Pseudotime

Ramiro

April 8, 2020

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Load library

```
library(monocle3,verbose = FALSE)
library(dplyr,verbose = FALSE)
```

Read the data

```
## Loads a sparse matrix RawCountsPseudotime
load("RawCountsPseudotime.rda")
dim(RawCountsPseudotime)
```

```
## [1] 20271  3127
```

Load cluster data

```
## Loads a factor variabele ClusterPseudotime containing cluster identities
load("ClusterPseudotime.rda")
## Change the name of variable to remind
DataClusters <- ClusterPseudotime

table(DataClusters)
```

```
## DataClusters
##        0.0        0.1        1.0        1.1       10.0       10.1       10.2
##        238        213        410         30          0          0         34
##   11.0_1_2       11.3       11.4       12.0       12.1       13.0       14.0
##          0          0          0          0          0          0          0
##       16.0       16.1       17.0       17.1       19.0       19.1        2.0
##        160         42        169         36          0          0        383
##       20.0       22.0       22.1     22.2_3       22.4       26.0       26.1
##          0          0          0          0          0          0          0
```

```
##       26.2     27.0_3       27.1       27.2      27.NA 28.0_2_3_4       28.1
##          0          0          0          0          0          0          0
##       29.0       29.1        3.0        3.1       30.0       31.0       33.0
##          0          0        253        119          0          0         90
##       34.0       35.0       35.1       36.0       37.0       37.1       38.0
##          0          0          0          0          0          0          0
##       39.0        4.0        4.1       40.0       41.0       42.0       42.1
##          0        206        142          0          0         40         27
##       43.0       44.0       45.0       46.0       48.0       49.0        5.0
##          0          0          0          0          0          0          0
##        5.1       50.0       51.0       51.1       52.0       53.0       54.0
##          0          0          0          0          0          0          0
##       55.0       56.0       57.0       59.0        6.0       60.0       61.0
##          0         43          0          0          0          0          0
##       62.0       63.0        7.0        8.0        8.1        8.2        8.3
##          0          0          0          0         77         27         15
##     10.0.0   10.0.1_4   10.0.2_3     10.0.5     10.1.0     10.1.1      8.0.0
##         45         59         57          9         33         21        103
##      8.0.1
##         46
```

```
length(DataClusters)
```

```
## [1] 3127
```

Convert DataClusters to a matrix format for input to Monocle

```
DataCluster.ID <- matrix(as.numeric(levels(DataClusters))[DataClusters], ncol = 1)
```

```
## Warning in matrix(as.numeric(levels(DataClusters))[DataClusters], ncol = 1): NAs
## introduced by coercion
```

```
rownames(DataCluster.ID) <- names(DataClusters)
colnames(DataCluster.ID) <- "Cluster.IDs"
DataCluster.ID[1:10,]
```

```
## cele-001-008.GATCAGTCAT cele-001-027.ACTCCGCCAA cele-001-042.TTCCTAGACC
##                       0                       0                       0
## cele-001-046.TTCTACGCCA cele-001-047.TTCGCTGCCT cele-001-047.ATGGAAGCAT
##                       0                       0                       0
## cele-001-064.AAGCTGACCT cele-001-065.GCCATCAACT cele-001-068.ACGGCAACCA
##                       0                       0                       0
## cele-001-071.GTCATTGCGC
##                       0
```

Generate matrix of gene short names for Monocle

```
geneNames <- matrix(rownames(RawCountsPseudotime), ncol = 1)
rownames(geneNames) <- rownames(RawCountsPseudotime)
colnames(geneNames) <- "gene_short_name"
head(geneNames)
```

```
##       gene_short_name
```

```
## aap-1 "aap-1"
## aat-1 "aat-1"
## aat-2 "aat-2"
## aat-3 "aat-3"
## aat-4 "aat-4"
## aat-5 "aat-5"
```

Initiate Monocle object

```
cds <- new_cell_data_set(expression_data = RawCountsPseudotime,
                         cell_metadata = DataCluster.ID,
                         gene_metadata = geneNames)


cds <- cds[,names(ClusterPseudotime[ClusterPseudotime %in% c("4.0","4.1")])] #previous trajectories
# cds <- cds[,names(ClusterPseudotime[ClusterPseudotime %in% c("3.0",
#                                                              "3.1",
#                                                              "4.0",
#                                                              "4.1",
#                                                              "16.0",
#                                                              "16.1",
#                                                              "17.0",
#                                                              "17.1",
#                                                              "0.0",
#                                                              "0.1",
#                                                              "1.0",
#                                                              "1.1",
#                                                              "2.0",
#                                                              "33.0",
#                                                              "42.0",
#                                                              "42.1",
#                                                              "56.0",
#                                                              "8.0.0",
#                                                              "8.0.1",
#                                                              "8.1",
#                                                              "8.2",
#                                                              "8.3",
#                                                              "10.0.0",
#                                                              "10.0.1_4",
#                                                              "10.0.2_3",
#                                                              "10.0.5",
#                                                              "10.1.0",
#                                                              "10.1.1",
#                                                              "10.2",
#                                                              "56.0")])] #all data for trajectories



colData(cds)
```

```
## DataFrame with 348 rows and 2 columns
##                          Cluster.IDs       Size_Factor
##                            <numeric>         <numeric>
```

