# Mammalian Sleep Data Analysis

Bin Han

2025-06-24

### Project discription

**In this project we will explore the sleep times of different mammals. The dataset is called msleep and comes with the ggplot2 package (which in turn comes with tidyverse). All visualizations are made with `ggplot`.**

### Running Code

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.2     v tibble    3.2.1
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.0.4
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to beco
```

```
library(dplyr)
library(ggplot2)
data(msleep)
```

1. The variables `genus`, `vore`, `order` and `conservation` are of character type. Convert them into factors. **Convert the character variable to the factors.**

```
msleep%>%
  mutate(
    genus=as.factor(genus),
    vore=as.factor(vore),
    order=as.factor(order),
    conservation=as.factor(conservation)
  )%>%
  head()
```

```
# A tibble: 6 x 11
  name    genus vore  order conservation sleep_total sleep_rem sleep_cycle awake
  <chr>   <fct> <fct> <fct> <fct>               <dbl>     <dbl>       <dbl> <dbl>
1 Cheetah Acin~ carni Carn~ lc                   12.1      NA        NA      11.9
2 Owl mo~ Aotus omni  Prim~ <NA>                 17         1.8      NA       7
3 Mounta~ Aplo~ herbi Rode~ nt                   14.4       2.4      NA       9.6
4 Greate~ Blar~ omni  Sori~ lc                   14.9       2.3       0.133   9.1
5 Cow     Bos   herbi Arti~ domesticated         4         0.7       0.667  20
6 Three-~ Brad~ herbi Pilo~ <NA>                 14.4       2.2       0.767   9.6
# i 2 more variables: brainwt <dbl>, bodywt <dbl>
```

2. Store the sleep time of this animal in a numeric variable called `shortest_sleep` and the name of the animal in a character/string variable called `shortest_sleep_mammal`. **Giraffe has the shortest sleep 1.9h**

```
shortest_sleep<-min(msleep$sleep_total)

msleep%>%
     filter(sleep_total==shortest_sleep, na.rm=TRUE)%>%
     pull(name)%>%
     as.character()
```

```
[1] "Giraffe"
```

3. Store the name of the variable in a character/string variable called `most_missing` and the number of missing values (for that same variable) in a numeric variable called `missing_values`. **sleep_cycle column has the most_missing, 51 missing**

```
na_summary<-msleep%>%
  summarise(across(everything(), ~sum(is.na(.)))) %>%
  pivot_longer(everything(), names_to="variable", values_to="na_count")

most_missing<-na_summary %>%
```

```
  filter(na_count==max(na_count))%>%
  pull(variable)%>%
  as.character()

max(na_summary$na_count)
```

```
[1] 51
```

4. Compute the correlations between all numeric variables in the dataset. Save the result in a variable called `correlations`. **correlations matrix**

```
numeric_msleep<- msleep %>% select(where(is.numeric))
correlations<- cor(numeric_msleep, use="pairwise.complete.obs")
head(correlations)
```

```
             sleep_total  sleep_rem sleep_cycle      awake    brainwt     bodywt
sleep_total   1.0000000  0.7517550  -0.4737127 -0.9999986 -0.3604874 -0.3120106
sleep_rem     0.7517550  1.0000000  -0.3381235 -0.7517713 -0.2213348 -0.3276507
sleep_cycle  -0.4737127 -0.3381235   1.0000000  0.4737127  0.8516203  0.4178029
awake        -0.9999986 -0.7517713   0.4737127  1.0000000  0.3604874  0.3119801
brainwt      -0.3604874 -0.2213348   0.8516203  0.3604874  1.0000000  0.9337822
bodywt       -0.3120106 -0.3276507   0.4178029  0.3119801  0.9337822  1.0000000
```

5. Copy the correlation matrix from Task 4 to a new variable, find the second most correlated variables, and save the correlation in a numeric variable called `highest_corr`. **Brain weight and body weight has the highest correlations 0.934.**

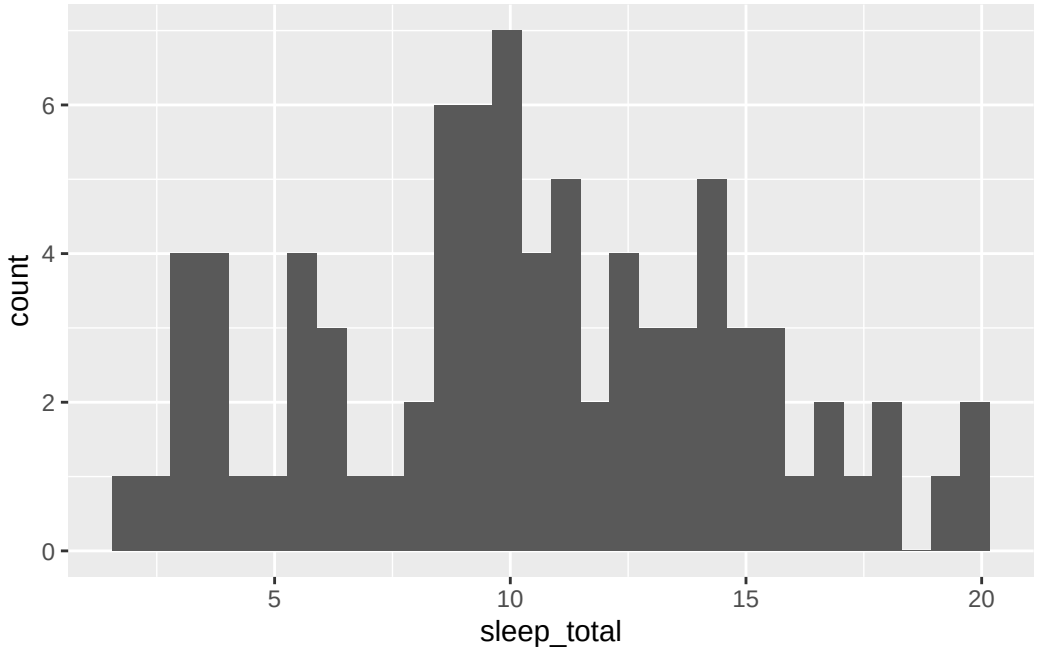
```
correlations_copy<-correlations

