VISUALIZING TIME SERIES DATA IN PYTHON

Welcome to the course!

Thomas Vincent
Senior Data Science Engineer, DigitalOcean

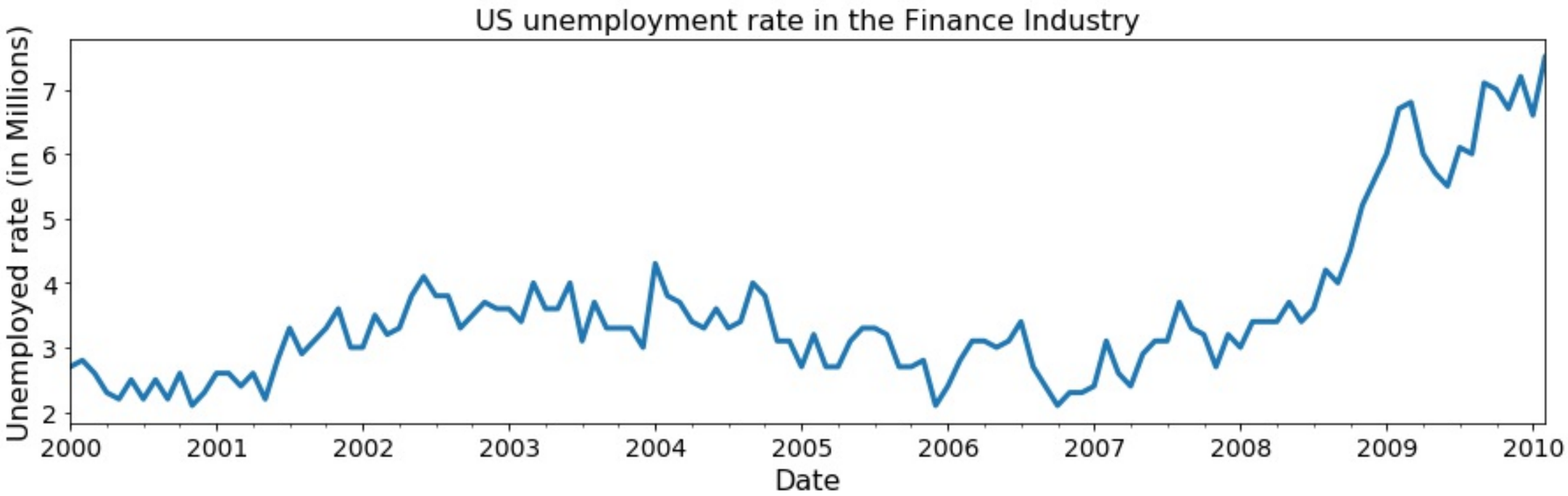
Prerequisites

- Intro to Python for Data Science
- Intermediate Python for Data Science

Time series in the field of Data Science

- Time series are a fundamental way to store and analyse many types of data.
- Financial, weather and device data are all best handled as time series.

Time series in the field of Data Science



Course overview

- Chapter 1: Getting started and personalizing your first time series plot
- Chapter 2: Summarizing and describing time series data
- Chapter 3: Advanced time series analysis
- Chapter 4: Working with multiple time series
- Chapter 5: Case Study

Reading data with Pandas

```
In [1]: import pandas as pd

In [2]: df = pd.read_csv('ch2_co2_levels.csv')

In [3]: print(df)
        datestamp    co2
0      1958-03-29  316.1
1      1958-04-05  317.3
2      1958-04-12  317.6
...
...
...
2281   2001-12-15  371.2
2282   2001-12-22  371.3
2283   2001-12-29  371.5
```

Preview data with Pandas

```
In [4]: print(df.head(n=5))
    datestamp    co2
0  1958-03-29  316.1
1  1958-04-05  317.3
2  1958-04-12  317.6
3  1958-04-19  317.5
4  1958-04-26  316.4

In [5]: print(df.tail(n=5))
       datestamp    co2
2279  2001-12-01  370.3
2280  2001-12-08  370.8
2281  2001-12-15  371.2
2282  2001-12-22  371.3
2283  2001-12-29  371.5
```

Check data types with Pandas

```
In [6]: print(df.dtypes)
datestamp     object
co2          float64
dtype: object
```

Working with dates

To work with time series data in pandas, your date columns needs to be of the datetime64 type.

```
In [7]: pd.to_datetime(['2009/07/31', 'test'])
ValueError: Unknown string format

In [8]: pd.to_datetime(['2009/07/31', 'test'], errors='coerce')
Out[8]: DatetimeIndex(['2009-07-31', 'NaT'],
    dtype='datetime64[ns]', freq=None)
```

VISUALIZING TIME SERIES DATA IN PYTHON

Let's get started!

