

▼ pandas.plotting subpackage

Name : Jann Moises Nyll B. De los Reyes

Section : CPE22S3

Submitted to: Engr. Roman Richard

Date : March 28, 2024

Pandas provides some extra plotting functions for a few select plot types.

About the Data

In this notebook, we will be working with Facebook's stock price throughout 2018.

▼ Setup

```
1 %matplotlib inline
2 import matplotlib.pyplot as plt
3 import numpy as np
4 import pandas as pd
5
6 fb = pd.read_csv('/content/drive/MyDrive/Module 9: Data Visualization using Pandas, Matplotlib and Seaborn/fb_stock_prices_2018.csv',
7                 index_col = 'date', parse_dates = True)
```

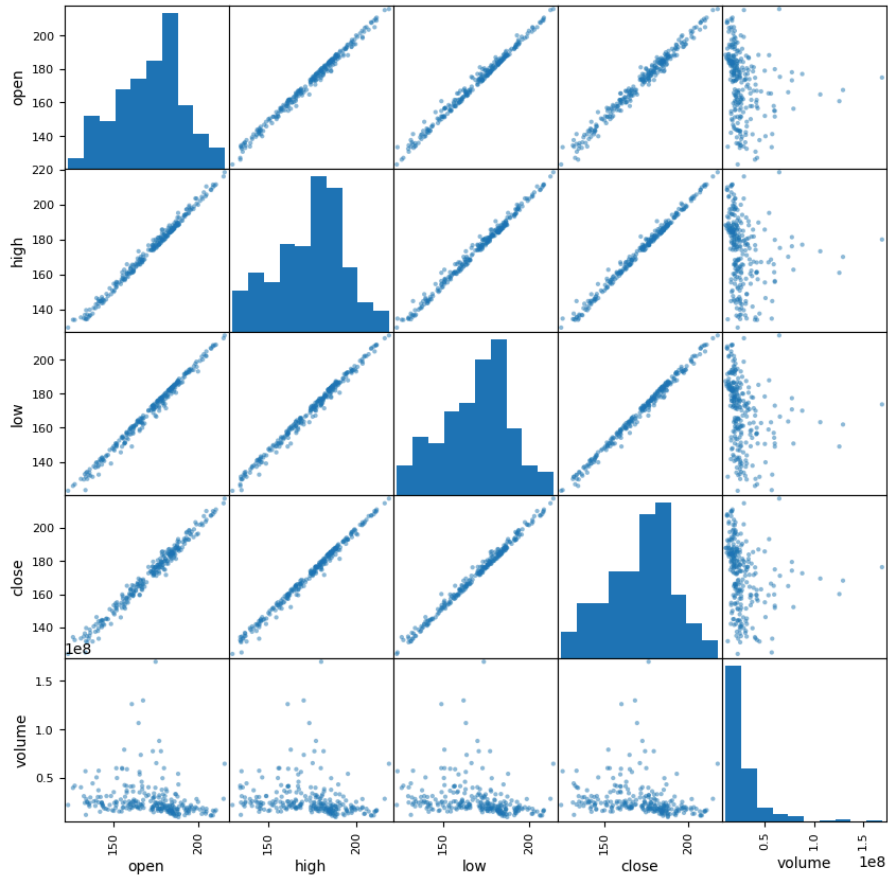
▼ Scatter matrix

```
1 from pandas.plotting import scatter_matrix
2 scatter_matrix(fb, figsize=(10,10))
```

```

<Axes: xlabel='low', ylabel='high'>,
<Axes: xlabel='close', ylabel='high'>,
<Axes: xlabel='volume', ylabel='high'>],
[<Axes: xlabel='open', ylabel='low'>,
<Axes: xlabel='high', ylabel='low'>,
<Axes: xlabel='low', ylabel='low'>,
<Axes: xlabel='close', ylabel='low'>,
<Axes: xlabel='volume', ylabel='low'>],
[<Axes: xlabel='open', ylabel='close'>,
<Axes: xlabel='high', ylabel='close'>,
<Axes: xlabel='low', ylabel='close'>,
<Axes: xlabel='close', ylabel='close'>,
<Axes: xlabel='volume', ylabel='close'>],
[<Axes: xlabel='open', ylabel='volume'>,
<Axes: xlabel='high', ylabel='volume'>,
<Axes: xlabel='low', ylabel='volume'>,
<Axes: xlabel='close', ylabel='volume'>,
<Axes: xlabel='volume', ylabel='volume'>]], dtype=object)

