1 Block 1

TOPICS

Variables and constants

Basic instructions: variable/constant declaration, assignment, reading, printing

Math operations

Control structures: sequence

1.1

Write a program that reads two numbers a and b and computes the sum z between them.

1.2

Consider $X \sim \mathcal{N}(x; \mu = 2.5, \sigma = 2.1)$. Write a program that computes $\mathbb{P}(X \leq x_0)$ with $x_0 \in \mathbb{R}$. Note that

$$\mathbb{P}(X \leq x_0) = \int_{-\infty}^{x_0} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right) dx$$

1.3

Consider $X \sim \mathcal{N}(x; \mu = 2.5, \sigma = 2.1)$. Write a program that computes $\mathbb{P}(X \geq x_0)$ with $x_0 \in \mathbb{R}$. Note that

$$\mathbb{P}(X \geq x_0) = \int_{x_0}^{\infty} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{1}{2}\left(\frac{(x-\mu)}{\sigma}\right)^2\right) dx$$

1.4

Consider $X \sim \mathcal{E}xp(x; \lambda = 1.1)$. Write a program that (i) plots the probability density function of X, (ii) computes $\mathbb{P}(X \geq x_0)$ with $x_0 \in \mathbb{R}^+$, (iii) represents graphically the quantile x_0 w.r.t. the density function.

1.5

Consider $X \sim \mathcal{N}(x; \mu = 0, \sigma = 2.5)$. Write a program that (i) plots the probability density function of X, (ii) computes $\mathbb{P}(x_1 \leq X \leq x_2)$ with $(x_1, x_2) \in \mathbb{R}^2$, (iii) represents graphically the quantiles x_1 and x_2 w.r.t. the density function.

1.6

Write an algorithm that reads two strings s_1 and s_2 as input and returns a new string s formed by concatenating s_1 and s_2.

1.7

Consider $X \sim \mathcal{P}oi(x; \lambda = 2)$. Write an algorithm that (i) plots the probability density function of X, (ii) computes $\mathbb{P}(X = x_0)$ with $x_0 \in \mathbb{N}$, (iii) represents graphically the quantile x_0 w.r.t. the density function.

1.8

Consider $X \sim \mathcal{N}(x; \mu = 0, \sigma = 1.5)$, $Y \sim \chi^2(y; k = 2)$ and $x_0 \in \mathbb{R}^+$. Write an algorithm that reads x_0, computes and prints out the following quantity:

$$r = \frac{\mathbb{P}(X \geq x_0)}{\mathbb{P}(Y \leq x_0)} \mathbb{P}(X \geq x_0)^2$$

1.9

(*cont. Exercise 1.8*) Write an algorithm that (i) reads five positive real numbers $x_1, \ldots, x_5$ such that $x_1 < x_2 < \ldots < x_5$, (ii) computes for each input the quantity r, (iii) plots r as a function of $\min\left(\mathbb{P}(X \geq x_i), \mathbb{P}(Y \leq x_i)\right)\ \ i = 1, \ldots, 5$.

1.10

Consider $Y \sim \mathcal{N}(y; \mu = 2.3, \sigma = 1)$ and $p_0 \in [0, 1]$. Write an algorithm that computes and prints out the quantile $y_0 = \inf\left\{x \in \mathbb{R} : \mathbb{P}(Y \leq y) = p_0\right\}$. Note that in this case the *quantile function* of a probability density function needs to be used.

2 Block 2

Topics

Control structures: one-way selection, multi-way selection, for-loop

Arrays: vectors

Functions and subroutines

2.1

Write an algorithm that reads two real numbers x and y and computes the minimum between them.

2.2

Write an algorithm that reads two real numbers x and y and executes the following instructions:

1. if $\lfloor \min(x, y) \rfloor$ is an even number then evaluates the expression $z = x^{|y|}$
2. if $\lfloor \min(x, y) \rfloor$ is an odd number then evaluates the expression $z = x^{|y-x|}$
3. outputs the value of z

Note that $\lfloor x \rfloor$ can be computed using the primitive function `floor(..)`, the absolute value $|x|$ can be computed by means of the primitive `abs(..)`, whereas to verify whether a number is odd or even, one can use the command `x%%2` which gives as output `True` if the number is even.

2.3

Write an algorithm that reads two real numbers x and y and computes the following expression:

$$z = \max\left(|x|, |y|\right) - \min(x, y) + (x - y)^2$$

If $z < 0$ then ask for a third number q and print z only if $z > q$.

2.4

Write an algorithms that reads two intervals $i_1 = [a, b]$ and $i_2 = [c, d]$ and evaluates the following instructions:

- if $i_1 \subset i_2$ then computes $h = (\max(a, b) - \min(a, b)) \frac{1}{j}$, where $j = \frac{a+b}{2}$
- if $i_1 \not\subset i_2$ then computes $h = (\max(a, b) - \min(a, b)) \frac{1}{j}$, where $j = \frac{a}{|b|+\epsilon}$ and $\epsilon = \max(a, b, c, d)^{\frac{1}{3}}$
- if $i_1 = i_2$ then computes $h = (\max(a, b) - \min(a, b))$
- outputs h

2.5

Rewrite Exercise 1.9 using vectors and for-loops. In this case, $x_i \sim \mathcal{U}(x; 0.5, 5),\ i = 1, \ldots, 5$.

2.6

Write an algorithm that reads a $I \times 1$ vector of real numbers $\mathbf{x} = (x_1, \dots, x_i, \dots, x_I)$ and gives as output the vector $\mathbf{y} = (x_I, \dots, x_i, \dots, x_1)$. Note $\mathbf{x}$ can be generated uniformly on $[-2, 2]$.

2.7

Write an algorithm that reads two natural numbers n_a and n_b and determines the vector $\mathbf{x}$ containing the sequence of numbers included between n_a and n_b.

2.8

Write an algorithm that given the array $\mathbf{x}_{n\times 1}$ of real positive numbers, evaluates the following expressions:

1. $x^* = \sum_{i=1}^n x_i$
2. $\overline{x} = \frac{1}{n}\sum_{i=1}^n x_i$
3. $x^\dagger = x^*/(\max(\mathbf{x}) - \min(\mathbf{x}))$
4. $z = (x^* - 2x^\dagger)$
5. write as output z

Note that x_i can be generated uniformly in a given interval.

2.9

Write an algorithm that given a vector of integers $\mathbf{n}_{I\times 1}$, computes the vector $\mathbf{f}_{I\times 1}$ containing the frequencies of the elements of $\mathbf{n}_{I\times 1}$. Note: To compute frequencies, write a function that counts how many times the elements of $\mathbf{n}$ occur. The vector $\mathbf{x}$ can be generated via Poisson random numbers generator.

2.10

(*cont. Exercise 2.9*) Write an algorithm that given as input the vector of frequencies $\mathbf{f}_{I\times 1}$, computes the vector $\mathbf{s}_{I\times 1}$ where the generic element s_i is defined as follows:

$$s_i = s_{i-1} + f_i$$

Note that $s_1 = f_1$ by definition.

2.11

Consider $X \sim \mathcal{N}(x; \mu = -2, \sigma = 1.5)$ and $Y \sim \mathcal{N}(y; \mu = 2, \sigma = 1.5)$. Write an algorithm that plots the densities $f_X(x; \mu_x, \sigma_x)$ and $f_Y(x; \mu_y, \sigma_y)$ and computes the degree of overlapping between them:

$$p_{XY} = \int_{-\infty}^{+\infty} \min_z \left(f_X(z; \mu_x, \sigma_x), f_Y(z; \mu_y, \sigma_y)\right)$$

2.12

Consider two vectors $\mathbf{x}_{n\times 1}$ and $\mathbf{y}_{n\times 1}$ where $x_i \sim \mathcal{N}(x; \mu = 0, \sigma = 2.0)$ and $y_i \sim \chi^2(y; k = 6)$. Write an algorithm that (i) plots the distribution of the input vectors, (ii) produces the scatter plot of the two vectors, and (iii) computes the following quantity:

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2}}$$

2.13

Consider $X \sim \mathcal{U}(x; a, b)$ with $a = 0.5$ and $b = 3.4$. Write an algorithm that

1. Generates a vector $\mathbf{x}_{n\times 1}$ containing random samples from X with $n = 100$
2. Computes and prints out the mean and the variance of $\mathbf{x}$
3. Compares the empirical mean and variance with the true mean and variance[1] and indicates whether these empirical moments are greater or less then the true moments
4. Computes the probability $\mathbb{P}(X \leq 1.7)$ using $\mathbf{x}$ and compares it with the true probability calculated using the cumulative density function of the Uniform distribution
5. Computes $U = 1/(X + 1)$
6. Compares the histograms of X and U graphically

2.14

Write an algorithm that (i) creates $n = 10000$ random m-grams of $m = 5$ letters from the English alphabet (26 letters) and (ii) computes the probability that the generated m-grams contain the word "nn". Note: Consider the case where letters are uniformly generated.

2.15

(Cont. Exercise 2.14) Compute the probability distribution that uniformly generated m-grams contain two consecutive letters (e.g., "aa","bb",...,"zz").

