Figure 4

Brittany A. Petros

2024-03-02

Set up the workspace.

```
rm(list=ls(all.names=TRUE))
setwd("/Users/bpetros/Desktop/PHIS")
libs <- c("cowplot", "ggplot2", "ggsignif", "lubridate", "scales", "tidyverse")
invisible(lapply(libs, function(x) suppressPackageStartupMessages(library(x, character.only = TRUE))))
options(stringsAsFactors=FALSE, scipen = 999)
```

Read in the cleaned input files.

```
# read in cleaned pt data
pt <- read.csv("cleaned/rsv_patient.csv")
pt$Date = ymd(pt$Discharge_Date)
```

This function takes a df with column "Admit_Age_In_Days" (integer) as input. It creates a column "Age_Group" that labels rows as belonging to one of the following age groups: 0-3 months, 3-6 months, 6-12 months, 1-2 years, 2-4 years, and >= 5 years.

```
add_age_group <- function(df) {
  df$Age_Group <- cut(
    df$Admit_Age_In_Days,
    breaks = c(0, 90, 180, 365, 730, 1460, Inf),
    labels = c("0-3 months", "3-6 months", "6-12 months", "1-2 years", "2-4 years", "5-17 years"),
    right = FALSE)
  return(df)}
```

This function takes as input a df with columns "col_name" (integer), "Age_Group" (character), and "phase" (character), with each row representing one patient encounter. It determines the proportion of patients for which col_name equals 1 (i.e., the outcome was experienced) for each Age_Group in each phase. It identifies differences in the proportion of patients of a particular Age_Group experiencing the outcome across phases using the two-proportion z-test. Finally, it plots the proportion of patients in each Age_Group and in each phase experiencing the outcome, with ylabel (character) labeling the y-axis and ypos_signif (numeric) providing the y-coordinates for significance bars.

```
calc_age_prop <- function(df, col_name, ylabel, ypos_signif) {
  # create df with proportion of pts in each age group and phase experiencing outcome
    prop <- df %>%
    group_by(Age_Group, phase) %>%
    summarise(tot = n(),
              proportion = mean({{col_name}}),
```

```
            success = sum({{col_name}}),
            std_error = sqrt(proportion * (1 - proportion) / n()),
            .groups = 'drop') %>%
  group_by(phase) %>%
  mutate(x_pos = as.numeric(factor(Age_Group)) + as.numeric(factor(phase)) * 1)

# calc p-vals for difference in outcome proportion across phases
pvals <- prop %>%
  filter(phase == "Pre-Pandemic") %>%
  group_by(Age_Group, .add = TRUE) %>%
  group_split() %>%
  map_dbl(~{
    group1 <- filter(prop, Age_Group == .x$Age_Group, phase == "Pre-Pandemic")$proportion
    group2 <- filter(prop, Age_Group == .x$Age_Group, phase == "Post-Emergence")$proportion
    test_result <- prop.test(c(sum(group1 * .x$tot), sum(group2 * .x$tot)), c(sum(.x$tot), sum(.x$tot)
    test_result$p.value})
prop <- merge(data.frame(Age_Group = unique(prop$Age_Group), p_value = pvals),
              prop, by = "Age_Group")
prop$phase = factor(prop$phase, levels = c("Pre-Pandemic", "Post-Emergence"))

# calc percent decrease in outcome proportion from pre-pandemic to post-emergence phase
prct <- prop %>%
  select(Age_Group, phase, proportion) %>%
  pivot_wider(names_from = phase, values_from = proportion) %>%
  mutate(percent_decrease = ((`Pre-Pandemic` - `Post-Emergence`) / `Pre-Pandemic`) * 100) %>%
  select(Age_Group, percent_decrease) %>%
  mutate(Age_Group = factor(Age_Group, levels = c("0-3 months", "3-6 months", "6-12 months",
                                                  "1-2 years", "2-4 years", "5-17 years"))) %>%
  arrange(Age_Group)

# plot the results
fig <- ggplot(prop, aes(x = x_pos, y = proportion, color = phase)) +
  geom_point(position = position_dodge(width = 1), size = 2) +
  geom_errorbar(aes(ymin = proportion - 1.96 * std_error,
                    ymax = proportion + 1.96 * std_error),
                position = position_dodge(width = 1), width = 0.3) +
  coord_cartesian(ylim = c(floor(min(prop$proportion)*10)/10, ceiling(max(prop$proportion)*10)/10)) +
  scale_y_continuous(breaks = seq(from = floor(min(prop$proportion)*10)/10,
                                  to = ceiling(max(prop$proportion)*10)/10, by = 0.1)) +
  scale_color_manual(values = c("black", "darkmagenta"), labels = c("Pre-Pandemic", "Post-Emergence"))
  scale_x_continuous(breaks = unique(prop$x_pos), labels = unique(prop$Age_Group)) +
  labs(x = "Age Group", y = ylabel, color = "") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1), legend.position = "top") +
  geom_signif(y_position = ypos_signif,
              xmin = sort(unique(prop$x_pos) - 0.25),
              xmax = sort(unique(prop$x_pos) + 0.25),
              annotation = c("***", "***", "***", "***", "***", "***"),
              tip_length = 0.01, size = 0.6)

# return data and figure
return(list(prct = prct, plot = fig))}
```

Add columns “Age_Group” and “phase” to the pt df.

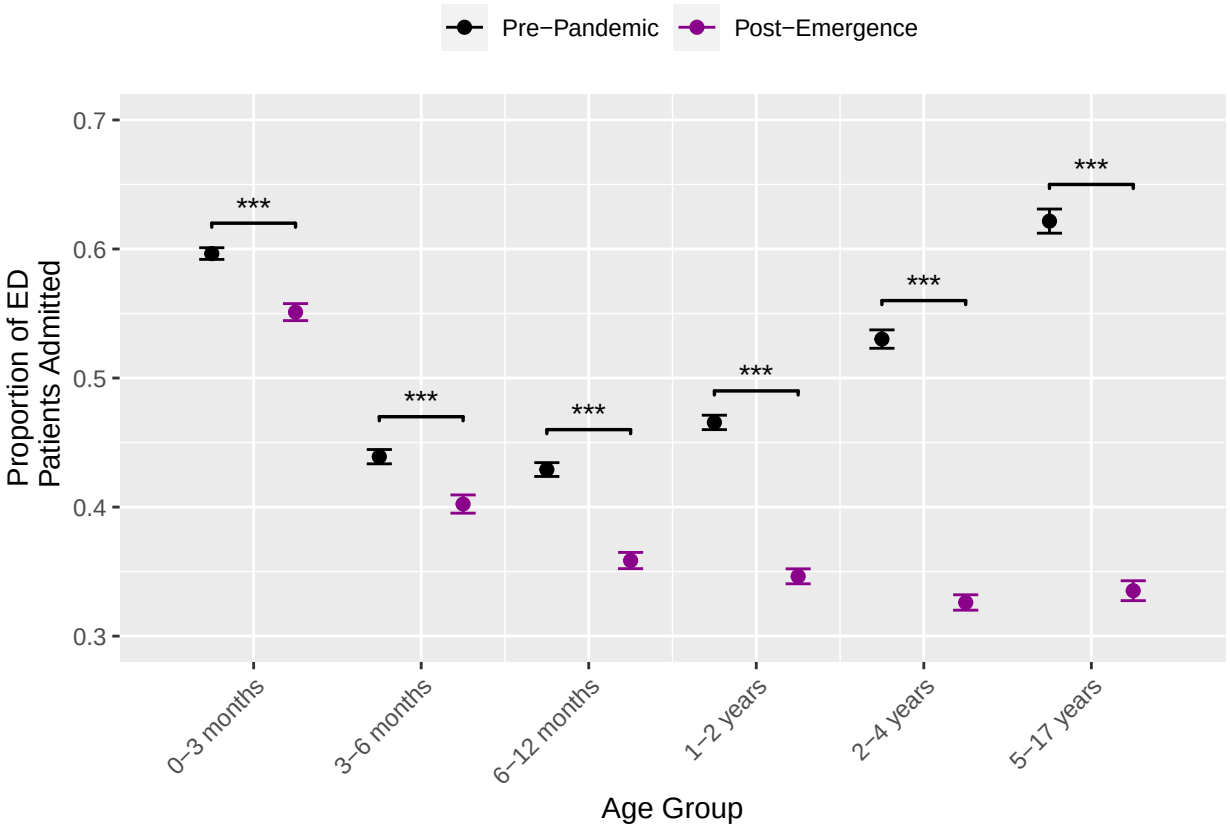
```r
# add age groups to pt df
pt = add_age_group(pt)

#add phase to pt df
pt = pt %>% filter(Date < as.Date("2020-04-01") | Date >= as.Date("2021-04-01"))
pt$phase = ifelse(pt$Date < as.Date("2020-04-01"), "Pre-Pandemic", "Post-Emergence")
```

Calculate the proportion of ED patients requiring hospitalization in each phase and in each age group (Figure 4A).

```r
# filter for pts seen in the ED and create ed_admit column
ed_prop = pt %>%
  filter(ED_entry == 1) %>%
  mutate(ed_admit = ifelse(ED_Dispo == "ED_Admission", 1, 0))

# run calc_age_prop function
ed_results = calc_age_prop(ed_prop, ed_admit, "Proportion of ED\n Patients Admitted",
                           c(0.62, 0.47, 0.46, 0.49, 0.56, 0.65))
ed_results$prct
```

```
## # A tibble: 6 x 2
##   Age_Group   percent_decrease
##   <fct>                  <dbl>
## 1 0-3 months              7.61
## 2 3-6 months              8.36
## 3 6-12 months            16.4
## 4 1-2 years              25.6
## 5 2-4 years              38.5
## 6 5-17 years             46.1
```

```r
ed_results$plot
```



