PyData 2013 Highlights

November 21, 2013

PyData 2013 Highlights

http://wiki.wgenhq.net/wiki/index.php/PyData_2013
Milen Pavlov
2013-11-21

0.1 Conference Themes

- data analysis
- scale and parallelism
- visualization

0.2 Conference dates

In [8]:
```
print "\n".join(['2013-11-%d' % day for day in range(8,11)])
```

```
2013-11-8
2013-11-9
2013-11-10
```

0.3 Venue

1 Chase Manhattan Plaza

In [2]:
```
from IPython.core.display import display, Image
Image(filename='img/venue1.jpg')
```

Out [2]:

In [10]:
```
Image(filename='img/venue2.jpg')
```

Out [10]:

0.4 General Observations

In [12]:
```
display(Image(filename='img/Borat_big_data.png'))
print "When dealing with big data avoid ETLs, bring code to data"
```


```
When dealing with big data avoid ETLs, bring code to data
```

In [13]:
```
display(Image('img/Borat_80_20_rule.png'))
print "These days learning algorithms are easy to apply, most time is spent exploring
```

Big Data Borat
@BigDataBorat

Follow

In Data Science, 80% of time spent prepare data, 20% of time spent complain about need for prepare data.

These days learning algorithms are easy to apply, most time is spent exploring the data and its features

In [5]:
```
display(Image("img/heart.png", width=100))
display(Image("http://ipython.org/ipython-doc/rel-1.1.0/_images/ipynb_icon_128x128.png
print "Presentation media: "
print "  iPython   20"
print "  Keynote    1"
print "  Google     1"
print "  PowerPoint 0"
```


```
Presentation media:
  iPython   20
  Keynote    1
  Google     1
  PowerPoint 0
```

0.5 Notable talks

scikit-learn: Machine learning at U of Washington (Jake Vanderplas)

In [15]:
```
from sklearn import datasets
digits = datasets.load_digits()
print digits.DESCR[:52]
print digits.data.shape
```

```
 Optical Recognition of Handwritten Digits Data Set

(1797, 64)
```

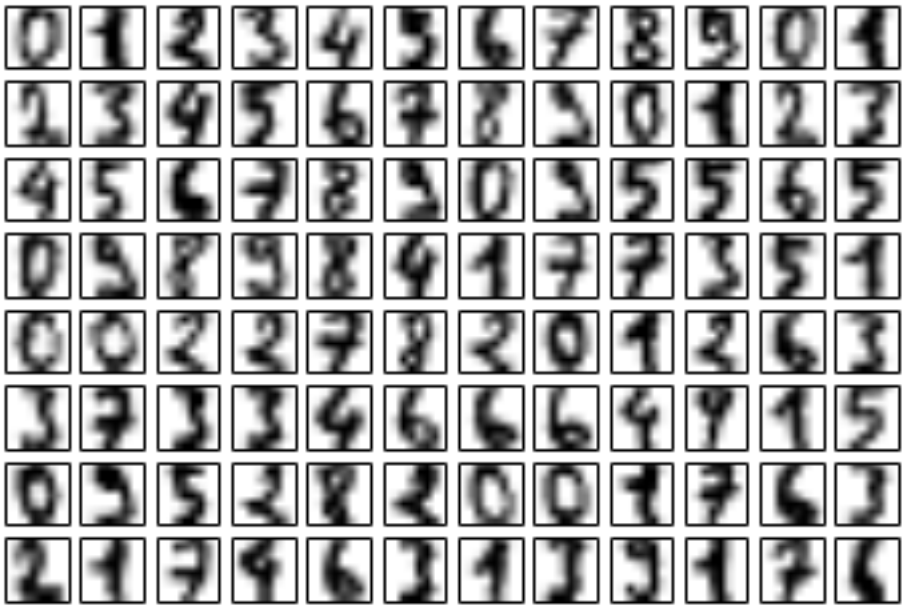
In [16]:
```
print digits.target[0]
print digits.data[0]
```

```
0
[  0.   0.   5.  13.   9.   1.   0.   0.   0.   0.  13.  15.  10.  15.
5.
   0.   0.   3.  15.   2.   0.  11.   8.   0.   0.   4.  12.   0.   0.
8.
   8.   0.   0.   5.   8.   0.   0.   9.   8.   0.   0.   4.  11.   0.
1.
  12.   7.   0.   0.   2.  14.   5.  10.  12.   0.   0.   0.   0.   6.
13.
  10.   0.   0.   0.]
```

In [17]:
```
for i in range(8):
    print ["%2.0f" % d for d in digits.data[0][8*i:8*i+8]]
```

```
[' 0', ' 0', ' 5', '13', ' 9', ' 1', ' 0', ' 0']
[' 0', ' 0', '13', '15', '10', '15', ' 5', ' 0']
[' 0', ' 3', '15', ' 2', ' 0', '11', ' 8', ' 0']
[' 0', ' 4', '12', ' 0', ' 0', ' 8', ' 8', ' 0']
[' 0', ' 5', ' 8', ' 0', ' 0', ' 9', ' 8', ' 0']
[' 0', ' 4', '11', ' 0', ' 1', '12', ' 7', ' 0']
[' 0', ' 2', '14', ' 5', '10', '12', ' 0', ' 0']
[' 0', ' 0', ' 6', '13', '10', ' 0', ' 0', ' 0']
```

