

```
!pip install altair
import altair as alt
import pandas as pd
import numpy as np
import requests
```

```
Requirement already satisfied: altair in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: toolz in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: entrypoints in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: pandas>=0.18 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: jsonschema in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: Jinja2 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: Pytz>=2017.2 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages
Requirement already satisfied: MarkupSafe>=0.23 in /usr/local/lib/python3.7/dist-packages
```

## ▼ Motivation

Covid-19 is an infectious disease which pose a threat to human's health. Although there are several kinds of vaccines invented recently, we are still facing risks of infection, because the virus are mutating and the vaccines are not 100% effective. Under such circumstances, now we are in the long chasing game of monitering and making plans for fighting the virus.

In this study, I want to explore whether the United Kingdom has responsive and effective policies for protecting people's health from Covid-19. This country is chosen because it has 4 different regions recorded which will be easy interesting to compare and doesn't has so many missing values.

## Dataset

The Oxford Covid-19 Government Response Tracker (OxCGRT) collects systematic information on different governments' policies and their corresponding time points, which helps us understand their responsiveness and effectiveness.

The OxCGRT tracks several different common policy responses governments have taken, records these policies on a scale to reflect the extent of government action, and aggregates these scores into a suite of policy indices.

In this study, because there are so many policies, we will use the aggregated policy indices to represent different types of policies. The indices are [Government response index, Containment and health index, Stringency index, Economic support index, Legacy stringency index]. Details about the policies is [here](#). Details about these indices is [here](#).

The data source and reference are list below:

Github Link:

<https://github.com/OxCGRT/covid-policy-tracker>

## Citation:

Thomas Hale, Noam Angrist, Rafael Goldszmidt, Beatriz Kira, Anna Petherick, Toby Phillips, Samuel Webster, Emily Cameron-Blake, Laura Hallas, Saptarshi Majumdar, and Helen Tatlow. (2021). "A global panel database of pandemic policies (Oxford COVID-19 Government Response Tracker)." Nature Human Behaviour. <https://doi.org/10.1038/s41562-021-01079-8>

To sum up,

- 1) the dataset I am using is the [Oxford Covid-19 Government Response Tracker \(OxCGRT\)](#);
- 2) my question is, in the United Kingdom, whether the policy is responsive and effective enough to fight the pandemic.

## ▼ Methodology

In this study, I will use three platforms to generate three different visualizations to answer this question. I will use 1) Altair, 2) Charticulator and 3) D3 to visualize the data separately. The details will be introduced in the following parts.

```
#data = pd.read_csv('https://raw.githubusercontent.com/OxCGRT/covid-policy-tracker/
# preprocessing
data = pd.read_csv('https://raw.githubusercontent.com/OxCGRT/covid-policy-tracker/m
## remove missing values
na_deleted = data.dropna(axis=0, how='any', inplace=False)
## select the data (UK data) we need
select_by_country = na_deleted[na_deleted.values[:, 0]=='United Kingdom']
## transform date format
date = select_by_country['Date'].values
current_dates = []
for i in range(len(date)):
    current_date = str(date[i])
    current = current_date[0:4] + '-' + current_date[4:6] + '-' + current_date[6:8]
    current_dates.append(current)
select_by_country['Date'] = current_dates
## calculate new cases for each day
arr_new_cases = np.zeros_like(select_by_country['ConfirmedCases'].values)
arr_new_cases[0] = select_by_country['ConfirmedCases'].values[0]
for i in range(len(arr_new_cases)-1):
    if select_by_country['RegionName'].values[i+1] == select_by_country['RegionName
        arr_new_cases[i + 1] = select_by_country['ConfirmedCases'].values[i+1] - se
    else:
        arr_new_cases[i + 1] = select_by_country['ConfirmedCases'].values[i + 1]
select_by_country['NewCasesByDay'] = arr_new_cases

select_by_country.to_csv('/content/oxford_policy_uk.csv', index=0)
```

```
/usr/local/lib/python3.7/dist-packages/IPython/core/interactiveshell.py:2718:
  interactivity=interactivity, compiler=compiler, result=result)
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:16: SettingWithCo
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: <https://pandas.pydata.org/pandas-docs/st>  
 app.launch\_new\_instance()

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:25: SettingWithCo
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

See the caveats in the documentation: <https://pandas.pydata.org/pandas-docs/st>

```
data = pd.read_csv('/content/oxford_policy_uk.csv', header=0)
data.head()
```

	CountryName	CountryCode	RegionName	RegionCode	Jurisdiction	Date	C1_
0	United Kingdom	GBR	England	UK_ENG	STATE_TOTAL	2020-03-06	
1	United Kingdom	GBR	England	UK_ENG	STATE_TOTAL	2020-03-07	
2	United Kingdom	GBR	England	UK_ENG	STATE_TOTAL	2020-03-08	
3	United Kingdom	GBR	England	UK_ENG	STATE_TOTAL	2020-03-09	
4	United Kingdom	GBR	England	UK_ENG	STATE_TOTAL	2020-03-10	

## ▼ Altair

The first visualization is a line chart plotted in Altair platform to mirror the responsiveness and effectiveness of the policies in terms of reducing daily new Covid-19 cases across time. Each of the four regions in the UK is plotted separately to show whether the policies and disease progression are consistent across regions in the country.

