# clinchy-et-al_2016.R

shony

2022-05-25

```r
# summary stats for clinchy et al. 2016

setwd("C:\\Users\\shony\\Nextcloud\\Work\\PhD\\Thesis\\review\\Analysis\\scripts-lit")

data <- read.csv("../data-lit/clinchy_et_al_2016_data.csv")

# libraries

library(tidyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
# data cleaning

data <- data%>%
  gather(c(latency:individuals), key = "vars", value = "value")%>%
  mutate(value = as.numeric(value))
```

```
## Warning in mask$eval_all_mutate(quo): NAs introduced by coercion
```

```r
# transforming data as per methods

library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```r
## function for box cox transformation

bc_trans <- function(data) {

  bc <- boxcox(data$value ~ data$treatment, plotit = F)
```

```r
  lambda <- bc$x[which.max(bc$y)]

  bc_value <- (data$value^lambda - 1)/lambda

  return(bc_value)
}

## function to calculate lambda

lam <- function(data) {

  bc <- boxcox(data$value ~ data$treatment, plotit = F)

  lambda <- bc$x[which.max(bc$y)]

  return(lambda)
}

## tranforming data

library(purrr)

data <- data%>%
  group_by(vars)%>%
  nest()%>%
  mutate(bc_value = map(data, ~bc_trans(.)),
         lambda = map(data, ~lam(.)))%>%
  unnest()%>%
  ungroup()
```

```
## Warning: `cols` is now required when using unnest().
## Please use `cols = c(data, bc_value, lambda)`
```

```r
# data summary - transformed

## means

data%>%
  group_by(vars, treatment)%>%
  summarise(mean = mean(bc_value, na.rm = T),
            sd = sd(bc_value, na.rm = T),
            n = n(),
            lambda = last(lambda),
            .groups = "drop")%>%
  # inverse box cox trasnform
  mutate(mean = exp(log(1 + lambda*mean)/lambda),
         sd = exp(log(1 + lambda*sd)/lambda),
         se = sd/sqrt(n))%>%
  dplyr::select(treatment, vars, mean)%>%
  spread(vars, mean)
```

```
## # A tibble: 5 x 6
##   treatment foraging individuals latency vigilance visits
##   <chr>        <dbl>       <dbl>   <dbl>     <dbl>  <dbl>
```

```
## 1 Bear           124.          3.24     87.6      21.0   3.93
## 2 Dog            121.          3.62     73.8      17.9   4.16
## 3 Human          126.          2.20    140.       13.8   2.80
## 4 Sheep          130.          3.62     27.4      17.6   4.88
## 5 Wolf           123.          2.95     35.7      23.3   3.31
```

```r
data%>%
  group_by(vars, treatment)%>%
  summarise(mean = mean(bc_value, na.rm = T),
            sd = sd(bc_value, na.rm = T),
            n = n(),
            lambda = last(lambda),
            .groups = "drop")%>%
  # inverse box cox trasnform
  mutate(mean = exp(log(1 + lambda*mean)/lambda),
         sd = exp(log(1 + lambda*sd)/lambda),
         se = sd/sqrt(n))%>%
  dplyr::select(treatment, vars, se)%>%
  spread(vars, se)
```

```
## # A tibble: 5 x 6
##   treatment foraging individuals latency vigilance visits
##   <chr>        <dbl>       <dbl>   <dbl>     <dbl>  <dbl>
## 1 Bear          21.7       0.794    2.10      1.33  0.810
## 2 Dog           26.6       0.662    4.18      2.90  0.935
## 3 Human         29.7       0.874    2.73      2.17  0.902
## 4 Sheep         17.8       0.662    3.32      1.34  0.706
## 5 Wolf          26.6       0.655    4.17      2.57  0.821
```

## *Data summary - raw*

## *mean*

```r
data%>%
  group_by(vars, treatment)%>%
  summarise(mean = mean(value, na.rm = T),
            sd = sd(value, na.rm = T),
            n = n(),
            se = sd/sqrt(n))%>%
  dplyr::select(treatment, vars, mean)%>%
  spread(vars, mean)
```

```
## `summarise()` has grouped output by 'vars'. You can override using the `.groups` argument.
```

```
## # A tibble: 5 x 6
##   treatment foraging individuals latency vigilance visits
##   <chr>        <dbl>       <dbl>   <dbl>     <dbl>  <dbl>
## 1 Bear         123.         3.33    89.1      22.3   4.29
## 2 Dog          120.         3.67    78.4      22.4   4.71
## 3 Human        123          2.33   143.       16.6   3.29
## 4 Sheep        130.         3.67    30.7      18.9   5.14
## 5 Wolf         121          3       40.3      27.3   3.71
```

```
## se

data%>%
  group_by(vars, treatment)%>%
  summarise(mean = mean(value, na.rm = T),
            sd = sd(value, na.rm = T),
            n = n(),
            se = sd/sqrt(n))%>%
  dplyr::select(treatment, vars, se)%>%
  spread(vars, se)
```

```
## `summarise()` has grouped output by 'vars'. You can override using the `.groups` argument.
```

```
## # A tibble: 5 x 6
##   treatment foraging individuals latency vigilance visits
##   <chr>        <dbl>       <dbl>   <dbl>     <dbl>  <dbl>
## 1 Bear          5.09       0.577    9.14      3.95  0.714
## 2 Dog           8.24       0.436   13.7       7.44  1.04
## 3 Human        10.5        0.577   16.6       4.84  1.02
## 4 Sheep         3.20       0.436    8.21      3.41  0.769
## 5 Wolf          7.91       0.378   12.1       7.28  0.969
```

```
# plot

library(ggplot2)

data%>%
  ggplot(aes(treatment, value))+
  geom_boxplot()+
  facet_wrap(~vars, scales = "free_y")
```

```
## Warning: Removed 20 rows containing non-finite values (stat_boxplot).
```