• Class web page (check regularly!):
  www.cs.ubc.ca/~murphyk/Teaching/cs340-fall07
• TAs:
  Hoyt Koepke
• hoytak@cs.ubc.ca
  Erik Zawadzki
  epz@cs.ubc.ca
• Tutorials: T1A Thur 3.30-4.30 (Hoyt),
  Frank Forward Building (behind barn), room 317
  T1B Wed 4-5 (Erik), MacLeod 214
• Office hours: By appointment
• Midterm: Wed 10 October
• Grading
  – Midterm: 25%
  – Final: 50%
  – Weekly Assignments: 25%

• Collaboration policy:
  – You can collaborate on homeworks if you write the name of your collaborators on what you hand in; however, you must understand everything you write, and be able to do it on your own (eg. in the exam!)

• Sickness policy:
  – If you cannot do an assignment or an exam, you must come see me in person; a doctor's note (or equivalent) will be required.
Pre-requisites

• You should know (or be prepared to learn)
  – Basic multivariate calculus e.g.,
    \[
    \frac{\partial}{\partial x_j} \bar{x}^T \bar{x} = 2x_j
    \]
  – Basic linear algebra e.g.,
    \[
    A\bar{u}_i = \lambda_i \bar{u}_i
    \]
  – Basic probability/ statistics e.g.
    \[
    \]
  – Basic data structures and algorithms (e.g., trees, lists, sorting, dynamic programming, etc)
None required – I will give handouts and slides. However, the following are recommended.

- Bishop
- HTF
- DHS (Duda, Hart, Stork)
More recommended books

Artificial Intelligence: A Modern Approach
Stuart Russell • Peter Norvig

All of Statistics: A Concise Course in Statistical Inference
Larry Wasserman

Bayesian Methods for Nonlinear Classification and Regression
D. G. T. Denison • C. C. Holmes
B. K. Mallick • A. F. M. Smith
Matlab

- Matlab is a mathematical scripting language widely used for machine learning (and engineering and numerical computation in general).
- Everyone should have access to Matlab via their CS account. If not, you can ask the TAs for a CS guest account.
- You can buy a student version for $170 from the UBC bookstore, but you will also need the stats toolbox (and sometimes also the optimization toolbox).
- Hoyt will give a brief introduction to Matlab in class on Mon 10th.
- Prof Mitchell will give a brief Matlab tutorial on Wed Sept 12, 5pm - 7pm, DMP 110
- The first homework (due on Mon 17th) consists of some simple Matlab programming. Check you have Matlab today!
By the end of this class, you should be able to

- Understand basic principles of machine learning and its connection to other fields
- Derive, in a precise and concise fashion, the relevant mathematical equations needed for familiar and novel models/ algorithms
- Implement, in reasonably efficient Matlab, various familiar and novel ML model/ algorithms
- Choose an appropriate method and apply it to various kinds of data/ problem domains
Ask questions early and often!
End of administrivia
What is machine learning?
What is machine learning?

• "Learning denotes changes in the system that are adaptive in the sense that they enable the system to do the task or tasks drawn from the same population more efficiently and more effectively the next time." -- Herbert Simon

• Closely related to
  – Statistics (fitting models to data and testing them)
  – Data mining/ exploratory data analysis (discovering patterns in data)
  – Adaptive control theory (learning models online and using them to achieve goals)
  – AI (building intelligent machines by hand)
Types of machine learning

• Supervised Learning
  – Predict output from input

• Unsupervised Learning
  – Find patterns in data

• Reinforcement Learning
  – Learn how to behave in novel environments (e.g., robot navigation)
  – not covered in this class – see e.g., CS422
Why “Learn”? 

• Machine learning is programming computers to optimize a performance criterion using example data or past experience.
• There is no need to “learn” to calculate payroll.
• Learning is used when:
  – Humans not in loop (navigating on Mars)
  – Humans are unable to explain their expertise (speech recognition)
  – Solution changes in time (routing on a computer network)
  – Solution needs to be adapted to particular cases (user biometrics)
## Binary classification - credit card scoring

<table>
<thead>
<tr>
<th>Income</th>
<th>Savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>50</td>
</tr>
<tr>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>50</td>
<td>95</td>
</tr>
<tr>
<td>500</td>
<td>93</td>
</tr>
</tbody>
</table>

- **Risk**
  - hi
  - lo

**Training data**

<table>
<thead>
<tr>
<th>Income</th>
<th>Savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>98</td>
<td>49</td>
</tr>
<tr>
<td>100</td>
<td>102</td>
</tr>
<tr>
<td>400</td>
<td>20</td>
</tr>
</tbody>
</table>

