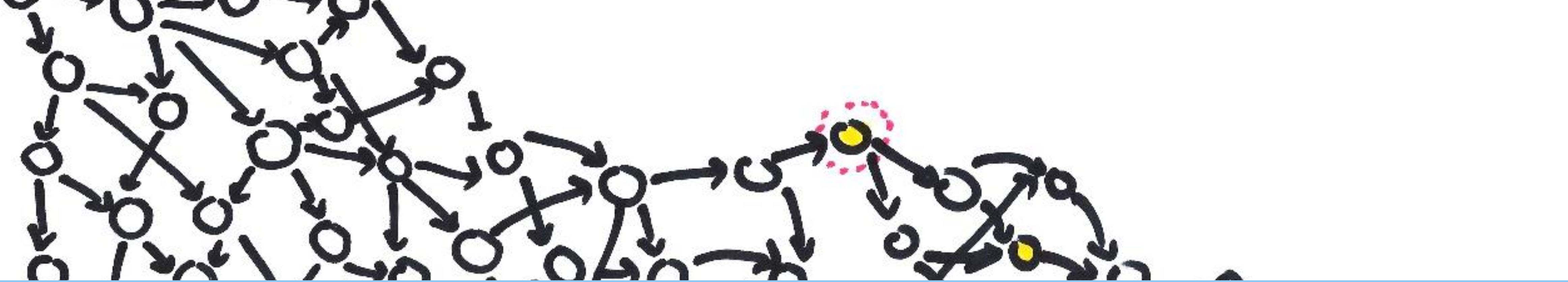
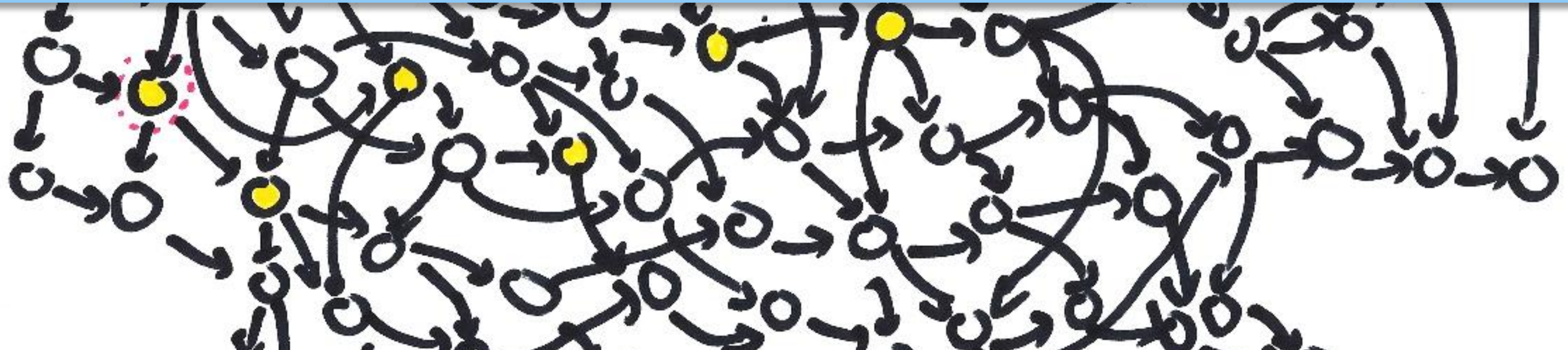