```
alt.data_transformers.disable_max_rows()

chart1 = alt.Chart(data).mark_line(color='blue').encode(
    alt.X('Date:T'),
    alt.Y('GovernmentResponseIndex:Q', axis=alt.Axis(title=None)),
    tooltip=['Date:T', 'GovernmentResponseIndex:Q'])

chart2 = alt.Chart(data).mark_line(color='yellow').encode(
    alt.X('Date:T'),
    alt.Y('StringencyIndex:Q', axis=alt.Axis(title=None, labels=False, ticks=False)),
    tooltip=['Date:T', 'StringencyIndex:Q'])
```

```
chart3 = alt.Chart(data).mark_line(color= green ).encode(
    alt.X('Date:T'),
    alt.Y('StringencyLegacyIndex:Q', axis=alt.Axis(title=None, labels=False, ticks=
tooltip=[ 'Date:T', 'StringencyLegacyIndex:Q' ]
)

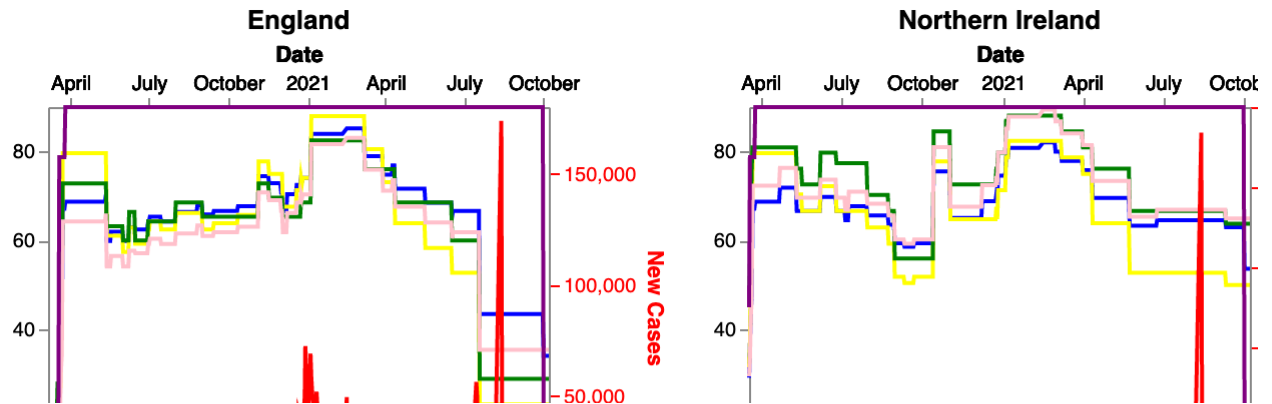
chart4 = alt.Chart(data).mark_line(color='pink').encode(
    alt.X('Date:T'),
    alt.Y('ContainmentHealthIndex:Q', axis=alt.Axis(title=None, labels=False, ticks=
tooltip=[ 'Date:T', 'ContainmentHealthIndex:Q' ]
)

chart5 = alt.Chart(data).mark_line(color='purple').encode(
    alt.X('Date:T'),
    alt.Y('EconomicSupportIndex:Q', axis=alt.Axis(title=None, labels=False, ticks=F
tooltip=[ 'Date:T', 'EconomicSupportIndex:Q' ]))

chart_new_case = alt.Chart(data).mark_line(color='red').encode(
    alt.X('Date:T'),
    alt.Y('NewCasesByDay:Q', axis=alt.Axis(title='New Cases', titleColor='red', tic
tooltip=[ 'Date:T', 'NewCasesByDay:Q' ]))

