

Probability and Statistics

Lab assignment 2: Probability, random variables, and limit theorems

1 General comments and instructions

- Complete solution will give you **3** points (working code with explanations + oral defense). Submission deadline **October 29, 2023, 23:59**.
- You have to submit both the **Rmd** source file and the **html** output (links to GitHub repositories are NOT accepted!)
- There is a starter R notebook file that you can use. Its purpose is to provide some standard **R** functions and plotting tools to simplify the coding part; you are encouraged to modify it as you want.
- **At the beginning of the notebook, provide a work-breakdown structure estimating efforts of each team member.**
- For each task, include:
 - problem formulation and discussion (what is a reasonable answer to discuss);
 - the corresponding **R** code with comments (usually it is just a couple of lines long);
 - the statistics obtained (like sample mean or anything else you use to complete the task) as well as histograms etc to illustrate your findings;
 - justification of your solution (e.g. refer to the corresponding theorems from probability theory);
 - conclusion (e.g. how reliable your answer is, does it agree with common sense expectations).
- The **team id number** referred to in tasks is the **two-digit** ordinal number of your team on the list. You **must** include the line `set.seed(**)` at the beginning of your code (with ****** being the **team id number**) to make your calculations reproducible. Also observe that the answers **do** depend on this **team id number**!
- Take into account that not complying with these instructions may result in point deduction regardless of whether or not your implementation is correct.

2 Assignment

Task 1. In this task, we discuss the $[7, 4]$ Hamming code and investigate its reliability. That coding system can correct single errors in the transmission of 4-bit messages and proceeds as follows:

- given a message $\mathbf{m} = (a_1 a_2 a_3 a_4)$, we first encode it to a 7-bit codeword $\mathbf{c} = \mathbf{m}G = (x_1 x_2 x_3 x_4 x_5 x_6 x_7)$, where G is a 4×7 *generator matrix*
- the codeword \mathbf{c} is transmitted, and \mathbf{r} is the received message
- \mathbf{r} is checked for errors by calculating the *syndrome vector* $\mathbf{z} := \mathbf{r}H$, for a 7×3 *parity-check matrix* H
- if a single error has occurred in \mathbf{r} , then the binary $\mathbf{z} = (z_1 z_2 z_3)$ identifies the wrong bit no. $z_1 + 2z_2 + 4z_3$; thus (000) shows there was no error (or more than one), while (110) means the third bit (or more than one) got corrupted
- if the error was identified, then we flip the corresponding bit in \mathbf{r} to get the corrected $\mathbf{r}^* = (r_1 r_2 r_3 r_4 r_5 r_6 r_7)$; the decoded message is then $\mathbf{m}^* := (r_3 r_5 r_6 r_7)$.

The *generator matrix* G and the *parity-check matrix* H are given by

$$G := \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, \quad H^\top := \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Assume that each bit in the transmission $\mathbf{c} \mapsto \mathbf{r}$ gets corrupted independently of the others with probability $p = \text{team id number}/100$. Your task is the following one.

1. Simulate the encoding-transmission-decoding process N times and find the estimate \hat{p} of the probability p^* of correct transmission of a single message \mathbf{m} . Comment why, for large N , \hat{p} is expected to be close to p^* .
2. By estimating the standard deviation of the corresponding indicator of success by the standard error of your sample and using the CLT, find the half-length ε of the *confidence* interval $(\hat{p} - \varepsilon, \hat{p} + \varepsilon)$, which contains the true value p^* with probability at least 0.95. What choice of N guarantees that $\varepsilon \leq 0.03$?
3. Draw the histogram of the number $k = 0, 1, 2, 3, 4$ of errors while transmitting a 4-digit binary message. Do you think the random variable that counts the number of wrong bits in a decoded message has one of the known distributions? Justify your answer.

Task 2. In this task, we discuss a real-life process that is well modelled by a Poisson distribution. As you remember, a Poisson random variable describes occurrences of rare events, i.e., counts the number of successes in a large number of independent random experiments. One of the typical examples is the **radioactive decay** process.

