Scaling up reproducible research for single cell transcriptomics using MetaNeighbor (Protocol 3)

Protocol 3: Functional characterization of replicating clusters

Protocol 3 demonstrates how to characterize functional gene sets contributing to cell type identity. Once replicating cell types have been identified with unsupervised MetaNeighbor (as in Protocols 1 and 2), supervised MetaNeighbor enables the functional interpretation of the biology contributing to each cell type's identity. In this protocol, we will focus on the characterization of inhibitory neuron subclasses from the mouse primary cortex as provided by the BICCN. The BICCN has shown that subclasses are strongly replicable across datasets and provided marker genes that are specific to each subclass. MetaNeighbor can be used to further quantify which pathways contribute to the subclasses' unique biological properties.

Step 1 - Creation of biologically relevant gene sets (1 minute)

1. To compute the functional characterization of clusters, we first need an ensemble of gene sets sampling relevant biological pathways. In this protocol we will consider the Gene Ontology (GO) annotations for mouse. The scripts used to build up-to-date gene sets can be found on Github, gene sets can be downloaded directly on FigShare.

```
go_sets = readRDS("go_mouse.rds")
```

Gene sets are stored as a named list, each element of the list corresponds to a gene set and contains a vector of gene symbols.

2. We load the dataset containing inhibitory neurons from the BICCN. The scripts used to build the dataset can be found on Github, the dataset can be downloaded on FigShare.

```
library(SingleCellExperiment)
biccn_gaba = readRDS("biccn_gaba.rds")
dim(biccn_gaba)
```

```
## [1] 24140 71368
```

3. Next we restrict our gene sets to genes that are present in the dataset. We then filter gene sets to keep gene sets of meaningful size: large enough to learn expression profiles (> 10), small enough to represent specific biological functions or processes (< 100).

```
known_genes = rownames(biccn_gaba)
go_sets = lapply(go_sets, function(gene_set) { gene_set[gene_set %in% known_genes] })
min_size = 10
max_size = 100
go_set_size = sapply(go_sets, length)
go_sets = go_sets[go_set_size >= min_size & go_set_size <= max_size]
length(go_sets)
```

```
## [1] 6488
```

Step 2: Functional characterization with supervised MetaNeighbor (30-90 minutes)

4. Once the gene set list is ready, we run the supervised *MetaNeighbor* function. Its inputs are similar to *MetaNeighborUS*, but it assumes that cell types have already been matched across datasets (i.e., they have identical names). Here we use joint BICCN subclasses, for which names have been normalized across datasets ("Pvalb", "Sst", "Sst Chodl", "Vip", "Lamp5", "Sncg"). Note that, because we are testing close to 6,500 gene sets, this step is expected to take a long time for large datasets. We recommend using this function inside a script and always save results to a file as soon as computations are done.

