Code hosting platforms

Overview

This analysis is of Q8, “Where have you shared the code and/or hardware designs for your open-source projects?”

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")))
```

Define functions

get_counts_for_platform_type: Given a broad category of platform, e.g. “vc hosting service”, return a df with cols platform, count, and prop. By default, prop is the proportion of total survey respondents who selected that option, but actually it just counts the rows of whatever data frame you drop in for total_df, and divides by that.

```r
get_counts_and_props_for_platform_type <- function(
  pf_type,
  long_df = platforms_long_labeled,
  total_df = platforms
) {
  pfcounts <- long_df %>%
    filter(platform_type == pf_type) %>%
    group_by(platform, platform_type) %>%
    summarise(count = n(), .groups = "drop") %>%
    select(-platform_type)

  pfcounts <- pfcounts %>% arrange(desc(count))

  pfcounts$platform <- factor(
    pfcounts$platform,
    levels = pfcounts$platform
  )

  pfcounts$prop <- pfcounts$count / nrow(total_df)

  return(pfcounts)
}
```

Load data

```
platforms_raw <- load_qualtrics_data("clean_data/hosting_services_Q8.tsv")
other_quant <- load_qualtrics_data("clean_data/other_quant.tsv")
qual_raw <- load_qualtrics_data("qual_responses.tsv")
```

Wrangle data

Bind the columns we're interested in.

```
platforms <- cbind(platforms_raw, other_quant$campus, other_quant$field_of_study)
# Rename cols
names(platforms)[ncol(platforms)-1] <- "campus"
names(platforms)[ncol(platforms)] <- "field_of_study"

head(platforms)
```

```
  Bitbucket Codeberg GitHub Gitea GitLab Launchpad SourceForge Other Zenodo
1         0        0      1     0      1         0           0     0      1
```

```
2         0        0      1     0      0         0           0     0      1
3         0        0      1     0      0         0           0     1      1
4         0        0      1     0      0         0           0     0      1
5         0        0      1     0      0         0           0     0      1
6         0        0      0     0      0         0           0     0      0
  Dryad Figshare OSF Mendeley Data Vivli Dataverse Custom Website Thingiverse
1     0        1   1             0     0         0              1           0
2     1        0   0             0     0         0              1           0
3     0        0   0             0     0         0              0           0
4     1        1   1             0     0         0              1           0
5     1        1   0             0     0         0              0           0
6     0        0   0             0     0         0              0           0
  Article Supplement           campus field_of_study
1                  1 UC Santa Barbara    Math and CS
2                  0 UC Santa Barbara  Life sciences
3                  0 UC Santa Barbara     Humanities
4                  1 UC Santa Barbara    Math and CS
5                  0 UC Santa Barbara  Life sciences
6                  0 UC Santa Barbara    Math and CS
```

```
nrow(platforms)
```

```
[1] 332
```

Discard rows from people who didn't answer the Q about platforms.

```
keep <- which(rowSums(platforms_raw) != 0)
platforms <- platforms[keep,]
nrow(platforms)
```

```
[1] 233
```

Create a long data frame and label rows with category of platform (platform_type). The fact that the row exists means someone selected that combination of variables.

```
platforms_long <- platforms %>%
  pivot_longer(
    cols = -c(campus, field_of_study),
    names_to = "platform",
    values_to = "flag"
  ) %>%
  filter(flag == 1) %>%
  select(-flag)

platforms_long_labeled <- platforms_long %>%
  mutate(
    platform_type = case_when(
      platform %in%
        c(
          "GitHub",
          "GitLab",
          "Bitbucket",
          "Codeberg",
          "Gitea",
          "Launchpad",
          "SourceForge"
        ) ~
        "vc hosting service",
      platform %in%
        c(
          "Zenodo",
          "Figshare",
          "Dryad",
          "Dataverse",
          "Mendeley Data",
          "OSF",
          "Vivli"
        ) ~
        "data repository",
      platform %in% c(
        "Custom Website"
        ) ~ "custom website",
      platform %in% c(
        "Article Supplement"
        ) ~ "article supplement",
      TRUE ~ "other" # TRUE ~ is like "else", basically
    )
  )

platforms_long_labeled
```

```
# A tibble: 582 x 4
```

```
   campus           field_of_study platform           platform_type
   <chr>            <chr>          <chr>              <chr>
 1 UC Santa Barbara Math and CS    GitHub             vc hosting service
 2 UC Santa Barbara Math and CS    GitLab             vc hosting service
 3 UC Santa Barbara Math and CS    Zenodo             data repository
 4 UC Santa Barbara Math and CS    Figshare           data repository
 5 UC Santa Barbara Math and CS    OSF                data repository
 6 UC Santa Barbara Math and CS    Custom Website     custom website
 7 UC Santa Barbara Math and CS    Article Supplement article supplement
 8 UC Santa Barbara Life sciences  GitHub             vc hosting service
 9 UC Santa Barbara Life sciences  Zenodo             data repository
10 UC Santa Barbara Life sciences  Dryad              data repository
# i 572 more rows
```

Qualitative responses

```
qual <- qual_raw$hosting_services_10_TEXT
qual_clean <- qual[nzchar(qual)]
qual_clean
```

```
 [1] "PyPi"
 [2] "CRAN"
 [3] "stackexchange.com,webwork.maa.org"
 [4] "R"
 [5] "packages.debian.org"
 [6] "Forgejo - FOSS Fork of gitea also git.lsit.ucsb.edu"
 [7] "email diffs, bugzilla bug reporting"
 [8] "github.berkeley.edu"
 [9] "NIH"
[10] "google drive for my college"
[11] "Sofitware Heritage, and local Github Enterprise Server"
[12] "Software Heritage"
[13] "Printables"
[14] "R-Forge"
[15] "gnu.org"
[16] "NIH Managed Data Repository"
[17] "nemar.org"
[18] "Higher Ed Community called SAKAI"
[19] "CRAN, PyPI"
[20] "sourcehut.org"
```

```
[21] "ARXIV"
[22] "Mailing list (x264), Direct to maintainer (Linux kernel)"
[23] "sourcehut"
[24] "Wolfram Mathematica notebook archive"
[25] "Private institutional Git repository"
[26] "CRAN"
```

I'm just going to manually tally the ones that I find interesting right here.

A private or institutional git server: 4
CRAN/R: 4
PyPi: 2
Software Heritage: 2
SourceHut: 2
Printables (similar to thingiverse): 1
R-forge: 1
Wolfram Notebook Archive: 1

Well, definitely some lessons learned for the next time we run this survey. I think the omission of PyPi/CRAN and private git servers was an oversight. We should note this as a "threat to validity".

Exploration

First, I'd like counts for both individual platforms and broader categories of platforms: version control hosting services, data repositories, custom website, article supplement, other.

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

```
                   count
GitHub               222
Custom Website        71
GitLab                69
Article Supplement    35
Zenodo                34
Bitbucket             33
Other                 26
```

```
SourceForge          18
OSF                  14
Thingiverse          12
Dryad                11
Figshare             11
Gitea                 7
Codeberg              6
Dataverse             6
Launchpad             5
Mendeley Data         2
Vivli                 0
```

```r
# Includes all platforms, not just hosting services
ordered_platforms <- rownames(counts)
```

Unsurprisingly, GitHub is very popular. Perhaps surprising, perhaps not, Custom Website is basically tied with GitLab for the second-most popular way to share code.

