



UPPSALA
UNIVERSITET

Classification

Computer Assisted Image Analysis I

Nataša Sladoje

Natasa.sladoje@it.uu.se



Centre for Image Analysis
Uppsala University

November 19, 2019





UPPSALA
UNIVERSITET

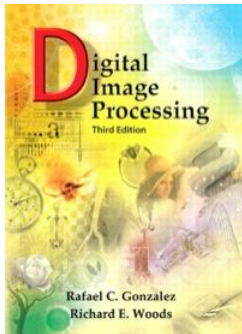
Today's lecture

- Classification - Examples and Problem statement
- Supervised classification
 - Nearest Neighbour classifier
 - Support Vector Machine (SVM)
 - Optimal (Bayes) classifier
 - Discriminant analysis
- Clustering (unsupervised classification)
 - K-means
- Concluding notes



UPPSALA
UNIVERSITET

Reading



Ch. 12 Object recognition

12.1 Patterns and Pattern Classes

12.2 Recognition Based on Decision-Theoretic Methods

12.2.1 Matching

12.2.2 Optimum Statistical Classifier

12.2.3 Neural Networks – Next time!

12.3 Structural Methods

12.3.1 Matching Shape Numbers

12.3.2 String Matching

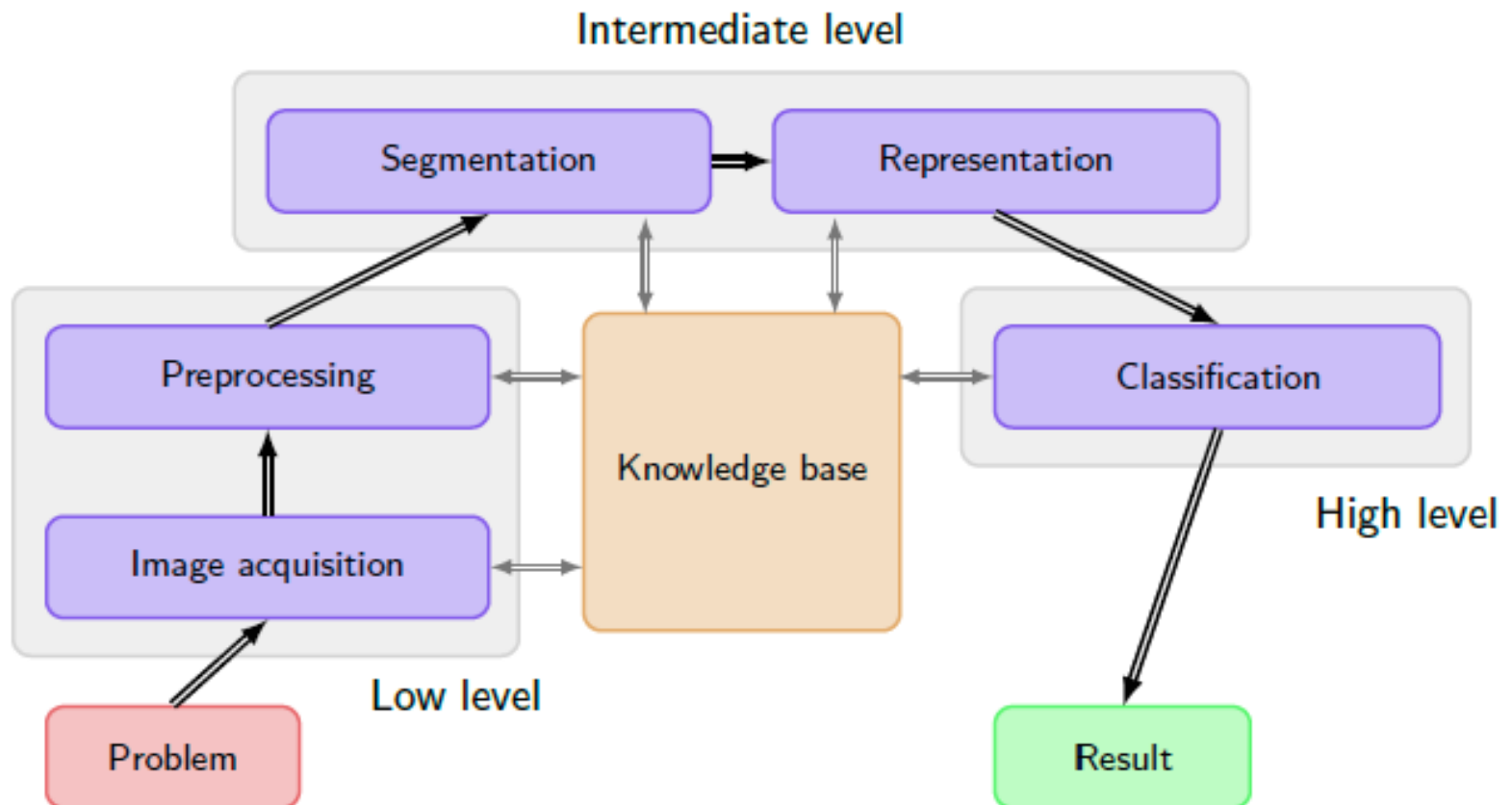


UPPSALA
UNIVERSITET

Image processing pipeline

and where we are in it today

Fundamental steps





UPPSALA
UNIVERSITET

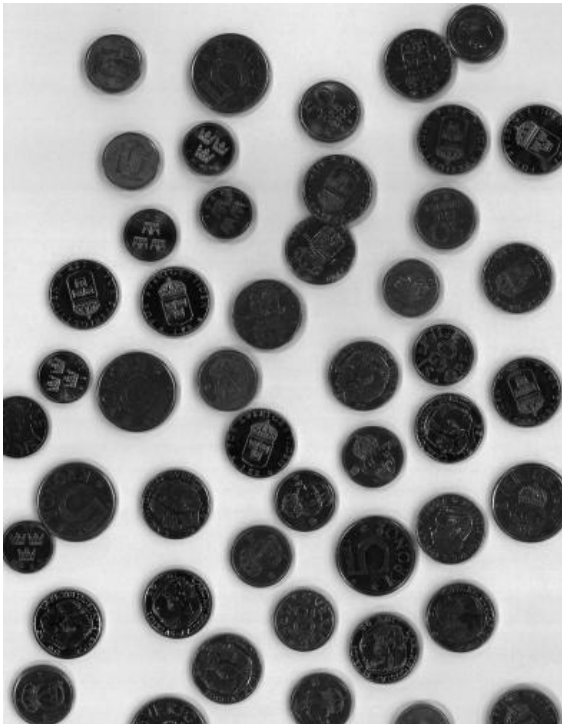
Classification

EXAMPLES AND PROBLEM STATEMENT



UPPSALA
UNIVERSITET

Classification, Example 1



Your task in Lab 3:

**Given the image with coins,
estimate the amount of money**

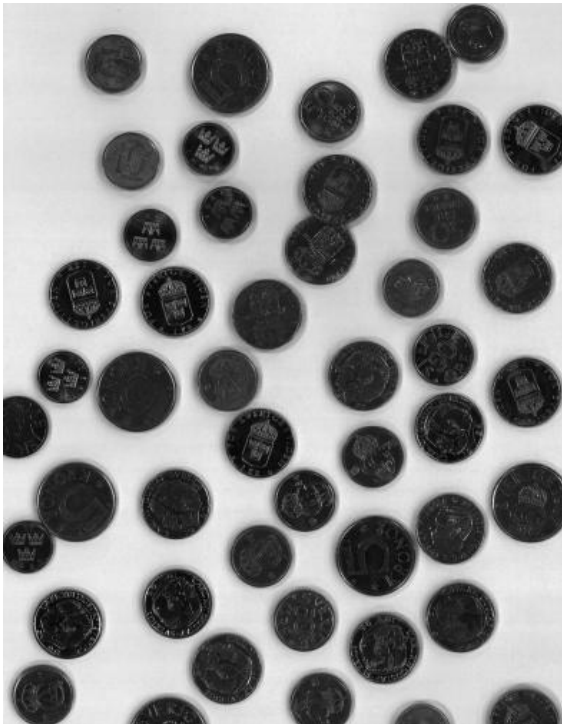


UPPSALA
UNIVERSITET

Classification, Example 1

Your task in Lab 3:

**Given the image with coins,
estimate the amount of money**



1. Segment
2. Measure the size
3. Group
4. Count
5. Interpret

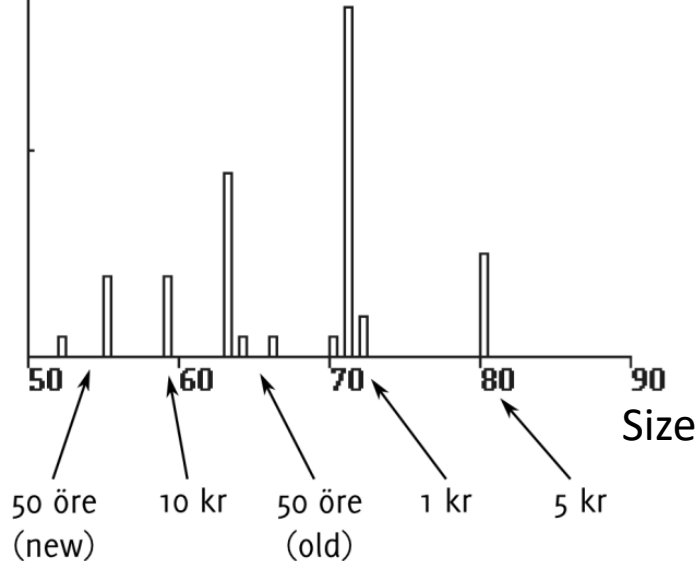


UPPSALA
UNIVERSITET

Classification, Example 1

No. of objects

20



Histogram of radii

Area (size) of the coin
– **discriminative feature**

Interpretation:
A priori knowledge about
relation between
the size of a coin
and
its nominal value.

Classification based on **one discriminative feature** and a **a priori knowledge**.



UPPSALA
UNIVERSITET

Classification, Example 2



Iris setosa



Iris versicolor



Iris virginica

Images: Wikipedia

***Iris* flower data set:** 50 samples from each of three species of *Iris*.
Four features were measured from each sample:
the length and the width of the sepals and petals, in centimeters.

