Unique Games

Saturday, April 25, 2015
12:09 PM

Recall the Label Cover problem from last class:

- Bipartite graph $G = (U, V, E)$
- Alphabet Σ
- Relations $\{\pi_e \subseteq \Sigma \times \Sigma : e \in E\}$ with projection property

(Aside: defining π_e as map from Σ to Σ also works)

Unique Game Problem: Label Cover where each π_e is a bijection (permutation).

Rem: Clearly, $[1, s]$-Gap-UG is easy. Specifying label for any one vertex fixes labels for all others in same component.

Unique Game Conjecture: Given $\varepsilon > 0$, $\exists\ \Sigma$ s.t. $[1-\varepsilon, \varepsilon]$-Gap-UG is NP-hard on instances with alphabet Σ and left-regular graph.

Why important?

- Allows proving lower bounds for 2CSP's (like Max-Cut and Max-2SAT) that we don't know how to approach using standard Label Cover
- Seems to be intimately tied to power of SDP's
- For the canonical SDP's for CSP's we saw earlier, tight relationships between UG-hardness and integrality gaps
- Both proving UG-hardness as well as approx algorithms for UG have yielded lots of new math insights
- Upper bound status: $[1-\varepsilon, 1-\varepsilon\sqrt{\log n}]$-Gap-UG is in P

by solving the following SDP:

$$\max \frac{1}{|E|} \sum_{e=(u,v)\in E} \sum_{\sigma\in\Sigma} \langle x_{u,\sigma}, x_{v,\pi_e(\sigma)} \rangle$$

$$\text{s.t. } \forall v, \sum_{\sigma\in\Sigma} \|x_{v,\sigma}\|^2 = 1$$

$$\forall v, \sigma\neq\sigma' \in \Sigma, \langle x_{v,\sigma}, x_{v,\sigma'} \rangle = 0$$

$$\forall u, v \in V, \sigma, \sigma' \in \Sigma, \langle x_{u,\sigma}, x_{v,\sigma'} \rangle \geq 0.$$

- No natural distribution known such that the problem is hard for this algorithm.

Reduction from Unique Games

- Divide lecture into two parts
  - Low-tech reductions (fairly direct from def of problem)
  - High-tech reductions (using the high-tech gadget of dictatorship tests)

Actually, also high-tech because we use an alt. version of UGC established using high tech means

- We'll see hardness for:
  - Multicut
  - Min-2CNF-Deletion
  - Max-Cut

  Decreasing amount of details

Multicut

- Will start from an equiv. formulation of UG.
- Given system of linear equations in n variables, with each equation of the form $x_i - x_j = a_{ij} \pmod{k}$, the Max-2Lin-$k$ problem is finding an assignment of $x_1, \ldots, x_n$ to $\{0, 1, \ldots, k-1\}$ that maximizes # of satisfied eqn's.
- Linear Unique Games Conjecture: $\forall \varepsilon, \delta > 0, \exists k$ s.t. $[1-\delta, \varepsilon]$-Gap-Max-2Lin-k is NP-hard.
- Equivalent to UGC [... O'Donnell]

[Khot - Kindler - Mossel - O'Donnell]

- Recall Multicut problem: given graph $G = (V, E)$ and pairs $\{(s_1, t_1), (s_2, t_2), \dots, (s_k, t_k)\}$, find min # of edges to remove to disconnect all k pairs.
  - This is the unweighted version. Hardness for this problem implies hardness for more general weighted version.
  - We've seen $O(\lg k)$ approx in class earlier.
- Theorem: For any const. $\alpha > 0$, α-approx of Multicut does not exist, assuming UGC.
- Reduction from $[1-\varepsilon, \delta]$-Gap-Max-2Lin-k to $\left[\frac{1-\delta}{2}, \varepsilon\right]$-Gap-Multicut
- Reduction: Given Max-2Lin-k instance on n variables with m equations, construct graph G on nk vertices with mk edges. For each variable v, have k vertices $\{(v, i) : 0 \le i < k\}$, and for any equation $u - v = c_{uv} \pmod k$, add edges between (u, i) and (v, j) if $i - j = c_{uv} \pmod k$.
  Let the set of source-destination pairs be:
  $\{((u, i), (u, j)) : i \neq j,\ u \text{ var in Max-2Lin-}k \text{ instance}\}$

