Alex Tsun
CSE 312: Foundations of Computing II

PSet #4

With problems from several past UW CSE 312 instructors (Martin Tompa, Anna Karlin, Larry Ruzzo) and Stanford CS 109 instructors (Chris Piech, David Varodayan, Lisa Yan, Mehran Sahami)

Groups: This pset may be done in groups of **up to 2 people**. This means that if you work with a partner, only one person will submit on Gradescope to "PSet 4 [Written]" and add their partner as a collaborator. For this pset, you may *also work on the [Coding] part in your pair*. The pair must be the same pair as the [Written]. Individuals and groups are encouraged to discuss problem-solving strategies with other classmates as well as the course staff, but each group must write up their own solutions.

Instructions: For each problem, remember you must briefly explain/justify how you obtained your answer, as correct answers without an explanation will receive **no credit**. Moreover, in the event of an incorrect answer, we can still try to give you partial credit based on the explanation you provide. It is fine for your answers to include summations, products, factorials, exponentials, or combinations; you dont need to calculate those all out to get a single numeric answer.

Submission: You must upload your written compiled LaTeX PDF to Gradescope under "PSet 4 [Written]" and your code file `mcmc_knapsack.py` to "PSet 4 [Coding]". You must tag your written problems on Gradescope, or you will receive **no credit** as mentioned in the syllabus. Please cite any collaboration at the top of your submission (beyond your group members, which should already be listed).

1. Suppose we run an experiment where we hold CSE 312 office hours for 2 straight days! The number of "customers" on Monday is X, and the number of "customers" on Tuesday is Y, where X and Y are independent. Let $Z = X + Y$ be the total number of customers across both days. We will try to find the (conditional) distribution of the number of students on Monday X, given the total number of students on both days Z.

   (a) Suppose $X \sim Bin(n, p)$ and $Y \sim Bin(m, p)$. What is the conditional PMF $P(X = k|Z = z)$ for integers $0 \leq k \leq z$ and $z \geq 0$? Actually, $(X|Z = z)$ is a parametrized distribution we know. What is its name and what parameter(s) does it have? Explain in at most 1-2 sentences intuitively why this makes sense. (Hint: You know the distribution of Z, and can look up its PMF!)

   (b) For this part, instead suppose $X \sim Poi(\lambda_1)$ and $Y \sim Poi(\lambda_2)$. What is the conditional PMF $P(X = k|Z = z)$ for integers $0 \leq k \leq z$ and $z \geq 0$? Actually, $(X|Z = z)$ is a parametrized distribution we know. What is its name and what parameter(s) does it have? Explain in at most 1-2 sentences intuitively why this makes sense. (Hint: You know the distribution of Z, and can look up its PMF!)

Algorithm 1 Llama Flu Disease Spread Model

```
function NUM_INFECTED(immune_prob)
    is_immune ← Ber(immune_prob).        ▷ 1 with probability immune_prob, 0 otherwise
    if is_immune = 1: then return 0                                  ▷ No one is infected
    spread = 0
    num_to_expose ← Bin(n = 50, p = 0.22)        ▷ Can be any integer in {0, 1, . . . , 50}
    for i = 1, . . . , num_to_expose: do
        spread ← spread + NUM_INFECTED(immune_prob)
    return spread + 1          ▷ Including ourself, since we were infected (not immune)
```

2. Our ability to fight contagious diseases depends on our ability to model them. One person is initially *exposed* to llama-flu. The following recursive function models the total number of individuals who will get *infected* (which could actually be 0 if the initially-exposed person is immune!).

   Compute the expected number of people *infected* when immune_prob = 0.91 (this is just the expected return value when calling this function!). **Give your answer to 4 decimal places.** Hint: Be careful that the number of times we call this recursive function is *random* and not fixed. You may use the result from the last example in 5.3 (slides/notes) about summing a random number of random variables without proof.

