

```
# Covid-19_SIR-Model The SIR Model for Spread of Disease
```

Covid-19 Predictions for 2021

by Ali Boukrich

Usage (details: policy measures)

Here, we will analyse the relationship of policy measures and parameter values of SIR-derived ODE models. Policy measures are tracked by [Oxford Covid-19 Government Response Tracker \(OxCGRT\)](#).

Preparation

In [2]:

```
# Standard users
# !pip install covsirphy
# Developers (Note: this notebook is in example directory)
import os
os.chdir("../")
```

In [3]:

```
from pprint import pprint
```

In [4]:

```
import covsirphy as cs
cs.__version__
```

Out[4]:

```
'2.13.3-epsilon-fu4.new.420.422.429.fix.438.441'
```

In [6]:

```
# Standard users and developers
data_loader = cs.DataLoader("input")
# The number of cases (JHU style)
jhu_data = data_loader.jhu(verbose=False)
# Population in each country
population_data = data_loader.population(verbose=False)
# Government Response Tracker (OxCGRT)
oxcgrt_data = data_loader.oxcgrt(verbose=False)
```

```
# Description/citation
#print(oxcgrt_data.citation)
```

Class for policy measure analysis

We will `PolicyMeasures` class to find relationship of policy measures and parameter values of SIR-derived ODE models.

In [5]:

```
# Create analyser with tau value 360 [min] (can be changed)
analyser = cs.PolicyMeasures(jhu_data, population_data, oxcgrt_data, tau=360)
# The number of registered countries
len(analyser.countries)
```

Out[5]: 196

In [6]:

```
# We can get Scenario instance for detailed scenario analysis
type(analyser.scenario("Canada"))
```

Out[6]: `covsirphy.analysis.scenario.Scenario`

Set phases with S-R trend analysis

We will perform S-R trend analysis to set phases in all countries. Countries which do not have **16th phases** will be un-registered for the next analysis. This number will be adjusted for your analysis.

In [7]:

```
# Show the number of phases of all countries
analyser.trend()
pprint(analyser.phase_len(), compact=True)
```

```
{2: ['Timor-Leste', 'Laos', 'Solomon Islands', 'Holy See'],
 3: ['Saint Kitts and Nevis'],
 4: ['Dominica', 'Fiji', 'Tanzania', 'Brunei'],
 5: ['Saint Lucia', 'Grenada', 'Equatorial Guinea'],
 6: ['Antigua and Barbuda', 'Cambodia', 'Mauritius'],
 7: ['Saint Vincent and the Grenadines'],
 8: ['Bahamas', 'Trinidad and Tobago', 'Madagascar', 'Barbados',
      'Northern Mariana Islands', 'Kyrgyzstan', 'Seychelles'],
 9: ['Liechtenstein', 'Vietnam', 'Ghana', 'Gambia', 'Malawi',
      'Sao Tome and Principe', 'Uzbekistan', 'Philippines', 'Bolivia', 'Oman',
```

```
'Monaco', 'Australia', 'Jamaica'],
10: ['Andorra', 'Mongolia', 'Benin', 'Kazakhstan', 'Iceland', 'Djibouti',
    'Burundi', 'Virgin Islands, U.S.', 'Bermuda', 'Kuwait', 'Kenya',
    'El Salvador', 'Venezuela', 'Namibia', 'Angola', 'Singapore', 'Suriname',
    'Guatemala', 'Guam', 'Libya', 'Mozambique', 'Republic of the Congo',
    'Senegal', 'Ethiopia', 'Bahrain', 'South Africa', 'Zimbabwe', 'Zambia',
    'Eritrea', 'Papua New Guinea', 'Lesotho', 'Cape Verde', 'Maldives',
    'Botswana', 'Iraq', 'Chile', 'Guyana', 'India', 'Bhutan', 'Honduras'],
11: ['Ecuador', 'Peru', 'Algeria', 'Comoros', 'Gabon', 'Mexico', 'Pakistan',
    'Israel', 'Belize', 'New Zealand', 'Nicaragua', 'Swaziland', 'Paraguay',
    'Argentina', 'Colombia', 'Yemen', 'Syria', 'Bangladesh', 'Saudi Arabia',
    'Cote d'Ivoire', 'Brazil', 'Costa Rica', 'Taiwan', 'Dominican Republic',
    'Rwanda', 'Liberia'],
12: ['Panama', 'Egypt', 'Guinea-Bissau', 'Sudan', 'South Sudan', 'Slovenia',
    'Guinea', 'Central African Republic', 'Niger', 'Nigeria', 'Kosovo',
    'Togo', 'Puerto Rico', 'Burkina Faso', 'Latvia', 'Montenegro', 'Cameroon',
    'Georgia', 'Democratic Republic of the Congo', 'Somalia', 'Jordan',
    'Nepal', 'Sierra Leone', 'Uganda'],
13: ['San Marino', 'Mali', 'Afghanistan', 'Japan', 'Malta', 'Croatia',
    'Azerbaijan', 'Myanmar', 'Qatar', 'Serbia', 'United Arab Emirates',
    'China', 'Tajikistan', 'Palestine', 'Indonesia', 'Chad', 'Cuba',
    'Armenia', 'Morocco', 'Moldova'],
14: ['Bosnia and Herzegovina', 'Iran', 'Russia', 'Ireland', 'Slovakia',
    'Tunisia', 'Spain', 'Sri Lanka', 'Belarus', 'Hungary', 'Lithuania',
    'Haiti', 'United States'],
15: ['Sweden', 'North Macedonia', 'Belgium', 'Mauritania', 'Lebanon',
    'Ukraine', 'Luxembourg', 'Turkey', 'Bulgaria', 'Albania', 'Cyprus',
    'Greece', 'Uruguay', 'France', 'Estonia', 'Malaysia', 'Thailand'],
16: ['Poland', 'Switzerland', 'Austria', 'Canada', 'Romania', 'Netherlands',
    'Czech Republic', 'Denmark'],
17: ['Norway', 'United Kingdom', 'Italy', 'Finland'],
18: ['Germany', 'Portugal']]
```

