Data and Text Mining

Petra Kralj Novak

December 16, 2019

http://kt.ijs.si/petra_kralj/dmtm2.html

In previous episodes …

- 23-Oct-19
  - **Data**, data types
  - Interactive **visualization** (Orange)
  - **Classification** with decision trees (root, leaves, rules, entropy, info gain, TDIDT, ID3)
- 6-Nov-19
  - Classification: train – test (evaluate) - apply
  - **Decision tree** example (on blackboard)
  - Decision tree language bias (Orange workflow)
  - Homework:
    - InfoGain questions
    - Orange workflow
    - Reading “Classification and regression by randomForest” by Liaw & Wiener, 2002
- 25-Nov-19
  - **Evaluation**:
    - Methods: train-test, leave-one-out, randomized sampling,…
    - Metrics: accuracy, confusion matrix, precision, recall, F1,…
  - Homework: XOR, questions, precision and recall

… continued …

- 2-Dec-19
  - Evaluation: **ROC**
  - **Naïve Bayes** classifier
  - Probability estimation: relative frequency, Laplace estimate
  - **Numeric prediction (**linear regression, regression tree, model tree, KNN) and **evaluation** (MSE, MAE, RMSE)
  - Homework:
    - ➢ Express F1 in terms of the entries in the confusion matrix (TP, FP, TN, FN) and simplify the equation.
    - ➢ Learn about the derivation of the Naïve Bayes formula https://en.wikipedia.org/wiki/Naive_Bayes_classifier
    - ➢ Compare the Naïve Bayes classifier with decision trees.
    - ➢ How do we evaluate the Naïve Bayes classifier? Methods, metrics.
    - ➢ Estimate the probabilities of C1 and C2 in the table below by relative frequency and Laplace estimate.
    - ➢ Loh, Wei-Yin. "Classification and regression trees." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 1.1 (2011): 14-23.
    - ➢ Compare decision and regression trees.
    - ➢ Rules of thumb when choosing the k parameter of KNN.
    - ➢ RRSE

Assignment 1:

- Express F1 in terms of the entries in the confusion matrix (TP, FP, TN, FN) and simplify the equation.

$$F_1 \quad F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}$$

		Predicted class		Total instances
		$+$	$-$	
Actual class	$+$	TP	FN	P
	$-$	FP	TN	N

$$p(C_k, x_1, \dots, x_n) = \frac{p(C_k)\ p(\mathbf{x} \mid C_k)}{p(\mathbf{x})} = \dots = p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$$

Assignment 2

$$P(A|B) = P(A,B)/P(B)$$

- The derivation of the Naïve Bayes formula

$$\begin{aligned} p(C_k, x_1, \dots, x_n) &= p(x_1, \dots, x_n, C_k) \\ &= p(x_1 \mid x_2, \dots, x_n, C_k)\ p(x_2, \dots, x_n, C_k) \\ &= p(x_1 \mid x_2, \dots, x_n, C_k)\ p(x_2 \mid x_3, \dots, x_n, C_k)\ p(x_3, \dots, x_n, C_k) \\ &= \dots \\ &= p(x_1 \mid x_2, \dots, x_n, C_k)\ p(x_2 \mid x_3, \dots, x_n, C_k) \dots p(x_{n-1} \mid x_n, C_k)\ p(x_n \mid C_k)\ p(C_k) \end{aligned}$$

Now the "naive" conditional independence assumptions come into play: assume that all features in $\mathbf{x}$ are mutually independent, conditional on the category C_k. Under this assumption,

$$p(x_i \mid x_{i+1}, \dots, x_n, C_k) = p(x_i \mid C_k)\,.$$

Thus, the joint model can be expressed as

$$\begin{aligned} p(C_k \mid x_1, \dots, x_n) &\propto p(C_k, x_1, \dots, x_n) \\ &= p(C_k)\ p(x_1 \mid C_k)\ p(x_2 \mid C_k)\ p(x_3 \mid C_k)\ \cdots \\ &= p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)\,, \end{aligned}$$

Assignment 3

- Compare the Naïve Bayes classifier with decision trees.
- How do we evaluate the Naïve Bayes classifier? Methods, metrics.
- Estimate the probabilities of C1 and C2 in the table below by relative frequency and Laplace estimate.

Number of events		**Relative frequency**		**Laplace estimate**	
Class C1	Class C2	P(C1)	P(C2)	P(C1)	P(C2)
0	2				
12	88				
12	988				
120	880				

Assignment 3

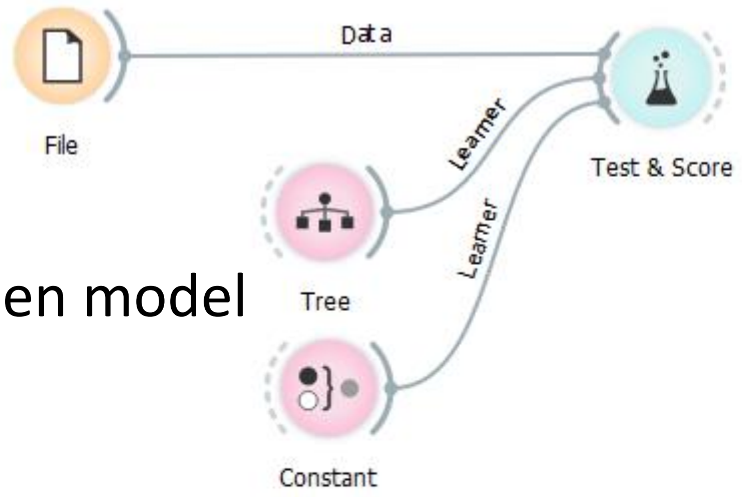
- Compare the Naïve Bayes classifier with decision trees.
- How do we evaluate the Naïve Bayes classifier? Methods, metrics.
- Estimate the probabilities of C1 and C2 in the table below by relative frequency and Laplace estimate.

Number of events		**Relative frequency**		**Laplace estimate**	
Class C1	Class C2	P(C1)	P(C2)	P(C1)	P(C2)
0	2	0	1	0.25	0.75
12	88	0.12	0.88	0.127451	0.872549
12	988	0.012	0.988	0.012974	0.987026
120	880	0.12	0.88	0.120758	0.879242

Assignment 4

- Loh, Wei-Yin. "Classification and regression trees." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 1.1 (2011): 14-23.
  - Different ideas to overcome the shortcoming of primitive decision trees
  - Different algorithms for DT construction yield different results
  - Regression trees algorithms
- Compare decision and regression trees.
- Rules of thumb when choosing the k parameter of KNN.

Assignment 5

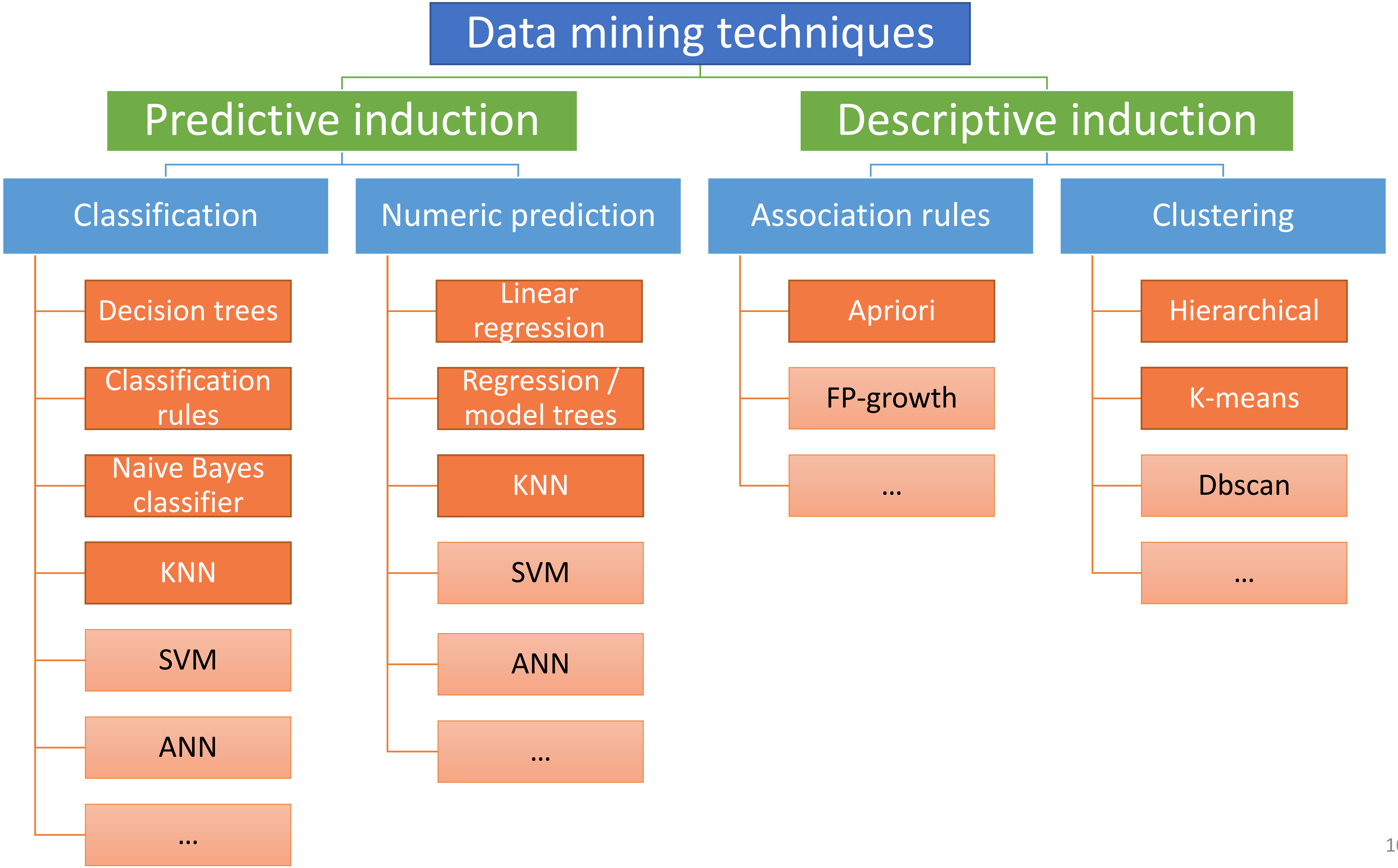


- Use Orange and a calculator to compute RRSE for a chosen model
- Data: regressionAgeHeight.csv

