

5-Scoring

Joan Abinet

2025-06-25 15:59:40 +0200

Contents

Introduction	2
Loading packages	2
Loading annotated Cell	2
loading Tissue leukocyte from GSE142754	2
QC	3
Pre-processing workflow	4
clustering	6
Create a subset with the neutrophils	8
Calculate gene signature for the PBS and Treated cells	8
Signature PBS	8
Signature treated	9

Introduction

Gene signatures specific of transcriptional change in neutrophils were obtained by selecting the top 20 genes with the highest absolute log fold-change in each condition (control and MarNeu-targeted). The 2 signatures were then tested on neutrophils from different tissues (from GSE142754), using the AddModuleScore function.

Loading packages

```
suppressMessages({  
  library(dplyr)  
  library(Seurat)  
  library(patchwork)  
  library(ggplot2)  
  library(formatR)  
  library(dittoSeq)  
  library(ggrepel)  
  library(xlsx)  
})
```

Loading annotated Cell

```
neutro_depletion <- readRDS("../06-Neutrophils_Processing/neutro_depleted_Part1.rds")
```

loading Tissue leukocyte from GSE142754

I didn't include the sample PBZT13_2 that seemed to have problems

```
# specify the directory where you saved the download the  
# cellranger output  
all_dirs <- list.dirs(path = "Data", full.names = TRUE, recursive = F)  
  
list_sample_name <- c("BM1", "BM2", "Lung1", "Lung2", "PBZT5", "PBZT13_1",  
  "Spleen1", "Spleen2")  
  
list_sample <- list()  
for (i in 1:length(all_dirs)) {  
  
  sample_file <- all_dirs[i]  
  Seq_raw_file <- Read10X(data.dir = sample_file)  
  Seurat_file <- CreateSeuratObject(counts = Seq_raw_file, project = list_sample_name[i],  
    min.cells = 3, min.features = 200)  
  
  list_sample <- append(list_sample, Seurat_file)  
}  
list_sample  
  
## [[1]]  
## An object of class Seurat  
## 12328 features across 1469 samples within 1 assay  
## Active assay: RNA (12328 features, 0 variable features)  
##  
## [[2]]
```

```

## An object of class Seurat
## 13189 features across 7379 samples within 1 assay
## Active assay: RNA (13189 features, 0 variable features)
##
## [[3]]
## An object of class Seurat
## 11497 features across 960 samples within 1 assay
## Active assay: RNA (11497 features, 0 variable features)
##
## [[4]]
## An object of class Seurat
## 14160 features across 3189 samples within 1 assay
## Active assay: RNA (14160 features, 0 variable features)
##
## [[5]]
## An object of class Seurat
## 11665 features across 1334 samples within 1 assay
## Active assay: RNA (11665 features, 0 variable features)
##
## [[6]]
## An object of class Seurat
## 14078 features across 7555 samples within 1 assay
## Active assay: RNA (14078 features, 0 variable features)
##
## [[7]]
## An object of class Seurat
## 12183 features across 1444 samples within 1 assay
## Active assay: RNA (12183 features, 0 variable features)
##
## [[8]]
## An object of class Seurat
## 13710 features across 5977 samples within 1 assay
## Active assay: RNA (13710 features, 0 variable features)
tissue_leukocyte <- merge(list_sample[[1]], y = list_sample[-1], add.cell.ids = c("1",
  "2", "3", "4", "5", "6", "7", "8"), project = "Tissue_leukocyte")

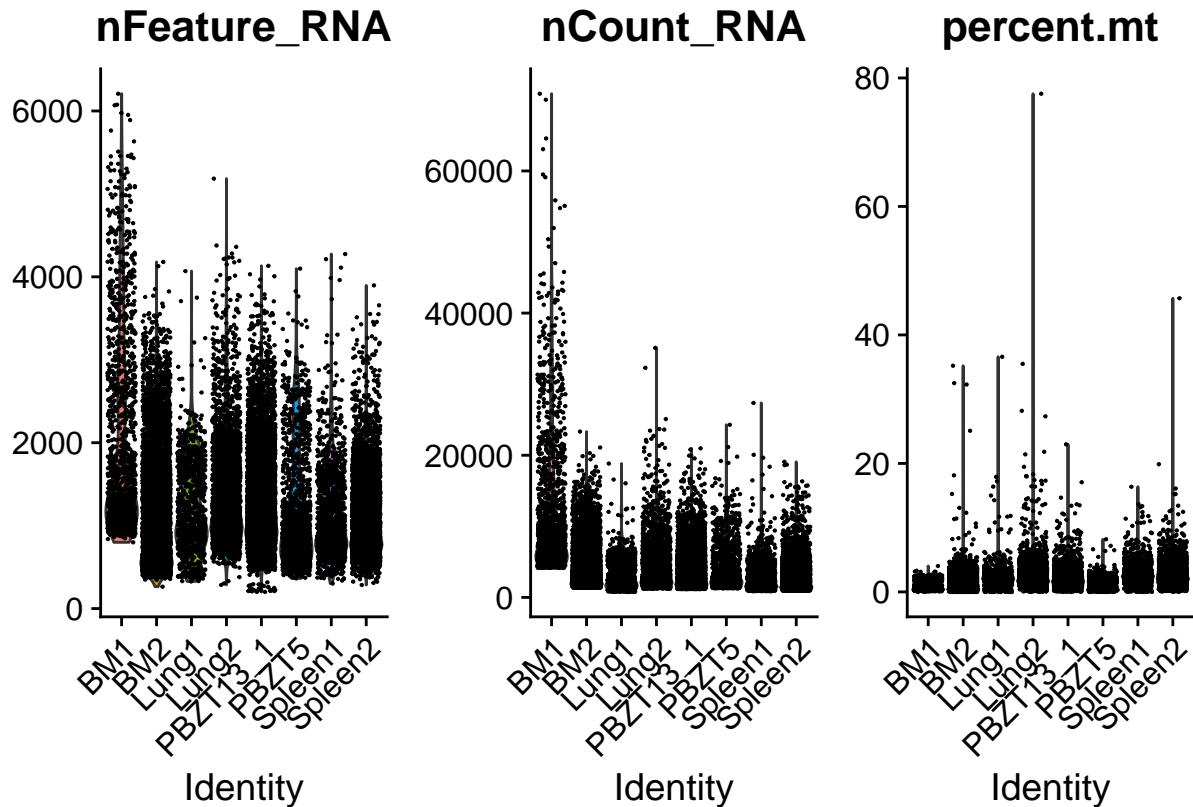
```

