derivative

Qi Yuchen, yq2279

2020/4/23

data import and cleaning

```
df.raw = read.csv("covid19-1.csv")
df = df.raw %>%
  janitor::clean_names() %>%
  dplyr::select(country_region, province_state, date, confirmed_cases) %>%
  filter(confirmed_cases != 0)
```

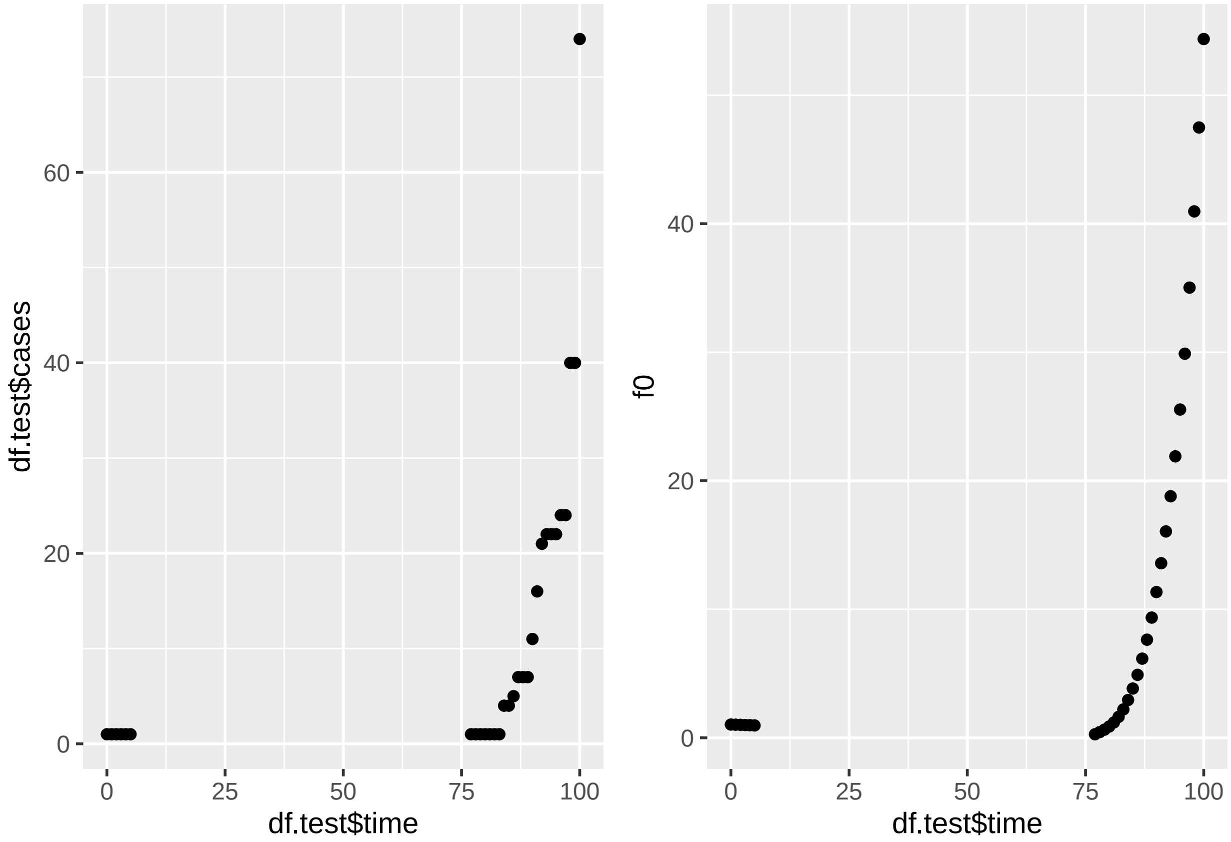
```
## Warning in FUN(X[[i]], ...): strings not representable in native encoding
## will be translated to UTF-8
```

```
region.index = levels(df$country_region)

# first step to write a function
df.china = df %>%
  filter(country_region == region.index[30]) %>%
  group_by(country_region, date) %>%
  summarise(cases = sum(confirmed_cases)) %>%
  mutate(formal_date = as.numeric(as.character(as.Date(date, '%m/%d/%Y'), format="%Y%m%d"))) %>%
  mutate(time = formal_date-min(formal_date)) %>%
  dplyr::select(region = country_region, date, time, cases) %>%
  arrange(time)

# function to get data for a specific region
df.region = function(df, region) {
  df.r = df %>%
    filter(country_region == region) %>%
    group_by(country_region, date) %>%
    summarise(cases = sum(confirmed_cases)) %>%
    mutate(formal_date = as.numeric(as.character(as.Date(date, '%m/%d/%Y'), format="%Y%m%d"))) %>%
    mutate(time = formal_date-min(formal_date)) %>%
    dplyr::select(region = country_region, date, time, cases) %>%
    arrange(time)

  df.r
}


df.china = df.region(df, region.index[30])
```

