Lecture 5 : Projections

EE227C . Lecturer: Professor Martin Wainwright . Scribe: Alvin Wan

Up until now, we have seen convergence rates of unconstrained gradient descent. Now, we consider a constrained minimization problem where the set of constraints $C \in \mathbb{R}^d$ is convex and closed.

$$\min_{x \in C} f(x)$$

1 Objective Function

Consider some $x^0 \in C$. It might be that our gradient takes us outside of C.

$$x^0 = x^0 - \alpha \nabla f(x^0)$$

The simplest thing we can do is to project the point back onto the set, to get an $x^1 \in C$.

$$x^1 = \Pi_e(x^0 - \alpha \nabla f(x^0))$$

where Π is the projection onto C. More explicitly, we are looking for the following.

$$\text{argmin}_{x \in C} \|x - (x^0 - \alpha \nabla f(x^0))\|_2$$

This algorithm is called **projected gradient descent**. With extra projection properties and geometry, we can develop more understanding for this constrained version of gradient descent. Is this problem hard? If C is very complex - perhaps a polytope with many linear constraints - the problem may be too difficult. The answer is thus dependent on C.

2 Background

Lemma: Say f is convex and differentiable. Then

$$x^* \in \text{argmin}_{x \in C} f(x) \leftrightarrow \langle \nabla f(x^*), z - x \rangle \geq 0$$

Without constraints, we're simply looking for a point with a gradient of zero. The above is the natural generalization with a constraint set C. Consider the geometry of this optimality condition. Take the optimal value x^* to be at the boundary of C and a gradient that points out of C. The condition gives us information about the gradient vector and any other vector pointing inwards. In effect, the condition tells us that we cannot find any feasible descent direction; equivalently, the angle between the descent direction $-\nabla f(x^*)$ and feasible direction $z - x^*$ is at least 90 degrees.

What happens if the above condition holds for some $x^* \in \text{int}(C)$? We'll see that this actually implies $\nabla f(x^*) = 0$. To some degree, this means your constraints had no effect.

3 Projection Operator

To ensure that the objective function makes sense, we have to check that there exists a unique solution. Take the Euclidean projection on C, a closed, convex subset of $\mathbb{R}^d$. Consider $\min_{x \in C} \|y - x\|_2^2$ for some fixed $y \in \mathbb{R}^d$. Examine $\|y - x\|_2^2$, we see the function value diverges or goes to infinity. We can also consider sub-level sets of g $(x \in \mathbb{R}^d | g(x) \leq \gamma)$. Compactness and closedness implies existence. Strict convexity implies uniqueness. So, the objective function makes sense. We can then define an operator.

Hence: $\Pi_e : \mathbb{R}^d \to C$, where $\Pi_e(y) = \text{argmin}_{x \in C} \|x - y\|_2^2$ is well-defined.

In general, the projection is not linear. However, it still has a few nice properties, characterized by the following inequalities:

$$\langle \Pi_C(y) - y, z - \Pi_C(y) \rangle \geq 0, \forall z \in C$$

The angle between the (a) difference between the projection $\Pi_C(y)$ and the original point y and (b) any feasible direction is non-negative. This characterizes, in short, the angle between those two vectors. To prove this, we see $\Pi_C(y) - y = \nabla f(\Pi_C(y))$.

Consider a linear subspace C and the projection of some y on C, $\Pi_C(y)$. This lemma is saying that you can move along any direction in C. In $\mathbb{R}^2$ with C as a line, we have two possible directions along C from $\Pi_C(y)$, both of which form angles with $y-\Pi_C(y)$ that total to 180. This means both are 90 and thus $y - \Pi_C(y)$ is orthogonal to C.

$$\langle \Pi_C(y) - y, z - \Pi_C(y) \rangle = 0$$

The projection also has one more important property

Consider the constraint set C and two points outside of C, $y, \tilde{y} \in \mathbb{R}^d$. We are interested in the difference between the projections $\Pi_C(y) - \Pi_C(\tilde{y})$ compared to the difference between the original points $y - \tilde{y}$. We claim the projection is **non-expansive** or that

$$\|\Pi_C(y) - \Pi_C(\tilde{y})\|_2 \leq \|y - \tilde{y}\|_2$$

If $y, \tilde{y} \in C$, then the above inequality is an equality. Note that this is not true in general if C is not convex. For example, take the Euclidean sphere $S = \{x \in \mathbb{R}^d | \|x\|_2 = 1\}$, which is comprised of only the "shell" of a ball. Consider two points in the sphere, and their projection onto the shell. Note that the distance between the projections can be greater than the distance between the original points. This matters in practice. Often, when solving for eigenvalues, which is a quadratic program over a sphere, we cannot assume that algorithms such as the power method are non-expansive.