**Test data**
Supervised learning as function fitting

- Given parametric function $f$ in hypothesis class $H$

  $$f \in \mathcal{H} : \mathcal{X} \times \Theta \rightarrow \mathcal{Y}$$

  $$\hat{y} = f(x, \theta)$$

- And labeled training data

  $$D = \{x_1, y_1, \ldots, x_N, y_N\}$$

- Estimate parameters $\theta$ given $D$ so that predictions on test set are as accurate as possible
Example function

\[ \mathcal{X} = R^2, \mathcal{Y} = \{ hi, lo \} \]

\[ f(x, \theta) = \text{IF } \text{income} > \theta_1 \text{ AND savings} > \theta_2 \]
\[ \text{THEN low-risk ELSE high-risk} \]
Decision trees

Decision tree:

- $x_1 > \theta_1$
  - $y$
    - $x_2 > \theta_2$
      - hi
    - $x_2 \leq \theta_2$
      - lo
        - hi

- $y$
  - $x_1 \leq \theta_1$

$H = \{\text{Axis-parallel hyper-planes}\}$
Multi-class Decision trees

- $x_1 > \theta_1$
  - $y$
    - $x_2 > \theta_2$
      - hi
        - $\theta_2$
          - $\theta_1$
    - lo
      - hi
        - $\theta_0$
          - $\theta_1$

- $x_1 > \theta_1$
  - $y$
    - $x_2 > \theta_2$
      - hi
        - $\theta_2$
          - $\theta_0$
          - $\theta_1$
    - $x_2 > \theta_3$
      - lo
        - med
What's the right hypothesis class $H$?
Linearly separable means if $f$ is a linear function of $x$, we can perfectly fit the training data

$$f(x, \theta) = \text{sgn}(\theta^T x) = \text{sgn}(\theta_0 + \theta_1 x_1 + \theta_2 x_2) \in \{-1, +1\}$$
Not linearly separable
Quadratically separable

\[ f(x, \theta) = \text{sgn}(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2) \]
Noisy/ mislabeled data
• An overly flexible function memorizes irrelevant details of training set
Overfitted functions do not predict test data

- Predict label of green points
Overfitted functions do not predict test data

• Test points are mis-predicted
Tradeoff simplicity for model fit
Tradeoff simplicity for model fit
Occam’s razor

• If two models fit the data equally well, pick the simpler one
• In general, since our goal is to predict the test data, we may choose to incur errors on the training set if it results in a simpler function
Function fitting

1. Choose right hypothesis class $H$ given $D$

   - **linear**
   - **quadratic**

   - Depth-2 decision tree

2. Fit parameters of function $\theta$ given $H$ and $D$

   $$f(x, \theta) = sgn(\theta^T x) = sgn(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$
Hypothesis class depends on amount of data

- More complex function is ok if we have more data, because we have more evidence for it
Hypothesis class depends on type of data

- Decision regions may be discontinuous
Hypothesis class depends on type of data

Input features may be discrete ($X$ not Euclidean space)
Classifying gene microarray data

What’s the right hypothesis class now?
Handwritten digit recognition

- $x^t \in \mathbb{R}^{16 \times 16}$, $y^t \in \{0, \ldots, 9\}$

What’s the right hypothesis class now?
Face Recognition

Training examples of a person

Test images

Possibly no negative examples

What's the right hypothesis class now?
Face detection in images
Face detection in images
Face detection in images
Face detection in images
Face detection in images
Face detection in images
Car detection
Classifying image patches

- Texture classification using SVMs
  - foliage, building, sky, water

Image Retrieval

Source: Mike Cora, UBC, 2005
Place recognition for a wearable computer

$\text{t}=930, \text{truth } = 400$-fl6-visionArea1
Place recognition for a mobile robot
Natural language processing (NLP)

• We do not yet know good ways to represent the "meaning" of a sentence (this is called the knowledge representation problem in AI)

• Current approaches to statistical NLP involve shallow parsing, where the meaning of a sentence can be represented by fields in a database eg
  - "Microsoft acquired AOL for $1M yesterday"
  - "Yahoo failed to avoid a hostile takeover from Google"

<table>
<thead>
<tr>
<th>Buyer</th>
<th>Buyee</th>
<th>When</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>MS</td>
<td>AOL</td>
<td>Yesterday</td>
<td>$1M</td>
</tr>
<tr>
<td>Google</td>
<td>Yahoo</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
Learning how to talk: nettalk

Mary had a little lamb, its fleece...

Source: Sejnowski & Rosenberg, 1987