CS4801 : Classification

Sahely Bhadra
16/8/2017

1. Introduction to Classification
2. Nearest Neighbour Classification
3. Bayes decision rule
   1. Classification error
   2. Minimum error rate classification
      1. Two category
      2. Multi category

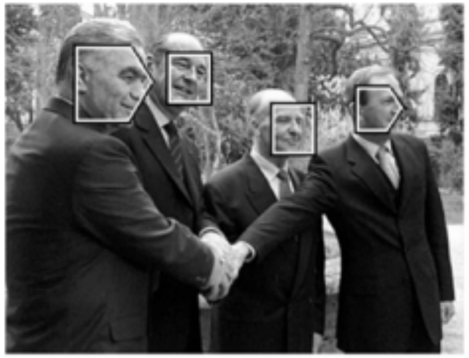
Classification

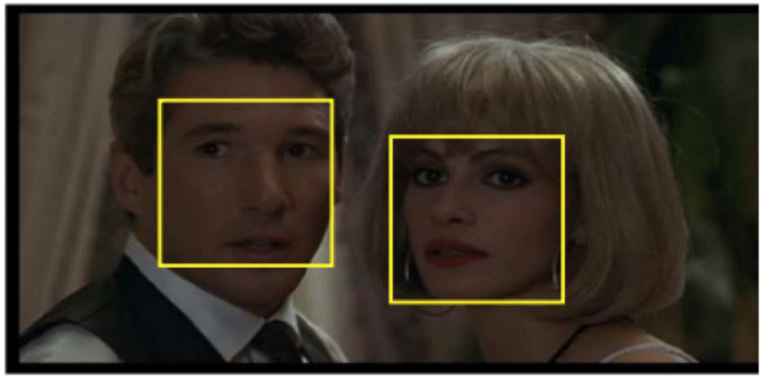
- In classification problems, each entity in some domain can be placed in one of a discrete set of categories: yes/no, friend/foe, good/bad/indifferent, blue/red/green, etc.
- Given a training set of labeled entities, develop a rule for assigning labels to entities in a test set
- Many variations on this theme:
  - binary classification
  - multi-category classification
- Many criteria to assess rules and their predictions
  - overall errors
  - costs associated with different kinds of errors

Dataset

- Each training data point to be classified is represented as a pair (x, y):
  - where x is a description of the object : f**eature vector**
  - where y is a **label** (assumed binary for now $y = \{+1, -1\}$)
- Success or failure of a machine learning classifier often depends on **choosing good set of features or descriptions of objects**
  - the choice of description can also be viewed as a learning problem
  - but good human intuitions are often needed here

Binary classification

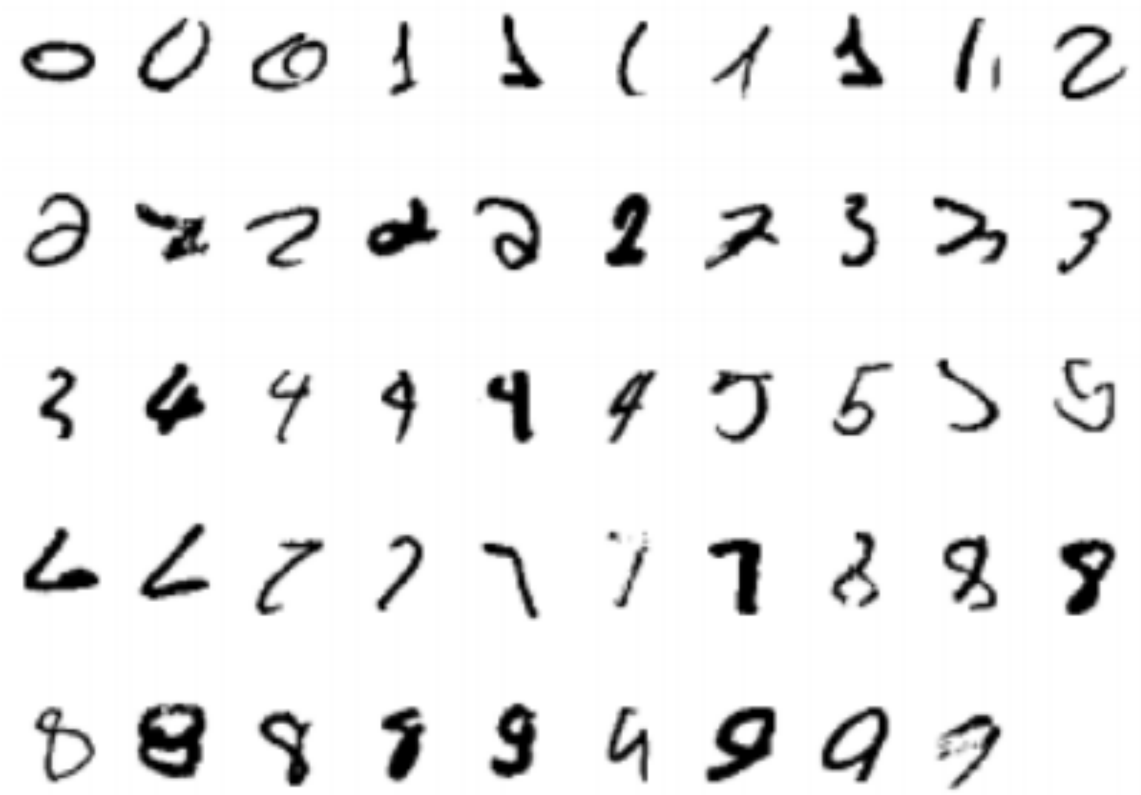




Face detection:

1. Facial part
2. Non facial part

Multiclass classification



Hand written digit classification

Class label : [0,1,2,…,9]

Examples of classifications

- Fraud detection (input: account activity, classes: fraud / no fraud)
- Web page spam detection (input: HTML/rendered page, classes: spam / ham)
- Speech recognition and speaker recognition (input: waveform, classes: phonemes or words)
- Medical diagnosis (input: symptoms, classes: diseases)
- Automatic essay grader (input: document, classes: grades)
- Customer service email routing and foldering
- Link prediction in social networks
- Catalytic activity in drug design
- … many many more

- Classification is an important commercial technology

State-of-art Classifiers

I) Instance-based methods:
  1) Nearest neighbor

II) Probabilistic models:
  1) Naïve Bayes
  2) Logistic Regression

III) Linear Models:
  1) Perceptron
  2) Support Vector Machine

IV) Decision Models:
  1) Decision Trees
  2) Boosted Decision Trees
  3) Random Forest

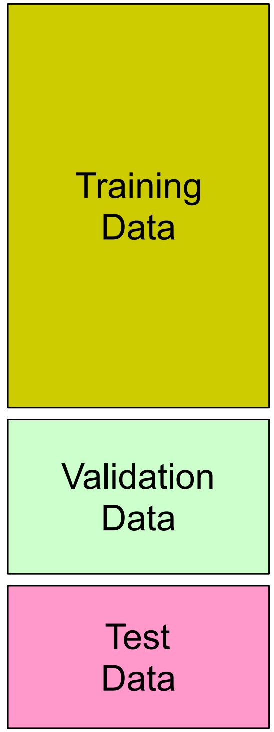
Training, Validation and Test

- Data: labeled instances, e.g. emails marked spam/ham
  - Training set
  - Validation set
  - Test set

- Training
  - **Estimate parameters** on training set
  - **Tune hyperparameters** on validation set
  - **Report results** on test set
  - Anything short of this yields **over-optimistic claims**

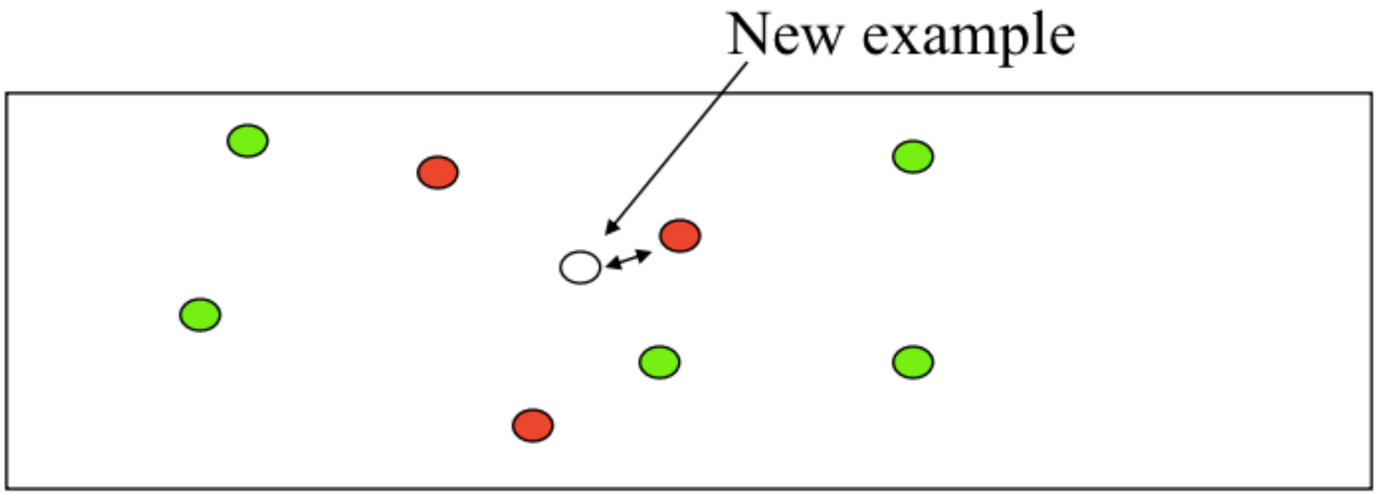
- Evaluation
  - Many different metrics
  - Ideally, the criteria used to train the classifier should be closely related to those used to evaluate the classifier

