mock_microbiome_analysis

Sean Lu

2025-06-10

Setup

```r
library("phyloseq")
library("ggplot2")
library("dplyr")
library("tibble")
library("ggpubr")
library("phylosmith")
library("DESeq2")
library("EnhancedVolcano")
library("SpiecEasi")
library("ggsci")
library("colorspace")
library("flextable")
```

```r
priors_df <- read.csv("data_processed/kozich_2013/prior_weights/kozich_prior_weights_1.0.csv",
                      head = TRUE)
species_ground_truth <- priors_df$species

create_summary_df <- function(csv_paths, pipeline_name) {
  summary_df <- data.frame(
    Species = c("No Call", "Incorrect Call",priors_df$species))
  n = 1
  for (i in csv_paths) {
    # Get Sample Names
    sample_name <- sub("\\..*$", "", 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)
    #metascope_df[metascope_df == "Phocaeicola vulgatus"] <- 'Bacteroides vulgatus'
    #metascope_df[metascope_df == "Schaalia odontolytica"] <- 'Actinomyces odontolyticus'
    #metascope_df[metascope_df == "Cutibacterium acnes"] <- 'Propionibacterium acnes'
    #metascope_df[metascope_df == "Cereibacter sphaeroides"] <- 'Rhodobacter sphaeroides'
    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% priors_df$species) |>
      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", priors_df$species))
  summary_df_long$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.
```

