# CSE427 – Homework 9

M. Neumann

Due THU 04/14/2016 10am

## Getting Started

Update your SVN repository. Find instructions on how to checkout, update, and commit to your SVN repository here: `http://sites.wustl.edu/neumann/resources/cse427s_resources/`

When needed, you will find additional materials for *homework x* in the folder `hwx`. So, for the current assignment the folder is `hw9`.

## Indicating Group Work

Use the file `partners.txt` to indicate group work. Follow these instructions exactly, to ensure to get credit!

- `partners.txt` needs to include up to 2 wustlkeys in the first two lines (one line per wustlkey)
- **first line/wustlkey is the repository, where the solution is located**. We will **only** consider the submission in this repository!
- Every student in a group needs to have **the same** `partners.txt` in the hwx folder in their repository (indicating that the partnership are **mutually accepted**)!
- If you do not have a partner, try to find one. If you want to submit on your own, indicate your wusltkey in the first line of `partners.txt` and leave the second line blank.

## Usage Agreement

By using the dataset from the Netflix Prize you agree as follows:

- I agree to the terms specified in Netflix Prize Rules (cf. README file provided in your SVN repo).
- I agree to **delete** this dataset once the project has been completed.
- I will not redistribute this data in any form.

## Problem 1: Top-15-List of Most Popular Movies (70%)

**If you agree to the usage agreement above,[1] download the data from:**

```
https://classes.cec.wustl.edu/cse427/netflix_subset.zip
```

**Do NOT add this data to your SVN repositories!!!!!!!!**

This data is a subset of the training data from the Netflix Prize. The Netflix Prize aimed at substantially improving the accuracy of predictions about how much someone is going to enjoy a movie based on their movie preferences. It was issued by the Netflix company and on September 21, 2009 a $1mio Grand Prize was awarded to the winning team.[2]

The format of this dataset is slightly different than the one used in the original Netflix challenge. It is described in the `description.txt` file.

(a) Write a MapReduce program to compute the $N$ most popular movies in the `TrainingRatings.txt` file across all users using the MapReduce algorithm discussed in-class. $N$ should be a parameter, that you can provide via the command line. You may use Mahmoud Parsian's implementation to get started: `https://github.com/mahmoudparsian/data-algorithms-book/tree/master/src/main/java/org/dataalgorithms/chap03/mapreduce`.

Use the **sum of the ratings** as measure for popularity! Note, that the implementation linked above takes as input a pair of two comma separated values; our data has three: `movieID,userID,rating`; simply ignore the userID. You do not have to use SequenceFiles (ignore `SequenceFileWriterForTopN.java`); so, adapt the input format accordingly. You will also need to add a look-up of the movie titles using the `movie_titles.txt` file.

Here is some test input and output for a top-3-list:

Input:

```
8,1148143,2.0
8,1174811,5.0
9,63493,5.0
9,516722,4.0
1,1232582,2.0
5,1631874,4.0
5,721546,4.0
8,2035299,3.0
5,826193,5.0
8,1793777,4.0
3,125713,3.0
```

Output:

[1] If not, you will need to talk to me.

[2] You can read more about it here: `http://www.netflixprize.com/`.

```
15 What the #$*! Do We Know!?
13 The Rise and Fall of ECW
9 Class of Nuke 'Em High 2
```

Use this for testing and debugging. Once, your implementation works, run it on the `TrainingRatings.txt` for $N = 15$.

(b) Analyze the data in `TrainingRatings.txt` according to the **average rating per user**. You can use PIG or MapReduce .

- Plot the distribution of average ratings. Save this figure as `hw9_p1b.png`.
- What fraction of users have a high (larger than 4) or low (smaller than 2) average indicating overly enthusiastic or overly pessimistic raters?
- Regardless of your findings in part (b), how *could* you adjust your top-$N$-list program (no implementation required!!) to cope with overly enthusiastic and overly pessimistic raters?

Submit the top-15-list and your answers to (b) by editing the `hw9.txt` file and add your plot (`hw9_p1b.png`) and `.java` classes to the `hw9` folder in your SVN repository.
**Add the files to your SVN repo before committing:**

```
$ svn add hw9_p1b.png
$ svn add *.java
$ svn commit -m 'hw9 submission' .
```

## Problem 2: Collaborative Filtering - Similarity Measures (30%)

(a) Show (formally) that the normalized cosine similarity measure corresponds to the Pearson correlation.

(b) **Quality vs. implementation effort and efficiency**

- From a quality perspective, what is the benefit of using the normalization (i.e., Pearson correlation instead of cosine similarity)?
- From an implementation perspective, what is the disadvantage of using the normalization (i.e., Pearson correlation instead of cosine similarity)?

(c) What is the problem of the Jaccard similarity measure? Can you think of a way to pre-process the rating data to overcome this problem?

Submit your answer for (a) as `hw9_p2a.pdf` (**PDF format**) and add it to the `hw9` folder in your SVN repository. Submit your answers for (b) and (c) by editing the `hw9.txt` file. You may submit your implementation (it will not be used for grading). **Add the `hw9_p2a.pdf` file to your SVN repo before committing:**

```
$ svn add hw9_p2a.pdf
$ svn commit -m 'hw9 submission' .
```

## Bonus Problem (5% up to a max. of 100%) - no group work!

Write a review for this homework and store it in the file `hw9_review.txt` provided in your SVN repository (and commit your changes). This file should only include the review, **no other information** such as name, wustlkey, etc. Remember that you are not graded for the content of your review, solely it's completion.

You can only earn bonus points if you write **at least 50 words**. Bonus points are given to the **owner of the repository only** (no group work!).

## Copyrights

The data used in this problem is taken from Pedro Domingos' class on Data Mining/Machine Learning at University of Washington, 2012.