CPT302 W9

Forming Coalitions

Coalitional games (联盟博弈) 模拟了一个场景，其中代理可以通过合作受益。

Issues in coalitional games：

- Coalition Structure Generation
- Teamwork
- Dividing the benefits of cooperation

Coalition Structure Generation

- 原则上决定谁将一起工作（代理要加入哪个联盟）
- 结果：将 agent 划分为不连续的联盟
- 总体的划分就是联盟结构

Teamwork

解决每个联盟的优化问题：

- 决定如何一起工作
- 解决联盟的"联合问题 (joint problem)"
- 寻找如何最大化联盟本身的效用 (utility)
- 通常涉及联合规划等。

Dividing the Benefits

- 决定收益中的"谁得到什么"
- 联盟成员不能忽视彼此的偏好，因为成员可以背叛：如果你试图给我一个不好的回报，我总是可以离开
- 我们可能要考虑诸如分配的公平性等问题

Formalizing Cooperative Scenarios

A coalitional game:

$$< Ag, v >$$

其中：

- $Ag = \{1, \ldots, n\}$ 是一组代理
- $v = 2^{Ag}$ $-> R$ 是博弈的 characteristic function，R 是一个实数
- 通常的解释：如果 ν(C) = k，那么联盟 C 可以以这样的方式合作，他们将获得效用 k，然后可以分配给团队成员

联盟博弈中最重要的问题：联盟稳定吗？也就是说，联盟的所有成员都留在联盟中是否合理，或者他们是否可以通过背叛联盟而受益？

稳定是联盟形成的必要条件，但不是充分条件。

联盟博弈的 core 是向联盟成员分配一组可行的报酬分配，任何子联盟都不能合理地反对这些收益。

博弈 <Ag, ν> 中联盟 C 的 outcome 是对 C 中成员的报酬分配，用 $< x_1, \ldots, x_k >$ 表示支付给 Ag 成员的可行分配。

可行意为分给成员的报酬不能多于总的效用：

$$\nu(C) \geq \sum_{i \in C} x_i$$

比如 ν({1, 2}) = 20，可能的 outcome 为：<20, 0>, <18, 1> 等。

Objections and core

直觉上，如果联盟 C 有其他某种更好的结果让他们所有人都过得更好，他们就会反对当前这种结果。

形式上，如果对于大联盟 $C \subseteq Ag$ 来说，如果存在一些结果 $< x'_1, \ldots, x'_k >$，C 会拒绝结果 $< x_1, \ldots, x_n >$，其中：

$$x'_i > x_i \text{ for all } i \in C$$

core 是大联盟的一系列结果，其中没有联盟反对。如果 core 是非空的，那么大联盟就是稳定的，因为没有人可以从叛逃中受益。

因此，问大联盟是否稳定，就是在问：core 是非空的吗？

举个例子，Ag = {1, 2}, ν({1}) = 5, ν({2}) = 5, ν({1, 2}) = 20 (两个代理，自己干可以都得到5，一起干得到 20)。然后 outcome <20, 0>（即代理 1 得到一切）不在核心中，因为代理 {2} 会反对（它自己干可以得到 5）。然而，outcome <15, 5> 在核心：尽管这对代理2 来说似乎不公平，但这个代理没有理由拒绝，因为它没有损失什么。

Share Benefits of Cooperation

Shapley value 是定义如何公平分配合作利益的最著名尝试。它通过考虑代理的贡献来做到这一点。

代理 i 的 Shapley value 是 i 预期为联盟贡献的平均值。

Shapley value 是满足以下公理的值：

- symmetry (假设 i 和 j 可以互相替代，那它们收益应该一样)
- dummy player (未做贡献的话收益为 0)
- additivity (如果同一批人完成两项任务，那么两项任务的收益一起分配应该和分开分配的结果一致)

Shapley Value

让 $\delta_i(S)$ 表示代理 i 加入 $S \subseteq Ag$ 的数额：

$$\delta_i(S) = \nu(S \cup \{i\}) - \nu(S)$$

上式表示 i 加入 S 后增加的贡献，即 i 对 S 的边际贡献。

然后 i 的 shapley value 表示为 $\mathrm{Sh(S,i)}/\varphi_i$:

$$\mathrm{Sh(S,i)}/\varphi_i = \frac{\sum_{r \in R} \delta_i\left(S_i(r)\right)}{|Ag|!}$$

其中 R 是 Ag 的所有排序的集合，$S_i(r)$ 是排序 r 中 **i** 之前的代理的集合 (这里对应 $\delta_i(S)$ 公式中的 S，因为要看 i 的边际贡献，所以不能包括 i)。