VISUALIZING TIME SERIES DATA IN PYTHON

Plot your first time series

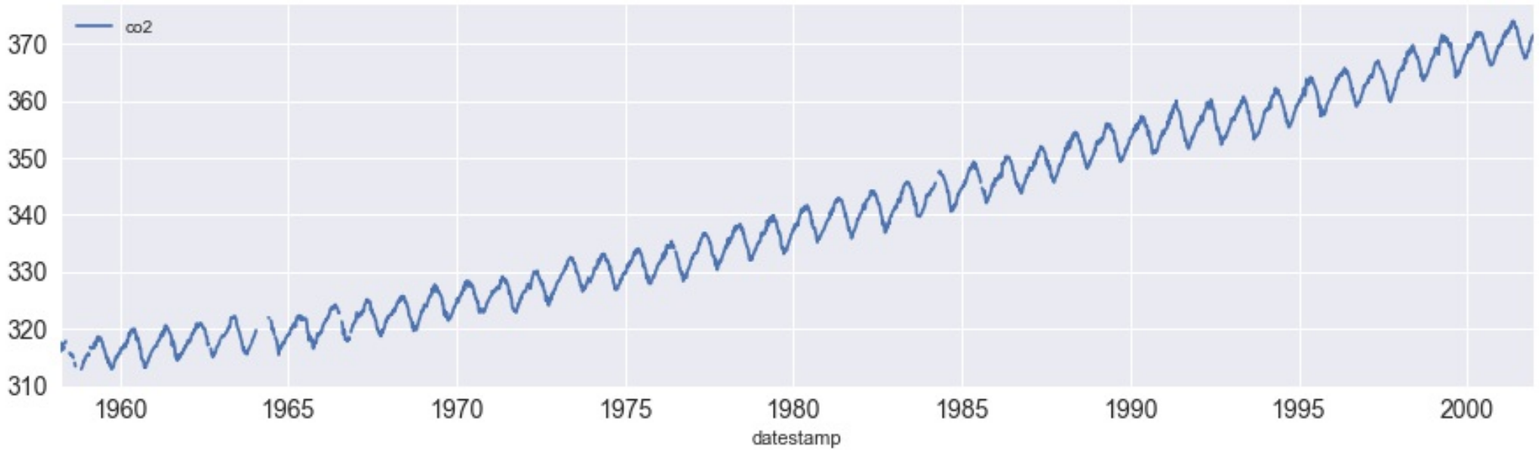
Thomas Vincent
Senior Data Science Engineer, DigitalOcean

The Matplotlib library

- In Python, matplotlib is an extensive package used to plot data
- The pyplot submodule of matplotlib is traditionally imported using the `plt` alias

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

Plotting time series data



Plotting time series data

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

In [2]: import pandas as pd

In [3]: df = df.set_index('date_column')

In [4]: df.plot()

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

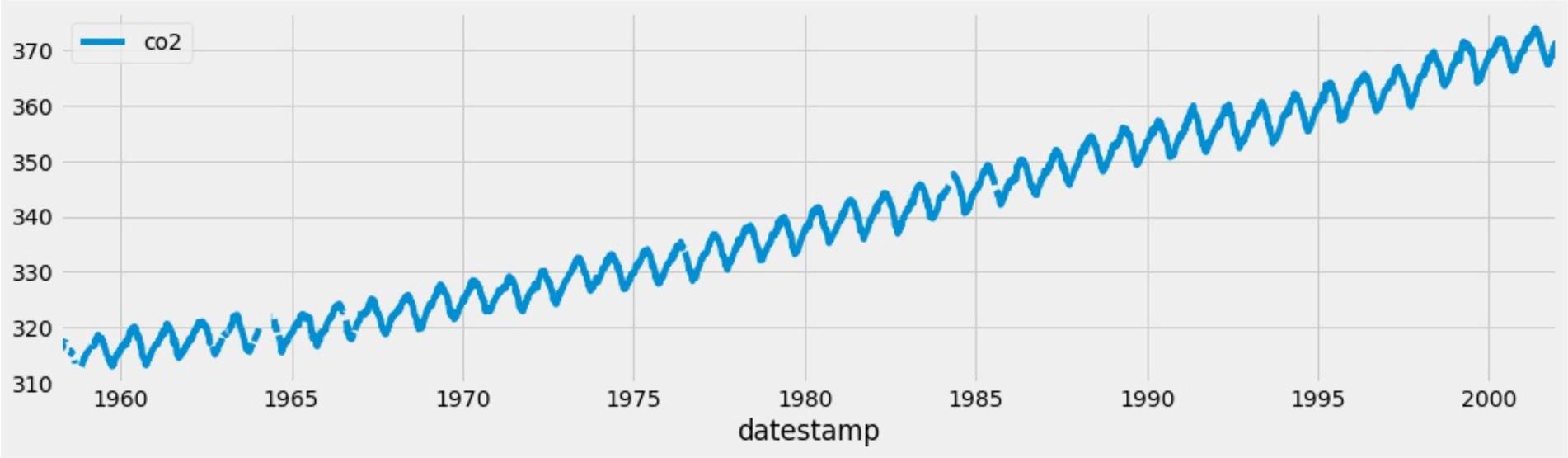
Adding style to your plots

```
In [6]: plt.style.use('fivethirtyeight')

In [7]: df.plot()

In [8]: plt.show()
```

FiveThirtyEight style



Matplotlib style sheets

```
In [9]: print(plt.style.available)
['seaborn-dark-palette', 'seaborn-darkgrid',
'seaborn-dark', 'seaborn-notebook', 'seaborn-pastel',
'seaborn-white', 'classic', 'ggplot', 'grayscale',
'dark_background', 'seaborn-poster',
'seaborn-muted', 'seaborn', 'bmh', 'seaborn-paper',
'seaborn-whitegrid', 'seaborn-bright', 'seaborn-talk',
'fivethirtyeight', 'seaborn-colorblind', 'seaborn-deep'
'seaborn-ticks']
```

Describing your graphs with labels

```
In [10]: ax = df.plot(color='blue')

In [11]: ax.set_xlabel('Date')

In [12]: ax.set_ylabel('The values of my Y axis')

In [13]: ax.set_title('The title of my plot')

In [14]: plt.show()
```

