MPCS 58020 2017: Homework 2

Due date: Monday April 17, 2017 5:30pm (before class)

Solve the following problems and show your work, or if specified in the question write a program to solve the problem. Create a README file describing how the solutions are organized, requirements for running the programs, and how to run them. The README can be short and sweet, this is not a project on github, but always defining a README is a good habit and will make things easier for grading. Upload your solution directory to your private dropbox directory (you will receive an email invite) with name `hw2`. For non programming problems, scanned handwritten work is fine.

Programming Guidelines

You may use any programming language, but not high level functionality specific to the homework problem. In particular, you may use basic linear algebra routines, but not specialized statistics features. You may use uniform random number generation on $(0, 1)$ and integer generation within a range, but not more complicated distributions. If you are unsure if a feature is allowed, ask on slack #general. You may of course use any features you want to debug and check your results, but the main solution cannot use the them. If you are unsure on what language to use, I recommend Python+numpy, Julia, Matlab, or Octave.

For problems 6-8, your program should support running with no arguments and use enough realizations (trials) to get a good estimate of the result without an unreasonably long exeuction time (this can be determined by trial and error). It should take an optional argument specifying the number of simulations. A command line program that prints the results is preferred, but a main script (e.g. in Matlab) is also fine.

Probability

For questions 1-5, your solutions should be exact and not a simulation, except where specified for problem of 4. Consider using simulation to check your answers, but don't include that with your solution set.

1. You ask your neighbor to feed your beloved 17-year-old dog, Charlie, while you are on vacation. If he isn't fed, there is an 80% probability that Charlie will die. If he is fed, there is still a 15% probability that he'll die. You are 90 percent certain that your neighbor will remember to feed Charlie.

   (a) What is the probability that Charlie will be alive when you return?

   (b) If Charlie is dead, what is the probability your neighbor forgot to feed him?

   Solution We are given the following information:

   $$\begin{aligned} P(\text{dead} \mid \text{not fed}) &= 0.8 \\ P(\text{dead} \mid \text{fed}) &= 0.15 \\ P(\text{fed}) &= 0.9 \end{aligned}$$

   (a)

   $$\begin{aligned} P(\text{dead}) &= P(\text{dead} \mid \text{fed})P(\text{fed}) + P(\text{dead} \mid \text{not fed})P(\text{not fed}) \\ &= 0.8 \cdot 0.1 + 0.15 \cdot 0.9 = 0.215 \\ P(\text{alive}) &= 1 - P(\text{dead}) = 0.785 \end{aligned}$$

   (b) Using Baye's theorem:

   $$\begin{aligned} P(\text{not fed} \mid \text{dead}) &= \frac{P(\text{dead} \mid \text{not fed})P(\text{not fed})}{P(\text{dead})} \\ &= \frac{0.8 \cdot 0.1}{0.215} \\ &\approx 0.372 \end{aligned}$$

2. The continuous random variable X has the following probability density function, with fixed parameters x_m and α, both of which are greater than zero.

$$f(x) = \begin{cases} 0 & \text{if } x < x_m \\ \dfrac{\alpha x_m^\alpha}{x^{\alpha+1}} & \text{if } x \geq x_m \end{cases}$$

(a) What is the cumulative distribution function, $F(x)$?

(b) What is $E[X]$?

(c) What is $Var(X)$?

Solution

(a) The CDF is the integral of the PDF:

$$F(x) = \int_{-\infty}^{x} f(x')dx' = \begin{cases} 0 & \text{if } x < x_m \\ \int_{x_m}^{x} \dfrac{\alpha x_m^\alpha}{x'^{\alpha+1}} dx' & \text{if } x \geq x_m \end{cases}$$

$$\begin{aligned} \int_{x_m}^{x} \frac{\alpha x_m^\alpha}{x'^{\alpha+1}} dx' &= \alpha x_m^\alpha \int_{x_m}^{x} x'^{-\alpha-1} dx \\ &= \left[\alpha x_m^\alpha \frac{x'^{-\alpha}}{-\alpha} \right]_{x'=x_m}^{x} \\ &= \frac{-x_m^\alpha}{x^\alpha} - \frac{-x_m^\alpha}{x_m^\alpha} \\ &= 1 - \frac{x_m^\alpha}{x^\alpha} \end{aligned}$$

