ISyE 6416: Computational Statistics
Homework 7 (Last Homework)
(100 points total.)

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

1. **Bootstrapping.** (25 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.

2. **Random forest for email spam classifier** (25 points)

   Your task for this question is to build a spam classifier using the UCR email spma dataset `https://archive.ics.uci.edu/ml/datasets/Spambase` came from the postmaster and individuals who had filed spam. Please download the data from that website. The collection of non-spam e-mails came from filed work and personal e-mails, and hence the word 'george' and the area code '650' are indicators of non-spam. These are useful when constructing a personalized spam filter. You are free to choose any package and any language to choose for this homework.

   One would either have to blind such non-spam indicators or get a very wide collection of non-spam to generate a general purpose spam filter. Load the data.

   (a) (5 points) How many instances of spam versus regular emails are there in the data? How many data points there are? How many features there are?

   Note: there may be some missing values, you can just fill in zero.

(b) (10 points) Build a classification tree model (also known as the CART model). In R, this can be done using library(rpart). In our answer, you should report the tree models fitted similar to what is shown in the "Random forest" lecture, Page 16, the tree plot. In R, getting this plot can be done using prp function in library(rpart).

(c) (15 points) Also build a random forrest model. In R, this can be done using library(randomForest).

Now partition the data to use the first 80% for training and the remaining 20% for testing. Your task is to compare and report the test error for your classification tree and random forest models on testing data, respectively. To report your results, please try different tree sizes. Plot the curve of test error versus the number of trees used in random forest, similar to our lecture.

3. **Importance sampling.** (25 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 $\widehat{I}_2$.

4. **Bayesian inference using Metropolis-Hastings algorithm.** (25 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, $1/10, \ldots, 1$]. 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).