Final Exam

Mori Jamshidian

Spring 2024

Contents

Exam Rules and Instructions **2**

Problem 1 (a) [5 points] . . . 3

Problem 1(b) [3 Points] . . . 3

Problem 2(a) [2 points] . . . 3

Problem 2(b) [3 points] . . . 3

Problem 2 (c) [5 points] . . . 3

Problem 3 . . . 4

Problem 3(a) [3 points] . . . 4

Problem 3 (b) [5 points] . . . 4

Problem 3 (c) [5 points] . . . 4

Problem 3 (d) [3 points] . . . 4

Problem 3 (e) [5 points] . . . 4

Problem 4 . . . 5

Problem 4 (a) [3 points] . . . 5

Problem 4 (b) [5 points] . . . 5

Problem 4 (c) [3 points] . . . 5

Exam Rules and Instructions

(1) You must work on this exam individually. Any communication with others about this exam in any form is considered cheating. Should you have any questions, please send an email to mori@fullerton.edu. Do not email from the Canvas!

(2) Copying solutions from an Internet source or any other source would be considered plagiarism and will be dealt with according to university policy.

(3) You are not allowed to distribute the questions on this exam in any form or share the questions with anyone during the exam or anytime after the due-date. This includes posting questions to the Internet.

(4) Your solutions must be typewritten. You must submit a single pdf file that would include your solutions, R codes, and R outputs. Additionally submit your Rmarkdown file.

(5) Your solutions must appear in the order of the problem numbers. If you don't know the answer to a problem, write the problem number, and leave a blank space.

(6) In multi-part problems where one part depends on another, you may submit your solution for grading **no later than midnight of Sunday May 12th**. If your answer is correct, you will get **deducted one point or 10% of the total grade for the part, whichever is greater**. But if your solution is incorrect, you will lose the points for that part, and I will provide you with the correct answer (**in most cases partial credit will be given**).

(7) I reserve the right to interview you about the exam after I grade your exam.

(8) Submit your solution to Canvas.

(9) A total of 50 points is possible.

Problem 1 (a) [5 points]

Use importance sampling with standardized weights to estimate $\theta = Var(X)$, where X is a random variable with a density proportional to $f(x) = \sin(\sin(x))$ with support $0 \leq x \leq \pi/2$. Use 10,000 random values form the standard normal distribution (the proposal density) using the following specific R code:

```r
set.seed(2024)
x <- rnorm(100000)
```

Use stamnardized weights for your resampling. Resample 5000 values, and use the resampled values to estimate $\theta = Var(X)$. Include your R code and print the estimate of $Var(X)$ with at least 6 decimal places.

Problem 1(b) [3 Points]

Suppose that you want to obtain an estimate of the standard error for your estimate in part (a). Explain mathematically what you need to compute, and what quantities would be difficult to compute and why. [2 points for what you need to compute, and 1 point for pointing out possible difficulties.]

Problem 2(a) [2 points]

Consider the random variable Y with density $g(y) = \sin(y)$, with support in the interval $[0, \pi/2]$. Explain how you can use inverse transformation method to generate values from this random variable. Describe the method specifically with the functions given in this problem. You won't get any points for describing the method in general terms. No R code needed here.

Problem 2(b) [3 points]

Write an R code to generate 10,000 values from the random variable Y using the inverse transformation method that you described in part (a). Use seed 2024. Draw a histogram of the values you obtain, superimposed by a smoothed density, using the following R code [Y is a vector of generated values in the following code]:

```r
hist(Y, freq = FALSE, breaks = 50)
curve(sin(x), 0, pi/2, type = "l", add = TRUE, col="red")
```

Show your R code. Hint: On one of the boundaries of the distribution, the generated values might not look quite right due to rounding error. Ignore issue due to rounding errors.

Problem 2 (c) [5 points]

Write an R code to implement the accept-reject algorithm that uses the random values of Y that you generated in part (b), to generate random values from the random variable X with the density $f(x) = 8x/\pi^2$ with support in $[0, pi/2]$. To obtain an envelope obtain the optimal value of α such that $g(y)/ \geq f(x)$. .Show how you find α. Also show how many of the 10,000 values that you generated are accepted. Note: There is no need to use loops. Implement your code in vectorized form. Show your R code. Use the following code to show your generated values superimposed by the density $f(x)$.

```r
hist(X, freq = FALSE, breaks = 50)
curve(8*x/pi^2, 0, pi/2, type = "l", add = TRUE, col="red")
```

Problem 3

Consider the trivariate random vector (Y_1, Y_2, Y_3) with the density

$$f(y_1, y_2, y_3) = c \times \exp\left\{-(y_1 + y_2 + y_3 + y_1y_2 + 2y_2y_3 + 4y_1y_3)\right\}; \quad y_1 \geq 0, y_2 \geq 0, y_3 \geq 0.$$

where $c > 0$ is the normalizing constant so that the density integrates to 1.

