# MATS442 Stochastic simulation — Problems 3, 31.1.2020

The problems marked with **C** have a part which needs to be implemented in a computer programming language. Please return your implementation by email to ⟨santeri.j.karppinen(·at·)jyu.fi⟩.

**1.** Consider the following properties of min and max.

(a) Prove the following result: Let $x, y \in A \subset \mathbb{R}$ and let $f : A \to \mathbb{R}$ be non-decreasing. Then, $f(\min\{x, y\}) = \min\{f(x), f(y)\}$.

(b) Suppose $x, y \in A \subset \mathbb{R}$ and $f : A \to \mathbb{R}$ is non-increasing. What is the expression for $f(\min\{x, y\})$?

(c) What happens if you replace min with max in (a) and (b)?

**2.C** (Construction of IS confidence intervals.)

Consider Example 4.16, with $x_0 = 4$, that is, consider the importance sampling estimator

$$I_{p,q}^{(n)}(\mathbf{1}\,(\,\cdot\, \ge 4)) = \frac{1}{n4\sqrt{2\pi}} \sum_{k=1}^{n} \exp\bigg( - \frac{Y_i^2}{2} + 4(Y_i - 4)\bigg),$$

where $Y_i = \tilde{Y}_i + x_0$ where $(\tilde{Y}_i) \overset{\text{i.i.d.}}{\sim} \text{Exp}(4)$ (cf. Example 2.2).

(a) Implement a function which calculates the importance sampling estimator and the associated 95% confidence interval.

(b) Use antithetic variable technique to reduce the variance of the estimator, and construct a confidence interval for the antithetic estimator.

**3.C** Let $m \ge 1$ and suppose $p(1), \ldots, p(m) > 0$ with $\sum_{i=1}^{m} p(i) = 1$, that is, $p(\,\cdot\,)$ defines a p.m.f. on $\mathbb{X} = \{1, \ldots, m\}$, and let $f : \mathbb{X} \to \mathbb{R}$.

(a) The given Julia function `discrete_from_uniform` implements the method of Theorem 2.3 (finds the $K$ corresponding a $U$).

(b) Implement a stratified sampling approach for uniforms as in Example 5.10, and use the given Julia function to approximate

$$\mathbb{E}_p[f(X)] = \sum_{i=1}^{m} p(i) f(i).$$

(c) Try your function with $f(i) = i$ and $p(i) \propto i$ for $i \in \mathbb{X} = \{1, \ldots, 10\}$.

(d) Try what happens with $n = 55$ (repeat the experiment a few times). Can you explain why?

(* Optional extra): Think what is the *complexity* of your algorithm in (b). That is, what is the order of computer operations your algorithm needs to perform in terms of $m$ and $n$? Try to design an algorithm which is $O(m + n)$.

**4.C** Suppose that we are interested in estimating the mean and the second moment of the random variable $Y$, where $Y \mid X \sim N(X, X^2)$ (that is, $Y$ given $X = x$ is a Gaussian with mean $x$ and variance $x^2$) and where $X \sim \mathcal{U}(-1, 1)$.

(a) Use Monte Carlo directly: Produce realisations of $Y$ and compute the empirical mean and the second moment of $Y$.

(b) Use Rao-Blackwellisation and produce the same estimates of $Y$ using only realisations $(X_k) \overset{\text{i.i.d.}}{\sim} p$.
(Hint: Compute first $\mathbb{E}[Y \mid X = x]$ and $\mathbb{E}[Y^2 \mid X = x]$.)

(c) Compare the sample variance of the two estimators.

**5.** Complete the proof of Theorem 4.23 (ii): assuming $q(x) = 0 \implies p(x) = 0$, $\mathbb{E}_p[w(X)] < \infty$ and and $\bar{\sigma}^2_{p,q} := \mathbb{E}_p[w(X)\bar{f}^2(X)] < \infty$, where $\bar{f}(x) = f(x) - \mathbb{E}_p[f(X)]$, show that

$$v^{(n)}_{p,q} := \sum_{k=1}^{n} (W_k^{(n)})^2 \big[f(Y_k) - \hat{I}^{(n)}_{p,q}(f)\big]^2 \qquad \text{satisfies} \qquad n v^{(n)}_{p,q} \xrightarrow{n\to\infty} \bar{\sigma}^2_{p,q}.$$

Hints:

(i) Recall that $f(Y_k) - \hat{I}^{(n)}_{p,q}(f) = \bar{f}(Y_k) - \hat{I}^{(n)}_{p,q}(\bar{f})$.

(ii) Notice that (for $n$ large enough such that $\sum_{j=1}^{n} w_u(Y_j) > 0$)

$$n(W_k^{(n)})^2 = \frac{\frac{1}{n} w_u^2(Y_k)}{\left(\frac{1}{n}\sum_{j=1}^{n} w_u(Y_j)\right)^2}.$$

(iii) Using these, show that

$$n v^{(n)}_{p,q} = \frac{\left(\frac{1}{n}\sum_{k=1}^{n} w_u^2(Y_k)\bar{f}^2(Y_k)\right) + R(n)}{\left(\frac{1}{n}\sum_{j=1}^{n} w_u(Y_j)\right)^2},$$

and observe that the term $R(n) \to 0$ as $n \to \infty$.

(iv) Conclude your proof by showing that the remaining expression (with $R(n)$ removed) converges to $\bar{\sigma}^2_{p,q}$.

**6.** Consider the following *Fisher-Yates shuffle* algorithm: Define the vector $a := (1, 2, \ldots, n)$. Then, for $k = n$, $k = n - 1$, …, $k = 2$ repeat

(i) Pick an independent $J_k \sim \mathcal{U}(\{1, \ldots, k\})$.

(ii) Exchange the elements $a_k \leftrightarrow a_{J_k}$.

Report the final vector $a$ as a random permutation of $\{1, \ldots, n\}$, that is, each permutation is equally likely.

(a) How can you transform $U \sim \mathcal{U}(0, 1)$ into $j \sim \mathcal{U}(\{1, \ldots, k\})$ efficiently?

(b) Show that the vector $a$ after applying the above algorithm is a random permutation of $\{1, \ldots, n\}$.
(Hint: Start by computing the distribution of $a_n$, then $a_{n-1} \mid a_n$, …and finally $a_1 \mid a_2, \ldots, a_n$.)