Mathematical Statistics

SF1901 Probability Theory and Statistics: Autumn 2016
Lab 1 for
TCOMK

Introduction

This is the instructions for computer exercise 1, please bring a printed copy to computer exercise 1. Please read the instructions carefully, and make sure that you understand what the MATLAB code included does. The computer exercise is pass/fail. In order to pass you first have to be able to present written solutions to the preparatory exercises. Each student should have prepared solutions **individually**. For the computer exercise it is allowed to work in groups of **at most two** persons per group. If you pass the computer exercise you will be given 3 bonus points at the written exam Wednesday October 26 2016, **given that you at the exam obtain at least 20 points without the bonus points**.

Preparatory exercises

1. Consider a continuous random variable X. For this random variable write down the definition of the cumulative distribution function and the density function, and the relationship between the two.

2. Suppose that the random variable X has a density function of the following form

$$f_X(x) = \lambda e^{-\frac{x}{\lambda}} + \frac{\lambda}{x}, \quad x \in [1,\, 10] \tag{1}$$

where λ is a real number.

   (a) Determine (approximately) the value of λ that makes f_X a density function.

   Answer: .....

   (b) Determine the cumulative distribution function of X.

   Answer: .....

(c) Determine the probability that X is less than 7. Use the value $\lambda = 0.4267$.

Answer: .....

(d) Compute $E[X]$.

Answer: .....

3. Let U be uniformly distributed over the interval $[0, 2\pi]$, and compute

(a) $E[\cos(U)]$

Answer: .....

(b) $E[\sin(U)^2]$

Answer: .....

4. State and explain the contents of the Law of Large Numbers (LLN).

5. State and explain the contents of the Central Limit Theorem (CLT).

6. This is optional! Read about bootstrap at https://en.wikipedia.org/wiki/Bootstrapping_(statistics).

Purpose and further introduction

Start by downloading the following files

1. `plot_mvnpdf.m`
2. `hist_density.m`
3. `birth.dat`
4. `birth.txt` - description of the data `birth.dat`

from the homepage of the course. Make sure the files are downloaded to the directory you will be working in. To make sure the files are in the right directory type `ls` to list the files in the current working directory.
You can write your commands directly at the prompt in MATLAB, but usually it is easier to work in the editor. If the editor is not open you can open it and create a new file by typing `edit lab1.m`. The code included below is written in sections. A new section is begun by typing two percent signs. The code `Ctrl+Enter` executes the commands in a section.

The theme for this computer exercise is *simulation*. In the part of the course which treats probability theory you will learn how to compute different quantities, such as probabilities, expected values etc., given a certain random distribution. For more complicated random models it can be very time-consuming, or even impossible to do exact computations.

In such circumstances simulation can be an alternative. You then use a computer to simulate outcomes from a certain distribution, and then you can use for instance the mean of the outcomes to estimate the expected value, or some empirical quantile to estimate a probability. In this computer exercise we will do this for some rather simple problems (where explicit computations are possible), but the same basic principles can be applied to more difficult problems where explicit computations are not possible.

Problem 0 - Computing probabilities

Read the help-files of the functions `binocdf`, `binopdf`, `normcdf`, `normpdf`, `expcdf` and `exppdf`.
Let $X_1 \in Bin(10, 0.3)$, $X_2 \in N(5, 3)$, and $X_3 \in Exp(7)$, and compute (using the above mentioned functions) for $k = 1, 2, 3$, i.e. for each of the random variables X_1, X_2, and X_3

1. $P(X_k \leq 3)$

   Answer: .....

2. $P(X_k > 7)$

   Answer: .....

3. $P(3 < X_k \leq 4)$

   Answer: .....

Problem 1 - Probability density functions

Plot the density function of an exponentially distributed random variable with expectation μ.

```
%% Problem 1: exp-pdf
    dx = 0.1;
    x = 0:dx:15;        % Creates vector with increments of dx
    mu = 1;
    y = exppdf(x, mu);  % the density function of Exp(mu)
    plot(x,y)
```

Now do the same thing for the density function you worked with in the preparatory exercise 2.

```
%% Problem 1: lambda-plot
    lambda = 0.4267;
    f=(lambda*exp(-x/lambda)+lambda./x).*(x >= 1 & x <= 10);
    plot(x, f)
```

Discuss the differences between the two distributions.

Comments: .....

.....

Problem 2 - Multivariate normal distribution

The multivariate normal distribution can be visualized using `plot_mvnpdf`. Investigate what the function does and try some different values for the parameters.

```
%% Problem 2: Multivariate normal
    mux = 0; muy = 100; sigmax = 1; sigmay = 4; rho = 0.7;
    plot_mvnpdf(mux, muy, sigmax, sigmay, rho)
```

How does changing the parameters values affect the plot?

