Improving Understanding and Exploration of Data by Non-Database Experts

Rachel Pottinger

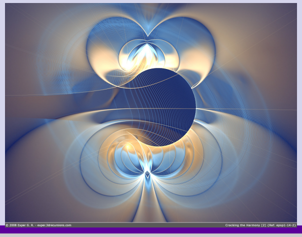
University of British Columbia

Joint work with lots of great students, including Zainab Zolaktaf, Reza Babanezhad, Jian Xu, Omar AlOmeir, and Janik Andreas

Exploring and understanding data

- More users have more data
- This is particularly challenging for users without much database background
- I like to work with data and users who have real world problems. Then I extend to a more general scenario.
- How can we help users with little database expertise to understand and explore their data?

Exploring and understanding data



- **Exploration**: recommend items beyond the popular items in recommender systems
- **Understand**: help users understand the range of possible answers in data aggregated from multiple sources
- **Exploration** and **understanding**: Ongoing work on exploring and understanding

Exploration: Recommend long tail items
(joint with Zainab Zolaktaf and Reza Babanezhad)

- Standard recommender systems algorithms tend to emphasize popular items
- This tends to cause recommendation consumers to only find things they already know
- But most items are “long tail”
- Presented at ICDE (International Conference on Data Engineering) last week

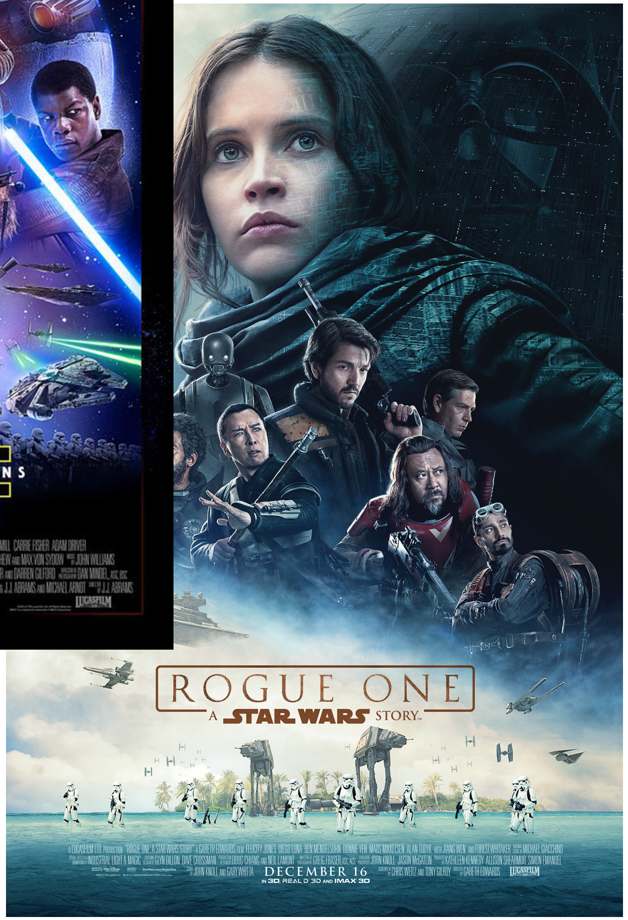
Motivating Example

Top-N recommendation

- Recommend to each user a set of N items from a large collection of items
- Used in Netflix, Amazon, IMDB, etc.

Problem

- Tend to recommend things users are already aware of
- E.g., Suggests “Star Wars: The Force Awakens” to users who have seen “Star Wars: Rogue One”



Motivating Example

- Many recommendation systems
  - Take as input a set of users and their ratings (e.g., ratings on movies)
  - Focus on accurately predicting user preferences based on history
  - Use a subset of data as "gold standard"
- Interaction data often suffers from popularity bias and sparsity
  - Have to recommend popular items to maintain performance accuracy
  - Rich get richer effect
- Accuracy alone is not leading to effective suggestions?

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

Why long-tail items matter

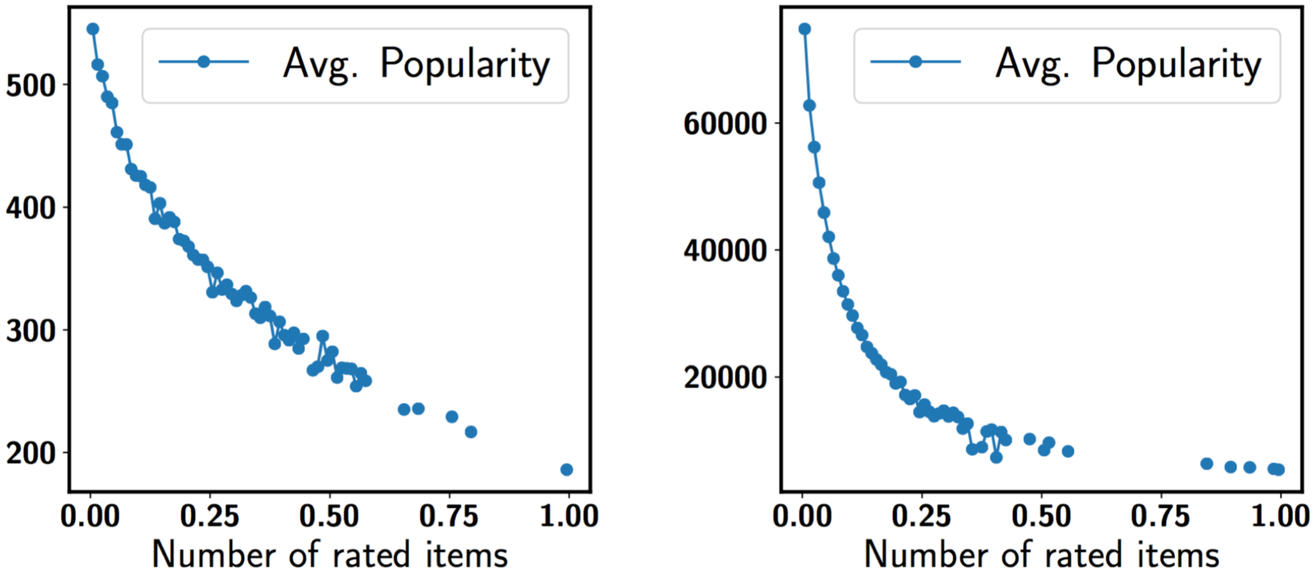
Consumers want

- **Accuracy**
- **Novelty**
- …

Providers of items want

- Keep consumers happy
- Item-space **coverage**
  - Generates revenue
- …

Less focus on popular items



(a) ML-1M (b) Netflix

- Long-tail items
  - Generate the lower 20% of the observations
  - Empirically validated: Correspond to almost 85% of the items in several datasets

