# Learning from Data for Linguists
# Lecture 1: Introduction and Setup

Malvina Nissim and Johannes Bjerva
m.nissim@rug.nl, j.bjerva@rug.nl

22 August 2016

- **Instructors**:
  - Malvina Nissim: `m.nissim@rug.nl`
  - Johannes Bjerva: `j.bjerva@rug.nl`
- **Schedule**:
  - 17:00–18:30
  - both theory and practice

# Take home message and skills

- basic knowledge on what learning from data means and how it works
- general settings and procedures
- main, classic, algorithms
- tools to run your own experiments on your own datasets

# Live poll

http://etc.ch/dE36

# Live poll – results

http://directpoll.com/r?
XDbzPBd3ixYqg8JtyeQnN7sHzV0fIrJoNZMoCc3lBd

# Learning from Data

Learning from Data

learning what?

Learning from Data

what data?

learning to **predict**

# Prediction

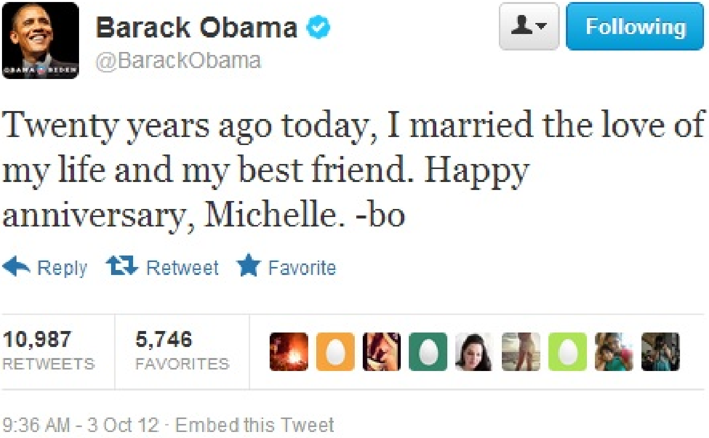
you are given some object — you have to **make a prediction**:

- is today a good day for playing football?
- is this tweet positive or negative?
- is the fourth word in this sentence a verb?
- is this article about the New York marathon?
- does this image contain a train?

# Prediction

you are given some object — you have to **make a prediction**:

- is today a good day for playing football?
- is this tweet positive or negative?



Barack Obama

@BarackObama

Twenty years ago today, I married the love of my life and my best friend. Happy anniversary, Michelle. -bo

10,987 RETWEETS 5,746 FAVORITES

9:36 AM - 3 Oct 12 · Embed this Tweet

# Prediction

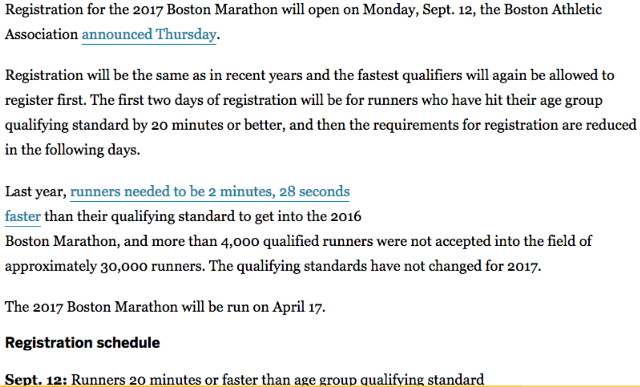
you are given some object — you have to **make a prediction**:

- is today a good day for playing football?
- is this tweet positive or negative?
- is the fourth word in this sentence a verb?
- is this article about the New York marathon?
- does this image contain a train?

# Prediction

you are given some object — you have to **make a prediction**:

- is today a good day for playing football?
- is this tweet positive or negative?
- is the fourth word in this sentence a verb?
- is this article about the New York marathon?



Registration for the 2017 Boston Marathon will open on Monday, Sept. 12, the Boston Athletic Association announced Thursday.

Registration will be the same as in recent years and the fastest qualifiers will again be allowed to register first. The first two days of registration will be for runners who have hit their age group qualifying standard by 20 minutes or better, and then the requirements for registration are reduced in the following days.

Last year, runners needed to be 2 minutes, 28 seconds faster than their qualifying standard to get into the 2016 Boston Marathon, and more than 4,000 qualified runners were not accepted into the field of approximately 30,000 runners. The qualifying standards have not changed for 2017.

The 2017 Boston Marathon will be run on April 17.

**Registration schedule**

**Sept. 12:** Runners 20 minutes or faster than age group qualifying standard

# Prediction

you are given some object — you have to **make a prediction**:

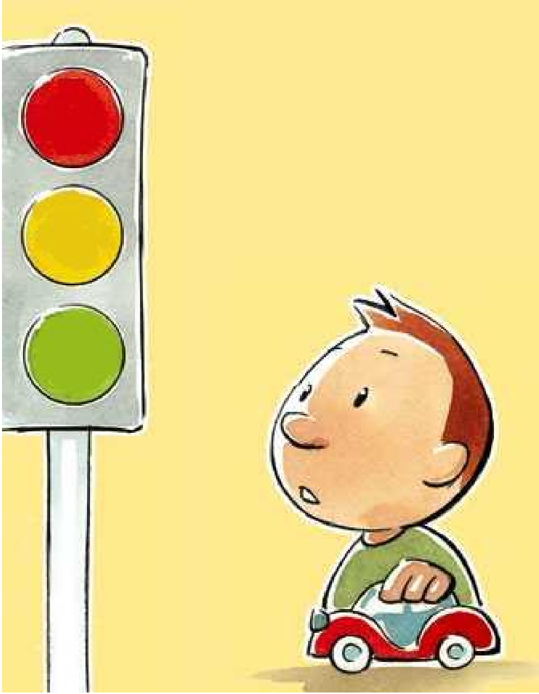
- is today a good day for playing football?
- is this tweet positive or negative?
- is the fourth word in this sentence a verb?
- is this article about the New York marathon?
- does this image contain a train?

# Prediction

you are given some object — you have to **make a prediction**:

- is today a good day for playing football?
- is this tweet positive or negative?
- is the fourth word in this sentence a verb?
- is this article about the New York marathon?
- does this image contain a train?

$$\boxed{\text{learning = making such predictions by } \textbf{observing data}}$$



# What to do in front of a traffic light?

$\boxed{\text{STOP}}$ or $\boxed{\text{GO}}$ ?

# What to do in front of a traffic light?

$\boxed{\text{STOP}}$ or $\boxed{\text{GO}}$ ?

Options to teach the appropriate behaviour:

- create a set of *ad hoc rules*, as exhaustive as possible
- collect a set of *real examples* of people's behaviour at a traffic light

# What to do in front of a traffic light?

$\boxed{\text{STOP}}$ or $\boxed{\text{GO}}$ ?

Options to teach the appropriate behaviour:

- create a set of *ad hoc rules*, as exhaustive as possible
- collect a set of *real examples* of people's behaviour at a traffic light

- rules:
  - *if* the light is red, *then* stop
  - *if* the light is green, *then* go
  - *if* the light is yellow, *then* if . . .

# What to do in front of a traffic light?

$$\boxed{\text{STOP}} \text{ or } \boxed{\text{GO}} \text{ ?}$$

Options to teach the appropriate behaviour:

- create a set of *ad hoc rules*, as exhaustive as possible
- collect a set of *real examples* of people's behaviour at a traffic light

- examples:
  - collection of examples of behaviour at a traffic light
  - cases are characterised by
    - a set of **features** (light colour, speed, distance from traffic light, . . . )
    - and a **result** (stop, go)
  - induction and generalisation from observed examples

why do we want to **build** a predicting function from the examples rather than just implementing it?

why do we want to **build** a predicting function from the examples rather than just implementing it?

- often we don't know how to write down the function
- often a hand-written function isn't complete
- what is more expensive here: (acquiring accurate) knowledge or data?

- we have a set of examples and we want to obtain an inference scheme to model our data: we want to **generalise**
- our model is **general enough** if it can describe yet unseen examples (with an acceptable error rate)

learning from data = inferring what we don't know from what we know

# A classic: Text classification

Text classification:

- topic classification
- spam detection
- authorship identification
- author profiling (age, gender, etc)
- sentiment analysis
- . . .

# A classic: Text classification

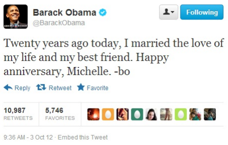
input:

- a document $d$
- a fixed set of classes $C = \{c_1, c_2, ..., c_n\}$

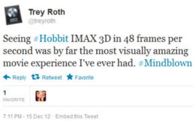
output:

- a predicted class $c \in C$

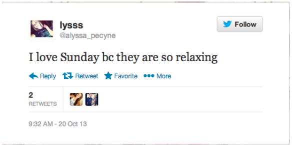
# Learning from examples


Barack Obama
@BarackObama

Twenty years ago today, I married the love of my life and my best friend. Happy anniversary, Michelle. -bo

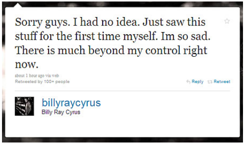

Trey Roth
@treyroth

Seeing #Hobbit IMAX 3D in 48 frames per second was by far the most visually amazing movie experience I've ever had. #Mindblown

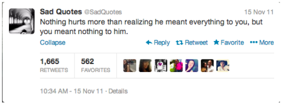

lysss
@alyssa_pecyne

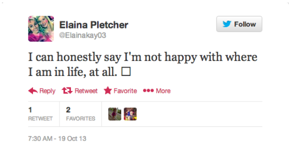
I love Sunday bc they are so relaxing

positive


Sorry guys. I had no idea. Just saw this stuff for the first time myself. Im so sad. There is much beyond my control right now.

billyraycyrus
Billy Ray Cyrus


Sad Quotes @SadQuotes 15 Nov 11
Nothing hurts more than realizing he meant everything to you, but you meant nothing to him.

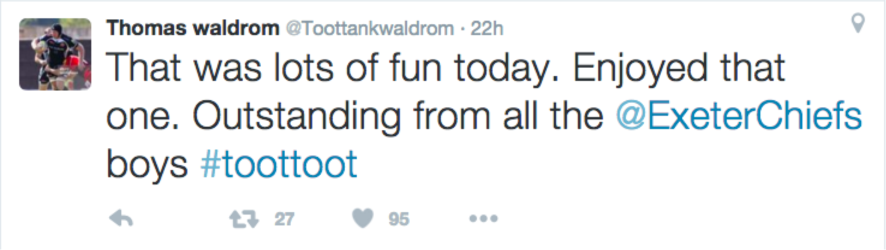

Elaina Pletcher
@Elainakay03

I can honestly say I'm not happy with where I am in life, at all. □

negative

# Learning from examples

**predict**:



Thomas waldrom @Toottankwaldrom · 22h

That was lots of fun today. Enjoyed that one. Outstanding from all the @ExeterChiefs boys #toottoot

[positive] or [negative]?

“to poach”

"to poach"

- → to steal
- → to boil

$$\left.\begin{array}{l}\text{Some swindlers are trying to \textbf{poach} upon the rich preserves}\\ \text{Firms began to \textbf{poach} partners and to recruit dozens of [\ldots]}\\ \text{[\ldots] that will allow them to \textbf{poach} workers or markets}\end{array}\right\}\text{“steal”}$$

$$\left.\begin{array}{l}\text{[\ldots] fry a teaspoonful of the pate or \textbf{poach} it in [\ldots]}\\ \text{[\ldots] gently, and \textbf{poach} spoonfuls of meringue in this}\\ \text{Let them \textbf{poach} for 3 to 4 minutes}\end{array}\right\}\text{“boil”}$$

**predict:**

“I might add them to a salad, or gently grill or poach them to bring out their natural flavours.”

[to steal] or [to boil]?

# Using examples

Can we just use examples as they are? (well, no)

# Using examples

Can we just use examples as they are? (well, no)

- we need to transform examples into something a machine can understand
- we need to tell the machine what to look for,
  what the relevant aspects of the phenomenon are.

# Using examples

in other words:

- we need to turn each example into some sort of machine-readable summary of itself (choosing relevant features)
- → our examples must become vectors of feature values

what *are* relevant features?

# Clues as Features

- we know what we want to learn (target class):
  - for example: $\boxed{\text{positive}}$ or $\boxed{\text{negative}}$

## Clues as Features

- we know what we want to learn (target class):
  - for example: $\boxed{\text{positive}}$ or $\boxed{\text{negative}}$
- we have a set of examples to learn from (instances)

# Clues as Features

- we know what we want to learn (target class):
  - for example: $\boxed{\text{positive}}$ or $\boxed{\text{negative}}$
- we have a set of examples to learn from (instances)
- what clues might be useful to guess the class from the examples?

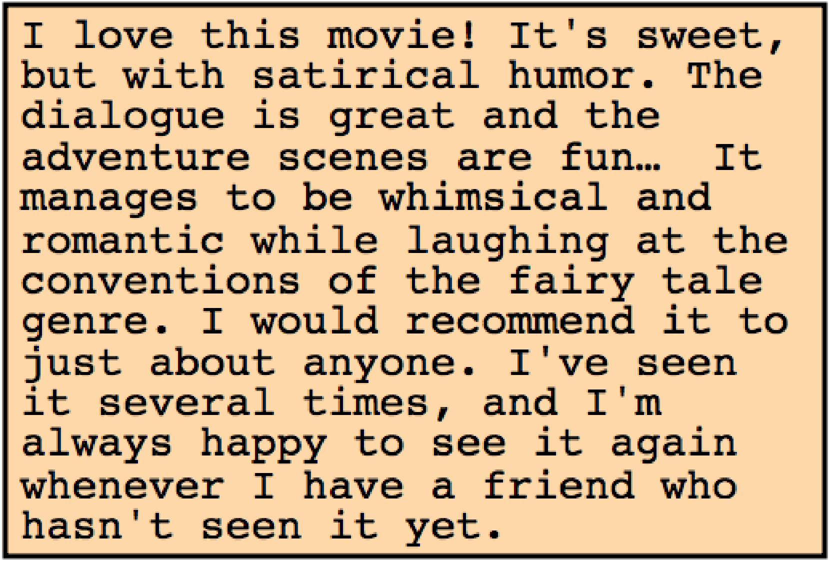
# Clues as Features

- we know what we want to learn (target class):
  - for example: $\boxed{\text{positive}}$ or $\boxed{\text{negative}}$
- we have a set of examples to learn from (instances)
- what clues might be useful to guess the class from the examples?
  - words in the text
  - types of words in the text (nouns, adjectives, adverbs, . . . )
  - (time of) day
  - id of the twitter user
  - . . .

clues → features (possible predictors)

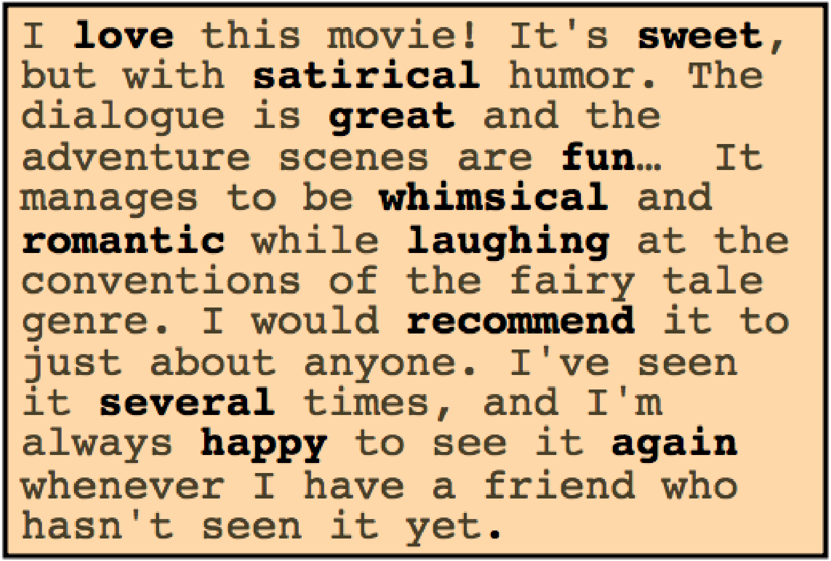
observed occurrences → feature values

# Bag of words



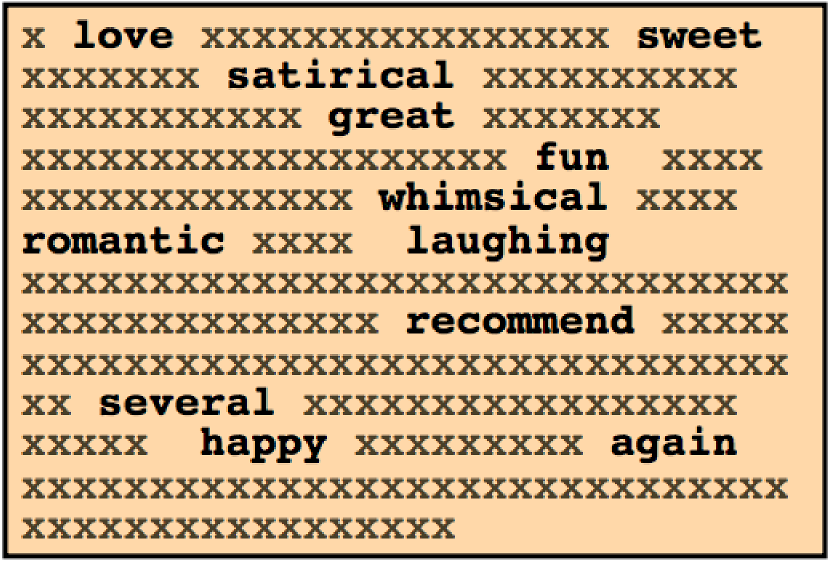
I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun… It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet.

# Bag of words



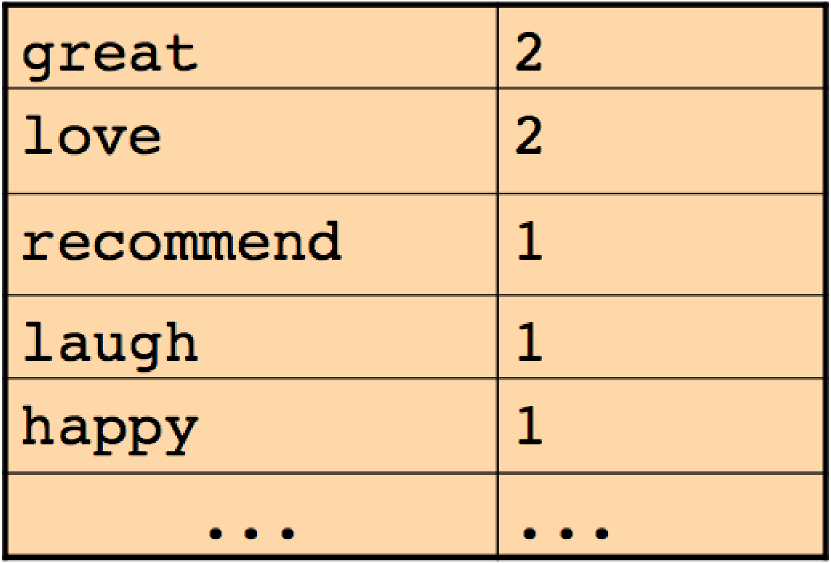
I **love** this movie! It's **sweet**, but with **satirical** humor. The dialogue is **great** and the adventure scenes are **fun...** It manages to be **whimsical** and **romantic** while **laughing** at the conventions of the fairy tale genre. I would **recommend** it to just about anyone. I've seen it **several** times, and I'm always **happy** to see it **again** whenever I have a friend who hasn't seen it yet.

# Bag of words



x love xxxxxxxxxxxxxxxx sweet
xxxxxxx satirical xxxxxxxxxx
xxxxxxxxxxx great xxxxxxx
xxxxxxxxxxxxxxxxxxx fun xxxx
xxxxxxxxxxxxx whimsical xxxx
romantic xxxx laughing
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxx recommend xxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xx several xxxxxxxxxxxxxxxxx
xxxxx happy xxxxxxxxx again
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxx

# Bag of words



| | |
|---|---|
| great | 2 |
| love | 2 |
| recommend | 1 |
| laugh | 1 |
| happy | 1 |
| ... | ... |

"Some swindlers are trying to **poach** upon the rich preserves"

```
@feature1 word-2
@feature2 word-1
@feature3 word+1
@feature4 word+2
@class {steal,boil}
```

```
@feature1 word-2
@feature2 word-1
@feature3 word+1
@feature4 word+2
@class {steal,boil}
```

- Some swindlers are trying to **poach** upon the rich preserves
- Firms began to **poach** partners and to recruit dozens of [. . . ]
- that will allow them to **poach** workers or markets
- fry a teaspoonful of the pate or **poach** it in [. . . ]
- gently, and **poach** spoonfuls of meringue in this
- Let them **poach** for 3 to 4 minutes

```
@feature1 word-2
@feature2 word-1
@feature3 word+1
@feature4 word+2
@class {steal,boil}
```

