mock_microbiome_analysis

Sean Lu

2025-07-08

Setup

```r
library("phyloseq")
library("ggplot2")
library("dplyr")
library("tibble")
library("ggpubr")
library("DESeq2")
library("ggsci")
library("colorspace")
library("flextable")
library("purrr")
```

```r
ground_truth_taxonomy <- read.csv("data_processed/kozich_2013/ground_truth_taxonomy.csv",
                      head = TRUE)
species_ground_truth <- ground_truth_taxonomy$species
#ground_truth_taxids <- ground_truth_taxonomy |>
#  tidyr::pivot_longer(everything(), names_to = "Taxonomy Level", values_to = "taxon") |>
#  dplyr::distinct(taxon, .keep_all = TRUE) |>
#  mutate(taxid = taxonomizr::getId(taxon, sqlFile = accessions_path)) |>
#  rowwise() |>
#  mutate(taxid = as.numeric(strsplit(taxid, split = ",")[[1]][1]))
#write.csv(ground_truth_taxids,
#          file = "data_processed/kozich_2013/ground_truth_taxids.csv",
#          row.names = FALSE)
ground_truth_taxids <- read.csv("data_processed/kozich_2013/ground_truth_taxids.csv",)

create_summary_df <- function(csv_paths, pipeline_name) {
  summary_df <- data.frame(
    Species = c("No Call", "Incorrect Call",species_ground_truth))
  n = 1
  for (i in csv_paths) {
    # Get Sample Names
    sample_name <- sub("\\..*$", "", basename(i))
    # Clean MetaBlast Table
    metascope_df <- read.csv(i, head = TRUE)
    head(metascope_df)
    metascope_df <- metascope_df |>
      dplyr::select(Genome, read_count) |>
      dplyr::mutate_if(is.numeric, ~ . / sum(.))
    colnames(metascope_df) <- c("Species", sample_name)
    summary_df <- dplyr::left_join(summary_df, metascope_df, by = "Species")
    summary_df[is.na(summary_df)] <- 0

    no_call.metascope <- metascope_df |>
      dplyr::filter(is.na(Species)) |>
      dplyr::select(sample_name) |>
      sum()
    correct_call.metascope <- metascope_df |>
      dplyr::filter(Species %in% species_ground_truth) |>
      dplyr::select(sample_name) |>
      sum()
    incorrect_call.metascope <- 1 - correct_call.metascope - no_call.metascope

    summary_df[1,n+1] <- no_call.metascope
    summary_df[2,n+1] <- incorrect_call.metascope
    n = n + 1
  }


  summary_df_long <- tidyr::pivot_longer(
    summary_df,
    cols = c(2:ncol(summary_df)),
    values_to = "prop"
  )
  summary_df_long$Species <- factor(summary_df_long$Species,
                                    levels = c("No Call", "Incorrect Call", species_ground_truth))
  summary_df_long$name <- factor(summary_df_long$name)
  summary_df_long$pipeline <- factor(rep(pipeline_name, nrow(summary_df_long)))
  return(summary_df_long)
}

ms_df <- create_summary_df(list.files(path = "data_processed/kozich_2013/results",
                             pattern = ".metascope_id.csv",
                             full.names = TRUE,
                             recursive = TRUE),
                  pipeline = "MetaScope")
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use ‘all_of()‘ or ‘any_of()‘ instead.
##   # Was:
##   data %>% select(sample_name)
##
##   # Now:
##   data %>% select(all_of(sample_name))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
ms_p_df <-create_summary_df(list.files(path = "data_processed/kozich_2013/results_priors",
                             pattern = ".metascope_id.csv",
                             full.names = TRUE,
                             recursive = TRUE),
                  pipeline = "MetaScope Priors")
ms_p_b_df <-create_summary_df(list.files(path = "data_processed/kozich_2013/results_metablast_priors_1.
