Approximating a distribution using weight queries

ICML 2021

Nadav Barak & Sivan Sabato

Ben-Gurion University of the Negev
Department of Computer Science

Motivation: learning without a random sample

Source Hospital

- Full access to patients records

Different patient distributions

Target Hospital

- No direct access to patients records
- No random sample
- Access only via database counting queries

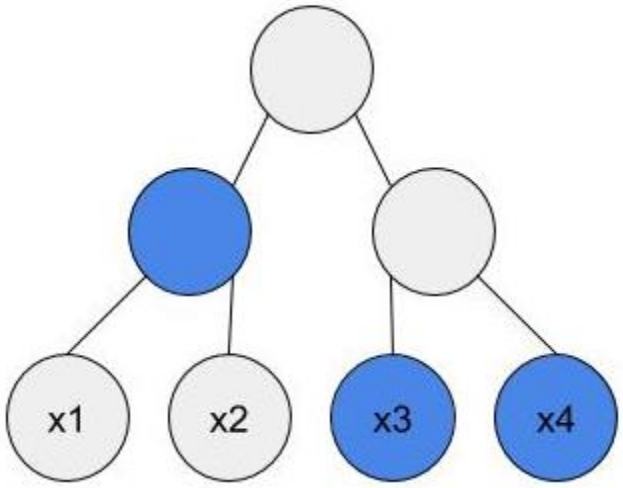
Can we successfully perform learning tasks under this setting?

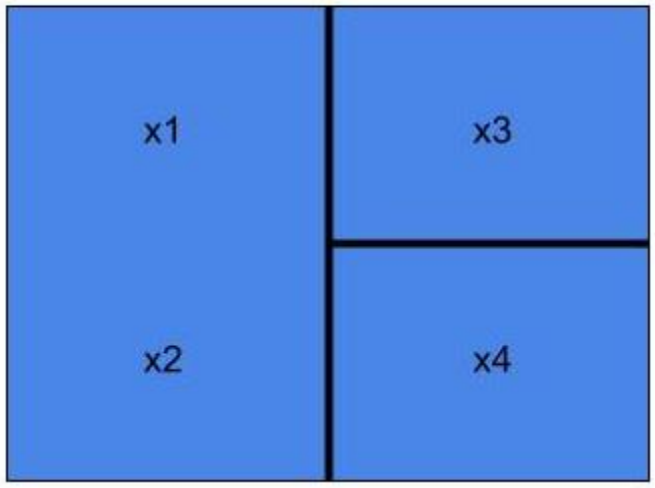
Problem Setting

- Input: Dataset of domain examples
- No random sample from target distribution
- Access to the target distribution **only via weight queries**
- Goal: Reweight the dataset to match the target distribution
  - The reweighted dataset can be used for various machine learning tasks

A structure over the input dataset

- Which weight queries are allowed?
- Input: A binary tree whose leaves are the dataset examples.
- Each node represents a queryable subset of the domain.
- A **pruning** of the input tree induces a partition of the domain.





Main result

- A new active querying algorithm (AWP) for reweighting the source dataset to approximate the target distribution.
- AWP finds a small pruning that defines a reweighting of the source dataset.

Theorem: Let K be the requested pruning size. AWP finds a near-optimal reweighting of the source data set for the target distribution.