```

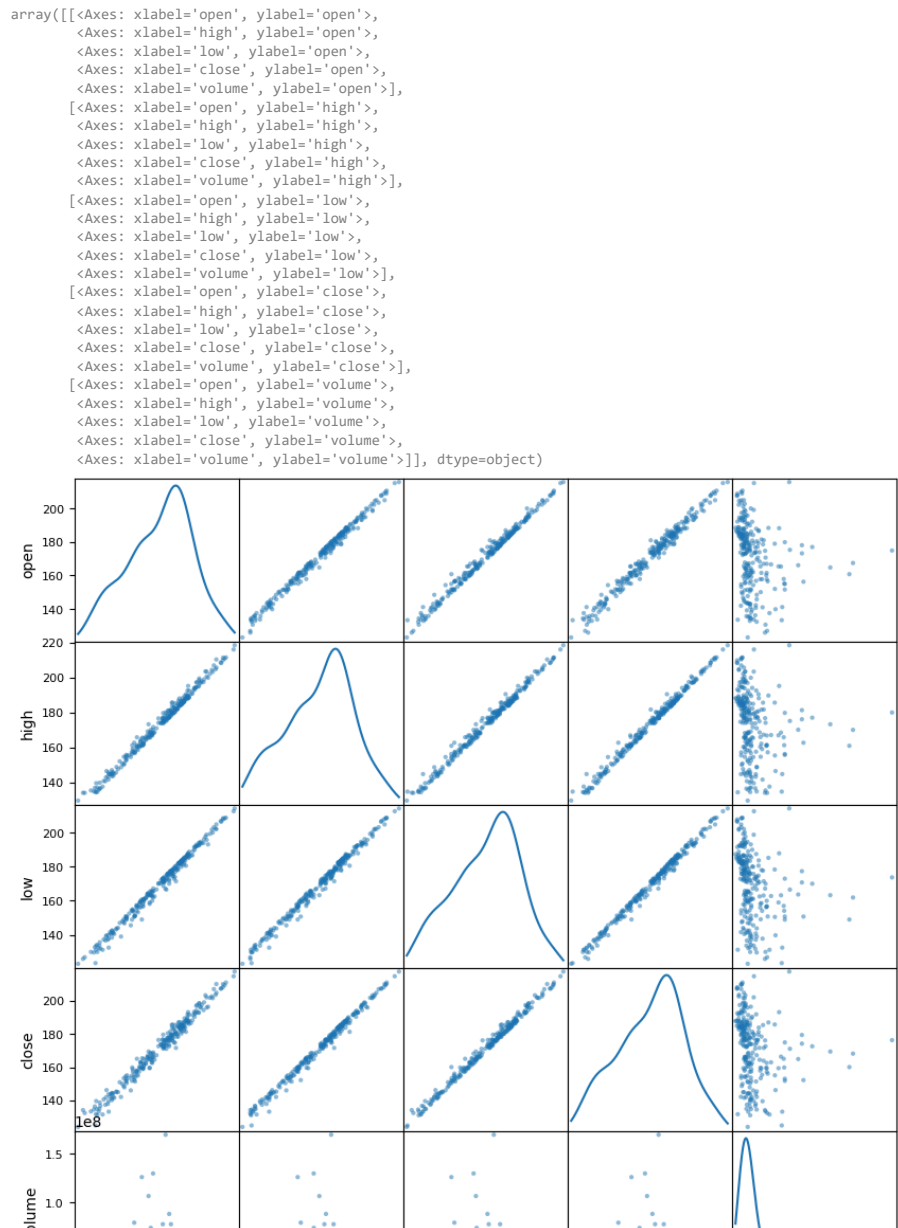


Changing the diagonal from histograms to KDE

```

1 scatter_matrix(fb, figsize = (10,10), diagonal = 'kde')

