52414 - lab 1

52414

27/4/2021

Lab 1: Basic Data Wrangling and Plotting

Submission Deadline: 19/5/2021 at 23:59

The only allowed libraries are the following (**please do not add your own**):

Solution:

Write your solutions here seperately for each question in the followng format:

1. [MY SOLUTION TEXT - EXPLANATIONS]

```
# R code for my solution
data<-read.csv("/Users/elkysandor/Downloads/owid-covid-data.csv")
data$date<-as.Date(data$date)
class(data$date)#check
```

```
## [1] "Date"
```

[MY SOLUTION TEXT - DESCRIPTION OF RESULTS]

2. [MY SOLUTION TEXT - EXPLANATIONS]

```
# R code for my solution
max_date_val_3<-data$date[max(which(!is.na(data$total_cases_per_million)))]
table_cases_p_m<-data %>% filter(date==max_date_val_3)%>%
arrange(desc(total_cases_per_million))%>%
select(location,date,total_cases_per_million)%>%
head(5)
knitr::kable(table_cases_p_m, caption = "Top 5 countries of current total_cases_per_million."
)
```

Top 5 countries of current total_cases_per_million.

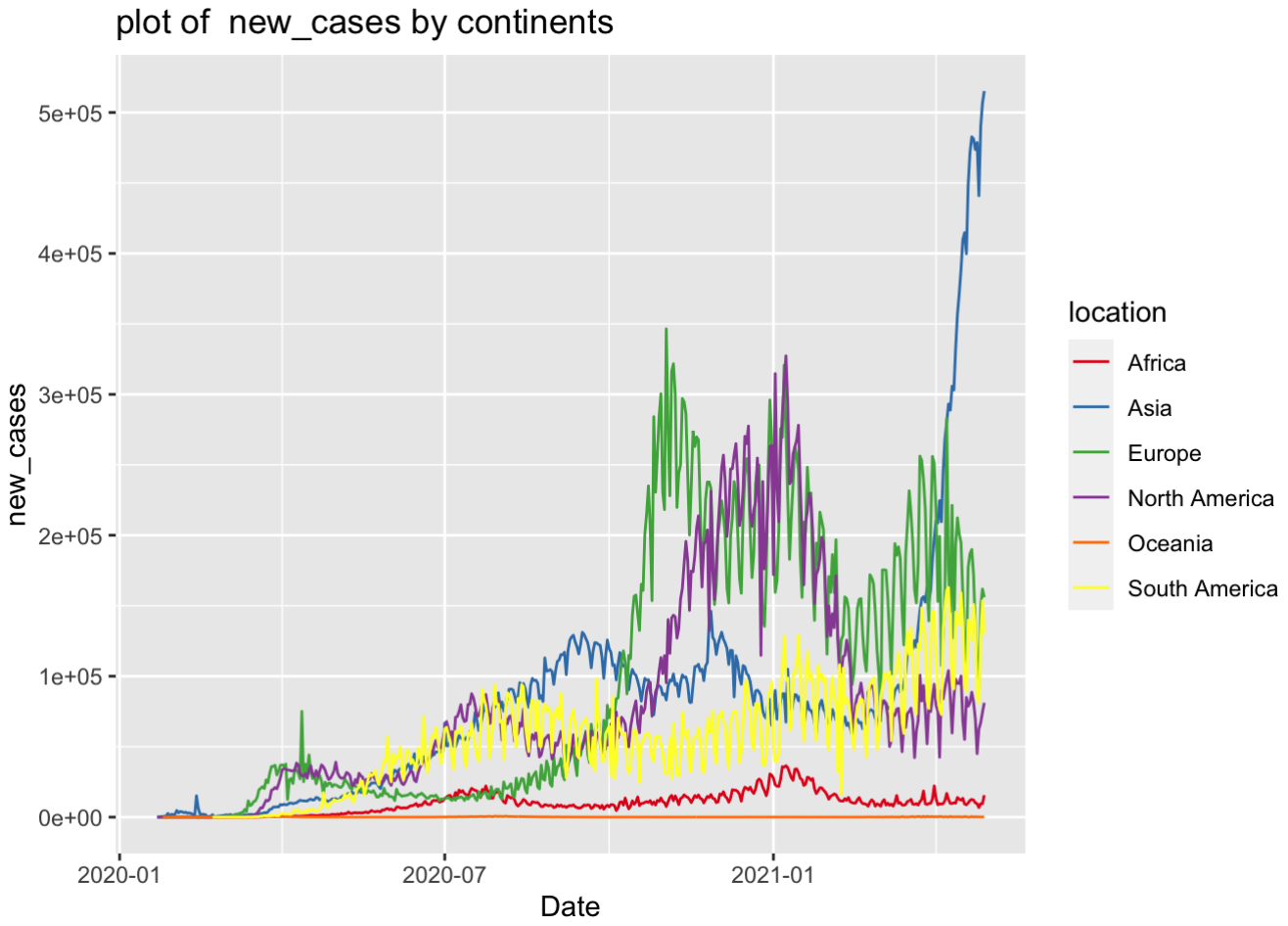
location	date	total_cases_per_million
Andorra	2021-04-29	170814.7
Montenegro	2021-04-29	154852.5
Czechia	2021-04-29	152072.0
San Marino	2021-04-29	149095.4
Slovenia	2021-04-29	115125.9

```
max_date_val_2<-data$date[max(which(!is.na(data$total_deaths_per_million)))]

table_total_death_m<-data %>% filter(date==max_date_val_2)%>%
arrange(desc(total_deaths_per_million))%>%
select(location,date,total_deaths_per_million)

table_total_death_m<-data %>% filter(date==max(data$date))%>%
arrange(desc(total_deaths_per_million))%>%
select(location,date,total_deaths_per_million)

max_date_val<-data$date[max(which(!is.na(data$total_vaccinations_per_hundred)))]

table_vacc_p_h<-data %>% filter(date==max_date_val)%>%
arrange(desc(total_vaccinations_per_hundred),na.rm=FALSE)%>%
select(location,date,total_vaccinations_per_hundred)

vec_of_conti<-c("South America","North America",
                "Asia","Oceania","Europe","Africa")
clean_conti<-filter(data,continent%in% vec_of_conti)
```

[MY SOLUTION TEXT - DESCRIPTION OF RESULTS]

3. [MY SOLUTION TEXT - EXPLANATIONS]

