Motivations for contributing to OS: statistical analysis

Overview

This script runs some statistical tests on data from Q6, which is about participants' reasons for contributing to open source.

Import packages and utilities

```
project_root <- here::here() # requires that you be somewhere in the
# project directory (not above it)
# packages
suppressMessages(source(file.path(project_root, "scripts/packages.R")))
# functions and objects used across scripts
suppressMessages(source(file.path(project_root, "scripts/utils.R")))
```

Set seed

```
set.seed(42)
```

Function definition

pairwise_z_test_lessthan

- Arguments:

  - `df`: A data frame where rows are participants, and columns are a predictor, either `job_category` or `Role`, plus at least one response variable, e.g. `Skills`, `Give back`, etc. Extra columns are okay. The response variable of interest should be a column of 0s and 1s.
  - `outcome_col`: A string. `Skills` by default, but could be any of the 7 response variables, e.g. `Improve tools`, `Job`, etc.
  - `predictor_col`: A string. The name of a factor column containing 2 groups to compare. Currently, should be either `Role` or `job_category` (default).
  - `group1`: A string. The job category that you suspect has a lower "success rate", of the two.
  - `group2`: A string. The job category that you suspect has a higher "success rate", of the two.

- Details:

  - A simple function that performs a pairwise z-test for equality of two proportions. Most of the function is just summing across the data frame. it calls stats::prop.test to run a one-sided z-test testing whether the proportion of "successes" in group1 is less than that of group2.

- Outputs:

  - An "htest" object, from stats::prop.test.

```r
pairwise_z_test_lessthan <- function(
  df,
  outcome_col = "Skills",
  predictor_col = "job_category",
  group1,
  group2,
  alternative = "less"
) {
  # Count total and 'yes' outcomes for each group
  n1 <- sum(df[[predictor_col]] == group1)
  y1 <- sum(df[[predictor_col]] == group1 & df[[outcome_col]] == 1)

  n2 <- sum(df[[predictor_col]] == group2)
  y2 <- sum(df[[predictor_col]] == group2 & df[[outcome_col]] == 1)

  # Perform the one-sided prop test (testing if group1 < group2)
  result <- prop.test(
    x = c(y1, y2),
    n = c(n1, n2),
    alternative = alternative,
```

```
  )

  return(result)
}
```

Load data

```
motivations <- load_qualtrics_data("clean_data/motivations_Q6.tsv")
other_quant <- load_qualtrics_data("clean_data/other_quant.tsv")
```

Wrangle data

```
motivations_job_staff <- cbind(motivations, other_quant$job_category)
# Rename columns
names(motivations_job_staff)[length(names(motivations_job_staff))] <- "job_category"
motivations_job_staff <- cbind(motivations_job_staff, other_quant$staff_categories)
names(motivations_job_staff)[length(names(motivations_job_staff))] <- "staff_category"
# Remove any rows where the job_category or staff_category are missing
motivations_job_staff_clean <- exclude_empty_rows(motivations_job_staff, strict=TRUE)
# Remove rows of all 0s
motivations_job_staff_clean <- motivations_job_staff_clean %>%
  filter(!if_all(Job:Other, ~ .x == 0))
head(motivations_job_staff_clean)
```

```
  Job Improve Tools Customize Network Give back Skills Fun Other
1   0             1         1       0         0      1   1     0
2   1             0         0       0         1      0   0     0
3   0             1         1       1         0      0   0     0
4   1             0         0       1         1      0   0     0
5   0             1         1       0         1      1   1     1
6   1             1         1       1         1      1   1     1
        job_category                  staff_category
1 Non-research Staff                           Other
2 Non-research Staff DevOps or System Administration
3 Non-research Staff DevOps or System Administration
4 Non-research Staff     Information Technology (IT)
```

```
5 Non-research Staff DevOps or System Administration
6 Non-research Staff                           Other
```

```r
# Do the same, but dropping staff categories (e.g. IT)
motivations_job <- subset(motivations_job_staff, select=-staff_category)
# Remove any rows where the job_category is missing
motivations_job_clean <- exclude_empty_rows(motivations_job, strict=TRUE)
# Remove rows of all 0s
motivations_job_clean <- motivations_job_clean %>%
  filter(!if_all(Job:Other, ~ .x == 0))
head(motivations_job_clean)
```

```
  Job Improve Tools Customize Network Give back Skills Fun Other
1   1             1         1       1         1      1   1     0
2   0             1         1       1         0      1   0     0
3   0             1         1       0         0      1   1     0
4   1             1         1       0         1      0   0     0
5   0             1         1       0         1      1   1     0
6   0             1         1       0         0      0   0     1
          job_category
1              Faculty
2             Post-Doc
3 Other research staff
4              Faculty
5              Faculty
6              Faculty
```

```r
# This will also come in handy later.
motivation_cols <- names(motivations_job_clean)[-length(names(motivations_job_clean))]
motivation_cols
```

```
[1] "Job"            "Improve Tools" "Customize"      "Network"
[5] "Give back"      "Skills"        "Fun"            "Other"
```

Here, we use some functions in my utility script (scripts/utilities.R) to clean up the data for Q6. We'll call the resulting data frame `motivations_raw`. This data frame also has a `Role` column indicating the participant's job category.

Here, we combine postdocs and other research staff into one category. We'll call the resulting data frame `motivations_processed`. We will use this for most of our statistical analysis. It gives us more statistical power, and I think it is reasonable in terms of interpretability.

```
motivations_job_clean_streamlined <- motivations_job_clean %>%
  mutate(
    job_category = case_when(
      job_category %in% c("Post-Doc", "Other research staff") ~
        "Postdocs and Staff Researchers",
      TRUE ~ job_category
    )
  )
```

Create the regression model

I'm interested in the whether a person's job category affects how they will answer this question. In other words, can we predict their profile of motivations significantly better when taking job category into account? I am doing a multivariate logistic regression predicting a vector of binary responses. It's multivariate because instead of doing Y ~ X, we are now doing [Y1, Y2, Y3…] ~ X. It's logistic because all response variables are binary. I'm using the mvabund() package, which is designed for non-continuous data: counts and binary outcomes (they call these "abundance data" in the packaage documentation). Base R's lm() is for continuous data, and I didn't see a function in base R for this type of analysis.

