

## 3.2 Tutorial-2: scRNA longitudinal data (n=4 and 6 weeks follow-up)

This tutorial allows users to explore single cell RNAseq data measured from 4 healthy donors over 6 time points (week 2-7). Single cell data available at [GSE190992](#). (1) AIFI-scRNA-PBMC-FinalData.RDS (Normalized scRNA seurat object) (2) [AIFI-Metadata.Rda](#) (clinical metadata). Longitudinal data set includes 4 donors and 24 samples. To infer inter-donor, intra-donor variations, and stable features, please follow following steps.

### 3.2.1 Load Library

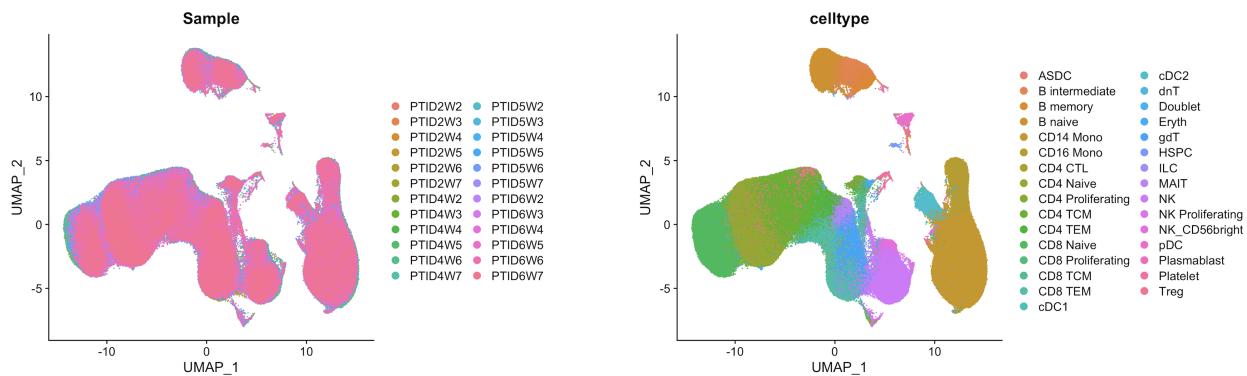
```
#Load Library
library("PALMO")
library("Hmisc")
library("ggpubr")
library("cowplot")
```

### 3.2.2 Load single cell data and metadata

```
#scRNA seurat object
pbmc <- readRDS("data/AIFI-scRNA-PBMC-FinalData.RDS")
metaData <- pbmc@meta.data
pbmc@meta.data$Sample <- pbmc@meta.data$orig.ident
pbmc@meta.data$celltype <- gsub(" ", "_", pbmc@meta.data$celltype)

#Load annotation data
load("data/AIFI-Metadata.Rda")

library("Seurat")
#UMAP plot
p1 <- DimPlot(object = pbmc, reduction = 'umap', group.by = "Sample", label = F)
p2 <- DimPlot(object = pbmc, reduction = 'umap', group.by = "celltype", label = F)
print(plot_grid(p1, p2, align="hv", ncol=2))
```



```
avgGroup <- "celltype"
#Celltypes observed in dataset
cell_type <- sort(unique(pbmc@meta.data$celltype))
#Celltypes selected for analysis consisting atleast >5% of cells in each celltype.
celltype_oi <- c("CD4_Naive", "CD4_TEM", "CD4_TCM", "CD4_CTL", "CD8_Naive",
                 "CD8_TEM", "CD8_TCM", "Treg", "MAIT", "gdT",
```

```

    "NK", "NK_CD56bright",
    "B_naive", "B_memory", "B_intermediate",
    "CD14_Mono", "CD16_Mono",
    "cDC2", "pDC")

```

### 3.2.3 Create PALMO object (Time ~ 2min)

```

#Create PALMO object
palmo_obj <- createPALMOobject(anndata=ann, data=pbmc)

#Assign Sample, PTID and Time parameters
palmo_obj <- annotateMetadata(data_object=palmo_obj,
                                sample_column= "Sample",
                                donor_column= "PTID",
                                time_column= "Time")

#Sample overlap and final matrix

```

For single cell data merge annotation and single cell metadata

```

palmo_obj <- mergePALMOdata(data_object=palmo_obj, datatype="singlecell")

#Aggregate data (Psuedo-bulk)

```

Aggregated samples noted by sample group. Define sample group and Calculate average expression. Keep genes with avgExpression > zero.

```

palmo_obj <- avgExpCalc(data_object=palmo_obj,
                          assay="RNA", group_column="celltype")
head(palmo_obj@curated[["anndata"]]) #merged annotation data
head(palmo_obj@curated[["data"]]) #scRNA average expression data

