Course. Introduction to Machine Learning
Work 1. Clustering Exercise

Dr. Maria Salamó Llorente

Dept. Mathematics and Informatics,

Faculty of Mathematics and Informatics,
University of Barcelona

Contents

1. **Clustering exercise**
    1. Preprocess the data
    2. Agglomerative Clustering with sklearn
    3. K-Means (your own code)
    4. K-Modes (your own code)
    5. K-Prototype (your own code)
    6. Fuzzy clustering (your own code)
    7. Validation techniques (using sklearn validation metrics)

Preprocess the data

Data in

- You need to read the .arff file
  - You can implement your own code or use scipy.io.arff.loadarff
- Data needs pre-processing
  - Features may contain **different ranges**
    - Normalize or Standarize the machine learning data
  - Features may have **different types**
    - Categorical, Numerical, and mix-type data
  - Data may contain **missing values**
    - Use the median (for example)

Data pre-processing

- **To deal with different ranges**
  - Normalize or scale features
- Alternatives
  - **Standardisation**: Standardisation replaces the values by their Z scores. `sklearn.preprocessing.scale`
  - **Mean normalisation**: This distribution will have values between **-1 and 1**with **μ=0**. `sklearn.preprocessing.StandardScaler`
  - **Min-Max scaling**: This scaling brings the value between 0 and 1. `sklearn.preprocessing.MinMaxScaler`
  - **Unit vector**: Scaling is done considering the whole feature vector to be of unit length. `sklearn.preprocessing.Normalizer`

Data pre-processing

- **To deal with different types**
- Alternatives
  - **Label encoding**: convert to a number

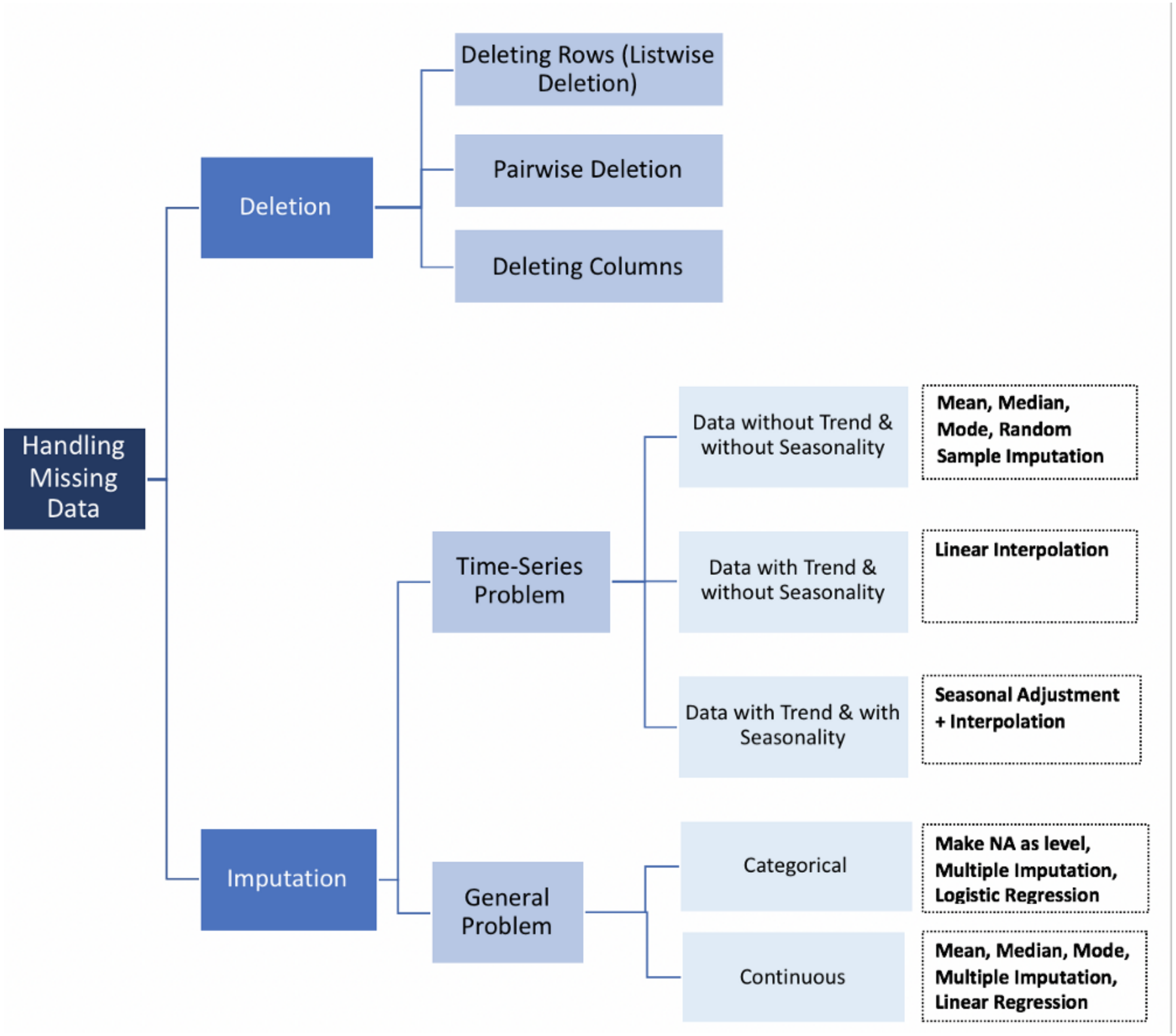
    ```
    sklearn.preprocessing.LabelEncoder
    ```