First, we just split our data frame into two. For each observation, X is a single categorical outcome, and we have seven Ys which are binary outcomes.

```
Y <- as.matrix(motivations_job_clean_streamlined[, motivation_cols])
X <- motivations_job_clean_streamlined$job_category
head(Y)
```

```
     Job Improve Tools Customize Network Give back Skills Fun Other
[1,]   1             1         1       1         1      1   1     0
[2,]   0             1         1       1         0      1   0     0
[3,]   0             1         1       0         0      1   1     0
[4,]   1             1         1       0         1      0   0     0
[5,]   0             1         1       0         1      1   1     0
[6,]   0             1         1       0         0      0   0     1
```

```
head(X)
```

```
[1] "Faculty"                        "Postdocs and Staff Researchers"
[3] "Postdocs and Staff Researchers" "Faculty"
[5] "Faculty"                        "Faculty"
```

Create the model. I'm mostly using the default settings.

```
fit <- mvabund::manyglm(Y ~ X, family = "binomial", show.coef=TRUE)
fit
```

```
Call:  mvabund::manyglm(formula = Y ~ X, family = "binomial", show.coef = TRUE)
[1] "binomial(link=logit)"
Coefficients:
                                 Job     Improve Tools  Customize  Network
(Intercept)                      -0.448   2.380          1.168     -0.989
XGrad Student                     0.448  -0.343         -0.169      0.353
XNon-research Staff               0.167  -0.979         -0.543      0.415
XPostdocs and Staff Researchers   0.930  -0.609         -0.187      0.350
XUndergraduate                   -0.468  -1.463         -2.959      0.701
                                 Give back  Skills  Fun     Other
(Intercept)                       0.593     -0.448   0.170  -1.264
XGrad Student                     0.405      1.883   0.641  -0.773
XNon-research Staff               0.356      1.230   0.207  -1.014
XPostdocs and Staff Researchers  -0.411      0.778  -0.426  -1.282
XUndergraduate                   14.222     15.264   0.118   0.347

Degrees of Freedom: 232 Total (i.e. Null); 228 Residual

                    Job     Improve Tools  Customize  Network  Give back
2*log-likelihood:   -314.0  -192.4         -276.4     -295.4   -284.7
Residual Deviance:   314.0   192.4          276.4      295.4    284.7
AIC:                 324.0   202.4          286.4      305.4    294.7
                    Skills  Fun     Other
2*log-likelihood:   -286.1  -314.6  -171.1
Residual Deviance:   286.1   314.6   171.1
AIC:                 296.1   324.6   181.1
```

Immediately, we notice that the coefficients for Undergraduates on "Skills" and "Give back" are very different from all other coefficients. This is presumably because all 7 undergraduates who answered this question selected both those options.

The residual deviance statistic is close to the degrees of freedom, which I think suggests a good fit? https://online.stat.psu.edu/stat504/lesson/2/2.5

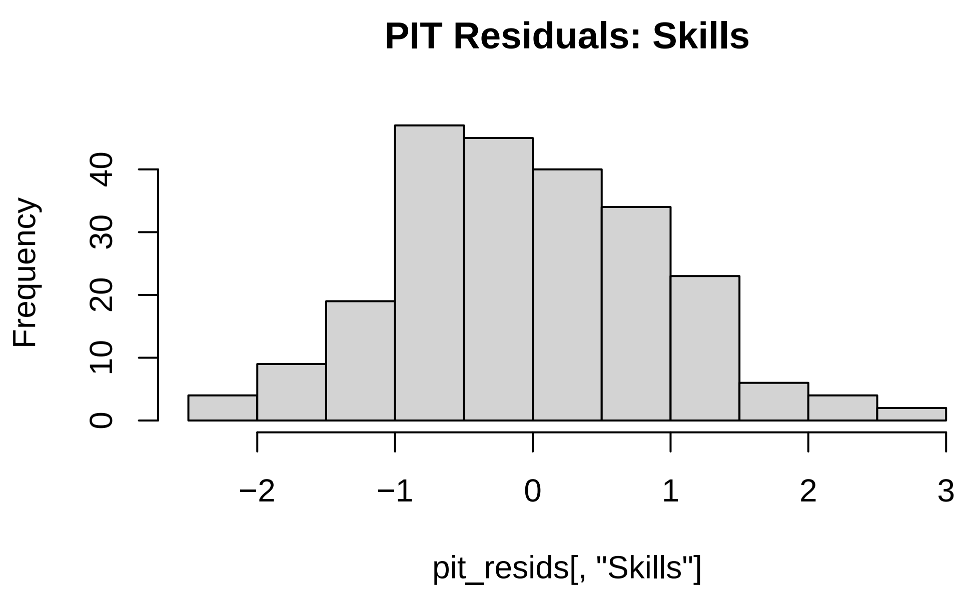
This is a really dumb way to assess goodness of fit, but I also like to look at the AIC, and if it's in the thousands, I start to get nervous. Ours are in the hundreds, so that seems promising?

Assess goodness of fit

Next, I'm eyeballing some plots to assess goodness of fit.

