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.character(levels(DataClusters))[DataClusters], ncol = 1)
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"                   "0.0"                   "0.0"
## cele-001-046.TTCTACGCCA cele-001-047.TTCGCTGCCT cele-001-047.ATGGAAGCAT
##                   "0.0"                   "0.0"                   "0.0"
## cele-001-064.AAGCTGACCT cele-001-065.GCCATCAACT cele-001-068.ACGGCAACCA
##                   "0.0"                   "0.0"                   "0.0"
## cele-001-071.GTCATTGCGC
##                   "0.0"
```

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("3.0","3.1","4.0","4.1","16.0","16.1","1
cds <- cds[,names(ClusterPseudotime[ClusterPseudotime %in% c(
                                                                    # "0.0",
                                                                    # "0.1",
                                                                    # "1.0",
                                                                    # "1.1",
                                                                    # "2.0",
                                                                    # "3.0",
                                                                    # "3.1",
                                                                    # "4.0",
                                                                    # "4.1",
                                                                    # "16.0",
                                                                    # "16.1",
                                                                    # "17.0",
                                                                    # "17.1",
                                                                    # "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",
                                                                    # "33.0",
                                                                    # "42.0",
                                                                    "42.1",
                                                                    "56.0"
                                                                    )])] #all data for trajectories


colData(cds)
```

```
## DataFrame with 70 rows and 2 columns
##                           Cluster.IDs       Size_Factor
##                              <factor>         <numeric>
## cele-001-020.GCTCTCGCCT        42.1 0.557984675773772
## cele-001-031.GGATCTGCAG        42.1  1.14746848647026
```

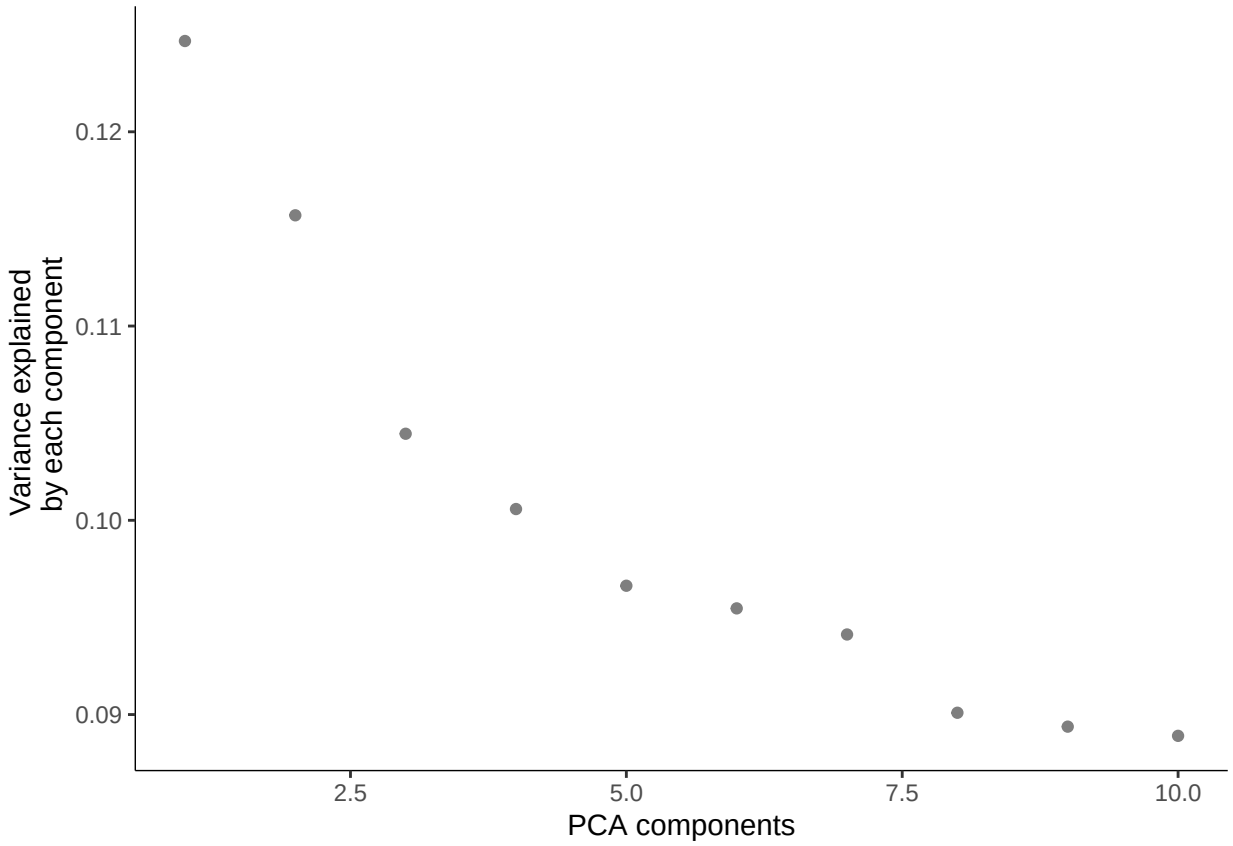
```
## cele-002-014.GTATACCGAA        42.1  1.30046428466629
## cele-002-033.TGATTCCTCA        42.1 0.625482822036728
## cele-003-074.ACGCGCTCCT        42.1 0.715480350387336
## ...                             ...               ...
## cele-009-075.AATACCAGTT        56.0  1.00347244110928
## cele-009-080.TGGCCATGCA        56.0  4.05888852861244
## cele-010-034.ACGCGTATCG        56.0   0.9764731826041
## cele-010-055.CTGGCTTCCT        56.0  2.71792535618837
## cele-010-071.TGCAGCCTAC        56.0  3.63140026894705
```

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

```
## DataFrame with 70 rows and 2 columns
##                          Cluster.IDs       Size_Factor
##                             <factor>         <numeric>
## cele-001-020.GCTCTCGCCT        42.1 0.557984675773772
## cele-001-031.GGATCTGCAG        42.1  1.14746848647026
## cele-002-014.GTATACCGAA        42.1  1.30046428466629
## cele-002-033.TGATTCCTCA        42.1 0.625482822036728
## cele-003-074.ACGCGCTCCT        42.1 0.715480350387336
## ...                             ...               ...
## cele-009-075.AATACCAGTT        56.0  1.00347244110928
## cele-009-080.TGGCCATGCA        56.0  4.05888852861244
## cele-010-034.ACGCGTATCG        56.0   0.9764731826041
## cele-010-055.CTGGCTTCCT        56.0  2.71792535618837
## cele-010-071.TGCAGCCTAC        56.0  3.63140026894705
```

Step 1: Normalize and pre-process the data

```
cds <- preprocess_cds(cds, num_dim = 10)
plot_pc_variance_explained(cds)
```



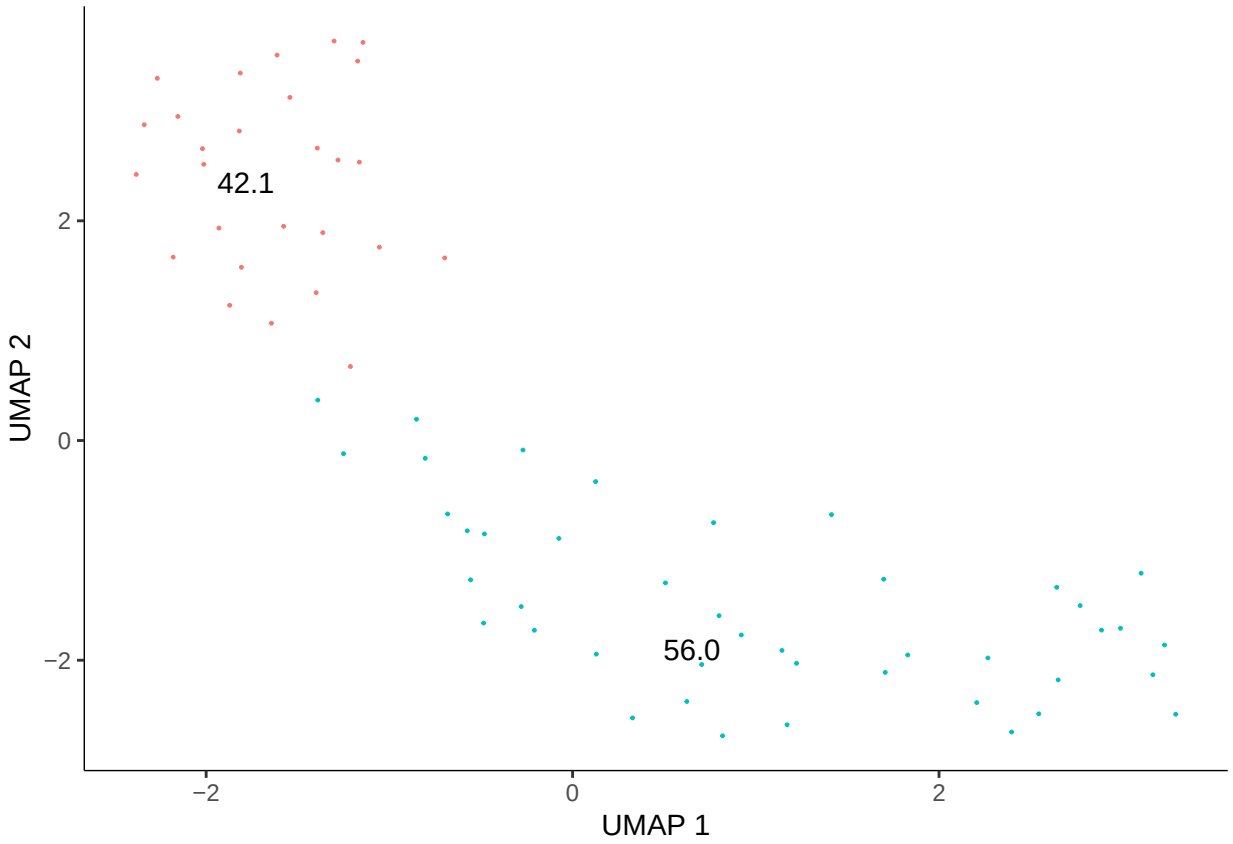
Step 2: Reduce the dimensions using UMAP

```
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,show_trajectory_g
```

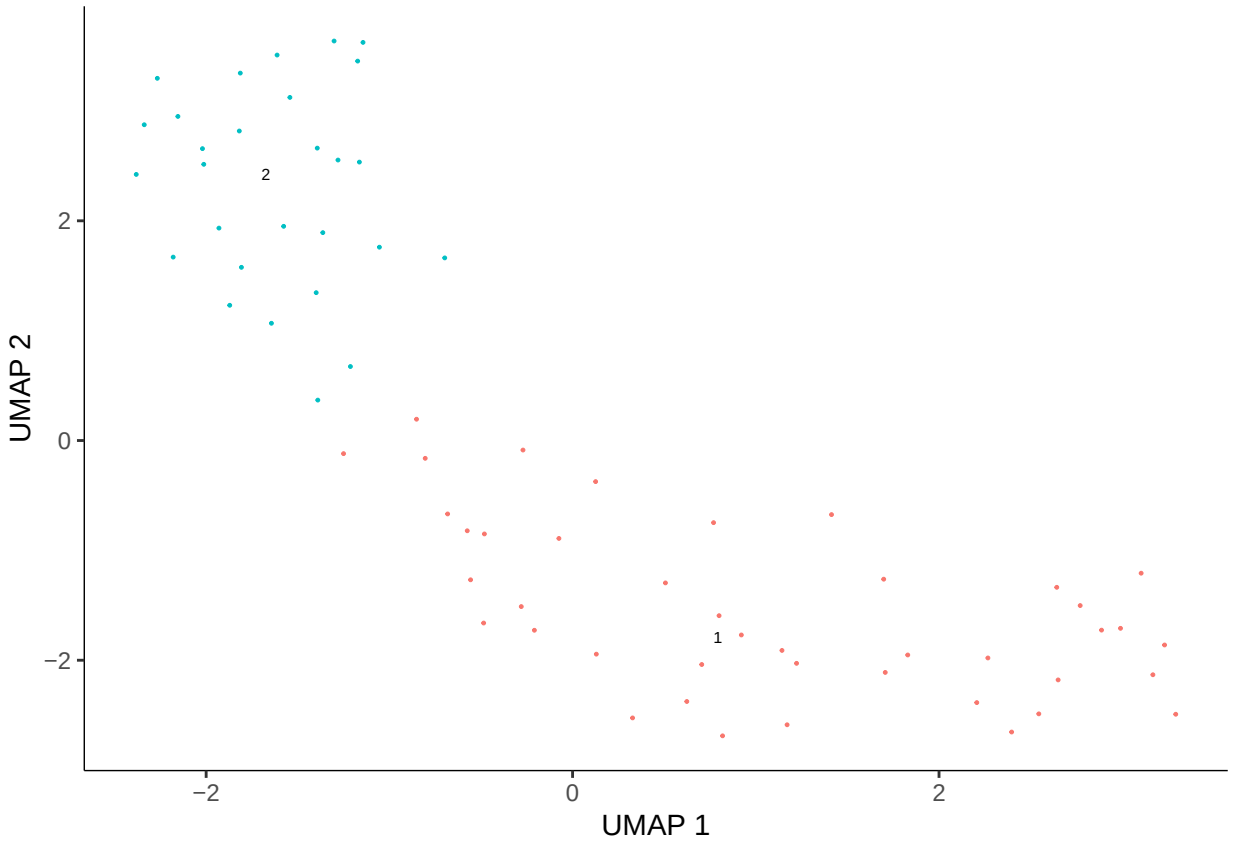
```
## No trajectory to plot. Has learn_graph() been called yet?
```



