CoopGT

Vincent Knight

Mar 27, 2024

CONTENTS

A library for the study of cooperative game theory.

CHAPTER

ONE

TUTORIAL: SHAPLEY VALUE REGRESSION

The Shapley value is a cooperative game theoretic tool used to share a resource between players.

In this tutorial we will use it to identify the importance of different variables to a linear regression. This is commonly referred to as Shaply Value Regression.

1.1 Installing CoopGT

With a working installation of Python, open a command line tool and type:

```
$ python -m pip install coopgt
```

1.2 Linear Regression

In cooperative game theory a characteristic function is a mapping from all groups of players to a given value. In this case it will correspond to the R^2 value for a linear model for some data. The y variable is going to be predicted by fitting a linear model to three variables:

$$y = c_1x_1 + c_2x_2 + c_3x_3$$

Here are the R^2 values (you are welcome to see `main.py` for the code used to generate them):

Model	R^2
$y = c_1x_1$	0.075
$y = c_2x_2$	0.086
$y = c_3x_3$	0.629
$y = c_1x_1 + c_2x_2$	0.163
$y = c_1x_1 + c_3x_3$	0.63
$y = c_2x_2 + c_3x_3$	0.906
$y = c_1x_1 + c_2x_2 + c_3x_3$	0.907

1.3 Defining the characteristic function

We can use that table of R^2 values to create the characteristic function:

```
>>> characteristic_function = {
...     (): 0,
...     (1,): 0.075,
...     (2,): 0.086,
...     (3,): 0.629,
...     (1, 2): 0.163,
...     (1, 3): 0.63,
...     (2, 3): 0.906,
...     (1, 2, 3): 0.907,
... }
```

1.4 Obtaining the Shapley value

We now compute the Shapley value:

```
>>> import coopgt.shapley_value
>>> shapley_value = coopgt.shapley_value.calculate(characteristic_
↪function=characteristic_function)
>>> shapley_value.round(4)
array([0.0383, 0.1818, 0.6868])
```

From this analysis we would conclude that the parameter that contributes the most is in fact x_3.

CHAPTER

TWO

HOW TO

How to:

2.1 Install CoopGT

To install from the Python Package index (PyPi) run the following command:

```
$ python -m pip install coopgt
```

To install a development version from source:

```
$ git clone https://github.com/drvinceknight/coopgt.git
$ cd coopgt
$ python -m pip install flit
$ python -m flit install --symlink
```

2.2 Create a characteristic function

To create a characteristic function use a Python `dict` to map tuples of player indices to the payoff values. For example to create the following characteristic function:

$$v(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 3, & \text{if } C = \{2\} \\ 12, & \text{if } C = \{1, 2\} \end{cases}$$

Write:

```
>>> characteristic_function = {(): 0, (1,): 6, (2,): 3, (1, 2): 12}
>>> characteristic_function
{(): 0, (1,): 6, (2,): 3, (1, 2): 12}
```

2.3 Check if a characteristic function is valid

To check if a characteristic function is valid use `coopgt.characteristic_function_properties.is_valid`.

For example to check if the following characteristic function which does not map all elements of the power set of the set of players is valid:

$$v_1(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 10, & \text{if } C = \{1, 2\} \\ 42, & \text{if } C = \{2, 3\} \\ 42, & \text{if } C = \{1, 2, 3\} \end{cases}$$

First *create the characteristic function*:

```
>>> characteristic_function = {
...     (): 0,
...     (1,): 6,
...     (2,): 12,
...     (3,): 42,
...     (1, 2): 10,
...     (2, 3): 42,
...     (1, 2, 3): 42,
... }
```

Then:

```
>>> import coopgt.characteristic_function_properties
>>> coopgt.characteristic_function_properties.is_valid(characteristic_
↪function=characteristic_function)
False
```

2.4 Check if a characteristic function is monotone

To check if a characteristic function is *monotone* use `coopgt.characteristic_function_properties.is_monotone`.