- Some swindlers are trying to **poach** upon the rich preserves (steal)

```
@feature1 word-2
@feature2 word-1
@feature3 word+1
@feature4 word+2
@class {steal,boil}
```

- Some swindlers are trying to **poach** upon the rich preserves (steal)

```
trying,
@feature1 word-2
@feature2 word-1
@feature3 word+1
@feature4 word+2
@class {steal,boil}
```

- Some swindlers are trying to **poach** upon the rich preserves (steal)

```
trying,to,
@feature1 word-2
@feature2 word-1
@feature3 word+1
@feature4 word+2
@class {steal,boil}
```

- Some swindlers are trying to **poach** upon the rich preserves (steal)

```
trying,to,upon,
@feature1 word-2
@feature2 word-1
@feature3 word+1
@feature4 word+2
@class {steal,boil}
```

- Some swindlers are trying to **poach** upon the rich preserves (steal)

```
trying,to,upon,the,
@feature1 word-2
@feature2 word-1
@feature3 word+1
@feature4 word+2
@class {steal,boil}
```

- Some swindlers are trying to **poach** upon the rich preserves (steal)

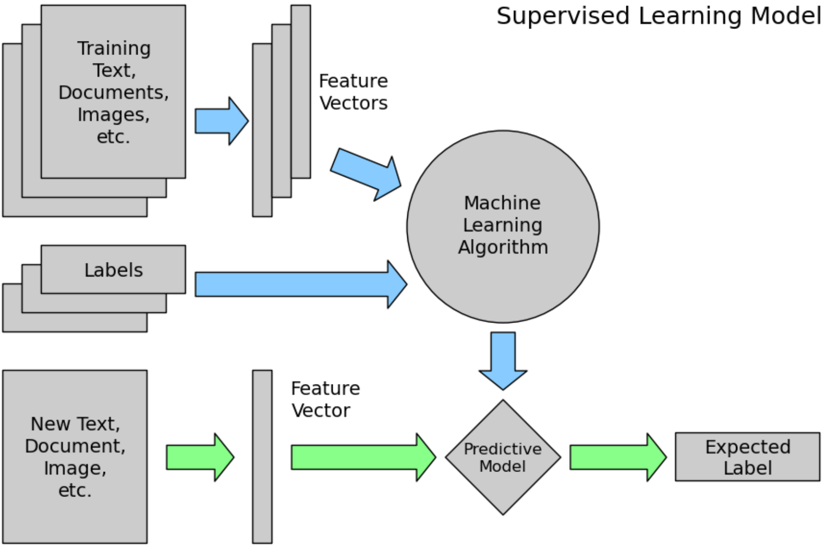
```
trying,to,upon,the,steal
```

```
@feature1 word-2 {began,gently,let,pate,them,trying}
@feature2 word-1 {and,or,them,to}
@feature3 word+1 {for,it,partners,spoonfuls,upon,workers}
@feature4 word+2 {3,and,in,of,or,the}
@class {steal,boil}

@instances

trying,to,upon,the,steal
began,to,partners,and,steal
them,to,workers,or,steal
pate,or,it,in,boil
gently,and,spoonfuls,of,boil
let,them,for,3,boil
...
```

# What happens in learning, then?

- the learning algorithm observes **given examples**
- it tries to find common patterns that explain the data: it tries to **generalise** so that predictions can be made for **new examples**
- exactly how this is done depends on what **algorithm we are using**

# What happens in learning, then?


Supervised Learning Model
Training Text, Documents, Images, etc.
Feature Vectors
Machine Learning Algorithm
Labels
New Text, Document, Image, etc.
Feature Vector
Predictive Model
Expected Label

source: `http://www.astroml.org/`

# What happens in learning, then?

- the learning algorithm observes **given examples**
- it tries to find common patterns that explain the data: it tries to **generalise** so that predictions can be made for **new examples**
- exactly how this is done depends on what **algorithm we are using**

keywords here:

- given/new examples
- generalising
- algorithm we are using

# What happens in learning, then?

- the learning algorithm observes **given examples**
- it tries to find common patterns that explain the data: it tries to **generalise** so that predictions can be made for **new examples**
- exactly how this is done depends on what **algorithm we are using**

keywords here:

- given/new examples
    - the settings of a learning experiment are important
- generalising
- algorithm we are using

# What happens in learning, then?

- the learning algorithm observes **given examples**
- it tries to find common patterns that explain the data: it tries to **generalise** so that predictions can be made for **new examples**
- exactly how this is done depends on what **algorithm we are using**

keywords here:

- given/new examples
- generalising
  - what does it mean to generalise well?
- algorithm we are using

# What happens in learning, then?

- the learning algorithm observes **given examples**
- it tries to find common patterns that explain the data: it tries to **generalise** so that predictions can be made for **new examples**
- exactly how this is done depends on what **algorithm we are using**

keywords here:

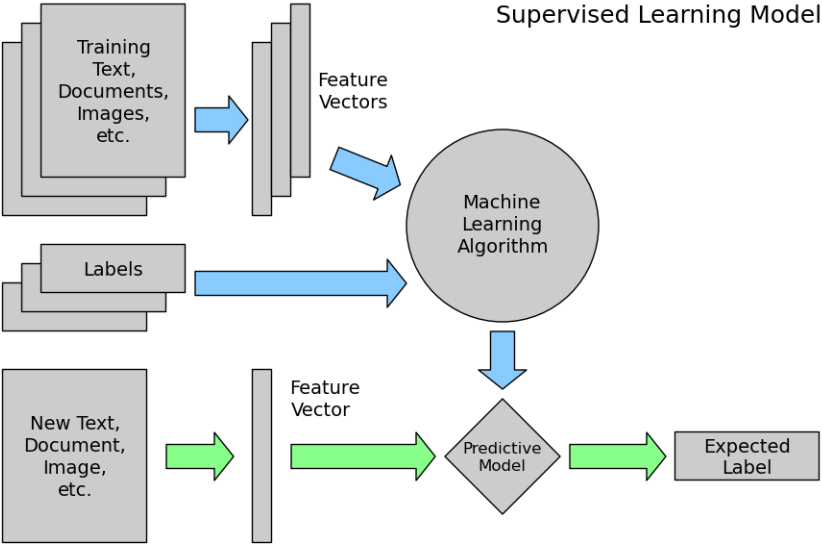
- given/new examples
- generalising
- algorithm we are using
  - we will see and use various, exploiting existing implementations

setup

# Procedure for a classification task

1. problem formulation
2. collection (and annotation) of examples
3. representation of instances
4. choice of learning algorithm
5. training
6. testing
7. evaluation

# Procedure for a classification task


Supervised Learning Model
Training Text, Documents, Images, etc.
Feature Vectors
Machine Learning Algorithm
Labels
New Text, Document, Image, etc.
Feature Vector
Predictive Model
Expected Label

# This course's setup

- 3 datasets
  1. sentiment analysis (running example)
  2. language identification
  3. animacy classification
- libraries
  1. scikit-learn (`http://scikit-learn.org/stable/`): collection of tools for machine learning
  2. NLTK (Natural Language ToolKit)
  3. (→ install anaconda (`https://www.continuum.io/downloads`))
- this course's utilities

# Procedure

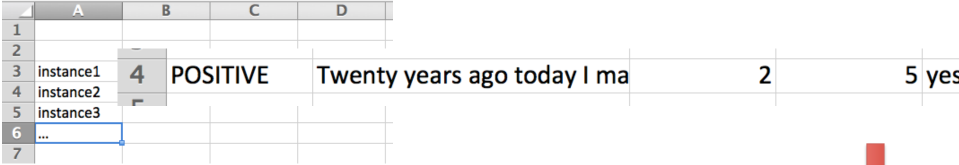
1. read in data (personalised)
2. extract features (partially supported by scikit-learn)
3. make model with chosen algorithm (supported by scikit-learn: one line!)
4. test model on new data (supported by scikit-learn: one line!)

# Procedure

1. read in data (personalised)
   - your preferred method, including by hand if you wish, for the purpose of learning
2. extract features (partially supported by scikit-learn)
   - you choose the features and store the values in a `.csv` file, one instance per line (we provide a readme for the exact format)
   - we provide a script that will take those features and will feed them into scikit-learn, in the required format (magic)
3. make model with chosen algorithm (supported by scikit-learn: one line!)
   - we provide a script that will fit the model using scikit-learn, and through a small modification you can choose the learning algorithm
4. test model on new data (supported by scikit-learn: one line!)
   - the same script will also classify new instances, using the scikit-learn implementation

words
sentences
documents
...

.csv



|   | A | B | C | D |
|---|---|---|---|---|
| 1 |   |   |   |   |
| 2 |   |   |   |   |
| 3 | instance1 |   |   |   |
| 4 | instance2 |   |   |   |
| 5 | instance3 |   |   |   |
| 6 | ... |   |   |   |
| 7 |   |   |   |   |

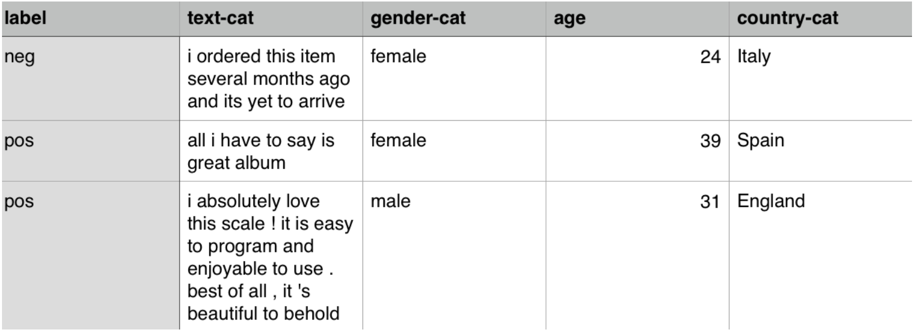
|   |   |   |   |   |   |
|---|---|---|---|---|---|
| 4 | POSITIVE | Twenty years ago today I ma | 2 | 5 | yes |

**feature_extractor.py**

**learn_from_data.py**

# What does a CSV file look like?

- Comma Separated Values
- Column format (like in Excel)



| label | text-cat | gender-cat | age | country-cat |
|---|---|---|---|---|
| neg | i ordered this item several months ago and its yet to arrive | female | 24 | Italy |
| pos | all i have to say is great album | female | 39 | Spain |
| pos | i absolutely love this scale ! it is easy to program and enjoyable to use . best of all , it 's beautiful to behold | male | 31 | England |

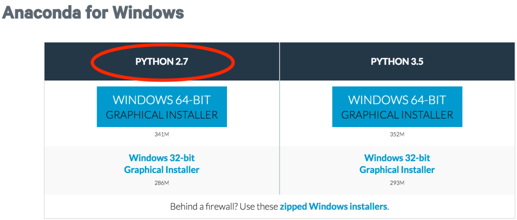
# Preparing your computer for machine learning

Installation (option a is preferred, option b is possible if a fails)

- Installing libraries
  - a Anaconda (`https://www.continuum.io/downloads`)
  - b Docker + Anaconda (`https://docs.docker.com/`, `https://goo.gl/HVDzqc`)
- Getting this course's utilities: Download a zip archive from `https://goo.gl/fD5IfG`.

