Pythia: Algorithm Selection for Differential Privacy

Ios Kotsogiannis[†], Ashwin Machanavajjhala[†] (PI), Gerome Miklau[✣] (PI), Michael Hay[‡] (PI)

† Duke University, ✣ University of Massachusetts Amherst, ‡ Colgate University

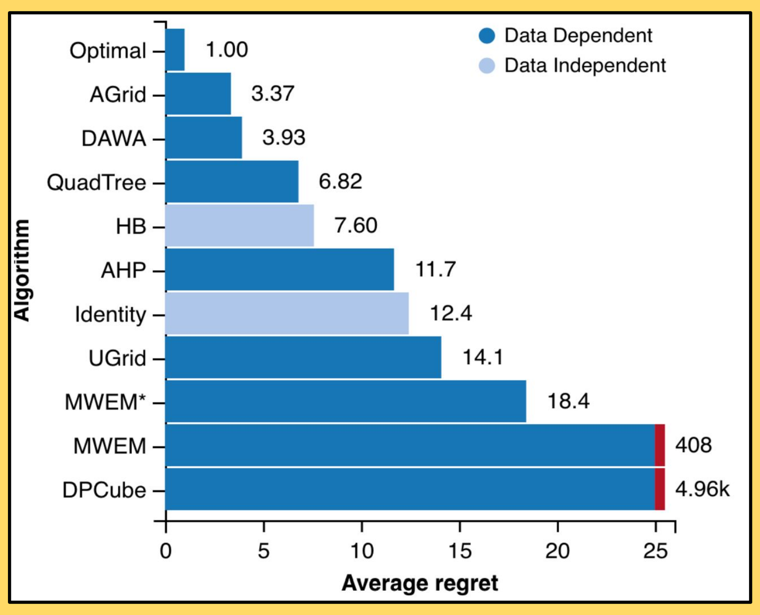
Introduction

Recent differentially private algorithms have allowed for a significant improvement of error rates by adapting to properties of the input data. These so-called data-dependent algorithms have *different error rates* for *different inputs*. There is now a complex and growing landscape of algorithms, without a clear winner that can offer low error over all datasets. As a result, the best possible error rates are not attainable in practice, because the data curator cannot know which algorithm to select prior to actually running the algorithm. This motivates the problem of *Algorithm Selection*. We propose ***Pythia*** an end-to-end differentially private meta-algorithm for Algorithm Selection. Using Pythia, data curators do not have to understand available algorithms, or analyze subtle properties of their input data, but can nevertheless enjoy reduced error rates that may be possible for their inputs.

Choice Matters

In *"Principled Evaluation of Differentially Private Algorithms using DPBench"*, Hay et. al demonstrated that the accuracy of differentially private algorithms depends on characteristics of the input i.e., the workload of queries, the sensitive dataset, and the privacy parameter.

- *"One size fits all"* does not apply to DP algorithms.
- The curator is burdened with the *dilemma of choice*.
- Input characteristics are not known a-priori. Even if they were using them might violate privacy.



Algorithm Selection

Given an input (**W**, **x**), a desired privacy parameter ϵ and a set of DP algorithms $\mathcal{A}$ our goal is to select an algorithm A* from $\mathcal{A}$ to answer **W** on **x**.

Desiderata:

a) ***Differentially Private*** - any use of input data must be included in an end-to-end privacy guarantee.

b) ***Competitive*** - solutions to Algorithm Selection should offer low error rates for arbitrary inputs.

c) ***Agnostic*** - algorithms in $\mathcal{A}$ should be treated as black boxes - i.e., the curator shouldn't worry about the internal workings of different algorithms.

We measure competitiveness via *regret* - the ratio between an algorithm's error and the best possible error on that input.

Baseline Approaches

Blind Choice

Arbitrarily choose an algorithm and use it to answer all the input tasks.

Private | Agnostic | Competitive

Informed Decision

Run all differentially private algorithms on target input, check their empirical error rates, and choose the one with the least error.

Private | Agnostic | Competitive

Private Informed Decision

Follow Informed Decision, but add appropriate noise to the computed error rates.

