

Lecture 22: The Augmented Lagrangian

Charles A. Bouman

Purdue University

Friday, October 16, 2026

Outline

- 1 The problem: we need λ^* , but $g(\lambda)$ has no explicit form
- 2 The augmented Lagrangian idea
- 3 The canonical convex problem, its dual, and λ^*
- 4 The augmented Lagrangian $L(x; a, \lambda)$ and the role of a
- 5 The Lemma, and its proof
- 6 The augmented Lagrangian algorithm and the dual-space recursion
- 7 Convergence of the augmented Lagrangian
- 8 The scaled form $u = \lambda/a$, and pseudocode
- 9 How to choose a

Core idea

Augment the problem with the quadratic term $\frac{a}{2} \|Cx - b\|^2$. An unexpected result of adding this term is that we can then derive a recursion for λ that converges to the desired value of λ^* .

Today's Big Ideas

- The optimal Lagrange multiplier maximizes the dual function, but the dual function usually has no explicit form, so the multiplier cannot be found by maximizing it directly.
- The augmented Lagrangian adds a quadratic term in the constraint error to the Lagrangian, which drives the solution toward the constraint.
- One unconstrained minimization of the augmented Lagrangian is equivalent to one restrained maximization of the dual function, and it produces a simple update of the multiplier.
- The augmented Lagrangian algorithm alternates that minimization with that update, it converges to the optimal multiplier, and it is usually implemented in a scaled form.
- Convergence does not depend on the size of the quadratic term, but a larger term converges in fewer iterations while making each minimization harder.

Where we are stuck

Lagrange multipliers provide a beautiful theory for solving constrained optimization problems. However, at this point, it is unclear how to practically compute the value of λ^* that will properly enforce the constraint.

- In some analytical problems, it is possible to explicitly solve the constraint for the desired value of λ^* . However, for large numerical problems, this is not possible.
- Another approach is to determine λ^* by maximizing the dual function $g(\lambda)$. However, the function $g(\lambda)$ usually does not have an explicit form that can be easily maximized.

So at this point, the framework of Lagrange multipliers and primal-dual functions seems beautiful but perhaps somewhat useless.

The augmented Lagrangian idea

Fortunately, the **augmented Lagrangian** method (Hestenes 1969, Powell 1969) solves this problem by providing a numerically stable approach for computing λ^* by **simultaneously solving both the primal and the dual problem**.

- The augmented Lagrangian approach results in an explicit algorithm that is easily implemented and that is theoretically guaranteed to converge to the optimal Lagrange multiplier, λ^* .
- It works by augmenting the basic optimization problem with an additional quadratic term that helps drive the solution toward the constraint.
- An unexpected result of adding this additional term is that we can then derive a recursion for λ that converges to the desired value of λ^* .

The canonical convex optimization problem

The canonical convex optimization problem is the basis for the methods and theory of augmented Lagrangian optimization. It is defined as

$$p^* = \inf_{\substack{x \in \Omega \\ Cx=b}} f(x) .$$

It has the following properties:

- $f : \Omega \rightarrow \mathbb{R}$ is a convex function on a closed convex set $\Omega \subset \mathbb{R}^N$.
- Slater's conditions hold.
- There exists a solution $x^* \in \Omega$ such that $p^* = f(x^*)$ and $Cx^* = b$.

The associated dual function for this problem is then given by

$$g(\lambda) = \inf_{x \in \Omega} \{f(x) + \lambda^t(Cx - b)\} ,$$

and by Slater's Theorem, the dual function takes on its maximum,

$$\lambda^* = \arg \max_{\lambda \in \mathbb{R}^K} g(\lambda) .$$

The augmented Lagrangian cost function

The augmented Lagrangian cost function is defined as

$$L(x; a, \lambda) = f(x) + \lambda^t(Cx - b) + \frac{a}{2}\|Cx - b\|^2 ,$$

where the additional term, $\frac{a}{2}\|Cx - b\|^2$, is added in order to drive the solution toward the constraint $Cx = b$.

