Lovász Local Lemma

Let $E_1, \ldots, E_n$ be a set of "bad" events.

Say want to show $\Pr(\bigcap_{i=1}^n \overline{E_i}) > 0$ "pos prob that nothing bad happens".

2 cases where easy

1. $E_i$ are mutually indep

2. $\sum_{i=1}^n \Pr(E_i) < 1$ union bound suffices

LLL is clever combination

Defn: $E$ mutually indep of $E_1, \ldots, E_n$ if for any subset $I \subseteq \{1, \ldots, n\}$

$$\Pr(E \mid \bigcap_{j \in I} \overline{E_j}) = \Pr(E)$$

Defn: A dependency graph for $E_1, \ldots, E_n$ is $G = (V, E)$

$V = \{1, \ldots, n\}$ and $E_i$ mutually indep of $\{E_j \mid (i, j) \notin E\}$
Thm Lovász Local Lemma

Let $E_1, \ldots, E_n$ be set of events s.t.

1. $\Pr(E_i) < p \quad \forall i$

2. the max degree in dependency graph is $d$

3. $4dp < 1$

Then $\Pr(\bar{E}) > 0$

Proof

Show $\Pr(\cap_{i \in S} \bar{E}_i) > 0$ and $\Pr(E_k | \cap_{i \in S} \bar{E}_i) \leq 2p \quad \forall k$

by induction on $|S|$

Base case: $|S| = 1$

$\Pr(\bar{E}_i) = \Pr(E_i) = 1 - p > 0$

Case 1: no edge $k - i$

$\Pr(E_k | \bar{E}_i) = \Pr(E_k) \leq p$

Case 2: $\exists \text{ some } k - i$

$\Pr(E_k | \bar{E}_i) = \frac{\Pr(E_k \cap \bar{E}_i)}{\Pr(\bar{E}_i)} \leq \frac{p}{1 - p} < 2p$

$p < \frac{1}{2}$

Induction step:

IH: $|T| < s \quad \Pr(\cap_{i \in T} \bar{E}_i) > 0 \quad \Pr(E_k | \cap_{i \in T} \bar{E}_i) \leq 2p$

wlog $S = \{1, \ldots, s\}$

$\Pr(\cap_{i=1}^s \bar{E}_i) = \Pr(E_1) \Pr(E_2 | \bar{E}_1) \Pr(E_3 | E_2, \bar{E}_2) \ldots \Pr(E_s | E_1, \ldots, \bar{E}_{s-1})$

$= \prod_{i=1}^s (1 - \Pr(E_i | \cap_{j<i} \bar{E}_j)) \geq \prod_{i=1}^s (1 - 2p) > 0$
Next, prove \( \Pr(E_k \mid \bigcap_{i \in S} \bar{E}_i) \leq a \cdot \rho \)

If \( S_a = S \),

\( \Pr(E_k \mid \bigcap_{i \in S} \bar{E}_i) = \Pr(E_k) \leq \rho \)

So assume \( S_a < S \)

\[ F_S = \bigcap_{i \in S} \bar{E}_i \quad F_{S_1} = \bigcap_{i \in S_1} \bar{E}_i \quad F_{S_a} = \bigcap_{i \in S_a} \bar{E}_i \]

\[ \Pr(E_k \mid F_S) = \frac{\Pr(E_k \cap F_S)}{\Pr(F_S)} = \frac{\Pr(E_k \cap F_S, 1F_{S_a}) \Pr(F_{S_a})}{\Pr(F_S, 1F_{S_a}) \Pr(F_{S_a})} = \frac{\Pr(E_k \cap F_S, 1F_{S_a})}{\Pr(F_S, 1F_{S_a})} \]

\( \Pr(E_k \cap F_S, 1F_{S_a}) \leq \Pr(E_k \mid F_{S_a}) \Pr(E_k) \leq \rho \)

\[ = 1 - \Pr(\bigcup_{i \in S} E_i \mid F_{S_a}) \]

\[ \geq 1 - \sum_{i \in S} \Pr(E_i \mid F_{S_a}) \quad \text{union bound} \]

\[ \geq 1 - \sum_{i \in S} \Pr(E_i \mid F_{S_a}) \]

\[ \geq 1 - a \cdot \rho \geq \frac{1}{\alpha} \]

