Mixture Selection, Mechanism Design, and Signaling

Yu Cheng Ho Yee Cheung Shaddin Dughmi
Ehsan Emamjomeh-Zadeh Li Han Shang-Hua Teng

University of Southern California

Mixture Selection

- Optimization over distributions shows up everywhere in AGT.

Mixture Selection

- Optimization over distributions shows up everywhere in AGT.
  - Mixed strategies, lotteries, beliefs.

Mixture Selection

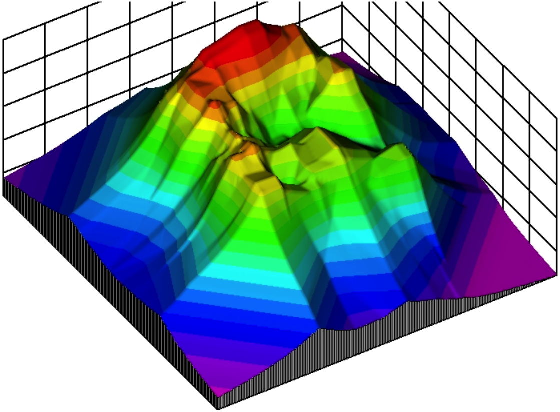
- Optimization over distributions shows up everywhere in AGT.
  - Mixed strategies, lotteries, beliefs.

Definition (Mixture Selection)

- *Parameter:* A function $g : [0,1]^n \to [0,1]$.
- *Input:* A matrix $A \in [0,1]^{n \times m}$.
- *Goal:* $\max_{x \in \Delta_m} g(Ax)$.

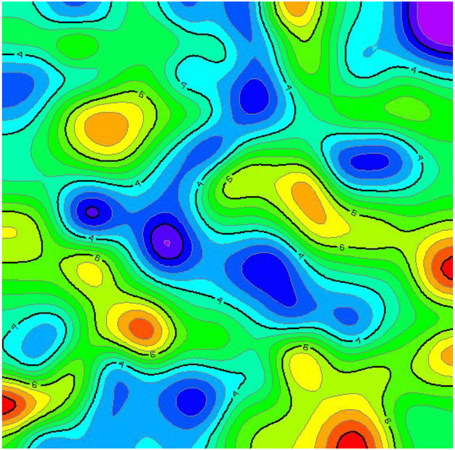
Mixture Selection

$$\max_{x \in \Delta_m} g(Ax)$$



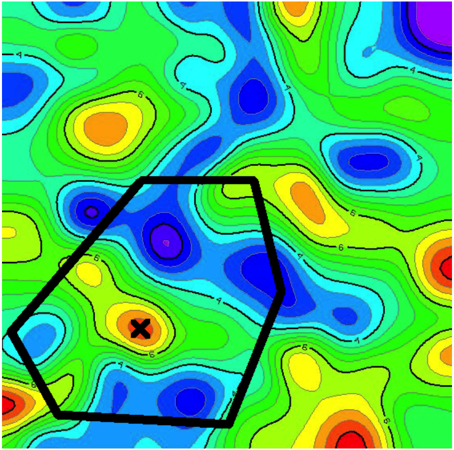
Mixture Selection

$$\max_{x \in \Delta_m} g(Ax)$$



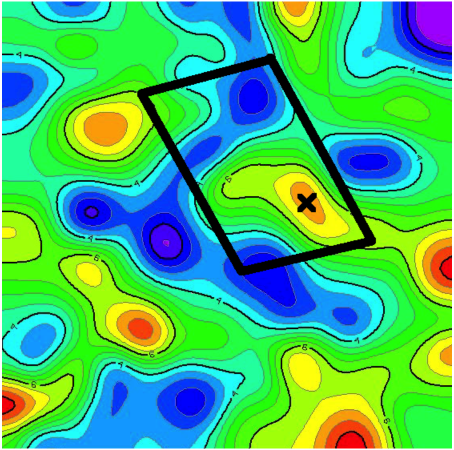
Mixture Selection

$$\max_{x \in \Delta_m} g(Ax)$$



Mixture Selection

$$\max_{x \in \Delta_m} g(Ax)$$



Mixture Selection: An Example

- Single buyer (with Bayesian prior) unit-demand pricing problem.
- Design a single lottery to maximize revenue.

