Skeleton_of_Outputs

3/19/2020

This is a document for a code skeleton for the outputs section of the project.

Loading in relevant data sources for this:

```
#setwd('~/Dropbox/COVID-BaltimoreCity//')

require(socialmixr)
require(xlsx)
require(magrittr)
require(stringr)
require(reshape2)
require(dplyr)
require(ggplot2)
require(tidyr)
#load in data, subset incident cases/day (eventually should be symptomatic incident cases)
data_infected<- read.csv("SEIR_results_test.csv")

data_incident<- data_infected%>%
  select(incid1,incid2,incid3,incid4,incid5,incid6,incid7,incid8,incid9,incid10,incid11,
         incid12,incid13,incid14)

#consolidate age categories, 1-4 is now cat1, 5-6 is now cat2, 7-9 is now cat3, 10 is cat 4, 11 is cat5


data_incident<- data_incident%>%
  mutate(cat1=rowSums(.[1:4]))%>%
  mutate(cat2=rowSums(.[5:6]))%>%
  mutate(cat3=rowSums(.[7:9]))%>%
  mutate(cat4=rowSums(.[10]))%>%
  mutate(cat5=rowSums(.[11]))%>%
  mutate(cat6=rowSums(.[12]))%>%
  mutate(cat7=rowSums(.[13]))%>%
  mutate(cat8=rowSums(.[14]))

data_combined_cats<- data_incident%>%
  select(cat1,cat2,cat3,cat4,cat5,cat6,cat7,cat8)

#working probabilities for severe cases, hospitalization, ICU, vent and death.
#these are nested conditional probabilities, and they rely on the following  assumptions:
#the percentage of severe cases in each categories is roughly an average of the percentage of people
#with diabetes, the percentage of people with cancer and the percentage of people with hypertension in
#we assume now that 100% of severe cases get hospitalized (this could be adjusted)
#we assume that globally, 24% of hospitalized cases get moved to ICU so this global percentage gets pro
#we assume that 25% of ICU cases are put on a vent again this gets projected down to age groups
# we assume that 50% of ventilated cases die, again this is projected down to age groups.
#HCW and homeless people have the same probabilities as people in category 4

prob_severe<- c(0.0318, 0.043, 0.197, 0.336,0.378,0.409,0.336,0.336)
```

```
prob_hospitalized<- rep(1,ncol(data_combined_cats))

prob_ICU<- c(0.15,0.15,0.2,0.223,0.27,0.3,0.223,0.223)

prob_vent<-c(0.1,0.1,0.125,0.2,0.318,0.337,0.2,0.2)

prob_death<-c(0.2,0.2,0.2,0.46,0.49,0.59,0.46,0.46)

prob_matrix<- cbind(prob_severe,prob_hospitalized,prob_ICU,
                    prob_vent,prob_death)
```

Now calculate the number in of severe, hospitalized, ICU and vented infections at each timepoint (day) in each group (age groups 1-12 and healthcare workers and homeless).

```
#calculate number of incident cases that get classified as severe,  of those,
#the number get hospitalized (currently 100%), of that then those that get classified as ICU and then o
#this is for each age group (+ HCW and homeless)
#note this categorization is NOT a disease progression, a person gets assigned based on the worst state


counts_array<- array(NA, dim=c(nrow(data_combined_cats),ncol(data_combined_cats),ncol(prob_matrix)))


for(j in 1:ncol(data_combined_cats)){
  for(i in 1:nrow(data_combined_cats)){
      counts_array[i,j,1]<- data_combined_cats[i,j]*prob_matrix[j,1]
      counts_array[i,j,2]<- counts_array[i,j,1]*prob_matrix[j,2]
      counts_array[i,j,3]<- counts_array[i,j,2]*prob_matrix[j,3]
      counts_array[i,j,4]<- counts_array[i,j,3]*prob_matrix[j,4]
  }
  for(k in 1:nrow(data_combined_cats)){
    if(k<=11){
      counts_array[k,j,5]<- 0
    }
    else{
      counts_array[k,j,5]<- counts_array[k-11,j,4]*prob_matrix[j,5]
    }
  }
}
```

Now load in a dataset that gives estimates of how long people stay hospitalized, in ICU or on vents Compute a running total of each category.