Private | Agnostic | Competitive

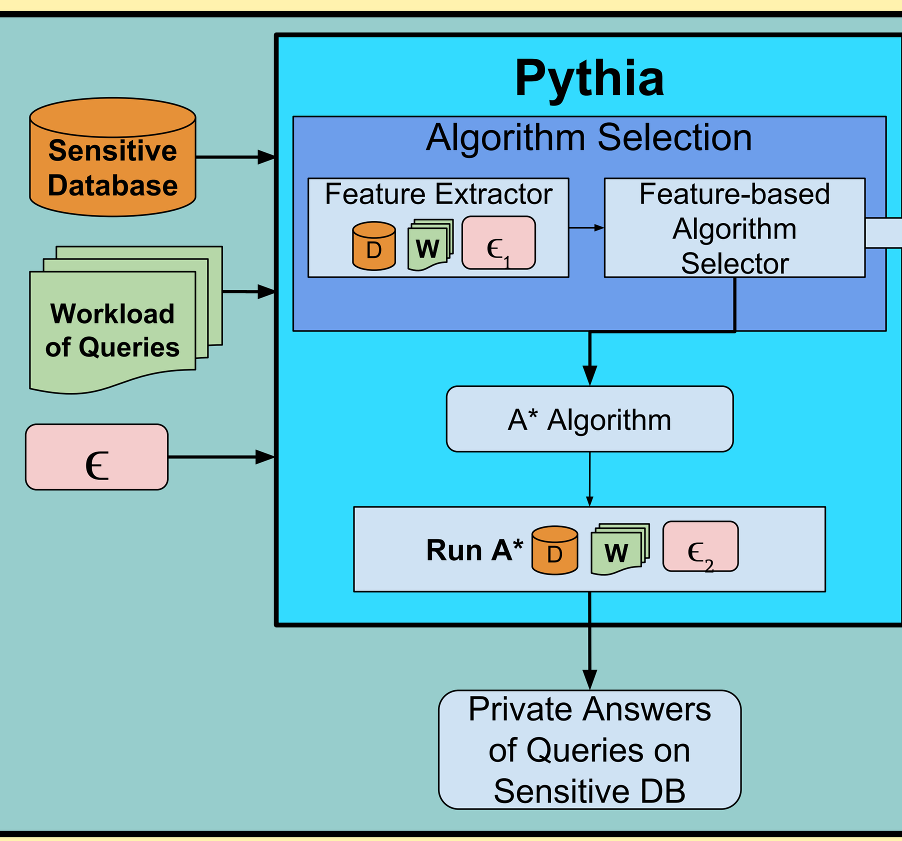
Pythia Overview

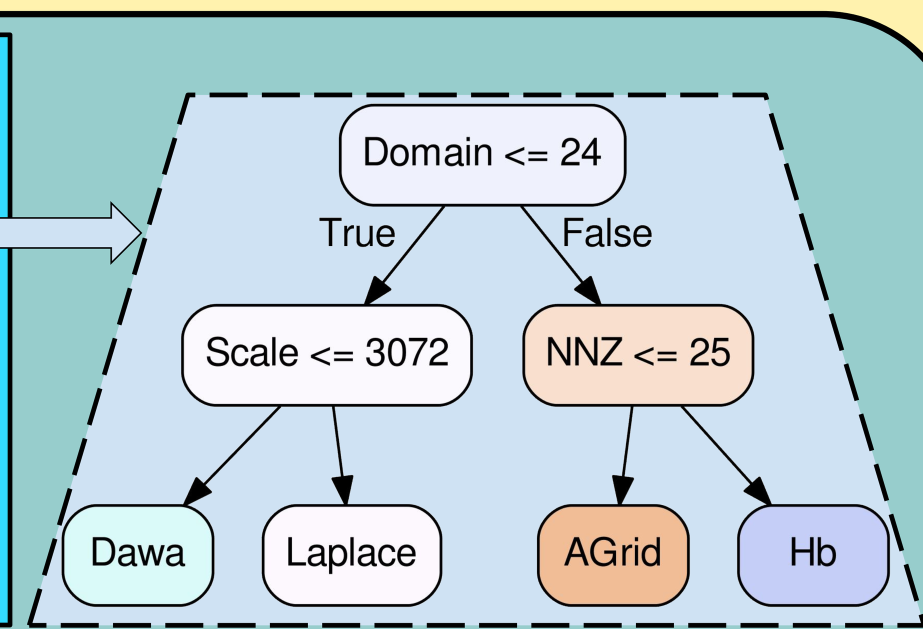
Pythia is an end-to-end differentially private meta-algorithm for achieving near-optimal error rates using a suite of available privacy algorithms.

Pythia works in *three* steps.

1. Extracts a set of feature values from the given input.
2. Applies the extracted features on feature-based algorithm selector (FAS).
3. Runs the selected algorithm A* on the given input.

The success of Pythia depends on the FAS. We developed Delphi, a *task* specific method that constructs a FAS.





An example of a Feature-based Algorithm Selector

Delphi: Learning a Feature-based Algorithm Selector

Delphi is the process that builds a FAS that is used by Pythia for performing algorithm selection. Our goal with Delphi is to produce a FAS that is: (a) *efficient*, (b) *highly interpretable*, and (c) *reusable*. Towards these goals, Delphi's design is based on the following key ideas:

Data Independence

Promotes reusability and ease of use.

Rule-based Selector

Allows for high model interpretability, robust performance w.r.t. outliers, and fast runtime.

Regret-based Learning

Standard classification treats all mispredictions as equally bad. Delphi does not distinguish between algorithms with similar regrets (since these would all be good choices).

