

se\_paul

Claire Man

3/23/2020

```
load("../data/se_paul.rda")
```

```
library(SingleCellExperiment)
```

```
## Loading required package: SummarizedExperiment
## Warning: package 'SummarizedExperiment' was built under R version 3.6.2
## Loading required package: GenomicRanges
## Loading required package: stats4
## Loading required package: BiocGenerics
## Loading required package: parallel
##
## Attaching package: 'BiocGenerics'
## The following objects are masked from 'package:parallel':
##
##   clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##   clusterExport, clusterMap, parApply, parCapply, parLapply,
##   parLapplyLB, parRapply, parSapply, parSapplyLB
## The following objects are masked from 'package:stats':
##
##   IQR, mad, sd, var, xtabs
## The following objects are masked from 'package:base':
##
##   anyDuplicated, append, as.data.frame, basename, cbind, colnames,
##   dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
##   grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
##   order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##   rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
##   union, unique, unsplit, which, which.max, which.min
## Loading required package: S4Vectors
## Warning: package 'S4Vectors' was built under R version 3.6.2
##
## Attaching package: 'S4Vectors'
## The following object is masked from 'package:base':
##
##   expand.grid
## Loading required package: IRanges
```

```
## Warning: package 'IRanges' was built under R version 3.6.2
## Loading required package: GenomeInfoDb
## Loading required package: Biobase
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase)", and for packages 'citation("pkgname)".
## Loading required package: DelayedArray
## Warning: package 'DelayedArray' was built under R version 3.6.2
## Loading required package: matrixStats
##
## Attaching package: 'matrixStats'
##
## The following objects are masked from 'package:Biobase':
##
##     anyMissing, rowMedians
## Loading required package: BiocParallel
## Warning: package 'BiocParallel' was built under R version 3.6.2
##
## Attaching package: 'DelayedArray'
##
## The following objects are masked from 'package:matrixStats':
##
##     colMaxs, colMins, colRanges, rowMaxs, rowMins, rowRanges
## The following objects are masked from 'package:base':
##
##     aperm, apply, rowsum
sim <- SingleCellExperiment(se)
counts <- counts(assay(sim))
sim <- SingleCellExperiment(assays = List(counts = counts))
```

## Gene Filtering (unnecessary)

```
geneFilter <- apply(counts,1,function(x){ sum(x >= 3) >= 10 }) sim <- sim[geneFilter, ]
```

## Normalization

```
FQnorm <- function(counts){
  rk <- apply(counts,2,rank,ties.method='min')
  counts.sort <- apply(counts,2,sort)
  reldist <- apply(counts.sort,1,median)
  norm <- apply(rk,2,function(r){ reldist[r] })
  rownames(norm) <- rownames(counts)
  return(norm)
```

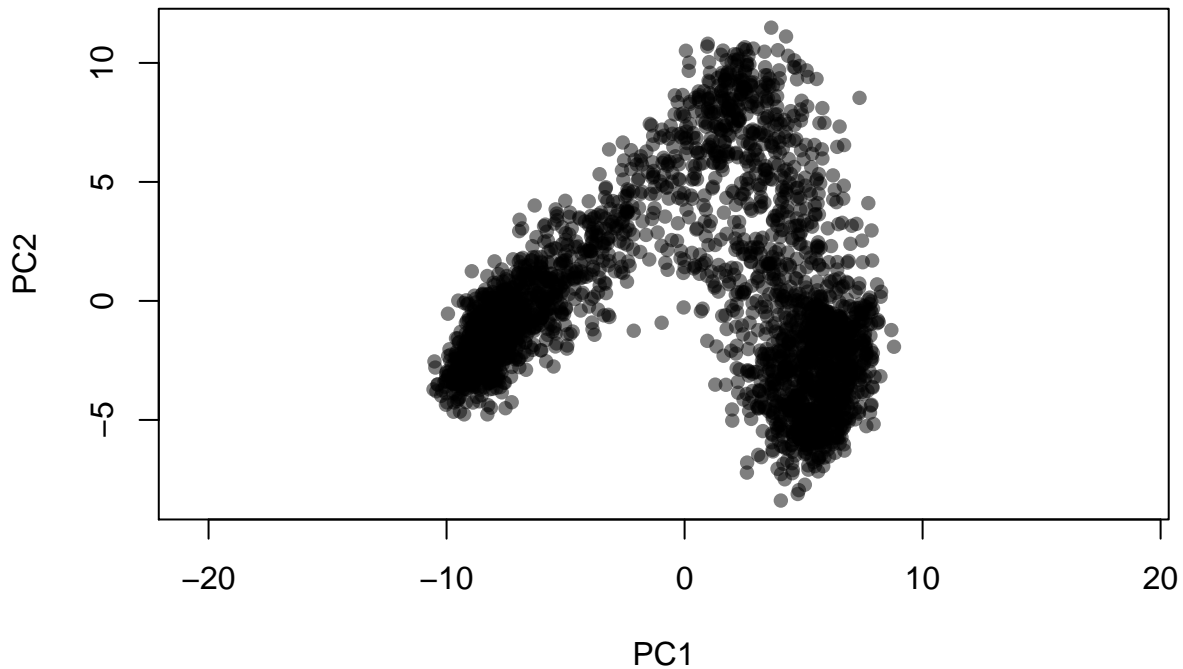
```
}
assays(sim)$norm <- FQnorm(assays(sim)$counts)
```