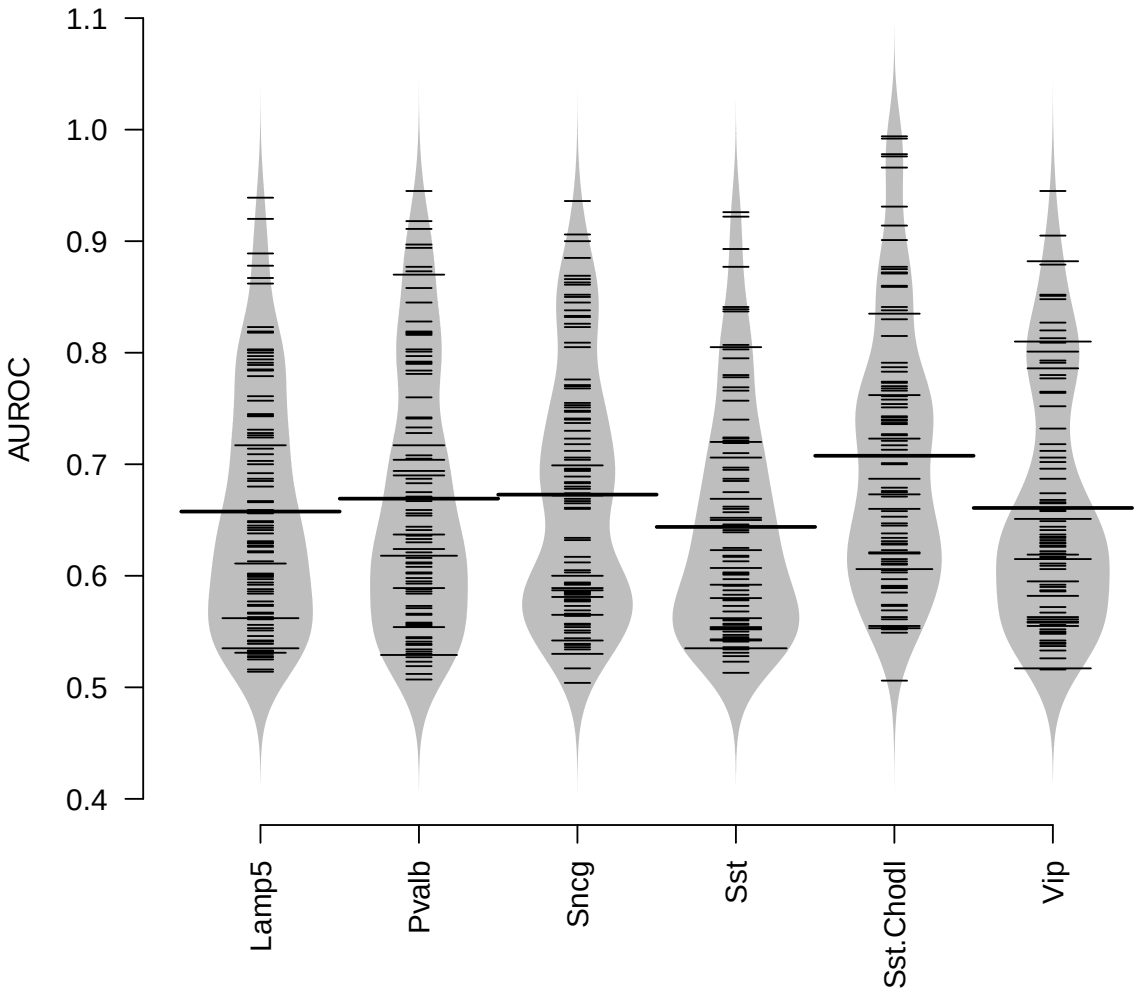
```
library(MetaNeighbor)

aurocs = MetaNeighbor(dat = biccn_gaba,
                      experiment_labels = biccn_gaba$study_id,
                      celltype_labels = biccn_gaba$joint_subclass_label,
                      genesets = go_sets[1:10],
                      fast_version = TRUE, bplot = FALSE, batch_size = 50)
#write.table(aurocs, "functional_aurocs.txt")
```

Later, results can be retrieved with the *read.table* function:

```
aurocs = read.table("functional_aurocs.txt")
```

5. We use the *plotBPlot* function on the first 100 gene sets to visualize how replicability depends on gene sets.

```
plotBPlot(head(aurocs, 100))
```



In this representation, large segments represent average gene set performance and short segments represent the performance of individual gene sets. We note that most gene sets contribute moderately to replicability (AUROC ~ 0.7), numerous gene sets have a performance close to random (AUROC ~ 0.5 - 0.6) and some gene sets have exceedingly high performance (AUROC > 0.8).

6. To focus on gene sets that contribute highly to cell type specificity, we create a summary table containing, for each gene set, cell type specific AUROCs, average AUROCs across cell types and gene set size.

```
gs_size = sapply(go_sets, length)
aurocs_df = data.frame(go_term = rownames(aurocs), aurocs)
aurocs_df$average = rowMeans(aurocs)
aurocs_df$n_genes = gs_size[rownames(aurocs)]
```