```

```

#Check for replicates
palmo_obj <- checkReplicates(data_object=palmo_obj, mergeReplicates = T)

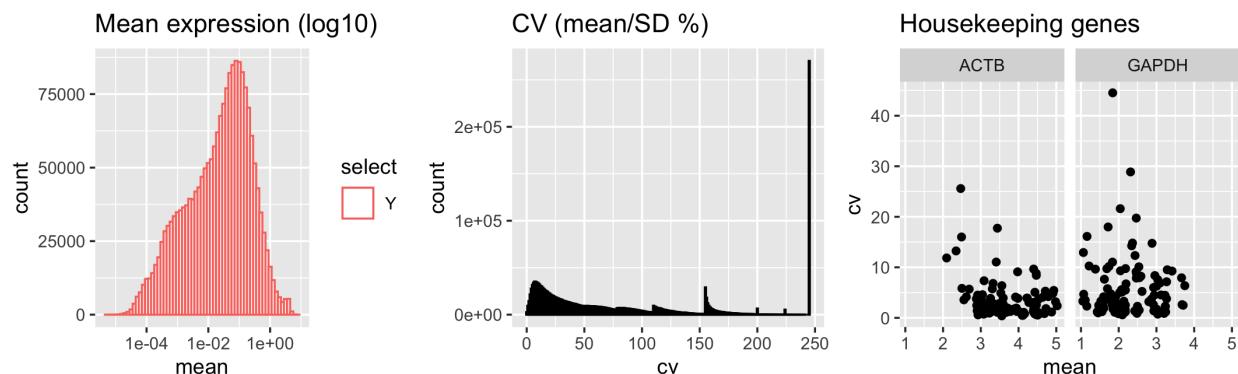
```

### 3.2.4 CV profile (Time ~ 5min)

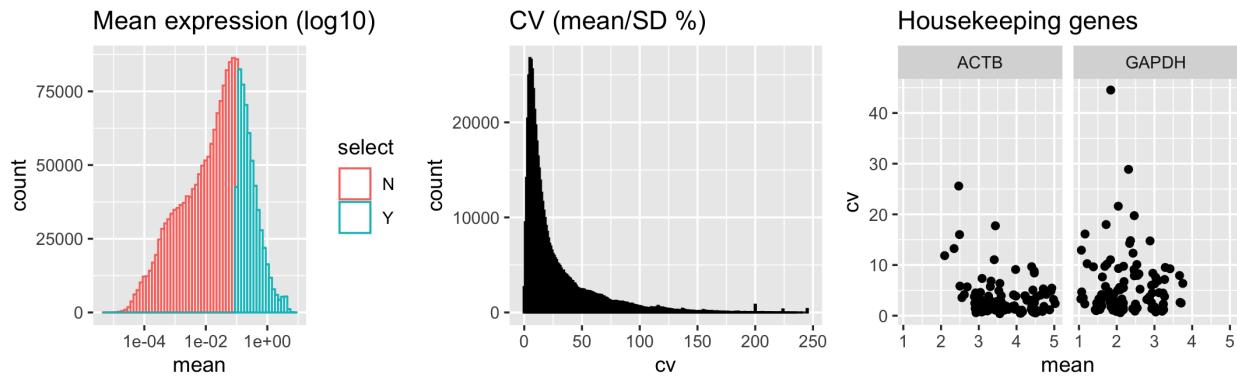
```

#Check the mean expression and CV cross groups (celltypes)
palmo_obj <- cvCalcSCPProfile(data_object=palmo_obj,
                                 housekeeping_genes=c("GAPDH", "ACTB"),
                                 fileName="scrna")

```



```
#Lowly expressed genes show abnormal CV, which needs to be filtered.
palmo_obj <- cvCalcSCPProfile(data_object=palmo_obj, meanThreshold = 0.1,
                                housekeeping_genes=c("GAPDH", "ACTB"),
                                fileName="scrna")
```



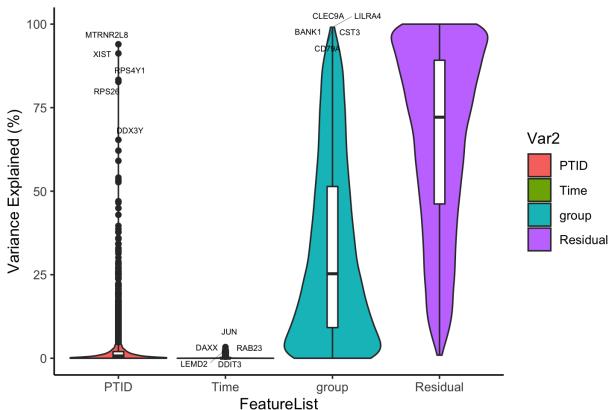
### 3.2.5 Donorwise CV profile over longitudinal timepoints (Time ~ 4min)

```
library("ggrepel")
#Sample Celltype Mean-CV plot (check output folder)
cvSCsampleprofile(data_object=palmo_obj, meanThreshold = 0.1,
cvThreshold = 10)
```

