Exploratory Analysis
Feature Selection and Clustering

Davide Bacciu

Computational Intelligence & Machine Learning Group
Dipartimento di Informatica
Università di Pisa
bacciu@di.unipi.it

Introduzione all'Intelligenza Artificiale - A.A. 2012/2013

Lecture Outline

Exploratory Data Analysis

- Discover structure in data
  - Find unknown patterns in the data that cannot be predicted using current expert knowledge
  - Formulate new hypotheses about the causes of the observed phenomena
- Finding informative attributes
  - Feature Extraction
  - Feature Selection
- Finding natural groups
  - Clustering

Feature Selection Vs Feature Extraction

- Two approaches to dimensionality reduction
  - Feature Extraction - Create a new, lower dimensional, representation of some input data by transforming the existing features with a given function
  - Feature Selection - Select a subset of the existing features without transforming the input data
- Feature extraction generates new features by optimizing
  - Signal representation (PCA)
  - Signal classification (LDA)
- Feature selection looks at dimensionality reduction from a different perspective
  - Different methodologies (e.g. computational costs, . . .)
  - Different applications

Why Feature Selection?

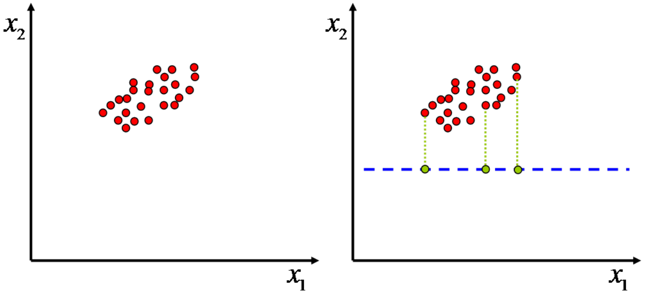
- Straightforward answer is: for several good reasons discussed previously for feature extraction
  - Reduce problem complexity
  - Reduce noise
  - Find *good* data representations
- So, why do not always use feature extraction?
  - It can be computationally expensive to generate new features
  - Don't want to transform data canceling its semantics
  - Data is not always numeric
  - Data may fail to meet the theoretical assumptions underlying feature extraction
  - It allows to explicitly select good predictors, i.e. features that *behave well* on a specific supervised task

What About Projection?

Feature selection is rarely used for visualization

Nevertheless, it actually implements a projection

It is equivalent to projecting data onto lower-dimensional linear subspace perpendicular to the removed feature



Some Use Cases

Case I - Lung cancer

- Features are biomedical information or aspects of a patient's medical history
- Which features best predict whether lung cancer will develop?

Case II -Image Understanding

- Samples are images and features are pixels
- Which regions of an image is more likely to provide useful/discriminative information?

We seek a compact data representation that is interpretable and that tells us which features are relevant

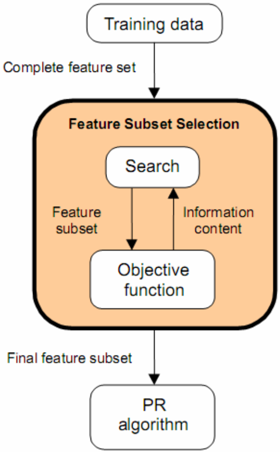
Feature Selection

- Definition - A process that chooses a D'-dimensional subset of all the features according to an objective function

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ \\ x_D \end{bmatrix} \xrightarrow{\text{select } i_1,\dots,i_{D'}} \mathbf{y} = \begin{bmatrix} x_{i_1} \\ x_{i_2} \\ \dots \\ x_{i_{D'}} \end{bmatrix}$$

- Three characterizing aspects
  - Subset $\Rightarrow$ no creation of new features
  - Objective function $\Rightarrow$ measure of subset optimality
  - Process $\Rightarrow$ need a subset search strategy

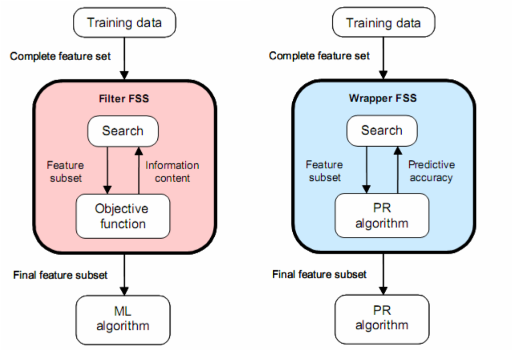
Search Strategy and Objective Function



- Feature selection requires
  - A search strategy to select candidate subsets
  - An objective function to evaluate these candidates
- Objective function
  - Evaluates candidate subsets and returns a measure of their optimality that is used to select new candidates
- Search Strategy
  - Exhaustive evaluation of all feature subsets from N samples is $O(2^D)$
  - Need a smart strategy to direct the subset selection process

Objective Functions - Two Approaches

- Filters - Evaluate feature subsets by their information content, e.g. interclass distance, statistical dependence or information-theoretic measures
- Wrappers - Use a pattern classifier which evaluates feature subsets by their predictive accuracy (e.g. recognition rate on validation data) using re-sampling or cross-validation



Filter Approach

Basic Idea

Assign an heuristic score to each feature to filter out the useless ones

- Separate feature selection phase from pattern recognition/classifier learning
- Evaluate features by measuring their informative content
  - Does the individual feature seems to help prediction?
  - Is it redundant?
  - Is it reliable?
- The scoring metric can be unsupervised or exploit supervised information

Measuring Feature Relevance

- Distance metrics
  - Measure separability with respect to some target attribute (e.g. a class)
  - Select those features which have the largest separability
- Correlation metrics
  - Good features are highly correlated with the class but are uncorrelated with each other
- Information-theoretic measures
  - Measure the reduction in uncertainty (entropy) between a target variable (e.g. the class) and the feature
  - E.g. Mutual information
- Consistency measures
  - Find a minimum number of features that separate classes as consistently as the full set
  - An inconsistency are two samples from different classes having the same feature values

Wrapper Approach

Basic Idea

Select those features that make my supervised learning model perform best

- Feature selection relies on the preselected supervised learning model
- The learning model is re-trained each time a new subset is selected
  - Quality of the feature subset is measured based on the empirical error of the learned model
  - Need to use robust validation strategies to ensure generalization
- Selected features are, tipically, effective class predictors for the model

Filters vs. Wrappers (I)

Filter Approaches

- Pro
  - Fast execution - Non-iterative dataset processing which can execute much faster than a classifier training
  - Generality - Evaluate intrinsic properties of the data, rather than interactions with a particular classifier, hence solutions will be *good* for a larger family of classifiers
- Cons
  - Select large subsets - Since the filter objective functions are generally monotonic, the filter tends to select the full feature set as the optimal solution. This forces the user to select an arbitrary cutoff on the number of features to be selected

Filters vs. Wrappers (II)

Wrapper Approaches

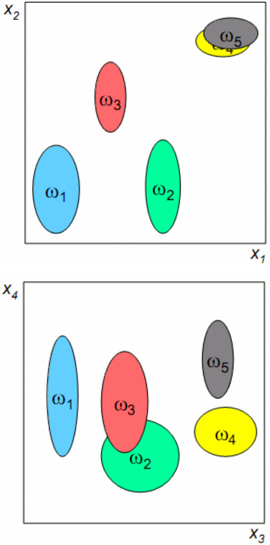
- Pro
  - Accuracy - Achieve better recognition rates since selected feature are tuned to the classifier
  - Test generalization - Use cross-validation measures of predictive accuracy to avoid overfitting
- Cons
  - Slow - Must train a classifier for each feature subset
  - Lack of generality - Solutions are tied to the bias of the classifier used in the evaluation function

Characterization of a Search Routine

- Search starting point
  - Empty set
  - Full set
  - Random subset
- Search directions
  - Sequential selection/elimination
  - Random generation
    - Perform randomized exploration of the search space where next direction is sample from a given probability
    - E.g. genetic algorithms, simulated annealing, . . .
- Search Strategy
  - Exhaustive/complete search
  - Heuristic search
  - Nondeterministic search

