Nimbus: Model-based Pricing for Machine Learning in a Data Marketplace

L. Chen[1], H. Wang[1], L. Chen[2], P. Koutris[1], A. Kumar[3]

[1]University of Wisconsin Madison

[2]University of Pennsylvania, [3]University of California, San Diego

1. Executive Summary

Machine Learning: Critical for data analytics systems.

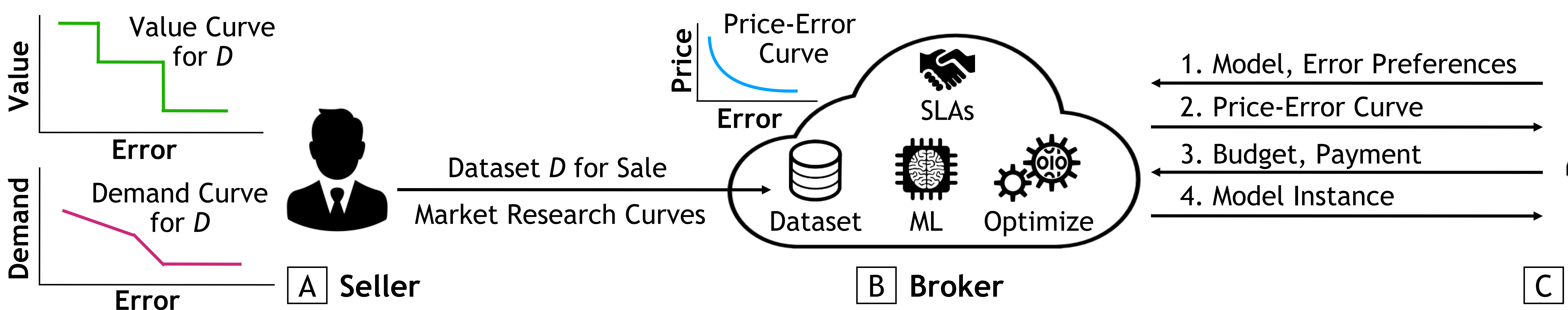
Google Microsoft ORACLE amazon IBM intel

Problem:
- Loss of Accessibility (Buyer)
- Loss of Revenue (Seller)

Our Idea: Sell ML models directly ⇓ Trade-offs between price and model accuracy

Data for ML: Many works for ML accuracy, efficiency, scalability, etc.
But little work on cost of data acquisition
Existing approach: buy the whole/ a fixed sample of datasets

This work:
- A formal framework describing the desiderata
- An instance using noise injection that achieves all the desiderata
- Extensive experiments and executable demo with GUIs



2. Example and Motivation

Task Y	Feature X		
Income	Age	Race	Sex
?	?	?	?
?	?	?	?
?	?	?	?
…	…	…	…

Dataset Price: $1000

Buyer: I need to write an article for a journal about relation between demographics and economic factors. The data is available online, but it is too expensive.

Buyer: I may not need the dataset. A ML model with not too low accuracy is enough.

Buyer: Can I purchase such a model with low cost?

Seller: Generate model for different buyers' accuracy requirement?

Seller: Determine price to maximize my revenue?

Lin Reg Model Error: 20% Price: $100

3. Nimbus: Our Proposed Approach

Model Generation: $h_\lambda^\delta(D) = \mathcal{K}(h_\lambda^*(D), w) \in \mathcal{H}$, $w \sim \mathcal{W}_\delta$

Pricing Function: $p_{\epsilon,\lambda}(\delta, D)$ $\mathcal{H}$: hypothesis space D: dataset

δ: (noise) control parameter λ, ϵ: training/testing error function

3.1 Pricing Function Desiderata

- **Non-negative:** $p_{\epsilon,\lambda}(\delta, D) \geq 0$

- **Error-Monotone:**

$$\mathbb{E}\left[\epsilon(\hat{h}_\lambda^{\delta_1}(D), D)\right] \leq \mathbb{E}\left[\epsilon(\hat{h}_\lambda^{\delta_2}(D), D)\right]$$

$$\Downarrow$$

$$p_{\epsilon,\lambda}(\delta_1, D) \geq p_{\epsilon,\lambda}(\delta_2, D)$$

- **Arbitrage-Freeness:** There is no **K-arbitrage** for any K.

