Cytokine Activity Estimation using SCAPE (Single cell transcriptomics-level Cytokine Activity Prediction and Estimation): An Analysis using the CytoSig database.

Azka Javaid and H. Robert Frost

```
library(scaper)
library(Seurat)
library(SeuratObject)
library(pheatmap)
```

In this vignette, we perform gene set scoring for 41 cytokines using the CytoSig database, detailing the genes and the associated log2 fold change values upon treatment, for a sample of the Peripheral Blood Mononuclear Cells (PBMC) dataset.

In the `scaper` package, we detail the development of five functions. The first function, `supportedCytokines`, returns a list of the scored cytokines for the CytoSig and the Reactome databases. Next two functions, `genesetCytoSig` and `genesetReactome`, perform gene set construction for the CytoSig and the Reactome databases, respectively. Lastly, the `scapeForSeurat` function, performs gene set scoring using the constructed databases with the Seurat framework integrated. In comparison, the `scape` function performs gene set scoring with the normalized counts matrix (i.e., without relying on the Seurat framework).

We first show the functionality of our package using an example function call for the `geneCytoSig` function. In the call below, we first find a list of all scored cytokines for the CytoSig database. We next create a gene set for IL6 and extract the top 10 targets that are differentially expressed upon treatment with the cytokine using the CytoSig database. This function is currently operational for the IL4 and IL6 cytokines. For all other cytokines, specify the cytokine-specific output file.

```
supportedCytokines(database = "cytosig")
#>  [1] "ActivinA" "BDNF"     "BMP2"     "BMP4"     "BMP6"     "CD40LG"
#>  [7] "CXCL12"   "EGF"      "FGF2"     "GCSF"     "GDF11"    "GMCSF"
#> [13] "HGF"      "IFNG"     "IFNL"     "IL10"     "IL12"     "IL13"
#> [19] "IL15"     "IL17A"    "IL1A"     "IL1B"     "IL2"      "IL21"
#> [25] "IL22"     "IL27"     "IL3"      "IL4"      "IL6"      "LIF"
#> [31] "LTA"      "MCSF"     "NO"       "OSM"      "TGFB1"    "TGFB3"
#> [37] "TNFA"     "TNFSF12"  "TRAIL"    "VEGFA"    "WNT3A"
genesetCytoSig(cytokine.eval = "IL6",
               file.name = system.file("extdata", "IL6_output.csv",
                                       package = "scaper")) %>% head(10)
#>       gene weight cytokine
#> 1    TEAD3  6.218      IL6
#> 2   SHROOM3  5.65      IL6
#> 3     CCL8  5.051      IL6
#> 4    HSPA1A  4.927      IL6
#> 5     SGPL1  4.817      IL6
#> 6     PNMT   4.75      IL6
#> 7     CCL4  4.714      IL6
#> 8     IL1B   4.65      IL6
```
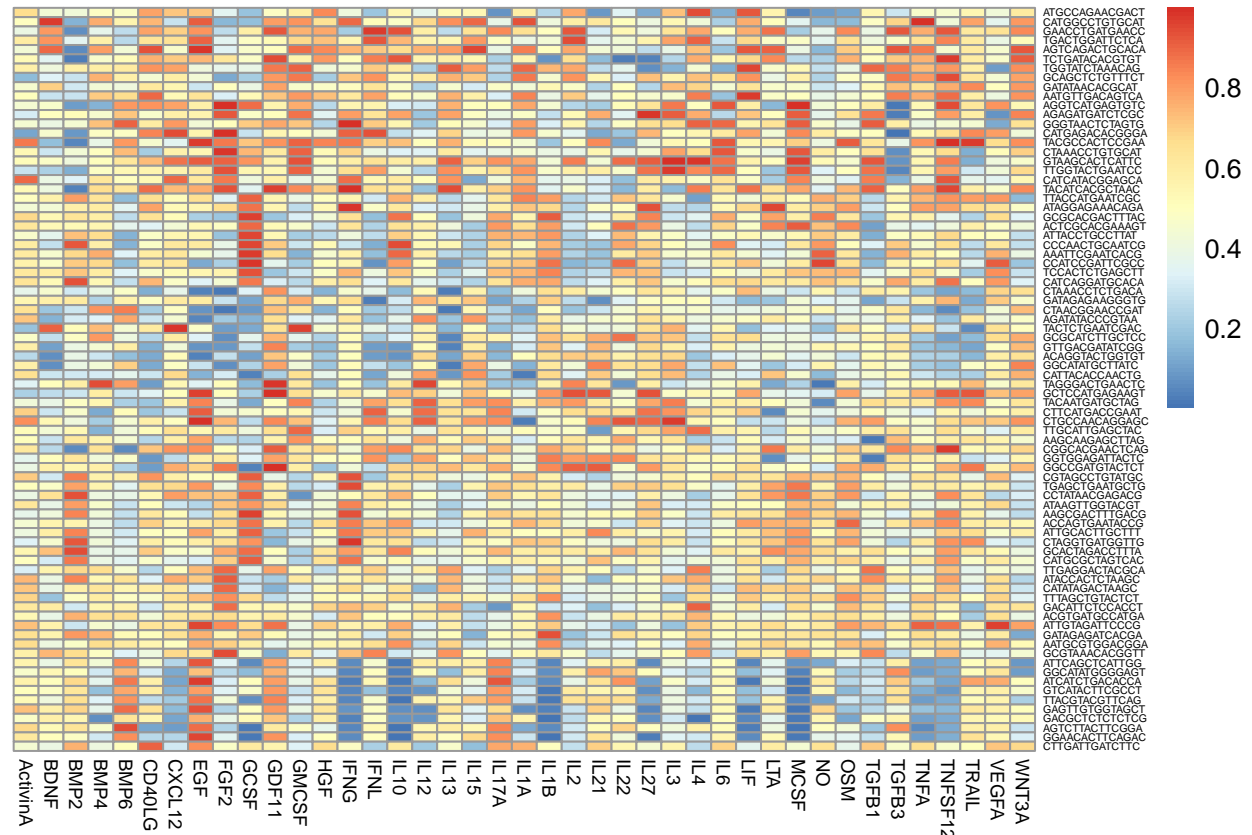
```
#> 9     ICOS  4.454     IL6
#> 10  HSPA1B  4.388     IL6
```

Next, we perform gene set scoring using the `scapeForSeurat` function, which accepts a Seurat object, with the CytoSig gene sets on the `pbmc_small` sampled dataset from the Seurat package, which contains about 230 features/genes assessed over 80 samples/cells. The output is a `scape` assay in the Cytosig.score.output object which is 41 by 80 and contains scored expression for 41 cytokines in 80 cells. As indicated below, we can extract that object and create an unclustered and clustered representation of the heatmap constructed based on the output scores.
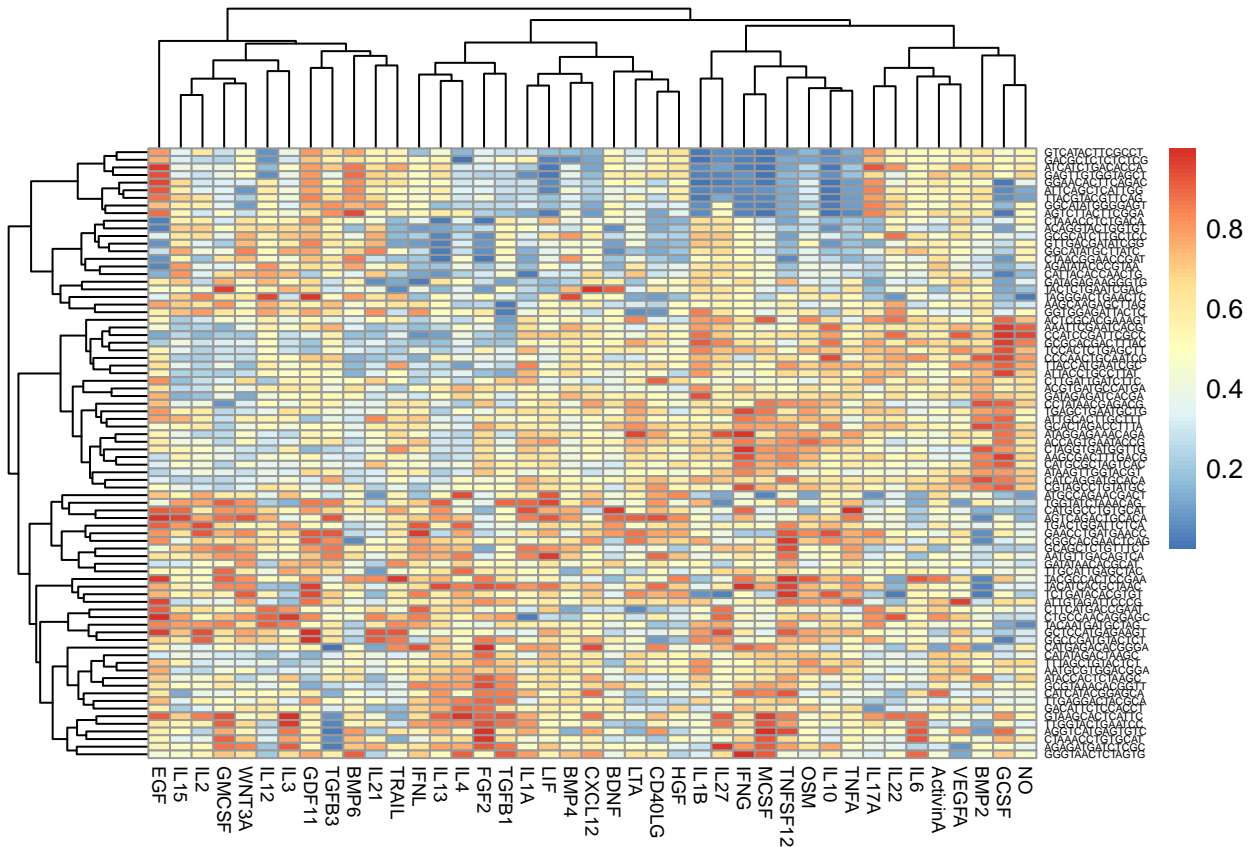
```
CytoSig.score.output <- scapeForSeurat(seurat.object = pbmc_small,
                                       database = "cytosig", cytokine = "all", normalize = TRUE)
class(CytoSig.score.output)
#> [1] "Seurat"
#> attr(,"package")
#> [1] "SeuratObject"
GetAssay(object = CytoSig.score.output, assay = "scape")
#> Assay data with 41 features for 80 cells
#> First 10 features:
#>  ActivinA, BDNF, BMP2, BMP4, BMP6, CD40LG, CXCL12, EGF, FGF2, GCSF
#DefaultAssay(CytoSig.score.output) <- "scape"
#GetAssayData(object = CytoSig.score.output, slot = "data")
cytosig_mat <- as.data.frame(t(as.matrix(CytoSig.score.output@assays$scape@data)))
pheatmap(cytosig_mat, fontsize_row = 4, fontsize_col = 7,
         cluster_rows = FALSE, cluster_cols = FALSE)
```



```
pheatmap(cytosig_mat, fontsize_row = 4, fontsize_col = 7)
```

Lastly, we perform gene set scoring using the `scape` function which accepts a m by n matrix as input (as compared to a Seurat object), with m samples and n genes.

```
pbmc_small <- NormalizeData(pbmc_small)
counts.matrix2 <- as.data.frame(t(as.matrix(pbmc_small@assays$RNA@data)))
CytoSig.score.output <- scape(counts.matrix = counts.matrix2,
                              database = "cytosig", cytokine = "all")
pheatmap(CytoSig.score.output, fontsize_row = 4, fontsize_col = 7,
         cluster_rows = FALSE, cluster_cols = FALSE)
```