Data Visualization: Assignment 2

Student 1: Michał Kałmucki 151944

Student 2: Dominik Ludwiczak 151948

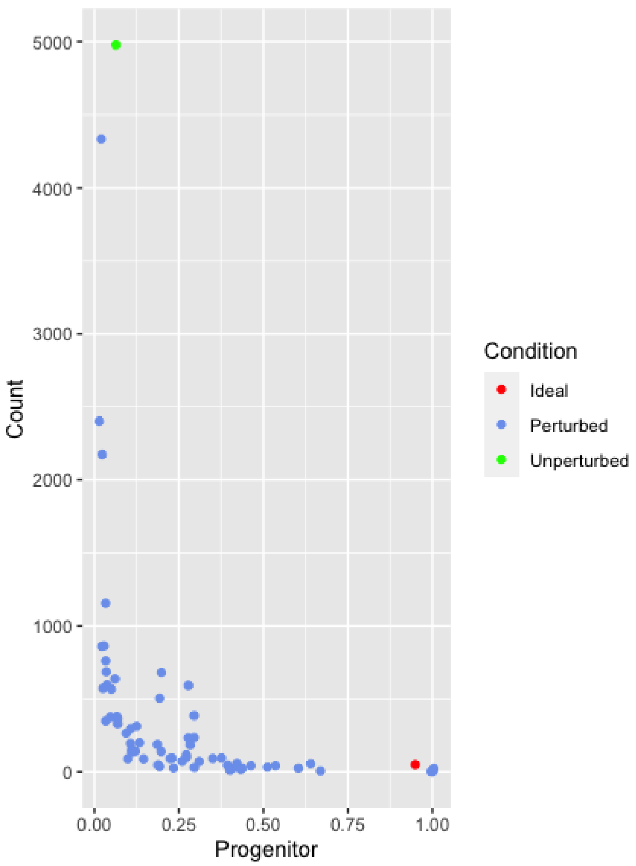
Three preliminary visualizations

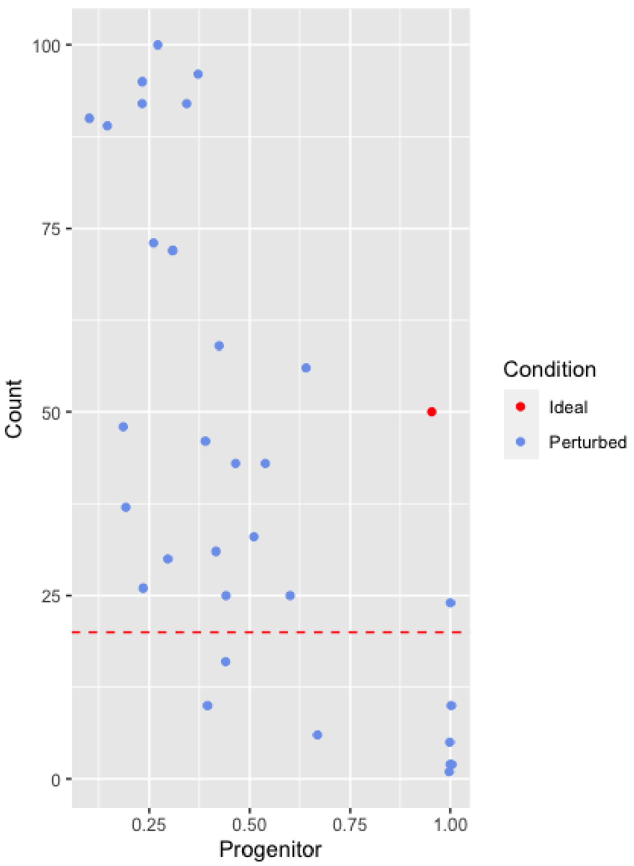
Sketch 1

Implementation 1

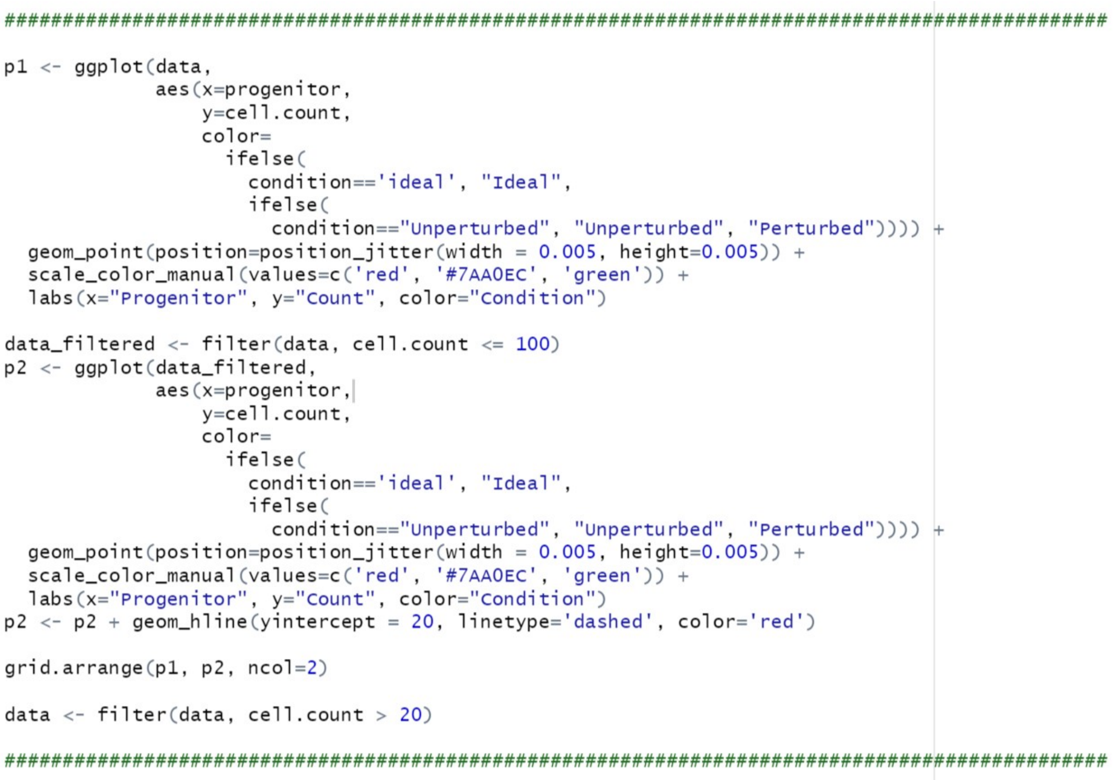
We have decided to remove entries for which count was lower then 20 since we believe they don’t show to reliable of data and can lead to strange outliers when plotted.

Also we introduced in our graphs “Ideal” entry that had desired distribution of (0.95,0,0,0.05,0) to easier compare tested methods.