  - **One hot encoding**: where a categorical variable is converted into a binary vector, each possible value of the categorical variable becomes the variable itself with default value of zero and the variable which was the value of the categorical variable will have the value 1.

    ```
    sklearn.preprocessing.OneHotEncoder
    ```

Data pre-processing

- **To deal with missing values**


Handling Missing Data
Deletion
Deleting Rows (Listwise Deletion)
Pairwise Deletion
Deleting Columns
Imputation
Time-Series Problem
Data without Trend & without Seasonality
Mean, Median, Mode, Random Sample Imputation
Data with Trend & without Seasonality
Linear Interpolation
Data with Trend & with Seasonality
Seasonal Adjustment + Interpolation
General Problem
Categorical
Make NA as level, Multiple Imputation, Logistic Regression
Continuous
Mean, Median, Mode, Multiple Imputation, Linear Regression

Agglomerative Clustering

Using sklearn

Agglomerative Clustering

Some Videos

- https://www.youtube.com/watch?v=VMyXc3SiEqs
- https://www.youtube.com/watch?v=RdT7bhm1M3E
- https://www.youtube.com/watch?v=Cy3ci0Vqs3Y
- http://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

K-Means

Implement your own code

K-Means basis

- It is a partitional algorithm that …
  - Assumes instances are **real-valued vectors**
  - Clusters based on ***centroids, center of gravity***, or **mean of points** in a cluster, ***c***:

$$\vec{\mu}(\mathrm{c}) = \frac{1}{|c|} \sum_{\vec{x} \in c} \vec{x}$$

  - Reassignment of instances to clusters is **based on distance** to the current cluster centroids
    - Manhattan distance (L_1 norm), Euclidean distance (L_2 norm), Cosine similarity

Discussion on the K-Means method

- K-Means clustering often **terminates at a local optimal**
  - Initialization can be important to find high-quality clusters
- **Need to specify K**, the number of clusters, in advance
  - There are ways to automatically determine the "*best*" K
  - In practice, one often runs a range of values and selected the "*best*" K value
- **Sensitive to noisy data and outliers**
  - Variations: Using K-medians, K-medoids, etc.
- K-Means is applicable only to objects in a **continuous n-dimensional space**
  - Using the K-Modes for **categorical data**
- Non suitable to discover clusters with **non-convex shapes**
  - Using density-based clustering, kernel k-means, etc.

Variations of K-Means

- There are many variants of the K-Means methods, varying different aspects
  - Choosing better initial centroid estimates
    - K-Means++, Intelligent K-Means, Genetic K-Means
  - Choosing different representatives for the clusters
    - K-Medoids, K-Medians, K-Modes
  - Applying feature transformation techniques *(explained at the supervised part of the course)*
    - Weighted K-Means, Kernel K-Means

Initialization of K-Means

- Different initializations may generate rather different clustering results
- Original proposal (MacQueen,1967): selects the k seed randomly
    - Need to run the algorithm multiple times using different seeds
- There are many methods proposed for better initialization of K seeds
    - K-Means++ (Arthur and Vassilvitskii,2007):
        - The first centroid is selected randomly
        - The next centroid selected is the one that is farthest from the currently selected (selection is based on a weighted probability score).
        - The selection continues until K centroids are obtained

Some k-Means references

- MacQueen, J. B. (1967). **Some Methods for classification and Analysis of Multivariate Observations**. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press. pp. 281–297. **(in RACÓ)**

- Celebi, M. E., Kingravi, H. A., and Vela, P. A. (2013). **A comparative study of efficient initialization methods for the k-means clustering algorithm**. Expert Systems with Applications. 40 (1): 200–210. **(in RACÓ)**

