

**A GRADUATE COURSE  
IN  
OPTIMIZATION**



---

**A GRADUATE COURSE**  
**IN**  
**OPTIMIZATION**  
**CIE6010 Notebook**

---

**Prof. Yin Zhang**

*The Chinese University of Hong Kong, Shenzhen*



香港中文大學(深圳)

The Chinese University of Hong Kong, Shenzhen



# Contents

Acknowledgments	xi
Notations	xiii
<b>1 Week1</b>	<b>1</b>
1.1 Monday	1
1.1.1 Introduction to Optimizaiton	1
1.2 Wednesday	2
1.2.1 Reviewing for Linear Algebra	2
1.2.2 Reviewing for Calculus	2
1.2.3 Introduction to Optimization	3
<b>2 Week2</b>	<b>7</b>
2.1 Monday	7
2.1.1 Reviewing and Announments	7
2.1.2 Quadratic Function Case Study	8
2.2 Wednesday	11
2.2.1 Convex Analysis	11
<b>3 Week3</b>	<b>17</b>
3.1 Wednesday	17
3.1.1 Convex Analysis	17
3.1.2 Iterative Method	18
3.2 Thursday	22
3.2.1 Announcement	22
3.2.2 Sparse Large Scale Optimization	22

<b>4</b>	<b>Week4</b>	<b>27</b>
<b>4.1</b>	<b>Wednesday</b>	<b>27</b>
4.1.1	Comments for MATLAB Project	27
4.1.2	Local Convergence Rate	28
4.1.3	Newton's Method	29
4.1.4	Tutorial: Introduction to Convexity	30
<b>5</b>	<b>Week5</b>	<b>33</b>
<b>5.1</b>	<b>Monday</b>	<b>33</b>
5.1.1	Review	33
5.1.2	Existence of solution to Quadratic Programming	36
<b>5.2</b>	<b>Wednesday</b>	<b>39</b>
5.2.1	Comments about Newton's Method	39
5.2.2	Constant Step-Size Analysis	40
<b>6</b>	<b>Week6</b>	<b>45</b>
<b>6.1</b>	<b>Monday</b>	<b>45</b>
6.1.1	Announcement	45
6.1.2	Introduction to Quasi-Newton Method	45
6.1.3	Constrained Optimization Problem	46
6.1.4	Announcement on Assignment	47
6.1.5	Introduction to Stochastic optimization	49
<b>6.2</b>	<b>Tutorial: Monday</b>	<b>49</b>
6.2.1	LP Problem	49
6.2.2	Gauss-Newton Method	50
6.2.3	Introduction to KKT and CQ	51
<b>6.3</b>	<b>Wednesday</b>	<b>52</b>
6.3.1	Review	52
6.3.2	Dual-Primal of LP	53

<b>7</b>	<b>Week7</b>	<b>57</b>
<b>7.1</b>	<b>Monday</b>	<b>57</b>
7.1.1	Announcement	57
7.1.2	Recap about linear programming	57
7.1.3	Optimization over convex set	60
<b>7.2</b>	<b>Wednesday</b>	<b>62</b>
7.2.1	Motivation	62
7.2.2	Convex Projections	63
7.2.3	Feasible direction method	65
<b>8</b>	<b>Week8</b>	<b>69</b>
<b>8.1</b>	<b>Monday</b>	<b>69</b>
8.1.1	Constraint optimization	70
8.1.2	Inequality Constraint Problem	71
<b>8.2</b>	<b>Monday Tutorial: Review for CIE6010</b>	<b>71</b>
<b>9</b>	<b>Week9</b>	<b>79</b>
<b>9.1</b>	<b>Monday</b>	<b>79</b>
9.1.1	Reviewing for KKT	79
<b>9.2</b>	<b>Monday Tutorial: Reviewing for Mid-term</b>	<b>82</b>
<b>10</b>	<b>Week10</b>	<b>83</b>
<b>10.1</b>	<b>Monday</b>	<b>83</b>
10.1.1	Duality Theory	83
10.1.2	Penalty Algorithms	86
<b>10.2</b>	<b>Wednesday</b>	<b>89</b>
10.2.1	Introduction to penalty algorithms	89
10.2.2	Convergence Analysis	90