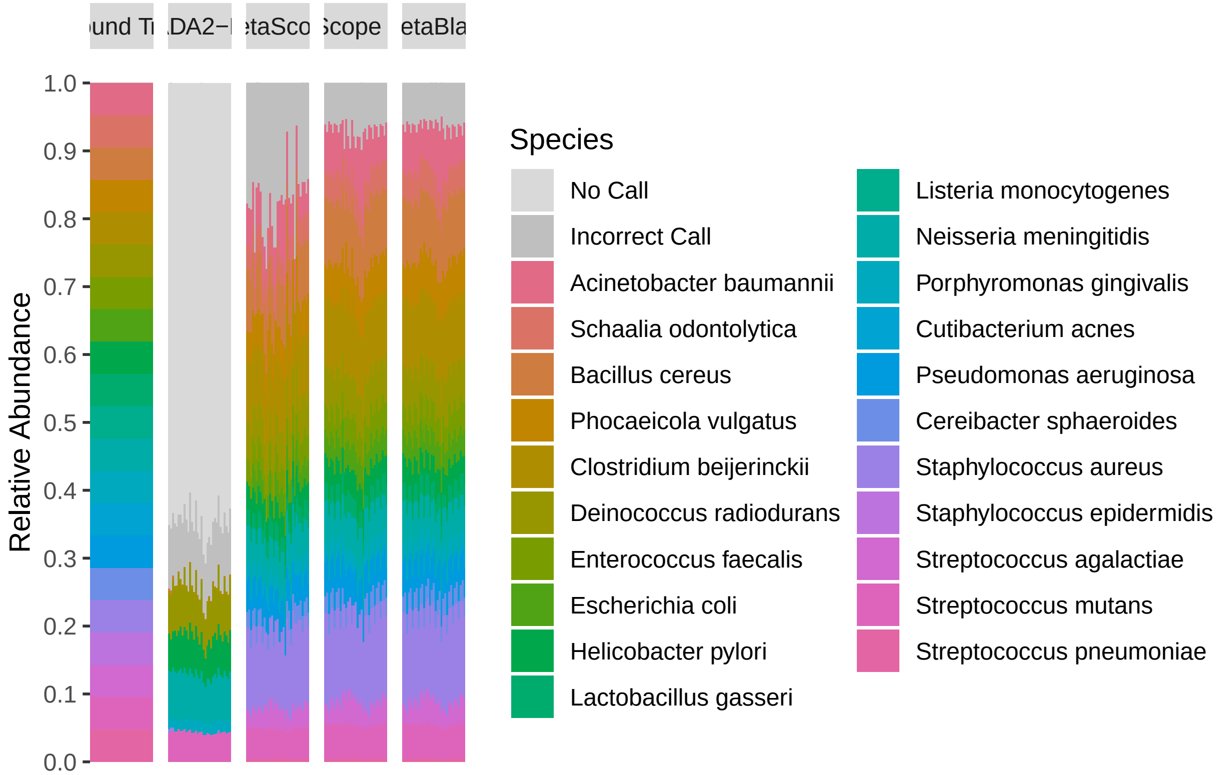
```
ms_p_df <-create_summary_df(list.files(path = "data_processed/kozich_2013/results_priors",
                             pattern = ".metascope_id.csv",
                             full.names = TRUE,
```

```r
                              recursive = TRUE),
                  pipeline = "MetaScope Priors")
ms_p_b_df <-create_summary_df(list.files(path = "data_processed/kozich_2013/results_metablast_priors_1.0
                              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",priors_df$species))
ground_truth_df <- cbind(ground_truth_df,
                         as.data.frame(do.call(cbind, replicate(33, c(0,0,rep(1/21, 21)), simplify = FAL
colnames(ground_truth_df) <-  c("Species", as.character(unique(summary_df$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)
```

```r
summary_df_dada2 <- data.frame(
  Species = c("No Call", "Incorrect Call",priors_df$species))
# Left joining with aggregate because duplicate species names
dada2_files <- list.files(path = "data_processed/kozich_2013/dada2_results",
                          full.names = TRUE)
for (i in dada2_files) {
  # Get Sample Names
  sample_name <- sub("\\..*$", "", basename(i)) |> strsplit(split = "dada2_")
  sample_name <- sample_name[[1]][2]
  # Clean MetaBlast Table
  dada2_df <- read.csv(i)
  dada2_df <- dplyr::mutate(dada2_df, Species = ifelse(is.na(Species), NA, paste0(Genus, " ", Species))
    dplyr::select(Species, reads_count) |>
    dplyr::mutate_if(is.numeric, ~ . / sum(.))
  colnames(dada2_df) <- c("Species", sample_name)
  dada2_df[dada2_df == "Phocaeicola vulgatus"] <- 'Bacteroides vulgatus'
  dada2_df[dada2_df == "Schaalia odontolytica"] <- 'Actinomyces odontolyticus'
  dada2_df[dada2_df == "Cutibacterium acnes"] <- 'Propionibacterium acnes'
  dada2_df[dada2_df == "Cereibacter sphaeroides"] <- 'Rhodobacter sphaeroides'
  dada2_df[dada2_df == "Clostridium sensu stricto 1 beijerinckii"] <- 'Clostridium beijerinckii'


  summary_df_dada2 <- dplyr::left_join(summary_df_dada2, dada2_df, by = "Species")
  summary_df_dada2[is.na(summary_df_dada2)] <- 0

  no_call.metascope <- dada2_df |>
    dplyr::filter(is.na(Species)) |>
    dplyr::select(sample_name) |>
    sum()
  correct_call.metascope <- dada2_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_dada2[1,sample_name] <- no_call.metascope
  summary_df_dada2[2,sample_name] <- incorrect_call.metascope
}

summary_df_dada2_longer <- tidyr::pivot_longer(
  summary_df_dada2,
  cols = c(2:34),
  values_to = "prop"
)

summary_df_dada2_longer$pipeline <- "DADA2-NB"

summary_df <- rbind(summary_df, summary_df_dada2_longer)
```

