freeze_vs_cold

Lauren Gill

31/07/2023

This is an analysis of the data for freeze vs cold

Load libraries

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- 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(officer)
library(rvg)
```

Read in dataset

```
cold <- read_csv("../data/coldvsfreeze.csv")
```

```
## Rows: 11 Columns: 18
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr   (3): date_frozen, acclimation_day_length, treatment
## dbl  (10): minutes_cold, exposure_temp, acclimation_temp, acclimation_length...
## time  (5): time_stamp_freeze, time_stamp_end, time_spent_frozen, cold_exposu...
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

Plotting

How to display survival and freezing .... Trying a stacked bar chart showing time spent frozen and time spent at cold exposure First create data that will suit this type of plot

```
cold$slug_number <- as.factor(cold$slug_number)
cold$survival <- as.factor(cold$survival)
cold$frozen <- as.factor(cold$frozen)

stacked_bar <- cold %>%
  dplyr::select(survival, cold_exposure_length, time_spent_frozen, slug_number) %>%
  mutate(only_cold = cold_exposure_length - time_spent_frozen) %>%
  pivot_longer(cols = c(time_spent_frozen, only_cold),
               names_to = "state",
               values_to = "time")

stacked_bar
```

```
## # A tibble: 22 x 5
##    survival cold_exposure_length slug_number state             time
##    <fct>    <time>               <fct>       <chr>             <time>
##  1 1        37’26"               1           time_spent_frozen 00’00"
##  2 1        37’26"               1           only_cold         37’26"
##  3 0        36’04"               2           time_spent_frozen 12’59"
##  4 0        36’04"               2           only_cold         23’05"
##  5 1        36’19"               3           time_spent_frozen 00’00"
##  6 1        36’19"               3           only_cold         36’19"
##  7 0        35’48"               4           time_spent_frozen 00’55"
##  8 0        35’48"               4           only_cold         34’53"
##  9 0        37’02"               5           time_spent_frozen 01’07"
## 10 0        37’02"               5           only_cold         35’55"
## # i 12 more rows
```

#Stats Testing this with logistic regression

```
cold.glm <- glm(survival ~ weight + cold_exposure_length + frozen, data = cold, family="binomial")
summary(cold.glm)
```

```
##
## Call:
## glm(formula = survival ~ weight + cold_exposure_length + frozen,
##     family = "binomial", data = cold)
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)           2.457e+01  6.464e+05   0.000        1
## weight               -4.138e-11  1.498e+05   0.000        1
## cold_exposure_length -2.534e-14  2.711e+02   0.000        1
## frozen1              -4.913e+01  8.541e+04  -0.001        1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 1.5158e+01  on 10  degrees of freedom
## Residual deviance: 4.7154e-10  on  7  degrees of freedom
## AIC: 8
##
## Number of Fisher Scoring iterations: 23
```

```
anova(cold.glm,test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: survival
##
## Terms added sequentially (first to last)
##
##
##                      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                    10     15.158
## weight                1   0.2803         9     14.878 0.5964813
## cold_exposure_length  1   1.3373         8     13.541 0.2475144
## frozen                1  13.5406         7      0.000 0.0002335 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

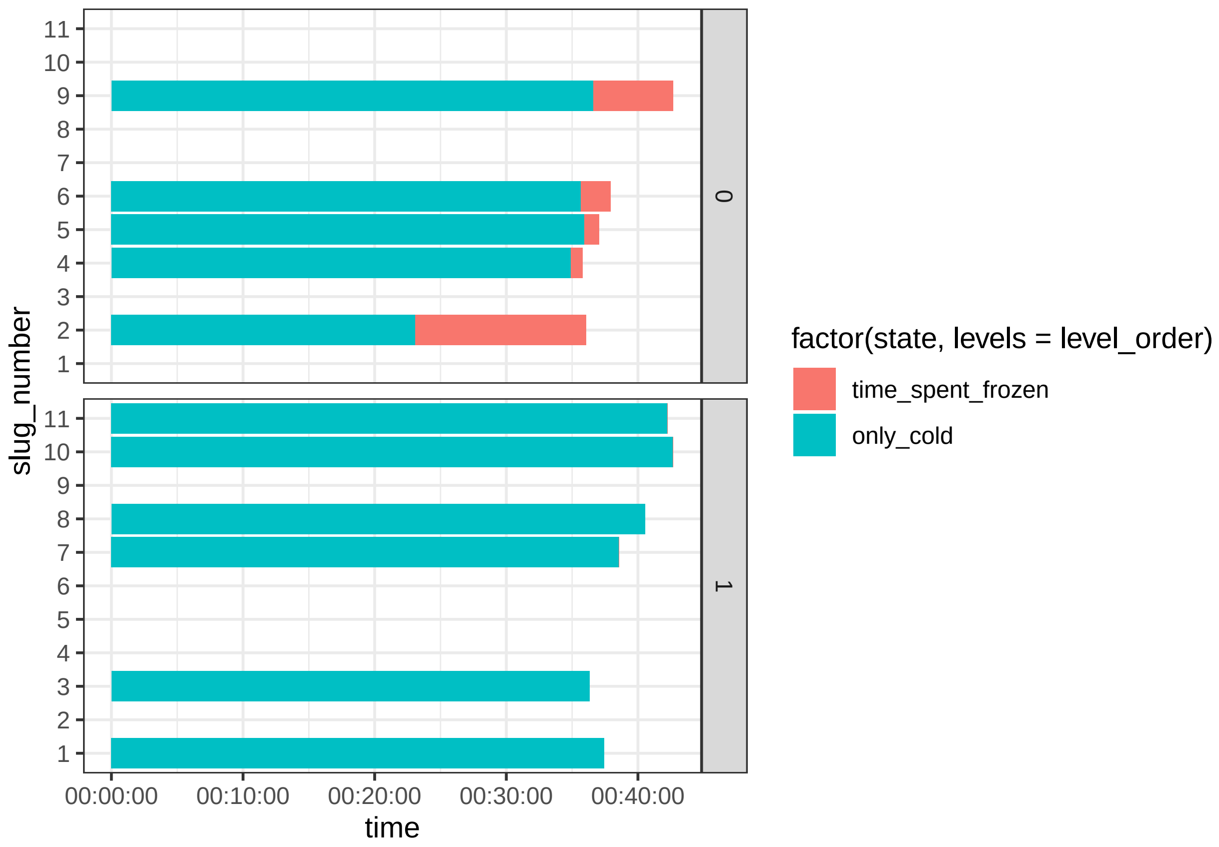
More Plotting

Now creating a stacked bar plot

```
level_order <- c("time_spent_frozen", "only_cold")

cold_bar <- ggplot(data = stacked_bar, aes(x = time, y = slug_number, fill = factor(state, levels = leve
    geom_bar(stat='identity') +
  facet_grid(survival ~ .) +
  theme_bw()

cold_bar
```



Exporting it to power point to make it look pretty https://rpubs.com/techanswers88/VectorGraphicsChartsUsingGGPLOT

```
#first export to vector object
#myplot <-  dml(ggobj = cold_bar
              ,bg = "white"
              ,pointsize = 12
              ,editable = TRUE)
#doc <- read_pptx()


#doc <- add_slide(doc, "Title and Content", "Office Theme")
#doc <- ph_with(doc, myplot, location = ph_location_fullsize())

# If you want to create multiple charts then create your second chart and call it myplot2

#doc <- add_slide(doc, "Title and Content", "Office Theme")
#doc <- ph_with(doc, myplot, location = ph_location_fullsize())

#fileout <- "mygraph.pptx"
#print(doc, target = fileout)
```