                             pattern = ".metascope_id.csv",
                             full.names = TRUE,
                             recursive = TRUE),
                  pipeline = "MetaBlast")
summary_df <- rbind(ms_df, ms_p_df, ms_p_b_df)

# Adding Ground Truth
ground_truth_df <- data.frame(
  Species = c("No Call", "Incorrect Call",species_ground_truth))
ground_truth_df <- cbind(ground_truth_df,
                         as.data.frame(do.call(cbind, replicate(33, c(0,0,rep(1/21, 21)), simplify = FA
colnames(ground_truth_df) <-  c("Species", as.character(unique(summary_df$name)))
ground_truth_df <- tidyr::pivot_longer(
  ground_truth_df,
  cols = c(2:34),
  values_to = "prop"
)
ground_truth_df$pipeline <- "Ground Truth"
summary_df <- rbind(summary_df, ground_truth_df)
```

Summarize DADA2 Results

```
# Left joining with aggregate because duplicate species names
# dada2_files <- list.files(path = "data_processed/kozich_2013/dada2_results",
#                            full.names = TRUE)
# taxonomy_cols <- c("Species", "Genus", "Family", "Order", "Class", "Phylum", "Kingdom")
#
# # Join DADA2 Files together
# dada2_df <- purrr::map_dfr(dada2_files, function(x){
#   sample_name <- sub("\\..*$", "", basename(x)) |> strsplit(split = "dada2_")
#   sample_name <- sample_name[[1]][2]
#   res <- read.csv(x) |>
#     dplyr::mutate(Species = ifelse(is.na(Species), NA, paste0(Genus, " ", Species))) |>
#     dplyr::mutate_if(is.numeric, ~ . / sum(.)) |>
#     dplyr::mutate(pipeline = "DADA2-NB", name = sample_name) |>
#     dplyr::rowwise() |>
#     dplyr::mutate(
#       taxon = {
#         tax_vals <- c_across(all_of(taxonomy_cols))
#         non_na_vals <- tax_vals[!is.na(tax_vals)]
#         if (length(non_na_vals) > 0) non_na_vals[1] else NA_character_
#       }
#     ) |>
#     dplyr::ungroup()
# })
```

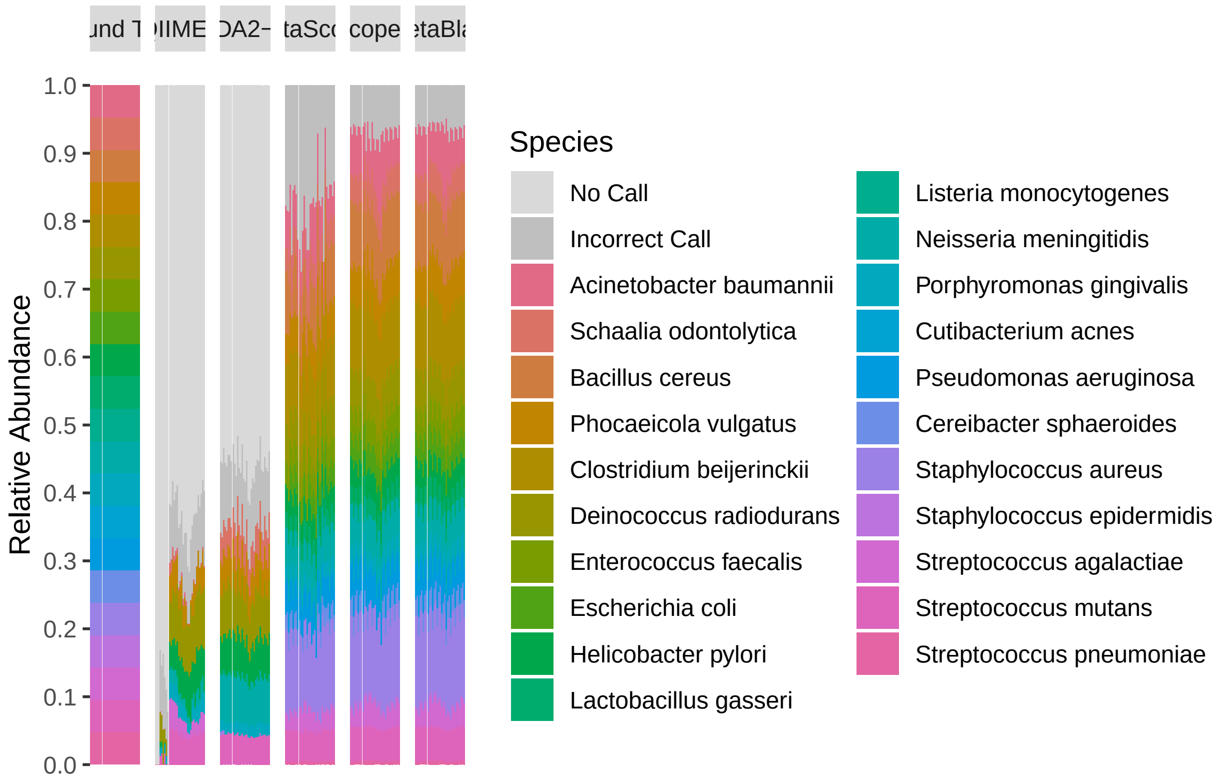
```
#
# # Clean up old taxonomy names
# dada2_df$taxon[dada2_df$taxon == "Bacteroides vulgatus"] <- 'Phocaeicola vulgatus'
# dada2_df$taxon[dada2_df$taxon == 'Actinomyces odontolyticus'] <- "Schaalia odontolytica"
# dada2_df$taxon[dada2_df$taxon == "Propionibacterium acnes"] <- 'Cutibacterium acnes'
# dada2_df$taxon[dada2_df$taxon == "Rhodobacter sphaeroides"] <- 'Cereibacter sphaeroides'
# dada2_df$taxon[dada2_df$taxon == "Clostridium sensu stricto 1 beijerinckii"] <- 'Clostridium beijerin
# dada2_df$taxon[dada2_df$taxon == "Escherichia-Shigella"] <- 'Escherichia' # Assuming DADA2 meant Esch
# dada2_df$taxon[dada2_df$taxon == "Prevotella_9"] <- 'Segatella'
# dada2_df$taxon[dada2_df$taxon == "Prevotella_9 copri"] <- 'Segatella corpi'
# dada2_df$taxon[dada2_df$taxon == "Rhodobacteraceae"] <- 'Paracoccaceae'
# dada2_df$taxon[dada2_df$taxon == "Bacteroides massiliensis"] <- 'Phocaeicola massiliensis'
# dada2_df$taxon[dada2_df$taxon == "Ruminococcaceae"] <- 'Oscillospiraceae'
#
### Add NCBI taxonomy ids
# dada2_df <- dada2_df |>
#   mutate(taxid_raw = taxonomizr::getId(taxon, sqlFile = accessions_path)) |>
#   rowwise() |>
#   mutate(taxid = as.numeric(strsplit(taxid_raw, split = ",")[[1]][1])) |>
#   ungroup() |>
#   mutate(taxid = if_else(is.na(taxid), 9999999, taxid)) |> # REPLACING ALL UNKNOWN TAXONS WITH TAXID
#   dplyr::select(!taxid_raw)
#
### Adding categories
# dada2_df <- dada2_df |>
#   dplyr::mutate(category = case_when(
#     (taxid %in% ground_truth_taxids$taxid) & !is.na(Species) ~ taxon,
#     (taxid %in% ground_truth_taxids$taxid) & is.na(Species) ~ "No Call",
#     !(taxid %in% ground_truth_taxids$taxid) ~ "Incorrect Call")) |>
#   dplyr::group_by(category, name) |>
#   dplyr::summarise(prop = sum(reads_count))
# dada2_df$pipeline = "DADA2-NB"
# colnames(dada2_df) <- c("Species", "name", "prop", "pipeline")
#
# #write.csv(dada2_df, file = "data_processed/kozich_2013/dada2_df.csv",
#           row.names = FALSE)
dada2_df <- read.csv("data_processed/kozich_2013/dada2_df.csv")


summary_df <- rbind(summary_df, dada2_df)
```

Summarize QIIME2 Results

```
# summary_df_qiime2 <- data.frame(
#   Species = c("No Call", "Incorrect Call",species_ground_truth))
#
# qiime2_otu_table <- read.table("data_processed/kozich_2013/qiime2_results/feature_table.tsv",
#            header = TRUE, sep = "\t") |>
#   dplyr::mutate_if(is.numeric, ~ . / sum(.)) |>
#   pivot_longer(cols = !OTU.ID)
#
```

```
# taxonomy_cols <- c("Species", "Genus", "Family", "Order", "Class", "Phylum", "Superkingdom")
# qiime2_tax_table <- read.table("data_processed/kozich_2013/qiime2_results/taxonomy.tsv",
#                                header = TRUE, sep = "\t") |>
#   separate(Taxon, into = c("Superkingdom", "Phylum", "Class",
#                          "Order", "Family", "Genus", "Species"),
#            sep = ";", fill = "right", extra = "drop") |>
#   mutate(across(Superkingdom:Species, ~ str_remove(., "^\\s*[a-z]__"))) |>
#   mutate(Species = sub("_", " ", Species)) |>
#   rowwise() |>
#   mutate(
#     taxon = {
#       tax_vals <- c_across(all_of(taxonomy_cols))
#       non_na_vals <- tax_vals[!is.na(tax_vals)]
#       if (length(non_na_vals) > 0) non_na_vals[1] else NA_character_0}) |>
#   ungroup()
#
# # Manually Address NAs
# qiime2_tax_table$taxon[qiime2_tax_table$taxon == "Clostridium_sensu_stricto_1"] <- "Clostridium"
# qiime2_tax_table$taxon[qiime2_tax_table$taxon == "Escherichia-Shigella"] <- "Escherichia" # Escherich
# qiime2_tax_table$taxon[qiime2_tax_table$taxon == "Bacteroides vulgatus"] <- "Phocaeicola vulgatus"
# qiime2_tax_table$taxon[qiime2_tax_table$taxon == "Bacteroides massiliensis"] <- "Phocaeicola massilie
# qiime2_tax_table$taxon[qiime2_tax_table$taxon == "Alistipes obesi"] <- "Alistipes communis"
# qiime2_tax_table$taxon[qiime2_tax_table$taxon == "Candidatus_Udaeobacter"] <- "Candidatus Udaeobacter
# qiime2_tax_table$taxon[qiime2_tax_table$taxon == "Ruminococcaceae"] <- "Oscillospiraceae"
#
# # Add taxids to QIIME2 taxonomy table
# qiime2_tax_table <- qiime2_tax_table |>
#   mutate(taxid_raw = taxonomizr::getId(taxon, sqlFile = accessions_path)) |>
#   rowwise() |>
#   mutate(taxid = as.numeric(strsplit(taxid_raw, split = ",")[[1]][1])) |>
#   ungroup() |>
#   mutate(taxid = if_else(is.na(taxid), 9999999, taxid)) |> # REPLACING ALL UNKNOWN TAXONS WITH TAXID
#   dplyr::select(!taxid_raw)
#
#
# qiime2_df <- full_join(qiime2_otu_table, qiime2_tax_table, by=join_by(OTU.ID == Feature.ID))
#
#
# qiime2_df <- qiime2_df |>
#   dplyr::mutate(category = case_when(
#     (taxid %in% ground_truth_taxids$taxid) & !is.na(Species) ~ taxon,
#     (taxid %in% ground_truth_taxids$taxid) & is.na(Species) ~ "No Call",
#     !(taxid %in% ground_truth_taxids$taxid) ~ "Incorrect Call")) |>
#   dplyr::group_by(category, name) |>
#   dplyr::summarise(prop = sum(value))
# qiime2_df$pipeline <- "QIIME2"
# colnames(qiime2_df) <- c("Species", "name", "prop", "pipeline")
#
# summary_df_qiime2 <-
#   expand.grid(Species = c("No Call", "Incorrect Call", species_ground_truth),
#             name = unique(qiime2_df$name),
#             stringsAsFactors = TRUE) |>
#   merge(qiime2_df, by = c("Species", "name"), all.x = TRUE) |>
#   replace_na(list(prop = 0, pipeline = "QIIME2"))
#
# write.csv(summary_df_qiime2, file = "data_processed/kozich_2013/qiime2_df.csv",
#           row.names = FALSE)
summary_df_qiime2 <- read.csv("data_processed/kozich_2013/qiime2_df.csv")