# Installing Anaconda

- Anaconda (`https://www.continuum.io/downloads`)
- Get the version for your OS with Python 2.7 and follow installation instructions once download is complete
- When installed, open a terminal / command line
  - On Windows: Press Win + X, and click/tap on Command Prompt
  - On Mac OS X: Press cmd + space, type Terminal and press Enter
- In your terminal, type 'python' and press Enter. The version information displayed should contain the text 'Continuum Analytics'.



**Anaconda for Windows**

| PYTHON 2.7 | PYTHON 3.5 |
|---|---|
| WINDOWS 64-BIT GRAPHICAL INSTALLER<br>341M | WINDOWS 64-BIT GRAPHICAL INSTALLER<br>352M |
| Windows 32-bit Graphical Installer<br>286M | Windows 32-bit Graphical Installer<br>293M |

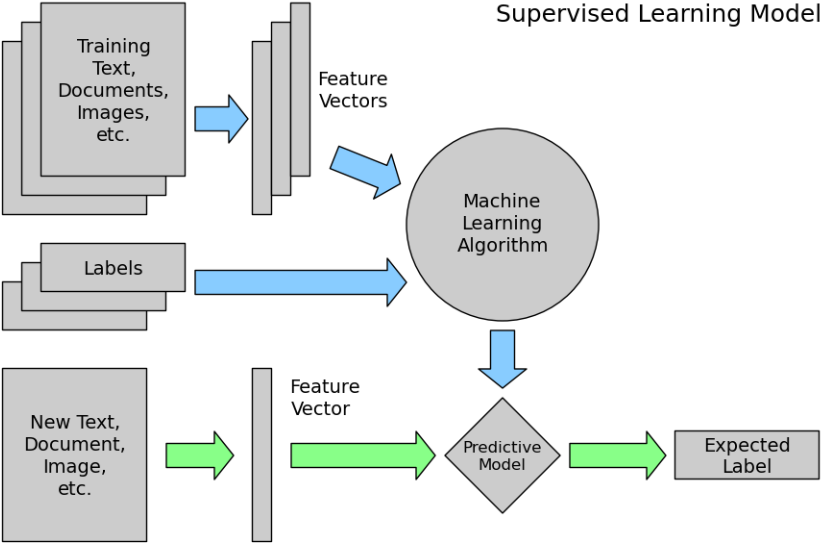
Behind a firewall? Use these zipped Windows installers.

# Getting this course's utilities:

- Download the archive from `https://goo.gl/fD5IfG`
- Unzip the archive
- In the terminal, navigate to the directory where you unzipped the archive
    - On Windows type 'cd' or 'chdir' followed by the directory `chdir Downloads\learning_from_data`)
    - On Mac OS X: type 'cd' followed by the directory `cd Downloads/learning_from_data`
- Run the test script by typing: `python test.py`
- Your output should say: 'Test successful!'

Settings

# Supervised Learning Model



Training Text, Documents, Images, etc.

Feature Vectors

Machine Learning Algorithm

Labels

New Text, Document, Image, etc.

Feature Vector

Predictive Model

Expected Label

source: `http://www.astroml.org/`

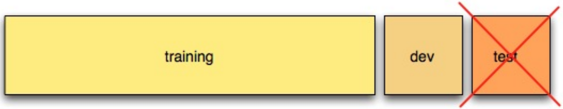
# Splitting the data

- training set: instances for training the system
- development set: instances for tuning the system and estimate error
- test or evaluation set: previously unseen instances on which model can be tested to asses its performance

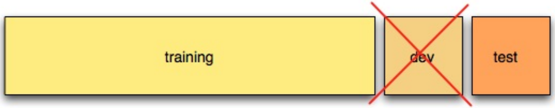
# Splitting the data

- training set: instances for training the system
- development set: instances for tuning the system and estimate error
- test or evaluation set: previously unseen instances on which model can be tested to asses its performance

building and tuning the model (repeatedly)


training
dev
test

evaluating the model (just once!)


training
dev
test

# Cross-validation

what if we don't have a lot of labelled data?

a separate test-set (e.g. 20%) might be not representative and could contain particularly easy/difficult instances

# Cross-validation

what if we don't have a lot of labelled data?

a separate test-set (e.g. 20%) might be not representative and could contain particularly easy/difficult instances

