Introduction to Machine Learning

Work 1
Clustering exercise

Course 2020-2021

Contents

1 Description of the work

The aim of the exercise is to analyze different clustering algorithms using several data sets from the UCI repository. To this end, first of all you will implement several clustering algorithms using **Python 3.6 and PyCharm IDE**.

1.1 Methodology of the analysis

You will analyze the behavior of different clustering algorithms in well-known data sets from the UCI repository. These data sets are defined in **.arff** format. So, you will be able to analyze them with the Weka environment, too. A guide can be found at:

https://www.cs.auckland.ac.nz/courses/compsci367s1c/tutorials/IntroductionToWeka.pdf
http://www.cs.utexas.edu/users/ml/tutorials/Weka-tut/

The Weka is used to analyze if your code in Python is correct or not.

This work is divided in six tasks:

1. Implement your code for reading the arff file in Python and store the information in memory. Be careful, some of the data sets contain numerical and categorical data and may also contain missing values. For this exercise it is not necessary to store the class of the data set. However, for future assignments you will be requested to use the class. In the following address:

   https://docs.scipy.org/doc/scipy-0.19.1/reference/generated/scipy.io.arff.loadarff.html

   You will find a code for reading and writing arff files in Python. You can implement your own parser (do not use any other library) or you can analyze the code based on `scipy.io` library in the abovementioned link, execute it, and modify it accordingly to your needs. If you need it, for the numerical datasets, you can transform the continuous variables/attributes into categorical variables using function `pandas.`**`qcut`**`.`

2. Implement in a Python file the code that uses the **DBSCAN** from the `sklearn` library and apply it to the data of the file. Use *Euclidean*, *Cosine* or l1 or l2, and Manhattan similarity functions and evaluate what happen when you set up the algorithm with *auto*, *ball_tree, kd_tree* or *brute*.

3. Implement **your own K-Means** (KM) and **your own Bisecting k-Means** (BKM) algorithms and apply it to the data of the file. **Note that you are not allowed to use** `sklearn` library or any other library.

4. Choose one of the following algorithms and **implement your own code**: K-Medians, K-Means++ or K-harmonic means (KHM). **Note that you are not allowed to use** `sklearn` or any other library that implements these algorithms. Next, apply it to different data sets.

5. Implement **your own code** in one of the fuzzy clustering algorithms: Fuzzy C-Means (FCM) (Bezdek, 1981) or Possibilistic C-Means (PCM) (Krishnapuram - Keller, 1993).

6. Analyze the algorithms in three data sets (see Section 2). The restrictions are: two should contain all the attributes categorical or mixed attributes, and one with all the attributes numerical attributes. At least two of them should be large enough to be able to extract conclusions.

1.2 Work to deliver

In this work, you will use and analyze DBSCAN algorithm and you will implement and analyze K-Means, Bisecting k-Means, another algorithm of your choice (K-Medians or K-Means++ or k-harmonic means), and a Fuzzy Clustering algorithm. You may select 3 data sets for your analysis. At the end, you will find a list of the data sets available (see Section 2). The data sets are in a zip file in Racó at UPC and in campus virtual at UB (`campusvirtual.ub.edu`).

The idea is that you implement **your own code in Python 3.6** and you will use it to produce the results of the analysis. The development will be in the **PyCharm IDE** and you will deliver the project with your code in campus virtual at UB.

Once you have obtained the results, you will show them in several ways:

1. Compare using tables and/or graphs the clustering algorithms using some clustering validation metrics. Some examples are: **Adjusted Rand Index**, **Purity**, **Davies–Bouldin index**, **F-measure**. You can use these ones or other ones from the literature that best suit your evaluation. For the evaluation metrics, you can use the ones defined in `sklearn` library.
2. The results obtained can be compared to the true values. To show the results, you can use a confusion matrix, for example.

From the tables and graphs, you will reason and extract conclusions about the results obtained. For example, some questions that may help you to comment your results:

- Which information can be obtained for each data set using each algorithm? Is it the same or not?
- Which clustering algorithm do you consider is the best one for datasets with categorical data, with numerical data and with mixed data?
- Did you find differences among algorithms? According to the data sets chosen, which algorithm gives you more advice for knowing the underlying information in the data set?
- Can you explain the setup that you have used for each algorithm?
- In the case of the K-Means and the other algorithms where you have to choose the K, which has been the best K value? Have you implemented any improvement on the basic algorithms? For example, you can introduce/use a performance measure, like Silhouette, to decide which the best K value is. Another example, in K-Means++ it is defined an algorithm

for choosing the initial values (or "seeds") for the k-means clustering algorithm. Alternatively, you can use G-Means that is an algorithm for determining automatically the best K value.

- In the case of Fuzzy Clustering algorithm, you can optimize the C value. Have you done the optimization? Which are the results? In case that you have not included the optimization, how many C- values have you tested for each data set? And which value do you consider it is the best one?

You should deliver a report as well as the code in Python 3.6 and PyCharm project in Campus virtual in a zip file by November 1st, 2020. Please, the name of the zip file should contain the name and surname of each member of the group, and the number of your work. For example, considering that this assignment is Work 1 (we will use the acronym w1) and the students are Bart and Lisa Simpson, the name of the file will be: *BartSimpsonLisaSimpson_w1.zip*

The maximum extension allowed for the report is 10 pages in a two-column report or 20 pages in a one-column report.