summary_df <- rbind(summary_df, summary_df_qiime2)
```

Plotting Relative Abundance of Mock Microbiome



```
## Warning: fonts used in ‘flextable‘ are ignored because the ‘pdflatex‘ engine is
## used and not ‘xelatex‘ or ‘lualatex‘. You can avoid this warning by using the
## ‘set_flextable_defaults(fonts_ignore=TRUE)‘ command or use a compatible engine
## by defining ‘latex_engine: xelatex‘ in the YAML header of the R Markdown
## document.
```

Profiler	Correct Call	No Call	Incorrect Call
QIIME2	0.211 ± 0.116	0.708 ± 0.135	0.081 ± 0.028
DADA2-NB	0.347 ± 0.023	0.56 ± 0.023	0.093 ± 0.006
MetaScope	0.821 ± 0.047	0 ± 0	0.179 ± 0.047

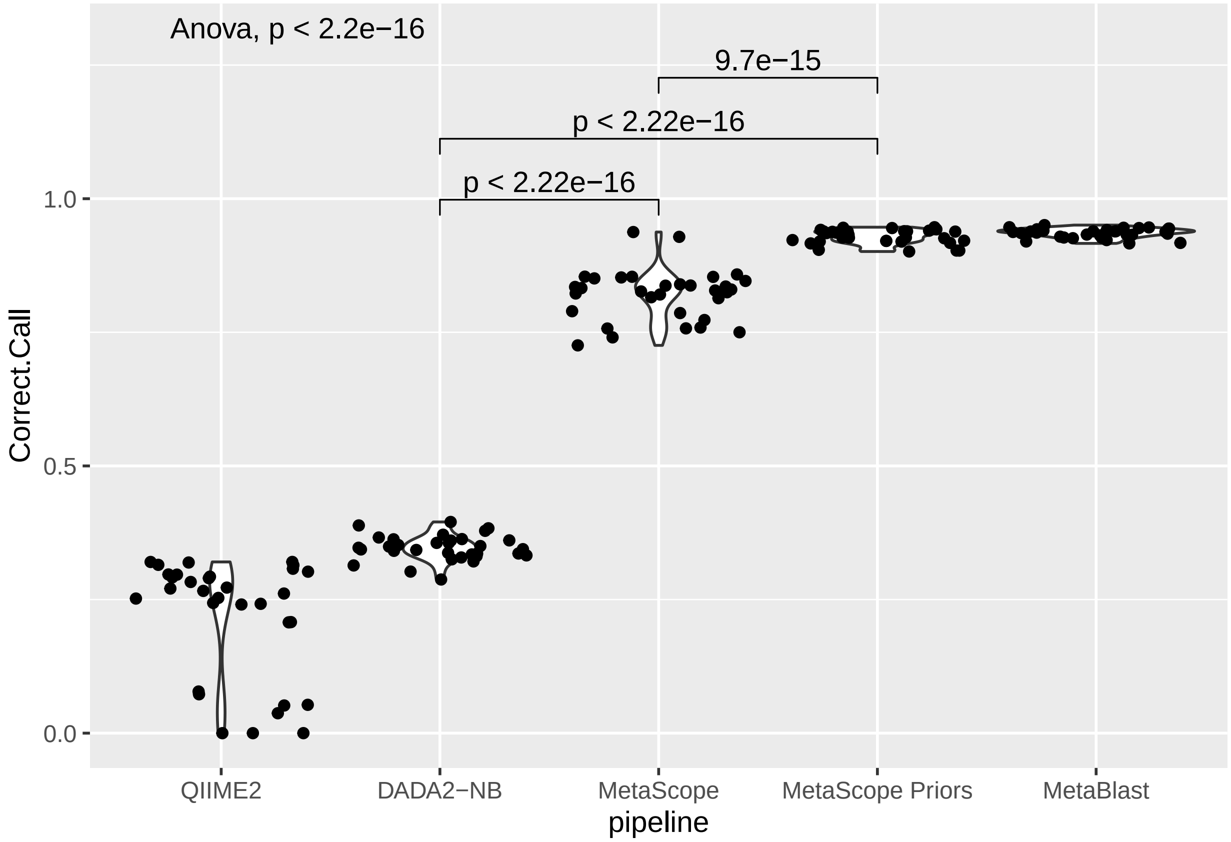
Profiler	Correct Call	No Call	Incorrect Call
MetaScope Priors	0.929 ± 0.013	0 ± 0	0.071 ± 0.013
MetaBlast	0.936 ± 0.009	0 ± 0	0.064 ± 0.009

```r
# This metascope is k - 25
p_bar_with_stats<- summary_df |> dplyr::filter(Species %in% c("No Call", "Incorrect Call")) |>
  tidyr::pivot_wider(names_from = Species, values_from = prop, names_repair = "universal") |>
  dplyr::mutate(Correct.Call = 1 - (No.Call + Incorrect.Call)) |>
  dplyr::filter(pipeline != "Ground Truth") |>
  dplyr::group_by(pipeline) |>
  dplyr::group_by(name) |>
  ggplot(aes(x = pipeline, y = Correct.Call)) +
  geom_violin() +
  geom_jitter() +
  ylim(0,1.3) +
  stat_compare_means(label.y = 1.3, method = "anova", paired = TRUE) +
  stat_compare_means(comparisons = list(c("DADA2-NB", "MetaScope"),
                                        c("DADA2-NB", "MetaScope Priors"),
                                        c("MetaScope", "MetaScope Priors"),
                                        c("MetaScope Priors", "MetaBlast")),
                     method = "t.test",
                     paired = TRUE)
```

```
## New names:
## * ‘No Call‘ -> ‘No.Call‘
## * ‘Incorrect Call‘ -> ‘Incorrect.Call‘
```

```r
p_bar_with_stats
```

```
## Warning: Removed 1 row containing non-finite outside the scale range
## (‘stat_ydensity()‘).

## Warning: Removed 1 row containing non-finite outside the scale range
## (‘stat_compare_means()‘).

## Warning: Removed 1 row containing non-finite outside the scale range
## (‘stat_signif()‘).

## Warning: Removed 1 row containing missing values or values outside the scale range
## (‘geom_point()‘).