alt.concat(*(
    alt.layer(chart1, chart2,
        chart3, chart4,
        chart5, chart_new_case,
        title=f"{val}",
        width=250, height=200)\
        .transform_filter(alt.datum.RegionName == val).resolve_scale(y='indepe
    for val in data.RegionName.unique()
)).properties(columns=2,
    title='Policies')
```

## Policies



## Analysis

From the figure above, we can observe that the disease progression and the policies are very similar across regions. Generally, at the beginning of 2021, there was a peak for new cases in all the 4 regions. Correspondingly, the policies became stricter at the time point of this peak. Then, the daily new cases decreased and the policies are becoming less strict after that.

From July 2021 to October 2021, the policies became much more relaxed than before. However, the daily new cases increased dramatically after the policies are relaxed.

As a result, the policies are not always responsive to the disease progression (daily new cases). And the policies are not effective in a long-term perspective, because they didn't reduce the speed of the disease progression.

## Charticulator

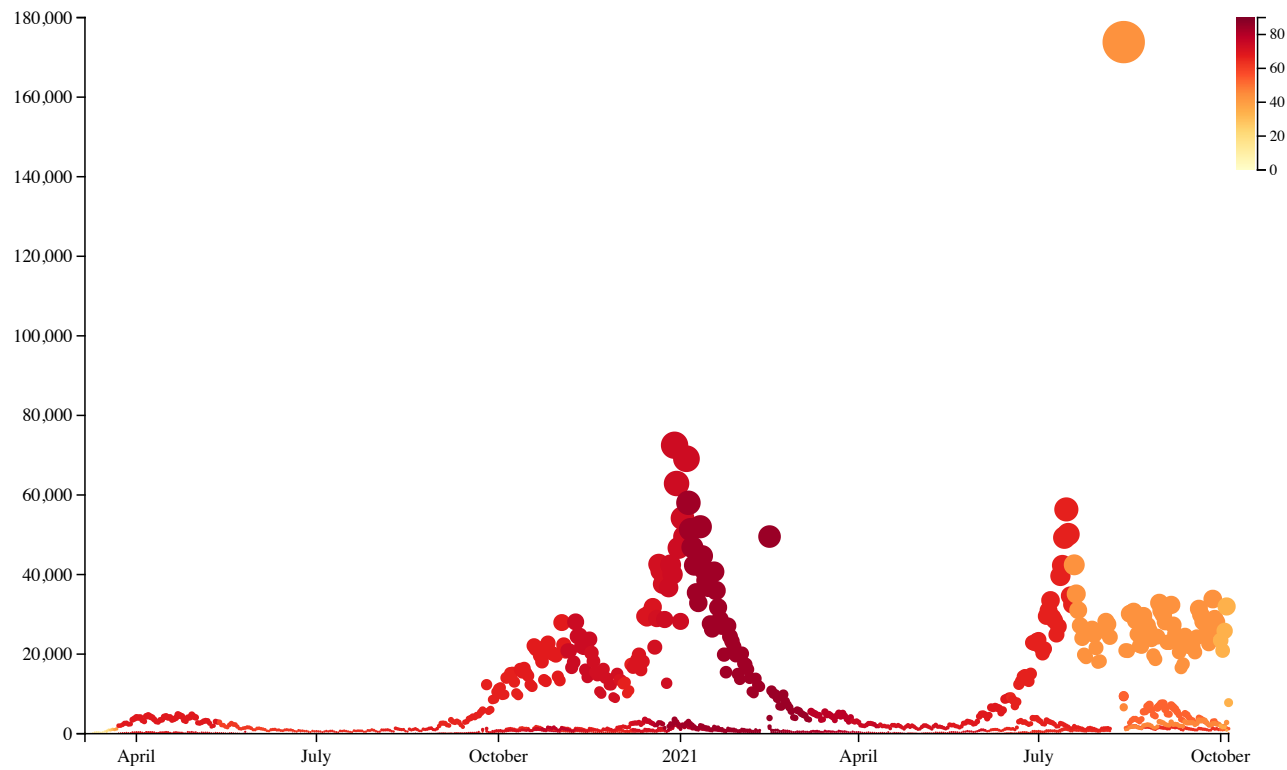
Charticulator is used to make a scatter plot to show how the responsiveness and effectiveness of policies change across time.

The color represents different levels of the policies. Briefly speaking, the redder the color is, the stricter the policies are set. The height of the scatters and the size of the scatters both represent the number of daily new cases.

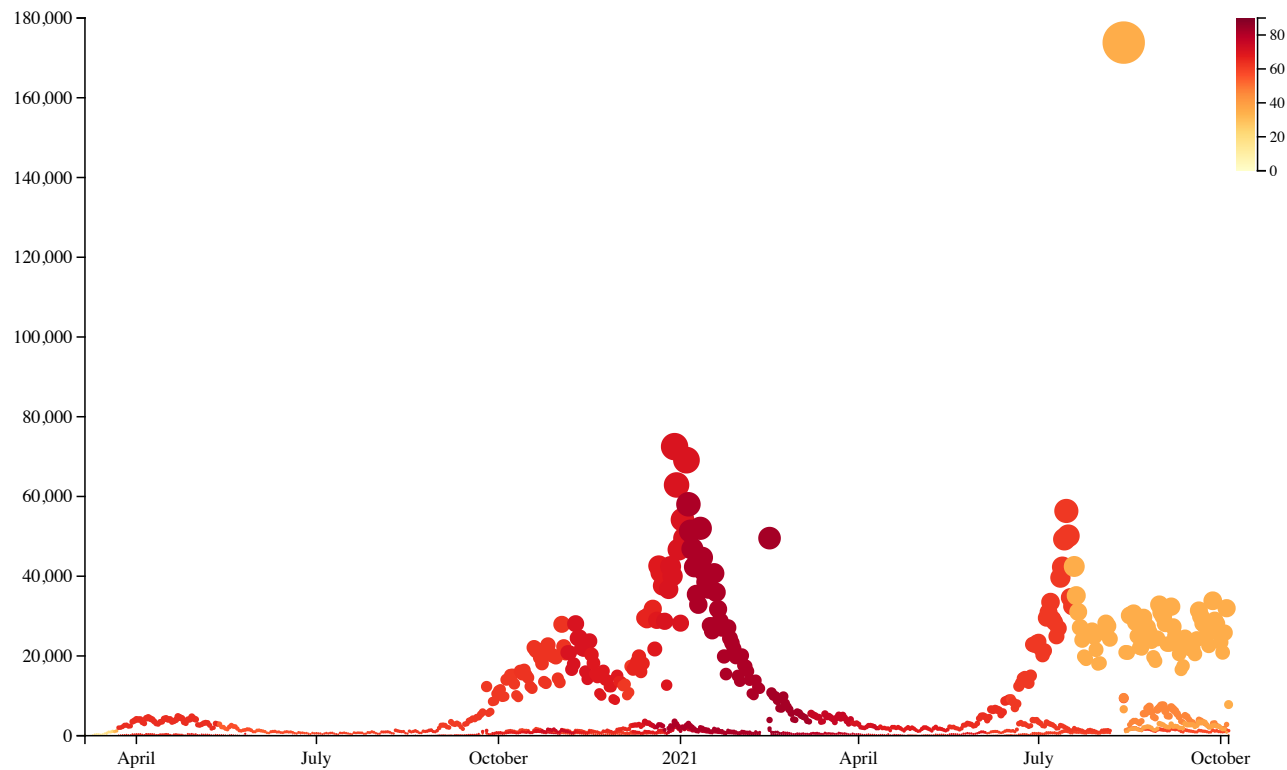