3. Recall the kittens and mittens problem from PSet 2: You have 8 pairs of mittens, each a different color. Left and right mittens are distinct. Suppose that you are fostering (possibly imaginary) kittens, and you leave them alone for a few hours with your mittens. When you return, you discover that they have hidden 4 mittens! Suppose that your kittens are equally likely to hide any 4 of your 16 distinct mittens. For $i = 1, \ldots, 8$, let X_i be the indicator/Bernoulli rv of whether or not pair i is complete after this fiasco. Let $Y = \sum_{i=1}^{8} X_i$ be the *total* number of complete pairs of mittens that you have left.

   (a) What is the covariance matrix Σ of the 8-dimensional random vector of indicators $\mathbf{X} = (X_1, \ldots, X_8)$? Recall that Σ is the 8×8 matrix whose entries are $\Sigma_{ij} = Cov(X_i, X_j)$. You don't actually have to create a 8×8 matrix here, just explicitly define what each entry should be! **Give your answers to 4 decimal places**. Hint: Treat the diagonal and off-diagonal elements separately.

   (b) What is $Var(Y)$? **Give your answer to 4 decimal places**. Hint: Covariances should be involved.

4. Use the Central Limit Theorem to approximate the following probabilities, and **state explicitly at which step you invoke it**. Use the continuity correction if and only if it is necessary.

   (a) The Internet fire marshal has declared that if too many people show up to my Zoom office hours, it would constitute a fire hazard. There are 234 students registered in the class. On the day homework is due, each student comes to my office hour with probability 1/6, independently of the other students. What is the approximate probability that between 37 and 41 people (inclusive) come to my office hours? **Give your answer to 4 decimal places.**

(b) A fair 6-sided die is repeatedly rolled until the total sum of all the rolls reaches 180. What is the approximate probability that *at least* 49 rolls are necessary to reach a sum that reaches (or surpasses) 180? **Give your answer to 4 decimal places.** Hint: Try to come up with an equivalent statement where the number of die rolls is fixed and not random.

5. You're a cutting-edge geneticist who just invented an Animal Generator, which has an important compartment for figurines. There are 20 figurines of each type of small animal (turtle, duck, platypus) and 10 figurines of each type of large animal (bear, lion, bison, dragon), for a total of 100 figurines. When you click Go, it randomly chooses a figurine from the compartment and creates a real animal of that species.

   (a) Your machine actually consumes a figurine during the process of creating that animal. If you run the machine 15 times (so you are left with 85 figurines at the end), what is the probability you end up with exactly 3 animals from one species and exactly 2 animals of each of the other species? **Give your answer to 4 decimal places.**

   (b) You figured out how to prevent that machine from consuming the figurine when creating an animal. If you run the machine 15 times (so you still have 100 figurines at the end), what is the probability you end up with exactly 3 animals from one species and exactly 2 animals of each of the other species? **Give your answer to 4 decimal places.**

6. Let $\mathbf{X} = (X_1, ., X_n)$ be any **discrete** random variables. Without any independence assumptions, specifying the joint PMF is extremely complex; the chain rule from section 2.3 says that

$$P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \prod_{i=1}^{n} P\left(X_i = x_i \mid \bigcap_{j=1}^{i-1} X_j = x_j\right).$$

   This is not only extremely nasty to compute, but requires a lot of information to fully specify. So we often (incorrectly) assume mutual independence of all the random variables, to simplify our model to

$$P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \prod_{i=1}^{n} P(X_i = x_i).$$

   Without any independence assumptions, we have a realistic but intractable model. With full independence assumptions, we have a less realistic but tractable model. Is there some kind of middle ground?

   A **Bayesian network** is a directed acyclic graph (DAG) describing probabilistic dependencies between random variables, which fully specifies their joint distribution as the product of local conditional distributions. From a Bayesian network over $\mathbf{X} = (X_1, ., X_n)$, we define the joint distribution to be computed by:

$$P(X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \prod_{i=1}^{n} P\left(X_i = x_i \mid \bigcap_{j \in \mathsf{Parents}(X_i)} X_j = x_j\right)$$

   where $\mathsf{Parents}(X_i) = \{j \in \{1, \ldots, n\} : \text{there is a directed edge } X_j \to X_i\}$. That was a ton of notation, so let's see some concrete examples with a small number, $n = 4$.

