

# Mathematic Modeling the Outline

## – 数学建模知识点大纲 –

$$e^{i\theta} = \cos(\theta) + i \sin(\theta)$$

Lumin

cdluminare@163.com

This Document is distributed under the MIT license.

August 22, 2015

## Contents

<b>1</b>	<b>Optimization Model</b>	<b>2</b>
1.1	Lineaer Programming	2
1.1.1	Standard Form	2
1.1.2	Standarlization Methods	2
1.1.3	Solving	2
1.1.4	Application	2
1.2	Integer Linear Programming	2
1.3	Mixed Integer Programming	2
1.4	Dynamic Programming	2
1.4.1	Basic Concept	2
1.4.2	Optimization Theory	2
1.4.3	逆序法, 正序法	2
1.4.4	Applications	2
1.5	Non-Linear Programming	2
1.5.1	Optimal problem without constraint	2
1.5.2	Optimal problem with constraints	3
1.5.3	Application	3
1.6	Software/Tools	3
<b>2</b>	<b>Dynamics Model</b>	<b>4</b>
2.1	Differential Function Model	4
2.1.1	Exponential Model	4
2.1.2	SI Model: susceptible infective : Logistic	4
2.1.3	SIS Model	4
2.1.4	SIR Model	4
2.2	Stability	4
2.3	Application	4
2.4	Software	4
<b>3</b>	<b>Possibility</b>	<b>5</b>
3.1	Probability Base	5
3.2	Computer Emulating – Monte Carlo	5
3.2.1	Monte Carlo Precision	5
3.3	Random Number Generator	5

3.3.1	Probability – Reference	5
<b>4</b>	<b>Graph Theory</b>	<b>6</b>
4.1	Basic concept of graph theory	6
4.2	一些定理	6
4.3	Shortest Path Problem	6
4.3.1	Dijkstra Algorithm	6
4.3.2	Floyd Algorithm	6
4.4	Application	6
4.5	Euler Graph and Hamilton Cycle	6
4.5.1	Traveling Salesman Problem	6
4.6	Trees and spanning trees	7
4.6.1	无向生成树	7
4.6.2	最优连线问题	7
4.6.3	最大流问题	7
4.7	Python3	7
4.8	Matlab	7
<b>5</b>	<b>Statistics and Regression</b>	<b>8</b>
5.1	聚类分析	8
5.2	回归分析	8
5.2.1	主要内容	8
5.2.2	Single Variable Linear Regression	8
5.2.3	Multi Variable Linear Regression	8
5.2.4	Multi Variable Non-Linear Regression	8
5.2.5	keys	8
<b>6</b>	<b>Markov Chain</b>	<b>8</b>
6.1	placeholder	8
<b>7</b>	<b>Fuzzy Mathematics</b>	<b>8</b>
7.1	Fuzzy Subset	8
7.1.1	$\lambda$ Cut Set	8
7.1.2	Fuzzy Relationship	8
7.1.3	Fuzzy Relation Synthesis	8
7.1.4	Fuzzy Clustering	8
<b>8</b>	<b>排队论</b>	<b>9</b>
<b>A</b>	<b>Paper composing</b>	<b>10</b>
A.1	Reference Structure	10
<b>B</b>	<b>Common References</b>	<b>11</b>

# 1 Optimization Model

## 1.1 Lineaer Programming

### 1.1.1 Standard Form

The object is

$$\text{Minimize } Z = C^T X$$

Subject to

$$Ax = b$$

Bounds

$$x \geq 0$$

Where

$$C = \begin{bmatrix} c_1 \\ c_2 \\ \dots \\ c_n \end{bmatrix}, x = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix}, A = \begin{bmatrix} a_{11} & \dots \\ \dots & a_{nn} \end{bmatrix}, b = \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{bmatrix}$$

### 1.1.2 Standarlization Methods

- 最大最小转换:  $Z' = -Z$
- 不等式约束变为等式约束: 不等式中添加松弛变量/剩余变量  $x_{n+i}$ 。
- $b_i \geq 0, (i = 1, 2, \dots, n)$ , 乘-1反转公式以标准化。
- $X_i$  无限制时, 引入非负变量  $m, n$ , 令  $x_i = m - n$  并代入目标方程, 转化为非负限制。