Step 3: Cluster the cells

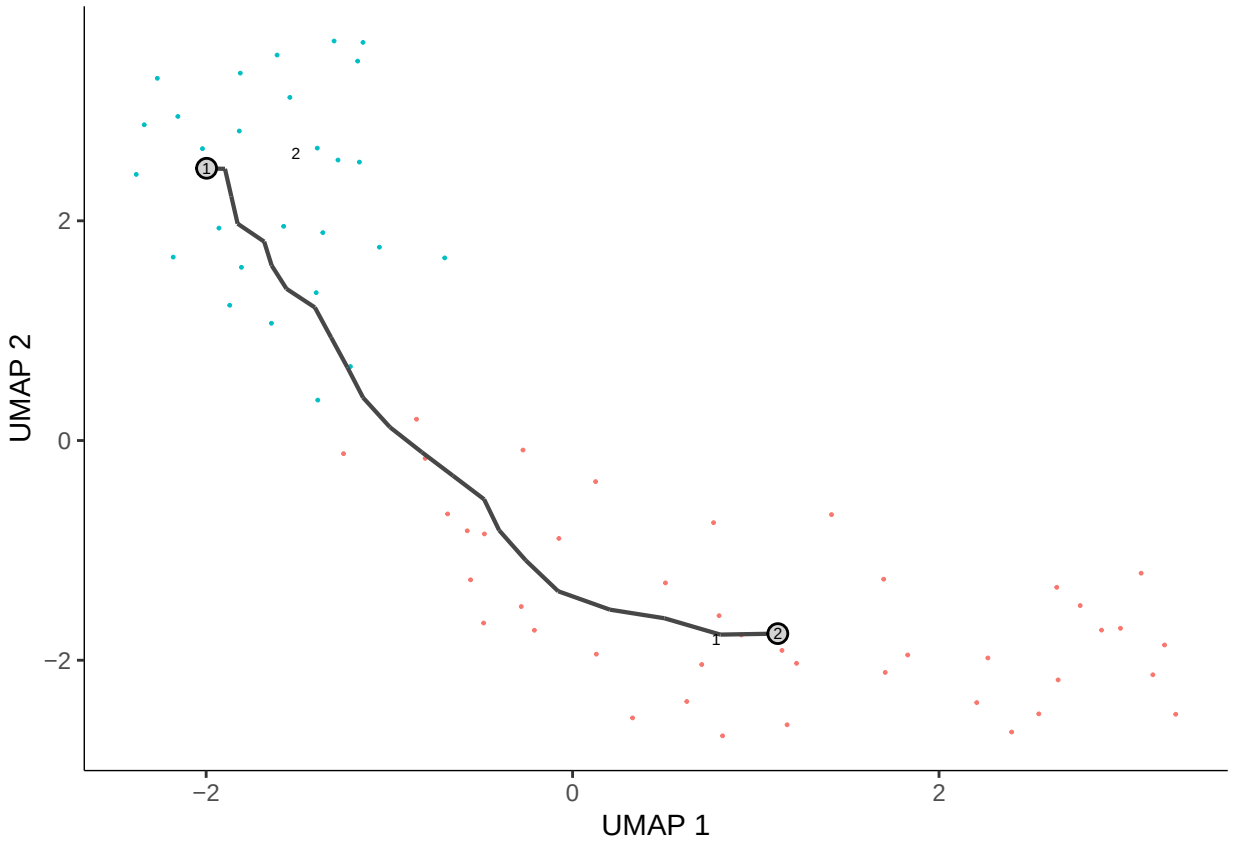
```
# cds = cluster_cells(cds, resolution=0.02)
cds = cluster_cells(cds, resolution=0.2)
plot_cells(cds, cell_size = 0.5)
```

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



Step 4: Learn a graph and order cells

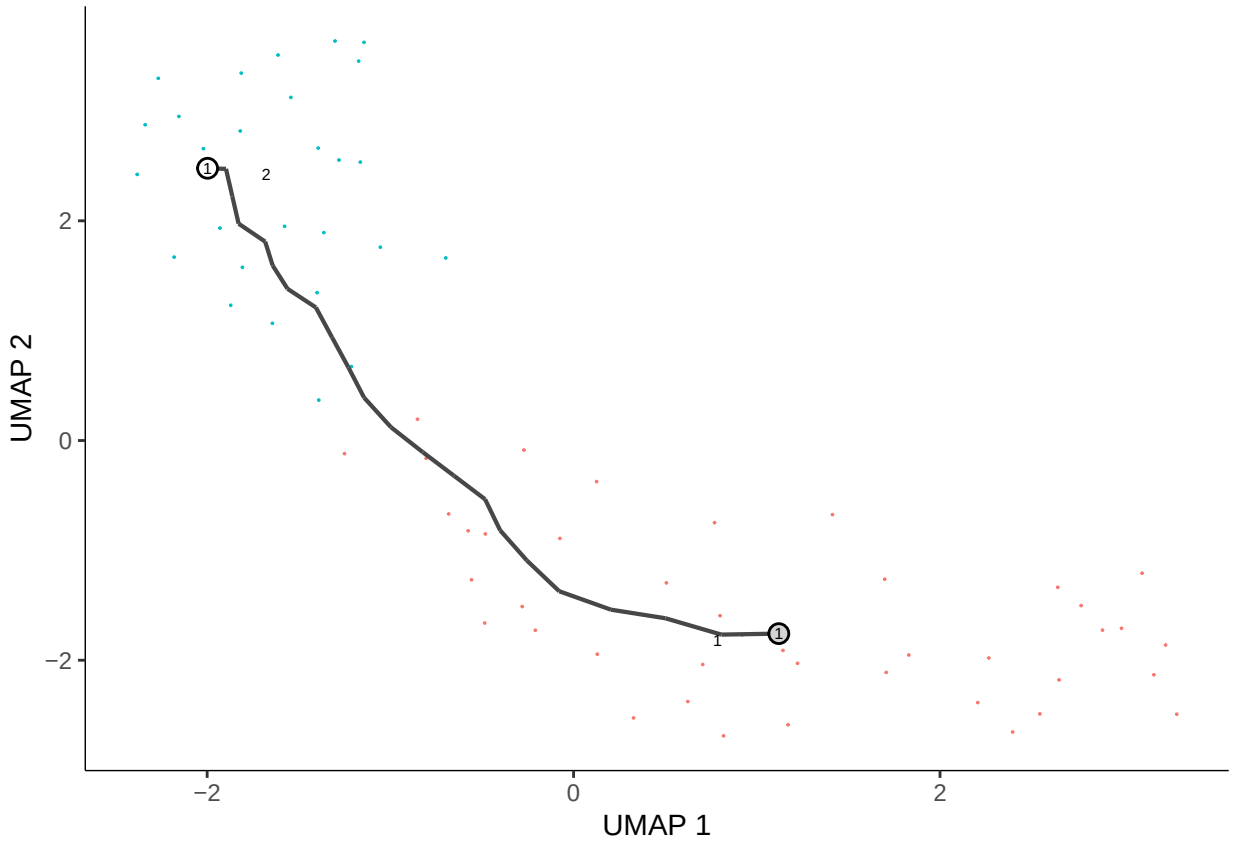
```
cds <- learn_graph(cds)
plot_cells(cds, cell_size = 0.5)
```




```
# ## 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)
```

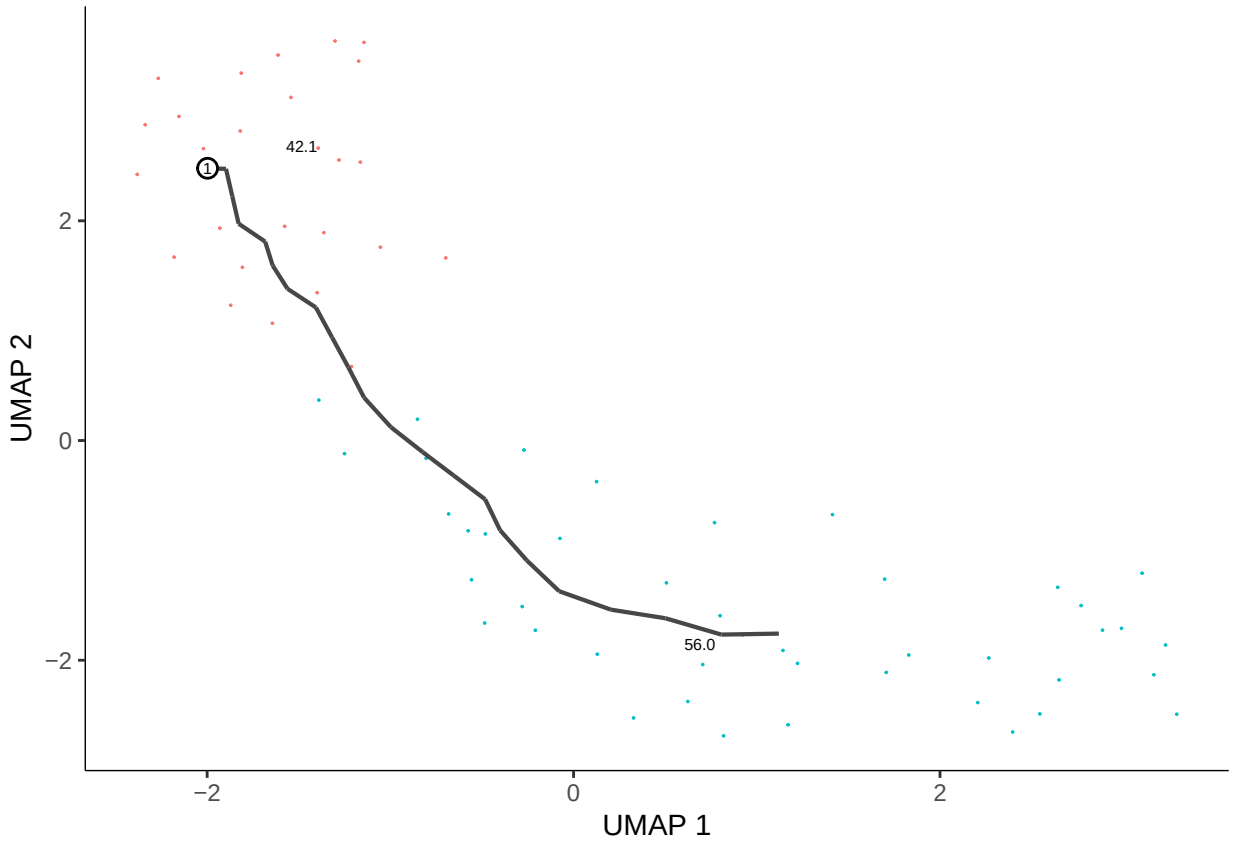
```
## [1] "Y_1"
```

```
plot_cells(cds)
```



