Predicting Cellular Automata

Jameson Toole

Massachusetts Institute of Technology
Cambridge, MA
jltoole@mit.edu

Scott E. Page

University of Michigan
Ann Arbor, MI

We explore the ability of a locally informed individual agent to predict the future state of a cell in systems of varying degrees of complexity using Wolfram's one-dimensional binary cellular automata. We then compare the agent's performance to that of two small groups of agents voting by majority rule. We find stable rules (class I) to be highly predictable, and most complex (class IV) and chaotic rules (class III) to be unpredictable. However, we find rules that produce regular patterns (class II) vary widely in their predictability. We then show that the predictability of a class II rule depends on whether it produces vertical or horizontal patterns. We comment on the implications of our findings for the limitations of collective wisdom in complex environments.

1. Introduction

Market economies and democratic political systems rely on collections of individuals to make accurate or nearly accurate predictions about the value of variables in the future. Explanations of aggregate predictive success take two basic forms. Social science explanations tend to rely on a statistical framework in which independent errors cancel (for a survey see [1]; see also [2, 3]). Computer science and statistical models rely on a logic built on diverse feature spaces [4]. These two approaches can be linked by showing that if agents rely on diverse predictive models of binary outcomes then the resulting errors will be negatively correlated [5]. In the statistical approach to prediction, the probability that a signal is correct captures the difficulty of the predictive task. Yet, given the assumptions of the models, if each individual is correct more than half of the time, then the aggregate forecast will become perfectly accurate as the number of predictors becomes large. This statistical result runs counter to experience. Some processes are very difficult to predict. In [6] it is shown that experts fare only slightly better than random guesses on complex policy predictions.

In this paper, we explore the relationship between the complexity of a process and the ability of a locally informed agent to predict the future state of that process. We then compare the forecast ability of a single agent to small groups of agents. We presume that more complex phenomena will be harder to predict. To investigate how complexity influences predictability, we sweep over all 256 possible one-dimensional nearest neighbor rules [7]. These rules have been categorized as either stable (class I), periodic (class II), chaotic (class III), or complex (class IV).

We first consider the ability of a locally informed agent to predict the future state of a single cell. This agent knows the initial state of the cell and the states of the two neighboring cells. Its task is to predict the state of the center cell a fixed number of steps in the future. We then add other agents who also have local knowledge. Two of these agents are informed about neighboring cells, and two of these agents know the initial states of random cells. We find that for complex predictive tasks, the groups of agents cannot predict any more accurately, on average, than the individual agent. This occurs because their predictions are not independent of the individual agent's nor of one another's predictions and because these other agents are not very accurate.

Our analysis of predictability as a function of process complexity yields one very surprising result. We find that three classes—ordered, complex, and chaotic—sort as we expected. Most chaotic rules cannot be predicted with more than 50% accuracy. Complex rules also prove difficult to predict, while stable rules are predicted with nearly perfect accuracy. Performance on periodic rules, however, was not what we expected. We found that performance runs the gamut from nearly perfect to no better than random. By inspection of the various rules in class II, we can explain this variation. Some class II rules produce vertical patterns. Under these rules, the initial local information produces an ordered sequence. Considering the rule that switches the state of the cell, the rule can be predicted with 100% accuracy with only local information. Contrast this to the rule that copies the state of the cell on the left. This rule produces a diagonal pattern, yet it cannot be predicted with local information. To know the state of a cell in 100 steps requires knowing the initial state of the cell 100 sites to the left.

2. The Model

We construct a string of binary cellular automata of length L with periodic boundary conditions (creating a cylinder) and random initial conditions [7]. Each cell updates its state as a function of its state and the state of its two neighboring cells. Therefore, 256 rules exist. For each of these, we test the ability to predict the state of a cell K steps in the future, knowing only the initial state of the cell and the initial states of its two neighbors.

We first consider a single agent who constructs a predictive model. This agent knows only the initial state of a single cell as well as the states of the two neighboring cells. In other words, this agent has the same information as does the cell itself. Following standard practice for the construction of predictive models, we create a learning stage in which the agent develops its model, and then create a testing stage in which we evaluate the model's accuracy.

