Overview

Submit your writeup, including any code, as a PDF via gradescope.[1] We recommend reading through the entire homework beforehand and carefully using functions for testing procedures, plotting, and running experiments. Taking the time to reuse code will help in the long run!

Data science is a collaborative activity. While you may talk with others about the homework, please write up your solutions individually. If you discuss the homework with your peers, please include their names on your submission. Please make sure any handwritten answers are legible, as we may deduct points otherwise.

The One with all the Beetles

1. (9 points) Cindy has an inordinate fondness for beetles and for statistical modeling. She observes one beetle everyday and keeps track of their lengths. From her studies she feels that the beetle lengths she sees are uniformly distributed. So she chooses a model that the lengths of the beetles come from a uniform distribution on $[0, w]$: here w is an unknown parameter corresponding to the size of the largest possible beetle. Since the maximum size w is unknown to her, she would like to estimate it from the data. She observes lengths of n beetles, and calls them $x_1, \ldots, x_n$.

   (a) (1 point) What is the likelihood function of the observations $x_1, \ldots, x_n$? Express your answer as a function of the largest size parameter w.

   Hint: Your answer should include the indicator function $\mathbb{1}(\max_i x_i \leq w)$. To see why, consider what happens if $w = 3$ cm and $x_1 = 5$ cm.

   (b) (2 points) Use your answer from part (a) to explain why the maximum likelihood estimate for w is $\max_i x_i$.

   (c) (2 points) Cindy decides to instead use a Bayesian approach. She has a prior belief that w follows a *Pareto distribution* with parameters $\alpha, \beta > 0$. We can write:

   $$w \sim \text{Pareto}(\alpha, \beta)$$
   $$p(w) = \frac{\alpha \beta^\alpha}{w^{\alpha+1}} \mathbb{1}(w > \beta)$$

   If she observed values $x_1, \ldots, x_n$, show that the posterior distribution for w is also a Pareto distribution, and compute the parameters as a function of α, β, and the observations $x_1, \ldots, x_n$.

   (d) (2 points) Provide a short description in plain English that explains what the parameters of the Pareto distribution mean, in the context of the Pareto-uniform conjugate pair.

   Hint: For the Beta-Binomial conjugate pair that we explored in class, the answer would be that the Beta parameters acted as pseudo-counts of observed positive and negative examples.

---

[1] In Jupyter, you can download as PDF or print to save as PDF

(e) (2 points) Let us say Cindy started with the initial prior values $\alpha = 1$ and $\beta = 10$ on day 0. Use the code in beetledata.py to generate the data for the lengths of the beetles she sees, starting with day one to day one hundred. Use the data to make a graph of one curve for each of the days $1, 10, 50$ and 100 (so four curves total), where each curve is the probability density function of Cindy's posterior for the respective day. As with other familiar distributions, you can find the Pareto distribution in scipy.

(f) (0 points) (Optional) Use pymc3 to sample from the posterior for days $1, 10, 50$ and 100 and plot a density function for each of the cases. Compare the results from the analytic and simulation based computation of the densities.

Bayesian Fidget Spinners

2. (10 points) Nat's company manufactures fidget spinners. The company uses two factories, which we'll call factory 0 and factory 1. Each fidget spinner from factory k is defective with probability q_k ($k \in \{0, 1\}$). Nat knows that factory 0 produces fewer defective fidget spinners than factory 1 (in other words, $q_0 < q_1$).

She receives n boxes full of fidget spinners, but the boxes aren't labeled (in other words, she doesn't know which box is from which factory). For each box, she starts randomly pulling out fidget spinners until she finds a defective one, and records how many fidget spinners she pulled out (including the defective one). She calls this number x_i for box i, for $i = 1, \ldots, n$.

She wants to estimate the following pieces of information:

- Which boxes came from factory 0, and which came from factory 1? She defines a binary random variable for each box z_i with the factory label (i.e., $z_i = 0$ if box i is from factory 0, and $z_i = 1$ if box i is from factory 1).
- How reliable is each factory? In other words, what are q_0 and q_1?

