Logistics

- Problem Set 4 is due Monday 4/22 at 11:59pm.
- No quiz this week.
Summary

Last Class:

• Leverage score intuition.

• Connection to spectral graph sparsification

• Connection to effective resistances in electrical networks. **Note:** I am not going to finish this full derivation – see Lecture 17 slides if you are interested.
Summary

Last Class:

• Leverage score intuition.
• Connection to spectral graph sparsification
• Connection to effective resistances in electrical networks. **Note:** I am not going to finish this full derivation – see Lecture 17 slides if you are interested.

Today:

• New unit: Markov Chains.
• Markov chain based algorithms for 2-SAT and 3-SAT.
Markov Chain Definition

- A **discrete time stochastic process** is a collection of random variables $X_0, X_1, X_2, \ldots$.
- A discrete time stochastic process is a **Markov chain** if it is memoryless:

  \[ \Pr(X_t = a_t|X_{t-1} = a_{t-1}, \ldots, X_0 = a_0) = \Pr(X_t = a_t|X_{t-1} = a_{t-1}) = P_{a_{t-1}, a_t}. \]

**Question:** In a Markov chain, is $X_t$ independent of $X_{t-2}, X_{t-3}, \ldots, X_0$?
A Markov chain $X_0, X_1, \ldots$ where each $X_i$ can take $m$ possible values, is specified by the transition matrix $P \in [0, 1]^{m \times m}$ with

$$P_{j,k} = \Pr(X_{i+1} = k | X_i = j).$$
A Markov chain $\mathbf{X}_0, \mathbf{X}_1, \ldots$ where each $\mathbf{X}_i$ can take $m$ possible values, is specified by the transition matrix $P \in [0, 1]^{m \times m}$ with

$$
P_{j,k} = \Pr(\mathbf{X}_{i+1} = k | \mathbf{X}_i = j).$$

Let $q_i \in [0, 1]^{1 \times m}$ be the distribution of $\mathbf{X}_i$. Then $q_{i+1} = q_i P$. 

\[ 
P = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 \\ 0.25 & 0.25 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{bmatrix} \]
A Markov chain $X_0, X_1, \ldots$ where each $X_i$ can take $m$ possible values, is specified by the transition matrix $P \in [0, 1]^{m \times m}$ with

$$P_{j,k} = \Pr(X_{i+1} = k \mid X_i = j).$$

Let $q_i \in [0, 1]^{1 \times m}$ be the distribution of $X_i$. Then $q_{i+1} = q_i P$. 

\[
\begin{align*}
q_0 & = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix} \\
\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 \\
0.25 & 0.25 & 0.5 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0.5 & 0.5 & 0 & 0 \end{bmatrix} & = \begin{bmatrix} 0.5 & 0.5 & 0 & 0 & 0 \end{bmatrix}
\end{align*}
\]
A Markov chain $X_0, X_1, \ldots$ where each $X_i$ can take $m$ possible values, is specified by the transition matrix $P \in [0, 1]^{m \times m}$ with

$$P_{j,k} = \Pr(X_{i+1} = k|X_i = j).$$

Let $q_i \in [0, 1]^{1 \times m}$ be the distribution of $X_i$. Then $q_{i+1} = q_i P$. 

\[
\begin{bmatrix}
  q_1 \\
  .5 & .5 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
  P \\
  .5 & .5 & 0 & 0 \\
  .25 & .25 & .5 & 0 \\
  0 & 1 & 0 & 0 \\
  0 & .5 & .5 & 0 \\
\end{bmatrix}
= 
\begin{bmatrix}
  q_2 \\
  .375 & .375 & .25 & 0 \\
\end{bmatrix}
\]
Often viewed as an underlying state transition graph. Nodes correspond to possible values that each $X_i$ can take.
Graph View

Often viewed as an underlying state transition graph. Nodes correspond to possible values that each \( X_i \) can take.

