Track 2 – Assignment 2011

<table>
<thead>
<tr>
<th>#User</th>
<th>#Items</th>
<th>#Ratings</th>
<th>#Train Ratings</th>
<th>#Test Ratings</th>
</tr>
</thead>
<tbody>
<tr>
<td>249.012</td>
<td>296.111</td>
<td>62.551.438</td>
<td>61.944.406</td>
<td>607.032</td>
</tr>
</tbody>
</table>

mit > 80 %

nicht gerated
ABSTRACT
KDD Cup 2007 focuses on predicting aspects of movie rating behavior. We present our prediction method for Task 1 “Who Rated What” in 2007 when the task is to predict which user rated which movie in 2007. We use the combination of the following predictors, listed in the order of their efficiency in the prediction:
- The predicted number of ratings for each movie based on time series prediction, also using movie and DVD release dates and movie score selections by the AFID distance of the titles.
- The predicted number of ratings by each user by using the fact that ratings were sampled proportional to the number of ratings by each user.

Categories and Subject Descriptors
C.2.4 [Computer Applications], Social and Behavioral Sciences: C.2.4.1 [Mathematics of Computing], Numerical Analysis: C.6.7 [Theory]: Computational Geometry and Object Modeling

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions Dept, ACM Inc., fax +1 (212) 869-0481.

1. INTRODUCTION
Recommender systems predict the preference of a user on a given item based on known ratings. In order to evaluate methods, in October 2006 Netflix provided movie ratings from anonymous customers on nearly 15,000 movies under license to researchers. The KDD Cup tasks related to this data set. For Task 1 “Who Rated What” the model has to predict which users rated which movie in 2007 while for Task 2 “How Many Ratings in dbbi” the task was to predict the number of additional ratings of movies.
In this paper we present our method for Task 1 “Who Rated What”. The task was to predict the probability that a user rated a movie in 2006 (with the actual date and rating being consistent) for a given list of 100,000 movies per user. The movie and user data derive from the Prime data set, i.e., the movies appeared (or at least received ratings) before 2006 and the users also gave their first rating before 2006 that some of the pairs were rated in the training set. We give a detailed description of the sampling process and 28 rules to get information that we use for the prediction.

1.1. Method prediction

Method fusion in the machine-learning methodology. We use the cross-entropy squared error as loss function.

$$\text{loss} = \sum_{ij} (\log p_{ij})^2$$

General Terms
data mining, recommender systems

Keywords
similarity rating reconstruction, item-item similarity, frequent sequence mining

2. Place: “A classical predictive modeling approach”
Neo Metrics – Spain

A classical predictive modeling approach for Task “Who rated what?” of the KDD CUP 2007
Jorge SUEIRAS
Neo Metrics
C/ Acajuela 1
28043 Madrid, Spain
+j4 91 382 45 54
jorge.sueiras@neometrics.com
Alfonso Salafantra
Neo Metrics
C/ Acajuela 1
28043 Madrid, Spain
+j4 91 382 45 54
alfonso.salalfantra@neometrics.com
Jose Luis Florez
Neo Metrics
C/ Acajuela 1
28043 Madrid, Spain
+j4 91 382 45 54
jose.luis.florez@neometrics.com

ABSTRACT
This paper describes one way possible to solve task “Who rated what?” of the KDD Cup 2007. The proposed solution is a linearly-based model that predicts whether a user will rate a given movie. Key points to our approach are: (1) the estimation of the model baseline, (2) the definition of the explanatory variables and (3) the mathematical model itself. Given the binary outcome of the problem, the estimation of the parameters follow a binomial distribution. To simplify and parallelize the model, we have defined a matrix estimation of the model variables. These explanatory variables can be grouped as: user rating behavior variables, the movie characteristics and auto-correlation interactions. Finally, the mathematical model form (linear logistic regression) has been chosen among various model-fitting computer.

Categories and Subject Descriptors
1.3.4 (Pattern Recognition): Models – statistical

Keywords
Predictive modeling, data mining

1. INTRODUCTION
This task is to predict whether a user will rate a given movie. The KDD Cup 2007 data set contains 1,000,000 user-movie pairs, each pair consists of user id, movie id and the time the movie was rated. The user id is the unique identifier of the user, whereas the movie id is the unique identifier or Netflix Id of the movie.

1.1. Model and variable selection
We build a predictive model whose target variable is the binary event of rating a movie in 2007 and whose input variables are binary time series data of the movie release. This may include variable and model form selection.

1.2. Baseline estimation
In order to estimate the baseline we must pay attention to the KDD Cup 07 FAQ. The FAQ document states that the 100,000 most pairs were selected by randomly picking up pairs (user-movie) with probability proportional to the number of times each component appears in the 2006 ratings. Furthermore, the user and the movie are chosen independently.

We consider that current estimation of the baseline is important in order to attain a good solution to the problem posed. For baseline estimation we shall perform to implement the procedures used to create the scoring data, in order to produce a training dataset with similar characteristics. The sampling algorithm is as follows:

1. Define the time range for the target variable, in our case:

$\text{baseline} = \frac{\text{time}_{\text{range}}}{2006}$

Winner of 2007

Who Rated What: a combination of SVD, correlation and frequent sequence mining
Miklós Kurucz
András A. Benczur
Tamás Kiss
Iszán Nagy
Adrienn Szabó
Balázs Torma
Data Mining and Web search Research Group, Informatics Laboratory
Computer and Automation Research Institute of the Hungarian Academy of Sciences
(realsac, benczur, kisstom, iszani, aszabo, torma)@iibi.sztaki.hu

ABSTRACT
KDD Cup 2007 focuses on predicting aspects of movie rating behavior. We present our prediction method for Task 1 “Who Rated What” in 2007 when the task is to predict which user rated which movie in 2007. We use the combination of the following predictors, listed in the order of their efficiency in the prediction:
- The predicted number of ratings for each movie based on time series prediction, also using movie and DVD release dates and movie score selections by the AFID distance of the titles.
- The predicted number of ratings by each user by using the fact that ratings were sampled proportional to the number of ratings by each user.

Categories and Subject Descriptors
C.2.4 [Computer Applications], Social and Behavioral Sciences: C.2.4.1 [Mathematics of Computing], Numerical Analysis: C.6.7 [Theory]: Computational Geometry and Object Modeling

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions Dept, ACM Inc., fax +1 (212) 869-0481.

1. INTRODUCTION
Recommender systems predict the preference of a user on a given item based on known ratings. In order to evaluate methods, in October 2006 Netflix provided movie ratings from anonymous customers on nearly 15,000 movies under license to researchers. The KDD Cup tasks related to this data set. For Task 1 “Who Rated What” the model has to predict which users rated which movie in 2007 while for Task 2 “How Many Ratings in dbbi” the task was to predict the number of additional ratings of movies.
In this paper we present our method for Task 1 “Who Rated What”. The task was to predict the probability that a user rated a movie in 2006 (with the actual date and rating being consistent) for a given list of 100,000 movies per user. The movie and user data derive from the Prime data set, i.e., the movies appeared (or at least received ratings) before 2006 and the users also gave their first rating before 2006 that some of the pairs were rated in the training set. We give a detailed description of the sampling process and 28 rules to get information that we use for the prediction.

1.1. Method prediction

Method fusion in the machine-learning methodology. We use the cross-entropy squared error as loss function.

$$\text{loss} = \sum_{ij} (\log p_{ij})^2$$

General Terms
data mining, recommender systems

Keywords
similarity rating reconstruction, item-item similarity, frequent sequence mining

1. INTRODUCTION
This task is to predict whether a user will rate a given movie. The KDD Cup 2007 data set contains 1,000,000 user-movie pairs, each pair consists of user id, movie id and the time the movie was rated. The user id is the unique identifier of the user, whereas the movie id is the unique identifier or Netflix Id of the movie.

1.1. Model and variable selection
We build a predictive model whose target variable is the binary event of rating a movie in 2007 and whose input variables are binary time series data of the movie release. This may include variable and model form selection.

1.2. Baseline estimation
In order to estimate the baseline we must pay attention to the KDD Cup 07 FAQ. The FAQ document states that the 100,000 most pairs were selected by randomly picking up pairs (user-movie) with probability proportional to the number of times each component appears in the 2006 ratings. Furthermore, the user and the movie are chosen independently.

We consider that current estimation of the baseline is important in order to attain a good solution to the problem posed. For baseline estimation we shall perform to implement the procedures used to create the scoring data, in order to produce a training dataset with similar characteristics. The sampling algorithm is as follows:

1. Define the time range for the target variable, in our case:

$\text{baseline} = \frac{\text{time}_{\text{range}}}{2006}$
Track 1 – Assignment 2007

“Which user rated which movies in 2006”

- Data set of the last years
- 100,000 user-movie pairs of 2006

→ Probability that user-movie-pair is rated

0.7
“How many Ratings”

→ number of total ratings
Paper 1 – Result & Approach

Result: RMSE (stdv) = 0.256

0.5533 * base prediction
+ 0.1987 * singular value decomposition
+ 0.029 * item item similarity
+ -0.0121 * association rules – 0.0042
Paper 1 – Base Prediction

