Exercises for Randomised Algorithms

tms41@cam.ac.uk

last updated: 7.3.2022, 13:56

Lecture 1 (Introduction, Randomised Max-Cut and Coupon Collecting)

The question highlighted are those discussed in the Example Classes.

1. We consider the coupon collecting problem with n coupons.
   a) Prove that it takes $n\sum_{k=1}^{n}\frac{1}{k}$ days on expectation to collect all coupons.
   b) Deduce that the probability it takes more than $n\log n + cn$ days is at most e^{-c}.

2. (a bit difficult.) Consider the following, continuous-time version of the Coupon Collecting Problem. We are collecting coupons in parallel so that the waiting time of each coupon is an independent exponential random variable with parameter $1/p_i$ (and expectation p_i). Further, assume $\sum_{i=1}^{n} p_i = 1$. Hence it is possible to get several coupons on the same day but it is also possible to get no coupon).
   a) What is the expected time until all n coupons have been seen?
   b) Which answer do you get if $p_1 = p_2 = \cdots = p_n = 1/n$?

   Hint: For a continuous, non-negative random variable Y, it holds that:

   $$\mathbb{E}\left[\,Y\,\right] := \int_{t=0}^{\infty} \mathbb{P}\left[\,Y > t\,\right] dt.$$

3. Can you find a deterministic polynomial-time algorithm for the MAX-CUT problem with approximation ratio 1/2? [We will get back to this question a bit later when learning how to derandomise algorithms.]

4. Consider the randomised algorithm for MAX-CUT. Using Markov's inequality, prove a lower bound on the probability that the solution returns a cut with at least $|E|/4$ edges.

5. Consider the randomised algorithm for MAX-CUT. Using Chebyshev's inequality, prove that the algorithm returns a cut with at least $|E|/2 - \sqrt{C \cdot |E|}$ edges with probability at least $1 - 1/C$.
   Hint: Before applying Chebyshev's inequality, analyse and upper bound the second moment $\mathbb{E}\left[\,e(S, S^c)^2\,\right]$.

6. A matching in a graph is a set of edges without common vertices. In the Maximum Bipartite Matching problem, we are given a bipartite graph $G(L \cup R, E)$ (without multiple edges), and we want to find a matching of maximum cardinality. Consider the following simple randomised algorithm for this problem: Each edge is selected independently with probability p. All edges that have common endpoints are discarded. To keep the analysis easier, let us assume that the bipartite graph has $|L| = |R| = n$ and that every vertex has degree 3.
   (a) What is the expected cardinality of the matching returned by the algorithm as a function of p?
   (b) Find the value of p that maximises the expected cardinality of the matching. What is the expected cardinality of the matching in this case?
   (c) Assume now the graph is regular of degree $d \geqslant 3$, not necessarily constant. Would you choose a constant value of p or a value that depends on d and/or n? Explain your choice.

Lecture 2-3 (Concentration and Chernoff Bounds)

1. Compare the Central Limit Theorem to Chernoff bounds. What are the advantages and disadvantages of Chernoff bounds?

2. Let $X_1, X_2, \ldots, X_n$ be n independent geometric random variables, each with parameter p (so $\mathbb{E}[X_i] = 1/p$ for each $i = 1, 2, \ldots, n$). Derive a Chernoff bound for $X := \sum_{i=1}^{n} X_i$.

3. (This is from the textbook [Mitzenmacher & Upfal]). The following extension of the Chernoff bound is often implicitly assumed to be true. Here we will prove this formally. Let $X_1, X_2, \ldots, X_n$ be n independent Bernoulli random variables. Let $X := \sum_{i=1}^{n} X_i$ and $\mu = \mathbb{E}[X]$. Choose any μ_L and μ_H such that $\mu_L \leqslant \mu \leqslant \mu_H$. Then, for any $\delta > 0$,

$$\mathbb{P}[X \geqslant (1+\delta)\mu_H] \leqslant \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu_H}.$$

   Similarly, for any $\delta > 0$,

$$\mathbb{P}[X \leqslant (1-\delta)\mu_L] \leqslant \left(\frac{e^{\delta}}{(1-\delta)^{1-\delta}}\right)^{\mu_L}.$$

4. Consider again the randomised quick sort algorithm that selects a pivot uniformly at random. Derive a formula for the expected number of comparisons.

5. Using the concentration result for quick sort in class, prove that it implies a bound of $O(n \log n)$ for the expected number of comparisons.