The Markov chain is **irreducible** if the underlying graph consists of single strongly connected component.
Motivating Example: Find a satisfying assignment for a 2-CNF formula with $n$ variables.

$$(x_1 \lor \bar{x}_2) \land (\bar{x}_1 \lor \bar{x}_3) \land (x_1 \lor x_2) \land (x_4 \lor \bar{x}_3) \land (x_4 \lor \bar{x}_1)$$
Motivating Example: Find a satisfying assignment for a 2-CNF formula with $n$ variables.

$$(x_1 \lor \bar{x}_2) \land (\bar{x}_1 \lor \bar{x}_3) \land (x_1 \lor x_2) \land (x_4 \lor \bar{x}_3) \land (x_4 \lor \bar{x}_1)$$

A simple ‘local search’ algorithm:

1. Start with an arbitrary assignment.
2. Repeat $2mn^2$ times, terminating if a satisfying assignment is found:
   • Chose an arbitrary unsatisfied clause.
   • Pick one of the variables in the clause uniformly at random, and switch the assignment of the variable.
3. If a valid assignment is not found, return that the formula is unsatisfiable.
Motivating Example: Find a satisfying assignment for a 2-CNF formula with $n$ variables.

$$(x_1 \lor \bar{x}_2) \land (\bar{x}_1 \lor \bar{x}_3) \land (x_1 \lor x_2) \land (x_4 \lor \bar{x}_3) \land (x_4 \lor \bar{x}_1)$$

A simple ‘local search’ algorithm:

1. Start with an arbitrary assignment.
2. Repeat $2mn^2$ times, terminating if a satisfying assignment is found:
   - Chose an arbitrary unsatisfied clause.
   - Pick one of the variables in the clause uniformly at random, and switch the assignment of the variable.
3. If a valid assignment is not found, return that the formula is unsatisfiable.

Claim: If the formula is satisfiable, the algorithm finds a satisfying assignment with probability $\geq 1 - 2^{-m}$. 
Randomized 2-SAT Analysis

Fix a satisfying assignment $S$. Let $X_i \leq n$ be the number of variables that are assigned the same values as in $S$, at step $i$.

- $X_{i+1} = X_i \pm 1$ since we flip one variable in an unsatisfied clause.
Randomized 2-SAT Analysis

Fix a satisfying assignment $S$. Let $X_i \leq n$ be the number of variables that are assigned the same values as in $S$, at step $i$.

\[
\begin{array}{ccccccccc}
1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
\end{array}
\]

assignment $i$

\[
\begin{array}{ccccccccc}
1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\
\end{array}
\]

- $X_{i+1} = X_i \pm 1$ since we flip one variable in an unsatisfied clause.
- $\Pr(X_{i+1} = X_i + 1) \geq$
- $\Pr(X_{i+1} = X_i - 1) \leq$

\[
(x_1 \lor \overline{x}_2) \land (\overline{x}_1 \lor \overline{x}_3) \land (x_1 \lor x_2) \land (x_4 \lor \overline{x}_3) \land (x_4 \lor \overline{x}_1)
\]
The number of correctly assigned variables at step $i$, $X_i$, obeys

$$\Pr(X_{i+1} = X_i + 1) \geq \frac{1}{2} \quad \text{and} \quad \Pr(X_{i+1} = X_i - 1) \leq \frac{1}{2}.$$
The number of correctly assigned variables at step $i$, $X_i$, obeys

$$\Pr(X_{i+1} = X_i + 1) \geq \frac{1}{2} \quad \text{and} \quad \Pr(X_{i+1} = X_i - 1) \leq \frac{1}{2}.$$ 

Is $X_0, X_1, X_2, \ldots$ a Markov chain?
Coupling to a Markov Chain

The number of correctly assigned variables at step $i$, $X_i$, obeys

\[ \Pr(X_{i+1} = X_i + 1) \geq \frac{1}{2} \quad \text{and} \quad \Pr(X_{i+1} = X_i - 1) \leq \frac{1}{2}. \]

Is $X_0, X_1, X_2, \ldots$ a Markov chain?