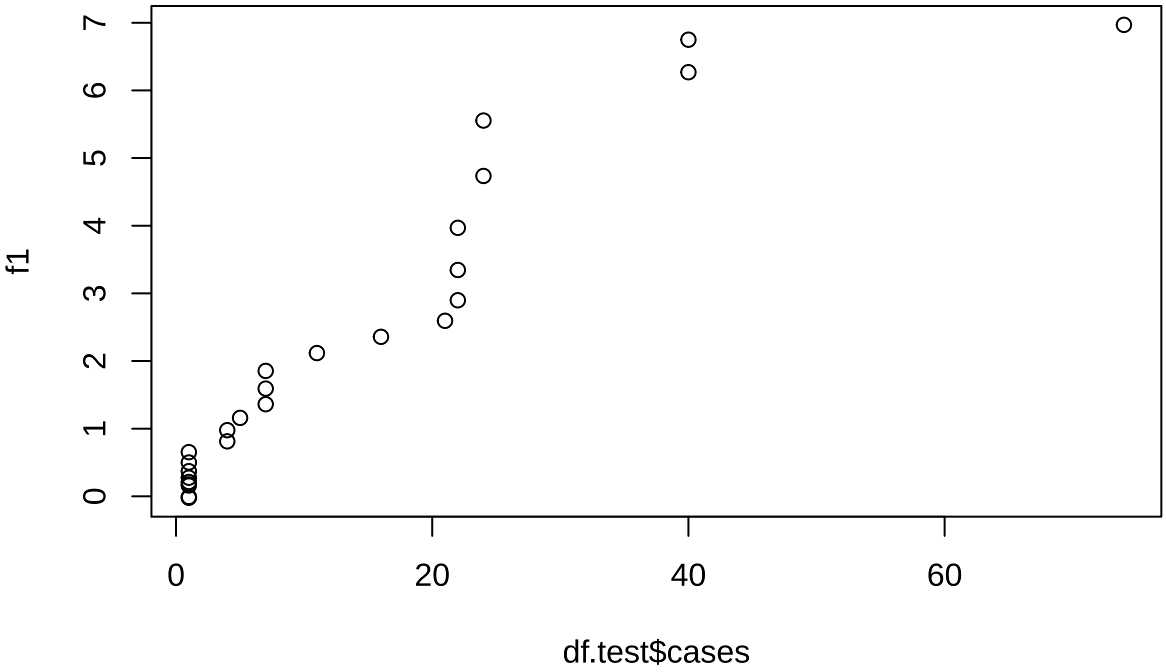
analyze

```
df.test = df.region(df, region.index[1])

p1 = ggplot(mapping = aes(x = df.test$time, y = df.test$cases)) + geom_point()
smooth.fit = smooth.Pspline(x = df.test$time, y = df.test$cases,df=6, method=2)
```

```
f0 = predict(smooth.fit, df.test$time, nderiv=0)
f1 = predict(smooth.fit, df.test$time, nderiv=1)
p2 =  ggplot(mapping = aes(x = df.test$time, y = f0)) + geom_point()
p1 + p2
```



```
plot(x = df.test$cases, y = f1)
```


f1
0
1
2
3
4
5
6
7
0
20
40
60
df.test$cases