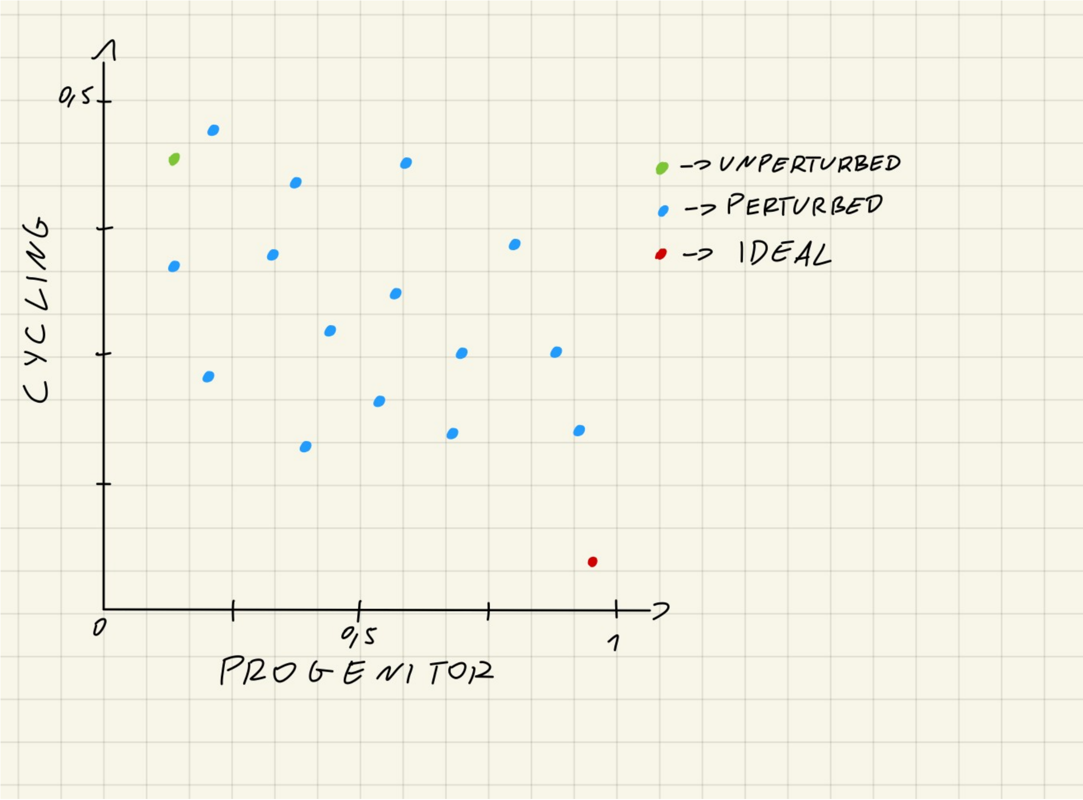


Implementation code 1

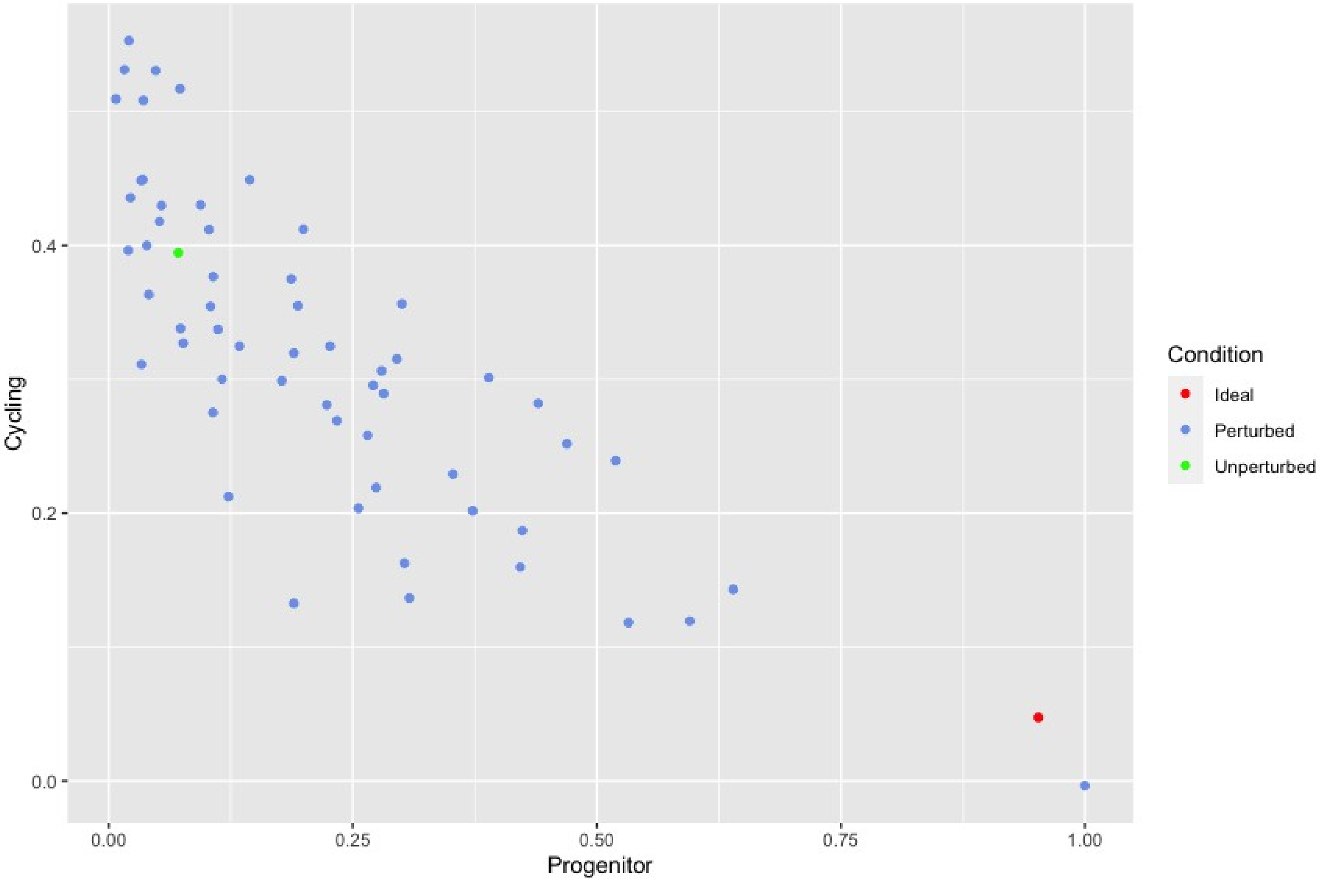


```
###############################################################################################

p1 <- ggplot(data,
             aes(x=progenitor,
                 y=cell.count,
                 color=
                   ifelse(
                     condition=='ideal', "Ideal",
                     ifelse(
                       condition=="Unperturbed", "Unperturbed", "Perturbed")))) +
  geom_point(position=position_jitter(width = 0.005, height=0.005)) +
  scale_color_manual(values=c('red', '#7AA0EC', 'green')) +
  labs(x="Progenitor", y="Count", color="Condition")

data_filtered <- filter(data, cell.count <= 100)
p2 <- ggplot(data_filtered,
             aes(x=progenitor,
                 y=cell.count,
                 color=
                   ifelse(
                     condition=='ideal', "Ideal",
                     ifelse(
                       condition=="Unperturbed", "Unperturbed", "Perturbed")))) +
  geom_point(position=position_jitter(width = 0.005, height=0.005)) +
  scale_color_manual(values=c('red', '#7AA0EC', 'green')) +
  labs(x="Progenitor", y="Count", color="Condition")
p2 <- p2 + geom_hline(yintercept = 20, linetype='dashed', color='red')

grid.arrange(p1, p2, ncol=2)

data <- filter(data, cell.count > 20)

###############################################################################################
```

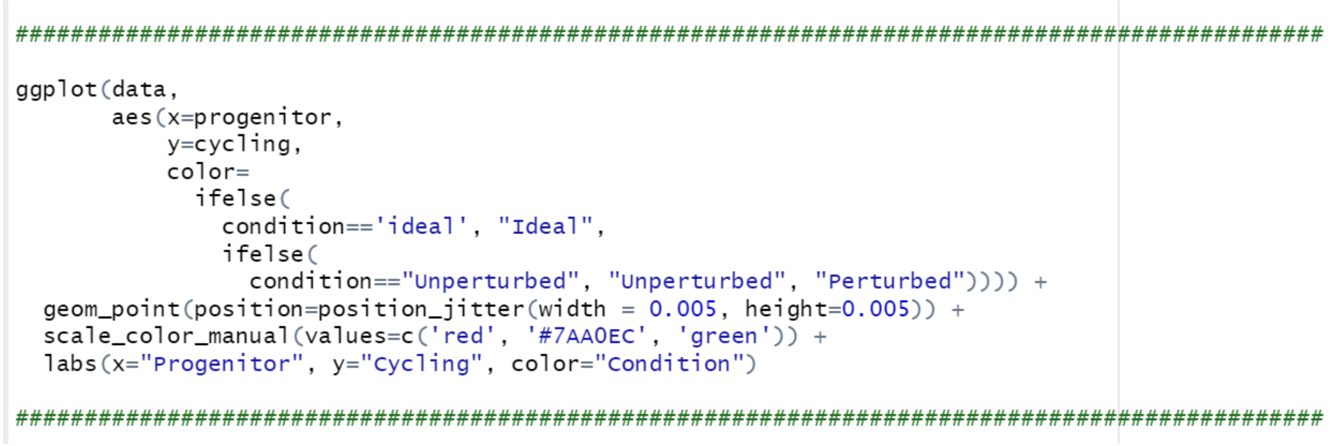
Sketch 2 (optional)



