Time series analysis with updated window definition

Cheyenne Ehman, Ziyan Zhu

3/29/2021

Select varible of interests

```
source("step2_data_wrangle.R")
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
##
## Attaching package: 'reshape2'
## The following object is masked from 'package:tidyr':
##
##     smiths
## `summarise()` regrouping output by 'county', 'studentmaskpolicy' (override with `.groups` argument)
## `summarise()` regrouping output by 'county' (override with `.groups` argument)
## `summarise()` regrouping output by 'county' (override with `.groups` argument)
## `summarise()` regrouping output by 'county', 'opendategrouped' (override with `.groups` argument)
## `summarise()` regrouping output by 'county' (override with `.groups` argument)
```

```
################### school reopen dates ###################
district_policies <- OH_K12 %>%
  distinct(county,county_enroll,leaid,district_enroll,schooltemporaryshutdown,opendategrouped,teachingm
```

Calculate the proportion and generate date brackets

```
prop_opendate <- district_policies%>%
  filter(!schooltemporaryshutdown %in% c('Closed indefinitely','Pending','Unknown'))%>%
  group_by(county,county_enroll,opendategrouped)%>%
  summarise(n_opendate = sum(district_enroll))%>% # number of students under certain date for each coun
  mutate(prop_opendate = round(n_opendate/county_enroll,2))%>% # proportion
  group_by(county)%>%
  #filter(prop_opendate>0.6)%>%
  slice(which.max(prop_opendate))%>% # filter large proportions of students with same reopen dates #can
```

```
  mutate(threeweeks_lag_open = opendategrouped+21,sixweeks_lag_open = opendategrouped+42,twomonths_lag_
  select(-n_opendate)
```

```
## `summarise()` regrouping output by 'county', 'county_enroll' (override with `.groups` argument)
```

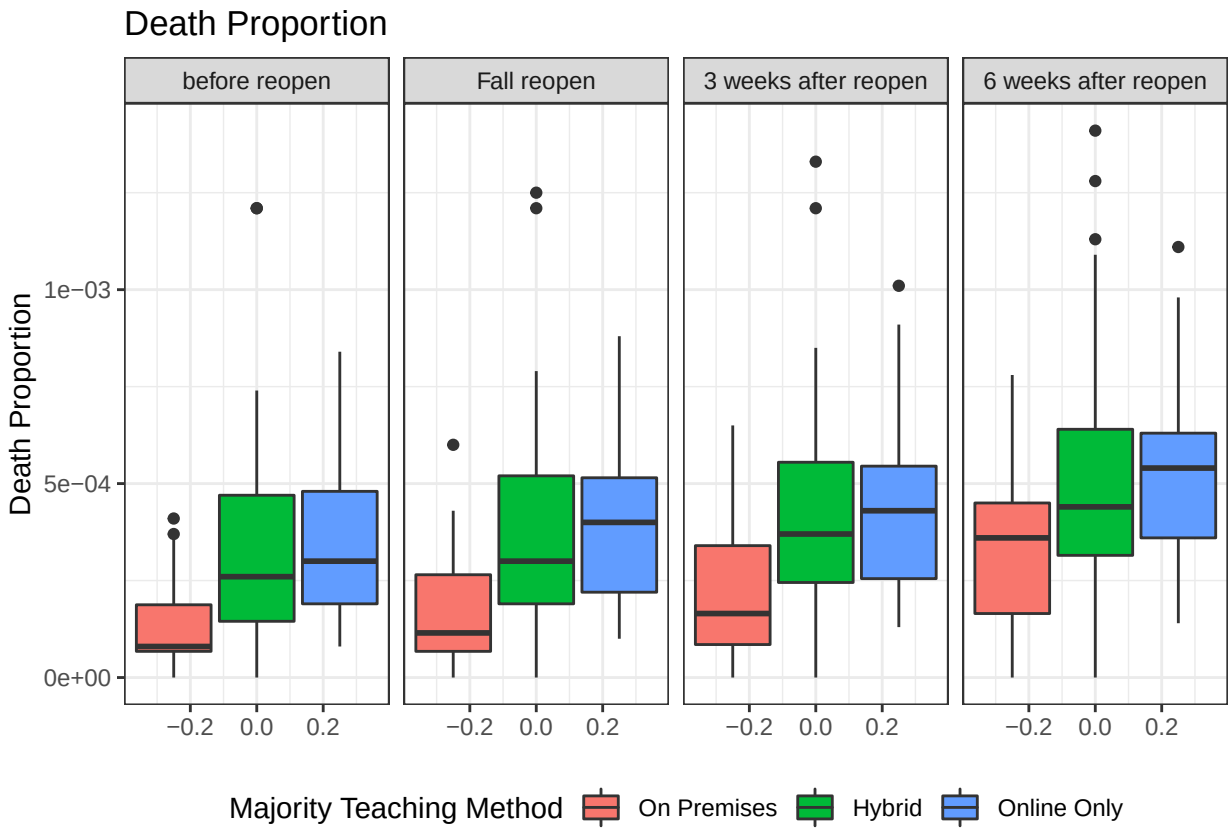
```
opendate_cases <- cases%>%
  inner_join(prop_opendate,by=c('COUNTY'='county'))%>%
  group_by(COUNTY)%>%
  filter(DATE>=opendategrouped & DATE<=beforechristmas)%>%
  group_by(COUNTY)%>%
  mutate(window_id = case_when(DATE>=opendategrouped & DATE<threeweeks_lag_open~"fall_reopento21d",
    DATE>=threeweeks_lag_open & DATE<sixweeks_lag_open~'reopen_21dto42d',
    DATE>= sixweeks_lag_open & DATE<twomonths_lag_open ~ 'reopen_42dto63d',
    TRUE ~ 'before_christmas'
  ))%>%
  select(-STATE,-STUSAB,-ST_LAT,-ST_LONG,-STATEFP,-GNISID,-UID,-CODE3)%>%
  mutate(death_prop = round(CUMDEATHS/POPULATION,5),
         window_id = as.factor(window_id))%>%
  left_join(wide_teaching_enroll,by=c('COUNTY'='county','county_enroll'))