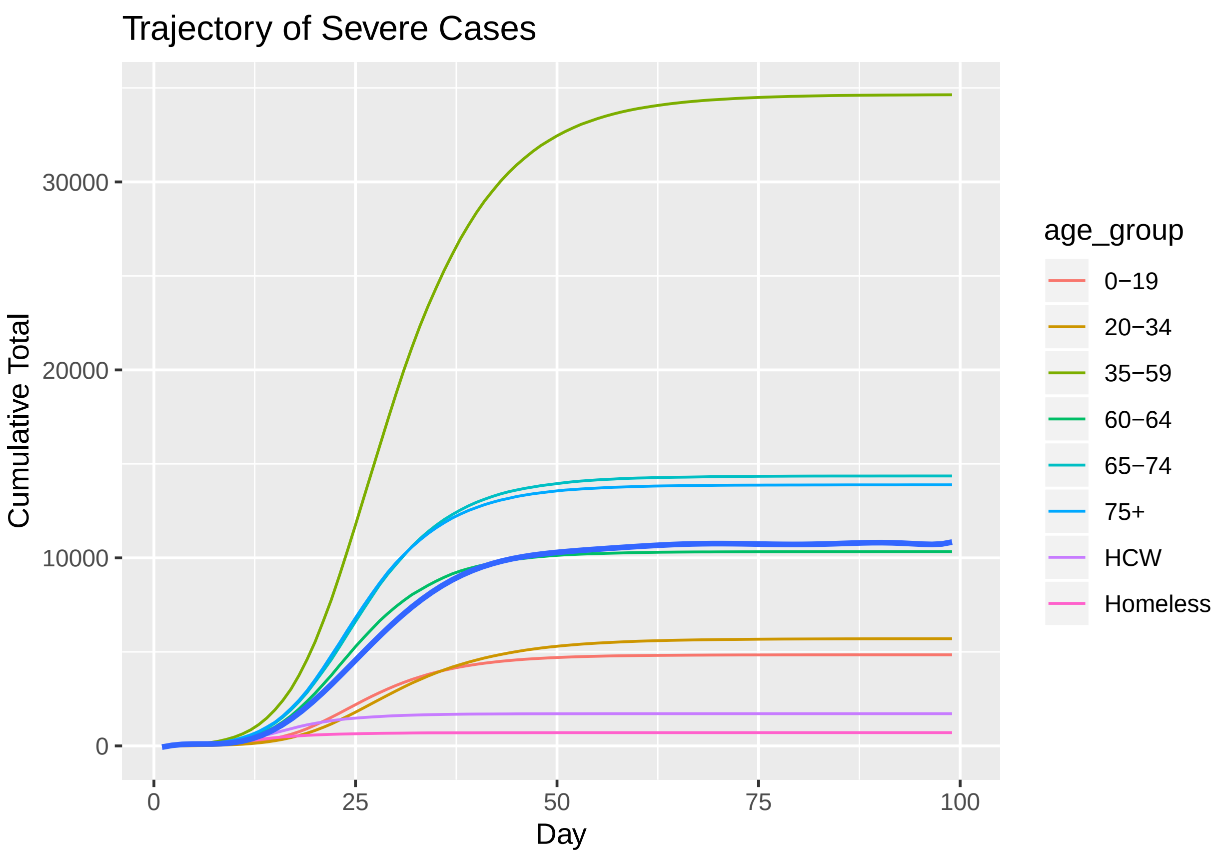
```
#for now we are supposing we get a point estimate to determine how long people stay in each of the stat

length_of_stay<- c(NA,11,11,11,NA)

cumulative_array<- array(NA, dim=c(nrow(data_combined_cats),ncol(data_combined_cats),ncol(prob_matrix))


  for(j in 1:ncol(data_combined_cats)){
    cumulative_array[1,j,]<- counts_array[1,j,]

    for(i in 2:nrow(data_combined_cats)){
      cumulative_array[i,j,1]<-cumulative_array[i-1,j,1]+counts_array[i,j,1]
```

```r
      cumulative_array[i,j,5]<-cumulative_array[i-1,j,5]+counts_array[i,j,5]
    }

    for(k in 2:nrow(data_combined_cats)){
      if(k<=length_of_stay[2]){
      cumulative_array[k,j,2]<- cumulative_array[k-1,j,2]+counts_array[k,j,2]
      }
      else {
        cumulative_array[k,j,2]<-cumulative_array[k-1,j,2]+
          counts_array[k,j,2]-counts_array[k-length_of_stay[2],j,2]
      }
    }
  for(l in 2:nrow(data_combined_cats)){
      if(l <= length_of_stay[3]){
        cumulative_array[l,j,3]<- cumulative_array[l-1,j,3]+counts_array[l,j,3]
      }
    else{
      cumulative_array[l,j,3]<-cumulative_array[l-1,j,3]+
          counts_array[l,j,3]-counts_array[l-length_of_stay[3],j,3]
    }
  }
    for(m in 2:nrow(data_combined_cats)){
    if(m<=length_of_stay[4]){
      cumulative_array[m,j,4]<- cumulative_array[m-1,j,4]+counts_array[m,j,4]
    }
    else{
      cumulative_array[m,j,4]<-cumulative_array[m-1,j,4]+
          counts_array[m,j,4]-counts_array[m-length_of_stay[4],j,4]
      }
    }

}

cumulative_array[,2,3]
```

```
##   [1]   0.18705   0.41280   0.76755   1.27065   1.85115   2.67675   3.76680
##   [8]   5.26965   7.30785  10.07490  13.46115  18.04710  23.76825  30.86325
##  [15]  39.99645  50.77440  63.90660  79.05765  96.18885 116.06775 138.06225
##  [22] 161.54025 187.06290 212.42430 237.40515 261.64425 284.54820 304.65930
##  [29] 321.96465 336.46425 346.95195 353.26650 354.36945 352.00230 346.43595
##  [36] 337.70265 326.33130 311.78655 296.60970 280.26540 263.51475 245.61600
##  [43] 227.79465 212.19855 195.39630 180.07110 165.71340 151.53630 139.57155
##  [50] 127.23915 116.25480 105.15435  95.90505  87.09435  78.59970  70.96935
##  [57]  63.79695  57.28890  51.83220  46.54965  41.87985  37.53255  33.88185
##  [64]  30.23115  27.17385  24.40035  21.97515  19.76925  17.76330  15.67995
##  [71]  13.80945  12.51300  11.26815  10.19745   9.17835   8.20440   7.57230
##  [78]   6.88215   6.09525   5.56635   5.02455   4.60530   4.27635   3.77325
##  [85]   3.40560   2.95410   2.64450   2.36070   2.06400   1.87050   1.67055
##  [92]   1.49640   1.37385   1.17390   1.12230   0.96750   0.89010   0.82560
##  [99]   0.69015        NA
```