Inspired by what she learned about Gaussian mixture models, she sets up the following probability model:

$$z_i \sim \text{Bernoulli}(\pi)$$
$$q_k \sim \text{Beta}(\alpha_k, \beta_k)$$
$$x_i | z_i, q_0, q_1 \sim \text{Geometric}(q_{z_i})$$

(a) (1 point) Draw a graphical model for the probability model described above if $n = 2$ (i.e., there are only two boxes of fidget spinners).

Nat decides to implement the model above setting the following hyperparameters:

$$\pi = 0.45, \quad q_0 \sim \text{Beta}(1, 5), \quad q_1 \sim \text{Beta}(5, 1)$$

(b) (2 points) The choices of the parameters in Nat's model represents her beliefs about the factories.

     i. (1 point) From her choice for π, what can we infer about her beliefs about the two factories?

     ii. (1 point) Similarly, from her choices for α and β, what can we infer about her beliefs about the factories?

(c) (5 points) Use `fidget_data.py` to generate the data that Nat observes, then, using PyMC3, fit the model outlined above, setting the hyperparameters to the values that Nat chose. Obtain 1000 samples from the posterior distribution $p(q_0, q_1 | x_1, \ldots, x_n)$, and generate a scatterplot (one point per sample).

You can use the code below (also provided to you in `fidget_model.py`) to help you get started.

```python
import pymc3 as pm


alphas = ...
betas = ...
pi = ...

with pm.Model() as model:
    z = pm.Bernoulli(
        # Define the Bernoulli Model Here
    )

    # Hint: you should use the shape= parameter here so that
    # q is a PyMC3 array with both q0 and q1.
    q = ...

    # Hint: it may be useful to use "fancy indexing" like we did in
    class.
    # See below for an example
    X = pm.Geometric(
        # DEFINE THE GEOMETRIC MODEL HERE
    )

    trace = ....

# FANCY INDEXING

my_binary_array = np.array([0, 0, 1, 1, 0, 1])
my_real_array = np.array([0.27, 0.34])
print(my_real_array[my_binary_array])
```

     i. Under the posterior, what is the probability that factory 0 produces more boxes than factory 1?

     ii. What is your median estimate of factory 0's defect rate, based on the samples from the posterior?

(d) (2 points) Nat's friend Yaro suggests using Gibbs sampling. What is the Gibbs sampling update for q_k? Your answer should be in the form of a well-known distribution, along with values for the parameter(s) of that distribution. Justify your answer.

3

Hint: you can derive the update analytically, or you can use the fact that the Beta distribution is a conjugate prior for a Geometric likelihood.

Rejection Sampling

3. (6 points) Consider the function

$$g(x) = \cos^2(12x) \times |x^3 + 6x - 2| \times \mathbb{1}_{x \in (-1, -.25) \cup (0, 1)}.$$

In this problem, we use rejection sampling to generate random variables with pdf $f(x) = cg(x)$.

(a) (2 points) Plot g over its domain. What is a uniform proposal distribution q that covers the support of f? What is the largest possible constant M such that the scaled target distribution $p(x) = Mg(x)$ satisfies $p(x) \leq q(x)$ for all x?

(b) (2 points) Suppose you run rejection sampling with target p and proposal q from part (a) until you generate n samples and your sampler runs a total of $N \geq n$ times, including n acceptances and $N - n$ rejections. Explain how you can use n, N and M to estimate c.
Hint: the ratio of acceptances n to total runs N is an approximation of the ratio between the area under the curve $p(x)$ and the area under $q(x)$.
Hint: remember what happens if you integrate a pdf over its entire support.

(c) (2 points) Use rejection sampling to generate a sample of size 10^3 from $p(x)$. Since $f(x)$ is a pdf and it's proportional to $p(x)$, we can display its estimate easily: plot a normalized histogram of your sample, and overlay a smooth kernel density estimate, that will provide more information on the shape of the estimated distribution. Repeat the previous steps increasing the number of samples to 10^6.