Assignment #3 STA410H1F/2102H1F

due Wednesday November 15, 2017

Instructions: Solutions to problems 1 and 2 are to be submitted on Blackboard (PDF files strongly preferred). You are strongly encouraged to do problems 3–6 but these are not to be submitted for grading.

1. Suppose that S is an $n \times n$ matrix where n may be very large and the elements of S may not be explicitly defined. We are interested in approximating the trace of S, that is, the sum of its diagonal elements. For example, if S is a smoothing matrix in regression ($\widehat{\boldsymbol{y}} = S\boldsymbol{y}$) then the trace of S gives a measure of the effective number of parameters using in the smoothing method. (In multiple regression models, the smoothing matrix is the projection matrix $X(X^TX)^{-1}X^T$ whose trace is the number of columns of X.)

(a) Show that if A and B are $m \times n$ and $n \times m$ matrices, respectively, then $\text{tr}(AB) = \text{tr}(BA)$. (This is a well-known fact but humour me with a proof!)

(b) Suppose that $\boldsymbol{V}$ is a random vector of length n such that $E[\boldsymbol{V}\boldsymbol{V}^T] = I$. If S is an $n \times n$ non-random matrix, show that

$$E\left[\boldsymbol{V}^T S \boldsymbol{V}\right] = E\left[\text{tr}\left(S\boldsymbol{V}\boldsymbol{V}^T\right)\right] = \text{tr}\left[SE\left(\boldsymbol{V}\boldsymbol{V}^T\right)\right] = \text{tr}(S)$$

and so $\text{tr}(S)$ can be estimated by

$$\widehat{\text{tr}(S)} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{V}_i^T S \boldsymbol{V}_i$$

where $\boldsymbol{V}_1, \cdots, \boldsymbol{V}_m$ are independent random vectors with $E[\boldsymbol{V}_i \boldsymbol{V}_i^T] = I$.

(c) Suppose that the elements of each $\boldsymbol{V}_i$ are independent, identically distribution random variables with mean 0 and variance 1. Show that $\text{Var}(\widehat{\text{tr}(S)})$ is minimized by taking the elements of $\boldsymbol{V}_i$ to be ± 1 each with probability 1/2.
Hint: This is easier than it looks – $\text{Var}(\boldsymbol{V}^T S \boldsymbol{V}) = E[(\boldsymbol{V}^T S \boldsymbol{V})^2] - \text{tr}(S)^2$ so it suffices to minimize

$$E[(\boldsymbol{V}^T S \boldsymbol{V})^2] = \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{\ell=1}^{n} s_{ij}s_{k\ell}E(V_iV_jV_kV_\ell).$$

Given our conditions on the elements of $\boldsymbol{V}_i$, $V_1, \cdots, V_n$, most of $E(V_iV_jV_kV_\ell)$ are either 0 or 1. You should be able to show that

$$E[(\boldsymbol{V}^T S \boldsymbol{V})^2] = \sum_{i=1}^{n} s_{ii}^2 E(V_i^4) + \text{constant}$$

and find V_i to minimize $E(V_i^4)$ subject to $E(V_i^2) = 1$.

(d) Suppose we estimate the function g in the non-parametric regression model

$$y_i = g(x_i) + \varepsilon_i \quad \text{for } i = 1, \cdots, n$$

using loess (i.e. the R function `loess`) where the smoothness is determined by the parameter `span` lying between 0 and 1. Given a set of predictors $\{x_i\}$ and a value of `span`, write an R function to approximate the effective number of parameters.

2. Suppose that $X_1, \cdots, X_n$ are independent Gamma random variables with common density

$$f(x; \alpha, \lambda) = \frac{\lambda^\alpha x^{\alpha-1} \exp(-\lambda x)}{\Gamma(\alpha)} \quad \text{for } x > 0$$

where $\alpha > 0$ and $\lambda > 0$ are unknown parameters.

(a) The mean and variance of the Gamma distribution are α/λ and α/λ^2, respectively. Use these to define method of moments estimates of α and λ based on the sample mean and variance of the data $x_1, \cdots, x_n$

(b) Derive the likelihood equations for the MLEs of α and λ and derive a Newton-Raphson algorithm for computing the MLEs based on $x_1, \cdots, x_n$. Implement this algorithm in R and test on data generated from a Gamma distribution (using the R function `rgamma`). Your function should also output an estimate of the variance-covariance matrix of the MLEs – this can be obtained from the Hessian of the log-likelihood function.

Important note: To implement the Newton-Raphson algorithm, you will need to compute the first and second derivatives of $\ln \Gamma(\alpha)$. These two derivatives are called (respectively) the digamma and trigamma functions, and these functions are available in R as `digamma` and `trigamma`; for example,

```
> gamma(2) # gamma function evaluated at 2
[1] 1
> digamma(2) # digamma function evaluated at 2
[1] 0.4227843
> trigamma(2) # trigamma function evaluated at 2
[1] 0.6449341
```

Supplemental problems:

3. Consider LASSO estimation in linear regression where we define $\widehat{\boldsymbol{\beta}}_\lambda$ to minimize

$$\sum_{i=1}^{n}(y_i - \bar{y} - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 + \lambda\sum_{j=1}^{p}|\beta_j|$$

for some $\lambda > 0$. (We assume that the predictors are centred and scaled to have mean 0 and variance 1, in which case $\bar{y}$ is the estimate of the intercept.) Suppose that the least squares estimate (i.e. for $\lambda = 0$) is non-unique — this may occur, for example, if there is some exact linear dependence in the predictors or if $p > n$. Define

$$\tau = \min_{\beta}\sum_{i=1}^{n}(y_i - \bar{y} - \boldsymbol{x}_i^T\boldsymbol{\beta})^2$$

and the set

$$\mathcal{C} = \left\{\boldsymbol{\beta} : \sum_{i=1}^{n}(y_i - \bar{y} - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 = \tau\right\}.$$

We want to look at what happens to the LASSO estimate $\widehat{\boldsymbol{\beta}}_\lambda$ as $\lambda \downarrow 0$.

