Lecture Notes of Lovasz Local Lemma

Fu Lingyue

April 7, 2020

1 Dependency Graph

The dependency graph describes the dependencies between events.

$E_1, E_2, \ldots, E_n$ be n events on a probability space Ω. The **dependency graph** is a graph $G = (V, E)$ on the set of vertices $V = 1, 2, \ldots, n$(corresponding to $E_1, E_2, \ldots, E_n$ if for each $1 \leq i \leq n$, E_i is mutually independent of all events $\{E_j : (i, j) \notin E\}$.

My question here: Why such definition is completed?

For example, if E_1 only has neighbor E_2, E_4, then E_1 is independent of $E_3 \cap E_5$, i.e.,

$$P(E_1 \cap E_3 \cap E_5) = P(E_1) \cdot P(E_3 \cap E_5).$$

My question here: Why dependency graph is not unique?

In this case, we can conclude that E_i are pairwise independent. Thus graph A,B have the same meaning of dependency. Because of this, we raise a definition of minimum dependency graph(See definition in YANG Zonghan's note).

2 Lovasz Local Lemma(1975)

We have N events $E_1, E_2, \ldots, E_N$, and $G = (V, E)$ is their dependency graph.

Denotation In order to describe the theorem more clearly, we introduce some notations.

1. $[N] := S$, where $|S| = N$.

2. Based on (1), we denote $\bigcap_{[s]} E_i := \bigcap_{1 \leq i \leq s} E_i$.

Theorem 1. *If graph G satisfies:*

1. $\exists P \in (0, 1), \forall i\ P(E_i) \leq p$.

2. $\max_{i \in [V]}(\deg(V_i)) \leq d$.

3. $4dp \leq 1$.

then $P(\bigcap_{i \in [N]} E_i^C) > 0$.

Proof. Use Induction to prove the theorem. We apply induction on the set $S := [s](s = 0, 1, \ldots, N)$. The propositions that need to be proved are

$$P(\bigcap_{[s]} E_i^C) > 0. \tag{1}$$

$$\forall k \in V \backslash S, P(E_k \cap \bigcap_{[s]} E_i^C) \leq 2pP(\bigcap_{[s]} E_i^C). \tag{2}$$

Let'e begin our induction.

Basis Step   When $s = 0, S = \varnothing$. $P(\bigcap_{[0]} E_i^C) = P(\Omega) = 1 > 0$, thus (1) holds. We can also obtain

$$P(E_k \cap \bigcap_{[0]} E_i^C) = P(E_k \cap \Omega)$$

$$= P(E_k) \leq p < 2p \quad = 2pP(\Omega)$$

$$= 2pP(\bigcap_{[0]} E_i^C).$$

Hence, (2) also holds when $S = \varnothing$.

Induction Step   Assume that (1)(2) hold when $s = 0, 1, \ldots, t - 1$. Then we have to prove their correctness when $s = t$.

First, if $d = 0$, then the graph has no edges. In this case, (1)(2) holds. Thus we discuss those cases $d \neq 0$.

Write $P(\bigcap_{[t]} E_i^C)$ as

$$P(\bigcap_{[t]} E_i^C) = \frac{P(\bigcap_{[t]} E_i^C)}{P(\bigcap_{[t-1]} E_i^C)} \cdot \frac{P(\bigcap_{[t-1]} E_i^C)}{P(\bigcap_{[t-2]} E_i^C)} \cdots \frac{P(E_1^C)}{P(\varnothing^C)}$$

It is clear that

$$P(E_k \cap \bigcap_{[s]} E_i^C) + P(E_k^C \cap \bigcap_{[s]} E_i^C) = P(\bigcap_{[s]} E_i^C) \tag{3}$$

So let's plug (3) on our hypothesis (2):

$$P(E_k^C \cap \bigcap_{[s]} E_i^C) = P(\bigcap_{[s]} E_i^C) - P(E_k \cap \bigcap_{[s]} E_i^C)$$

$$\geq (1 - 2p)P(\bigcap_{[s]} E_i^C). \tag{4}$$

According to the hypothesis(1), $P(\bigcap_{[s]} E_i^C) > 0 (s < t)$. We can transpose (4) into

$$\frac{P(E_k^C \cap \bigcap_{[s]} E_i^C)}{P(\bigcap_{[s]} E_i^C)} \geq 1 - 2p(s < t). \tag{5}$$

