P259: Combined data cleaning

Dendritic cells (pDC)

Kim Dill-McFarland, kadm@uw.edu

version September 09, 2020

Contents

Background **1**

Setup **1**

Read in and format data **3**
- Metadata . . . 3
- Counts table . . . 3
- Sample summary . . . 3

Data cleaning **3**
- Median CV coverage vs. mapped duplicate reads . . . 3
- Total aligned counts . . . 4
- Filter by quality and coverage . . . 4
- Filter PCA outliers . . . 7
- Normalize for RNA composition . . . 10
- Filter rare genes . . . 10
- PCA . . . 10
- Summary . . . 11

Save clean data **13**

R session **13**

Background

P259.1: pDC were isolated from 4 healthy donors' whole blood or buffy coat. Cells were adapted to media with IL3 and then cultured with eosinophil (EOS) supernatant and/or human rhinovirus (RV).

P259.2: pDC were isolated from 7 asthmatic donors' whole blood, 4 of which were on Anti-IL5 therapy. Cells were adapted to media with IL3 and then cultured with or without RV.

The purpose of this workflow is to complete quality control and data cleaning of metadata and RNA-seq libraries generated from the above experiments. This includes 1) removing low coverage libraries, 2) filtering rare genes, 3) removing outlying libraries, and 4) normalizing for RNA composition.

Setup

Load packages

```
# Data manipulation and figures
library(tidyverse)
  #Multi-panel figures
  library(cowplot)
  # Modify ggplot figures to non-overlapping text labels
  library(ggrepel)
  # Modify ggplot data order within facets
  library(drlib)
  # Plot log scales
  library(scales)
# Reading Excel files
library(readxl)
# Working with dates
library(lubridate)

# Empirical analysis of digital gene expression data
## Data normalization
library(edgeR)

# Print pretty table to knit file
library(knitr)
library(kableExtra)
  options(knitr.kable.NA = '')

#Define colors for plots
group.cols <- c("none:none"="#dadaeb",
                "none:AntiIL5"="#9e9ac8",
                "none:EOS.supp"="#54278f",
                "HRV:none"="#c7e9c0",
                "HRV:AntiIL5"="#74c476",
                "HRV:EOS.supp"="#006d2c",
                "flu:none"="#fdae6b",
                "flu:AntiIL5"="#e6550d")
samp.cols <- c("AC1"="#969696",
                "AC2"="#a6cee3",
                "AC3"="#1f78b4",
                "AC4"="#b2df8a",
                "AC5"="#33a02c",
                "AT1"="#fb9a99",
                "AT2"="#e31a1c",
                "AT3"="#fdbf6f",
                "AT4"="#ff7f00",
                "donor1"="#cab2d6",
                "donor2"="#6a3d9a",
                "donor3"="#ffff99",
                "donor4"="#b15928")
```

Set seed

```
set.seed(4389)
```

Read in and format data

Metadata

Counts table

Read in counts table from P259.1 and P259.2 (`data_raw/*combined_counts.csv`) and then combine to one data frame.

Read in and filter associated gene key (`data_raw/2020.06.18_HGNC.gene.key.txt`, downloaded from https://www.genenames.org/download/custom/) to protein coding (pc) genes that occur in the count data set and have valid hgnc symbols.

Then filter the count data to pc genes and save.

```
#Save counts table
write_csv(count.pDC, "data_clean/P259_pDC_counts.csv")
```

Verify data order To ensure that all data match, check that all libraries in the count data are present and in the same order in the metadata. And vice versa.

```
## [1] TRUE
```

Sample summary

At this point, the following objects exist in our environment. Protein coding = pc

```
## [1] "count.pDC"  "group.cols" "key.pc"     "meta"       "samp.cols"
```

