

# Harmony batch correction

```
library(Seurat)
library(cowplot)
library(harmony)
```

```
## Loading required package: Rcpp
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##   filter, lag
```

```
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

## load data

```
load("CM1_2.rds")
load("CM3_4.rds")
load("CM5.rds")
load("CR13_14.rds")
load("CR15_16.rds")
load("CR17_18.rds")
load("CS7_8.rds")
load("CS9_10.rds")
load("CS11_12.rds")
```

## Merge data

```
covid=merge(x=CM1_2, y=c(CM3_4, CM5, CR13_14, CR15_16, CR17_18, CS7_8, CS9_10, CS11_12))
```

```
## Warning in CheckDuplicateCellNames(object.list = objects): Some cell names are
## duplicated across objects provided. Renaming to enforce unique cell names.
```

## preprocessing for normalization, scaling, run PCA

```
covid<- covid%>%
  Seurat::NormalizeData(verbose = T) %>%
  FindVariableFeatures(selection.method = "vst", nfeatures = 5000) %>%
  ScaleData(verbose = T, features = rownames(covid)) %>%
  RunPCA(pc.genes = covid@var.genes, npcs = 20, verbose = T)
```

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

```
## PC_ 1
## Positive: IL7R, LTB, IL32, CD3D, TRBC2, CD69, GYPC, TRAC, LEF1, TRAT1
## PRDX2, EPB41, MAL, TENT5C, RCAN3, IFITM1, UBB, CAMK4, INPP4B, TRBC1
## CD7, ISG20, AC058791.1, SOCS1, HBA1, NOSIP, OXNAD1, TRABD2A, PRDM1, PIM1
## Negative: CST3, FCN1, LYZ, AIF1, MNDA, CFD, CSTA, CEBPD, S100A9, FCER1G
## S100A8, CD14, CTSS, LST1, VCAN, TYROBP, GRN, IGSF6, SERPINA1, S100A12
## PSAP, CD68, SPI1, IFITM3, PLBD1, AC020656.1, MS4A6A, S100A11, TIMP2, FCGRT
## PC_ 2
## Positive: SLC4A1, AHSP, HBD, ALAS2, CA1, HBA2, HBA1, HBM, SELENBP1, HBB
## SNCA, SLC25A37, TRIM58, GYPA, KLF1, GMPR, GYPB, BLVRB, FECH, HMBS
