ykim_rnaseq_pca

Kate Weaver

4/12/2022

Exploratory Data Analysis of MSH-5 RNA-seq experiment

```
library(SummarizedExperiment)
library(ggplot2)
library(rtracklayer)
library(edgeR)
library(tidyverse)
library(GGally)
library(ggpubr)
library(DESeq2)
library(ggrepel)
```

```
load("~/mccoyLab/collabs/ykim_rnaseq_20220329/results/ykim_rnaseq_20220329_summarized_experiment.Rdata")
counts <- as.matrix(assays(seAll)$counts)
for (col in 1:ncol(counts)){
  colnames(counts)[col] <- strsplit(sub("Aligned.sortedByCoord.out.bam", "", colnames(counts)[col]), "_")[[1]][1]
}
colnames(counts)
```

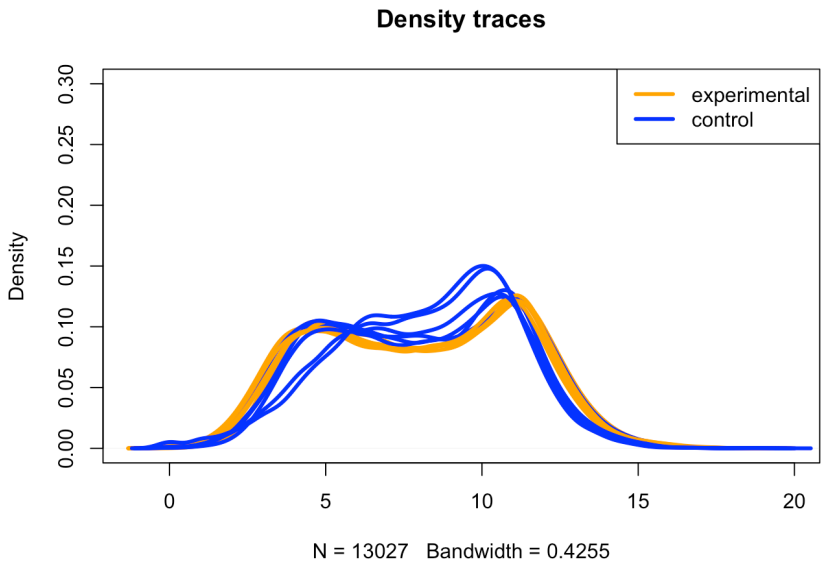
```
##  [1] "MSH5-V5-13A-1"    "MSH5-V5-13A-2"    "MSH5-V5-13A-3"    "MSH5-V5-1"
##  [5] "MSH5-V5-2"        "MSH5-V5-3"        "MSH5-V5-Trunc4-1" "MSH5-V5-Trunc4-2"
##  [9] "MSH5-V5-Trunc4-3" "N2-1"             "N2-2"             "N2-3"
```

```
metadf <- data.frame(samples = c(paste0("MSH5-V5-13A-", 1:3), paste0("MSH5-V5-", 1:3), paste0("MSH5-V5-Trunc4-", 1:3), paste0("N2-", 1:3)),
                     genotype = c(rep("MSH5-V5-13A", 3), rep("MSH5-V5",3), rep("MSH5-V5-Trunc4", 3), rep("N2", 3)),
                     full_genotype_colors = c(rep("green", 3), rep("blue", 3), rep("orange", 3), "purple", "brown", "pink"),
                     genotype_colors = c(rep("green", 3), rep("blue", 3), rep("orange", 3), rep("purple", 3)),
                     replicate = rep(1:3, 4),
                     simplified_genotype1 = c(rep("experimental-13A", 3), rep("control", 3), rep("experimental-Trunc4", 3), rep("control",3)),
                     simplified_genotype2 = c(rep("experimental", 3), rep("control", 3), rep("experimental",3), rep("control",3)),
                     simplified_genotype3 = c(rep("experimental", 3), rep("control-V5",3), rep("experimental",3), rep("control-N2",3)),
                     simplified_genotype2_colors = c(rep("orange", 3), rep("blue", 3), rep("orange",3), rep("blue",3)))
```

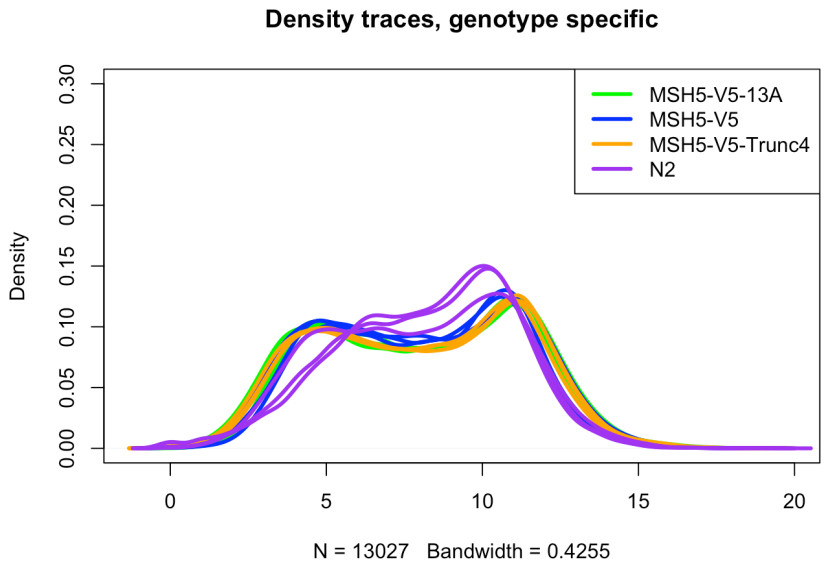
```
gtf <- rtracklayer::import("~/genomes/WBcel235/Caenorhabditis_elegans.WBcel235.97.gtf") %>%
  as.data.frame() %>%
  dplyr::filter(type=="gene") %>%
  dplyr::select(gene_id, gene_name, seqnames, width) %>%
  dplyr::rename(ensembl_gene_id = gene_id) %>%
  dplyr::rename(chromosome_name = seqnames) %>%
  dplyr::rename(length = width)

gene_table <- gtf[match(rownames(counts), gtf$ensembl_gene_id),]
gene_table <- gene_table[gene_table$chromosome_name %in% c("I", "II","III", "IV", "V", "X"),]
counts <- counts[gene_table$ensembl_gene_id,]
```

Density traces of sample expression

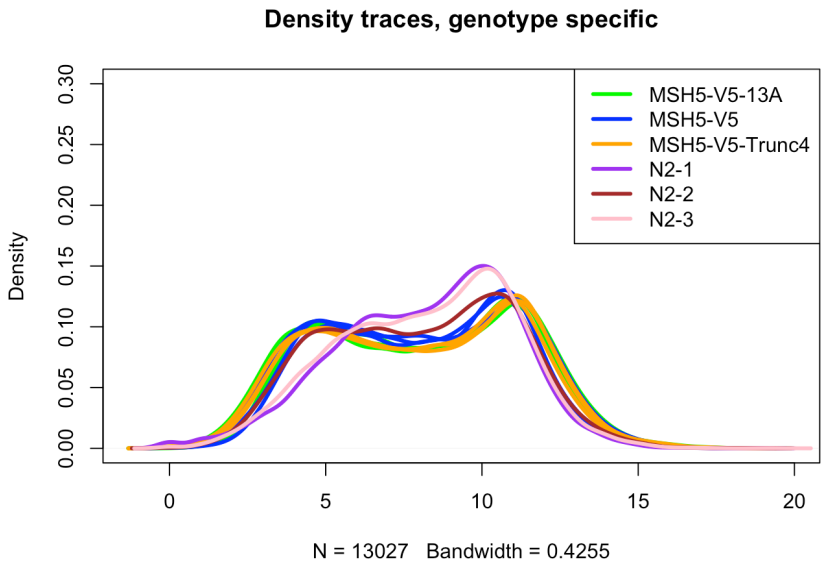
```
logdata <- log2(counts +1)
edata = logdata[rowMeans(logdata) > 3, ]
plot(density(edata[,1]), col=metadf$simplified_genotype2_colors[1], lwd=3, ylim=c(0, 0.3), main="Density traces")
for (i in 2:ncol(edata)){lines(density(edata[,i]), lwd=3, col=metadf$simplified_genotype2_colors[i])}
legend("topright", c("experimental", "control") , col=c("orange", "blue"), lwd=3)
```




```
plot(density(edata[,1]), col=metadf$genotype_colors[1], lwd=3, ylim=c(0,0.3), main="D
ensity traces, genotype specific")
for (i in 2:ncol(edata)){lines(density(edata[,i]), lwd=3, col=metadf$genotype_colors
[i])}
legend("topright", c("MSH5-V5-13A", "MSH5-V5", "MSH5-V5-Trunc4", "N2") , col=c("gree
n", "blue", "orange", "purple"), lwd=3)
```




```
plot(density(edata[,1]), col=metadf$full_genotype_colors[1], lwd=3, ylim=c(0,0.3), ma
in="Density traces, genotype specific")
for (i in 2:ncol(edata)){lines(density(edata[,i]), lwd=3, col=metadf$full_genotype_co
lors[i])}
legend("topright", c("MSH5-V5-13A", "MSH5-V5", "MSH5-V5-Trunc4", "N2-1", "N2-2", "N2-
3") , col=c("green", "blue", "orange", "purple", "brown", "pink"), lwd=3)
```



compute TMM and TPM from counts

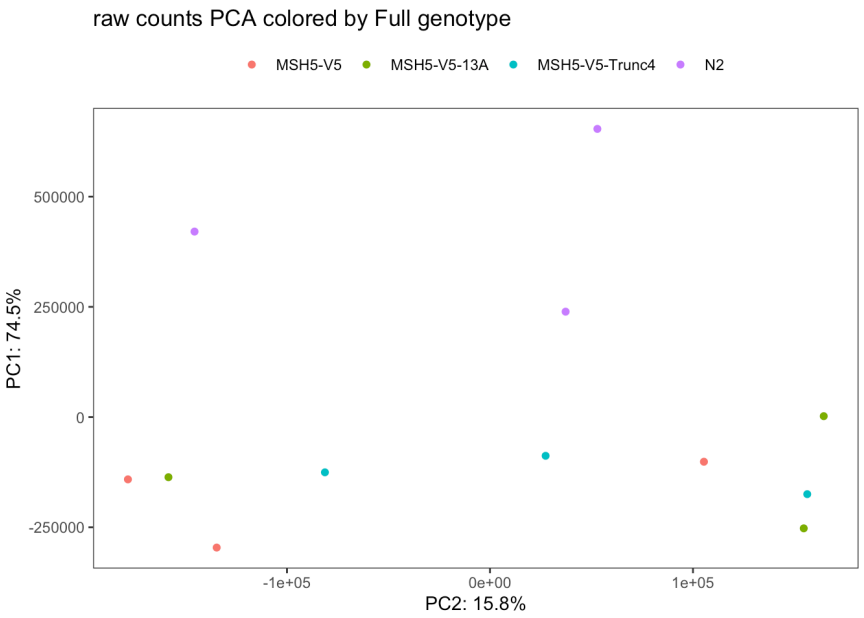
```
#TPM Calculation
calc_tpm <- function(x, gene.length) {
  x <- as.matrix(x)
  len.norm.lib.size <- colSums(x / gene.length)
  return((t(t(x) / len.norm.lib.size) * 1e06)/ gene.length)
}
```

```
dgeFullData <- DGEList(counts, group=as.factor(metadf$genotype))
TMMFullData <- calcNormFactors(dgeFullData, method="TMM")$counts
TPMFullData <- calc_tpm(TMMFullData, gene.length = gene_table$length)
#If I use this filtering method, msh-5 is removed :(
#logcounts = log2(counts + 1)
#tokeep <- rowMeans(logcounts >3)
#msh-5 retained with this filtering
tokeep <- rowSums(counts) > 1
tokeep_stringent <- rowMeans(logdata) > 3
counts_filtered = counts[tokeep, ]
TMMFiltData <- TMMFullData[tokeep,]
TPMFiltData <- TPMFullData[tokeep,]
gene_table_filtered <- gene_table[tokeep,]


counts_stringent <- counts[tokeep_stringent,]
TMMStringent <- TMMFullData[tokeep_stringent,]
TPMStringent <- TPMFullData[tokeep_stringent,]
```

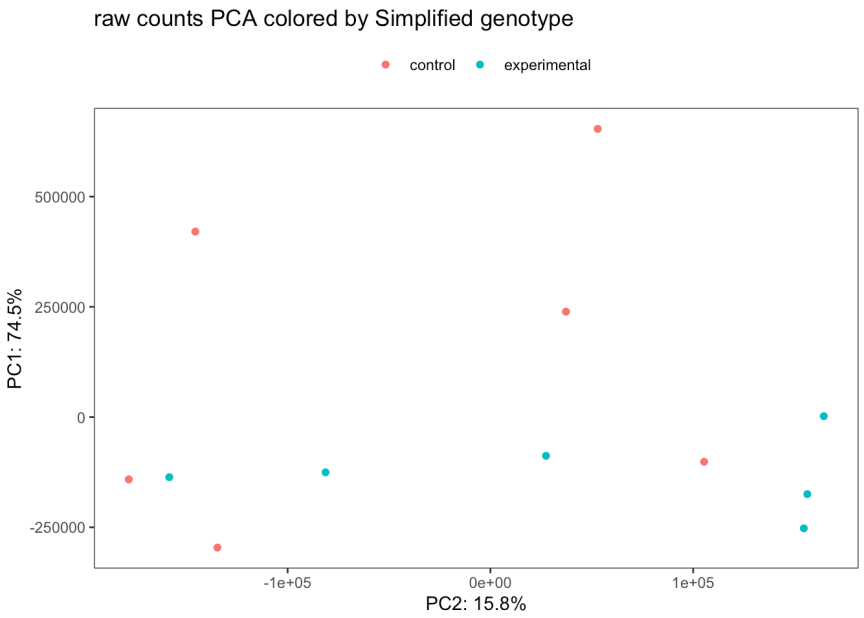
PCA with lightly filtered counts

```
pca_res_counts <- prcomp(t(counts_filtered))
var_explained_counts <- pca_res_counts$sdev^2/sum(pca_res_counts$sdev^2)
```

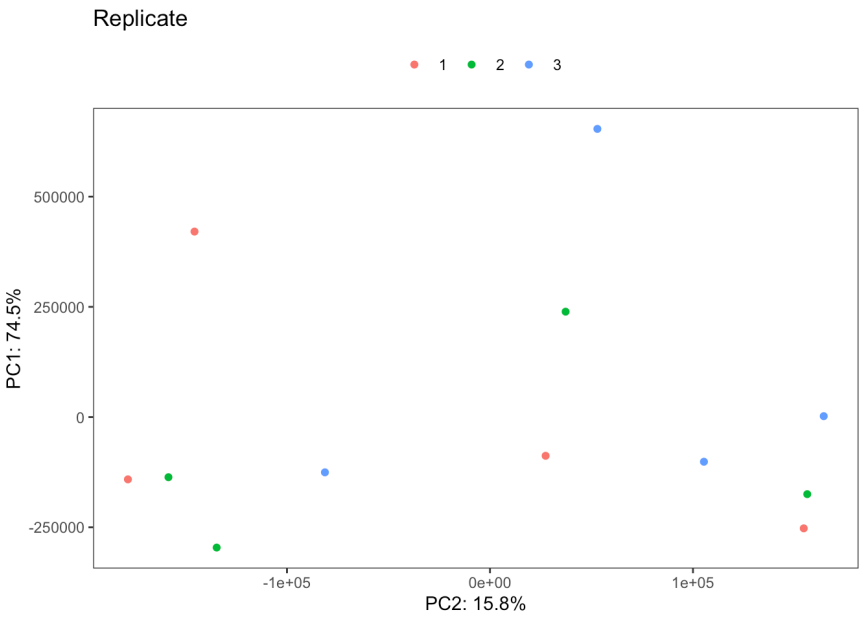
```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by Full genotype") + guides(color = guide_legend
(""))
```



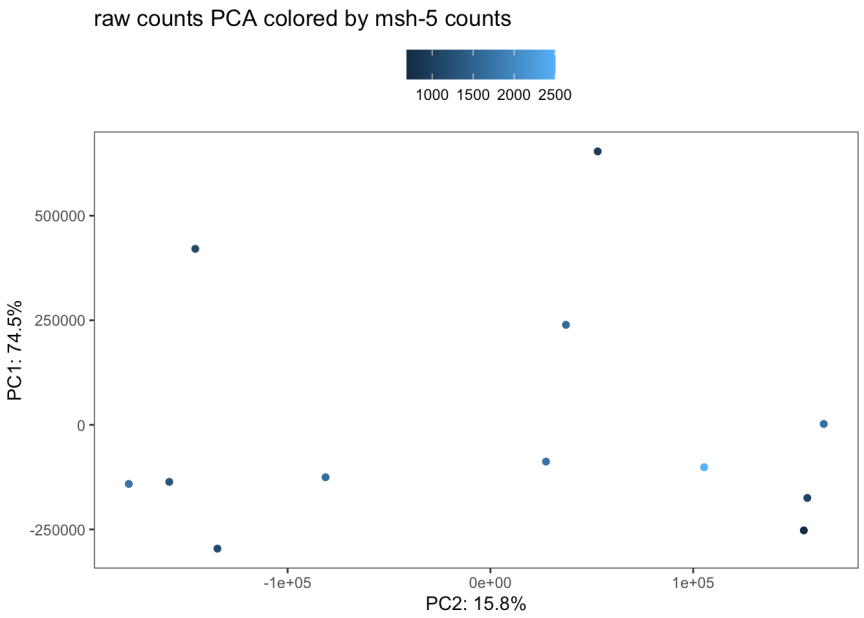

```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by Simplified genotype") + guides(color = guide_leg
end(""))
```



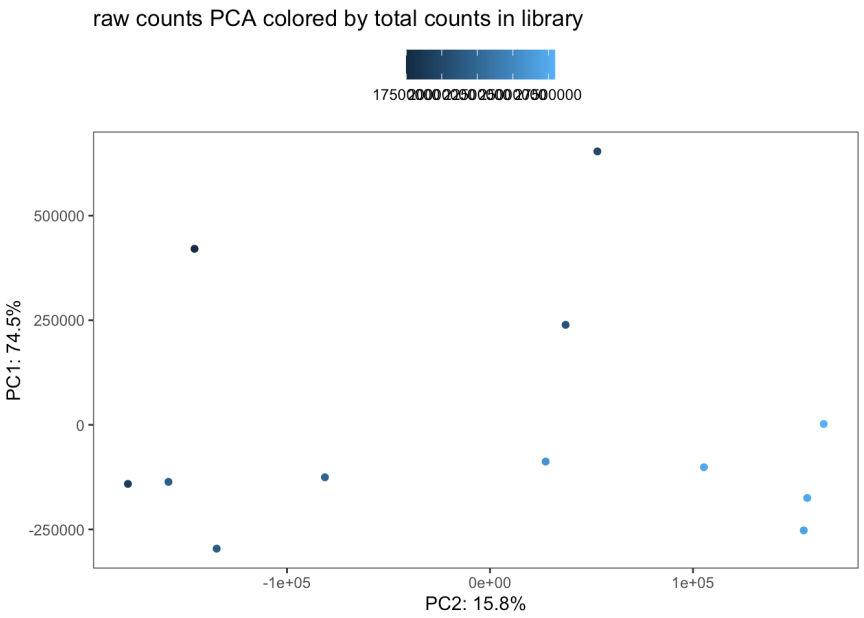

```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("Replicate") + guides(color = guide_legend(""))
```

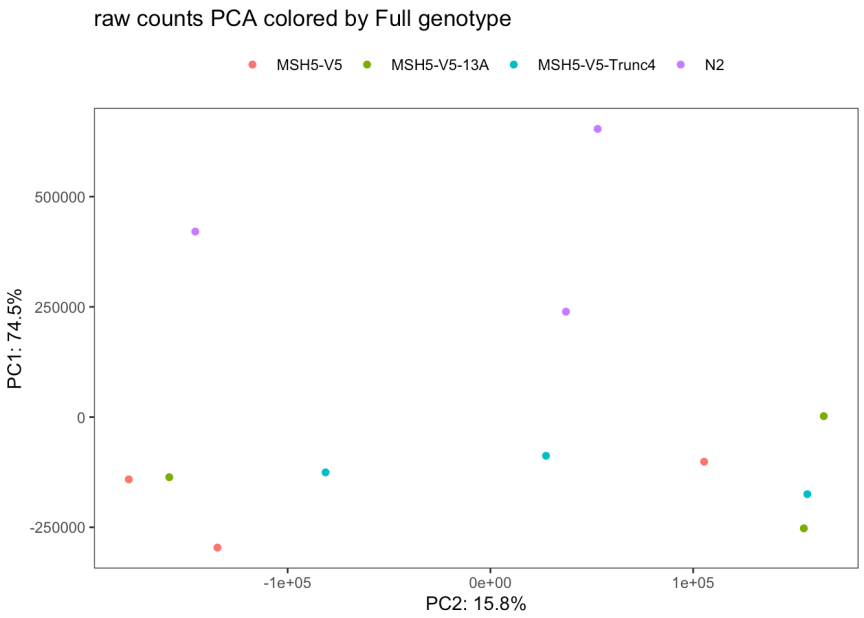



```
roi <- which(rownames(counts_filtered) == gene_table_filtered$ensembl_gene_id[which(g
ene_table_filtered$gene_name == "msh-5")])

pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.numeric(counts_filtered[roi,]))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by msh-5 counts") + labs(color="")
```




```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=colSums(counts_filtered))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by total counts in library") + labs(color="")
```



