Symbiont density analysis for POC TPC

#Title: POC TPC #Project: Moorea #Author: HM Putnam #Edited by: DM Becker-Polinski #Date Last Modified: 20220818 #See Readme file for details

Import data

```
# Cell count data
sym_counts <- read_csv("baseline_TPC_physio/data/TPC_sym_counts/TPC_POC_symbiont_counts.csv")
```

```
## Rows: 32 Columns: 17
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): fragment_ID, timepoint, Counter
## dbl (12): Squares.Counted, Count1, Count2, Count3, Count4, Count5, Count6, C...
## lgl  (2): Dilution, Date.Counted
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Surface area data
sa <- read.csv("baseline_TPC_physio/output/surface.area.calc.csv")

# Tissue homogenate volume data
homog_vols <- read_csv("baseline_TPC_physio/data/TPC_homogenate_vols/homogenate_vols.csv") %>% select(1
```

```
## Rows: 32 Columns: 5
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): fragment_ID, initials, note
## dbl (2): homog_vol_ml, date_airbrushed
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Coral sample metadata
metadata <- read_csv("baseline_TPC_physio/metadata_POC_TPC.csv") %>% select(1:4)
```

```
## Rows: 32 Columns: 4
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (3): fragment_ID, site, species
## lgl (1): date.TPC
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Join homogenate volumes and surface area with sample metadata
metadata <- full_join(metadata, homog_vols)  %>%
  full_join(sa)
```

```
## Joining, by = "fragment_ID"
```

```
## Joining, by = "fragment_ID"
```

Calculate cells per square centimeter

```
# Calculate mean counts for each sample
sym_counts <- sym_counts %>%
  select(fragment_ID, Squares.Counted, matches("Count[0-9]")) %>%
  gather("rep", "count", -fragment_ID, -Squares.Counted) %>%
  group_by(fragment_ID, Squares.Counted) %>%
  summarise(mean_count = mean(count, na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'fragment_ID'. You can override using the `.groups` argument.
```

```
# Join mean counts with sample metadata
sym_counts <- full_join(sym_counts, metadata)
```

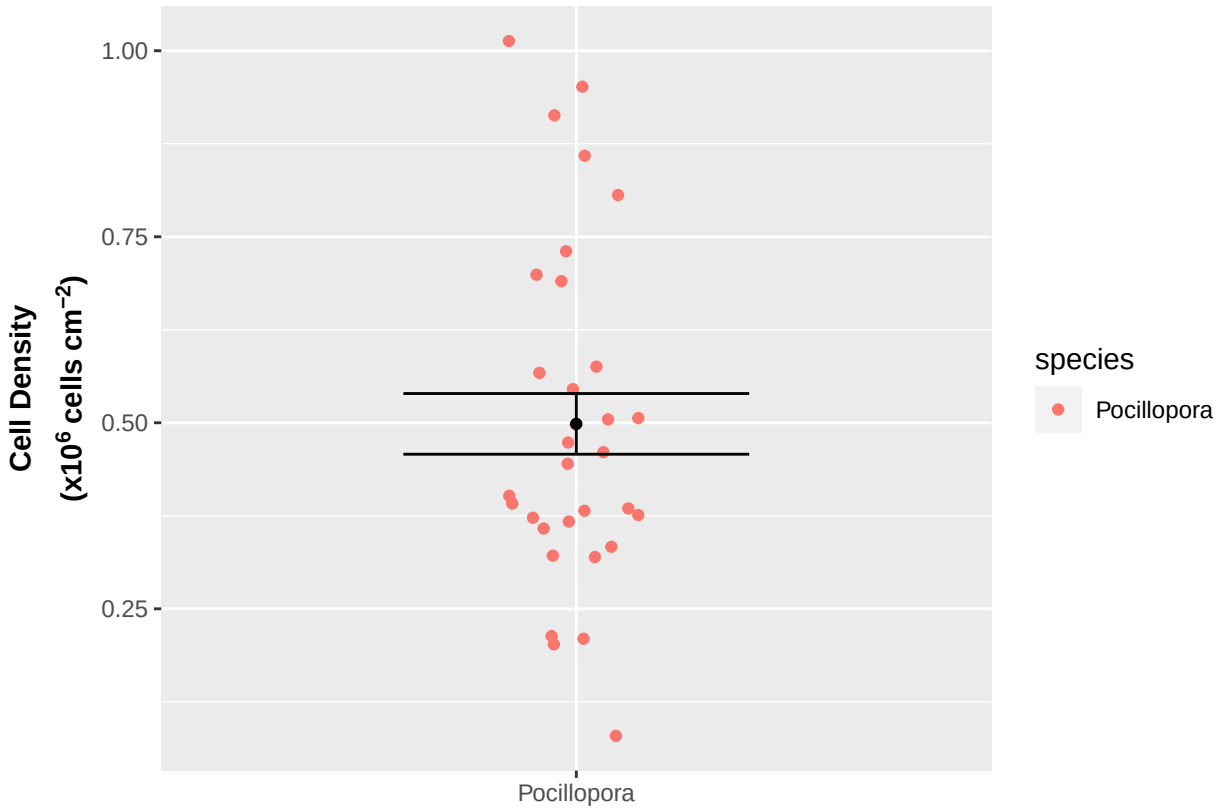
```
## Joining, by = "fragment_ID"
```

```
# Normalize counts by homogenat volume and surface area
sym_counts <- sym_counts %>%
  mutate(cells.mL = mean_count * 10000 / Squares.Counted,
         cells = cells.mL * homog_vol_ml,
         cells.cm2 = cells / surface.area.cm2)
```

Plot data

```
sym_counts %>%
  #filter(!is.na(site)) %>%
  ggplot(aes(x = species, y = cells.cm2 / 10^6, color = species)) +
  labs(x = "") +
  ylab(expression(bold(paste(atop("Cell Density", "(" *x*"10"^"6" *~cells *~ cm^"-2"*)"))))) + #using
  #facet_wrap(~ species) +
geom_jitter(width = 0.1) +                                          # Plot all points
  stat_summary(fun.data = mean_cl_normal, fun.args = list(mult = 1),    # Plot standard error
               geom = "errorbar", color = "black", width = 0.5) +
  stat_summary(fun.y = mean, geom = "point", color = "black")           # Plot mean
```

```
## Warning: `fun.y` is deprecated. Use `fun` instead.
```



Output data to file.

```
sym_counts %>%
  select(fragment_ID, species, cells.cm2) %>%
  mutate(timepoint="MAY")%>%
  write_csv(path = "baseline_TPC_physio/output/sym_counts.csv")
```

```
## Warning: The `path` argument of `write_csv()` is deprecated as of readr 1.4.0.
## Please use the `file` argument instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
```