Machine Learning

Introduction, Data, Modeling

Jeff Abrahamson

9, 16 décembre 2016 13 janvier, 9 mars 2017

Supervised

Unsupervised

Reinforcement

Course structure

Ten afternoons:

- 8, 22 Nov
- 6, Dec
- 10, 24 Jan
- 14, 28 Feb
- 14, 28 Mar
- 11 Apr

Course structure

- Email `jeff@p27.eu`
- Mailing list:
- Github: `https://github.com/mlweek/MLWeek_v3`

Curse of Dimensionality

Machine learning is not magic

Machine learning is mathematics

Mostly, it's these maths:

- Probability
- Statistics
- Linear algebra
- Optimisation theory
- Differential calculus

Unless you want to, we'll skip the maths.

Probability

events

- independent
- dependent

Statistics

What is Statistics

1. Identify a question or problem.
2. Collect relevant data on the topic.
3. Analyze the data.
4. Form a conclusion.

What is Statistics

1. Identify a question or problem.
2. Collect relevant data on the topic.
3. Analyze the data.
4. Form a conclusion.

Sadly, sometimes people forget 1.

What is Statistics

1. Identify a question or problem.
2. Collect relevant data on the topic.
3. Analyze the data.
4. Form a conclusion.

Statistics is about making 2–4 efficient, rigorous, and meaningful.

OpenIntro Statistics, 2nd edition, D. Diez, C. Barr, M. Çetinkaya-Rundel, 2013.

What is data science?

(Exercise: Is this the same question as the last slide?)

1. Define the question of interest
2. Get the data
3. Clean the data
4. Explore the data
5. Fit statistical models
6. Communicate the results
7. Make your analysis reproducible

What is data science?

(Exercise: Is this the same question as the last slide?)

1. Define the question of interest
2. Get the data
3. Clean the data
4. Explore the data
5. Fit statistical models
6. Communicate the results
7. Make your analysis reproducible

What the public thinks.

What is data science?

(Exercise: Is this the same question as the last slide?)

1. Define the question of interest
2. Get the data
3. Clean the data
4. Explore the data
5. Fit statistical models
6. Communicate the results
7. Make your analysis reproducible

Where we spend most of our time.

What is data science?

(Exercise: Is this the same question as the last slide?)

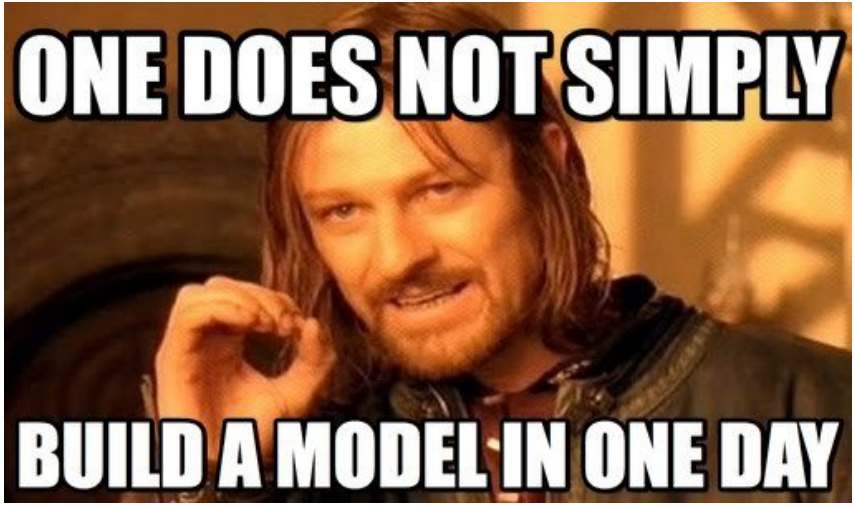
1. Define the question of interest
2. Get the data
3. Clean the data
4. Explore the data
5. Fit statistical models
6. Communicate the results
7. Make your analysis reproducible

The easiest part to forget.

What is data science?

http://simplystatistics.org/2015/03/17/data-science-done-well-looks-easy-and-that-is-a-big-problem-for-data-scientists/

What is data science?



