Maximum Entropy Distributions: Optimization and Applications to Approximation Algorithms

Mohit Singh

Georgia Tech

Overview

1. What are maximum entropy distributions?
   - Relationship to partition functions.

2. What can we say about computation of maximum entropy distributions?
   - Ellipsoid method and equivalences in optimization and separation.
   - Implications to max-entropy distributions.

3. How are they useful for combinatorial optimization problems?

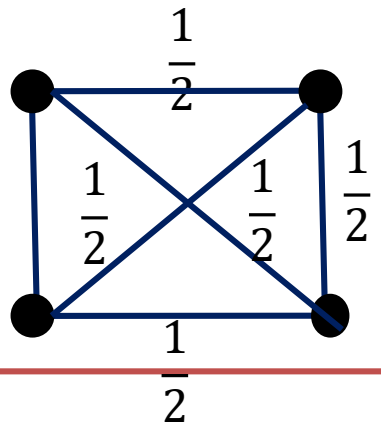
4. Scaling as convex optimization.
   - Can be used more general counting and optimization problems.
   - Gurvits' proof of Van-der-Warden Conjecture and its extensions.

Inferring distributions from limited information

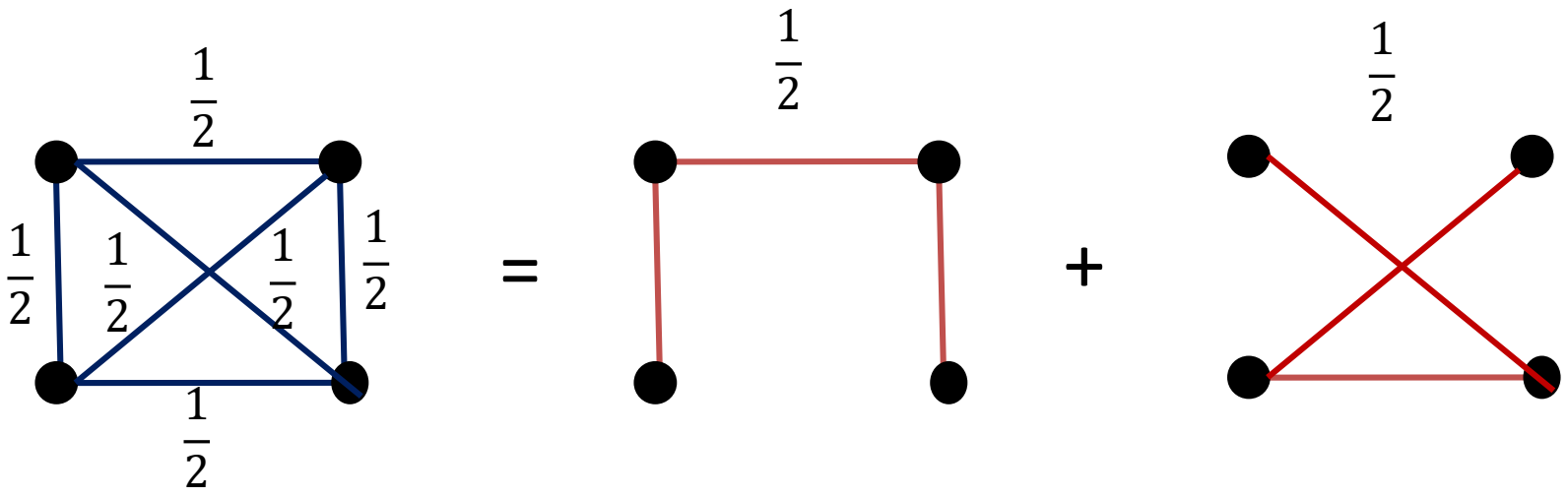
- Suppose there is an unknown distribution p over a collection $\mathcal{M}$ of subsets of {1,…,m}.
- We observe $\theta_e = \Pr_{S \leftarrow p}[e \in S]$ for each $e \in [m]$.
- Equivalently, we are given $\theta \in P(\mathcal{M}) \coloneqq conv(1_S : S \in \mathcal{M})$

