Public Opinion of COVID Policies

Using Twitter Sentiment and Policy Data

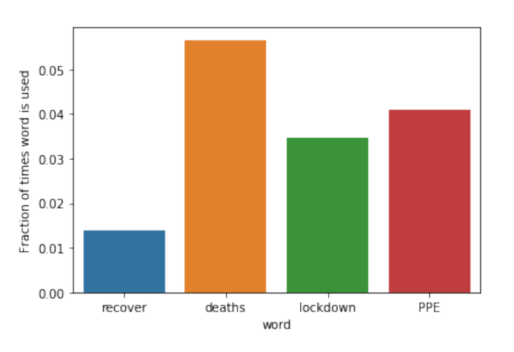
Elena Stein

Covid Tweets EDA

The tasks for this EDA are to:

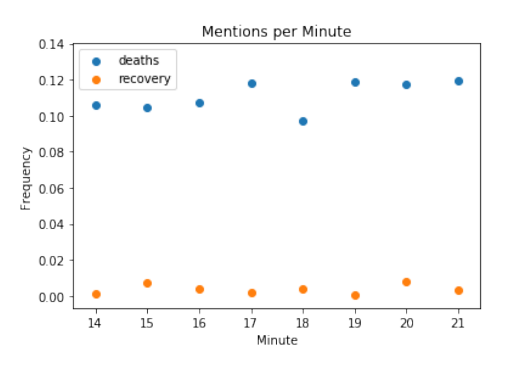
1. Read in the tweets JSON and process
2. Investigate word use in the text
3. Look and word use and sentiment over time
4. Investigate which fields to use for Geographical analysis later on

In [58]:
```
fig = sns.barplot(words,words_use_frac)
plt.ylabel('Fraction of times word is used')
plt.xlabel('word')
plt.savefig('Word_usage.png')
```



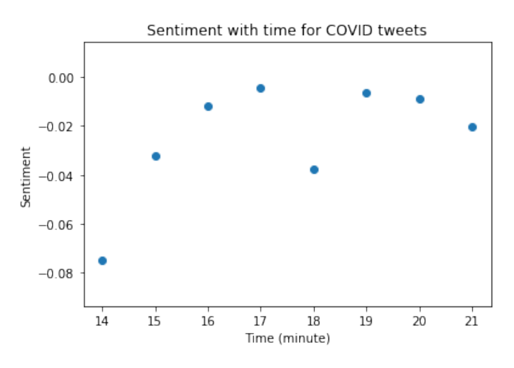
Investigating the use of the word "recovery" versus "death" over time

In [8]:
```
plt.scatter(mean_death.index.minute, mean_death)
plt.scatter(mean_recover.index.minute, mean_recover)
plt.xlabel('Minute')
plt.ylabel('Frequency')
plt.title('Mentions per Minute')
plt.legend(('deaths', 'recovery'))
plt.show()
```



Looking at sentiment over time for the 6th of May

In [10]:
```
sentiment_covid = sentiment.resample('1 min').mean()
plt.scatter(sentiment_covid.index.minute, sentiment_covid, )
plt.xlabel('Time (minute)')
plt.ylabel('Sentiment')
plt.title('Sentiment with time for COVID tweets')
plt.show()
```



Time Series Analysis of Policy Data

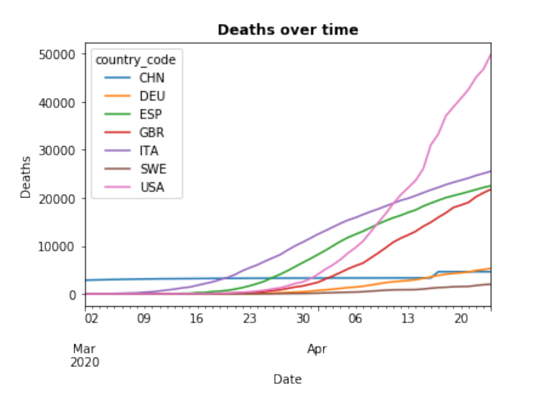
This analysis investigates the evolution of Stringency Index in different countries over time, with the death rate. For information of how it is calculated see the link https://covidtracker.bsg.ox.ac.uk, we use the API data instead of the csv files.

