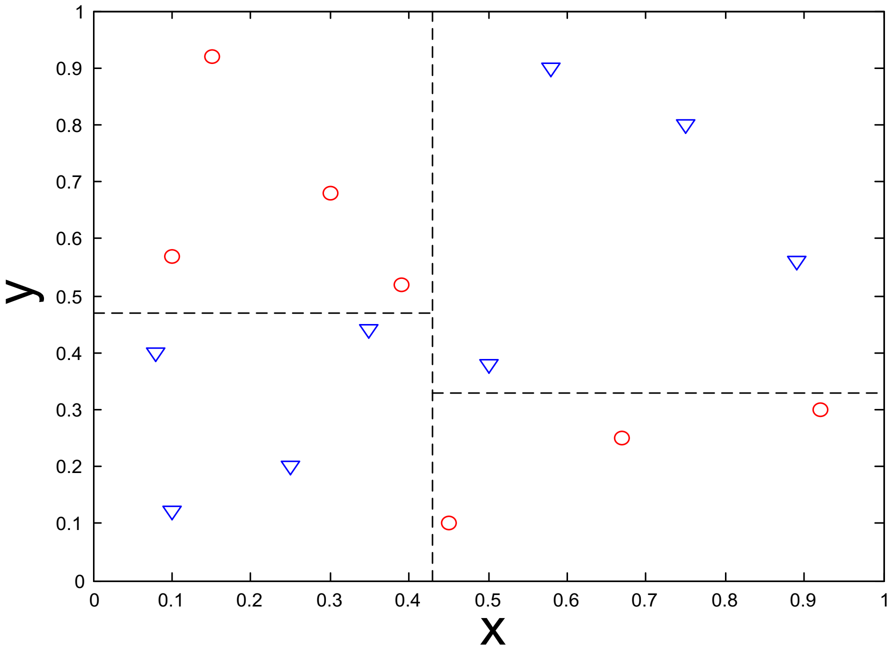
No assumptions made

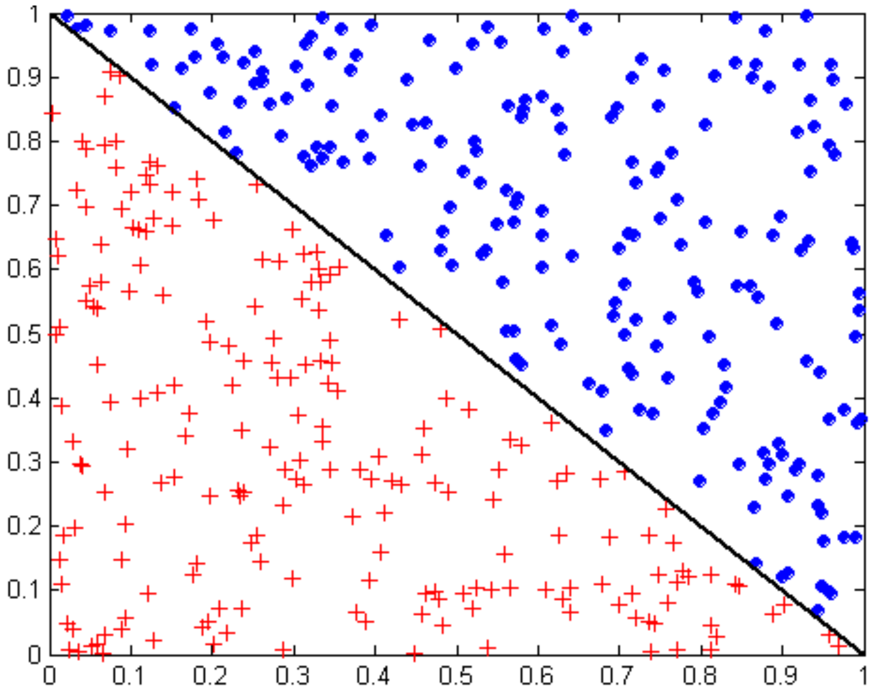
- Remember the independence assumption in naive Bayes.

Works well when the decision function to be learned is very complex

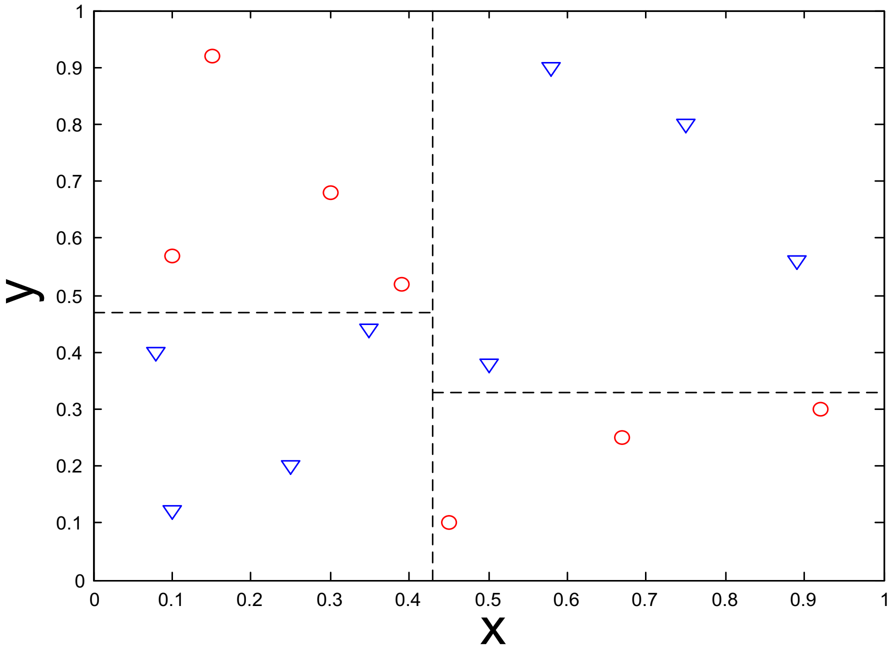


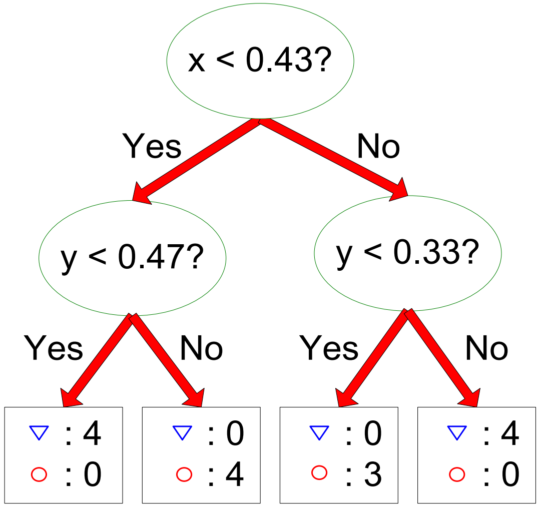
THE SHAPE OF DECISION BOUNDARY MATTERS





DECISION BOUNDARY OF DECISION TREE MODELS



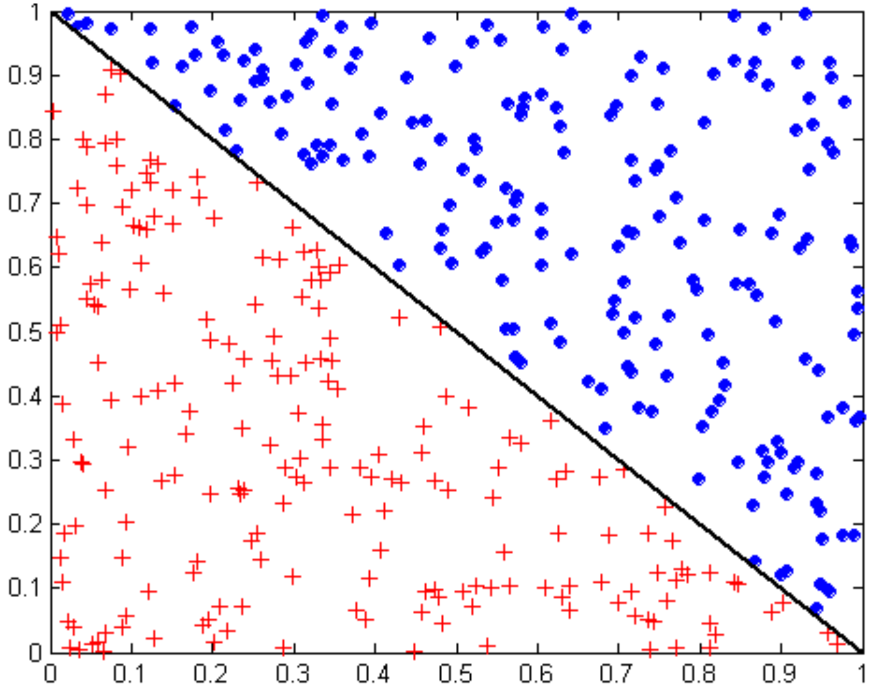


DECISION BOUNDARY OF LINEAR MODELS

Linear: Naive Bayes, SVM

How many parameters to determine a line in 2D space?