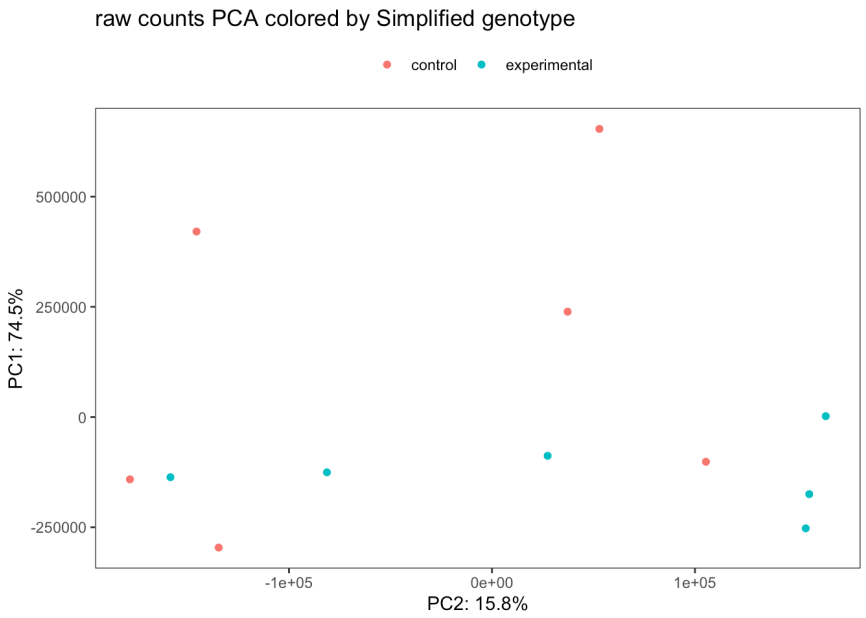


PCA with counts stringently filtered

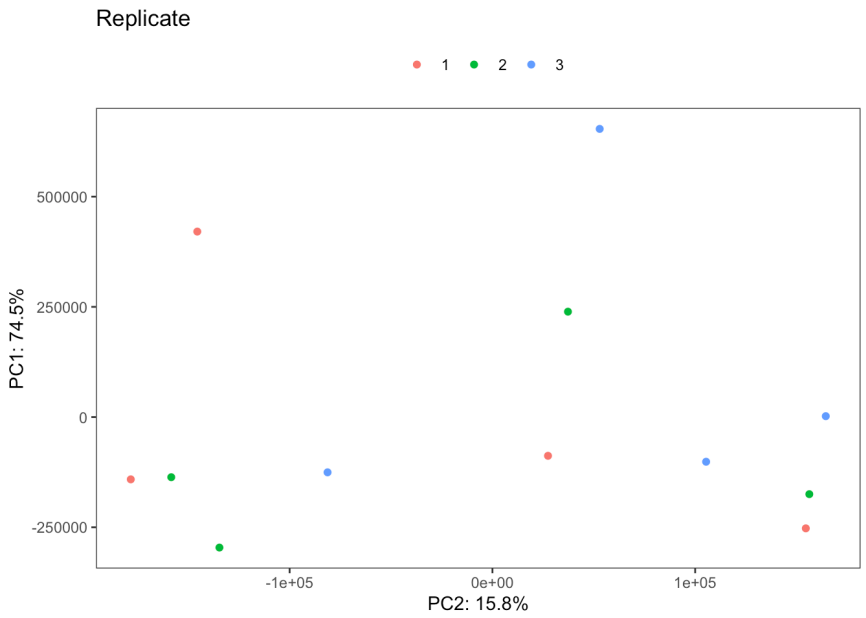
```
pca_res_counts <- prcomp(t(counts_stringent))
var_explained_counts <- pca_res_counts$sdev^2/sum(pca_res_counts$sdev^2)
```

```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by Full genotype") + guides(color = guide_legend
(""))
```

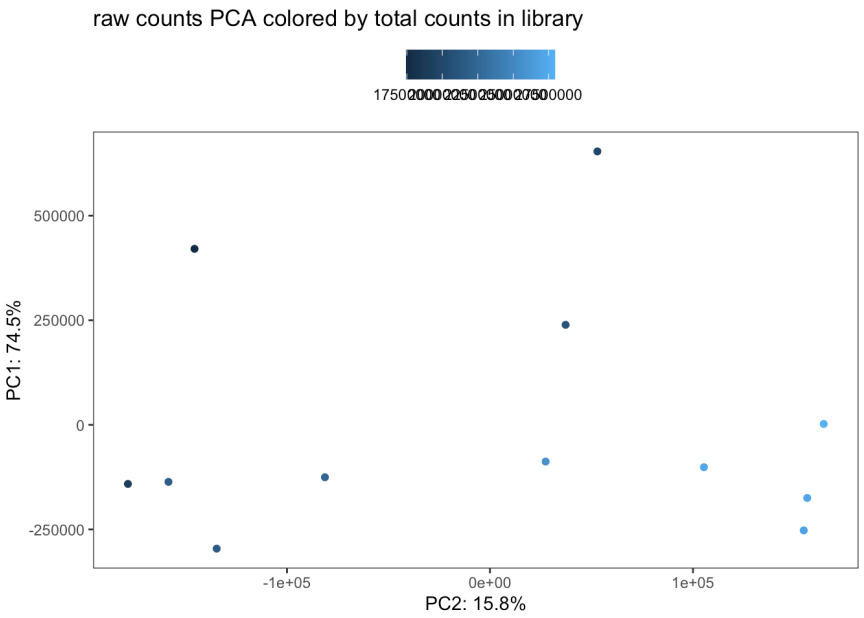
```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by Simplified genotype") + guides(color = guide_leg
end(""))
```

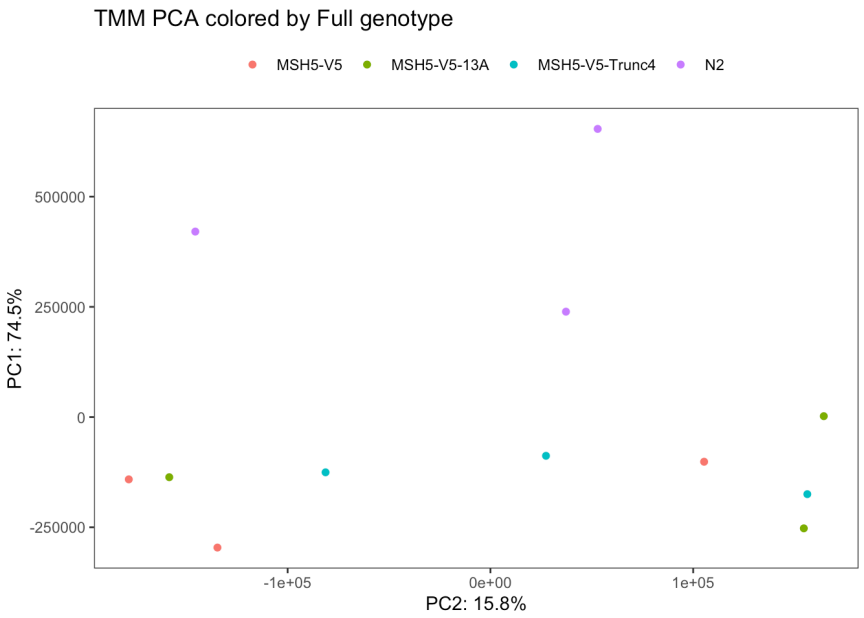



```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("Replicate") + guides(color = guide_legend(""))
```




```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=colSums(counts_stringent))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by total counts in library") + labs(color="")
```



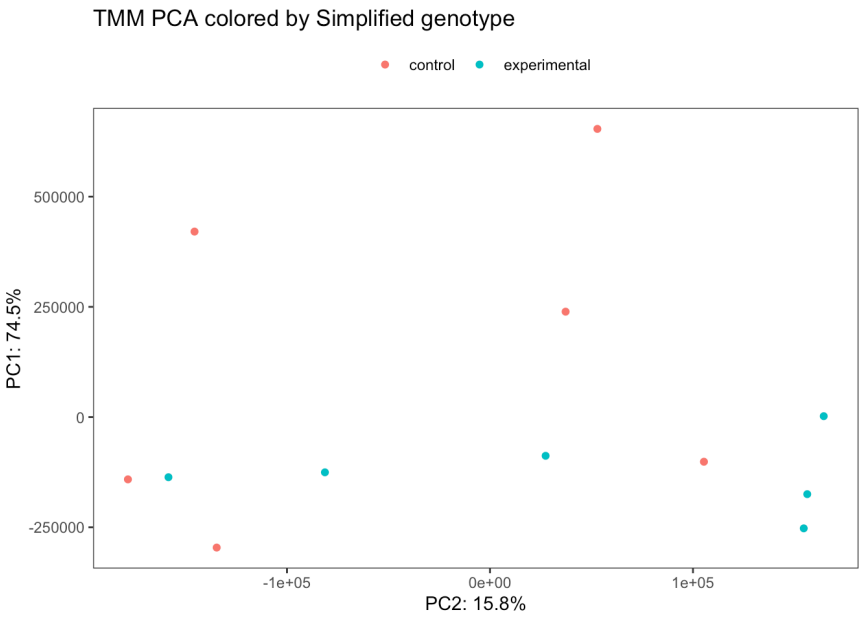


PCA with lightly filtered TMM

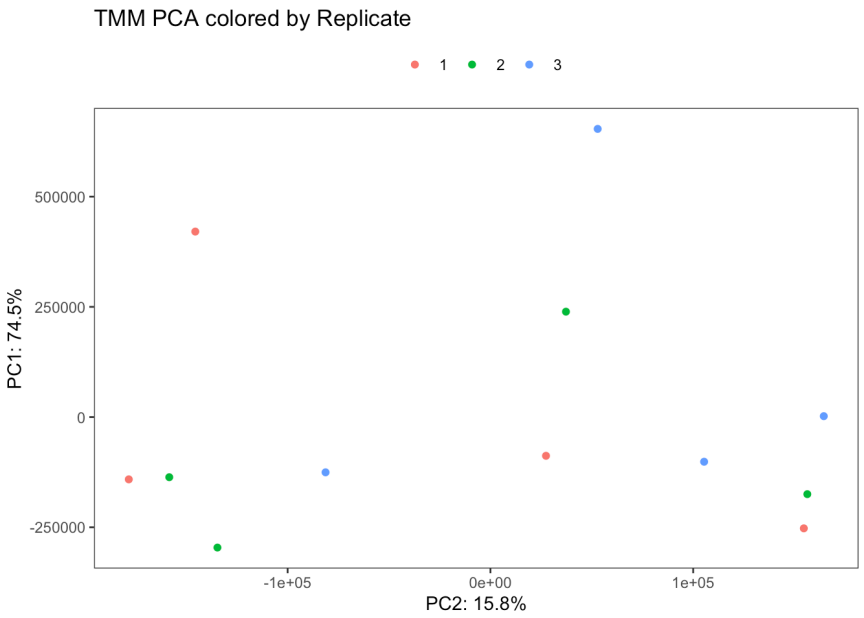
```
pca_res_tmm <- prcomp(t(TMMFiltData))
var_explained_tmm <- pca_res_tmm$sdev^2/sum(pca_res_tmm$sdev^2)
```

```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Full genotype") + guides(color = guide_legend(""))
```

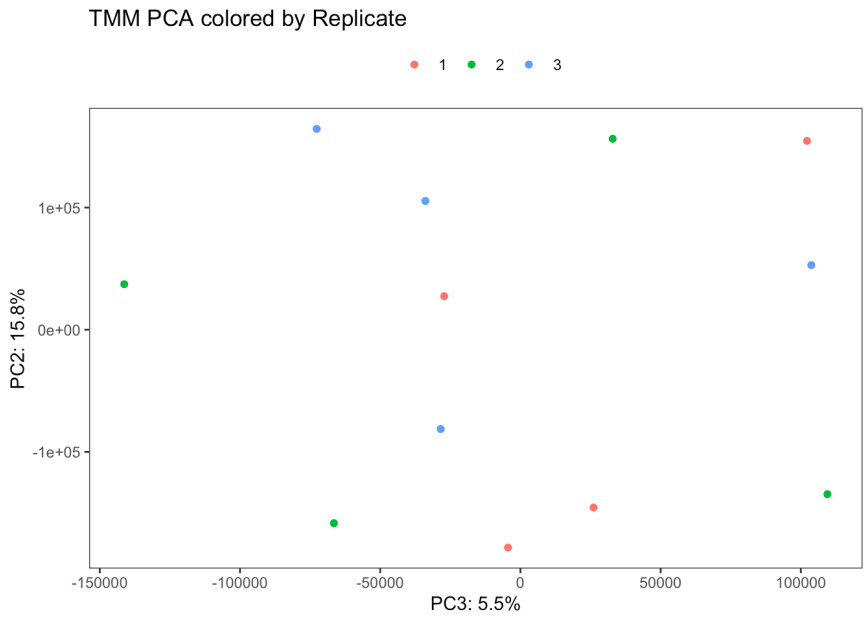
```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Simplified genotype") + guides(color = guide_legend
(""))
```



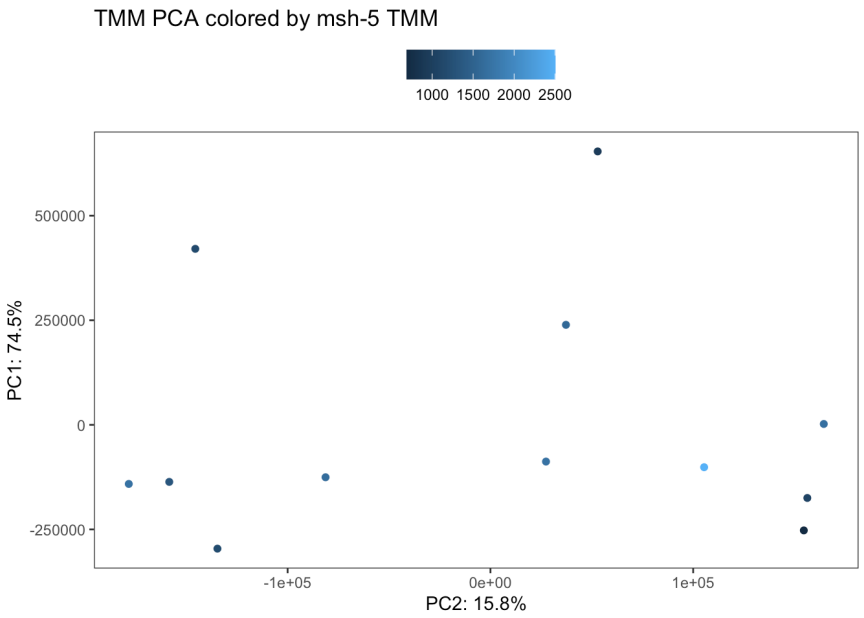

```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Replicate") + guides(color = guide_legend(""))
```




```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC2, x=PC3, color=as.factor(metadf$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%"),
       x=paste0("PC3: ", round(var_explained_tmm[3]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Replicate") + guides(color = guide_legend(""))
```

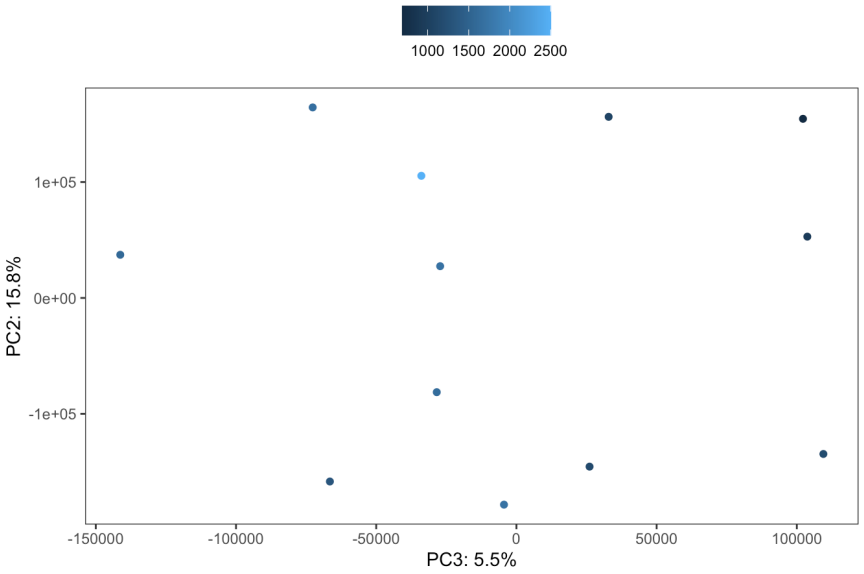



```
roi <- which(rownames(counts_filtered) == gene_table_filtered$ensembl_gene_id[which(g
ene_table_filtered$gene_name == "msh-5")])

pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.numeric(TMMFiltData[roi,]))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by msh-5 TMM") + labs(color="")
```



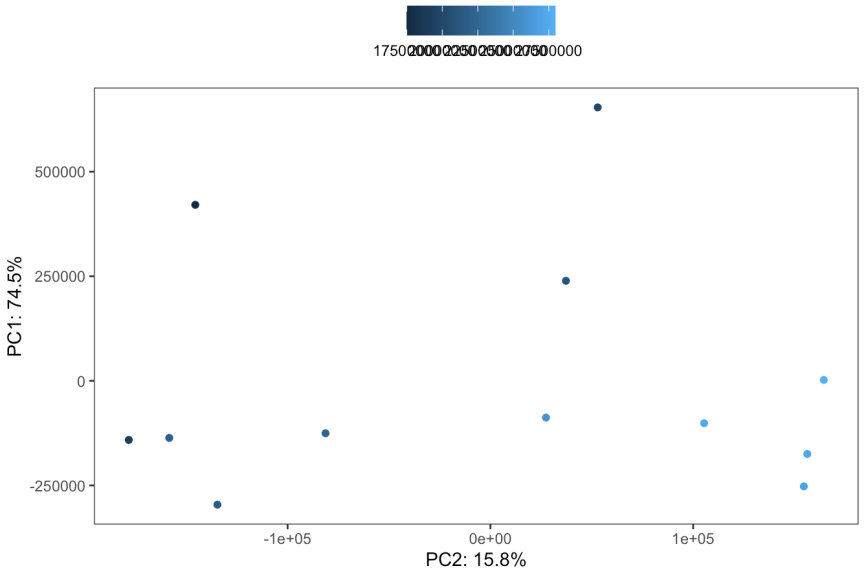

```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC2, x=PC3, color=as.numeric(TMMFiltData[roi,]))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%"),
       x=paste0("PC3: ", round(var_explained_tmm[3]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by msh-5 TMM") + labs(color="")
```

TMM PCA colored by msh-5 TMM




```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=colSums(TMMFiltData))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by TMM sum in library") + labs(color="")
```

TMM PCA colored by TMM sum in library

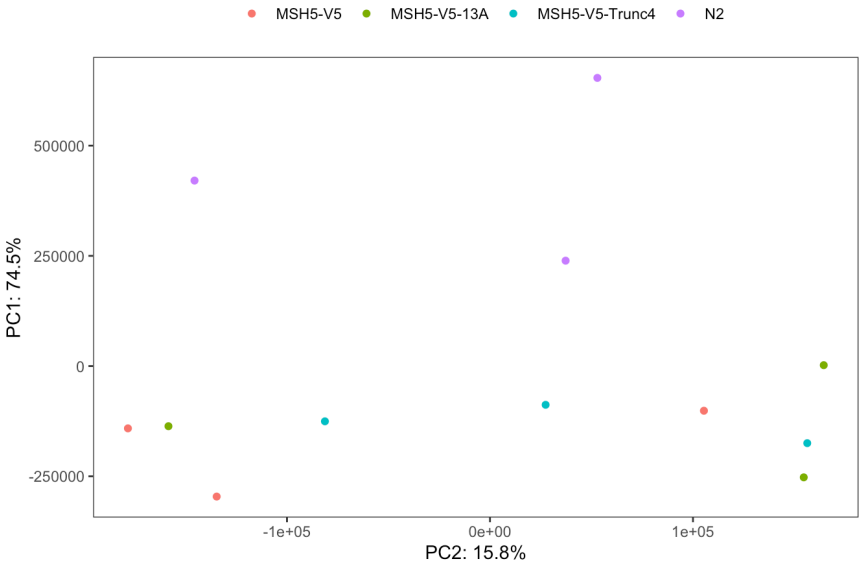


PCA with stringently filtered TMM

```
pca_res_tmm <- prcomp(t(TMMStringent))
var_explained_tmm <- pca_res_tmm$sdev^2/sum(pca_res_tmm$sdev^2)
```

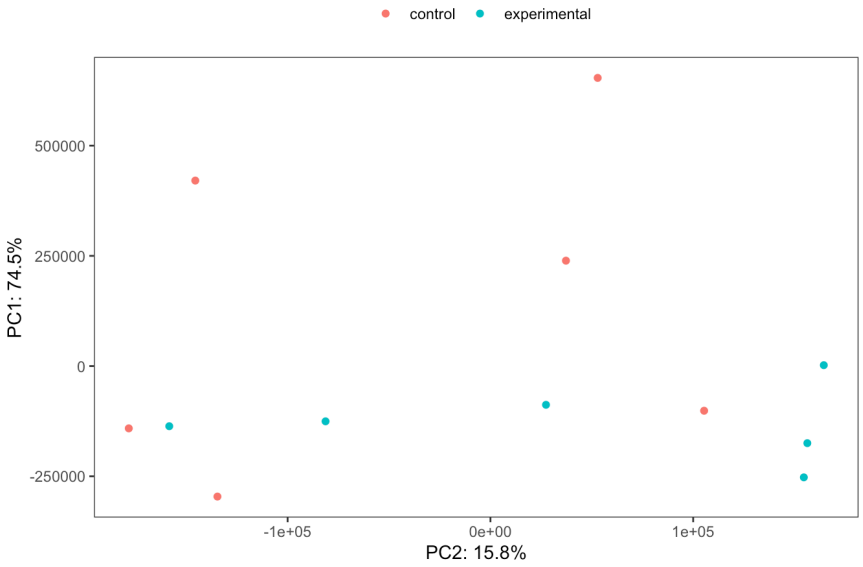
```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Full genotype") + guides(color = guide_legend(""))
```