```r
counts["GitHub","count"]/nrow(platforms)
```

```
[1] 0.9527897
```

```r
counts["Custom Website","count"]/nrow(platforms)
```

```
[1] 0.304721
```

```r
counts["GitLab","count"]/nrow(platforms)
```

```
[1] 0.2961373
```

Plots: vc hosting services

Get counts and proportions (of total respondents) for usage of each version control hosting service.

```r
hosting_platform_data <- get_counts_and_props_for_platform_type("vc hosting service")
```

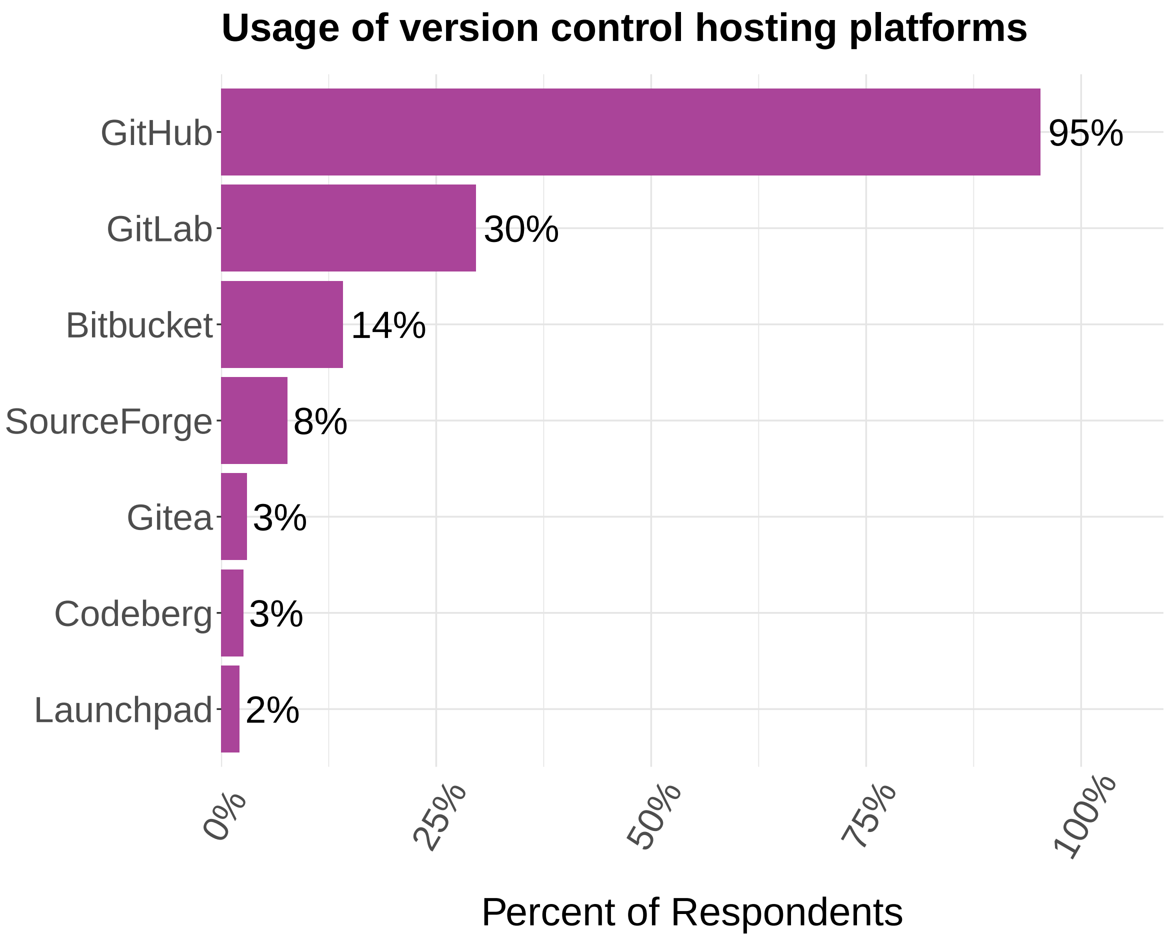
Since we're making a horizontal bar plot, reverse the factor level order.

```r
hosting_platform_data$platform <- factor(
  hosting_platform_data$platform,
  levels = rev(ordered_platforms)
)
```

```r
basic_bar_vc <- basic_bar_chart(
  df = hosting_platform_data,
  x_var = "platform",
  y_var = "prop",
  title = "Usage of version control hosting platforms",
  ylabel = "Percent of Respondents",
  show_axis_title_x = TRUE,
  show_axis_title_y = FALSE,
  show_bar_labels = TRUE,
  label_position = "above",
  label_color = "black",
  percent = TRUE,
  horizontal = TRUE,
  color_index = 9
)

basic_bar_vc <- basic_bar_vc +
   # Expands y-axis by 15% on the upper end
  scale_y_continuous(
  labels = percent,
  expand = expansion(mult = c(0, .15))
  )
```

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

```r
basic_bar_vc
```



Save the plot

```
save_plot("vc_hosting.tiff", 12, 6, p=basic_bar_vc)
```

By campus

Now let's do the same thing, but including campus. Let's only include campuses that have at least 10 responses from experienced contributors.