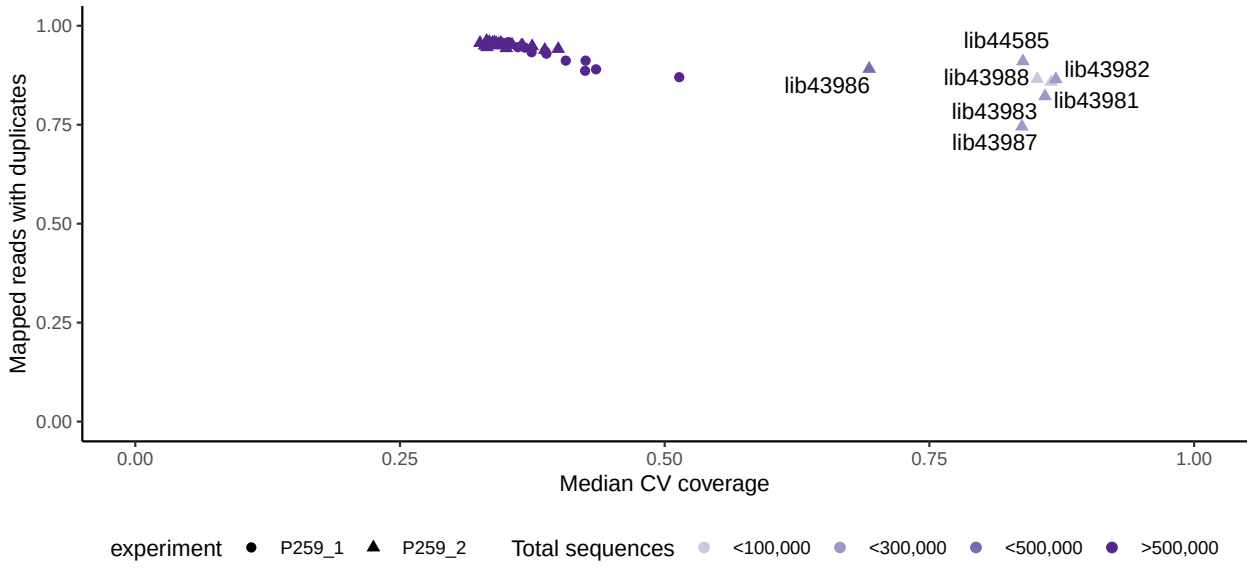
And the sample numbers are

experiment	asthma	IL5	virus	n
P259_1	healthy	none	none	4
		none	HRV	4
		EOS.supp	none	3
		EOS.supp	HRV	4
P259_2	asthma	none	none	5
		none	HRV	10
		AntiIL5	none	4
		AntiIL5	HRV	8

Data cleaning

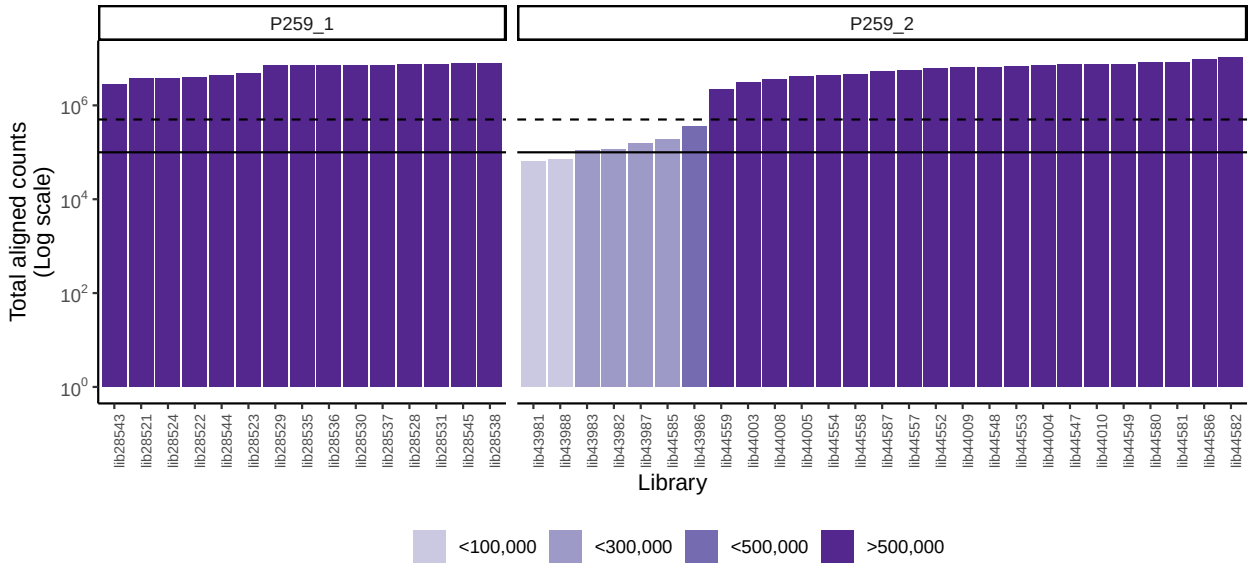
Median CV coverage vs. mapped duplicate reads

Compare the median coefficient of variation (CV) coverage (`median_cv_coverage`) and percent alignment of reads with duplicates (`mapped_reads_w_dups`).



