Pre Post Titration Mystery

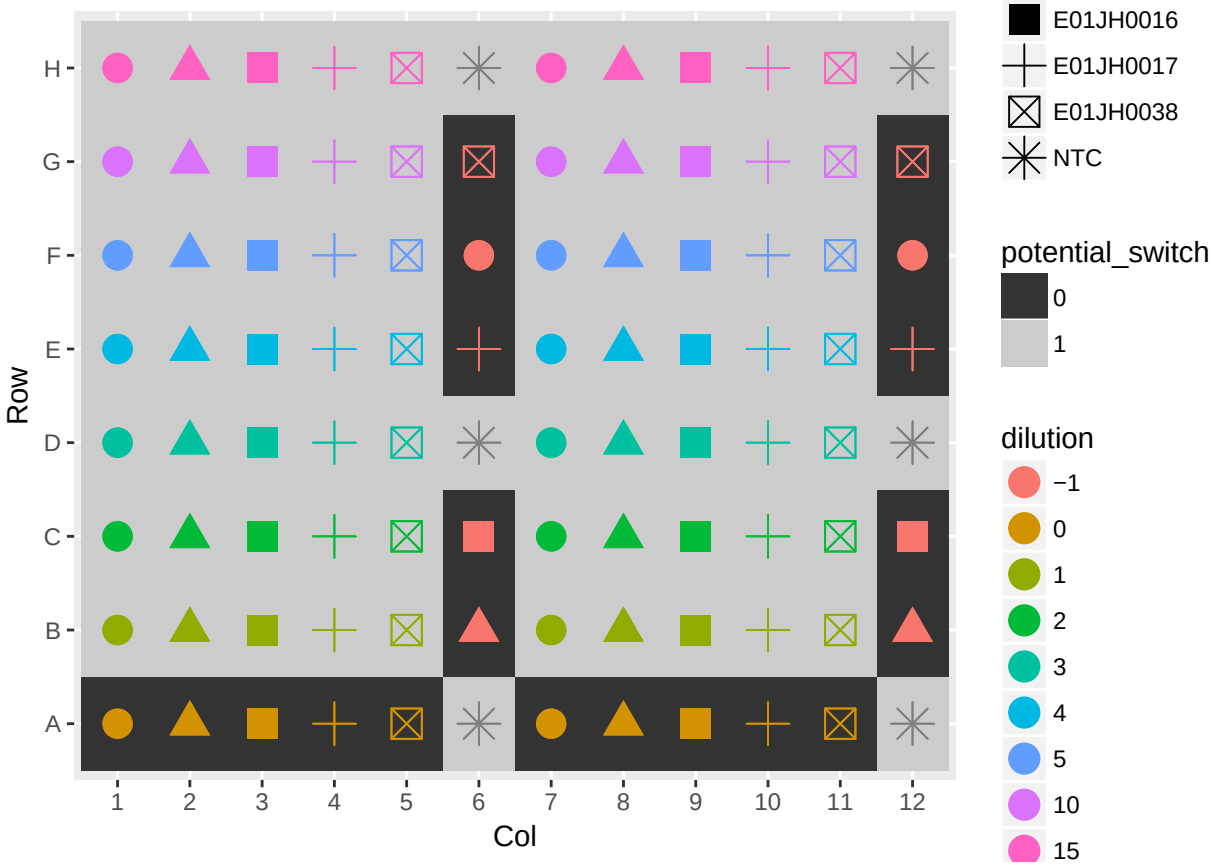
Nate Olson

2017-02-21

Objective

Determine which wells/ samples in the PCR plate correspond to the unmixed pre and post samples. Based on the experimental design, the samples with dilution factors of 0 are unmixed post-treatment and -1 are unmixed pre-treatment. However, preliminary analysis indicates that the unmixed samples were potentially switched, where the unmixed pre and post-treatment samples were mixed relative to the PCR plate layout specified in the experimental design (black highlighted areas in PCR plate diagram below.

```
sampleSheet %>% filter(barcode_lab == "JHU", seq_lab == "JHU") %>%
      mutate(Row = str_sub(pos, 1,1), Col = str_sub(pos,2,4),
             dilution = if_else(titration == 20, -1, titration),
             dilution = factor(dilution),
             Col = as.numeric(Col) %>% factor(),
             potential_switch = if_else(titration %in% c(0,20), "0", "1")) %>%
      ggplot() + geom_raster(aes(x = Col, y = Row, fill = potential_switch)) +
            geom_point(aes(x = Col, y = Row,
                           shape = biosample_id, color = dilution), size = 5) +
      scale_fill_grey()
```



Four lines of evidence supporting that the samples were switched.

1. ERCC Spike-in results: the slope of the regression between Ct and titration factors was consistent with the expected slope of -1 for the ERCC control plasmid spiked into post-treatment samples.
2. The overall the normalized root mean squared error was lower for the count values across normalization methods assuming the samples were switched.
3. For features that present in all four pre-treatment samples (based on the experimental design) the observed count values decrease with increasing titration factor, the expected trend for post-treatment specific features. The expectation is that the count values will decrease with titration for post-treatment specific features.
4. The based on the experimental design sample lables the titration factor -1 samples clustered closest to titration factor 1 based on ordination analysis of titration factors -1, 0, 1, and 15. The expectation is that the unmixed post-treatment samples will cluster closest to the titration factor 1 samples, therefore titration factor -1 samples are most likely the unmixed post-treatment samples.

ERCC Spike-in results

Based on qPCR results from `titration_validation.pdf` the ERCC spiked into the unmixed post-treatment samples had the highest concentration (lowest Ct values) for titration factor 0. Therefore, titration factor 0 is the unmixed post-treatment sample. Note that the sample used in the qPCR assays and 16S PCR are in individual tubes and not in 96 well plates and different sample ids were used for qPCR. It is not unreasonable for the samples to be correctly labeled for the qPCR assays but switched for the 16S PCRs.

Expected Count Values

Based on the sample labels defined in the experimental design, the expectd count values should be calculated as follows, with $p = 2^{-}t$ and t is the titration factor.

$$C_{exp} = [C_{post} \times p] + [C_{pre} \times (1 - p)]$$

If the pre- and post-treatment unmixed samples were switched in the 16S PCR plate the expected count values would be calculated as follows.

$$C_{exp} = [C_{pre} \times p] + [C_{post} \times (1 - p)]$$

```
count_df <- readRDS("../data/normalize_count_df.rds")
```

```
pre_count <- count_df %>% filter(dilution == -1) %>%
      dplyr::rename(pre = count) %>% select(-dilution, -samID)
post_count <- count_df %>% filter(dilution == 0) %>%
      dplyr::rename(post = count) %>% select(-dilution, -samID)
pre_post_count <- left_join(pre_count, post_count)
```

```
## Joining, by = c("norm_method", "pipe", "featureIndices", "sampleID", "pcr_rep")
```

```
rm(pre_count, post_count)

count_exp_obs <- count_df %>%
      filter(!(dilution %in% c(0,-1))) %>%
      left_join(pre_post_count) %>%
      mutate(p = 2^(-dilution),
             exp_count_pre_p = post * (1-p) + pre * p,
             exp_count_post_p = pre * (1-p) + post * p)
```

```
## Joining, by = c("norm_method", "pipe", "featureIndices", "sampleID", "pcr_rep")
```

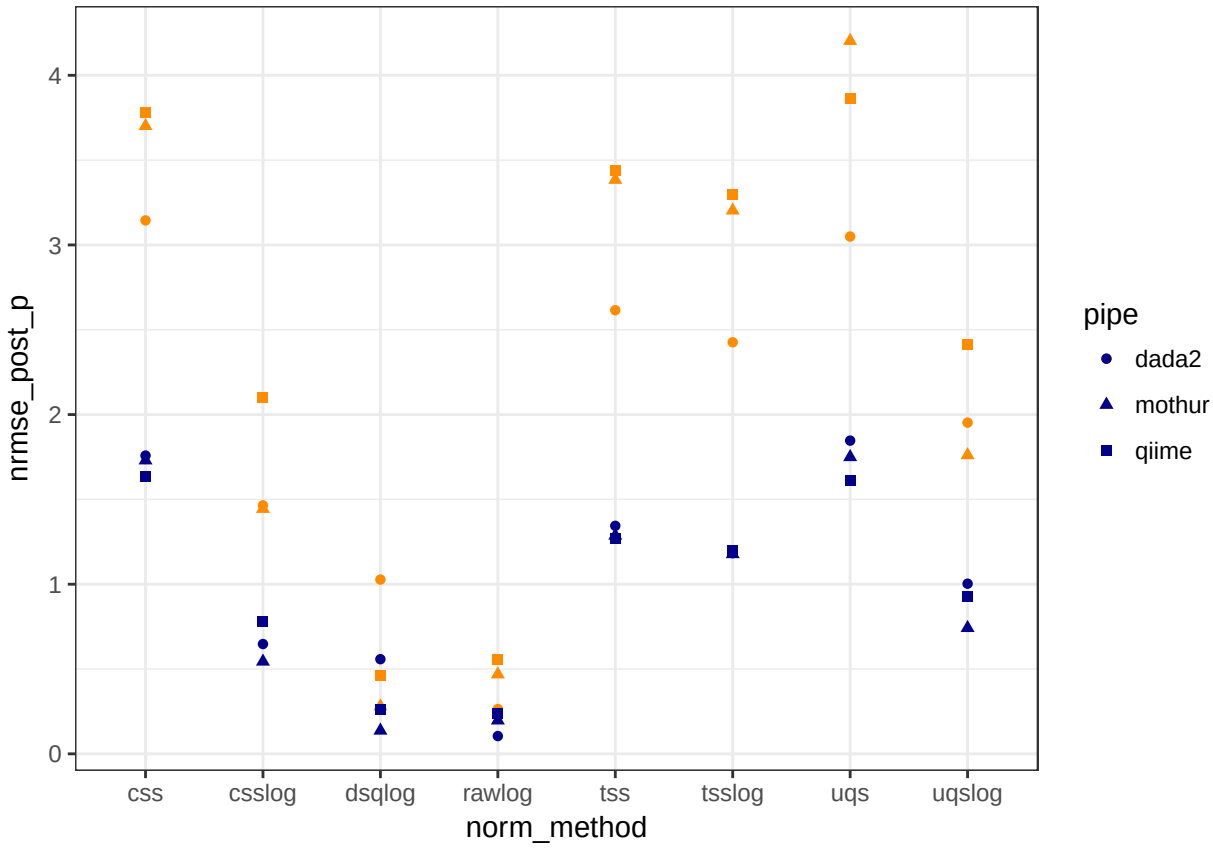
Metrics for evaluating count values

The overall pipeline and normalization method performance was evaluated using root mean squared error (RMSE) and the normalized RMSE (NRMSE) or coefficient of variation of the RMSE.

```
count_rmse <- count_exp_obs %>% filter(sampleID != "NTC") %>%
  filter(pre != 0, post != 0, count != 0) %>%
  mutate(residual_pre_p = (exp_count_pre_p - count)^2,
         residual_post_p = (exp_count_post_p - count)^2) %>%
      group_by(pipe, norm_method) %>%
      summarise(mse_pre_p = mean(residual_pre_p),
                rmse_pre_p = sqrt(mse_pre_p),
                nrmse_pre_p = rmse_pre_p/mean(exp_count_pre_p),
                mse_post_p = mean(residual_post_p),
                rmse_post_p = sqrt(mse_post_p),
                nrmse_post_p = rmse_post_p/mean(exp_count_post_p))
```

