

Wissenschaftliches Publizieren mit Python

Pycon DE 2013, Köln

Yves Hilpisch

Continuum Analytics Europe GmbH

yves@continuum.io

@dyjh

Agenda

- Fortschritt und Sprache
- Python Ecosystem
- Python Tools
- Case Study
- Open, Collaborative Research

Fortschritt und Sprache

Fortschritt im Allgemeinen

- **Effizienz:** Etwas Gegebenes besser machen

100m schneller laufen.

- **Effektivität:** Mehr machen mit dem Gegebenen

Den Speer weiter werfen.

- **Qualität:** Neues machen, schaffen

Alleine um die Welt segeln

(1895–1898 Joshua Slocum, erste Alleinumsegelung der Erde.)

Wissenschaftlicher Fortschritt

*„Scientific progress is the idea that science increases its **problem-solving ability** through the application of the scientific method.“*

http://en.wikipedia.org/wiki/Scientific_progress

Wissenschaftliche Methode

*„The scientific method is a body of techniques [**and tools**] for investigating phenomena, acquiring new knowledge, or correcting and integrating previous knowledge. To be termed scientific, a method of inquiry must be based on empirical and measurable evidence subject to specific principles of reasoning.“*

http://en.wikipedia.org/wiki/Scientific_method

Die Rolle der Sprache

"It is argued that language plays an active role in the development of scientific ideas. ...

And is mathematics somehow more or less than a language? ...

A particular characteristic of mathematics which appears in one aspect to differentiate it from language is its appeal to visual thinking."

Alan Ford and F. David Peat (1988): "The Role of Language in Science."

Sprachen in der Wissenschaft

- **Englisch** – geschriebene/gesprochene Sprache
- **Mathematik** – Symbolsprache, Logik
- **Code** – „Pseudo-Code“, Computersprache

Code + Struktur + Highlighting/Coloring

Python Ecosystem

Python Ecosystem

- **Python:** Basissystem bzw. Interpreter
- **Scientific Stack:** NumPy, SciPy, pandas, Cython, matplotlib, scikit-learn
- **Tools:** IPython, Spyder, Sphinx, Latex

Python spielt vierfache Rolle

- **Python:** Programmiersprache zur Erarbeitung von Forschungsergebnissen
- **Kommunikation:** Python zur Kommunikation von z.B. Algorithmen
- **Dokumentation:** Python zur Dokumentation von Algorithmen und Darstellung von Ergebnissen
- **Publikation:** Tools zur Publikation

Python Tools

Python Tools

- **Spyder**: Integrierte Entwicklungsumgebung
- **IPython**: Python's Killer App, insbesondere die Notebook Variante
- **Sphinx**: Dokumentationsumgebung und flexibles Autorensystem
- **PythonTEX**: Python für den Latex-Compiler

Spyder

- Code Editor mit Highlighting
- Code Checking, z.B. für PEP8
- Debugging
- Object und Variable Inspector
- Python CLI
- IPython Integration

Spyder

Editor - /Users/yhilpisch/Documents/Work/1_Python/3_LSM/LS2001_Table_1_Bench.py

```
1#
2# Least-Squares Monte Carlo for American Put
3# with Antithetic Paths and Moment Matching
4# Using In-The-Money Paths Only for Regression
5# (c) Visixion GmbH -- For Illustration Purposes Only.
6#
7from pylab import *
8from time import time
9import warnings
10warnings.simplefilter('ignore', np.RankWarning)
11t0=time()
12seed(100000)
13
14#
15# Parameters
16#
17# Option Parameters
18s0 = [36., 38., 40., 42., 44.] # Initial Index Levels
19vol = [0.2, 0.4] # Constant Volatilities
20tL = [1.0, 2.0] # Times-to-maturity
21K = 40. # Strike Price
22r = 0.06 # Risk-Free Short Rate
23
24# Simulation Parameters
25M = 50 # Time Steps p.a.
26l = 10000 # Simulation Paths
27
28# Variance Reduction Techniques
29antiPaths = True # Antithetic Paths
30noMatch = True # Moment Matching
31
32# Benchmark Values
33bl = (4.478, 4.840, 7.101, 8.508,
34      3.250, 3.745, 6.148, 7.670,
35      2.314, 2.885, 5.312, 6.920,
36      1.617, 2.212, 4.582, 6.248,
```

Variable explorer

Name	Type	Size	Value
C	float64	(10000,)	array([0., 0., 0., ..., 0., 0., 0.])
l	int	1	10000
K	float	1	40.0
M	int	1	50
M0	int	1	50
S	float64	(51, 10000)	array([[1., 1., 1., ..., 1., 1., 1.], [0., 0., 0., ..., 0., 0., 0.]])
S0	float	1	44.0
T	float	1	1.0
V	float64	(51, 10000)	array([[0., 0., 0., ..., 0., 0., 0.], [0., 0., 0., ..., 0., 0., 0.]])
V0	float64	1	6.2615727209650602
aerr	float64	1	0.21957468241668021
antiPaths	bool	1	True
bl	tuple	20	<tuple @ 0x104F02128>

Object inspector Variable explorer File explorer

Console

Python 1 00:08:50

```
Python 2.7.5 [Anaconda 1.7.0 (x86_64)] (default, Jun 28 2013, 22:20:13)
[GCC 4.0.1 (Apple Inc. build 5493)] on darwin
Type "help", "copyright", "credits" or "license" for more information.

Imported NumPy 1.7.1, SciPy 0.12.0, Matplotlib 1.3.0
Type "scientific" for more details.
>>> runfile('/Users/yhilpisch/Documents/Work/1_Python/3_LSM/LS2001_Table_1_Bench.py', wdir=r'/Users/yhilpisch/Documents/Work/1_Python/3_LSM')
S0 36 VOL 0.20 T 1 Value 4.493 Benchmark 4.478 Abs 0.015 Rel 0.337
S0 36 VOL 0.20 T 2 Value 4.832 Benchmark 4.840 Abs -0.008 Rel -0.155
S0 36 VOL 0.40 T 1 Value 7.096 Benchmark 7.101 Abs -0.005 Rel -0.067
S0 36 VOL 0.40 T 2 Value 8.502 Benchmark 8.508 Abs -0.006 Rel -0.070
S0 38 VOL 0.20 T 1 Value 3.268 Benchmark 3.250 Abs 0.018 Rel 0.567
S0 38 VOL 0.20 T 2 Value 3.757 Benchmark 3.745 Abs 0.012 Rel 0.327
S0 38 VOL 0.40 T 1 Value 6.174 Benchmark 6.148 Abs 0.026 Rel 0.416
S0 38 VOL 0.40 T 2 Value 7.667 Benchmark 7.670 Abs -0.003 Rel -0.039
S0 40 VOL 0.20 T 1 Value 2.343 Benchmark 2.314 Abs 0.029 Rel 1.266
S0 40 VOL 0.20 T 2 Value 2.900 Benchmark 2.885 Abs 0.015 Rel 0.508
S0 40 VOL 0.40 T 1 Value 5.342 Benchmark 5.312 Abs 0.030 Rel 0.567
S0 40 VOL 0.40 T 2 Value 6.925 Benchmark 6.920 Abs 0.005 Rel 0.072
```

