Introduction to data science

Patrick Shafto

Department of Math and Computer Science

Plan for today

- Programming for data science

HW 2

- Write a short tutorial on dicts (yes, again), functions, and classes
- Cover basic functionality with worked examples
- Due Sunday by 11:59 pm

- Evaluations due Tuesday by 11:59pm
    - Use the rubric!
    - Justify your scores!

Feedback to Learner

A well structure and explained tutorial. All the topics had been covered clearly not only with sufficient examples but also with good introduction for each topic. I found a little difficulty in understanding your programs. It could have been better if you had given like 2-3 lines of explanation for each code snippet.

Feedback to Learner

Superb job

Feedback to Learner

Very well done. Brief intro and detailing is up to the mark. Some tidiness to be maintained. Rest is perfect.

Feedback to Learner

Its a good tutorial but there are some points that I think is missing. 1. Put the Area of topics that you are going to cover in the tutorial at the top 2. Dictionaries are very flexible and can store different data types like it can store a string, int, float, even a list in a single dictionary. So check on that too. 3. In dictionary you can also remove elements by using methods like del, pop and popitem so have a go through of that. 4. Not much has been covered in the topic of classes. 5. Classes and objects are very important topics and it has many concepts to cover in it such as Inbuilt classes, The init() Function used in classes (You have defined it but have not explained what it is and what it does), Difference between class variables and instance variables, Instance and class methods, Modify and delete properties. 6. Provide the reference link from where you learned the topics at the end of the tutorial.

Feedback to Learner

Tutorial covers the basics of all the three topics dictionaries functions and classes in python. It is well written but you could have used the markdown more efficiently to categorize the main topics and sub topics to make it more understandable.

Feedback to Learner

Formatting could better

One last view on data science: Interview questions!

One last view on data science: Interview questions!

Q1. Explain what regularization is and why it is useful.

Q2. Which data scientists do you admire most? which startups?

Q3. How would you validate a model you created to generate a predictive model of a quantitative outcome variable using multiple regression.

Q4. Explain what precision and recall are. How do they relate to the ROC curve?

Q5. How can you prove that one improvement you've brought to an algorithm is really an improvement over not doing anything?

One last view on data science: Interview questions!