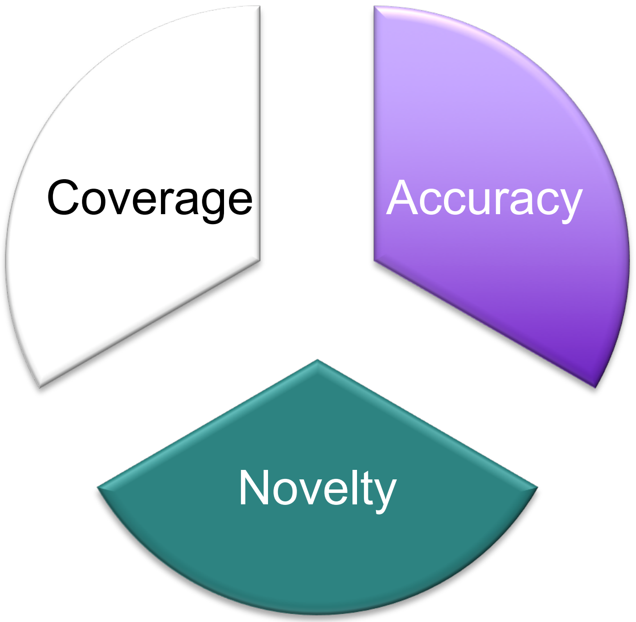
Selected related work

- Accuracy Focused
  - KBV09- Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." *Computer*42.8 (2009).
  - WKL+08- Weimer, Markus, et al. "Cofi rank-maximum margin matrix factorization for collaborative ranking." *Advances in neural information processing systems*. 2008.
- Re-ranking frameworks
  - AK12- Adomavicius, Gediminas, and YoungOk Kwon. "Improving aggregate recommendation diversity using ranking-based techniques." *IEEE Transactions on Knowledge and Data Engineering* 24.5 (2012): 896-911.
  - HCH14- Ho, Yu-Chieh, Yi-Ting Chiang, and Jane Yung-Jen Hsu. "Who likes it more?: mining worth-recommending items from long tails by modeling relative preference." *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, 2014.
- Evaluation of top-N recommendation
  - CKT10- Cremonesi, Paolo, Yehuda Koren, and Roberto Turrin. "Performance of recommender algorithms on top-n recommendation tasks." *Proceedings of the fourth ACM conference on Recommender systems*. ACM, 2010.
  - Ste11- Steck, Harald. "Item popularity and recommendation accuracy." *Proceedings of the fifth ACM conference on Recommender systems*. ACM, 2011.
  - Ste13- Steck, Harald. "Evaluation of recommendations: rating-prediction and ranking." *Proceedings of the 7th ACM conference on Recommender systems*. ACM, 2013.

Challenges: Accuracy, novelty, and coverage trade-offs

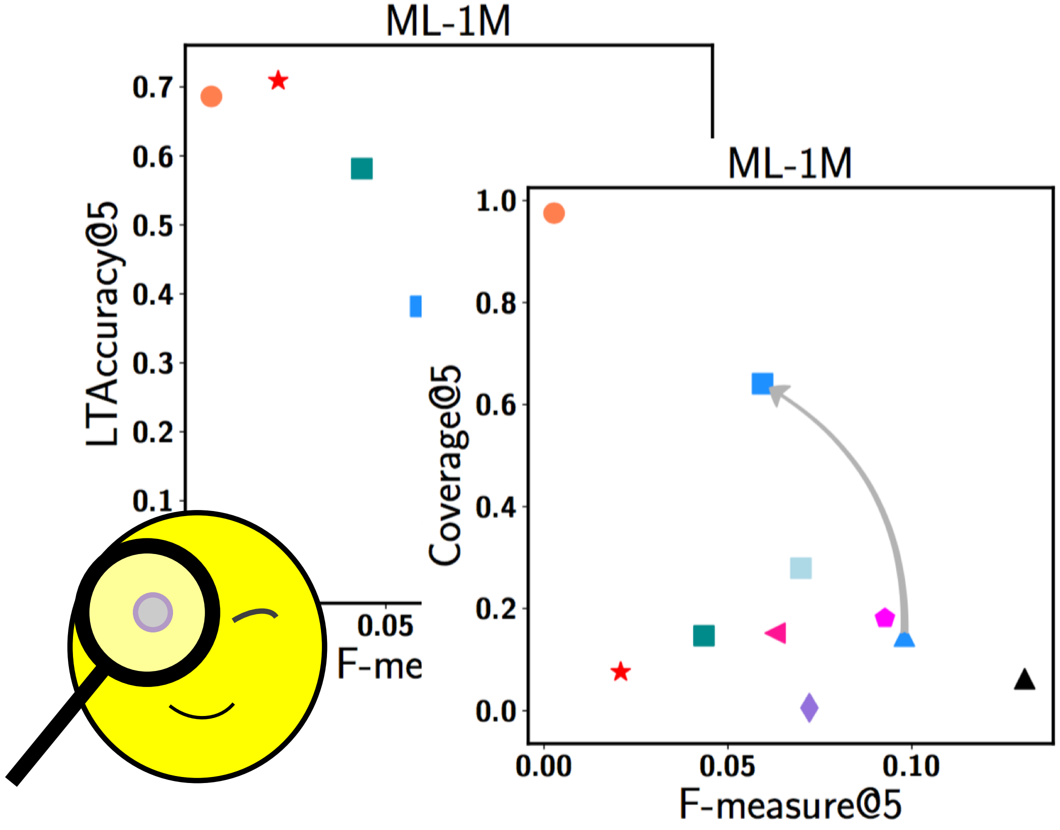
- ✓ Promoting long-tail item can increase novelty [Ste11]
  - Long-tail items are more likely to be unseen
- ✓ Promoting long-tail items increases coverage [Ste11]
  - Generates revenue for providers of items
- × Long-tail promotion can reduce accuracy [Ste11]

Not all users receptive of long-tail items



Challenges: Recommendation system evaluation

- Need to assess multiple aspects
  - Accuracy, novelty, and coverage
  - No single measure that combines all aspects. Report trade-offs?
- Need to consider real-world settings
  - Datasets are sparse
  - Users provide little feedback
- Test ranking protocols [Ste13, CKT10]
  - Do not reward popularity-biased algorithms
  - Offline accuracy should be close to what user experiences in real-world