## Dimensionality Reduction

### PCA

```
pca <- prcomp(t(log1p(assays(sim)$norm)), scale. = FALSE)
rd1 <- pca$x[,1:2]
```

```
plot(rd1, col = rgb(0,0,0,.5), pch=16, asp = 1)
```



```
reducedDims(sim) <- SimpleList(PCA = rd1)
```

### UMAP

```
library(umap)
```

```
# sce <- runUMAP(sce, dimred="PCA")
```

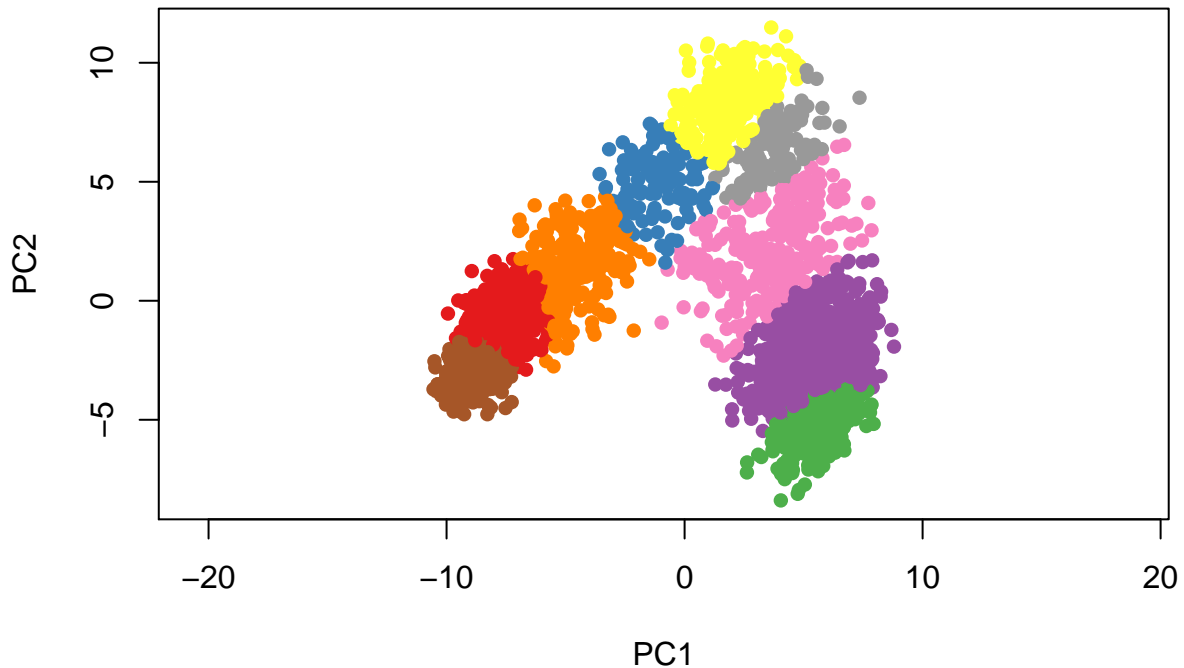
## Clustering Cells

```
library(mclust, quietly = TRUE)
```

```
## Package 'mclust' version 5.4.5
## Type 'citation("mclust")' for citing this R package in publications.
```

```
cl1 <- Mclust(rd1)$classification
colData(sim)$GMM <- cl1
```

```
library(RColorBrewer)
plot(rd1, col = brewer.pal(9,"Set1")[c11], pch=16, asp = 1)
```



