CS 540: Machine Learning
Lecture 1: Introduction

AD

January 2008
Acknowledgments

Thanks to

- Nando de Freitas
- Kevin Murphy
Administrivia & Announcement

- Office hours: Tuesday & Thursday 11.30-12.30.
- Introductory tutorial Matlab session by Ian Mitchell TODAY, 5pm - 7pm, DMP 110
- Matlab resources:
  http://www.cs.ubc.ca/~mitchell/matlabResources.html
Pre-requisites

- Maths: multivariate calculus, linear algebra, probability.
- CS: programming skills, knowledge of data structures and algorithms helpful but not crucial.
- Prior exposure to statistics/machine learning is highly desirable but you will be ok as long as your math is good.
- I will discuss Bayesian statistics at length.
(Tentative) grading

- The midterm will be sometimes around the end of February and is worth 30%.
- The final project will be in April ?? and is worth 40%.
- There will be three (long) assignments worth 30%.
- Additional optional exercises will be given.
Textbook

- Check the webpage http://research.microsoft.com/~cmbishop/PRML/
- Corrections of exercises available on this webpage.
- Each week, there will be required reading, optional reading and background reading.
Related classes

- CPSC 532L - Multiagent Systems (Leyton-Brown).
- CPSC 532M - Dynamic Programming (Ian Mitchell)
- Statistics courses.
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” – Herb Simon.

- Machine Learning is an interdisciplinary field at the intersection of Statistics, CS and EE etc.

- At the beginning, Machine Learning was fairly heuristic but it has now evolved and become -in my opinion- Applied Statistics with a CS flavour.

- Aim of this course: introducing basic models, concepts and algorithms.
Main Learning Tasks

- **Supervised learning**: given a training set of $N$ input-output pairs \( \{x_n, t_n\} \in \mathcal{X} \times \mathcal{T} \), construct a function \( f : \mathcal{X} \rightarrow \mathcal{T} \) to predict the output \( \hat{t} = f(x) \) associated to a new input \( x \).
  - Each input \( x_n \) is a \( p \)-dimensional feature vector (covariates, explanatory variables).
  - Each output \( t_n \) is a target variables (response).

- Regression corresponds to \( \mathcal{T} = \mathbb{R}^d \).
- Classification corresponds to \( \mathcal{T} = \{1, \ldots, K\} \).
- Aim is to produce the correct output given a new input.
Polynomial regression

Figure: Polynomial regression where $x \in \mathbb{R}$ and $t \in \mathbb{R}$
Figure: Linear least square fitting for $x \in \mathbb{R}^2$ and $t \in \mathbb{R}$. We seek the linear function of $x$ that minimizes the sum of square residuals for $y$. 
Piecewise linear regression

Figure: Piecewise linear regression function
Figure: Linear classifier where $x_n \in \mathbb{R}^2$ and $t_n \in \{0, 1\}$
Figure: Examples of handwritten digits from US postal employes
**Figure:** Can you pick out the tufas?
Practical examples of classification

- Email spam filtering (feature vector = “bag of words”)
- Webpage classification
- Detecting credit card fraud
- Credit scoring
- Face detection in images
Unsupervised learning: we are given training data \( \{x_n\} \).

Aim is to produce a model or build useful representations for modeling the distribution of the data, clustering, data association, dimensionality reduction, structure learning.
Hard and Soft Clustering

**Figure:** Data (left), hard clustering (middle), soft clustering (right)
Finite Mixture Models

- Finite Mixture of Gaussians

\[ f(x|\theta) = \sum_{i=1}^{k} p_i \mathcal{N}(x; \mu_i, \sigma_i^2) \]

where \( \theta = \{\mu_i, \sigma_i^2, p_i\}_{i=1,...,k} \) is estimated from some data \((x_1, ..., x_n)\).

