Project Types

Overview

This analysis is of Q7 about the types of projects people have contributed to.

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
ptypes_raw <- load_qualtrics_data("clean_data/project_types_Q7.tsv")
sizes_raw <- load_qualtrics_data("clean_data/project_size_Q5.tsv")
other_quant <- load_qualtrics_data("clean_data/other_quant.tsv")
qual <- load_qualtrics_data("qual_responses.tsv")
```

Wrangle data

Discard rows from people who didn't answer the Q about project types

```
ptypes <- ptypes_raw[which(rowSums(ptypes_raw) != 0),]
```

Discard rows from people who didn't answer the Q about project types OR who didn't answer the Q about job category

```
ptypes_job <- cbind(ptypes_raw, other_quant$job_category)
# Rename column
names(ptypes_job)[ncol(ptypes_job)] <- "job_category"

keep1 <- which(rowSums(ptypes_raw) != 0)
keep2 <- which(ptypes_job$job_category != "")

#Only keep people who answered both questions
keep <- intersect(keep1, keep2)
ptypes_job <- ptypes_job[keep,]
nrow(ptypes_job)
```

```
[1] 233
```

Inspect data

```
counts <- data.frame(colSums(ptypes))
names(counts)[1] <- "count"
counts <- counts %>% arrange(desc(count))
counts
```

```
                                   count
Libraries, packages, or frameworks   157
Applications                         156
Website code                         106
Plug-ins or extensions                98
Automation scripts                    95
Hardware                              30
Other                                 17
```

```
ordered_proj_types <- rownames(counts)

to_print <- cbind(ordered_proj_types, counts)
names(to_print)[1] <- "Project type"
```

Save for supplement

```r
write_df_to_file(to_print, "supplementary_tables/proj_type_counts.tsv")
```

On average, how many project types does each person contribute to?

```r
get_mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}
mean(rowSums(ptypes))
```

```
[1] 2.828326
```

```r
median(rowSums(ptypes))
```

```
[1] 3
```

```r
get_mode(rowSums(ptypes))
```

```
[1] 3
```

So, about three project types, on average.

Bring in job category

Let's plot the distribution of project types for each job. Since there are very different sample sizes among the groups, we'll plot the proportion of each group that selected each project type.

```r
long_data <- ptypes_job %>%
  pivot_longer(
    cols = -job_category,
    names_to = "project_type",
    values_to = "flag"
  ) %>%
  filter(flag == 1) %>%
  select(project_type, job_category)