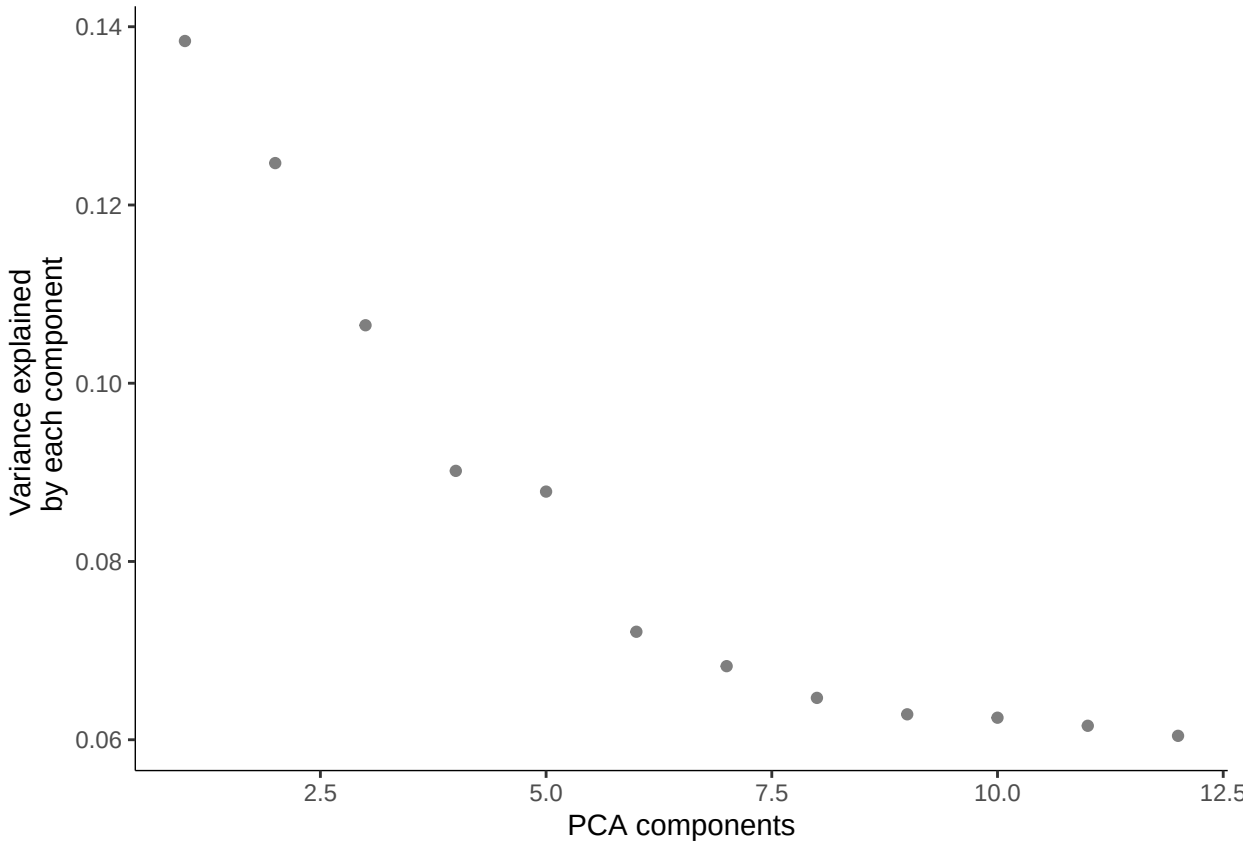
```
## cele-001-001.CTGATCGACC           4 0.602983439949076
## cele-001-004.GCCTCAGCAT           4 0.773978743815232
## cele-001-007.CTGATTAAGA           4 0.701980721134745
## cele-001-011.AGGTAATAGG           4 0.557984675773772
## cele-001-011.CAGGAGGAGA           4 0.458987394588103
## ...                             ...               ...
## cele-010-078.CCATAAGTCC         4.1  1.55695724046552
## cele-010-084.TAGAATAGCC         4.1  1.40396144226949
## cele-010-086.CCTATAAGCT         4.1  4.97236344137111
## cele-010-088.TATCGTCGGC         4.1  2.14194117474448
## cele-010-091.AAGTACGTTA         4.1  1.25996539690852
```

```
## Column Cluster.IDs contains the original DataCluster IDs
colData(cds)$Cluster.IDs <- factor(colData(cds)$Cluster.IDs)
colData(cds)
```

```
## DataFrame with 348 rows and 2 columns
##                          Cluster.IDs       Size_Factor
##                             <factor>         <numeric>
## cele-001-001.CTGATCGACC           4 0.602983439949076
## cele-001-004.GCCTCAGCAT           4 0.773978743815232
## cele-001-007.CTGATTAAGA           4 0.701980721134745
## cele-001-011.AGGTAATAGG           4 0.557984675773772
## cele-001-011.CAGGAGGAGA           4 0.458987394588103
## ...                             ...               ...
## cele-010-078.CCATAAGTCC         4.1  1.55695724046552
## cele-010-084.TAGAATAGCC         4.1  1.40396144226949
## cele-010-086.CCTATAAGCT         4.1  4.97236344137111
## cele-010-088.TATCGTCGGC         4.1  2.14194117474448
## cele-010-091.AAGTACGTTA         4.1  1.25996539690852
```

Step 1: Normalize and pre-process the data

```
cds <- preprocess_cds(cds, num_dim = 12)
plot_pc_variance_explained(cds)
```



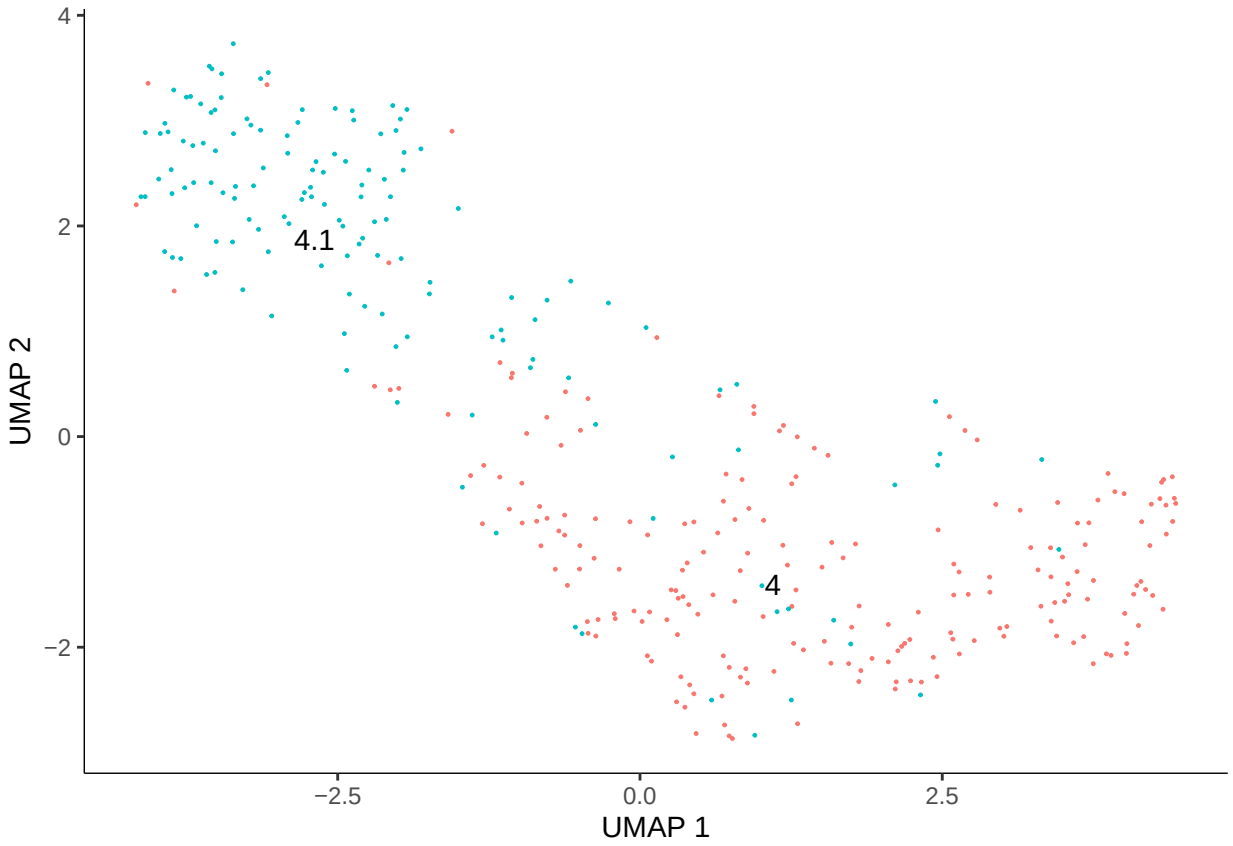
Step 2: Reduce the dimensions using UMAP

```
cds <- reduce_dimension(cds,umap.min_dist = 0.1,cores = 8)
```

```
## No preprocess_method specified, using preprocess_method = 'PCA'
## Note: reduce_dimension will produce slightly different output each time you run it unless you set 'u
```

```
plot_cells(cds, color_cells_by = "Cluster.IDs", group_label_size = 4, cell_size = 0.5)
```