## Analysis

To summarize, except for Economic Support, the policies are largely relaxed after July, 2021. However, the pandemic did not leave.

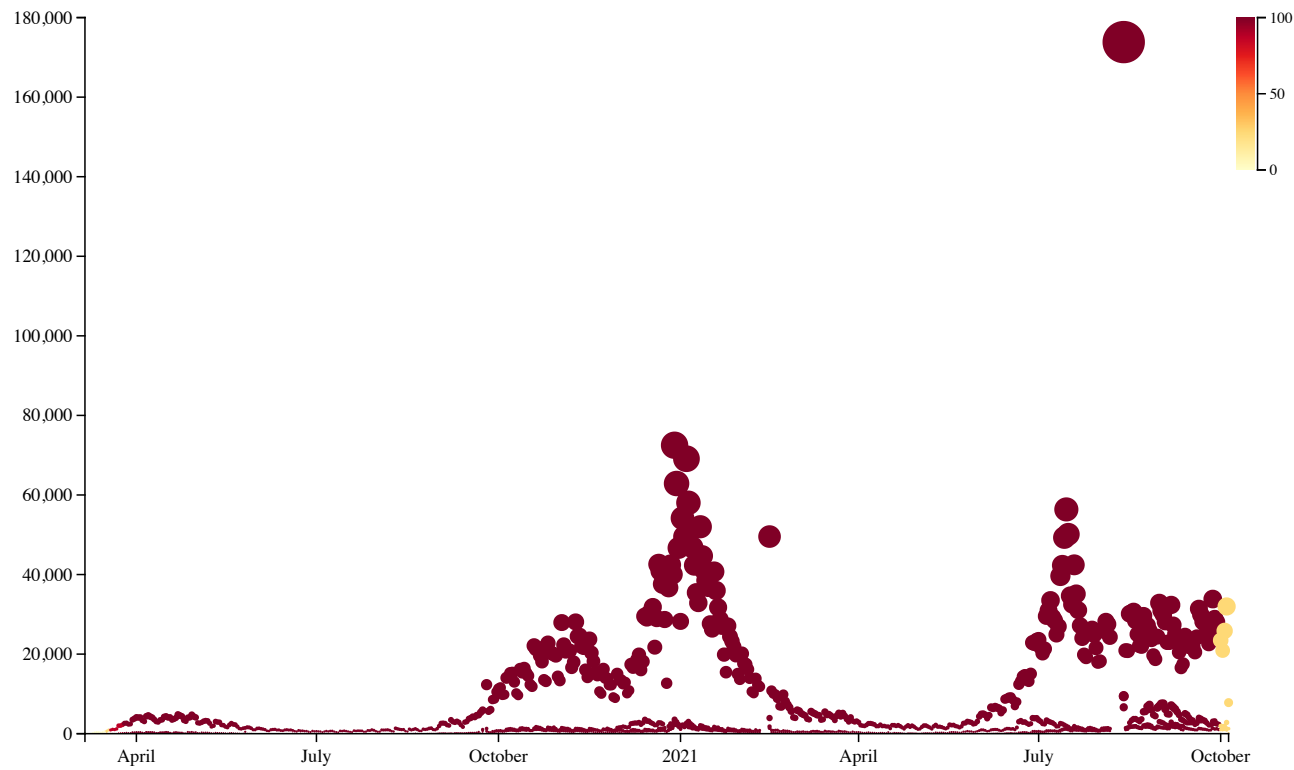
Responsiveness of Covid-19 Related Policies in UK (Governmmnet Response)



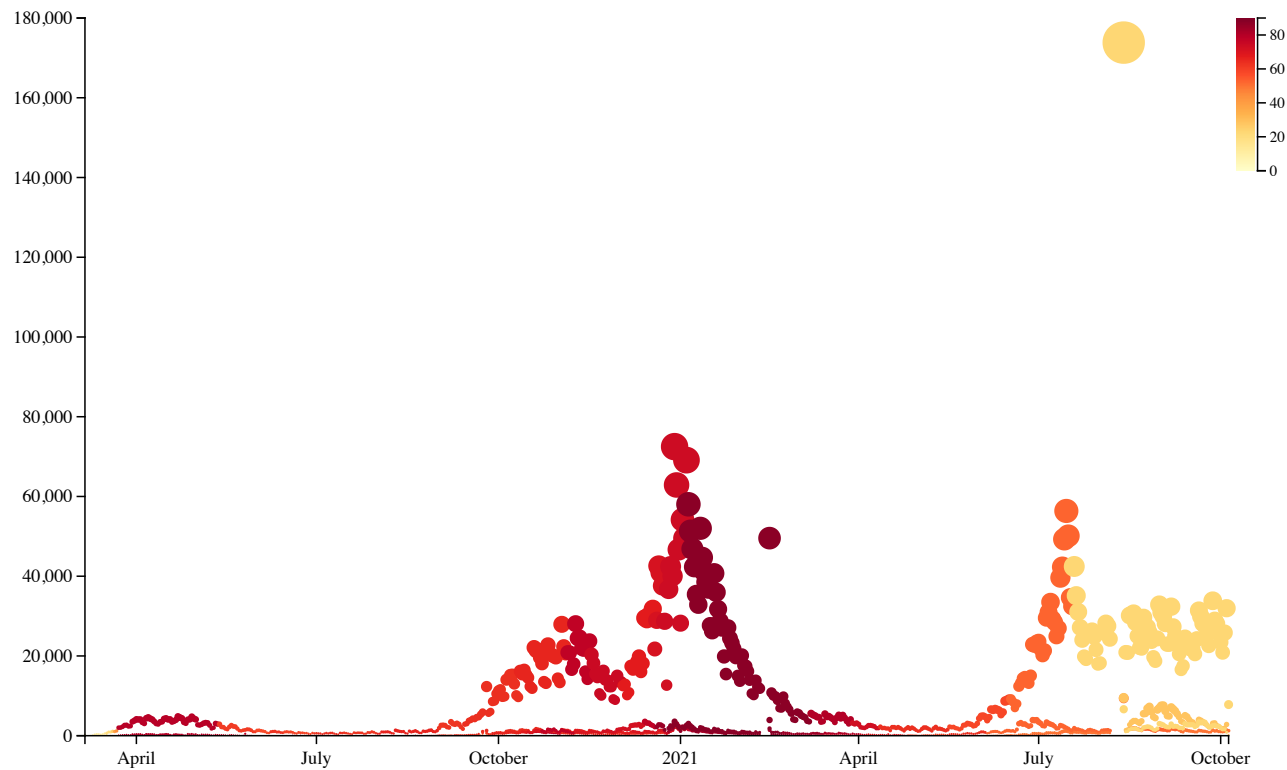
Responsiveness of Covid-19 related policies in UK (Containmmnet and Health)



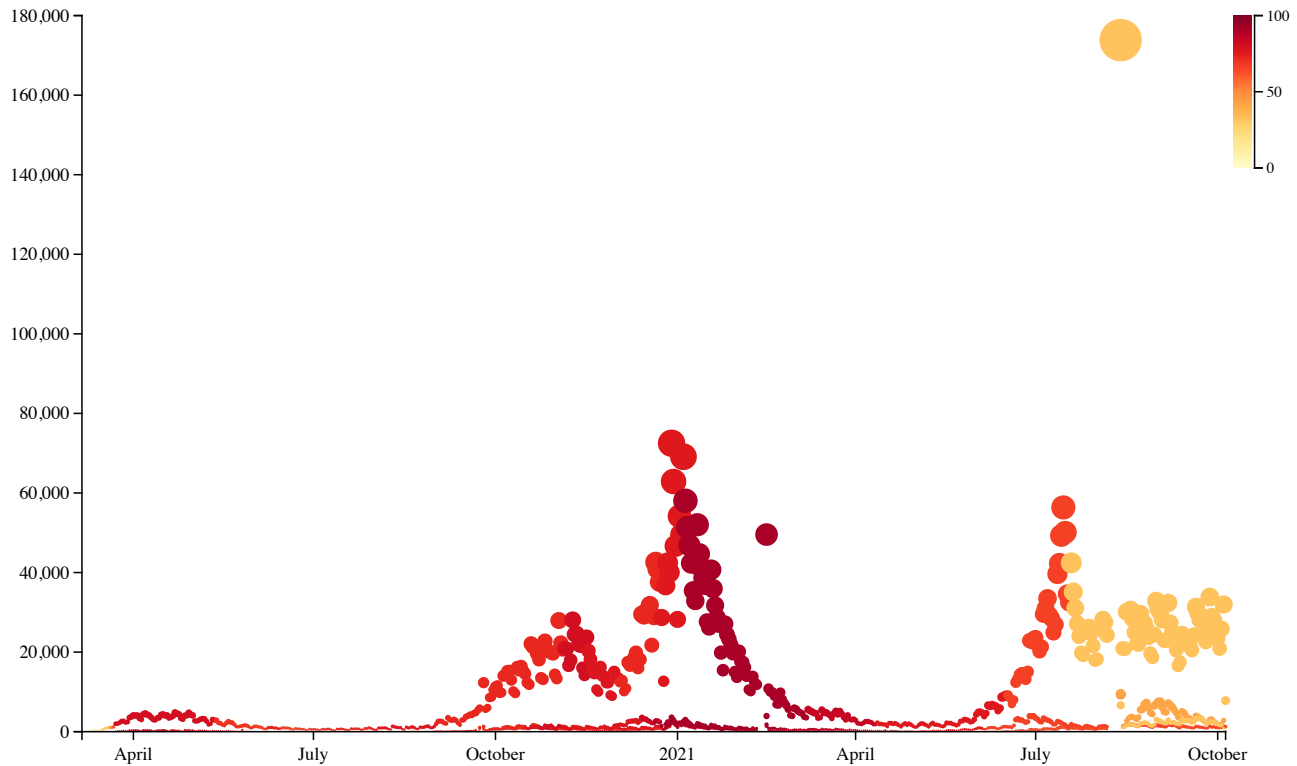
Responsiveness of Covid-19 Related Policies in UK (Economic Support)



Responsiveness of Covid-19 Related Policies in UK (Stringency)



## Responsiveness of Covid-19 Related Policies in UK (Stringency Legacy)



### D3

D3 is used to visualize the correlation between the daily new cases and policy strictness. Different colors of scatters represents different regions in the UK. If the scatters look like a straight