

<ul style="list-style-type: none"> • Registration in QIS (only ONCE for the graded alternative examination): every participant has to register in QIS FROM 27th of April UNTIL 15th June • Exercises (presented by students): 27th April until 15th June • Project presentations (presented by students): 15th June until 13th July • Upload Project report: until 22nd June • Upload implementation in github: until 22nd June • Upload Project slides: until 13th July • Upload Badges from cognitiveclass.ai: until 13th July 	A Latex template for the project report is available in Moodle <ul style="list-style-type: none"> • 1 Problem Description • 2 Description of Specific Methods and Algorithms • 3 Data Set Description • 4 Description of Python libraries used • 5 Description of Evaluation Module • 6 Web Frontend and User Manual • 7 Conclusion
--	--

TOPIC T1 RULE MINING

- 1) Briefly describe the problem of rule mining in your own words; if you are a team, clearly mark which participant wrote which part of the text.

- 2) Describe the a priori algorithm in your own words. Describe another itemset mining algorithm (like ECLAT, FPGrowth). [Note: if you are a team of n participants, choose and describe n algorithms; clearly mark which participant wrote which part of the text/described which algorithm]

- 3) From a public data repository, choose (at least) two data sets that you can apply itemset mining on (e.g. https://en.wikipedia.org/wiki/List_of_datasets_for_machine-learning_research, <http://archive.ics.uci.edu/ml/index.php>, <http://www.timeseriesclassification.com/dataset.php>, <http://kdd.ics.uci.edu/summary.data.application.html>, <http://fimi.uantwerpen.be/data/>) and write a brief description of each of the datasets. [Note: if you are a team of n participants, choose and describe n data sets; clearly mark which participant wrote which part of the text/described which data set]

- 4) Choose public implementations (in Python) of A priori and the algorithm(s) chosen under 2), and apply them on the datasets chosen under 3) such that the support threshold can be flexibly set as a parameter. Describe your implementation; if you are a team, clearly mark which participant wrote which part of the text.

- 5) Implement an evaluation module that compares the outcomes of the 2 (or n) chosen algorithms with respect to average support, confidence, lift and conviction of the obtained rules (again, the support threshold can be flexibly set as a parameter). Describe your implementation; if you are a team, clearly mark which participant wrote which part of the text.

- 6) Implement a simple web frontend (e.g. using streamlit or svelte) to select the data sets, visualize the data sets, set parameters (like support threshold) and display the obtained rules (from 4)) as well as the comparison (from 5)) of the two (or n) algorithms. Describe the implementation and write a brief user manual with screenshots; if you are a team, clearly mark which participant wrote which part of the text.