Lecture 8: Classification

Damian Matuszewski

damian.matuszewski@it.uu.se

Image analysis fundamental steps

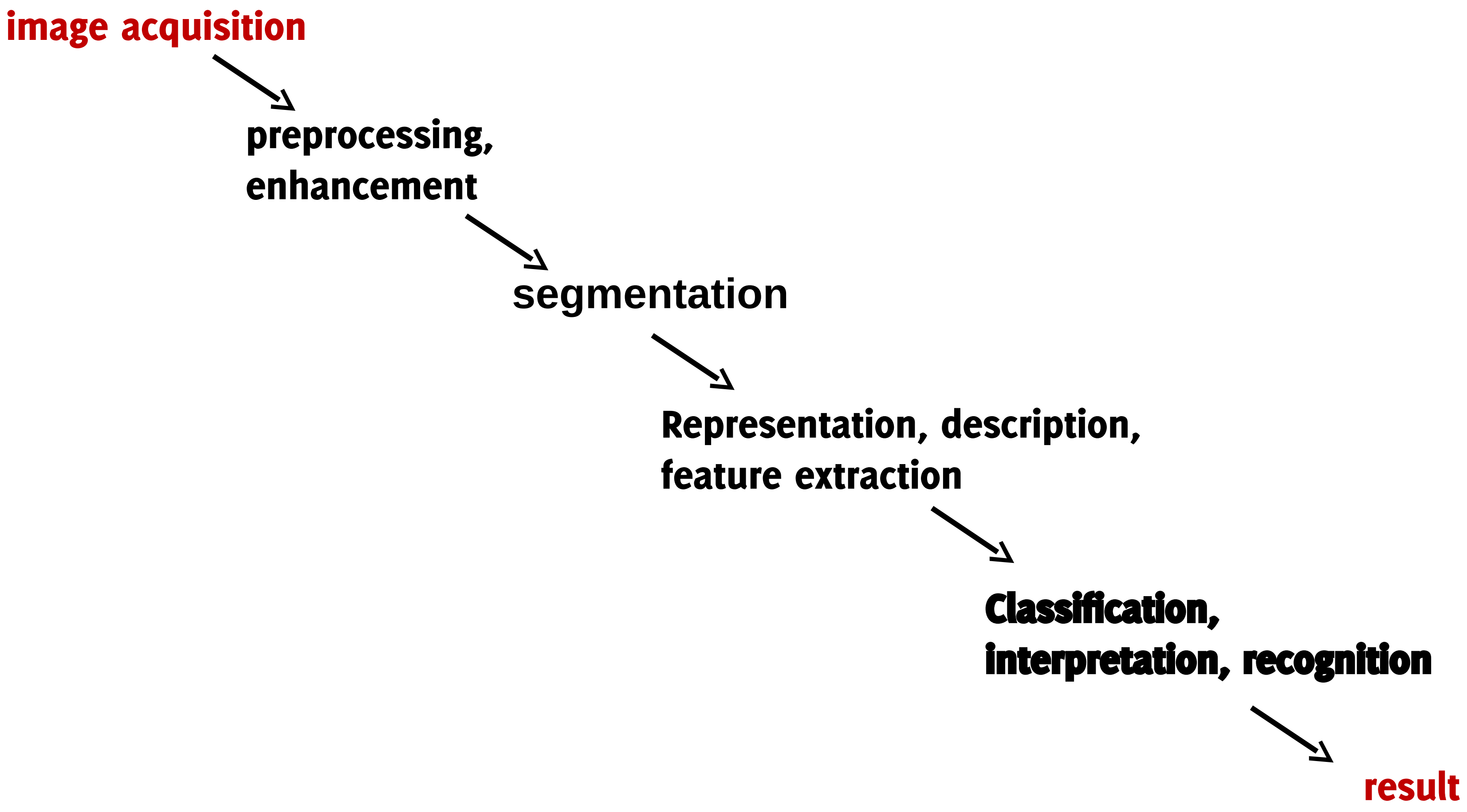


image acquisition

preprocessing,
enhancement

segmentation

Representation, description,
feature extraction

Classification,
interpretation, recognition

result

In this episode of Image Analysis season 1...