Anecdote

Some properties of anecdote:

- is data
- haphazardly collected
- is generally not representative
- sometimes result of selective retention
- does not accumulate to be representative
- might be true (by chance)
- is ok to use as hypothesis, but be clear that hypothesis is anecdote

Study Types

- Observational
- Experimental

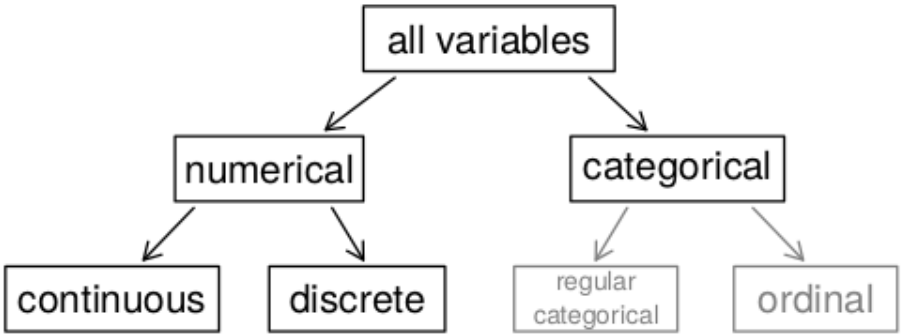
Study Types

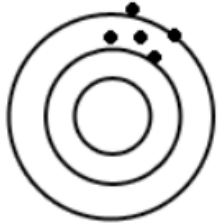
- Observational
- Experimental

What can go wrong?

- Forgetting that association $\neq$ causation
- Not random
- Confounding variables

Variable types





High bias, low variance

Low bias, high variance

High bias, high variance

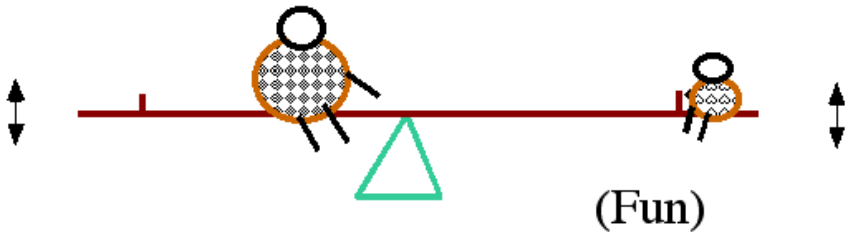
Low bias, low variance

Mean

- Weighted and unweighted
- Centroid to physicists

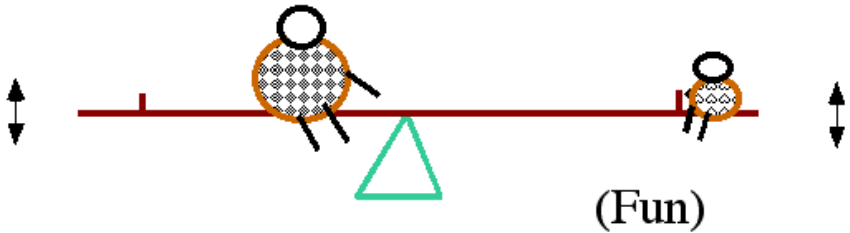
Mean

- Weighted and unweighted
- Centroid to physicists



Mean

- Weighted and unweighted
- Centroid to physicists

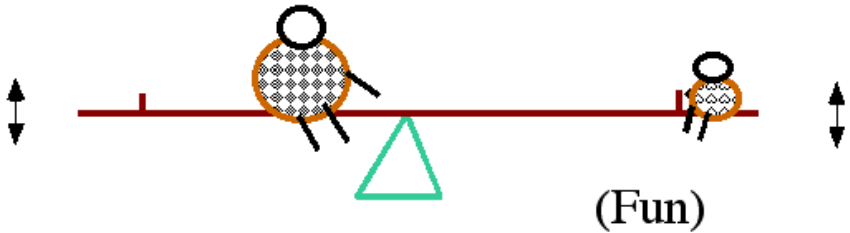


$$\mu = E(X) = \sum w_i x_i = \mathbf{w} \cdot \mathbf{x}$$

http://telescopes.stardate.org/images/research/teeter-totter/TT4.gif

Mean

- Weighted and unweighted
- Centroid to physicists



$$\mu = E(X) = \sum \Pr(X = x_i) x_i$$

http://telescopes.stardate.org/images/research/teeter-totter/TT4.gif

Mean

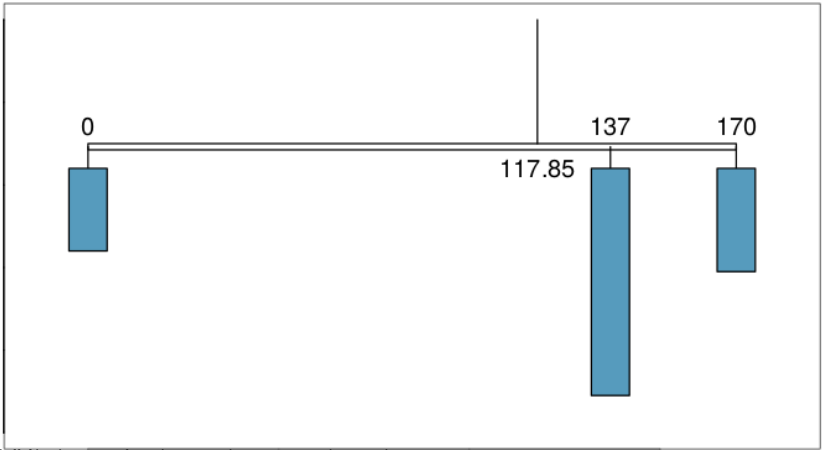
- Weighted and unweighted
- Centroid to physicists