- Notice that as a becomes larger, optimization will naturally tend to reduce the value of $\|Cx - b\|^2$ and therefore enforce the constraint $Cx = b$. This would appear to be an effective practical technique to enforce the desired constraint.
- However, if a is chosen to be too large, then the optimization problem can become very stiff and convergence will be very slow.

So our goal will be to find a tractable recursion for λ that will produce the correct value of the Lagrange multiplier.

Lemma: the augmented Lagrangian equality

Lemma

Consider the convex optimization problem above, with dual function $g(\lambda)$ and augmented Lagrangian $L(x; a, \lambda)$. Then for all $\lambda \in \mathbb{R}^K$,

$$\inf_{x \in \Omega} L(x; a, \lambda) = \sup_{\lambda' \in \mathbb{R}^K} \left\{ g(\lambda') - \frac{1}{2a} \|\lambda' - \lambda\|^2 \right\},$$

where the infimum and supremum are achieved. Furthermore, the values $x^{**} \in \Omega$ and $\lambda^{**} \in \mathbb{R}^K$ that achieve the infimum and supremum satisfy the recursion

$$\lambda^{**} = \lambda + a(Cx^{**} - b).$$

- Left side: one unconstrained minimization over x . Right side: a restrained maximization of the dual function.

Proof, step 1: x^{**} and λ^{**} exist

The proof we present is adapted from Bertsekas (1989), p. 244.

- First, notice that $g(\lambda)$ is concave and, due to Slater's conditions, it achieves its maximum.
- Therefore, $g(\lambda') - \frac{1}{2a} \|\lambda' - \lambda\|^2$ must be strictly concave and must also uniquely achieve its maximum.
- Next, since $f(x)$ is convex and x^* achieves the constrained minimum, the set $\{x \in \Omega : L(x; a, \lambda) \leq T\}$ must be compact for some T .
- Therefore, $L(x; a, \lambda)$ must take on a global minimum.

Proof, step 2: a sequence of equalities

$$\begin{aligned} & \inf_{x \in \Omega} L(x; a, \lambda) \\ &= \inf_{x \in \Omega} \left\{ f(x) + \lambda^t (Cx - b) + \frac{a}{2} \|Cx - b\|^2 \right\} \\ &= \inf_{\substack{x \in \Omega, z \in \mathbb{R}^K \\ z = Cx - b}} \left\{ f(x) + \lambda^t z + \frac{a}{2} \|z\|^2 \right\} \\ &= \sup_{\lambda'} \inf_{z \in \mathbb{R}^K, x \in \Omega} \left\{ f(x) + [\lambda']^t (Cx - b - z) + \lambda^t z + \frac{a}{2} \|z\|^2 \right\} \\ &= \sup_{\lambda'} \left\{ \inf_{x \in \Omega} \left\{ f(x) + [\lambda']^t (Cx - b) \right\} + \inf_{z \in \mathbb{R}^K} \left\{ (\lambda - \lambda')^t z + \frac{a}{2} \|z\|^2 \right\} \right\} \\ &= \sup_{\lambda'} \left\{ g(\lambda') + \inf_{z \in \mathbb{R}^K} \left\{ (\lambda - \lambda')^t z + \frac{a}{2} \|z\|^2 \right\} \right\} . \end{aligned}$$

Proof, step 2: why each equality holds

- **First equality:** the definition of the function $L(x; a, \lambda)$.
- **Second equality:** substituting z for the expression $Cx - b$ and applying the constraint $z = Cx - b$.
- **Third equality:** introducing a new Lagrange multiplier, λ' , to enforce the constraint $z = Cx - b$, and then applying the strong duality property.
- **Fourth equality:** collecting terms in x and z .
- **Fifth equality:** the definition of $g(\lambda)$.

The new multiplier λ' is what turns the minimization over x into a maximization over the dual variable.

