1. Lab：3、5、6、8、9

Final agent submission ： 5 January 2021 5pm

Report ： 12 January 2021 5pm

⇓

计划，比较花时间

2. ANAC ： The Automated Negotiating Agents Competition

3、protocol

domain

4. Design ：

① Concession strategy

② offer producting strategy

③ oppnent model

④ preference uncertainty

⑤ preference elicitation

5、4 guided labs.

6、上传的全是自己做的，虽然可以用别人的 idea.

7. Agent performance (15%)

平均分是 65（总分）

8. 上网站，可以下载PPT和实验资料等

9. java 8. genius.

Week 3

1、 Approaches for conflict resolution between agents

① agents 个 冲突 when 有不同的目的表现

② multi-agent 系统 个 when agents are 自私的 (self-interested) 不同的 stakeholders (利益相关者)

解决方法: ① Auctions 竞拍 allocate scarce 分配稀有

② voting

③ Negotiation

I、protocol = rules of encounter

通常是 bilateral (双边) 但有些也适用于 multi-party

II. 特点

1). Negotiation environment

2). Agent preferences

3)、Agent negotiation strategies

III、典型的例子.

1). The Ultimatum Game 最后通牒游戏

每轮双方都立一个 offer

若双方都接受对方 则 其中一个 就 采用

若不接受则进入下一轮，下一轮至少有一方退让(concede)

否则 结束交易

2. 实用

Utility function，$U(o)$，$o \in O$

顺序的

Ordinal preferences:

eg. $U(o_1) > U(o_2)$

主要

cardinal perferences:

eg. $U(o_1) = 0.78$ $U(o_2) = 0.5$

(1) 例子：Buyer / Seller - Price Negotiation

I The cardinal utility functions are given by:

$U_{seller}(p) = p - r$

r: minimum price (seller)

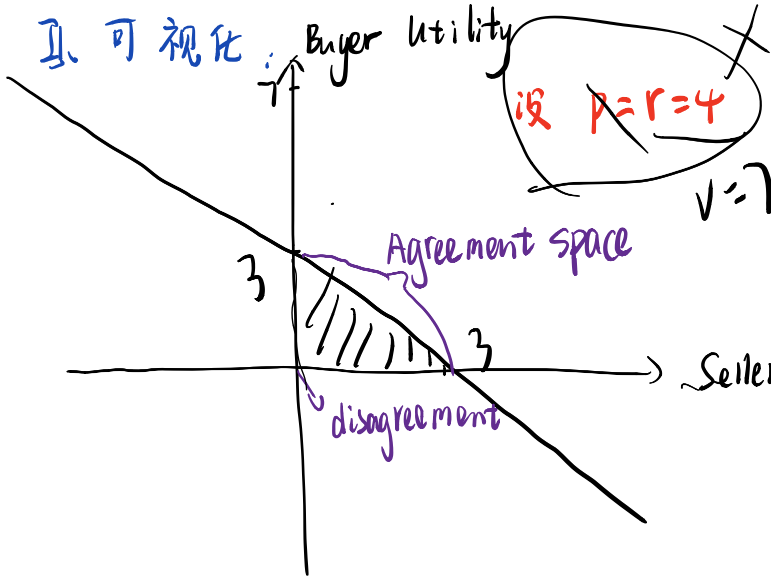
$U_{seller}(disagreement) = 0$

v: maximum willing pay (buyer)

$U_{buyer}(p) = v - p$

$U_{buyer}(disagreement) = 0$

I. 可视化：



$U_{seller}(p) = p-4$

$U_{seller}(disagreement)=0$

$U_{buyer}(p) = 7-p$

$U_{buyer}(disagreement)=0$

Agreement space: $p \in [4,7]$

II. Time pressure

例如: Oddl.

② Break-off probability (协议决定或自行中断)

③ Bargaining costs 议价成本

I. Fixed costs

例如 1) c_i (代理 i 的花费)

$$u_i^t = u_i - t\cdot c_i$$

2) 冰融化：等的越久，融化的越快

$\delta_i<1$ 是贬损因素

$$u_i^t = u_i \cdot \delta_i^t$$

IV. multi - Issue Negotiation

(outcome) O_j $\Leftrightarrow$ each issue j.

$$U_i(o) = \sum_j w_{i,j} \cdot U_{i,j}(o_j)$$

例如： agent 1, 2 分蛋糕, O_j.

agent 1 $\Rightarrow$ cake $j \in \{1, 2\}$

∴ agent 2 $\Rightarrow$ $(1 - O_j)$

∴ $\begin{cases} U_{1,j}(O_j) = O_j \\ U_{2,j}(O_j) = 1 - O_j \end{cases}$ $(j \in \{1, 2\})$

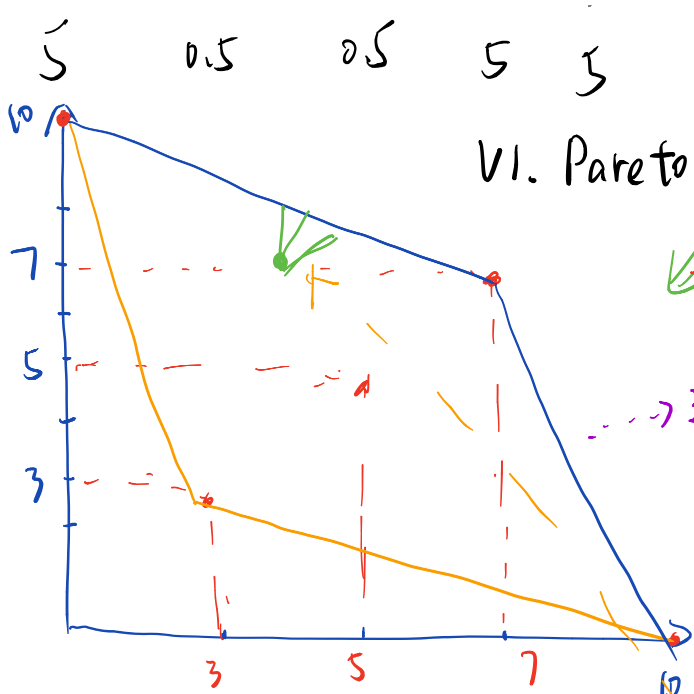
设 agent 1 and agent 2 have weights 7, 3 and 3, 7 分别

得： $U_1(o) = 7 * O_1 + 3 * O_2$

$U_2(o) = 3 * (1 - O_1) + 7 * (1 - O_2)$

V. 练习

offer	O_1	O_2	U_1	U_2
1	1	1	10	0
2	0	0	0	10
3	1	0	7	7
4	0	1	3	3



VI. Pareto - Efficient Agreement

若是在已知的offer的情况下

则无需划线,

寻找已知的是不是真更优解

边界

Parote 有效更像是上帝视角,同时调整2个都能更好

(对比,NASH 均衡 只调整一方)

VII. Desirable properties

aggreement:

① $U(o) > U(dis)$

② Pareto effient

③ fair

VIII. fairness

多种定义:

① Utilitarian social welfare:

$\max_{o \in O} U_1(o) + U_2(o)$ [取max的和]

② Egalitarian social welfare:

$\max_{o \in O} \min_{i \in \{1,2\}} U_i(o)$ [最大化最小值]

③ Nash bargaining solution：

$$\max_{O \in O} \left(U_1(O) - U_1(\text{disagreement})\right) \cdot \left(U_2(O) - U_2(\text{disagreement})\right)$$

1). Individual rationally ： $U_1 \geq U_1(dis)$ & $U_2 \geq U_2(dis)$

2). Pareto efficiency

3) Invariance to equivalent utility representations
等价实用程序表示的不变性

(4) IIA 无关选项独立性

5) Sym 对称：agent 具有相同的表现时，程序相同

④ Envy-freeness：设 agent 对 resource 感兴趣

1). 只在资源分配问题下才有意义

2) 例子： O_j 是 Pie j 给 1

$1-O_j$ 给 2

Agent 1 会嫉妒 if ： $U_1(1-O_1, 1-O_2) > U(O_1, O_2)$

同理 Agent 2 …… if ： $U_1(O_1, O_2) > U_2(1-O_1, 1-O_2)$

X. 练习（例子）

Example 1 Revisited

$$u_1(o) = 7*o_1 + 3*o_2$$

$$u_2(o) = 3*(1-o_1) + 7*(1-o_2)$$

如果u_1'或u_2'比u_1和u_2，对应的一方就会envy

完成表格：

2个中最小的 ↑ 2个相乘 ↑ 交换o_1, o_2

Offer	o_1	o_2	u_1	u_2	Utilitrian Sum	Egalitarian min	NBS Product	u_1'	u_2'	Envious?
1	1	1	10	0	10	0	0	0	10	agent2 ×
2	0	0	0	10	10	0	0	10	0	agent1 ×
3	1	0	7	7	14	7	49	3	3	✓
4	6	1	3	3	6	3	9	7	7	互相嫉妒
5	0.5	0.5	5	5	10	5	25	5	5	✓

Envy Free (Agent 2 is getting what Agent 1 is getting

反之亦然，

Vice versa)

另外，要想不envy-free，需要1和2得到的相同

3.(part 3)

(1) Negotiation Strategies、Approaches

I. Game theoretic

① 游戏规则、所有玩家的偏好和信念都是常识

② 所有参与者完全理性.

③ preferences encoded in a (limited) set of player types

④ 封闭的系统、预定互动、小型游戏

equilibrium

⑤ Nash 均衡

Heuristic perspective

I、启发式的角度来看

① 不需要常识或完美的理性假设

② 代理行为是直接建模的

③ 适用于开放的、动态的环境.

④ 可能性的空间非常大

Heuristic 用在对 opponent 有未知

例: strategy they are using

* 将 strategy 分为 2个部分:

让步策略

I、consession strategy = { Time-dependent tactics (只取决于time) ; Tit-for-tat (behaviour) }

多个issue

II muti-issue offer producting strategy

Time-dependent tactics:

$$U_{target}(t) = U_{min} + (1-F(t))(U_{max} - U_{min})$$

($F(x) \in [0,1]$) 表示最好与最坏 offer 之间的距离;

$$F(t) = \left(\frac{\min(t, T_{max})}{T_{max}}\right)^{\frac{1}{\beta}}$$

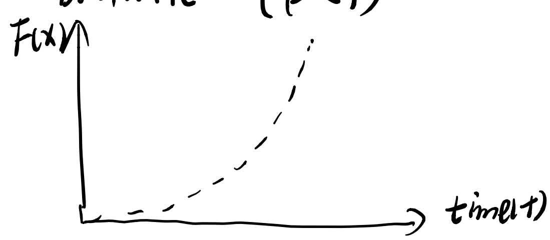
$[\ F(t) = 0 \quad F(T_{max}) = 1\ ]$

例： ① Hard-headed ($\beta \to 0$)　没有让步，坚持最初价

希望对手让步

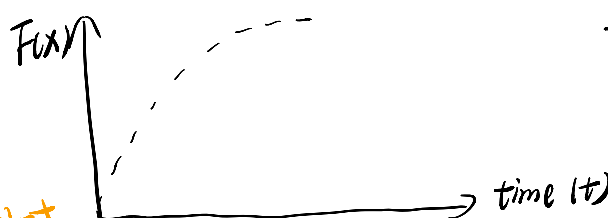
② Linear time-dependent concession ($\beta = 1$)

③ Boulware ($\beta < 1$)



让步缓慢.
最初报价维持到ddl

④ Conceder ($\beta > 1$)



让步迅速

针锋相对

Tit-for-tat concession strategy

① agent 根据自身效用函数的增加来检测对手在前一轮谈判中做出的让步.

② 下一轮的让步取决于对手上一轮让步

配对配的让步

$$\text{Concession} \le U_{own}(O^{t}_{opponent}) - U_{own}(O^{t-1}_{opponent})$$

③ 只要报价在可接受的范围内

(高于卖方最初价，低于买方保留价)

Optimal Concession strategy.

① 一切已知，可在对手的 utility 和 strategy 下计算出最优让步。(依赖对手和时间压力)

② 若对手策略未知，通过机器学习等，模拟对手的 model.

③ 同样，对手也如此，则用博弈论方法

(对对手进行推理的方法.

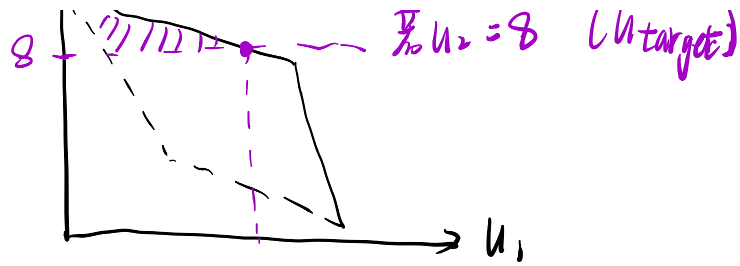
Offer - producing Strategy

① 只在提供 a target utility 下使用，为 multi-issue 的每个issue 生成个值.

② 确保报价总是 Pareto efficiency

③ 若 Utility 函数已知，则可计算给定的 target utility 的 Pareto 的有效报价.

U_2↑



Unknown Opponent Utility

① 不知对手的 utility. (private information)

② 从已知的表现(让步和以前的offer)计算.

例：对手很可能先从他们最不喜欢的问题上让步. 在附加效用函数的情况下

Preference Uncertainty and Elicitation

① 自己的 utility 函数也未知.

(代理的代表人不知)

② utility 函数通过偏好诱导过程得到

(最小化认知成本和最大化 utility 之间的权衡)

Week 4

1. Basic Setting:

Let $S=\{s_1,\dots,s_m\}$ (agent 可选的 choice)

Let $\Omega=\{w_1\dots w_t\}$ outcomes.

