# Big Data vs. Right Data

Boi Faltings

Applied Machine Learning Days 2017

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

# Machine learning in AI

- Machine learning is a great way to build models for AI systems.
- AI system will influence its environment
- => training data no longer representative.
- => learned knowledge not valid.
- How do we correct for this effect?

# Recommender Systems

- Very popular research topic (conference with over 500 participants)
- Widely used in practice.
- Gap between academic research:
  - uses fixed datasets collected without recommendation.
- and actual application:
  - recommendations influence behavior.

# News Recommendation


- Keep readers on the site to increase revenue.
- Session-based: personalize based on browsing behavior.
- Algorithms tuned on user behavior.

# Online vs. Offline

- Online behavior: user behavior when exposed to recommendations.
  - Separate online data for each algorithm.
- Offline behavior: user behavior without recommendations.
  - Independent of algorithm: one online data collection allows testing many algorithms.
- Is offline a good proxy for online?

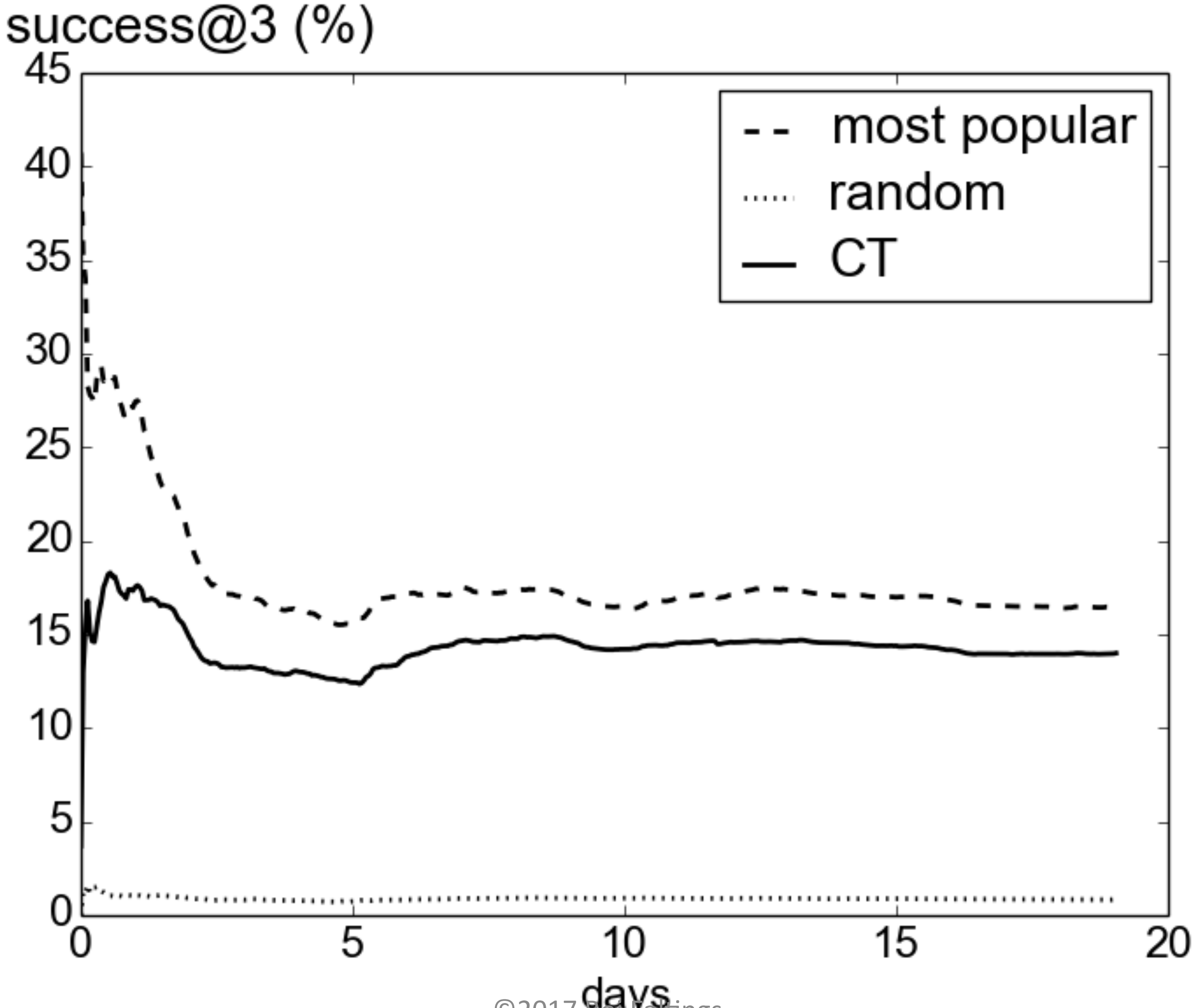
# Accuracy Metrics

- Offline: predict what the user will read next.
  - success@k = 1 when next viewed article is in the recommended set of size k.
- Online: observe what the user clicks.
  - Click-through rate (CTR) is the number of clicks on recommended articles over the total number of displayed recommendations.