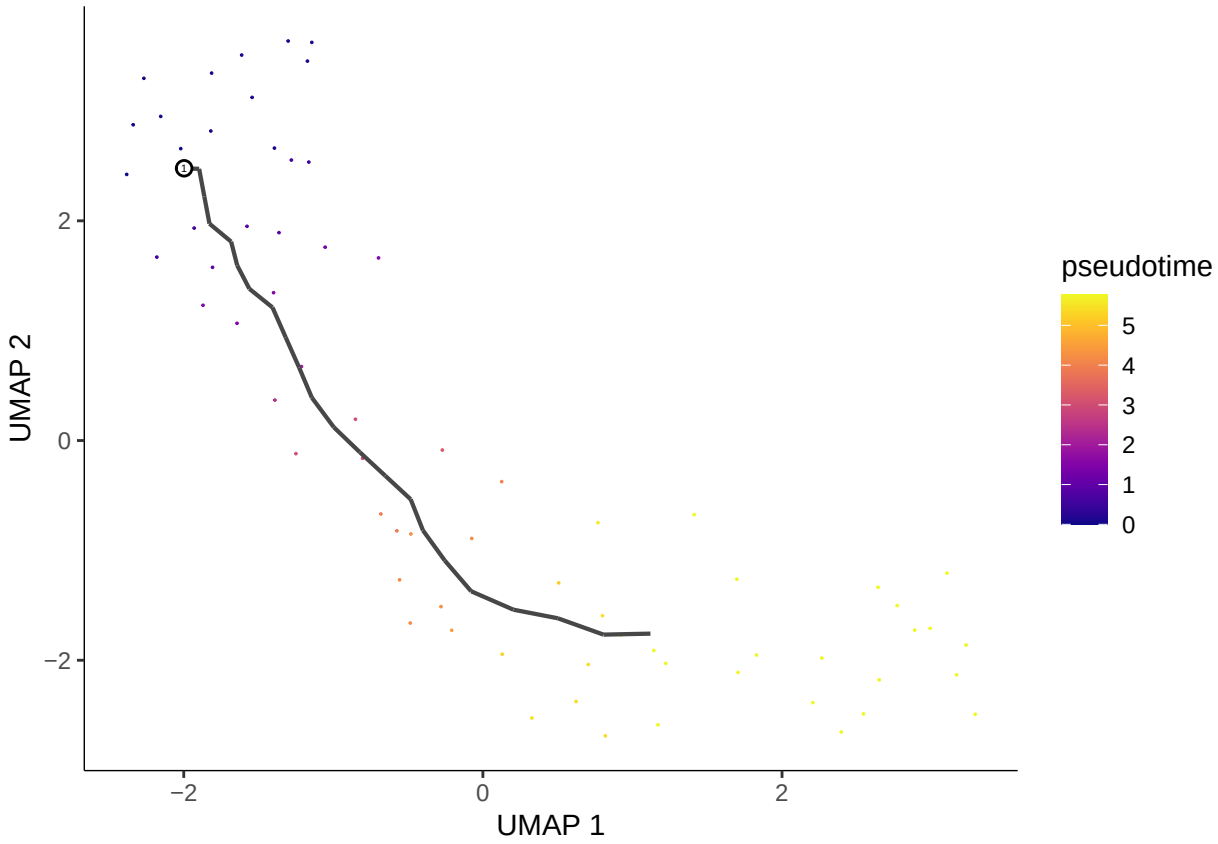
Plot the pseudotime graph

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



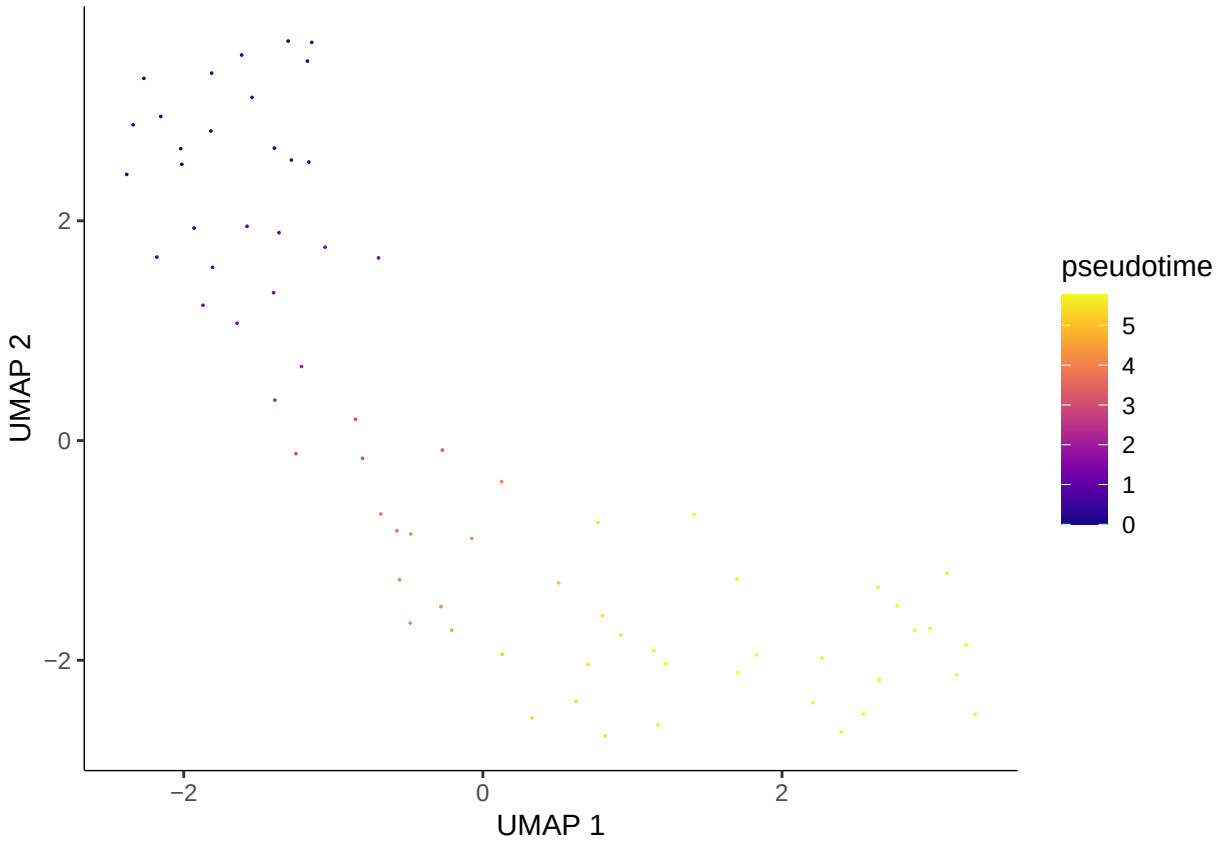
Plot the pseudotime graph

```
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

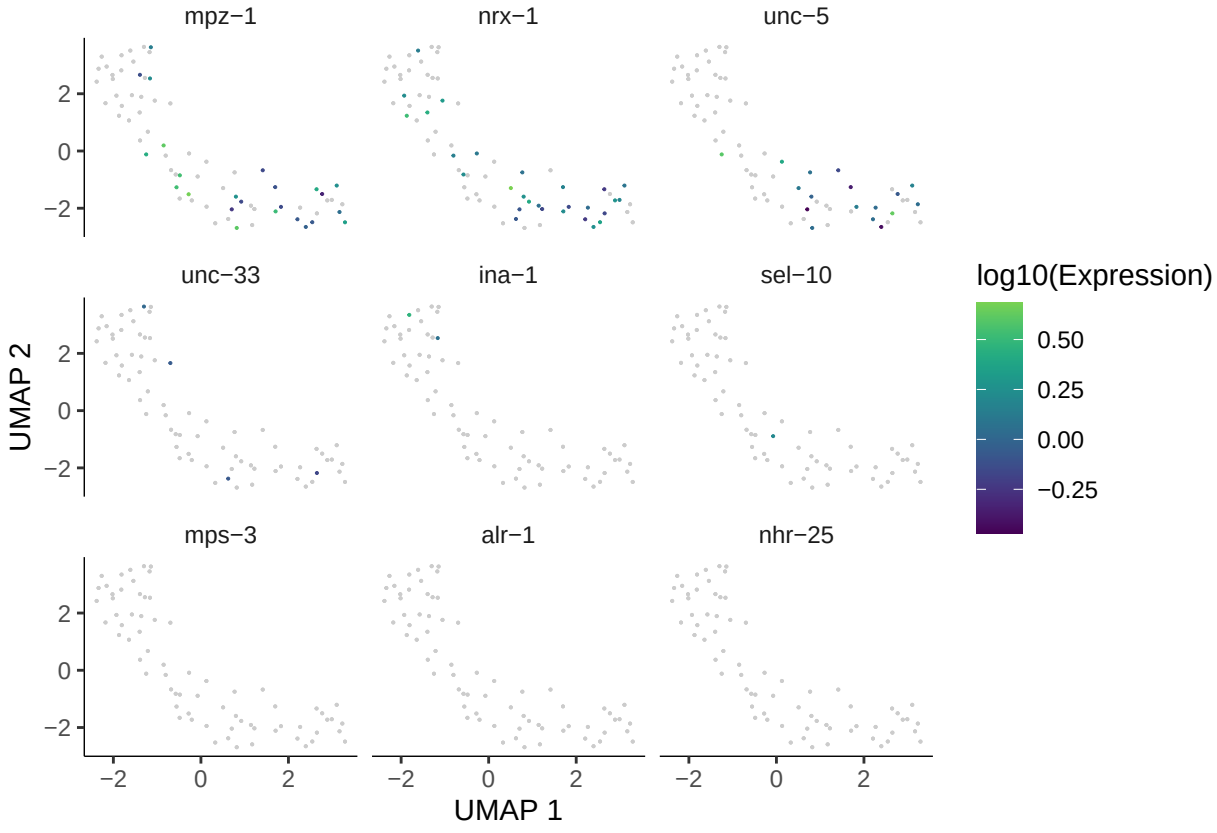
```
# cluster16.cellNames <- rownames(pData(cds))[pData(cds)$Cluster.IDs %in% c(16, 16.1)]

# cds_16 <- cds[,cluster16.cellNames]

cds_pg <- graph_test(cds, neighbor_graph="principal_graph", cores=4, verbose = F)

cds_genes <- cds_pg %>%
  filter(q_value < 0.05) %>%
  arrange(desc(morans_I)) %>%
  select(gene_short_name)
```

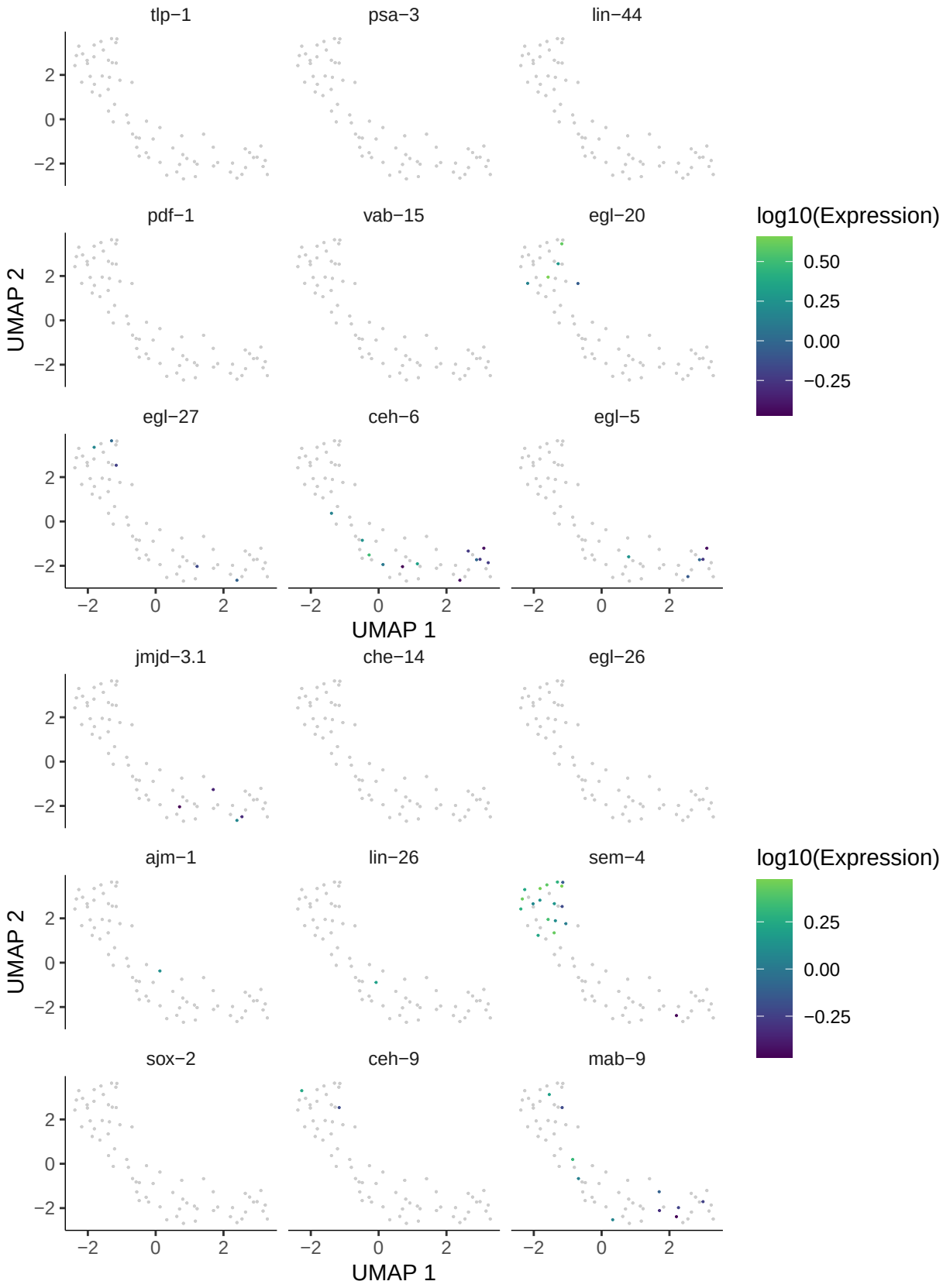
```
cds_genes$gene_short_name
```

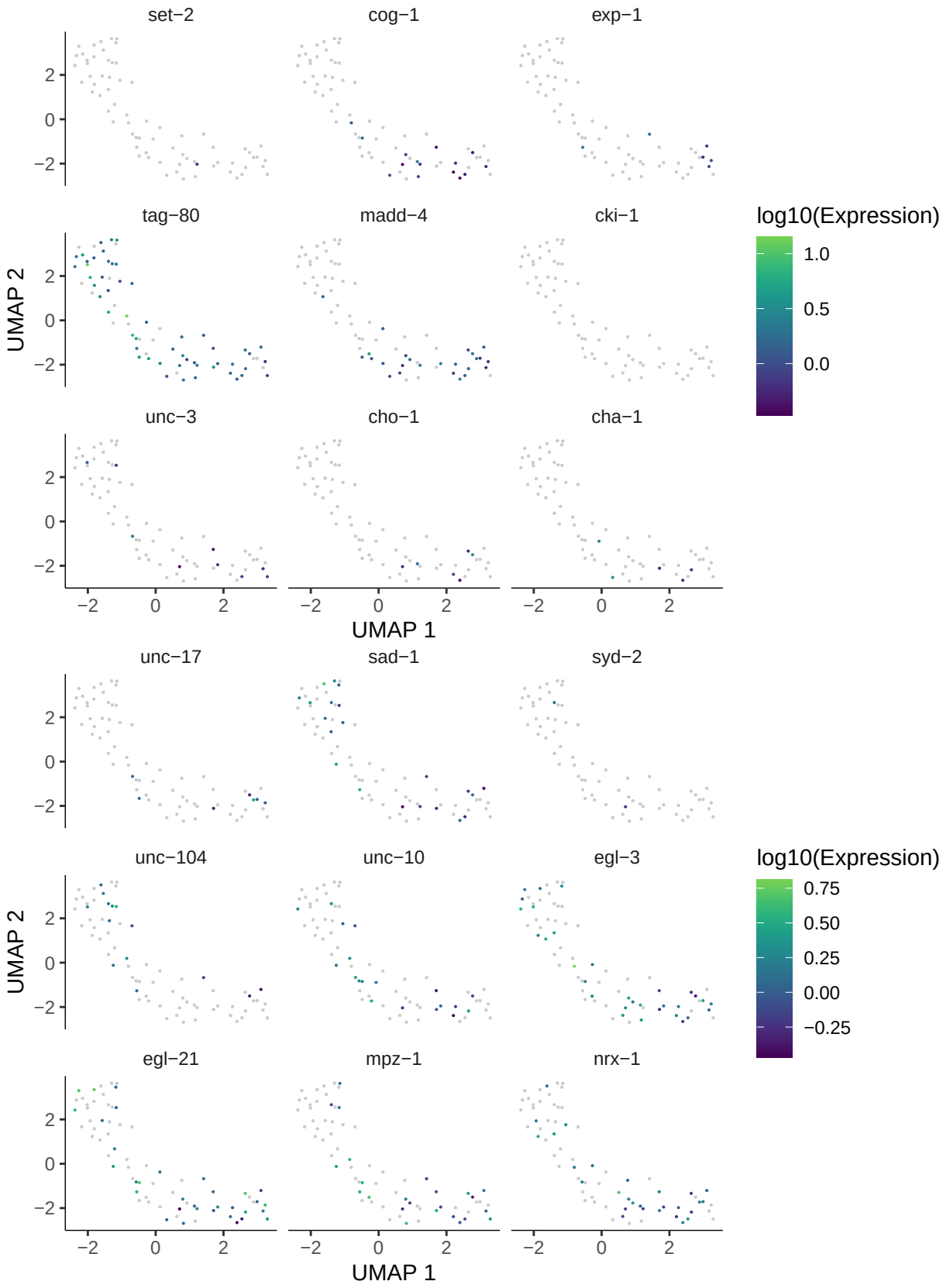
```
##   [1] sem-4      F54G2.1    ttn-1      lst-4      unc-68     madd-4
##   [7] rps-18     C06E7.2    gon-1      lim-7      nlp-13     egl-23
##  [13] unc-80     T24H7.2    ZC581.3    ZC506.1    Y44E3A.4   Y64G10A.7
##  [19] gbb-2      flp-6      cab-1      Y62E10A.13 cle-1      lgc-4
##  [25] apl-1      unc-64     sor-1      R05D7.3    slo-1      F44E2.3
##  [31] tag-196    des-2      rad-26     mel-46     gbb-1      lrp-2
##  [37] T19D12.6   lgc-31     arrd-6     ben-1      let-526    kin-2
##  [43] nrx-1      gly-12     F42H10.3   duxl-1     F46H5.4    ced-10
##  [49] twk-39     C11E4.6    M70.5      snb-1      spg-7      R173.3
##  [55] cog-1      gcy-28     slo-2      unc-53     F13E9.11   unc-5
##  [61] fat-1      aqp-7      dip-2      let-60     hum-5      rpl-36.A
```

