eman ta zabal zazu

Universidad del País Vasco   Euskal Herriko Unibertsitatea

Activity 1

1 Synopsis.

What we are set about to do. In a previous seminar we saw a simple example on how to estimate areas and integrals by simulation, using the so-called Monte Carlo method. In this activity you will put to good use your new skill to obtain an approximation of a not-so-obvious integral.

What you need. You need to be fully acquainted with the content of the Seminar 1 handout. You will also need access to a computer equiped with R.

2 Problem

Consider the function,

$$f(x) = \frac{1}{2^{k/2}\Gamma(k/2)} x^{k/2-1} e^{-x/2}. \tag{1}$$

This is the so-called chi square distribution with k degrees of freedom[1]. We need not to worry about $\Gamma(k/2)$; it will appear later in the course.

When we graph this function for $k = 3$ we obtain something like Figure 1. Notice that $f(x)$ is defined for all $x > 0$, but we only represent part of that range, since all we want is the integral from 1 to 10 (shaded area). Notice also that it integrates to 1 over $(0, \infty)$ (it is a density function), so the shaded area is of course less than 1.

You can very easily obtain values of the density. For example, for $x = 1.4$, close to the maximum, we obtain:

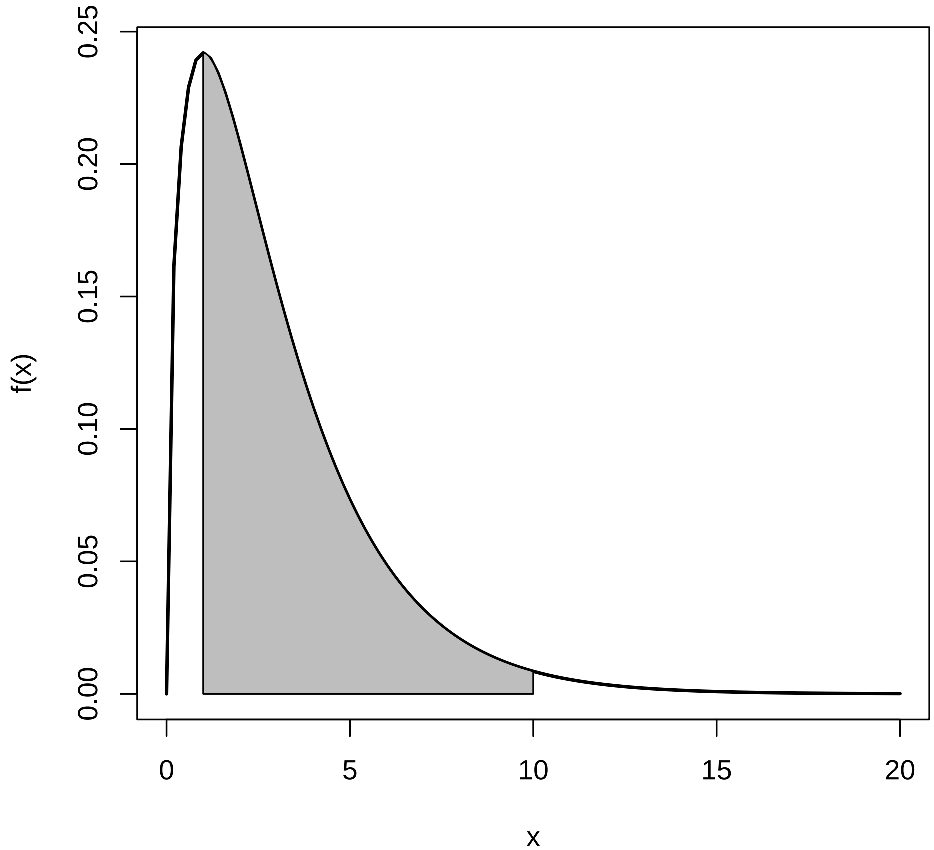
```
> dchisq(x=1.4, df=3)
```

```
[1] 0.2344056
```

while for $x = 4$ we obtain:

[1] For information on the chi square, you can turn to any statistics text or Wikipedia, https://en.wikipedia.org/wiki/Chi-squared_distribution. We will make full use of this distribution (with various degrees of freedom) in the near future.

Figure 1: Chi square density with 3 degrees of freedom (χ^2_3). The area over the region $(1, 10)$ which we want to calculate integrating the function is shaded.




```
> dchisq(x=4, df=3)
```

```
[1] 0.1079819
```

Your problem is to compute using Monte Carlo the area indicated. You can do it in various ways, some more efficient than others: look at the hints and comments below and do not just settle for the easiest solution.

Notice that for this particular function you will never need to use Monte Carlo, for function `pchisq` computes the distribution function of the chi square. The desired area could be easily obtained by:

```
> pchisq(10, df=3) - pchisq(1, df=3)
```

```
[1] 0.7826858
```

This value will serve you as a check: you should obtain something close to it.

3 Hints and comments

1. Mimic the strategy outlined in Seminar 1. The area you want is inside the "box" $B = [1, 10] \times [0, h]$; h is the largest possible value the function can take. Not much is lost if you take an upper bound, for instance (from the graph) 0.25.

   The area of the box B is: $9 \times 0.25 = 2.25$.

2. You can generate points with uniform distribution inside B quite easily: take X uniform in $[1, 10]$, Y uniform in $[0, 0.25]$ and that will give you a random (X, Y) uniformly distributed in the box.