So

$$F(x) = \begin{cases} 0 & \text{if } x < x_m \\ 1 - \dfrac{x_m^\alpha}{x^\alpha} & \text{if } x \geq x_m \end{cases}$$

(b)

$$\begin{aligned} E[X] &= \int_{-\infty}^{\infty} x f(x) dx \\ &= \int_{x_m}^{\infty} x \frac{\alpha x_m^\alpha}{x^{\alpha+1}} dx \\ &= \alpha x_m^\alpha \int_{x_m}^{\infty} x^{-\alpha} dx \\ &= \alpha x_m^\alpha \left[\frac{x^{1-\alpha}}{1-\alpha} \right]_{x=x_m}^{\infty} && \text{for } \alpha \neq 1 \\ &= \frac{\alpha x_m^\alpha}{1-\alpha} \left[\left(\lim_{x\to\infty} x^{1-\alpha} \right) - x_m^{1-\alpha} \right] \\ &= \frac{\alpha x_m^\alpha}{1-\alpha} \left(\lim_{x\to\infty} x^{1-\alpha} \right) - \frac{\alpha x_m}{1-\alpha} \end{aligned}$$

The limit is 0 for $\alpha > 1$; for $\alpha < 1$ it's ∞. For $\alpha = 1$, the antiderivative is different:

$$\begin{aligned} E[X] &= x_m \int_{x_m}^{\infty} x^{-1} dx \\ &= x_m \left[\ln(x) \right]_{x=x_m}^{\infty} \\ &= x_m \left[\left(\lim_{x\to\infty} \ln(x) \right) - \ln(x_m) \right] \\ &= \infty \end{aligned}$$

Therefore:

$$E[X] = \begin{cases} \infty & \text{if } \alpha \leq 1 \\ \dfrac{\alpha x_m}{\alpha - 1} & \text{if } \alpha > 1 \end{cases}$$

(c) When $\alpha \leq 1$ and $E[X] = \infty$, it doesn't make sense to talk about variance - the deviation from ∞ is not well defined. For $\alpha > 1$:

$$\begin{aligned}
Var(X) &= \int_{-\infty}^{\infty} x^2 f(x)dx - E[X]^2 \\
\int_{-\infty}^{\infty} x^2 f(x)dx &= \int_{x_m}^{\infty} x^2 \frac{\alpha x_m^\alpha}{x^{\alpha+1}} dx \\
&= \alpha x_m^\alpha \int_{x_m}^{\infty} x^{1-\alpha} dx \\
&= \alpha x_m^\alpha \left[\frac{x^{2-\alpha}}{2-\alpha}\right]_{x=x_m}^{\infty} \qquad \text{for } \alpha \neq 2 \\
&= \frac{\alpha x_m^\alpha}{2-\alpha}\left[\left(\lim_{x\to\infty} x^{2-\alpha}\right) - x_m^{2-\alpha}\right] \\
&= \frac{\alpha x_m^\alpha}{2-\alpha}\left(\lim_{x\to\infty} x^{2-\alpha}\right) - \frac{\alpha x_m^2}{2-\alpha}
\end{aligned}$$

The limit is 0 for $\alpha > 2$, and ∞ for $\alpha < 2$. For $\alpha = 2$ the antiderivative is different:

$$\begin{aligned}
\int_{-\infty}^{\infty} x^2 f(x)dx &= 2x_m^2 \int_{x_m}^{\infty} x^{-1} dx \\
&= 2x_m^2 \left[\ln(x)\right]_{x_m}^{\infty} \\
&= \infty
\end{aligned}$$