### 3.2.6 Features contributing towards donor variations (Time ~ 10min)

```
#Check the group of interest
head(palmo_obj@curated$anndata)

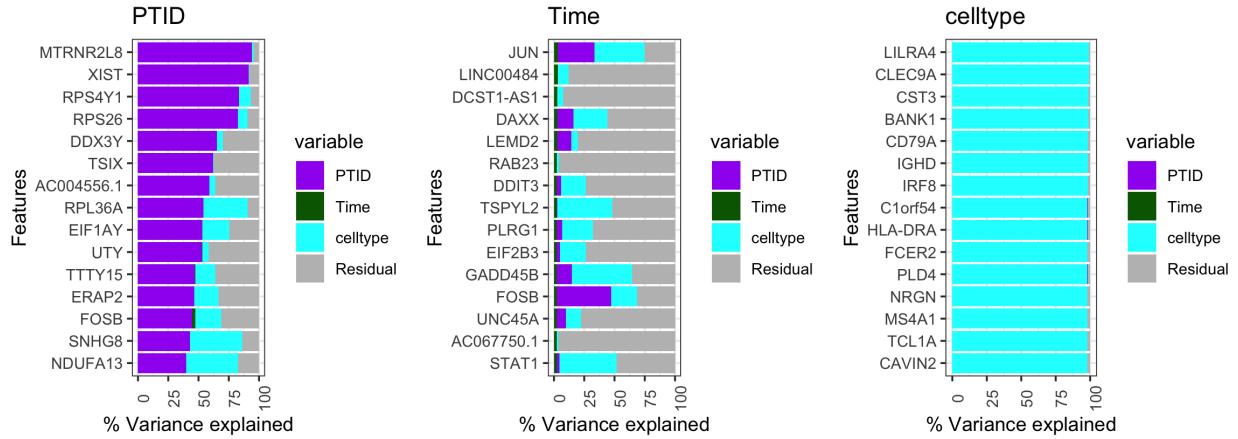
#Variance decomposition
featureSet <- c("PTID", "Time", "celltype")
palmo_obj <- lmeVariance(data_object=palmo_obj,
                           featureSet=featureSet,
                           meanThreshold=0.1, cl=4,
                           fileName="scrna")
var_decomp <- palmo_obj$result$variance_decomposition
```



### 3.2.7 Variance plot

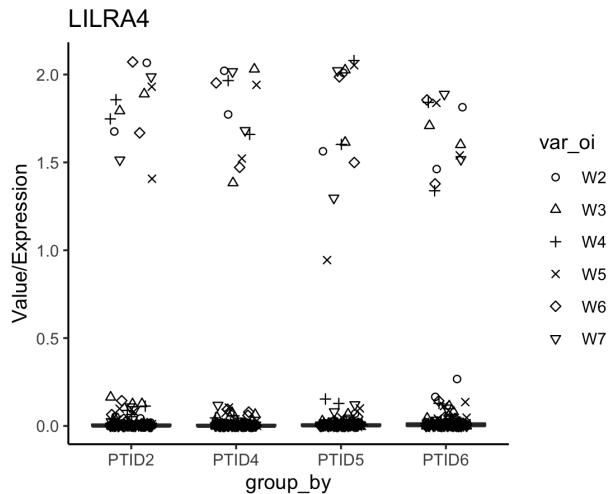
Donor-specific, Time- and celltype-attributed variance contributing features.

```
plots <- variancefeaturePlot(vardata=var_decomp, featureSet=featureSet,
                               Residual=F, cols=c("purple", "darkgreen", "cyan"))
plot_grid(plotlist = plots, ncol=3)
```

