MATH 565: Lecture 17 (03/10/2026)

Today:
* subgradients with coordinate descent
* proximal gradient method
* constrained optimization

Subgradients with Coordinate Descent (CD)

We saw CD in lecture 9. We now combine CD with subgradients

We consider regression with L_1-regularization.

$$J(\bar{w}) = \frac{1}{2}\|D\bar{w} - \bar{y}\|^2 + \lambda \sum_{j=1}^{d} |w_j|$$

Recall, the residuals $\bar{r} = \bar{y} - D\bar{w} \Rightarrow \bar{r} = \underbrace{\left(\bar{y} - \sum_{k \neq j} \bar{d}_k w_k\right)}_{\bar{\rho}_j} - \bar{d}_j w_j$.

For coordinate j, the loss function is

$$J(w_j) = \frac{1}{2}\|\bar{\rho}_j - \bar{d}_j w_j\|^2 + \lambda |w_j| + \underbrace{\text{constants}}_{\text{not dependent on } w_j}$$

$\Rightarrow$ First order optimality condition is:

$$0 \in \frac{\partial J(w_j)}{\partial w_j} = -\bar{d}_j^T(\bar{\rho}_j - \bar{d}_j w_j) + \lambda s_j$$

(s_j: subgradient; 0 is in the subgradient set)

If data is normalized, then $\bar{d}_j^T \bar{d}_j = 1 \ \forall j$.

This gives the following update step:

$$w_j = \bar{d}_j^T \bar{\rho}_j - \lambda s_j.$$

If working directly with $\bar{r}$, we get the update as

$$w_j \leftarrow w_j + \bar{d}_j^T \bar{r} - \lambda s_j.$$

Finally, if the data is not normalized, we get the following update:

$$\boxed{w_j \leftarrow w_j + \frac{\bar{d}_j^T \bar{r} - \lambda s_j}{\|\bar{d}_j\|^2}}.$$

More Guidance on choosing s_j

Soft thresholding (Idea: set $w_j \leftarrow 0$ when it is close to 0)

$$w_j = \begin{cases} 0, & \text{if } \frac{-\lambda}{\|\bar{d}_j\|^2} \leq w_j + \frac{\bar{d}_j^T \bar{r}}{\|\bar{d}_j\|^2} \leq \frac{\lambda}{\|\bar{d}_j\|^2} \\ w_j + \frac{\bar{d}_j^T \bar{r} - \lambda \, \text{sign}(w_j)}{\|\bar{d}_j\|^2}, & \text{otherwise} \end{cases}$$

Just as in CD, we cycle through all dimensions till convergence.

E. Proximal Gradient Method (PGM)

PGM is particularly useful for loss functions that split as

$$J(\bar{w}) = G(\bar{w}) + H(\bar{w})$$

where $G(\bar{w})$ is differentiable, but $H(\bar{w})$ is not.

Idea of PGM: take a GD step on $G(\cdot)$ (with stepsize α) and then a "proximal" step on $H(\cdot)$.

↳ "find minimum of $H(\cdot)$ in proximity of current $\bar{w}$"

We define the proximal operator for H as

$$P_{H,\alpha}(\bar{w}) = \underset{\bar{u}}{\text{argmin}} \left\{ \alpha H(\bar{u}) + \frac{1}{2} \|\bar{u} - \bar{w}\|^2 \right\} \quad \text{——— (*)}$$

For Regression with L_1-regularization, $P_{H,\alpha}$ function is separable, and hence can be solved individually for each j.

The proximal operator for dimension j is given as

$$\left[P_{H,\alpha}(\bar{w})\right]_j = \operatorname*{argmin}_u \left\{\alpha\lambda|u| + \frac{1}{2}(u-w_j)^2\right\}$$

To find $\left[P_{H,\alpha}(\bar{w})\right]_j$, we consider its first order optimality condition, which gives

$$\alpha\lambda s(u) + u - w_j = 0, \quad \text{where } s(u) = \begin{cases} +1, & u>0 \\ -1, & u<0 \\ \in[-1,1], & u=0 \end{cases}.$$

subgradient

We consider the three cases to specify u

- $u>0$: we get $u = w_j - \alpha\lambda$
  which requires $w_j > \alpha\lambda$
- $u<0$: we get $u = w_j + \alpha\lambda$,
  which requires $w_j < -\alpha\lambda$.
- if $-\alpha\lambda \le w_j \le \alpha\lambda$, then
  we set $u=0$.