R. A. Fisher (1936). "The use of multiple measurements in taxonomic problems". Annals of Eugenics. 7 (2): 179–188.



UPPSALA
UNIVERSITET

Classification, Example 2



Iris setosa



Iris versicolor



Iris virginica

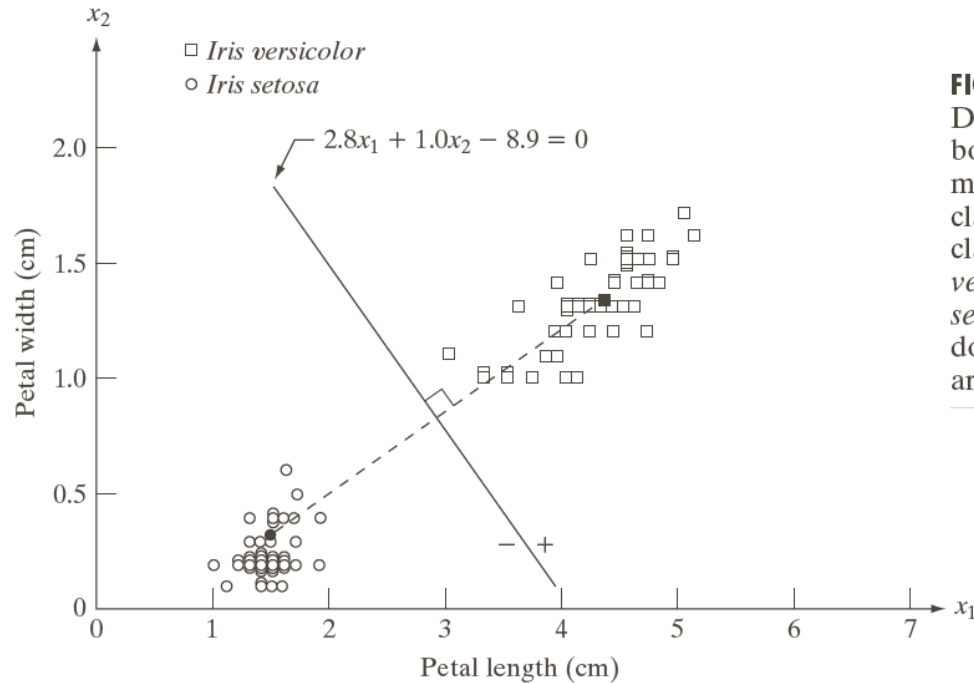
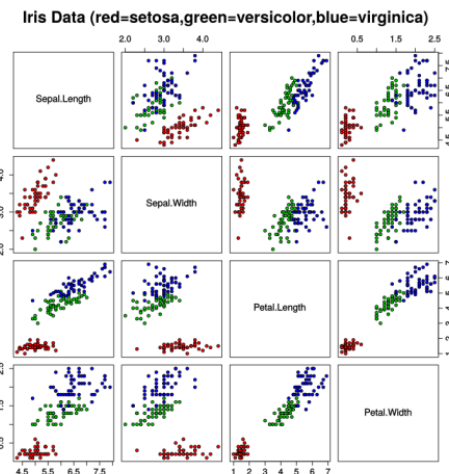


FIGURE 12.6
Decision boundary of minimum distance classifier for the classes of *Iris versicolor* and *Iris setosa*. The dark dot and square are the means.



R. C. Gonzalez and R. E. Woods,
"Digital Image Processing"

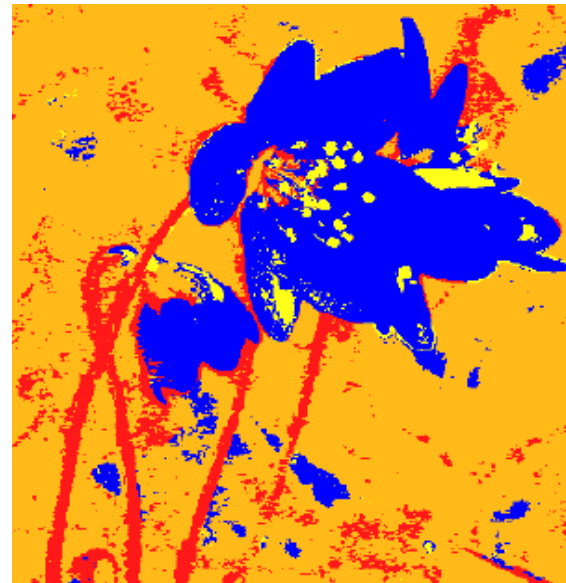


UPPSALA
UNIVERSITET

Classification, Example 3

Pixel-wise classification, Lab 4

Segmentation



Patterns: 256 x 256 pixels

Features: red, green, and blue color

Classes: stamen, leaf, stalk and background

Classification - problem statement

Classification is a process in which individual items

images

image regions (objects)

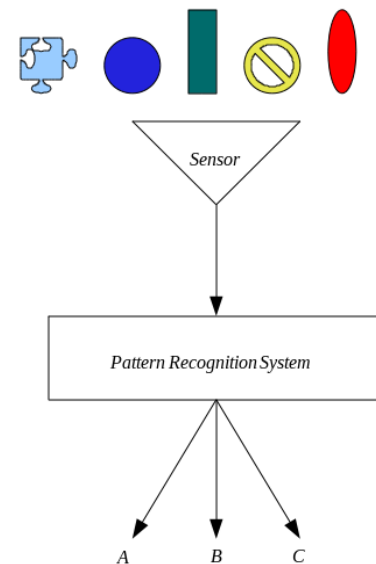
pixels

are **grouped** based on **similarity** between the item and the (description of the) group.

Measurements, descriptors, attributes: **Features**

Feature representations

of images, images regions (objects), pixels: **Patterns**





UPPSALA
UNIVERSITET

Feature space

Measure certain characteristic properties:

area, perimeter

texture

color

...

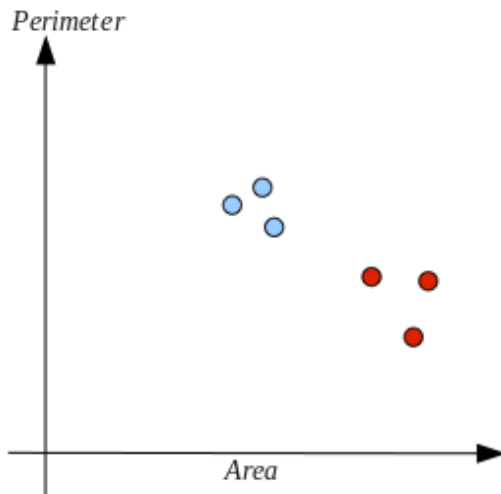
These are called **features**.

Find

discriminative features

independent features

to allow **pattern vectors** belonging to different classes
to occupy compact and disjoint regions.



Features are represented in the **feature space**.

Number of features – Dimension of the feature space

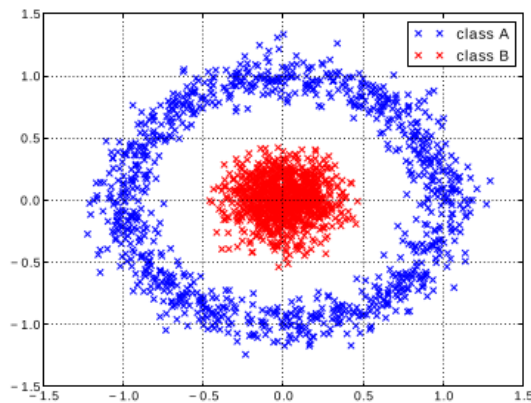


UPPSALA
UNIVERSITET

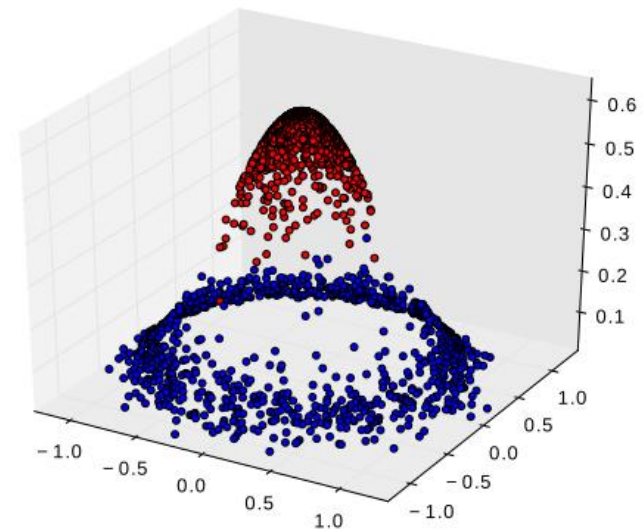
Feature space

Usually of high dimensionality -
hundreds of features may be observed!

Sometimes it is beneficial to increase dimensionality.



2D: Linearly non-separable dataset.



3D: Possible alternative representation
in a higher-dimensional space.



UPPSALA
UNIVERSITET

...but not always! Curse of dimensionality

Increasing dimensionality – Increasing the volume of the space.

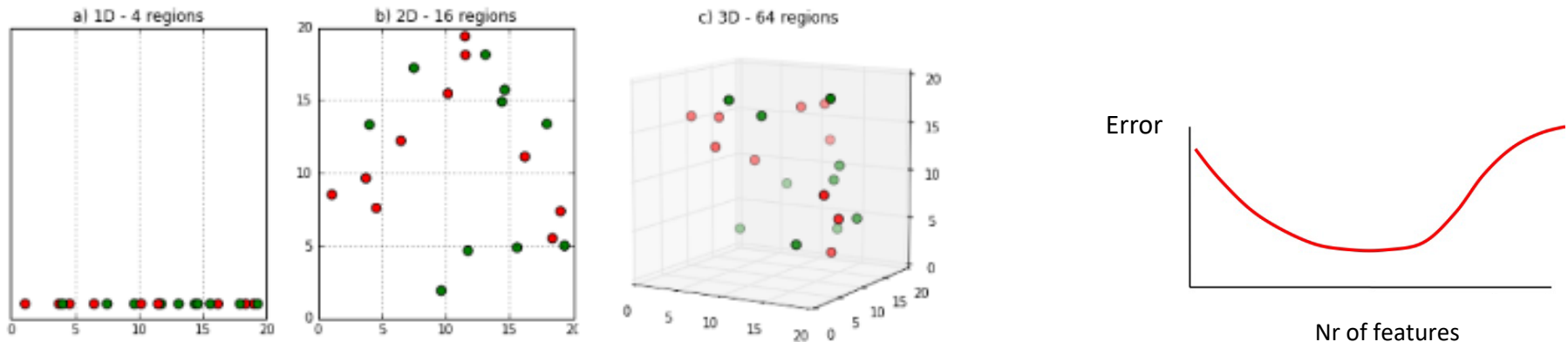


Image: Nikolay Manchev

Available data may become **too sparse** for statistical analysis.
Similarity and grouping appear less prominent.

Amount of data should* grow exponentially with the dimensionality.

*) According to classic statistical theories



UPPSALA
UNIVERSITET

Feature extraction and selection – dimensionality reduction

Feature extraction

The process of **generating features** to be used in classification.

Feature selection

The process of **choosing relevant features** from the original set.

Relevance depends on some predefined criteria.

Typically, discriminative features are relevant.

Tradeoff

Large number of features may lead to slower and less efficient classification.

Small number of features may lead to a decreased discriminatory power and lower accuracy.