```

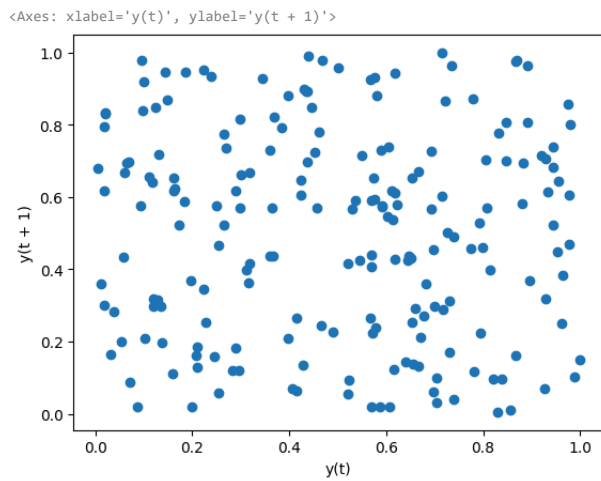


Double-click (or enter) to edit

Lag plot

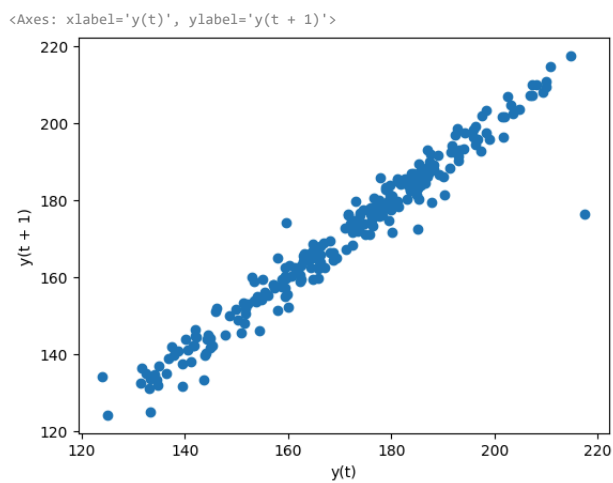
Lag plots let us see how the variable correlations with past observations of itself. Random data has no pattern

```
1 from pandas.plotting import lag_plot
2 np.random.seed(0) #make this repeatable
3 lag_plot(pd.Series(np.random.random(size=200)))
```



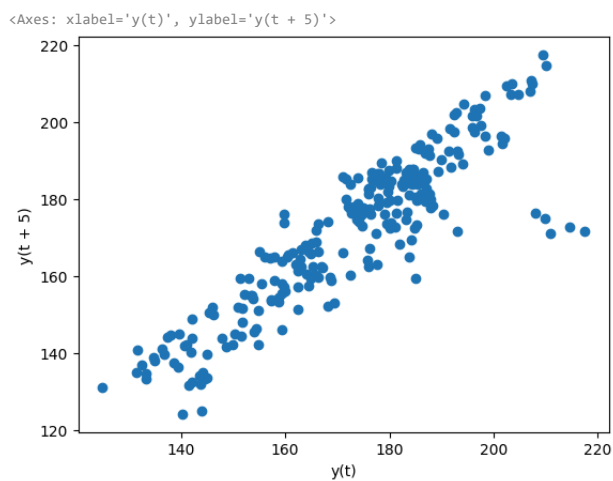
Data with some level of correlation to itself (autocorrelation) may have patterns. Stock prices are highly auto-correlated:

```
1 lag_plot(fb.close)
```



The default lag is 1, but we can alter this with the lag parameter. Let's look at a 5 day lag (a week of trading activity):

```
1 lag_plot(fb.close, lag = 5)
```

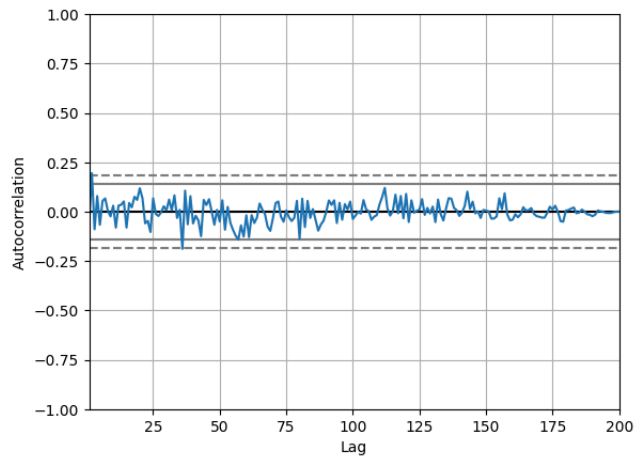


▼ Autocorrelation plots

We can use the autocorrelation plot to see if this relationship may be meaningful or just noise. Random data will not have any significant autocorrelation (it stays within the bounds below)

```
1 from pandas.plotting import autocorrelation_plot
2 np.random.seed(0) #make this repeatable
3 autocorrelation_plot(pd.Series(np.random.random(size = 200)))
```

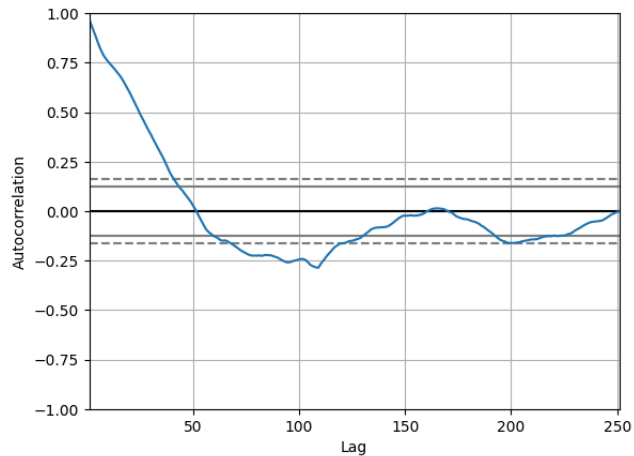
<Axes: xlabel='Lag', ylabel='Autocorrelation'>



Stock data, on the other hand, does have significant autocorrelation:

```
1 autocorrelation_plot(fb.close)
```

<Axes: xlabel='Lag', ylabel='Autocorrelation'>



✓ Bootstrap plot

This plot helps us understand the uncertainty in our summary statistics:

```
1 from pandas.plotting import bootstrap_plot
2 fig = bootstrap_plot(fb.volume,fig = plt.figure(figsize=(10,6)))
```