\( \text{induction hypothesis, } |S_a| < |S| \)
\[ \Pr(E_k | E_S) \leq \frac{p}{q^*} = 2^p \]

Application 1: k-SAT

Given k-SAT formula \( n \) vars, \( m \) clauses

**Thm** If no var appears \( n^* > T = \frac{\alpha^*}{4k} \) clauses, then formula has satisfying assignment

**Proof** Consider random assignment where each var is true with prob \( \frac{1}{2} \) independently.

\( E_i \) event that clause \( i \) not satisfied

- \( p = \Pr(E_i) = 2^{-k} \)
- \( E_i \) mutually indep of clauses that don't share vars

\( d \leq kT = 2^{k-a} \) \( \Rightarrow \) \( 4dp \leq 1 \) \( \Rightarrow \) \( \Pr(\cap_i \overline{E_i}) > 0 \) Satisfying assignment
Efficient Algs for finding outcome guaranteed by LLL

\[ \text{Thm} \]

S set of \( m \) length \( k \) clauses, s.t. support of each clause intersects \( \leq 2^{k-c} \) clauses \( [c \text{ is a sufficiently large constant}] \)

Then the clauses in \( S \) simultaneously satisfiable

**Proof:**

*Constructive version*

Fix ordering \( C_1, \ldots, C_m \) of clauses

Solve SAT

Choose random assignment of vars \( x_1, \ldots, x_n \)

while \( \exists \) clause not satisfied, pick arbitrary clause that is not satisfied, and assign new random values to the vars init
Claim: This terminates in poly time w.h.p.

Whenever random bit
needed, say for $x_i$, go to $x_i$
column & take next "unused" bit.

Consider sequence of clauses alg
reamples
$c_1, c_2, \ldots, c_t$

Describe part of execution (source of random bits)
relevant to $C_t$ using tree whose nodes are clauses

built inductively $i=t\ldots 1$

$T_t^+$ single node: $C_t$

$T_t^i$ derived from $T_t^{i+1}$

if $C_i \cap C_j \neq \emptyset \text{ and } C_j \in T_t^{i+1}$, then $T_t^i = T_t^{i+1}$
Claim: $T'_1$ uniquely determines which locations in $R$ the random values for vars of each clause are taken from.

∀ clauses in tree

Pf. Postorder traversal

Observation: If 2 clauses $C_i$ & $C_j$ $i<j$ in tree are at same depth in tree, then they are disjoint.

Consider which came first.

Tree is feasible if all clauses in tree are not satisfied by the corresponding values in $R$.

Pr(tree $T'_1$ with $q$ clauses is feasible) = $p^q$

$p = 2^{-k}$

(tree being feasible doesn't mean actually generated)
If alg runs for $q m$ steps, $\implies R$ contains feasible tree of size $\geq q$

$q m$ clauses $\implies$ some clause repeated $\geq q$ times

all occurrences in same tree

$\implies \Pr(\text{alg runs for } \geq q m \text{ steps}) \leq \Pr( R \text{ contains feasible tree of size } \geq q )$

**Claim:** \# legally labelled trees of size $q \leq m \binom{d q}{q-1}$

- node labelled w/ clauses
- adjacent nodes overlap
- nodes at same level don't intersect

**Proof of claim:** Fix ordering of clauses

- $m$ choices for root
- for each clause chosen, append vector of length $d$ of 0's
- scan coordinates one-by-one, setting entries corresponding to "children" clauses (and append d coordinates for each)

any legal tree can be represented this way and
any vector corresponds to at most one legal tree
\[ \Rightarrow \text{\# legal trees using \( n \) nodes} \leq m \binom{n}{q-1} \leq m(d^q)^{(q-1)/q} \]

Stirling's Formula

\[ \Rightarrow \Pr_r(\exists \text{feasible legal tree of size} \geq Q \text{ or more}) \]

\[ \leq m \sum_{q=2}^n (d^q)^q p^q \leq m \frac{(dpe)^Q}{1-dpe} \]

\[ d = 2^{-\epsilon} \quad p = 2^{-k} \]

\[ = o(1) \quad \text{for} \quad Q = 2 \log m \]

\[ \Rightarrow \Pr(\# \text{ clause resamplings} \geq 2 \log m) = o(1) \]

Can be viewed as "entropy compression" argument.