University of Illinois Fall 2018

ECE 544NA: Pattern Recognition
Lecture 7: Optimization Dual
(Sep 18 and Sep 20)

Lecturer: Alexander Schwing Scribe: Arvind Kamal

Goals for Lecture 7:

- Examining constrained optimization
- Understanding duality of optimization

Related Reading(s):

- S. Boyd and L. Vandenberghe; Convex Optimization; Chapter 5

1 Recap Time!

So far we have studied 2 kinds optimization problems, formulated their programs and derived ways to compute their optimum solutions. These have been summarized below:

Linear Regression	**Logistic Regression**
Program $\min_{\boldsymbol{w}} \frac{1}{2} \sum_{(x^{(i)},y^{(i)})\in\mathcal{D}} \left(y^{(i)} - \phi(x^{(i)})^\top \boldsymbol{w}\right)^2$	Program $\min_{\boldsymbol{w}} \sum_{(x^{(i)},y^{(i)})\in\mathcal{D}} \log\left(1+\exp(-y^{(i)}\boldsymbol{w}^T\phi(x^{(i)}))\right)$
Analytical Solution	No Analytical Solution; Use Gradient Descent

In both cases there is a common goal, which is to find the smallest value **w*** that results in the smallest function value. More generally, both the programs have the following form:

$$\begin{aligned} &\min_{\boldsymbol{w}} && f_0(\boldsymbol{w}) \\ &\text{s.t.} && f_i(\boldsymbol{w}) \leq 0 \quad \forall i \in \{1,\dots,C\} \end{aligned}$$

But there is a minor problem!

2 Addressing Constraints

We successfully optimized the function at hand, but what about the constraints! More specifically, we did not take into account that $f_i(\mathbf{w}) \leq 0, \forall\ i\ \epsilon\ \{1, 2, ..., C\}$. In order to address these constraints we introduce a new function $h_i(w)$, with the following property:

$$\begin{array}{lll} \min_{\boldsymbol{w}} & f_0(\boldsymbol{w}) & \\ \text{s.t.} & f_i(\boldsymbol{w}) \leq 0 & \forall i \in \{1, \ldots, C_1\} \\ & h_i(\boldsymbol{w}) = 0 & \forall i \in \{1, \ldots, C_2\} \end{array} \tag{1}$$

Here $h_i(w) = 0 \Rightarrow$ that $\mathbf{w}$ is a feasible solution.
Now that we have determined a metric that can represent whether the constraints are being met or not, it is time to consider $f_0(\mathbf{w})$, $f_i(\mathbf{w})$ and $h_i(\mathbf{w})$ together.

2.1 Hello, Lagrangians!

For the purpose of considering the constraints of the original problem and working with $f_0(w)$, $f_i(w)$ and $h_i(w)$ together, we introduce the Lagrange Multipliers $\lambda_\mathbf{i}$ and $\nu_\mathbf{i}$, to compute the overall Lagrangian $\mathbf{L}$, as follows:

$$L(\boldsymbol{w}, \lambda, \nu) = f_0(\boldsymbol{w}) + \sum_{i=1}^{C_1} \lambda_i f_i(\boldsymbol{w}) + \sum_{i=1}^{C_2} \nu_i h_i(\boldsymbol{w}) \tag{2}$$

It is worthwhile to note:

- $\lambda_\mathbf{i}$: Lagrange Multiplier associated with the i^{th} inequality constraint, where $f_i(\mathbf{w}) \leq 0$.
- $\nu_\mathbf{i}$: Lagrange Multiplier associated with the i^{th} equality constraint, where $h_i(\mathbf{w}) = 0$.
- $\mathbf{dom(L)} = D \quad \text{x} \quad R^{C_1} \quad \text{x} \quad R^{C_2}$, where $D = \cap_{i=0}^{C_1} \mathbf{dom}\ f_i(w) \quad \cap \cap_{i=0}^{C_2} \mathbf{dom}\ h_i(w)$**.**

More generally, the Lagrangian duality is required as it modifies the objective function by taking into account weighted measures of the constraint functions. Technically speaking, the λ and ν vectors, are called the dual vectors or the Lagrange Multiplier vectors associated with $\mathbf{L}$.