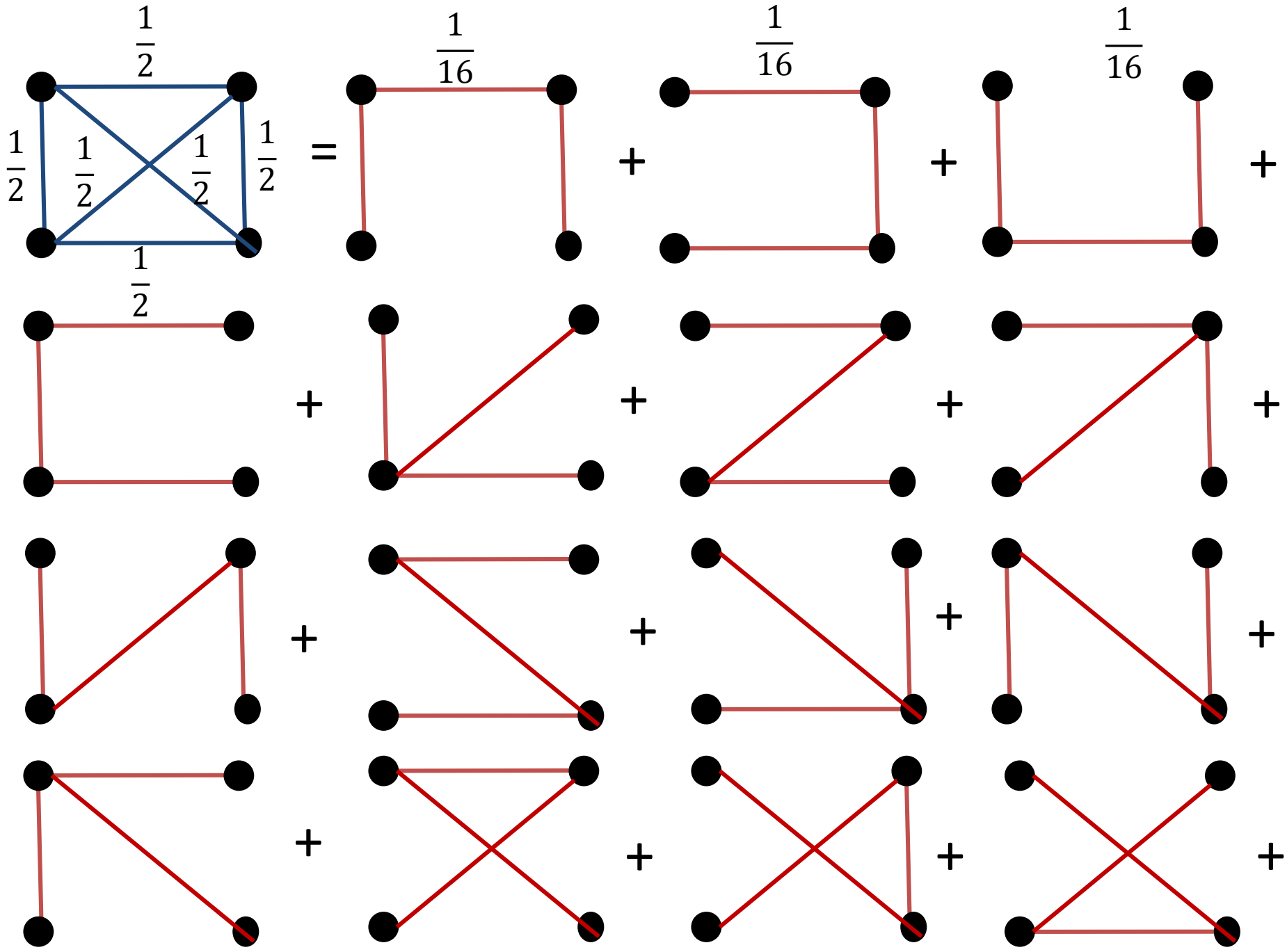
Given θ, what is the best guess for p?

Example: Given $G = (V, E)$, let $m = |E|$, $\mathcal{M} = \{S \subseteq E : S \text{ is a spanning tree of } G\}$



Spanning Trees




1/2
1/2
1/2
1/2
1/2
=
1/16
+
1/16
+
1/16
+
1/2
+
+
+
+
+
+
+
+
+
+
+
+
+
+
+
+

Principle of Maximum Entropy

- Pick the distribution which maximizes entropy.

  Given $\theta \in P(\mathcal{M})$, solve this convex program.

$$\begin{aligned} \max \quad & H(p) = \sum_{S \in \mathcal{M}} p_S \ln \frac{1}{p_s} \\ \text{s.t.} \quad & \sum_{S: e \in S} p_S = \theta_e \quad \forall e \in [m] \\ & \sum_{S \in \mathcal{M}} p_S = 1 \\ & p \geq 0 \end{aligned}$$

The guessed p can be used to obtain samples and infer other statistics about the distribution.