Calculate the proportion of inpatients requiring intensive care in each phase and in each age group (Figure 4B).

```
# filter for inpatients and create icu column
icu_prop = pt %>%
  filter(Patient_Type_Title != "ED Visit") %>%
  mutate(icu = ifelse(ICU_Flag == "Y" | NICU_Flag == "Y", 1, 0))

# run calc_age_prop function
icu_results = calc_age_prop(icu_prop, icu, "Proportion of Inpatients\n Admitted to ICU or NICU",
                            c(0.395, 0.33, 0.30, 0.305, 0.31, 0.365))
icu_results$prct
```

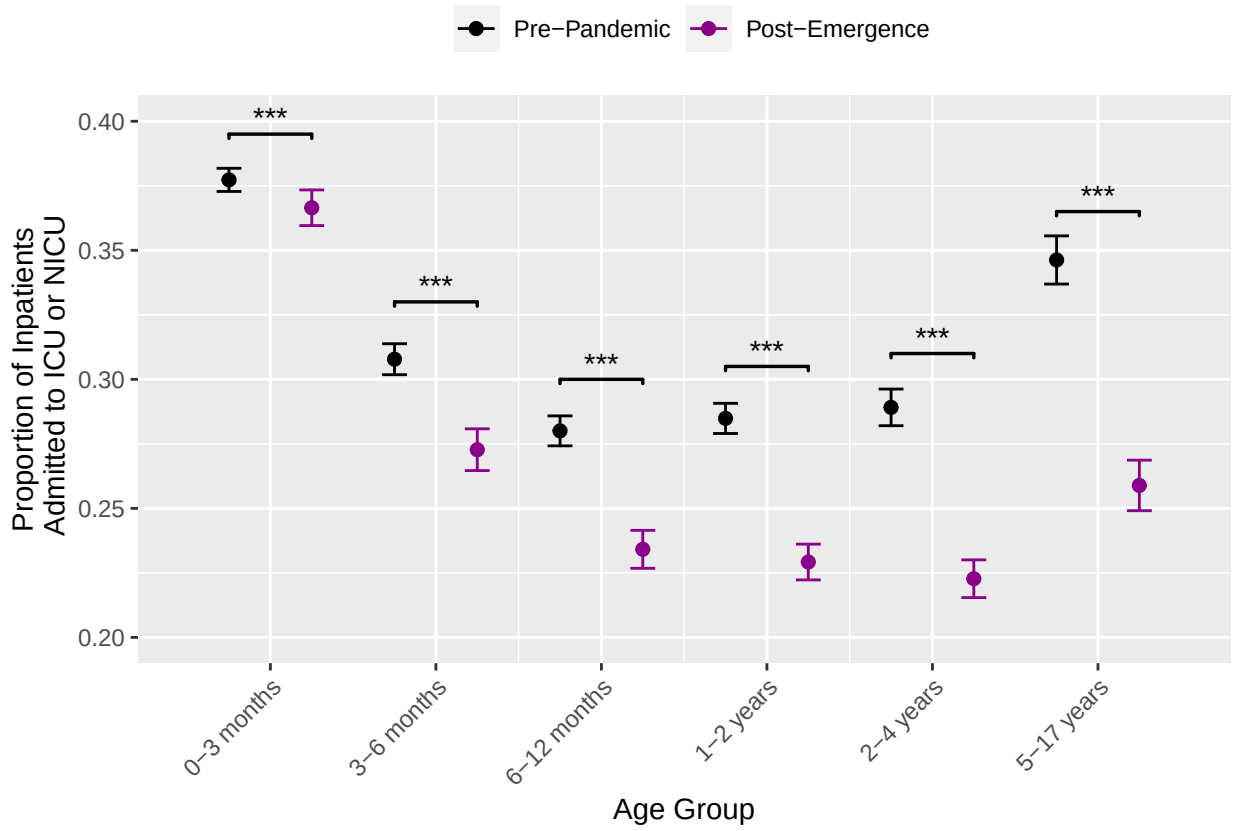
```
## # A tibble: 6 x 2
##   Age_Group   percent_decrease
##   <fct>                  <dbl>
## 1 0-3 months              2.86
## 2 3-6 months             11.4
## 3 6-12 months            16.4
## 4 1-2 years              19.5
## 5 2-4 years              23.0
## 6 5-17 years             25.2
```

```
fig4b = icu_results$plot + scale_y_continuous(breaks = seq(from = 0.2, 0.4, by = 0.05))
```

```
## Scale for y is already present.
```

```
## Adding another scale for y, which will replace the existing scale.
```

```
fig4b
```



Calculate the proportion of inpatients requiring mechanical ventilation in each phase and in each age group (Figure 4C).

