Mandatory Exercise 2

Børge Olav Haug

October 18, 2018

Problem 1

a) If theoretical odds is given by $TO = 1 + \frac{1-p}{p} = \frac{1}{p}$, then $p = \frac{1}{TO}$.

	TO_H	TO_U	TO_B	p_H	p_U	p_B	$p_H + p_U + p_B$
Norge - Kypros	1.30	4.10	8.10	$\frac{10}{13}$	$\frac{10}{41}$	$\frac{10}{81}$	≈ 1.366
Tyskland - Frankrike	2.35	3.00	2.65	$\frac{20}{47}$	$\frac{1}{3}$	$\frac{20}{53}$	≈ 1.362
Portugal - Kroatia	2.20	2.90	2.85	$\frac{5}{11}$	$\frac{10}{29}$	$\frac{20}{57}$	≈ 1.150
Nederland - Peru	1.60	3.40	4.45	$\frac{5}{8}$	$\frac{5}{17}$	$\frac{20}{89}$	≈ 1.144

Since home win, draw, and away win are the only possible outcomes of a football match, the probabilities should add to 1. Since that's not the case here, it could be that the companies that arrange these betting games take advantage of the fact that betters are unlikely to check that the odds are correct, and fiddle with the numbers to save money.

b) To estimate the expected profit, let 1 represent a home win, and 0 represent a draw or loss (away win), and let p_i be the probability of a home win in game i, and let n be the number of games we bet on. Then the algorithm is:

1. Simulate game outcomes X_i by drawing a 1 or 0, with probabilites p_i and $1 - p_i$, respectively.

2. To get the profit of game outcome X_i, multiply it by the stake S and by the odds $TO_i = \frac{1}{p_i}$ and subtract the stake S, i.e. calculate $S \cdot TO_i \cdot X_i - S = S(TO_i \cdot X_i - 1)$
3. Add the profit of each individual game together to get the total profit of all games, i.e. calculate $P_{s,(j)} = \sum_{i=1}^{n} S(TO_i \cdot X_i - 1)$
4. Repeat the steps above m times to get a vector of m profits.
5. Calculate the average of these profits, $\frac{1}{m}\sum_{j=1}^{m} P_{s,(j)}$.

Instead of doing steps 1-3 in a for-loop, we can simulate all game outcomes first and put them in a matrix, and then use vectorized operations to do the rest.

Plug in $n = 4$, $\mathbf{p} = (0.67, 0.38, 0.40, 0.54)$, $S = 100$, and $m = 10000$ for this case.

c) See R-code

d)

$$X_i = \begin{cases} 1, & \text{with probability } \frac{1}{TO_i} \\ 0, & \text{with probability } 1 - \frac{1}{TO_i} \end{cases}$$

$\Rightarrow E(X_i) = 1 \cdot \frac{1}{TO_i} + 0 \cdot (1 - \frac{1}{TO_i}) = \frac{1}{TO_i}$

$E(P_s) = E[(\sum_{i=1}^{n} S \cdot TO_i \cdot X_i) - nS] = (\sum_{i=1}^{n} E(S \cdot TO_i \cdot X_i)) - E(nS) =$
$S \cdot (\sum_{i=1}^{n} TO_i \cdot E(X_i)) - nS = S \cdot (\sum_{i=1}^{n} TO_i \cdot \frac{1}{TO_i}) - nS =$
$S \cdot (\sum_{i=1}^{n} 1) - nS = nS - nS = 0$

$$X_c = \begin{cases} 1, & \text{with probability } \prod_{i=1}^{n} \frac{1}{TO_i} \\ 0, & \text{with probability } 1 - \prod_{i=1}^{n} \frac{1}{TO_i} \end{cases}$$

$\Rightarrow E(X_c) = 1 \cdot (\prod_{i=1}^{n} \frac{1}{TO_i}) + 0 \cdot (1 - (\prod_{i=1}^{n} \frac{1}{TO_i})) = \prod_{i=1}^{n} \frac{1}{TO_i}$

$E(P_c) = E[nSX_c \cdot (\prod_{i=1}^{n} TO_i) - nS] = nS \cdot (\prod_{i=1}^{n} TO_i) \cdot E(X_c) - nS = nS \cdot (\prod_{i=1}^{n} TO_i) \cdot (\prod_{i=1}^{n} \frac{1}{TO_i}) - nS = nS \cdot (\prod_{i=1}^{n} 1) - nS = nS - nS = 0$

The expected values of P_s and P_c are both equal to 0, so both approaches are equally good if we only take expected values into account. They are not equal in every respect however, as we we will see in e).

e) See R-code

Problem 2

a) In a nonhomogenous Poisson process with intensity $\lambda(t)$, $N(s+t)-N(s)$ has a Poisson distribution with expectation $\int_s^{s+t} \lambda(u)du$. In this case, the number of visitors during a day, $N(0+24)-N(0) = N(24)$, therefore has a Poisson distribution with expectation $\int_0^{0+24} \lambda(u)du = \int_0^{24} \lambda(u)du$, and since the expectation λ is part of the Poisson density function $\frac{\lambda^x e^{-\lambda}}{x!}$, we need this number to be able to make probability calculations related to $N(24)$.

To approximate the integral using the hit or miss method, we:

1. Decide on a number c such that $\lambda(t) \leq c$ for any $t \in [0, 24]$.
2. Simulate $X_1, ..., X_n$ from the $U[0, 24]$ distribution.
3. Simulate $Y_1, ..., Y_n$ from the $U[0, c]$ distribution.
4. Calculate $\frac{24c}{n} \sum_{i=1}^n I(Y_i \leq \lambda(X_i))$, where

$$I(Y_i \leq \lambda(X_i)) = \begin{cases} 1, & \text{if } Y_i \leq \lambda(X_i) \\ 0, & \text{otherwise} \end{cases}$$

   and n is the number of simulations we want to do.

One simple approach to selecting a c is to just calculate $\lambda(t)$ for a range of values in $[0, 24]$ and then select the t that gave the highest $\lambda(t)$ value. We can add a small number on top of this to compensate for the fact that we might have missed a t-value that would have given a slightly higher $\lambda(t)$ value.

The general hit or miss estimator is given by:

$\hat{\theta}_{HM} = \frac{c(b-a)}{n} \sum\limits_{i=1}^{n} I(Y_i \leq g(X_i))$

The $I(Y_i \leq g(X_i))$ can be viewed as iid Bernoulli random variables with expectation p and variance $p(1-p)$. The $c(b-a)I(Y_i \leq g(X_i))$ are then iid random variables with expectation $c(b-a)p$ and variance $c^2(b-a)^2p(1-p)$. $\hat{\theta}_{HM}$ can then be viewed as the average of these, with expectation:

$E(\hat{\theta}_{HM}) = E[\frac{1}{n} \sum_{i=1}^n c(b-a)I(Y_i \leq g(X_i))]$

$= \frac{1}{n}\sum_{i=1}^{n} E[c(b-a)I(Y_i \leq g(X_i))] = \frac{1}{n}nc(b-a)p = c(b-a)p,$

and variance:

$Var(\hat{\theta}_{HM}) = Var[\frac{1}{n}\sum_{i=1}^{n} c(b-a)I(Y_i \leq g(X_i))] =$

$\frac{1}{n^2}\sum_{i=1}^{n} Var[c(b-a)I(Y_i \leq g(X_i))] = \frac{1}{n^2}nc^2(b-a)^2p(1-p)$

$= \frac{c^2(b-a)^2}{n}p(1-p)$

Since $\hat{\theta}_{HM}$ is the average of n iid random variables, the central limit theorem applies and, $\frac{\hat{\theta}_{HM}-E(\hat{\theta}_{HM})}{\sqrt{Var(\hat{\theta})_{HM}}} = \frac{\hat{\theta}_{HM}-c(b-a)p/n}{c(b-a)\sqrt{p(1-p)/n}}$ approaches the $N(0,1)$ distribution as n grows large.