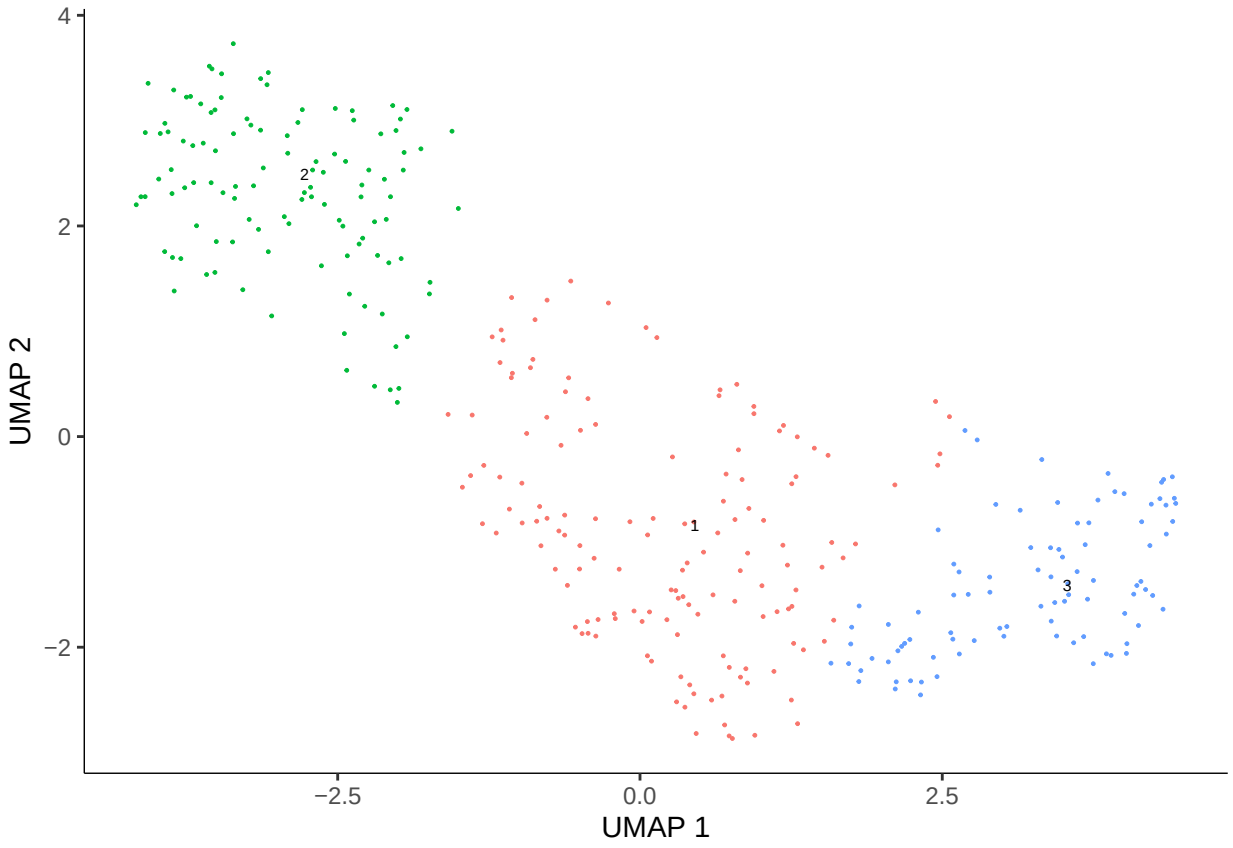
```
## No trajectory to plot. Has learn_graph() been called yet?
```



##Step 3: Cluster the cells

```
# cds = cluster_cells(cds, resolution=0.02)
cds = cluster_cells(cds, resolution=0.02)
plot_cells(cds, cell_size = 0.5)
```

```
## No trajectory to plot. Has learn_graph() been called yet?
```