```
campus_counts <- data.frame(table(platforms$campus))
campus_counts <- campus_counts %>%
  rename(campus = Var1, total = Freq)
at_least_ten <- as.character(
  subset(campus_counts, total > 10)$campus
```

```r
)

ordered_campuses <- campus_counts %>%
  filter(campus %in% at_least_ten) %>%
  arrange(desc(total)) %>%
  pull(campus)

platforms_campus_long_valid <- subset(platforms_long_labeled, campus %in% at_least_ten)

# Reorder factor levels
platforms_campus_long_valid$campus <- factor(
  platforms_campus_long_valid$campus,
  levels = ordered_campuses
)

campus_counts
```

```
             campus total
1          Other UC    19
2       UC Berkeley    26
3          UC Davis    29
4         UC Irvine     2
5    UC Los Angeles    40
6         UC Merced     8
7      UC San Diego     9
8  UC San Francisco     7
9  UC Santa Barbara    61
10    UC Santa Cruz    32
```

```r
nrow(platforms_long_labeled)
```

```
[1] 582
```

```r
nrow(platforms_campus_long_valid)
```

```
[1] 532
```

```r
unique(platforms_campus_long_valid$campus)
```

```
[1] UC Santa Barbara UC Los Angeles   UC Davis         UC Santa Cruz
[5] UC Berkeley      Other UC
6 Levels: UC Santa Barbara UC Los Angeles UC Santa Cruz ... Other UC
```

Select only vc hosting services and get counts.

```
hosting_campus_counts <- platforms_campus_long_valid %>%
  filter(platform_type == "vc hosting service") %>%
  group_by(platform, platform_type, campus) %>%
  summarise(count = n(), .groups = "drop") %>%
  select(-platform_type)

hosting_campus_counts <- hosting_campus_counts %>% arrange(desc(count))

hosting_campus_counts
```

```
# A tibble: 32 x 3
   platform  campus           count
   <chr>     <fct>            <int>
 1 GitHub    UC Santa Barbara    59
 2 GitHub    UC Los Angeles      37
 3 GitHub    UC Santa Cruz       31
 4 GitHub    UC Davis            26
 5 GitHub    UC Berkeley         26
 6 GitLab    UC Santa Barbara    20
 7 GitHub    Other UC            19
 8 GitLab    UC Los Angeles      12
 9 GitLab    UC Santa Cruz       11
10 Bitbucket UC Santa Barbara     8
# i 22 more rows
```

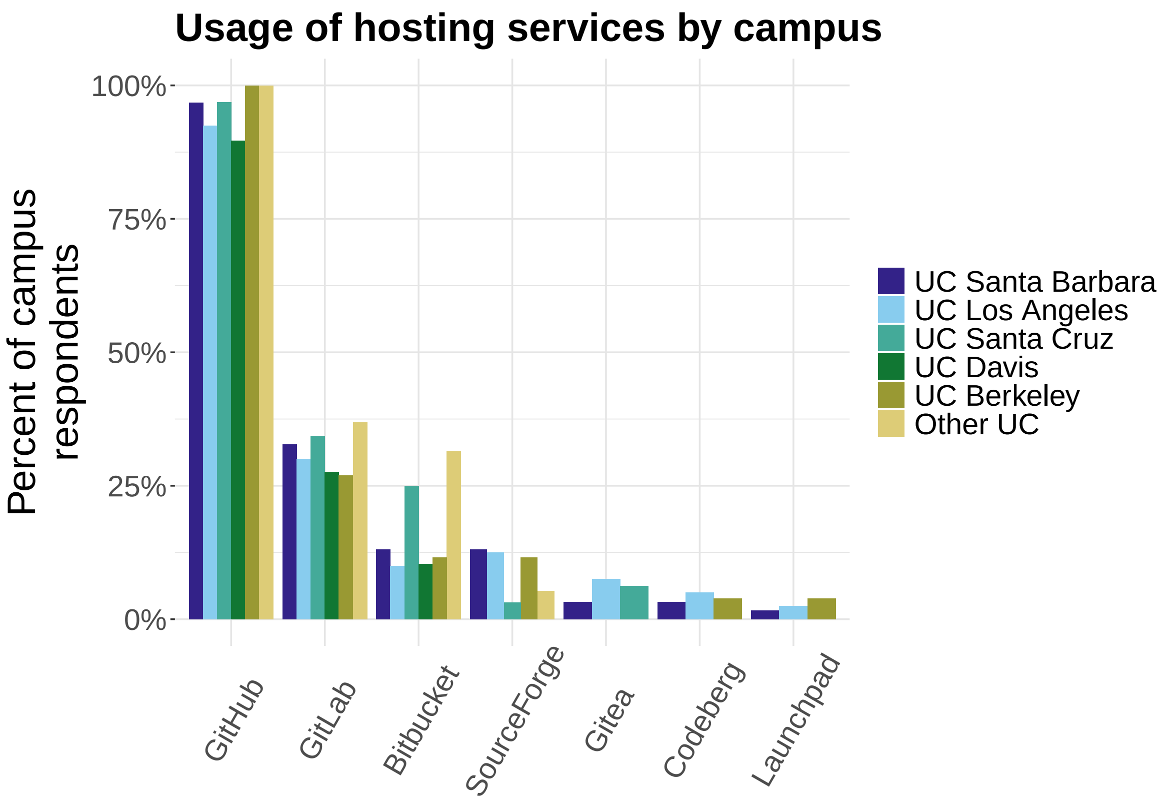
Get proportion of respondents from each campus that selected each platform type.

```
hosting_campus_data <- hosting_campus_counts %>%
  left_join(campus_counts, by = "campus") %>%
  mutate(prop = count / total) %>%
  select(platform, campus, count, prop)
```

Reorder factor levels

```r
hosting_campus_data$platform <- factor(
  hosting_campus_data$platform,
  levels = ordered_platforms
)
```

```r
vc_hosting_campus_plot <- ggplot(
  hosting_campus_data,
  aes(
    x = platform,
    y = prop,
    fill = campus
  )
) +
  geom_bar(stat = "identity", position = "dodge") +
  ggtitle("Usage of hosting services by campus") +
  labs(y = "Percent of campus\nrespondents") +
  scale_fill_manual(values = COLORS) +
  scale_y_continuous(labels = scales::percent) +
  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, size = 24, face = "bold"),
    plot.margin = unit(c(0.3, 0.3, 0.3, 0.3), "cm")
  )

vc_hosting_campus_plot
```



Save the plot

```
save_plot("vc_hosting_campus.tiff", 10, 6, p=vc_hosting_campus_plot)
```

By field of study

Get counts of total (experienced) participants for each field of study.

