Imperial College London

25 January 2021

Dynamics of Games Coursework

Colonel Blotto Games and an application to U.S. Elections

Matteo Mario Di Venti

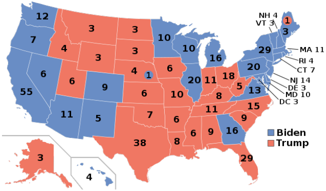
Imperial College London

Introduction

- A 'Colonel Blotto' game is a type of two person game in which the two players must distribute a finite predetermined quantity of resources over a finite number of objects denoted as 'battlefields'.
- When the resources are allocated to each battlefield, they are confronted and the player that allocated most resources 'wins' the battlefield.
- The payoff for each player is the number of battlefields won. The winner of the game is whoever has won the most battlefields, with a draw if all players have the same number.
- Significant improvement to the problem of determining Nash Equilibria of the game was reached thanks to Regret-matching algorithm by Hart and Mas-Colell.

An application to U.S. Elections

- Colonel Blotto games have been used extensively in strategical environments.
- In the following section, the Colonel Blotto setting will be used to model strategies for election funds allocation in the key 'Swing states' of 2020 U.S. Presidential Elections.
- Through the use of regret matching algorithm, the best mixed strategies for both candidates will be determined.



Imperial College London

Key challenges:

- Modelling the different battlefields with different values

Imperial College London

Key challenges:

- Modelling the different battlefields with different values
- Given the initial condition, set the correct number of "electors" each party needs to win to elect the president

Key challenges:

- Modelling the different battlefields with different values
- Given the initial condition, set the correct number of "electors" each party needs to win to elect the president
- Resolve battlefields draws in favour of one party or the other

Imperial College London

Key challenges:

- Modelling the different battlefields with different values
- Given the initial condition, set the correct number of "electors" each party needs to win to elect the president
- Resolve battlefields draws in favour of one party or the other
- Model the payoff accurately: two options zero-sum or costant-sum

Key challenges:

- Modelling the different battlefields with different values
- Given the initial condition, set the correct number of "electors" each party needs to win to elect the president
- Resolve battlefields draws in favour of one party or the other
- Model the payoff accurately: two options zero-sum or costant-sum
- Timing vs complexity tradeoff

Key challenges:

- Modelling the different battlefields with different values
- Given the initial condition, set the correct number of "electors" each party needs to win to elect the president
- Resolve battlefields draws in favour of one party or the other
- Model the payoff accurately: two options zero-sum or costant-sum
- Timing vs complexity tradeoff

Main solutions:

- A vector of battlefield values is dotted with the winner of each battlefield

Main solutions:

- A vector of battlefield values is dotted with the winner of each battlefield
- A 'difference from 270' system is employed

Main solutions:

- A vector of battlefield values is dotted with the winner of each battlefield
- A 'difference from 270' system is employed
- Introduce a status quo resolution

Main solutions:

- A vector of battlefield values is dotted with the winner of each battlefield
- A 'difference from 270' system is employed
- Introduce a status quo resolution
- Compare and contrast the strategies of the two models

Main solutions:

- A vector of battlefield values is dotted with the winner of each battlefield
- A 'difference from 270' system is employed
- Introduce a status quo resolution
- Compare and contrast the strategies of the two models
- 5 swing states selected and number of resources is selected to be 6

Imperial College London

Main solutions:

- A vector of battlefield values is dotted with the winner of each battlefield
- A 'difference from 270' system is employed
- Introduce a status quo resolution
- Compare and contrast the strategies of the two models
- 5 swing states selected and number of resources is selected to be 6

Assumptions for the model example

Table 1: Number of members of Electoral College for each Swing State

Arizona	15
Georgia	16
Pennsylvania	20
North Carolina	15
Wisconsin	10

Table 2: Status Quo of Swing States

Arizona	Trump
Georgia	Trump
Pennsylvania	Biden
North Carolina	Trump
Wisconsin	Biden

Payoff Matrices Explained

$$payoff_B(i,j) = sign(\delta + sign((q + 2sign(\nu_i - \nu_j)) \cdot \mu))$$

- takes the sign of the difference between Biden allocation ν_i and Trump allocation ν_j to find the winner in each battlefield. Zero values appear here when there are battlefield draws.

Payoff Matrices Explained

$$payoff_B(i,j) = sign(\delta + sign((q + 2sign(\nu_i - \nu_j)) \cdot \mu))$$

- takes the sign of the difference between Biden allocation ν_i and Trump allocation ν_j to find the winner in each battlefield. Zero values appear here when there are battlefield draws.
- The status quo vector q is added to two times the previous calculated vector to resolve draws. Once again, each element of the vector is mapped to $+1$ for Biden victory and -1 for Trump's.

Payoff Matrices Explained

$$payoff_B(i,j) = sign(\delta + sign((q + 2sign(\nu_i - \nu_j)) \cdot \mu))$$

- takes the sign of the difference between Biden allocation ν_i and Trump allocation ν_j to find the winner in each battlefield. Zero values appear here when there are battlefield draws.
- The status quo vector q is added to two times the previous calculated vector to resolve draws. Once again, each element of the vector is mapped to $+1$ for Biden victory and -1 for Trump's.