- Y = ax + b
- Weight
- Intercept



WHY IS NAIVE BAYES A LINEAR CLASSIFIER?

The decision function can be rewritten to a linear function.

Original decision function:

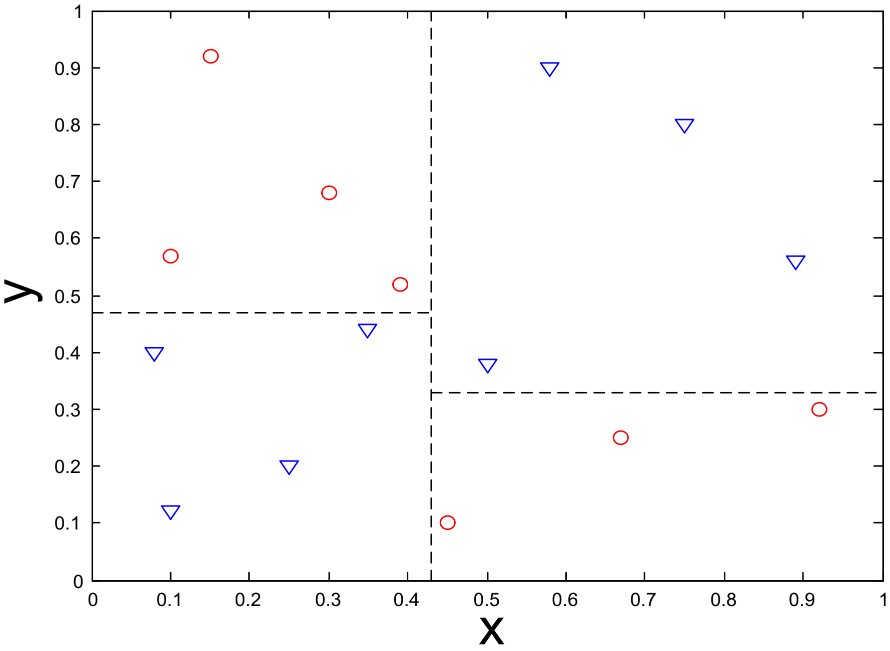
Prob(Ci)*Prob(T1|Ci)*Prob(T2|Ci) * … *Prob(Tm|Ci))

Apply log transformation:

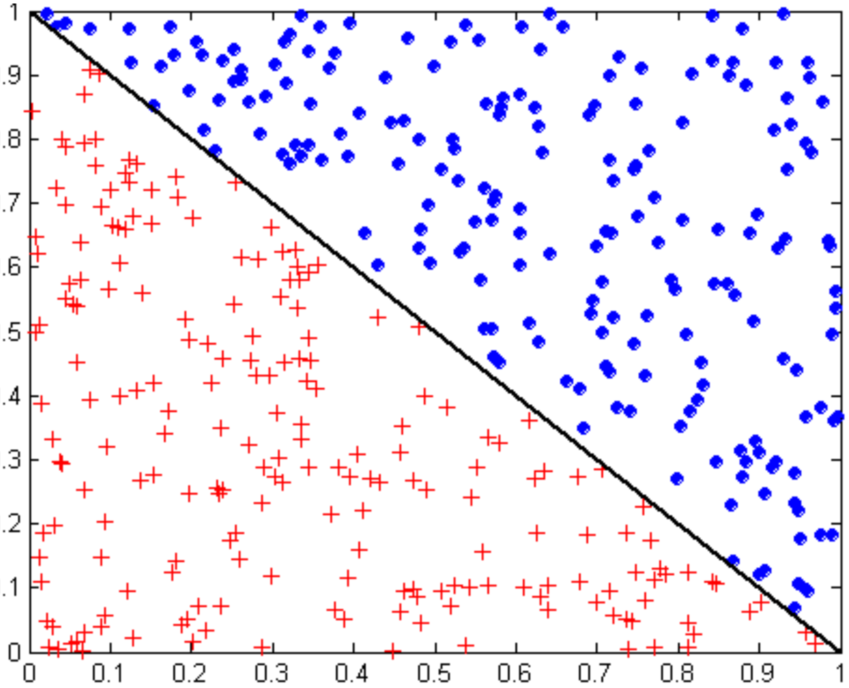
log(Prob(Ci)) + log(Prob(T1|Ci)) + log(Prob(T2|Ci)) + … + log(Prob(Tm|Ci))

http://cs.nyu.edu/faculty/davise/ai/bayesText.html

THE SHAPE OF DECISION BOUNDARY MATTERS



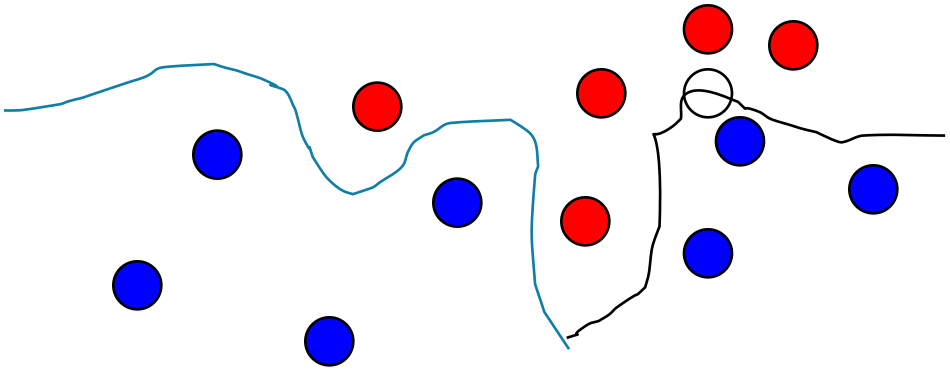
Decision tree model fits; not the linear model



Linear model fits; not decision tree model

ADVANTAGE OF K-NN

The decision boundary has no predefined shape.



DISADVANTAGES OF K-NN

Sensitive to noisy training data

All attributes participate in classification.

If only a few relevant attributes are relevant to prediction, the participation of those irrelevant attributes would harm the prediction performance.

DISADVANTAGES OF K-NN

High computational cost

Precomputed models can be quickly applied to test data.

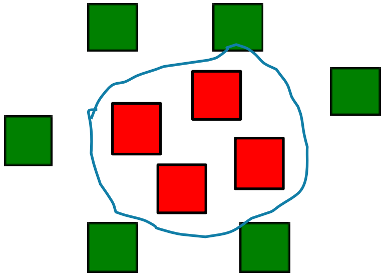
Since there is no training step, nearly all computation takes place in the prediction step.

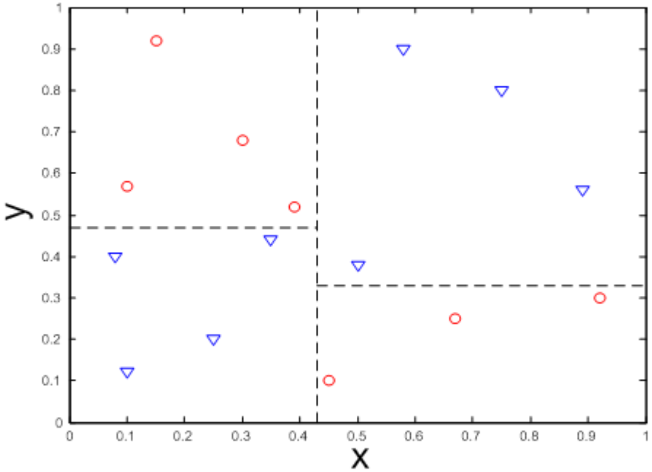
MAX-MARGIN SVMS

SYRACUSE UNIVERSITY
School of Information Studies

THE SHAPE OF DECISION BOUNDARY

Some data are not linearly separable.