```
academics <- subset(platforms, field_of_study != "")

field_counts <- data.frame(table(academics$field_of_study))
field_counts <- field_counts %>%
  rename(field_of_study = Var1, total = Freq)

field_counts
```

```
     field_of_study total
```

```
1        Humanities     4
2     Life sciences    34
3       Math and CS    72
4 Physical sciences    27
5   Social sciences    10
```

```
# Total number of academic experienced contributors
sum(field_counts$total)
```

```
[1] 147
```

```
ordered_fields <- field_counts$field_of_study
```

Limit our data to just vc hosting services and academics, and get counts.

```
academics_long <- subset(platforms_long_labeled, field_of_study != "")

hosting_field_counts <- academics_long %>%
  filter(platform_type == "vc hosting service") %>%
  group_by(platform, platform_type, field_of_study) %>%
  summarise(count = n(), .groups = "drop") %>%
  select(-platform_type)

hosting_field_counts <- hosting_field_counts %>% arrange(desc(count))

hosting_field_counts
```

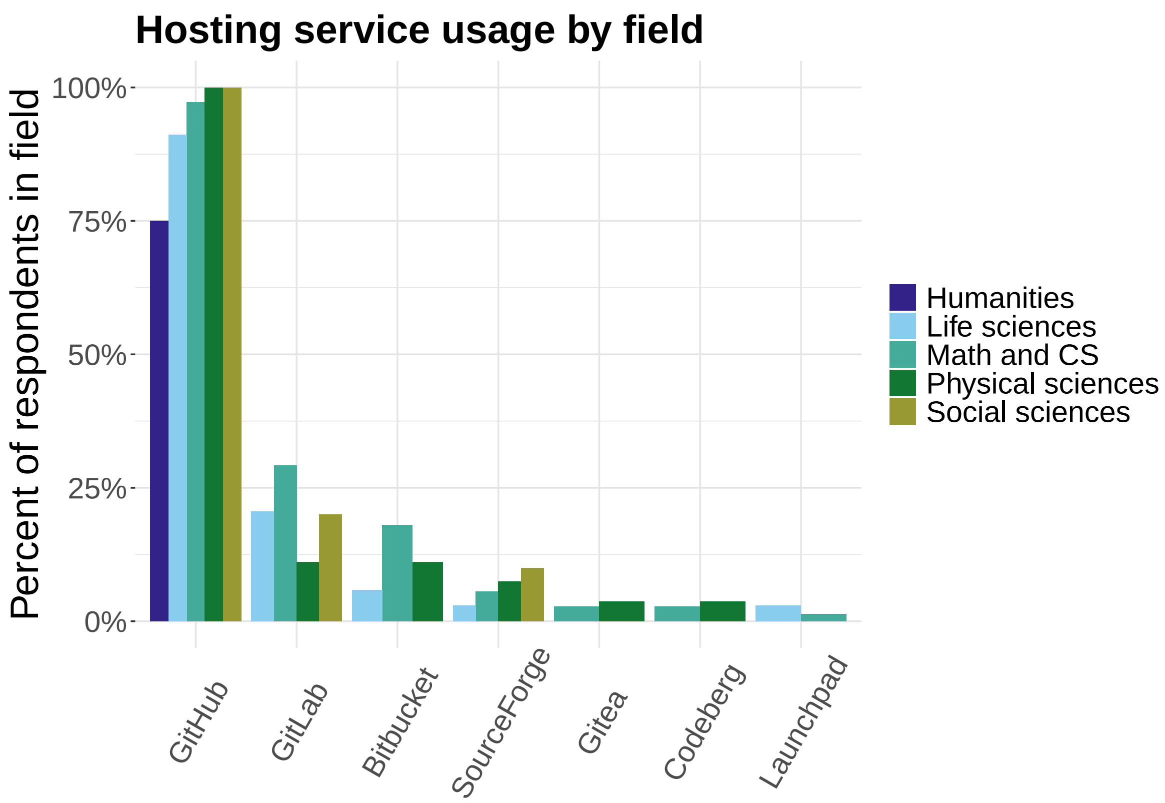
```
# A tibble: 22 x 3
   platform    field_of_study    count
   <chr>       <chr>             <int>
 1 GitHub      Math and CS          70
 2 GitHub      Life sciences        31
 3 GitHub      Physical sciences    27
 4 GitLab      Math and CS          21
 5 Bitbucket   Math and CS          13
 6 GitHub      Social sciences      10
 7 GitLab      Life sciences         7
 8 SourceForge Math and CS           4
 9 Bitbucket   Physical sciences     3
10 GitHub      Humanities            3
# i 12 more rows
```

Get proportions from counts.

```r
hosting_field_data <- hosting_field_counts %>%
  left_join(field_counts, by = "field_of_study") %>%
  mutate(prop = count / total) %>%
  select(platform, field_of_study, count, prop)

# Reorder factor levels
hosting_field_data$platform <- factor(
  hosting_field_data$platform,
  levels = ordered_platforms
)
hosting_field_data$field_of_study <- factor(
  hosting_field_data$field_of_study,
  levels = ordered_fields
)

head(hosting_field_data)
```

```
# A tibble: 6 x 4
  platform  field_of_study    count  prop
  <fct>     <fct>             <int> <dbl>
1 GitHub    Math and CS          70 0.972
2 GitHub    Life sciences        31 0.912
3 GitHub    Physical sciences    27 1
4 GitLab    Math and CS          21 0.292
5 Bitbucket Math and CS          13 0.181
6 GitHub    Social sciences      10 1
```

```r
vc_hosting_field_plot <- ggplot(
  hosting_field_data,
  aes(
    x = platform,
    y = prop,
    fill = field_of_study
  )
) +
  geom_bar(stat = "identity", position = "dodge") +
  ggtitle("Hosting service usage by field") +
  labs(y = "Percent of respondents in field") +
  scale_fill_manual(values = COLORS) +
  scale_y_continuous(labels = scales::percent) +
  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, size = 24, face = "bold"),
    plot.margin = unit(c(0.3, 0.3, 0.3, 0.3), "cm")
  )

vc_hosting_field_plot
```



Meh, not super interesting.

Save the plot

```r
save_plot("vc_hosting_field.tiff", 10, 6, p=vc_hosting_field_plot)
```

Side note: when I saw this, I was a bit confused about the humanities, because it doesn't total up to 100%. The reason is that I'm not showing all options here, just the VC hosting platforms. So of the 4 humanities people, 3 use GitHub, and the 4th said "Article Supplement" only. In other words, I'm just showing what percent of people in this field ticked this option, so the numbers across options don't necessarily add up to 100%, because not all options are shown.

