15-859: Information Theory and Applications in TCS **Spring 2013**

Lecture 15: Applications of Graph Entropy

March 22, 2013

Lecturer: Mahdi Cheraghchi *Scribe: Euiwoong Lee*

1 Recap

- Graph Entropy: Given $G = (V, E)$, we define $H(G) = \min I(X;Y)$ over joint distributions (X, Y) where X is a uniformly random vertex in V, and Y is an independent subset of V that contains X.
- $H(G_1 \cup G_2) \le H(G_1) + H(G_2)$.
- $H(G_1) \le H(G_1 \cup G_2)$
- If $G_1, ..., G_k$ are connected component of G, $H(G) = \sum_{i \in [k]} \rho_i H(G_i)$ where $\rho_i := \frac{|V(G_i)|}{|V(G)|}$.

2 Number of bipartite graphs to cover the complete graph

Suppose that we have the complete graph $K_n = (V, \binom{V}{2})$. We want to *cover* K_n by l bipartite graphs, $G_1, ..., G_l$ in a sense that

- For each i, $G_i = (V, E_i)$ is a bipartite graph.
- For each $(u, v) \in \binom{V}{2}$, $(u, v) \in E_i$ for some i. In other words, $K_n = G_1 \cup ... \cup G_l$.

Question: What is the minimum number l of bipartite graphs needed to cover K_n?

Construction: Identify each vertex with a binary string of length $\lceil \log n \rceil$. The ith bipartite graph connects every two vertices whose binary representations differ at the ith position. It is easy to see that these $\lceil \log n \rceil$ bipartite graphs cover all the pairs.

Lower bound: In the previous lecture, we saw that

- $H(K_n) = \log n$
- $K_n = G_1 \cup ... \cup G_l$ implies $H(K_n) \le \sum_i H(G_i)$
- $H(G_i) \le 1$

Therefore, $\log n = H(K_n) \le \sum_i H(G_i) \le l \le \lceil \log n \rceil$. Generally, given a graph $G = (V, E)$, the same upper and lower bound techniques work to show that $H(G) \le l \le \lceil \log \chi(G) \rceil$ (for the upper bound, identify each color with a binary string). $\log \chi(G)$, which is always at least $H(G)$, gives one intuition about $H(G)$, even though the difference can be made arbitrarily large.

3 Perfect Hash Families

Setting: A database where each *file* is an element of $[N]$. A hash function maps a file to a much smaller domain; $h : [N] \to [b]$ where $b \ll N$.

Suppose we have a hash family $\mathcal{H} = \{h_1, ..., h_t\}$ where for each i, $h_i : [N] \to [b]$ is a hash function. Our goals is to design $\mathcal{H}$ such that it can differentiate between up to k files ($k < b$). In other words,

$$\forall S \subseteq [N], |S| = k : \exists h \in \mathcal{H} \text{ such that } h \text{ is injective on } S$$

If we think $\mathcal{H}$ as a $N \times t$ matrix (each row corresponds to a file x, each column corresponds to a hash function h, and $\mathcal{H}(x, h) = h(x)$), we require that for every choice of k rows $(x_1, ..., x_k)$, there exists a column h such that $h(x_1), ..., h(x_k)$ are pairwise distinct. We call $\mathcal{H}$ k-perfect hash familiy if the above condition is satisfied. The question is, how small can t be?

3.1 Upper bound

Claim 3.1. *Assume $b \geq k^2$. Then $t = O(k \log \frac{N}{k})$ suffices.*

Proof. Pick each $h_i : [N] \to [b]$ uniformly and independently at random. Fix $S \subseteq [N], |S| = k$.

$$\begin{aligned}
& Pr[h_1 \text{ is injective on } S] = 1 \cdot \frac{b-1}{b} \cdot ... \cdot \frac{b-k+1}{b} \geq (1 - \frac{k}{b})^k \geq (1 - \frac{1}{k})^k \geq \frac{1}{4} \\
\Rightarrow \quad & Pr[\forall i, h_i \text{ is not injective on } S] \leq (\frac{3}{4})^t \\
\Rightarrow \quad & Pr[\mathcal{H} \text{ is not } k\text{-perfect}] \leq \binom{N}{k} (\frac{3}{4})^t \leq (\frac{Ne}{k})^k (\frac{3}{4})^t = 2^{O(k \log(N/k)) - \Omega(t)}
\end{aligned}$$

The probability can be made less than 1 for some $t = O(k \log \frac{N}{k})$. □

3.2 Lower bound

Claim 3.2. *For all $k \geq 2$, $t \geq \frac{\log N}{\log b}$*

Proof. It follows from the pigeonhole principle: $\forall x_1 \neq x_2 \in [N]$, we must have $(h_1(x_1), ..., h_t(x_1)) \neq (h_2(x_1), ..., h_t(x_2))$. Therefore, $N \leq b^t \Rightarrow t \geq \frac{\log N}{\log b}$. □

There is a stronger lower bound due to Fredman Komlós in 1984.