			
	\$ 1	\$ 1/2	\$ 1/3
	\$ 1/3	\$ 1	\$ 1/2
	\$ 1/2	\$ 1/3	\$ 1

Mixture Selection: An Example

- Single buyer (with Bayesian prior) unit-demand pricing problem.
- Design a single lottery to maximize revenue.

			
	\$ 1	\$ 1/2	\$ 1/3
	\$ 1/3	\$ 1	\$ 1/2
	\$ 1/2	\$ 1/3	\$ 1

- A_{ij}: Type i's value for item j.
- x: Lottery to design.
- $g(Ax)$: Expected revenue of x with optimal price.

- Single buyer (with Bayesian prior) unit-demand pricing problem.
- Design a single lottery to maximize revenue.

			
	\$ 1	\$ 1/2	\$ 1/3
	\$ 1/3	\$ 1	\$ 1/2
	\$ 1/2	\$ 1/3	\$ 1

$x = (1, 0, 0) =$ 

$g(Ax) = 1/3$ with optimal price $p \in \{\$1, \$1/2, \$1/3\}$.

- Single buyer (with Bayesian prior) unit-demand pricing problem.
- Design a single lottery to maximize revenue.

			
	\$ 1	\$ 1/2	\$ 1/3
	\$ 1/3	\$ 1	\$ 1/2
	\$ 1/2	\$ 1/3	\$ 1

$$x = (1/3, 1/3, 1/3) =$$ 

$$g(Ax) = p = (\$1 + \$1/2 + \$1/3)/3 = 11/18.$$

Motivation

$$\max_{x \in \Delta_m} g(Ax)$$

- Building block in a number of game-theoretic applications.
- Mixture Selection problems naturally arise in mechanism design and signaling.

Motivation

$$\max_{x \in \Delta_m} g(Ax)$$

- Building block in a number of game-theoretic applications.
- Mixture Selection problems naturally arise in mechanism design and signaling.
- Information Revelation (signaling): design information sharing policies, so that the players arrive at "good" equilibria.
- The beliefs of the agents are distributions.

Our Results: Framework

Framework

Two "smoothness" parameters that tightly control the complexity of Mixture Selection.

A polynomial-time approximation scheme (PTAS) when both parameters are constants:

Our Results: Framework

Framework

Two "smoothness" parameters that tightly control the complexity of Mixture Selection.

A polynomial-time approximation scheme (PTAS) when both parameters are constants:

- $O(1)$-Lipschitz in L^∞ norm:
$$|g(v_1) - g(v_2)| \leq O(1) \cdot \|v_1 - v_2\|_\infty;$$

Our Results: Framework

Framework

Two "smoothness" parameters that tightly control the complexity of Mixture Selection.

A polynomial-time approximation scheme (PTAS) when both parameters are constants:

- $O(1)$-Lipschitz in L^∞ norm:
$$|g(v_1) - g(v_2)| \leq O(1) \cdot \|v_1 - v_2\|_\infty;$$
- $O(1)$-Noise stable:
Controls the degree to which low-probability (possibly correlated) errors in the inputs of g can impact its output.

Our Results: Noise Stability

Definition (β-Noise Stable)

A function g is β-Noise Stable if whenever

- a random process corrupts its input,

Our Results: Noise Stability

Definition (β-Noise Stable)

A function g is β-Noise Stable if whenever

- a random process corrupts its input,
- and the probability each entry gets corrupted is at most α,

The output of g decreases by no more than $\alpha\beta$ in expectation.

Our Results: Noise Stability

Definition (β-Noise Stable)

A function g is β-Noise Stable if whenever

- a random process corrupts its input,
- and the probability each entry gets corrupted is at most α,

The output of g decreases by no more than $\alpha\beta$ in expectation.

Must hold for all inputs, even when the corruptions are arbitrarily correlated.

Our Results: Applications

Game-theoretic problems in mechanism design and signaling.

Problem	Algorithm	Hardness
Unit-Demand Lottery Design [Dughmi, Han, Nisan '14]		
Signaling in Bayesian Auctions [Emek et al. '12] [Miltersen and Sheffet '12]		
Signaling to Persuade Voters [Alonso and Câmara '14]		
Signaling in Normal Form Games [Dughmi '14]		

Our Results: Applications

Game-theoretic problems in mechanism design and signaling.

