Assigned: 4/5/2017
Due: Fri. 4/21/2017, 11:59pm

Instructions: This project will cover some questions related to topics of text mining, recommender systems, association analysis and final topics of the course.

Submission Requirements: Your answers must be computer generated (including text and diagrams). Your final document submission should include text responses to questions and description of your efforts, tables, R/Matlab/Python code used to calculate answers, and figures. As well as the code to carry out the work.

Formatting of submissions: The following methods are acceptable ways to submit your assignment:

- Word + code, Open Office + code
  This option may require taking screenshots or printing figures created in R/MATLAB/Python and importing them into the word processing software. Additional code and results should also be inserted into the word documents.
- If you are using R consider:
  - Rmd + PDF, Rmd + HTML
    Use `knitr` or `rmarkdown` to collect all text responses, figures, tables, and code in the Rmarkdown file and process it to produce a PDF or HTML file.
  - Snw + PDF, Stex + PDF
    Use `Sweave` to collect all text responses, figures, tables, and code in the Snw file and process it to produce a PDF.
- If you are using MATLAB consider:
  - .m file + markup, publishing matlab code → HTML
    Incorporate your answers directly into your MATLAB code (code, comments, results), publish the code creating an HTML file.
  - .m files + *Your favorite document editor*
    Answer your questions in your text editor, embedding code and results from matlab .m file
- If you are using Python consider:
  - iPython → HTML
  - LaTeX + Sphinx → PDF / HTML

I highly recommend following the ideas of reproducible research and embed the code, images, and results directly into the text using packages like `knitr/rmarkdown`, `Sweave`, `publish`, or `iPython` (other packages follow this practice using Latex and reStructureText as well and are open for you to use).

If you want to follow the style of the R introduction documents on Canvas (e.g., introA.html, introB.html, etc.), please use the provided CSS style file `min.css`), and follow instructions provided in R Studio documentation and the `rmarkdown` package. There are also a number of style and code highlighting styles available using `Bootstrap` themes.

Name your main submission files as *P5_GroupName*, create a zip-file called *Project5_GroupName.zip* and submit on Canvas. For example, if I was using R, and part of "GroupName", I would submit either:

- *P5_GroupName.Rmd*, *P5_GroupName.pdf*, or
- *P5_GroupName.Rmd*, *P5_GroupName.html*, or
- *P5_GroupName.Snw*, *P5_GroupName.pdf*

along with any other supplemental `.R` files I created in *Project5_GroupName.zip*.

You must sign up for the P5 groups on Canvas!

Questions:

Part 1: Answer by Hand
For the first four problems you do not need to use R/Matlab/Python, rather this is showing you understand definitions, concepts, and how the algorithms work.

1. Text Mining
   Consider the following documents:

   Doc 1: cat, cat, bat, rat, fat, cat
   Doc 2: mat, pat, bat, bat, bat, rat
   Doc 3: fat, rat, mat, pat, sat, cat

   (a) (4 points) Construct the Term Document Matrix
   (b) (6 points) Construct the TF-IDF Matrix
   (c) (2 points) What is the term-document pair(s) with the highest TF-IDF value.

2. Recommendation Systems
   Consider the following ratings matrix:

	Avengers	Iron Man 3	The Dark Knight Rises	Captain America 2
Ann	3.5	3	5	3.5
Ben	5	2	3	3
Chris	4	2.5	4	4
Dana	??	3	4.5	2.5

   (a) (6 points) Fill in the missing rating for Dana using user-based collaborative filtering with $k = 2$ nearest neighbors using Manhattan distance to compare users. Indicate which other users are closest to Dana and the resulting rating calculated (use simple averaging of nearest neighbors).
   (b) (6 points) Fill in the missing rating using item-based collaborative filtering with $k = 2$ nearest neighbors using Manhattan distance to compare items. Indicate which other items are closest to Avengers and the resulting rating calculated (use simple averaging of nearest neighbors).
   (c) (4 points (bonus)) Calculate the missing rating same as above, but incorporate the similarity measure (distance measure) into the prediction. Because Manhattan distance was used rather than similarity convert it to a similarity with the following process normalize the distance between min-distance = 0 and max-distance = 3*(5-1) = 12; then subtract this value from 1 to get a similarity score.

3. Association Analysis I
   Given a database of transactions and a min-support = 2,

Trans.	Items
T1	I1, I2, I3, I4, I5, I6, I7, I8, I9, I10
T2	I1, I2, I3, I4, I5, I6, I7, I8
T3	I1, I2, I3, I4, I5
T4	I6, I7, I8
T5	I100, I101, I102, I103

   (a) (1 point) How many frequent patterns exist?

   (b) (3 points) What is the set of frequent closed patterns?

   (c) (2 points) What is the set of frequent max-patterns?

   (d) (4 points) Find an example of an association rule that matches the following pattern with min-support = 2 and min-conf = 70

$$(I1, I2, I3, I4, IX \rightarrow IY)$$

   (e) (8 points) For the association rule $I1 -> I6$, compute the support, confidence, lift, and interest

4. Data Mining Book: 6.6
   For each algorithm, show the major steps through the algorithm. Also, report at the end the frequent items sets identified.

   (a) (24 points) For Apriori, present L_i and C_i for each level i considered

   (b) (22 points) For FP-growth, present the FP-tree like Fig. 6.7 (p. 258) and present the generated the frequent patterns like Table 6.2 (p. 259) (the conditional FP-trees do not need to be illustrated explicitly).

Part 2: Use Software

For the first four problems you do not need to use R/Matlab/Python, rather this is showing you understand definitions, concepts, and how the algorithms work.

5. (12 points) Confirm the results above of the Apriori algorithm. For R, the `arules` package is available or Matlab has the Association Rules package available from File Exchange[1].

[1] http://www.mathworks.com/matlabcentral/fileexchange/42541-association-rules