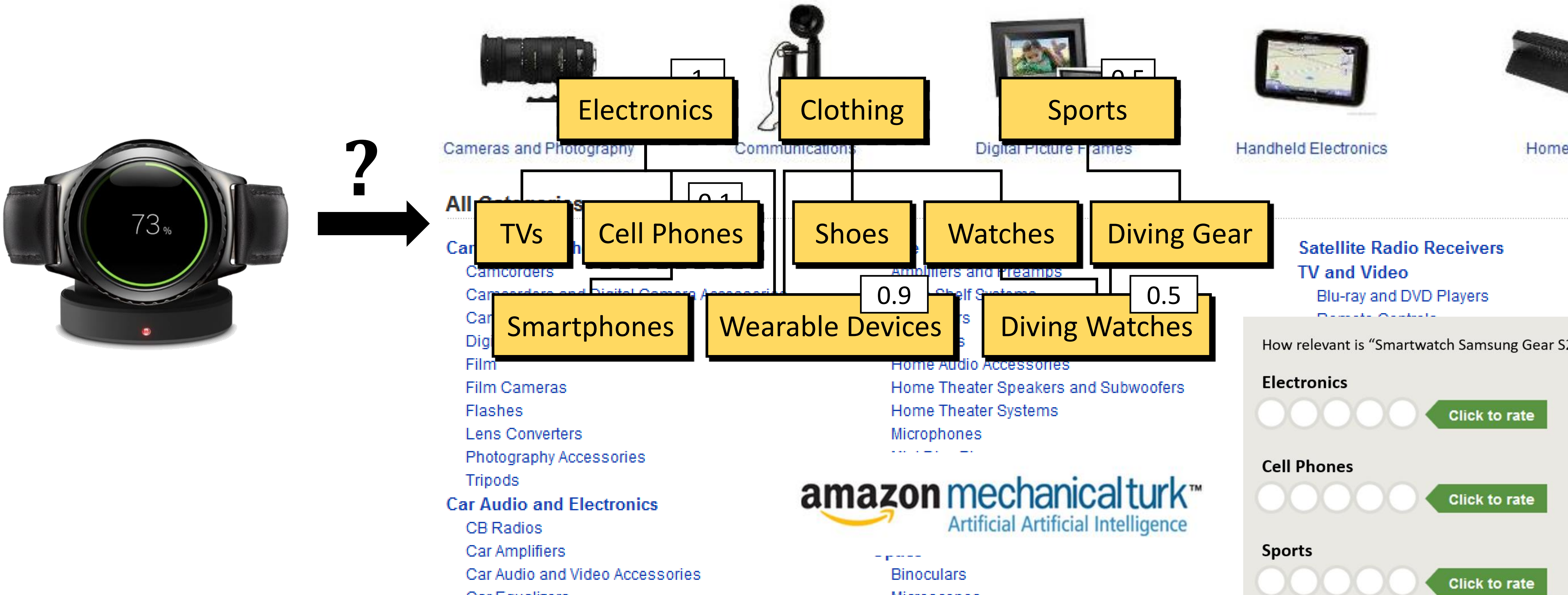
Top-k Querying of Unknown Values under Order Constraints

Antoine Amarilli, Yael Amsterdamer, Tova Milo and Pierre Senellart



Motivation

Find the top-k most compatible (end) categories for a given product



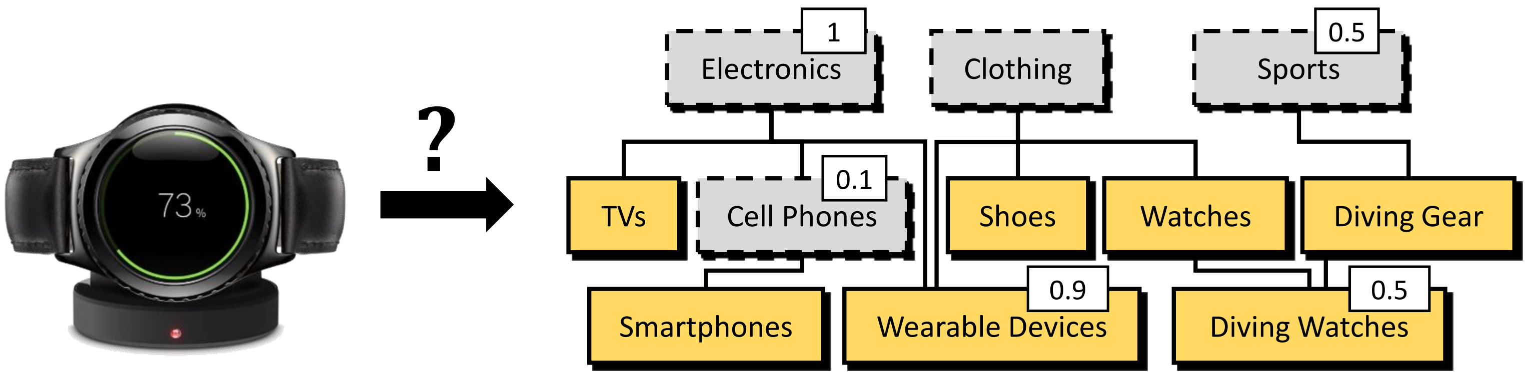
Motivation

We can consider only categories with known compatibility scores
{Wearable Devices, Diving Watches}

Assume we have a structural constraint:
scores for more general categories are always higher

Can we leverage the product hierarchy in estimating the top-k?