- Completeness: Reduction carries Max-2Lin-k instances of value $\ge 1-\varepsilon$ to Multicut instances of value $\le \varepsilon$.
- Proof: Let I be a Max-2Lin-k instance on variable set V of size n, and suppose Max-2Lin-$k(I) \ge 1 - \varepsilon$. Let $L : V \to \{0, \dots, k-1\}$ be an assignment of max value.
  For $c \in \{0, \dots, k-1\}$, let $V'_c = \{(u, i) : u \in V,\ i = L(u) + c\}$.
  Note that each V'_c contains one representative of u. Think of each V'_c as a "cluster".
  Claim: # of inter-cluster edges is $\le \varepsilon m k$ (m = # eqn's)
  Pf: Observe that if there's an intercluster edge $((u,i), (v,j))$, then $L(u) - L(v) \neq c_{uv} \pmod k$. Why? We know $i - j = c_{uv} \pmod k$
  $(u, i) \in V'_c$, $(v, j) \in V'_{c'}$ with $c \neq c'$. Then, $L(u) - L(v)$

$= (i-c)-(j-c') \pmod k = C_{uv} - (c-c') \pmod k \neq C_{u,v} \pmod k$.

- **Soundness**: Reduction carries Max-2Lin-k instances of value $\leq \delta$ to Multicut instances of value $\geq \frac{1-\delta}{2}$.
- **Proof**: Suppose reduction outputs Multicut instance of value $< \frac{1-\delta}{2}$.

Let $V_1', \ldots, V_\ell'$ be the cc's left after removing $< \frac{1-\delta}{2} mk$ edges, ordered in random order. Note that each V_i' contains only one rep of any variable. Again, think of each V_i' as a "cluster".

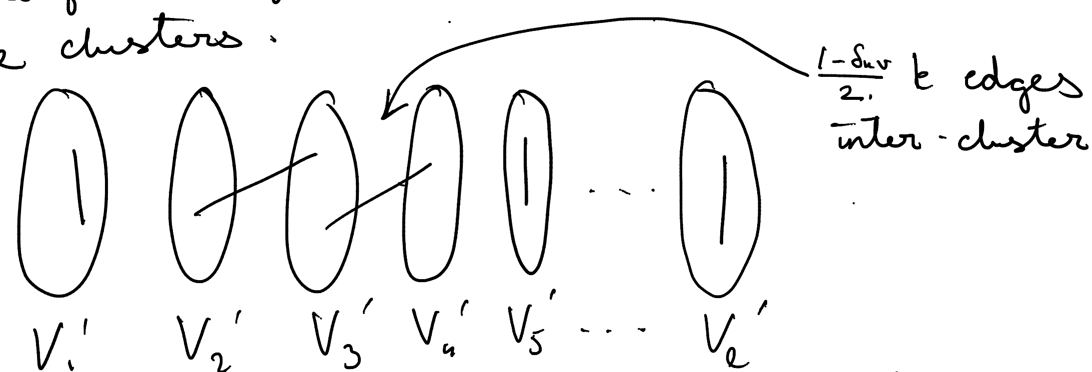
For any variable u, let $c(u) = \arg\min_{c \in [\ell]} \{ V_c' \text{ contains rep of } u \}$

and let $L(u) = i$ where $(u,i) \in V_{c(u)}$.

Consider an equation $u - v = C_{uv} \pmod k$. We bound the prob. that $L(u) - L(v) \neq C_{uv} \pmod k$.

Note that if $((u, L(u)), (v, L(v)))$ is an edge, then $L(u) - L(v) = C_{u,v}$.

Let $\frac{1-\delta_{uv}}{2}$ be the fraction of edges of the form $((u,\cdot),(v,\cdot))$ that straddle clusters.



Call a cluster good if it contains reps of both u and v, connected by an edge. So, V_1', V_5' and V_ℓ' are good in above figure.

