World Journal of Mathematics and Physics
ISSN: 2960-0103
DOI: https://doi.org/10.61784/wjmp3011

DATA MODEL SPECIFIC MATRIX AND GAME THEORY APPLICATIONS

LinFeng Murong
Yuan Yang Volunteer Team of Xinghua Public Welfare Foundation, Hangzhou 310020, Zhejiang, China.
Corresponding Email: muronglinfeng2024@qq.comn

Abstract: Under conventional game theory models, a deeper analysis of players' mixed strategies through probabilistic methods has become a mainstream analytical approach. By infinitely approaching a fixed probabilistic expected value point, traditional equilibrium algorithms in game theory can calculate equilibrium points. However, in most cases, it is not possible to directly make relevant final predictions in games through probabilistic means. As a result, the practical application of game theory is very limited, making it difficult to be widely applied as a module of applied mathematics. This paper provides an overview of the model architecture for the application of game theory through computational mathematical methods based on matrix model calculations. Although it considers fundamentals similar to Nash equilibrium, there are still essential differences.
Keywords: Data model specific matrix; Computational mathematics; Game theory; Gray number model; Dimension reduction and increase

1 INTRODUCTION

The theories related to game theory focus on how to achieve effects similar to the optimal equilibrium through probability theory[1]. In fact, there is still some controversy regarding mixed strategies. This article mainly aims to elaborate on the relevant applications of computational mathematics. When dealing with a large number of nested models, detailed mathematical reasoning is still required to verify the winning probabilities of special matrices in the data model and the optimal solutions.

2 DATA MODEL AND MATRIX ARCHITECTURE PROCESS

2.1 Data Model Background Establishment

First, let's take a look at the traditional game theory model. In the example of a competition, both A and B have two levels of teams, with equal strength. a_1、a_2、b_1、b_2 represent A's upper-level team, A's lower-level team, B's upper-level team, and B's lower-level team, respectively. Thus, the box represents the payoff matrix. If we directly represent the results of the single-team confrontations in a cross-competitive manner, it is as follows (Table 1):

Table 1 Single Team Adversarial Benefit Matrix

	b_1	b_2
a_1	(0,0)	(1,0)
a_2	(0,1)	(0,0)

In a real gaming environment, both Team A and Team B need to compete against each other, but they often cannot appear on the field at the same time; in fact, competitions typically take place separately. Therefore, we assume that two teams compete simultaneously at the same point in time, which means that only the outcomes represented in the green squares below are the possible simultaneous results of the game. Even with multiple matrices, this event cannot be effectively resolved. Thus, a completely new matrix model is needed to simulate such scenarios. For example, the possible options might be as follows: after a_1 competes against b_1, the remaining matchup must be a_2 against b_2. Besides this, the only other possibility is that a_1 competes against b_2, which means that the remaining matchup must be a_2 against b_1 (Table 2-3).

Table 2 The First Possibility of the Single Team Confrontation Payoff Matrix's Mutual Exclusivity

	b_1	b_2
a_1	$P(a_1, b_1)$	$P(a_1, b_2)$
a_2	$P(a_2, b_1)$	$P(a_2, b_2)$

Table 3 The Second Possibility of the Mutual Exclusivity of the Single Team Confrontation Payoff Matrix

	b_1	b_2
a_1	$P(a_1, b_1)$	$P(a_1, b_2)$
a_2	$P(a_2, b_1)$	$P(a_2, b_2)$

© By the Author(s) 2024, under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0).

The construction of this matrix will be carried out in 2.2 through the "mutually exclusive probability matrix," which effectively eliminates the matrix parts where simultaneous participation occurs. At the same time, we incorporate the differences in game sessions to describe and include time series factors. In this way, there is a certain time series order, resulting in a very complex representation of the game matrix profit matrix (Table 4).

