

Aula1_2_VisualizaçãodeSériesTemporais

July 14, 2020

1 Visualização de Séries Temporais

por Cibele Russo

ICMC/USP - São Carlos SP

```
[1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

%matplotlib inline
```

```
[2]: covidSP = pd.read_csv('covidSP.csv', index_col='date', parse_dates=True)
```

```
[3]: covidSP.head()
```

```
[3]:
```

	confirmed	deaths
date		
2020-02-26	0	0
2020-02-27	0	0
2020-02-28	1	0
2020-02-29	0	0
2020-03-01	0	0

1.1 Lidando com datas

```
[4]: covidSP.index
```

```
[4]: DatetimeIndex(['2020-02-26', '2020-02-27', '2020-02-28', '2020-02-29',
                  '2020-03-01', '2020-03-02', '2020-03-03', '2020-03-04',
                  '2020-03-05', '2020-03-06',
                  ...,
                  '2020-07-02', '2020-07-03', '2020-07-04', '2020-07-05',
                  '2020-07-06', '2020-07-07', '2020-07-08', '2020-07-09',
                  '2020-07-10', '2020-07-11'],
                  dtype='datetime64[ns]', name='date', length=135, freq=None)
```

Caso o índice não seja do tipo DatetimeIndex, algumas alternativas podem ser utilizadas:

```
[5]: idx = pd.date_range('25/2/2020', periods=7, freq='D')
idx
```

```
[5]: DatetimeIndex(['2020-02-25', '2020-02-26', '2020-02-27', '2020-02-28',
                    '2020-02-29', '2020-03-01', '2020-03-02'],
                    dtype='datetime64[ns]', freq='D')
```

```
[6]: idx = pd.date_range('25/02/2020', periods=7, freq='M')
idx
```

```
[6]: DatetimeIndex(['2020-02-29', '2020-03-31', '2020-04-30', '2020-05-31',
                    '2020-06-30', '2020-07-31', '2020-08-31'],
                    dtype='datetime64[ns]', freq='M')
```

```
[7]: idx = pd.to_datetime(['Feb 02, 2020', '2/25/2020', '25-Feb-2020', None])
idx
```

```
[7]: DatetimeIndex(['2020-02-02', '2020-02-25', '2020-02-25', 'NaT'],
                    dtype='datetime64[ns]', freq=None)
```

```
[8]: np.array(['2020-02-25', '2020-02-26', '2020-02-27'], dtype='datetime64[h]')
```

```
[8]: array(['2020-02-25T00', '2020-02-26T00', '2020-02-27T00'],
          dtype='datetime64[h]')
```

```
[9]: datas = np.array(['2020-02-25', '2020-02-26', '2020-02-27'],
                      dtype='datetime64[D]')
```

```
[10]: idx = pd.DatetimeIndex(datas)
idx
```

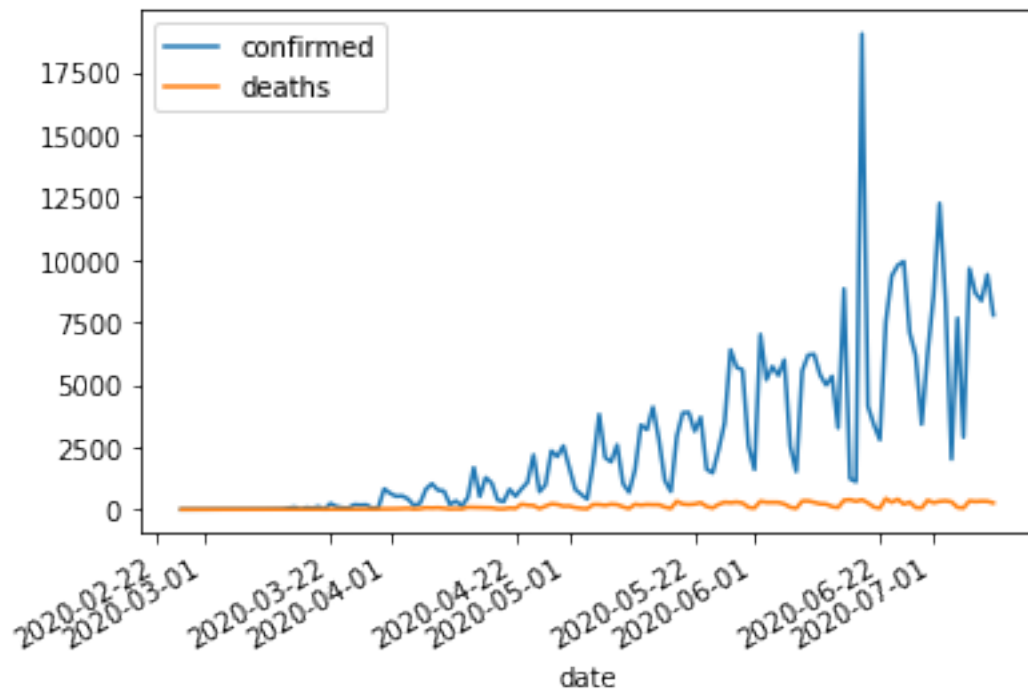
```
[10]: DatetimeIndex(['2020-02-25', '2020-02-26', '2020-02-27'],
                    dtype='datetime64[ns]', freq=None)
```