[1] `https://en.wikipedia.org/wiki/Continuous_uniform_distribution`

3 Block 3

TOPICS

Control structures: while-loop

Arrays: matrices

Lists, dataframes

3.1

Write an algorithm that computes the square root of a number $s > 0$ using the Heron's approximation:

$$\sqrt{s} = \lim_{n\to\infty} x_n$$

where the generic element x_n is defined as:

$$x_n = \frac{1}{2}\left(x_{n-1} + \frac{s}{x_{n-1}}\right)$$

with $x_1 = 1.5$. The iterative approximation must be executed until the condition $(x_n - x_{n-1}) < \epsilon$ is met, with ϵ being a user's defined small real number (e.g., $\epsilon = 0.001$).

3.2

Write an algorithm that given a $N \times M$ matrix $\mathbf{X}$ with $N = M$, computes the following quantities:

1. $q = \sum_{i=1}^{N} \mathrm{diag}(\mathbf{X})_i + \max(\mathbf{X})$
2. $p = \prod_{i=1}^{N} \mathrm{diag}(\mathbf{X})_i + \min(\mathbf{X})$
3. if $q > p$ outputs p, otherwise outputs q

Note that elements of X can be uniformly generated random numbers in a given interval.

3.3

Write an algorithm that reads two numbers $N > 0$ and K, then finds a square matrix $\mathbf{A}_{N\times N}$ such that the sum of elements in every row and column is equal to K. Note: (a) A can be populated using uniformly generated discrete numbers and (b) K should be large enough.

3.4

Write an algorithm that given a matrix of reals $\mathbf{X}_{I\times I}$, computes the following quantities:

1. $a = \sum_{i=1}^{I}\sum_{j=1}^{J} \mathbf{X}_{i,j}$ with $i \neq j$
2. $b = \sum_{l\in\mathcal{L}}\sum_{h\in\mathcal{H}} \mathbf{X}_{l,h}$ with $\mathcal{L}$ being the set of *even* indices and $\mathcal{H}$ the set of *odd* indices
3. computes $q = a + b$ if $a > b$, otherwise $q = \frac{1}{2}(a-b)^2$
4. outputs q

3.5

Write an algorithm that given a matrix of reals $\mathbf{X}_{N\times N}$ and an integer N, sorts $\mathrm{diag}(\mathbf{X})$ in ascending order.

3.6

Write an algorithm that (i) finds a matrix $\mathbf{Y}_{n\times n}$ such that $\sum_{i=1}^{n} \text{diag}(\mathbf{Y})_i > \frac{1}{2}(a+b)$ with $Y_{ij} \sim \mathcal{U}(y; a, b)$, $a < b$, (ii) computes the sum of the elements of $\mathbf{Y}$ which are less then the quantity $b - a$.

3.7

Write an algorithm that

1. finds $K > 0$ matrices $\mathbf{Y}_{n\times n}$ such that $\sum_{i<j} Y_{ij} > \frac{1}{2}(a+b)$ with $Y_{ij} \sim \mathcal{U}(y; a, b)$, $a < b$
2. saves the K matrices into a list of K elements
3. computes $\mathbb{P}(\sum_{i=1}^{n} \text{diag}(\mathbf{Y}^{(k)})_i > (\frac{a+b}{2})^2)$

3.8

Write an algorithm that given a matrix $\mathbf{X}_{I\times J}$, (i) computes the global maximum c columnwise, (ii) computes the global minimum r rowwise, (iii) prints out 0 if $c > r$, 1 otherwise.

3.9

Write an algorithm that (i) finds a matrix $\mathbf{X}_{I\times J}$ ($I > J$) such that $s = \left(\sum_{j=1}^{J} s_j\right) < \epsilon$, ($\epsilon = 1e^{-03}$) with $s_j = \frac{1}{I}\sum_{i=1}^{I} x_{ij}$ for each $j = 1, \ldots, J$ (i.e., column-wise mean), (ii) counts how many times q such constrain is not satisfied.

3.10

Write an algorithm that

1. generates a vector $\mathbf{x}_{n\times 1}$ with $x_i \sim \mathcal{P}oi(x; \lambda = 1.25)$
2. generates a vector $\mathbf{s}_{n\times 1}$ of uniformly distributed letters (consider the 26 letters of the English alphabet)
3. generates a vector $\mathbf{b}_{n\times 1}$ with $b_i \sim \mathcal{B}ern(b; \pi = 0.5)$
4. saves $\mathbf{x}$, $\mathbf{s}$, and $\mathbf{b}$ into a new dataframe $\mathcal{D}$
5. computes $\frac{1}{n}\sum_{i=1}^{n} x_i$ conditioned on the value $b = 0$
6. computes $\frac{1}{n}\sum_{i=1}^{n} x_i^2$ conditioned on the value $b = 0$ and $s = \text{a}$
7. computes $\frac{1}{2n}\sum_{i=1}^{n} x_i$ conditioned on the value $b = 1$ and $s \neq \text{c}$
8. compares $\mathbf{x}$ w.r.t. the levels of $\mathbf{b}$ graphically (i.e., boxplot)
9. compares $\mathbf{x}$ w.r.t. $s \neq \text{c}$ and $s \neq \text{d}$ graphically (i.e., boxplot)

4 Block 4

4.1

Write an algorithm that approximates and plots the distribution of the following random variable:

$$Z = \frac{Y - \lambda}{\sqrt{\lambda}}$$

where $Y \sim \mathcal{P}oi(y; \lambda)$ and $\lambda > 0$. Note that (i) a Monte Carlo based solution should be used with B large enough and (ii) several values of λ should be considered.

