

COSICC_kinetics

Magdalena Strauss

2025-06-20

This tutorial illustrates how to use COSICC_kinetics with SingleCellExperiments.

```
knitr::opts_chunk$set(  
  warning = FALSE,  
  message = FALSE,  
  error = FALSE  
)  
  
suppressPackageStartupMessages(library(COSICC))
```

First, we simulate a chimera-style data set, where cells have a fluorescent marker tdTomato or not, and which have several lineage trajectories.

```
sim_data_sce <- simulate_sce_kinetics(seed =10,batchCells=5000)  
  
print(sim_data_sce$plots$case$Path1)
```

```
## NULL
```

The function above simulated a knockout and a wild-type chimera data set.

```
sce_knockout_kinetics <- sim_data_sce$case  
sce_WT_kinetics <- sim_data_sce$control
```

Let's have a look at the simulated data sets:

```
colData(sce_knockout_kinetics)
```

```
## DataFrame with 4258 rows and 7 columns  
##           Cell      Batch   Group ExpLibSize      pt  tdTomato  
##           <character> <character> <factor>   <numeric> <numeric> <logical>  
## Cell11      Cell11      Batch1    Path1      66621.0     11      TRUE  
## Cell12      Cell12      Batch1    Path5      72881.1     69      FALSE  
## Cell13      Cell13      Batch1    Path3      59400.4     11      TRUE  
## Cell14      Cell14      Batch1    Path2      92392.0     16      FALSE  
## Cell15      Cell15      Batch1    Path5      43452.9     78      TRUE  
## ...         ...         ...         ...         ...         ...         ...  
## Cell14995   Cell14995   Batch1    Path5      62765.9     80      FALSE  
## Cell14996   Cell14996   Batch1    Path4      56514.5     39      FALSE  
## Cell14998   Cell14998   Batch1    Path5      30729.4     43      FALSE  
## Cell14999   Cell14999   Batch1    Path5      72140.7     98      FALSE  
## Cell15000   Cell15000   Batch1    Path4      57851.1     88      TRUE  
##           sizeFactor  
##           <numeric>  
## Cell11      0.969250  
## Cell12      0.964994
```

```
## Cell13      0.995339
## Cell14      1.456470
## Cell15      0.771821
## ...
## Cell4995    1.004818
## Cell4996    1.053613
## Cell4998    0.448054
## Cell4999    1.358044
## Cell5000    0.859470
```

We need to rename the colData.

```
names(colData(sce_WT_kinetics))[names(colData(sce_WT_kinetics))=="tdTomato"] <- "marked"
names(colData(sce_knockout_kinetics))[names(colData(sce_knockout_kinetics))=="tdTomato"] <- "marked"
```

We also make sure that the cells from the WT and knockout data sets have different names.

```
colnames(sce_WT_kinetics) <- paste0(colnames(sce_WT_kinetics), "_WT")
colnames(sce_knockout_kinetics) <- paste0(colnames(sce_knockout_kinetics), "_knockout")
```

Now the data look as follows:

```
head(colData(sce_WT_kinetics))

## DataFrame with 6 rows and 7 columns
##           Cell      Batch   Group ExpLibSize      pt      marked
##           <character> <character> <factor>   <numeric> <numeric> <logical>
## Cell11_WT      Cell11    Batch1    Path1      66621.0      11      TRUE
## Cell12_WT      Cell12    Batch1    Path5      72881.1      69     FALSE
## Cell13_WT      Cell13    Batch1    Path3      59400.4      11      TRUE
## Cell14_WT      Cell14    Batch1    Path2      92392.0      16     FALSE
## Cell15_WT      Cell15    Batch1    Path5      43452.9      78      TRUE
## Cell16_WT      Cell16    Batch1    Path1      73600.7      61     FALSE
##           sizeFactor
##           <numeric>
## Cell11_WT    0.969250
## Cell12_WT    0.964994
## Cell13_WT    0.995339
## Cell14_WT    1.456470
## Cell15_WT    0.771821
## Cell16_WT    1.351423
```

