VISUALIZING TIME SERIES DATA IN PYTHON

Autocorrelation and Partial autocorrelation

Thomas Vincent
Head of Data Science, Getty Images

Autocorrelation in time series data

- Autocorrelation is measured as the correlation between a time series and a delayed copy of itself
- For example, an autocorrelation of order 3 returns the correlation between a time series at points (t_1 , t_2 , t_3 , ...) and its own values lagged by 3 time points, i.e. (t_4 , t_5 , t_6 , ...)
- It is used to find repetitive patterns or periodic signal in time series

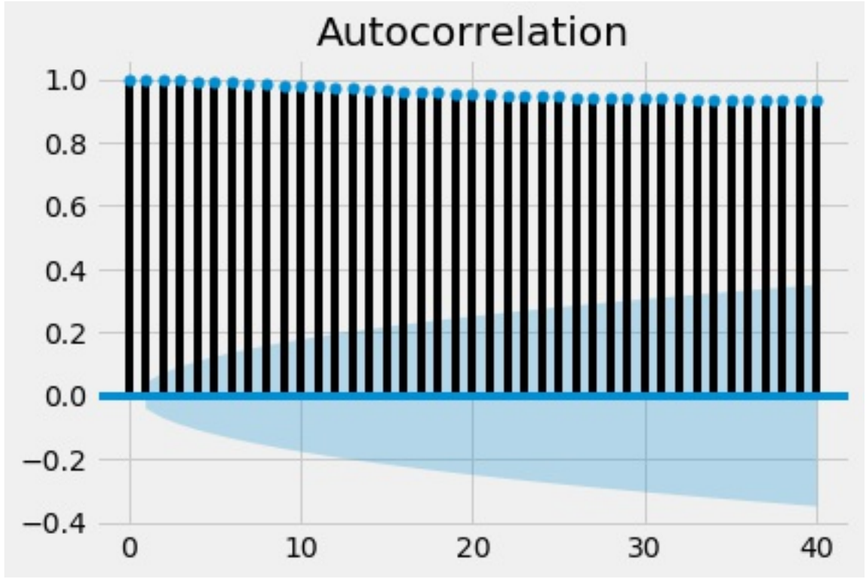
statsmodels

statsmodels is a Python module that provides classes and functions for the estimation of many different statistical models, as well as for conducting statistical tests, and statistical data exploration.

Plotting autocorrelations

```
In [1]: import matplotlib.pyplot as plt

In [2]: from statsmodels.graphics import tsaplots

In [3]: fig = tsaplots.plot_acf(co2_levels['co2'], lags=40)

In [4]: plt.show()
```

Interpreting autocorrelation plots



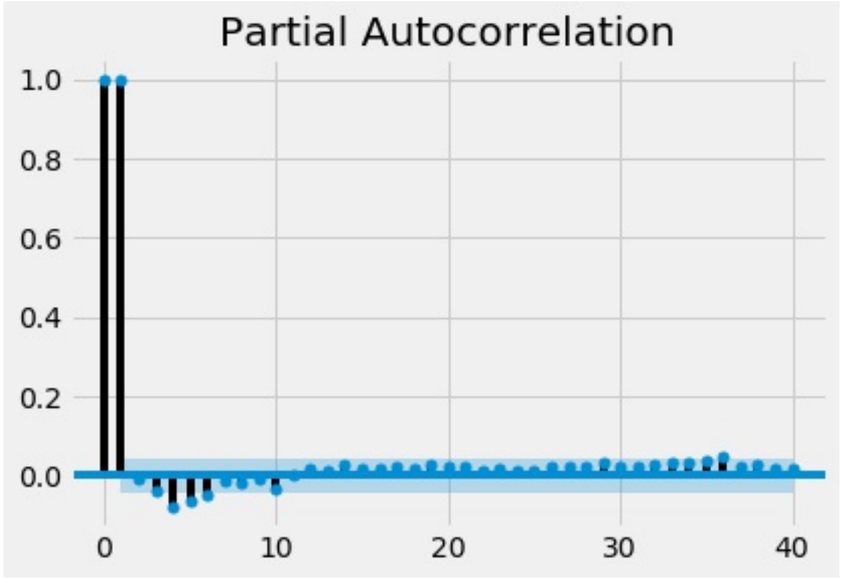
Partial autocorrelation in time series data