- Statistical issues
  - Want a classifier which does well on *test* data
  - Overfitting: fitting the training data very closely, but not generalizing well
  - Error bars: want realistic (conservative) estimates of accuracy

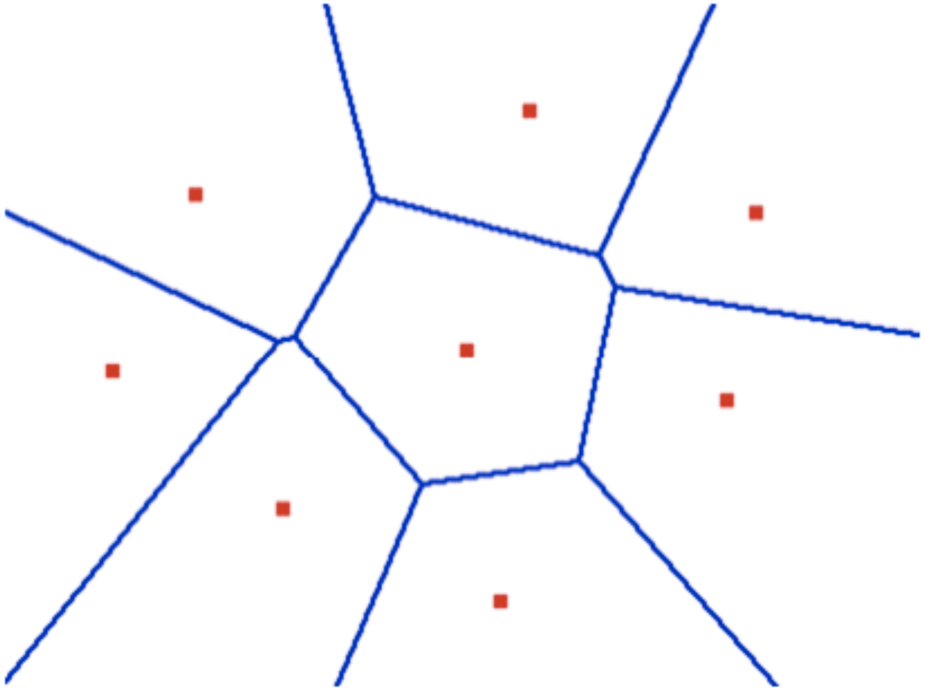


Nearest Neighbour Classifier

- **Remember** all training data
- Decide a **distance** function
- When a test data point comes find the nearest neighbour of the **closest point from training data**
- Assign the **label of nearest neighbour** to the test data