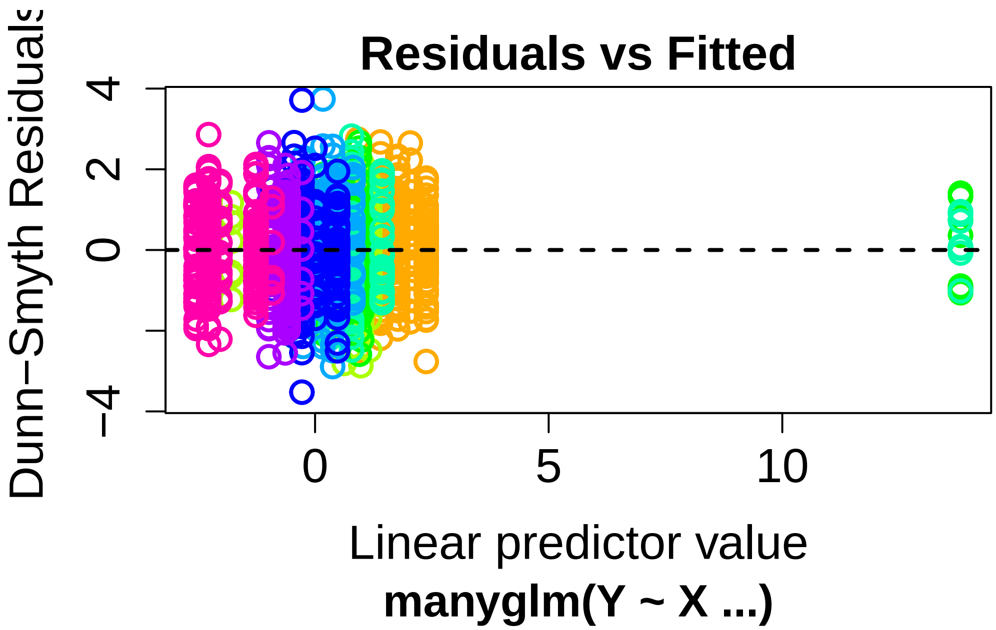
The distribution of the residuals seems normal-ish (shape varies with my chosen random seed). Note I'm just picking one of the categories that had a weird coefficient, to make sure the data in that category aren't too weird.

```
pit_resids <- residuals(fit, type = "pit.trap")
hist(pit_resids[, "Skills"], main = "PIT Residuals: Skills")
```



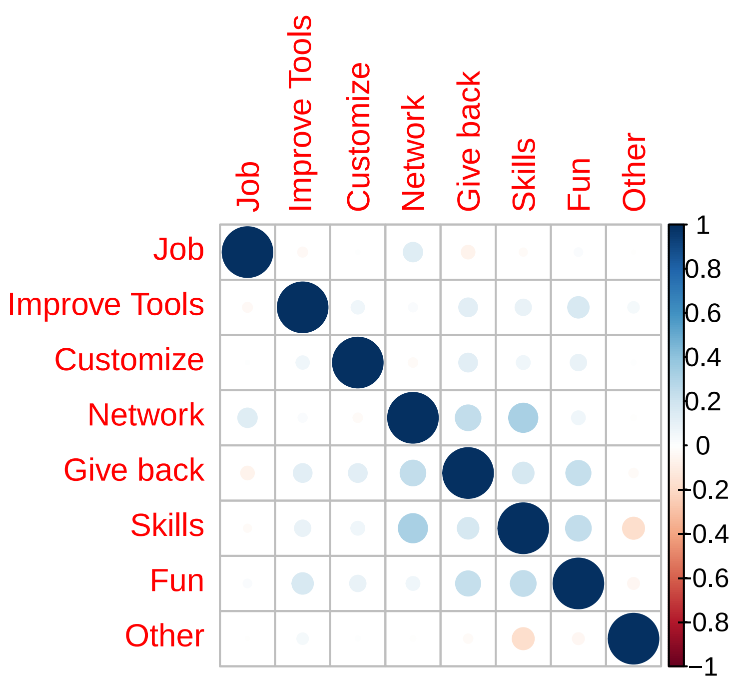
Admittedly, I don't fully understand this next plot, but the mvabund paper emphasizes it. (https://doi.org/10.1111/j.2041-210X.2012.00190.x) I think the point is that the residuals should be distributed around zero. What you don't want is a "fan shape", where as the predictions get more extreme, the residuals do, too. This tutorial from the package author shows an example of this. https://cran.r-project.org/web/packages/ecostats/vignettes/Chapter14Solutions.html I think my plot look as good as the plots he approves of in that tutorial.

```
plot(fit)
```



I think a correlation matrix is also applicable here. The correlations between variables are near zero (ish), suggesting that the model captures the important relationships.

```
corrplot(cor(pit_resids), method = "circle")
```



Hypothesis testing

Now we want to know whether this model is significantly better than a model where the probability of a particular set of motivations is the same for all job categories. anova() summarizes the statistical significance of the fitted model. test="LR" is the default, and specifies a likelihood ratio test. So I guess we are using the likelihood ratio test statistic instead of the standard anova F-statistic, but I think this might be the kind of situation where those two statistics are basically the same? (nested models for hypothesis vs. null) The resamp="pit.trap" ("probability integral transform" residuals) argument is the default resampling method. I think the function resamples the data to get a null distribution.

Global model fit + univariate tests

```
anova_result <- anova(fit, resamp = "pit.trap", test = "LR", p.uni = "adjusted")
```

```
Time elapsed: 0 hr 0 min 0 sec
```

```
anova_result
```

```
Analysis of Deviance Table

Model: Y ~ X

Multivariate test:
            Res.Df Df.diff   Dev Pr(>Dev)
(Intercept)    232
X              228       4 76.37    0.001 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Univariate Tests:
              Job          Improve Tools          Customize          Network
              Dev Pr(>Dev)           Dev Pr(>Dev)       Dev Pr(>Dev)     Dev
(Intercept)
X           8.032    0.404         4.805    0.583    11.753    0.133   1.671
                   Give back           Skills              Fun           Other
            Pr(>Dev)       Dev Pr(>Dev)     Dev Pr(>Dev)    Dev Pr(>Dev)   Dev
(Intercept)
X              0.804    10.943    0.165 25.603    0.001 5.715    0.583 7.852
```

```
            Pr(>Dev)
(Intercept)
X              0.404
Arguments:
 Test statistics calculated assuming uncorrelated response (for faster computation)
P-value calculated using 999 iterations via PIT-trap resampling.
```

The anova output starts with a table of the multivariate test statistics. This tests for the global effect of Role, by resampling the whole response vector.