Feature selection stays out of the scope of this lecture (course).

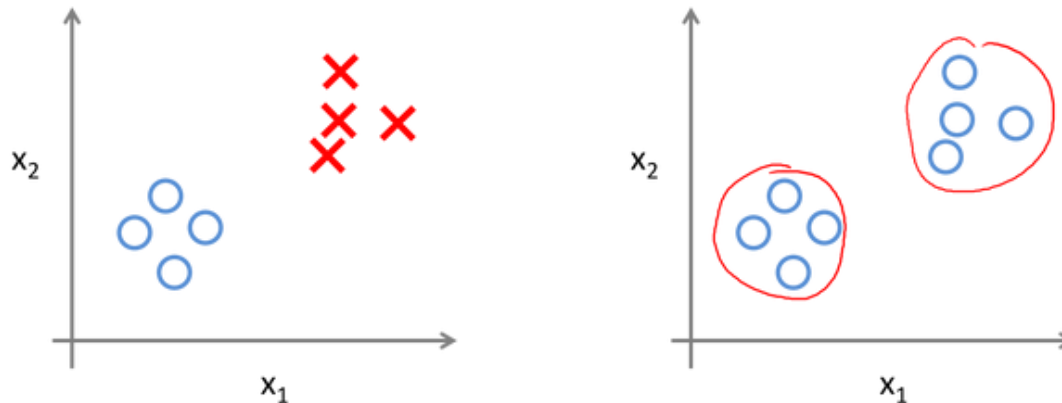
Supervised vs. unsupervised classification

Supervised

First apply knowledge, then group - **Classification**

Unsupervised

First group, then apply knowledge - **Clustering**



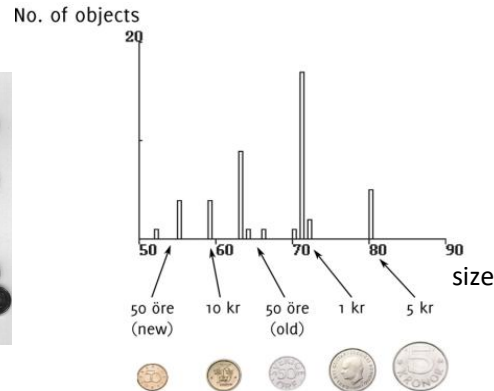
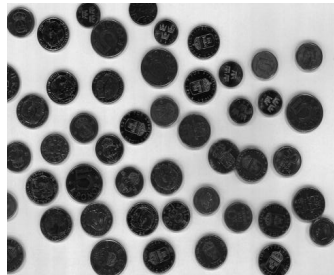
Supervised and unsupervised classification



UPPSALA
UNIVERSITET

Supervised or unsupervised?

Example 1

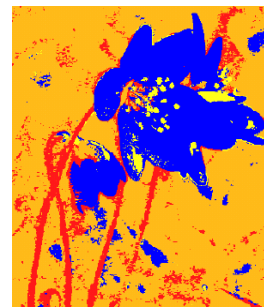


Example 2



Dataset Order	Sepal length	Sepal width	Petal length	Petal width	Species
1	5.1	3.5	1.4	0.2	<i>I. setosa</i>
2	4.9	3.0	1.4	0.2	<i>I. setosa</i>
3	4.7	3.2	1.3	0.2	<i>I. setosa</i>
4	4.6	3.1	1.5	0.2	<i>I. setosa</i>
5	5.0	3.6	1.4	0.3	<i>I. setosa</i>
6	5.4	3.9	1.7	0.4	<i>I. setosa</i>

Example 3





UPPSALA
UNIVERSITET

(SUPERVISED) CLASSIFICATION

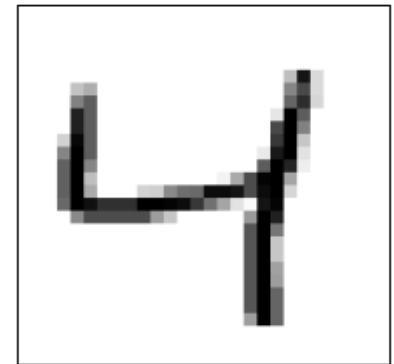
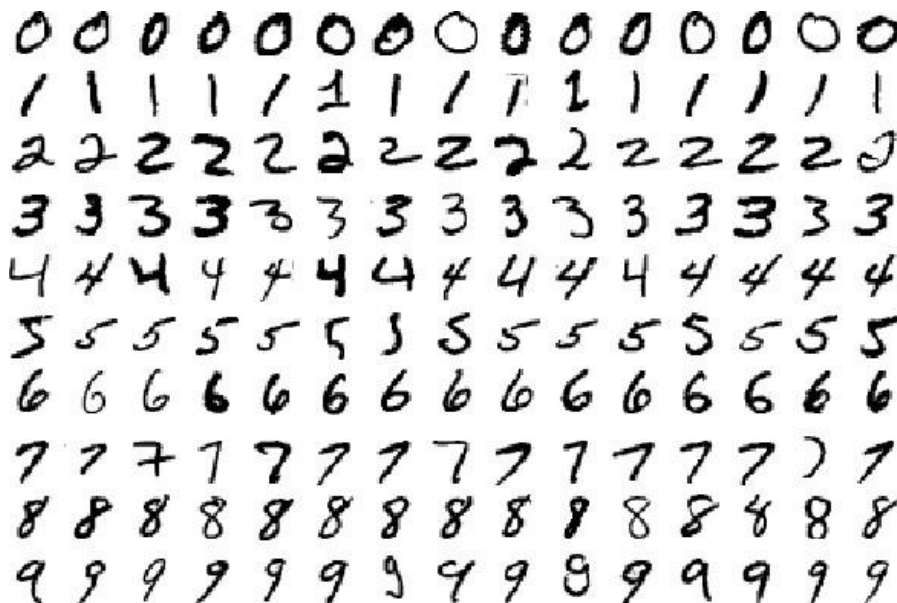


UPPSALA
UNIVERSITET

Supervised classification

MNIST dataset of handwritten digits

Task: Recognize isolated handwritten digits



Size-normalized, centered
28x28 pixels

MNIST dataset: 10 000 images of isolated digits to be recognized
60 000 labeled images of isolated digits

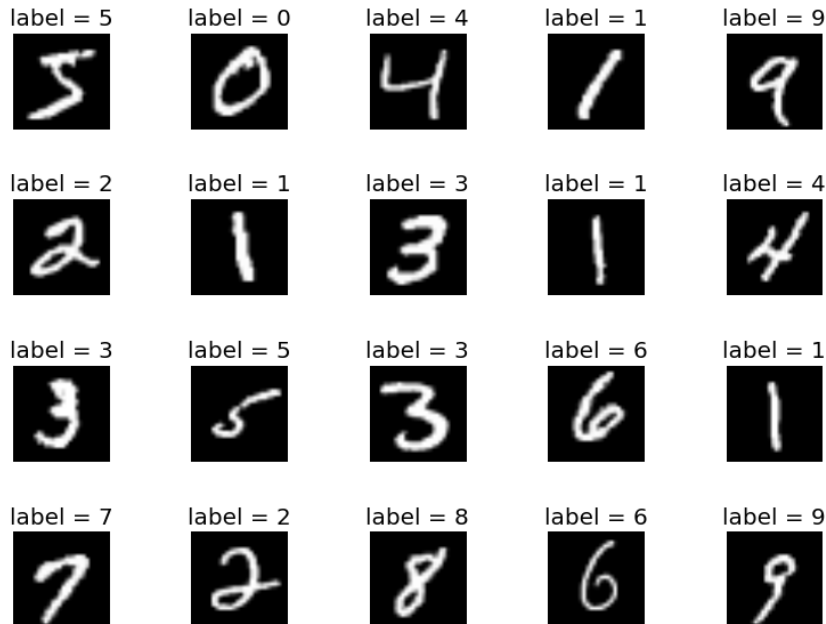
<http://yann.lecun.com/exdb/mnist/>



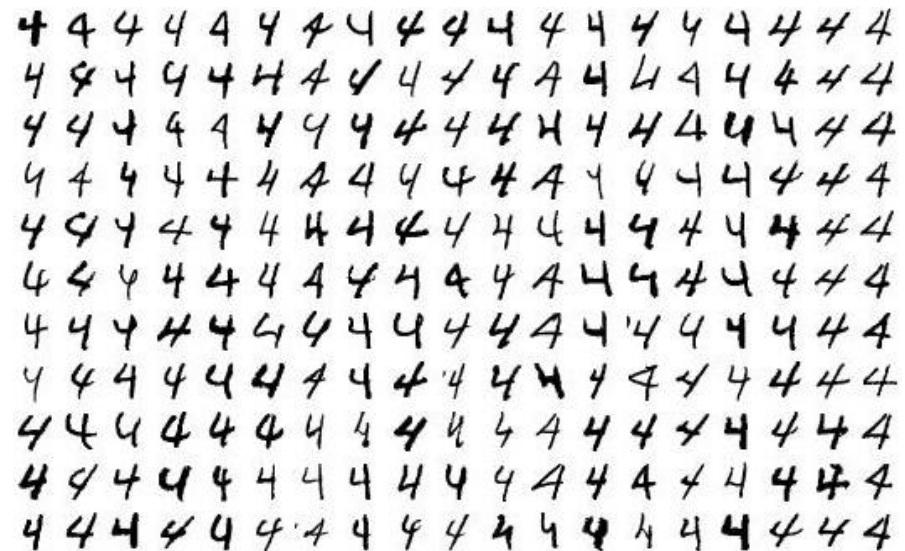
UPPSALA
UNIVERSITET

Supervised classification

MNIST dataset of handwritten digits



60 000 **labeled** images



Training data

Some examples of class 4 from the database
(**M**odified **N**ational **I**nstitute of **S**tandards)



UPPSALA
UNIVERSITET

Supervised classification

Training and testing

Training

Use **labeled** (correctly classified, known) examples to define a classifier (discriminant function) that separates patterns into different classes.

Classification

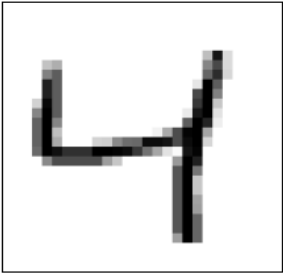
Take a new **unknown** example and assign it to the **predicted** class using the defined classifier (discriminant function).



UPPSALA
UNIVERSITET

Supervised classification

Nearest Neighbour (NN) classifier



Task: Given the input image – **pattern**,
assign it to one of the classes (categories) $0, 1, 2, \dots, 9$

Approach:

Compare the input pattern with all patterns in the labeled dataset.
Assign the label of the **most similar (closest)** pattern in the labeled set.