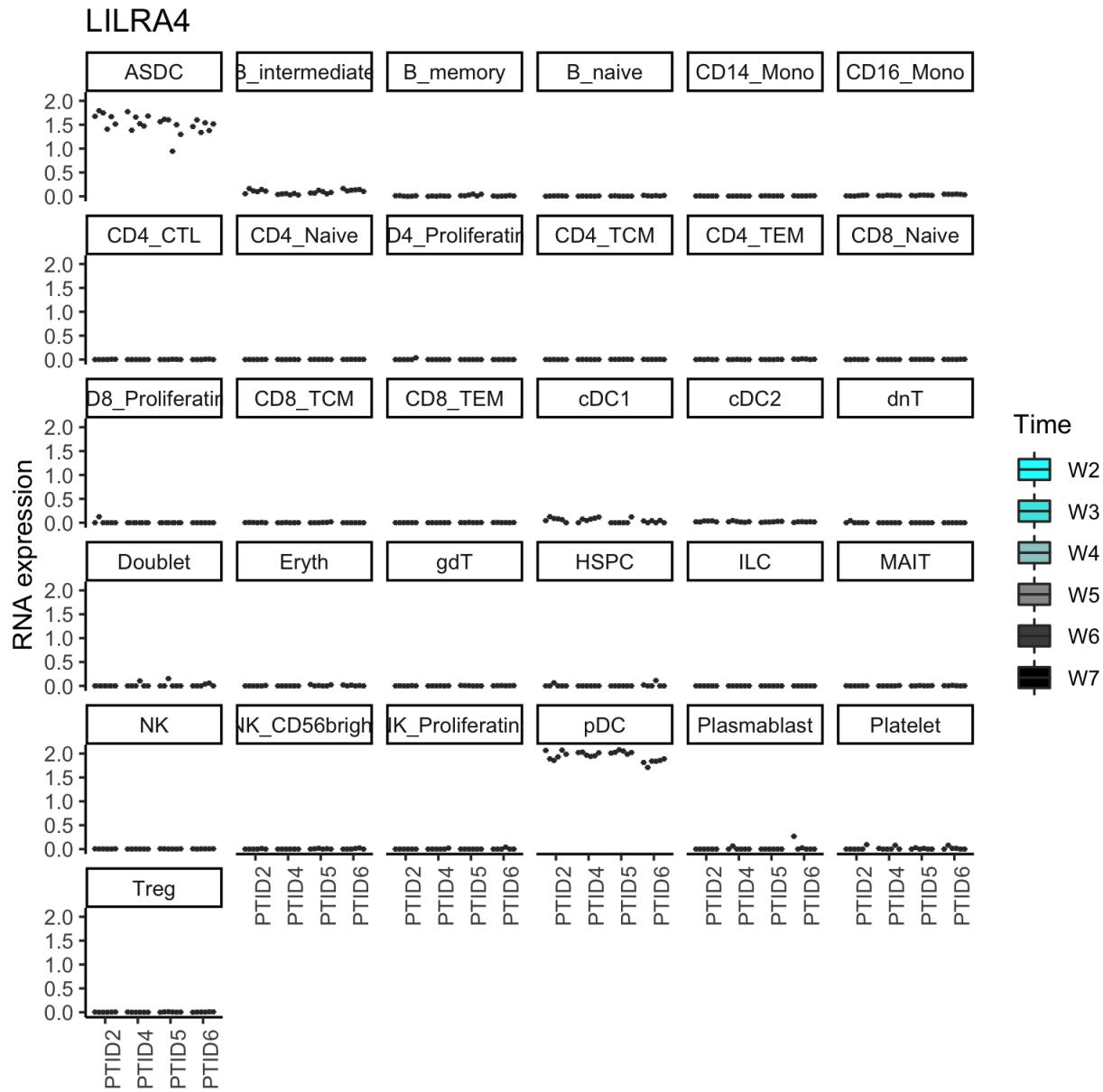


### 3.2.8 Plot the top features

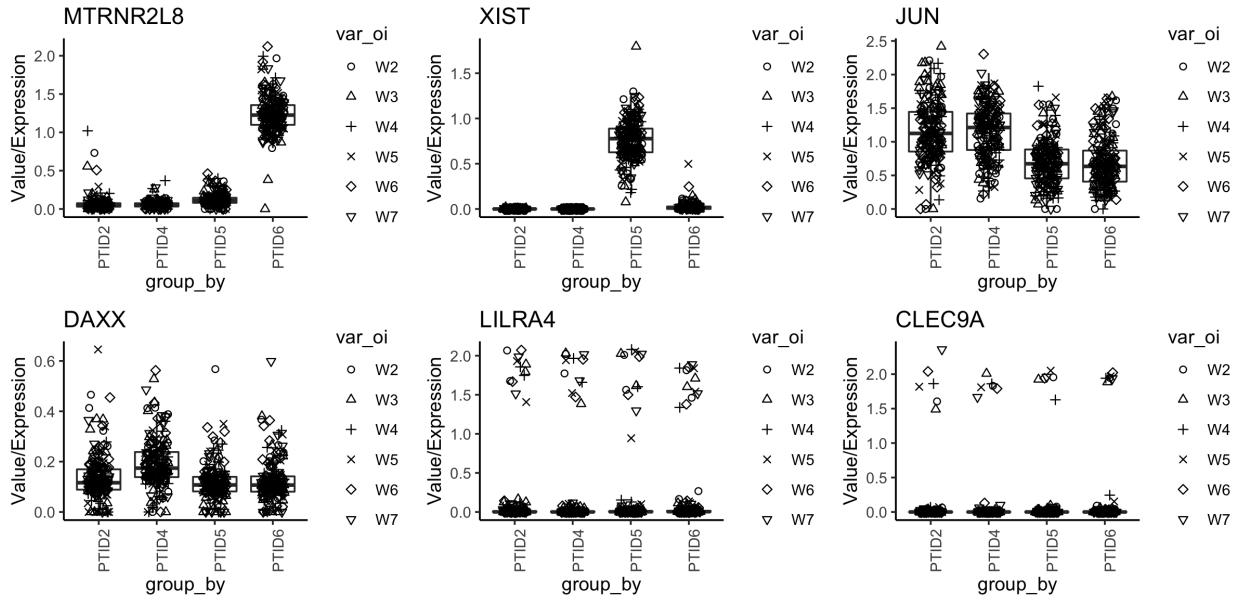
```
gene_featureplot(data_object=palmo_obj, featureList="LILRA4")
```



```
gene_featureplot(data_object=palmo_obj, featureList="LILRA4", facet_by="group")
```