Occur in machine learning, economics, information theory, statistical physics, combinatorics and in approximation algorithms.

Where do marginals come from

Consider your favorite combinatorial optimization problem:

Traveling Salesman problem: Given n cities with distances between any pair, find a Hamiltonian cycle of minimum total distance.

$$\begin{aligned} &\min \quad c\,x \\ &x \in P \\ &x \in \{0,1\}^m \end{aligned}$$

$$\begin{aligned} &\min \quad c\,x \\ &x \in P \\ &x \in [0,1]^m \end{aligned}$$

Marginals: Use the linear relaxation.
Family $\mathcal{M}$: Judicious choice based on the problem.

Principle of Maximum Entropy

- Pick the distribution which maximizes entropy.

  Given $\theta \in P(\mathcal{M})$, solve this convex program.

$$\begin{aligned} \max \quad & H(p) = \sum_{S \in \mathcal{M}} p_S \ln \frac{1}{p_S} \\ \text{s.t.} \quad & \sum_{S: e \in S} p_S = \theta_e \quad \forall e \in [m] \\ & \sum_{S \in \mathcal{M}} p_S = 1 \\ & p \geq 0 \end{aligned}$$

- Entropy is a concave function.
  - Duality.
  - Efficient Convex optimization.

Representation of max-entropy distributions

- Given any $\gamma_e > 0$, let p^γ denote the probability distribution where $p^\gamma(S) \propto \prod_{e \in S} \gamma_e$ for each $S \in \mathcal{M}$.
- Convex duality implies that there exists γ^* such that p^{γ^*} is optimal if $\theta \in int(P(\mathcal{M}))$.
- Thus the optimal distribution can be represented by m numbers, γ_e^* for each $e \in [m]$.

$p^\gamma(S) = \frac{\prod_{e \in S} \gamma_e}{Z(\gamma)}$ where $Z(\gamma) = \sum_{S \in \mathcal{M}} \prod_{e \in S} \gamma_e$ is the partition function.

Computational Questions

Compute p^{γ^*}. Typically, $\mathcal{M}$ is implicit and is exponential size in the input of the problem.
Examples: Spanning trees of a graph, Perfect matchings in a (bipartite) graph.

- Compute γ such that p^{γ} has marginals close to θ and entropy close to $H\left(p^{\gamma^*}\right)$?
- Does there exists such γ that can be succinctly described?

Counting implies Optimization

Lemma [S. , Vishnoi'13]: Succint representation of approximate max-entropy distributions always exist (under mild assumptions).

Theorem [S., Vishnoi'13]: There exists an algorithm that given a counting oracle computes approximate max-entropy distributions in polynomial time.

Counting oracle: Given γ_e for each $e \in \{1, \dots, m\}$ computes $\sum_{S \in \mathcal{M}} \prod_{e \in S} \gamma_e$ in time polynomial in bits needed to represent γ.

Counting oracle can be approximate and/or randomized.

Spanning trees: Kirchoff's theorem.

Perfect matchings in bipartite graphs: Jerrum, Sinclair, Vigoda'01.

Computation of Maximum Entropy Distributions

Primal Program

$$\max \quad H(p) = \sum_{S \in \mathcal{M}} p_S \ln \frac{1}{p_S}$$
$$\text{s.t.} \quad \sum_{S: e \in S} p_S = \theta_e \quad \forall e \in [m]$$
$$\sum_{S \in \mathcal{M}} p_S = 1$$
$$p \geq 0$$

Dual Program

$$\min g(\lambda) = \min \ln \left(\sum_{S \in \mathcal{M}} e^{\lambda \cdot 1_S} \right) - \lambda \cdot \theta$$
$$\lambda \in R^m$$

Lemma: Strong duality implies that there exists (p^*, λ^*) s.t. $p^*(S) \propto e^{\lambda^* \cdot 1_S}$ for each $S \in \mathcal{M}$.

Defining $\gamma_e^* = e^{\lambda_e^*}$, we obtain $p^{\gamma^*} = p^*$.

We will solve the dual using the ellipsoid algorithm.

Discovering Lions in Sahara: Ellipsoid Algorithm

Lion: Given a target τ, find λ s.t. $g(\lambda) \leq \tau$.

