Final Exam Review

BIOL 305

2025-12-08

Introduction

These are questions that are posed as they would be on a final. Please complete each part of each question; we will review the answers. The following data are either imaginary or pulled from publicly available sources like Wikipedia.

For each question:

- Determine which test should be used
- Write your null and alternative hypotheses
- Perform the appropriate test
- Evaluate the results using an $\alpha = 0.05$
- Write a conclusion that relates back to the biological hypothesis
- If needed, plot the data - Don't forget a title and/or detailed caption and appropriate axes names

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.6
## v forcats   1.0.1     v stringr   1.6.0
## v ggplot2   4.0.1     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.2.0
## -- 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 become errors
```

```
library(UNKstats)
```

Happiness

You decide to ask your friends to rate their happiness on a scale of 0-100 before and after you bother them to ask how happy they are, thereby inciting an existential crises within each of them. You predict that asking this question will decrease the happiness they are feeling at that moment. These data are ranked data, but assume they are parametric given the size of the scale being used.

```
before_asking <- c(96,80,86,92,100,92,95,91,87)
after_asking <- c(72,53,90,85,90,86,83,79,62)

happiness <- cbind(before_asking, after_asking) |>
  as.data.frame()

happiness
```

```
##   before_asking after_asking
```

```
## 1            96            72
## 2            80            53
## 3            86            90
## 4            92            85
## 5           100            90
## 6            92            86
## 7            95            83
## 8            91            79
## 9            87            62
```

statistic = 3.86, p = 0.002

Running and grades

You hear that going running is good for your grades. You decide to look and see if the amount of time people spend running is related to their grades. You obtain the following data:

```
run_time <- c(20,25,22,50,30,10,6)
grade <- c(94,84,95,72,88,90,85)

run_grades <- cbind(run_time, grade) |>
  as.data.frame()

run_grades
```

```
##   run_time grade
## 1       20    94
## 2       25    84
## 3       22    95
## 4       50    72
## 5       30    88
## 6       10    90
## 7        6    85
```

Test statistic = -1.92, p = 0.11, other value = -0.65

Fosbury Flop

High-jumpers use the "Fosbury Flop" because it improves their performance by allowing their center of mass to pass *under* the high-jump bar while their bodies pass *over* the bar. Below are jump heights (in meters) for world records from before the Fosbury Flop was widely used and after the Fosbury Flop was widely used. Does the flop significantly improve athlete performace?

```
pre_flop <- c(2.09,2.12,2.15,2.18,2.17,2.28,2.29)
post_flop <- c(2.30,2.33,2.39,2.34,2.42,2.45,2.44)

jump_heights <- cbind(pre_flop, post_flop) |>
  as.data.frame()

jump_heights
```

```
##   pre_flop post_flop
## 1     2.09      2.30
## 2     2.12      2.33
## 3     2.15      2.39
## 4     2.18      2.34
## 5     2.17      2.42
```

```
## 6     2.28      2.45
## 7     2.29      2.44
```

Stat = -5.47, p < 0.0001

Sandhills, Stonehills

You decide to look at concentrations of Greater Prairie-Chickens *Tympanuchus cupido* in fields a set amount of years after control burns. The following table shows the count of prairie-chickens in each field from the year of the burn until 5 years after the burn. Conditions are lettered to ensure sorting is performed correctly.

Assume these data are normally distributed.

```r
Location <- c(rep("Sandhills", 4), rep("Stonehills", 4))
a_burn_year <- c(1,0,3,2,1,0,0,2)
b_one_year_post_burn <- c(3,5,7,6,5,3,2,5)
c_five_years_post_burn <- c(20,21,15,8,8,7,10,8)

burn_counts <- c(a_burn_year, b_one_year_post_burn, c_five_years_post_burn)

# prairie_chickens <- cbind(Location, a_burn_year, b_one_year_post_burn, c_five_years_post_burn) |>
  # as.data.frame()

time <- c(rep("a_burn_year", 8),
          rep("b_one_year_post_burn", 8),
          rep("c_five_years_post_burn", 8))


prairie_chickens <- cbind(Location, time, burn_counts) |>
  as.data.frame() |>
  mutate(Location = as.factor(Location)) |>
  mutate(time = as.factor(time)) |>
  mutate(burn_counts = as.numeric(burn_counts))

prairie_chickens
```

```
##      Location                   time burn_counts
## 1   Sandhills            a_burn_year           1
## 2   Sandhills            a_burn_year           0
## 3   Sandhills            a_burn_year           3
## 4   Sandhills            a_burn_year           2
## 5  Stonehills            a_burn_year           1
## 6  Stonehills            a_burn_year           0
## 7  Stonehills            a_burn_year           0
## 8  Stonehills            a_burn_year           2
## 9   Sandhills   b_one_year_post_burn           3
## 10  Sandhills   b_one_year_post_burn           5
## 11  Sandhills   b_one_year_post_burn           7
## 12  Sandhills   b_one_year_post_burn           6
## 13 Stonehills   b_one_year_post_burn           5
## 14 Stonehills   b_one_year_post_burn           3
## 15 Stonehills   b_one_year_post_burn           2
## 16 Stonehills   b_one_year_post_burn           5
## 17  Sandhills c_five_years_post_burn          20
## 18  Sandhills c_five_years_post_burn          21
## 19  Sandhills c_five_years_post_burn          15
```

```
## 20  Sandhills c_five_years_post_burn           8
## 21 Stonehills c_five_years_post_burn           8
## 22 Stonehills c_five_years_post_burn           7
## 23 Stonehills c_five_years_post_burn          10
## 24 Stonehills c_five_years_post_burn           8
```

Statistic = 26.31, $p < 0.0001$.