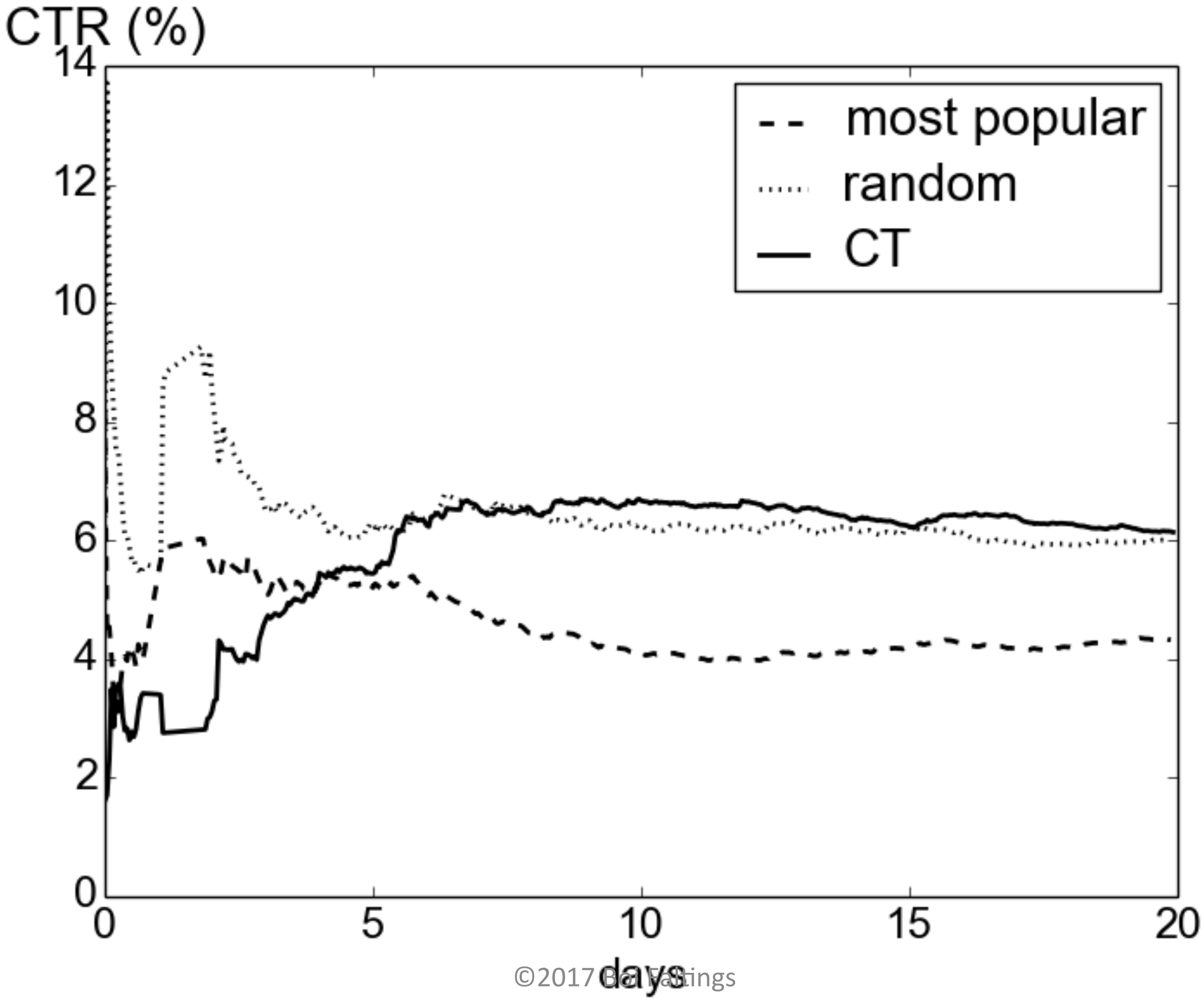
# Recommendation Algorithms

- Recommend the most popular items.
- Recommend random items.
- Recommend preferred items, as learned from user behavior
  - here: context tree (CT)
  - variable-order Markov model continuously adapted to new observations.