## Warning: Removed 3 rows containing missing values or values outside the scale range
## (‘geom_signif()‘).
```

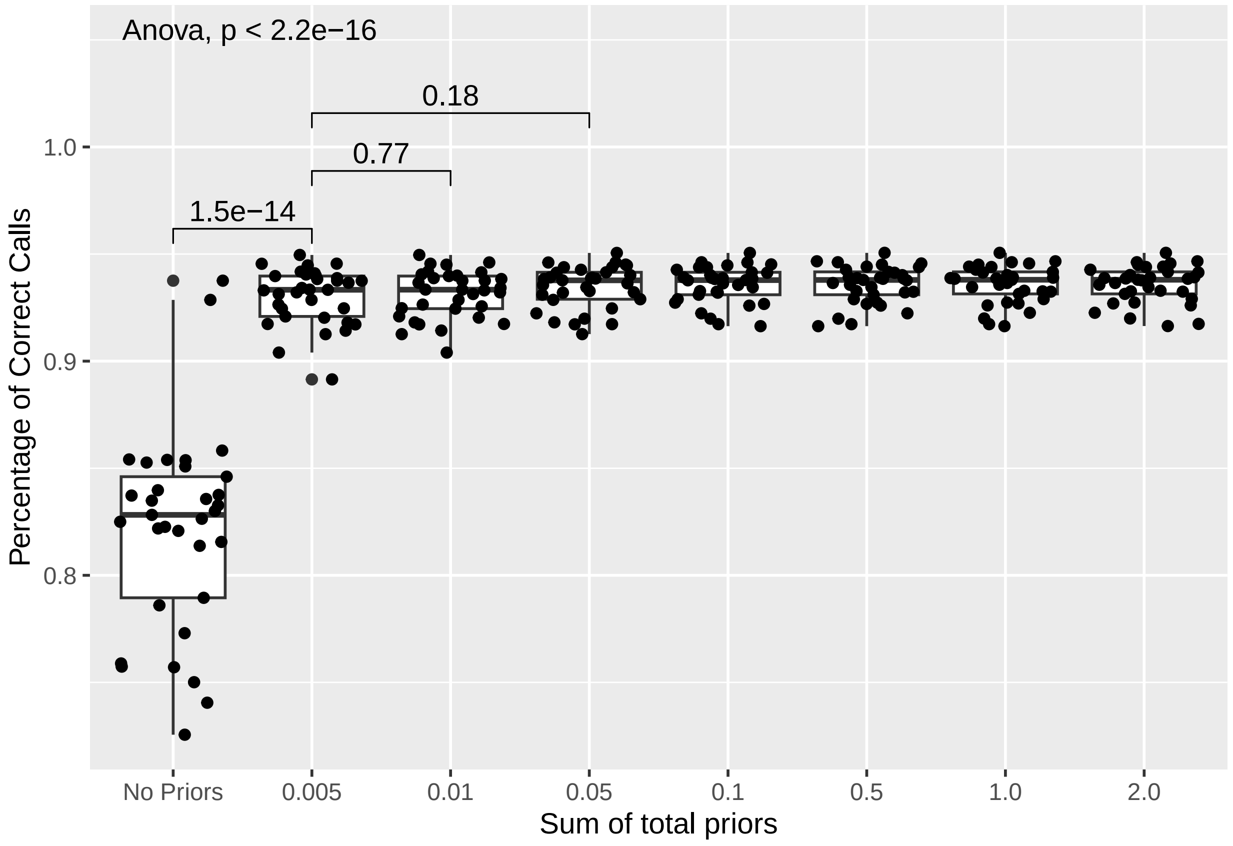


Sensitivity Analysis

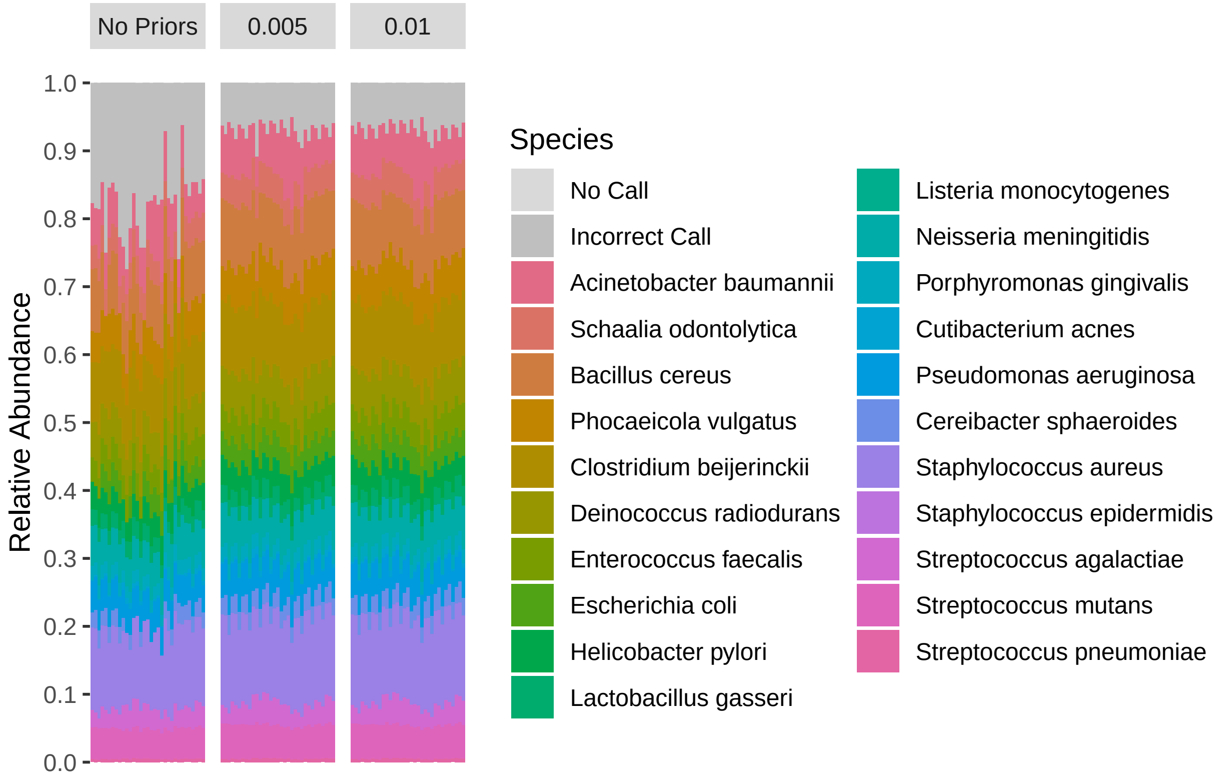
```
priors_0<- create_summary_df(
  list.files(
    path = "data_processed/kozich_2013/results",
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "No Priors"
)

priors_0.005 <- create_summary_df(
  list.files(
    path = "data_processed/kozich_2013/results_priors_0.005",
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "0.005"
)

priors_0.01 <- create_summary_df(
  list.files(
    path = "data_processed/kozich_2013/results_priors_0.01",
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "0.01"
)

priors_0.05 <- create_summary_df(
  list.files(
    path = "data_processed/kozich_2013/results_priors_0.05",
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "0.05"
)

priors_0.1 <- create_summary_df(
  list.files(
    path = "data_processed/kozich_2013/results_priors_0.1",
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "0.1"
)

priors_0.5 <- create_summary_df(
  list.files(
    path = "data_processed/kozich_2013/results_priors_0.5",
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "0.5"
)

priors_1.0 <- create_summary_df(
  list.files(
    path = "data_processed/kozich_2013/results_priors_1.0",
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "1.0"
)

priors_2.0 <- create_summary_df(
  list.files(
    path = "data_processed/kozich_2013/results_priors_2.0",
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "2.0"
)