Step 4: Learn a graph and order cells

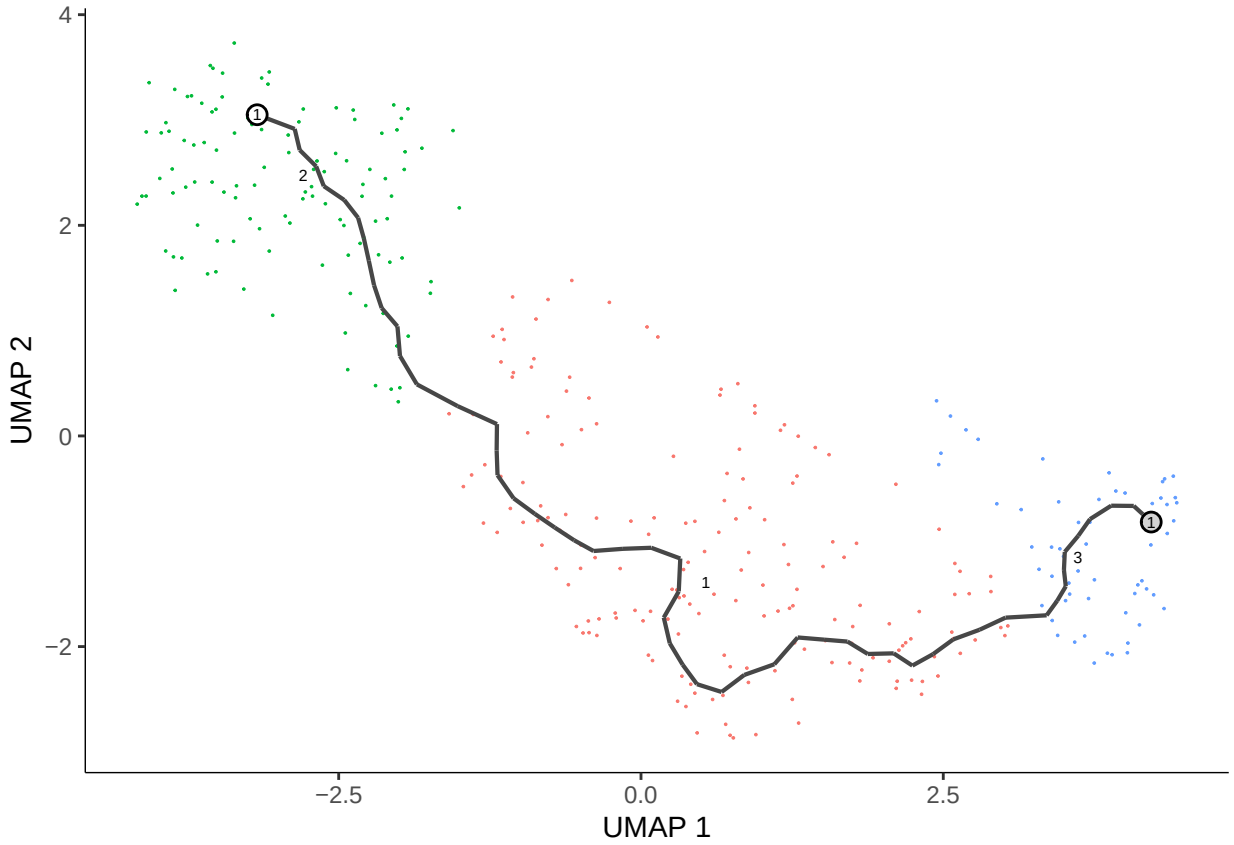
```
cds <- learn_graph(cds)
```

```
##
  |
  |                                                                      |   0%
  |
  |======================================================================| 100%
```

```
## With Shiny
# cds <- order_cells(cds)
# #
# save(file = "cdsRoots.rda", cds) #Save object state after selection of roots

## Without Shiny
load("cdsRoots.rda") #Load object

## Get info by: cds@principal_graph_aux$UMAP$root_pr_nodes

# cds <- order_cells(cds, root_pr_nodes = c("Y_8", "Y_9", "Y_12", "Y_14", "Y_15", "Y_31", "Y_34", "Y_43
# cds <- order_cells(cds, root_pr_nodes = c("Y_4",   "Y_17",  "Y_30",  "Y_33",  "Y_46",  "Y_84",  "Y_12

print(cds@principal_graph_aux$UMAP$root_pr_nodes)
```

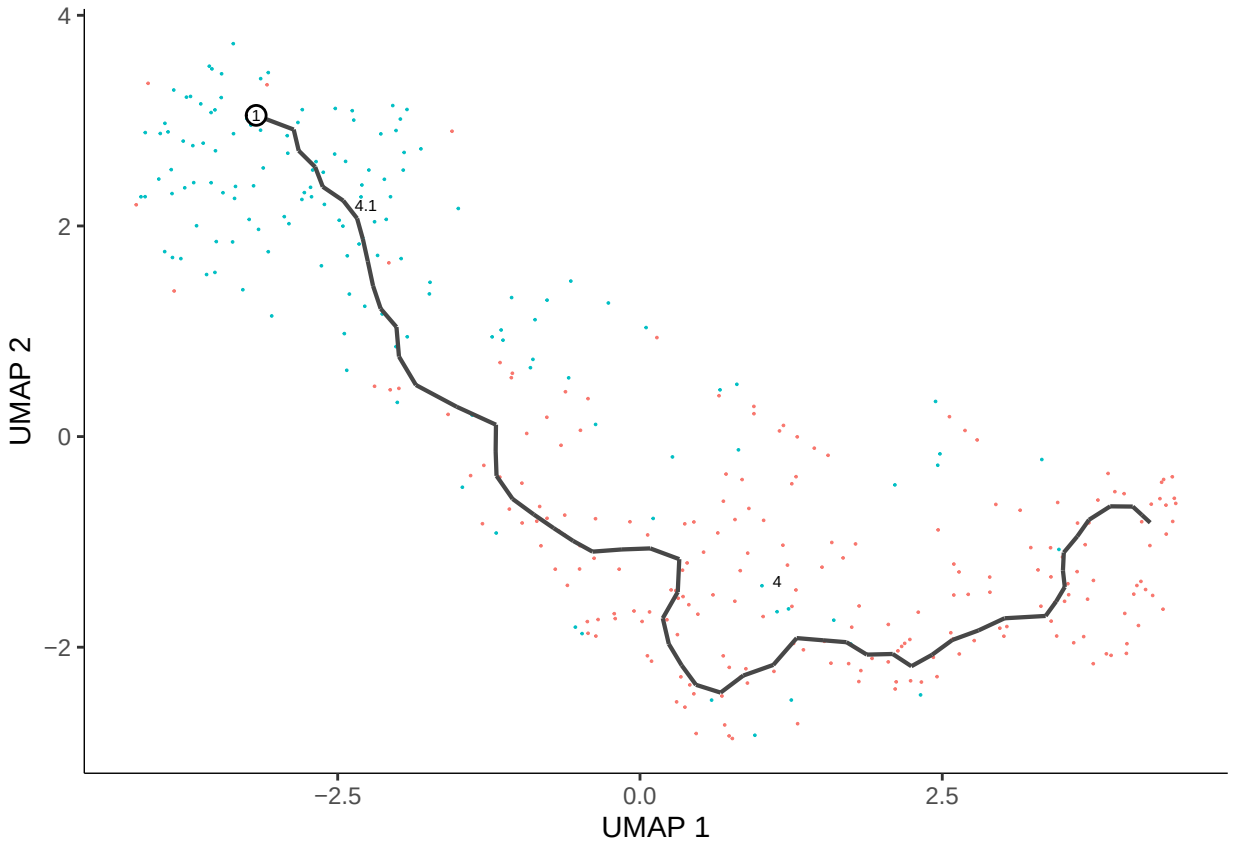
```
## [1] "Y_52"
```

```
plot_cells(cds)
```