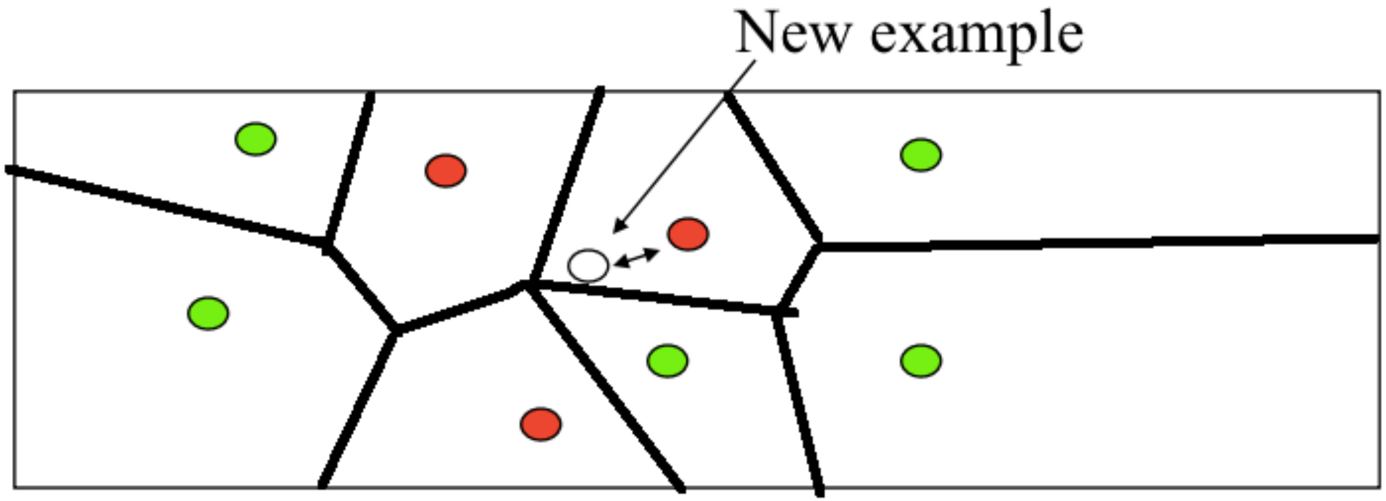


Voronoi Diagram



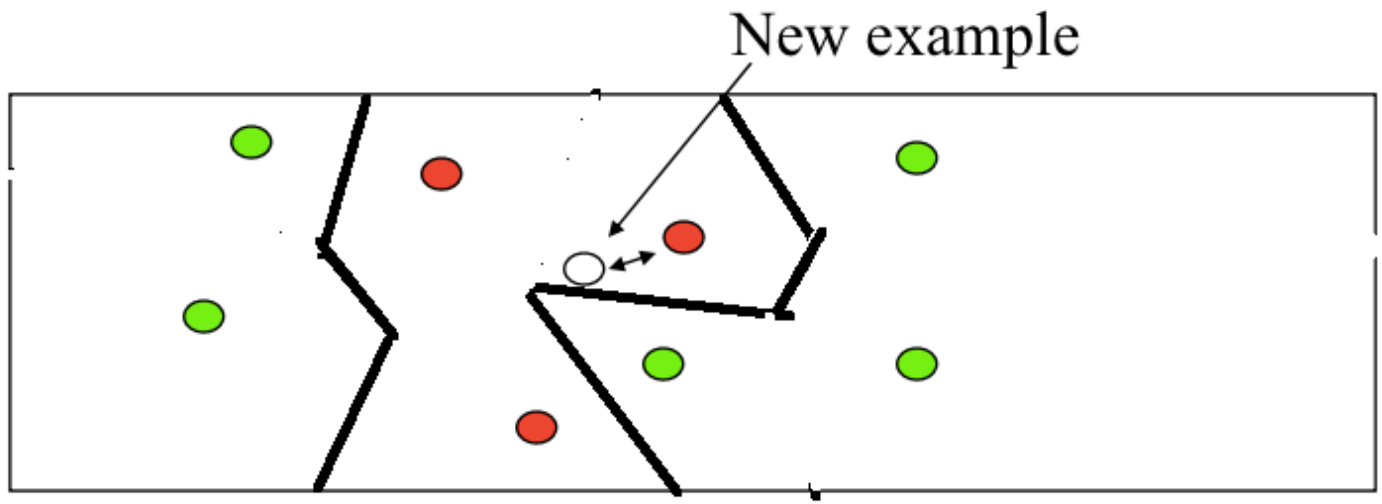
Given a set of points, a Voronoi diagram describes the areas that are nearest to any given point.