line, then there is a strong correlation between policies and disease progression (daily new cases) and we can infer that the policy maker is using the daily new cases to make the decision. Otherwise, there might be other factors which may influence the policy making process.

The HTML file (including the interactive part) can be accessed with the 5 hyperlinks below.

[Government Response Index](#)

[Containment Health Index](#)

[Stringency](#)

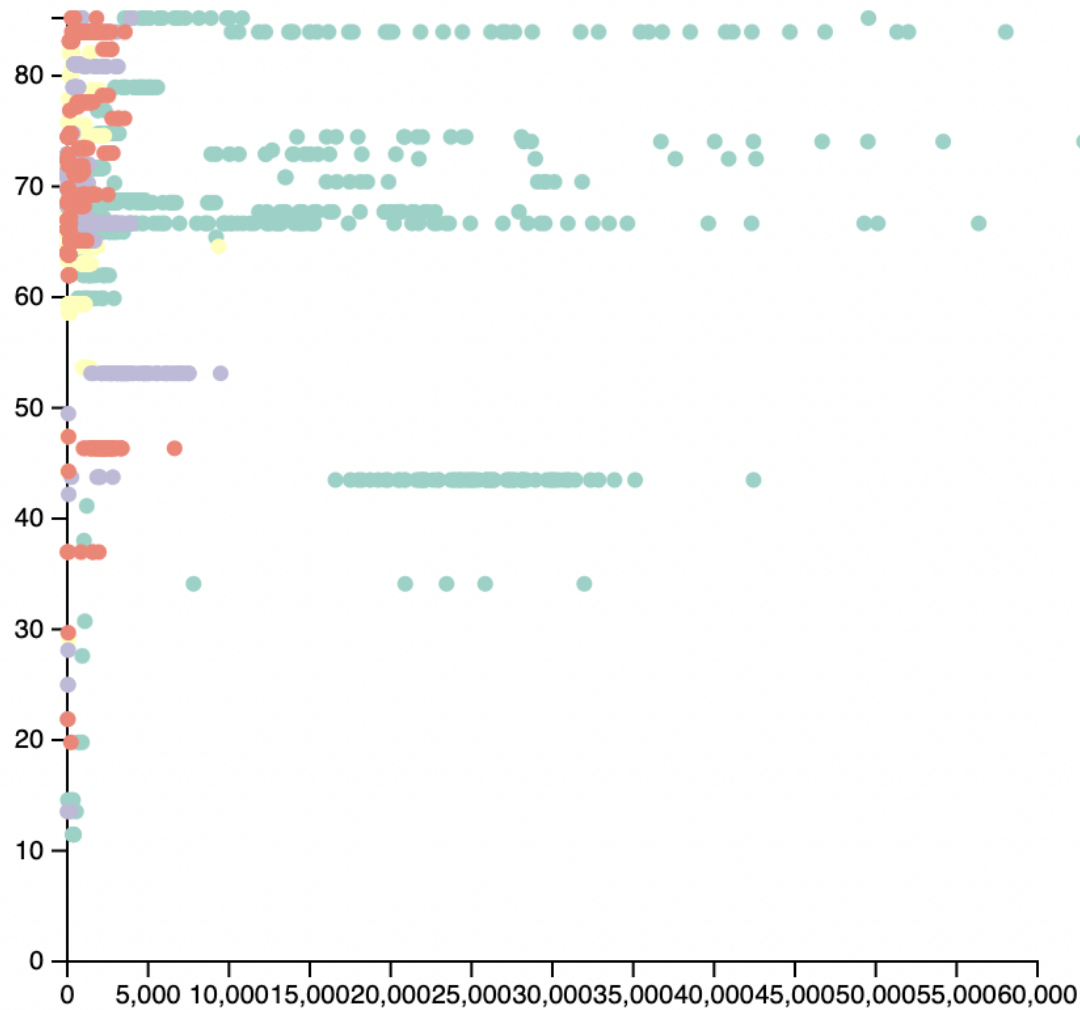
[Stringency Legacy](#)

[Economic Support](#)

Screenshots are also shown below.

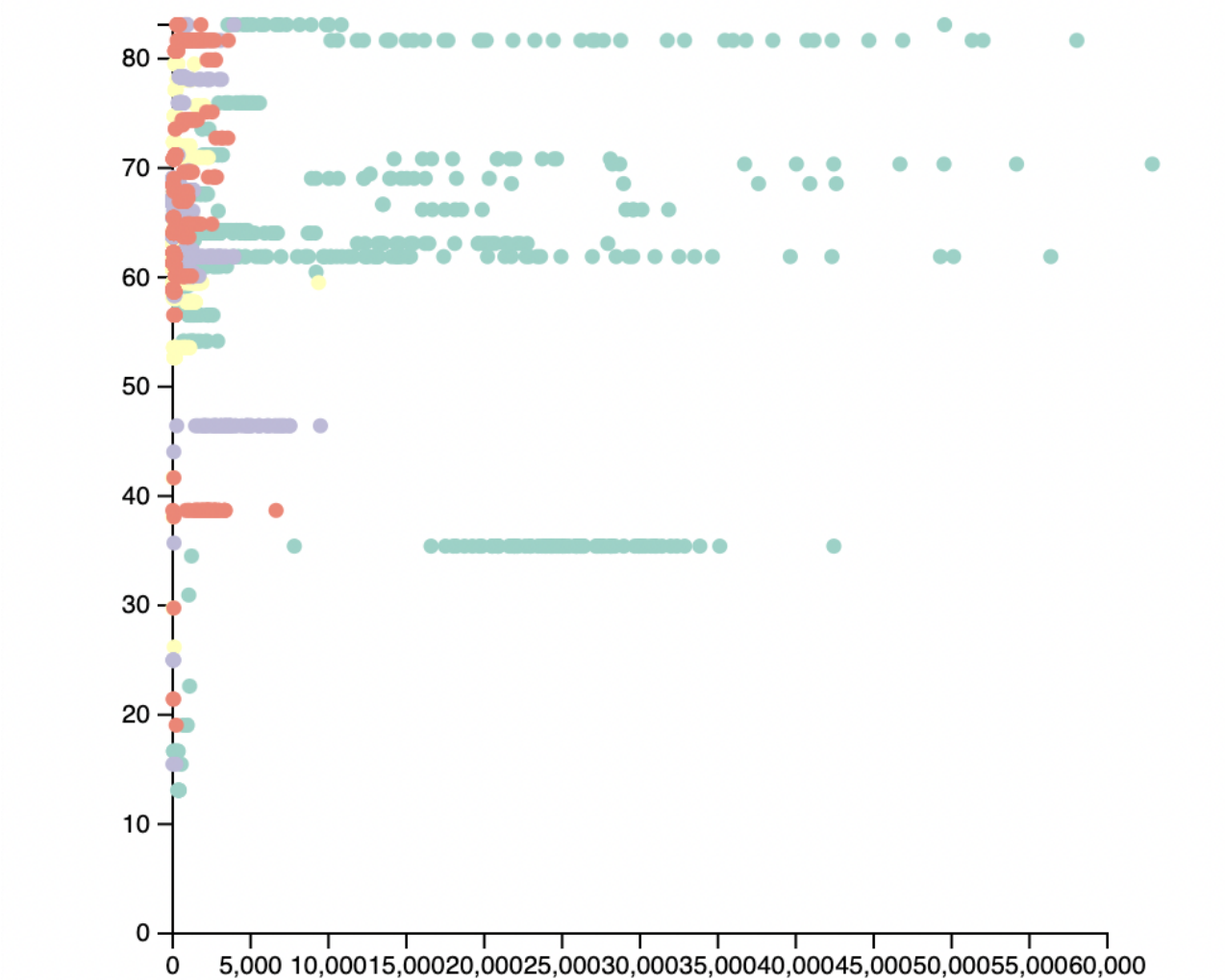


Responsiveness of Covid-19 Related Policies in UK (Government Response)



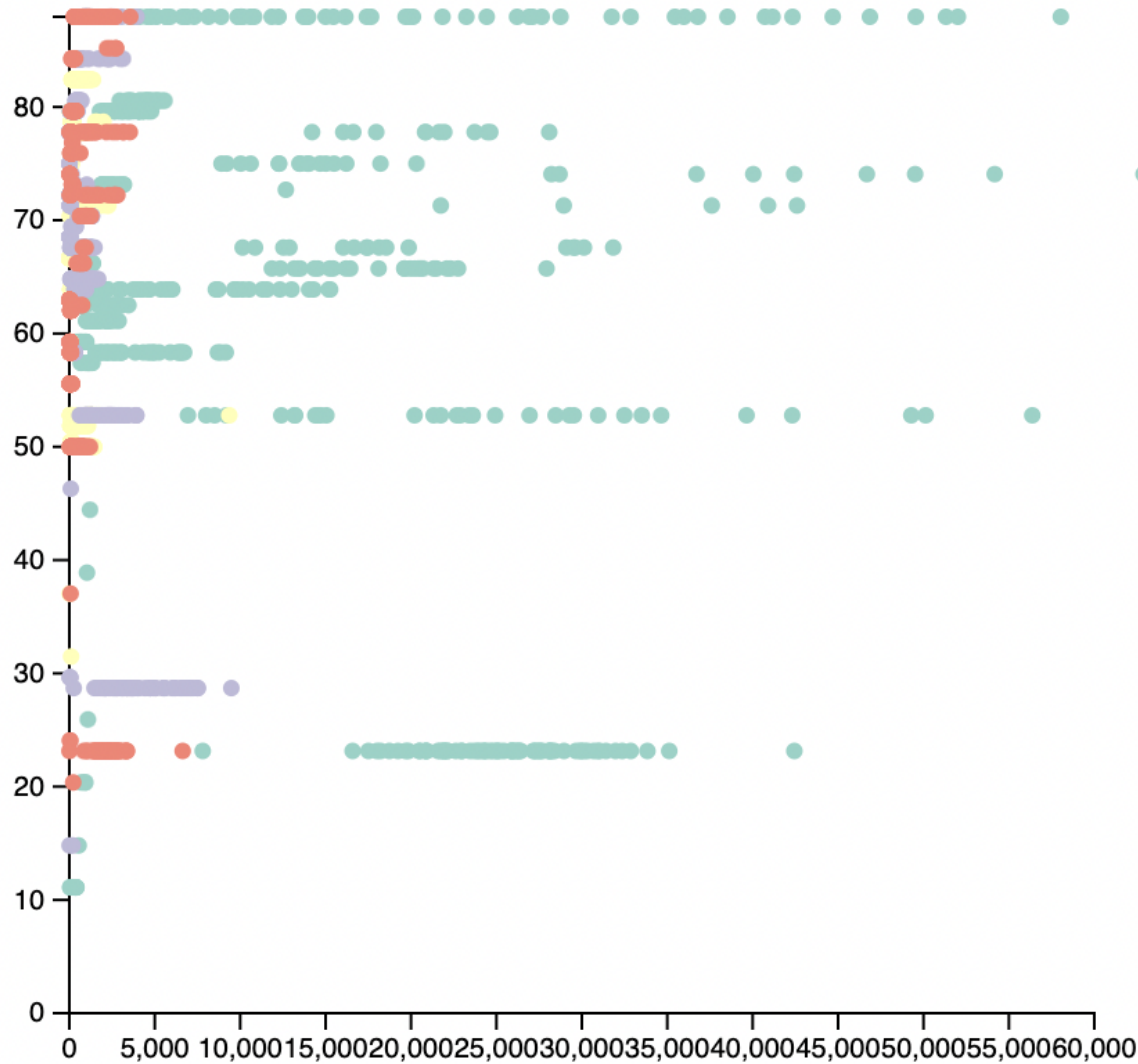
X-Axis: New Covid-19 Cases, Y-Axis: Government Response