summary_df_sensitivity <- rbind(priors_0, priors_0.005, priors_0.01, priors_0.05, priors_0.1, priors_0.5
summary_df_sensitivity |> dplyr::filter(Species %in% c("Correct Call", "Incorrect Call")) |>
  tidyr::pivot_wider(names_from = Species, values_from = prop, names_repair = "universal") |>
  dplyr::mutate(Correct.Call = 1 - as.numeric(Incorrect.Call)) |>
  dplyr::filter(pipeline != "Ground Truth") |>
  dplyr::group_by(pipeline) |>
  dplyr::group_by(name) |>
  ggplot(aes(x = pipeline, y = Correct.Call)) +
  geom_boxplot() +
  geom_jitter() +
  #theme(axis.text.x = element_text(angle = 90)) +
  xlab("Sum of total priors") +
  ylab("Percentage of Correct Calls") +
  stat_compare_means(label.y = 1.05, method = "anova", paired = TRUE) +
  stat_compare_means(comparisons = list(c("No Priors", "0.005"),
                                        c("0.005", "0.01"), c("0.005", "0.05")),
                     method = "wilcox.test",
                     paired = FALSE)
```

```
## New names:
## * ‘Incorrect Call‘ -> ‘Incorrect.Call‘
```

```
## Warning in wilcox.test.default(c(0.936613634323669, 0.92448846455617,
## 0.941782795512704, : cannot compute exact p-value with ties
```



```
summary_df_final_3 <- rbind(priors_0, priors_0.005, priors_0.01)
p3 <- ggplot(data = summary_df_final_3  , aes(fill = Species, y = prop, x = name)) +
  geom_bar(position ="stack", stat = "identity")+
  scale_fill_manual(values = wheel_colors, name = "Species") +
  ylab("Relative Abundance") +
  xlab("") +
  scale_y_continuous(breaks = seq(0,1, by = 0.1)) +
  facet_grid(~pipeline) +
  theme(axis.text.x=element_blank(),