Lower normalized RMSE for all normalization methods when $2^{-}t$ is the proportion of pre-treatment sample in the mix.

```
count_rmse %>% filter(norm_method != "dsq", norm_method != "raw") %>%
  ggplot() +
    geom_point(aes(x = norm_method, y = nrmse_post_p, shape = pipe), color = "darkorange") +
    geom_point(aes(x = norm_method, y = nrmse_pre_p, shape = pipe), color = "darkblue") +
  theme_bw()
```



Looking at pre and post specific features

```
feature_specificity_df <- readRDS("../data/feature_specificity_df.rds")
count_df <- readRDS("../data/expected_counts_df.rds")
```

```
feature_anno <- feature_specificity_df %>%
  select(pipe, featureIndices, sampleID, specific_anno) %>%
  mutate(otuID = featureIndices) %>% unique()
```

```
feature_explore <- count_df %>% filter(sampleID != "NTC") %>%
  mutate(featureIndices = as.numeric(featureIndices)) %>%
  left_join(feature_anno)
```

```
## Joining, by = c("pipe", "featureIndices", "sampleID")
```

```
top_pre <- feature_explore %>% filter(specific_anno == "pre_full", norm_method == "rawlog") %>%
  group_by(pipe, norm_method, specific_anno, sampleID, otuID) %>%
  summarise(total_count = sum(count)) %>%
  arrange(desc(total_count)) %>% group_by(pipe, specific_anno, norm_method) %>% top_n(3, total_count)
```

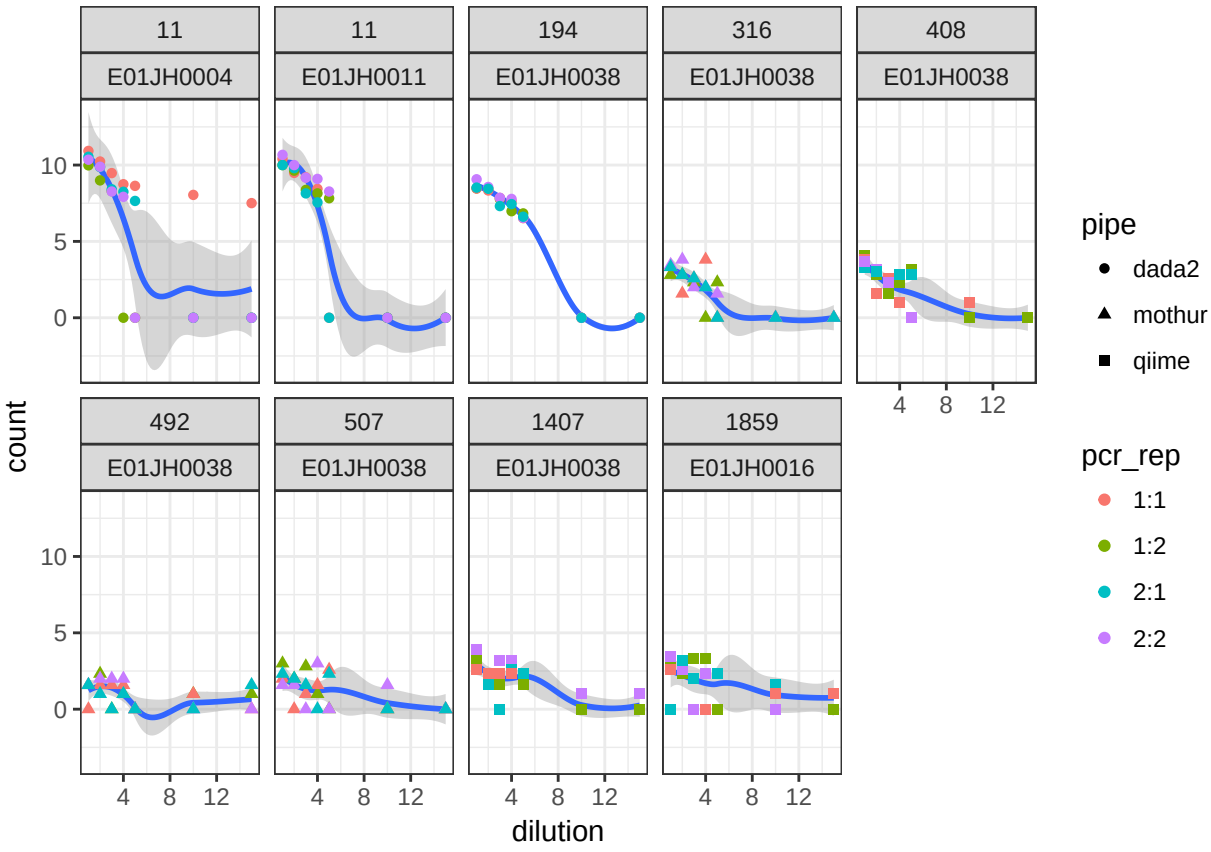
```
top_pre_features <- top_pre %>% left_join(feature_explore)
```

```
## Joining, by = c("pipe", "norm_method", "specific_anno", "sampleID", "otuID")
```

Pre-treatment specific features should increase in abundance with titration factor not decrease!

```
top_pre_features %>%
  ggplot(aes(x = dilution, y = count)) +
      geom_smooth() +
      geom_point(aes(color = pcr_rep, shape = pipe)) +
      facet_wrap(~otuID*sampleID, nrow = 2) + theme_bw()
```

```
## `geom_smooth()` using method = 'loess'
```




```
top_post <- feature_explore %>% filter(specific_anno == "post_full", norm_method == "rawlog") %>%
  group_by(pipe, norm_method, specific_anno, sampleID, otuID) %>%
  summarise(total_count = sum(count)) %>%
  arrange(desc(total_count)) %>% group_by(pipe, specific_anno, norm_method) %>% top_n(3, total_count)
```

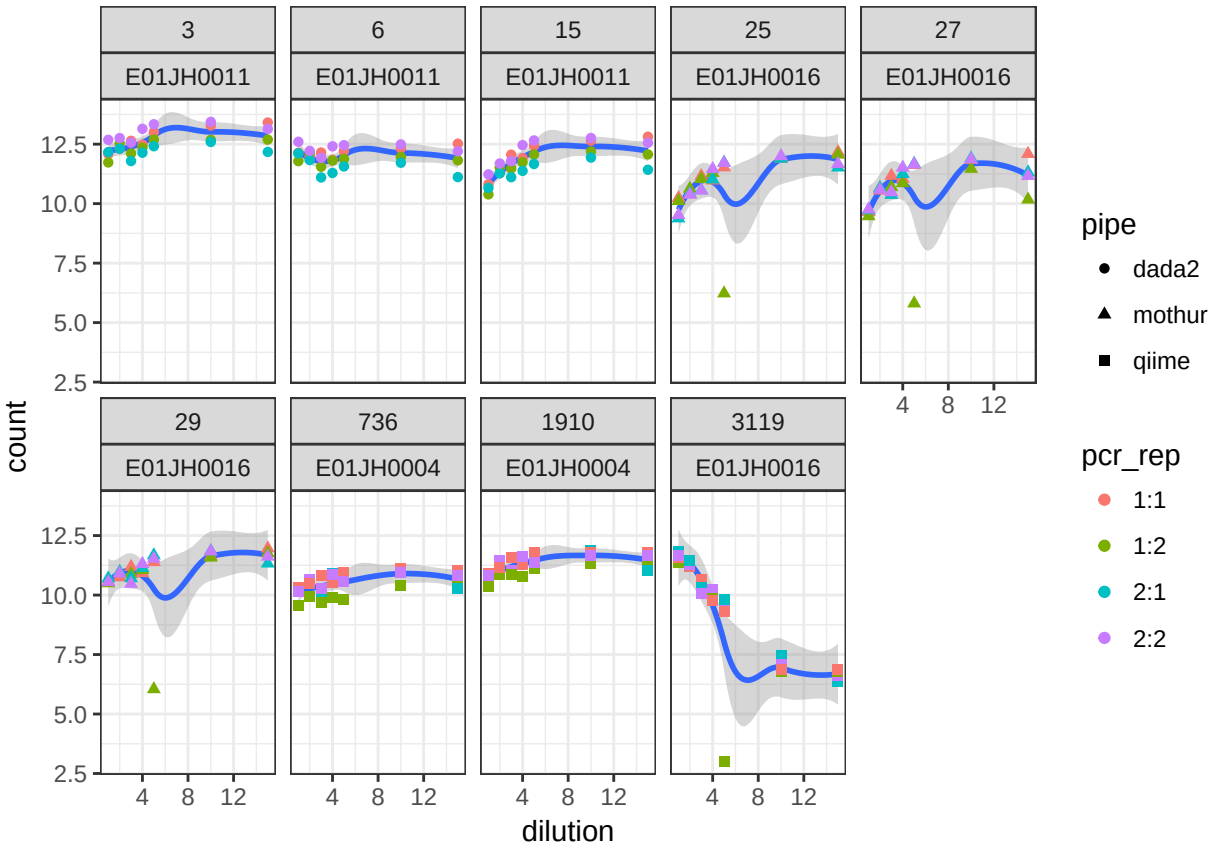
```
top_post_features <- top_post %>% left_join(feature_explore)
```

```
## Joining, by = c("pipe", "norm_method", "specific_anno", "sampleID", "otuID")
```

Post-treatment specific features tend to increase in abundance or remain high, excluding feature 3119 E01JH0016.

```
top_post_features %>%
  ggplot(aes(x = dilution, y = count)) +
      geom_smooth() +
      geom_point(aes(color = pcr_rep, shape = pipe)) +
      facet_wrap(~otuID*sampleID, nrow = 2) + theme_bw()
```

```
## `geom_smooth()` using method = 'loess'
```



Beta-Diversity Cluster Analysis

Based on the experimental design the expectation is that the PCR replicates for titration factors -1 will be closest to 15, and 0 closest to 1, assuming the titration factor -1 is the unmixed pre-treatment samples.

```
ps_files <- list(
      dada2  = "../data/phyloseq_dada2.RDS",
      mothur = "../data/phyloseq_mothur.RDS",
      qiime  = "../data/phyloseq_qiime.RDS"
)
ps <- ps_files %>% map(readRDS)
```

```
sample_set <- sampleSheet %>% filter(barcode_lab == "JHU", seq_lab == "JHU",
                      titration %in% c(20,0,1,15)) %>%
      select(biosample_id, titration, pcr_16S_plate, pos) %>%
      unite(sample_name, pcr_16S_plate, pos, sep = "-")
```

