Problem Set: ADMM

1

Performing simple method of multipliers.

1.1

Say we are interested in optimizing the following constrained problem:

Minimize: $f(x)$ under constraint: $Ax = b$

The Lagrangian for this problem can be written as:
$L(x, \lambda) = f(x) + \lambda(Ax - b)$

The optimal solution can therefore be found by iterating through x values, such that:

$x_{k+1} = argmin_x L(x, \lambda_k)$

$\lambda_{k+1} = \lambda_k + \alpha_k(Ax_{k+1} - b)$

1.1.1 Write an expression for the method of multipliers (augmented Lagrangian) that uses a penalty ρ to form a regularization term.

1.1.2 Using this new expression, what are we trying to minimize and what is the constraint?

1.1.3 Finally, what are the expressions for the iterations x_{k+1} and λ_{k+1} that can be used to optimize this method of multipliers (augmented Lagrangian) problem?

1.2

How does alternating direction method of multipliers (ADMM) differ from the typical method of multipliers?

2

Describing the advantages and disadvantages of ADMM and versions of ADMM.

2.1

What is an advantage of decentralized ADMM, when compared to centralized?

2.2

What is a disadvantage of decentralized ADMM, when compared to centralized?

2.3

What is an advantage of asynchronous ADMM, when compared to synchronous?

2.4

What is an advantage of distributed algorithms?

3

Implementing ADMM for use in the consensus of distributed systems.

3.1

What is the meaning of "consensus" in regards to problems of distributed systems?

3.2

Suppose we would like to minimize $f(x) = \sum_{i=1} N f_i(x)$, in which our objective function is divided into N parts $f_i(x), i = 1, ..., N$. Also suppose that we implement a global consensus constraint, so that all of the local x_i should be equal. Our problem then becomes:

Minimize $f(x) = \sum_{i=1} N f_i(x)$ under constraint $x_i - z = 0, i = 1, ..., N$

3.2.1 Develop an expression for the augmented Lagrangian $L(x_1, ..., x_n, \lambda)$ for this problem.

3.2.2 What are the resulting ADMM algorithmic expressions for the iterations x_{k+1}, z_{k+1}, and λ_{k+1} that can be used to optimize this problem?

3.2.3 How would the ADMM algorithmic expression for the z_{k+1} iterations change if we wanted to use the median of the x_i^{k+1} components, rather than the mean of the x_i^{k+1}, to calculate z_{k+1}?