Project sizes: exploratory plots

Overview

This notebook explores Q5: "How frequently have you contributed to projects of the following size?".

Import packages and utilities

```r
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")))
```

Load data

```r
sizes_raw <- load_qualtrics_data("clean_data/project_size_Q5.tsv")
other_quant <- load_qualtrics_data("clean_data/other_quant.tsv")
```

Wrangle data

Drop rows with no data

```
sizes <- exclude_empty_rows(sizes_raw)
nrow(sizes)
```

```
[1] 233
```

Let's create a long-format version for plotting.

```
sizes_long <- sizes %>%
  pivot_longer(
    cols = everything(),
    names_to  = "size",
    values_to = "frequency"
  )

sizes_long
```

```
# A tibble: 699 x 2
   size   frequency
   <chr>  <chr>
 1 Small  Relatively frequently
 2 Medium Occasionally
 3 Large  Relatively infrequently
 4 Small  Occasionally
 5 Medium Relatively infrequently
 6 Large  Never
 7 Small  Occasionally
 8 Medium Relatively infrequently
 9 Large  Never
10 Small  Relatively frequently
# i 689 more rows
```

Inspect data

Let's look at the counts.

```
sizes_counts <- sizes_long %>%
  count(size, frequency, name = "n")

sizes_counts[
  order(
    sizes_counts$n,
    decreasing = TRUE
  ),
]
```

```
# A tibble: 12 x 3
   size   frequency                    n
   <chr>  <chr>                    <int>
 1 Small  Relatively frequently      109
 2 Large  Never                       82
 3 Large  Relatively infrequently     74
 4 Medium Relatively infrequently     69
 5 Medium Occasionally                68
 6 Small  Occasionally                67
 7 Medium Relatively frequently       53
 8 Medium Never                       43
 9 Small  Relatively infrequently     41
10 Large  Occasionally                39
11 Large  Relatively frequently       38
12 Small  Never                       16
```

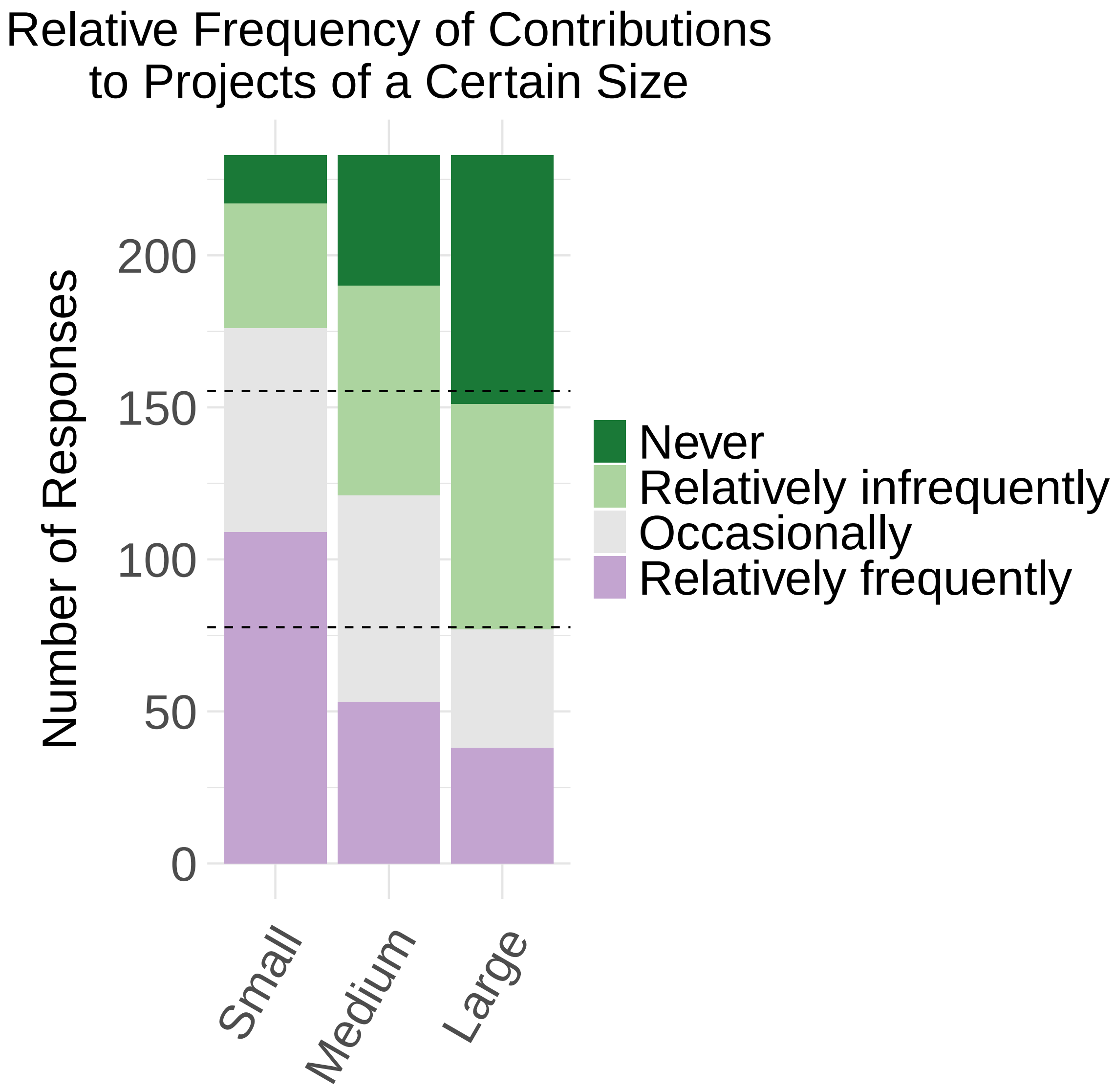
Reorder factor levels

```
ordered_freqs <- c(
  "Never",
  "Relatively infrequently",
  "Occasionally",
  "Relatively frequently"
)

sizes_counts$frequency <- factor(
  sizes_counts$frequency,
  levels = ordered_freqs
)

ordered_sizes <- c(
  "Small",
  "Medium",
  "Large"
)
```

```r
sizes_counts$size <- factor(
  sizes_counts$size,
  levels = ordered_sizes
)
```

Stacked bar chart

```r
stacked_bar <- stacked_bar_chart(
  sizes_counts,
  x_var = "size",
  y_var = "n",
  fill = "frequency",
  title = "Relative Frequency of Contributions\nto Projects of a Certain Size",
  cpalette = c(
    "#1a7937", # dark green
    "#acd49f", # light green
    "#e5e5e5", # light gray
    "#c3a4d0", # light purple
    "#752a82" # dark purple
  )
)
stacked_bar <- stacked_bar +
  geom_hline(yintercept = 155.4, linetype = "dashed", color = "black") +
  geom_hline(yintercept = 77.7, linetype = "dashed", color = "black")
stacked_bar
```



The dashed lines indicate 1/3 and 2/3 of the total number of responses.