Table 4 Dual Team Competitive Payoff Matrix

	b_1+b_2	b_2+b_1
a_1+a_2	P（a_1b_1,a_2b_2）	P（a_1b_2,a_2b_1）
a_2+a_1	P（a_2b_1,a_1b_2）	P（a_2b_2,a_1b_1）

Due to considerations based on time series, it is assumed that $P(a_1b_1, a_2b_2) = P(a_2b_2|a_1b_1)$, where the correlation equilibrium of a_2b_2 is influenced by a_1b_1, which is the probability of the first match. The results of the second match need to take this condition into account when constructing the model. Finally, we need an evaluation hypothesis to determine whether a match is won, and we use the number of matches won as the result, which is also the mainstream strategy in competitions.

2.2 Spatial Architecture of the Game Model

Based on the assumptions of the above game model, the first step is to construct a payoff matrix to express the related benefits. In this paper, we refer to it as the "spatial architecture of the game model," which is used to elucidate a series of decision-making processes that follow.

2.2.1 Spatial Expansion Matrix

For Team A, we represent the aforementioned strength quantitatively, essentially as a one-dimensional spatial vector $v_a(1, 0.5)$, which respectively represents the quantitative strength of Team A's upper and lower teams. Similarly, the spatial vector $v_b(0.8, 0.4)$ represents the quantitative strengths of Team B's two teams. Let XA and XB be two vector spaces, and by organizing them according to the number of matches, we fill the spatial expansion matrix based on team strengths:

By copying teams A and B into a matrix with the same number of rows, two matches represent a 2*2 matrix. By setting the "spatial expansion matrix" row number for 2, we can obtain:

The spatial expansion matrix for A's competition team strength is $XA\begin{bmatrix}1 & 0.5\\ 1 & 0.5\end{bmatrix}$, and for B's competition team strength is $XB\begin{bmatrix}0.8 & 0.4\\ 0.8 & 0.4\end{bmatrix}$.

2.2.2 Sorting Matrix

It is called a sorting matrix because this matrix is used to sort the resource elements of the aforementioned spatial expansion matrix according to different rules. For example, in the above case, the intersection of events A and B is empty; A and B are mutually exclusive events, meaning they cannot happen simultaneously. Assuming $A \cap B$ is an impossible event, then $A \cap B = \Phi$, indicating that events A and B are mutually exclusive, and they will not occur at the same time in any single trial. For the mutual exclusivity representation, it corresponds to n linearly independent characteristic vectors, based on which a diagonal matrix can be constructed[2], forming an abstracted "mutually exclusive probability matrix" as the sorting matrix. If we use rows as matches, the sorting matrix can be constructed as $\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix},\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$, representing that Team A and Team B's selections cannot repeat their own team. For instance, in the same match, two teams cannot participate simultaneously. In the case of multiple teams, we can construct a more complex "mutually exclusive probability matrix." After transforming the original spatial expansion matrix through Hadamard product matrix operations, we can obtain new sorting results:

Sorting Matrix:

$$X_1\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} 、 X_2\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix} \tag{1}$$

By conducting Hadamard product matrix transformations, the results of $XA \bigcirc X1$ and $XB \bigcirc X2$ yield the actual operational matrices:

$$\begin{aligned} XA_1\begin{bmatrix}1 & 0\\ 0 & 0.5\end{bmatrix} 、 XB_1\begin{bmatrix}0.8 & 0\\ 0 & 0.4\end{bmatrix} \\ XA_2\begin{bmatrix}0 & 0.5\\ 1 & 0\end{bmatrix} 、 XB_2\begin{bmatrix}0 & 0.4\\ 0.8 & 0\end{bmatrix} \end{aligned} \tag{2}$$

2.2.3 Dimensionality Reduction Operation

Since we believe that the results of matches are not matrix operations, actual calculations will also consume a lot of resources (Figure 1).