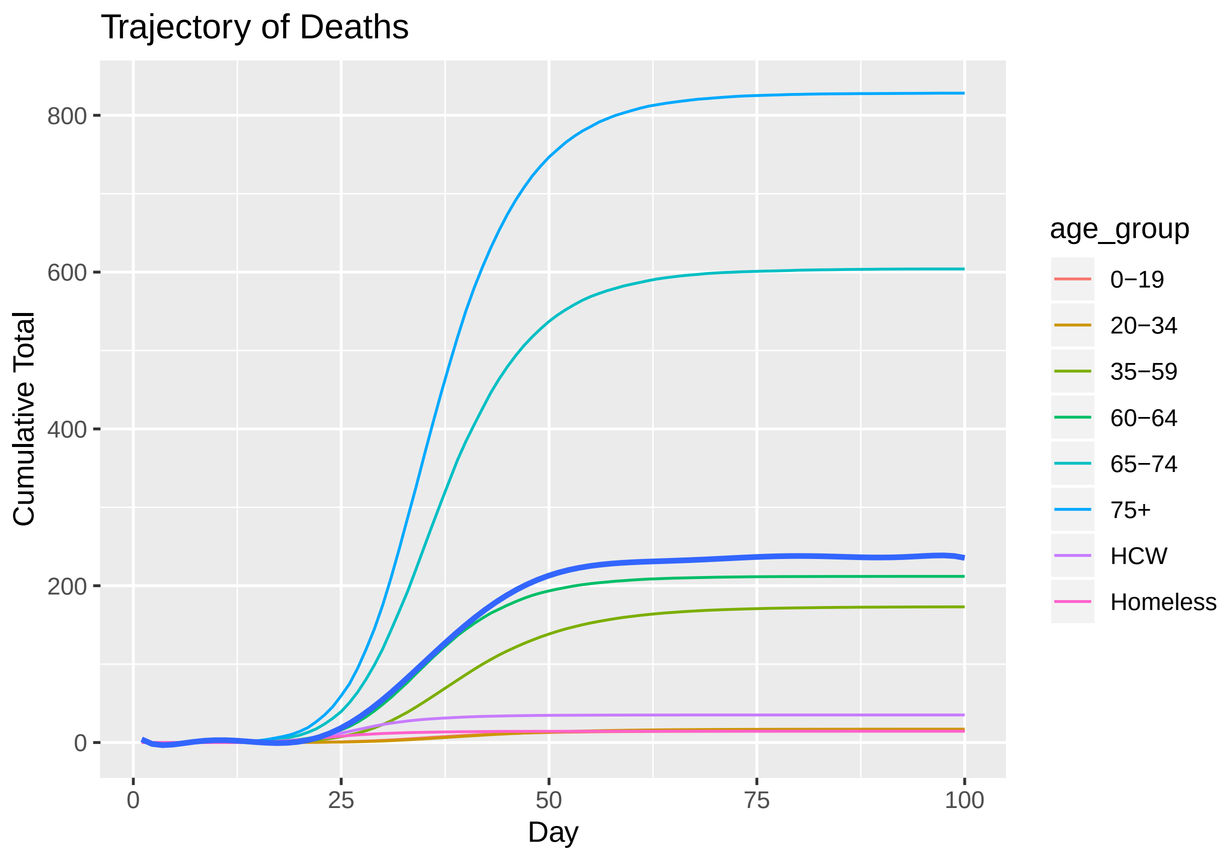
Plot cumulative values over 99 days.

```
#return to long format for each state
#severe

severe_wide<-as.data.frame(cbind(rep(1:100),cumulative_array[,,1]))
colnames(severe_wide)<- c("Day","0-19","20-34","35-59","60-64","65-74","75+","HCW","Homeless")

severe_long<- gather(severe_wide,age_group, cumulative_total, '0-19':Homeless,factor_key=TRUE)



plt_severe<- ggplot(severe_long)+
  geom_line(aes(y= cumulative_total, x=Day,color=age_group))+
  stat_smooth(aes(y= cumulative_total,x=Day), method=lm, formula=y~poly(x,10), se=FALSE)+
  labs(title="Trajectory of Severe Cases",y="Cumulative Total")

plt_severe
```

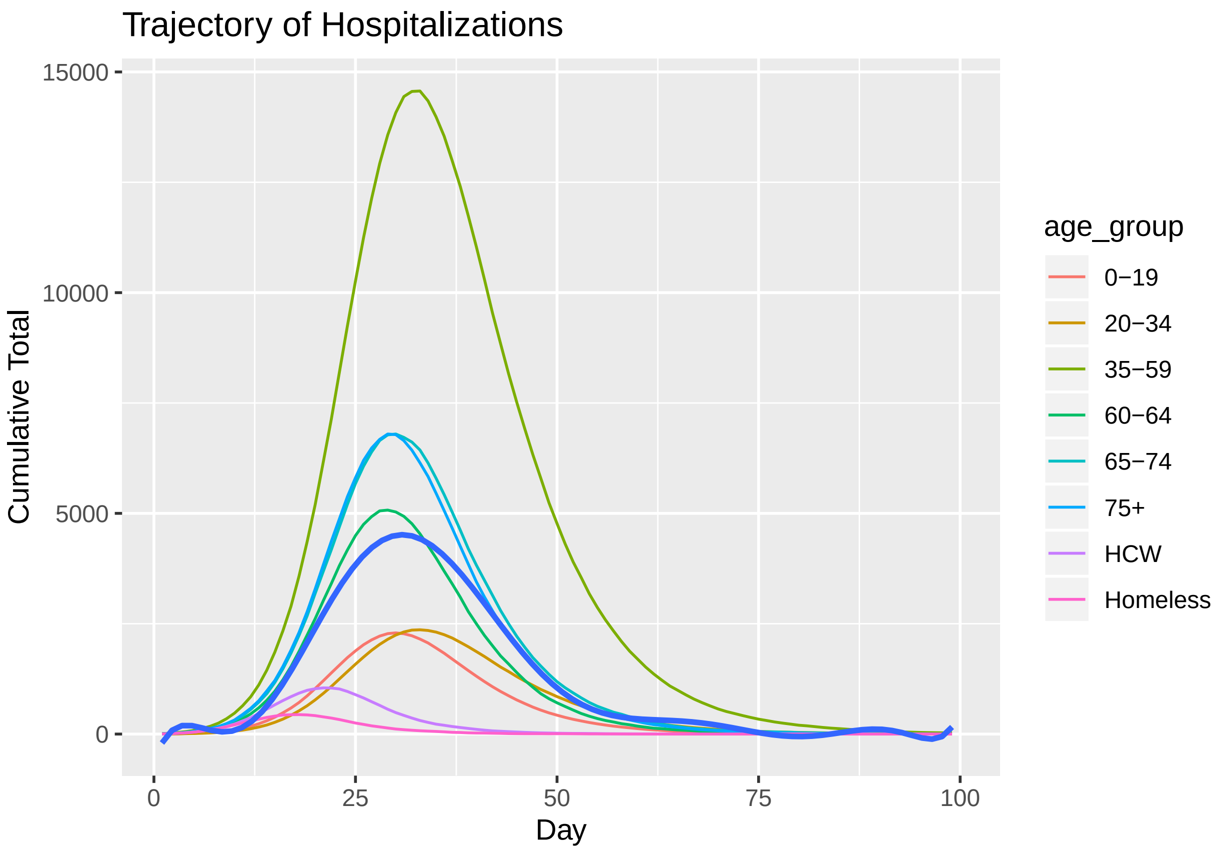



```
################################
deaths_wide<-as.data.frame(cbind(rep(1:100),cumulative_array[,,5]))
colnames(deaths_wide)<- c("Day","0-19","20-34","35-59","60-64","65-74","75+","HCW","Homeless")

deaths_long<- gather(deaths_wide,age_group, cumulative_total, '0-19':Homeless,factor_key=TRUE)



plt_deaths<- ggplot(deaths_long)+
  geom_line(aes(y= cumulative_total, x=Day,color=age_group))+
```