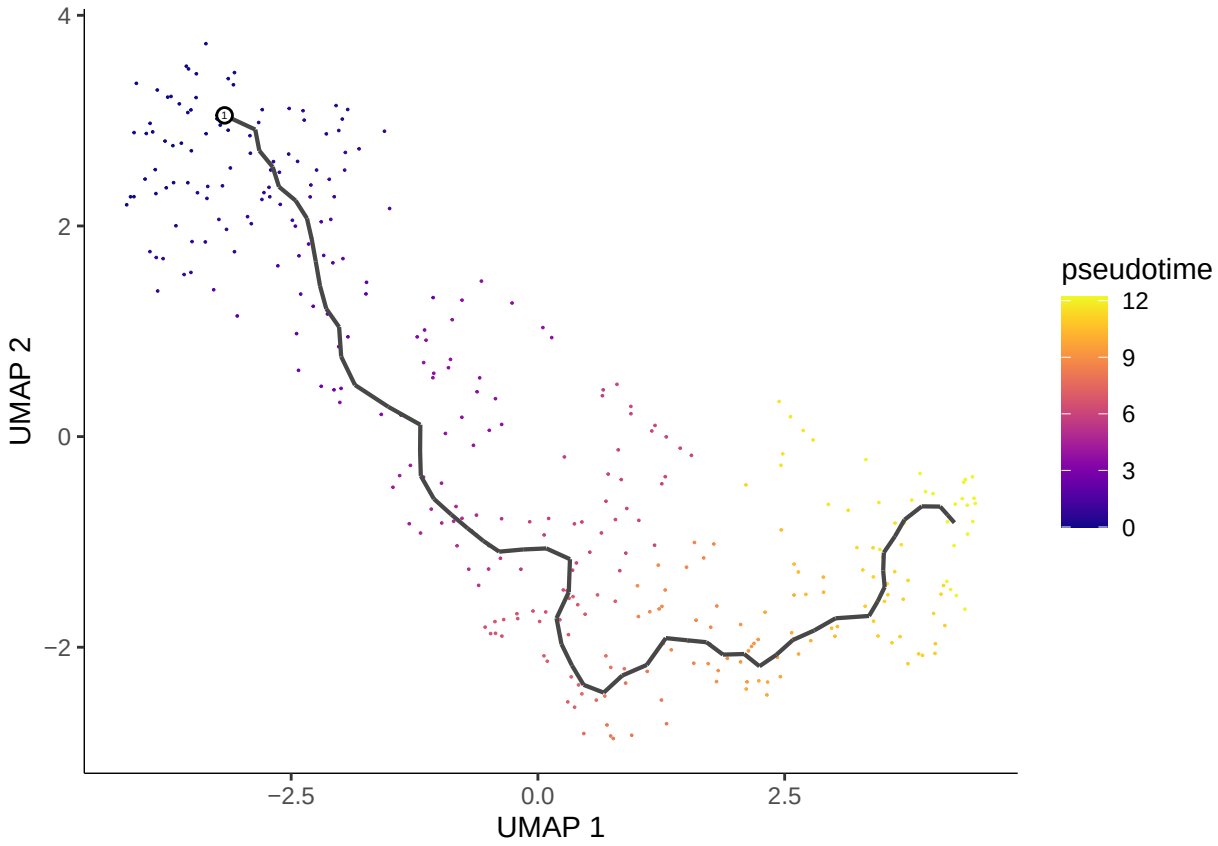
##Plot the pseudotime graph

```
plot_cells(cds,
           color_cells_by = "Cluster.IDs",
           label_groups_by_cluster=FALSE,
           label_leaves=FALSE,
           label_branch_points=FALSE)
```



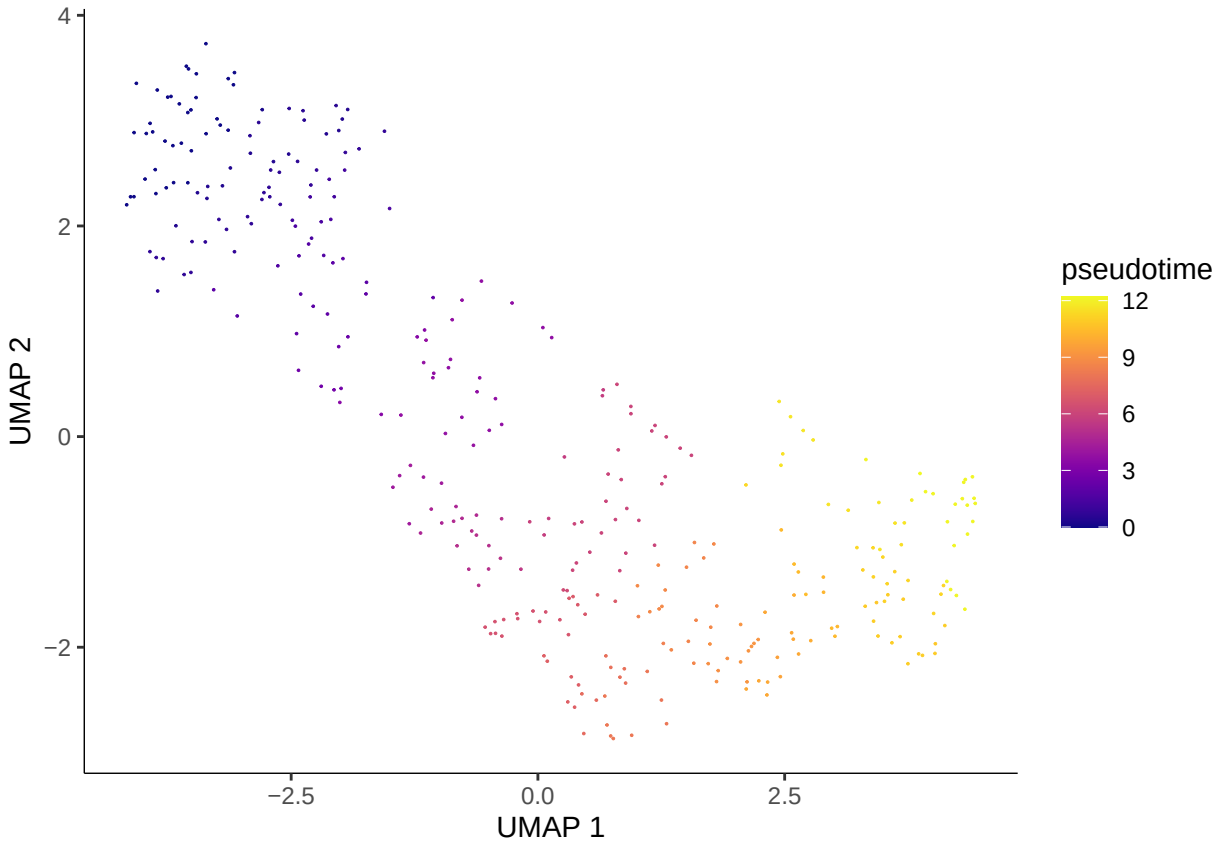
##Plot the pseudotime graph

```
plot_cells(cds,
           color_cells_by = "pseudotime",
           label_cell_groups=FALSE,
           label_leaves=FALSE,
           label_branch_points=FALSE,
           graph_label_size=1.5)
```



##Plot the pseudotime graph without trajectory

```
plot_cells(cds,
           color_cells_by = "pseudotime",
           label_cell_groups=FALSE,
           label_leaves=FALSE,
           label_branch_points=FALSE,
           graph_label_size=1.5,
           show_trajectory_graph = FALSE)
```



