E6893 Big Data Analytics:

Demo Session II: Mahout working with Eclipse and Maven for Collaborative Filtering

Aonan Zhang

Dept. of Electrical Engineering

October 9th, 2014

The Apache Mahout™ project's goal is to build a scalable machine learning library

Applicable Models

Latest release version 0.9 has

- User and Item based recommenders
- Matrix factorization based recommenders
- K-Means, Fuzzy K-Means clustering
- Latent Dirichlet Allocation
- Singular Value Decomposition
- Logistic regression classifier
- (Complementary) Naive Bayes classifier
- Random forest classifier
- High performance java collections
- A vibrant community

1. Download Eclipse

- https://www.eclipse.org/downloads/

2. Install Maven

- -Help -Install New Software
- -Add -Name: m2eclipse -Location: http://download.eclipse.org/technology/m2e/releases (Google "install m2eclipse")

1. Collaborative Filtering

Users

Items

1		2		4		5
	3	1		2		
5		3		4	3	
	1		1	2		3
	5		4			

1. Start a Maven project:

- -File -New -Other -Maven Project
- -maven-archetype-quickstart

2. Add Mahout dependency in pom.xml (your version might be 0.9)

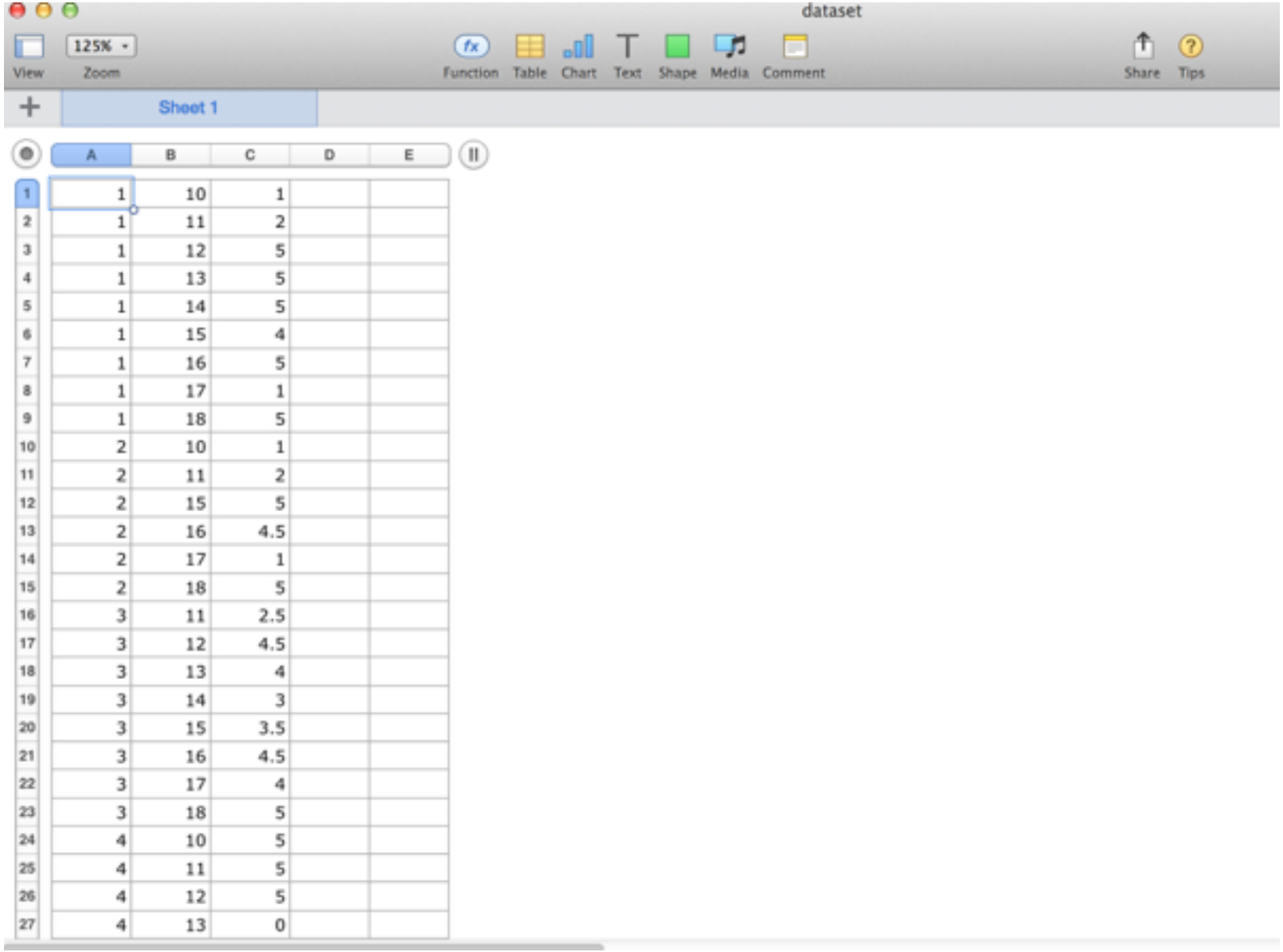
```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSch
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.prediction</groupId>
  <artifactId>RecommendApp</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>RecommendApp</name>
  <url>http://maven.apache.org</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.apache.mahout</groupId>
      <artifactId>mahout-core</artifactId>
      <version>0.7</version>
</dependency>

    <dependency>
```