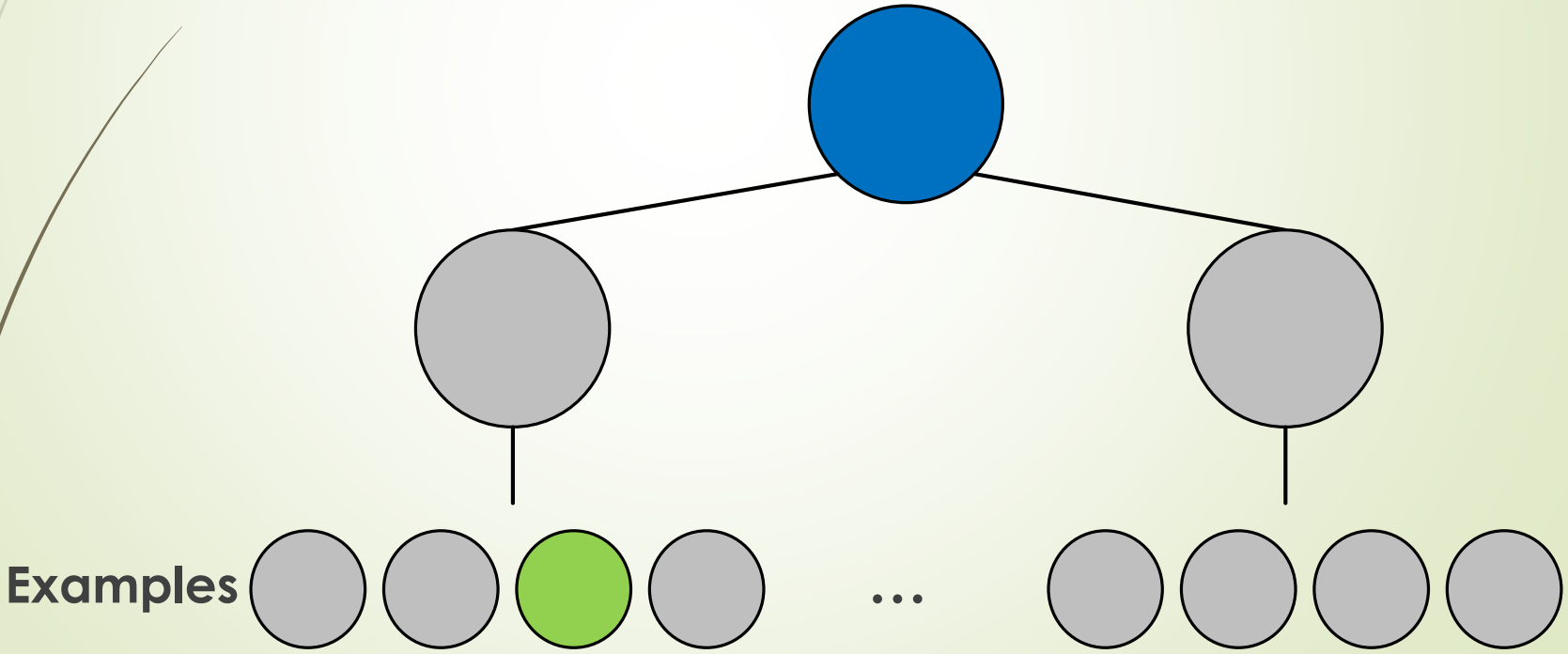
- The optimality approximation factor is $O(\log(K))$
- The number of weight queries is $O(K^3)$

The AWP Algorithm

- We define a new quantity called **discrepancy** that measures the weight mismatch on a subset of the domain.
- AWP uses adaptive sampling of weight queries and a new discrepancy estimator.
- AWP runs an iterative top-down procedure to achieve a low discrepancy pruning of the input tree.