Solution overview: GANC

A Generic top-N recommendation framework that provides customized balanced between Accuracy, Novelty, and Coverage

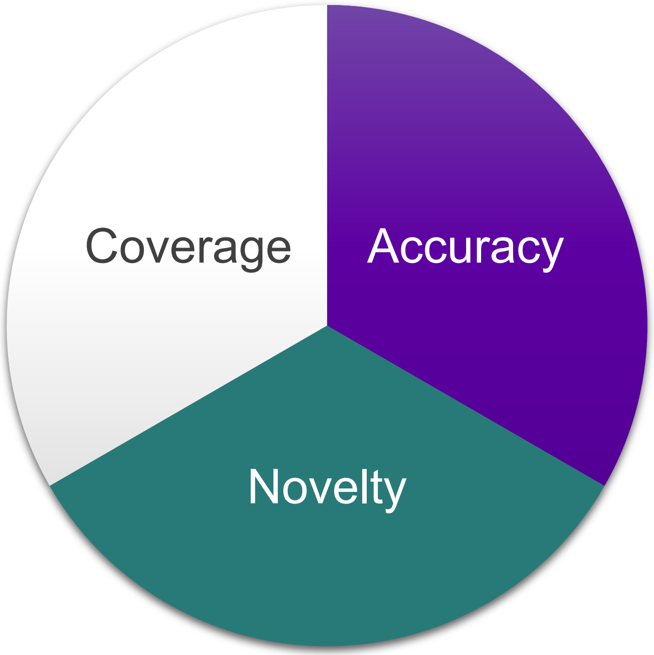
Objective: Assign top-N sets to all users

Find $\mathcal{P} = \{\mathcal{P}_u\}_{u=1}^{|\mathcal{U}|}$, the collection of top-N sets to maximize

$$v(P) = \sum_u v_u(P_u)$$

$$= \sum_u (1-\theta_u) a(P_u) + \theta_u c(P_u)$$

$$= \sum_u (1-\theta_u) \sum_{i \in P_U} a(i) + \theta_u \sum_{i \in P_U} c(i)$$



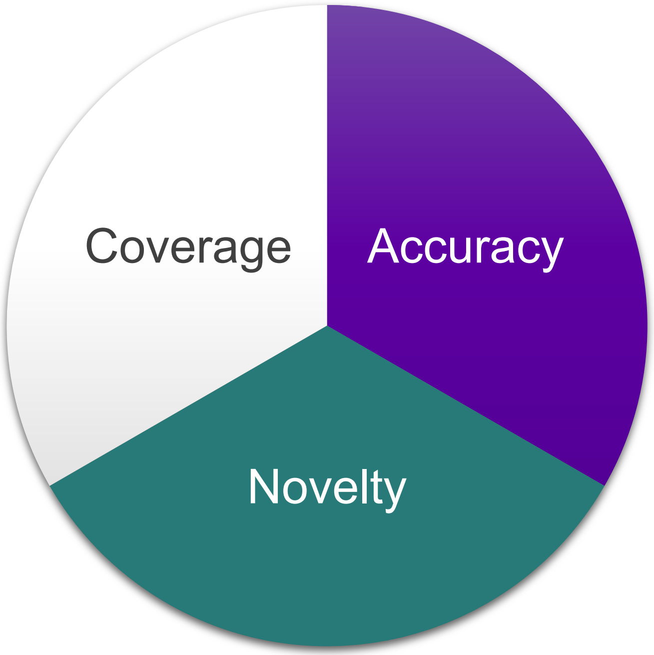
Solution overview: GANC

Main features of our solution

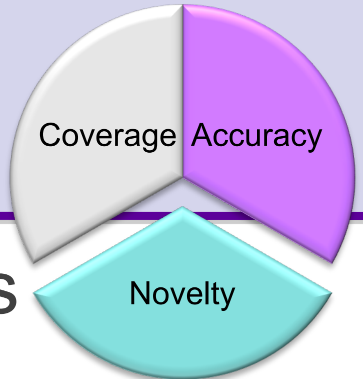
1. Directly infer user long-tail novelty preference θ_u from interaction data

   Customize trade-off parameters per user

2. Integrate θ_u into a generic re-ranking framework
   - θ_u independent of any base recommender
   - Plugin a suitable base recommender w.r.t. factors such as dataset density



Long-tail novelty preference model (θ_u)



We created and evaluated 4 long-tail novelty preference models

(1) Activity

- Number observations in the train set (e.g., number of rated items)
- Does not distinguish between long-tail and popular items

(2) Normalized long-tail measure

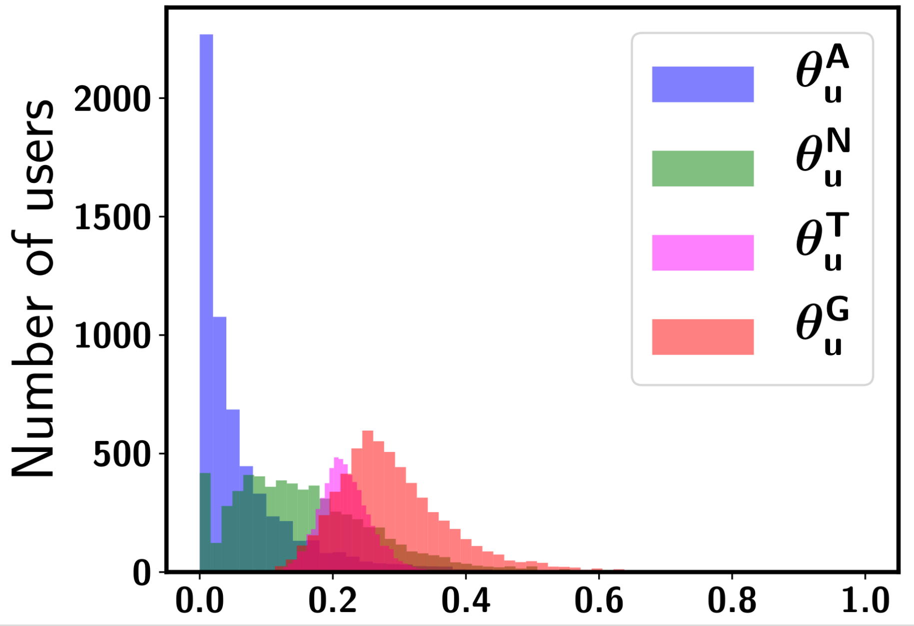
- Ratio of long-tail items rated in train set
- Does not consider if user liked item