Sequential Feature Ranking

- The simplest strategy
  - Weight and rank each feature
  - Select top-K ranked features
  - No need of subset search
- Advantages
  - $O(D)$ complexity
  - Easy to implement
- Disadvantages
  - How to determine top-K the threshold?
  - Does not consider feature correlation

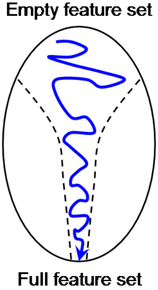


Sequential Search (I)

Greedy search algorithms adding or removing single features sequentially

Sequential Forward Selection

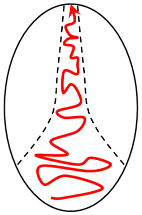
- Starting from the empty set, sequentially add the feature x_d that results in the highest objective function $J(X_{D'} \cup \{x_d\})$ when combined with the features $X_{D'}$ that have already been selected
- Performs best when the optimal subset has a small number of features



Sequential Search (II)

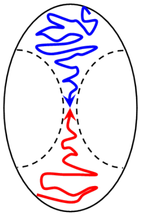
Sequential Backward Elimination

- Starting from the full set, sequentially remove the feature $x_d \in X_{D'}$ that results in the smallest decrease of $J(X_{D'} \setminus \{x_d\})$
- Performs best when the optimal subset has a large number of features



Bi-directional Search

- Forward and backward searches are performed in parallel
- They are forced to converge to the same solution by ensuring that
  - Features selected in forward pass are not removed by backward search
  - Features removed in backward search are not selected in the forward pass



Search Strategies Cookbook

	Accuracy	Complexity	Advantages	Disadvantages
Exhaustive	Always finds the optimal solution	Exponential	High accuracy	High complexity
Sequential	Good if no backtracking needed	Quadratic $O(N_{EX}^2)$	Simple and fast	Cannot backtrack
Randomized	Good with proper control parameters	Generally low	Designed to escape local minima	Difficult to choose good parameters

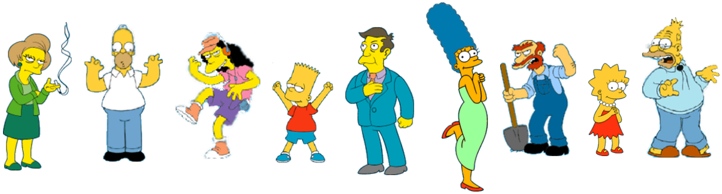
Unsupervised Feature Selection

- So far feature selection approaches seem to exploit only relevance measures relying on supervised class information
- Is there any unsupervised feature selection approach? The answer is... yes (surprised?)
- Feature redundancy and relevance is measured with respect to clusters instead of classes

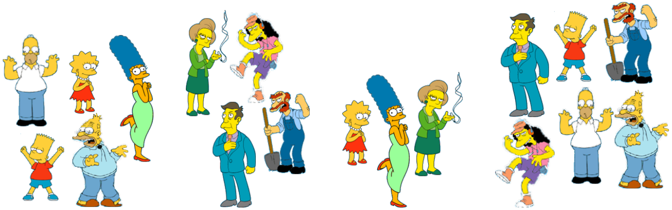
Definition (Clustering)

The process of organizing objects into natural groups whose members are similar in way determined by a given metrics

Clustering = Seeking a natural grouping



What is a natural grouping?



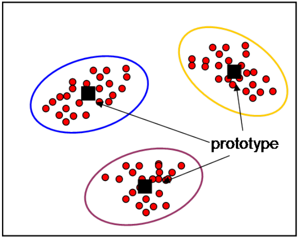
Family School Females Males

Credit goes to Eamonn Keogh @ UCR

The Clustering Problem (I)

Clustering objective

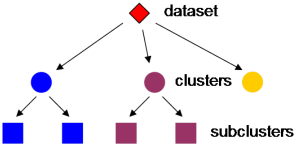
- Maximizing intra-cluster similarity
- Minimizing inter-cluster similarity

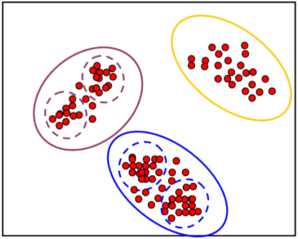


- Prototype-based clustering
  - Find a representative $\mathbf{c}_i$ (prototype) for each cluster i
  - Minimizing the distance w.r.t the cluster members
- Results depend on the choice of the distance metric
  - Euclidean $\|\mathbf{c}_i - \mathbf{x}_n\|_2$
  - Mahalanobis $(\mathbf{c}_i - \mathbf{x}_n)^T S^{-1}(\mathbf{c}_i - \mathbf{x}_n)$
  - ...

The Clustering Problem (II)

It is not always easy to define what is a cluster and what is not

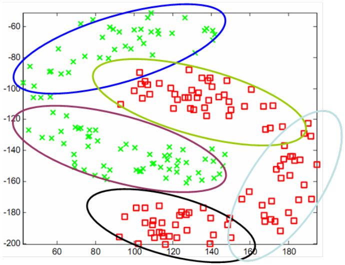


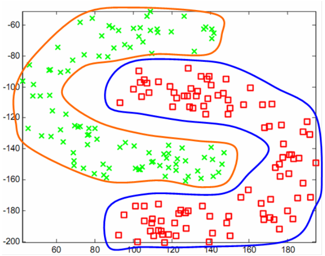


- Hierarchical Clustering
  - Prototype-based
  - Finds a hierarchy of nested clusters
- Useful to convey a multi-resolution view of data
  - Bio-medical data
  - Clusters ⇒ tumors
  - Sub-clusters ⇒ tumor subtypes

The Clustering Problem (II)

As we have seen with the Swiss roll in feature extraction, some data might lay on particularly nasty surfaces





Prototype-based clustering may not be effective with non-linearly separable clusters

K-means Clustering

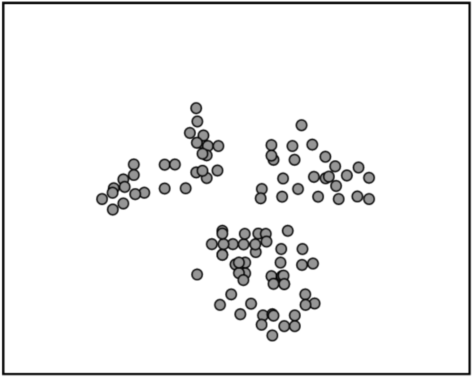
- The simplest clustering algorithm
  - Lots of limitations
  - Still widely used (easy to understand and implement)
- The algorithm in brief
  - Receives as input k, i.e. the number of clusters to seek in the data
  - Starts by picking k points at random as cluster prototypes (centroids)
  - Assigns each sample to the nearest prototype and recomputes the cluster centroids

The Algorithm

Algorithm k-means($\mathcal{X}$,K)

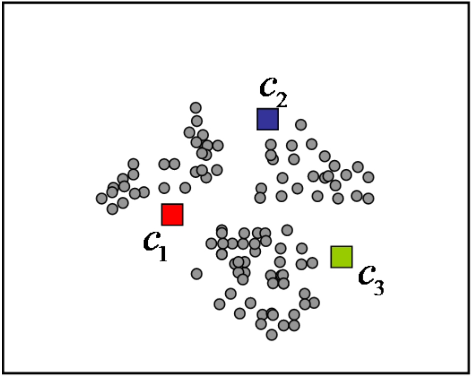
$N \leftarrow |\mathcal{X}|$;
for all $i = 1$ to K **do**
  $c[i] \leftarrow \mathrm{rand}()$;
end for
repeat {Update prototypes}
  for all $n = 1$ to N **do**
    $\mathrm{winner} \leftarrow \arg\min_{i=1,\ldots,K} \|c[i] - x_n\|_2$
    $tot[\mathrm{winner}] \leftarrow tot[\mathrm{winner}] + 1$
    $cnew[\mathrm{winner}] \leftarrow cnew[\mathrm{winner}] + x_n$
  end for
  for all $i = 1$ to K **do**
    $c[i] \leftarrow \frac{cnew[i]}{tot[i]}$;
  end for
until MaxIterations **or** $\|c - cnew\| < \epsilon$
return Cluster prototypes c

k-Means Example (I)



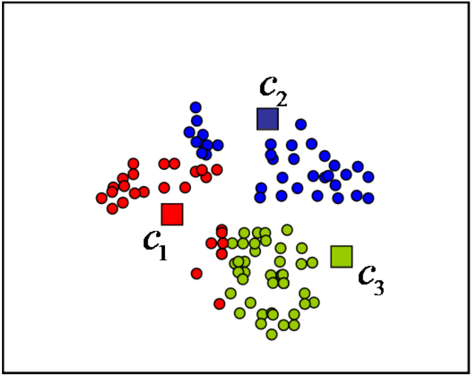
Input data $\mathcal{X}$

k-Means Example (II)



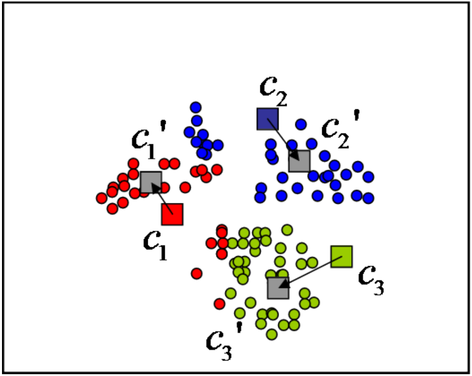
Initialize $K = 3$ prototypes

k-Means Example (III)



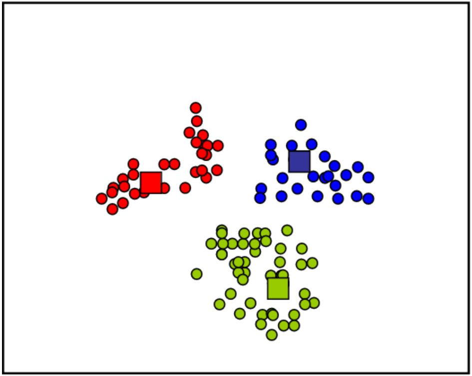
Assign samples to the nearest prototype/cluster

k-Means Example (IV)



Compute the updated prototypes c_i'

k-Means Example (V)



Update cluster assignment

Cluster number estimate

- In practice, we seldom know the cluster number K
  - Trial-and-error to find the most suitable K
  - Use algorithm that estimate the cluster number
- Several approaches but no killer algorithm
  - Agglomerative Clustering - Start with all samples being a separate cluster and iteratively aggregate existing cluster until an error criterion is satisfied
  - Partitional Clustering - Start with a single cluster and keep splitting it until an error criterion is satisfied
- Error is often composed of
  - A similarity term favoring aggregation of samples into clusters
  - A penalization term discouraging the creation of too many clusters

Clustering and Feature Selection

Cluster labels can be used for performing feature selection when class information is not available

- Filter approach
  - Step 1 Partition the data using a clustering algorithm
  - Step 2 Run a filter model evaluating feature relevance w.r.t. how much an attribute helps in separating clusters
- Wrapper approach
  - Step 1 Select a set of features
  - Step 2 Evaluate if they optimize the error criterion set by the clustering algorithm

Feature selection can be used to help clustering algorithms!

Take-home Messages

- Feature selection
  - Extracts a subset of informative input features
  - Preserves the data semantics
  - Faster than feature extraction
- Two main strategies
  - Filters - Separate feature selection from pattern recognition
  - Wrappers - Select features that work best with a given learning model
- Clustering
  - Finding natural groups in the data
  - Cluster number estimate $\Rightarrow$ no definitive answer
  - Can be used to support feature selection when class information is not available