



# A Lagrange Multipliers Refresher, For Idiots Like Me

Jul 27, 2019

## What Are Lagrange Multipliers?

Lagrange multipliers are a tool for doing constrained optimization. Say we are trying to minimize a function  $f(x)$ , subject to the constraint  $g(x) = c$ .

$$\begin{aligned} \min f(x) \\ \text{subject to } g(x) = c \end{aligned}$$

To solve this, you define a new function.

$$\mathcal{L}(x, \lambda) = f(x) - \lambda(g(x) - c)$$

The optimum lies at a stationary point of  $\mathcal{L}$  (a point where the gradient in  $x$  and  $\lambda$  is both zero).

This is all true, but it doesn't explain why it works.

## Why Do Lagrange Multipliers Work?

Let's consider a variant of the problem. We want to minimize  $f(x)$  subject to  $g(x) \geq 0$ .

$$\begin{aligned} \min f(x) \\ \text{subject to } g(x) \geq 0 \end{aligned}$$

Let's define the following min-max optimization problem.

$$\min_x \max_{\lambda \geq 0} f(x) - \lambda g(x)$$

I claim the solution  $x$  of this optimization problem occurs at the smallest  $f(x)$  that satisfies the constraint  $g(x) \geq 0$ . Why?

As written, we first choose an  $x$ , then choose a  $\lambda$  that maximizes the objective, and we want to choose an  $x$  that minimizes how much an adversarial  $\lambda$  can hurt us. Suppose we are

violating the constraint  $g(x) \geq 0$ . Then we have  $g(x) < 0$ . At such an  $x$ ,  $-g(x) > 0$ , so we can pick  $\lambda = \infty$  to drive the objective value to  $\infty$ .

If we are not violating the constraint, then  $-g(x)$  is 0 or negative, and since  $\lambda$  is constrained to  $\lambda \geq 0$ , the optimal  $\lambda$  is  $\lambda = 0$ , giving objective value  $f(x)$ .

In other words, the solution of the unconstrained problem is the same as the solution to the original constrained problem.

This handles the  $g(x) \geq 0$  case. What if we have multiple constraints?

$$\begin{aligned} \min \quad & f(x) \\ \text{subject to} \quad & g_1(x) \geq 0 \\ & g_2(x) \geq 0 \end{aligned}$$

We can define a similar min-max problem by adding a Lagrange multiplier  $\lambda_i$  for each constraint  $i$ .

$$\min_x \max_{\lambda_1, \lambda_2 \geq 0} f(x) - \lambda_1 g_1(x) - \lambda_2 g_2(x)$$

By a similar argument, the optimal solution to this min-max problem occurs at  $\lambda_1 = 0, \lambda_2 = 0$ , and  $x$  at the solution of the original constrained optimization problem, assuming it exists. If either constraint was violated, then we could have driven the corresponding  $\lambda_i$  to  $\infty$ , like before.

What if we have a constraint  $g(x) \geq c$ ? We can rearrange it to the constraint  $g(x) - c \geq 0$ .

$$\begin{aligned} \min \quad & f(x) \\ \text{subject to} \quad & g(x) - c \geq 0 \end{aligned}$$

This is solved the same way.

$$\min_x \max_{\lambda \geq 0} f(x) - \lambda(g(x) - c)$$

If we have a constraint  $g(x) \leq 0$ , we can negate both sides to get  $-g(x) \geq 0$ . If we have a constraint  $g(x) \leq c$ , we can rearrange it to  $c - g(x) \geq 0$ . This reduces everything to the  $g(x) \geq 0$  case we know how to solve.

Bringing it back to the equality case, let's say we have a constraint  $g(x) = c$ . How do we reduce this to the previous cases? This equality is the same as the pair of constraints  $g(x) \geq c, g(x) \leq c$ , which are only both satisfied at  $g(x) = c$ . This time, I'll write the Lagrange multipliers as  $\lambda_+$  and  $\lambda_-$ .

$$\begin{aligned} \min \quad & f(x) \\ \text{subject to} \quad & g(x) \geq c \\ & g(x) \leq c \end{aligned}$$

$$\min_x \max_{\lambda_+, \lambda_- \geq 0} f(x) - \lambda_+(g(x) - c) - \lambda_-(c - g(x))$$

Like before, if we ever have  $g(x) \neq c$ , then we can choose  $\lambda_+, \lambda_-$  such that the objective value shoots up to  $\infty$ . But, since  $g(x) - c$  and  $c - g(x)$  are just negations of each other, we can simplify this further.

$$\min_x \max_{\lambda_+, \lambda_- \geq 0} f(x) - (\lambda_+ - \lambda_-)(g(x) - c)$$

It's possible to make  $\lambda_+ - \lambda_-$  equal any real number while still satisfying  $\lambda_+ \geq 0, \lambda_- \geq 0$ , so let's just replace  $\lambda_+ - \lambda_-$  with  $\lambda$ , and say  $\lambda$  can be any real number instead of only nonnegative ones.

$$\min_x \max_{\lambda} f(x) - \lambda(g(x) - c)$$

Now, we're back to the equality case we started at. The solution to this must lie at a saddle point where the gradient with respect to  $x$  and  $\lambda$  is both zero.

## Why Do I Like This Explanation?

When I was re-learning Lagrange multipliers a while back, I was upset that all the most popular results were targeted towards people taking their first multivariate calculus course. These explanations exclusively covered the  $g(x) = c$  case, and the constraints I wanted to add were more like  $a \leq g(x) \leq b$ .

It's a shame that most people's first introduction to Lagrange multipliers only covers the equality case, because inequality constraints are more general, the concepts needed to understand that case aren't much harder, and it's clearer how you'd apply an optimization algorithm to solve the resulting unconstrained problem. Like all other min-max optimization (GANs, actor-critic RL, etc.), doing alternating gradient descent updates on  $x$  then  $\lambda$  is both simple and works out fine.

This procedure doesn't guarantee that your constraint  $g(x) \geq c$  is met over the entire optimization process, but it does quickly penalize the optimization for violating the constraint, since the  $\lambda$  for that constraint will quickly rise and keep rising until the constraint is satisfied again, at which point  $\lambda$  will quickly regress towards 0 until it is needed once more.

As one final detail, handling inequalities requires handling optimization over  $\lambda \geq 0$ . The common trick is to rewrite  $\lambda$  as  $\lambda = \exp(w)$  or  $\lambda = \text{softplus}(w)$ , then do gradient descent over variable  $w$  instead.

[← I'm Still Here](#)

[Four Years Later →](#)

ALSO ON SORTA INSIGHTFUL

Three Years Later

2 years ago • 1 comment

Sorta Insightful turns three years old today! Whether you were here from ...

MIT Mystery Hunt 2020, Part 2

4 months ago • 2 comments

Now thatMy Little Pony: Puzzles are Magic haswrapped up, I have ...

The Blogging Gauntlet: May 27 - Structural ...

4 years ago • 1 comment

This is part of The Blogging Gauntlet of May 2016, where I try to write500 ...

One

4 ye

Ove

set u

port

2 Comments

Sorta Insightful

Disqus' Privacy Policy

Sandeep ▾

Recommend 1

Tweet

Share

Sort by Best ▾

Join the discussion...

Tim Vieira • a year ago

Langrange -> Lagrange

1 ^ | ▾ • Reply • Share ›

alexirpan Mod ➔ Tim Vieira • a year ago

should be fixed now

^ | ▾ • Reply • Share ›

Subscribe

Add Disqus to your siteAdd DisqusAdd

Do Not Sell My Data

Sorta Insightful

Email: alexirpan [at] berkeley [dot] edu

alexirpan