In [8]:

```
# Set phases with S-R trend analysis
analyser.trend(min_len=16)
# The number of countries
len(analyser.countries)
```

Out[8]: 14

In [9]:

```
phase_len_dict = analyser.phase_len()
pprint(phase_len_dict, compact=True)
```

```
{16: ['Poland', 'Switzerland', 'Austria', 'Canada', 'Romania', 'Netherlands',
    'Czech Republic', 'Denmark'],
```

```
17: ['Norway', 'United Kingdom', 'Italy', 'Finland'],
18: ['Germany', 'Portugal']}
```

In [10]:

```
# Here, only show the countries which have the most phases
countries_show = max(phase_len_dict.items(), key=lambda x: x[0])[1]
# Show summary
analyser.summary(countries=countries_show)
```

Out[10]:

	Type	Start	End	Population
Country	Phase			
Germany	0th	Past	19Jan2020	23Mar2020
	1st	Past	24Mar2020	04Apr2020
	2nd	Past	05Apr2020	21Apr2020
	3rd	Past	22Apr2020	17Jun2020
	4th	Past	18Jun2020	15Aug2020
	5th	Past	16Aug2020	16Sep2020
	6th	Past	17Sep2020	08Oct2020
	7th	Past	09Oct2020	20Oct2020
	8th	Past	21Oct2020	27Oct2020
	9th	Past	28Oct2020	03Nov2020
	10th	Past	04Nov2020	10Nov2020
	11th	Past	11Nov2020	17Nov2020
	12th	Past	18Nov2020	24Nov2020
	13th	Past	25Nov2020	01Dec2020
	14th	Past	02Dec2020	08Dec2020
	15th	Past	09Dec2020	14Dec2020
	16th	Past	15Dec2020	20Dec2020
	17th	Past	21Dec2020	28Dec2020
Portugal	0th	Past	14Mar2020	03Apr2020

Country	Phase	Type	Start	End	Population
	1st	Past	04Apr2020	22Apr2020	10283822
	2nd	Past	23Apr2020	01Jun2020	10283822
	3rd	Past	02Jun2020	02Jul2020	10283822
	4th	Past	03Jul2020	11Aug2020	10283822
	5th	Past	12Aug2020	11Sep2020	10283822
	6th	Past	12Sep2020	30Sep2020	10283822
	7th	Past	01Oct2020	13Oct2020	10283822
	8th	Past	14Oct2020	21Oct2020	10283822
	9th	Past	22Oct2020	28Oct2020	10283822
	10th	Past	29Oct2020	04Nov2020	10283822
	11th	Past	05Nov2020	14Nov2020	10283822
	12th	Past	15Nov2020	25Nov2020	10283822
	13th	Past	26Nov2020	02Dec2020	10283822
	14th	Past	03Dec2020	09Dec2020	10283822
	15th	Past	10Dec2020	15Dec2020	10283822
	16th	Past	16Dec2020	21Dec2020	10283822
	17th	Past	22Dec2020	28Dec2020	10283822

Parameter estimation

we will estimate the parameter values with SIR-derived ODE models in each country.

In [11]:

```
# Parameter estimation with SIR-F model
analyser.estimate(cs.SIRF, timeout=10)
```

```
<SIR-F model: parameter estimation>
Running optimization with 4 CPUs...
```

Completed optimization. Total: 11 min 34 sec

In [12]:

```
# Show summary
analyser.summary(countries=countries_show)
```

Out[12]:

	Type	Start	End	Population	ODE	Rt	theta	kappa	rho	sigma	tau	alpha1 [-]	1/beta [day]	
Country	Phase													
Germany	0th	Past	19Jan2020	23Mar2020	82905782	SIR-F	7.94	0.131362	0.00440854	0.0405287	2.7087e-05	360	0.131	€
	1st	Past	24Mar2020	04Apr2020	82905782	SIR-F	3.97	0.0266054	0.00158046	0.0204331	0.00343009	360	0.027	12
	2nd	Past	05Apr2020	21Apr2020	82905782	SIR-F	0.55	0.0273973	0.00181889	0.0124329	0.0201689	360	0.027	20
	3rd	Past	22Apr2020	17Jun2020	82905782	SIR-F	0.51	0.0273973	0.00181889	0.0135866	0.023853	360	0.027	18
	4th	Past	18Jun2020	15Aug2020	82905782	SIR-F	1.23	0.0273973	0.00181889	0.0209579	0.0147464	360	0.027	11
	5th	Past	16Aug2020	16Sep2020	82905782	SIR-F	1.27	0.00323185	0.000309142	0.0181085	0.0139587	360	0.003	13
	6th	Past	17Sep2020	08Oct2020	82905782	SIR-F	1.45	0.00323185	0.000309142	0.0208669	0.0140351	360	0.003	11
	7th	Past	09Oct2020	20Oct2020	82905782	SIR-F	2.12	0.0266054	0.00158046	0.0203071	0.00775686	360	0.027	12
	8th	Past	21Oct2020	27Oct2020	82905782	-	-	-	-	-	-	-	-	
	9th	Past	28Oct2020	03Nov2020	82905782	-	-	-	-	-	-	-	-	
	10th	Past	04Nov2020	10Nov2020	82905782	-	-	-	-	-	-	-	-	
	11th	Past	11Nov2020	17Nov2020	82905782	-	-	-	-	-	-	-	-	
	12th	Past	18Nov2020	24Nov2020	82905782	-	-	-	-	-	-	-	-	
	13th	Past	25Nov2020	01Dec2020	82905782	SIR-F	0.82	0.0273973	0.00181889	0.0165861	0.017755	360	0.027	15