2. An outcome 函数:

$$g: S\to\Omega$$

具体的一个 choice

分为 certainty or uncertainty

知道选择后的结果 不知道选择后的结果

例: a binary relation $\succsim$ on $\Omega\times\Omega$

Reflexive: $w'\succsim w$ for 所有 $w\in\Omega$

Total: for all $w, w'\in\Omega$, $w\succsim w'$ 或 $w'\succsim w$

Transitive: for all $w, w', w''\in\Omega$

if $w\succsim w'$ 及 $w'\succsim w''$

$\therefore$ $w\succsim w''$

给出一对 outcomes w, w'

$$w\succ w'$$

等同于 an agent prefer w at least as much as w'

if $w \succsim w'$ and $w' \succsim w$

∴ $w \sim w'$

if $w \succsim w'$ but not $w' \succsim w$

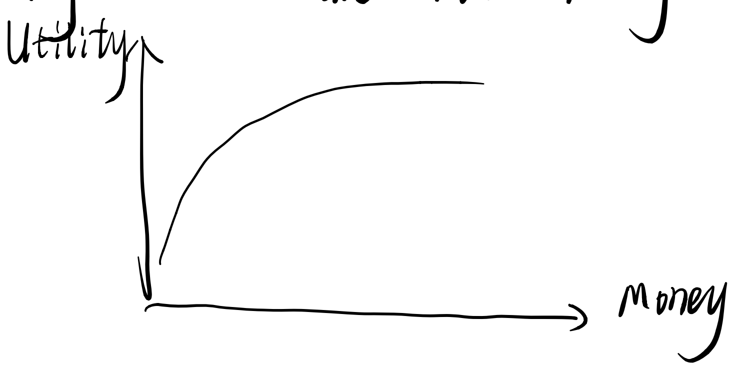
∴ $w \succ w'$

utility 函数:

$$u \to \Omega \to R \qquad (w' \in \Omega)$$

$$w \succsim w' \text{ iff } u(w) \geq u(w')$$

3. Utility values are not money



4. Formal model of decision

$\langle S, \Omega, a, \succsim, u \rangle$ ⇒ utility function choice outcome relation

Rational Decision-MAKER: (最好的 choice)

(也就是 maximise their utility)

例: $s \in \underset{s \in S}{\operatorname{argmax}}\ u(g(s))$.

5. Lottery: 用 L 表示

outcomes Ω is a Ω 的概率分布

例: 设 $\Omega = \{\text{chocolate}, \text{vanilla}, \text{strawberry}\}$

μ 是 Ω 的概率分布.

若 $\mu(\text{chocolate}) = 0.2$ $\mu(v) = 0.5$ $\mu(s) = 0.3$

$\therefore L = [0.2(\text{chocolate}), 0.5(v\dots), 0.3(s\dots)]$

Compound Lotteries: a lottery of lotteries

例 $\mathcal{L} = \{L_1, \dots, L_n\}$

$L^* = [q_1(L_1), \cdots q_n(L_n)]$ 其中 $q_i = \mu(L_i)$

复合

例: 选择去上学的方式取决于天气

设 Ω 为上学花的时间

L_{rain} 和 L_{sun} 分别代表具体上学时间

定义 lottery：

$$L^* = [p L_{rain},\ (1-p) L_{sun}]$$

这其中的 L^* 则是个 compound Lottery，描述上学时间的不确定性

6. Expected Utility

当 utility 函数 $u: \Omega \to \mathbb{R}$

$$EU(\overset{\text{Lottery}}{L}) = \sum_{w \in \Omega} u(w)\,\mu(w)$$

从概率分布的期望值　　　概率分布

需要满足：

① Axiom (Continuity) 公理.

每个 Lotteries $L_1 \succ L_2 \succ L_3$ 且 $\alpha, \beta \in [0,1]$

有 $[\alpha L_1,\ (1-\alpha) L_3] \succ L_2 \succ [\beta L_1,\ (1-\beta) L_3]$

② Axiom (Independence)

所有的 Lotteries， L_1 L_2 L_3 $\alpha \in [0,1]$

$L_1 \succsim L_2$ IFF $[\alpha L_1, (1-\alpha) L_3] \succsim [\alpha L_2, (1-\alpha) L_3]$

7、Theorem：

① $\succsim$ 满足 continuity 和 independence

② utility function $\Rightarrow \Omega$

$L_1 \succsim L_2$ IFF $EU(L_1) \succsim EU(L_2)$

有且仅当

Week 4 ~ part 2. Strategic-form GAMES (pure strategies)

1、每个角色独立做决定

例：

Prisoner 2

		C	B
prisoner 1	C	1y, 1y	6y, 0y
	B	0y, 6y	3y, 3y

4个可能： $\Omega = \{0y, 1y, 3y, 6y\}$

具体的：设如下为 outcomes：

Prisoner 2

		C	B
prisoner1	C	8, 8	0, 10
	B	0	5

10 | 5 |

$u_i(0y)=10$ $u_i(1y)=8$ $u_i(3y)=5$ $u_i(6y)=0$

2. 引出 strategic-form game

$\langle N, S_1, \dots S_n, u_1, \dots u_n \rangle$

其中. $N=\{1,\dots,n\}$ 第n个 player

S_i → i的 strategy

$u_i: S_1\times\dots\times S_n \to R$ ⇒ play i 的 utility function

S_{-i} 表示除 S_i 以外的合集

当无论另个人选择什么，i 选 B 都更好时:

$u_i(s_i, s_{-i}) < u_i(s_i', s_{-i})$

前提是游戏者是理性的.

所以最后的决定会是 B, B.
(outcome)

3. 另一个 example (并不是所有的都能像上一个例子一样有一个 outcome)

player 2

play 1	C	D
A	1, 2	2, 3
B	2, 2	2, 0

1选A、B都可

$u_i(s_i, t_{-i}) < u_i(s_i', t_{-i})$

4. 再一个example.

player 2

player 1	D	E	F
×A	1, 2	2, 3	0, 3
√B	2, 2	2, 1	3, 2
C	2, 1	0, 0	1, 0

可以得出一个outcome

(先去A，则为BD)

也可以先去掉C，则outcome为BF.

所以有outcome时，也可有多个outcome

NASH EQUILIBRIA

5. 又一个 example

player 2

player 1	D	E	F
A	0, 6	6, 0	4, 3
B	6, 0	0, 6	4, 3
C	3, 3	3, 3	5, 5

① 不能排出消减顺序了

② 用 NASH Equilibrium

best response to S_{-i} if:

$$u_i(s_i, S_{-i}) = \max_{s_i' \in S_i} u_i(s_i', s_{-i})$$

例: Play2 选 D, 则对 play1 来说 (S_i)

play 2

play 1	D	E	F
A	0, 6	6, 0	4, 3

① CF 是双方的 best response

B	6, 0	0, 0	4, 5
C	3, 3	3, 3	5, 5

② CF 是 Nash equilibrium 的例子

For player 1:

B is a best response to D

A is a best response to E

C ... ... F

For player 2:

D ... ... A

E B

F C

NASH equilibria 的定义:

A strategy combination $(s_1, \cdots, s_n)$ is a Nash equilibria if s_i is a best response to s_{-i} for every player $i \in N$

② 或，对每个策略组合，检查是否有玩家可以通过偏离来增加他们的效用，若不能，则是 NASH 均衡

例：CF 往上或往左都是减少 不能增加。

③ NASH 均衡 might not be unique.
唯一

协调博弈

④ Coordination game 是 multiple Nash 均衡的例子

⑤ 均衡 是指在同一 strategy 下进行协调
(Equilibria)

例：

player2

		C	D
player1	A	10, 10	0, 8
	B	8, 0	7, 7

player1 选A时，2选10
B时，2选7
player2 选C 时，A1选10
D时 1选7

AC 和 BD 都是 NASH 均衡

例：

	H	T
H	1, -1	-1, 1
T	-1, 1	1, -1

没有 NASH 均衡

① 不是所有game 都有 NASH 均衡

② Matching pennies 是其中一个例子
同时掷一个硬币

相同 则1留2个

不同 则2留2个.