(3) TFIDF-Measure

- Incorporates rating and popularity of items
- Does not consider view of other users

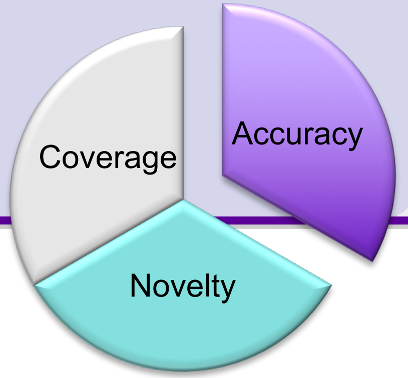
(4) Generalized measure

- Optimization approach
- Incorporates rating information, popularity of items, and view of other users



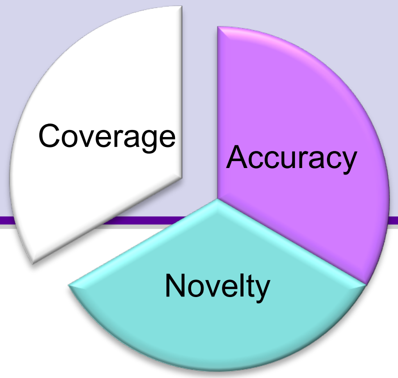
ML-1M (More dense)

GANC: Accuracy recommenders



- Focuses on making accurate suggestions
- Evaluated existing models from literature
  - PureSVD [CKT10]
  - Regularized SVD [KBV09]
  - Most Popular [CKT10]

GANC: Coverage recommenders



- Focus on increasing coverage
  - Random coverage recommender
  - Static coverage recommender
    - Consider how many times the item was rated in the past
      - Gain of recommending an item is proportionate to the inverse of its frequency in train set
  - Dynamic coverage recommender
    - Consider how many times item has been recommended so far
      - Gain of recommending an item is proportionate to the inverse of item recommendation frequency

Empirical Evaluation

Dataset	#Ratings	#Users	#Items	Density	Long-Tail %
ML-100K	100K	943	1682	6.30	66.98
ML-1M	1M	6,040	3,706	4.47	67.58
ML-10M	10M	69,878	10,677	1.34	84.31
MT-200k	172,506	7,969	13,864	0.16	86.84
Netflix	98,754,394	459,497	17,770	1.21	88.27

- ML = Movie Lens MT = Movie Tweetings.
- ML, MT, and Netflix these are common recommender datasets
- Datasets have varying level of density
- Long-tail items correspond to approximately 85% in three datasets

Empirical Evaluation

- Performance metrics
  - Local ranking accuracy metrics
    - Precision, Recall, F-measure
  - Long-tail promotion metrics
    - LTAccuracy (emphasizes novelty and coverage), Stratified Recall (emphasizes novelty and accuracy)
  - Coverage metrics
    - Coverage, Gini

- Test ranking protocol [Ste13, CKT10]
  - "All unrated items test ranking protocol"
    - Generate the top-N set of each user, by ranking all items that do not appear in the train set of that user
    - Closer to accuracy the user experiences in real-world settings

Algorithms Compared

- Re-ranking frameworks for rating prediction
  - Regularized SVD (RSVD)
  - Resource Allocation (5D)
  - Ranking-based Techniques (RBT)
  - Personalized Ranking adaptation (PRA)
- Report results for two variants of each algorithm

Comparison with re-rankings models for rating-prediction

Dense dataset

- ML-1M

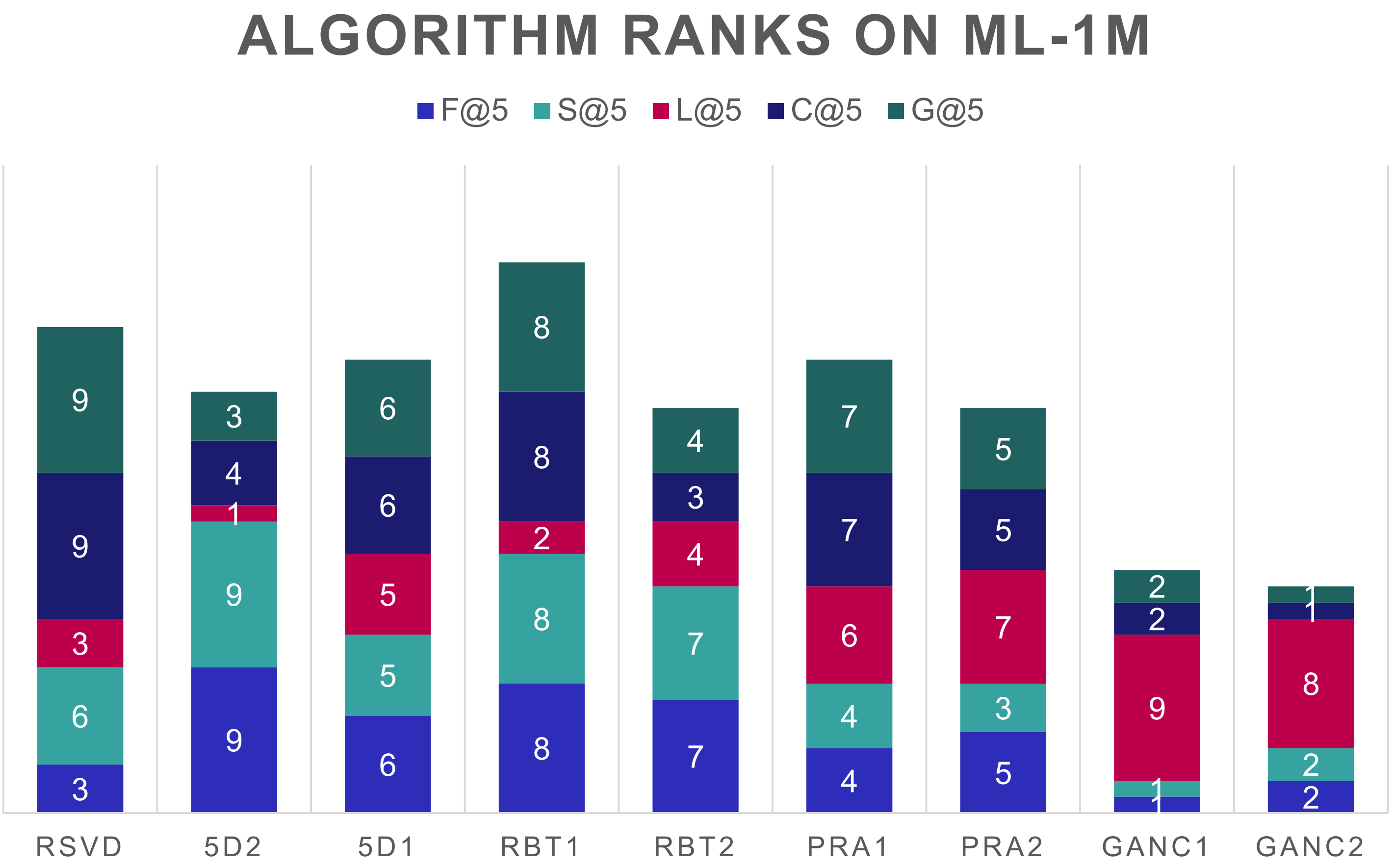
RSVD is base accuracy recommender

Lower height is better

- Corresponds to better rank

GANC outperforms RSVD in all metrics

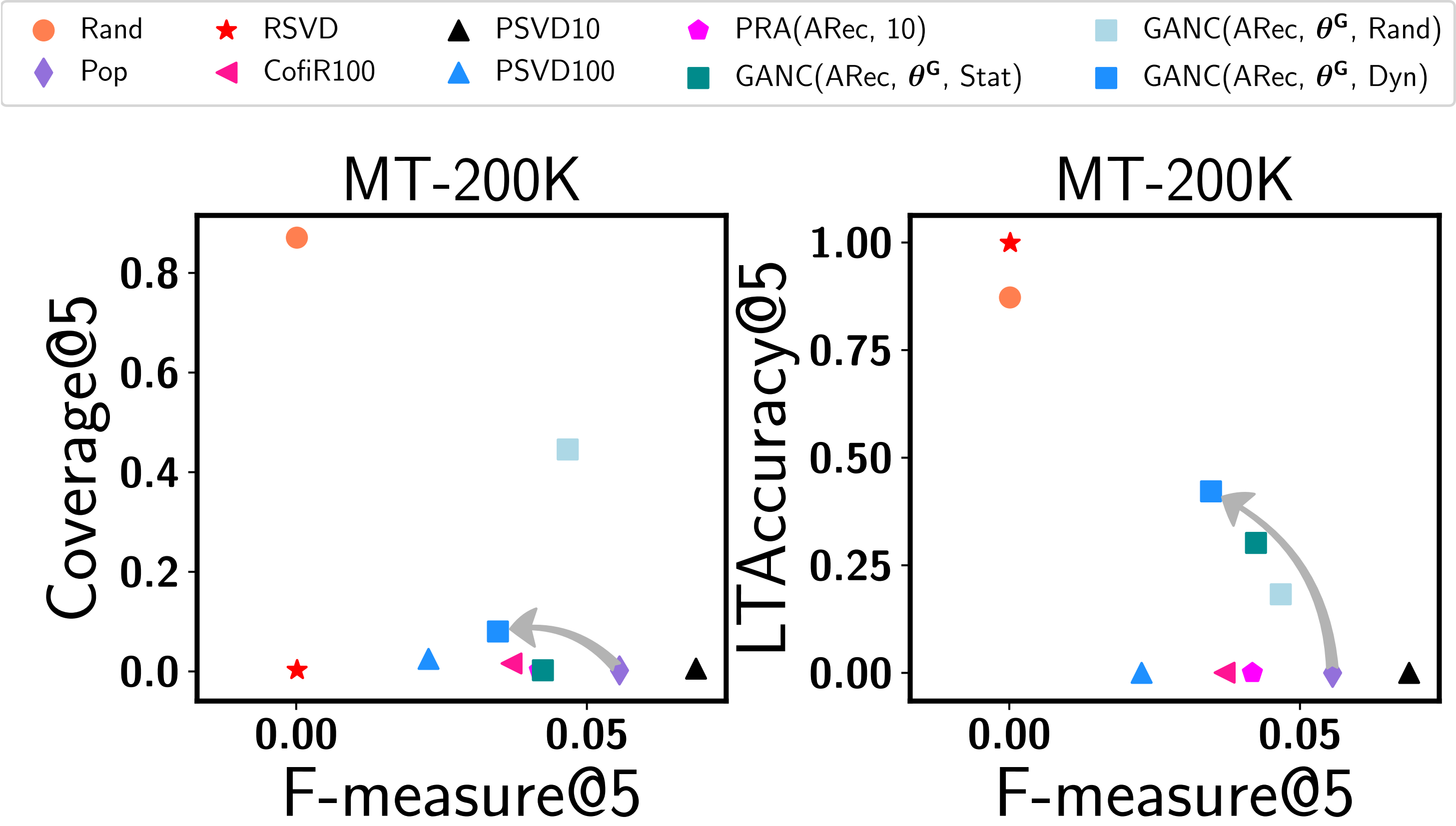
GANC obtains lowest overall performance across 5 metrics