For example, suppose we have 4 random variables X_1, X_2, X_3, X_4. Let's see the joint distribution decomposed in each of the three cases described above. For brevity, we will temporarily abuse notation and write $P(X_i)$ to mean $P(X_i = x_i)$.

1. No independence assumptions (the chain rule):

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_1, X_2)P(X_4 \mid X_1, X_2, X_3)$$

2. Full independence assumption:

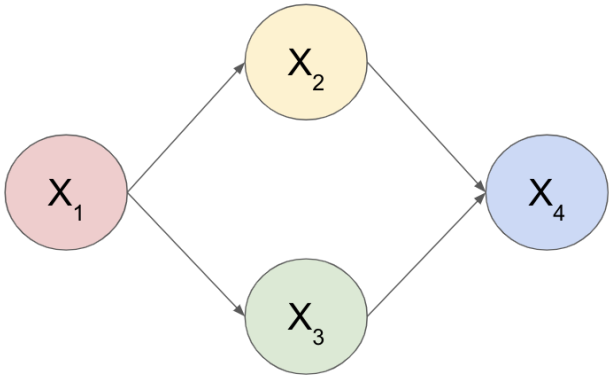
$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2)P(X_3)P(X_4)$$

3. The Bayesian network in Figure 1 below:

$$P(X_1, X_2, X_3, X_4) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_1)P(X_4 \mid X_2, X_3)$$

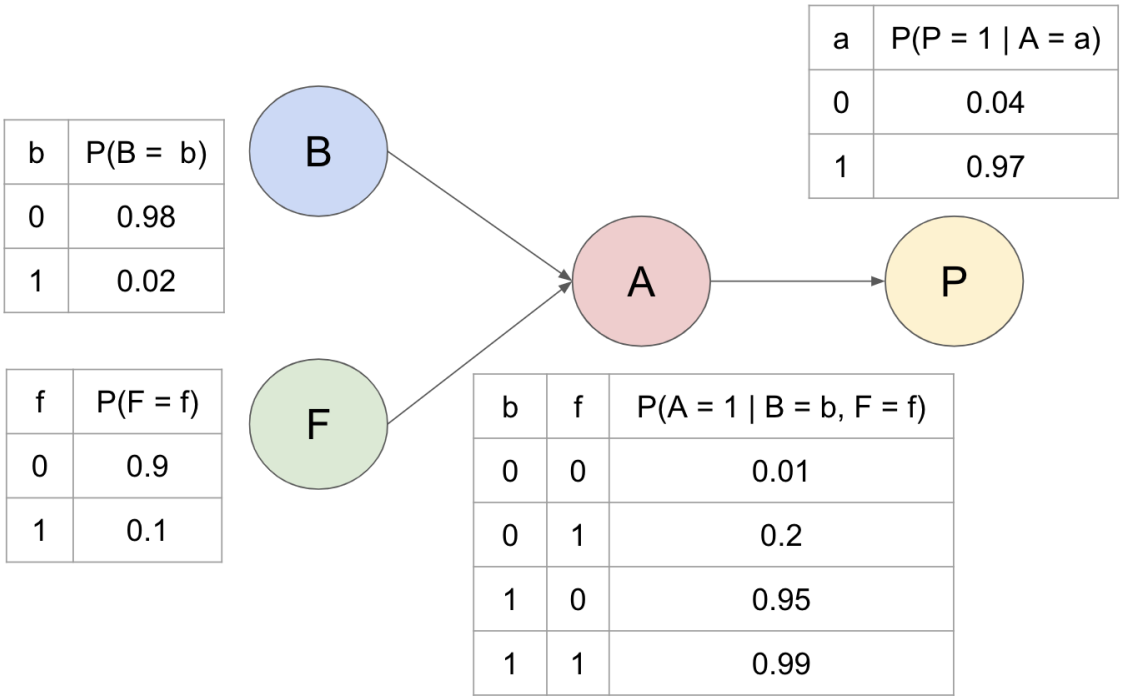
   since X_1 has no parents, X_2 and X_3 have just X_1 as a parent, and X_4 has X_2, X_3 as parents. Note how this is a "compromise" between the first two cases!

Figure 1: A Sample Bayesian network



Now it is your turn! Consider the following "Crime" Bayesian network in Figure 2, made of 4 **Bernoulli** RVs (only taking on values 0 or 1), where B indicates whether a burglary occurred, F indicates whether a fire occurred, A indicates whether the alarm sounds, and P indicates whether the police arrive. Convince yourself the Bayesian network's edges (specifying probabilistic dependencies) are reasonable. As for explaining some "missing" arrows, consider the following explanations. For example, we may reasonably assume that fires and burglaries occur independently, and we can further simplify our network to assume that the arrival of police is solely dependent on an alarm. Note that in practice, these may not entirely true, but these are approximations that help simplify our model. Next to each node (random variable), there is a **conditional probability table (CPT)**, describing the conditional distribution of the node given its parents.

Figure 2: The "Crime" Bayesian network



b	P(B = b)
0	0.98
1	0.02

a	P(P = 1 | A = a)
0	0.04
1	0.97

f	P(F = f)
0	0.9
1	0.1

b	f	P(A = 1 | B = b, F = f)
0	0	0.01
0	1	0.2
1	0	0.95
1	1	0.99

(a) The "Crime" Bayesian network specifies that the joint distribution $P(B = b, F = f, A = a, P = p)$ can be decomposed into what product of conditional probabilities? Then, compute $P(B = 1, F = 1, A = 0, P = 0)$ using your decomposition: the probability that there is a burglary and a fire, but the alarm doesn't sound and the police don't arrive. **Give your answer to 8 decimal places.**

(b) Compute the probability that there is no fire nor burglary, and the police don't arrive: $P(B = 0, F = 0, P = 0)$. Hint: Compute the entire joint distribution first like you did in the previous part, marginalizing (summing) over unspecified variable(s) as we learned in section 5.1. **Give your answer to 8 decimal places.**

(c) Compute the probability the alarm sounds, given there is no fire nor burglary and the police don't arrive: $P(A = 1 \mid B = 0, F = 0, P = 0)$. Hint: Use the definition of conditional probability $P(G \mid H) = P(G \cap H)/P(H)$. **Give your answer to 8 decimal places.**

(d) Now let's measure much we traded off in terms of being close to reality (no independence assumptions), and being computationally efficient (full independence assumption), with this particular Bayesian network made of 4 **Bernoulli** RV's, B, F, A, P. Note to fully specify the joint PMF $P(B = b, F = f, A = a, P = p)$, we need to be able to evaluate it all $2^4 = 16$ possible values of (b, f, a, p) which each take on either 0 or 1.

Important note for the below parts regarding the definition of "minimum": If we had a single discrete random variable X taking on values in $\Omega_X = \{1, 2, 3\}$, the **minimum** number of probabilities to specify this PMF is **two (not three)**, since if we provide just $p_X(1)$ and $p_X(2)$ for example, we can infer $p_X(3) = 1 - p_X(1) - p_X(2)$.

i. If we make no independence assumptions at all, what is the **minimum** number of probabilities we have to specify to be able to compute the joint PMF $P(B = b, F = f, A = a, P = p)$ at any inputs (b, f, a, p)?

ii. If we make the assumption that all the RV's are (mutually) independent of each other, what is the **minimum** number of probabilities we have to specify to be able to compute the joint PMF $P(B = b, F = f, A = a, P = p)$ at any inputs (b, f, a, p)?

iii. If we make the independence assumptions given by the "Crime" Bayesian network, what is the **minimum** number of probabilities we have to specify to be able to compute the joint PMF $P(B = b, F = f, A = a, P = p)$ at any inputs (b, f, a, p)?

(e) The Naïve Bayes classifier we discussed in section 9.3 was actually a special instance of a Bayesian network, as we made certain conditional independence assumptions! Assume the features are $(X_1, X_2, \ldots, X_n)$ (whether each of n words in the dictionary of all possible words appears or not in an email) and the label is Y (spam or ham). Sketch a diagram of what it looks like (there should be $n + 1$ nodes with some directed edges), and **make sure the directions of your edges are clear**! Your diagram should be a (directed acyclic) graph, which has the same format as the example Bayesian networks above (but without the CPT's at each node).