Total aligned counts

Plot total counts per library. Here, we compare our standard cutoff of 500,000 to a less strict cutoff of 100,000, because 7 pDC libraries would be removed at the standard cutoff.



Filter by quality and coverage

Thus, the following libraries with questionable metrics (highlighted in red in HTML format) may be removed.

experiment

libID

donorID

asthma

IL5

virus

median_cv_coverage

mapped_reads_w_dups

total_sequences

P259_2

lib43981

AC1

asthma

none

none

0.86501

0.858165459

62823

P259_2

lib43982

AC1

asthma

none

HRV

0.869398

0.864978903

113480

P259_2

lib43983

AC1

asthma

none

HRV

0.85921

0.821943431

112154

P259_2

lib43986

AC2

asthma

none

none

0.693137

0.89115488

347291

P259_2

lib43987

AC2

asthma

none

HRV

0.837354

0.745407193

157832

P259_2

lib43988

AC2

asthma

none

HRV

0.851868

0.865923487

69503

P259_2

lib44585

AT4

asthma

AntiIL5

none

0.838316

0.910325198

188722

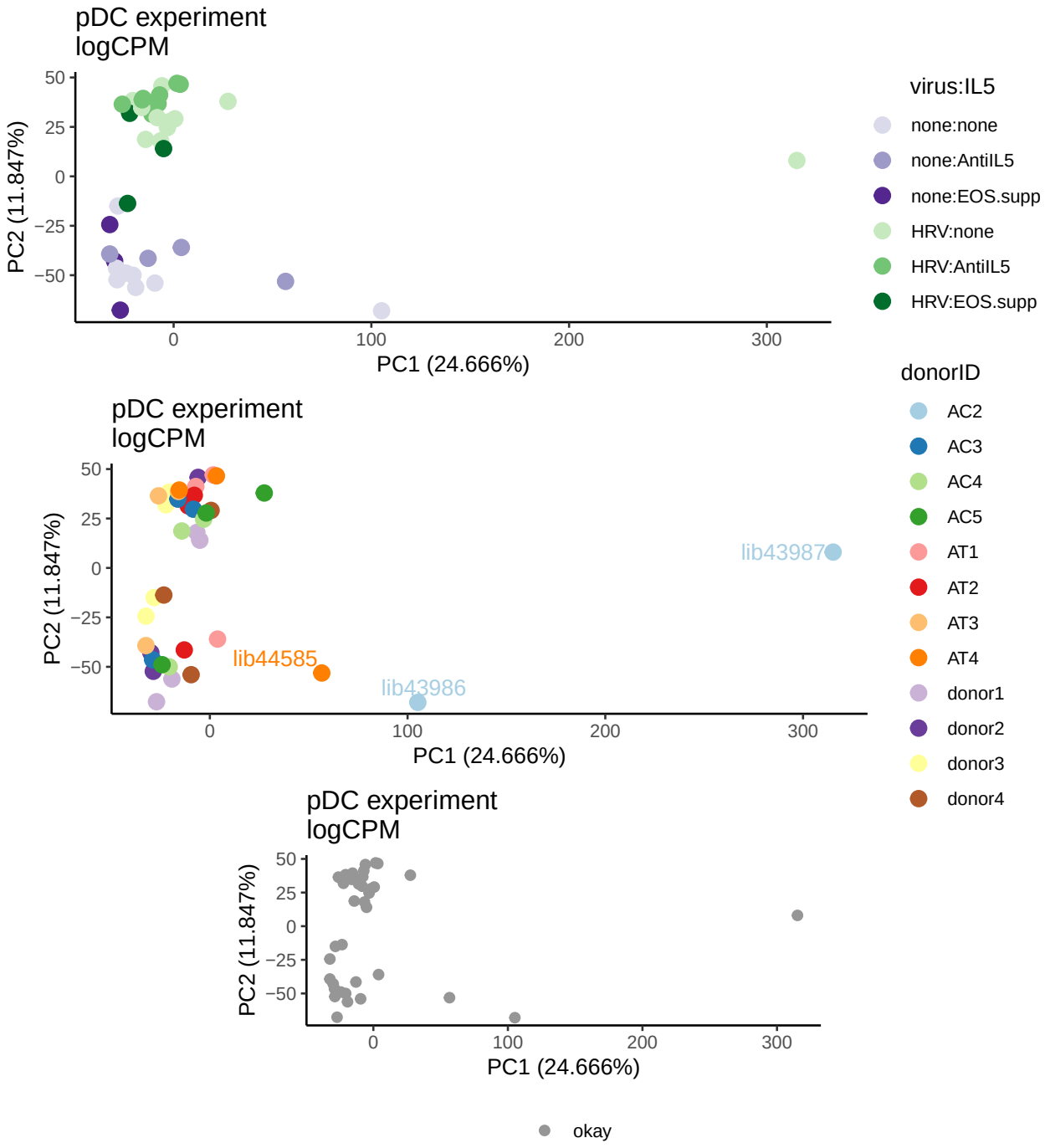
First, 3 libraries with < 100,000 sequences are removed. Since this removes the media (none:none) sample for donor AC1, all other AC1 samples are also removed. Of note, if the other AC1 samples are retained, they appears as PCA outliers in the next step.

```
#Metadata
meta.filter <- meta %>%
  # CV coverage and alignment
  filter(total_sequences >= 1E5) %>%
  filter(donorID != "AC1") %>%
```

```
  droplevels()

#Count data
count.pDC.filter <- count.pDC %>%
  # Keep libraries (columns) remaining filtered metadata table (rows)
  dplyr::select(geneName, as.character(meta.filter$libID))
```