若要使NASH均衡始终存在 则要使用混合概念

1. 使用消减法可能会遗漏个 成叶 out come

strictly：对于某一玩家来说；所有选择都优于其它选择

1\2	C	D	E	F	G
A	5	6	7	8	9
B	4	5	6	7	8

只说player 1

weekly：策略A中至少有一个选优于其它. /其他相同

1\2	C	D	E	F	G
A	5	6	7	8	9
B	5	6	7	8	8

Week 4 Mixed NASH Equilibria

- σ_i 是 player i 的概率密度分布，strategies $S_i \to [0,1]$

$$\sum_{s_i \in S_i} \sigma_i(s_i) = 1$$

① 随机化

② maximise outcome.

③ 将 expected utilities 理想化

例：

Player 2

player1	H	T
H	1, -1	-1, 1
T	-1, 1	1, -1

当 $\sigma_1(H) = 0.5$
$\sigma_2(H) = 0.5$ 时
有且仅有一个
mixed NASH equilibrium

$\sigma_1(H) = 0.4$ $\sigma_1(T) = 0.6$

$\sigma_2(H) = 0.3$ $\sigma_2(T) = 0.7$

$$\therefore EU_1(\sigma_1, \sigma_2) = \sigma_1(H)\sigma_2(H)\, u_1(H,H) + \sigma_1(H)\sigma_2(T)\, u_1(H,T) + \sigma_1(T)\sigma_2(H)\, u_1(T,H) + \sigma_1(T)\sigma_2(T)\, u_1(T,T)$$

expected utility

$$= (0.4 \times 0.3 \times 1) + [(0.4 \times 0.7) \times (-1)] + (0.6 \times 0.3 \times (-1)) + (0.6 \times 0.7 \times 1)$$

$= 0.08$

同理，player 2 一样的算法

设 Σ_i 为 S_i 所有 mixed strategies 的集合.

$$\Sigma_i = \{ \sigma_i \mid \sigma_i : S_i \to [0,1], \sum_{s_i \in S_i} \sigma_i(s_i) = 1 \}$$

设 $\Sigma = \Sigma_1 \times \cdots \times \Sigma_n$

$(\sigma_1 \cdots \sigma_n) \in \Sigma$

有：

$$EU_i(\sigma_1, \cdots \sigma_n) = \sum_{(s_1, \cdots s_n) \in S_1 \times \cdots S_n} \left(\left(\prod_{j \in N} \sigma_j(s_j) \right) u_i(s_1, \cdots s_n) \right)$$

设 $G = \langle N, S_1 \cdots S_n, u_1 \cdots u_n \rangle$

mixed extension of G is the game

$$\Gamma = \langle N, \Sigma_1 \cdots \Sigma_n, U_1 \cdots U_n \rangle$$

定义：设 G：strategic-form game / Γ 是 G 的 mixed extension，σ_{-i} 是 mixed strategy vector ⇒ 所有除 i 以外的 player

一个

player i's mixed strategy σ_i 是 σ_{-i} 的 best response

当:

$$EU_i(\sigma_i, \sigma_{-i}) = \max_{\sigma_i' \in \Sigma_i} EU_i(\sigma_i', \sigma_{-i})$$

● 定义：若：σ_i is a best response to σ_{-i}

(for every player $i \in N$).

则：$(\sigma_1, \dots \sigma_n)$ 是一个 mixed strategy Nash equilibrium

1、每个 strategic-form game 都有 mixed extension ⇒ 有 mixed strategy Nash equilibria

2. Finding mixed Nash equilibria in a strategic-form game is PPAD-complete

● 计算.

1、例

player 2

		L y	R $1-y$
player1	T x	$u_2(T,L)$ $u_1(T,L)$	$u_2(T,R)$ $u_1(T,R)$

1 v	B	1-x $u_2(\cdot\cdot)$ $u_1(B,2)$	$u_2(\ )$ $u_1(\ )$

- best response 函数：

$$br_1(y) = \underset{x\in[0,1]}{\arg\max}\ EU_1(x,y)$$

$$br_2(x) = \underset{y\in[0,1]}{\arg\max}\ EU_2(x,y)$$

例：

		player 2	
		H	T
player1	H	2, 1	0, 0
	T	0, 0	1, 2

其中 (H, H) 和 (T, T) 是 NASH 均衡

计算 mixed …

$$EU_1(x,y) = 2xy + 0x(1-y) + 0(1-x)y + 1(1-x)(1-y)$$
$$= x(3y-1) - y + 1$$

对每个 y 值需找到所有 x 使得 $[x(3y-1)-y+1]$ 最大

当 $(3y-1)=0$ 时 即与 x 无关了.

同理：

$$EU_2(x,y) = y(3x-2) - 2x + 2$$

当 $(3x-2)=0$ 时，取最大值与 y 无关

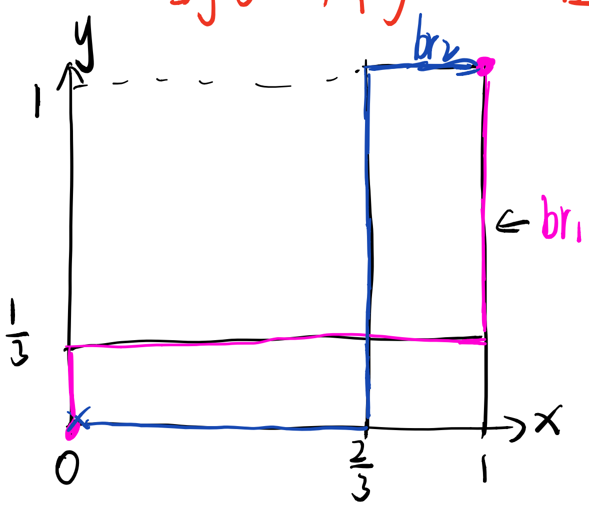
$x\downarrow$ $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ $y\downarrow$

得出 $br_1(y) = \begin{cases} 0 & y<\frac{1}{3} \\ [0,1] & y=\frac{1}{3} \\ 1 & y>\frac{1}{3} \end{cases}$ $\qquad$ $br_2(x) = \begin{cases} 0 & x<\frac{2}{3} \\ [0,1] & x=\frac{2}{3} \\ 1 & x>\frac{2}{3} \end{cases}$

代表EU最大值

当 $y=\frac{1}{3}$ 时，player1可任意选择

作图：



full mixed 不包含0,1

pure = (0,0) (1,1)

mixed：三个

- 定义：s_i 和 s_i' 是两个 pure strategy of player i

  if $\sigma_i(s_i)>0$ and $\sigma_i(s_i')>0$

  then： $EU_i(s_i,\sigma_{-i}) = EU_i(s_i',\sigma_{-i})$

- $\sigma_i(s_i)>0$ for every pure strategy $s_i\in S_i$

若对于每个 player i 来说，σ_i is completely mixed，

则 a mixed Nash equilibrium $(\sigma_1 \cdots \sigma_n)$ 是个 completely mixed Nash equilibrium

indeferent 原则：如果存在一个完全混合的纳什均衡，那么对于每个玩家，在给定的均衡下，使用任何纯策略预期效用都是相同的

- 每个玩家都不关心对另一个玩家使用 pure strategy

设 $EU_1(T, y) = U_1(T,L)y + U_1(T,R)(1-y)$

...