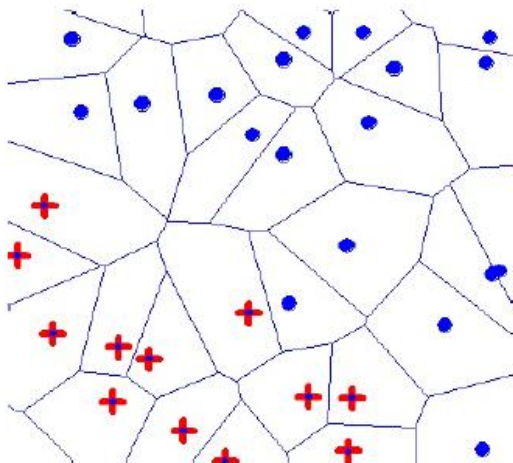
One of the most fundamental machine learning algorithms:
1-NN (the One-Nearest Neighbour) classifier

Supervised classification

k-Nearest Neighbour (k-NN) classifier

Pattern: Vector of 28x28 intensity values representing each MNIST image.
Can be some other suitable set of features (not easy to define which!)
Each pattern is a point in the feature-space.

Similarity? E.g., Euclidean distance between the points (vectors)
in the (high-dimensional) feature space.
Many others can be found (or designed to be) suitable!



1-NN classifier partitions the feature space
into Voronoi regions of the labeled samples.

To classify a new sample:
Check in which region it falls.
Assigned the corresponding label.



UPPSALA
UNIVERSITET

1-NN classifier

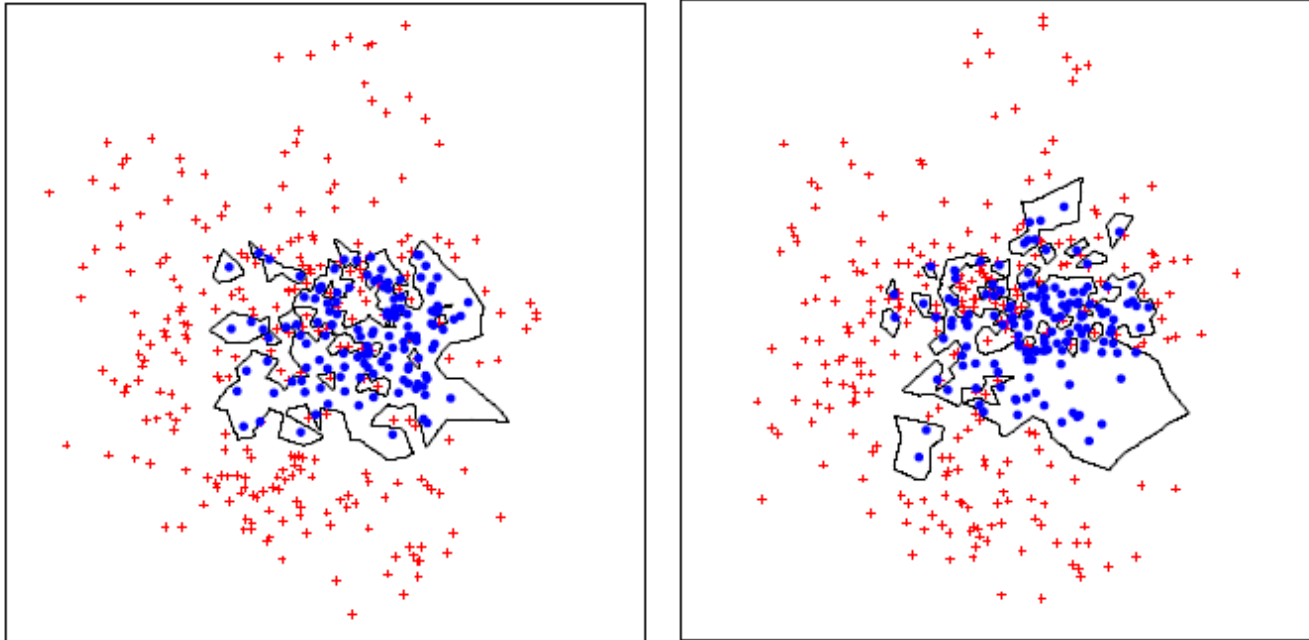


Image: Fred Hamprecht

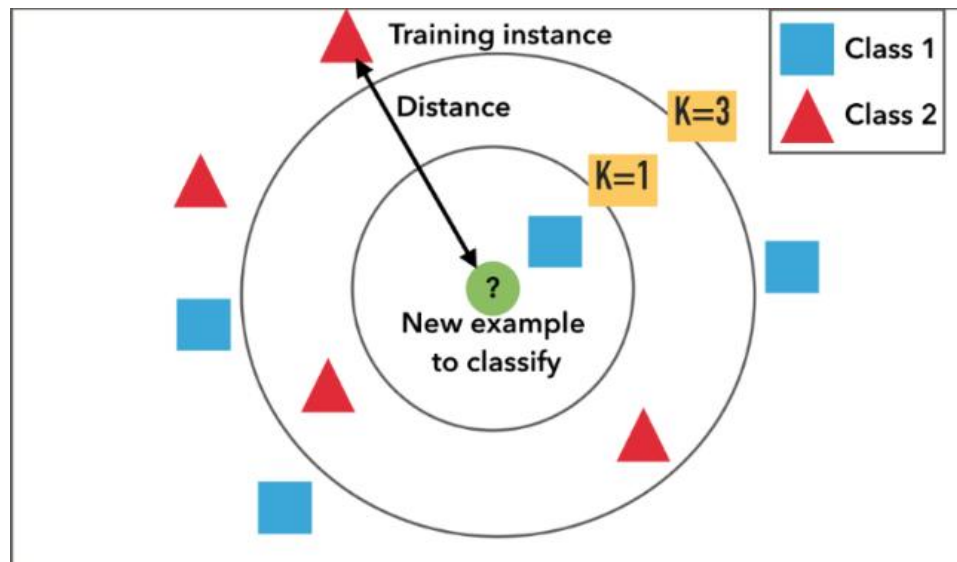
1-NN classifier derived from two disjunct halves of one training set.

Unstable behaviour in the regions where the classes overlap:
Decision boundary changes significantly.



k-NN classifier

Improve stability of 1-NN classifier by assigning the label of **the most dominant class** among the k nearest neighbours.



1-NN: Class 1

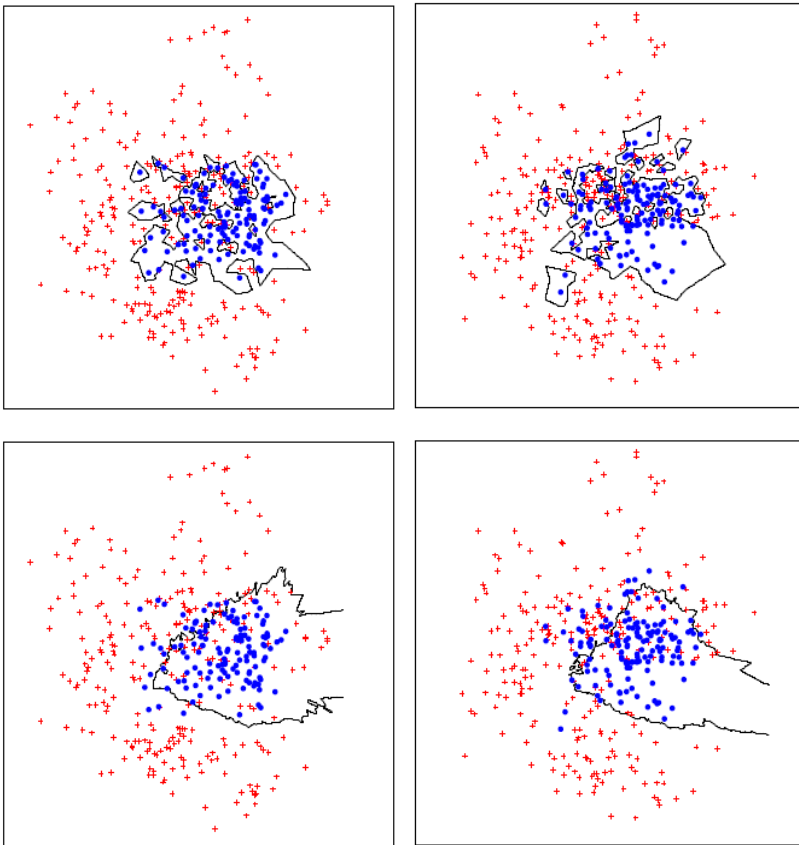
3-NN: Class 2



UPPSALA
UNIVERSITET

k-NN classifier

1-NN Classifier



41-NN Classifier

Classifiers derived from two halves of the same training dataset (left and right).

Bigger k – more stable and smoother decision boundary

How to select proper k ?

If too small – non-stability.

High variance.

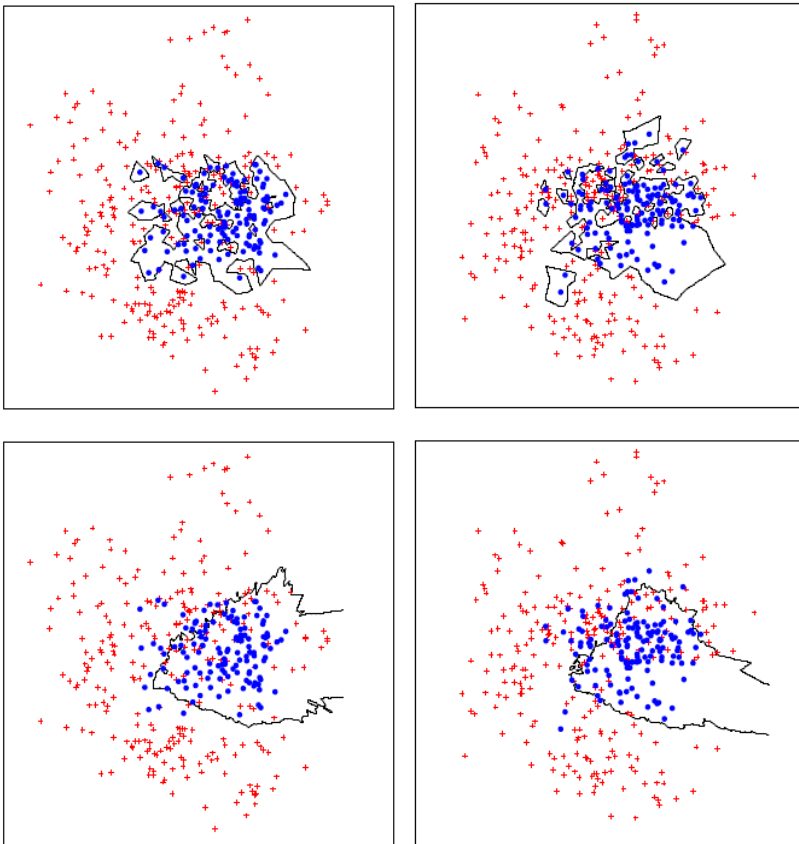
If too big?



