

# TechTalk Project Jupyter

February 24, 2015

## 1 Reproducible research with Project Jupyter

### 1.1 Working with the notebook

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

```
In [1]: 4 + 5
```

```
Out[1]: 9
```

```
In [2]: x = 4
        y = 3
```

```
In [3]: print('The sum of {} and {} is {}'.format(x, y, x + y))
```

```
The sum of 4 and 3 is 7.
```

```
In [4]: def sum(x, y):
        return x + y
```

```
In [5]: sum(3, 2)
```

```
Out[5]: 5
```

### 1.2 Meta data

```
In [6]: %load_ext watermark
        %watermark
```

```
19/02/2015 14:26:09
```

```
CPython 3.4.2
IPython 3.0.0-rc1
```

```
compiler : MSC v.1600 64 bit (AMD64)
system   : Windows
release  : 7
```

```
machine      : AMD64
processor    : Intel64 Family 6 Model 58 Stepping 9, GenuineIntel
CPU cores   : 4
interpreter : 64bit
```

### 1.3 Example: Analysis of grid frequency time series

```
In [7]: PURPLE = (107.0/255, 36.0/255, 124.0/255)
        FREQUENCY_COLOR = (16.0/255, 192.0/255, 225.0/255)
```

```
FREQUENCY_FILE = 'path/to/file'
```

```
In [8]: import pandas as pd
        %matplotlib inline
        import seaborn as sns
```

```
In [9]: freq = pd.read_hdf(FREQUENCY_FILE, '/frequency')
        print('Index of time series: {} to {}'.format(freq.index[0], freq.index[-1]))
        print('Time resolution of time series: {}'.format(freq.index.freq))
        print('Size of time series: {:.2f} MB.'.format(freq.nbytes / 1024 / 1024))
```

Index of time series: 2012-01-01 00:00:00+00:00 to 2013-12-31 23:59:59+00:00.

Time resolution of time series: <Second>.

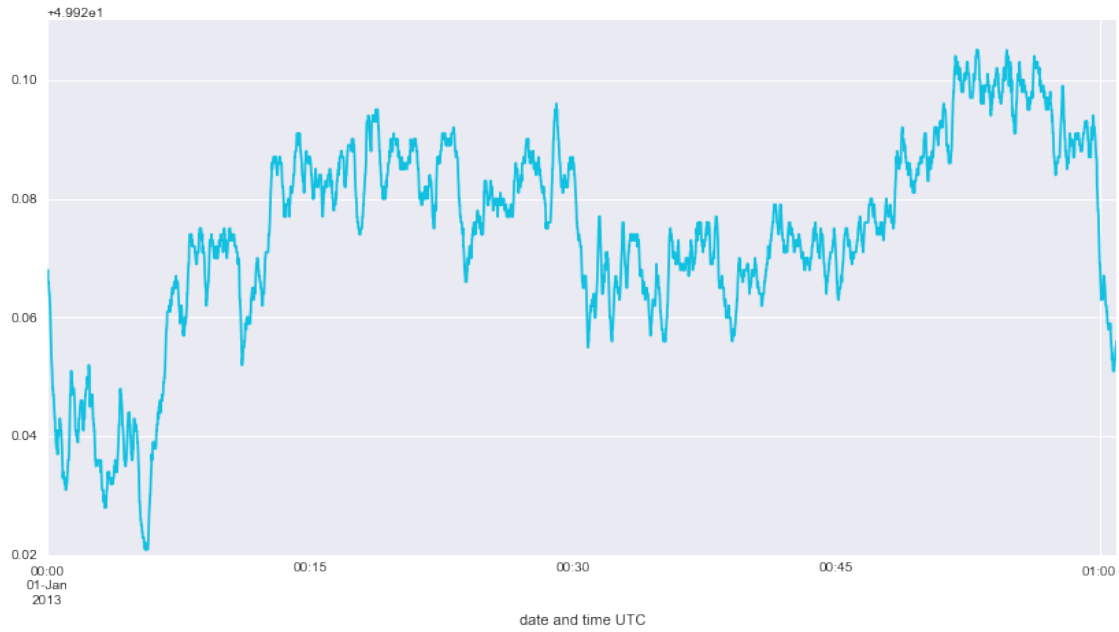
Size of time series: 481.86 MB.

```
In [10]: freq.describe()
```

```
Out[10]: count      63158400.000000
         mean         49.999966
         std           0.021858
         min          49.849000
         25%          49.986000
         50%          50.000000
         75%          50.014000
         max          50.145000
         Name: grid frequency in Hz, dtype: float64
```