An overview of classification

Basics of classification

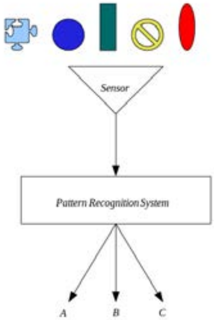
How to choose appropriate features (descriptors)

How to perform classification

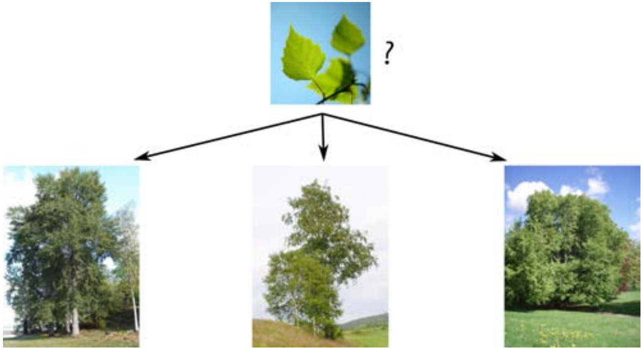
Classifiers

What is classification?

Classification is a process in which individual items (objects/patterns/image regions/pixels) are grouped based on the similarity between the item and the description of the group



Classification - example



Terminology

Object = pattern = point = sample = vector

Feature = descriptor = attribute = measurement

Classifier = decision function (boundary)

Class

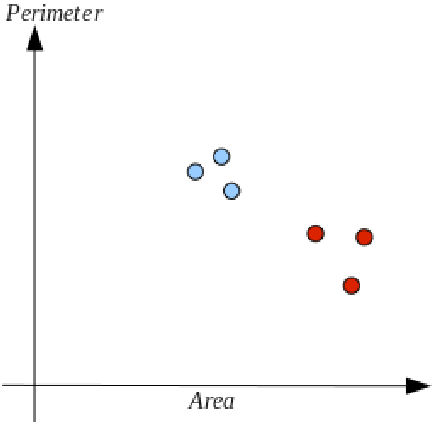
Cluster

Dataset

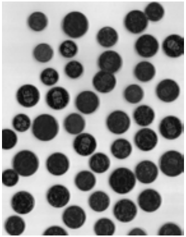
By measuring the features of many objects we construct dataset

Object	Perimeter	Area	Label
Apple 1	25	80	1
Apple 2	30	78	1
Apple 3	29	81	1
Pear 1	52	65	2
Pear 2	48	66	2
Pear 3	51	63	2

Data set and a representation in the feature space



Feature set



Colour?

Area?

Perimeter?

...

Features (Descriptors)

Measuring certain characteristic properties

Discriminating (effective) features

Independent features

Features:

- area, perimeter
- texture
- color
- ...

What are good features?

Each pattern is represented in terms of n features $\mathbf{x} = (x_1, ..., x_n)$

The goal is to choose those features that allow pattern vectors belonging to different classes to occupy compact and disjoint regions

It is application dependent

You might try many, many features, until you find the right ones

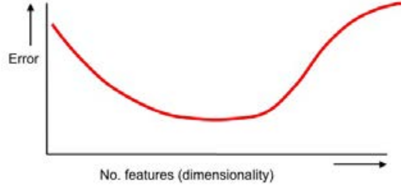
Often, people compute 100s of features, and put them all in a classifier

- "The classifier will figure out which ones are good"
- This is wrong!!!

Peaking phenomenon (Curse of dimensionality)

Additional features may actually degrade the performance of a classifier

This paradox is called peaking phenomenon (curse of dimensionality)



Dimensionality reduction

Keep the number of features as small as possible

- Measurements cost
- Accuracy

Simplify pattern representation

The resulting classifier will be faster and will use less memory

A reduction in the number of features may lead to a loss in the discriminatory power and thereby lower the accuracy

Dimensionality reduction – Feature extraction and feature selection

Feature extraction is the process of generating features to be used in the selection and classification tasks

Feature selection methods choose features from the original set based on some criteria

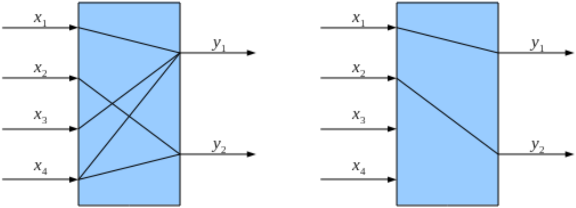


Figure: Feature extraction (for example PCA) and feature selection.

