EECS 510: SOCIAL MEDIA MINING
SPRING 2015

Data Mining Essentials 2: Data Mining in Practice, with Python

Rosanne Liu

rosanne.liu@northwestern.edu

NORTHWESTERN UNIVERSITY

Outline

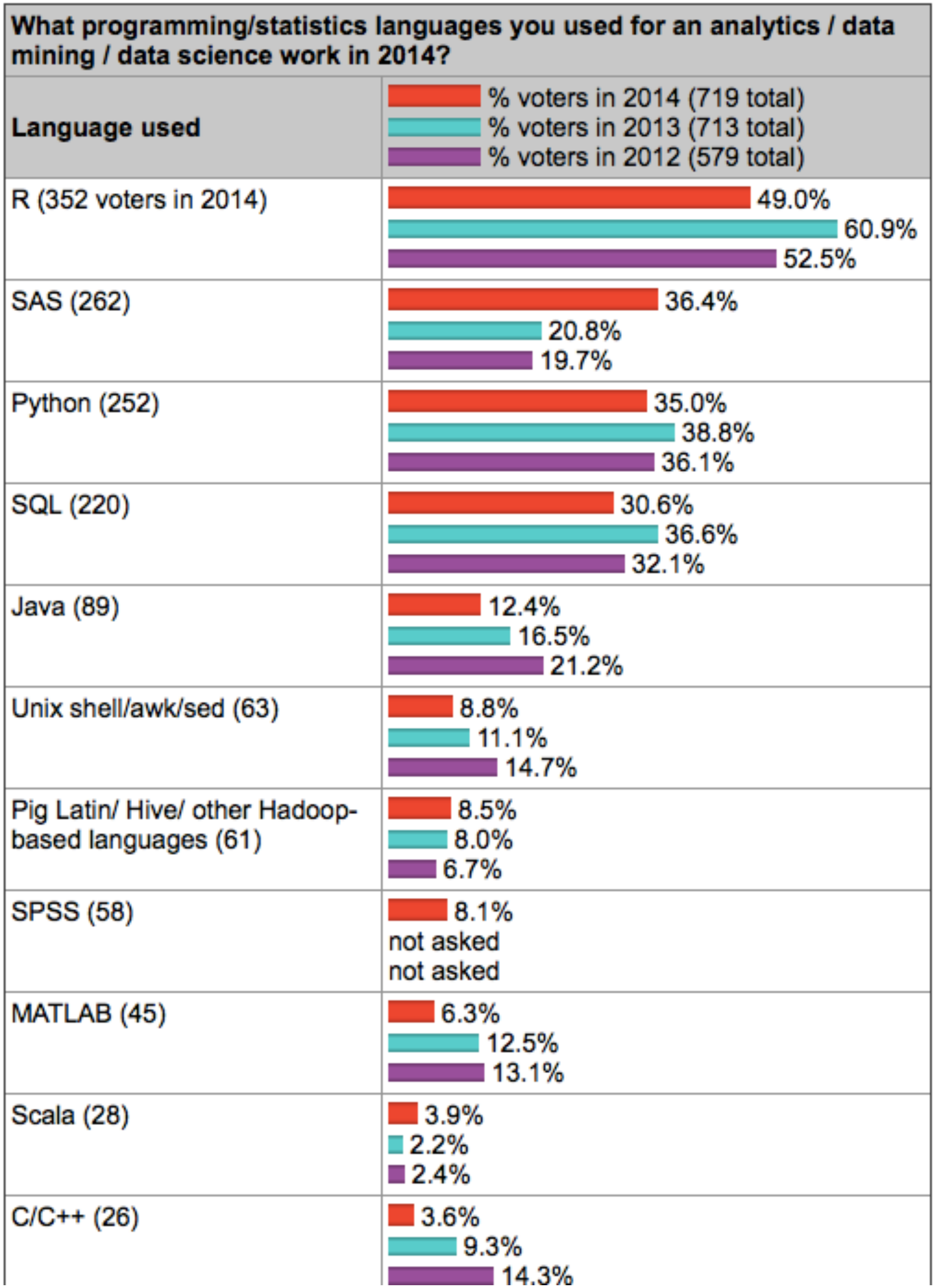
- Why Python?
- Intro to Python
- Intro to Scikit-Learn
- Unsupervised Learning
  - Demo on PCA, K-Means
- Supervised Learning
  - Demo on Linear Regression, Logistic Regression

Outline

- **Why Python?**
- Intro to Python
- Intro to Scikit-Learn
- Unsupervised Learning
  - Demo on PCA, K-Means
- Supervised Learning
  - Demo on Linear Regression, Logistic Regression

- **Why Python?**

What programming language do you use for data mining?