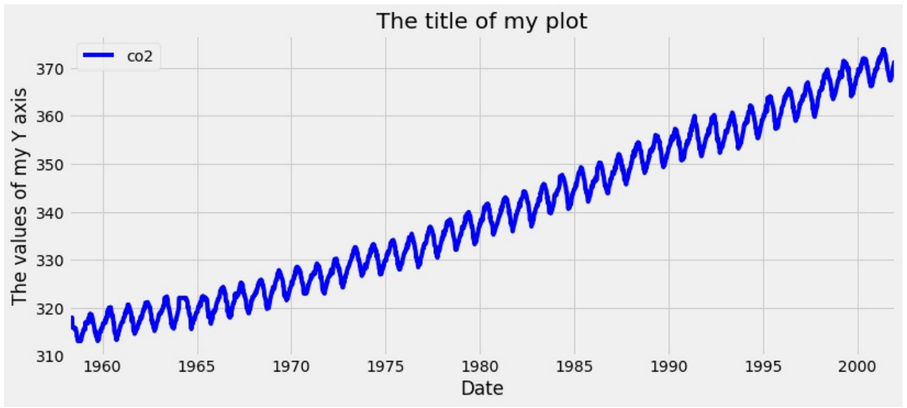


Figure size, linewidth, linestyle and fontsize

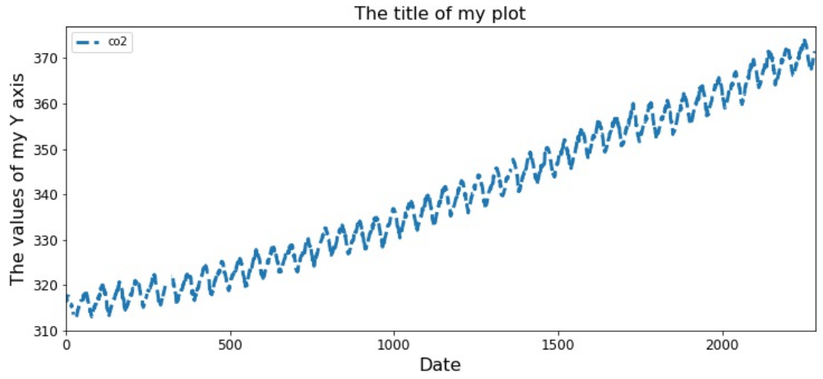
```
In [15]: ax = df.plot(figsize=(12, 5),
                      fontsize=12,
                      linewidth=3,
                      linestyle='--')

In [16]: ax.set_xlabel('Date', fontsize=16)

In [17]: ax.set_ylabel('The values of my Y axis', fontsize=16)

In [18]: ax.set_title('The title of my plot', fontsize=16)

In [19]: plt.show()
```



VISUALIZING TIME SERIES DATA IN PYTHON

Let's practice!

VISUALIZING TIME SERIES DATA IN PYTHON

Customize your time series plot

Thomas Vincent
Senior Data Science Engineer, DigitalOcean

Slicing time series data

```
In [1]: discoveries['1960':'1970']

In [2]: discoveries['1950-01':'1950-12']

In [3]: discoveries['1960-01-01':'1960-01-15']
```

Plotting subset of your time series data

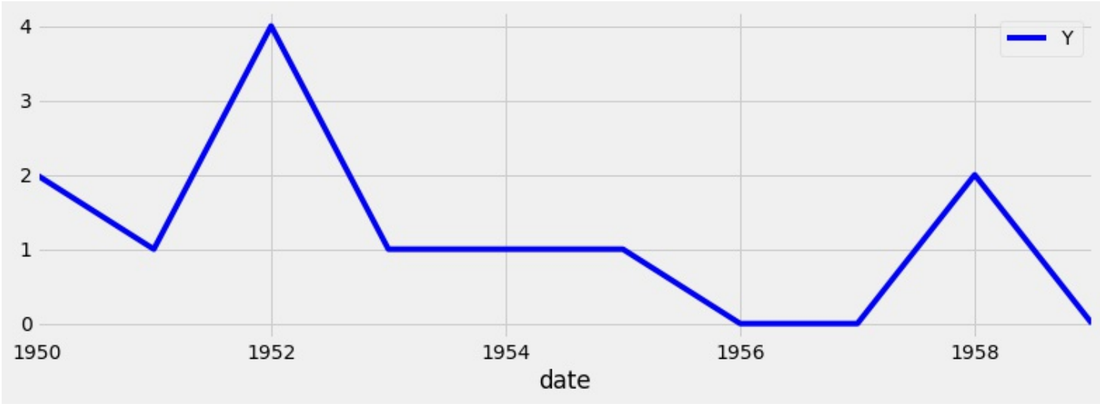
```
In [4]: import matplotlib.pyplot as plt

In [5]: plt.style.use('fivethirtyeight')

In [6]: df_subset = discoveries['1960':'1970']

In [7]: ax = df_subset.plot(color='blue', fontsize=14)

In [8]: plt.show()
```



Adding markers

```
In [1]: ax.axvline(x='1969-01-01', color='red', linestyle='--')
```

```
In [2]: ax.axhline(y=100, color='green', linestyle='--')
```