- RRSE = root relative squared error
  - Nominator: sum of squared differences between the actual and the expected values
  - Denominator: sum of squared errors

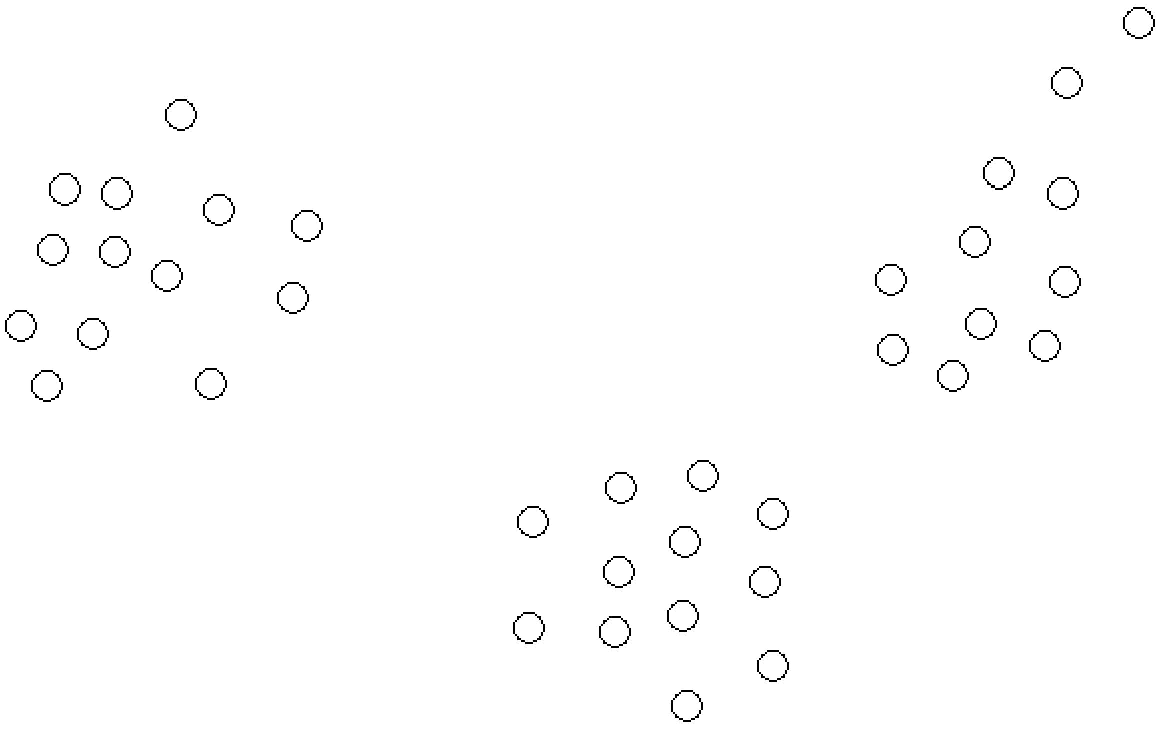
$$RRSE = \sqrt{\frac{\sum_{i=1}^{n}(p_i - a_i)^2}{\sum_{i=1}^{n}(\overline{a} - a_i)^2}}$$

  - RRSE: Ratio between the error of the model and the error of the naïve model (predicting the average)
  - Hint: If we divide both the nominator and the denominator by n we get RSE of the model and const model.

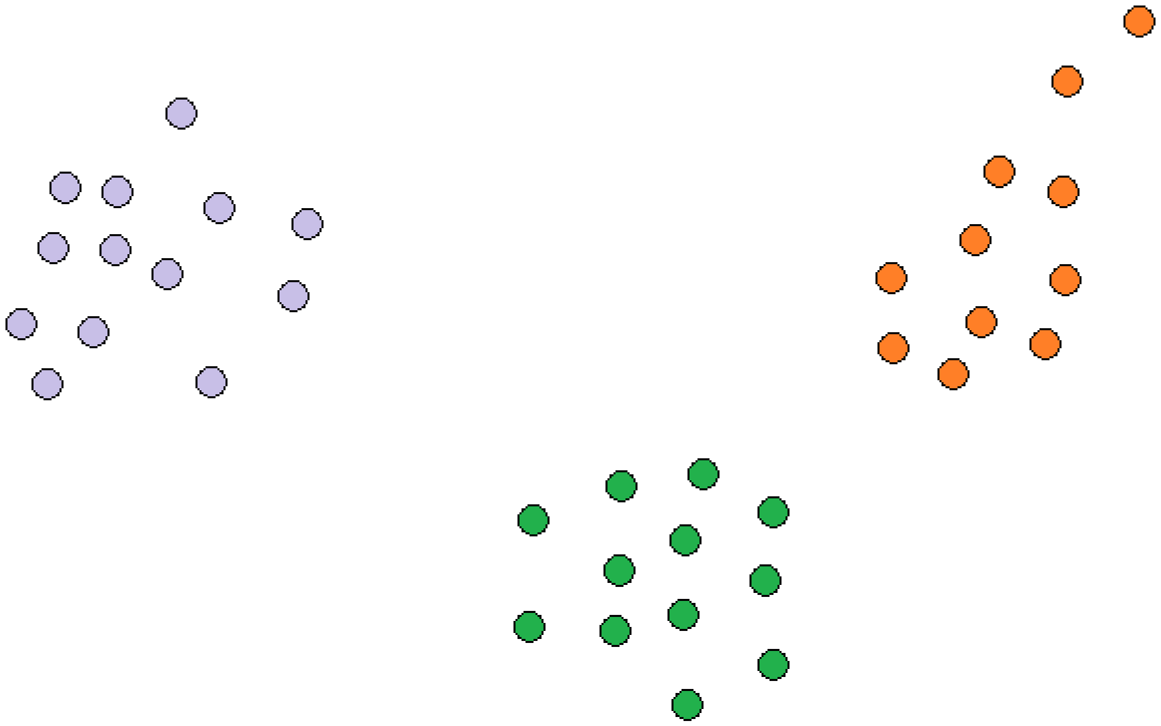

Data mining techniques
Predictive induction
Descriptive induction
Classification
Numeric prediction
Association rules
Clustering
Decision trees
Classification rules
Naive Bayes classifier
KNN
SVM
ANN
…
Linear regression
Regression / model trees
KNN
SVM
ANN
…
Apriori
FP-growth
…
Hierarchical
K-means
Dbscan
…

Clustering

Clustering



Clustering

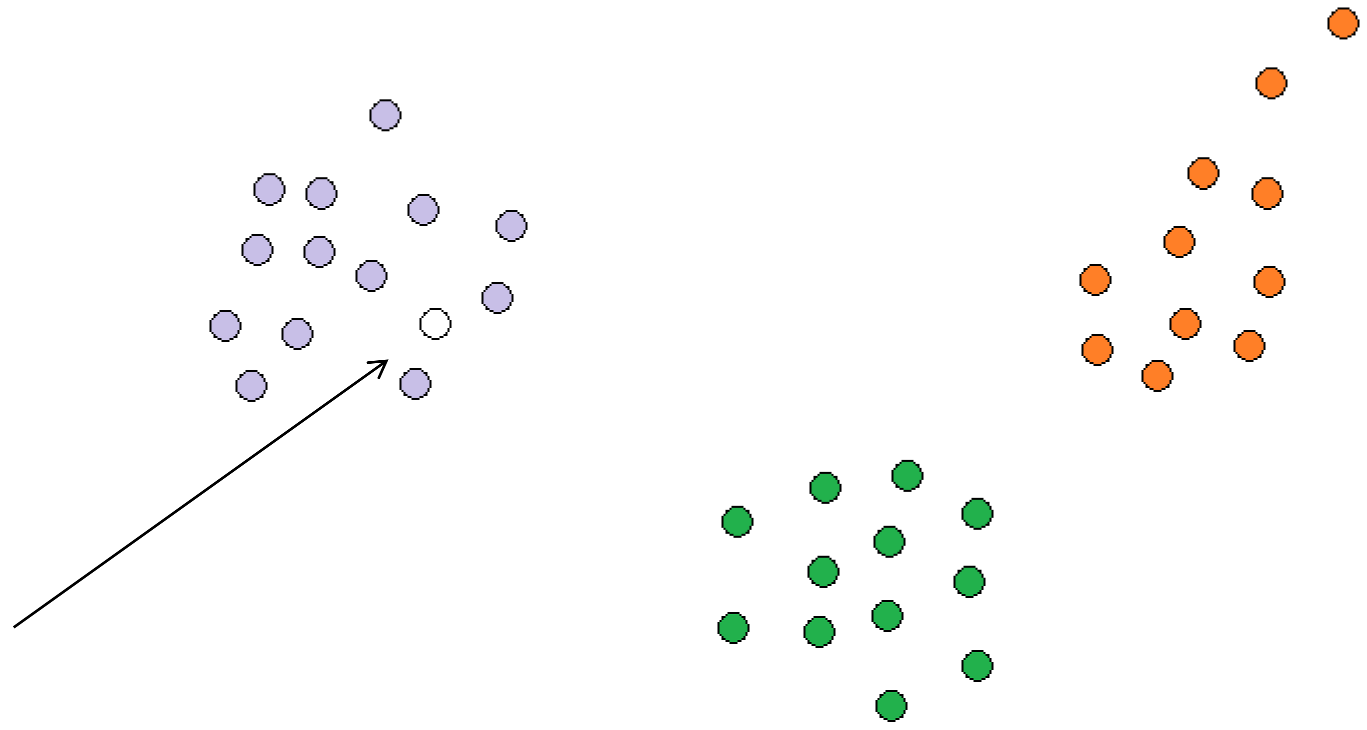


Clustering

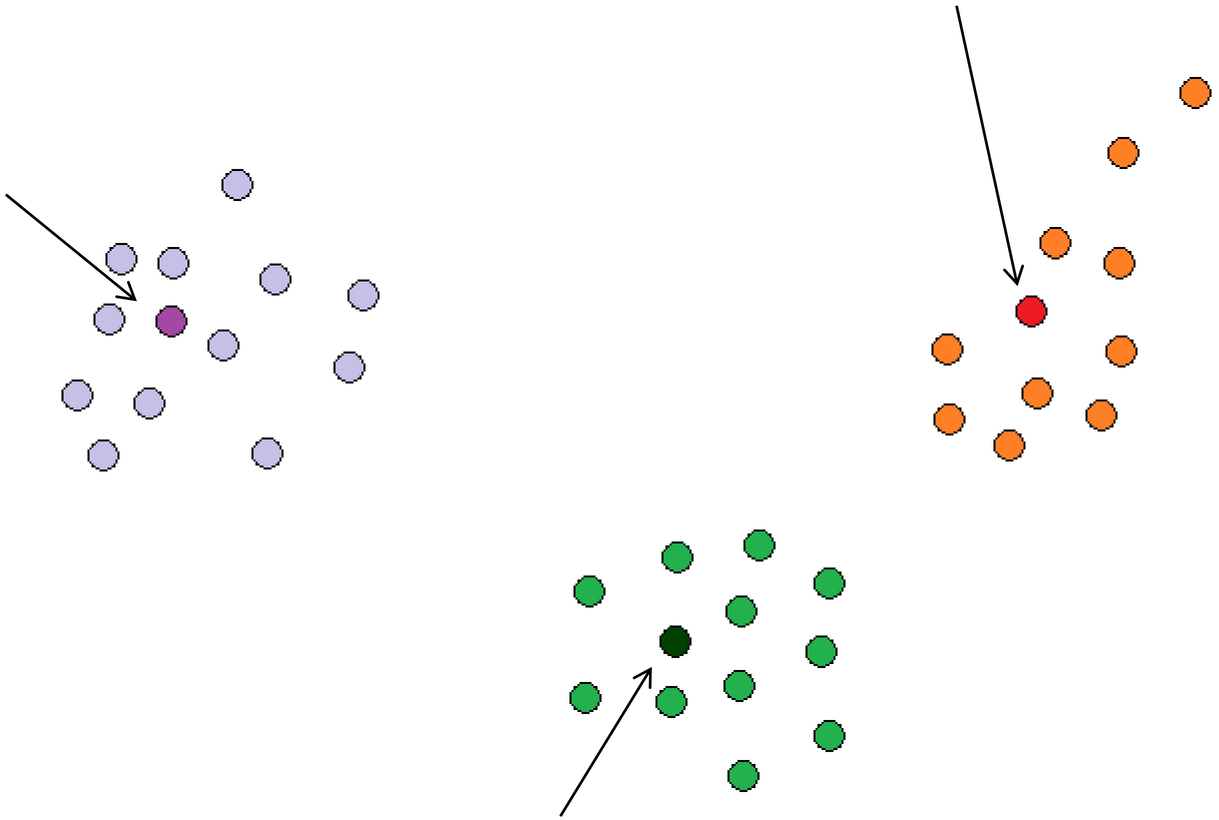
- … is the process of grouping the data instances into clusters so that objects within a cluster have high similarity but are very dissimilar to objects in other clusters.

- Wish list:
  - Identity clusters irrespective of their shapes
  - Scalability
  - Ability to deal with noisy data
  - Insensitivity to the order of input records

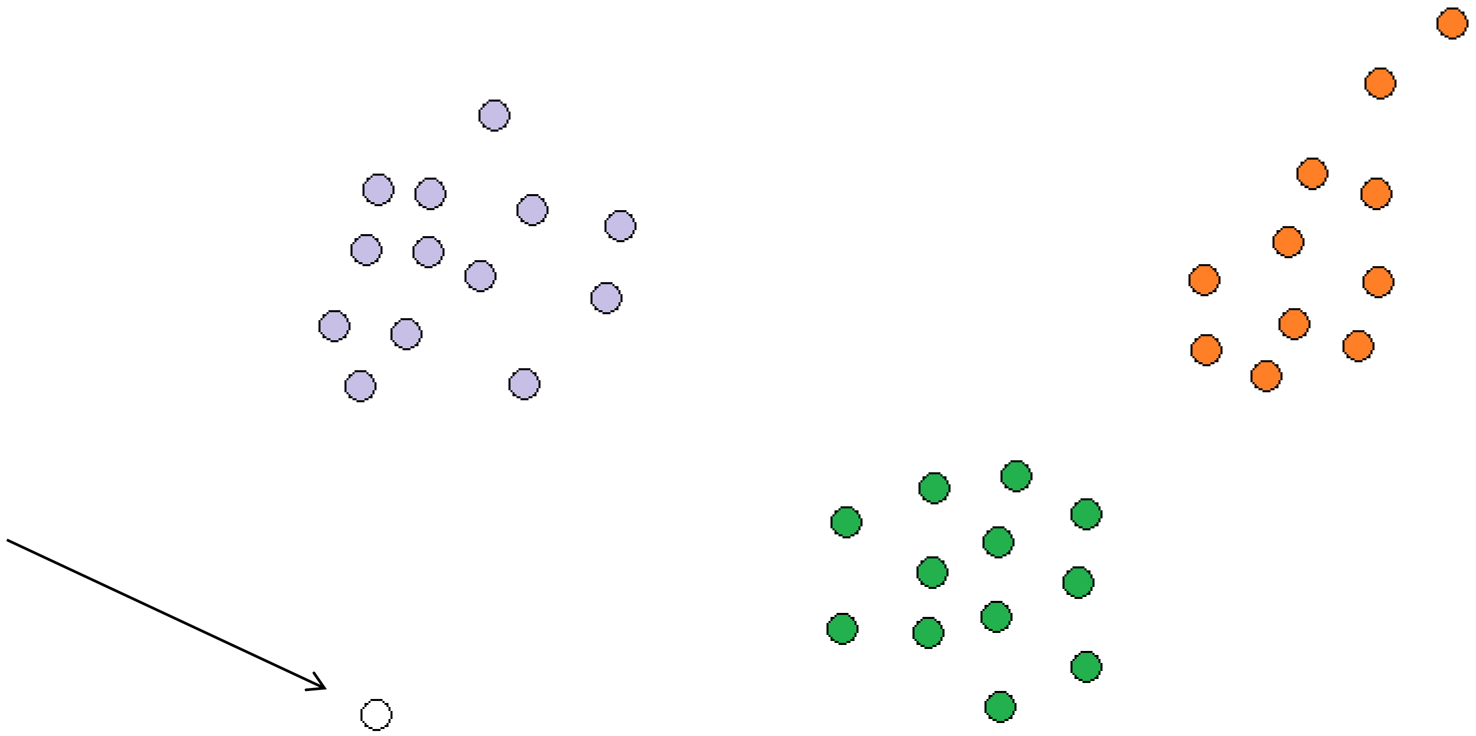
Unsupervised classification



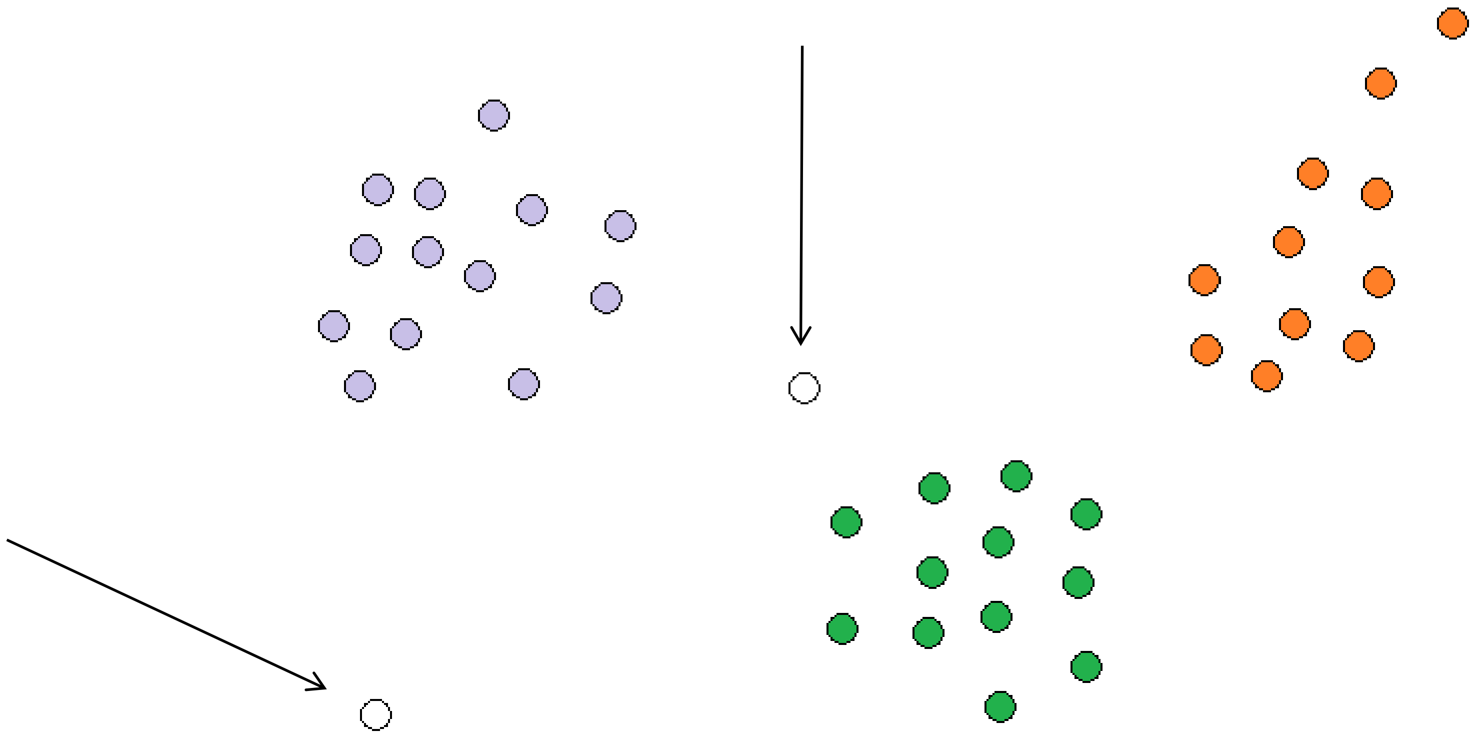
Data summarization: centroid, medoid



Outlier detection



Outlier detection



Applications

- Data mining
  - Unsupervised classification
  - Data summarization
  - Outlier analysis
  - …
- Customer segmentation and collaborative filtering
- Text applications
- Social network analysis

Text applications

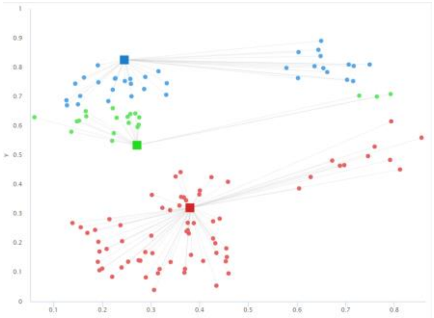
Clustering types

- Partitioning
  - k-means, k-medoids, k-modes
- Hierarchical
  - Agglomerative
- Grid-based
  - Multi-resolution grid structure
  - Efficient and scalable
- Density-based
  - A cluster is a dense region of points, which is separated by low density regions, from other regions of high density
  - Algorithms: DBSCAN, OPTICS, DenClue

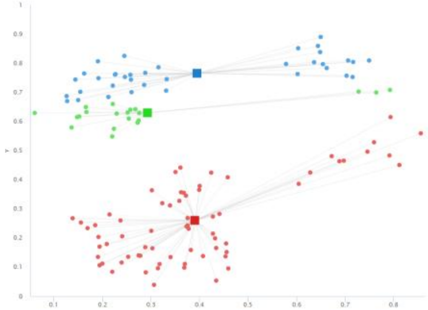
Interactive k-Means

K-Means example

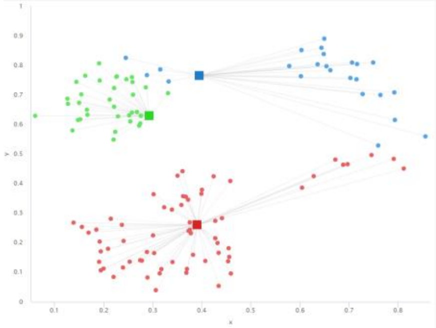
Random initialization



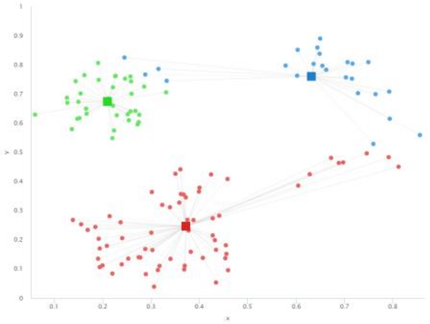
Centroid computation



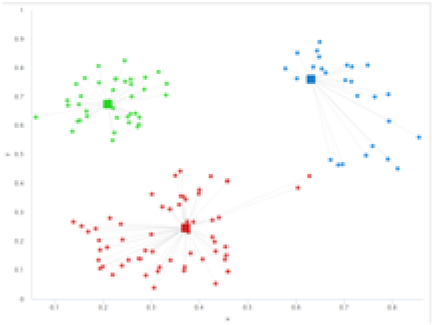
Assignment of points to the nearest centroid



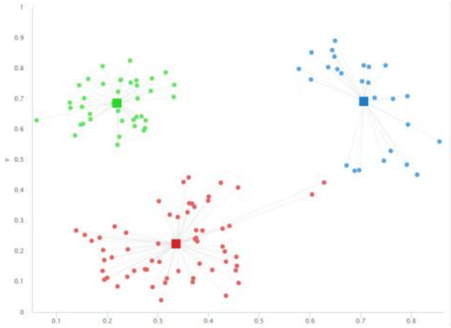
Centroid computation



Assignment of points to the nearest centroid



Centroid computation



K-means

1. Choose **k** random instances as cluster centers
2. Assign each instance to its closest cluster center
3. Recompute cluster centers by computing the average (aka *centroid*) of the instances pertaining to each cluster
4. If cluster centers have moved, go back to Step 2

(Equivalent termination criterion: stop when assignment of instances to cluster centers has not changed)

Alternatives: K-medoids, K-modes

- Might get stuck in local minima
- Silhuette for finding the optimal K

Clustering evaluation

- Clustering analysis doesn't have a solid evaluation metric

- External validation criteria
  - Using the ground truth to evaluate to evaluate the clustering result
- Internal validation criteria
  - Sum of distances to centroids
  - Intracluster to intercluster distance ratio
  - Silhouette coefficient

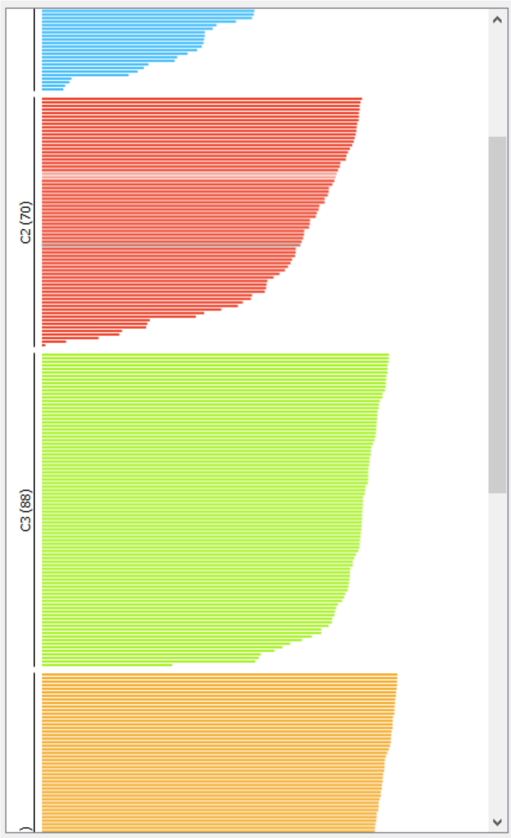
  - Parameter tuning – the "elbow" method

Aggarwal, Charu C. *Data mining: the textbook*. Springer, 2015. Chapter 6: cluster analysis, pgs 195 -201

Silhouette coefficient

- The silhouette value is a measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation).

