

Figure: the scatterblob

Goal: This is a summary figure, e.g., “here are all the ways and here are the entry points with low technical difficulty, and some for beginner s and the more advanced ways to use it”. There are ways to use github that allows different degrees of collaboration that do not require too many technical skills.

Use even notation across text and figure: homogenizing titles of ways to use github

The data

Data was generated using Google docs, then manually saved as csv file “scatterblob_data_raw.csv” in the data folder.

Data for this figure was generated on a collaborative google spread sheet.

Data, version 2.

Measuring degree of technical difficulty

Cognitive load was measured in a different file.

Cognitive load, version 2.

Reading the data csv file:

```
scatterblob_data <- read_csv(here("data", "scatterblob_data_raw.csv"))

## New names:
## Rows: 20 Columns: 10
## -- Column specification
## ----- Delimiter: "," chr
## (5): Ways to use GitHub, Alternative title, Example, ...9, ...10 dbl (5):
## Cognitive load - real, Cognitive load - sequential, Cognitive load ...
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * ` -> `...9`
## * ` -> `...10`

# View(scatterblob_data)
```

Creating the my_data object from :

```
names(scatterblob_data)

## [1] "Ways to use GitHub"
## [2] "Alternative title"
## [3] "Cognitive load - real"
## [4] "Cognitive load - sequential"
## [5] "Cognitive load - real - non-overlapping"
## [6] "Min degree of collaborativeness X1"
## [7] "Max degree of collaborativeness X2"
## [8] "Example"
## [9] "...9"
## [10] "...10"
```

```
scatterblob_data$x1 <- scatterblob_data$`Min degree of collaborativeness X1`
scatterblob_data$x2 <- scatterblob_data$`Max degree of collaborativeness X2`
scatterblob_data$y_real <- scatterblob_data$`Cognitive load - real - non-overlapping`
scatterblob_data$y_seq <- scatterblob_data$`Cognitive load - sequential`
my_data <- scatterblob_data[1:12,c("Ways to use GitHub", "y_seq", "y_real", "x1", "x2")]
my_data
```

```
## # A tibble: 12 x 5
##   `Ways to use GitHub`      y_seq y_real    x1    x2
##   <chr>                <dbl>  <dbl> <dbl> <dbl>
## 1 Archiving citeable code and data      1   2.2     1     4
## 2 Storing a research compendium          2   2.8     1     4
## 3 Project management                    3    4      2     5
## 4 Project continuity                    4   4.7     2     4
## 5 Open scientific discussions            5   4.35    4     5
## 6 Virtual lab notebook                   6   5.35    1     4
## 7 Educational materials                  7   5.7     1     5
## 8 Academic Websites                      8   6.05    2     5
## 9 Writing a manuscript                   9   6.4     2     4
## 10 GitHub Organizations                 10   6.75    4     5
## 11 Peer-reviews                        11   8.4     1     3
## 12 Asynchronous collaborative coding and writing 12   9.5     4     5
```

Plotting the data

```
outpath <- here("content", "images")
```

1) Using raw cognitive values.