# Cumulative Offline Accuracy



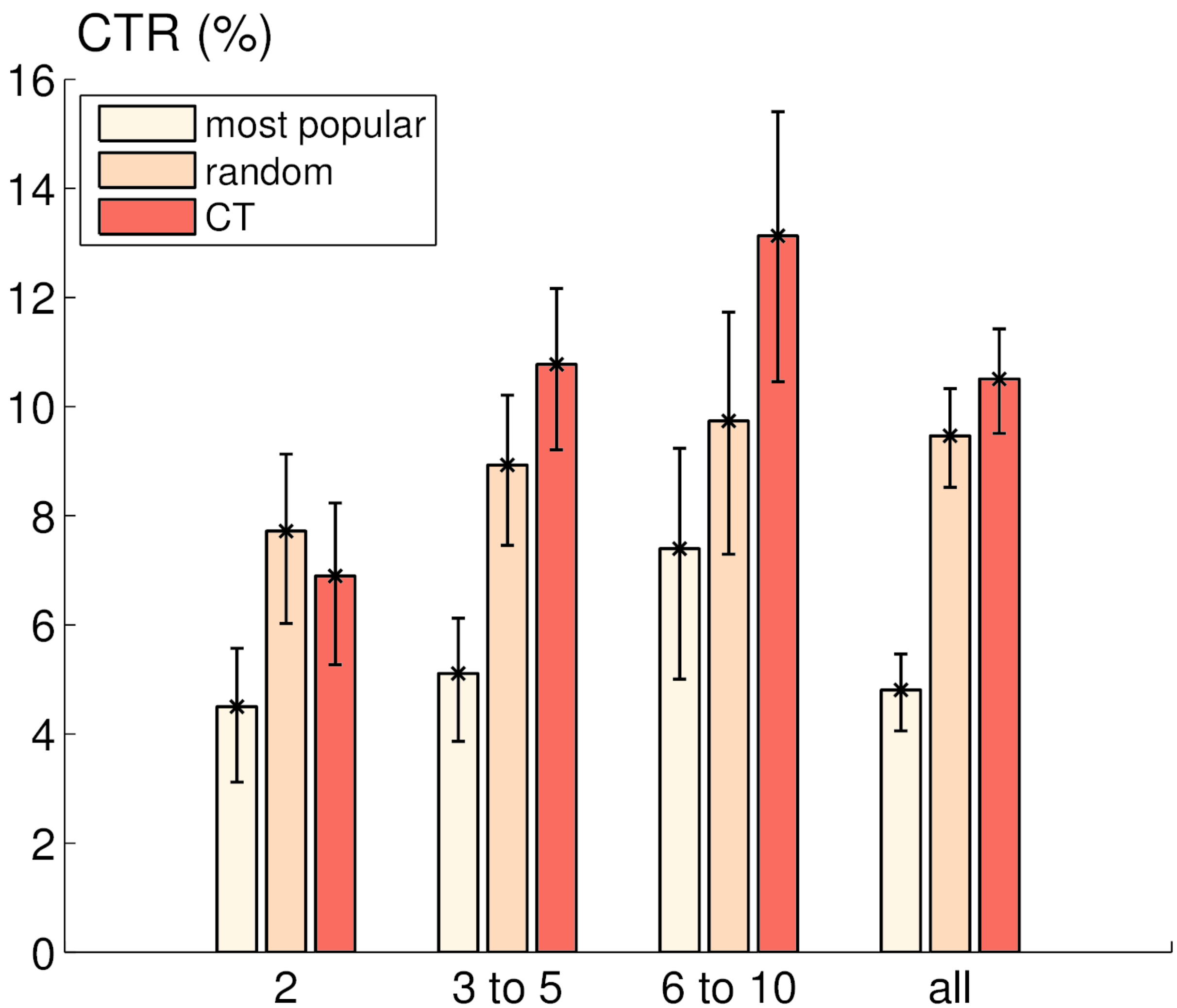