E.g., Smartphones are irrelevant; Watches may be better than Diving Watches



General Problem

Order constraints: some roads are busier than others

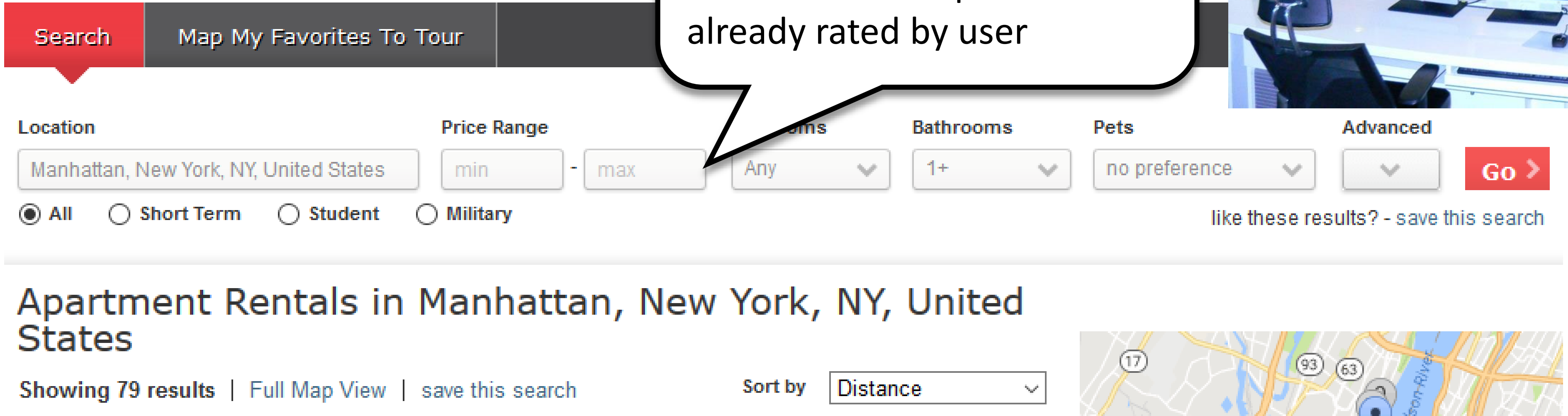
Known values: road parts where vehicle number can be accurately measured

- A set of items
- Some with known values ($x = 0.9$) *"exact-value constraints"*
- Some *order constraints* on (un)known values ($x \leq y$)

  - ➢ Estimate top-k items
  - ➢ Estimate their values

Order constraints: some apartments dominate others

Known values: apartments already rated by user



Previous Work

- A vast body of work on order queries over uncertain data
- Various semantics for queries, including top-k, over uncertain data
- Assuming an independent marginal distribution of unknown values
- However, order constraints naturally lead to dependencies
  - E.g. if a category has many sub-categories, it seems more likely that the product belongs to one of them

E.g. Cormode, Li & Yi 2009;
Haghani, Michel & Aberer 2009;
Jestes et al. 2011;
Soliman, Ilyas & Ben-David 2010

Present work: a foundational study of uncertain top-k computation, accounting for dependencies

- Related work from computational geometry (next)
- Consider general linear constraints or only order constraints, not top-k

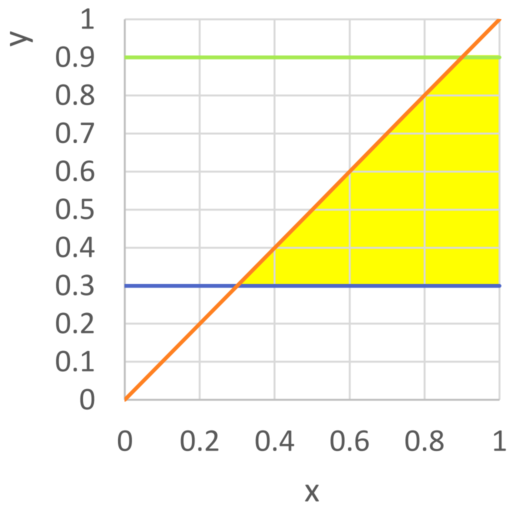
Kannan, Lovász & Simonovits 1997
Lawrence 1991
Maire 2003
Rademacher 2007

Roadmap

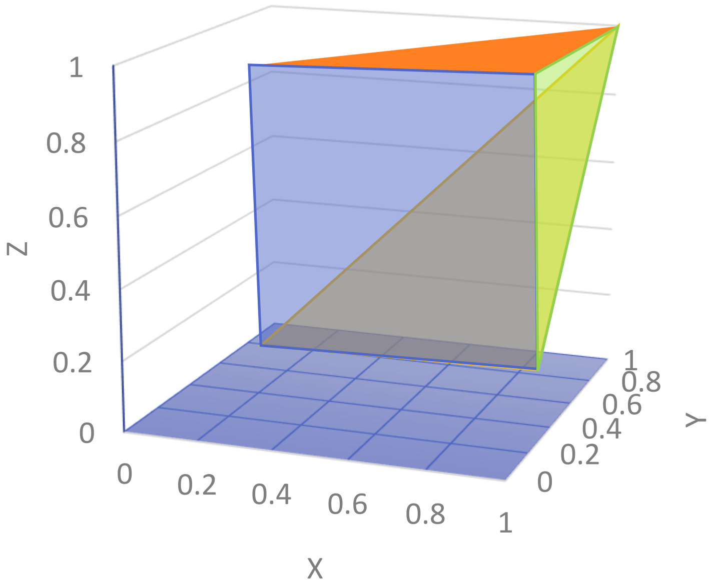
- Motivation
- Model
- General Scheme
- Hardness Results
- Tractable Cases