Thus, we specify the proximal update as follows:

$$\left[P_{H,\alpha}(\bar{w})\right]_j = \begin{cases} w_j + \alpha\lambda, & w_j < -\alpha\lambda \\ 0, & -\alpha\lambda \le w_j \le \alpha\lambda \\ w_j - \alpha\lambda, & w_j > \alpha\lambda \end{cases}$$

This is called the soft threshold operator, and when applied iteratively, is called the iterative soft thresholding algorithm (ISTA).

To summarize this topic, we discussed the following approaches to handle the challenges faced by basic Newton method.

A. Trust Region Method (TRM)
B. Conjugate Gradient Method (CGM)
C. Quasi Newton method and BFGS.
D. Subgradiend method
E. Proximal Gradient Method (PGM)

Gradient descent and variants the Newton method, and the variants listed above (mostly) considered minizing loss functions without any constraints (or limits) on $\bar{w}$. We now study optimization of loss functions with additional constraints.

Constrained Optimization and Duality

We consider optimizing a loss function with constraints of the form

$$\min J(\bar{w})$$
$$\text{s.t. } \bar{w} \in \Omega$$

where Ω is the feasible region defined using a set of constraints.

If we apply GD directly to $J(\bar{w})$, we may get $\bar{w} \notin \Omega$.
There are two broad approaches to address this challenge.

1. Primal approach: modify GD to ensure $\bar{w} \in \Omega$ at each step.

2. Dual approach: Lagrangian relaxation:
   create a new loss (or objective) function L in which the (primal) constraints are captured as dual variables.

   While the structure of L may be simpler, we may still have to handle restricted feasible regions for which we may use primal-based approaches.

Luckily for us, ML problems often have "simpler" constraint sets (Ω). For instance, we often have

* linear or convex constraint sets: Ω is specified by
  $G(\bar{w}) \leq \bar{b}$ where $G(\bar{w})$ is linear or convex

* norm constraints, e.g., $\|\bar{w}\|^2 = 1$.
  ← in clustering, PCA, etc.

Such constraint sets Ω are often easier to work with than general (nonconvex and nonlinear sets). We present details of the primal approach first.

Primal Methods

These are also known as feasible direction methods

The idea is to take a GD step. If (updated) $\bar{w} \in \Omega$, proceed. But if the updated $\bar{w} \notin \Omega$, then project it to a closest point in Ω.

We present the details for

$$\begin{aligned} &\min \ F(\bar{w}) \\ &\text{s.t.} \ \ \bar{w} \in C \end{aligned}$$

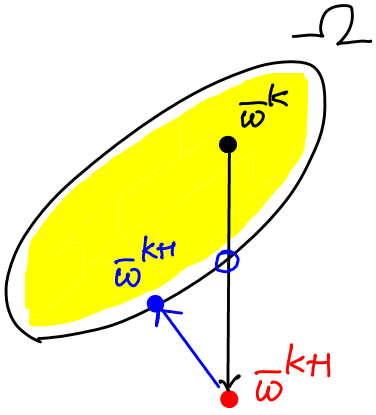
where $F(\cdot)$ is a convex function and C is a convex set.

Typical instances of C:

* $A\bar{w} \leq \bar{b}$ (system of linear inequalities)
* intersection of convex constraints:
  $g_i(\bar{w}) \leq \bar{0}$, $i = 1, \ldots, m$
  where each $g_i(\cdot)$ is convex.

1. $\bar{w}^{k+1} = \bar{w}^k - \alpha \nabla F(\bar{w}^k)$ → GD update for $F(\cdot)$

2. If $\bar{w}^{k+1} \notin C$, then
   $\bar{w}^{k+1} \leftarrow \underset{\bar{v} \in \Omega}{\operatorname{argmin}} \| \bar{w}^{k+1} - \bar{v} \|^2$



Repeat steps 1 and 2 until convergence.