Problem	Algorithm	Hardness
Unit-Demand Lottery Design [Dughmi, Han, Nisan '14]	PTAS	No FPTAS
Signaling in Bayesian Auctions [Emek et al. '12] [Miltersen and Sheffet '12]	PTAS	No FPTAS
Signaling to Persuade Voters [Alonso and Câmara '14]	PTAS[1]	No FPTAS
Signaling in Normal Form Games [Dughmi '14]	Quasi-PTAS[2]	No FPTAS[3]

[1] Bi-criteria.

[2] $n^{O(\log n)}$ for all fixed ϵ. Bi-criteria.

[3] Assume hardness of planted clique. Recently [Bhaskar, Cheng, Ko, Swamy '16] rules out PTAS.

Simple Algorithm for Mixture Selection

Inspired by ϵ-Nash algorithm in [Lipton, Markakis, Mehta '03].

Support enumeration

- Enumerate all s-uniform mixtures $\tilde{x}$ for $s = O(\log(n)/\epsilon^2)$.
- Check the values of $g(A\tilde{x})$ and return the best one.

Simple Algorithm for Mixture Selection

Inspired by ϵ-Nash algorithm in [Lipton, Markakis, Mehta '03].

Support enumeration

- Enumerate all s-uniform mixtures $\tilde{x}$ for $s = O(\log(n)/\epsilon^2)$.
- Check the values of $g(A\tilde{x})$ and return the best one.

Proof

- Take the optimal solution x^*.
- Draw s samples from x^* and let $\tilde{x}$ be the empirical distribution.

Simple Algorithm for Mixture Selection

Inspired by ϵ-Nash algorithm in [Lipton, Markakis, Mehta '03].

Support enumeration

- Enumerate all s-uniform mixtures $\tilde{x}$ for $s = O(\log(n)/\epsilon^2)$.
- Check the values of $g(A\tilde{x})$ and return the best one.

Proof

- Take the optimal solution x^*.
- Draw s samples from x^* and let $\tilde{x}$ be the empirical distribution.
- Tail bound + union bound: $\Pr\left[\|Ax^* - A\tilde{x}\|_\infty < \epsilon\right] > 0$.
- Probabilistic method: there exists a s-uniform $\tilde{x}$ s.t. $\|Ax^* - A\tilde{x}\|_\infty < \epsilon$.
- If g is $O(1)$-Lipschitz in L_∞, $g(A\tilde{x}) \geq g(Ax^*) - O(\epsilon)$.

Simple Algorithm for Mixture Selection

- Running Time: Evaluate $g(\cdot)$ on m^s inputs.
- A Quasi-PTAS for Mixture Selection when g is $O(1)$-Lipschitz in L_∞.

Simple Algorithm for Mixture Selection

- Running Time: Evaluate $g(\cdot)$ on m^s inputs.
- A Quasi-PTAS for Mixture Selection when g is $O(1)$-Lipschitz in L_∞.

Bypass the Union Bound

Sample $s = O(\log n/\epsilon^2)$ times.

Simple Algorithm for Mixture Selection

- Running Time: Evaluate $g(\cdot)$ on m^s inputs.
- A Quasi-PTAS for Mixture Selection when g is $O(1)$-Lipschitz in L_∞.

Bypass the Union Bound

Sample $s = O(\log n/\epsilon^2)$ times.

Each entry $(Ax)_i$ gets changed by at most ϵ, with probability $(1-\epsilon)$.

Works if g is Noise Stable.

Simple Algorithm for Mixture Selection

- Running Time: Evaluate $g(\cdot)$ on m^s inputs.
- A Quasi-PTAS for Mixture Selection when g is $O(1)$-Lipschitz in L_∞.

Bypass the Union Bound

Sample $s = O(\log n/\epsilon^2)$ times.

Each entry $(Ax)_i$ gets changed by at most ϵ, with probability $(1 - \epsilon)$.

Works if g is Noise Stable.

Summary

- High probability "small errors" (Lipschitz Continuity).
- Low probability "large errors" (Noise Stability).

Our results: Main Theorem

Theorem (Approximate Mixture Selection)

If g is β-Stable and c-Lipschitz, there is an algorithm with

Runtime: $m^{O(c^2 \log(\beta/\epsilon)/\epsilon^2)} \cdot T_g$,

Approximation: $OPT - \epsilon$.

When $\beta, c = O(1)$, this gives a PTAS.

Our results: Main Theorem

Theorem (Approximate Mixture Selection)

If g is β-Stable and c-Lipschitz, there is an algorithm with

Runtime: $m^{O(c^2 \log(\beta/\epsilon)/\epsilon^2)} \cdot T_g$,

Approximation: $OPT - \epsilon$.