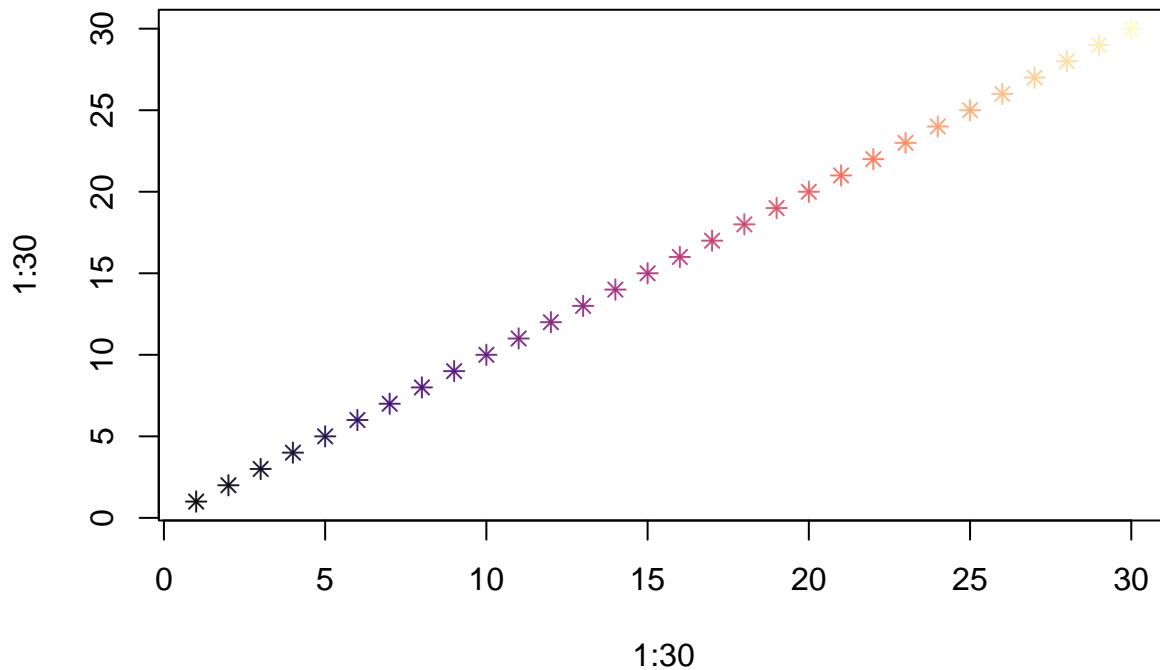
Define colors:

```
some_colors <- viridis::turbo(n = nrow(my_data), alpha = 0.9)

some_colors1 <- viridis::turbo(n = 30, alpha = 0.9)

some_colors2 <- viridis::magma(n = 30, alpha = 0.9)

plot(1:30, 1:30, col = some_colors2, pch = 8)
```



```
my_colors <- some_colors1[c(3:4,
                             8,10,12,
                             20,21,22,23,24,
                             28,30)]
```

```
#####
# start the pdf/png connection and set graphical parameters
#####
#plot_format <- "pdf"
# plot_format <- "svg"
plot_format <- "png"

if (plot_format == "pdf") {
  pdf(file = here(outpath, "scatterblob_0.pdf"),
      width = 8.25,
      height = 6,
      bg = "white")
}
if (plot_format == "svg") {
  svg(filename = here(outpath, "scatterblob_0.svg"),
      width = 8.25,
      height = 6,
      bg = "white")
}
if (plot_format == "png") {
  png(file = here(outpath, "scatterblob_0.png"),
      width = 8.25,
      height = 6,
      units = "in",
      res = 300,
      bg = "white")
}
par(xpd = NA,
```

```

    mai = c(1.02, 1, 0.82, 4),
    bty="n")
#####
# create the plot background
#####
plot(x = c(0.5, 5.5),
     y = c(1.75, 9.75),
     xlab = "",
     ylab = "",
     col = "white",
     # "hide" the tick labels so we can put some words instead of numbers:
     col.axis = "white",
     tck = 0.02)
title(ylab="Technical difficulty", line=4, cex.lab=1)
title(xlab="Degree of collaboration", line= 3.5, cex.lab=1)
#####
# add tick labels
#####
text(x = 1:5,
     y = c(1, rep(1.05, 4)),
     cex = 0.7,
     labels = c("Personal",
                "Lab \nMembers",
                "Other \nLabs",
                "EEB \nCommunity",
                "All \nUsers"))
text(x = c(1, 3, 5),
     y = 0.5,
     labels = c("Low", "Medium", "High"))
# Technical difficulty levels from https://www.londonschool.com/level-scale/
text(x = 0.2,
     y = c(2, 4, 6, 8, 10),
     cex = 0.7,
     adj = 1,
     labels = c("Elementary",
                "Low \nIntermediate",
                "Intermediate",
                "Pre \nAdvanced",
                "Very \nAdvanced"))
#####
# plot scatter blobs
#####
for (i in seq(nrow(my_data))) {
  X0 <- my_data$x1[i]
  X1 <- my_data$x2[i]
  Y <- my_data$y_real[i]
  if (X0 == X1) {
    X0 <- X0 - 0.01
    X1 <- X1 + 0.01
  }
  segments(x0 = X0,
           x1 = X1,
           y0 = Y,

```

```

    col = my_colors[i],
    lwd = 13)
}
#####
# add github uses text
#####
text(x = 5.5,
     y = my_data$y_real,
     labels = my_data$`Ways to use GitHub`,
     adj = 0,
     col = my_colors)
dev.off()

## pdf
## 2

