Conference on Nonparametric Statistics and Statistical Learning
Information Science and Technology

The LASSO-Patternsearch Algorithm: Univariate and Multivariate Bernoulli Patterns of Inputs and Outputs

Grace Wahba

Joint work with Weiliang Shi, Steve Wright, Kristine Lee, Ronald Klein and Barbara Klein in Statistics and Its Interface (2008) and work in progress including Xiwen Ma, Bin Dai and Shilin Ding.

Ohio State University, Columbus, May 19-22, 2010

These slides at
`http://www.stat.wisc.edu/~wahba/` → TALKS

Papers/preprints including this one at
`http://www.stat.wisc.edu/~wahba/` − > TRLIST

Abstract

We describe the LASSO-Patternsearch (LPS) algorithm, a two or three step procedure whose core applies a LASSO (ℓ_1 penalized likelihood) to Bernoulli ($\{0, 1\}$) response data y given a very large attribute vector x from a sparse multivariate Bernoulli distribution. Sparsity here means that the conditional distribution of y given x is assumed to have very few terms, but some may be of higher order (patterns). An algorithm which can handle a very large number (2×10^6) of candidate terms in a global optimization scheme is given, and it is argued that global methods have a certain advantage over greedy methods in the variable selection problem. Applications to demographic and genetic data are described. Ongoing work on correlated multivariate Bernoulli outcomes including tuning is briefly described.

Outline

1. The LASSO - ℓ_1 penalties.
2. The multivariate Bernoulli distribution: Bernoulli response $y \in \{0, 1\}$), Bernoulli attributes $x \in (\{0, 1\}, \cdots, \{0, 1\})$.
3. Tuning: GACV and BGACV for Bernoulli responses.
4. LASSO-Patternsearch (LPS) core algorithm for global variable selection.
5. The post LASSO step.
6. Simulations: global vs greedy algorithms.
7. Examples: demographic study, genetic data.
8. Summary and conclusions.

The LASSO

Given $y_i, x(i)$, where $x(i) = (x_1(i), \cdots, x_p(i)$, we find $f = f(x)$ to minimize

$$I_\lambda(y, f) = \mathcal{L}(y, f) + \lambda J(f)$$

where $\mathcal{L}(y, f)$ is $\frac{1}{n}$ times the negative log likelihood of y given x (or some other measure of the fit of y to f), $J(f)$ is a penalty functional on f and λ is a tuning parameter. If

$$f(x) = \mu + \sum c_\ell B_\ell(x),$$

for some specified basis functions, and the penalty functional is

$$J(f) = \sum_\ell |c_\ell|,$$

then the solution is called the LASSO (Tibshirani, 1996).(A similar idea was proposed by Chen, Donoho and Saunders, 1998).

Copious literature exists for the LASSO with Gaussian data

$$\mathcal{L}(y, f) = \sum (y_i - f(x(i))^2,$$

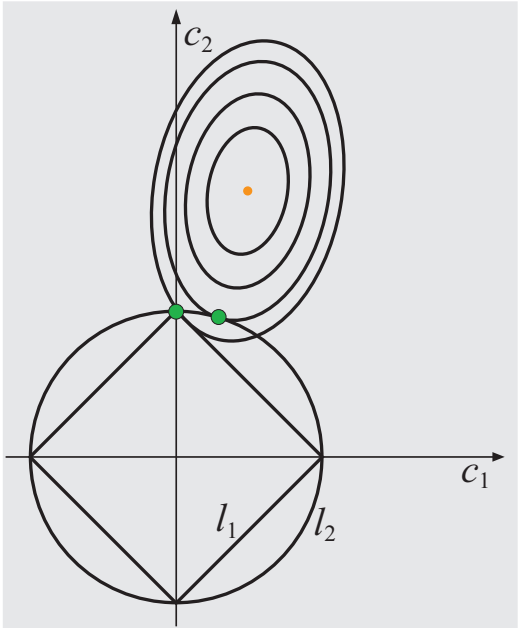
as well as for the support vector machine

$$\mathcal{L}(y, f) = \sum (1 - y_i f(x(i))_+.$$

However there are relatively few results for Bernoulli data, $y_i \in \{0, 1\}$, with the log likelihood

$$\mathcal{L}(y, f) = \sum_{i=1}^{n} [-y_i f(x(i)) + \log(1 + e^{f(x(i))})],$$

specifically with respect to choosing the tuning parameter λ.