Existem outras possibilidades!

1.2 Visualização de dados de séries temporais:

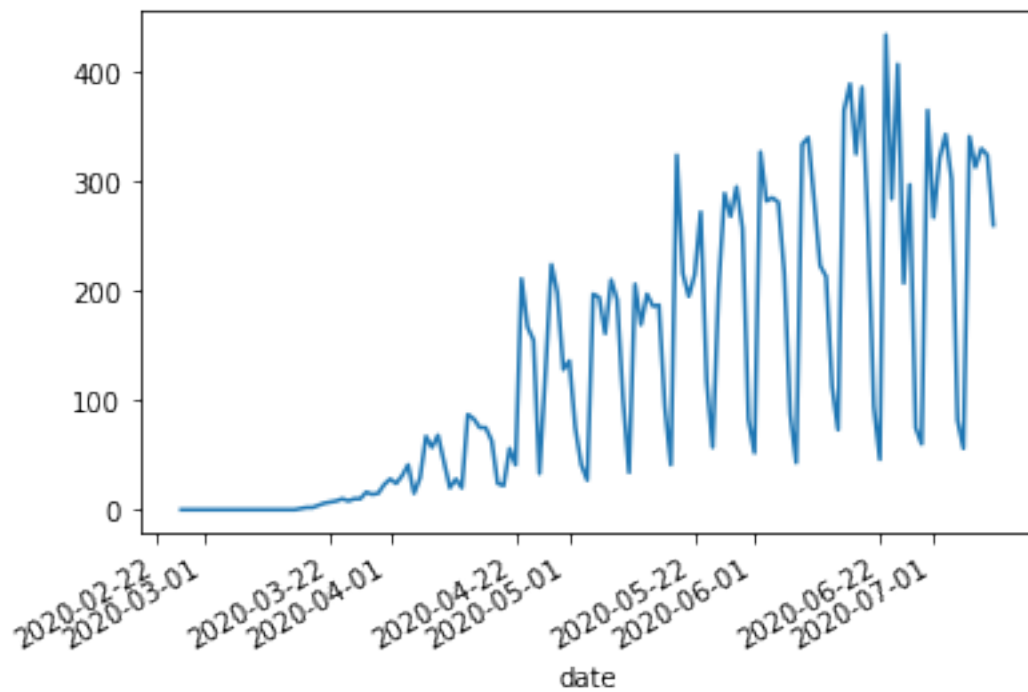
```
[11]: covidSP.plot()
```

```
[11]: <matplotlib.axes._subplots.AxesSubplot at 0x7f1cf2ed03d0>
```