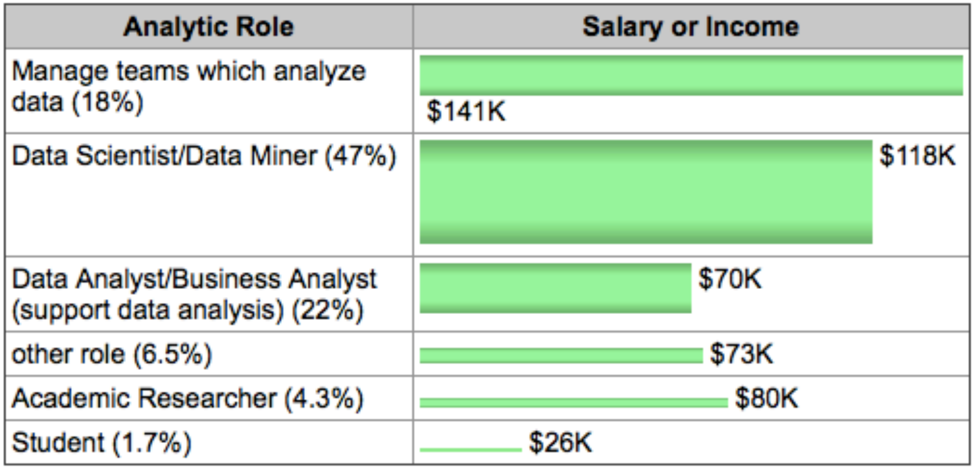


What programming/statistics languages you used for an analytics / data mining / data science work in 2014?

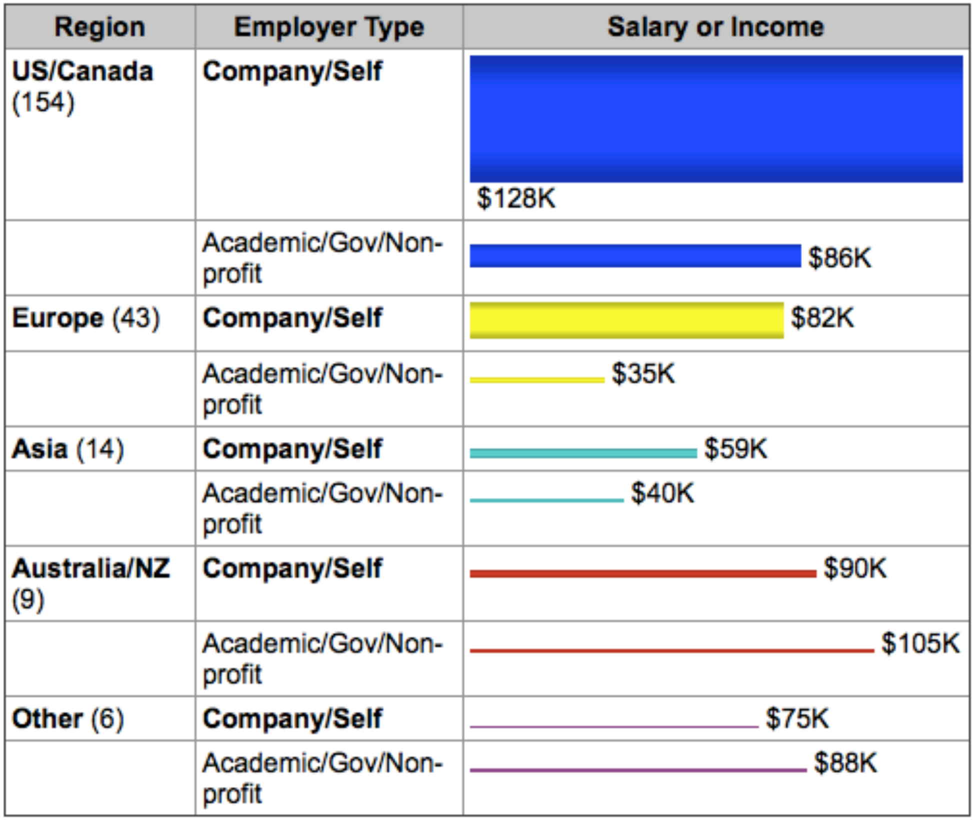
Language used	🟥 % voters in 2014 (719 total)	🟦 % voters in 2013 (713 total)	🟪 % voters in 2012 (579 total)
R (352 voters in 2014)	49.0%	60.9%	52.5%
SAS (262)	36.4%	20.8%	19.7%
Python (252)	35.0%	38.8%	36.1%
SQL (220)	30.6%	36.6%	32.1%
Java (89)	12.4%	16.5%	21.2%
Unix shell/awk/sed (63)	8.8%	11.1%	14.7%
Pig Latin/ Hive/ other Hadoop-based languages (61)	8.5%	8.0%	6.7%
SPSS (58)	8.1%	not asked	not asked
MATLAB (45)	6.3%	12.5%	13.1%
Scala (28)	3.9%	2.2%	2.4%
C/C++ (26)	3.6%	9.3%	14.3%

