Lecture #23: Optimality Conditions

Last time we talked about a class of optimization problems:

"equality constrained quadratic programming"

we reasoned about their optimal solutions via closed form expressions, e.g.

\[
\text{(LS) } \min \|x\|^2 \\
\text{s.t. } Ax = b \\
\Rightarrow \quad x^* = A^T(AA^T)^{-1}b
\]

Key Point: For more complex problems we won't always have closed-form solutions!

Instead we will use iterative methods to find an optima
Main Question: How will we know when to stop? i.e., what properties hold at optimal $x^*$?

Before moving on to harder problems, let’s think about optimality conditions for QP

First we need to understand how to take the derivative in matrix-vector notation.

**Def:** The gradient of function $f(x_1, x_2, \ldots, x_d)$ is a d-dimensional vector

$$
\nabla f = \left[ \frac{\partial f(x_1, \ldots, x_d)}{\partial x_1}, \ldots, \frac{\partial f(x_1, \ldots, x_d)}{\partial x_d} \right].
$$

Let’s revisit the example from last time:

$$
f(x, y) = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 2 \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} + 1
$$
Q: So what is the gradient?

$$\frac{1}{\partial x} f(x,y) = \frac{1}{\partial x} (2x^2 - 2xy + 2y^2) = 4x - 2y$$

Similarly, $$\frac{1}{\partial y} f(x,y) = 4y - 2x$$

We can actually express the answer in matrix-vector notation:

$$\nabla f = 2 \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 4x - 2y \\ 4y - 2x \end{bmatrix}$$

This is true more generally!

**Fact:** Let $$f(z) = z^T A z$$ then $$\nabla f(z) = 2A z$$

This should look sort of familiar from univariate calculus:

$$f(z) = az^2 \Rightarrow \frac{1}{\partial z} f(z) = 2az$$
Let's do another important example

**Fact:** Let $f(z) = z^T b$. Then $\nabla f(z) = b$

**Poll:** Let $f(z) = b^T z$. What is $\nabla f(z)$?

(a) $b^T$  (b) $b$

Now that we understand gradients, let's return to QP

$$\min_x \frac{x^T P x + q^T x}{2} = f(x)$$

s.t. $Ax = b$

Now the set of feasible points, defined as $\{ x | Ax = b \}$ is a plane (not necessarily thru origin)
And the gradient is the direction of largest increase

\[ f(x + \delta) \approx f(x) + \delta^T \nabla f(x) \]

Two things can happen:

**Case #1:** The gradient is not orthogonal to the directions you can move while maintaining feasibility

Q2: Is the current solution \( x \) optimal?

No, because you can move a small amount in \( N(A^T) \) and decrease the objective.
Q3: Why did I say a small amount?
Because \( f(x) + \delta^T Qf(x) \) is only a good approximation when \( \delta \) is small.

Case #2: The gradient is orthogonal to \( N(A) \).

Lemma: If \( f_f(x) \perp N(A) \) then \( x \) is locally optimal.

Let's figure out what this means for QPs specifically.

\[
\nabla f(x) = Px + q \perp N(A)
\]

This is the same thing as

\[
Px + q \in N(A)^\perp
\]
But recall $C(A^T) = N(A)^\perp$, thus

**Lemma:** For equality constrained $\mathcal{Q} \mathcal{P}$, a feasible point $x$ is locally optimal iff

$$(A\mathbf{x} = \mathbf{b})$$

$$p\mathbf{x} + \mathbf{e} \in C(A^T)$$

or alternatively if there is $\mathbf{w}$ s.t.

$$p\mathbf{x} + \mathbf{e} = A^T\mathbf{w}$$

We can actually incorporate feasibility into the condition too:

$$\exists \mathbf{w} \text{ s.t. } x \text{ is feasible and locally optimal } \iff \begin{bmatrix} p & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} x \\ \mathbf{w} \end{bmatrix} = \begin{bmatrix} -\mathbf{e} \\ \mathbf{b} \end{bmatrix}$$

Actually things are even nicer:

**Lemma:** For equality constrained $\mathcal{Q} \mathcal{P}$ with PSD $p$ and a feasible point $x$,

$$x \text{ is locally optimal } \iff x \text{ is globally optimal}$$
Let's see why this is true.

Let \( x = \) feasible and locally optimal.

Let \( x + \varepsilon = \) feasible.

Now let's compare their objective values

\[
    f(x + \varepsilon) = \frac{1}{2} (x + \varepsilon)^T P (x + \varepsilon) + (x + \varepsilon)^T q
    
    = \frac{1}{2} \varepsilon^T P \varepsilon + \varepsilon^T P x + \varepsilon^T q + f(x)
\
\]

I claim this zero.

Let's see why: From the local optimality condition we have \( \exists \nu \)

\[
    \begin{bmatrix}
        P & A^T \\
        A & 0
    \end{bmatrix}
    \begin{bmatrix}
        x \\
        \nu
    \end{bmatrix}
    =
    \begin{bmatrix}
        -q \\
        b
    \end{bmatrix}
\
\]

\( \Rightarrow \) \( P x + A^T \nu = -q \)

\( \Rightarrow \) \( P x = -q - A^T \nu \)
\[ \Rightarrow \mathbf{z}^T P \mathbf{x} = -\mathbf{z}^T \mathbf{q} + \mathbf{z}^T \mathbf{A} \mathbf{z} \quad \text{for } \mathbf{z} \in \mathcal{N}(\mathbf{A}) \]
\[ \Rightarrow \mathbf{z}^T P \mathbf{x} + \mathbf{z}^T \mathbf{q} = 0 \quad \text{same as } \mathbf{z}^T \mathbf{A} \mathbf{z} \]

Now putting it all together:

\[ f(\mathbf{x} + \mathbf{z}) = \frac{1}{2} \mathbf{z}^T P \mathbf{z} + f(\mathbf{x}) \]
\[ \geq 0 \quad \text{since } P \text{ is PSD} \]

Thus \( \mathbf{x} \) is globally optimal (though there could be others).

Now let's return to an earlier thread:

**Nature can solve interesting optimization problems like LS**

**Anything else?**

**Slime mold can solve shortest path**
First it coats the entire maze, then settles on most efficient route to food source.

What next? Sudoku?

Another example: Foams/Bubbles

- smallest surface area of given volume
What if we give it some landmarks to attach to?

This looks like the **Steiner tree problem**:

Given a set of nodes, connect them using minimum total length

e.g.

Natural Conjecture: Nature finds the optimal
The trouble is this problem is “hard,” like “concave minimization.”

Hmm. If my laptop can’t always find the best solution, but soap does, should I switch CPUs?

( Blog Post )

Well no, because it doesn’t find a global optima

Scott Aaronson, ca. 2007
Let's put this all together:

Steiner tree / non-convex optimization

Iterative methods / convex optimization

QP with equality

Linear Algebra

Next time: QPs with inequalities, and applications