例子：

给定两个代理 1 和 2，以及相应的 outcomes，分别计算 1 和 2 的 shapley value：

S	$v(S)$
()	0
(1)	1
(2)	3
(12)	6

Ag = {1, 2} 有两种排序：{12}, [21]，因此 R = {{12}, [21]}，|Ag| = 2。

$$\begin{aligned}
Sh(\{1,2\},1) &= \frac{1}{2}\cdot(v(1)-v()+v(21)-v(2)) \\
&= \frac{1}{2}\cdot(1-0+6-3)=2 \\
Sh(\{1,2\},2) &= \frac{1}{2}\cdot(v(12)-v(1)+v(2)-v()) \\
&= \frac{1}{2}\cdot(6-1+3-0)=4
\end{aligned}$$

以 1 为例，第一种排序为 {12}，计算 $\delta_1(\{\}) = v(1) - v() = 1 - 0 = 1$，然后是第二种排序 [21]，计算 $\delta_1(\{2\}) = v(21) - v(2) = 6 - 3 = 3$。

Representing Coalitional Games

对于代理来说，重要的是要知道联盟的 core 是否是非空的。然而联盟博弈的明显表示会在 Ag 的大小上呈指数级增长（如上面的例子，2 个代理需要 4 行的表格表示 characteristic function），这在实践中是不可能实现的。

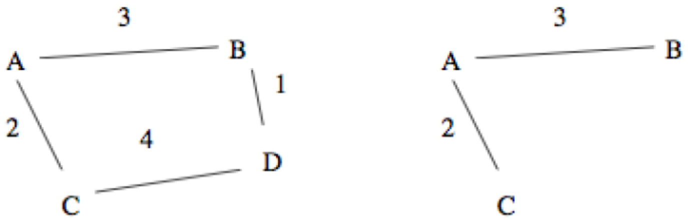
Two approaches to this problem：

- 尝试找到一个在“大多数”情况下简洁的完整表示
- 尝试找到一个不完整但总是简洁的表示
- 一种常见的方法：在组合结构上解释 characteristic function

Induced Subgraph

将 v 表示为基于 Ag 的无向图，其中两个 node i 和 j 之间的权重为 $w_{i,j}$。那么联盟 C 的值为：

$$\nu(C) = \sum_{\{i,j\}\subseteq Ag} w_{i,j}$$



the original graph defining v

subgraph induced by {A,B,C} giving v({A,B,C}) = 3 + 2 = 5

Marginal Contribution Nets

把 characteristic function 表示为规则：pattern → value。

pattern 是代理的结合，如果 C 是 pattern 中代理的父集，则规则应用于一组代理 C。联盟的价值是适用于该联盟的所有规则的值相加 (看例子)。

我们还可以允许在规则中存在否定 (代表代理不存在，看例 3)。

例子 1:

$$\begin{aligned} a \wedge b &\longrightarrow 5 \\ b &\longrightarrow 2 \end{aligned}$$

We have: v({a}) = 0, v({b}) = 2, and v({a, b}) = 7。

因为 a 没有 rule，所以 v({a}) = 0；因为联盟的价值是适用于该联盟的所有规则的值相加，所以 v({a, b}) = 5 + 2 + 0 = 7。

例子 2:

Consider the marginal contribution net:

$$\begin{array}{lll} a & \longrightarrow & 2 \\ a \wedge b & \longrightarrow & 7 \\ b & \longrightarrow & 3 \\ c & \longrightarrow & 4 \\ b \wedge c & \longrightarrow & -3 \end{array}$$

(a) Let ν be the characteristic function defined by these rules. Give the values of the following:

1. $\nu(\varnothing)$
2. $\nu(\{a\})$
3. $\nu(\{b\})$
4. $\nu(\{a,b\})$
5. $\nu(\{a,b,c\})$

Answer:

$$\nu(\varnothing)=0, \nu(\{a\})=2, \nu(\{b\})=3, \nu(\{a,b\})=12, \nu(\{a,b,c\})=13.$$

例子 3:

We can also allow negations in rules (i.e. for when an agent is not present).

rule set 2:

$$\begin{array}{r} a \wedge b \rightarrow 5 \\ b \rightarrow 2 \\ c \rightarrow 4 \\ b \wedge \neg c \rightarrow -2 \end{array}$$

$v_{rs2}(\{a\}) = 0$	no rules apply
$v_{rs2}(\{b\}) = 2+ -2 = 0$	2nd and 4th rules
$v_{rs2}(\{c\}) = 4$	3rd rule
$v_{rs2}(\{a, b\}) = 5+2+-2 = 5$	1st, 2nd and 4th rules
$v_{rs2}(\{a, c\}) = 4$	3rd rule
$v_{rs2}(\{b, c\}) = 2+4 = 6$	2nd and 3rd rules
$v_{rs2}(\{a, b, c\}) = 5+2+4 = 11$	1st, 2nd and 3rd rules

注意：计算 v(b) 的结果为 0，因为 $b \wedge \neg c \rightarrow -2$ 中使用了否定，因此这里 c 实际上不存在，因此实际上的 rule 为 $b \rightarrow -2$。而因为最后一个 rule 中 c 不存在，因此计算和 b, c 有关的值时不可以包括该 rule (例如 v({b, c}) 和 v({a, b, c}))。