Source from: http://www.kdnuggets.com/polls/index.html

How much is your salary as analytics, data mining, data science professionals?



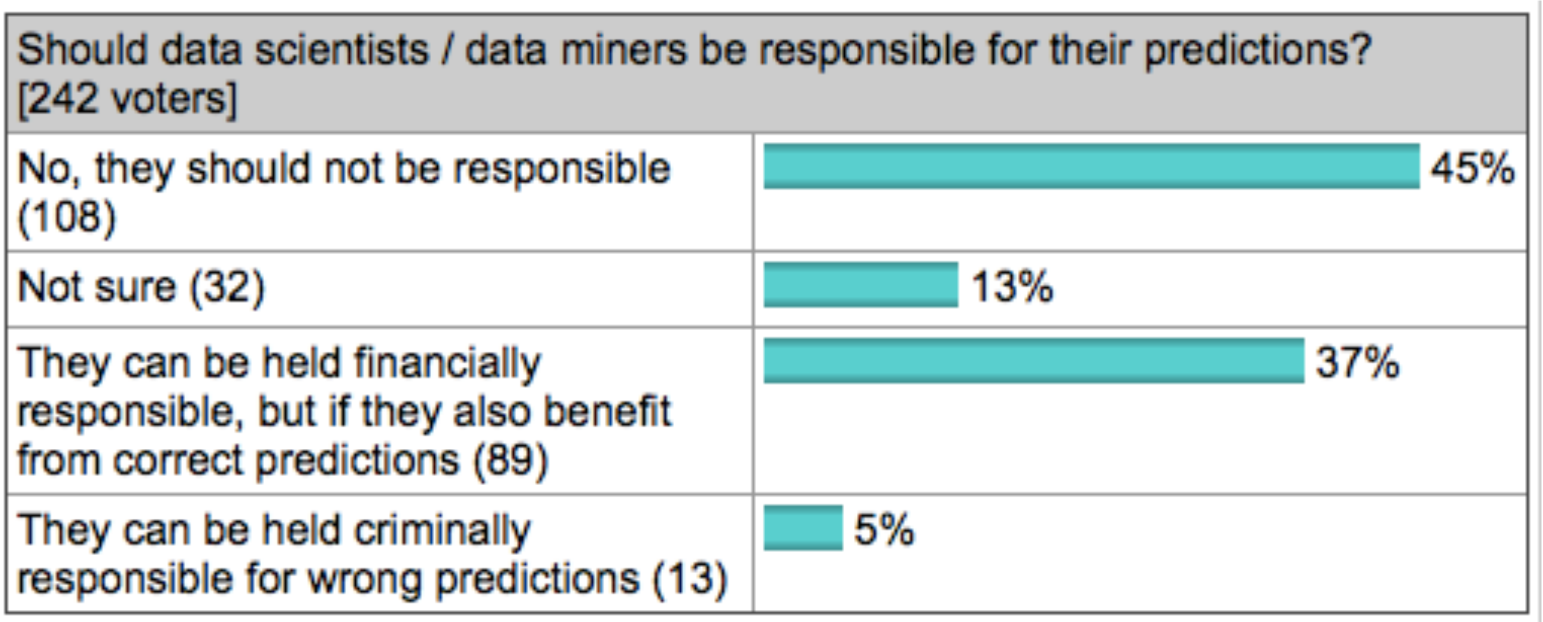
Analytic Role	Salary or Income
Manage teams which analyze data (18%)	$141K
Data Scientist/Data Miner (47%)	$118K
Data Analyst/Business Analyst (support data analysis) (22%)	$70K
other role (6.5%)	$73K
Academic Researcher (4.3%)	$80K
Student (1.7%)	$26K