Implementation 2



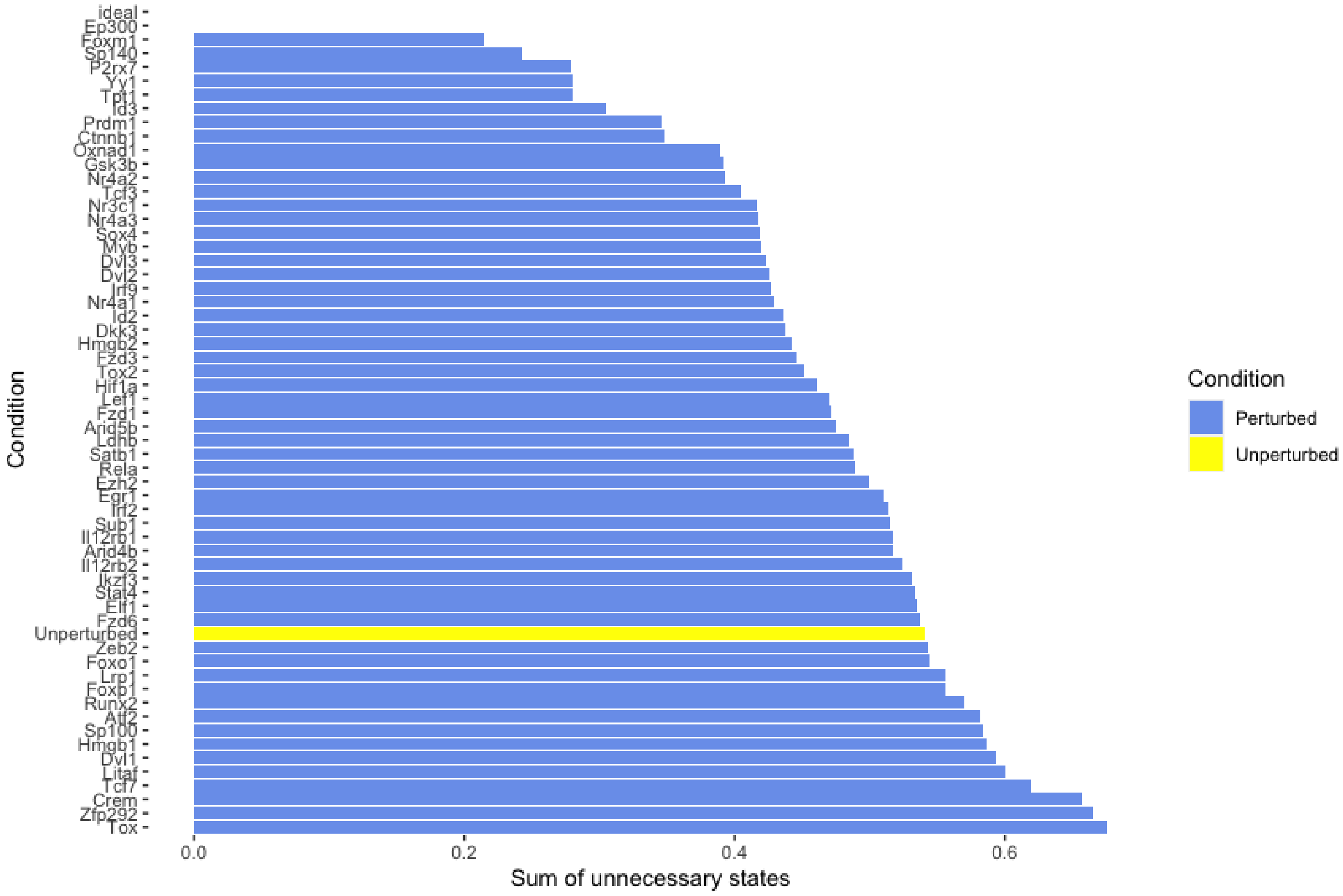
Implementation code 2



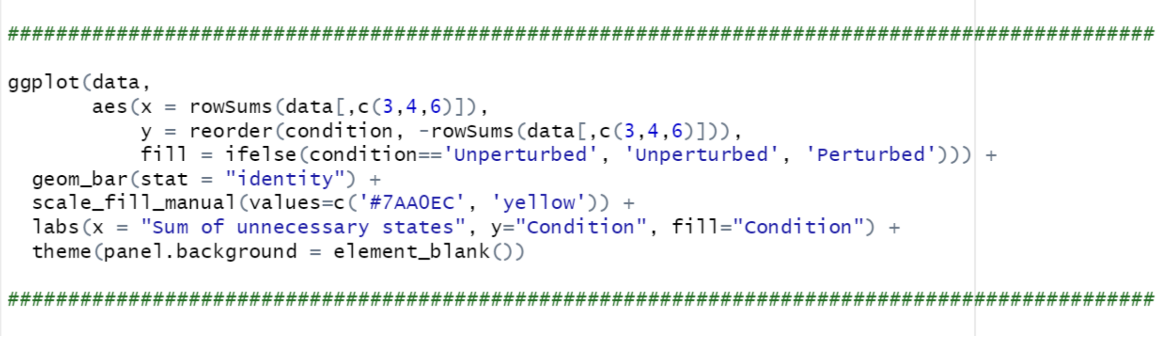
```
##########################################################################################

ggplot(data,
       aes(x=progenitor,
           y=cycling,
           color=
             ifelse(
               condition=='ideal', "Ideal",
               ifelse(
                 condition=="Unperturbed", "Unperturbed", "Perturbed")))) +
  geom_point(position=position_jitter(width = 0.005, height=0.005)) +
  scale_color_manual(values=c('red', '#7AA0EC', 'green')) +
  labs(x="Progenitor", y="Cycling", color="Condition")

##########################################################################################
```