The report will contain:

- Details about the implementation of your algorithms, including the decisions made during the implementation and the setup of the different parameters.
- The evaluation of the algorithms, including tables and/or graphs that show your results with comments about them.
- Justify your results and, in addition, reason each one of the questions defined above in your evaluation. Moreover, add any comment or observation that you consider important from your results.
- **Additionally, you should explain how to execute your code.**

According to the Merriam-Webster online dictionary, to "plagiarize" means:

- to steal and pass off (the ideas or words of another) as one's own
- to use (another's production) without crediting the source
- to commit literary theft
- to present as new and original an idea or product derived from an existing source

In other words, **Plagiarism** is an act of fraud. It involves both stealing someone else's work and lying about it afterwards. **A copy of the practical implies a mark of 0 for both the group that makes the copy and the group who passed the code (if it exists).**

2 Data sets

Below, you will find a table that shows in detail the data sets that you can use in this work. All these data sets are obtained from the UCI machine learning repository. First column describes the name of the domain or data set. Next columns show #Cases = Number of cases or instances in the data set, #Num. = Number of numeric attributes, #Nom = Number of nominal attributes, #Cla. = Number of classes, Dev.Cla. = Deviation of class distribution, Maj.Cla. = Percentage of instances belonging to the majority class, Min.Cla. = Percentage of instances belonging to the minority class, MV = Percentage of values with missing values (it means the percentage of unknown values in the data set). When the columns contain a '-', it means a 0. For example, the Glass data set contains 0 nominal attributes and it is complete as it does not contain missing values.

	Domain	#Cases	#Num.	#Nom.	#Cla.	Dev.Cla.	Maj.Cla.	Min.Cla.	MV
	Adult	48,842	6	8	2	26.07%	76.07%	23.93%	0.95%
	Audiology	226	-	69	24	6.43%	25.22%	0.44%	2.00%
	Autos	205	15	10	6	10.25%	32.68%	1.46%	1.15%
*	*Balance scale*	625	4	-	3	18.03%	46.08%	7.84%	-
*	*Breast cancer Wisconsin*	699	9	-	2	20.28%	70.28%	29.72%	0.25%
*	*Bupa*	345	6	-	2	7.97%	57.97%	42.03%	-
*	*cmc*	1,473	2	7	3	8.26%	42.70%	22.61%	-
	Horse-Colic	368	7	15	2	13.04%	63.04%	36.96%	23.80%
*	*Connect-4*	67,557	-	42	3	23.79%	65.83%	9.55%	-
	Credit-A	690	6	9	2	5.51%	55.51%	44.49%	0.65%
*	*Glass*	214	9	-	2	12.69%	35.51%	4.21%	-
*	*TAO-Grid*	1,888	2	-	2	0.00%	50.00%	50.00%	-
	Heart-C	303	6	7	5	4.46%	54.46%	45.54%	0.17%
	Heart-H	294	6	7	5	13.95%	63.95%	36.05%	20.46%
*	*Heart-Statlog*	270	13	-	2	5.56%	55.56%	44.44%	-
	Hepatitis	155	6	13	2	29.35%	79.35%	20.65%	6.01%
	Hypothyroid	3,772	7	22	4	38.89%	92.29%	0.05%	5.54%
*	*Ionosphere*	351	34	-	2	14.10%	64.10%	35.90%	-
*	*Iris*	150	4	-	3	-	33.33%	33.33%	-
*	*Kropt*	28,056	-	6	18	5.21%	16.23%	0.10%	-
*	*Kr-vs-kp*	3,196	-	36	2	2.22%	52.22%	47.78%	-
	Labor	57	8	8	2	14.91%	64.91%	35.09%	55.48%
*	*Lymph*	148	3	15	4	23.47%	54.73%	1.35%	-
	Mushroom	8,124	-	22	2	1.80%	51.80%	48.20%	1.38%
*	*Mx*	2,048	-	11	2	0.00%	50.00%	50.00%	-
*	*Nursery*	12,960	-	8	5	15.33%	33.33%	0.02%	-
*	*Pen-based*	10,992	16	-	10	0.40%	10.41%	9.60%	-
*	*Pima-Diabetes*	768	8	-	2	15.10%	65.10%	34.90%	-
*	*SatImage*	6,435	36	-	6	6.19%	23.82%	9.73%	-
*	*Segment*	2,310	19	-	7	0.00%	14.29%	14.29%	-
	Sick	3,772	7	22	2	43.88%	93.88%	6.12%	5.54%
*	*Sonar*	208	60	-	2	3.37%	53.37%	46.63%	-
	Soybean	683	-	35	19	4.31%	13.47%	1.17%	9.78%
*	*Splice*	3,190	-	60	3	13.12%	51.88%	24.04%	-
*	*Vehicle*	946	18	-	4	0.89%	25.77%	23.52%	-
	Vote	435	-	16	2	11.38%	61.38%	38.62%	5.63%
*	*Vowel*	990	10	3	11	0.00%	9.09%	9.09%	-
*	*Waveform*	5,000	40	-	3	0.36%	33.84%	33.06%	-
*	*Wine*	178	13	-	3	5.28%	39.89%	26.97%	-
*	*Zoo*	101	1	16	7	11.82%	40.59%	3.96%	-