Define a Markov chain $Y_0, Y_1, \ldots$ such that $Y_0 = X_0$ and:

\[
\begin{align*}
\Pr(Y_{i+1} = 1|Y_i = 0) &= 1 \\
\Pr(Y_{i+1} = j + 1|Y_i = j) &= 1/2 \quad \text{for } 1 \leq j \leq n - 1 \\
\Pr(Y_{i+1} = j - 1|Y_i = j) &= 1/2 \quad \text{for } 1 \leq j \leq n - 1 \\
\Pr(Y_{i+1} = n|Y_i = n) &= 1.
\end{align*}
\]
Coupling to a Markov Chain

The number of correctly assigned variables at step $i$, $X_i$, obeys

$$\Pr(X_{i+1} = X_i + 1) \geq \frac{1}{2} \quad \text{and} \quad \Pr(X_{i+1} = X_i - 1) \leq \frac{1}{2}. $$

Is $X_0, X_1, X_2, \ldots$ a Markov chain?

Define a Markov chain $Y_0, Y_1, \ldots$ such that $Y_0 = X_0$ and:

$$\Pr(Y_{i+1} = 1|Y_i = 0) = 1$$

$$\Pr(Y_{i+1} = j + 1|Y_i = j) = \frac{1}{2} \text{ for } 1 \leq j \leq n - 1$$

$$\Pr(Y_{i+1} = j - 1|Y_i = j) = \frac{1}{2} \text{ for } 1 \leq j \leq n - 1$$

$$\Pr(Y_{i+1} = n|Y_i = n) = 1.$$

- Our algorithm terminates as soon as $X_i = n$. We expect to reach this point only more slowly with $Y_i$. So it suffices to argue that $Y_i = n$ with high probability for large enough $i$.

- Formally could use a coupling argument (will see later on).
Simple Markov Chain Analysis

Want to bound the expected time required to have $Y_i = n$. 

$$h_j = 0 \quad h_0 = h_1 + 1 \quad h_j = h_{j-1} + \frac{1}{2} + h_{j+1}$$ 

for $1 \leq j \leq n-1$
Want to bound the expected time required to have $Y_i = n$.

Let $h_j$ be the expected number of steps to reach $n$ when starting at node $j$ (i.e., the expected termination time when $j$ variables are assigned correctly.)
Simple Markov Chain Analysis

Want to bound the expected time required to have $Y_i = n$.

Let $h_j$ be the expected number of steps to reach $n$ when starting at node $j$ (i.e., the expected termination time when $j$ variables are assigned correctly.)

\[
\begin{align*}
    h_n &= 0 \\
    h_0 &= h_1 + 1 \\
    h_j &= \frac{h_{j-1}}{2} + \frac{h_{j+1}}{2} + 1 \text{ for } 1 \leq j \leq n - 1
\end{align*}
\]
Claim: $h_j = h_{j+1} + 2j + 1$.
Claim: $h_j = h_{j+1} + 2j + 1$. Can prove via induction on $j$. 
Claim: $h_j = h_{j+1} + 2j + 1$. Can prove via induction on $j$.

- $h_0 = h_1 + 1$, satisfying the claim in the base case.
Claim: $h_j = h_{j+1} + 2j + 1$. Can prove via induction on $j$.

- $h_0 = h_1 + 1$, satisfying the claim in the base case.

\[
\begin{align*}
\frac{h_j}{2} &= \frac{h_{j-1}}{2} + \frac{h_{j+1}}{2} + 1 \\
\frac{h_j}{2} &= \frac{h_j}{2} + (j - 1) + \frac{1}{2} + \frac{h_{j+1}}{2} + 1 \\
\frac{h_j}{2} &= \frac{h_j}{2} + \frac{h_{j+1}}{2} + j + \frac{1}{2}.
\end{align*}
\]
Claim: $h_j = h_{j+1} + 2j + 1$. Can prove via induction on $j$.

- $h_0 = h_1 + 1$, satisfying the claim in the base case.

\[
\begin{align*}
    h_j &= \frac{h_{j-1}}{2} + \frac{h_{j+1}}{2} + 1 \\
    h_j &= \frac{h_j}{2} + (j - 1) + \frac{1}{2} + \frac{h_{j+1}}{2} + 1 \\
    h_j &= \frac{h_j}{2} + \frac{h_{j+1}}{2} + j + \frac{1}{2}.
\end{align*}
\]

- Rearranging gives: $h_j = h_{j+1} + 2j + 1$. 
**Simple Markov Chain Analysis**

**Claim:** $h_j = h_{j+1} + 2j + 1$. Can prove via induction on $j$.

- $h_0 = h_1 + 1$, satisfying the claim in the base case.

\[
h_j = \frac{h_{j-1}}{2} + \frac{h_{j+1}}{2} + 1
\]

\[
h_j = \frac{h_j}{2} + (j - 1) + \frac{1}{2} + \frac{h_{j+1}}{2} + 1
\]

\[
h_j = \frac{h_j}{2} + \frac{h_{j+1}}{2} + j + \frac{1}{2}.
\]

- Rearranging gives: $h_j = h_{j+1} + 2j + 1$.