(当选T时的 EU) $EU_2(L, X) = U_2(T,L)X + U_2(B,L)(1-X)$

...

在 2x2 的博弈中，当且仅当

$EU_1(T,y) = EU_1(B,y)$ and $EU_2(L,x) = EU_2(R,x)$ 时

概率密度 (x,y) 完全 mixed Nash 均衡

证明：

假设 $EU_1(T,y) = EU_1(B,y)$

且 $x' \in (0,1)$ 是随机 mixed strategy.

则有：$EU_1(x,y) = xEU_1(T,y) + (1-x)EU_1(B,y)$

$= EU_1(T,y)$

$= x' EU_1(T,y) + (1-x') EU_1(T,y)$

$= x' EU_1(T,y) + (1-x') EU_1(B,y)$

$= EU_1(x',y)$

所以：首先使 $EU_1(T,y) = EU_1(B,y)$

则计算出 y

再使 $EU_2(L,x) = EU_2(R,x)$

可计算出 x

Any pair of solutions (x,y) is a mixed Nash 均衡.

例：

Player 2

player 1	L	R
T	1, -1	0, 2
B	0, 1	2, 0

没有 pure Nash 均衡；

证：$EU_1(T,y) = EU_1(B,y)$

得 $u = 2(1-u)$

$\therefore y = \frac{2}{3}$

$EU_2(L, x) = EU_2(R, x)$

得：$-x+1-x = 2x$

$\therefore x = \frac{1}{4}$

$\therefore (\frac{1}{4}, \frac{2}{3})$ 是个 completely mixed Nash equilibrium

例：

Player 2

Player 1	L y	R $1-y$
T x	4, 4	2, 8
B $1-x$	8, 2	1, 1

(T,R) 和 (B,L) 是 2个 NASH 均衡．看是否还存在其他均衡

当 $EU_1(T, y) = EU_1(B, y)$ 时，

$4y + 2(1-y) = 8y + (1-y)$

$\therefore y = \frac{1}{5}$

当 $EU_2(L, X) = EU_2(R, X)$

$4X + 2(1-X) = 8X + (1-X)$

$\therefore x = \frac{1}{5}$

$\therefore (\frac{1}{5}, \frac{2}{5})$ is a fully mixed Nash equilibrium

- 若计算出 x 或 y 不在 $[0,1]$ 的范围内，则无 completely mixed equilibria

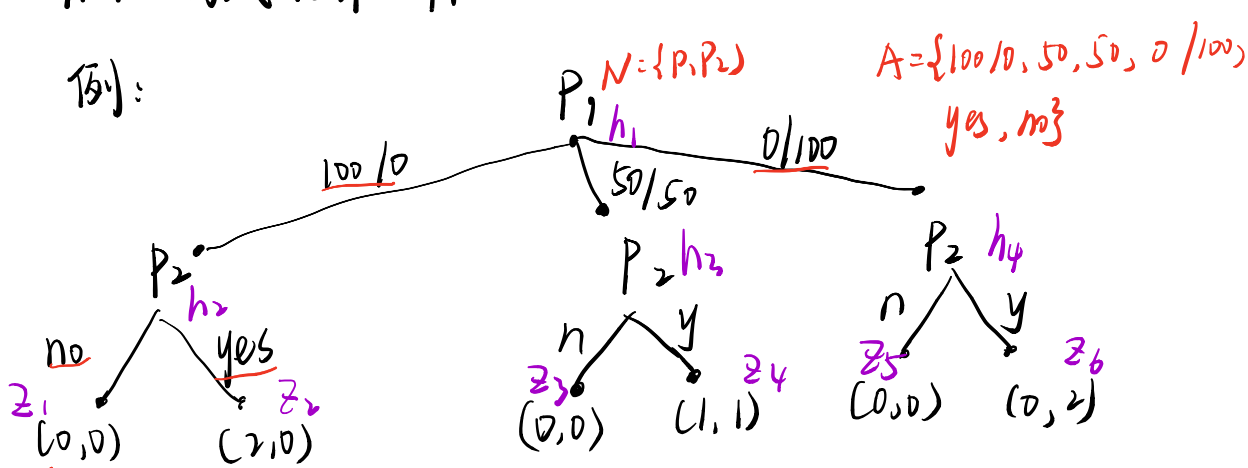
同理 可以反证一个 (x,y) 值是否是完全 NASH 混合均衡

week 5. 具有 perfect information 的游戏.

①.可以理解为机器型选手(啥都知道)

1、把游戏比作一棵 tree.

例:



① Extensive-form games

$$G=(N, A, H, Z, \alpha, \pi, \sigma, u)$$

- N ↓ 选手
- A ↓ action
- H ↓ 非终端选择节点
- Z → 终端节点,
- α ↓ $H \to 2^A$
- π → $H \to N$
- σ ↓ $H \times A \to H \cup Z$
- u ↑ utility function

函数 α : $h_1 \overset{\alpha}{\longleftrightarrow} \{100/0,\ 50/50,\ 0/100\}$ $\quad h_2 \overset{\alpha}{\longleftrightarrow} \{no, yes\}$ $\quad h_3 \cdot h_4 \cdots$

函数 π : $h_1 \overset{\pi}{\longleftrightarrow} P_1 \quad h_2 \overset{\pi}{\longleftrightarrow} P_2 \quad h_3 \overset{\pi}{\longleftrightarrow} P_2 \quad h_4 \overset{\pi}{\longleftrightarrow} P_2$