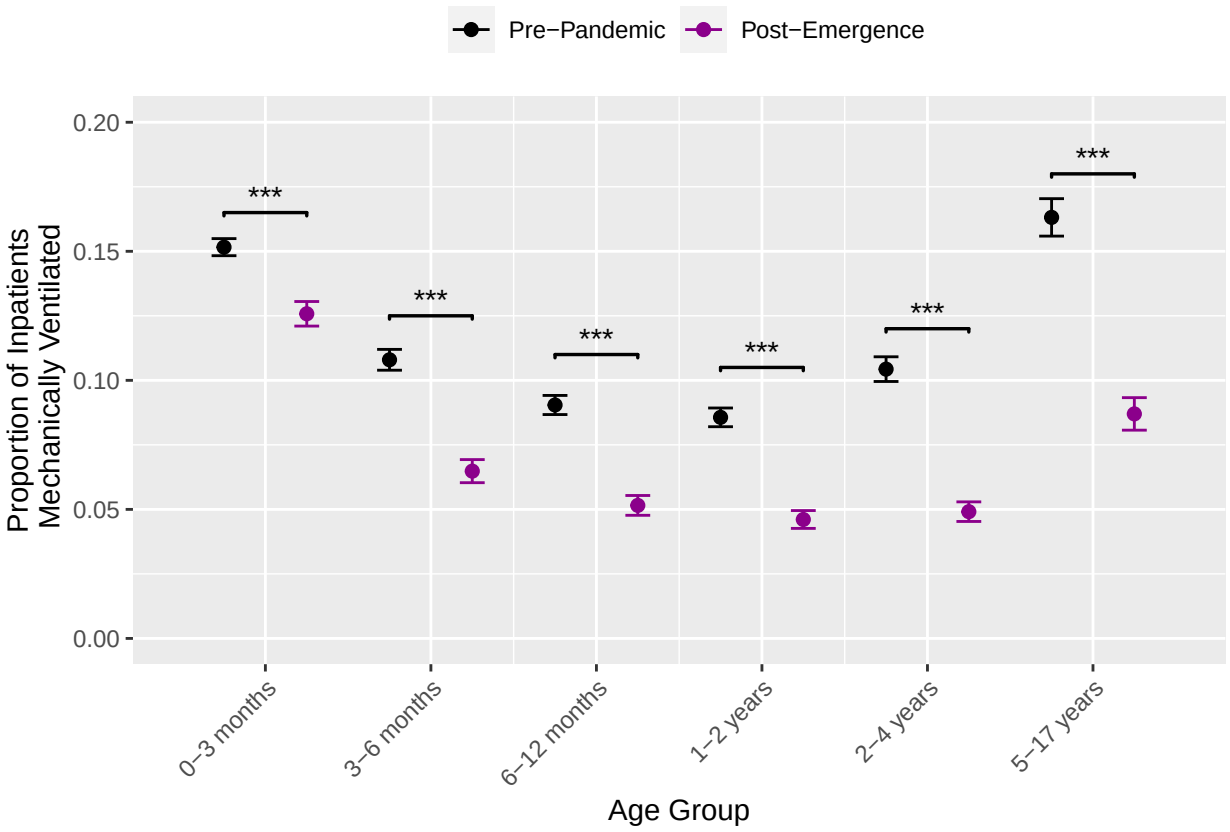
```
# filter for inpatients and create mv column
mv_prop = pt %>%
  filter(Patient_Type_Title != "ED Visit") %>%
  mutate(mv = ifelse(Mechanical_Vent_Flag == "Y", 1, 0))

# run calc_age_prop function
mv_results = calc_age_prop(mv_prop, mv, "Proportion of Inpatients\n Mechanically Ventilated",
                           c(0.165, 0.125, 0.11, 0.105, 0.12, 0.18))
mv_results$prct
```

```
## # A tibble: 6 x 2
##   Age_Group   percent_decrease
##   <fct>                  <dbl>
## 1 0-3 months              17.0
## 2 3-6 months              40.0
## 3 6-12 months             43.0
## 4 1-2 years               46.2
## 5 2-4 years               52.9
## 6 5-17 years              46.7
```

```
fig4c = mv_results$plot + scale_y_continuous(breaks = seq(from = 0, 0.2, by = 0.05))
```

```
## Scale for y is already present.
## Adding another scale for y, which will replace the existing scale.
```

```
fig4c
```



Generate Figure 4.

```
fig4 <- plot_grid(ed_results$plot, fig4b, fig4c, ncol = 1, labels = c("A", "B", "C"))
ggsave("figs/fig4.pdf", plot = fig4, width = 4, height = 12)
```