Modern Statistics 52311 - 2014-15 Final Project

June 21, 2015

To get full credit for the course, you are required to complete and present a final project.

The project can be theoretical (propose a a new statistical model, a new algorithm, or obtain new results on the performance of an existing algorithm) or more practical (test and compare different methods on real datasets, analyze results and conclude on the performance of different algorithms in this domain). You can work alone or in pairs.

Project should consist of developing a new method, or studying the properties of an existing method, or an application of an existing method. You should apply and test your method on one of the project's datasets.

You can choose one of the projects presented below, or propose a project related to your own interests, provided that it is related to the course material, and can be applied to one of the datasets proposed. If you're not sure, you can also propose a general subject from the course you're interested in (e.g. 'wavelets', or 'regression with cross-validation' or 'dimensionality reduction') and we will try to find a related project. In any case, we will discuss the project details together and I will approve it.

Dates:

Submit a detailed proposal and preliminary results by July 1st.

Final project is due to end of July. Upload a report describing your work and results. Please upload your code, documents, and additional datasets you've used (including simulated datasets). We will also have a poster session where each person/pair can present their work.

Grade:

If you complete the minimal requirements we agree on in the proposal correctly, your grade for the project will be 85. Grade beyond 85 will be given according to quality of results, poster presentation, code contributions generalizations etc.

1 Technicalities

We've opened a project in github for the project, at : `https://github.com/orzuk/course_52311`

Datasets:

There are two datasets available in the project's git repository:

1. A gene-expression dataset, in the GTEx subdirectory. In this data, gene expression was measured for multiple individuals in multiple tissues. The data can be represented as a 3-dimensional array, where $D(i,j,k)$ is the gene-expression level of gene i at individual j and in tissue k.

2. There is a video mp4 file in the Video subdirectory. The data can be represented as a 3-dimensional array, where $D(i,j,k)$ is the intensity level of pixel (i,j) at frame (time) k.

Code: There is some preliminary code in Matlab at the src directory which may help you read the data. If you add code for reading/manupulating the data, please add it to the repository.

Organization You will need to write code, documents etc. Please create a directory under your name under 'docs' with a proposal providing a short description of your project. When you write code - please create a directory under your name under 'src' with your code - if you think your code can be of usage to others, move it to the 'src' directory (or possibly to the appropriate language, for example: 'src/R')

Proposed Projects

1. **Testing Higher-order interactions Permutation Testing:**

   The goal of this project is to develop efficient non-parametric statistical tests for detecting higher order dependencies.

   Suppose $X_1, ..., X_n$ are random variables. Let k be the interaction order we are interested in. We are interested in detecting k-wise interactions for $k > 2$ (e.g. $k = 3$). It is possible that a set of k random variables are dependent, but any subset of $k-1$ is independent (or close to independent), and therefore only testing for k-th order interaction will reveal the dependency between the random variables. We assume here that we have a black-box test for testing weather a set of variables $X_1, .., X_k$ are dependent. The problem is that enumerating all $\binom{n}{k}$ subsets of size k may be computationally too costly. We can overcome this problem by noticing that if a subset of a set of variables is dependent, then so is the entire set. Therefore we can take subsets of c variables, for some $c > k$ such that all subsets of k variables are covered. Testing independence for all subsets in our

collection will enable us to discover the interactions. The goal of the project is to analyze this approach in details:

(a) For a given triplet of values (n, k, c) design a 'cover', that is a set of subsets $S_1, ..., S_M \subset 1, .., n$ with $|S_i| = c$, such that, if possible, any subset of size k of $1, .., n$ is contained in a set S_j in the cover. Calculate the size of the design (number of sets to be tested) and the coverage probability (probability that a given subset of size k will be covered). You can try both randomized and deterministic designs

(b) Modify and implement the dCov ([6, 7]) and HHG ([4]) multidimensional tests for independence to test for dependence of a set of k scalar random variables (you can use existing R packages).

(c) Simulate data from a joint distribution over $X_1, ..., X_n$ which have strong k-th order interactions (but not lower order)

(d) Study the trade-off between number of tests performed and discovery power. (i.e. for a given k, compare different values of c in terms of the computational complexity required and the power to detect association).