long_data
```

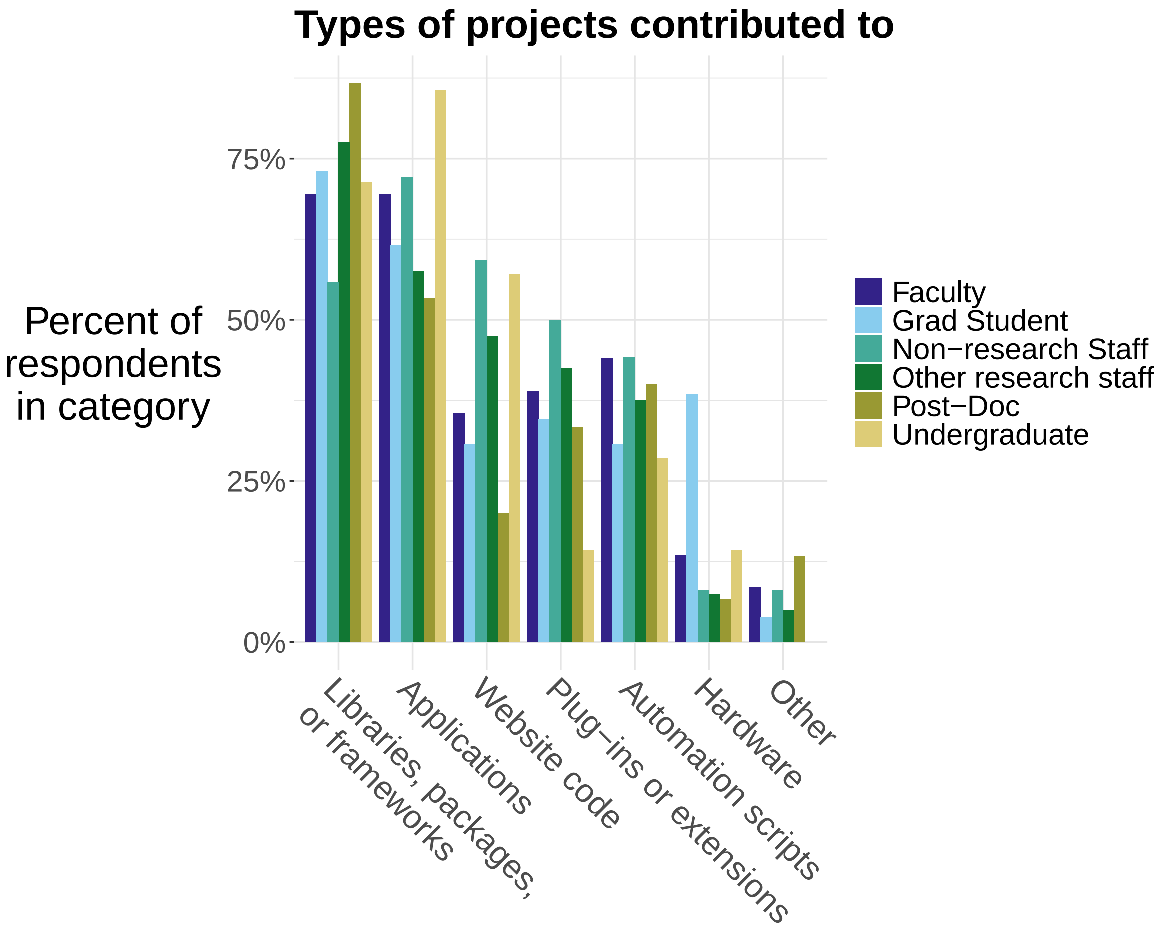
```
# A tibble: 659 x 2
   project_type                      job_category
   <chr>                             <chr>
 1 Applications                      Faculty
 2 Plug-ins or extensions            Faculty
 3 Libraries, packages, or frameworks Faculty
 4 Automation scripts                Faculty
 5 Libraries, packages, or frameworks Post-Doc
 6 Applications                      Other research staff
 7 Website code                      Other research staff
 8 Plug-ins or extensions            Other research staff
 9 Libraries, packages, or frameworks Other research staff
10 Automation scripts                Other research staff
# i 649 more rows
```

```
get_proportion_of_job_category <- function(x) {
    tmp <- ptypes_job %>% filter(job_category == x)
    tmp <- tmp %>% select(-job_category)
    sums <- colSums(tmp)
    return(
        sums / nrow(tmp)
    )
}

props <- as.data.frame(
    sapply(
    unique(ptypes_job$job_category),
    function(x) get_proportion_of_job_category(x)
    )
)

props$project_type <- rownames(props)

props_long <- props %>%
    pivot_longer(
        cols = -project_type,
        names_to = "job_category",
        values_to = "proportion"
    )
```

Reorder factor levels

```r
props_long$project_type <- factor(
    props_long$project_type,
    levels = ordered_proj_types
)
```