Region	Employer Type	Salary or Income
US/Canada (154)	**Company/Self**	$128K
	Academic/Gov/Non-profit	$86K
Europe (43)	**Company/Self**	$82K
	Academic/Gov/Non-profit	$35K
Asia (14)	**Company/Self**	$59K
	Academic/Gov/Non-profit	$40K
Australia/NZ (9)	**Company/Self**	$90K
	Academic/Gov/Non-profit	$105K
Other (6)	**Company/Self**	$75K
	Academic/Gov/Non-profit	$88K

Source from: http://www.kdnuggets.com/polls/index.html

Should data scientist / data miners be responsible for their predictions?



Should data scientists / data miners be responsible for their predictions? [242 voters]	
No, they should not be responsible (108)	45%
Not sure (32)	13%
They can be held financially responsible, but if they also benefit from correct predictions (89)	37%
They can be held criminally responsible for wrong predictions (13)	5%

Source from: http://www.kdnuggets.com/polls/index.html

Why Python?

- **Why Python?**

  Not

Think about the scientist's needs:

- Get data (simulation, experiment control)
- Manipulate and process data.
- Visualize results... to understand what we are doing!
- Communicate results: produce figures for reports or publications, write presentations.

Why Python?

- **Why Python?**

  Not

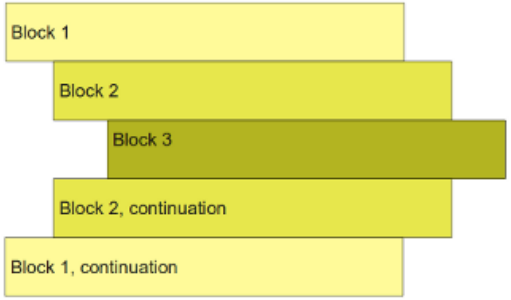
  - Easy
    - Easy to learn, easily readable
    - Scientists first, programmers second
  - Efficient
    - Managing memory is easy – if you just don't care
  - A single Language for everything
    - Avoid learning a new software for each new problem

More to Take Away

- Free distribution from http://www.python.org
- Known for it's "batteries included" philosophy

  Similar to R, Python has a fantastic community around it and, luckily for you, this community can write

- Two popular versions, 2.7 or 3.x
- A single-click installer: Enthought Canopy
- Prepare yourself for code indentation heaven



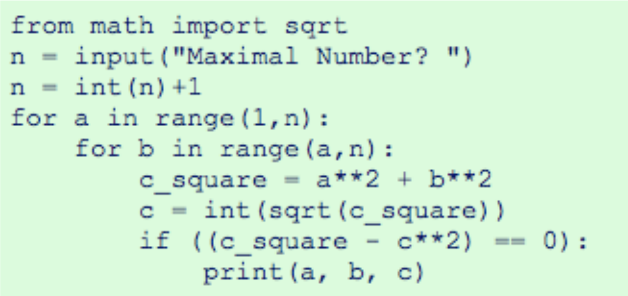
Block 1

Block 2

Block 3

Block 2, continuation

Block 1, continuation



```
from math import sqrt
n = input("Maximal Number? ")
n = int(n)+1
for a in range(1,n):
    for b in range(a,n):
        c_square = a**2 + b**2
        c = int(sqrt(c_square))
        if ((c_square - c**2) == 0):
            print(a, b, c)
```

