Application of VAM to Seurat pbmc_small scRNA-seq data using Seurat log normalization.

H. Robert Frost

1 Load the VAM package

Loading VAM will also load the required packages Seurat and MASS.

```
> library(VAM)
> if (!requireNamespace("Seurat", quietly=TRUE)) {
+    stop("Seurat package not available!")
+ }
```

2 Summary statistics for the pbmc_small scRNA-seq data

This example uses the pbmc_small data set included in the Seurat package and a single contrived gene set. Please see the other vignettes for more realistic examples using larger scRNA-seq data sets and gene set collections based on MSigDB.

```
> Seurat::pbmc_small
```

```
An object of class Seurat
230 features across 80 samples within 1 assay
Active assay: RNA (230 features, 20 variable features)
 2 dimensional reductions calculated: pca, tsne
```

```
> gene.names = rownames(Seurat::pbmc_small)
> gene.names[1:5]
```

```
[1] "MS4A1"   "CD79B"   "CD79A"   "HLA-DRA" "TCL1A"
```

```
> Seurat::VariableFeatures(Seurat::pbmc_small)[1:5]
```

```
[1] "PPBP"   "IGLL5"  "VDAC3"  "CD1C"   "AKR1C3"
```

3 Define gene set collection

A gene set collection containing just a single contrived set (containing the top 5 variable genes) will be used for this example.

```
> gene.set.name = "Test"
> gene.ids = c("PPBP", "IGLL5", "VDAC3", "CD1C", "AKR1C3")
> # Create a collection list for this gene set
> gene.set.id.list = list()
> gene.set.id.list[[1]] = gene.ids
> names(gene.set.id.list)[1] = gene.set.name
> gene.set.id.list
```

```
$Test
[1] "PPBP"   "IGLL5"  "VDAC3"  "CD1C"   "AKR1C3"

> # Create the list of gene indices required by vamForSeurat()
> (gene.set.collection = createGeneSetCollection(gene.ids=gene.names,
+         gene.set.collection=gene.set.id.list))

$Test
  PPBP  IGLL5  VDAC3   CD1C AKR1C3
   174     28    203    133    208

> gene.indices = gene.set.collection[[1]]
> (gene.names = gene.names[gene.indices])

[1] "PPBP"   "IGLL5"  "VDAC3"  "CD1C"   "AKR1C3"
```

4 Execute VAM method

Since the scRNA-seq data has been processed using Seurat, we execute VAM using the vamForSeurat() function. We have set return.dist=T so that the squared adjusted Mahalanobis distances will be returned in a "VAMdist" Assay.

```
> pbmc.vam = vamForSeurat(seurat.data=Seurat::pbmc_small,
+     gene.set.collection=gene.set.collection,
+     center=F, gamma=T, sample.cov=F, return.dist=T)
```

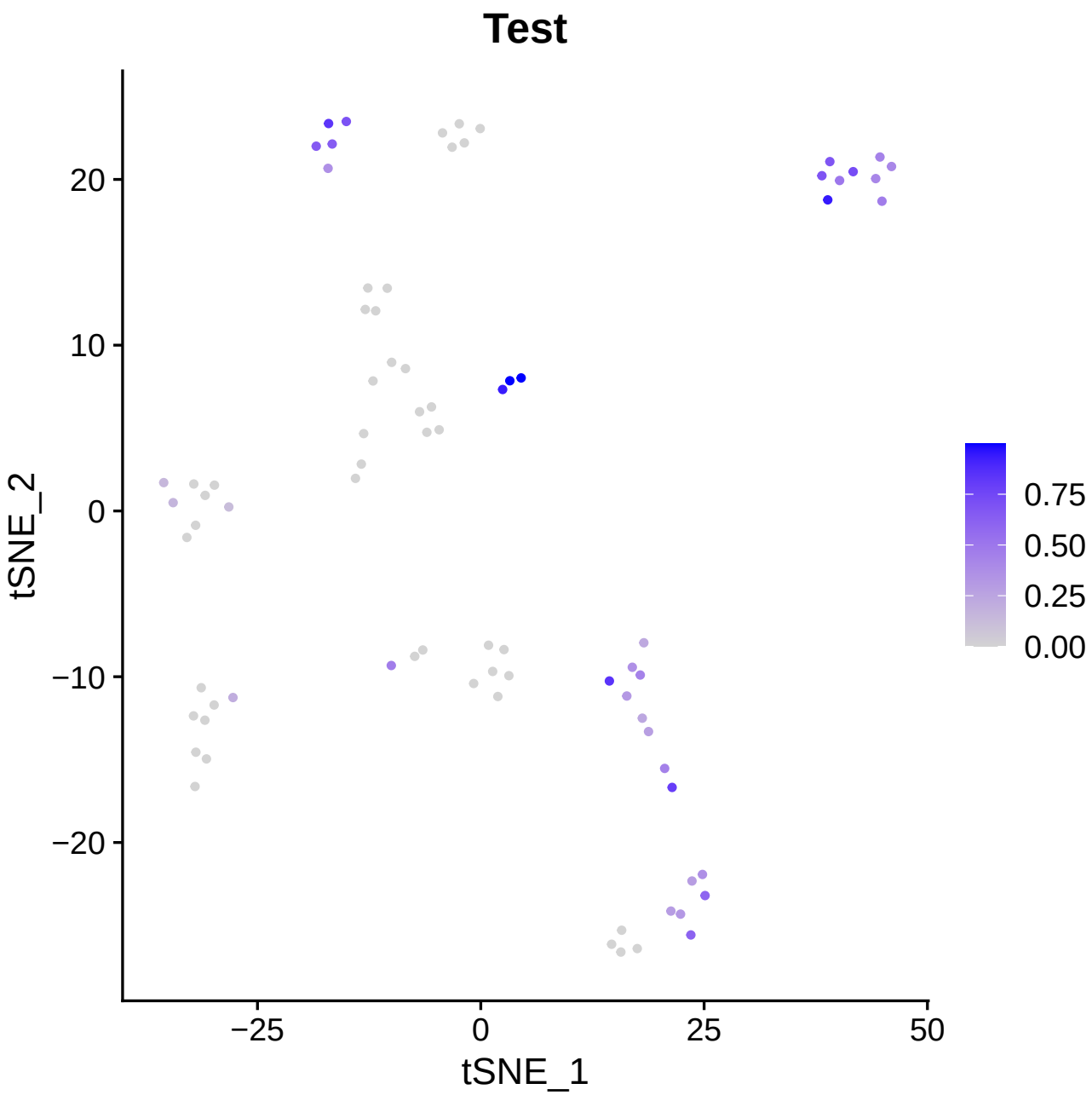
Look at the first few entries in the "VAMdist" and "VAMcdf" Assays.

```
> pbmc.vam@assays$VAMdist[1,1:10]

1 x 10 sparse Matrix of class "dgCMatrix"

Test . . 8.598434 17.68048 . . 23.5895 . . 19.64062

> pbmc.vam@assays$VAMcdf[1,1:10]

1 x 10 sparse Matrix of class "dgCMatrix"

Test . . 0.1202776 0.3193758 . . 0.4426203 . . 0.3616033
```

5 Visualize VAM scores

Visualize VAM scores using Seurat FeaturePlot(). The default Assay must first be changed to "VAMcdf".

```
> Seurat::DefaultAssay(object = pbmc.vam) = "VAMcdf"
> Seurat::FeaturePlot(pbmc.vam, reduction="tsne", features=gene.set.name)
```


Test
20
10
0
−10
−20
tSNE_2
−25
0
25
50
tSNE_1
0.75
0.50
0.25
0.00