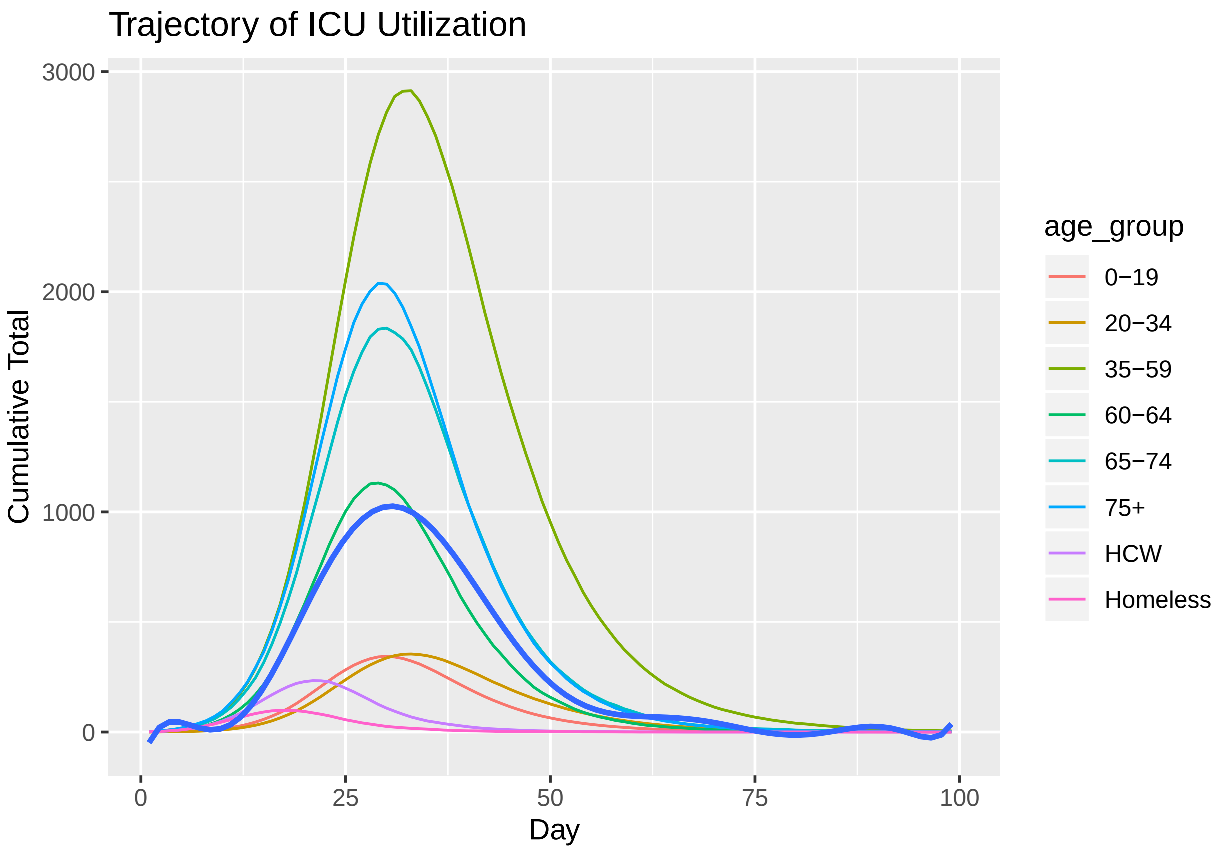
```
  stat_smooth(aes(y= cumulative_total,x=Day), method=lm, formula=y~poly(x,10), se=FALSE)+
  labs(title="Trajectory of Deaths",y="Cumulative Total")

plt_deaths
```