6. We form a random graph G on $n = 4k$ vertices by starting with 4 even sized vertex classes $V_0, \ldots, V_3$ and placing an edge between $x \in V_i$ and $y \in V_j$ with probability $\frac{|i-j|}{4}$.
   (a) Show that $\mathbb{E}[|E(G)|] = 5n^2/32$

(b) Using the Chernoff bound for $X := \sum_{i=1}^{n} X_i$, where the X_i are independent Bernoulli random variables

$$\mathbb{P}[\, X \leqslant (1-\delta)\mu \,] \leqslant \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}} \right)^{\mu},$$

give an estimate on how large must n be to ensure that it has at least $n^2/8$ edges with probability at least $1/2$?

7. Design a randomised algorithm for the following problem. The input consists of an $n \times n$ matrix A with entries in $\{0,1\}$ and a vector x of length n with entries in the real interval $[0,1]$. The goal is to return a vector y of length n with entries in $\{0,1\}$ such that

$$\max_{i=1,\ldots,n} |(Ax)_i - (Ay)_i| \leqslant 2\sqrt{n \log n}$$

with probability at least $1 - n^{-2}$.
Hint: Your algorithm should have the property that for any $1 \leqslant i, j \leqslant n$, $\mathbb{E}[\, A_{i,j} \cdot y_j \,] = A_{i,j} x_j$.

8. Consider an undirected, regular graph with degree d, i.e., every vertex has exactly d neighbours. Apply the Method of Bounded Differences in order to prove concentration for the randomised MAX-CUT algorithm. What is the problem of applying it to an arbitrary graph?

9. In this exercise, you will analyse the balls-into-bins problem for the case $m > 2n \log n$.
   (a) Let X_i^m be the load of bin $i \in [n]$ after m balls and $X = \max_{i \in [n]} X_i^m$. Prove that for $\alpha > 0$, $\mathbb{E}\left[e^{\alpha X_i^m} \right] \leqslant e^{\frac{m}{n} \cdot (e^{\alpha} - 1)}$.
   (b) Consider $0 < \alpha < 1$. Using that $1 < 1 + \alpha + \alpha^2$ and Markov's inequality, show that

$$\mathbb{P}\left[X_i^m < \frac{m}{n} + \frac{2 \log n}{\alpha} + \frac{m}{n} \cdot \alpha \right] > 1 - n^{-2}.$$

   (c) By a suitable choice of α, deduce that w.h.p. $X < \frac{m}{n} + 8 \cdot \sqrt{\frac{m}{n} \log n}$.

10. Let X be a Poisson random variable of mean μ. Prove that

$$\mathbb{E}\left[e^{\lambda X} \right] = e^{\mu(e^{\lambda} - 1)}$$

and deduce that for any $\delta > 0$,

$$\mathbb{P}[\, X \geqslant (1+\delta)\mu \,] \leqslant \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}} \right)^{\mu}.$$

Lecture 4-5 (Markov Chains)

1. Consider the riffle operation. Given two decks of cards A and B with a and b cards, at each step, the next card is chosen from A with probability $\frac{a}{a+b}$ and otherwise from B. Prove that when starting with n cards in total, drawing n cards using the above operation results into a uniform distribution over all permutations such that the subsequences of cards in A (and in B, respectively) are ordered increasingly.

2. Prove that a random walk on a graph is periodic if the graph G is bipartite. Extension: Can you also prove that the random walk is aperiodic if G is not bipartite?

3. Let X_n be the sum of n independent rolls of a fair die. Show that, for any $k \geqslant 2$,

$$\lim_{n \to \infty} \mathbb{P}\left[\, X_n \text{ is divisible by } k \,\right] = \frac{1}{k}.$$

4. When the Uni-bus arrives outside the Computer Lab, the next bus arrives in $1, 2, \ldots, 20$ minutes with equal probability. You arrive at the bus stop without checking the schedule, at some fixed time n.

   (a) How could you model X_n, the number of minutes until the next bus when you arrive at time n, as a Markov chain?

   (b) Buses have been coming and going all day so we can assume the chain has mixed when you arrive. What is the probability of waiting i minutes for a bus in relation to the Chain?

   (c) How long, on average, do you wait until the next bus arrives?