(F)measure@5 (S)tratified Recall@5 (L)Taccuracy@5 (C)overage@5 (G)ini@5

Changing accuracy recommenders explores tradeoffs between accuracy and coverage

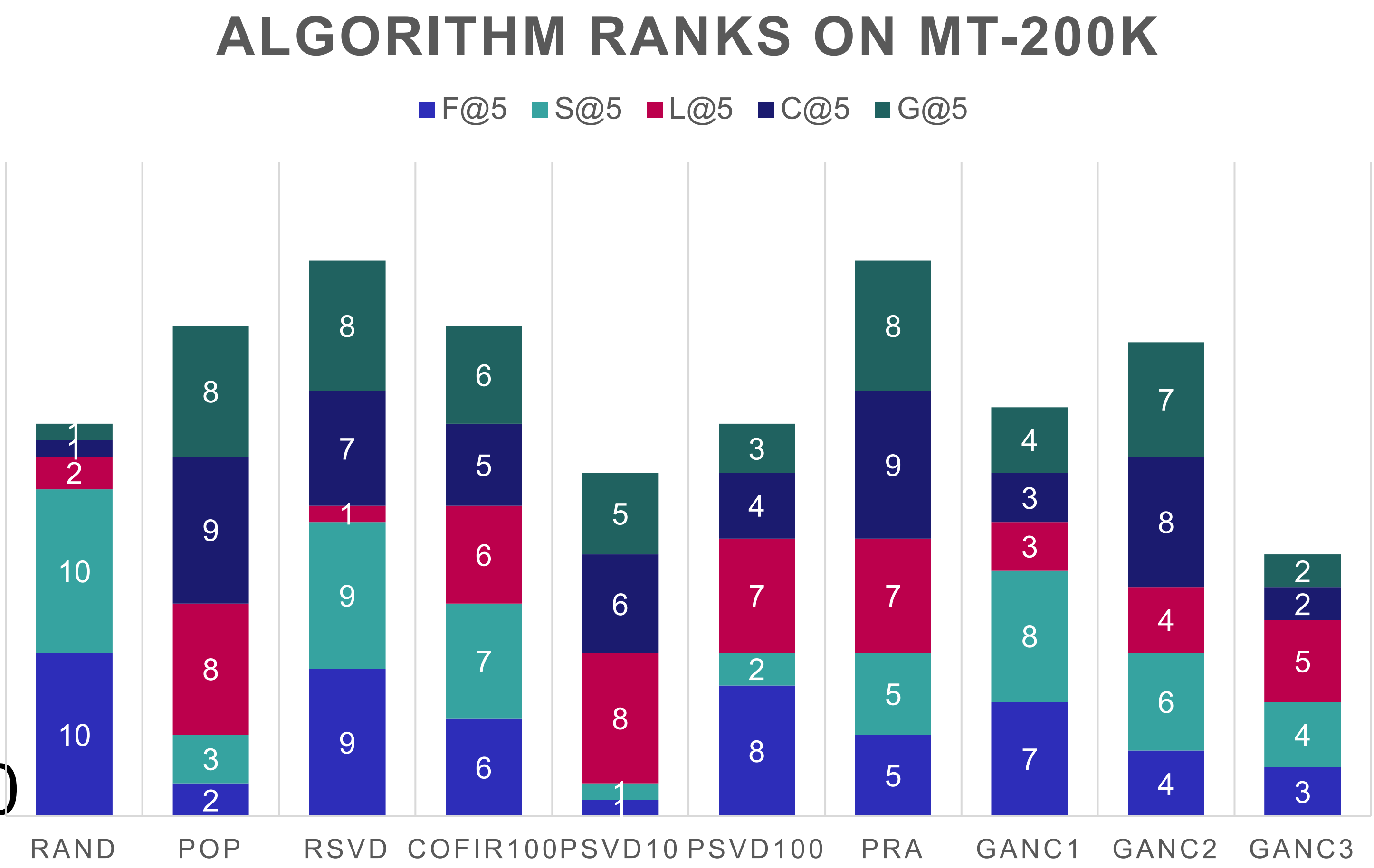
- GANC allows different accuracy recommenders
- Plugging the non-personalized algorithm Pop as accuracy recommender
- Competitive with more sophisticated algorithms like CofiR100



Rachel Pottinger
http://www.cs.ubc.ca/~rap

Comparison with top-N recommendation algorithms

- Sparse dataset
  - MT-200K
- Pop is base accuracy recommender
- Lower height is better
  - Corresponds to better rank
- Three variations of GANC competitive with more PSVD100 and Cofi100



(F)measure@5 (S)tratified Recall@5 (L)Taccuracy@5 (C)overage@5 (G)ini@5

Act 2

- The first part of the talk described how to help users explore data beyond the most popular in a recommendation setting
- Next we'll help users understand the range of possible answers in data aggregated from multiple sources
- Published in Extending DataBase Technology (EDBT) 2015 (joint with Zainab Zolaktaf and Jian Xu)

Looking for climate change: what is the average high temperature across BC for each year?

- Averaging across readings over the entire province seems reasonable
- But there are problems, e.g., inconsistent values

Location	Avg Temp	Date
Burnaby	21	10-June-06
Vancouver	19	11-June-06
...	...	...

City	Temp	Date
Burnaby	21	06/10/06
Vancouver	22	06/11/06
Richmond	18	06/12/06
Richmond	18	06/13/06
...	...	...

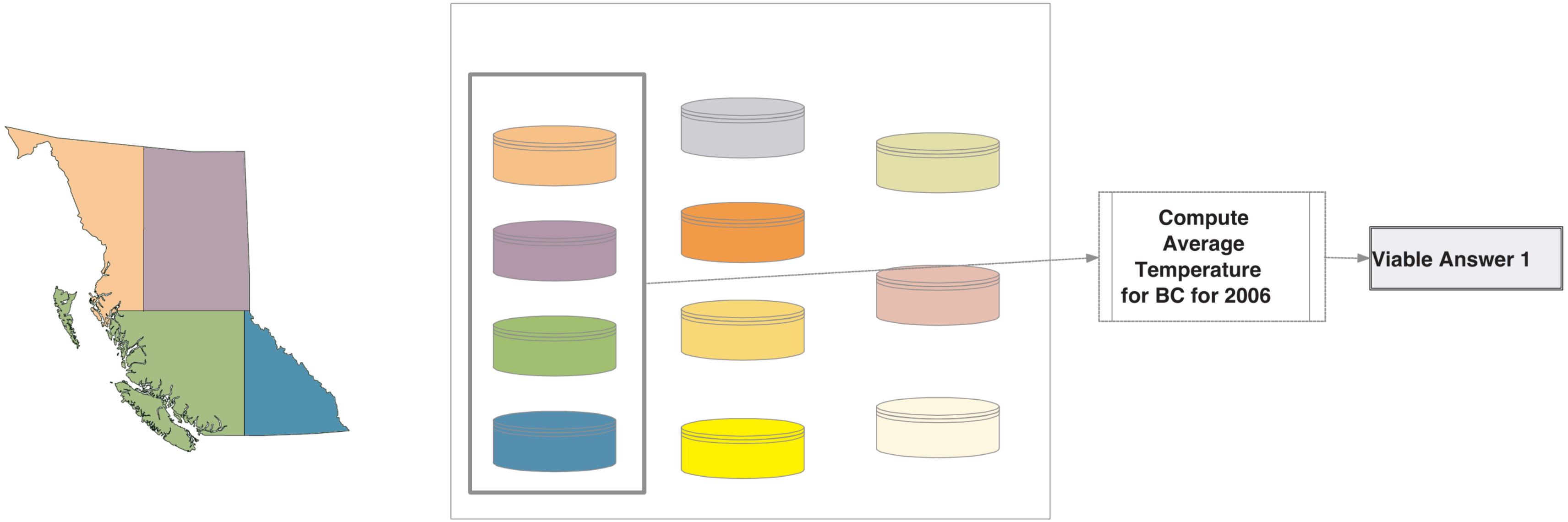
City	Temp	Date	...	Total Rain
Burnaby	19	10-June-06	...	0.2
Vancouver	17	11-June-06	...	0.0
Surrey	15	11-June-06	...	0.0
Vancouver	20	12-June-06	...	1.4
...	...	...	...	...