For $\alpha > 2$:

$$\begin{aligned}
Var(X) &= \frac{\alpha x_m^2}{\alpha-2} - \frac{\alpha^2 x_m^2}{(\alpha-1)^2} \\
&= \alpha x_m^2 \left(\frac{1}{\alpha-2} - \frac{\alpha}{(\alpha-1)^2}\right) \\
&= \alpha x_m^2 \frac{(\alpha-1)^2 - \alpha(\alpha-2)}{(\alpha-2)(\alpha-1)^2} \\
&= \alpha x_m^2 \frac{\alpha^2 - 2\alpha + 1 - \alpha^2 + 2\alpha}{(\alpha-2)(\alpha-1)^2} \\
&= \alpha x_m^2 \frac{1}{(\alpha-2)(\alpha-1)^2} \\
&= \frac{\alpha}{\alpha-2}\left(\frac{x_m}{\alpha-1}\right)^2
\end{aligned}$$

Therefore:

$$Var(X) = \begin{cases} \infty & \text{if } \alpha \leq 2 \text{ and } \alpha > 1 \\ \dfrac{\alpha}{\alpha-2}\left(\dfrac{x_m}{\alpha-1}\right)^2 & \text{if } \alpha > 2 \end{cases}$$

3. The lifetime in hours of a certain kind of radio tube is a random variable having a probability density function given by:

$$f(x) = \begin{cases} 0 & \text{if } x \leq 100 \\ \frac{100}{x^2} & \text{if } x > 100 \end{cases}$$

In a set of 5 radio tubes, what is the probability that at least one tube will need to be replaced within the first 150 hours of operation?

Solution The probability of a single tube failing in 150 hours is:

$$\begin{aligned}
P(X \leq 150) &= \int_{100}^{150} 100x^{-2} \\
&= -100x^{-1}\big|_{100}^{150} \\
&= \frac{-100}{150} - \frac{-100}{100} \\
&= 1 - \frac{2}{3} \\
&= \frac{1}{3}
\end{aligned}$$

The chances of it not failing in the first 150 hours is $\frac{2}{3}$, so the chance of five tubes not failing is $\frac{2}{3}^5$. The chance of at least one failing in 150 hours is therefore $1 - \frac{2}{3}^5$, or $\frac{211}{234}$.

4. The random variables X and Y have a joint PDF specified by:

$$f(x,y) = 2e^{-(x+2y)} \qquad 0 < x < \infty, 0 < y < \infty$$

What is $P\{X < Y\}$?. Calculate the answer analytically, and then write a program that simulates drawing from $f(x,y)$ to estimate the probability. Compare your results.

Solution

$$\begin{aligned}
P\{X < Y\} &= \int_0^\infty \int_x^\infty 2e^{-(x+2y)} dy dx \\
&= \int_0^\infty \left(-e^{-(x+2y)}\right)|_x^\infty dx \\
&= \int_0^\infty \left(\lim_{y\to\infty} -e^{-(x+2y)}\right) - -e^{-3x} dx \\
&= \int_0^\infty e^{-3x} dx \\
&= -\frac{1}{3} e^{-3x}|_0^\infty \\
&= \left(\lim_{x\to\infty} -\frac{1}{3} e^{-3x}\right) + \frac{1}{3} e^{-3(0)} \\
&= \frac{1}{3}
\end{aligned}$$

For simulation, note that X and Y are actually independent:

$$\begin{aligned}
f_x(x) &= \int_0^\infty 2e^{-(x+2y)} dy \\
&= -e^{-(x+2y)}|_{y=0}^\infty \\
&= \lim_{y\to\infty} \left(-e^{-(x+2y)}\right) - -e^{-x} \\
&= e^{-x} \\
f_y(y) &= \int_0^\infty 2e^{-(x+2y)} dx \\
&= -2e^{-(x+2y)}|_{x=0}^\infty \\
&= \lim_{x\to\infty} \left(-2e^{-(x+2y)}\right) - -2e^{-2y} \\
&= 2e^{-2y} \\
f_x(x) f_y(y) &= e^{-x} \cdot 2e^{-2y} = 2e^{-x-2y} = f(x,y)
\end{aligned}$$

Furthermore the marginal PDFs are easy to integrate and invert:

$$\begin{aligned}
F_x(x) &= \int_0^x f_x(x') dx' = \int_0^x e^{-x'} dx' \\
&= -e^{-x'}|_{x'=0}^x = 1 - e^{-x} \\
F_x^{-1}(u) &= -\log_e(1-u) \\
F_y(y) &= \int_0^y f_y(y') dy' = \int_0^y 2e^{-2y'} dy' \\
&= -e^{-2y'}|_{y'=0}^y = 1 - e^{-2y} \\
F_y^{-1}(v) &= -\frac{1}{2}\log_e(1-v)
\end{aligned}$$

See `p4.jl` for an example simulation.