2.2 The Lagrange Dual Function

If we know that if $\hat{\mathbf{w}}$ is a feasible solution for Equation **(1)**, and $\lambda_\mathbf{i} \geq 0, \forall i$, then:

$$f_0(\hat{\boldsymbol{w}}) \geq L(\hat{\boldsymbol{w}}, \lambda, \nu) \quad \text{and} \quad L(\hat{\boldsymbol{w}}, \lambda, \nu) \geq \min_{\boldsymbol{w} \in \mathcal{W}} L(\boldsymbol{w}, \lambda, \nu)$$

where $\mathcal{W}$ stands for all constraints not represented in the Lagrangian
& belong to a larger feasible set.

Let $\min_{\boldsymbol{w} \in \mathcal{W}} L(\boldsymbol{w}, \lambda, \nu)$ be represented as $g(\lambda, \nu)$. This is called the Lagrange Dual Function, or simply the Dual Function.

$$\implies f_0(\boldsymbol{w}^*) \geq g(\lambda, \nu) \quad \forall \lambda \geq 0, \nu \tag{3}$$

The Dual Function, $g(\lambda, \nu)$, more formally, will represent the minimum value of the Lagrangian over $\mathbf{w}$, for $\lambda \in \mathbf{R}^{C_1}$, $\nu \in \mathbf{R}^{C_2}$ and $g: \mathbf{R}^{C_1}$ x $\mathbf{R}^{C_2} \rightarrow \mathbf{R}$.

$$g(\lambda, \nu) = \inf_{w \in D} L(\hat{\mathbf{w}}, \lambda, \nu) = \inf_{w \in D} f_0(\boldsymbol{w}) + \sum_{i=1}^{C_1} \lambda_i f_i(\boldsymbol{w}) + \sum_{i=1}^{C_2} \nu_i h_i(\boldsymbol{w}) \tag{4}$$

From (4), we can see that the function will take on a $-\infty$ value, when the Lagrangian is unbounded over $\mathbf{w}$. Additionally, it is worth noting that g does not depend on $\mathbf{w}$, unlike the Lagrangian L. This is because the only arguments λ and ν are required as input requirements, the value of $L(\mathbf{w}, \lambda, \nu)$ is computed within the function itself.

Interestingly, equation (3) implies that, in order to get the best/largest lower bound, we should minimize the distance $f_0(\hat{w}) - g(\lambda, \nu)$, and if possible equate it to 0. This means that our aim is to maximize $g(\lambda, \nu)$. And this is our Dual Program!

2.3 Computing Dual Programs

As stated in our previous subsection, the Dual Program is $\max_{\lambda,\nu} g(\lambda, \nu)$, such that $\lambda_i \geq 0, \forall i$.
But how do we compute Dual Programs, in general? Turns out that there is a general format that we can be follow to generate these programs. The steps involved are listed below.

- Bring primal program into standard form.
- Assign Lagrange Multipliers to a set of suitable constraints
- Include all the other constraints in $\mathcal{W}$
- Formally, write down the Lagrangian, L
- Minimize the Lagrangian, L, with respect to all primal variables, for $\mathbf{w} \in \mathcal{W}$
  (could also mean compute gradient set)

With the aforementioned steps, we have streamlined an approach to compute Dual Programs. Let's try our hand at a few examples now.

Example 1: Linear Program

Given: $\min_w \mathbf{c}^T\mathbf{w}$ such that $\mathbf{Aw} \leq \mathbf{b}$

Bring into standard form:
$\min_w \mathbf{c}^T\mathbf{w}$, such that $\mathbf{Aw - b} \leq 0$

Below, in a single step we finish assigning Lagrange Multipliers to the given constraint, including any remaining constraints in $\mathcal{W}$, and writing a formal definition of $\mathbf{L}$, given $\lambda \geq 0$.

$$L(\boldsymbol{w}, \lambda, \nu) = c^T(w) + \lambda^T(Aw - b) = (c + A^T\lambda)^T w - b^T w \tag{5}$$

Minimize the Lagrangian, **L**, w.r.t. primal variables.