Proof, step 3: the explicit minimum over z

Next, we can simplify the expression in the last term by calculating the explicit minimum. This yields the result that

$$g(\lambda') + \inf_{z \in \mathbb{R}^K} \left\{ (\lambda - \lambda')^t z + \frac{a}{2} \|z\|^2 \right\} = g(\lambda') - \frac{1}{2a} \|\lambda - \lambda'\|^2 ,$$

where it is easily shown that minimization over z results when

$$z = \frac{\lambda' - \lambda}{a} .$$

Using this result, we have that

$$\begin{aligned} \inf_{x \in \Omega} L(x; a, \lambda) &= \sup_{\lambda'} \left\{ g(\lambda') - \frac{1}{2a} \|\lambda - \lambda'\|^2 \right\} \\ &= g(\lambda^{**}) - \frac{1}{2a} \|\lambda - \lambda^{**}\|^2 , \end{aligned}$$

where λ^{**} is defined as the value that achieves the maximum. So this proves the first result of the lemma.

Proof, step 4: the recursion for λ

Notice that when $\lambda' = \lambda^{**}$, then the expression for z becomes

$$z^{**} = \frac{\lambda^{**} - \lambda}{a},$$

where z^{**} is the associated value of z when $\lambda' = \lambda^{**}$.

- This is the result of solving the constrained optimization with $z = Cx - b$. So, in particular, it must be that $z^{**} = Cx^{**} - b$, where x^{**} is the value of x that achieved the minimum of $L(x; a, \lambda)$.

Substituting this result then yields that

$$Cx^{**} - b = \frac{\lambda^{**} - \lambda}{a},$$

and, rearranging terms, we have the final result that

$$\lambda^{**} = \lambda + a(Cx^{**} - b),$$

which completes the proof.

The augmented Lagrangian algorithm

The **augmented Lagrangian algorithm** follows from the lemma. It is the following conceptually simple recursion in k :

$$x^{(k+1)} = \arg \min_{x \in \Omega} L(x; a, \lambda^{(k)}) ,$$

$$\lambda^{(k+1)} = \lambda^{(k)} + a(Cx^{(k+1)} - b) .$$

- The optimization step computes the unconstrained minimum of the augmented Lagrangian cost function with respect to x .
- The update step computes a very simple update of the Lagrange multiplier based on the error in the constraint.

The equivalent recursion in dual space

We can now show that these iterations converge by using the Lemma. From the lemma, we know that each update of this recursion is equivalent to the following recursion in $\lambda^{(k)}$:

$$\lambda^{(k+1)} = \arg \max_{\lambda \in \mathbb{R}^K} \left\{ g(\lambda) - \frac{1}{2a} \|\lambda - \lambda^{(k)}\|^2 \right\} .$$

In fact, it is easily shown that this iteration must converge to

$$\lim_{k \rightarrow \infty} \lambda^{(k)} = \lambda^* = \arg \max_{\lambda \in \mathbb{R}^K} g(\lambda) .$$

Intuition

Each iteration attempts to maximize $g(\lambda)$ but is restricted by the term $\frac{1}{2a} \|\lambda - \lambda^{(k)}\|^2$ which “pulls” the solution toward the previous value of $\lambda^{(k)}$. However, after each new iteration, this restriction becomes weaker, and the solution asymptotically converges to the global maximum of $g(\lambda)$.

Property: convergence of augmented Lagrangian

Convergence of augmented Lagrangian

Consider the convex optimization problem above. Then the augmented Lagrangian algorithm converges in the sense that

$$\lim_{k \rightarrow \infty} \lambda^{(k)} = \lambda^*, \quad x^* = \arg \min_{x \in \Omega} L(x; a, \lambda^*),$$

where $\lambda^* = \arg \max_{\lambda \in \mathbb{R}^K} g(\lambda)$ and x^* is a solution to the constrained optimization problem.

Proof. We have already proved the convergence of $\lambda^{(k)}$ to the maximum of $g(\lambda)$. By a combination of Slater's Theorem and the optimal Lagrange multiplier property, strong duality must hold and any solution to the constrained problem must be a solution to $x^* = \arg \min_{x \in \Omega} L(x; 0, \lambda^*)$, where $a = 0$. Since x^* is assumed to meet the constraint, we know that $Cx^* - b = \mathbf{0}$ and therefore that $x^* = \arg \min_{x \in \Omega} L(x; a, \lambda^*)$, which completes the proof.