Payoff Matrices Explained

$$payoff_B(i,j) = sign(\delta + sign((q + 2sign(\nu_i - \nu_j)) \cdot \mu))$$

- The vector of battlefield results is dotted with the vector of battlefield values μ to determine the advantage (or disadvantage) Biden gained in the swing states.

Payoff Matrices Explained

$$payoff_B(i,j) = sign(\delta + sign((q + 2sign(\nu_i - \nu_j)) \cdot \mu))$$

- The vector of battlefield results is dotted with the vector of battlefield values μ to determine the advantage (or disadvantage) Biden gained in the swing states.
- The advantage is then added to the difference between the predetermined states results δ. If the sum is positive, Biden wins (payoff $+1$), if negative, Trump wins (payoff -1), if it is 0 then a draw is reached (payoff 0).

Imperial College London

Payoff Matrices Explained

$$payoff_B(i,j) = sign(\delta + sign((q + 2sign(\nu_i - \nu_j)) \cdot \mu))$$

- The vector of battlefield results is dotted with the vector of battlefield values μ to determine the advantage (or disadvantage) Biden gained in the swing states.
- The advantage is then added to the difference between the predetermined states results δ. If the sum is positive, Biden wins (payoff $+1$), if negative, Trump wins (payoff -1), if it is 0 then a draw is reached (payoff 0).

A few notes on payoff

$$payoff_T(i,j) = sign(-\delta + sign(-q + 2sign(\nu_i - \nu_j) \cdot \mu))$$

- Trump payoff is not the transpose of Biden's due to the introduction of the status quo vector. Trump's payoff formula makes use of a negative status quo vector and negative predetermined difference

A few notes on payoff

$$payoff_T(i,j) = sign(-\delta + sign(-q + 2sign(\nu_i - \nu_j) \cdot \mu))$$

- Trump payoff is not the transpose of Biden's due to the introduction of the status quo vector. Trump's payoff formula makes use of a negative status quo vector and negative predetermined difference
- In the case of the more 'relaxed' elections, the last sign functions is replaced by a conditional that maps to payoff $+1$ in case of win and payoff 0 in case of draw or defeat.

A few notes on payoff

$$payoff_T(i,j) = sign(-\delta + sign(-q + 2sign(\nu_i - \nu_j) \cdot \mu))$$

- Trump payoff is not the transpose of Biden's due to the introduction of the status quo vector. Trump's payoff formula makes use of a negative status quo vector and negative predetermined difference
- In the case of the more 'relaxed' elections, the last sign functions is replaced by a conditional that maps to payoff $+1$ in case of win and payoff 0 in case of draw or defeat.

Imperial College London

The results for the 'harsh' competition elections

Strat.	**Trump**	**Biden**	**AZ**	**GA**	**PN**	**NC**	**Wi**
121	0.044757	1.47E-05	1	2	0	1	2
156	0.046759	1.74E-05	2	1	0	1	2
107	0.044084	3.10E-05	1	1	0	2	2
15	0.022954	0.007103	0	0	2	2	2
149	0.022426	0.007139	2	0	2	0	2
58	0.022901	0.007353	0	2	2	0	2
167	4.76E-09	0.02515	2	2	0	2	0
166	0.018924	0.03401	2	2	0	1	1
169	0.017848	0.034252	2	2	1	1	0
126	0.020346	0.034254	1	2	1	2	0
157	0.019412	0.034594	2	1	0	2	1
122	0.021662	0.034596	1	2	0	2	1
161	0.018719	0.034826	2	1	1	2	0
160	5.68E-06	0.036156	2	1	1	1	1
125	1.27E-06	0.036307	1	2	1	1	1
112	3.43E-08	0.036682	1	1	1	2	1

The results for the 'relaxed' competition elections

	Trump	Biden	AZ	GA	PN	NC	WI
101	0.102891	1.20E-05	1	0	3	1	1
118	0.096829	5.86E-06	1	1	3	0	1
162	0.078647	6.23E-05	2	1	1	2	0
38	0.06653	7.35E-07	0	1	1	3	1
153	0.060524	6.58E-06	2	0	3	0	1
170	0.048435	0.000103	2	2	1	1	0
98	0.042363	3.55E-05	1	0	2	2	1
128	0.04236	2.83E-05	1	2	2	0	1
40	4.23E-02	1.43E-08	0	1	2	0	3
36	0.03629	1.43E-08	0	1	1	1	3
35	0.036284	1.43E-08	0	1	1	0	4
55	0.036275	3.97E-08	0	2	1	0	3
45	0.030322	2.33E-06	0	1	3	1	1
160	0.030261	4.99E-06	2	1	1	0	2
62	0.024201	1.39E-06	0	2	3	0	1
119	0.018227	0.998063	1	1	3	1	0

Imperial College London

Thank you for the attention