When $\beta, c = O(1)$, this gives a PTAS.

Problem	c (Lipschitzness)	β (Stablility)	Runtime
Unit-Demand Lottery Design	1	1	PTAS
Signaling in Bayesian Auctions	1	2	PTAS
Signaling to Persuade Voters	$O(1)$	$O(1)$	PTAS
Signaling in Normal Form Games	2	$\text{poly}(n)$	Quasi-PTAS

Our results: Main Theorem

Theorem (Approximate Mixture Selection)

If g is β-Stable and c-Lipschitz, there is an algorithm with

Runtime: $m^{O(c^2 \log(\beta/\epsilon)/\epsilon^2)} \cdot T_g$,

Approximation: $OPT - \epsilon$.

When $\beta, c = O(1)$, this gives a PTAS.

Problem	c (Lipschitzness)	β (Stablility)	Runtime
Unit-Demand Lottery Design	1	1	PTAS
Signaling in Bayesian Auctions	1	2	PTAS
Signaling to Persuade Voters	$O(1)$	$O(1)$	PTAS
Signaling in Normal Form Games	2	$\text{poly}(n)$	Quasi-PTAS

Our results: Lottery Design

			
	\$ 1	\$ 1/2	\$ 1/3
	\$ 1/3	\$ 1	\$ 1/2
	\$ 1/2	\$ 1/3	\$ 1

Let $v = Ax$. v_i is type i's expected value for lottery x.

$$g^{(\text{lottery})}(v) := \max_p \left\{ p \cdot \frac{|\{i : v_i \geq p\}|}{n} \right\}.$$

Our results: Lottery Design

			
	\$ 1	\$ 1/2	\$ 1/3
	\$ 1/3	\$ 1	\$ 1/2
	\$ 1/2	\$ 1/3	\$ 1

Let $v = Ax$. v_i is type i's expected value for lottery x.

$$\mathrm{g}^{(\text{lottery})}(v) := \max_p \left\{ p \cdot \frac{\left|\{i : v_i \geq p\}\right|}{n} \right\}.$$

$\mathrm{g}^{(\text{lottery})}$ is 1-Lipschitz

Lower the price by ϵ.

Our results: Lottery Design

			
	\$ 1	\$ 1/2	\$ 1/3
	\$ 1/3	\$ 1	\$ 1/2
	\$ 1/2	\$ 1/3	\$ 1

Let $v = Ax$. v_i is type i's expected value for lottery x.

$$g^{(\text{lottery})}(v) := \max_p \left\{ p \cdot \frac{|\{i : v_i \geq p\}|}{n} \right\}.$$

$g^{(\text{lottery})}$ is 1-Stable

Buyer walks away with probability at most ϵ.

Hardness Results

Neither Lipschitz Continuity nor Noise Stability suffices by itself for a PTAS.

Absence of L_∞-Lipschitz Continuity

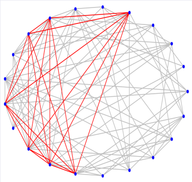
NP-Hard (even when g is $O(1)$-Lipschitz in L_1).
Reduction from Maximum Independent Set.

Hardness Results

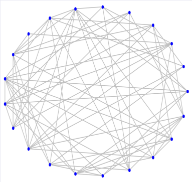
Neither Lipschitz Continuity nor Noise Stability suffices by itself for a PTAS.

Absence of Noise Stability

As hard as Planted Clique.



$$\max_x g(Ax) = 1$$



$$\max_x g(Ax) < 0.8$$

Hardness Results

FPTAS with Lipschitz Continuity and Noise Stability

NP-Hard.

Both assumptions together do not suffice for an FPTAS.

Conclusion

Our Contributions

- Define Mixture Selection.
- Simple meta algorithm.
- PTAS when g is $O(1)$-Stable and $O(1)$-Lipschitz.
- Applications to a number of game-theoretic problems.
- Matching lower bounds.

Conclusion

Our Contributions

- Define Mixture Selection.
- Simple meta algorithm.
- PTAS when g is $O(1)$-Stable and $O(1)$-Lipschitz.
- Applications to a number of game-theoretic problems.
- Matching lower bounds.

- Find more applications.
- [Barman'15]: PTAS when A is sparse, and g is Lipschitz but not Stable.