IPython

- Interaktive Python Shell
- Vielzahl von „%magic“ Funktionen
- Command Line Historie
- Shell, QT version und Notebook (Browser)
- Parallele Programmausführung (ipcluster)
- viele Konvertierungsmöglichkeiten

IPython Shell

```
bash python bash python python python
In [3]: exit
Yvess-MacBook-Air:Sphinx yhilpisch$ ipython --pylab
Python 2.7.5 |Anaconda 1.7.0 (x86_64)| (default, Jun 28 2013, 22:20:13)
Type "copyright", "credits" or "license" for more information.

IPython 1.0.0 -- An enhanced Interactive Python.
?      -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help    -> Python's own help system.
object? -> Details about 'object', use 'object??' for extra details.
Using matplotlib backend: MacOSX

In [1]: a = linspace(0, 100, 33)

In [2]: a
Out[2]:
array([ 0.    ,  3.125,  6.25 ,  9.375, 12.5 , 15.625,
        18.75 , 21.875, 25.    , 28.125, 31.25 , 34.375,
        37.5 , 40.625, 43.75 , 46.875, 50.    , 53.125,
        56.25 , 59.375, 62.5 , 65.625, 68.75 , 71.875,
        75.    , 78.125, 81.25 , 84.375, 87.5 , 90.625,
        93.75 , 96.875, 100.   ])

In [3]: a ** 2
Out[3]:
array([ 0.00000000e+00,  9.76562500e+00,  3.90625000e+01,
        8.78906250e+01,  1.56250000e+02,  2.44140625e+02,
        3.51562500e+02,  4.78515625e+02,  6.25000000e+02,
        7.91015625e+02,  9.76562500e+02,  1.18164062e+03,
        1.40625000e+03,  1.65039062e+03,  1.91406250e+03,
        2.19726562e+03,  2.50000000e+03,  2.82226562e+03,
        3.16406250e+03,  3.52539062e+03,  3.90625000e+03,
        4.30664062e+03,  4.72656250e+03,  5.16601562e+03,
        5.62500000e+03,  6.10351562e+03,  6.60156250e+03,
        7.11914062e+03,  7.65625000e+03,  8.21289062e+03,
        8.78906250e+03,  9.38476562e+03,  1.00000000e+04])

In [4]: a[3:10]
Out[4]: array([ 9.375, 12.5 , 15.625, 18.75 , 21.875, 25.    , 28.125])

In [5]: █

In [2]: █

In [4]: array([ 0.00000000e+00,  9.76562500e+00,  3.90625000e+01,
        8.78906250e+01,  1.56250000e+02,  2.44140625e+02,
        3.51562500e+02,  4.78515625e+02,  6.25000000e+02,
        7.91015625e+02,  9.76562500e+02,  1.18164062e+03,
        1.40625000e+03,  1.65039062e+03,  1.91406250e+03,
        2.19726562e+03,  2.50000000e+03,  2.82226562e+03,
        3.16406250e+03,  3.52539062e+03,  3.90625000e+03,
        4.30664062e+03,  4.72656250e+03,  5.16601562e+03,
        5.62500000e+03,  6.10351562e+03,  6.60156250e+03,
        7.11914062e+03,  7.65625000e+03,  8.21289062e+03,
        8.78906250e+03,  9.38476562e+03,  1.00000000e+04])

In [4]: a[3:10]
Out[4]: array([ 9.375, 12.5 , 15.625, 18.75 , 21.875, 25.    , 28.125])
```

IPython QT Console

```
Python 2.7.5 |Anaconda 1.7.0 (x86_64)| (default, Jun 28 2013, 22:20:13)  
Type "copyright", "credits" or "license" for more information.
```

```
IPython 1.0.0 -- An enhanced Interactive Python.
```

```
?          -> Introduction and overview of IPython's features.
```

```
%quickref  -> Quick reference.
```

```
help       -> Python's own help system.
```

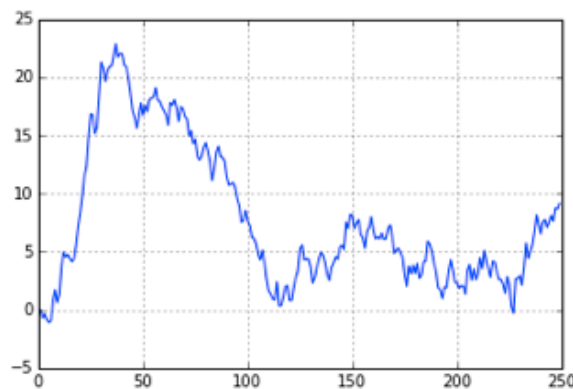
```
object?    -> Details about 'object', use 'object??' for extra details.
```

```
%gui       -> A brief reference about the graphical user interface.
```

```
In [1]: a = standard_normal(250)
```

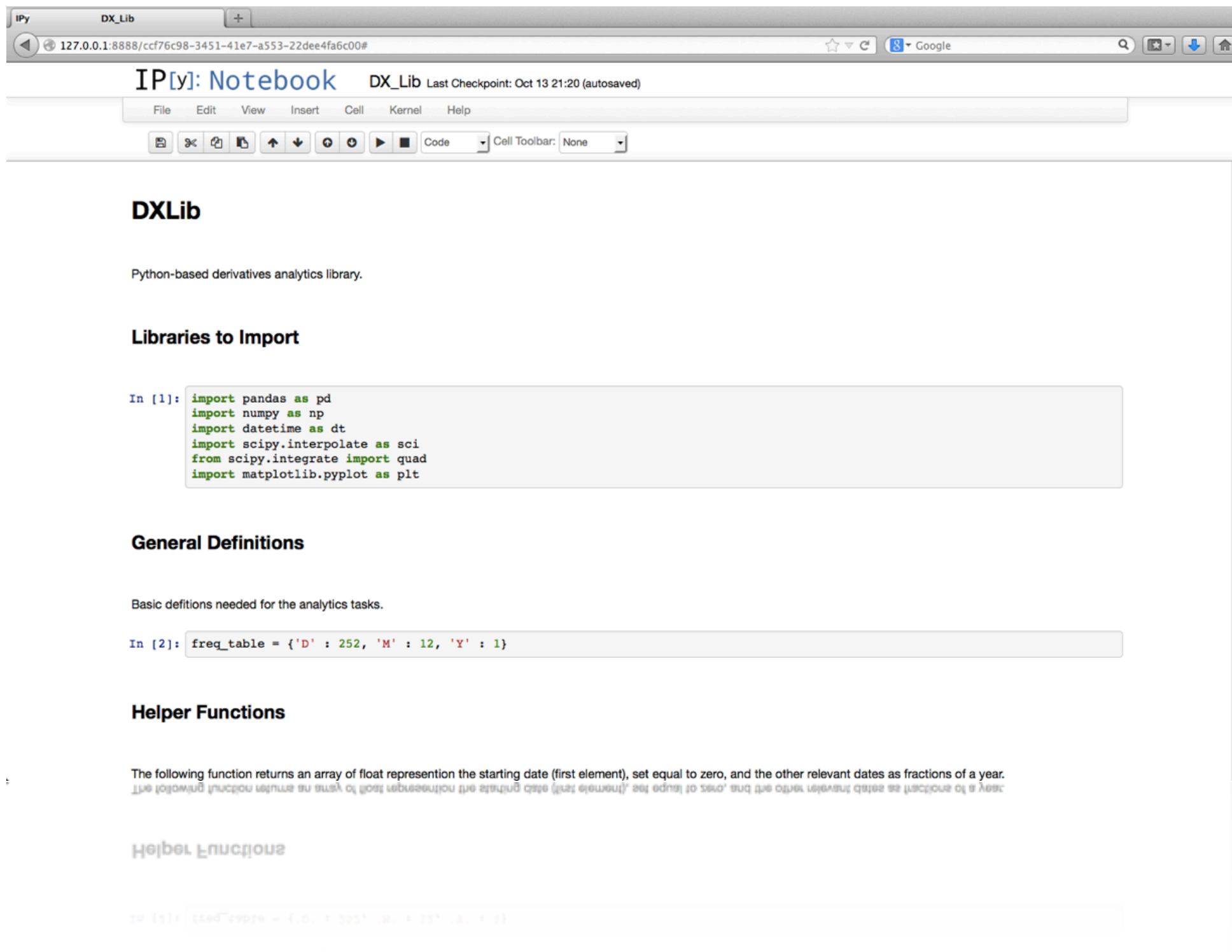
```
In [2]: c = cumsum(a)
```

```
In [3]: figure()  
...: plot(c)  
...: grid(True)
```



```
In [4]: figure()  
...: hist(a)  
...: grid(True)
```