Voronoi Diagram

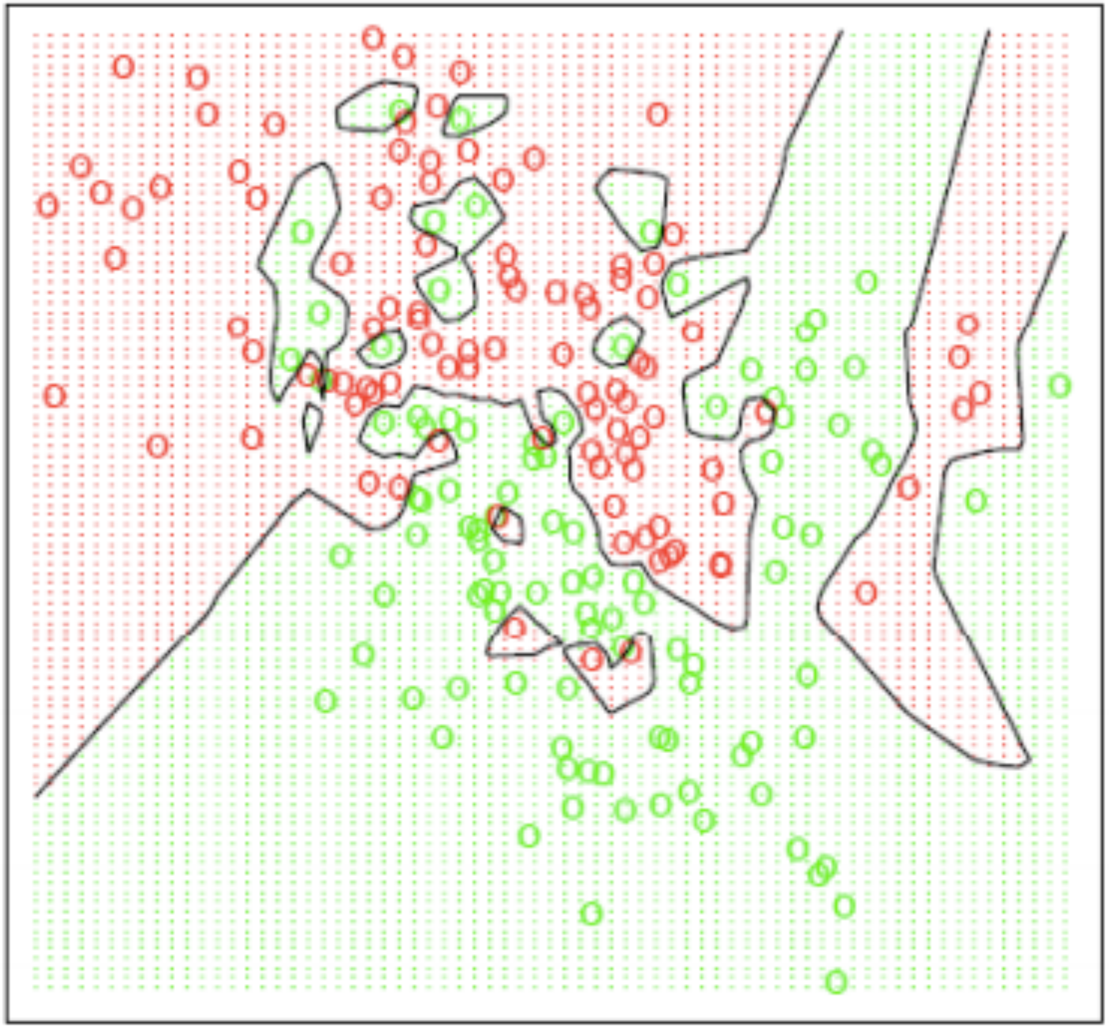


Given a set of points, a Voronoi diagram describes the areas that are nearest to any given point.

Decision Boundary : Voronoi diagram



NN classifier

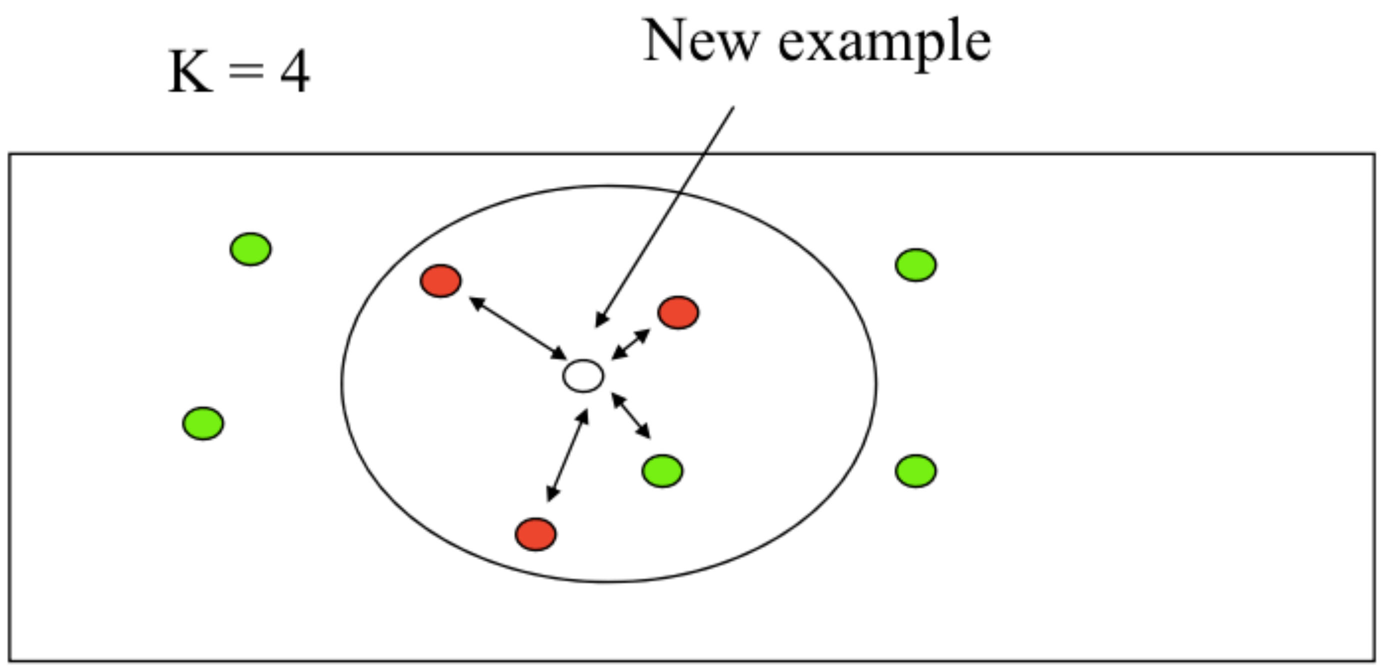


- **Overfitting**

- Solution :
    - Regularisation
    - Smoothing the boundary
    - **k neighbours**