        axis.ticks.x = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.background = element_blank())
plot(p3)
```



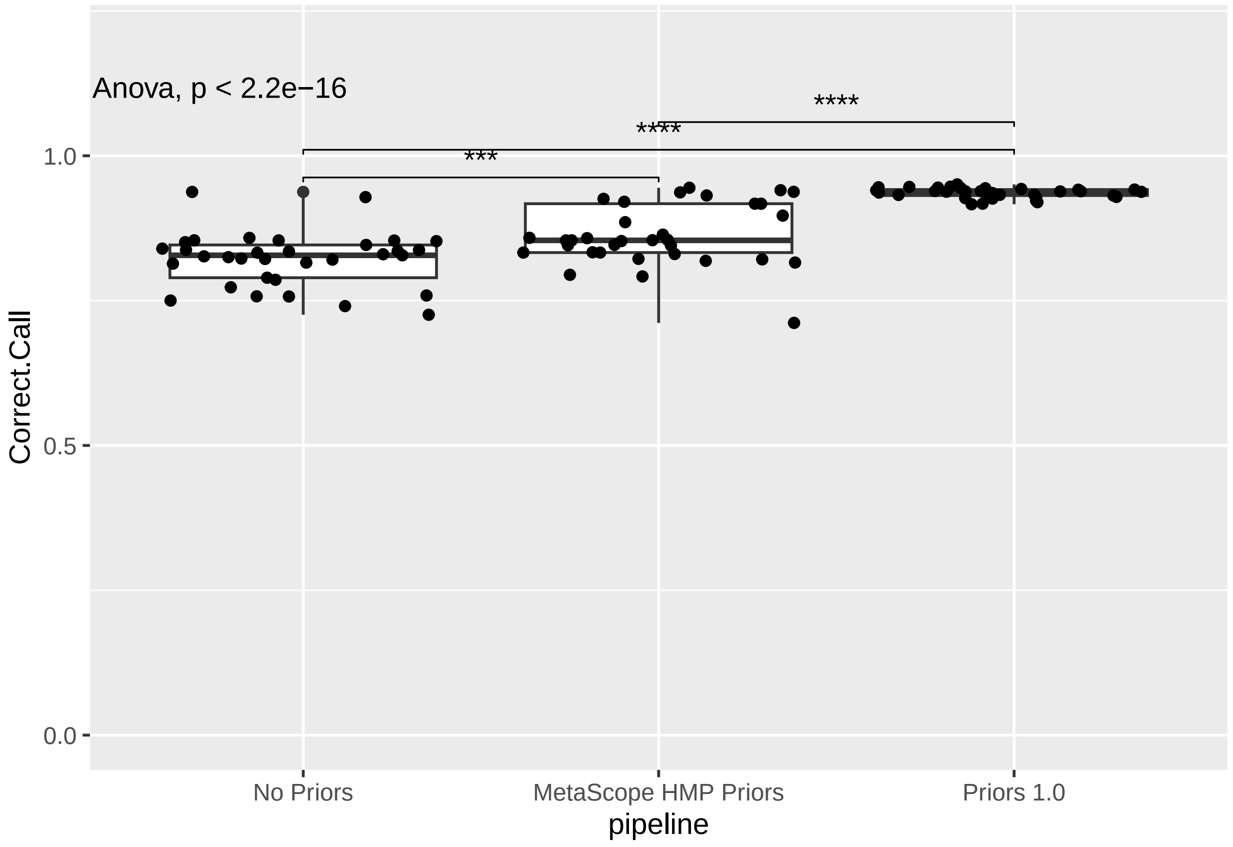
#HMP priors

```
hmp_priors<- create_summary_df(
  list.files(
    path = "data_processed/kozich_2013/results_hmp_priors",
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "MetaScope HMP Priors"
)
```

```r
hmp_test <- rbind(priors_0, hmp_priors, priors_1.0) |>
  dplyr::filter(Species %in% c("No Call", "Incorrect Call")) |>
  tidyr::pivot_wider(names_from = Species, values_from = prop, names_repair = "universal") |>
  dplyr::mutate(Correct.Call = 1 - (No.Call + Incorrect.Call))
```

```
## New names:
## * ‘No Call‘ -> ‘No.Call‘
## * ‘Incorrect Call‘ -> ‘Incorrect.Call‘
```

```r
ggplot(hmp_test, aes(x=pipeline, y=Correct.Call)) +
  geom_boxplot() +
  stat_compare_means(label.y = 1.1, label.x = 0.55, method = "anova", paired = TRUE) +
  stat_compare_means(comparisons = list(c("No Priors", "MetaScope HMP Priors"),
                                        c("No Priors", "1.0"), c("MetaScope HMP Priors", "1.0")),
                     method = "wilcox.test",
                     paired = TRUE,
                     step.increase = .2,
                     symnum.args = list(cutpoints = c(0, 0.0001, 0.001, 0.01, 0.05, 1),
                                        symbols = c("****", "***", "**", "*", "ns"))) +
  geom_jitter() +
  scale_x_discrete(labels=c("1.0" = "Priors 1.0")) +
  scale_y_continuous(breaks=c(0,0.5,1),
                     limits = c(0,1.2))
```


Anova, p < 2.2e−16
****
****
***
Correct.Call
No Priors
MetaScope HMP Priors
Priors 1.0
pipeline