IPython Notebook



The screenshot shows a web browser window with the address bar displaying `127.0.0.1:8888/ccf76c98-3451-41e7-a553-22dee4fa6c00#`. The browser's search bar contains the text "Google". The notebook interface has a title bar "IP[y]: Notebook DX_Lib" and a status bar "Last Checkpoint: Oct 13 21:20 (autosaved)". Below the title bar is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". A toolbar contains icons for saving, opening, and running cells, along with a "Code" dropdown and a "Cell Toolbar" dropdown set to "None".

DXLib

Python-based derivatives analytics library.

Libraries to Import

```
In [1]: import pandas as pd
import numpy as np
import datetime as dt
import scipy.interpolate as sci
from scipy.integrate import quad
import matplotlib.pyplot as plt
```

General Definitions

Basic definitions needed for the analytics tasks.

```
In [2]: freq_table = {'D' : 252, 'M' : 12, 'Y' : 1}
```

Helper Functions

The following function returns an array of float representation the starting date (first element), set equal to zero, and the other relevant dates as fractions of a year.

Helper Functions

```
In [3]: freq_table = {'D' : 252, 'M' : 12, 'Y' : 1}
```

Sphinx

- Umgebung zur Dokumentation von Python-Projekten und vielem mehr ...
- Syntax basiert i.V. auf reStructuredText (.rst)
- Source Code comilierbar in verschiedene Formate (HTML, Latex/PDF)
- Gute Erweiterbarkeit (extensions) und Anpassbarkeit (CSS)
- Integration mit IPython (über extension)

PythonTEX – Source Code

```
\section{Plots with matplotlib}

1 \section{Plots with matplotlib}
2
3 We can create plots with matplotlib, perfectly matching the plot fonts with
  the document fonts. No more searching for the code that created a figure!
4
5 You may want to use matplotlib's PGF backend when creating plots.
6
7 \begin{pylabblock}
8 rc('text', usetex=True)
9 rc('font', family='serif')
10 rc('font', size=10.0)
11 rc('legend', fontsize=10.0)
12 rc('font', weight='normal')
13 x = linspace(0, 10)
14 figure(figsize=(4, 2.5))
15 plot(x, sin(x), label='$\sin(x)$')
16 xlabel(r'$x\mathrm{-axis}$')
17 ylabel(r'$y\mathrm{-axis}$')
18 legend(loc='lower right')
19 savefig('myplot.pdf', bbox_inches='tight')
20 \end{pylabblock}
21
22 \begin{center}
23 \includegraphics{myplot.pdf}
24 \end{center}
```

```
\set /guq{cgu{cL}}
\set /tuc{naed{sbu{c{w\brof*bq{}
```



CONTINUUM
ANALYTICS

PythonTEX – Output

```
phi = Symbol(r'\phi')  
h = Integral(exp(-phi**2), (phi, 0, oo))
```

$$\int_0^{\infty} e^{-\phi^2} d\phi = \frac{1}{2}\sqrt{\pi}$$

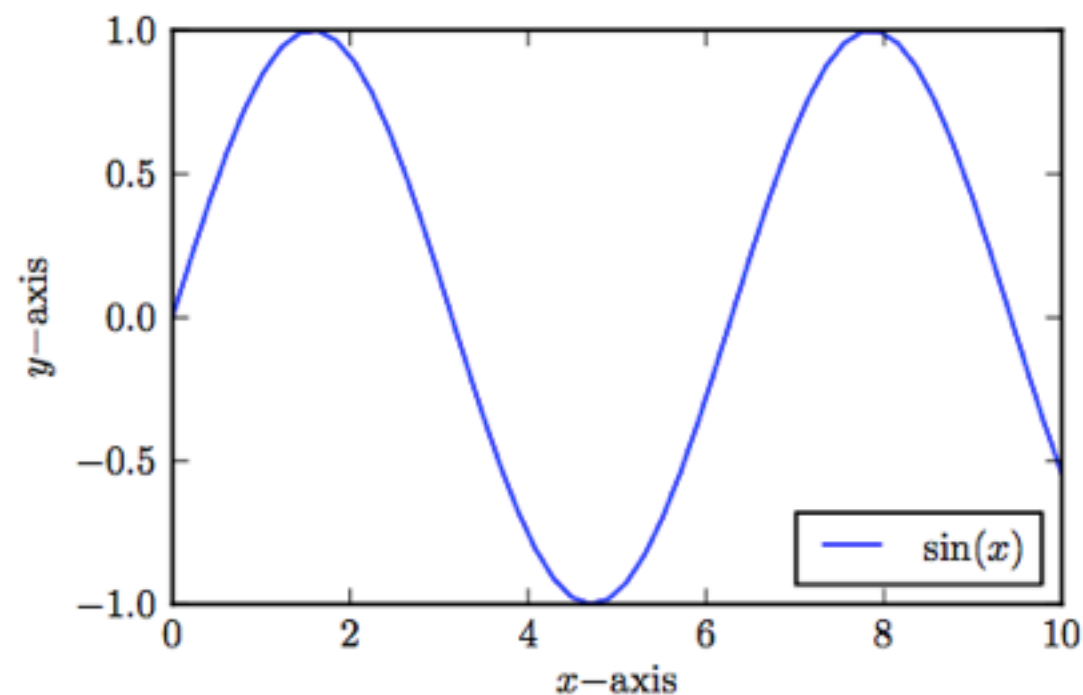
5 Plots with matplotlib

We can create plots with matplotlib, perfectly matching the plot fonts with the document fonts. No more searching for the code that created a figure!