In [18]:
```
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
fig, ax = plt.subplots(8, 12, subplot_kw={'xticks':[], 'yticks':[]})
for i in range(ax.size):
    ax.flat[i].imshow(digits.data[i].reshape(8, 8), cmap=plt.cm.binary)
```



In [19]:
```
from sklearn.naive_bayes import GaussianNB
X = digits.data
y = digits.target

# Instantiate the estimator
clf = GaussianNB()

# Fit the estimator to the data, leaving out the last five samples
clf.fit(X[:-15], y[:-15])

# Use the model to predict the last five labels
y_pred = clf.predict(X[-15:])

print y_pred
print y[-15:]
```

```
[2 8 5 7 9 5 4 8 1 4 9 0 8 9 8]
[2 2 5 7 9 5 4 8 8 4 9 0 8 9 8]
```

pymc3: Bayesian data analysis (Thomas Wiecki, Quantopian & Brown University)

- scikit learn is cool, but models are black boxes
- not good at conveying **what** they have learned
- probabilistic programming: black box inference engine, but open box models
- pymc3
  - good at linear regression, bayesian inference, state of the art MCMC methods
  - good at speed: compiles to C just-in-time using Theano
  - not good at usability

In [163]:
```
print "sorry, no easy example to show :("
```

```
sorry, no easy example to show :(
```

Packaging and distributing with Anaconda (Travis Oliphant, Continuum Analytics)

- conda = a cross-platform system-level language-independent virtualenv
  - good scientific packages support currently, not all packages
  - user installable (don't need root)
- conda package
- conda build
- wakari = conda + ipython-notebook

In [164]:
```
!~/anaconda/bin/conda info -e
```

```
# conda environments:
#
py3                      /Users/mpavlov/anaconda/envs/py3
root                  *  /Users/mpavlov/anaconda
```

In [165]:
```
!~/anaconda/bin/conda remove --name py3 --all --yes
!~/anaconda/bin/conda info -e
```

```
Package plan for package removal in environment
/Users/mpavlov/anaconda/envs/py3:

The following packages will be UN-linked:

    package                    |            build
    ---------------------------|-----------------
    distribute-0.6.45          |           py33_1
    pip-1.4.1                  |           py33_0
    python-3.3.2               |                1
    readline-6.2               |                1
    sqlite-3.7.13              |                1
    tk-8.5.13                  |                1
    zlib-1.2.7                 |                1

Unlinking packages ...
 ||   0% [distribute          ] ||  14% [pip                 ] ||  28%
[python              ] ||  42% [readline            ] ||  57% [sqlite
] ||  71% [tk                  ] ||  85% [zlib                ] ||
100% [      COMPLETE      ] || 100%
# conda environments:
#
root                  *  /Users/mpavlov/anaconda
```

In [166]:
```
!~/anaconda/bin/conda create --name py3 python=3.3 pip --yes
```

```
Package plan for creating environment at
/Users/mpavlov/anaconda/envs/py3:

The following packages will be linked:

    package                    |            build
    ---------------------------|-----------------
    distribute-0.6.45          |           py33_1   hard-link
    pip-1.4.1                  |           py33_0   hard-link
    python-3.3.2               |                1   hard-link
    readline-6.2               |                1   hard-link
    sqlite-3.7.13              |                1   hard-link
    tk-8.5.13                  |                1   hard-link
    zlib-1.2.7                 |                1   hard-link

Linking packages ...
 ||   0% [distribute          ] ||  14% [pip                 ] ||  28%
[python              ] ||  42% [readline            ] ||  57% [sqlite
] ||  71% [tk                  ] ||  85% [zlib                ] ||
100% [      COMPLETE      ] || 100%
#
# To activate this environment, use:
# $ source activate py3
#
# To deactivate this environment, use:
# $ source deactivate
#
```

In [167]:
```
!source ~/anaconda/bin/activate py3 && which python
```

```
prepending /Users/mpavlov/anaconda/envs/py3/bin to PATH
/Users/mpavlov/anaconda/envs/py3/bin/python
```

In [168]:
```
!~/anaconda/bin/conda list --name py3
```

```
# packages in environment at /Users/mpavlov/anaconda/envs/py3:
#
distribute                0.6.45                   py33_1
pip                       1.4.1                    py33_0
python                    3.3.2                         1
readline                  6.2                           1
setuptools                0.6c11                    <pip>
sqlite                    3.7.13                        1
tk                        8.5.13                        1
zlib                      1.2.7                         1
```

- tried running disco_profiling_task_status_service on python 3.3 in a conda env
  - while pyramid is python 3.2 compatible, pyramid_whoauth has invalid syntax (except Exception, e)
  - inconsistencies between tools: conda install, pip, easy_install
  - needs more work

ddpy: Data-driven music for big data analysis (Thomas Levine, CSV Soundsystem)

- visualizations alone can only show you 2D, if you're skilled you can do 5-6 dimensions at most
- music can show you dozens
- example: http://fms.csvsoundsystem.com/