The ℓ_1 penalty $\sum_\ell |c_\ell|$ is known to give a sparse representation - as λ increases, an increasing number of the c_ℓ become 0. This is in contrast to the ℓ_2 penalty $\sum_\ell c_\ell^2$, in which, typically the c_ℓ tend to all be non-zero. The choice of λ is an important issue in practice.

The Multivariate Bernoulli Distribution

Let $(x_0, x_1, \cdots, x_p)$ be a $p+1$ dimensional vector of possibly correlated Bernoulli random variables. The most general form $p(x_0, x_1, \cdots, x_p)$ of the joint density is (Whittaker 1990)

$$p(0,0,\cdots,0)^{[\pi_{j=0}^{p}(1-x_j)]} p(1,0,\cdots,0)^{[x_0\pi_{j=1}^{p}(1-x_j)]} \cdots p(1,1,\cdots,1)^{[\pi_{j=0}^{p}x_j]}.$$

We can extract from this the conditional logit f, which appears in the likelihood

$$f(x) = \log\left(\frac{prob(y=1|x)}{(1-prob(y=1|x)}\right)$$

as

$$f(x_1, \cdots x_p) = \log p(1, x_1, \cdots, x_p) - \log p(0, , x_1, \cdots, x_p).$$

After some reorganization the logit has the representation

$$f(x_1, \cdots, x_p) = \mu + \sum_{j=1}^{p} c_j B_j(x) + \sum_{1<j<p} c_{jk} B_{jk}(x) + \cdots + c_{12\ldots p} B_{12\ldots p}(x)$$

where $B_j(x) = x_j, B_{jk} = x_j x_k$, and so forth, and the c's are parameters to be estimated. Note that $B_{j_1 j_2 \cdots j_r}$ is 1 if $x_{j_1}, x_{j_2}, \cdots, x_{j_r}$ are all 1 and 0 otherwise.

Patterns

We will call $B_{j_1 j_2 .. j_r}(x)$ an rth order pattern. Let q be the highest order considered. Then there will be $N_B = \sum_{\nu=0}^{q} \binom{p}{\nu}$ patterns. If $q = p$, there is a complete set of $N_B = 2^p$ patterns (including the constant function μ), spanning all possible patterns. If $p = 7$, say then there will be $2^7 = 128$ patterns and standard software can be used to solve the Bernoulli LASSO problem. If $p = 1,000$ there are over half a million unknown coefficients just considering "main effects" and order 2 patterns, so special purpose software is required. To consider all possible patterns of order three requires preprocessing to reduce the number of candidates. In any case, however, the ℓ_1 penalty, with λ chosen with variable selection in mind, is ideally suited to find those (assumed) small number of patterns which influence y.

Tuning

The Comparative Kullback-Liebler (CKL) distance: Letting f and f_λ be the true and estimated parameters of the distribution, then the KL distance, defined as

$$KL(f, f_\lambda) = E_f \log\left(\frac{G(y, f)}{G(y, f_\lambda)}\right)$$

is a measure of how close the fitted distribution $G(y, f_\lambda)$ is to the true, but unknown distribution $G(y, f)$. $CKL(\lambda)$ is $KL(\lambda)$ minus any terms not dependent on λ. In the Gaussian case with known σ^2 $CKL(\lambda) = \sum \frac{1}{2\sigma^2}(f(x(i) - f_\lambda(x(i))^2$, a. k. a. $PMSE$. In the Bernoulli case

$$CKL(\lambda) = \sum -p(x(i))f_\lambda(x(i)) + log(1 + e^{f_\lambda(x(i))}).$$

where $p = e^f/(1 + e^f)$.

In the Gaussian case, when $y = f(x) + \epsilon$ with $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ with σ^2 known, there is the well known unbiassed risk estimate for $PMSE(\lambda)$ (Mallows(1973), Hudson(1978)). In Poisson, Gamma, Binomial with $m > 2$ and other distributions, unbiassed estimates of $CKL(\lambda)$ are well known, (Hudson(1978),Wong(2006)). In the Gaussian case with σ^2 unknown, an exact unbiassed estimate for $PMSE(\lambda)$ is not available but a good approximation to the minimizer is available (GCV, for example). Similarly, in the Bernoulli case Wong shows that there is no unbiassed estimator for $CKL(\lambda)$ but a good approximation for finding the minimum (GACV) has been given in Xiang and Wahba(1996) based on a leaving-out-one argument.