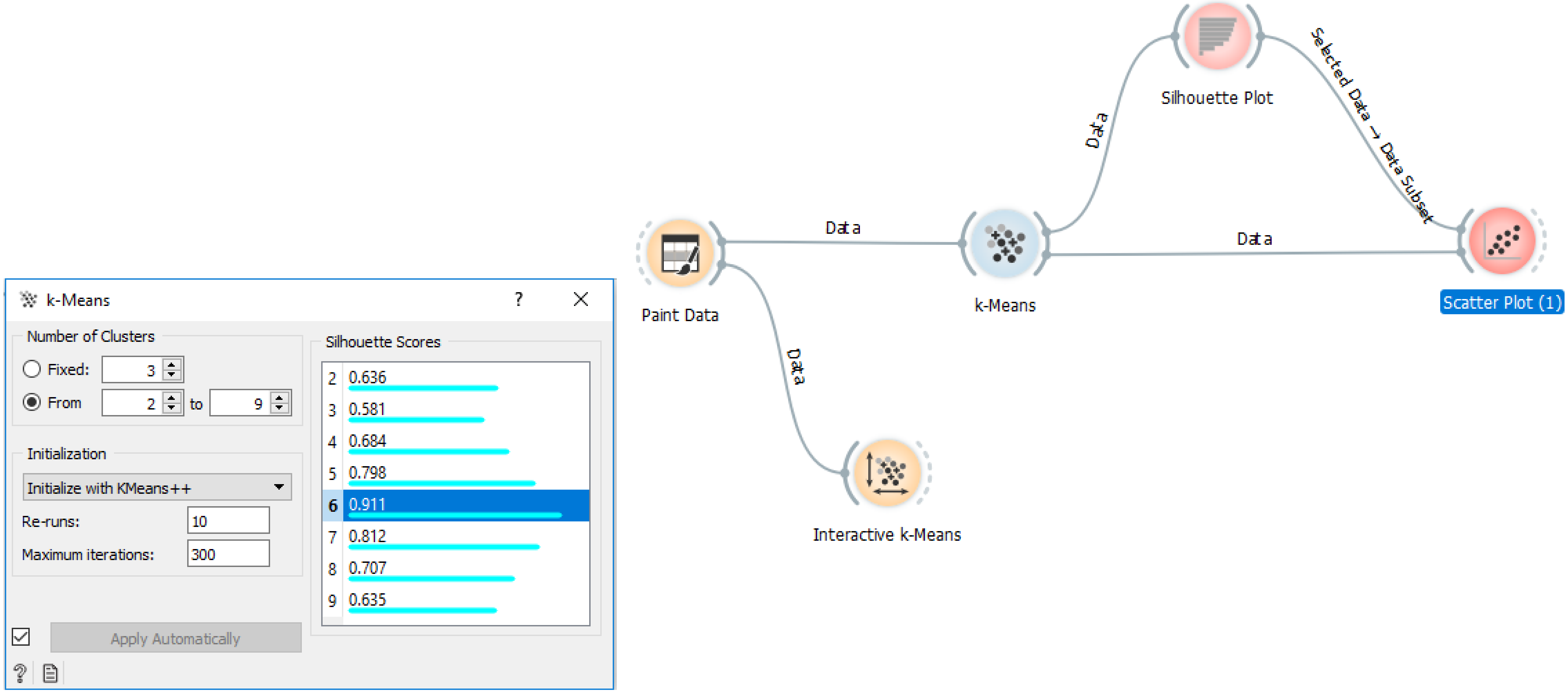
- For example x_i, its silhouette coefficient is $s_i = (b_i - a_i) / \max(a_i, b_i)$
  - a_i average distance between x_i to all other examples in its cluster.
  - b_i average distance between x_i to the examples in the "neighboring" cluster
- The overall silhouette coefficient is the average of the data point-specific coefficients.



Homework

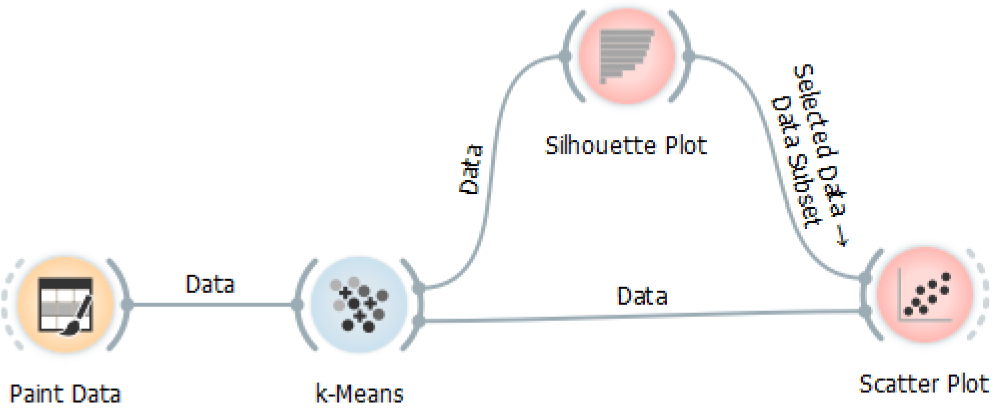
- How can we use the silhouette coefficient for searching for outliers in classification problems?

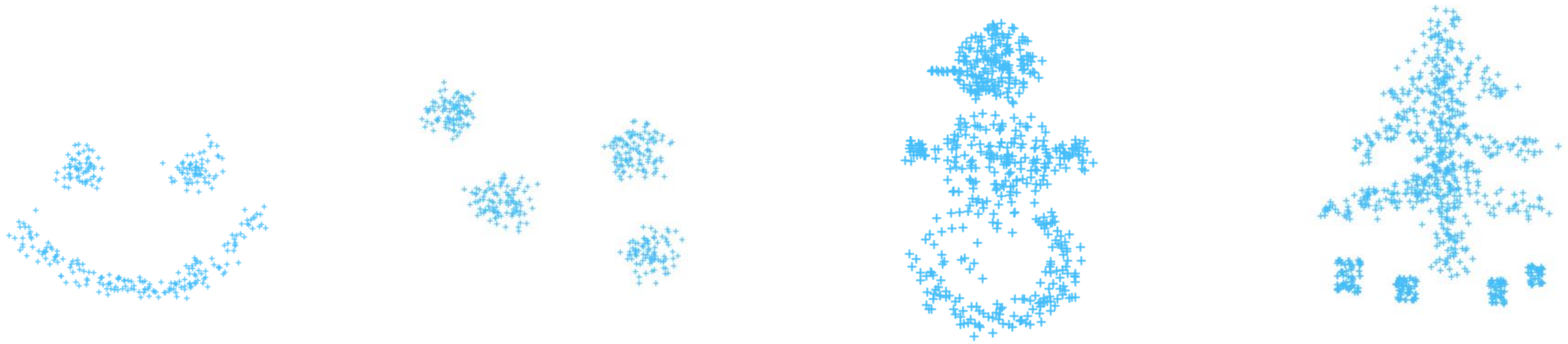
k-Means + Silhouette + „reruns“



Lab exercise: clustering Christmas drawings

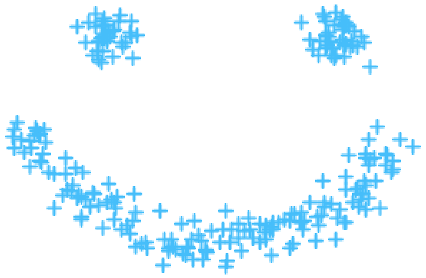
- Groups, each clusters one drawing:
  - Four snowballs
  - A smiley face
  - A Christmas tree with presents
  - A snowman

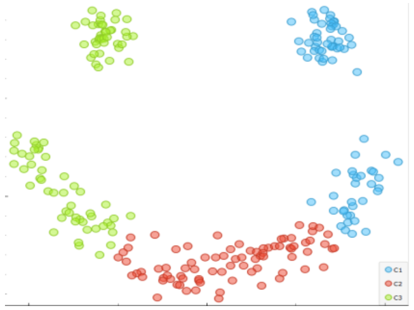


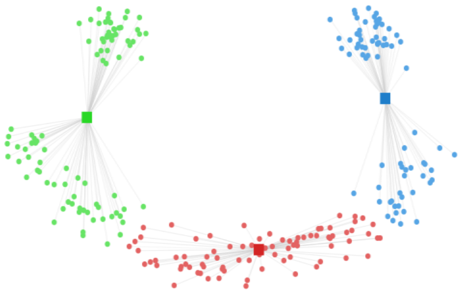


Properties of k-Means

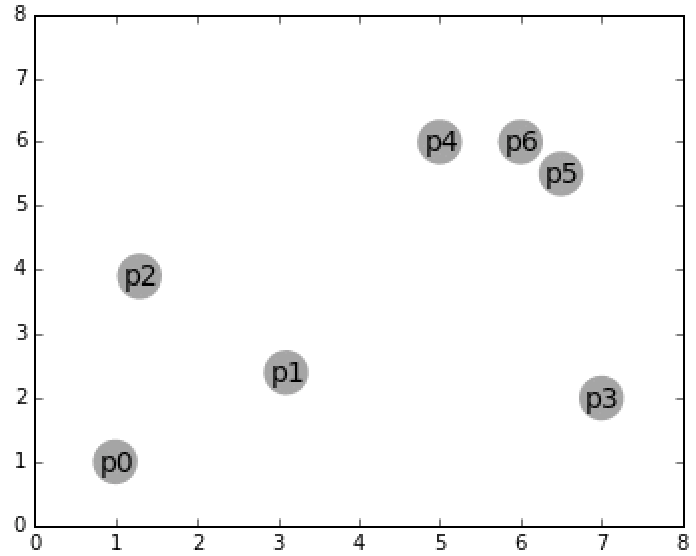
- The number of clusters ***k*** is fixed in advance
- It is fast, it always converges
- Can converge into a local minima (bad solution because of unlucky start)
- Finds “spherical” shaped clusters
- K-Means will cluster the data even if it can’t be clustered (e.g. data that comes from uniform distributions)



