A brief introduction to inference using Lagrangian relaxation

Vivek Srikumar

9 April, 2012

1 Introduction

This note briefly summarizes the use of Lagrangian relaxation for inference. Suppose we want to find the solution to the following problem

\[
\max_x f(x) \quad (1)
\]

\[
\text{st.} \quad x \in X, \quad (2)
\]

\[
e_i^T x = b_i; \quad \forall i = 1 \cdots n \quad (3)
\]

Further, suppose that solving the following maximization problem \( P' \) is computationally easier:

\[
\max_{x \in X} f(x) \quad (4)
\]

Lagrangian relaxation is a technique that allows us to use the computationally easier \( P' \) as a sub-routine to solve \( P \).

2 The algorithm

Let \( \lambda_i \) be the Langrange multipliers corresponding to the constraints 3. Then, the Lagrangian is

\[
L(\lambda, x) = f(x) + \sum_{i=1}^{n} \lambda_i (e_i^T x - b_i) \quad (5)
\]

This gives us the following dual objective for the problem \( P \):

\[
\max_{x \in X} L(\lambda, x) \equiv \Theta(\lambda) \quad (6)
\]

This objective function is a convex function in \( \lambda \). Note that some of the constraints \( (x \in X) \) are not moved into the Lagrangian, while the “difficult” constraints are associated with dual variables. The dual objective gives us an
optimization problem equivalent to $P$. Denote the following dual problem as $D$:

$$\min_{\lambda \in \mathbb{R}^n} \Theta(\lambda)$$  \hspace{1cm} (7)

To solve the problem $P$, we solve the dual $D$. Suppose $f(x)$ is linear in $x$, denoted by $a^T x$. So, we have

$$L(\lambda, x) = a^T x + \sum_{i=1}^{n} \lambda_i \left( c_i^T x - b_i \right)$$  \hspace{1cm} (8)

$$= \left( a^T + \sum_{i} \lambda_i c_i^T \right) x - \sum_{i} \lambda_i b_i$$  \hspace{1cm} (9)

$$\equiv \hat{a}^T x - \hat{b}$$  \hspace{1cm} (10)

Here, we use the notation $\hat{a}$ to denote $a + \sum \lambda_i c_i$.

The dual objective $\Theta$ is defined as the maximum of the Lagrangian over all $x \in X$. That is, we have

$$\Theta(\lambda) = \max_{x \in X} L(\lambda, x)$$  \hspace{1cm} (11)

$$= \max_{x \in X} \hat{a}^T x - \hat{b}$$  \hspace{1cm} (12)

$$= \max_{x \in X} \hat{a}^T x$$  \hspace{1cm} (13)

The last step has the same functional form as the problem $P'$, which is computationally efficient to solve. Thus, we can compute the dual objective efficiently.

To solve the actual dual $D$, we employ sub-gradient descent over $\Theta$. The partial derivative of the function $\Theta(\lambda)$ with respect to $\lambda_i$ is given by

$$\frac{\partial \Theta}{\partial \lambda_i} = c_i^T x^\star - b_i$$  \hspace{1cm} (14)

where, $x^\star = \arg \max_{x \in X} \hat{a}_i^T x$  \hspace{1cm} (15)

Using this gradient, we can define the algorithm (Algorithm 1) that optimizes the problem $P$ using the solver for $P'$ as a sub-routine.

In this algorithm, the $\lambda$'s are updated only if at least one constraint is violated. Otherwise, the algorithm has found the optimal solution for $P$ and returns the value.

**Notes:**

1. It is possible that even after $T$ iterations of the gradient descent, the solution has not yet been found. In such a case, can something be said about the quality of the solution?
Algorithm 1 Using Lagrangian relaxation for solving a “hard” inference problem $P$ using the solver for an “easier” problem $P'$ as a sub-routine

1: $\lambda^{(0)} \leftarrow 0$
2: for $t = 1 \cdots T$ do
3: $x^* \leftarrow \arg \max_{x \in X} \left( a + \sum \lambda_i c_i \right)^T x$
4: if $x^*$ satisfies all the $n$ constraints then
5: return $x^*$
6: else
7: for $i = 1 \cdots n$ do
8: $\lambda_i^{(t)} \leftarrow \lambda_i^{(t-1)} - \alpha^{(t)} (c_i^T x^* - b_i)$
9: end for
10: end if
11: end for

2. The overall problem $P$ asks for the solution in the space that is the intersection of the feasible space for $P'$ and the additional constraints. The problem $P'$ could hide the “integer” constraints over the inference variables. That is, the solution to $P'$ could be a dynamic program such as maximum flow.

3. Even though this note considers equality constraints of the form $c_i^T x = b_i$, it is easy to extend these to inequality constraints. Doing so will introduce additional box constraints in the dual which can be dealt with in the gradient descent using a projection step.

References