All the Good Modules

- **numpy, scipy**: basics for almost everything
- **Matplotlib**, a Python 2D plotting library http://matplotlib.org
- **NLTK**, Natual Language Toolkit http://www.nltk.org
- **Pandas**, Python Data Analysis Library http://pandas.pydata.org
- **mrjob**, route to writing MapReduce jobs https://pythonhosted.org/mrjob/
- **IPython,** Interactive console with IDE-like features http://ipython.org
- **Scikit-Learn**, ML resource and library http://scikit-learn.org/dev/index.html
- **Theano/Pylearn2**, deep learning

  http://deeplearning.net/software/theano/

  http://deeplearning.net/software/pylearn2/
- More: mlpy, PyBrain, Orange, Scrapy, …

Outline

- Why Python?
- **Intro to Python**
- Intro to Scikit-Learn
- Unsupervised Learning
  - Demo on PCA, K-Means
- Supervised Learning
  - Demo on Linear Regression, Logistic Regression

The Use of Python: Simple demos

0 – Python Intro.ipynb

Outline

- Why Python?
- Intro to Python
- **Intro to Scikit-Learn**
- Unsupervised Learning
  - Demo on PCA, K-Means
- Supervised Learning
  - Demo on Linear Regression, Logistic Regression

What is Scikit-learn

- A Python Machine Learning Library
- Focused on modeling data
- Developed by David Cournapeau as a Google summer of code project in 2007.
- First public release (v0.1 beta) published in late January 2010.
- Now has more than 30 active contributors and has had paid sponsorship from INRIA, Google, Tinyclues and the Python Software Foundation.
- The library is built upon the SciPy that must be installed before you can use scikit-learn.

Outline

- Why Python?
- Intro to Python
- Intro to Scikit-Learn
- **Unsupervised Learning**
  - Demo on PCA, K-Means
- Supervised Learning
  - Demo on Linear Regression, Logistic Regression

The use of Scikit-Learn: unsupervised learning demos

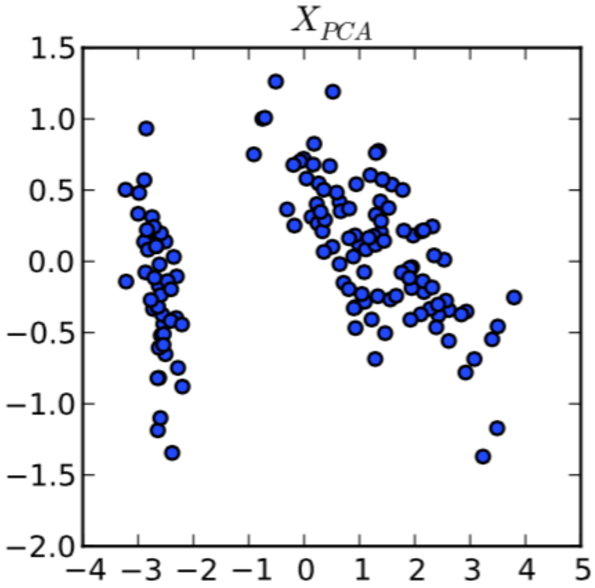
PCA Summary

- PCA projects to axis with greatest variance
- Often provides good first insight into dataset

$$\bar{X} \leftarrow X - \mathrm{mean}(X) \qquad \bar{X} \in \mathbb{R}^{n \times N}$$

$$W \leftarrow \mathrm{PCA}(\bar{X}, 2) \qquad W \in \mathbb{R}^{N \times M}$$

$$X_{\mathrm{PCA}} \leftarrow \bar{X} \cdot W \qquad X_{\mathrm{PCA}} \in \mathbb{R}^{n \times M}$$

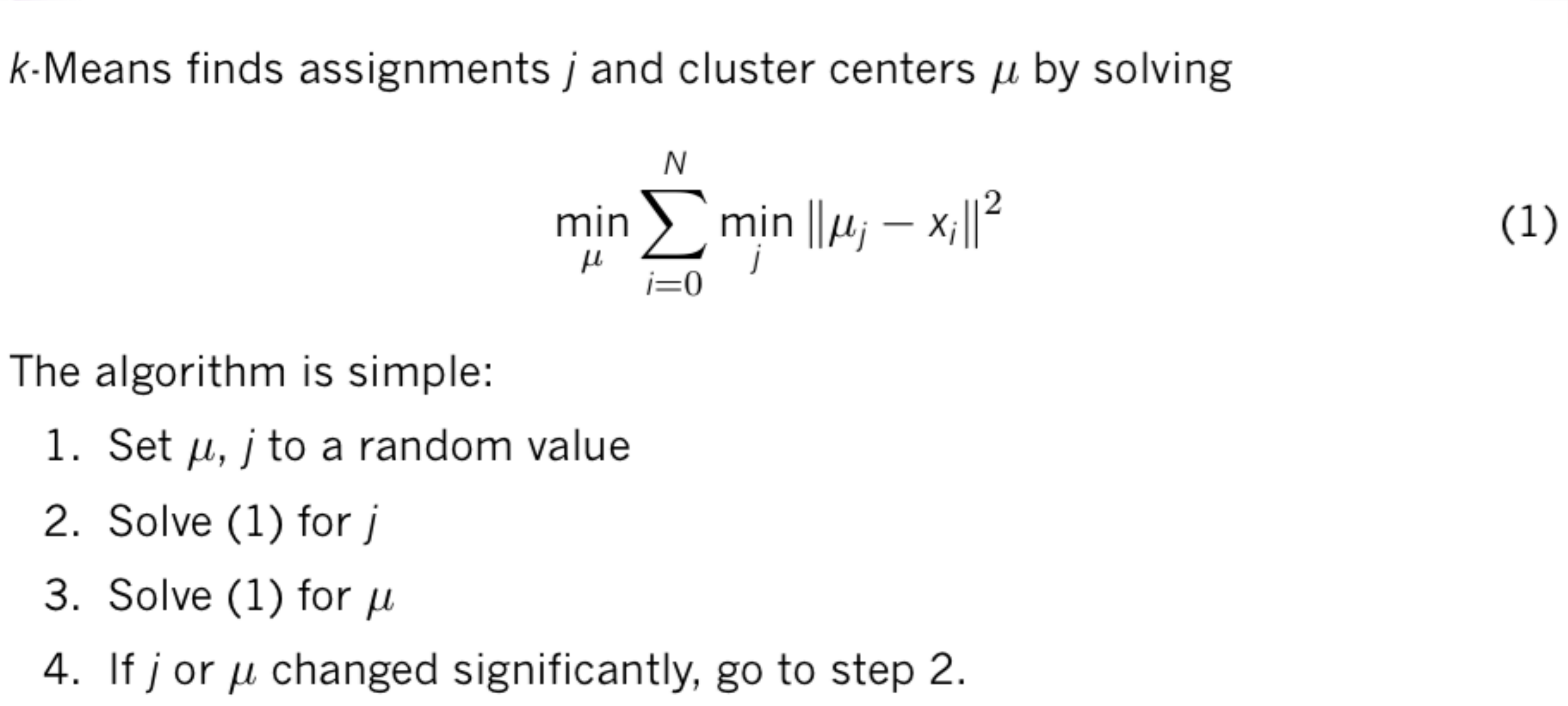


- Identify important variables in projection matrix *W*:

```
W = [[ 0.36 -0.08 0.85 0.35]
     [-0.65 -0.72 0.17 0.07]]
```

1 – PCA.ipynb

K-Means Algorithm



k-Means finds assignments *j* and cluster centers μ by solving

$$\min_{\mu} \sum_{i=0}^{N} \min_{j} \|\mu_j - x_i\|^2 \tag{1}$$

The algorithm is simple:

1. Set μ, *j* to a random value
2. Solve (1) for *j*
3. Solve (1) for μ
4. If *j* or μ changed significantly, go to step 2.

2 – k means.ipynb

Outline

- Why Python?
- Intro to Python
- Intro to Scikit-Learn
- Unsupervised Learning
  - Demo on PCA, K-Means
- **Supervised Learning**
  - Demo on Linear Regression, Logistic Regression, kNN

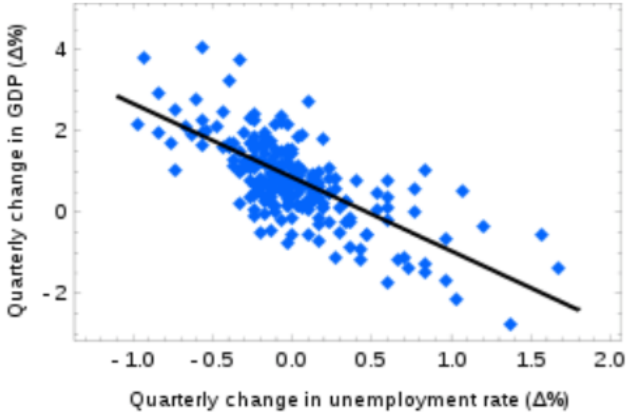
The use of Scikit-Learn: supervised learning demos

NORTHWESTERN
UNIVERSITY

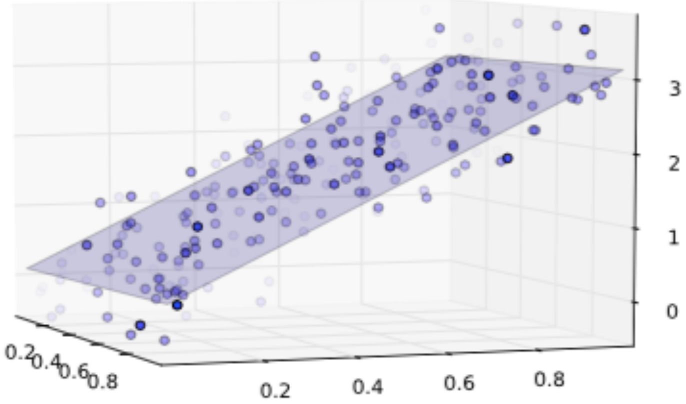
Linear Regression

$y = w_1 x_1 + b$

$y = w_2 x_2 + w_1 x_1 + b$



1D



2D

To find w and b, minimize the error:

$$E = \sum_{i=0}^{N} (y_i - (w_i x_i + b))^2$$

3 – LinearRegression1.ipynb

3 – LinearRegression2.ipynb

Logistic Regression

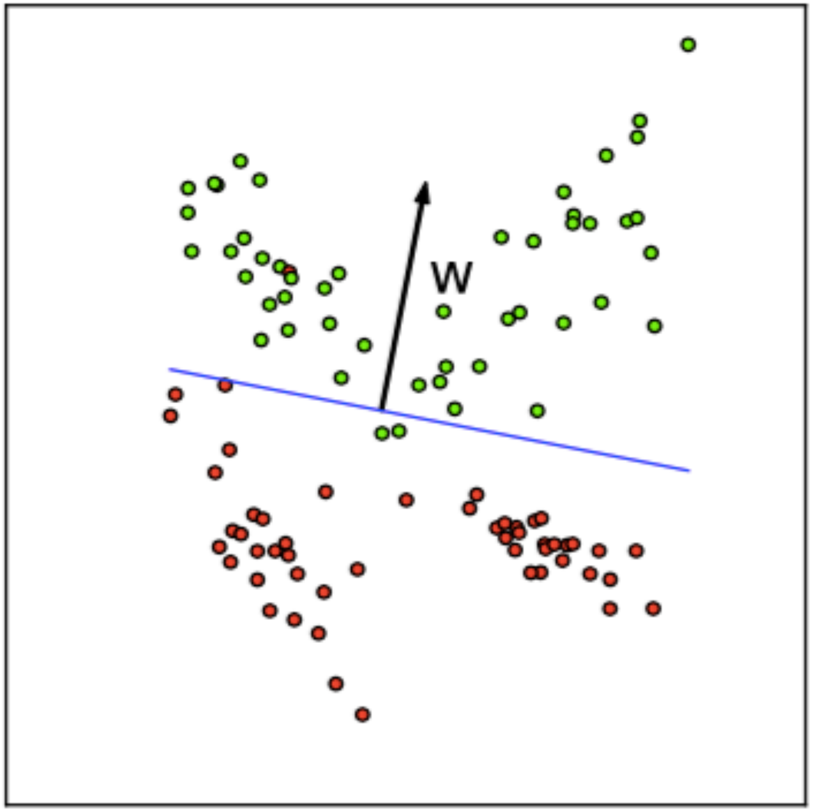
For two classes $-1, +1$.

Decision boundary given by hyperplane.