8. What is statistical power?

9. Explain what resampling methods are and why they are useful. Also explain their limitations.

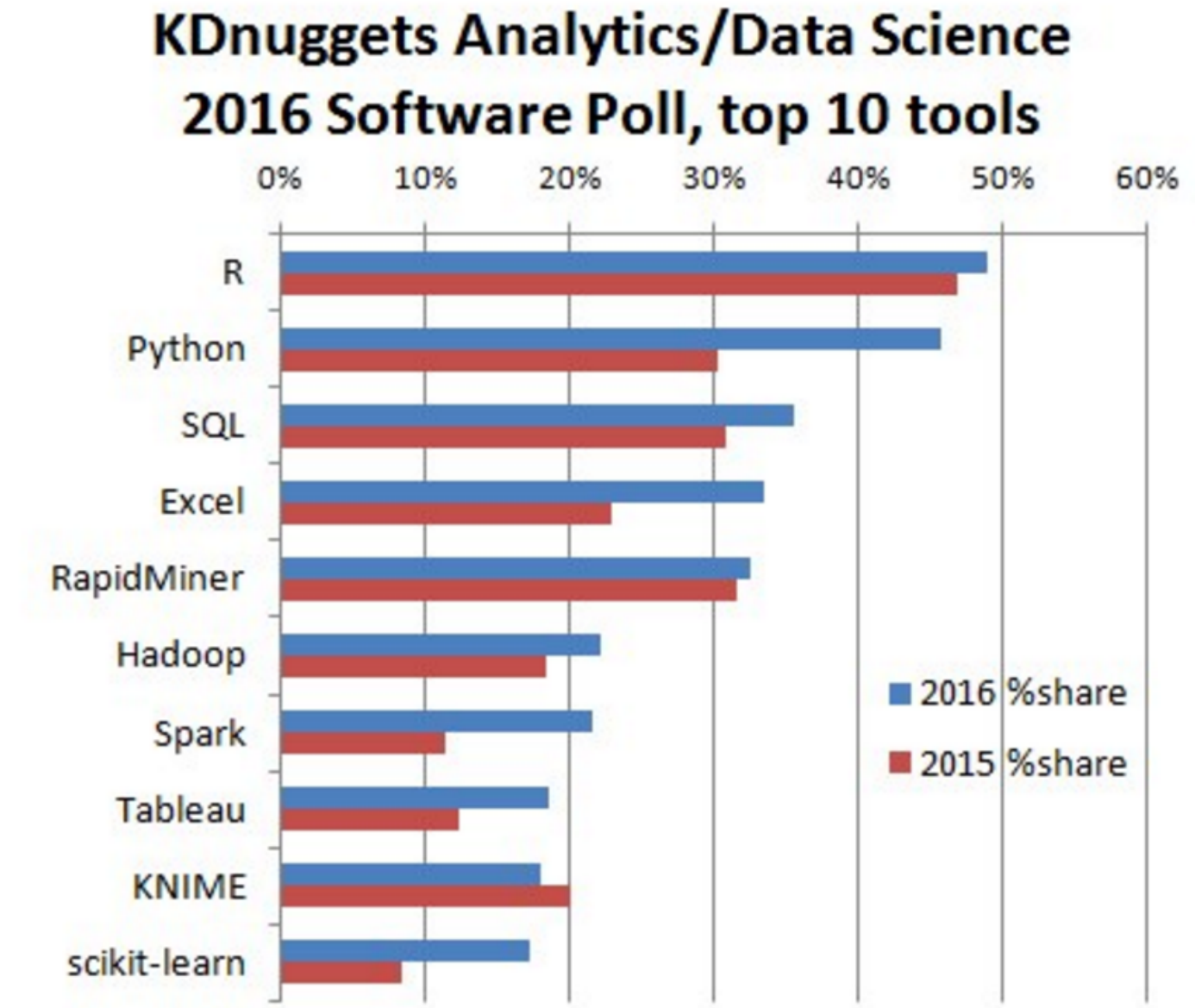
10. Is it better to have too many false positives, or too many false negatives? Explain.