### 1.1.3 Solving

图解法。单纯形法 (需求标准式或矩阵式问题)。

### 1.1.4 Application

各类资源规划, 投资, 生产/存储控制, 下料。

## 1.2 Integer Linear Programming

分支界定法 Branch&Cut

## 1.3 Mixed Integer Programming

## 1.4 Dynamic Programming

### 1.4.1 Basic Concept

1. Step  $k = 1 \dots n$
2. Status  $S_k$

3. Decision  $u_k(x_k)$

4. Policy

$$P_{kj} = u_k(x_k), \dots, u_j(x_j)$$

5. Status Transition Equation

$$x_{k+1} = T_k(x_k, u_k(x_k)), k = 1, \dots, n$$

6. Objective function

$$V_{k,n}(x_k, p_{k,n}(x_k)) = \Phi_k(x_k, u_k, V_{k+1,n})$$

7. Step Profit

$$V_{k,n}(x_k, p_{k,n}(x_k)) = \text{opt}_{k \leq j \leq n} V_j(x_j, u_j)$$

8. Optimal Value Function

$$f_k(x_k) = \text{opt} V_{k,n}(x_k, p_{k,n}(x_k))$$

### 1.4.2 Optimization Theory

○ Bellman Optimization Theory

### 1.4.3 逆序法, 正序法

### 1.4.4 Applications

When put Dynamic Programming into practice, pay attention to 1. definition of steps; 2. policies allowed within each step; 3. status transition equation.

- Shortest path problem

- Knapsack Problem

- Resource Assignment

- ...

## 1.5 Non-Linear Programming

### 1.5.1 Optimal problem without constraint

General form

$$\text{Min } f(x), x = (x_1, x_2, \dots, x_n)^T \in R^n$$

Local solution optimized  $\rightarrow$  一阶必要

$$\nabla f(x^*) = 0$$

Local solution optimized  $\rightarrow$  二阶充分

$$\nabla f(x^*) = 0 \text{ and } \nabla^2 f(x^*) \text{ is Positive definite}$$

## 1.5.2 Optimal problem with constraints

General form

$$\text{Min } f(x), x \in R$$

$$\text{s.t. } c_i(x) = 0, i \in E = \{1, 2, \dots, l\}$$

$$c_i(x) = 0, i \in I = \{l+1, l+2, \dots, l+m\}$$

Local solution optimized  $\rightarrow$  必要

$$\exists \text{vector } \lambda^*$$

$$\nabla_x L(x^*, \lambda^*) = \nabla f(x^*) + \sum_{i=1}^{l+m} \lambda_i^* \nabla c_i(x^*) = 0$$

$$c_i(x^*) = 0, i \in E$$

$$c_i(x^*) \leq 0, i \in I$$

$$\lambda_i^* \geq 0, i \in I$$

$$\lambda_i^* c_i(x^*) = 0, i \in I$$

Where the  $L$  is the Lagrange function

$$L(x, \lambda) = f(x) + \sum_{i=1}^{l+m} \lambda_i c_i(x)$$

## 1.5.3 Application

订购/存储模型, 投资/组合问题

lingo例子

e.g.

$$\min f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$$

sets:

var/1..2/: x;

endsets

[OBJ] min = x(1)^2 + x(2);

[C1] x(1)^2 + x(2)^2 <= 9;

[C2] x(1) + x(2) <= 1;

@for(var: @free(x));

## 1.6 Software/Tools

Matlab/Octave, Lingo/Lindo, IBM ILOG CPLEX.

## 2 Dynamics Model

### 2.1 Differential Function Model

#### 2.1.1 Exponential Model

$$x(t + \Delta t) - x(t) = kx(t)\Delta t$$

when  $\Delta x \rightarrow 0$

$$\begin{cases} \frac{dx}{dt} = kx \\ x(0) = x_0 \end{cases}$$

the solution

$$x(t) = x_0 e^{kt}$$

#### 2.1.2 SI Model: susceptible infective : Logistic

$$s(t) + i(t) = N$$

$$k(s) = \frac{ks}{N} = k(1 - \frac{i}{n})$$

$$\begin{cases} \frac{di}{dt} = k(1 - \frac{i}{n})s \\ x(0) = x_0 \end{cases}$$

the solution

$$x(t) = \frac{n}{1 + (\frac{n}{x_0})e^{-kt} - 1}$$

#### 2.1.3 SIS Model

$$\begin{cases} \frac{dx}{dt} = k(1 - \frac{x}{n})x - lx \\ x(0) = x_0 \end{cases}$$

#### 2.1.4 SIR Model

$$\begin{cases} \frac{dx}{dt} = \frac{kxs}{n} - lx \\ \frac{ds}{dt} = -\frac{kxs}{n} \\ \frac{dr}{dt} = lx \\ x(0) = x_0, r(0) = r_0, s(0) = s_0 \end{cases}$$

### 2.2 Stability

定理: 设  $x_0$  是微分方程  $dx(t)/dt = f(x)$  的平衡点且  $f'(x_0) \neq 0$ , 若  $f'(x_0) \leq 0$  则  $x_0$  稳定; 若  $f'(x_0) \geq 0$ , 则  $x_0$  不稳定。

### 2.3 Application

Best fishing strategy.