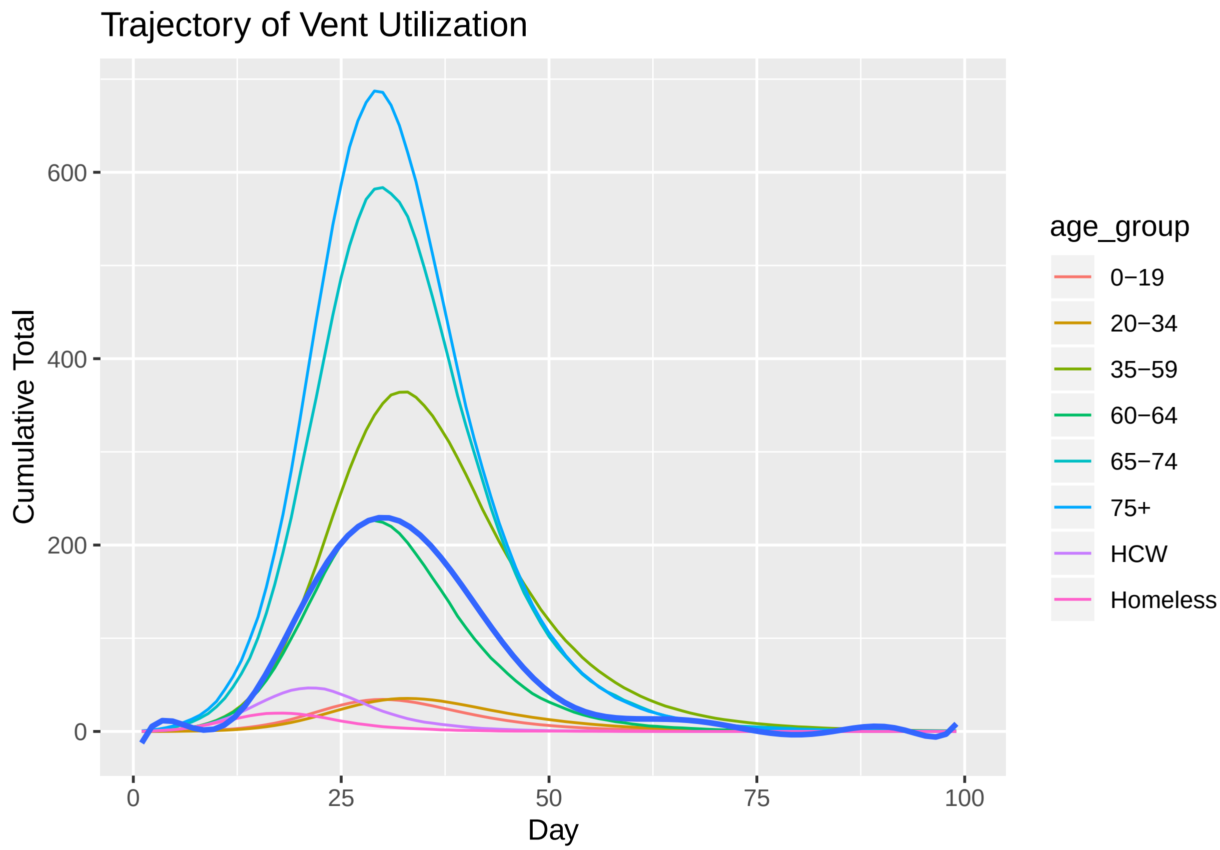

```
##################################


hosp_wide<-as.data.frame(cbind(rep(1:100),cumulative_array[,,2]))
colnames(hosp_wide)<- c("Day","0-19","20-34","35-59","60-64","65-74","75+","HCW","Homeless")

hosp_long<- gather(hosp_wide,age_group, cumulative_total, '0-19':Homeless,factor_key=TRUE)



plt_hosp<- ggplot(hosp_long)+
  geom_line(aes(y= cumulative_total, x=Day,color=age_group))+
  stat_smooth(aes(y= cumulative_total,x=Day), method=lm, formula=y~poly(x,10), se=FALSE)+
  labs(title="Trajectory of Hospitalizations",y="Cumulative Total")

plt_hosp
```



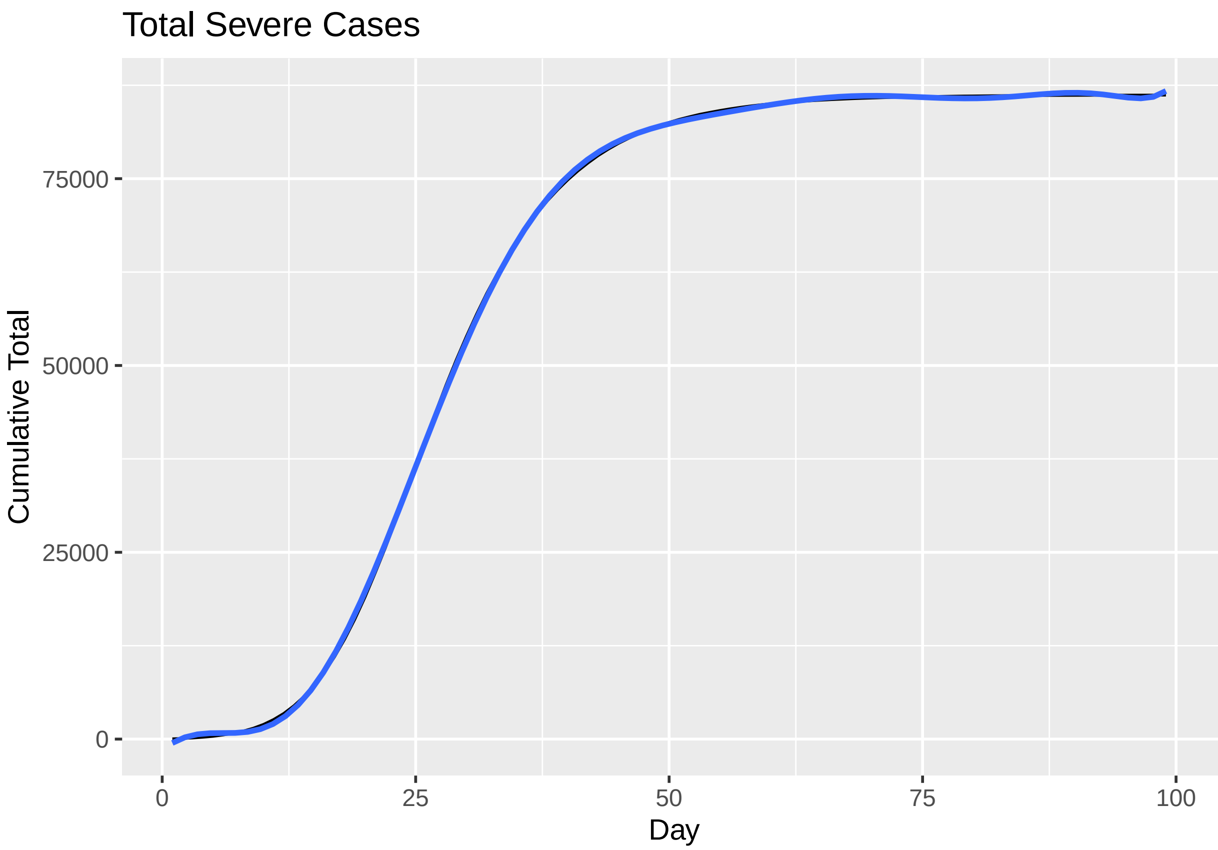

```
####################################



ICU_wide<-as.data.frame(cbind(rep(1:100),cumulative_array[,,3]))
colnames(ICU_wide)<- c("Day","0-19","20-34","35-59","60-64","65-74","75+","HCW","Homeless")

ICU_long<- gather(ICU_wide,age_group, cumulative_total, '0-19':Homeless,factor_key=TRUE)




plt_ICU<- ggplot(ICU_long)+
  geom_line(aes(y= cumulative_total, x=Day,color=age_group))+
  stat_smooth(aes(y= cumulative_total,x=Day), method=lm, formula=y~poly(x,10), se=FALSE)+
  labs(title="Trajectory of ICU Utilization",y="Cumulative Total")

plt_ICU
```