For example to check if the following characteristic function is monotone:

$$v_1(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 10, & \text{if } C = \{1, 2\} \\ 42, & \text{if } C = \{1, 3\} \\ 42, & \text{if } C = \{2, 3\} \\ 42, & \text{if } C = \{1, 2, 3\} \end{cases}$$

First *create the characteristic function*:

```
>>> characteristic_function = {
...     (): 0,
...     (1,): 6,
...     (2,): 12,
...     (3,): 42,
...     (1, 2): 10,
...     (1, 3): 42,
...     (2, 3): 42,
...     (1, 2, 3): 42,
... }
```

Then:

```
>>> import coopgt.characteristic_function_properties
>>> coopgt.characteristic_function_properties.is_monotone(
...     characteristic_function=characteristic_function
... )
False
```

2.5 Check if a characteristic function is superadditive

To check if a characteristic function is *superadditive* use `coopgt.characteristic_function_properties.is_superadditive`.

For example to check if the following characteristic function is superadditive:

$$v_1(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 10, & \text{if } C = \{1, 2\} \\ 42, & \text{if } C = \{1, 3\} \\ 42, & \text{if } C = \{2, 3\} \\ 42, & \text{if } C = \{1, 2, 3\} \end{cases}$$

First *create the characteristic function*:

```
>>> characteristic_function = {
...     (): 0,
...     (1,): 6,
...     (2,): 12,
...     (3,): 42,
...     (1, 2): 10,
...     (1, 3): 42,
...     (2, 3): 42,
...     (1, 2, 3): 42,
... }
```

Then:

```
>>> import coopgt.characteristic_function_properties
>>> coopgt.characteristic_function_properties.is_superadditive(
...     characteristic_function=characteristic_function
... )
False
```

2.6 Identify the predecessors of a player for a given permutation

To find the *predecessors* $S_\pi(i)$ of a player i for a permutation π use `coopgt.shapley_value.predecessors`.

For example to find $S_{(3,2,1)}(1)$:

```
>>> import coopgt.shapley_value
>>> pi = (3, 2, 1)
>>> coopgt.shapley_value.predecessors(permutation=pi, i=1)
{2, 3}
```

2.7 Calculate the marginal contribution of a player

To find the *marginal contribution* $\Delta_\pi^G(i)$ of a player i for a permutation π in a game $G = (N, v)$ use `coopgt.shapley_value.marginal_contribution`.

For example for $G = (3, v)$ and $\pi = (3, 2, 1)$ to find $\Delta_\pi^G(i)(1)$:

$$v(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 12, & \text{if } C = \{1,2\} \\ 42, & \text{if } C = \{2,3\} \\ 42, & \text{if } C = \{1,2,3\} \end{cases}$$

First *create the characteristic function*:

```
>>> characteristic_function = {
...     (): 0,
...     (1,): 6,
...     (2,): 12,
...     (3,): 42,
...     (1, 2): 12,
...     (1, 3): 42,
...     (2, 3): 42,
...     (1, 2, 3): 42,
... }
```

Then:

```
>>> import coopgt.shapley_value
>>> pi = (3, 2, 1)
>>> coopgt.shapley_value.marginal_contribution(
...     characteristic_function=characteristic_function, permutation=pi, i=1
... )
0
```

2.8 Calculate the Shapley value

To find the *Shapley value* for a game $G = (N, v)$ use `coopgt.shapley_value.calculate`.

For example for $G = (3, v)$:

$$v(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 12, & \text{if } C = \{1, 2\} \\ 42, & \text{if } C = \{2, 3\} \\ 42, & \text{if } C = \{1, 2, 3\} \end{cases}$$

First *create the characteristic function*:

```
>>> characteristic_function = {
...     (): 0,
...     (1,): 6,
...     (2,): 12,
...     (3,): 42,
...     (1, 2): 12,
...     (1, 3): 42,
...     (2, 3): 42,
...     (1, 2, 3): 42,
... }
```

Then:

```
>>> import coopgt.shapley_value
>>> coopgt.shapley_value.calculate(characteristic_function=characteristic_function)
array([ 2.,  5., 35.])
```