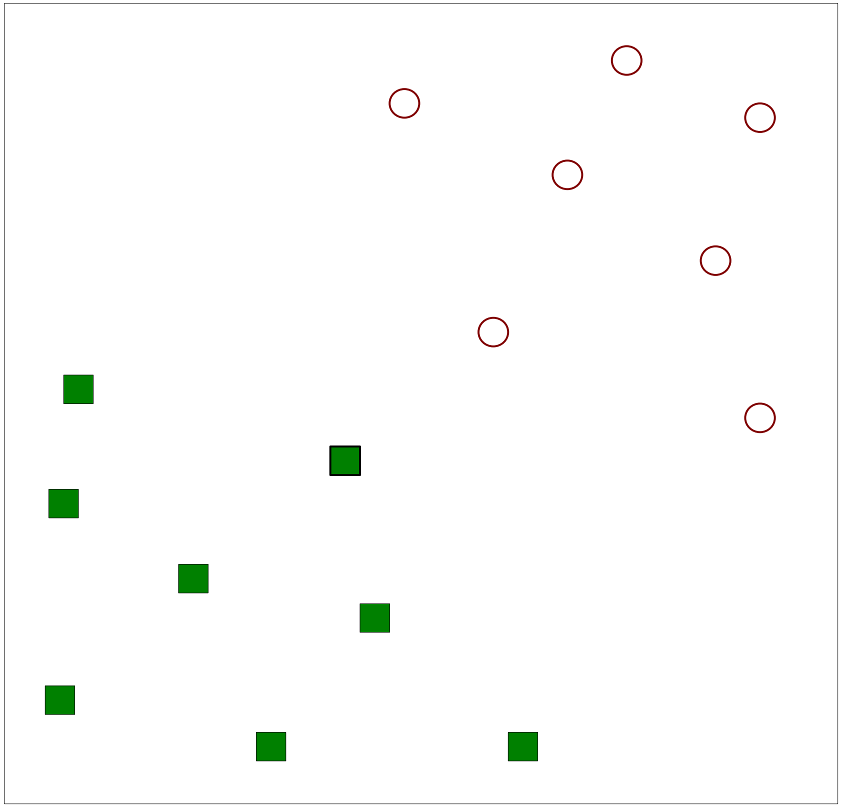




Support vector machine (SVM): An algorithm that can solve both linearly separable and inseparable problems

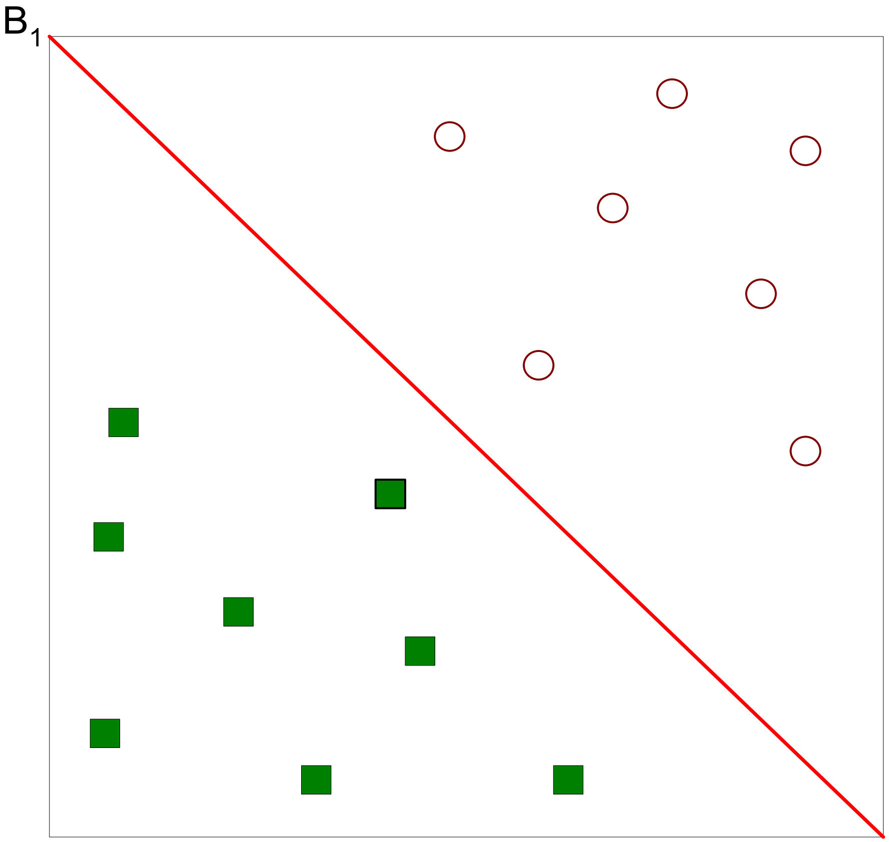
SUPPORT VECTOR MACHINES

Find a linear hyperplane (decision boundary) that can separate the data.



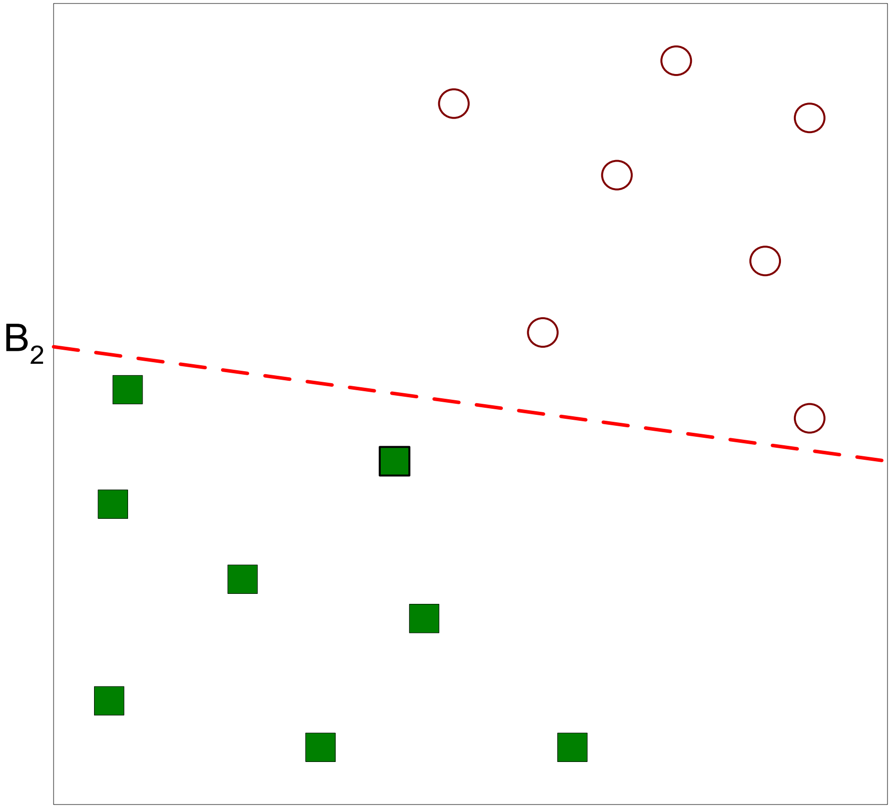
SUPPORT VECTOR MACHINES

One possible solution:



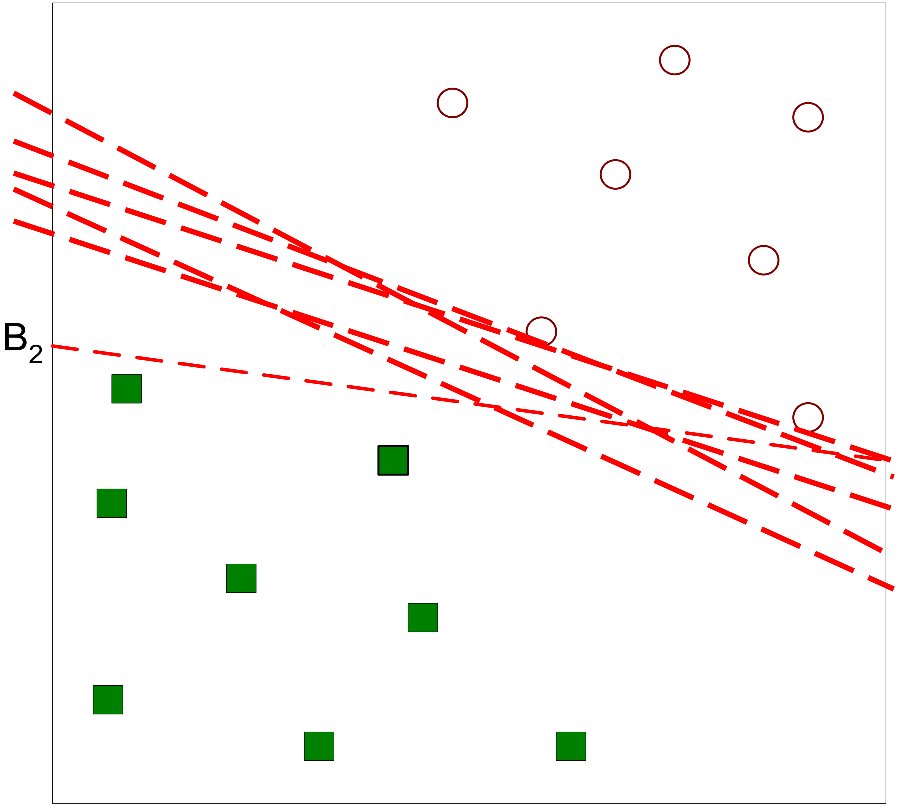
SUPPORT VECTOR MACHINES

Another possible solution:



SUPPORT VECTOR MACHINES

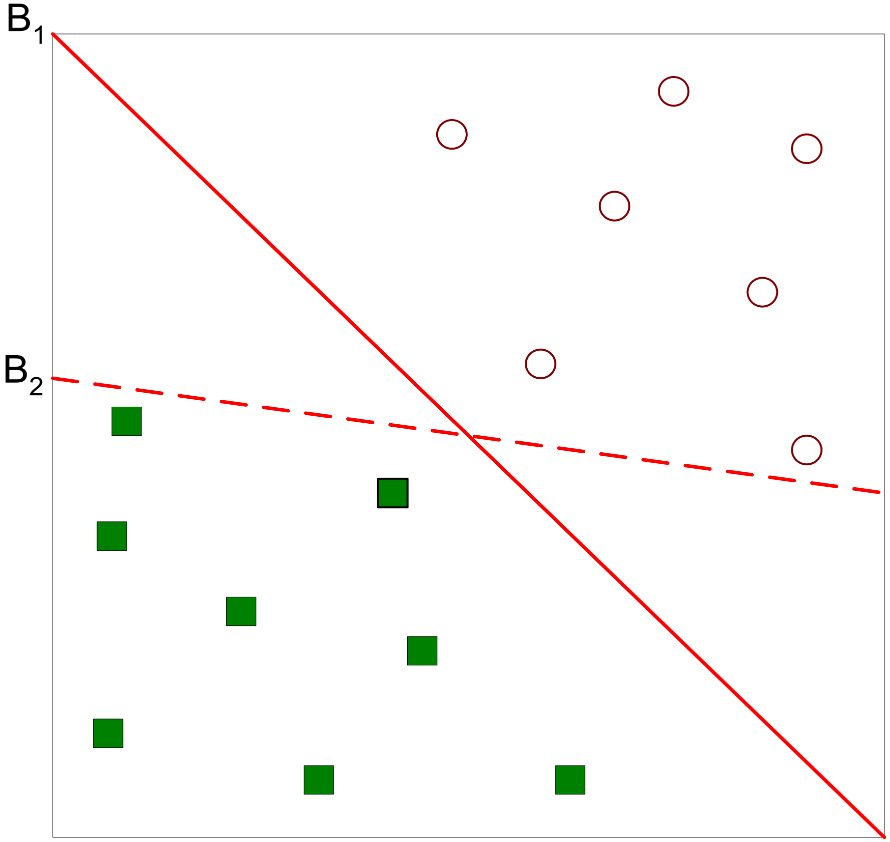
Numerous possible solutions:



SUPPORT VECTOR MACHINES

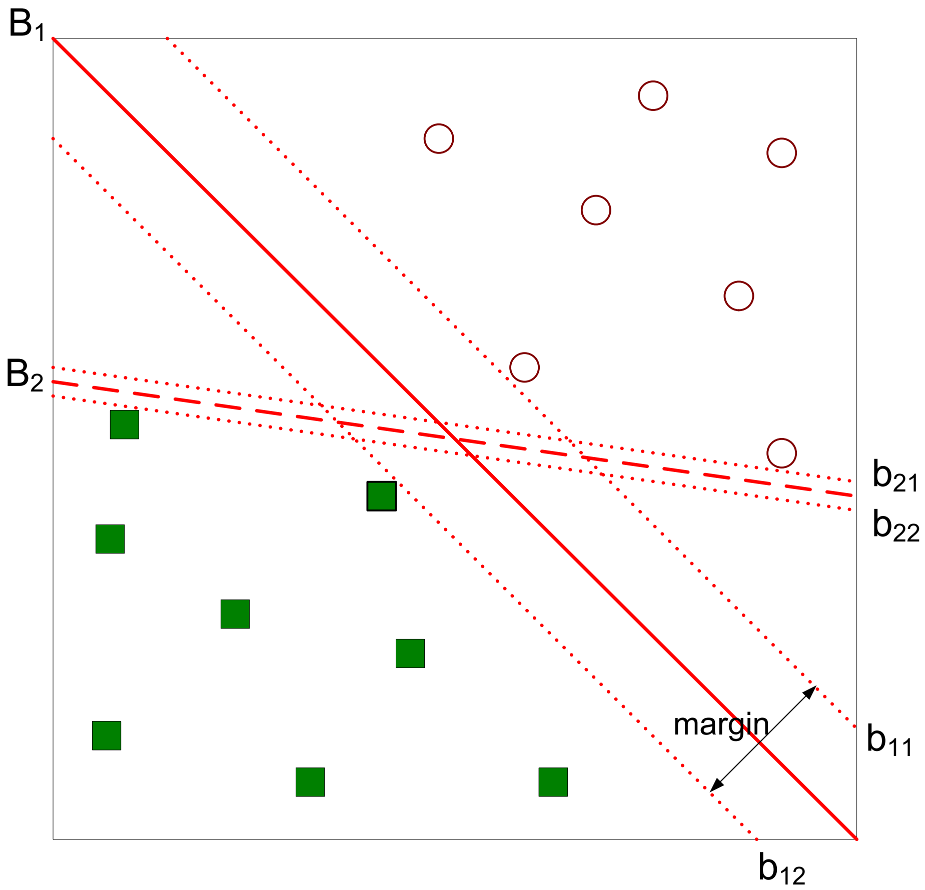
Which one is better: B1 or B2?

How do you define better? (E.g., “least square fitting”?)

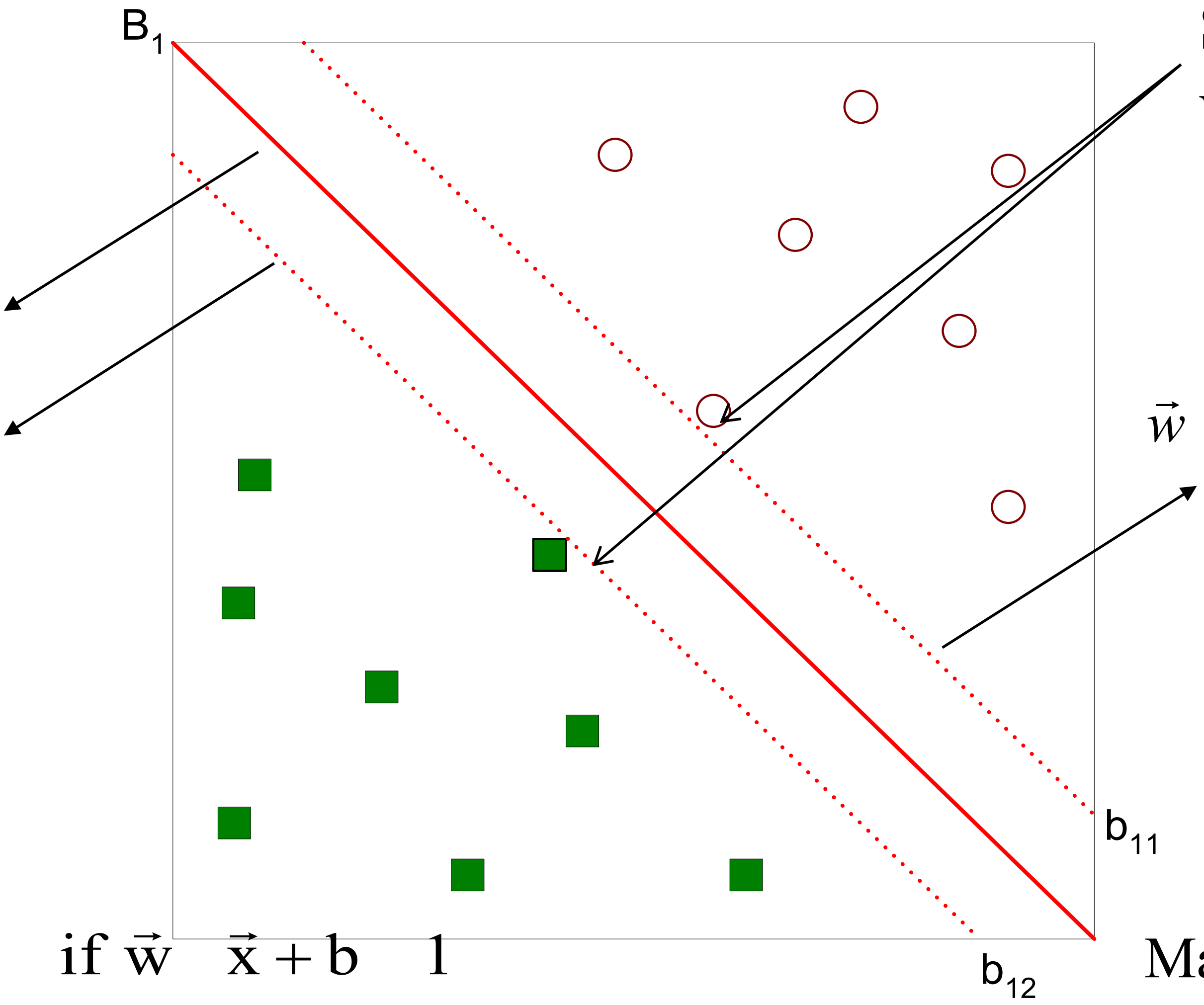


SUPPORT VECTOR MACHINES

Find a hyperplane that maximizes the margin => B1 is better than B2.