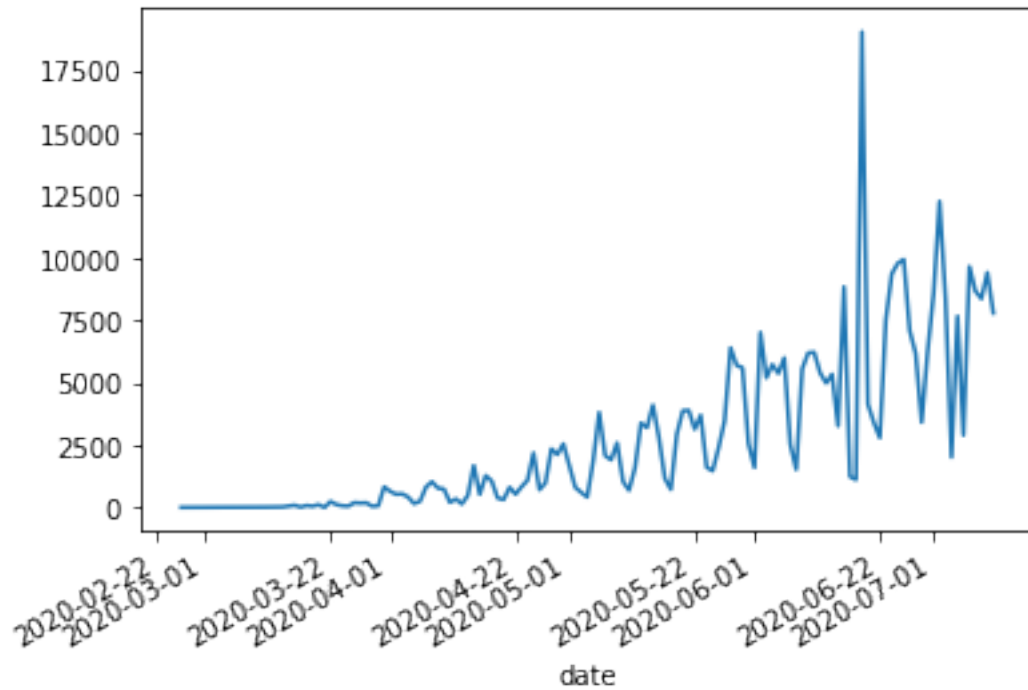
```
[12]: covidSP['deaths'].plot()
```

```
[12]: <matplotlib.axes._subplots.AxesSubplot at 0x7f1cf26a7f10>
```



```
[13]: covidSP['confirmed'].plot()
```

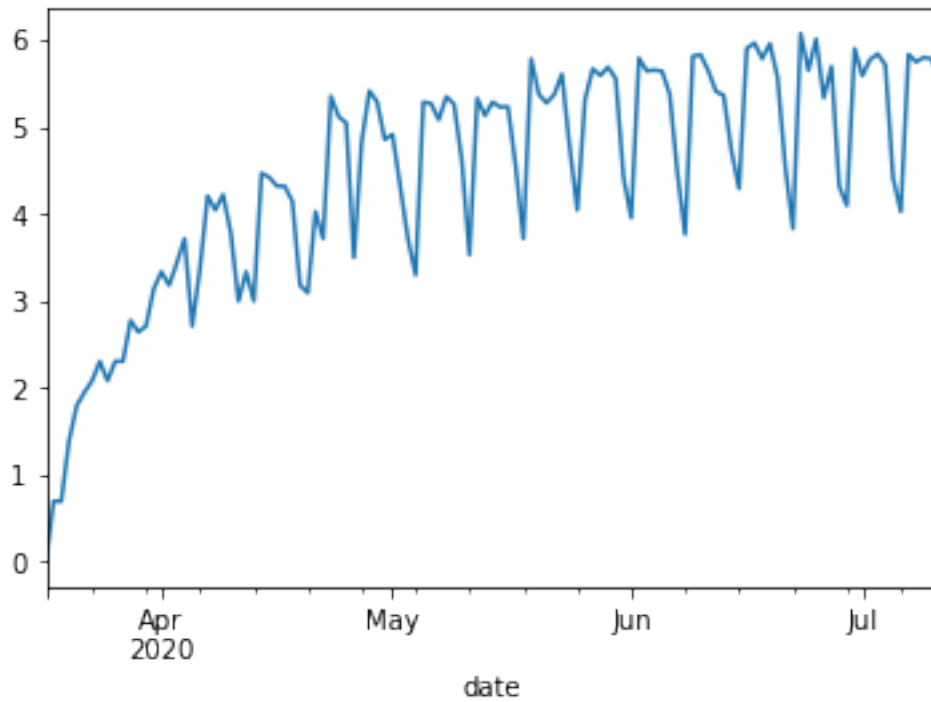
```
[13]: <matplotlib.axes._subplots.AxesSubplot at 0x7f1cf25bc190>
```



```
[14]: # Gráfico do log das mortes
```