5. If the random variables X and Y have a joint PDF $f(x,y)$ such that:

$$P\{X \in C, Y \in D\} = \int_D \int_C f(x,y)\, dx\, dy$$

Then the *marginal PDFs* $f_X(x)$ and $f_Y(y)$ are the univariate PDFs given by:

$$\begin{aligned}
f_X(x) &= \int_D f(x,y)\, dy \\
f_Y(y) &= \int_C f(x,y)\, dx
\end{aligned}$$

And it follows that:

$$E[X] = \int_C x f_X(x)\, dx$$
$$E[Y] = \int_D y f_Y(y)\, dy$$

Consider the following PDF:

$$f(x,y) = \frac{9}{10}xy^2 + \frac{1}{5} \qquad 0 < x < 2, \quad 0 < y < 1$$

What is $\mathrm{Cov}(X,Y)$?

Solution

$$\begin{aligned}
f_X(x) &= \int_0^1 \left(\frac{9}{10}xy^2 + \frac{1}{5}\right) dy \\
&= \left[\frac{3}{10}xy^3 + \frac{1}{5}y\right]_{y=0}^{1} \\
&= \frac{3}{10}x + \frac{1}{5} \\
E[X] &= \int_0^2 x\left(\frac{3}{10}x + \frac{1}{5}\right) dx \\
&= \int_0^2 \left(\frac{3}{10}x^2 + \frac{1}{5}x\right) dx \\
&= \left[\frac{1}{10}x^3 + \frac{1}{10}x^2\right]_0^2 \\
&= \frac{8}{10} + \frac{4}{10} = \frac{6}{5} \\
f_Y(y) &= \int_0^2 \left(\frac{9}{10}xy^2 + \frac{1}{5}\right) dx \\
&= \left[\frac{9}{20}x^2y^2 + \frac{1}{5}x\right]_{x=0}^{2} \\
&= \frac{9}{5}y^2 + \frac{2}{5} \\
E[Y] &= \int_0^1 y\left(\frac{9}{5}y^2 + \frac{2}{5}\right) dy \\
&= \int_0^1 \left(\frac{9}{5}y^3 + \frac{2}{5}y\right) dy \\
&= \left[\frac{9}{20}y^4 + \frac{2}{10}y^2\right]_0^1 \\
&= \frac{9}{20} + \frac{2}{10} = \frac{13}{20}
\end{aligned}$$

$$\begin{aligned}
E[XY] &= \int_0^1 \int_0^2 xy\left(\frac{9}{10}xy^2 + \frac{1}{5}\right) dxdy \\
&= \int_0^1 \int_0^2 \left(\frac{9}{10}x^2y^3 + \frac{1}{5}xy\right) dxdy \\
&= \int_0^1 \left[\frac{3}{10}x^3y^3 + \frac{1}{10}x^2y\right]_0^2 dy \\
&= \int_0^1 \left(\frac{12}{5}y^3 + \frac{2}{5}y\right) dy \\
&= \left[\frac{3}{5}y^4 + \frac{1}{5}y^2\right]_0^1 \\
&= \frac{3}{5} + \frac{1}{5} = \frac{4}{5}
\end{aligned}$$

$$\begin{aligned} Cov(X,Y) &= E[XY] - E[X]E[Y] \\ &= \frac{4}{5} - \frac{6}{5} \cdot \frac{13}{20} = \frac{1}{50} \end{aligned}$$

Simulation

6. ***Random permutations*** *can be generated using a variation of the discrete inverse transform method, as described in Ross, Example 4b.*

   A deck of 50 cards are labeled with the numbers 1,2,...,50. The cards are shuffled and then turned over one card at a time. Say that a "hit" occurs whenever card labeled when the ith card to be turned over is labeled with the number i. Let the random variable X be the total number of hits after all cards have been turned over.