Comments: .....

.....

Problem 3 - Simulating random numbers

In this exercise you should generate a large number of random numbers, draw a histogram of the simulated data, and finally plot the true density function in the same figure as the histogram.

```
%% Problem 3: Simulating random numbers
mu = 10;
N = 1e4;
y = exprnd(mu, N, 1); % Generates N Exp(mu) random numbers
hist_density(y);      % Creates a normalized histogram
t = linspace(0, 100, N/10); % Vector of N/10 points
hold on
plot(t, exppdf(t, mu), 'r') % 'r' means red line
hold off
```

Please note that the function `exprnd` in MATLAB takes the expectation μ as a parameter, just like the textbook [2]. Other textbooks and programs use $\lambda = 1/\mu$ as the parameter. Redo the simulation and study how the histogram changes. What is the relation between the red line and the histogram, and how would you explain the fluctuations around the red line?

Comments: .....

.....

Problem 4 - LLN, Monte Carlo and CLT

In this section you should again generate a large number of random numbers. This time they will be used to estimate expectations and probabilities. Suppose that you for some reason are interested in the expected value of the roll of a dice. This is not hard to compute analytically, but you could also imagine throwing the dice a large number of times and take the mean value of the throws. If $X_1, X_2, \ldots, X_n$ are identically distributed with expectation μ then according to the law of large numbers in [2] it holds that

$$P\left(\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| < \varepsilon\right) \to 1 \tag{2}$$

for every $\varepsilon > 0$ when $n \to \infty$. This means that the probability that the difference between the true value of the expectation and the mean value is less

than ε tends to one as the number of observations (in the example above the number of throws) tends to infinity. This approach to estimating expected values is known as a Monte Carlo method.

The idea behind the Monte Carlo method is old and can be found in mathematics from the 18th century, but it was not until the second half of the 20th century, when computers made it possible to do massive amounts of computations, that the modern versions of the method were developed. In the late 1940s Stanislaw Ulam and John von Neumann developed methods to make the "throw of a dice" using a computer, see [1]. The method has been named after the Monte Carlo Casino in Monaco, and was used in the simulations for the Manhattan Project, a research and development project that produced the first nuclear weapons.

Illustration of the Law of Large Numbers

The code below simulates M random numbers from an exponential distribution. The mean value of the generated outcomes is plotted as the simulation proceeds. The plot thus shows how your estimate, the mean value of the outcomes, develops as the number of simulated outcomes `k` increases.

```
%% Law of large numbers
mu = 0.5;
M = 500;
X = exprnd(mu, M, 1);
plot(ones(M, 1)*mu, 'r-.')
hold on
for k = 1:M
    plot(k, mean(X(1:k)), '.')
    if k == 1
        legend('True \mu', 'Estimated \mu')
    end
    xlabel(num2str(k)), pause(0.001)
end
hold off
```

If you get tired of waiting you can comment out the row with `pause` by typing a percent sign at the beginning of the row. Does the plot look the way you expected it to?

Answer: .....

Illustration of the Central Limit Theorem

The code below simulates random numbers from an exponential distribution and then adds them up. Study the code and explain what `N` represents.

Answer: .....

```
%% CGS
M = 1e3;
N = 4;
mu = 5;
X = exprnd(mu, M, N);
S = cumsum(X, 2);
for k = 1:N
    hist(S(:, k), 30)
    xlabel(num2str(k))
    pause(0.1)
end
```

Change the value of `N`, what happens when you increase and decrease the value, respectively? Why?

Comments: .....

.....

For what value of `N` does it look like there is no change if you increase it further?

Answer: .....

What distribution does the sums appear to have? Why do they have this distribution?

Answer: .....

Expectation

Now you should try to estimate the expectations you computed exactly in the preparatory exercise 3. The second expectation $E[\sin(U)^2]$ can be estimated by running the code below.

```
%% Monte Carlo
N = 1e5;
U = rand(N, 1)*2*pi;
mean(sin(U).^2);
```

Check that your estimate is in agreement with what you would expect from the exact calculation. Then rewrite the code to estimate the first expectation $E[\cos(U)]$. Again check that your estimate is in agreement with what you would expect.
Now let X and Y be independent random variables, where X is $Exp(\lambda)$, $\lambda = 4$ (recall that $\mu = 1/\lambda$!) and Y is normally distributed with expectation 0 and standard deviation 1.
Use the Monte Carlo method to estimate $E[(e^X)^{\cos(Y)}]$.