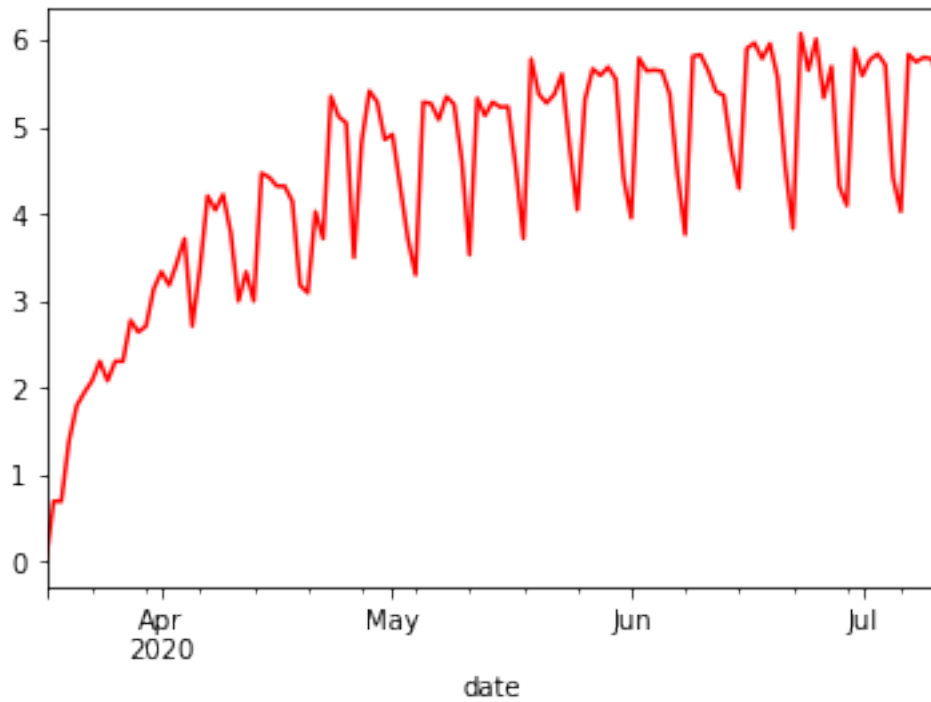
```
np.log(covidSP.loc[covidSP['deaths']>0]['deaths']).plot()
```

```
[14]: <matplotlib.axes._subplots.AxesSubplot at 0x7f1cf2525e90>
```



```
[15]: # Gráfico do log das mortes - mudando a cor
      np.log(covidSP.loc[covidSP['deaths']>0]['deaths']).plot(c='red')
```