	Type	Start	End	Population	ODE	Rt	theta	kappa	rho	sigma	tau	alpha1 [-]	1/beta [day]	
Country	Phase													
Portugal	14th	Past	02Dec2020	08Dec2020	82905782	SIR-F	0.91	0.0273973	0.00181889	0.0188878	0.0183849	360	0.027	13
	15th	Past	09Dec2020	14Dec2020	82905782	SIR-F	1.2	0.00323185	0.000309142	0.0195129	0.0159425	360	0.003	12
	16th	Past	15Dec2020	20Dec2020	82905782	-	-	-	-	-	-	-	-	
	17th	Past	21Dec2020	28Dec2020	82905782	-	-	-	-	-	-	-	-	
	0th	Past	14Mar2020	03Apr2020	10283822	SIR-F	99.46	0.0404589	0.000183921	0.0423652	0.000224807	360	0.04	5
	1st	Past	04Apr2020	22Apr2020	10283822	SIR-F	6.82	0.137787	0.000450842	0.0100539	0.000819638	360	0.138	24
	2nd	Past	23Apr2020	01Jun2020	10283822	SIR-F	1.18	0.157772	0.001978	0.00387129	0.000776423	360	0.158	62
	3rd	Past	02Jun2020	02Jul2020	10283822	SIR-F	0.92	0.126088	0.00155109	0.00720328	0.00532846	360	0.126	32
	4th	Past	03Jul2020	11Aug2020	10283822	SIR-F	0.73	0.216201	0.000768367	0.00546097	0.00508911	360	0.216	45
	5th	Past	12Aug2020	11Sep2020	10283822	SIR-F	1.15	0.157772	0.001978	0.00596871	0.0023952	360	0.158	41
	6th	Past	12Sep2020	30Sep2020	10283822	SIR-F	1.8	0.0273973	0.00181889	0.0093469	0.00322609	360	0.027	26
	7th	Past	01Oct2020	13Oct2020	10283822	SIR-F	1.82	0.137787	0.000450842	0.00939535	0.00400316	360	0.138	26
	8th	Past	14Oct2020	21Oct2020	10283822	SIR-F	1.39	0.0273973	0.00181889	0.0148859	0.00860955	360	0.027	16
	9th	Past	22Oct2020	28Oct2020	10283822	SIR-F	1.64	0.0273973	0.00181889	0.0180317	0.00889327	360	0.027	13
	10th	Past	29Oct2020	04Nov2020	10283822	-	-	-	-	-	-	-	-	
	11th	Past	05Nov2020	14Nov2020	10283822	SIR-F	1.58	0.00323185	0.000309142	0.0197602	0.012141	360	0.003	12

	Type	Start	End	Population	ODE	Rt	theta	kappa	rho	sigma	tau	alpha1 [-]	1/beta [day]
Country	Phase												
	12th	Past	15Nov2020	25Nov2020	10283822	SIR-F	0.92	0.0273973	0.00181889	0.0200534	0.0193053	360	0.027
	13th	Past	26Nov2020	02Dec2020	10283822	-	-	-	-	-	-	-	-
	14th	Past	03Dec2020	09Dec2020	10283822	-	-	-	-	-	-	-	-
	15th	Past	10Dec2020	15Dec2020	10283822	-	-	-	-	-	-	-	-
	16th	Past	16Dec2020	21Dec2020	10283822	-	-	-	-	-	-	-	-
	17th	Past	22Dec2020	28Dec2020	10283822	-	-	-	-	-	-	-	-



Compare the history of parameters

Here, compare the history of parameter values if countries.

In [13]:

```
# All parameter/day parameter/Rt/OxCGRT
analyser.track().head()
```

Out[13]:

