DS102 - Homework 1

If you are submitting a handwritten version please make sure your answers are legible as you may lose points otherwise.
Data science is a collaborative activity. While you may talk with others about the homeworks, we ask that you write your solutions individually. If you do discuss the homework with others please include their name in your submission.
Due by: 1:59pm, Tuesday 22th October, 2019

1. (20 points) **Gibb's sampling for hierarchical models.**

   This problem will look at using Gibb's sampling for two different simulation strategies in hierarchical models. The simulations we will study have different goals, and the strategies will reflect these goals. At the end of this question you'll be asked to compare and contrast the goals, the strategies, and the models we choose to represent them.

   The setting is the following: there is a bike rideshare company that rents out bikes in a certain town, and they'd like to estimate the number of bikes that will be rented on a given day, so that they know how many bikes to bring in to the shop for service (if it's projected to be a slow day for bike rentals, the company will service more bikes on that day). We have a prior belief that people will be more or less likely to rent bikes from the rideshare company depending on (a) the weather, and (b) whether it is a working day, or a weekend day.

   Please use the provided skeleton code in `gibbs_hm_skeleton_code.ipynb`. There are sections in that code for you to answer every questions, so for this problem please save the notebook output as a pdf (exactly how you do so for labs), append it to your answers for other questions, and submit that. The major steps are detailed here with explanations, and the notebook has some additional print statements for you to run to help you debug your work.

   The first part of this problem will model this process under a hierarchical model, and draw simulations from this model to answer questions.

   (a) (1 point) To get started, import the dataset in `bikeshare.csv` (we suggest you use pandas for this).[1]

   Each row corresponds to a unique day with the following associated information:

sunny	working_day	month	num_renters

   Plot a histogram (with 20 bins) of the total number of bikes rented on a given day in the dataset. Remember to label the axes.

[1] If you're interested, the data in this file is a subset of data from:
`https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset`

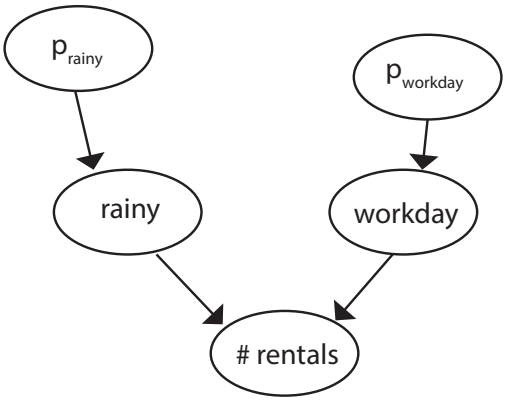

Figure 1: Hierarchical model for bikeshare rentals.

(b) (4 points) Now we'll use Gibb's sampling to generate samples from the hierarchical model described in Figure 1. We have the dataset of possible conditions and outcomes available to us, so we'll use that to seed the distributions in our hierarchical model.

Specifically, given $i \in \{0, 1\}$ representing workday ($i = 1$) or weekend ($i = 0$) and $j \in \{0, 1\}$ representing a sunny day ($j = 1$) or a rainy day ($j = 0$), we will model the number of bike rentals on a day x as

$$\mathbb{P}(x|\text{workday} = i, \text{sunny} = j) \sim \mathcal{N}(\bar{\mu}_{ij}, \bar{\sigma}_{ij})$$

where $\bar{\mu}_{ij}$ is the average number of rentals in our data set for entries that agree with the settings workday $= i$ and sunny $= j$, and $\bar{\sigma}_{ij}$ is the analogous sample standard deviation.

(i) Calculate the means and standard deviations $\bar{\mu}_{ij}, \bar{\sigma}_{ij}$ by filling in the function `get_subgroup_statistics()`

(ii) The simulation sampling procedure proceeds as follows. For T rounds, do the following:

1. Set p_{sun} to be the fraction of sunny days you see in the dataset.
   Set $p_{workday} = 5/7$.
2. Sample $i \sim Bernoulli(p_{sun})$, $j \sim Bernoulli(p_{workday})$
3. Sample $x \sim \mathcal{N}(\bar{\mu}_{ij}, \bar{\sigma}_{ij})$
4. Append x to the sample of simulated rental counts.

Implement this sampling procedure in the function called `simulate_rentals()`.

(c) (2 points) Draw 1000 samples; plot the histogram of the resulting draws for the number of bikes in a given day.

(d) (3 points) Let's say the forecast tomorrow says there's an 80% chance of rain. Change p_{sun} to reflect this probability (ignoring any previous rates of rain). Run 1000 simulations with this new probability of rain. Plot the resulting histogram on the number of bikes rented that day, under the knowledge that tomorrow is a workday. Plot another histogram if tomorrow is actually a weekend day (you should be plotting two histograms).

(e) (2 points) Do the same thing as above, but using only the first ten rows of the dataset. This simulates the event where we only collected data for ten days, and that's all we've got. Can you run the procedure to its completion?

(f) (4 points) Now, we'll switch strategy slightly. Rather than incorporating our dataset to seed individual probabilities governing each of the parameters $\mu_{00}, \mu_{01}, \mu_{10}, \mu_{11}$, we'll use the data to sample from the joint distribution on $\mathbb{P}(x, \theta)$, where x is now a dataset of observations. Recall that

$$\mathbb{P}(x, \theta) = \mathbb{P}(x|\theta)\mathbb{P}(\theta)$$

where $\theta = [\mu_{00}, \mu_{01}, \mu_{10}, \mu_{11}]$, and $x \in \mathbb{R}^n$ is the number of rentals on each day.

Similarly to before, we'll model the likelihood as a a set of Normal distribution. Let $s_i \in \{0, 1\}$, $w_i \in \{0, 1\}$ denote whether the ith day in the dataset was sunny, and a workday, respectively. Then we model the likelihood for the rental counts of that day as:

$$\mathbb{P}(x_i|\theta) \sim \mathcal{N}(\mu_{w_i,s_i}, \sigma_{w_i,s_i})$$

We'll also put independent prior probabilities on each of the means:
$\mathbb{P}(\theta) = \mathbb{P}(\mu_{00}, \mu_{01}, \mu_{10}, \mu_{11}) = \mathbb{P}(\mu_{00})\mathbb{P}(\mu_{01})\mathbb{P}(\mu_{10})\mathbb{P}(\mu_{11})$ as normal with specified means and standard deviations (see the notebook for details).

Using the prior probabilities and the likelihood, we will now use the following sample procedure.
For $t = 1, ..., T$:

1. Sample $\mu_{00} \sim \mathbb{P}(\mu_{00})$, $\mu_{01} \sim \mathbb{P}(\mu_{01})$, $\mu_{10} \sim \mathbb{P}(\mu_{10})$, and $\mu_{11} \sim \mathbb{P}(\mu_{11})$.
2. Compute $l = \mathbb{P}(x, \theta)$.
3. Append $[\mu_{00}, \mu_{01}, \mu_{10}, \mu_{11}, l]$ to the sample.

Code up this sampling scheme in the code provided, including the function called `gibbs_sampling_for_posterior()`.

(g) (2 points) Plot the estimated distributions of the posterior marginals $\mathbb{P}(\mu_{00}), \mathbb{P}(\mu_{01}), \mathbb{P}(\mu_{10})$, and $\mathbb{P}(\mu_{11})$ from your sample. To do so, plot a histogram of each draw of each mean in your sample, weighted by the calculated posterior density associated with that draw. Do this using first $n = 10$ datapoints, and then $n = 100$ datapoints from the total data.

(h) (2 points) Compare and contrast the two motivations above. Specifically, address (i) what quantity are you sampling and plotting in each method, and (ii) which approach would you prefer for a small dataset, and why?

2. (20 points) In this problem we will solve the German Tanks problem from lecture using both Frequentist and Bayesian approaches. The setup is as follows. During World War II, the German army began construction of new, more agile tanks. Luckily for the Allies, the Germans labeled their tanks $1, 2, 3, ..., N$, though N was unknown. Allied intelligence indicated that $N = 1550$. The Allies' statisticians, however, used the labels from tanks that were destroyed in battle to estimate N, and came up with a much better estimate of $N = 327$. After the war, German records showed that $N = 342$. In this problem we will perform the same analysis as the Allied statisticians.

   The statisticians had a sample of k destroyed tanks, with serial numbers $Y_1, ..., Y_k$ which they assumed had been sampled uniformly at random from $\{1, ..., N\}$ without replacement. Let us denote $Y_{(k)}$ as the largest serial number in the sample of k tanks.