Save the plot

```r
save_plot("proj_sizes_bar.tiff", 8, 6, p=stacked_bar)
```

Incorporate job category

```
sizes_job <- cbind(sizes_raw, other_quant$job_category)
# Rename column
names(sizes_job)[ncol(sizes_job)] <- "job_category"
# Filter out people who didn't answer either question
sizes_job <- exclude_empty_rows(sizes_job, strict = TRUE)
```

```
sizes_job_long <- sizes_job %>%
  pivot_longer(
    cols = -job_category,
    names_to = "size",
    values_to = "frequency"
  )
```

```
# three way cross tabs (xtabs) and flatten the table
# code from: https://ladal.edu.au/tutorials/regression/regression.html
ftable(xtabs(~ job_category + size + frequency, data = sizes_job_long))
```

```
                            frequency Never Occasionally Relatively frequently Relatively in
job_category         size
Faculty              Large               26            6                     8
                     Medium              13           17                    10
                     Small                6           17                    28
Grad Student         Large               11            7                     1
                     Medium               8           10                     2
                     Small                0            7                    14
Non-research Staff   Large               15           17                    20
                     Medium              11           28                    22
                     Small               10           25                    33
Other research staff Large               17            5                     8
                     Medium               6            8                    14
                     Small                0           11                    22
Post-Doc             Large                8            3                     1
                     Medium               1            4                     4
                     Small                0            6                     8
Undergraduate        Large                5            1                     0
                     Medium               4            1                     1
                     Small                0            1                     4
```

Maybe these data are more suited to line plots than to bar plots. Also, maybe we should fold in the smaller job categories, like we did with the regressions.

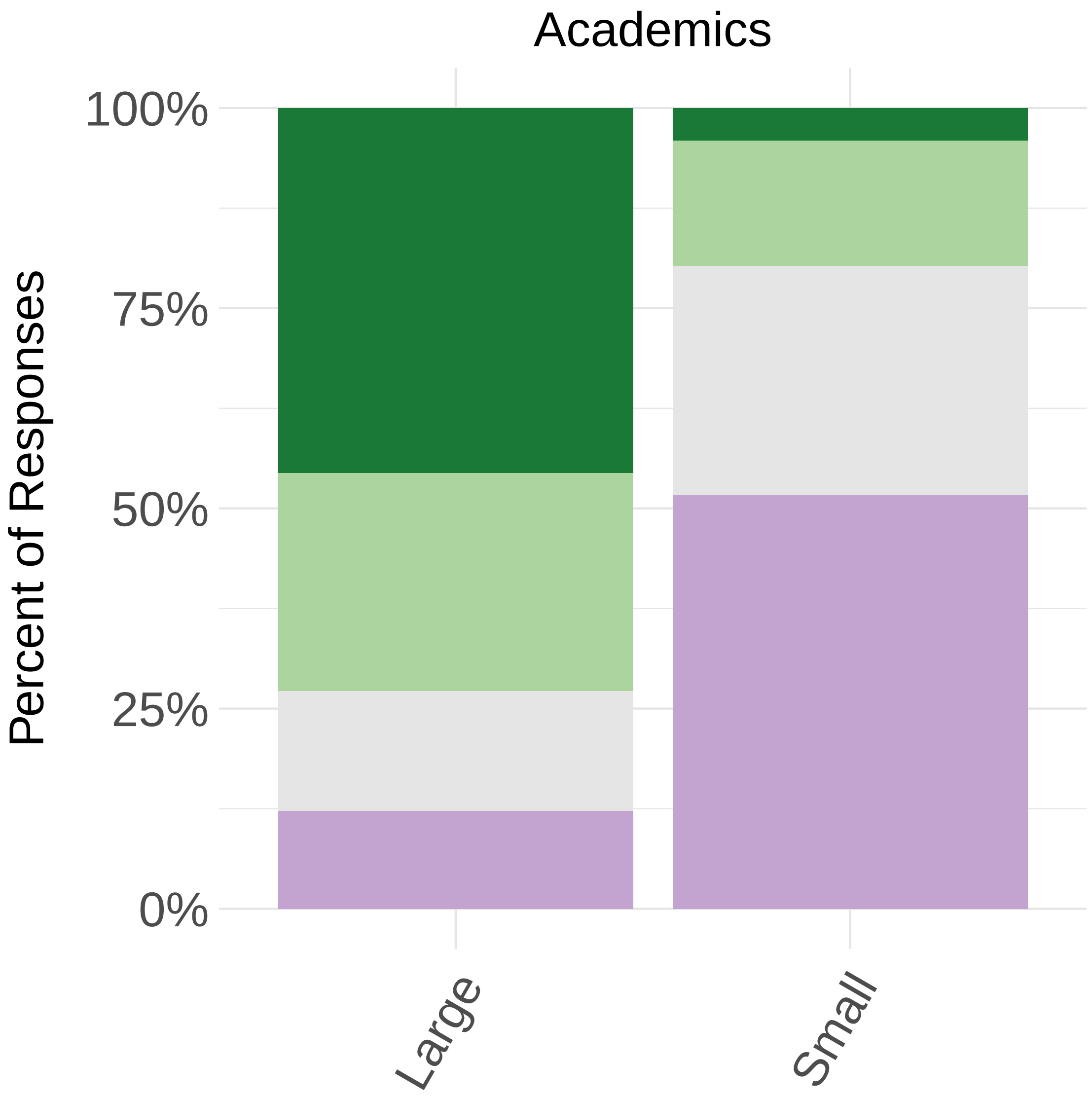
```r
combined <- sizes_job_long %>%
  mutate(
    job_category = recode(
      job_category,
      "Post-Doc" = "Postdocs and Staff Researchers",
      "Other research staff" = "Postdocs and Staff Researchers"
    )
  )

combined <- combined %>%
  mutate(
    job_category = recode(
      job_category,
      "Grad Student" = "Students",
      "Undergraduate" = "Students"
    )
  )
```

What if we separated this stacked bar into academics vs. non-research staff (or IT, maybe)? Maybe just do small and large, to make things visually simpler. Let's build each plot separately and then stitch them together using patchwork.