In equation (5), if $(c + A^T\lambda)^T$ is non-zero, then we could multiply it by any constant and drive its value to -∞, otherwise, if it is zero, then we get a Lagrangian value of $-b^T w$. More formally presented as:

$$\min_{\boldsymbol{w}} L() = \begin{cases} -\mathbf{b}^\top \lambda & \boldsymbol{A}^\top \lambda + \boldsymbol{c} = 0 \\ -\infty & \text{otherwise} \end{cases}$$

Dual Program: $\max_{\lambda \geq 0}(-b^T\lambda)$, such that $A^T + c = 0$

Example 2: Logistic Regression

Given: $\min_{\boldsymbol{w}} \sum_{(x^{(i)}, y^{(i)}) \in \mathcal{D}} \log\left(1 + \exp(-y^{(i)} \boldsymbol{w}^T \phi(x^{(i)}))\right)$

This is already in standard form.
As there are no constraints, we try reformulating the given equation.

$$\min_{\boldsymbol{w}, z^{(i)}} \frac{C}{2}\|\boldsymbol{w}\|_2^2 + \sum_{(x^{(i)}, y^{(i)}) \in \mathcal{D}} \log(1 + \exp(-z^{(i)})) \quad \text{s.t.} \quad z^{(i)} = y^{(i)} \boldsymbol{w}^\top \phi(x^{(i)})$$

Next step would be to formulate the Lagrangian, **L**

$$\begin{aligned} L(\boldsymbol{w}, \lambda, \nu) \;=\; & \frac{C}{2}\|\boldsymbol{w}\|_2^2 - \sum_{(x^{(i)}, y^{(i)}) \in \mathcal{D}} \lambda^{(i)} y^{(i)} \boldsymbol{w}^\top \phi(x^{(i)}) \\ & + \sum_{(x^{(i)}, y^{(i)}) \in \mathcal{D}} \left[\log(1 + \exp(-z^{(i)})) + \lambda^{(i)} z^{(i)}\right] \end{aligned}$$

Minimize the Lagrangian, **L**, w.r.t. primal variables.
Unlike the linear program, in this example we will be taking partial derivatives with respect to the variables.

$$\frac{\partial L}{\partial \boldsymbol{w}} : \quad C\boldsymbol{w} = \textstyle\sum_{(x^{(i)}, y^{(i)}) \in \mathcal{D}} \lambda^{(i)} y^{(i)} \phi(x^{(i)})$$

$$\frac{\partial L}{\partial z^{(i)}}: \quad \lambda^{(i)} = \frac{\exp(-z^{(i)})}{1+\exp(-z^{(i)})} \qquad \implies \lambda^{(i)} \geq 0$$
$$\implies z^{(i)} = \log \frac{1-\lambda^{(i)}}{\lambda^{(i)}} \qquad \implies \lambda^{(i)} \leq 1$$

As seen above, we introduced 2 more constraints namely $\lambda^{(i)} \geq 0$ and $\lambda^{(i)} \leq 1$. In order to compute our Dual Function, we will now plug in the previous partial differential equations into $L(\boldsymbol{w}, \lambda, \nu)$, as follows:

$$g(\lambda) = -\frac{1}{2C}\| \sum_{(x^{(i)},y^{(i)})\in\mathcal{D}} \lambda^{(i)} y^{(i)} \phi(x^{(i)})\|_2^2 + \sum_{(x^{(i)},y^{(i)})\in\mathcal{D}} H(\lambda^{(i)}) \tag{6}$$

where, $H(\lambda^{(i)}$ is binary entropy.

