CS722/822: Machine Learning

Instructor: Jiangwen Sun
Computer Science Department

Classification: Definition

- Given a collection of examples (*training set*)
  - Each example contains a set of *variables (or features*, X*)*, and the target variable (y, categorical), which offers *class* info.
- Find a *model* for class variable as a function of other variables.
- Goal: previously unseen examples should be assigned a class as accurately as possible.
  - A *test set* is used to determine the accuracy of the model. Usually, the given data set is divided into training and test sets, with training set used to build the model and test set used to validate it.

Classification Application 1

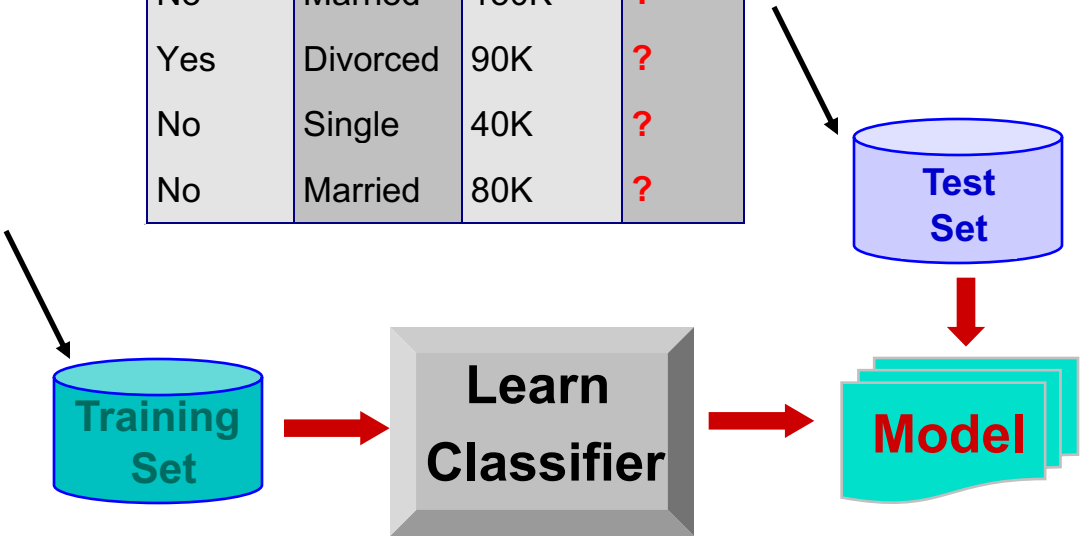
categorical | categorical | continuous | class

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Past transaction records, label them

Current data, want to use the model to predict

Refund	Marital Status	Taxable Income	Cheat
No	Single	75K	?
Yes	Married	50K	?
No	Married	150K	?
Yes	Divorced	90K	?
No	Single	40K	?
No	Married	80K	?



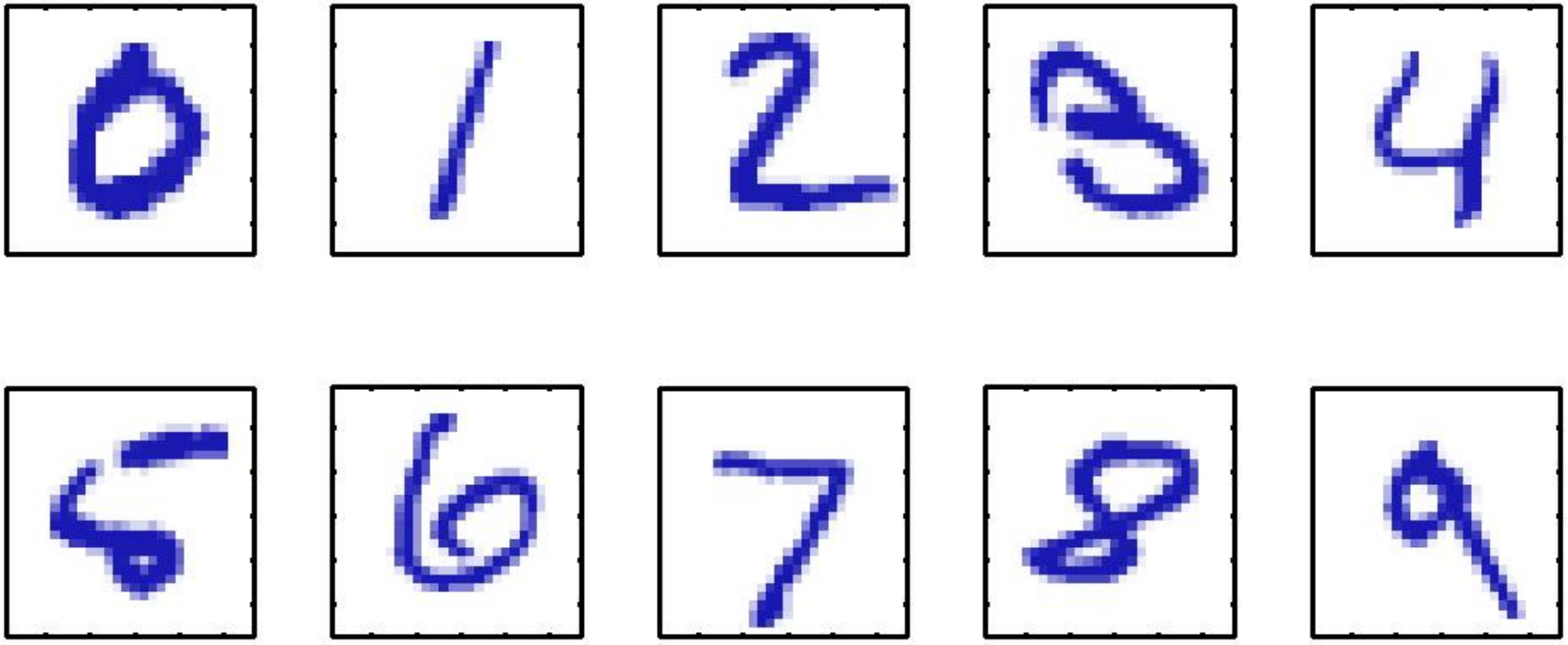
Fraud detection – goals: Predict fraudulent cases in credit card transactions.

Classification: Application 2

Handwritten Digit Recognition

Goal: Identify the digit of a handwritten number in an image

- Approach:
  - Align all images to derive the features (HOG, SURF, etc)
  - Model the class (identity) based on these features