SUPPORT VECTOR MACHINES



Support vectors

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

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

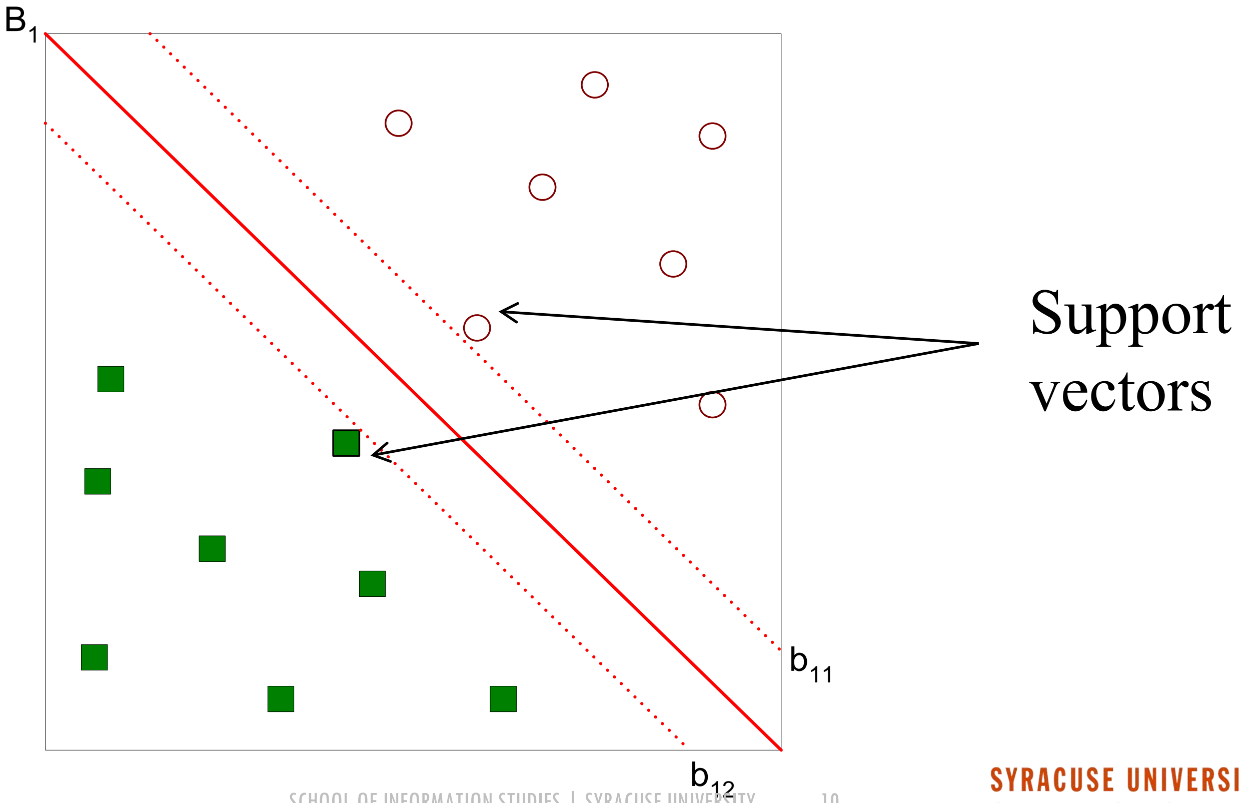
$\vec{w} \quad \vec{x} + b = +1$

$$f(\vec{x}) = \begin{matrix} 1 & \text{if } \vec{\mathrm{w}} \quad \vec{\mathrm{x}} + \mathrm{b} \quad 1 \\ 1 & \text{if } \vec{\mathrm{w}} \quad \vec{\mathrm{x}} + \mathrm{b} \quad 1 \end{matrix}$$

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

SUPPORT VECTORS

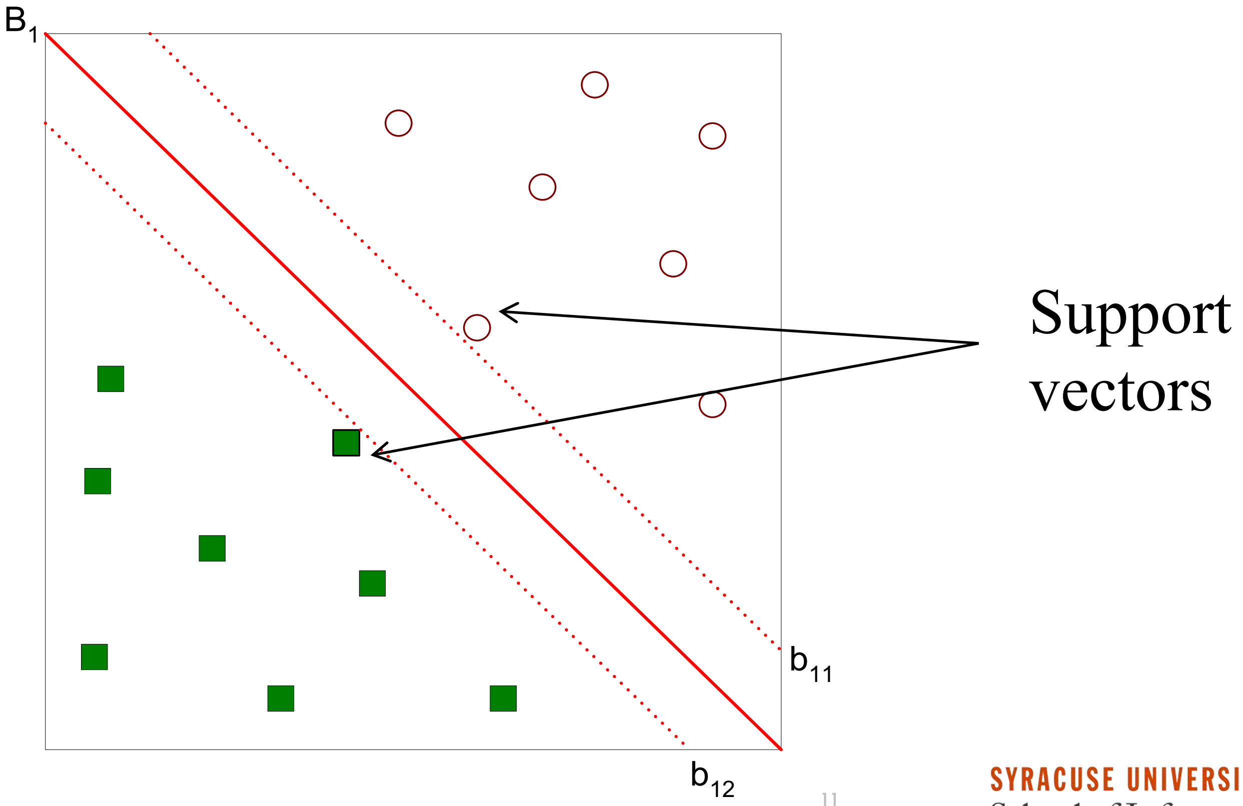
Support vectors are the training examples (“vectors”) that are located on the margins.