1. Divide the Sahara in half with a fence.
2. if you find the lion, we are done;
3. otherwise, determine one half of the Sahara is empty (does not contain the lion),
4. repeat with the nonempty half,
5. continue until the fenced area is smaller than a lion.

Ellipsoid Algorithm with Ellipsoids

Given a target τ, find λ s.t. $g(\lambda) \leq \tau$.

1. Set $E_0 = B(0, R)$ ball of radius R s.t. $\lambda^* \in E_0$.
2. At any iteration i, maintain

$$\mathrm{E}_i = \mathrm{E}(\mathrm{z_i}, \mathrm{A}) \coloneqq \{z: (z - z_i)^T A^{-1} (z - z_i) \leq 1\}$$

3. If $g(z_i) \leq \tau$, then return $\lambda = z_i$ and update τ.
4. Else, $\{\lambda: g(\lambda) \leq \tau\} \subseteq \{z: g(z_\mathrm{i}) + \nabla g(z_i)^T (z - z_i) \leq \tau\}$.

   (Separating hyperplane).
5. Set $E_{i+1} \supseteq E_i \cap \{z: g(z_\mathrm{i}) + \nabla g(z_i)^T (z - z_i) \leq \tau\}$.
6. End if volume$(E_{i+1}) < C \cdot \delta^m$.

Lemma 1: We can find E_{i+1} s.t. $vol(E_{i+1}) \leq e^{-\frac{1}{2m}} vol(E_i)$.

Iterations

- If g is $L - lipschitz$, then

$$B\left(\lambda^*, \frac{\epsilon}{L}\right) \subseteq \{z: g(z) \leq g(\lambda^*) + \epsilon\}$$

Lemma: If $\tau > g(\lambda^*) + \epsilon$, then the algorithm finds a feasible point.

Pf: If not, $B\left(\lambda^*, \frac{\epsilon}{L}\right)$ remains inside the ellipsoid. Contradiction.

Number of iterations: $O\left(m^2 \log \frac{RL}{\epsilon}\right)$ to find z s.t. $g(z) \leq g(\lambda^*) + \epsilon$.

Ellipsoid Algorithm with Ellipsoids

Given a target τ, find λ s.t. $g(\lambda) \leq \tau$.

1. Set $E_0 = B(0, R)$ ball of radius R s.t. $\lambda^* \in E_0$.
2. At any iteration i, maintain

$$\mathrm{E}_i = \mathrm{E}(\mathrm{z_i}, \mathrm{A}) \coloneqq \{z: (z - z_i)^T A^{-1} (z - z_i) \leq 1\}$$

3. If $g(z_i) \leq \tau$, then return $\lambda = z_i$ and update τ.
4. Else, $\{\lambda: g(\lambda) \leq \tau\} \subseteq \{z: g(z_\mathrm{i}) + \nabla g(z_i)^T (z - z_i) \leq \tau\}$.
   (Separating hyperplane).
5. Set $E_{i+1} \supseteq E_i \cap \{z: g(z_\mathrm{i}) + \nabla g(z_i)^T (z - z_i) \leq \tau\}$.
6. End if volume$(E_{i+1}) < C \cdot \delta^m$.

Separation Oracle and Value oracle can be implemented using the counting oracle (assuming self-reducibility).

$$\nabla g(\lambda)_e = \frac{\sum_{S \in \mathcal{M}: e \in S} \prod_{f \in S} \exp(\lambda \cdot 1_S)}{\sum_{S \in \mathcal{M}} \prod_{f \in S} \exp(\lambda \cdot 1_S)} - \theta_e$$

Ellipsoid Algorithm with Ellipsoids

Given a target τ, find λ s.t. $g(\lambda) \leq \tau$.

1. Set $E_0 = B(0, R)$ ball of radius R s.t. $\lambda^* \in E_0$.
2. At any iteration i, maintain

$$\mathrm{E}_i = \mathrm{E}(\mathrm{z_i}, \mathrm{A}) \coloneqq \{z: (z - z_i)^T A^{-1} (z - z_i) \leq 1\}$$

3. If $g(z_i) \leq \tau$, then return $\lambda = z_i$ and update τ.
4. Else, $\{\lambda: g(\lambda) \leq \tau\} \subseteq \{z: g(z_\mathrm{i}) + \nabla g(z_i)^T (z - z_i) \leq \tau\}$.
   (Separating hyperplane).
5. Set $E_{i+1} \supseteq E_i \cap \{z: g(z_\mathrm{i}) + \nabla g(z_i)^T (z - z_i) \leq \tau\}$.
6. End if volume$(E_{i+1}) < C \cdot \delta^n$.