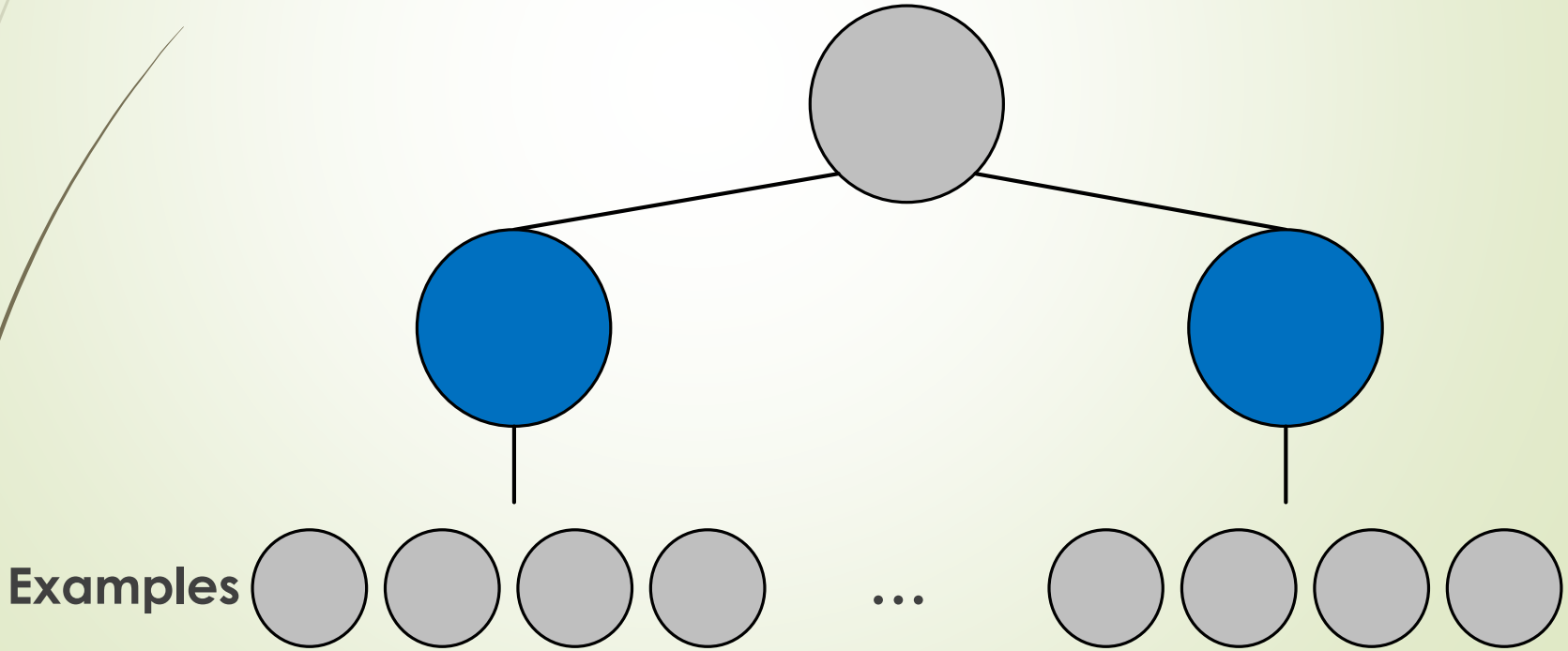
The AWP Algorithm

- Phase 1:
  - Select a node in current pruning;
  - Query the weight of a random example under this node;
  - Repeat until identifying a relevant node to split.