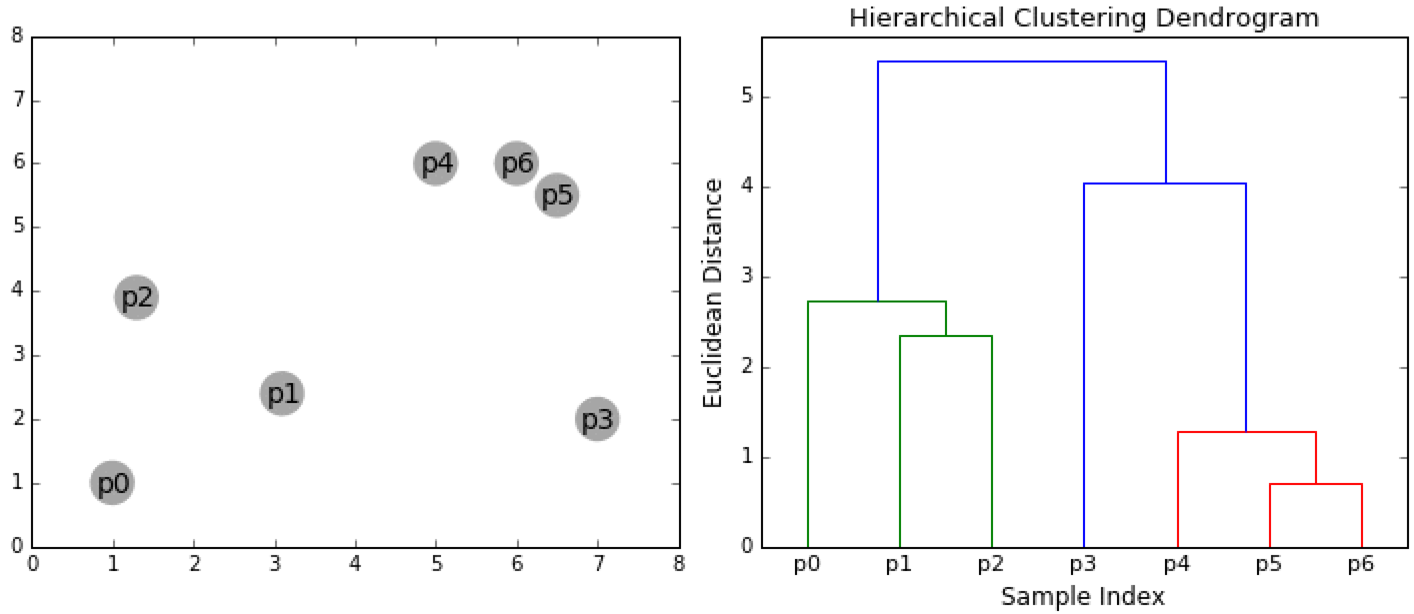




Agglomerative clustering - example



Agglomerative clustering - dendrogram



Agglomerative clustering

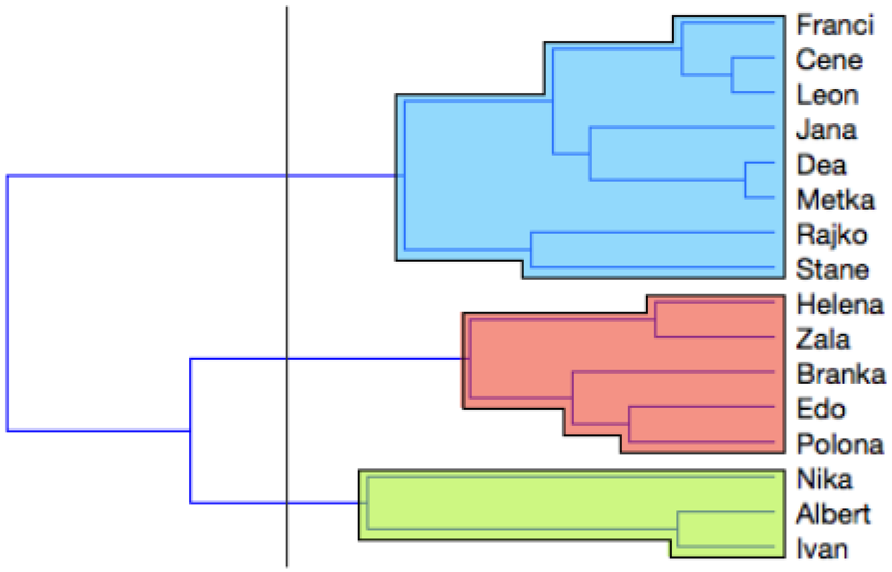
1. Start with a collection **C** of **n** singleton clusters
   - Each cluster contains one data point $\mathbf{c_i} = \{\mathbf{x_i}\}$
2. Repeat until only one cluster is left:
   1. Find a pair of clusters that is closest: min $\mathbf{D(c_i, c_j)}$
   2. Merge the clusters $\mathbf{c_i}$ and $\mathbf{c_j}$ into $\mathbf{c_{i+j}}$
   3. Remove $\mathbf{c_i}$ and $\mathbf{c_j}$ from the collection **C**, add $\mathbf{c_{i+j}}$

Some new index, not a sum

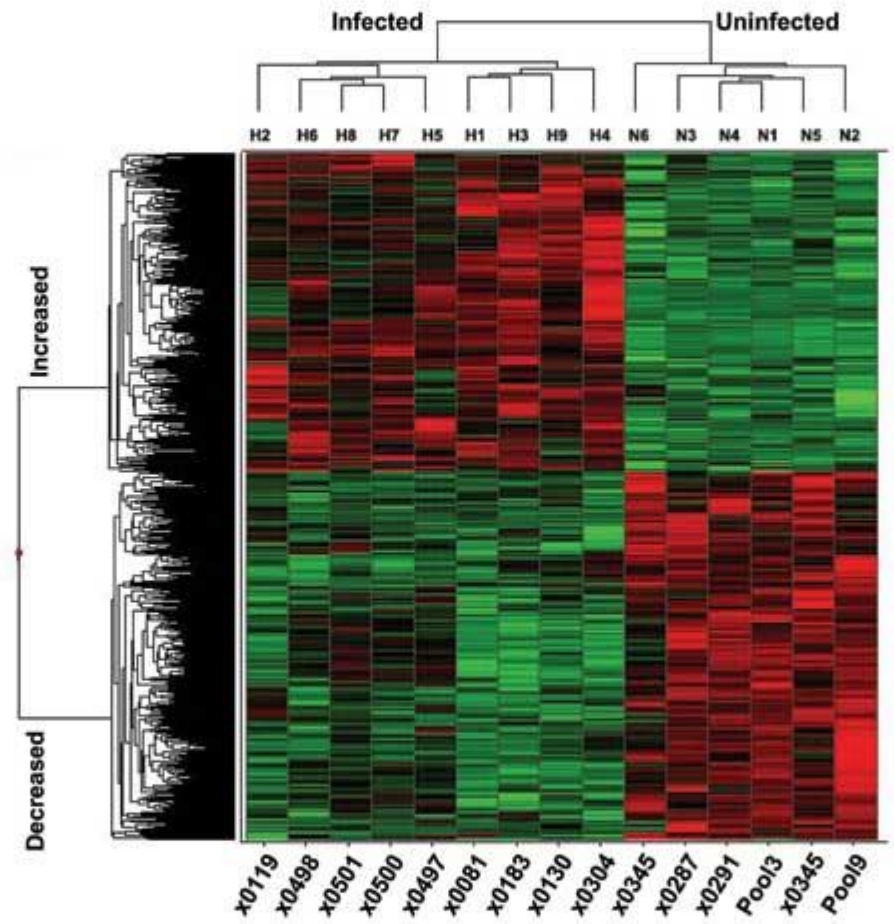
- Time and space complexity
- Sensitive to noisy data

Dendrogram

- The agglomerative hierarchical clustering algorithms result is commonly displayed as a tree diagram called a dendrogram.
- Dendrogram a tree diagram for showing taxonomic relationships.