Feature extraction by Principal Component Analysis

Find the direction with the maximal variance

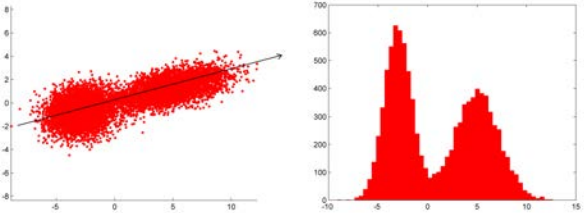


Figure: Two-dimensional feature space, principal direction and feature values projected onto principal axis

Feature selection methods

Exhaustive search

Best individual features

Sequential forward selection

Sequential backward selection

Exhaustive search

Evaluate all $\binom{d}{m}$ possible subsets

Guaranteed to find the optimal subset

Very expensive!

Example: If $d = 50$, $m = 5$ then there are 2 118 760 possible subset (classifications) to evaluate

Best individual features

Evaluate all m features individually

Select the best $l \leq m$ individual features

Simple and not likely to lead to an optimal subset

Example:

- Feature 1 is best
- Feature 2 is best
- Maybe features 3 and 4 outperform features 1 and 2!

Sequential forward selection

Selects the best feature and then add one feature at a time

Once a feature is retained, it cannot be discarded

Computationally fast (for a subset of size 2 examine $d - 1$ possible subsets)

Sequential backward selection

Starts with all d features and successive delete one feature at a time

Once a feature is deleted, it cannot be brought back into the optimal subset

Require more computation time than sequential forward selection

Choice of a criterion function

The main issues in dimensionality reduction is the choice of a criterion function

A most commonly used is the classification error of a feature subset

Another option is a measure of distance between two distributions f_1 and f_2

Mahalanobis distance: Main assumption is that Gaussian distributions have equal covariance matrix Σ

$$D_M(f_1, f_2) = (m_1 - m_2)^T \Sigma^{-1} (m_1 - m_2)$$

m_1 is a mean of objects in class 1
m_2 is a mean of objects in class 2

Variance and covariance

Variance : spread a randomness for class

Covariance : influence (dependency) between different features

Covariance: $cov(x_i, x_j) = \frac{1}{n-1} \sum_{k=1}^{n} (x_{i,k} - x_{mean_i})(x_{j,k} - x_{mean_j})$

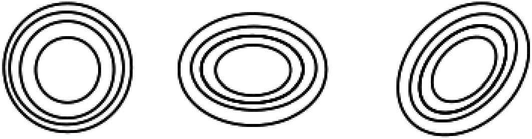
$$\Sigma = \left[\begin{array}{ccc} cov(x_1, x_1) & cov(x_1, x_2) & cov(x_1, x_3) \\ cov(x_2, x_1) & cov(x_2, x_2) & cov(x_2, x_3) \\ cov(x_3, x_1) & cov(x_3, x_2) & cov(x_3, x_3) \end{array} \right]$$

Density functions

Equal variance

Different variance

Covariance is different from zero



Assumptions on covariance matrix

Case 1 (Minimal distance)

- No covariance, equal variance

Case 2 (Equal covariance)

- Same covariance for all classes

Case 3 (Uncorrelated)

- No covariance, different variance

Case 4 (General)