Plotting Relative Abundance of Mock Microbiome



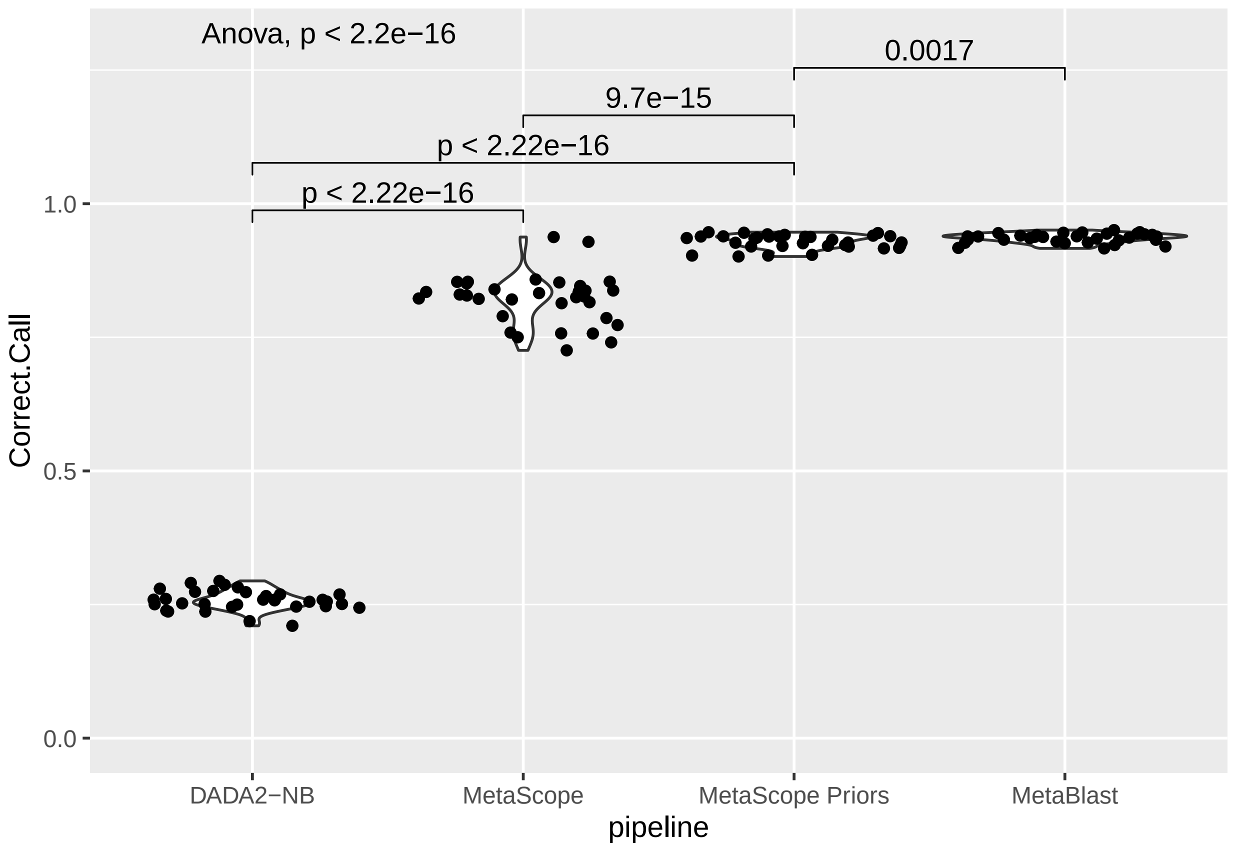
```
## Warning: fonts used in ‘flextable‘ are ignored because the ‘pdflatex‘ engine is
## used and not ‘xelatex‘ or ‘lualatex‘. You can avoid this warning by using the
## ‘set_flextable_defaults(fonts_ignore=TRUE)‘ command or use a compatible engine
## by defining ‘latex_engine: xelatex‘ in the YAML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

Profiler	Correct Call	No Call	Incorrect Call
DADA2-NB	0.258 ± 0.019	0.65 ± 0.023	0.092 ± 0.005
MetaScope	0.821 ± 0.047	0 ± 0	0.179 ± 0.047
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
```