Problem 3(a) [3 points]

Obtain the distribution of conditional densities of $Y_i|Y_j, Y_k$ for all possible distinct values of$i \neq j \neq k$. Show your work.

Problem 3 (b) [5 points]

Implement the Gibbs sampling algorithm to generate values from the joint density of (Y_1, Y_2, Y_3). Specifically, run a chain with 10,000 points, and burn the first 1000 initially generated values. Then, print the approximation for the mean and covariance matrix of (Y_1, Y_2, Y_3) based on your generated sample. Also, plot the histogram of the marginals distribution of Y_1 only superimposed by a smoothed density curve. Set `breaks = 50` for the histogram, and use the `density()` function for the density curve to be superimposed.

Problem 3 (c) [5 points]

Apply the Metropolis-Hastings algorithm to generate values from (Y_1, Y_2, Y_3) using a chain with 50,000 points. Set the proposal density to the trivariate normal density with covariance matrix equal to two times the identity matrix (i.e., $2 * I$). Your code should implement the following:

- Use the `mvrnorm()` function in the package `MASS` to generate multivariate normal values.
- Begin your simulation with seed 2024.
- Start with the initial value $y_1 = y_2 = y_3 = 1$.
- Your code should compute the acceptance rate and print it at the end.
- Burn the first 5000 points from the chain, and plot the histogram of the marginals of Y_1, Y_2, and Y_3, superimposed by a smoothed density curve. Set `breaks = 50` for the histogram, and use the `density()` function for the density curve to be superimposed.

Problem 3 (d) [3 points]

Use the library `coda` to print the summary of the chain, and the trace plots for the chain (use the function `traceplot()`. Explain why the chain is not mixing well.

Problem 3 (e) [5 points]

Repeat Problem 3(c), except this time adjust the covariance matrix for the proposal density so that the chain mixes well and in particular the chain has acceptance rate somewhere between 25% to 40%. Again use the `coda` package to print the summary of the chain, the trace plots for the chain. Also print the acceptance rate.

Problem 4

In this problem, we revisit the example that we worked on in class about Kevin Mitchell's batting average, and we imagine that it is 1988 now! Assuming that the outcomes of Kevin Mitchell's hits are independent, the number of his home runs, call it X, has the following distribution:

$$X \sim Binomial(N, \theta),$$

where N is the number of times he is at bat, and θ is his batting average (the probability that he makes a home-run each time he is at bat). We assume that θ has a beta prior distribution with parameters $\alpha = 2$ and $\beta = 5$. Using the Bayesian framework, our goals is to obtain a distribution for θ, his probability of making a home run. Assume that we have observed him in his first 10 games of the season where he was at bat 44 times, and he had made 4 home runs. What will be different than what we did in class is that we will use a different prior for θ. Specifically, we use the following shifted and truncated double exponential prior:

$$f(\theta) = \frac{50}{1 - (e^{-95} + e^{-5})/2} \exp\{-100 \times |\theta - 0.05|\} \quad 0 \le \theta \le 1.$$

Problem 4 (a) [3 points]

Use the *rexp*() function in R to generate 10,000 random values from the prior distribution of θ. Use seed 2024. Then, plot the histogram of the generated values, superimposed by a smoothed density curve. Set `breaks = 50` for the histogram, and use the `density()` function for the density curve to be superimposed. The only functions allowed for generating the values are `rexp()`, and one of `sample()` or `runif`. You will not get credit if you use a packaged function to generate from the prior. [Hint: Generate from exponential with appropriate rate parameter, and then shift and truncate the values. Generate more values than needed (around 11000), and then truncate the values to obtain 10,000 values.]

Problem 4 (b) [5 points]

Use the 10,000 values that you generated from the prior in part (a) in a Sampling Importance Resampling (SIR) algorithm to draw 5000 values from the posterior for θ. Insert your R code below. Also, draw a frequency histogram (not a density histogram) of the posterior distribution and insert it below. Use the option `breaks = 100` for your histogram.

Problem 4 (c) [3 points]

Using the 5000 values that you generated in part (b) approximate the probability that Kevin Mitchell makes 4 home runs when he is at bat 50 times. Insert your R code and your final solution below.