Correlation Analysis

Cluster 4

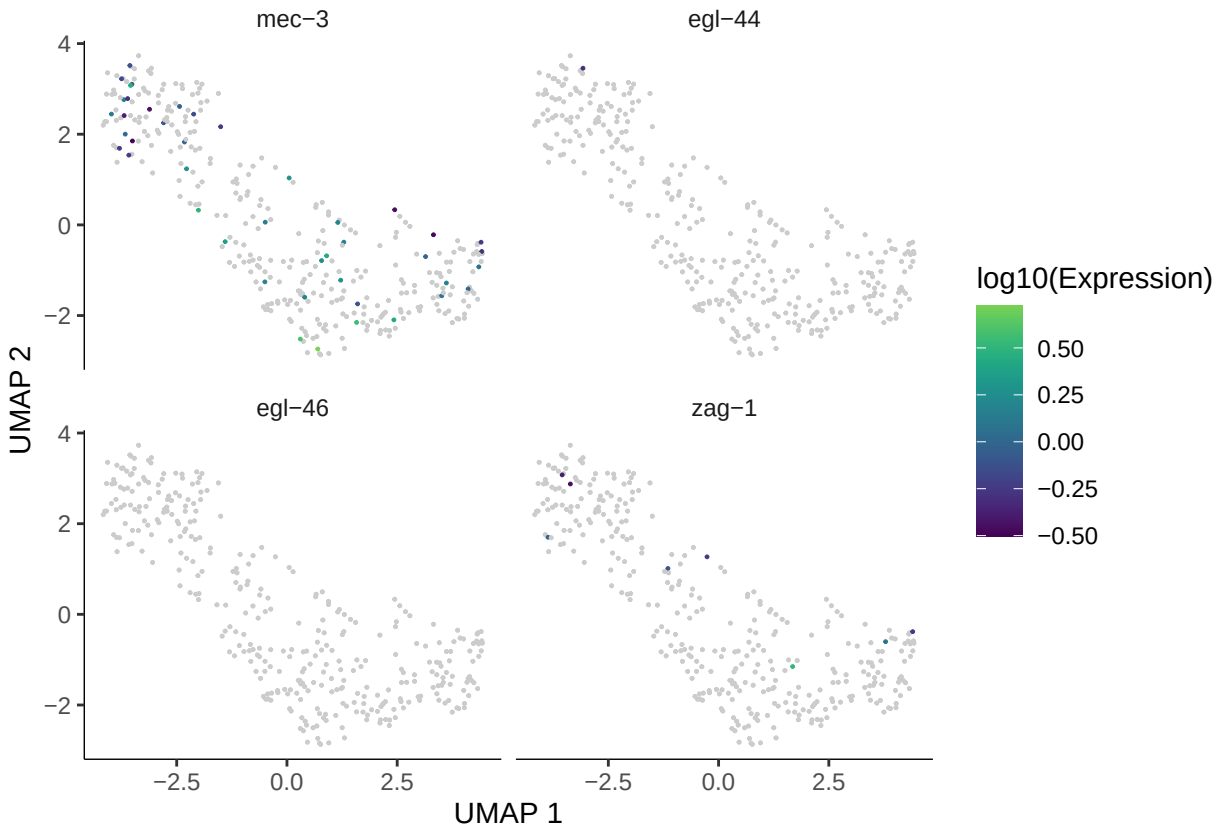
```
cluster4.cellNames <- rownames(pData(cds))[pData(cds)$Cluster.IDs %in% c(4, 4.1)]

cds_4 <- cds[,cluster4.cellNames]

cds4_pg <- graph_test(cds_4, neighbor_graph="principal_graph", cores=4, verbose = F)

cds4_genes <- cds4_pg %>%
  filter(q_value < 0.05) %>%
  arrange(desc(morans_I)) %>%
  select(gene_short_name)
```

```
cds4_genes$gene_short_name
```