QC

```

tissue_leukocyte[["percent.mt"]] <- PercentageFeatureSet(tissue_leukocyte,
  pattern = "^\$mt\$")
VlnPlot(tissue_leukocyte, features = c("nFeature_RNA", "nCount_RNA",
  "percent.mt"), ncol = 3)

```



Pre-processing workflow

```

# Filtering
tissue_leukocyte <- subset(tissue_leukocyte, subset = nFeature_RNA >
  500 & percent.mt < 5)

# Data normalization
tissue_leukocyte <- NormalizeData(tissue_leukocyte, normalization.method = "LogNormalize",
  scale.factor = 10000)

# Feature selection
tissue_leukocyte <- FindVariableFeatures(tissue_leukocyte, selection.method = "vst",
  nfeatures = 2000)
top10 <- head(VariableFeatures(tissue_leukocyte), 10)
p1 <- VariableFeaturePlot(tissue_leukocyte)
p1 <- LabelPoints(plot = p1, points = top10, repel = T)

## When using repel, set xnudge and ynudge to 0 for optimal results

# Scaling the data
all.genes <- rownames(tissue_leukocyte)
tissue_leukocyte <- ScaleData(tissue_leukocyte, features = all.genes)

## Centering and scaling data matrix

# linear dimensional reduction
tissue_leukocyte <- RunPCA(tissue_leukocyte, features = VariableFeatures(object = tissue_leukocyte))

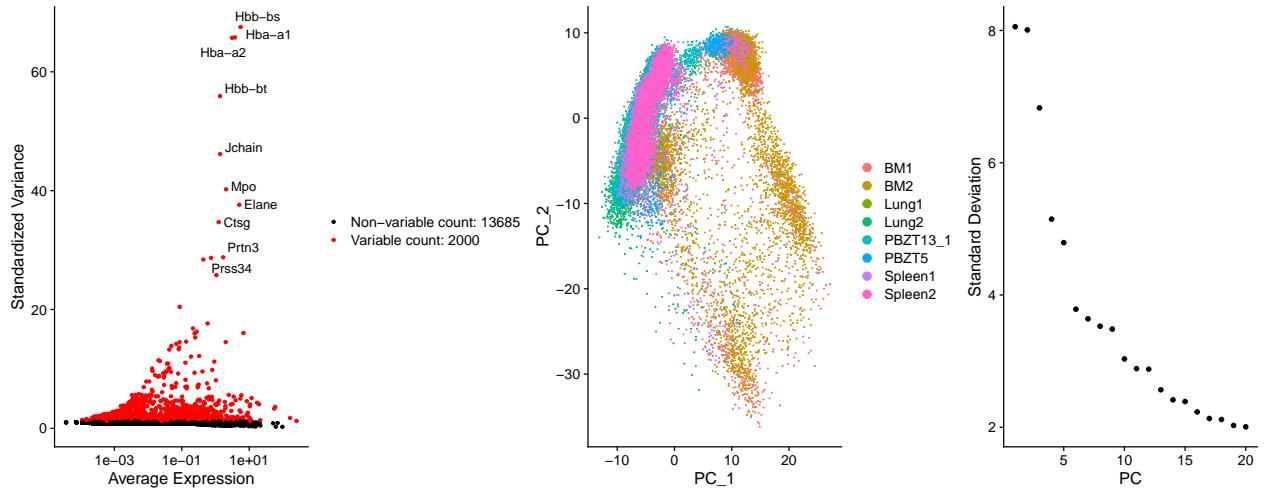
## PC_ 1