UPPSALA
UNIVERSITET

k-NN classifier

1-NN Classifier



Classifiers derived from two halves of the same training dataset (left and right).

Bigger k – more stable and smoother decision boundary

How to select proper k ?

If too small – non-stability.

High variance.

If too big?

High bias.

41-NN Classifier



UPPSALA
UNIVERSITET

How good is it?

k-NN on MNIST

Performance measure of a classifier

classification **accuracy** (misclassification rate)

more sophisticated **loss function** (e.g., different treatment of errors)

3-NN on the raw image intensities (784 dimensional feature space),
with Euclidean distance, reaches 5% misclassification rate.

Consider: More advanced and better suited image representations
Alternative (better suited) distance measure.

We have reached 0.94% misclassification rate with an alternative distance measure

More advanced classifiers...

J. Lindblad, N. Sladoje. Linear time distances between fuzzy sets with applications to pattern matching and classification. *IEEE Transactions on Image Processing*, 23(1), pp. 126-136, 2014.



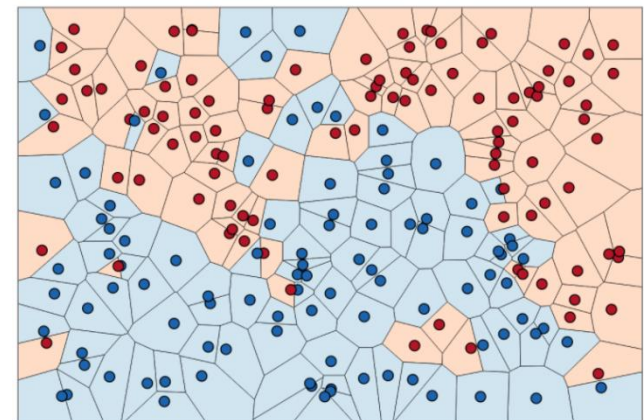
UPPSALA
UNIVERSITET

Computation

k-NN has **zero training time** (training data is only stored)
high testing time (all the computational efforts
in pairwise comparisons)

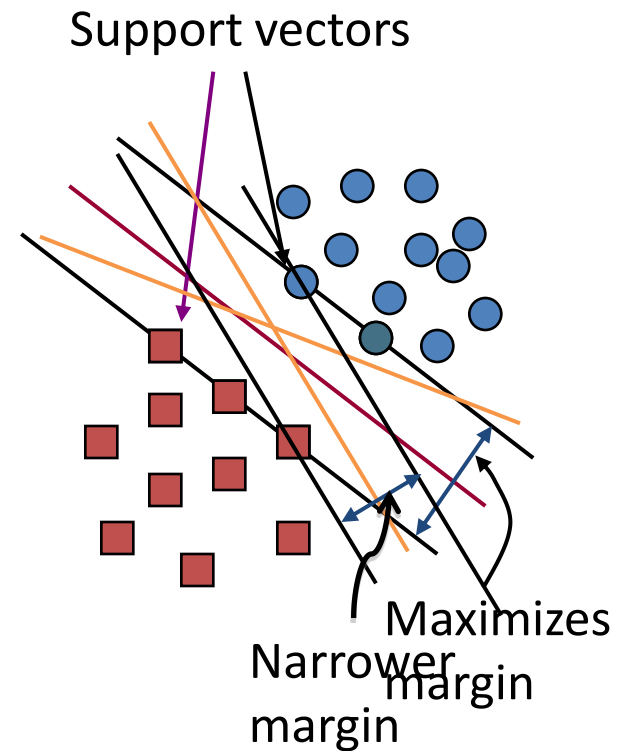
Only a fraction of the training data affects the actual
decision boundary of the k-NN classifier. (Which?)

Can we use this to improve the classifier?
(Increase its speed and robustness?)



Using only the important points Support Vector Machine (SVM)

- Separate the two classes by a line (hyperplane)
- Find “the best” line (hyperplane)
The best: **has the largest margin**
- Only **a subset of points** affect the selection of the separating line
- **Support vectors:** the points closest to the separating line





UPPSALA
UNIVERSITET

Linear SVM

Separating hyperplane

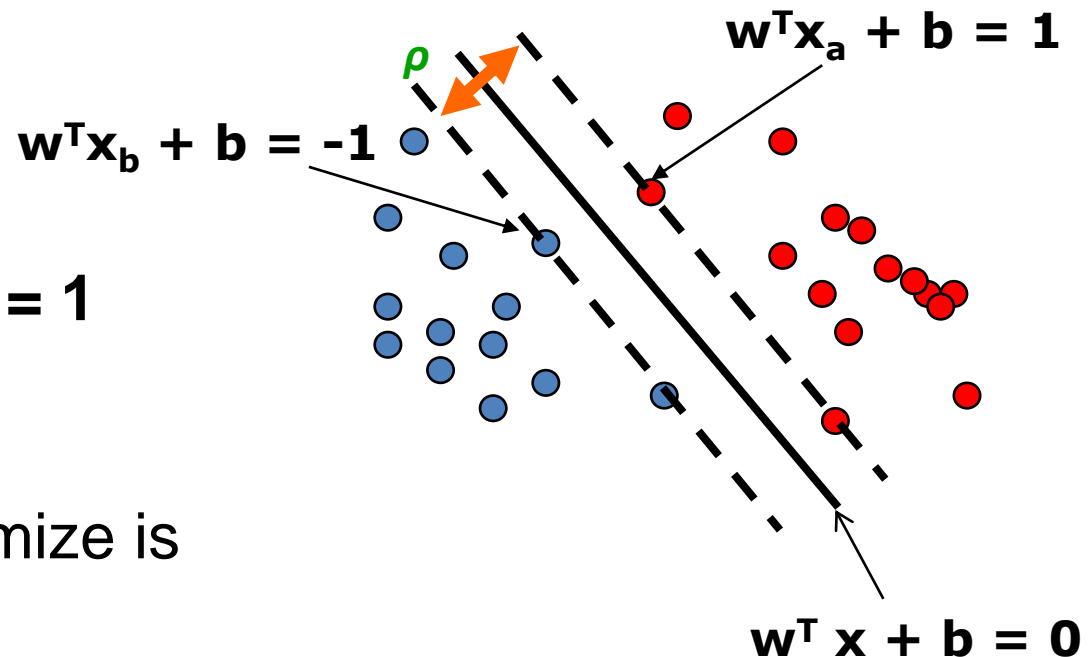
$$\mathbf{w}^T \mathbf{x} + b = 0$$

Extra scale constraint:

$$\min_{i=1,\dots,n} |\mathbf{w}^T \mathbf{x}_i + b| = 1$$

Then the margin to maximize is

$$\rho = 2/||\mathbf{w}||_2$$





UPPSALA
UNIVERSITET

SVM - summary

- SVM is solved by **quadratic programming**.
- The inner products between all (labeled) data points are required for the solution.
- **Support vectors** are identified during the process.
- Classification requires computing the **inner products** of the **test point** and all the **support vectors**.
- There are ways to deal with **non-linearly separable** classes
 - Allow for errors (non-separating planes) at some cost; minimize the cost.
 - Move to a higher dimensional space – data might become linearly separable (Kernel trick enables efficient computation)
- SVMs work well, but may be slow for big datasets.



Decision boundary and Discriminant functions

A **discriminant function** d_i for a class i , $i = 1, 2, \dots, N$ has a property that

$$d_i(x) > d_j(x), \quad j = 1, 2, \dots, N, \quad j \neq i$$

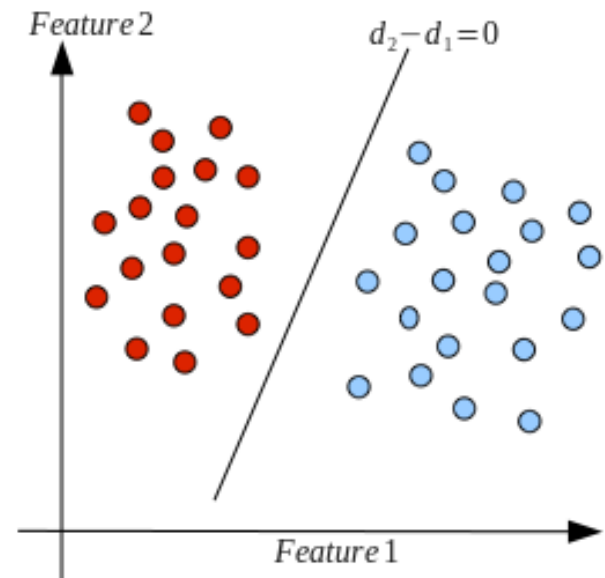
for every pattern x which belongs to the class, and any other class j .

The number of discriminant functions is equal to the number of classes.

If we know the discriminant functions of the observed classes, we can classify any pattern.

The **decision boundary** between class i and class j satisfies

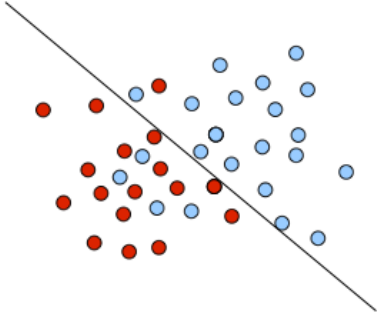
$$d_i(x) - d_j(x) = 0.$$



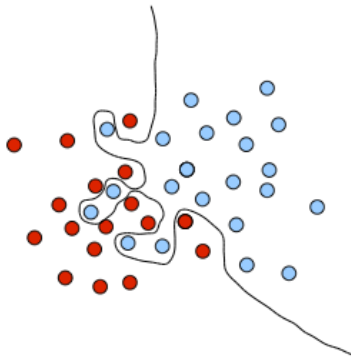


UPPSALA
UNIVERSITET

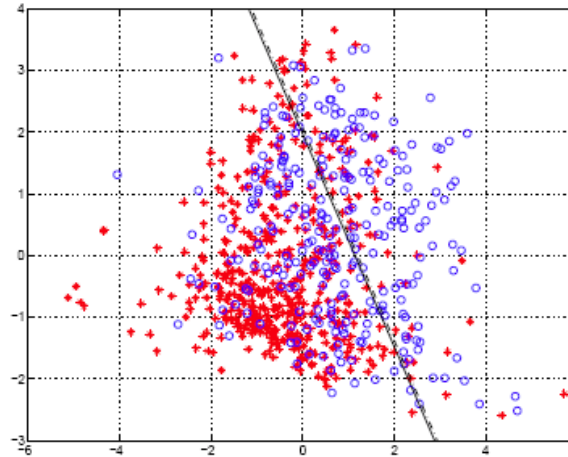
Decision boundary and Discriminant functions



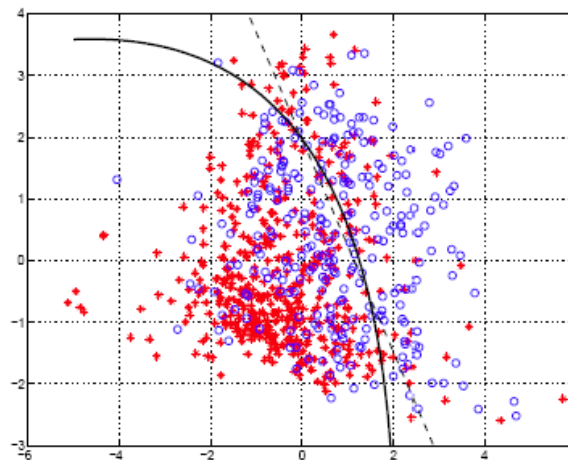
Linear decision boundary?
Non-separable data



Non-linear decision boundary
– overfitting?