TMM PCA colored by Full genotype

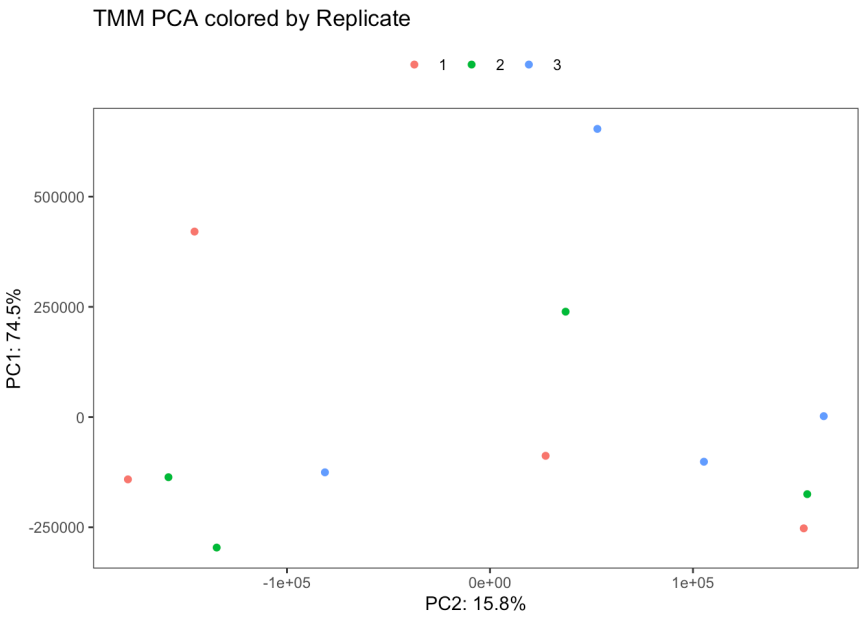



```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Simplified genotype") + guides(color = guide_legend
(""))
```

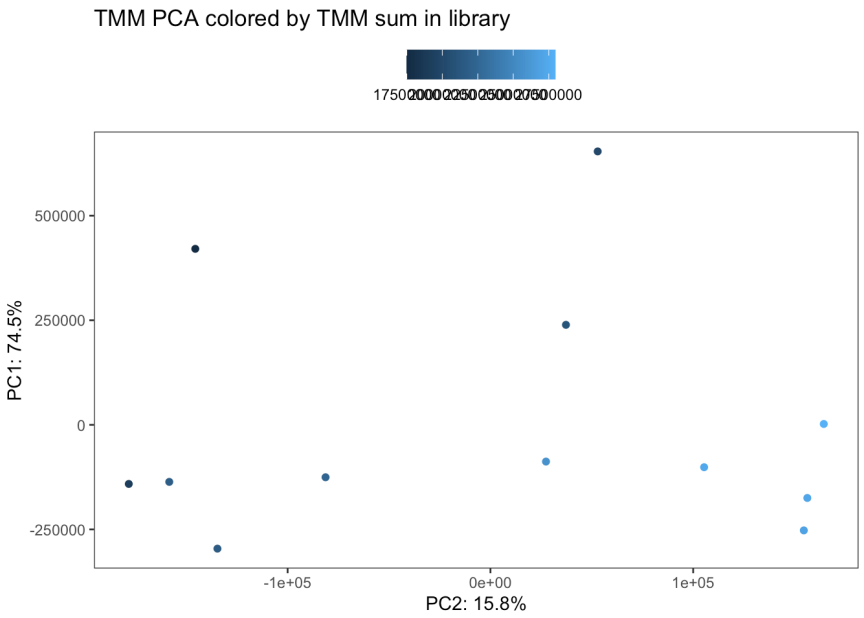
TMM PCA colored by Simplified genotype




```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Replicate") + guides(color = guide_legend(""))
```



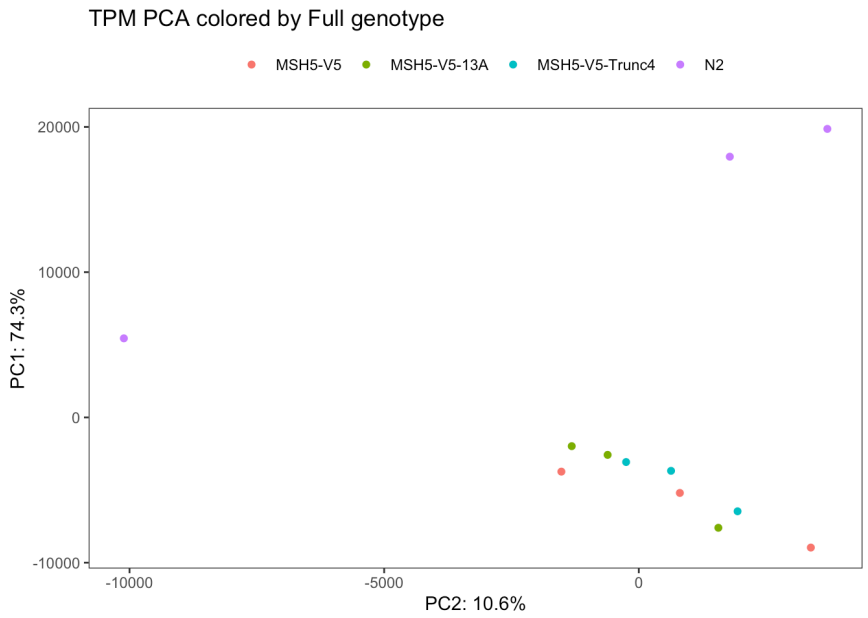

```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=colSums(TMMFiltData))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by TMM sum in library") + labs(color="")
```



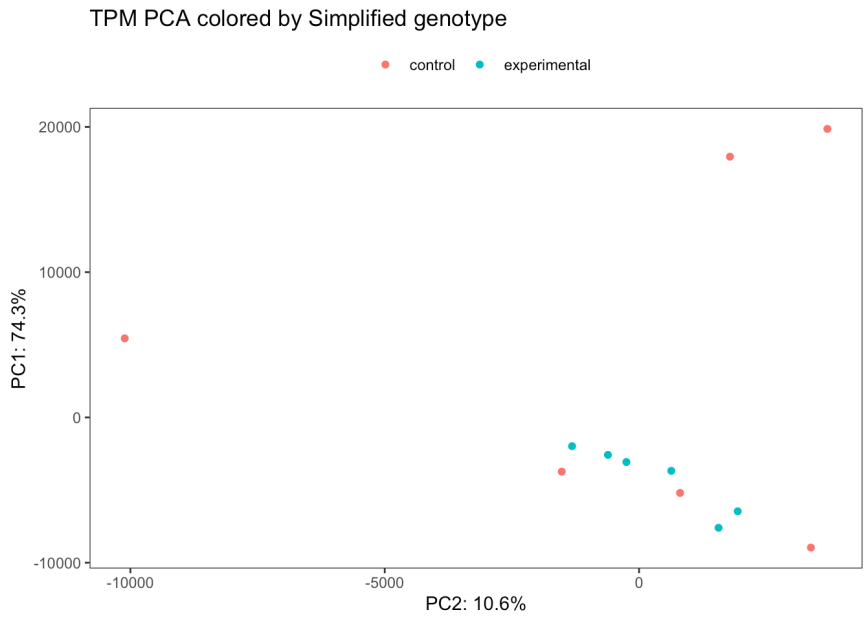
PCA with lightly filtered TPM

```
pca_res_tpm <- prcomp(t(TPMFiltData))
var_explained_tpm <- pca_res_tpm$sdev^2/sum(pca_res_tpm$sdev^2)
```

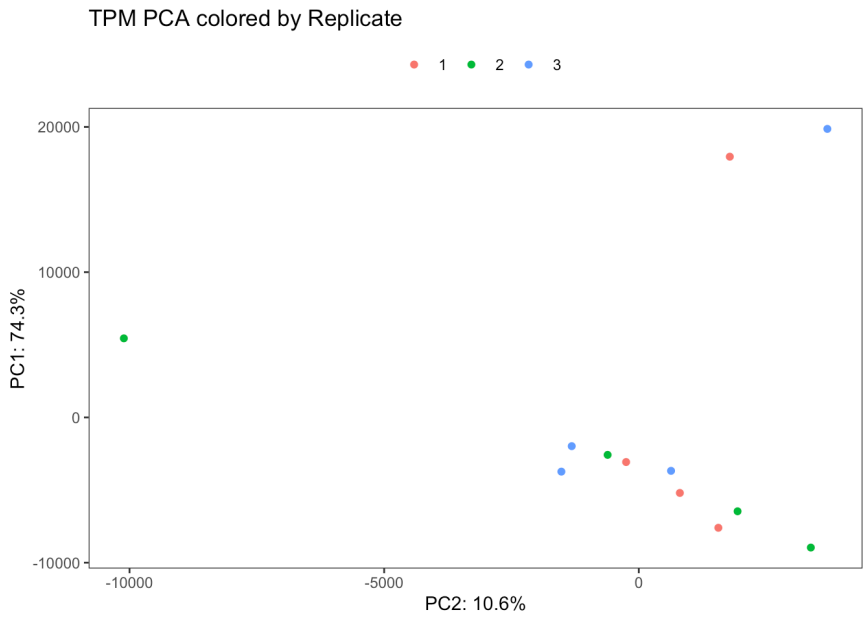
```
pca_res_tpm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tpm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tpm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TPM PCA colored by Full genotype") + guides(color = guide_legend(""))
```



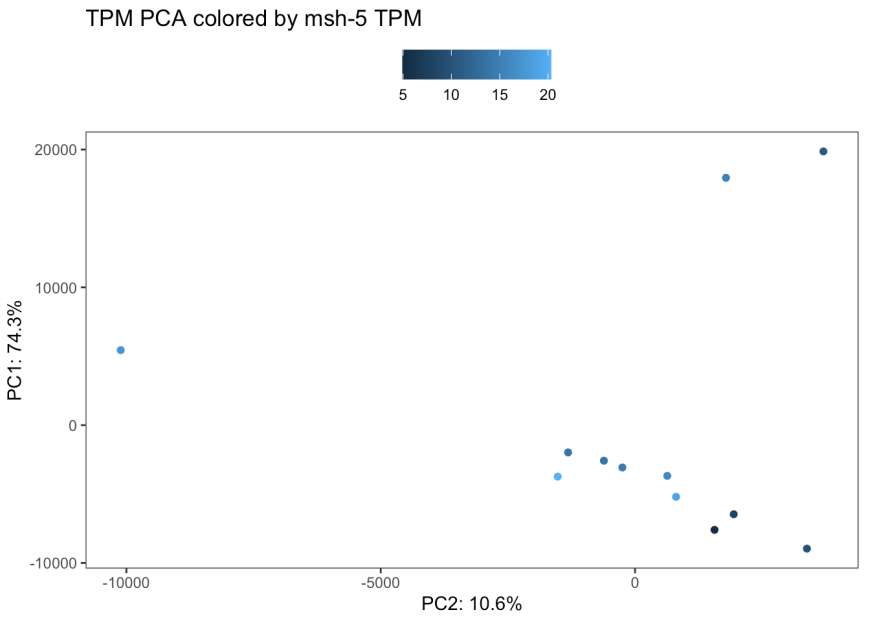

```
pca_res_tpm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tpm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tpm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TPM PCA colored by Simplified genotype") + guides(color = guide_legend
(""))
```




```
pca_res_tpm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tpm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tpm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TPM PCA colored by Replicate") + guides(color = guide_legend(""))
```



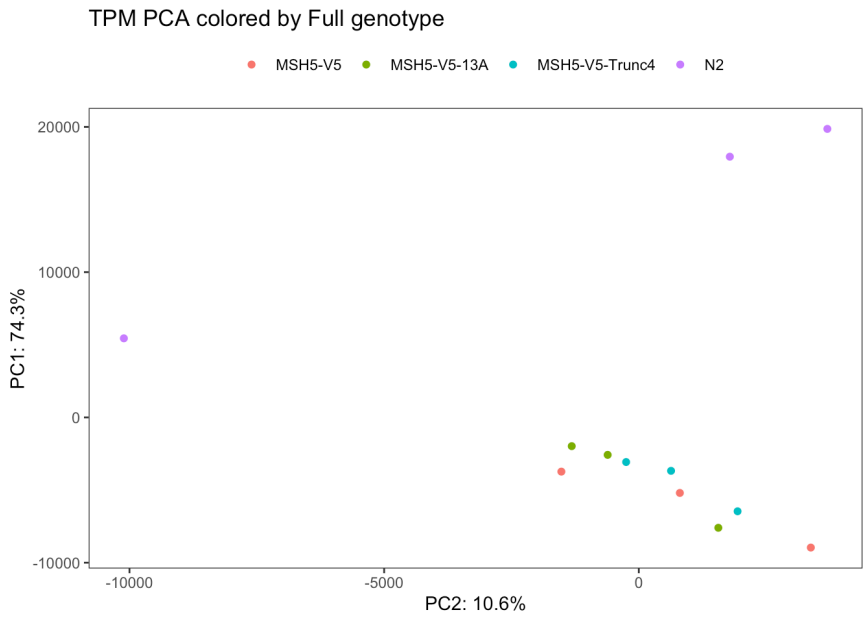

```
roi <- which(rownames(counts_filtered) == gene_table_filtered$ensembl_gene_id[which(g
ene_table_filtered$gene_name == "msh-5")])

pca_res_tpm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.numeric(TPMFiltData[roi,]))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tpm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tpm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TPM PCA colored by msh-5 TPM") + labs(color="")
```



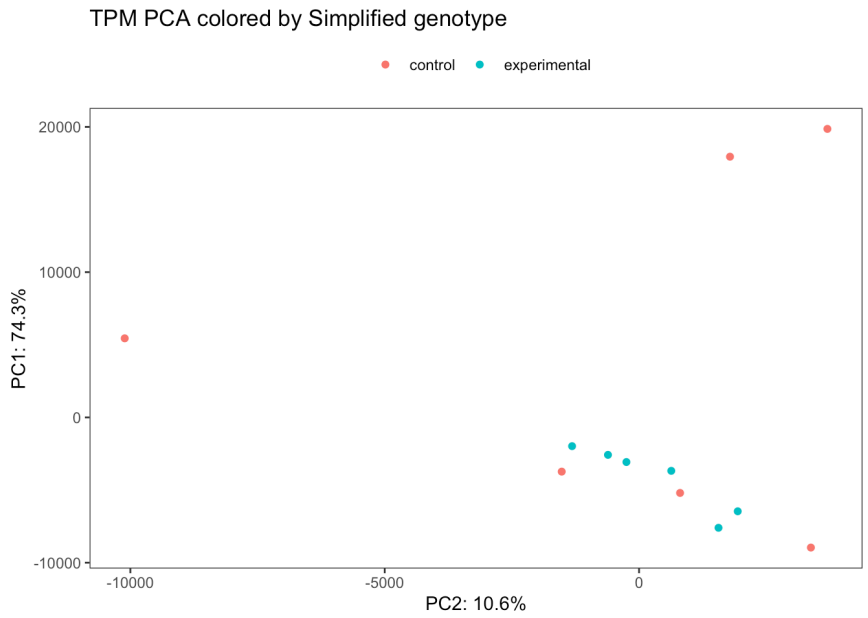
PCA with stringently filtered TPM

```
pca_res_tpm <- prcomp(t(TPMStringent))
var_explained_tpm <- pca_res_tpm$sdev^2/sum(pca_res_tpm$sdev^2)
```

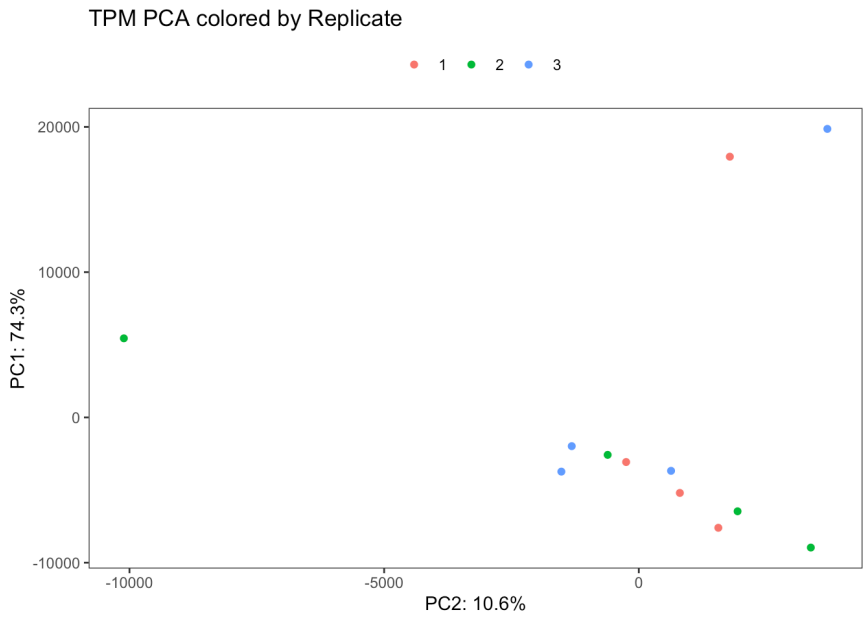
```
pca_res_tpm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tpm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tpm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TPM PCA colored by Full genotype") + guides(color = guide_legend(""))
```




```
pca_res_tpm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tpm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tpm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TPM PCA colored by Simplified genotype") + guides(color = guide_legend
(""))
```




```
pca_res_tpm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tpm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tpm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TPM PCA colored by Replicate") + guides(color = guide_legend(""))
```



removing N2 from samples

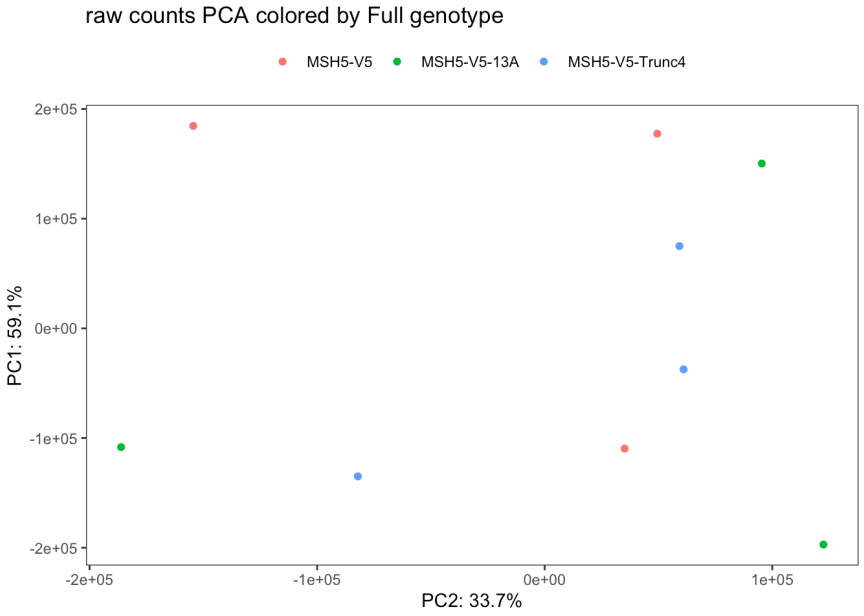
```
metadf_non2 <- metadf[which(metadf$genotype!="N2"),] %>% mutate_at(vars(genotype, sim
plified_genotype2, replicate), factor)
counts_non2 <- counts[,1:9]
TMMFull_non2 <- TMMFullData[,1:9]
TPMFull_non2 <- TPMFullData[,1:9]

tokeep <- rowSums(counts_non2) > 1
logdata_non2 <- log2(counts_non2 + 1)
tokeep_stringent <- rowMeans(logdata_non2) > 3
counts_non2_filtered = counts_non2[tokeep, ]
TMMFiltData_non2 <- TMMFull_non2[tokeep,]
TPMFiltData_non2 <- TPMFull_non2[tokeep,]
gene_table_filtered <- gene_table[tokeep,]


counts_non2_stringent <- counts_non2[tokeep_stringent,]
TMMStringent_non2 <- TMMFull_non2[tokeep_stringent,]
TPMStringent_non2 <- TPMFull_non2[tokeep_stringent,]
```

repeating PCAs without n2 samples

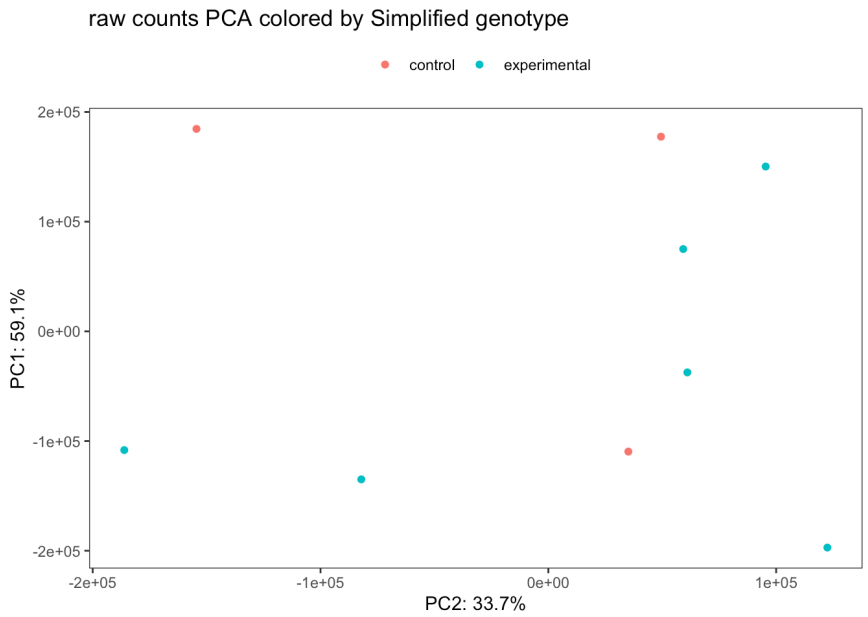
counts based

lite filter

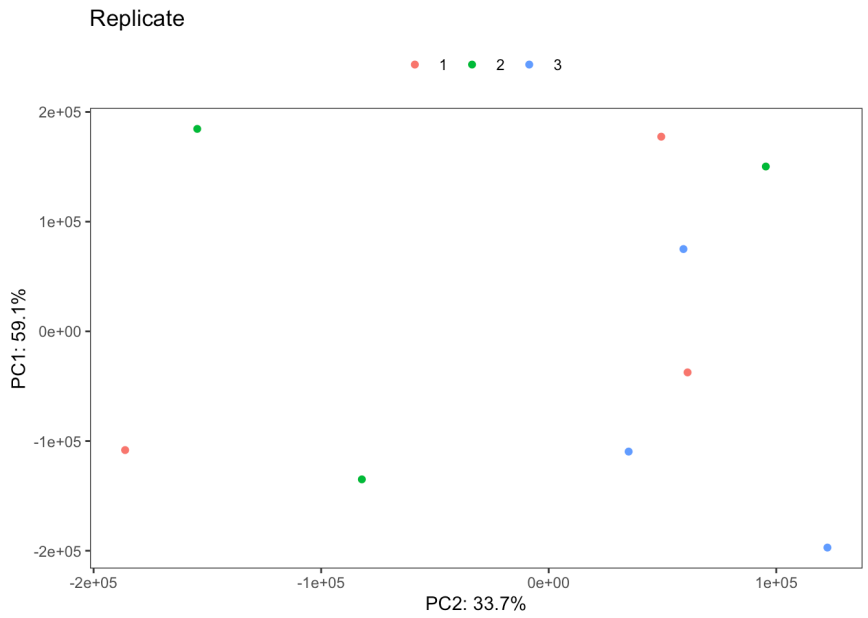
```
pca_res_counts <- prcomp(t(counts_non2_filtered))
var_explained_counts <- pca_res_counts$sdev^2/sum(pca_res_counts$sdev^2)

pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by Full genotype") + guides(color = guide_legend
(""))
```



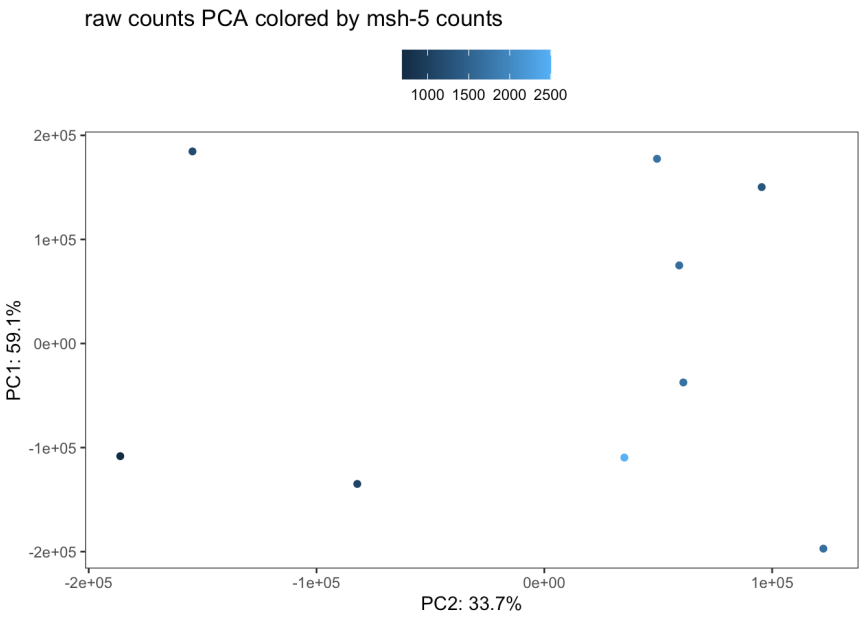
```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by Simplified genotype") + guides(color = guide_leg
end(""))
```



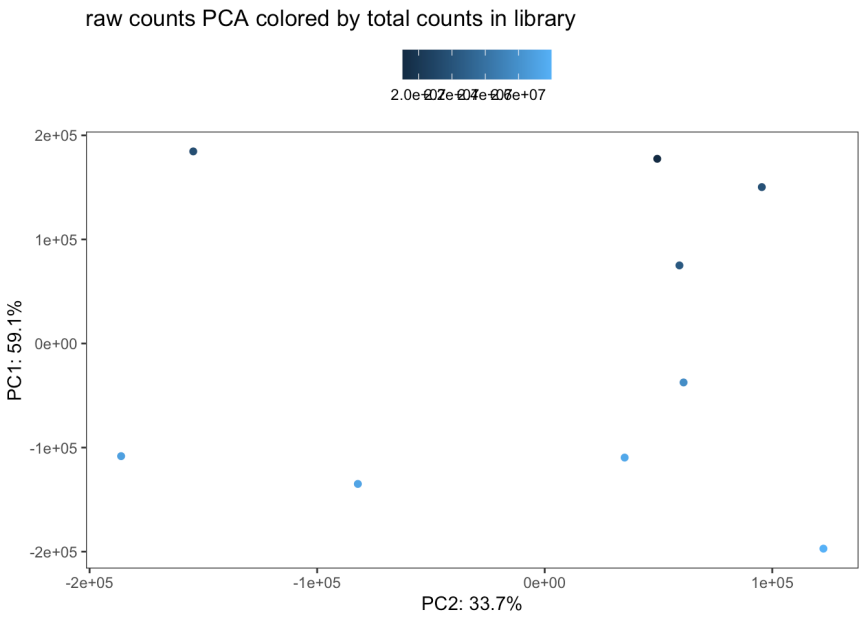

```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("Replicate") + guides(color = guide_legend(""))
```




```
roi <- which(rownames(counts_non2_filtered) == gene_table_filtered$ensembl_gene_id[wh
ich(gene_table_filtered$gene_name == "msh-5")])

pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.numeric(counts_non2_filtered[roi,]))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by msh-5 counts") + labs(color="")
```

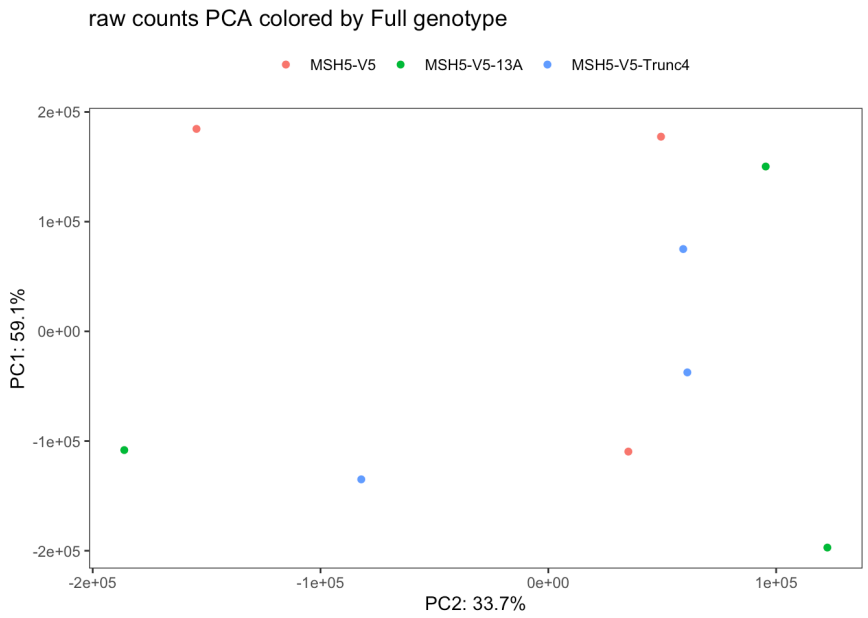



```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=colSums(counts_non2_filtered))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by total counts in library") + labs(color="")
```

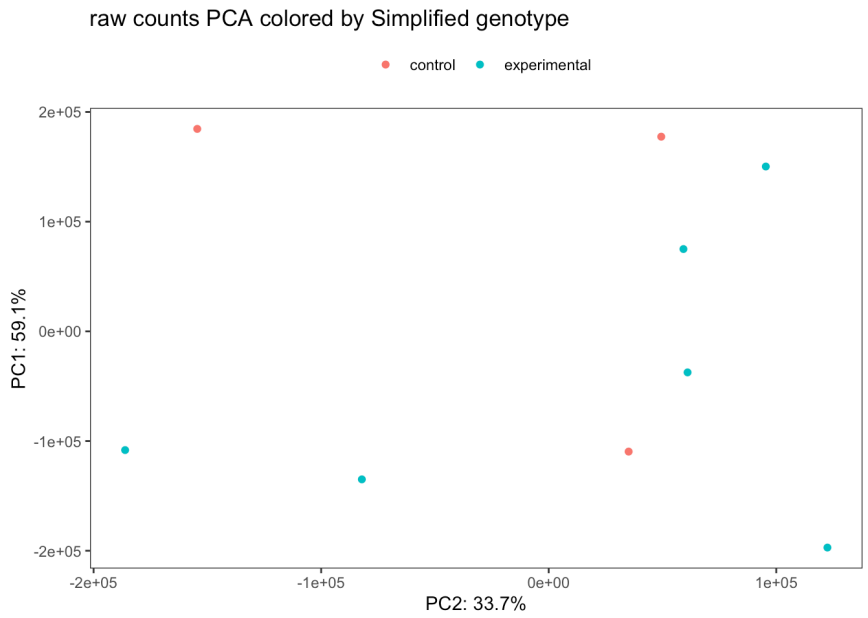


stringent

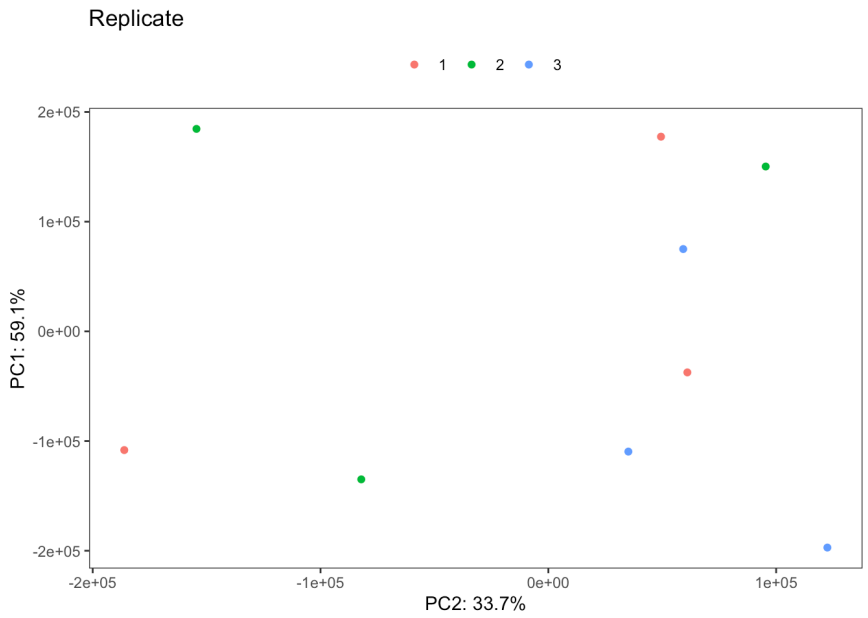
```
pca_res_counts <- prcomp(t(counts_non2_stringent))
var_explained_counts <- pca_res_counts$sdev^2/sum(pca_res_counts$sdev^2)

pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by Full genotype") + guides(color = guide_legend
(""))
```



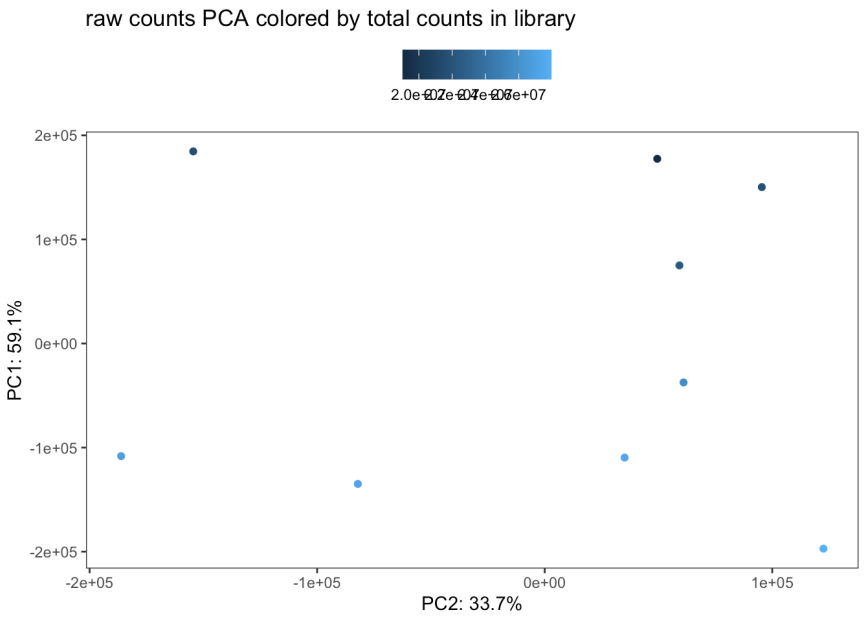

```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by Simplified genotype") + guides(color = guide_leg
end(""))
```




```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("Replicate") + guides(color = guide_legend(""))
```



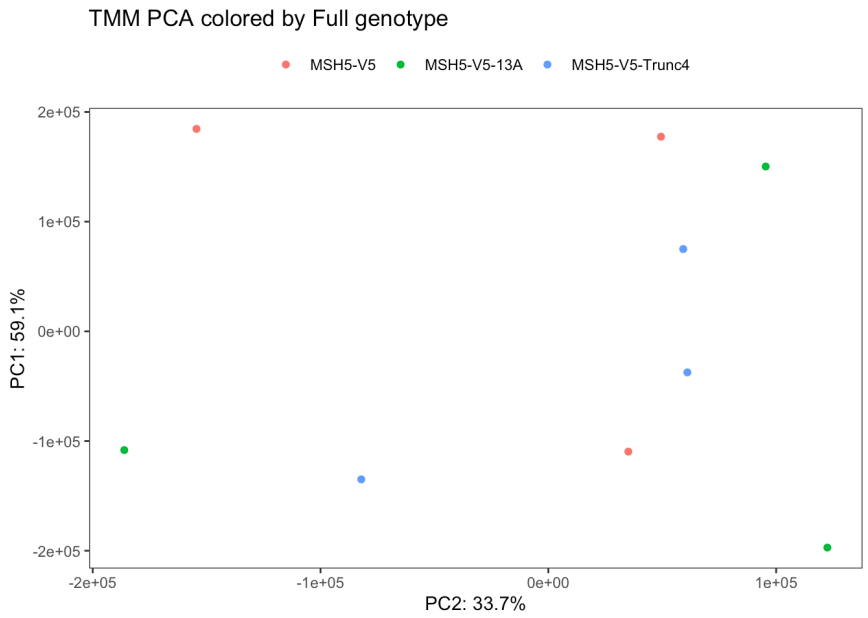

```
pca_res_counts$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=colSums(counts_non2_stringent))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_counts[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_counts[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("raw counts PCA colored by total counts in library") + labs(color="")
```



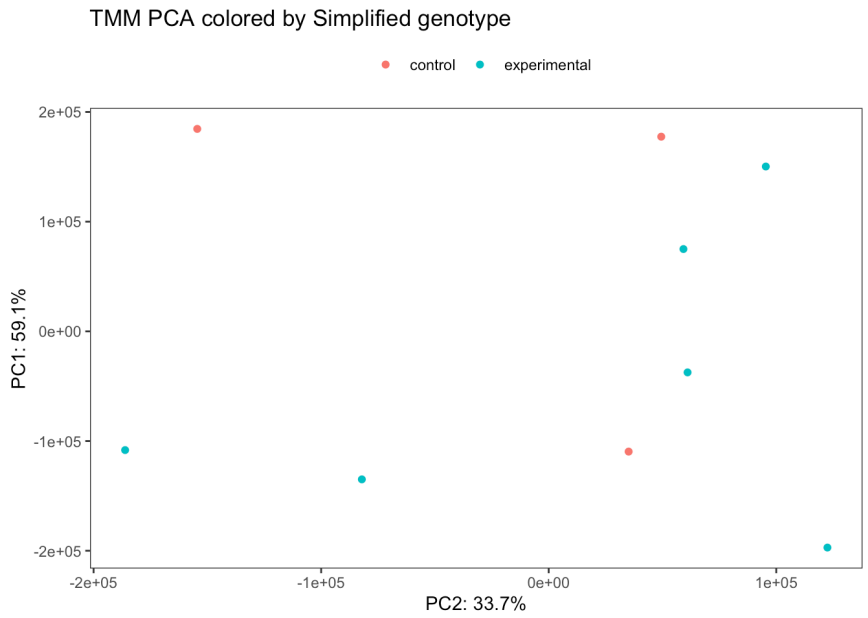
TMM based

lite filter

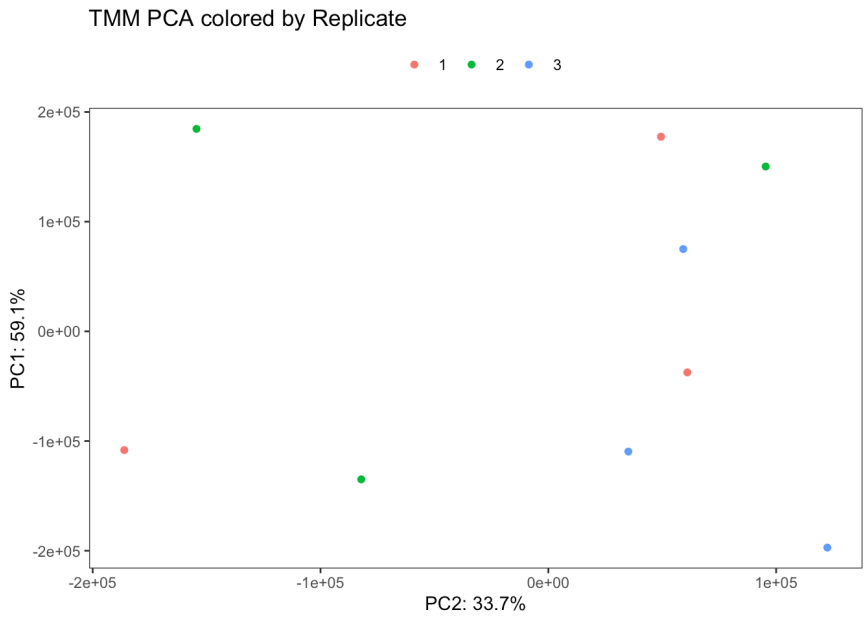
```
pca_res_tmm <- prcomp(t(TMMFiltData_non2))
var_explained_tmm <- pca_res_tmm$sdev^2/sum(pca_res_tmm$sdev^2)

pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Full genotype") + guides(color = guide_legend(""))
```



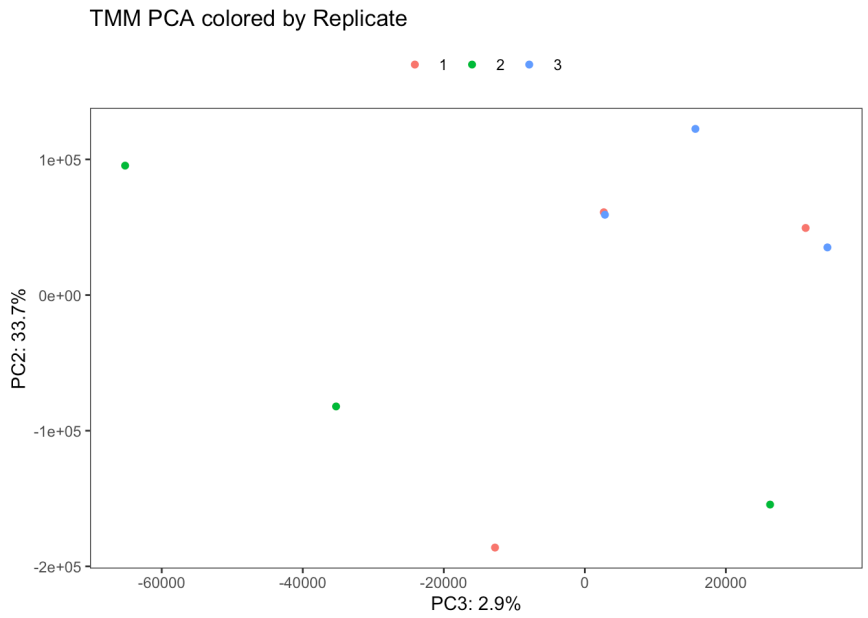

```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Simplified genotype") + guides(color = guide_legend
(""))
```



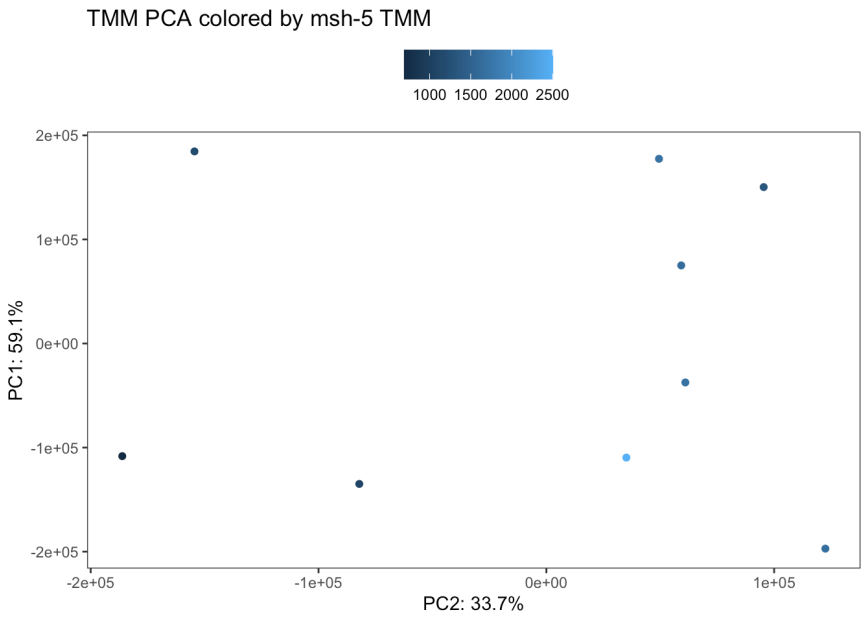

```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Replicate") + guides(color = guide_legend(""))
```



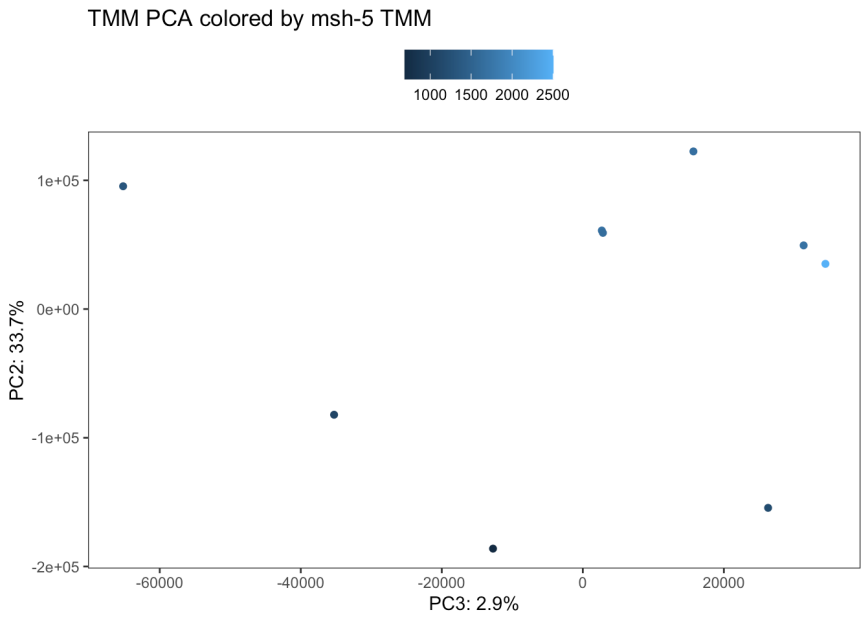

```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC2, x=PC3, color=as.factor(metadf_non2$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%"),
       x=paste0("PC3: ", round(var_explained_tmm[3]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Replicate") + guides(color = guide_legend(""))
```



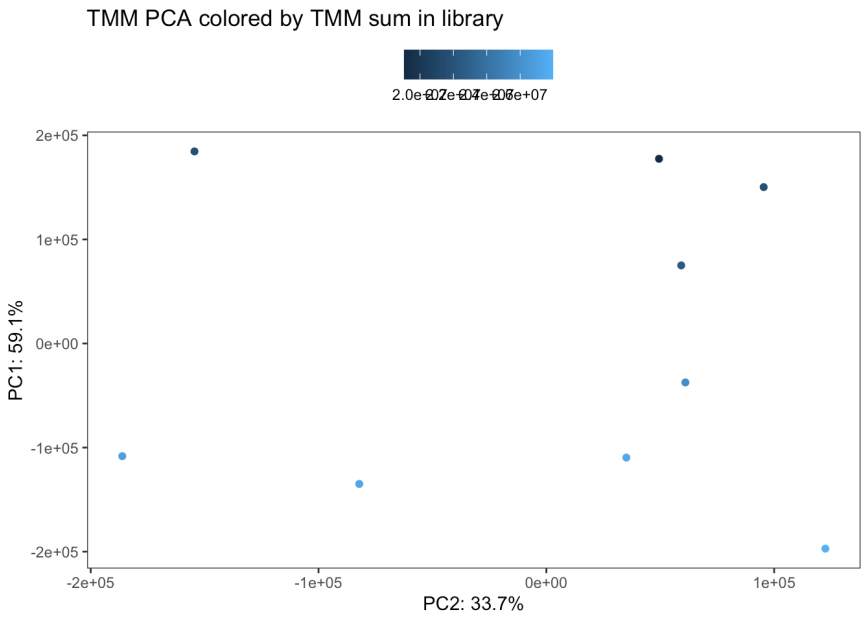

```
roi <- which(rownames(counts_non2_filtered) == gene_table_filtered$ensembl_gene_id[wh
ich(gene_table_filtered$gene_name == "msh-5")])

pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.numeric(TMMFiltData_non2[roi,]))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by msh-5 TMM") + labs(color="")
```

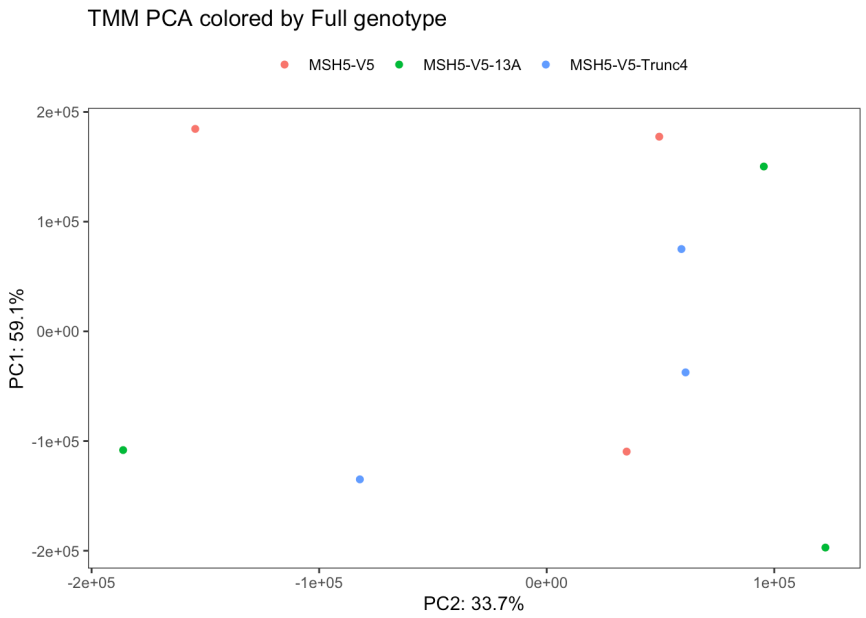



```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC2, x=PC3, color=as.numeric(TMMFiltData_non2[roi,]))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%"),
       x=paste0("PC3: ", round(var_explained_tmm[3]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by msh-5 TMM") + labs(color="")
```




```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=colSums(TMMFiltData_non2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by TMM sum in library") + labs(color="")
```



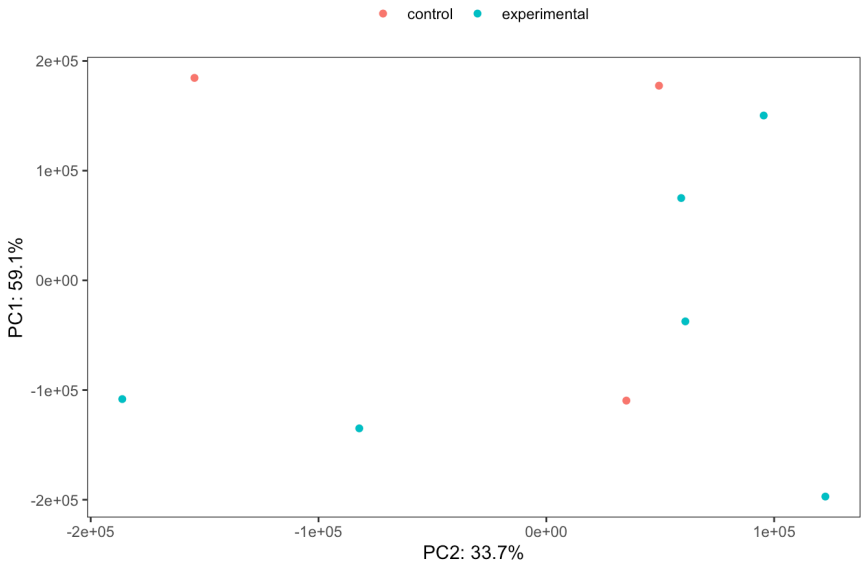


stringent

```
pca_res_tmm <- prcomp(t(TMMStringent_non2))
var_explained_tmm <- pca_res_tmm$sdev^2/sum(pca_res_tmm$sdev^2)

pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$genotype))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Full genotype") + guides(color = guide_legend(""))
```

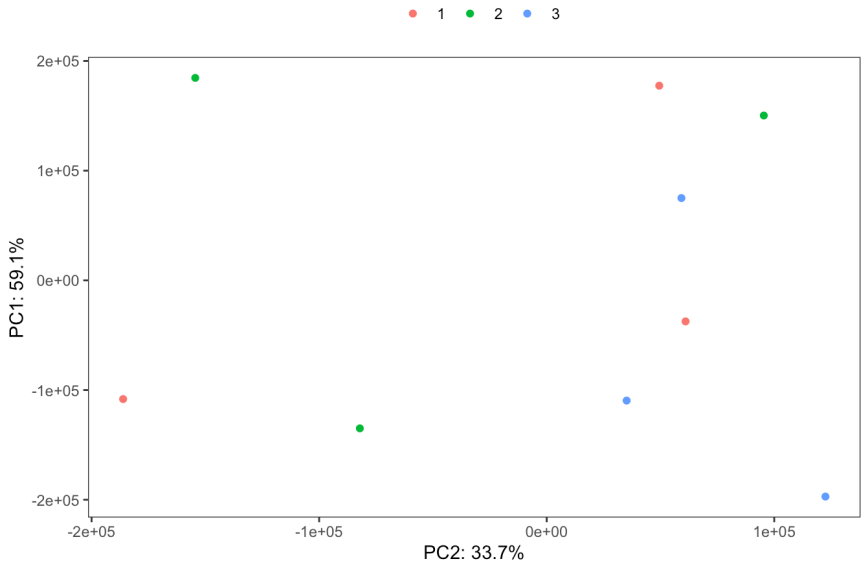
```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$simplified_genotype2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Simplified genotype") + guides(color = guide_legend
(""))
```

TMM PCA colored by Simplified genotype

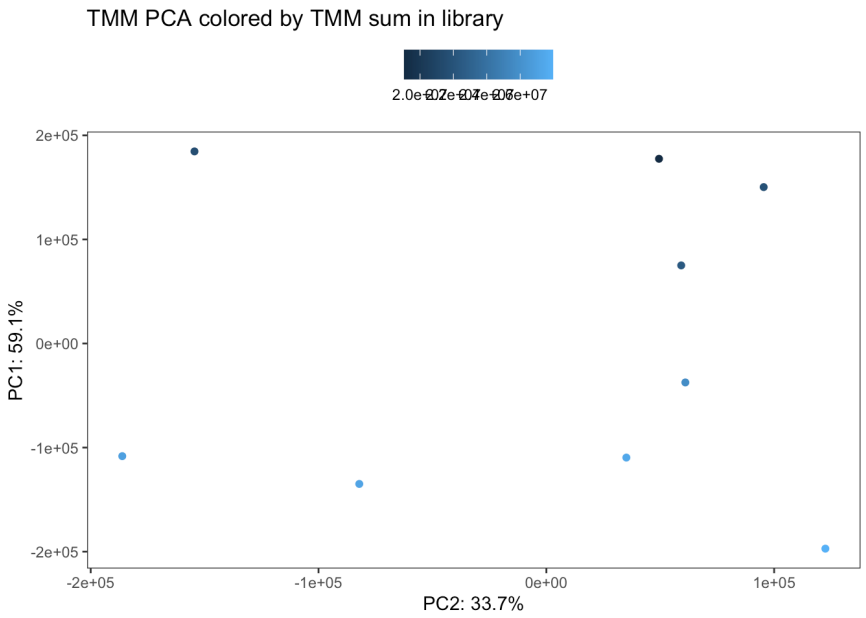


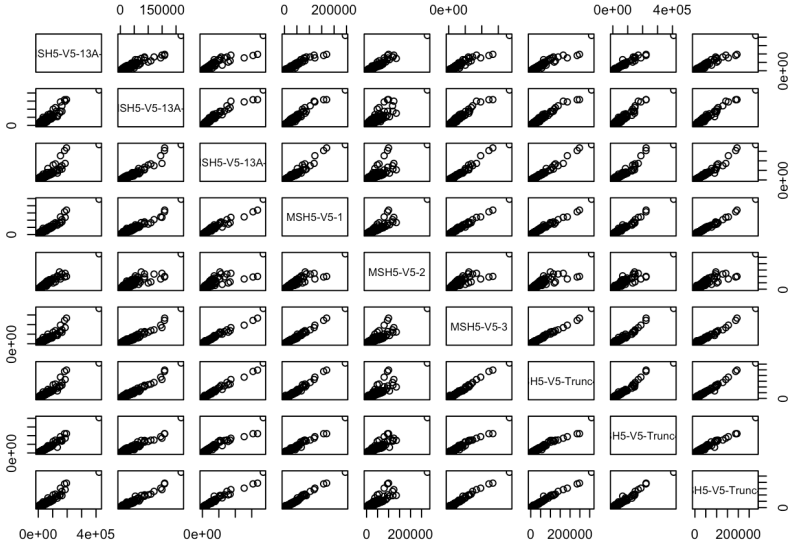

```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=as.factor(metadf_non2$replicate))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by Replicate") + guides(color = guide_legend(""))
```

TMM PCA colored by Replicate




```
pca_res_tmm$x %>%
  as.data.frame %>%
  ggplot(aes(y=PC1, x=PC2, color=colSums(TMMFiltData_non2))) +
  geom_point() + theme_bw() +
  labs(y=paste0("PC1: ", round(var_explained_tmm[1]*100, 1), "%"),
       x=paste0("PC2: ", round(var_explained_tmm[2]*100, 1), "%")) +
  theme(legend.position = "top", panel.background = element_blank(), panel.grid = ele
ment_blank()) +
  ggtitle("TMM PCA colored by TMM sum in library") + labs(color="")
```





compare samples pairwise in scatter plots

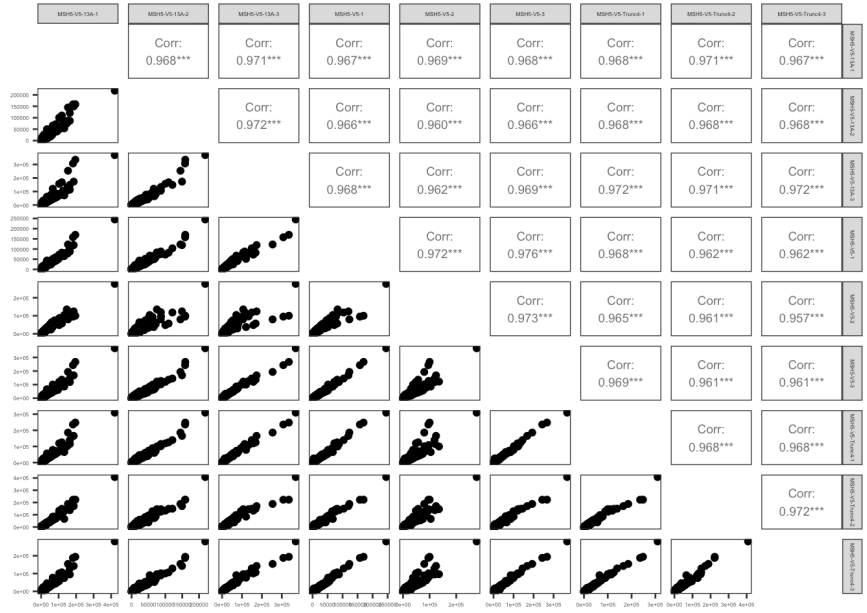
```
pairs(as.data.frame(TMMFiltData_non2), pch=1)
lines(0:max(TMMFiltData_non2), 0:max(TMMFiltData_non2))
```

```
ggpairs(as.data.frame(TMMFiltData_non2), diag = list(continuous = "blankDiag"), upper
= list(continuous = wrap("cor", method="spearman", size = 2.5))) + theme_bw() + theme
(panel.background = element_blank(), panel.grid = element_blank()) + theme(text = ele
ment_text(size = 4))
```

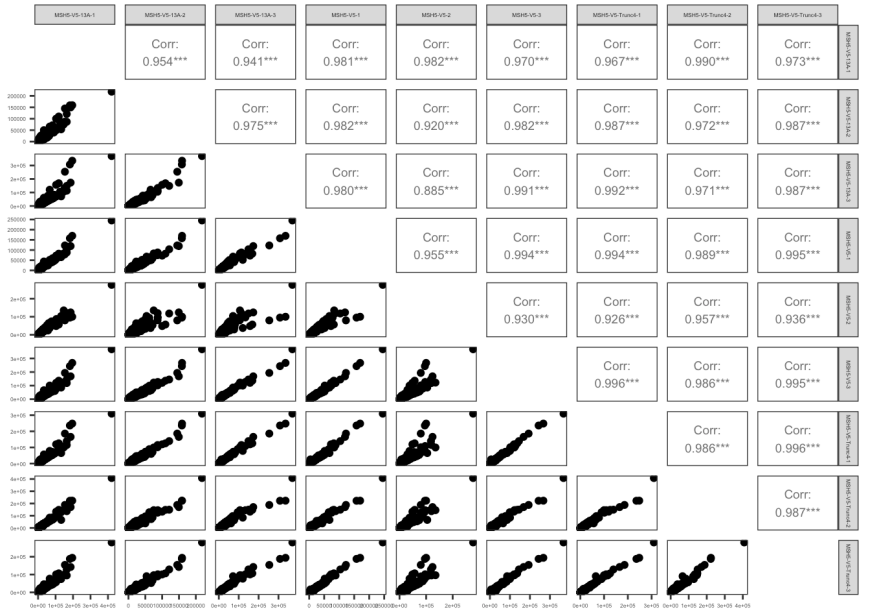
```
## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties
```

```
## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties
```

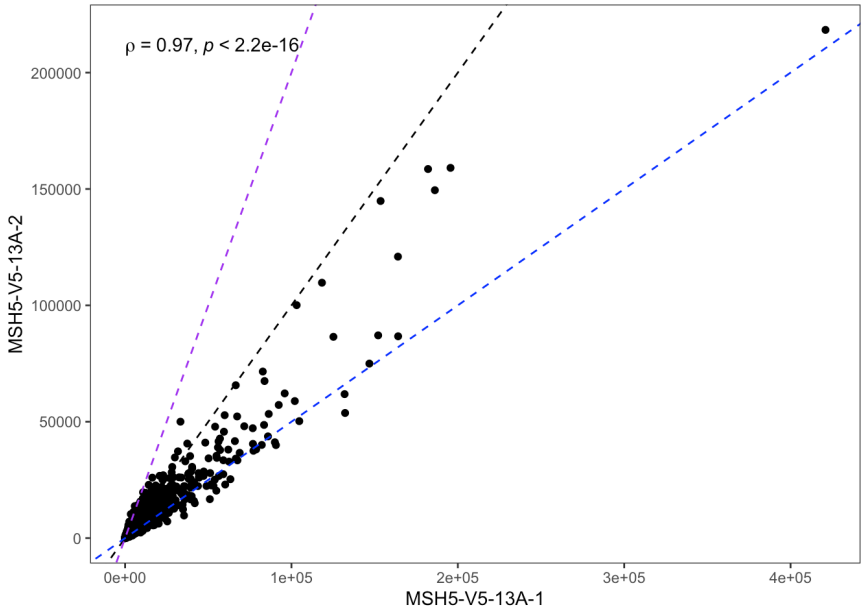
```
## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties

## Warning in cor.test.default(x, y, method = method, use = use): Cannot compute
## exact p-value with ties
```

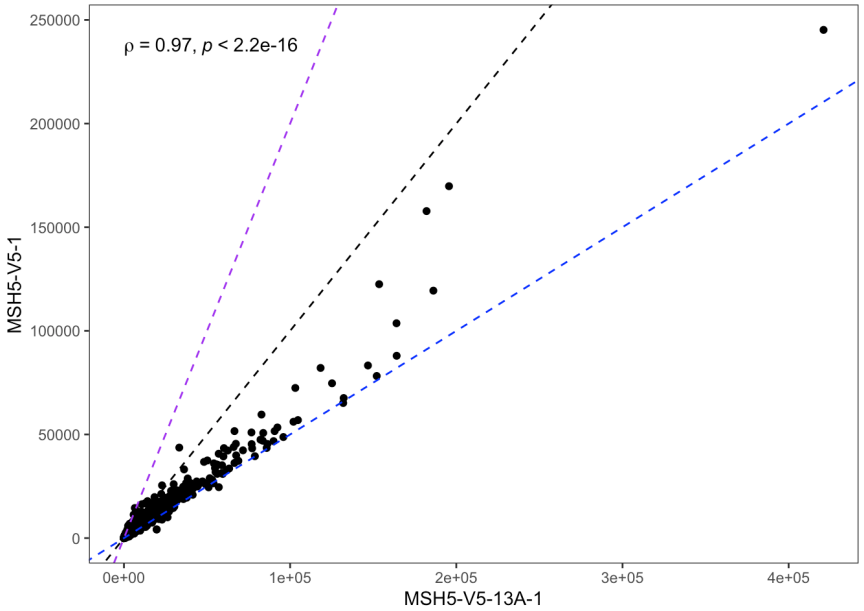


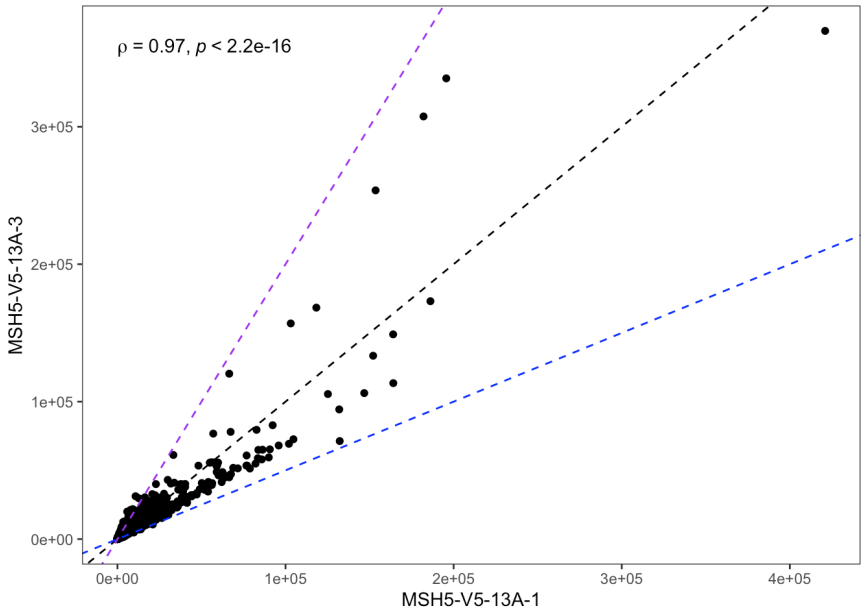

```
ggpairs(as.data.frame(TMMFiltData_non2), diag = list(continuous = "blankDiag"), upper = list(continuous = wrap("cor", size = 2.5))) + theme_bw() + theme(panel.background = element_blank(), panel.grid = element_blank()) + theme(text = element_text(size = 4))
```

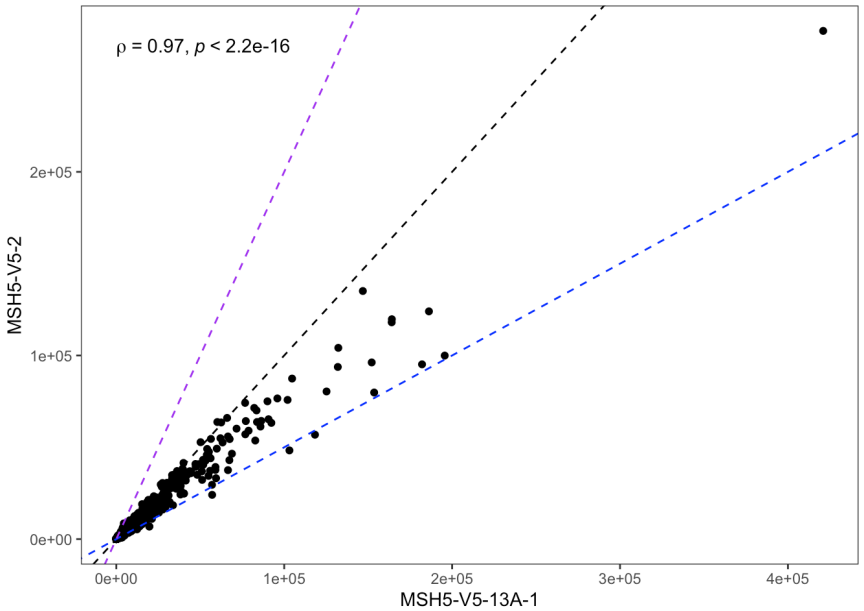


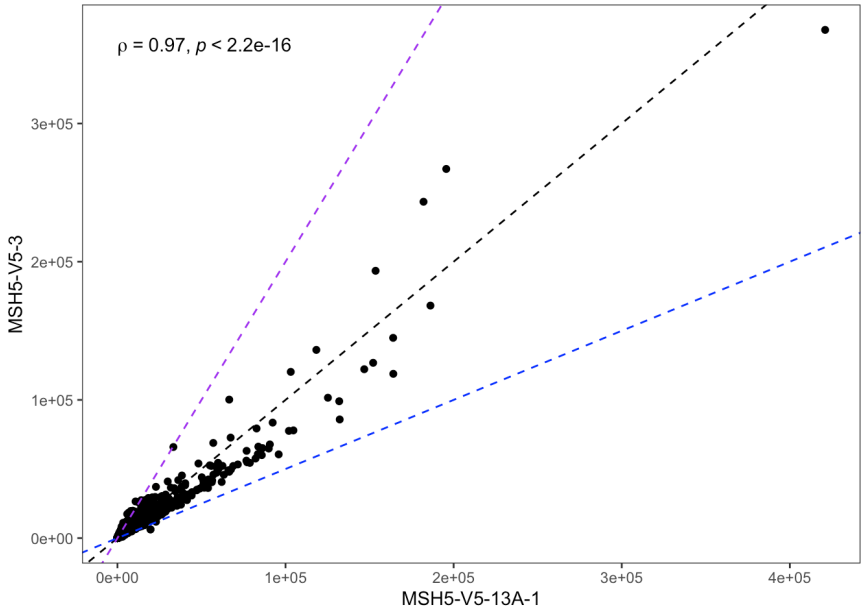

```
combos <- combn(metadf_non2$samples, 2)
for (i in 1:dim(combos)[2]){
  combo1 <- combos[1,i]
  combo2 <- combos[2,i]
  toplot <- as.data.frame(TMMFiltData_non2)[,combos[1:2,i]] %>% `colnames<-`(c("sample1", "sample2"))
  plot <- ggplot(toplot, aes(x=sample1, y = sample2)) + geom_point() + stat_cor(method="spearman", cor.coef.name = "rho" ) + ylab(combo2) + xlab(combo1) + geom_abline(intercept=0, slope=1, linetype='dashed', color='black') + geom_abline(intercept = 0, slope = 0.5, linetype='dashed', color='blue') + geom_abline(intercept=0, slope=2, linetype='dashed', color='purple') + theme_bw() + theme(panel.background = element_blank(), panel.grid = element_blank()) + scale_color_manual(values = c('black', 'blue', 'purple'), labels = c("x=y", 'y = x/2', 'y = 2x'))
  print(plot)
}
```

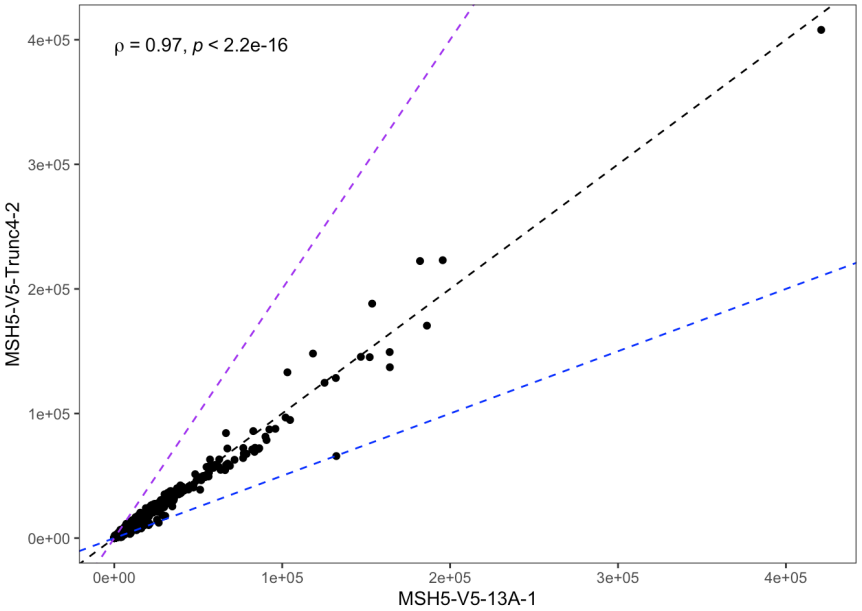

ρ = 0.97, p < 2.2e-16
MSH5-V5-13A-2
MSH5-V5-13A-1
0
50000
100000
150000
200000
0e+00
1e+05
2e+05
3e+05
4e+05

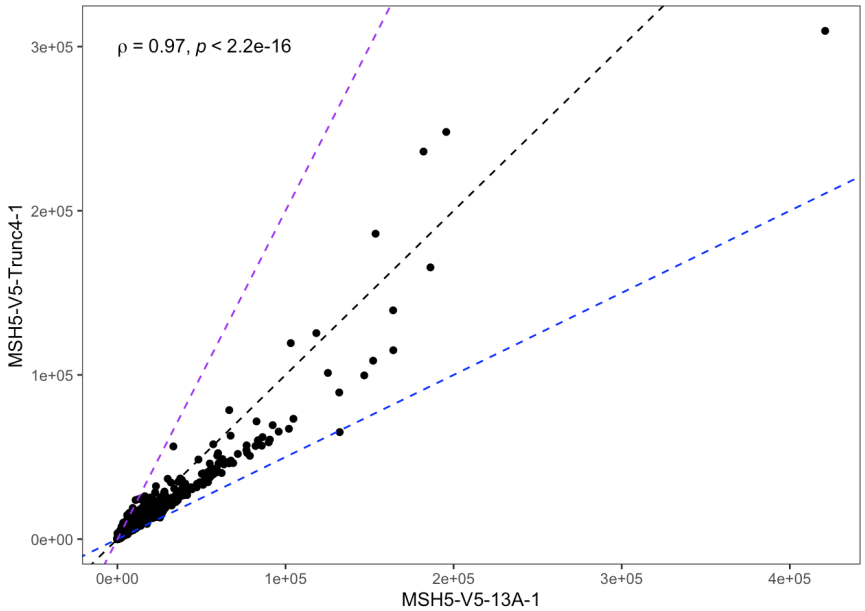

ρ = 0.97, p < 2.2e-16
MSH5-V5-1
MSH5-V5-13A-1
0
50000
100000
150000
200000
250000
0e+00
1e+05
2e+05
3e+05
4e+05

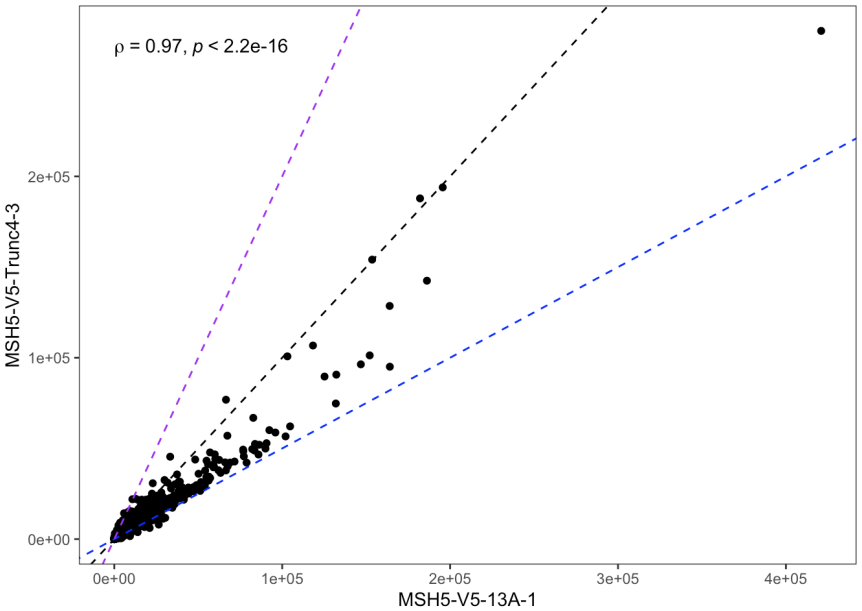

ρ = 0.97, p < 2.2e-16
MSH5-V5-13A-3
MSH5-V5-13A-1
0e+00
1e+05
2e+05
3e+05
0e+00
1e+05
2e+05
3e+05
4e+05

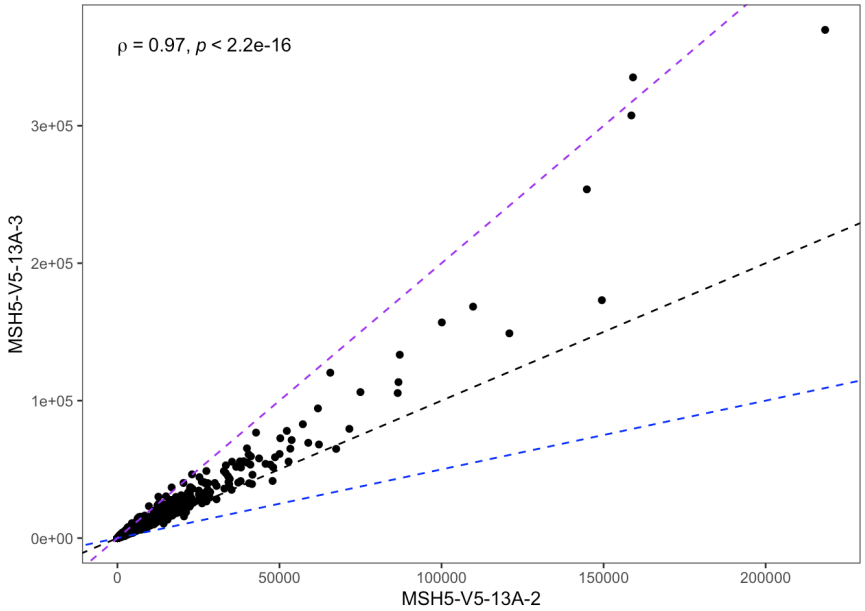

ρ = 0.97, p < 2.2e-16
MSH5-V5-2
MSH5-V5-13A-1
0e+00
1e+05
2e+05
0e+00
1e+05
2e+05
3e+05
4e+05

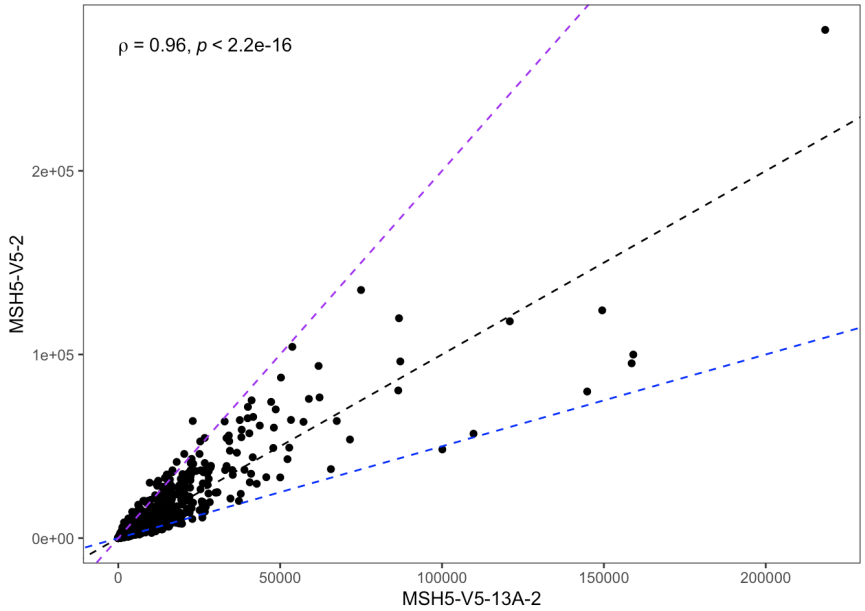

ρ = 0.97, p < 2.2e-16
MSH5-V5-3
MSH5-V5-13A-1
0e+00
1e+05
2e+05
3e+05
4e+05

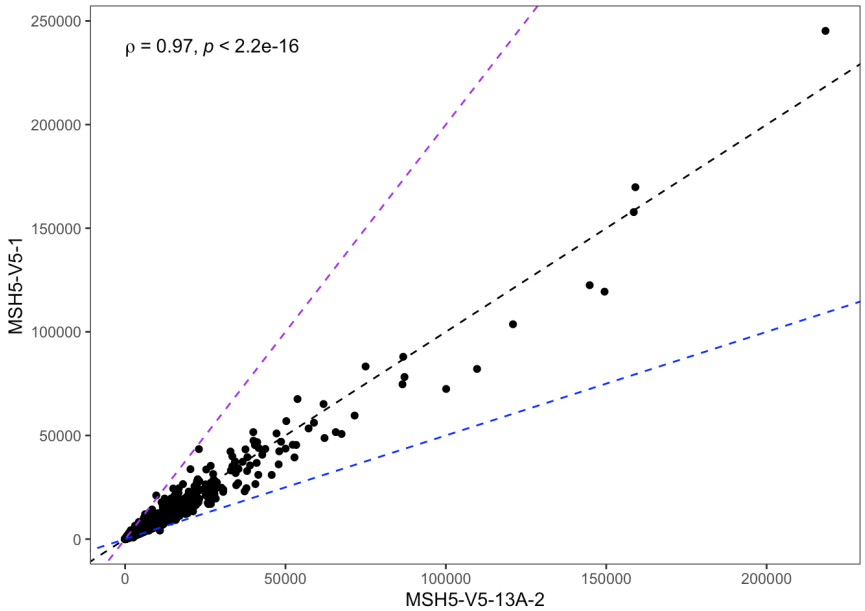

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-2
MSH5-V5-13A-1
0e+00
1e+05
2e+05
3e+05
4e+05

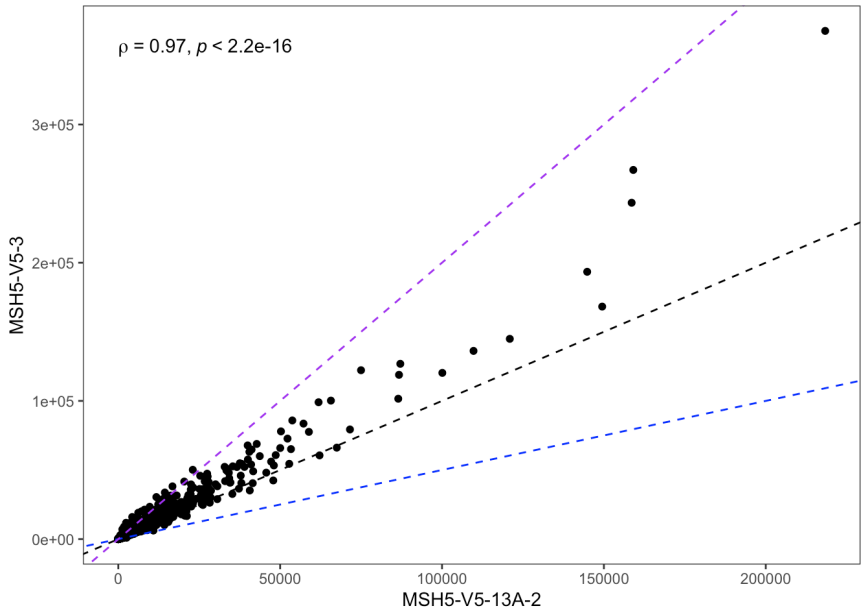

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-1
MSH5-V5-13A-1
0e+00
1e+05
2e+05
3e+05
4e+05

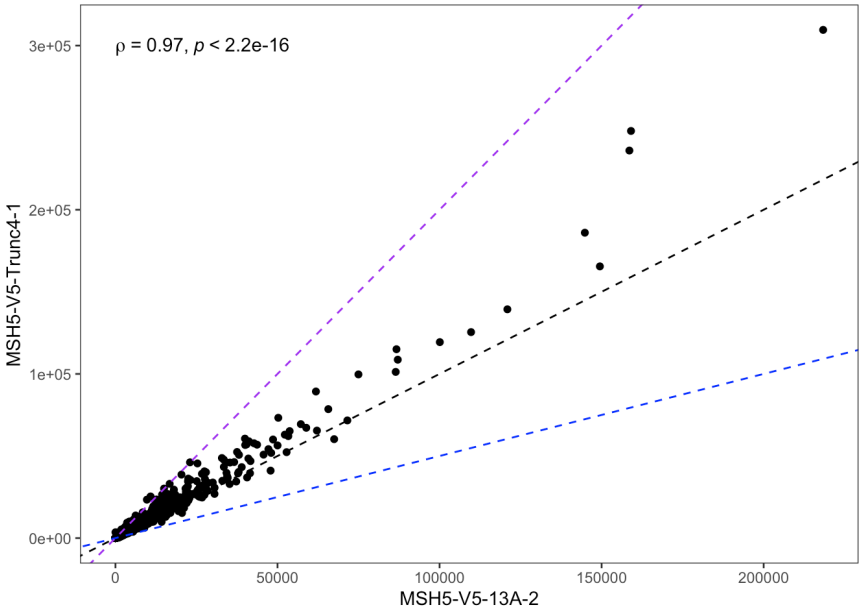

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-3
MSH5-V5-13A-1
0e+00
1e+05
2e+05
3e+05
4e+05

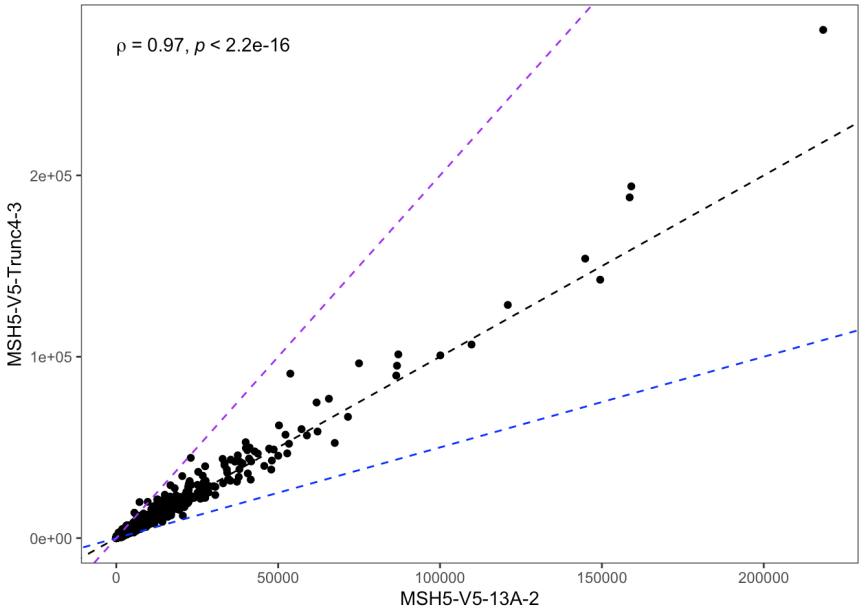

ρ = 0.97, p < 2.2e-16
MSH5-V5-13A-3
MSH5-V5-13A-2
0e+00
1e+05
2e+05
3e+05
0
50000
100000
150000
200000

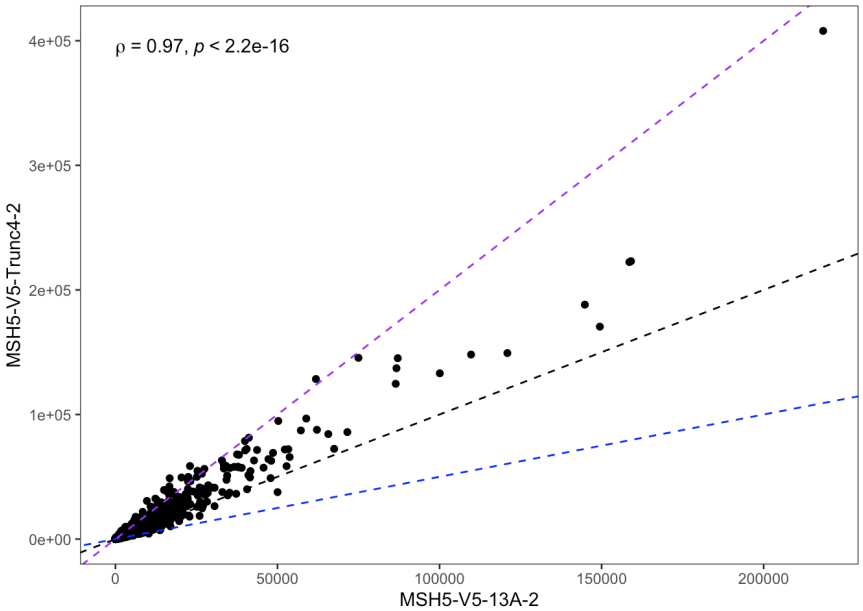

ρ = 0.96, p < 2.2e-16
MSH5-V5-2
MSH5-V5-13A-2
0e+00
1e+05
2e+05
0
50000
100000
150000
200000

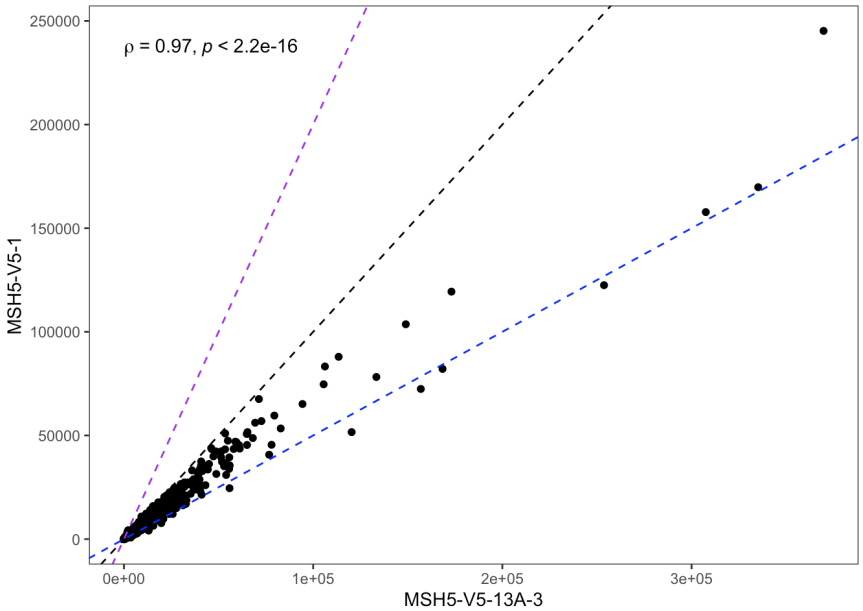

ρ = 0.97, p < 2.2e-16
MSH5-V5-1
MSH5-V5-13A-2
0
50000
100000
150000
200000
250000

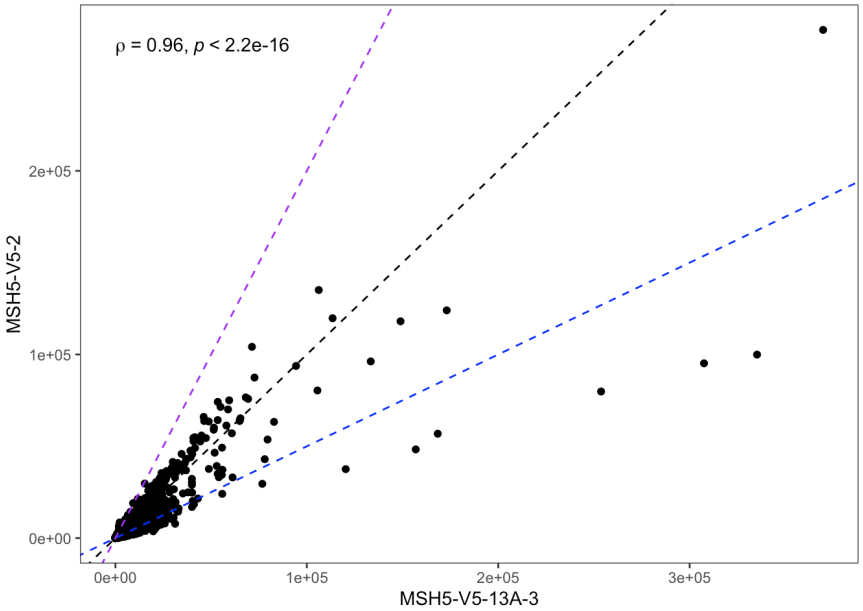

ρ = 0.97, p < 2.2e-16
MSH5-V5-3
MSH5-V5-13A-2
0e+00
1e+05
2e+05
3e+05
0
50000
100000
150000
200000

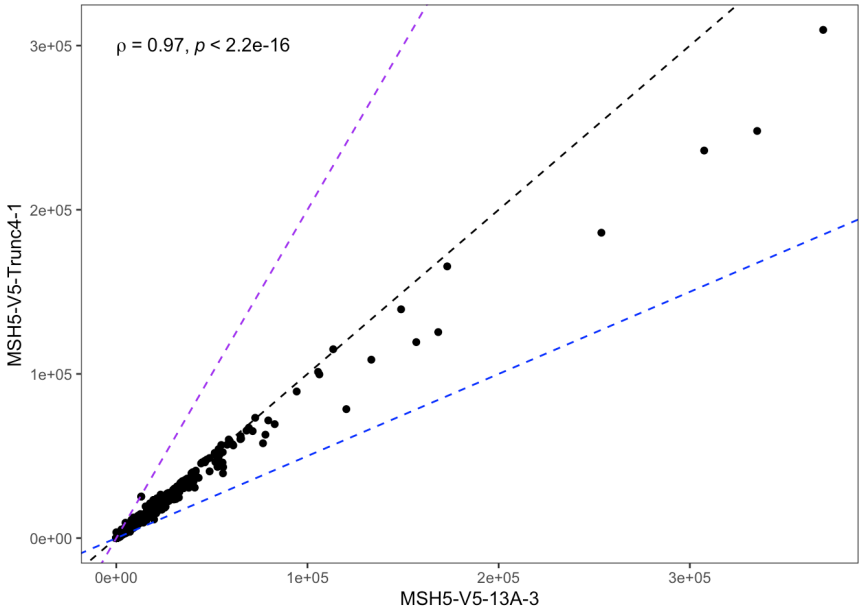

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-1
MSH5-V5-13A-2
0e+00
1e+05
2e+05
3e+05
0
50000
100000
150000
200000

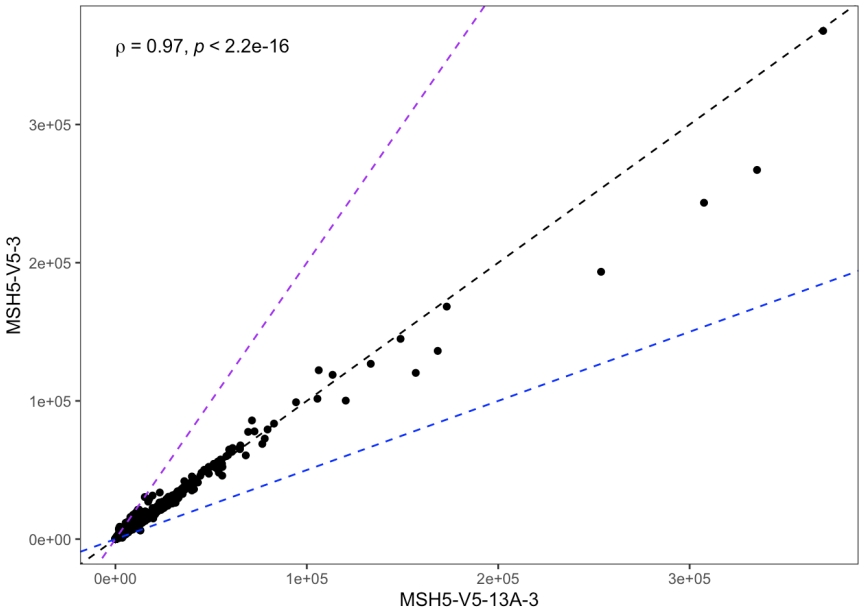

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-3
MSH5-V5-13A-2
0e+00
1e+05
2e+05
0
50000
100000
150000
200000

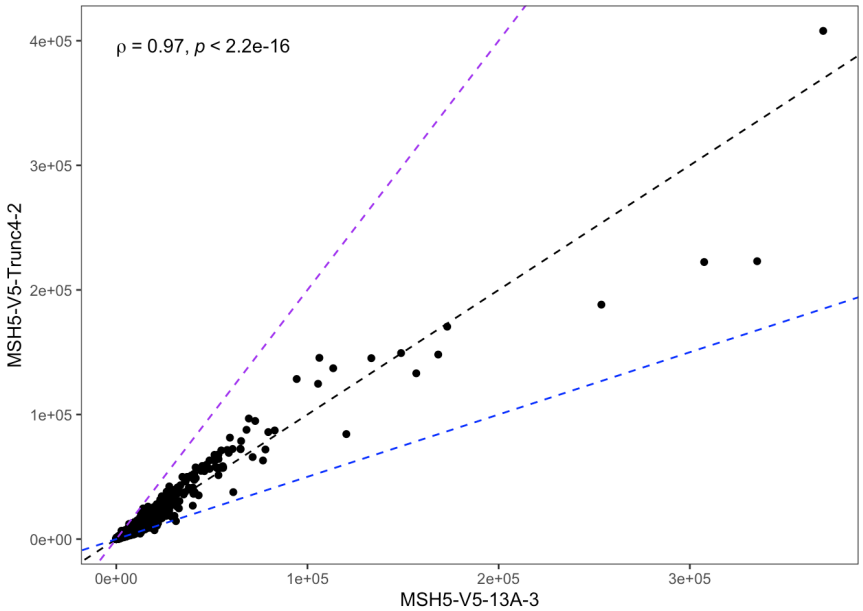

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-2
MSH5-V5-13A-2
0e+00
1e+05
2e+05
3e+05
4e+05
0
50000
100000
150000
200000

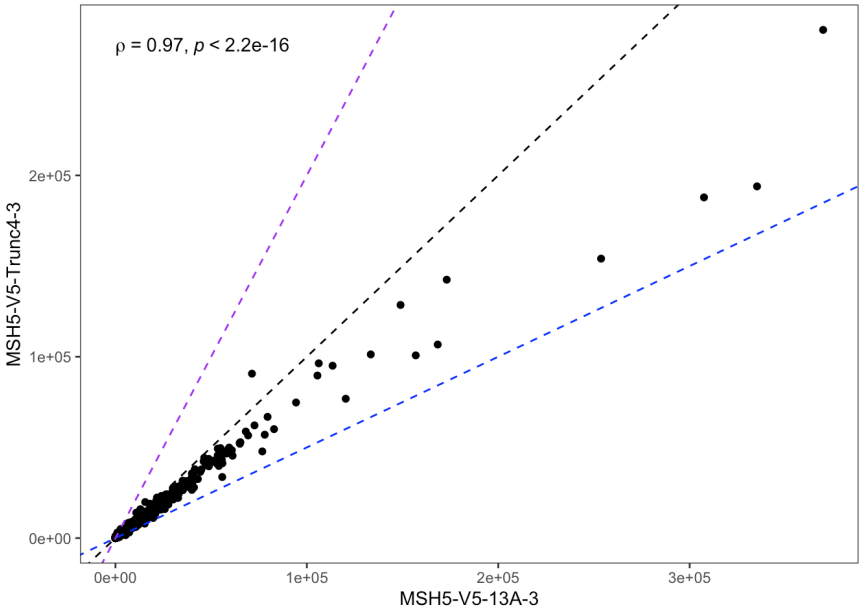

ρ = 0.97, p < 2.2e-16
MSH5-V5-1
MSH5-V5-13A-3
0
50000
100000
150000
200000
250000
0e+00
1e+05
2e+05
3e+05

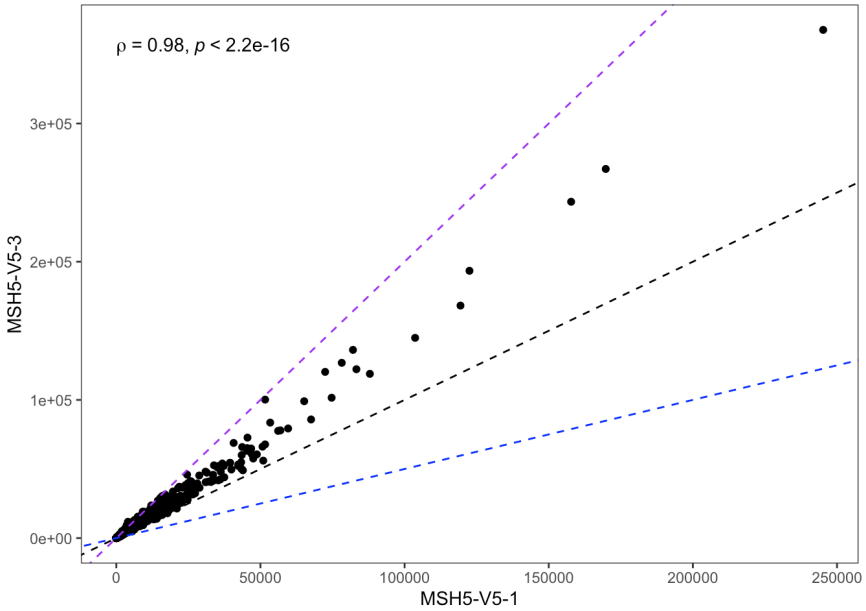

ρ = 0.96, p < 2.2e-16
MSH5-V5-2
MSH5-V5-13A-3
0e+00
1e+05
2e+05
3e+05

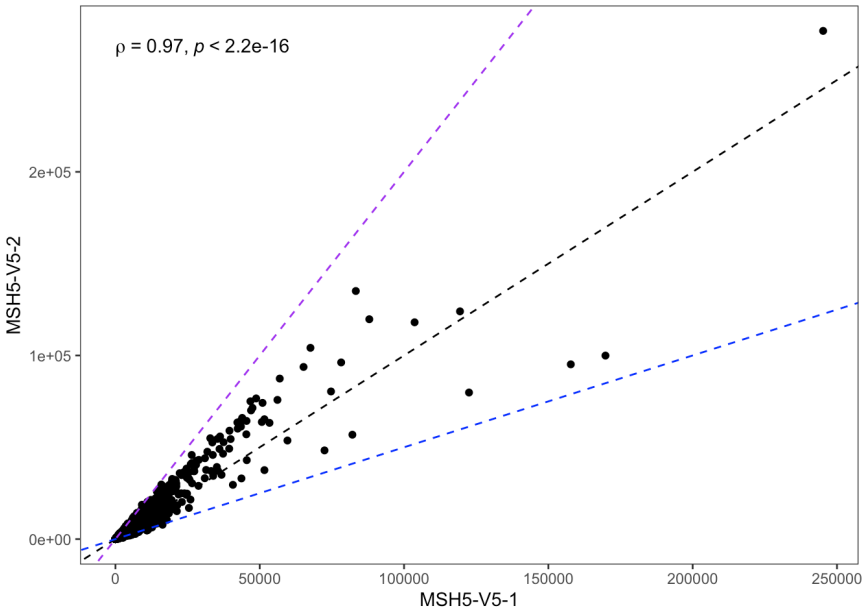

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-1
MSH5-V5-13A-3
0e+00
1e+05
2e+05
3e+05

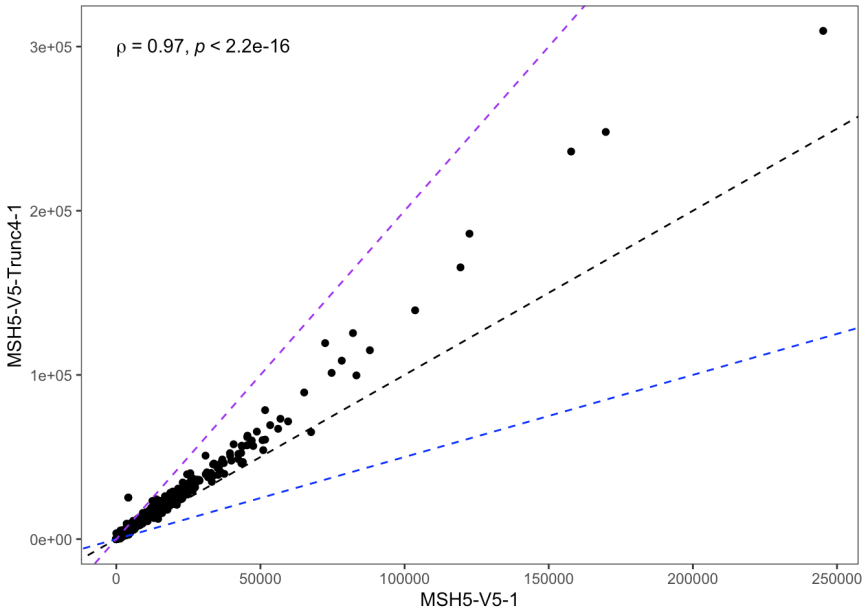

ρ = 0.97, p < 2.2e-16
MSH5-V5-3
MSH5-V5-13A-3
0e+00
1e+05
2e+05
3e+05

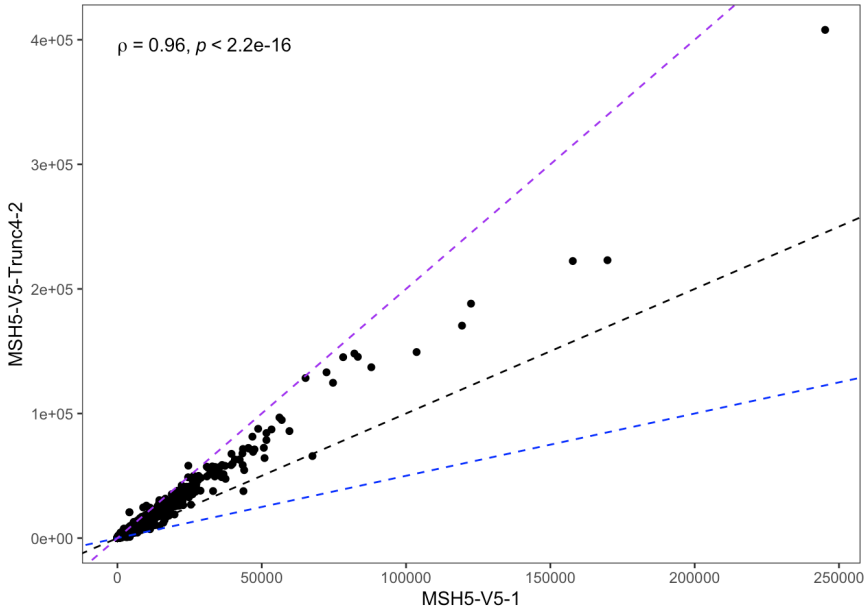

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-2
MSH5-V5-13A-3
0e+00
1e+05
2e+05
3e+05
4e+05

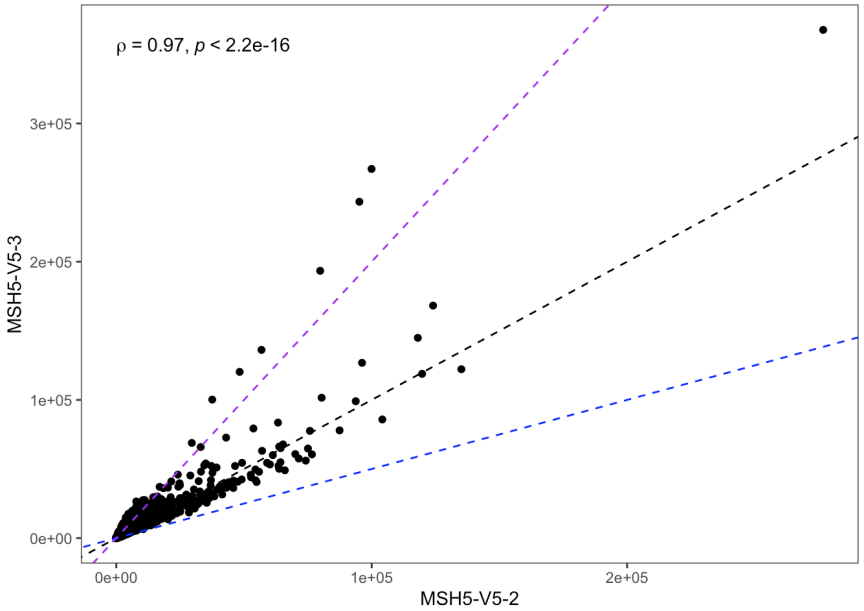

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-3
MSH5-V5-13A-3
0e+00
1e+05
2e+05
3e+05

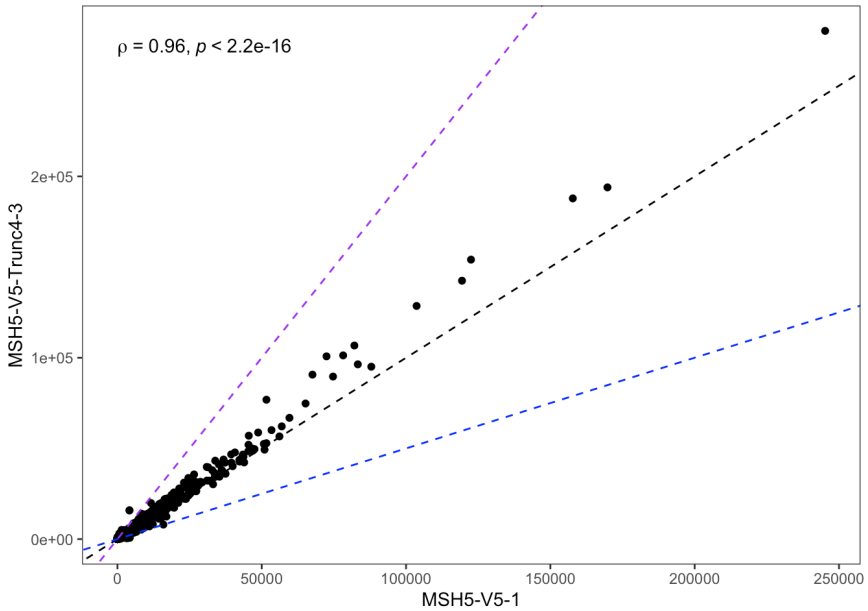

ρ = 0.98, p < 2.2e-16
MSH5-V5-3
MSH5-V5-1
0e+00
1e+05
2e+05
3e+05
0
50000
100000
150000
200000
250000

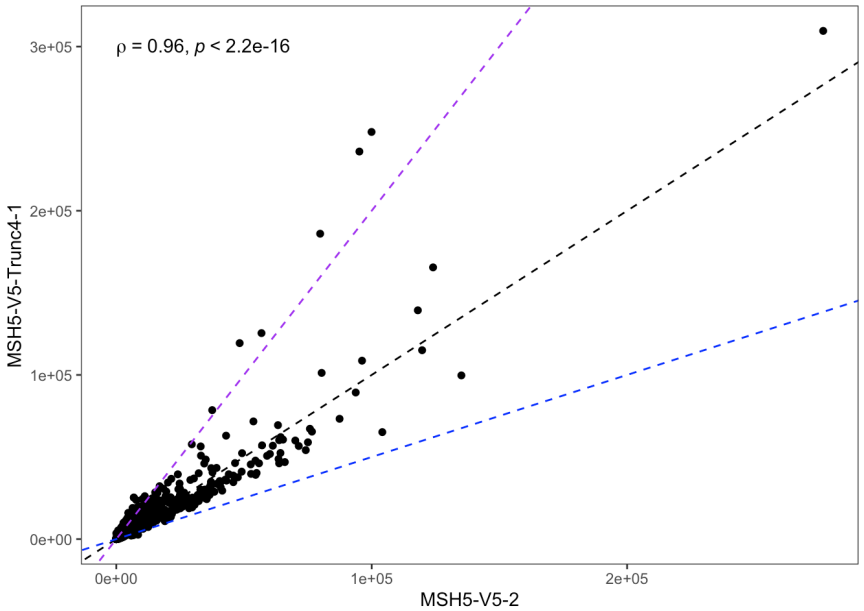

ρ = 0.97, p < 2.2e-16
MSH5-V5-2
MSH5-V5-1
0e+00
1e+05
2e+05
0
50000
100000
150000
200000
250000

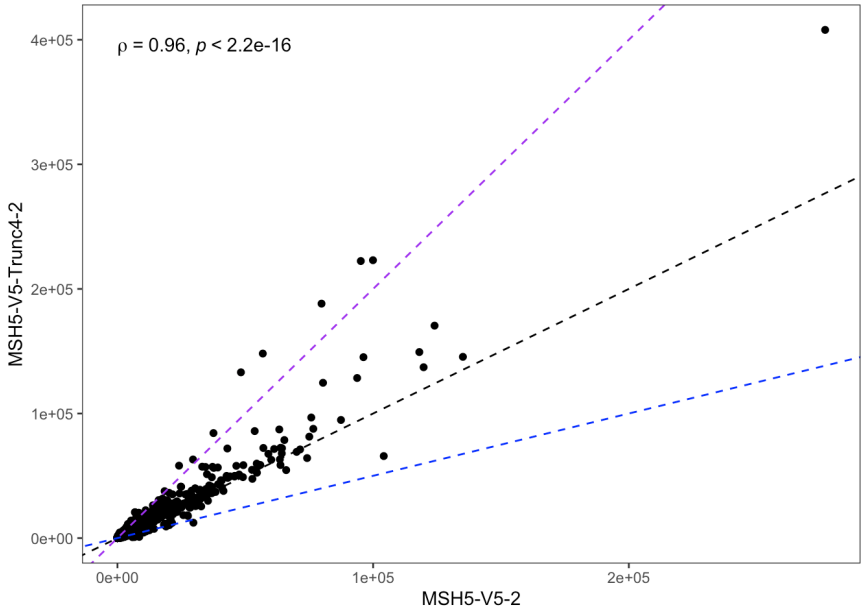

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-1
MSH5-V5-1
0e+00
1e+05
2e+05
3e+05
0
50000
100000
150000
200000
250000

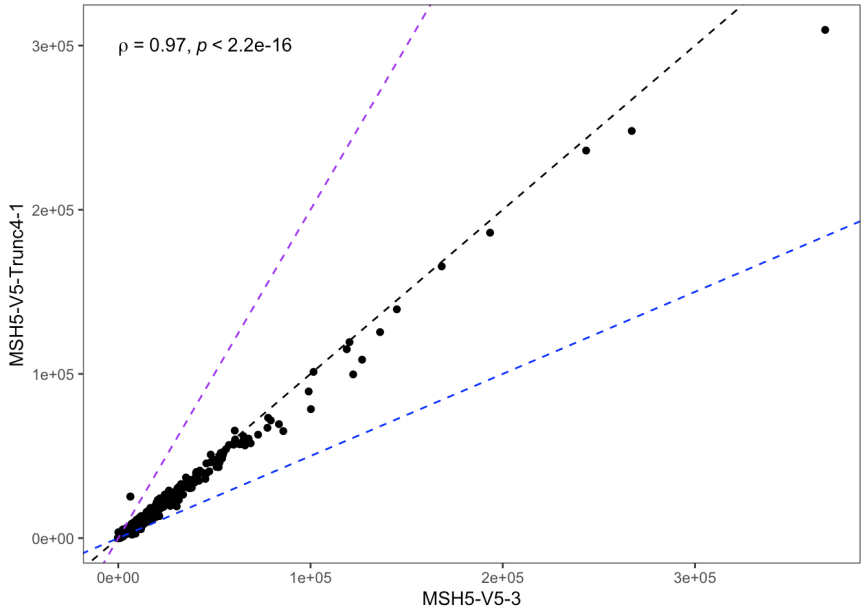

ρ = 0.96, p < 2.2e-16
MSH5-V5-Trunc4-2
MSH5-V5-1
4e+05
3e+05
2e+05
1e+05
0e+00
0
50000
100000
150000
200000
250000

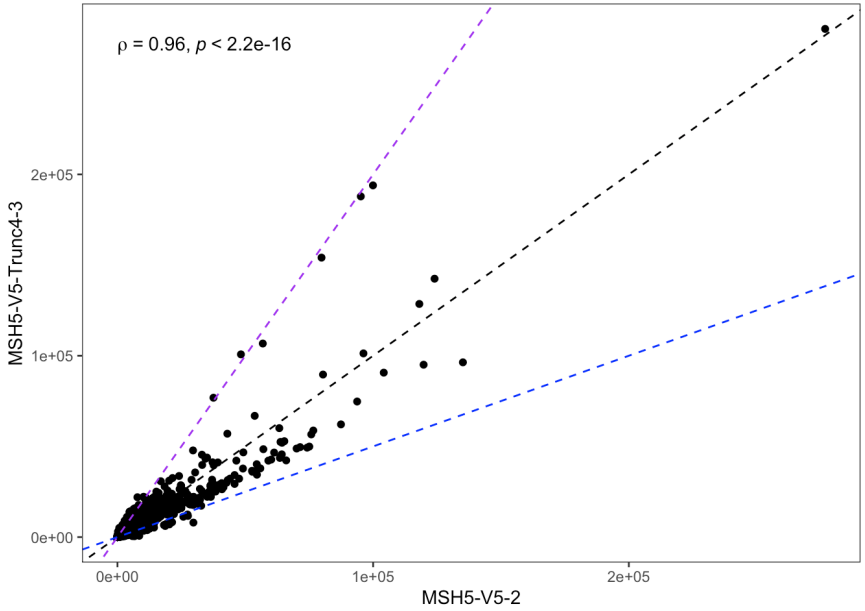

ρ = 0.97, p < 2.2e-16
MSH5-V5-3
MSH5-V5-2
3e+05
2e+05
1e+05
0e+00
0e+00
1e+05
2e+05

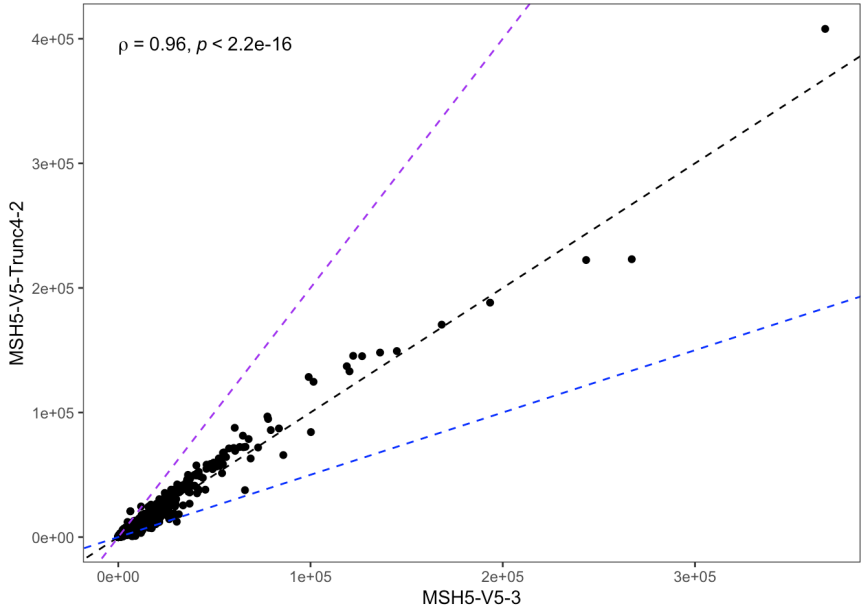

ρ = 0.96, p < 2.2e-16
MSH5-V5-Trunc4-3
MSH5-V5-1
2e+05
1e+05
0e+00
0
50000
100000
150000
200000
250000

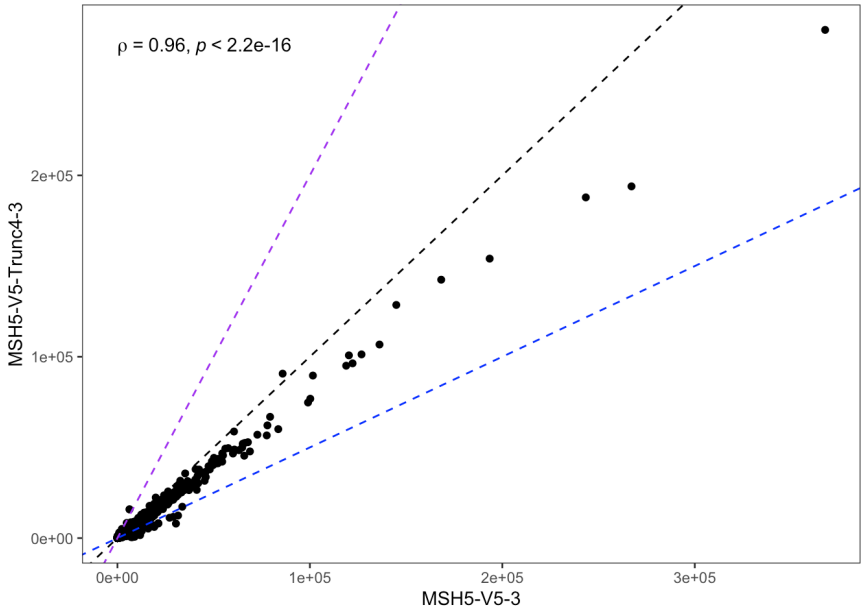

ρ = 0.96, p < 2.2e-16
MSH5-V5-Trunc4-1
MSH5-V5-2
3e+05
2e+05
1e+05
0e+00
0e+00
1e+05
2e+05

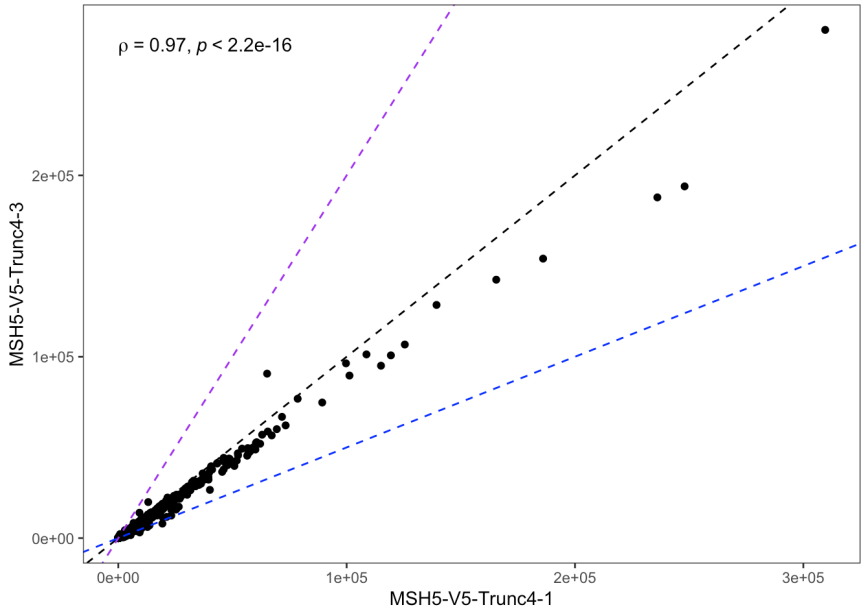

ρ = 0.96, p < 2.2e-16
MSH5-V5-Trunc4-2
MSH5-V5-2
0e+00
1e+05
2e+05
3e+05
4e+05

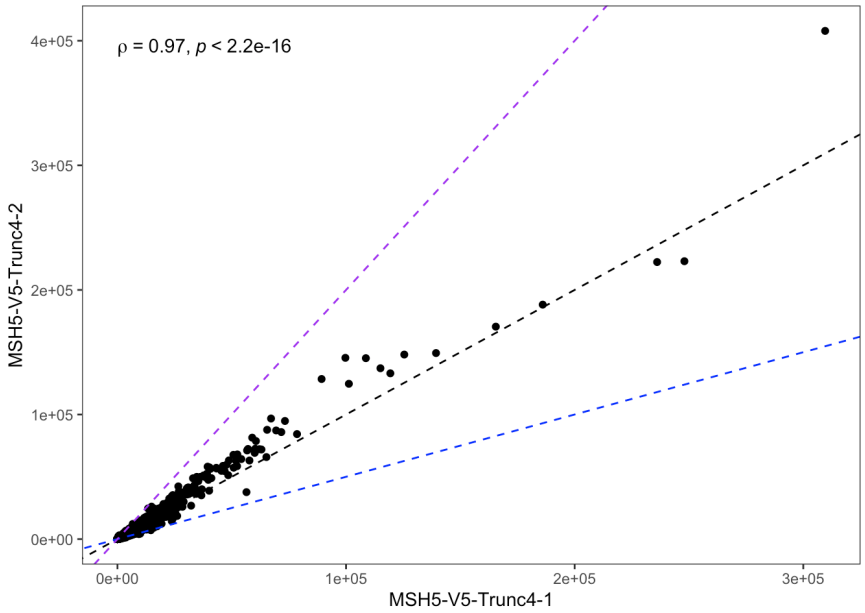

ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-1
MSH5-V5-3
0e+00
1e+05
2e+05
3e+05

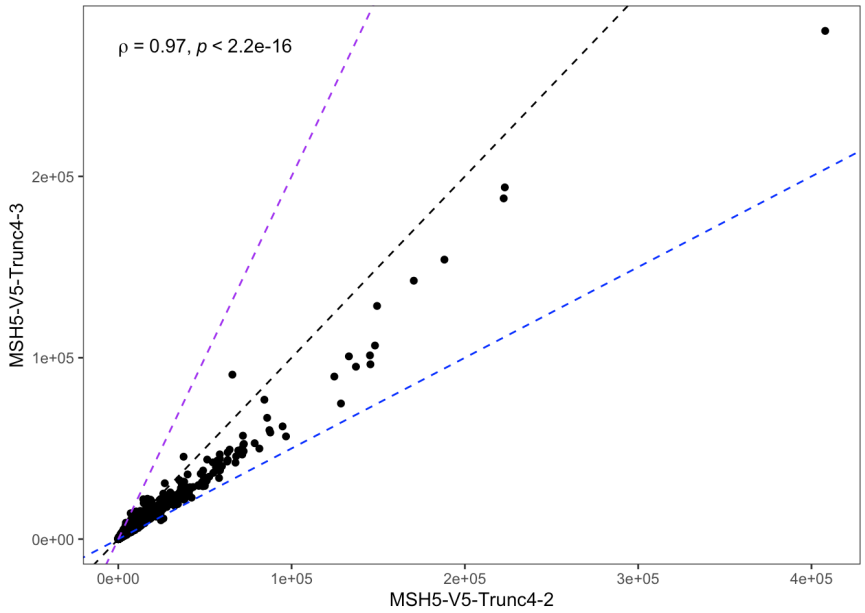

ρ = 0.96, p < 2.2e-16
MSH5-V5-Trunc4-3
MSH5-V5-2
0e+00
1e+05
2e+05


ρ = 0.96, p < 2.2e-16
MSH5-V5-Trunc4-2
MSH5-V5-3
0e+00
1e+05
2e+05
3e+05
4e+05


ρ = 0.96, p < 2.2e-16
MSH5-V5-Trunc4-3
MSH5-V5-3
0e+00
1e+05
2e+05
3e+05


ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-3
MSH5-V5-Trunc4-1
0e+00
1e+05
2e+05
3e+05


ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-2
MSH5-V5-Trunc4-1
0e+00
1e+05
2e+05
3e+05
4e+05


ρ = 0.97, p < 2.2e-16
MSH5-V5-Trunc4-3
MSH5-V5-Trunc4-2
0e+00
1e+05
2e+05
3e+05
4e+05