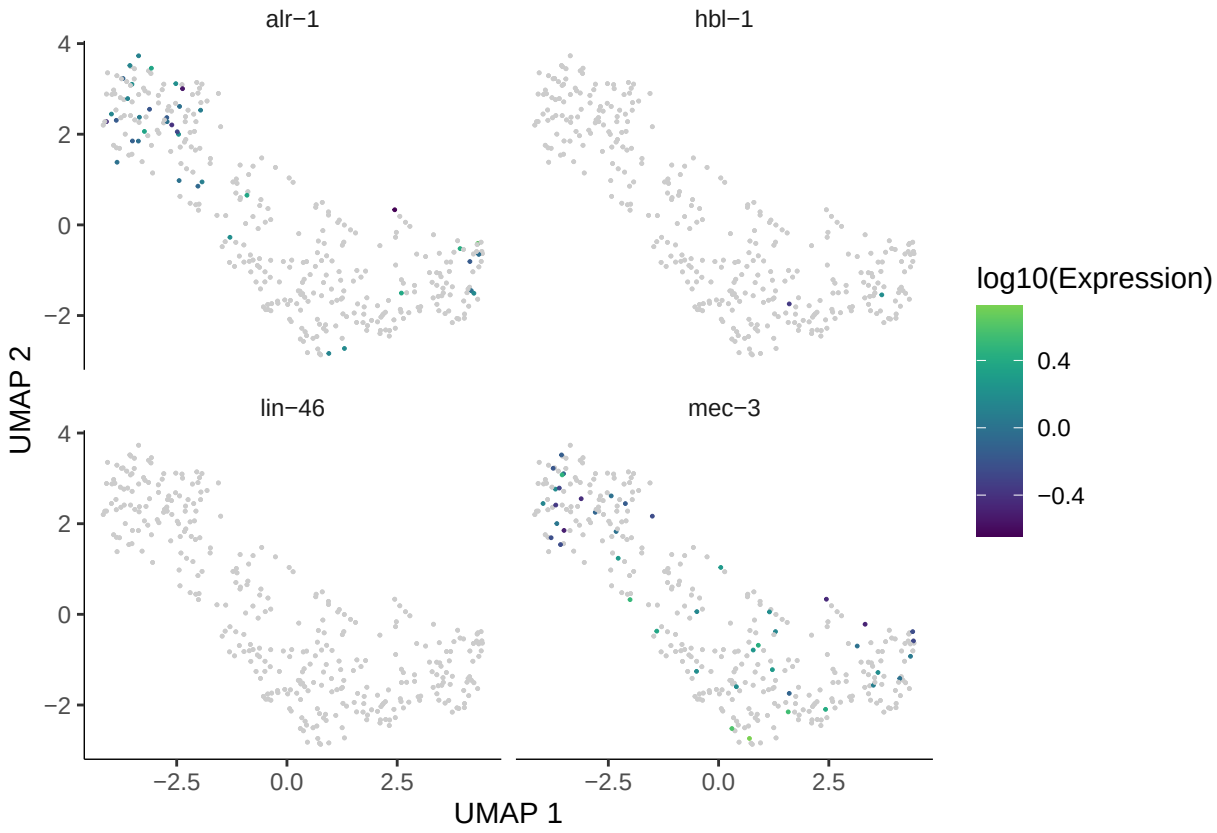
```
##   [1] mig-39     mec-12     T05A8.3    flp-14     mec-7      flp-8
##   [7] nep-21     mtd-1      R07A4.3    mec-17     Y48A5A.1   rsy-1
##  [13] cpna-4     hmr-1      C05D2.10   C03A3.2    egl-5      par-5
##  [19] art-1      glb-15     piki-1     cpna-2     his-71     hars-1
##  [25] nrde-2     F52H2.7    ZK265.7    vps-4      chdp-1     acp-6
##  [31] C44B11.4   tsg-101    frm-9      hot-4      ZK809.5    R106.5
##  [37] ahcy-1     F54G2.1    mlk-1      ain-1      Y69A2AR.16 nhr-6
##  [43] unc-14     cmd-1      tom-1      F28E10.1   rskn-2     mec-10
```

```
##  [49] F13H6.5     hex-1       mdf-1       F17C8.3     ddr-2       C17F4.7
##  [55] Y39G10AR.8  mca-3       cdh-4       kcnl-1      mam-2       ftt-2
##  [61] eat-4       duxl-1      aap-1       R08D7.5     F22F1.3     unc-15
##  [67] cyk-1       tag-180     mev-1       T01E8.8     Y57G11C.38  snt-6
##  [73] cct-8       flp-20      wrb-1       Y105C5B.25  tag-80      sto-5
##  [79] C53D5.1     T07F10.1    let-2       unc-104     eva-1       C09G1.4
##  [85] pll-1       lips-1      ZK1248.13   nlp-11      T24B1.1     C33D9.3
##  [91] rilp-1      dig-1       H28G03.2    enpl-1      ZK632.4     sem-4
##  [97] F08F8.9     cnnm-1      gcy-35      iars-2      npr-23      egl-21
## [103] Y52B11A.4   asg-2       eef-1A.2    Y69A2AR.1   nrx-1       nlp-7
## [109] C17H12.10   evl-20      F52D10.2    frm-1       F55A12.5    ncx-2
## [115] F52H2.6     glb-29      cutl-19     frpr-15     mtm-6       unc-115
## [121] sqv-7       cab-1       cyn-7       C02B8.3     C53C11.5    pgs-1
## [127] ubc-3       haf-3       nrfl-1      siah-1      F58D5.5     spc-1
## [133] mom-5       cct-5       ada-2       ctc-2       mec-9       cdc-5L
## [139] C50F2.3     txdc-9      T22C8.1     nuo-6       C08G5.6     ced-3
## [145] K02F2.5     gpd-2       atg-7       jnk-1       Y45G5AL.1   T11G6.4
## [151] T16H12.4    pbs-5       M176.5      cul-3       ced-4       acp-2
## [157] casy-1      cap-2       acc-1       stn-1       ekl-5       R02D3.7
## [163] unc-68      egl-3       nid-1       golg-4      lin-65      E01A2.2
## [169] C34C6.4     F32D1.11    eri-6       hke-4.1     F07C3.3     hsp-16.2
## [175] arrd-18     C06G3.5     Y54E10BR.3  epg-9       tbcb-1      nkb-1
## [181] R07E4.5     dhhc-3      unc-50      T02E9.5     alr-1       ufl-1
## [187] nra-1       lgc-12      mrp-1       Y82E9BR.3   glb-18      sup-12
## [193] egl-4       ddx-15      T14B1.1     aldo-1      H35B03.1    mec-4
## [199] Y119C1B.6   cpf-2       tppp-1      R08D7.4     lite-1      T01H3.2
## [205] deg-1       hmg-1.2     K07F5.8     M05D6.2     Y66D12A.8   ZK1098.2
## [211] mec-1       wts-1       cct-4       nol-5       B0261.8     cnb-1
## [217] ife-3       E04A4.5     trak-1      W02B8.2     F57G12.1    F40B5.2
## [223] die-1       soap-1      cdc-48.2    F18C12.3    T13H5.4     Y53F4B.13
## [229] T22F3.3     mdt-29      sup-1       imp-1       usp-5       flp-7
## [235] myo-2       B0491.5     tra-1       dhhc-2      tam-1       ZC506.1
## [241] K08E4.2     unc-22      tyra-3      Y57G11C.9   R09H10.3    F43D9.3
## [247] nhr-88      rig-1       col-166     nlp-29      Y43F4A.1    R151.8
## [253] F11A10.5    smo-1       icl-1       mab-5       sol-2       ceh-39
## [259] Y25C1A.7    K09A9.6     F46G11.4    B0495.8     sms-3       swsn-3
## [265] mmaa-1      mma-1       cdkr-3      erm-1       pfd-3       npp-16
## [271] mrpl-22     gcst-1      ptb-1       C07A12.7    mkk-4       ztf-8
## [277] athp-1      nlp-13      Y49A3A.3    R06A4.2     sumv-2      rnf-121
## [283] B0334.5     lad-2       C25H3.7     eat-17      csb-1       taco-1
## 20271 Levels: 2L52.1 2RSSE.1 4R79.2 6R55.2 aagr-1 aagr-2 aagr-3 aagr-4 ... zyx-1
```

