

431 Part A: Extra Example

Thomas E. Love

2017-11-09

Sleep and Mammals

This example reviews some of the key work we did in Part A of the course. It uses the `msleep` data set, which is part of the `ggplot2` package. Use the help file for that data set or visit <http://ggplot2.tidyverse.org/reference/msleep.html> to assist in understanding the data.

1. Identify the number of rows and number of variables in the `msleep` data set.

```
dim(msleep)
```

```
[1] 83 11
```

There are 83 rows and 11 variables.

2. Specify the variable with the largest number of missing values in `msleep`. How many values are missing?

```
summary(msleep)
```

name	genus	vore
Length:83	Length:83	Length:83
Class :character	Class :character	Class :character
Mode :character	Mode :character	Mode :character

order	conservation	sleep_total	sleep_rem
Length:83	Length:83	Min. : 1.90	Min. :0.100
Class :character	Class :character	1st Qu.: 7.85	1st Qu.:0.900
Mode :character	Mode :character	Median :10.10	Median :1.500
		Mean :10.43	Mean :1.875
		3rd Qu.:13.75	3rd Qu.:2.400
		Max. :19.90	Max. :6.600
			NA's :22

sleep_cycle	awake	brainwt	bodywt
Min. :0.1167	Min. : 4.10	Min. :0.00014	Min. : 0.005
1st Qu.:0.1833	1st Qu.:10.25	1st Qu.:0.00290	1st Qu.: 0.174
Median :0.3333	Median :13.90	Median :0.01240	Median : 1.670
Mean :0.4396	Mean :13.57	Mean :0.28158	Mean :166.136
3rd Qu.:0.5792	3rd Qu.:16.15	3rd Qu.:0.12550	3rd Qu.: 41.750
Max. :1.5000	Max. :22.10	Max. :5.71200	Max. :6654.000
NA's :51		NA's :27	

`sleep_cycle` is missing for 51 of the 83 mammals.

3. Identify the mammal who remains awake the longest, per day. What is the Z score for this mammal?

Let's calculate the z scores, and then sort by the awake value.

```
msleep %>% select(name, awake) %>%  
  mutate(z_awake = round((awake - mean(msleep$awake))/sd(msleep$awake),2)) %>%  
  arrange(desc(awake))
```

```
# A tibble: 83 x 3  
  name awake z_awake  
  <chr> <dbl> <dbl>  
1 Giraffe 22.10 1.92  
2 Pilot whale 21.35 1.75  
3 Horse 21.10 1.69  
4 Roe deer 21.00 1.67  
5 Donkey 20.90 1.65  
6 African elephant 20.70 1.60  
7 Caspian seal 20.50 1.56  
8 Sheep 20.20 1.49  
9 Asian elephant 20.10 1.47  
10 Cow 20.00 1.44  
# ... with 73 more rows
```

It's the giraffe, whose Z score is 1.92

4. Display your R code to create a data set (called sleep2) from msleep which contains the variables name, order, vore, sleep_cycle and sleep_rem for those animals who have no missing values in any of those four variables, then convert the vore information into a factor.

```
sleep2 <- msleep %>%  
  select(name, order, vore, sleep_cycle, sleep_rem) %>%  
  filter(complete.cases(name, order, vore, sleep_cycle, sleep_rem)) %>%  
  mutate(vore = factor(vore))
```

```
sleep2
```

```
# A tibble: 31 x 5  
  name order vore sleep_cycle sleep_rem  
  <chr> <chr> <fctr> <dbl> <dbl>  
1 Greater short-tailed shrew Soricomorpha omni 0.1333333 2.3  
2 Cow Artiodactyla herbi 0.6666667 0.7  
3 Three-toed sloth Pilosa herbi 0.7666667 2.2  
4 Northern fur seal Carnivora carni 0.3833333 1.4  
5 Dog Carnivora carni 0.3333333 2.9  
6 Guinea pig Rodentia herbi 0.2166667 0.8  
7 Chinchilla Rodentia herbi 0.1166667 1.5  
8 Lesser short-tailed shrew Soricomorpha omni 0.1500000 1.4  
9 Long-nosed armadillo Cingulata carni 0.3833333 3.1  
10 North American Opossum Didelphimorphia omni 0.3333333 4.9  
# ... with 21 more rows
```