```

```

## Positive: Ngp, Camp, Lcn2, Wfdc21, S100a8, S100a9, Ltf, Hmgn2, Cebpe, Retnlg
##     Anxa1, Ly6g, Lrg1, Pglyrp1, Cd63, Cd177, Chil1, Adpgk, Serpinb1a, Hp
##     Trem3, Cd24a, AA467197, Mmp9, Hmgb2, Itgb21, Mmp8, Dstn, Fcnb, Ckap4
## Negative: Rps8, Rpl10a, Crip1, Rps2, Rpl36a, Ctss, Rps20, Rpl12, Rps12, S100a10
##     Cd74, Rps17, Hspa8, Eef1g, Wdr89, Gngt2, Rpl28, Rpl10-ps3, Ccl5, Lgals1
##     Junb, H2-DMa, Npm1, H2-Ab1, Pou2f2, Gm10073, Apoe, Ly86, Gm2a, H2-Aa
## PC_ 2
## Positive: Gzma, Ccl5, Ncr1, Mmp9, Nkg7, Klra9, Gzmb, Ly6g, Cma1
##     Klra4, Klra7, Mmp8, Ceacam10, Klrc1, Klri2, Mxd1, Pglyrp1, AA467197, Serpinb9
##     Ccnd2, Chil1, Klra3, Atp1b1, Lrg1, Klrg1, Klrb1a, Ccl4, Fpr2, Cxcr2
## Negative: 2810417H13Rik, Cks1b, Tuba1b, Birc5, Ms4a6c, Plac8, Rrm2, Top2a, Spc24, Stmn1
##     Cdca3, Cdk1, Nusap1, Ccna2, Ptma, Cdca8, Ifi30, F13a1, Pld4, Ctss
##     Dbi, Psap, Ccnb2, Ube2c, Lamp1, Aurkb, S100a4, Napsa, Mki67, Cst3
## PC_ 3
## Positive: Ifitm3, Lgals3, Ccl6, Ifitm6, Cebpb, Hp, Cst3, Ifi27l2a, Slpi, Wfdc17
##     Clec4a3, Psap, Ctss, Tgfbi, Fos, Sirpb1c, Mpeg1, Slfn1, Plac8, Csf1r
##     Clec4a1, Sirpb1b, Smpd13a, Mmp8, Mmp9, Gngt2, Apoe, AF251705, Fth1, Ms4a6c
## Negative: Nkg7, Gzma, Ccl5, Ncr1, Lgals1, Gzmb, Ppia, Klra8, Klra9, Ccnd2
##     Klra7, Cst7, Klra4, Ptma, H2afz, Cma1, S100a10, Atp1b1, Klri2, Serpinb9
##     Klrc1, Serpinb6b, Klra3, Klrb1a, Cd7, Ran, Klrg1, Irf8, Hmgb1, Ccl4
## PC_ 4
## Positive: Ms4a6c, Ms4a4c, Plac8, Ifitm3, S100a4, Ifi27l2a, Ms4a6b, Ly86, Napsa, F13a1
##     Ccr2, Ccl9, Clec4a1, Ifi204, Crip1, Mnda, Wfdc17, Gm9733, Pld4, Emb
##     Pid1, Ccdc109b, Ppbp, Ptpro, Gng11, Irf7, AI839979, Pyhin1, Pou2f2, Pf4
## Negative: Plet1, Ltc4s, Atp6v0d2, Ear1, Krt19, Fabp1, Hebp1, Mrc1, Dab2, Krt79
##     Cidec, F7, Car4, Slc7a2, Mt1, Lpl, Abcg1, Ctsk, Axl, Olr1
##     Acaa1b, Dst, Dapk1, Clec4n, Scgb1a1, Kazald1, Tcf7l2, Cpne5, Nceh1, Tns1
## PC_ 5
## Positive: Lgals1, H2afz, Nkg7, Gzma, Ncr1, Ccl5, Gzmb, S100a10, Klra9, Klra8
##     Ccr2, Cma1, Klra7, Klra4, Ccl4, Ly6c2, Ccl3, Ctla2a, Serpinb9, Cst7
##     F13a1, Klrc1, Fn1, Klri2, Klrg1, Klra3, Klrb1a, Ccl9, Dbi, Irf8
## Negative: Cd79a, H2-DMb2, Ly6d, Cd79b, Ms4a1, Ebf1, H2-Eb1, H2-Aa, Cd74, H2-Ab1
##     Mzb1, H2-Ob, Siglecg, Fcmr, Bank1, H2-Oa, Fcrla, Gm43603, Ltb, Cd19
##     Ly6a, Blnk, Ralgps2, Spib, Pou2af1, Tnfrsf13c, Snn, Blk, Gm43291, Fcer2a
p2 <- DimPlot(tissue_leukocyte, reduction = "pca")
p3 <- ElbowPlot(tissue_leukocyte)
p1 + p2 + p3