- Contrary to autocorrelation, partial autocorrelation removes the effect of previous time points
- For example, a partial autocorrelation function of order 3 returns the correlation between our time series (t1, t2, t3, ...) and lagged values of itself by 3 time points (t4, t5, t6, ...), but only after removing all effects attributable to lags 1 and 2

Plotting partial autocorrelations

```
In [1]: import matplotlib.pyplot as plt

In [2]: from statsmodels.graphics import tsaplots

In [3]: fig = tsaplots.plot_pacf(co2_levels['co2'], lags=40

In [4]: plt.show()
```

Interpreting partial autocorrelations plot



VISUALIZING TIME SERIES DATA IN PYTHON

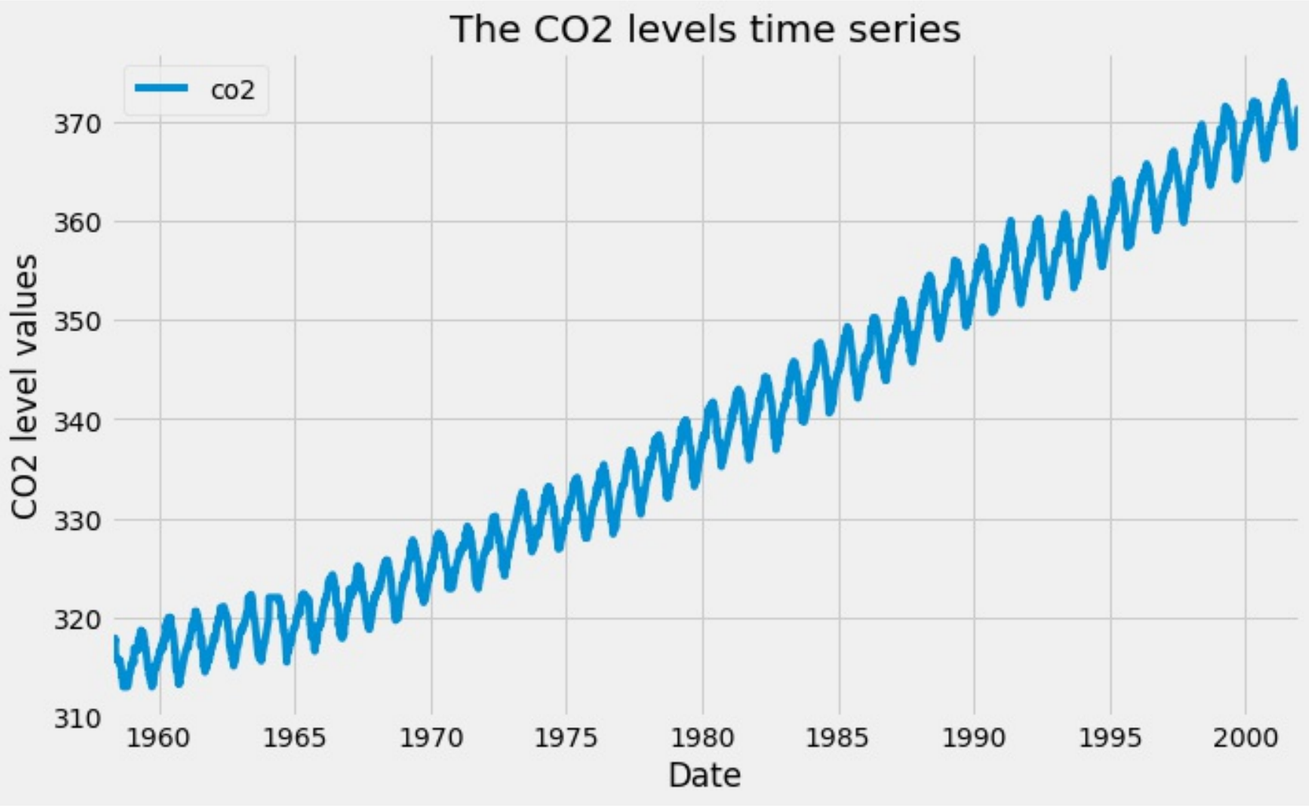
Let's practice!