$\#$ good clusters $= \frac{1+\delta_{uv}}{2} k$, $\#$ of bad clusters $\leq (1-\delta_{u,v}) k$

$$\Pr[\text{first cluster containing rep of } u \text{ or } v \text{ is bad}] \leq \frac{\#\text{bad}}{\#\text{bad} + \#\text{good}} = 1 - \frac{\#\text{good}}{\#\text{bad} + \#\text{good}} \leq 1 - \frac{\frac{1+\delta_{uv}}{2}}{1-\delta_{uv} + \frac{1+\delta_{uv}}{2}}$$

$$= 1 - \frac{1+\delta_{uv}}{3-\delta_{uv}}$$

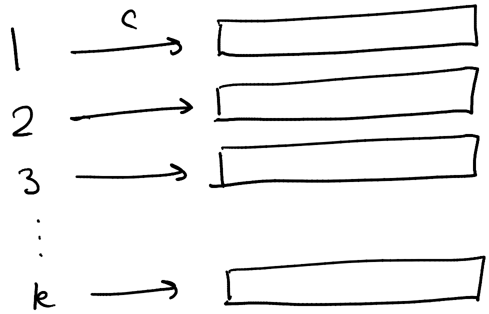
$$= \frac{2}{3-\delta_{uv}} \cdot (1-\delta_{uv}) \leq 1-\delta_{uv}$$

So, $\Pr[L(u) - L(v) = c_{uv} \bmod k]$
$\geq \Pr[\text{first cluster containing reps of } u \text{ and } v \text{ is good}]$
$\geq \delta_{u,v}$

But, $\mathbb{E}[\delta_{u,v}] = \delta$. This finishes claim.

High-tech reductions

- These are also gadget reductions. Start from UG instance on alphabet Σ. Goal is to come up with a PCP verifier that performs special kinds of tests. For example, for MaxCut, want a verifier that queries two bits in the proof and checks whether they're unequal (Actually, if the verifier queries some pairs with more prob than others, the reduction is to a weighted MaxCut instance.)
- First consider how to convert a good labeling to a proof. Consider a map $C : \Sigma \to \{0,1\}^l$ converting labels into a bitstring of some length l.
If $L : V_1 \cup V_2 \to \Sigma$ is a good labeling for UG instance, the corresponding proof is $(C(L(v)) : v \in V_1 \cup V_2)$.
- What should C be?
For soundness, want that verifier should be able to check if proof is in correct format by making very restricted types of queries (inequality on two bits for Max Cut)
- Intuitively, want C to be an error-correcting code which is "locally testable"
- What's the largest l that makes sense? Suppose $|\Sigma| = k$



Take $l = 2^k$. If $l > 2^k$, two columns will be identical and so extra info won't help verifier

So let $f_i : \{0,1\}^k \to \{0,1\}$ be defined as $f_i(x) = x_i$

- For $i \in [k]$, let $f_i : \{0,1\}^k \to \{0,1\}$ be defined as $f_i(x) = x_i$ and we interpret f_i as the 2^k-bit long string given by the evaluations of f_i on $\{0,1\}^k$.
- We denote this mapping $i \longmapsto f_i$ as the "long code" encoding of i.
- Verifier does two things
  1. Checks whether proof consists of long code encodings of labels
  2. Checks whether the labeling which the proof encodes is good.
- Let's go back to Max Cut. Need a 2-query $\neq$ test for dictators.
- Notation: For $\rho \in (-1,1)$, $x \in \{0,1\}^n$, define $y \sim_\rho x$ to denote the random process of choosing y s.t. $y_i \neq x_i$ w.p. $\frac{1-\rho}{2}$ $\forall i \in [n]$ independently (and $y_i = x_i$ o.w.)
- Note that $\Pr_{\substack{x \in \{0,1\}^k \\ y \sim_\rho x}} [f(y) \neq f(x)] = \frac{1-\rho}{2}$
- Definition: For $f : \{0,1\}^k \to \{0,1\}$ and $\rho \in [-1,1]$, let
  "Noise sensitivity" $\longrightarrow$ $NS_\rho(f) = \Pr_{\substack{x \in \{0,1\}^k \\ y \sim_\rho x}} [f(x) \neq f(y)]$
- From above, if f is a dictator $NS_\rho(f) = \frac{1-\rho}{2}$. If we want checking NS_ρ to be a 2-query tester for dictators, we need some sort of converse!
- If $f : \{0,1\}^k \to \{0,1\}$ and $i \in [k]$, let $Inf_i(f) = \Pr_x [f(x) \neq f(x^i)]$ where $x^i = x + \delta^i$ and $\delta^i \in \{0,1\}^k$ is 1 only at i'th position. Clearly, $Inf_i(f_i) = 1$.
- Majority is stablest: $\forall \rho \in (-1,0)$ and $\gamma > 0$, $\exists \beta$ s.t. if $f : \{0,1\}^k \to \{0,1\}$ has $Inf_i(f) \leq \beta$ $\forall i \in [k]$, then
$$NS_\rho(f) \leq \frac{1}{\pi} \cos^{-1} \rho + \gamma$$
...