Unknown Data Values under Constraints

- $\mathcal{X} = \{x_1, \dots, x_n\}$ a set of variables, $\mathcal{X}_\sigma$ selected variables
  - We assume x_i takes values in a bounded, continuous domain, w.l.o.g [0,1]
- $\mathcal{C}$ a set of order and exact-value constraints over $\mathcal{X}$
  - Exact values are rationals
- $\mathrm{pw}(\mathcal{C})$ - the set of possible worlds, all valuations of $\mathcal{X}$ that satisfy $\mathcal{C}$



$x \geq y, 0.3 = w \leq y \leq z = 0.9$



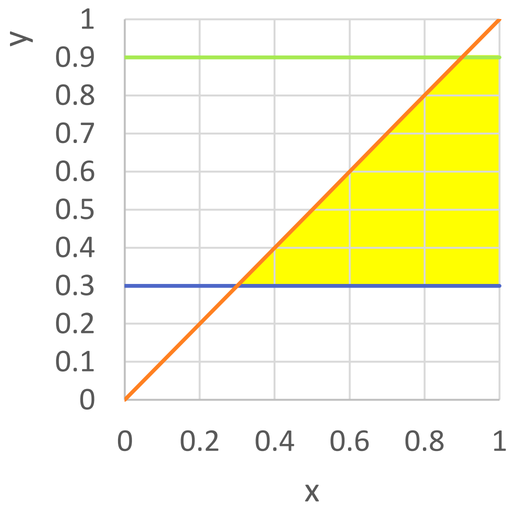
$x \geq y, 0.3 = w \leq y \leq z$

Can be characterized as a convex d-dimensional polytope, $d \leq n$

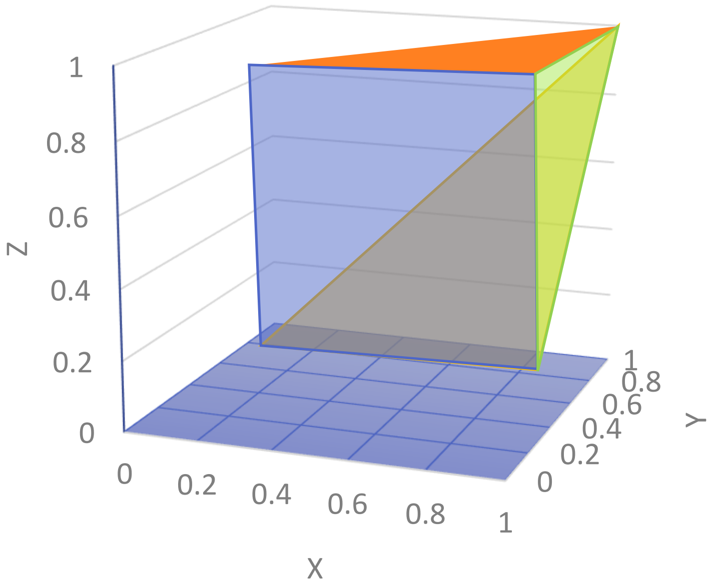
Probability Distribution

- We assume a uniform pdf over $\mathrm{pw}(\mathcal{C})$ - minimum knowledge, all worlds are equally likely
- Base case that can be extended to more complex distributions
- This can be directly defined given the *d-volume* $V(\mathcal{C})$ of the polytope:

$$p(x_1, \dots, x_n) = p(w) := {}^{1}/_{V(\mathcal{C})}$$



$x \geq y, 0.3 = w \leq y \leq z = 0.9$



$x \geq y, 0.3 = w \leq y \leq z$

Top-k Semantics

- We focus on a simple yet powerful one: k with highest expected values
  - Estimates of item values consistent with top-k
  - Other desirable properties

- Independent contribution – interpolation in posets

Interpolation problem: compute the expected value of x under the uniform distribution

Top-k problem: find k items in $\mathcal{X}_\sigma$ with highest expected values, and their expected values, under the uniform distribution

Interpolation and Top-k

Obviously, in our semantics top-k $\leq_P$ interpolation

Q1: Time complexity of interpolation?

Q2: Can we do better for top-k? If we do not return expected values of top-k items?