Filter PCA outliers

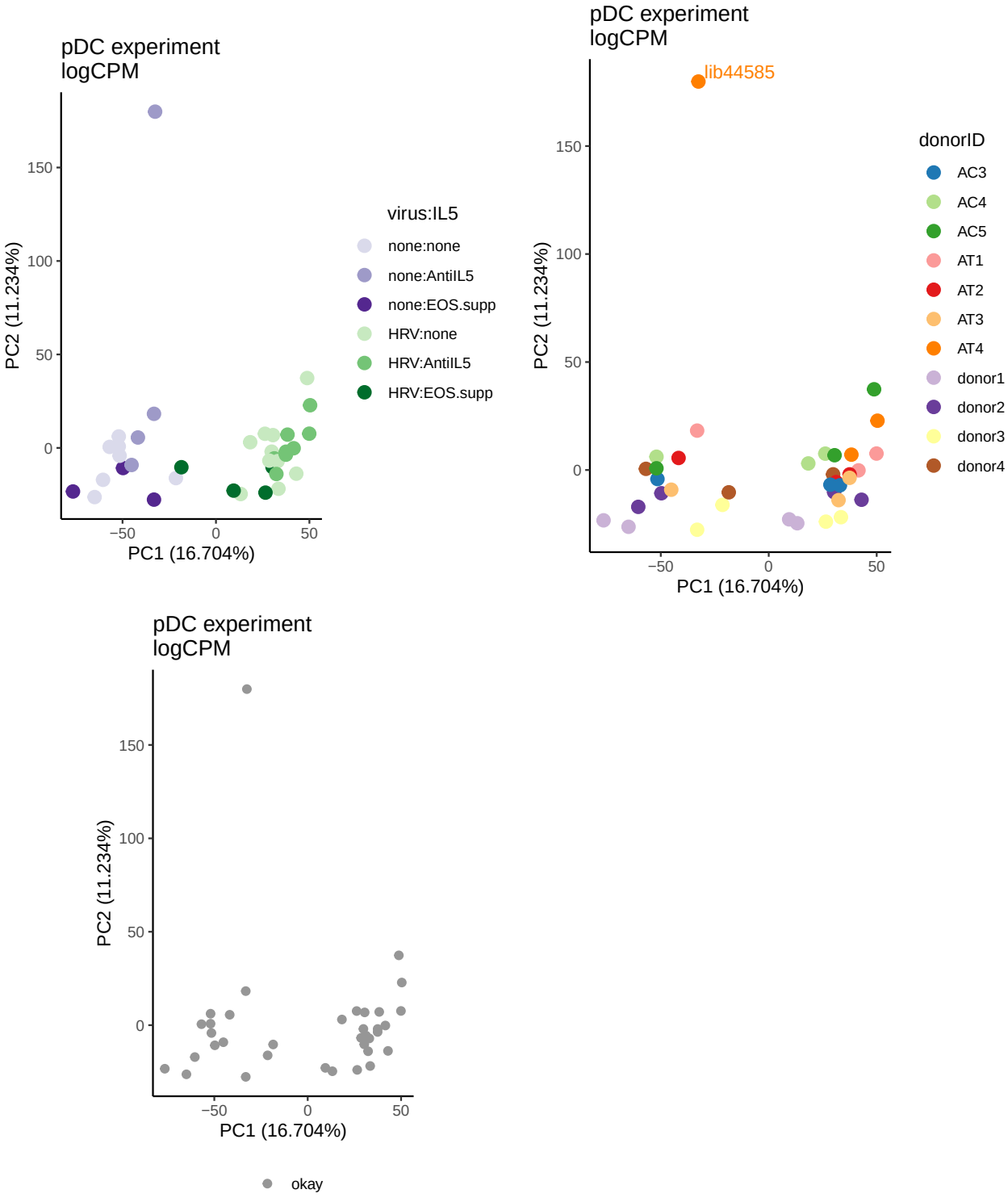
Next, samples are assessed in PCA to determine if remaining questionable quality samples (labeled with libID) appear as PCA outliers.