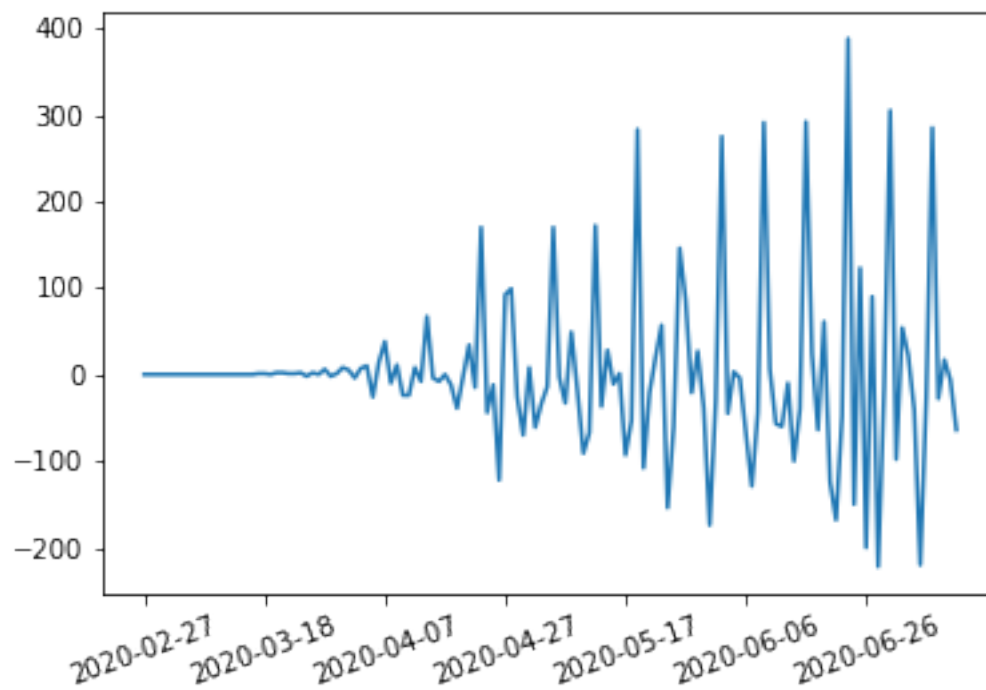
```
[15]: <matplotlib.axes._subplots.AxesSubplot at 0x7f1cf044d450>
```



```
[16]: # Gráfico da diferença das mortes - Uma tentativa de eliminar a tendência

y = np.diff(covidSP['deaths'])
x = covidSP.index[1:]

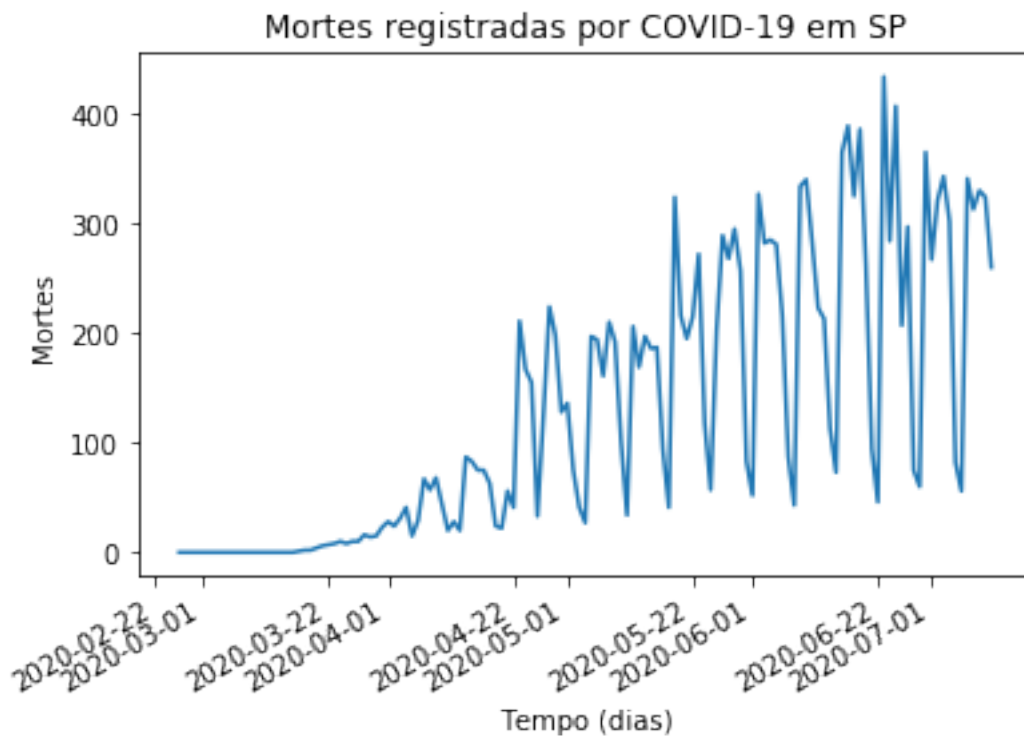
plt.plot(x,y)
plt.xticks(np.arange(x.min(),x.max(), 20, dtype='datetime64[D]'),rotation=20);
```



```
[17]: title = 'Mortes registradas por COVID-19 em SP'
      ylabel = 'Mortes'
      xlabel = 'Tempo (dias)'

      ax=covidSP['deaths'].plot(title=title)
      ax.autoscale(axis='both')
      ax.set(xlabel=xlabel,ylabel=ylabel)
```

