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

```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")))
```

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.

```r
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", 4,
  "University leadership, norms, and priorities","Culture",       20,
  "Code review and replication",                 "Culture",        4,
  "OS education and careers",                    "Culture",        8,
  "Value of OS tools",                           "Culture",        7
)

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

Plot a treemap

Now let’s plot a treemap! I will use the treemapify package because it integrates nicely with ggplot2.

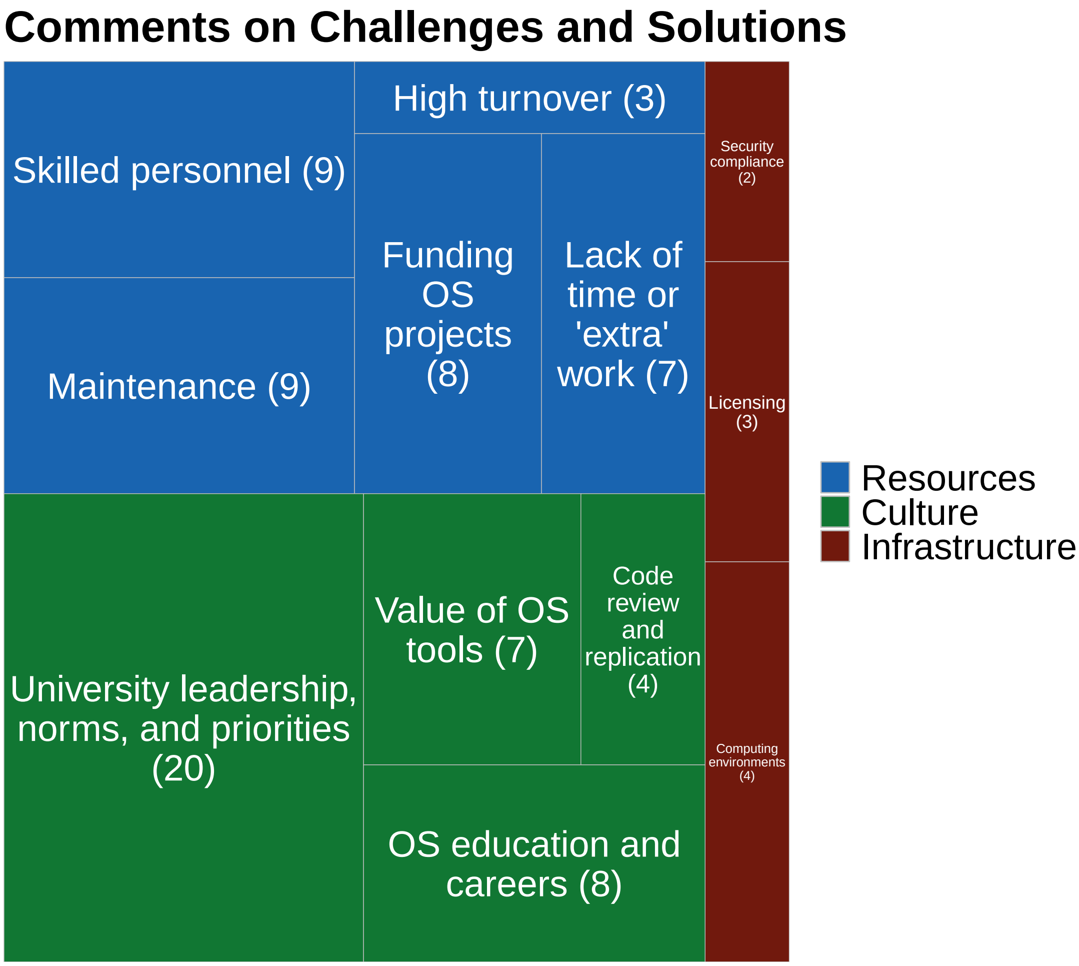
Reorder factor levels so that categories appear in the order my eye notices them on the plot.

```
results$category <- factor(
  results$category,
  levels = c("Resources", "Culture", "Infrastructure")
)
```

```
pal <- c("#1964b0", "#117733", "#71190d")

ggp <- ggplot(
  results,
  aes(
    area = count, # sizes
    fill = category, # colors
    label = theme,
    subgroup = category
  )
) +
  treemapify::geom_treemap() +
  treemapify::geom_treemap_text(
    reflow = TRUE,
    place = "centre",
    grow = FALSE, # cap label font size at 'size'
    size = 20,
    colour = "white"
  ) +
  labs(title = "Comments on Challenges and Solutions") +
  scale_fill_manual(values = pal, guide = guide_legend(title = "Category")) +
  theme(
    legend.text = element_text(size = 20),
    legend.title = element_blank(),
    panel.background = element_blank(),
    plot.margin = margin(1, 1, 1, 1, unit = "cm"),
    plot.title = element_text(size = 24, hjust = 0, face = "bold")
  )

ggp
```



Save the plot

```r
save_plot("treemap.tiff", 12, 9, p=ggp)
```

Let's do that again, but with smaller font sizes according to PLOS submission guidelines.