Hyperplane defined by normal vector and offset:

$$y = \text{sign}(\langle w, x\rangle + b)$$

$$w \in \mathbb{R}^n, b \in \mathbb{R}$$

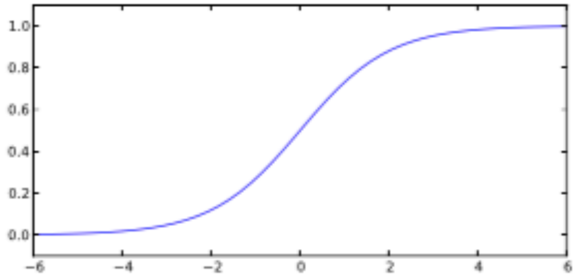


NORTHWESTERN
UNIVERSITY

Logistic Regression

Relation to regression:

$$p(y = +1 \,|\, x) = \text{logistic}(\langle w, x \rangle + b)$$



As probabilities are between 0 and 1, the logistic function squashes the regression result:

$$p(y = +1 \,|\, x) > 0.5 \Leftrightarrow \langle w, x \rangle + b > 0$$

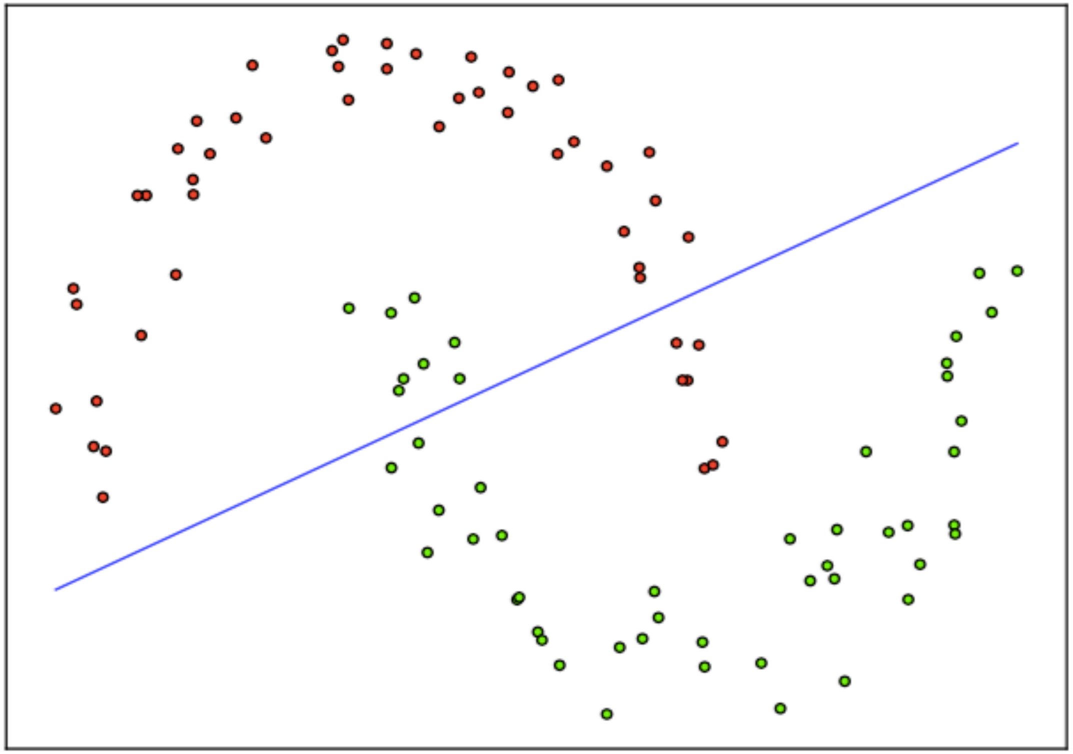
Need to solve:

$$\max_{w} \sum_{i=0}^{n} \log(p(Y = y_i | x_i))$$

4 – LogisticRegression.ipynb

Nonlinear Problems

- Logistic regression works well if the data is linearly separable, but…



K Nearest Neighbors

- Classification: same setup as logistic regression.
- Very simple but powerful idea: Do as your neighbors do.
- For a new point *x* look at the nearest (or the two nearest or three nearest, ...) point(s) in the training data for a label.
- Usual distance measure: Euclidean distance

Simple Algorithm

- Pick a k, for example k = 3.
- Want to classify new example x.
- Compute $d_i = d(x_i, x)$, i.e. $d(x_i, x) = ||x_i - x||$.
- Sort d_i , take k smallest: d_{i0} , d_{i1} , d_{i2} .
- Assign y that appears most often among y_{i0} , y_{i1} , y_{i2} .

5 – kNN.ipynb