```
## [67] C33D9.3    F08F8.10   F13C5.1
## 20271 Levels: 2L52.1 2RSSE.1 4R79.2 6R55.2 aagr-1 aagr-2 aagr-3 aagr-4 ... zyx-1
```

```
# Plot a few genes
list_genes <- unique(c("mpz-1","nrx-1", "unc-5", "unc-33","ina-1", "sel-10","mps-3","alr-1", "nhr-25",

nplots <- 9
x <- seq_along(list_genes)
toplot <- split(list_genes, ceiling(x/nplots))
fillplot <- nplots-length(toplot[[length(toplot)]])
toplot[[length(toplot)]] <- c(toplot[[length(toplot)]],toplot[[1]][1:fillplot])


for (x in toplot) {
  print(plot_cells(cds, genes= x,
             show_trajectory_graph=FALSE,
             label_cell_groups=FALSE,
             label_leaves=FALSE,
             cell_size = .4
            ))
}
```



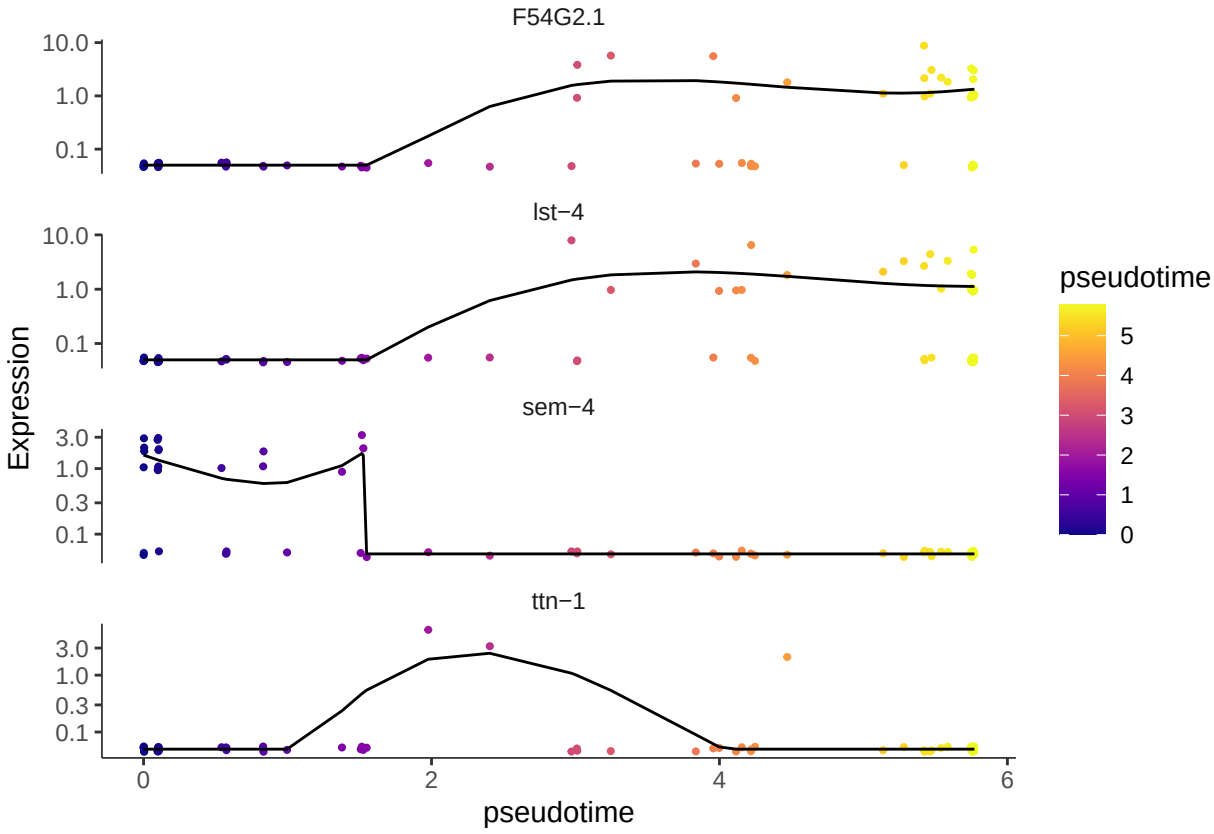

tlp−1
psa−3
lin−44
pdf−1
vab−15
egl−20
egl−27
ceh−6
egl−5
jmjd−3.1
che−14
egl−26
ajm−1
lin−26
sem−4
sox−2
ceh−9
mab−9
UMAP 2
UMAP 1
log10(Expression)
0.50
0.25
0.00
−0.25
log10(Expression)
0.25
0.00
−0.25

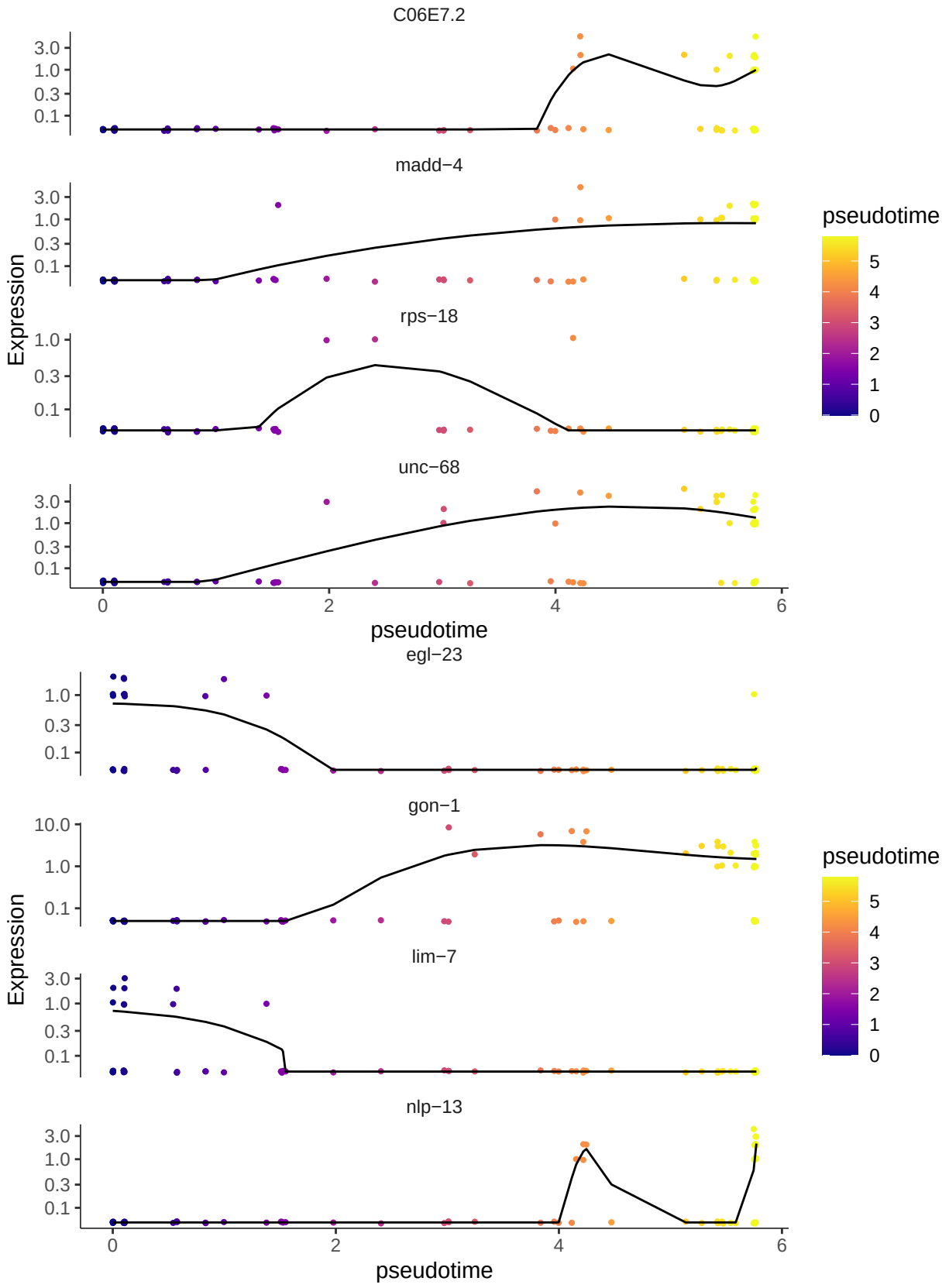

set−2
cog−1
exp−1
tag−80
madd−4
cki−1
unc−3
cho−1
cha−1
unc−17
sad−1
syd−2
unc−104
unc−10
egl−3
egl−21
mpz−1
nrx−1
UMAP 1
UMAP 2
log10(Expression)
1.0
0.5
0.0
0.75
0.50
0.25
0.00
−0.25

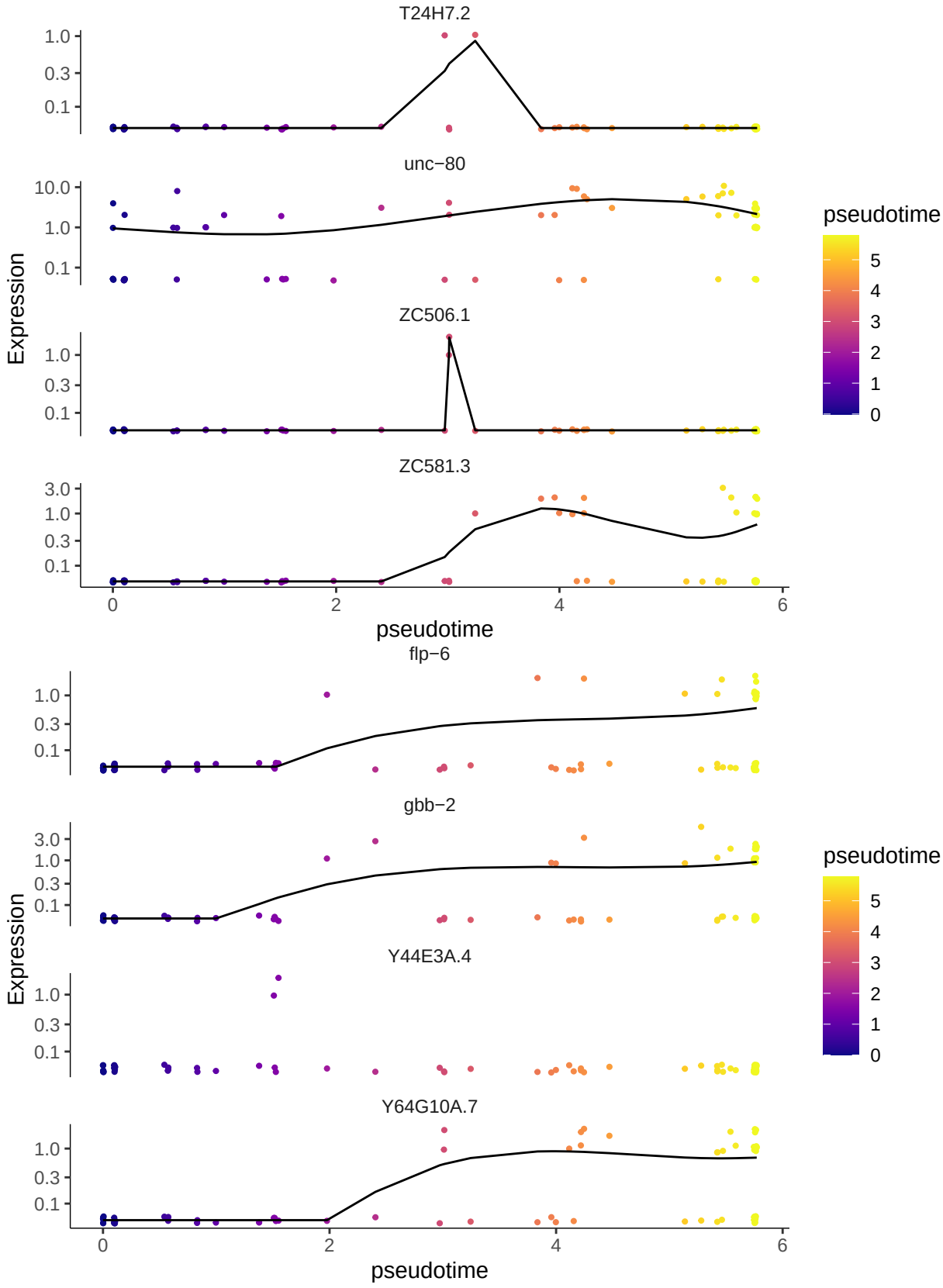
Pseudotemporal Expression Pattern

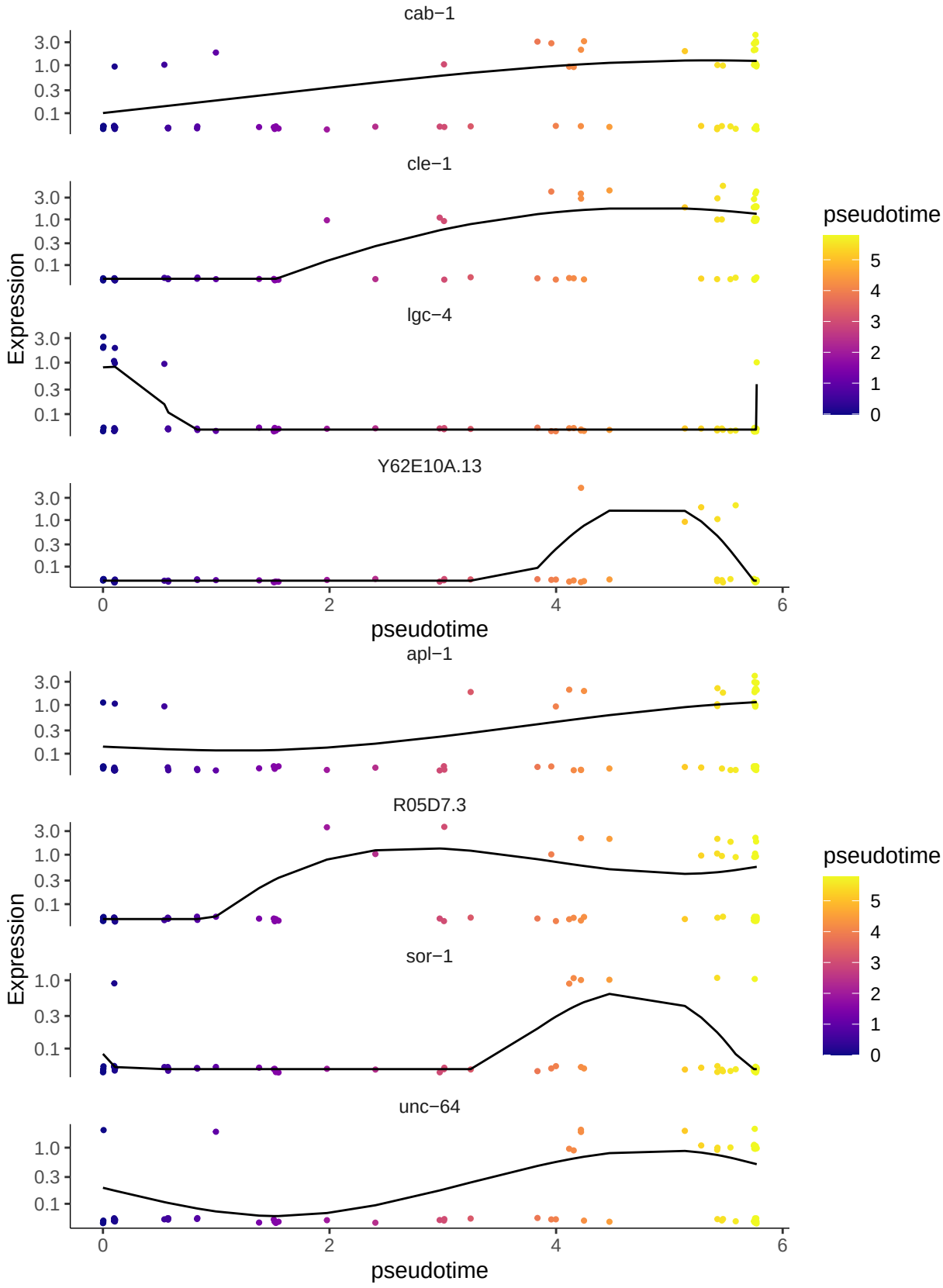
Diferentially expressed