The $GACV$ was derived from a leaving-out-one argument on the CKL. In the LASSO-Patternsearch case it has a simple form:

$$GACV(\lambda) = \sum_i -y_i f_\lambda(x(i)) + log(1 + e^{f_\lambda(x(i))}) + trH \frac{\sum_i y_i(y_i - p_\lambda(x(i)))}{(n - N_{B*})}$$

where $H = B*(B*' W B*)^{-1} B*'$ with $B*$ is the design matrix for the basis functions in the fitted model and W is the $n \times n$ diagonal matrix with ii th element the estimated variance $\sigma_\lambda(i)$. N_{B*} is the number of non-zero coefficients in the fitted model.

BGACV

It has long been understood in the Gaussian data-quadratic penalty case that tuning for $PMSE$ is not the same as tuning for variable selection, indeed, this is the difference between the well known AIC (prediction) and BIC (variable selection with prior belief in sparsity). (George(2000), Leng, Lin and Wahba(2006)). Where the AIC has a 2 in front of the degrees of freedom, the BIC has a $\log n$. For $\log n > 2$, BIC gives a bigger λ. For the present work being specifically targeted to obtain sparse models, will will do the same thing, giving

$$BGACV(\lambda) = \sum_i -y_i f_\lambda(x(i)) + log(1+e^{f_\lambda(x(i))}) + \frac{\log n}{2} trH \frac{\sum_i y_i(y_i - p_\lambda(x(i))}{(n - N_{B*})}$$

Steve Wright's LPS code:

http://pages.cs.wisc.edu/~swright/LPS/

LASSO-Patternsearch

Matlab software implementing the algorithm in the paper

W. Shi, G. Wahba, S. J. Wright, K. Lee, R. Klein, and B. Klein, "LASSO-Patternsearch Algorithm with Application to Ophthalmology and Genomic Data," *Statistics and its Interface* 1 (2008), pp. 137-153.

The code does l1-regularized linear logistic regression with on data with Bernoulli outcomes (indicated by +/-1). The algorithm uses a gradient projection / iterative shrinkage approach, with gradient sampling and a modified reduced Newton scaling technique on the space of nonzero variables.

The original code was written initially by W. Shi and S. Wright in 2006-2008 and was rewritten for distribution in 2008-2010 by S. Wright.

Global LPS Algorithm

(2×10^6 unknowns.) The minimization problem has the form

$$\min_c I_\lambda(y, f) \equiv \min\ I_\lambda(c) := F(c) + \lambda \|c\|_1.$$

We describe a separable approximation algorithm with reduced Newton acceleration. Generates a sequence c^k, $k = 1, 2, \ldots$.

At iteration k, form a linearized model with damping and solve for step d:

$$\min_d\ F(c^k) + \nabla F(c^k)^T d + \frac{1}{2}\alpha_k d^T d + \lambda \|c^k + d\|_1.$$

Larger $\alpha_k \Rightarrow$ shorter step.

Can be solved trivially ($O(n)$ operations) as it is separable in the components of d.

Get an estimate of the basis (set of nonzero indices):
$\mathcal{B}_k := \{i \,|\, c_i^k + d_i = 0\}$.

Keep doubling α_k until the step gives a "sufficient decrease" in I_λ:

$$I_\lambda(c^k + d) \leq I_\lambda(c^k) - 10^{-3}\alpha_k d^T d.$$

If $\mathcal{B}_k$ not too large (less than 500 elements, say), try to enhance by taking a reduced Newton step in the $\mathcal{B}_k$ components. Define

$$H_k := \left[\frac{\partial^2 F(c^k)}{\partial c_i \partial c_j}\right]_{i\in\mathcal{B}_k, j\in\mathcal{B}_k}.$$

Compute reduced Newton step $d_{\mathcal{B}_k}$ by solving

$$(H_k + \delta_k I)d_{\mathcal{B}_k} = -[\nabla F(c^k)]_{\mathcal{B}_k},$$

for a damping parameter δ_k that shrinks to zero as c^k approaches the solution.

Replace the $\mathcal{B}_k$ components of d by the reduced Newton step and do a line search. Revert to original step if the enhanced step fails to give a sufficient decrease.

Get further savings by not evaluating the whole vector $\nabla F(c^k)$.

At iteration k, define working set $\mathcal{G}_k$ to contain

- components i with $c_i^k \neq 0$; and
- some random fraction of the other components (say, 1% or 5%).