Let E_k be E_s, thus we can conclude that

$$\frac{P(\bigcap_{[s]} E_i^C)}{P(\bigcap_{[s-1]} E_i^C)} \geq 1 - 2p.$$

Hence, $P(\bigcap_{[t]} E_i^C) \geq (1-2p)^t$. Also, $2p \leq \frac{1}{2d} \leq \frac{1}{2}(d \geq 1)$, thus $1 - 2p > 0$. Therefore, (1) holds when $s = t$, i.e.,

$$P(\bigcap_{[t]} E_i^C) > 0.$$

Then we have to prove (2) holds when $s = t$. Denote $S_1 := \{j \in [t] : (j, k) \in E\}$, $S_2 := S \backslash S_1 = \{j \in [t] : (j, k) \notin E\}$.

If $S_1 = \varnothing$, then E_k is independent to $\bigcap_{[t]} E_i$, i.e.,

$$P(E_k \cap \bigcap_{[s]} E_i^C) = P(E_k) \cdot P(\bigcap_{[s]} E_i^C) \leq pP(\bigcap_{[s]} E_i^C) < 2pP(\bigcap_{[s]} E_i^C).$$

If $S_1 \neq \varnothing$, then $|S_2| < t$. Use the similar trick, write $P(E_k \cap \bigcap_{[s]} E_i^C)$ as

$$
\begin{aligned}
P(E_k \cap \bigcap_{[s]} E_i^C) &= P(E_k \cap \bigcap_{S_1} E_i^C \cap \bigcap_{S_2} E_i^C) \\
&= \frac{P(E_k \cap \bigcap_{S_1} E_i^C \cap \bigcap_{S_2} E_i^C)}{P(\bigcap_{S_1} E_i^C \cap \bigcap_{S_2} E_i^C)} \cdot P(\bigcap_{[s]} E_i^C)
\end{aligned}
\tag{6}
$$

Denote $F_{S_1} := E_k \cap \bigcap_{S_1} E_i^C$ and $F_{S_2} := E_k \cap \bigcap_{S_2} E_i^C$. Then we can write a equation based on (6)

$$
\begin{aligned}
\frac{P(E_k \cap \bigcap_{[s]} E_i^C)}{P(\bigcap_{[s]} E_i^C)} &= \frac{P(E_k \cap F_{S_1} \cap F_{S_2})}{P(F_{S_1} \cap F_{S_2})} \\
&= \frac{P(E_k \cap F_{S_1} \cap F_{S_2})}{P(F_{S_2})} \cdot \frac{P(F_{S_2})}{P(F_{S_1} \cap F_{S_2})}
\end{aligned}
\tag{7}
$$

Then we can write the inequality

$$\frac{P(E_k \cap F_{S_1} \cap F_{S_2})}{P(F_{S_2})} \leq \frac{P(E_k \cap F_{S_2})}{P(F_{S_2})}$$

For E_k is independent of $E_i \in S_2$, thus $P(E_k \cap F_{S_2}) = P(E_k) \cdot P(F_{S_2})$. Hence, we can obtain

$$\frac{P(E_k \cap F_{S_1} \cap F_{S_2})}{P(F_{S_2})} \leq \frac{P(E_k \cap F_{S_2})}{P(F_{S_2})} = P(E_k) \leq p. \tag{8}$$

The another inequality is

$$
\begin{aligned}
\frac{P(F_{S_1} \cap F_{S_2})}{P(F_{S_2})} &= 1 - \frac{P(\bigcap_{S_1} E_i \cap F_{S_2})}{P(F_{S_2})} \\
&\geq 1 - \Sigma_{S_1} \frac{P(E_i \cap F_{S_2})}{P(F_{S_2})}
\end{aligned}
\tag{9}
$$

According to the hypothesis, for $|S_2| < t$,

$$\frac{P(E_i \cap F_{S_2})}{P(F_{S_2})} \le 2p.$$

Hence, we can limit the second component in (7) to

$$\frac{P(F_{S_2})}{P(F_{S_1} \cap F_{S_2})} \le \frac{1}{1 - \Sigma_{S_1} \frac{P(E_i \cap F_{S_2})}{P(F_{S_2})}}$$

$$\le \frac{1}{1 - |S_1| \cdot 2p} \le \frac{1}{1 - 2pd} \le 2. \tag{10}$$

Combine (8) and (10), we finally prove the second proposition when $s = t$:

$$\frac{P(E_k \cap \bigcap_{[s]} E_i^C)}{P(\bigcap_{[s]} E_i^C)} \le p * 2 = 2p.$$

$\square$

3 LLL's application on k-SAT

3.1 Description of the problem

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a finite set of boolean variables. Then we need to introduce some definitions for better explanation.