CHAPTER

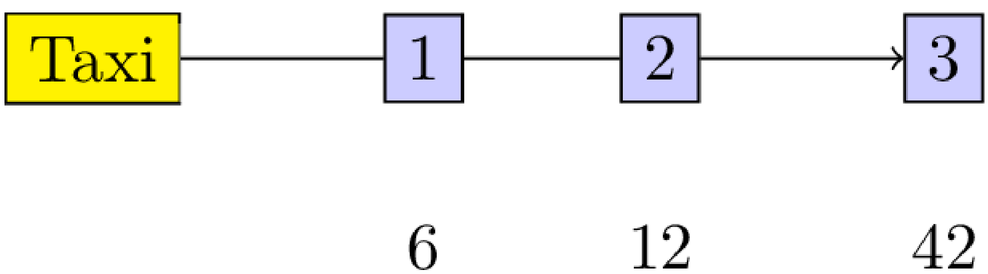
THREE

COOPERATIVE GAME THEORY TEXT BOOK

3.1 Characteristic Function Games

3.1.1 Motivating example: a taxi trip

Consider the following situation:

> 3 players share a taxi. Here are the costs for each individual journey:
>
> - Player 1: 6
> - Player 2: 12
> - Player 3: 42
>
> As illustrated here:
>
> 
>
>
> How can we represent this situation mathematically?

3.1.2 Definition of a characteristic function game

A **characteristic function game** G is given by a pair (N, v) where N is the number of players and $v : 2^{[N]} \to \mathbb{R}$ is a **characteristic function** which maps every coalition of players to a payoff.

Question

For the *taxi fare* what is the characteristic function?

Answer

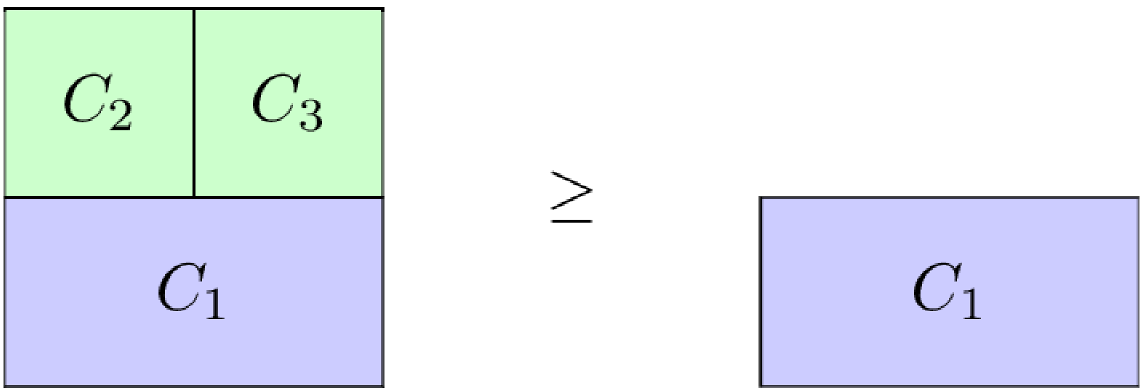
The number of players $N = 3$ and to construct the characteristic function we first obtain the power set (ie all possible coalitions) $2^{\{1,2,3\}} = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \Omega\}$ where Ω denotes the set of all players $\Omega = \{1, 2, 3\}$.

The characteristic function is given below:

$$v(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 12, & \text{if } C = \{1, 2\} \\ 42, & \text{if } C = \{1, 3\} \\ 42, & \text{if } C = \{2, 3\} \\ 42, & \text{if } C = \{1, 2, 3\} \end{cases}$$

3.1.3 Definition of a monotone characteristic function game

A characteristic function game $G = (N, v)$ is called **monotone** if it satisfies $v(C_2) \geq v(C_1)$ for all $C_1 \subseteq C_2$.