# select the start date and end date data for each window of time

start_of_window <- opendate_cases%>%
  group_by(COUNTY,window_id)%>%
  arrange(DATE)%>%
  filter(row_number()==1)%>%
  ungroup()%>%
  mutate(y_label = case_when(window_id == "fall_reopento21d"~'before reopen',
    window_id == "reopen_21dto42d" ~ 'Fall reopen',
    window_id == "reopen_42dto63d" ~ '3 weeks after reopen',
    window_id == "before_christmas" ~ '6 weeks after reopen'
  ))%>%
  select(-opendategrouped,-threeweeks_lag_open,-sixweeks_lag_open,-twomonths_lag_open,-beforechristmas,
```

```
start_of_window$y_label <- factor(start_of_window$y_label,
         levels = c("before reopen",
         "Fall reopen",
         "3 weeks after reopen",
         "6 weeks after reopen"))

start_of_window$major_teaching <- factor(start_of_window$major_teaching,
         levels = c("On Premises",
         "Hybrid",
         "Online Only"))

start_of_window %>%
  ggplot(aes(y = death_prop,
             fill = major_teaching))+
  geom_boxplot(na.rm = T) +
  facet_grid(~y_label)+
  theme_bw()+
  labs(y = "Death Proportion",
       fill = "Majority Teaching Method",
       title = "Death Proportion")+theme(legend.position = "bottom")
```




```
#ggsave("Deathprop_box.jpg", width = 8.5, height = 5)

require(scales)
```

```
## Loading required package: scales
```