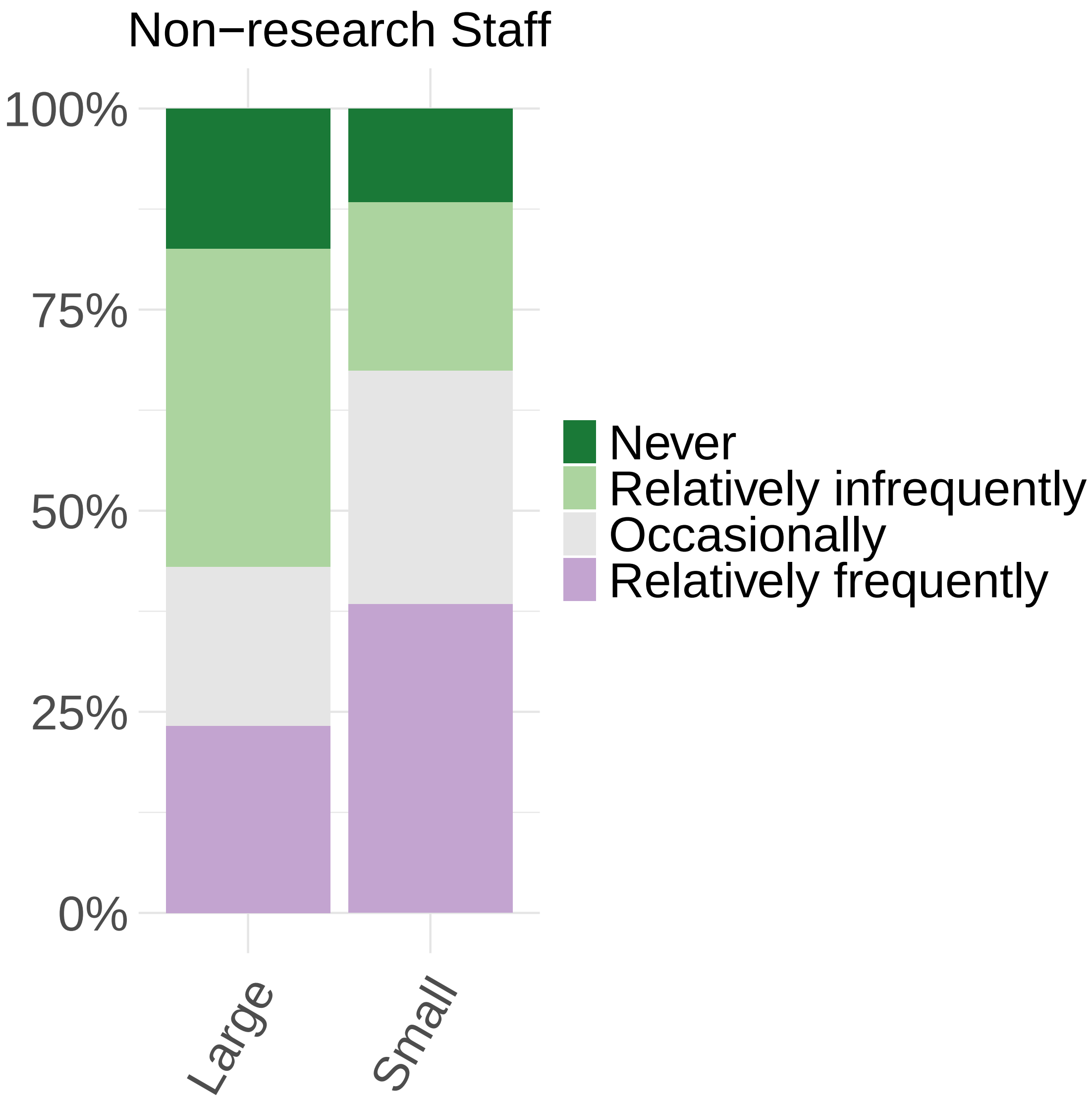
```r
# A version of the df where all academics
# have been relabeled to academic
combined_acad_nrstaff <- combined %>%
  mutate(
    job_category = recode(
      job_category,
      "Students" = "Academic",
      "Postdocs and Staff Researchers" = "Academic",
      "Faculty" = "Academic"
    )
  )

combined_acad_nrstaff$frequency <- factor(
  combined_acad_nrstaff$frequency,
  levels = ordered_freqs
  )
```

```r
acad_counts <- combined_acad_nrstaff %>%
  filter(job_category == "Academic") %>%
  filter(size != "Medium") %>%
  count(size, frequency, name = "n")

nrstaff_counts <- combined_acad_nrstaff %>%
  filter(job_category != "Academic") %>%
  filter(size != "Medium") %>%
  count(size, frequency, name = "n")
```

```r
stacked_bar_acad <- stacked_bar_chart(
  acad_counts,
  x_var = "size",
  y_var = "n",
  fill = "frequency",
  title = "Academics",
  ylabel = "Percent of Responses",
  proportional = TRUE,
  show_legend = FALSE,
  cpalette = c(
    "#1a7937", # dark green
    "#acd49f", # light green
    "#e5e5e5", # light gray
    "#c3a4d0", # light purple
    "#752a82" # dark purple
  )
)

stacked_bar_acad <- stacked_bar_acad +
  scale_y_continuous(labels = scales::percent)
# stacked_bar_acad <- stacked_bar_acad +
#   geom_hline(yintercept = 155.4, linetype = "dashed", color = "black") +
#   geom_hline(yintercept = 77.7, linetype = "dashed", color = "black")
stacked_bar_acad
```

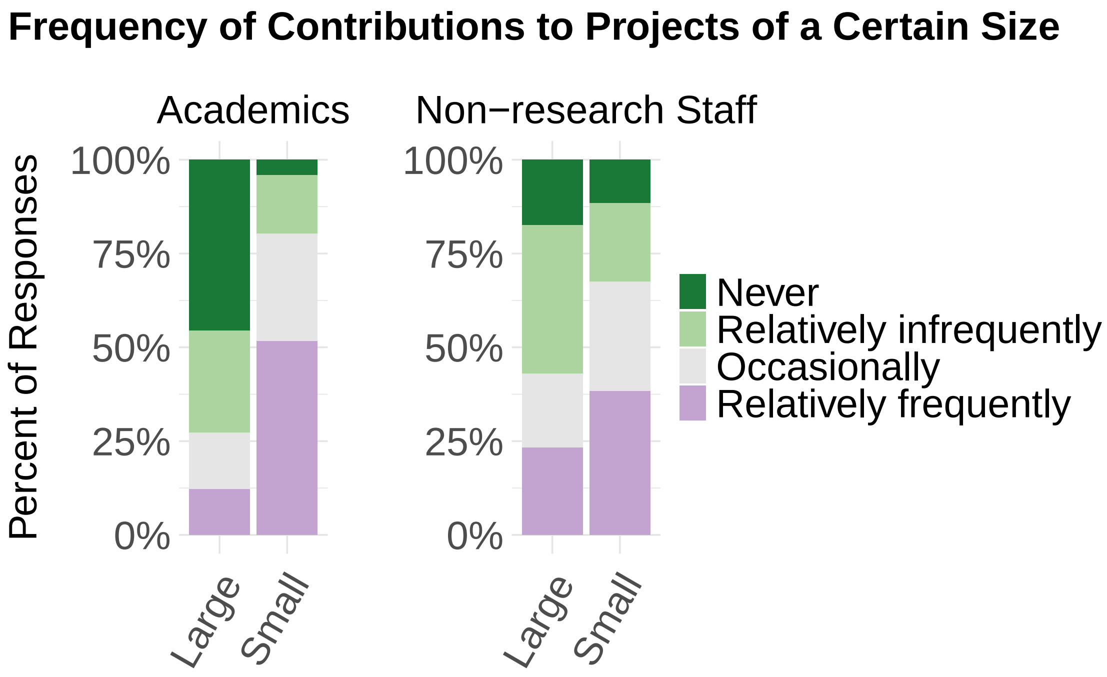


```
stacked_bar_nrstaff <- stacked_bar_chart(
  nrstaff_counts,
  x_var = "size",
  y_var = "n",
  fill = "frequency",
  title = "Non-research Staff",
  ylabel = "Percent of Responses",
  proportional = TRUE,
  show_axis_title_y = FALSE,
  cpalette = c(
    "#1a7937", # dark green
    "#acd49f", # light green
    "#e5e5e5", # light gray
    "#c3a4d0", # light purple
    "#752a82" # dark purple
  )
)

stacked_bar_nrstaff <- stacked_bar_nrstaff +
  scale_y_continuous(labels = scales::percent)

stacked_bar_nrstaff
```