Question

Which of the following characteristic function games are monotone:

1. *The taxi fare*.
2. $G = (3, v_1)$ with v_1 defined as:

$$v_1(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 10, & \text{if } C = \{1, 2\} \\ 42, & \text{if } C = \{1, 3\} \\ 42, & \text{if } C = \{2, 3\} \\ 42, & \text{if } C = \{1, 2, 3\} \end{cases}$$

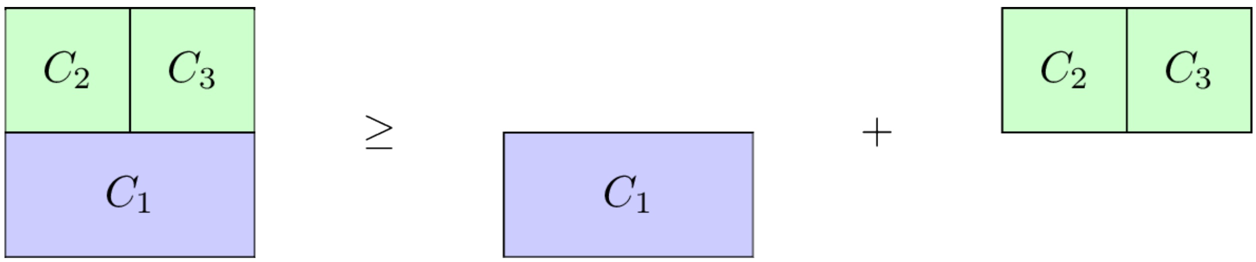
Answer

1. The taxi fare characteristic function is monotone.

2. This game is not as $\{2\} \subseteq \{1, 2\}$ however $v_1(\{2\}) > v_1(\{1, 2\})$.

3.1.4 Definition of a superadditive characteristic function game

A characteristic function game $G = (N, v)$ is called **superadditive** if it satisfies $v(C_1 \cup C_2) \geq v(C_1) + v(C_2)$.



Question

Which of the following characteristic function games are superadditive:

1. *The taxi fare*.
2. $G = (3, v_2)$ with v_2 defined as:

$$v_2(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 18, & \text{if } C = \{1, 2\} \\ 48, & \text{if } C = \{1, 3\} \\ 55, & \text{if } C = \{2, 3\} \\ 80, & \text{if } C = \{1, 2, 3\} \end{cases}$$

Answer

1. The taxi fare characteristic function is not superadditive as $v(\{1\}) + v(\{2\}) = 18$ but $v(\{1, 2\}) = 12$.
2. This game is superadditive.

[Maschler2013] is recommended for further reading.

3.2 Shapley Value

3.2.1 Motivating example: Sharing a taxi fare

For the *taxi trip game* with characteristic function:

$$v(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 6, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 12, & \text{if } C = \{1,2\} \\ 42, & \text{if } C = \{1,3\} \\ 42, & \text{if } C = \{2,3\} \\ 42, & \text{if } C = \Omega = \{1,2,3\} \end{cases}$$

How much should each individual contribute?

3.2.2 Payoff vector

This corresponds to a payoff vector $\lambda \in \mathbb{R}^N_{\geq 0}$ that divides the value of the grand coalition Ω between the various players. Thus λ must satisfy:

$$\sum_{i=1}^{N} \lambda_i = v(\Omega)$$

Thus one potential solution to our taxi example would be $\lambda = (14, 14, 14)$. Obviously this is not ideal for player 1 and/or 2: they actually pay more than they would have paid without sharing the taxi!

Another potential solution would be $\lambda = (6, 6, 30)$, however at this point sharing the taxi is of no benefit to player 1. Similarly $(0, 12, 30)$ would have no incentive for player 2.

To find a "fair" distribution of the grand coalition we must define what is meant by "fair". We require four desirable properties:

- *Efficiency*.
- *Null player*.
- *Symmetry*.
- *Additivity*.

Definition of efficiency

For $G = (N, v)$ a payoff vector λ is **efficient** if:

$$\sum_{i=1}^{N} \lambda_i = v(\Omega)$$

Question

For the *taxi fare* which of the following payoff vectors are **efficient**?

- $\lambda = (42, 0, 0)$.
- $\lambda = (12, 12, 18)$.
- $\lambda = (14, 14, 14)$.
- $\lambda = (1, 14, 28)$.