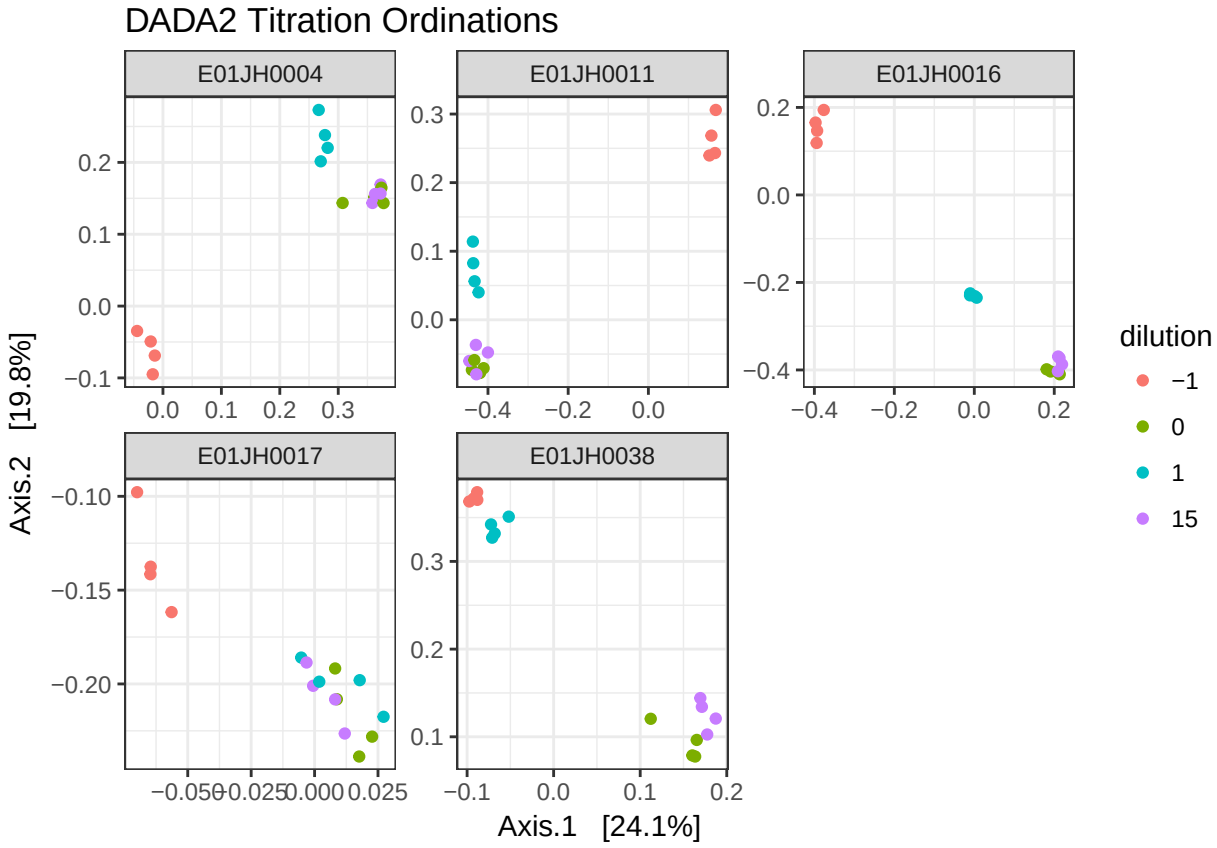
```
subset_ps <- function(ps_obj){

      ## subset by titrations of interest
      ps_subset <- subset_samples(ps_obj, dilution %in% c(-1,0,1,15))
      ps_subset@sam_data$dilution <- factor(ps_subset@sam_data$dilution)

      ps_subset
}
```

```
sample_ids <- sampleSheet$biosample_id %>% unique()
subset_obj <- ps %>% map(subset_ps)
subset_ord <- subset_obj %>% map(ordinate, "PCoA", "bray")
```

```
pl_list <- map2(subset_obj, subset_ord, plot_ordination,
                type = "sample",color = "dilution")
```

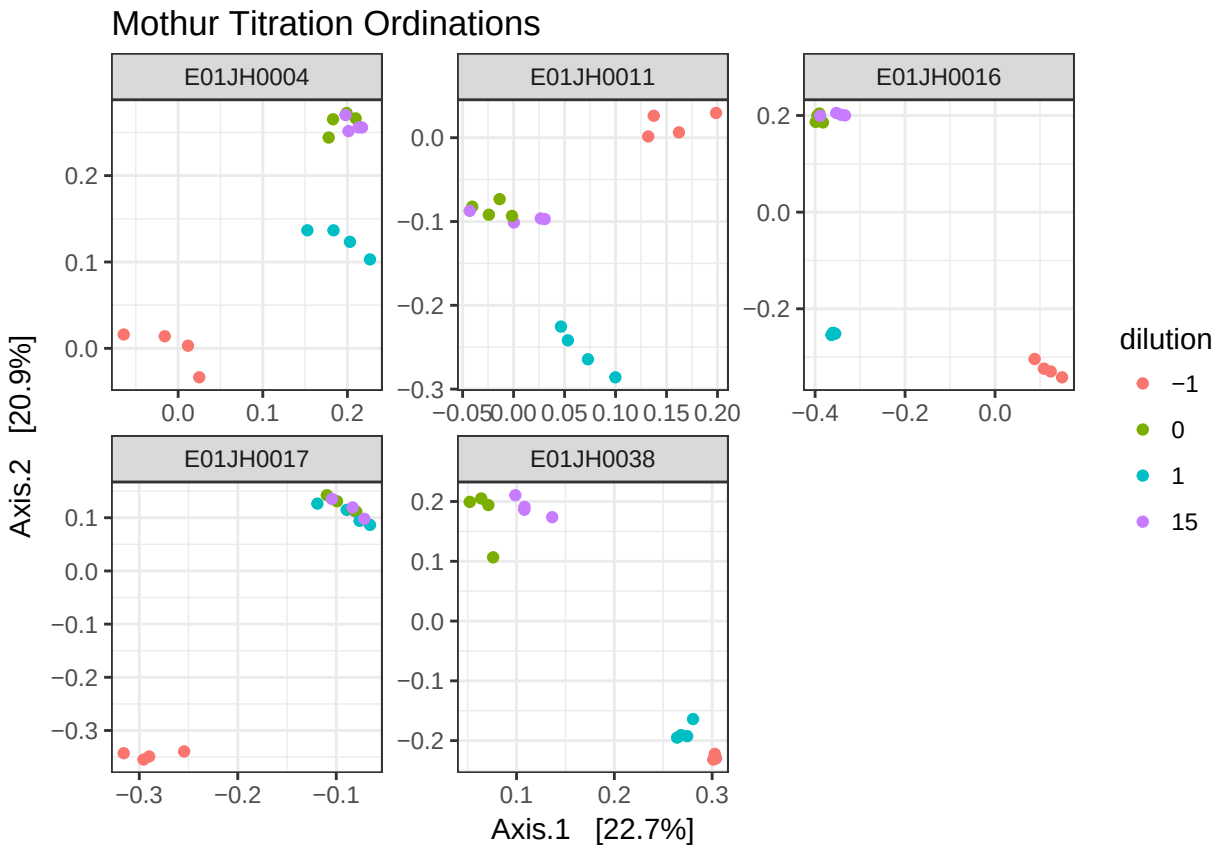
For four of the biological replicates (4, 11, 16 and 38) dilution 1 is closest to dilution -1, supporting the theory that the samples were switched. Dilutions 0, 1, and 15 for biological replicate 17 were tightly clustered which also suggest that the samples were switched, though better separation would better support our theory that the samples are switched.

```
pl_list$dada2 + ggtitle("DADA2 Titration Ordinations") +
      facet_wrap(~sampleID, scales = "free") + theme_bw()
```



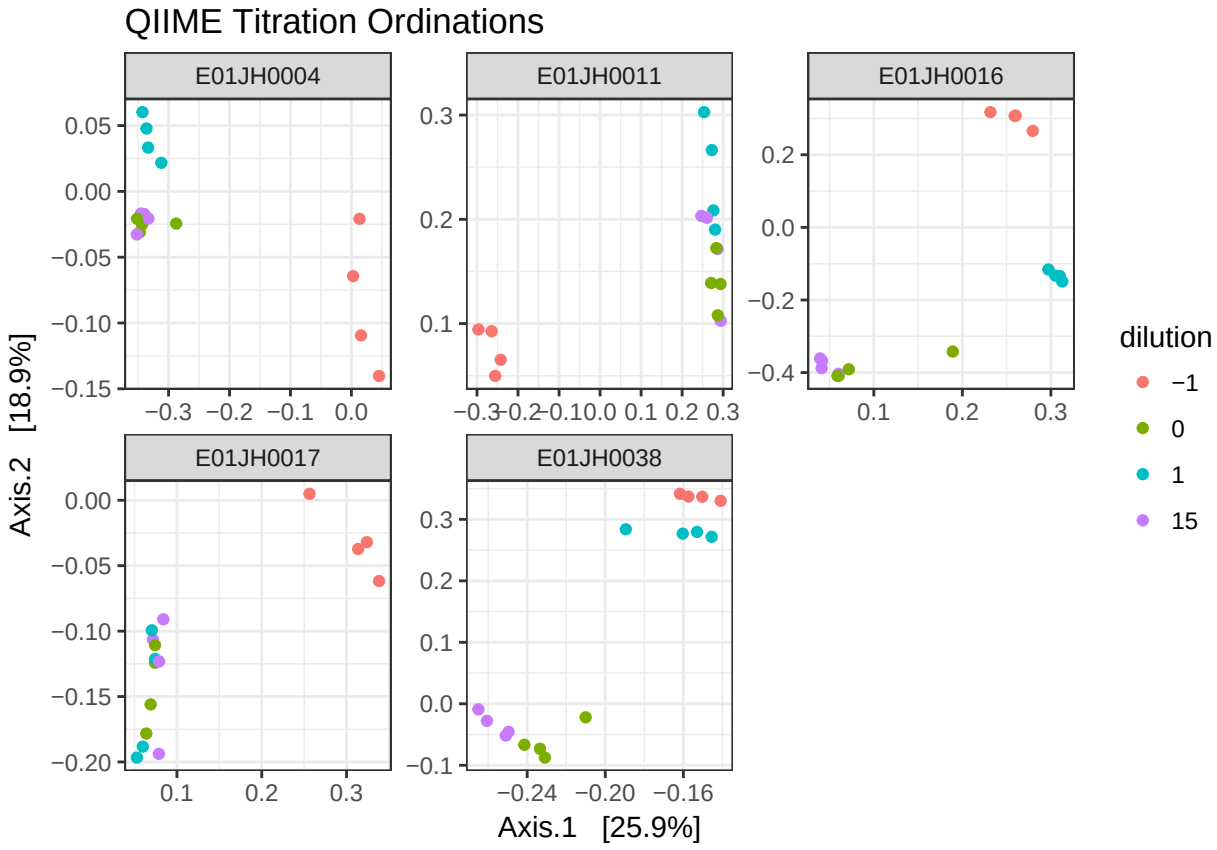
Dilutions 0, 1, and 15 are tightly grouped for biological replicate 17.
For biological replicates 4, 11, 16 and 38, dilution -1 is closest to dilution 1.

```
pl_list$mothur + ggtitle("Mothur Titration Ordinations") +
      facet_wrap(~biosample_id, scales = "free") + theme_bw()
```



Dilutions 0, 1, and, 15 overlap for biological replicate 17. For biological replicates 16, and 38 dilution -1 is closest to 1. Dilutions 0 and 15 are closer to dilution -1 than dilution 1 for biological replicates 4 and 11.

```
pl_list$qiime + ggtitle("QIIME Titration Ordinations") +
      facet_wrap(~sampleID, scales = "free") + theme_bw()
```