5. According to the order variable, how many Primates (see the order variable) exist in your sleep2 data set?

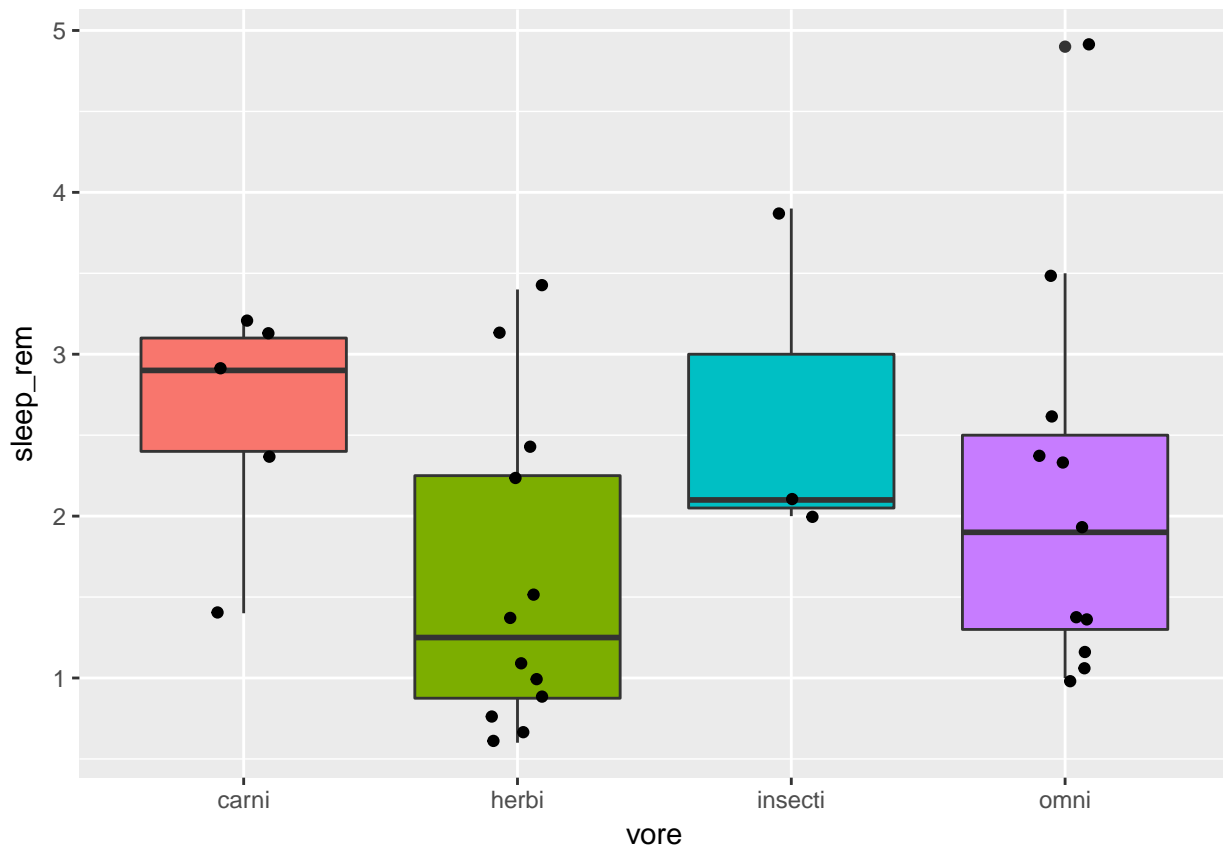
```
sleep2 %>% filter(order == "Primates")
```

```
# A tibble: 5 x 5
  name    order vore sleep_cycle sleep_rem
  <chr>   <chr> <fctr>    <dbl>     <dbl>
1  Galago Primates omni    0.5500000     1.1
2   Human Primates omni    1.5000000     1.9
3 Macaque Primates omni    0.7500000     1.2
4 Chimpanzee Primates omni    1.4166667     1.4
5  Baboon Primates omni    0.6666667     1.0
```

There should be five.

6. Draw a plot to compare the sleep_rem levels by vore group using your sleep2 data. What do you conclude?

```
ggplot(sleep2, aes(x = vore, y = sleep_rem, fill = vore)) +
  geom_boxplot() +
  geom_jitter(width = 0.1) +
  guides(fill = FALSE)
```



In general, carnivores have the longest sleep_rem while herbivores have the shortest, at least in terms of

medians. There are very few values (looks like just three) in the insectivores, and it looks like just five in the carnivores.

7. Produce R code using the `%>%` pipe to produce a table which answers these two questions for the `sleep2` data: [a] Which vore group has the largest mean `sleep_rem` level? [b] Which vore group has the largest mean `sleep_cycle`?

```
sleep2 %>% group_by(vore) %>% summarize(n(), mean(sleep_rem), mean(sleep_cycle))
```

```
# A tibble: 4 x 4
  vore `n()` `mean(sleep_rem)` `mean(sleep_cycle)`
  <fctr> <int>         <dbl>         <dbl>
1  carni     5         2.600000         0.3733333
2  herbi    12         1.591667         0.4180556
3  insecti   3         2.666667         0.1611111
4   omni    11         2.154545         0.5924242
```