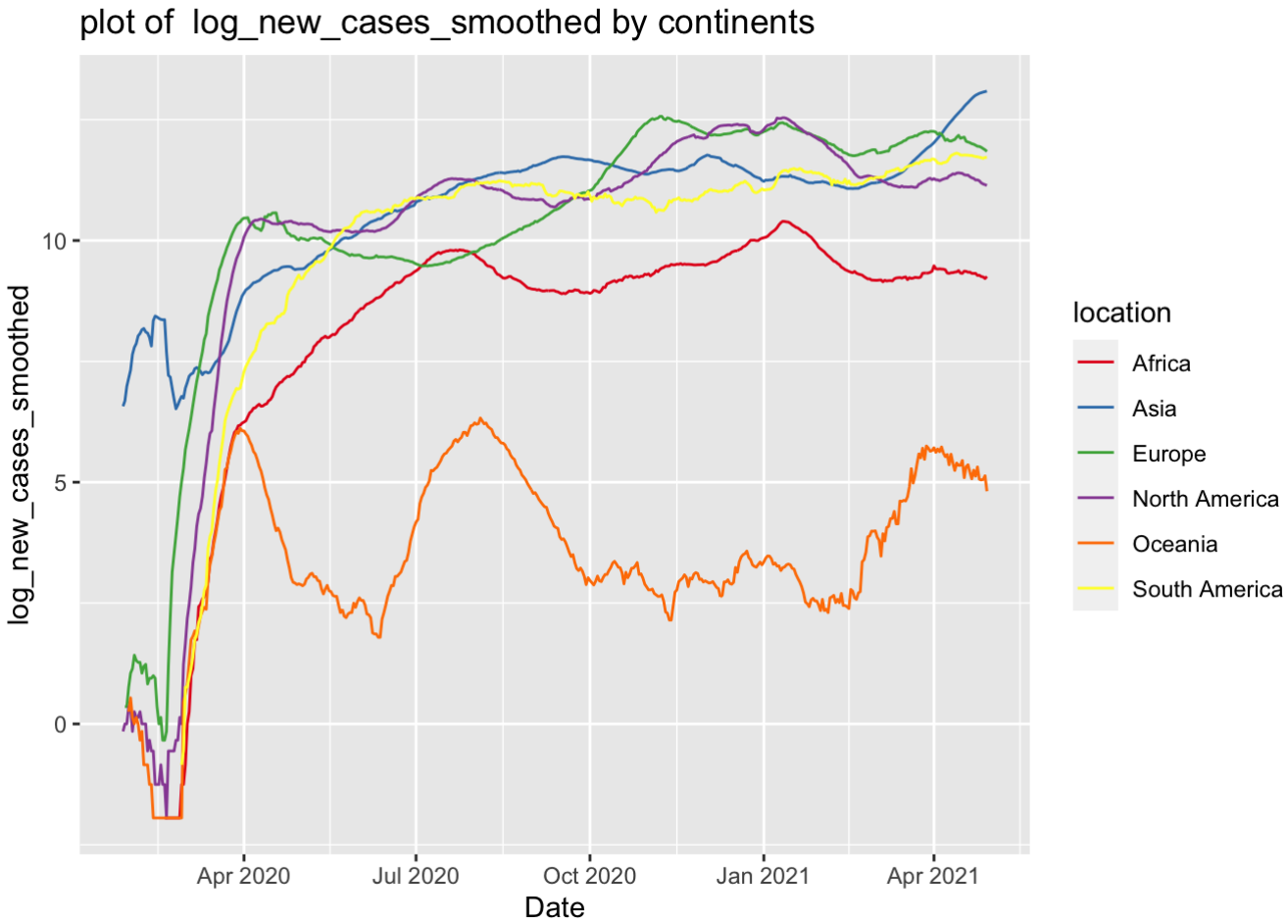
```
ploting <- function(data_f,colom_name){
  only_conti<-filter(data_f,location%in% vec_of_conti)
  col_en<-only_conti[[colom_name]]
    ggplot(only_conti,aes(x=date,y=col_en))+
    geom_line(aes(color=location),size=0.5)+
    scale_color_brewer(palette = "Set1")+
    labs(title =paste("plot of ",colom_name,"by continents"),x = "Date",
         y=colom_name)
}
#B
ploting(data,"new_cases")
```




```
fun_log <- function(vector_with_nas) {
  ind_bad<-which(is.na(vector_with_nas)|vector_with_nas==0)
  return(ind_bad)
}
prep_to_log<-fun_log(data$new_cases_smoothed)
data_to_log<-data[-prep_to_log,]
data_to_log$log_new_cases_smoothed<-log(data_to_log$new_cases_smoothed)
```

```
## Warning in log(data_to_log$new_cases_smoothed): NaNs produced
```

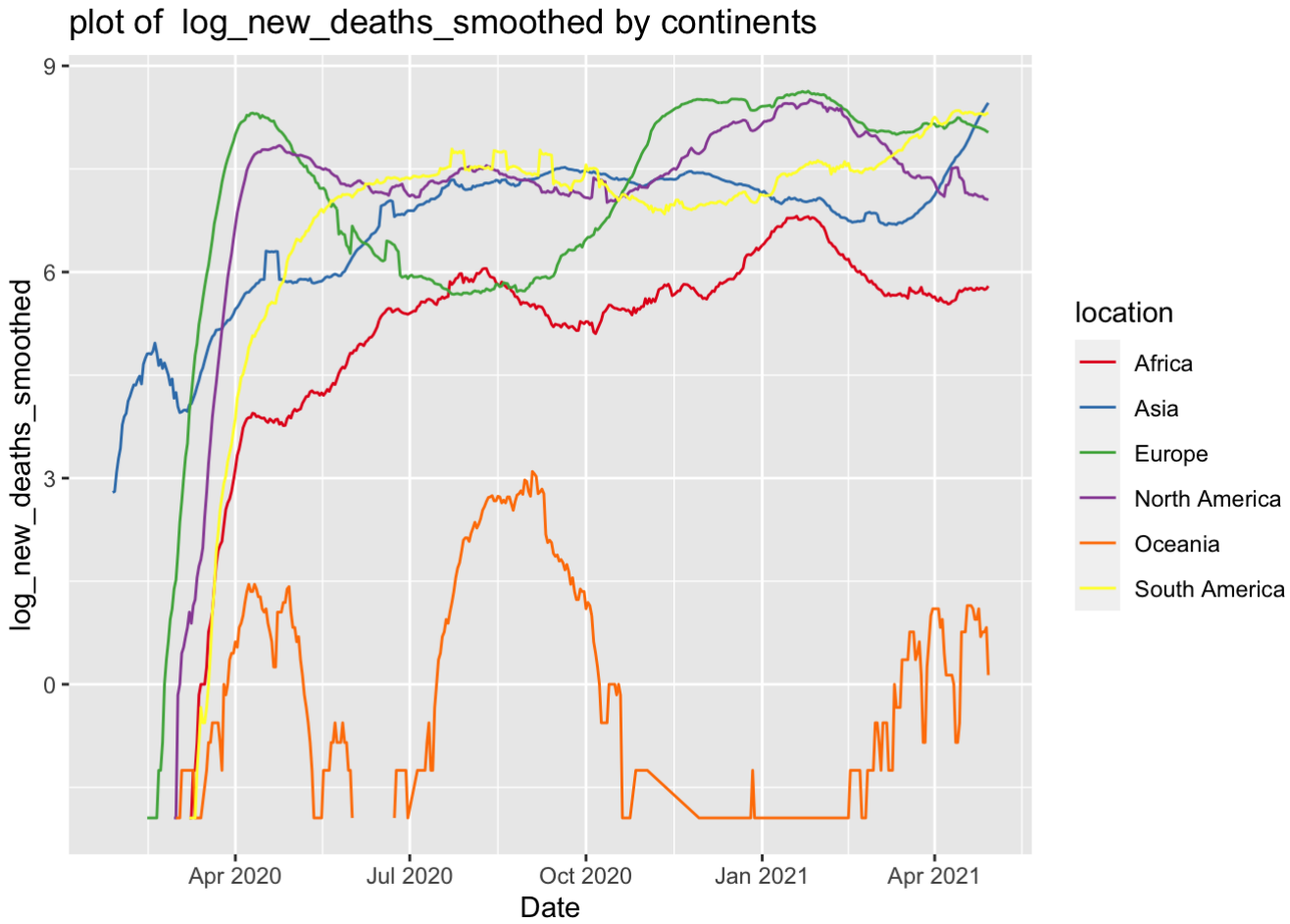
```
ploting(data_to_log,"log_new_cases_smoothed")
```