Donor AC2 appears to be an outlier for all of its samples and will be removed.

```
#Metadata
meta.filter2 <- meta.filter %>%
  filter(donorID != "AC2") %>%
  droplevels()

#Count data
count.pDC.filter2 <- count.pDC.filter %>%
  # Keep libraries (columns) remaining filtered metadata table (rows)
  dplyr::select(geneName, as.character(meta.filter2$libID))
```

Re-assess PCA.



Additionally, lib44585 was flagged for poor-quality above and is labeled in the PCA. While this sample is somewhat of an outlier in PCA, it is not > 3 s.d. away from its group's mean. Thus, it will remain and be re-assessed after gene filtering.

Therefore, the filtered data have

- max median CV coverage = 0.84

- min mapped reads with duplicates = 0.87
- min total sequences = 188,722

Normalize for RNA composition

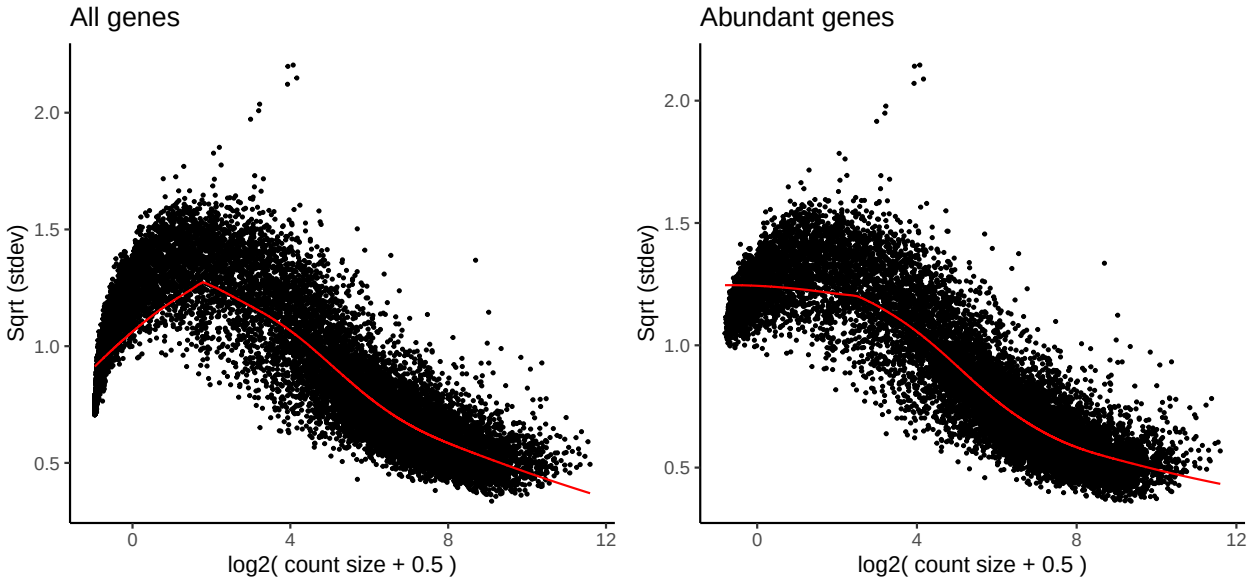
Create DGEList object

```
dat.pDC.filter2 <- DGEList(
  #count table. move gene names to column names
  counts=as.matrix(column_to_rownames(count.pDC.filter2, "geneName")),
  #metadata
  samples=meta.filter2,
  #keep genes in count table
  genes=filter(key.pc, geneName %in% count.pDC.filter2$geneName))
```

Filter rare genes

The raw gene sets contain highly variable, low abundance/rare genes. Genes must be at least 1 CPM in at least 3 samples to remain.