### 2.4 Software

Matlab:

dsolve, solver = { ode45, ode... }

## 3 Possibility

### 3.1 Probability Base

Include [1]

### 3.2 Computer Emulating - Monte Carlo

Statistical simulation method: Combine the emulation of random events with the probability feature of different kinds of random events.

e.g. Solve  $I = \int_a^b g(x)dx$  with Monte Carlo

sol. Cast uniformly randomized points to the rectangle range, then count how many point dropped under the curve  $g(x)$ .

1. Build the probability model; 2. extract samples from known probability distributions; 3. Setup statistic variables in need.

#### 3.2.1 Monte Carlo Precision

CLT: Central-Limit Theorem

1. Randomly Casting Points

$$E(X) \approx \bar{p} = \frac{k}{n}$$

2. Mean Value

$$\bar{x} = \frac{1}{n}(x_1 + x_2 + \dots + x_n) = \frac{1}{n} \sum_{i=1}^n x_i$$

### 3.3 Random Number Generator

1.  $x_i = \lambda x_{i-1}(\text{mod} M)$ .

2.  $x_i = (\lambda x_i + c)(\text{mod} M)$ .

Tests:

1. Parameter test
2. Uniformity test
3. Independency test

In matlab:

rand\*, normrand

#### 3.3.1 Probability - Reference

## References

[1] 概率论与数理统计，浙江大学出版社

## 4 Graph Theory

### 4.1 Basic concept of graph theory

- 矩阵  
邻接矩阵 node - node  
关联矩阵 node - edge  
赋权矩阵 weight
- 各种类型的图  
完全图，立方体（偶图），完全偶图，赋权图，有向图，无向图  
子图

$$V' \subset V, E' \subset E$$

生成子图

$$V' = V, E' \subset E$$

补图

- 树  
树，生成树，无向生成树

### 4.2 一些定理

1. Given simple graph  $G$  contains no isolated node, and contains  $m$  edges, then the number of all generated subgraph is  $2^m$

$$\binom{m}{0} + \binom{m}{1} + \dots + \binom{m}{m} = 2^m$$

2. For  $\forall$  undirected graph

$$\sum_{v \in V} d(v) = 2m$$

3. For  $\forall$  directed graph

$$\sum_{v \in V} d^+(v) + \sum_{v \in V} d^-(v) = m$$

### 4.3 Shortest Path Problem

#### 4.3.1 Dijkstra Algorithm

Base of this algorithm

$$d(u_0, \bar{S}) = \min_{u \in S, v \in \bar{S}} \{d(u_0, u) + w(u, v)\}$$

where  $u, v$  denotes the source and destination node,  $d(u, v)$  denotes the distance between  $u$  and  $v$ ,  $S \in V$  and  $u_0 \in S$  and  $\bar{S} = V \setminus S$ .

This algorithm can not only find the shortest path from  $u_0$  to  $v_0$ , but also all the shortest path from  $u_0$  to any other nodes in graph  $G$ .

#### 4.3.2 Floyd Algorithm

Get the minimum distance between to given nodes.

### 4.4 Application

1. 运输问题
2. 转运问题
3. 最优指派问题: 匈牙利算法
4. 中国邮递员问题: Fleury Algorithm

### 4.5 Euler Graph and Hamilton Cycle

#### 4.5.1 Traveling Salesman Problem

Figure out the Hamilton cycle which possess the minimum weight.

Assume that:

- $w_{ij}$  denotes the distance between city  $i$  to city  $j$
- $x_{ij}$  denotes the decision if going from city  $i$

The solution is:

$$\text{Minimize } C = \sum_{i=1}^n \sum_{j=1}^n w_{ij} x_{ij} \quad (1)$$

s.t. :

$$\sum_{i=1}^n x_{ij} = 1, j = 1, 2, \dots, n$$

$$\sum_{j=1}^n x_{ij} = 1, i = 1, 2, \dots, n$$

$$u_i - u_j + n x_{ij} \leq n - 1; i \neq j; i, j = 2, 3, \dots, n$$

$$x_{ij} = 0 \text{ OR } 1; i \neq j; i, j = 1, 2, \dots, n$$

$$u_j \geq 0; j = 1, 2, \dots, n$$

Note that, there are some TSP example program available in the LINGO's user manual[1].