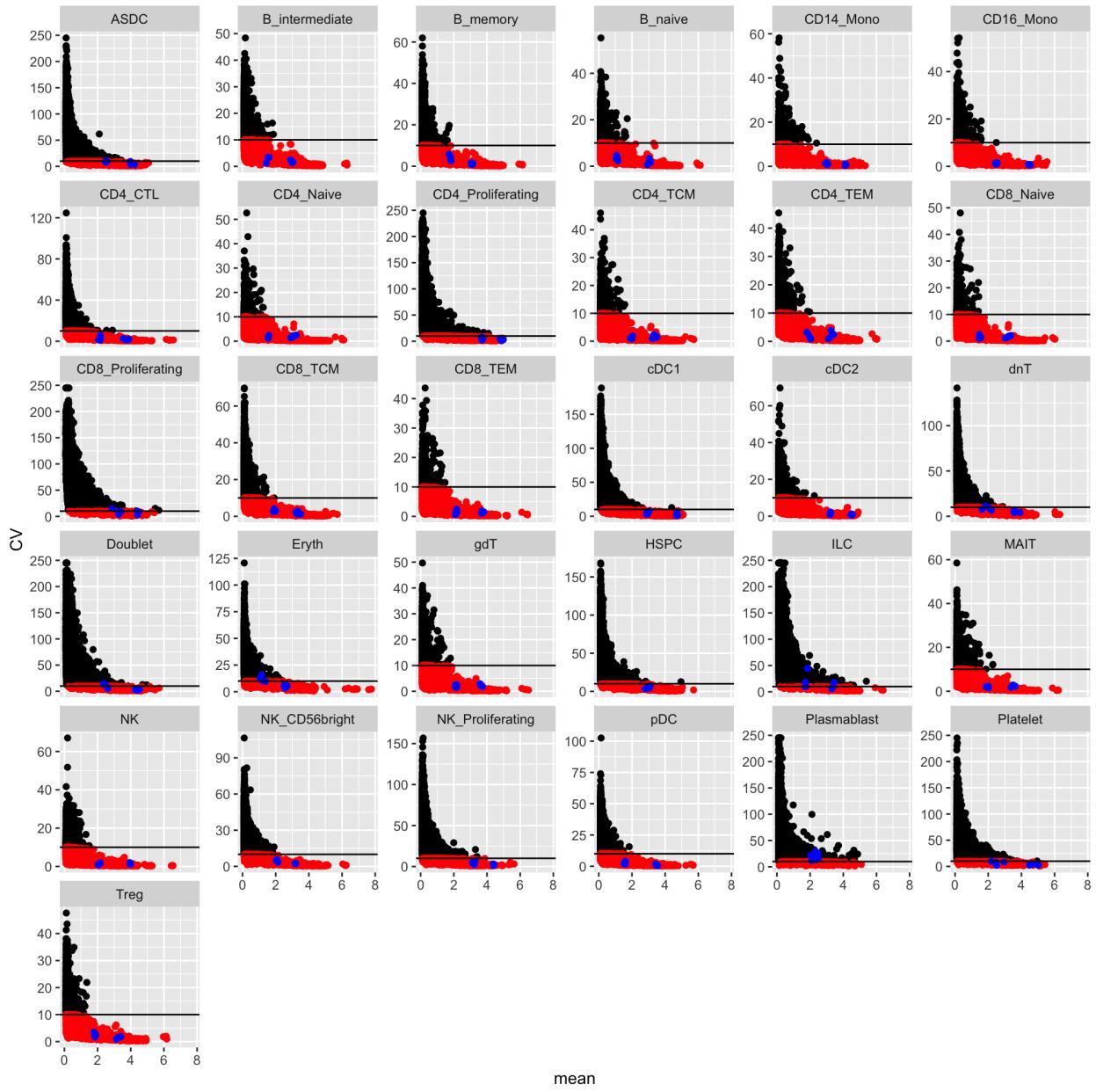
```
#Multiple-gene visualization
plots <- gene_featureplot(data_object=palmo_obj,
                            featureList=c("MTRNR2L8", "XIST",
                                         "JUN", "DAXX",
                                         "LILRA4", "CLEC9A"),
                            x_group_by="PTID", var_oi="Time", x_text_angle=90)
plot_grid(plotlist=plots, ncol= 3, align="hv")
```



### 3.2.9 Intra-donor variations over time (Time ~10min)

```
#Calculate CV
palmo_obj <- cvCalcSC(data_object=palmo_obj,
meanThreshold=0.1, cvThreshold=10,
housekeeping_genes=c("GAPDH", "ACTB"),
fileName="scRNA")
```

Plots saved in user-defined output directory



CV profile in celltypes (black) as well as CV for house-keeping genes (blue). Based on CV of house-keeping genes 10% CV cut-off used and genes considered stable below 10% CV.