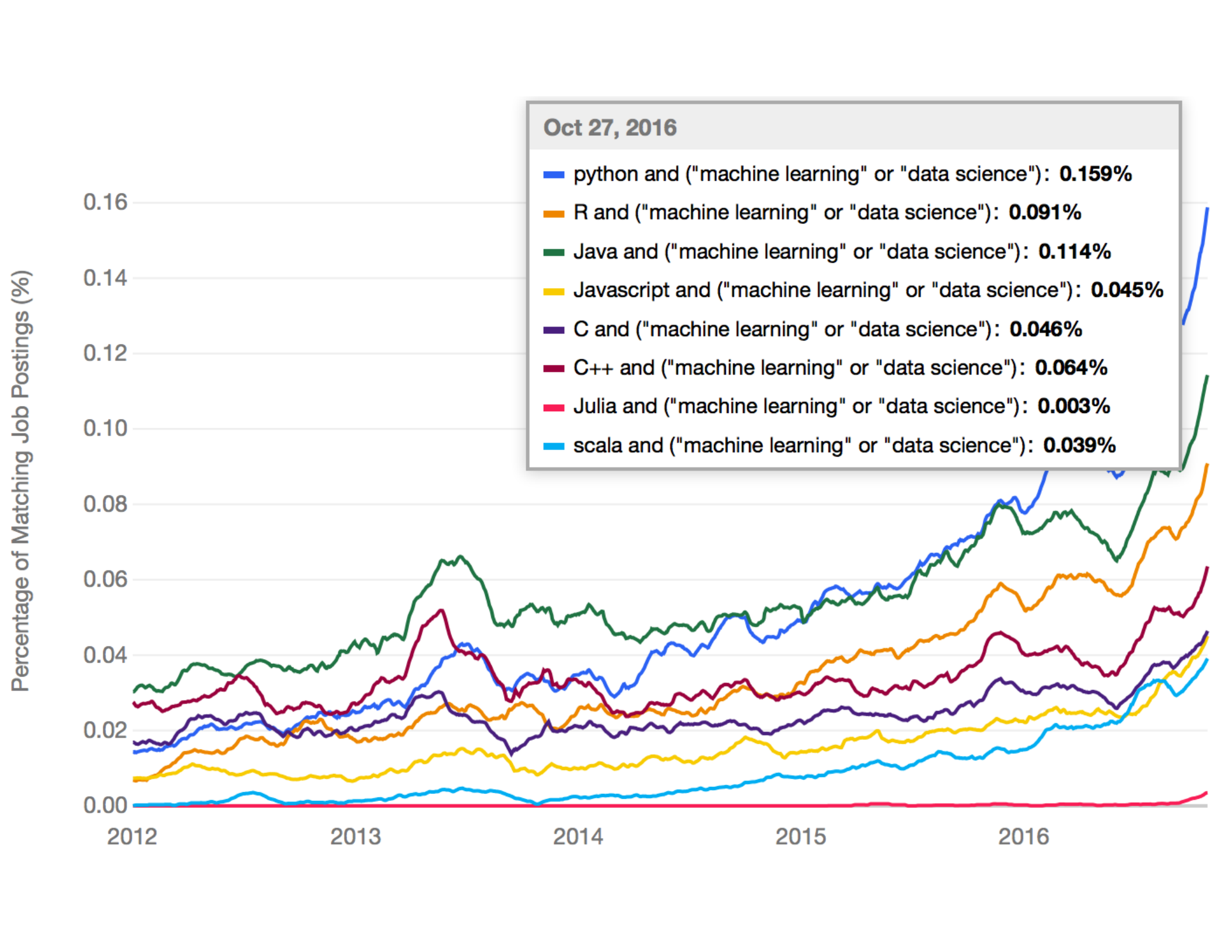
11. What is selection bias, why is it important and how can you avoid it?