VISUALIZING TIME SERIES DATA IN PYTHON

Seasonality, trend and noise in time series data

Thomas Vincent
Head of Data Science, Getty Images

Properties of time series



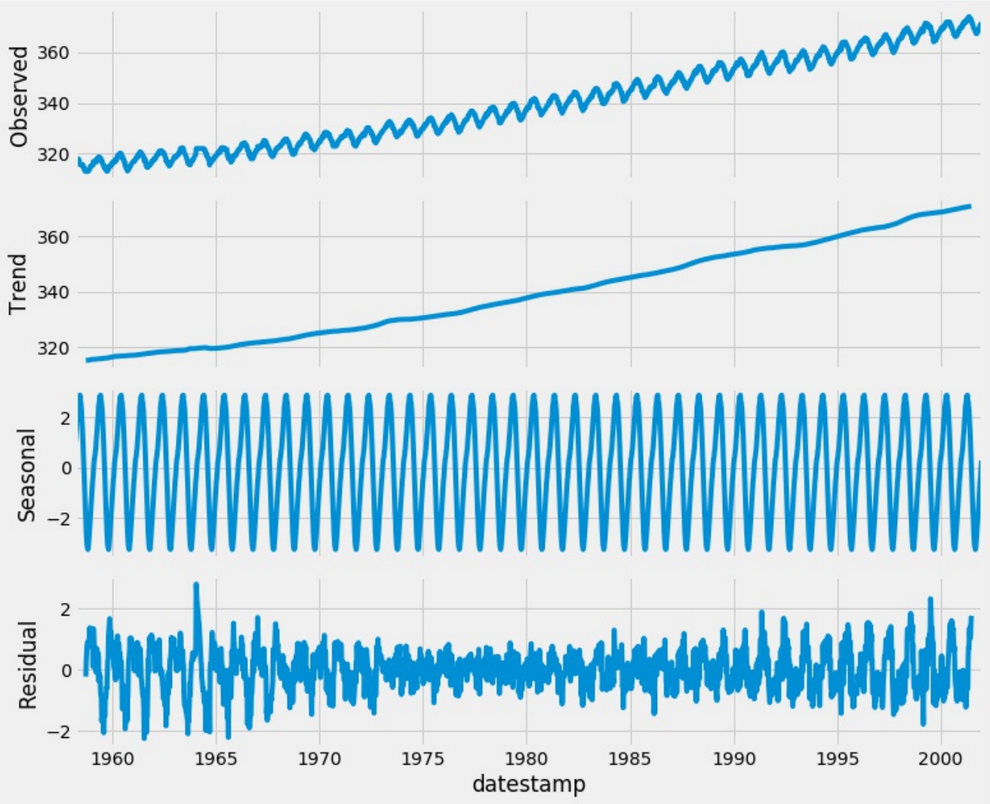
The properties of time series

- Seasonality: does the data display a clear periodic pattern?
- Trend: does the data follow a consistent upwards or downwards slope?
- Noise: are there any outlier points or missing values that are not consistent with the rest of the data?