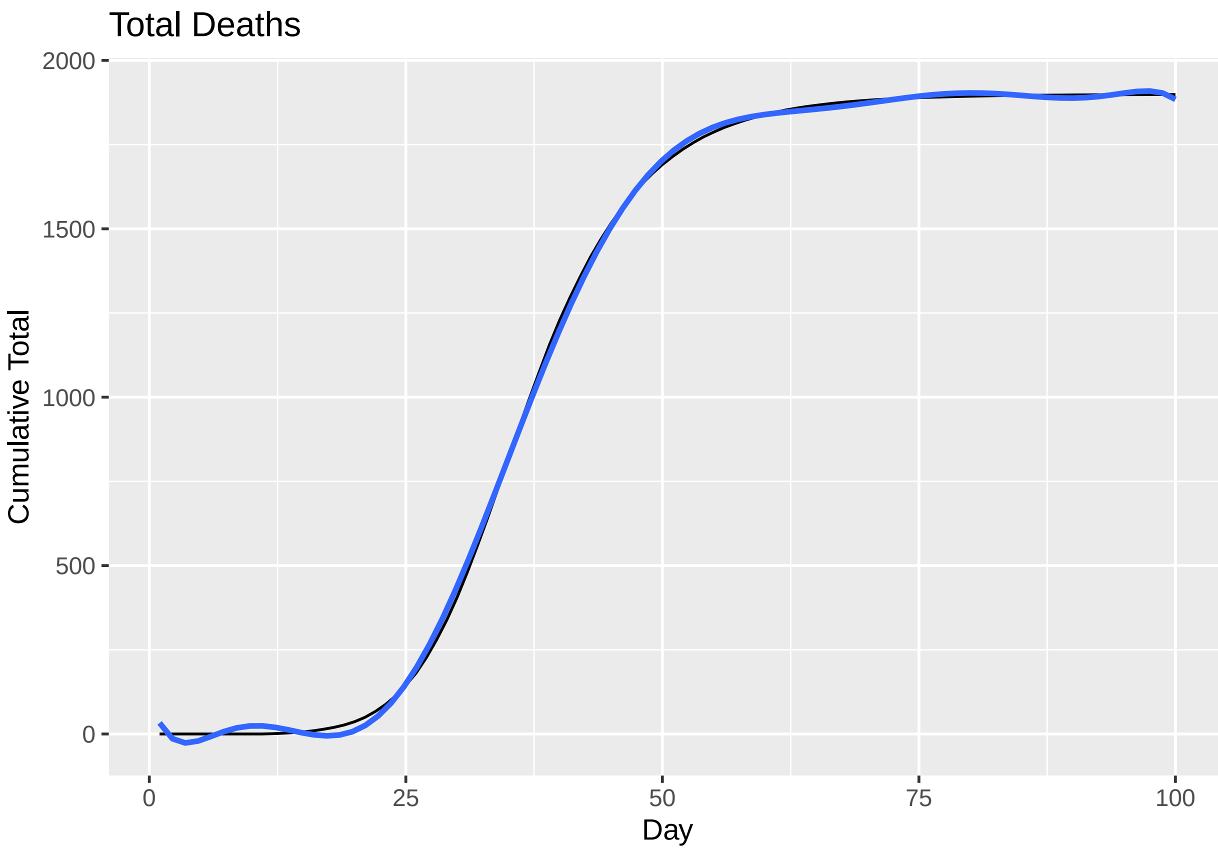

```
################################################


vent_wide<-as.data.frame(cbind(rep(1:100),cumulative_array[,,4]))
colnames(vent_wide)<- c("Day","0-19","20-34","35-59","60-64","65-74","75+","HCW","Homeless")

vent_long<- gather(vent_wide,age_group, cumulative_total, '0-19':Homeless,factor_key=TRUE)


plt_vent<- ggplot(vent_long)+
  geom_line(aes(y= cumulative_total, x=Day,color=age_group))+
  stat_smooth(aes(y= cumulative_total,x=Day), method=lm, formula=y~poly(x,10), se=FALSE)+
  labs(title="Trajectory of Vent Utilization",y="Cumulative Total")

plt_vent
```



