

Machine Learning

TP 1 : données et statistiques

Useful tools

- `virtualenv`
- `pip`
- `ipython`
- `ipython notebook`
- `conda.pydata.org`

Notes

- `pip install -r requirements.txt`
- `ipython` offers tab completion (vs `python`)
- `ipython notebook` opens in a browser, caches cell output but not cell state

```
import pandas as pd
import numpy as np
import scipy
import matplotlib.pyplot as plt
```

Dataframe has many constructors. For example,

```
1 In [5]: pd.DataFrame({ 'A' : 1.,
2                        'B' : pd.Timestamp('20161209'),
3                        'C' : pd.Series(1,index=list(range(4)),dtype='float32'),
4                        'D' : np.array([3] * 4, dtype='int32'),
5                        'E' : pd.Categorical(["test","train","test","train"]),
6                        'F' : 'hello' })
7 Out[5]:
```

	A	B	C	D	E	F
0	1	2016-12-09	1	3	test	hello
1	1	2016-12-09	1	3	train	hello
2	1	2016-12-09	1	3	test	hello
3	1	2016-12-09	1	3	train	hello

Viewing data

```
1 In [16]: dates = pd.date_range('20161209', periods=4, freq='1w')
2
3 In [17]: df = pd.DataFrame(np.random.randn(4,5), index=dates,
4                             columns=list('ABCDE'))
5
6 In [18]: df.head()
7 Out[18]:
```

	A	B	C	D	E
2016-12-11	-1.303610	-1.235823	0.621914	0.379340	-0.326934
2016-12-18	-1.218197	-1.113826	0.546314	-0.255001	-0.135573
2016-12-25	-0.124625	0.337268	-0.406295	0.587049	-0.904906
2017-01-01	-0.283182	-0.866213	0.051509	0.693037	-0.661055

Basic data exploration

```
1 In [19]: df.describe()
2 Out[19]:
```

	A	B	C	D	E
count	4.000000	4.000000	4.000000	4.000000	4.000000
mean	-0.732403	-0.719648	0.203361	0.351106	-0.507117
std	0.614672	0.721194	0.478728	0.424558	0.342755
min	-1.303610	-1.235823	-0.406295	-0.255001	-0.904906
25%	-1.239550	-1.144325	-0.062942	0.220755	-0.722018
50%	-0.750689	-0.990019	0.298912	0.483195	-0.493995
75%	-0.243543	-0.565343	0.565214	0.613546	-0.279094
max	-0.124625	0.337268	0.621914	0.693037	-0.135573

Select a column (series)

```
1 In [20]: df.loc[dates[1]]
2 Out[20]:
3 A      -1.218197
4 B      -1.113826
5 C       0.546314
6 D      -0.255001
7 E      -0.135573
8 Name: 2016-12-18 00:00:00, dtype: float64
```


Select a range

```
1 In [21]: df.loc[:, ['A', 'C']]
2 Out[21]:
3           A          C
4 2016-12-11 -1.303610  0.621914
5 2016-12-18 -1.218197  0.546314
6 2016-12-25 -0.124625 -0.406295
7 2017-01-01 -0.283182  0.051509
```

Boolean selection criteria

```
1 In [23]: df[df.D > 0]
```

```
2 Out[23]:
```

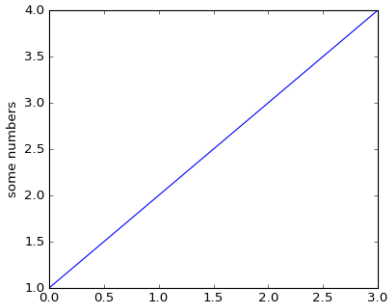
	A	B	C	D	E
2016-12-11	-1.303610	-1.235823	0.621914	0.379340	-0.326934
2016-12-25	-0.124625	0.337268	-0.406295	0.587049	-0.904906
2017-01-01	-0.283182	-0.866213	0.051509	0.693037	-0.661055

Recommended [http://www.gregreda.com/2013/10/26/
intro-to-pandas-data-structures/](http://www.gregreda.com/2013/10/26/intro-to-pandas-data-structures/)

Plotting

Draw a line

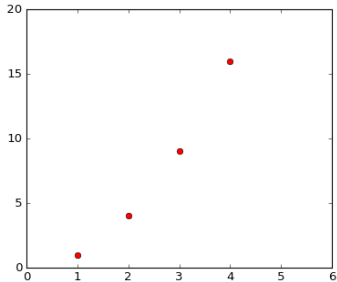
```
1 import matplotlib.pyplot as plt
2 plt.plot([1,2,3,4])
3 plt.ylabel('some numbers')
4 plt.show()
```



Plotting

Draw a line

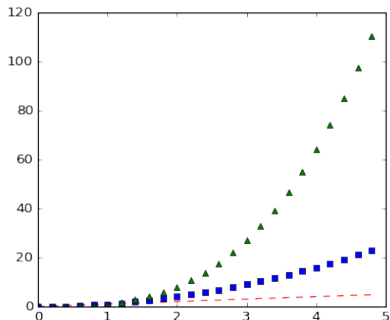
```
1 import matplotlib.pyplot as plt
2 plt.plot([1, 2, 3, 4], [1, 4, 9, 16])
3 plt.ylabel('some numbers')
4 plt.show()
```