In [169]:
```
print "The Future: data gastronomification can use all 5 senses"
Image("https://github.com/csv/ddpy/raw/master/img/artichoke.jpg")
```

```
The Future: data gastronomification can use all 5 senses
```

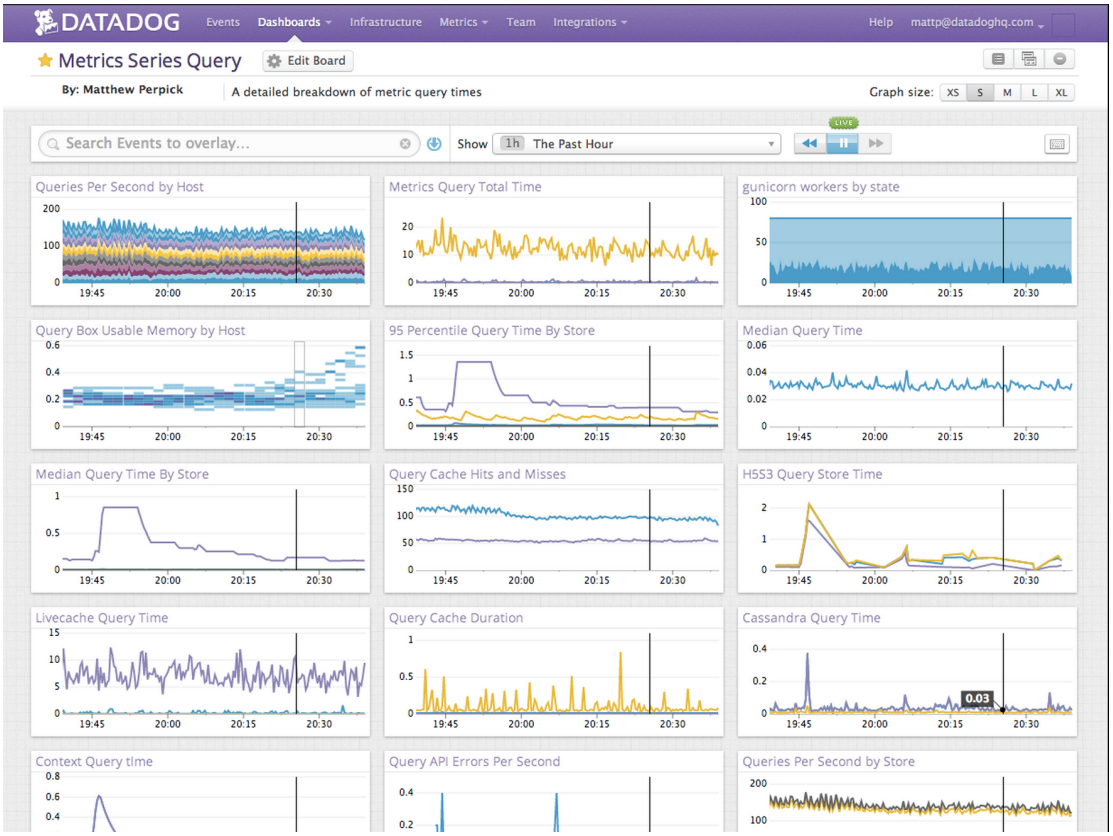
Out [169]:

Python @ Datadog: Building High-Volume Data Systems in the Python Ecosystem (Matt Perpick, Datadog)

- dashboards, metrics, alerting

In [170]:
```
Image("img/datadog.png")
```

Out [170]:



- billions of data points daily, low latency
- how to eliminate Python overhead (for the code sections that matter)
  - use c bindings, skip dynamic types, interpreter
  - cython is your friend: compiles to c, reads like python
- concurrency soup - datadog has tried everything and it all sucks; no miracle solutions

Keynote: iPython, attributes of software, and our work (Brian Granger, Cal Poly State University, IPython)

- the tools we use affect our behavior
- our behavior affects our work
- tools' attributes get inherited, through our behavior, by our work output
- ipython's attributes
  - useful in multiple contexts
    - Individual, interactive exploration, Debugging, testing, Production runs, Parallel computing, Collaboration, Publication, Presentation, Teaching/Learning
  - close to data
    - visualize, interact, compute, repeat
  - open
    - everything is public: code, data formats, issues, roadmap, chatroom, google hangout meetings broadcasted live, recorded and available online
  - multilingual
    - supported: ruby, bash, julia, many others
    - not yet: R (wanted!)

- also supports
  - latex equations
  - audio
  - video
  - inline html
  - iframes
  - interactive plots
  - partial example: http://nbviewer.ipython.org/url/github.com/ipython/ipython/raw/master/examples/notebooks/Part%204%20%20Markdown%20Cells.ipynb

Lightning talk: prettyplotlib

- matplotlib wrapper
- sensible defaults
- see https://github.com/olgabot/prettyplotlib

0.6 Summary of cool tools

- ipython and nbviewer.ipython.org
- anaconda
- python 3
- pandas, numpy, cython
- scikit-learn and pymc3
- prettyplotlib

---

the end