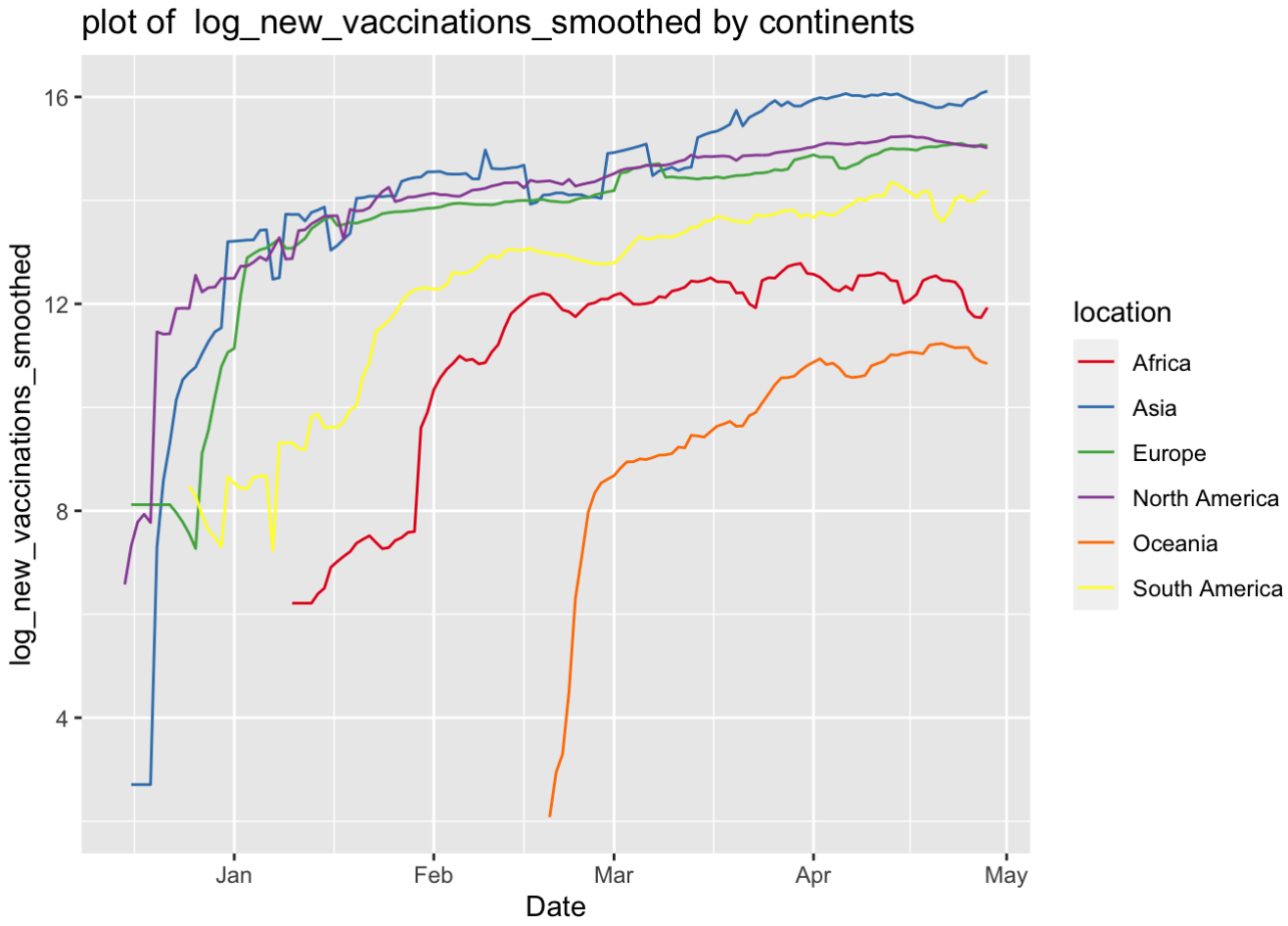

```
prep_to_log_2<-fun_log(data$new_deaths_smoothed)
data_to_log_2<-data[-prep_to_log_2,]
data_to_log_2$log_new_deaths_smoothed<-log(data_to_log_2$new_deaths_smoothed)
```

```
## Warning in log(data_to_log_2$new_deaths_smoothed): NaNs produced
```

```
ploting(data_to_log_2,"log_new_deaths_smoothed")
```




```
prep_to_log_3<-fun_log(data$new_vaccinations_smoothed)
data_to_log_3<-data[-prep_to_log_3,]
data_to_log_3$log_new_vaccinations_smoothed<-log(data_to_log_3$new_vaccinations_smoothed)
ploting(data_to_log_3,"log_new_vaccinations_smoothed")
```



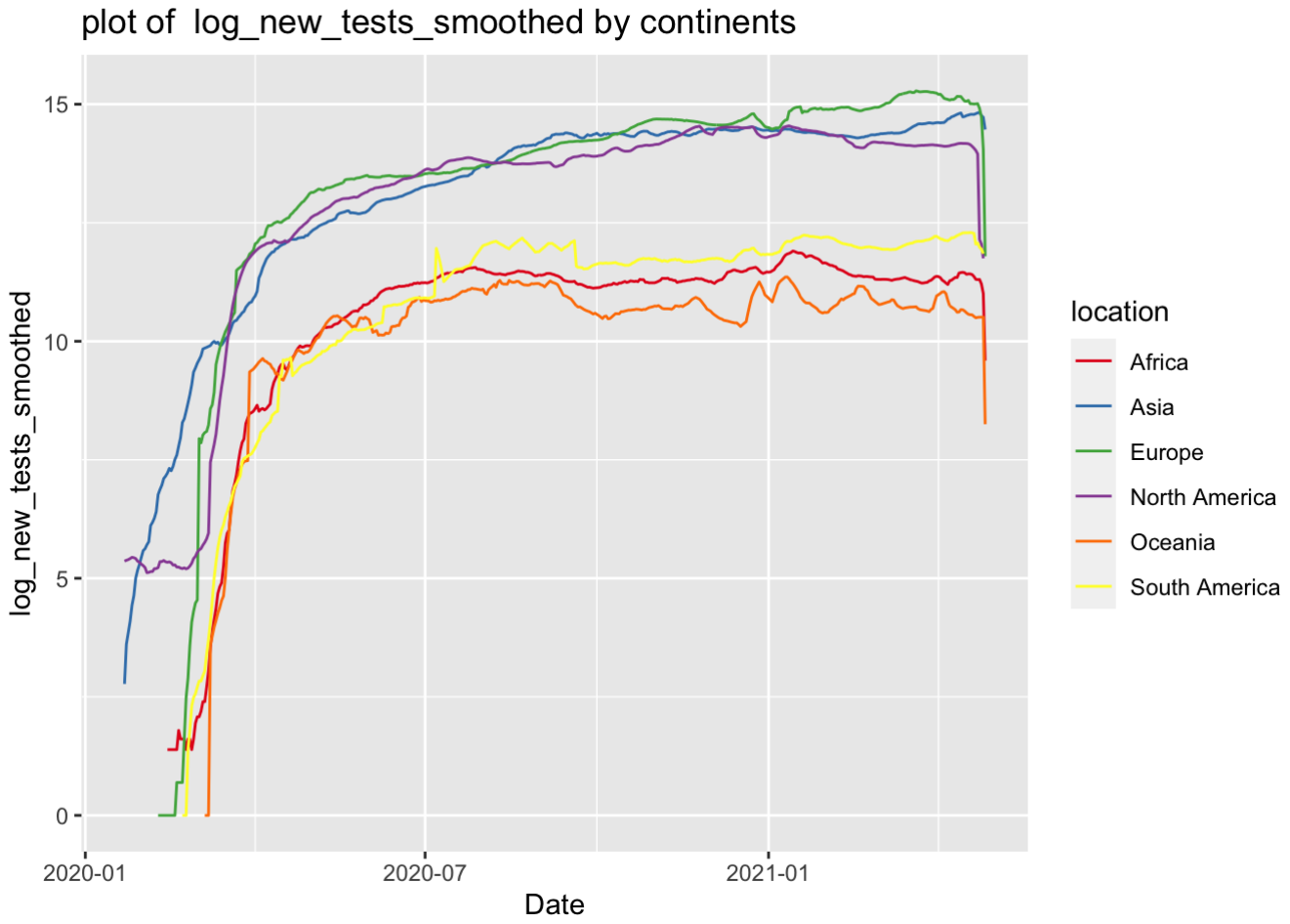
[MY SOLUTION TEXT - DESCRIPTION OF RESULTS]