You may want to use matplotlib's PGF backend when creating plots.

```
rc('text', usetex=True)  
rc('font', family='serif')  
rc('font', size=10.0)  
rc('legend', fontsize=10.0)  
rc('font', weight='normal')  
x = linspace(0, 10)  
figure(figsize=(4, 2.5))  
plot(x, sin(x), label='$\sin(x)$')  
xlabel(r'$x\mathrm{-axis}$')  
ylabel(r'$y\mathrm{-axis}$')  
legend(loc='lower right')  
savefig('myplot.pdf', bbox_inches='tight')
```

PythonTEX – Output



6 Basic pylab interaction

```
from scipy.integrate import quad  
myintegral = quad(lambda x: e**-x**2, 0, inf)[0]
```

$$\int_0^{\infty} e^{-x^2} dx = 0.886226925453$$

$$\int_{-\infty}^0 e^{-x^2} dx = 0.886226925453$$

Case Study

Case Study

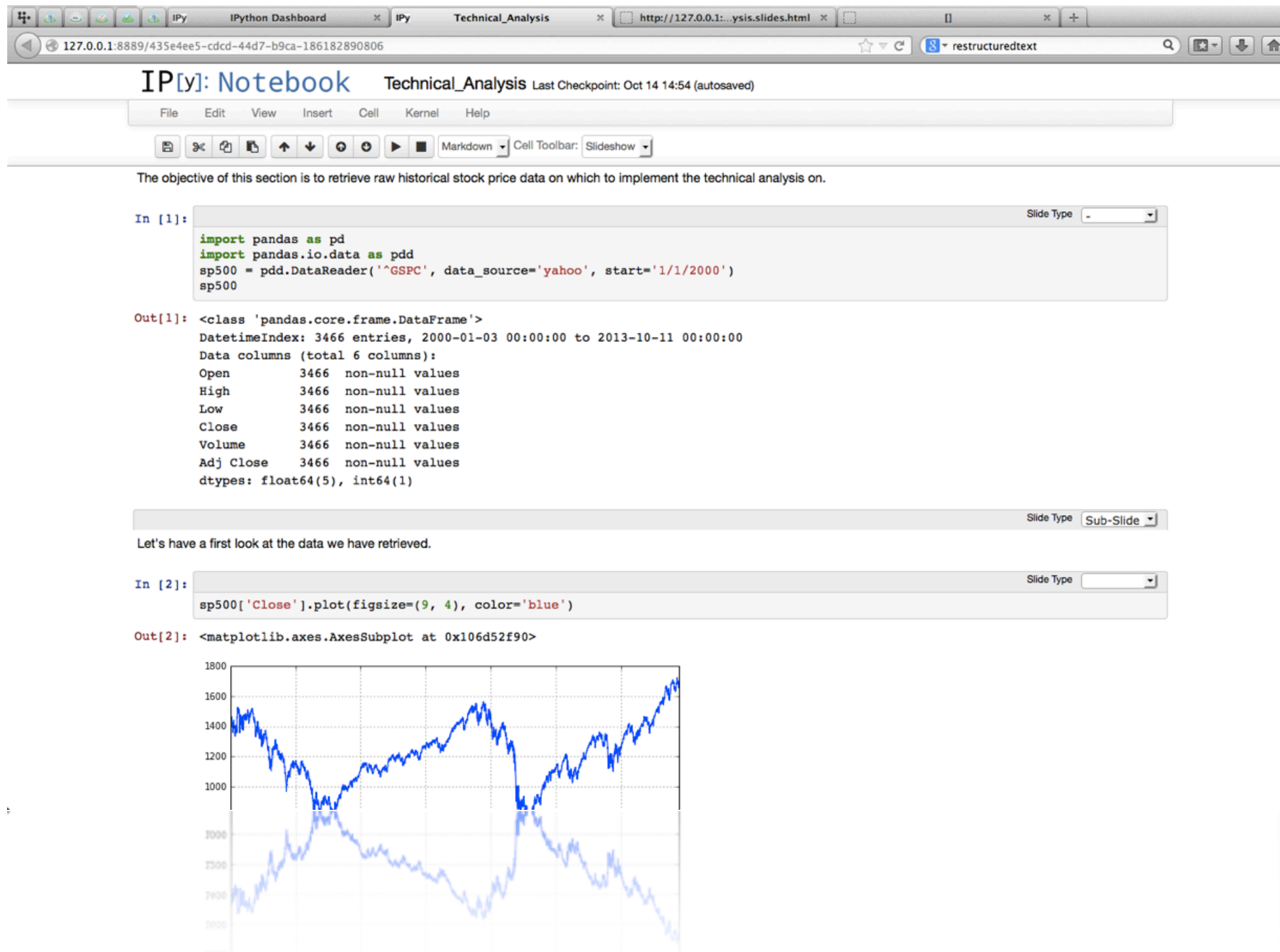
Einfaches Beispiel aus dem Finanzbereich, das

- Text
- Struktur
- Code
- Graphiken
- Formeln

aufweist.

IPython Notebook Umsetzung

Das IPython Notebook (ipynb)



IPython Notebook Markdown

Conclusions

Slide Type Sub-Slide ▾

IPython is an excellent environment to not only document Python projects and code. It is rather a great tool for interactively developing professional and scientific publications. It allows to generate output in such standard formats as:

- * **HTML**, i.e. a Web page
- * **PDF**, i.e. via Latex
- * **ePub**, i.e. for electronic books

Since Python code and its results are already included in the Notebook, it saves a lot of the typical copy & paste tasks.

In addition, it is also quite easy to include Latex code/formulas directly into the document.

