A random walk through complex statistical learning problems

Hugh Chipman, Acadia University

---

1. Active learning with Bayesian Additive Regression Trees
2. Drug discovery problems
   - Bayesian LAGO
   - DD subspace models
3. Transactions on graphs/social networks
4. Unsupervised learning with curves

Joint work with Erika Nahm (M.Sc. candidate), Wanhua Su, Sofia Mosesova, Sunny Wang (Ph.D. candidates), Alberto Nettel-Aguirre (PDF), Will Welch, Mu Zhu, and others

---

Active learning with Bayesian Additive Regression Trees

"BART in one powerpoint slide":

$y = g(x; T_1, \Theta_1) + g(x; T_2, \Theta_2) + \ldots + g(x; T_M, \Theta_M) + N(0, \sigma^2)$ errors

- T_i is a decision tree
- Θ_i is terminal node parameters of T_i,
- x is an input vector.
- $g(x; T, \Theta)$ is a generic function that generates an output at $X = x$ with tree T and parameters Θ.

Bayesian formulation

- Parameters $T_1, \ldots, T_M, \Theta_1, \ldots, \Theta_M, \sigma$
- Posterior inference via MCMC - sample instances of model from posterior.
- Joint predictive distributions for response Y at multiple input points $X_{new,1}, X_{new,2}, \ldots$.

Active learning with Bayesian Additive Regression Trees

Active learning: We have response for some input values, and want to choose more inputs at which we measure the response.

Goal: By sequentially selecting observations, we want to build the most accurate model (for future data).

- Further, I'll assume a finite population of unlabeled observations, with known input values.
- Uncertainty should help active learning (It's a foundation of sequential experimental design).

Two approaches to *picking the next points:*

1. Pick the single point with the widest uncertainty bound for predictions (learn where we know least).
2. Pick the point that will give the largest expected improvement in the fit of the model. Expectation is over the unobserved response.

Active learning with Bayesian Additive Regression Trees

- Here, I'll focus on (1).
- A common problem with (1) is that it's ineffective for picking more than one point at a time. The two points with the greatest uncertainty may be quite close, and given that we observe the response for one point, the other may be redundant.
- You really should account for their covariance . Perhaps minimize $\text{Var}(Y_1 - Y_2) = \text{Var}(Y_1) + \text{Var}(Y_2) - 2\text{Cov}(Y_1, Y_2)$?
- **What I think is interesting:** Probability model and Bayesian machinery enables us to quantify uncertainty about the predictions our model makes. This seems an important ingredient in deciding what future points should be sampled.

Active learning with Bayesian Additive Regression Trees

Tough questions:

- As with other approaches, if the model has serious deficiencies, uncertainty bounds will not help.
- How do we do diagnostics for such a model? Especially if data are sparse? What if the model doesn't fit well in some places only?
- How do you control complexity of the model if the size of the training set is changing substantially?
- How to include information about the "data generation"? In sequential experimentation, analysis usually includes block effects for different experimental runs. Random effects might sometimes be appropriate (e.g. Protein Homology, KDD Cup 2004).
  - Interpreting the experimental conditions to identify such effects can be subtle.
  - Probability models are well-equipped to handle such "add-ons".
- Computational cost of second approach (integrating over future responses) makes it tricky. In principle it is possible, since conditional on each set of M trees, we have simple models.

Drug discovery problems

Past industrial partner: GlaxoSmithKline

Joint work with Will Welch, Mu Zhu, Sunny Wang, Wanhua Su.

General issues:

- descriptors of molecular structure
- active learning
- diversity of compounds desired
- supervised learning adapted to drug discovery context

I'll discuss two bodies of work in the last area.

Drug discovery problems

1. Bayesian formulation of LAGO model

(with Wanhua Su and Mu Zhu; Mu discussed LAGO earlier).

- Current LAGO model is more of an algorithm than a statistical model. For example, although LAGO output (prediction) is bounded by 0 and 1, it is clearly not a probability.
- Idea: formulate a probability model for response, using a logistic structure. Simplest case is

$$\text{logit}(\Pr(Y = 1)) = \beta_0 + \beta_1 S(x; k, \alpha)$$

  where $S(x; k, \alpha)$ is the usual LAGO prediction. We have four parameters to estimate $(\beta_0, \beta_1, k, \alpha)$.
- Basically a calibration of the LAGO model, with addition of probabilistic framework (could do active learning).
- MCMC likely used for computation.
- Further generalizations possible - for example every kernel in S could have it's own β coefficient.

Drug discovery problems

2. Classification via constrained mixture discriminant analysis

(with Sunny Wang, Will Welch)

Observation:

- Activity of a compound is believed to be governed by a few descriptors. But different compounds may be active due to different descriptors.
- Previous work (Marcia Wang) has successfully capitalized on this, by averaging across subspaces.

Mixture Discriminant Analysis:

- Mixture discriminant analysis idea is that conditional on the class label, distribution of inputs is modelled as a mixture distribution.
- Combine this with subspace idea: Each mixture component is characterized by class-specific parameters in a small subset of inputs; parameters for all other inputs are not class-specific.

$$f(x|y=k) = \prod_{j=1}^{p} \pi_{j,k} f(x_j;\theta_{j,k}) \prod_{l\neq j}^{p} f(x_l;\theta_l)$$

$x = (x_1, x_2, \ldots, x_p)$ is an observation in p-space, $\sum_{j=1}^{p} \pi_{j,k} = 1$.

Drug discovery problems

2. Classification via constrained mixture discriminant analysis

- Yields a model whose dimensionality grow more slowly with large descriptor sets than unconstrained MDA.
- Estimation issues are interesting, since the problem has unobserved mixture component labels, as well as depending on observed class labels.
- Broad "blue sky" observation: We're doing all of this because we want to use the (conventional-looking) matrix format supervised learning problems, rather than learning on the 3D molecule structure itself. It'd be nice if we could eventually do the 3D modelling directly.

Transactions on graphs/social networks

Industrial Partner: Government of Canada

Joint work with Alberto Nettel-Aguirre (PDF), Erika Nahm (M.Sc. candidate)

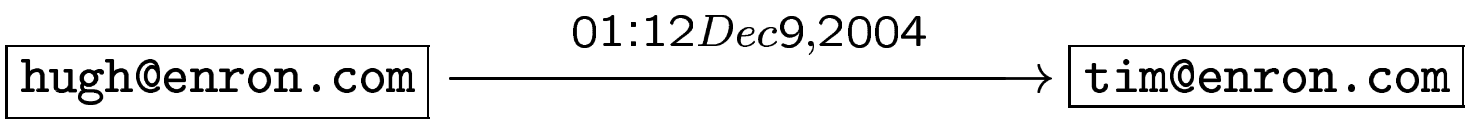
- Focus here on one public domain example: Enron dataset
  - Enron was a large US corporation implicated in various corporate wrongdoings. Some specific employees were charged and convicted.
  - As part of the legal proceeding, the US court system publicly released a large corpus of email messages between Enron employees, sent between 1998 and 2001.
  - 153 employees, about 600,000 messages.
  - **Our focus:** "header" information can be thought of as a transaction on a graph.
    - Sender, receiver are nodes, message is a directed edge, this "transaction" occurs at a specific time.

Transactions on graphs/social networks

Example

```
From:  hugh@enron.com
To:  tim@enron.com
CC: sue@com.com, ken@enron.com
Date:  01:12 Dec 9, 2004 (ADT)
Subject:  We're really in it now

Hi Tim, BlahBlahBlahBlah implicate BlahBlah criminal BlahBlahBlah....
```



Sender	Recipient	Time (sec. since 1998)	Type
hugh@enron.com	tim@enron.com	217598400	To
hugh@enron.com	sue@com.com	217598400	CC
hugh@enron.com	ken@enron.com	217598400	CC

Transactions on graphs/social networks

- Considerable data cleaning, most of it not interesting
- Note that we are breaking up a one-to-many transaction into a series of two-node transactions.

Questions of interest:

In all cases, we are seeing what can be extracted from the "header" information $\rightarrow$ more general format

Also, temporal structure of data interesting.

1. Supervised learning:
   - *Observation* is a node, *class label* may or may not be based on graph properties, *inputs* are based on node's "connection patterns".
   - Enron example: 153 observations (nodes), class label = employee rank (Senior Admin/regular employee), features include: sending frequency, receiving frequency, in-degree, out-degree, proportion of messages sent on weekend, variability in sending, etc...

Transactions on graphs/social networks

2. Semi-supervised learning:
   - Some nodes are labelled, many are not
   - Enron example: people who committed crime
3. Unsupervised learning:
   - What groups of nodes form communities (clusters)?
   - How does the behaviour of these clusters change over time?
   - How does the membership of these clusters change over time?
   - A person's communications are likely to be a mixture of communications with different groups (e.g. work-related vs. personal)
   - Probabilistic questions abound! What changes are more than random noise? Can we cluster probabilistically? Can we fit a mixture model to characterize communication mixture property?

Transactions on graphs/social networks

Specific approach for the unsupervised problem: Social network models

Hoff, Raftery and Handcock (JASA 2002,*Latent space approaches to social network analysis*)

- Assumes a static graph - each (directional) edge on the graph is binary (communication present/absent). More general data on edges possible (eg Poisson counts).
- Clustering nodes may fail because people don't just belong to one social group.
- Instead assume that each node occupies a position in a latent low-dimensional space, and tendency to communicate with another node is based on a measure of closeness in the latent space (and potentially other covariates).

Transactions on graphs/social networks

- Model for probability :

$$\text{logit}(Y_{ij} = 1 | z_i, z_j, x_{ij}, \alpha, \beta) = \alpha + \beta' x_{ij} + |z_i - z_j|$$

$y_{ij} = I(\text{edge between nodes } i \text{ and } j)$,
$z_i =$ position of node i in latent space
$x_{ij} =$ additional covariates on nodes i and j.

- This model says that nodes closer in latent space are more likely to talk.

Transactions on graphs/social networks

- Conceptually like simultaneously learning a multidimensional scaling (the Z's of the nodes, with the MDS positions determining edge activity.
- Bayesian formulation, MCMC algorithm used to get posterior on parameters and latent space positions.
- Interesting issues/questions:
  - Latent space can be used to interpret results. How many dimensions do we need in the latent space to get a good fit to the data?
  - Although this model is static, we could consider dividing time into chunks, fitting the model to each chunk, and studying how much the model parameters change over the different chunks.
  - Will posterior uncertainties characterize instability of the graph from data perturbation? Can posterior be used to decide what a "big change" in the parameters is?

Transactions on graphs/social networks

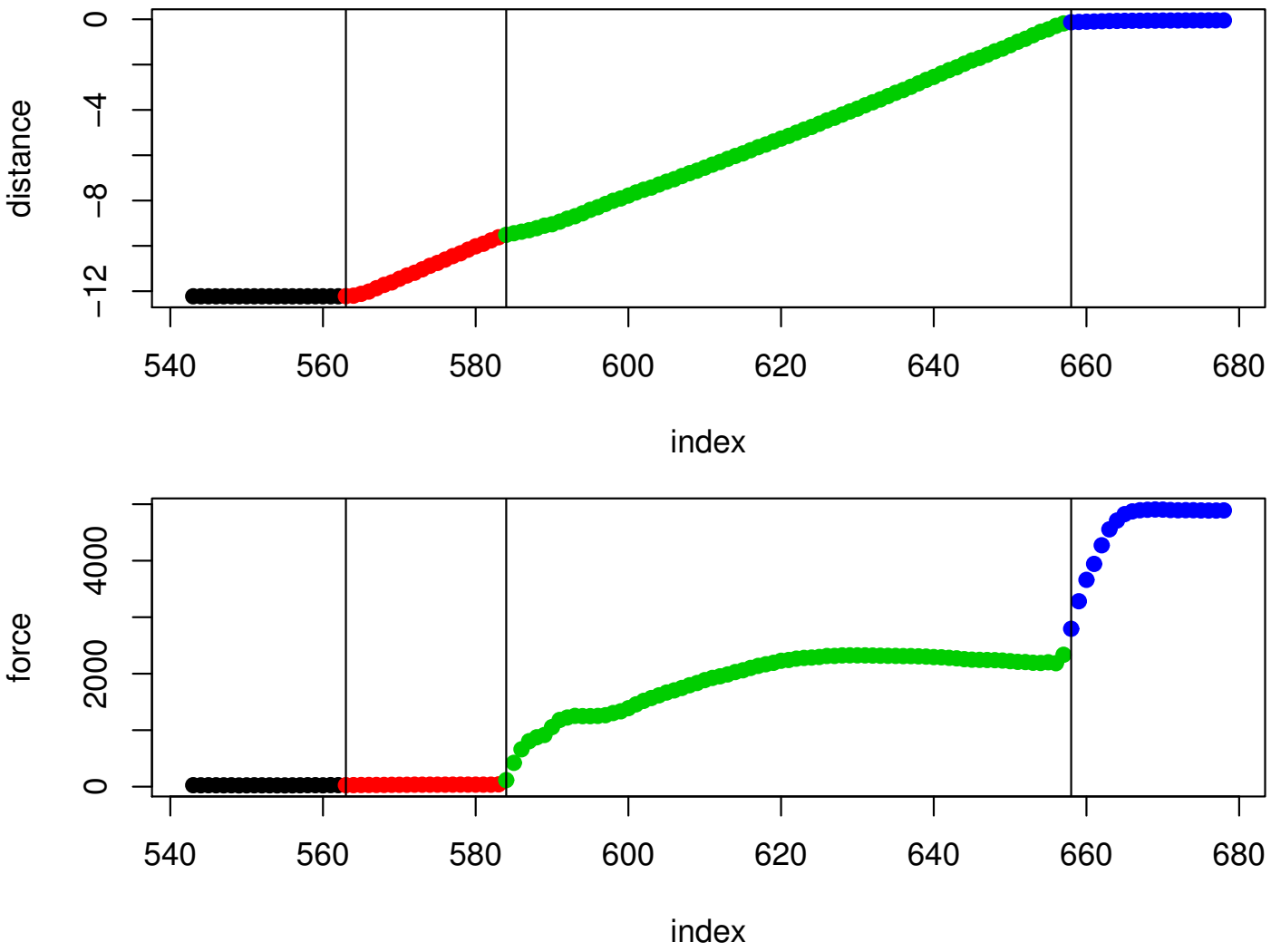
Some details on perturbation analysis

- We are analyzing the effect that small perturbations in communications have on the overall configurations.
- Assessing the 'goodness of fit' of the model to:
  - unperturbed data,
  - perturbed data,
  - model generated data.
    - Comparing probabilities of communication between i and j, ($P(y_{ij} = 1)$) obtained from 'true model' to those of simulated perturbed data.
    - Analysis of 'residuals' $y_{ij} - P(y_{ij} = 1)$ Wang & Wong (1987) , Holland & Lienhardt (1981).
- Assess changes in latent space configuration via correlation of internode distances for:
  - posterior mean positions on latent space,
  - MLE positions in latent space.

Unsupervised learning with functional data

(joint work with Sofia Mosesova)

- **Data example:** distance/time ($d(t)$) and force/time curves ($f(t)$) for manufacture of one part.
- 8 such insertions for one part, data for over 6000 parts
- What curves are similar in terms of shape?



Unsupervised learning with functional data

- Each curve represented by about 100 $f(t)$ values.
- **First attempt:** register curves, and treat each curve as a point in 100-dimensions. Cluster. Problem: highly correlated data
- **Second attempt:** reduce each curve to a lower dimension, by estimating a basis expansion (B-spline) and using the expansion coefficients as "data". Apply (model-based) clustering to the coefficients. Problems: still considerable correlation, and we ignore noise level in the original data.
- **Third attempt:** Formulate the whole thing as a hierarchical Bayesian mixture model, estimating parameters via MCMC. This gives posterior uncertainty bounds, and the ability to make inference about things such as cluster memberships, and cluster-specific parameters (central curves and modes of variation).
- Possible extensions include inference for the dimensionality of the basis function space

Unsupervised learning with functional data

Bringing more of the "data generation" into the model:

- Design structure: The curves are part of a 4-cylinder engine, with 2 different insertions (big/little) per cylinder. Error structures within the 8 insertions (or 2 groups of 4) likely to involve covariances. Could model this with random effects at the engine level? Alternately, we may want to focus on clustering curves after removing the effect of individual locations.
- Time-series structure: Engines are produced in sequence, so if there are problems with one part, others produced after may be more likely to have problems. Model the evolution of parameters over time?

So what?

Yoshua's challenge to us:

> maybe most importantly, present the kinds of questions that you would like to attack, your philosophical stance about what important problems you think we should work on, and what expertise you think you need to make progress wrt these questions.

So what is important to me?

- Uncertainty: how to represent it, cope with it (full Bayes machine vs. cross-validation), decide which uncertainties are unimportant, all in the context of complex models for complex data structures.
- Algorithms that serve to learn models (with uncertainty) from data. For me, this is often MCMC.
- Inclusion of "data generation" in the model - models that enable it, and "good statistical practice"
- Specifics:
  - Interesting ways to use uncertainty, such as active learning.
  - Unsupervised learning for complex, high-dimensional structures, using probabilistic mixture models.