```



clustering

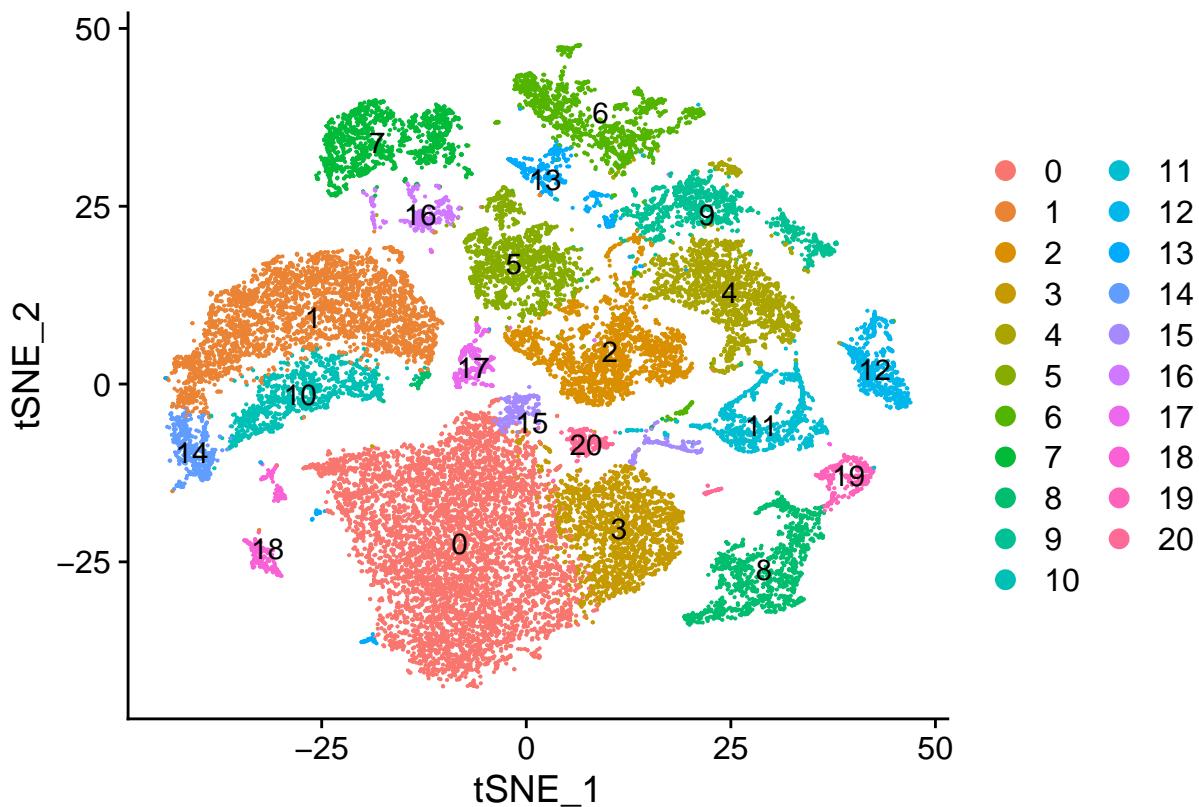
```
tissue_leukocyte <- FindNeighbors(tissue_leukocyte, dims = 1:15)

## Computing nearest neighbor graph
## Computing SNN
tissue_leukocyte <- FindClusters(tissue_leukocyte, resolution = 0.4)

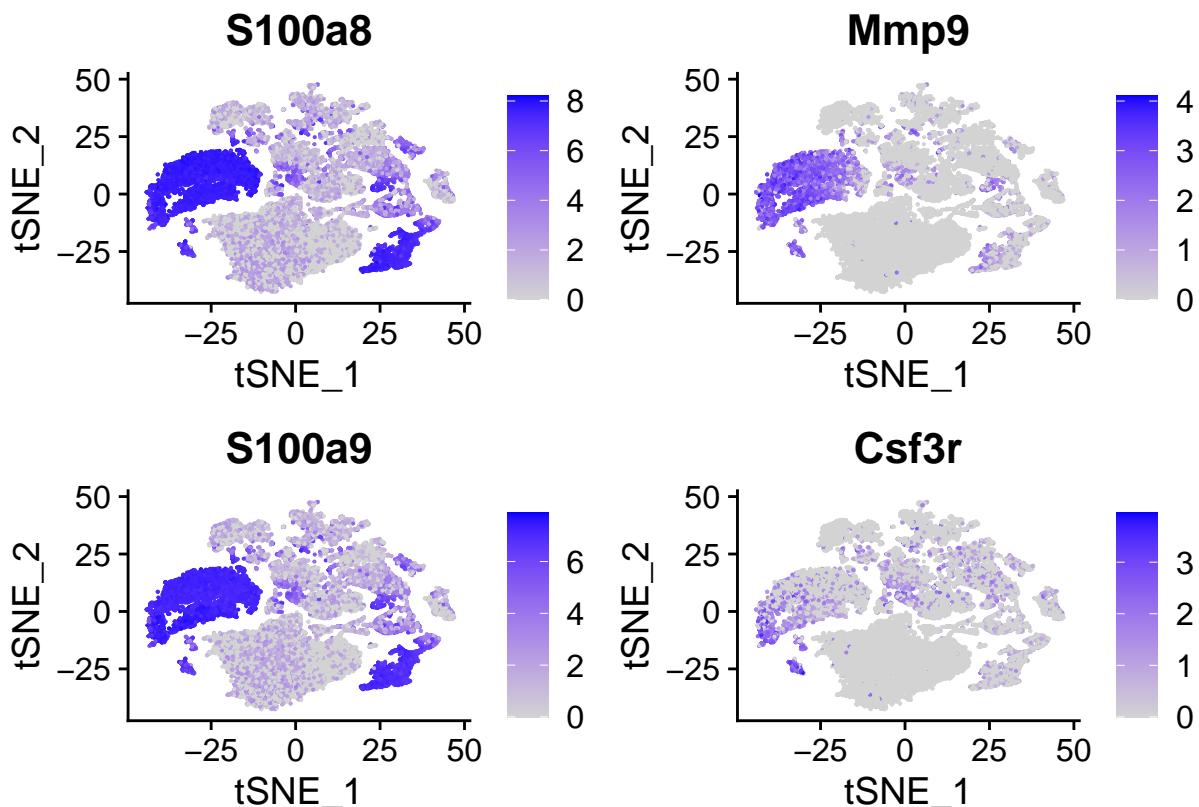
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 27123
## Number of edges: 908599
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.9447
## Number of communities: 21
## Elapsed time: 4 seconds

tissue_leukocyte <- RunTSNE(tissue_leukocyte, dims = 1:15, check_duplicates = FALSE)

DimPlot(tissue_leukocyte, reduction = "tsne", label = T)
```



```
FeaturePlot(tissue_leukocyte, features = c("S100a8", "Mmp9", "S100a9",
                                         "Csf3r"), reduction = "tsne")
```



Based on the expression of the 4 markers, the clusters 1,8,10,14,18 and 19 were identified as neutrophils

