Exercises & Model Answers for Randomised Algorithms

tms41@cam.ac.uk

last updated: 12.3.2023, 21:50

Acknowledgements.

I am very grateful to the supervisors of this course for pointing out various errors and suggesting improvements, especially Dimitrios Los, Charlotte Out and Fredrik Rømming.

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

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:

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

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. Prove the following Chernoff Bound (lower tail version; see also slide 6 from Lecture 2). We have $X_1, X_2, \ldots, X_n$ independent Bernoulli random variables with parameter p_i each, $X := \sum_{i=1}^{n} X_i$ and $\mu = \mathbb{E}[\, X \,] = \sum_{i=1}^{n} p_i$. Then, for any $\delta > 0$ it holds that

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

4. (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 $\delta > 0$,

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

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

6. Using the concentration result for quick sort in class (i.e., the result that the number of comparisons is at most $24n \log n$ with probability at least $1 - n^{-1}$), prove that it implies a bound of $O(n \log n)$ for the expected number of comparisons.

7. 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 $1/2$?

8. 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 $1 \leqslant i, j \leqslant n$, $\mathbb{E}[\,A_{i,j} \cdot y_j\,] = A_{i,j} x_j$.

9. 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?

10. We form a random graph $G = (V, E)$ by connecting each pair of the n vertices with probability $1/2$, independently. Prove that, with probability $1 - o(1)$, the randomised algorithm for MAX-CUT returns a solution that is within $1 + o(1)$ of the optimal solution.
    Hint: We need to lower bound the cut returned by the randomised algorithm, and we need to upper bound the optimal cut. For both bounds, first prove a concentration result on $e(U, V \setminus U)$, strong enough so that it holds for all subsets $U \subseteq V$, by exploiting the fact that the graph is randomly generated.

    Answer:

    Consider an arbitrary fixed subset U. Then,
    $$\mathbb{E}[\,e(U, U^c)\,] = |U| \cdot (n - |U|) \cdot (1/2).$$
    but note that here the randomness is over the graph and not the algorithm! By the nicer Chernoff Bound (slide 7) for $X(U) := e(U, U^c) = \sum_{x \in U, y \in U^c} \mathbf{1}_{\{x,y\} \in E(G)}$ (which is a sum of at most $|U| \cdot (n - |U|) \leqslant n^2/4$ random variables!), $\mu(U) = |U| \cdot (n - |U|) \cdot (1/2)$,
    $$\mathbb{P}[\,|X(U) - \mu(U)| \geqslant t\,] \leqslant 2 \cdot e^{-2t^2/(n^2/4)}.$$

We now pick $t = n^{3/2}$ (Remark: This is a specific choice made with some foresight. Other choices are possible, too, and may even lead to a slightly better approximation ratio. However, the error probability must be small enough so that the Union Bound application below over all subsets works.)

$$\mathbb{P}\left[|X(U) - \mu(U)| \geqslant n^{3/2} \right] \leqslant 2 \cdot e^{-2(n^{3/2})^2/(n^2/4)} \leqslant 4^{-n}.$$

By the Union Bound,

$$\mathbb{P}\left[\cup_{U \subseteq V} \left\{ |X(U) - \mu(U)| \geqslant n^{3/2} \right\} \right] \leqslant \sum_{U \subseteq V} \mathbb{P}\left[\left\{ |X(U) - \mu(U)| \geqslant n^{3/2} \right\} \right] \leqslant 2^n \cdot 4^{-n} = 2^{-n}.$$

Therefore, with probability at least $1 - 2^{-n}$, the maximum cut e_{opt} satisfies,

$$\begin{aligned} e_{opt} &\leqslant \max_{U \subseteq V} \left\{ \mu(U) + n^{3/2} \right\} \\ &\leqslant \max_{U \subseteq V} \left\{ |U| \cdot (n - |U|) \cdot (1/2) \right\} + n^{3/2} \\ &\leqslant n^2/8 + n^{3/2}. \end{aligned}$$

Let us now analyse $e_{rand} = e(S, S^c)$, the size of the cut returned by the randomised algorithm, where S is a random set. Note that $|S| = Bin(n, 1/2)$, and using a "nice" Chernoff Bound,

$$\mathbb{P}\left[|S - n/2| \geqslant t \right] \leqslant 2 \cdot e^{-2t^2/n}.$$

We will now choose $t = \sqrt{n \log n}$ (again with some foresight), to get

$$\mathbb{P}\left[|S - n/2| \geqslant \sqrt{n \log n} \right] \leqslant 2n^{-2}.$$

