ISyE 6416: Computational Statistics
Homework 6 (Last Homework)
(100 points total.)

- This homework is due on Friday April 19.
- Please write your team member's name is you collaborate.

1. **Recommender systems.** (20 points)

   Take our movie recommender systems data, and perform recommendation using the following methods. Hint, when using features, expanding the categorical features using "one-hot" keying.

   (a) User-based collaborative filter to recommend 5 movies to each user. Try several similarity metrics, which are defined in the forms of

   $$\text{sim}(u, v) = e^{-(d(u,v))^2},$$

   where

   i. (1) $d(u, v) = \ell_2$ distance of the features, i.e., if x_u is the feature vector of the first movie, and x_v is the feature vector of the second movie, then $d(u, v) = \|x_u - x_v\|_2$.

   ii. (2) $d(u, v) = \ell_1$ distance of the features, i.e., $d(u, v) = \|x_u - x_v\|_1$;

   iii. (3) $d(u, v)$ =Hamming distance of the features, i.e., $d(u, v) = \|x_u - x_v\|_0$;

   (b) Item-based recommendation to recommend 5 movies to each user. try using the above three metrics respectively.

   (c) Using matrix completion to fill out missing entries to recommend 5 movies to each user.

   Report all of the three findings in an excel spread sheet (with 3 tabs for each of methods), and each row is in the format of (`name, recommended movie 1, score of the recommended movie 1, ..., recommended movie 5, score of the recommended movie 5`).

2. **Importance sampling.** (20 points)

   Similar to the example we had in lecture, using importance sampling to evaluate tail probability of Gaussian random variable. Assume X is $\mathcal{N}(\mu_0, 1)$. We want the right tail probability $\alpha = \mathbb{P}\{X \geq z\}$. For $z \gg \mu_0$, α is very small, and estimating this small probability accurately is not easy. Now to improve accuracy, we samples from another Gaussian random variable mean $\mu_1 = z$ and variance equal to 1.

   (a) Derive the important ratio.

   (b) Write down the importance sampling algorithm

   (c) Now assume $\mu_0 = 1$, $z = 10$. Using $N = 100$ Monte Carlo trials. Evaluate the tail probability using direct Monte Carlo $\widehat{I}_1$, and using importance sampling using $\widehat{I}_2$. Now repeat this 500 times, to compare the variance of $\widehat{I}_1$ and $\widehat{I}_2$.

3. **Bootstrapping.** (20 points)

Efron (1982) analyzes data on law school admission, with the object being to examine the correlation between LSAT score and the first year GPA. For each of 15 law schools, we have the following pair of data points:

(576, 3.93)	(635, 3.30)	(558, 2.81)	(578, 3.03)	(666, 3.44)
(580, 3.07)	(555, 3.00)	(661, 3.43)	(651, 3.36)	(605, 3.13)
(653, 3.12)	(575, 2.74)	(545, 2.76)	(572, 2.88)	(594, 2.96)

(a) calculate the correlation coefficient between LSAT and GPA.

(b) use the nonparametric bootstrapping to estimate the standard deviation and confidence interval of the correlation coefficient. Use $B = 1000$ batches, and each batch consists of $N = 15$ re-samples. For confidence interval, use $\alpha = 0.05$.

(c) use the parametric bootstrapping to estimate the standard deviation of the correlation coefficient. Assume that $(LSAT, GPA)$ has bivariate normal distribution and estimate the five parameters. Then generate 1000 batches of 15 samples from this bivariate normal distribution.

4. **Bayesian inference using Metropolis-Hastings algorithm.** (20 points)

Implement the Metropolis algorithm. Parameter for binomial distribution is probability of success $\theta \in [0, 1]$, $n = 20$. Assume the observed data vector gives $S_n = 5$.

(a) Assume the prior distribution as in our lecture, $\pi(\theta) = 2\cos^2(4\pi\theta)$. Generate samples from the posterior distribution $\pi(\theta|Y)$. Discretize θ to be a uniform grid of points [0, 1/10, ..., 1]. Run the chain for $n = 100$, 500, 1000, and 5000 time steps, respectively. For each time step, compare the empirical distributions with the desired posterior distribution $\pi(\theta|Y)$. (Hint: you may use ergodicity: hence the distribution of states can be estimated from one sample path when the number of time steps is large (e.g. 500).)

(b) Following from the previous question, evaluate the mean of the posterior distribution (this gives an estimator for the parameter value), and $\mathbb{E}^{\pi(\theta|Y)}\{[\theta - 1/2]^2\} = \int (\theta - 1/2)^2 \pi(\theta|Y) d\theta$.

(c) Now assume the the prior distribution is given by $\pi(\theta)$ is a uniform distribution over [0, 1]. Repeat the above questions, (a) and (b).

5. **Gibbs sampling for change-point detection.** (20 points)

Suppose that we observe a sequence of counts $x_1, x_2, \ldots, x_N$ where the average of the counts has some value for time steps 1 to n, and a different value for time steps $n + 1$ to N. We model and counts at each time step i as a Poisson variable, which has the following density function

$$\mathbb{P}(X = x; \lambda) = e^{-\lambda} \frac{\lambda^x}{x!}$$

where λ is the mean of the distribution. We model and mean λ as a Gamma distribution, which has the following density function:

$$\mathbb{P}(X = \lambda) = \frac{1}{\Gamma(a)} b^a \lambda^{a-1} \exp(-b\lambda).$$

The initial mean λ_1 jumps to a new value λ_2 after a random time step n. Thus the generative model is as follows:

$$\begin{aligned} n &\sim \text{Uniform}(1, 2, \ldots, N) \\ \lambda_i &\sim \text{Gamma}(a, b), i = 1, 2 \\ x_i | n, \lambda_1, \lambda_2 &\sim \begin{cases} \text{Poisson}(\lambda_1) & 1 \leq i \leq n \\ \text{Poisson}(\lambda_2) & n < i \leq N \end{cases} \end{aligned}$$

The problem of inferring the posterior distribution over the latent variables n, λ_1, λ_2 can be solved via Bayes formula:

$$\mathbb{P}(\lambda_1, \lambda_2, n | x_{1:N}) \propto \mathbb{P}(x_{1:n} | \lambda_1, n) \mathbb{P}(x_{n+1:N} | \lambda_2, n) \mathbb{P}(\lambda_1) \mathbb{P}(\lambda_2) \mathbb{P}(n).$$

(a) Derive the posterior distribution $\mathbb{P}(\lambda_1 | n, \lambda_2, x_{1:N})$

(b) Derive the posterior distribution $\mathbb{P}(\lambda_2 | n, \lambda_1, x_{1:N})$

(c) Finally, derive the posterior distribution $\mathbb{P}(n | \lambda_1, \lambda_2, x_{1:N})$

(d) Now using posterior conditionals for all the latent variables, write code to simulate samples to draw inference. Start by simulating data from the generative process. Then using the simulated data, report the results for one MCMC chain. Ran the chain for 5200 iterations. Discard the first 200 iterations as burn-in. Show result (plot the histogram of the posterior distribution for λ_1, λ_2, and n for the remaining 5000 iterations.)