Some may ask, why do we maximize $g(\lambda)$ in this case?
- We have a negative quadratic term in equation (6)
- Binary Entropy, $H(\lambda^{(i)}$, is a concave function

These points suggest that a maximum can be computed in such a case. Finally, our program would look as follows:

Dual Program: $\max_{\lambda}(g(\lambda))$
such that $0 \leq \lambda^{(i)} \leq 1 \quad \forall (x^{(i)}, y^{(i)}) \epsilon \mathcal{D}$

Now, that we have some understanding of how to compute the dual program, , it shouldn't be hard to understand that in some cases, instead of optimizing the primal, we can just optimize the dual and later translate the result. Still not convinced? Let's see why optimizing the dual instead of the primal is better sometimes:

- Might have less constraints
- Will be easier to optimize
- Could get lower bounds
- Would offer interesting insights

3 Properties of Dual Programs

Refering back to section 2.2, where we derived that our Dual Program would be of the form: $\max_{\lambda,\nu} g(\lambda,\nu),\ \textit{such that}\ \lambda_i \geq 0, \forall i$, we can talk about the important properties of such programs.

$$g(\lambda,\nu) = \min_{\boldsymbol{w}\in\mathcal{W}} L(\boldsymbol{w},\lambda,\nu) := f_0(\boldsymbol{w}) + \sum_{i=1}^{C_1} \lambda_i f_i(\boldsymbol{w}) + \sum_{i=1}^{C_2} \nu_i h_i(\boldsymbol{w})$$

- Dual programs have simple constraints, if any at all
- They can be used for sensitivity analysis
- Dual programs enable lower-bounding the optimal primal value
- As we usually take point-wise minimums, they are always concave (though point-wise minimum of a linear function is a convex function)
- A dual program is the infimum over affine functions in λ, ν These programs can hold weak duality or strong duality (covered next)

3.1 Weak Duality

We mentioned about weak duality in the properties of Dual Programs. But what does that mean? Well, weak duality occurs when $f(\boldsymbol{w}^*) \geq g(\lambda^*, \nu^*)$. The difference $f - g$ is called as the optimal duality gap. This is because it represents the gap between the optimal value of the primal problem and the best (or greatest) lower bound obtained from the dual function. As can be observed from the condition, this optimal duality gap is always non-negative. It is also interesting to know that weak duality:

- Always holds for both convex and non-convex problems
- Can be used to find nontrivial lower bounds

3.2 Strong Duality

We mentioned about weak duality in the properties of Dual Programs. But what does that mean? Well, weak duality occurs when $f(\boldsymbol{w}^*) = g(\lambda^*, \nu^*)$. This means that the optimal duality gap is 0, implying that the best bound obtained from the dual function would be very tight. Some other information about strong duality tells us:

- Strong Duality does not always hold, usually true for convex problems
- The conditions within these convex problems that guarantee Strong Duality are called *constraint qualifications*. An example of such a constraint is Slater's Condition.

3.3 What does this mean Geometrically?

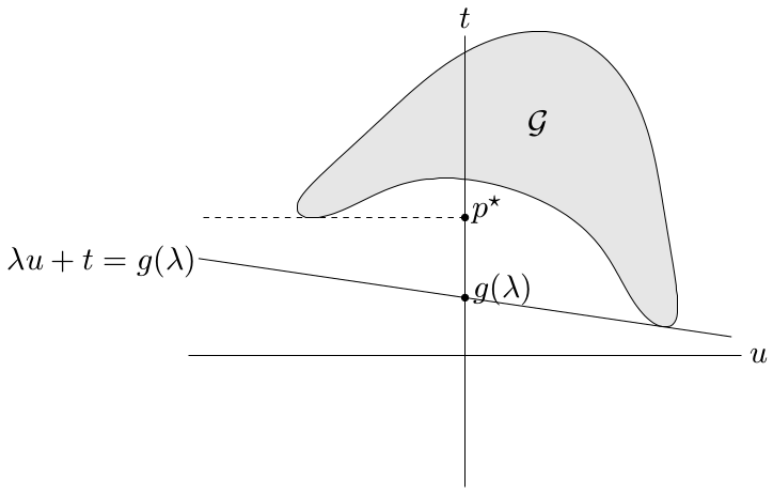
Now that we have established what it means to hold a weak or strong duality, let's understand the geometric interpretation of the dual function. Before graphically explaining different scenarios, it is good to know what $\mathcal{A}$ and $\mathcal{G}$ refer to:

$$\mathcal{A} = \{(u, v, t) \mid \exists x \in \mathcal{D},\ f_i(x) \leq u_i,\ i = 1, \ldots, m,\\ h_i(x) = v_i,\ i = 1, \ldots, p,\ f_0(x) \leq t\},$$

$$\mathcal{G} = \{(f_1(x), f_0(x)) \mid x \in \mathcal{D}\}, \textit{ simply put it refers to a hyperplane}$$