Time series decomposition

```
In [1]: import statsmodels.api as sm

In [2]: import matplotlib.pyplot as plt

In [3]: from pylab import rcParams

In [4]: rcParams['figure.figsize'] = 11, 9

In [5]: decomposition = sm.tsa.seasonal_decompose(co2_levels['co2'])

In [6]: fig = decomposition.plot()

In [7]: plt.show()
```

A plot of time series decomposition on the CO2 data



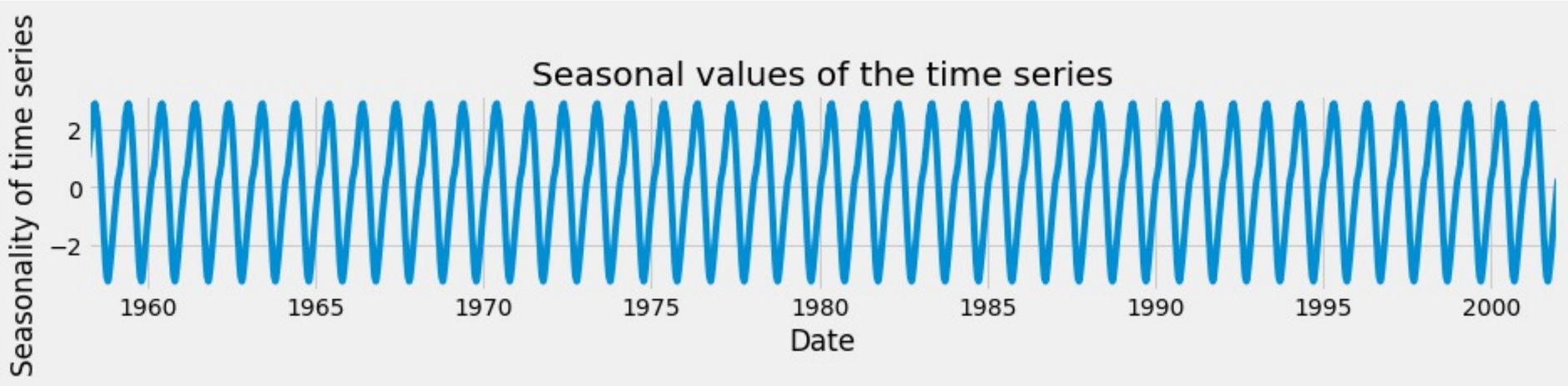
Extracting components from time series decomposition

```
In [1]: print(dir(decomposition))
['__class__',
 '__delattr__',
 '__dict__',
 ...
 'plot',
 'resid',
 'seasonal',
 'trend']

In [2]: print(decomposition.seasonal)
datestamp
1958-03-29    1.028042
1958-04-05    1.235242
1958-04-12    1.412344
1958-04-19    1.701186
```

Seasonality component in time series

```
In [1]: decomp_seasonal = decomposition.seasonal

In [1]: ax = decomp_seasonal.plot(figsize=(14, 2))

In [2]: ax.set_xlabel('Date')

In [3]: ax.set_ylabel('Seasonality of time series')

In [4]: ax.set_title('Seasonal values of the time series')

In [5]: plt.show()
```