# Cumulative Online CTR



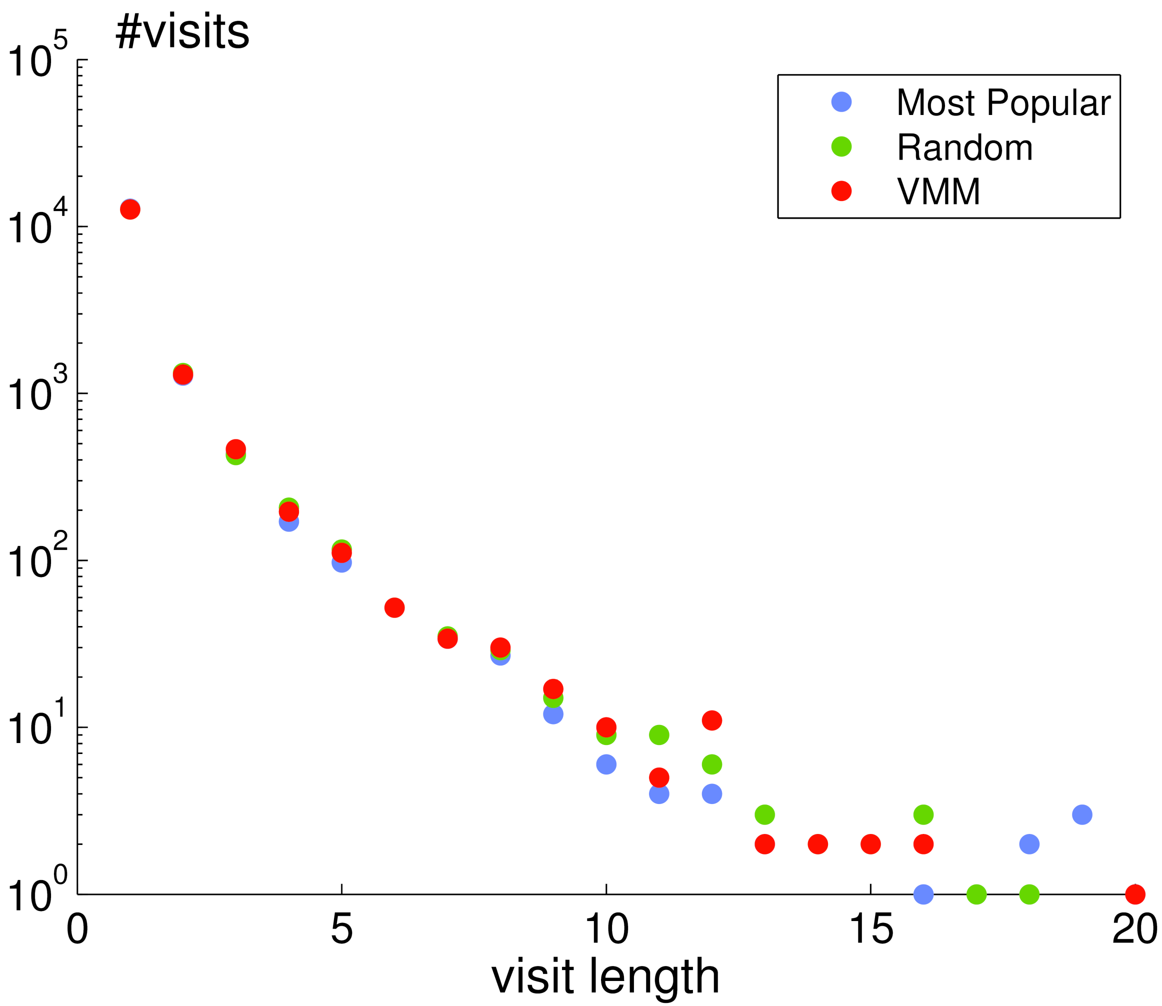
# Is random really that good?