(e) Apply the test on one of the dataset - find pairs of genes which have significant dependency across the samples

2. **Testing Independence using random projections:**

In this project we want to test dependency between X and Y, where X and Y are high-dimensional vectors. We can apply the tests learned in class (e.g. dCov ([6, 7])), but in the project we want to study an alternative method using projections of X and Y to lower dimensional spaces (see here [3]). The simplest method is performing multiple random projections of X and Y.

(a) Simulate data from different dependency types for vectors.

(b) Use very low dimensional projections (e.g. univariate), test for independence using a univariate test, and compare them to the high-dimensional test (e.g. dCov and HHG) in terms of speed (projections should be of course faster but if you need to perform many projections to gain power they may be heavy computationally) and power (which will perform better?).

(c) Compare also projections to intermediate dimension (maybe good to project from n to say log n)

(d) Employ also non-random projections which may be better at detecting the signal: (to leading PCA for each, or to the Canonical Correlation Analysis subspaces) you can

always control for using a favored subspace using permutation tests. Compare power and speed to the randomized projection approach. (note that hear computing each projection is more costly)

3. **Improved False-Discovery-Rate with mean and variance of effect size:**

   In most FDR method, the input to the method is only the set of p-values, $p_1, ..., p_m$. However, usually we have access to more than just the p-values, and would like to use additional data to improve inference. Here we assume that for each hypothesis i we have the estimated effect size $\hat{\beta}_i$ and standard deviation of effect size $\hat{\sigma}_i$. We treat the effects as Gaussians and can compute a normalized z-score $z_i = \frac{\hat{\beta}_i}{\hat{\sigma}_i}$, and from this compute a p-value, $p_i = \Phi(\frac{z_i}{\sigma_i})$.

   For example, for two hypotheses i and j we may have exactly the same p-value (say $p_i = p_j = 0.001$) but for the first hypothesis we have a big effect size and a small sample size, whereas for the second hypothesis we have a small effect size and a large sample size. The information we have for these two is different.

   Following Stephens' new proposal:
   `https://github.com/stephens999/ash/blob/master/talks/BOG2014-slides-MS.pdf`
   our goal is to develop an empirical Bayes' procedure taking into account both the mean effect size and the standard deviation for each hypothesis. We model the effect size β_i, rather than the z-score z_i.

   (a) Develop the formulation of the empirical Bayes' procedure using $\hat{\beta}_i$ and $\hat{\sigma}_i$

   (b) Implement the empirical Bayes' procedure. Then simulate data from different distributions (both independent and dependent) and apply the Benjamini-Hochberg procedure, the empricial Bayes' procedure due to Efron (using only the p_i) and the empirical Bayes' procedure using both $\hat{\beta}_i$ and σ_i.

   (c) Analyze empirical data using the three methods and compare power.

   (d) Suppose that we can further estimate the second moment of the **joint** distribution of effect sizes, i.e. we can compute the correlations $\sigma_{ij} = corr(\beta_i, \beta_j)$. Develop and implement the empirical Bayes' procedure for this case, and compare the approach to the previous approach (ignoring the correlation structure) on simulated data.

   Additional reading - a description of the approach by Matthew Stephens:
   `https://github.com/stephens999/ash/blob/master/talks/BOG2014-slides-MS.pdf`
   and more infor in general in the ash package: `https://github.com/stephens999/ash`

4. **High Dimensional Regularized Poisson Regression in the Simplex:**

   In this project the goal is to study the following Poison regression model:

$$Y|X \sim Poiss(X\beta) \tag{1}$$

   Here $X \in \mathbb{R}^p$ is a vector of features, and Y is a count data vector of length n for $n < p$.

   We perform the following regularization:

$$\hat{\beta} = argmin_{\beta \in \Delta^n} - logP(Y|\beta, X) + \lambda||\beta||_1 \tag{2}$$

   where $P(Y|\beta, X)$ is the likelihood of Y (according to the Poisson model) and Δ^n is the n-dimensional simplex. (note that this is a convex set).