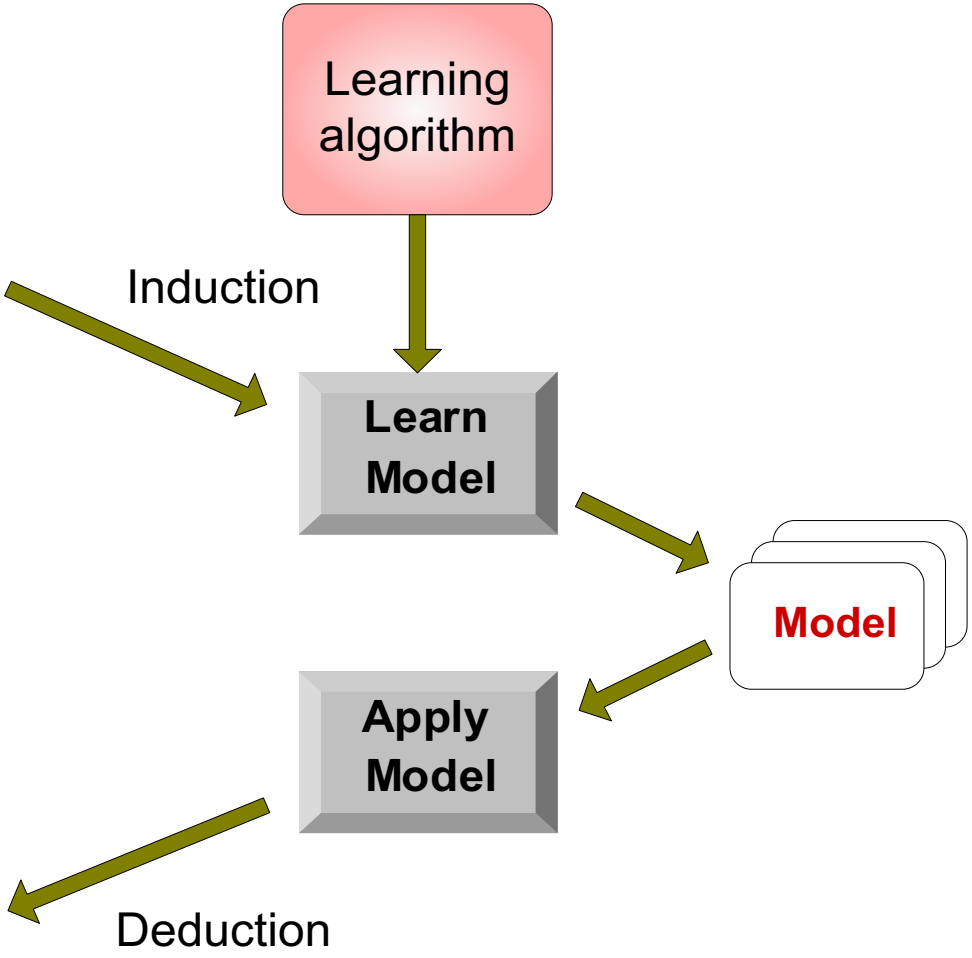
Illustrating Classification Task

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



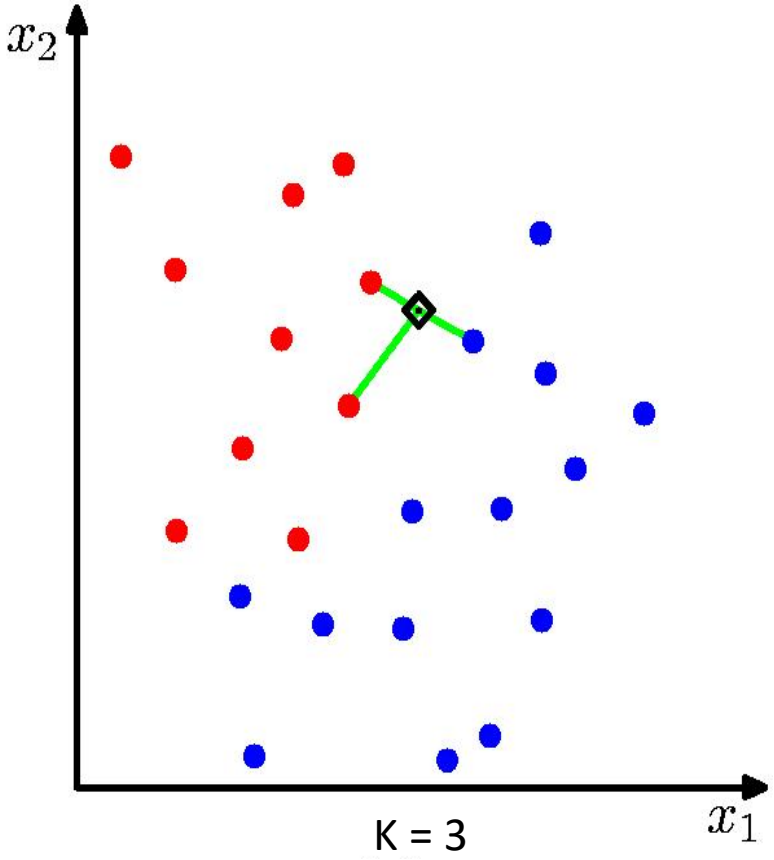
Classification algorithms

- K-Nearest-Neighbor classifiers
- Linear Discriminant Analysis (LDA)
- Logistic Regression
- Decision Tree
- Support Vector Machines (SVM)
- Neural Networks
- Naïve Bayes classifier
- Probabilistic Graphical models