```r
ggp <- ggplot(
  results,
  aes(
    area = count, # sizes
    fill = category, # colors
    label = theme,
    subgroup = category
  )
) +
  treemapify::geom_treemap() +
  treemapify::geom_treemap_text(
    reflow = TRUE,
    place = "centre",
    grow = FALSE, # cap label font size at 'size'
    size = 10,
    colour = "white"
  ) +
  labs(title = "Comments on Challenges and Solutions") +
  scale_fill_manual(values = pal, guide = guide_legend(title = "Category")) +
  theme(
    legend.text = element_text(size = 10),
    legend.title = element_blank(),
    panel.background = element_blank(),
    plot.margin = margin(1, 1, 1, 1, unit = "cm"),
    plot.title = element_text(size = 10, hjust = 0, face = "bold")
  )
```

```r
ggsave(
  filename = file.path(FIGURE_PATH, "fig10.tif"),
  plot = ggp,
  device = "tiff",
  width = 7.5, height = 4, units = "in",
  dpi = 300,
  compression = "none",
  bg = "white"
)
```

```r
sessionInfo()
```

```
R version 4.4.2 (2024-10-31)
Platform: aarch64-apple-darwin20
Running under: macOS 26.1

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] C.UTF-8/C.UTF-8/C.UTF-8/C/C.UTF-8/C.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] treemapify_2.5.6      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] FSA_0.10.0            fpc_2.2-13            forcats_1.0.0
[19] factoextra_1.0.7      ggplot2_3.5.2         emmeans_1.11.2
[22] dplyr_1.1.4           corrplot_0.95         ComplexHeatmap_2.22.0
[25] cluster_2.1.8.1       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] clue_0.3-66         GetoptLong_1.0.5    matrixStats_1.5.0
 [7] compiler_4.4.2      flexmix_2.3-20      systemfonts_1.2.3
[10] png_0.1-8           callr_3.7.6         vctrs_0.6.5
[13] pkgconfig_2.0.3     shape_1.4.6.1       crayon_1.5.3
[16] fastmap_1.2.0       rmarkdown_2.29      ggfittext_0.10.2
[19] tzdb_0.5.0          ps_1.9.1            nloptr_2.2.1
[22] purrr_1.1.0         xfun_0.53           modeltools_0.2-24
[25] jsonlite_2.0.0      tweenr_2.0.3        parallel_4.4.2
[28] prabclus_2.3-4      R6_2.6.1            stringi_1.8.7
```

```
[31] RColorBrewer_1.1-3  boot_1.3-31         diptest_0.77-2
[34] numDeriv_2016.8-1.1 estimability_1.5.1  Rcpp_1.1.0
[37] iterators_1.0.14    knitr_1.50          IRanges_2.40.1
[40] splines_4.4.2       nnet_7.3-19         tidyselect_1.2.1
[43] yaml_2.3.10         doParallel_1.0.17   codetools_0.2-20
[46] processx_3.8.6      lattice_0.22-6      tibble_3.3.0
[49] withr_3.0.2         evaluate_1.0.4      polyclip_1.10-7
[52] xml2_1.4.0          circlize_0.4.16     mclust_6.1.1
[55] kernlab_0.9-33      pillar_1.11.0       renv_1.1.5
[58] foreach_1.5.2       stats4_4.4.2        reformulas_0.4.1
[61] generics_0.1.4      rprojroot_2.1.1     S4Vectors_0.44.0
[64] hms_1.1.3           minqa_1.2.8         xtable_1.8-4
[67] class_7.3-22        glue_1.8.0          robustbase_0.99-4-1
[70] mvtnorm_1.3-3       rbibutils_2.3       colorspace_2.1-1
[73] nlme_3.1-166        cli_3.6.5           textshaping_1.0.1
[76] gtable_0.3.6        DEoptimR_1.1-4      digest_0.6.37
[79] BiocGenerics_0.52.0 ucminf_1.2.2        ggrepel_0.9.6
[82] rjson_0.2.23        farver_2.1.2        htmltools_0.5.8.1
[85] lifecycle_1.0.4     GlobalOptions_0.1.2 MASS_7.3-61
```