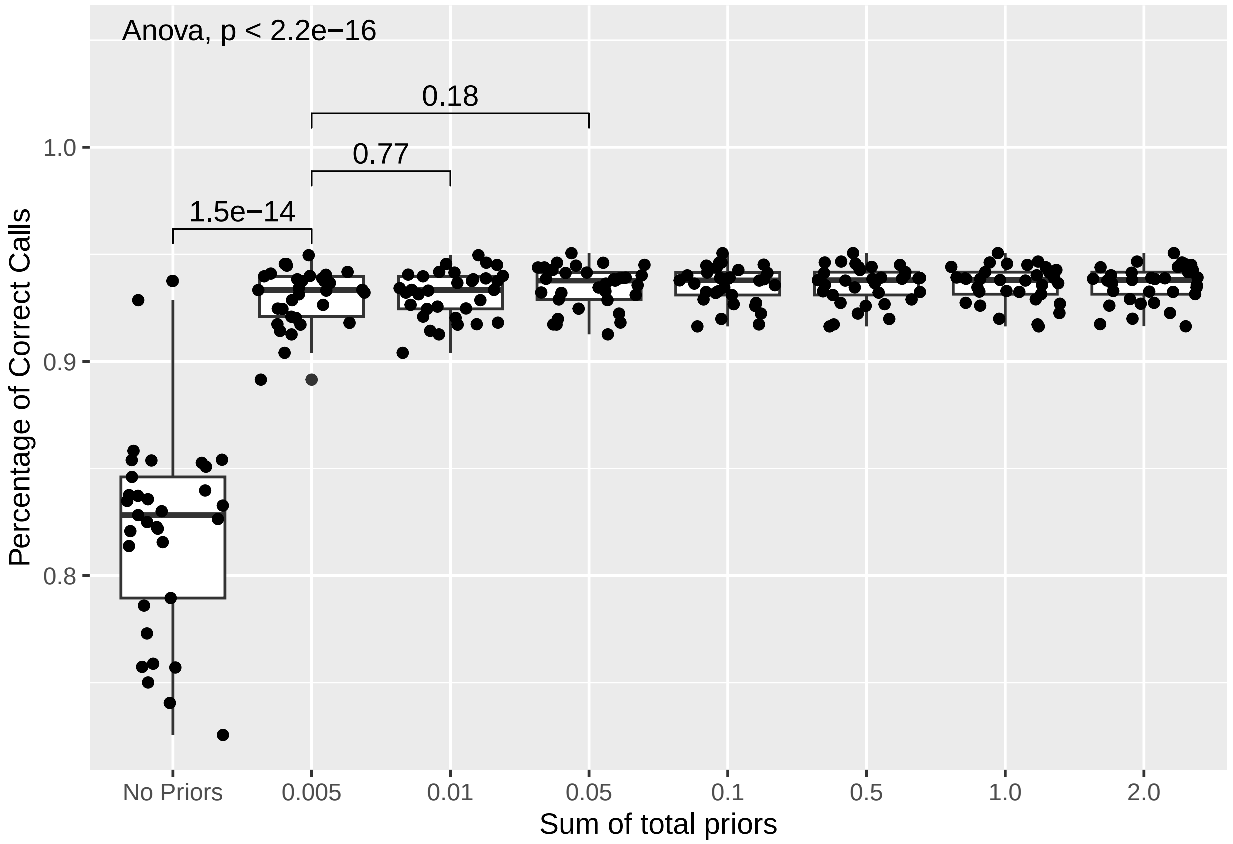
Sensitivity Analysis

```
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",
```

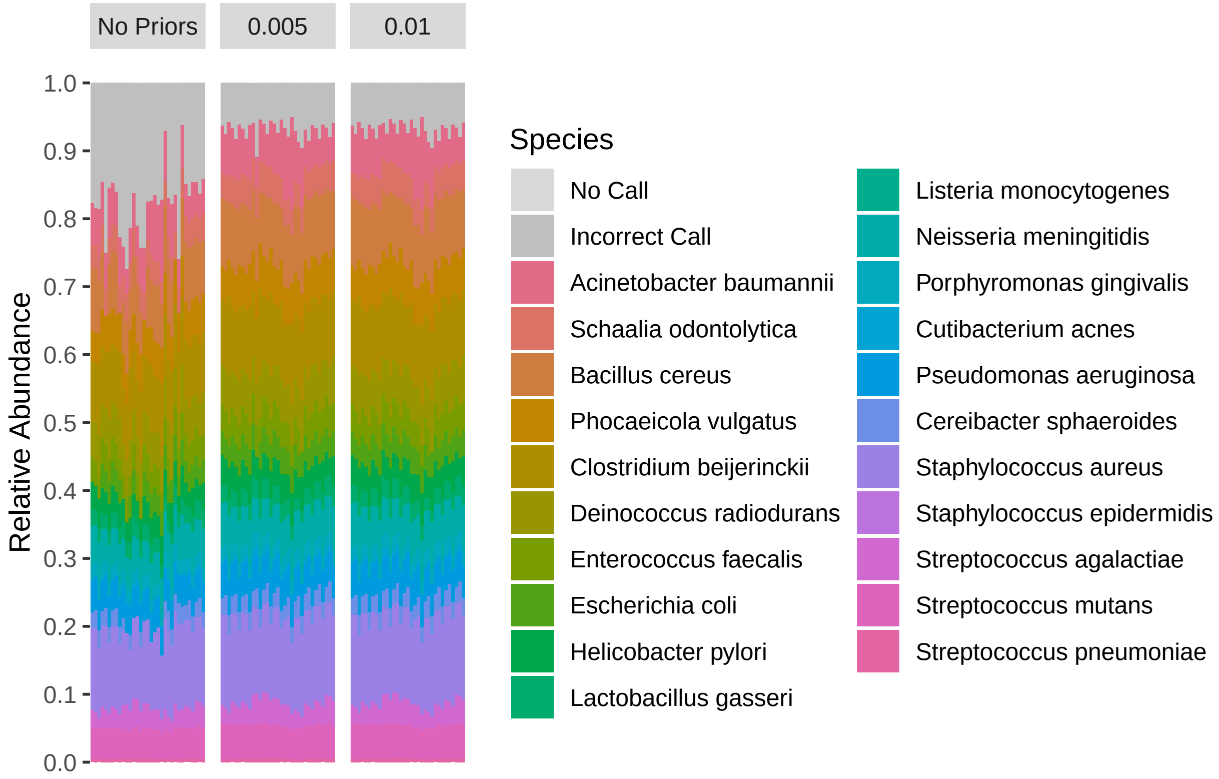
```r
    pattern = ".metascope_id.csv",
    full.names = TRUE,
    recursive = TRUE),
  pipeline_name = "0.01"
)

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.
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)
```