4. [MY SOLUTION TEXT - EXPLANATIONS]

```
# R code for my solution

fill_column<-function(data_fr,col_ne){
  clean_conti<-filter(data_fr,continent%in% vec_of_conti)
  col_vec<-data_fr[[col_ne]]
  df_to_fill<-filter(data_fr,location%in%vec_of_conti)
  stat_for_fill<-aggregate(col_vec~continent+date,data=data_fr,function(x)
  sum(x,na.rm = TRUE))
  stat_for_fill<-filter(stat_for_fill,continent%in% vec_of_conti)
  colnames(stat_for_fill)[3]<-col_ne
  df_to_fill<-select(df_to_fill,location,date,col_ne)
  colnames(df_to_fill)[1]<-"continent"
  check<-semi_join(stat_for_fill,df_to_fill,by=c("continent","date"))
  colnames(check)[1]<-"location"
  vec_ind<-c()
  for (i in seq(1:dim(check)[1])) {
    val<-as.character(check$location[i])
    val_2<-check$date[i]
    ind<-which(data_fr$location==val&data_fr$date==val_2)
    vec_ind<-c(vec_ind,ind)
   }
  col_vec[vec_ind]<-check[,3]
  data_fr[,col_ne]<-col_vec
  view(check)
  return(data_fr)
}

#B
filled_df<-fill_column(data,"new_tests_smoothed")
```

```
## Note: Using an external vector in selections is ambiguous.
## ℹ Use `all_of(col_ne)` instead of `col_ne` to silence this message.
## ℹ See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

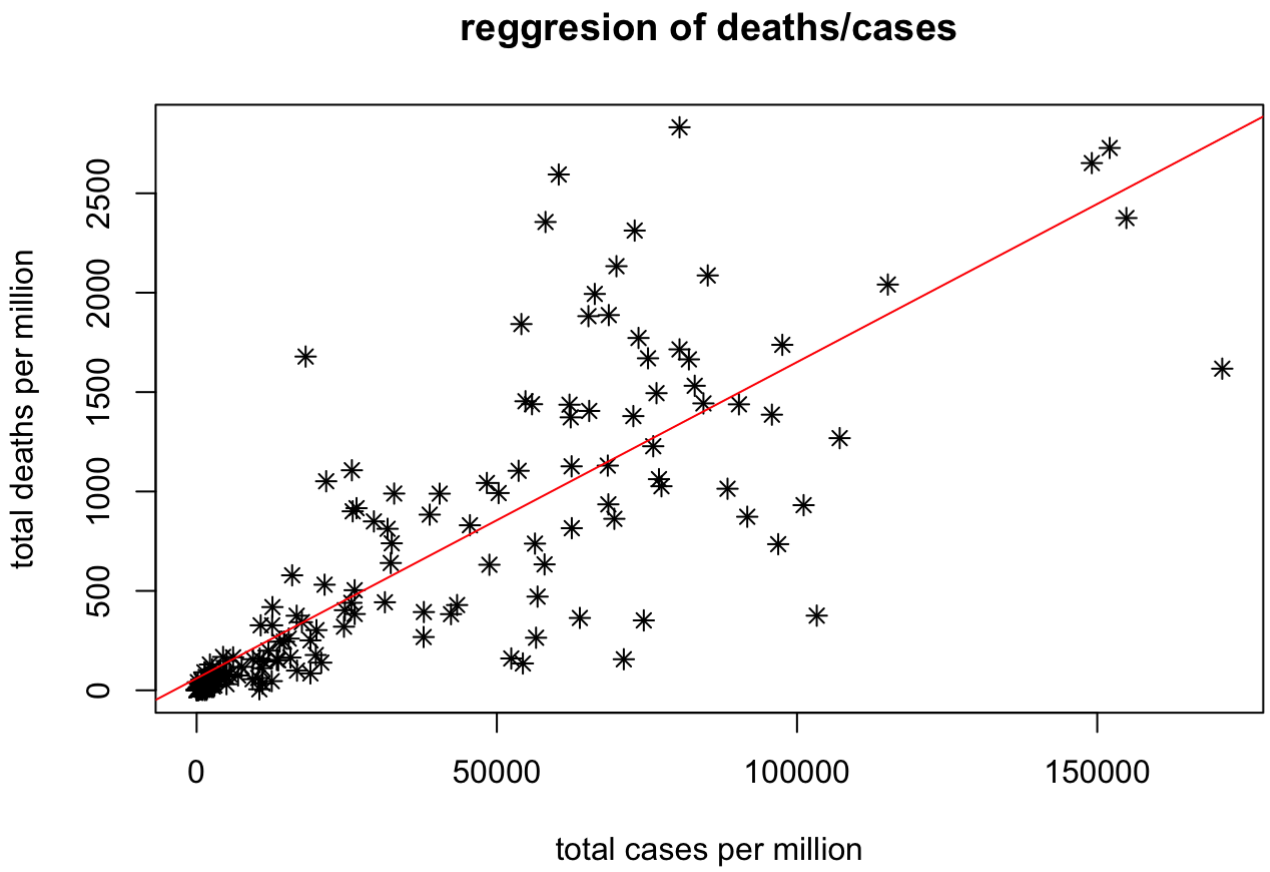
```
prep_to_log_4<-fun_log(filled_df["new_tests_smoothed"])
data_to_log_4<-filled_df[-prep_to_log_4,]
data_to_log_4$log_new_tests_smoothed<-log(data_to_log_4$new_tests_smoothed)
ploting(data_to_log_4,"log_new_tests_smoothed")
```



[MY SOLUTION TEXT - DESCRIPTION OF RESULTS]

5. [MY SOLUTION TEXT - EXPLANATIONS]

```
# R code for my solution

total_df<-aggregate(cbind(total_cases_per_million,total_deaths_per_million)~location,
    data=clean_conti,function(x) max(x,na.rm = TRUE))
plot(total_df$total_cases_per_million,total_df$total_deaths_per_million,main = "reggresion of
deaths/cases",xlab = "total cases per million",
     ylab = "total deaths per million",pch = 8)
rg_line<-lm(total_deaths_per_million~total_cases_per_million,data = total_df)
abline(rg_line,col="red")
```



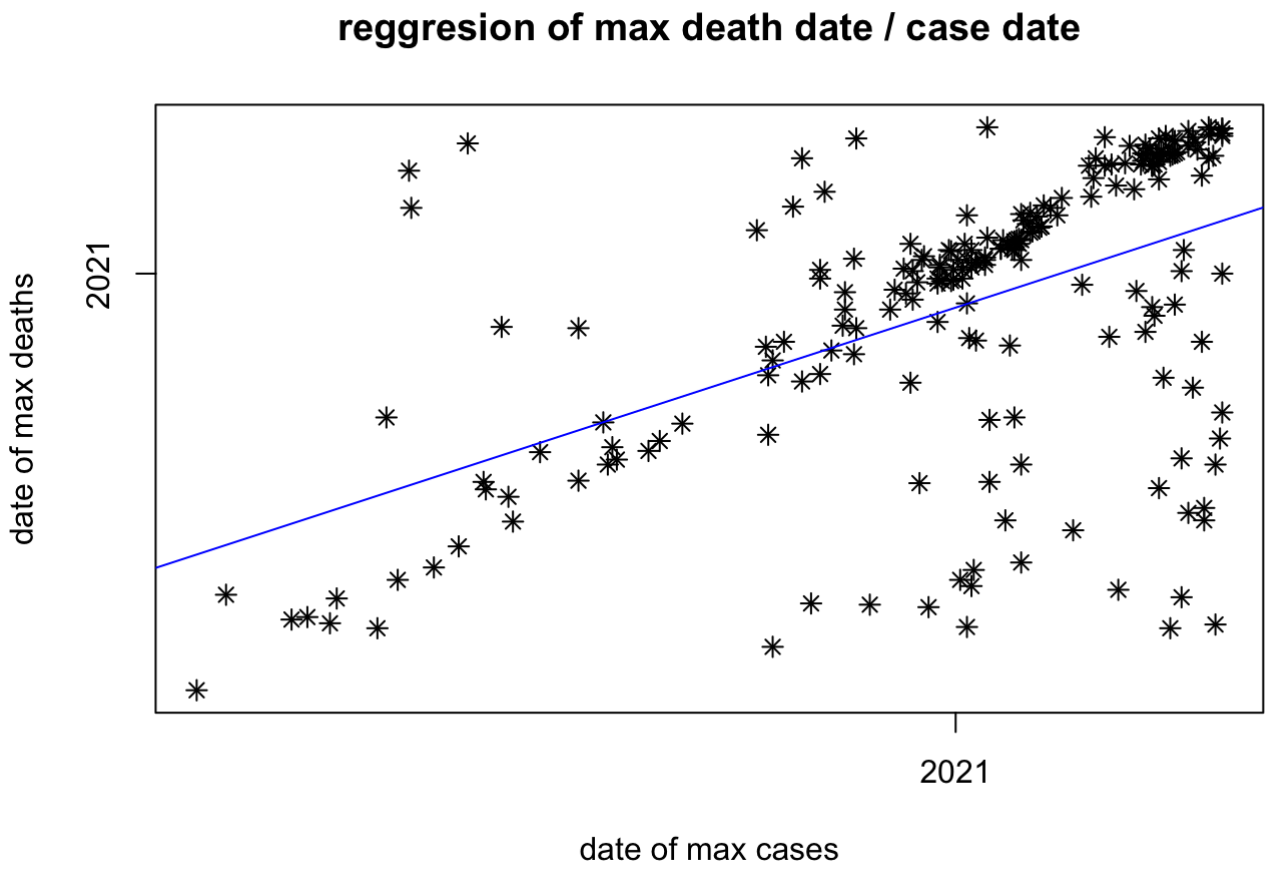