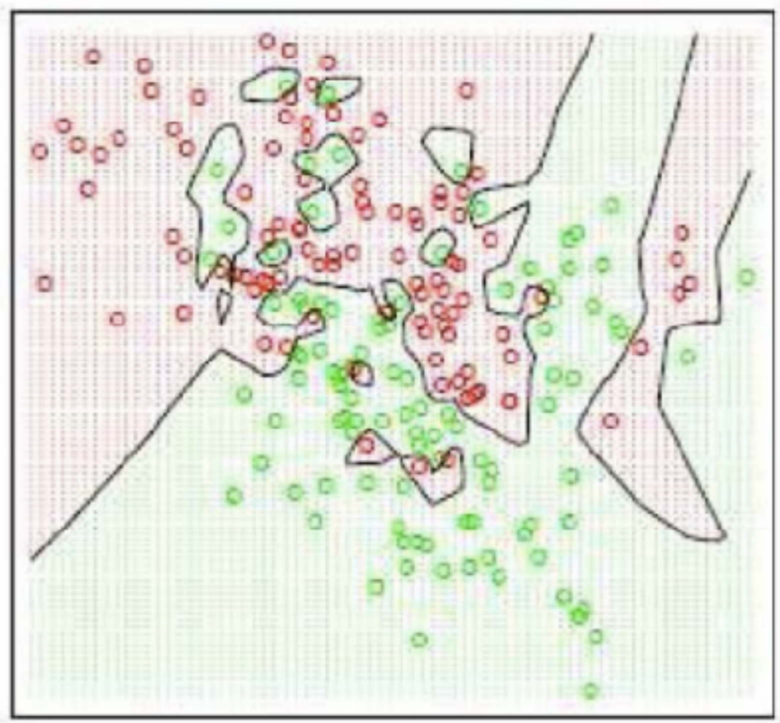
k-NN classifier

- Find the k nearest neighbors
- Have them **vote**
- Has a smoothing effect
- This is especially good when there is noise in the class labels.



Effect of "K"

K=1



K=15

Figures from Hastie, Tibshirani and Friedman (Elements of Statistical Learning)

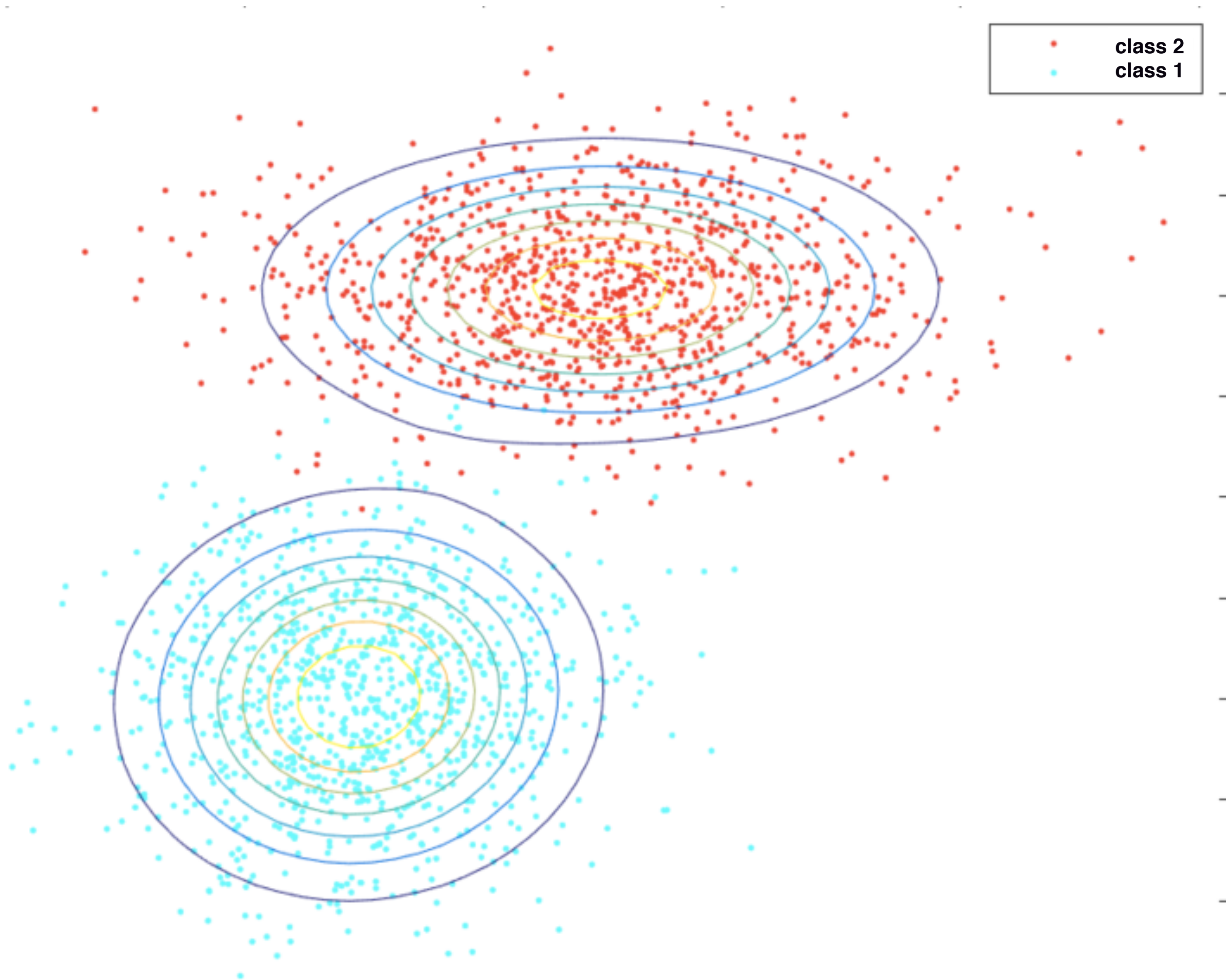
As K increases:

- Classification boundary becomes smoother
- Training error can increase

Choose (learn) K by **cross-validation**

- Split training data into training and validation
- Hold out validation data and measure error on this

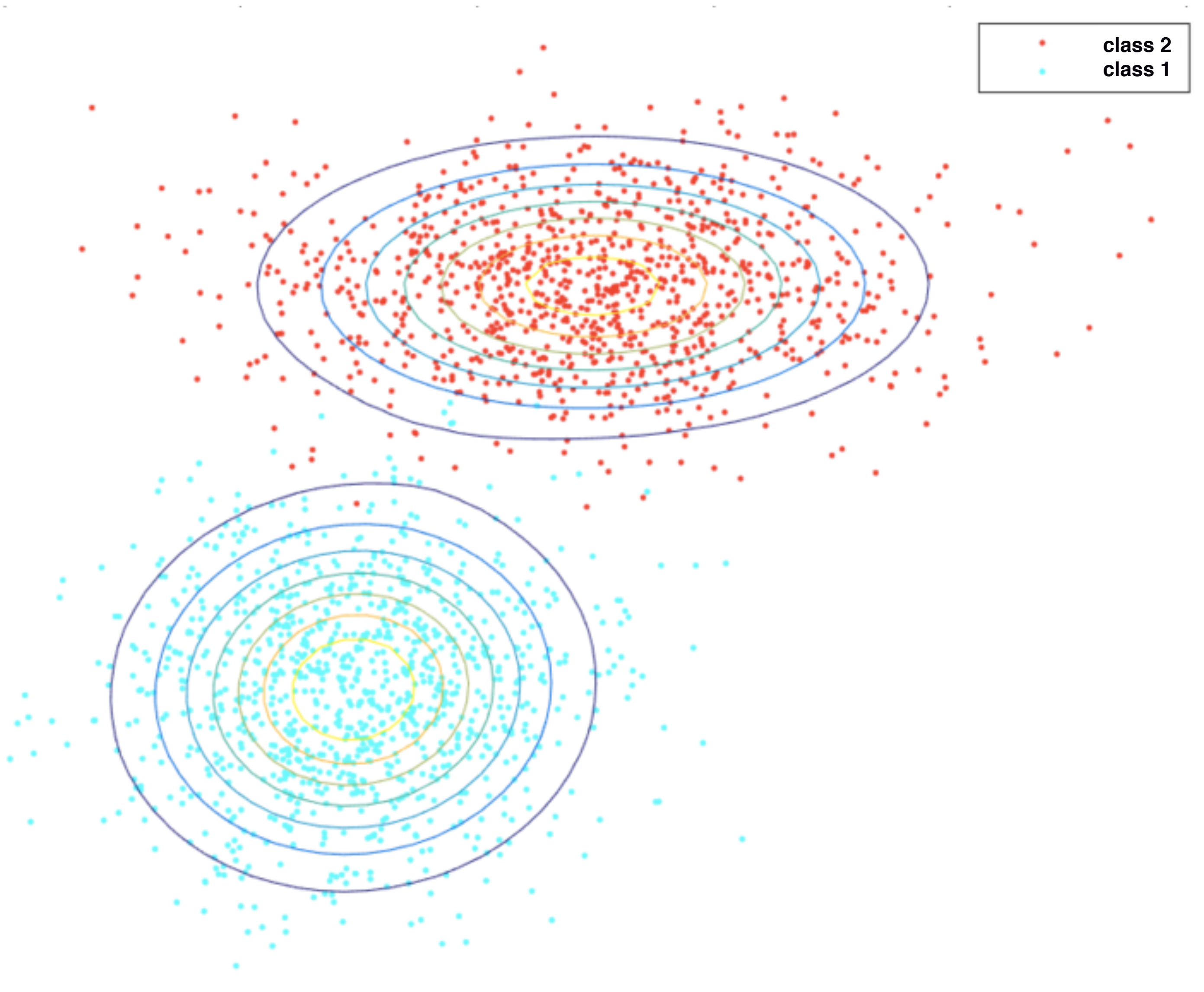
Probabilistic Decision Boundary



Probabilistic Decision Boundary

prior probability
$P(C_1)$ and $P(C_2)$

$P(C_1)$= fraction points having cyan colour



Decision Rule with prior probability

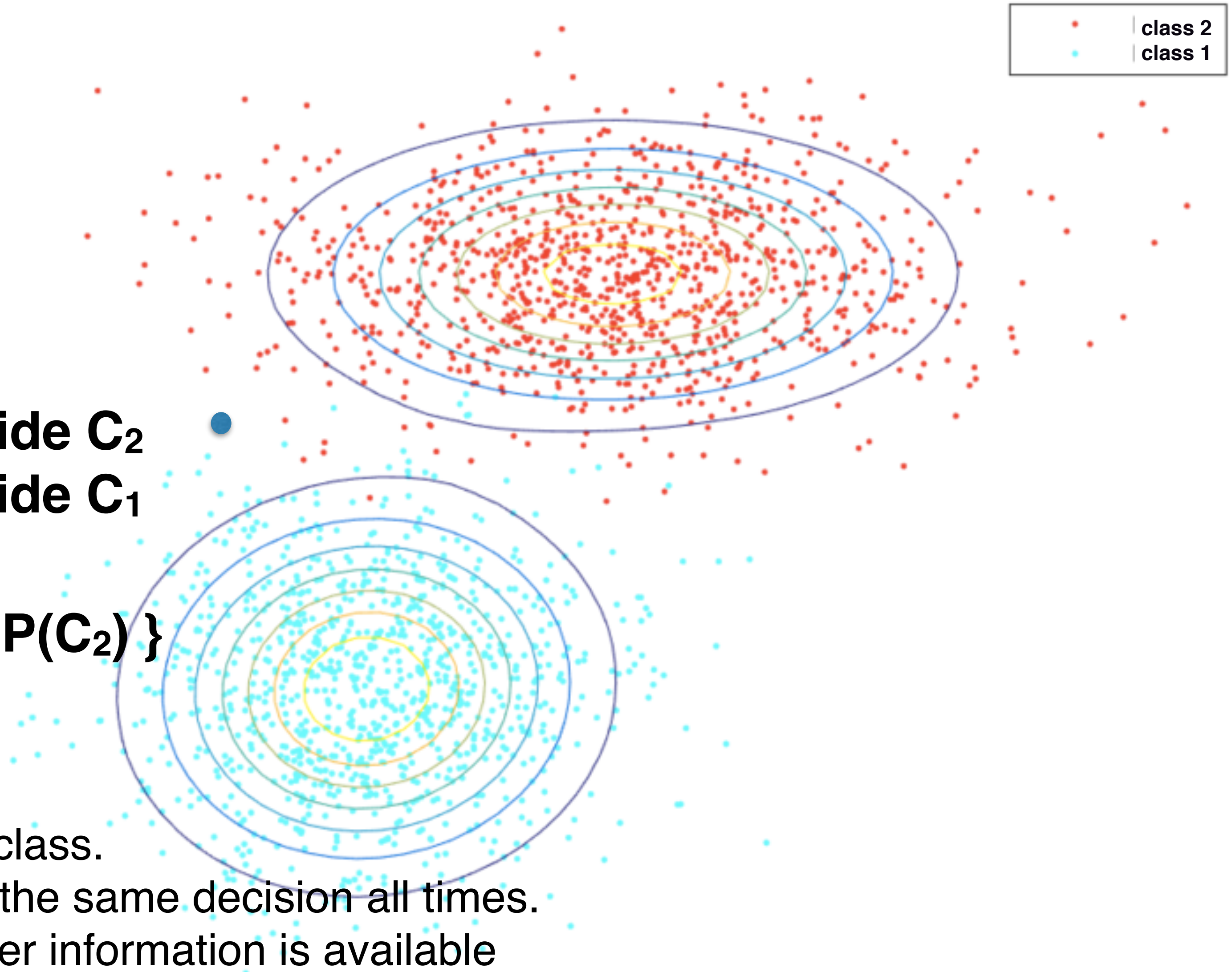
Decide:

Label =1 [class C_1]

if $P(C_1) > P(C_2)$

Label =-1 [class C_2]

if $P(C_2) > P(C_1)$



$$\textbf{P(error)=} \begin{cases} P(C_1) \text{ if decide } C_2 \\ P(C_2) \text{ if decide } C_1 \end{cases}$$

$$\textbf{P(error)= min \{ } P(C_1), P(C_2) \textbf{ \}}$$

- Favours the most likely class.
- This rule will be making the same decision all times.
- – i.e., optimum if no other information is available

Classification error for random classifier

Decide:
Label =1 [class C_1]
if $P(C_1) > P(C_2)$
Label =-1 [class C_2]
if $P(C_2) > P(C_1)$

P(error)= **$P(C_1)$ if decide C_2**
$P(C_2)$ if decide C_1

P(error)= min { $P(C_1)$, $P(C_2)$ }

Binary classification

Random classifiers

Decide:
Label =1 and Label =-1
randomly with probability 0.5

Assumes $P(C_1) = P(C_2)=0.5$

p(error)= min { $P(C_1)$, $P(C_2)$ } = 0.5

Next Class

- 17/8
  - Bayes Classifier, Naive Bayes Classifier
  - Logistic Regression