2.1 The Learning Stage

During the learning stage, the agent keeps a tally of outcomes given its initial state. Over a number of training runs, these tallies accumulate, allowing the agents to predict final states based on frequency distributions of outcomes. Recall that the agent looks at the initial state of a single cell as well as the states of the two cells on its left and right. These three sites create a set of eight possible initial states.

The agent follows this procedure in the learning stage: note the cell's and its neighbors' initial states, then keep tallies of the cell's state in step K (either 0 or 1). After the learning stage is complete, the agent's prediction given the initial states corresponds to the final state with the most tallies.

For example, consider the following partial data from 1000 training runs. The first column denotes the initial states of the cell and its neighbors. The second and third columns correspond to the frequencies of a cell starting from that initial state, being in states 0 and 1 at step K. The agent's predictions, which correspond to the more likely outcome, appear in the rightmost column.

Initial	Outcome after K Periods		Best
State	0	1	Prediction
000	63	75	1
001	82	75	0
010	47	101	1
⋮	⋮	⋮	⋮

Thus, when asked to predict the future state given an initial state of 000, the agent would choose 1 because it was the more frequent outcome during the learning phase. If it saw the initial state 001, it would predict 0 for the same reason.

We next consider cases in which we include predictions by the agents centered on the cell to the left and right of the cell of the first agent. For ease of explanation, we refer to this as the *central cell*. In these cases as well, the agents also look at the initial states of their two neighboring cells. However, these agents do not predict the state of the cell on which they are centered but of the central cell. To test the accuracy of the three predictors—the agent and its two neighbors—we rely on simple majority rule.

Finally, we also include agents who look at the initial state of two random cells as well as of the central cell. The random cells chosen remain fixed throughout the learning stage. These agents' predictive models consider the eight possible initial states for the three cells and then form a predictive model based on the frequency of outcomes during a training stage. These agents using random predictors can be combined with the other agents to give five total predictors. We define the collective prediction to be the majority prediction.

2.2 The Testing Stage

At the completion of the learning stage, each of the agents has a *predictive rule*. These predictive rules map the initial state into a predicted outcome. To assess the accuracy of these predictions, we create M random initial conditions. All L cells iterate for K steps according to whichever of the 256 rules is being studied. The state of the central cell is then compared to the prediction.

We define the *accuracy* of an agent or a collection of agents using majority rule to be the percentage of correct predictions.

To summarize, for each of the 256 rules, we perform the following steps.

- *Step 1*. Create N random initial conditions and evolve the automaton K steps, keeping tallies of outcomes.
- *Step 2*. From the tallies, make predictions by selecting the majority outcome.
- *Step 3*. Create M additional random initial conditions and evolve the automaton K steps.
- *Step 4*. Compare predictions from the training stage to actual outcomes from testing and compute accuracy.

There is a concern that our testing stage is unfair to agents attempting to predict future states because we randomly re-initialize automata before testing. Our goal, however, is to test an agent's ability to learn rules based on multiple outcomes of the same process, rather than learning from a single instance of a process. Thus, we do not bias our results by randomizing initial test states rather than continuing the evolution of training states. Furthermore, for the vast majority of rules, the automata reach a steady state (or steady distribution) before the K^{th} step. If we attempt to test automata while initializing them in their steady state (or distribution), we would expect that their predictive power would simply be the predictability of whatever distribution of states the rule produces. For rules that do not reach a steady state quickly and are still in a random configuration after K steps, continuing to evolve automata from this state is no different than re-randomizing. As a check, we have implemented both re-randomization and continued evolution algorithms and find that they are in agreement under our measures of accuracy.

3. Results

We present our results in three parts. We first present analytic results for rule 232, which is the majority rule. We calculate the expected accuracy for the single agent located at the central cell as well as for the group of three agents that includes the two agents on either side of the central cell. We then examine all 256 rules computationally. Our analytic results provide a check on our computational analysis as well as insights into the difficulties of making accurate predictions given only local information.