### 3.2.10 Find stable and variable features in longitudinal data (Time <1 min)

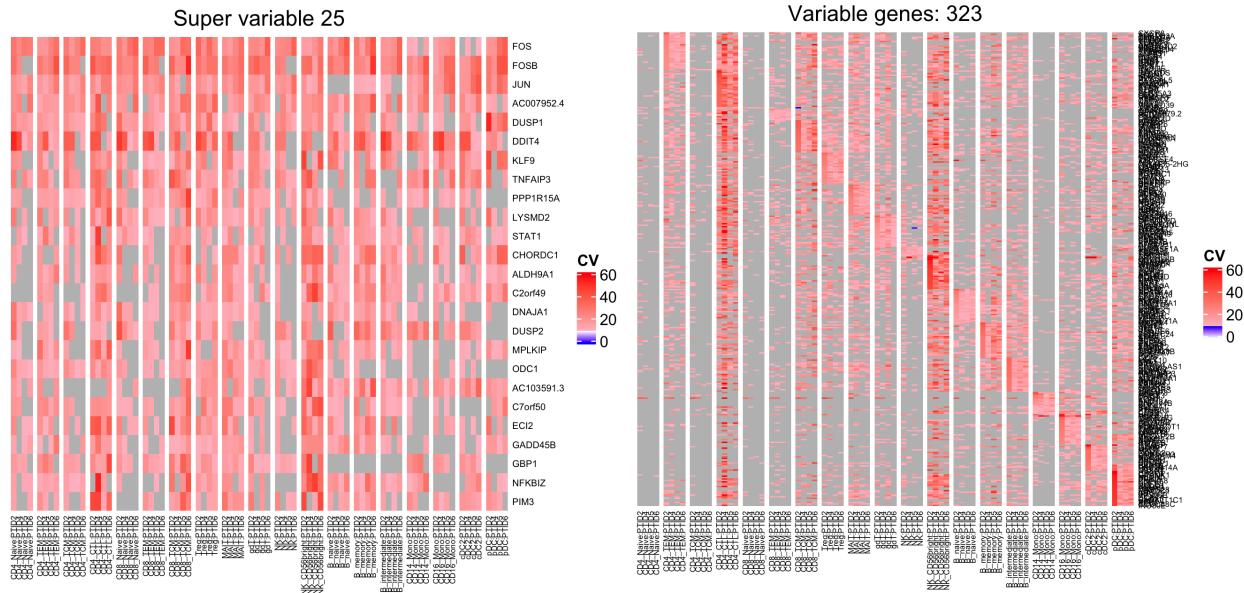
```

donorThreshold <- 4
groupThreshold <- 40 #number of donors * number of celltypes/2 (4x19/2)

palmo_obj <- VarFeatures(data_object=palmo_obj, group_oi=celltype_oi,
                           cvThreshold=10,
                           donorThreshold=4, groupThreshold=40,
                           topFeatures=25,
                           fileName="scrna")
var_genes <- palmo_obj@result$var_genes

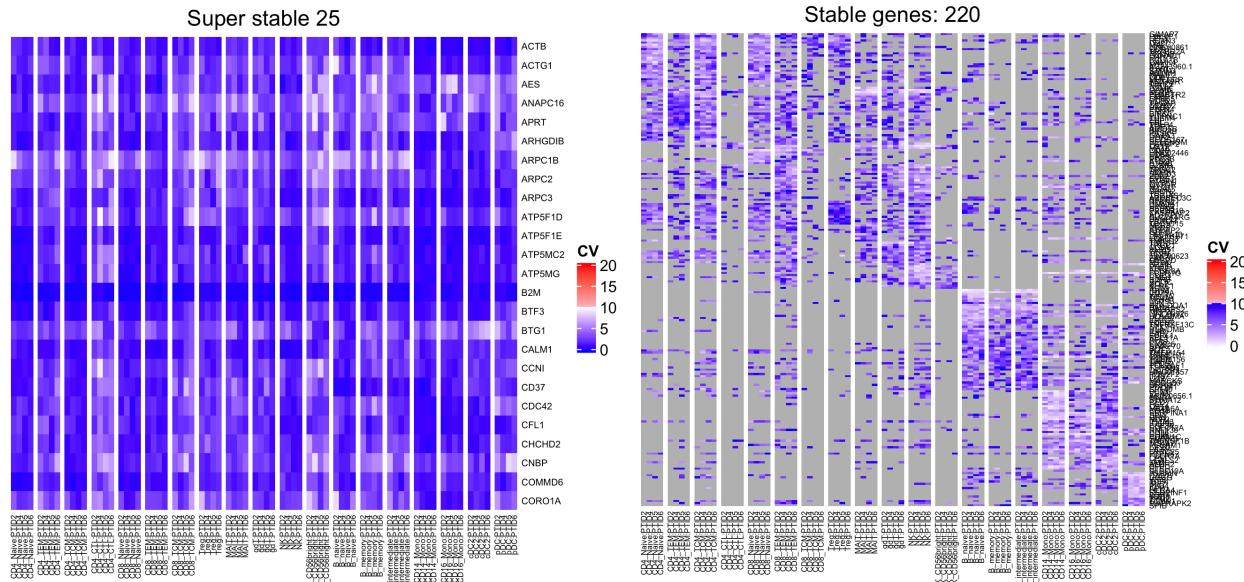
```

Variable genes observed in longitudinal data (CV>10%)



```
palmo_obj <- StableFeatures(data_object=palmo_obj, group_oi=celltype_oi,
                               cvThreshold=10,
                               donorThreshold=4, groupThreshold=40,
                               topFeatures=25,
                               fileName="scRNA")
stable_genes <- palmo_obj@result$stable_genes
```

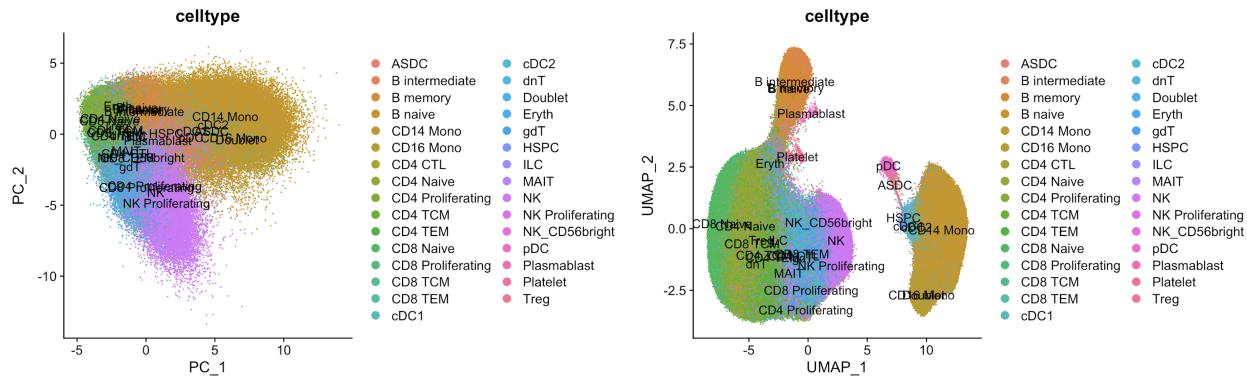
Stable genes observed in longitudinal data (CV<10%)