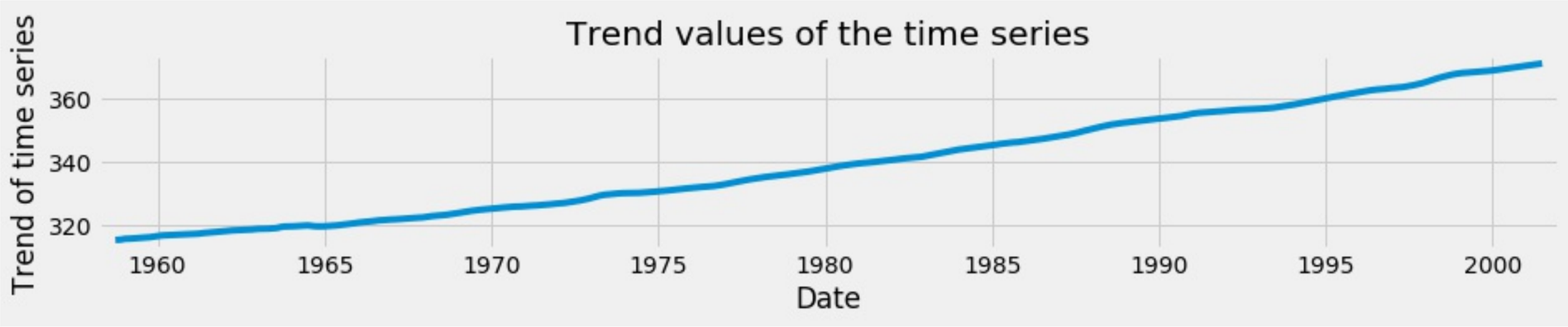
Seasonality component in time series



Trend component in time series

```
In [1]: decomp_trend = decomposition.trend

In [2]: ax = decomp_trend.plot(figsize=(14, 2))

In [3]: ax.set_xlabel('Date')

In [4]: ax.set_ylabel('Trend of time series')

In [5]: ax.set_title('Trend values of the time series')

In [6]: plt.show()
```

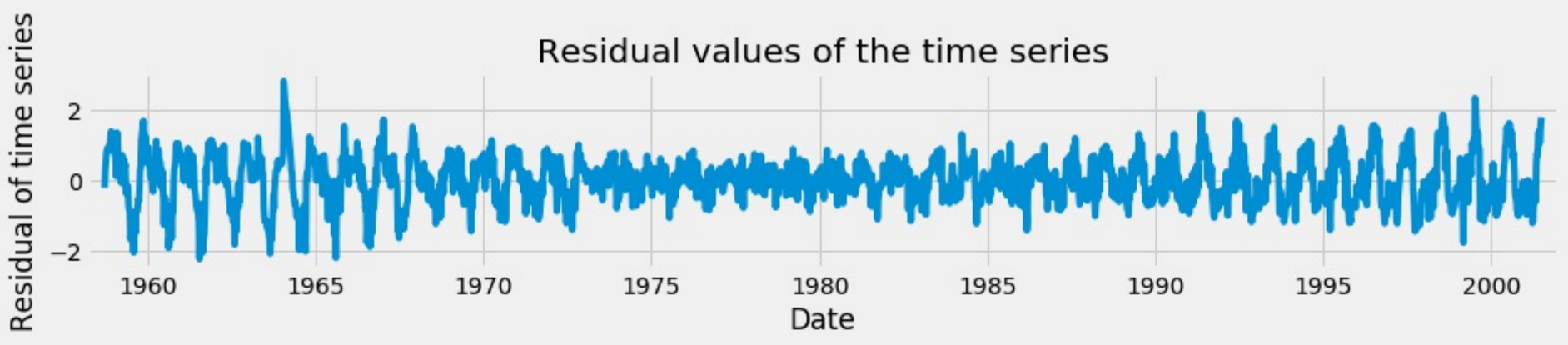
Trend component in time series



Noise component in time series

```
In [1]: decomp_resid = decomp.resid

In [3]: ax = decomp_resid.plot(figsize=(14, 2))

In [4]: ax.set_xlabel('Date')

In [4]: ax.set_ylabel('Residual of time series')

In [5]: ax.set_title('Residual values of the time series')

In [6]: plt.show()
```

Noise component in time series



VISUALIZING TIME SERIES DATA IN PYTHON

Let's practice!