So in total we have:

\[
h_0 = h_1 + 1 = h_2 + 3 + 1 = \ldots = \sum_{j=0}^{n-1} (2j + 1)
\]
Claim: \( h_j = h_{j+1} + 2j + 1 \). Can prove via induction on \( j \).

- \( h_0 = h_1 + 1 \), satisfying the claim in the base case.

\[
\begin{align*}
    h_j &= \frac{h_{j-1}}{2} + \frac{h_{j+1}}{2} + 1 \\
    h_j &= \frac{h_j}{2} + (j - 1) + \frac{1}{2} + \frac{h_{j+1}}{2} + 1 \\
    h_j &= \frac{h_j}{2} + \frac{h_{j+1}}{2} + j + \frac{1}{2}.
\end{align*}
\]

- Rearranging gives: \( h_j = h_{j+1} + 2j + 1 \).

So in total we have:

\[
h_0 = h_1 + 1 = h_2 + 3 + 1 = \ldots = \sum_{j=0}^{n-1} (2j + 1) = n^2.
\]
Upshot: Consider the Markov chain $Y_0, Y_1, \ldots$, and let $i^*$ be the minimum $i$ such $Y_{i^*} = n$. Then $\mathbb{E}[i^*] \leq n^2$. 
Upshot: Consider the Markov chain \( Y_0, Y_1, \ldots \), and let \( i^* \) be the minimum \( i \) such \( Y_{i^*} = n \). Then \( \mathbb{E}[i^*] \leq n^2 \).

- Thus, by Markov’s inequality, with probability \( \geq 1/2 \), our 2-SAT algorithms finds a satisfying assignment within \( 2n^2 \) steps.
**Upshot:** Consider the Markov chain \( Y_0, Y_1, \ldots \), and let \( i^* \) be the minimum \( i \) such \( Y_{i^*} = n \). Then \( \mathbb{E}[i^*] \leq n^2 \).

- Thus, by Markov’s inequality, with probability \( \geq 1/2 \), our 2-SAT algorithms finds a satisfying assignment within \( 2n^2 \) steps.
- Splitting our \( 2mn^2 \) total steps into \( m \) periods of \( 2n^2 \) steps each, we fail to find a satisfying assignment in all \( m \) periods with probability at most \( 1/2^m \).
More Challenging Problem: Find a satisfying assignment for a 3-CNF formula with $n$ variables.

$$(x_1 \lor \bar{x}_2 \lor \bar{x}_3) \land (\bar{x}_1 \lor \bar{x}_3 \lor x_4) \land (x_1 \lor x_2 \lor \bar{x}_3).$$
More Challenging Problem: Find a satisfying assignment for a 3-CNF formula with $n$ variables.

$$(x_1 \lor \overline{x}_2 \lor \overline{x}_3) \land (\overline{x}_1 \lor \overline{x}_3 \lor x_4) \land (x_1 \lor x_2 \lor \overline{x}_3).$$

- 3-SAT is famously NP-hard. What is the naive deterministic runtime required to solve 3-SAT?
3-SAT

More Challenging Problem: Find a satisfying assignment for a 3-CNF formula with $n$ variables.

$$(x_1 \lor \bar{x}_2 \lor \bar{x}_3) \land (\bar{x}_1 \lor \bar{x}_3 \lor x_4) \land (x_1 \lor x_2 \lor \bar{x}_3).$$

- 3-SAT is famously NP-hard. **What is the naive deterministic runtime required to solve 3-SAT?**
- The current best known runtime is $O(1.307^n)$ [Hansen, Kaplan, Zamir, Zwick, 2019].
- Will see that our simple Markov chain approach gives an $O(1.3334^n)$ time algorithm.
More Challenging Problem: Find a satisfying assignment for a 3-CNF formula with $n$ variables.