- How can random recommendations be as good as the learning algorithm?
- Learning requires data => cannot work on short traces.
- When too little data, approaches most popular (common issue with recommenders).

# Random stronger on short visits



# Most visits are very short



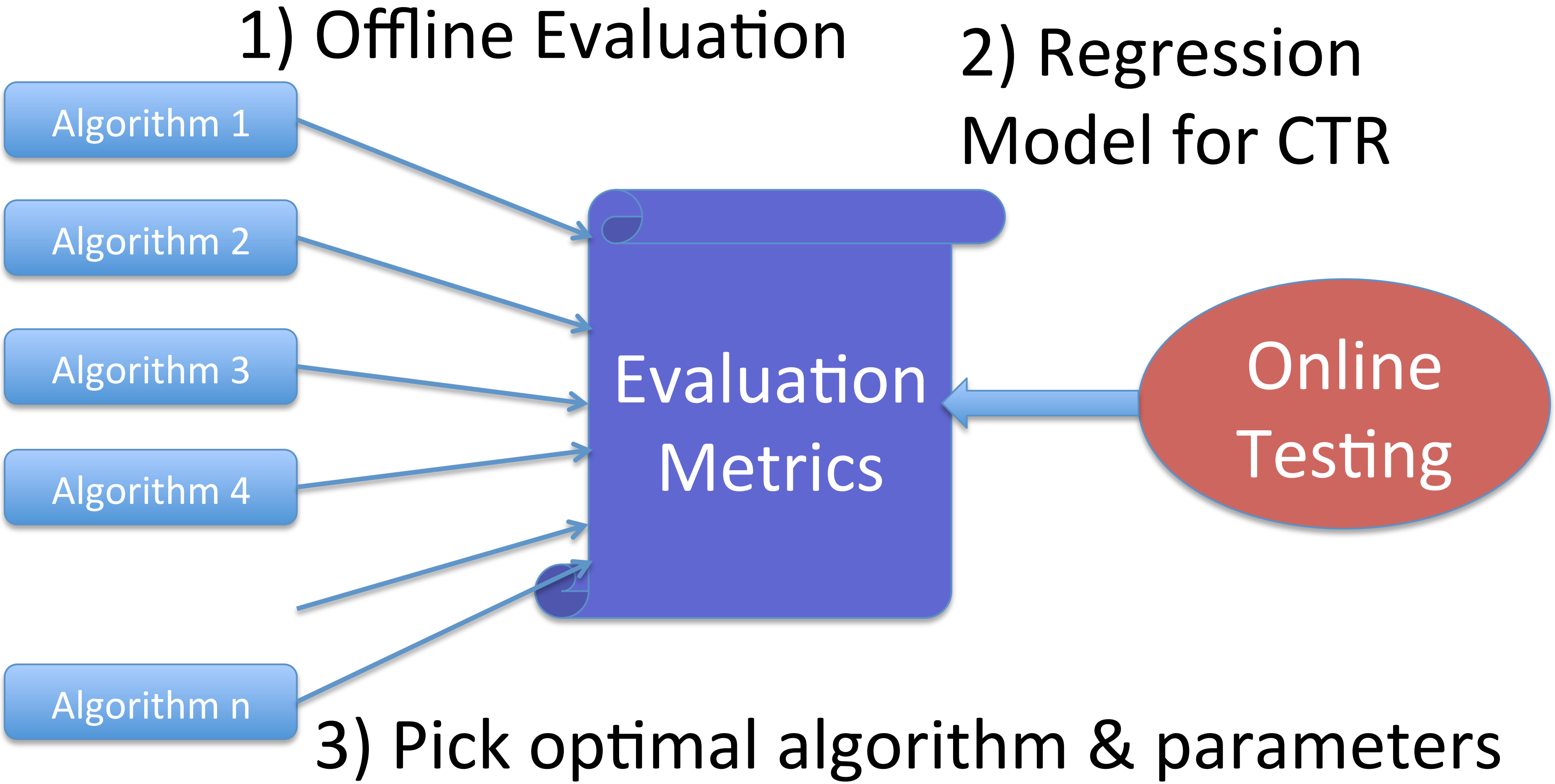
# How could we do better?