## Fit Slingshot

```
library(slingshot)

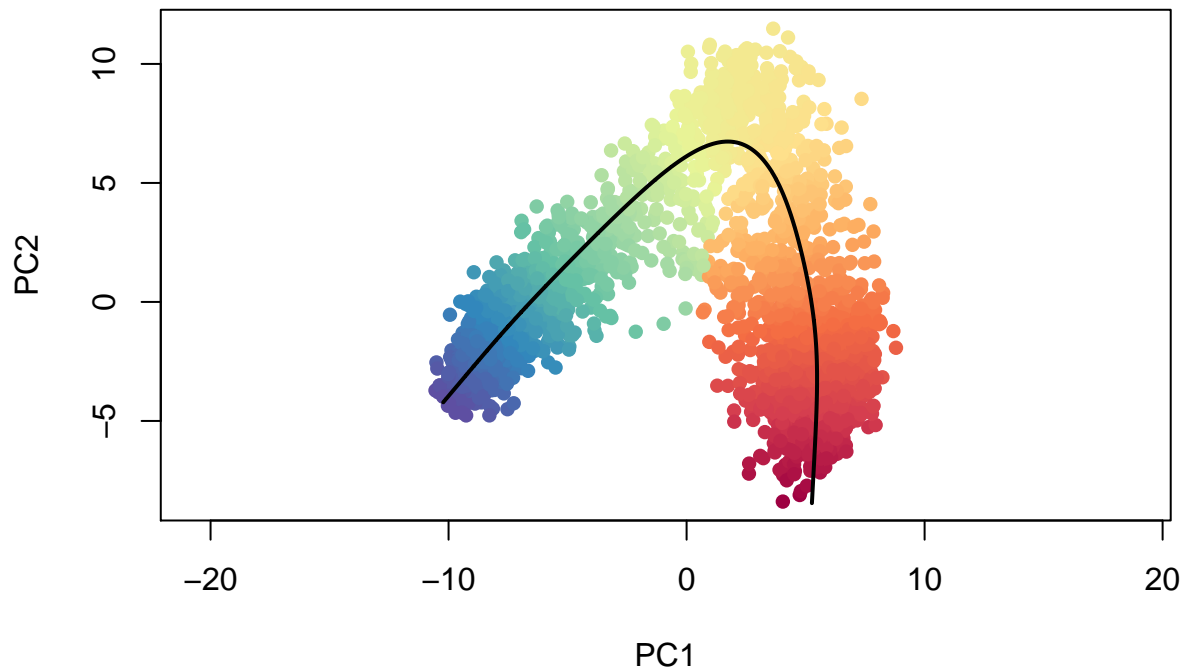
## Loading required package: printrcurve
sim <- slingshot(sim, clusterLabels = 'GMM', reducedDim = "PCA")

## Using full covariance matrix
summary(sim$slingPseudotime_1)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.000   6.459   15.856   16.431   27.541   33.006

colors <- colorRampPalette(brewer.pal(11,'Spectral')[-6])(100)
plotcol <- colors[cut(sim$slingPseudotime_1, breaks=100)]

plot(reducedDims(sim)$PCA, col = plotcol, pch=16, asp = 1)
lines(SlingshotDataSet(sim), lwd=2, col='black')
```



## Extract Curves

```
SlingshotDataSet(sim)
```

```
## class: SlingshotDataSet
##
## Samples Dimensions
## 2660 2
##
## lineages: 1
## Lineage1: 3 4 8 9 6 2 5 1 7
##
## curves: 1
## Curve1: Length: 33.006 Samples: 2660
```

```
# slingCurves(sim)$curve1
```

```
head(slingPseudotime(sim))
```

```
##      curve1
## W31105 16.526780
## W31106  5.553923
## W31107 29.179169
## W31108  2.416634
## W31109 29.561765
## W31110  6.631315
```