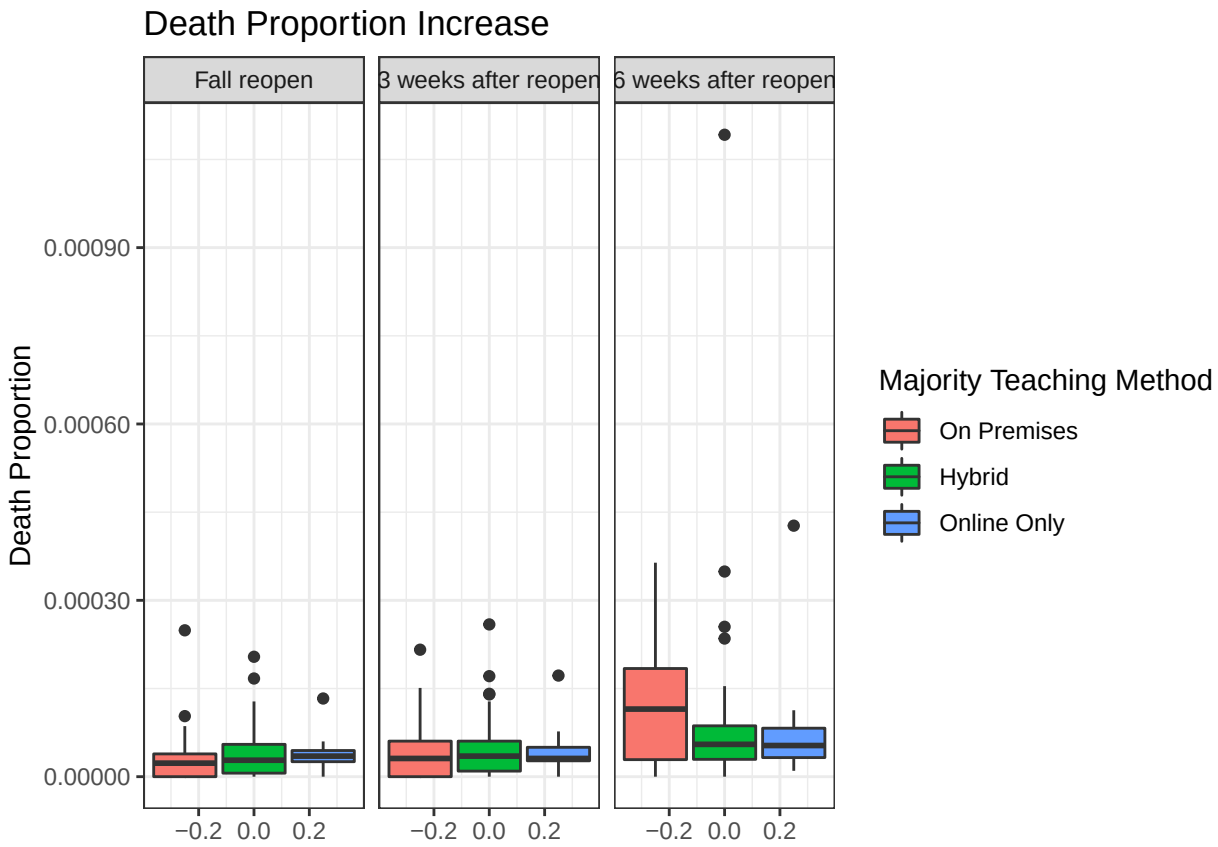
```
##
## Attaching package: 'scales'
```

```
## The following object is masked from 'package:purrr':
##
##     discard
```

```
## The following object is masked from 'package:readr':
##
##     col_factor
```

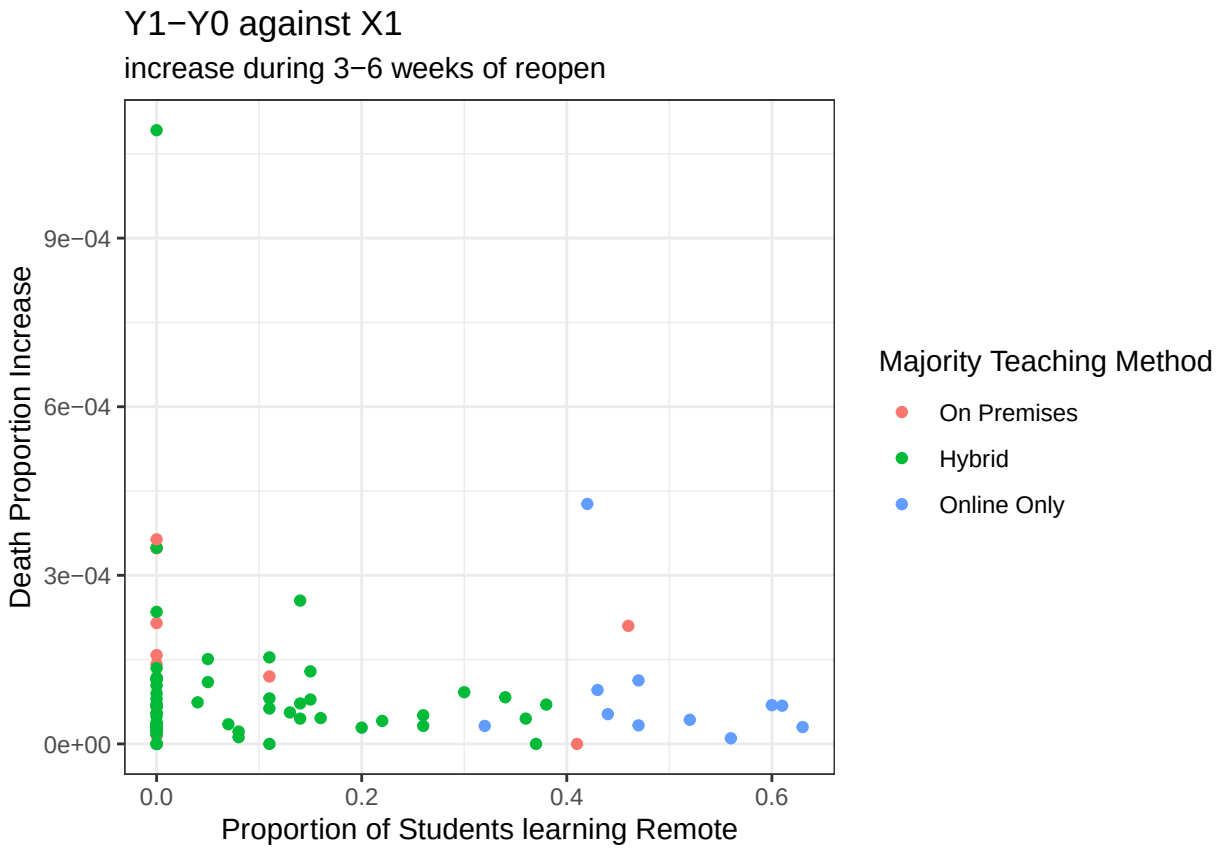
```
start_of_window %>%
  group_by(COUNTY)%>%
  mutate(death_inc = CUMDEATHS-lag(CUMDEATHS))%>%
  drop_na()%>%
  mutate(death_prop_inc = round(death_inc/POPULATION,6))%>%
  ggplot(aes(y = death_prop_inc,
             fill = major_teaching))+
  geom_boxplot(na.rm = T) +
  theme_bw()+
  labs(y = "Death Proportion",
       fill = "Majority Teaching Method",
       title = "Death Proportion Increase") +
  facet_grid(~y_label)+
```

```
  scale_y_continuous(labels = comma)
```




```
#ggsave("Deathpropinc_box.jpg", width = 8.5, height = 5)
```

```
start_of_window %>%
  group_by(COUNTY)%>%
  mutate(death_inc = CUMDEATHS-lag(CUMDEATHS))%>%
  drop_na()%>%
  mutate(death_prop_inc = round(death_inc/POPULATION,6))%>%
  filter(y_label == "6 weeks after reopen") %>%
  ggplot(aes(x = Online_Only, y = death_prop_inc, color = major_teaching)) +
  geom_point()+
  theme_bw() +
  labs(x = "Proportion of Students learning Remote",
       y = "Death Proportion Increase",
       title = "Y1-Y0 against X1",