σ : $(h_1, 100/0) \overset{\sigma}{\longleftrightarrow} h_2 \quad \cdots\cdots$

$u = (u_1 \cdots u_n)$

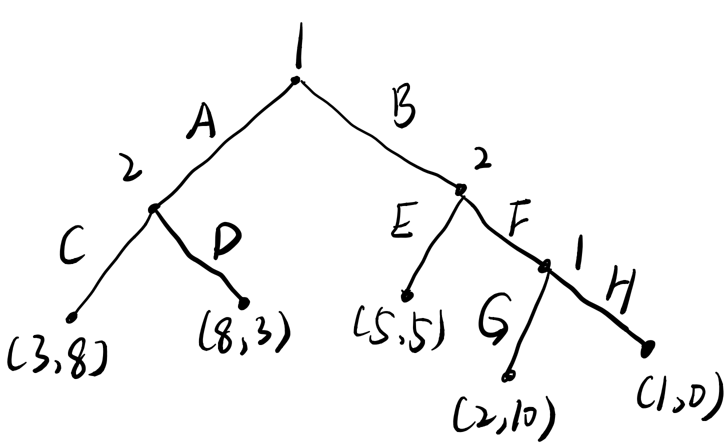
$u_1(z_1) = 0 \qquad u_2(z_1) = 2$

pure strategies of player i of the Cartesian product:

$$\prod_{h \in H, \pi(h) = i} \alpha(h)$$

② Strategic-form transformation

例:



strategies for player 1 和 2

$$S_1 = \{A, B\} \times \{G, H\}$$
$$= \{(A,G), (A,H), (B,G), (B,H)\}$$

$$S_2 = \{C, D\} \times \{E, F\}$$
$$= \{(C,E), (C,F), (D,E), (D,F)\}$$

	(C,E)	(C,F)	(D,E)	(D,F)
(A,G)	3, 8	3, 8	8, 3	8, 3
(A,H)	3, 8	3, 8	8, 3	8, 3
(B,G)	5, 5	2, 10	5, 5	2, 10
(B,H)	5, 5	1, 0	5, 5	1, 0

将 strategic-form 游戏转化为 extensive-form 游戏并不总可能

③ Equibibria

I、对于 extensive-form 游戏; pure Nash equilibria 总是存在

II、以上面的题为例: 看 tree 得来的

(A,G), (C,F)

$\Downarrow$

player1 选A后，G 和 H 中，G更优
(2) (1)

player2 中 C 优于D，E 优于F（前提是1选了G

(A, H), (C, F)

(B, H), (C, E)

⊕ 某些均衡不能令人满意：引入子游戏概念
(subgame)

例：G H 为1的一个子游戏.

同理，EF / CD / AB 也是子游戏

I. 每个 extensive-form game 都有一个 subgame 的完美均衡.

II. subgame 的完美均衡都是 pure strategy Nash 均衡的子集

例：(B,H) (C,E) 不是子游戏的 NASH 均衡.

因为1在 G H 中会选G

同理 ((A,H) (C,F)) 也不是

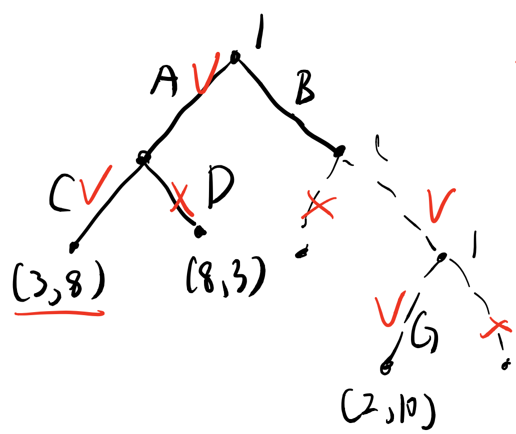
G is a NASH 均衡 ⇒ (G, F) 是个NASH均衡

同理 C is a ......

(AG) (CF) 是每个 subgame 的 NASH 均衡.
(perfect)

④ backward induction 逆向推导

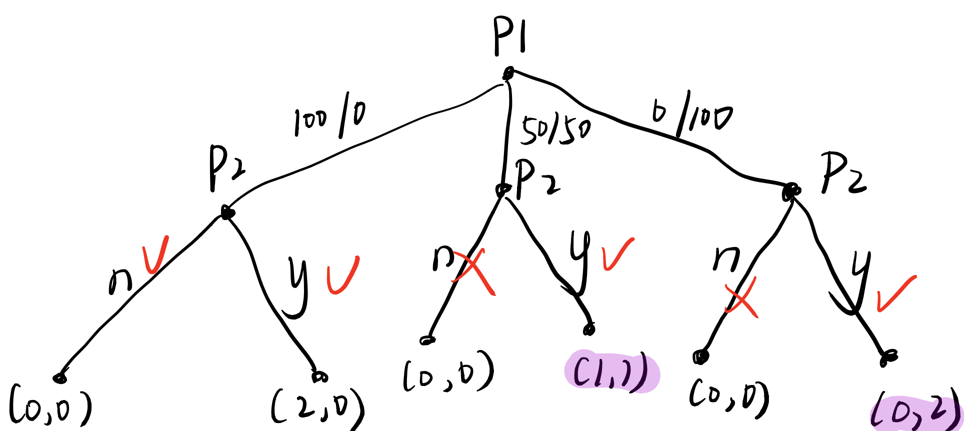
从子游戏出发，选了后去除，再次从子游戏出发
(代替)



所以最后

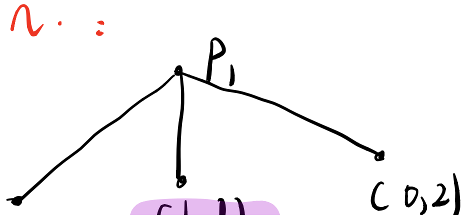
(A,C), (F,G) 为 (3,8)

例题：(回到最开始的例题)



无论选 y or n 都是0

I, 二选一：若选 n：



(0,0)

第一个NASH均衡是 (50/50, (n, y, y))

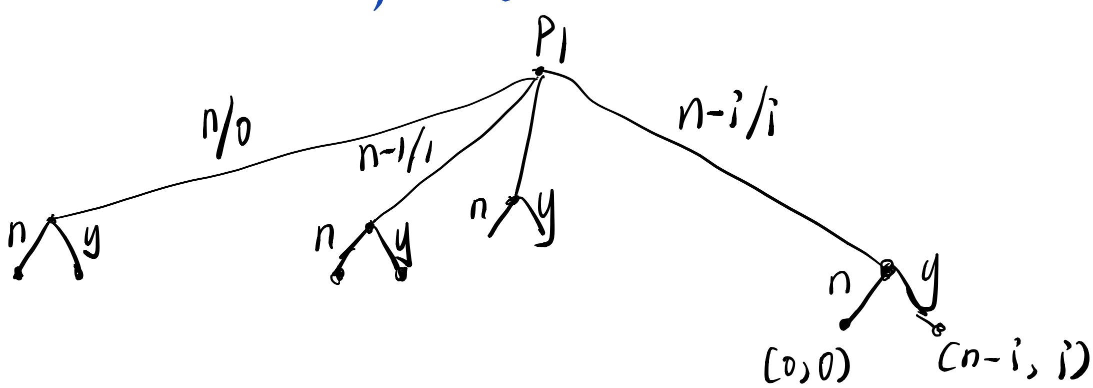
Ⅱ. 二选一; 若选 y:

第二个NASH均衡是: (100/0, (y, y, y))

例题2: 最后通牒游戏

player 1 分成n份. 每份 player2 选 yes or no

选no的话则 (0, 0)



P1 选(n/0) 那同样的无论 P2 选啥都是 0了

例题3:

100步, 奇数由 player 1 决定 $\begin{cases} \text{stop game, 得}(t, t-1) \\ \text{继续} \end{cases}$

偶数由 player 2 决定 $\begin{cases} \text{stop game, 得}(t-2, t+1) \\ \text{继续} \end{cases}$

若无人stop, 则进行到 100 步

逆向推导中，每步都会 stop，但实际会更好

W6 Linear programming

1. 不允许：$x_1+x_2-2<4$
2. 决策变量是非负的
   即 $\vec{x}\geq 0$

3、求解线性规划

① objective value
② feasible solution （满足范围的所有解）
③ 问题建模：
   列方程式.