```
slope<-rg_line$coefficients

date_max_cases<-clean_conti %>% group_by(location) %>%
  slice_max(new_cases,with_ties = FALSE)%>%
  select(date)
```

```
## Adding missing grouping variables: `location`
```

```
date_max_deaths<-clean_conti %>% group_by(location) %>% slice_max(new_deaths,with_ties = FALS
E)%>%
  select(date)
```

```
## Adding missing grouping variables: `location`
```

```
plot(date_max_cases$date,date_max_deaths$date,main = "reggresion of max death date / case dat
e",xlab = "date of max cases",ylab = "date of max deaths",pch=8)
rg_line_2<-lm(date_max_deaths$date~date_max_cases$date)
abline(rg_line_2,col="blue")
```

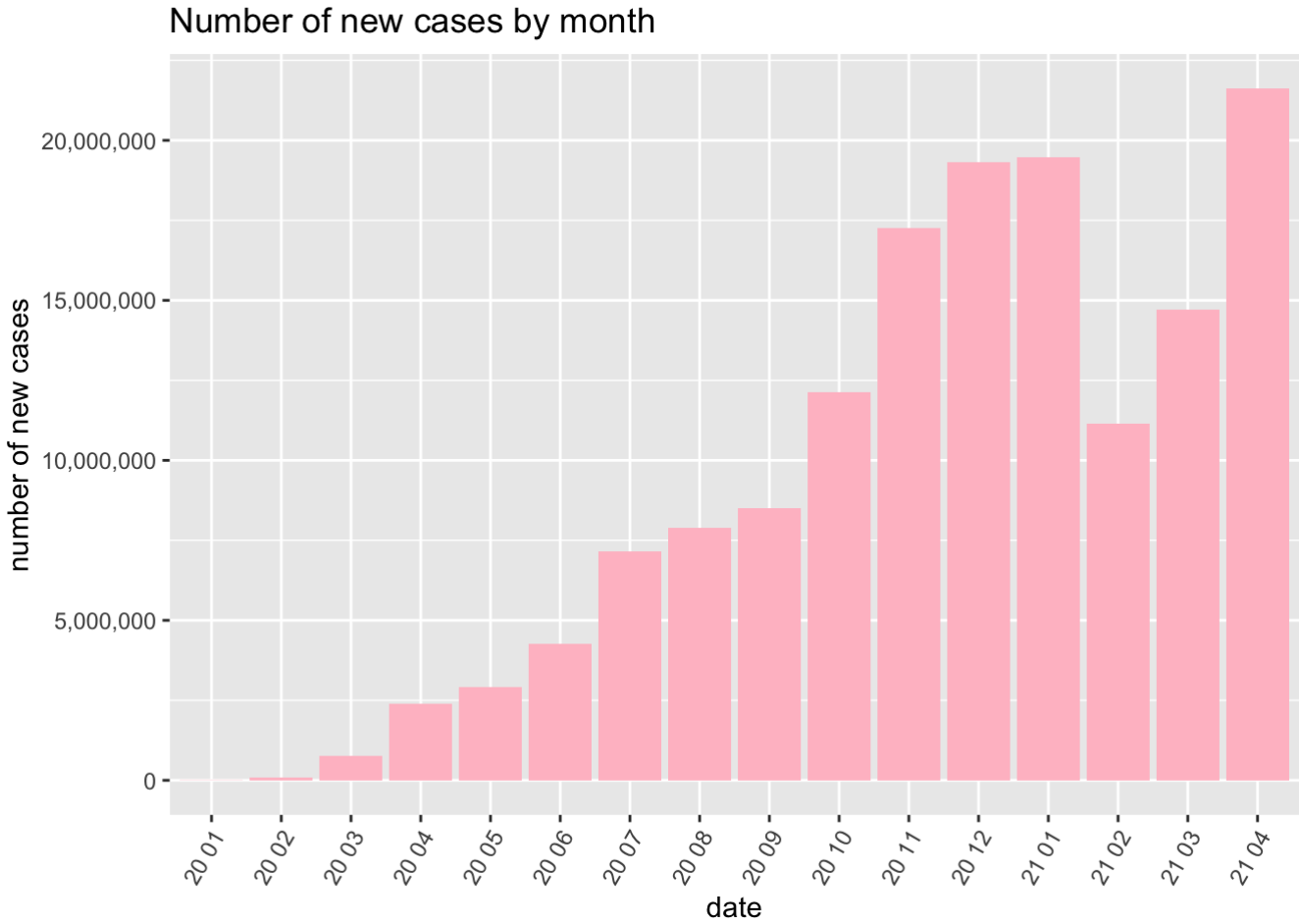



```
inter<- rg_line_2$coefficients[1]
```

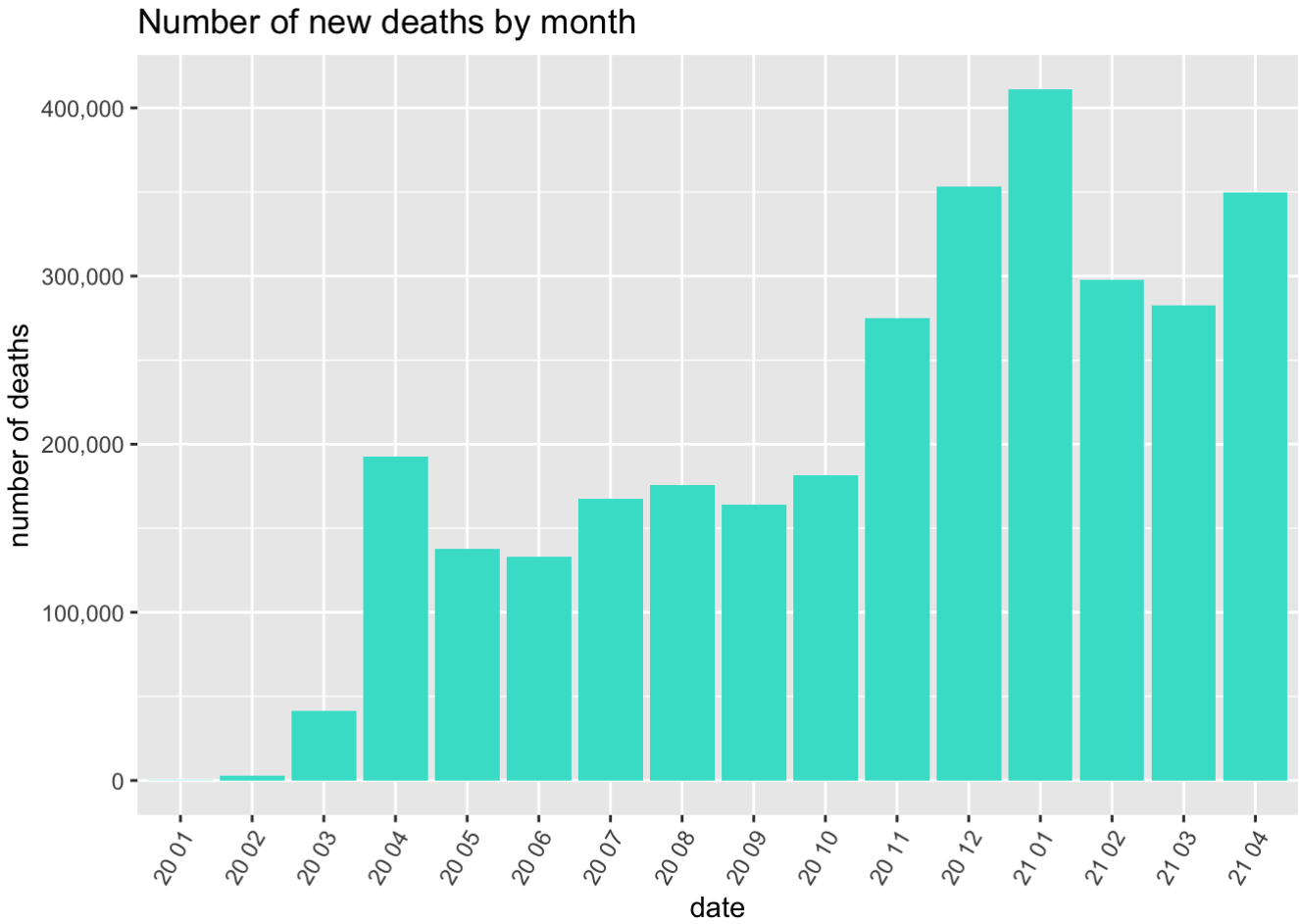
we can see that intrtsect is 7554.9108059