Separate Ordination by Sample ID

As dilutions -1, 0, and 1 overlap for some of the ordination plots, will calculate the ordination plots for the biological replicates indvidually to get better separation.

```
subset_ps2 <- function(ps_obj, biorep){
      sams <- sample_data(ps_obj) %>% as_data_frame() %>%
            rownames_to_column(var = "rownames") %>%
            filter(dilution %in% c(-1,0,1,15), sampleID == biorep) %>%
            .$rownames
      ps_subset <- prune_samples(samples = sams, ps_obj)
      ps_subset@sam_data$dilution <- factor(ps_subset@sam_data$dilution)

      ps_subset
}

plot_ord <- function(subset_obj, ord, biorep, pipe){
      plot_ordination(subset_obj, ord, color = "dilution") +
                  ggtitle(paste("Ordination", biorep, pipe)) + theme_bw()
}
```

```
sample_ids <- sampleSheet %>% filter(biosample_id != "NTC") %>%
      .$biosample_id %>% unique()
ps_df <- data_frame(biorep = rep(sample_ids, each = 3)) %>%
      add_column(pipe = rep(names(ps),5)) %>%
      add_column(ps_obj = rep(ps, 5))
```

```r
ps_subset_df <- ps_df %>%
      mutate(subset_obj = map2(.x = ps_obj, .y = biorep, .f = subset_ps2))
```

```
## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object
```

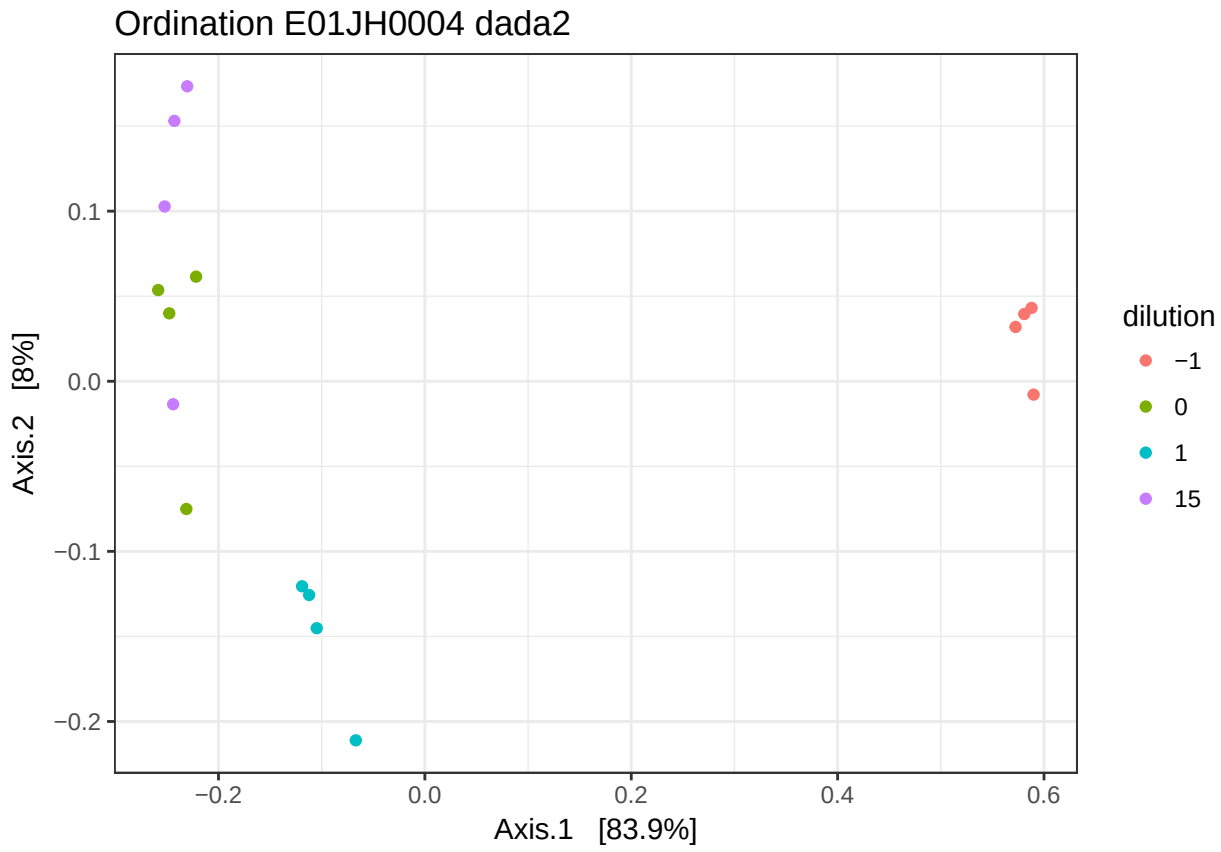
```
## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object

## Warning in class(x) <- c("tbl_df", "tbl", "data.frame"): Setting class(x)
## to multiple strings ("tbl_df", "tbl", ...); result will no longer be an S4
## object
```

```
ps_ord <- ps_subset_df %>%
      mutate(ord = map(subset_obj, ordinate,
                       method = "PCoA",distance = "bray"))
ps_plot <- ps_ord %>%
      mutate(ord_plots = pmap(list(subset_obj, ord, biorep, pipe), plot_ord))
```

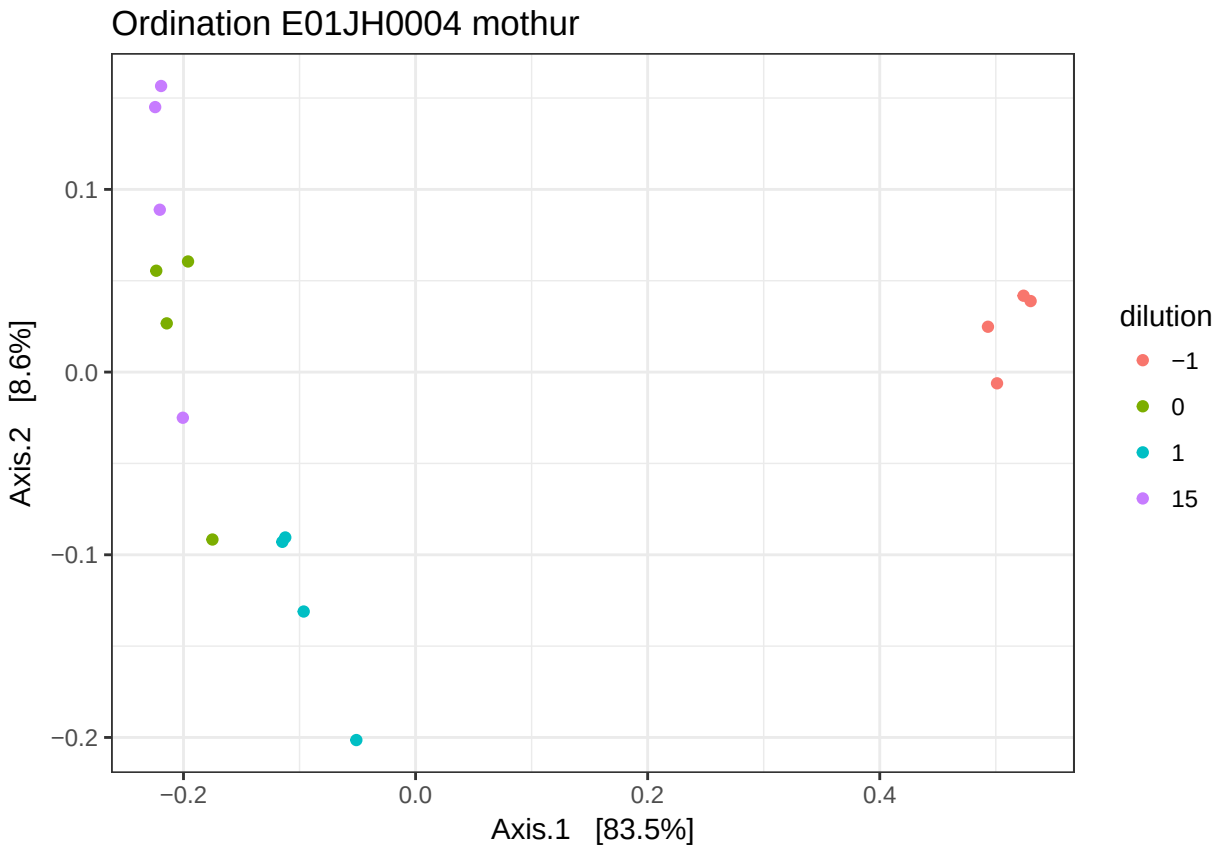
For ordinations of individual biological replicates and pipelines dilution -1 is closest to dilution 1 for all ordinations excluding biological replicate 17, where the closest dilution to -1 is ambiguous. The percent of the observed variability explained by the principal corrdiate separating dilution -1 from other dilutions is greater than 60 percent for all ordinations.

```
ps_plot$ord_plots
```