   The motivation comes from next-generation sequencing experiments in genomics, where β represents a vector of frequencies of different molecules (hence $\beta_j \geq 0$ and $\sum_j \beta_j = 1$ the vector y represents the count observed for each sequence read, and X the design matrix is given, representing which reads can be obtained from appear on with molecule (see [8] for a description of the problem and a simplified statistical model and treatment)

   (a) Develop and implement the coordinate-descent algorithm for this model (see glmnet package). Your solver should output the entire regularization path.

   (b) Generalize the optimization problem to get the generalized lasso problem:

$$\hat{\beta} = argmin_{\beta \in \Delta^n} - logP(Y|\beta, X) + \lambda||G\beta||_1 \tag{3}$$

   where G is a given graph (in the application, G represents a phylogenetic distance between different molecules).

   (c) Simulate sequence data according to the Poisson model, fit the model and measure training and test errors

   (d) The matrix X and the graph G are sparse. Modify the implementation of your algorithms to exploit this sparsity. Study the improvement in performance.

   (e) Run algorithms on real-data (will be provided) and analyze results

5. **Translation-invariant WHMT:**

   (Remark: This project may be suitable for those with strong background and interest in graphical models)

We have learned in the course how wavelets-shrinkage can be improved by using a Markovian structure on the wavelets coefficients using a hidden Markov tree (see [2],[5]). In addition, we have seen that efficient translation-invariant denoising leads to improved performance [1]. The goal of this project is to unify these two improvements

(a) Implement a 'standard' WHMT model, and an (non efficient) enumeration on cyclic shift of the data to obtain a translation invariant WHMT

(b) Simulate noisy data obtained from several smooth curves, and denoise it using the standard and translation invariant WHMT. Compare results of both models.

(c) Write a graphical model representing all wavelets coefficients relating to all possible shifts of the data

(d) Generalize the EM algorithm used for WHMT to the translation-invariant case. The graph you obtain will contain loops, so you will need to implement an approximate inference technique for a loopy graph (e.g. loopy belief propagation)

(e) Compare the naive enumerative approach to the approximate inference approach in terms of both speed and accuracy

I have code which I can provide for fitting the WHMT model (without translation-invariance) using the EM algorithm, which you can use.

6. **Time-Changing PCA:** In this project the goal is to learn a dimensionality reduction of a dataset, which may change with time. That is - consider K different datasets $X_1, ..., X_K$ - each a data-matrix. We can compute the top principal components of each dataset separately. However, we would like to use information from multiple samples in order to estimate the principal components.

References

[1] Ronald R Coifman and David L Donoho. Translation-invariant de-noising. *Lecture Notes In Statistics-New York-Springer-Verlag*, pages 125–150, 1995.

[2] Matthew S Crouse, Robert D Nowak, and Richard G Baraniuk. Wavelet-based statistical signal processing using hidden markov models. *Signal Processing, IEEE Transactions on*, 46(4):886–902, 1998.

[3] Juan Antonio Cuesta-Albertos, Eustasio del Barrio, Ricardo Fraiman, and Carlos Matrán. The random projection method in goodness of fit for functional data. *Computational statistics & data analysis*, 51(10):4814–4831, 2007.

[4] Ruth Heller, Yair Heller, and Malka Gorfine. A consistent multivariate test of association based on ranks of distances. *Biometrika*, 100(2):503–510, 2013.

[5] Justin K Romberg, Hyeokho Choi, and Richard G Baraniuk. Bayesian tree-structured image modeling using wavelet-domain hidden markov models. *Image Processing, IEEE Transactions on*, 10(7):1056–1068, 2001.

[6] Gábor J Székely, Maria L Rizzo, Nail K Bakirov, et al. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794, 2007.

[7] Gábor J Székely, Maria L Rizzo, et al. Brownian distance covariance. *The annals of applied statistics*, 3(4):1236–1265, 2009.

[8] Or Zuk, Amnon Amir, Amit Zeisel, Ohad Shamir, and Noam Shental. Accurate profiling of microbial communities from massively parallel sequencing using convex optimization. In *String Processing and Information Retrieval*, pages 279–297. Springer, 2013.