```
# R code for my solution
time_df<-data%>%filter(continent%in%vec_of_conti)
months<-strftime(time_df$date,"%m")
years<-strftime(time_df$date,"%y")
time_df<-time_df%>%select(new_cases,new_deaths,new_vaccinations)%>%
  mutate(month = months, year = years)%>%
  group_by(month, year)
ans<-aggregate(cbind(new_cases,new_deaths)~month+year,
               data = time_df,
               function(x) sum(x,na.rm = TRUE))
ans_2<-aggregate(new_vaccinations~month+year,
               data = time_df,
               function(x) sum(x,na.rm = TRUE))
monthly<-left_join(ans,ans_2,by = c("month", "year"))
monthly[is.na(monthly)] <- 0
monthly$date <- sprintf(paste(monthly$year, monthly$month), "%Y %m")
```

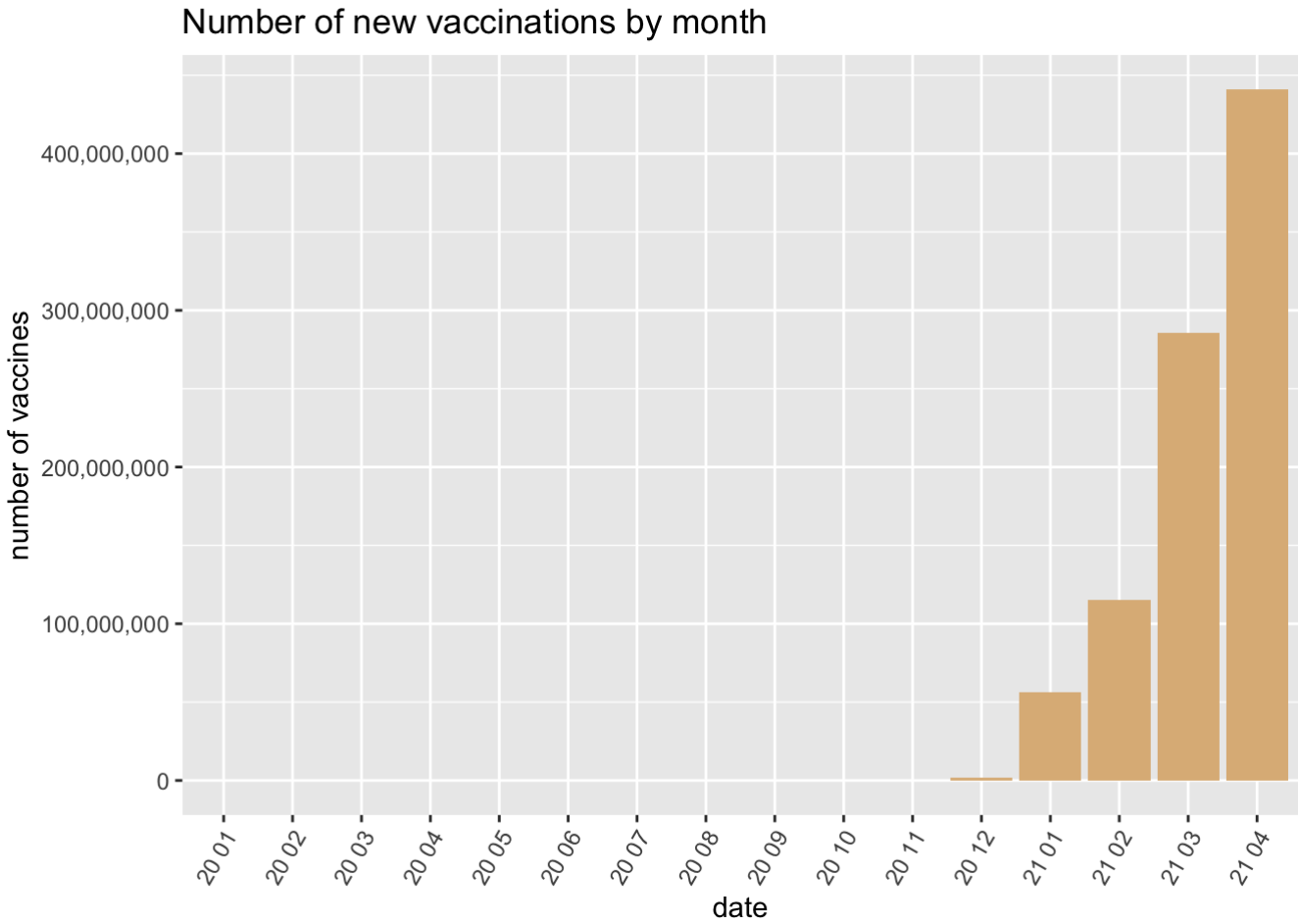
```
#ggPlot for new_cases
plot_newcases <- ggplot(monthly, aes(date, new_cases))+
  geom_bar(stat="identity", fill="pink") + theme(axis.text.x = element_text(angle = 60, hjust
= 1))+
  scale_y_continuous(name="number of new cases", labels = scales::comma)+
  labs(title="Number of new cases by month")

plot_newcases
```




```
#ggPlot for new_deaths
plot_newdeaths <-ggplot(monthly, aes(date, new_deaths))+
  geom_bar(stat="identity", fill="turquoise") + theme(axis.text.x = element_text(angle = 60,
 hjust = 1))+
  scale_y_continuous(name="number of deaths", labels = scales::comma)+
  labs(title="Number of new deaths by month")

plot_newdeaths
```




```
#ggPlot for vacc
plot_vacc <-ggplot(monthly, aes(date, new_vaccinations))+
  geom_bar(stat="identity", fill="burlywood") +
  theme(axis.text.x = element_text(angle = 60, hjust = 1))+
  scale_y_continuous(name="number of vaccines", labels = scales::comma)+
  labs(title="Number of new vaccinations by month")

plot_vacc
```

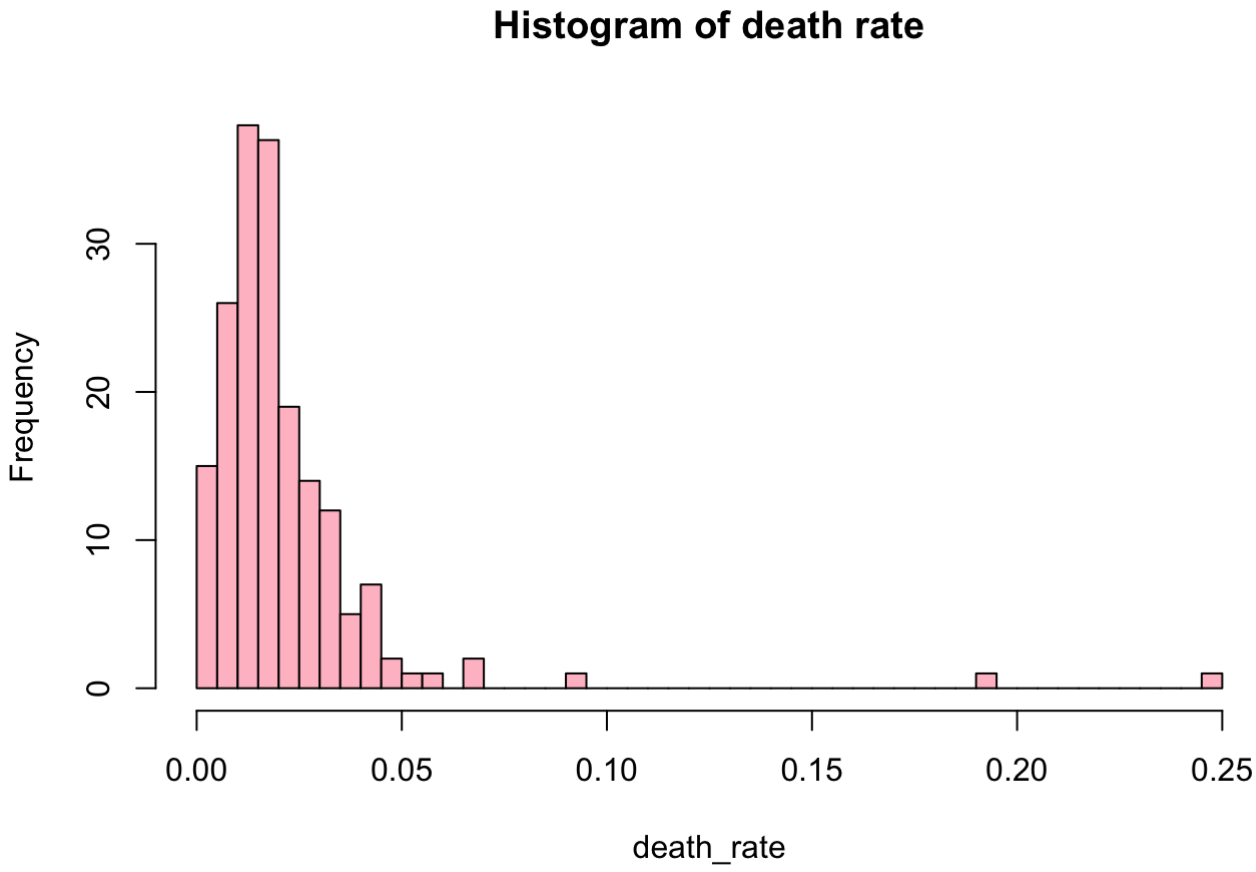


7. [MY SOLUTION TEXT - EXPLANATIONS]