```r
subset(platforms, field_of_study == "Humanities")
```

```
    Bitbucket Codeberg GitHub Gitea GitLab Launchpad SourceForge Other Zenodo
3           0        0      1     0      0         0           0     1      1
38          0        0      1     0      0         0           0     0      0
196         0        0      1     0      0         0           0     0      0
253         0        0      0     0      0         0           0     0      0
    Dryad Figshare OSF Mendeley Data Vivli Dataverse Custom Website Thingiverse
3       0        0   0             0     0         0              0           0
38      0        0   0             0     0         0              0           1
196     0        0   0             0     0         0              0           0
253     0        0   0             0     0         0              0           0
    Article Supplement           campus field_of_study
3                    0 UC Santa Barbara     Humanities
38                   0   UC Los Angeles     Humanities
196                  0 UC Santa Barbara     Humanities
253                  1    UC Santa Cruz     Humanities
```

It might be interesting to show the broad category breakdown by field: vc hosting platform vs. custom website vs. article supplement?

Tables: custom website and article supplement, by field

```r
subset(platforms, field_of_study == "Social sciences")
```

```
    Bitbucket Codeberg GitHub Gitea GitLab Launchpad SourceForge Other Zenodo
28          0        0      1     0      0         0           0     0      0
44          0        0      1     0      1         0           0     0      0
56          0        0      1     0      0         0           0     0      0
```

```
73          0        0      1     0      1         0           1     0      0
78          0        0      1     0      0         0           0     0      0
88          0        0      1     0      0         0           0     0      0
104         0        0      1     0      0         0           0     0      0
112         0        0      1     0      0         0           0     0      0
147         0        0      1     0      0         0           0     0      0
325         0        0      1     0      0         0           0     0      0
    Dryad Figshare OSF Mendeley Data Vivli Dataverse Custom Website Thingiverse
28      0        0   0             0     0         0              0           0
44      0        0   0             0     0         0              0           0
56      0        0   0             0     0         0              0           0
73      0        0   0             0     0         1              1           0
78      0        0   0             0     0         0              0           0
88      0        0   0             0     0         0              0           0
104     0        0   0             0     0         0              0           0
112     0        0   0             0     0         0              1           0
147     0        0   0             0     0         0              0           0
325     0        0   0             0     0         0              0           0
    Article Supplement         campus  field_of_study
28                   1 UC Los Angeles Social sciences
44                   0 UC Los Angeles Social sciences
56                   0 UC Los Angeles Social sciences
73                   1 UC Los Angeles Social sciences
78                   0 UC Los Angeles Social sciences
88                   0 UC Los Angeles Social sciences
104                  0    UC Berkeley Social sciences
112                  0    UC Berkeley Social sciences
147                  0    UC Berkeley Social sciences
325                  0    UC Berkeley Social sciences
```

Meh, I dunno. Maybe just custom website would be interesting.

Select only custom website, and then get counts.

```
website_field_counts <- academics_long %>%
  filter(platform_type == "custom website") %>%
  group_by(platform, platform_type, field_of_study) %>%
  summarise(count = n(), .groups = "drop") %>%
  select(-platform_type)

website_field_counts
```

```
# A tibble: 4 x 3
```

```
  platform       field_of_study    count
  <chr>          <chr>             <int>
1 Custom Website Life sciences        10
2 Custom Website Math and CS          29
3 Custom Website Physical sciences     6
4 Custom Website Social sciences       2
```

Get propotion of total respondents in each field

```
website_field_prop <- website_field_counts %>%
  left_join(field_counts, by = "field_of_study") %>%
  mutate(prop = count / total) %>%
  select(platform, field_of_study, count, prop)

website_field_prop
```

```
# A tibble: 4 x 4
  platform       field_of_study    count  prop
  <chr>          <chr>             <int> <dbl>
1 Custom Website Life sciences        10 0.294
2 Custom Website Math and CS          29 0.403
3 Custom Website Physical sciences     6 0.222
4 Custom Website Social sciences       2 0.2
```

```
# Also note the total proportion of academics who
# have shared code on a custom website
sum(website_field_prop$count)
```

```
[1] 47
```

```
nrow(academics)
```

```
[1] 147
```

```
sum(website_field_prop$count) / nrow(academics)
```

```
[1] 0.3197279
```

That's mildly interesting. On average, 32% of academics report that they've shared their code on a custom website. Math and CS people were almost twice as likely to share their code on a custom website than Physical Science or Social Science. Frequency for Life Sciences is in between.

```
website_field_prop %>%
  left_join(field_counts, by = "field_of_study") %>%
  select(field_of_study, count, total, prop)
```

```
# A tibble: 4 x 4
  field_of_study    count total  prop
  <chr>             <int> <int> <dbl>
1 Life sciences        10    34 0.294
2 Math and CS          29    72 0.403
3 Physical sciences     6    27 0.222
4 Social sciences       2    10 0.2
```

What about article supplement, since we're here and it's easy?

Select only article supplement, and then get counts.

```
article_field_counts <- academics_long %>%
  filter(platform_type == "article supplement") %>%
  group_by(platform, platform_type, field_of_study) %>%
  summarise(count = n(), .groups = "drop") %>%
  select(-platform_type)

article_field_counts
```

```
# A tibble: 5 x 3
  platform           field_of_study    count
  <chr>              <chr>             <int>
1 Article Supplement Humanities            1
2 Article Supplement Life sciences         9
3 Article Supplement Math and CS          13
4 Article Supplement Physical sciences     8
5 Article Supplement Social sciences       2
```

Get proportion of total respondents in each field