- How to estimate the parameters?
- How to estimate the number of components \(k\)?
Modelling dependent data
In this case, we have

\[ x_n \mid x_{n-1} \sim f(\cdot \mid x_{n-1}), \]
\[ y_n \mid x_n \sim g(\cdot \mid x_n). \]

For example, stochastic volatility model

\[
\begin{align*}
    x_n &= \alpha x_{n-1} + \sigma v_n \text{ where } v_n \sim \mathcal{N}(0,1) \\
    y_n &= \beta \exp(x_n/2) w_n \text{ where } w_n \sim \mathcal{N}(0,1)
\end{align*}
\]

where the process \((y_n)\) is observed but \((x_n, \alpha, \sigma, \beta)\) are unknown.
Tracking Application
Hierarchical Clustering

Figure: Dendogram from agglomerative hierarchical clustering with average linkage to the human microarray data
Dimensionality reduction

Figure: Simulated data in three classes, near the surface of a half-sphere
Figure: The best rank-two linear approximation to the half-sphere data. The right panel shows the projected points with coordinates given by the first to principal components of the data.
Manifold learning

Figure: Data lying on a low-dimensional manifold
Structure learning

Figure: Graphical model
Ockham’s razor

- How can we predict the test set if it may be arbitrarily different from the training set?
- We need to make an assumption that the future will be like the present in statistical terms.
- Any hypothesis that agrees with all is as good as any other, but we generally prefer “simpler” ones.
Active learning is a principled way of integrating decision theory with traditional statistical methods for learning models from data.

In active learning, the machine can query the environment. That is, it can ask questions.

Decision theory leads to optimal strategies for choosing when and what questions to ask in order to gather the best possible data.

“Good” data is often better than a lot of data.
Active learning and surveillance

- In network with thousands of cameras, which camera views should be presented to the human operator?
Active learning and sensor networks

- How do we optimally choose among a subset of sensors in order to obtain the best understanding of the world while minimizing resource expenditure (power, bandwidth, distractions)?
Active learning example

Input Data

First Query

Second Query

Result
Agents (humans, animals, machines) always have to make decisions under uncertainty.

Uncertainty arises because we do not know the “true” state of the world; e.g. because of limited field of view, because of “noise”, etc.

The optimal way to behave when faced with uncertainty is as follows:

- Estimate the state of the world, given the evidence you have seen up until now (this is called your belief state).
- Given your preferences over possible future states (expressed as a utility function), and your beliefs about the effects of your actions, pick the action that you think will be the best.
You should choose the action with maximum expected utility

\[
a^*_t = \arg \max_{a_{t+1} \in \mathcal{A}} \sum_x P_\theta (X_{t+1} = x \mid a_{1:t+1}, y_{1:t}) U_\phi (x)
\]

where

- \(X_t\) represents the “true” (unknown) state of the world at time \(t\)
- \(y_{1:t} = y_1, \ldots, y_t\) represents the data I have seen up until now
- \(a_{1:t}\) are the actions that I have taken up until now
- \(U(x)\) is a utility function on state \(x\)
- \(\theta\) are parameters of your world model
- \(\phi\) are parameters of your utility function
Decision making under uncertainty

- We will later justify why your beliefs should be represented using probability theory.
- It can also be shown that all your preferences can be expressed using a scalar utility function (shocking, but true).
- Machine learning is concerned with estimating models of the world ($\theta$) given data, and using them to compute belief states (i.e., probabilistic inference).
- Preference elicitation is concerned with estimating models ($\phi$) of utility functions (we will not consider this issue).
Decision making under uncertainty: examples

- Robot navigation: $X_t =$ robot location, $Y_t =$ video image, $A_t =$ direction to move, $U(X) =$ distance to goal.
- Medical diagnosis: $X_t =$ whether the patient has cancer, $Y_t =$ symptoms, $A_t =$ perform clinical test, $U(X) =$ health of patient.
- Financial prediction: $X_t =$ true state of economy, $Y_t =$ observed stock prices, $A_t =$ invest in stock $i$, $U(X) =$ wealth.
Tentative topics

- Review of probability and Bayesian statistics
- Linear and nonlinear regression and classification
- Nonparametric and model-based clustering
- Finite mixture and hidden Markov models, EM algorithms.
- Graphical models.
- Variational methods.
- MCMC and particle filters.
- PCA, Factor analysis
- Boosting.
A list of potential projects will be listed shortly.
You can propose your own idea.
I expect the final project to be written like a conference paper.