# eCB expression analysis of Hypothalamus development with focus on PVN

Evgenii O. Tretiakov

2024-11-25

## Setup parameters

```r
# Load tidyverse infrastructure packages
suppressPackageStartupMessages({
  library(future)
  library(here)
  library(tidyverse)
  library(magrittr)
  library(stringr)
  library(skimr)
  library(RColorBrewer)
  library(viridis)
})


# Load packages for scRNA-seq analysis and visualisation
suppressPackageStartupMessages({
  library(UpSetR)
  library(ggplot2)
  library(cowplot)
  library(patchwork)
  library(ggstatsplot)
  library(anndata)
```

```
  library(sceasy)
  library(Seurat)
  library(SeuratDisk)
  library(SeuratWrappers)
  library(scCustomize)
})

sc <- import("scanpy", convert = FALSE)
```

### Set paths

```
src_dir <- here("code")
data_dir <- here("data")
output_dir <- here("output")
plots_dir <- here(output_dir, "figures/")
tables_dir <- here(output_dir, "tables/")
```

### Load helper functions and gene-sets

```
source(here(src_dir, "genes.R"))
source(here(src_dir, "functions.R"))
```

### Set fixed variables

```
# set seed
reseed <- 42
set.seed(seed = reseed)

# Parameters for parallel execution
n_cores <- 8
plan("multisession", workers = n_cores)
options(
  future.globals.maxSize = 100000 * 1024^2,
  future.rng.onMisuse = "ignore"
)
```

```r
plan()
```

```
multisession:
- args: function (..., workers = 8, envir = parent.frame())
- tweaked: TRUE
- call: plan("multisession", workers = n_cores)
```

```r
# ggplot2 theme
theme_set(ggmin::theme_powerpoint())
```

```r
bioproject <- "PRJNA547712"
project <- "kim2020_Hypoth-dev"
cb_fpr <- 0.001
low_cutoff_gene <- 500
high_cutoff_gene <- NULL
high_cutoff_gene <- 5000
low_cutoff_umis <- NULL
low_cutoff_umis <- -Inf
high_cutoff_umis <- 25000
high_cutoff_pc_mt <- 15
high_cutoff_pc_ribo <- 20
high_cutoff_pc_hb <- 0.1
high_cutoff_doublet_score <- 0.33
high_cutoff_complexity <- 0.85
connectivity_model <- "min_tree"
k <- 10
metric <- "euclidean"
signature <- 100
```

## Load Kim DW et al 2020

```r
anndata <- sc$read(here(
  "kim2020_combined.h5ad"
))
```

### Convert adata object to R AnnDataR6 object.

```
adata <- py_to_r(anndata)
class(adata)
```

```
[1] "AnnDataR6" "R6"
```

```
class(adata$X)
```

```
[1] "dgRMatrix"
attr(,"package")
[1] "Matrix"
```

```
adata
```

```
AnnData object with n_obs × n_vars = 128006 × 27998
    obs: 'bc_name', 'orig.ident', 'nCount_RNA', 'nFeature_RNA', 'X1', 'X', 'Y', 'Z', '
    var: '_index', 'features'
```

```
srt_path <- here(
  "data",
  sprintf("%s-whole_dataset-simple.h5Seurat", bioproject)
)

expr_mtx <- t(as.matrix(adata$X))
colnames(expr_mtx) <- rownames(adata$X)
rownames(expr_mtx) <- adata$var$features
srt <- CreateSeuratObject(
  expr_mtx,
  assay = "RNA",
  project = "kim2020_Hypoth_dev",
  meta.data = as.data.frame(adata$obs)
)

X_umap <- adata$obs |>
  select(X, Y) |>
```

```
  as.matrix()
colnames(X_umap) <- c("UMAP_1", "UMAP_2")
rownames(X_umap) <- colnames(expr_mtx)
srt[["umap"]] <- CreateDimReducObject(embeddings = X_umap, key = "umap_", assay = De

Idents(srt) <- "age"
srt <- Store_Palette_Seurat(seurat_object = srt, palette = rev(brewer.pal(n = 11, na
```

## Load Romanov et al 2020

```
print(srt)
```

```
An object of class Seurat
27998 features across 128006 samples within 1 assay
Active assay: RNA (27998 features, 0 variable features)
 1 layer present: counts
 1 dimensional reduction calculated: umap
```

```
rar2020.srt.pub <- readRDS("/data/1_heteroAstrocytes/PRJNA548917/old/oldCCA_nae_srt.
rar2020.srt.pub <- UpdateSeuratObject(rar2020.srt.pub)
Idents(rar2020.srt.pub) <-
  factor(rar2020.srt.pub$wtree,
    ordered = TRUE
  )

# Consistent colours and clusters names
colours_wtree <- setNames(read_lines(here(data_dir, "colours_wtree.tsv")), 1:45)

rar2020.srt.pub$age <-
  Cells(rar2020.srt.pub) |>
  str_split(pattern = ":", simplify = T) %>%
  .[, 1] %>%
  str_split_fixed(pattern = "_", n = 3) %>%
  .[, 3]
print(rar2020.srt.pub)
```

```
An object of class Seurat
24340 features across 51199 samples within 1 assay
Active assay: RNA (24340 features, 3500 variable features)
 3 layers present: counts, data, scale.data
 3 dimensional reductions calculated: pca, tsne, umap
```

```
glimpse(rar2020.srt.pub@meta.data)
```

```
Rows: 51,199
Columns: 20
$ nGene            <int> 1652, 782, 447, 1706, 1106, 894, 727, 734, 669, 617, ~
$ nUMI             <dbl> 2787, 1090, 544, 2709, 1817, 1220, 995, 1036, 920, 86~
$ orig.ident       <fct> Hypothalamus, Hypothalamus, Hypothalamus, Hypothalamu~
$ res.0.2          <chr> "23", "23", "23", "23", "23", "23", "23", "23", "23",~
$ res.0.4          <chr> "34", "34", "34", "34", "34", "34", "34", "34", "34",~
$ res.0.8          <chr> "42", "42", "42", "42", "42", "42", "42", "42", "42",~
$ res.1.2          <chr> "47", "47", "47", "47", "47", "47", "47", "47", "47",~
$ res.2            <chr> "54", "54", "54", "54", "54", "54", "54", "54", "54",~
$ tree.ident       <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
$ pro_Inter        <chr> "41", "41", "41", "41", "41", "41", "41", "41", "41",~
$ pro_Enter        <chr> "41", "41", "41", "41", "41", "41", "41", "41", "41",~
$ tree_final       <fct> 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 1~
$ subtree          <fct> 41, 41, 41, 41, 41, 41, 41, 41, 41, 41, 41, 41, 41, 4~
$ prim_walktrap    <fct> 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 3~
$ umi_per_gene     <dbl> 1.687046, 1.393862, 1.217002, 1.587925, 1.642857, 1.3~
$ log_umi_per_gene <dbl> 0.22712693, 0.14421974, 0.08529138, 0.20082998, 0.215~
$ nCount_RNA       <dbl> 2787, 1090, 544, 2709, 1817, 1220, 995, 1036, 920, 86~
$ nFeature_RNA     <int> 1652, 782, 447, 1706, 1106, 894, 727, 734, 669, 617, ~
$ wtree            <fct> 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 3~
$ age              <chr> "P23", "3P2", "3P2", "P2", "P2", "P2", "P2", "P2", "P~
```

```
table(Idents(rar2020.srt.pub))
```

```

    1     2     3     4     5     6     7     8     9    10    11    12    13 
 2344  8146   395   402  3234   712   552   374   259   952 13727  1615   765 
   14    15    16    17    18    19    20    21    22    23    24    25    26 
  832  1244   792   590   808  2486  1683   628  1039  1750   292   394   547 
```

```
   27    28    29    30    31    32    33    34    35    36    37    38    39 
  391   407   507    93    81   402   143   701   222   353   324    73    78 
   40    41    42    43    44    45 
  328   190    73    37   179    55 
```

```r
rar2020.srt.pub %<>% RenameIdents(object = ., `43` = "mneOXY")
rar2020.srt.pub %<>% RenameIdents(object = ., `26` = "mneVAS")
rar2020.srt.pub %<>% RenameIdents(object = ., `31` = "pneSS")
rar2020.srt.pub %<>% RenameIdents(object = ., `24` = "pneCRH")
rar2020.srt.pub %<>% RenameIdents(object = ., `15` = "pneTRH")
```

```r
rar2020.srt.pub$stage <-
  rar2020.srt.pub$age %>%
  forcats::fct_collapse(
    Embryonic = c("E15", "E17"),
    Neonatal = c("P0", "P2", "3P2"),
    Pubertal = c("1P10", "P10"),
    Adult = c("P23")
  )
rar2020.srt.pub$stage %<>% factor(levels = c("Embryonic", "Neonatal", "Pubertal", "A
rar2020.srt.pub$stage %>% forcats::fct_count()
```

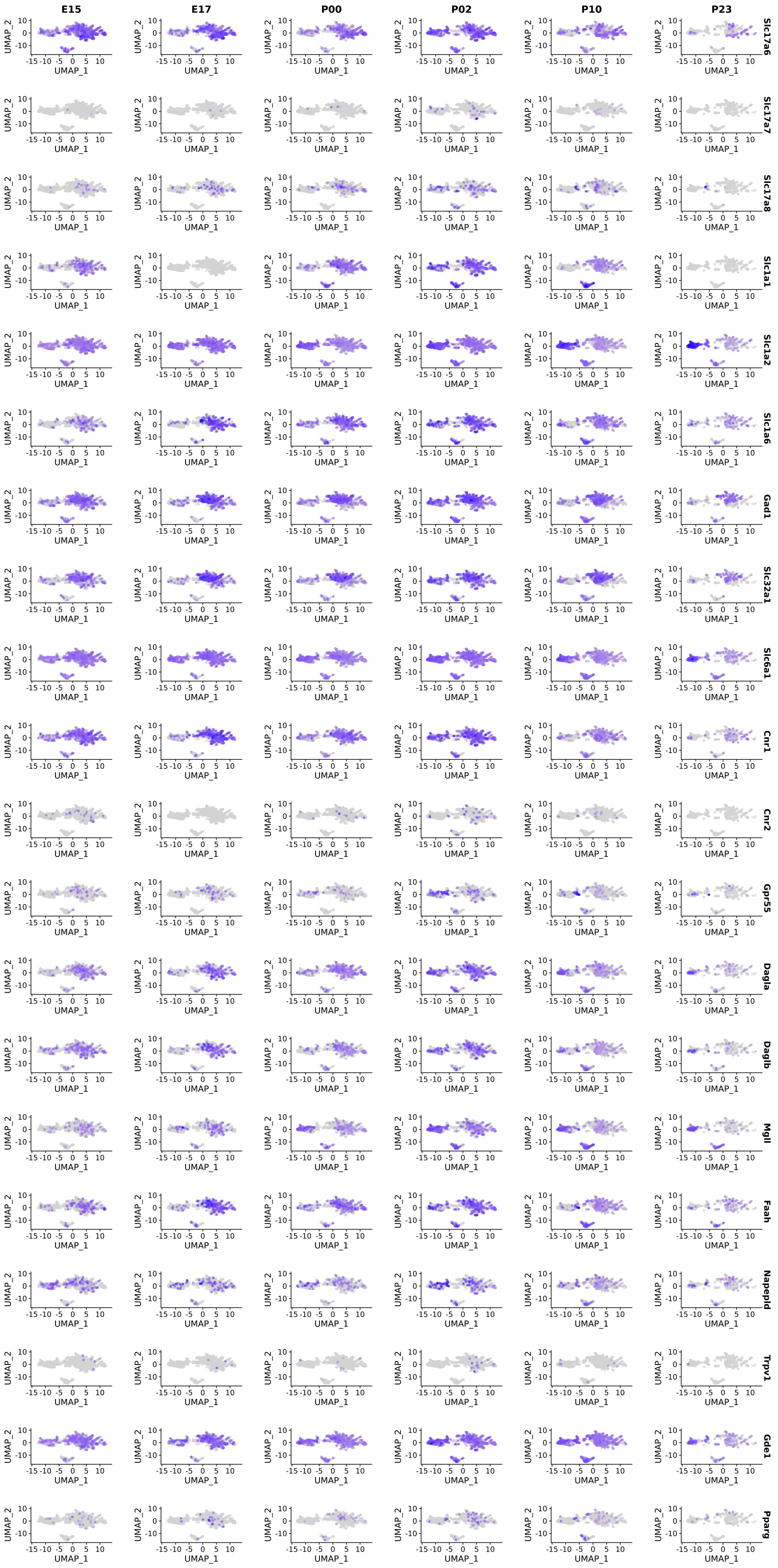
```
# A tibble: 4 x 2
  f             n
  <fct>     <int>
1 Embryonic 19503
2 Neonatal  20316
3 Pubertal   8965
4 Adult      2415
```

```r
rar2020.srt.pub$age <-
  plyr::mapvalues(
    x = rar2020.srt.pub$age,
    from = c("E15", "E17", "P0", "P2", "3P2", "1P10", "P10", "P23"),
    to = c("E15", "E17", "P00", "P02", "P02", "P10", "P10", "P23")
  )
```

```
rar2020.srt.pub$age %>% forcats::fct_count()
```

```
# A tibble: 6 x 2
  f         n
  <fct> <int>
1 E15    8290
2 E17   11213
3 P00    7492
4 P02   12824
5 P10    8965
6 P23    2415
```

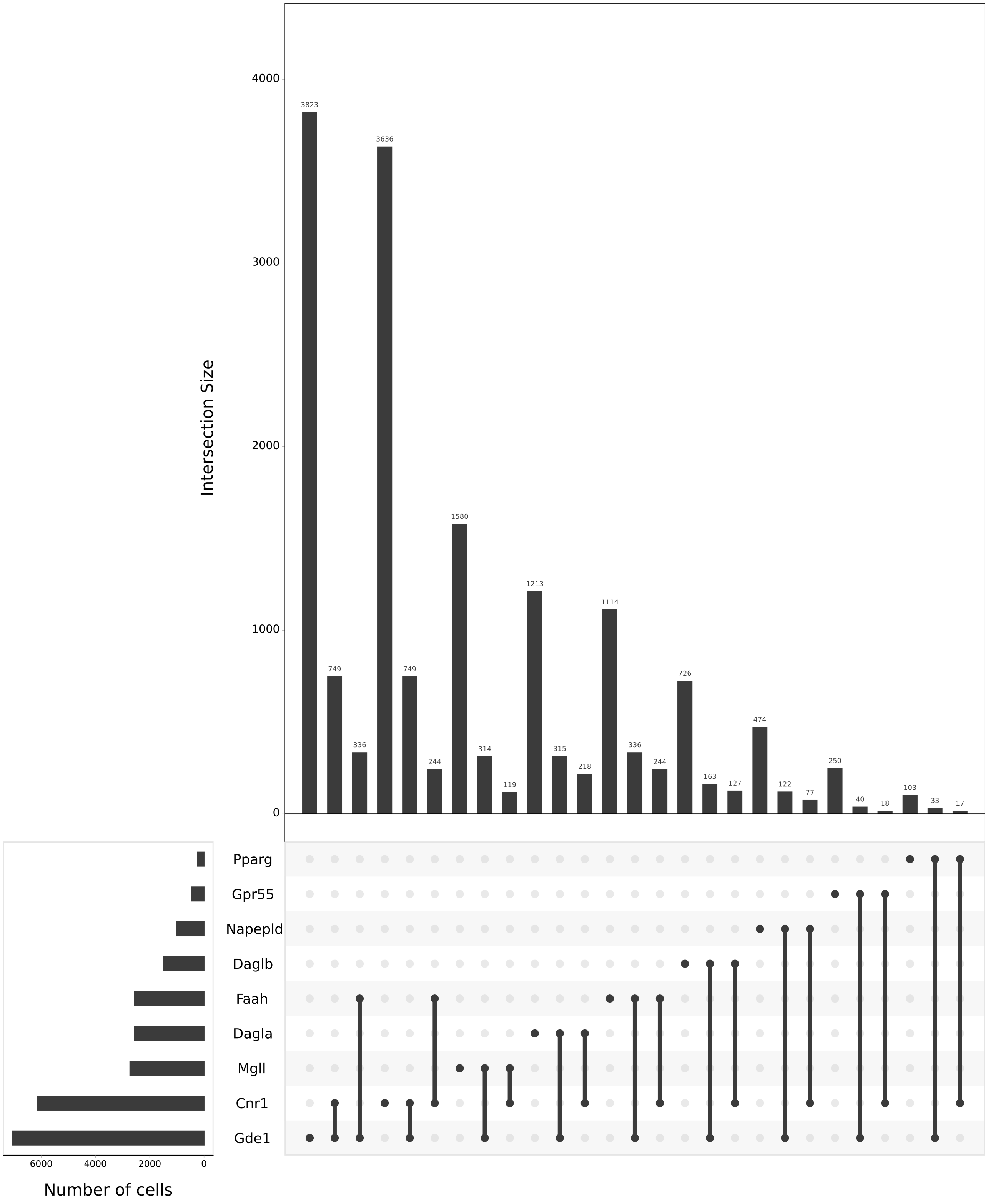
```
FeaturePlot(
  rar2020.srt.pub,
  features = c(neurotrans, cnbn),
  label = F,
  blend = F,
  order = TRUE,
  pt.size = 1.2,
  raster.dpi = c(1024, 1024),
  alpha = 0.5,
  split.by = "age"
)
```


E15
E17
P00
P02
P10
P23
Slc17a6
Slc17a7
Slc17a8
Slc1a1
Slc1a2
Slc1a6
Gad1
Slc32a1
Slc6a1
Cnr1
Cnr2
Gpr55
Dagla
Daglb
Mgll
Faah
Napepld
Trpv1
Gde1
Pparg
UMAP_2
10
0
-10
UMAP_1
-15 -10 -5 0 5 10

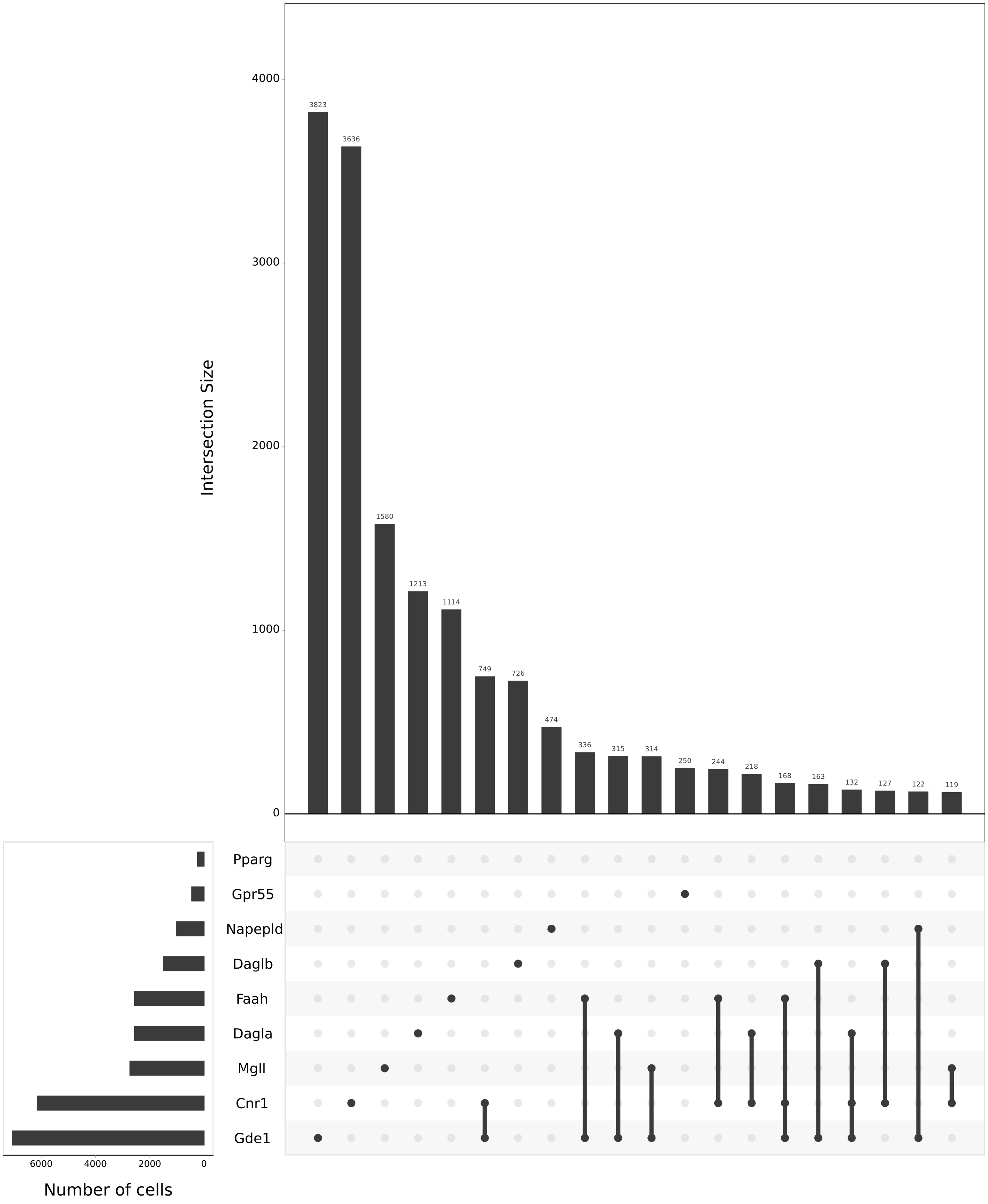
```r
sbs_mtx <-
  rar2020.srt.pub@assays$RNA@data %>%
  as.data.frame() %>%
  t()
rownames(sbs_mtx) <- colnames(rar2020.srt.pub)

# Filter features
filt_low_genes <-
  colSums(sbs_mtx) %>%
  .[. > quantile(., 0.4)] %>%
  names()
sbs_mtx %<>% .[, filt_low_genes]

min_filt_vector2 <-
  sbs_mtx %>%
  as_tibble() %>%
  select(all_of(filt_low_genes)) %>%
  summarise(across(.fns = ~ quantile(.x, .005))) %>%
  as.list() %>%
  map(as.double) %>%
  simplify() %>%
  .[filt_low_genes]

# Prepare table of intersection sets analysis
content_sbs_mtx <-
  (sbs_mtx > min_filt_vector2) %>%
  as_tibble() %>%
  mutate_all(as.numeric)
```

```r
upset(
  as.data.frame(content_sbs_mtx),
  order.by = "freq",
  group.by = "sets",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
```

```
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
4000
3000
2000
1000
0
3823
749
336
3636
749
244
1580
314
119
1213
315
218
1114
336
244
726
163
127
474
122
77
250
40
18
103
33
17
Pparg
Gpr55
Napepld
Daglb
Faah
Dagla
Mgll
Cnr1
Gde1
6000
4000
2000
0
Number of cells

```
upset(
  as.data.frame(content_sbs_mtx),
  order.by = "freq",
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 15,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx)],
  nintersects = 20,
  empty.intersections = NULL
)
```


Intersection Size
4000
3000
2000
1000
0
3823
3636
1580
1213
1114
749
726
474
336
315
314
250
244
218
168
163
132
127
122
119
Pparg
Gpr55
Napepld
Daglb
Faah
Dagla
Mgll
Cnr1
Gde1
6000
4000
2000
0
Number of cells

```
sbs_mtx_full <- content_sbs_mtx |>
  select(any_of(c(neurotrans, cnbn))) |>
  dplyr::bind_cols(rar2020.srt.pub@meta.data)

sbs_mtx_full |> glimpse()
```

```
Rows: 51,199
Columns: 38
$ Slc17a6          <dbl> 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Slc17a8          <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Slc1a1           <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Slc1a2           <dbl> 0, 1, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Slc1a6           <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Gad1             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0,~
$ Slc32a1          <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Slc6a1           <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Cnr1             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Gpr55            <dbl> 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Dagla            <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Daglb            <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Mgll             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Faah             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Napepld          <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ Gde1             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,~
$ Pparg            <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
$ nGene            <int> 1652, 782, 447, 1706, 1106, 894, 727, 734, 669, 617, ~
$ nUMI             <dbl> 2787, 1090, 544, 2709, 1817, 1220, 995, 1036, 920, 86~
$ orig.ident       <fct> Hypothalamus, Hypothalamus, Hypothalamus, Hypothalamu~
$ res.0.2          <chr> "23", "23", "23", "23", "23", "23", "23", "23", "23",~
$ res.0.4          <chr> "34", "34", "34", "34", "34", "34", "34", "34", "34",~
$ res.0.8          <chr> "42", "42", "42", "42", "42", "42", "42", "42", "42",~
$ res.1.2          <chr> "47", "47", "47", "47", "47", "47", "47", "47", "47",~
$ res.2            <chr> "54", "54", "54", "54", "54", "54", "54", "54", "54",~
$ tree.ident       <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
$ pro_Inter        <chr> "41", "41", "41", "41", "41", "41", "41", "41", "41",~
$ pro_Enter        <chr> "41", "41", "41", "41", "41", "41", "41", "41", "41",~
$ tree_final       <fct> 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 1~
$ subtree          <fct> 41, 41, 41, 41, 41, 41, 41, 41, 41, 41, 41, 41, 41, 4~
$ prim_walktrap    <fct> 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 3~
$ umi_per_gene     <dbl> 1.687046, 1.393862, 1.217002, 1.587925, 1.642857, 1.3~
```

```
$ log_umi_per_gene <dbl> 0.22712693, 0.14421974, 0.08529138, 0.20082998, 0.215~
$ nCount_RNA       <dbl> 2787, 1090, 544, 2709, 1817, 1220, 995, 1036, 920, 86~
$ nFeature_RNA     <int> 1652, 782, 447, 1706, 1106, 894, 727, 734, 669, 617, ~
$ wtree            <fct> 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 38, 3~
$ age              <chr> "P23", "P02", "P02", "P02", "P02", "P02", "P02", "P02~
$ stage            <ord> Adult, Neonatal, Neonatal, Neonatal, Neonatal, Neonat~
```

## Prepare query mapping between datasets

```
srt <- NormalizeData(srt)
srt <- FindVariableFeatures(srt, selection.method = "vst", nfeatures = 3000)
# all.genes <- rownames(srt)
# srt <- ScaleData(srt, features = all.genes)
srt <- ScaleData(srt)
```

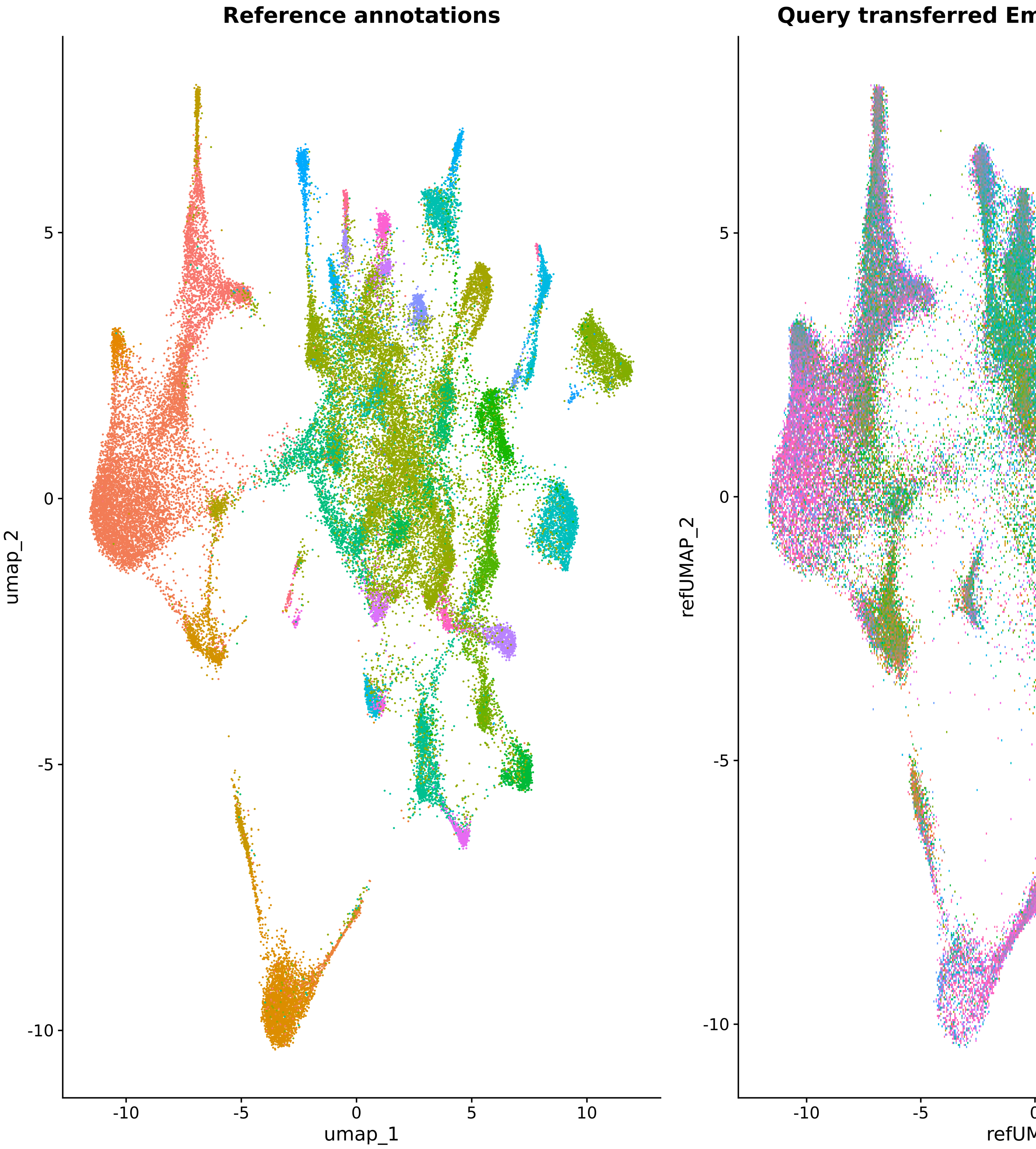
```
hypoth.anchors <- FindTransferAnchors(
  reference = rar2020.srt.pub, query = srt, dims = 1:30,
  reference.reduction = "pca"
)
predictions <- TransferData(anchorset = hypoth.anchors, refdata = rar2020.srt.pub$wt
srt <- AddMetaData(srt, metadata = predictions)
table(srt$predicted.id)
```