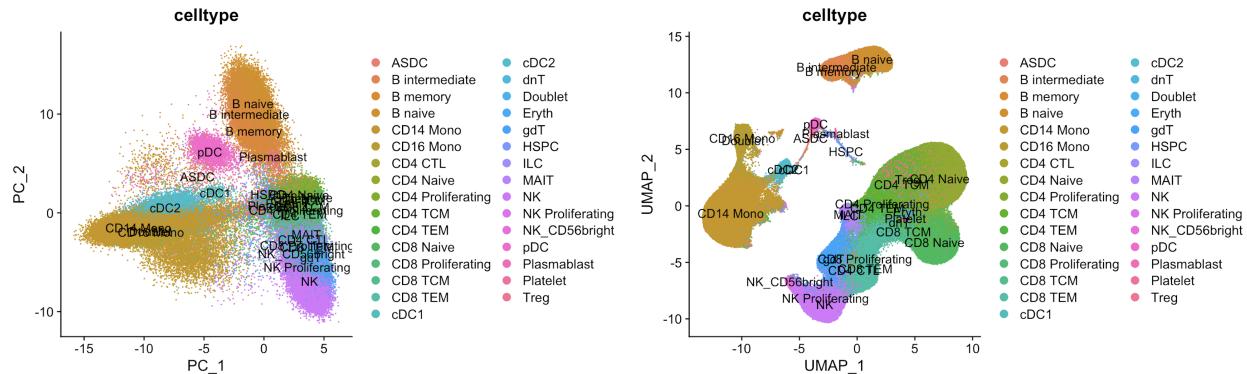
### 3.2.11 UMAP Plot (Stable/variable) (Time ~ 5min)

```
group_column <- "celltype"
```

```
#Top variable and stable features used for UMAP
dimUMAPPPlot(data_object=palmo_obj, nPC=15,
              gene_oi=unique(var_genes$gene),
              group_column=group_column, plotname="variable",
              fileName="scRNA")
```

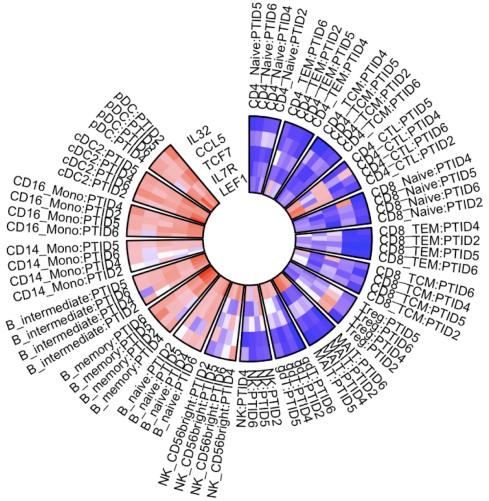


```
dimUMAPPPlot(data_object=palmo_obj, nPC=15,
              gene_oi=unique(stable_genes$gene),
              group_column=group_column, plotname="stable",
              fileName="scRNA")
```



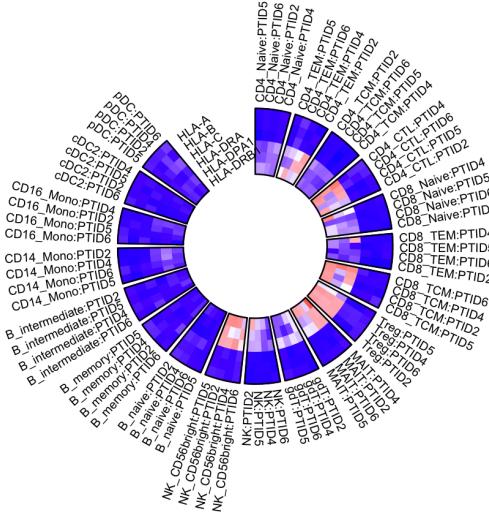
### 3.2.12 Circular gene expression plot

```
geneList <- c("IL32", "CCL5", "TCF7", "IL7R", "LEF1") #T-cell
plotres <- genecircosPlot(data_object=palmo_obj, geneList=geneList,
                           group_oi=celltype_oi, colorThreshold=10)
```



#Users can also load PALMO output result

```
cv_res <- palmo_obj@result[["cv_all"]]
geneList <- c("HLA-A", "HLA-B", "HLA-C", "HLA-DRA", "HLA-DPA1", "HLA-DRB1")
plotres <- genecircosPlot(data=cv_res, geneList=geneList,
                           titleName="HLA", group_oi=celltype_oi,
                           colorThreshold=10)
```