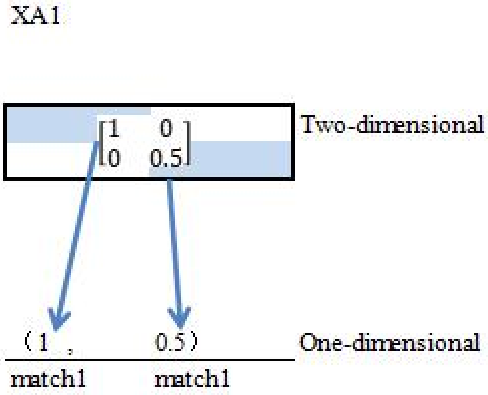

Figure 1 Dimensionality Reduction Operation

By projecting the vector space through eigenvalue decomposition and vectorizing the non-zero elements[3], we can express it as:

$$\begin{aligned} E_1&=vec(XA_1)-vec(XB_1)=(1,0.5)-(0.8,0.4)=(0.2,0.1)\\ E_2&=vec(XA_1)-vec(XB_2)=(1,0.5)-(0.4,0.8)=(0.6,-0.3)\\ E_3&=vec(XA_2)-vec(XB_1)=(0.5,1)-(0.8,0.4)=(-0.3,0.6)\\ E_4&=vec(XA_2)-vec(XB_2)=(0.5,1)-(0.4,0.8)=(0.1,0.2) \end{aligned} \quad (3)$$

This way, using the logical judgment formula, we can calculate that the results of E_1 and E_4 indicate victory, while the other two are draws, leading to a scenario where one team ultimately wins 100%.

2.2.4 Example: Discovery of a Special Relative Winning Matrix

Under the assumptions of the game's outcomes and determinations, if we operate according to the Nash equilibrium method[4], we can find that even after calculating the expected equilibrium point under mixed strategies to determine the most favorable returns for both parties, in real life, there is no necessity for opponents to engage in the game because there is no probability of winning at all. This matrix is essentially referred to as a permanently relative winning matrix for teams with significant strength. Through the matrix simulation described above, there will always be a quantifiable interval of strength differences that has both an upper and a lower limit, inevitably including gray factors[5], known as the gray number interval. At the same time, it needs to be recognized that there might be exceptional cases where the relative winning matrix is reversed, such as in the cases of E_2 and E_3. In situations of stark strength disparity, gains can be achieved through advantageous positioning. In three-player game scenarios, results can be reversed even with a comprehensive disadvantage, as long as certain fixed combinations appear on the sorting matrix, it is possible to obtain a relative winning matrix of strength disadvantage but always winning. Furthermore, there exists a relative winning matrix that can never achieve victory. Since this paper primarily elaborates on the matrix simulation of game theory architecture, we will not delve into proofs here. This differentiates it from traditional game theory, where Nash equilibrium is based on probabilistic expectations of payoff results, while the purpose of the relative winning matrix here is to construct a range of game arrays to obtain the absolute relative winning difference interval in the game. The reason for using matrix simulation to express victory in game theory is that multi-dimensional game operations in practical situations are very complex; appropriate matrix algorithms must be adjusted to save considerable time[6]. Meanwhile, it also includes the impact of time series on the outcomes of the next game, so we still need to discuss the overview of the game matrix under time series considerations.

2.3 The Impact of Time Series, Logic Matrix, and Compensation Matrix

In real life, there are various situations such as match sorting, substitute players, and morale effects, while particularly in project planning, there may be cases where a team, after winning, is completely supplemented to other teams. This implies that a single traditional enumerative matrix cannot meet the requirements of the game theory model and that it is necessary to additionally incorporate a compensation matrix based on time series as a foundation, with prior result probabilities serving as judgment criteria. For example, after winning, the coach of the upper-ranked team may subconsciously request outstanding players to substitute in the next match and boost morale, which means that the order of participation may affect the matrix state of the next game, inevitably changing with the occurrence of the previous outcome to adjust the capabilities of the team for the next match. Thus, a corresponding compensation matrix arises, which belongs to a concept of "logic matrix" that generates compensation matrix results through logical judgment formulas[7]. For example, directly judging the first column of the matrix formed by E_1-E_4 can yield the matrix result for whether compensations will be made. However, this primarily relates to the field of computational mathematics, so more complex issues regarding logical matrices will not be discussed here. In fact, through this logical judgment, we can calculate that the compensation matrices for E_1-E_4 are special matrices with the first row (representing the first match) being 0:

