

# Evaluation of Recommender Systems

Radek Pelánek

2014

# Summary

Proper evaluation is really difficult.

# Proper Evaluation is Difficult ...

- hypothetical examples
- illustrations of flaws in evaluation

# Case I

- e-commerce systems for selling foobars
- recommendations available, can be used without recommendations
- compare:
  - group 1: users using recommendations
  - group 2: users not using recommendations
- measurement: number of visited pages
- result:  $\text{mean}(\text{group 1}) > \text{mean}(\text{group 2})$
- conclusion: recommendations work!

flaws?

- what do we measure: number of pages vs sales
- division into groups: potentially biased (self-selection) vs randomized
- statistics: comparison of means is not sufficient (p-value, effect size)

# Case II

- two models for predicting ratings of foobars (1 to 5 stars)
- metric for comparison: how often model predicts correct rating
- Model 1 has better score than Model 2
- conclusion: using Model 1 is better than using Model 2

flaws?

# Issues

- over-fitting, train/test set division
- metric:
  - models usually give float; exact match not important
  - we care about the size of the error
- statistical issues
- better performance wrt metric  $\Rightarrow$  better performance of the recommender system ???

# Research Methods

- experimental
  - at least one variable manipulated, units randomly assigned
  - ideally “randomized controlled trial”
  - “online experiments”
- non-experimental
  - “offline experiments”
  - historical data
- simulation experiments
  - simulated data, limited validity
  - “ground truth” known, good for “debugging”

# Offline Experiments

- data: “user, product, rating”
- cross-validation
- performance of model – difference between predicted and actual rating

predicted	actual
2.3	2
4.2	3
4.8	5
2.1	4
3.5	1
3.8	4

# Overfitting

- model performance good on the data used to build it; poor generalization
- too many parameters
- model of random error (noise)
- typical illustration: polynomial regression

# Cross-validation

- aim: avoid overfitting
- split data: training, testing set
- training set – setting model “parameters” (may include selection of fitting procedure, number of latent classes, ...)
- testing set – evaluation of performance

(more details: machine learning)

# Cross-validation

train/validation set division:

- typical ratio: 80 % train, 20 % test
- $N$ -fold cross validation:  $N$  folds, in each turn one fold is the testing set
- how to divide the data: time, user-stratified, ...

# Note on Experiments

- (unintentional) “cheating” is easier than you may think
- data leakage
  - training data corrupted by some additional information
- useful to separate test set as much as possible

predicted	actual
2.3	2
4.2	3
4.8	5
2.1	4
3.5	1
3.8	4

- MAE (mean absolute error)

$$MAE = \frac{1}{n} \sum_{i=1}^n |a_i - p_i|$$

- RMSE (root mean square error)

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (a_i - p_i)^2}$$

- correlation coefficient

# Normalization

- used to improve interpretation of metrics
- e.g., normalized MAE

$$NMAE = \frac{MAE}{r_{max} - r_{min}}$$

# Note on Likert Scale

- 1 to 5 “stars” ~ Likert scale (psychometrics)  
strongly agree, agree, neutral, disagree, strongly disagree
- ordinal data? interval data?
- for ordinal data some operation (like computing averages) are not meaningful
- in RecSys commonly treated as interval data

# Binary Predictions

- relevant / not-relevant
- like / dislike
- known / not-known (educational systems)

note: quite closely related to evaluation of models for weather forecasting

# Metrics for Binary Predictions

- do not use:
  - MAE can be misleading (not a “proper score”)
  - correlation harder to interpret
- reasonable metrics:
  - RMSE
  - log-likelihood

$$LL = \sum_{i=1}^n c_i \log(p_i) + (1 - c_i) \log(1 - p_i)$$

# Information Retrieval Metrics

		Reality		
		Actually Good	Actually Bad	
Prediction	Rated Good	True Positive (tp)	False Positive (fp)	All recommended items
	Rated Bad	False Negative (fn)	True Negative (tn)	

