

Lab 3: Of Big- and Small- Hearted Cats

36-350, Statistical Computing

SOLUTIONS

1. SOLUTION: The output is

```
> summary(cats)
Sex      Bwt      Hwt
F:47   Min.    :2.000   Min.    : 6.30
M:97   1st Qu.:2.300   1st Qu.: 8.95
       Median :2.700   Median :10.10
       Mean   :2.724   Mean   :10.63
       3rd Qu.:3.025   3rd Qu.:12.12
       Max.   :3.900   Max.   :20.50
```

Running `summary()` on a data frame runs it separately on each column. `Sex` is a categorical variable (“factor”), so it gives the count of each level or value: there are 47 female and 97 male cats. For numerical variables, it calculate the minimum value, the first quartile, the median, the mean, the third quartile, and the maximum. `Hwt` is (from the name) heart weight, so `Bwt` must be total or body weight. It may look peculiar that the *smallest* number for heart weight is less than the *largest* for body weight, since a heart can’t very well weigh more than the whole cat, but remember `Hwt` is in grams and `Bwt` is in kilograms.

2. SOLUTION:

```
> mean(cats$Hwt)
[1] 10.63056
> sd(cats$Hwt)
[1] 2.434636
> var(cats$Hwt)
[1] 5.927451
```

In words: a mean heart weight of 10.6 grams, a standard deviation of 2.43 grams, and a variance of 5.93 grams².

(By examining the values in the data, we see that `Hwt` is never recorded to more than three significant digits, so reporting more digits in our calculations from that data is spurious precision¹.

¹Wikipedia has a good entry on “significant digits” if this concept is unfamiliar. The R command `signif(x,digits)` rounds `x` to the specified number of digits.

3. SOLUTION: Since the mean $\mu = as$ and the variance $v = as^2$,

$$\frac{v}{\mu} = \frac{as^2}{as} = s$$

4. SOLUTION: Since the mean $\mu = as$,

$$\frac{\mu}{s} = \frac{as}{s} = a$$

5. SOLUTION: Introducing variables at this point is optional, but may make things clearer.

```
> heart.mean <- mean(cats$Hwt)
> heart.var <- var(cats$Hwt)
> s.est <- heart.var/heart.mean
> a.est <- heart.mean/s.est
> s.est
[1] 0.5575862
> a.est
[1] 19.06531
```

so our estimate is that s is 0.558 grams, and a is 19.1 (with no units since a is a dimensionless number).

6. SOLUTION:

```
gamma.est <- function(data) {
  m <- mean(data)
  v <- var(data)
  s <- v/m
  a <- m/s
  return(list(a=a,s=s))
}
```

It would also be acceptable to return a two-component vector, to do sanity-checking on the inputs, etc.

7. SOLUTION: Run the function with the given data:

```
> gamma.est(cats$Hwt)
$a
[1] 19.06531

$s
[1] 0.5575862
```

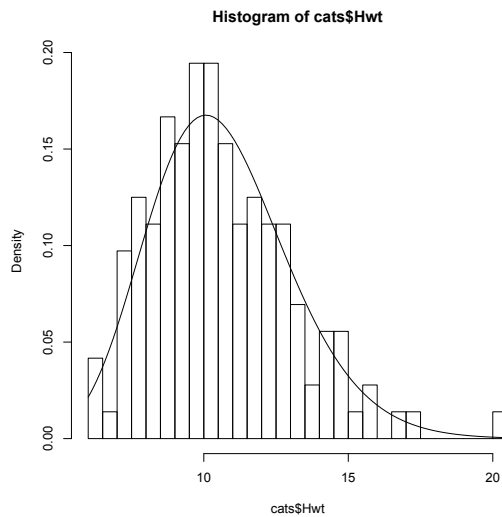


Figure 1: Histogram of cat heart weights (bars), and fitted gamma density (curve).

This agrees with our answer in question 5 as it should — it’s the same calculation on the same data, just wrapped up in a function. To be *really* sure of this:

```
> identical(list(a=a.est,s=s.est), gamma.est(cats$Hwt))
[1] TRUE
```

8. SOLUTION: First, we plot the histogram, setting it to use probabilities (rather than total counts or frequencies), and a reasonably large number of bins. The number of bins is so it looks nicer, but the probabilities are for comparison with the probability density function.

```
hist(cats$Hwt,probability=TRUE,n=31)
```

Now we add the curve:

```
curve(dgamma(x,shape=a.est,scale=s.est),add=TRUE)
```

This tells it to plot the curve of the `dgamma` function, sweeping over `x`, with our estimated parameters. By default, `curve` creates a new plot, wiping out the old one, but now we want to add to the previous one. Figure 1 shows the result.

9. Estimate the a and s separately for all the male cats and all the female cats, using `gamma.est`. Give the commands you used and the results. (10)
SOLUTION: Use comparisons to pick out the two subsets of the data by sex:

```
f.est <- gamma.est(cats$Hwt[cats$Sex=="F"])
m.est <- gamma.est(cats$Hwt[cats$Sex=="M"])
```

Alternatively, you could look at the whole data frame and see that all of the female cats come before all of the male cats, so

```
f.est <- gamma.est(cats$Hwt[1:47])
m.est <- gamma.est(cats$Hwt[48:144])
```

Using the condition is better, since it makes it clearer what you are doing. The estimates are

```
> f.est
$a
[1] 45.93998

$s
[1] 0.2003076

> m.est
$a
[1] 19.83576

$s
[1] 0.5708216
```

10. SOLUTION: The numbers don't look particularly close, but that's not a great standard — how do you know whether a difference of ≈ 36 in a is big or small? Better approaches would be to try plotting the corresponding probability density functions and seeing how far apart the two curves are, or trying to get some standard errors. For instance, we might divide the cats into two groups *at random*, keeping the number of cats at 47 and 97, and see how different the parameter estimates we get for the two groups are. You did not have to actually work through an answer here, however.