$$(x_1 \lor \overline{x}_2 \lor \overline{x}_3) \land (\overline{x}_1 \lor \overline{x}_3 \lor x_4) \land (x_1 \lor x_2 \lor \overline{x}_3).$$

- 3-SAT is famously NP-hard. What is the naive deterministic runtime required to solve 3-SAT?
- The current best known runtime is $O(1.307^n)$ [Hansen, Kaplan, Zamir, Zwick, 2019].
- Will see that our simple Markov chain approach gives an $O(1.3334^n)$ time algorithm.
- Note that the exponential time hypothesis conjectures that $O(c^n)$ is needed to solve 3-SAT for some constant $c > 1$. The strong exponential time hypothesis conjectures that for $k \to \infty$, solving $k$-SAT requires $O(2^n)$ time.
Randomized 3-SAT Algorithm

1. Start with an arbitrary assignment.

2. Repeat $m$ times, terminating if a satisfying assignment is found:
   - Chose an arbitrary unsatisfied clause.
   - Pick one of the variables in the clause uniformly at random, and switch the assignment of the variable.

3. If a valid assignment is not found, return that the formula is unsatisfiable.
Randomized 3-SAT Analysis

As in the 2-SAT setting, let $X_i$ be the number of correctly assigned variables at step $i$. We have:

$$\Pr(X_i = X_{i-1} + 1) \geq$$

$$\Pr(X_i = X_{i-1} - 1) \leq$$
Randomized 3-SAT Analysis

As in the 2-SAT setting, let $X_i$ be the number of correctly assigned variables at step $i$. We have:

\[
\begin{align*}
\Pr(X_i &= X_{i-1} + 1) \geq \\
\Pr(X_i &= X_{i-1} - 1) \leq
\end{align*}
\]

Define the coupled Markov chain $Y_0, Y_1, \ldots$ as before, but with $Y_i = Y_{i-1} + 1$ with probability $1/3$ and $Y_i = Y_{i-1} - 1 = 2/3$. 

[Diagram of a Markov chain with transitions labeled as 1, 1/3, 1/3, 1/3, 1/3, 1/3, 1/3, 1/3, 1/3, 1.]
Randomized 3-SAT Analysis

As in the 2-SAT setting, let $X_i$ be the number of correctly assigned variables at step $i$. We have:

$$\Pr(X_i = X_{i-1} + 1) \geq \Pr(X_i = X_{i-1} - 1) \leq$$

Define the coupled Markov chain $Y_0, Y_1, \ldots$ as before, but with $Y_i = Y_{i-1} + 1$ with probability $1/3$ and $Y_i = Y_{i-1} - 1 = 2/3$.

How many steps do you expect are needed to reach $Y_i = n$?
Letting $h_j$ be the expected number of steps to reach $n$ when starting at node $j$,

\[
\begin{align*}
    h_n &= 0 \\
    h_0 &= h_1 + 1 \\
    h_j &= \frac{2h_{j-1}}{3} + \frac{h_{j+1}}{3} + 1 \text{ for } 1 \leq j \leq n - 1
\end{align*}
\]
Randomized 3-SAT Analysis

Letting $h_j$ be the expected number of steps to reach $n$ when starting at node $j$,

$$h_n = 0$$

$$h_0 = h_1 + 1$$

$$h_j = \frac{2h_{j-1}}{3} + \frac{h_{j+1}}{3} + 1 \text{ for } 1 \leq j \leq n - 1$$

• We can prove via induction that $h_j = h_{j+1} + 2^{j+2} - 3$ and in turn, $h_0 = 2^{n+2} - 4 - 3n$. 
Randomized 3-SAT Analysis

Letting $h_j$ be the expected number of steps to reach $n$ when starting at node $j$,

\[ h_n = 0 \]
\[ h_0 = h_1 + 1 \]
\[ h_j = \frac{2h_{j-1}}{3} + \frac{h_{j+1}}{3} + 1 \text{ for } 1 \leq j \leq n - 1 \]

