additional exercises for working with RNA and ADT data

```r
library(Seurat)
library(ggplot2)

# print the first 5 rows/columns of raw RNA data
cbmc@assays$RNA@counts[1:5,1:5]
```

```
## 5 x 5 sparse Matrix of class "dgCMatrix"
##          CTGTTTACACCGCTAG CTCTACGGTGTGGCTC AGCAGCCAGGCTCATT
## A1BG                    .                .                .
## A1BG-AS1                .                .                .
## A1CF                    .                .                .
## A2M                     .                .                .
## A2M-AS1                 .                .                .
##          GAATAAGAGATCCCAT GTGCATAGTCATGCAT
## A1BG                    .                .
## A1BG-AS1                .                .
## A1CF                    .                .
## A2M                     .                .
## A2M-AS1                 .                .
```

```r
# print the first 5 rows/columns of raw ADT data
cbmc@assays$ADT@counts[1:5,1:5]
```

```
## 5 x 5 sparse Matrix of class "dgCMatrix"
##        CTGTTTACACCGCTAG CTCTACGGTGTGGCTC AGCAGCCAGGCTCATT GAATAAGAGATCCCAT
## CD3                  60               52               89               55
## CD4                  72               49              112               66
## CD8                  76               59               61               56
## CD45RA              575             3943              682              378
## CD56                 64               68               87               58
##        GTGCATAGTCATGCAT
## CD3                  63
## CD4                  80
## CD8                  94
## CD45RA              644
## CD56                104
```

```r
# log normalize RNA data and print the same subset
cbmc <- NormalizeData(cbmc)
cbmc@assays$RNA@data[1:5,1:5]
```

```
## 5 x 5 sparse Matrix of class "dgCMatrix"
##          CTGTTTACACCGCTAG CTCTACGGTGTGGCTC AGCAGCCAGGCTCATT
## A1BG                    .                .                .
## A1BG-AS1                .                .                .
## A1CF                    .                .                .
## A2M                     .                .                .
## A2M-AS1                 .                .                .
##          GAATAAGAGATCCCAT GTGCATAGTCATGCAT
## A1BG                    .                .
## A1BG-AS1                .                .
## A1CF                    .                .
## A2M                     .                .
## A2M-AS1                 .                .
```

```
# CLR normalize ADT data and print the same subset
cbmc <- NormalizeData(cbmc, assay = "ADT", normalization.method = "CLR")
```

```
## Normalizing across features
```

```
cbmc@assays$ADT@data[1:5,1:5]
```

```
##        CTGTTTACACCGCTAG CTCTACGGTGTGGCTC AGCAGCCAGGCTCATT GAATAAGAGATCCCAT
## CD3           0.3411314        0.3018312        0.4718965        0.3167504
## CD4           0.3097615        0.2208286        0.4477722        0.2873132
## CD8           0.7908888        0.6604879        0.6767416        0.6356011
## CD45RA        0.2939403        1.2069275        0.3402440        0.2026413
## CD56          0.8350917        0.8698654        1.0202972        0.7805532
##        GTGCATAGTCATGCAT
## CD3           0.3554797
## CD4           0.3389295
## CD8           0.9126178
## CD45RA        0.3240441
## CD56          1.1380466
```

```
# scale RNA data and print the same subset
cbmc <- ScaleData(cbmc)
```

```
## Centering and scaling data matrix
```

```
cbmc@assays$RNA@scale.data[1:5,1:5]
```

```
##        CTGTTTACACCGCTAG CTCTACGGTGTGGCTC AGCAGCCAGGCTCATT GAATAAGAGATCCCAT
## A4GALT      -0.04300511      -0.04300511      -0.04300511      -0.04300511
## ABCB10      -0.12796529      -0.12796529      -0.12796529      -0.12796529
## ABCC3       -0.08952736      -0.08952736      -0.08952736      -0.08952736
## ABCG2       -0.04421835      -0.04421835      -0.04421835      -0.04421835
## ABI3        -0.31871010      -0.31871010      -0.31871010      -0.31871010
##        GTGCATAGTCATGCAT
## A4GALT      -0.04300511
## ABCB10      -0.12796529
## ABCC3       -0.08952736
## ABCG2       -0.04421835
## ABI3        -0.31871010
```

```
# scale ADT data and print the same subset
cbmc <- ScaleData(cbmc, assay = "ADT")
```

```
## Centering and scaling data matrix
```

```
cbmc@assays$ADT@scale.data[1:5,1:5]
```

```
##        CTGTTTACACCGCTAG CTCTACGGTGTGGCTC AGCAGCCAGGCTCATT GAATAAGAGATCCCAT
## CD3          -0.7350491       -0.7804836       -0.5838729      -0.76323574
## CD4          -0.8431841       -0.9652564       -0.6537459      -0.87399734
## CD8          -0.0608686       -0.1989948       -0.1817782      -0.22535594
## CD45RA       -0.8814259        0.4566811       -0.8135615      -1.01523700
## CD56          0.1397395        0.2080884        0.5037679       0.03254183
##        GTGCATAGTCATGCAT
## CD3         -0.71846116
## CD4         -0.80314707
## CD8          0.06807201
## CD45RA      -0.83730468
## CD56         0.73520890
```

```
# for this example I will use CD4 and CD8, as I already know there's a good amount
# of single positive cells for each as well as a few double positive cells.

# count number of cells positive for CD4
cd4plus <- WhichCells(cbmc, expression= CD4 > 0)
length(cd4plus)
```

```
## [1] 1102
```

```
# count number of cells positive for CD8
cd8plus <- WhichCells(cbmc, expression= CD8A > 0)
length(cd8plus)
```

```
## [1] 340
```

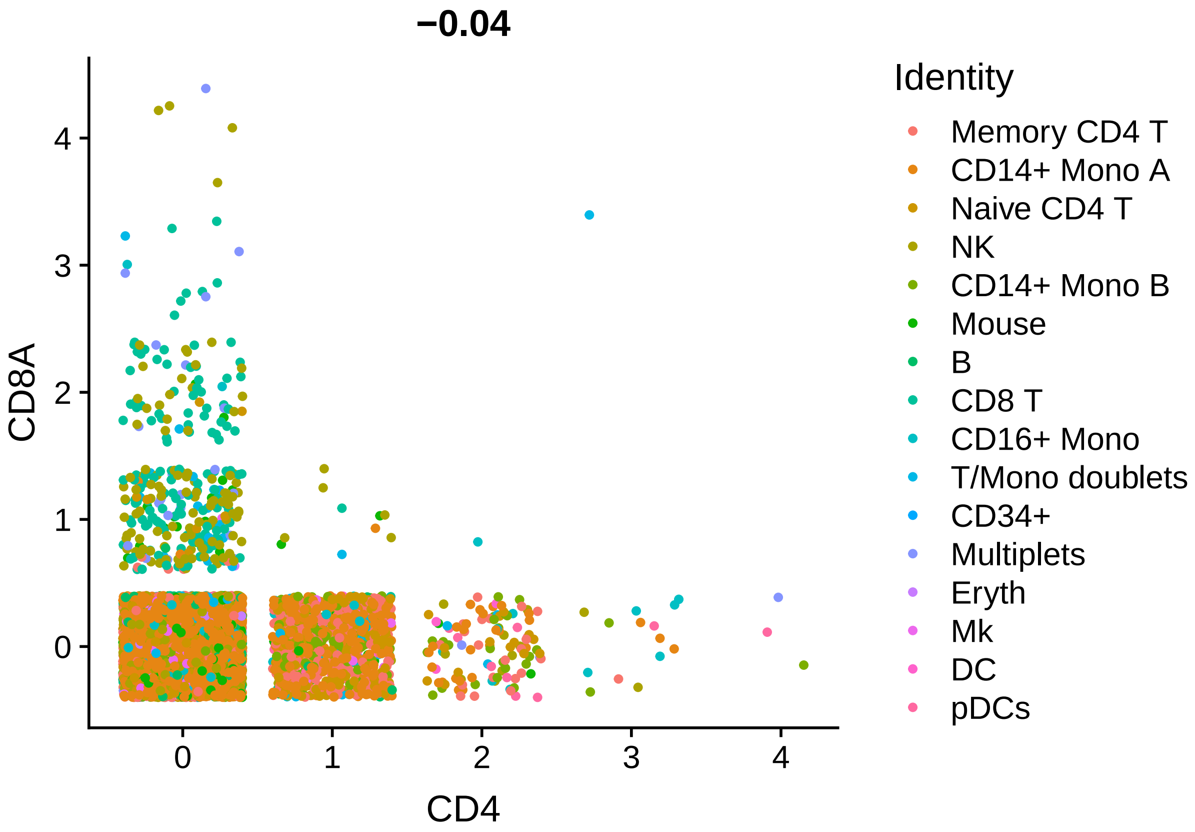
```
# count number of double positive cells
length(WhichCells(object = cbmc, cells = cd8plus, expression= CD4 > 0))
```

```
## [1] 12
```

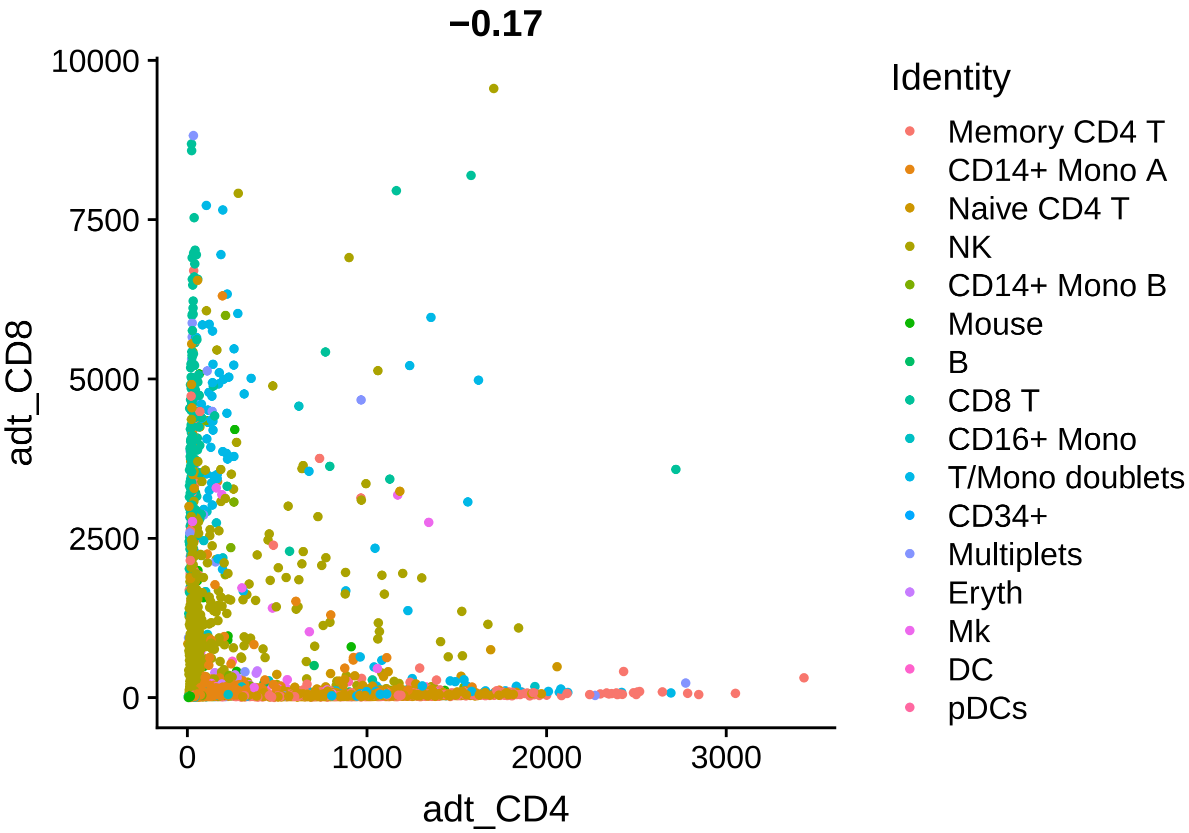
```
# count the number of double negative cells
cd4minus <- WhichCells(cbmc, expression= CD4 == 0)
cd4minuscd8minus <- WhichCells(cbmc, cd4minus, expression = CD8A == 0)
length(cd4minuscd8minus)
```

```
## [1] 7187
```

```
# let's do a scatter plot of CD4 vs CD8A RNA expression. These values seem to match up pretty well with
plot(FeatureScatter(cbmc, feature1 = "CD4", feature2 = "CD8A", slot = "counts", identity= "Raw CD4 vs CI
```



```
# Without subsetting the data, let's do a scatter plot of adt_CD4 vs adt_CD8.
plot(FeatureScatter(cbmc, feature1 = "adt_CD4", feature2 = "adt_CD8", slot = "counts", identity= "Raw C
```



```
# this seems pretty different from the RNA scatter plot, and there are definitely a lot more
# than 12 double positive cells. Let's try subsetting the data to only look at "T cells"
```

```
# first need to cluster based on RNA data
cbmc <- RunPCA(cbmc, verbose = FALSE)
cbmc <- FindNeighbors(cbmc, dims = 1:25)
```

```
## Computing nearest neighbor graph
```

```
## Computing SNN
```

```
cbmc <- FindClusters(cbmc, resolution = 0.8)
```

```
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 8617
## Number of edges: 347548
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.8592
## Number of communities: 19
## Elapsed time: 2 seconds
```

```
cbmc <- RunTSNE(cbmc, dims = 1:25, method = "FIt-SNE")
new.cluster.ids <- c("Memory CD4 T", "CD14+ Mono A", "Naive CD4 T", "NK", "CD14+ Mono B", "Mouse", "B",
    "CD8 T", "CD16+ Mono", "T/Mono doublets", "NK", "CD34+", "Multiplets", "Mouse", "Eryth", "Mk",
    "Mouse", "DC", "pDCs")
names(new.cluster.ids) <- levels(cbmc)
```

```
cbmc <- RenameIdents(cbmc, new.cluster.ids)

# Let's look at the number of CD4+ T cells (T helper cells):
length(WhichCells(cbmc, idents = "Naive CD4 T")) + length(WhichCells(cbmc, idents = "Memory CD4 T"))
```

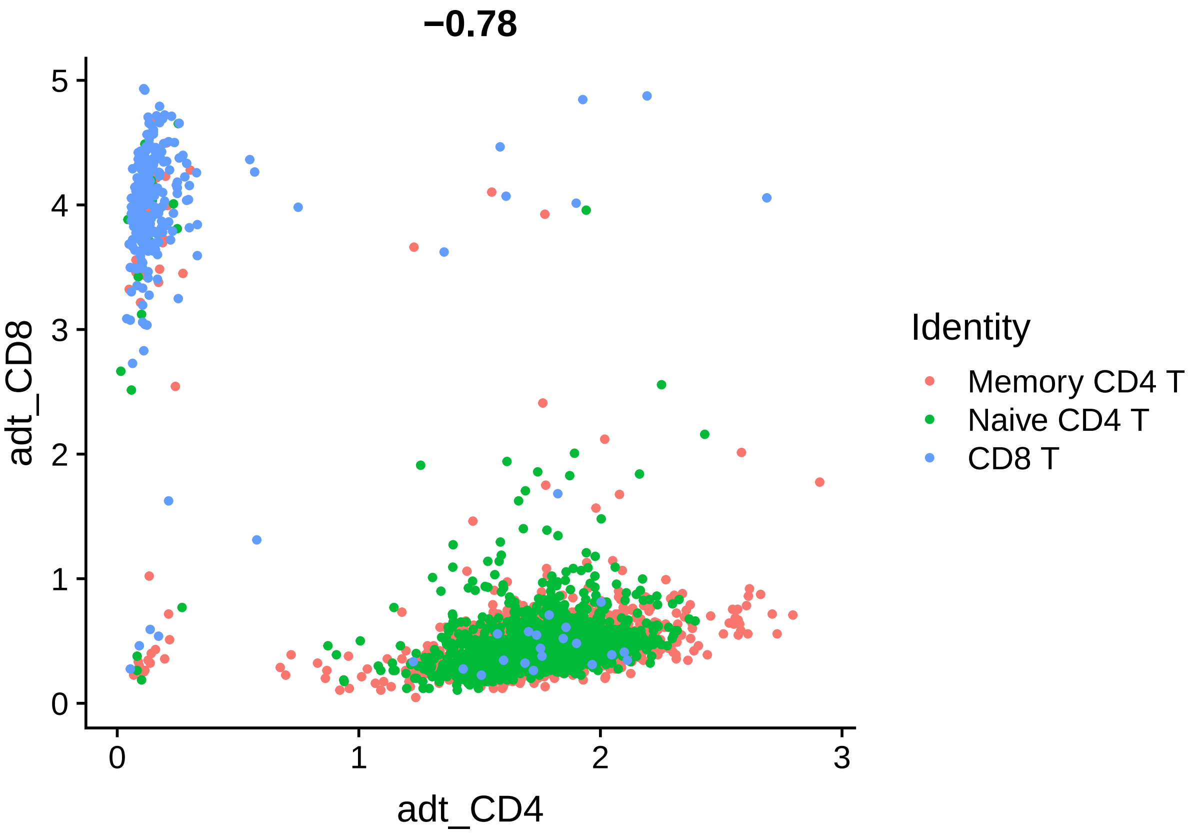
```
## [1] 3039
```

```
# Now let's see the number of CD8+ cells:
length(WhichCells(cbmc, idents = "CD8 T"))
```

```
## [1] 273
```

```
# RNA clustering identified about the same number of CD8+ cells, but
# around 3x the number of CD4+ cells. This could be due to high dropout of CD4 mRNAs.

# Now we'll do another ADT feature scatter based on only the CD4+, CD8+, and memory T cell data.
# This should look a lot more like the RNA plot, because it will be adjusted to account for
# the higher protein copy number (compared to mRNA count).
tcells <- subset(cbmc, idents = c("Naive CD4 T", "Memory CD4 T", "CD8 T"))
FeatureScatter(tcells, feature1 = "adt_CD4", feature2 = "adt_CD8")
```



```
# Now let's have a look at the top 5 gene markers for three phagocyte subsets.
# We'll examine CD16+ Monocytes and two groups of CD14+ Monocytes.
phagocytes <- subset(cbmc, idents = c("CD14+ Mono A", "CD14+ Mono B", "CD16+ Mono"))
phagocytes <- FindVariableFeatures(phagocytes, selection.method = "vst", nfeatures = 2000)
# find all markers of CD14 + Monocytes Group A
cd14a.markers <- FindMarkers(phagocytes, ident.1 = "CD14+ Mono A", min.pct = 0.25)
cd14b.markers <- FindMarkers(phagocytes, ident.1 = "CD14+ Mono B", min.pct = 0.25)
```

```
cd16.markers <- FindMarkers(phagocytes, ident.1 = "CD16+ Mono", min.pct = 0.25)
head(cd14a.markers, n = 5)
```

```
##                  p_val avg_logFC pct.1 pct.2     p_val_adj
## S100A8   5.753004e-282  1.753127 1.000 0.898 1.179423e-277
## S100A9   6.940300e-252  1.386136 0.999 0.947 1.422831e-247
## HLA-DPA1 9.193169e-252 -1.637089 0.356 0.938 1.884692e-247
## HLA-DPB1 1.469596e-238 -1.664394 0.316 0.908 3.012819e-234
## S100A12  1.530059e-212  1.688431 0.940 0.578 3.136773e-208
```

```
head(cd14b.markers, n = 5)
```

```
##                  p_val avg_logFC pct.1 pct.2     p_val_adj
## HLA-DRA  1.317083e-160 0.9489242 0.998 0.889 2.700151e-156
## HLA-DRB1 3.114205e-160 0.9936157 1.000 0.796 6.384431e-156
## HLA-DPB1 2.589395e-137 1.1522385 0.902 0.388 5.308518e-133
## CD74     1.026933e-135 0.8036277 0.998 0.901 2.105315e-131
## HLA-DPA1 2.101442e-132 1.0307715 0.931 0.427 4.308166e-128
```

```
head(cd16.markers, n = 5)
```

```
##                p_val avg_logFC pct.1 pct.2     p_val_adj
## FCGR3A  0.000000e+00  2.481356 0.865 0.046  0.000000e+00
## CDKN1C 2.563935e-199  1.355555 0.509 0.015 5.256323e-195
## HES4   2.195841e-163  1.184574 0.487 0.023 4.501694e-159
## S100A9 1.118468e-125 -2.733032 0.839 0.997 2.292972e-121
## S100A8 4.583887e-125 -3.279581 0.683 0.996 9.397426e-121
```

```
# load 10x data that I downloaded from their website
pbmc10k.data <- Read10X(data.dir = "~/HarderLab/singlecellgenomicspractice/multimodal_tutorial/filtered_
```

```
## 10X data contains more than one type and is being returned as a list containing matrices of each type
```

```
rownames(x = pbmc10k.data[["Antibody Capture"]]) <- gsub(pattern = "_[control_]*TotalSeqB", replacement
    x = rownames(x = pbmc10k.data[["Antibody Capture"]]))

# load a seurat object with chosen cutoff values
pbmc10k <- CreateSeuratObject(counts = pbmc10k.data[["Gene Expression"]], min.cells = 3, min.features =

# log normalize RNA data
pbmc10k <- NormalizeData(pbmc10k)

# create ADT assay object
pbmc10k[["ADT"]] <- CreateAssayObject(pbmc10k.data[["Antibody Capture"]][, colnames(x = pbmc10k)])

# CLR normalize ADT data
pbmc10k <- NormalizeData(pbmc10k, assay = "ADT", normalization.method = "CLR")
```

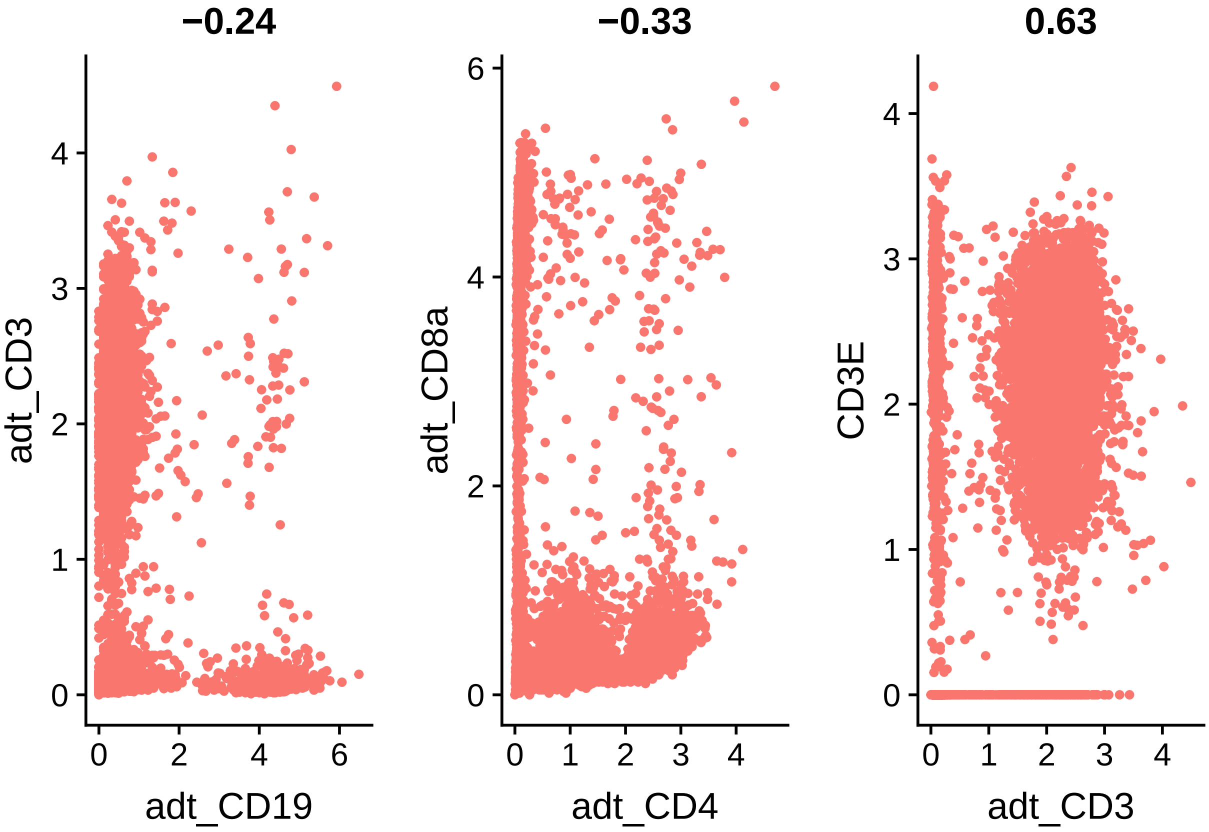
```
## Normalizing across features
```

```
# plot CD19 ADT vs CD3 ADT
# should show strong separation because CD3 is expressed on T cells, CD19 on B cells
plot1 <- FeatureScatter(pbmc10k, feature1 = "adt_CD19", feature2 = "adt_CD3", pt.size = 1)

# plot CD4 ADT vs CD8A ADT
# Should be about the same as we saw before, a lot of double negative/single positive
plot2 <- FeatureScatter(pbmc10k, feature1 = "adt_CD4", feature2 = "adt_CD8a", pt.size = 1)
```

```
# plot CD3 ADT against CD3E mRNA reads
# these can be plotted together because they've both been normalized
plot3 <- FeatureScatter(pbmc10k, feature1 = "adt_CD3", feature2 = "CD3E", pt.size = 1)

# display plots side by side
plot(CombinePlots(plots = list(plot1, plot2, plot3), ncol = 3, legend = "none"))
```


adt_CD3
adt_CD19
adt_CD8a
adt_CD4
CD3E
adt_CD3