(f) **(Extra Credit)**: Define your own Bayesian network modelling a real-world scenario with at least 5 nodes and 4 (directed) edges. Explain why the choice of edges (and direction) make sense, specify the CPT (conditional probability table) with reasonable probabilities for each node given its parents, and compute an interesting probability! Your RV's need not be Bernoulli; they can take on more than 2 values if you wish (e.g., a weather RV W which could be in {rainy, cloudy, sunny, snowy}.)

7. A **discrete-time stochastic process (DTSP)** is a sequence of random variables $X_0, X_1, X_2, \ldots$, where X_t is the value at time t. For example, the temperature in Seattle or stock price of TESLA each day, or which node you are at after each time step on a random walk on a graph.

   In Figure 3, we have a graph with 5 nodes. Suppose we start at node 1, and at each time step, independently step to a neighboring node (with equal probability). For example; $X_0 = 1$ if we start at node 1, then X_1 could be either 2 or 3 (but not 4 or 5), etc. In fact, this DTSP has lots of nice properties, and is actually an example of a special type of DTSP called a **Markov Chain**, since it satisfies three properties:

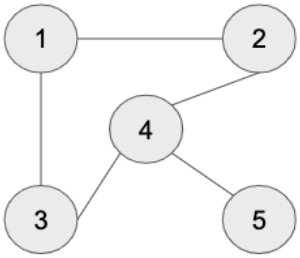
   I. We only have finitely many states (we have 5 in this example: $\{1, 2, 3, 4, 5\}$).

   II. We dont care about the past, given the present (in this example, at each time step, the distribution of where we go next ONLY depends on where we are currently).

   III. The transition probabilities are the same at every time step (in our example if we're at node 1, we go to node 2 or node 3 with probability 1/2, *regardless of what time it is*). For example, if we start at state 1 (meaning $X_0 = 1$), and we are also in state 1 at time $t = 152$ (meaning $X_{152} = 1$)), the probabilities of transitioning to 2 or 3 remain the same at $1/2$ each.

   Formally, a **Markov Chain** is a DTSP, with the additional following three properties:

Figure 3: Random Walk on a Graph



I. ...has a finite (or countably infinite) **state space** $\mathcal{S} = \{s_1, \ldots, s_n\}$ which it bounces between, so each $X_t \in \mathcal{S}$.

II. ...satisfies the **Markov property**. A DTSP satisfies the Markov property if the future is (conditionally) independent of the past given the present. Mathematically, it means, $P(X_{t+1} = x_{t+1} | X_0 = x_0, X_1 = x_1, \ldots, X_{t-1} = x_{t-1}, X_t = x_t) = P(X_{t+1} | X_t = x_t)$.

III. ...has **stationary transition probabilities**. Meaning, if we are at some state s_i, we transition to another state s_j with probability *independent* of the current time. Due to this property and the previous, the transitions are governed by n^2 probabilities: the probability of transitioning from one of n current states to one of n next states. These are stored in a square $n \times n$ **transition probability matrix (TPM)** P, where $P_{ij} = P(X_{t+1} = s_j | X_t = s_i)$ is the probability of transitioning from state s_i to state s_j for any/every value of t.

(a) Suppose we perform a random walk on the graph from Figure 3. Fill out the 5×5 transition probability matrix P in Figure 4 with simplified fractions; we've filled out the first row for you. The row represents the current state and the column represents the next state. So the first row represents our transition probabilities *from* state 1 to the other five states. It has zero probability of transitioning to state 1,4, and 5, but equal $\frac{1}{2}$ probability of transitioning to 2 and 3. Note: You are secretly computing the conditional PMF $P(X_{t+1} = j \mid X_t = i)$ for any $i, j \in \{1, 2, 3, 4, 5\}$ and fixed $t \in \{0, 1, 2, \ldots\}$.