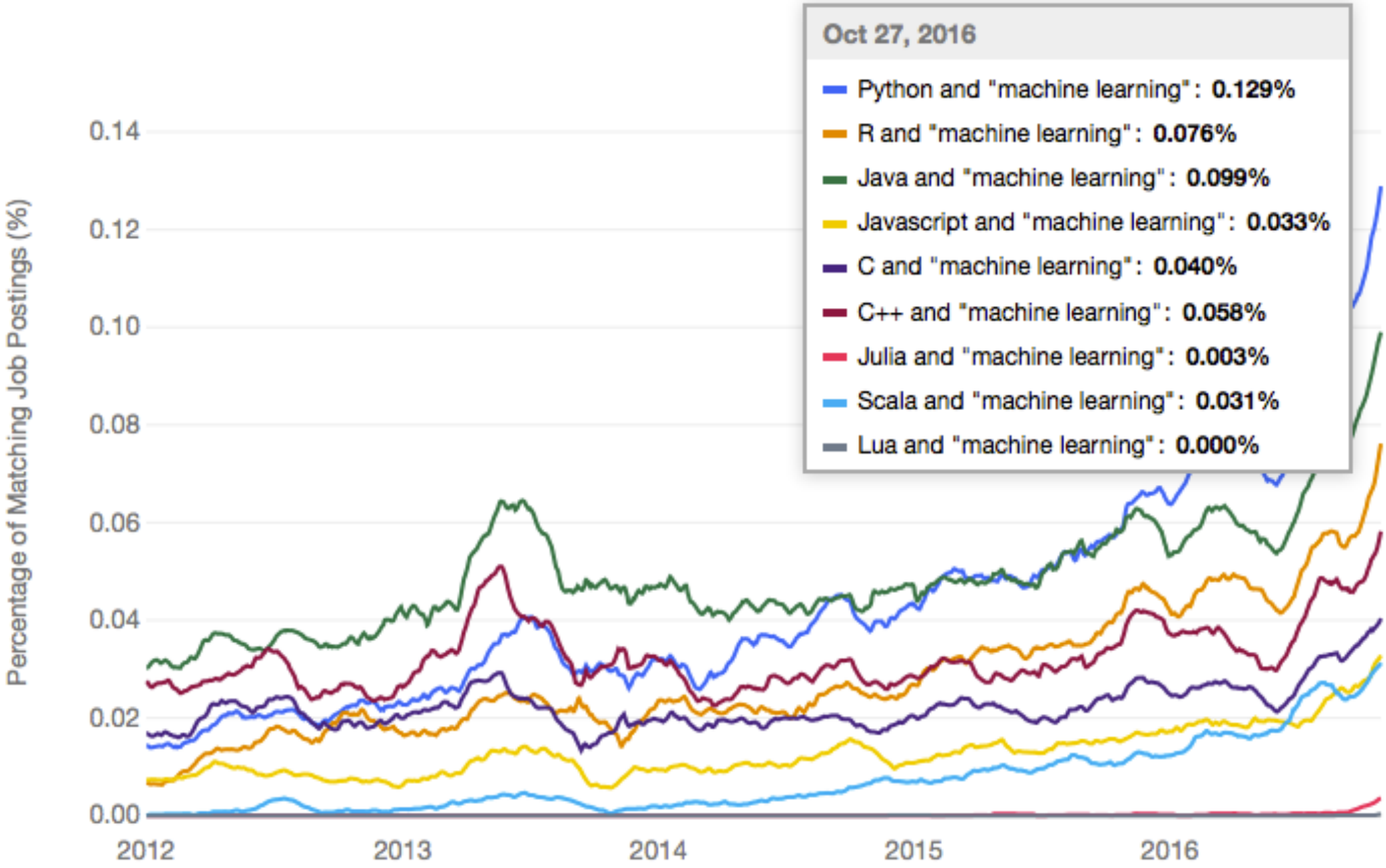
Programming for data science

What languages and packages are being used?

How is this changing over time?


KDnuggets Analytics/Data Science
2016 Software Poll, top 10 tools
0%
10%
20%
30%
40%
50%
60%
R
Python
SQL
Excel
RapidMiner
Hadoop
Spark
Tableau
KNIME
scikit-learn
2016 %share
2015 %share

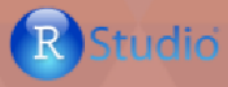

Oct 27, 2016
python and ("machine learning" or "data science"): 0.159%
R and ("machine learning" or "data science"): 0.091%
Java and ("machine learning" or "data science"): 0.114%
Javascript and ("machine learning" or "data science"): 0.045%
C and ("machine learning" or "data science"): 0.046%
C++ and ("machine learning" or "data science"): 0.064%
Julia and ("machine learning" or "data science"): 0.003%
scala and ("machine learning" or "data science"): 0.039%
Percentage of Matching Job Postings (%)
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
2012
2013
2014
2015
2016


Oct 27, 2016
Python and "machine learning": 0.129%
R and "machine learning": 0.076%
Java and "machine learning": 0.099%
Javascript and "machine learning": 0.033%
C and "machine learning": 0.040%
C++ and "machine learning": 0.058%
Julia and "machine learning": 0.003%
Scala and "machine learning": 0.031%
Lua and "machine learning": 0.000%
Percentage of Matching Job Postings (%)
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
2012
2013
2014
2015
2016


R
VS.
python

Getting Started

IDE



Popular Packages

- ✓ dplyr, plyr and data.table to easily manipulate data.
- ✓ stringr to manipulate strings.
- ✓ zoo to work with regular and irregular time series
- ✓ ggvis, lattice and ggplot2 to visualize data.
- ✓ caret for machine learning.

Tip: check out DataCamp's online interactive courses and tutorials!

IDE

There are many Python IDEs to chose from. However, Spyder and IPython Notebook are most popular.

Tip: also look up Rodeo, the "data science IDE for Python"

Popular Libraries

- ✓ pandas to easily manipulate data.
- ✓ SciPy /NumPy for scientific computing.
- ✓ sckikit-learn to use machine learning methods.
- ✓ matplotlib to make graphics.
- ✓ statsmodels to explore data, estimate statistical models, and perform statistical tests and unit tests.