- Arthur, D.; Vassilvitskii, S. (2007). **K-means++: the advantages of careful seeding**. Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics Philadelphia, PA, USA. pp. 1027–1035. **(in RACÓ)**

K-Modes

K-Modes for categorical data

- K-Means cannot handle non-numerical (categorical) data
  - Mapping categorical value to 1/0 cannot generate quality clusters for high-dimensional data
- K-Modes is a variation of the *K-Means* Method (Huang'98)
  - Replacing means of clusters with modes
  - Using new dissimilarity measures to deal with categorical objects
  - Using a frequency-based method to update modes of clusters

UNIVERSITAT DE BARCELONA

K-Modes basis

- K-Modes: an extension to K-Means by replacing means with **modes**

$$\Phi(x_j, z_j) = 1 - n_j^r/n_l \text{ when } x_j = z_j \text{ ; 1 when } x_j \neq z_j$$

where z_j is the categorical value of attribute j in Z_l, n_l is the number of objects in cluster l, and n_j^r is the number of objects whose attribute value is r

- Dissimilarity measure between object X and the center of a cluster Z
- The dissimilarity measure (distance function) is **frequency-based**

$$d(X_i, X_l) \equiv \sum_{j=1}^{m} \delta(x_{i,j}, x_{l,j})$$

where

$$\delta(x_{i,j}, x_{l,j}) = \begin{cases} 0, & x_{i,j} = x_{l,j} \\ 1, & x_{i,j} \neq x_{l,j} \end{cases}$$

K-Modes algorithm

- K-Modes deals with categorical attributes

```
Insert the first K objects into K new clusters.
Calculate the initial K modes for K clusters.
Repeat {
    For (each object O) {
       Calculate the similarity between object O and the
       modes of all clusters.
       Insert object O into the cluster C whose mode is the
       least dissimilar to object O.
    }
       Recalculate the cluster modes so that the cluster
       similarity between mode and objects is maximized.
} until (num_iterations or few objects change clusters).
```

K-Modes

- Algorithm is still based on iterative object cluster assignment and centroid update
- A **fuzzy k-modes** method is proposed to calculate a **fuzzy cluster membership** value for each object to each cluster
- A mixture of categorical and numerical data: Using a **K-prototype** method

References of K-Modes

- Zhexue Huang and Michael K. Ng. 2003. **A Note on K-Modes Clustering**. J. Classif. 20, 2 (September 2003), 257-261. DOI=http://dx.doi.org/10.1007/s00357-003-0014-4 **(in RACÓ)**
- Anil Chaturvedi, Paul E. Green, and J. Douglas Caroll. 2001. **K-Modes Clustering**. J. Classif. 18, 1 (January 2001), 35-55. DOI=http://dx.doi.org/10.1007/s00357-001-0004-3 **(in RACÓ)**
- Zengyou He, **Approximation algorithms for K-Modes clustering**. https://arxiv.org/pdf/cs/0603120.pdf
- Fuyuan Cao, Jive Liang, Deyu Li, Liang Bai, Chuangyin Dang. **A dissimilarity measure for the K-Modes clustering algorithm**. Knowledge-based Systems, Volume 26, 2012, ISSN 0950-7051.DOI= https://doi.org/10.1016/j.knosys.2011.07.011.
  http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.652.5571&rep=rep1&type=pdf

K-Prototypes

K-prototypes Algorithm

- To integrate the k-means and k-modes algorithms into the k-prototypes algorithm that is used to cluster the mixed-type objects

- $A_1^r, A_2^r, ..., A_p^r, A_{p+1}^c, ..., A_m^c$, *m* is the attribute numbers the first *p* means numeric data, the rest means categorical data

K-prototypes Algorithm(cont.)

$$d_2(X,Y) = \sum_{j=1}^{p} (x_j - y_j)^2 + \gamma \sum_{j=p+1}^{m} \delta(x_j, y_j)$$

- The first term is the Euclidean distance measure on the numeric attributes and the second term is the simple matching dissimilarity measure on the categorical attributes
- The weight γ is used to avoid favoring either type of attribute

References of K-prototypes

- Zhexue Huang, **Clustering large datasets with mixed numerical and categorical values**. https://pdfs.semanticscholar.org/d42b/b5ad2d03be6d8fefa63d25d02c0711d19728.pdf
- Byoungwook Kim. **A Fast K-prototypes Algorithm Using Partial Distance Computation**. https://www.researchgate.net/publication/316348009_A_Fast_K-prototypes_Algorithm_Using_Partial_Distance_Computation