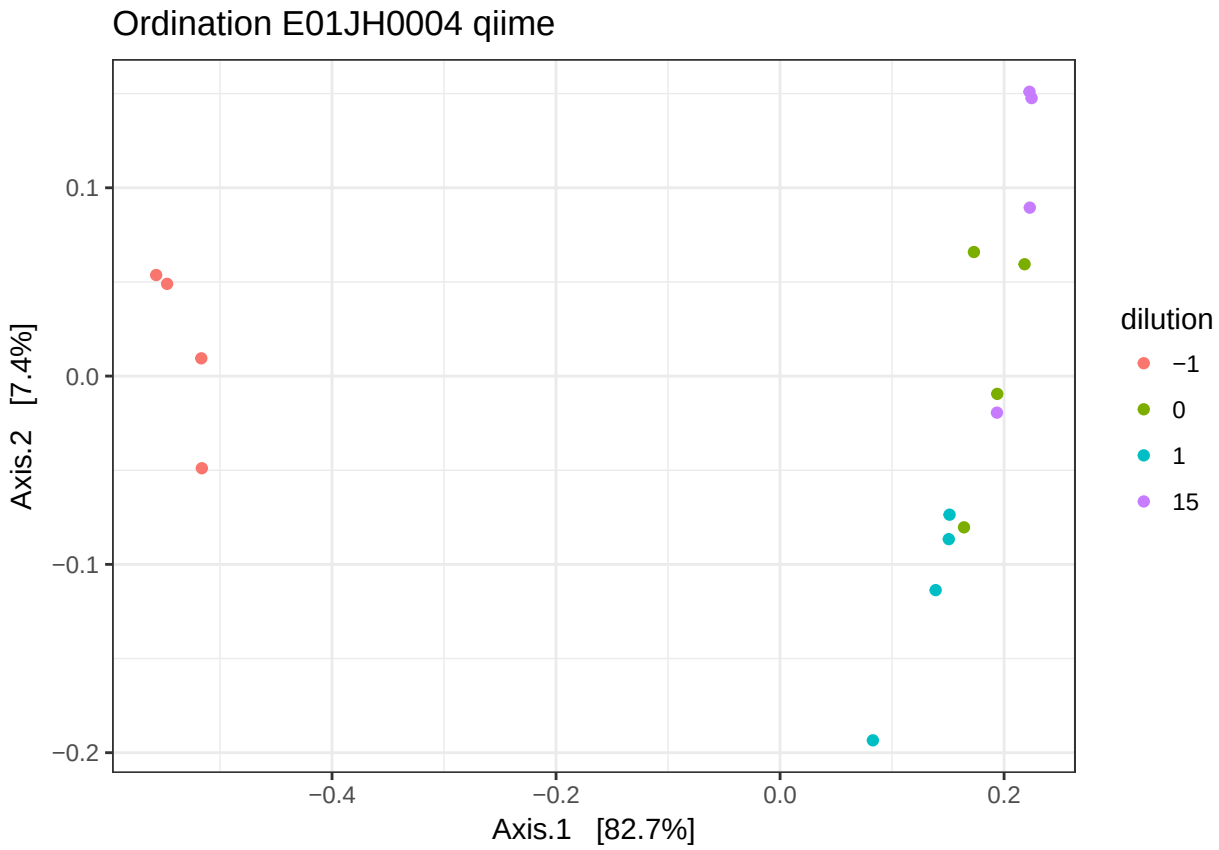
```
## $dada2
```



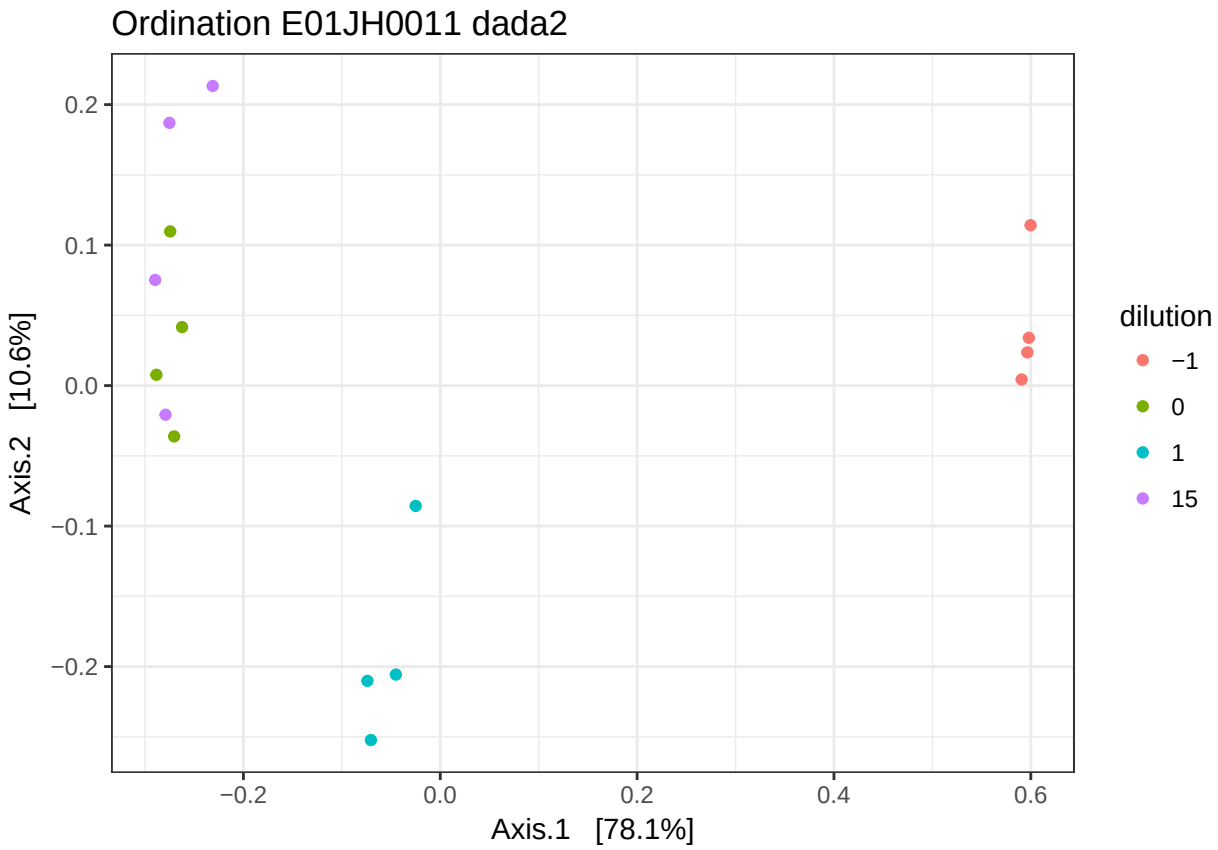

```
##
## $mothur
```


Ordination E01JH0004 mothur
0.1
0.0
−0.1
−0.2
Axis.2 [8.6%]
−0.2
0.0
0.2
0.4
Axis.1 [83.5%]
dilution
−1
0
1
15

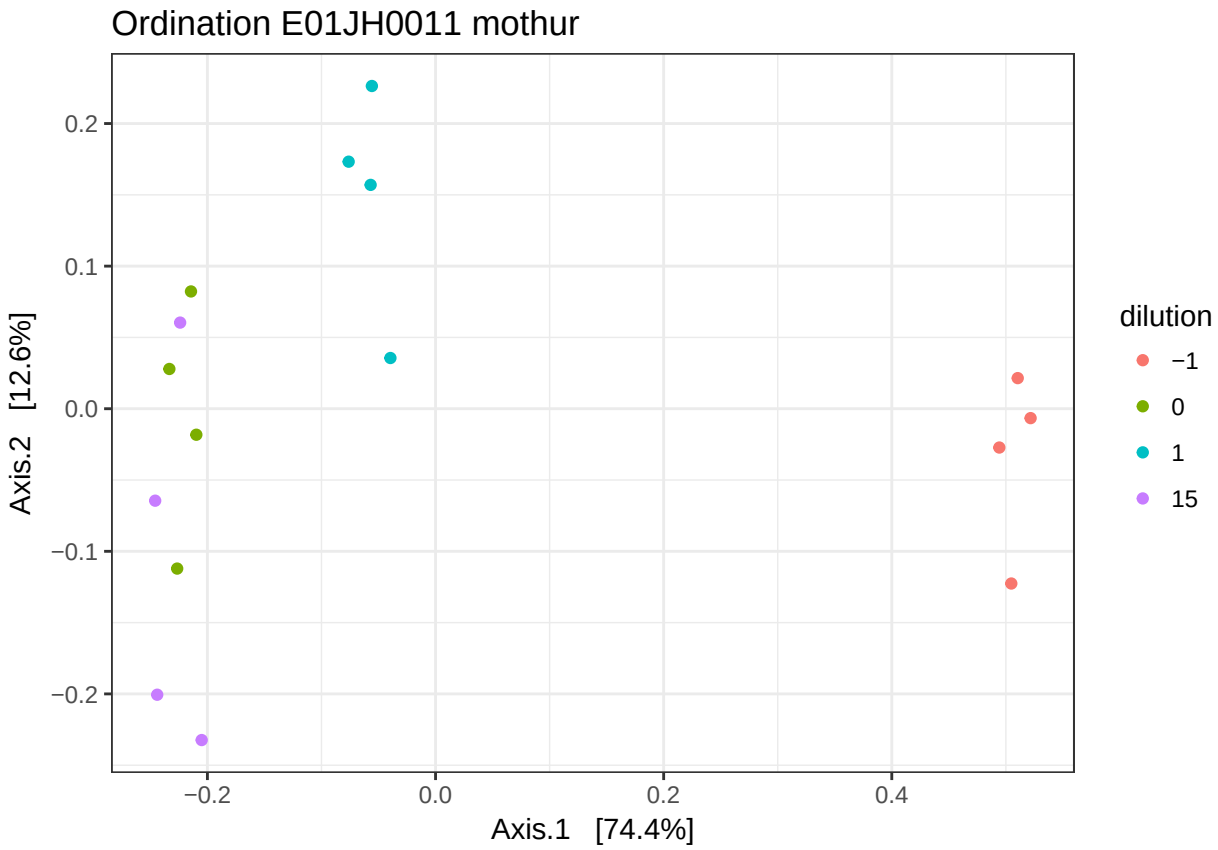
```
##
## $qiime
```


Ordination E01JH0004 qiime
0.1
0.0
−0.1
−0.2
Axis.2 [7.4%]
−0.4
−0.2
0.0
0.2
Axis.1 [82.7%]
dilution
−1
0
1
15

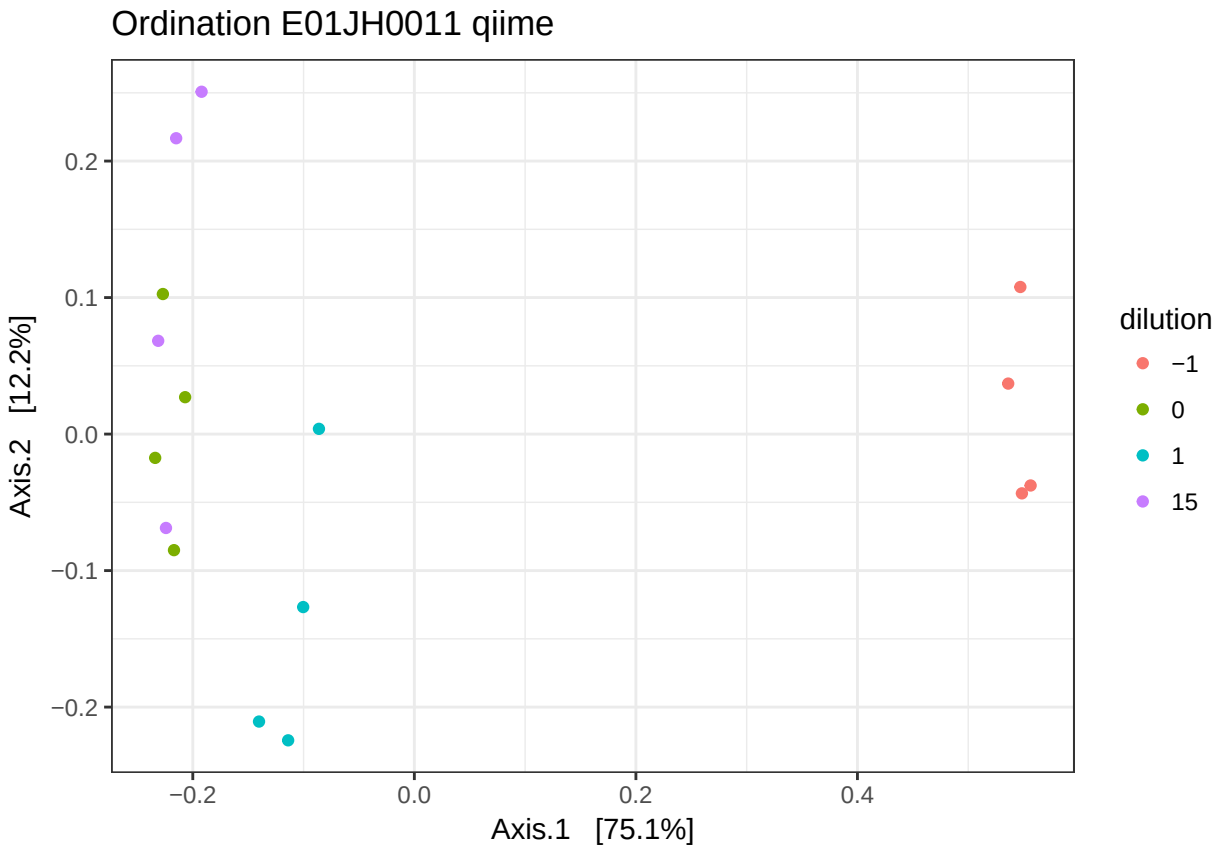
```
##
## $dada2
```


Ordination E01JH0011 dada2
Axis.2 [10.6%]
Axis.1 [78.1%]
0.2
0.1
0.0
−0.1
−0.2
−0.2
0.0
0.2
0.4
0.6
dilution
−1
0
1
15