④ Matlab 求解，Lingo，GUROBI
   老师推荐

4、用线性规划计算 NASH 均衡

1、2 玩家 zero-sum game：
   player 1 + player 在任一处相加 =0

		player 2 Se y	player 2 T $1-y$
player1	E x	3, -3	-1, 1
player1	S $1-x$	-2, 2	1, -1

纯strategy S_2 能最大化 player 2 的 u 则最小化 player 1 的 u

分 z

$\therefore u_1 = \min(3x - 2(1-x), \ -x + (1-x))$

取出能使 u_1 最大的 x 值

即: $\max z$

约束: $3x - 2(1-x) - z \geq 0$

$-x + (1-x) - z \geq 0$

$x \leq 1$

$x \geq 0$

$(\frac{3}{7}, \frac{4}{7})$ $z = \frac{1}{7}$

求得 x, z 则可得 u_1, x, $1-x$.

同理 $y \in [0,1] \Rightarrow$ player 2:

if player1 选 E, 则 $-3y + (1-y)$

选 S ： $2y-(1-y)$

$$u_2 = \min\left(-3y+(1-y),\ 2y-(1-y)\right)$$

（为 w）

同理求得 y，$\max w = u_2$ 的最大值 $\left(\frac{2}{7}, \frac{5}{7}\right)$，$w = -\frac{1}{7}$

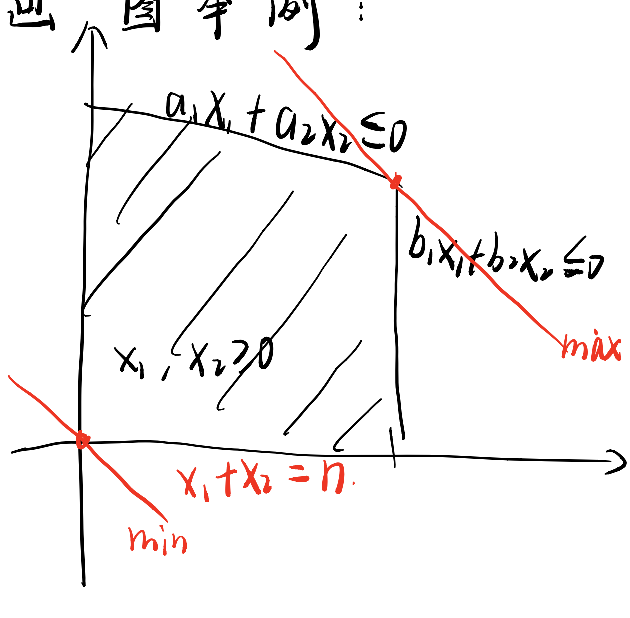
if player 2 选 $\left(\frac{2}{7}, \frac{5}{7}\right)$，$u_1 \le \frac{1}{7}$　$u_2 \ge -\frac{1}{7}$

player 1 选 $\left(\frac{3}{7}, \frac{4}{7}\right)$，$u_2 \le -\frac{1}{7}$　$u_1 \ge \frac{1}{7}$

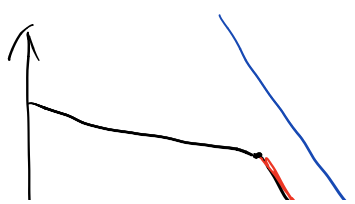
所以 $(x, y) \Rightarrow \left(\frac{2}{7}, \frac{3}{7}\right)$ 是 (fully mixed) Nash Equilibrium

5. 线性规划无解时

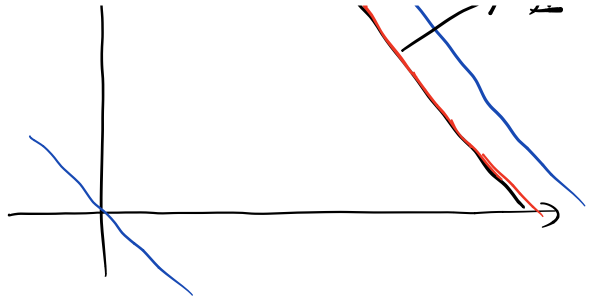
① 画图举例：



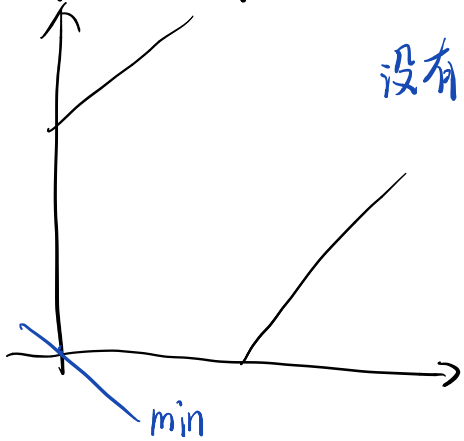
但是，若：



→ 整[redacted]点都是 u_{max}



若：约束是无界的：



6、2种标准表示

① Stand form : maximise : $\vec{c}\,\vec{x}$

线性不等式 约束 : $A\vec{x} \le \vec{b}$

$\vec{x} \ge 0$

② slack (a.k.a. normal) form :

线性等式 maximise : $\vec{c} \cdot \vec{x}$

约束 : $A\vec{x} = \vec{b}$

$\vec{x} \ge 0$

④ 不等式 + 等式 ⇒ 不等式

等式 ⇒ 不等式 $\vec{a_1} \cdot \vec{x} = c \Rightarrow \vec{a_1} \cdot \vec{x} \geq c$, $\vec{a_1} \cdot \vec{x} \leq c$

1个没有非负约束的变量 = 2个非负变量

$x_2 = x_2' - x_2''$ $(x_2', x_2'' \geq 0)$

7、整数线性规划，变量必取整

二次规划：线性约束 + 二次目标函数

工程应用

(多项式解决)

Week 7 Perference Elicitation

1、定义 $\succsim$

$y \succsim z$ ⇒ weakly prefer y to z

$y \succ z$ ⇒ strictly perfer y to z

当且仅当 $y \succsim z$ and $z \not\succsim y$

$y \sim z$ ⇒ 与 y 和 z 之间无关 等级相同

当且仅当 $y \succsim z$ and $z \succsim y$ y, z 互相想要

① Reflexive :

$x \succsim x$ for all $x \in O$

Transitive :

若 $x, y, z \in O \Rightarrow$ if $x \succsim y$ and $y \succsim z$

有 $x \succsim z$

Total :

$y \succsim z$ or $z \succsim y$ for all y and $z \in O$

若满足，则称为 total preorder

例：若 $x \succsim y$ and $y \succ z \Rightarrow x \succ z$

若 $x \succ y$ and $y \succsim z \Rightarrow x \succ z$

2. 给定 n个 结果，需要多少成对的查询才能知道结果

① 最坏的情况 $n\log n$

② 最好的情况 $n-1$ 次，恰好都按顺序排

3. 面试：

Bob $\succ$ Camille $\succ$ Diana $\sim$ Edward $\succ$ Alex

给定n个人，在没排好序时，需要多少场面试

① 最坏情况：

② 最好情况：n.

画成比"2"更耗时，来两两比较

2.3 两种方法随机选择，有时结合使用.

4、喜好不一定完全确定，也会出现概率

$P(x_1 > x_2 > x_3) = 0.3$

$P(x_2 > x_1 > x_3) = 0.2$ ……

Part 2

1、MCDA

outcomes	Education g_1	Work experience g_2	Interview Assessment g_3
Alice	100	10	70
Grace	95	12	85
Nicole	75	15	85

$U(o) = \sum_{i=1}^{n} u_i(g_i(o))$　　表示：$a > b \Leftrightarrow U(a) > U(b)$

① 有权重时：

$u_i = w_i\, g_i(c)$

例 $w_1 = 1$ - $w_2 = 1.2$ $w_3 = 1.5$

$u(\text{Alice}) = 100 + 1.2 \times 10 + 1.5 \times 70$?

2. UTA：通过线性规划的有序回归

① c is at least as good as d

$c \succsim d$

② c is prefered to d

$[c \succsim d$ but not $(d \succsim c)] \Leftrightarrow c \succ d$

③ c is indifferent to d

$c \sim d$

④ Piece-wise Linear marginal value Functions

均分 = $\dfrac{\beta_i - \alpha_i}{r_i}$ $r_i = 4$

x_i^0 x_i^1 x_i^2 x_i^3 x_i^4

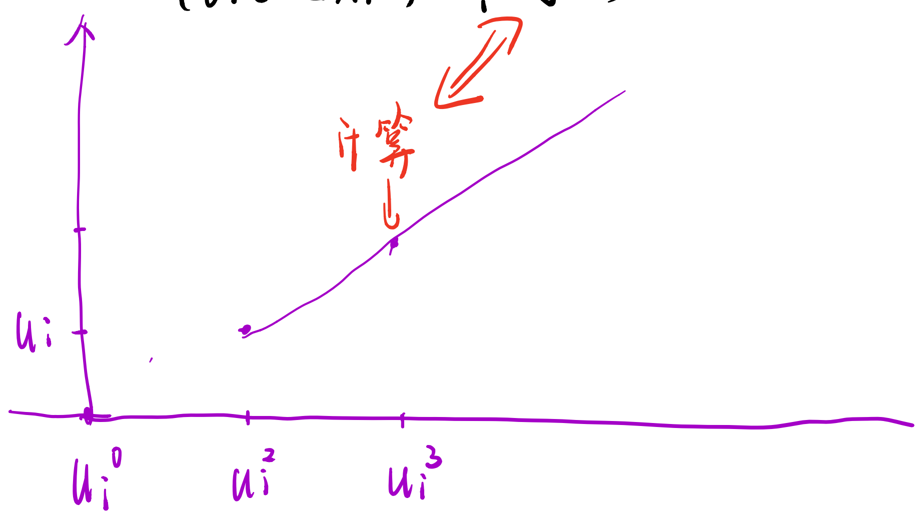
$x_i^0 = \alpha_i$, $x_i^4 = \beta_i$

$[x_i^0, x_i^1]$ ....

$$x_i^j = \alpha_i + \frac{j}{r_i}(\beta_i - \alpha_i), \quad (j = 0 \cdots r_i)$$

$$u_i(o_i) = u_i(x_i^j) + \frac{o_i - x_i^j}{x_i^{j+1} - x_i^j}\left(u_i(x_i^{j+1}) - u_i(x_i^j)\right)$$

$$(o_i \in [x_i^j, x_i^{j+1}])$$



$$u_i(x_i^0) = u_i(\alpha_i),\ u_i(x_i^1) \cdots u_i(x_i^{r_i}) = u_i(\beta_i)$$

④ 单项递增

建约束条件时用

① $\begin{cases} u(d) > u(c) \Leftrightarrow d \succ c \\ u(d) = u(c) \Leftrightarrow d \sim c \end{cases}$

② $u_i(x_i^{j+1}) - u_i(x_i^j) \geq 0$

③ $u_i(\alpha_i) = 0$

④ $\sum_i^n u_i(\beta_i) = 1$

例：线性规划中　参考第6题 ☆

Minimise $F = \cdots\cdots$

subject to：辅助变量（可去）　约 = 0

$U(c) + \sigma^+(c) - \sigma^-(c) \geq U(d) + \cdots + \varepsilon \iff c \succ d$

$U(c) + \sigma^+(d) - \sigma^-(d) = U(d) + \cdots \iff c \sim d$

$U_i(x_i^{j+1}) - U_i(x_i^j) \geq 0$

$U_i(\alpha_i) = 0$

$\sum_{i=1}^{n} U_i(\beta_i) = 1$

⑤. 有解 或无解

I. 有解：目标函数 = 0

II. 无解：目标 ⋯ > 0.

part 3

1、necessary weak perference relation $\succsim^N$

$a \succsim^N b \iff U(a) \geq U(b)$

A is nessarcay preferred to B

一定比 B　　$U_A - U_B \geq 0$　则 $U_A - U_B$ 的 min ≥ 0

2、A is possibly preferred to B

存在比 B　　$U_A - U_B$ 存在 ≥ 0　$U_A - U_B$ 的 max ≥ 0

3. 约束：

与上个 part 相比，

$U_i(g_i(a_{z_i(1)})) \geq 0$　　$U_i(g_i(a_{z_i(m)}) \leq U_i(\beta_i)$

week Auctions 拍卖

1. 拍卖 single item

n 个 bidders (卖家)

2. Social welfare : $SW = \sum_{i=1}^{n} \theta_i x_i \leq 1$

$x_i = 1$　i wins

$x_i = 0$　i loses

3. ① 从低价起拍 ② 或从高价降价 (第一个喊停的 win)

③ 密闭，保密下互不相知

④ 最高价者付第二高价

week voting 投票

1、Condorcet winner

① 选票最高无法分出胜负时

② 两两对比，看谁赢的次数多

2. Copeland winner

即便是没有 condorcet winner 也一样有效

例：31: A > E > C > D > B　　30: B > A > E

29: C > D > B　　10: D > A > E

		Winner
A vs B	41/59	B
A vs C		A
A vs D		A
A vs E		A

I. 没有人赢了所有人
则为 condorcet winner

II. 计算得分，分胜负

例 Score A = 3-1 =2

3. Borda Count

x个候选人，给x分给第一选择　x-1 分给

第二选择，以此类推。

计算最后得分，排出顺序。

例：

	A	B	C
1	1	2	3
2	1	2	3
3	1	2	3
4	3	1	2
5	3	1	2

① A是 condorcet winner

② 但 borda count 下，B分高，B是 Borda winner

4. IIA

$A >_i B >_i C >_i D$

固定A,C位置 $C >^{*} A$

调换BD不影响AC：

$A \succ_i B \succ_i D \succ_i C$

5. 独裁的：Pareto / IIA

weak Pareto / monotonicity