Combine onto one plot

```r
p_combined <- patchwork::wrap_plots(
  stacked_bar_acad,
  stacked_bar_nrstaff
) +
  plot_annotation(
    title = "Frequency of Contributions to Projects of a Certain Size",
    theme = theme(plot.title = element_text(size = 24, face = "bold"))
  )

p_combined
```



Save the plot

```
save_plot("proj_sizes_acad_nrstaff.tiff", 14, 6, p=p_combined)
```

Panel of line plots

Recycling some old code I wrote many months ago to display stacked rows of line plots. Not sure if this will look decent.

Reorder factor levels

```
ordered_jobs <- c(
  "Students",
  "Postdocs and Staff Researchers",
  "Faculty",
  "Non-research Staff"
)

combined$size <- factor(combined$size, levels = ordered_sizes)
combined$frequency <- factor(combined$frequency, levels = ordered_freqs)
combined$job_category <- factor(combined$job_category, levels = ordered_jobs)
```

Recode frequency from categorical to a numeric score

```
combined_coded_all <- combined %>%
  mutate(
    frequency_score = recode(
      frequency,
      "Never" = 0L,
      "Relatively infrequently" = 1L,
      "Occasionally" = 2L,
      "Relatively frequently" = 3L
    )
  ) %>%
    select(-frequency)

combined_coded_all
```

```
# A tibble: 699 x 3
   job_category                     size   frequency_score
   <fct>                            <fct>            <int>
 1 Faculty                          Small                3
 2 Faculty                          Medium               2
 3 Faculty                          Large                1
 4 Postdocs and Staff Researchers Small                2
 5 Postdocs and Staff Researchers Medium               1
 6 Postdocs and Staff Researchers Large                0
 7 Postdocs and Staff Researchers Small                2
 8 Postdocs and Staff Researchers Medium               1
 9 Postdocs and Staff Researchers Large                0
10 Faculty                          Small                3
# i 689 more rows
```

Sum up frequency scores

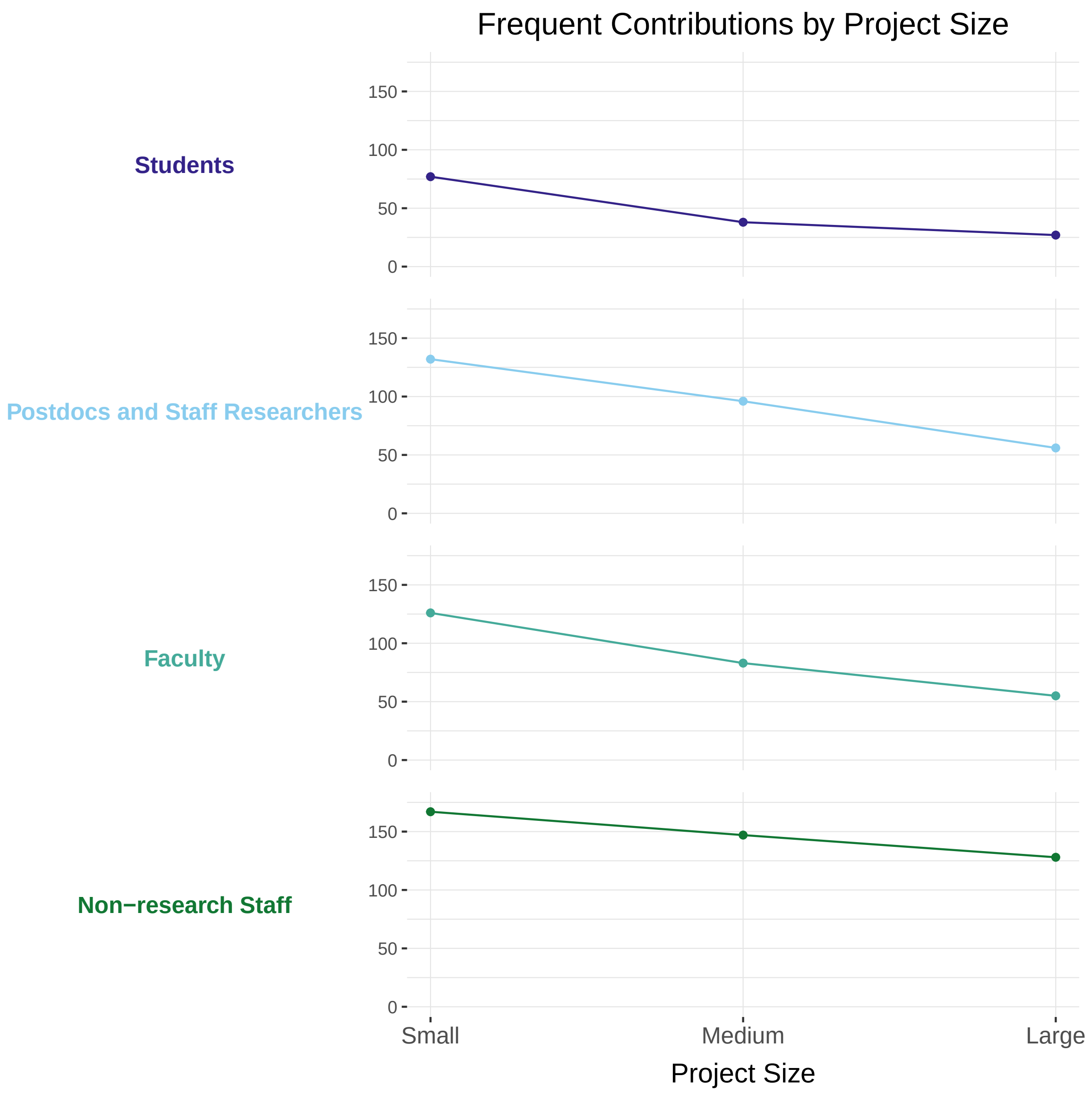
```
combined_scores <- combined_coded_all %>%
  count(job_category, size, wt = frequency_score, name = "total_score")

# Reorder factor levels
combined_scores$size <- factor(combined_scores$size, levels = ordered_sizes)

combined_scores
```

```
# A tibble: 12 x 3
   job_category                   size   total_score
   <fct>                          <fct>        <int>
 1 Students                       Small           77
 2 Students                       Medium          38
 3 Students                       Large           27
 4 Postdocs and Staff Researchers Small          132
 5 Postdocs and Staff Researchers Medium          96
 6 Postdocs and Staff Researchers Large           56
 7 Faculty                        Small          126
 8 Faculty                        Medium          83
 9 Faculty                        Large           55
10 Non-research Staff             Small          167
11 Non-research Staff             Medium         147
12 Non-research Staff             Large          128
```