How big is the Sahara or equivalently λ^*?

Bound on λ^*

- Lemma: We have $|\lambda^*|_2 \leq \frac{m}{\eta}$ if θ in η-interior of $P(\mathcal{M})$.

Def: Polar of P, $P^* = \{x : x^T y \leq 1 \; \forall y \in P\}$

Fact 1: P, Q convex $P \subseteq Q$ implies that $P^* \supseteq Q^*$.

Fact 2: $B(0,r)^* = B\left(0, \frac{1}{r}\right)$.

Bound on λ^*

- Lemma: We have $|\lambda^*|_2 \leq \frac{m}{\eta}$ if θ in η-interior of $P(\mathcal{M})$.
- Proof: $g(\lambda^*) = ln\left(\sum_{S\in\mathcal{M}} e^{\lambda^*\cdot 1_S}\right) - \lambda^*\cdot\theta \leq m$

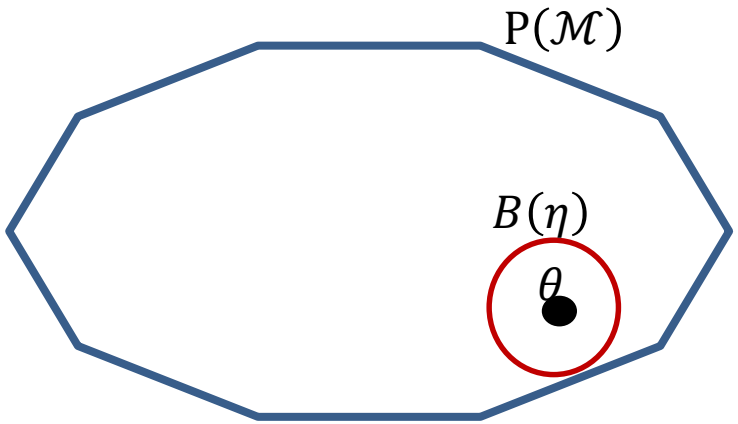
by duality and bound on entropy .

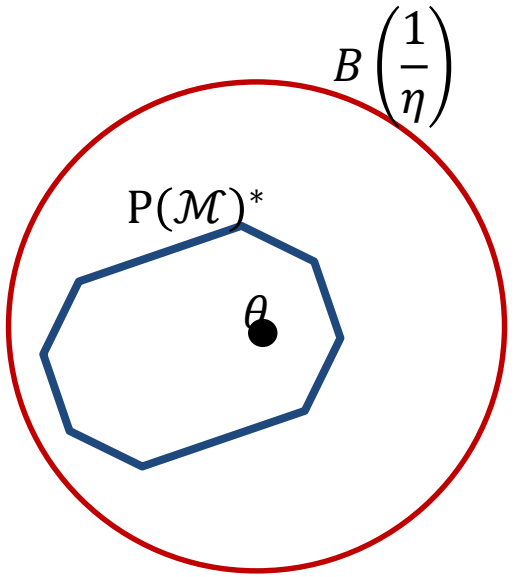
Thus $\lambda^*\cdot 1_S - \lambda^*\cdot\theta \leq m \quad for\ all\ S\in\mathcal{M}$

$\lambda^*\cdot x - \lambda^*\cdot\theta \leq m \quad x\in P(\mathcal{M})$

We get λ^*/m is in the polar of $P(\mathcal{M})$ with origin shifted to θ.

But then $\frac{\lambda^*}{\mathrm{m}} \in B(\eta)^* = B\left(\frac{1}{\eta}\right)$





Counting implies Optimization

- Theorem (counting to optimization): There is an algorithm that given a counting oracle for $\mathcal{M}$, and a point θ in the η-interior of $\mathcal{M}$, an $\epsilon > 0$ returns a $\tilde{\lambda} \in R^m$ such that

$$\left|\tilde{\lambda} - \lambda^*\right|_2 \leq \epsilon$$

The number of oracle calls is polynomial in m, $\ln 1/\epsilon$, $\ln 1/\eta$.