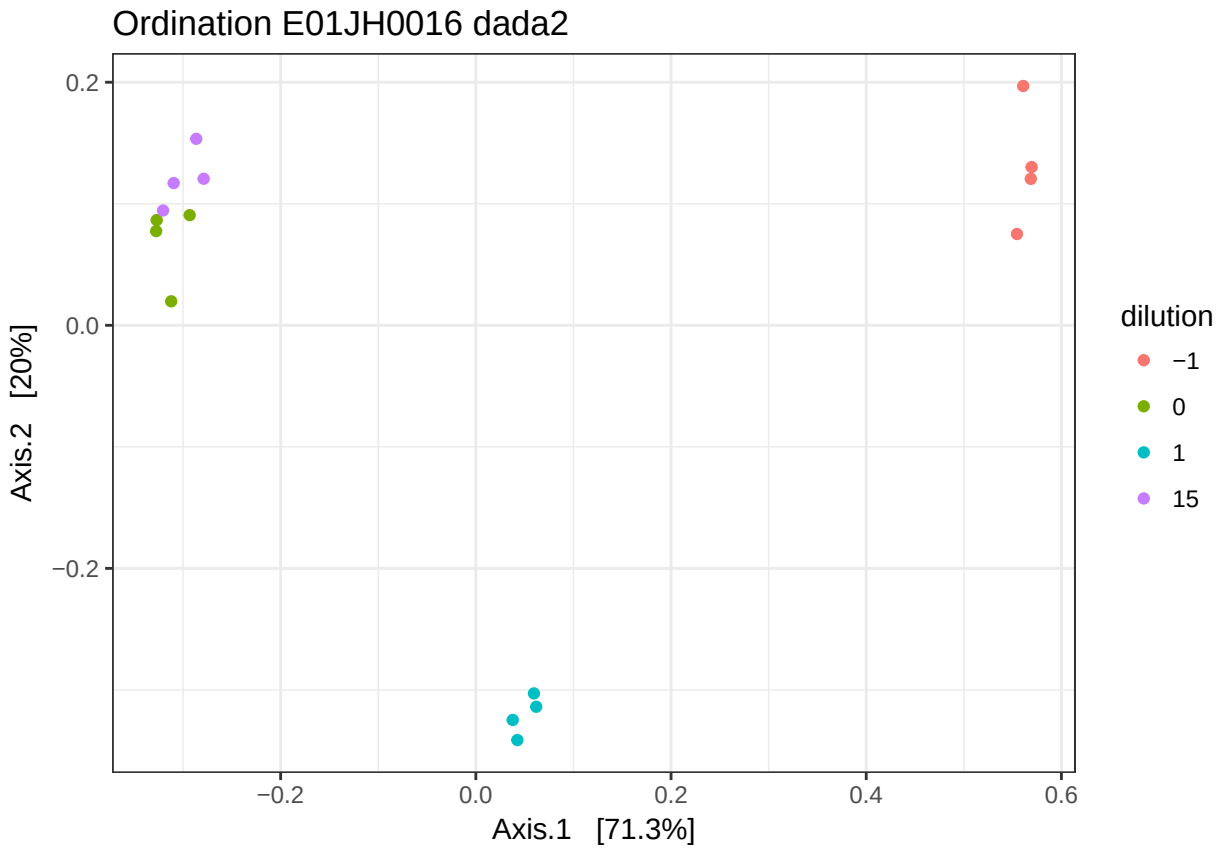
```
##
## $mothur
```


Ordination E01JH0011 mothur
0.2
0.1
0.0
−0.1
−0.2
Axis.2 [12.6%]
−0.2
0.0
0.2
0.4
Axis.1 [74.4%]
dilution
−1
0
1
15

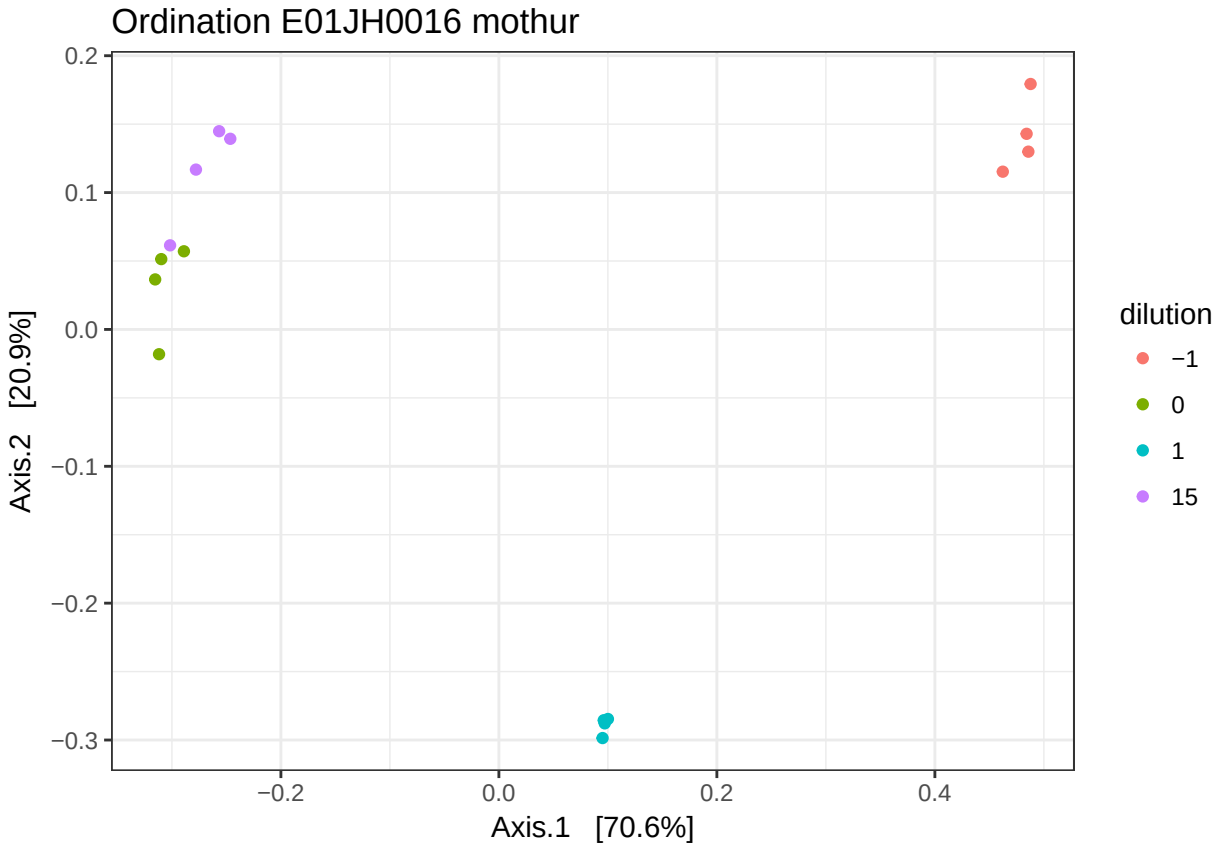
```
##
## $qiime
```


Ordination E01JH0011 qiime
0.2
0.1
0.0
−0.1
−0.2
Axis.2 [12.2%]
−0.2
0.0
0.2
0.4
Axis.1 [75.1%]
dilution
−1
0
1
15

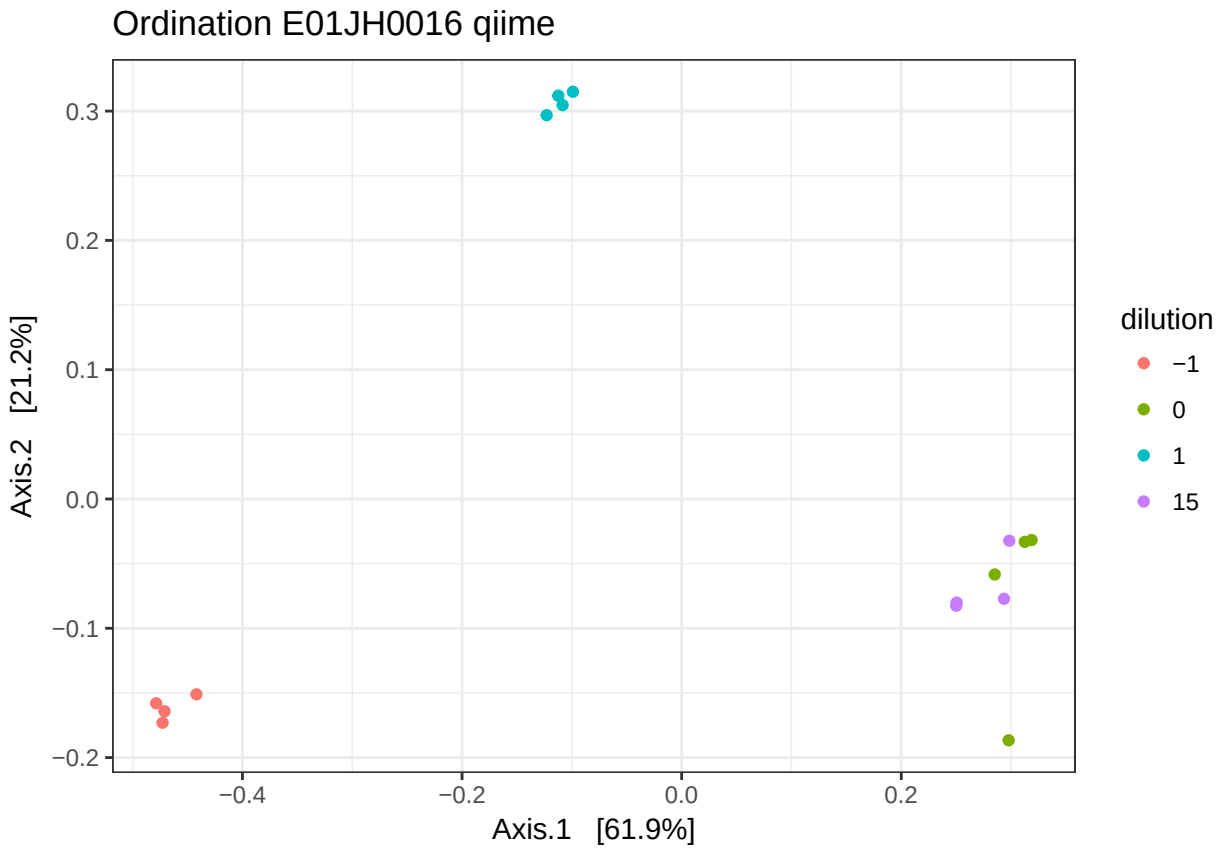
```
##
## $dada2
```


Ordination E01JH0016 dada2
0.2
0.0
−0.2
Axis.2 [20%]
−0.2
0.0
0.2
0.4
0.6
Axis.1 [71.3%]
dilution
−1
0
1
15