Linear
Decision boundary



Quadratic
decision boundary



UPPSALA
UNIVERSITET

Optimal classifier

- Patterns are (assumed to be) randomly generated
 - Probabilistic approach to classification
- Classification error – misclassified pattern
- **Optimal classifier**
 - On average, lowest probability of committing classification error.



Optimal (Bayes) classifier

$p(w_i | \mathbf{x})$ - probability that a pattern \mathbf{x} belongs to a class w_i

$L(w_i, w_j)$ - loss (cost) if \mathbf{x} from w_i is classified as w_j

Conditional average loss (risk)

$$r_i(\mathbf{x}) = \sum_{j=1}^c L(w_j, w_i) p(w_j | \mathbf{x})$$

Bayes formula

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

$$r_i(\mathbf{x}) = \sum_{j=1}^c L(w_j, w_i) p(\mathbf{x} | w_j) P(w_j)$$

PDF of the patterns from class w_j

Prior probability



UPPSALA
UNIVERSITET

Optimal (Bayes) classifier

Assign \mathbf{x} to a class w_i if $r_i(\mathbf{x})$ is the lowest.

Assuming **binary classification**, and

loss 0 for correct classification,

loss 1 for incorrect classification:

The decision function of the Bayes classifier is

$$d_i(\mathbf{x}) = p(\mathbf{x} | w_i) P(w_i), \quad i = 1, 2, \dots, c$$

If the **distributions are known**, ensured minimal average loss.

Prior – known/inferred from the knowledge of the problem.

PDF of the class – estimated... or assumed to be Gaussian.



Bayes classifier for Gaussian distribution

Bayes classifier:

Probability distribution of the patterns in each class is required.
Each can be described by its **mean vector** and **covariance matrix**.

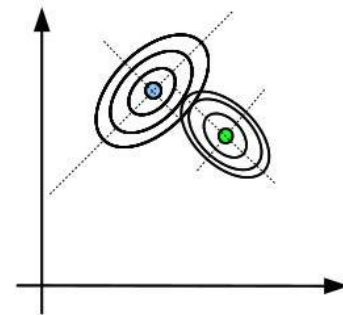
Assuming **Gaussian distributions** of the patterns in the classes:

$$d_i(\mathbf{x}) = \ln P(w_i) - \frac{1}{2} \ln |\mathbf{C}_i| - \frac{1}{2} [(\mathbf{x} - \mathbf{m}_i)^T \mathbf{C}_i^{-1} (\mathbf{x} - \mathbf{m}_i)]$$

$$\mathbf{m}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in w_i} \mathbf{x}$$

$$\mathbf{C}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in w_i} \mathbf{x} \mathbf{x}^T - \mathbf{m}_i \mathbf{m}_i^T$$

Quadratic function
(QDA – quadratic discriminant analysis)





UPPSALA
UNIVERSITET

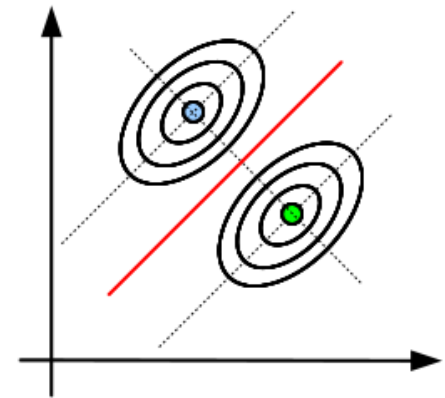
Linear discriminant analysis

Assume that all the covariance matrices are equal:

$$\mathbf{C}_j = \mathbf{C}, \quad j = 1, 2, \dots, c$$

decision function is linear – **Linear discriminant analysis (LDA)**

$$d_i(\mathbf{x}) = \ln P(w_i) + \mathbf{x}^T \mathbf{C}^{-1} \mathbf{m}_i - \frac{1}{2} \mathbf{m}_i^T \mathbf{C}^{-1} \mathbf{m}_i$$



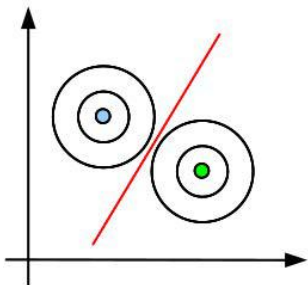


Minimal distance classifier

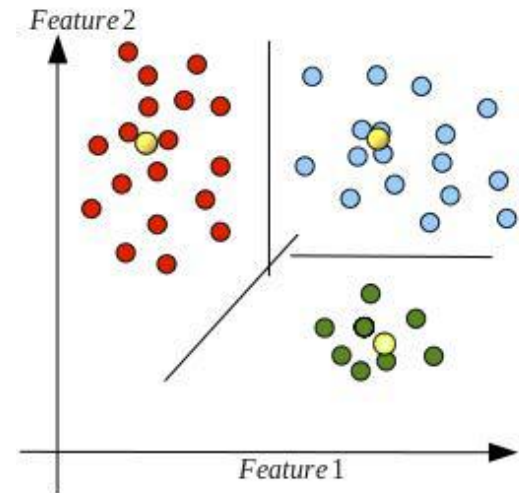
Assume

- all the covariance matrices are equal to the identity matrix
- all the classes are equally likely

Decision function becomes



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



New objects are classified to the class with the closest mean vector.



UPPSALA
UNIVERSITET

DATA



UPPSALA
UNIVERSITET

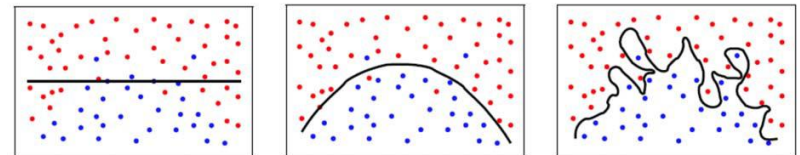
Aim – generalization

Starting from the labeled patterns, we aim to correctly classify **new patterns** that are **not known** (“not seen by the classifier”) during the training.

We want a classifier which **generalizes** well.

Tuning the classifier (selecting its parameter values) to perform very well on the training data may lead to poor generalization (overfitting).

Example: k-NN with low value of k.



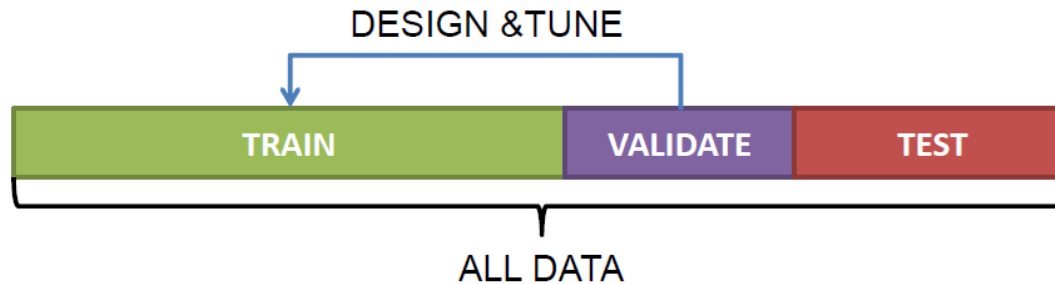
By A Zisserman

We learn the training data “specificities”, not necessarily the distribution.

It is desirable to learn the regularities, but not the noise!

Training, validation, testing: Classifier and its parameters

Divide the set of all available labeled samples (patterns) into:
training, validation, and test sets.



Training set: Represents data faithfully and reflects all the variation.
Contains large number of training samples.
Used to define the classifier.

Validation set: Used to tune the parameters of the classifier.
(Bias –Variance trade-off to prevent over-fitting)

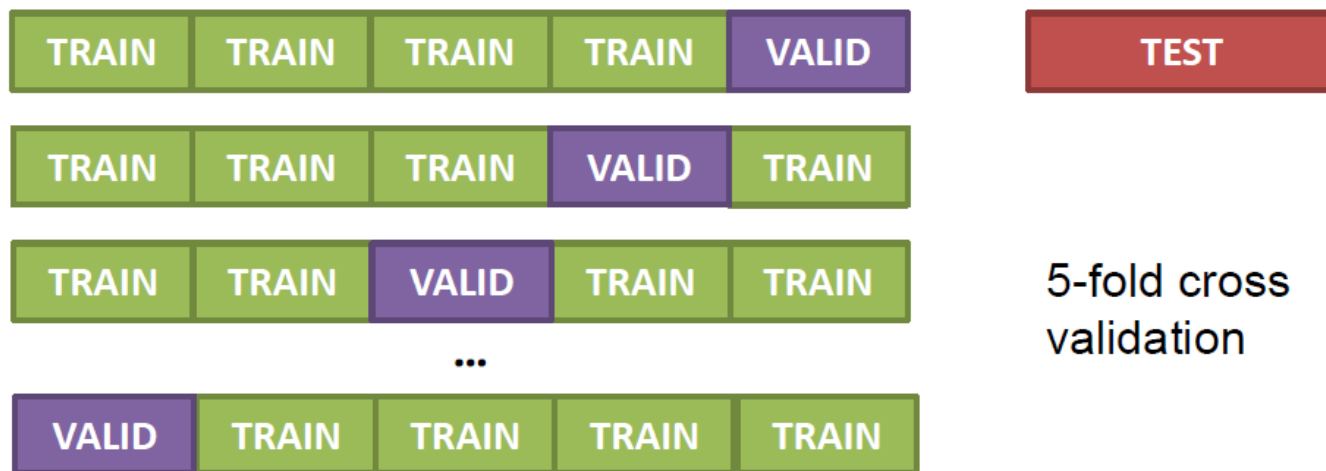
Test set: Used for final evaluation (estimation) of the classifier's performance on the samples not used during the training.



UPPSALA
UNIVERSITET

Cross-validation

Recycling available data – we usually need more than what we have!



Compute average (and/or some other statistics) of the performances over the folds on the validation sets,
or keep an independent test set for final performance estimation.

Training, validation, testing

A question

Two students are designing a classifier for binary image classification. Each student has their own set of 100 labeled images, 90% of which are used for training and 10% for validating the model.