## TMCC2, SPTA1, CA2, RHAG, EPB42, ANKRD9, FHDC1, HEMGN, DCAF12, NFIX
## Negative: ACTB, IL32, IFITM1, ACTG1, JUN, IFITM2, CD3D, CTSW, HCST, IER2
## NKG7, TRBC2, JUNB, CCL5, PPIB, CST7, CD69, IRF1, TRAC, GAPDH
## ARPC3, LTB, LY6E, GNLY, GZMM, GZMA, ARPC1B, HSP90B1, IL7R, ABHD17A
## PC_ 3
## Positive: MS4A1, CD79A, IGHM, HLA-DRA, BANK1, RALGPS2, CD74, MEF2C, LINC00926, CD79B
## HLA-DQA1, AFF3, IGHD, BCL11A, TNFRSF13C, IGKC, HLA-DRB1, GNG7, SPIB, ADAM28
## EAF2, TCL1A, HLA-DQB1, VPREB3, HLA-DMA, HLA-DPA1, FCER2, FCRL1, CD40, P2RX5
## Negative: NKG7, CTSW, CST7, CCL5, GNLY, GZMA, IL32, GZMH, KLRD1, GZMB
## PRF1, HCST, SH3BGRL3, FGFBP2, GZMM, CCL4, S100A4, HOPX, RARRES3, FCGR3A
## C12orf75, EFHD2, IFITM2, ANXA1, IFITM1, CD247, SPON2, CTSC, ABHD17A, GAPDH
## PC_ 4
## Positive: NKG7, HLA-DPA1, CST7, GNLY, HLA-DPB1, CTSW, GZMB, GZMA, KLRD1, PRF1
## CCL5, HLA-DRB1, GZMH, CD74, FCGR3A, FGFBP2, HOPX, PLEK, CCL4, SPON2
## MS4A1, HLA-DQB1, CD79A, HLA-DQA1, BANK1, KLRF1, HLA-DRA, IGHM, CLIC3, EFHD2
## Negative: IL7R, LEF1, TRAT1, MAL, LTB, RCAN3, NOSIP, CAMK4, INPP4B, TRABD2A
## OXNAD1, TNFRSF25, TSHZ2, AQP3, SOCS3, RGS10, ANK3, LPAR6, AL445686.2, TRAC
## CD28, DPP4, CORO1B, ADTRP, CD3D, NFKBIZ, ITGA6, FAM13A, BEX3, VIM
## PC_ 5
## Positive: FTH1, ACTB, LTB, TRAC, ARPC3, LY6E, COTL1, IFITM1, GAPDH, VIM
## ISG20, IL7R, ARPC1B, CD3D, IL32, SH3BGRL3, MX1, PSME2, PRDX2, S100A10
## ATP5MF, TRAT1, MYL6, NOSIP, ALOX5AP, AQP3, IRF7, TXN, H2AFZ, IFIT3
## Negative: ZEB2, MT-CO3, NEAT1, MT-CO2, SYNE1, MYO1F, GNLY, MTRNR2L12, MT-ND3, NKG7
## KLRD1, ARHGAP26, CCL5, LYST, EFHD2, SPON2, AOA1, ITGAX, AHNK, VCAN
## DMXL2, PRF1, GZMB, ADGRG1, GZMH, FCGR3A, HIST1H1E, SLC8A1, CLIC3, TNFAIP2
```

run harmony for batch correction