Support vectors

DECISION BOUNDARY

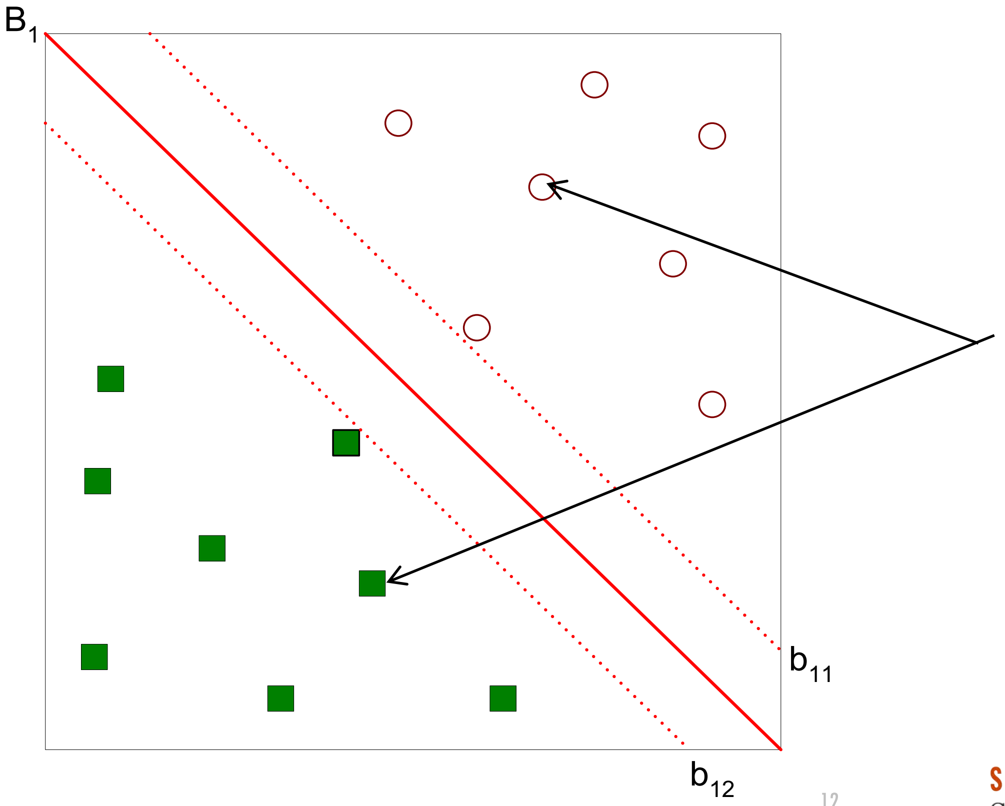
Only support vectors determine the decision boundary.



Support vectors

NON–SUPPORT VECTORS

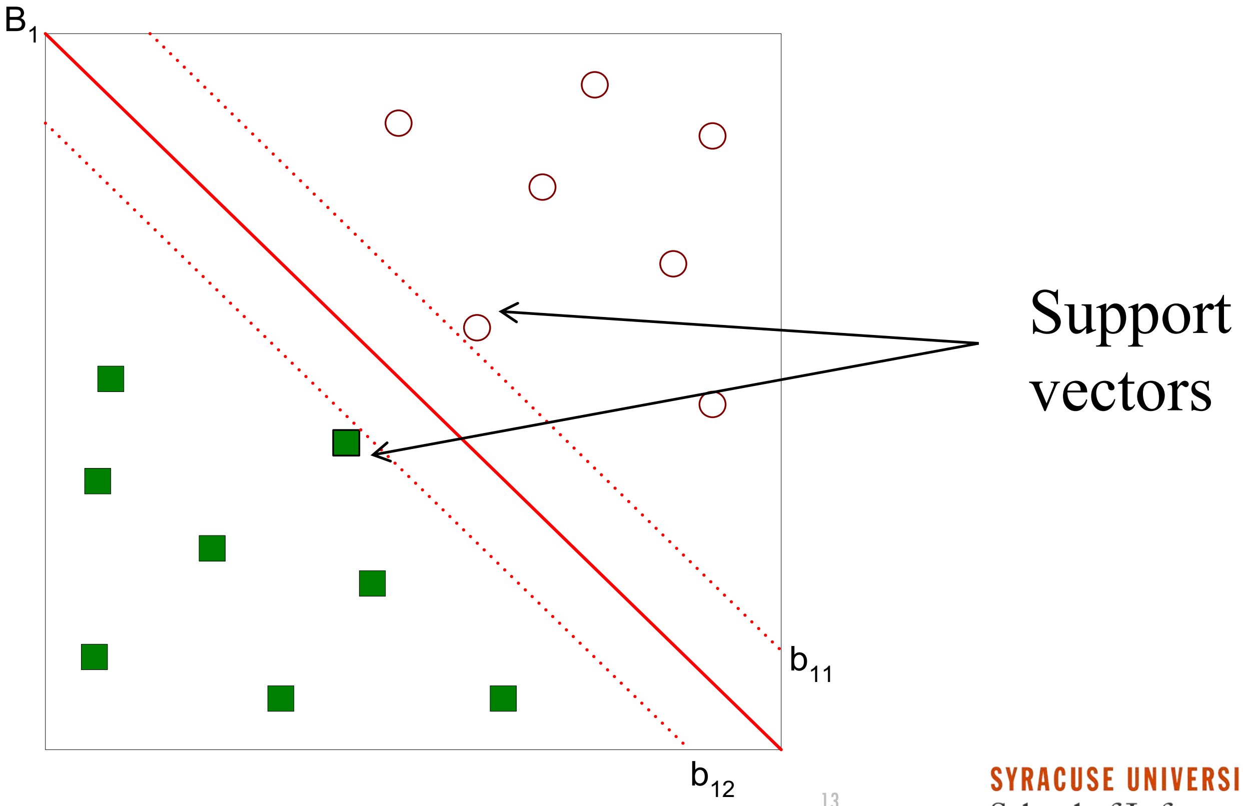
Training examples that are not support vectors do not participate in prediction.



Non–support vectors

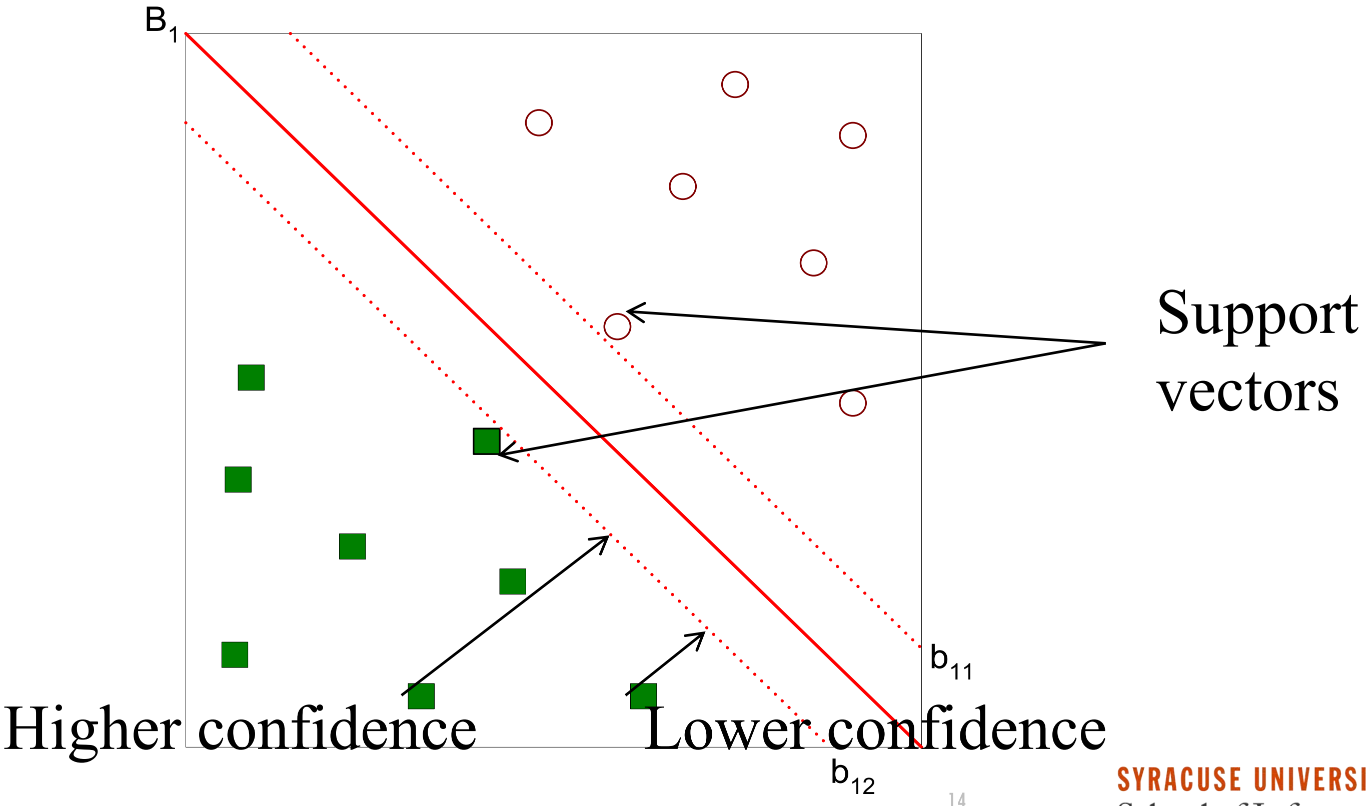
MODEL COMPLEXITY

The number of support vectors is an indicator of the complexity of the trained SVM model.



PREDICTION CONFIDENCE

The distance between the example and the decision boundary is an indicator of prediction confidence: The farther the better.



Support vectors

Higher confidence

Lower confidence

PREDICTION CONFIDENCE

SVM’s prediction result can be sorted by confidence and thus is suitable for semi-supervised learning and active learning.

Variant SVMs algorithm can be used for regression.