Theorem 3.3. $t \geq \frac{b^{k-1}}{b(b-1)...(b-k+2)} \frac{\log(N-k+2)}{\log(b-k+2)}$

Proof. Assume $b|N$. Define $G = (V, E)$ such that

- $V = \{(D, x) : D \subseteq [N], |D| = k - 2, x \in [N] - D\}$.
- $E = \{((D, x_1), (D, x_2)) : \forall D, x_1 \neq x_2\}$.

G has $\binom{N}{k-2}$ connected components, each is a clique (of size $N-k+2$) corresponding to some D. From the last lecture, $H(G) = H(\text{each component}) = \log(N-k+2)$.

Given a k-perfect hash family $\mathcal{H}$, we construct $\{G_h\}$ such that $G = \cup_{h\in\mathcal{H}} G_h$. The construction is as the following.

- $V(G_h) = V(G)$.
- $E = \{((D, x_1), (D, x_2)) : h \text{ is injective on } D \cup \{x_1, x_2\}\}$.

Every $\{(D, x_1), (D, x_2)\} \in E(G)$ is covered by G_h where h is injective on $D \cup \{x_1, x_2\}$, so $G = \cup_{h\in\mathcal{H}} G_h$.

Now we want to argue that each $H(G_h)$ is small. Fix h. For a choice of D,

- If h is not injective on D, $H(G_{h,D}) = 0$ where $G_{h,D}$ indicates the connected component of G_h corresponding to D.
- If h is injective on D, $G_{h,D}$ is $(b-k+2)$-partite. This can be shown by defining $A_i := \{(D, x) : h(x) = i\}$ for each $i \notin h(D)$. Since h is injective there are exactly $b-k+2$ choices of i, and there is no edge between (D, x_1) and (D, x_2) if $h(x_1) = h(x_2)$. From the last lecture, $H(G_h, D) \le \log(b-k+2)$.

In any case, $H(G_h, D) \le \log(b-k+2)$ and $H(G_h) \le \log(b-k+2)$. Together with $H(G) = \log(N-k+2)$, we can conclude that $t \ge \frac{\log(N-k+2)}{\log(b-k+2)}$.

To get a better bound, we want to show that G_h has a large fraction of isolated vertices. Define p the probability that a uniform random vertex of G_h is isolated. Let $\mathcal{E}$ be the set of isolated vertices. The same argument shows that $H(G_h - \mathcal{E}) \le \log(b-k+2)$ as well, so we have

$$H(G_h) = pH(\mathcal{E}) + (1-p)H(G_h - \mathcal{E}) \le (1-p)\log(b-k+2)$$

Therefore, an upper bound of $1-p$ is needed to achieve a better lower bound on t. (D, x) is isolated if and only if h is not injective on $D \cup \{x\}$, so p is the probability over uniformly chosen $(k-1)$-subset S that h is not injective on S.

Claim 3.4. *Without loss of generality, we can assume that* $|h^{-1}(1)| = ... = |h^{-1}(b)| = \frac{N}{b}$. *In other words, maximum p (minimum* $1-p$*) is achieved by* $|h^{-1}(1)| = ... = |h^{-1}(b)| = \frac{N}{b}$.

Proof. Assume that $|h^{-1}(1)| > |h^{-1}(2)| + 1$. Take any x such that $h(x) = 1$ and change h such that $h(x) = 2$.

$$\begin{aligned} p = \Pr_S[h \text{ is injective on } S] \quad &= \quad \Pr(x \in S)\Pr_S[h \text{ is injective on } S | x \in S] + \\ &\quad\;\; \Pr(x \notin S)\Pr_S[h \text{ is injective on } S | x \notin S] \end{aligned}$$

Since we only changed $h(x)$, the second term does not change. The first term increases since given that $x \in S$, $S - \{x\}$ needs to be disjoint from $h^{-1}(h(x))$ and the size of it became smaller.

□

Now, $1-p \leq 1 \cdot \frac{b-1}{b} \cdot ... \cdot \frac{b-k+2}{b}$ and $H(G_h) \leq (1-p)\log(b-k+2)$ Therefore,

$$t \geq \frac{H(G)}{\max_h H(G_h)} \geq \frac{\log(N-k+2)}{(1-p)\log(b-k+2)} \geq \frac{b^{k-1}}{b(b-1)...(b-k+2)} \frac{\log(N-k+2)}{\log(b-k+2)}$$

$\square$