What have we been talking about?

What have we been talking about?

Why have we been talking about it?

What have we been talking about?

Why have we been talking about it?

How have we been talking about it?

What have we been talking about?

Why have we been talking about it?

How have we been talking about it?

Show, don’t tell!

Python for data science

Libraries for python:

- **NumPy** stands for Numerical Python. The most powerful feature of NumPy is n-dimensional array. This library also contains basic linear algebra functions, Fourier transforms, advanced random number capabilities and tools for integration with other low level languages like Fortran, C and C++
- **SciPy** stands for Scientific Python. SciPy is built on NumPy. It is one of the most useful library for variety of high level science and engineering modules like discrete Fourier transform, Linear Algebra, Optimization and Sparse matrices.
- **Matplotlib** for plotting vast variety of graphs, starting from histograms to line plots to heat plots.. You can use Pylab feature in ipython notebook (ipython notebook –pylab = inline) to use these plotting features inline. If you ignore the inline option, then pylab converts ipython environment to an environment, very similar to Matlab. You can also use Latex commands to add math to your plot.
- **Pandas** for structured data operations and manipulations. It is extensively used for data munging and preparation. Pandas were added relatively recently to Python and have been instrumental in boosting Python's usage in data scientist community.
- **Scikit Learn** for machine learning. Built on NumPy, SciPy and matplotlib, this library contains a lot of effiecient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction.

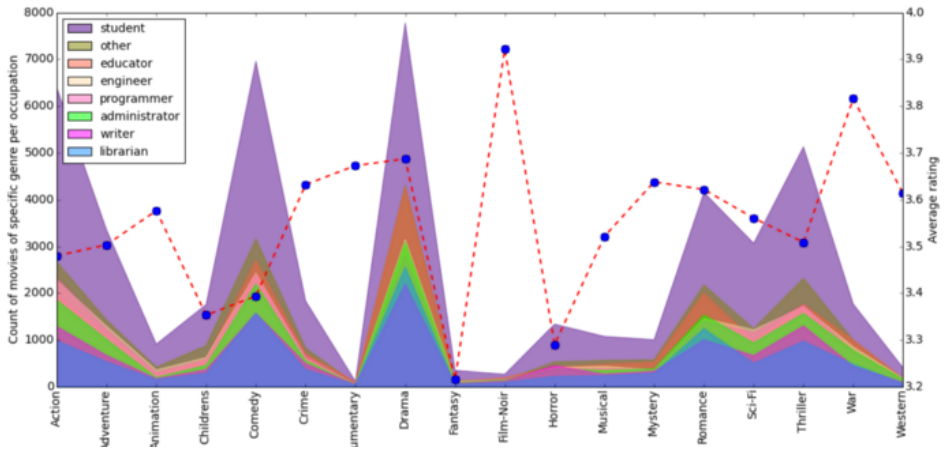
- **Statsmodels** for statistical modeling. Statsmodels is a Python module that allows users to explore data, estimate statistical models, and perform statistical tests. An extensive list of descriptive statistics, statistical tests, plotting functions, and result statistics are available for different types of data and each estimator.
- **Seaborn** for statistical data visualization. Seaborn is a library for making attractive and informative statistical graphics in Python. It is based on matplotlib. Seaborn aims to make visualization a central part of exploring and understanding data.
- **Bokeh** for creating interactive plots, dashboards and data applications on modern web-browsers. It empowers the user to generate elegant and concise graphics in the style of D3.js. Moreover, it has the capability of high-performance interactivity over very large or streaming datasets.
- **Blaze** for extending the capability of Numpy and Pandas to distributed and streaming datasets. It can be used to access data from a multitude of sources including Bcolz, MongoDB, SQLAlchemy, Apache Spark, PyTables, etc. Together with Bokeh, Blaze can act as a very powerful tool for creating effective visualizations and dashboards on huge chunks of data.
- **Scrapy** for web crawling. It is a very useful framework for getting specific patterns of data. It has the capability to start at a website home url and then dig through web-pages within the website to gather information.