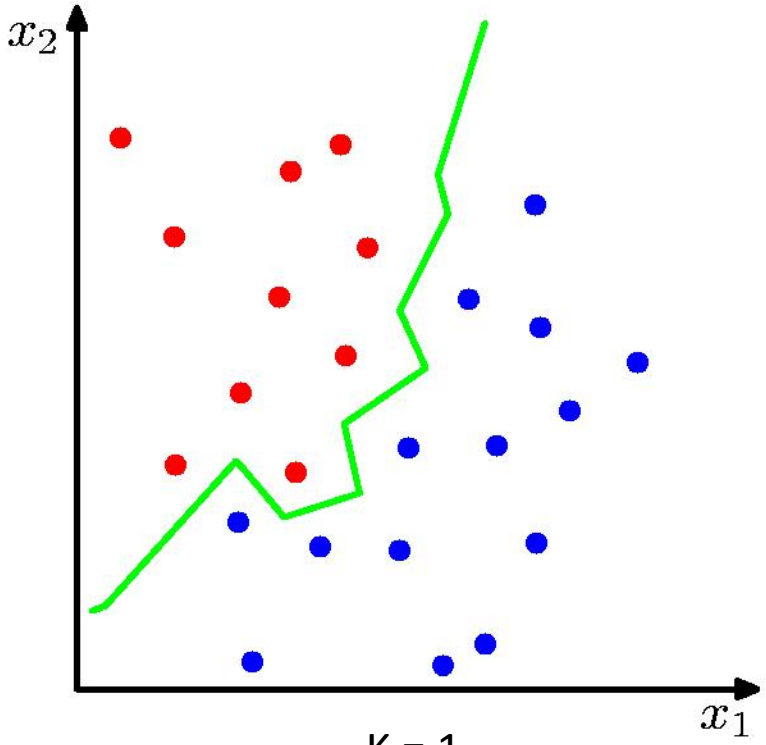
K-nearest neighbor (K-NN)

- K-NN is one of the simplest machine learning algorithm
- K-NN is a method for classifying test examples based on closest training examples in the feature space
- An example is classified by a majority vote of its neighbors
- *k* is a positive integer, typically small. If *k* = 1, then the example is simply assigned to the class of its nearest neighbor.