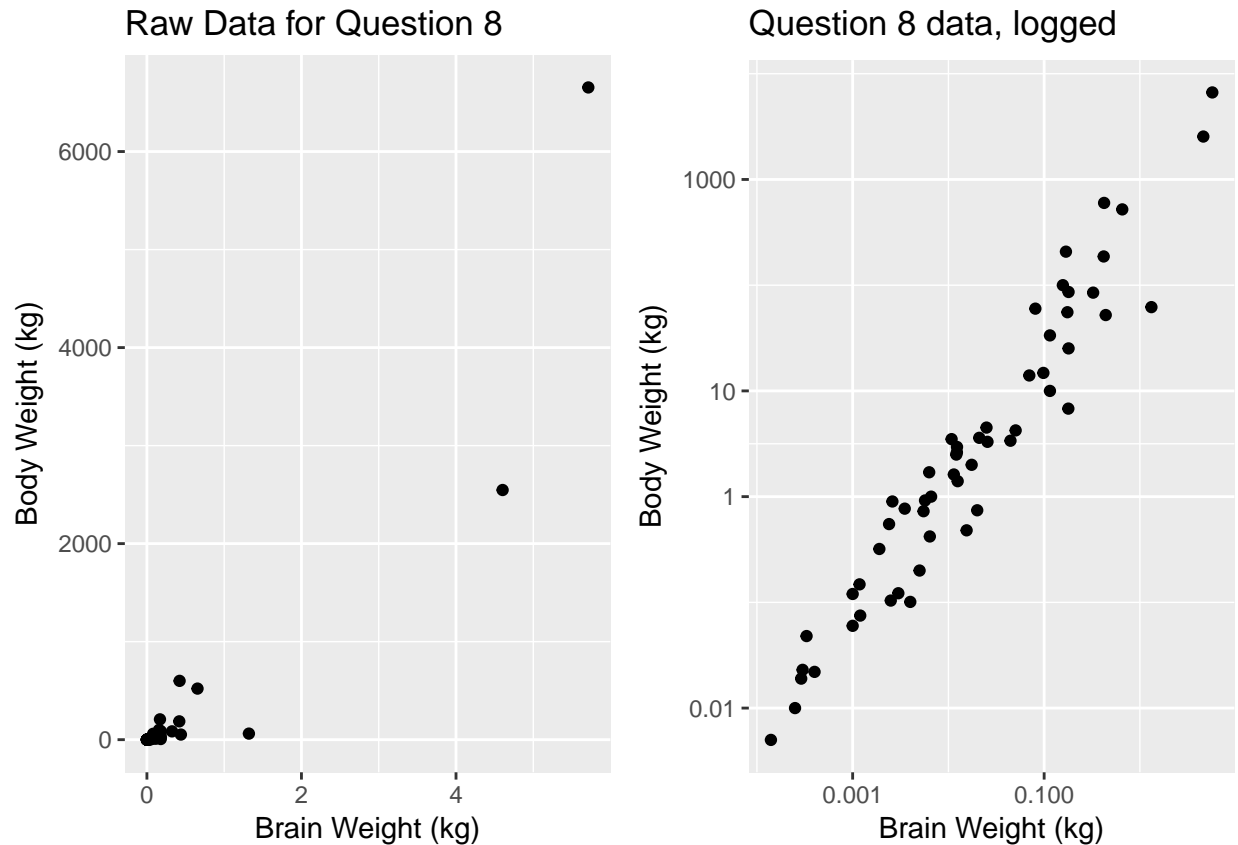
8. Now, return to the original `msleep` data for questions 8-10. Build a scatterplot of `brainwt` and `bodywt`, first using the raw data and then using a logarithmic scale for each variable.

```
sleep1 <- msleep %>%
  filter(complete.cases(brainwt, bodywt))

p1 <- ggplot(sleep1, aes(x = brainwt, y = bodywt)) +
  geom_point() +
  labs(title = "Raw Data for Question 8",
       x = "Brain Weight (kg)",
       y = "Body Weight (kg)")

p2 <- ggplot(sleep1, aes(x = brainwt, y = bodywt)) +
  geom_point() +
  scale_y_log10(breaks = c(0.01, 1, 10, 1000), labels = c(0.01, 1, 10, 1000)) +
  scale_x_log10() +
  labs(title = "Question 8 data, logged",
       x = "Brain Weight (kg)",
       y = "Body Weight (kg)")

gridExtra::grid.arrange(p1, p2, nrow = 1)
```



9. Fit a linear model to the scatterplot you drew in part 8 for which a linear model seems more appropriate, and specify and describe (in a sentence of two) both the fitted least squares equation *and* the Pearson correlation coefficient.