Evaluate only the $\mathcal{W}_k$ components of $\nabla F(c^k)$. In the step calculation, set $d_i = 0$ for $i \notin \mathcal{W}_k$.

To solve for 21 values of λ on a 3 GHz dual-core PC reqired

- a few seconds for 128 basis functions
- 4.5 minutes for 403,000 basis functions.

LPS Code and Ongoing Work

Code available from `http://www.cs.wisc.edu/~swright/LPS`

Distributed version requires the full design matrix to be supplied explicitly.

Recent enhancements: Replace the reduced Hessian H_k by an estimate based on a random sample of the terms in the summation. Gives fast linear convergence at much reduced cost.

Ongoing: Many enhancements planned to algorithm and code, driven by needs of large problems.

Step 2: Parametric Linear Logistic Regression

The N_B* patterns surviving the LASSO step are entered into a linear logistic regression model using `glmfit` in MATLAB and final pattern selection is then carried out by backward elimination. (Step 2). This is a heuristic step which has given us good results in a large number of simulations and a small number of applications. It was initially motivated by observing that putting the entire set of N_B* surviving patterns in a linear logistic regression resulted in patterns whose coefficients were not significantly different from zero. One of the N_B* patterns is removed, the model is fit, and the $BGACV$ score is computed. The pattern that gives the best score to the model after being taken out is removed from the model. This continues to the end and the final model is chosen from the pattern set with the best tuning score.

Global vs. Greedy Algorithms

Conjecture: It is harder to pick out the right patterns if variables that are not directly related to the response are more highly correlated with those that are. Loosely speaking this is verified by the "Irrepresentable Condition" of Zhao and Yu(2006), which can be thought of as a measure of how much unimportant variables are correlated with important ones. This condition is almost necessary and sufficient for the oracle property.

Conjecture: The LASSO-Patternsearch algorithm does better than greedy competitors in these hard cases.

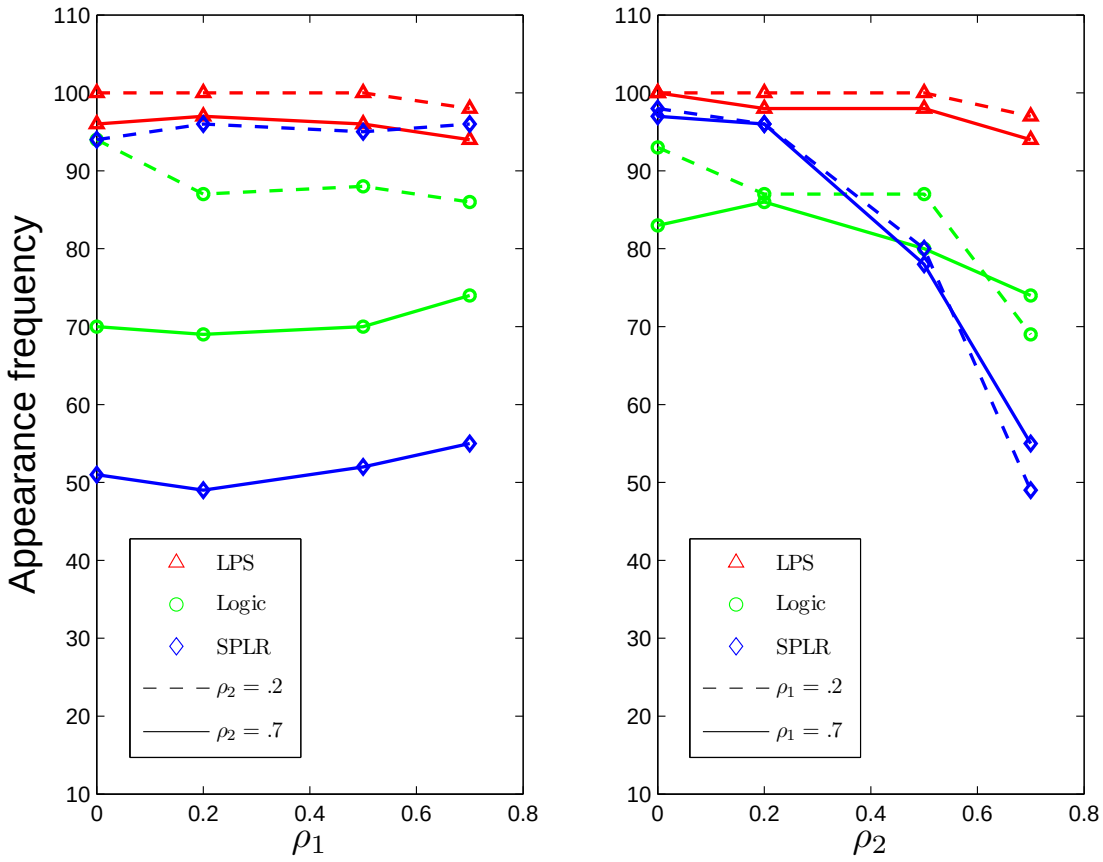
Correlated attribute vectors, an experiment