We then order gene sets by average AUROC and look at the top scoring gene sets.

```
head(aurocs_df[order(aurocs_df$average, decreasing = TRUE),],10)
```

go_term	Lamp5	Pvalb	Sncg	Sst	Sst.Chod	Vip	average	n_genes
GO:0007215%7Cglutamate receptor signaling pathway|BP	0.97	0.98	0.97	0.98	1.00	0.99	0.98	92
GO:0051966%7Cregulation of synaptic transmission, glutamatergic|BP	0.96	0.97	0.98	0.96	0.99	0.97	0.97	75
GO:0060076%7Cexcitatory synapse|CC	0.96	0.97	0.99	0.96	0.99	0.96	0.97	75
GO:0033555%7Cmulticellular organismal response to stress|BP	0.95	0.98	0.98	0.95	1.00	0.98	0.97	98
GO:0098839%7Cpostsynaptic density membrane|CC	0.92	0.97	0.98	0.98	0.98	0.97	0.97	93
GO:0099565%7Cchemical synaptic transmission, postsynaptic|BP	0.97	0.98	0.97	0.95	0.99	0.96	0.97	91
GO:0008306%7Cassociative learning|BP	0.97	0.98	0.96	0.96	0.99	0.95	0.97	100
GO:0099601%7Cregulation of neurotransmitter receptor activity|BP	0.96	0.98	0.96	0.95	0.99	0.98	0.97	61
GO:0060079%7Cexcitatory postsynaptic potential|BP	0.97	0.98	0.97	0.95	0.99	0.95	0.97	83
GO:0010771%7Cnegative regulation of cell morphogenesis involved in differentiation|BP	0.98	0.98	0.97	0.96	0.99	0.92	0.97	98

Without surprise, replicability is mainly driven by gene sets related to neuronal functions that are immediately relevant to the physiology of inhibitory neurons, such as "glutamate receptor signaling pathway", "regulation of synaptic transmission, glutamatergic", or "chemical synaptic transmission, postsynaptic". Note that most of the top scoring gene sets have a large number of genes, as larger sets of genes make it easier to learn generalizable expression profiles. To obtain even more specific biological functions, we can further filter for gene sets that have fewer than 20 genes.

```
small_sets = aurocs_df[aurocs_df$n_genes < 20,]
head(small_sets[order(small_sets$average, decreasing = TRUE),],10)
```