## 6 Session info

```
sessionInfo()
#> R version 4.0.3 (2020-10-10)
#> Platform: x86_64-apple-darwin17.0 (64-bit)
#> Running under: macOS Catalina 10.15.7
#>
#> Matrix products: default
#> BLAS:    /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRblas.dylib
#> LAPACK:  /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
#>
#> locale:
#> [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
#>
#> attached base packages:
#> [1] grid      stats     graphics  grDevices  utils      datasets  methods
#> [8] base
#>
#> other attached packages:
#> [1] PALMO_0.99.0
#>
#> loaded via a namespace (and not attached):
#> [1] readxl_1.3.1           backports_1.2.0
#> [3] circlize_0.4.11        plyr_1.8.6
#> [5] igraph_1.2.8           lazyeval_0.2.2
#> [7] splines_4.0.3          listenr_0.8.0
#> [9] scattermore_0.7         GenomeInfoDb_1.24.2
#> [11] ggplot2_3.3.5          digest_0.6.28
#> [13] htmltools_0.5.2        fansi_0.5.0
#> [15] magrittr_2.0.1          tensor_1.5
#> [17] cluster_2.1.0          ROCR_1.0-11
#> [19] ComplexHeatmap_2.4.3   globals_0.14.0
#> [21] readr_1.4.0             modelr_0.1.8
#> [23] matrixStats_0.61.0     colorspace_2.0-2
#> [25] rvest_0.3.6            blob_1.2.1
#> [27] ggrepel_0.9.1          haven_2.3.1
#> [29] xfun_0.25              dplyr_1.0.7
#> [31] RCurl_1.98-1.2          crayon_1.4.2
#> [33] jsonlite_1.7.2          lme4_1.1-25
#> [35] spatstat_1.64-1         spatstat.data_2.1-0
#> [37] survival_3.2-7          zoo_1.8-9
#> [39] glue_1.5.0              polyclip_1.10-0
#> [41] gtable_0.3.0             zlibbioc_1.34.0
#> [43] XVector_0.28.0          leiden_0.3.9
#> [45] DelayedArray_0.14.1     GetoptLong_1.0.4
#> [47] SingleCellExperiment_1.10.1 future.apply_1.8.1
#> [49] shape_1.4.5              BiocGenerics_0.34.0
#> [51] abind_1.4-5              scales_1.1.1
#> [53] pheatmap_1.0.12          DBI_1.1.0
#> [55] miniUI_0.1.1.1          Rcpp_1.0.7
#> [57] viridisLite_0.4.0         xtable_1.8-4
#> [59] clue_0.3-57              reticulate_1.22
#> [61] stats4_4.0.3              htmlwidgets_1.5.4
```

```

#> [63] httr_1.4.2
#> [65] ellipsis_0.3.2
#> [67] factoextra_1.0.7.999
#> [69] farver_2.1.0
#> [71] uwot_0.1.10
#> [73] deldir_1.0-6
#> [75] tidyselect_1.1.1
#> [77] reshape2_1.4.4
#> [79] munsell_0.5.0
#> [81] tools_4.0.3
#> [83] broom_0.7.2
#> [85] evaluate_0.14
#> [87] fastmap_1.1.0
#> [89] goftest_1.2-3
#> [91] fs_1.5.0
#> [93] purrrr_0.3.4
#> [95] pbapply_1.5-0
#> [97] nlme_3.1-149
#> [99] xml2_1.3.2
#> [101] plotly_4.10.0
#> [103] spatstat.utils_2.2-0
#> [105] tweenr_1.0.1
#> [107] statmod_1.4.35
#> [109]forcats_0.5.0
#> [111] Matrix_1.3-4
#> [113] vctrs_0.3.8
#> [115] lifecycle_1.0.1
#> [117] GlobalOptions_0.1.2
#> [119] bitops_1.0-7
#> [121] cowplot_1.1.1
#> [123] GenomicRanges_1.40.0
#> [125] patchwork_1.1.1
#> [127] promises_1.2.0.1
#> [129] gridExtra_2.3
#> [131] parallelly_1.28.1
#> [133] boot_1.3-25
#> [135] assertthat_0.2.1
#> [137] MAST_1.14.0
#> [139] SeuratObject_4.0.2
#> [141] GenomeInfoDbData_1.2.3
#> [143] mgcv_1.8-33
#> [145] hms_0.5.3
#> [147] tidyverse_1.3.0
#> [149] minqa_1.2.4
#> [151] Rtsne_0.15
#> [153] Biobase_2.48.0
#> [155] lubridate_1.7.9

RColorBrewer_1.1-2
Seurat_4.0.0
ica_1.0-2
pkgconfig_2.0.3
dbplyr_1.4.4
utf8_1.2.2
rlang_0.4.12
later_1.3.0
cellranger_1.1.0
generics_0.1.1
ggridges_0.5.3
stringr_1.4.0
yaml_2.2.1
knitr_1.30
fitdistrplus_1.1-6
RANN_2.6.1
future_1.23.0
mime_0.12
compiler_4.0.3
png_0.1-7
reprex_0.3.0
tibble_3.1.6
stringi_1.7.5
lattice_0.20-41
nloptr_1.2.2.2
pillar_1.6.4
lmtest_0.9-39
RcppAnnoy_0.0.19
data.table_1.14.2
irlba_2.3.3
httpuv_1.6.3
R6_2.5.1
KernSmooth_2.23-17
IRanges_2.22.2
codetools_0.2-16
MASS_7.3-53
SummarizedExperiment_1.18.2
rjson_0.2.20
sctransform_0.3.2
S4Vectors_0.26.1
parallel_4.0.3
rpart_4.1-15
tidyR_1.1.4
rmarkdown_2.5
ggforce_0.3.2
shiny_1.7.1

```