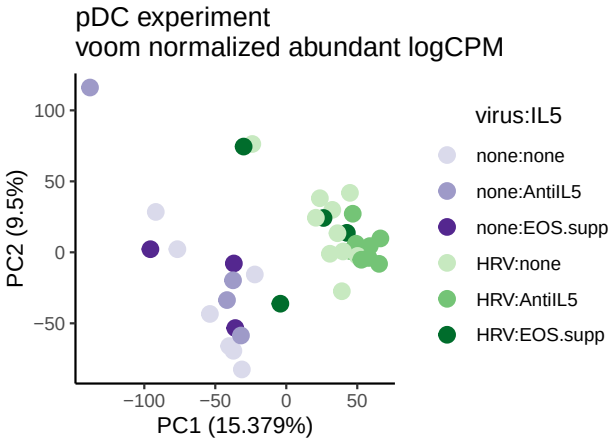
```
source("https://raw.githubusercontent.com/kdillmcfarland/R_bioinformatic_scripts/master/RNAseq_rare_gene

rare.gene.filter(dat.pDC.filter2,
                 min.CPM=1,
                 min.sample=3,
                 name="dat.pDC.filter2.abund")
```

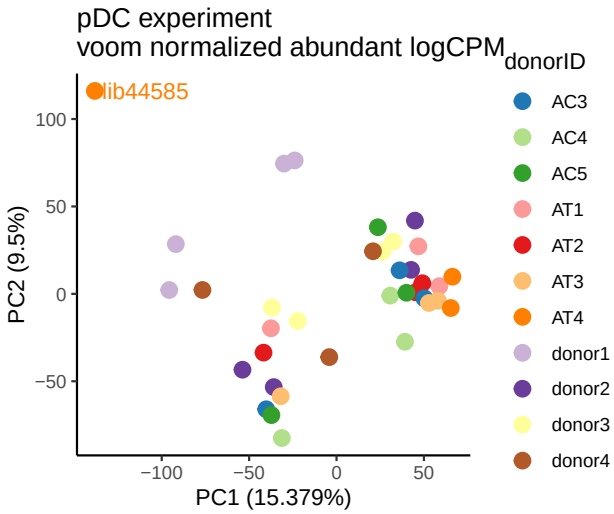


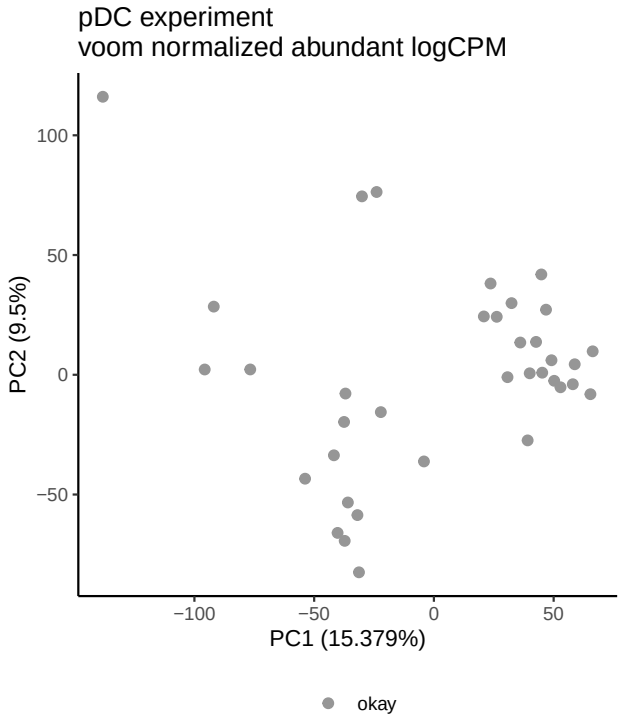
This removes 5324 (~ 28%) genes. This sufficiently filters these data as indicated by a decreasing variance trend for lowly abundant genes.

PCA

We see that previously questionable samples (labeled with libID) are sufficiently normalized to not appear as PCA outliers.



