Statistical Models STAT2011

Semester 1, 2012

Week 6 Computer Exercise

Suppose we are given counts $x_1, x_2, \ldots, x_n$ modelled as observed values of independent $B(2, p)$ random variables $X_1, X_2, \ldots, X_n$. How best to use the x_i's to estimate p? We shall explore this question via simulation.

1. Set `p=0.4`.

2. We are going to perform simulations and make comments on various aspects of it. If you `process()` your work again, the numbers will change and some comments may be wrong. To avoid this you can set the random number generator to always give you the same numbers by setting the *seed*, e.g. by placing a command like

   ```
   set.seed(198456373)
   ```

   or any other large-ish integer (e.g. your SID) at this point in your file.

3. Let `s` be a simulated random sample of size `n=250` from a $B(2, \mathtt{p})$ population. You can use a command of the form `s=rbinom(...)` for this. Use `?rbinom` to see the help page. We are going to go ahead and imagine that we *didn't* know what the true p generating the data is; rather we are going to estimate it using `s`, using various methods.

4. Obtain a first estimate of p by obtaining the sample mean, equating it to its expected value and solving for p; call it `p.hat1` (be careful here: what is the expected value of the sample mean?).

5. The observed proportions of 0's, 1's and 2's should be close to the corresponding probabilities $(1-p)^2$, $2p(1-p)$ and p^2. These suggest two other (reasonable) estimates:

   (a) equate the proportion of zeroes to $(1-p)^2$ and solve for p; call this `p.hat0` (a quick way to obtain the proportion of zeroes is `prop0=mean(s==0)`, indeed this is preferable to using `table()` which can cause problems in a simulation below if any of the observed frequencies are zero);

   (b) equate the proportion of twos to p^2 and solve for p; call this `p.hat2`.

6. Print the (absolute) estimation error of each of the 3 estimates (remember, you know what the true value is, it's saved in the memory as `p`!). Note which of your three has the smallest error (you may find this changes if you re-process your file without setting the seed in part 2 above).

7. We have only performed the simulation once; we can't easily tell just from this which procedure would work best in the long term. Define vectors

```
p.hat1.sim=0
p.hat0.sim=0
p.hat2.sim=0
```

and repeat your simulation above 1000 times, saving your estimates in the corresponding vector (so each of the `p.hat*.sim` vectors should end up with 1000 numbers in them). A `for`-loop to do this would look something like:

```
for(i in 1:1000){
s=...
mean(s)=...
p.hat1.sim[i]=...
prop0=...
p.hat0.sim[i]=...
prop2=...
p.hat2.sim[i]=...
}
```

8. Determine the square-root of the average squared error of the estimates in each of the 3 vectors. Comment on which estimation procedure is better in this sense. Which is the worst?

9. Repeat the whole simulation, but this time with $p = 0.09$ (you should be able to copy-and-paste most of your earlier commands). **Note:** you should use different names for the simulated estimates, e.g. `p.hat1.sim.09`.

10. Repeat the whole simulation, but this time with $p = 0.95$, using names like `p.hat1.sim.95`.

11. Discuss how each procedure's performance changes as p changes, that is compare firstly the three average squared errors obtained using `p.hat1`, then the three for `p.hat2`, etc. For what value of p did each procedure work the best?

12. Which one would you prefer to use overall (i.e. if you didn't really know the true p)? Explain clearly why.