Location	Temp	Date	Total Snow	Total Rain
Surrey	15	06/11/06	0.0	0.0
Surrey	19	06/12/06	0.0	1.2
...	...	...	...	...

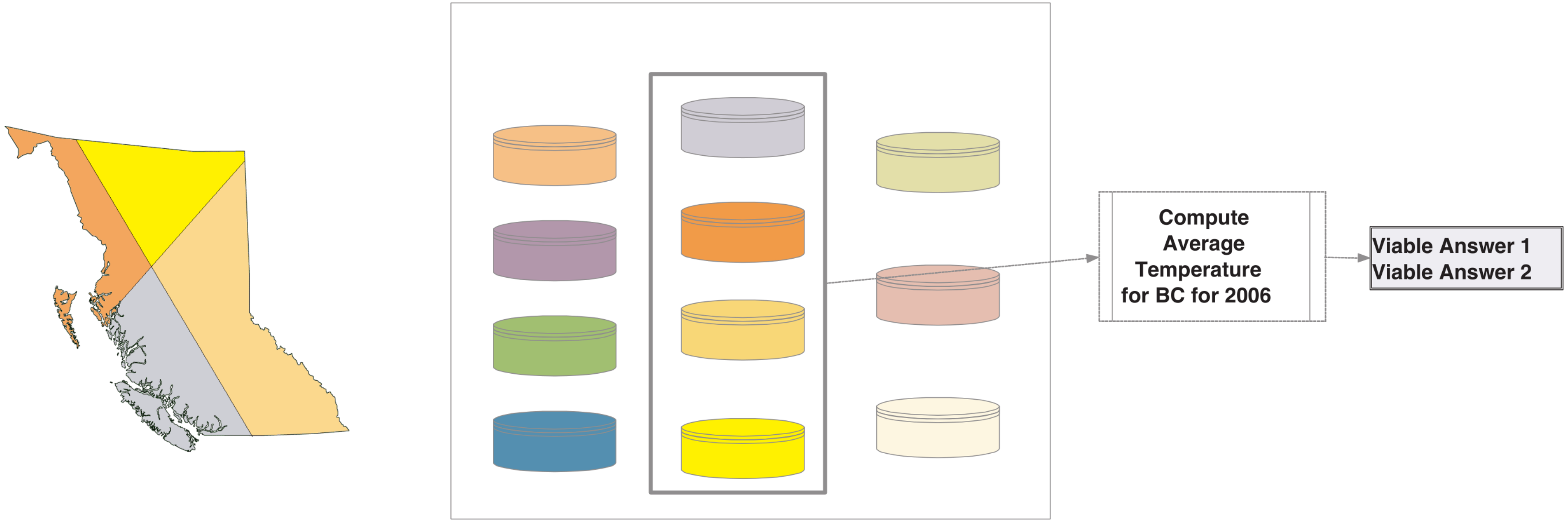
In this work, we helped users understand aggregation query results from multiple sources

- Answering queries in integration contexts requires combining sets of data that are segmented across multiple sources
- Averaging over all the points doesn't work
  - Some data points have duplicates across the sources
  - The duplicates may have different values in the sources
  - Which set of sources and value combinations do we use?
  - We define a ***viable*** answer as a possible answer