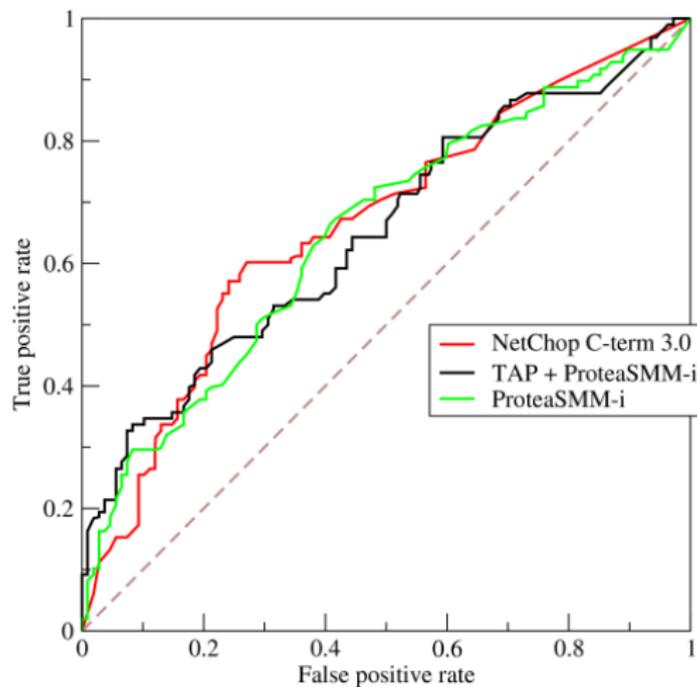
All good items

- $\text{precision} = \frac{TP}{TP+FP}$   
good items recommended / all recommendations
- $\text{recall} = \frac{TP}{TP+FN}$   
good items recommended / all good items
- $F1 = \frac{2TP}{2TP+FP+FN}$   
harmonic mean of precision and recall

# Receiver Operating Characteristic

- to use precision, recall, we need classification into two classes
- probabilistic predictors: value  $\in [0, 1]$
- fixed threshold  $\Rightarrow$  classification
- what threshold to use? (0.5?)
- evaluate performance over different threshold  $\Rightarrow$  Receiver Operating Characteristic (ROC)
- metrics: area under curve (AUC)

# Receiver Operating Characteristic



Source: Wikipedia

(with IR metrics, AUC)

- often difficult to establish the ground truth
- skewed distribution of classes – hard interpretation  
always use baselines
- AUC used in many domains, sometimes overused

# More Issues

(with all metrics)

- ratings not distributed uniformly across users/items
- averaging:
  - global
  - per user?
  - per item?

# Ranking

- typical output of RS: **ordered** list of items
- ranking metrics – extensions of precision/recall
- swap on the first place matters more than swap on the 10th place

# Ranking Metrics

- Spearman correlation coefficient
- half-life utility
- liftindex
- discounted cumulative gain
- average precision

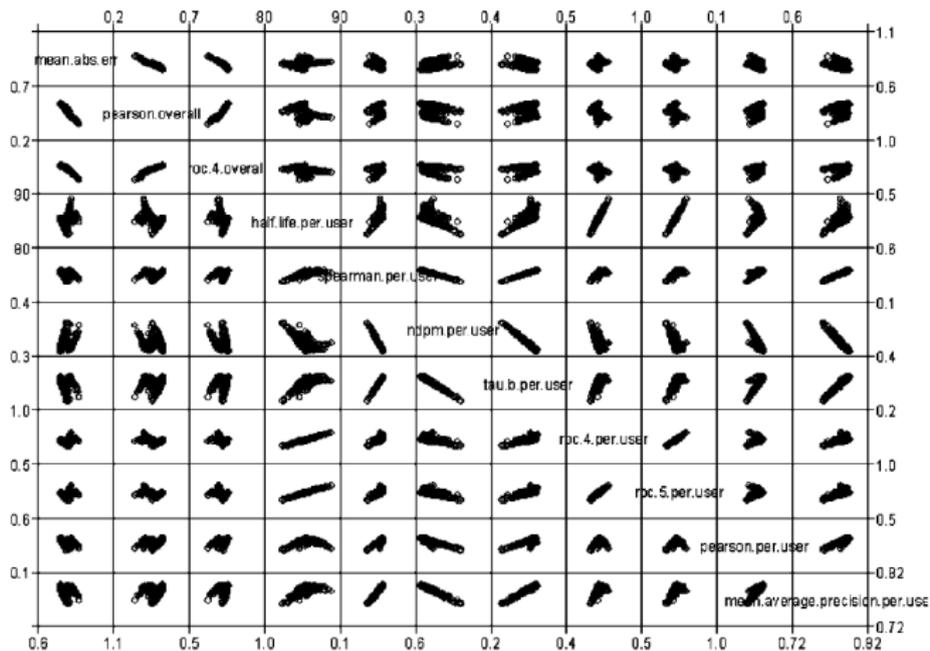
# Metrics

- which metric should we use in evaluation?
- does it matter?

# Metrics

- which metric should we use in evaluation?
- does it matter?
  