Sample size $n = 2000$, $p = 20$ variables, $x = (x_1, \cdots, x_{20})$.

$$f(x) = -2 + 2B_9(x) + 2B_{67}(x) + 2B_{1234}(x),$$

so there are three patterns relevant to y. Variables x_1, x_2, x_3, x_4 pairwise correlated among each other according to ρ_1. Variables x_5, x_6, x_7, x_8 are pairwise correlated between important and unimportant variables, x_5 with x_1, and so forth, according to ρ_2. Variables $9, \cdots, 20$ are uncorrelated. All patterns up to order 4 are tentatively in the model, which results in $N_B = 6196$ basis functions.

The next slide compares LASSO-Patternsearch, Logic Regression (Ruczinski, Kooperberg and LeBlanc(2003)) and SPLR (Stepwise Penalized Logistic Regression, Park and Hastie(2008)) according to their ability to select B_{1234}.



Correlation Experiment

Number of times in which B_{1234} appears in the fitted model, 100 replications. Left: LASSO-Patternsearch, Logic Regression, and SPLR, as it varies with ρ_1 (within) for two values of ρ_2 (between). Right: Same data as ρ_2 varies, for two values of ρ_1 (dashed = low, continuous = high). LPR is robust against high ρ_1 and ρ_2.

Details for the simulation example

Table 1: In each row of a cell the first three numbers are the appearance frequencies of the three important patterns and the last number is the appearance frequency of patterns not in the model. Summary: LR and especially SPLR generate irrelevant patterns.

$\rho_2 \backslash \rho_1$		0	0.2	0.5	0.7
0	LPS	96/100/100/54	98/100/100/46	100/100/100/43	100/100/100/44
	Logic	100/98/96/120	98/95/93/107	99/94/92/83	100/98/83/134
	SPLR	100/100/100/527	100/100/98/525	100/100/98/487	100/100/97/489
0.2	LPS	99/100/100/46	100/100/100/49	100/100/100/39	100/100/98/36
	Logic	99/97/94/96	100/99/87/94	100/100/88/73	100/99/86/117
	SPLR	100/100/94/517	100/99/96/530	100/97/95/495	100/100/96/485
0.5	LPS	99/100/99/47	99/100/100/51	100/100/99/51	100/100/98/46
	Logic	99/96/86/162	100/95/87/109	100/96/78/122	100/99/80/143
	SPLR	100/98/75/548	100/96/80/552	100/99/80/531	100/98/78/518
0.7	LPS	100/99/96/44	99/99/97/51	100/99/96/67	100/99/94/65
	Logic	100/83/70/195	100/88/69/167	100/85/70/153	100/89/74/126
	SPLR	100/91/51/580	100/85/49/594	100/81/52/584	100/72/55/570

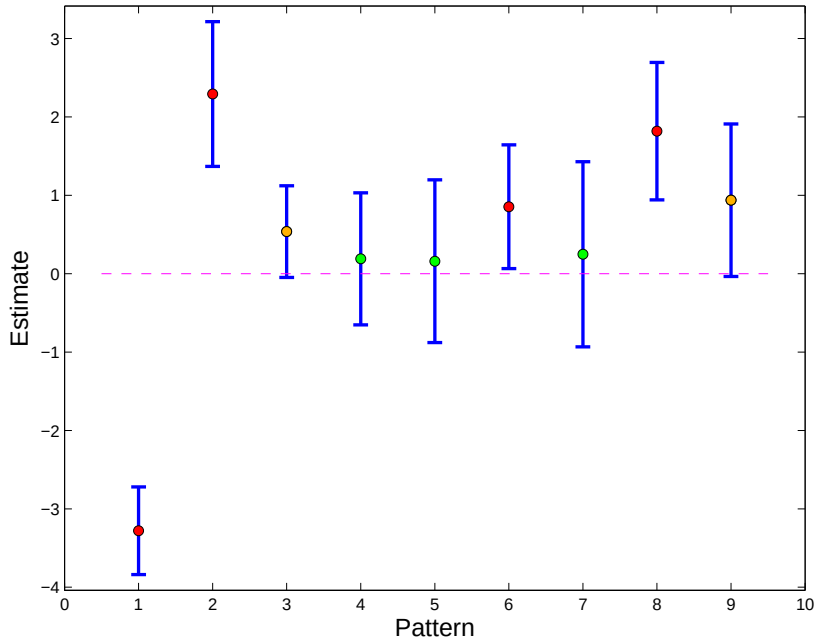
Example 1: Risk of myopic change in a demographic study.