We can use this to find a $(1-\alpha)100\%$ confidence interval for $\hat{\theta}_{HM}$: $c(b-a)p \pm c(b-a)z_{\alpha/2}\sqrt{\frac{p(1-p)}{n}}$.

To have a $(1-\alpha)100\%$ probability that $\hat{\theta}_{HM}$ falls less than e from the true integral, we need:

$c(b-a)z_{\alpha/2}\sqrt{\frac{p(1-p)}{n}} < e \Rightarrow \sqrt{n} > c(b-a)z_{\alpha/2}\frac{\sqrt{p(1-p)}}{e}$

$\Rightarrow n > c^2(b-a)^2z_{\alpha/2}^2\frac{p(1-p)}{e^2}$

Since $p(1-p) < 0.25$, we have that

$n > c^2(b-a)^2z_{\alpha/2}^2\frac{0.25}{e^2}$

With $a = 0$, $b = 24$, $c = 207.6511$ (calculated in R),
$\alpha = 0.05$, $z_{\alpha/2} = 1.96$, and $e = 10$, we get the following result:

$n > 207.6511^2 \cdot 24^2 \cdot 1.96^2 \cdot \frac{0.25}{10^2} = 238530.1$

With 238531 simulations we can be 95% certain that our estimate is no more than 10 from the true answer.

b) See R-code

c) The transformation method works very well if the integrated intensity is easily inverted. The integral of the last part of $\lambda(t)$, i.e. $\int_0^t 190e^{-(u-20)^2/3}du$, can be rewritten in terms of the error function, $a \cdot \mathrm{erf}(x)$ where $\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ [1]. The error function has a unique inverse only on the interval $(-1, 1)$, and thus we are better off using the thinning method in this case.

Finding a reasonable value for λ_{max} can be done in the same way as when finding a value for c, explained in point a). Using the R code written in b), we found a reasonable value to be $\lambda_{max} = c = 207.6511$.

To find out how large the proportion of deleted arrival times is when $\lambda_{max} = 207.6511$, we have to take another look at the expectation of $N(24)$. As mentioned in a), $N(24)$ has a Poisson distribution with expectation $E[N(24)] = \int_0^{24} \lambda(u) du$.

When $\lambda(t) = 5 + 50 \sin{(\pi \cdot t/24)}^2 + 190e^{-(t-20)^2/3}$,
$E[N_{\lambda(t)}(24)] \underset{\text{hit or miss}}{\approx} 1303.66$ (This number will vary).

A HPP is a special case of the NHPP where the intensity $\lambda(t)$ is constant. So in the case where the intensity equals $\lambda_{max} = 207.6511$,

$E[N_{\lambda_{max}}(24)] = \int_0^{24} \lambda_{max} du = 207.6511 \cdot (24 - 0) = 207.6511 \cdot 24 = 4983.626$.

So we expect $E[N_{\lambda(t)}(24)] = 1303.66$ events in an NHPP with expectation $\lambda(t)$, and $E[N_{\lambda_{max}}(24)] = 4983.626$ events in a (N)HPP with expectation λ_{max}. When using the thinning algorithm we therefore expect to keep $\frac{E[N_{\lambda(t)}(24)]}{E[N_{\lambda_{max}}(24)]} \cdot 100\%$ of the values, and to delete
$(1 - \frac{E[N_{\lambda(t)}(24)]}{E[N_{\lambda_{max}}(24)]}) \cdot 100\% = (1 - \frac{1303.66}{4983.626}) \cdot 100\% \approx 74\%$ of the values.

i) To estimate the probability that the maximum number of active visitors exceeds a certain number a, we:

1. Run the queue system algorithm repeatedly, say, n times.

   1.1. Simulate arrivals using the thinning method, with $\lambda(t) = 5+50\sin{(\pi \cdot t/24)}2^2+190e^{-(t-20)^2/3}$ and $\lambda_{max} = 207.6511$.

   1.2. Simulate the amounts of time visitors spend on the website using the gamma distribution with parameters $\alpha = 2$

and $\beta = 3$. These times are measured in minutes, and should be converted to hours by dividing by 60.