(a) Show that $\widehat{\boldsymbol{\beta}}_\lambda$ minimizes

$$\frac{1}{\lambda}\left\{\sum_{i=1}^{n}(y_i - \bar{y} - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 - \tau\right\} + \sum_{j=1}^{p}|\beta_j|.$$

(b) Find the limit of

$$\frac{1}{\lambda}\left\{\sum_{i=1}^{n}(y_i - \bar{y} - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 - \tau\right\}$$

as $\lambda \downarrow 0$ as a function of $\boldsymbol{\beta}$. (What happens when $\boldsymbol{\beta} \notin \mathcal{C}$?) Use this to deduce that as $\lambda \downarrow 0$, $\widehat{\boldsymbol{\beta}}_\lambda \to \widehat{\boldsymbol{\beta}}_0$ where $\widehat{\boldsymbol{\beta}}_0$ minimizes $\sum_{j=1}^{p}|\beta_j|$ on the set $\mathcal{C}$.

(c) Show that $\widehat{\boldsymbol{\beta}}_0$ is the solution of a linear programming problem. (Hint: Note that $\mathcal{C}$ can be expressed in terms of $\boldsymbol{\beta}$ satisfying p linear equations.)

4. Consider minimizing the function

$$g(x) = x^2 - 2\alpha x + \lambda|x|^\gamma$$

where $\lambda > 0$ and $0 < \gamma < 1$. (This problem arises, in a somewhat more complicated form, in shrinkage estimation in regression.) The function $|x|^\gamma$ has a "cusp" at 0, which mean that if λ is sufficient large then g is minimized at $x = 0$.

(a) g is minimized at $x = 0$ if, and only if,

$$\lambda \geq \frac{2}{2-\gamma}\left[\frac{2-2\gamma}{2-\gamma}\right]^{1-\gamma}|\alpha|^{2-\gamma}. \tag{1}$$

Otherwise, g is minimized at x^* satisfying $g'(x^*) = 0$. Using R, compare the following two iterative algorithms for computing x^* (when condition (1) does not hold):

(i) Set $x_0 = \alpha$ and define

$$x_k = \alpha - \frac{\lambda\gamma}{2}\frac{|x_{k-1}|^\gamma}{x_{k-1}} \quad k = 1, 2, 3, \cdots$$

(ii) The Newton-Raphson algorithm with $x_0 = \alpha$.

Use different values of α, γ, and λ to test these algorithms. Which algorithm is faster?

(b) Functions like g arise in so-called bridge estimation in linear regression (which are generalizations of the LASSO) – such estimation combines the features of ridge regression (which shrinks least squares estimates towards 0) and model selection methods (which produce exact 0 estimates for some or all parameters). Bridge estimates $\widehat{\boldsymbol{\beta}}$ minimize (for some $\gamma > 0$ and $\lambda > 0$),

$$\sum_{i=1}^{n}(y_i - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 + \lambda\sum_{j=1}^{p}|\beta_j|^\gamma. \tag{2}$$

See the paper by Huang, Horowitz and Ma (2008) ("Asymptotic properties of bridge estimators in sparse high-dimensional regression models" *Annals of Statistics.* **36,** 587–613) for details. Describe how the algorithms in part (a) could be used to define a coordinate descent algorithm to find $\widehat{\boldsymbol{\beta}}$ minimizing (2) iteratively one parameter at a time.

(c) Prove that g is minimized at 0 if, and only if, condition (1) in part (a) holds.

5. Suppose that A is a symmetric non-negative definite matrix with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$. Consider the following algorithm for computing the maximum eigenvalue λ_1:

$$\text{Given } \boldsymbol{x}_0, \text{ define for } k = 0, 1, 2, \cdots, \boldsymbol{x}_{k+1} = \frac{A\boldsymbol{x}_k}{\|A\boldsymbol{x}_k\|_2} \text{ and } \mu_{k+1} = \frac{\boldsymbol{x}_{k+1}^T A\boldsymbol{x}_{k+1}}{\boldsymbol{x}_{k+1}^T\boldsymbol{x}_{k+1}}.$$

Under certain conditions, $\mu_k \to \lambda_1$, the maximum eigenvalue of A; this algorithm is known as the **power method** and is particularly useful when A is sparse.

(a) Suppose that $\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n$ are the eigenvectors of A corresponding to the eigenvalues $\lambda_1, \cdots, \lambda_n$. Show that $\mu_k \to \lambda_1$ if $\boldsymbol{x}_0^T\boldsymbol{v}_1 \neq 0$ and $\lambda_1 > \lambda_2$.

(b) What happens to the algorithm if if the maximum eigenvalue is not unique, that is, $\lambda_1 = \lambda_2 = \cdots = \lambda_k$?

6. Consider the estimation procedure in problem 2 of Assignment #2 (where we used the Gauss-Seidel algorithm to estimate $\{\theta_i\}$). Use both gradient descent and accelerated gradient descent to estimate $\{\theta_i\}$. To find an appropriate value of ϵ, it is useful to approximate the maximum eigenvalue of the Hessian matrix of the objective function – the algorithm in problem 5 is useful in this regard.