3. Copy data file into the project

- Go to https://mahout.apache.org/users/recommender/userbased-5-minutes.html and download the data

- Create data/dataset.csv

Mahout Working with Eclipse and Maven



	A	B	C	D	E
1	1	10	1		
2	1	11	2		
3	1	12	5		
4	1	13	5		
5	1	14	5		
6	1	15	4		
7	1	16	5		
8	1	17	1		
9	1	18	5		
10	2	10	1		
11	2	11	2		
12	2	15	5		
13	2	16	4.5		
14	2	17	1		
15	2	18	5		
16	3	11	2.5		
17	3	12	4.5		
18	3	13	4		
19	3	14	3		
20	3	15	3.5		
21	3	16	4.5		
22	3	17	4		
23	3	18	5		
24	4	10	5		
25	4	11	5		
26	4	12	5		
27	4	13	0		

4. Create a recommender: edit App.java

```
22 public class App
23 {
24     public static void main( String[] args ) throws IOException, TasteException
25     {
26         DataModel model = new FileDataModel(new File("data/dataset.csv"));
27         UserSimilarity similarity = new PearsonCorrelationSimilarity(model);
28         UserNeighborhood neighborhood = new ThresholdUserNeighborhood(0.1, similarity, model);
29         UserBasedRecommender recommender = new GenericUserBasedRecommender(model, neighborhood, similarity);
30         List<RecommendedItem> recommendations = recommender.recommend(2, 3);
31         for (RecommendedItem recommendation : recommendations) {
32           System.out.println(recommendation);
33         }
34     }
35 }
```

5. Import packages

```
3 import java.io.File;
4 import java.io.IOException;
5 import java.util.List;
6
7 import org.apache.mahout.cf.taste.common.TasteException;
8 import org.apache.mahout.cf.taste.impl.model.file.FileDataModel;
9 import org.apache.mahout.cf.taste.impl.neighborhood.ThresholdUserNeighborhood;
10 import org.apache.mahout.cf.taste.impl.recommender.GenericUserBasedRecommender;
11 import org.apache.mahout.cf.taste.impl.similarity.PearsonCorrelationSimilarity;
12 import org.apache.mahout.cf.taste.model.DataModel;
13 import org.apache.mahout.cf.taste.neighborhood.UserNeighborhood;
14 import org.apache.mahout.cf.taste.recommender.RecommendedItem;
15 import org.apache.mahout.cf.taste.recommender.UserBasedRecommender;
16 import org.apache.mahout.cf.taste.similarity.UserSimilarity;
17
```

6. Run eclipse and finish!

```
RecommendedItem[item:12, value:4.8328104]
RecommendedItem[item:13, value:4.6656213]
RecommendedItem[item:14, value:4.331242]
```

- You may also want to evaluate the recommender. See https://mahout.apache.org/users/recommender/userbased-5-minutes.html for details

Apache Mahout
Home Page: https://mahout.apache.org/

Apache Mahout: What?

Scalable machine learning library.

Mahout and its associated frameworks are Java-based and therefore platform-independent, so you should be able to use it with any platform that can run a modern JVM.
Note that Mahout requires Java 6.
-Mahout in Action

Download Link: http://mahout.apache.org/general/downloads.html

Latest Release: 0.9 - mahout-distribution-0.9.tar.gz

MacOS:
brew install mahout

Say, we want to run collaborative filtering:

- Collaborative filtering—producing recommendations based on, and only based on, knowledge of users' relationships to items.
- These techniques require no knowledge of the properties of the items themselves. This is, in a way, an advantage.
- This recommender framework doesn't care whether the items are books, theme parks, flowers, or even other people, because nothing about their attributes enters into any of the input.

Example: Collaborative Filtering

Listing 2.1. Recommender input file, intro.csv

```
1,101,5.0
1,102,3.0
1,103,2.5

2,101,2.0
2,102,2.5
2,103,5.0
2,104,2.0

3,101,2.5
3,104,4.0
3,105,4.5
3,107,5.0

4,101,5.0
4,103,3.0
4,104,4.5
4,106,4.0

5,101,4.0
5,102,3.0
5,103,2.0
5,104,4.0
5,105,3.5
5,106,4.0
```

User 1 has preference 3.0 for item 102

User ID, item ID, preference value

To run locally:
export MAHOUT_LOCAL="any value"

bin/mahout recommenditembased -s SIMILARITY_LOGLIKELIHOOD -i /path/to/input/file -o /path/to/output/folder/ —numRecommendations 1

Note: The output folder should not exist.

Output:

```
1    [104:2.8088317]
2    [105:3.5743618]
3    [103:4.336442]
4    [105:3.6903737]
5    [107:3.663558]
```

The recommender engine recommended book 104 to user 1, book 105 to user and so on…

To run on hadoop:

unset MAHOUT_LOCAL

export HADOOP_CONF_DIR=/usr/local/hadoop/etc/hadoop/

MAHOUT_CONF_DIR=/Users/bhavdeepsethi/Downloads/mahout-distribution-0.9/conf
(if not installed from brew)

Running with Hadoop- Problem. 0.9 does not have support for Hadoop 2.x

Support will come in Mahout 1.0

https://issues.apache.org/jira/browse/MAHOUT-1329

So we have to build from source:

http://mahout.apache.org/developers/buildingmahout.html

Pre-requistie: Git and Maven

MacOS:
brew install git
brew install maven

Ubuntu:
sudo apt-get install git
sudo apt-get install maven

Steps:
git clone git://git.apache.org/mahout.git mahout-trunk

mvn clean package -Dhadoop2.version=2.5.1 -Dhbase.version=0.98.6.1-hadoop2 -DskipTests

Data Set
http://www.grouplens.org/system/files/ml-100k.zip

hdfs dfs -mkdir /input

hdfs dfs -put ~Downloads/ml-100k/u.data /input

hdfs dfs -ls /input/

Run:
hadoop jar /Users/bhavdeepsethi/CU/BigDataAnalytics/mahout-again/mahout-trunk/mrlegacy/target/mahout-mrlegacy-1.0-SNAPSHOT-job.jar org.apache.mahout.cf.taste.hadoop.item.RecommenderJob -s SIMILARITY_COOCCURRENCE --input /input/u.data --output outputNew

Apache Mahout: Example

Output:

```
1	[845:5.0,550:5.0,546:5.0,25:5.0,531:5.0,529:5.0,527:5.0,31:5.0,515:5.0,514:5.0]
2	[546:5.0,288:5.0,11:5.0,25:5.0,531:5.0,527:5.0,515:5.0,508:5.0,496:5.0,483:5.0]
3
[137:5.0,284:5.0,508:4.8327274,248:4.826923,285:4.80597,845:4.754717,124:4.7058825,319:4.703242,293:4
.6792455,591:4.6629214]
4	[748:5.0,1296:5.0,546:5.0,568:5.0,538:5.0,508:5.0,483:5.0,475:5.0,471:5.0,876:5.0]
5	[732:5.0,550:5.0,9:5.0,546:5.0,11:5.0,527:5.0,523:5.0,514:5.0,511:5.0,508:5.0]
6	[739:5.0,9:5.0,546:5.0,11:5.0,25:5.0,531:5.0,528:5.0,527:5.0,526:5.0,521:5.0]
7	[879:5.0,845:5.0,751:5.0,750:5.0,748:5.0,746:5.0,742:5.0,739:5.0,735:5.0,732:5.0]
```

Each line represents the recommendation for a user. The first number is the user id and the 10 number pairs represents a movie id and a score.
If we are looking at the first line for example, it means that for the user 1, the 10 best recommendations are for the movies 845, 550, 546, 25 ,531, 529, 527, 31, 515, 514.

Source:

Example:
kMeans Clustering:

Data:
Download: http://archive.ics.uci.edu/ml/databases/synthetic_control/synthetic_control.data

hdfs dfs -mkdir -p /user/<username/whoami>/testdata
hdfs dfs -put synthetic_control.data testdata/synthetic_control.data

hadoop jar examples/target/mahout-examples-1.0-SNAPSHOT-job.jar org.apache.mahout.clustering.syntheticcontrol.kmeans.Job

Any Questions?