5. (See Slide 17, Lecture 4) Prove the following inequality for finite Markov chains: For any initial distribution μ over Ω,

$$\|P_\mu^t - \pi\|_{tv} \leqslant \max_{x \in \Omega} \|P_x^t - \pi\|_{tv}.$$

6. Let P be a transition matrix of a Markov chain with state space Ω and μ and ν be two probability distributions on Ω. Prove that

$$\left\|\mu P - \nu P\right\|_{tv} \leqslant \left\|\mu - \nu\right\|_{tv}.$$

   What does this imply for the total variation distance of a Markov chain from its stationary distribution π?

7. What is the stationary distribution of the Ehrenfest Chain? Does the Ehrenfest Chain converge to the stationary distribution?

8. Consider the Ehrenfest Markov Chain with state space $\Omega = [d] = \{0, 1, \ldots, d\}$, and assume that the chain starts from state $\{0\}$. Can you express the Variation Distance of the Markov chain via a random walk on the d-dimensional hypercube? Hint: Use some symmetry argument.

9. Consider the argument for analysing the mixing time of a random walk on a hypercube. In order to formally prove an upper bound on the mixing time, first upper bound the probability that after $t := O(d^2)$ steps all coupons are collected. Then use this to upper bound the variation distance after t steps.

10. Most Markov chains covered in this course never reach a stationary distribution exactly, but only get arbitrarily close. Can you find an irreducible Markov chain with n states such that for any starting state x there is an integer t such that $P_x^t = \pi$?

11. Verify that $\pi(u) = \frac{\deg(u)}{2|E|}$ is a stationary distribution of a simple random walk on a graph G. Also show that this holds for a lazy random walk. Which properties of the graph G do you need?

12. Prove the so-called "Essential Edge Lemma", that is, for any undirected graph $G = (V, E)$ the hitting time satisfies $h(u, v) \leqslant 2|E|$ for any $\{u, v\} \in E(G)$. (Note that this is slightly weaker than $h(u, v) + h(v, u) \leqslant 2|E|$, which was used in Lecture 5).

13. Analyse the cover time of a simple random walk on the complete graph (clique), i.e., the graph where each pair of vertices is connected by an undirected edge.

14. Consider a path P_n with vertex set $\{0, 1, \ldots, n\}$ for even n. Can you determine the cover time? Bonus-Question (a bit hard): What is the cover time of a cycle C_n?

    Hint: What is the worst-case start vertex which maximises the time until all vertices are visited?

15. Prove rigorously the claim made in lecture that the expected time for RAND 2-SAT to find a given solution is at most the hitting time $h(0, n)$ of the random walk on a path.

16. (a bit tricky.) For any regular graph $G = (V, E)$, derive an upper bound on the cover time based on the mixing time $t := t_{\text{mix}}(1/n)$ which is $O(n \log n \cdot t)$.

Lecture 6-7 (Linear Programming)

1. [CLRS: 29.1-5] Convert the following LP into slack form. Also state the set of basic and non-basic variables.

$$
\begin{array}{lrcrcrcr}
\text{maximise} & 2x_1 & & & - & 6x_3 & & \\
\text{subject to} & & & & & & & \\
& x_1 & + & x_2 & - & x_3 & \leqslant & 7 \\
& 3x_1 & - & x_2 & & & \geqslant & 8 \\
& -x_1 & + & 2x_2 & + & 2x_3 & \geqslant & 0 \\
& & & & x_1, x_2, x_3 & & \geqslant & 0
\end{array}
$$

2. [CLRS: 29.1-6] Show that the following LP is infeasible:

$$
\begin{array}{lrcrcr}
\text{maximise} & 3x_1 & - & 2x_2 & & \\
\text{subject to} & & & & & \\
& x_1 & + & x_2 & \leqslant & 2 \\
& -2x_1 & - & 2x_2 & \leqslant & -10 \\
& & x_1, x_2 & & \geqslant & 0
\end{array}
$$

3. [CLRS: 29.1-7] Show that the following LP is unbounded:

$$\begin{array}{lrcrcr}
\text{maximise} & x_1 & - & x_2 & & \\
\text{subject to} & & & & & \\
& -2x_1 & + & x_2 & \leqslant & -1 \\
& -x_1 & - & 2x_2 & \leqslant & -2 \\
& & x_1, x_2 & & \geqslant & 0
\end{array}$$

4. Consider the linear program for the shortest path problem from s to t.

   a) What happens if there is a negative-weight cycle?

   b) Prove that, if there are no negative-weight cycles, the optimal solution $\overline{d}_t$ of the linear program equals the correct distance d_t.

   c) How would your formulate the single-source shortest path problem as a linear program?