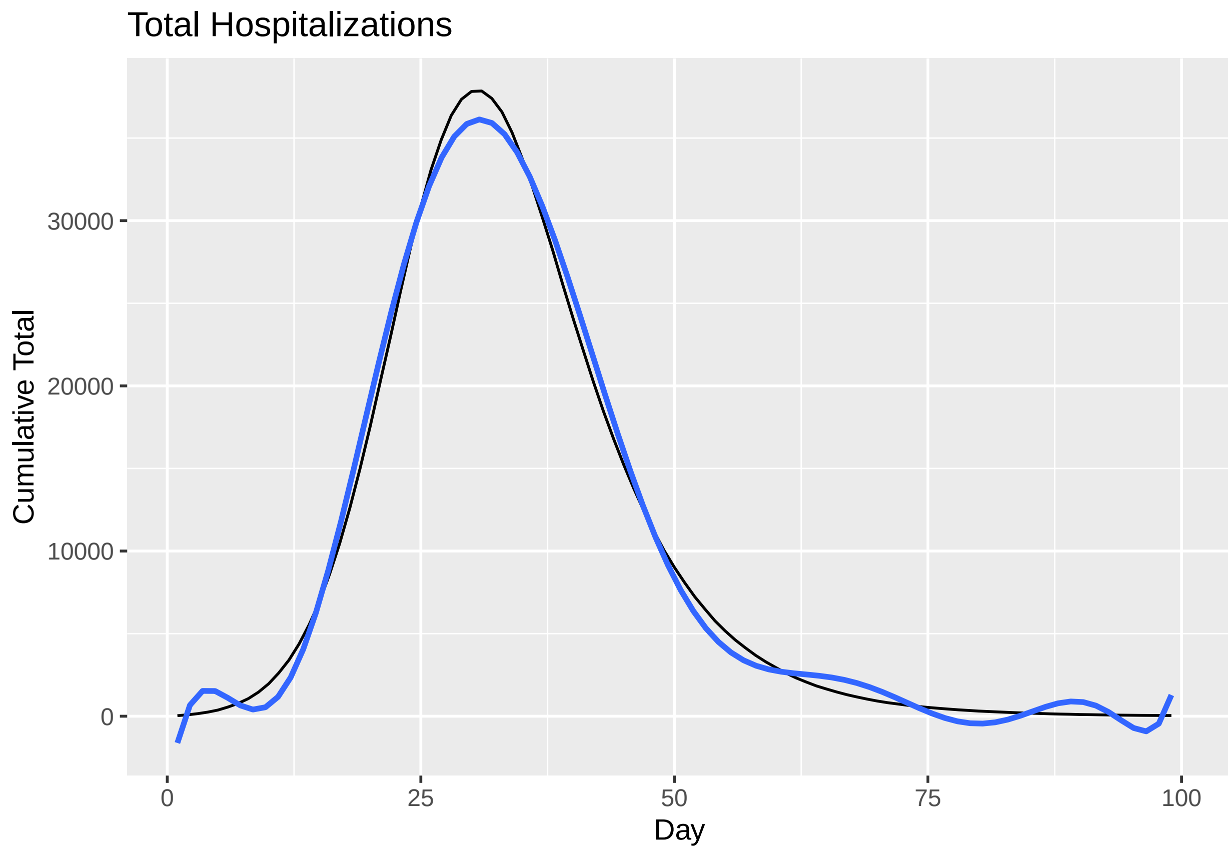

```
cumulative_totals<- matrix(NA, nrow=nrow(data_combined_cats),ncol=ncol(prob_matrix))

for(j in 1:ncol(prob_matrix)){
  for(i in 1:nrow(data_combined_cats)){
  cumulative_totals[i,j]<- sum(cumulative_array[i,,j])
  }
}

cumulative_totals<- cbind(rep(1:100), cumulative_totals)


plt_totals_severe<- ggplot() +
  geom_line(aes(y=cumulative_totals[,2],x=cumulative_totals[,1]))+
  stat_smooth(aes(y=cumulative_totals[,2],x=cumulative_totals[,1]),
              method=lm, formula=y~poly(x,10),se=FALSE)+
  labs(title="Total Severe Cases", x="Day", y="Cumulative Total")
plt_totals_severe
```



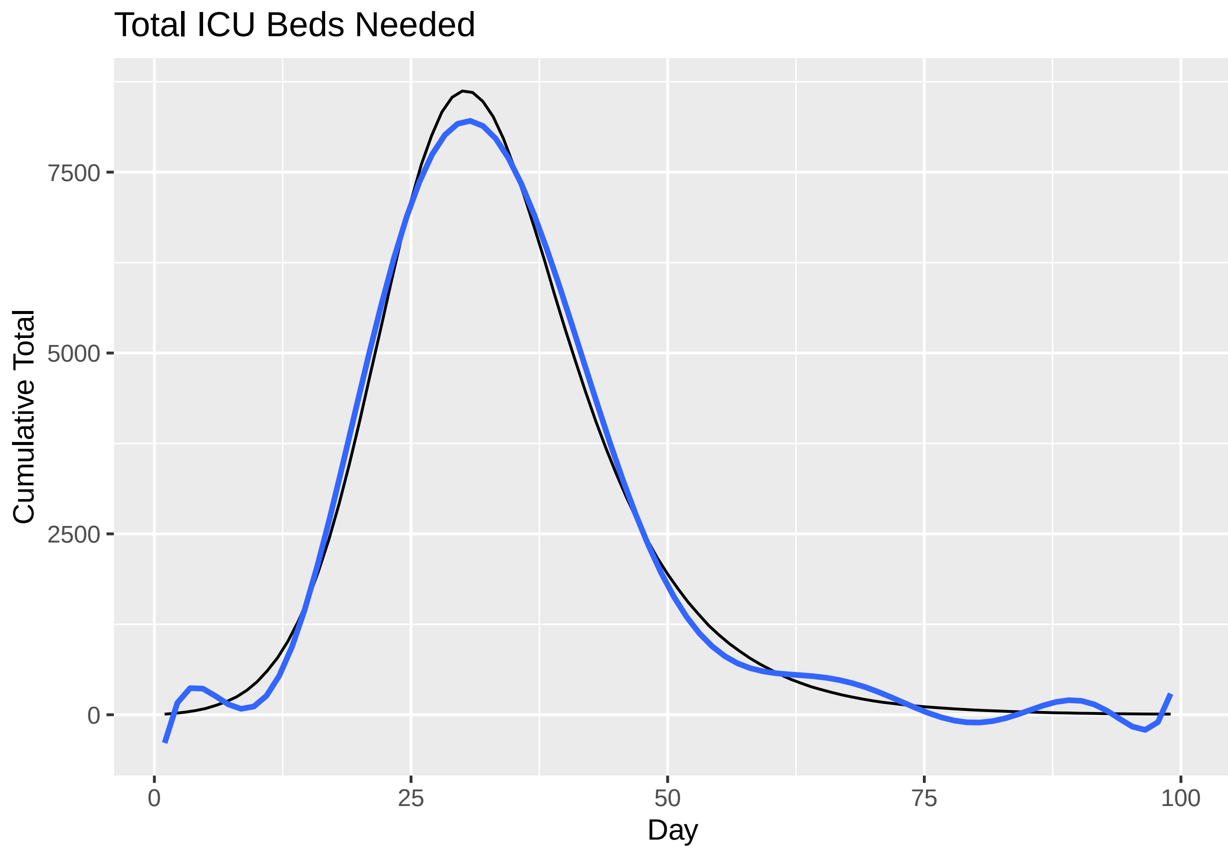

```
plt_totals_deaths<- ggplot() +
  geom_line(aes(y=cumulative_totals[,6],x=cumulative_totals[,1]))+
  stat_smooth(aes(y=cumulative_totals[,6],x=cumulative_totals[,1]),
              method=lm, formula=y~poly(x,10),se=FALSE)+
  labs(title="Total Deaths", x="Day", y="Cumulative Total")
plt_totals_deaths
```