The next part of the table is the univariate test statistics, which are separate logistic regressions for each response variable, ignoring the other variables.

Our Pr(>Dev) is a statistically significant p-value, indicating that Role significantly predicts motivation profile. Basically, the model including role better explains the data than the null model.

Only "Skills" shows a significant univariate effect of Role after multiple testing correction (p.uni = "adjusted"). In other words, when considering whether Role can predict a single motivation, it can only predict Skills.

I think this also means that we have enough umdergraduates, because if we didn't, we wouldn't have enough statistical power to reject the null hypothesis, right?

Pairwise tests for job categories

I believe we can use the pairwise.comp argument to test whether pairs of categories in our explanatory variable are significantly different from each other.

```r
anova_pw <- anova(
  fit,
  resamp = "pit.trap",
  test = "LR",
  p.uni = "adjusted",
  pairwise.comp = X
)
```

```
Time elapsed: 0 hr 0 min 0 sec
```

```r
anova_pw
```

```
Analysis of Deviance Table

Model: Y ~ X

Multivariate test:
            Res.Df Df.diff   Dev Pr(>Dev)
(Intercept)    232
X              228       4 76.37    0.001 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Pairwise comparison results:
                                                     Observed statistic
Faculty vs Undergraduate                                         31.555
Postdocs and Staff Researchers vs Undergraduate                  30.628
Faculty vs Non-research Staff                                    25.727
Faculty vs Postdocs and Staff Researchers                        19.755
Non-research Staff vs Undergraduate                              19.450
Faculty vs Grad Student                                          18.768
Grad Student vs Undergraduate                                    18.417
Non-research Staff vs Postdocs and Staff Researchers             15.943
Grad Student vs Postdocs and Staff Researchers                   13.104
Grad Student vs Non-research Staff                                4.494
                                                     Free Stepdown Adjusted P-Value
Faculty vs Undergraduate                                                      0.016
Postdocs and Staff Researchers vs Undergraduate                               0.016
Faculty vs Non-research Staff                                                 0.038
Faculty vs Postdocs and Staff Researchers                                     0.170
Non-research Staff vs Undergraduate                                           0.170
Faculty vs Grad Student                                                       0.170
Grad Student vs Undergraduate                                                 0.170
Non-research Staff vs Postdocs and Staff Researchers                          0.196
Grad Student vs Postdocs and Staff Researchers                                0.267
Grad Student vs Non-research Staff                                            0.790

Faculty vs Undergraduate                             *
Postdocs and Staff Researchers vs Undergraduate      *
Faculty vs Non-research Staff                        *
Faculty vs Postdocs and Staff Researchers
Non-research Staff vs Undergraduate
Faculty vs Grad Student
Grad Student vs Undergraduate
Non-research Staff vs Postdocs and Staff Researchers
```

```
Grad Student vs Postdocs and Staff Researchers
Grad Student vs Non-research Staff
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Univariate Tests:
              Job          Improve Tools          Customize          Network
              Dev Pr(>Dev)           Dev Pr(>Dev)       Dev Pr(>Dev)     Dev
(Intercept)
X           8.032    0.443         4.805    0.576    11.753    0.166   1.671
                     Give back          Skills             Fun          Other
            Pr(>Dev)       Dev Pr(>Dev)    Dev Pr(>Dev)   Dev Pr(>Dev)   Dev
(Intercept)
X              0.794    10.943    0.198 25.603    0.001 5.715    0.575 7.852

            Pr(>Dev)
(Intercept)
X              0.443
Arguments:
 Test statistics calculated assuming uncorrelated response (for faster computation)
P-value calculated using 999 iterations via PIT-trap resampling.
```

The results indicate three significant pairwise comparisons:
* Faculty vs Undergraduate
* Postdocs and Staff Researchers vs Undergraduate
* Faculty vs Non-research Staff

Test for trend in "skills"

In my other script, motivations_plots, we have one plot where we apparently see a trend: the probability of a respondent choosing "skills" as a motivator appears to decrease as they advance in their academic career. We will use a Cochrane-Armitage test for trend to evaluate whether this trend is real. More precisely, I believe we are evaluating whether the order "P(Yes | Undergrad) > P(Yes | Grad) > P(Yes | Postdoc) > P(Yes | Faculty)" is highly unlikely (<95% chance) given the null hypothesis that all four categories have the same probability of a "yes" response.

Full disclosure: I'm being a little p-hacky here, because I'm only trying this after I tried a series of pairwise z-tests to see whether the proportion of "yes" for "skills" was significantly different from undergrads vs. grads, grads vs. postdocs, etc. That analysis is after this section. In all seriousness, I don't actually feel that I am p-hacking because I'm not just using a new test to try and make the same claim; this is a different test and we will interpret it appropriately. I'm not claiming that undergrads are more likely than grads to select skills; I'm just claiming that there is a trend across the 4 categories.

```
# Here, I haven't combined post-docs and other research staff
n_postdoc <- sum(motivations_job_clean$job_category == "Post-Doc")
n_postdoc_yes <- sum(
  motivations_job_clean$job_category == "Post-Doc" &
    motivations_job_clean$Skills == 1
)
# For the other groups, it doesn't matter if we use the raw or processed data
n_faculty <- sum(motivations_job_clean$job_category == "Faculty")
n_faculty_yes <- sum(
  motivations_job_clean$job_category == "Faculty" &
    motivations_job_clean$Skills == 1
)

n_grad <- sum(motivations_job_clean$job_category == "Grad Student")
n_grad_yes <- sum(
  motivations_job_clean$job_category == "Grad Student" &
    motivations_job_clean$Skills == 1
)

n_undergrad <- sum(motivations_job_clean$job_category == "Undergraduate")
n_undergrad_yes <- sum(
  motivations_job_clean$job_category == "Undergraduate" &
    motivations_job_clean$Skills == 1
)

n_yes <- c(
  n_undergrad_yes,
  n_grad_yes,
  n_postdoc_yes,
  n_faculty_yes
)

n_tot <- c(
  n_undergrad,
  n_grad,
  n_postdoc,
  n_faculty
```

```
)

# Assign scores 1,2,3,4 for Undergrad --> Faculty
# To indicate the ordering
scores <- 1:4

stats::prop.trend.test(
  x = n_yes,
  n = n_tot,
  score = scores
)
```

```
    Chi-squared Test for Trend in Proportions

data:  n_yes out of n_tot ,
 using scores: 1 2 3 4
X-squared = 19.818, df = 1, p-value = 8.518e-06
```