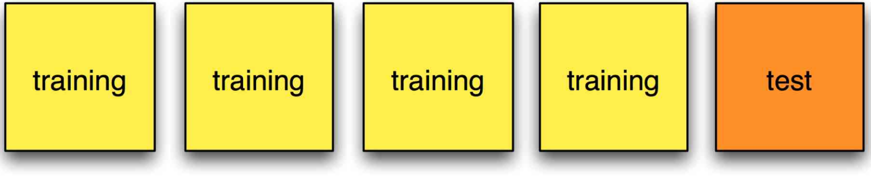
possible solution: cross-validation

- the whole dataset is split $k$ times (e.g. $k = 5$)
- training/testing is repeated $k$ times
- the whole dataset gets tested

# Cross-validation

possible solution: cross-validation

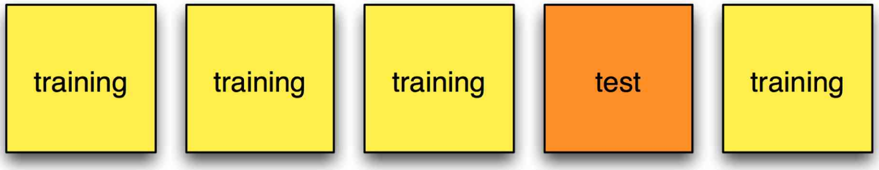
- the whole dataset is split $k$ times (e.g. $k = 5$)
- training/testing is repeated $k$ times
- the whole dataset gets tested



# Cross-validation

possible solution: cross-validation

- the whole dataset is split $k$ times (e.g. $k = 5$)
- training/testing is repeated $k$ times
- the whole dataset gets tested

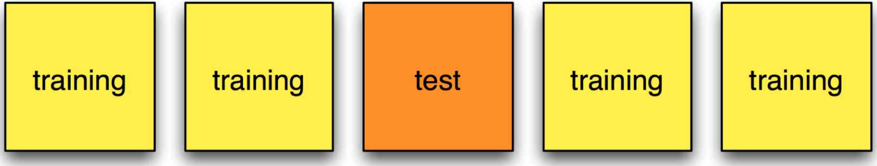


training training training test training

# Cross-validation

possible solution: cross-validation

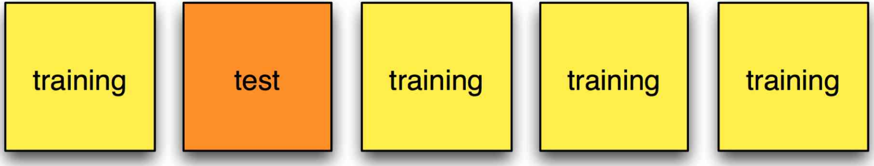
- the whole dataset is split $k$ times (e.g. $k = 5$)
- training/testing is repeated $k$ times
- the whole dataset gets tested



# Cross-validation

possible solution: cross-validation

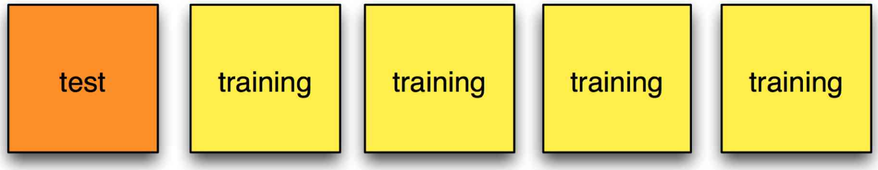
- the whole dataset is split $k$ times (e.g. $k = 5$)
- training/testing is repeated $k$ times
- the whole dataset gets tested



# Cross-validation

possible solution: cross-validation

- the whole dataset is split $k$ times (e.g. $k = 5$)
- training/testing is repeated $k$ times
- the whole dataset gets tested



test training training training training

# Evaluation

# Evaluation of results

→ is the system really able to generalise?

# Evaluation of results

→ is the system really able to generalise?

- the test set is equipped with class labels, manually assigned (gold standard)
- for each instance in the test set, we compare the class predicted by the classifier with the class specified in the gold standard
- how do we *measure* performance?
- when is a model good enough?

# Evaluation measures

- accuracy: percentage of correct decisions overall

# Evaluation measures

- accuracy: percentage of correct decisions overall

# Evaluation measures

Consider class “X”

- **true positive** (**TP**): X classified as X
- **true negative** (**TN**): ¬X classified as ¬X
- **false positive** (**FP**): ¬X classified as X
- **false negative** (**FN**): X classified as ¬X

# Evaluation measures

Consider class “X”

- **true positive (TP)**: X classified as X
- **true negative (TN)**: ¬X classified as ¬X
- **false positive (FP)**: ¬X classified as X
- **false negative (FN)**: X classified as ¬X

- precision: correct decisions over instances assigned to class “X” $TP/(TP + FP)$
- recall: correct assignments to class “X” over all instances of class “X” in test set $TP/(TP + FN)$
- f-score: combined measure of precision and recall $F = \frac{2PR}{P+R}$

# Evaluation measures

confusion matrix

| response → |  | X | ¬X |
|---|---|---|---|
| gold<br>↓ | X | TP | FN |
|  | ¬X | FP | TN |

# Evaluation measures

what is good enough?

- upperbound: inter-annotator agreement
- baseline: performance of basic, simple model
  for example: **assignment of most frequent class in data set**
  - $\text{sense}_1$ $9/10$ and $\text{sense}_2$ $1/10$
  - $\text{sense}_1$ $6/10$ and $\text{sense}_2$ $4/10$

# Live poll

http://etc.ch/tvz6

# Live poll – results

http://directpoll.com/r?XDbzPBd3ixYqg81LygsVoSIvClR6cnLre6kxM2M3