With probability $1 - 2n^{-2}$, the event $|S - n/2| \leqslant \sqrt{n \log n}$ occurs, but also with probability $1 - 2^{-n}$, we have for all possible subsets $U \subseteq V$, that $|X(U) - \mu(U)| \leqslant n^{3/2}$. Hence by the union bound, with probability $1 - 2n^{-2} - 2^{-n} \gg 1 - 3n^{-2}$, we have

$$e(S, S^c) \geqslant \mu(S) - n^{3/2} \geqslant |S| \cdot (n - |S|) \cdot (1/2) - n^{3/2},$$

and now using $|S - n/2| \leqslant \sqrt{n \log n}$,

$$e(S, S^c) \geqslant (n/2 - \sqrt{n \log n}) \cdot (n/2 - \sqrt{n \log n}) \cdot (1/2) - n^{3/2} = n^2/8 - 2n^{3/2}\sqrt{\log n}.$$

In conclusion, with probability at least $1 - 3n^{-2}$, we have

$$\frac{e_{opt}}{e(S, S^c)} \leqslant \frac{n^2/8 + 2n^{3/2}}{n^2/8 - 2n^{3/2}\sqrt{\log n}} \leqslant 1 + o(1).$$

While the above completes the solution, it is also instructive to see a different way of analysing the number of cut edges produced by the randomised algorithm using the Method of Bounded

Independent Differences. To this end, let

$$Z := \sum_{i=1}^{n} \sum_{j=i+1}^{n} \mathbf{1}_{((i \in S \cap v \notin S) \cup (i \notin S, v \notin S)) \cap \{u,v\} \in E(G)}$$

Thus Z is a random function which depends on the n coin flips deciding for each vertex whether it is in S or not, and an additional $\binom{n}{2}$ coin flips deciding for each vertex. Further,

$$\mathbb{E}[\, Z \,] = \binom{n}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} = \frac{n(n-1)}{8}.$$

Changing the appearance of one edge may change Z by at most one. Further, changing for a vertex whether it is in S or not may change Z by at most $n-1$. Hence by the MOBD,

$$\mathbb{P}[\, |Z - \mu| \geqslant t \,] \leqslant 2 \exp\left(- \frac{2t^2}{n \cdot (n-1)^2 + \binom{n}{2} \cdot 1^2} \right).$$

Thus if we choose $t = n^{3/2}\sqrt{\log n}$, we have

$$\mathbb{P}\left[\left| Z - \frac{n(n-1)}{8} \right| \geqslant n^{3/2}\sqrt{\log n} \right] \leqslant 2n^{-1}.$$

11. 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 $e^{\alpha} < 1 + \alpha + \alpha^2$ 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. $X < \frac{m}{n} + 8 \cdot \sqrt{\frac{m}{n} \log n}$.

12. 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 $\delta > 0$,

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

13. Recall the two general (but "not nice" versions) of Chernoff bounds,

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

and

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

a) Prove that for any fixed $\delta \in (0, 1)$, the lower tail bound is smaller than the upper tail bound.

b) Prove that if $X = X_1 + X_2 + \cdots + X_n$, $X_i \sim Ber(p_i)$ satisfies $\mu = p_1 + p_2 + \cdots + p_n = n/2$, then

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

Hint: Exploit that if $X_i \sim Ber(p_i)$, then for $Y_i \sim Ber(1 - p_i)$, $1 - Y_i$ has the same distribution as X_i.

14. Consider the following randomised version of Bubble-Sort on an array of n distinct numbers $A[1], A[2], \ldots, A[n]$. At each step, pick one index $i \in \{1, 2, \ldots, n-1\}$ uniformly at random and compare $A[i]$ with $A[i+1]$. If they are in the wrong order, swap them, otherwise keep them. For an arbitrary array, prove an upper bound on the time until the array is sorted.

Lecture 4-5 (Markov Chains and Random Walks)

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 simple 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 $k \geqslant 2$,

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

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,

$$\|P^t_\mu - \pi\|_{tv} \leqslant \max_{x \in \Omega} \|P^t_x - \pi\|_{tv}.$$

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

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

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

7. Prove that the total variation distance between two distributions over a finite probability space Ω is a value in $[0,1]$.