4 Examples

Consider some examples of projections.

(a) $O_+ = \{x \in \mathbb{R}^d | x_j \geq 0, \forall j = 1 \ldots d\}$. This, intuitively, clips the function $\Pi_{O+}(y) = \max\{0, y\}$ component-wise.

(b) Box $[0,1]^d = \{x \in \mathbb{R}^d | \|x\|_\infty \leq 1\}$ where

$$\Pi_{[0,1]^d}(y) = \begin{cases} 0 & \text{if } y \leq 0 \\ y & \text{if } y \in [0,1] \\ 1 & \text{if } y \geq 1 \end{cases}$$

(c) Kite $C = \{x \in \mathbb{R}^d | \|x\|_1 \leq R\}$ forces a sparse solution. You may believe that a large set of features can be reduced to a small number of important features. This is called soft sparsity. We can show that

$$[\Pi_e(y)]_j = \mathrm{sgn}(y_i) \max\{0, \|y_j\| - \lambda\}$$

, where λ is chosen such that $\Pi_C(y) \leq 1$. this is called soft thresholding. What does soft thresholding do? Define a mapping that takes some real-valued u and gives a continuous function. Between $[-\lambda, \lambda]$, the function is constant at 0. After λ, it slopes up up, and less than $-\lambda$, it also slopes up.

$$T_\lambda(u) = \mathrm{sgn}(u) \max\{0, |u| - \lambda\}$$

This function is non-expansive. On the other hand, hard-thresholding is discontinuous

$$H_\lambda(u) = \begin{cases} 0 & \text{if } |u| \leq \lambda \\ u & \text{otherwise} \end{cases}$$

(d) Matrix problems $S_+^{d \times d} = \{X \in \mathbb{R}^{d \times d} | X = X^T, X \geq 0\}$. Consider the frobenius norm$\|X - Y\|_F^2 = \sum_{i,j=1}^d (X_{ij} - Y_{ij})^2$. What would the projection on the PSD cone of a diagonal matrix look like? We would see each entry along the diagonal clipped by a minimum of 0. Say Y is not diagonal. If it is symmetric, we can diagonalize it. $Y = U^T D U$ where $U \in \mathbb{R}^{d \times d}$ is orthogonal and $U^T U = I$. D is diagonal diag$\{\lambda_1, \ldots \lambda_d\}$. For any $X \in S_+^{d \times d}$, we have $\tilde{X} = U X U^T$ which is also in $S_+^{d \times d}$.

$$\|Y - X\|_F^2 = \|U^T(D - \tilde{X})U\|_F^2 = \|D - \tilde{X}\|_F^2$$

In other words, the frobenius norm is unitary invariant. This means that multiplying by unitary matrices does not change its value. This also means that there is no dependence on the eigenvectors. This means the norm is purely a spectrum of the matrix. If you minimize over all X, is it the same as minimizing over all $\tilde{X}$. For a symmetric matrix, simply diagonalize, clip and the re-assemble.

(e) Take the nuclear norm. For symmetric $X \in \mathbb{R}^{d \times d}$, we have $\|X\|_{nuc} = \sum_{j=1}^d |\lambda_j(X)|$. Note that the frobenius norm is effectively L-2 applied to the eigenspectrum. $\|X\|_F = \sqrt{\sum_{j=1}^d \lambda_j^2(D)}$, whereas the nuclear is effectively L-1 applied to the eigenspectrum. The nuclear norm is useful in practice because it is a convex surrogate for "low-rankness". When you run PCA, you are minimizing the frobenius norm between the matrix A and its low-rank approximations. There are other forms of low-rank approximations that are computationally intractable. Take the nuclear norm and consider its ratio with the frobenius norm. Suppose your matrix is rank $r << d$.

$$\frac{\|X\|_{nuc}}{\|X\|_F} \leq \sqrt{r}$$

This tells you how close to low rank your matrix X is. This is analogous to the L-1 , L-2 norms for vectors.

$$\frac{\|x\|_1}{\|x\|_2} \leq \sqrt{k}$$

for all k-sparse $x \in \mathbb{R}^d$ with at most k non-zeros. What does the projection onto the nuclear norm ball look like? We are soft-thresholding the eigenvalues. We can check that the nuclear norm will be unitarily invariant, since it only depends on the eigenvalues.

The above are all polynomial time examples. Next time, we will analyze the projected gradient descent algorithm itself:

$$x^{l+1} = \Pi_C(x^l - \alpha \nabla f(x^l))$$

We will show that with (m, M)-convex, smooth guarantees, we will show $\|x^l - x^*\|_2 \leq (\frac{1-m/M}{1+m/M})^l \|x^0 - x^*\|_2$.