Homework 5

Jimmy Hickey

Due @ 11:59pm on April 18, 2020

Part 1. In this homework we will study two algorithms for iteratively computing the generalized lasso solution.

Let $\mathbf{y} \in \mathbb{R}^n$ and $\mathbf{X} \in \mathbb{R}^{n \times p}$ denote a response and design matrix. The generalized lasso is the solution to the following optimization problem.

$$\text{minimize } \frac{1}{2}\|\mathbf{y} - \mathbf{X}\mathbf{b}\|_2^2 + \lambda\|\mathbf{D}\mathbf{b}\|_1,$$

where $\lambda \geq 0$ is a tuning parameter that trades off sparsity in the linear transformation $\mathbf{D}\mathbf{b}$ of $\mathbf{b}$ and the discrepancy between the linear model $\mathbf{X}\mathbf{b}$ and the response $\mathbf{y}$. In this assignment we consider the simpler problem where $\mathbf{X} = \mathbf{I}$.

$$\text{minimize } \frac{1}{2}\|\mathbf{y} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{D}\mathbf{b}\|_1,$$

This is the case, for example, for the fused lasso or trend filtering. The optimization problem is challanging to solve due to the term $\|\mathbf{D}\mathbf{b}\|_1$.

Thus, consider the following equivalent equality constrained problem.

$$\begin{aligned} &\text{minimize } \frac{1}{2}\|\mathbf{y} - \mathbf{b}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1 \\ &\text{subject to } \mathbf{D}\mathbf{b} = \boldsymbol{\theta}. \end{aligned}$$

1. Show that the dual problem is given by

$$\begin{aligned} &\text{maximize } \frac{1}{2}\|\mathbf{y}\|_2^2 - \frac{1}{2}\|\mathbf{y} - \mathbf{D}^\mathsf{T}\mathbf{v}\|_2^2 \\ &\text{subject to } \|\mathbf{v}\|_\infty \leq \lambda. \end{aligned}$$

2. What are the KKT conditions for this primal-dual pair of optimization problems?

3. Convert your KKT conditions into a single scalar equation involving a KKT residual. You will use this and the duality gap to evaluate the correctness of your algorithms in part 2.

4. How do you map a dual variable $\mathbf{v}$ to a primal variable $\mathbf{b}$?

Note that we may equivalently solve the following box constrained least squares problem.

$$\begin{aligned} &\text{minimize } \frac{1}{2}\|\mathbf{y} - \mathbf{D}^\mathsf{T}\mathbf{v}\|_2^2 \\ &\text{subject to } \|\mathbf{v}\|_\infty \leq \lambda. \end{aligned}$$

5. Prove that the projection of $x \in \mathbb{R}$ onto the interval $[-\lambda, \lambda]$ is given by

$$P_{[-\lambda,\lambda]}(x) = \begin{cases} \lambda & x > \lambda \\ -\lambda & x < -\lambda \\ x & |x| \leq \lambda \end{cases}$$

6. Derive a coordinate descent algorithm for solving the dual problem and write out pseudocode for your algorithm.

```
repeat
  Do something
  Do something else
  etc.
until convergence
```

7. Prove that the dual objective is Lipschitz differentiable with constant $L = \|\mathbf{D}\|_{\text{op}}^2$.

8. Derive a proximal gradient algorithm for solving the dual problem and write out pseudocode for your algorithm.

```
repeat
  Do something
  Do something else
  etc.
until convergence
```

Part 2. You will implement a coordinate descent algorithm and proximal gradient algorithm for solving the trend filtering problem.

Step 1: Write a function to compute the kth order differencing matrix $\mathbf{D}_n^{(k)}$.

```r
#' Compute kth order differencing matrix
#'
#' @param k order of the differencing matrix
#' @param n Number of time points
#' @export
myGetDkn <- function(k, n) {

}
```

Step 2: Write a function to compute the KKT residual you derived in part 1.

```r
#' Compute KKT residual
#'
#' @param y response
#' @param b primal variable
#' @param theta primal variable
#' @param v dual variable
#' @param D differencing matrix
#' @param lambda regularization parameter
#' @export
kkt_residual <- function(y, b, theta, v, D, lambda) {

}
```

Step 3: Write a function to compute the duality gap.

```
#' Compute duality gap
#'
#' @param y response
#' @param b primal variable
#' @param v dual variable
#' @param D differencing matrix
#' @param lambda regularization parameter
#' @export
duality_gap <- function(y, b, v, D, lambda) {

}
```

Step 4: Write a coordinate descent algorithm for solving the dual problem. Use the relative change in function values as a stopping criterion.

```
#' Solve trend-filtering by coordinate descent on dual problem
#'
#' @param y response
#' @param k order of differencing matrix
#' @param v Initial dual variable
#' @param lambda regularization parameter
#' @param max_iter maximum number of iterations
#' @param tol convergence tolerance
trend_filter_cd <- function(y, k, v, lambda=0, max_iter=1e2, tol=1e-3) {

}
```

Your function should return

- The final iterate value
- The objective function values
- The relative change in the function values
- The relative change in the iterate values
- The KKT residual after every iteration
- The duality gap after every iteration

Step 5: Write a proximal gradient algorithm for solving the dual problem. Use the relative change in function values as a stopping criterion. You may either use a fixed step size or write your own backtracking line search functions.

```r
#' Solve trend-filtering by proximal gradient on dual problem
#'
#' @param y response
#' @param k order of differencing matrix
#' @param v Initial dual variable
#' @param lambda regularization parameter
#' @param max_iter maximum number of iterations
#' @param tol convergence tolerance
trend_filter_pg <- function(y, k, v, lambda=0, max_iter=1e2, tol=1e-3) {

}
```

Your function should return

- The final iterate value
- The objective function values
- The relative change in the function values
- The relative change in the iterate values
- The KKT residual after every iteration
- The duality gap after every iteration

Step 6: Use your two trend filtering function to smooth some interesting time series data. For example, you might use the tseries R package on CRAN (see the function **get.hist.quote**) to download historical financial data for the daily closing prices of Apple stock over the past two years. You may use the same data used in Homework 4. Try several λ values - different enough to generate noticably different smoothed estimates - and at least two differencing matrix orders, e.g. $\mathbf{D}_n^{(2)}$ and $\mathbf{D}_n^{(3)}$.

For both algorithms (and all λ and all differencing matrices) plot the following

- The noisy data and smoothed estimates.

For both algorithms (and one λ and one differencing matrix) plot the following against the iteration

- The relative change in the function values
- The relative change in the iterate values
- The KKT residual after every iteration
- The duality gap after every iteration