We can now apply COSICC_kinetics to each of the lineages separately. In our case the column of the colData that refers to the lineage trajectories is Group. Note that the colData needs to include the columns 'pt' for the pseudotime values and 'marked' for whether they have a fluorescent marker (e.g. tdTomato). Also note that the values in the column marked need to be logical (TRUE vs FALSE).

```
lineages <- unique(sce_knockout_kinetics$Group)
COSICC_kinetics_results <- list()
for (j in seq_along(lineages)){
  COSICC_kinetics_results[[j]] <-
    COSICC_kinetics(sce_case=sce_knockout_kinetics[,sce_knockout_kinetics$Group == lineages[j]],
                    sce_control=sce_WT_kinetics[,sce_WT_kinetics$Group == lineages[j]])
}
```

We summarise the results for the separate lineage trajectories as a joint data frame.

```
names(COSICC_kinetics_results) <- lineages
df_COSICC_kinetics_results <- do.call(rbind, lapply(names(COSICC_kinetics_results), function(nm) {
  cbind(path = nm, as.data.frame(COSICC_kinetics_results[[nm]], stringsAsFactors = FALSE))
}))
rownames(df) <- NULL
df_COSICC_kinetics_results
```

```
##   path p_value_case lower_ci_bound_control upper_ci_bound_control
## 1 Path1 2.450799e-23          -1.999946          6.0000801
## 2 Path5 1.402215e-02          -8.000037         -0.9999762
## 3 Path3 1.851820e-25          -2.999960          4.9999739
## 4 Path2 6.431823e-30          -3.999994          3.0000274
## 5 Path4 8.166119e-01          -3.000052          4.0000326
##   lower_ci_bound_case upper_ci_bound_case
## 1          -26.000048          -17.9999741
## 2           -8.000037           -0.9999762
## 3          -27.999956          -19.0000567
## 4          -29.999970          -21.0000269
## 5           -3.000052           4.0000326
##   convidence_interval_overlapping_with_control   sig
## 1                                     FALSE  TRUE
## 2                                     TRUE  FALSE
## 3                                     FALSE  TRUE
## 4                                     FALSE  TRUE
## 5                                     TRUE  FALSE
```

The plot below illustrates the data frame. COSICC_kinetics performs a Wilcoxon rank-sum test for the WT data and for the knockout data. This gives us a confidence interval for the knockout data and for the WT data. We only consider the result significant, if the Wilcoxon rank-sum test between the tdTomato positive and tdTomato negative cells is significant AND there is no overall between the confidence intervals for the knockout and the WT data, i.e. the two bars in the same row in the figure below do not overlap.

```
# Prepare for plotting
df_COSICC_kinetics_results$path <- factor(df_COSICC_kinetics_results$path, levels = df_COSICC_kinetics_

ggplot(df_COSICC_kinetics_results) +
  # Control CI
  geom_errorbarh(aes(y = path, xmin = lower_ci_bound_control, xmax = upper_ci_bound_control),
    height = 0.3, color = "#0072B2", size = 2, alpha = 0.6) + # blue
  # Case CI
  geom_errorbarh(aes(y = path, xmin = lower_ci_bound_case, xmax = upper_ci_bound_case,
    color = convidence_interval_overlapping_with_control),
    height = 0.1, size = 2) +
  scale_color_manual(
    name = "CI Overlap",
    values = c("TRUE" = "#E69F00", "FALSE" = "#CC79A7"), # orange and purple
    labels = c("TRUE" = "Overlap", "FALSE" = "No Overlap")
  ) +
  labs(
    x = "Location parameter (Wilcoxon test)",
    y = "Lineage trajectory",
    title = "Confidence intervals from COSICC_kinetics\n(blue for WT data,
    magenta/yellow for knockout data"
  ) +
```

```
theme_classic()
```

Confidence intervals from COSICC_kinetics
(blue for WT data,
magenta/yellow for knockout data)

