67978: A Needle in a Data Haystack
Introduction to Data Science
Homework #1: Frequent Itemset, Recommender Systems
Due: **29 Nov, 11:59pm, on Moodle**

THE HEBREW UNIVERSITY OF JERUSALEM

Problem 1 (Frequent Itemsets, Association Rules).

(a) What is the confidence for the rules $\emptyset \to A$ and $A \to \emptyset$?

(b) Suppose the confidence of the rules $A \to B$ and $B \to C$ are larger than some threshold, **minconf**. Is it possible that $A \to C$ has a confidence less than **minconf**?

(c) Consider Table 1. Suppose the sup threshold = 30% and conf threshold = 50%. Find all *frequent itemsets* using Apriori. Show the candidate and frequent itemsets for each iteration.

(d) Consider Slide 65, Class 01. Write down the pseudo-code for finding all rules above the confidence threshold from a frequent itemset X using the observation.

TID	Itemset
1	$\{a, b, c\}$
2	$\{a, b, c, d\}$
3	$\{a, c, d, e\}$
4	$\{b, c, d\}$
5	$\{b, c, d, e\}$
6	$\{b, d\}$
7	$\{c, d, e\}$
8	$\{a, c, e\}$
9	$\{a, d, e\}$
10	$\{b, d\}$

Table 1: Transactions

Problem 2 (Strange Baskets).

Imagine there are 100 baskets, numbered 1,2,...,100, and 100 items (also numbered 1,2,...,100). Item i is in basket j if and only if i divides j with no remainder. For example, basket 24 is the set of items {1,2,3,4,6,8,12,24}.

(a) Describe all the association rules that have 100% confidence. Give an example.

(b) If the support threshold is 5, which items are frequent? Which *pairs* of items are frequent?

(c) Association rule mining often generates a large number of rules. To reduce the number of rules we can post-process them and only output maximal frequent itemsets. If we are given a support threshold s, then we say an itemset is maximal if no superset is frequent. If the support threshold is 5, find the maximal frequent itemsets.

Problem 3 (Recommender Systems). Consider the following utility matrix, representing the ratings, on a 1-5 star scale, of eight items, a through h, by three users A, B, and C. Compute the following from the data of this matrix.

	a	b	c	d	e	f	g	h
A	4	5		5	1		3	2
B		3	4	3	1	2	3	
C	2		1	3		4	5	

(a) Treating the utility matrix as boolean, compute the Jaccard similarity between each pair of users.

(b) Repeat Part (a), but use cosine similarity.

(c) Repeat Part (a), but use centered cosine similarity.

(d) Which of your answers would change if we add another item, i, that nobody has rated? Explain.

Problem 4 (Creepy Crawlers).

In this question, you will use crawling to create a (tiny) dataset.
Crawling websites is impolite in general, so MAKE SURE you are following these rules:

- Start by downloading a single page to your computer. Only when you have it right (that is, you can open the local file, parse the html and save the output in the format you want), you can add a loop in and go for multiple pages.
- Make sure there is a significant delay between each request to the website (several seconds should do the trick).
- Be **really** careful. :)

Instructions: Pick a website. Should be either a store or a forum.
Your goal is to crawl products (if a store) or threads (if a forum). For the sake of this homework, we only need **50 pages**, but your code should be able to crawl the entire site, if we remove that restriction. (In other words, do not hand-code 50 URLs you want to crawl)
You should submit three things:

(a) The code.

(b) Output should be **a JSON file** (indented for readability, please). Each entry should contain the URL of the crawled page and at least four other meaningful fields. If a product page, you could include Title, Price, Description, Category (if applicable), Image URL, and so on (extra credit: include reviews). For a discussion thread, you could include Thread Title, Name of person who started the thread (OP), Time-stamp, Content (extra credit: include all messages in the thread).

(c) In addition, attach **one example**: a screenshot of one of the pages you crawled, plus the corresponding JSON.

You can use existing code for a web crawler/scraper in your favorite language (or write your own, if you so choose). A part of the assignment is experimenting with tools, so I will not recommend specific ones. I will, however, recommend considering some tool for extracting content from html, if html is too convoluted (e.g., Beautiful Soup, lxml).
Have fun. :)

P.S. Want to help research? Consider these websites (could be more difficult, and you might need to create an account): `kickstarter.com` (treat a project as a product), `www.innocentive.com/ar/challenge/browse` , `quirky.com` , `challenges.openideo.com/challenge` , or anything relevant from this page: `http://www.boardofinnovation.com/list-open-innovation-crowdsourcing-examples/`

Problem 5 (Meta). How long (in hours) did this assignment take?

Problem 6 (Project). List the three people on your team, and three ideas for a project (only one person on the team needs to submit the ideas; the others should just write down their team members). What problem will you be solving? What data will you use (and how will you get it)? What will your output look like? For your reference, you will eventually be graded on:

(a) Overall ambition, creativity, difficulty of project (20%)

(b) Execution and implementation (45%)

(c) Use of appropriate and compelling visualizations (15%)

(d) Written part (20%)