+ **K-Arbitrage:** $\exists \delta_0, \delta_1, \delta_2, \cdots, \delta_k$, and a function $g : \mathcal{H}^k \to \mathcal{H}$:

$$\textstyle\sum_{i=1}^k p_{\epsilon,\lambda}(\delta_i, D) < p_{\epsilon,\lambda}(\delta_0, D)$$

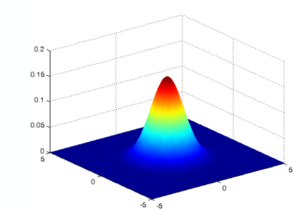
$$\mathbb{E}\left[\epsilon(\tilde{h}, D)\right] \leq \mathbb{E}\left[\epsilon(\hat{h}_\lambda^{\delta_0}(D), D)\right], \text{ where } \tilde{h} \text{ is the model}$$

$$\tilde{h} = g(\hat{h}_\lambda^{\delta_1}(D), \hat{h}_\lambda^{\delta_2}(D), \ldots, \hat{h}_\lambda^{\delta_k}(D)) \text{ s.t. } \mathbb{E}\left[\tilde{h}\right] = h_\lambda^*(D).$$

3.2 Gaussian Mechanism

$$\mathcal{K}_G(h_\lambda^*(D), w) = h_\lambda^*(D) + w, \quad w \sim \mathcal{N}(\mathbf{0}, (\delta/d)\cdot \mathbf{I}_d)$$

d-dimensional Gaussian Distribution



Mean: $\mathbf{0}_{d\times 1}$

Covariance: $\frac{\delta}{d}\mathbf{I}_{d\times d}$

Random sample

Optimal model instance vector

1.2	-3.1	0.5	0.1	-2.3	7.2	-0.9	5.5

+

Random noise vector

0.1	0.2	-0.1	0	0.1	0.3	0	-0.2

=

Model instance for buyer

1.3	-2.9	0.4	0.1	-2.2	7.5	-0.9	5.3

Insight: A concise characterization of well-behaved pricing functions

Thm: Assume that the error function is strictly convex. Then $\exists \phi(\cdot) s.t. \delta = \phi\left(\mathbb{E}\left[\epsilon(\hat{h}_\lambda^\delta(D), D)\right]\right)$. $p_{\epsilon,\lambda}$ is well-behaved iff $\hat{p}(x) \triangleq p_{\epsilon,\lambda}(1/\phi(x), D)$ is non-negative, montane, and subadditive.

3.3 Revenue Optimization

Problem Formulation
- What is Opt Variable: Pricing functions
- What is Goal: Max Expected Revenue
- What is Given: Market Info Estimation
  - Number of Buyers interested
  - Utility in their minds

T

Results
- Hardness: Co-NP-hard
- Approx Algorithm:
  - ½ approx. ratio
  - O(n^2) comp complex

4. Implementation and Experiments

Setup Intel i5-6600 3.3 GHz cores, 16 GB memory, Ubuntu 14.04 LTS

- A fixed market info
- Logistic regression
- Dataset YearMSD

MBP Gains	Lin	MaxC	MedC	OptC
Revenue	33.6x	37.0x	2.1x	1.4x
Affordability	55.9x	121x	1.9x	2.3x

Gains of Nimbus over Naïve Approaches

More in the paper
- More datasets, ML tasks
- More Market Scenarios
- Runtime Study
- GUIs
- Plenty of Open Questions
- …

Key takeaways: Cost of acquiring data is a key bottomneck for ML democratization. Nimbus proposes exchanging ML models directly instead of data with formal desiderata. A concrete Nimbus instance enables all desiderata, and thus optimizes accessibility of ML models for buyers and revenue for sellers.

Reference: L. Chen et al Model-based Pricing for ML in a Data Marketplace, SIGMOD 2019.
L. Chen et al Demo of Nimbus: Model-based Pricing for ML in a Data Marketplace, SIGMOD 2019.

Acknowledgement: We thank Jeffrey Naughton and Xi Wu for invaluable discussions. Google