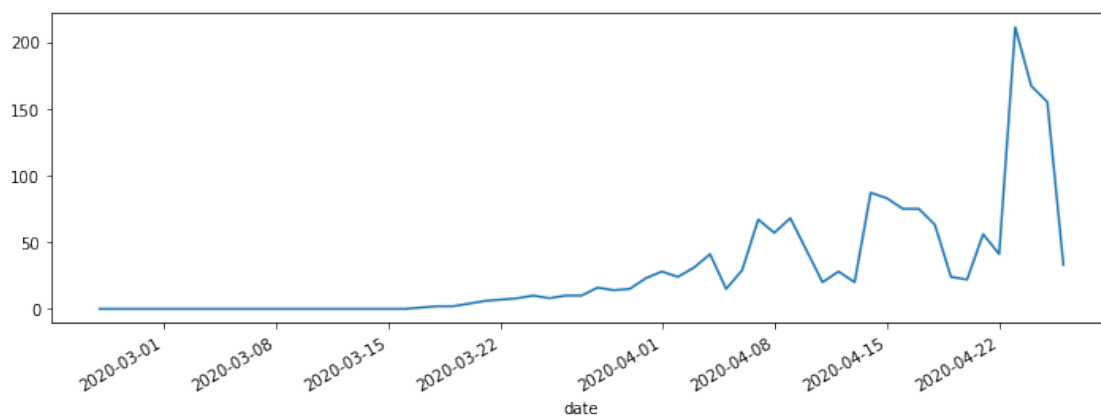
```
[17]: [Text(0, 0.5, 'Mortes'), Text(0.5, 0, 'Tempo (dias)')]
```



```
[18]: # Visualizando apenas uma parte da série

covidSP['deaths']['2020-02-26':'2020-04-26'].plot(figsize=(12,4))
```

```
[18]: <matplotlib.axes._subplots.AxesSubplot at 0x7f1cf02b4590>
```

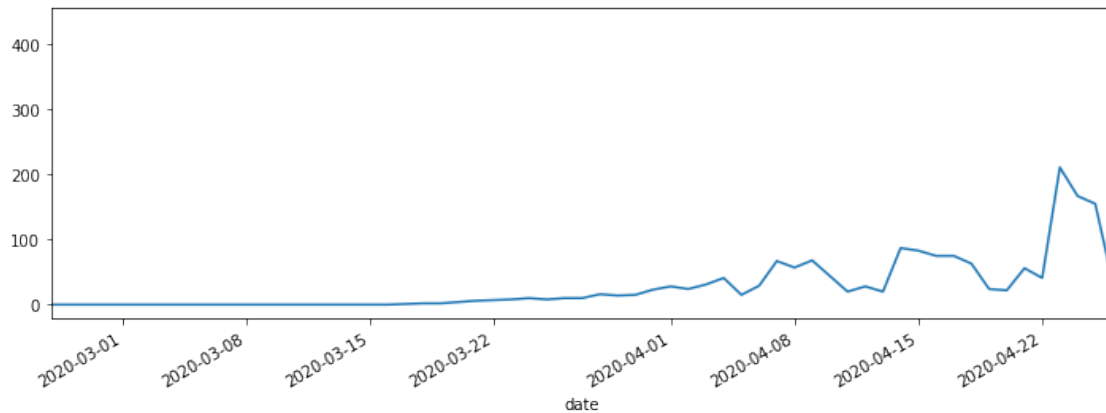


```
[19]: # Outra forma de visualizar apenas uma parte da série
```