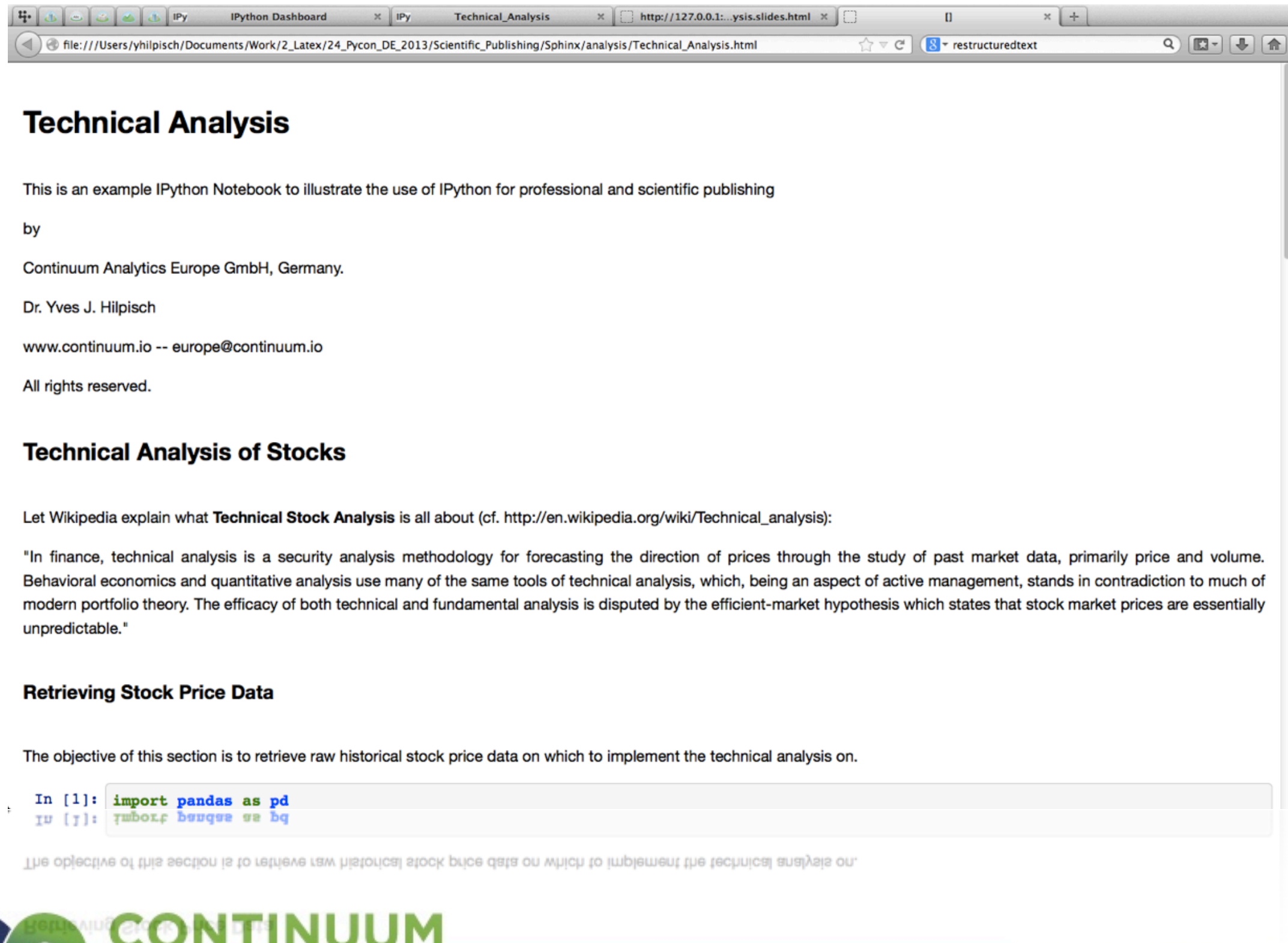
$$dS_t = r S_t dt + \sigma S_t dZ_t$$

IPython seems the right choice when it comes to documents that include

- * English
- * Mathematics
- * Python

- * Ελληνικά
- * Математика

IPython HTML output im Browser



Technical Analysis

This is an example IPython Notebook to illustrate the use of IPython for professional and scientific publishing
by
Continuum Analytics Europe GmbH, Germany.
Dr. Yves J. Hilpisch
www.continuum.io -- europe@continuum.io
All rights reserved.

Technical Analysis of Stocks

Let Wikipedia explain what **Technical Stock Analysis** is all about (cf. http://en.wikipedia.org/wiki/Technical_analysis):

"In finance, technical analysis is a security analysis methodology for forecasting the direction of prices through the study of past market data, primarily price and volume. Behavioral economics and quantitative analysis use many of the same tools of technical analysis, which, being an aspect of active management, stands in contradiction to much of modern portfolio theory. The efficacy of both technical and fundamental analysis is disputed by the efficient-market hypothesis which states that stock market prices are essentially unpredictable."

Retrieving Stock Price Data

The objective of this section is to retrieve raw historical stock price data on which to implement the technical analysis on.

```
In [1]: import pandas as pd
In [2]: import numpy as np
```

The objective of this section is to retrieve raw historical stock price data on which to implement the technical analysis on.

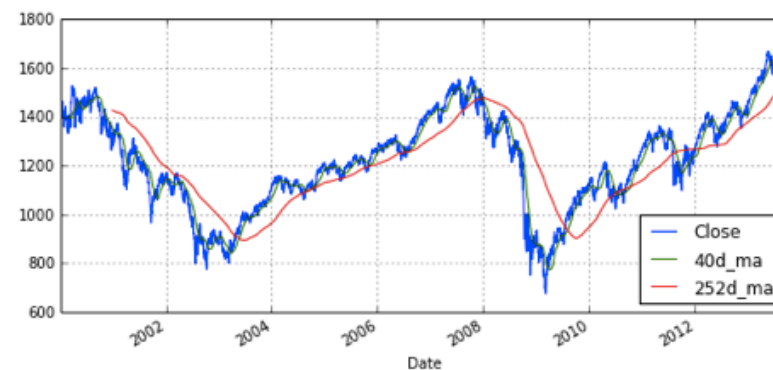
IPython HTML5 Slides (mit reveal.js)



pandas makes it very convenient to operate on time series data. Moving averages are easily added.

```
In [3]: sp500['40d_ma'] = pd.rolling_mean(sp500['Close'], window=40)
        sp500['252d_ma'] = pd.rolling_mean(sp500['Close'], window=252)
        sp500[['Close', '40d_ma', '252d_ma']].plot(figsize=(9, 4))
```

Out[3]: <matplotlib.axes.AxesSubplot at 0x10598a610>



IPython Latex und PDF Output

The screenshot displays the TeXworks application interface. The left pane shows the LaTeX source code for a document titled 'Technical_Analysis.tex'. The code includes a header section with a title 'Technical Analysis' and author 'Yves Hilpisch', a date 'October 14, 2013', and a brief description of the document's purpose. It also includes a section '1 Technical Analysis of Stocks' and a subsection '1.1 Retrieving Stock Price Data'. The code uses LaTeX commands like `\end{ColorVerbatim}`, `\makebox`, `\hspace`, `\vspace`, `\begin{InvisibleVerbatim}`, `\begin{alltt}`, `\end{alltt}`, `\end{InvisibleVerbatim}`, and `\begin{ColorVerbatim}` to format the document. The right pane shows the resulting PDF document, which is titled 'Technical_Analysis.pdf'. The PDF content matches the LaTeX source code, showing the title, author, date, and the beginning of the '1 Technical Analysis of Stocks' section. The status bar at the bottom of the TeXworks window indicates 'Zeile 237 von 456; Spalte 8' and 'Seite 1 von 3'.

```
\end{ColorVerbatim}

% If the first block is an image, minipage the image. Else
% request a certain amount of space for the input text.
\nneedspace{4\baselineskip}

% Add document contents.

\makebox[0.1\linewidth]{\smaller\hfill\textcolor{nbframe-out-prompt}Out\hspace{4pt}{[1]}}:
\hspace{4pt}}\}*
\vspace{-2.55\baselineskip}\begin{InvisibleVerbatim}
\vspace{-0.5\baselineskip}
\begin{alltt}<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 3466 entries, 2000-01-03 00:00:00 to 2013-10-11
00:00:00
Data columns (total 6 columns):
Open      3466 non-null values
High      3466 non-null values
Low       3466 non-null values
Close     3466 non-null values
Volume    3466 non-null values
Adj Close  3466 non-null values
dtypes: float64(5), int64(1)\end{alltt}

\end{InvisibleVerbatim}

Let's have a first look at the data we have retrieved.

% Make sure that atleast 4 lines are below the HR
\nneedspace{4\baselineskip}

\vspace{6pt}
\makebox[0.1\linewidth]{\smaller\hfill\textcolor{nbframe-in-prompt}In\hspace{4pt}{[2]}}:
\hspace{4pt}}\}*
\vspace{-2.65\baselineskip}
\begin{ColorVerbatim}
```