Lemma: Let $\tilde{p}$ be the distribution such that $\tilde{p}(S) \propto e^{\tilde{\lambda} \cdot 1_S}$. We have

1. $g(\tilde{\lambda}) - g(\lambda^*) \leq \sqrt{m}\left|\tilde{\lambda} - \lambda^*\right|_2$
2. $D_{KL}(\tilde{p}\,||p^*) = g(\tilde{\lambda}) - g(\lambda^*)$.

Corollary: We have that

1. $H(\tilde{p}) \geq H(p^*) - \epsilon_1$
2. For each e $\in [m]$, $\sum_{e \in S:\, S \in \mathcal{M}} \tilde{p}(S) \cong \theta_e$

Interiority

- What if θ is close to boundary?
  - Pull it inside. Find any point $\hat{\theta}$ deep in interior and work with $\epsilon\hat{\theta} + (1-\epsilon)\theta$.
  - Project on lower dimensional face. Obtain distribution over vertices of the face.

- Technical difficulties [Straszak, Vishnoi'17]

Optimization implies Counting

- Since counting algorithms are few and far between, are they necessary to solve the max-entropy convex program?

Theorem[S., Vishnoi'13]: There is an algorithm that can approximately estimate $|\mathcal{M}|$ given an oracle for solving the max-entropy convex program.

Results together imply an equivalence between counting problems and max-entropy convex programs.

Special case of equivalence between the separation problem and optimization problem [Grostchel, Lovasz, Schrijver'81].

Optimization to Counting

Given any $\theta \in P(\mathcal{M})$, the optimization oracle returns λ^* optimizing

$$\min_\lambda g_\theta(\lambda) = \min_\lambda \ln \sum_{S \in \mathcal{M}} e^{\lambda \cdot 1_S} - \lambda \cdot \theta$$

We are going to solve

$$\max_{\theta \in P(\mathcal{M})} \min_\lambda g_\theta(\lambda)$$

Lemma: We have

$$\max_{\theta \in P(\mathcal{M})} \min_\lambda g_\theta(\lambda) = \ln |\mathcal{M}|$$

Another Convex Program

Lemma: $\max_{\theta \in P} \min_{\lambda} \; g_\theta(\lambda) = \ln |\mathcal{M}|$.

Proof: For any θ, $\min_{\lambda} \; g_\theta(\lambda) = H(p^*)$

But $H(p^*) \leq \ln |\mathcal{M}|$ since the support of p^* is $\mathcal{M}$.

Thus $\max_{\theta \in P} \min_{\lambda} \; g_\theta(\lambda) \leq \ln |\mathcal{M}|$.

Set $\theta^* = \sum_{S \in \mathcal{M}} \frac{1_S}{|\mathcal{M}|} \in P(\mathcal{M})$. Then $\min_{\lambda} \; g_{\theta^*}(\lambda) = \ln |\mathcal{M}|$.

Odds and Ends

We want to solve

$$\max_{\theta \in P(\mathcal{M})} \; min_{\lambda} \; g_{\theta}(\lambda)$$

But we can solve

$$\max_{\theta \in int\left(P(\mathcal{M}), \frac{\epsilon}{m}\right)} \; min_{\lambda} \; g_{\theta}(\lambda)$$

Lemma 1: A separation oracle for P gives an approximate separation oracle for $int\left(P, \frac{\epsilon}{m}\right)$.

Lemma 2 : $f(\theta) \coloneqq \min_{\lambda} g_{\theta}(\lambda)$ is Lipschitz.

Now, we use the machinery of ellipsoid algorithm.

Optimization to counting

- Theorem 3(restated): There is an algorithm that given $P(\mathcal{M})$ by a separation oracle, an optimization oracle for maximum entropy convex program for $\mathcal{M}$ returns Z such that

$$(1-\epsilon)\ln|\mathcal{M}| \leq Z \leq (1+\epsilon)\ln|\mathcal{M}|$$

The number of oracle calls is polynomial in m, $\ln 1/\epsilon$.

Remarks:

1. The optimization oracle for maximum entropy convex program will be called only at points that are in $\frac{\epsilon}{m} - interior$.

Overview

1. What are maximum entropy distributions?
   - Relationship to partition functions.

2. What can we say about computation of maximum entropy distributions?
   - Ellipsoid method and equivalences in optimization and separation.
   - Implications to max-entropy distributions.

3. How are they useful for combinatorial optimization problems?

4. Scaling as convex optimization.
   - Its generalizations and applications.
   - Gurvits' proof of Van-der-Warden Conjecture and its extensions.