

Homework 4: Standard Errors of the Cat Heart

36-350, Fall 2011

Due at 11:59 pm on Tuesday, 27 September 2011

INSTRUCTIONS: Submit a single plain text file, whose name clearly includes both your Andrew ID and the assignment number. Inside the file, clearly indicate which parts of your responses go with which problems. Raw R output is not acceptable, and will be marked down accordingly; you are communicating with a human being, and need to write in a human language. Files in other formats (Word, PDF, etc.) will receive a grade of zero. Homework not submitted through Blackboard will receive a grade of zero.

Direct objective: Practice with top-down design. *Indirect objectives:* Fitting statistical models; using simulations to test statistical procedures; testing parts of the code.

In the last lab, we fit a gamma distribution to the weight of cat's hearts. We did this by adjusting the parameters so that the theoretical values of the mean and variance matched the observed, sample values. Since the mean and variance are the first two **moments** of the distribution, this is an example of the **method of moments** for estimation.

The method of moments gives a point estimate, say of a parameter. To use a point estimate, we need to know how **precise** it is, i.e., how different it would be if we repeated the experiment with new data from the same population. We often measure imprecision by the **standard error**, which is the standard deviation of the point estimates.

It is hard to find a simple formula for the standard error of most estimates. In this lab, we will see how to approximate them computationally. Since we have only one value of $\hat{\theta}$ from the data, we cannot calculate a sample standard deviation for it. However, we can draw random values from a gamma distribution using the `rgamma()` function, which works like `rexp()` from lab 1.

```
rgamma(n=35, shape=0.57, scale=15)
```

would generate a vector of 35 random values, drawn from the gamma distribution with “shape” parameter $a = 0.57$ and “scale” $s = 15$. (Alternately, one can give the argument `rate`, which is $1/s$.) By applying the estimator to random samples drawn from the distribution, we can see how much the estimates will change purely due to noise.

1. *Start with the point estimate*
 - (a) Write a function, `gamma.est`, which takes as input a vector of data values, and returns a vector containing the two estimated parameters of the gamma distribution, with components named `shape` and `scale` as appropriate. You can re-use your code from lab, or use the code from the lab solutions. If you use the solutions, say so, and modify the return value appropriately. (4)
 - (b) Verify that your function implements the appropriate formulas by showing that it matches the results in the lab solutions for all cats together, for the female cats, and for the male cats. (1)
2. *Draws from the gamma distribution*
 - (a) Generate a vector containing ten thousand random values from the gamma distribution with $a = 19$ and $s = 0.56$. What are the theoretical values of the mean and of the variance? What are their sample values? (5)
 - (b) Plot the histogram of the random values, and add the curve of the theoretical probability density function. *Hint*: see lab solutions. (5)
 - (c) Apply your `gamma.est` function to your random sample. Report the estimated parameters and how far they are from the true values. (5)
3. *Top-level function* Write a function, `gamma.est.se`, to calculate the standard error of your estimates of the gamma parameters, on simulated data drawn from the gamma distribution. It should take the following arguments: true shape parameter `shape` (or a), true scale parameter `scale` (or s), size of each sample `n`, and number of repetitions at that sample size `B`. It should return two standard errors, one for the shape parameter a and one for the scale parameter s . (These can be either in a vector or in a list, but should be named clearly.) It should call a function `gamma.est.rep` which takes the same arguments as `gamma.est.se`, and returns an array with two rows and `B` columns, one row holding shape estimates and the other row scale estimates. Your `gamma.est.se` function should *not*, itself, estimate any parameters or generate any random values. (25)
4. *Testing with a dummy function.* To check that `gamma.est.se` works properly, we write a *dummy* version of `gamma.est.rep`, which takes the correct arguments and returns an array of the proper size, but whose entries are fixed so that it's easy for us to calculate what `gamma.est.se` ought to do.
 - (a) Write `gamma.est.rep` so that the entries in the first row of the returned array alternate between `shape` and `shape+1`, and those in the second row alternate between `scale` and `scale+n`. It should match the following, except for the row names, which are optional. (5)

```

> gamma.est.rep(2,1,10,10)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
shapes    2    3    2    3    2    3    2    3    2    3
scales    1   11    1   11    1   11    1   11    1   11
> gamma.est.rep(2,8,5,7)
      [,1] [,2] [,3] [,4] [,5] [,6] [,7]
shapes    2    3    2    3    2    3    2
scales    8   13    8   13    8   13    8

```

- (b) Calculate the standard deviations of each row in the two arrays above. (2)
- (c) Run your `gamma.est.se`, with this version of `gamma.est.rep`. Do its standard errors match the standard deviations you just calculated? Should they? (8)

5. Replacing the dummy

- (a) Write the actual `gamma.est.rep`. Each of the `B` columns in its output should be the result of applying `gamma.est` to a vector of `n` random numbers generated by a different call to `rgamma`, all with the same `shape` and `scale` parameters. For full credit, use `replicate` rather than looping. *Hint*: Look at lecture 3, towards the end. (20)
- (b) Run `gamma.est.se`, calling your new `gamma.est.rep`, with `shape=2`, `scale=1`, `n=10` and `B=1e5`. Check that the standard error for `shape` is approximately 1.6 and that for `scale` approximately 0.54. Explain why this problem cannot give exact control values. (5)

6. Using the standard errors

- (a) Run `gamma.est.se` with the `shape`, `scale`, and sample size (`n`) parameters appropriate to the female cats, and `B=1e5`. Report both standard errors. (5)
- (b) Repeat the previous problem with the male cats. (5)
- (c) The standard error of a difference, $d_1 - d_2$, is $\sqrt{se_1^2 + se_2^2}$. Calculate the standard errors for the two differences in parameter estimates between the two sexes of cat, and the ratio of the differences to their standard errors. (5)