Sphinx Umsetzung

Sphinx Source Code

```
technical_analysis.rst x conclusion.rst x index.rst x
23
24 =====
25 Retrieving Stock Price Data
26 =====
27
28 The objective of this section is to retrieve raw historical stock price data on which to implement the technical
29 analysis on.
30 .. ipython:: python
31
32     import pandas as pd
33
34     import pandas.io.data as pdd
35
36     sp500 = pdd.DataReader('^GSPC', data_source='yahoo', start='1/1/2000')
37
38     sp500
39
40
41 Let's have a first look at the data we have retrieved.
42
43 .. ipython:: python
44
45     # @savefig sp500.png
46     sp500['Close'].plot(figsize=(9, 4), color='blue')
47
48
49 :num: `Figure #figure1` shows the resulting figure.
50
51 .. _figure1:
52
53 .. figure:: sp500.png
54     :scale: 70 %
55     :align: center
56
57     The S&P 500 index since the beginning of 2000.
58
59
```

Line 43, Column 11

Spaces: 2

reStructuredText

Line 43, Column 11

Spaces: 2

reStructuredText

```
20
21
22 The S&P 500 index since the beginning of 2000.
```


Sphinx Source Code (Analytics)

```
29
30 .. ipython:: python
31
32     import pandas as pd
33
34     import pandas.io.data as pdd
35
36     sp500 = pdd.DataReader('^GSPC', data_source='yahoo', start='1/1/2000')
37
38     sp500
39
30
38     zb200
31
30     zb200 = pdd.DataReader('^DAX', data_source='yahoo', start='1/1/2000')
```

Sphinx Source Code (Graphics)

```
43 .. ipython:: python
44
45     # @savefig sp500.png
46     sp500['Close'].plot(figsize=(9, 4), color='blue')
47
48
49 :num: `Figure #figure1` shows the resulting figure.
50
51 .. _figure1:
52
53 .. figure:: sp500.png
54     :scale: 70 %
55     :align: center
56
57     The S&P 500 index since the beginning of 2000.
58
59
```

Line 43, Column 11

Line 43, Column 11

20

28

21

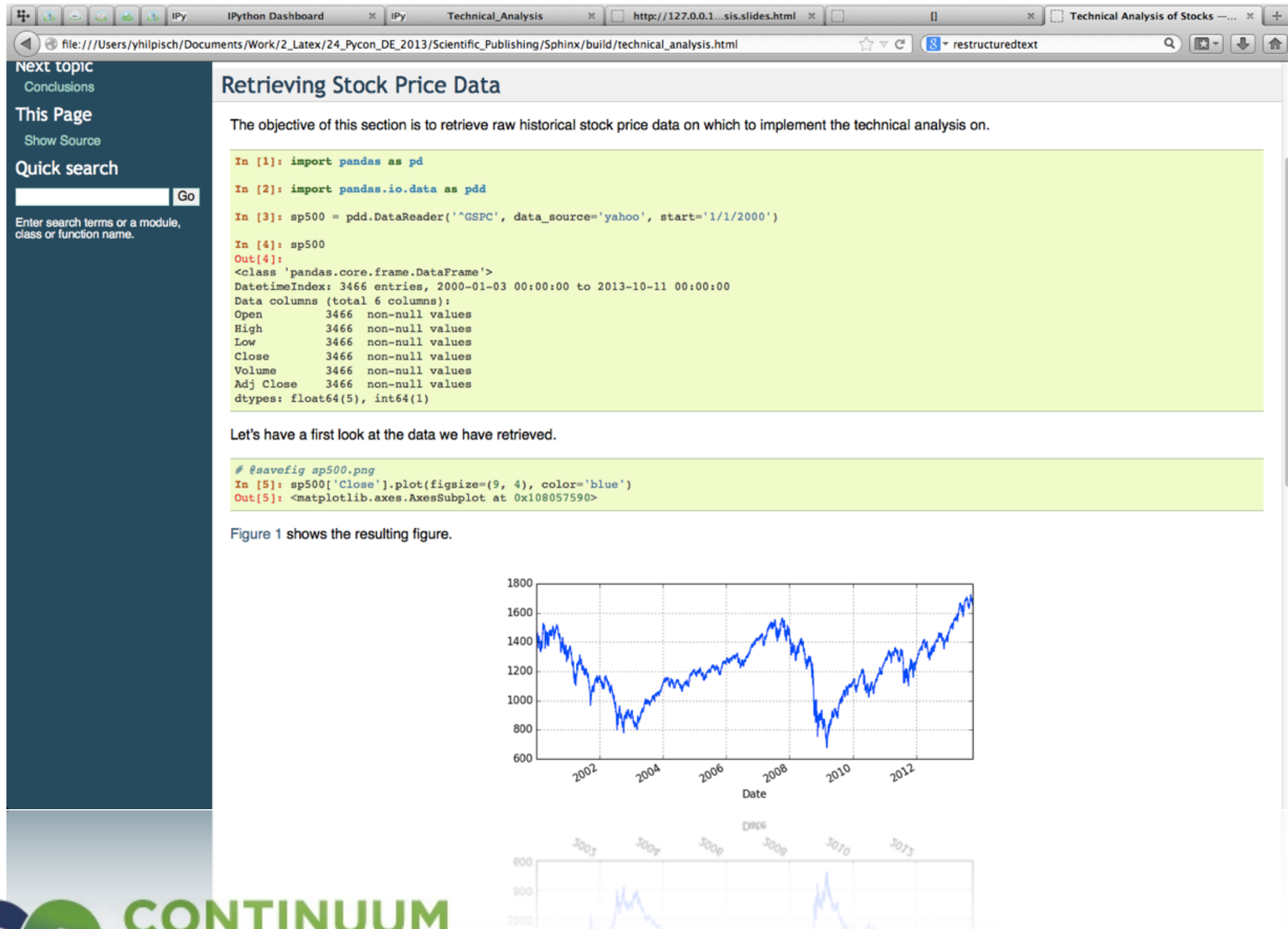
20

20

The S&P 500 index since the beginning of 2000.