Top-k requires comparisons, not necessarily exact values

Spoiler: we can show that comparisons can be used to compute exact values

General Algorithm for Interpolation and Top-k

Proposition: if $\mathcal{C}$ implies a total order, then the expected value of $x \in \mathcal{X}$ can be computed in PTIME

Fragment. Distribution independent from other fragments. Marginals follow (rescaled) Beta distribution by connection to order statistics

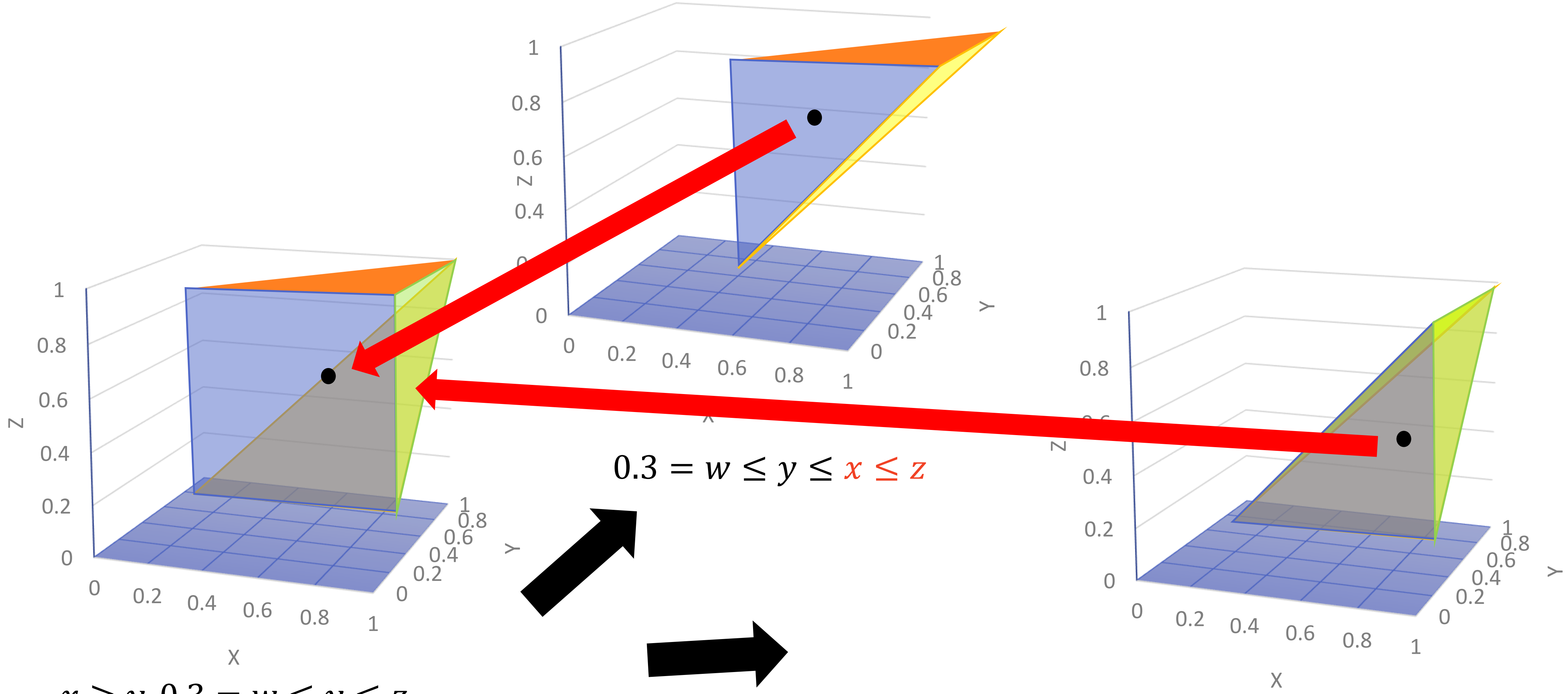
$$\underset{\alpha}{\underset{\shortparallel}{x_0}} \leq x_1 \leq \cdots \leq x_{i-1} \leq \overbrace{\underset{v_i}{\underset{\shortparallel}{x_i}} \leq x_{i+1} \leq \cdots \leq x_{j-1} \leq \underset{v_j}{\underset{\shortparallel}{x_j}}} \leq x_{j+1} \leq \cdots \leq x_n \leq \underset{\beta}{\underset{\shortparallel}{x_{n+1}}}$$

Theorem: for general $\mathcal{C}$, interpolation and top-k are in $\mathrm{FP}^{\#P}$

Orderings of $\mathcal{X}$ compatible with $\mathcal{C}$

- See paper for full algorithm
- General idea: weighted sum of expected values over *linear extensions* of $\mathcal{C}$, weights by the probability of each ordering
- Probability: via d-volume computation
- Nondeterministically sum over linear extensions