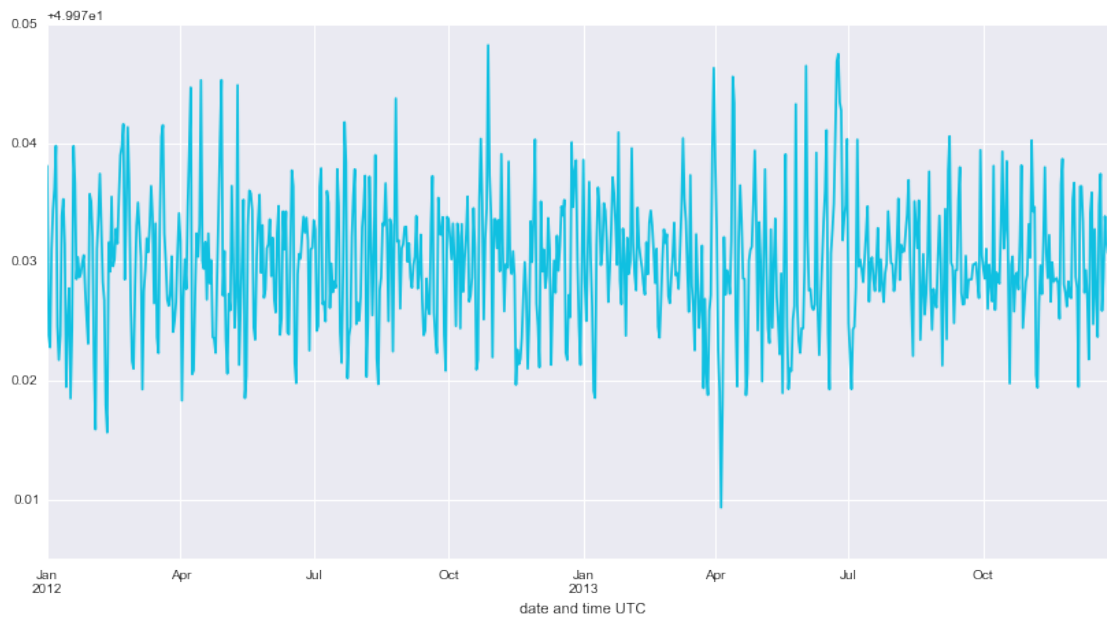
```
In [11]: freq['2013-01-01':'2013-01-01 01:00'].plot(figsize=(14,7), color=FREQUENCY_COLOR)
```

```
Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x90e0240>
```



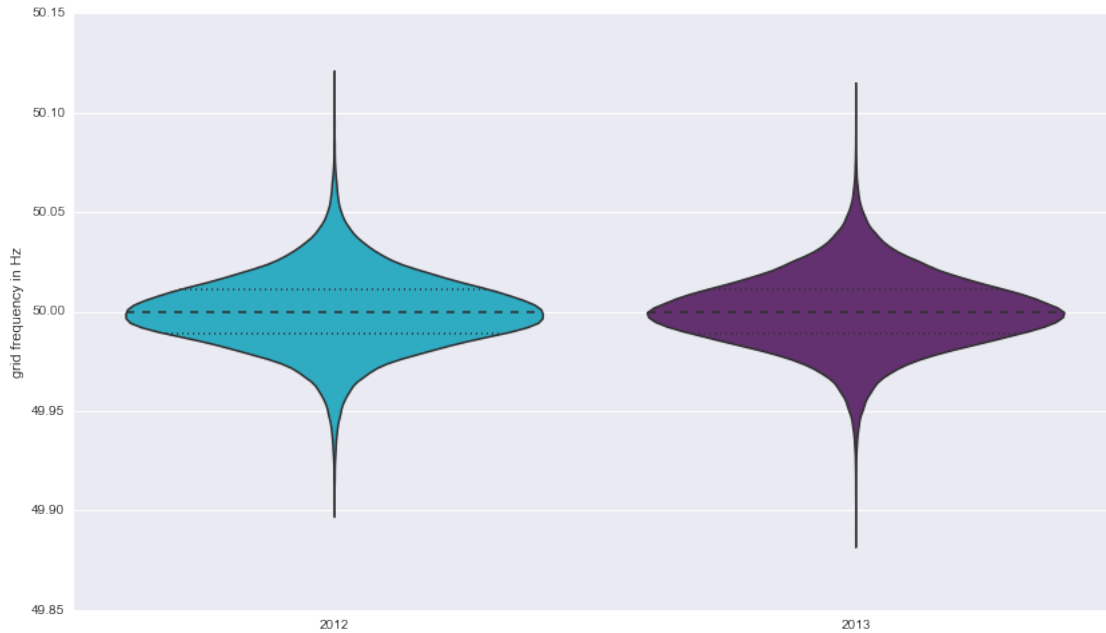
```
In [12]: freq.resample('D').plot(figsize=(14,7), color=FREQUENCY_COLOR)
```

```
Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0x9012c88>
```



```
In [13]: sns.set(rc={"figure.figsize": (14, 8)})
sns.violinplot(freq.resample('10Min'), groupby=lambda x : x.year,
                color=[FREQUENCY_COLOR, PURPLE])
```