Using markers: the full code

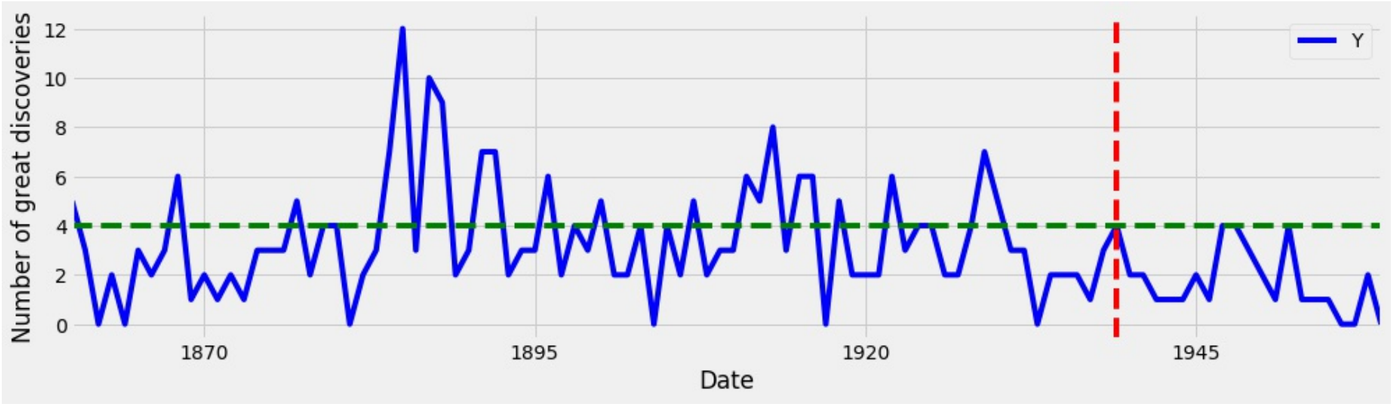
```
In [1]: ax = discoveries.plot(color='blue')

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

In [3]: ax.set_ylabel('Number of great discoveries')

In [4]: ax.axvline('1969-01-01', color='red', linestyle='--')

In [5]: ax.axhline(4, color='green', linestyle='--')
```



Highlighting regions of interest

```
In [1]: ax.axvspan('1964-01-01', '1968-01-01', color='red', alpha=0.5)

In [2]: ax.axhspan(8, 6, color='green', alpha=0.2)
```

Highlighting regions of interest: the full code

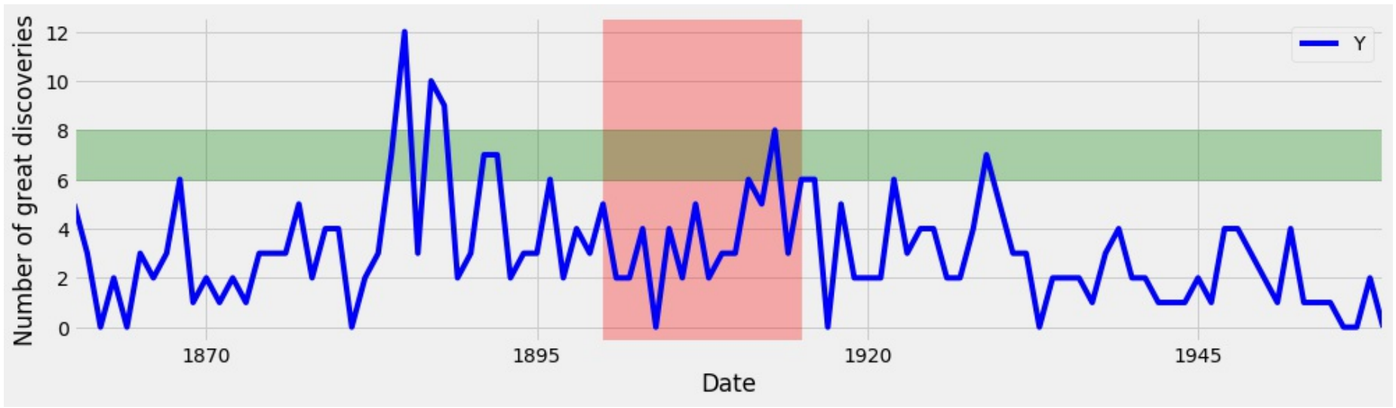
```
In [1]: ax = discoveries.plot(color='blue')

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

In [3]: ax.set_ylabel('Number of great discoveries')

In [4]: ax.axvspan('1964-01-01', '1968-01-01', color='red', alpha=0.3)

In [5]: ax.axhspan(8, 6, color='green', alpha=0.3)
```



VISUALIZING TIME SERIES DATA IN PYTHON

Let's practice!