The puzzle that arises from our computational results concerns ordered, or what are called class II, rules. Some of these rules are as difficult to predict as chaotic rules (class IV). In the third part, we analyze rule 170, otherwise known as "pass to the left." This rule creates a pattern so it belongs to class II, but the long run future state of the central cell appears random to our locally informed agents. We show why that is the case analytically.

3.1 Analytic Results for Rule 232: Majority Rule

In rule 232, the cell looks at its state and the state of the two neighboring cells and matches the state of the majority. We denote the central cell by x and the two neighboring cells by w and y. It can be written as follows:

Rule 232								
$w^t\, x^t\, y^t$	000	001	010	011	100	101	110	111
x^{t+1}	0	0	0	1	0	1	1	1

In six of the eight initial states, the central cell and one of its neighbors are in the same state. In those cases, the state of the central cell and that neighbor remain fixed in that state forever. In those cases, the predictive rule for the agent located at the central cell will be to predict an unchanging state. That rule will be correct 100% of the time.

In the two other cases 010 and 101, the eventual state of the central cell depends on the states of its neighbors. To compute the optimal prediction and its accuracy in these cases, we need to compute probabilities of neighboring states. Note that by symmetry, we need only consider the case where x and its neighbors are in states 010. We construct the following notation. Let ℓ_i be the i^{th} cell to the left of 010 and r_i be the i^{th} cell to the right. Thus, we can write the region around 010 as $\ell_3\,\ell_2\,\ell_1\,010\,r_1\,r_2\,r_3$. Consider first the case where $r_1 = 0$. By convention, we let a question mark ? denote an indeterminate state. The states of the automaton iterate as follows:

$$\begin{array}{ccccccccc}
\ell_3 & \ell_2 & \ell_1 & 0 & 1 & 0 & 0 & r_2 & r_3 \\
\ell_3 & \ell_2 & \ell_1 & ? & 0 & 0 & 0 & r_2 & r_3 \\
\ell_3 & \ell_2 & \ell_1 & ? & 0 & 0 & 0 & r_2 & r_3
\end{array}$$

By symmetry, if $\ell_1 = 0$, x will also be in state 0. Therefore, the only case left to consider is where $\ell_1 = r_1 = 1$. Suppose in addition that $r_2 = 1$. The states iterate as follows:

$$\begin{array}{ccccccccc} \ell_3 & \ell_2 & 1 & 0 & 1 & 0 & 1 & 1 & r_3 \\ \ell_3 & \ell_2 & ? & 1 & 0 & 1 & 1 & 1 & r_3 \\ \ell_3 & \ell_2 & ? & ? & 1 & 1 & 1 & 1 & r_3 \\ \ell_3 & \ell_2 & ? & ? & 1 & 1 & 1 & 1 & r_3 \end{array}$$

It follows then that if either r_2 or ℓ_2 is in state 1, then the central cell will be in state 1 in step K.

Given these calculations, we can determine the probability distribution over the state of the central cell if it and its neighbors start in states 010. From the preceding, unless $r_1 = \ell_1 = 1$, then x will be in state 0. Therefore, with probability 3 / 4, it locks into state 0 in one step. With probability 1 / 4, it does not lock into state 0. In those cases, $r_1 = \ell_1 = 1$. And, as found earlier, with probability 3 / 4, x will lock into state 1. It follows that the probability that x ends up in state 0 with initial condition 010 is given by the following infinite sum:

$$\Pr(x = 0 \mid w\,x\,y = 010) = \frac{3}{4} + \frac{1}{4}\frac{1}{4}\left[\frac{3}{4} + \frac{1}{4}\frac{1}{4}\left[\frac{3}{4} + \frac{1}{4}\frac{1}{4} \cdots\right]\right].$$

This expression takes the form $p + q\,p + q^2\,p^2 + q^3\,p^3 + \cdots$. A straightforward calculation gives that the value equals

$$\frac{1}{\frac{61}{64}} + \frac{3}{4} - 1 = \frac{64}{61} - \frac{1}{4} = 0.799.$$

Given this calculation, we can characterize the agent's predictions in the case where the training set is infinitely large as follows:

Rule 232: Optimal Predictions at x and Accuracy								
$w\,x\,y$	000	001	010	011	100	101	110	111
Prediction	0	0	0	1	0	1	1	1
Accuracy	1.0	1.0	0.8	1.0	1.0	0.8	1.0	1.0

Summing over all cases gives that, on average, the agent's accuracy equals 95%.