Way #1 to compute average temp

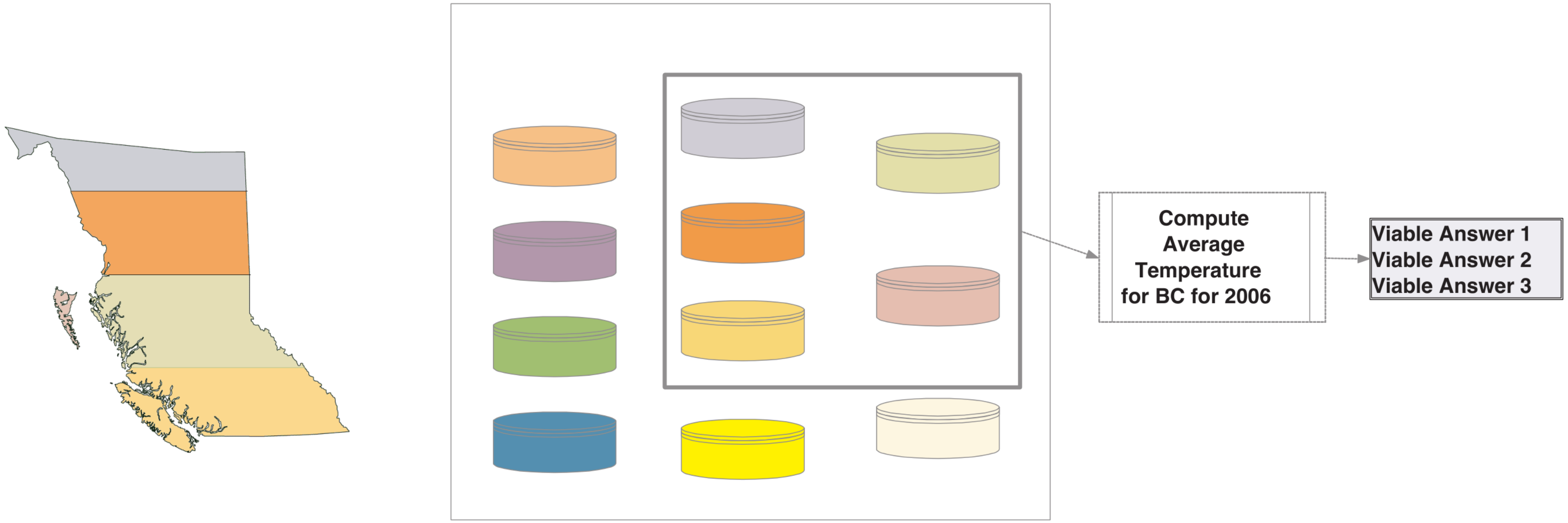


Way #2 to compute average temp



Rachel Pottinger
http://www.cs.ubc.ca/~rap

Way #3 to compute average temp



Contributions of this part

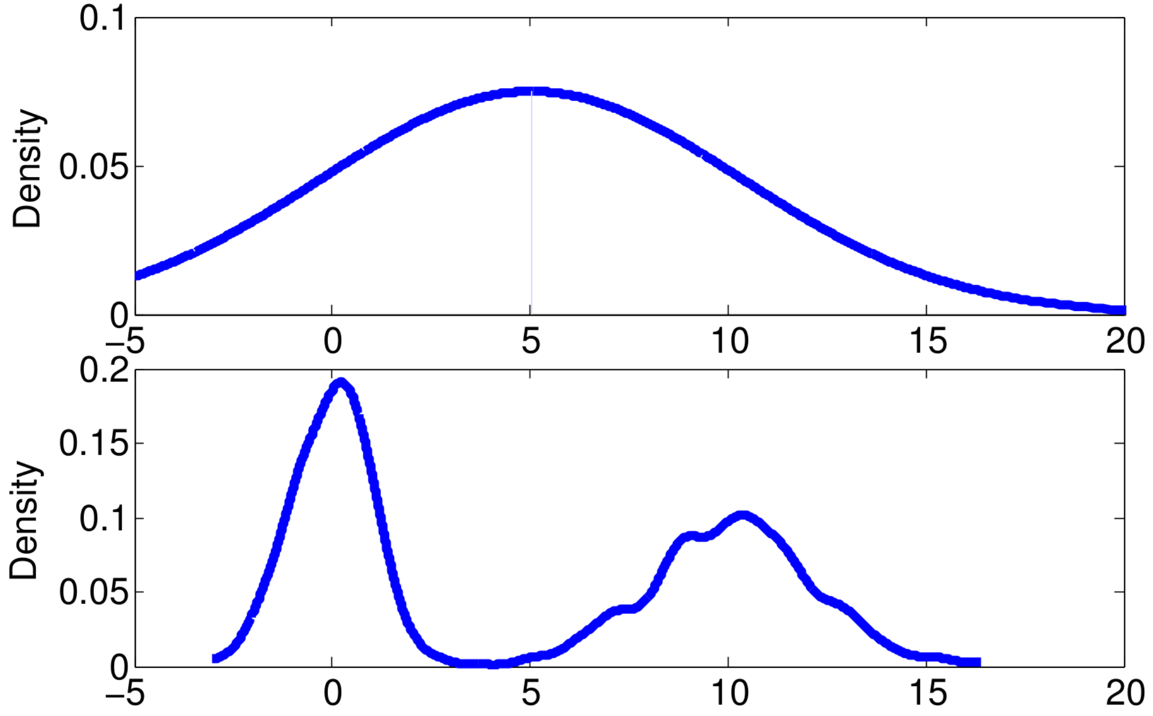
- We define aggregate answers as a distribution of viable answers
- We provide summary statistics and algorithms for the viable answer distribution
  - Key point statistics
  - High coverage intervals
  - Stability score
- We verify the effectiveness of our methods using real-life and synthetic data

Contributions of this part

- We defined aggregate answers as a distribution of viable answers
- We provided summary statistics and algorithms for the viable answer distribution
    - Key point statistics
    - **High coverage intervals**
    - Stability score
- We verified the effectiveness of our methods using real-life and synthetic data

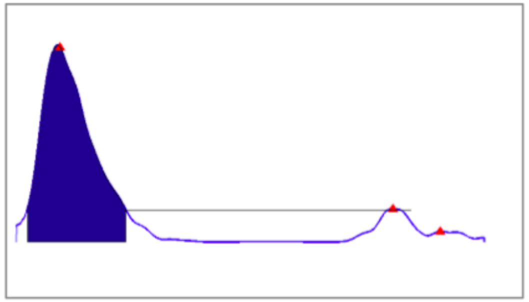
High coverage intervals and optimization

Point statistics such as mean and variance are insufficient

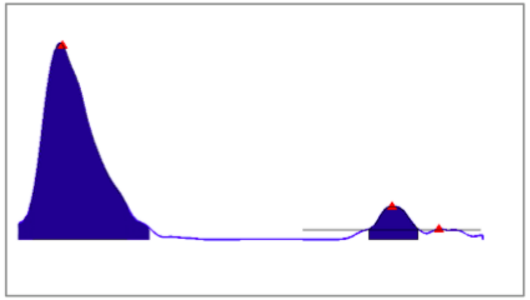


Computing high coverage intervals

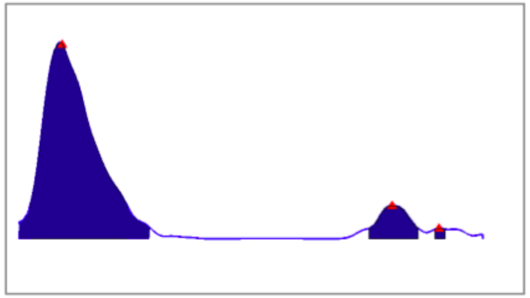
- The ideal, full viable answer distribution is prohibitively expensive to obtain
- We used sampling, bootstrapping and a greedy algorithm to minimize interval length so that coverage of viable answers is above a set threshold



(a) initial high coverage interval finding



(b) intermediate



(c) high coverage intervals that are above our threshold ($\theta\%$)

Act three: ongoing work

Understand: help users understand data provenance (joint work with Omar AlOmeir)

- Database researchers have done a great job of exploring different provenance definitions and how to calculate it
- However, this information is difficult to understand by non-DBA users, which makes it hard for users to trust their data
- We created a desirable set of features for provenance exploration systems and implemented such a system
- Our case study was on Global Legal Entity Identifiers
- We're looking for more data

Understand: help users understand open data (joint work with Janik Andreas)

- Governments are increasingly creating open data sites
- However, these open data sites are hard to use – it's hard to find the data that users are looking for
- We're doing a case study on local data to look at some common open data issues:
  - Quality – granularity and details of available data
  - Metadata and data formatting
  - Availability and completeness

Understand: how can we help users understand why they got the wrong answer?

I'd love to have more people to work with

- If you have data or ideas that you think would fit in, I'd love to talk… especially if you are looking for a postdoc position!