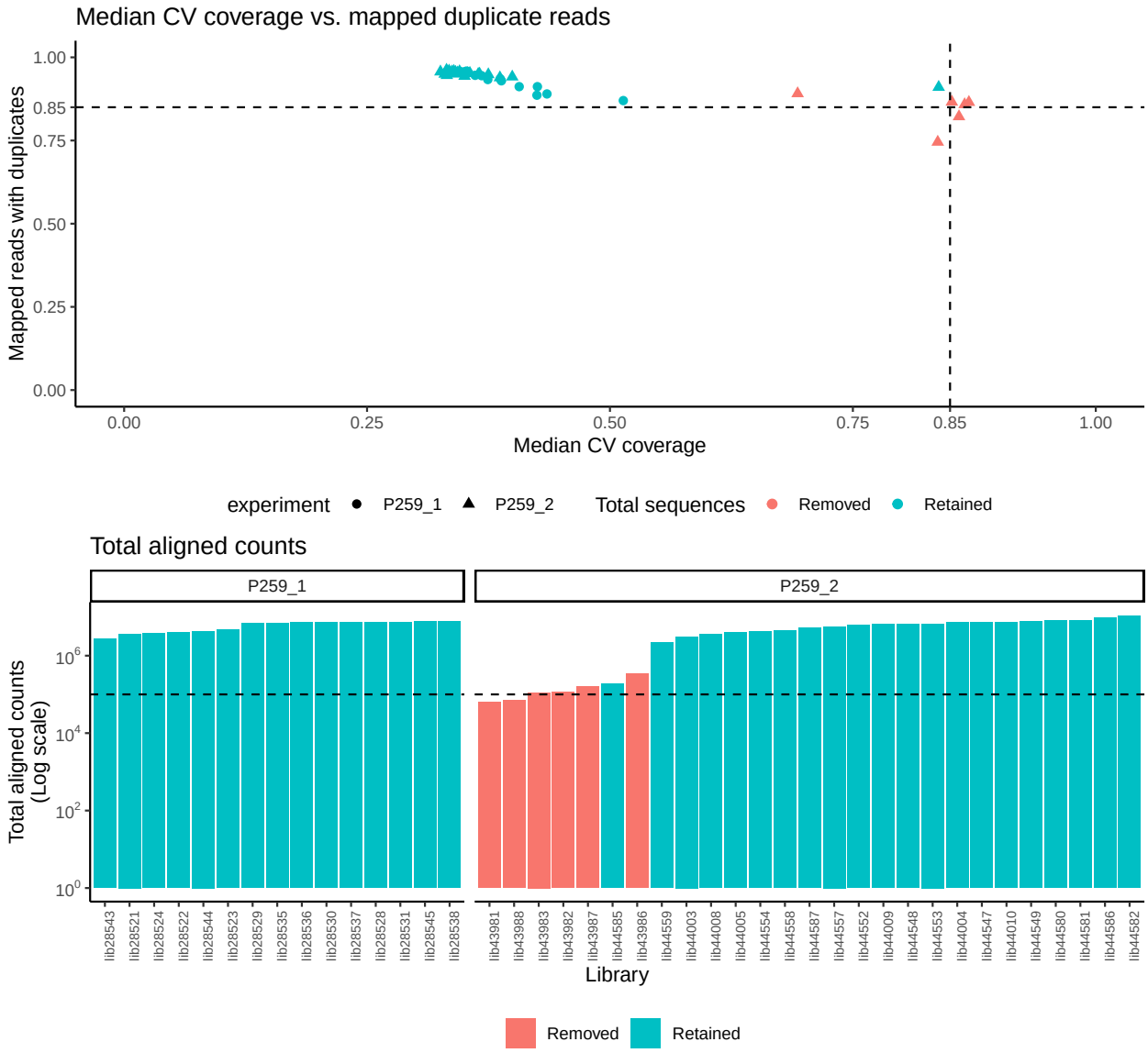




Summary

The following final quality cutoffs were applied to the data:

- Library median CV coverage ≤ 0.84
- Library mapped reads with duplicates ≥ 0.87
- Library total counts $\geq 1.88722 \times 10^5$
- Libraries from donor AC1 due to loss of media sample in previous filtering
- Within 3 sd of group mean on PCA (*e.g.* not outliers)
- Genes at least 1 CPM in at least 3 samples



experiment	libID	donorID	asthma	IL5	virus	filters
P259_2	lib43981	AC1	asthma	none	none	CV coverage, total seqs
P259_2	lib43983	AC1	asthma	none	HRV	mapped dup reads, CV coverage, no media-only sample
P259_2	lib43982	AC1	asthma	none	HRV	CV coverage, no media-only sample
P259_2	lib43986	AC2	asthma	none	none	PCA outlier
P259_2	lib43987	AC2	asthma	none	HRV	mapped dup reads, PCA outlier
P259_2	lib43988	AC2	asthma	none	HRV	CV coverage, total seqs, PCA outlier

This results in the following samples for statistical analysis.

experiment	asthma	IL5	virus	n
P259_1	healthy	none	none	4
		none	HRV	4
		EOS.supp	none	3
		EOS.supp	HRV	4
P259_2	asthma	none	none	3
		none	HRV	6
		AntiIL5	none	4
		AntiIL5	HRV	8

Save clean data

Normalized dat objects.

```
#Save R object
#Rename all pDC object
dat.pDC.voom <- dat.pDC.filter2.abund.norm.voom

#P259_1
dat.pDC.voom_1 <- dat.pDC.voom

dat.pDC.voom_1$targets <- dat.pDC.voom_1$targets %>%
  rownames_to_column() %>%
  filter(experiment == "P259_1") %>%
  droplevels() %>%
  column_to_rownames()

dat.pDC.voom_1$E <- as.data.frame(dat.pDC.voom_1$E) %>%
  rownames_to_column() %>%
  select(rowname, dat.pDC.voom_1$targets$libID) %>%
  column_to_rownames()

#P259_2
dat.pDC.voom_2 <- dat.pDC.voom

dat.pDC.voom_2$targets <- dat.pDC.voom_2$targets %>%
  rownames_to_column() %>%
  filter(experiment == "P259_2") %>%
  droplevels() %>%
  column_to_rownames()

dat.pDC.voom_2$E <- as.data.frame(dat.pDC.voom_2$E) %>%
  rownames_to_column() %>%
  select(rowname, dat.pDC.voom_2$targets$libID) %>%
  column_to_rownames()

save(dat.pDC.voom, dat.pDC.voom_1, dat.pDC.voom_2,
     file="data_clean/P259_pDC_clean.RData")
```

R session

```
sessionInfo()
```

```
## R version 4.0.0 (2020-04-24)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Catalina 10.15.5
##
## 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] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] kableExtra_1.1.0 knitr_1.29       edgeR_3.30.3     limma_3.44.3
##  [5] lubridate_1.7.9  readxl_1.3.1     scales_1.1.1     drlib_0.1.1
##  [9] ggrepel_0.8.2    cowplot_1.0.0    forcats_0.5.0    stringr_1.4.0
## [13] dplyr_1.0.0      purrr_0.3.4      readr_1.3.1      tidyr_1.1.0
## [17] tibble_3.0.3     ggplot2_3.3.2    tidyverse_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.5        locfit_1.5-9.4    lattice_0.20-41   assertthat_0.2.1
##  [5] digest_0.6.25     R6_2.4.1          cellranger_1.1.0  backports_1.1.8
##  [9] reprex_0.3.0      evaluate_0.14     httr_1.4.2        pillar_1.4.6
## [13] rlang_0.4.7       rstudioapi_0.11   blob_1.2.1        Matrix_1.2-18
## [17] rmarkdown_2.3     labeling_0.3      webshot_0.5.2     munsell_0.5.0
## [21] broom_0.7.0       compiler_4.0.0    modelr_0.1.8      xfun_0.16
## [25] pkgconfig_2.0.3   htmltools_0.5.0   tidyselect_1.1.0  fansi_0.4.1
## [29] viridisLite_0.3.0 crayon_1.3.4      dbplyr_1.4.4      withr_2.2.0
## [33] grid_4.0.0        jsonlite_1.7.0    gtable_0.3.0      lifecycle_0.2.0
## [37] DBI_1.1.0         magrittr_1.5      cli_2.0.2         stringi_1.4.6
## [41] farver_2.0.3      fs_1.4.2          xml2_1.3.2        ellipsis_0.3.1
## [45] generics_0.0.2    vctrs_0.3.2       tools_4.0.0       glue_1.4.1
## [49] hms_0.5.3         yaml_2.2.1        colorspace_1.4-1  rvest_0.3.6
## [53] haven_2.3.1
```