```
nplots <- 4
x <- seq_along(as.character(cds_genes$gene_short_name))
toplot <- split(as.character(cds_genes$gene_short_name), ceiling(x/nplots))
fillplot <- nplots-length(toplot[[length(toplot)]])
toplot[[length(toplot)]] <- c(toplot[[length(toplot)]],toplot[[1]][1:fillplot])



for (x in toplot) {
  cds_lineage_cds <- cds[rowData(cds)$gene_short_name %in% x,]
  print(plot_genes_in_pseudotime(cds_lineage_cds,
                         # color_cells_by="embryo.time.bin",
                         min_expr=0.05))
}
```

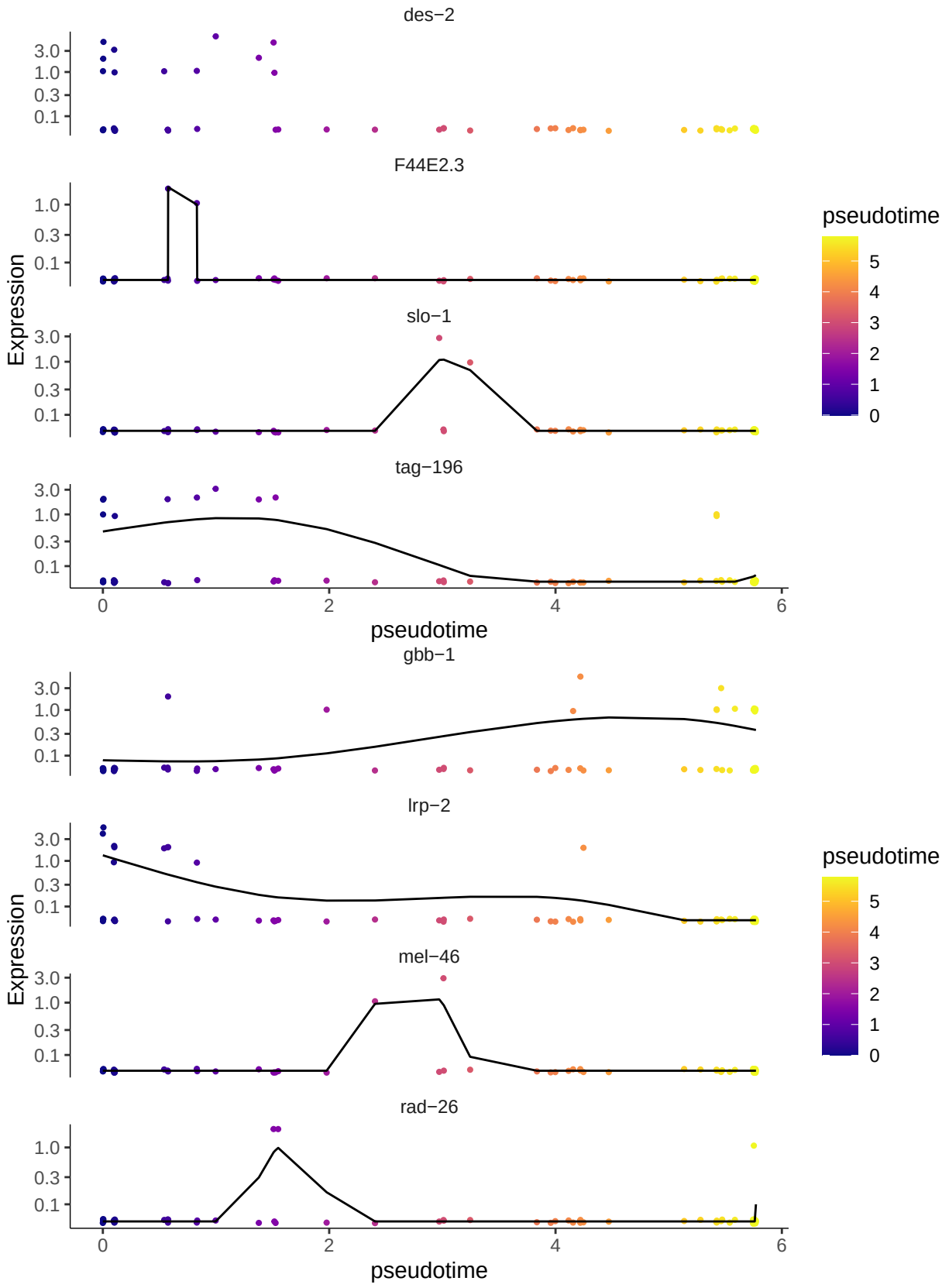


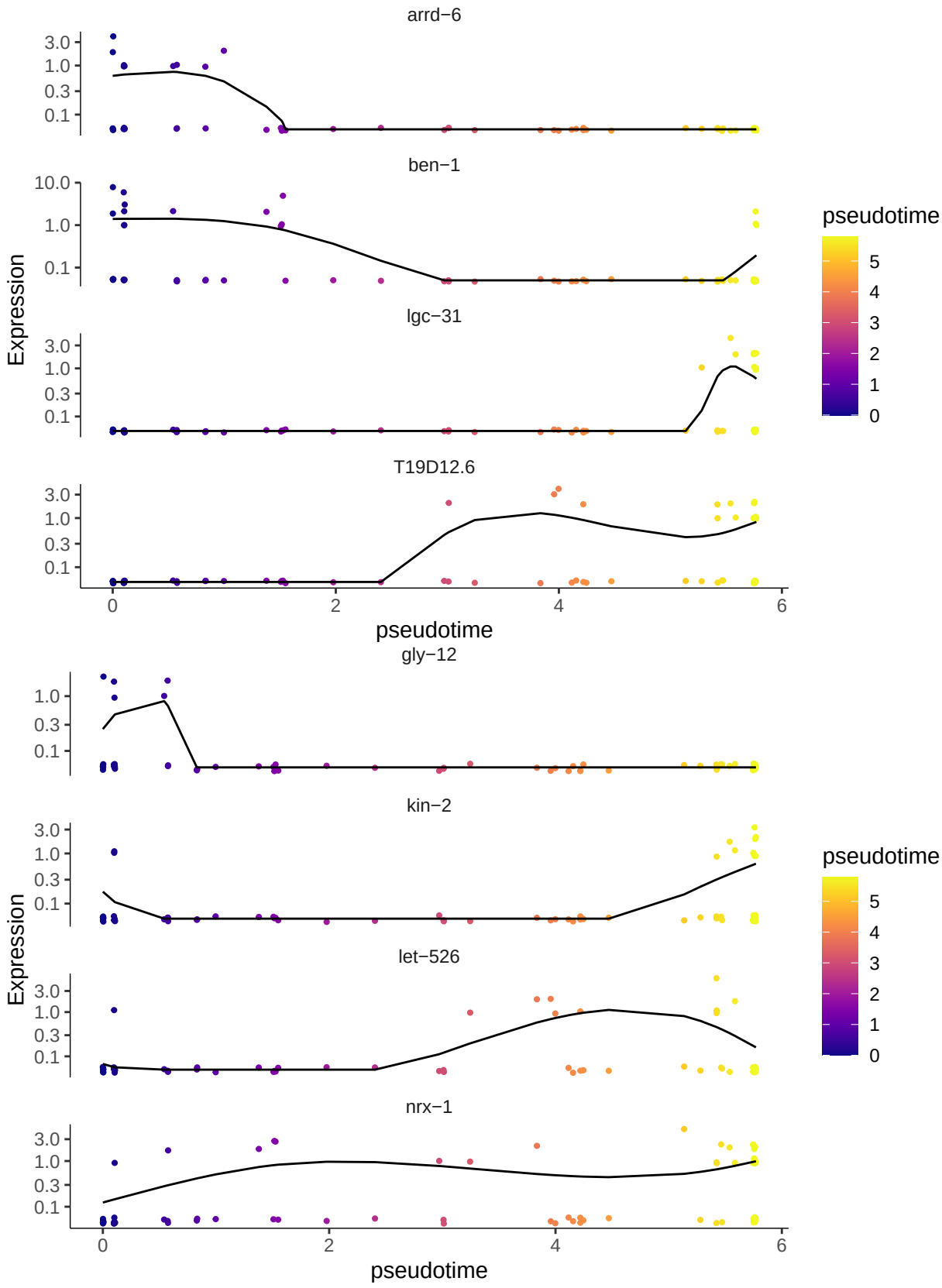

C06E7.2
madd−4
rps−18
unc−68
egl−23
gon−1
lim−7
nlp−13
Expression
pseudotime
pseudotime
3.0
1.0
0.3
0.1
10.0
0
2
4
6
5
4
3
2
1
0

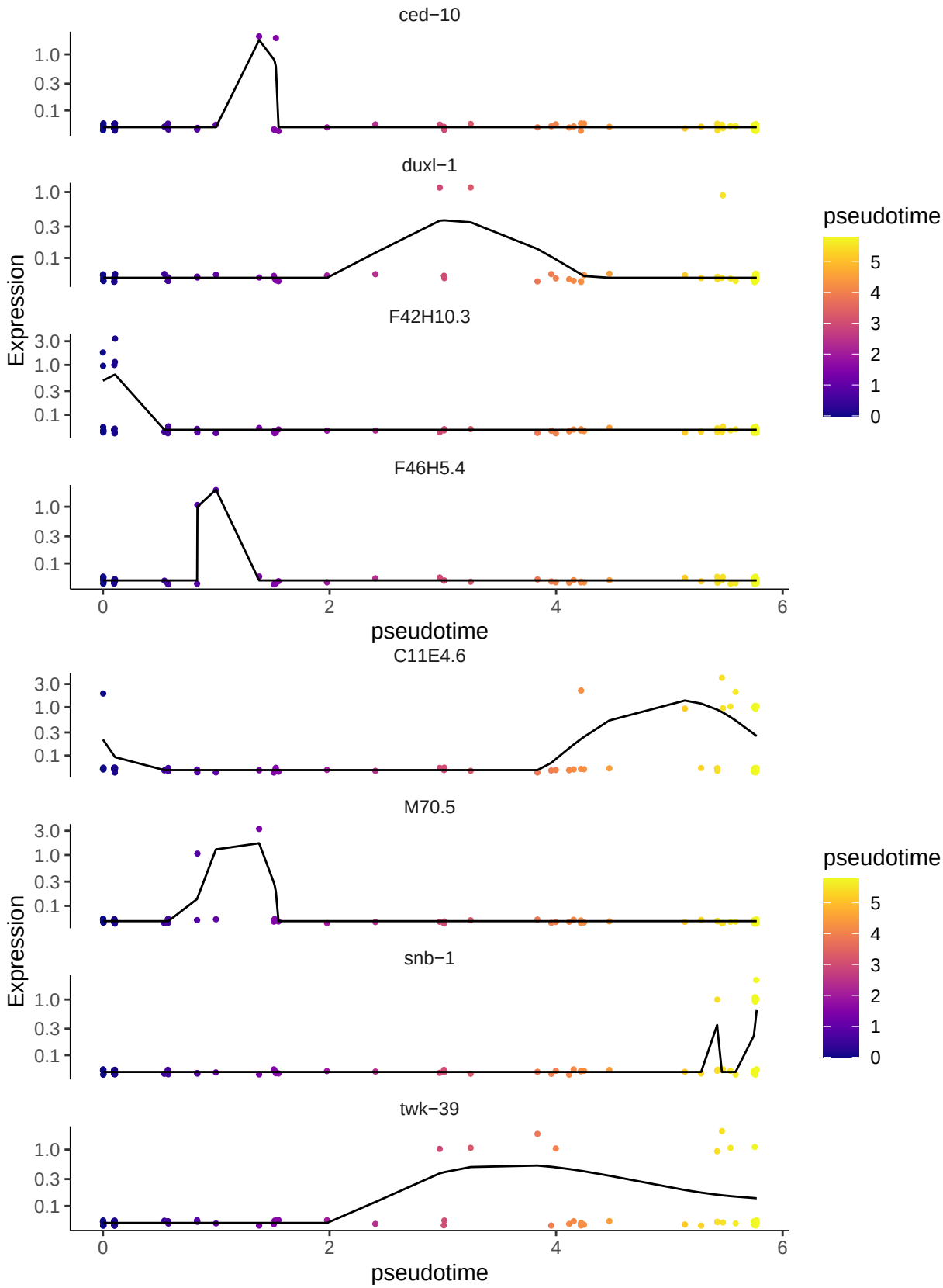

T24H7.2
unc−80
ZC506.1
ZC581.3
flp−6
gbb−2
Y44E3A.4
Y64G10A.7
Expression
pseudotime
pseudotime
pseudotime
pseudotime

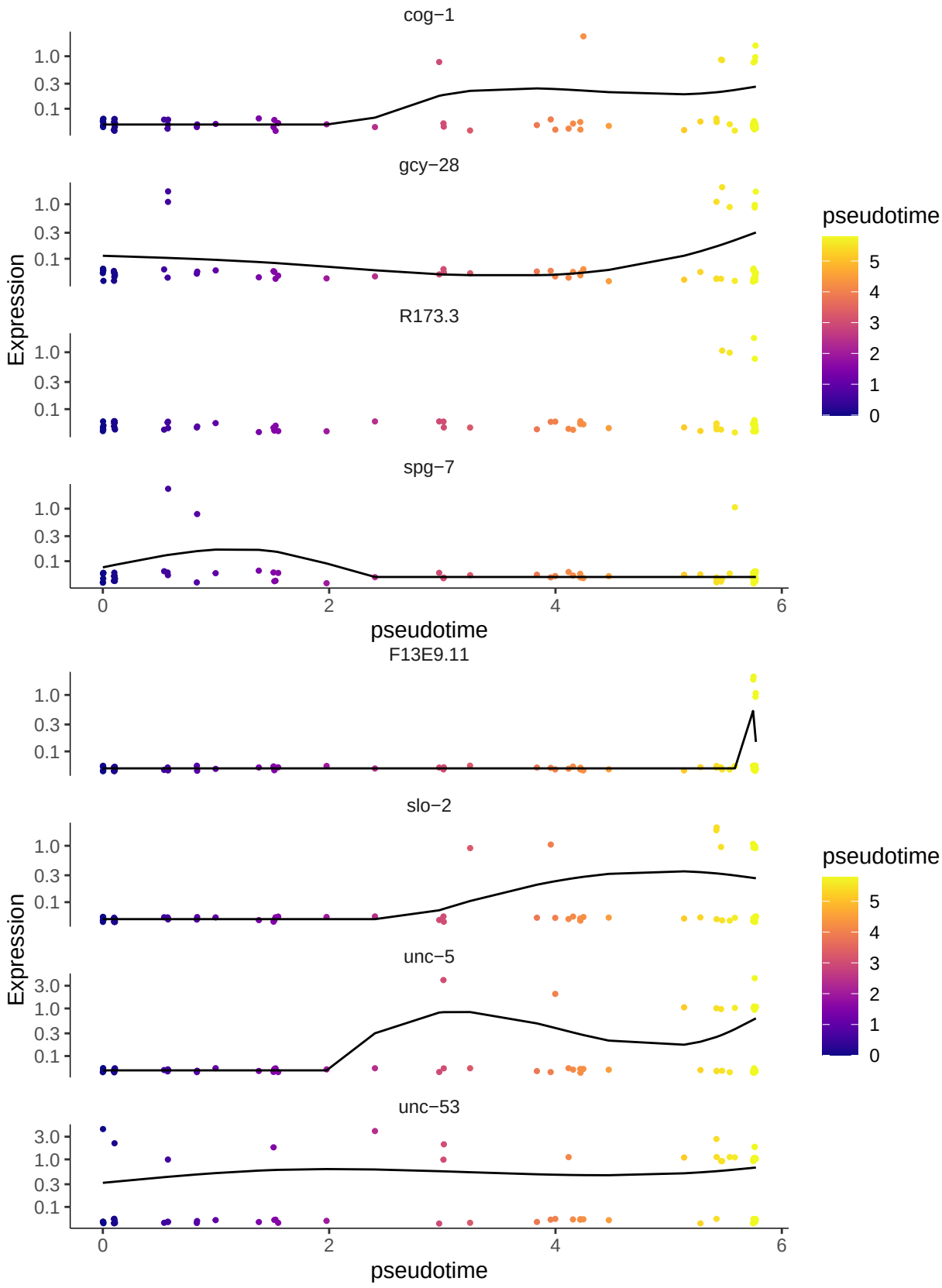


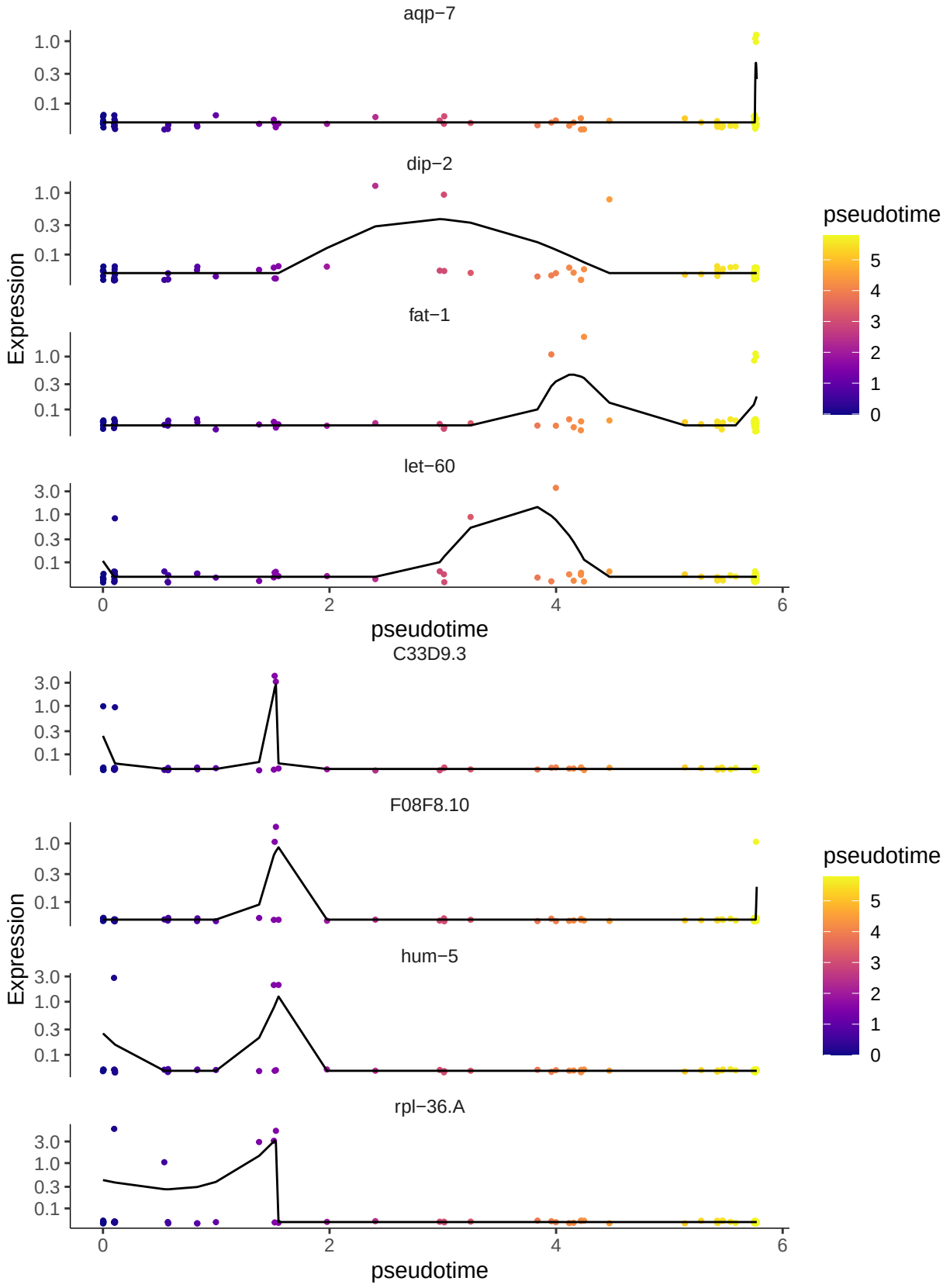

```
## Warning: Removed 70 row(s) containing missing values (geom_path).
```


des−2
F44E2.3
slo−1
tag−196
gbb−1
lrp−2
mel−46
rad−26
Expression
pseudotime
pseudotime
3.0
1.0
0.3
0.1
0
2
4
6
5
4
3
2
1
0

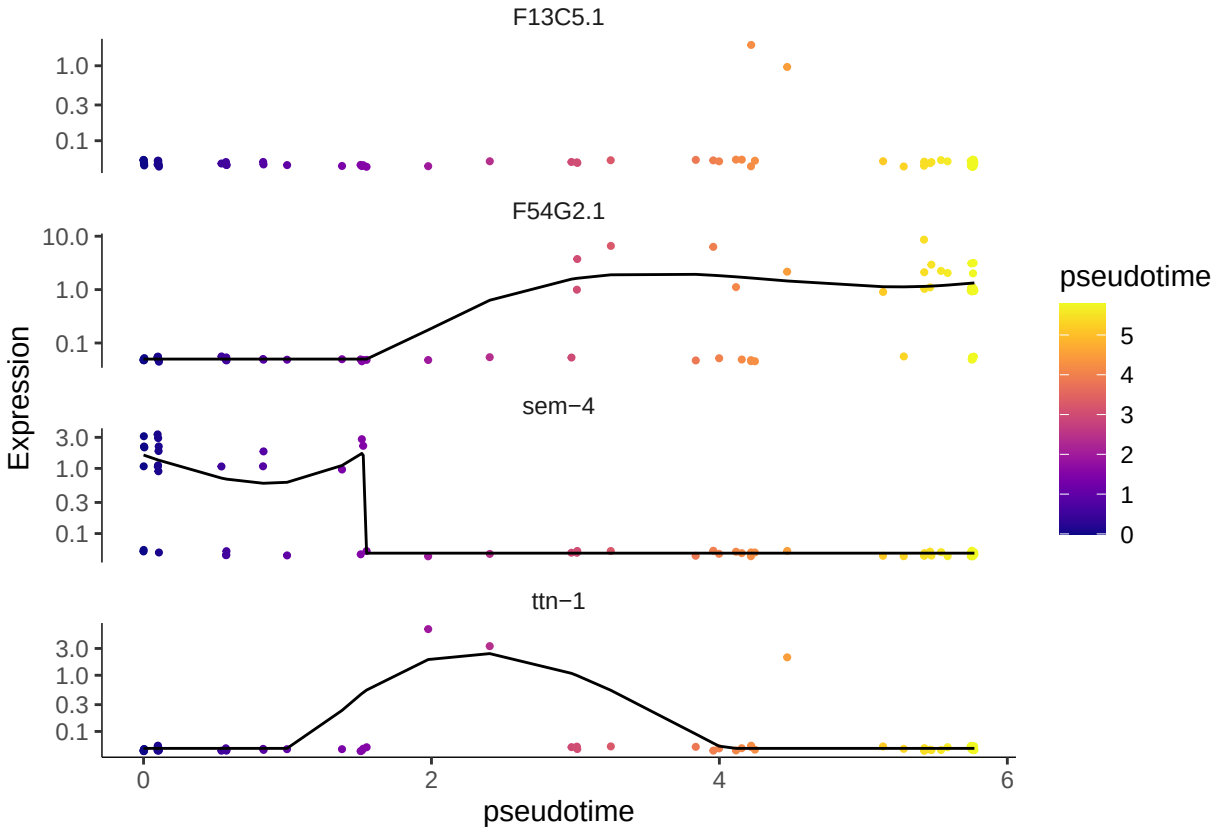

arrd−6
ben−1
lgc−31
T19D12.6
gly−12
kin−2
let−526
nrx−1
Expression
pseudotime
pseudotime
pseudotime
pseudotime
3.0
1.0
0.3
0.1
10.0
0
2
4
6
5
4
3
2
1
0


ced−10
duxl−1
F42H10.3
F46H5.4
C11E4.6
M70.5
snb−1
twk−39
Expression
pseudotime
pseudotime
1.0
0.3
0.1
3.0
0
2
4
6
5
4
3
2
1
0


cog−1
gcy−28
R173.3
spg−7
F13E9.11
slo−2
unc−5
unc−53
Expression
pseudotime
pseudotime
1.0
0.3
0.1
3.0
0
2
4
6
5
4
3
2
1
0

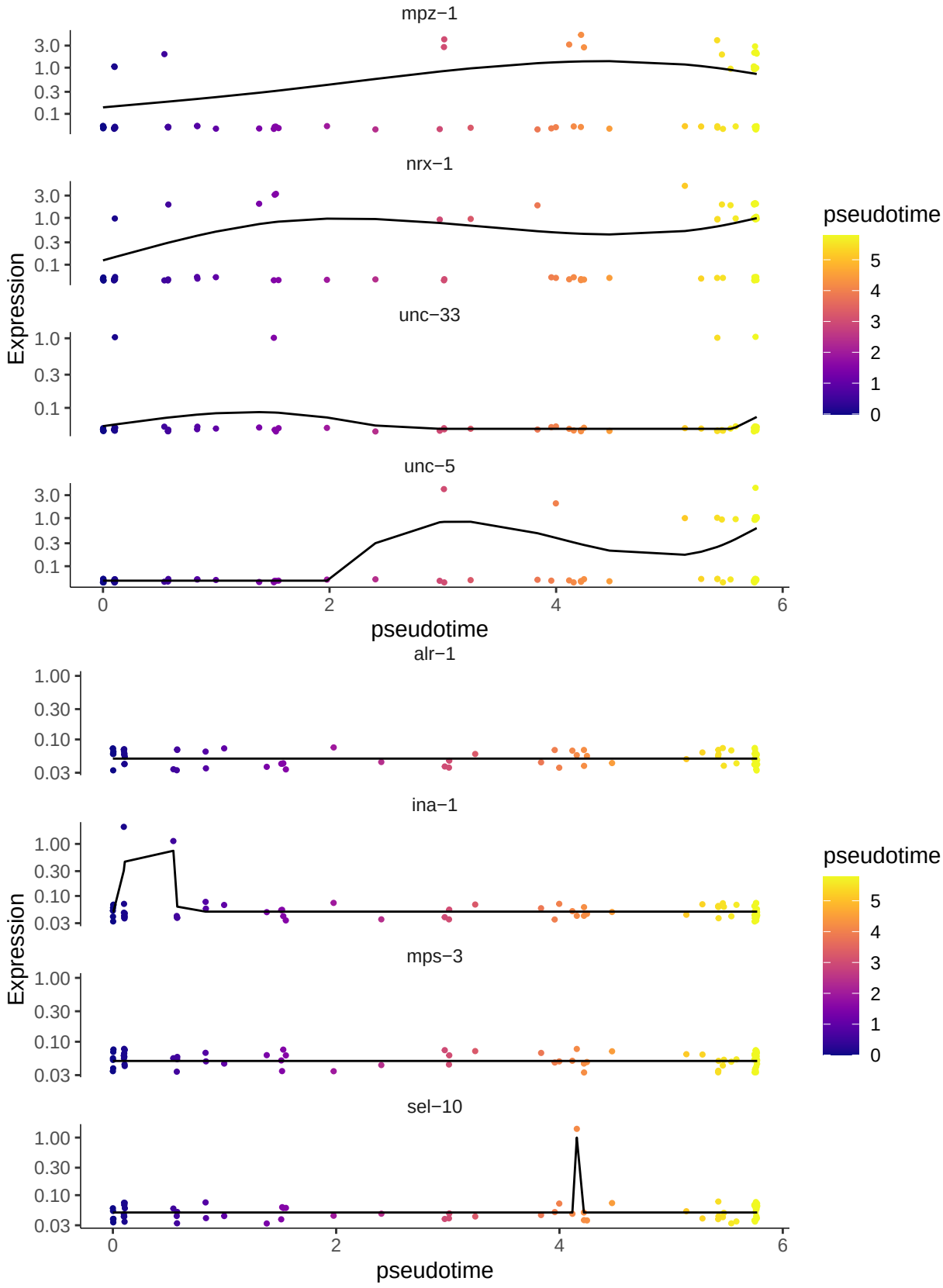


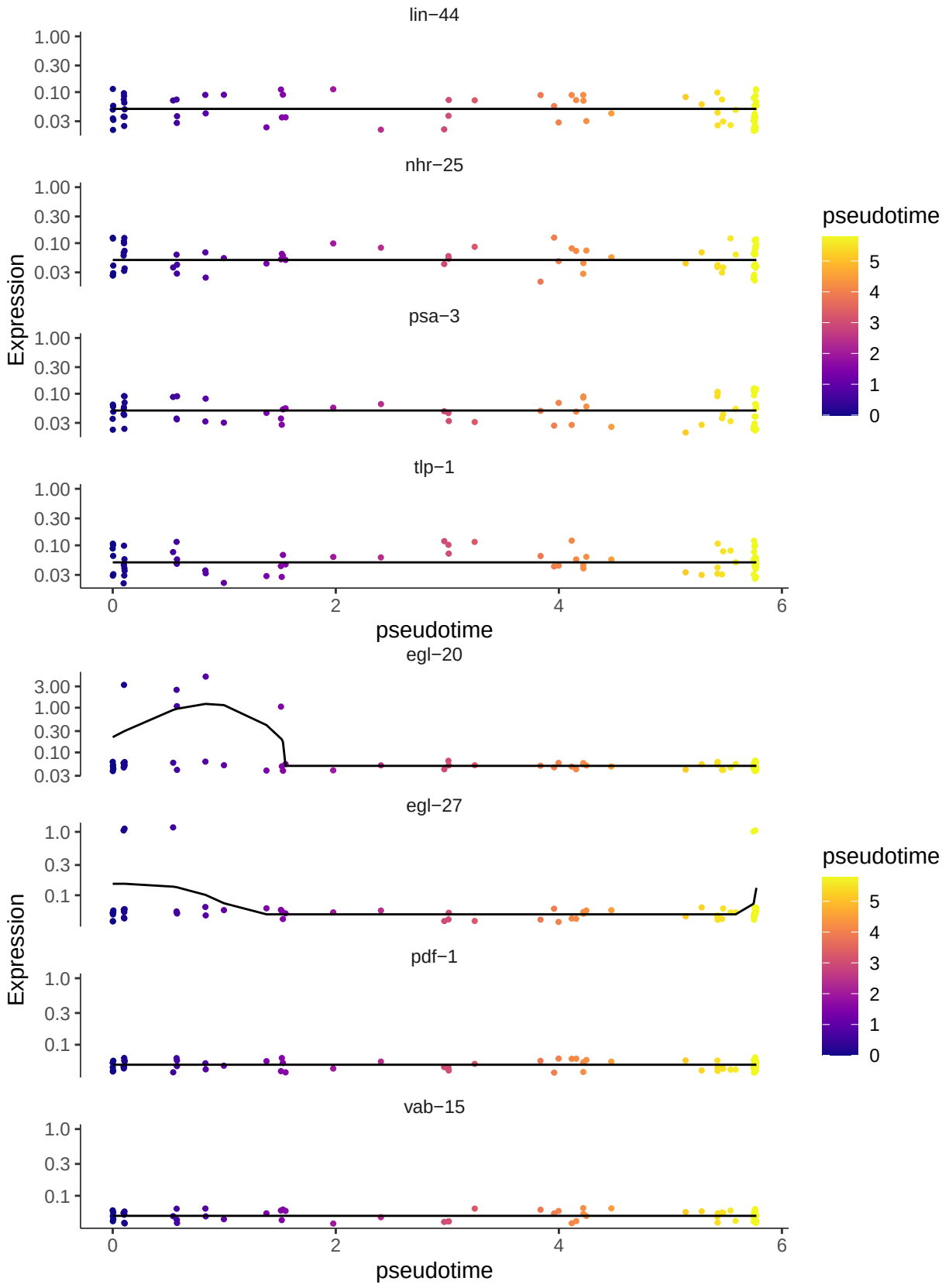

```
## Warning: Removed 70 row(s) containing missing values (geom_path).
```

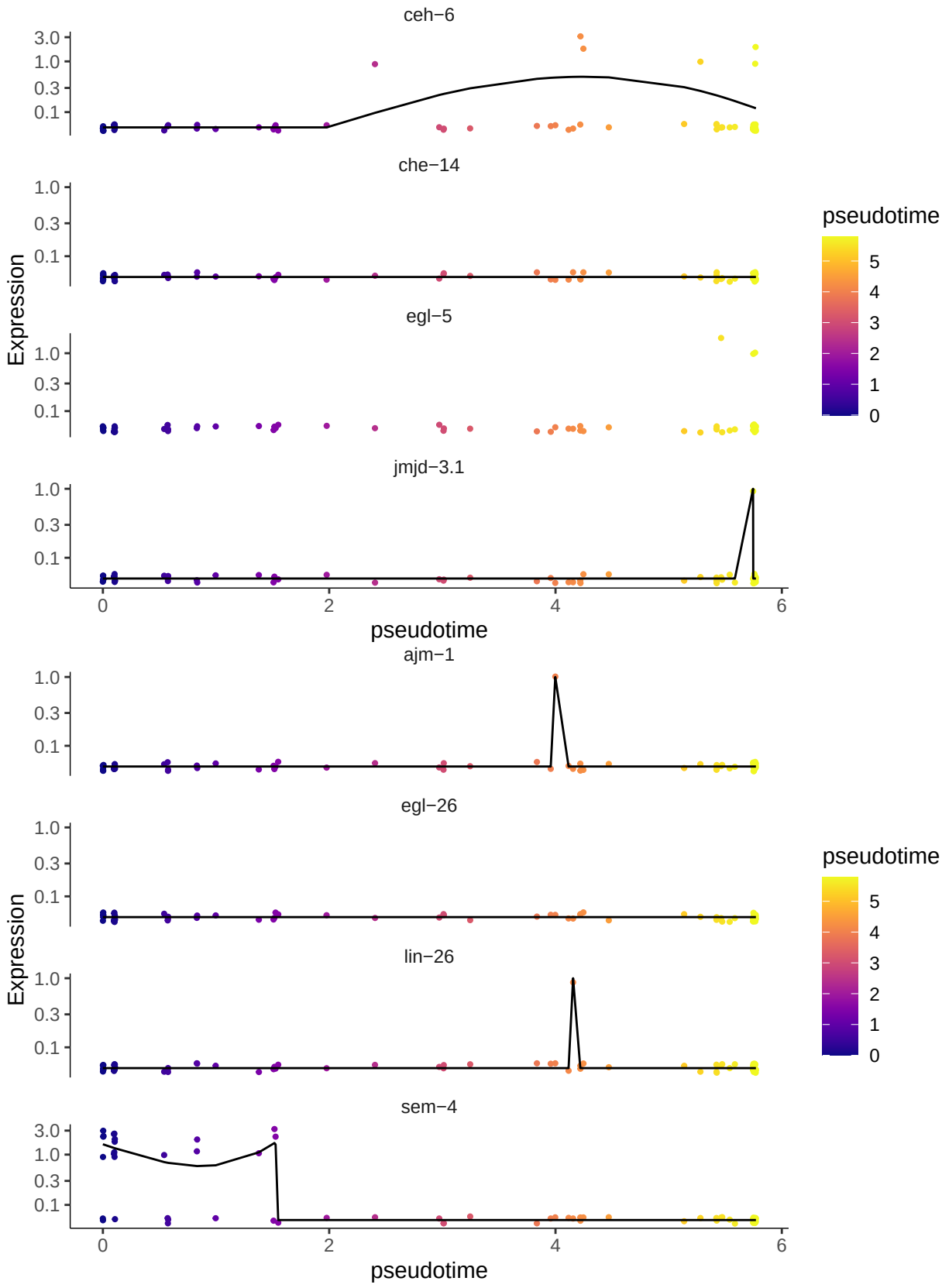


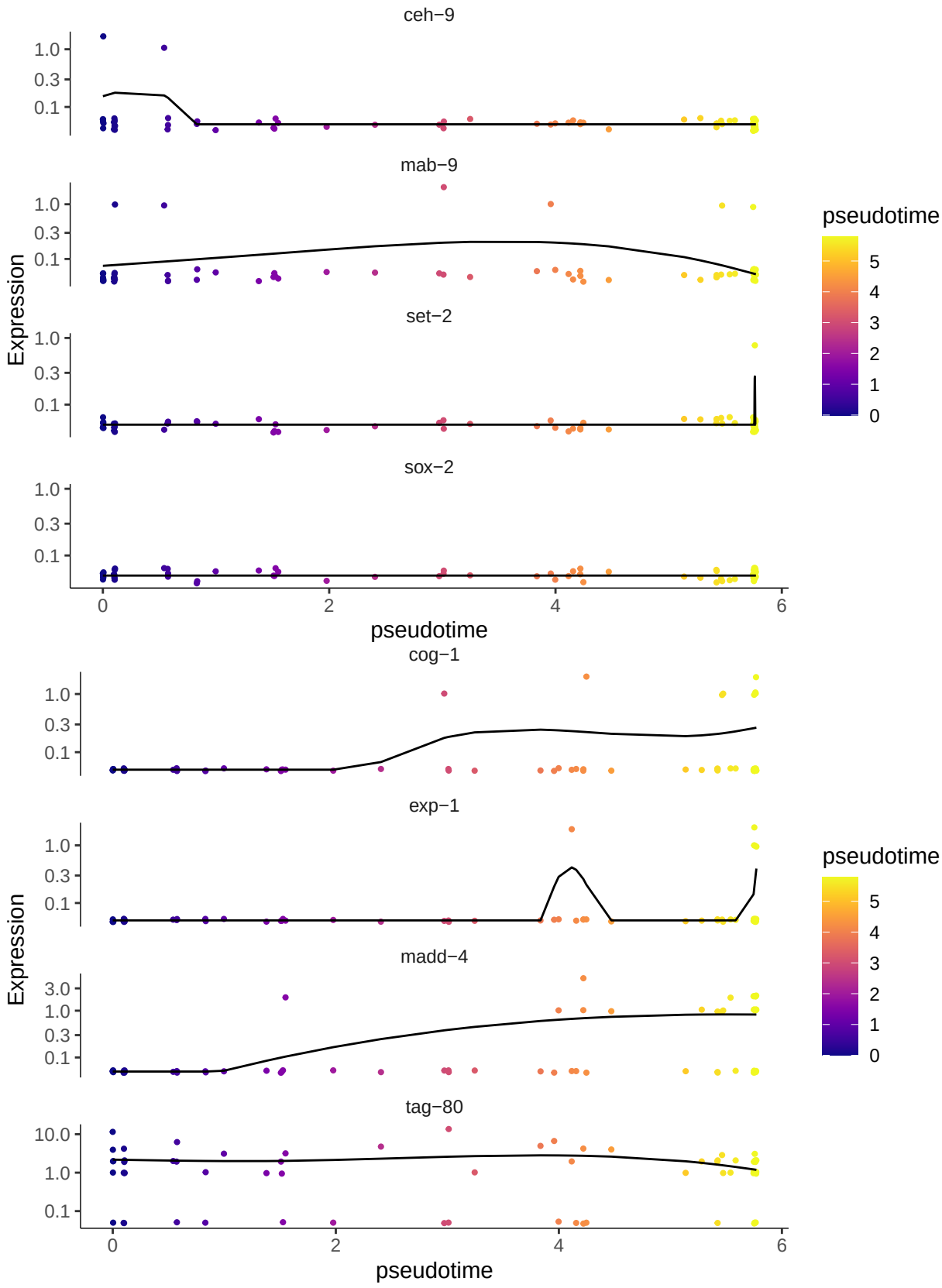
Selected genes

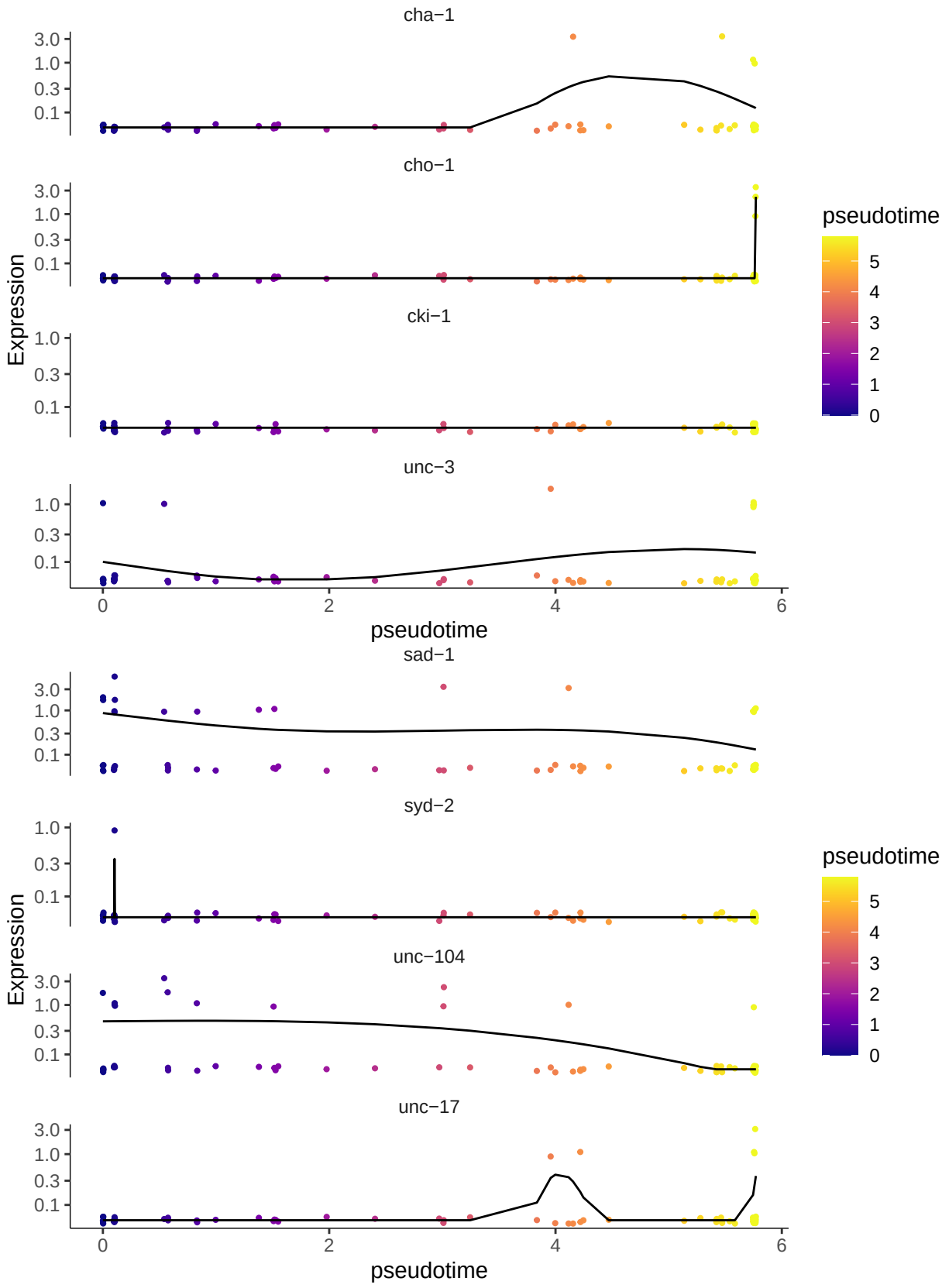
```
nplots <- 4
x <- seq_along(list_genes)
toplot <- split(list_genes, ceiling(x/nplots))
fillplot <- nplots-length(toplot[[length(toplot)]])
toplot[[length(toplot)]] <- c(toplot[[length(toplot)]],toplot[[1]][1:fillplot])



for (x in toplot) {
  cds_lineage_cds <- cds[rowData(cds)$gene_short_name %in% x,]
  print(plot_genes_in_pseudotime(cds_lineage_cds,
                         # color_cells_by="embryo.time.bin",
                         min_expr=0.05))
}
```

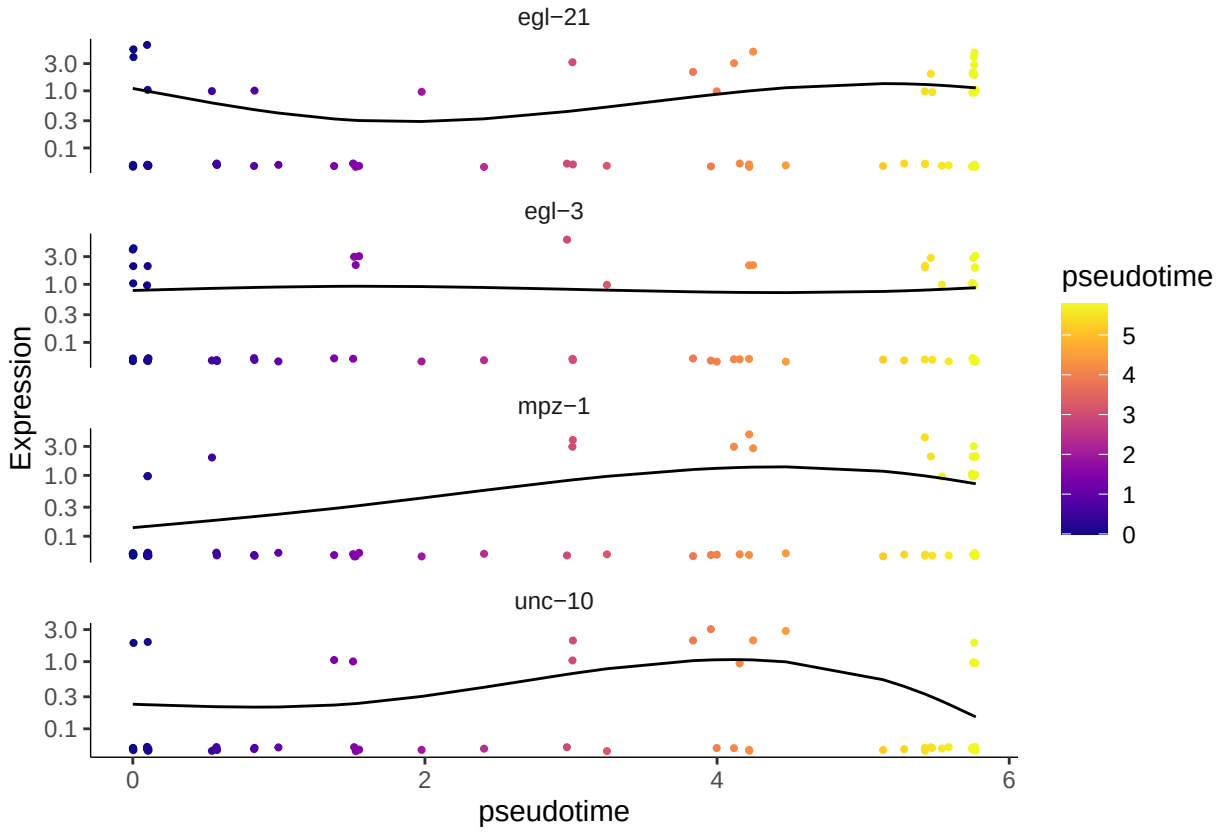

mpz−1
nrx−1
unc−33
unc−5
alr−1
ina−1
mps−3
sel−10
Expression
pseudotime
pseudotime
3.0
1.0
0.3
0.1
1.00
0.30
0.10
0.03
0
2
4
6
5
4
3
2
1
0


lin−44
nhr−25
psa−3
tlp−1
egl−20
egl−27
pdf−1
vab−15
Expression
pseudotime
pseudotime
1.00
0.30
0.10
0.03
3.00
1.0
0.3
0.1
0
2
4
6
5
4
3
2
1
0


ceh−6
che−14
egl−5
jmjd−3.1
ajm−1
egl−26
lin−26
sem−4
Expression
pseudotime
pseudotime
3.0
1.0
0.3
0.1
0
2
4
6
5
4
3
2
1
0


ceh−9
mab−9
set−2
sox−2
cog−1
exp−1
madd−4
tag−80
Expression
pseudotime
pseudotime
1.0
0.3
0.1
3.0
10.0
0
2
4
6
5
4
3
2
1
0


cha−1
cho−1
cki−1
unc−3
sad−1
syd−2
unc−104
unc−17
Expression
pseudotime
3.0
1.0
0.3
0.1
0
2
4
6
5
4
3
2
1
0


egl−21
egl−3
mpz−1
unc−10
Expression
3.0
1.0
0.3
0.1
0
2
4
6
pseudotime
pseudotime
5
4
3
2
1
0