1 K-Hamiltonian Path

Question: Randomized algorithm for finding a Hamiltonian path of length $k$ in a given graph $G$.

1. Randomly $k$-color the graph.

2. Run deterministic algorithm to find the shortest path that visits $k$ distinct colors. Using dynamic programming. [$\text{STATE} =$ where you end up & Which colors have seen so far.] Can be done in $n^2 \cdot 2^k$ time, for $n$ steps and $2^k$ states.

3. Repeat $\log(\frac{1}{\delta})e^k$ times.

Analysis: We only care about coloring true path/set of $k$ nodes. The chance of having a correct Hamiltonian path of length $k$ (correct coloring) is

$$\frac{\# \text{ of valid coloring of the set}}{\# \text{ of total coloring}} = \frac{k!}{k^k} \approx \frac{1}{e^k}$$

If we repeat $\log(\frac{1}{\delta})e^k$ times, we’ll get the correct result with high probability $(1 - \delta)$ The total time taken is:

$$O(n^2 \cdot 2^{O(k)})$$

2 Sampling

Question: There is some $S \subseteq SPACE(U)$. We have an oracle to query if $x \in S$ for $\forall x$. Goal: estimate $Vol(S)$.

Simple algorithm: Pick $x_1, x_2, \ldots, x_m \in U$ uniformly, and query if $x_i \in S$. Let $Z_i$ be the indicator event whether $x_i \in S$. Then,

$$\frac{\# \text{ lie in } S}{\# \text{ picked}} \approx \frac{Vol(S)}{Vol(U)} = p$$

There are many factors that $p$ can depend upon. For example, it’ll depend on how large $S$ and $U$ are. One could imagine the above process of sampling and estimating in 2-dimension. In high dimensional space, it’ll look as estimating the volume of some $d$-dimensional polytope.
**Question**  How many samples are needed to learn $p$ with estimator satisfying $\tilde{p} \in (1 \pm \epsilon)p$ with probability $1 - \delta$. That is, an $(\epsilon, \delta)$ approximation.

One could recollect that we’re dealing with a similar event we have studied before - of tossing a coin and estimating probability of getting a head. Just for the sake of completeness, we’ll derive the result here again. Let’s assume we sample $n$ points $x_1, x_2, \cdots, x_n$. Then, we know that the expected number of points lying in $S$ will be $np$. That is,

$$\mathbb{E}\left[\sum_{i=1}^{n} Z_i\right] = p$$

Then,

$$\mathbb{P}\left[\left|\frac{1}{n}\sum_{i=1}^{n} Z_i - p\right| > \epsilon p\right] = \mathbb{P}\left[\left|\sum_{i=1}^{n} Z_i - np\right| > np\epsilon\right] \leq 2e^{-\epsilon^2 np}$$

Thus, in order for this probability to be less than $\delta$, we get $n \geq \frac{3}{p\epsilon^2} \log\left(\frac{2}{\delta}\right)$.

One might be tempted to sample $\hat{n}$ elements such that $\sum_{i=1}^{\hat{n}} Z_i = \frac{3}{p\epsilon^2} \log\left(\frac{2}{\delta}\right)$, and estimate the probability $p$ as $\hat{p} = \frac{\sum_{i=1}^{\hat{n}} Z_i}{n}$. But we can’t be sure that this is indeed a correct estimation. Consider the following picture.

Where $\hat{\mu} = \frac{3}{p\epsilon^2} \log\left(\frac{2}{\delta}\right)$, and the red line represents the actual $\sum Z_i$ vs $n$ curve. For $\sum Z_i = \hat{\mu}$, the actual $n$ value is $n = \hat{\mu}/p$, whereas we get the number of samples where $\sum Z_i = \hat{\mu}$ as $\hat{n}$. Hence, we estimate

$$\hat{p} = \frac{\hat{\mu}}{n}$$

$$\implies \hat{n} = \frac{\hat{\mu}}{p}$$

$$\implies \hat{n} \in \frac{\hat{\mu}}{p} \left[1 + \epsilon, \frac{1}{1 + \epsilon}\right]$$
The previous result tells us about the accuracy of $\sum Z_i$, that is, the value of $\sum Z_i$ will be within $(1 \pm \epsilon)$ actual mean $\hat{\mu}$ (w.h.p.). What we moreover need is that the number of samples $\hat{n}$ is within the range as specified above. We’ll prove that it’s indeed in this range with high probability.

Consider the number of samples $n' = \frac{\hat{\mu}}{p(1-\epsilon)}$. For this, the actual mean is $\mu = \frac{\hat{\mu}}{1-\epsilon}$. We need to show that $\Pr \left[ \sum_{i=1}^{n'} Z_i < \frac{\hat{\mu}}{1-\epsilon} (1-\epsilon) \right] \leq e^{-\frac{\epsilon^2}{8} n' p} \leq \delta$.