Applied to to "myopic change" from the Beaver Dam Eye Study, BDES 1 to BDES 2, five years apart. $n = 876$ records of persons aged 60-69 at BDES 1. A person whose "worse eye" scored at a decrease of .75 diopters or more is labeled $y = 1$, and 0 otherwise. Which variables or clusters of variables are predictive of this outcome? Consider seven variables of possible interest and want to see if there are high order interactions among the variables. The continuous variables are dichotomized so as to be able to do this.

Table 1: Trial Variables and Cutpoints

	variable	description	binary cut point (higher risk) ($X = 1$)
X_1	sex	sex	Male
X_2	inc	income	< 30
X_3	jomyop	juvenile myopia	< 21
X_4	catct	cataract	4-5
X_5	pky	packyear	>30
X_6	asa	aspirin	not taking
X_7	vtm	vitamin	not taking

There are 2^7 possible subsets (clusters) of variables that could be important.



Eight patterns (plus μ) that survived the LASSO (Step 1), entered in a parametric linear logistic regression. Patterns numbered 2,6,8 and 9 passed Step 2, giving the final model

$$\begin{aligned} f &= -2.84 + 2.42 \cdot catct + 1.11 \cdot pky \cdot vtm \\ &\quad +1.98 \cdot sex \cdot inc \cdot jomyop \cdot asa + 1.15 \cdot sex \cdot inc \cdot catct \cdot asa. \end{aligned}$$

The significance levels for the coefficients of the four patterns in this model can be formally computed and are, respectively 3.3340e-21, 1.7253e-05, 1.5721e-04, and 0.0428.

Smokers, Vitamins and Cataracts

Recalling "smoking = 1", "not taking vitamins = 1" "having cataract(s) = 1", we can read from this model:

$$\begin{aligned} f &= -2.84 + 2.42 \cdot catct + 1.11 \cdot pky \cdot vtm \\ &\quad +1.98 \cdot sex \cdot inc \cdot jomyop \cdot asa + 1.15 \cdot sex \cdot inc \cdot catct \cdot asa. \end{aligned}$$

Letting variable indicate "0": we see that smokers with cataract or without cataract are protected by taking vitamins, $B_{pky \cdot vtm} = 0$. For non-smokers $B_{pky \cdot vtm} = 0$, taking or not taking vitamins makes no (significant) difference.

Smokers, Vitamins and Cataracts

This result is physiologically meaningful-recent literature suggests:

a) Certain vitamins are good for eye health.

b) Smoking depletes the serum and tissue vitamin level, especially Vitamins C and E.

"However, our data are observational and subject to uncontrolled confounding. A randomized controlled clinical trial would provide the best evidence of any effect of vitamins on myopic change in smokers. " (R. Klein in Shi *et al (2008))*

Example 2. Rheumatoid Arthritis and SNPS in a Generative Model From GAW 15

The 15th Genetic Analysis Workshop (GAW 15) provided an extensive simulation data set of cases and controls with simulated single nucleotide polymorphisms (snp's) related to the risk of rheumatoid arthritis. This was an opportuniity to apply LPS to a set of large genetic attribute vectors with a known (hopefully realistic!) architecture, and compare the results with the description of the architecture generating the data. There were 1500 cases and 2000 controls, and 100 replicates. For the analysis we used the 674 snps in chromosome 6 along with three environmental variables, age, sex and smoking. Older than 55, female and smoking are the risky ($y = 1$) attributes. Most of the snps have three levels, normal, one variant allele and two variant alleles so two dummy variables were created to code this.