Create a subset with the neutrophils

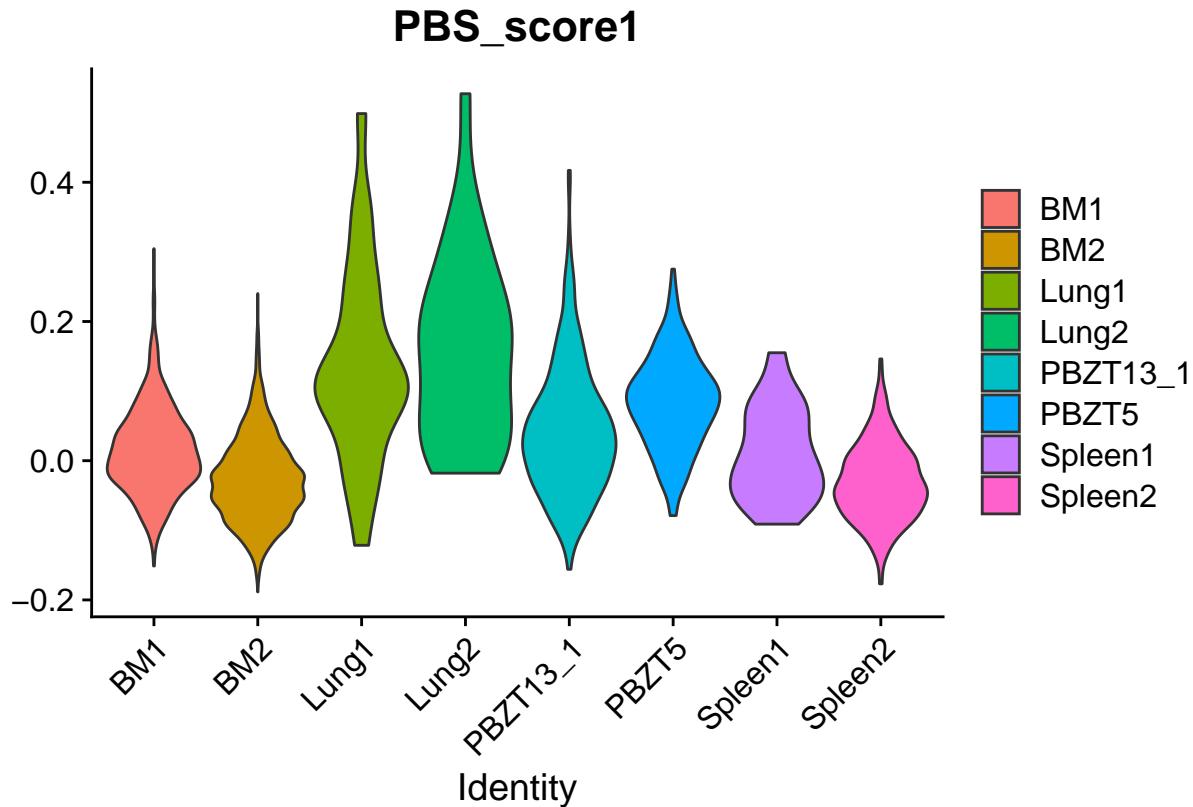
```
tissue_neutrophils <- subset(tissue_leukocyte, subset = seurat_clusters %in%  
  c(1, 10, 14, 18, 8, 19))
```

Calculate gene signature for the PBS and Treated cells

```
Neutro.markers <- FindMarkers(neutro_depletion, ident.1 = "PBS", ident.2 = c("acxcr2",  
  "aLy6G"), min.pct = 0.1, group.by = "orig.ident", logfc.threshold = 0.25)  
  
Neutro.markers <- Neutro.markers[Neutro.markers$p_val_adj < 0.05,  
  ]  
  
Neutro.markers <- Neutro.markers[order(Neutro.markers$avg_log2FC,  
  decreasing = T), ]  
  
signature_PBS <- rownames(Neutro.markers)[1:60]  
signature_treated <- rownames(Neutro.markers)[61:81]
```