Plotting

Draw a line

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 t = np.arange(0., 5., 0.2)
4 # r-- red dashes
5 # bs blue squares
6 # g^ green triangles
7 plt.plot(t, t,
8          'r--', t,
9          t**2, 'bs',
10         t, t**3, 'g^')
11 plt.show()
```

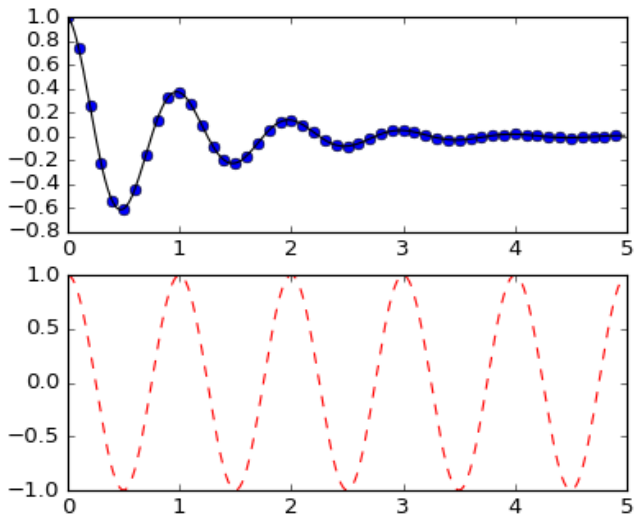


Plotting

Draw two curves

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 def f(t):
5     return np.exp(-t) * np.cos(2*np.pi*t)
6
7 t1 = np.arange(0.0, 5.0, 0.1)
8 t2 = np.arange(0.0, 5.0, 0.02)
9
10 plt.figure(1)
11 plt.subplot(211)
12 plt.plot(t1, f(t1), 'bo', t2, f(t2), 'k')
13
14 plt.subplot(212)
15 plt.plot(t2, np.cos(2*np.pi*t2), 'r--')
```

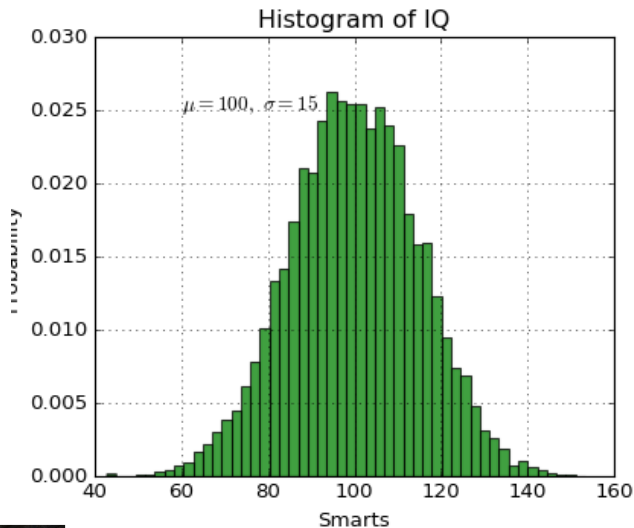
Plotting



Plotting

Histogram

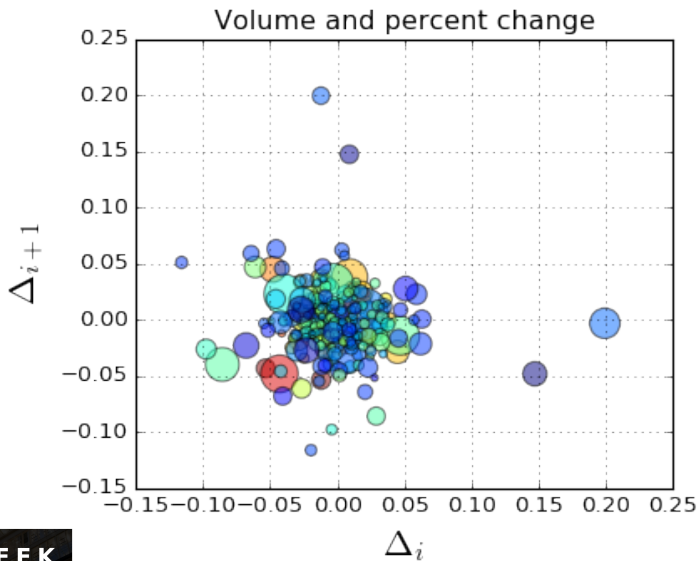
```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 mu, sigma = 100, 15
5 x = mu + sigma * np.random.randn(10000)
6
7 n, bins, patches = plt.hist(x, 50, normed=1, facecolor='g', alpha=0.75)
8
9 plt.xlabel('Smarts')
10 plt.ylabel('Probability')
11 plt.title('Histogram of IQ')
12 plt.text(60, .025, r'$\mu=100,\ \sigma=15$')
13 plt.axis([40, 160, 0, 0.03])
14 plt.grid(True)
15 plt.show()
```



Plotting

Scatter plot

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 fig, ax = plt.subplots()
5 ax.scatter(delta1[:-1], delta1[1:], c=close, s=volume, alpha=0.5)
6
7 ax.set_xlabel(r'$\Delta_i$', fontsize=20)
8 ax.set_ylabel(r'$\Delta_{i+1}$', fontsize=20)
9 ax.set_title('Volume and percent change')
10
11 ax.grid(True)
12 fig.tight_layout()
13
14 plt.show()
```



http://matplotlib.org/users/pyplot_tutorial.html

<http://matplotlib.org/users/beginner.html>

Avez-vous des questions ?