- Offline evaluation completely wrong, but online evaluation much too costly for optimizing recommender algorithms.
- Collect data with random recommendations
  => all sequences present in data.
  => shows reaction for any recommendation.

# Random Recommendations

- Solution explored by Li et al. (2011):
  - collect logs with random recommendations
  - given browsing history $p_1..p_k$, find log sequences $p_1..p_k$ and recommend most likely next item x
  - predict CTR for x from CTR in log
- Works for Yahoo home page: only 20 items.
- Doesn't work for news: thousands of items; random generation only shows a few of them.

# Algorithm Selection and Tuning



# Evaluation Metrics

Considered 17 metrics grouped into:

- Accuracy: is recommended item chosen?
- Diversity: dissimilarity of recommendations.
- Coverage: are all items recommended?
- Serendipity: unexpected and useful?
- Novelty: is recommendation long-tail item?

# Online experiment to find user model

- Test a few recommendation strategies.
- Measure their success rate, CTR and evaluation metrics.
- Feature selection using least angle regression:
  – Regularizer to minimize number of features.
  – Decrease multiplier of regularizer.
  – Order features by when they enter model.

# Building Regression Model

- Feature selection (Swissinfo):

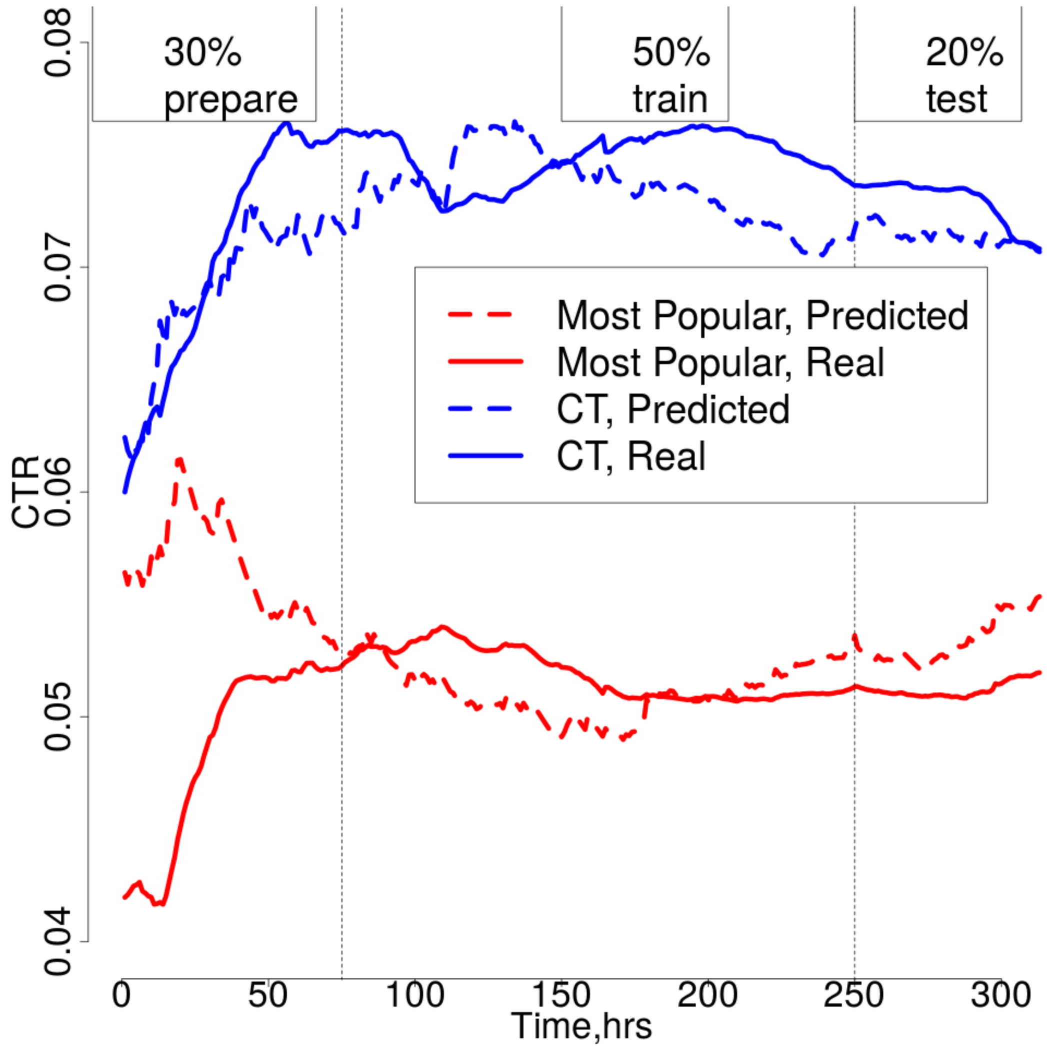
| Metric Group | First to enter | Avg. entry value (± std. dev.) |
|---|---|---|
| Diversity | Personalization | 2.53 ± 0.65 |
| Serendipity | Serendipity | 2.71 ± 0.58 |
| Accuracy | Markedness | 2.82 ± 0.76 |
| Coverage | Shannon Entropy | 5.94 ± 0.80 |
| Novelty | Novelty | 10.27 ± 2.77 |

- Accuracy is only the third most important predictor!

# Online CTR prediction

- Given an algorithm:
  - Measure performance metrics on offline data.
  - Apply regression model to predict CTR.
- Quite accurate:
  - RMSE around 0.5% of actual CTR
  - At least 2x better than accuracy alone.

# Example Predictions



# Methodology

- Develop broader performance features besides accuracy.
- Train model to predict online accuracy from these features.
- => optimize online performance with offline data.

# Conclusions

- Challenge for using machine learning in AI: *Training data not representative of application*
- Cleanest: collect data with random actions.
- Common: incremental deployment, maybe with reinforcement learning.
- Alternative: learn model to map offline performance to online performance.