- Interpretation: $\frac{1}{\pi}\cos^{-1}\rho$ is the noise sensitivity of the majority function. MIS thm above states that among the functions which have low influence on all indices, majority has the largest noise sensitivity.

The name "majority is stablest" sounds paradoxical: the thm above says maj is the function most sensitive to noise! The catch is that $\rho < 0$ above. When $\rho \geq 0$, the thm works "in reverse" to show that majority has the least noise sensitivity among balanced functions.

Pf sketch: Here, consider only the case when $f(x) = \mathrm{sgn}(\sum a_i x_i)$ (here, it's easier to convert 0/1 notation to +1/−1). Then, all influences small means all a_i's small. Now, consider $\mathrm{sgn}(a \cdot x)$ and $\mathrm{sgn}(b \cdot x)$, where $b_i = -a_i$ w.p. $\frac{1-\rho}{2}$. Then, consider the prob. that $a \cdot x$ and $b \cdot x$ have different signs (we've fixed a and b here). By a Central limit theorem, joint distribution of $a \cdot x$ and $b \cdot x$ for $x \in_U \{+1,-1\}^k$ is close to joint distribution of $a \cdot g$ and $b \cdot g$ for Gaussian g. But then $\Pr[\mathrm{sgn}(a \cdot g) \neq \mathrm{sgn}(b \cdot g)] = \frac{1}{\pi}\cos^{-1}\rho$, as we saw in our analysis for the GW theorem!

- We're now ready to give the PCP verifier.
- For $v \in V_1 \cup V_2$, let $f_v : \{0,1\}^k \to \{0,1\}$ be the portion of the proof supposed to be the long code encoding for the label of v.
- Our strategy will be to check whether f_v is a dictator by testing if its noise sensitivity is large. By MIS, we know that if $NS_\rho(f) \gg \frac{1}{\pi}\cos^{-1}\rho$, then it has influential variables.
- But we also need to check if f_v's are encodings of good labels
- We'll do both at once!
- For $\pi : [k] \to [k]$ and $x \in \{0,1\}^k$, let $x_\pi = (x_{\pi^{-1}(1)}, x_{\pi^{-1}(2)}, \ldots, x_{\pi^{-1}(k)})$

Clearly, $f_i(x) = f_{\pi(i)}(x_\pi)$.

- So, if f_v's are long code encodings of correct labels, then if $(u,v) \in E$, $f_u(x) = f_v(x_{\pi_{uv}})$

- An immediate test suggests itself: Choose random edge $(u, v) \in E$, random $x \in \{0,1\}^k$, $y \sim_\rho x_{\pi_{uv}}$ and pass if $f_u(x) \neq f_v(y)$
- Completeness: For every satisfied edge, test will pass w. prob. $\frac{1-\rho}{2}$
- Soundness: Fails badly! Suppose $f_u(x) = 1 \ \forall u \in V_1$, $f_v(x) = -1 \ \forall v \in V_2$.

