Welcome to Inf2b!

Inf2b: Algorithms, Data Structures (ADS), and Learning

Today’s Schedule:

1. Course structure
2. What is (machine) learning? (and why should you care?)
3. Administrative stuff
   - How to do well
4. Setting up a learning problem

(time allowing)

Course structure

http://www.inf.ed.ac.uk/teaching/courses/inf2b/

Constituents:

- 30 lectures (including review)
- Tutorials starting in week 2
- 2 assessed assignments (with drop-in labs)

CW1 (ADS): 12/02 – 06/03
CW2 (Learning): 10/03 – 27/03

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 Shimodaira
  - Building models that describe a data set and can make predictions about new data

Applications of machine learning

Within informatics:
- Vision: as we’ve seen
- 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
- Discounts in Tescos: http://www.corporate.tesco.co.uk/dms/file/scene-fat-taste.html
- Speech recognition and synthesis (e.g. Siri), Machine Translation, … with self-driving cars

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

Face detection

How would you detect a face?

How does album software tag your friends?

Hiding from the machines (cameras)

The Viola-Jones face detector is fast, but has some drawbacks.

Taken from: http://ahprojects.com/projects/cv-dazzle

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
A nice demo: http://vimeo.com/12774628
Course structure

What is machine learning

Administrative stuff

Setting up a learning problem

Private study

- ~2 hours private study per lecture in addition to tutorials & assignments
- No required textbook for Inf2b. There are notes. See those for recommended books.
- Required maths skills (especially linear 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]

Maths skills

Useful webpage to check your maths:
http://www.mathsisfun.com/algebra

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

Maths skills (cont.)

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

NB: See Section 4 of Learning Note No. 1 for the notation we use.

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

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

Class reps

WANTED: Inf2b class reps (for ADS & Learning)

Email: h.shimodaira@ed.ac.uk

your name, degree, email address.

Classification of oranges and lemons

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

Oranges: 🍊

Lemons: 🍋
Handwritten digits recognition

A 64-dimensional space

Euclidean distance

Distance between 2D vectors: \( u = (u_1, u_2) \) and \( v = (v_1, v_2) \)

\[ r_d(u, v) = \sqrt{(u_1 - v_1)^2 + (u_2 - v_2)^2} \]

Distance between \( d \)-dimensional vectors: \( u \) and \( v \)

\[ r_d(u, v) = \sqrt{\sum_{k=1}^{d} (u_k - v_k)^2} \]

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

Question

Have high-resolution scans of digits.

How many pixels should be sample?

What are pros and cons of:

\( 2 \times 2, 4 \times 4, 16 \times 16, \) or \( 100 \times 100? \)