3. Do it many times, and count in how many cases $f(X) > Y$, where $f(X)$ is the density function of the chi square with 3 degrees of freedom; these are the "hits" in the shaded area.

4. If you do it, say, for $N = 10000$ times and 30% of the time you find $f(X) > Y$, you would reason: "The box B area is 2.25; and about 30% of the time one point of the box is in the shaded area; hence, that surface is about $0.30 \times \text{area}(B) = 0.30 \times 2.25 = 0.675$."

5. Lets call the shaded area S. Your guess of the shaded area (we will soon be calling such a guess an *estimate* and using the notation $\hat{S}$ for it) is of the form:

   $$\hat{S} = \frac{\text{Hits}}{N} \times B;$$

   Now, you can easily convince yourself of the following facts:

   (a) The number of hits is binomially distributed, $b(p = S/B, N)$.

   (b) The variance of $\frac{\text{Hits}}{N}$ will therefore be pq/N. For this to be small, you want p very close to 0 or 1.

   (c) The variance of $\hat{S}$ is $B^2 \times pq/N$. For this to be small, you want B small.

   (d) From the previous two observations, if you were able to enclose the shaded area in a region (not necessarily a rectangular box) of area close to S, you would benefit twice: B would be small and $p = S/B$ would be close to 1, so pq/N would also be small.

   The idea suggests itself, then, of picking a region made of two or more boxes that encloses the target area more tightly, such as in Figure 2. We would then compute approximations to the shaded areas within each box and add them up.

6. The upper limit line has been set at 0.25 for $x \in [0, 5)$ and 0.074 for $x \in [5, 10]$ (we could use slightly tighter bounds if we like, finding what is the exact maximum of the density and the value of the density at $x = 5$).

7. All you have to do now is:

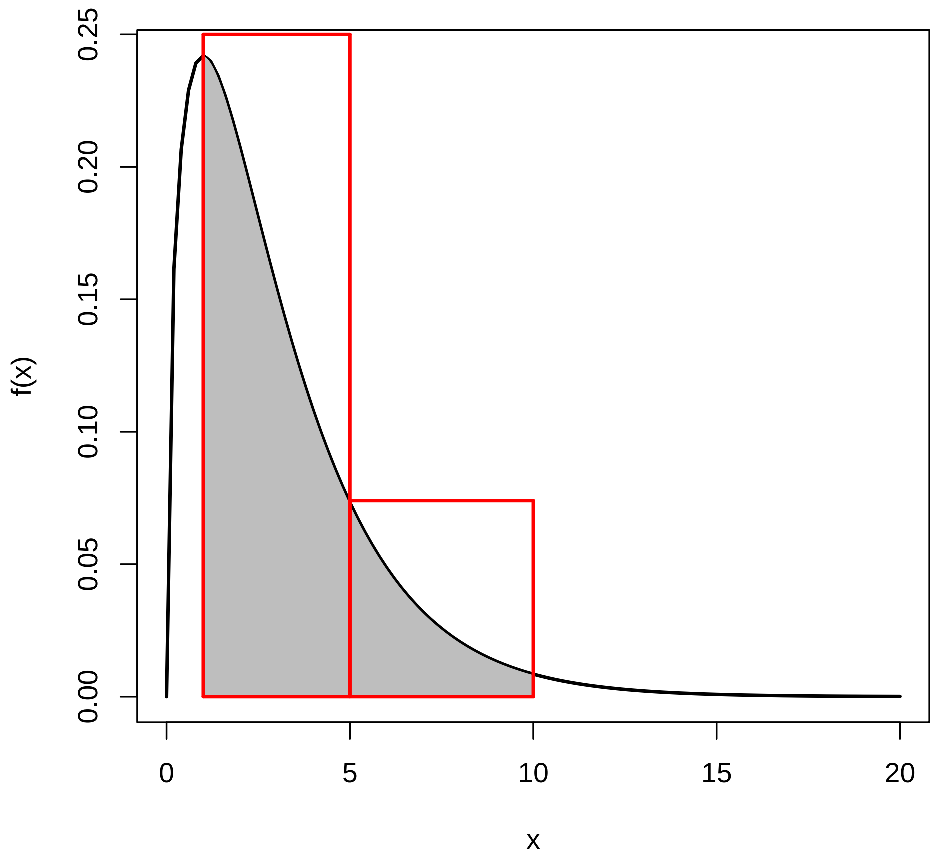
   (a) Generate uniforms in the first box to approximate the area in that box.

   (b) Do the same with the second box.

   (c) Add together the two areas.

   How many uniforms in each box? For the time being, lets assume one half for each, but we reconsider that next.

Figure 2: The targeted area is now enclosed in a non-rectangular region limited by the red lines.



8. A little more thinking shows us that most of the area is in the first box. All other things being equal, we would rather make a greater effort evaluating that area, which accounts for most of the total[2]. If we plan to generate N random points, rather than allocate half to each box we could allocate $0.75N$ to the first box and $0.25N$ to the second, roughly in proportion of their relative sizes.

9. What do you have to show for your work to be graded? A computer printout with both your code and comments interspersed, explaining what you have done *and the values effectively printed by your program*. It is not enough to say "... and running this program we get 0.5". You need to produce the code and actual output, so we can check that your program effectively calculates what you say it calculates.

[2] We will revisit this idea towards the end of the course when we deal with *stratified sampling*.