I'm honestly not sure whether this is a one-tailed or two-tailed test… I would assume one-tailed, but the documentation is terse. Anyway, even if we divide that p-value by two it's still well under p=0.05.

Negative/abandoned analysis: Pairwise z-tests

I also looked at that skills trend from a different perspective: is each pair of consecutive categories significantly different? So, are faculty significantly less likely to select "Skills" than postdocs, are postdocs significantly less likely than grad students to select it, etc.?

The results of this analysis are both less significant and harder to interpret than the trend test, but I'm including it for posterity.

I also did some post-hoc power analyses, because we have small sample sizes: 15 postdocs and 7 undergraduates. If we fail to reject the null hypothesis, it could just be because we lack statistical power. Here are some links that I based this on:
https://rpubs.com/sypark0215/223385
https://cran.r-project.org/web/packages/pwr/vignettes/pwr-vignette.html
Essentially I am asking: what ratio of group1:group2 is needed to achieve 80% power?

Graduate students vs. Undergraduates

Let's start with the power analysis.

First, let's prepare the proportions we'll need to run the power test. We might as well do this for all four job categories of interest.

```
# If this were python I would make a class, but I'm not that good
# at R coding so I'm just making a bunch of variables LOL.

p_grad_yes <- n_grad_yes / n_grad
p_undergrad_yes <- n_undergrad_yes / n_undergrad
p_faculty_yes <- n_faculty_yes / n_faculty
p_postdoc_yes <- n_postdoc_yes / n_postdoc
```

Calculate Cohen's h, the effect size.

```
h <- pwr::ES.h(p_grad_yes, p_undergrad_yes)
```

Now, what ratio of n_undergrads to n_gradstudents is needed to achieve 80% power? This one-sided test allows us to specify our unequal group sizes.

```
pwr::pwr.2p2n.test(
  h = h,
  n1 = n_grad,
  sig.level = 0.05,
  power = 0.8,
  alternative = "less"
)
```

```

     difference of proportion power calculation for binomial distribution (arcsine transforma

              h = -0.9079225
             n1 = 26
             n2 = 10.54087
      sig.level = 0.05
          power = 0.8
    alternative = less

NOTE: different sample sizes
```

So we would need 10.5 undergrads to achieve 80% power. Alas, we only have 7. So there is no point in proceeding with the hypothesis test.

Postdocs vs. Graduate students

Calculate Cohen's h, the effect size.

```
h <- pwr::ES.h(p_postdoc_yes, p_grad_yes)
```

```
tryCatch(
  pwr::pwr.2p2n.test(
    h = h,
    n1 = n_grad,
    sig.level = 0.05,
    power = 0.8,
    alternative = "greater"
  ),
  error = function(e) e
)
```

```
<simpleError in uniroot(function(n2) eval(p.body) - power, c(2 + 1e-10, 1e+09)): f() values a
```

This test fails to give an answer. I think the problem is that the difference in proportions is so small (81% vs. 73%), and the number of grad students is also so small (26), that we will never have enough postdocs to achieve 80% power. With a Cohen's h of 0.5 or greater, it says we would need at least 500 postdocs. With h less than 0.5, the function breaks. So if the absolute value of the effect size were larger, we would have more power, which makes sense. I could plot this function for various h values, but honestly I don't care.

```
pwr::pwr.2p2n.test(
  h = 0.5,
  n1 = n_grad,
  sig.level = 0.05,
  power = 0.8,
  alternative = "greater"
)
```

```

     difference of proportion power calculation for binomial distribution (arcsine transforma

              h = 0.5
             n1 = 26
             n2 = 506.3794
      sig.level = 0.05
```

```
          power = 0.8
    alternative = greater

NOTE: different sample sizes
```

Faculty vs. Postdocs

Calculate Cohen's h, the effect size.

```r
h <- pwr::ES.h(p_faculty_yes, p_postdoc_yes)
```

```r
pwr::pwr.2p2n.test(
  h = h,
  n1 = n_faculty,
  sig.level = 0.05,
  power = 0.8,
  alternative = "less"
)
```

```
     difference of proportion power calculation for binomial distribution (arcsine transforma

              h = -0.7076801
             n1 = 59
             n2 = 15.61167
      sig.level = 0.05
          power = 0.8
    alternative = less

NOTE: different sample sizes
```

We have 15 postdocs, and we need 15.6 postdocs for a one-sided test. Good enough.

```r
pairwise_z_test_lessthan(
  motivations_job_clean, # Note we are just looking at post docs
  group1 = "Faculty",
  group2 = "Post-Doc"
)
```

```
	2-sample test for equality of proportions with continuity correction

data:  c(y1, y2) out of c(n1, n2)
X-squared = 4.383, df = 1, p-value = 0.01815
alternative hypothesis: less
95 percent confidence interval:
 -1.00000000 -0.08679992
sample estimates:
   prop 1    prop 2 
0.3898305 0.7333333 
```

It appears that faculty are significantly less likely than postdocs to select "Skills" as a motivator.

By popular demand: IT vs. Academics

Greg raised an interesting question: what about IT staff vs. academics? Let's play around with this. It's basically the same question we had before, but now our predictor variable only has two categories.