Python libraries for data science 2

1. NumPy (Commits: 15980, Contributors: 522)
2. SciPy (Commits: 17213, Contributors: 489)
3. Pandas (Commits: 15089, Contributors: 762)

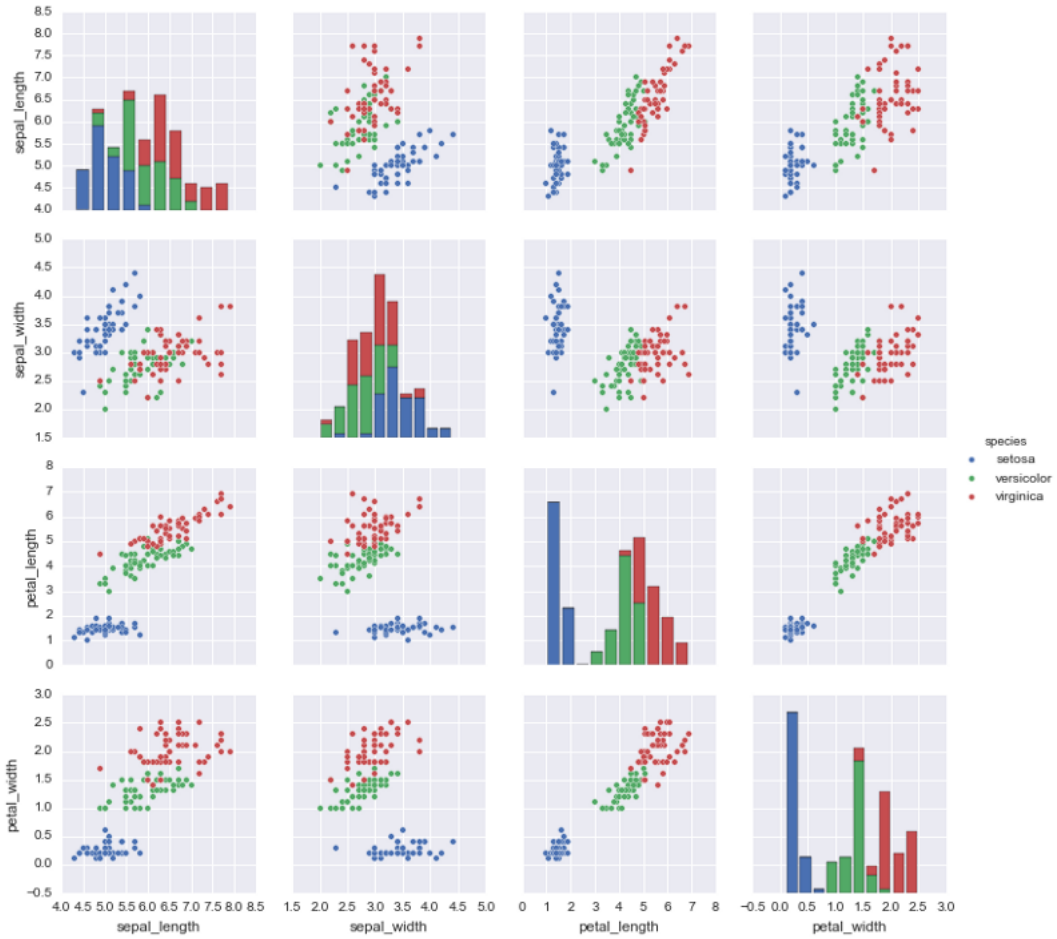
	A	B	C	D
0	A0	B0	C0	D0
1	A1	B1	C1	D1
2	A2	B2	C2	D2
3	A3	B3	C3	D3
4	X0	X1	X2	X3

4.Matplotlib (Commits: 21754, Contributors: 588)