SOFT-MARGIN SVMS

SYRACUSE UNIVERSITY
School of Information Studies

SOFT-MARGIN SVMS

No perfect linear boundary can be found between the two classes due to outliers.

Introduce a slack variable ξ to pay a cost for each misclassified example.

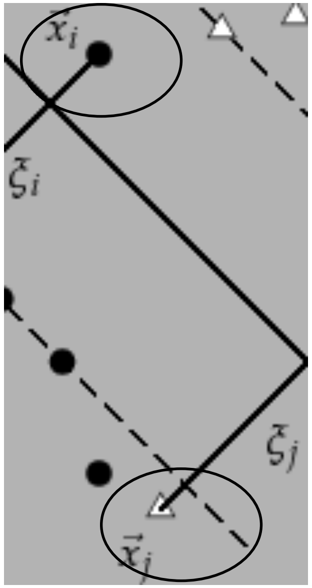


Figure 15.5 from http://nlp.stanford.edu/IR-book/html/htmledition/soft-margin-classification-1.html

REGULARIZATION IN C-SVC

Tune the regularization parameter C (cost for misclassification).

- Default value: C = 1

When C is large (high cost), the algorithm tries to build model with fewest training errors, resulting in narrow margin and high chance of overfitting.

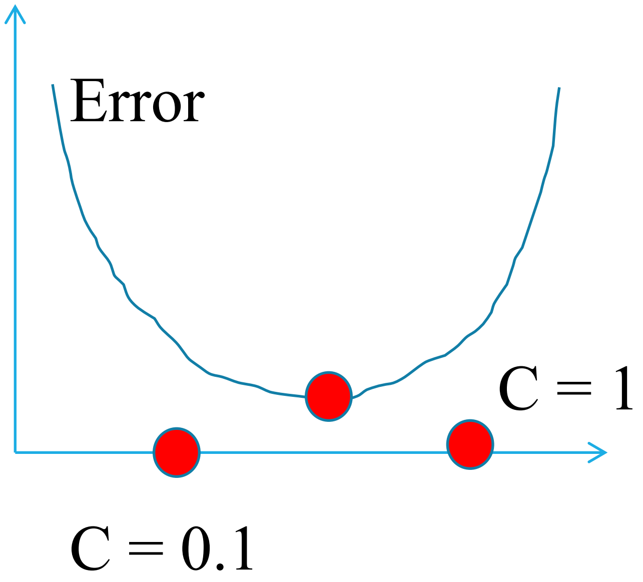
When C is small (low cost), the margin is wider, more robust.

However, C cannot be too small, or else it does not respect the data at all.

REGULARIZATION

Use manual tuning or gradient descent search to find the best C.

E.g., set C’s search range from 0.1 to 1.0 and increase with step size 0.05.



A VISUALIZATION FROM COURSERA

https://class.coursera.org/ml-003/lecture/72

07:13–9:00

SVM KERNELS (CONT.)

SYRACUSE UNIVERSITY
School of Information Studies

KERNEL FUNCTIONS

SVM algorithm maximizes the margin between the two separating hyperplanes by finding the maximum of the function:

$$W(\quad) = \mathop{}_{i=1}^{l} {}_{i} \quad \frac{1}{2} \mathop{}_{i=1}^{l} \mathop{}_{j=1}^{l} {}_{i} \quad {}_{j} y_i y_j \boxed{K(x_i, x_j)}$$

Subject to the constraints:

$$\mathop{}_{i=1}^{l} {}_{i} y_i = 0, \quad {}_{i} \quad 0, i = 1,2,\ldots,l$$

SVM: KERNEL FUNCTIONS

Linear kernel: $K(X_i, X_j) = X_i \cdot X_j$ (cosine similarity)

Higher rank kernels: Instead of computing on the transformed data tuples, it is mathematically equivalent to instead applying a kernel function $K(X_i, X_j)$ to the original data, i.e., $K(X_i, X_j) = \Phi(X_i)\ \Phi(X_j)$

Typical kernel functions:

$$\text{Polynomial kernel of degree } h: \quad K(X_i, X_j) = (X_i \cdot X_j + 1)^h$$

$$\text{Gaussian radial basis function kernel}: \quad K(X_i, X_j) = e^{-\|X_i - X_j\|^2/2\sigma^2}$$

$$\text{Sigmoid kernel}: \quad K(X_i, X_j) = \tanh(\kappa X_i \cdot X_j - \delta)$$

SVM KERNELS

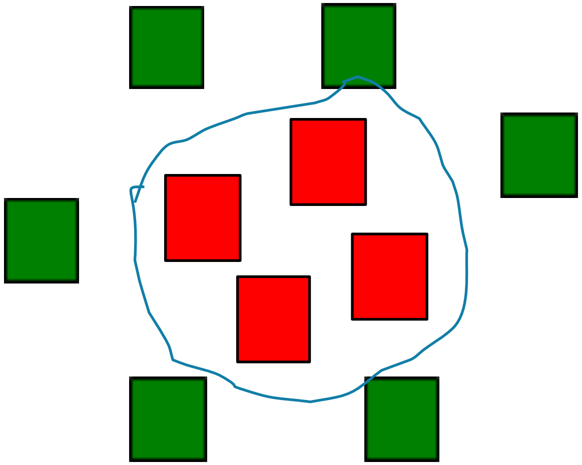
SYRACUSE UNIVERSITY
School of Information Studies

THE KERNEL TRICK IN SVM

Some data are not linearly separable.

But they are linearly separable in higher dimensional space.

Map the data to higher dimensional space, so that inseparable problems become separable.



STRENGTH AND WEAKNESS

SYRACUSE UNIVERSITY
School of Information Studies

EXTEND BINARY CLASSIFICATION TO MULTICLASS

Given n classes, e.g.

Sentiment = {positive, negative, neutral, no opinion}

One-versus-one (pairwise) strategy:

Create n(n – 1)/2 classifiers: pos|neg, pos|neu, pos|np, neg|neu, neg|np, neu|np

Pick the most confident prediction.

One-versus-all strategy:

Create n classifiers: positive or not, negative or not, neutral or not, np or not

Pick the most confident prediction.

SVMS' STRENGTH

High tolerance to noisy data

Flexibility in data representation: Well suited for continuous- or discrete-valued inputs and outputs

Probabilistic prediction result

Scalability: Successful on extremely large problems

Successful on a wide array of real-world data

PROBABILISTIC OUTPUT OF SVMS

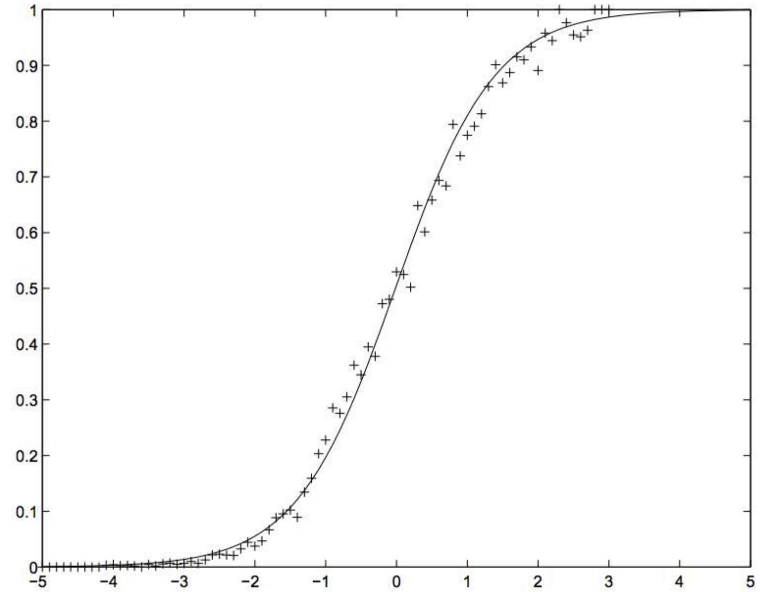


Figure 2: The fit of the sigmoid to the data for a linear SVM on the Adult data set (as in Figure 1). Each plus mark is the posterior probability computed for all examples falling into a bin of width 0.1. The solid line is the best-fit sigmoid to the posterior, using the algorithm described in this chapter.

Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers, 10*(3), 61– 74.

