EE 381V: Large Scale Optimization **Fall 2012**

Lecture 24 — April 11

Lecturer: Caramanis & Sanghavi *Scribe: Tao Huang*

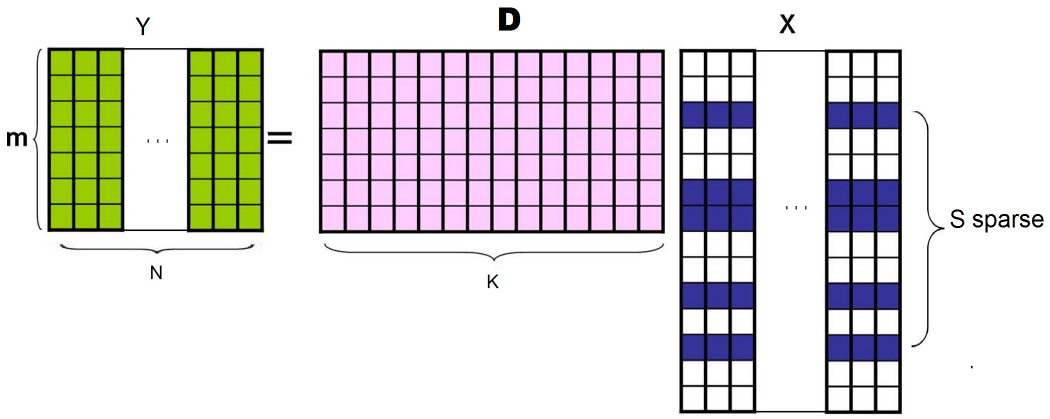
24.1 Review

In past classes, we studied the problem of sparsity. Sparsity problem is that we are given a set of basis D (dictionary, design matrix) and a signal y, and we want to find a data x of high dimensions, but of few elements being nonzero. If the measurements we have are linear measurements, then the problem can be expressed as $y = Dx$, where x is the data we want to recover from the linear measurements y and matrix D. If written down as a formulation, it is represented as $\min_x ||x||_0$ s.t. $y = Dx$, or approximately $y \approx Dx$, satisfying $\|y - Dx\|_p \leq \varepsilon$. Or if we limited the sparsity to be s, then it is represented as $\min_x ||y - Dx||_2^2$ s.t. $||x||_0 \leq s$.

24.2 Dictionary Learning

Now we look at the reverse problem: could we design dictionaries based on learning? Our goal is to find the dictionary D that yields sparse representations for the training signals. We believe that such dictionaries have the potential to outperform commonly used predetermined dictionaries. With evergrowing computational capabilities, computational cost may become secondary in importance to the improved performance achievable by methods that adapt dictionaries for special classes of signals.

Given $y_1, y_2, \cdots, y_N$, where N is large, find a dictionary matrix D with columns number $K \ll N$, such that every y_i is a sparse combination of columns of D. From now on, denote dictionary matrix as D, with columns $\{d_i\}_{1=1}^K$; collection of signals as a $m \times N$ matrix Y, whose columns are points y_is; representation matrix X, a $n \times N$ matrix with columns x_is being representation of y_is in dictionary D.



Intuitively, under natural idea, we would have the formulation represented as:

$$\min_{D,\{x_i\}} \sum_i \|y_i - Dx_i\|^2 \qquad \text{s.t. } \|x_i\|_0 \leq s \tag{24.1}$$

But representation (24.1) has several issues to be considered:

- Issue 1: there is a combinatorial sparsity constraint;
- Issue 2: Optimization is over both D and $\{x_i\}$, so this is a non-convex problem due to bi-linearity in objective function.

24.2.1 Dictionary Learning, Idea 1

Do Lagrangian relaxation to the formulation (24.1) and relax the ℓ_0 norm to ℓ_1 norm, we have

$$\min_{D,\{x_i\}} \|Y - DX\|_F^2 + \lambda \|x\|_1 \tag{24.2}$$

Here $\|Y - DX\|_F^2 = \sum_i \|y_i - Dx_i\|^2$. Formulation (24.2) does penalize ℓ_1 norm of x, but does not penalize the entries of D as it does for x. Thus, the solution will tend to increase the dictionary entries values, in order to allow the coefficients to become closer to zero. This difficulty has been handled by constraining the ℓ_2 norm of each basis element, so that the output variance of the coefficients is kept at an appropriate level.

An iterative method was suggested for solving (24.2). It includes two main steps in each iteration:

- Sparse coding: calculate the coefficients x_is;
- Dictionary update: fix $\{x_i\}$, update D.

The iterative method would lead to a local-minimum, so the performance of the solution is sensitive to the initial D and $\{x_i\}$ we choose and the method we use to upgrade dictionary and coefficients.

In *sparse coding* part, there is a bunch of methods that could be used: OMP, ℓ_1 minimization, iterative thresholding, etc, which, more or less, were already been covered in former classes.

In dictionary update part, fix $\{x_i\}$, we are left with solving

$$\min_D \|Y - DX\|_F^2 \tag{24.3}$$

Option 1: directly solve the least square problem.