$$\mu = E(X) = \int x f(x)\,\mathrm{d}x$$

http://telescopes.stardate.org/images/research/teeter-totter/TT4.gif

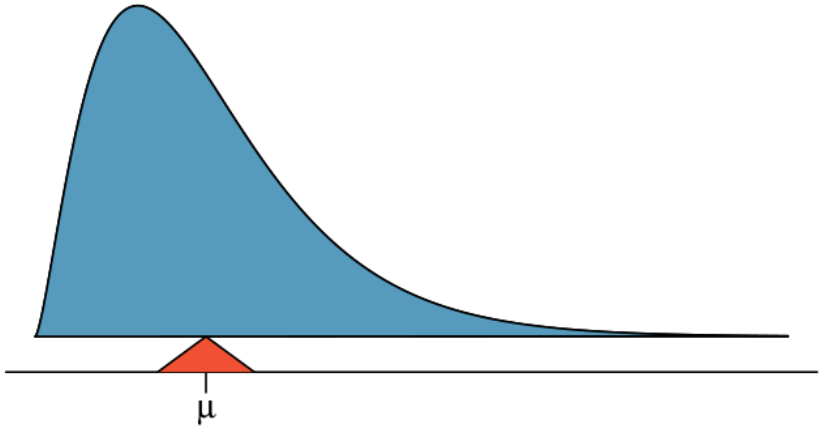
Mean

- Weighted and unweighted
- Centroid to physicists



Mean

- Weighted and unweighted
- Centroid to physicists



Population statistics

Deviation is distance from mean.

Population statistics

Variance is mean square of deviations

Population statistics

Standard deviation is square root of variance

Population statistics

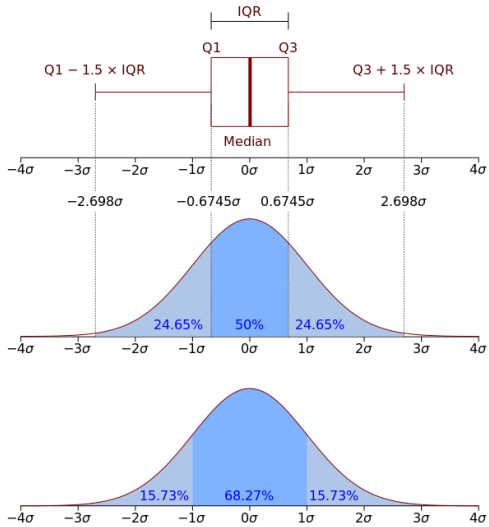
$$s^2 = \frac{(\overline{x} - x_1)^2 + \cdots (\overline{x} - x_n)^2}{n - 1}$$

Population statistics

$$\sigma^2 = \frac{(\overline{x} - x_1)^2 + \cdots (\overline{x} - x_n)^2}{n}$$

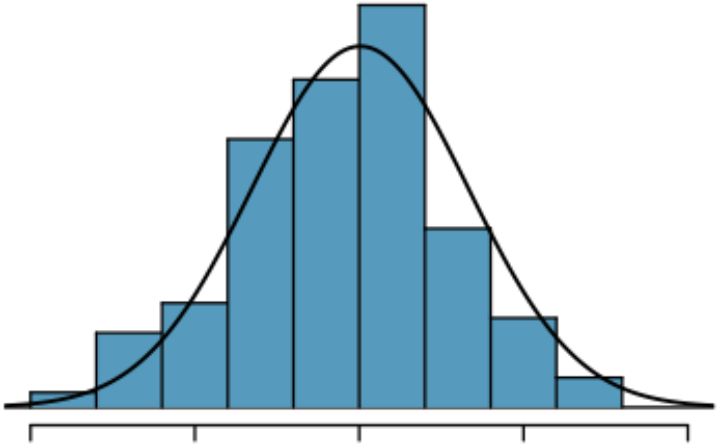
Population statistics

$$\mathrm{Var}(X) = \sigma^2 = (\overline{x} - x_1)^2 \Pr(X = x_1) + \cdots (\overline{x} - x_n)^2 \Pr(X = x_n)$$



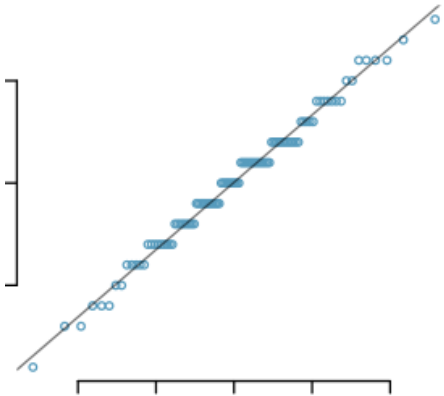
Evaluating Normal Approximations

Easy technique 1: visually compare to normal plot.