Answer

For all of these cases we need $v(\Omega) = v(\{1, 2, 3\}) = 42$.

- $\lambda = (42, 0, 0)$ is efficient as $42 + 0 + 0 = 42$.
- $\lambda = (12, 12, 18)$ is efficient as $12 + 12 + 18 = 42$.
- $\lambda = (14, 14, 14)$ is efficient as $14 + 14 + 14 = 42$.
- $\lambda = (1, 14, 28)$ is not efficient as $1 + 14 + 28 = 43$.

Definition of null player

For $G(N, v)$ a payoff vector possesses the **null player property** if $v(C \cup \{i\}) = v(C)$ for all $C \in 2^\Omega$ then:

$$x_i = 0$$

Question

1. For the *taxi fare* which of the following payoff vectors possess the **null player property**?

- $\lambda = (42, 0, 0)$.
- $\lambda = (12, 12, 18)$.
- $\lambda = (14, 14, 14)$.
- $\lambda = (1, 14, 28)$.

2. For game $G(3, v_3)$ with v_3 defined as:

$$v_3(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 0, & \text{if } C = \{1\} \\ 12, & \text{if } C = \{2\} \\ 42, & \text{if } C = \{3\} \\ 12, & \text{if } C = \{1, 2\} \\ 42, & \text{if } C = \{1, 3\} \\ 42, & \text{if } C = \{2, 3\} \\ 42, & \text{if } C = \Omega = \{1, 2, 3\} \end{cases}$$

which of the following payoff vectors possess the **null player property**?

- $\lambda = (42, 0, 0)$.
- $\lambda = (12, 12, 18)$.
- $\lambda = (14, 14, 14)$.
- $\lambda = (0, 15, 28)$.

Answer

1. For the *taxi fare* there is no player i such that $v(C \cup \{i\}) = v(C)$ for all $C \in 2^{\Omega}$. Indeed, $v(\{1\} \cup \{2\}) \neq v(\{1\})$ and $v(\{1\} \cup \{3\}) \neq v(\{1\})$ and $v(\emptyset \cup \{1\}) \neq v(\emptyset)$. Thus, all the payoff vector have the null property.
2. **For v_3 we have that $v(C \cup \{1\}) = V(C)$ for all**
   $C \in 2^{\Omega}$. Thus the only payoff vector that has the null player property is $\lambda = (0, 15, 28)$.

Definition of symmetry

For $G(N, v)$ a payoff vector possesses the **symmetry property** if $v(C \cup i) = v(C \cup j)$ for all $C \in 2^{\Omega} \setminus \{i, j\}$ then:

$$x_i = x_j$$

Question

1. For the *taxi fare* which of the following payoff vectors possess the **symmetry property**?
   - $\lambda = (42, 0, 0)$.
   - $\lambda = (12, 12, 18)$.
   - $\lambda = (14, 14, 14)$.
   - $\lambda = (1, 14, 28)$.
2. For game $G(3, v_4)$ with v_4 defined as:

$$v_4(C) = \begin{cases} 0, & \text{if } C = \emptyset \\ 2, & \text{if } C = \{1\} \\ 2, & \text{if } C = \{2\} \\ 2, & \text{if } C = \{3\} \\ 12, & \text{if } C = \{1, 2\} \\ 12, & \text{if } C = \{1, 3\} \\ 42, & \text{if } C = \{2, 3\} \\ 42, & \text{if } C = \Omega = \{1, 2, 3\} \end{cases}$$

which of the following payoff vectors possess the **null player property**?

- $\lambda = (42, 0, 0)$.
- $\lambda = (12, 12, 18)$.
- $\lambda = (14, 14, 14)$.
- $\lambda = (0, 15, 28)$.

Answer

1. For the *taxi fare* there is no pair of players i and j such that $v(C \cup i) = v(C \cup j)$ for all $C \in 2^{\Omega} \setminus \{i, j\}$. Indeed, $v(\{1\} \cup \{2\}) \neq v(\{1\} \cup \{3\})$ and $v(\{2\} \cup \{3\}) \neq v(\{2\} \cup \{1\})$. Thus, all the payoff vector have the symmetry property.