```
covid<- covid%>%
  RunHarmony("patient", plot_convergence = TRUE)
```

```
## Warning: Quick-TRANSfer stage steps exceeded maximum (= 3564350)
```

```
## Harmony 1/10
```

```
## Harmony 2/10
```

```
## Harmony 3/10
```

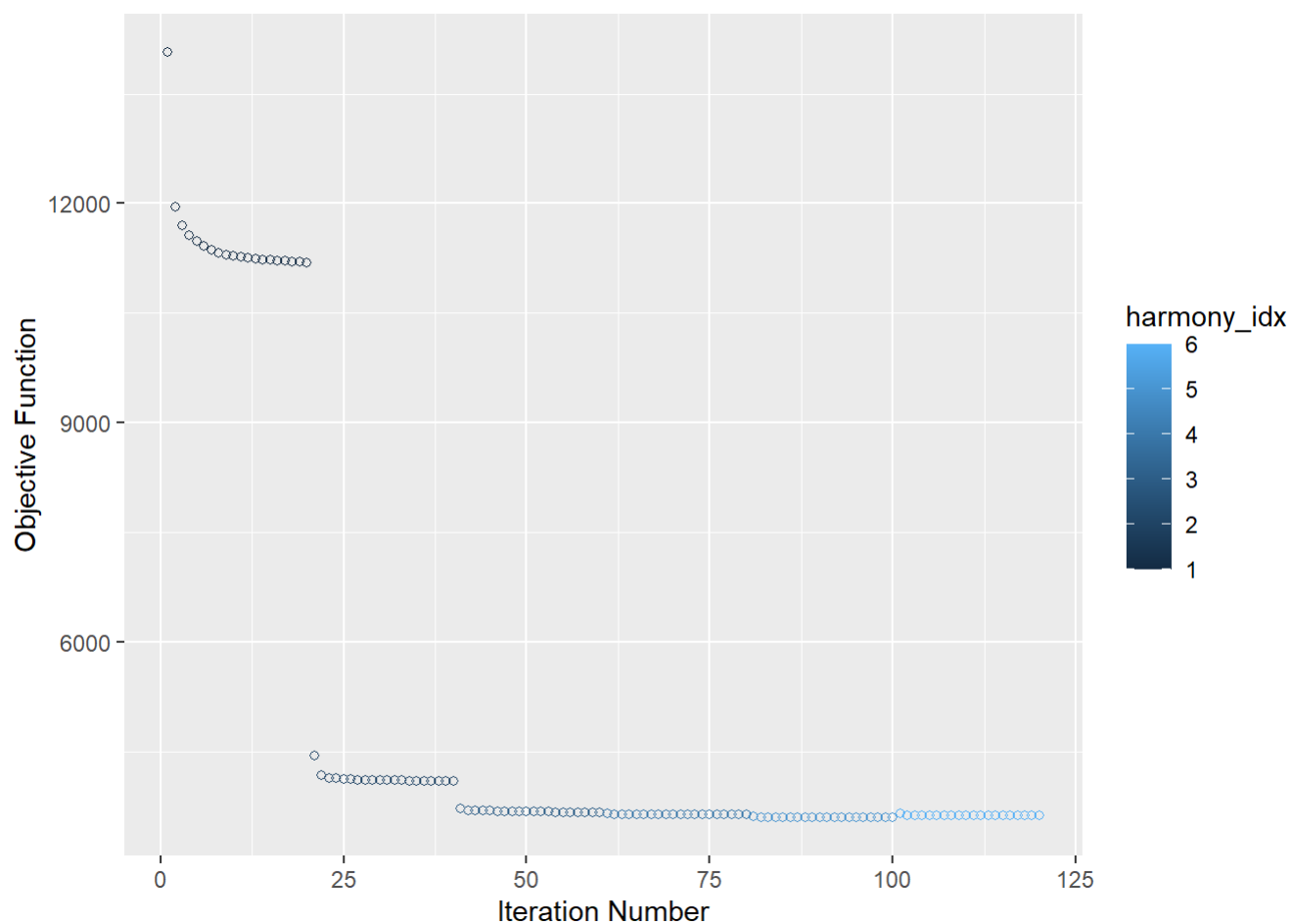
```
## Harmony 4/10
```