Evaluating Normal Approximations

Easy technique 2: normal probability plot.



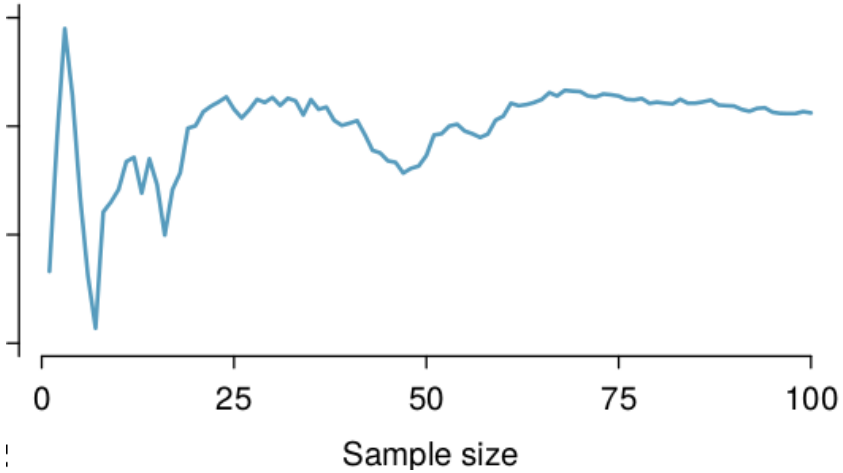
Also known as a quantile-quantile plot.

$$\overline{x} \neq \mu$$

Inference Concepts

Running mean. Sequence of partial sums (divided by number in sum).

Running mean. Sequence of partial sums (divided by number in sum).



Inference Concepts

Running mean. Sequence of partial sums (divided by number in sum).

Sampling variation. Change of $\overline{x}$ from one sample to the next.

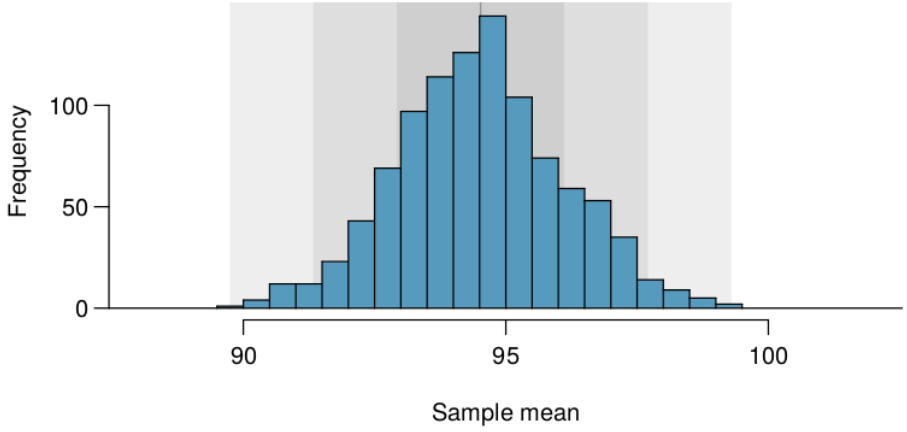
Inference Concepts

Running mean. Sequence of partial sums (divided by number in sum).

Sampling variation. Change of $\overline{x}$ from one sample to the next.

Sampling distribution. The distribution of possible point samples of a fixed size from a given population.

Sampling distribution



Confidence intervals

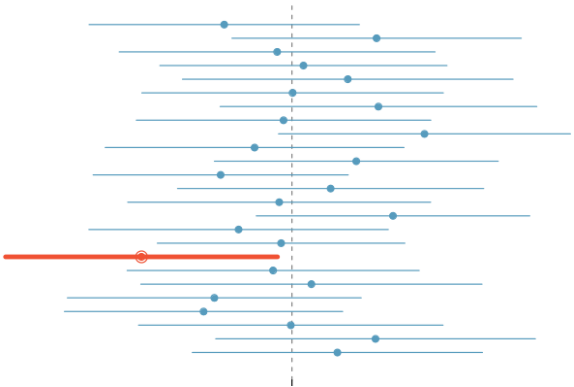
Sample n points, choose an interval around the sample mean.

A 95% confidence interval means if we sample repeatedly, about 95% of the samples will contain the population mean.

Confidence intervals

Sample n points, choose an interval around the sample mean.