Geometrically – Centroid/Center of Mass Computation



$0.3 = w \leq y \leq x \leq z$

$x \geq y, 0.3 = w \leq y \leq z$

$0.3 = w \leq y \leq z \leq x$

Hardness Results – Tight Bounds

Theorem [Rademacher 2007]: interpolation is $\mathrm{FP}^{\#P}$-hard even without exact-value constraints

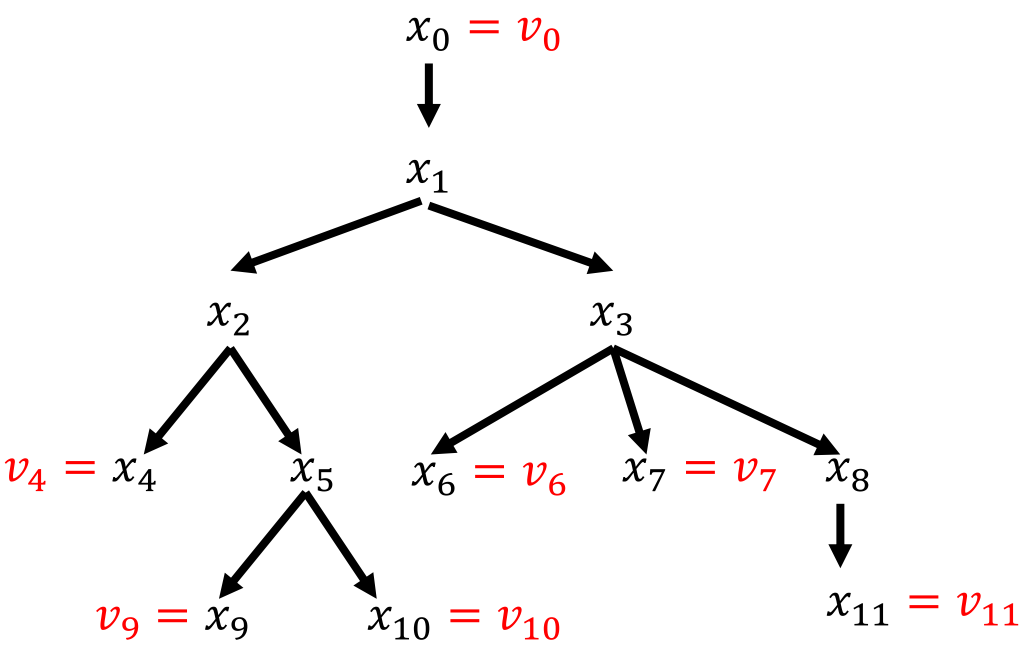
Theorem: top-k is $\mathrm{FP}^{\#P}$-hard even without returning expected values

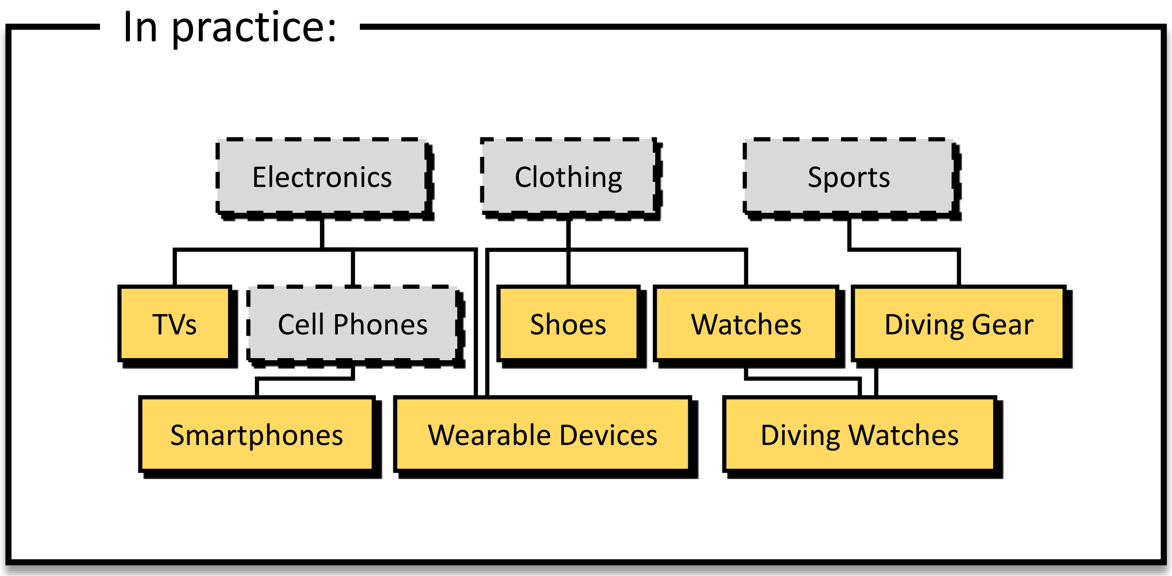
Proof sketch: reduction from interpolation!