```
## Harmony 5/10
```

```
## Harmony 6/10
```

```
## Harmony converged after 6 iterations
```

```
## Warning: Invalid name supplied, making object name syntactically valid. New  
## object name is Seurat..ProjectDim.RNA.harmony; see ?make.names for more details  
## on syntax validity
```



run UMAP dimension reduction and cluster the cells

```
covid<- covid%>%
  RunUMAP(reduction = "harmony", dims = 1:20) %>%
  FindNeighbors(reduction = "harmony", dims = 1:20) %>%
  FindClusters(resolution = 0.4) %>%
  identity()
```

```
## Warning: The default method for RunUMAP has changed from calling Python UMAP via reticulate to the R-native UWOT using the cosine metric
## To use Python UMAP via reticulate, set umap.method to 'umap-learn' and metric to 'correlation'
## This message will be shown once per session
```

```
## 16:11:10 UMAP embedding parameters a = 0.9922 b = 1.112
```

```
## 16:11:10 Read 71287 rows and found 20 numeric columns
```

```
## 16:11:10 Using Annoy for neighbor search, n_neighbors = 30
```

```
## 16:11:10 Building Annoy index with metric = cosine, n_trees = 50
```

```
## 0%   10   20   30   40   50   60   70   80   90  100%
```

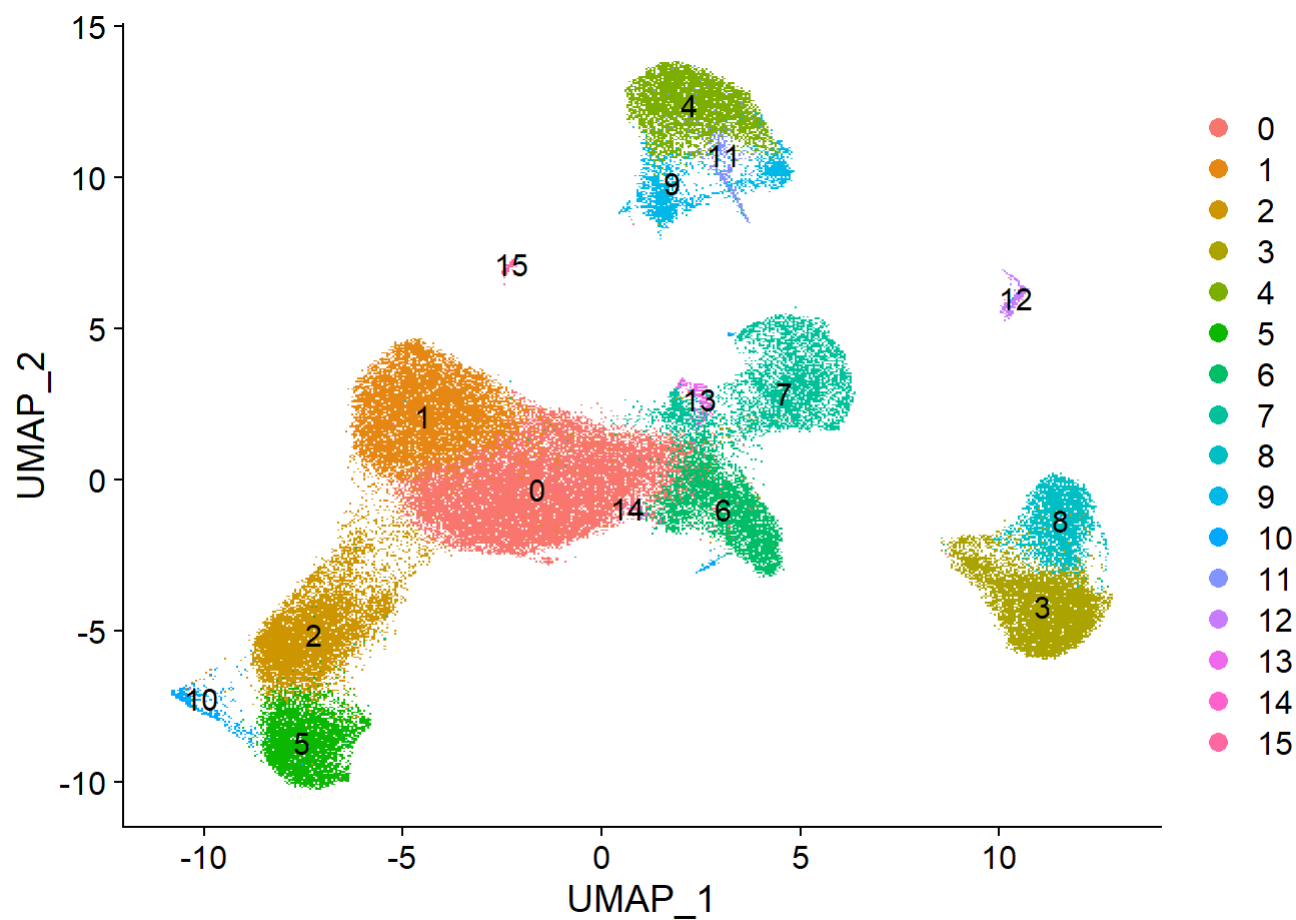
```
## [----|----|----|----|----|----|----|----|----|----|
```