Example: Hierarchical clustering of genes

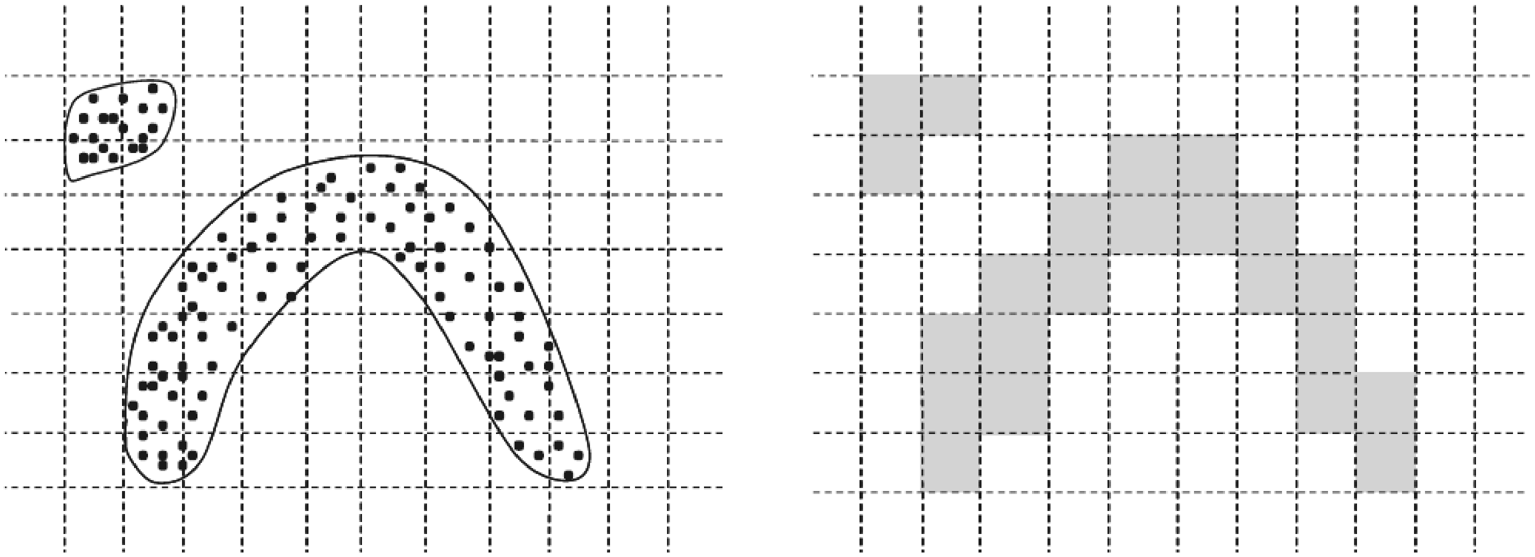


Grid-based (parameters **p** and **τ**)

1. Discretize each dimension of **D** into **p** ranges
2. Determine dense grid cells at level **τ**
3. Create graph where dense grid cells are connected if they are adjacent
4. Determine connected components of graph
5. Return: points in each connected component as a cluster

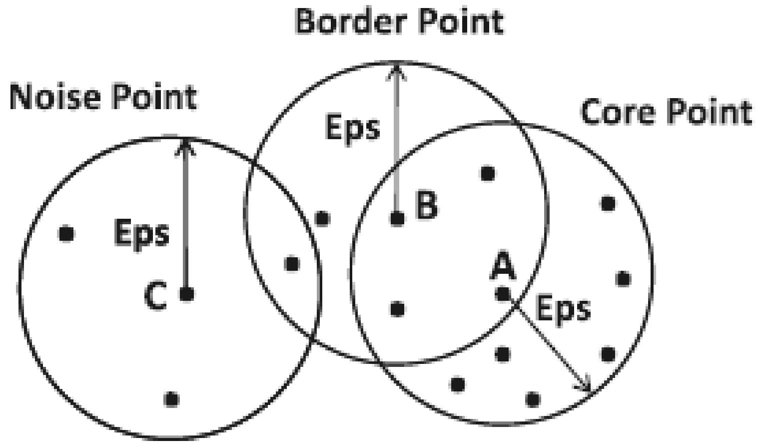
Grid-based (parameters **p** and **τ**)

1. Discretize each dimension of **D** into **p** ranges
2. Determine dense grid cells at level **τ**
3. Create graph where dense grid cells are connected if they are adjacent
4. Determine connected components of graph
5. Return: points in each connected component as a cluster



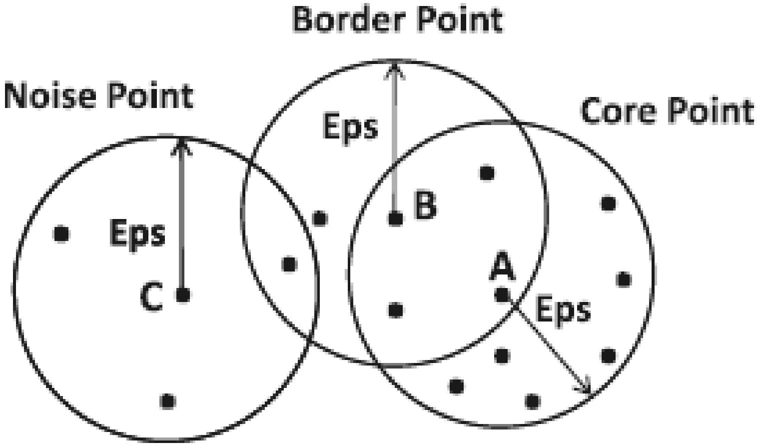
Density based clustering
DBSCAN(Data: *D*, Radius: *Eps*, Density: τ)

- *Core point:* A data point is defined as a *core* point, if it contains at least τ data points within a radius *Eps* within a radius *Eps*.
- *Border point:* A data point is defined as a *border* point, if it contains less than τ points, but it also contains at least one core point within a radius *Eps*.
- *Noise point:* A data point that is neither a core point nor a border point is defined as a *noise* point.



Density based clustering
DBSCAN(Data: *D*, Radius: *Eps*, Density: τ)

1. Determine core, border and noise points of *D* at level (*Eps, τ*);
2. Create graph in which core points are connected if they are within *Eps* of one another;
3. Determine connected components in graph;
4. Assign each border point to connected component with which it is best connected;
5. **Return** points in each connected component as a cluster;



Aggarwal, Charu C. *Data mining: the textbook*. Springer, 2015. Chapter 6: cluster analysis, pg 183

DBSCAN properties

Similar to grid-based approaches, except that it uses circular regions as building blocks.

Advantages of DBSCAN:

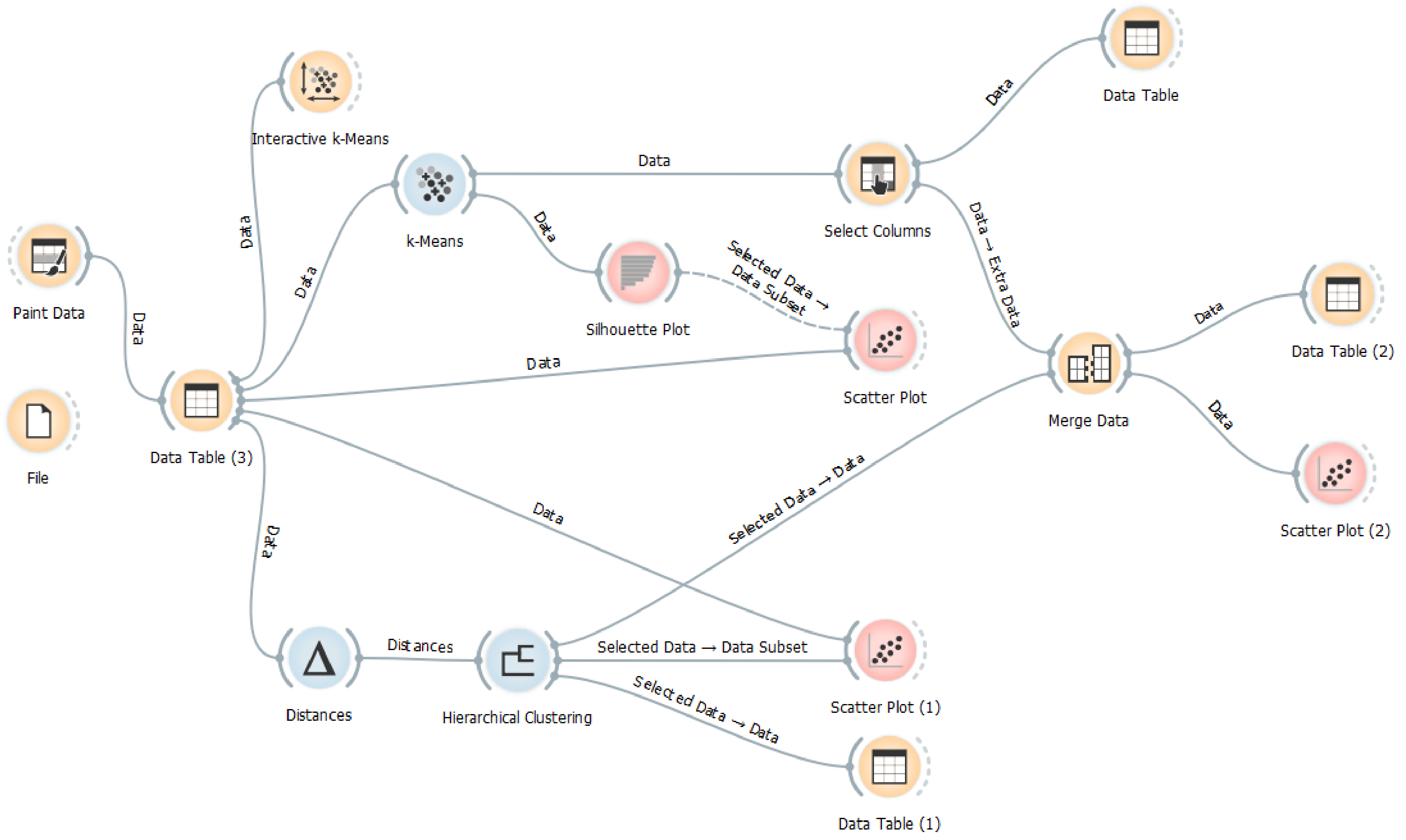
- Can detect clusters of arbitrary shape.
- Does not require the number of clusters as an input parameter.
- Detects clusters of different shapes.
- Not sensitive to outliers.