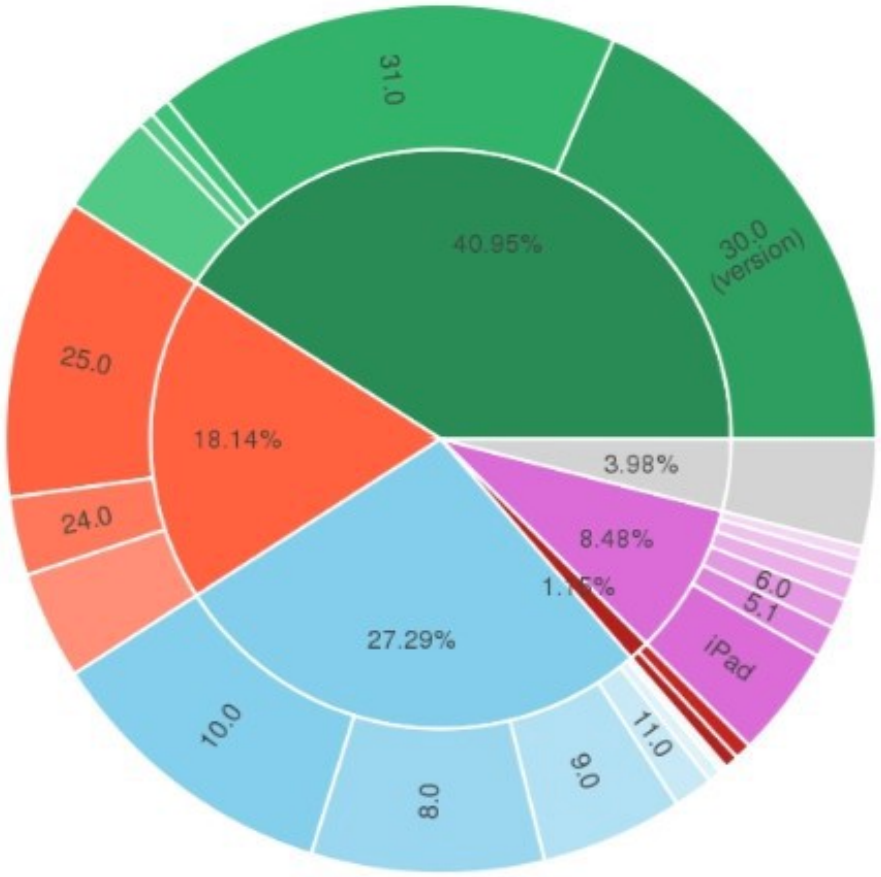
Python libraries for data science 2

5. Seaborn (Commits: 1699, Contributors: 71)



Python libraries for data science 2

6. Bokeh (Commits: 15724, Contributors: 223)



Python libraries for data science 2

7. Plotly (Commits: 2486, Contributors: 33)
8. SciKit-Learn (Commits: 21793, Contributors: 842)
9.Theano. (Commits: 25870, Contributors: 300)
10. TensorFlow. (Commits: 16785, Contributors: 795)
11. Keras. (Commits: 3519, Contributors: 428)
12. NLTK (Commits: 12449, Contributors: 196)
13. Gensim (Commits: 2878, Contributors: 179)
14. Scrapy (Commits: 6325, Contributors: 243)
15. Statsmodels (Commits: 8960, Contributors: 119)

Exploring data science

Hadley Wickham:
http://hadley.nz/

Exploring data science

Andreas Mueller:
http://datascience.columbia.edu/andreas-mueller

Exploring data science

Hillary Mason
https://hilarymason.com/

Exploring data science

Travis Oliphant
https://en.wikipedia.org/wiki/Travis_Oliphant

Exploring data science

Julia Language
https://julialang.org/

Exploring data science

Tensor flow
https://www.tensorflow.org/

Exploring data science

Pyro
https://github.com/uber/pyro

Exploring data science

Edward
http://edwardlib.org/

Exploring data science

The need for openness in data journalism
https://nbviewer.jupyter.org/github/brianckeegan/Bechdel/blob/master/Bechdel_test.ipynb

HW 3

- Write a short tutorial on numpy and scipy
- Cover basic functionality with worked examples
- Due Sunday by 11:59 pm
- Also read the Betchel test notebook and come prepared to answer questions
- Evaluations due Tuesday by 11:59pm
    - Use the rubric!
    - Justify your scores!