Example 2. Rheumatoid Arthritis and SNPS in a Generative Model From GAW 15

A screen step one variable at a time with a (generous) passing criteria of at least one p value less than 0.05 resulted in 72 snps, sex and smoking. Using all main effects and second order patterns from these 72 snps, sex and smoking resulted in 10371 basis functions. After some correction for an obvious miscoding in the GAW data, a model with five main effects and five two factor interactions was fit. (Next slide, above the double line). The LPS found the important variables.

Simulated and Fitted Models, With and Without Third Order Patterns

	Variable 1	Level 1	Variable 2	Level 2	Coef	Est
Main effects	constant	-	-	-	-4.8546	-4.6002
	smoking	-	-	-	0.8603	0.9901
	SNP6_153	1	-	-	1.8911	1.5604
	SNP6_162	1	-	-	2.2013	1.9965
	SNP6_154	2	-	-	0.7700	1.0808
Second order patterns	*sex*	-	*SNP6_153*	1	0.7848	0.9984
	sex	-	*SNP6_154*	2	0.9330	0.9464
	SNP6_153	2	*SNP6_154*	2	4.5877	4.2465
	SNP6_153	1	*SNP6_553*	2	0.4021	0
	SNP6_154	2	*SNP6_490*	1	0.3888	0
Added						
Third order pattern	*sex* · *SNP6_108_2* · *SNP6_334_2*				3	2.9106

Simulated model adapted from GAW 15 analysis. "Coef" is simulated coefficients. LPS run with main effects and two factor terms. "Est" is the estimated coefficients. LPS run with third order patterns (403,594 patterns) resulted in the same fit. Then a third order pattern added to the model, LPS successfully fitted it.

Correlated K-variate Bernoulli Outcomes

Generalizes Gao *et al*, JASA 2001

Given $(y(i), x(i))$, where now the response is $y(i) = (y_1(i), \cdots, y_K(i))$. As before, $x(i) = (x_1(i), \cdots, x_p(i))$. Find $f = f_1(x), \cdots, f_K(x), f_{12}(x), \cdots, f_{(K-1)K}(x), \cdots, f_{1,2\cdots K}(x)$ to minimize

$$I_\lambda(y, f) = \mathcal{L}(y, f) + \lambda \cdot J(f).$$

The negative log likelihood $\mathcal{L}$ for bivariate $(K = 2)$ y is given by

$$\mathcal{L}(y, f) = -[f_1(x)y_1 + f_2(x)y_2 + f_{12}(x)y_1y_2 - b(f(x))],$$

where

$$b(f(x)) = \log(1+\exp\{f_1(x)\}+\exp\{f_2(x)\}+\exp\{f_1(x)+f_2(x)+f_{12}(x)\})$$

and $\lambda \cdot J(f)$ is shorthand for $\lambda_1 J(f_1) + \lambda_2 J(f_2) + \lambda_{12} J(f_{12})$.

Each component of f is represented as a sum of basis functions (patterns) as in the univariate response case and an l_1 penalty applied to the coefficients. In particular, it may be of interest to examine patterns that enter into f_{12}, which are a proxy for correlation. $f_{12} = \log \frac{p_{11}p_{00}}{p_{01}p_{10}}$ where $p_{rs} = Pr(y_1 = r, y_2 = s)$. Thus $f_{12} = 0$ if and only if $p_{rs} = p_r p_s$ and so is 0 if and only if y_1 and y_2 are independent. Note that it can be shown that $cov(y_1, y_2) = Ey_1y_2 - Ey_1Ey_2 = p_{11}p_{00} - p_{01}p_{10}$. This is work in progress with Bin Dai, Shilin Ding, Steve Wright, Kristine Lee and Ron and Barbara Klein. In particular interaction of genetic markers and environmental covariates and how they relate to multiple correlated outcomes (diseases/phenotypes) is of interest.

Summary and Conclusions

Results can be generalized in several ways. The message is:

1. The log linear expansion of the multivariate Bernoulli distribution is good way to think about Bernoulli responses with attribute vectors consisting of bits.
2. Bernoulli data can tuned in a manner appropriate for Bernoulli data. BGACV is good for the variable selection problem with Bernoulli data.
3. An algorithm for fitting a very large number of unknowns simultaneously in a LASSO model is available.
4. It is argued that global methods based on simultaneous fits are better at fitting correlated predictors in the variable selection problem than greedy methods.
5. Firm theoretical foundations for items 2. and 4. would be nice to have.