Responsiveness of Covid-19 Related Policies in UK (Containment and Health)



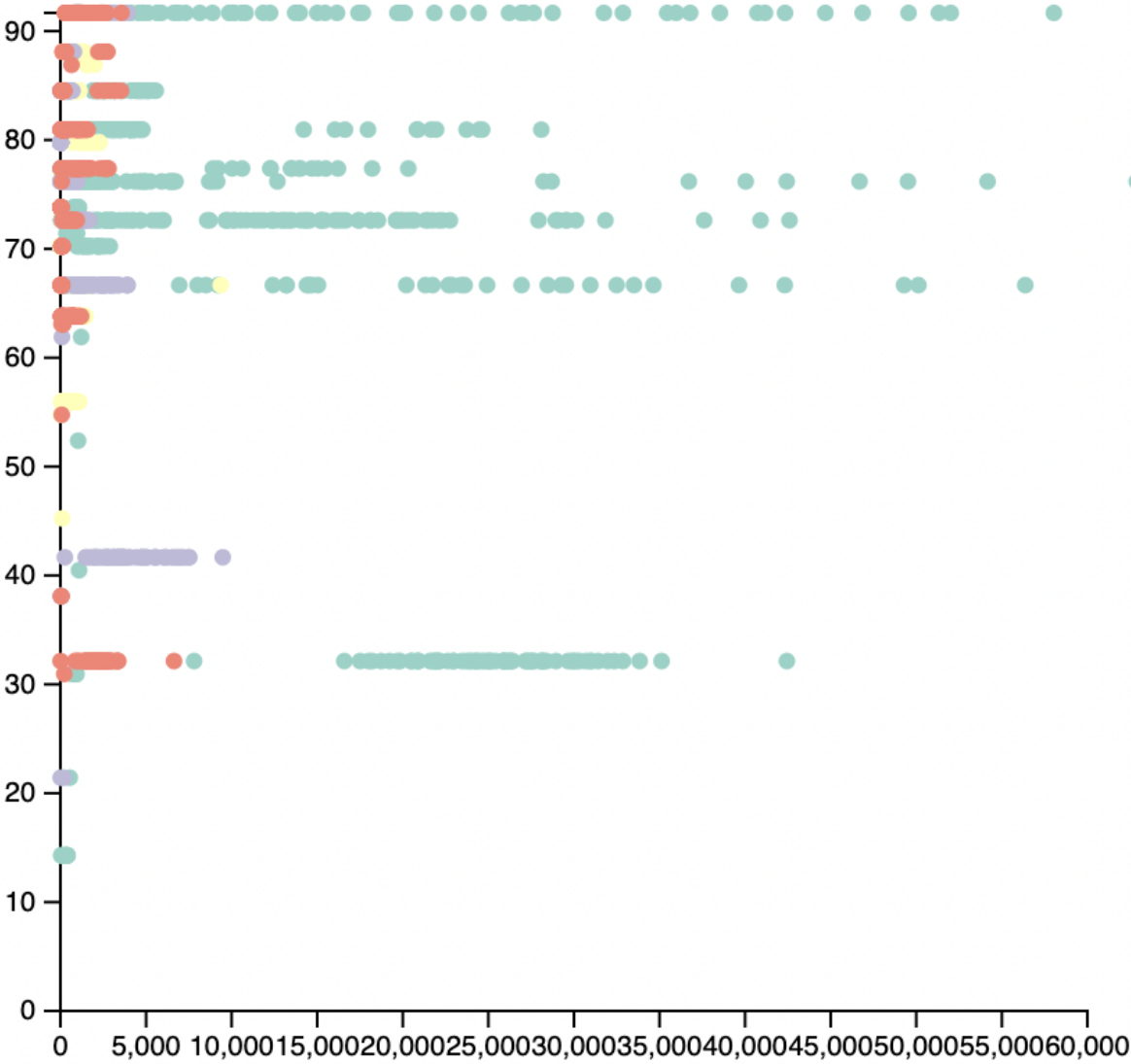
X-Axis: New Covid-19 Cases, Y-Axis: ContainmentHealth

# Responsiveness of Covid-19 Related Policies in UK (Stringency)

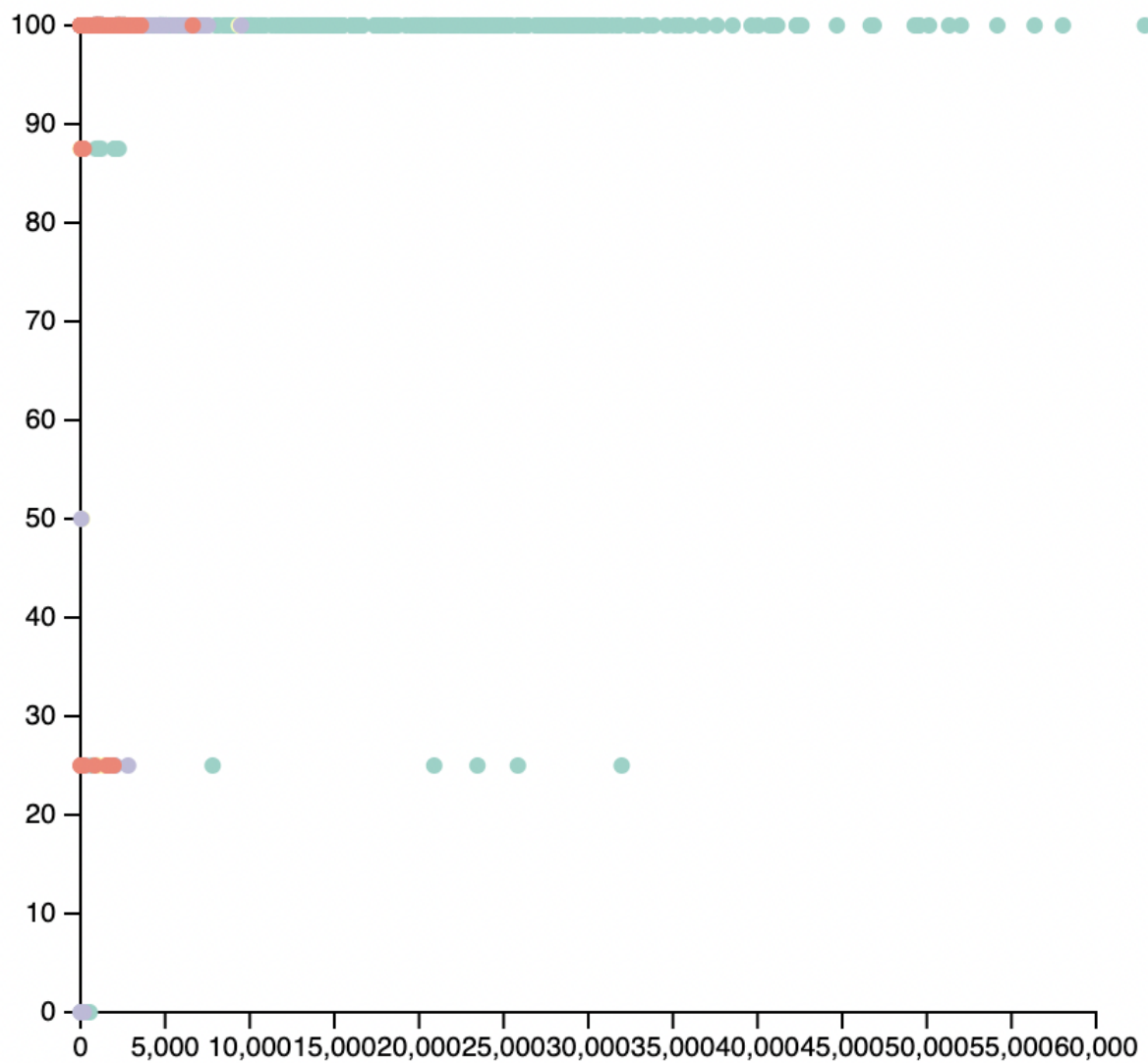


X-Axis: New Covid-19 Cases, Y-Axis: Stringency

Responsiveness of Covid-19 Related Policies in UK (Stringency Legacy)



## Responsiveness of Covid-19 Related Policies in UK (Economic Support)



X-Axis: New Covid-19 Cases, Y-Axis: Economic Support

### Analysis

We can observe that there is not a strong correlation between new cases and policy strictness. However, when the daily new cases are less than 5000, the policy might be quite strict, but not necessarily. When the daily new cases are above 50000, the policy will be stricter.

We can also observe that same daily new cases might correspond to different strictness of policy. Why? We can recall the second visualization, recently, the policies are getting relaxed even when the daily new cases is increasing, probably because people have more knowledge about Covid-19 and become more confident when facing the pandemic.

As a result, policies in UK, especially [GovernmentResponse, Containment and Health, Stringency, Stringency Response] are kind of responsive in the early stage before 2021. After 2021, policies in UK are not so responsive. Policies in UK are not very effective in the long-term perspective because it did not reduce the speed of the disease progression. However, in short-term perspective, it is effective for decreasing daily new cases.

---

✓ 2 秒    完成时间: 上午11:29

● ×