

UC Berkeley - Physics 5CL

Statistics Jupyter Notebook

This notebook is a supplement to the Statistics Reference Sheet. Here, I provide examples of all of the calculations mentioned in the Reference Sheet using Python. In particular, I take advantage of numpy, scipy and matplotlib. An extra section is included here for nonlinear curve fitting.

Some Useful Python Commands for Handling Data

Below we load the relevant packages for this document, *numpy*, *scipy*, *pandas* and *matplotlib*:

```
In [60]: import numpy as np
import scipy.optimize as sp
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
print(plt.style.available)
plt.style.use('fivethirtyeight')

['seaborn-darkgrid', 'Solarize_Light2', 'seaborn-notebook', 'c
lassic', 'seaborn-ticks', 'grayscale', 'bmh', 'seaborn-talk',
'dark_background', 'ggplot', 'fivethirtyeight', '_classic_test
', 'seaborn-colorblind', 'seaborn-deep', 'seaborn-whitegrid',
'seaborn-bright', 'seaborn-poster', 'seaborn-muted', 'seaborn-
paper', 'seaborn-white', 'fast', 'seaborn-pastel', 'seaborn-da
rk', 'tableau-colorblind10', 'seaborn', 'seaborn-dark-palette
']
```

Here are some useful commands for handling data in spreadsheets:

```
In [61]: #df2 = pd.DataFrame({'x': x, 'y': y})
#df2.to_csv('sample_data2.csv')          # create csv from dat
a. See also: to_excel()

# random data
N = 1000
x = np.arange(0, N, 1)
y = np.random.normal(5, 2, N)

# Create pandas DataFrame from arrays
df = pd.DataFrame({
```