Out [13]: <matplotlib.axes.\_subplots.AxesSubplot at 0x8f56358>



### 1.3.1 Estimate number of grid events

In the following an estimation shall be made how many grid events occurred in the years 2012 and 2013.

```
In [14]: threshold = 49.87
         n_events = (freq < threshold).diff().sum() / 2
         print("""In the years 2012 and 2013 there were in total {:.0f} events in which
               the grid frequency layed below {} Hz.""").format(n_events, threshold)
```

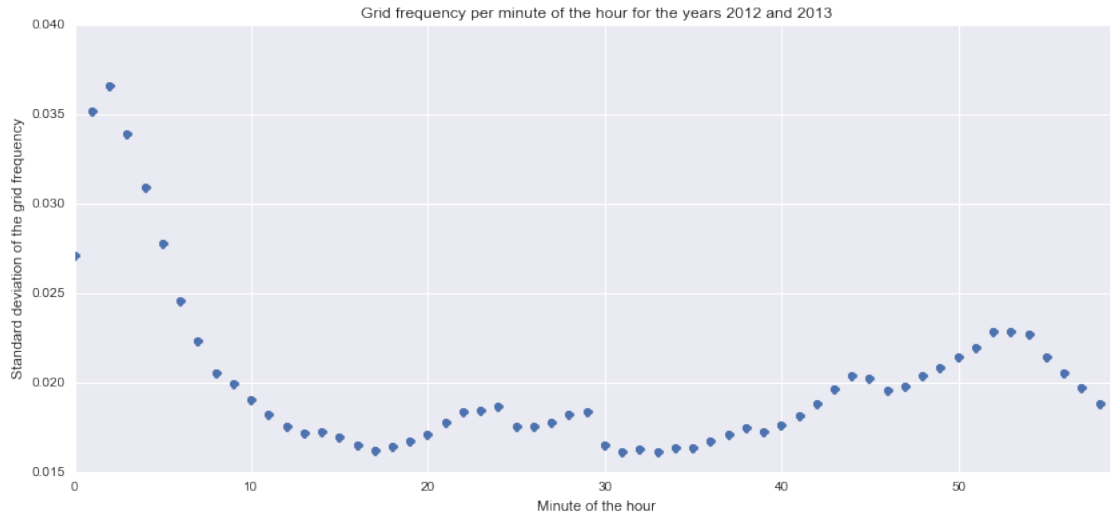
In the years 2012 and 2013 there were in total 29 events in which the grid frequency layed below 49.87 Hz.

### 1.3.2 Analysis of the standard deviation of grid frequency

In the following the standard deviation of the grid frequency shall be further examined. Main focus lays on the influence of the minute of the hour.

```
In [15]: lines = freq.resample('Min').groupby(lambda x : x.minute).std().plot(style='o',
                                       figsize=(14,6))
         lines.xaxis.set_label_text('Minute of the hour')
         lines.yaxis.set_label_text('Standard deviation of the grid frequency')
         lines.set_title('Grid frequency per minute of the hour for the years 2012 and 2013')
```

Out [15]: <matplotlib.text.Text at 0xd3b1390>



## 1.4 More

- Export to
  - latex
  - markdown
  - pdf
  - html
  - slideshow
- Supported languages:
  - Python
  - Julia
  - R
  - Octave
  - Matlab
  - Haskell
  - Perl
  - Ruby
  - Erlang
  - ...
- Try it: [try.jupyter.org](http://try.jupyter.org)
- See what others are using it for:
  - [nbviewer.ipython.org](http://nbviewer.ipython.org)
  - [A gallery of interesting Notebooks](#)
- Installation of the current version (still using the name IPython notebook) can be done easiest with [Python Scientific Distribution Anaconda](#)