5. Prove that the set of feasible solutions of a linear program forms a convex set. Recall a set S is convex if for every $x, y \in S$, $\lambda x + (1-\lambda)y \in S$ for all $\lambda \in [0,1]$.

6. Find a linear program which has more than one optimal solution.

7. [CLRS: 29.1-8] Suppose we have general linear program (not necessarily in standard or slack form) with n variables and m constraints, and suppose we convert it into standard form. Given an upper bound on the number of variables and constraints in the resulting linear program. (By constraint we do not count non-negativity constraints)

8. [CLRS: 29.1-9] Give an example of a linear program for which the feasible region is not bounded, but the optimal objective value is finite.

9. [CLRS: 29.3-6] Solve the following linear program using Simplex:

$$\begin{array}{lrcrcr}
\text{maximise} & 5x_1 & - & 3x_2 & & \\
\text{subject to} & & & & & \\
& x_1 & - & x_2 & \leqslant & 1 \\
& 2x_1 & + & x_2 & \leqslant & 2 \\
& & x_1, x_2 & & \geqslant & 0
\end{array}$$

10. [CLRS: 29.5-5] Solve the following linear program using Simplex:

$$\begin{array}{lrcrcr}
\text{maximise} & x_1 & + & 3x_2 & & \\
\text{subject to} & & & & & \\
& x_1 & - & x_2 & \leqslant & 1 \\
& 2x_1 & + & x_2 & \leqslant & 2 \\
& & x_1, x_2 & & \geqslant & 0
\end{array}$$

CLRS3 Show that when the main loop of SIMPLEX is run by INITIALIZE-SIMPLEX, it can never return "unbounded".

11. Consider the SIMPLEX algorithm with the pivot-rule of always picking as entering variable the one with the largest coefficient in the objective function:

$$\begin{array}{rcrcrcrcrcr} z & = & 3 & + & \frac{3}{4}x_1 & - & 20x_2 & + & \frac{1}{2}x_3 & - & 6x_4 \\ x_5 & = & 0 & - & \frac{1}{4}x_1 & + & 8x_2 & + & x_3 & - & 9x_4 \\ x_6 & = & 0 & - & \frac{1}{2}x_1 & + & 12x_2 & + & \frac{1}{2}x_3 & - & 3x_4 \\ x_7 & = & 1 & & & & & - & x_3 & & \end{array}$$

Prove that on this instance, the SIMPLEX algorithms runs into a cycle and does not terminate.

Note: Verifying this by hand is a bit tedious, so you may also consider to write a small program or use other tools.

Lecture 9-10 (Randomised Approximation Algorithms)

1. Consider a CNF formula where each clause has at least 4 literals. Design a randomised approximation algorithm and analyse its approximation ratio.

2. Apply the derandomisation trick based on conditional expectation for the randomised 2-approximation algorithm for MAX-CUT. Can you interpret the resulting algorithm?

3. Apply the derandomisation approach based conditional expectation to the MAX-CNF problem.

4. Prove that the greedy algorithm for the Unweighted Vertex Cover problem achieves an approximation ratio of 2. Bonus-Question: What is the problem behind the "more natural" greedy approach where instead of both endpoints of an uncovered edge, we only include one of the two endpoints into our cover?

5. Consider an instance of the unweighted SET-COVER problem with the condition that no element $x \in X$ appears in more than k many subsets. Design an approximation algorithm based on deterministic rounding which achieves an approximation ratio of at most $O(k)$.

6. Consider the randomised approximation algorithm for the weighted SET-COVER problem. Translate the algorithm from the course into one based on non-linear randomised rounding such that, given the LP solution $\overline{y}$, we directly round this LP solution to get a solution y which covers all elements with probability $1 - 1/n$.

7. Consider a "random" instance of a 3-CNF instance. Given m clauses and n variables, each clause is constructed by choosing 3 variables uniformly at random and without replacement from $\{1, \ldots, n\}$, and then negating each variable with probability $1/2$ independently. Prove that for any $\varepsilon > 0$, if $m \geqslant c \cdot n^{8/7}$, for some constant $c = c(\varepsilon) > 0$, then the formula is not satisfiable with probability $1 - \varepsilon$.

Hint: Make use of the following generalised birthday paradox for any integer $k \geqslant 2$. If m balls are assigned to n bins chosen uniformly at random, then if $m = \Omega(n^{(k-1)/k})$, there will be at least one bin which receives at least k balls with constant probability > 0.