       subtitle = "increase during 3-6 weeks of reopen",
       color = "Majority Teaching Method")
```



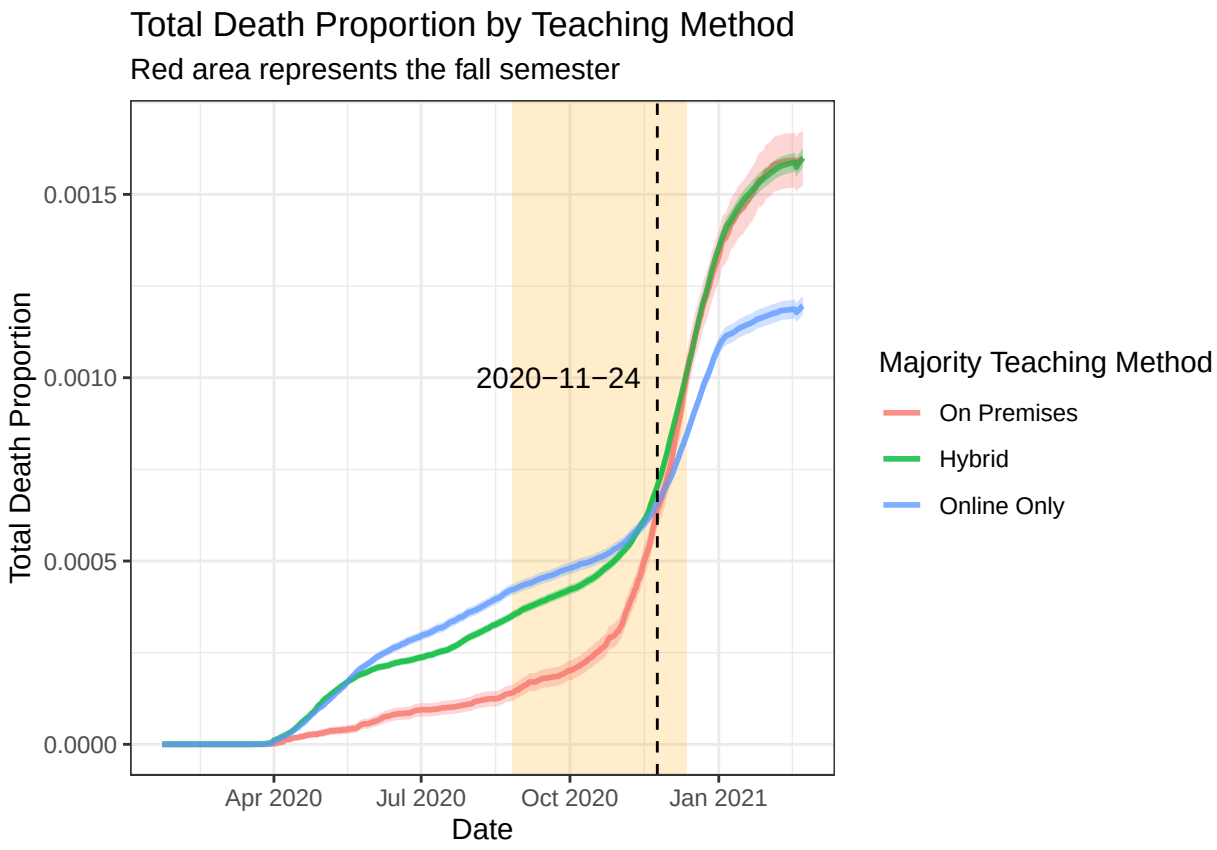

```
ggsave("y1x1.jpg", width = 7, height = 5)
```

```
county_policy_wide$major_teaching <- factor(county_policy_wide$major_teaching,
         levels = c("On Premises",
         "Hybrid",
         "Online Only"))
# see when the intesection happens
date.intercept <- as.Date("2020-11-24")

# add 95% confidence bans
confidence_level <- .95
z_cl <- qnorm(confidence_level)


cases %>%
  left_join(county_policy_wide[,c("county","major_teaching")],
            by = c("COUNTY" = "county")) %>%
  na.omit() %>%
  group_by(DATE, major_teaching) %>%
  summarise(total_deaths = sum(CUMDEATHS),
            total_pop = sum(POPULATION),
            death_prop = total_deaths/total_pop,
            death_prop_upper = death_prop + z_cl*sqrt(death_prop*(1 - death_prop)/total_pop),
            death_prop_lower = death_prop - z_cl*sqrt(death_prop*(1 - death_prop)/total_pop),
            .groups = "drop") %>%
  ggplot(aes(x = DATE, y = death_prop, group = major_teaching))+
    geom_rect(data=opendate_cases[1,],
```

```
          aes(xmin=as.Date("2020/08/26"), xmax=as.Date("2020/12/12"),
              ymin=-Inf,ymax=Inf),
          color = NA,alpha=0.2, show.legend = F, fill = "orange") +