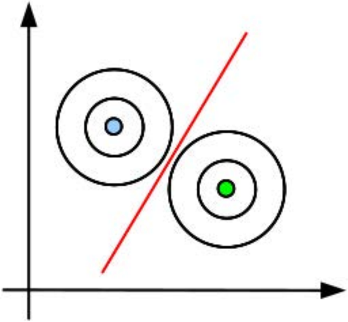
- Different covariances

Case 1

Independent features

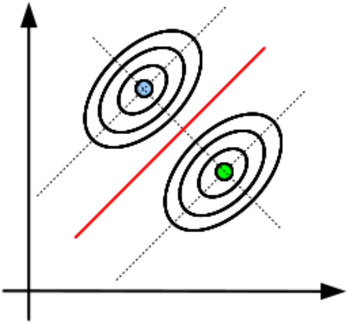
Equal variance

Minimal distance classifier



Case 2

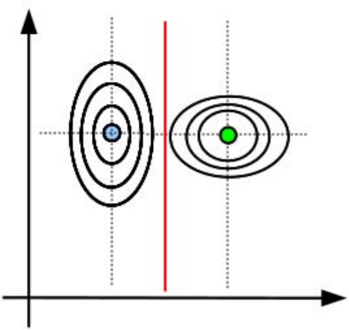
Equal covariances for all classes



Case 3

Independent features - no covariance

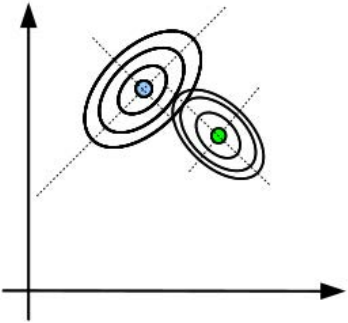
Different variance for different features



Case 4

General

Complicated decision boundaries



Supervised vs. Unsupervised classification

Supervised

- First apply knowledge, then classify

Unsupervised

- First classify (cluster / group), then apply knowledge

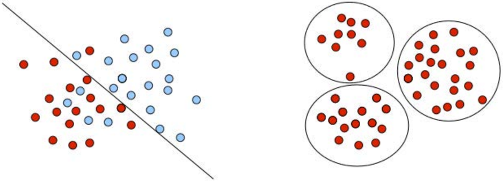


Figure: Supervised and unsupervised classification.

Object-wise and pixel-wise classification

Object-wise classification

- Uses shape information to describe patterns
- Size, mean intensity, mean color, etc.

Pixel-wise classification

- Uses information from individual pixels
- Intensity, color, texture, spectral information

Object-wise classification

Segment the image into regions and label them

Extract features for each pattern

Train classifier on examples with known class to find discriminant function in the feature space

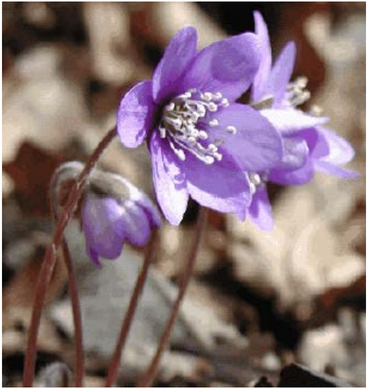
For new examples decide their class using the discriminant function

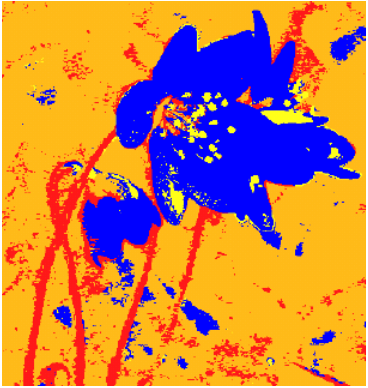
Pixel-wise classification

256×256 patterns (pixels)

3 features (red, green and blue color)

Four classes (stamen, leaf, stalk and background)





Pixel-wise classification

The pattern is a pixel in a non-segmented image

Extract (calculate) features for each pattern (pixel) e.g., color, gray-level representation of texture

Train classifier

New samples are classified by classifier

Train and classify

Training

- Find rules and discriminant function that separate patterns in different classes using known examples

Classification

- Take a new unknown example and put it into the correct class using discriminant function

Training data

Training data can be obtained from available samples

>> divide data into training, validation and testing sets

>> tune the classifier parameters measuring performance on the validation set

Training set should be representative of all variation in data (often difficult to satisfy!)

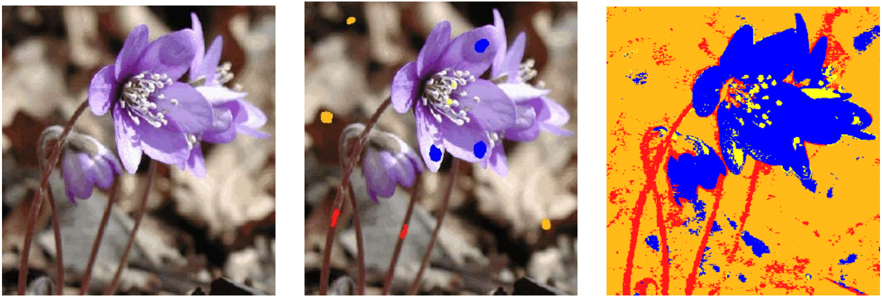
>> Number of training samples should not be too small!