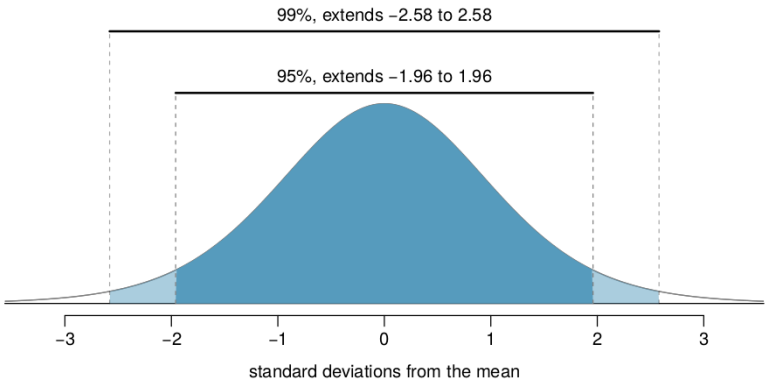
A 95% confidence interval means if we sample repeatedly, about 95% of the samples will contain the population mean.



Confidence intervals

Sample n points, choose an interval around the sample mean.

A 95% confidence interval means if we sample repeatedly, about 95% of the samples will contain the population mean.



Linear Algebra

Linear algebra: basics

$$v = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \in \mathbb{R}^n$$

Linear algebra: basics

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{bmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{pmatrix}$$

$$= \begin{Bmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{Bmatrix} \in \mathbb{R}^{n \times n}$$

Linear algebra: basics

$$u + v = \begin{pmatrix} u_1 + v_1 \\ u_2 + v_2 \\ \vdots \\ u_n + v_n \end{pmatrix}$$

Linear algebra: basics

$$\alpha \boldsymbol{v} = \begin{pmatrix} \alpha v_1 \\ \alpha v_2 \\ \vdots \\ \alpha v_n \end{pmatrix} \qquad (\alpha \in \mathbb{R})$$

Linear algebra: basics

$$\| v \| = \sqrt{v_1^2 + \cdots + v_n^2}$$

Linear algebra: basics

$$u \cdot v = u_1 \cdot v_1 + \cdots + u_n \cdot v_n$$

$$= \| u \| \| v \| \cos\theta$$

Linear algebra: basics

$$C = A + B \iff c_{ij} = a_{ij} + b_{ij}$$

$$C = AB \iff c_{ij} = \sum_k a_{ik} b_{kj}$$

$$A = B^T \iff a_{ij} = b_{ji}$$

$$AA^{-1} = A^{-1}A = \mathrm{diag}(1)$$

Linear algebra: transformations

$$Ax = y \qquad f = T_A : \mathbb{R}^n \to \mathbb{R}^n$$

$$x = A^{-1}Ax = A^{-1}y \qquad f^{-1} = T_{A^{-1}} : \mathbb{R}^n \to \mathbb{R}^n$$

Linear algebra: transformations

B is a basis for V iff any of these conditions are met:

- B is a minimal generating set of V
- B is a maximal set of linearly independent vectors
- Every vector $v \in V$ can be expressed in a unique way as a sum of $b_i \in B$

(The conditions are equivalent.)

Linear algebra: transformations

B is a basis for V iff any of these conditions are met:

- B is a minimal generating set of V
- B is a maximal set of linearly independent vectors
- Every vector $v \in V$ can be expressed in a unique way as a sum of $b_i \in B$

(The conditions are equivalent.)

Bases are not unique.

Linear algebra: transformations

Eigenvectors, eigenvalues:

$$Av = \lambda v$$

Linear algebra: transformations

Eigenvectors, eigenvalues:

$$Av = \lambda v$$

$$Av = \lambda 1 v \iff (A - \lambda 1)v = 0$$

Linear algebra: transformations

Eigenvectors, eigenvalues:

$$Av = \lambda v$$

Some matrices are diagonalisable. Then

$$A = Q\Lambda Q^{-1} \qquad \text{with } \Lambda = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & 0 \\ 0 & 0 & \lambda_n \end{bmatrix}$$

$$\text{and } Q = \begin{bmatrix} | & & | \\ v_1 & \cdots & v_n \\ | & & | \end{bmatrix}$$