  geom_line(aes(color = major_teaching),size = 1, alpha = .8) +
  geom_ribbon(aes(ymin = death_prop_lower, ymax = death_prop_upper,
                  fill= major_teaching),
              alpha = .3, show.legend = F)+
  geom_vline(xintercept = date.intercept, linetype = "dashed") +
  annotate("text",x = date.intercept,y = .001,
           label = date.intercept,
           hjust = 1.1) +
  theme_bw() +
  labs(x = "Date", y = "Total Death Proportion",
       title = "Total Death Proportion by Teaching Method",
       subtitle = "Red area represents the fall semester",
       color = "Majority Teaching Method")
```

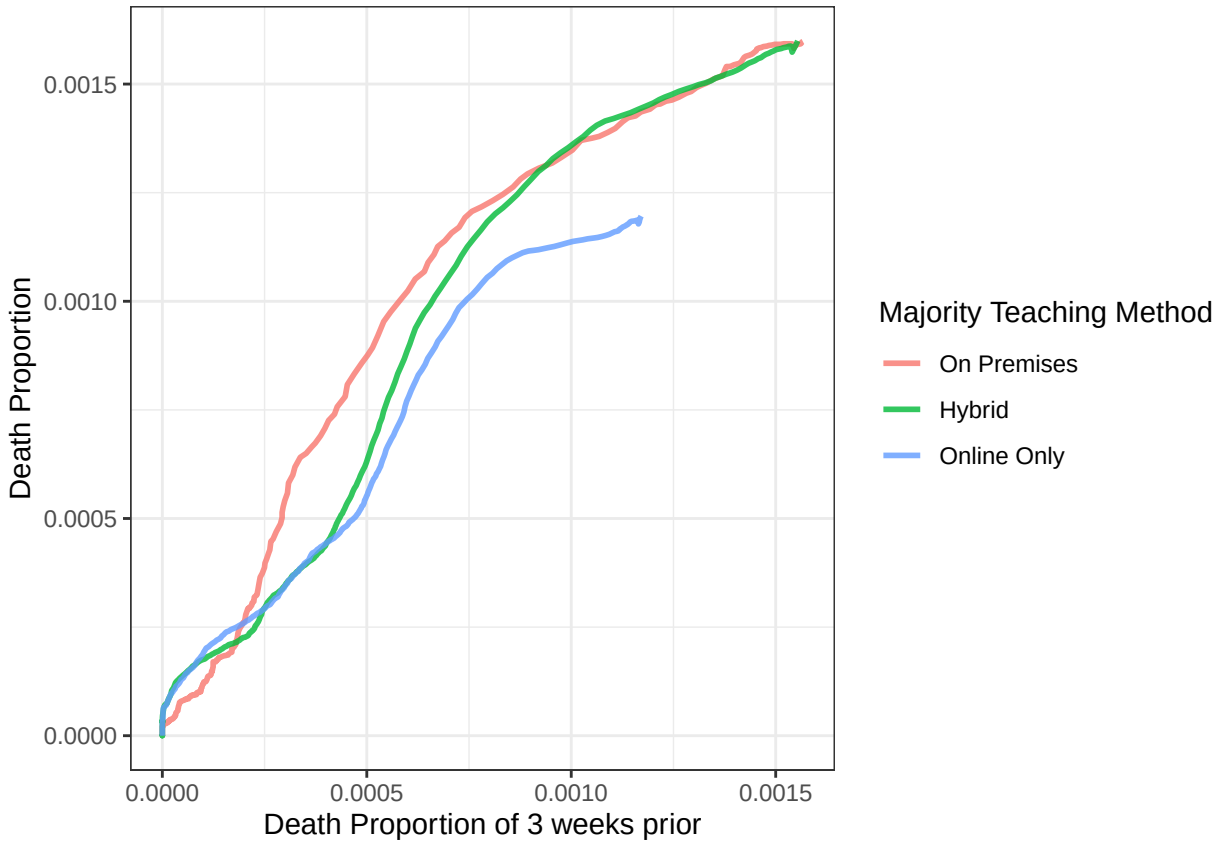



```
# +
#   scale_x_date(date_breaks = "2 month", date_labels = "%b-%y")
```

```
lag_cases <- cases %>%
  left_join(county_policy_wide[,c("county","major_teaching")],
            by = c("COUNTY" = "county")) %>%
  na.omit()%>%
  select(COUNTY,DATE,CUMDEATHS,POPULATION,major_teaching)%>%
  group_by(COUNTY) %>%
  mutate(lag_total_deaths = lag(CUMDEATHS,21)) %>%
```

```
  ungroup()%>%
  group_by(DATE,major_teaching) %>%
  summarise(total_deaths = sum(CUMDEATHS),
            total_deaths_lag = sum(lag_total_deaths),
            total_pop = sum(POPULATION),
            death_prop = total_deaths/total_pop,