>> Many classifiers require balanced training set

Classify / evaluate the final performance on samples which are not used during the training

The performance of a classifier depends on the number of available training samples as well as the specific values of the samples

How to choose appropriate training set?



How to choose appropriate training set?

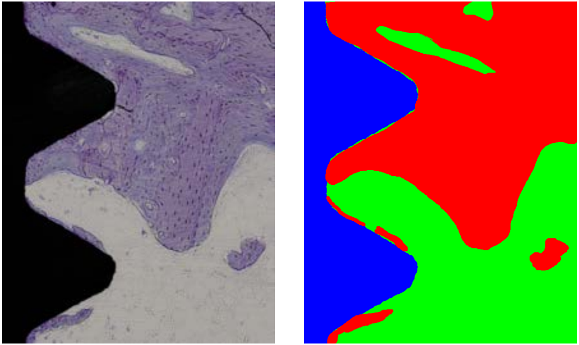


Figure: Original image and training image.

How to choose appropriate training set?

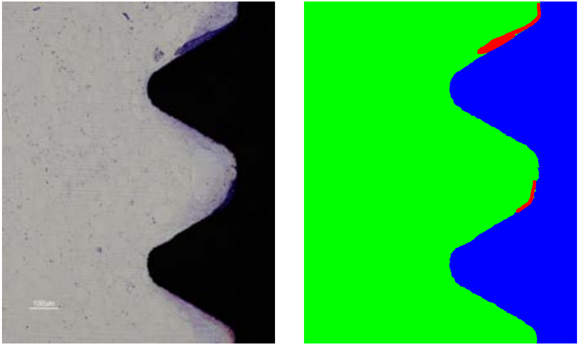


Figure: Original image and training image.

Classifiers

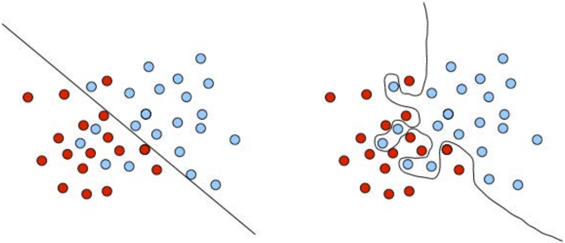
Once a feature selection finds a proper representation, a classifier can be designed using a number of possible approaches

The performance of the classifier depends on the interrelationship between sample size, number of features and classifier complexity

The choice of a classifier is a difficult problem!!!

It is often based on which classifier(s) happen to be available or best known to the user

Decision boundary



Linear and quadratic classifier

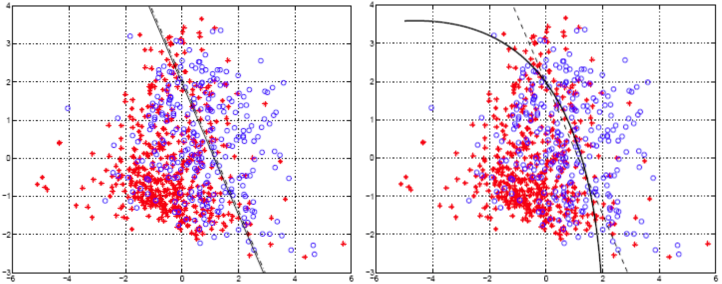


Figure: Linear (left) and quadratic (right) classifier.

Most commonly used classifiers

The user can modify several associated parameters and criterion function

There exist some classification problem for which they are the best choice

Classifiers:

>> Bayesian classifiers
>> Nearest neighbour classifier
>> Support vector machine
>> Linear and quadratic discriminant analysis
>> Neural networks
>> Decision trees
>> Random forest
>>...