Student 1 runs a k-NN classification algorithm and reports 80% accuracy on her validation set.

Student B experiments with over 100 different learning algorithms, training each one on his training set, and recording the accuracy on the validation set. His best formulation achieves 90% accuracy.

Whose algorithm would you pick to solve the given task?
Why?



UPPSALA
UNIVERSITET

LAB 4



UPPSALA
UNIVERSITET

Lab 4 on classification

- Pixel-wise classification based on
 - grey levels
 - colour
- Pixel-wise linear discriminant analysis
 - Selection of training data, training and classification
 - MATLAB function “**classify**”
 - Different feature spaces and (basic) feature selection
- Texture-based object classification
 - Texture features
 - Outlier detection (by cluster analysis)



UPPSALA
UNIVERSITET

Unsupervised classification

CLUSTERING



UPPSALA
UNIVERSITET

Unsupervised classification

Cluster analysis

find groups of patterns that **“somehow belong together”**.

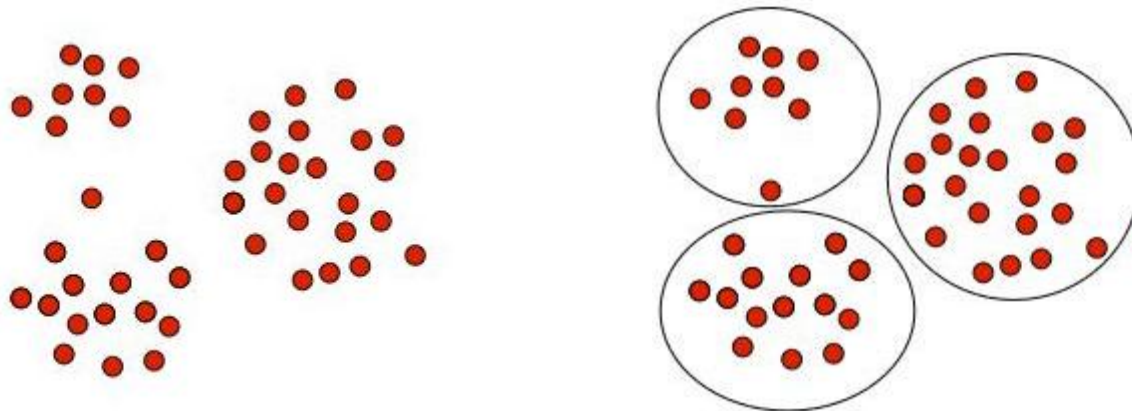
Humans tend to see patterns and clusters everywhere
– even where there are no such!

Much harder to do automatically.

Unsupervised classification is far less developed than supervised.



Man in the moon. NASA





UPPSALA
UNIVERSITET

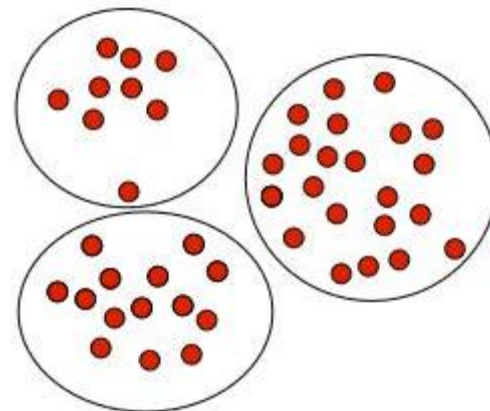
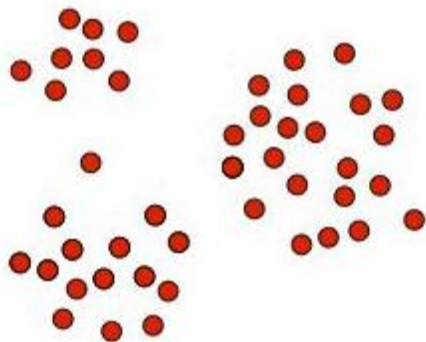
Unsupervised classification

Vagueness of the task prevents formalism towards solution.

But: Difficult, expensive or even impossible to reliably label training samples with its true category.

Assumption:

Patterns within a cluster are **more similar** to each other than patterns belonging to different clusters.

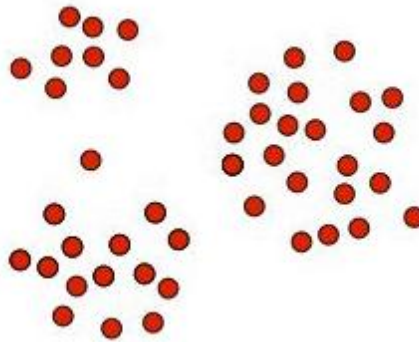




UPPSALA
UNIVERSITET

Unsupervised classification

How to determine the **number of clusters**?

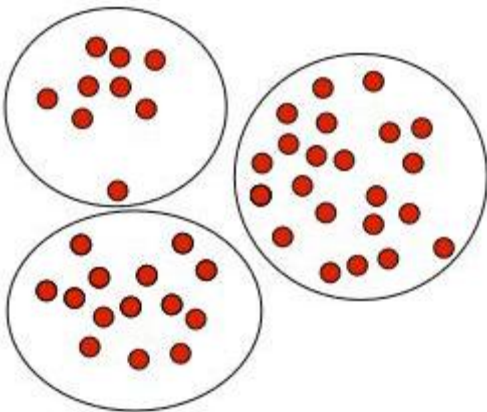
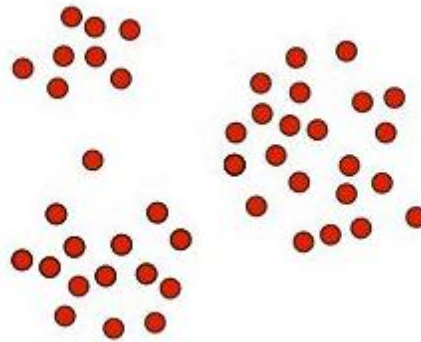




UPPSALA
UNIVERSITET

Unsupervised classification

The key question - How to determine the **number of clusters**?

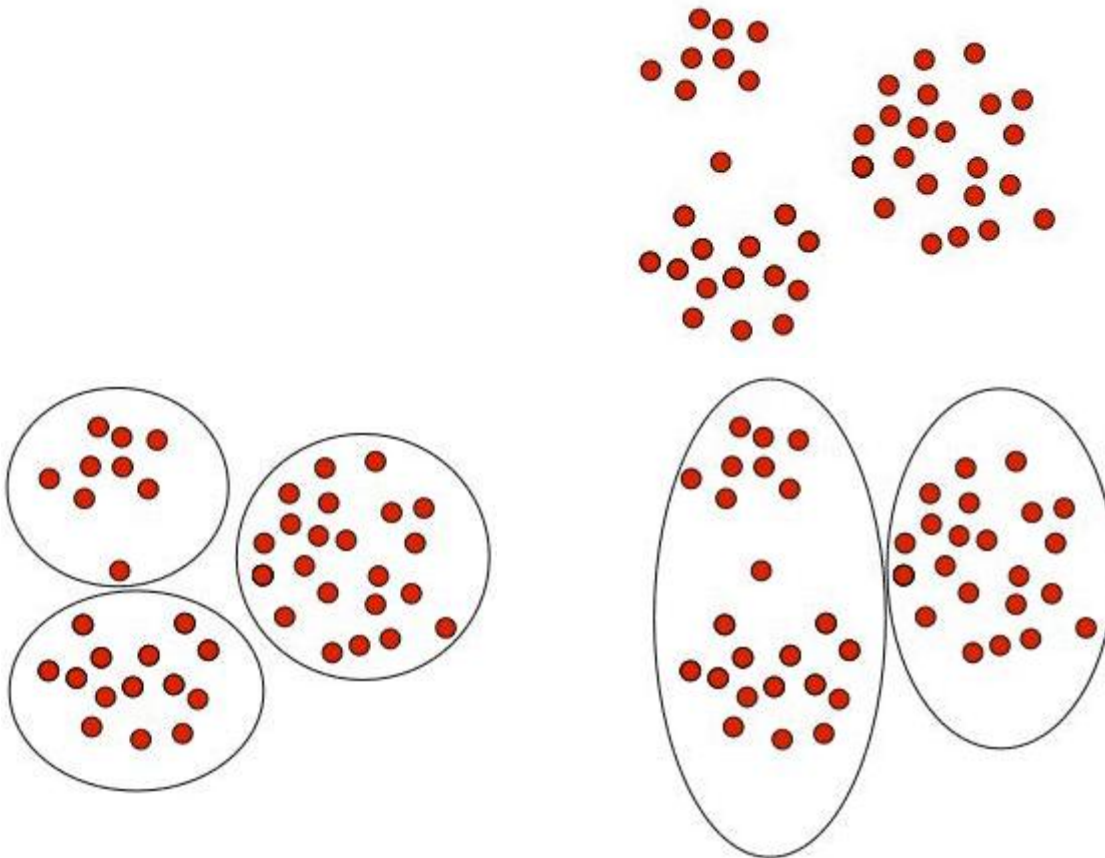




UPPSALA
UNIVERSITET

Unsupervised classification

The key question - How to determine the **number of clusters**?