continuum

Sphinx HTML Output



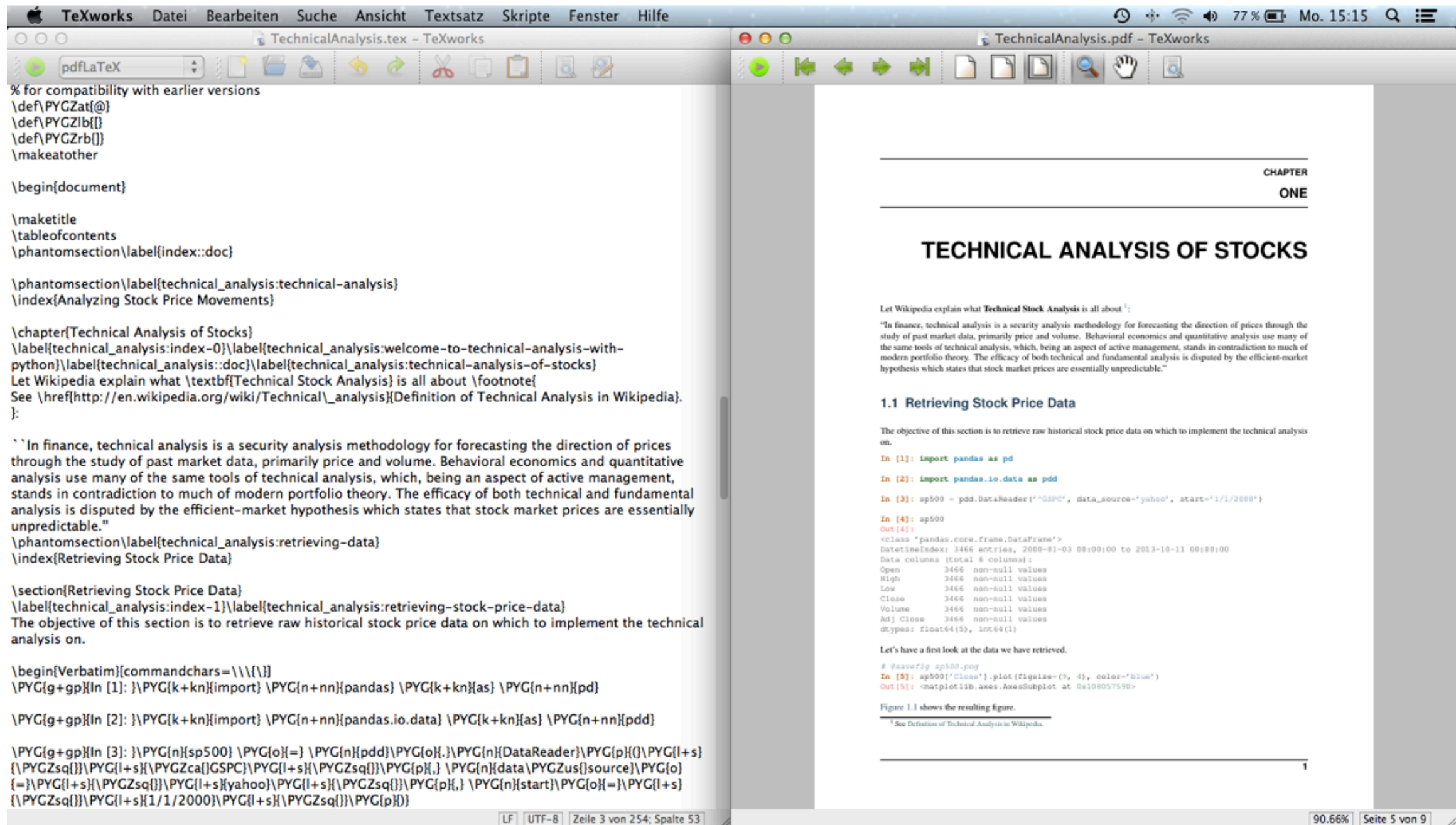
Sphinx Customized HTML Site

The screenshot shows a web browser window displaying a Sphinx-generated HTML site. The browser's address bar shows the URL `www.eurexchange.com/vstoxx/`. The site's header features the Eurex logo and a search bar. The main content area is titled "Welcome to the Expand VSTOXX Tutorials based on Python". On the left, there is a sidebar with "Overview Tutorials" and a list of topics: Python Preliminaries, Analyzing Historical VSTOXX Data, Calculating the VSTOXX Index, Valuing Volatility Options with GL96, Automated Monte Carlo Tests for GL96, Calibration of GL96 Model, and Backtesting of VSTOXX Strategies. The main content area lists the following topics:

- Python Preliminaries
 - The Python Universe
 - Related Topics
 - Recommended Readings
- Analyzing Historical VSTOXX Data
 - Creation of a Database
 - Retrieving Raw Data
 - Import Data into Python
 - The Data Frames
 - Saving in HDF5 Database
 - Data Analysis
 - Open Data Frame from the HDF5Store
 - Data Description
 - Computation of Changes
 - Calculation of Log>Returns
 - Correlation between EURO STOXX 50 and VSTOXX
 - Further Information
- Calculating the VSTOXX Index
 - Introduction
 - Collecting the Data
 - Background Information
 - Python Implementation
 - Further Information
 - Calculating the Sub-Indexes
 - The Algorithm
 - Implementation for a Single Index Value
 - Implementation for Time Series
 - Calculating VSTOXX
 - The Algorithm
 - The Implementation
 - Further Information
 - The Module `dateFunctions`
 - Further Information

The footer contains links for Disclaimer, Privacy policy, Imprint, Rules and Regulations, and Sitemap.

Sphinx Latex und PDF Output



Open, Collaborative Research

Open Research in Physik mit IPython

Das Forschungspapier auf arxiv.org

Submitted for publication in ApJ Letters
Preprint typeset using L^AT_EX style emulateapj v. 12/16/11

<http://arxiv.org/abs/1303.2690>

THE KINEMATICS OF THE LOCAL GROUP IN A COSMOLOGICAL CONTEXT

J. E. FORERO-ROMERO¹, Y. HOFFMAN², S. BUSTAMANTE³, S. GOTTLÖBER⁴, G. YEPES⁵

(Dated: March 10, 2013)

Submitted for publication in ApJ Letters

ABSTRACT

Recent observations constrained the tangential velocity of M31 with respect to the Milky Way (MW) to be $v_{\text{M31,tan}} < 34.4 \text{ km s}^{-1}$ and the radial velocity to be in the range $v_{\text{M31,rad}} = -109 \pm 4.4 \text{ km s}^{-1}$ (van der Marel et al. 2012). In this study we use a large volume high resolution N-body cosmological simulation (Bolshoi) together with three constrained simulations to statistically study this kinematics in the context of the Λ CDM. The comparison of the ensembles of simulated pairs with the observed LG at the $1\text{-}\sigma$ level in the uncertainties has been done with respect to the radial and tangential velocities, the reduced orbital energy (e_{tot}), angular momentum (l_{orb}) and the dimensionless spin parameter, λ . Our main results are: (i) the preferred radial and tangential velocities for pairs in Λ CDM are $v_r = -80 \pm 20 \text{ km s}^{-1}$, $v_t = 50 \pm 10 \text{ km s}^{-1}$, (ii) pairs around that region are 3 to 13 times more common than pairs within the observational values, (iii) 15% to 24% of LG-like pairs in Λ CDM have energy and angular momentum consistent with observations while (iv) 9% to 13% of pairs in the same sample show similar values in the inferred dimensionless spin parameter. It follows that within current observational uncertainties the quasi-conserved quantities that characterize the orbit of the LG, i.e. e_{tot} , l_{orb} and λ , do not challenge the standard Λ CDM model, but the model is in tension with regard to the actual values of the radial and tangential velocities. This might hint to a problem of the Λ CDM model to reproduce the observed LG.