```r
ggplot(props_long, aes(
    fill = job_category,
    y = proportion,
    x = project_type
  )
) +
  geom_bar(position = "dodge", stat = "identity") +
  scale_fill_manual(values = COLORS) + # from utils.R
  scale_y_continuous(labels = scales::percent) +
  scale_x_discrete(
  # add whitespace to long labels
    labels = ~ str_replace(
      .x,
      fixed("Libraries, packages, or frameworks"),
      "Libraries, packages,\nor frameworks"
    )
  ) +
  labs(y = "Percent of\nrespondents\nin category") +
  ggtitle("Types of projects contributed to") +
  theme(
    axis.title.x = element_blank(),
    axis.title.y = element_text(angle = 0, vjust = 0.5, size = 24),
    #axis.text.x = element_text(angle = 60, vjust = 0.6, size = 18),
    axis.text.x = element_text(
      angle = -45,
      hjust = 0,
      vjust = 1,
      size = 20,
      margin = margin(t = 0)
    ),
    axis.text.y = element_text(size = 18),
    axis.ticks.x = element_blank(),
    legend.title = element_blank(),
    legend.text = element_text(size = 18),
    panel.background = element_blank(),
    panel.grid = element_line(linetype = "solid", color = "gray90"),
    plot.title = element_text(hjust = 0, size = 24, face = "bold"),
    plot.margin = unit(c(0.3, 0.3, 0.3, 0.3), "cm")
  )
```



This plot feels a bit busy. Let's make a simpler version by combining some of the job categories. Let's also make it horizontal, which will make the labels easier to read.

```
combined <- props_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"
    )
  )
```

Reorder factor levels

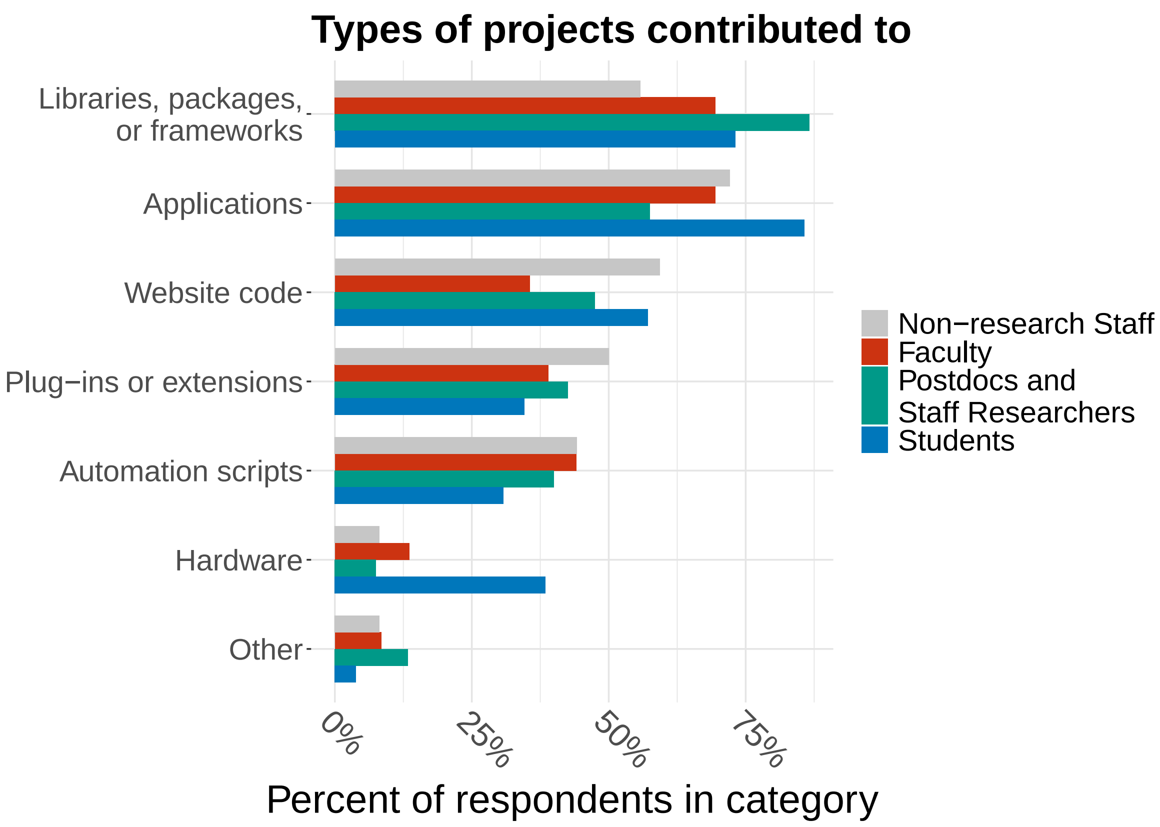
```
combined$project_type <- factor(
  combined$project_type,
  levels = rev(ordered_proj_types))

combined$job_category <- factor(
    combined$job_category,
    levels = c(
        "Students",
        "Postdocs and Staff Researchers",
        "Faculty",
        "Non-research Staff"
    )
)
```

```
cpalette <- c(
  "#0077bb", # medium blue
  "#009988", # green-blue
  "#cc3311", # red
  "#c6c6c6" # gray
)
```

```
grouped_plot <- ggplot(
  combined,
  aes(fill = job_category, y = proportion, x = project_type)
) +
  geom_bar(position = "dodge", stat = "identity", width = 0.75) +
  scale_fill_manual(
    values = cpalette,
    # add whitespace to long labels
    labels = function(x) {
```

```
    str_replace(
      x,
      fixed("Postdocs and Staff Researchers"),
      "Postdocs and\nStaff Researchers"
    )
  }
) +
scale_y_continuous(labels = scales::percent) +
scale_x_discrete(
# add whitespace to long labels
  labels = ~ str_replace(
    .x,
    fixed("Libraries, packages, or frameworks"),
    "Libraries, packages,\nor frameworks"
  )
) +
labs(y = "Percent of respondents in category") +
ggtitle("Types of projects contributed to") +
coord_flip() +
theme(
  axis.title.y = element_blank(),
  axis.title.x = element_text(angle = 0, vjust = 0.5, size = 24),
  #axis.text.x = element_text(angle = 60, vjust = 0.6, size = 18),
  axis.text.x = element_text(
    angle = -45,
    hjust = 0,
    vjust = 1,
    size = 20,
    margin = margin(t = 0)
  ),
  axis.text.y = element_text(size = 18),
  axis.ticks.x = element_blank(),
  legend.title = element_blank(),
  legend.text = element_text(size = 18),
  panel.background = element_blank(),
  panel.grid = element_line(linetype = "solid", color = "gray90"),
  plot.title = element_text(hjust = 0, size = 24, face = "bold"),
  plot.margin = unit(c(0.3, 0.3, 0.3, 0.3), "cm")
) +
# reverse legend so bar order matches label order
guides(fill = guide_legend(reverse = TRUE))
```

```
grouped_plot
```



Save the plot

```
save_plot("proj_types_by_job.tiff", 10, 6, p=grouped_plot)
```

A hunch: what about people who contribute to large projects relatively frequently?

Just following a hunch that it might be interesting to look at people who said they contribute to large projects relatively frequently, and then look at the types of projects they contribute to. We didn't directly ask them the types of large projects they contribute to, so this is sort of circumstantial evidence. I think this would only be interesting if there's a really clear trend.

```
ptypes_sizes <- cbind(ptypes_raw, sizes_raw)
head(ptypes_sizes)
```

```
  Applications Other Website code Plug-ins or extensions
1            1     0            0                      1
2            0     0            0                      0
3            1     0            1                      1
4            1     0            1                      1
5            1     0            0                      1
6            0     0            0                      0
  Libraries, packages, or frameworks Automation scripts Hardware
1                                  1                  1        0
2                                  1                  0        0
3                                  1                  1        1
4                                  1                  0        0
5                                  0                  1        0
6                                  0                  0        0
                  Small                  Medium                    Large
1 Relatively frequently            Occasionally Relatively infrequently
2          Occasionally Relatively infrequently                    Never
3          Occasionally Relatively infrequently                    Never
4 Relatively frequently Relatively infrequently                    Never
5 Relatively frequently            Occasionally Relatively infrequently
6
```

```
ptypes_sizes_large <- subset(ptypes_sizes, Large == "Relatively frequently")
nrow(ptypes_sizes_large)
```

```
[1] 38
```

Oof, the data are pretty sparse, with only 38 responses, which we'll be spreading across 7 project types. Let's beef it up a little by including folks who said they occasionally contribute to large projects.

```
ptypes_sizes_medlarge <- subset(
  ptypes_sizes,
  Large == "Relatively frequently" | Large == "Occasionally"
)
nrow(ptypes_sizes_medlarge)
```

```
[1] 77
```

That's better.

```
# We don't need the proj sizes columns anymore
ptypes_medlarge <- ptypes_sizes_medlarge %>%
  select(all_of(ordered_proj_types))

counts_medlarge <- data.frame(colSums(ptypes_medlarge))
names(counts_medlarge)[1] <- "count"
counts_medlarge <- counts_medlarge %>% arrange(desc(count))
counts_medlarge
```

```
                                   count
Libraries, packages, or frameworks    52
Applications                          52
Website code                          44
Plug-ins or extensions                35
Automation scripts                    31
Hardware                              11
Other                                  6
```

Meh, not very interesting. Pretty consistent with the data from the overall pool.

Question: what's with all these students building hardware?

I'd like to see what campuses these students are from. Maybe they are all friends contributing to the same project.

```
big_data <- cbind(ptypes_raw, other_quant)
students_hardware <- big_data %>%
  filter(
    Hardware == 1
  ) %>%
  filter(
    job_category == "Undergraduate" |
    job_category == "Grad Student"
  )

students_hardware
```

```
   Applications Other Website code Plug-ins or extensions
1             1     0            0                      0
```

```
2             1     0            0                      1
3             1     0            0                      0
4             0     0            0                      0
5             1     0            0                      1
6             1     0            0                      0
7             0     0            0                      0
8             0     0            0                      1
9             0     1            0                      0
10            1     0            0                      1
11            1     0            1                      1
   Libraries, packages, or frameworks Automation scripts Hardware
1                                   0                  0        1
2                                   1                  1        1
3                                   0                  0        1
4                                   1                  0        1
5                                   0                  0        1
6                                   1                  0        1
7                                   1                  1        1
8                                   1                  0        1
9                                   1                  0        1
10                                  1                  1        1
11                                  1                  1        1
             campus      favorite_solution    field_of_study  job_category
1    UC Los Angeles  Industry partnerships       Math and CS  Grad Student
2    UC Los Angeles  Sustainability grants Physical sciences  Grad Student
3    UC Los Angeles   A learning community Physical sciences  Grad Student
4    UC Los Angeles  Sustainability grants Physical sciences  Grad Student
5    UC Los Angeles Computing environments       Math and CS  Grad Student
6    UC Los Angeles Computing environments   Social sciences  Grad Student
7  UC San Francisco  Sustainability grants     Life sciences  Grad Student
8  UC Santa Barbara  Sustainability grants     Life sciences  Grad Student
9          UC Davis  Sustainability grants Physical sciences  Grad Student
10     UC San Diego   Help finding funding       Math and CS Undergraduate
11    UC Santa Cruz Computing environments       Math and CS  Grad Student
   staff_categories
1
2
3
4
5
6
7
8
```

```
9
10
11
```

Hm, okay. Only at most three students are from the same campus and broad field of study. I'm kind of curious about their academic subfields, particularly these UCLA students. If they're all from the same subfield, that would be consistent with the possibility of sampling bias, e.g., one hardware student sent the survey link out to their friends. If they're from different subfields, that doesn't rule out the possibility of sampling bias, but it seems like sampling bias would be a bit less likely.

```r
big_data2 <- cbind(ptypes_raw, other_quant, qual)
students_hardware2 <- big_data2 %>%
  filter(
    Hardware == 1
  ) %>%
  filter(
    job_category == "Undergraduate" |
    job_category == "Grad Student"
  ) %>%
    select (Hardware, campus, field_of_study, subfield)

students_hardware2
```

```
   Hardware           campus    field_of_study               subfield
1         1   UC Los Angeles       Math and CS       Computer Science
2         1   UC Los Angeles Physical sciences             Geophysics
3         1   UC Los Angeles Physical sciences Mechanical Engineering
4         1   UC Los Angeles Physical sciences         Nanotechnology
5         1   UC Los Angeles       Math and CS mechanical engineering
6         1   UC Los Angeles   Social sciences           econometrics
7         1 UC San Francisco     Life sciences         bioinformatics
8         1 UC Santa Barbara     Life sciences                Ecology
9         1         UC Davis Physical sciences      materials science
10        1     UC San Diego       Math and CS Electrical Engineering
11        1    UC Santa Cruz       Math and CS         Bioinformatics
```

Hm, okay, well, it doesn't scream "a bunch of friends from the same department", but it also doesn't rule out sampling bias.

Can we say whether this high rate of hardware contributors is statistically significant? Let's compare them to the group with the next-highest rate of hardware contributors: faculty.

Quick statistics

Let's start with a power analysis to see whether we have an adequate sample size.

```
n_grad <- sum(ptypes_job$job_category == "Grad Student")
n_grad_yes <- sum(
  ptypes_job$job_category == "Grad Student" &
    ptypes_job$Hardware == 1
)

n_faculty <- sum(ptypes_job$job_category == "Faculty")
n_faculty_yes <- sum(
  ptypes_job$job_category == "Faculty" &
    ptypes_job$Hardware == 1
)

# Sanity check
n_grad
```

```
[1] 26
```

```
n_grad_yes
```

```
[1] 10
```

```
n_faculty
```

```
[1] 59
```

```
n_faculty_yes
```

```
[1] 8
```

```
p_grad_yes <- n_grad_yes / n_grad
p_faculty_yes <- n_faculty_yes / n_faculty
```

Calculate Cohen's h, the effect size.

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

```
[1] 0.5837194
```

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

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

```

     difference of proportion power calculation for binomial distribution (arcsine transforma

              h = 0.5837194
             n1 = 26
             n2 = 60.06115
      sig.level = 0.05
          power = 0.8
    alternative = greater

NOTE: different sample sizes
```

So we would need 60 faculty to achieve 80% power.

```r
n_faculty
```

```
[1] 59
```

We have 59. Good enough for me.

```r
# Count total and 'yes' outcomes for each group
n1 <- sum(ptypes_job[["job_category"]] == "Grad Student")
y1 <- sum(
  ptypes_job[["job_category"]] == "Grad Student" & ptypes_job[["Hardware"]] == 1
)

n2 <- sum(ptypes_job[["job_category"]] == "Faculty")
y2 <- sum(
  ptypes_job[["job_category"]] == "Faculty" & ptypes_job[["Hardware"]] == 1
)

# Perform the one-sided prop test (testing if group1 > group2)
stats::prop.test(
  x = c(y1, y2),
  n = c(n1, n2),
  alternative = "greater",
)
```

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

data:  c(y1, y2) out of c(n1, n2)
X-squared = 5.2957, df = 1, p-value = 0.01069
alternative hypothesis: greater
95 percent confidence interval:
 0.04809964 1.00000000
sample estimates:
   prop 1    prop 2 
0.3846154 0.1355932 
```

Cool beans. The difference in proportions is statistically significant, according to a simple z-test. There is of course a difference between statistical significance and real-world signficance for practical purposes, however.