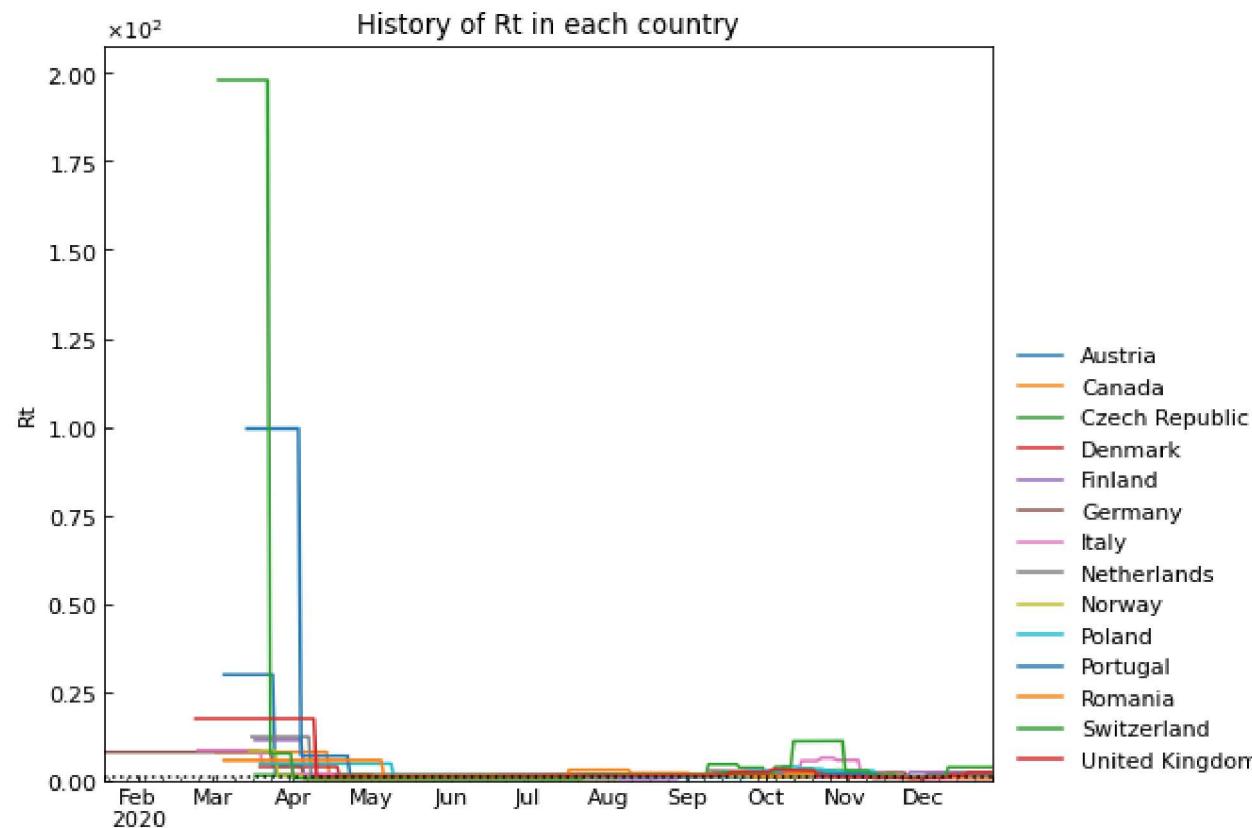
	Date	Country	theta	kappa	rho	sigma	alpha1 [-]	1/alpha2 [day]	1/beta [day]	1/gamma [day]	...	Cancel_events	Gatherings_restrictions
0	2020-03-19	Poland	0.160605	0.00181376	0.0221956	0.00214247	0.161	137	11	116	...	2	0
1	2020-03-19	Poland	0.160605	0.00181376	0.0221956	0.00214247	0.161	137	11	116	...	0	0
2	2020-03-19	Poland	0.160605	0.00181376	0.0221956	0.00214247	0.161	137	11	116	...	0	0
3	2020-03-19	Poland	0.160605	0.00181376	0.0221956	0.00214247	0.161	137	11	116	...	0	0
4	2020-03-19	Poland	0.160605	0.00181376	0.0221956	0.00214247	0.161	137	11	116	...	2	0

5 rows × 23 columns

Rt: Reprodcution number [-]

In [14]:

```
analyser.history("Rt", roll_window=None).tail()
```



Out[14]:

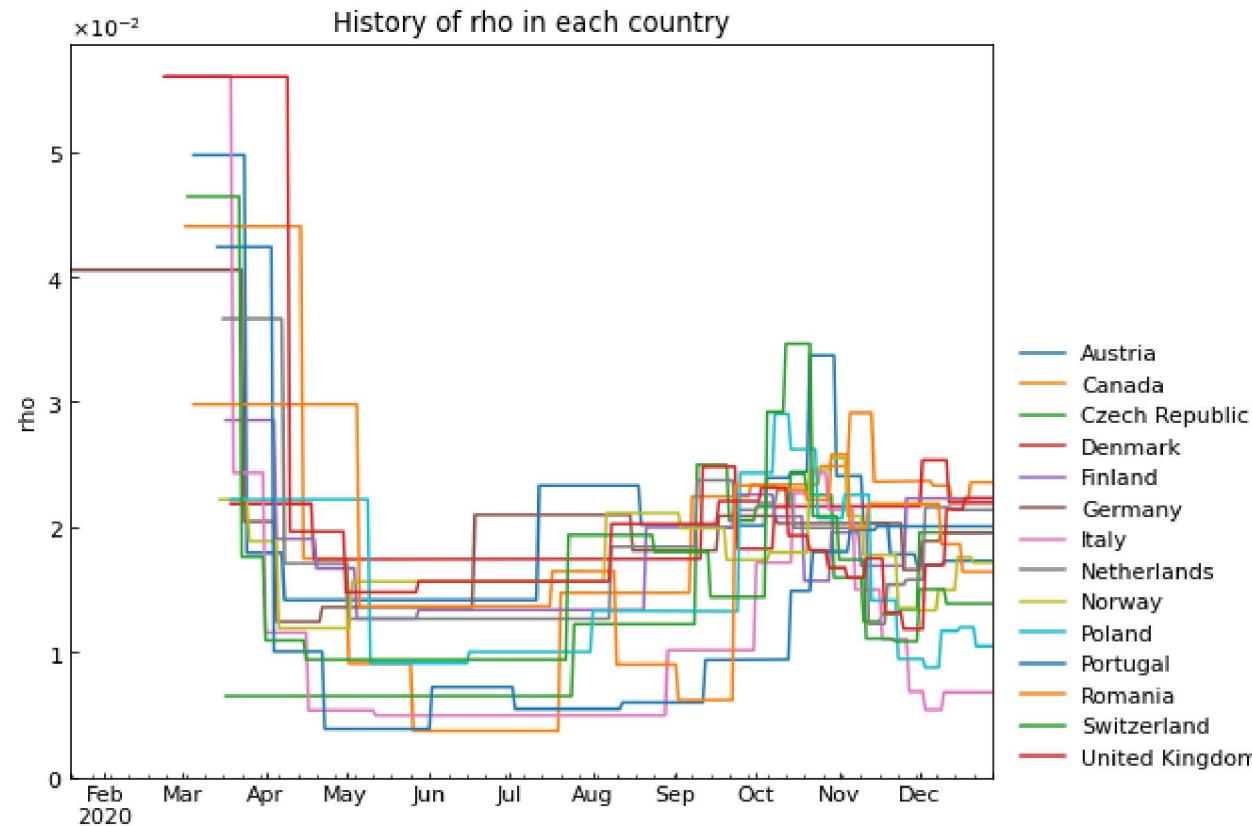
Country	Austria	Canada	Czech Republic	Denmark	Finland	Germany	Italy	Netherlands	Norway	Poland	Portugal	Romania	Switzerland	Uni Kingd	
Date	2020-12-24	0.74	1.01	0.8	1.75	2.18	1.2	0.57	2.0	0.98	0.87	0.92	0.64	3.66	2

