Memorial University of Newfoundland
Department of Mathematics and Statistics

DSCI 6607 – Programmatic Data Analysis Using Python and R – Fall 2024

Instructor: Dr. Armin Hatefi

Thursday, December 12, 2024

Final Exam

Student Number: ______________

Family Name: ________________ **First Name:** ________________

Instructions:

2. There are 5 questions (100 points).
3. Marks are shown in brackets.
4. Show all your mathematical and computational work.
7. This exam has 4 pages, including this page.

1. Consider generating random observations from the following distribution

$$f(x) = \frac{1}{c(1+x)^{3/2}}, \quad x > 0$$

a) Find constant c such that $f(x)$ is a legitimate probability density function (pdf).

b) Find the cumulative density function (cdf) corresponding to the pdf $f(x)$?

c) Use part (b), and apply the inverse cdf method to generate data from the distribution. Show you mathematical calculations.

d) Use **python**, and write a function which takes $N = 2000$ and generates N observations from the distribution.

e) Plot the histogram of the generated data and plot the curve of the $f(x)$ on top of them. Explain if your simulated data follow the corresponding distribution or not. **[20 points]**

2. In this question, we plan to develop a maximum likelihood estimation using **R**. Let X follow probability density function $f(x;\theta)$, where θ denotes the unknown parameter(s). The maximum likelihood function of θ based on n observations $\mathbf{x} = (x_1, \dots, x_n)$ is calculated by

$$L(\theta|\mathbf{x}) = \prod_{i=1}^{n} f(x_i;\theta)$$

Hence the maximum likelihood estimate of θ is given by

$$\widehat{\theta} = \arg\max_{\theta} \ell(\theta|\mathbf{x})$$

where $\ell(\theta|\mathbf{x})$ is the logarithm of the $L(\theta|\mathbf{x})$.

a) Consider X follows normal distribution with pdf

$$f(x;\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right),$$

find the loglikelihood function of $\theta = (\mu, \sigma^2)$ based on n observations.

b) Write an R function called 'loglike.function' which takes two arguments, including data and params (as a vector of two arguments). Then the function returns the loglikelihood function. Then, evalaute the log-likelihood function based on params as ($\mu = 5, \sigma^2 = 1$) and data

$$5.1, 3.2, 5.8, 6.1, 6.2, 6.8, 5.2, 5.01, 6.5, 6.8$$

c) Write function(s) which takes data and returns the MLE of parameters μ and σ^2 by maximizing the loglikelihood function using the numerical optimization function ‘optim‘ in R.

d) Recall the above function, initialize the optim function by starting point (mean(data), var(data)) and find the MLE of the parameters based on the above data in part (b). **[20 points]**

3. We want to implement an introduction to gradient descent optimisation with Python in this question. Consider the quadratic equation **[20 points]**

$$f(x) = ax^2 + bx + c$$

a) Write a python function, ‘quadratic.function‘ that takes a,b,c as three inputs and computes the function.

b) Write the gradient function ‘grad.function‘ which computes

$$f'(x) = 2ax + b$$

c) Write a functional program to find the minimum value of the $f(x)$ using the numerical optimization through a customized gradient descent. To do that, write a function which takes a,b,c and ‘learning.rate‘ (default 0.1), ‘tol‘ (default 10^{-5}) and ‘max.iter‘ (default 1000). The function iterates, updating the gradient using ‘grad.function‘ and updating x by ‘$x - learning.rate * grad$‘. These two iterative steps are alternated until either the gradient becomes less than the ‘tol‘ or the number of iterations reaches the ‘max.iter‘.

d) Test your program with the values a=3, b=-12, c=9.

e) Explain your solution and whether your numerical optimization converged or not. Why?

4. Import the ‘diamond‘ data set from the repository. Use R to analyze and explain how the price of a diamond varies by clarity and cut. We plan to use the ggplot2 package to answer this question.

a) Focus on the diamond observations whose carat is greater than 0.5 and whose price is less than 5000.

b) Use ggplot and study the relationship between the two variables of price and carat from part (a) data. Explain which graphical display should be used and why.

c) Facet the plot by ‘cut‘ so that each facet corresponds to a different diamond cut.

d) Color the points by ‘clarity‘. Carefully add the labels for the x-axis, y-axis, title and legend. **[20 points]**

5. Import the ‘sales‘ data set from the repository. Use Python to analyze and explain sales data, which contains monthly sales information for different regions.

a) Load the dataset as a data frame. Convert the ‘Date‘ column to a datetime object and extract the year.

b) Group the data by ‘Region‘ and calculate the total sales for each year.

c) Create a line plot to show the yearly sales trend for each region. Then, add appropriate labels for the x-axis and y-axis and a title for the plot.

d) Highlight the region with the highest total sales across all years in the plot using a thicker line width or a different line style. [**20 points**]

Good luck!