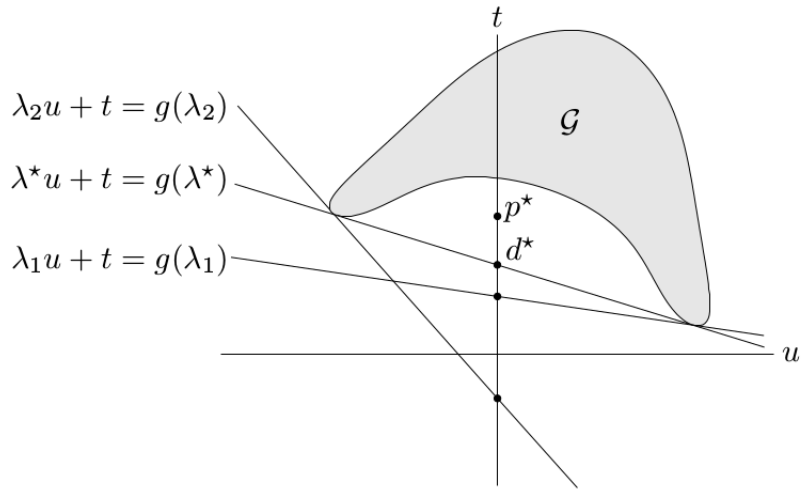
p* *refers to the optimal value of the equation system* (1)

Case I: When $g(\lambda) \leq p*$, for a problem with one inequality constraint



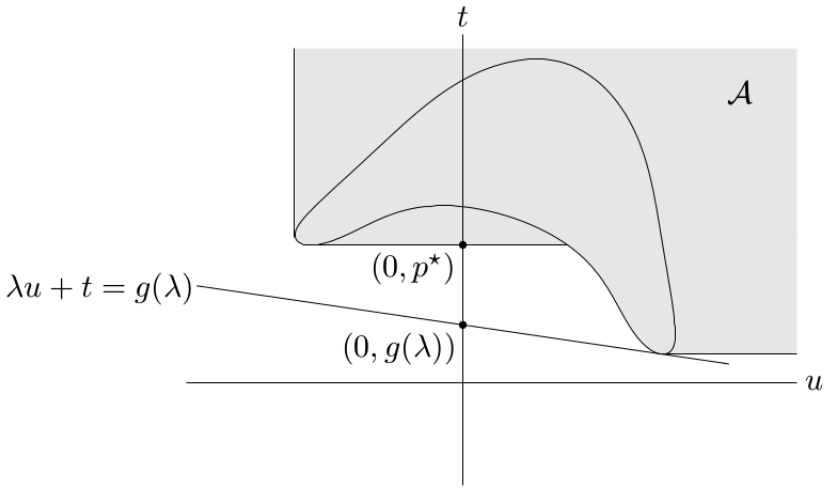
Here, for a given λ, we minimize $(\lambda, 1)^T\ (u, t)$ over $\mathcal{G} = (f_1(x), f_0(x)) \mid x \epsilon \mathcal{D}$. We can also see the resulting hyperplane which has a slope λ. The intersection of this hyperplane with the u = 0 axis gives $g(\lambda)$.

Case II: Hyperplanes associated with the three λ feasible values, including the optimum $\lambda*$



The optimal duality gap $p* - g*$ is non-negative. This is a case of Weak Duality, as talked about in section 3.1.

Case III: Again when $g(\lambda) \leq p*$, for a problem with one inequality constraint (we minimize over $\mathcal{A}$ instead of $\mathcal{G}$)



Now, for a given λ, we minimize $(\lambda, 1)^T\ (u, t)$ over $\mathcal{A} = (u, t)|x\epsilon\mathcal{D}, f_0(x) \leq t, f_1(x) \leq u$. We can also see the resulting hyperplane which has a slope λ. The intersection of this hyperplane with the u = 0 axis gives $g(\lambda)$.
Overall, the dual program does end up finding the largest y-intercept in each case.