  Test passes w. prob. 1 on all instances of UG!
- Reason is that we are reducing to a bipartite instance of MaxCut, which is trivial. Not checking whether individual f_v's are long code encodings
- To fix the problem, need to be a bit more subtle.
- Assume left vertices of UG instance is regular. This is wlog.
- Pick $u \in V_1$ randomly and two random edges from u, say (u, v_1) and (u, v_2).
- Pass if $f_{v_1}(x_{\pi_{uv_1}}) \neq f_{v_2}(y_{\pi_{uv_2}})$ where $y \sim_\rho x$.
- Note that tests are only on encodings of labels of V_2.
- Again, completeness is easy:

  Suppose labels satisfy $\geq 1 - \varepsilon$ fraction of edges. Prob. that both (u, v_1) and (u, v_2) satisfied is $\geq 1 - 2\varepsilon$. In this case, the test passes w. prob. $\frac{1-\rho}{2}$. So, overall, $\Pr[\text{test passes}] \geq (1-2\varepsilon) \cdot \frac{1-\rho}{2}$
- Soundness Theorem: Suppose we start with UG instance of value $< \delta$. Then above reduction produces MaxCut instance of value $< \frac{\cos^{-1}\rho}{\pi} + \varepsilon$, where $\delta = \delta(\varepsilon)$.
- So we have reduction from $[1-\delta, \delta]$-Gap-UG to ... $(1-\rho)$ ... So assuming

$\left[\frac{1-\rho}{2}(1-\varepsilon),\ \frac{\cos^{-1}\rho}{\pi}+\varepsilon\right]$ - Gap- Max Cut $\forall \rho \in [-1, 0), \varepsilon > 0$,

UGC, it's NP-hard to approx Max Cut to factor

$> \min_{\rho \in (-1,0)} \frac{2\cos^{-1}\rho}{\pi(1-\rho)} > 0.878$, the GW factor!

- Proof sketch of soundness theorem:

By def'n of verifier,

$$\Pr[\text{verifier accepts}] = \mathop{\mathbb{E}}_{u} \mathop{\mathbb{E}}_{v_1, v_2 \sim u} \mathop{\mathbb{E}}_{x} \mathop{\mathbb{E}}_{y \sim_\rho x} \left[\frac{1}{2} - \frac{1}{2} f_{v_1}(x_{\pi_{uv_1}}) f_{v_2}(y_{\pi_{uv_2}})\right]$$

Suppose this is $\geq \frac{\cos^{-1}\rho}{\pi} + \varepsilon$. Then, for $\geq \frac{\varepsilon}{2}$ fraction of u's

$$\mathop{\mathbb{E}}_{v_1, v_2 \sim u} \mathop{\mathbb{E}}_{x, y} \left[\frac{1}{2} - \frac{1}{2} f_{v_1}(x_{\pi_{uv_1}}) f_{v_2}(y_{\pi_{uv_2}})\right] \geq \frac{\cos^{-1}\rho}{\pi} + \frac{\varepsilon}{2}$$

(by Markov-Type argument). For such a u, we have:

$$\mathop{\mathbb{E}}_{\substack{x \\ y \sim_\rho x}} \left[\frac{1}{2} - \frac{1}{2} \mathop{\mathbb{E}}_{v_1 \sim u}[f_{v_1}(x_{\pi_{uv_1}})] \mathop{\mathbb{E}}_{v_2 \sim u}[f_{v_2}(y_{\pi_{uv_2}})]\right]$$

$$= \Pr_{\substack{x \\ y \sim_\rho x}} [g_u(x) \neq g_u(y)] = NS_\rho(g_u) \geq \frac{\cos^{-1}\rho}{\pi} + \frac{\varepsilon}{2}$$

where $g_u(x) = \mathop{\mathbb{E}}_{v \sim u}[f_v(x_{\pi_{uv}})]$. Ideally, $g_u = f_u$, but anyways, we can now use MIS theorem to get that some variable i_u is influential for g_u.

We now give a labeling for UG instance that satisfies $\geq \delta$ fraction of edges. If $u \in V_1$, label u with i_u, the influential var for g_u. For $v \in V_2$, the argument is more complicated. It's possible to show that if g_u has an influential variable i_u, then for at least $\frac{\beta}{2}$ fraction of u's nbrs v (β is the param from MIS thm), f_v has $\pi_{uv}(i_u)$ as an influential variable. Moreover, these f_v's don't have too many influential variables, and so, if we make a list of all the influential variables of ... one of them randomly, there's a significant ... satisfying the edge (u,v).

prob. of choosing $\pi_{uv}(i_u)$, and so [illegible]

In this way, $\geq \delta$ fraction of the edges can be satisfied.