K-NN

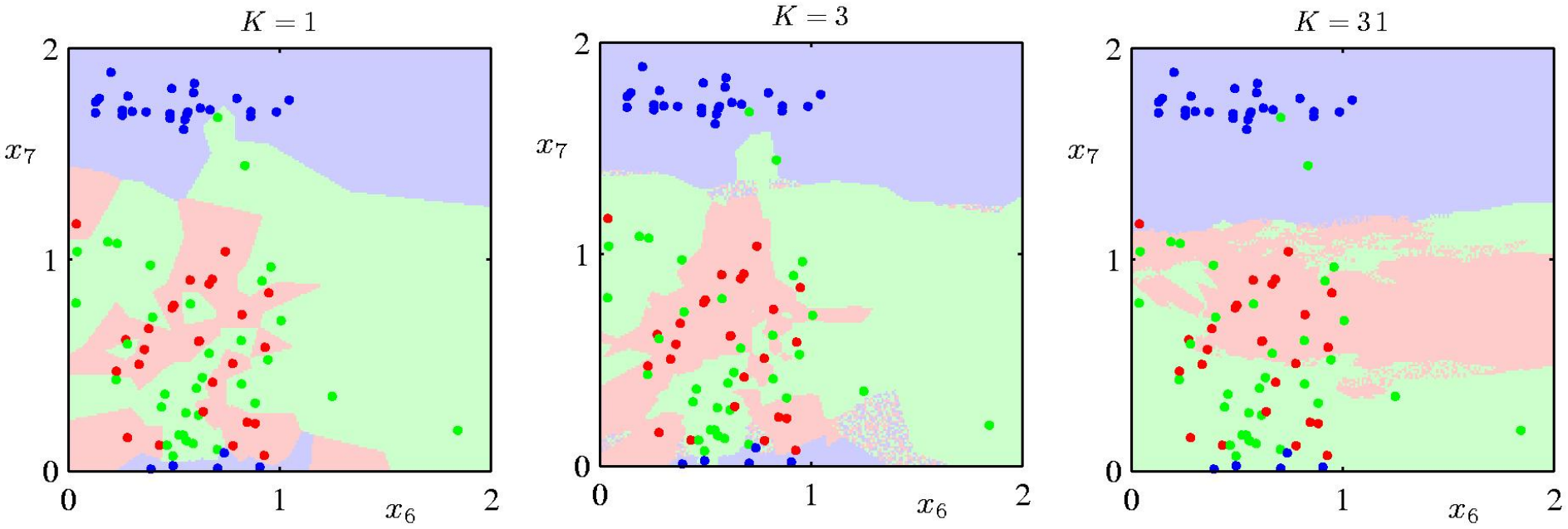


K = 3



K = 1

K-NN on real problem data



- Three classes, two variables
- K acts as a smoother, choosing K is model selection

Limitation of K-NN

- K-NN is a nonparametric model (no any particular function is fitted)
- Nonparametric models requires storing and computing with the entire data set while making prediction.
- Parametric models, once fitted, are much more efficient in terms of storage and computation.

Probabilistic interpretation of K-NN

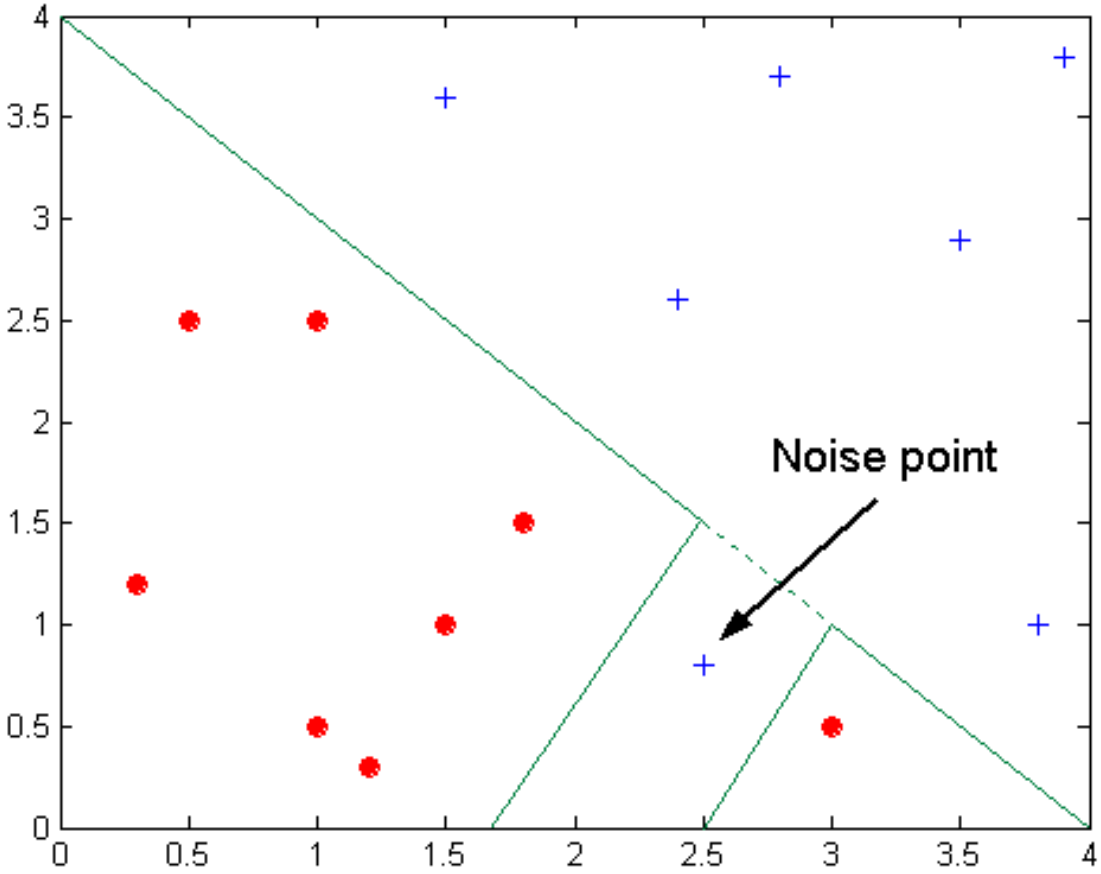
- Given a data set with N_c data points from class c and $\sum_c N_c = N$
- To make prediction for any data point x, draw a sphere centred on x containing precisely K data points in the dataset. Let V represent the volume of the sphere, the we have:

$$p(x) = \frac{K}{NV}$$

- Since $p(\mathrm{x}|c) = \frac{K_c}{N_c V}$ and $p(c) = \frac{N_c}{N}$, Bayes' theorem gives

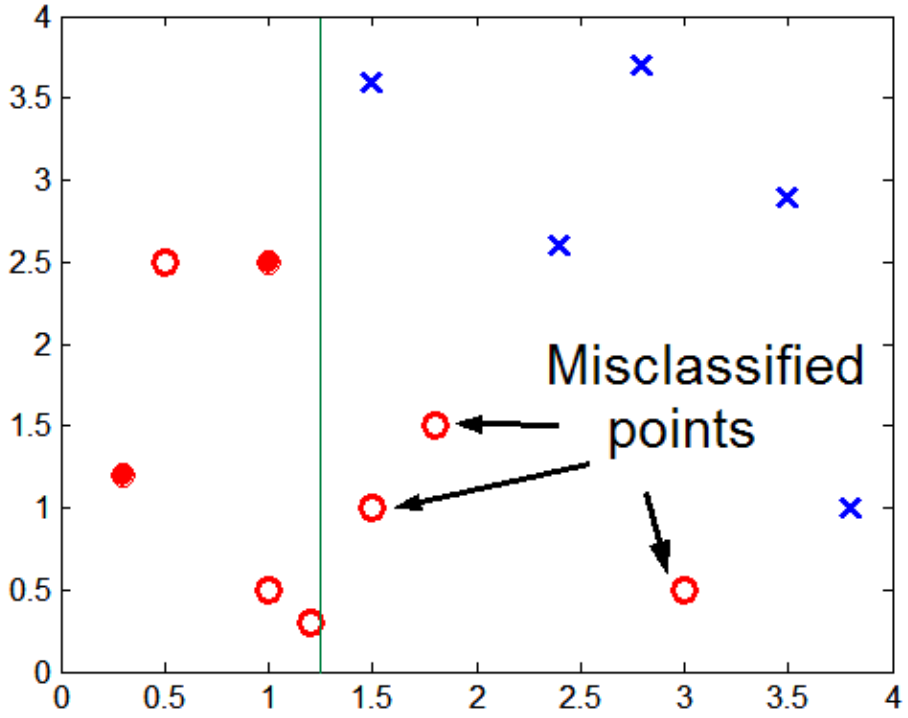
$$p(c|\mathrm{x}) = \frac{p(\mathrm{x}|c)p(c)}{p(\mathrm{x})} = \frac{K_c}{K}$$

Overfitting due to Noise



Decision boundary is distorted by noise point

Overfitting due to Insufficient Examples



Lack of data points in the lower half of the diagram makes it difficult to predict correctly the class labels of that region

Notes on Overfitting

- Overfitting results in classifiers (a neural net, or a support vector machine) that are more complex than necessary
- Training error no longer provides a good estimate of how well the classifier will perform on previously unseen records
- Need better estimate of the error on the true population – generalization error

$$P_{population}(\text{ f(x) not equal to y })$$

  - In theoretical analysis, find an analytical bound to bound the generalization error
  - In practice, design a procedure that gives better estimate of the error than training error

Occam's Razor

- Given two models of similar generalization errors, one should prefer the simpler model over the more complex model

- For complex models, there is a greater chance that it was fitted accidentally by errors in data

- Therefore, one should include model complexity when evaluating a model