I plotted the data (see motivations_plots.qmd), and it appears that these groups are somewhat different. The "Job" motivation looks to be the most different, just by eyeballing it. But let's see what the statistics say.

Data Wrangling

```r
it <- motivations_job_staff_clean %>%
  filter(staff_category == "Information Technology (IT)") %>%
  select(-c(job_category, staff_category))
it$Role <- "IT"
head(it)
```

```
  Job Improve Tools Customize Network Give back Skills Fun Other Role
1   1             0         0       1         1      0   0     0   IT
2   1             1         1       1         1      1   1     0   IT
3   0             1         1       1         1      1   1     0   IT
4   0             1         0       0         0      0   0     0   IT
5   0             1         0       0         1      1   0     0   IT
6   0             1         1       0         1      1   1     0   IT
```

```
dim(it)
```

```
[1] 33  9
```

```
# Everyone except non-research staff
academics <- motivations_job_clean_streamlined %>%
  filter(
    job_category == "Faculty" |
    job_category == "Grad Students" |
    job_category == "Postdocs and Staff Researchers" |
    job_category == "Undergraduates"
  ) %>%
    select(-job_category)
academics$Role <- "Academic"
head(academics)
```

```
  Job Improve Tools Customize Network Give back Skills Fun Other     Role
1   1             1         1       1         1      1   1     0 Academic
2   0             1         1       1         0      1   0     0 Academic
3   0             1         1       0         0      1   1     0 Academic
4   1             1         1       0         1      0   0     0 Academic
5   0             1         1       0         1      1   1     0 Academic
6   0             1         1       0         0      0   0     1 Academic
```

```
dim(academics)
```

```
[1] 114   9
```

```
it_acad <- rbind(it, academics)
it_acad$Role <- as.factor(it_acad$Role)
dim(it_acad)
```

```
[1] 147   9
```

Regression

Let's try the logistic regression. This model may be more powerful than we need, but since I already have the code, let's just do it.

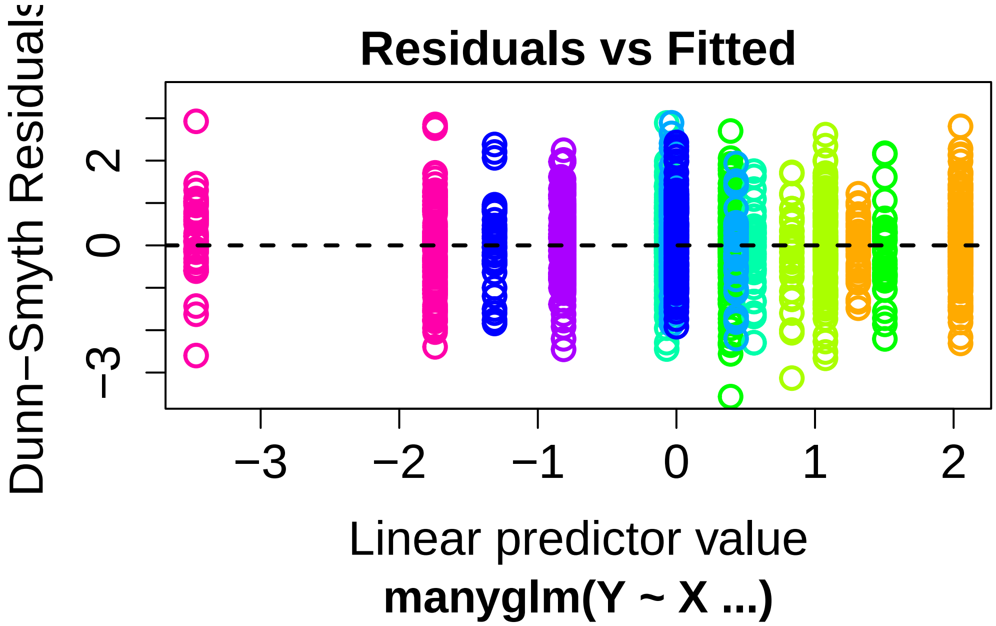
```r
Y <- as.matrix(it_acad[, motivation_cols])
X <- it_acad$Role
```

I'm just glancing at the AIC values, degrees of freedom, and the residual plot. They seem fine. It would be a bit weird if they weren't, since this is the same data as before; the categories are just grouped differently (and in fact, our minimum sample size is larger).

```r
fit <- mvabund::manyglm(Y ~ X, family = "binomial", show.coef=TRUE)
fit
```

```
Call:  mvabund::manyglm(formula = Y ~ X, family = "binomial", show.coef = TRUE)
[1] "binomial(link=logit)"
Coefficients:
             Job     Improve Tools  Customize  Network  Give back  Skills
(Intercept)   0.000   2.050          1.075     -0.814    0.391     -0.070
XIT          -1.312  -0.738         -0.242     -0.019    1.113      0.630
             Fun     Other
(Intercept)  -0.035  -1.741
XIT           0.466  -1.724

Degrees of Freedom: 146 Total (i.e. Null); 145 Residual

                    Job     Improve Tools  Customize  Network  Give back
2*log-likelihood:   -192.1  -115.0         -169.8     -181.1   -185.1
Residual Deviance:   192.1   115.0          169.8      181.1    185.1
AIC:                 196.1   119.0          173.8      185.1    189.1
                    Skills  Fun     Other
2*log-likelihood:   -201.2  -202.3  -105.0
Residual Deviance:   201.2   202.3   105.0
AIC:                 205.2   206.3   109.0
```

```r
plot(fit)
```




```
anova_result <- anova(fit, resamp = "pit.trap", test = "LR", p.uni = "adjusted")
```

```
Time elapsed: 0 hr 0 min 0 sec
```

```
anova_result
```

```
Analysis of Deviance Table

Model: Y ~ X

Multivariate test:
            Res.Df Df.diff   Dev Pr(>Dev)
(Intercept)    146
X              145       1 25.49    0.008 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Univariate Tests:
              Job          Improve Tools          Customize          Network
              Dev Pr(>Dev)           Dev Pr(>Dev)       Dev Pr(>Dev)     Dev
(Intercept)
X           9.179    0.038         1.918    0.526     0.305    0.829   0.002
```

```
                      Give back           Skills            Fun          Other
            Pr(>Dev)       Dev Pr(>Dev)     Dev Pr(>Dev)   Dev Pr(>Dev)    Dev
(Intercept)
X              0.977     5.963    0.107  2.456    0.460 1.361    0.560  4.31

            Pr(>Dev)
(Intercept)
X              0.221
Arguments:
 Test statistics calculated assuming uncorrelated response (for faster computation)
P-value calculated using 999 iterations via PIT-trap resampling.
```

ANOVA/LR test shows the groups are significant. They help explain the data.

