Examination Computational Statistics

Linköpings Universitet, IDA, Statistik

Course:	732A90 Computational Statistics
Date:	2023/05/12, 8–13
Teacher:	Krzysztof Bartoszek
Allowed aids:	Printed books, 100 page computer document,
Provided aids:	material in the zip file **exam_material_732A90.zip**
Grades:	A= $[18-20]$ points
	B= $[16-18)$ points
	C= $[14-16)$ points
	D= $[12-14)$ points
	E= $[10-12)$ points
	F= $[0-10)$ points
Instructions:	Provide a detailed report that includes plots, conclusions and interpretations. If you are unable to include a plot in your solution file clearly indicate the section of R code that generates it. Give motivated answers to the questions. If an answer is not motivated, the points are reduced. **Provide all necessary codes in an additional script text file that is directly runnable. Provide example calls to your implementations.** In a number of questions you are asked to do plots. Make sure that they are informative, have correctly labelled axes, informative axes limits and are correctly described. Points may be deducted for poorly done graphs. Name your solution files as: **[your exam account id]_[own file description].[format]** If you have problems with creating a pdf you may submit your solutions in text files with unambiguous references to graphics and code that are saved in separate files. There are **TWO** assignments (with sub–questions) to solve. Provide a separate solution file for each assignment. Include all R code that was used to obtain your answers in your solution files. Make sure it is clear which code section corresponds to which question. If you also need to provide some hand–written derivations please number each page according to the pattern: Question number . page in question number i.e. Q1.1, Q1.2, Q1.3,. . ., Q2.1, Q2.2, . . ., Q3.1, ...

NOTE: If you fail to do a part on which subsequent question(s) depend on describe (maybe using dummy data, partial code e.t.c.) how you would do them given you had done that part. You *might* be eligible for partial points.

Assignment 1 (10p)

Source of question Dominik Burek, Delta 1 (584), 2023, p. 5 `www.deltami.edu.pl`.
Consider the real numbers $x_1, x_2, \ldots, x_{2023}$ that have the property (P) that the sequence

$$\frac{x_1 + x_2}{2}, \frac{x_2 + x_3}{2}, \ldots, \frac{x_{2023} + x_1}{2}$$

is a permutation of the $x_1, x_2, \ldots, x_{2023}$ sequence.

Implement a genetic algorithm that investigates if only $x_1 = x_2 = \ldots = x_{2023}$ can satisfy property (P). Provide plots, and summaries of results that support your statement concerning the sets of numbers that can satisfy property (P).

If running times are too large, you may reduce the number 2023 to a smaller one, but not smaller than 20.

You might find the function `setequal()` useful.

It can actually be mathematically shown that only numbers such that $x_1 = x_2 = \ldots = x_{2023}$ can satisfy property (P). A completely correct, formal, mathematical proof will also result in 10points for this question.

Assignment 2 (10p)

Source of question Dominik Burek, Delta 1 (584), 2023, p. 5 `www.deltami.edu.pl`.
Let p_n be the n–th prime number. The prime numbers $2, 3, 5, \ldots, p_n$ are divided into two sets and the product of the numbers in each set is calculated, resulting in values v_1 and v_2.

Question 2.1 (5p)

Estimate the probability that $v_1 + v_2$ is prime. Report on the standard deviation of the estimator. Investigate how your estimate of this probability and its standard deviation behaves for different values of n. Provide plots or tables. Remember that you need to randomly divide your prime numbers into the two sets. Take values of n at least up to 10000. However, if your running times are too large you can reduce this number, but then discuss what part of your code is causing the bottleneck.

Question 2.2 (5p)

It actually turns out that if $v_1 + v_2 < p_{n+1}^2$, then $v_1 + v_2$ will be prime. Compare the probability from Question 2.1 with the probability that a randomly sampled pair of numbers, $w_1, w_2 < p_{n+1}^2$, will be such that $w_1 + w_2 < p_{n+1}^2$. Do this for various values of n, at least up to 10000. Check also how often $w_1 + w_2$ is prime. How does this probability compare with the one from Question 2.1? Do not forget about the standard deviation. Provide plots or tables in your report.

You might find the functions `primes::generate_primes()` and `primes::is_prime()` useful.