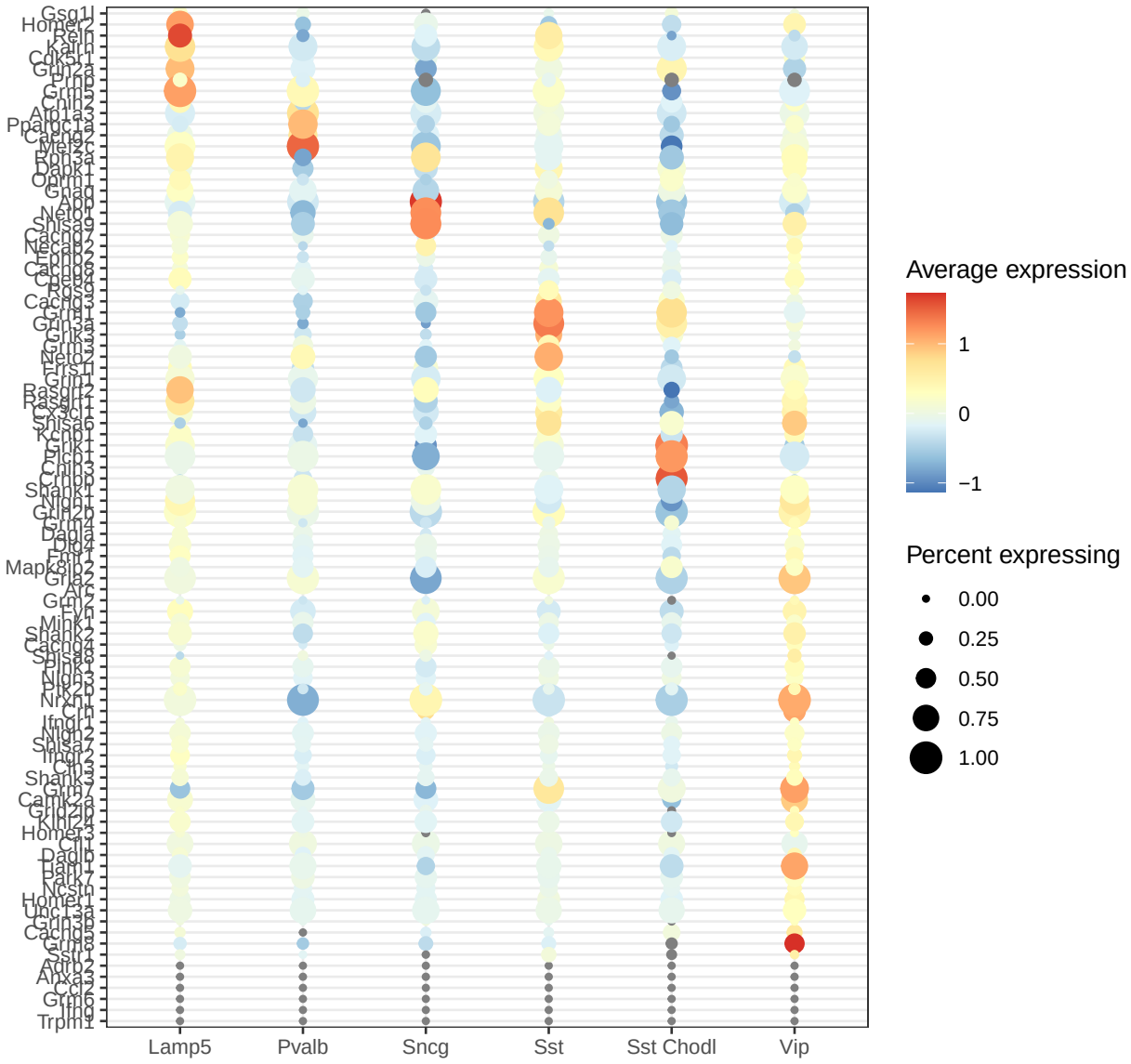
go_term	Lamp5	Pvalb	Sncg	Sst	Sst.Chod	Vip	average	n_genes
GO:0004970%7Cionotropic glutamate receptor activity|MF	0.90	0.92	0.91	0.96	0.97	0.92	0.93	19
GO:0035235%7Cionotropic glutamate receptor signaling pathway|BP	0.82	0.82	0.91	0.93	0.94	0.87	0.88	16
GO:0032230%7Cpositive regulation of synaptic transmission, GABAergic|BP	0.84	0.86	0.82	0.92	0.98	0.83	0.88	16
GO:0007216%7CG protein-coupled glutamate receptor signaling pathway|BP	0.89	0.85	0.76	0.92	0.95	0.84	0.87	16
GO:1905874%7Cregulation of postsynaptic density organization|BP	0.83	0.86	0.87	0.90	0.92	0.83	0.87	19
GO:0099150%7Cregulation of postsynaptic specialization assembly|BP	0.83	0.89	0.86	0.91	0.91	0.80	0.87	18
GO:0150052%7Cregulation of postsynapse assembly|BP	0.83	0.89	0.86	0.91	0.91	0.80	0.87	18
GO:0021889%7Colfactory bulb interneuron differentiation|BP	0.81	0.91	0.82	0.88	0.89	0.86	0.86	15
GO:0070679%7Cinositol 1,4,5 trisphosphate binding|MF	0.92	0.94	0.79	0.81	0.86	0.85	0.86	15
GO:1902711%7CGABA-A receptor complex|CC	0.82	0.87	0.87	0.80	0.99	0.80	0.86	19

Again, the top scoring gene sets are dominated by biological functions immediately relevant to inhibitory neuron physiology, such as "ionotropic glutamate receptor signaling pathway", "positive regulation of synaptic