Country	Austria	Canada	Czech Republic	Denmark	Finland	Germany	Italy	Netherlands	Norway	Poland	Portugal	Romania	Switzerland	Uni Kingd
Date														
2020-12-25	0.74	1.01	0.8	1.75	2.18	1.2	0.57	2.0	0.98	0.87	0.92	0.64	3.66	2
2020-12-26	0.74	1.01	0.8	1.75	2.18	1.2	0.57	2.0	0.98	0.87	0.92	0.64	3.66	2
2020-12-27	0.74	1.01	0.8	1.75	2.18	1.2	0.57	2.0	0.98	0.87	0.92	0.64	3.66	2
2020-12-28	0.74	1.01	0.8	1.75	2.18	1.2	0.57	2.0	0.98	0.87	0.92	0.64	3.66	2



In []: *### ρ : Effective contact rate $\mathrm{[-]}$*

In [15]: `analyser.history("rho", roll_window=None).tail()`



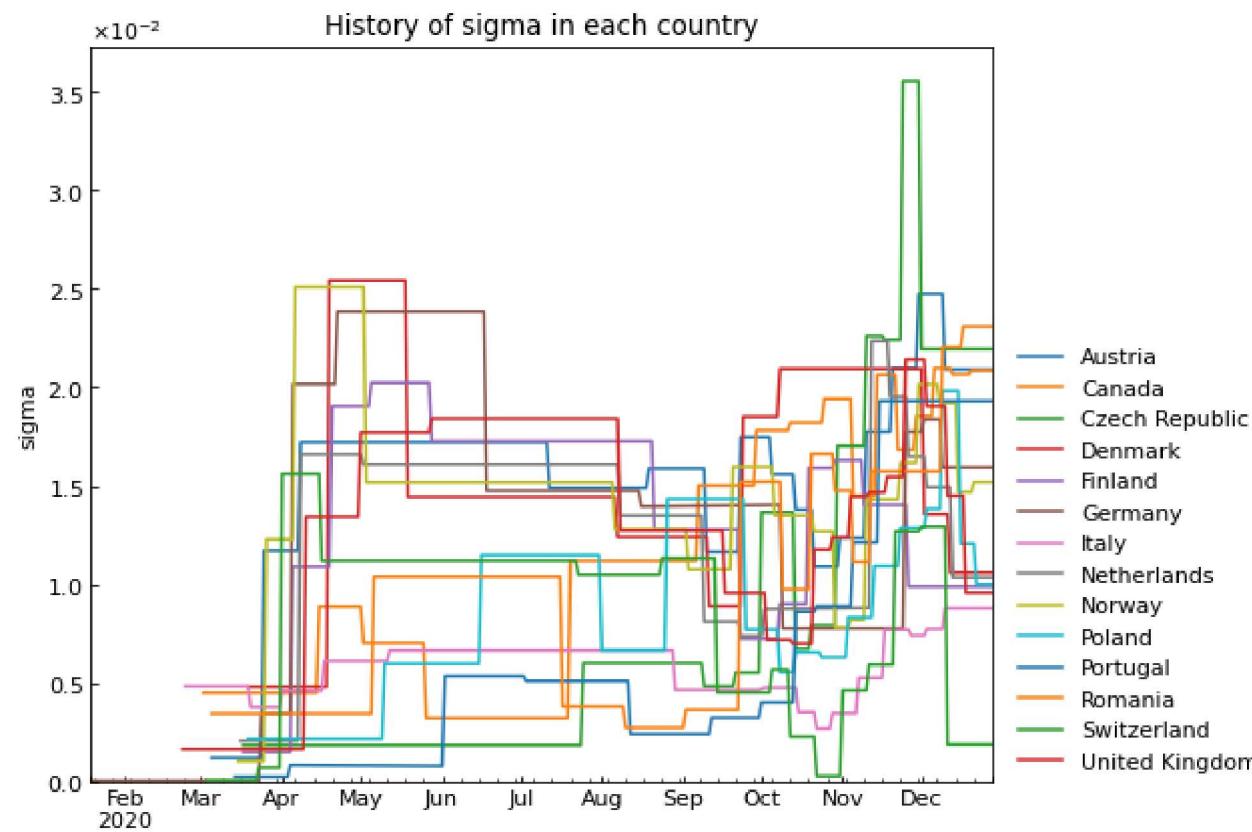
Out[15]:

Country	Austria	Canada	Czech Republic	Denmark	Finland	Germany	Italy	Netherlands	Norway	Poland	Portugal	Romania	Switzerland	
Date														
2020-12-24	0.017295	0.02356	0.019554	0.021919	0.022279	0.019513	0.006786		0.02137	0.017128	0.010471	0.020053	0.016433	0.013882
2020-12-25	0.017295	0.02356	0.019554	0.021919	0.022279	0.019513	0.006786		0.02137	0.017128	0.010471	0.020053	0.016433	0.013882
2020-12-26	0.017295	0.02356	0.019554	0.021919	0.022279	0.019513	0.006786		0.02137	0.017128	0.010471	0.020053	0.016433	0.013882
2020-12-27	0.017295	0.02356	0.019554	0.021919	0.022279	0.019513	0.006786		0.02137	0.017128	0.010471	0.020053	0.016433	0.013882
2020-12-28	0.017295	0.02356	0.019554	0.021919	0.022279	0.019513	0.006786		0.02137	0.017128	0.010471	0.020053	0.016433	0.013882

σ : Recovery rate [—]

In [16]:

```
analyser.history("sigma", roll_window=None).tail()
```



Out[16]:

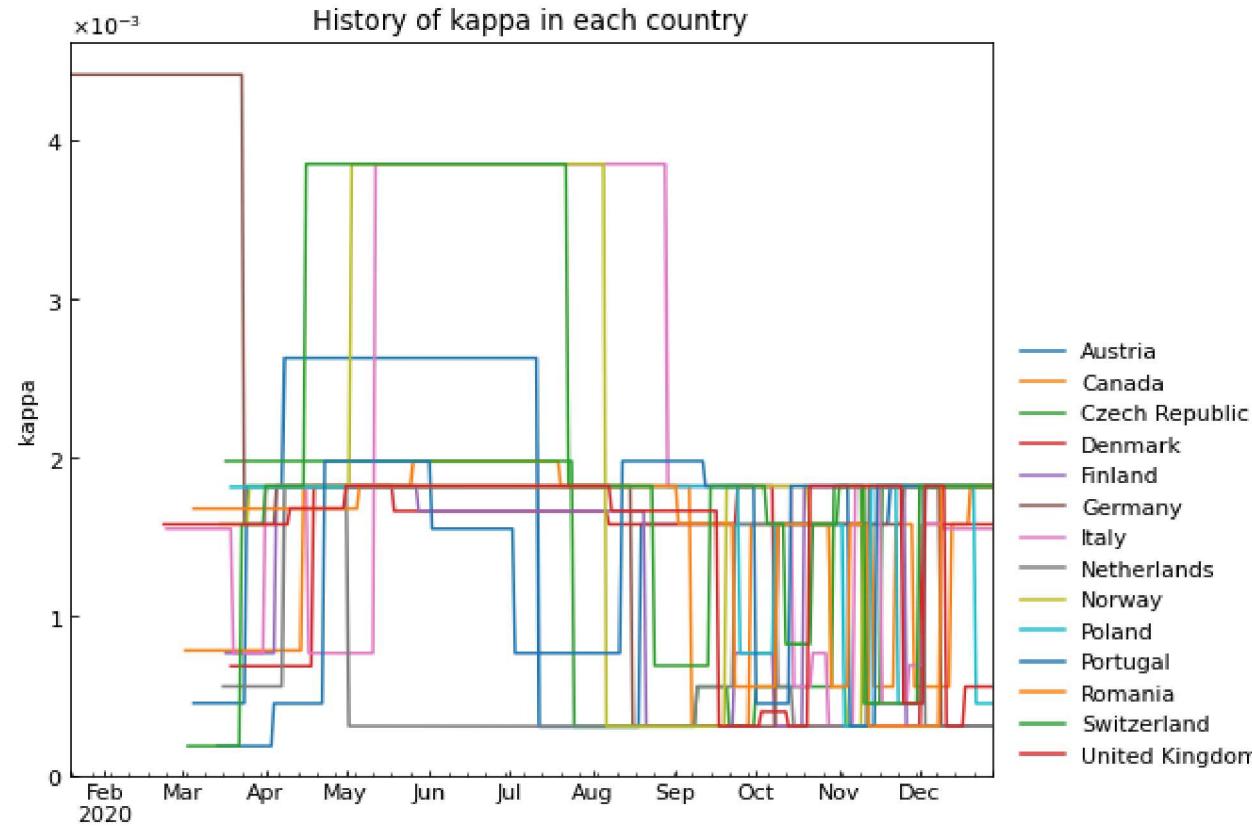
Country	Austria	Canada	Czech Republic	Denmark	Finland	Germany	Italy	Netherlands	Norway	Poland	Portugal	Romania	Switzerland
Date													
2020-12-24	0.020912	0.02087	0.021945	0.010594	0.009875	0.015942	0.008792	0.010353	0.015194	0.009986	0.019305	0.0231	0.001866
2020-12-25	0.020912	0.02087	0.021945	0.010594	0.009875	0.015942	0.008792	0.010353	0.015194	0.009986	0.019305	0.0231	0.001866