(b) What is $P(X_2 = 4 \mid X_0 = 1)$? Show your work; an intuitive answer is not sufficient. Hint: Condition on what X_1 is and use the LTP.

(c) Suppose we weren't sure where we started. That is, let $v = \left(\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}\right)$ be such that $P(X_0 = i) = v_i$, where v_i is the i^{th} element of v (i.e., we start at one of the 5 positions uniformly at random). Think of this vector v as our belief distribution of where we are at time $t = 0$. First, compute vP, the matrix-product of v and P, the transition probability matrix. What does vP represent? If you haven't taken linear algebra yet, don't worry: vP is the following 5-dimensional row vector:

$$vP = \left(\sum_{i=1}^{5} P_{i1} v_i, \quad \sum_{i=1}^{5} P_{i2} v_i, \quad \sum_{i=1}^{5} P_{i3} v_i, \quad \sum_{i=1}^{5} P_{i4} v_i, \quad \sum_{i=1}^{5} P_{i5} v_i \right)$$

Give your answer to the first question as **5 simplified fractions**.
Hint 1: You can approach this by substituting each $P_{ij} = P(X_1 = j \mid X_0 = i)$ since the

TPM works for any time t. For example, the third of the 5 entries in vP is actually just $\sum_{i=1}^{5} P(X_1 = 3|X_0 = i)P(X_0 = i)$ after additionally plugging in $P(X_0 = i)$ for v_i.
Hint 2: The interpretation of what vP represents is very nice and simple.

(d) The **stationary distribution** of a Markov chain is the $|\mathcal{S}|$-dimensional row vector π such that the matrix equation $\pi P = \pi$ holds (and π is a valid probability mass function). For our example, $\pi = (\pi_1, \pi_2, \pi_3, \pi_4, \pi_5)$ is 5-dimensional, and contains 5 probabilities which sum to 1. The intuition/interpretation of π is that it gives the probabilities of being in each state in the "long run". That is, for t large enough, $\pi_i = P(X_t = i) = P(X_{t+1} = i) = P(X_{t+2} = i) = \ldots$. For example, if $\pi = (0.25, 0.15, 0.45, 0.05, 0.1)$, then after a "long time", we expect to be in state 2 with probability 0.15 at *every* time step. Using your answer to the previous part, explain in 1-2 sentences why solving $\pi P = \pi$ would give us the stationary distribution.

Figure 4: Transition Probability Matrix P of Random Walk

$$P = \begin{bmatrix} 0 & 1/2 & 1/2 & 0 & 0 \\ TODO & TODO & TODO & TODO & TODO \\ TODO & TODO & TODO & TODO & TODO \\ TODO & TODO & TODO & TODO & TODO \\ TODO & TODO & TODO & TODO & TODO \end{bmatrix}$$

8. [**Coding+Written**] Markov Chain Monte Carlo (MCMC) is a technique which can be used to solve hard optimization problems (among other things). The general strategy is as follows:

   I. Define a Markov Chain with states being possible solutions, and (implicitly defined) transition probabilities that result in the stationary distribution π having higher probabilities on "good" solutions to our problem. We don't actually compute π, but we just want to define the Markov Chain such that the stationary distribution would have higher probabilities on more desirable solutions.

   II. Run MCMC (simulate the Markov Chain for many iterations until we reach a "good" state/solution). This means: start at some initial state, and transition according to the transition probability matrix (TPM) for a long time. This will eventually take us to our stationary distribution which has high probability on "good" solutions!

   In this question, there is a collection of n items available to us, and each has some value $v_i > 0$ and weight $w_i > 0$ (and there is only one item of each type available - we either take it or leave it). We want to find the optimal subset of items to take which maximize the total value (the sum of the values of the items we take), subject to the total weight (the sum of the weights of the items we take) being less than some $W > 0$. (This is known as the **knapsack problem**, and is known to be NP-Hard). In `items.txt`, you'll find a list of potential items with each row containing the name of the item (string), and its value and weight (positive floats).

   You will implement an MCMC algorithm which attempts to solve this NP-Hard problem. Pseudocode is provided below, and a detailed explanation is provided immediately after.