## Trade Seq

```
library(tradeSeq)
```

```

## Warning: namespace 'bigmemory' is not available and has been replaced
## by .GlobalEnv when processing object 'se'

## Warning: namespace 'bigmemory' is not available and has been replaced
## by .GlobalEnv when processing object 'se'

## tradeSeq has been updated to accommodate singleCellExperiment objects as output, making it much more
cl <- kmeans(rd1, centers = 7)$cluster
lin <- getLineages(rd1, clusterLabels = cl, start.clus = 1)

## Using full covariance matrix
crv <- getCurves(lin)

library(BiocParallel)
library(magrittr)

counts_mat <- as.matrix(counts)
sce <- fitGAM(counts = counts_mat,
              sds = crv)

## Warning in newton(lsp = lsp, X = G$X, y = G$y, Eb = G$Eb, UrS = G$UrS, L =
## G$L, : Iteration limit reached without full convergence - check carefully

## Warning in newton(lsp = lsp, X = G$X, y = G$y, Eb = G$Eb, UrS = G$UrS, L =
## G$L, : Iteration limit reached without full convergence - check carefully

## Warning in newton(lsp = lsp, X = G$X, y = G$y, Eb = G$Eb, UrS = G$UrS, L =
## G$L, : Iteration limit reached without full convergence - check carefully

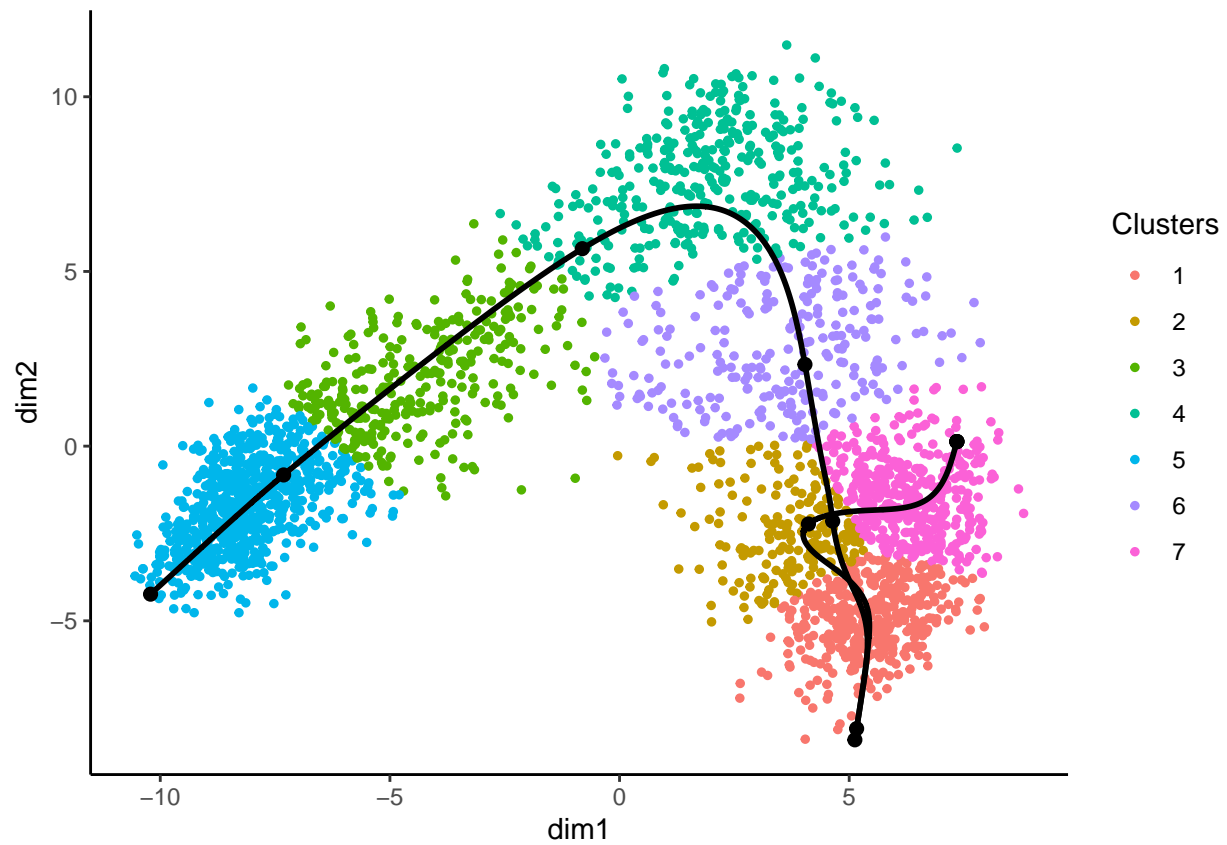
## Warning in newton(lsp = lsp, X = G$X, y = G$y, Eb = G$Eb, UrS = G$UrS, L =
## G$L, : Fitting terminated with step failure - check results carefully

## Warning in newton(lsp = lsp, X = G$X, y = G$y, Eb = G$Eb, UrS = G$UrS, L =
## G$L, : Fitting terminated with step failure - check results carefully

## Warning in newton(lsp = lsp, X = G$X, y = G$y, Eb = G$Eb, UrS = G$UrS, L =
## G$L, : Fitting terminated with step failure - check results carefully

## Warning in newton(lsp = lsp, X = G$X, y = G$y, Eb = G$Eb, UrS = G$UrS, L =
## G$L, : Iteration limit reached without full convergence - check carefully
plotGeneCount(curve = crv, counts = counts, clusters = cl,
              models = sce)

```



```
startRes <- startVsEndTest(sce)
oStart <- order(startRes$waldStat, decreasing = TRUE)
sigGeneStart <- names(sce)[oStart[1]]
plotSmoother(sce, counts, gene = sigGeneStart)
```