Recycling some old code to create a stack of line plots.

```
labeled_colors <- setNames(as.list(COLORS), ordered_jobs)

lineplot <- function(df, current_job_cat) {
  x <- ggplot(
    subset(df, job_category == current_job_cat),
    aes(x = size, y = total_score, group = job_category, color = job_category)
  ) +
    geom_line() +
    geom_point() +
    ylim(0, 175) +
    scale_x_discrete(expand = c(0.025, 0.025)) +
    ylab(current_job_cat) +
    xlab("Project Size") +
    ggtitle("Frequent Contributions by Project Size") +
    scale_color_manual(values = c(labeled_colors[[current_job_cat]])) +
    # Use different theme options depending on whether this is
    # the first plot, a middle plot, or the last plot in the stack
    # I know this code is painfully "wet" as opposed to "d.r.y" but it gets the job done
    {
      if (current_job_cat == ordered_jobs[[1]]) {
        theme(
          axis.title.y = element_text(
            angle = 0,
            vjust = 0.5,
            color = labeled_colors[[current_job_cat]],
            size = 12,
            face = "bold"
          ),
          axis.title.x = element_blank(),
          axis.text.x = element_blank(),
          axis.ticks.x = element_blank(),
          panel.background = element_blank(),
          panel.grid.major = element_line(linewidth = 0.25, linetype = "solid", color = "gra
          panel.grid.minor = element_line(linewidth = 0.25, linetype = "solid", color = "gra
          plot.margin = unit(c(0.3, 0.5, 0, 0), "cm"),
          plot.title = element_text(hjust = 0.5, size = 16),
          legend.position = "none"
        )
      }
    } +
    {
      if (
        current_job_cat != ordered_jobs[[length(ordered_jobs)]] &
          current_job_cat != ordered_jobs[[1]]) {
        theme(
          axis.title.y = element_text(
            angle = 0,
            vjust = 0.5,
            color = labeled_colors[[current_job_cat]],
            size = 12,
            face = "bold"
          ),
          axis.title.x = element_blank(),
          axis.text.x = element_blank(),
          axis.ticks.x = element_blank(),
          panel.background = element_blank(),
          panel.grid.major = element_line(linewidth = 0.25, linetype = "solid", color = "gra
          panel.grid.minor = element_line(linewidth = 0.25, linetype = "solid", color = "gray
          plot.margin = unit(c(0.3, 0.5, 0, 0), "cm"),
          plot.title = element_blank(),
          legend.position = "none"
        )
      }
    } +
    {
      if (current_job_cat == ordered_jobs[[length(ordered_jobs)]]) {
        theme(
          axis.title.y = element_text(
            angle = 0,
            vjust = 0.5,
            color = labeled_colors[[current_job_cat]],
            size = 12,
            face = "bold"
          ),
          axis.title.x = element_text(size = 14, vjust = -0.5),
          axis.text.x = element_text(size = 12),
          panel.background = element_blank(),
          panel.grid.major = element_line(linewidth = 0.25, linetype = "solid", color = "gray
          panel.grid.minor = element_line(linewidth = 0.25, linetype = "solid", color = "gray
          plot.margin = unit(c(0.3, 0.5, 0.3, 0), "cm"),
          plot.title = element_blank(),
          legend.position = "none"
        )
      }
    }
}
```

```
plotlist <- lapply(
  ordered_jobs,
  function(x) lineplot(combined_scores, x)
)

patchwork::wrap_plots(plotlist, nrow = 4, ncol = 1)
```