8. (Difficult) Construct a (family of) irreducible, finite Markov chains with n states so that $\tau(1/2) = O(1)$ but $\tau(1/2-\delta) = \Omega(n \cdot \delta)$ for any $\delta \in (0,1/2)$. (Note this is related to the comment on slide 18 that for the mixing time $\tau_x(\varepsilon)$ we should not take $\varepsilon = 1/2$!). Hint: Consider a simple (i.e., non-lazy) random walk on a graph that consists of two cliques of size $n/2$ connected by a matching. To prove $\tau(1/2-\delta) = \Omega(n \cdot \delta)$, prove an upper bound on the probability that the random walk moves from one clique to another and use this to lower bound the variation distance.

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

10. 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.

11. (Difficult.) Consider the lazy random walk on the d-dimensional hypercube with $n = 2^d$ vertices, and prove that $t_{\mathrm{mix}}(1/4) = O(d \log d)$. Hint: Consider $t = d \log d + cd$ for a suitable constant $c > 0$, and use the result from the coupon collecting process to bound the probability that all d coupons have been obtained after t steps. Then by conditioning on this event, derive an upper bound on $|P^t_{u,v} - \frac{1}{n}|$. Then use this bound to derive an upper bound on $\|P^t_{u,\cdot} - \pi\|_{tv}$.

12. 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 $P^t_x = \pi$?

13. Verify that $\pi(u) = \frac{\deg(u)}{2|E|}$ is a stationary distribution of a simple (i.e., non-lazy) 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?

14. 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 $\{u,v\} \in E(G)$.

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.

Remark: The next three questions involve the cover time of a random walk, which is no longer taught in this course (it is defined as the expected time for a random walk on a graph to visit all vertices, from a worst-case start vertex)

16. 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.

17. 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?

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

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

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

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

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

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

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 you 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 a 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. Solve the following linear program using SIMPLEX:

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

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

12. [CLRS: 29.5-5] Solve the following linear program using SIMPLEX:

$$\begin{array}{lrcrcr} \text{maximise} & x_1 & + & 3x_2 & & \\ \text{subject to} & & & & & \\ & x_1 & - & x_2 & \leqslant & 8 \\ & -x_1 & - & x_2 & \leqslant & -3 \\ & -x_1 & + & 4x_2 & \leqslant & 2 \\ & & x_1, x_2 & & \geqslant & 0 \end{array}$$

Lecture 9-10 (Randomised Approximation Algorithms)

1. Consider a MAX-SAT 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 on conditional expectation to the MAX-CNF problem.

4. Analyse the greedy algorithm for the *Unweighted* Vertex Cover problem that achieves an approximation ratio of 2 (Slide 14 of Lecture 9). 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. (easy). Prove that the LP formulation of the SET-COVER problem (Lecture 10, slide 4) is feasible if and only if the SET-COVER instance has a feasible solution.

6. 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 $O(k)$.

7. Consider a generalisation of the SET-COVER problem, called SET-MULTI-COVER. Here, each element $x \in X$ needs to be covered $r_x \in \mathbb{N}$ many times. Any subset $S \in \mathcal{F}$ can be selected multiple times, but if it is selected k times then the cost is $k \cdot c(S)$. Design an algorithm based on randomised rounding, and prove that that the expected approximation ratio is $O(\log n)$.

8. [Williamson, Shmoys, Exercise 4.1] The following problem arises in telecommunication networks, and is known as the SONET ring loading problem. The network consists of a cycle on n nodes, numbered 0 through $n-1$ clockwise around the cycle. Some set C of calls is given; each call is a pair (i,j) originating at node i and destined to node j. The call can be routed either clockwise or counterclockwise around the ring. The objective is to route the calls so as to minimise the total load on the network. The load L_i on the link $(i,(i+1) \pmod n)$ is the number of calls routed through $(i,(i+1) \pmod n)$, and the total load is $\max_{1 \leqslant i \leqslant n} L_i$.

   Give a 2-approximation algorithm for the SONET ring loading problem.

9. [Williamson, Shmoys, Exercise 4.6] (difficult) Let $G = (A \cup B, E)$ be a bipartite graph. Assume that $|A| \leqslant |B|$ and that we are given non-negative costs $c_{i,j} \geqslant 0$ for each edge $(i,j) \in E$. A complete matching of A is a subset $M \subseteq E$ such that each vertex in A has exactly one edge in M incident on it, and each vertex in B has at most one edge of M incident on it. We wish to find a minimum-cost complete matching.

   a) Express the problem as an integer program.

   b) Consider the linear program relaxation of the integer program and show that given any fractional solution to the linear program relaxation, it is possible in polynomial time to find in polynomial time an integer solution that costs no more than the fractional solution. Then conclude that there is a polynomial-time algorithm for finding a minimum-cost complete matching.
   Hint: Given a set of fractional variables, find a way to modify their values repeatedly such that the solution stays feasible, the overall cost does not increase, and at least one additional fractional value becomes 0 or 1.