diag(correlations_copy)<-NA
correlations_copy["sleep_total", "awake"] <-NA
correlations_copy["awake", "sleep_total"] <-NA

highest_corr<-max(correlations_copy, na.rm=TRUE)
highest_corr
```

```
[1] 0.9337822
```

6. Create a histogram for the `sleep_total` variable. Save the histogram in a variable called `sleep_histogram`. **Total sleep time frequency**

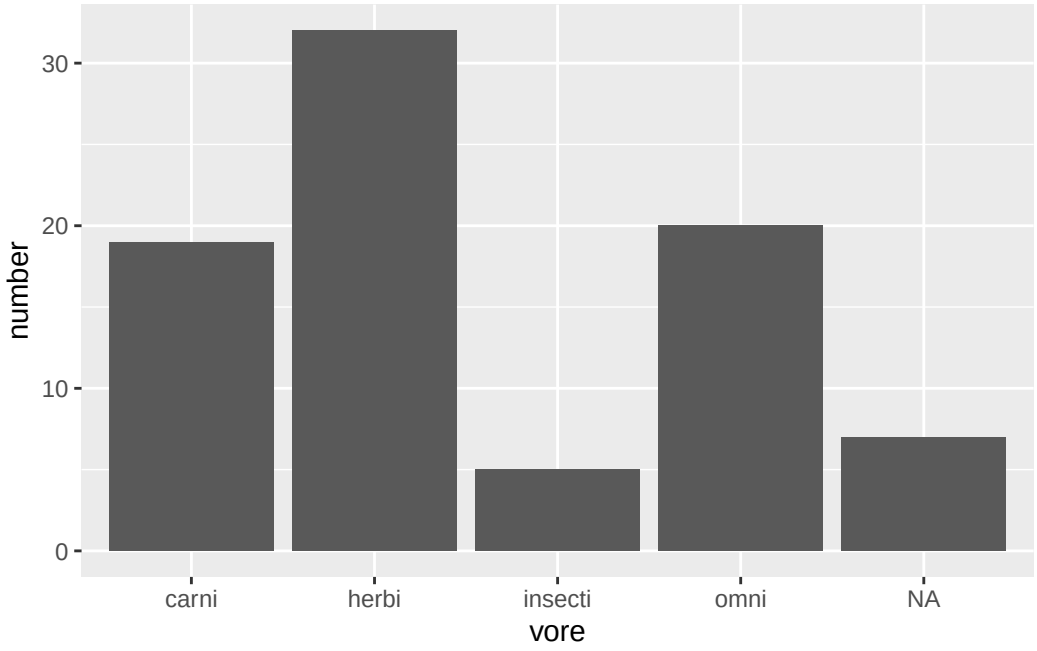
```
sleep_histogram <-ggplot(msleep, aes(x=sleep_total))+geom_histogram()
sleep_histogram
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



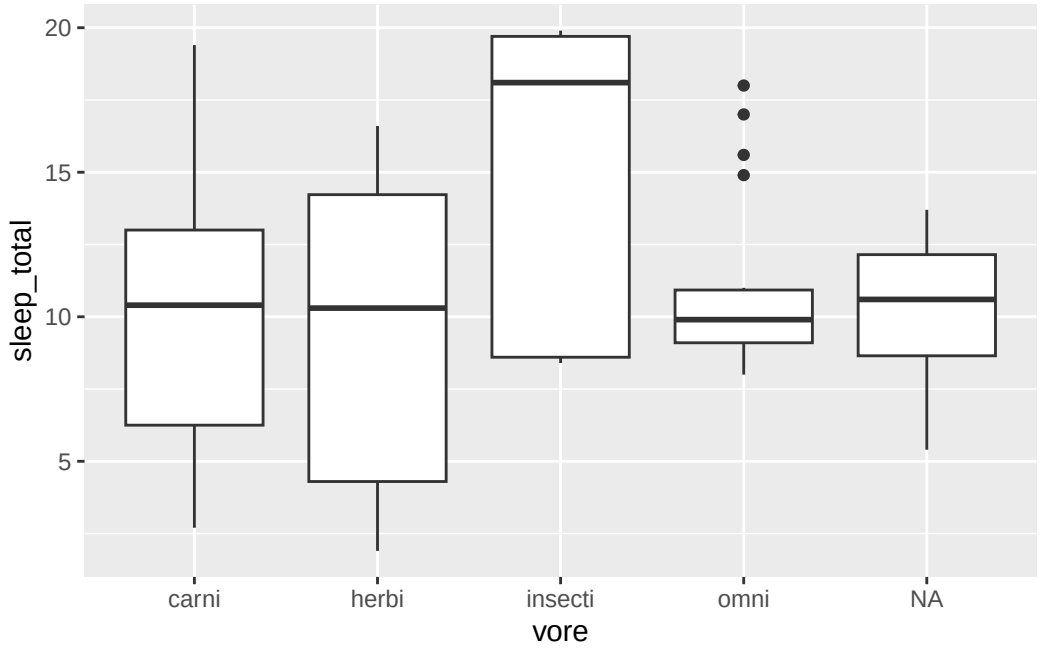

7. Different mammals eat different things. We have data on which animals are herbivores, carnivores, omnivores, insectivores (some mammals are also missing this type of data). Create a bar chart with each bar containing the number of mammals in each food category. Store the plot in a variable called `food_barchart`. **Bar chart for the number of mammals in each food category**

```
food_barchart<-msleep%>%
  group_by(vore)%>%
  summarize(number=n())%>%
  ggplot(aes(x=vore, y=number))+
  geom_bar(stat="identity")
food_barchart
```




8. Create a boxplot for the `sleep_total` variable, grouped so that there is one box for each food category. **bar plot for the sleep time for different food category**

```
sleep_boxplot<-msleep %>%
  ggplot(aes(x=vore, y=sleep_total))+
  geom_boxplot()
sleep_boxplot
```