Experimental Setup

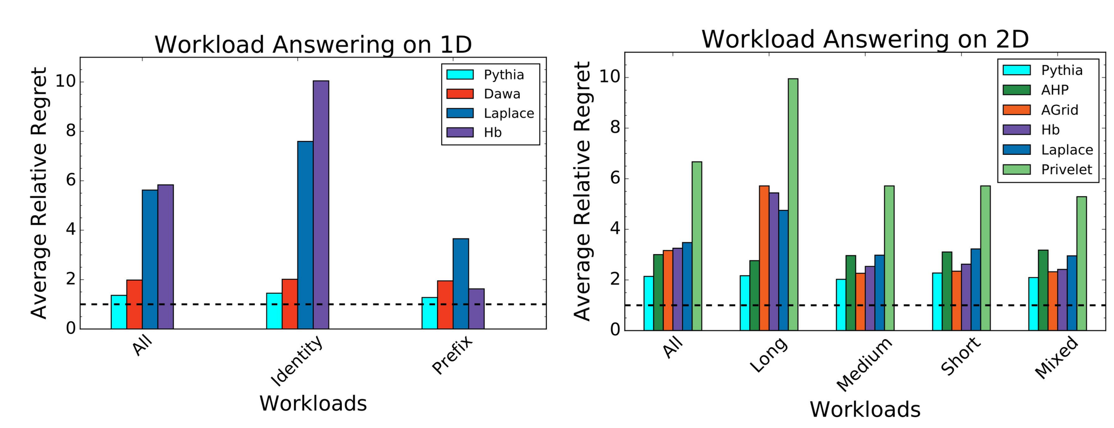
We consider 2 *tasks*: 1D and 2D range queries. We instantiate Pythia once per task. We consider 2 use cases: range query answering, and building a naive bayes classifier. Pythia's training is always done on a disjoint set of input datasets than the ones that it is evaluated on. Overall we have 1080 inputs for the 2D task and 980 for the 1D task.

Range Queries: We evaluate the performance of DP algorithms for different range query workloads for all the inputs of each task.

Naive Bayes Classifier: In an NBC a number of histograms are extracted from the dataset, which are used to perform classification. Extracting these histograms via Differential Privacy ensures the overall privacy of the system.

Use Case: Range Queries

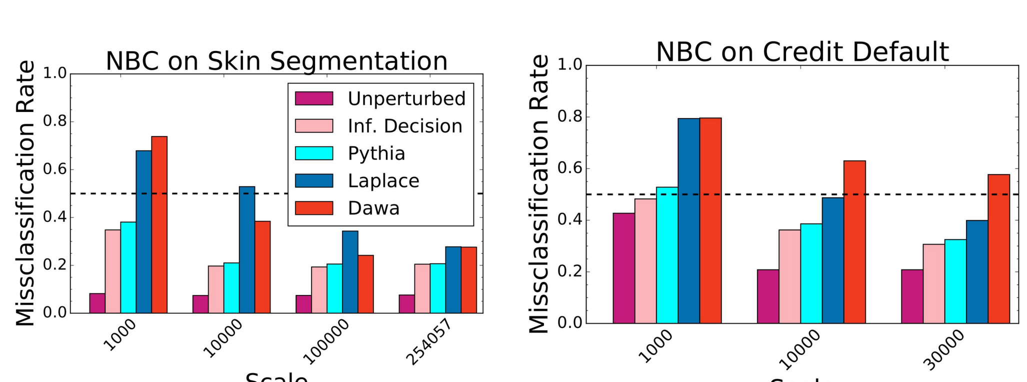
Due to the differences of the inputs Pythia always outperforms the best 'blind choice' strategy for both the 1D and the 2D tasks.



Use Case: Naive Bayes Classifier

We evaluate the performance in terms of misclassification rate. Each bar in the plots denotes a different strategy of extracting the sensitive histograms.

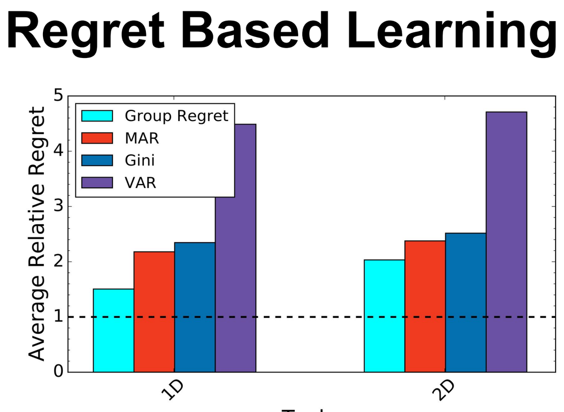
Due to the heterogeneous nature of the extracted histograms Pythia achieves a near-optimal misclassification rate for a differentially private NBC.



Skin Dataset: 7 histograms extracted

Credit Dataset: 47 histograms extracted

Regret Based Learning



Our novel learning method outperforms the standard method (Gini) as well as other methods we tested against.

Beyond Pythia

Pythia is just one example of error optimization in Differential Privacy for low dimensional inputs. ***Building a universal optimizer for answering workloads under differential privacy*** is a natural forward direction.

Automatic extraction of features from inputs will provide insights into the conditions under which existing algorithms work well, and may facilitate the development of new specialized algorithms.

Interested in meeting the PIs? Attach post-it note below!