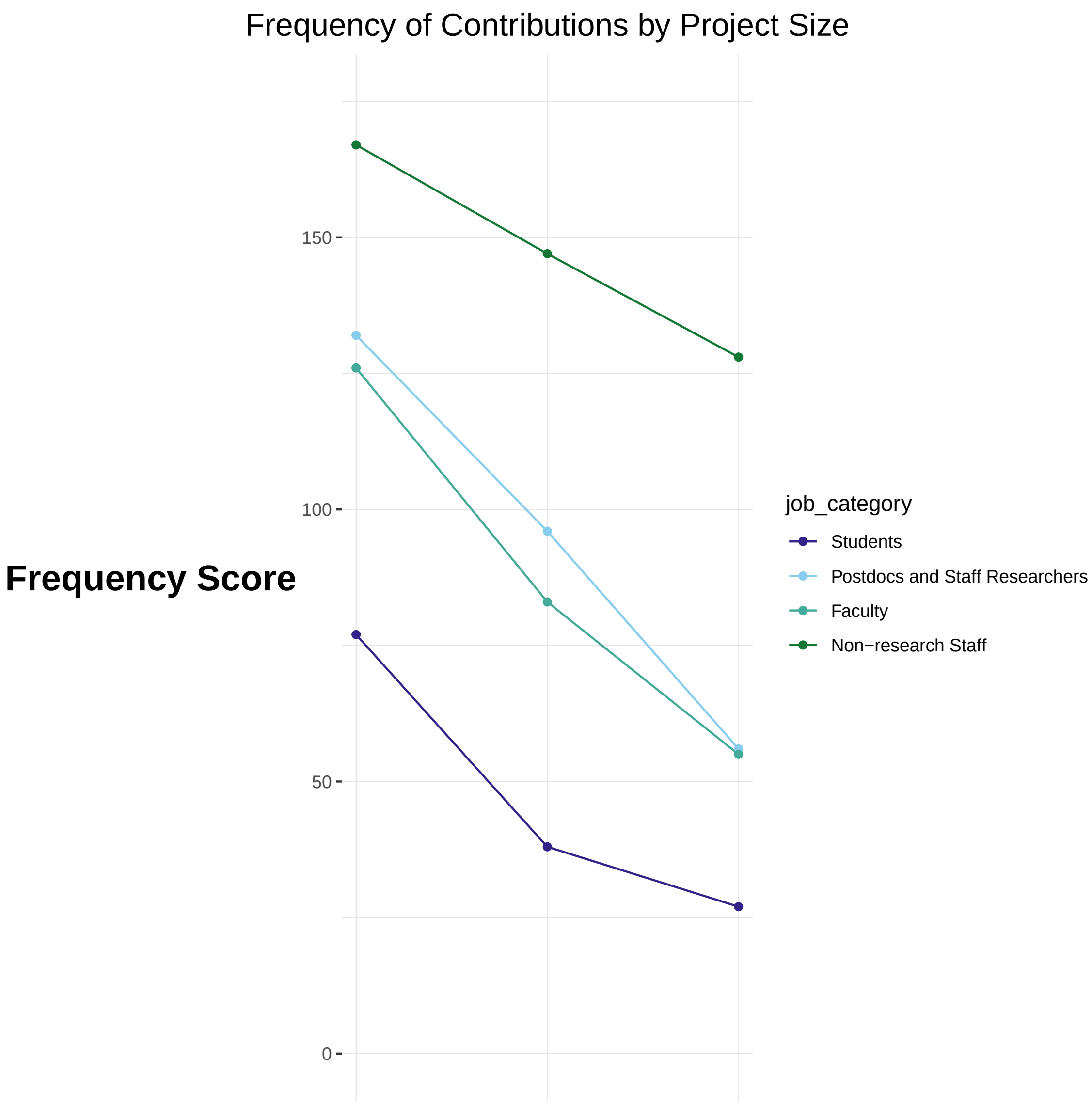


Eh, I think they would look better if they were all on the same plot.

Normal line plot

```
ggplot(
    combined_scores,
    aes(x = size, y = total_score, group = job_category, color = job_category)
```

```
) +
  geom_line() +
  geom_point() +
  ylim(0, 175) +
  scale_x_discrete(expand = c(0.025, 0.025)) +
  ylab("Frequency Score") +
  xlab("Project Size") +
  ggtitle("Frequency of Contributions by Project Size") +
  scale_color_manual(values = COLORS) +

  theme(
        axis.title.y = element_text(
          angle = 0,
          vjust = 0.5,
          size = 18,
          face = "bold"
        ),
        axis.title.x = element_blank(),
        axis.text.x = element_blank(),
        axis.ticks.x = element_blank(),
        panel.background = element_blank(),
        panel.grid.major = element_line(linewidth = 0.25, linetype = "solid", color = "gray
        panel.grid.minor = element_line(linewidth = 0.25, linetype = "solid", color = "gray
        plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm"),
        plot.title = element_text(hjust = 0.5, size = 16),
      )
```



Nah, still needs work. How about we just plot the trend for large projects?

Large projects

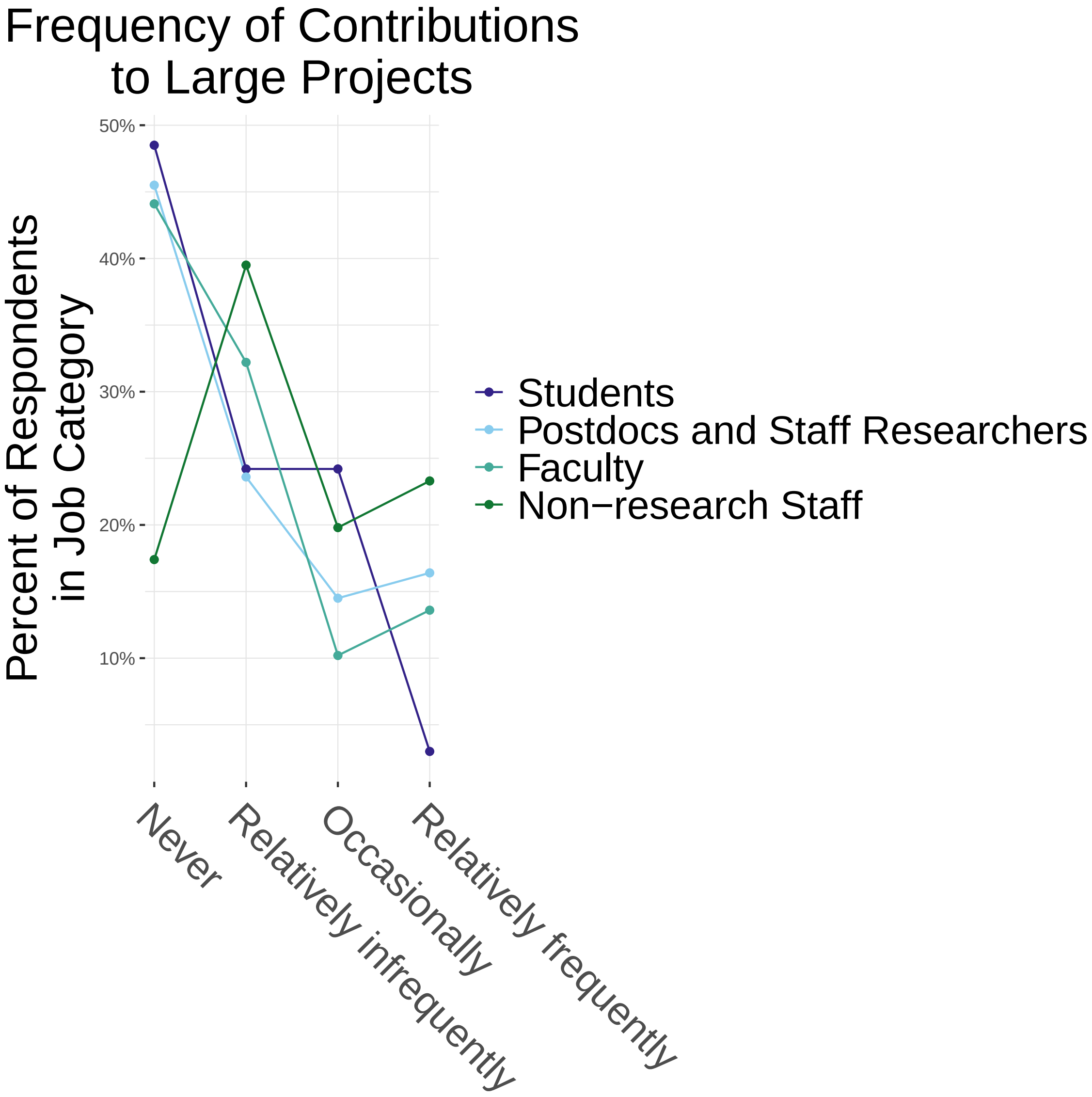
```
large <- subset(combined, size == "Large")
large_counts <- large %>%
  count(job_category, frequency, name = "n")
```

```r
large_counts <- large_counts %>%
  group_by(job_category) %>%
  mutate(perc_total = round(100 * n / sum(n), 1)) %>%
  ungroup()
```

```r
large_line <- ggplot(
    large_counts,
    aes(x = frequency, y = perc_total, group = job_category, color = job_category)
  ) +
    geom_line() +
    geom_point() +
    ylim(0, 50) +

    scale_x_discrete(expand = c(0.025, 0.025)) +
    scale_y_continuous(labels = scales::percent_format(accuracy = 1, scale = 1)) +
    scale_color_manual(values = COLORS) +

    ylab("Percent of Respondents\nin Job Category") +
    xlab("Project Size") +
    ggtitle("Frequency of Contributions\nto Large Projects") +


    theme(
          axis.title.y = element_text(
            size = 22,
            #face = "bold"
          ),
          axis.title.x = element_blank(),
          axis.text.x = element_text(
            angle = -45,
            hjust = 0,
            vjust = 1,
            size = 20,
            margin = margin(t = 6)),
          #axis.ticks.x = element_blank(),
          legend.text = element_text(size = 20),
          legend.title = element_blank(),
          panel.background = element_blank(),
          panel.grid.major = element_line(linewidth = 0.25, linetype = "solid", color = "gray
          panel.grid.minor = element_line(linewidth = 0.25, linetype = "solid", color = "gray
          plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm"),
          plot.title = element_text(hjust = 0.5, size = 24),
        )
```

```
Scale for y is already present.
Adding another scale for y, which will replace the existing scale.
```

```
large_line
```



Hard to discern a clear trend. Let's save the plot anyway.