Tasks:

1. Fill the df with json data
2. Plot deaths and the stringency index over time for March and April
3. Plot selected countries to see the affect of death toll on stringency

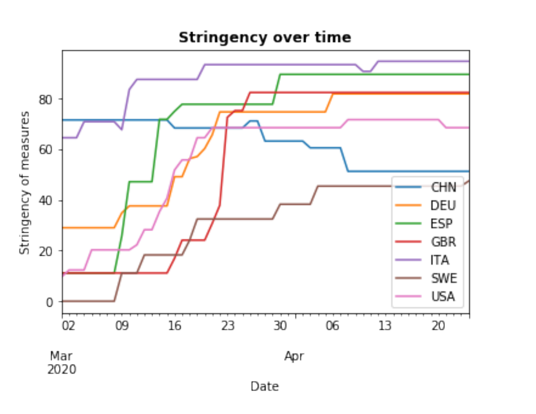
In [15]:
```
import matplotlib.pyplot as plt
df_pivot = df_selected.pivot(index='date_value', columns='country_code', values=
['stringency','deaths'])
df_pivot.plot(y='deaths')
plt.title('Deaths over time', fontweight='semibold')
plt.xlabel('Date')
plt.ylabel('Deaths')
plt.xlim('2020-03-01','2020-04-24')
```

Out[15]:
```
(18322, 18376)
```



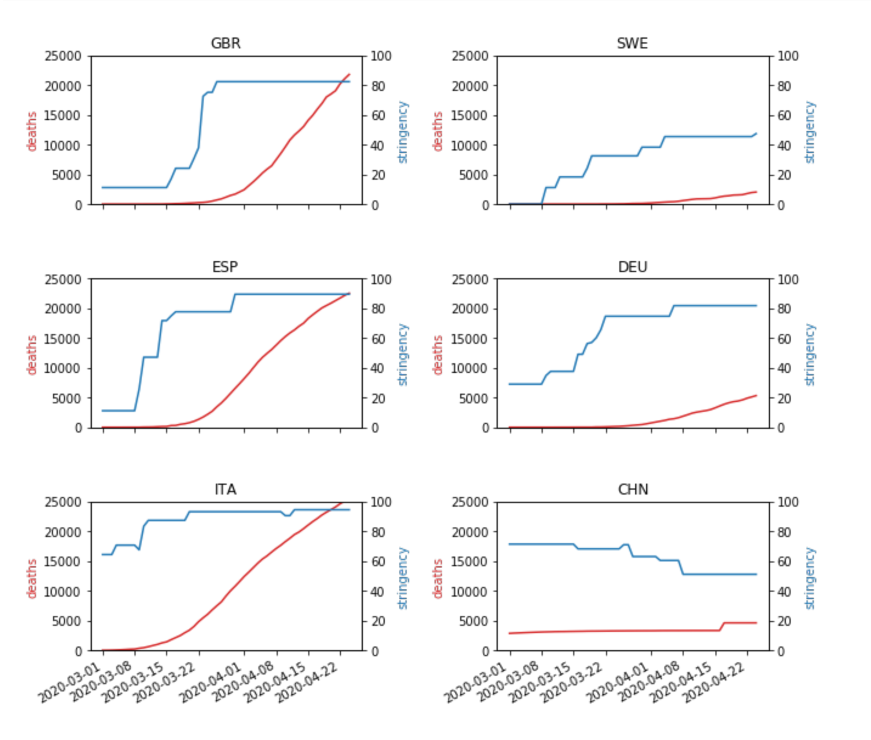
In [16]:
```
df_pivot.plot(y='stringency')
plt.xlabel('Date')
plt.ylabel('Stringency of measures')
plt.xlim('2020-03-01','2020-04-24')
plt.legend(loc=4)
plt.title('Stringency over time', fontweight='bold')
```

Out[16]:
```
Text(0.5, 1.0, 'Stringency over time')
```



In [45]:
```
fig
```

Out[45]:



Geographical Analysis of Policy Data

Tasks:

1. Investigate the completeness of the data set, to identify how much missing data there is
2. Convert the date to datetime values
3. Convert the data to a geopanda to plot on a map
4. Plot a Chloropleth of the Stringency Index by country for a particular day

We plot the Stringency Index of each country on a map, shaded according to the level of Stringency

In [49]:
```
m
```

Out[49]:

Geographical Analysis of Tweets

Tasks:

1. Import may 5th tweets
2. Carry out sentiment analysis of the tweets
3. Geocode the tweets based on user location
4. Plot them on a map based on Sentiment

Geocode the tweets based on user location description

In [47]:
```
m_2
```

Out[47]: