Course basics

- Ling 696: Advanced Statistical Methods in Computational Linguistics
- Thursday 7:00–9:40
- Instructor: Rob Malouf, `rmalouf@mail.sdsu.edu`
- Office hrs: BA 310A, Mondays 11:00–12:00, Thursdays 2:00-3:00
- `http://rohan.sdsu.edu/~malouf/ling696.html`

Course basics

- Registration
- Prerequisites (Ling 681)
- Lab
- Schedule change

Requirements

- Textbook:

  Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing.* MIT Press.

- Additional readings
- Reference room
- Assessment:
  - Homeworks (30%)
  - Final project (70%)

Schedule

- Week 1 **Introduction**
  Background · Mathematical background · Machine learning applications · Types of models
- Week 2–4 **Non-parametric methods**
  Decision trees · Memory-based learning · Rule induction
- Week 5–7 **Bayesian methods**
  Naive Bayes classifiers · Improved priors · Maximum Entropy classifiers · Conditional random fields
- Week 8–10 **Ensemble machines**
  Weighted voting · bagging · boosting · co-training

Schedule

- Week 11–13 **Kernel methods**
  Linear classifiers · Perceptron · Kernel functions · Support Vector Machines
- Week 14 **Odds and ends**
  Training data · Running experiments · Computational realities
- Week 15 **Projects**
  Final project due May 13

Probability

- Probability theory predicts long-term frequency of events

  If we put \$100 on “Black” in American roulette, what fraction of the spins will we win, on average?

- We can use this to get to expected values:

  If we put \$100 on “Black” in American roulette 12 times, how much will we win, on average?

- Probability theory provides a collection of rules for answering these kinds of questions.

Probability

- To apply probability theory to a problem, we need to construct a *model*
- The model should capture the essential properties of the problem
- while abstracting away from irrelevant details
- To model a coin toss: we could try to capture all of the physical, aerodynamic, metallurgical, numismatic properties of the coin/hand interface…
- Or, we could use a Bernoulli variable as a model

Probability

- We choose a model class (Bernoulli variable, second order Markov process, etc.) based on our understanding of the problem
- We then need to find the particular instantiation of the model class
- The model+parameters gives us probabilities, which we can use to estimate long-term frequencies
- Presumably, then, we have some reason for caring about long-term frequencies

Inference

- Most of the time, probabilities are used to make informed decisions in the face of partial information
- What is a fair payoff for an outside bet in American roulette?
- Can I conclude that my patients got better because of the new treatment, or might they just have been lucky?
- What is the right sequence of part-of-speech tags for this sentence?

Classification

- Part of speech tagging is a *classification* problem: assign one or more labels L from a finite set to instances I
- Classification problems come up frequently in NLP, and can be approached probabilistically by using a model to estimate $P(I, L)$
- Classifiers can also be built by hand, e.g., as a cascade of finite state transducers which map from I to L
- A wide range of NLP tasks can be cast as classification problems

Classification

- Part of speech tagging, chunking, named entity recognition, word sense disambiguation
- Spelling correction
- Text classification, information retrieval, automated metadata generation, message routing
- Text segmentation, text summarization
- Adjective ordering
- Anything else?

Evaluation

- We can judge the performance of a classifier by its*accuracy* (the fraction of instances which it correctly labels) or *error* $(1 - \text{accuracy})$
- In some applications, it can be more useful to distinguish between types of errors
- For each class, we count false positives (FP), true positives (TP), false negatives (FN), and true negatives (TN)
- Precision is the proportion of correct positive class assignments ($\frac{TP}{TP+FP}$), recall is the proportion of class members which are correctly labeled ($\frac{TP}{TP+FN}$), and fallout is the proportion of non-members which are incorrectly labeled ($\frac{FP}{FP+TN}$)

Evaluation

- Since precision, recall, and fallout all involve tradeoffs, we sometimes combine them into a composite score (F score, breakeven)
- *Utility* is the most general metric, with arbitrary weights for different kinds of errors
- For multi-class problems, an overall score can be computed either by microaveraging (by instance) or macroaveraging (by class)

Evaluation

- Training, validation, test data
- Cross validation
- The *learning curve* shows how performance increases (we hope!) with experience
- Sometimes, performance will decrease slightly after a point
- When this happens, it can be a symptom of *overtraining*

Classification

- The classification problems which we face in CL are often very complex and poorly understood, so that neither probability models nor rules are very helpful
- Ideally, we would like to present the computer with a set of properly labeled instances, and get back a classifier
- Supervised vs. unsupervised learning
- But, learning is never *really* unsupervised

Machine Learning

- The field of *machine learning* studies methods for writing programs which can improve their performance (given some metric) based on experience
- A bigram tagger is an example of a machine learning method
- Other, more general methods for learning make fewer assumptions about the underlying concept
- Related to data mining: tell me something I didn't know (unsupervised learning)

Machine Learning

- There are slightly more machine learning techniques than there are machine learning researchers
- Most “X-based Learning” algorithms fall into a few general classes
- *Parametric* methods use a probability distribution to find the most probable solution
- Early *non-parametric* machine learning methods used common sense strategies, plus ad hoc heuristics (decision trees, memory based learning)
- Statistical learning theory has developed to the point that it is driving development of new machine learning methods

Machine Learning

- Classifiers differ in the range of concepts they are capable of learning
- Parametric models, number of parameters
- Decision boundaries for non-parameteric methods
- Machine learning algorithms also differ in their computational properties

TANSTAAFL

- No Free Lunch Theorem (Wolpert and Macready 1994)
  If problems are uniformly distributed, then on average all optimization algorithms will perform the same
- There is no “best” machine learning method *if problems are uniformly distributed*
- Then why don’t we just randomly generate solutions?
- Problems aren’t uniformly distributed, of course!

TANSTAAFL

- Much machine learning lore relates to the kinds of problems that particular algorithms are particularly well suited for
- Much more rarely, we see the kinds of problems that particular algorithms are particularly ill suited for
- Choosing the best method in a particular situation is often a matter of trial and error (good for thesis topics!)
- Most of the time, different methods give pretty much the same results (even better for thesis topics!)

Bias/variance decomposition

- A general theme in machine learning is a tradeoff between prior information and properties of the training data
- We need prior assumptions to narrow the space of possible solutions, but if incorrect this can lead to errors
- We can formalize this (Geman 1992):

$$\text{error} = \text{bias error}^2 + \text{variance}$$

or, more formally:

$$\frac{1}{NM}\sum_i^N\sum_j^M(t_i - y_{ij})^2 = \frac{1}{N}\sum_i^N(t_i - \overline{y_i})^2 + \frac{1}{NM}\sum_i^N\sum_j^M(\overline{y_i} - y_{ij})^2$$

Bias/variance decomposition

- Different methods provide different types of bias, either implicitly or explicitly
- Appropriate bias reduces variance
- Inappropriate bias reduces variance, but increases bias error
- When we don't have enough training data, variance is more of a problem than bias error (Curse of Dimensionality)
- Deliberately increasing bias (smoothing, e.g.) can reduce variance enough that overall error drops

Homework

- Register
- Register
- Register
- Read Manning and Schütze Chapters 2, 3, and 16 (up to page 389)
- Register