Save the plot

```
save_plot("proj_sizes_large_line.tiff", 10, 6, p=large_line)
```

Medium projects

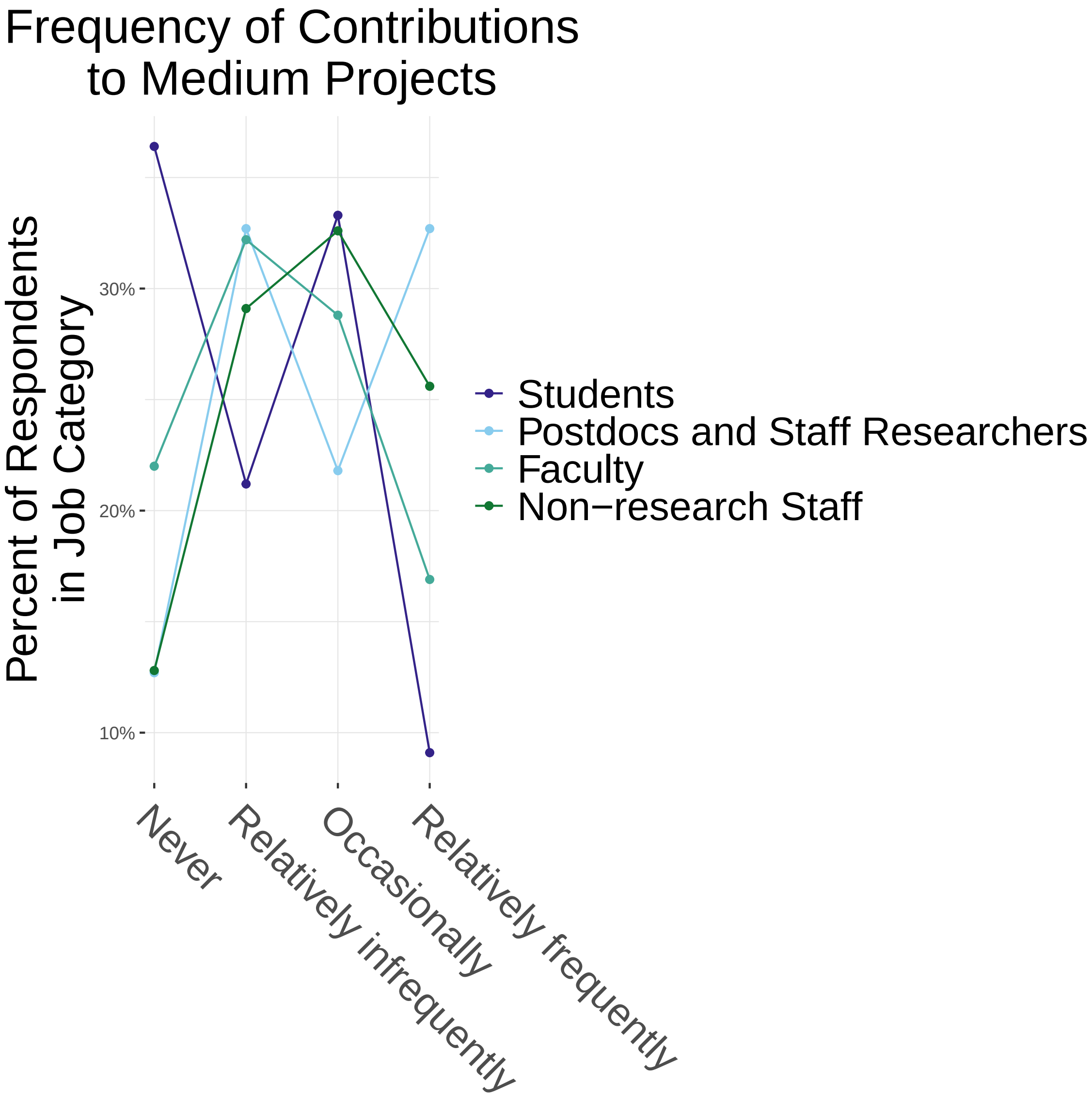
What about Medium projects? Do the same trends hold?

```
med <- subset(combined, size == "Medium")
med_counts <- med %>%
  count(job_category, frequency, name = "n")

med_counts <- med_counts %>%
  group_by(job_category) %>%
  mutate(perc_total = round(100 * n / sum(n), 1)) %>%
  ungroup()
```

```
med_line <- ggplot(
    med_counts,
    aes(x = frequency, y = perc_total, group = job_category, color = job_category)
  ) +
    geom_line() +
    geom_point() +
    ylim(0, 50) +

    scale_x_discrete(expand = c(0.025, 0.025)) +
    scale_y_continuous(labels = scales::percent_format(accuracy = 1, scale = 1)) +
    scale_color_manual(values = COLORS) +

    ylab("Percent of Respondents\nin Job Category") +
    xlab("Project Size") +
    ggtitle("Frequency of Contributions\nto Medium Projects") +


    theme(
          axis.title.y = element_text(
            size = 22,
            #face = "bold"
          ),
          axis.title.x = element_blank(),
          axis.text.x = element_text(
            angle = -45,
            hjust = 0,
            vjust = 1,
            size = 20,
            margin = margin(t = 6)),
          #axis.ticks.x = element_blank(),
          legend.text = element_text(size = 20),
          legend.title = element_blank(),
          panel.background = element_blank(),
          panel.grid.major = element_line(linewidth = 0.25, linetype = "solid", color = "gray
          panel.grid.minor = element_line(linewidth = 0.25, linetype = "solid", color = "gray
          plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm"),
          plot.title = element_text(hjust = 0.5, size = 24),
        )
```

```
Scale for y is already present.
Adding another scale for y, which will replace the existing scale.
```

```
med_line
```