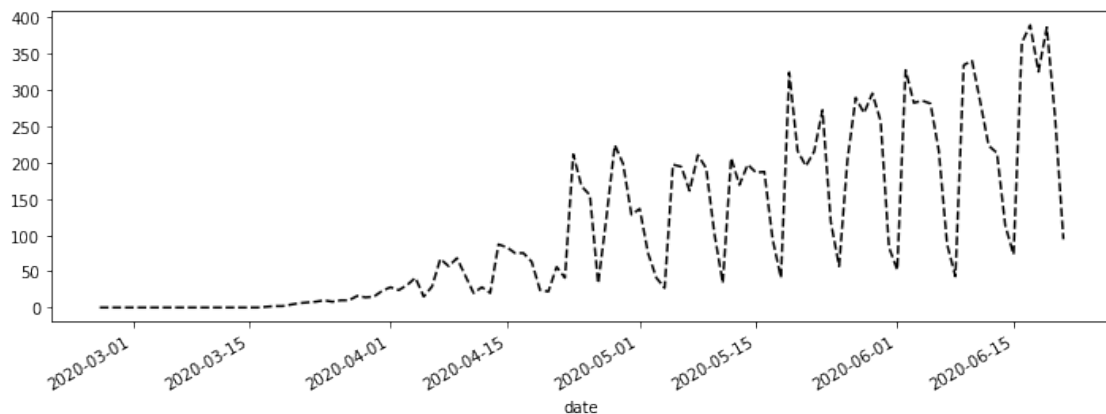
```
covidSP['deaths'].plot(figsize=(12,4),xlim=['2020-02-26','2020-04-26'])
```

[19]: <matplotlib.axes._subplots.AxesSubplot at 0x7f1cf022d550>



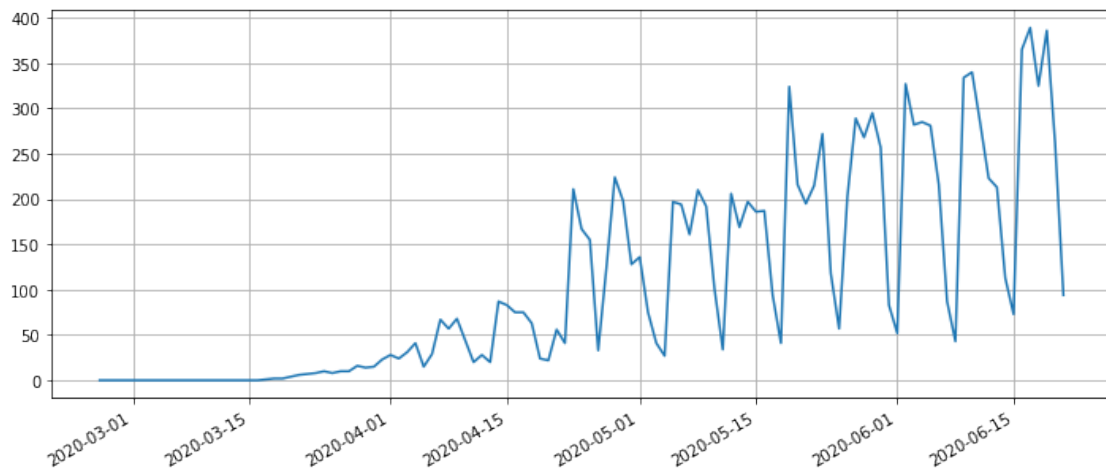
```
[20]: # Altera tipo de linha  
  
covidSP['deaths']['2020-02-26':'2020-06-21'].  
    →plot(figsize=(12,4),ls='--',c='black')
```

[20]: <matplotlib.axes._subplots.AxesSubplot at 0x7f1cf019a6d0>



```
[21]: # Adicionando linhas de grade ao gráfico  
  
ax=covidSP['deaths']['2020-02-26':'2020-06-21'].plot(figsize=(12,5))  
ax.set(xlabel='')  
  
ax.xaxis.grid(True)
```

```
ax.yaxis.grid(True)
```



Exercício:

Repita os gráficos acima considerando os dados covidRJ.csv de exercício anterior, e/ou de outros estados. Busque outras formas de alterar as representações gráficas.