Result: RMSE (stdv) = 0.256

\[
\text{prob (user-movie pair } x) = 0
\]

\[\rightarrow 10^{th} - 13^{th} \text{ place with stdv } = 0.279\]

\[
\text{guessing correct factors for baseline}
\]

\[\rightarrow 5^{th} - 6^{th} \text{ Place with stdv } = 0.268\]
Paper 1 – Base Prediction

\[ p_{um} = \frac{(N_u \times N_m)}{M \times U} \]

user-movie-relation independent

\( N_u = \) number of ratings of the user
\( N_m = \) number of ratings of the movie
\( M = \) total number of movies
\( U = \) total number of users

to estimate (Track 2 - 2007)
Paper 1 – Base Prediction

\[ p_{um} = \left( N_u \times N_m \right) / M \times U \]

user-movie-relation independent

- \( N_u \) = number of ratings of the user
- \( N_m \) = number of ratings of the movie
- \( M \) = total number of movies
- \( U \) = total number of users

we know (Track 2 - 2011)
Paper 1 – Prediction #Ratings/Movie $N_m$

Secondary Information:
(DVD-Release, IMDB Movie Release, series continuation releases)

→ analyse time distribution and continue
Paper 1 – Prediction #Ratings/User $N_u$ 

- same sampling-method as in KDD-Cup 2007
- stdv of sampled ratings of 2005
- stdv of base predictions of 2006
- Compare $\text{stdv}_{2006}$ and $\text{stdv}_{2005}$
- adapt 2005 to 2006

„Who Rated What [...]“ - ilab
Paper 1 – SVD

\[ C_k = U \Sigma_k V^T \]

**In:** u-m-matrix with predictions from several base prediction-values

**Out:** denser matrix

**Eckhart-Young Theorem:**
after using svd you got a rank-k-matrix, which is an approximation of the original matrix

**Implementing "Lanczos" (SVD-pack)**
Too high number of dimensions leads to "overfitting"
→ Machine learning approach to get optimal k for SVD-partition
→ calculate optimal partition with Frobenius Norm = error value

→ important for us
Figure 1: The distribution of the 10-dimensional approximation for user–movie pairs with and without ratings.
# Paper 1 – Item Item Similarity

Cosine similarity:

\[
sim(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\|_2 \times \|\vec{j}\|_2}
\]

<table>
<thead>
<tr>
<th>vectors</th>
<th>Item a</th>
<th>Item b</th>
<th>Item c</th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>20</td>
<td>30</td>
<td>40</td>
</tr>
<tr>
<td>j</td>
<td>42</td>
<td>23</td>
<td>66</td>
</tr>
<tr>
<td>k</td>
<td>10</td>
<td>90</td>
<td>30</td>
</tr>
</tbody>
</table>
Figure 2: The distribution of the item-item similarity based prediction for user–movie pairs with and without ratings for a similarity top list of size $K = 5$. 
Weka toolkit:
- training data: sample of 2005 (sampling method 2007)
- Applied to data of 2006
Paper 2 – Result & Approach

Result: RMSE (stdv) = 0,263

- deliberate selection of variables
- constructing more variables with SVD
- Machine learning over all variables with own training sample

„classical modeling approach [...]“ – Neo Metrics
guessed baseline: ca. 20%

baseline after data cleaning: 3.8%

cleaning over time dependent informations
-> new users and movies are "outlier"
(in 2004 avg. more ratings, in 2005 less)
-> eliminate "outliers" to create time independent model

"real" baseline: 7.8%
User Variables:
- Number of historic user ratings
- Percent of 1-star ratings of the user
- Stdv of user ratings
- Number of months since the first rating of the user
- ...

Movie Variables:
- Number of historic ratings received by the movie
- Percent of 1-star ratings received by the movie
- Stdv of ratings received by the movie
- ...

User-Movie Interactions (after SVD):
- Likelihood of rating similar movies more than the mean
- Likelihood of similar users rating the movie more than the mean
Paper 2 – SVD, Cluster Analysis

- SVD with user-movie-matrix with ratings

→ to group by users / movies in matrix

→ cluster analysis
Paper 2 – Machine Learning

- training data: sample of 2004 (sample method of 2007)
- Applied to 2005
  → Weighting of all variables
Conclusion / our possible Approach

- baseline in both papers very important
  → we cannot use such kind of baseline
- SVD used in two different ways
  → could also be important for us
- Item Item Similarity was less efficient
  → we think more efficient for us
- Machine learning
  → perhaps to weight our methods)

- Work with hierarchies → for clustering
  - n songs of the same album rated
  - delete users from training data users with incalculable music taste?
Sources:

http://kddcup.yahoo.com/


Su, Xiaoyuan & Khoshgoftaar, Taghi M.: "A survey of collaborative filtering techniques"

Miklós Kurucz, András A. Benczúr, Tamás Kiss, István Nagy, Adrienn Szabó & Balázs Torma: "Who Rated What: a Combination of SVD, Correlation and Frequent Sequence Mining"


Miklós Kurucz, András A. Benczúr, Károly Csalogány: “Methods for large scale SVD with missing values”

Book “???‘‘ of Arvid 😊 with explanation of SVD