Sketch 3 (optional)

Implementation 3



Implementation code 3

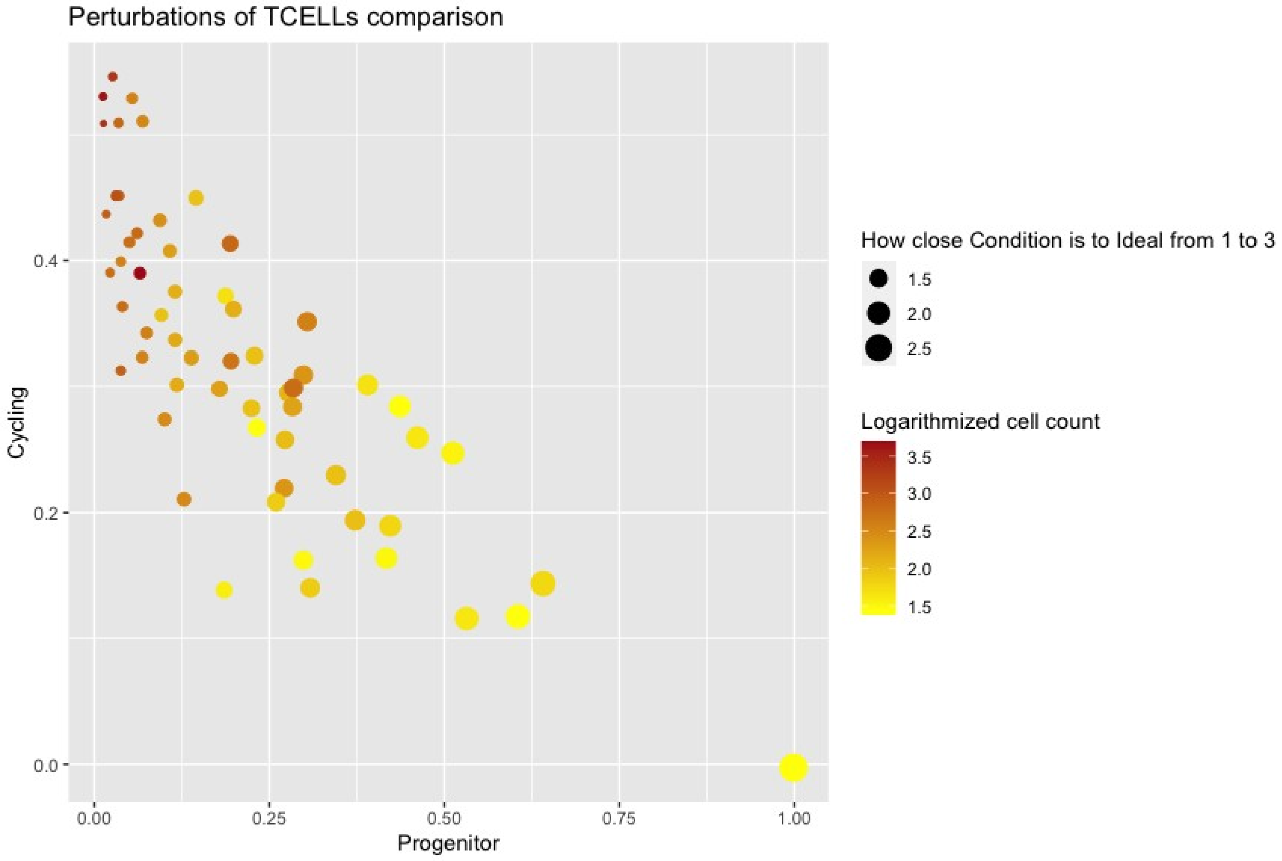


```
################################################################################################

ggplot(data,
       aes(x = rowSums(data[,c(3,4,6)]),
           y = reorder(condition, -rowSums(data[,c(3,4,6)])),
           fill = ifelse(condition=='Unperturbed', 'Unperturbed', 'Perturbed'))) +
  geom_bar(stat = "identity") +
  scale_fill_manual(values=c('#7AA0EC', 'yellow')) +
  labs(x = "Sum of unnecessary states", y="Condition", fill="Condition") +
  theme(panel.background = element_blank())

################################################################################################
```

