

Data Visualization in Python

Árpád Horváth

October 8, 2018

Vázlat

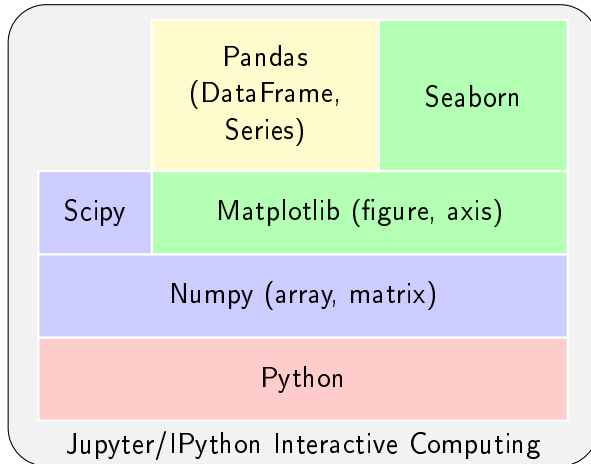
1 Initial settings and imports

2 Examples

- Matplotlib
- Pandas and Seaborn
- Further reading

Packages for Data Visualization

Python Scientific Environment



- Pylab =
Numpy +
Matplotlib
≈ MATLAB
- Web-centric
Visualisation
Libraries:
bokeh, plotly

Vázlat

1 Initial settings and imports

2 Examples

- Matplotlib
- Pandas and Seaborn
- Further reading

Jupyter notebook vs script

Jupyter notebook

- We need a web browser.
- Set a password (bash commands).
- Easy to run the parts several times with small modifications.
- Titles and texts can be added easily (LaTeX equations as well).
- Easy to share (as is, as HTML, PDF...).
- We can restart the kernel and run all the cells.

Script

- Easy to structure (modules, packages).
- Doesn't need Jupyter to install.
- IDE-s with more capabilities (version control, renaming).

ms.version

Imports used here

```
from matplotlib import pyplot as plt
import numpy as np
import pandas as pd
from pandas import DataFrame, Series
import seaborn as sns
```

In interactive sessions, instead of the first two lines:

```
from pylab import *
```

Do not use in scripts!

Vázlat

1 Initial settings and imports

2 Examples

- Matplotlib
- Pandas and Seaborn
- Further reading

Vázlat

1 Initial settings and imports

2 Examples

- Matplotlib
- Pandas and Seaborn
- Further reading

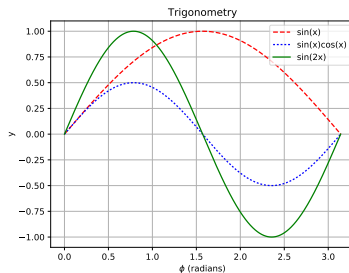
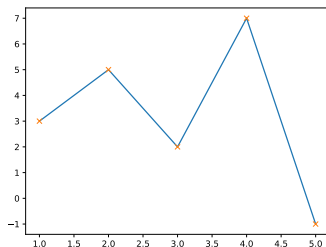
First plots with matplotlib

```
x = [1, 2, 3, 4, 5]
y = [3, 5, 2, 7, -1]
plt.plot(x, y)
plt.plot(x, y, 'x')
plt.savefig('first.pdf')
plt.close()
```

```
x = np.linspace(0, np.pi, 777)
y = np.sin(x)
plt.plot(x, np.sin(x), 'r--', label='sin(x)')
plt.plot(x, np.sin(x)*np.cos(x), 'b:', label='sin(x)cos(x)')
plt.plot(x, np.sin(2*x), 'g-', label='sin(2x)')
plt.legend(loc='upper right')
plt.title('Trigonometry')
plt.xlabel('$\phi$ (radians)')
plt.ylabel('y')
plt.grid(True)
```

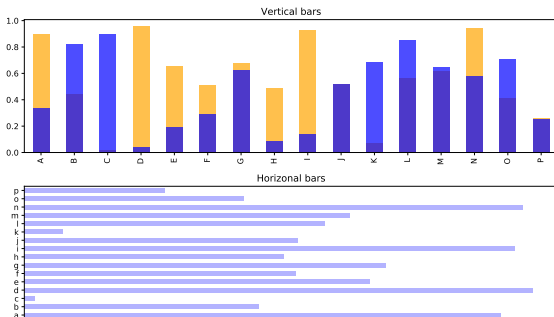
No needs of `np.` and `plt.` with `from pylab import *` or
`from numpy import linspace, pi`
`from matplotlib.pyplot import (plot, legend, savefig,`
`close, xlabel, ylabel, title, grid)`

The results



Bar plots and colors, tight layout

```
fig, axes = plt.subplots(2, 1)
data = pd.Series(np.random.rand(16), index=list('ABCDEFGHGIJKLMNOP'))
data2 = pd.Series(np.random.rand(16), index=list('abcdefghijklmnop'))
data2.plot.bar(ax=axes[0], color='orange', alpha=0.7)
data.plot.bar(ax=axes[0], color='b', alpha=0.7)
data2.plot.barh(ax=axes[1], color='#0000ff', alpha=0.3)
axes[0].set_title('Vertical bars')
axes[1].set_title('Horizontal bars')
fig.tight_layout()
```



Vázlat

1 Initial settings and imports

2 Examples

- Matplotlib
- **Pandas and Seaborn**
- Further reading

Vázlat

1 Initial settings and imports

2 Examples

- Matplotlib
- Pandas and Seaborn
- Further reading