Signature PBS

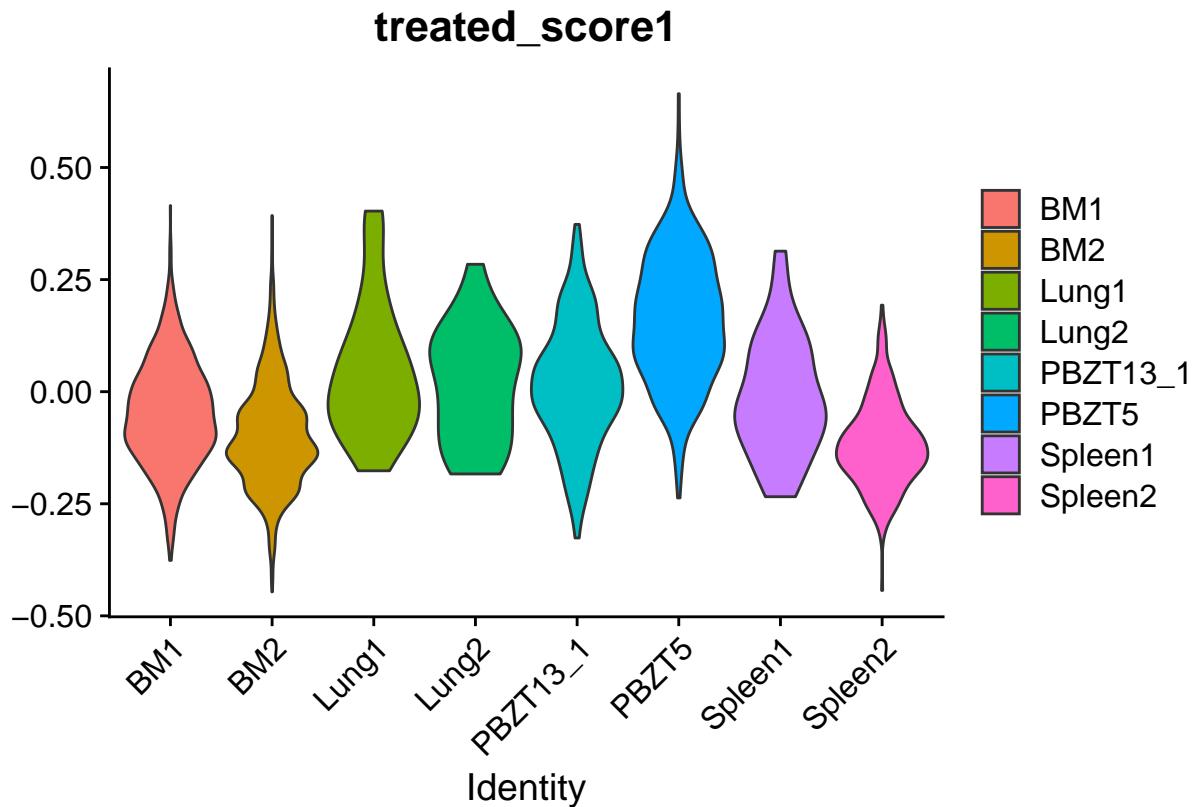
```
tissue_neutrophils <- AddModuleScore(tissue_neutrophils, features = list(signature_PBS),  
  name = "PBS_score")  
  
## Warning: The following features are not present in the object: Slc49a4,  
## Gm49339, not searching for symbol synonyms  
VlnPlot(tissue_neutrophils, features = "PBS_score1", group.by = "orig.ident",  
  pt.size = 0)
```



Signature treated

```
tissue_neutrophils <- AddModuleScore(tissue_neutrophils, features = list(signature_treated),
                                         name = "treated_score")

## Warning: The following features are not present in the object: Arhgap26.1, not
## searching for symbol synonyms
VlnPlot(tissue_neutrophils, features = "treated_score1", group.by = "orig.ident",
        pt.size = 0)
```



```
# ggsave('vlnplot_Treated.png', width = 10, height = 7)

sessionInfo()

## R version 4.3.3 (2024-02-29)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 24.04.2 LTS
##
## Matrix products: default
## BLAS:    /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
## LAPACK:  /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblas-pthread-r0.3.26.so;  LAPACK version 3.12.0
##
## locale:
## [1] LC_CTYPE=en_US.UTF-8          LC_NUMERIC=C
## [3] LC_TIME=fr_BE.UTF-8          LC_COLLATE=en_US.UTF-8
## [5] LC_MONETARY=fr_BE.UTF-8       LC_MESSAGES=en_US.UTF-8
## [7] LC_PAPER=fr_BE.UTF-8          LC_NAME=fr_BE.UTF-8
## [9] LC_ADDRESS=fr_BE.UTF-8         LC_TELEPHONE=fr_BE.UTF-8
## [11] LC_MEASUREMENT=fr_BE.UTF-8    LC_IDENTIFICATION=fr_BE.UTF-8
##
## time zone: Europe/Brussels
## tzcode source: system (glibc)
##
## attached base packages:
## [1] stats      graphics   grDevices utils      datasets   methods    base
##
## other attached packages:
## [1] xlsx_0.6.5        ggrepel_0.9.3      dittoSeq_1.12.0    formatR_1.14
## [5] ggplot2_3.4.2      patchwork_1.1.2     SeuratObject_4.1.3  Seurat_4.3.0
```

```

## [9] dplyr_1.1.2
##
## loaded via a namespace (and not attached):
##   [1] RColorBrewer_1.1-3           rstudioapi_0.14
##   [3] jsonlite_1.8.7              magrittr_2.0.3
##   [5] ggbeeswarm_0.7.2            spatstat.utils_3.0-3
##   [7] farver_2.1.1                rmarkdown_2.23
##   [9] zlibbioc_1.46.0             vctrs_0.6.3
##  [11] ROOCR_1.0-11               spatstat.explore_3.2-1
##  [13] RCurl_1.98-1.12            S4Arrays_1.2.1
##  [15] htmltools_0.5.5            sctransform_0.3.5
##  [17] parallelly_1.36.0          KernSmooth_2.23-22
##  [19] htmlwidgets_1.6.2          ica_1.0-3
##  [21] plyr_1.8.8                 plotly_4.10.2
##  [23] zoo_1.8-12                igraph_1.5.0.1
##  [25] mime_0.12                  lifecycle_1.0.3
##  [27] pkgconfig_2.0.3            Matrix_1.6-1
##  [29] R6_2.5.1                   fastmap_1.1.1
##  [31] GenomeInfoDbData_1.2.10    MatrixGenerics_1.12.2
##  [33] fitdistrplus_1.1-11       future_1.33.0
##  [35] shiny_1.7.4.1              digest_0.6.33
##  [37] colorspace_2.1-0          S4Vectors_0.38.1
##  [39] tensor_1.5                 irlba_2.3.5.1
##  [41] GenomicRanges_1.52.0        labeling_0.4.2
##  [43] progressr_0.13.0           fansi_1.0.4
##  [45] spatstat.sparse_3.0-2     httr_1.4.6
##  [47] polyclip_1.10-4           abind_1.4-5
##  [49] compiler_4.3.3             withr_2.5.0
##  [51] highr_0.10                 R.utils_2.12.2
##  [53] MASS_7.3-60.0.1            DelayedArray_0.26.3
##  [55] tools_4.3.3                viper_0.4.5
##  [57] lmtest_0.9-40              beeswarm_0.4.0
##  [59] httpuv_1.6.11              future.apply_1.11.0
##  [61] goftest_1.2-3              R.oo_1.25.0
##  [63] glue_1.6.2                 nlme_3.1-164
##  [65] promises_1.2.0.1           grid_4.3.3
##  [67] Rtsne_0.16                 cluster_2.1.6
##  [69] reshape2_1.4.4              generics_0.1.3
##  [71] gtable_0.3.3               spatstat.data_3.0-1
##  [73] R.methodsS3_1.8.2           tidyR_1.3.0
##  [75] data.table_1.14.8          XVector_0.40.0
##  [77] sp_2.2-0                   utf8_1.2.3
##  [79] BiocGenerics_0.46.0         spatstat.geom_3.2-4
##  [81] RcppAnnoy_0.0.21            RANN_2.6.1
##  [83] pillar_1.9.0                stringr_1.5.0
##  [85] limma_3.56.2               spam_2.9-1
##  [87] later_1.3.1                rJava_1.0-6
##  [89] splines_4.3.3              lattice_0.22-5
##  [91] survival_3.5-8              deldir_1.0-9
##  [93] tidyselect_1.2.0             SingleCellExperiment_1.22.0
##  [95] miniUI_0.1.1.1              pbapply_1.7-2
##  [97] knitr_1.43                  gridExtra_2.3
##  [99] IRanges_2.34.0              SummarizedExperiment_1.30.2
## [101] scattermore_1.2             stats4_4.3.3

```

```
## [103] xfun_0.39
## [105] matrixStats_1.0.0
## [107] stringi_1.8.4
## [109] yaml_2.3.7
## [111] evaluate_0.21
## [113] tibble_3.2.1
## [115] uwot_0.1.16
## [117] reticulate_1.30
## [119] Rcpp_1.0.11
## [121] globals_0.16.2
## [123] png_0.1-8
## [125] parallel_4.3.3
## [127] dotCall64_1.0-2
## [129] listenv_0.9.0
## [131] scales_1.2.1
## [133] crayon_1.5.2
## [135] purrr_1.0.1
## [137] cowplot_1.1.1

## [103] xfun_0.39
## [105] matrixStats_1.0.0
## [107] stringi_1.8.4
## [109] yaml_2.3.7
## [111] evaluate_0.21
## [113] tibble_3.2.1
## [115] uwot_0.1.16
## [117] reticulate_1.30
## [119] Rcpp_1.0.11
## [121] globals_0.16.2
## [123] png_0.1-8
## [125] parallel_4.3.3
## [127] dotCall64_1.0-2
## [129] listenv_0.9.0
## [131] scales_1.2.1
## [133] crayon_1.5.2
## [135] purrr_1.0.1
## [137] cowplot_1.1.1
```