We can also look at the Pr(>Dev) for each univariate test, which I'm assuming are p-values. These show that the only motivation that is predicted on its own by these two groups is "Job", with a p-value of 0.02.

```
tryCatch(
  anova_pw <- anova(
    fit,
    resamp = "pit.trap",
    test = "LR",
    p.uni = "adjusted",
    pairwise.comp = X
  ),
  error = function(e) e
)
```

```
Time elapsed: 0 hr 0 min 0 sec
```

```
<simpleError in do_pairwise_comp(pairwise.comp, anova, object, resamp = resamp,     cor.type
```

Uh-oh. The pairwise comparison fails. Well, actually, that makes sense. The two groups must be different from each other, or else the global model fit wouldn't be significant. So we actually don't need this test.

Power analysis

So, we already know that the IT/Academic distinction is a pretty good predictor of the "Job" motivation. But since I already have the code for the z-test of proportions, I kind of want to do that test, too, just out of curiosity. I feel pretty confident that the regression is working well, but I will feel even better if the z-test is significant, too.

```r
n_it <- sum(it_acad$Role == "IT")
n_it_yes <- sum(
  it_acad$Role == "IT" &
    it_acad$Job == 1
)
p_it_yes <- n_it_yes / n_it

n_acad <- sum(it_acad$Role == "Academic")
n_acad_yes <- sum(
  it_acad$Role == "Academic" &
    it_acad$Job == 1
)
p_acad_yes <- n_acad_yes / n_acad

paste0("IT total: ", n_it)
```

```
[1] "IT total: 33"
```

```r
paste0("IT yes to 'Job': ", n_it_yes)
```

```
[1] "IT yes to 'Job': 7"
```

```r
paste0("IT proportion yes: ", p_it_yes)
```

```
[1] "IT proportion yes: 0.212121212121212"
```

```r
paste0("Academics total: ", n_acad)
```

```
[1] "Academics total: 114"
```

```r
paste0("Academics yes to 'Job': ", n_acad_yes)
```

```
[1] "Academics yes to 'Job': 57"
```

```r
paste0("Academics proportion yes: ", p_acad_yes)
```

```
[1] "Academics proportion yes: 0.5"
```

Power analysis

```r
h <- pwr::ES.h(p_it_yes, p_acad_yes)

pwr::pwr.2p2n.test(
  h = h,
  n1 = n_acad,
  sig.level = 0.05,
  power = 0.8,
  alternative = "less"
)
```

```

     difference of proportion power calculation for binomial distribution (arcsine transforma

              h = -0.6135304
             n1 = 114
             n2 = 19.18938
      sig.level = 0.05
          power = 0.8
    alternative = less

NOTE: different sample sizes
```

Great. We only need 19 IT people, but we have 33.

Now let's proceed with our hypothesis test.

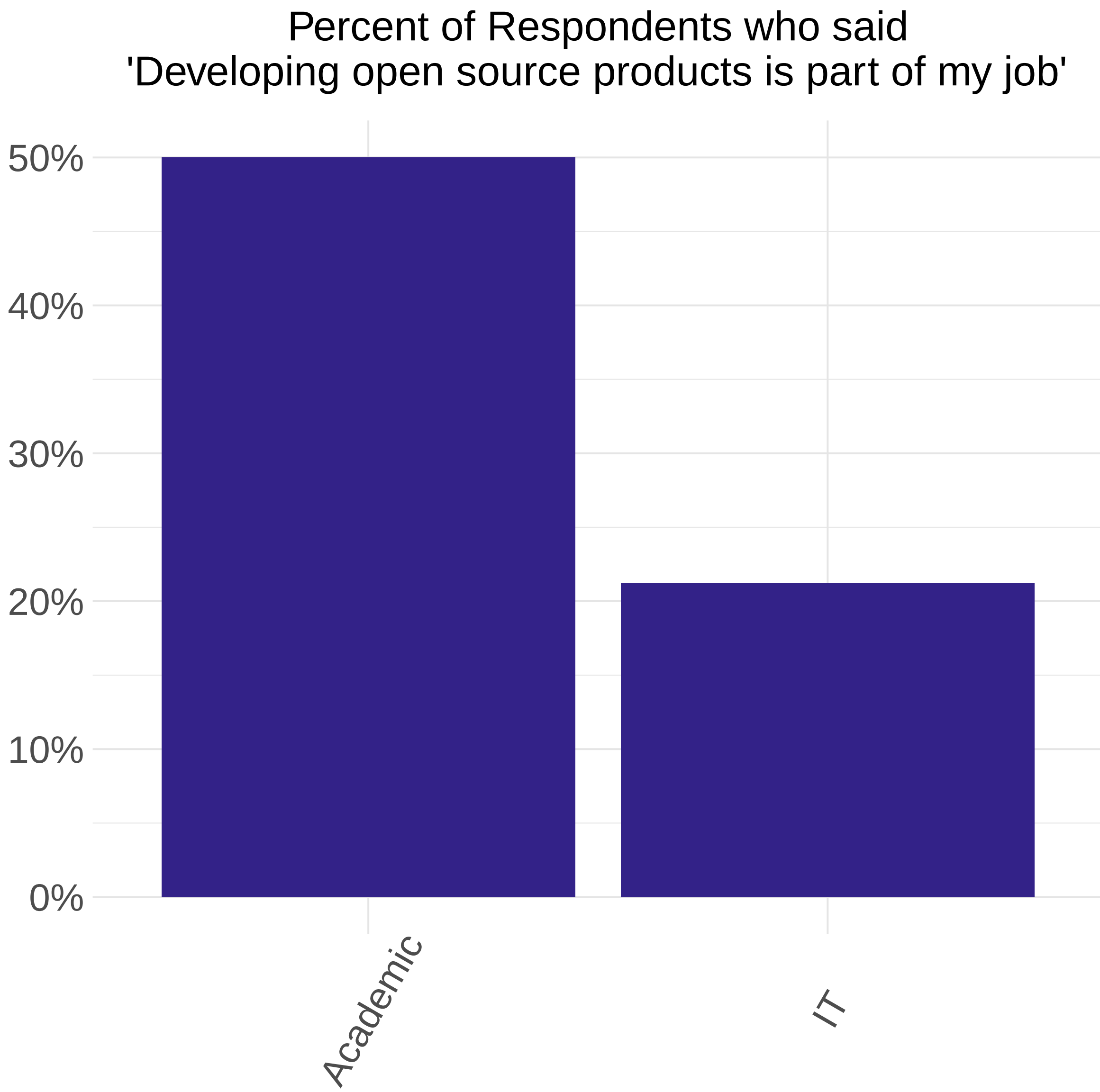
```r
pairwise_z_test_lessthan(
  it_acad,
  outcome_col = "Job",
  predictor_col = "Role",
  group1 = "IT",
  group2 = "Academic"
)
```

```
    2-sample test for equality of proportions with continuity correction

data:  c(y1, y2) out of c(n1, n2)
X-squared = 7.4964, df = 1, p-value = 0.003091
alternative hypothesis: less
95 percent confidence interval:
 -1.0000000 -0.1282156
sample estimates:
   prop 1    prop 2
0.2121212 0.5000000
```

