Challenges + job category

Overview

Secondary analysis of survey Q9: "How frequently have you encountered the following challenges while working on open-source projects?"

In this script, I am considering challenges in light of job category.

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
challenges <- load_qualtrics_data("clean_data/challenges_Q9.tsv")
other_quant <- load_qualtrics_data("clean_data/other_quant.tsv")
```

Wrangle data

```r
challenges_and_job <- challenges
challenges_and_job$job_category <- other_quant$job_category

head(challenges_and_job)
```

```
  Coding time Documentation time Managing issues Attracting users  Recognition
1      Always             Always          Always           Always       Always
2  Frequently       Occasionally    Occasionally     Occasionally Occasionally
3  Frequently             Always    Occasionally           Always Occasionally
4      Always             Always      Frequently     Occasionally   Frequently
5      Always             Always          Rarely     Occasionally   Frequently
6
      Hiring     Security Finding peers Finding mentors Education time
1     Always       Always        Always          Always         Always
2     Rarely   Frequently  Occasionally      Frequently     Frequently
3 Frequently   Frequently  Occasionally    Occasionally         Rarely
4     Always Occasionally        Rarely          Rarely     Frequently
5      Never        Never         Never           Never         Always
6
  Educational resources        Legal Finding funding Securing funding
1                Always       Always          Always           Always
2            Frequently   Frequently      Frequently     Occasionally
3                Rarely       Always          Always           Always
4                Rarely Occasionally      Frequently       Frequently
5          Occasionally Occasionally          Rarely           Always
6
          job_category
1              Faculty
2             Post-Doc
3 Other research staff
4              Faculty
5              Faculty
6 Other research staff
```

Remove rows that contain any empty entries.

```r
nrow(challenges_and_job)
```

```
[1] 332
```

```r
challenges_and_job <- exclude_empty_rows(challenges_and_job, strict = TRUE) # from scripts/u
nrow(challenges_and_job)
```

```
[1] 233
```

For visual clarity in our plots, let’s combine postdocs and other staff researchers.

```r
challenges_and_job <- challenges_and_job %>%
  mutate(
    job_category = recode(
      job_category,
      "Post-Doc" = "Postdocs and\nStaff Researchers",
      "Other research staff" = "Postdocs and\nStaff Researchers"
    )
  )

head(challenges_and_job)
```

```
  Coding time Documentation time Managing issues Attracting users  Recognition
1      Always             Always          Always           Always       Always
2  Frequently       Occasionally    Occasionally     Occasionally Occasionally
3  Frequently             Always    Occasionally           Always Occasionally
4      Always             Always      Frequently     Occasionally   Frequently
5      Always             Always          Rarely     Occasionally   Frequently
7  Frequently         Frequently      Frequently       Frequently   Frequently
      Hiring     Security Finding peers Finding mentors Education time
1     Always       Always        Always          Always         Always
2     Rarely   Frequently  Occasionally      Frequently     Frequently
3 Frequently   Frequently  Occasionally    Occasionally         Rarely
4     Always Occasionally        Rarely          Rarely     Frequently
5      Never        Never         Never           Never         Always
7     Always        Never         Never           Never     Frequently
  Educational resources        Legal Finding funding Securing funding
1                Always       Always          Always           Always
2            Frequently   Frequently      Frequently     Occasionally
3                Rarely       Always          Always           Always
4                Rarely Occasionally      Frequently       Frequently
5          Occasionally Occasionally          Rarely           Always
7                 Never       Always          Always           Always
                     job_category
1                         Faculty
2 Postdocs and\nStaff Researchers
3 Postdocs and\nStaff Researchers
4                         Faculty
5                         Faculty
7                         Faculty
```

Let's reshape the data from wide to long format for easier counting and plotting.

```r
long_data <- challenges_and_job %>%
  pivot_longer(
    cols = -last_col(),
    names_to = "challenge",
    values_to = "challenge_level"
  )

long_data
```

```
# A tibble: 3,262 x 3
   job_category challenge          challenge_level
   <chr>        <chr>              <chr>
 1 Faculty      Coding time        Always
 2 Faculty      Documentation time Always
 3 Faculty      Managing issues    Always
 4 Faculty      Attracting users   Always
 5 Faculty      Recognition        Always
 6 Faculty      Hiring             Always
 7 Faculty      Security           Always
 8 Faculty      Finding peers      Always
 9 Faculty      Finding mentors    Always
10 Faculty      Education time     Always
# i 3,252 more rows
```

Since it's overwhelming to look at the distribution of challenge levels for all groups, let's just look at the proportion of that group who said "frequently" or "always".

```r
# Calculate proportion of TRUEs by taking the mean of a logical vector,
# created by %in%.
to_plot <- long_data %>%
  group_by(job_category, challenge) %>%
  summarize(proportion = mean(challenge_level %in% c("Frequently", "Always"))) %>%
  ungroup()
```

```
`summarise()` has grouped output by 'job_category'. You can override using the
`.groups` argument.
```

```r
to_plot
```

```
# A tibble: 70 x 3
   job_category challenge              proportion
   <chr>        <chr>                       <dbl>
 1 Faculty      Attracting users            0.356
 2 Faculty      Coding time                 0.712
 3 Faculty      Documentation time          0.763
 4 Faculty      Education time              0.492
 5 Faculty      Educational resources       0.186
 6 Faculty      Finding funding             0.627
 7 Faculty      Finding mentors             0.220
 8 Faculty      Finding peers               0.169
 9 Faculty      Hiring                      0.475
10 Faculty      Legal                       0.169
# i 60 more rows
```

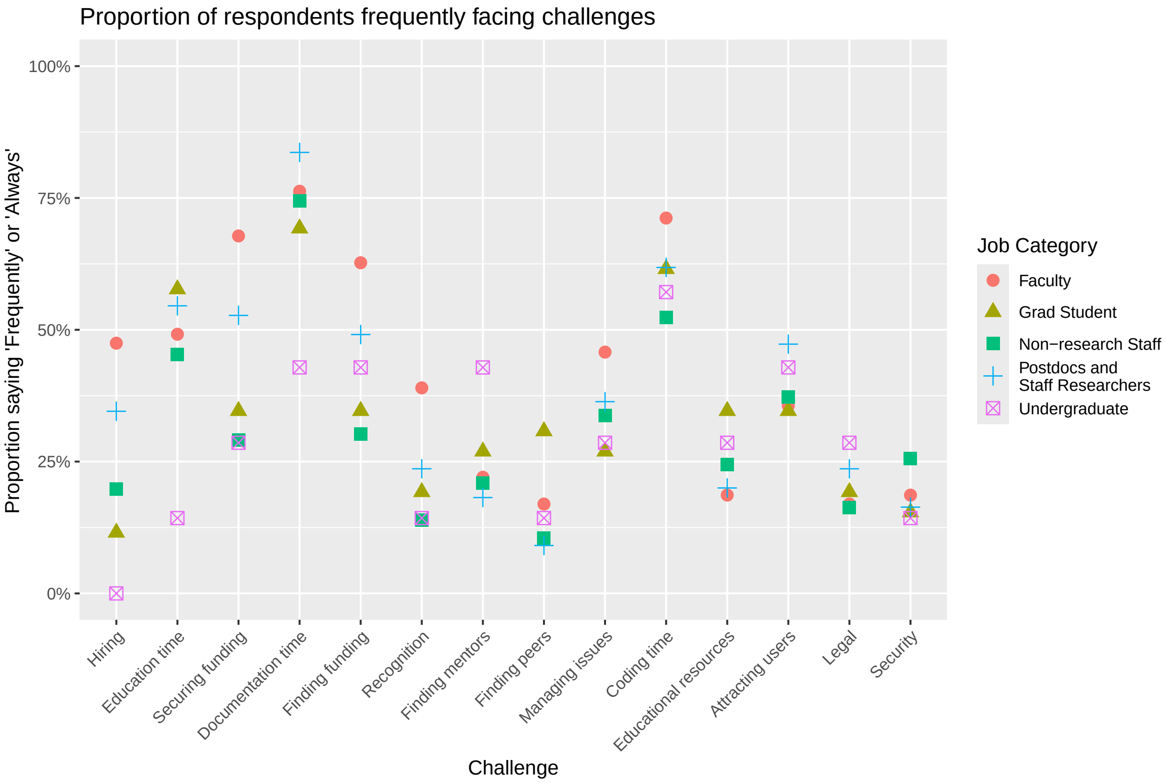
Calculate the standard deviation for each challenge and reorder the factor levels by stdev in our plot. (It looks nicer.)

```
stdev_df <- to_plot %>%
  group_by(challenge) %>%
  summarise(
    st_dev = sd(proportion, na.rm = TRUE)
  ) %>%
  ungroup()

# Order by stdev
stdev_df <- stdev_df %>%
    arrange(desc(st_dev))

# Reorder factor levels
to_plot$challenge <- factor(to_plot$challenge, levels = stdev_df$challenge)
```

```
detailed_challenges_plot <- ggplot(to_plot, aes(x = challenge, y = proportion, group = job_ca
  geom_point(size = 3) +
  scale_y_continuous(labels = scales::percent, limits = c(0, 1)) +
  labs(
    x = "Challenge",
    y = "Proportion saying 'Frequently' or 'Always'",
    color = "Job Category",
    shape = "Job Category",
    title = "Proportion of respondents frequently facing challenges"
  ) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
detailed_challenges_plot
```



Save, if you wish.

```
#save_plot("detailed_challenges_by_job.tiff", 12, 10, p=detailed_challenges_plot)
```

That's a nice plot, but it's probably too information-dense for a presentation, or even a paper. Let's just look at the top 3 challenges for each group.

```
top3 <- to_plot %>%
  group_by(job_category) %>%
  slice_max(order_by = proportion, n = 3)
```

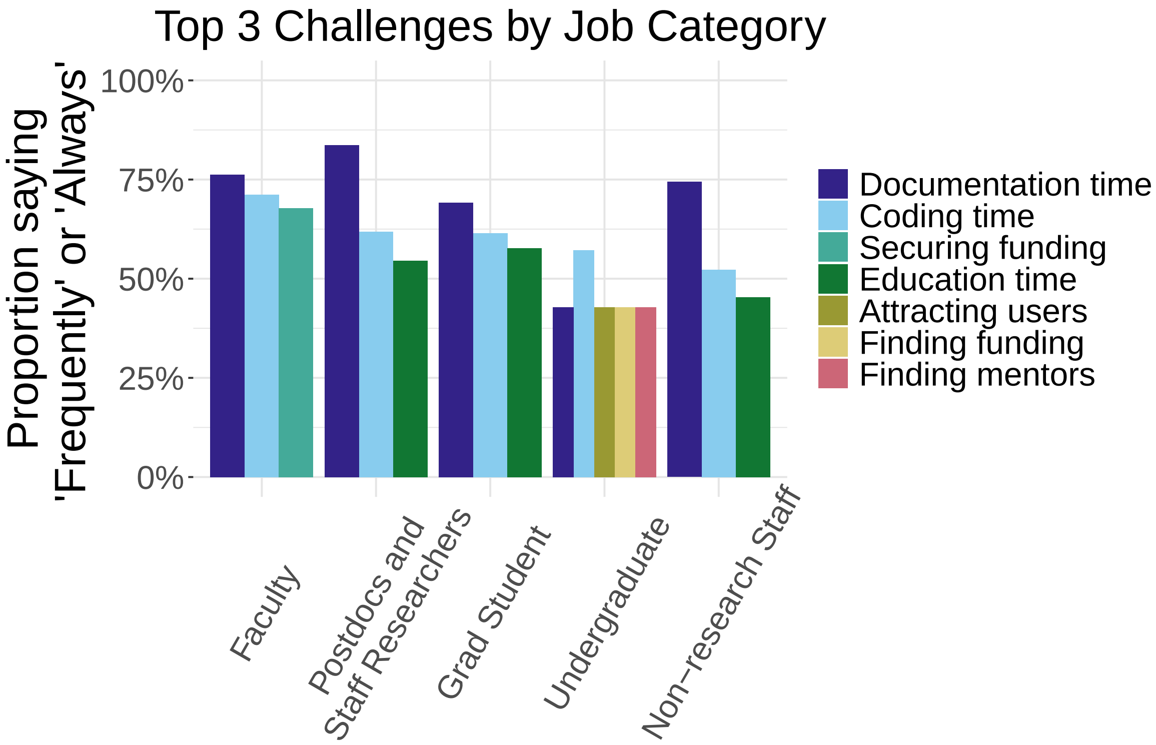
```
# Filter to include only challenges present in the top3 dataframe
filtered_plot <- to_plot %>%
  semi_join(top3, by = c("job_category", "challenge"))
```

```r
# Reorder fill factor levels so legend items are in order of appearance
desired_levels <- top3 %>%
  pull(challenge) %>%
  unique()

filtered_plot <- filtered_plot %>%
  mutate(
    challenge = factor(challenge, levels = desired_levels)
  )
```

```r
# Reorder x-axis factor levels to match academic advancement
job_level_order <- c(
  "Faculty",
  "Postdocs and\nStaff Researchers",
  "Grad Student",
  "Undergraduate",
  "Non-research Staff"
)
filtered_plot$job_category <- factor(
  filtered_plot$job_category,
  levels = job_level_order
  )
```

```r
job_challenge_plot <- ggplot(
  filtered_plot,
  aes(
    x = job_category,
    y = proportion,
    fill = challenge
  )
) +
  geom_col(position = position_dodge()) +
  scale_y_continuous(labels = scales::percent, limits = c(0, 1)) +
  scale_fill_manual(values = COLORS) +
  labs(
    x = "Job Category",
    y = "Proportion saying\n'Frequently' or 'Always'",
    fill = "Challenge",
    title = "Top 3 Challenges by Job Category"
  ) +
  theme(
    axis.title.x = element_blank(),
    axis.title.y = element_text(size = 24),
    axis.text.x = element_text(angle = 60, vjust = 0.6, size = 18),
    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 = 0.5, size = 24),
    plot.margin = unit(c(0.3, 0.3, 0.3, 0.3), "cm")
  )
job_challenge_plot
```



Save, if you wish.

```
#save_plot("top3_challenges_by_job.tiff", 12, 10, p=job_challenge_plot)
```