   (a) (2 points) Show that the probability that maximum serial number, $Y_{(k)} = i$ is given by:

$$\mathbb{P}(Y_{(k)} = i) = \begin{cases} \frac{\binom{i-1}{k-1}}{\binom{N}{k}} & ; \quad k \leq i \leq N \\ 0 & ; \quad \text{otherwise} \end{cases}$$

   Recall that we assumed that the k tanks were sampled uniformly at random without replacement from $\{1, ..., N\}$. (Hint: there are $\binom{N}{k}$ ways of sampling k tanks from $1, ..., N$ without replacement and N is fixed).

   (b) (3 points) Given the expression for $\mathbb{P}(Y_{(k)} = i)$, show that:

$$\mathbb{E}[Y_{(k)}] = k\frac{(N+1)}{k+1}$$

   (Hint: use the hockey-stick identity $\sum_{i=k}^{N} \binom{i}{k} = \binom{N+1}{k+1}$).

   (c) (5 points) Using the expression for $\mathbb{E}[Y_{(k)}]$, derive an unbiased estimator $\hat{N}$ for N, using only $Y_{(k)}$ and the sample size k. Code up this expression in the function `frequentist_estimator()` provided notebook `german_tanks_skeleton_code.ipynb`. (Recall that an unbiased estimator, $\hat{N}$ for N must satisfy $E[\hat{N}] = N$).

   (d) (2 points) We will now take the Bayesian approach. To do this, we must first define a prior $\pi(N)$ over N. For simplicity, we take a uniform prior over $N_{min} < Y_{(k)} < N_{max}$:

$$\pi(N) = \begin{cases} \frac{1}{(N_{max} - N_{min})} & ; N_{min} < N < N_{max} \\ 0 & ; \quad \text{otherwise} \end{cases}$$

   Given this prior derive an expression for the posterior $\mathbb{P}(N|Y_{(k)})$. You do not need to compute the normalizing constant.

(e) (8 points) In the Ipython notebook, fill in the function `posterior_distribution()` which returns $\mathbb{P}(N = n|Y_{(k)} = i)$ for a given prior $\pi(N)$ and `credible_interval()` which returns the 95% credible interval of N. Use the skeleton code to plot the frequentist estimates and credible intervals for the three sets of observations given: `serial_numbers_1`,`serial_numbers_2`,and `serial_numbers_3`.

3. (20 points) In this problem we will the Markov, Chebyshev, and Chernoff bounds for sums of independent but not identically distributed random variables. Suppose we have n soccer players each taking one penalty kick. We would like to get high probability bounds on the number of shots that are made by all the players. For each player $j \geq 1$ let I_j be the indicator that player j makes the shot independent of all the other players. Assume we know from looking at historical data that $\mathbb{P}(I_j = 1) = p_j$. Let $X = I_1 + I_2 + \ldots + I_n$, be the count of the total number of penalties scored by the players.

   (a) (2 points) Find $\mu = E(X)$ and $\sigma^2 = Var(X)$ in terms of $p_1, p_2, \ldots, p_n$.

   (b) (3 points) Find Markov's bound on $\mathbb{P}(X \geq (1 + c)\mu)$ for some $c > 0$.

   (c) (3 points) Find Chebyshev's bound on $\mathbb{P}(X \geq (1 + c)\mu)$ in terms of μ and σ.

   (d) (2 points) If all the p_j's are equal to p, what is the value of the bound in (c)? How does this compare to the bound you got in part (b)?

   (e) (2 points) Show that the moment generating function of I_j is given by $M_{I_j}(t) = 1 + p_j(e^t - 1)$ for all t. (Remember that the moment generating function is given by: $M_{I_j}(t) = \mathbb{E}[e^{tI_j}]$)

   (f) (4 points) An useful exponential bound is that $e^x \geq 1 + x$ for all x. Use the fact to show that $M_X(t) \leq \exp\big(\mu(e^t - 1)\big)$ for all t. (Hint: use the fact that X is the sum of independent random variables.)

   (g) (4 points) Use Chernoff's method and the bound in (c) to show that

$$\mathbb{P}\big(X \geq (1 + c)\mu\big) \;\leq\; \Big(\frac{\exp(c)}{(1 + c)^{1+c}}\Big)^{\mu}$$

   If $g(c) = \exp(c)/(1 + c)^{1+c}$ is small, and $p_j = p$ for all $j \geq 0$, how does this bound depend on n?