<b>11</b>	<b>Week11</b>	<b>93</b>
11.1	Monday	93
11.1.1	Equality Constraint Problem	93
11.1.2	ADMM	96
11.2	Wednesday	97
11.2.1	Comments on Assignment 6	97
11.2.2	Inequality Constraint Problem	98
11.2.3	Non-smooth unconstraint problem	99
<b>12</b>	<b>Week12</b>	<b>101</b>
12.1	Monday	101
12.1.1	Comments on Final Project	101
12.1.2	Trust Region Method	102
12.2	Monday Tutorial	104
12.2.1	Sub-gradient	104
12.3	Wednesday	107
12.3.1	Trust Region problem	107
12.4	Monday Tutorial: Trust Region Sub-problem	109
12.4.1	ADMM	109
12.4.2	Trust Region Subproblem	110
<b>13</b>	<b>Week13</b>	<b>111</b>
13.1	Wednesday	111
13.1.1	Approximate Gradient Projection	111
13.1.2	Conic Programming	112
<b>14</b>	<b>Week14</b>	<b>113</b>
14.1	Monday	113



<b>15</b>	<b>Week14</b> .....	<b>117</b>
<b>15.1</b>	<b>Wednesday</b>	<b>117</b>
15.1.1	Conic Programming .....	117
15.1.2	Algorithm to solve conic programming .....	119



# Acknowledgments

This book is from the CIE6010 in fall semester, 2018.

CUHK(SZ)



# Notations and Conventions

$X$	Set
$\inf X \subseteq \mathbb{R}$	Infimum over the set $X$
$\mathbb{R}^{m \times n}$	set of all $m \times n$ real-valued matrices
$\mathbb{C}^{m \times n}$	set of all $m \times n$ complex-valued matrices
$x_i$	$i$ th entry of column vector $\mathbf{x}$
$a_{ij}$	$(i, j)$ th entry of matrix $\mathbf{A}$
$\mathbf{a}_i$	$i$ th column of matrix $\mathbf{A}$
$\mathbf{a}_i^T$	$i$ th row of matrix $\mathbf{A}$
$\mathbb{S}^n$	set of all $n \times n$ real symmetric matrices, i.e., $\mathbf{A} \in \mathbb{R}^{n \times n}$ and $a_{ij} = a_{ji}$ for all $i, j$
$\mathbb{H}^n$	set of all $n \times n$ complex Hermitian matrices, i.e., $\mathbf{A} \in \mathbb{C}^{n \times n}$ and $\bar{a}_{ij} = a_{ji}$ for all $i, j$
$\mathbf{A}^T$	transpose of $\mathbf{A}$ , i.e, $\mathbf{B} = \mathbf{A}^T$ means $b_{ji} = a_{ij}$ for all $i, j$
$\mathbf{A}^H$	Hermitian transpose of $\mathbf{A}$ , i.e, $\mathbf{B} = \mathbf{A}^H$ means $b_{ji} = \bar{a}_{ij}$ for all $i, j$
$\text{trace}(\mathbf{A})$	sum of diagonal entries of square matrix $\mathbf{A}$
$\mathbf{1}$	A vector with all 1 entries
$\mathbf{0}$	either a vector of all zeros, or a matrix of all zeros
$\mathbf{e}_i$	a unit vector with the nonzero element at the $i$ th entry
$\mathcal{C}(\mathbf{A})$	the column space of $\mathbf{A}$
$\mathcal{R}(\mathbf{A})$	the row space of $\mathbf{A}$
$\mathcal{N}(\mathbf{A})$	the null space of $\mathbf{A}$
$\text{Proj}_{\mathcal{M}}(\mathbf{A})$	the projection of $\mathbf{A}$ onto the set $\mathcal{M}$



# Chapter 15

## Week14

### 15.1. Wednesday

#### 15.1.1. Conic Programming

The primal conic programming is given by:

$$\begin{aligned} \min \quad & \langle \mathbf{C}, \mathbf{X} \rangle \\ & \langle \mathbf{a}_i, \mathbf{X} \rangle = \mathbf{b}_i, i = 1, \dots, m \\ & \mathbf{X} \in \mathcal{K} \end{aligned}$$

LP, SDP, SOCP.

The dual form is given by:

$$\begin{aligned} \max \quad & \mathbf{b}^T \mathbf{y} \\ & \sum_{i=1}^m y_i \mathbf{a}_i + \mathbf{S} = \mathbf{C} \\ & \mathbf{S} \in \mathcal{K}^* \end{aligned}$$

Most problem setting is self-dual.