VISUALIZING TIME SERIES DATA IN PYTHON

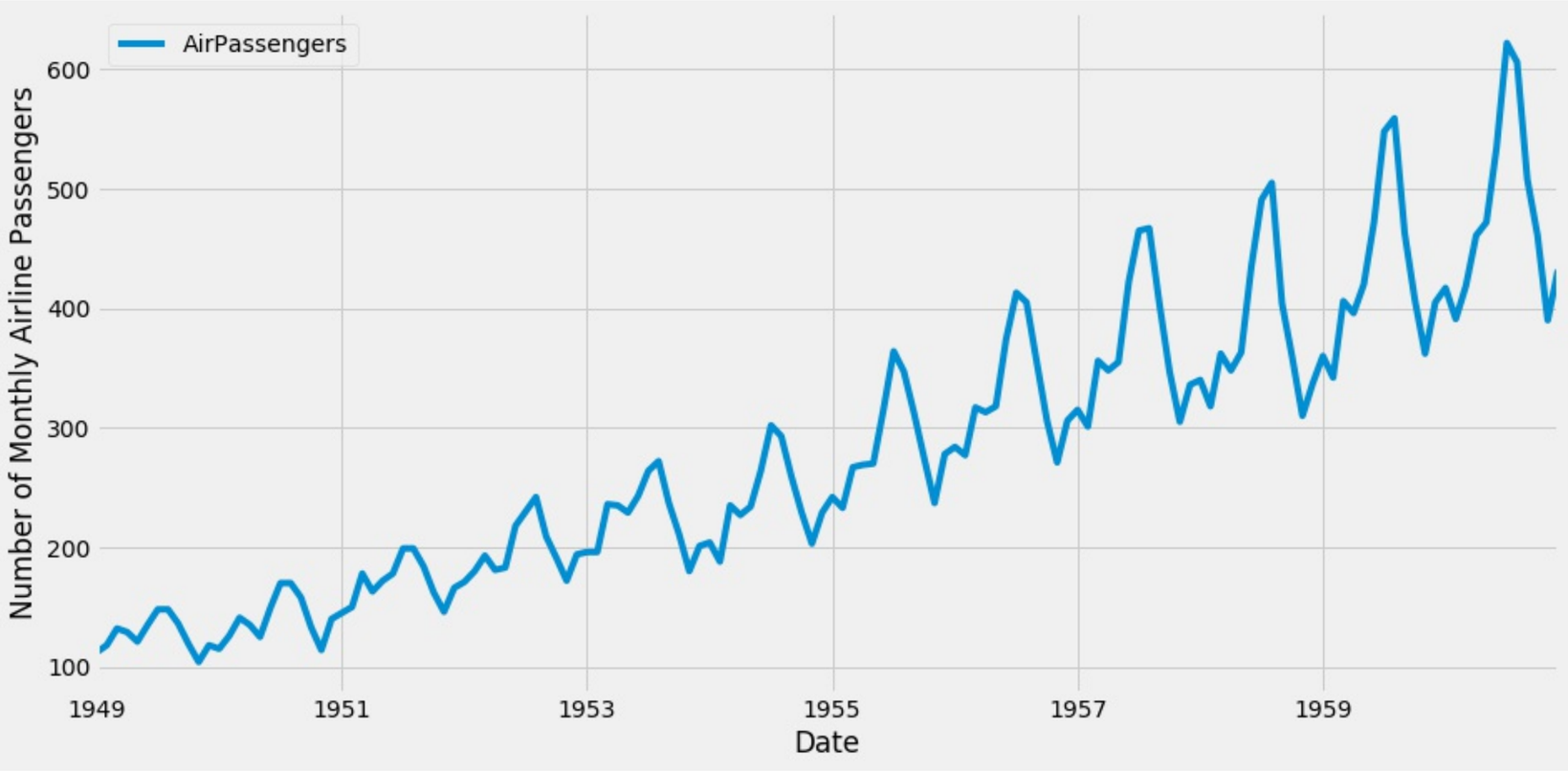
A review on what you have learned so far

Thomas Vincent
Head of Data Science, Getty Images

So far ...

- Visualize aggregates of time series data
- Extract statistical summaries
- Autocorrelation and Partial autocorrelation
- Time series decomposition

The airline dataset



VISUALIZING TIME SERIES DATA IN PYTHON

Let's analyze this data!