```
## *****|
## 16:11:19 Writing NN index file to temp file C:\Users\STRIPP~1\AppData\Local\Temp\RtmpYZ1de4\file6de0c7a33d3
## 16:11:19 Searching Annoy index using 1 thread, search_k = 3000
## 16:11:44 Annoy recall = 100%
## 16:11:44 Commencing smooth kNN distance calibration using 1 thread
## 16:11:47 Initializing from normalized Laplacian + noise
## 16:11:52 Commencing optimization for 200 epochs, with 3213164 positive edges
## 16:13:05 Optimization finished
## Computing nearest neighbor graph
##Computing SNN
```

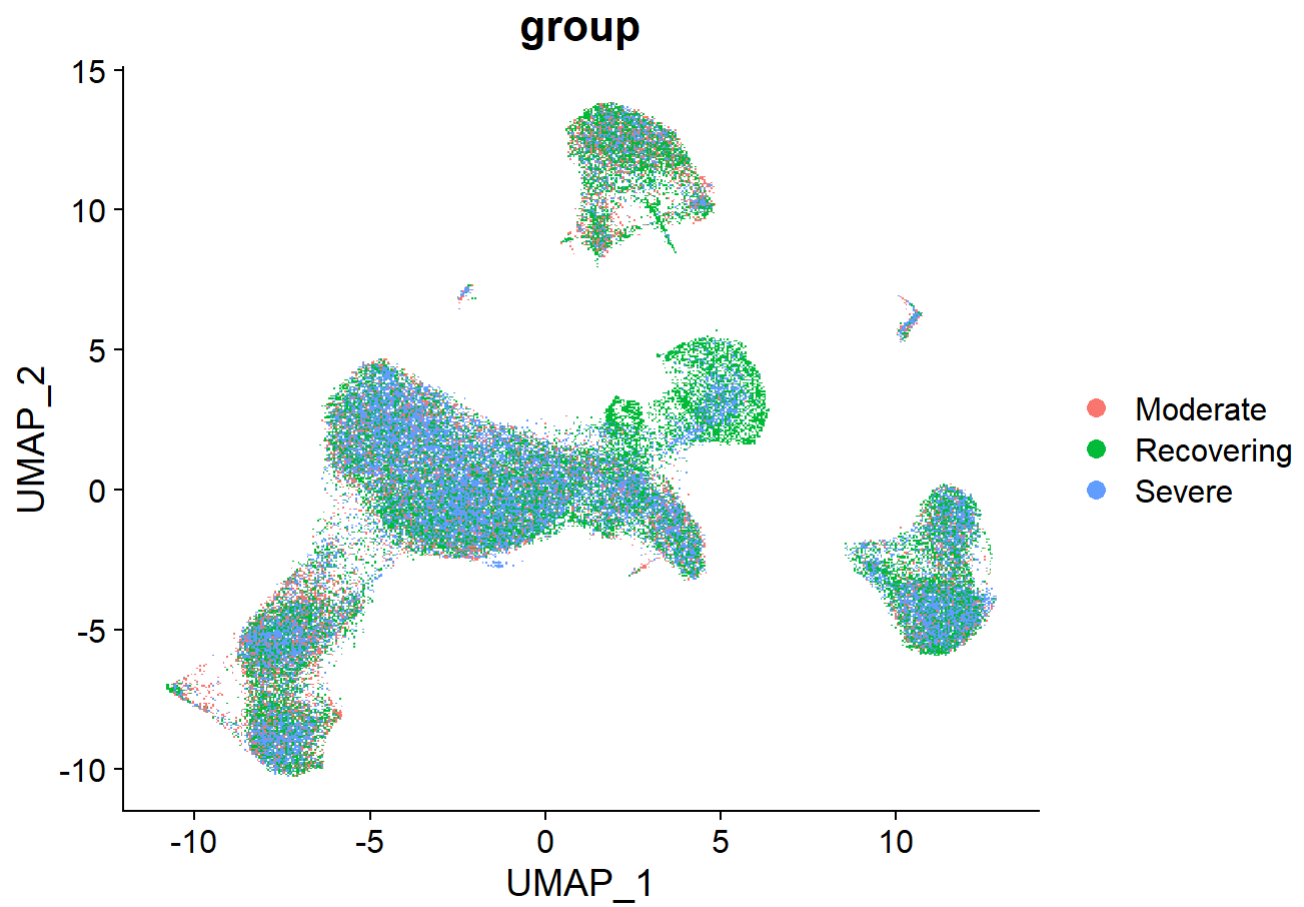
```
## Modularity Optimizer version 1.3.0 by Ludo Waltman and Nees Jan van Eck
##
## Number of nodes: 71287
## Number of edges: 2163450
##
## Running Louvain algorithm...
## Maximum modularity in 10 random starts: 0.9208
## Number of communities: 16
## Elapsed time: 38 seconds
```

```
DimPlot(covid, reduction="umap", label=T)
```

```
## Warning: Using `as.character()` on a quosure is deprecated as of rlang 0.3.0.  
## Please use `as_label()` or `as_name()` instead.  
## This warning is displayed once per session.
```



```
DimPlot(covid, reduction="umap", group.by="group")
```



save data

```
save(covid, file="covidharmony.rds")
```