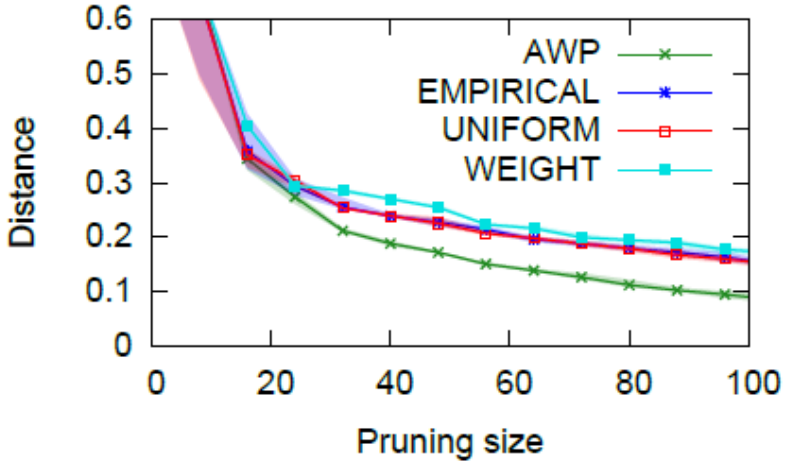


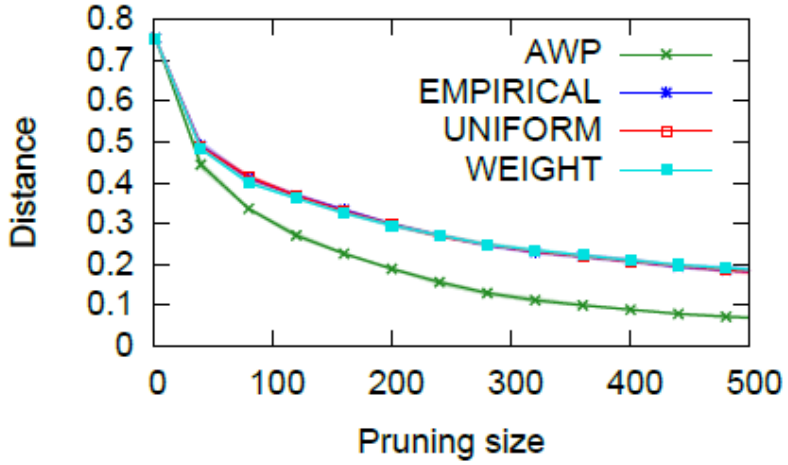
The AWP Algorithm

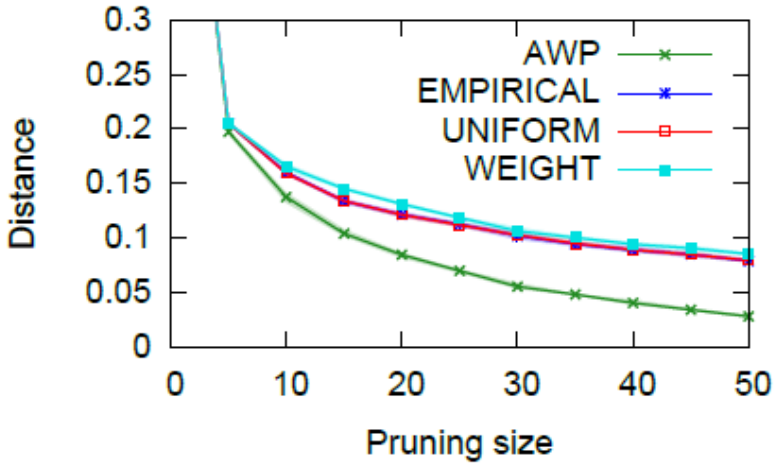
- Phase 2: Replace the node found in phase 1 with its children; This increases the pruning size by 1.
- Repeat both phases until a stopping condition is reached.

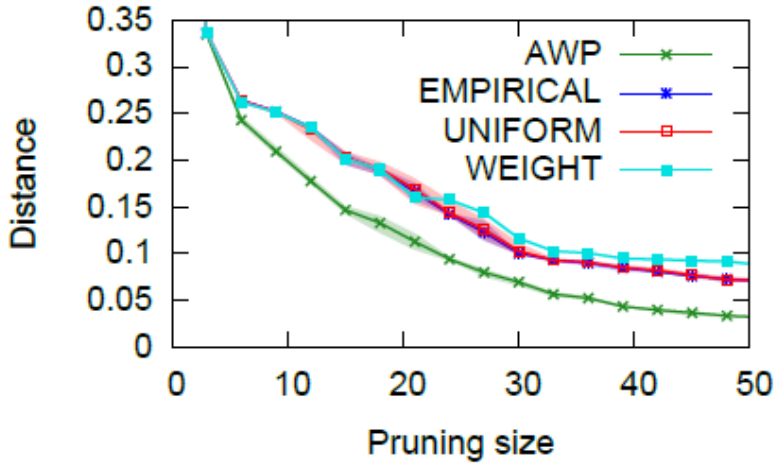


Experiments









Summary

- A novel setting in which weight queries are the only access to the target distribution.
- The AWP algorithm finds a reweighting of the dataset by finding a suitable pruning.
- Theoretical guaranties for the quality of the output reweighting and the number of weight queries.
- Experiments demonstrate the success of AWP.

Thank you for listening!