SVMS' WEAKNESS

Require a number of parameters for each kernel type

Interpretability

- Easy interpretation for linear kernel
- Difficult to interpret the model generated by nonlinear kernels

ENSEMBLE LEARNING

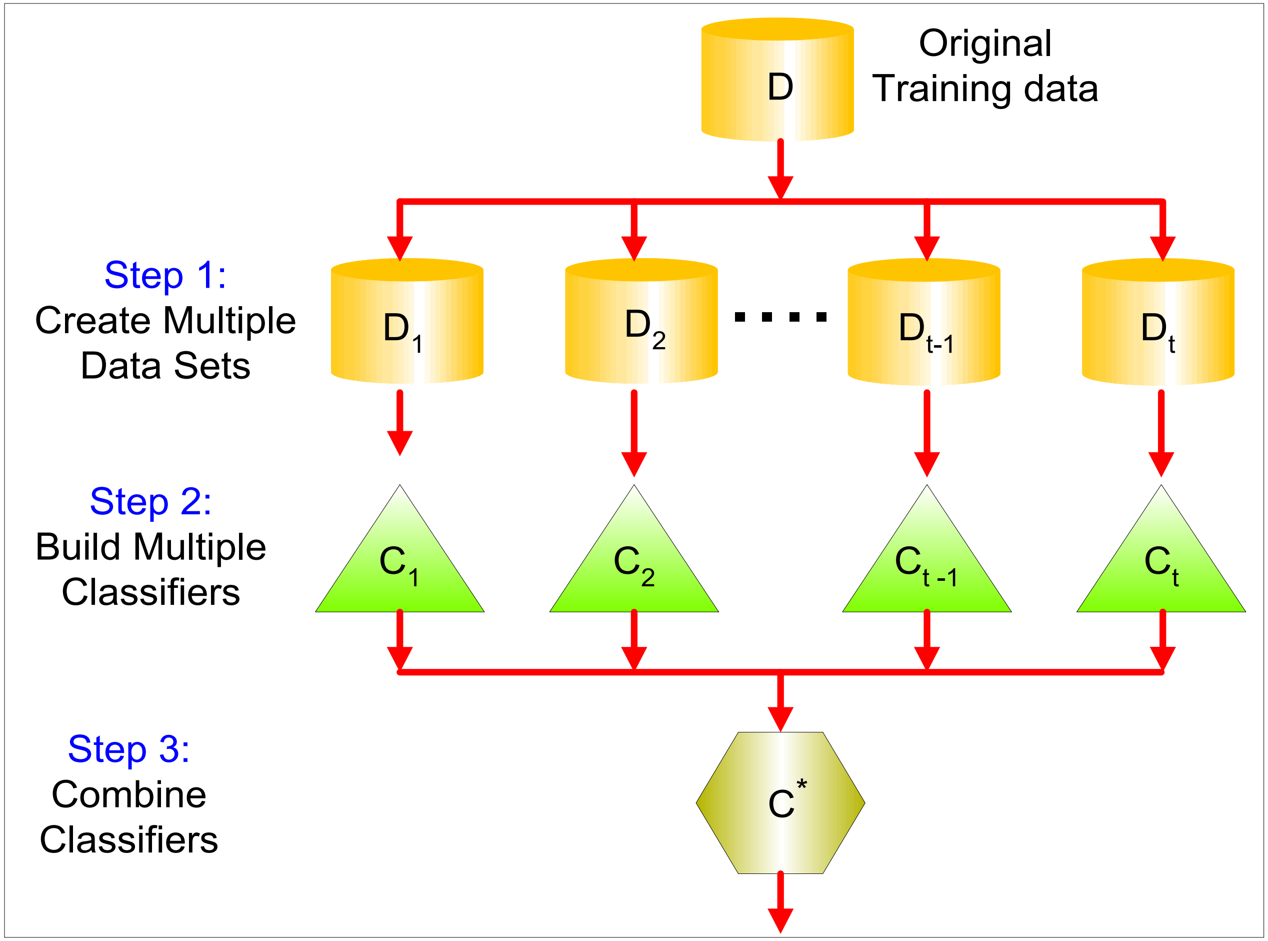
SYRACUSE UNIVERSITY
School of Information Studies

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.

GENERAL IDEA



WHY DOES ENSEMBLE WORK?

Suppose there are 25 base classifiers.

Each classifier has error rate, ε = 0.35 (weak learner).

Assume classifiers are independent.

Use majority voting to combine results, so ensemble makes a wrong prediction only if over half of the base classifiers are wrong.

Probability that the ensemble classifier makes a wrong prediction:

$$\overset{25}{\underset{i=13}{}} \quad \genfrac{}{}{0pt}{}{25}{i} \quad {}^{i}(1 \qquad)^{25 \quad i} = 0.06$$

Error rate is reduced from 0.35 to 0.06.

In practice, the base classifiers may not be totally independent for a reduction in error rate to occur.

BAGGING

Sampling with replacement:

Original Data	1	2	3	4	5	6	7	8	9	10
Bagging (Round 1)	7	8	10	8	2	5	10	10	5	9
Bagging (Round 2)	1	4	9	1	2	3	2	7	3	2
Bagging (Round 3)	1	8	5	10	5	5	9	6	3	7

Build classifier on each bootstrap sample.

Each sample has probability $(1 – 1/n)^n$ of being selected.

BOOSTING

An iterative procedure to adaptively change distribution of training data by focusing more on previously misclassified records.

Initially, all N records are assigned equal weights.

Unlike bagging, weights may change at the end of boosting round.

BOOSTING

Records that are wrongly classified will have their weights increased.

Records that are classified correctly will have their weights decreased.

Original Data	1	2	3	4	5	6	7	8	9	10
Boosting (Round 1)	7	3	2	8	7	9	4	10	6	3
Boosting (Round 2)	5	4	9	4	2	5	1	7	4	2
Boosting (Round 3)	4	4	8	10	4	5	4	6	3	4

Example 4 is hard to classify.

Its weight is increased; therefore, it is more likely to be chosen again in subsequent rounds.

RANDOM FOREST

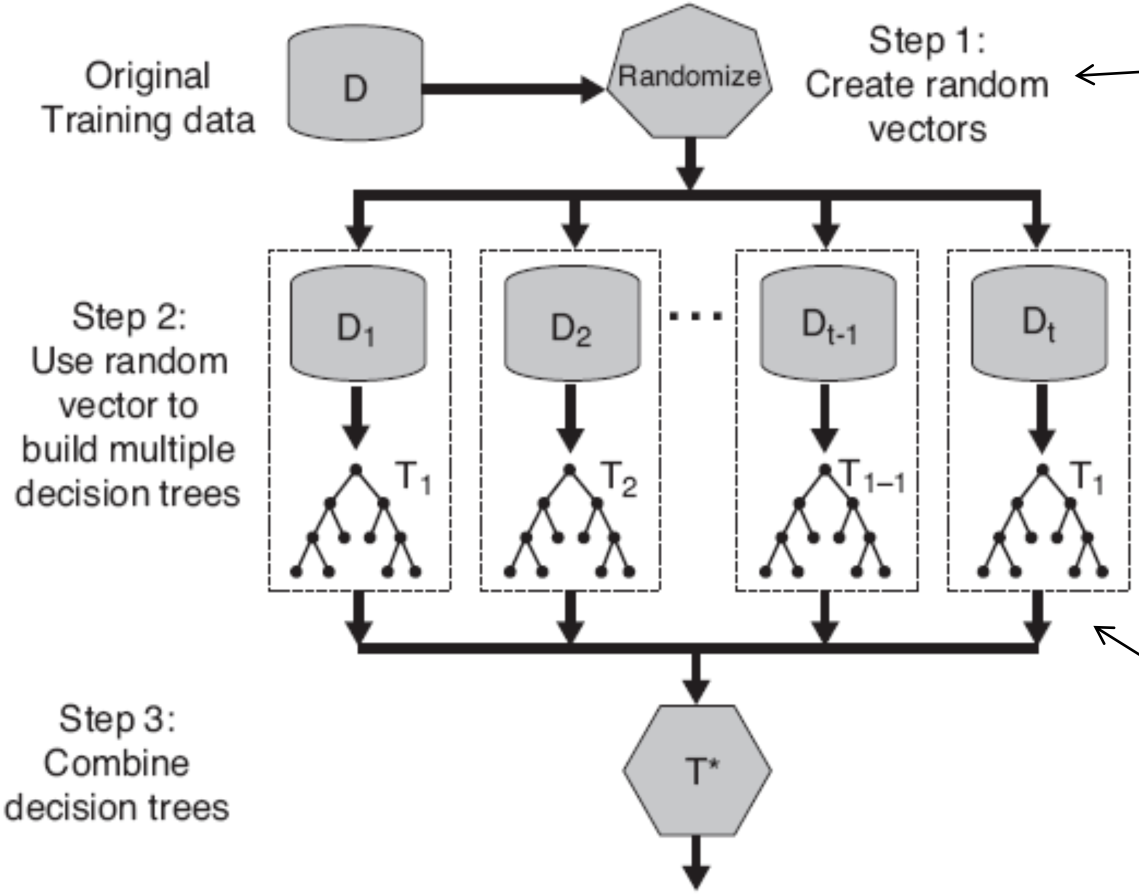


Figure 5.40. Random forests.

Randomly select subset of features.

Model with too many trees tends to overfit.