10. 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 cover $\mathcal{C}$ which (i) covers all elements with probability $1 - 1/n$, and (ii) has an expected cost which is at most $O(\log n)$ times the cost of the optimal cover. Hint: By non-linear we refer to the way of choosing the probability of setting a variable to 1. The randomised rounding rules in the

lecture is linear in the sense that the probability is equal to the fractional value of the LP solution.

11. Recall the randomised algorithm for SET-COVER presented in the lecture. As input, we have a SET-COVER instance with n elements; and let us additionally assume we have at most poly(n) many subsets (and also that we can cover all n elements, i.e., $\cup_{S \in \mathcal{F}} S = X$). The algorithm achieves with probability $1/3$ that the returned cover is correct and the cost of the cover is at most a factor of $4\ln(n)$ away from the optimal cost (see Lecture 10, slide 9.2). Turn this into a randomised algorithm such that:
    a) The algorithm terminates in a time that is polynomial in the input size, with probability 1 (i.e., always).
    b) The algorithm returns a correct solution, with probability 1 (i.e., always).
    c) The expected approximation ratio is $O(\log n)$.

12. Consider the analysis of the hybrid algorithm for MAX-SAT, specifically Lecture 10, slide 16, which was based on lower bounding α_ℓ and β_ℓ, as well as their average.
    a) Which algorithm would you use if the CNF-formula only contains clauses of length at least 3?
    b) Design an algorithm which works well for a CNF-formula that only contains clauses of length 1 or 3. Hint: Slightly modify the hybrid algorithm from the lecture.

Lecture 11-12 (Spectral Graph Theory and Clustering)

1. Prove that for any undirected, d-regular graph, the matrices $\mathbf{A}$ an $\mathbf{L}$ have the same set of eigenvectors. Also describe the correspondence between the eigenvalues (see Lecture 11, slide 10). Hint: Use $\mathbf{L} = \mathbf{I} - \frac{1}{d}\mathbf{A}$.

2. Compute the conductance of the complete graph with n vertices.

3. Compute the conductance of the cycle with n vertices.

4. (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?

5. 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 the Laplacian Matrix L are between $[0, 2]$).

6. 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 to construct a set S with the required conductance.

7. Show that if G is an undirected, d-regular, connected and bipartite graph, then the largest eigenvalue λ_n of the Laplacian matrix satisfies $\lambda_n = 2$ (this proves one direction of the fourth

statement in the Lemma from Lecture 11/slide 14).

Answer: Let V_1, V_2 be the two bipartite components. Then define a vector f by $f_w := 1$ if $w \in V_1$ and $f_w := -1$ if $w \in V_2$. It then follows that for any vertex $u \in V_1$,

$$[\mathbf{L} \cdot f]_u = \sum_{v \in V} \mathbf{L}_{u,v} f_v = \mathbf{L}_{u,u} f_u + \sum_{v \in V_2} \mathbf{L}_{u,v} f_v = 1 \cdot 1 + \sum_{v \in V_2 : \{u,v\} \in E(G)} -\frac{1}{d} \cdot (-1) = 2 = 2 \cdot f_u,$$

having used that u has exactly d neighbours, and they are all in V_2 as G is bipartite. The computation for $u \in V_2$ is analogous.

One can also extend this to graphs which are not necessarily regular. In this case, we have to switch to the normalised Laplacian matrix:

$$\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2}.$$

In particular, for any $\{u, v\} \in E(G)$, we have

$$\mathbf{L}_{u,v} = \frac{1}{\sqrt{\deg(u) \cdot \deg(v)}}.$$

Now define $f_u := \sqrt{\deg(u)}$ if $u \in V_1$, and $f_u := -\sqrt{\deg(u)}$ if $u \in V_2$. We claim that f is an eigenvector with eigenvalue 2:

$$\begin{aligned}
[\mathbf{L} \cdot f]_u &= \sum_{v \in V} \mathbf{L}_{u,v} f_v \\
&= \mathbf{L}_{u,u} f_u + \sum_{v \in V_2 :\, \{u,v\} \in E(G)} \mathbf{L}_{u,v} \cdot f_v \\
&= 1 \cdot \sqrt{\deg(u)} + \sum_{v \in V_2 :\, \{u,v\} \in E(G)} -\frac{1}{\sqrt{\deg(u) \cdot \deg(v)}} \cdot (-\sqrt{\deg(v)}) \\
&= \sqrt{\deg(u)} + \sqrt{\deg(u)} = 2 \cdot f_u,
\end{aligned}$$

8. (difficult.) 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 (i.e., there is a row vector f such that $fP = -f$), then the random walk is periodic.