- it depends...
- my advice: use RMSE as the basic metric

# Accuracy Metrics – Comparison



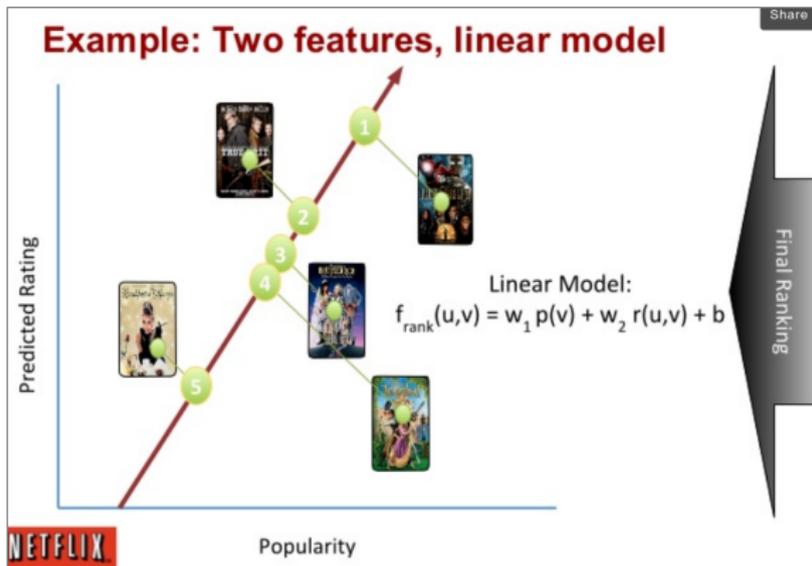
Evaluating collaborative filtering recommender systems, Herlocker et al., 2004

# Example



Introduction to Recommender Systems, Xavier Amatriain

# Example



Introduction to Recommender Systems, Xavier Amatriain

# Beyond Accuracy of Predictions

harder to measure (user studies may be required)  $\Rightarrow$  less used  
(but not less important)

- coverage
- confidence
- novelty, serendipity
- diversity
- utility
- robustness

# Coverage

- What percentage of items can the recommender form predictions for?
- consider systems X and Y:
  - X provides better accuracy than Y
  - X recommends only subset of “easy-to-recommend” items

# Novelty, Serendipity

- it is not that difficult to achieve good accuracy on common items
- valuable feature: novelty, serendipity
- serendipity  $\sim$  deviation from “natural” prediction
  - successful baseline predictor  $P$
  - serendipity – good, but deemed unlikely by  $P$

# Diversity

- often we want diverse results
- example: holiday packages
  - bad: 5 packages from the same resort
  - good: 5 packages from different resorts
- measure of diversity – distance of results from each other
- precision-diversity curve

# Online Experiments

- randomized control trial
- AB testing

# Online Experiments – Comparisons

we usually compare averages (**means**)

- are data (approximately) normally distributed?
- if not, averages can be misleading
- specifically: presence of outliers → use median or log transform

# Statistics Reminder

- statistical hypothesis testing
- t-test, ANOVA
- significance, p-value
- error bars

note: RecSys – very good opportunity to practice statistics

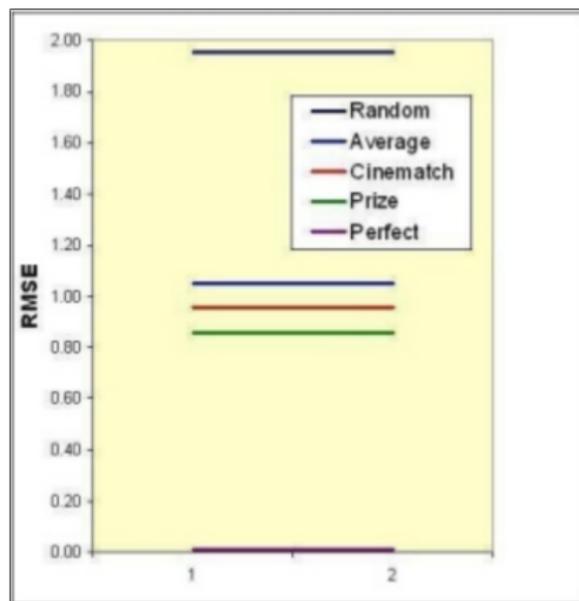
# Simulated Experiments

- simulate data according to a chosen model of users
- add some noise
- advantages:
  - simple, cheap, fast
  - very useful for testing implementation (bugs in models)
  - insight into behaviour, sensitivity analysis
- disadvantage: results are just consequence of used assumptions

# Interpretation of Results

- what do numbers mean?
- what do (small) differences mean?
- are they significant?
  - statistically?
  - practically?

# Interpretation of Results



Introduction to Recommender Systems, Xavier Amatriain

# What is Popular?

availability of data biases what is done

- evaluations on historical data sets
- accuracy of predictions: RMSE, MAE, precision/recall
- datasets: movies (Netflix, MovieLens), web 2.0 platforms

# Summary

Proper evaluation is difficult...

- not clear what to measure, how
- things we care about are hard to measure
- many different metrics
- different experimental settings
- it is easy to cheat (unintentionally)

specific examples (case studies) in next lectures

# Evaluation and Projects

- analysis of existing data (provided, yours)
  - offline experiments, cross-validation, RMSE, ...
- extension of an existing system
  - online experiments (A/B testing)
- new system “from scratch”
  - at least some basic summaries