```
article_field_prop <- article_field_counts %>%
  left_join(field_counts, by = "field_of_study") %>%
  mutate(prop = count / total) %>%
  select(platform, field_of_study, count, prop)

article_field_prop
```

```
# A tibble: 5 x 4
  platform           field_of_study    count  prop
  <chr>              <chr>             <int> <dbl>
1 Article Supplement Humanities            1 0.25
2 Article Supplement Life sciences         9 0.265
3 Article Supplement Math and CS          13 0.181
4 Article Supplement Physical sciences     8 0.296
5 Article Supplement Social sciences       2 0.2
```

Meh, not super interesting. Math and CS people are less likely to share their code this way than other groups. Not sure if this would be "statistically significant".

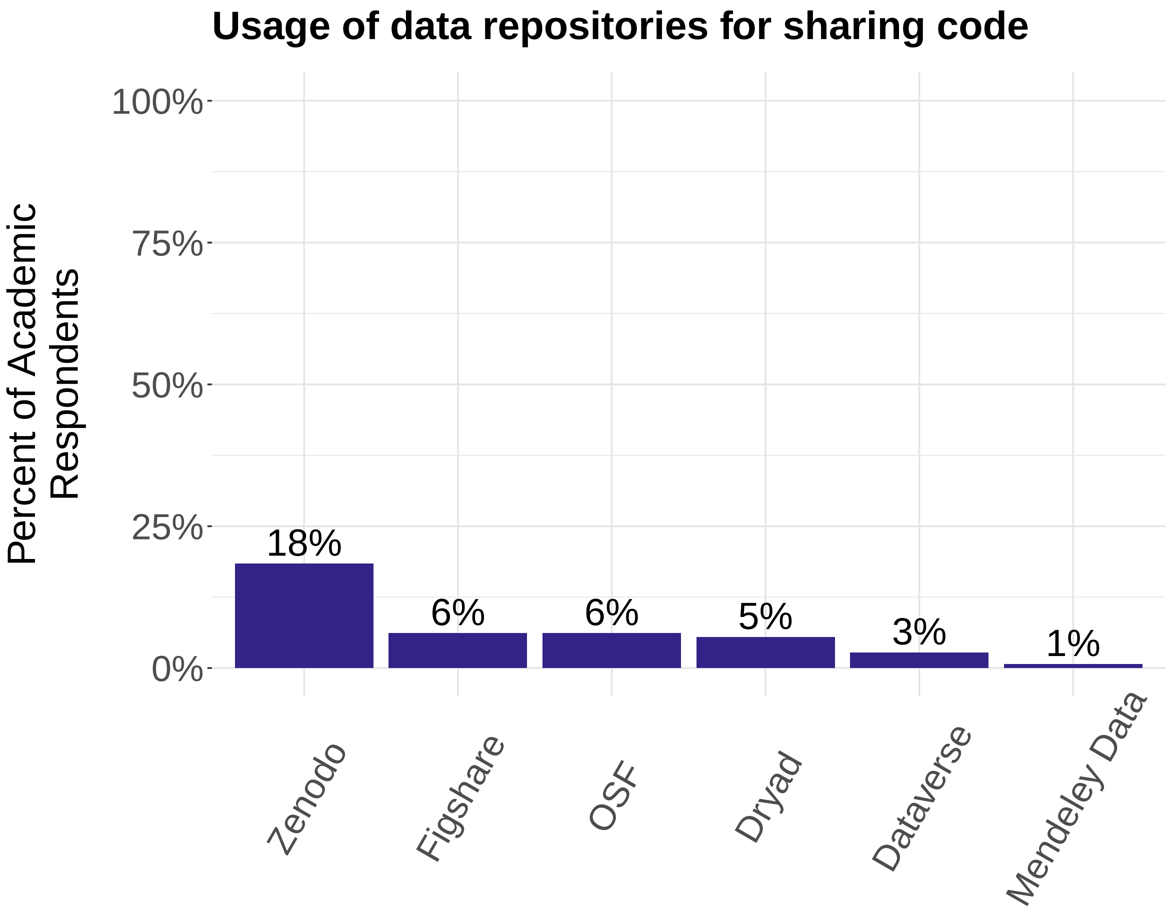
Plots: data repositories

Get counts and proportions (of total respondents) for usage of each data repository. Limit it to academics, since these repositories are intended for scholars.

```
data_repo_platform_data <- get_counts_and_props_for_platform_type(
  "data repository",
  long_df = academics_long,
  total_df = academics
)
```

```
basic_bar_data_repos <- basic_bar_chart(
  df = data_repo_platform_data,
  x_var = "platform",
  y_var = "prop",
  title = "Usage of data repositories for sharing code",
  ylabel = "Percent of Academic\nRespondents",
  show_bar_labels = TRUE,
  label_position = "above",
  label_color = "black",
  percent = TRUE
)

basic_bar_data_repos + scale_y_continuous(
  labels = scales::percent,
  limits = c(0, 1)
)
```

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



Save the plot

```
save_plot("data_repos.tiff", 10, 6, p=basic_bar_data_repos)
```

Quick sanity check

```
# Recall: total # of experienced academics
acad <- nrow(subset(platforms, field_of_study != ""))

# Academics who selected Zenodo
acad_zenodo <- nrow(subset(platforms, field_of_study != "" & Zenodo == 1))

# Total number of experienced nr staff
nrstaff <- nrow(subset(platforms, field_of_study == ""))

# NR Staff who selected Zenodo
nrstaff_zenodo <- nrow(subset(platforms, field_of_study == "" & Zenodo == 1))

acad_zenodo / acad
```

```
[1] 0.1836735
```

```
nrstaff_zenodo / nrstaff
```

```
[1] 0.08139535
```

8% of non-research staff have shared code on Zenodo. I'd bet these are probably library employees.

Let's include Article Supplement and Custom Website

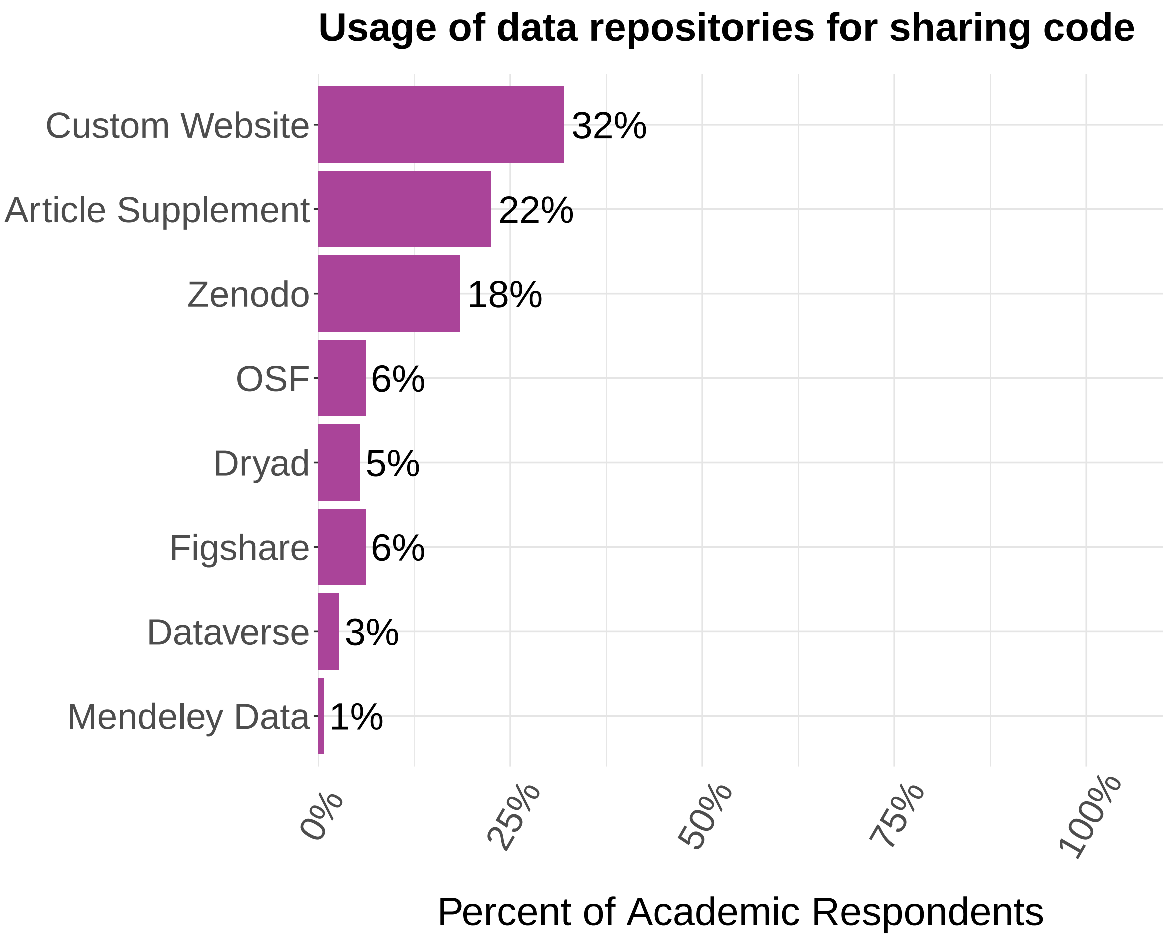
```
article_data <- get_counts_and_props_for_platform_type(
  "article supplement",
  long_df = academics_long,
  total_df = academics
)

website_data <- get_counts_and_props_for_platform_type(
  "custom website",
  long_df = academics_long,
  total_df = academics
)
```

```
expanded_data_repos <- bind_rows(data_repo_platform_data, article_data, website_data)

expanded_data_repos$platform <- factor(
  expanded_data_repos$platform,
  levels = rev(ordered_platforms)
)
```

```
basic_bar_expanded_data_repos <- basic_bar_chart(
  df = expanded_data_repos,
  x_var = "platform",
  y_var = "prop",
  title = "Usage of data repositories for sharing code",
  ylabel = "Percent of Academic Respondents",
  show_axis_title_x = TRUE,
  show_axis_title_y = FALSE,
  show_bar_labels = TRUE,
  label_position = "above",
  label_color = "black",
  percent = TRUE,
  horizontal = TRUE,
  color_index = 9
)

basic_bar_expanded_data_repos <- basic_bar_expanded_data_repos +
  scale_y_continuous(
  labels = scales::percent,
  limits = c(0, 1),
  expand = expansion(mult = c(0, .1))
)
```

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

```
basic_bar_expanded_data_repos
```



Save the plot

```
save_plot("expanded_data_repos.tiff", 12, 6, p=basic_bar_expanded_data_repos)
```

Data repositories by campus

Now let's do the same thing, but including campus. Let's only include campuses that have at least 10 responses from experienced contributors. We can use the platforms_campus_long_valid data frame we constructed earlier. Let's again limit our scope to academics.

Select only data repositories and get counts.

```
data_repo_campus_counts <- platforms_campus_long_valid %>%
  filter(platform_type == "data repository" & "field_of_study" != "") %>%
  group_by(platform, platform_type, campus) %>%
  summarise(count = n(), .groups = "drop") %>%
  select(-platform_type)

data_repo_campus_counts <- data_repo_campus_counts %>% arrange(desc(count))

data_repo_campus_counts
```

```
# A tibble: 25 x 3
   platform  campus           count
   <chr>     <fct>            <int>
 1 Zenodo    UC Santa Barbara    11
 2 Dryad     UC Santa Barbara     7
 3 Figshare  UC Santa Barbara     7
 4 Zenodo    UC Berkeley          6
 5 OSF       UC Berkeley          5
 6 Zenodo    UC Davis             5
 7 Zenodo    Other UC             5
 8 OSF       UC Santa Barbara     4
 9 Zenodo    UC Los Angeles       4
10 Dataverse UC Los Angeles       3
# i 15 more rows
```

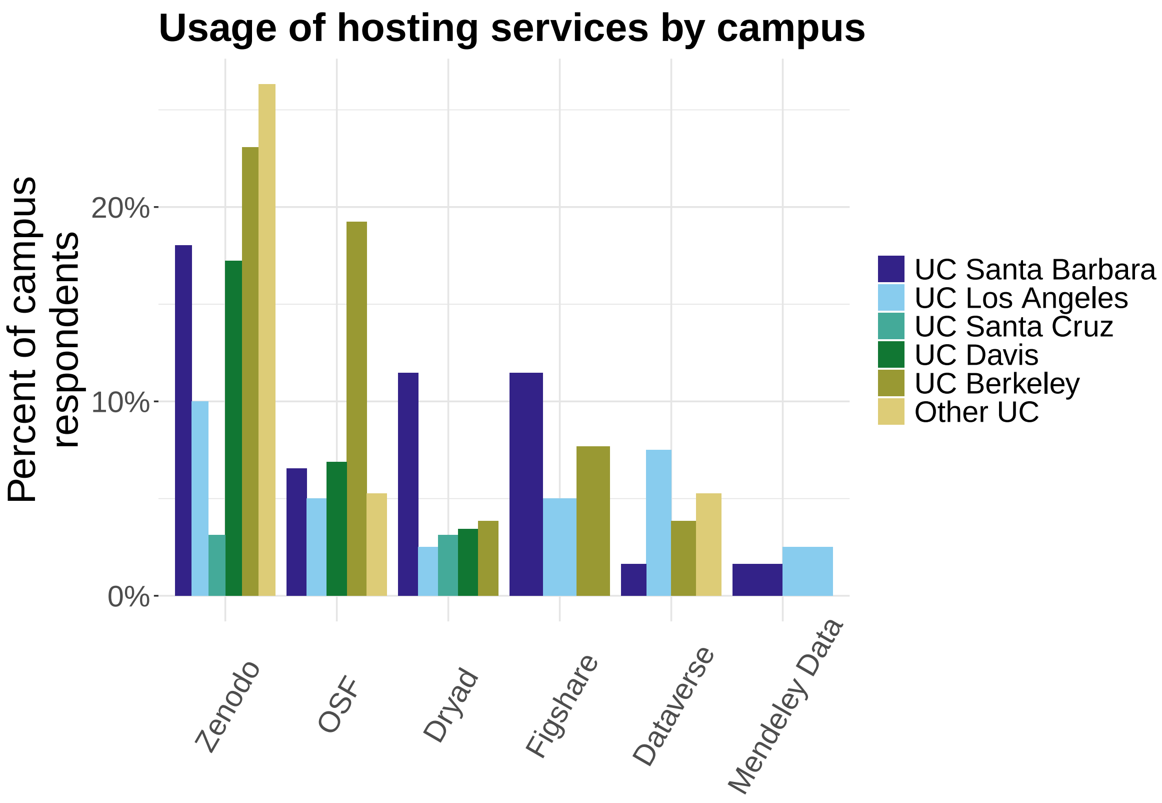
Get proportion of respondents from each campus that selected each platform type.

