

The solution must be produced as a Python Notebook.

The notebook must include appropriate comments and must operate as follows:

1. Read a transactions database in sparse format from a txt file into a list of lists. The file contains one transaction per line, each line has a variable number of items, separated by a separator. The external list will have a list for each transaction, each transaction is a list of items. Show the first two transactions (3 points)
2. Encode the list of lists into a binary representation and transform it into a dataframe whose columns are the items show the head of that dataframe
3. Find a value of min\_support such that the apriori algorithm generates at least 8 frequent itemsets with at least 2 items. Output the result with the message below (5 points)
4. Find the minimum metric threshold such that at least 10 association rules are extracted from the frequent itemsets found (5 points)
5. Print the first 10 rules found, sorted by descending confidence and support (3 points)
6. Plot confidence and support for all the sorted rules found (3 points)
7. Scatter plot the rules by confidence and support, labelling the points with the index value of the corresponding rule (3 points)

Quality of the code (6pt):

1. The python cells must be preceded by appropriate comments
2. Useless cells and pieces of code will be penalised
3. Naming style of variables must be uniform and in English
4. Bad indentation and messy code will be penalised

Additional directions, the assignments not compliant with the rules below will not be considered.

1. The notebook name must be **machineNumber\_lastname\_firstname.ipynb**, the number must have three digits, with leading zeroes, if necessary
  - a. for example, if I am sitting on the machine lab2, my notebook will be 002\_sartori\_claudio.ipynb
2. The first cell must contain the machine number, the last name and first name of the student.
3. The solution must directly access the data in the same folder of the notebook

Cooperative work will be **heavily sanctioned**.

The candidate can freely access the manuals available online in:

scikit-learn.org  
docs.scipy.org  
pandas.pydata.org  
matplotlib.org  
seaborn.pydata.org

The candidate can freely access the teaching materials available in the course website, including the available examples of python notebooks.

The notebook must be uploaded in both original and pdf form, as two separate files.