```

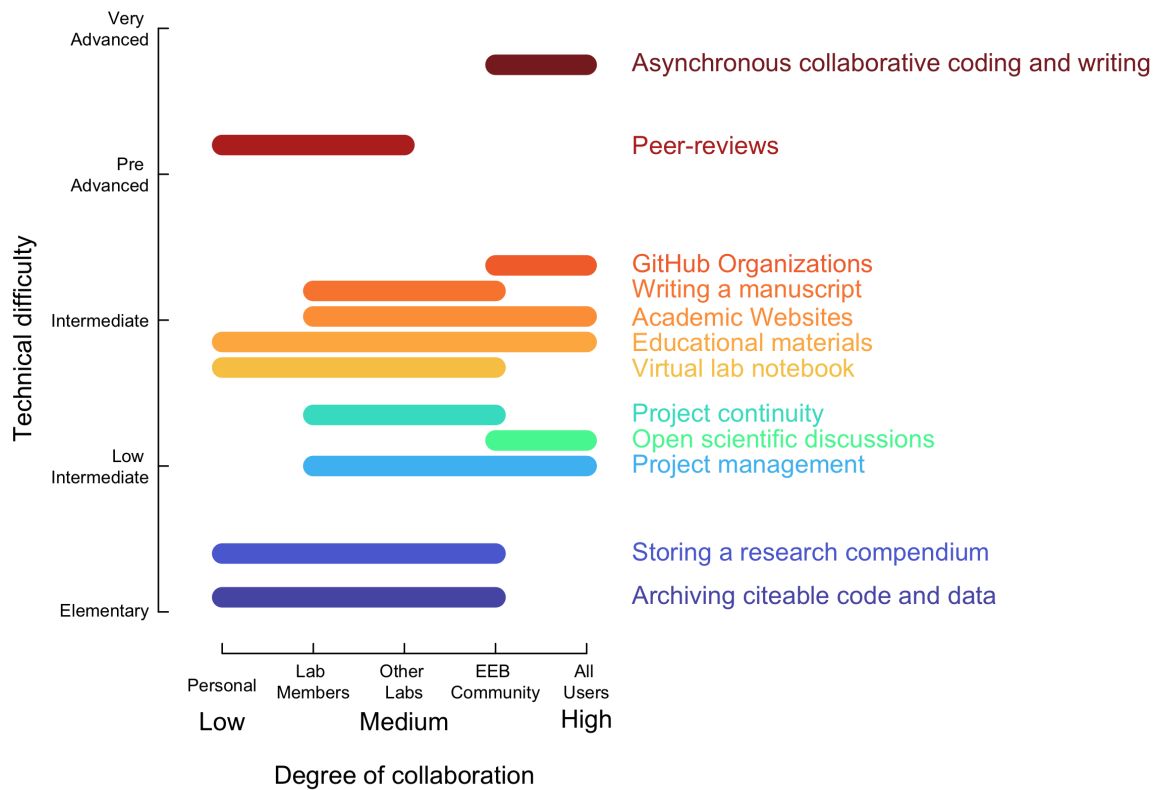


Figure 1: Degree of collaboration VS raw technical difficulty

2) Distributing cognitive load evenly:

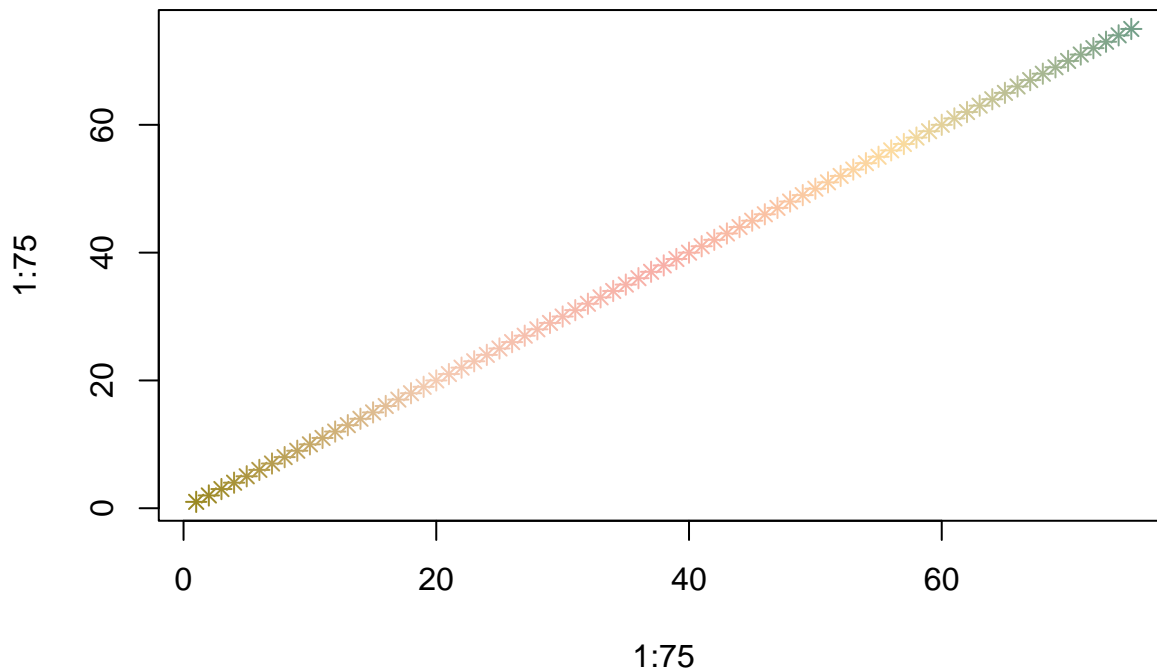
We tried different color palettes from two different R packages: viridis and wesanderson:

```

some_colors3 <- viridis::turbo(n = 100, alpha = 0.9)
some_colors4 <- wesanderson::wes_palette("Zissou1", 75, type = "continuous")
some_colors5 <- wesanderson::wes_palette("Darjeeling1", 75, type = "continuous")

```

```
some_colors6 <- wesanderson::wes_palette("Darjeeling2", 75, type = "continuous")
some_colors7 <- wesanderson::wes_palette("Royal2", 75, type = "continuous")
plot(1:75, 1:75, col = some_colors7, pch = 8)
```



We agreed that viridis-turbo was the nicest palette for the purposes of our figure.

```
my_colors <- some_colors3[c(5,15,
                           26,28,30,
                           68,72,76,80,84,
                           94,100)]

plot_name <- "scatterblob_1-viridis-turbo"
#my_colors <- some_colors4[round(my_data$y_real*10)-20]
#plot_name <- "scatterblob_1-zissou1"
# my_colors <- some_colors5[round(my_data$y_real*10)-20]
# plot_name <- "scatterblob_1-darjeeling1"
# my_colors <- some_colors6[round(my_data$y_real*10)-20]
# plot_name <- "scatterblob_1-darjeeling2"
# my_colors <- some_colors7[round(my_data$y_real*10)-20]
# plot_name <- "scatterblob_1-royal2"
#####
# start the pdf/png/svg connection and set graphical parameters
#####
# plot_format <- "pdf"
plot_format <- "png"
# plot_format <- "svg"

if (plot_format == "pdf") {
  pdf(file = here(outpath, paste0(plot_name, ".pdf")),
      width = 8.25,
      height = 6,
      bg = "white")
}
if (plot_format == "svg") {
```

```

svg(filename = here(outpath, paste0(plot_name, ".svg")),
     width = 8.25,
     height = 6,
     bg = "white")
}
if (plot_format == "png") {
  png(file = here(outpath, paste0(plot_name, ".png")),
      width = 8.25,
      height = 6,
      units = "in",
      res = 300,
      bg = "white")
}
par(xpd = NA,
    mai = c(1.02, 1, 0.82, 4),
    bty="n")
#####
# create the plot background
#####
plot(x = c(0.5, 5.5),
     y = c(0.4, 12.5),
     xlab = "",
     ylab = "",
     col = "white",
     # "hide" the tick labels so we can put some words instead of numbers:
     col.axis = "white",
     tck = 0.02)
title(ylab="Technical Difficulty", line=4, cex.lab=1)
title(xlab="Degree of Collaboration", line= 3.5, cex.lab=1)
#####
# add tick labels
#####
text(x = 1:5,
     y = c(-0.65, rep(-0.7, 4)),
     cex = 0.7,
     labels = c("Personal",
                "Lab \nMembers",
                "Other \nLabs",
                "EEB \nCommunity",
                "All \nUsers"))
text(x = c(1, 3, 5),
     y = -1.7,
     labels = c("Low", "Medium", "High"))
# Technical difficulty levels from https://www.londonschool.com/level-scale/
text(x = 0.2,
     y = c(0, 3, 6, 8.6, 12),
     cex = 0.7,
     adj = 1,
     labels = c("Beginner",
                "Low\nIntermediate",
                "Intermediate",
                "Pre \nAdvanced",
                "Advanced"))