```
data_repo_campus_data <- data_repo_campus_counts %>%
  left_join(campus_counts, by = "campus") %>%
  mutate(prop = count / total) %>%
  select(platform, campus, count, prop)
```

Reorder factor levels

```
data_repo_campus_data$platform <- factor(
  data_repo_campus_data$platform,
  levels = ordered_platforms
)
```

```r
data_repo_campus_plot <- ggplot(
  data_repo_campus_data,
  aes(
    x = platform,
    y = prop,
    fill = campus
  )
) +
  geom_bar(stat = "identity", position = "dodge") +
  ggtitle("Usage of hosting services by campus") +
  labs(y = "Percent of campus\nrespondents") +
  scale_fill_manual(values = COLORS) +
  scale_y_continuous(labels = scales::percent) +
  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, size = 24, face = "bold"),
    plot.margin = unit(c(0.3, 0.3, 0.3, 0.3), "cm")
  )

data_repo_campus_plot
```



Save the plot

```
save_plot("data_repos_campus.tiff", 10, 6, p=data_repo_campus_plot)
```

That's somewhat interesting. There are some differences between campuses.

TODOs Maybe: Three-way "venn diagram": How many people share code in a data repository only, vc hosting platform only, or both?

```
sessionInfo()
```

```
R version 4.4.2 (2024-10-31)
Platform: aarch64-apple-darwin20
Running under: macOS Sequoia 15.6.1

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     grid      stats     graphics  grDevices datasets  utils
[8] methods   base

other attached packages:
 [1] treemap_2.4-4         tidyr_1.3.1           svglite_2.2.1
 [4] stringr_1.5.1         scales_1.4.0          readr_2.1.5
 [7] pwr_1.3-0             patchwork_1.3.2       ordinal_2023.12-4.1
[10] lme4_1.1-37           Matrix_1.7-1          languageserver_0.3.16
[13] here_1.0.1            gtools_3.9.5          ggforce_0.5.0
[16] fpc_2.2-13            forcats_1.0.0         factoextra_1.0.7
[19] ggplot2_3.5.2         emmeans_1.11.2        dplyr_1.1.4
[22] corrplot_0.95         ComplexHeatmap_2.22.0 cluster_2.1.8.1
[25] BiocManager_1.30.26

loaded via a namespace (and not attached):
 [1] Rdpack_2.6.4        rlang_1.1.6         magrittr_2.0.3
 [4] gridBase_0.4-7      clue_0.3-66         GetoptLong_1.0.5
 [7] matrixStats_1.5.0   compiler_4.4.2      flexmix_2.3-20
[10] systemfonts_1.2.3   png_0.1-8           callr_3.7.6
[13] vctrs_0.6.5         pkgconfig_2.0.3     shape_1.4.6.1
[16] crayon_1.5.3        fastmap_1.2.0       labeling_0.4.3
[19] utf8_1.2.6          promises_1.3.3      rmarkdown_2.29
[22] tzdb_0.5.0          ps_1.9.1            nloptr_2.2.1
[25] purrr_1.1.0         xfun_0.53           modeltools_0.2-24
[28] jsonlite_2.0.0      later_1.4.3         tweenr_2.0.3
[31] parallel_4.4.2      prabclus_2.3-4      R6_2.6.1
[34] stringi_1.8.7       RColorBrewer_1.1-3  boot_1.3-31
[37] diptest_0.77-2      numDeriv_2016.8-1.1 estimability_1.5.1
[40] Rcpp_1.1.0          iterators_1.0.14    knitr_1.50
[43] IRanges_2.40.1      httpuv_1.6.16       igraph_2.1.4
[46] splines_4.4.2       nnet_7.3-19         tidyselect_1.2.1
[49] yaml_2.3.10         doParallel_1.0.17   codetools_0.2-20
[52] processx_3.8.6      lattice_0.22-6      tibble_3.3.0
[55] shiny_1.11.1        withr_3.0.2         evaluate_1.0.4
[58] polyclip_1.10-7     xml2_1.4.0          circlize_0.4.16
```

```
[61] mclust_6.1.1       kernlab_0.9-33      pillar_1.11.0
[64] renv_1.1.5         foreach_1.5.2       stats4_4.4.2
[67] reformulas_0.4.1   generics_0.1.4      rprojroot_2.1.1
[70] S4Vectors_0.44.0   hms_1.1.3           minqa_1.2.8
[73] xtable_1.8-4       class_7.3-22        glue_1.8.0
[76] data.table_1.17.8  robustbase_0.99-4-1 mvtnorm_1.3-3
[79] rbibutils_2.3      colorspace_2.1-1    nlme_3.1-166
[82] cli_3.6.5          textshaping_1.0.1   gtable_0.3.6
[85] DEoptimR_1.1-4     digest_0.6.37       BiocGenerics_0.52.0
[88] ucminf_1.2.2       ggrepel_0.9.6       rjson_0.2.23
[91] farver_2.1.2       htmltools_0.5.8.1   lifecycle_1.0.4
[94] mime_0.13          GlobalOptions_0.1.2 MASS_7.3-61
```