- We can prove via induction that $h_j = h_{j+1} + 2^{j+2} - 3$ and in turn, $h_0 = 2^{n+2} - 4 - 3n$.
- Thus, in expectation, our algorithm takes at most $\approx 2^{n+2}$ steps to find a satisfying assignment if there is one.
Randomized 3-SAT Analysis

Letting $h_j$ be the expected number of steps to reach $n$ when starting at node $j$,

\[
\begin{align*}
  h_n &= 0 \\
  h_0 &= h_1 + 1 \\
  h_j &= \frac{2h_{j-1}}{3} + \frac{h_{j+1}}{3} + 1 \text{ for } 1 \leq j \leq n - 1
\end{align*}
\]

- We can prove via induction that $h_j = h_{j+1} + 2^{j+2} - 3$ and in turn, $h_0 = 2^{n+2} - 4 - 3n$.
- Thus, in expectation, our algorithm takes at most $\approx 2^{n+2}$ steps to find a satisfying assignment if there is one.
- Is this an interesting result?
Modified 3-SAT Algorithm

**Key Idea:** If we pick our initial assignment uniformly at random, we will have $\mathbb{E}[X_0] = n/2$. With very small, but still non-negligible probability, $X_0$ will be much larger, and our random walk will be more likely to find a satisfying assignment.
Modified 3-SAT Algorithm

**Key Idea:** If we pick our initial assignment uniformly at random, we will have $E[X_0] = n/2$. With very small, but still non-negligible probability, $X_0$ will be much larger, and our random walk will be more likely to find a satisfying assignment.

**Modified Randomized 3-SAT Algorithm:**

Repeat $m$ times, terminating if a satisfying assignment is found:

1. Pick a uniform random assignment for the variables.
2. Repeat $3n$ times, terminating if a satisfying assignment is found:
   - Chose an arbitrary unsatisfied clause.
   - Pick one of the variables in the clause uniformly at random, and switch the assignment of the variable.

If a valid assignment is not found, return that the formula is unsatisfiable.
Consider a single random assignment with $X_0 = n - j$. I.e., we need to correct $j$ variables to find a satisfying assignment.
Consider a single random assignment with $X_0 = n - j$. I.e., we need to correct $j$ variables to find a satisfying assignment.

Let $q_j$ be a lower bound on the success probability in this case. Since $j \leq n$ and since we run the search process for $3n$ steps,

$$q_j = \Pr[X_{3n} = n] \geq \Pr[X_{3j} = n]$$
Consider a single random assignment with $X_0 = n - j$. I.e., we need to correct $j$ variables to find a satisfying assignment.

Let $q_j$ be a lower bound on the success probability in this case. Since $j \leq n$ and since we run the search process for $3n$ steps,

$$q_j = \Pr[X_{3n} = n] \geq \Pr[X_{3j} = n] \geq \Pr[\text{take exactly } 2j \text{ steps forward and } j \text{ steps back in } 3j \text{ steps}]$$
Consider a single random assignment with $X_0 = n - j$. I.e., we need to correct $j$ variables to find a satisfying assignment.

Let $q_j$ be a lower bound on the success probability in this case. Since $j \leq n$ and since we run the search process for $3n$ steps,

$$q_j = \Pr[X_{3n} = n] \geq \Pr[X_{3j} = n] \geq \Pr[\text{take exactly } 2j \text{ steps forward and } j \text{ steps back in } 3j \text{ steps}]$$

$$= \binom{3j}{j} \left(\frac{2}{3}\right)^j \cdot \left(\frac{1}{3}\right)^{2j}.$$
Consider a single random assignment with $X_0 = n - j$. I.e., we need to correct $j$ variables to find a satisfying assignment.