- Use top-k to compare the value of x to fresh exact-value
- Use rational number identification scheme using polynomial # comparisons

Tractable Cases

- Recall: for total orders interpolation and top-k are in PTIME
- We now extend to tree-shaped constraint sets (exponentially many linear extensions)





Interpolation and Top-k for Trees

Theorem: if $\mathcal{C}$ is tree-shaped, we can compute $V(\mathcal{C})$ in time $O(|\mathcal{X}^2|)$

- Bottom-up processing, propagating a piecewise polynomial function for the volume of subtree based on root's parent value
- Complexity proof by induction

Theorem: if $\mathcal{C}$ is tree-shaped, we can compute the marginal of x in time $O(|\mathcal{X}_{exact}| \cdot |\mathcal{X}^2|)$

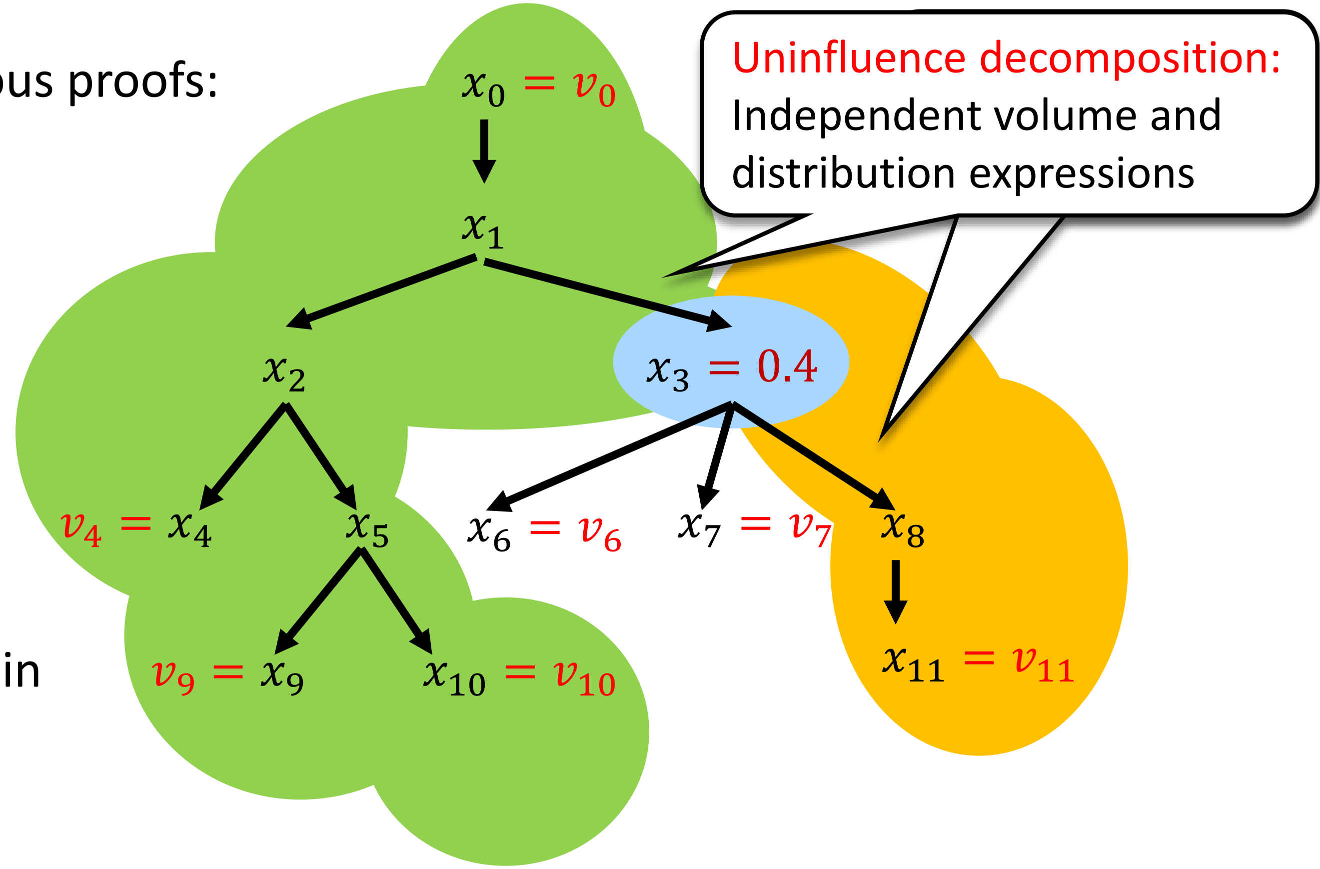
- A similar bottom-up scheme
- Computing the pdf as a function from v to $V(\mathcal{C}_{x=v})$
- The additional $|\mathcal{X}_{exact}|$ factor is intuitively due to the pieces of the polynomial

