STA414/2104: Week 1

Introduction and Review

Alex Stringer

Jan 8th - 12th, 2018

Welcome

This course: STA414/2104, Statistical Methods for Data Mining and Machine Learning

Instructor: Alex Stringer

Course Information

Course information will be posted on Blackboard/Portal: https://portal.utoronto.ca

This includes the syllabus, lecture slides, assignments, and communications from your instructor (me).

Course Content

This course will be a sort of “crash course” in methodology, tools, and philosophy relating to machine learning.

There will be emphasis on the statistical considerations and implications of the methods- but the methods themselves are all prediction-focussed, and that is where we will spend the most time.

Course Content

From the syllabus:

- Linear methods for regression and classification
- Generative and Discriminative Models
- Regularization
- Model Comparison and Selection
- Optimization
- Neural Networks
- Kernel Methods and Gaussian Processes
- Mixture Models and the EM Algorithm
- Variational Inference

These topics are preliminary and subject to change

Course Content

These can roughly be split up into **methodology** and **tools**.

Tools are out-of-the-box algorithms that you can use on any new dataset. You will be doing this if you go into industry.

E.g. Logistic regression, neural networks

Methodology refers to general techniques used to build and develop new tools.

E.g. Optimization, variational approximation

Course Content

Not everything is so clear. Regularization is both a tool used to improve out of the box models, and a methodology for developing new models.

E.g. The *LASSO* (L1-regularized regression) spawned many years of ongoing theoretical development, which all started by adding a particular form of penalty to the linear regression log-likelihood (**methodology**).

But, regularization can be blindly used on any loss function (mostly). For example, most of the estimator objects in scikit-learn in Python have an option to apply different types of regularization.

Structure

The primary source of material will be lecture slides. These are a combination of slides created by Prof. Ruslan Salakhutidnov (adapted from Bishop (2006)), slides created by me, and code examples.

Lectures will be supplemented by not-for-credit assignments. These assignments will contain many questions, all of which you are strongly recommended to do. The midterm and final will be closely related to the assignments.

Evaluation

Undergraduate and Graduate Students:

- Midterm: 40%
- Final Exam: 60%

The midterm will be held as follows:

- Date: Tuesday, February 13th, 7:00 - 9:00 pm
- Location: TBA

Help with Course Material

Instructor office hours: Mondays, 11:00 am - 1:00 pm, my office ???

TA Office Hours: TBA

Piazza discussion board: piazza.com/utoronto.ca/winter2018/sta414lec5101

We encourage you to come to our office hours and discuss the course materials. Please come prepared with either specific questions about lecture materials, or partially worked out assignment questions. Please do not come with nothing and say "how do I do it" or "I don't get it"- you must show that you have made some effort to attempt the problem.

Prerequisites and Related Courses

Prerequisites:

- STA302: Regression and data anlaysis. I am assuming that you are at the point where you could take a small dataset and decide what model to fit, fit it, and interpret the results, without assistence
- CSC411: Similar material, more focus on computation. If you took this course, you may find significant overlap.

Similar Courses:

- CSC321: Neural Networks
- CSC412: More advanced material. If you took CSC411, you may find that CSC412 is a more appropriate choice than STA414.

Textbooks

No required textbook. A list of optional recommended references is available in the syllabus. Some of the books may be available legally online for free from the authors' websites.

Bishop (2006): *Pattern Recognition and Machine Learning*. Slides are based on this book. Focusses on prediction and mathematical derivations. Also includes cool applications, though the details are often eschewed.

Hastie, Tibshirani and Friedman (2009): *Elements of Statistical Learning*. If you are a stats student, I *strongly* recommend this book. This communicates the concepts that we will cover in a way that should be familiar to statisticians.

Statistics and Machine Learning

Statistics: models are interpretable representations of the data

- Concerned with *estimation* and *inference*
- Given data as output of a stochastic process, what can we *infer* about that process?
- “Inverse probability”
- Analyze methods, make assumptions, very concerned with systematic errors

Statistics and Machine Learning

Machine Learning: the 3 P's: Prediction, Prediction, and Prediction

- Prediction is the end goal
- Methods that generalize to a wide range of potential datasets
- Methods that are extremely flexible given massive amount of data
- Models are evaluated *empircally*: pick the model that gives the best predictions on the test set
- Often practitioners and researchers are concerned with the statistical implications of the methods (see *estimation*, *inference*, and *systematic errors* above), but this concern is usually framed in terms of how these aspects affect prediction

Example: Logistic Regression

Statistician:

- Data came from a binomial process → member of exponential family of distributions
- Identify the canonical link function that relates the natural parameter to the sufficient statistics → logistic transform
- This implies a loss function
- Fit this using Fisher Scoring (modified Newton's method)
- Look at parameter estimates and asymptotic standard errors → confidence intervals, hypothesis tests about variable significance
- Analyze deviance, deviance residuals, look for breaking of assumptions. Fix by modified the estimation procedure with things like overdispersion/quasi-likelihood, Generalized Estimating Equations

Example: Logistic Regression

Machine Learning Practitioner:

- Simplest baseline model depends on the features only through a linear function of features and parameters, $w'x$
- Need to transform this so that predictions lie in $(0, 1) \rightarrow$ logistic transform has good properties
- This implies a loss function
- Fit model using gradient descent (can view as a relaxed Newton's method)
- Evaluate the predictions on new data (validation set) by plugging into the model equation with the optimized parameter vector
- Look to extend model using non-linear transformations of the features, by combining many similar models, etc

Example: Logistic Regression

These approaches are nearly entirely different. So which is right?

They are **both** right, in the right context.

The Statistician's approach involves a lot of thought and effort into evaluating the quality of the inferences we make on each feature. If the features themselves are of primary interest, then this is a good approach.

The Machine Learning Practitioner's approach is very general, and will apply to a broad class of prediction problems and types of data. If the final predictions made by the model are of primary interest, then this is a good approach.

Often, both approaches give the same actual answer, and understanding both ways of thinking will set you ahead of the pack.

Types of Learning

Supervised Learning: Given input and output, build a model that will allow you to predict output given new input.

Unsupervised Learning: Given input only, build a model that finds some unknown structure in the data, e.g. clusters of similar points.

Semi-Supervised Learning: Small number of labelled training examples and a large number of unlabelled. Build a model that incorporates both data sources into the training procedure

These all involve estimating $P(Y|X)$ and/or $P(X,Y)$ and/or $P(X), P(Y)$.

Types of Learning

Discriminative Models: Estimate $P(Y|X)$ directly. Used to predict Y given new X. Does not describe the process that generated the observed data, $P(Y, X)$.

- Example: regression methods, neural networks

Generative Models: Estimate $P(Y, X)$. Use to obtain $P(Y|X)$. This provides predictions for new cases, *and* a means of *generating* new cases.

- Example: naive bayes, linear discriminant analysis

Uses of Machine Learning

Endless and rapidly growing. It's hard to read the news today without hearing about this stuff. Some specific examples:

Image Classifcation: Regard an image as an array of pixels. Use these as features. Label the image by its contents, e.g. "cat". Predict whether new image contains a cat based on its pixel features. Or, build a generative model for images than can generate new pictures of cats

- If you have Google Photos, you can type a word in the search box and pictures in your folder that contain the thing you typed appear

Uses of Machine Learning

Credit Scoring and Pricing: lenders build discriminative models of new and existing customers' probability of default on a new/existing loan. This is used directly to decide how much (if any) new credit you qualify for, what type of credit you qualify for, and your interest rate.

This is a tightly regulated practice (Basel, IFRS 9), but there is still a lot of discretion on the lenders' part. The quality of their models (and their modellers, which might be you next year!) affect the lives of millions of Canadians every day.

- If you go to the websites of TransUnion or Equifax Canada, you can look up your credit score. This is literally the output of a probabilistic model describing your likelihood of defaulting on a loan in a certain future time period.

Themes of the Course

Key learning Objective: Fitting Models to Data.

First half of the course: learn the philosophy and building blocks of machine learning

- Optimization, probability distributions, loss functions
- Feature transformations, nearest neighbours, Bayesian statistics

Second half will focus more on actual methods, both tools and methodology.

- Latent variables, neural networks, mixture models
- Model comparison, evaluation, selection

Learning Outcomes

To put it another way, here is what you should learn in this course:

- Standard algorithms, how to fit them to data, and how to decide between models in real problems
- Main elements in the language of machine learning and how to combine them to create new techniques
- Standard computational tools and how to use them to apply new and existing techniques to data

Computation

The not-for-credit assignments will ask you to write programs. You (obviously) may use whatever language you want. I recommend Python + Numpy for the type of numerical programming that will be required. I can help you in Python and R.

On the test and exam, you will not be asked to understand a particular programming language. You will be asked to perform steps of computational algorithms *by hand*, so don't think that this is somehow bonus material. Fitting models to real data is the key learning objective of this course.