            lag_death_prop = total_deaths_lag/total_pop,
            death_prop_inc = (total_deaths-total_deaths_lag)/total_pop,
            .groups = "drop")

ggplot(lag_cases, aes(x = lag_death_prop, y = death_prop, color = major_teaching)) +
    geom_line(size = 1,alpha = .8, na.rm=T)+
  theme_bw() +
  labs(x = "Death Proportion of 3 weeks prior", y  = "Death Proportion",
       color = "Majority Teaching Method")
```




```
peak.date <- as.Date("2020-12-23")
ggplot(lag_cases,aes(x = DATE, y = death_prop_inc,
                     color = major_teaching,
                     fill = "red")) +
    geom_line(na.rm = T) +
    geom_rect(data = lag_cases[1,],
            aes(xmin=as.Date("2020/08/26"), xmax=as.Date("2020/12/12"),
                ymin=-Inf,ymax=Inf),
            color = NA,alpha=0.2, show.legend = F) +
    geom_vline(xintercept = peak.date, linetype = "dashed")+
    annotate("text",x = peak.date,y = .0005,
```

```
            label = peak.date,
            hjust = 1.2)  +
theme_bw() +
labs(x = "Date",
     y = "Death Proportion Increase",
     title = "Death Proportion Increase by Teaching Method",
     subtitle = "Increase compared to 3 Week Lag \nRed area represents Fall Semester",
     color = "Majority Teaching Method") +
scale_y_continuous(labels = comma)
```