Splitting Lemma

Splitting Lemma

A generalization of fragments, used in the previous proofs:

Proof by a bijection over possible worlds



Corollary: interpolation and top-k can be solved in PTIME for any constraint set decomposable to (reverse-)tree-shaped constraint sets

Other Variants

In presence of unknown values, there are multiple possible semantics

1. k with highest expected values
2. k with highest expected ranks
   - Related to expected values
3. Most likely ranked sequence of size k
4. k variables most likely to be among top-k
   - Even though defined independently for variables

Do not coincide with our def even for k=1

Top-k $\not\subseteq$ top-(k+1)

Related Work (Selected Subset)

Queries over uncertain data:

- G. Cormode, F. Li, and K. Yi. Semantics of ranking queries for probabilistic data and expected ranks. In *ICDE*, 2009.
- P. Haghani, S. Michel, and K. Aberer. Evaluating top-k queries over incomplete data streams. In *CIKM*, 2009.
- J. Jestes, G. Cormode, F. Li, and K. Yi. Semantics of ranking queries for probabilistic data. *IEEE TKDE*, 23(12), 2011.
- M. A. Soliman, I. F. Ilyas, and S. Ben-David. Supporting ranking queries on uncertain andincomplete data. *VLDB J.*, 19(4), 2010.

Computational geometry:

- R. Kannan, L. Lovász, and M. Simonovits. Random walks and an $O(n5)$ volume algorithm for convex bodies. *Random Struct. Algorithms*, 11(1), 1997.
- J. Lawrence. Polytope volume computation. *Mathematics of Computation*, 57(195), 1991.
- F. Maire. An algorithm for the exact computation of the centroid of higher dimensional polyhedra and its application to kernel machines. In *ICDM*, 2003.
- L. A. Rademacher. Approximating the centroid is hard. In *SCG*, 2007.

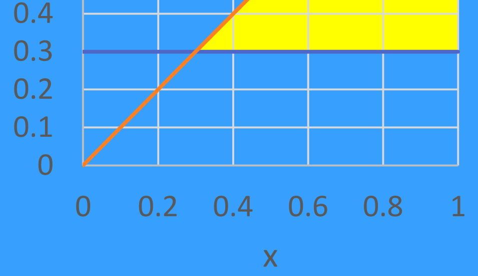
Summary

A foundational study of top-k over uncertain data

- Top-k expected values
- Uniform distribution over possible worlds
- General problem $\mathrm{FP}^{\#P}$-complete via a concrete computation scheme
- Tractable cases for constraints decomposable to trees

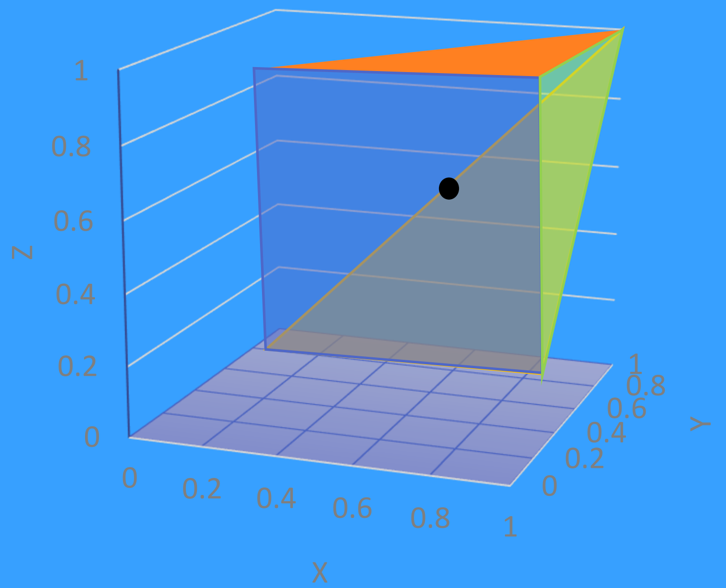
Future Work

- Additional tractable cases (bounded treewidth?)
- Interactively choosing the next exact value to fetch
- Different prior distributions



$x \geq y, 0.3 = w \leq y \leq z = 0.9$

Thank you!



$x \geq y, 0.3 = w \leq y \leq z$

Questions?