```
# R code for my solution
data$death_rate <- data$total_deaths/data$total_cases
max_date_val_4<-data$date[max(which(!is.na(data$death_rate)))]
current_df<-filter(data,date==max_date_val_4)
current_df <- filter(current_df,continent%in% vec_of_conti)
summary(current_df$death_rate)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
## 0.000491 0.010782 0.016758 0.021563 0.025581 0.250000        8
```

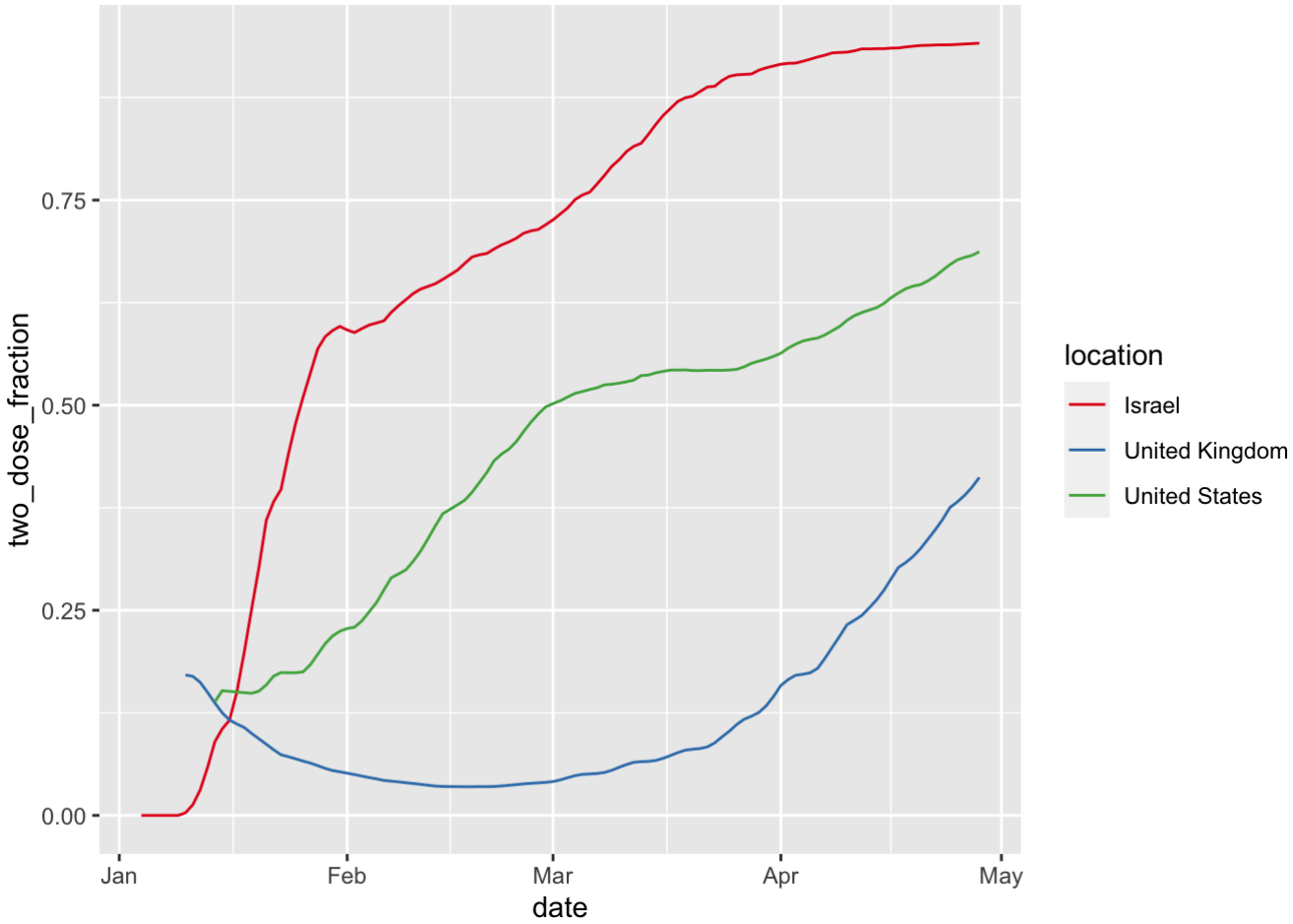
```
hist(current_df$death_rate,breaks = 50,col = "pink", border = "black", main = "Histogram of d
eath rate" , xlab = "death_rate")
```




```
top_d_rate<-current_df%>%arrange(desc(death_rate))%>%
  select(continent,location,date,death_rate)%>% head(3)
```

8. [MY SOLUTION TEXT - EXPLANATIONS]

```
# R code for my solution
data<-data%>%mutate(two_dose_fraction=people_fully_vaccinated/people_vaccinated)
vacc_strat<-data%>%filter(location%in%c("Israel","United Kingdom","United States"))
min_date_val<-vacc_strat$date[min(which(!is.na(vacc_strat$two_dose_fraction)))]
vacc_strat<-vacc_strat%>%filter(date>=min_date_val)
vacc_strat<-subset(vacc_strat,!is.na(two_dose_fraction))
ggplot(vacc_strat,aes(x=date,y=two_dose_fraction),na.omit())+
    geom_line(aes(color=location),size=0.5)+
    scale_color_brewer(palette = "Set1")
```



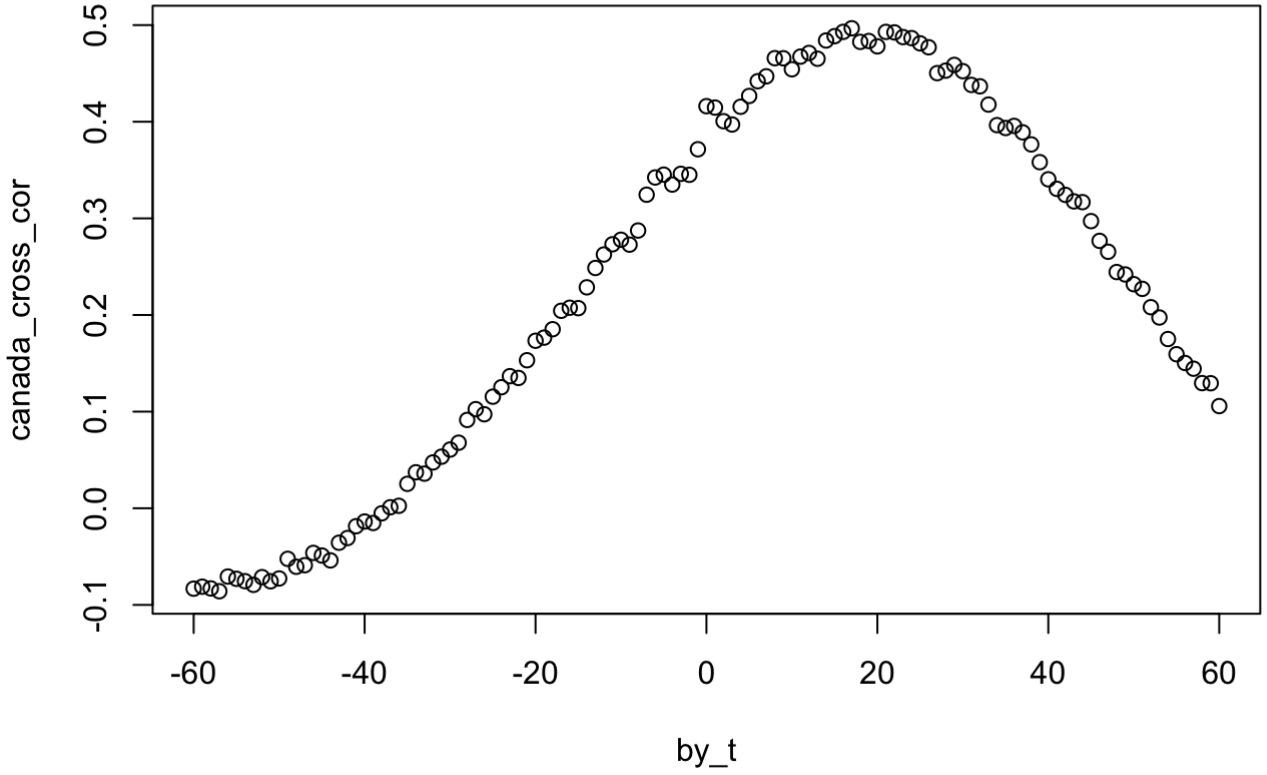
9. [MY SOLUTION TEXT - EXPLANATIONS]