Country	Austria	Canada	Czech Republic	Denmark	Finland	Germany	Italy	Netherlands	Norway	Poland	Portugal	Romania	Switzerland
Date													
2020-12-26	0.020912	0.02087	0.021945	0.010594	0.009875	0.015942	0.008792	0.010353	0.015194	0.009986	0.019305	0.0231	0.001866
2020-12-27	0.020912	0.02087	0.021945	0.010594	0.009875	0.015942	0.008792	0.010353	0.015194	0.009986	0.019305	0.0231	0.001866
2020-12-28	0.020912	0.02087	0.021945	0.010594	0.009875	0.015942	0.008792	0.010353	0.015194	0.009986	0.019305	0.0231	0.001866

κ : Mortality rate of infected cases [−]

In [17]:

```
analyser.history("kappa", roll_window=None).tail()
```



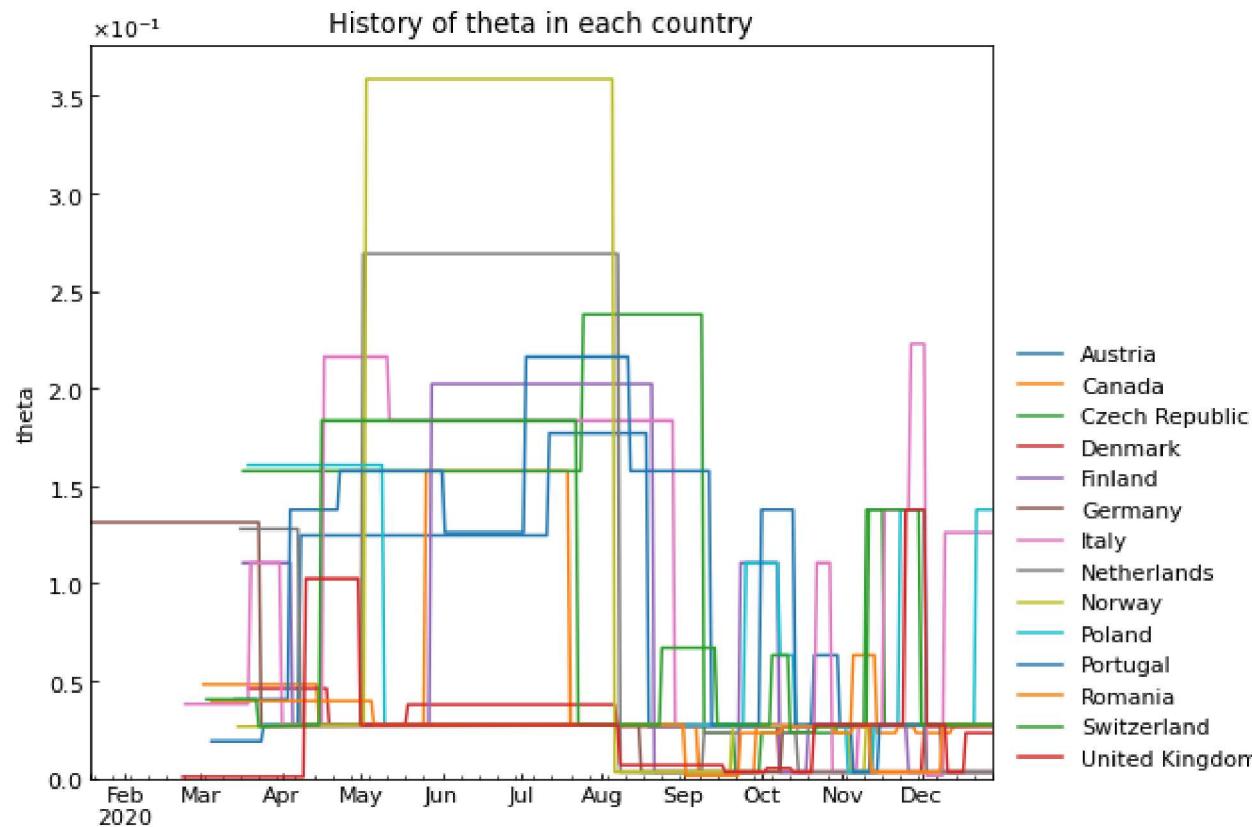
Out[17]:

Country	Austria	Canada	Czech Republic	Denmark	Finland	Germany	Italy	Netherlands	Norway	Poland	Portugal	Romania	Switzerland
Date													
2020-12-24	0.001819	0.001819	0.001819	0.00158	0.000309	0.000309	0.001551	0.000309	0.001819	0.000451	0.001819	0.001819	0.00181
2020-12-25	0.001819	0.001819	0.001819	0.00158	0.000309	0.000309	0.001551	0.000309	0.001819	0.000451	0.001819	0.001819	0.00181
2020-12-26	0.001819	0.001819	0.001819	0.00158	0.000309	0.000309	0.001551	0.000309	0.001819	0.000451	0.001819	0.001819	0.00181
2020-12-27	0.001819	0.001819	0.001819	0.00158	0.000309	0.000309	0.001551	0.000309	0.001819	0.000451	0.001819	0.001819	0.00181
2020-12-28	0.001819	0.001819	0.001819	0.00158	0.000309	0.000309	0.001551	0.000309	0.001819	0.000451	0.001819	0.001819	0.00181

θ : Mortality rate of S* cases [−]