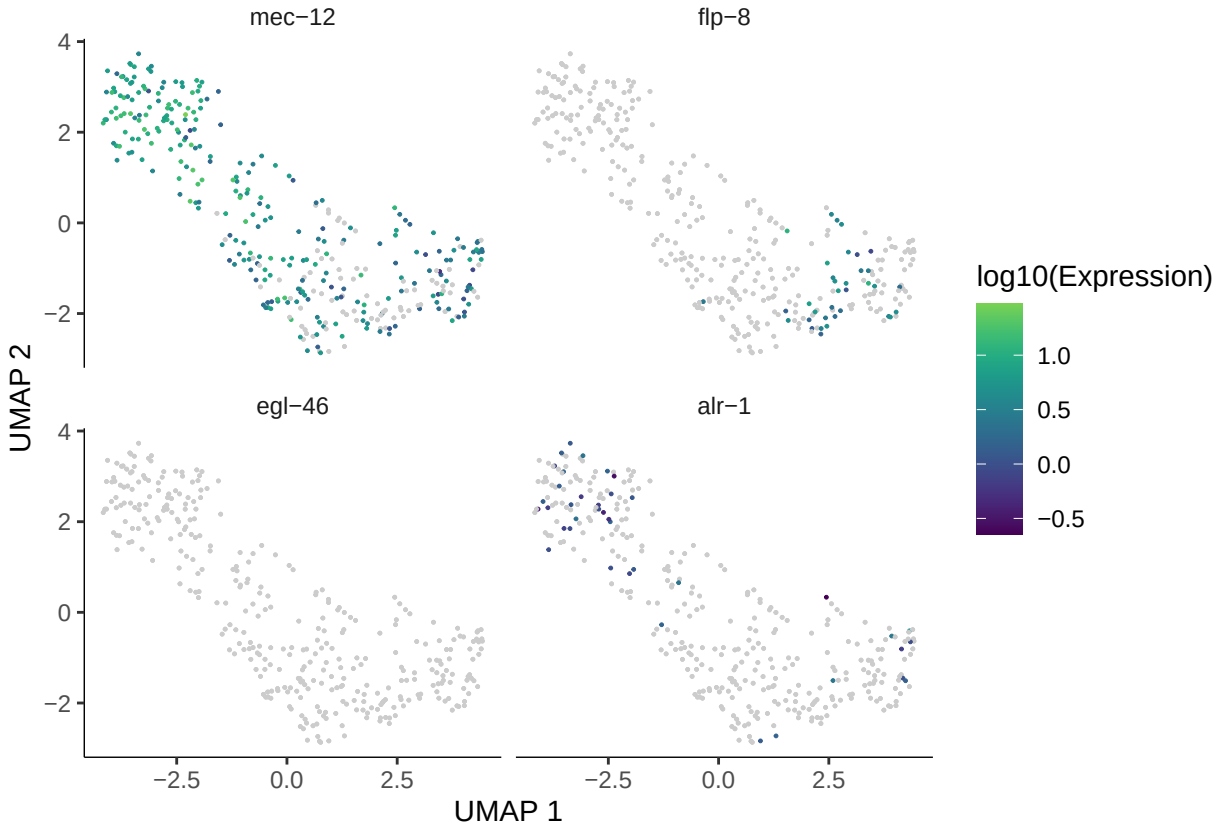
```
# Plot a few genes
plot_cells(cds_4, genes=c("mec-3", "egl-44", "egl-46", "zag-1"),
           show_trajectory_graph=FALSE,
           label_cell_groups=FALSE,
           label_leaves=FALSE,
           cell_size = 0.5)
```



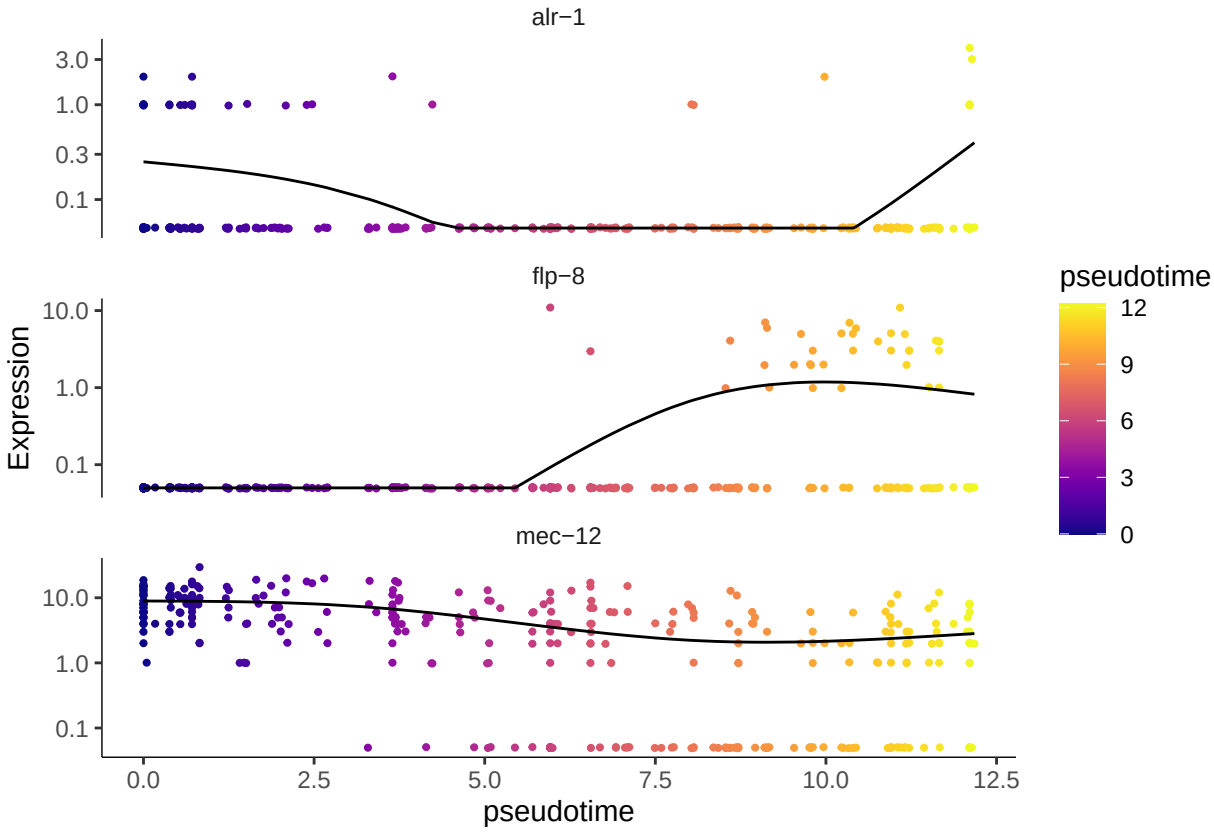

```
# Plot a few genes
plot_cells(cds_4, genes=c("alr-1", "hbl-1", "lin-46","mec-3"),
           show_trajectory_graph=FALSE,
           label_cell_groups=FALSE,
           label_leaves=FALSE,
           cell_size = 0.5)
```




```
# Plot a few genes
plot_cells(cds_4, genes=c("mec-12", "flp-8", "egl-46", "alr-1"),
           show_trajectory_graph=FALSE,
           label_cell_groups=FALSE,
           label_leaves=FALSE,
           cell_size = 0.5)
```




```
cds_4_lineage_cds <- cds_4[rowData(cds_4)$gene_short_name %in% c(2, "alr-1", "mec-12", "flp-8"),]
plot_genes_in_pseudotime(cds_4_lineage_cds,
                         # color_cells_by="embryo.time.bin",
                         min_expr=0.05)
```


alr−1
flp−8
mec−12
Expression
pseudotime
pseudotime
3.0
1.0
0.3
0.1
10.0
1.0
0.1
10.0
1.0
0.1
0.0
2.5
5.0
7.5
10.0
12.5
12
9
6
3
0