Good! A pairwise z-test for proportions says the difference in "success rates" (proportion of yes answers) for IT (21% said yes) and academics (50% said yes) for the "Job" motivation is indeed significant.

I know this notebook should be for stats, and the other should be for plots, but it's just easier right now to make a plot here using the data structures I already have, instead of reconstructing them again in the other script.

```
to_plot <- data.frame(proportion_yes=c(p_acad_yes, p_it_yes), role=c("Academic", "IT"))

basic_bar_chart(to_plot,
  x_var = "role",
  y_var = "proportion_yes",
  title = "Percent of Respondents who said\n'Developing open source products is part of my jo
  show_bar_labels = FALSE,
  show_ticks_y = FALSE,
  show_axis_title_y = FALSE,
  show_axis_title_x = FALSE,
  show_grid = TRUE,
  percent = TRUE
)
```



Save the plot if desired.

```
save_plot("acad_it_simple.tiff", 9, 9)
```

Session Info

```r
sessionInfo()
```

```
R version 4.4.2 (2024-10-31)
Platform: aarch64-apple-darwin20
Running under: macOS Sequoia 15.4.1

Matrix products: default
BLAS:   /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/4.4-arm64/Resources/lib/libRlapack.dylib;  

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

time zone: America/Los_Angeles
tzcode source: internal

attached base packages:
[1] tools     stats     graphics  grDevices datasets  utils     methods  
[8] base     

other attached packages:
 [1] treemap_2.4-4         tidyr_1.3.1           stringr_1.5.1        
 [4] scales_1.4.0          readr_2.1.5           pwr_1.3-0            
 [7] patchwork_1.3.0       mvabund_4.2.1         languageserver_0.3.16
[10] here_1.0.1            gtools_3.9.5          fpc_2.2-13           
[13] forcats_1.0.0         factoextra_1.0.7      ggplot2_3.5.2        
[16] dplyr_1.1.4           corrplot_0.95         cluster_2.1.8.1      

loaded via a namespace (and not attached):
 [1] gtable_0.3.6       xfun_0.52          ggrepel_0.9.6     
 [4] processx_3.8.6     lattice_0.22-6     callr_3.7.6       
 [7] tzdb_0.5.0         vctrs_0.6.5        ps_1.9.1          
[10] generics_0.1.4     stats4_4.4.2       parallel_4.4.2    
[13] flexmix_2.3-20     tibble_3.2.1       DEoptimR_1.1-3-1  
[16] pkgconfig_2.0.3    data.table_1.17.6  RColorBrewer_1.1-3
[19] lifecycle_1.0.4    compiler_4.4.2     farver_2.1.2      
[22] statmod_1.5.0      httpuv_1.6.16      htmltools_0.5.8.1 
[25] class_7.3-22       yaml_2.3.10        later_1.4.2       
[28] pillar_1.10.2      prabclus_2.3-4     MASS_7.3-61       
```

```
[31] diptest_0.77-1      mclust_6.1.1        mime_0.13
[34] robustbase_0.99-4-1 tidyselect_1.2.1    digest_0.6.37
[37] stringi_1.8.7       purrr_1.0.4         kernlab_0.9-33
[40] labeling_0.4.3      rprojroot_2.0.4     fastmap_1.2.0
[43] grid_4.4.2          colorspace_2.1-1    cli_3.6.5
[46] magrittr_2.0.3      withr_3.0.2         promises_1.3.3
[49] tweedie_2.3.5       rmarkdown_2.29      igraph_2.1.4
[52] nnet_7.3-19         modeltools_0.2-24   hms_1.1.3
[55] shiny_1.11.0        evaluate_1.0.3      knitr_1.50
[58] rlang_1.1.6         Rcpp_1.0.14         xtable_1.8-4
[61] gridBase_0.4-7      glue_1.8.0          xml2_1.3.8
[64] renv_1.1.4          jsonlite_2.0.0      R6_2.6.1
```