Answer: .....

Problem 5 - Descriptive statistics

In this exercise you should look at the difference between the expected values in two populations. For instance you could look at the difference in birth weight between children whose mothers smoked during the pregnancy, and children whose mothers did not smoke during the pregnancy (If you like you can of course take two other populations and/or some other variable to study).
In the file `birth.txt` you can read (in Swedish) that column number 20 of `birth.dat` contains information about smoking habits, and that the values 1 and 2 indicate that the mother did not smoke during the pregnancy, whereas the value 3 indicates that the mother did smoke during the pregnancy. You can therefore create two vectors `x` and `y` containing birth weights for children of non-smoking and smoking mothers, respectively, using the following code

```
>> x = birth(birth(:, 20) < 3, 3);
>> y = birth(birth(:, 20) == 3, 3);
```

The code `birth(:, 20) < 3` returns a vector of "true" (indicated by the value 1) and "false" (indicated by the value 0), and only those rows of column 3 (which contains the birth weights in `birth`) for which the comparison is true, end up in the vector `x`.

Use the code below to be able to visually assess data.

```
load lab2data/birth.dat
x = birth(birth(:, 20) < 3, 3);
y = birth(birth(:, 20) == 3, 3);
subplot(2,2,1)
boxplot(x)
axis([0 2 500 5000])
subplot(2,2,2)
boxplot(y)
axis([0 2 500 5000])
subplot(2,2,3:4)
ksdensity(x)
hold on
[fy, ty] = ksdensity(y);
plot(ty, fy, 'r')
hold off
```

What do the plots mean? Can you infer anything about the birth weights?

Comments: .....

.....

Problem 6 - Bootstrap (optional!)

The name bootstrap stems from the saying "to pull oneself up by one's bootstraps", which appeared in the early 19th century as an example of an impossible task. The term has also been attributed to the story about Baron von Münchausen in which he pulls himself (and his horse) out of a swamp by his hair. This actually describes the idea behind the statistical version rather well: Given access to a limited number of observations one would like to infer something about what could be said if one had access to more observations. Here this is done by creating new samples by random sampling with replacement from the existing observations using the function `randsample`.

In this exercise we will start by looking at an example with simulated data, which means that we know what the results *should* look like, and in the next computer exercise (computer exercise 2) you should continue by looking at real data, which is of course what you are usually interested in.

Simulation of sums of exponential random variables

If $X_1, X_2, \ldots, X_n$ are random variables which are independent and exponentially distributed with intensity λ (expectation $\mu = 1/\lambda$), then it holds that

$$Y_n = \sum_{k=1}^{n} X_k \in \Gamma(n, \frac{1}{\lambda}). \tag{3}$$

In other words, the sum of exponentially distributed random variables is gamma distributed. (In the special case when the first parameter n of the Γ distribution is a positive integer the distribution is also referred to as an *Erlang* distribution.) Let us start by simulating M sums as in (3). In the code below n from (3) is denoted by `n_sum`. Once the sums have been simulated you should study their histogram. Please study the code below. Read the help function for `randsample` och familiarize yourself with how the function works. The command `reshape` is used below to reshape the matrix `X` into a vector, which can be used as input to the function `randsample`. Please note that the expectation $\mu = \frac{1}{\lambda}$ is used as a parameter in the code.

```
%% Bootstrap - Simulation
% lambda = 1/5;
mu = 5;
M = 1e3;
n_sum = 5;
X = exprnd(mu, M, n_sum);
g = sum(X, 2);
subplot(211)
hist_density(g)
hold on
t = 0:0.01:mu*10;
plot(t, gampdf(t, n_sum, mu), 'r')
hold off
B = 1e3;    % Number of draws for the bootstrap sample
totalNoSamples = M*n_sum;
X = reshape(X, totalNoSamples, 1); % Reshape X into a ...
    vector
yBoot = zeros(B, 1);
for j = 1:B
    sampleDraws = X(randsample(totalNoSamples, n_sum, 1));
    yBoot(j) = sum(sampleDraws);
end
subplot(212)
hist_density(yBoot)
hold on
plot(t, gampdf(t, n_sum, mu), 'r')
hold off
```

Explain what the two rows in the for-loop do!

Comments: .....

.....

Change the values of `B`, `M`, and `mu`, respectively. What happens and why?

Comments: .....

.....

Referenser

[1] Eckhardt, Roger (1987) Stan Ulam, John Von, Neumann, and the Monte Carlo, Method *Los Alamos Sci.*, Vol **15**, p. 131-43.

[2] Blom, Gunnar., (1989). Probability and Statistics: theory and applications.