Algorithm 2 MCMC for 0-1 Knapsack Problem

subset $\leftarrow$ vector of n zeros, where subset is always a binary vector in $\{0, 1\}^n$ which represents whether or not we have each item. (Initially, start with an empty knapsack).
best_subset $\leftarrow$ vector of n zeros
for $t = 1, \ldots,$ NUM_ITER **do**
    $k \leftarrow$ a random uniform integer in $\{1, 2, \ldots, n\}$.
    new_subset $\leftarrow$ subset but with subset$[k]$ flipped ($0 \rightarrow 1$ or $1 \rightarrow 0$).
    $\Delta \leftarrow$ value(new_subset) $-$ value(subset)
    if new_subset satisfies weight constraint (total weight $\leq W$) **then**
        if $\Delta > 0$ OR ($T > 0$ AND Unif$(0, 1) < e^{\Delta/T}$) **then**
            subset $\leftarrow$ new_subset
    if value(subset) $>$ value(best_subset) **then**
        best_subset $\leftarrow$ subset

   The MCMC algorithm will have a "temperature" parameter T which controls the trade-off between exploration and exploitation. The state space $\mathcal{S}$ will be the set of all subsets of n items. We will start with a random state (subset). At at each iteration, propose a new state (subset) as follows: choose a random index i from $\{1, 2, \ldots, n\}$, and take item i if we don't already have it, or put it back if we do.

- If the proposed subset is infeasible (doesn't fit in our knapsack because of the weight constraint), we return to the start of the loop and abandon the newly proposed subset.
- Suppose then it is feasible. If it has higher total value (is better) than the current route, we will always transition to it (exploitation). Otherwise if it is worse but $T > 0$, with probability $e^{\Delta/T}$, update the current subset to the proposed subset, where $\Delta < 0$ is the decrease in total value. This allows us to transition to a "worse" subset occasionally (exploration), and get out of local optima! Repeat this for NUM_ITER transitions from the initial state (subset), and output the highest value subset during the entire process (which may not be the final subset).

(a) What is the size of the Markov Chain's state space $\mathcal{S}$ (the number of possible subsets)? As NUM_ITER $\to \infty$, are you guaranteed to eventually see all the subsets (consider the cases of $T = 0$ and $T > 0$ separately)? Briefly justify your answers.

(b) Let's try to figure out what the temperature parameter T does.

i. Suppose $T = 0$. Will we ever get to a worse subset than before as we transition?

ii. Suppose $T > 0$. For a fixed T, does the probability of transitioning to a worse subset increase or decrease with larger absolute values of Δ (larger meaning "more negative" values, since $\Delta < 0$)? For a fixed Δ, does the probability of transitioning to a worse subset increase or decrease with larger values of T? Explain briefly how the temperature parameter T controls the degree of exploration we do.

(c) Implement the functions `value`, `weight`, and `mcmc` in `mcmc_knapsack.py`.

You must use `np.random.rand()` to generate a continuous $Unif(0, 1)$ rv, and `np.random.randint(low (inclusive), high (exclusive))` to generate your random index(es). Make sure to read the documentation and hints provided! You **must use this strategy exactly** to get full credit - we will be setting the random seed so that everyone should get the same result if they follow the pseudocode. For Line 4 in the pseudocode, since Python is 0-indexed, generate a random integer in $\{0, 1, \ldots, n-1\}$ instead, otherwise the autograder may fail.

(d) We've called the `make_plot` function to make a plot where the x-axis is the iteration number, and the y-axis is the current knapsack value (not necessarily the current best), for `ntrials=10` different runs of MCMC. You should attach 4 plots which are generated for you (one per temperature), and each plot should have 10 curves (one per trial). Which value of T tended to most reliably produce high knapsack values?

9. **(Extra Credit)**: If you worked with a partner that you were randomly paired with during a social event or through the partner survey, attach a screenshot here to get extra credit (you can get extra credit even if you use the same partner)! If it was a social event zoom call, your screenshot must include the zoom meeting information to prove it was one of our social zoom meetings.