```
   1     10     11     12     13     15     16     17     19      2     20
3700      1   2198    225    545     58    218     19 100308   9886    924
  22     23     26     27     28     29      3     38      4      5      6
 163     11    115     24     76    481    951     49    156   2895   4756
   7      8      9
 199     47      1
```

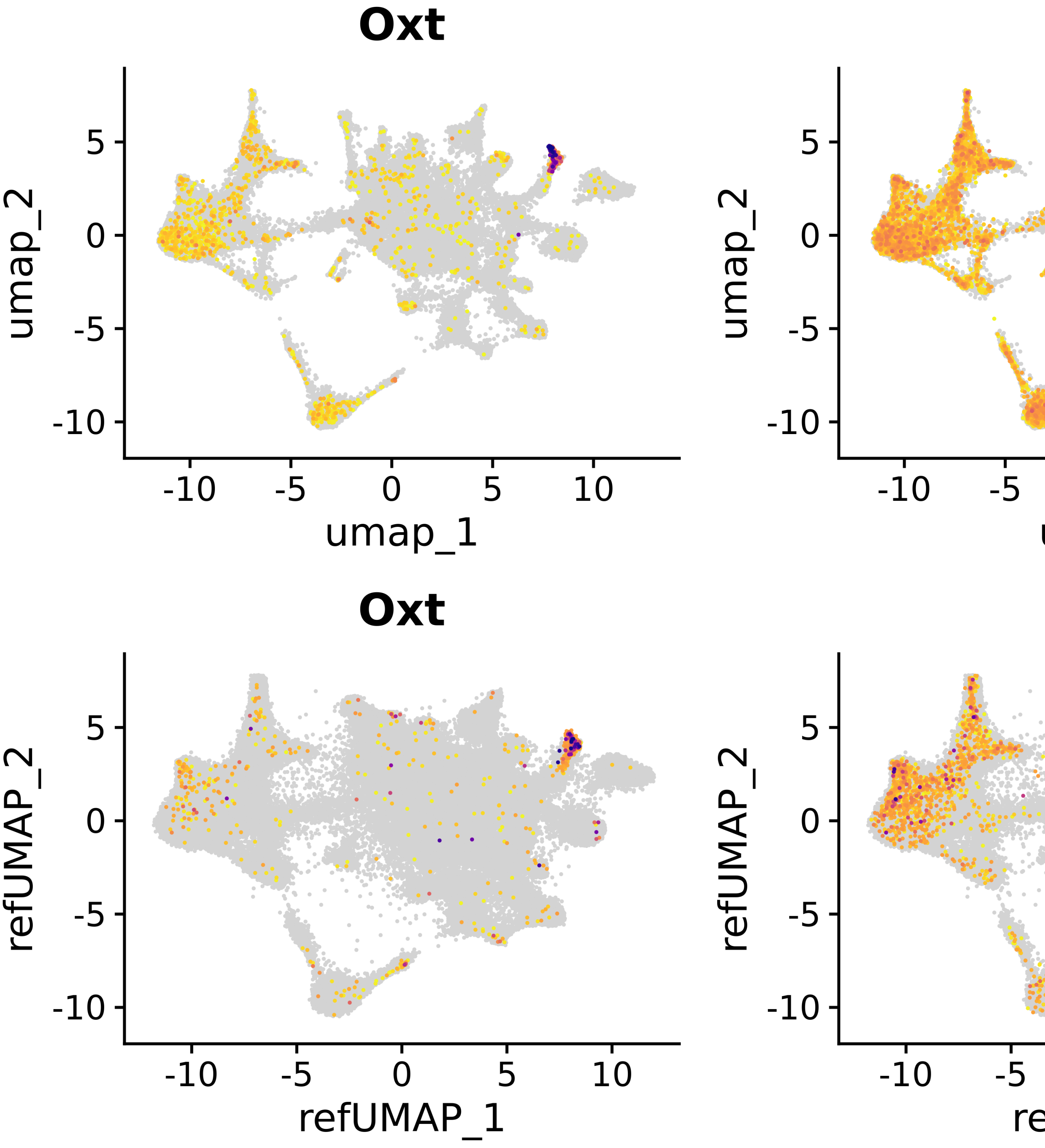
```
rar2020.srt.pub <- RunUMAP(rar2020.srt.pub, dims = 1:30, reduction = "pca", return.m
srt <- IntegrateEmbeddings(
  anchorset = hypoth.anchors, reference = rar2020.srt.pub, query = srt,
  new.reduction.name = "ref.pca"
)
srt <- ProjectUMAP(
  query = srt, query.reduction = "ref.pca", reference = rar2020.srt.pub,
```

```r
  reference.reduction = "pca", reduction.model = "umap"
)
Idents(srt) <- srt$Cluster
```

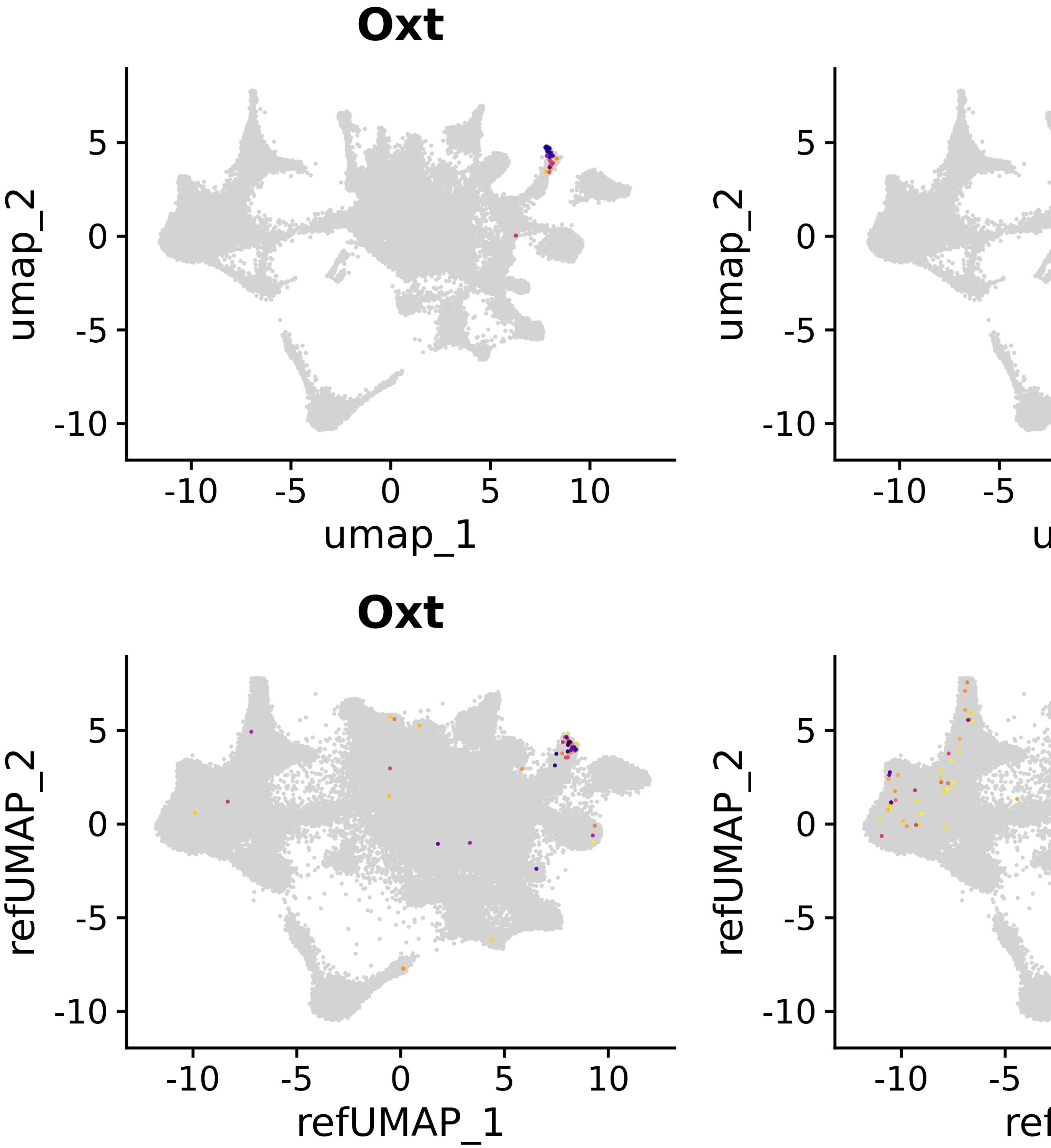
```r
p1 <- DimPlot(rar2020.srt.pub,
  reduction = "umap", group.by = "wtree", label = F
) + NoLegend() + ggtitle("Reference annotations")
p2 <- DimPlot(srt,
  reduction = "ref.umap", group.by = "Age", label = F
) + NoLegend() + ggtitle("Query transferred Embedding (more ages)")
p1 + p2
```


Reference annotations
Query transferred Em
umap_2
5
0
-5
-10
-10
-5
0
5
10
umap_1
refUMAP_2
5
0
-5
-10
-10
-5
0
refUM

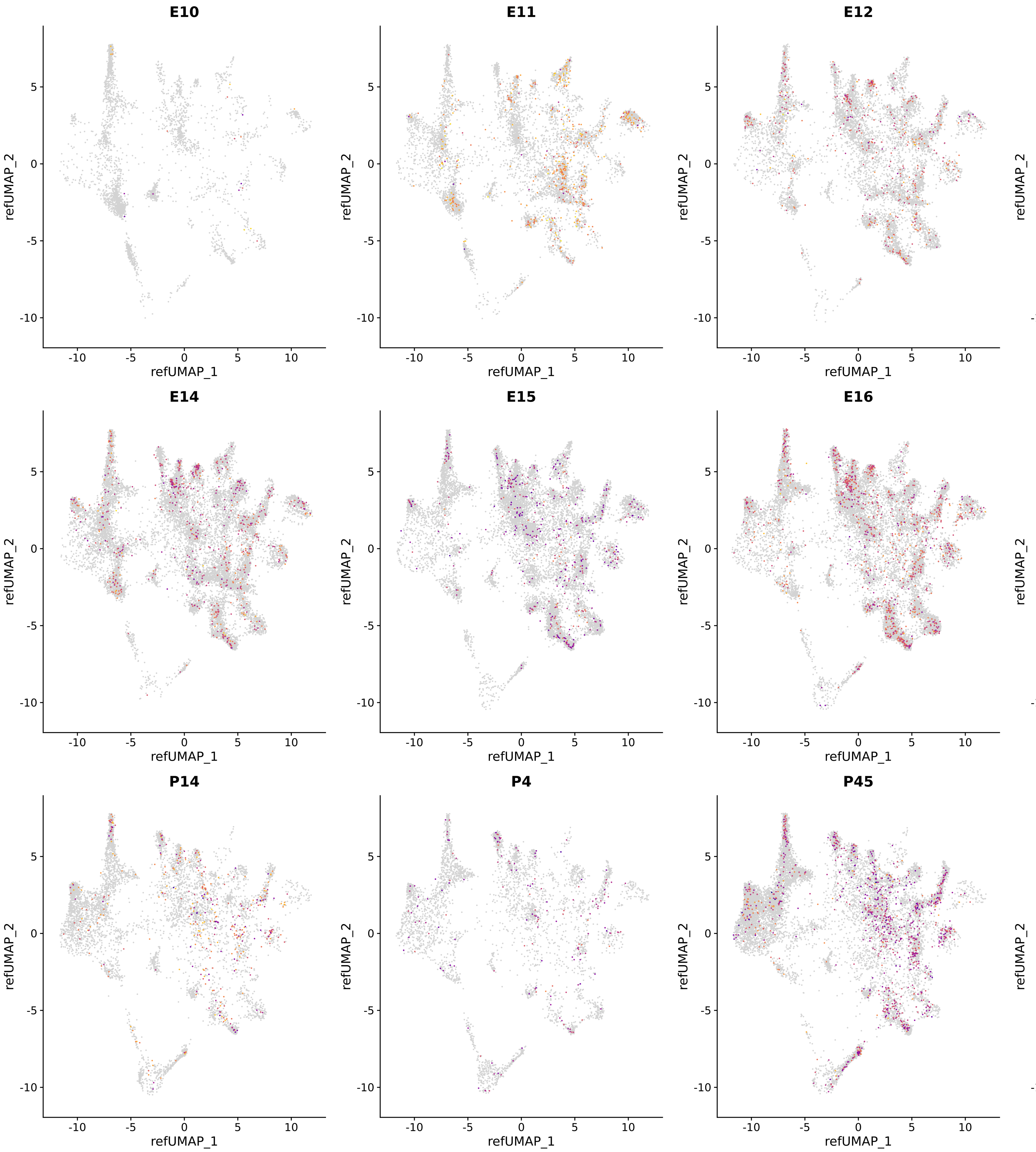
```r
p1 <- FeaturePlot_scCustom(
  rar2020.srt.pub,
  reduction = "umap",
  features = c(
    "Oxt",
    "Avp",
    "Sst",
    "Crh",
    "Trh"),
  label = F,
  num_columns = 5,
  min.cutoff = 'q05',
  na_cutoff = 2
) * NoLegend()
p2 <- FeaturePlot_scCustom(
  srt,
  reduction = "ref.umap",
  features = c(
    "Oxt",
    "Avp",
    "Sst",
    "Crh",
    "Trh"),
  label = FALSE,
  num_columns = 5,
  min.cutoff = 'q05',
  na_cutoff = 2
) * NoLegend()
(p1 / p2)
```


Oxt
umap_2
umap_1
5
0
-5
-10
-10
-5
0
5
10
umap_2
5
0
-5
-10
-10
-5
Oxt
refUMAP_2
refUMAP_1
5
0
-5
-10
-10
-5
0
5
10
refUMAP_2
5
0
-5
-10
-10
-5

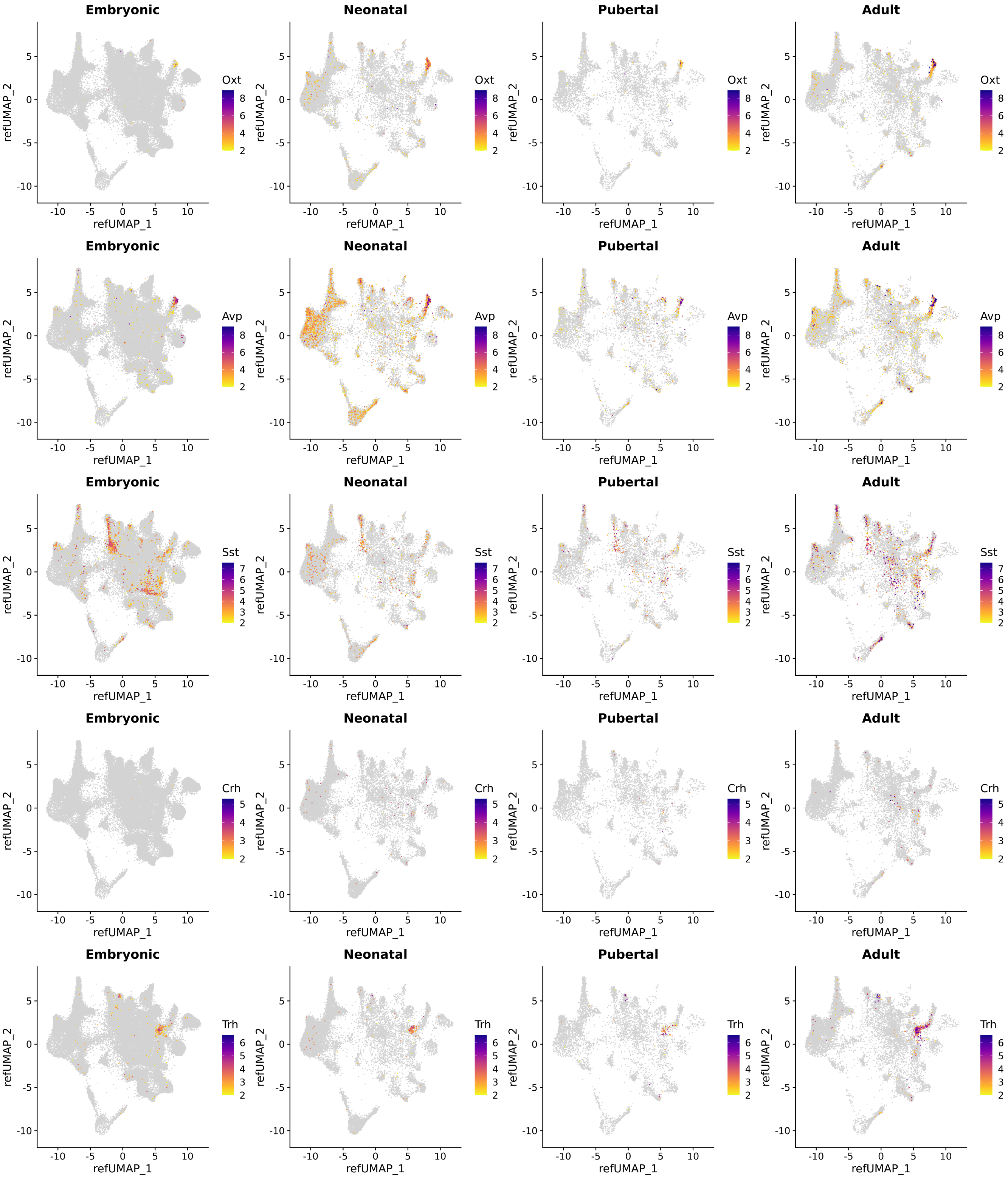
```r
p1 <- FeaturePlot_scCustom(
  rar2020.srt.pub,
  reduction = "umap",
  features = c(
    "Oxt",
    "Avp"),
  label = F,
  num_columns = 2,
  min.cutoff = 'q05',
  na_cutoff = 5
) * NoLegend()
p2 <- FeaturePlot_scCustom(
  srt,
  reduction = "ref.umap",
  features = c(
    "Oxt",
    "Avp"),
  label = FALSE,
  num_columns = 2,
  min.cutoff = 'q05',
  na_cutoff = 5
) * NoLegend()
(p1 / p2)
```


Oxt
umap_2
umap_1
5
0
-5
-10
-10
-5
0
5
10
umap_2
u
Oxt
refUMAP_2
refUMAP_1
refUMAP_2
ref

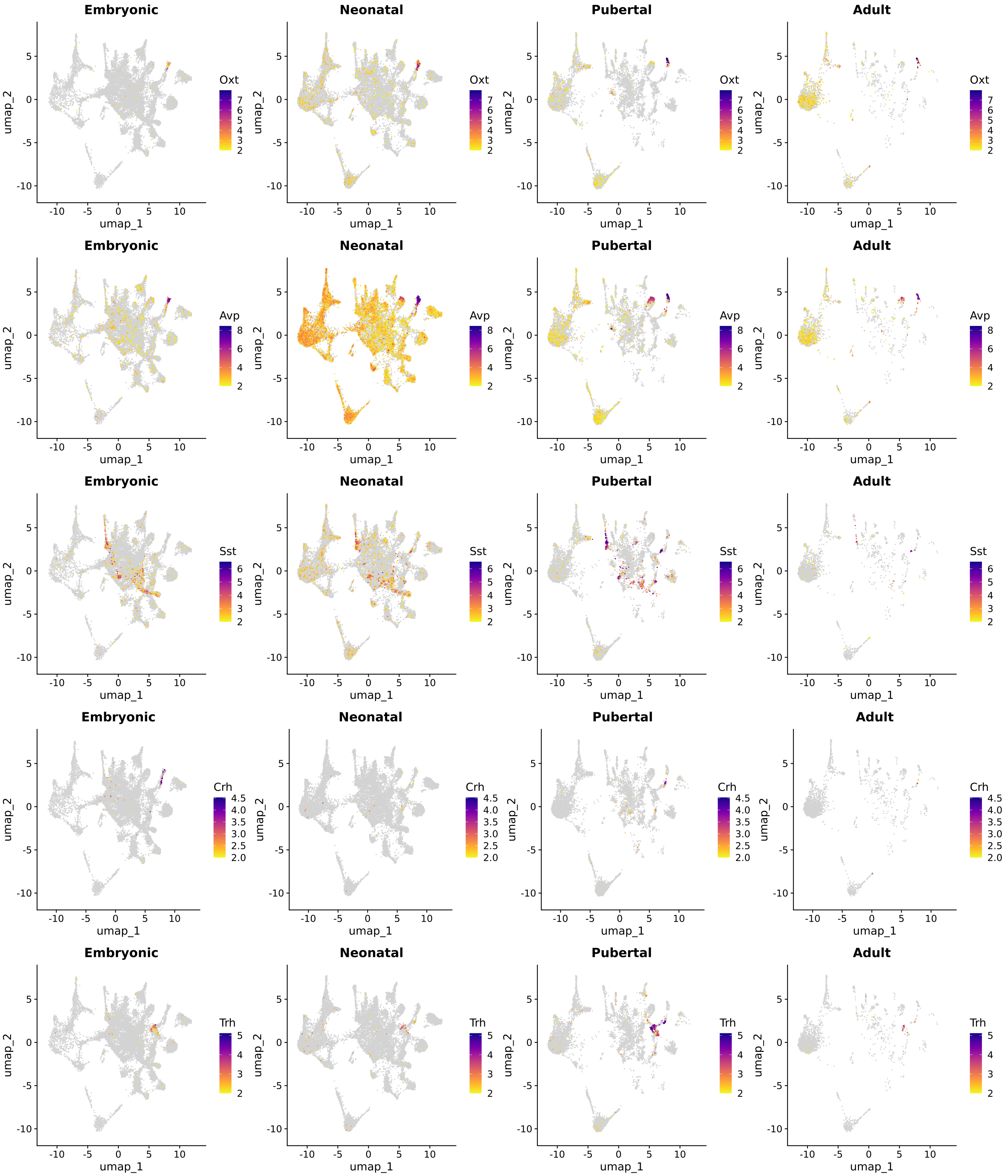
```
FeaturePlot_scCustom(srt, reduction = "ref.umap", features = c("Cnr1"), split.by = '
```


E10
E11
E12
E14
E15
E16
P14
P4
P45
refUMAP_1
refUMAP_2
-10
-5
0
5
10

```
srt$stage <-
  srt$Age %>%
  forcats::fct_collapse(
    Embryonic = c(
      "E10", "E11", "E12", "E13",
      "E14", "E15", "E16", "E18"
    ),
    Neonatal = c("P4", "P8"),
    Pubertal = c("P14"),
    Adult = c("P45")
  )
srt$stage %<>% factor(levels = c("Embryonic", "Neonatal", "Pubertal", "Adult"), orde
FeaturePlot_scCustom(
  srt,
  reduction = "ref.umap",
  features = c(
    "Oxt",
    "Avp",
    "Sst",
    "Crh",
    "Trh"),
  split.by = "stage",
  min.cutoff = 'q05',
  na_cutoff = 2,
  label = F,
  num_columns = 4
  ) * NoLegend()
```


Embryonic
Neonatal
Pubertal
Adult
Oxt
8
6
4
2
Avp
8
6
4
2
Sst
7
6
5
4
3
2
Crh
5
4
3
2
Trh
6
5
4
3
2
refUMAP_1
refUMAP_2
-10
-5
0
5
10

```r
FeaturePlot_scCustom(
  rar2020.srt.pub,
  reduction = "umap",
  features = c(
    "Oxt",
    "Avp",
    "Sst",
    "Crh",
    "Trh"),
  split.by = "stage",
  min.cutoff = 'q05',
  na_cutoff = 2,
  label = F,
  num_columns = 4
  ) * NoLegend()
```


Embryonic
Neonatal
Pubertal
Adult
Oxt
7
6
5
4
3
2
Avp
8
6
4
2
Sst
6
5
4
3
2
Crh
4.5
4.0
3.5
3.0
2.5
2.0
Trh
5
4
3
2
umap_1
umap_2
-10
-5
0
5
10

```r
if (!file.exists(here(data_dir, "kim2020_pvn_neurons.txt"))) {
  plot <- DimPlot(object = srt, reduction = "ref.umap")
  srt <- CellSelector(plot = plot, object = srt, ident = "SelectedCells")

  selected_cells <- Cells(subset(srt, idents = "SelectedCells"))
  write_lines(selected_cells, file = here(data_dir, "kim2020_pvn_neurons.txt"))
}
selected_cells <- read_lines(here(data_dir, "kim2020_pvn_neurons.txt"))
srt <- subset(srt, cells = selected_cells)
srt <- subset(srt, subset = refUMAP_1 > 4 & refUMAP_2 > -1)

srt@meta.data <- srt@meta.data |> rename(wtree = predicted.id, age = Age)

srt
```

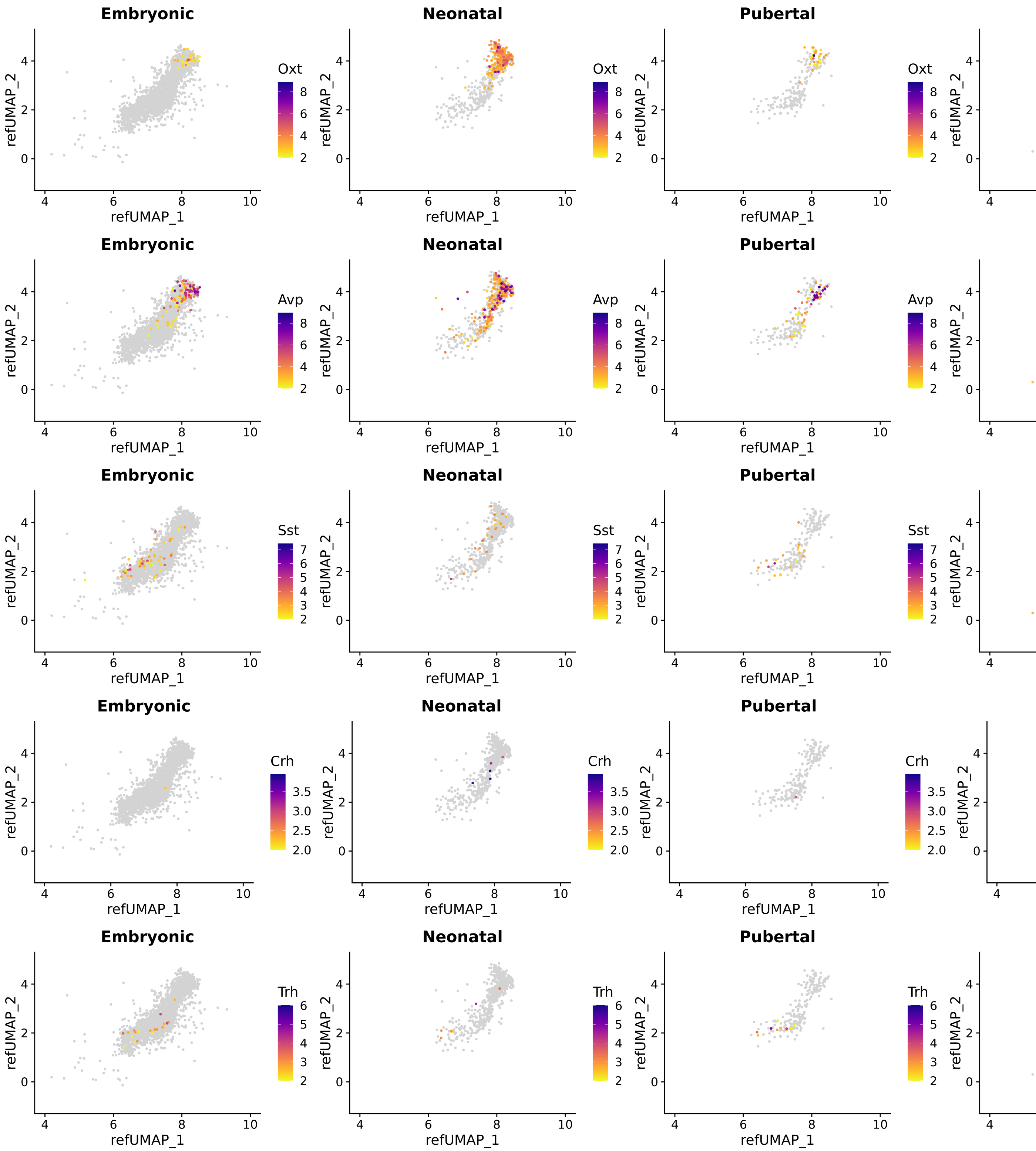
```
An object of class Seurat
27998 features across 4555 samples within 1 assay
Active assay: RNA (27998 features, 3000 variable features)
 3 layers present: counts, data, scale.data
 3 dimensional reductions calculated: umap, ref.pca, ref.umap
```

## Intersection sets analysis

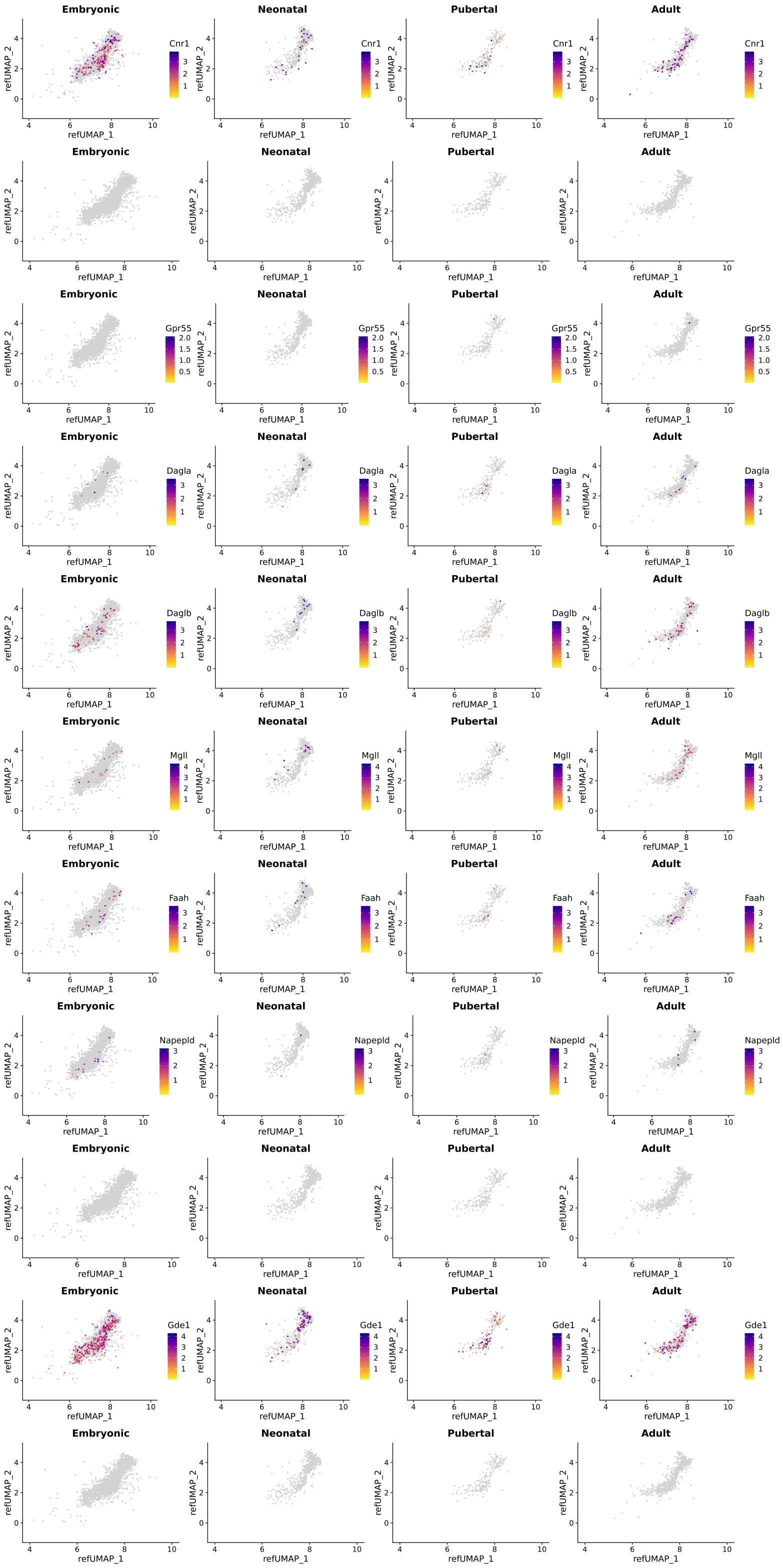
### PVN Neurons from Kim et al. 2020, Nature Communications

```r
FeaturePlot_scCustom(
  srt,
  reduction = "ref.umap",
  features = c(
    "Oxt",
    "Avp",
    "Sst",
    "Crh",
    "Trh"),
  split.by = "stage",
  na_cutoff = 2,
  label = F,
  num_columns = 4
```

```
    ) * NoLegend()
```


Embryonic
Neonatal
Pubertal
refUMAP_1
refUMAP_2
4
6
8
10
0
2
4
Oxt
8
6
4
2
Avp
8
6
4
2
Sst
7
6
5
4
3
2
Crh
3.5
3.0
2.5
2.0
Trh
6
5
4
3
2

```
FeaturePlot_scCustom(srt, reduction = "ref.umap", features = c(cnbn), split.by = "st
```

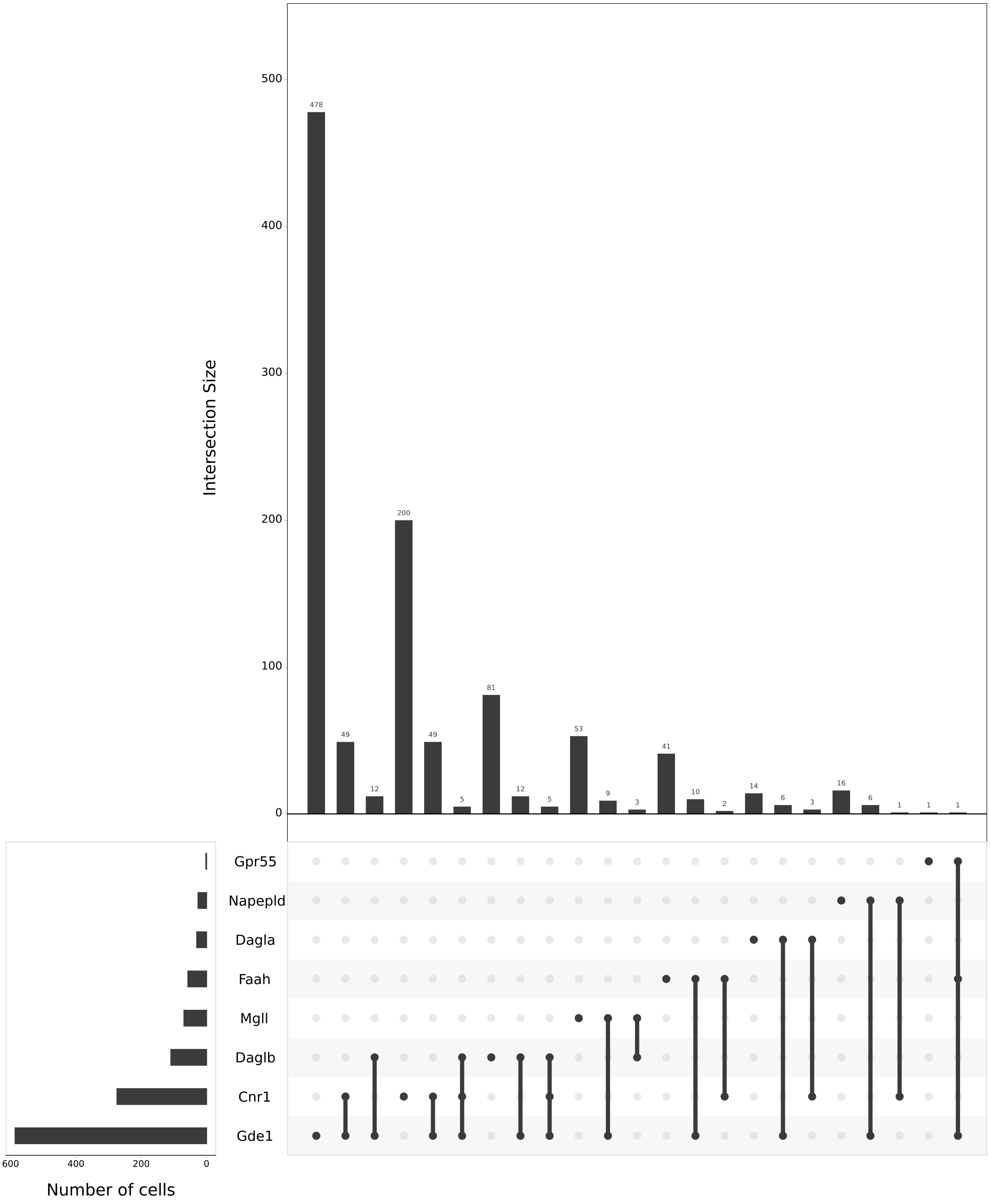

Embryonic
Neonatal
Pubertal
Adult
refUMAP_1
refUMAP_2
4
6
8
10
0
2
4
Cnr1
3
2
1
Gpr55
2.0
1.5
1.0
0.5
Dagla
3
2
1
Daglb
3
2
1
Mgll
4
3
2
1
Faah
3
2
1
Napepld
3
2
1
Gde1
4
3
2
1

```r
sbs_mtx <-
  srt@assays$RNA@layers$data %>%
  as.data.frame() %>%
  t()

rownames(sbs_mtx) <- colnames(srt)
colnames(sbs_mtx) <- rownames(srt)

# Filter features
filt_low_genes <-
  colSums(sbs_mtx) %>%
  .[. > quantile(., 0.4)] %>%
  names()
sbs_mtx %<>% .[, filt_low_genes]

min_filt_vector2 <-
  sbs_mtx %>%
  as_tibble() %>%
  select(all_of(filt_low_genes)) %>%
  summarise(across(.fns = ~ quantile(.x, .005))) %>%
  as.list() %>%
  map(as.double) %>%
  simplify() %>%
  .[filt_low_genes]

# Prepare table of intersection sets analysis
content_sbs_mtx_kim <-
  (sbs_mtx > min_filt_vector2) %>%
  as_tibble() %>%
  mutate_all(as.numeric) %>%
  bind_cols(
    srt@meta.data |> select(wtree, age, stage)
  )
```

## All

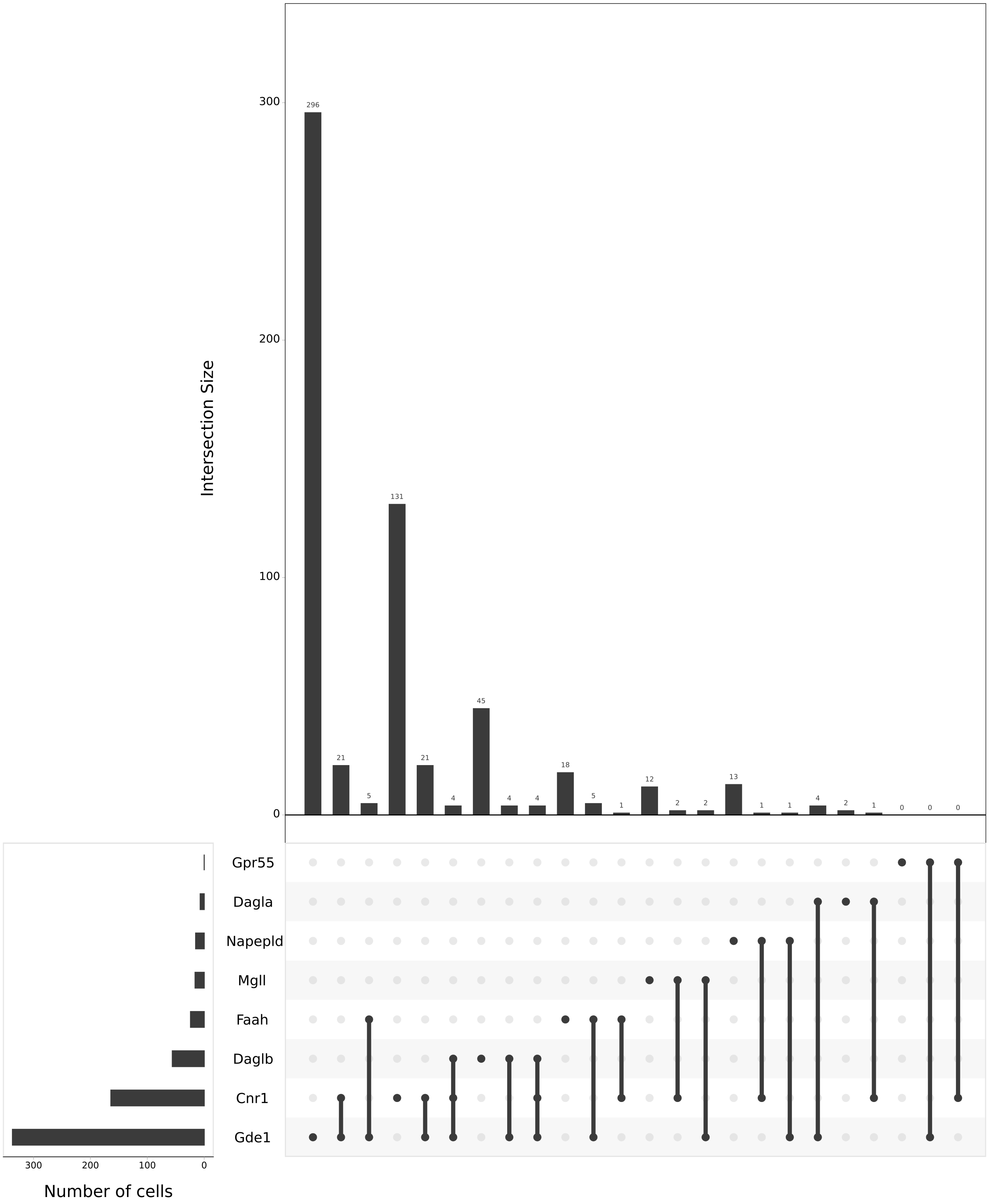
```r
upset(
  as.data.frame(content_sbs_mtx_kim),
  order.by = "freq",
```

```
  group.by = "sets",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_kim)],
  empty.intersections = NULL
)
```


Intersection Size
500
400
300
200
100
0
478
49
12
200
49
5
81
12
5
53
9
3
41
10
2
14
6
3
16
6
1
1
1
Gpr55
Napepld
Dagla
Faah
Mgll
Daglb
Cnr1
Gde1
600
400
200
0
Number of cells

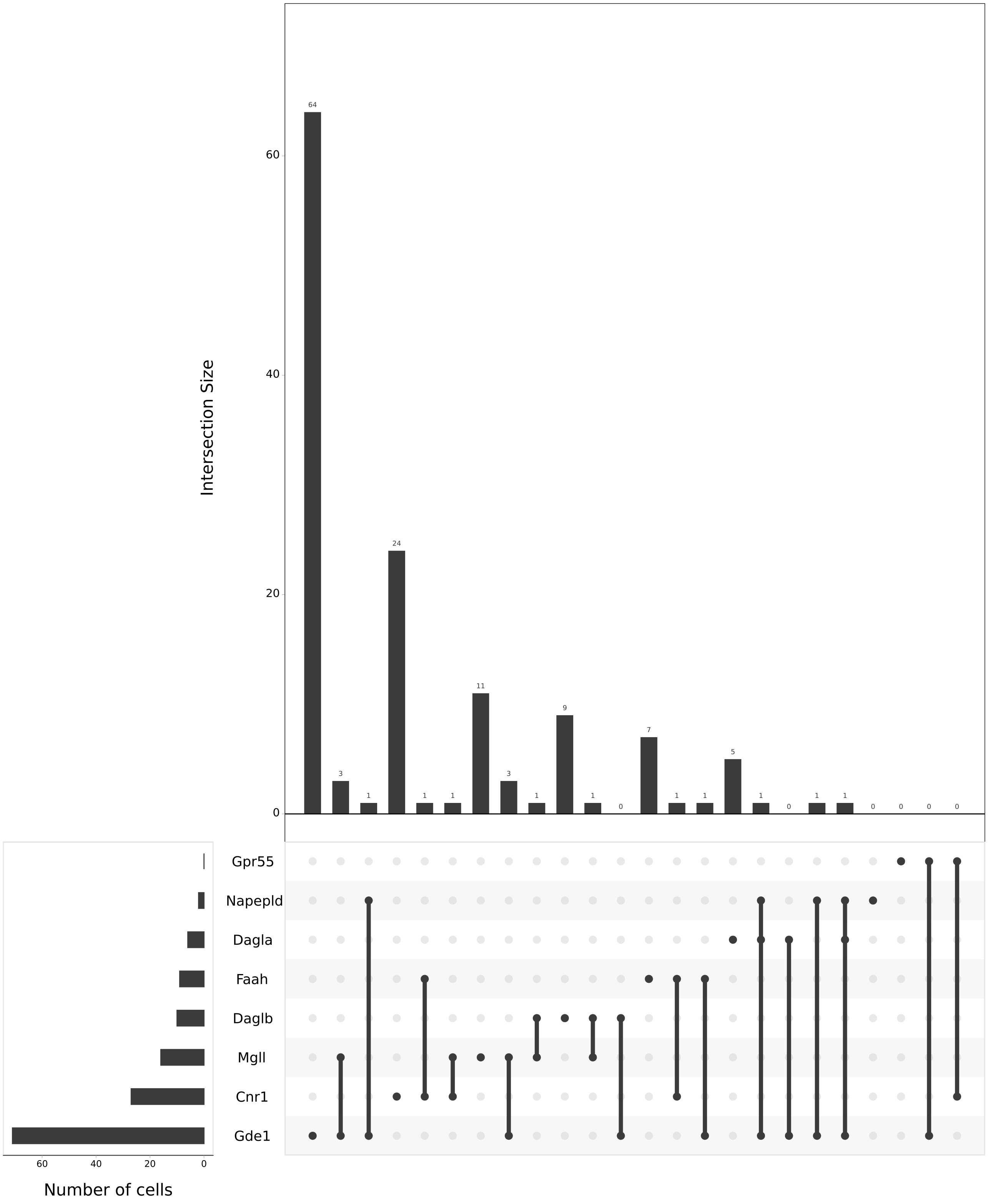
### Embryonic

```
upset(
  as.data.frame(
    content_sbs_mtx_kim |>
      filter(stage == "Embryonic") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx_kim)]
      )
  ),
  order.by = "freq",
  group.by = "sets",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_kim)],
  empty.intersections = F
)
```


Intersection Size
300
200
100
0
296
21
5
131
21
4
45
4
4
18
5
1
12
2
2
13
1
1
4
2
1
0
0
0
Gpr55
Dagla
Napepld
Mgll
Faah
Daglb
Cnr1
Gde1
300
200
100
0
Number of cells

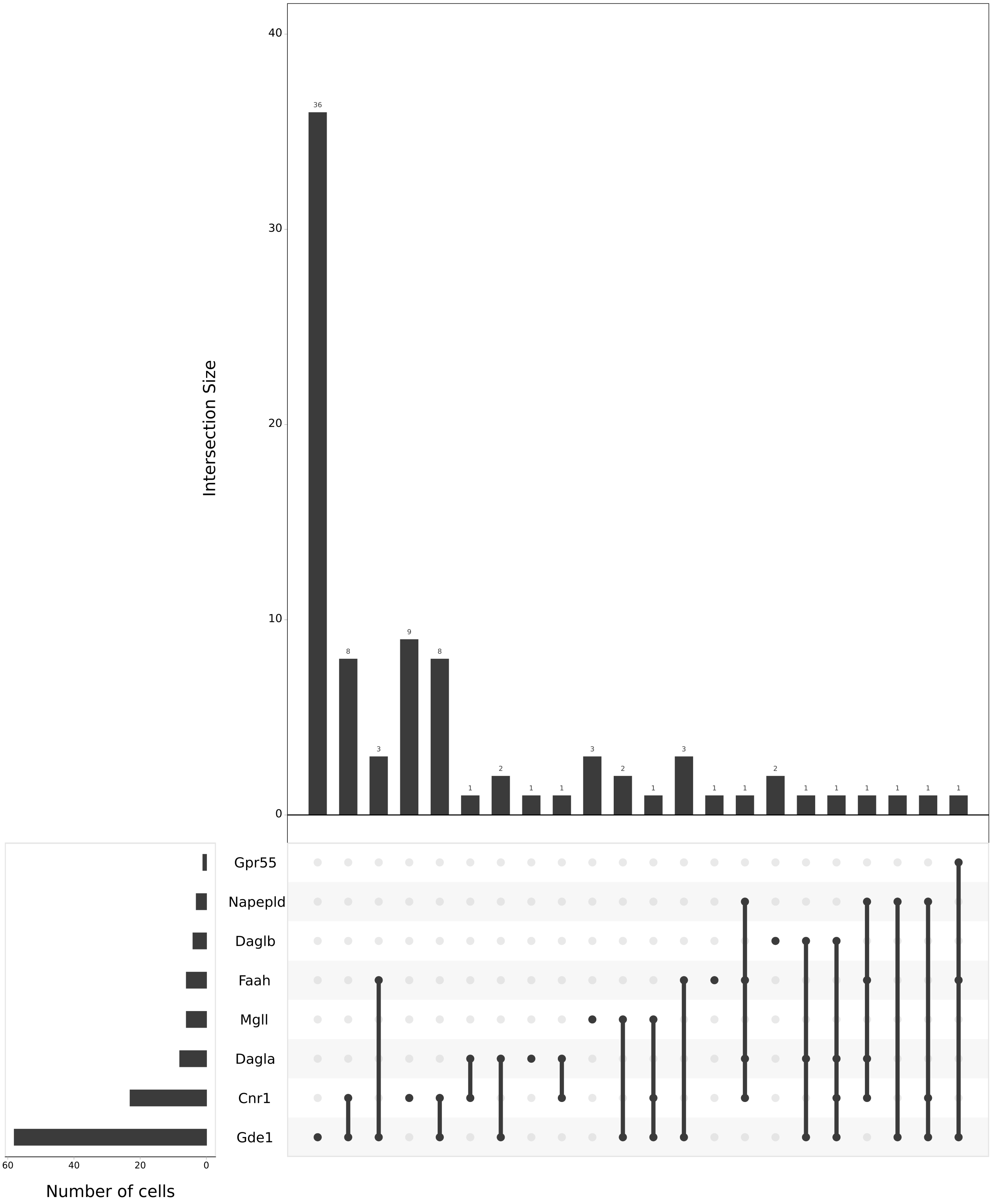
### Neonatal

```
upset(
  as.data.frame(
    content_sbs_mtx_kim |>
      filter(stage == "Neonatal") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx_kim)]
      )
  ),
  order.by = "freq",
  group.by = "sets",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_kim)],
  empty.intersections = F
)
```


Intersection Size
60
40
20
0
64
3
1
24
1
1
11
3
1
9
1
0
7
1
1
5
1
0
1
1
0
0
0
0
Gpr55
Napepld
Dagla
Faah
Daglb
Mgll
Cnr1
Gde1
60
40
20
0
Number of cells

### Pubertal

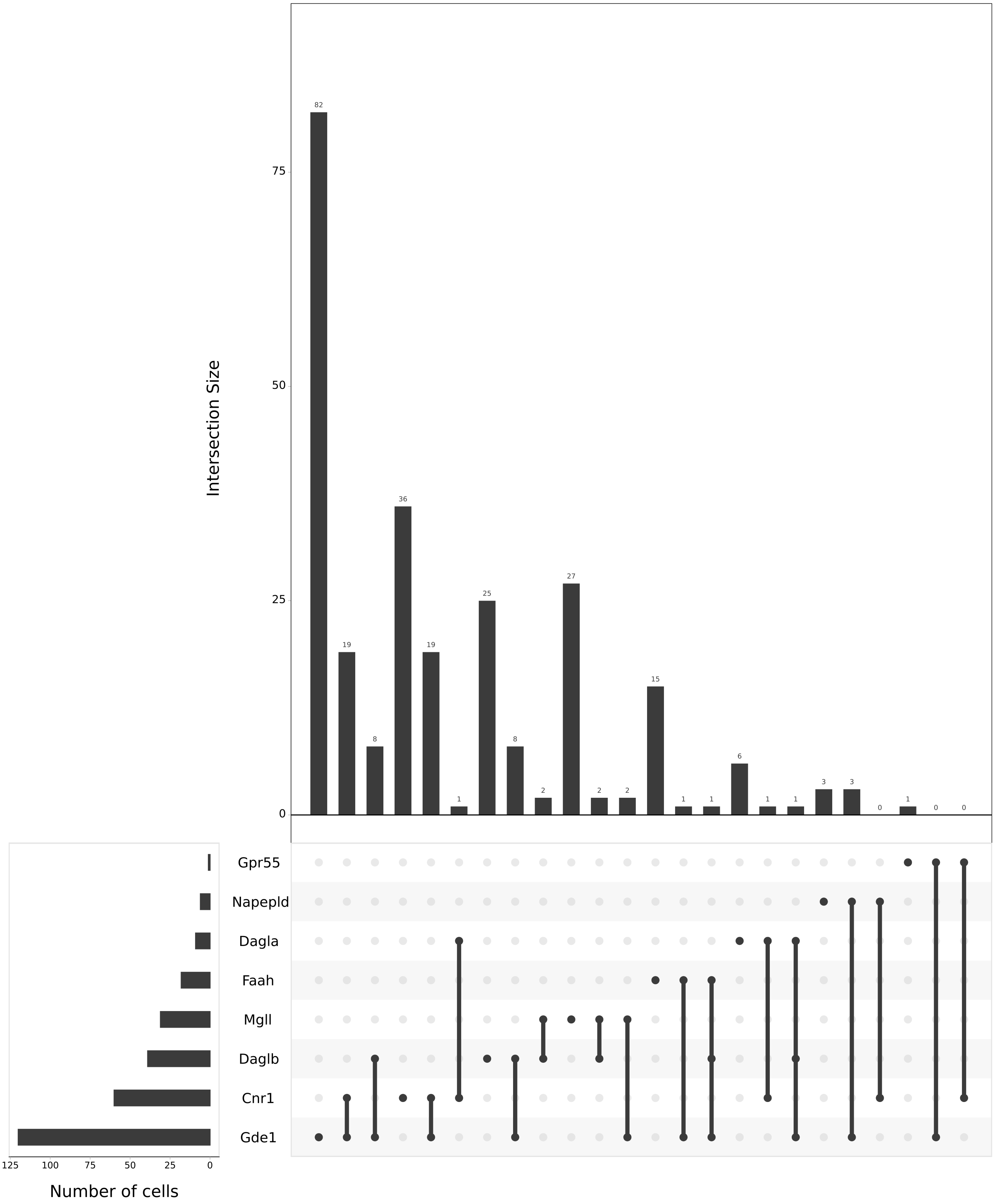
```r
upset(
  as.data.frame(
    content_sbs_mtx_kim |>
      filter(stage == "Pubertal") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx_kim)]
      )
  ),
  order.by = "freq",
  group.by = "sets",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_kim)],
  empty.intersections = NULL
)
```



Intersection Size
40
30
20
10
0
36
8
3
9
8
1
2
1
1
3
2
1
3
1
1
2
1
1
1
1
1
1
Gpr55
Napepld
Daglb
Faah
Mgll
Dagla
Cnr1
Gde1
60
40
20
0
Number of cells

### Adult

```r
upset(
  as.data.frame(
    content_sbs_mtx_kim |>
      filter(stage == "Adult") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx_kim)]
      )
  ),
  order.by = "freq",
  group.by = "sets",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_kim)],
  empty.intersections = F
)
```

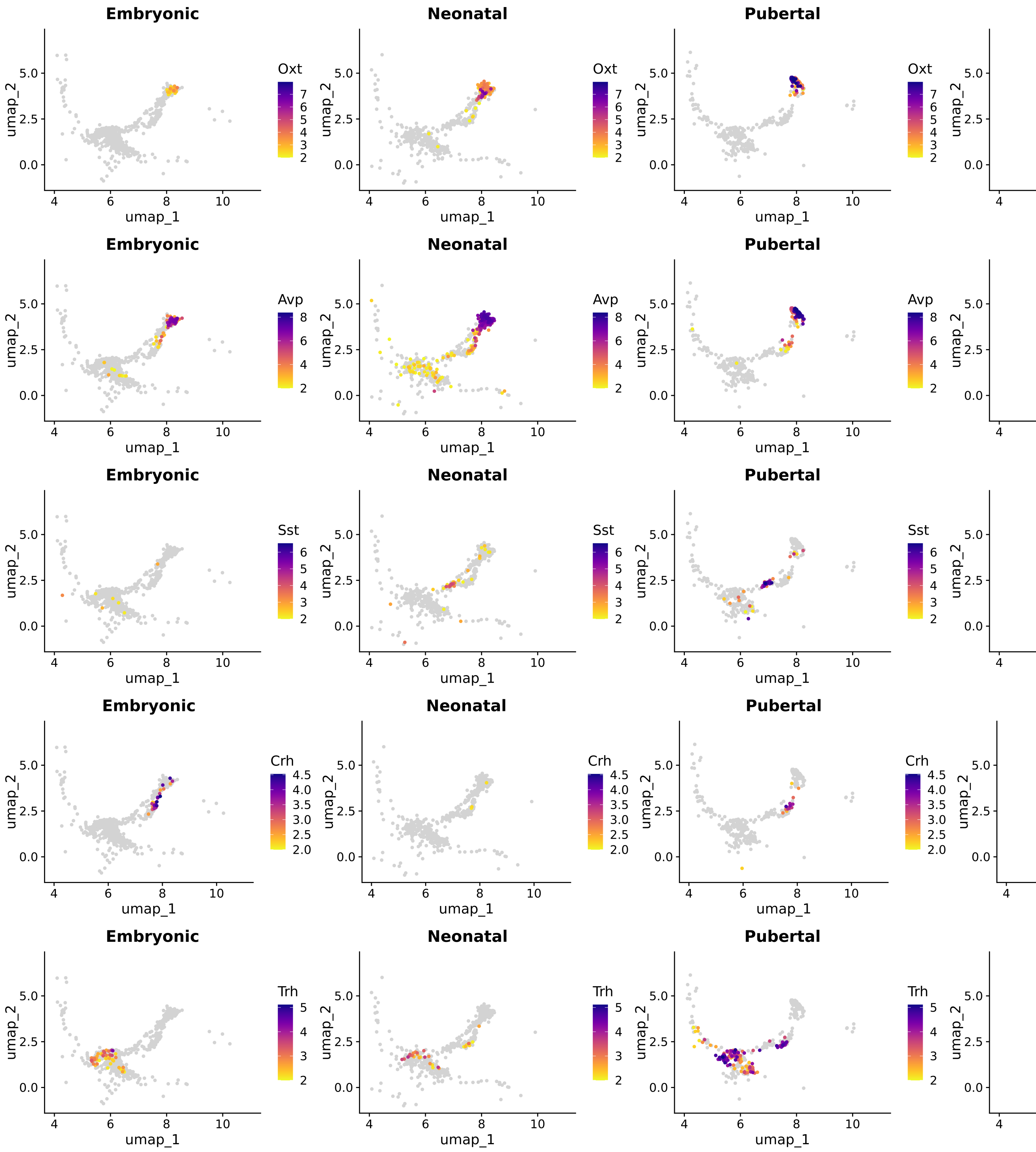


Intersection Size
0
25
50
75
82
19
8
36
19
1
25
8
2
27
2
2
15
1
1
6
1
1
3
3
0
1
0
0
Gpr55
Napepld
Dagla
Faah
Mgll
Daglb
Cnr1
Gde1
125
100
75
50
25
0
Number of cells

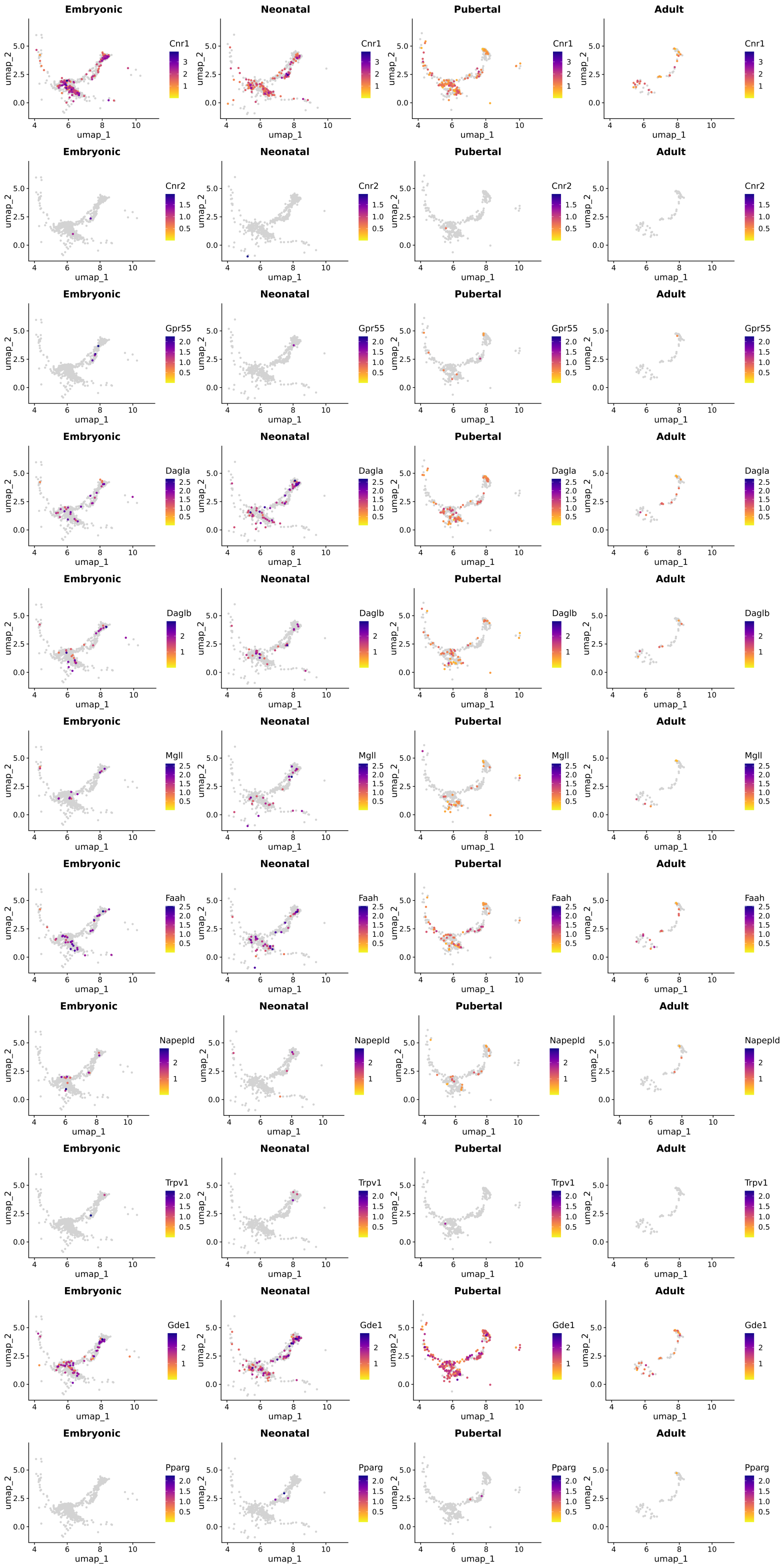
### PVN Neurons from Romanov et al. 2020, Nature

```r
rar2020.srt.pvn <-
  subset(
    x = rar2020.srt.pub,
    idents = c(
      "mneOXY", "mneVAS",
      "pneSS", "pneCRH", "pneTRH"
    ),
    invert = FALSE
  )

rar2020.srt.pvn <- subset(rar2020.srt.pvn, subset = umap_1 > 4 & umap_2 > -1)

rar2020.srt.pvn$age <-
  plyr::mapvalues(
    x = rar2020.srt.pvn$age,
    from = c("E15", "E17", "P0", "P2", "3P2", "1P10", "P10", "P23"),
    to = c("E15", "E17", "P00", "P02", "P02", "P10", "P10", "P23")
  )
```

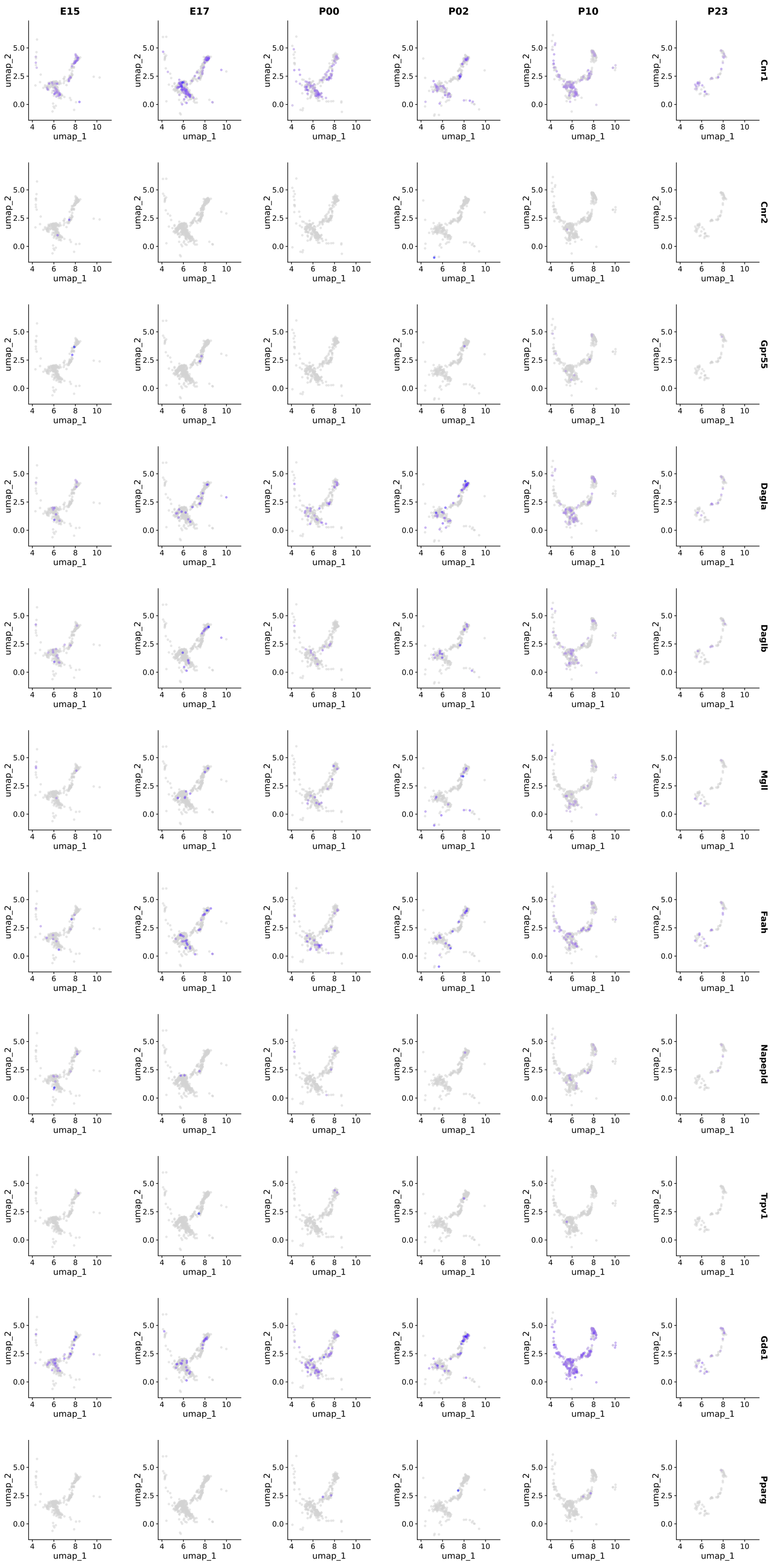
```r
FeaturePlot_scCustom(
  rar2020.srt.pvn,
  reduction = "umap",
  features = c(
    "Oxt",
    "Avp",
    "Sst",
    "Crh",
    "Trh"),
  split.by = "stage",
  na_cutoff = 2,
  label = F,
  num_columns = 4
  ) * NoLegend()
```


Embryonic
Neonatal
Pubertal
Oxt
7
6
5
4
3
2
Avp
8
6
4
2
Sst
6
5
4
3
2
Crh
4.5
4.0
3.5
3.0
2.5
2.0
Trh
5
4
3
2
umap_2
5.0
2.5
0.0
umap_1
4
6
8
10

```
FeaturePlot_scCustom(rar2020.srt.pvn, reduction = "umap", features = c(cnbn), split.
```


Embryonic
Neonatal
Pubertal
Adult
umap_1
umap_2
Cnr1
Cnr2
Gpr55
Dagla
Daglb
Mgll
Faah
Napepld
Trpv1
Gde1
Pparg

```r
FeaturePlot(
  rar2020.srt.pvn,
  features = c(cnbn),
  label = F,
  blend = F,
  order = TRUE,
  pt.size = 1.2,
  raster.dpi = c(1024, 1024),
  alpha = 0.5,
  split.by = "age"
)
```

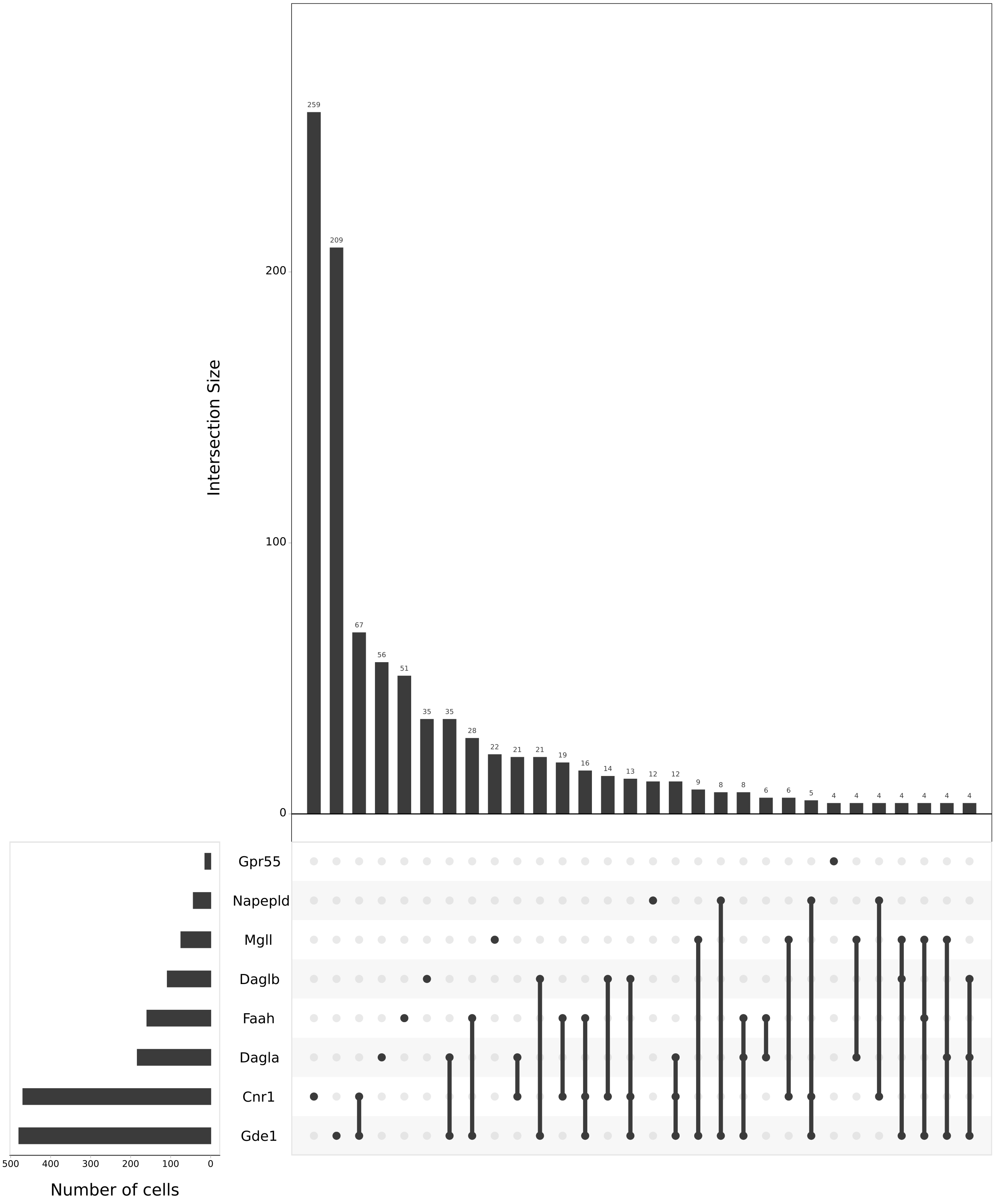

E15
E17
P00
P02
P10
P23
Cnr1
Cnr2
Gpr55
Dagla
Daglb
Mgll
Faah
Napepld
Trpv1
Gde1
Pparg
umap_1
umap_2
0.0
2.5
5.0
4
6
8
10

```
sbs_mtx <-
  rar2020.srt.pvn@assays$RNA@data %>%
  as.data.frame() %>%
  t()

# Filter features
filt_low_genes <-
  colSums(sbs_mtx) %>%
  .[. > quantile(., 0.4)] %>%
  names()
sbs_mtx %<>% .[, filt_low_genes]

min_filt_vector2 <-
  sbs_mtx %>%
  as_tibble() %>%
  select(all_of(filt_low_genes)) %>%
  summarise(across(.fns = ~ quantile(.x, .005))) %>%
  as.list() %>%
  map(as.double) %>%
  simplify() %>%
  .[filt_low_genes]

# Prepare table of intersection sets analysis
content_sbs_mtx_romanov <-
  (sbs_mtx > min_filt_vector2) %>%
  as_tibble() %>%
  mutate_all(as.numeric) %>%
  bind_cols(
    rar2020.srt.pvn@meta.data |> select(wtree, age, stage)
  )
```

### All

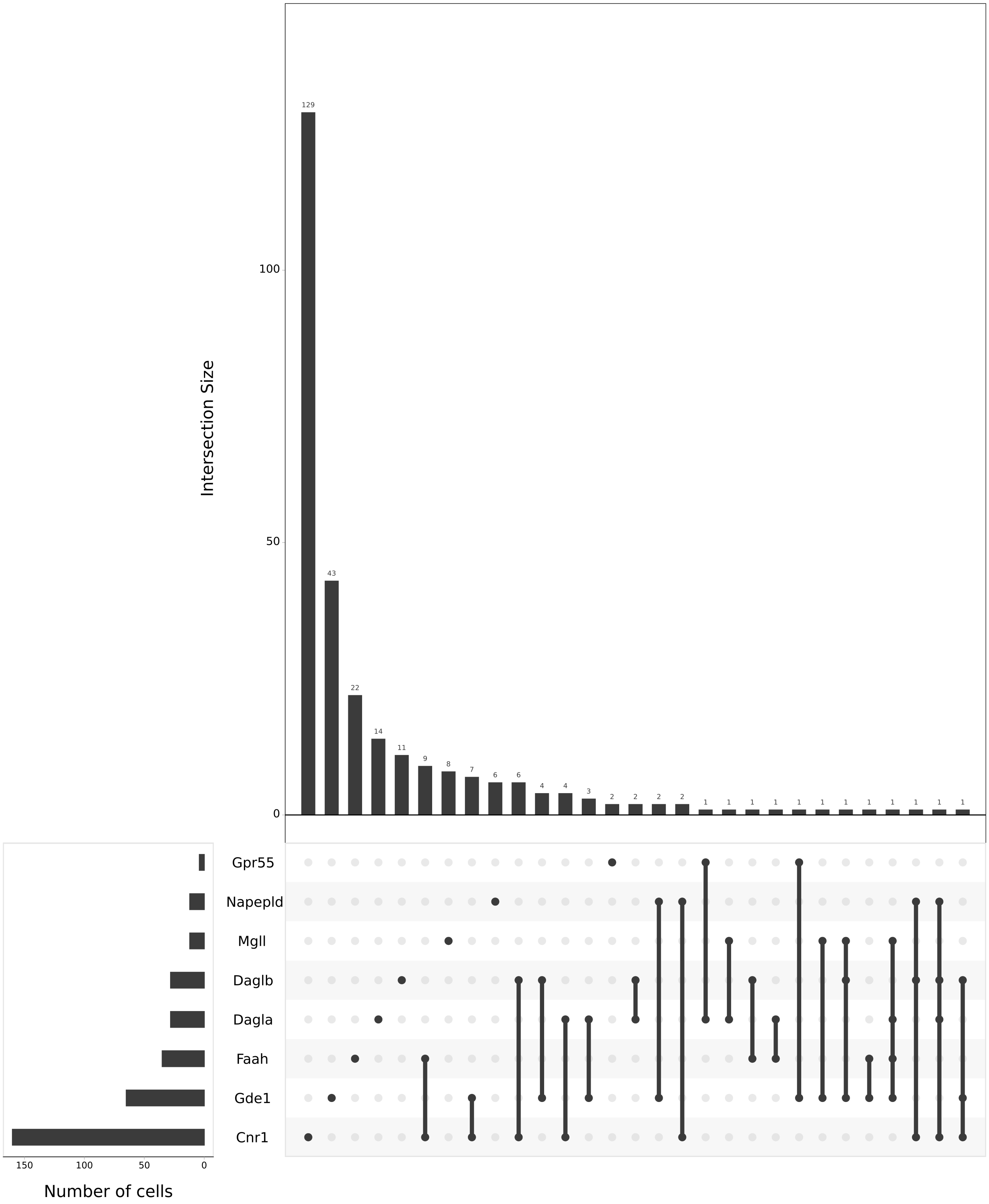
```
upset(
  as.data.frame(content_sbs_mtx_romanov),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
```

```
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_romanov)],
  empty.intersections = NULL
)
```


Intersection Size
0
100
200
259
209
67
56
51
35
35
28
22
21
21
19
16
14
13
12
12
9
8
8
6
6
5
4
4
4
4
4
4
4
Gpr55
Napepld
Mgll
Daglb
Faah
Dagla
Cnr1
Gde1
500
400
300
200
100
0
Number of cells

### Embryonic

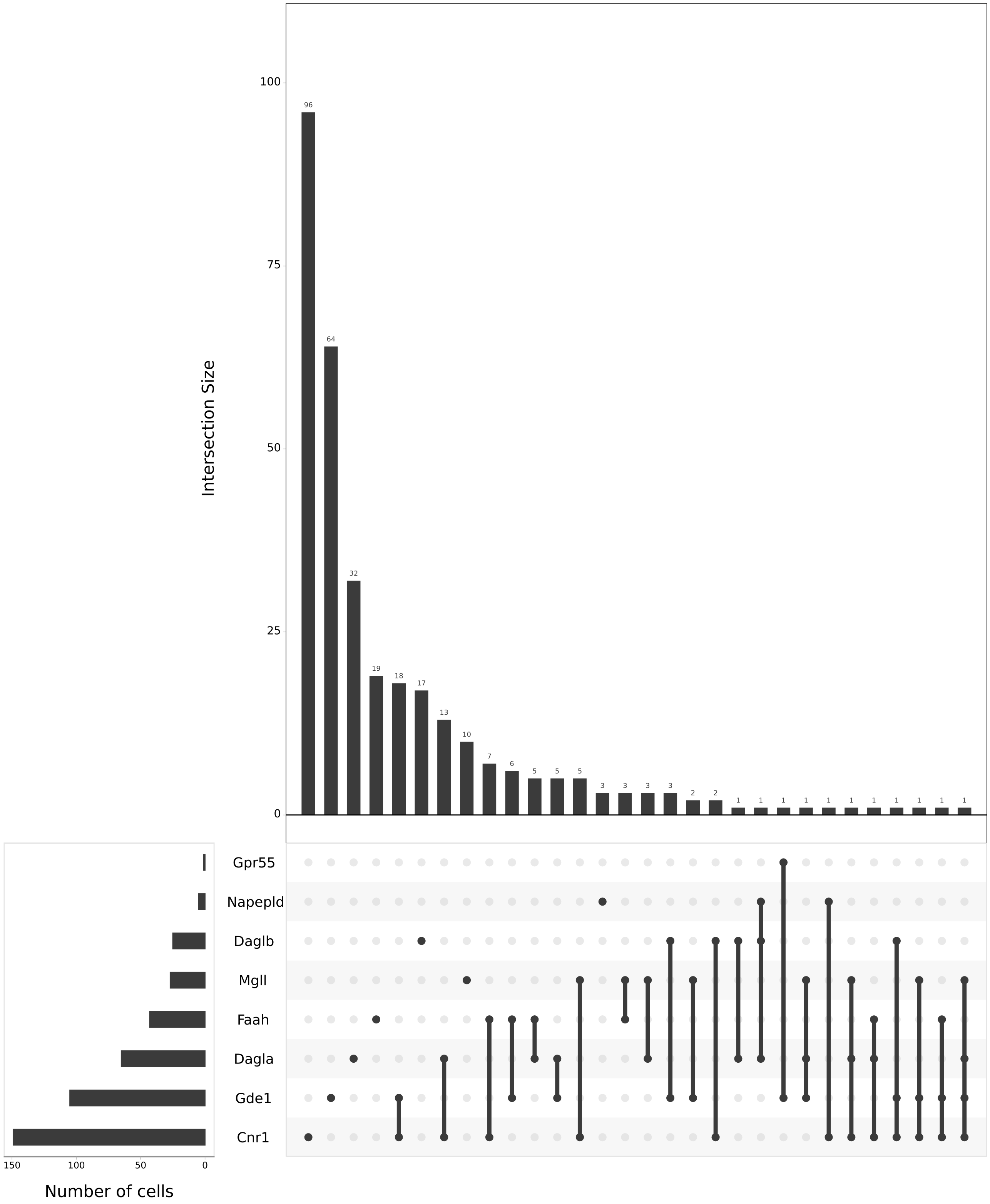
```
upset(
  as.data.frame(
    content_sbs_mtx_romanov |>
      filter(stage == "Embryonic") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx_romanov)]
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_romanov)],
  empty.intersections = NULL
)
```



Intersection Size
0
50
100
129
43
22
14
11
9
8
7
6
6
4
4
3
2
2
2
2
1
1
1
1
1
1
1
1
1
1
1
1
Gpr55
Napepld
Mgll
Daglb
Dagla
Faah
Gde1
Cnr1
150
100
50
0
Number of cells

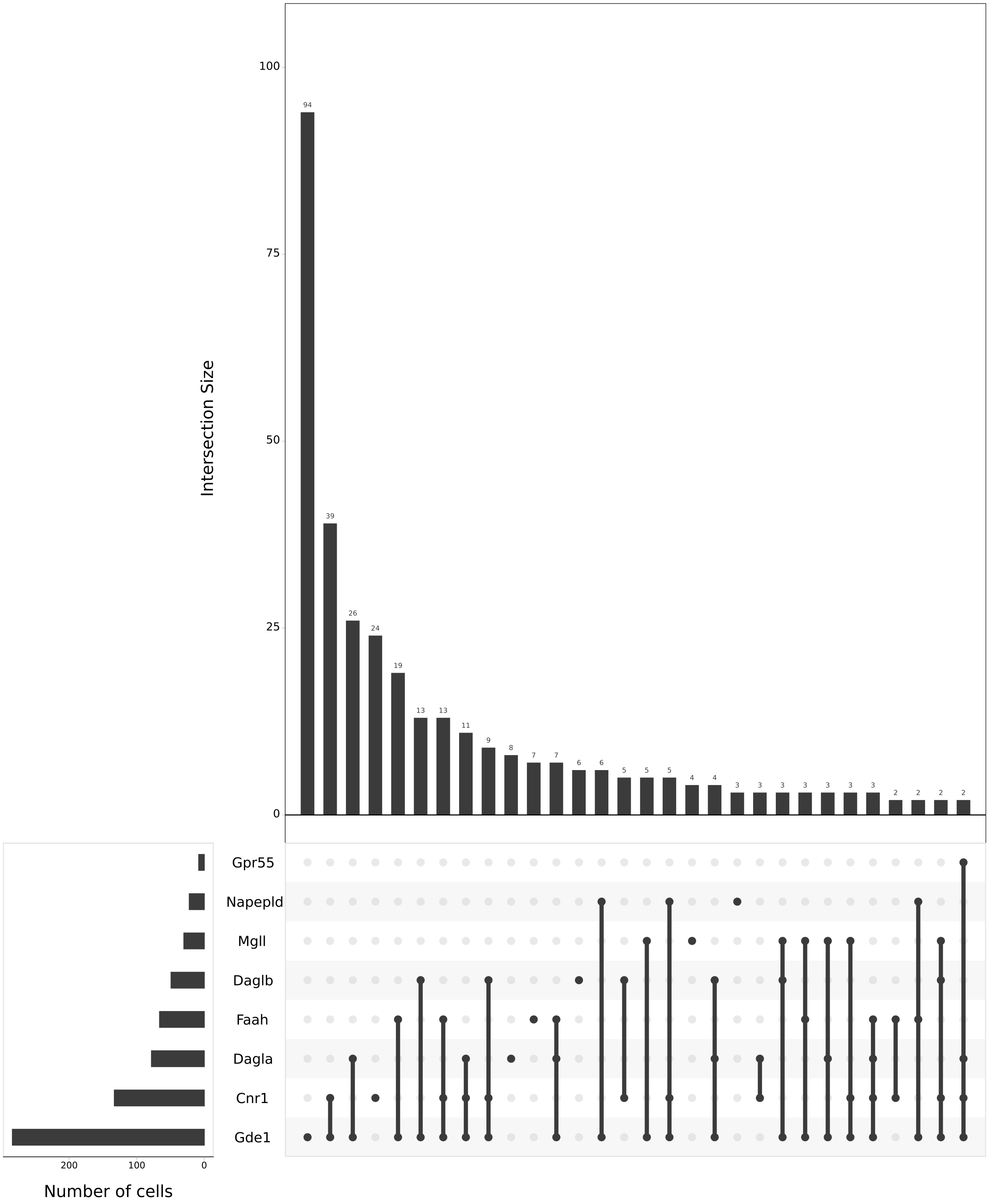
### Neonatal

```
upset(
  as.data.frame(
    content_sbs_mtx_romanov |>
      filter(stage == "Neonatal") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx_romanov)]
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_romanov)],
  empty.intersections = NULL
)
```


Intersection Size
100
75
50
25
0
96
64
32
19
18
17
13
10
7
6
5
5
5
3
3
3
3
2
2
1
1
1
1
1
1
1
1
1
1
1
Gpr55
Napepld
Daglb
Mgll
Faah
Dagla
Gde1
Cnr1
150
100
50
0
Number of cells

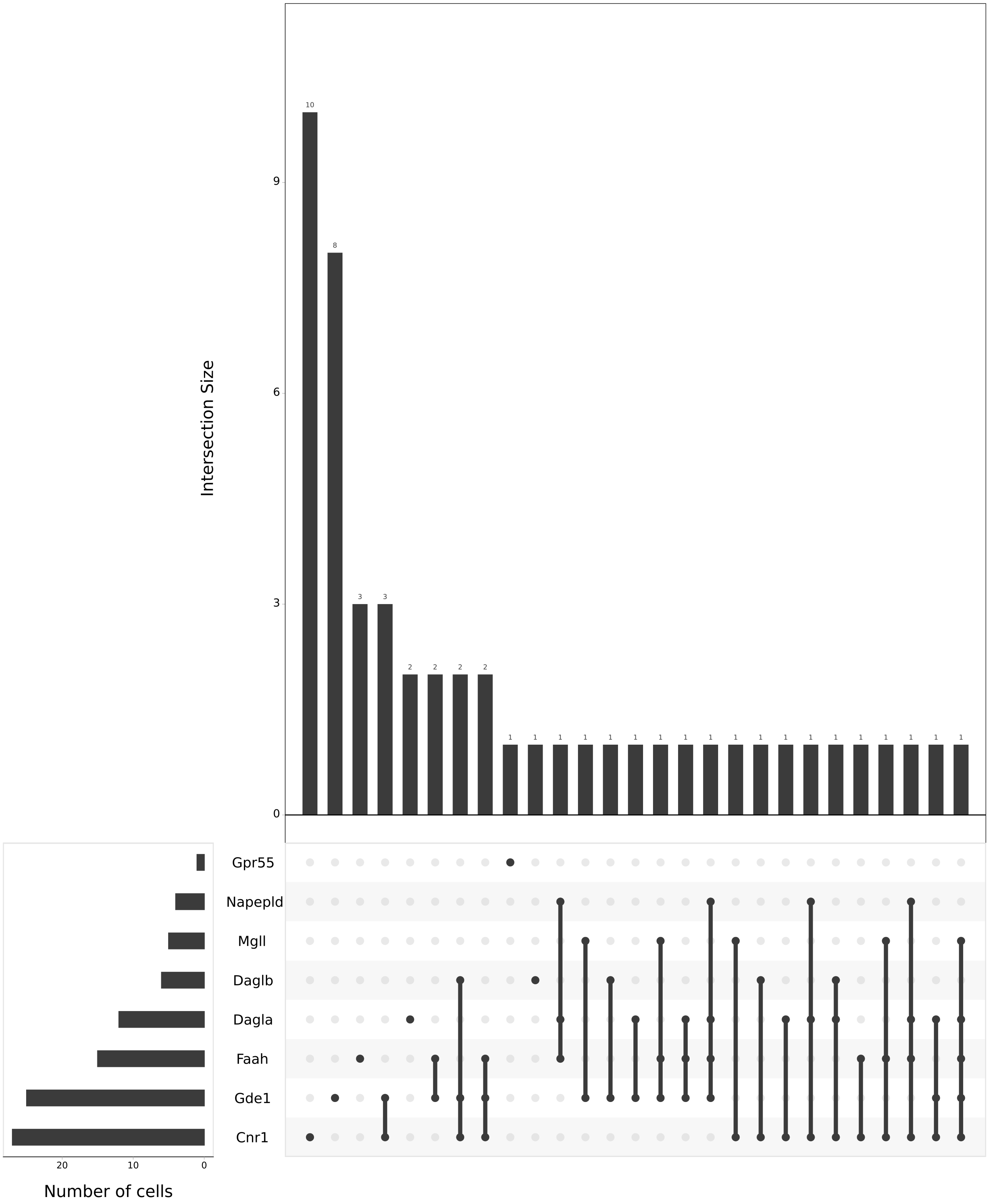
### Pubertal

```r
upset(
  as.data.frame(
    content_sbs_mtx_romanov |>
      filter(stage == "Pubertal") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx_romanov)]
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_romanov)],
  empty.intersections = NULL
)
```


Intersection Size
100
75
50
25
0
94
39
26
24
19
13
13
11
9
8
7
7
6
6
5
5
5
4
4
3
3
3
3
3
3
3
2
2
2
2
Gpr55
Napepld
Mgll
Daglb
Faah
Dagla
Cnr1
Gde1
200
100
0
Number of cells

### Adult

```
upset(
  as.data.frame(
    content_sbs_mtx_romanov |>
      filter(stage == "Adult") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx_romanov)]
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx_romanov)],
  empty.intersections = NULL
)
```

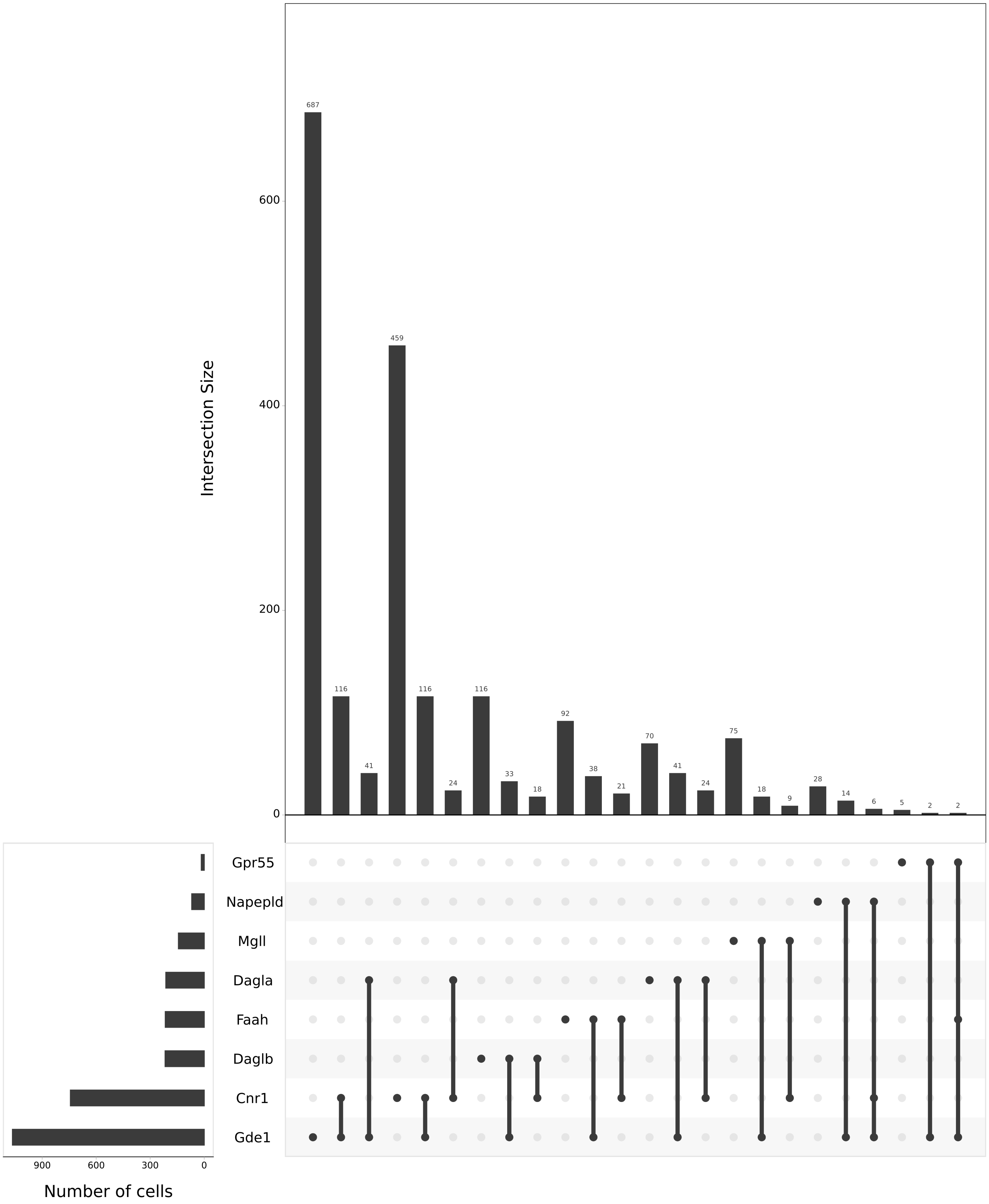

Intersection Size
0
3
6
9
10
8
3
3
2
2
2
2
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
1
Gpr55
Napepld
Mgll
Daglb
Dagla
Faah
Gde1
Cnr1
20
10
0
Number of cells

### PVN Neurons from both datasets joined

```r
# Prepare table of intersection sets analysis
to_select <-
  c(cnbn, "Oxt", "Crh", "Trh", "Avp", "Sst", "wtree", "age", "stage") %>%
  .[. %in% colnames(content_sbs_mtx_kim)] %>%
  .[. %in% colnames(content_sbs_mtx_romanov)]

content_sbs_mtx <-
  bind_rows(
    content_sbs_mtx_kim |> select(all_of(to_select)),
    content_sbs_mtx_romanov |> select(all_of(to_select))
  )
```

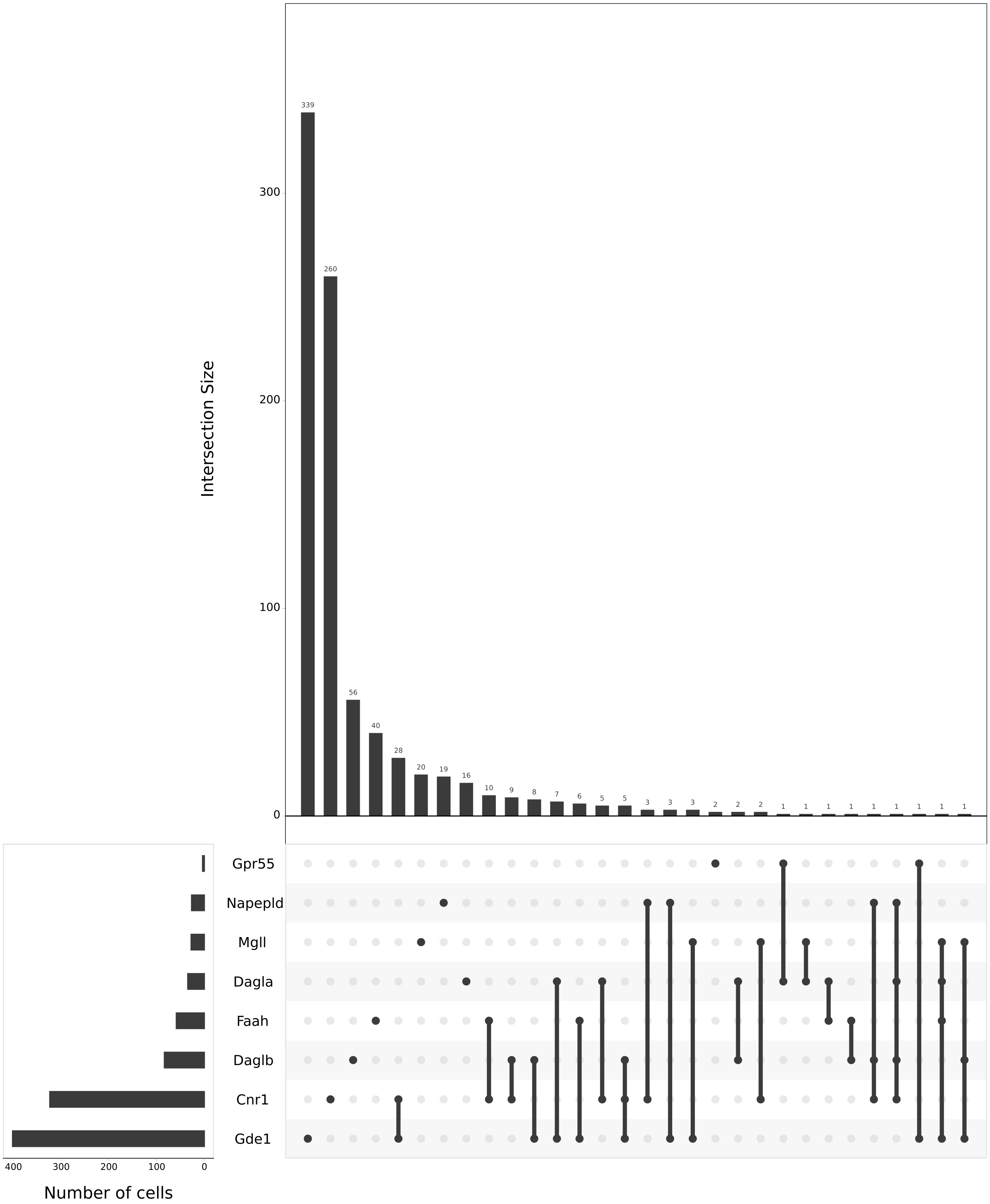
### All

```r
upset(
  as.data.frame(content_sbs_mtx),
  order.by = "freq",
  group.by = "sets",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```

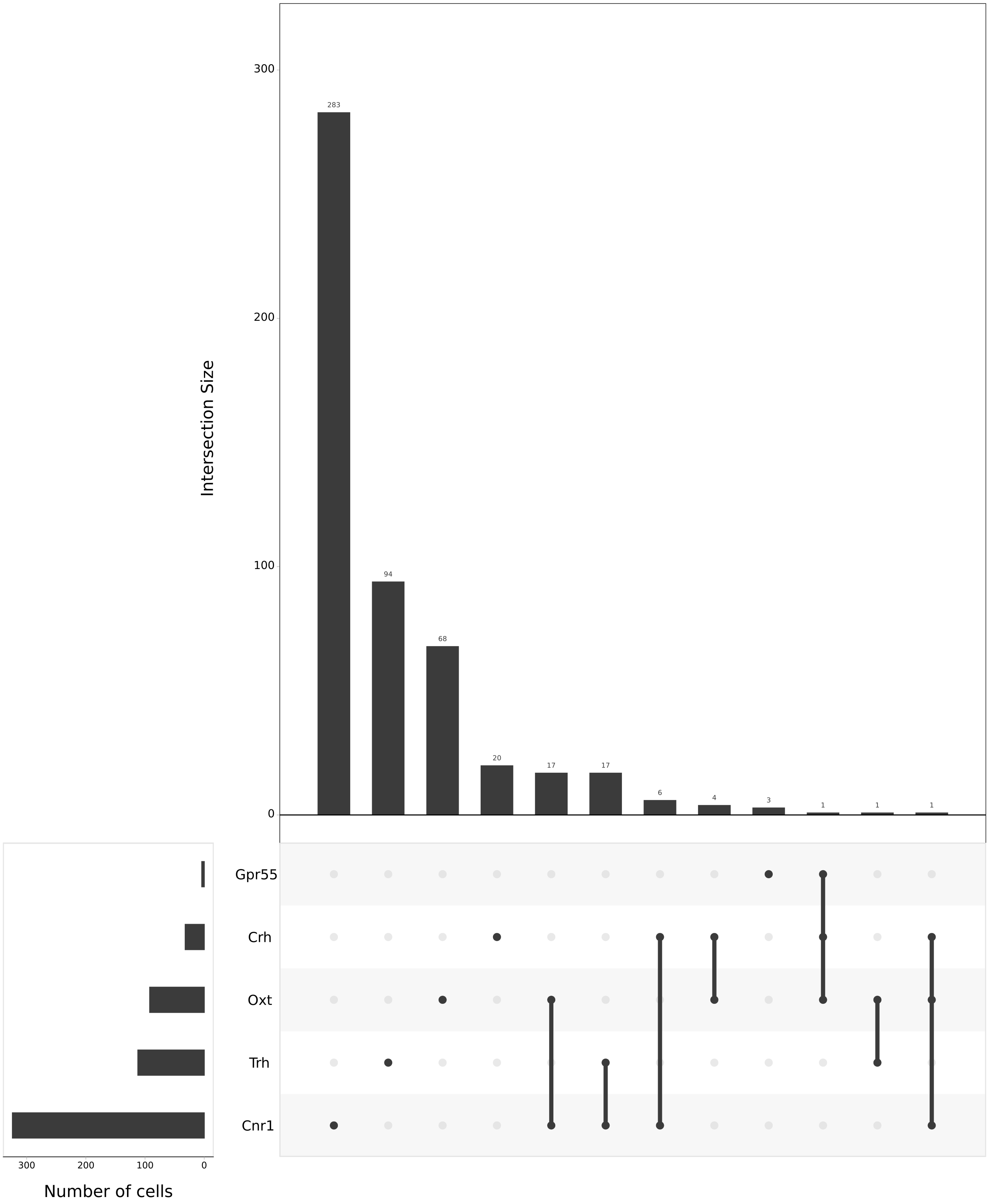

Intersection Size
600
400
200
0
687
116
41
459
116
24
116
33
18
92
38
21
70
41
24
75
18
9
28
14
6
5
2
2
Gpr55
Napepld
Mgll
Dagla
Faah
Daglb
Cnr1
Gde1
900
600
300
0
Number of cells

### Embryonic

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Embryonic") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx)]
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
0
100
200
300
339
260
56
40
28
20
19
16
10
9
8
7
6
5
5
3
3
3
2
2
2
1
1
1
1
1
1
1
1
1
Gpr55
Napepld
Mgll
Dagla
Faah
Daglb
Cnr1
Gde1
400
300
200
100
0
Number of cells

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Embryonic") |>
      select(
        c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conter
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
300
200
100
0
283
94
68
20
17
17
6
4
3
1
1
1
Gpr55
Crh
Oxt
Trh
Cnr1
300
200
100
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Embryonic") |>
      select(
        c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conter
      )
  ))
```

Table 1: Data summary

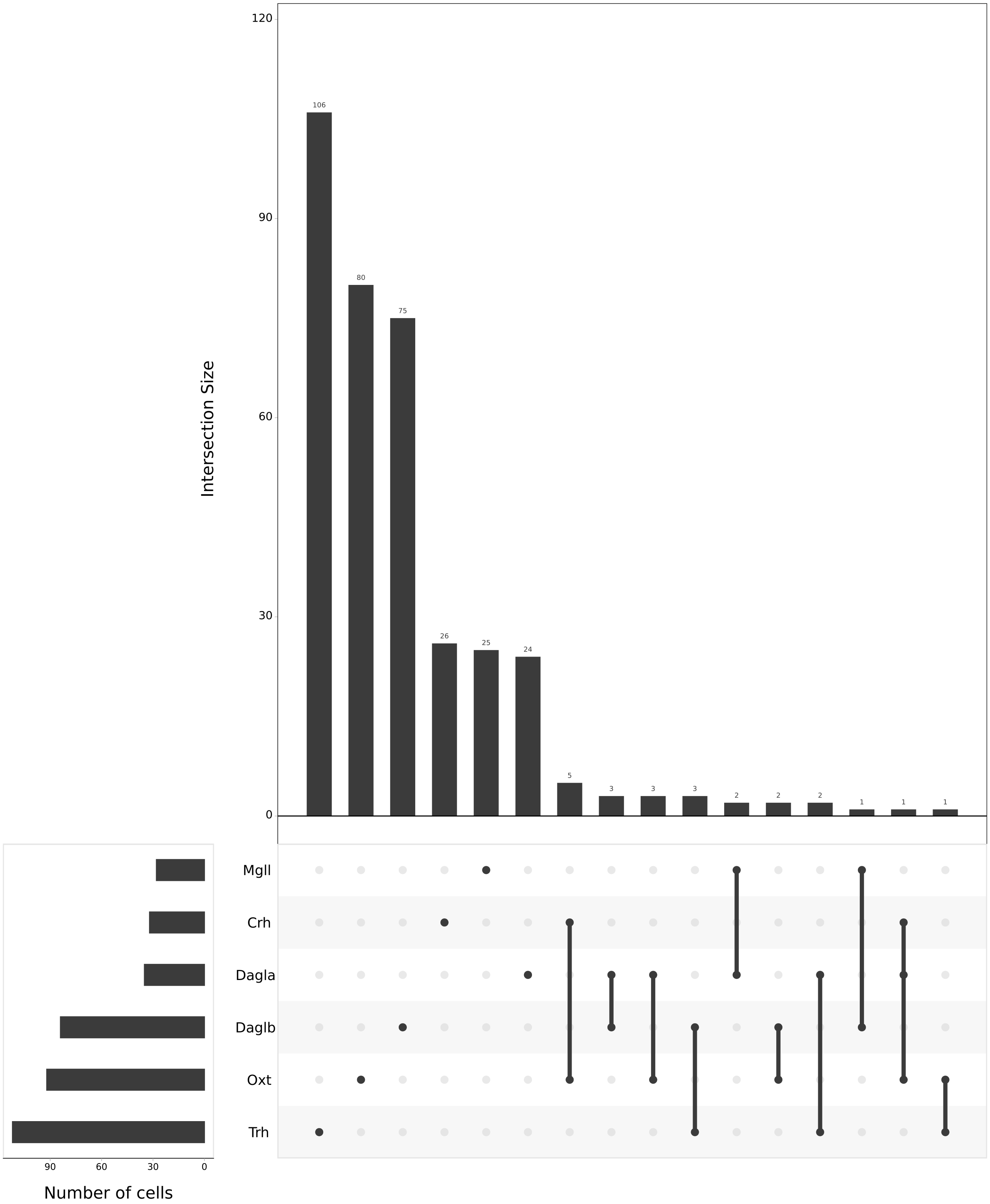
| | |
|---|---|
| Name | as.data.frame(...) |
| Number of rows | 3771 |
| Number of columns | 5 |
| Column type frequency: | |
| numeric | 5 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Cnr1 | 0 | 1 | 0.09 | 0.28 | 0 | 0 | 0 | 0 | 1 | |
| Gpr55 | 0 | 1 | 0.00 | 0.03 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.02 | 0.15 | 0 | 0 | 0 | 0 | 1 | |
| Crh | 0 | 1 | 0.01 | 0.09 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.03 | 0.17 | 0 | 0 | 0 | 0 | 1 | |

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Embryonic") |>
      select(
        c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conte
      )
  ),
  order.by = "freq",
  cutoff = 3,
```

```
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```



Intersection Size
120
90
60
30
0
106
80
75
26
25
24
5
3
3
3
2
2
2
1
1
1
Mgll
Crh
Dagla
Daglb
Oxt
Trh
90
60
30
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Embryonic") |>
      select(
        c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conte
      )
  ))
```

Table 3: Data summary

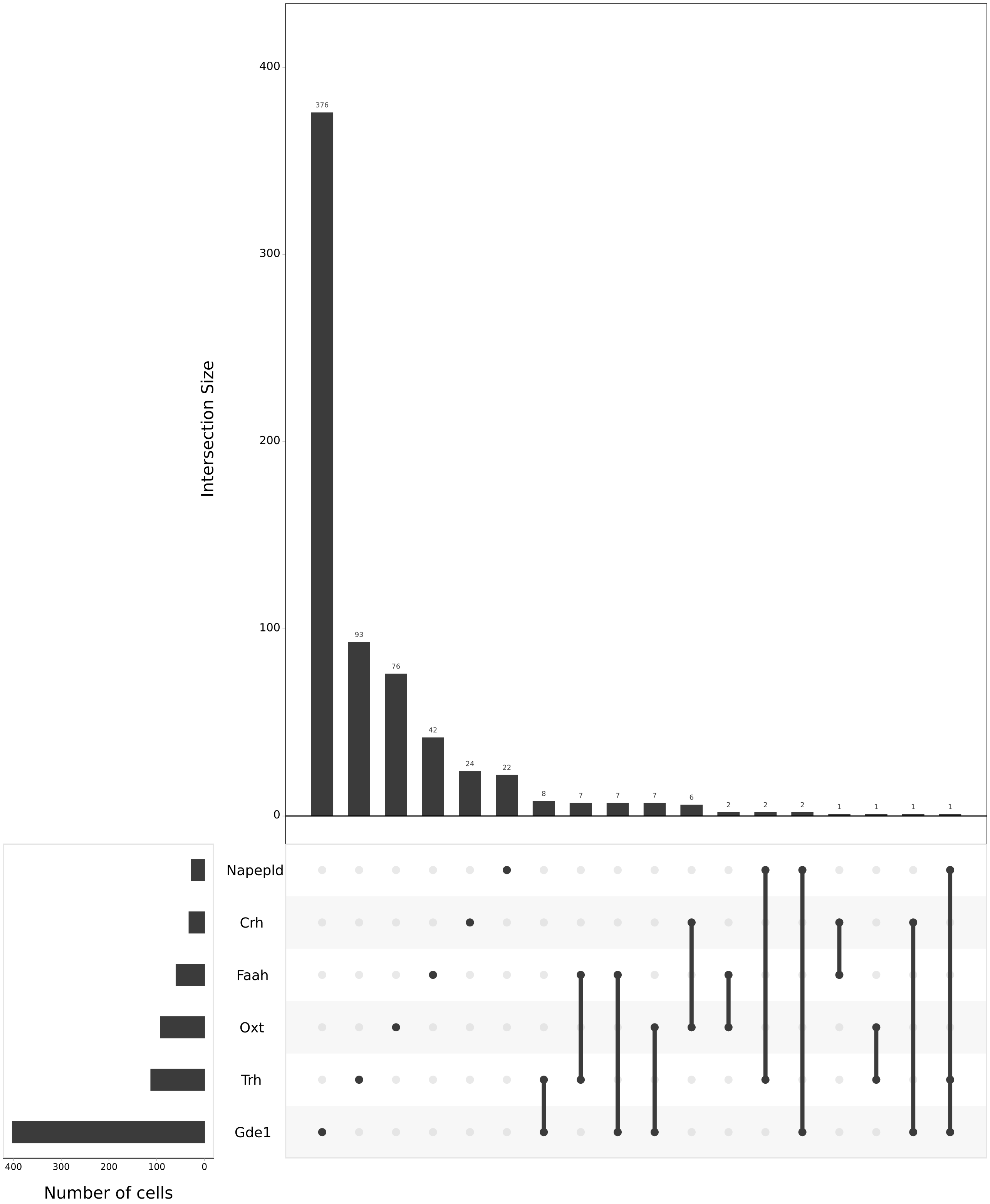
| | |
|---|---|
| Name | as.data.frame(...) |
| Number of rows | 3771 |
| Number of columns | 6 |
| Column type frequency: | |
| numeric | 6 |
| Group variables | None |

### Variable type: numeric

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Dagla | 0 | 1 | 0.01 | 0.10 | 0 | 0 | 0 | 0 | 1 | |
| Daglb | 0 | 1 | 0.02 | 0.15 | 0 | 0 | 0 | 0 | 1 | |
| Mgll | 0 | 1 | 0.01 | 0.09 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.02 | 0.15 | 0 | 0 | 0 | 0 | 1 | |
| Crh | 0 | 1 | 0.01 | 0.09 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.03 | 0.17 | 0 | 0 | 0 | 0 | 1 | |

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Embryonic") |>
      select(
        c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(cont
      )
  ),
  order.by = "freq",
```

```
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
400
300
200
100
0
376
93
76
42
24
22
8
7
7
7
6
2
2
2
1
1
1
1
Napepld
Crh
Faah
Oxt
Trh
Gde1
400
300
200
100
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Embryonic") |>
      select(
        c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(cont
      )
  ))
```

Table 5: Data summary

| | |
|---|---|
| Name | as.data.frame(…) |
| Number of rows | 3771 |
| Number of columns | 6 |
| Column type frequency: | |
| numeric | 6 |
| Group variables | None |

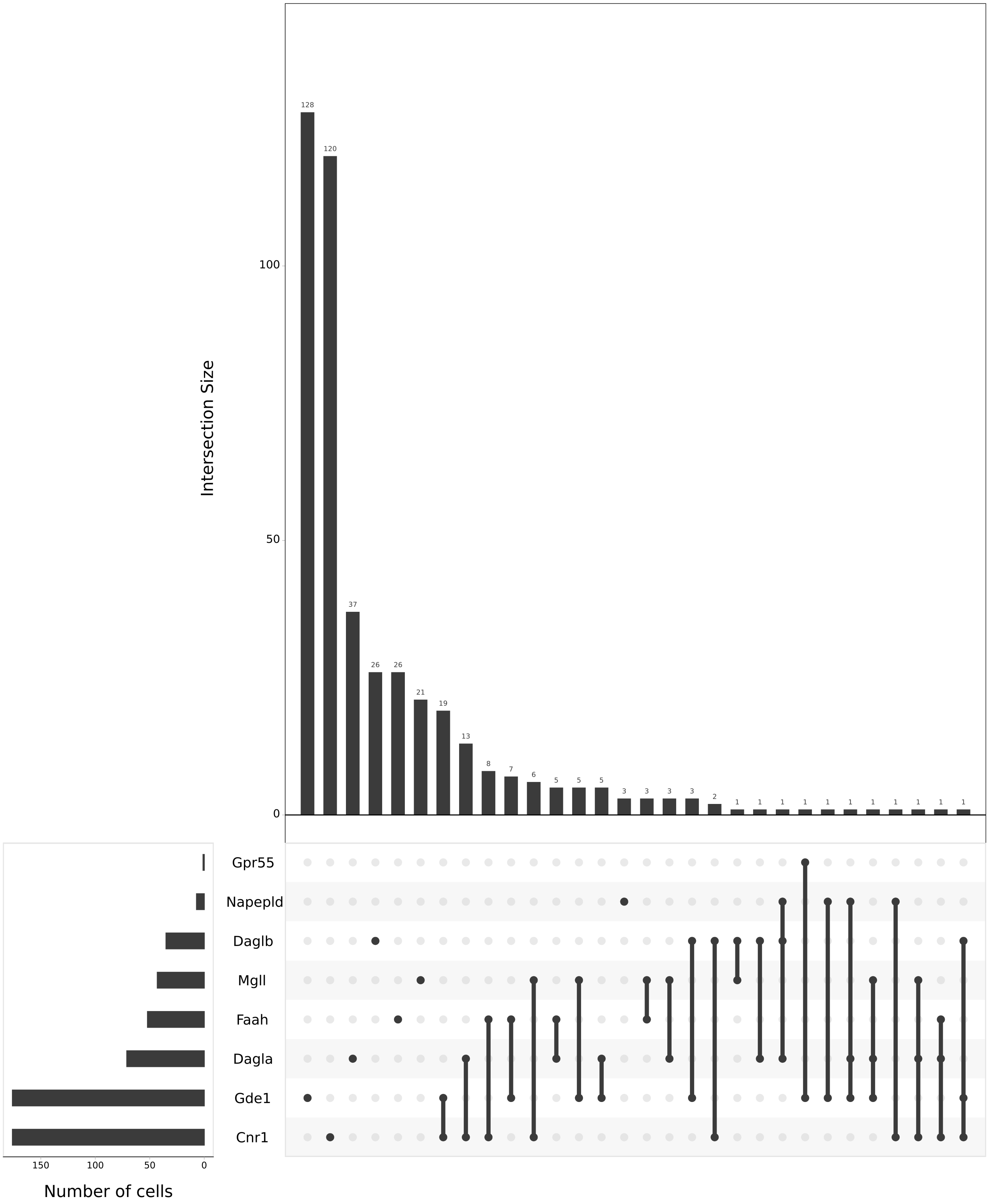
**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Napepld | 0 | 1 | 0.01 | 0.08 | 0 | 0 | 0 | 0 | 1 | |
| Gde1 | 0 | 1 | 0.11 | 0.31 | 0 | 0 | 0 | 0 | 1 | |
| Faah | 0 | 1 | 0.02 | 0.12 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.02 | 0.15 | 0 | 0 | 0 | 0 | 1 | |
| Crh | 0 | 1 | 0.01 | 0.09 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.03 | 0.17 | 0 | 0 | 0 | 0 | 1 | |

### Neonatal

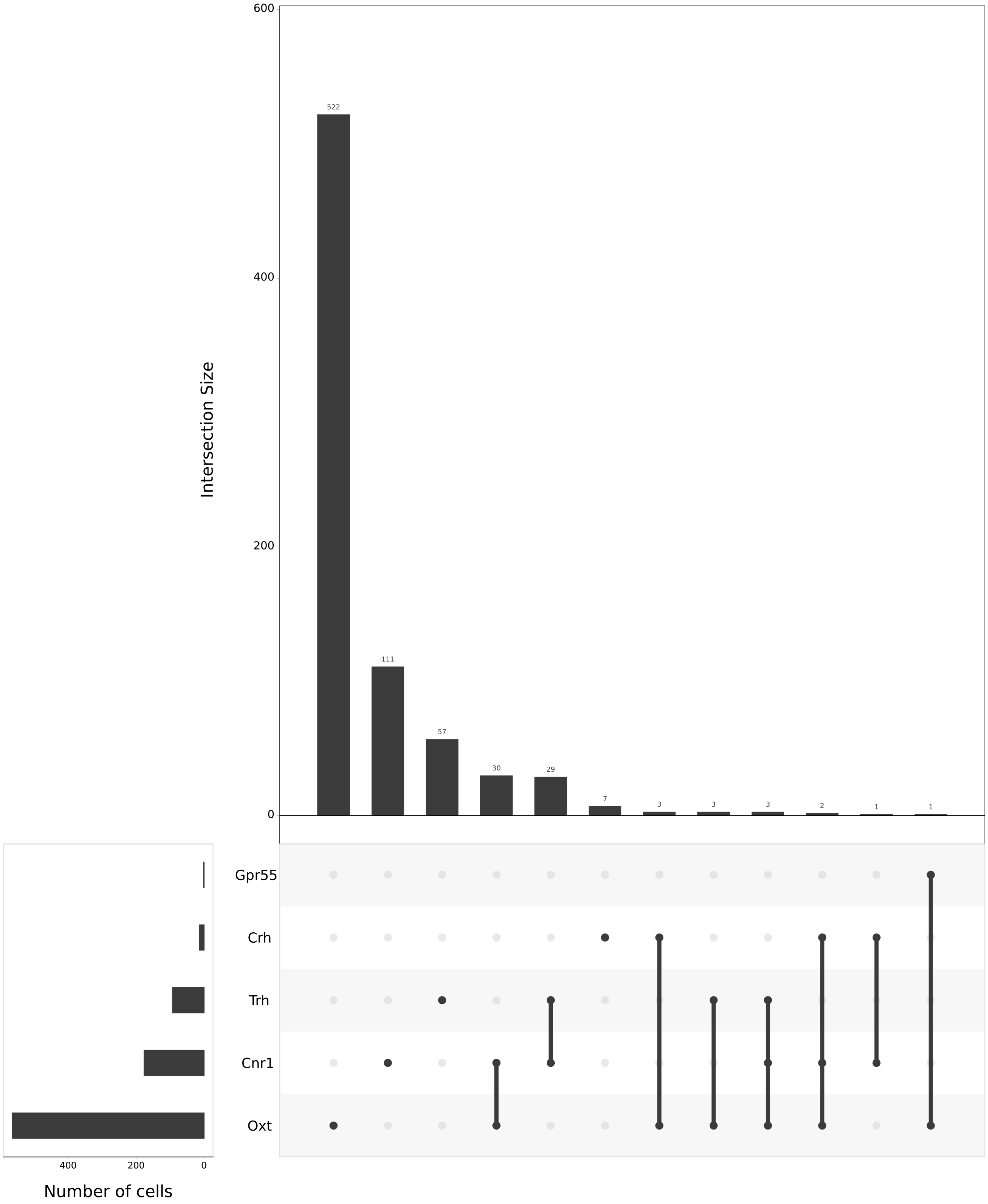
```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Neonatal") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx)]
      )
```

```r
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```



Intersection Size
100
50
0
128
120
37
26
26
21
19
13
8
7
6
5
5
5
3
3
3
3
2
1
1
1
1
1
1
1
1
1
1
1
Gpr55
Napepld
Daglb
Mgll
Faah
Dagla
Gde1
Cnr1
150
100
50
0
Number of cells

```r
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Neonatal") |>
      select(
        c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conten
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


600
400
200
0
Intersection Size
522
111
57
30
29
7
3
3
3
2
1
1
Gpr55
Crh
Trh
Cnr1
Oxt
400
200
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Neonatal") |>
      select(
        c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conter
      )
  ))
```

Table 7: Data summary

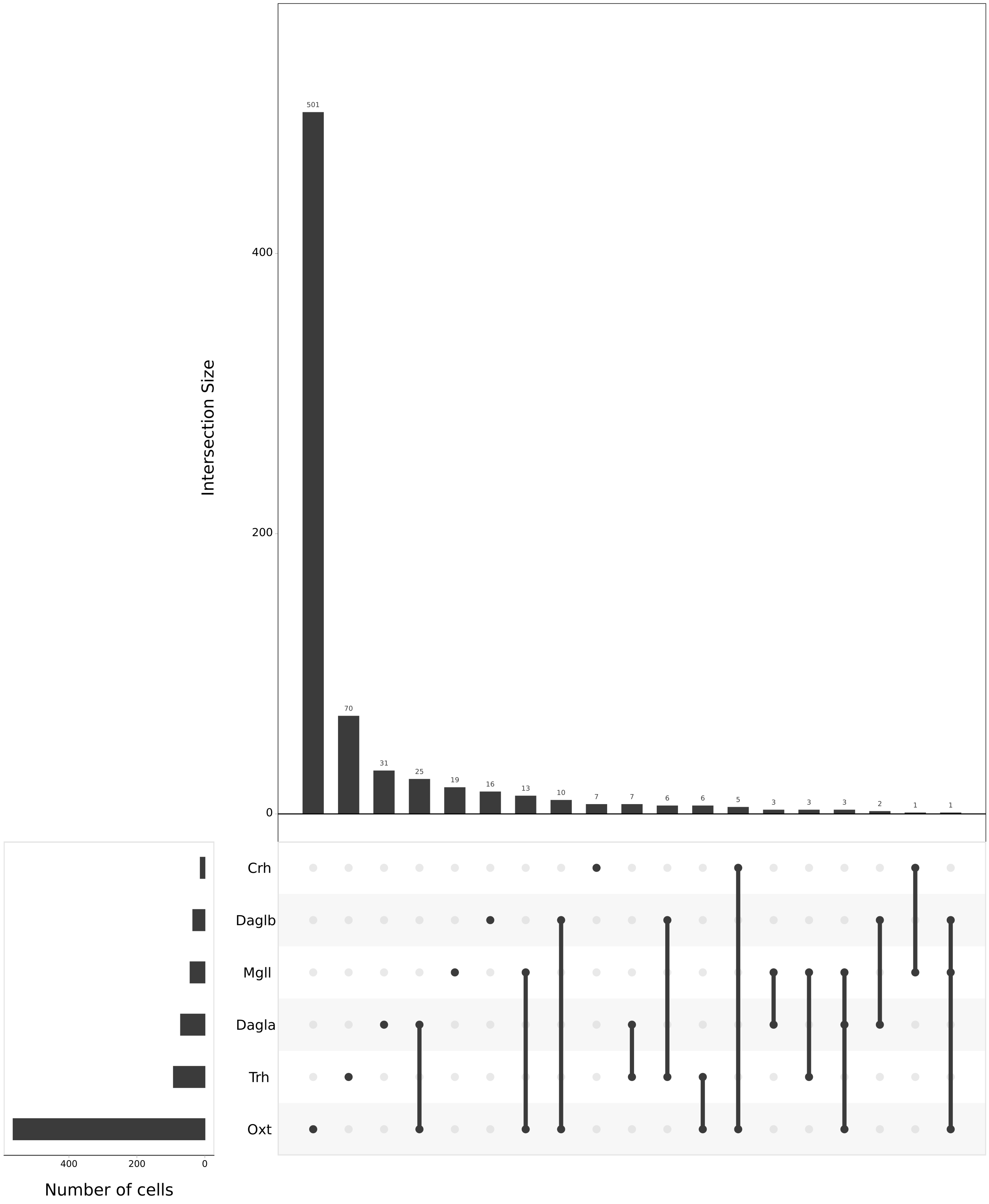
| | |
|---|---|
| Name | as.data.frame(…) |
| Number of rows | 1456 |
| Number of columns | 5 |
| Column type frequency: | |
| numeric | 5 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Cnr1 | 0 | 1 | 0.12 | 0.33 | 0 | 0 | 0 | 0 | 1 | |
| Gpr55 | 0 | 1 | 0.00 | 0.03 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.39 | 0.49 | 0 | 0 | 0 | 1 | 1 | |
| Crh | 0 | 1 | 0.01 | 0.09 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.06 | 0.24 | 0 | 0 | 0 | 0 | 1 | |

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Neonatal") |>
      select(
        c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conte
      )
  ),
  order.by = "freq",
  cutoff = 3,
```

```
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


501
70
31
25
19
16
13
10
7
7
6
6
5
3
3
3
2
1
1
400
200
0
Intersection Size
Crh
Daglb
Mgll
Dagla
Trh
Oxt
400
200
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Neonatal") |>
      select(
        c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conte
      )
  ))
```

Table 9: Data summary

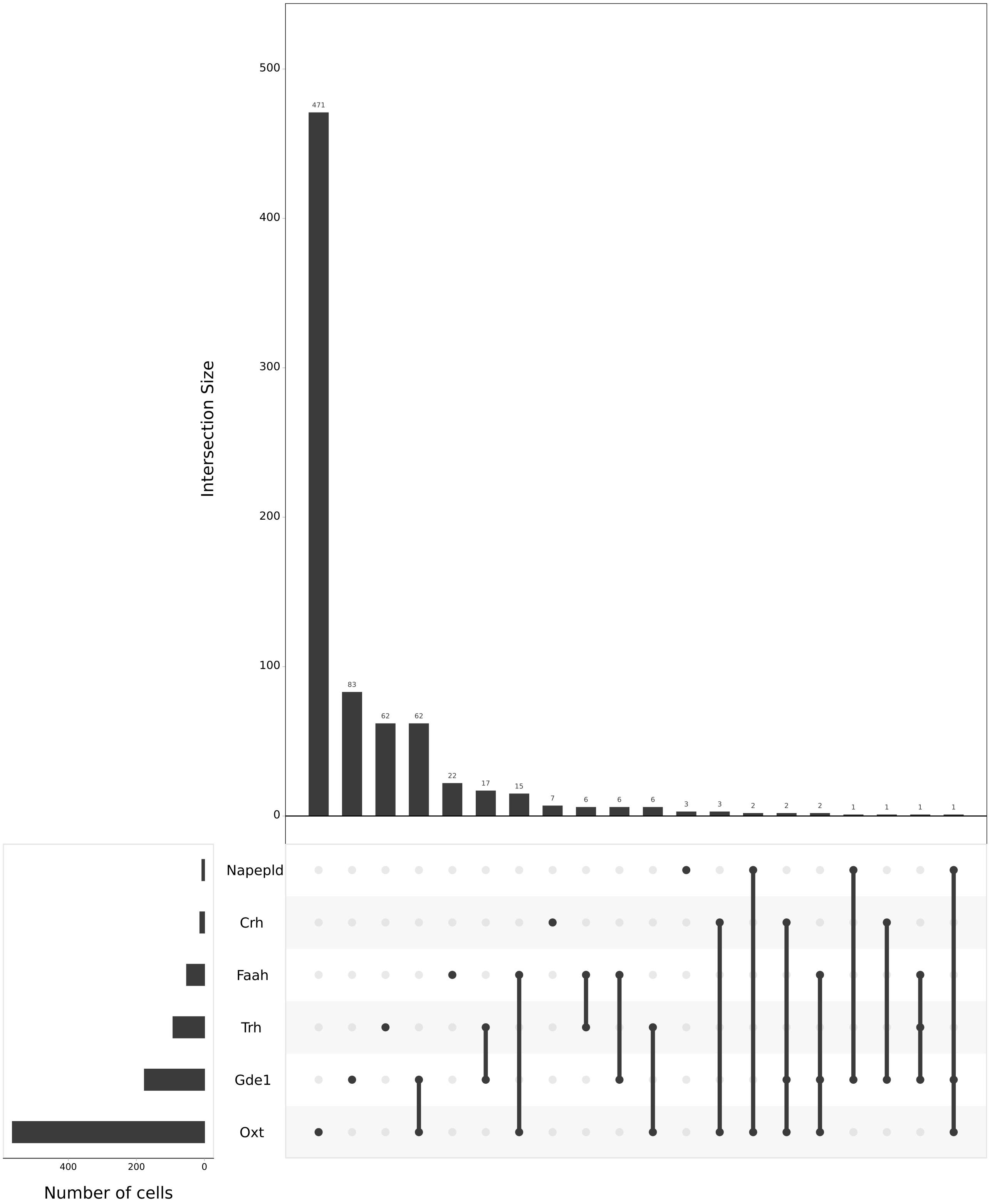
| | |
|---|---|
| Name | as.data.frame(...) |
| Number of rows | 1456 |
| Number of columns | 6 |
| Column type frequency: | |
| numeric | 6 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Dagla | 0 | 1 | 0.05 | 0.22 | 0 | 0 | 0 | 0 | 1 | |
| Daglb | 0 | 1 | 0.02 | 0.15 | 0 | 0 | 0 | 0 | 1 | |
| Mgll | 0 | 1 | 0.03 | 0.17 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.39 | 0.49 | 0 | 0 | 0 | 1 | 1 | |
| Crh | 0 | 1 | 0.01 | 0.09 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.06 | 0.24 | 0 | 0 | 0 | 0 | 1 | |

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Neonatal") |>
      select(
        c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(cont
      )
  ),
  order.by = "freq",
```

```
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
500
400
300
200
100
0
471
83
62
62
22
17
15
7
6
6
6
3
3
2
2
2
1
1
1
1
Napepld
Crh
Faah
Trh
Gde1
Oxt
400
200
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Neonatal") |>
      select(
        c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(cont
      )
  ))
```

Table 11: Data summary

| | |
|---|---|
| Name | as.data.frame(…) |
| Number of rows | 1456 |
| Number of columns | 6 |
| Column type frequency: | |
| numeric | 6 |
| Group variables | None |

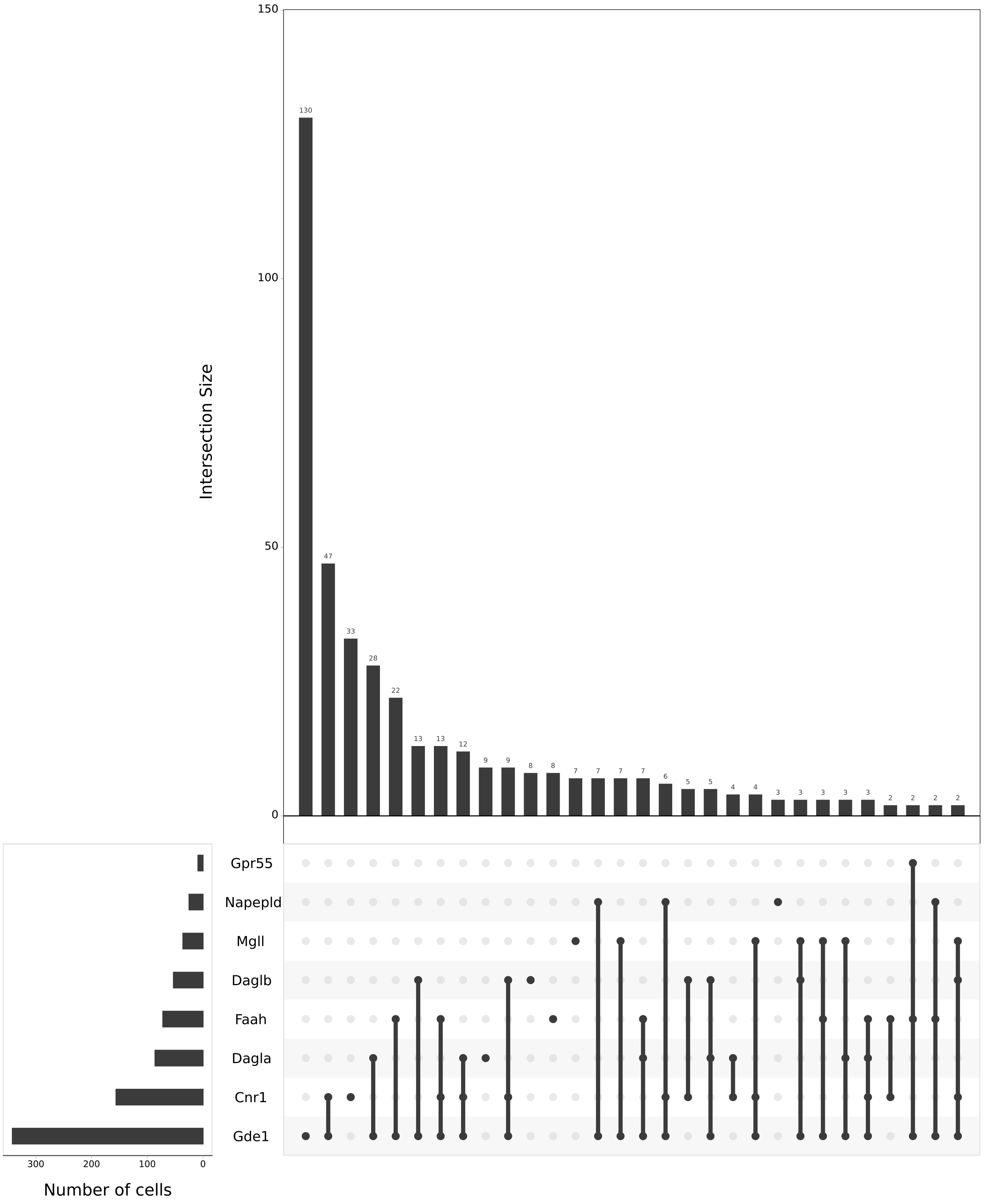
**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Napepld | 0 | 1 | 0.00 | 0.07 | 0 | 0 | 0 | 0 | 1 | |
| Gde1 | 0 | 1 | 0.12 | 0.33 | 0 | 0 | 0 | 0 | 1 | |
| Faah | 0 | 1 | 0.04 | 0.19 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.39 | 0.49 | 0 | 0 | 0 | 1 | 1 | |
| Crh | 0 | 1 | 0.01 | 0.09 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.06 | 0.24 | 0 | 0 | 0 | 0 | 1 | |

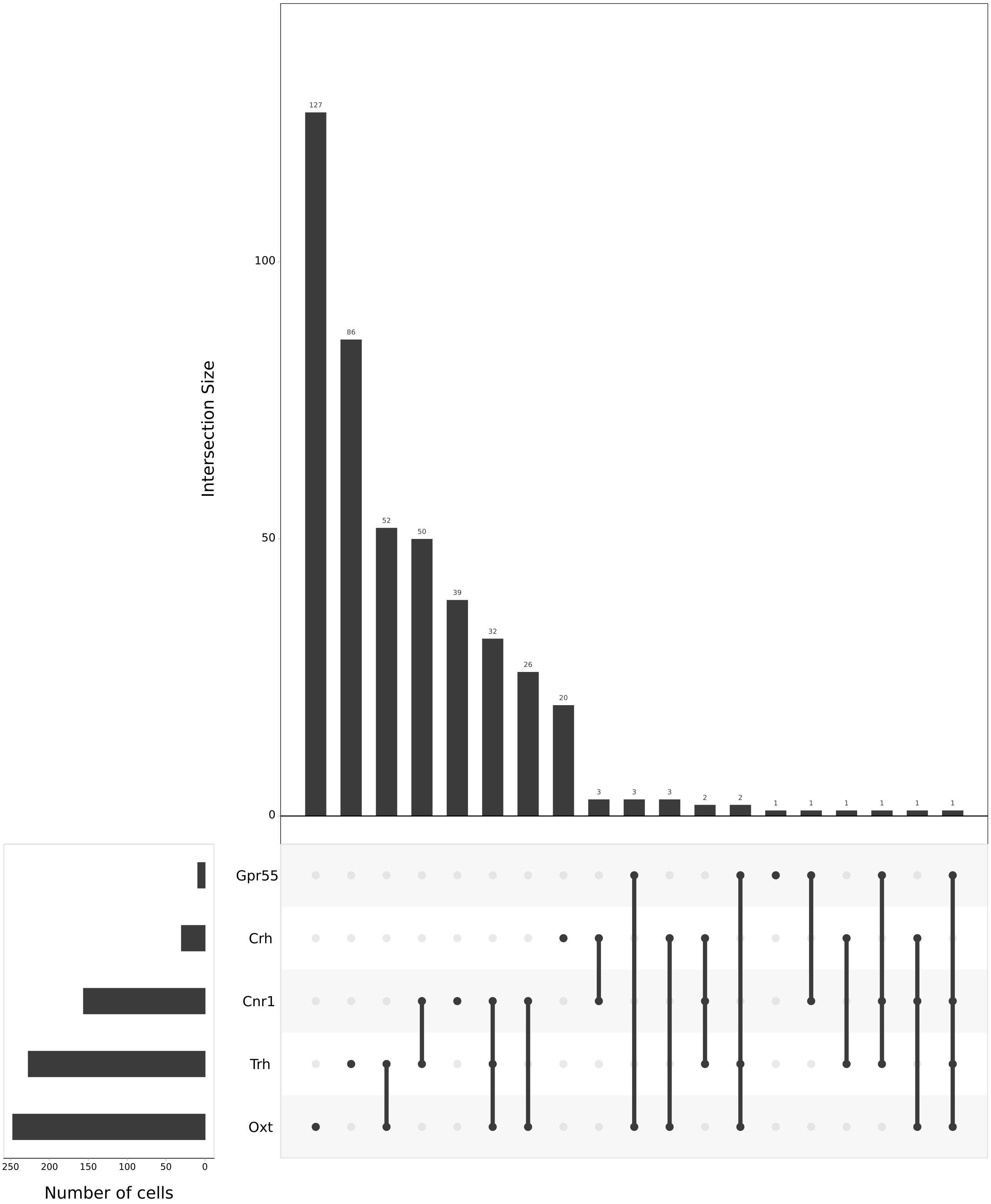
### Pubertal

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Pubertal") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx)]
      )
```

```
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c(cnbn) %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
150
100
50
0
130
47
33
28
22
13
13
12
9
9
8
8
7
7
7
7
6
5
5
4
4
3
3
3
3
3
2
2
2
2
Gpr55
Napepld
Mgll
Daglb
Faah
Dagla
Cnr1
Gde1
300
200
100
0
Number of cells

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Pubertal") |>
      select(
        c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conten
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```



Intersection Size
100
50
0
127
86
52
50
39
32
26
20
3
3
3
2
2
1
1
1
1
1
1
Gpr55
Crh
Cnr1
Trh
Oxt
250
200
150
100
50
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Pubertal") |>
      select(
        c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conter
      )
  ))
```

Table 13: Data summary

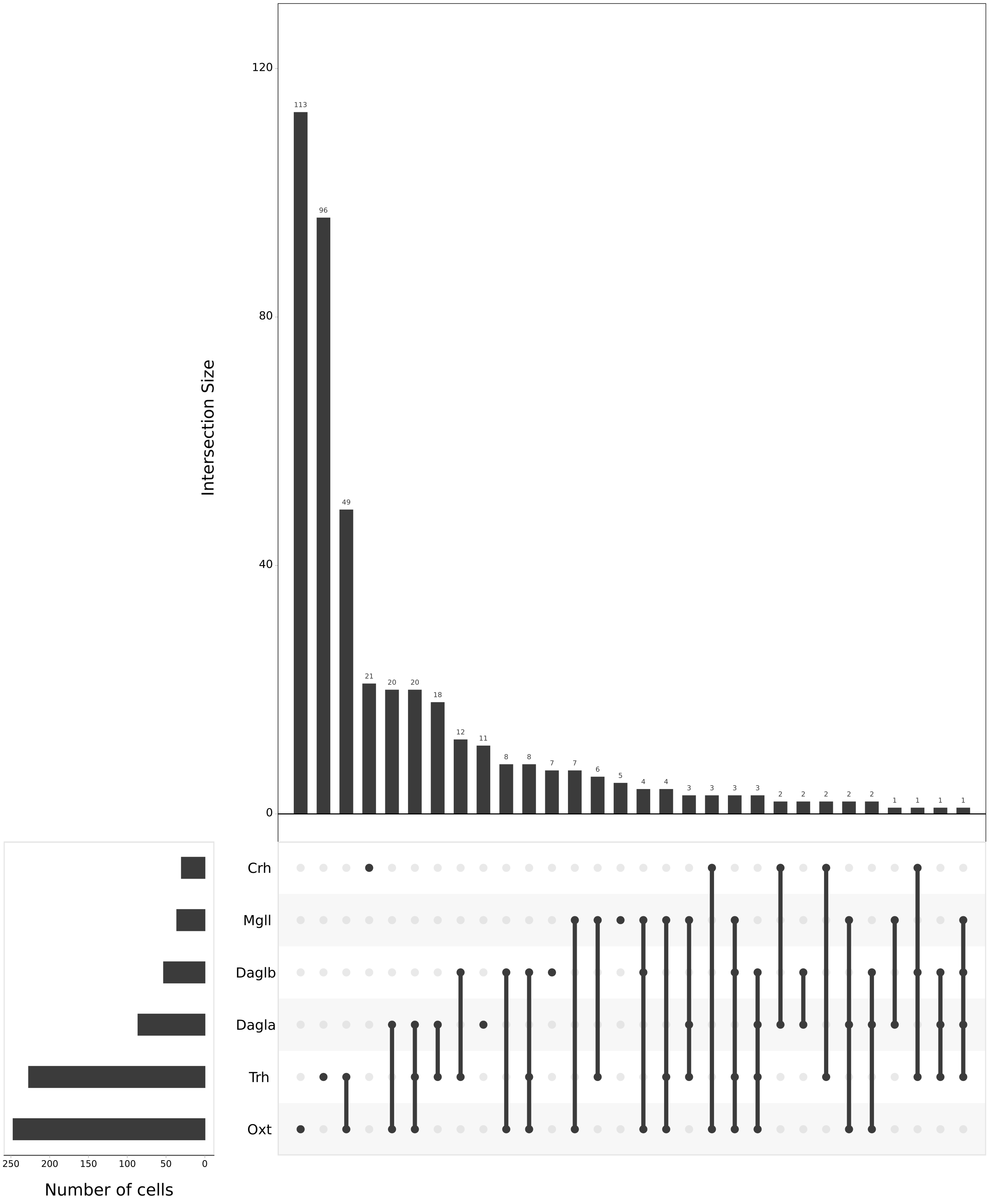
| | |
|---|---|
| Name | as.data.frame(…) |
| Number of rows | 641 |
| Number of columns | 5 |
| Column type frequency: | |
| numeric | 5 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Cnr1 | 0 | 1 | 0.24 | 0.43 | 0 | 0 | 0 | 0 | 1 | |
| Gpr55 | 0 | 1 | 0.01 | 0.12 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.39 | 0.49 | 0 | 0 | 0 | 1 | 1 | |
| Crh | 0 | 1 | 0.05 | 0.21 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.35 | 0.48 | 0 | 0 | 0 | 1 | 1 | |

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Pubertal") |>
      select(
        c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conte
      )
  ),
  order.by = "freq",
  cutoff = 3,
```

```r
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
120
80
40
0
113
96
49
21
20
20
18
12
11
8
8
7
7
6
5
4
4
3
3
3
3
2
2
2
2
2
1
1
1
1
Crh
Mgll
Daglb
Dagla
Trh
Oxt
250
200
150
100
50
0
Number of cells

```r
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Pubertal") |>
      select(
        c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conte
      )
  ))
```

Table 15: Data summary

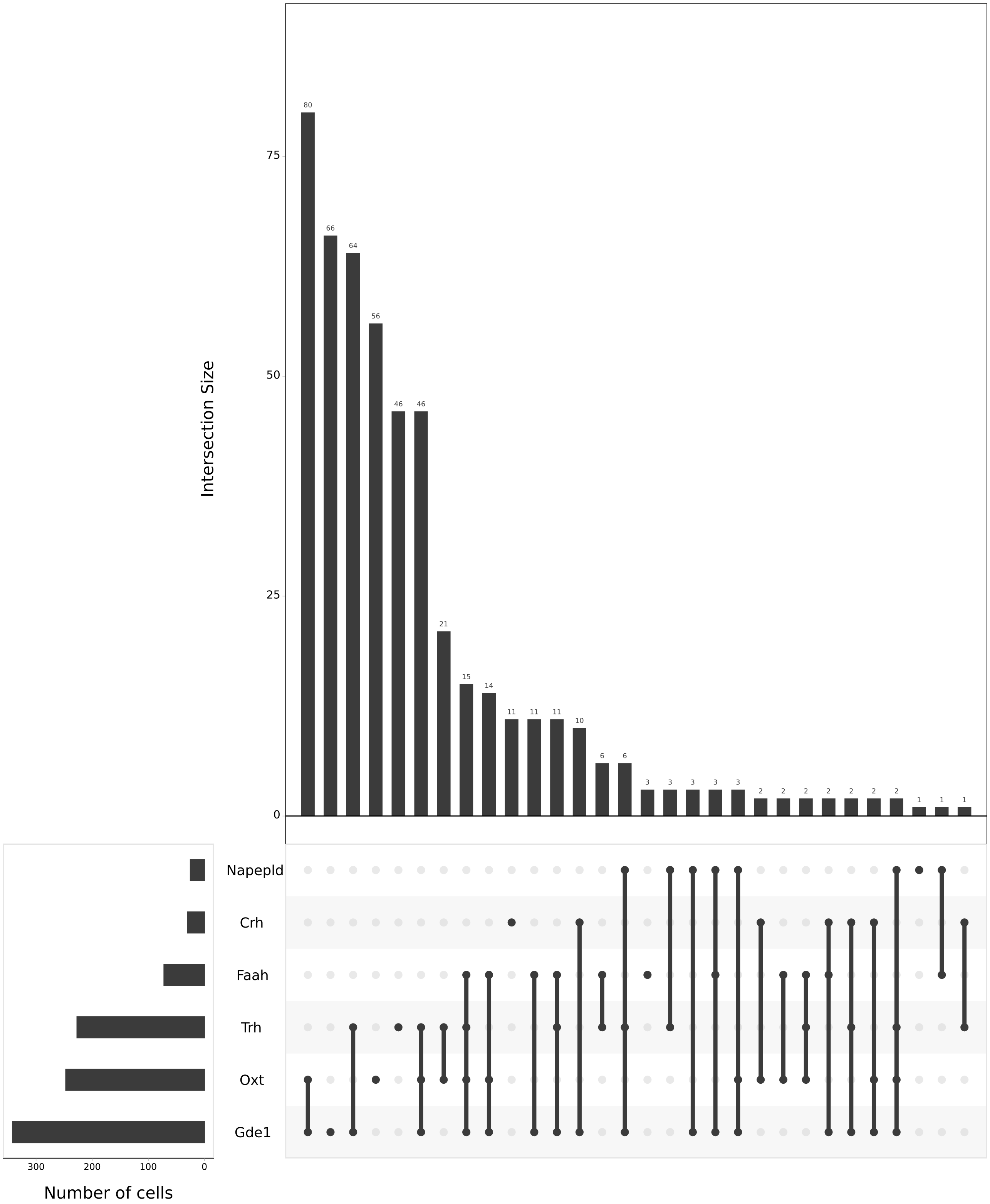
| | |
|---|---|
| Name | as.data.frame(...) |
| Number of rows | 641 |
| Number of columns | 6 |
| Column type frequency: | |
| numeric | 6 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Dagla | 0 | 1 | 0.13 | 0.34 | 0 | 0 | 0 | 0 | 1 | |
| Daglb | 0 | 1 | 0.08 | 0.28 | 0 | 0 | 0 | 0 | 1 | |
| Mgll | 0 | 1 | 0.06 | 0.23 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.39 | 0.49 | 0 | 0 | 0 | 1 | 1 | |
| Crh | 0 | 1 | 0.05 | 0.21 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.35 | 0.48 | 0 | 0 | 0 | 1 | 1 | |

```r
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Pubertal") |>
      select(
        c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(cont
      )
  ),
  order.by = "freq",
```

```r
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
0
25
50
75
80
66
64
56
46
46
21
15
14
11
11
11
10
6
6
3
3
3
3
3
2
2
2
2
2
2
2
1
1
1
Napepld
Crh
Faah
Trh
Oxt
Gde1
300
200
100
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Pubertal") |>
      select(
        c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(cont
      )
  ))
```

Table 17: Data summary

| | |
|---|---|
| Name | as.data.frame(…) |
| Number of rows | 641 |
| Number of columns | 6 |
| Column type frequency: | |
| numeric | 6 |
| Group variables | None |

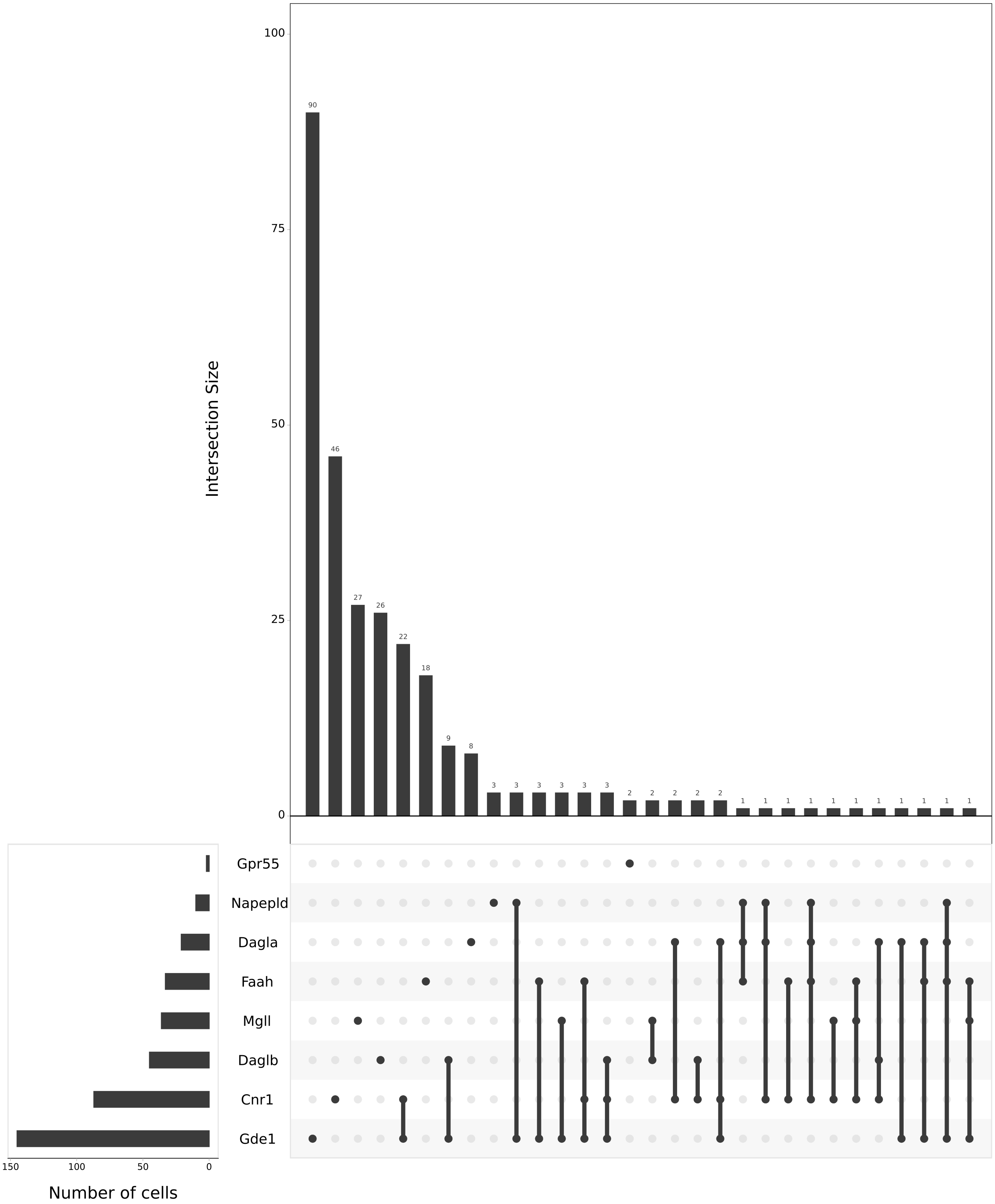
**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Napepld | 0 | 1 | 0.04 | 0.19 | 0 | 0 | 0 | 0 | 1 | |
| Gde1 | 0 | 1 | 0.53 | 0.50 | 0 | 0 | 1 | 1 | 1 | |
| Faah | 0 | 1 | 0.11 | 0.32 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.39 | 0.49 | 0 | 0 | 0 | 1 | 1 | |
| Crh | 0 | 1 | 0.05 | 0.21 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.35 | 0.48 | 0 | 0 | 0 | 1 | 1 | |

### Adult

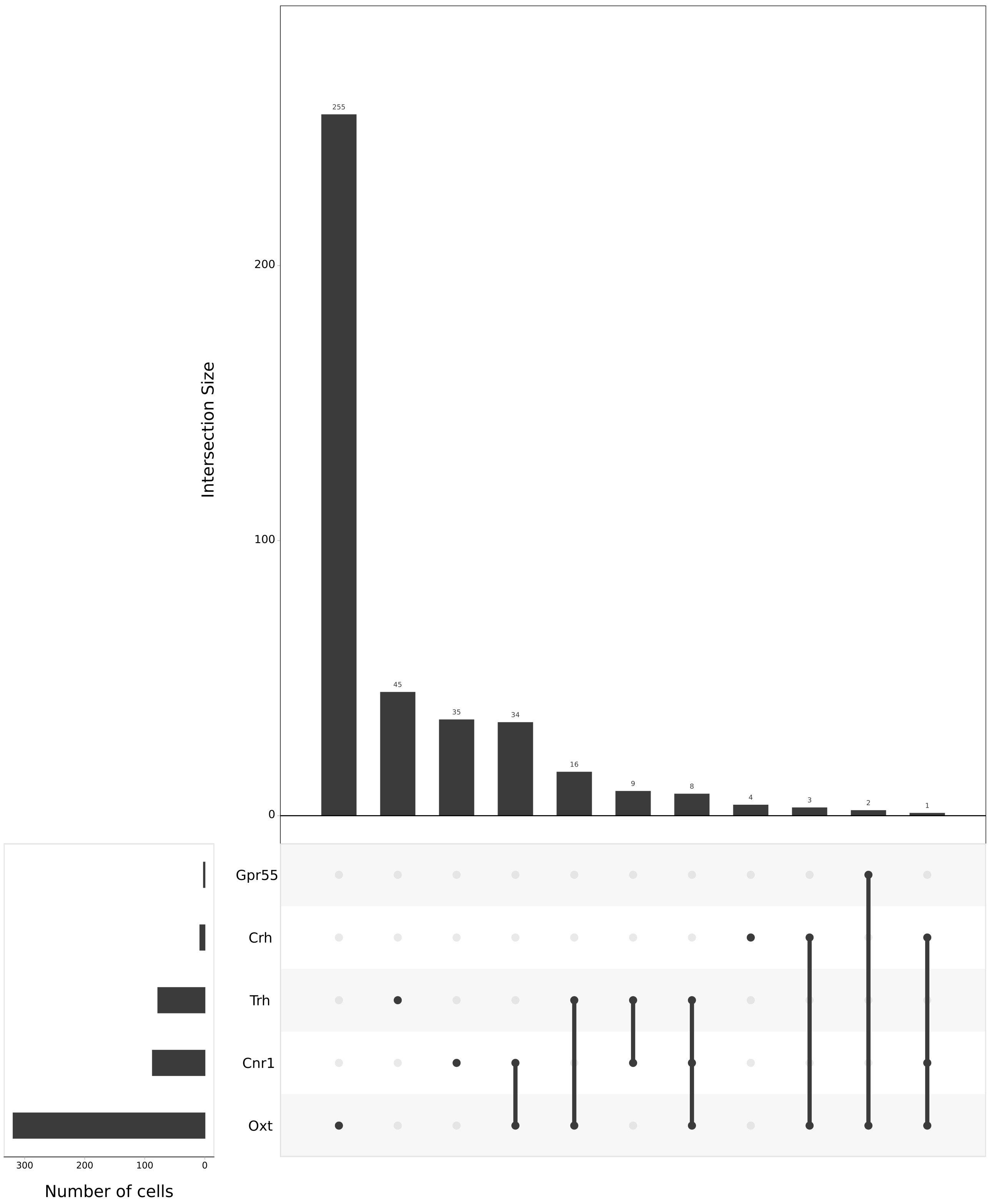
```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Adult") |>
      select(
        c(cnbn) %>% .[. %in% colnames(content_sbs_mtx)]
      )
```

```
    ),
    order.by = "freq",
    cutoff = 3,
    sets.x.label = "Number of cells",
    number.angles = 0,
    point.size = 3.5, line.size = 2,
    text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
    nsets = 30,
    nintersects = 30,
    sets = c(cnbn) %>%
      .[. %in% colnames(content_sbs_mtx)],
    empty.intersections = NULL
)
```



Intersection Size
100
75
50
25
0
90
46
27
26
22
18
9
8
3
3
3
3
3
3
2
2
2
2
2
1
1
1
1
1
1
1
1
1
1
1
Gpr55
Napepld
Dagla
Faah
Mgll
Daglb
Cnr1
Gde1
150
100
50
0
Number of cells

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Adult") |>
      select(
        c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conten
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```



Intersection Size
0
100
200
255
45
35
34
16
9
8
4
3
2
1
Gpr55
Crh
Trh
Cnr1
Oxt
300
200
100
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Adult") |>
      select(
        c("Cnr1", "Cnr2", "Gpr55", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conte
      )
  ))
```

Table 19: Data summary

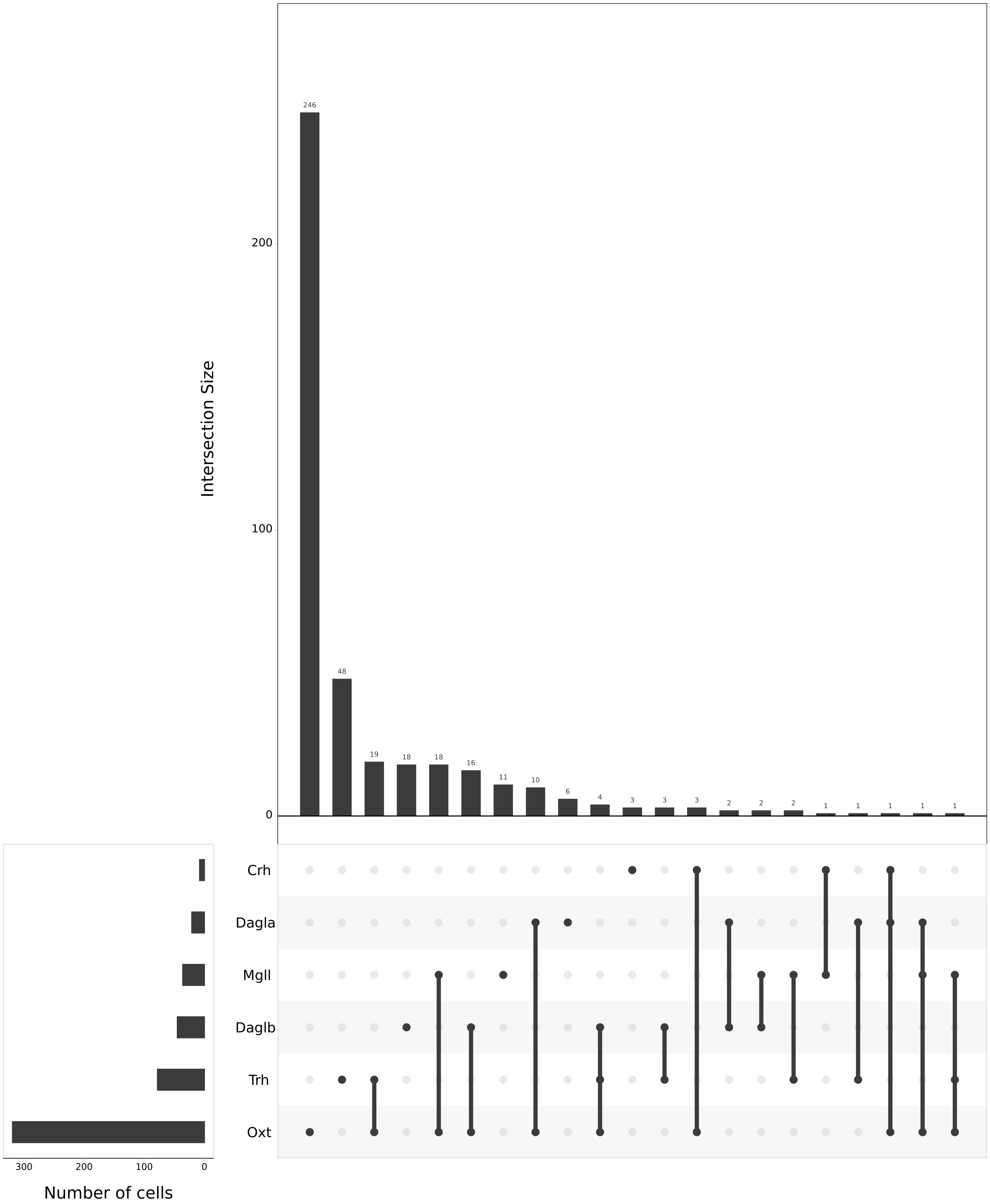
| | |
|---|---|
| Name | as.data.frame(…) |
| Number of rows | 801 |
| Number of columns | 5 |
| Column type frequency: | |
| numeric | 5 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Cnr1 | 0 | 1 | 0.11 | 0.31 | 0 | 0 | 0 | 0 | 1 | |
| Gpr55 | 0 | 1 | 0.00 | 0.05 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.40 | 0.49 | 0 | 0 | 0 | 1 | 1 | |
| Crh | 0 | 1 | 0.01 | 0.10 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.10 | 0.30 | 0 | 0 | 0 | 0 | 1 | |

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Adult") |>
      select(
        c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conte
      )
  ),
  order.by = "freq",
  cutoff = 3,
```

```
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
0
100
200
246
48
19
18
18
16
11
10
6
4
3
3
3
2
2
2
1
1
1
1
1
Crh
Dagla
Mgll
Daglb
Trh
Oxt
300
200
100
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Adult") |>
      select(
        c("Dagla", "Daglb", "Mgll", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(conte
      )
  ))
```

Table 21: Data summary

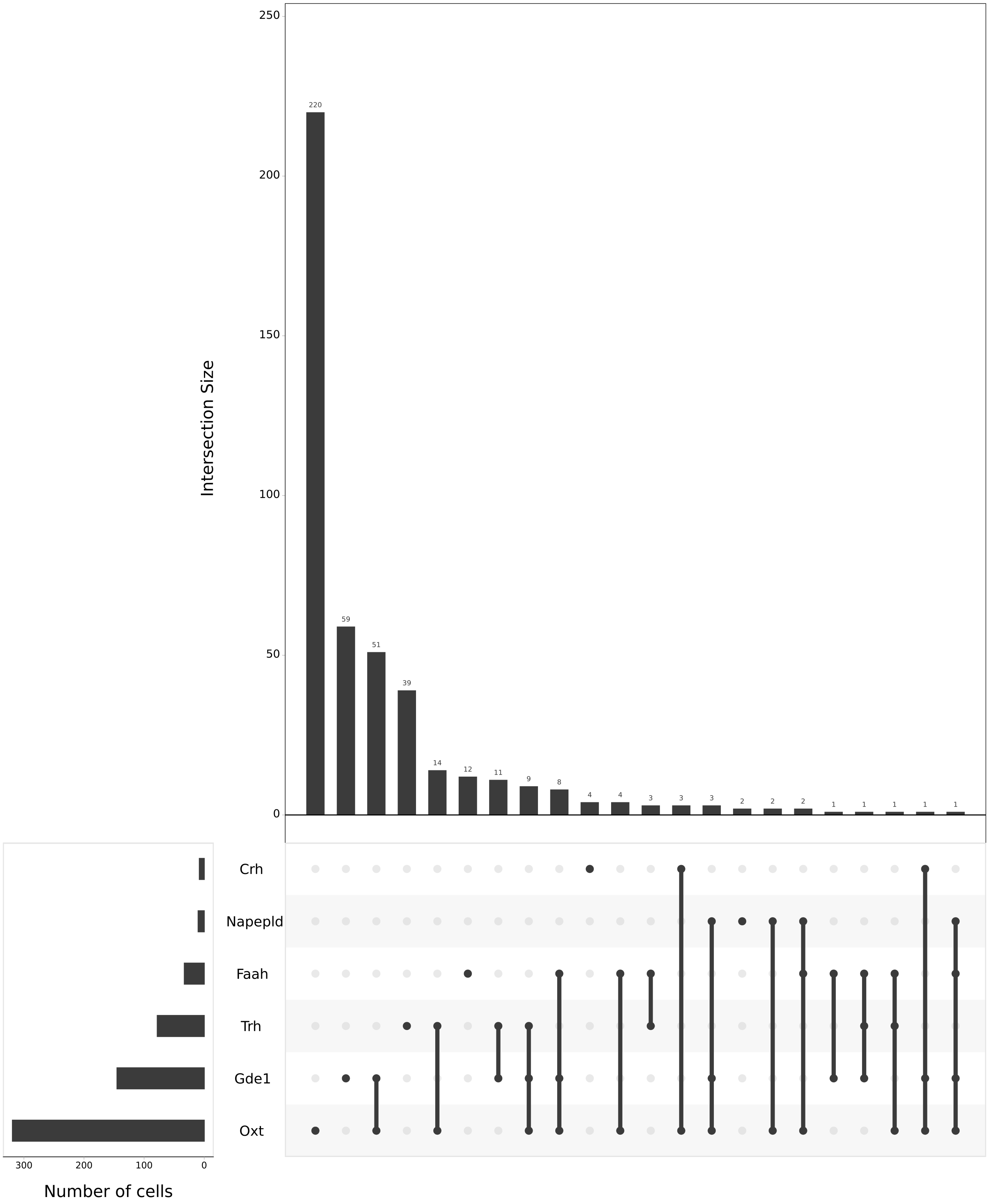
| | |
|---|---|
| Name | as.data.frame(…) |
| Number of rows | 801 |
| Number of columns | 6 |
| Column type frequency: | |
| numeric | 6 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Dagla | 0 | 1 | 0.03 | 0.16 | 0 | 0 | 0 | 0 | 1 | |
| Daglb | 0 | 1 | 0.06 | 0.23 | 0 | 0 | 0 | 0 | 1 | |
| Mgll | 0 | 1 | 0.04 | 0.21 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.40 | 0.49 | 0 | 0 | 0 | 1 | 1 | |
| Crh | 0 | 1 | 0.01 | 0.10 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.10 | 0.30 | 0 | 0 | 0 | 0 | 1 | |

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Adult") |>
      select(
        c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(cont
      )
  ),
  order.by = "freq",
```

```
    cutoff = 3,
    sets.x.label = "Number of cells",
    number.angles = 0,
    point.size = 3.5, line.size = 2,
    text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
    nsets = 30,
    nintersects = 30,
    sets = c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>%
      .[. %in% colnames(content_sbs_mtx)],
    empty.intersections = NULL
  )
```


Intersection Size
250
200
150
100
50
0
220
59
51
39
14
12
11
9
8
4
4
3
3
3
2
2
2
1
1
1
1
1
Crh
Napepld
Faah
Trh
Gde1
Oxt
300
200
100
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(stage == "Adult") |>
      select(
        c("Napepld", "Gde1", "Faah", "Oxt", "Crh", "Trh") %>% .[. %in% colnames(cont
      )
  ))
```

Table 23: Data summary

| | |
|---|---|
| Name | as.data.frame(...) |
| Number of rows | 801 |
| Number of columns | 6 |
| Column type frequency: | |
| numeric | 6 |
| Group variables | None |

### Variable type: numeric

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Napepld | 0 | 1 | 0.01 | 0.11 | 0 | 0 | 0 | 0 | 1 | |
| Gde1 | 0 | 1 | 0.18 | 0.39 | 0 | 0 | 0 | 0 | 1 | |
| Faah | 0 | 1 | 0.04 | 0.20 | 0 | 0 | 0 | 0 | 1 | |
| Oxt | 0 | 1 | 0.40 | 0.49 | 0 | 0 | 0 | 1 | 1 | |
| Crh | 0 | 1 | 0.01 | 0.10 | 0 | 0 | 0 | 0 | 1 | |
| Trh | 0 | 1 | 0.10 | 0.30 | 0 | 0 | 0 | 0 | 1 | |

### Contingency tables

```
rar2020.srt.pvn$age %>% forcats::fct_count()
```

```
# A tibble: 6 x 2
  f         n
  <fct> <int>
1 E15     317
```

```
2 E17      593
3 P00      350
4 P02      362
5 P10      425
6 P23       67
```

```
rar2020.srt.pvn$stage %>% forcats::fct_count()
```

```
# A tibble: 4 x 2
  f              n
  <fct>      <int>
1 Embryonic    910
2 Neonatal     712
3 Pubertal     425
4 Adult         67
```

```
srt$age %>% forcats::fct_count()
```

```
# A tibble: 12 x 2
   f         n
   <fct> <int>
 1 E10      22
 2 E11     168
 3 E12     374
 4 E13     290
 5 E14     540
 6 E15     444
 7 E16     655
 8 E18     368
 9 P14     216
10 P4      199
11 P45     734
12 P8      545
```

```
srt$stage %>% forcats::fct_count()
```

```
# A tibble: 4 x 2
  f             n
  <fct>     <int>
1 Embryonic  2861
2 Neonatal    744
3 Pubertal    216
4 Adult       734
```

## Load selected astrocytes data from Lopez JP et al (2021)

```
anndata <- sc$read(here(
  "../1_heteroAstrocytes/PRJNA679294/data/",
  "class_cello/PRJNA679294-whole_dataset-0.001-cello_annotation.h5ad"
))
```

### Convert adata object to R AnnDataR6 object.

```
adata <- py_to_r(anndata)
class(adata)
```

```
[1] "AnnDataR6" "R6"
```

```
class(adata$X)
```

```
[1] "dgCMatrix"
attr(,"package")
[1] "Matrix"
```

```
adata
```

```
AnnData object with n_obs × n_vars = 9572 × 22835
    obs: 'nCount_RAW', 'nFeature_RAW', 'nCount_RNA', 'nFeature_RNA', 'orig.ident', 'nF
    var: 'vst.mean', 'vst.variance', 'vst.variance.expected', 'vst.variance.standardiz
    uns: 'k_tree_colors', 'name', 'ora_celltype_colors'
    obsm: 'X_pacmap', 'X_pca', 'X_umap', 'ora_estimate', 'ora_pvals'
```

```r
expr_mtx <- t(as.matrix(adata$raw$X))
colnames(expr_mtx) <- rownames(adata$X)
rownames(expr_mtx) <- adata$var_names
srt <- CreateSeuratObject(
  expr_mtx,
  assay = "RNA",
  project = "individual_hypothalamic_nuclei_astrocytes_evaluation_dataset",
  meta.data = as.data.frame(adata$obs)
)

Idents(srt) <- "ora_celltype"
srt <- subset(srt, idents = c("Astrocytes"))

Idents(srt) <- "libname"

print(srt)
```

```
An object of class Seurat
22835 features across 2838 samples within 1 assay
Active assay: RNA (22835 features, 0 variable features)
 1 layer present: counts
```

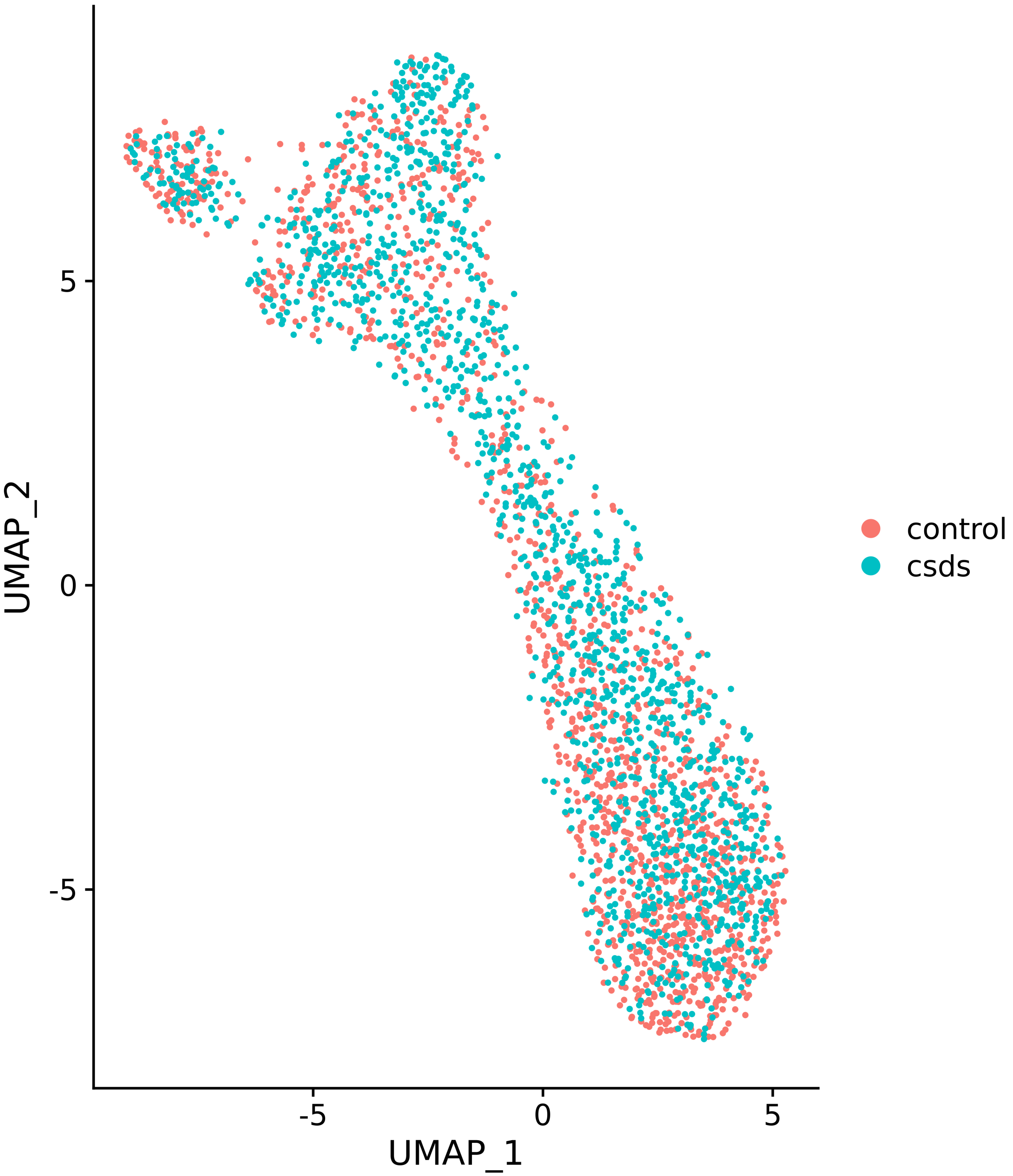
```r
rm(adata, anndata, expr_mtx)
invisible(gc())
```

```r
srt <- NormalizeData(srt)
srt <- FindVariableFeatures(srt, selection.method = "vst", nfeatures = 5000)
all.genes <- rownames(srt)
srt <- ScaleData(srt, features = c(VariableFeatures(srt), cnbn))
```

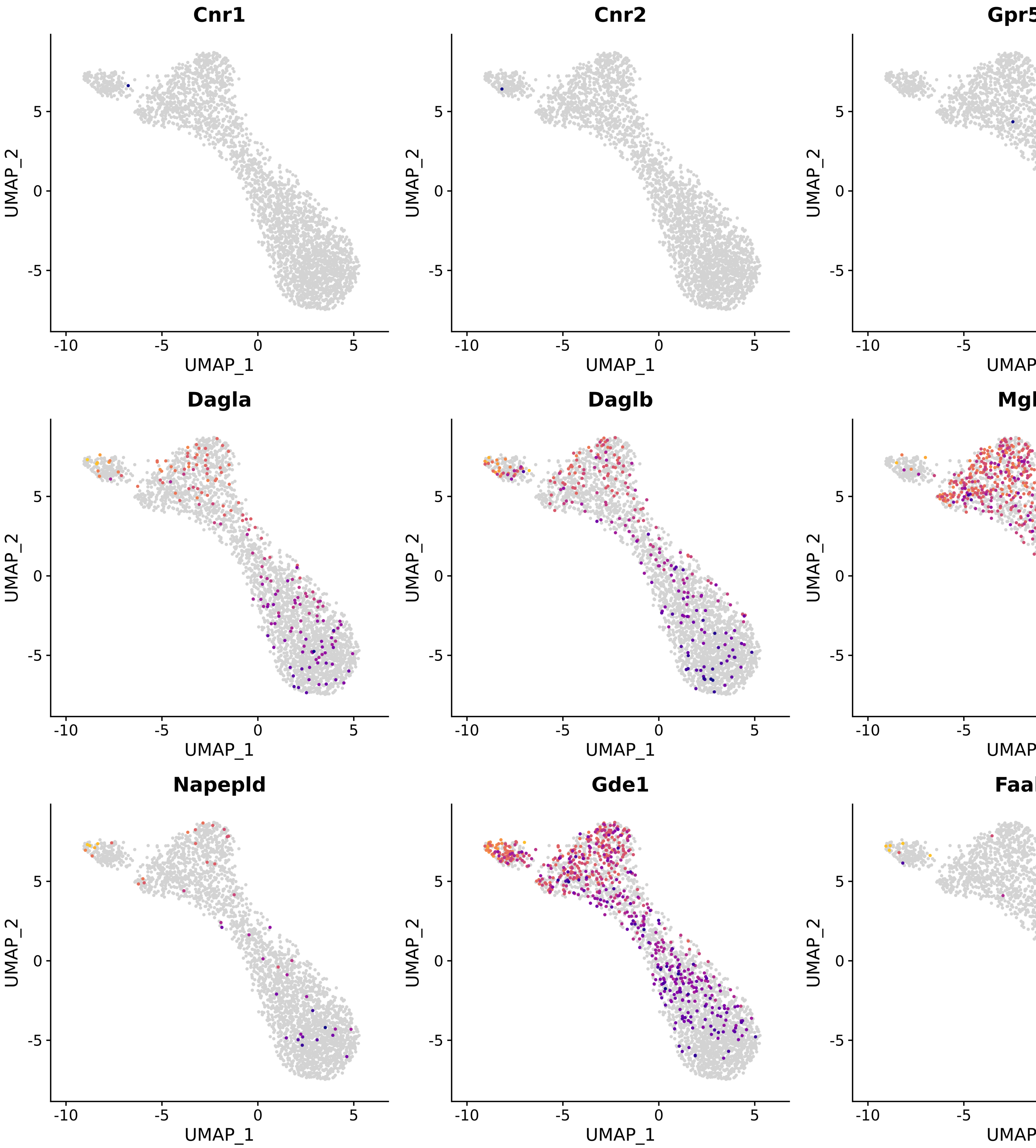
```r
srt <- RunPCA(srt, npcs = 50, verbose = FALSE)
srt <-
  srt |>
  FindNeighbors(
    dims = 1:40,
    k.param = 40,
    annoy.metric = "cosine",
    n.trees = 100,
```

```
    verbose = FALSE
  ) |>
  RunUMAP(
    dims = 1:50,
    reduction.name = "umap",
    reduction.key = "UMAP_",
    return.model = F,
    n.epochs = 1000L,
    n.neighbors = 50,
    min.dist = 0.5,
    metric = "cosine",
    seed.use = reseed,
    verbose = FALSE
  )
```

```
DimPlot(srt)
```


UMAP_2
5
0
-5
-5
0
5
UMAP_1
● control
● csds

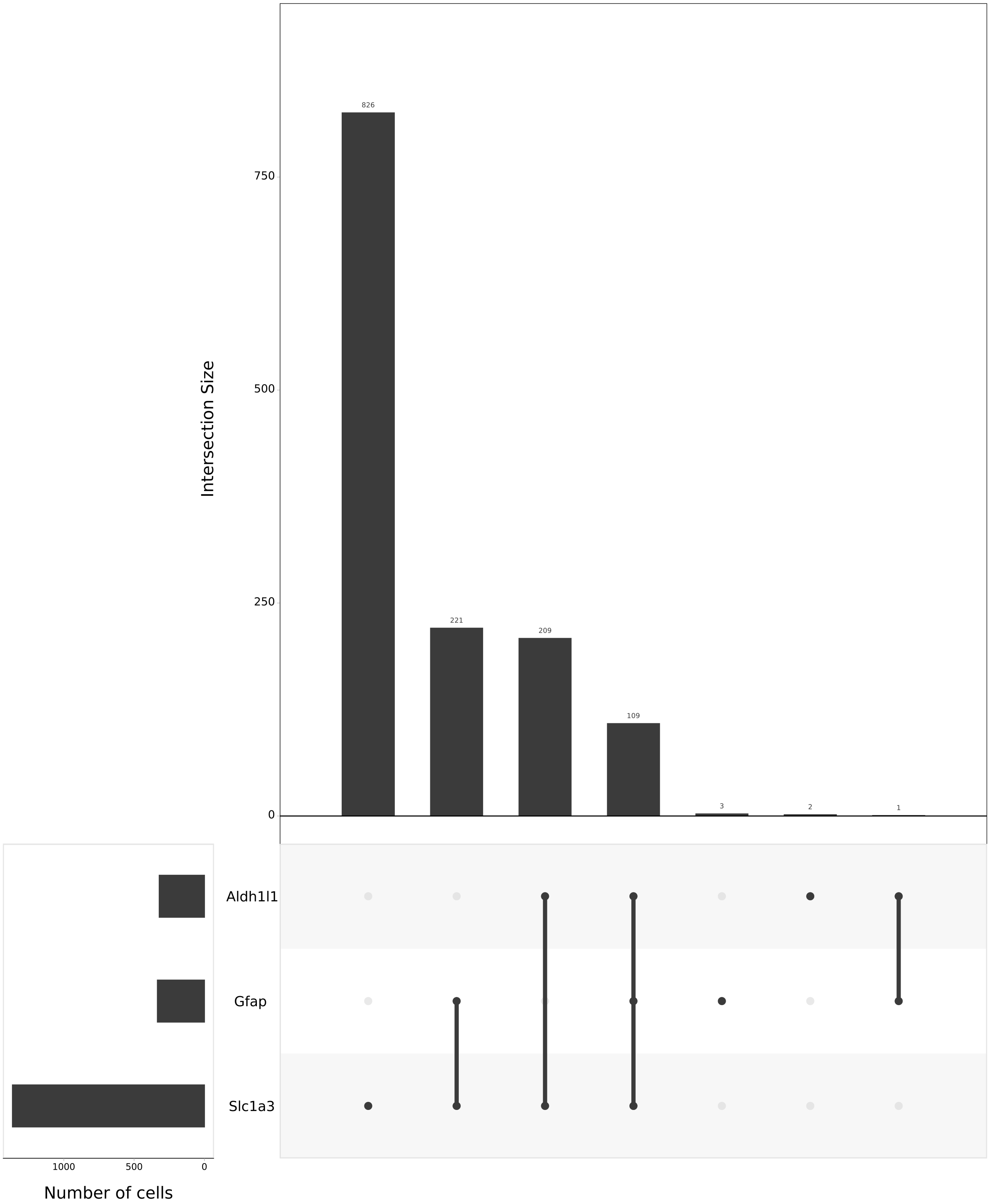
```
FeaturePlot_scCustom(
  srt,
  reduction = "umap",
  features = c(
    "Cnr1", "Cnr2", "Gpr55", "Slc1a3",
    "Dagla", "Daglb", "Mgll", "Gfap",
    "Napepld", "Gde1", "Faah", "Aldh1l1"
    ),
  label = F,
  num_columns = 4
) * NoLegend()
```


Cnr1
Cnr2
Gpr5
Dagla
Daglb
Mgl
Napepld
Gde1
Faa
UMAP_2
UMAP_1
5
0
-5
-10
-5
0
5

```r
sbs_mtx <-
  srt@assays$RNA@layers$data %>%
  as.data.frame() %>%
  t()

rownames(sbs_mtx) <- colnames(srt)
colnames(sbs_mtx) <- rownames(srt)

# Filter features
filt_low_genes <-
  colSums(sbs_mtx) %>%
  .[. > quantile(., 0.4)] %>%
  names()
sbs_mtx %<>% .[, filt_low_genes]

min_filt_vector2 <-
  sbs_mtx %>%
  as_tibble() %>%
  select(all_of(filt_low_genes)) %>%
  summarise(across(.fns = ~ quantile(.x, .005))) %>%
  as.list() %>%
  map(as.double) %>%
  simplify() %>%
  .[filt_low_genes]

# Prepare table of intersection sets analysis
content_sbs_mtx <-
  (sbs_mtx > min_filt_vector2) %>%
  as_tibble() %>%
  mutate_all(as.numeric) %>%
  bind_cols(
    srt@meta.data |> select(condit)
  )
```

```r
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(condit == 0) |>
      select(
```

```
        c("Cnr1", "Cnr2", "Gpr55", "Slc1a3", "Gfap", "Aldh1l1") %>% .[. %in% colname
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Cnr1", "Cnr2", "Gpr55", "Slc1a3", "Gfap", "Aldh1l1") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


826
221
209
109
3
2
1
750
500
250
0
Intersection Size
Aldh1l1
Gfap
Slc1a3
1000
500
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(condit == 0) |>
      select(
        c("Cnr1", "Cnr2", "Gpr55", "Slc1a3", "Gfap", "Aldh1l1") %>% .[. %in% colname
      )
  ))
```

Table 25: Data summary

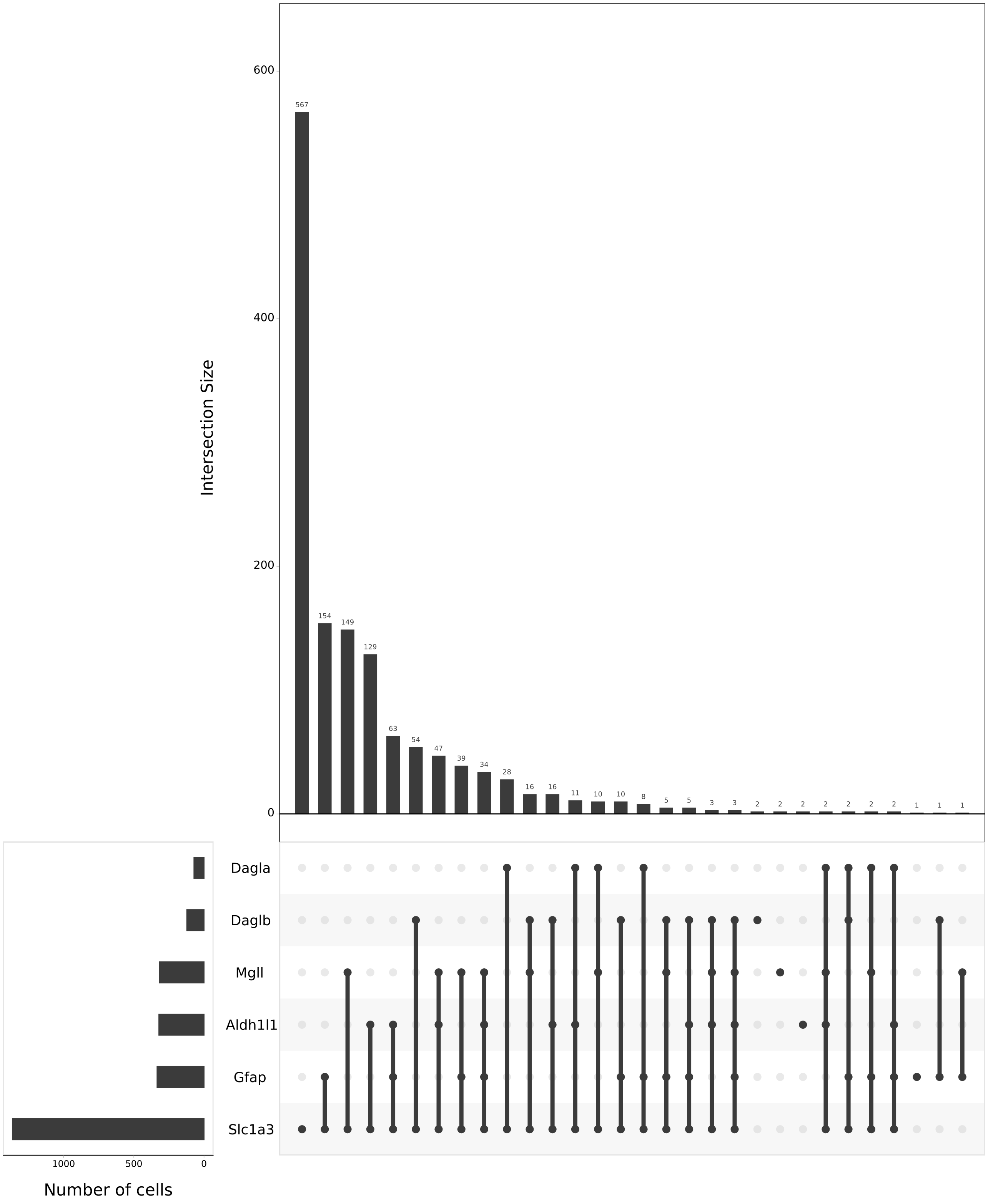
| | |
|---|---|
| Name | as.data.frame(…) |
| Number of rows | 1392 |
| Number of columns | 3 |
| Column type frequency: | |
| numeric | 3 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Slc1a3 | 0 | 1 | 0.98 | 0.14 | 0 | 1 | 1 | 1 | 1 | |
| Gfap | 0 | 1 | 0.24 | 0.43 | 0 | 0 | 0 | 0 | 1 | |
| Aldh1l1 | 0 | 1 | 0.23 | 0.42 | 0 | 0 | 0 | 0 | 1 | |

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(condit == 0) |>
      select(
        c("Dagla", "Daglb", "Mgll", "Slc1a3", "Gfap", "Aldh1l1") %>% .[. %in% colnam
      )
  ),
  order.by = "freq",
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
```

```
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Dagla", "Daglb", "Mgll", "Slc1a3", "Gfap", "Aldh1l1") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
600
400
200
0
567
154
149
129
63
54
47
39
34
28
16
16
11
10
10
8
5
5
3
3
2
2
2
2
2
2
2
1
1
1
Dagla
Daglb
Mgll
Aldh1l1
Gfap
Slc1a3
1000
500
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(condit == 0) |>
      select(
        c("Dagla", "Daglb", "Mgll", "Slc1a3", "Gfap", "Aldh1l1") %>% .[. %in% colnam
      )
  ))
```

Table 27: Data summary

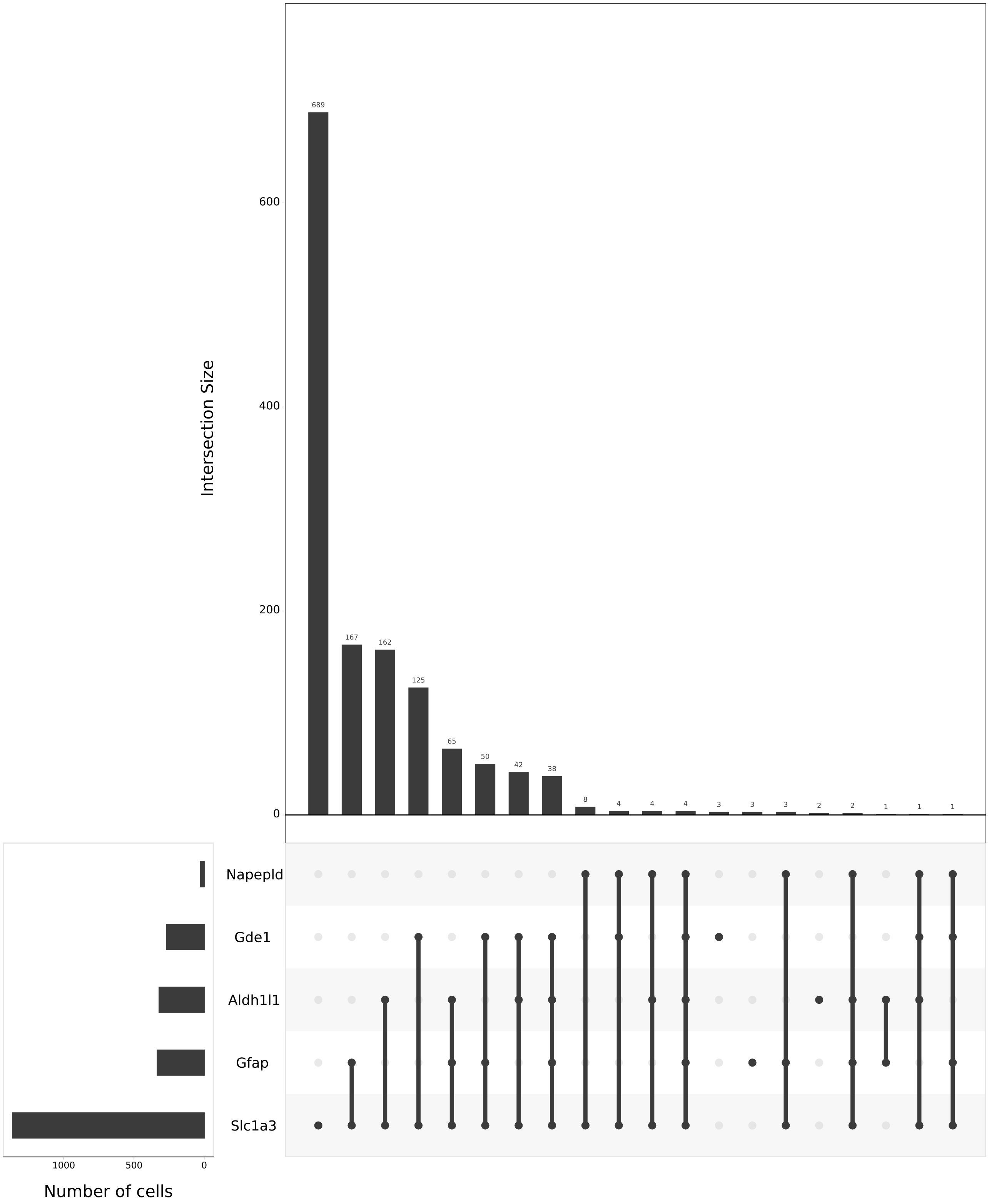
| | |
|---|---|
| Name | as.data.frame(...) |
| Number of rows | 1392 |
| Number of columns | 6 |
| Column type frequency: | |
| numeric | 6 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Dagla | 0 | 1 | 0.05 | 0.22 | 0 | 0 | 0 | 0 | 1 | |
| Daglb | 0 | 1 | 0.09 | 0.28 | 0 | 0 | 0 | 0 | 1 | |
| Mgll | 0 | 1 | 0.23 | 0.42 | 0 | 0 | 0 | 0 | 1 | |
| Slc1a3 | 0 | 1 | 0.98 | 0.14 | 0 | 1 | 1 | 1 | 1 | |
| Gfap | 0 | 1 | 0.24 | 0.43 | 0 | 0 | 0 | 0 | 1 | |
| Aldh1l1 | 0 | 1 | 0.23 | 0.42 | 0 | 0 | 0 | 0 | 1 | |

```
upset(
  as.data.frame(
    content_sbs_mtx |>
      filter(condit == 0) |>
      select(
        c("Napepld", "Gde1", "Faah", "Slc1a3", "Gfap", "Aldh1l1") %>% .[. %in% colna
      )
  ),
  order.by = "freq",
```

```
  cutoff = 3,
  sets.x.label = "Number of cells",
  number.angles = 0,
  point.size = 3.5, line.size = 2,
  text.scale = c(2, 1.6, 2, 1.3, 2, 1.1),
  nsets = 30,
  nintersects = 30,
  sets = c("Napepld", "Gde1", "Faah", "Slc1a3", "Gfap", "Aldh1l1") %>%
    .[. %in% colnames(content_sbs_mtx)],
  empty.intersections = NULL
)
```


Intersection Size
0
200
400
600
689
167
162
125
65
50
42
38
8
4
4
4
3
3
3
2
2
1
1
1
Napepld
Gde1
Aldh1l1
Gfap
Slc1a3
1000
500
0
Number of cells

```
skim(as.data.frame(
    content_sbs_mtx |>
      filter(condit == 0) |>
      select(
        c("Napepld", "Gde1", "Faah", "Slc1a3", "Gfap", "Aldh1l1") %>% .[. %in% colna
      )
  ))
```

Table 29: Data summary

| | |
|---|---|
| Name | as.data.frame(...) |
| Number of rows | 1392 |
| Number of columns | 5 |
| Column type frequency: | |
| numeric | 5 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| Napepld | 0 | 1 | 0.02 | 0.14 | 0 | 0 | 0 | 0 | 1 | |
| Gde1 | 0 | 1 | 0.19 | 0.39 | 0 | 0 | 0 | 0 | 1 | |
| Slc1a3 | 0 | 1 | 0.98 | 0.14 | 0 | 1 | 1 | 1 | 1 | |
| Gfap | 0 | 1 | 0.24 | 0.43 | 0 | 0 | 0 | 0 | 1 | |
| Aldh1l1 | 0 | 1 | 0.23 | 0.42 | 0 | 0 | 0 | 0 | 1 | |

```
sessioninfo::session_info()
```

```
- Session info ---------------------------------------------------------------
 setting  value
 version  R version 4.4.0 (2024-04-24)
 os       Ubuntu 22.04.4 LTS
 system   x86_64, linux-gnu
 ui       X11
 language en_US:en
```

```
 collate  en_US.UTF-8
 ctype    en_US.UTF-8
 tz       Etc/UTC
 date     2024-11-25
 pandoc   3.2 @ /opt/python/3.8.8/bin/ (via rmarkdown)

- Packages -----------------------------------------------------------------------
 package          * version     date (UTC) lib source
 abind              1.4-8       2024-09-12 [2] RSPM (R 4.4.0)
 anndata          * 0.7.5.6     2023-03-17 [2] RSPM (R 4.4.0)
 assertthat         0.2.1       2019-03-21 [2] RSPM (R 4.4.0)
 base64enc          0.1-3       2015-07-28 [2] RSPM (R 4.4.0)
 bayestestR         0.13.2      2024-02-12 [2] RSPM (R 4.4.0)
 beeswarm           0.4.0       2021-06-01 [2] RSPM (R 4.4.0)
 BiocManager        1.30.23     2024-05-04 [2] RSPM (R 4.4.0)
 bit                4.0.5       2022-11-15 [2] RSPM (R 4.4.0)
 bit64              4.0.5       2020-08-30 [2] RSPM (R 4.4.0)
 circlize           0.4.16      2024-06-19 [2] Github (jokergoo/circlize@9b21578)
 cli                3.6.3       2024-06-21 [2] RSPM (R 4.4.0)
 cluster            2.1.6       2023-12-01 [2] RSPM (R 4.4.0)
 coda               0.19-4.1    2024-01-31 [2] RSPM (R 4.4.0)
 codetools          0.2-20      2024-03-31 [2] RSPM (R 4.4.0)
 colorspace         2.1-1       2024-07-26 [2] RSPM (R 4.4.0)
 correlation        0.8.5       2024-06-16 [2] RSPM (R 4.4.0)
 cowplot          * 1.1.3       2024-01-22 [2] RSPM (R 4.4.0)
 crayon             1.5.3       2024-06-20 [2] RSPM (R 4.4.0)
 data.table         1.16.0      2024-08-27 [2] RSPM (R 4.4.0)
 datawizard         0.11.0      2024-06-05 [2] RSPM (R 4.4.0)
 deldir             2.0-4       2024-02-28 [2] RSPM (R 4.4.0)
 digest             0.6.37      2024-08-19 [2] RSPM (R 4.4.0)
 dotCall64          1.1-1       2023-11-28 [2] RSPM (R 4.4.0)
 dplyr            * 1.1.4       2023-11-17 [2] RSPM (R 4.4.0)
 effectsize         0.8.8       2024-05-12 [2] RSPM (R 4.4.0)
 emmeans            1.10.2      2024-05-20 [2] RSPM (R 4.4.0)
 estimability       1.5.1       2024-05-12 [2] RSPM (R 4.4.0)
 evaluate           1.0.0       2024-09-17 [2] RSPM (R 4.4.0)
 fansi              1.0.6       2023-12-08 [2] RSPM (R 4.4.0)
 farver             2.1.2       2024-05-13 [2] RSPM (R 4.4.0)
 fastDummies        1.7.4       2024-08-16 [2] RSPM (R 4.4.0)
 fastmap            1.2.0       2024-05-15 [2] RSPM (R 4.4.0)
 fitdistrplus       1.2-1       2024-07-12 [2] RSPM (R 4.4.0)
```

```
forcats          * 1.0.0       2023-01-29 [2] RSPM (R 4.4.0)
future           * 1.34.0      2024-07-29 [2] RSPM (R 4.4.0)
future.apply       1.11.2      2024-03-28 [2] RSPM (R 4.4.0)
generics           0.1.3       2022-07-05 [2] RSPM (R 4.4.0)
ggbeeswarm         0.7.2       2024-06-19 [2] Github (eclarke/ggbeeswarm@ce2da8a)
ggmin              0.0.0.9000  2024-06-19 [2] Github (sjessa/ggmin@8ada274)
ggplot2          * 3.5.1       2024-04-23 [2] RSPM (R 4.4.0)
ggprism            1.0.5       2024-06-19 [2] Github (csdaw/ggprism@b6e6c0e)
ggrastr            1.0.2       2024-06-19 [2] Github (VPetukhov/ggrastr@50ca3e0)
ggrepel            0.9.6       2024-09-20 [2] Github (slowkow/ggrepel@e94776b)
ggridges           0.5.6       2024-01-23 [2] RSPM (R 4.4.0)
ggstatsplot      * 0.12.3.9000 2024-06-19 [2] Github (IndrajeetPatil/ggstatsplot@d55f
GlobalOptions      0.1.2       2020-06-10 [2] RSPM (R 4.4.0)
globals            0.16.3      2024-03-08 [2] RSPM (R 4.4.0)
glue               1.7.0       2024-01-09 [2] RSPM (R 4.4.0)
goftest            1.2-3       2021-10-07 [2] RSPM (R 4.4.0)
gridExtra          2.3         2017-09-09 [2] RSPM (R 4.4.0)
gtable             0.3.5       2024-04-22 [2] RSPM (R 4.4.0)
hdf5r              1.3.10      2024-03-02 [2] RSPM (R 4.4.0)
here             * 1.0.1       2020-12-13 [2] RSPM (R 4.4.0)
hms                1.1.3       2023-03-21 [2] RSPM (R 4.4.0)
htmltools          0.5.8.1     2024-04-04 [2] RSPM (R 4.4.0)
htmlwidgets        1.6.4       2023-12-06 [2] RSPM (R 4.4.0)
httpuv             1.6.15      2024-03-26 [2] RSPM (R 4.4.0)
httr               1.4.7       2023-08-15 [2] RSPM (R 4.4.0)
ica                1.0-3       2022-07-08 [2] RSPM (R 4.4.0)
igraph             2.0.3       2024-03-13 [2] RSPM (R 4.4.0)
insight            0.20.1      2024-06-11 [2] RSPM (R 4.4.0)
irlba              2.3.5.1     2022-10-03 [2] RSPM (R 4.4.0)
janitor            2.2.0.9000  2024-06-19 [2] Github (sfirke/janitor@80cd1eb)
jsonlite           1.8.8       2023-12-04 [2] RSPM (R 4.4.0)
KernSmooth         2.23-24     2024-05-17 [2] RSPM (R 4.4.0)
knitr              1.48        2024-07-07 [2] RSPM (R 4.4.0)
labeling           0.4.3       2023-08-29 [2] RSPM (R 4.4.0)
later              1.3.2       2023-12-06 [2] RSPM (R 4.4.0)
lattice            0.22-6      2024-03-20 [2] RSPM (R 4.4.0)
lazyeval           0.2.2       2019-03-15 [2] RSPM (R 4.4.0)
leiden             0.4.3.1     2023-11-17 [2] RSPM (R 4.4.0)
lifecycle          1.0.4       2023-11-07 [2] RSPM (R 4.4.0)
listenv            0.9.1       2024-01-29 [2] RSPM (R 4.4.0)
lmtest             0.9-40      2022-03-21 [2] RSPM (R 4.4.0)
```

```
lubridate        * 1.9.3      2023-09-27 [2] RSPM (R 4.4.0)
magrittr         * 2.0.3      2022-03-30 [2] RSPM (R 4.4.0)
MASS               7.3-61     2024-06-13 [2] RSPM (R 4.4.0)
Matrix             1.7-0      2024-04-26 [2] RSPM (R 4.4.0)
matrixStats        1.4.1      2024-09-08 [2] RSPM (R 4.4.0)
mime               0.12       2021-09-28 [2] RSPM (R 4.4.0)
miniUI             0.1.1.1    2018-05-18 [2] RSPM (R 4.4.0)
multcomp           1.4-25     2023-06-20 [2] RSPM (R 4.4.0)
munsell            0.5.1      2024-04-01 [2] RSPM (R 4.4.0)
mvtnorm            1.2-5      2024-05-21 [2] RSPM (R 4.4.0)
nlme               3.1-165    2024-06-06 [2] RSPM (R 4.4.0)
paletteer          1.6.0      2024-01-21 [2] RSPM (R 4.4.0)
parallelly         1.38.0     2024-07-27 [2] RSPM (R 4.4.0)
parameters         0.21.7     2024-05-14 [2] RSPM (R 4.4.0)
patchwork        * 1.3.0.9000 2024-09-20 [2] Github (thomasp85/patchwork@2695a9f)
pbapply            1.7-2      2023-06-27 [2] RSPM (R 4.4.0)
pillar             1.9.0      2023-03-22 [2] RSPM (R 4.4.0)
pkgconfig          2.0.3      2019-09-22 [2] RSPM (R 4.4.0)
plotly             4.10.4     2024-01-13 [2] RSPM (R 4.4.0)
plyr               1.8.9      2023-10-02 [2] RSPM (R 4.4.0)
png                0.1-8      2022-11-29 [2] RSPM (R 4.4.0)
polyclip           1.10-7     2024-07-23 [2] RSPM (R 4.4.0)
progressr          0.14.0     2023-08-10 [2] RSPM (R 4.4.0)
promises           1.3.0      2024-04-05 [2] RSPM (R 4.4.0)
purrr            * 1.0.2      2023-08-10 [2] RSPM (R 4.4.0)
R.methodsS3        1.8.2      2022-06-13 [2] RSPM (R 4.4.0)
R.oo               1.26.0     2024-01-24 [2] RSPM (R 4.4.0)
R.utils            2.12.3     2023-11-18 [2] RSPM (R 4.4.0)
R6                 2.5.1      2021-08-19 [2] RSPM (R 4.4.0)
RANN               2.6.2      2024-08-25 [2] RSPM (R 4.4.0)
RColorBrewer     * 1.1-3      2022-04-03 [2] RSPM (R 4.4.0)
Rcpp               1.0.13     2024-07-17 [2] RSPM (R 4.4.0)
RcppAnnoy          0.0.22     2024-01-23 [2] RSPM (R 4.4.0)
RcppHNSW           0.6.0      2024-02-04 [2] RSPM (R 4.4.0)
readr            * 2.1.5      2024-01-10 [2] RSPM (R 4.4.0)
rematch2           2.1.2      2020-05-01 [2] RSPM (R 4.4.0)
remotes            2.5.0      2024-03-17 [2] RSPM (R 4.4.0)
repr               1.1.7      2024-03-22 [2] RSPM (R 4.4.0)
reshape2           1.4.4      2020-04-09 [2] RSPM (R 4.4.0)
reticulate       * 1.39.0     2024-09-05 [2] RSPM (R 4.4.0)
rlang              1.1.4      2024-06-04 [2] RSPM (R 4.4.0)
```

```
rmarkdown          2.28        2024-08-17 [2] RSPM (R 4.4.0)
ROCR               1.0-11      2020-05-02 [2] RSPM (R 4.4.0)
rprojroot          2.0.4       2023-11-05 [2] RSPM (R 4.4.0)
RSpectra           0.16-2      2024-07-18 [2] RSPM (R 4.4.0)
rstudioapi         0.16.0      2024-03-24 [2] RSPM (R 4.4.0)
rsvd               1.0.5       2021-04-16 [2] RSPM (R 4.4.0)
Rtsne              0.17        2023-12-07 [2] RSPM (R 4.4.0)
sandwich           3.1-0       2023-12-11 [2] RSPM (R 4.4.0)
scales             1.3.0       2023-11-28 [2] RSPM (R 4.4.0)
scattermore        1.2         2023-06-12 [2] RSPM (R 4.4.0)
scCustomize      * 2.1.2       2024-06-19 [2] Github (samuel-marsh/scCustomize@fc7a28
sceasy           * 0.0.7       2024-06-19 [2] Github (cellgeni/sceasy@c1c0bf9)
sctransform        0.4.1       2023-10-19 [2] RSPM (R 4.4.0)
sessioninfo        1.2.2       2021-12-06 [2] RSPM (R 4.4.0)
Seurat           * 5.1.0.9005  2024-09-20 [2] Github (satijalab/seurat@95de9dc)
SeuratDisk       * 0.0.0.9021  2024-06-19 [2] Github (mojaveazure/seurat-disk@877d4e1
SeuratObject     * 5.0.99.9001 2024-09-20 [2] Github (satijalab/seurat-object@1a140c7
SeuratWrappers   * 0.3.5       2024-06-19 [2] Github (satijalab/seurat-wrappers@8d46d
shape              1.4.6.1     2024-02-23 [2] RSPM (R 4.4.0)
shiny              1.9.1       2024-08-01 [2] RSPM (R 4.4.0)
skimr            * 2.1.5       2024-06-19 [2] Github (ropensci/skimr@d5126aa)
snakecase          0.11.1      2023-08-27 [2] RSPM (R 4.4.0)
sp               * 2.1-4       2024-04-30 [2] RSPM (R 4.4.0)
spam               2.10-0      2023-10-23 [2] RSPM (R 4.4.0)
spatstat.data      3.1-2       2024-06-21 [2] RSPM (R 4.4.0)
spatstat.explore   3.3-2       2024-08-21 [2] RSPM (R 4.4.0)
spatstat.geom      3.3-3       2024-09-18 [2] RSPM (R 4.4.0)
spatstat.random    3.3-2       2024-09-18 [2] RSPM (R 4.4.0)
spatstat.sparse    3.1-0       2024-06-21 [2] RSPM (R 4.4.0)
spatstat.univar    3.0-1       2024-09-05 [2] RSPM (R 4.4.0)
spatstat.utils     3.1-0       2024-08-17 [2] RSPM (R 4.4.0)
statsExpressions   1.5.4       2024-03-20 [2] RSPM (R 4.4.0)
stringi            1.8.4       2024-05-06 [2] RSPM (R 4.4.0)
stringr          * 1.5.1       2023-11-14 [2] RSPM (R 4.4.0)
survival           3.7-0       2024-06-05 [2] RSPM (R 4.4.0)
tensor             1.5         2012-05-05 [2] RSPM (R 4.4.0)
TH.data            1.1-2       2023-04-17 [2] RSPM (R 4.4.0)
tibble           * 3.2.1       2023-03-20 [2] RSPM (R 4.4.0)
tidyr            * 1.3.1       2024-01-24 [2] RSPM (R 4.4.0)
tidyselect         1.2.1       2024-03-11 [2] RSPM (R 4.4.0)
tidyverse        * 2.0.0.9000  2024-06-19 [2] Github (tidyverse/tidyverse@62f32d4)
```

```
 timechange         0.3.0       2024-01-18 [2] RSPM (R 4.4.0)
 tzdb               0.4.0       2023-05-12 [2] RSPM (R 4.4.0)
 UpSetR           * 1.4.0       2024-06-19 [2] Github (hms-dbmi/UpSetR@b14854a)
 utf8               1.2.4       2023-10-22 [2] RSPM (R 4.4.0)
 uwot               0.2.2       2024-04-21 [2] RSPM (R 4.4.0)
 vctrs              0.6.5       2023-12-01 [2] RSPM (R 4.4.0)
 vipor              0.4.7       2023-12-18 [2] RSPM (R 4.4.0)
 viridis          * 0.6.5       2024-01-29 [2] RSPM (R 4.4.0)
 viridisLite      * 0.4.2       2023-05-02 [2] RSPM (R 4.4.0)
 vroom              1.6.5       2023-12-05 [2] RSPM (R 4.4.0)
 withr              3.0.1       2024-07-31 [2] RSPM (R 4.4.0)
 xfun               0.47        2024-08-17 [2] RSPM (R 4.4.0)
 xtable             1.8-4       2019-04-21 [2] RSPM (R 4.4.0)
 yaml               2.3.10      2024-07-26 [2] RSPM (R 4.4.0)
 zeallot            0.1.0       2018-01-28 [2] RSPM (R 4.4.0)
 zoo                1.8-12      2023-04-13 [2] RSPM (R 4.4.0)

 [1] /home/etretiakov/R/x86_64-pc-linux-gnu-library/4.4
 [2] /opt/R/4.4.0/lib/R/library

- Python configuration --------------------------------------------------------
 python:         /opt/python/3.8.8/bin/python
 libpython:      /opt/python/3.8.8/lib/libpython3.8.so
 pythonhome:     /opt/python/3.8.8:/opt/python/3.8.8
 version:        3.8.8 | packaged by conda-forge | (default, Feb 20 2021, 16:22:27)  [
 numpy:          /opt/python/3.8.8/lib/python3.8/site-packages/numpy
 numpy_version:  1.23.5
 scanpy:         /home/etretiakov/.local/lib/python3.8/site-packages/scanpy

 NOTE: Python version was forced by RETICULATE_PYTHON

-------------------------------------------------------------------------------
```