## Intro summary

Fit numbers in a program to data (i.e. train machines on data)

- More robust than hand-fitted rules
- Can’t approach humans at some tasks (e.g., vision)
- Machines make better predictions in many other cases

## Applications of machine learning

Within informatics:
- **Vision**: as we’ve seen. [another e.g.]
- **Graphics**: increasingly data driven
- **AI & Natural Language Processing (NLP)**: text search/summarisation, speech recognition/synthesis, e.g., IBM Watson
- **Robotics**: vision, planning, control, . . .
- **Compilers**: learning how to optimise and beyond: data analysis across the sciences

Every day:
- Adverts / recommendations all over the web . . . Big Data
- Speech recognition and synthesis (e.g., Siri, Echo), Machine Translation, . . . with self-driving cars

## Course structure

- **30 lectures** (including review) - Tuesdays, Thursdays, Fridays
- **Tutorials** (starting in week 2)
- **Drop-in labs for Learning** (Fridays 14:10-15:00)
- **2 assessed assignments** (with drop-in labs)

  - **CW1 (ADS)**: 08/02 - 26/02 (Part 1), 09/03 (Part 2)
  - **CW2 (Learning)**: 13/03 - 04/04

**Equal split into two threads:**
- Algorithms and Data Structures – *KK* (Kyriakos Kalorkoti) Design and analysis of correct, efficient and elegant data structures and algorithms.
- Learning – Hiroshi Building models that describe a data set and can make predictions about new data

## Face detection

*How would you detect a face?*

![Face detection](http://www.inf.ed.ac.uk/teaching/courses/inf2b/

*How does album software tag your friends?*

![Album software](http://vimeo.com/12774628)

**Viola–Jones Face detection (2001)**

- Face detector consists of linear combination of ‘weak’ classifiers that utilise five types of primitive features.
- The detector is trained on a training data set of a large number of positive and negative samples.
- Scan the input image with a sub-window (24 x 24 pixels) to detect a face.

Taken from: [http://ahprojects.com/projects/cv-dazzle](http://ahprojects.com/projects/cv-dazzle)

A nice demo: [http://vimeo.com/12774628](http://vimeo.com/12774628)

## Drop-in labs for Learning

- Fridays 14:10-15:00 in AT-5.05 (West Lab)
- Worksheets available on the course webpage
- **Purposes of lab sessions**
  - Assistance in understanding basic algorithms and techniques of machine learning and data analysis
  - Assistance in programming with Matlab
  - Assistance in working on the 2nd assignment (CW2)
- **Practice on machine learning using Matlab**
- **Work on toy problems for the topics taught in the course**
- **Demonstrator**: Andreas Kapourain
**Classification of oranges and lemons**

Represent each sample as a point \((w, h)\) in a 2D space

![Oranges and Lemons](image_url)

**A two-dimensional space**

State that each sample is a point \((w, h)\) in a 2D space

**Photo image – pixels**

Use a photo image to represent pixels as points on a 2D plane.

---

### Private study

- ~2 hours private study per lecture, in addition to tutorials & assignments
- No required textbook for Inf2b: There are notes and slides. See those for recommended books.
- Importance of maths skills (especially algebra):
  - Why should you remember and get familiar with maths formulas for machine learning?
    - Good understanding of the ideas
    - Guessing reasonable output of the model
    - Identifying spotting the problems (bugs) with the system implemented
  - Importance of programming practice (with Matlab or Python) (attend the drop-in labs!)

### Private study (cont.)

- Warning: Inf2b is NOT an easy course
- Inf2b requires a solid maths background:
  - Linear Algebra
  - Calculus
  - Probability
  - Independent learning (self-directed learning) is essential.
  - See the following page regarding differences between secondary-school and university in terms of learning style and what is expected from you as a student.

### Maths skills

Useful webpage to check your maths:

http://www.mathsisfun.com/algebra

- Laws of exponents (Exponent rules)
  - \(x^a x^b = x^{a+b}\), \((x^a)^b = x^{ab}\)
- Log and exponential
  - \(\log(x^a y^b) = \log x + m \log y\), \(e^{\ln x} = x\)
- Quadratic equations and their solutions
  - \(ax^2 + bx + c = 0\), \(x = -\frac{b \pm \sqrt{b^2 - 4ac}}{2a}\)
- Vectors
  - \(v = (v_1, v_2, \ldots, v_n)^T\)
  - Notation: column/row vectors, transpose
  - Addition and subtraction e.g. \(u + v\)
  - Dot product (inner product) \(u \cdot v = u^T v\)
  - Equation of a straight line, linear equations

### Maths skills (cont.)

- Matrices \(A = (a_{ij})\), \(A_i = a_{ij}\)
  - Addition, subtraction \(A + B, A - B\)
  - Multiplication \((AB)_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}\)
  - Transpose \((ABC)^T = C^T B^T A^T\)
  - Determinant \(|A|\)
  - Inverse \(A^{-1} A = A A^{-1} = I\)
  - Eigen values and eigen vectors
  - Vector spaces, subspaces, linear independence, basis and dimension, rank and nullity
  - Linear transformations \(y = Ax\)

**Important notes**

- Two hours study this week?
  - Start to familiarise yourself with Matlab (or Octave)
    - Introductory worksheet on the course website
  - Many others at the end of a web search
  - Learn Matlab try the lab sheets for the 1st lab this week.
  - Love Python? Learn NumPy+SciPy+Matplotlib (instead, or as well)

**Vital skills**

- add, average, multiply vectors and matrices
- plot data stored in vectors
- save/read data to/from files

**Classroom**

- Have a look at the lecture note and slides in advance to the lecture.
- Have questions prepared to ask.
- Laptops, tablets, phones are not allowed to use during lectures unless permitted.

**NB:** Lectures will be recorded, and videos will be published in a few days after the lecture.
Pixel image to a feature vector

Turn each cell (pixel) into a number (somehow, see notes)
Unravel into a column vector, a feature vector
$$x = (x_1, x_2, \ldots, x_{64})^T, \quad x_i \in [0, 127] \text{ or } x_i \in [0, 1]$$

http://alex.seewald.at/digits/

Image data as a point in a vector space

Distance between 2D vectors: $$u = (u_1, u_2)^T$$ and $$v = (v_1, v_2)^T$$
$$d(u, v) = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2}$$

Distance between $$d$$-dimensional vectors: $$u = (u_1, \ldots, u_d)^T$$ and $$v = (v_1, \ldots, v_d)^T$$
$$d(u, v) = \sqrt{\sum_{k=1}^{d} (u_k - v_k)^2}$$

Measures similarities between feature vectors
i.e., similarities between digits, movies, sounds, galaxies, …

Euclidean distance

Question

Have high-resolution scans of digits.
How many pixels should be sample?
What are pros and cons of:
2×2, 4×4, 16×16, or 100×100?

Example of image resolutions

Exercises in the lecture note 1

Try the exercises in the lecture note 1.
No solutions will be published.
In case you’re not sure if your answers are correct:
Discuss them with your classmates
Use the Inf2b-Learning discussion board on Piazza

Summary

Self-study both ADS and Learning everyday.
Drop-in for Learning starts on Friday, 19th Jan at 14:10.
Try the worksheets before the lab.
Tutorial starts in Week 2.
Discussion forum for “Learning” in Piazza
Office hours: Wednesdays at 14:00-15:00 (TBC) in IF-3.04