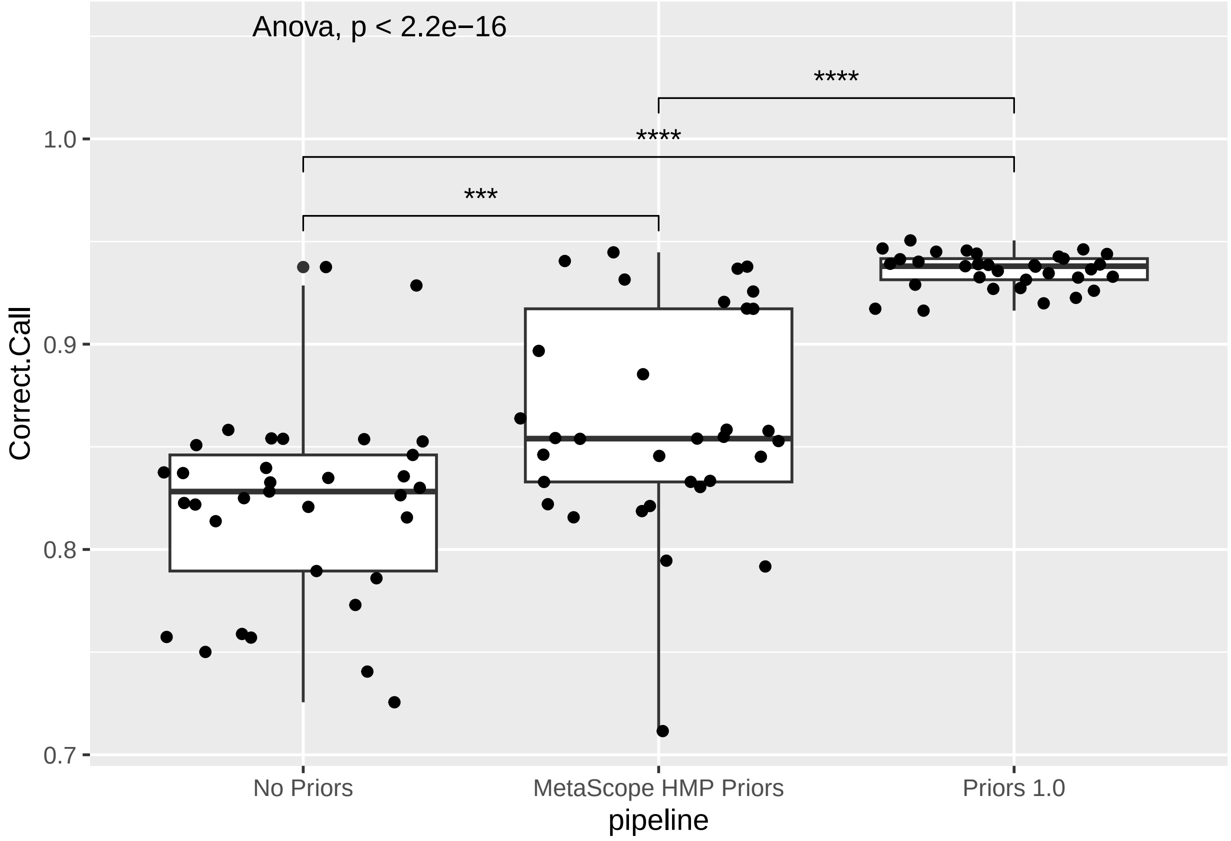
#HMP priors

```
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 = 1.05, 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,
                     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"))
```


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