9. On Lecture 11/slide 10 (Example 1) we determined the spectrum of the adjacency matrix A for the complete graph (a.k.a. clique) of size 3. Here we would like to generalise this to any complete graph of size $n \geqslant 3$. Prove that the spectrum consists of eigenvalues $n-1$ (with multiplicity 1) and -1 (with multiplicity $n-1$).

10. Consider an undirected, and d-regular graph $G = (V, E)$. Prove that there is a constant $c > 0$ (independent of n and m) such that the diameter of the graph G is bounded by $c \cdot \frac{\log n}{\Phi}$, where Φ is the conductance.

    Hint: Consider a vertex $u \in V$, and consider $B_{\leqslant r}(u)$, the set of vertices with distance at most r to u. Then use the definition of the conductance to prove a lower bound on $|B_{\leqslant r+1}(u)|$ in terms of $|B_{\leqslant r}(u)|$.

    Also you may use without a proof that $\log(1 + z) \geqslant (1/2) \cdot z$ for all $z \in [0, 1]$.

11. (easy) Consider an undirected, and d-regular graph $G = (V, E)$ with $n = 8$ vertices labelled

$1, 2, \ldots, 8$. Assume that the Laplacian Matrix has the second smallest eigenvalue $\lambda_2 = 0.1$ with corresponding eigenvector $(-0.2, -0.1, -0.3, 0.45, 0.45, -0.2, -0.3, 0.1)$. How may cuts are considered by Spectral Clustering, and which ones? Even without knowing the adjacency matrix $\mathbf{A}$, what can you conclude about the conductance of the graph, $\Phi(G)$?

12. (difficult. Also on a high level a bit similar to L2/3 Question 10) Consider the problem of finding the minimum cut in a random graph with n vertices, where each pair of vertices is connected with probability $1/2$ independently. Can you find an algorithm that (i) returns the minimum cut with probability at least $1 - O(n^{-1})$, and (ii) terminates in $O(n^2)$ time (assuming the graph is provided as an adjacency matrix)?
Hint: You may use without proof that: $\sum_{k=1}^{n/2} 1/\binom{n}{k} = O(1/n)$ and $\binom{n}{k} \leqslant \left(\frac{en}{k}\right)^k$ for any $1 \leqslant k \leqslant n/2$.

13. Prove that for any d-regular graph, the largest eigenvalue of the Laplacian $\mathbf{L}$ satisfies $\lambda_n \leqslant 2$.
Answer: Using the Courant-Fisher formula (with $k = n$ and $\mathbf{M} = \mathbf{L}$),

$$\lambda_n = \min_{\substack{x^{(1)},\ldots,x^{(n)} \in \mathbb{R}^n \setminus \{\mathbf{0}\},\\ x^{(i)} \perp x^{(j)}}} \max_{i \in \{1,\ldots,n\}} \frac{{x^{(i)}}^T \mathbf{L} x^{(i)}}{{x^{(i)}}^T x^{(i)}}$$

To prove $\lambda_n \leqslant 2$, it suffices to prove that for any vector $x \in \mathbb{R}^n$,

$$\max_{x \in \mathbb{R}^n} \frac{x^T \mathbf{L} x}{x^T x} \leqslant 2.$$

Using the quadratic form, this is equivalent to,

$$\max_{x \in \mathbb{R}^n} \frac{\sum_{\{u,v\} \in E(G)} \frac{1}{d}(x_u - x_v)^2}{x^T x} \leqslant 2.$$

Further, we have

$$\sum_{\{u,v\} \in E(G)} \frac{1}{d}(x_u - x_v)^2 \leqslant \frac{1}{d} \sum_{\{u,v\} \in E(G)} (2x_u^2 + 2x_v^2) = 2 \cdot \sum_{u \in V} x_u^2,$$

recalling that $\sum_{\{u,v\} \in E(G)}$ includes each edge once. Since $x^T x = \sum_{u \in V} x_u^2$, the claim follows.

Additional Hints and Solution Notes

Lecture 1

(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

(5) 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 $2n \ln n$.

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

Lecture 4-5

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

Lecture 6-7

(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 $\frac{64}{3}$.