Consider a sample of radioactive element of mass m , which has a big *half-life period* T ; it is vitally important to know the probability that during a one second period, the number of nuclei decays will not exceed some critical level k . This probability can easily be estimated using the fact that, given the *activity* λ of the element (i.e., the probability that exactly one nucleus decays in one second) and the number N of atoms in the sample, the random number of decays within a second is well modelled by Poisson distribution with parameter $\mu := N\lambda$. Next, for the sample of mass m , the number of atoms is $N = \frac{m}{M} N_A$, where $N_A = 6 \times 10^{23}$ is the Avogadro constant, and M is the molar (atomic) mass of the element. The activity of the element, λ , is $\log(2)/T$, where T is measured in seconds.

Assume that a medical laboratory receives n samples of radioactive element ^{137}Cs (used in radiotherapy) with half-life period $T = 30.1$ years and mass $m = \text{team id number} \times 10^{-6}$ g each. Denote by X_1, X_2, \dots, X_n the **i.i.d.** **r.v.**'s counting the number of decays in sample i in one second.

1. Specify the parameter of the Poisson distribution of X_i (you'll need the atomic mass of **Cesium-137**)
2. Show that the distribution of the sample means of X_1, \dots, X_n gets very close to a normal one as n becomes large and identify that normal distribution. To this end,
 - simulate the realization x_1, x_2, \dots, x_n of the X_i and calculate the sample mean $s = \bar{x}$;
 - repeat this K times to get the sample $\mathbf{s} = (s_1, \dots, s_K)$ of means and form the empirical cumulative distribution function $\hat{F}_{\mathbf{s}}$ of \mathbf{s} ;
 - identify μ and σ^2 such that the **c.d.f.** F of $\mathcal{N}(\mu, \sigma^2)$ is close to the **e.c.d.f.** $\hat{F}_{\mathbf{s}}$ and plot both **c.d.f.**'s on one graph to visualize their proximity (use the proper scales!);
 - calculate the maximal difference between the two **c.d.f.**'s;
 - consider cases $n = 5$, $n = 10$, $n = 50$ and comment on the results.
3. Calculate the largest possible value of n , for which the total number of decays in one second is less than 8×10^8 with probability more than 0.95. To this end,
 - simulate the realization x_1, x_2, \dots, x_n of the X_i and calculate the sum $s = x_1 + \dots + x_n$;
 - repeat this K times to get the sample $\mathbf{s} = (s_1, \dots, s_K)$ of sums;
 - calculate the number of elements of the sample which are less than critical value (8×10^8) and calculate the empirical probability;
 - obtain the theoretical bound using Markov inequality, Chernoff bound and Central Limit Theorem, and compare the results.

Task 3. In this task, we use the Central Limit Theorem approximation for continuous random variables.

One of the devices to measure radioactivity level at a given location is the Geiger counter. When the radioactive level is almost constant, the time between two consecutive clicks of the Geiger counter is an exponentially distributed random variable with parameter $\nu_1 = \text{team id number} + 10$. Denote by X_k the random time between the $(k-1)^{\text{st}}$ and k^{th} click of the counter.

1. Show that the distribution of the sample means of X_1, X_2, \dots, X_n gets very close to a normal one (which one?) as n becomes large. To this end,
 - simulate the realizations x_1, x_2, \dots, x_n of the **r.v.** X_i and calculate the sample mean $s = \bar{x}$;

- repeat this K times to get the sample $\mathbf{s} = (s_1, \dots, s_K)$ of means and then the *empirical cumulative distribution function* $F_{\mathbf{s}}$ of \mathbf{s} ;
 - identify μ and σ^2 such that the **c.d.f.** of $\mathcal{N}(\mu, \sigma^2)$ is close to the **e.c.d.f.** $F_{\mathbf{s}}$ of and plot both **c.d.f.**'s on one graph to visualize their proximity;
 - calculate the maximal difference between the two **c.d.f.**'s;
 - consider cases $n = 5$, $n = 10$, $n = 50$ and comment on the results.
2. The place can be considered safe when the number of clicks in one minute does not exceed 100. It is known that the parameter ν is proportional to the number N of the radioactive samples, i.e., $\nu = N\nu_1$, where ν_1 is the parameter for one sample. Determine the maximal number of radioactive samples that can be stored in that place so that, with probability 0.95, the place is identified as safe.