In [18]:

```
analyser.history("theta", roll_window=None).tail()
```



Out[18]:

Country	Austria	Canada	Czech Republic	Denmark	Finland	Germany	Italy	Netherlands	Norway	Poland	Portugal	Romania	Switzerland	United Kingdom
Date														
2020-12-24	0.027397	0.027397	0.027397	0.026605	0.003232	0.003232	0.126088	0.003232	0.027397	0.137787	0.027397	0.027397	0.02739	
2020-12-25	0.027397	0.027397	0.027397	0.026605	0.003232	0.003232	0.126088	0.003232	0.027397	0.137787	0.027397	0.027397	0.02739	

Country	Austria	Canada	Czech Republic	Denmark	Finland	Germany	Italy	Netherlands	Norway	Poland	Portugal	Romania	Switzerland
Date													
2020-12-26	0.027397	0.027397	0.027397	0.026605	0.003232	0.003232	0.126088	0.003232	0.027397	0.137787	0.027397	0.027397	0.02739
2020-12-27	0.027397	0.027397	0.027397	0.026605	0.003232	0.003232	0.126088	0.003232	0.027397	0.137787	0.027397	0.027397	0.02739
2020-12-28	0.027397	0.027397	0.027397	0.026605	0.003232	0.003232	0.126088	0.003232	0.027397	0.137787	0.027397	0.027397	0.02739

(Experimental): Relationship of OxCGRT index and parameter values

Government responses will impact on parameter values of ODE models about 14 days later. Here, we will investigate on the relationship of [stringency index \(from OxCGRT\)](#) and reproduction number/model parameters.

In [19]:

```
%matplotlib inline
from matplotlib import pyplot as plt
import seaborn as sns
import pandas as pd
```

In [20]:

```
df = analyser.track()
param_cols = ["Rt", *cs.SIRF.PARAMETERS]
oxcgrt_cols = oxcgrt_data.OXCGRT_VARS[:]
cols = param_cols + oxcgrt_cols
delay = 14

for col in param_cols:
    df[col] = pd.to_numeric(df[col], errors="coerce")
    df[col] = df.groupby("Country")[col].shift(0 - delay)

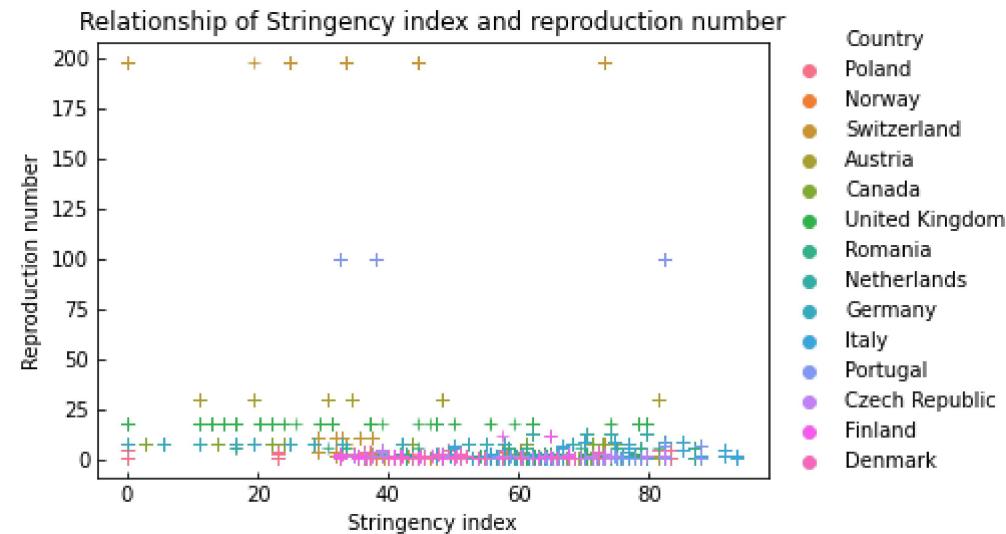
df = df.dropna(how="any")
df = df.loc[:, ["Date", "Country", *cols]]
track_df = df.copy()
track_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 52766 entries, 0 to 52947
Data columns (total 19 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Date              52766 non-null   datetime64[ns]
 1   Country           52766 non-null   object  
 2   Rt                52766 non-null   float64 
 3   theta              52766 non-null   float64 
 4   kappa              52766 non-null   float64 
 5   rho                52766 non-null   float64 
 6   sigma              52766 non-null   float64 
 7   School_closing    52766 non-null   int64  
 8   Workplace_closing 52766 non-null   int64  
 9   Cancel_events      52766 non-null   int64  
 10  Gatherings_restrictions 52766 non-null   int64 
 11  Transport_closing 52766 non-null   int64  
 12  Stay_home_restrictions 52766 non-null   int64 
 13  Internal_movement_restrictions 52766 non-null   int64 
 14  International_movement_restrictions 52766 non-null   int64 
 15  Information_campaigns 52766 non-null   int64  
 16  Testing_policy     52766 non-null   int64  
 17  Contact_tracing   52766 non-null   int64  
 18  Stringency_index   52766 non-null   float64 
dtypes: datetime64[ns](1), float64(6), int64(11), object(1)
memory usage: 8.1+ MB
```

In [21]:

```
# Correlation
df = track_df.copy()
df.corr().style.background_gradient(axis=None)
```

Out[21]: Rt theta kappa rho sigma School_closing Workplace_closing Cancel_events Gatherings_restrictions Transport_closing Stay_home_restrict



In []:

In []: