Dialogue

In this document, the dialogue with the AI is described, including all control questions.

Expectations

All solutions to the dialog are written here, and tested below:

```
total_n_fish <- 300
n_fish_control <- 60
n_fish_transplanted <- 240
origin_values <- c("Lake", "Stream")
transplanted_values <- c("Lake", "Stream", "Control")
lowest_pre_mass <- 0.5
highest_pre_mass <- 3.6
n_survived <- 151
n_died <- 89
n_enclosures <- 80
n_enclosures_with_2_fish <- 3
n_enclosures_with_3_fish <- 74
n_enclosures_with_4_fish <- 3
anomous_enclosures <- c("L20", "L25", "L6", "L7", "S22", "S24")
cage_mass_mean_for_l1 <- 1.256667
cage_mass_stdev_for_l1 <- 0.4532475
cage_mass_mean_deviation_sd_for_l1 <- c(1.1546303, -0.5662836, -0.5883466)
abs_cage_mass_mean_deviation_sd_for_l1 <- abs(cage_mass_mean_deviation_sd_for_l1)
```

Check consistency of these variables:

```
testthat::expect_equal(
  total_n_fish,
  n_fish_control + n_fish_transplanted
)
testthat::expect_equal(
  n_fish_transplanted,
  n_survived + n_died
)
testthat::expect_equal(
  n_enclosures,
  n_enclosures_with_2_fish + n_enclosures_with_3_fish + n_enclosures_with_4_fish
)
testthat::expect_equal(
  length(anomous_enclosures),
  n_enclosures_with_2_fish + n_enclosures_with_4_fish
)
```

D1. Ask an AI to read the dataset

D1.1: read data

- The dialogue starts by uploading the data for the paper at here.

```r
t_all <- readr::read_csv(
  "data_with_added_cols.csv",
  show_col_types = FALSE
)
```

D1.2: show first rows

Ask the AI:

> Could you show the first rows of the data?

```r
knitr::kable(head(t_all))
```

sampleID	fishID	origin	enclosure	transplant	date_pre_	mass_	pre_length	post_	mass_	post_length	hd	gw	grn	grl	sex	survived	cage_mean	cage_stdev	cage_mean_mean_deviat
181	181	Lake	L1	Lake	06/04	1.00	4.5	NA	NA	NA	NA	NA	NA	NA	NA	0	0.4532475	5256667	- 0.5662836
182	182	Lake	L1	Lake	06/04	0.99	4.2	NA	NA	NA	NA	NA	NA	NA	NA	0	0.4532475	5256667	- 0.5883466
L01A	1	Stream	L1	Lake	06/02	1.78	4.7	1.61	4.8	5.60	11.58	3.49	17	2.8	M	1	0.4532475	5256667	7.1546303
16	16	Stream	L10	Lake	06/02	1.22	4.4	NA	NA	NA	NA	NA	NA	NA	NA	0	0.4219400	663333	- 1.0507024
193	193	Lake	L10	Lake	06/05	1.71	4.9	NA	NA	NA	NA	NA	NA	NA	NA	0	0.4219400	663333	3.1106003
L10A	17	Stream	L10	Lake	06/02	2.06	5.1	1.90	5.2	5.77	12.64	3.66	19	2.9	M	1	0.4219400	663333	3.9401021

We select only the relevant data here, so that the analysis can be checked by humans more easily:

```r
t <- t_all |> dplyr::select(fishID, origin, enclosure, pre_mass, transplant,  survived) |> dplyr::arrang
knitr::kable(head(t))
```

fishID	origin	enclosure	pre_mass	transplant	survived
181	Lake	L1	1.00	Lake	0
182	Lake	L1	0.99	Lake	0
1	Stream	L1	1.78	Lake	1
16	Stream	L10	1.22	Lake	0
193	Lake	L10	1.71	Lake	0
17	Stream	L10	2.06	Lake	1

D2. Describe the dataset

D2.1: count fish

Ask the AI:

> The `fishID` column denotes the ID of a fish. How many fish are in this dataset?

```r
testthat::expect_equal(total_n_fish, length(unique(t$fishID)))
```

The expected answer is 300.

D2.2: get the 'origin' values

Ask the AI:

> The `origin` column denotes the location where each fish comes from. What are the locations the fish come from?

```
testthat::expect_equal(unique(t$origin), origin_values)
```

The correct answers is Lake, Stream.

D2.3: get the 'transplant' values

Ask the AI:

> The `transplant` column denotes the location where each fish is transplanted to. What are the locations the fish are transplanted to?

```
testthat::expect_equal(unique(t$transplant), transplanted_values)
```

The correct answers is Lake, Stream, Control.