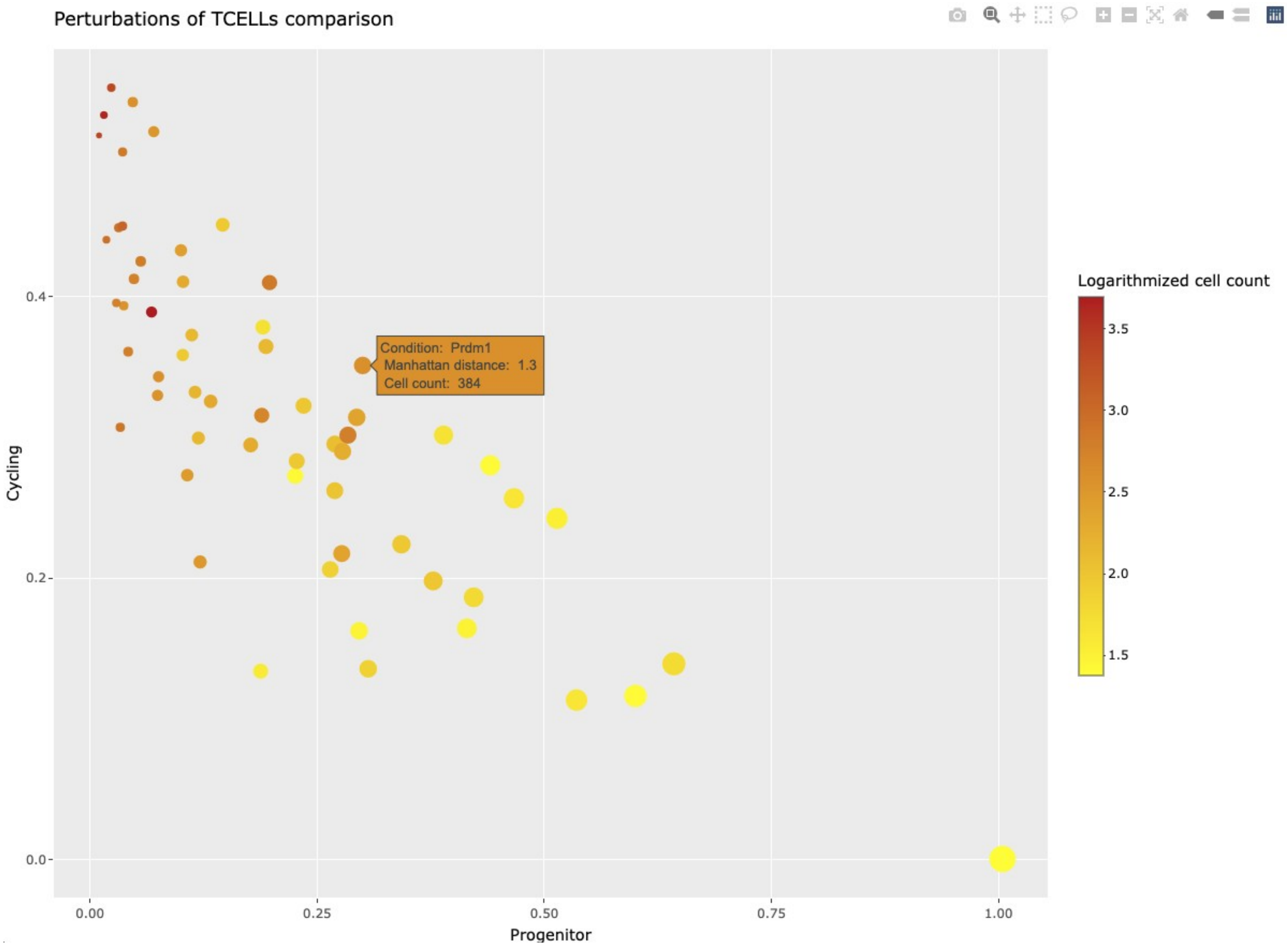
Selected final visualization

Implementation

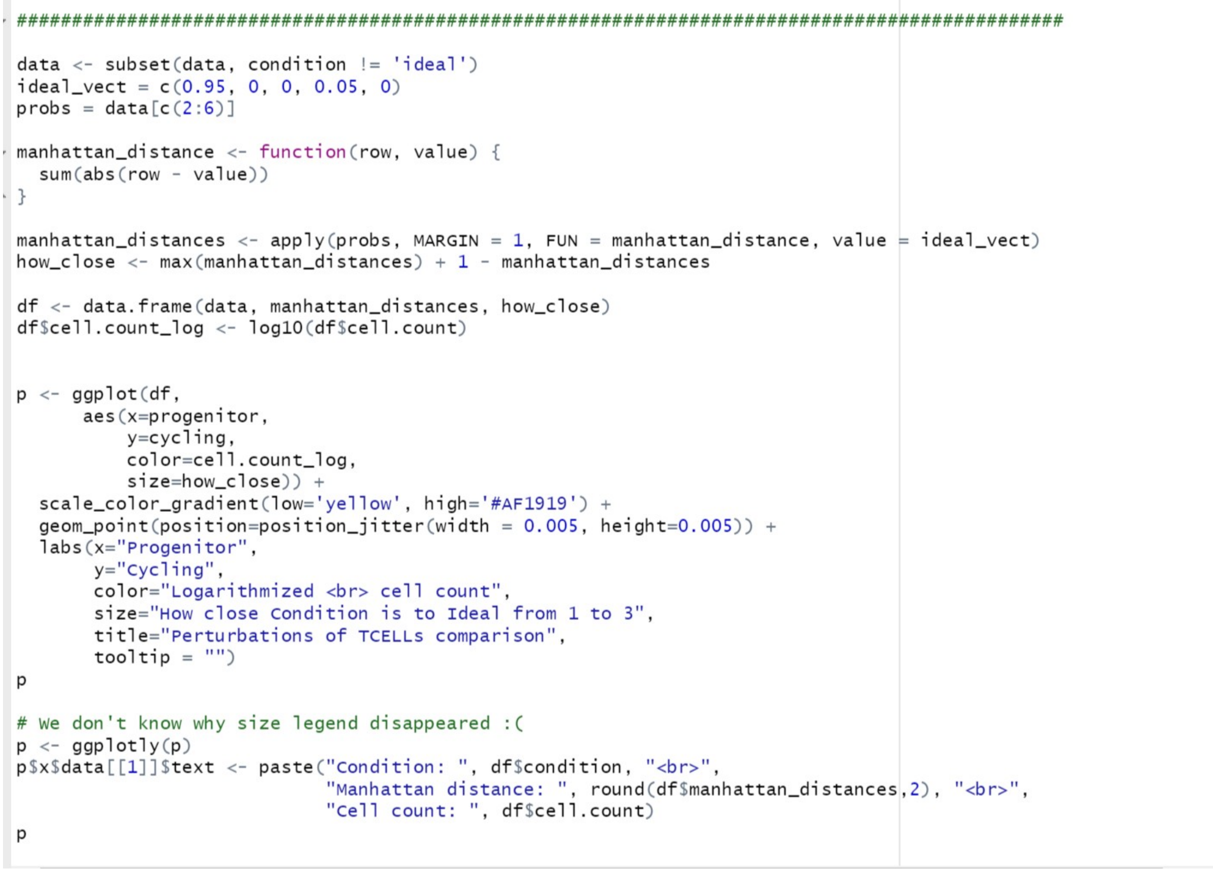
Cell count number is logarithmized because numbers were either very small or very large so to show actual differences and not just outliers we decided to normalize it this way.



Graph is interactive – hovering on each point shows us specific information about which method it regards and what is total variation distance from “ideal” method.



Implementation code



```
###############################################################################################

data <- subset(data, condition != 'ideal')
ideal_vect = c(0.95, 0, 0, 0.05, 0)
probs = data[c(2:6)]

manhattan_distance <- function(row, value) {
  sum(abs(row - value))
}

manhattan_distances <- apply(probs, MARGIN = 1, FUN = manhattan_distance, value = ideal_vect)
how_close <- max(manhattan_distances) + 1 - manhattan_distances

df <- data.frame(data, manhattan_distances, how_close)
df$cell.count_log <- log10(df$cell.count)


p <- ggplot(df,
      aes(x=progenitor,
          y=cycling,
          color=cell.count_log,
          size=how_close)) +
  scale_color_gradient(low='yellow', high='#AF1919') +
  geom_point(position=position_jitter(width = 0.005, height=0.005)) +
  labs(x="Progenitor",
       y="Cycling",
       color="Logarithmized <br> cell count",
       size="How close Condition is to Ideal from 1 to 3",
       title="Perturbations of TCELLs comparison",
       tooltip = "")
p

# We don't know why size legend disappeared :(
p <- ggplotly(p)
p$x$data[[1]]$text <- paste("Condition: ", df$condition, "<br>",
                            "Manhattan distance: ", round(df$manhattan_distances,2), "<br>",
                            "Cell count: ", df$cell.count)
p
```