Linear algebra: transformations

video time

Features and Modeling

Vector spaces

Vector spaces

Features are dimensions

Feature extraction

Feature engineering

Feature extraction

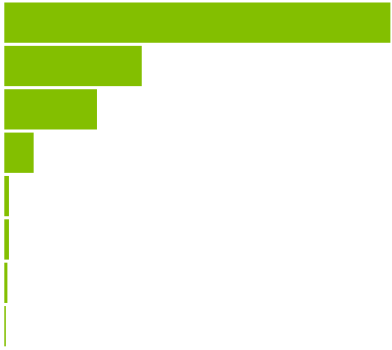
Feature engineering

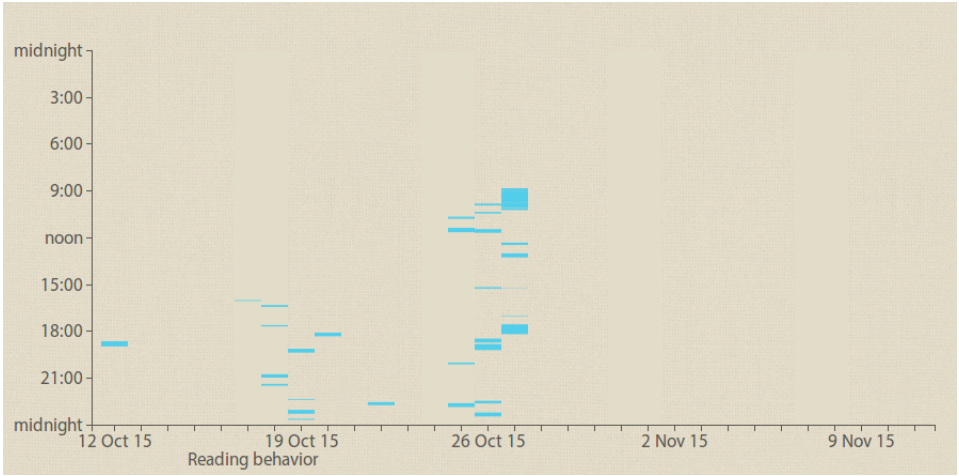
Synthetic features

Feature Engineering

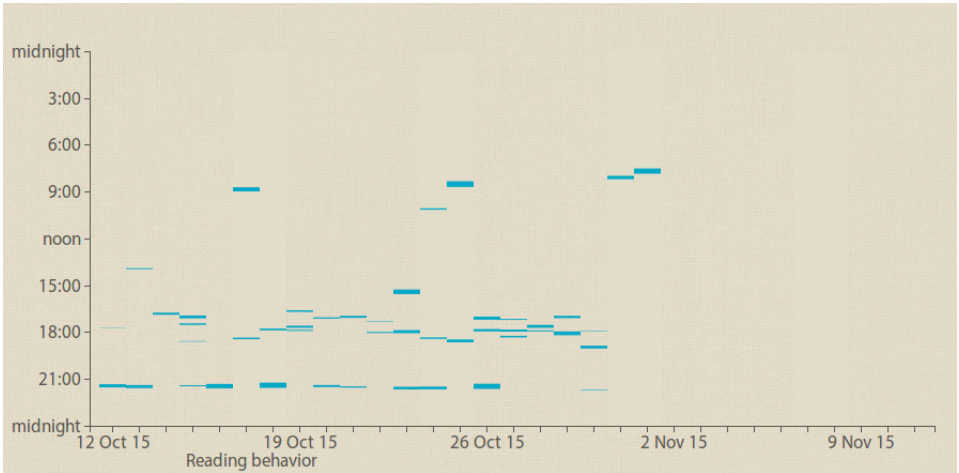
1. Brainstorm
2. Pick some
3. Make them
4. Evaluate
5. Repeat

Value	Count	Percent
Mr.	517	58.025%
Miss.	185	20.763%
Mrs.	125	14.029%
Master.	40	4.489%
Dr.	7	0.786%
Rev.	6	0.673%
Sir.	5	0.561%
Col.	2	0.224%
Jonkheer.	1	0.112%
Lady.	1	0.112%
the Countess.	1	0.112%
Ms.	1	0.112%





Jellybooks



Jellybooks

One of K = one-hot encoding

Text features

Bag of words

- Corpus (documents)
- Vocabulary (set of unique words)
- Words

Text features

Bag of words

- Order doesn't matter
- Stop words
- Stemming *(racinisation, désuffixation)*
- Lemmatisation *(transformer en lemme)*

Image features

- Corners, edges (rotation invariant, but scaling can hide)
- More complex: scale space or RNN
- Point matching is easy

Image features

Problems

- Illumination
- Scale
- Rotation
- Skew (perspective)
- Data size (matrices not sparse)

python

Useful tools

- virtualenv
- pip
- ipython
- ipython notebook
- `conda.pydata.org`

python

Notes

- `pip install -r requirements.txt`
- ipython offers tab completion (vs python)
- ipython notebook opens in a browser, caches cell output but not cell state

pandas

```
import pandas as pd
import numpy as np
import scipy
import matplotlib.pyplot as plt
```

pandas

`Dataframe` has many constructors. For example,

```
In [5]: pd.DataFrame({ 'A' : 1.,
                       'B' : pd.Timestamp('20161209'),
                       'C' : pd.Series(1,index=list(range(4)),dtype='float32'),
                       'D' : np.array([3] * 4, dtype='int32'),
                       'E' : pd.Categorical(["test","train","test","train"]),
                       'F' : 'hello' })
Out[5]:
   A          B  C  D      E      F
0  1 2016-12-09  1  3   test  hello
1  1 2016-12-09  1  3  train  hello
2  1 2016-12-09  1  3   test  hello
3  1 2016-12-09  1  3  train  hello

In [6]:
```

Viewing data

```
In [16]: dates = pd.date_range('20161209', periods=4, freq='1w')

In [17]: df = pd.DataFrame(np.random.randn(4,5), index=dates,
                           columns=list('ABCDE'))

In [18]: df.head()
Out[18]:
                   A         B         C         D         E
2016-12-11 -1.303610 -1.235823  0.621914  0.379340 -0.326934
2016-12-18 -1.218197 -1.113826  0.546314 -0.255001 -0.135573
2016-12-25 -0.124625  0.337268 -0.406295  0.587049 -0.904906
2017-01-01 -0.283182 -0.866213  0.051509  0.693037 -0.661055

In [19]:
```

Basic data exploration

```
In [19]: df.describe()
Out[19]:
              A         B         C         D         E
count  4.000000  4.000000  4.000000  4.000000  4.000000
mean  -0.732403 -0.719648  0.203361  0.351106 -0.507117
std    0.614672  0.721194  0.478728  0.424558  0.342755
min   -1.303610 -1.235823 -0.406295 -0.255001 -0.904906
25%   -1.239550 -1.144325 -0.062942  0.220755 -0.722018
50%   -0.750689 -0.990019  0.298912  0.483195 -0.493995
75%   -0.243543 -0.565343  0.565214  0.613546 -0.279094
max   -0.124625  0.337268  0.621914  0.693037 -0.135573

In [20]:
```

Select a column (series)

```
In [20]: df.loc[dates[1]]
Out[20]:
A   -1.218197
B   -1.113826
C    0.546314
D   -0.255001
E   -0.135573
Name: 2016-12-18 00:00:00, dtype: float64

In [21]:
```

Select a range

```
In [21]: df.loc[:, ['A', 'C']]
Out[21]:
                   A         C
2016-12-11 -1.303610  0.621914
2016-12-18 -1.218197  0.546314
2016-12-25 -0.124625 -0.406295
2017-01-01 -0.283182  0.051509

In [22]:
```

Boolean selection criteria

```
In [23]: df[df.D > 0]
Out[23]:
                    A         B         C         D         E
2016-12-11 -1.303610 -1.235823  0.621914  0.379340 -0.326934
2016-12-25 -0.124625  0.337268 -0.406295  0.587049 -0.904906
2017-01-01 -0.283182 -0.866213  0.051509  0.693037 -0.661055

In [24]:
```

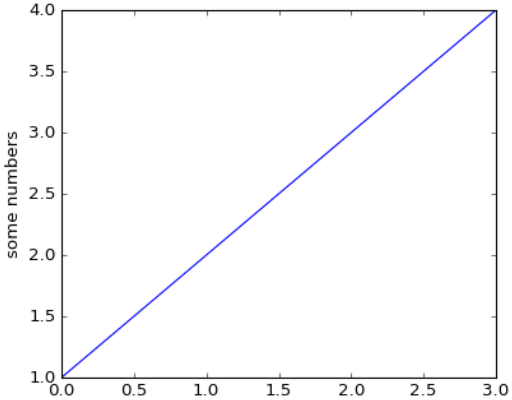
Recommended

http://www.gregreda.com/2013/10/26/intro-to-pandas-data-structures/

Plotting

Draw a line

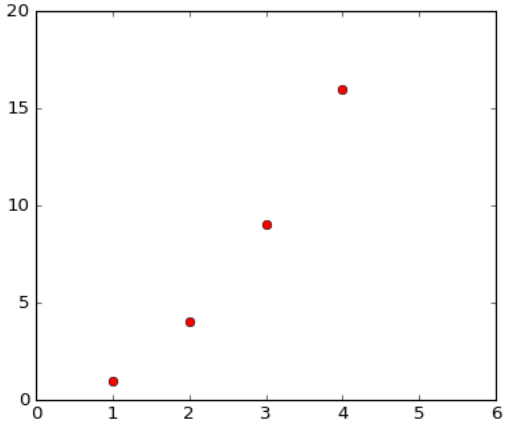
```
import matplotlib.pyplot as plt
plt.plot([1,2,3,4])
plt.ylabel('some numbers')
plt.show()
```



Plotting

Draw a line

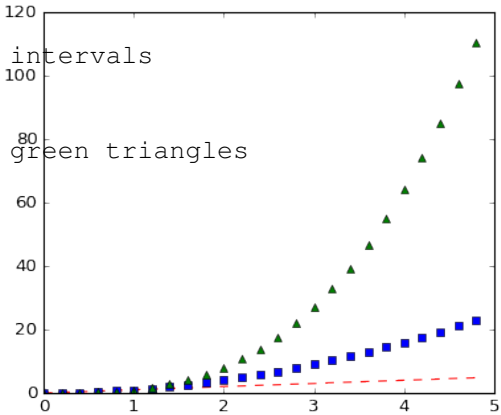
```
import matplotlib.pyplot as plt
plt.plot([1, 2, 3, 4], [1, 4, 9, 16])
plt.ylabel('some numbers')
plt.show()
```



Plotting

Draw a line

```
import numpy as np
import matplotlib.pyplot as plt

# evenly sampled time at 200ms intervals
t = np.arange(0., 5., 0.2)

# red dashes, blue squares and green triangles
plt.plot(t, t,
         'r--', t,
         t**2, 'bs',
         t, t**3, 'g^')
plt.show()
```

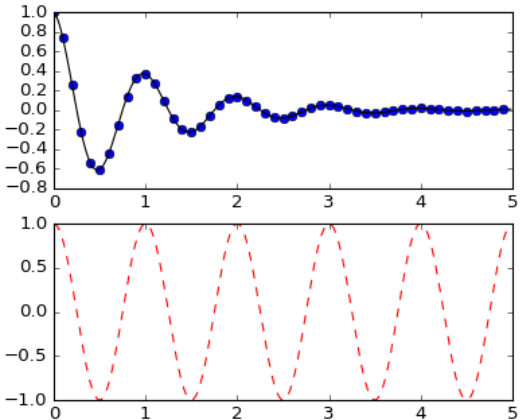


Plotting

Draw two curves

```
import numpy as np
import matplotlib.pyplot as plt

def f(t):
    return np.exp(-t) * np.cos(2*np.pi*t)

t1 = np.arange(0.0, 5.0, 0.1)
t2 = np.arange(0.0, 5.0, 0.02)

plt.figure(1)
plt.subplot(211)
plt.plot(t1, f(t1), 'bo', t2, f(t2), 'k')

plt.subplot(212)
plt.plot(t2, np.cos(2*np.pi*t2), 'r--')
plt.show()
```

Plotting

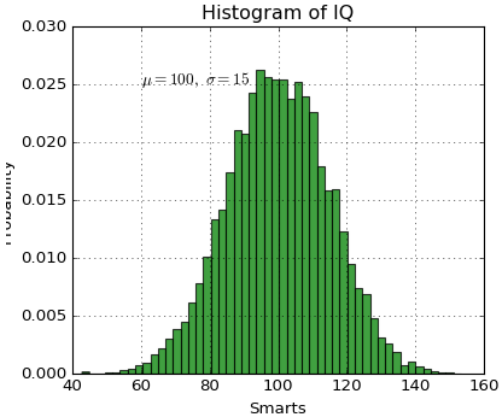


Plotting

Draw two curves

```
import numpy as np
import matplotlib.pyplot as plt

mu, sigma = 100, 15
x = mu + sigma * np.random.randn(10000)

n, bins, patches = plt.hist(x, 50, normed=1, facecolor='g', alpha=0.75)

plt.xlabel('Smarts')
plt.ylabel('Probability')
plt.title('Histogram of IQ')
plt.text(60, .025, r'$\mu=100,\ \sigma=15$')
plt.axis([40, 160, 0, 0.03])
plt.grid(True)
plt.show()
```

Plotting

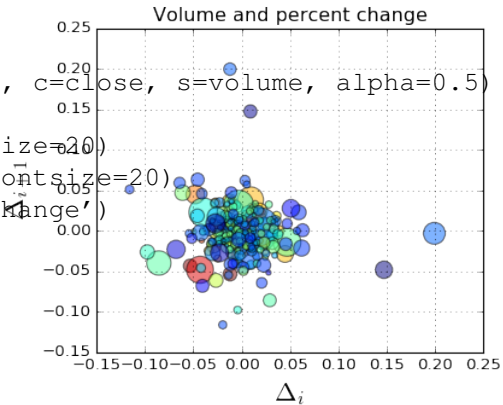


Plotting

Scatter plot

http://matplotlib.org/mpl_examples/pylab_examples/scatter_demo2.py

```
import numpy as np
import matplotlib.pyplot as plt

fig, ax = plt.subplots()
ax.scatter(delta1[:-1], delta1[1:], c=close, s=volume, alpha=0.5)

ax.set_xlabel(r'$\Delta_i$', fontsize=20)
ax.set_ylabel(r'$\Delta_{i+1}$', fontsize=20)
ax.set_title('Volume and percent change')

ax.grid(True)
fig.tight_layout()

plt.show()
```



Plotting

http://matplotlib.org/users/pyplot_tutorial.html

http://matplotlib.org/users/beginner.html

Questions?