

Final Thoughts

Overview

Plotting results of coding analysis of Q12: “Are there any other challenges you’ve encountered in open source, or types of support that you would find helpful?”

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")))
```

I did this analysis in Taguette. Then I manually copied the number of comments associated with each tag from Taguette to this notebook. Each comment can have multiple tags. I’m creating the dataframe row-wise using tribble because that makes it a little easier to read and change, and I expect these fluid categories might change.

```
results <- tribble(
  ~theme,                                ~category,      ~count,
  "Maintenance",                        "Resources",     9,
  "Funding OS projects",                 "Resources",     8,
  "High turnover",                       "Resources",     3,
  "Skilled personnel",                   "Resources",     9,
  "Lack of time or 'extra' work",         "Resources",     7,
  "Security compliance",                  "Infrastructure", 2,
  "Licensing",                           "Infrastructure", 3,
  "Computing environments",               "Infrastructure", 3,
```

```

"University leadership, norms, and priorities","Culture",      20,
"Code review and replication",          "Culture",          3,
"OS education and careers",             "Culture",          8,
"Value of OS tools",                   "Culture",          7
)

results$theme <- paste0(results$theme, " (", results$count, ")")

```

Now let's plot a treemap!

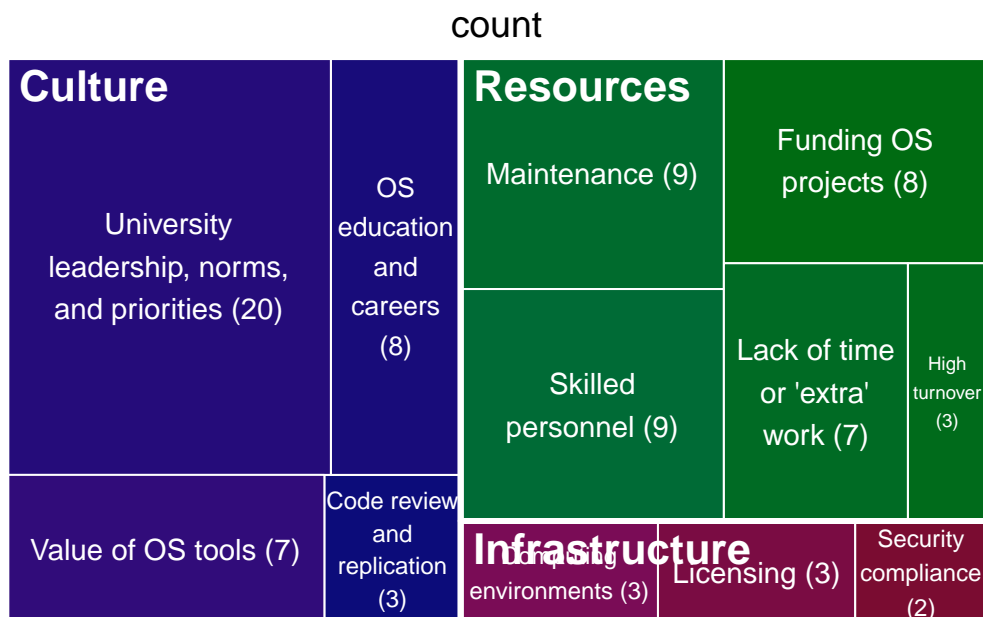
```

my_treemap <- treemap(
  results,
  index = c("category", "theme"), # first level "category", then theme
  vSize = "count", # area of each rectangle proportional to count
  type = "index", # color by index (i.e. category)
  palette = c(COLORS[1], COLORS[8], COLORS[4]), # from utils.R

  # Label styling
  lowerbound.cex.labels = 0, # multiplier between 0 and 1
  # that sets the lowerbound for the data label font sizes: 0 means draw all data labels,
  # and 1 means only draw data labels if they fit (given fontsize.labels).
  fontsize.labels = c(16, 11), # label size for categories, then themes
  fontcolor.labels = c("white", "white"), # white for themes and categories
  align.labels = list(
    c("left", "top"),
    c("center", "center")
  ),

  # aesthetics
  border.col = "white", # white borders for readability
  fontsize.title = 14,
  bg.labels = 0 # transparent label backgrounds
)

```



Oof, that's unfortunate. The labels for the infrastructure category are really cramped. Try as I might, I don't see a good way of allowing them to extend outside the chart. So instead I will just produce a chart where the Infrastructure labels are absent, and I'll add them in later using PowerPoint.

It looks like the `lowerbound.cex.labels` argument lets me control the plot's tolerance for ill-fitting labels. That's helpful. Also, I can hide the "category" labels by setting font size to 0. The way this plot looks is **HIGHLY** dependent on the size of the viewport, which in this case is controlled by the parameters of `png()`.

```
png(
  filename = file.path(Sys.getenv("FIGURE_PATH"), "treemap.png"),
  width = 1800,
  height = 1200,
  res = 300
)

treemap(
  results,
  index = c("category", "theme"),
  vSize = "count",
  type = "index",
  palette = c(COLORS[1], COLORS[8], COLORS[4]), # from utils.R
  title = "Comments from Free Response Box",
```

```

# Label styling
lowerbound.cex.labels = 0.5, # multiplier between 0 and 1 that sets
# the lowerbound for the data label font sizes: 0 means draw all data labels,
# and 1 means only draw data labels if they fit (given fontsize.labels).
fontsize.labels = c(0, 10), # first number is for categories, second is for themes
fontcolor.labels = c("white", "white"), # white for themes and categories
align.labels = list(
  c("center", "top"),
  c("center", "center")
),
inflate.labels = FALSE, # does nothing?
force.print.labels = FALSE, # does nothing?

# aesthetics
border.col = "white", # white borders for readability
fontsize.title = 14,
bg.labels = 0 # transparent label backgrounds
)

dev.off()

```

pdf
2

```
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] rprojroot_2.0.4    fastmap_1.2.0      grid_4.4.2
[43] colorspace_2.1-1   cli_3.6.5          magrittr_2.0.3
[46] withr_3.0.2        promises_1.3.3     tweedie_2.3.5
[49] rmarkdown_2.29     igraph_2.1.4       nnet_7.3-19
[52] modeltools_0.2-24  hms_1.1.3          shiny_1.11.0
[55] evaluate_1.0.3     knitr_1.50         rlang_1.1.6
[58] Rcpp_1.0.14        xtable_1.8-4       gridBase_0.4-7
[61] glue_1.8.0         xml2_1.3.8         renv_1.1.4
[64] jsonlite_2.0.0     R6_2.6.1
```