Let $q_j$ be a lower bound on the success probability in this case. Since $j \leq n$ and since we run the search process for $3n$ steps,

$$q_j = \Pr[X_{3n} = n] \geq \Pr[X_{3j} = n] \geq \Pr[\text{take exactly } 2j \text{ steps forward and } j \text{ steps back in } 3j \text{ steps}] = \left(\begin{array}{c}3j \\ j \end{array}\right) \left(\frac{2}{3}\right)^j \left(\frac{1}{3}\right)^{2j}.$$  

Via Stirling’s approximation, $\left(\begin{array}{c}3j \\ j \end{array}\right) \geq \frac{1}{\sqrt{j}} \cdot \frac{3^{3j-2}}{2^{2j-2}}$, giving:

$$q_j \geq \frac{2^j}{\sqrt{j} \cdot 2^{2j} \cdot 3^{3j}} \approx \frac{1}{\sqrt{j} \cdot 2^j} \geq \frac{1}{\sqrt{n} \cdot 2^j}.$$
Modified 3-SAT Analysis

Our overall probability of success in a single trial is then lower bounded by:

$$q \geq \sum_{j=0}^{n} \text{Pr}[X_0 = n - j] \cdot q_j$$
Our overall probability of success in a single trial is then lower bounded by:

\[
q \geq \sum_{j=0}^{n} \Pr[X_0 = n - j] \cdot q_j \\
\geq \sum_{j=0}^{n} \binom{n}{j} \cdot \frac{1}{2^n} \cdot \frac{1}{\sqrt{n} \cdot 2^j}
\]

Thus, if we repeat for \(m = \mathcal{O}\left(\sqrt{n} \cdot \left(\frac{4}{3}\right)^n\right) = \mathcal{O}(1.333 n)\) trials, with very high probability, we will find a satisfying assignment if there is one.
Modified 3-SAT Analysis

Our overall probability of success in a single trial is then lower bounded by:

\[ q \geq \sum_{j=0}^{n} \Pr[X_0 = n - j] \cdot q_j \]

\[ \geq \sum_{j=0}^{n} \binom{n}{j} \cdot \frac{1}{2^n} \cdot \frac{1}{\sqrt{n} \cdot 2^j} \]

\[ \geq \frac{1}{\sqrt{n} \cdot 2^n} \sum_{j=0}^{n} \binom{n}{j} \cdot \frac{1}{2^j} \]
Modified 3-SAT Analysis

Our overall probability of success in a single trial is then lower bounded by:

\[
q \geq \sum_{j=0}^{n} \Pr[X_0 = n - j] \cdot q_j \\
\geq \sum_{j=0}^{n} \binom{n}{j} \cdot \frac{1}{2^n} \cdot \frac{1}{\sqrt{n} \cdot 2^j} \\
\geq \frac{1}{\sqrt{n} \cdot 2^n} \sum_{j=0}^{n} \binom{n}{j} \cdot \frac{1}{2^j} \\
= \frac{1}{\sqrt{n} \cdot 2^n} \cdot \left( \frac{3}{2} \right)^n
\]
Our overall probability of success in a single trial is then lower bounded by:

\[
q \geq \sum_{j=0}^{n} \Pr[X_0 = n - j] \cdot q_j \\
\geq \sum_{j=0}^{n} \binom{n}{j} \cdot \frac{1}{2^n} \cdot \frac{1}{\sqrt{n} \cdot 2^j} \\
\geq \frac{1}{\sqrt{n} \cdot 2^n} \sum_{j=0}^{n} \binom{n}{j} \cdot \frac{1}{2^j} \\
= \frac{1}{\sqrt{n} \cdot 2^n} \cdot \left(\frac{3}{2}\right)^n = \frac{1}{\sqrt{n}} \cdot \left(\frac{3}{4}\right)^n.
\]
Our overall probability of success in a single trial is then lower bounded by:

\[ q \geq \sum_{j=0}^{n} \Pr[X_0 = n-j] \cdot q_j \]

\[ \geq \sum_{j=0}^{n} \binom{n}{j} \cdot \frac{1}{2^n} \cdot \frac{1}{\sqrt{n} \cdot 2^j} \]

\[ \geq \frac{1}{\sqrt{n} \cdot 2^n} \sum_{j=0}^{n} \binom{n}{j} \cdot \frac{1}{2^j} \]

\[ = \frac{1}{\sqrt{n} \cdot 2^n} \cdot \left( \frac{3}{2} \right)^n = \frac{1}{\sqrt{n}} \cdot \left( \frac{3}{4} \right)^n. \]

Thus, if we repeat for \( m = O \left( \sqrt{n} \cdot \left( \frac{4}{3} \right)^n \right) = O(1.33334^n) \) trials, with very high probability, we will find a satisfying assignment if there is one.