```
# R code for my solution
cross_cor <- function(data_f,country,col_name_1,col_name_2){
  rel_loca<-filter(data_f,location==country)
  X<-rel_loca[[col_name_1]]
  Y<-rel_loca[[col_name_2]]
  min_date<-min(rel_loca$date)
  max_date<-max(rel_loca$date)
  neg_vec<-seq(from=60,to=0,by=-1)
  pos_vec<-seq(1:60)
  cross_cor_vec=c()
  for (i in neg_vec) {
    delta_t_y<-seq(min_date,max_date-i,by="days")
    delta_t_x<-seq(min_date+i,max_date,by="days")
    y_t<-rel_loca%>%filter(date%in%delta_t_y)%>%select(col_name_2)
    x_t<-rel_loca%>%filter(date%in%delta_t_x)%>%select(col_name_1)
    cross_cor_vec=c(cross_cor_vec,cor(y_t,x_t,use = "complete.obs"))
  }
  for (i in pos_vec) {
    delta_t_x<-seq(min_date,max_date-i,by="days")
    delta_t_y<-seq(min_date+i,max_date,by="days")
    x_t<-rel_loca%>%filter(date%in%delta_t_x)%>%select(col_name_1)
    y_t<-rel_loca%>%filter(date%in%delta_t_y)%>%select(col_name_2)
    cross_cor_vec=c(cross_cor_vec,cor(x_t,y_t,use = "complete.obs"))
  }
  return(cross_cor_vec)
}
#B
by_t<-seq(from=-60,to=60,by=1)
canada_cross_cor<-cross_cor(data,"Canada","new_cases","new_deaths")
```

```
## Note: Using an external vector in selections is ambiguous.
## ℹ Use `all_of(col_name_2)` instead of `col_name_2` to silence this message.
## ℹ See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
## Note: Using an external vector in selections is ambiguous.
## ℹ Use `all_of(col_name_1)` instead of `col_name_1` to silence this message.
## ℹ See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

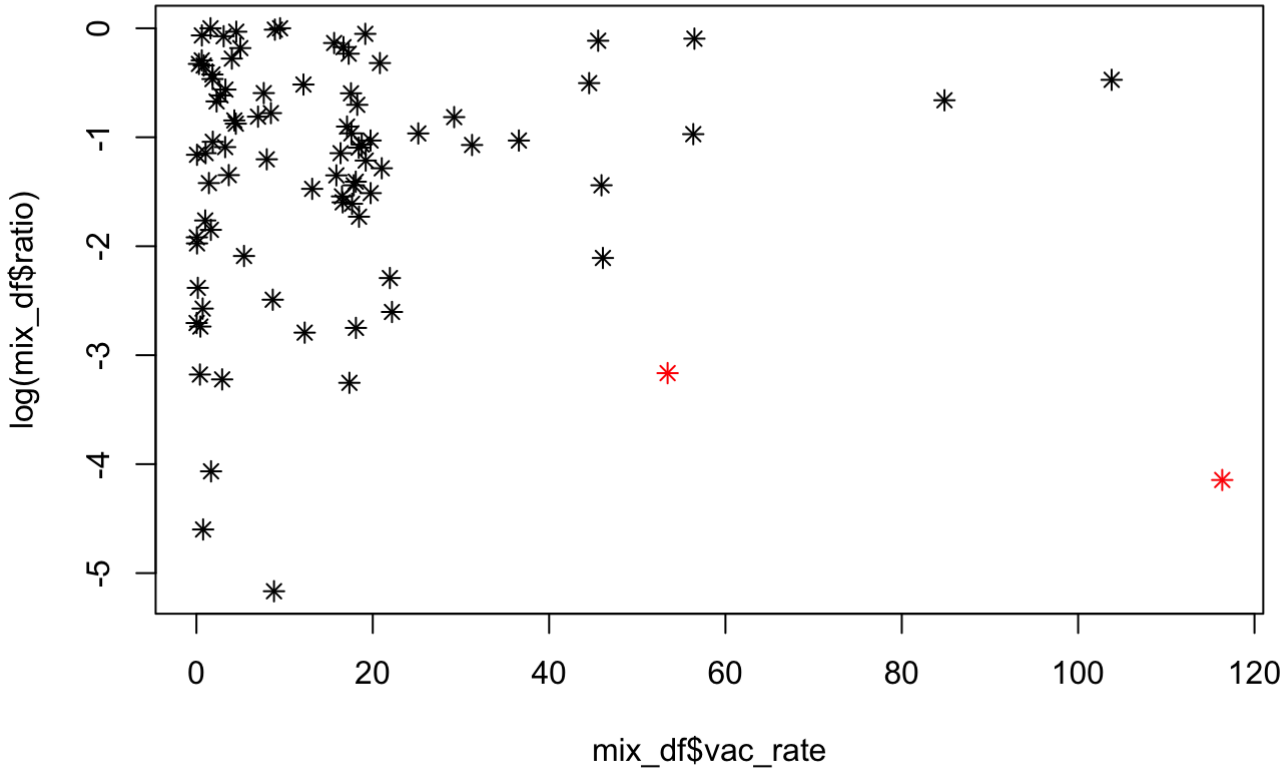
```
plot(by_t,canada_cross_cor)
```




```
max_cross_cor<-by_t[which.max(canada_cross_cor)]
```

10. [MY SOLUTION TEXT - EXPLANATIONS]

```
clean_conti<-filter(data,continent%in% vec_of_conti)
clean_conti$new_cases_smoothed[is.na(clean_conti$new_cases_smoothed)] <- 0
current_per_conti<-clean_conti%>%filter(date=="2021-04-23")%>%
  select(location,new_cases_smoothed)
colnames(current_per_conti)[2]<-"current_smoothed"
max_per_contin<-clean_conti %>% group_by(location)%>%
  slice_max(new_cases_smoothed,with_ties = FALSE)%>%
  select(location,new_cases_smoothed)
max_per_contin<-filter(max_per_contin,location%in%current_per_conti$location)
colnames(max_per_contin)[2]<-"max_smoothed"
effect_of_vacc<-current_per_conti$current_smoothed/max_per_contin$max_smoothed
current_vac_per_conti<-clean_conti%>%filter(date=="2021-04-01")%>%
  select(location,total_vaccinations_per_hundred)%>%
  filter(location%in%current_per_conti$location)
mix_df<-full_join(current_per_conti,max_per_contin,by="location")
mix_df$ratio<-effect_of_vacc
mix_df<-mix_df[-186,]
mix_df$vac_rate<-current_vac_per_conti$total_vaccinations_per_hundred
mix_df<-na.omit(mix_df)
plot(mix_df$vac_rate,log(mix_df$ratio),
col=ifelse(mix_df$location=="Israel"|mix_df$location=="United Kingdom","red","black"),pch=8)
```


log(mix_df$ratio)
0
-1
-2
-3
-4
-5
0
20
40
60
80
100
120
mix_df$vac_rate