Project 1 of CX4032

Your task includes the following:

1) Read and understand the paper: Bing Liu, Wynne Hsu, Yiming Ma:Integrating Classification and Association Rule Mining. KDD 1998: 80-86

2) Implement an algorithm of Classification based on Association rules. You can implement any algorithm in the paper, or a variant of these algorithms.

Your coding should have two parts:

- Mining Class association rules
- Building a classifier

You can refer to the online code, and reuse part of the code. But you must understand the code. If you reuse some online coding, you need to detail what code you use in your report.

One reference code:
https://cgi.csc.liv.ac.uk/~frans/KDD/Software/CBA/cba.html

3) Use 5 datasets used in the KDD'98 paper with the same setting to conduct classification tasks using the code you develop in Part 2. Report the classification results of your software in Part 2. You can choose any 5 datasets. These datasets can be obtained from the UCI machine learning portal (https://archive.ics.uci.edu/ml/datasets.php).
In addition, please choose another 2 datasets from the portal or other datasets you are interested (e.g., data in Kaggle) to report results. To make it easier, you are suggested to choose datasets with 2 categories.

4) Find open softwares for other classification methods: Decision trees, random forest, and SVM. Compare with them in terms of classification accuracy. You can use measures such as F-score.

5) Advanced part: Design an algorithm to improve the accuracy of the code that you develop in part 2. You can use the methods in the following papers. You can also design other improvement.

- Wenmin Li, Jiawei Han, Jian Pei. CMAR: accurate and efficient classification based on multiple class-association rules. Proceedings 2001 IEEE International Conference on Data Mining. **(relatively easier)**
- Gao Cong, Kian-Lee Tan, Anthony K. H. Tung, Xin Xu: Mining Top-k Covering Rule Groups for Gene Expression Data. SIGMOD Conference 2005: 670-681 **(more challenging. The rule mining algorithms cannot be used. And you need to design a new way to mine top-k covering rule group)**

Weight of grading:

- Part 2: 40%
- Part 3: 20%
- Part 4: 15%
- Part 5: 25%

You are expected to improve your problem solving, deep thinking, and self-learning ability through the project, which are very important skills to acquire in universities.

What to deliver:

Hardcopy: The final report is up to 8 A4 pages (not necessary to write 8 pages). The report should include the results of Part 2—Part 5 of the project

Softcopy: Your report, and your source code, which will be submitted through course website.

Live demo during presentation

Project are done in groups. Discussions with other students are allowed, but each group has to write your own code.

Demo + Presentation: will happen in week 12 and week 13. Detailed schedule will be announced.

Code submission + report: Due in the end of week 9 (Sunday, 17 Oct). You can submit the hardcopy of report to Software project lab on Monday of Week 10 (18 Oct.).

Grading will consider report + live demo + code + presentation

NOTE:

1. **MOSS**: Sharing code with your classmates is not acceptable!!! All programs will be screened using the Moss (Measure of Software Similarity.) system.
2. **You are not allowed to share your project code on the web publicly**.

TA for projects:

- Liu Shuai (shuai004@e.ntu.edu.sg)
- Zhao Yue (zhao0342@e.ntu.edu.sg)
- Shi Jiachen (JIACHEN001@e.ntu.edu.sg)
- Chen Yile (yile001@e.ntu.edu.sg)

If you have questions, please email all the TAs above and cc to me. TAs can only provide some consultation for projects, but you should NOT expect TAs to help to do any part of your project.