Subject headings: galaxies: kinematics and dynamics, Local Group, methods:numerical

Subject headings: galaxies: kinematics and dynamics, Local Group, methods:numerical
model to reproduce the observed LG.

to the actual values of the radial and tangential velocities. This might hint to a problem of the Λ CDM
to the standard Λ CDM model, but the model is in tension with regard

Das Github Repository

The screenshot shows the GitHub repository page for 'forero / LG_Kinematics'. The repository is public and has 84 commits, 1 branch, 0 tags, and 1 contributor. The main branch is 'master'. The repository description is 'Quantify how special is the observed Local Group kinematics in the LCDM framework'. The file list shows 'code' (minor update, 4 months ago), 'data' (removed environment data, 5 months ago), 'paper' (removed referee reply, 4 months ago), and 'README.md' (Update README.md, 4 months ago). The README content includes the title 'LG_Kinematics', the description, and a section 'Replicating the results' with instructions to clone the repository, open the IPython notebook 'main_analysis.ipynb', and execute all cells. There is also a 'Paper' section with a link to the ArXiv link. The right sidebar shows options to clone the repository via HTTPS, Subversion, or other methods, and a 'Download ZIP' button.

GitHub This repository Search or type a command Explore Features Enterprise Blog Sign up Sign in

PUBLIC forero / LG_Kinematics ★Star 2 Fork 0

Quantify how special is the observed Local Group kinematics in the LCDM framework

84 commits 1 branch 0 tags 1 contributor

branch: master LG_Kinematics

Update README.md

forero authored 4 months ago latest commit d446ad0e4

code	minor update	4 months ago
data	removed environment data	5 months ago
paper	removed referee reply	4 months ago
README.md	Update README.md	4 months ago

README.md

LG_Kinematics

Quantify how special is the observed Local Group (Andromeda & Milky Way) kinematics in the LCDM framework

Replicating the results

- Clone the repository
- Enter code/ and open the IPython notebook main_analysis.ipynb
- Execute all the cells in the notebook.

Paper

The kinematics of the Local Group in a cosmological context - ArXiv link

HTTPS clone URL
https://github.com/forero/LG_Kinematics
You can clone with HTTPS, Subversion, and other methods.
Download ZIP

https://github.com/forero/LG_Kinematics

Das IPython Notebook

IP[y]: Notebook

main_analysis

Last saved: Jul 02 09:55

File Edit View Insert Cell Kernel Help

Markdown Cell Toolbar: None

The Kinematics of the Local Group in a Cosmological Context

This notebook contains all data analysis for the ApJ Letter *The Kinematics of the Local Group in a Cosmological Context* by Jaime E. Forero-Romero, Yehuda Hoffman, Sebastian Bustamante, Stefan Gottloeber and Gustavo Yepes.

Follow through the notebook to generate the figures and numbers used in the paper. You can also generate results that were mentioned but not explicitly reported. For instance, the results for pairs obtained for a Friend-of-Friends halo finder.

```
In [1]: %pylab inline
        %load_ext autoreload
        %autoreload 2
        from make2DHistogram import *
        from generateInfo import *
```

```
In [2]: # Global paths and constants
        data_path = "../data/"

        G_GRAV = 4.54E-48 #units of Mpc^3 Msun^-1 s^-2
        KM_TO_MPC = 3.2E-20
        HUBBLE = 0.70
        E_UNITS = 1.0E-36
        L_UNITS = 1.0
```

Collaborative Research mit Wakari.io

Wakari.io

Wakari ist eine vollständige Python Umgebung

- Linux Server inkl. Shell & File Manager
- Cloud- & Browser-basiert
- Python, Anaconda, IPython
- Packaging & Sharing von IPython Notebooks
- Skalierbar (Zuschalten von Nodes)
- Wakari Enterprise ab November 2013

Wakari.io als Python Umgebung

The screenshot displays the Wakari.io web interface in a browser. The URL bar shows <https://www.wakari.io/wakari>. The top navigation bar includes 'Wakari', 'Auto Shutdown', 'Add Compute Nodes', 'Share', 'New Notebook', 'View', 'Tools', 'Help', and 'yves'. The left sidebar shows the file path '~/.PyconDE' and a file named 'Technical_Analysis.ipynb'. The main area displays the Jupyter Notebook 'Technical_Analysis' with a menu bar (File, Edit, View, Insert, Cell, Kernel, Help) and a toolbar. The notebook content includes a title 'Technical Analysis', a paragraph about IPython, and a section 'Technical Analysis of Stocks' with a Wikipedia link. The right sidebar shows a terminal window with IPython and pylab output.

Path: ~/.PyconDE
Go Back
Technical_Analysis.ipynb Share

notebook:Technical_Analysis.ipynb
IP[y] Technical_Analysis Last saved: Oct 14 16:09 link to this page

File Edit View Insert Cell Kernel Help

None Environment: np17py27-1.5

Technical Analysis

This is an example IPython Notebook to illustrate the use of IPython for professional and scientific publishing

by

Continuum Analytics Europe GmbH, Germany.

Dr. Yves J. Hilpisch

www.continuum.io -- europe@continuum.io

All rights reserved.

Technical Analysis of Stocks

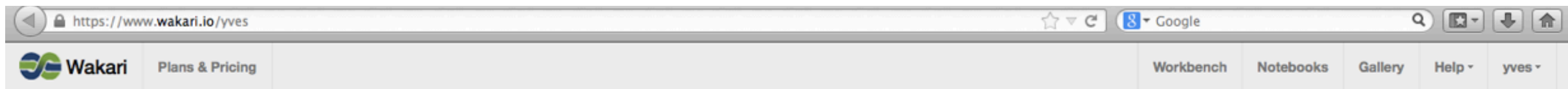
Let Wikipedia explain what **Technical Stock Analysis** is all about (cf. http://en.wikipedia.org/wiki/Technical_analysis):

"In finance, technical analysis is a security analysis methodology for forecasting the direction of prices through the study of past market data, primarily price and volume. Behavioral economics and quantitative analysis use many of the same tools of technical analysis, which, being an aspect of active management, stands in contradiction to much of modern portfolio theory. The efficacy of both technical and



CONTINUUM
ANALYTICS

Teilen von ganzen IPYNB Projekten



yves

User Info

Date Joined July 2, 2013, 6:12 a.m.

Shared Bundle list

Bundle Name	Created	Clone Count	View Count	Description
Continuum_EuroPython_2013_Better_Future_Keynote	July 2, 2013, 9:01 a.m.	0	28	The EuroPython 2013 Keynote of Yves Hilpisch, Managing Director, Continuum Analytics Europe.
Continuum_N_Body_Simulation_Numba	Aug. 1, 2013, 3:29 a.m.	0	29	IPython Notebook illustrating the use of Numba just-in-time compiling for performance enhancements.
Yves_Hilpisch_Python_in_Finance_Talk_NYC	Aug. 29, 2013, 4:30 p.m.	0	39	IPython Notebook underlying my talk at Night of Talks of New York City Python Meetup Group on Tuesday 27. August 2013 at Interactive Space.
Technical_Analysis	Oct. 14, 2013, 9:12 a.m.	0	0	IPython Notebook example for my talk "Scientific Publishing with Python" at Pycon DE 2013 in Cologne.

Home · Blog · Contact



CONTINUUM
ANALYTICS

Home · Blog · Contact



Continuum Analytics Europe GmbH
Rathausstrasse 75-79
66333 Voelklingen
Germany

www.continuum.io
europa@continuum.io

Dr. Yves J. Hilpisch | [@dyjh](#)