```

```
#####
# plot scatter blobs on sequential cognitive load
#####
for (i in seq(nrow(my_data))) {
  X0 <- my_data$x1[i]
  X1 <- my_data$x2[i]
  Y <- my_data$y_seq[i]
  if (X0 == X1) {
    X0 <- X0 - 0.01
    X1 <- X1 + 0.01
  }
  segments(x0 = X0,
           x1 = X1,
           y0 = Y,
           col = my_colors[i],
           lwd = 23)
}
#####
# add github uses text
#####
text(x = 5.5,
     y = my_data$y_seq,
     labels = my_data$`Ways to use GitHub`,
     adj = 0,
     col = my_colors)
dev.off()

## pdf
## 2
```

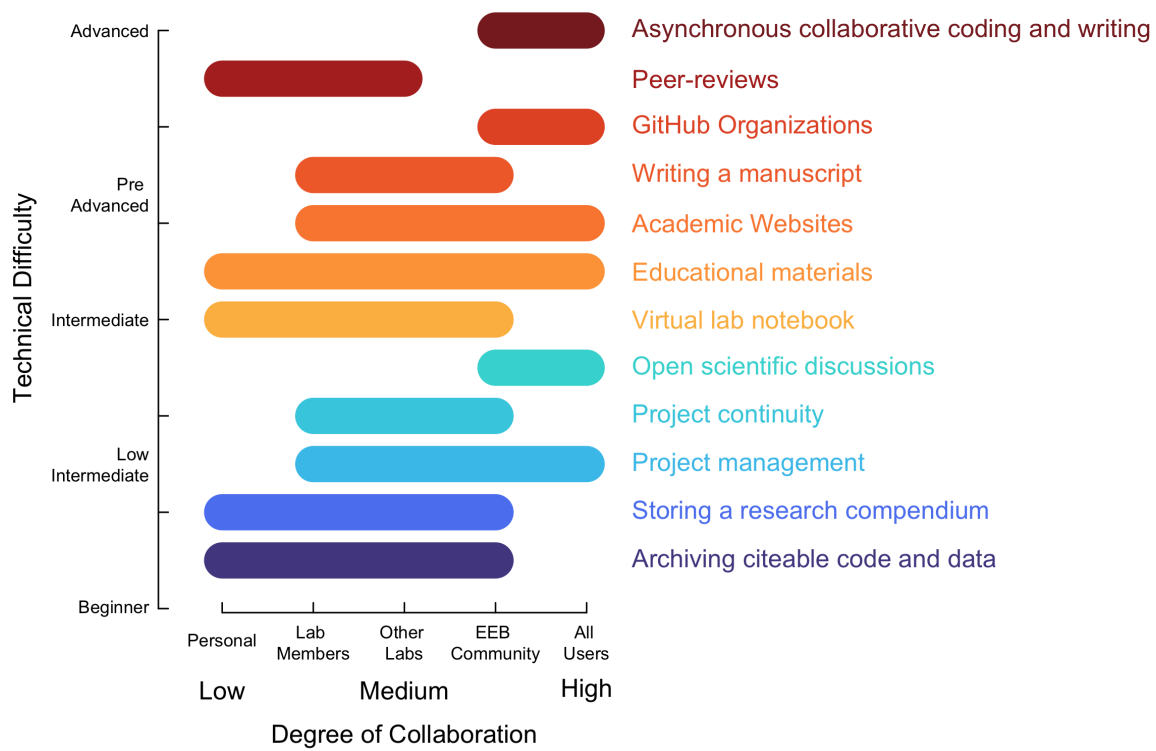



Figure 2: Degree of collaboration VS Even technical difficulty