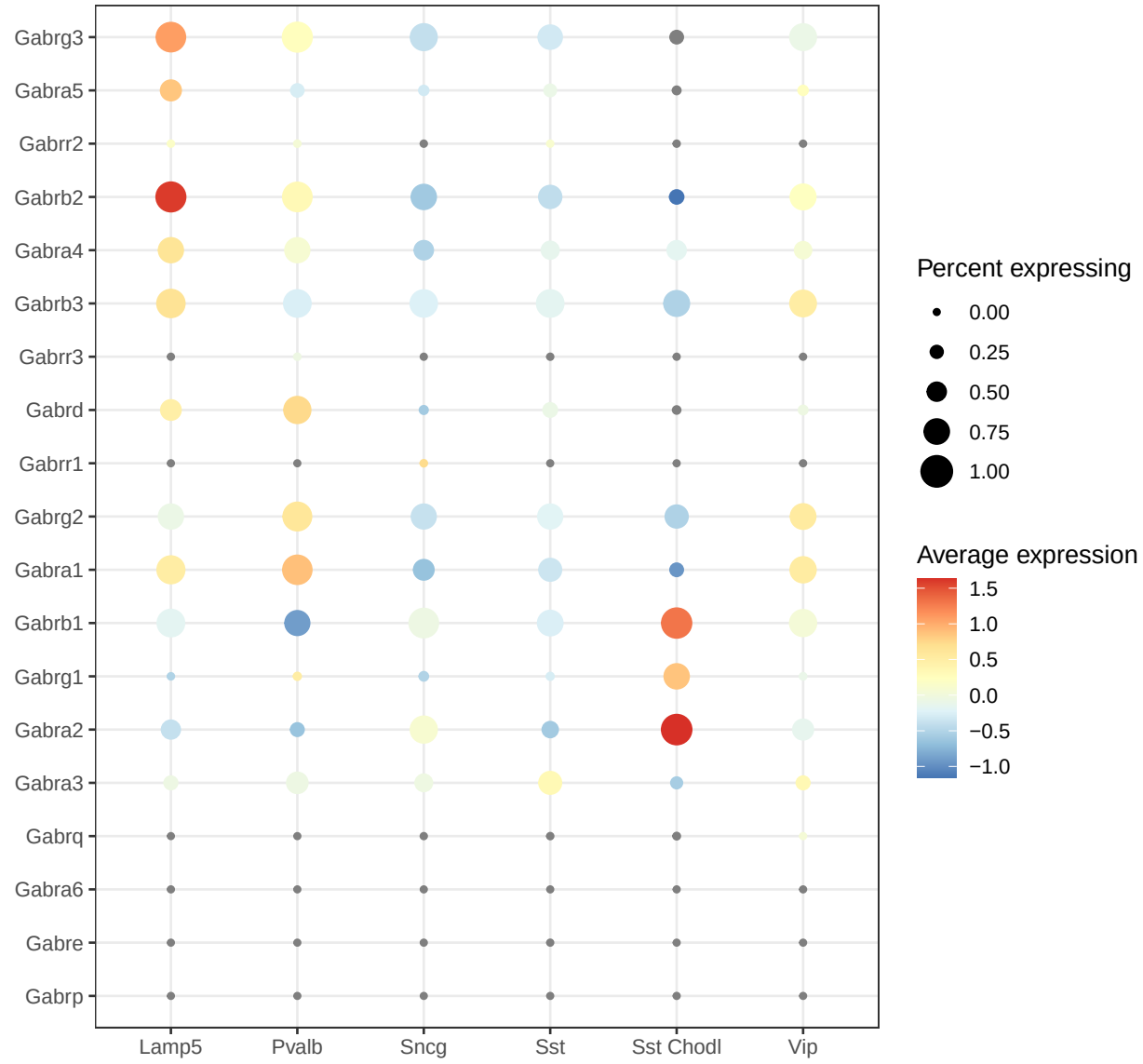
transmission, GABAergic", or "GABA-A receptor complex".

7. To understand how individual genes contribute to gene set performance, we use the *plotDotPlot* function, which shows the expression of all genes in a gene set of interest, averaged over all datasets.

```
plotDotPlot(dat = biccn_gaba,
            experiment_labels = biccn_gaba$study_id,
            celltype_labels = biccn_gaba$joint_subclass_label,
            gene_set = go_sets[["GO:0007215|glutamate receptor signaling pathway|BP"]])
```




```
plotDotPlot(dat = biccn_gaba,
            experiment_labels = biccn_gaba$study_id,
            celltype_labels = biccn_gaba$joint_subclass_label,
            gene_set = go_sets[["GO:1902711|GABA-A receptor complex|CC"]])
```



High scoring gene sets are characterized by the differential usage of genes from a given gene set. For example, when looking at the GABA-A receptor complex composition, Lamp5 preferentially uses the Gabrb2 and Gabrg3 receptors, Pvalb the Gabra1 receptor, and Sst Chodl the Gabra2, Gabrb1 and Gabrg1 receptors.