1.3. Caculate the queue using the calculatequeue function from lecture code example stochastic_processes_examples.R

2. Select the maximum number of active visitors of each repetition, m_i.
3. Count how many times m_i is larger than a, and divide by number of repetitions, i.e. calculate $\frac{1}{n}\sum_{i=1}^{n} I(m_i > a)$. This is the estimated proability.

ii) To estimate the median and the 5% and 95% quantiles of the number of active visitors at a certain time point t, we:

1. Run the queue system algorithm repeatedly, n times. Same as in point i) 1.1-1.3.
2. Select the number of active visitors at time point t of each repetition, v_i. Rarely will there be an event exactly at t, so we need another approach. Some options are (but not limited to) picking the number of active visitors at the time point closest to t, or picking active visitor numbers on an interval around t and calculate the mean or median of these.
3. Calculate the median and quantiles of the v_i, using the mean() and quantile() functions in R.

d) See R-code

e) To simulate arrival times from the NHPP with intensity $\lambda_2(t) = 10 + 10t$, we first have to calculate $\Lambda_2(t) = \int_0^t \lambda_2(u)du$, and then find the inverse $\Lambda_2^{-1}(w)$.

$$\Lambda_2(t) = \int_0^t \lambda_2(u)du = \int_0^t (10 + 10u)du = [10u + 5u^2]_0^t = 10t + 5t^2$$

Solve for t to find the inverse $\Lambda_2^{-1}(w)$:

$$\Lambda_2 = 10t + 5t^2 \Rightarrow 5t^2 + 10t - \Lambda_2 = 0$$

$$\underset{quadratic formula}{\Rightarrow} t = \frac{-10 \pm \sqrt{10^2 - 4\cdot5\cdot(-\Lambda_2)}}{2\cdot5} = \frac{-10 \pm \sqrt{100 + 20\Lambda_2}}{10}$$

$$t > 0 \Rightarrow \Lambda_2^{-1}(w) = \frac{-10 + \sqrt{100 + 20w}}{10}$$

The algorithm for simulating an NHPP with intensity $\lambda_2(t) = 10 + 10t$ then becomes:

1. Calculate the expected number of events. Since $\lambda_2(t)$ is easy to integrate, we could let this be an input to the algorithm, or we can use the hit or miss method to estimate the integral. Now, let $n = 3 \cdot \int_0^{24} \lambda_2(u)du$ (generally $n = 3 \cdot \int_a^b \lambda_2(u)du$), so that we generate enough events. Not a failproof approach, but good enough in most cases.
2. Simulate an HPP with intensity 1, which can be done by simulating interarrival times $t_1, t_2, \dots, t_n$ from the exponential distribution with parameter $\lambda = 1$, and converting them into arrival times $w_1, w_2, \dots w_n$ by calculating cumulative sums.
3. Calculate $s_1 = \Lambda_2^{-1}(w_1), s_2 = \Lambda_2^{-1}(w_2), \dots s_n = \Lambda_2^{-1}(w_n)$.
   (In the general case, calculate
   $s_1 = a + \Lambda_2^{-1}(w_1), s_2 = a + \Lambda_2^{-1}(w_2), \dots s_n = a + \Lambda_2^{-1}(w_n)$.
   In this case $a = 0$).
4. Throw away all $s_i > 24$ (or, generally $s_i > b$), (and rename the remaining s_i) so that we now have $s_1, s_2, \dots, s_m$.
5. Deliver $s_1, s_2, \dots, s_m$.

f) See R-code

g) See R-code

References

[1] Error function. (n.d.). In *Wikipedia.* Retrieved October 16, 2018, from `https://en.wikipedia.org/wiki/Error_function`

[2] Jan Terje Kvaløy, *Notes, lecture slides, and code examples from the course STA510 Statistical modelling and simulation*, 2018

[3] Maria L. Rizzo, *Statistical Computing with R*, Chapman & Hall/CRC, London, 1st Edition, 2007