Atom An atom variable, like x_1, x_2 in X.

Assignment Similar to that in mathematical logic, we can give every variable a boolean value, called an assignment.

Clause A disjunction of literals, like $x_1 \vee x_3 \vee x_4$.

CNF(Conjunctive Normal Form) If a formula only consists of $\wedge$s of several clause.
   We can easily rewrite a formula into CNF. And we write $\vee$ as $+$, $\wedge$ as $\cdot$ and $\neg$ as $-$ in this section.
   Let $F := F_1 \cdot F_2 \ldots F_m$ be a CNF on X, and every clause F_i as at least k different atoms. The k-SAT problem is to decide whether or not F is satisfiable, i.e., whether there exists an assignment β such that F evaluates to true.
   It has been proved that k-SAT problem is a NP-complete problem(you can use induction to prove it).

3.2 Use LLL to solve one situation

Note that LLL can only prove the situation that every clause exactly have k atoms.

Theorem 2. *If in a CNF F, no variables appears in more than $\frac{2^k}{4k}$ clauses, then F has some satisfying assignments.*

Proof. The whole space Ω is those assignment lists, like $(T, F, T, \ldots, F, T)$. And every assignment is a sample points. Let events E_i be the assignments such that F_i evaluates to false. Here we get a sequence of events $E_1, E_2, \ldots, E_m$.

Then we have to give an upper bound of $P(E_i)$. For every F_i has exactly k atoms,

$$P(E_i) \leq \frac{1}{2^k} := p.$$

And every variable at most occurs $\frac{2^k}{4k}$ times in other clauses, thus

$$\deg(V_i) \leq \Sigma_{[k]}(\frac{2^k}{4k} - 1) \leq 2^k/4.$$

Define $d = \max_{[n]} \deg(V_i) \leq 2^k/4$. Hence, we get the third condition of LLL:

$$4pd \leq 4 \cdot \frac{1}{2^k} \cdot 2^k/4 = 1.$$

Use LLL, we can conclude that

$$P(\bigcap_{[m]} E_i^C) > 0,$$

i.e., the set $\bigcap_{[m]} E_i^C$ is not empty. In this case, we can infer that there exists an assignment can evaluates F to true. $\square$

3.3 From exactly k to $k+$

It is clear that LLL can prove the "exactly k" cases, but the condition of k-SAT is "at least k" different atoms.

In order to solve such problem, we have to delete some variables in clauses. For example, we have to calculate the CNF

$$(x_1 + x_2 + x_3 + x_4 + x_5 + x_6)(x_5 + x_6 + x_7 + x_8 + x_9 + x_{10})(x_1 + x_2 + x_4 + x_5 + x_{11} + x_{12} + x_{13})$$

Here $k = 6$. We can draw a dependency graph G. When we remove the x_5 in the third clause, then we can generate a new graph G' and new formula F'

$$(x_1 + x_2 + x_3 + x_4 + x_5 + x_6)(x_5 + x_6 + x_7 + x_8 + x_9 + x_{10})(x_1 + x_2 + x_4 + x_{11} + x_{12} + x_{13}).$$

Then we can use LLL to new graph G', and it satisfies every variables only occurs at most $\frac{2^k}{4k} = 2$ times. In this case, we have an assignment for F'.

Then add x_5 into F'_3, for the internal operation is $\vee$, F_3 is still true. In this way, we can find an assignment for F as well.

By the way, if those removed variables are not occured in other clauses, then there is even no need to delete edge in G. If removed variables is too much, then there is hard to deal with deleting edges, but also it is meaningless for us to remove too much variables.

Therefore, in my opinion, the best choice is let "k" in LLL be the minimum variable numbers in those F_i, so that we can delete variables as little as possible. On the other hand, when we choose variables in $|F_i| > k$, we'd better choose those atoms occur too much time. In this case, we can apply an exactly-k into more $k+$ formulas.