$V_{be1}=\begin{bmatrix}0 & 0\\ 0 & 0.7\end{bmatrix}$, compensating on the XA_1 judgment in the second match,

$V_{be2}=\begin{bmatrix}0 & 0\\0 & 0.7\end{bmatrix}$, compensating on the XA_1 judgment in the second match,

$V_{be3}=\begin{bmatrix}0 & 0\\0 & 0.7\end{bmatrix}$, compensating on the XB_1 judgment in the second match,

$V_{be4}=\begin{bmatrix}0 & 0\\0.7 & 0\end{bmatrix}$, compensating on the XA_1 judgment in the second match,

Thus, the final calculation results become:

$$\begin{gathered}E_1=vec(XA_1)-vec(XB_1)=(1,1.2)-(0.8,0.4)=(0.2,0.8)\\E_2=vec(XA_1)-vec(XB_2)=(1,1.2)-(0.4,0.8)=(0.6,0.4)\\E_3=vec(XA_2)-vec(XB_1)=(0.5,1)-(0.8,1.1)=(-0.3,-0.1)\\E_4=vec(XA_2)-vec(XB_2)=(0.5,1.7)-(0.4,0.8)=(0.1,0.9)\end{gathered} \tag{4}$$

Thus, it can be observed that the outcome has changed. In a reasonable configuration that wins the first game, Team B, which is at a strength disadvantage, can achieve a two-game winning streak after obtaining the first victory. Compared to the first outcome, a winner-takes-all situation can also be seen, primarily stemming from the strong impact of substitutions after victories on the game model. This phenomenon can explain many scenarios of comebacks and winner-takes-all situations without merely attributing them to the original strength of the teams. Therefore, this is a very important model, and the verification of how to conduct compensation matrix validation can be implemented through historical data level comparative tests and other verification methods[8]. Here, based on time series considerations for the logical matrix judgment of the next matrix, a certain degree of model evolution explanation is provided.

Nature's games and evolution are complex, but regardless, it is still possible to restore evolutionary logic through mathematical model architecture. Here, some special matrix architectures are proposed for computer simulations of game-related models, such as "mutually exclusive probability matrices" and "logic matrices." Although these matrix architectures do not exist in traditional mathematics, it is necessary to develop innovative underlying matrix computation models to address large-scale data calculations in response to the complexities of reality.

3 CONCLUSION

Matrix simulation holds a significant position for quantification in complex game theory. The patterns of special matrices in data models can adjust parameters and perform complex function transformations. Through computer enumeration of time series operations, more diversified matrix combinations and analyses can be derived, presenting enormous potential and applications across engineering projects, physical numerical simulations, and economic management analysis.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] NEUMANN, MORGENSTERN, Theory of Games and Economic Behavior, Princeton University Press, 1944.
[2] Ren Fangguo. He JiaqI. On the Diagonalizability of Matrices. Advanced Mathematics Research, 2023, 26(01): 1-3+6.
[3] Bai Jianchao, Chen Honghua. Research on Change Detection Methods for SAR Images Based on SVM. Advanced Mathematics Research, 2023, 26(03): 76-82.
[4] John Nash. Annals of Mathematics, 1951, 54, 9(2): 286-295.
[5] Xu Huafeng, Li Lingling, Fang Zhigeng. Gray Double Matrix Game Model and Its Equilibrium Solution. Mathematics Practice and Understanding, 2012, 42(10): 174-179.
[6] Liu Zhaoxia, Shao Feng, Li Ye. Feature Point Matching Algorithm Based on Multi-scale Local Structural Constraints. Mathematics Practice and Understanding, 2023, 53(10): 160-168.
[7] Wang Min, He Tengsong, Peng Ding. Comparison of Fuzzy Time Series Prediction Models Based on Different FCM Clustering Algorithms. Mathematics Practice and Understanding, 2024, 54(05): 128-140.
[8] Jiang Cuicui, Luo Wanchun. A Brief Discussion on the Construction and Application of the GM(1,1) Model in Mathematical Modeling. Advanced Mathematics Research, 2023, 26(04): 31-33.