   (a) Without a simulation, derive the expected value and variance of X.

   (b) Compose and run a simulation to estimate the expected value and variance of X.

   Solution Since expected value of random variables is linear, even for non-independent variables, the expected value can be calculated using the sum of expected values of indicator random variables, X_i, defined to be 1 if the ith card is a hit and 0 otherwise. $E[X_i] = P[X_i = 1] = \frac{1}{50}$.

   $$E[X] = \sum_{i=1}^{50} E[X_i] = 50 * \frac{1}{50} = 1$$

   This is actually true for any number of cards. For large n, the distribution of X can be approximated with a Poisson variable, and the variance also approaches $np = 1$. This works reasonably well at $n = 50$.

   For a more exact calculation of the variance, i.e. at small values of n, a more precise analysis is required. Let X_n denote the random variable representing hits given a deck of n cards. Note that for $P[X_n = n - 1] = 0$, since there can't be only one mismatch, and $P[X_n = n] = \frac{1}{n!}$. For all remaining $i < n - 1$:

   $$P[X_n = i] = \frac{\binom{n}{i} N_{n-i}}{n!}$$

   where N_i is the number of permutations of i integers with no hits. This is because getting exactly i hits requires getting a hit in any of the $\binom{n}{i}$ subsets of positions, and getting no hits in the remaining $n - i$ positions.

   To have no hits, the first integer must not match, which leaves $i - 1$ possible integers in the first position. For each such choice, there are $i - 1$ un-chosen numbers, one of which cannot match because it is in the first position, out of play. If the permutation of the remaining numbers is such that it wouldn't have matched even without the missing number, it still results in no matches, and there are N_{i-1} such permutations. If the 1 is in the position of the missing number (that was swapped into the first position), then there are $i - 2$ remaining numbers that could match their own position, and there are N_{i-2} such permutations of those remaining numbers with no matches. This gives rise to the following recurrence:

   $$N_i = (i - 1)(N_{i-1} + N_{i-2})$$

   and it is easy to see that $N_1 = 0$ and $N_2 = 1$. Then for $i < n - 1$:

   $$\begin{aligned} P[X_n = i] &= \binom{n}{i} \frac{(n - i - 1)(N_{n-i-1} + N_{n-i-2})}{n!} \\ &= \frac{(n - i - 1)(N_{n-i-1} + N_{n-i-2})}{i!(n - i)!} \end{aligned}$$

   Plugging these formulas into a computer program with an arbitrary fixed precision fp library (e.g. python mpmath), and using $Var(X_n) = E[X_n^2] - E[X_n]^2$, one can see that $Var(X_{50}) - 1 < 10^{-62}$, so the Poisson variable is actually a very good approximation. Using this direct method, you can't even detect the deviation from the Poisson approximation using double precision float at $n = 50$.

7. ***Exponential random variables*** *have the PDF $f(x) = \lambda e^{-\lambda x}$ and the CDF $F(x) = 1 - e^{-\lambda x}\, dx$ over the interval $(0, \infty)$.*

   In many applications, the exponential distribution can describe a continuous quantity that may take on any positive value, but for which larger values are increasingly unlikely. For example, the time it takes for a radioactive particle to to decay is an exponential random variable. Ross, Example 5b, describes how to use the inverse transform method to simulate exponential random variables.

A casualty insurance company has 1000 policyholders, each of whom will independently present a claim in the next month with probability 0.05. Assuming that the amounts of the claims made are independent exponential random variables with mean \$800, use a simulation to estimate the probability that the sum of these claims will exceed \$50,000.

8. In class, we used the rejection method to generate random numbers with PDF:

$$f(x) = 3x^2 \quad 0 < x1$$

by generating numbers from:

$$g(x) = 1 \quad 0 < x < 1$$

. We can make a more efficient implementation by using a different PDF in place of $g(x)$.