MODEL:

! traveling seller;

SETS:

city/Pe T Pa M N L/: u;

link(city, city): w,x;

endsets

data:

!to Pe T Pa M N L;

w = 0 13 51 77 68 50

13 0 60 70 67 59

```

51 60 0 57 36 2
77 70 57 0 20 55
68 67 36 20 0 34
50 59 2 55 34 0;
enddata

```

```

n = @size(city);
min = @sum(link: w * x);
@for(city(k):
@sum(city(i)|i #ne# k: x(i,k)) = 1;
@sum(city(j)|j #ne# k: x(k,j)) = 1;
);

```

```

@for(link(i,j)|i #gt# 1 #and# j #gt# 1 #and#
u(i) - u(j) + n*x(i,j) <= n-1;
);

```

```

@for(link: @bin(x));
end

```

## 4.6 Trees and spinning trees

### 4.6.1 无向生成树

避圈法

### 4.6.2 最优连线问题

Kruskal

### 4.6.3 最大流问题

## 4.7 Python3

python3-networkx  
1 #ne# j:

## 4.8 Matlab

graph\* function set:  
graphminspantree  
graphshortpath



## 5 Statistics and Regression

### 5.1 聚类分析

拟合, 分段

K-means: 原则上需要预先知道类别数量。

### 5.2 回归分析

#### 5.2.1 主要内容

- 相关关系, 数学表达式
- 回归方程, 回归预测
- 估计的标准误差

#### 5.2.2 Single Variable Linear Regression

$$Y = a_0 + a_1X + \varepsilon$$

where

$$\varepsilon_i \sim N(0, \sigma^2)$$

Object:

$$Q = \sum (y - y_c)^2$$

Get the parameters: Least square Method

#### 5.2.3 Multi Variable Linear Regression

TODO: svm :: svr ?

$$Y = a_0 + a_1X_1 + a_2X_2 + \dots + a_nX_n + \varepsilon$$

#### 5.2.4 Multi Variable Non-Linear Regression

$$Y = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

#### 5.2.5 keys

- 选择主成分
- 相关系数
- 置信区间

## 6 Markov Chain

### 6.1 placeholder

## 7 Fuzzy Mathematics

### 7.1 Fuzzy Subset

While the feature function of certain subset can be represented as the mapping

$$X_A : U \rightarrow \{0, 1\}$$

where  $X_A(x) = 1$  when  $x \in A$ .

秃头悖论。

The fuzzy subset  $A \in U$  can be represented as the mapping

$$A(x) : U \rightarrow [0, 1]$$

#### 7.1.1 $\lambda$ Cut Set

$$A_\lambda = \{x | A(x) \geq \lambda\}$$

#### 7.1.2 Fuzzy Relationship

While the classical 2 variable relationship can be represented as the mapping

$$R : X \times Y \rightarrow \{0, 1\}$$

which is in fact the feature function of subset  $R$  of  $X \times Y$ .

The fuzzy relationship can be represented as the mapping

$$R : X \times Y \rightarrow [0, 1]$$

The fuzzy matrix means

$$R_{m \times n}(i, j) = R(x_i, y_j)$$

#### 7.1.3 Fuzzy Relation Synthesis

$$R_1 \circ R_2 = (c_{ij})_{m \times n}$$

where

$$c_{ij} = \bigvee \{(a_{ik} \wedge b_{kj}) | 1 \leq k \leq s\}$$

#### 7.1.4 Fuzzy Clustering

- Data Standarlizing
  1. 平移: 标准差变换, 极差变换
- Fuzzy Likeness Matrix
  1. Cosine of included angle

2. Correlation coefficient

- Distances  $r_{ij} = 1 - cd(x_i, x_j)$

1. Hamming Distance

2. Euclidean Distance

3. Chebyshev Distance

- Fuzzy equivalent matrix

## 8 排队论

## A Paper composing

1. Background of problem
2. Assumptions
3. Setup the Mathematical Model
4. Get the solution of model
5. Model Analyzation

6. Model Validation

### A.1 Reference Structure

摘要 问题重述与分析 问题假设 符号说明 模型建立与求解 结果分析 模型检验 模型推广 模型评价 参考文献和附录

## B Common References

### References

- [1] LINGO 官方文档 <http://www.lindo.com/downloads/PDF/LINGO.pdf>
- [2] 数学建模基础（第二版），薛毅，科学出版社
- [3] 本文的Git Repo，以及我写的一些数模相关程序
- [4] Octave PDF Document <http://www.gnu.org/software/octave/octave.pdf>