Bayesian classifiers

Based on a priori knowledge of class probability

Cost of errors

Minimum distance classifier

Maximum likelihood classifier

Minimum distance classifier

Each class is represented by its mean vector

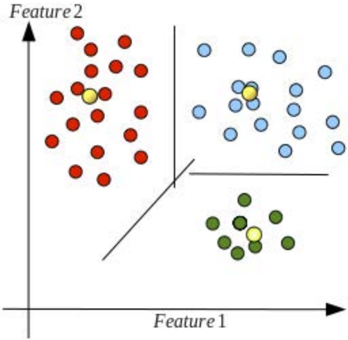
Training is done using the objects (pixels) of known class

Mean of the feature vectors for the object within the class is calculated

New objects are classified by finding the closest mean vector

$$d_j(\mathbf{x}) = \mathbf{x}^T \mathbf{m}_j - \frac{1}{2}\mathbf{m}_j^T \mathbf{m}_j$$

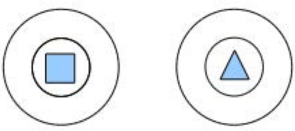
Minimum distance classifier



Limitations of Minimum distance classifier

Useful when distance between means is large compared to randomness of each class with respect to its mean

Optimal performance when distributions of the classes form spherical shape



Maximal likelihood classifier

Bayes' formula

$$p(A|B) = \frac{p(A)p(B|A)}{p(B)}$$

Use a priori knowledge obtained from training data

$p(A)$, $p(B)$: probabilities of observing A and B without regards to each other

$p(A|B)$: Conditional probability, probability of obtaining event A given that B is true

Maximal likelihood classifier

Bayes' formula

$$p(w_i|x) = \frac{p(x|w_i)p(w_i)}{p(x)}$$

$p(w_i|x)$: Posterior probability for each class i of measuring a particular feature x

$p(x)$: how frequently we will actually measure an object with value x (Scale Factor)

$p(w_i)$: prior probabilities (known from training data)

$p(x|w_i)$: class conditional probability, the proba. of measuring the value x, given that the object is in class i

Maximal likelihood classifier

Bayes' formula

$$p(w_i|x) = \frac{p(x|w_i)p(w_i)}{p(x)}$$

Decision rule: If

$$\frac{p(x|w_1)p(w_1)}{p(x)} > \frac{p(x|w_2)p(w_2)}{p(x)}$$

Choose w_1
else Choose w_2

Maximal likelihood classifier

Bayes' formula

$$p(w_i|x) = \frac{p(x|w_i)p(w_i)}{p(x)}$$

If not other specified, assume that distribution within each class is Gaussian

If we have a large number of samples in each class, than the proba. density function will be Gaussian in shape (Central Limit Theorem)

The distribution in each class can be described by a mean vector and covariance matrix

Maximal likelihood classifier

For each class from training data compute:

- Mean vector
- Covariance matrix

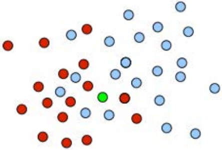
Form decision function for each class

New objects are classified to class with highest probability

Nearest neighbour

Stores all training samples

Assigns pattern to majority class among k nearest neighbour



$k-$nearest neighbour

Metric dependent

Might be slow

The function is approximated only locally

The best choice of k depends on data

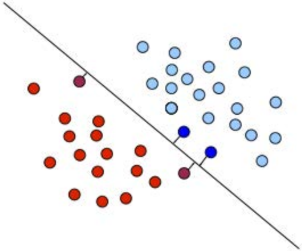
Sensitive to outliers

Larger values of k reduce effects of noise, but make boundaries between the classes less distinct

Support vector machine

Linear classifier
>> kernel tricks for non-linearity

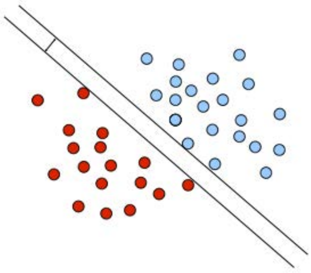
Based only on samples close to boundary



Support vector machine

It is primarily a two class classifier
>> tricks for multi-class classification

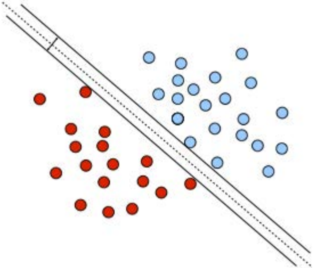
Maximize the width of the margin (empty area) between the classes by selecting a minimum number of support vectors



Support vector machine

Support vectors define the classification function (maximizing the margin ⇒ the number of support vector is minimized)

Metric dependent
>> most often used kernel: Radial Basis Function



Discriminant functions

A discriminant function for a class is a function that will yield larger values than functions for other classes if the pattern belongs to the class