Save the plot

```
save_plot("proj_sizes_med_line.tiff", 10, 6, p=med_line)
```

Small projects

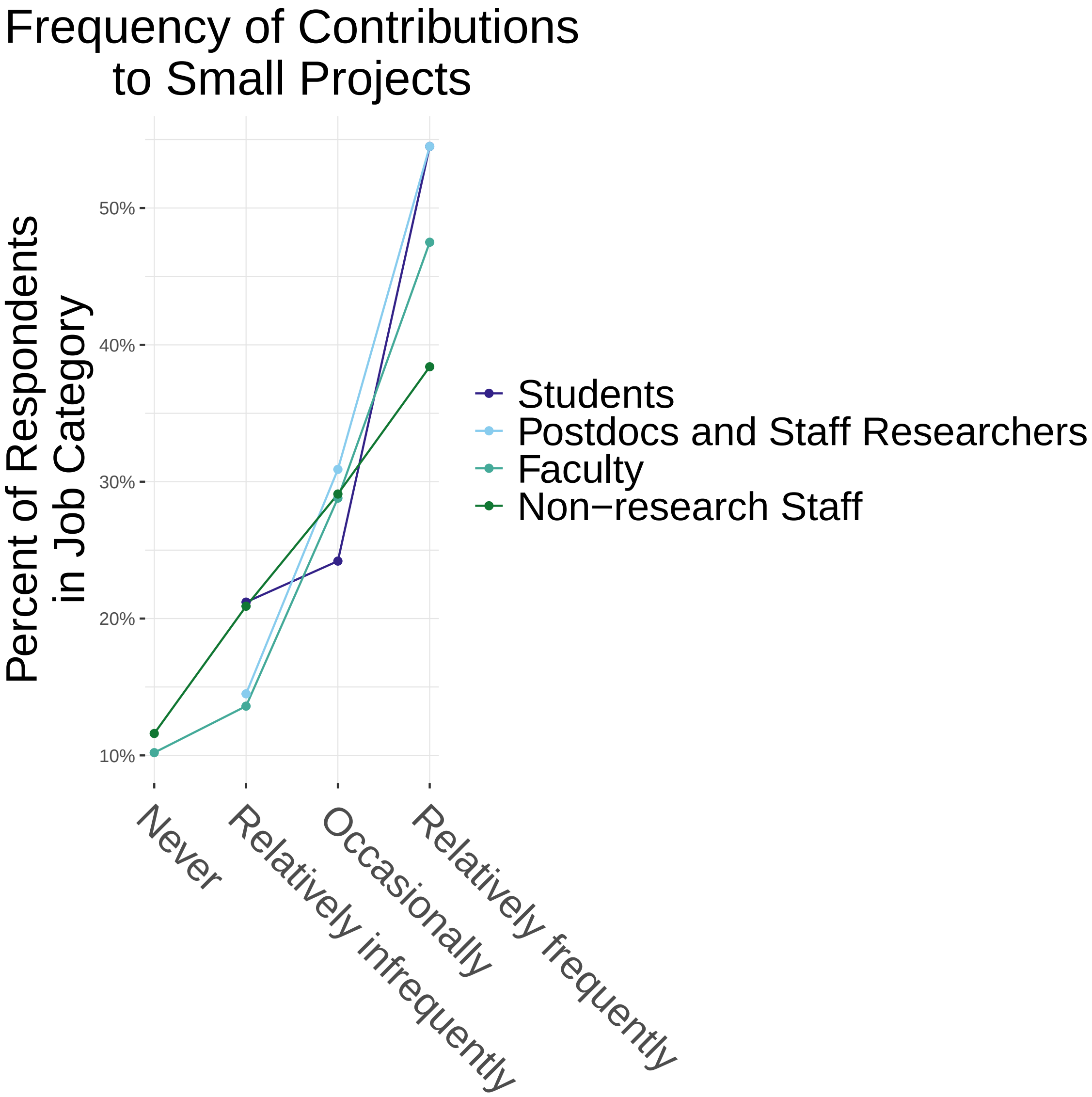
We've made it this far. We might as well look at small projects, too.

```
small <- subset(combined, size == "Small")
small_counts <- small %>%
  count(job_category, frequency, name = "n")

small_counts <- small_counts %>%
  group_by(job_category) %>%
  mutate(perc_total = round(100 * n / sum(n), 1)) %>%
  ungroup()
```

```
small_line <- ggplot(
    small_counts,
    aes(x = frequency, y = perc_total, group = job_category, color = job_category)
  ) +
    geom_line() +
    geom_point() +
    ylim(0, 50) +

    scale_x_discrete(expand = c(0.025, 0.025)) +
    scale_y_continuous(labels = scales::percent_format(accuracy = 1, scale = 1)) +
    scale_color_manual(values = COLORS) +

    ylab("Percent of Respondents\nin Job Category") +
    xlab("Project Size") +
    ggtitle("Frequency of Contributions\nto Small Projects") +


    theme(
          axis.title.y = element_text(
            size = 22,
            #face = "bold"
          ),
          axis.title.x = element_blank(),
          axis.text.x = element_text(
            angle = -45,
            hjust = 0,
            vjust = 1,
            size = 20,
            margin = margin(t = 6)),
          #axis.ticks.x = element_blank(),
          legend.text = element_text(size = 20),
          legend.title = element_blank(),
          panel.background = element_blank(),
          panel.grid.major = element_line(linewidth = 0.25, linetype = "solid", color = "gray
          panel.grid.minor = element_line(linewidth = 0.25, linetype = "solid", color = "gray
          plot.margin = unit(c(0.5, 0.5, 0.5, 0.5), "cm"),
          plot.title = element_text(hjust = 0.5, size = 24),
        )
```

```
Scale for y is already present.
Adding another scale for y, which will replace the existing scale.
```

```
small_line
```



Wow, that's much prettier.

Save the plot

```
save_plot("proj_sizes_small_line.tiff", 10, 6, p=small_line)
```

I'd like to know whether the proportion of academics who contribute to large projects with some frequency is significantly lower than the proportion of non-research staff who contribute to large projects with some frequency.

```
combined_counts <- combined %>%
  count(job_category, size, frequency, name = "n")

res <- combined_counts %>%
  filter(size == "Large") %>%
  mutate(
    group = if_else(job_category == "Non-research Staff",
                    "Non-research Staff", "Academics"),
    freq2 = if_else(frequency == "Never", "Never", "Other")
  ) %>%
  group_by(group, freq2) %>%
  summarise(n = sum(n), .groups = "drop_last") %>%
  mutate(prop = n / sum(n)) %>%
  ungroup()

# 2×2 table: proportions for each group
res_wide <- res %>%
  select(group, freq2, prop) %>%
  pivot_wider(names_from = freq2, values_from = prop) %>%
  arrange(match(group, c("Non-research Staff", "Academics")))

res_wide
```

```
# A tibble: 2 x 3
  group              Never Other
  <chr>              <dbl> <dbl>
1 Non-research Staff 0.174 0.826
2 Academics          0.456 0.544
```

Hmm. Seems promising. We should probably do a regression…