Issue: the "guess" X might be lousy. By doing the "exact" solving step, the whole process would be "pushed" to local minimum nearest the initial guess. The situation is similar to "over-fitting" current observation in regression model, where noise is unsuitably considered in optimization and "push" regression model to a lousy direction.

Option 2: simple gradient descent procedure with small step

$$D^{(n+1)} = D^{(n)} - \eta \sum_{i=1}^{N} (D^{(n)} x_i - y_i) x_i'$$

Initializing D: greedy algorithm

1 Pick largest (ℓ_2 norm) column of Y, move to D from Y;

2 To all columns that remain in Y, subtract their orthogonal projection to range(D)

repeat (1) and (2) till a full basis D is found.

24.2.2 Dictionary Learning, Idea 2

Unions of orthonormal dictionaries

Consider a dictionary composed as a union of orthonormal bases.

$$D = [D_1, \cdots, D_L]$$

where $D_i \in \mathcal{O}(m)$ (m is the dimension of y_is), that is, $D_i' D_i = D_i D_i' = I_{m \times m}$, for $i = 1, \cdots, L$.

Correspondingly, divide X to L submatrices, with each submatix X_i containing the coefficients of of D_i

$$X = [X_1', X_2', \cdots, X_L']'$$

The update algorithm need to preserve the orthonormality of D_is in each iteration. Assuming known coefficients, the proposed algorithm updates each orthonormal basis D_i sequentially. The update of D_i is done by first computing the residual matrix

$$E_i = Y - \sum_{j \neq i} D_j X_j$$

Then we solve $\min_{D_i \in \mathcal{O}(m)} \|E_i - D_i X_i\|_F^2$ for updated D_i

24.3 K-SVD algorithm

Overall, this algorithm is a "generalization" of K-means algorithm for clustering points $\{y_i\}_{i=1}^N$.

K-means algorithm

In K-means, first we are offered K centers, denoted as $C = [c_1, c_2, \cdots, c_K]$. This could also be seen as a basis or dictionary. The index j of point y_i is selected by finding j such that

$$\|y_i - Ce_j\|_2^2 \leq \|y_i - Ce_k\|, \forall k \neq j$$

This has analogy to sparse coding stage. We force the sparsity of coefficients to be 1 and the nonzero component should be 1. Written as formulation and in matrix form

$$\begin{aligned} \min_x \ & \|Y - CX\|_F^2 \\ s.t. \ & x_i = e_k \ \text{ for some } k, \forall i \\ & (\|x_i\|_0 = 1) \end{aligned}$$

And in updating $\{c_i\}$ stage, let R_k be the set of index of points clustered into kth cluster

$$c_k = \frac{1}{|R_k|} \sum_{i \in R_k} y_i = \text{argmin}_c \sum_{i \in R_k} \|y_i - c\|_2^2$$

combine all K such optimization problem, we have update of C is

$$C = \arg\min_C \|Y - CX\|_F^2$$

remember here X comes from sparse coding stage, defining which cluster each point in Y lies. Denote kth row of X as x_T^k, then

$$Y - CX = Y - \sum_k c_k x_T^k = \underbrace{Y - \sum_{j \neq k} c_j x_T^j}_{E_k} - c_k x_T^k$$

Since the whole problem is decomposable to subproblems of finding each $c_k, \forall k$, so it is equivalent to sequentially search c_k that minimizes $\|E_k - c_k x_T^k\|_F^2 = \sum_{i \in R_k} \|y_i - c_k\|_2^2$.

Back to dictionary learning problem,

$$Y - DX = \underbrace{(Y - \sum_{j \neq k} d_j x_T^j)}_{E_k} - d_k x_T^k$$

So

$$\|Y - DX\|_F^2 = \|E_k - d_k x_T^k\|_F^2$$

In dictionary update step, analogous to K-means, assuming known X, we could sequentially solve

$$\min_{d_k} \|E_k - d_k x_T^k\|_F^2$$

for update of columns of D.

K-SVD algorithm:
Data: $\{y_i\}_{i=1}^N$
Initialization: $D^{(0)} \in \mathbb{R}^{m \times K}$, with each column normalized; $t = 1$
Repeat until convergence:

1 Do sparse coding, solve for each $\{x_i\}_{i=1}^N$ using suitable algorithm;

2 **for** each column $k = 1, \cdots, K$ in $D^{(t-1)}$

1 Define $R_k = \{i | 1 \leq i \leq N, x_T^k(i) \neq 0\}$

2 Compute $E_k = Y - \sum_{j \neq k} d_j x_T^j$, extract columns with indices in R_k, denote as $E_k^{R_k}$; $E_k^{R_k} \in \mathbb{R}^{m \times |R_k|}$

3 Apply SVD, $E_k^{R_k} = U \Delta V^T$, choose updated dictionary column d_k as the first column of U, update the nonzero items of coefficient x_T^k by $x_{R_k}^k$, $x_{R_k}^k$ is the first column of V multiplied by $\sigma_1(E_k^{R_k})$

3 Set $t = t + 1$