   (a) Use the rejection method to generate random numbers from $f(x)$ by generating random numbers from:

$$h(x) = 2x \quad 0 < x < 1$$

   .

   (b) Attempt to find your own h(x) that overall give superior performance.

   (c) Compare the efficiency (average time and iterations per random number accepted; and associated variances for both) of the three implementations (with $g(x)$ and $h(x)$).

   (d) Derive a performance model for the expected number of iterations per random number accepted (and the associated variance) for both implementations. How does your performance model compare to your results?

   Define a main program that runs both methods, and prints out the mean and variance of the generated numbers, the mean and variance of the number of iterations per random number accepted, and the mean and variance of the run time in seconds.

   Include a copy of the output of your main program as plaintext in `problem8_output.txt`.

   Solution $h(x) = a\tan(x)$ on $(0, 1)$ seems like an interesting candidate, although one which may be expensive to draw from. We need to choose a such that the integral over $(0, 1)$ is 1.

$$\begin{aligned} H(x) &= \int a\tan(x)dx = a\ln(\sec(x)) \\ H(1) &= a(\ln(\sec(1)) - \ln(\sec(0))) \\ 1 &= a\ln(\sec(1)) \\ a &= \frac{1}{\ln(\sec(1))} \end{aligned}$$

   $H(x)$ is not hard to invert, so we can use the Inverse Transform method to generate random numbers with PDF $h(x)$:

$$H^{-1}(x) = \text{arcsec}(e^{x\ln(\sec(1))})$$

   To find c, we need to find the max of $r = f/h$ on the interval $(0,1)$:

$$\begin{aligned} 0 &= \frac{d}{dx}\frac{3x^2}{a\tan(x)} \\ 0 &= \frac{6x(a\tan(x)) - 3x^2(a\sec^2(x))}{a^2\tan^2(x)} \\ 6x(a\tan(x)) &= 3x^2(asec^2(x)) \\ 2a\tan(x) &= xsec^2(x) \\ x &= 2a\sin(x)\cos(x) = a\sin(2x) \\ x &= \frac{\sin(2x)}{\ln(\sec(1))} \end{aligned}$$

   I can't figure out how to solve that analytically. Looking at the graphs, using $c = 1.2$ is sufficient to insure that $f(x) \leq ch(x)$ on $(0, 1)$.

   See 'prob8.py' for implementation. The output is:

```
==== expected ====
mean 0.75
var  0.0375
==== rand_1 ====
mean 0.748564709507
var  0.0377884446976
iters   3.0013 6.00939831
seconds 1.82693481445e-06 1.21617530349e-12
==== rand_2x ====
mean 0.750425910777
var  0.0374728484848
iters   1.50072 0.7466794816
seconds 1.17320299149e-06 1.94740585306e-13
==== rand_tan ====
mean 0.749986704249
var  0.0373883141761
iters   1.19923 0.2397774071
seconds 2.36648797989e-06 5.52618070463e-13
==== rand_inv ====
mean 0.749397053944
var  0.0377179044969
iters   1.0 0.0
seconds 9.05940532684e-07 3.10152447526e-14
```

The two values after 'iters' and 'seconds' are the mean and variance. While it does use less iterations, $a\tan(x)$ is actually the slowest. This is likely because sampling from it and calculating the rejection threshold is expensive. The fastest is just using the Inverse Transform method, but of the Rejection method implementations, $2x$ was the fastest.

According to Ross, the number of iterations is a geometric random variable with mean c. This means (no pun intended) that the parameter p is $\frac{1}{c}$, so the variance should be:

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

The observed values for both mean and variance for each rejection method above fit with this model.

Time Series

These problems cover Chapter 2.2 and 2.3 of Shumway and Stoffer, *Time Series Analysis and its Applications* (the 3rd ed, with 603 pages and a yellow cover) and Lecture 3.

- Problem 2.1, page 78. Use either of the data files from the class dropbox in the `datasets` directory: `jj_tables.txt` or `jj_series.txt`. The series contains the same values as the table, listed in chronological order by (year,quarter).