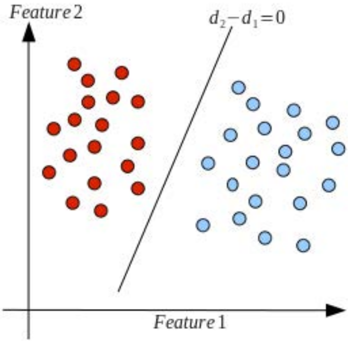
$$d_i(\mathbf{x}) > d_j(\mathbf{x}) \quad j = 1, 2, ..., N; \quad j \neq i$$

For N pattern classes, we have N discriminant functions

The decision boundary between class i and class j is

$$d_j(\mathbf{x}) - d_i(\mathbf{x}) = 0$$

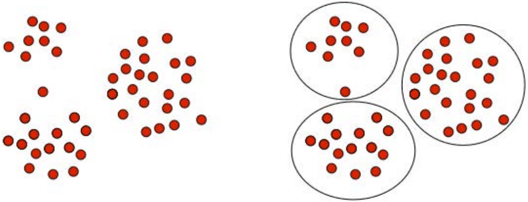
Decision boundary



Unsupervised classification

Why unsupervised? Difficult, expensive or even impossible to rely label training sample with its true category (It is not possible to obtain ground truth)

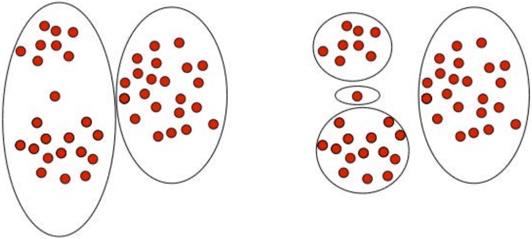
Assumption: Patterns within a cluster are more similar to each other than patterns belonging to different clusters



Unsupervised classification

How to determine the number of clusters K?

The number of clusters often depends on resolution (fine vs. coarse)



$K-$means

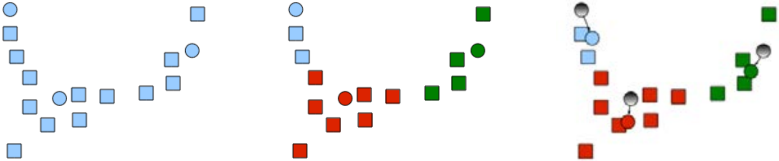
Step 1. Select an initial partition with K clusters. Repeat steps 2 through 4 until the cluster membership stabilizes

Step 2. Generate a new partition by assigning each pattern to its closest cluster center

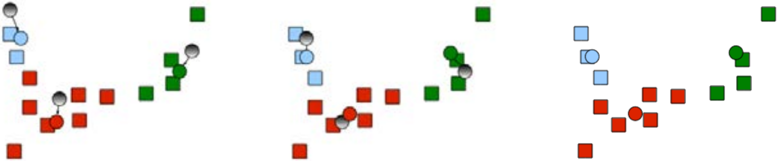
Step 3. Compute new cluster centres as the centroids of the clusters

Step 4. Repeat steps 2 and 3 until an optimum value of the criterion function is found (the cluster are stabilized)

$K-$means



$K-$means



$K-$means - disadvantages

Different initialization can result in different final clusters

Fixed number of clusters can make it difficult to predict what K should be

It is helpful to rerun the classification using the same as well as different K values, to compare the achieved results

10 different initializations for 2D data

For N-dimensional data 10 different initializations is often not enough!

Example

The following eight points $A1(2, 10)$, $A2(2, 5)$, $A3(8, 4)$, $A4(5, 8)$ $A5(7, 5)$ $A6(6, 4)$ $A7(1, 2)$ $A8(4, 9)$ should be classified into three clusters using $K-$means clustering. Initial cluster centres are: $A1(2, 10)$, $A4(5, 8)$ and $A7(1, 2)$. Find the three cluster centres after the first iteration.
The distance function between two points $A(x_a, y_a)$ and $B = (x_b, y_b)$ is defined as $d(A, B) = |x_a - x_b| + |y_a - y_b|$.

Summary and conclusions

Classification is important part of image analysis

It is highly application dependent - both in choice of features and classifiers

The use of more features is not a guarantee for better classification