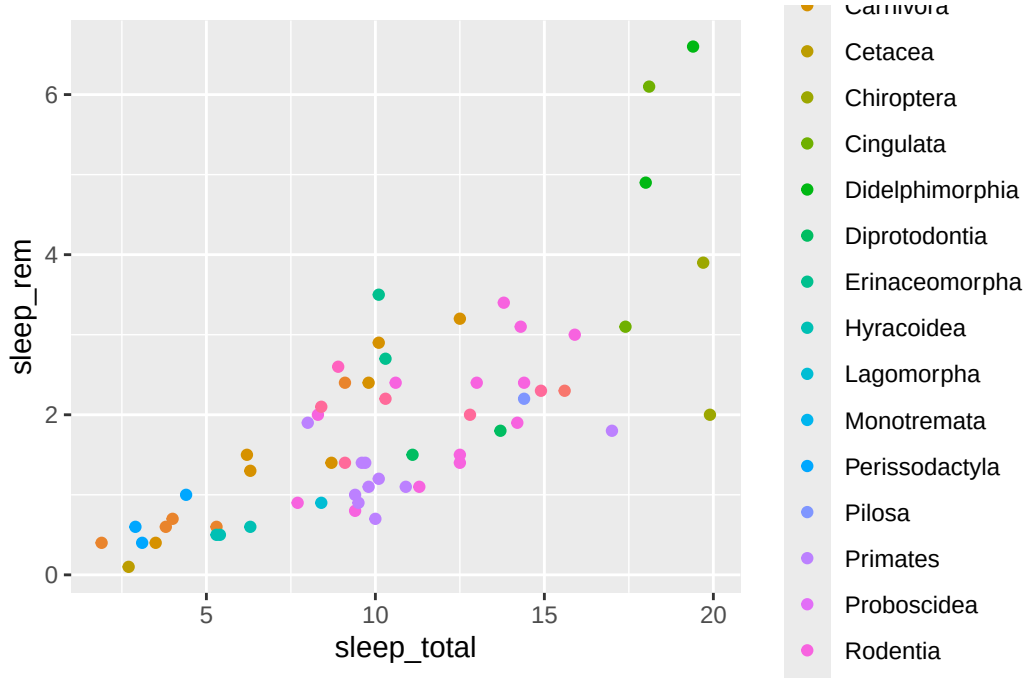

9. For which food category do the mammals have the longest average sleep time? Store the average time for this group in a variable called `highest_average`. **insecti have the longest average sleep time 14.9 h.**

```
highest_vore<- msleep %>%
  group_by(vore)%>%
  summarize(avg_sleep=sum(sleep_total)/n(),.groups="drop") %>%
  arrange(desc(avg_sleep)) %>%
  slice(1)
highest_average<-highest_vore$avg_sleep
highest_average
```

```
[1] 14.94
```

10. Do mammals who sleep more also have more rem sleep? Create a scatterplot plotting total sleep time against rem sleep time with rem sleep on the y axis. Note that we we also have different orders of animals. These have latin names such as Rodentia, Artiodactyla, etc… color the points by the order of each animal and store the resulting plot in a variable called `sleep_scatterplot`. **For different orders of animals, the longer the total sleep time, the longer the rem sleep time.**

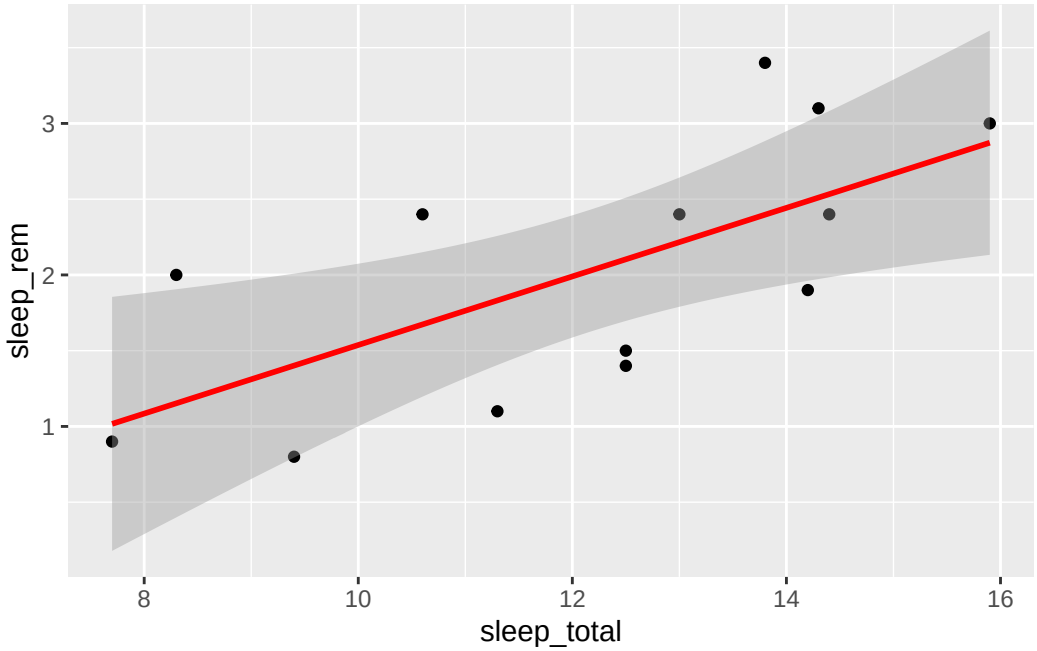
```
sleep_scatterplot<-msleep %>%
  ggplot(aes(x=sleep_total, y=sleep_rem, color=order))+
  geom_point(na.rm=TRUE)
sleep_scatterplot
```




11. Using only data for the most common order (most prevalent in the data), create another scatterplot plotting their total sleep time against their rem sleep. Again, put rem sleep on the y axis and save the resulting plot in a variable called `sleep_scatterplot2`. **For rodentia, the longer the total sleep time, the longer the rem sleep time.**

```
sleep_scatter1<-msleep %>%
  group_by(order) %>%
  summarize(n=n(), .groups="drop")%>%
  arrange(desc(n))%>%
  slice(1)
sleep_scatter<-as.character(sleep_scatter1$order)
sleep_scatterplot2<-msleep %>%
  filter(order==sleep_scatter) %>%
  drop_na(sleep_total, sleep_rem) %>%
  ggplot(aes(x=sleep_total, y=sleep_rem))+
  geom_point(na.rm=TRUE) +
  stat_smooth(method="lm", col="red")
sleep_scatterplot2
```

```
`geom_smooth()` using formula = 'y ~ x'
```


sleep_rem
3
2
1
8
10
12
14
16
sleep_total