3.4 Consequences of being Strong Dual

Let's delve a bit deeper into what it means to be strongly dual and examine its consequences. We can start be assuming that strong duality holds

$$
\begin{aligned}
& f(\boldsymbol{w}^*) = g(\lambda^*, \nu^*) \\
= \; & \min_{\boldsymbol{w} \in \mathcal{W}} \left(f_0(\boldsymbol{w}) + \sum_{i=1}^{C_1} \lambda_i^* f_i(\boldsymbol{w}) + \sum_{i=1}^{C_2} \nu_i^* h_i(\boldsymbol{w}) \right) \qquad (7) \\
= \; & f_0(\boldsymbol{w}^*) + \sum_{i=1}^{C_1} \lambda_i^* f_i(\boldsymbol{w}^*) + \sum_{i=1}^{C_2} \nu_i^* h_i(\boldsymbol{w}^*)
\end{aligned}
$$

We can see here that upon assuming a Strong Dual, we automatically only consider the optimal λ_i^* and ν_i^* values. Upon removing the min operation we also consider the optimal solution to equation (1), w^*.

Computing this further, we can cut off $f_0(\mathbf{w}^*)$ from LHS and RHS. Additionally even $\sum_{i=1}^{C_2} \nu_i^* h_i(\boldsymbol{w}^*)$ is equal to 0. This results in: $\lambda_i^* f_i(\mathbf{w}*)=0$, $\forall i \; \epsilon \{1, ..., C_1\}$

An interesting condition arises here, when:

- $\lambda_i^* \geq 0$, implying that $f_i(\mathbf{w}*) = 0$
- $f_i(\mathbf{w}*) \geq 0$, implying that $\lambda_i^* = 0$

We call this *complementary slackness*! This holds true for any primal optimal $\mathbf{w}^*$ and any dual optimal (λ^*, ν^*), when Strong Duality is true.

3.5 Karush-Kuhn-Tucker (KKT) Optimality Conditions

We will, lastly, cover another interesting case when the functions $f_1, ..., f_{C_1}$ and $h_1, ..., h_{C_2}$ have open domains, or in other words, are differentiable. The following set of conditions hold true in such a scenario:

- Primal feasibility: $f_i(\mathbf{w}) \leq 0$, $\forall i \; \epsilon \; 1, ..., C_1$, and $h_i(\mathbf{w}) = 0$, $\forall i \; \epsilon \; 1, ..., C_2$
- Dual feasibility: $lambda_i \geq 0$, $\forall i \; \epsilon \; 1, ..., C_1$
- Complementary slackness: $\lambda_i f_i(\mathbf{w}) = 0$, $\forall i \; \epsilon \; 1, ..., C_1$

- Lagrangian is stationary, i.e. its derivative is 0

$$\nabla f_0(\boldsymbol{w}) + \sum_{i=1}^{C_1} \lambda_i \nabla f_i(\boldsymbol{w}) + \sum_{i=1}^{C_2} \nu_i \nabla h_i(\boldsymbol{w}) = 0$$

Important Takeaways:

1. If Strong Duality is true and $\mathbf{w}, \lambda, \nu$ are optimal, then the KKT conditions must be satisfied!
2. Conversely, if $\mathbf{w}, \lambda, \nu$ satisfy KKT conditions for convex problems, then they are optimal!

4 Quiz Solutions

($\mathcal{Q}$) What to do before computing the Lagrangian?
($\mathcal{A}$) Bring primal program into standard form!

($\mathcal{Q}$) How to obtain the dual program?
($\mathcal{A}$) Follow the recipe provided in Section **2.3**.

($\mathcal{Q}$) Why duality?
($\mathcal{A}$) Duality, more often than not, makes it easier to solve by involving less constraints and making it simpler to optimize.

References

Book
Pattern Recognition and Machine Learning (Information Science and Statistics)
Springer-Verlag Berlin, Heidelberg ©2006
ISBN:0387310738

Book
Convex Optimization
Cambridge University Press New York, NY, USA ©2004
ISBN:0521833787