Clearly the log-log model is the better choice. I'll revert to a natural logarithm for the plot below, and the model.

```
model9 <- lm(log(bodywt) ~ log(brainwt), data = sleep1)
summary(model9)
```

Call:

```
lm(formula = log(bodywt) ~ log(brainwt), data = sleep1)
```

Residuals:

Min	1Q	Median	3Q	Max
-2.0902	-0.4282	0.1054	0.5723	1.6211

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	5.87919	0.21393	27.48	<2e-16 ***
log(brainwt)	1.21785	0.04482	27.17	<2e-16 ***

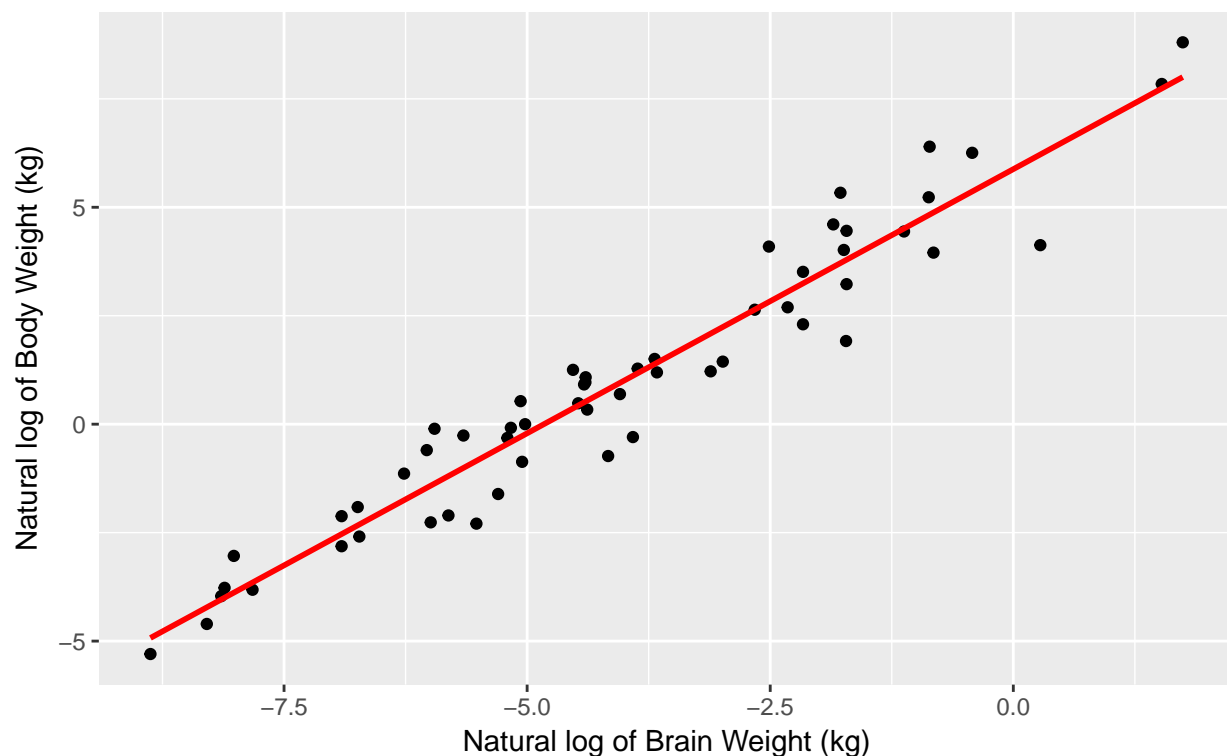
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8403 on 54 degrees of freedom
 Multiple R-squared: 0.9319, Adjusted R-squared: 0.9306
 F-statistic: 738.4 on 1 and 54 DF, p-value: < 2.2e-16

```
ggplot(sleep1, aes(x = log(brainwt), y = log(bodywt))) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, col = "red") +
  labs(title = "Question 8 data: Natural Logs, with Linear Model",
        subtitle = "log(Body Weight) = 5.88 + 1.22 log(Brain Weight)",
        x = "Natural log of Brain Weight (kg)",
        y = "Natural log of Body Weight (kg)")
```

Question 8 data: Natural Logs, with Linear Model

$\log(\text{Body Weight}) = 5.88 + 1.22 \log(\text{Brain Weight})$



10. Identify the mammal in your model (in part 9) with the largest (in absolute value) regression residual.

```
sleep1$abs_res <- abs(model9$residuals)

sleep1 %>% select(name, abs_res) %>% arrange(desc(abs_res))
```

A tibble: 56 x 2

	name	abs_res
	<chr>	<dbl>
1	Human	2.090169
2	Macaque	1.867117

```
3           Brazilian tapir 1.621108
4           Cow 1.565557
5           Owl monkey 1.538485
6 Thirteen-lined ground squirrel 1.447518
7           Squirrel monkey 1.411995
8           Giant armadillo 1.275983
9           Tenrec 1.264385
10          Galago 1.036076
# ... with 46 more rows
```

The answer is Human.