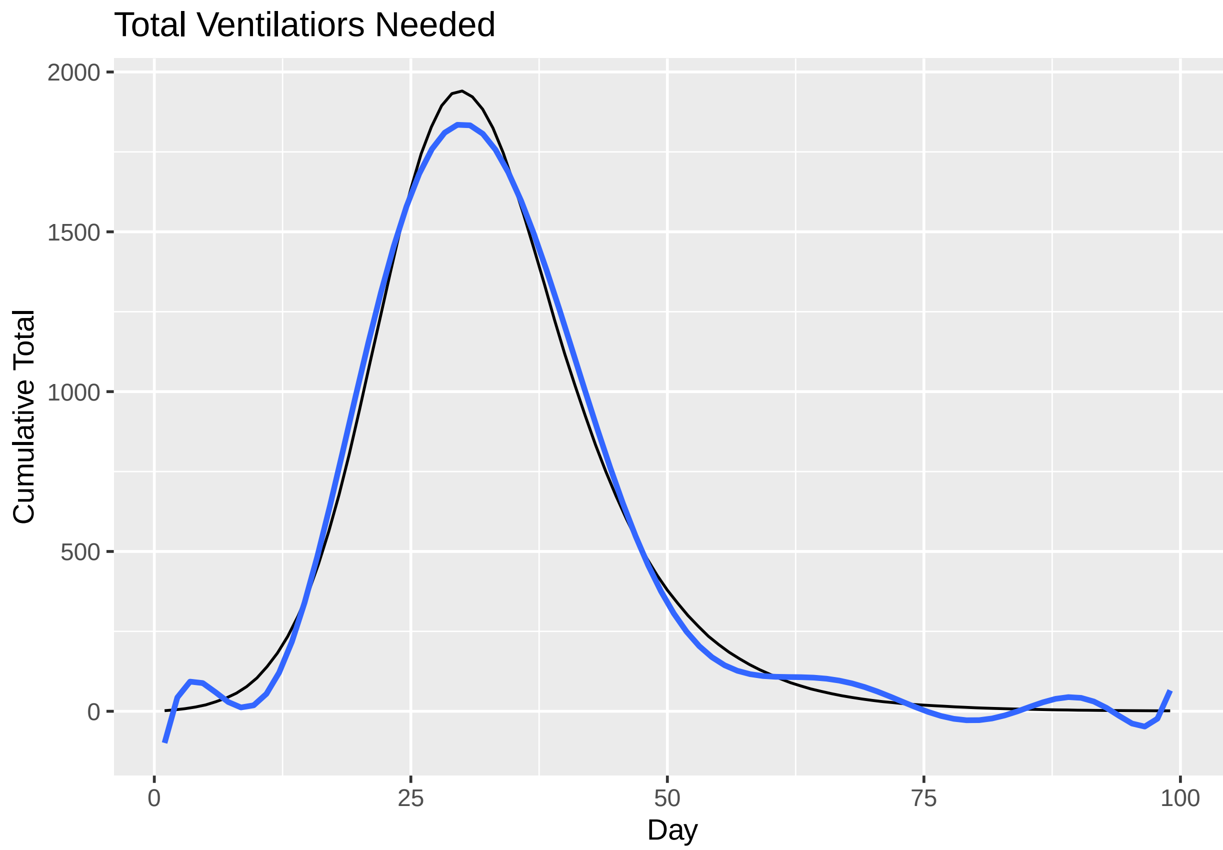

```
plt_totals_Hosp<- ggplot() +
  geom_line(aes(y=cumulative_totals[,3],x=cumulative_totals[,1]))+
  stat_smooth(aes(y=cumulative_totals[,3],x=cumulative_totals[,1]),
              method=lm, formula=y~poly(x,10),se=FALSE)+
  labs(title="Total Hospitalizations", x="Day", y="Cumulative Total")
plt_totals_Hosp
```




```
plt_totals_ICU<- ggplot() +
  geom_line(aes(y=cumulative_totals[,4],x=cumulative_totals[,1]))+
  stat_smooth(aes(y=cumulative_totals[,4],x=cumulative_totals[,1]),
              method=lm, formula=y~poly(x,10),se=FALSE)+
  labs(title="Total ICU Beds Needed", x="Day", y="Cumulative Total")
plt_totals_ICU
```




```r
plt_totals_vent<- ggplot() +
  geom_line(aes(y=cumulative_totals[,5],x=cumulative_totals[,1]))+
  stat_smooth(aes(y=cumulative_totals[,5],x=cumulative_totals[,1]),
              method=lm, formula=y~poly(x,10),se=FALSE)+
  labs(title="Total Ventilatiors Needed", x="Day", y="Cumulative Total")
plt_totals_vent
```




```
################################################

#number of severe cases total
cumulative_totals[99,2]
```

```
## [1] 86185.61
```

```
#number of deaths total
cumulative_totals[100,6]
```

```
## [1] 1898.571
```

```
#max number of hospital beds needed in one day (including ICU beds in this total)

max(cumulative_totals[,3],na.rm=TRUE)
```

```
## [1] 37850.58
```

```
#max number of ICU beds needed in one day (including people who might also need a vent)

max(cumulative_totals[,4],na.rm=TRUE)
```

```
## [1] 8621.777
```

```
#max number of vents needed in one day
max(cumulative_totals[,5],na.rm=TRUE)
```

```
## [1] 1940.527
```