Data Mining Project

(guided self study)

13.3.17

Simo Linkola
slinkola@cs.helsinki.fi

Motivation and Goal

- How DM can be applied for real world problems?
- Deep understanding of an algorithm by implementing it yourself
- Hands on experience using DM for real data
  - Real world data is usually noisy
  - What kind of preprocessing is needed?
- Analysis of the results
  - How the results should be understood?
  - What is interesting? Why?

Working Style and Grading

- 2 credits (or more if asked)
- Individual or group (2-4 persons) work
  - People looking for a group can stay after the session to form the groups
- Grading: Fail (fail) / Pass (adequate to good) / 5 (excellent)
- Grading criteria:
  - Correctness of the implementation
  - Implementation performance (speed, memory usage)
  - Cleanness of the code (incl. commented code)
  - The depth of analysis (ability to relate the findings to the big picture)
  - Quality of the presentation
  - **Stress on analysis and the algorithm implementation!**

Student’s Responsibilities

- **Select** your topic: DM task/algorithm and the data to be used
- **Implement** the data mining algorithm
- **Apply** your algorithm to your data
- **Analyse** your results
- **Write** a short report about your implementation and analysis
- **Return** your code and the report
- **Present** your work

TA's Responsibilities

- Supervise students on the need basis
- You can reserve a meeting with TA at any time (slinkola@cs.helsinki.fi)
- TA is available for questions weekly in B233: Wed 13-15
  - You can also try your luck and drop by B233 at any time
- Answer to questions in Moodle

Timeline

- Finding a team (or deciding to work alone)
- Selecting a topic
- Working on the topic to decide whether it is feasible to do in a few credits
- **DL 31.3.:** Submit the topic in Moodle
- Working on the topic (reserve guidance from slinkola@cs.helsinki.fi)
- Presenting your work at the start of May
- **DL 5.5. 23.59**: Submitting the source code and the report on the project
- Finish

The course has a Moodle page, where *all* submissions are done.
Enrolment key: dmp2017

Algorithm Implementation

- Should not require any proprietary software to run
- Can be written on any all-round programming language
  - Java, Python, C, C++, etc.
  - **No**: SQL, R, Matlab
- Should have minimum constraints on the datasets
- Write the code yourself
  - Use only data structures and library functions available in standard libraries
  - No copy-paste “implementation”
- After basic implementation, try to optimise speed and memory usage

Report

- Describe your goal
- What patterns are you looking for and which algorithm are you using
  - Itemsets, sequences, etc.
  - Describe your algorithm with pseudocode
- Implementation details: preprocessing of the data, etc.
- Running instructions for the code
- **Analysis of your results!**
- Any obvious future directions for the work, its strengths and flaws
- Time allocations for each group member and a short description of their work

Submissions

- The report and the source code are submitted in the Moodle
- **A single compressed file**
- Do not submit your dataset, but give a pointer to it in the report if it is easily downloadable

Topics

- All algorithms and tasks introduced during the DM course
- Frequent itemsets: FP-Growth, Apriori, Depth-first methods (e.g. Eclat)
- Association rules
- Sequence mining (e.g. text)
- Graph-mining (frequent subgraphs, etc.)
- Etc.

Requirements for the Dataset

- Not too small
  - Your implementation should handle reasonably large datasets
- No preprocessed datasets which have no meta-information
  - You should be doing the preprocessing of the data yourself
  - You probably need meta-information for meaningful analysis
- Finding a good dataset can be hard
  - You might need to gather it yourself
  - Some pointers to existing datasets on the course's cs-page

Thanks!

Do not hesitate to contact the TA on any issue regarding the course.

Remember that **the deadline for choosing your topic is 31.3.!**