Disadvantages of DBSCAN:

- Computationally expensive in the first step (Determine core, border and noise points of D at level (Eps, τ);
- Susceptible to variations in the local cluster density.
- Struggles with high dimensionality data.

Lab work in Orange

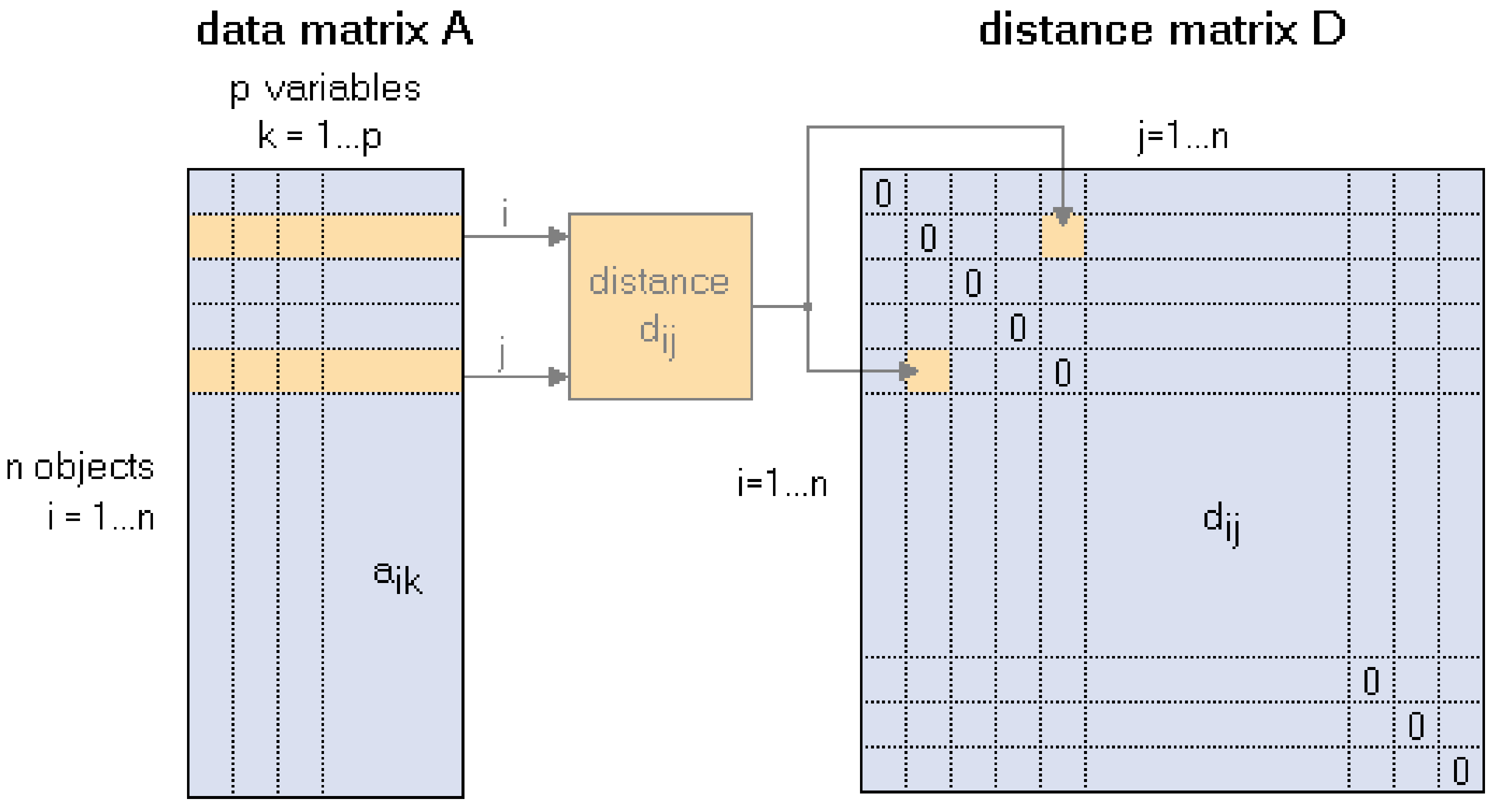
- Comparison of hierarchical and k-Means clustering on
- painted data
- „wine.tab“, where we compare also to the real classes



Similarity / distance measures

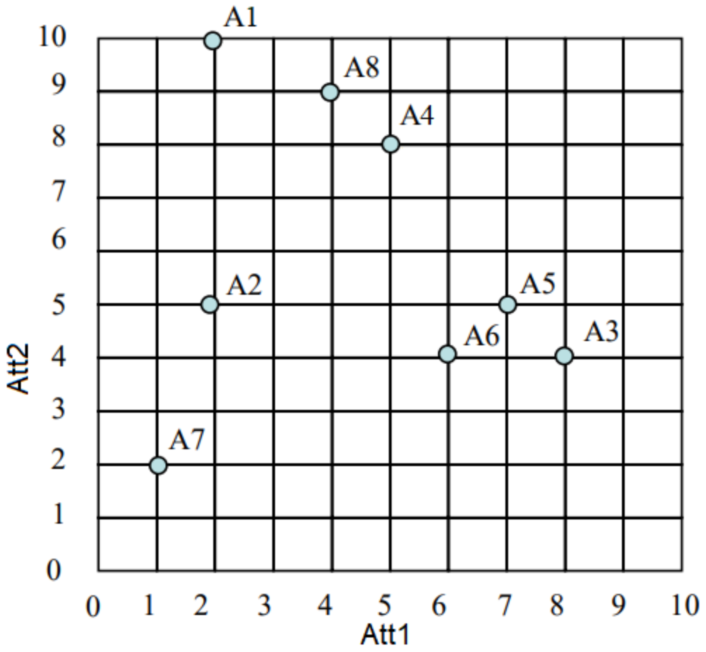
- The similarity measure depends on characteristics of the input data:
  - Attribute type: binary, categorical, continuous
  - Sparseness
  - Dimensionality
  - Type of proximity

Distance matrix



Distance matrix example

	Att1	Att2
A1	2	10
A2	2	5
A3	8	4
A4	5	8
A5	7	5
A6	6	4
A7	1	2
A8	4	9



	A1	A2	A3	A4	A5	A6	A7	A8
A1	0	$\sqrt{25}$	$\sqrt{36}$	$\sqrt{13}$	$\sqrt{50}$	$\sqrt{52}$	$\sqrt{65}$	$\sqrt{5}$
A2		0	$\sqrt{37}$	$\sqrt{18}$	$\sqrt{25}$	$\sqrt{17}$	$\sqrt{10}$	$\sqrt{20}$
A3			0	$\sqrt{25}$	$\sqrt{2}$	$\sqrt{2}$	$\sqrt{53}$	$\sqrt{41}$
A4				0	$\sqrt{13}$	$\sqrt{17}$	$\sqrt{52}$	$\sqrt{2}$
A5					0	$\sqrt{2}$	$\sqrt{45}$	$\sqrt{25}$
A6						0	$\sqrt{29}$	$\sqrt{29}$
A7							0	$\sqrt{58}$
A8								0

Euclidian → $Dist(A,B) = \sqrt[2]{(Att1(A) - Att1(B))^2 + (Att2(A) - Att2(B))^2}$

Distance measures

Euclidean	$d(x,y)=\sqrt{\sum(x_i-y_i)^2}$
Squared Euclidean	$d(x,y)=\sum(x_i-y_i)^2$
Manhattan	$d(x,y)=\sum\left\|(x_i-y_i)\right\|$
Canberra	$d(x,y)=\sum\frac{\left\|x_i-y_i\right\|}{\left\|x_i+y_i\right\|}$
Chebychev	$d(x,y)=\max\left(\left\|x_i-y_i\right\|\right)$
Bray Curtis	$d(x,y)=\frac{\sum\left\|x_i-y_i\right\|}{\sum x_i+y_i}$
Cosine Correlation	$d(x,y)=\frac{\sum(x_i y_i)}{\sqrt{\sum(x_i)^2\sum(y_i)^2}}$
Pearson Correlation	$d(x,y)=\frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum(y_i-\bar{y})^2}\sqrt{\sum(y_i-\bar{y})^2}}$
Uncentered Peason Correlation	$d(x,y)=\frac{\sum x_i y_i}{\sqrt{\sum(y_i-\bar{y})^2}\sqrt{\sum(y_i-\bar{y})^2}}$
Euclidean Nullweighted	Same as Euclidean, but only the indexes where both x and y have a value (not NULL) are used, and the result is weighted by the number of values calculated. Nulls must be replaced by the missing value calculator (in dataloader).

Minkowski distance

$$D(X,Y)=\left(\sum_{i=1}^{n}|x_i-y_i|^p\right)^{1/p}$$

Aggarwal, C. C. (2015). *Data mining: the textbook*. Springer. (Chapter 3)

Homework

- Similarity vs. distance
- List algorithms that are based on distance/similarity

Literature

- Max Bramer: Principles of data mining (2007)
  - 14. Clustering
- Aggarwal, Charu C. *Data mining: the textbook*. Springer, 2015. Chapter 6: cluster analysis, pgs 195 -201