

The goal: interactivity + reproducibility

Drug research and validation

Workflows benefit greatly from interactive apps, but analysis ultimately needs to be provided in a fully reproducible form

Teaching

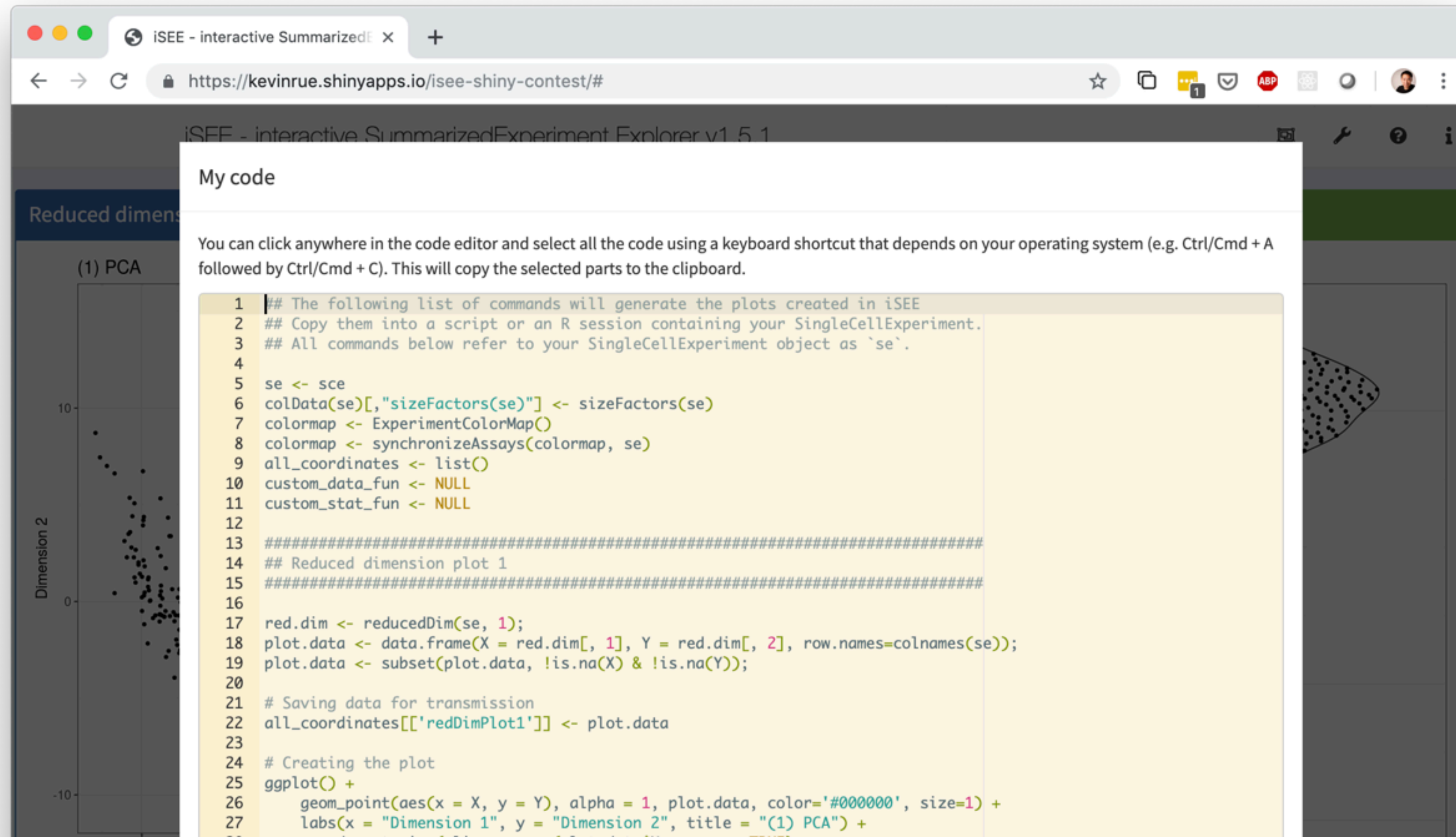
Interactive apps to teach statistical concepts, with corresponding code snippets to teach usage in R

Gadgets/RStudio Add-ins

Use an interactive user interface to build e.g. a ggplot2 plot, regular expression, or SQL query, then insert the corresponding code into the Source editor

Reproducible artifacts

- View an R snippet



The screenshot shows a web browser window with the URL <https://kevinrue.shinyapps.io/isee-shiny-contest/#>. The page title is "iSEE - interactive SummarizedExperiment Explorer v1.5.1". A modal window titled "My code" is open, displaying R code for generating a PCA plot. The code includes comments and R commands for loading data, creating a color map, and plotting the data. The background shows a PCA plot with "Dimension 1" on the x-axis and "Dimension 2" on the y-axis.

My code

You can click anywhere in the code editor and select all the code using a keyboard shortcut that depends on your operating system (e.g. Ctrl/Cmd + A followed by Ctrl/Cmd + C). This will copy the selected parts to the clipboard.

```
1 ## The following list of commands will generate the plots created in iSEE
2 ## Copy them into a script or an R session containing your SingleCellExperiment.
3 ## All commands below refer to your SingleCellExperiment object as `se`.
4
5 se <- sce
6 colData(se)[,"sizeFactors(se)"] <- sizeFactors(se)
7 colormap <- ExperimentColorMap()
8 colormap <- synchronizeAssays(colormap, se)
9 all_coordinates <- list()
10 custom_data_fun <- NULL
11 custom_stat_fun <- NULL
12
13 #####
14 ## Reduced dimension plot 1
15 #####
16
17 red.dim <- reducedDim(se, 1);
18 plot.data <- data.frame(X = red.dim[, 1], Y = red.dim[, 2], row.names=colnames(se));
19 plot.data <- subset(plot.data, !is.na(X) & !is.na(Y));
20
21 # Saving data for transmission
22 all_coordinates[["redDimPlot1"]] <- plot.data
23
24 # Creating the plot
25 ggplot() +
26   geom_point(aes(x = X, y = Y), alpha = 1, plot.data, color='#000000', size=1) +
27   labs(x = "Dimension 1", y = "Dimension 2", title = "(1) PCA") +
```

iSEE