To do this,

- (a) express the event of interest in terms of the **r.v.** $S := X_1 + \dots + X_{100}$;
- (b) obtain the theoretical bounds on N using the Markov inequality, Chernoff bound and Central Limit Theorem and compare the results;
- (c) with the predicted N and thus ν , simulate the realization x_1, x_2, \dots, x_{100} of the X_i and of the sum $S = X_1 + \dots + X_{100}$;
- (d) repeat this K times to get the sample $\mathbf{s} = (s_1, \dots, s_K)$ of total times until the 100th click;
- (e) estimate the probability that the location is identified as safe and compare to the desired level 0.95

Task 4. This task consists of two parts:

1. In this part, we discuss independence of random variables and its moments: expectation and variance.
 - (a) Suppose we have a random variable X . Explain why $\mathbb{E}(\frac{1}{X}) \neq \frac{1}{\mathbb{E}(X)}$;
 - (b) Let $X \sim \mathcal{N}(\mu, \sigma^2)$ with $\mu = 5$ and $\sigma^2 = 10$. Simulate realizations x_1, x_2, \dots, x_{100} of X and y_1, y_2, \dots, y_{100} of $Y := \frac{1}{X}$ to calculate the values of $\frac{1}{\bar{X}}$ and \bar{Y} . Comment on the received results;
 - (c) Let X and Y be exponentially distributed r.v.'s with parameter $\lambda = 2$. Set $Z := \log X + 5$. Plot the Quantile-Quantile plot and scatterplot of X and Y . Plot the Quantile-Quantile plot and scatterplot of X and Z . Explain the results. Comment on the difference of relations between the pairs of random variables. Which pair of r.v.'s is dependent and which one is similar?
2. You toss a fair coin three times and a random variable X records how many times the coin shows Heads. You convince your friend that they should play a game with the following payoff: every round (equivalent to three coin tosses) will cost £1. They will receive £0.5 for every coin showing Heads. What is the expected value and the variance of the random variable $Y := 0.5X - 1$?

To answer this,

- (a) Explain what type of random variable is X :
 - Normally distributed
 - Binomially distributed
 - Poisson distributed
 - Uniformly distributed
- (b) What are the expected value and variance of X ? Simulate realizations x_1, x_2, \dots, x_{100} of X to calculate the values of sample mean \bar{X} and sample variance $s^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}$. Comment on the results;
- (c) What are the expected value and variance of Y ? Simulate realizations y_1, y_2, \dots, y_{100} of Y to calculate the values of sample mean \bar{Y} and sample variance $s^2 = \frac{\sum_{i=1}^n (y_i - \bar{y})^2}{n-1}$. Comment on the results;

3 General instructions on R implementation

1. You can generate an $n \times K$ array of the data and then form a sample of K means by e.g. using `colMeans(matrix(rexp(n*K, rate = 1/nu), nrow=n))` for exponential distribution $\mathcal{E}(\nu)$ (note that the `rate` is the reciprocal of the parameter ν)
2. In **R**, the e.c.d.f. F_S of the data vector **s** is just one line `Fs <- ecdf(s)`
3. K must be at least 1000 to generate reliable **e.c.d.f.**'s of sample means
4. `pnorm` generates the **c.d.f.** F_Z of the normal distribution; see RStudio help.
5. To calculate the maximal difference between the two **c.d.f.**'s, take a grid of 200 points t_j within 3σ of μ and return $\max_{t_j}\{|F_S(t_j) - F_Z(t_j)|\}$; if **t** is the vector of t_j 's, then `max(abs(Fs(t) - Fz(t)))` does the job!
6. There is another **Rmd** file provided where you can find similar implementations