Thus, with very high probability $\hat{n}$ will be less than $n' = \frac{\hat{\mu}}{p(1-\epsilon)}$. Similarly, we can prove for $\frac{\hat{\mu}}{p(1+\epsilon)}$ and hence with high probability $\hat{n} \in \frac{\hat{\mu}}{p(1-\epsilon)} \left[ \frac{1}{1+k}, 1-\epsilon \right]$. Thus, we guarantee that sampling $\hat{n}$ elements such that $\sum_{i=1}^{\hat{n}} Z_i = \frac{3}{4} \log \left( \frac{2}{\epsilon} \right)$ elements gives a probability estimation $\hat{p} = \frac{\sum_{i=1}^{\hat{n}} Z_i}{\hat{n}}$ satisfying $\hat{p} \in p(1 \pm \epsilon)$ with high probability.

3 Median Finding

Question: Given $x_1, ..., x_n$, find the median $x_i$.

1. Sort & output median $\rightarrow O(n \log n)$.
2. Quick select, modified quick sort $T(n) = O(n) + T(\frac{3n}{4}) \rightarrow O(n)$ time and # of comparisons in expectation. Still has $\left( \frac{1}{k} \right)^k$ chance of $\Theta(nk)$ work.
3. Fancy deterministic algorithm: split $\frac{n}{5}$ sets of 5 elements each, apply the same divide and conquer method. Take the median of medians.

$T(n) = O(n) + T(\frac{n}{5}) + T(\frac{7n}{10}) \rightarrow O(n)$

Randomized Algorithm in $O(n)$ w.h.p.:

Sample $y_1, y_2, ..., y_s$ from $X[y_i = x_j$ for $j \in [n]$ uniformly at random]. Sort in $O(s \log s)$. We want to say

$y_{\frac{s}{2} - k} \leq \text{median } x \leq y_{\frac{s}{2} + k}$

w.h.p for $k = O(\sqrt{S \log n})$.

$Pr[y_{\frac{s}{2} - k} > \text{median } x] = Pr[\text{at least } \frac{s}{2} + k \text{ elements choices of } y \leq \text{median } x]$

Using indicator $Z_i = (y_i \leq \text{median } X)$

$Pr[Z_i] = \frac{1}{2}$

$E[\sum Z_i] = \frac{s}{2}$
$$\Pr[\sum Z_i \geq \frac{s}{2} + k] \leq e^{-\frac{2k^2}{s}}$$

Using the value $k = O(\sqrt{S \log n})$, we get the above probability being very low.

**Question:** How do we use this algorithm?

**Option 1:** Use $y_s^2$ for quick select. Rank of $y_s^2$ is $rac{n}{2} \pm O(n \sqrt{\log n})$ w.h.p.

**Option 2:** Scan through $x$, and put them in one of the following groups: $(x < y_L)$, $(x \in [y_L, y_H])$, or $(x > y_H)$ for $(L, H) = (\frac{s}{2} - k, \frac{s}{2} + k)$. Sort $x \in [y_L, y_H]$ and output the $(\frac{n}{2} - |x < y_L|)^{th}$ element.

$$\# \text{ of comparisons } \leq O(s \log s) \xleftarrow{\text{sort } y} + \leq 2n \xleftarrow{\text{partition}} + O(m \log m)$$

where $m = |x|x \in [y_L, y_H]|$. Notice that the $2n$ term is actually $1.5n$ as for almost half of the elements, we only compare with $y_L$.

Consider the following equation, which holds true for any fraction $f \in [0, 1]$

$$y_{f \cdot s - k} \leq x_{(f \cdot n)} \leq y_{f \cdot s + k} \quad \forall \text{fractions } f$$

What we want are the number of such $x$ such that

$$(x|x \in [y_s^{\frac{s}{2} - k}, y_s^{\frac{s}{2} + k}])$$

This only happens for $x_{fn}$ with $f \cdot s \geq \frac{1}{2}s - 2k$

$$\implies f = \frac{1}{2} - \frac{2k}{s} \text{ to } \frac{1}{2} + \frac{2k}{s}$$

Therefore, $(m \log(m)) \leq \frac{4k}{s} \cdot n = O(4n \sqrt{\frac{\log n}{s}}) = O(n \sqrt{\frac{\log n}{s}})$

Pick $\log n << s << \frac{n}{\log n}$, $\implies \# \text{ of comparisons is } 1.5n + O(n)$

$s = n^{\frac{2}{3}} \implies 1.5n + O(n^{\frac{2}{3}} \log n)$