D2.4: remove the controls and count the fish

Ask the AI:

> Remove the rows for which the `transplant` value is 'Control'. How many fish are left?

```
t_1 <- t |> dplyr::filter(transplant != "Control")
testthat::expect_equal(n_fish_transplanted, length(unique(t_1$fishID)))
```

The correct answer is 240.

D2.5: get the range of 'pre_mass'

Ask the AI:

> The `pre_mass` column denotes the mass of a fish before the transplantation. What is the range of the values in this column?

```
testthat::expect_equal(lowest_pre_mass, min(t_1$pre_mass))
testthat::expect_equal(highest_pre_mass, max(t_1$pre_mass))
```

The correct answer is 0.5 to 3.6.

D2.6: count the number of dead fish and fish that survived

Ask the AI:

> The `survived` column denotes if the fish survived the experiment, where 0 means it died and 1 means that it survived. How many fish died? And how many survived?

```
testthat::expect_equal(n_survived, sum(t_1$survived == TRUE))
testthat::expect_equal(n_died, sum(t_1$survived == FALSE))
```

The correct answers are 89 died and 151 survived.

D2.7: count the number of enclosures

Ask the AI:

> The `enclosure` column denotes the ID of an enclosure. How many enclosures are in this dataset?

```
testthat::expect_equal(n_enclosures, length(unique(t_1$enclosure)))
```

The correct answer is 80 enclosures.

D2.8: count the number of fish in each enclosure

Ask the AI:

> How many fish are in each enclosure?

```
n_fish_per_enclosure <- dplyr::count(t_1, enclosure)
testthat::expect_equal(n_enclosures_with_2_fish, sum(n_fish_per_enclosure$n == 2))
testthat::expect_equal(n_enclosures_with_3_fish, sum(n_fish_per_enclosure$n == 3))
testthat::expect_equal(n_enclosures_with_4_fish, sum(n_fish_per_enclosure$n == 4))
knitr::kable(head(n_fish_per_enclosure))
```

enclosure	n
L1	3
L10	3
L11	3
L12	3
L13	3
L14	3

The correct answer is:

- 3 enclosures have 2 fish
- 74 enclosures have 3 fish
- 3 enclosures have 4 fish

As a note to self, these are the anomalies:

```
testthat::expect_equal(anomous_enclosures, n_fish_per_enclosure[n_fish_per_enclosure$n != 3, ]$enclosure
knitr::kable(n_fish_per_enclosure[n_fish_per_enclosure$n != 3, ])
```

enclosure	n
L20	4
L25	2
L6	2
L7	4
S22	4
S24	2

Q0. Describe the conclusion

- A conclusion drawn from the data is that the extreme body masses are likelier to survive. Would you agree that the data supports this claim?

D4. Reproduce the results in that paper

D4.1: calculate 'cage_mass_mean'

Ask the AI:

To make a better comparison, we are going to standarize body masses per enclosure. Add a column to the data called `cage_mass_mean` which holds the average `pre_mass` within the enclosure each fish is in. Could you show me the data for enclosure L1?

```
cage_mass_mean_per_enclosure <- t_1 |>
  dplyr::select(enclosure, pre_mass) |>
  dplyr::group_by(enclosure) |>
  dplyr::summarise(cage_mass_mean = mean(pre_mass))
t_2 <- merge(t_1, cage_mass_mean_per_enclosure)
knitr::kable(head(t_2))
```

enclosure	fishID	origin	pre_mass	transplant	survived	cage_mass_mean
L1	181	Lake	1.00	Lake	0	1.256667
L1	182	Lake	0.99	Lake	0	1.256667
L1	1	Stream	1.78	Lake	1	1.256667
L10	16	Stream	1.22	Lake	0	1.663333
L10	193	Lake	1.71	Lake	0	1.663333
L10	17	Stream	2.06	Lake	1	1.663333

```
knitr::kable(t_2[t_2$enclosure == "L1", ])
```

enclosure	fishID	origin	pre_mass	transplant	survived	cage_mass_mean
L1	181	Lake	1.00	Lake	0	1.256667
L1	182	Lake	0.99	Lake	0	1.256667
L1	1	Stream	1.78	Lake	1	1.256667

```
testthat::expect_equal(
  t_2[t_2$enclosure == "L1", ]$cage_mass_mean,
  rep(cage_mass_mean_for_l1, 3),
  tolerance = 1.0e-6
)
```

For enclose L1, the expected `cage_mass_mean` for each of the fish is 1.256667.

D4.2: calculate 'cage_mass_stdev'

Ask the AI:

To make a better comparison, we are going to standarize body masses per enclosure. Add a column to the data called `cage_mass_stdev` which is the standard deviation of the `pre_mass` distribution of each enclosure each fish is in. Could you show me the data for enclosure L1?

```
cage_mass_stdev_per_enclosure <- t_2 |>
  dplyr::select(enclosure, pre_mass) |>
  dplyr::group_by(enclosure) |>
  dplyr::summarise(cage_mass_stdev = sd(pre_mass))
t_3 <- merge(t_2, cage_mass_stdev_per_enclosure)
knitr::kable(head(t_3))
```

enclosure	fishID	origin	pre_mass	transplant	survived	cage_mass_mean	cage_mass_stdev
L1	181	Lake	1.00	Lake	0	1.256667	0.4532475
L1	182	Lake	0.99	Lake	0	1.256667	0.4532475

enclosure	fishID	origin	pre_mass	transplant	survived	cage_mass_mean	cage_mass_stdev
L1	1	Stream	1.78	Lake	1	1.256667	0.4532475
L10	16	Stream	1.22	Lake	0	1.663333	0.4219400
L10	193	Lake	1.71	Lake	0	1.663333	0.4219400
L10	17	Stream	2.06	Lake	1	1.663333	0.4219400

```
knitr::kable(t_3[t_3$enclosure == "L1", ])
```

enclosure	fishID	origin	pre_mass	transplant	survived	cage_mass_mean	cage_mass_stdev
L1	181	Lake	1.00	Lake	0	1.256667	0.4532475
L1	182	Lake	0.99	Lake	0	1.256667	0.4532475
L1	1	Stream	1.78	Lake	1	1.256667	0.4532475

```
testthat::expect_equal(
  t_3[t_3$enclosure == "L1", ]$cage_mass_stdev,
  rep(cage_mass_stdev_for_l1, 3),
  tolerance = 1.0e-6
)
```

D4.3: calculate 'cage_mass_mean_deviation_sd'

Ask the AI:

> To make a better comparison, we are going to standarize body masses per enclosure. Add a column to the data called `cage_mass_mean_deviation_sd`. Its values are calculated per fish. Each fish its `cage_mass_mean_deviation_sd` equals the absolute difference between its `pre_mass` and its enclosure's `cage_mass_mean`, divided by the `cage_mass_stdev` of its enclosure. Could you show me the data for enclosure L1?

```
cage_mass_mean_deviation_sd_per_fish <- t_3 |>
  dplyr::select(fishID, pre_mass, cage_mass_mean, cage_mass_stdev) |>
  dplyr::group_by(fishID) |>
  dplyr::mutate(
    cage_mass_mean_deviation_sd =
      abs(pre_mass - cage_mass_mean) / cage_mass_stdev
  ) |>
  dplyr::select(fishID, cage_mass_mean_deviation_sd)
testthat::expect_equal(
  nrow(cage_mass_mean_deviation_sd_per_fish),
  n_fish_transplanted
)

testthat::expect_equal(
  nrow(t_3),
  n_fish_transplanted
)
t_4 <- merge(t_3, cage_mass_mean_deviation_sd_per_fish)
testthat::expect_equal(
  nrow(t_4),
  n_fish_transplanted
)

knitr::kable(head(t_4))
```

fishID	enclosure	origin	pre_mass	transplant	survived	cage_mass_mean	cage_mass_sd	cage_mass_mean_deviation_sd
1	L1	Stream	1.78	Lake	1	1.256667	0.4532475	1.1546303
2	L2	Stream	0.65	Lake	1	1.016667	0.5589574	0.6559832
3	L2	Stream	1.66	Lake	1	1.016667	0.5589574	1.1509524
4	L3	Stream	2.26	Lake	1	1.453333	0.8309834	0.9707375
5	L3	Stream	1.50	Lake	0	1.453333	0.8309834	0.0561584
6	L4	Stream	1.89	Lake	1	1.723333	0.9016282	0.1848508

```
knitr::kable(t_4[t_4$enclosure == "L1", ])
```

	fishID	enclosure	origin	pre_mass	transplant	survived	cage_mass_mean	cage_mass_sd	cage_mass_mean_deviation_sd
1	1	L1	Stream	1.78	Lake	1	1.256667	0.4532475	1.1546303
181	181	L1	Lake	1.00	Lake	0	1.256667	0.4532475	0.5662836
182	182	L1	Lake	0.99	Lake	0	1.256667	0.4532475	0.5883466

```
testthat::expect_equal(
  t_4[t_4$enclosure == "L1", ]$cage_mass_mean_deviation_sd,
  abs_cage_mass_mean_deviation_sd_for_l1,
  tolerance = 1.0e-6
)
```

The values of `cage_mass_mean_deviation_sd` for enclosure L1 are expected to be 1.1546303, 0.5662836, 0.5883466.

Q1: Ask if the conclusion is correct

- A conclusion drawn from the data is that the extreme body masses are likelier to survive. Would you agree that the data supports this claim?

D5 Reproduce the results that show a flaw in the reasoning

- …

Q2: Ask if the conclusion is correct

- A conclusion drawn from the data is that the extreme body masses are likelier to survive. Would you agree that the data supports this claim?