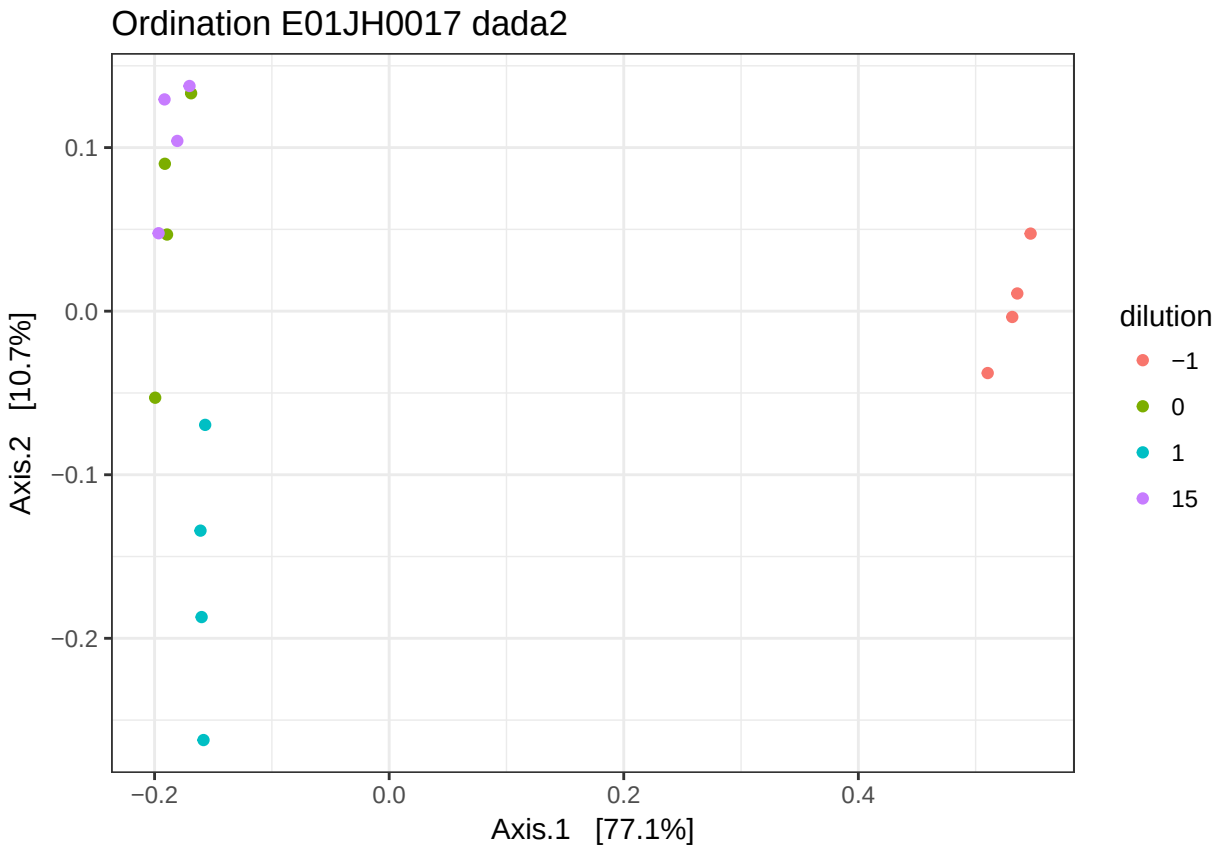
```
##
## $mothur
```


Ordination E01JH0016 mothur
0.2
0.1
0.0
−0.1
−0.2
−0.3
Axis.2 [20.9%]
−0.2
0.0
0.2
0.4
Axis.1 [70.6%]
dilution
−1
0
1
15

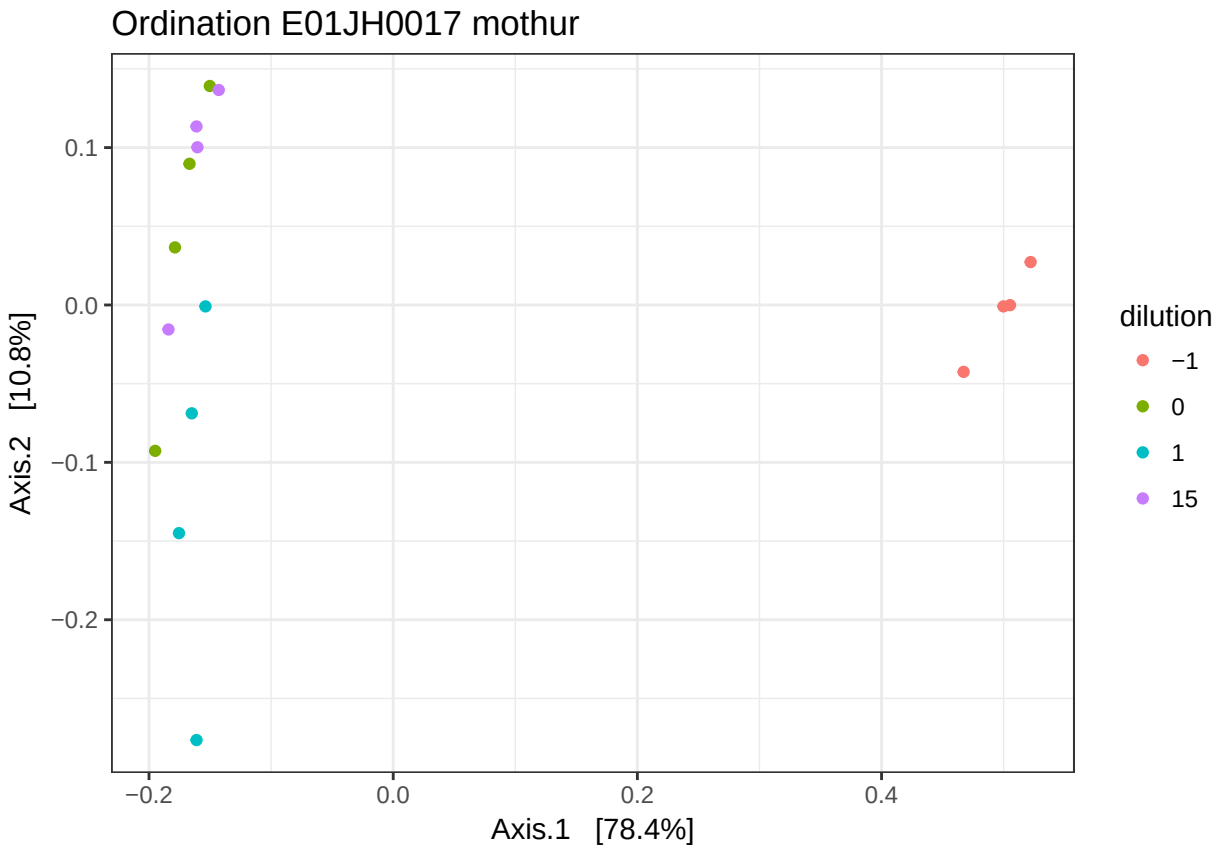
```
##
## $qiime
```


Ordination E01JH0016 qiime
0.3
0.2
0.1
0.0
−0.1
−0.2
Axis.2 [21.2%]
−0.4
−0.2
0.0
0.2
Axis.1 [61.9%]
dilution
−1
0
1
15

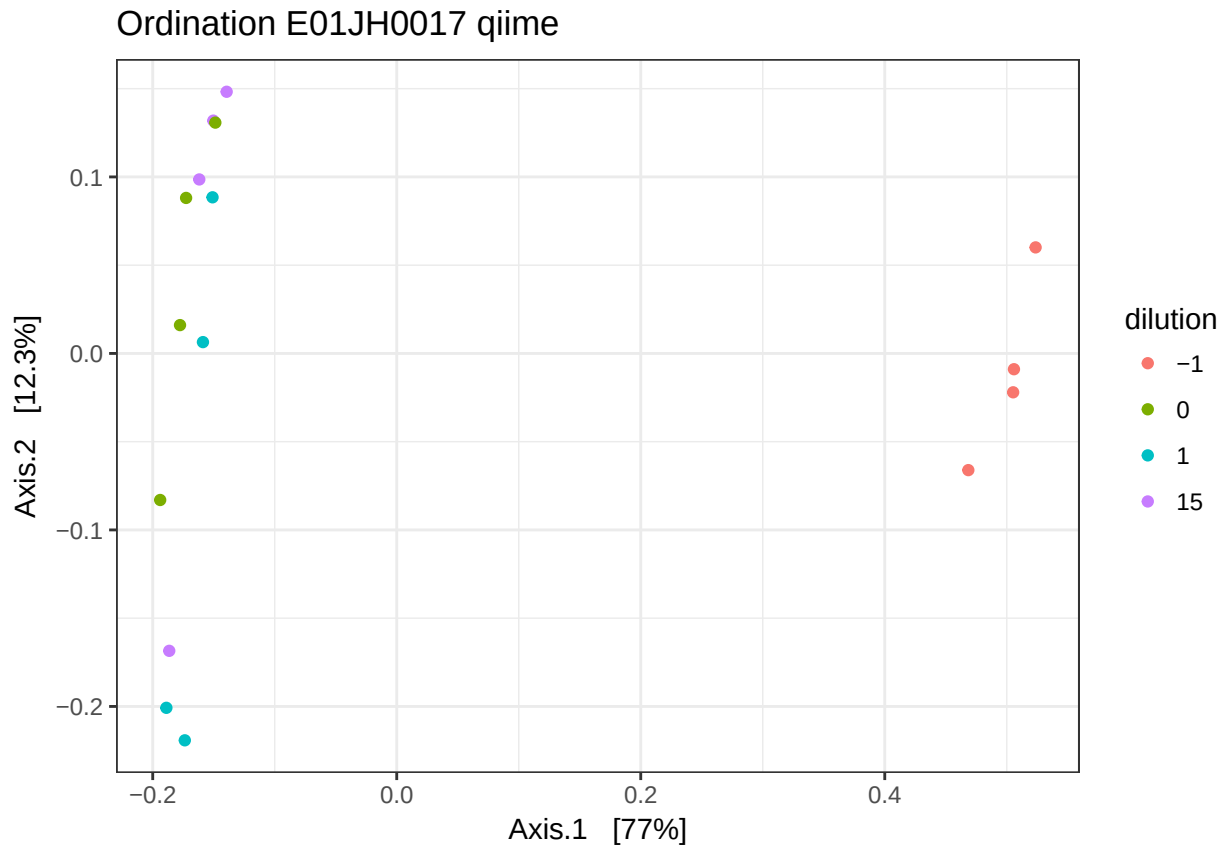
```
##
## $dada2
```


Ordination E01JH0017 dada2
0.1
0.0
−0.1
−0.2
Axis.2 [10.7%]
−0.2
0.0
0.2
0.4
Axis.1 [77.1%]
dilution
−1
0
1
15

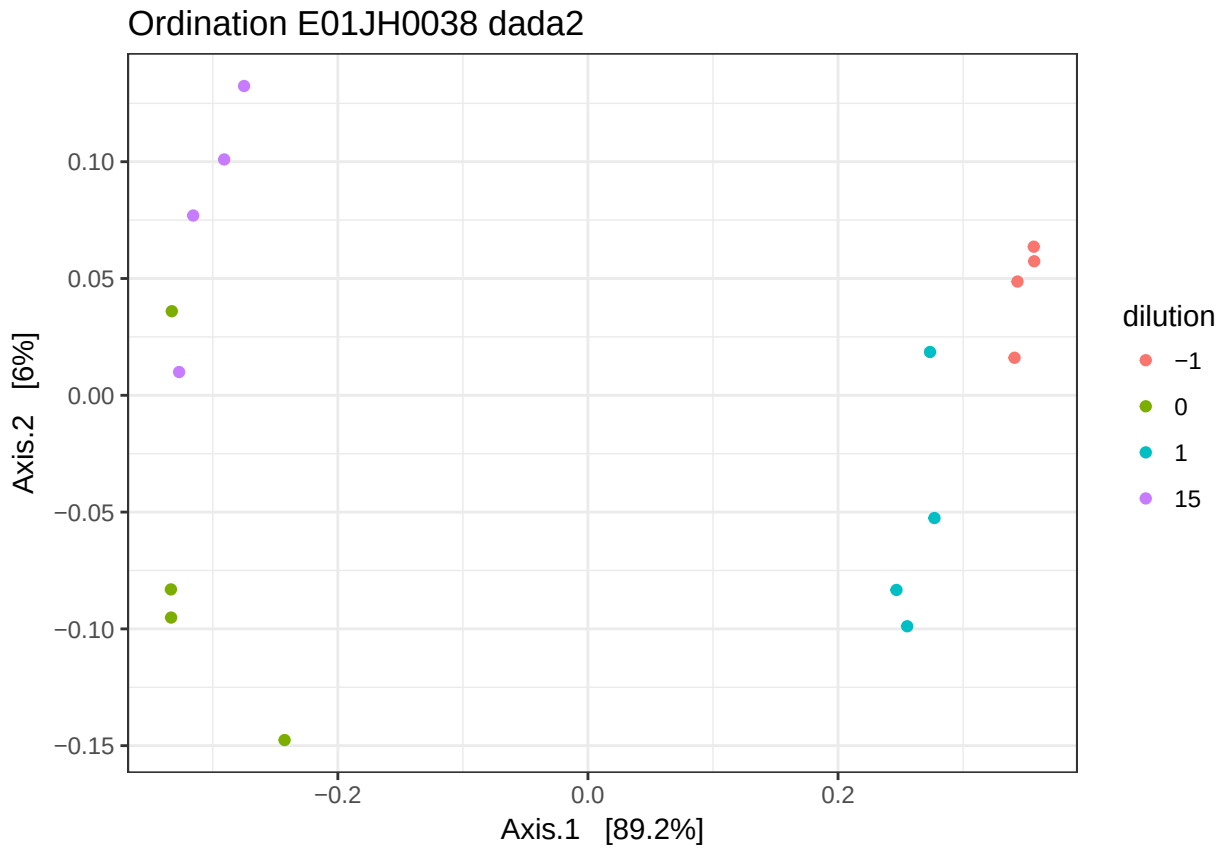
```
##
## $mothur
```


Ordination E01JH0017 mothur
0.1
0.0
−0.1
−0.2
Axis.2 [10.8%]
−0.2
0.0
0.2
0.4
Axis.1 [78.4%]
dilution
−1
0
1
15

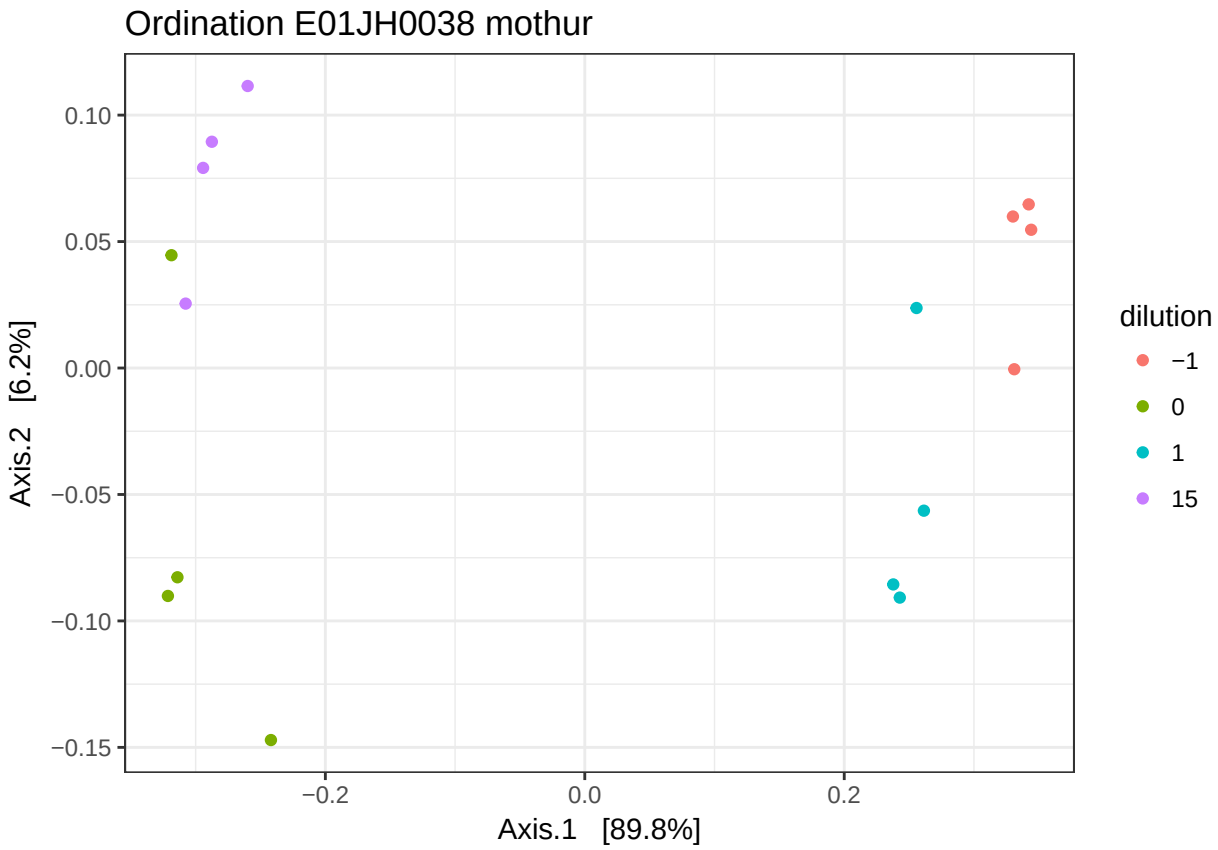
```
##
## $qiime
```


Ordination E01JH0017 qiime
0.1
0.0
−0.1
−0.2
Axis.2 [12.3%]
−0.2
0.0
0.2
0.4
Axis.1 [77%]
dilution
−1
0
1
15

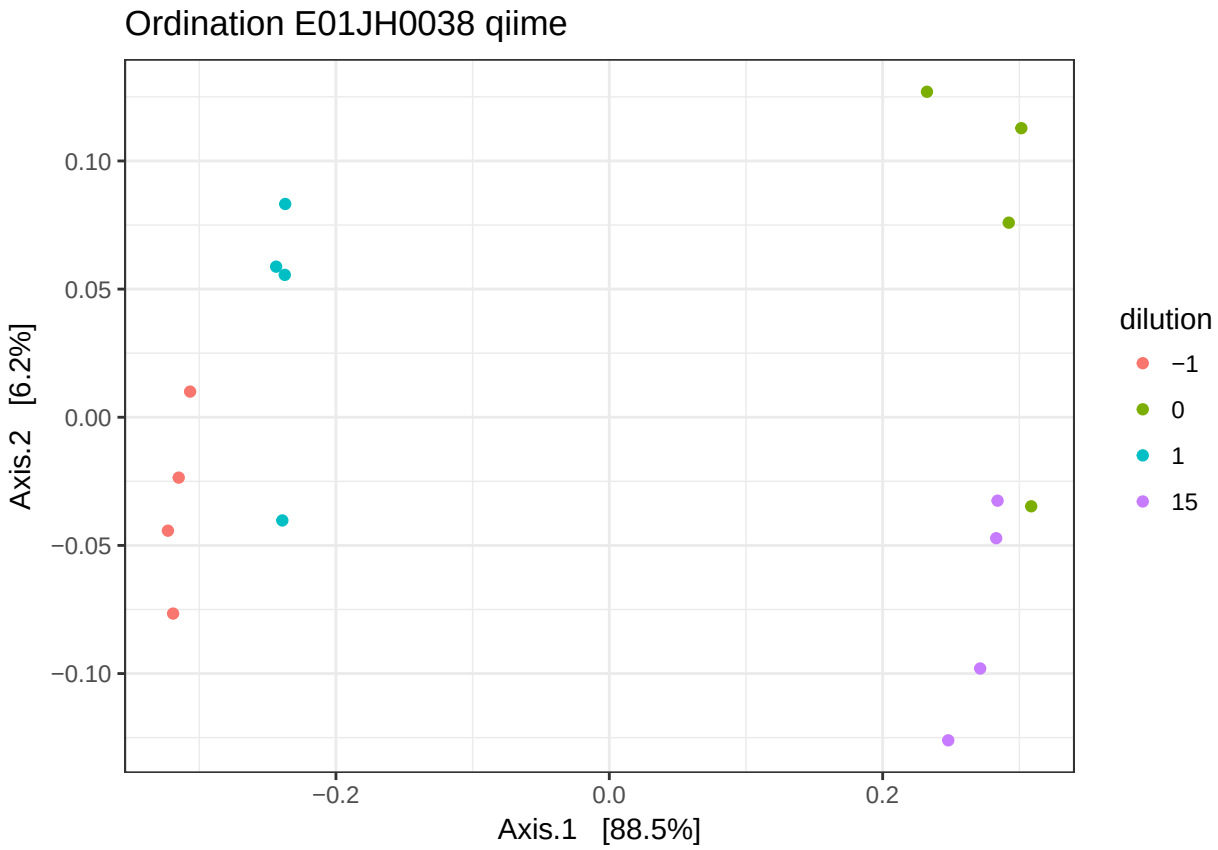
```
##
## $dada2
```


Ordination E01JH0038 dada2
0.10
0.05
0.00
−0.05
−0.10
−0.15
Axis.2 [6%]
−0.2
0.0
0.2
Axis.1 [89.2%]
dilution
−1
0
1
15

```
##
## $mothur
```


Ordination E01JH0038 mothur
0.10
0.05
0.00
−0.05
−0.10
−0.15
Axis.2 [6.2%]
−0.2
0.0
0.2
Axis.1 [89.8%]
dilution
−1
0
1
15

```
##
## $qiime
```



Conclusion

The results indicate that the pre and post treatment unmixed samples were most likely switched in the PCR plate. The ERCC spike-in results indicate that this error did not impact how the titrations were made, therefore expectations regarding the count values holds after correcting for the switched samples.