4.2

Write a Monte Carlo algorithm that computes the following integral: $\int_a^b x^4 \sin x \; dx$ with $a = 0$ and $b = 3$.

4.3

Write a Monte Carlo algorithm that computes the following integral: $\int_1^3 \int_1^4 \cos(x^2 \sin y) \; dx \; dy$.

4.4

Write a Monte Carlo algorithm that computes the following integral: $\int_0^3 \int_0^1 \cos(x - y) \; dx \; dy$.

4.5

Write a Monte Carlo algorithm that (i) estimates the mean $\mu \in \mathbb{R}^3$ of the model $\mathcal{N}_3(\mathbf{x}; \boldsymbol{\mu}_{3\times 1}, \boldsymbol{\Sigma}_{3\times 3})$ given a sample of iid data $\mathbf{Y}_{n\times 3}$ and (ii) plots the results of the Monte Carlo procedure. Note that data should be generated in advance.

4.6

Write a Monte Carlo algorithm that (i) estimates the parameter $\pi \in [0, 1]$ of the model $\mathcal{B}ern(y; \pi)$ given a sample of iid data $\mathbf{y}_{n\times 1}$ and (ii) plots the results of the Monte Carlo procedure. To simplify the procedure, the parameter π can be transformed via $\pi^* = \pi/(1 - \pi)$ so that $\pi^* \in \mathbb{R}$ and $\pi = \text{plogis}(\pi^*)$. Note that data should be generated in advance.

4.7

Write a Monte Carlo algorithm that (i) estimates the parameter $\lambda \in [0, +\infty)$ of the model $\mathcal{E}xp(y; \lambda)$ given a sample of iid data $\mathbf{y}_{n\times 1}$ and (ii) plots the results of the Monte Carlo procedure. Note that (a) data should be generated in advance and (b) a suitable transformation $g : [0, +\infty) \to \mathbb{R}$ should be used for λ.

4.8

Consider the statistical model $\mathbf{y}_{n\times 1} \overset{\text{iid}}{\sim} \mathcal{N}(y; \mu, \sigma)$ with $\boldsymbol{\theta} = \{\mu, \sigma\} \in \mathbb{R} \times \mathbb{R}^{+}$. Write a Monte Carlo algorithm to maximize the log-likelihood function $\mathcal{L}(\boldsymbol{\theta}; \mathbf{y})$ w.r.t. $\boldsymbol{\theta}$. Note that the log-likelihood surface can be easily explored by rejecting all those candidates which do not improve the maximization path.

4.9

Consider two vectors $\mathbf{x}_{n\times 1}$ and $\mathbf{y}_{n\times 1}$ of data and the linear correlation between them $r = \text{cor}(\mathbf{x}, \mathbf{y})$. Write an algorithm that computes the standard error and the $(1-\alpha)\%$ confidence interval for r via non-parametric bootstrap. Note that data should be generated in advance.

4.10

Consider the following linear models:

$$\begin{aligned} z_i &= \beta_0^{(z)} + x_i\beta_1^{(z)} + \epsilon_i^{(z)} \\ y_i &= \beta_0^{(y)} + x_i\beta_1^{(y)} + z_i\beta_2^{(y)} + \epsilon_i^{(y)} \end{aligned}$$

where $\epsilon_i^{(z)} \sim \mathcal{N}(\epsilon; 0, \sigma_{\epsilon_z})$, $\epsilon_i^{(y)} \sim \mathcal{N}(\epsilon; 0, \sigma_{\epsilon_y})$, and $\epsilon_i^{(z)} \perp\!\!\!\perp \epsilon_i^{(y)}$ (independence). Write an algorithm to compute standard error and $(1-\alpha)\%$ confidence interval for the parameter $\xi = \beta_1^{(z)}\beta_2^{(y)}$ (a.k.a. *mediation effect*) via non-parametric bootstrap. Note that (a) data $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ should be generated in advance and (b) parameters $\{\beta_0^{(z)}, \beta_1^{(z)}, \beta_0^{(y)}, \beta_1^{(y)}, \beta_2^{(y)}, \sigma_{\epsilon_z}, \sigma_{\epsilon_y}\}$ can be estimated using maximum likelihood or least squares estimators.