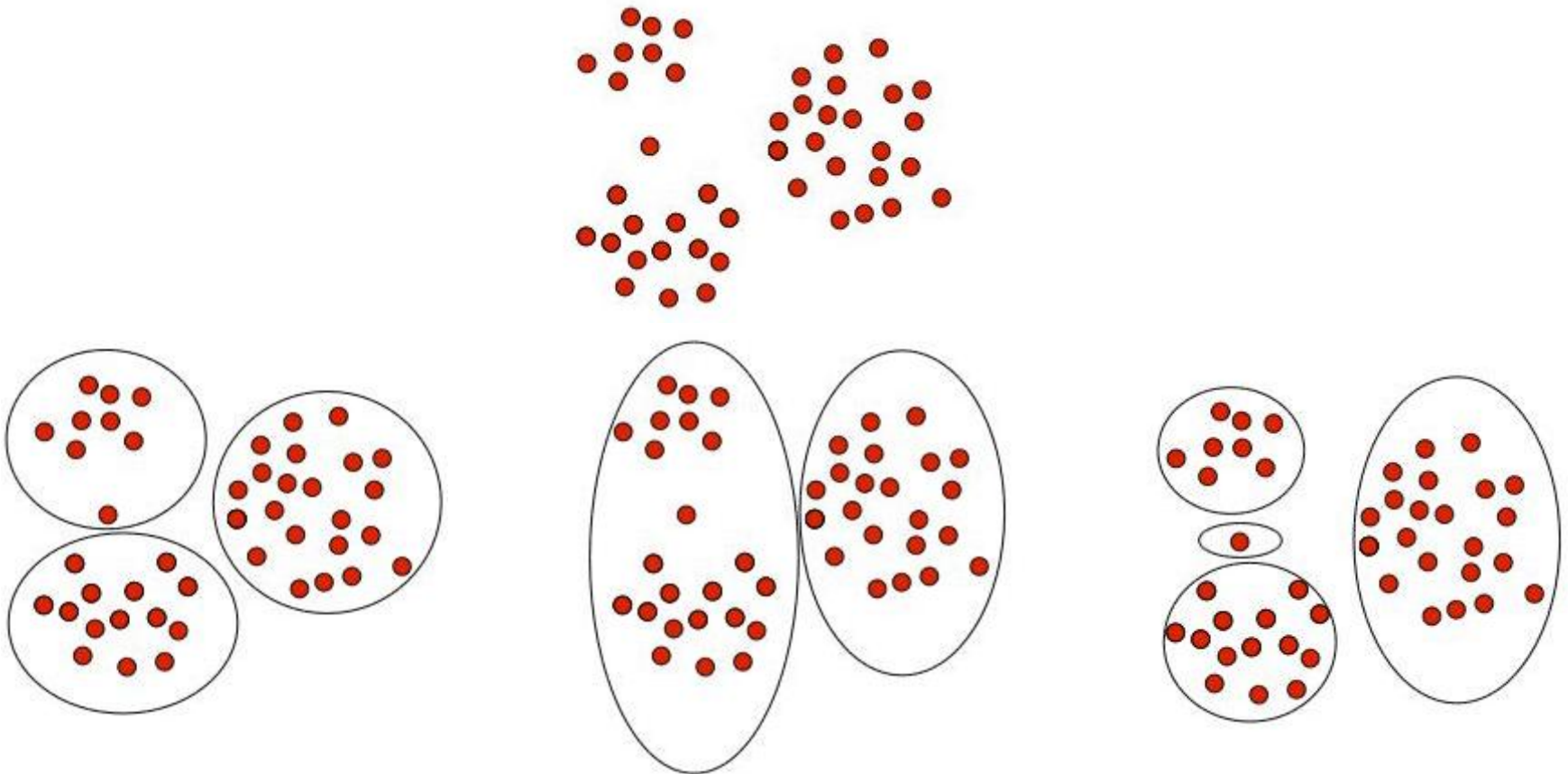




UPPSALA
UNIVERSITET

Unsupervised classification

The key question - How to determine the **number of clusters**?



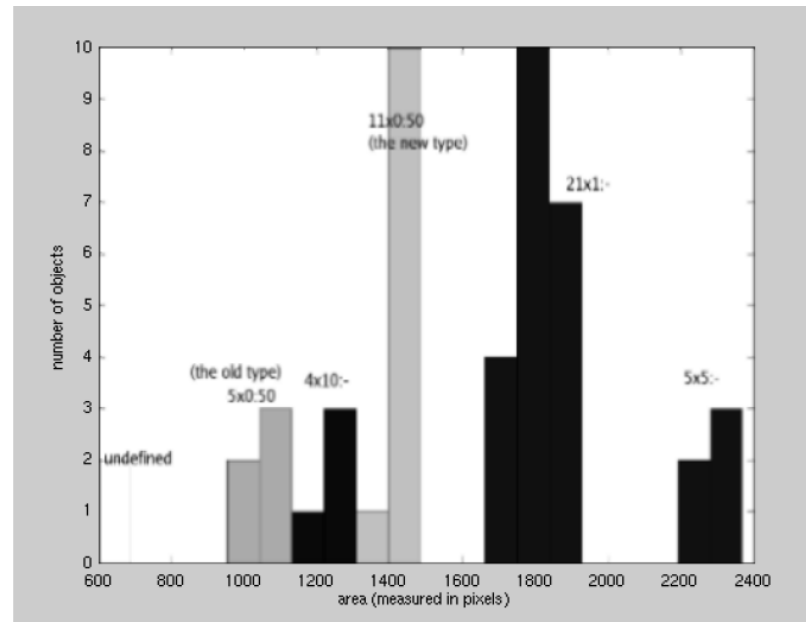
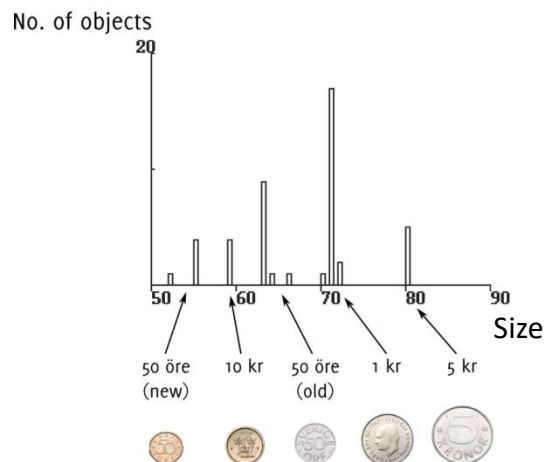
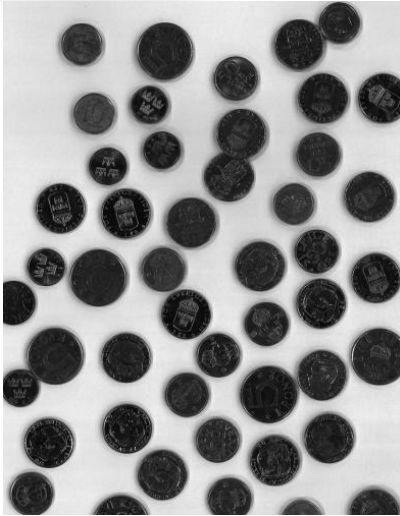
Depends on scale.



UPPSALA
UNIVERSITET

Histogram-based clustering

Example 1



How many clusters?

How many elements in each cluster?



UPPSALA
UNIVERSITET

K-means clustering

K-means clustering partitions n patterns into K clusters such that each pattern belongs to the cluster with the **nearest mean** - a prototype of the cluster.

The number K of clusters has to be defined in advance.

Performance strongly depends on the quality of this estimate.

There are procedures to estimate K from the data.

Works well for separable spherical clusters of similar sizes.

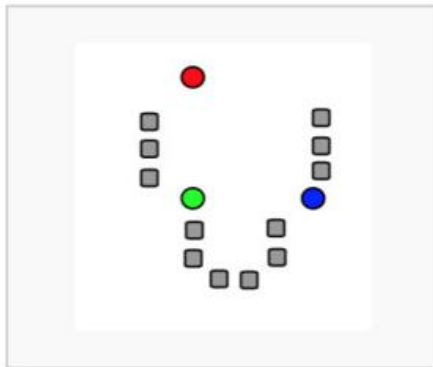
Different distance measures may be used to change the shape.



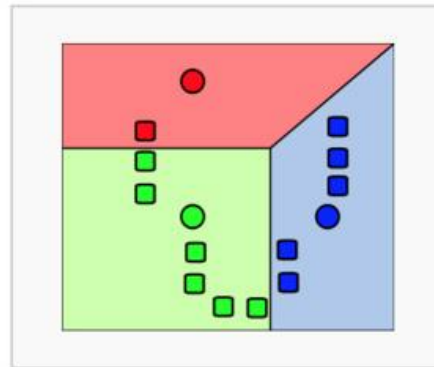
UPPSALA
UNIVERSITET

K-means clustering

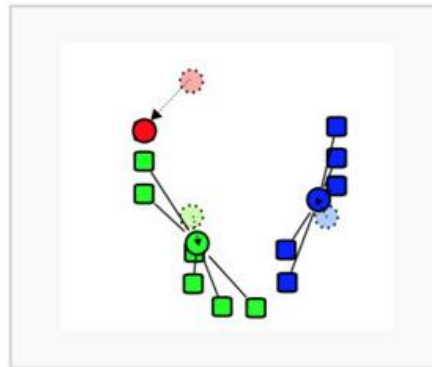
Demonstration of the standard algorithm



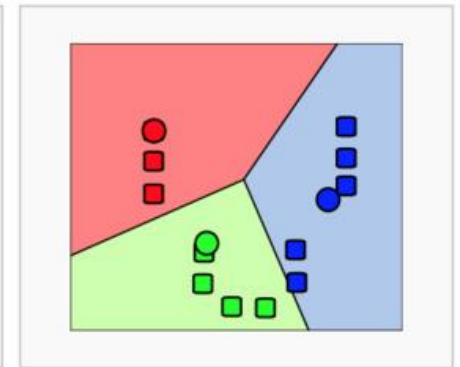
1. k initial "means" (in this case $k=3$) are randomly generated within the data domain (shown in color).



2. k clusters are created by associating every observation with the nearest mean. The partitions here represent the **Voronoi diagram** generated by the means.



3. The **centroid** of each of the k clusters becomes the new mean.



4. Steps 2 and 3 are repeated until convergence has been reached.

<https://brilliant.org/wiki/k-means-clustering/>



K-means clustering

K-means algorithm **minimizes the sum of within-cluster variances** for the K observed clusters:

$$E_K = \sum_{i=1}^K \sum_{j=1}^n I_{j,i} (d_{ji})^2$$

Element in a partition matrix:
"if pattern j belongs to cluster i "

(Euclidean) distance between
pattern j and the center of cluster i

Initialization is usually random (which affects the result):

- Random patterns are selected as cluster centres, or
- Random clusters are formed of patterns.

Convergence to a local minimum.



K-means clustering

Example: Clustering of $n=4$ points into $K=3$ clusters.

Partition matrix indicates belongingness of each point to each cluster.

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad n \times K$$

← ← ← Points

↑ ↑ ↑ Clusters

Point 4 (row) does not belong to cluster 2 (column).

Distance matrix contains the distances between each point and each cluster center.

$$D = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \\ d_{41} & d_{42} & d_{43} \end{bmatrix}$$

The distance between point 4 and center of cluster 2.



UPPSALA
UNIVERSITET

WRAPPING UP



UPPSALA
UNIVERSITET

Summary

Classification is a high-level image processing task

- data understanding and interpretation

Patterns are grouped into classes based on their similarity to the class “specification”.

Classification:

- supervised (true classes available for a set of patterns – training set)

- unsupervised (no labeling available – clustering)

Classifiers:

- k-NN – non-linear, simple, fast to train, slow to test

- SVM – linear (can be improved), good!, can be slow for big datasets

- Optimal (Bayes) – theoretical (requires known data distributions)

- LDA & min dist. classifier – Bayes with assumed Gaussian distributions

K-means clustering

- what is the correct number of clusters?

- local minimum can be an un-natural solution



UPPSALA
UNIVERSITET

Next time

Here discussed classification methods require
feature extraction (possibly with segmentation)
feature selection
classifier selection
suitable for the task and usually highly data-dependent.



F.-F.Li, A Karpathy, J. Johnson

Can we make it ... easier?

Deep learning and CNNs