Lecture 11-12 (Spectral Graph Theory and Clustering)

1. Compute the conductance of the complete graph with n vertices.
2. Compute the conductance of the cycle with n vertices.
3. (i) Prove that for every $n \geqslant 2$ there is an unweighted, undirected n-vertex graph with conductance 1.
   (ii) (Open-Ended Bonus Question): Can you characterise all graphs with that property?
4. Consider the transition matrix of a lazy random walks $\widetilde{P} = (P + I)/2$ on a d-regular graph (here I is the $n \times n$ identity matrix and P is the transition matrix of a simple random walk). Prove that all eigenvalues of $\widetilde{P}$ are non-negative.
   Hint: You may use the fact that the eigenvalues of P are between $[0, 1]$ (note that this follows from slide 14, where it is stated that the eigenvalues of L are between $[0, 2]$).
5. Prove that for any d-regular graph with $n \to \infty$ being large, the conductance satisfies $\Phi(G) \leqslant 1/2 + o(1)$. Hint: Use the probabilistic method.
6. Let G be a connected, undirected graph and P be the transition matrix of a simple random walk on G. Show that if -1 is a left eigenvalue of P, then the random walk is periodic.

Lecture 13 (Streaming Algorithms)

1. Consider the following streaming problem. There are n different packets labelled $1, 2, \ldots, n$ to be sent over a channel, and you know the value of n. Assuming that at most one packet will not be sent, design a streaming algorithm using $O(\log n)$ space which is able to return the ID of the lost packet (if there is any).
2. (Slide 9, Second Moment Analysis). If X_n is the counter of Morris' Algorithm after n updates, then prove that for every $n \geqslant 0$,
$$\mathbb{E}\left[2^{2 \cdot X_n} \right] = \frac{3}{2}n^2 + \frac{3}{2}n + 1.$$
3. [Slide 16, AMS]. Prove that by running $\Theta(\log(1/\delta))$ independent copies of the algorithm and returning the median, we obtain a $(O(1), \delta)$ approximation. Why do we return the median and not the mean?
4. Consider a run of Morris' Algorithm which includes n increments, where $n \to \infty$ (see the illustration on slide 11). What is the expected number of "crossings" between the estimate Θ_n and the true count n?

Lecture 14 (Weighted Majority)

1. Consider the online learning model with n experts. Show that for any deterministic algorithm and for any integer T, there exists an input such that $M^{(T)} \geqslant 2 \cdot \min_{i \in [n]} m_i^{(T)}$. Hint: Try to construct an instance with $n = 2$ experts.

2. Consider the deterministic weighted majority algorithm. Prove or disprove that the algorithm always satisfies: $M^{(T)} \geqslant \min_{i \in [n]} m_i^{(T)}$.

3. In the experiments described on slide 11/12, there is an expert which is always wrong in each round. Suggest a general adjustment of the weighted majority algorithm (ideally one which works for both the deterministic and randomised version) that exploits the existence of experts which always (or mostly) predict wrongly.

4. Consider the so-called "follow-the-leader" approach, which means that we are always following the prediction of the best expert so far. Show that for any $n \geqslant 2$ there exists an input such that $M^{(T)} \geqslant (1 - o(1))T$, but for the best expert $i \in [n]$, $m_i^{(T)} \leqslant T/n$.

Selected Hints and Solution Notes

Lecture 1

(4) Apply Markov to $|E| - e(S, S^c)$.

(6) Part a): $p \cdot (1-p)^4 \cdot |E|$. Part b): $p = 1/5$, Part c): $p = \frac{1}{2d-1}$ is the best choice.

Lecture 2-3

(4) Express the total number of comparisons X into a sum of indicator variables $X_{i,j}$ which is 1 if the i-th smallest and j-th smallest element are compared. After some further steps, you should be able to bound the expected number of comparisons from above by $2n \ln n$.

(9) Part c): One possible choice for α might be $\alpha = \sqrt{2 \cdot \frac{n}{m} \log n} \leqslant 1$.

Lecture 4-5

(5) This is very similar to Question 6, which is covered in the example class.

(16) Divide the time into segments of length $t_{\text{mix}}(1/n)$ and then upper bound the probability that a fixed vertex is visited after $O(n \log n t_{\text{mix}}(1/n))$ time.

Lecture 6-8

(9) The optimal solution is $(x_1, x_2) = (1, 0)$ with objective value 5.

(10) The optimal solution is $(x_1, x_2) = (\frac{34}{3}, \frac{10}{3})$ with objective value $\frac{64}{3}$.