The scaled form: $u = \lambda/a$

Finally, we can reorganize the terms of the augmented Lagrangian to put it in an equivalent but slightly more compact form:

$$x^{(k+1)} = \arg \min_{x \in \Omega} \left\{ f(x) + \frac{a}{2} \|Cx - b + u^{(k)}\|^2 \right\} ,$$

$$u^{(k+1)} = u^{(k)} + Cx^{(k+1)} - b .$$

- It is easy to show that this new form is equivalent to the earlier form by expanding the expression for $\frac{a}{2} \|Cx - b + u^{(k)}\|^2$.
- In the new form, the vector u serves the role of the Lagrange multiplier but it is scaled so that $u = \lambda/a$.

Pseudocode: the augmented Lagrangian algorithm

Augmented Lagrangian Algorithm

Initialize $u \leftarrow \mathbf{0}$

For K iterations {

$$\hat{x} \leftarrow \arg \min_{x \in \Omega} \left\{ f(x) + \frac{a}{2} \|Cx - b + u\|^2 \right\}$$

$$u \leftarrow u + C\hat{x} - b .$$

}

Pseudocode for implementing the augmented Lagrangian algorithm for minimizing $f(x)$ subject to the constraint $Cx = b$. This iteration can be shown to converge to the optimal value of the Lagrange multiplier u^* that achieves the desired constraint.

How to choose a

It is interesting to note that the convergence of the augmented Lagrangian algorithm **does not depend on the choice of a** . In fact, a very large value of a will result in much faster convergence in terms of number of iterations. This can be seen in two ways.

- If a is large, then the term $\frac{a}{2} \|Cx - b\|^2$ will drive the solution toward the constraint more rapidly.
- Alternatively, when $\frac{1}{2a}$ is small, then the dual-space iteration will be less restricted by the term $\frac{1}{2a} \|\lambda - \lambda^{(k)}\|^2$.

However, the price one pays for this fast convergence is that each iteration can be more difficult. This is because when a is very large, then the optimization of the x update becomes more ill conditioned and, therefore, more difficult.

So the trick then becomes to balance these two objectives of fast convergence and well-conditioned minimization.

Summary

- The optimal multiplier maximizes the dual function, $\lambda^* = \arg \max_{\lambda} g(\lambda)$ with $g(\lambda) = \inf_{x \in \Omega} \{f(x) + \lambda^t(Cx - b)\}$, but $g(\lambda)$ has no explicit form to maximize.
- Augmented Lagrangian: $L(x; a, \lambda) = f(x) + \lambda^t(Cx - b) + \frac{a}{2} \|Cx - b\|^2$.
- Lemma: $\inf_{x \in \Omega} L(x; a, \lambda) = \sup_{\lambda'} \{g(\lambda') - \frac{1}{2a} \|\lambda' - \lambda\|^2\}$, with $\lambda^{**} = \lambda + a(Cx^{**} - b)$.
- Algorithm: $x^{(k+1)} = \arg \min_{x \in \Omega} L(x; a, \lambda^{(k)})$ and $\lambda^{(k+1)} = \lambda^{(k)} + a(Cx^{(k+1)} - b)$, which in dual space is $\lambda^{(k+1)} = \arg \max_{\lambda} \{g(\lambda) - \frac{1}{2a} \|\lambda - \lambda^{(k)}\|^2\}$ and therefore converges, $\lambda^{(k)} \rightarrow \lambda^*$.
- Scaled form with $u = \lambda/a$ gives the implemented algorithm, $x^{(k+1)} = \arg \min_{x \in \Omega} \{f(x) + \frac{a}{2} \|Cx - b + u^{(k)}\|^2\}$ and $u^{(k+1)} = u^{(k)} + Cx^{(k+1)} - b$.
- Convergence does not depend on a . A large a means fewer iterations but a more ill-conditioned minimization at each one.

Reading: Section 9.3. Next lecture: Section 9.4, Proximal Maps and Shrinkage.