2. For v_4 we have that $v(\emptyset \cup \{2\}) = v(\emptyset \cup \{3\})$, $v(\{1\} \cup \{2\}) = v(\{1\}\emptyset \cup \{3\})$ so players 2 and 3 contribute the same to all subsets. However $v(\{2\} \cup \{3\}) \neq v(\{2\}\emptyset \cup \{1\})$ and $v(\{2\} \cup \{1\}) \neq v(\{2\}\emptyset \cup \{3\})$ thus player 1 does not contribute the same as either player 2 or player 3 to all subsets. Thus the payoff vectors that have the symmetry property are $\lambda = (42, 0, 0)$ and $\lambda = (14, 14, 14)$.

Definition of additivity

For $G_1 = (N, v_1)$ and $G_2 = (N, v_2)$ and $G^+ = (N, v^+)$ where $v^+(C) = v_1(C) + v_2(C)$ for any $C \in 2^\Omega$. A payoff vector possesses the **additivity property** if:

$$x_i^{(G^+)} = x_i^{(G_1)} + x_i^{(G_2)}$$

We will not prove in this course but in fact there is a single payoff vector that satisfies these four properties. To define it we need two last definitions.

Definition of predecessors

If we consider any permutation π of $[N]$ then we denote by $S_\pi(i)$ the set of **predecessors** of i in π:

$$S_\pi(i) = \{j \in [N] \mid \pi(j) < \pi(i)\}$$

For example for $\pi = (1, 3, 4, 2)$ we have $S_\pi(4) = \{1, 3\}$.

Definition of marginal contribution

If we consider any permutation π of $[N]$ then the **marginal contribution** of player i with respect to π is given by:

$$\Delta_\pi^G(i) = v(S_\pi(i) \cup i) - v(S_\pi(i))$$

3.2.3 Definition of the Shapley value

Given $G = (N, v)$ the **Shapley value** of player i is denoted by $\phi_i(G)$ and given by:

$$\phi_i(G) = \frac{1}{N!} \sum_{\pi \in \Pi_n} \Delta_\pi^G(i)$$

Question

Obtain the Shapley value for the *taxi fare*.

Answer

For $\pi = (1, 2, 3)$:

$$\Delta_\pi^G(1) = 6$$
$$\Delta_\pi^G(2) = 6$$
$$\Delta_\pi^G(3) = 30$$

For $\pi = (1, 3, 2)$:

$$\Delta_\pi^G(1) = 6$$
$$\Delta_\pi^G(2) = 0$$
$$\Delta_\pi^G(3) = 36$$

For $\pi = (2, 1, 3)$:

$$\Delta_\pi^G(1) = 0$$
$$\Delta_\pi^G(2) = 12$$
$$\Delta_\pi^G(3) = 30$$

For $\pi = (2, 3, 1)$:

$$\Delta_\pi^G(1) = 0$$
$$\Delta_\pi^G(2) = 12$$
$$\Delta_\pi^G(3) = 30$$

For $\pi = (3, 1, 2)$:

$$\Delta_\pi^G(1) = 0$$
$$\Delta_\pi^G(2) = 0$$
$$\Delta_\pi^G(3) = 42$$

For $\pi = (3, 2, 1)$:

$$\Delta_\pi^G(1) = 0$$
$$\Delta_\pi^G(2) = 12$$
$$\Delta_\pi^G(3) = 42$$

Using this we obtain:

$$\phi(G) = (2, 5, 35)$$

Thus the fair way of sharing the taxi fare is for player 1 to pay 2, player 2 to pay 5 and player 3 to pay 35.

[Maschler2013] is recommended for further reading.

CHAPTER

FOUR

REFERENCE

4.1 Bibliography

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search

BIBLIOGRAPHY

[Maschler2013] Maschler, M., Eilon Solan, and Shmuel Zamir. “Game theory. Translated from the Hebrew by Ziv Hellman and edited by Mike Borns.” (2013).