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{b} \\ \mathbf{A}^* \mathbf{y} + \mathbf{S} &= \mathbf{C} \end{aligned}$$

Fermat-Weber location problem: Given a set of points  $\mathbf{p}_i$ , our goal is to

$$\min_{\mathbf{y} \in \mathbb{R}^2} \sum_{i=1}^m \|\mathbf{y} - \mathbf{p}_i\|$$

Note that it is norm 2 instead of its square. It's relatively complicated problem.

Introduce variables  $\eta_1, \dots, \eta_m$ :

$$\begin{aligned} \min \quad & \eta_1 + \dots + \eta_m \\ & \|\mathbf{y} - \mathbf{p}_i\| \leq \eta_i, \quad i = 1, \dots, m \end{aligned}$$

Or equivalently,

$$\begin{aligned} \min_{\mathbf{y}, \boldsymbol{\eta}, \mathbf{z}} \quad & \mathbf{1}^T \boldsymbol{\eta} \\ & \mathbf{z}_i + \mathbf{y} = \mathbf{p}_i, \quad i = 1, \dots, m \\ & \|\mathbf{z}_i\| \leq \eta_i \end{aligned}$$

The dual problem is

$$\begin{aligned} \max \quad & \sum_{i=1}^m \mathbf{p}_i^T \mathbf{x}_i \\ & \sum_{i=1}^m \mathbf{x}_i = \mathbf{0} \\ & \|\mathbf{x}_i\| \leq 1, i = 1, \dots, m \end{aligned}$$

the last constraint is the second order cone.

For quadratic constraint with  $\mathbf{A} \succeq 0$ :

$$(\mathbf{A}\mathbf{y} + \mathbf{b})^T(\mathbf{A}\mathbf{y} + \mathbf{b}) - \mathbf{c}^T\mathbf{y} - \mathbf{d} \leq 0$$

which is equivalent to say

$$\begin{bmatrix} \mathbf{I} & \mathbf{A}\mathbf{y} + \mathbf{b} \\ (\mathbf{A}\mathbf{y} + \mathbf{b})^T & \mathbf{c}^T\mathbf{y} + \mathbf{d} \end{bmatrix} \succeq 0$$

QCQP can be converted into SDP when  $\mathbf{A}$  is convex.

$$\left(\mathbf{c}^T\mathbf{y} + \mathbf{d} - \frac{1}{4}\right)^2 + \|\mathbf{A}\mathbf{y} + \mathbf{b}\|^2 \leq \left(\mathbf{c}^T\mathbf{y} + \mathbf{d} + \frac{1}{4}\right)^2$$

Thus QCQP can be converted into SOCP as well.



### 15.1.2. Algorithm to solve conic programming

$$\begin{aligned} \min \quad & \langle \mathbf{C}, \mathbf{X} \rangle \\ & \langle \mathbf{A}_i, \mathbf{X} \rangle = \mathbf{b}_i, i = 1, \dots, m \\ & \mathbf{X} \succeq 0 \end{aligned}$$

$$\begin{aligned} \max \quad & \mathbf{b}^T \mathbf{y} \\ & \sum y_i \mathbf{A}_i + \mathbf{Z} = \mathbf{C} \\ & \mathbf{Z} \succeq 0 \end{aligned}$$

If both (P) and (D) are **strictly feasible**, then there exists  $\mathbf{X}^*, \mathbf{y}^*$  (feasible) such that

$$\langle \mathbf{C}, \mathbf{X}^* \rangle = \mathbf{b}^T \mathbf{y}^*$$

which follows that

$$\langle \mathbf{C}, \mathbf{X} \rangle - \mathbf{b}^T \mathbf{y} = \langle \mathbf{X}, \mathbf{Z} \rangle = 0$$

It suffices to let

$$0 = \text{trace}(\mathbf{XZ}) = \sum_i \lambda_i(\mathbf{Z}^{1/2} \mathbf{XZ}^{1/2})$$

which implies

$$\mathbf{Z}^{1/2} \mathbf{XZ}^{1/2} = 0 \iff \mathbf{Z}^{1/2} \mathbf{X}^{1/2} = 0$$