3.1.1 Predictions by Agents at Neighboring Cells

We next consider the predictions by the two agents on either side of the central cell. By symmetry, we need only consider the neighbor on the left, denoted by w. If w and x have the same initial state, then they remain in that state forever. In those four cases, the agent at w can predict the state of cell x with 100% accuracy.

This leaves the other four initial states centered at w denoted by 001, 110, 101, and 010. By symmetry these reduce to two cases. First, consider the initial state 001. To determine the future state of cell x, we need to know the state of the cell centered on y. If $y = 1$, then by construction x will be in state 1 forever. Similarly, if $y = 0$, then $x = 0$ forever. Therefore, the prediction by the agent at w can be correct only 50% of the time in these two initial states.

Next, consider the initial state 101. To calculate the future state of the central cell, we need to include the states for both y and r_1. We can write the initial states of these five cells as $101\,y\,r_1$. If $y = 1$, then $x = 1$ forever. If $y = 0$, then the value of x will depend on r_1. If $r_1 = 0$, then $x = 0$, but if $r_1 = 1$, then the value will depend on the neighbors of r_1. Therefore, the probability that x will end up in state 1 given $\ell_1\,w\,x = 101$ equals

$$\Pr(x = 1 \mid \ell_1\,w\,x = 010) = \frac{1}{2} + \frac{1}{4}\,\frac{1}{4}\bigg[\frac{3}{4} + \frac{1}{4}\,\frac{1}{4}\bigg[\frac{3}{4} + \frac{1}{4}\,\frac{1}{4}\cdots\bigg]\bigg]\bigg]$$

which by a calculation similar to that in Section 3.1 equals 0.549. We can now write the optimal predictions by an agent at cell w for the final state of cell x and the accuracy of those predictions.

Rule 232: Optimal Predictions at w and Accuracy								
$w\,x\,y$	000	001	010	011	100	101	110	111
Prediction	0	0	0,1	1	0	1	0,1	1
Accuracy	1.0	0.5	0.55	1.0	1.0	0.55	0.5	1.0

The average accuracy of an agent at w equals 76.2%. By symmetry, that also equals the accuracy of an agent at y. We can now compare the accuracy of the individual agent located at the central cell to the accuracy of the group of three agents. Recall that we assume the three agents vote, and the prediction is determined by majority rule.

By symmetry, we need only consider the cases where $x = 0$. There exist 16 cases to consider as shown in the following table. We denote the cases in which an agent's prediction is accurate only half the time by H. We let G denote the majority prediction with two random predictors and one fixed predictor of 0.

Rule 232: Comparison between x and Majority Rule of w, x, and y					
$w\,x\,y$	Prediction of x	Accuracy	Predictions of $w\,x\,y$	Majority	Accuracy
00000	0	1.0	0 0 0	0	1.00
00001	0	1.0	0 0 0	0	1.00
00010	0	1.0	0 0 0	0	1.00
00011	0	1.0	0 0 H	0	1.00
01000	0	1.0	0 0 0	0	1.00
01001	0	1.0	0 0 0	0	1.00
01010	1	0.2	0 1 0	0	0.80
01011	1	1.0	0 1 H	H	0.50
10000	0	1.0	0 0 0	0	1.00
10001	0	1.0	0 0 0	0	1.00
10010	0	1.0	0 0 0	0	1.00
10011	0	1.0	0 0 H	0	1.00
11000	0	1.0	H 0 0	0	1.00
11001	0	1.0	H 0 0	0	1.00
11010	1	1.0	H 1 0	H	0.50
11011	1	1.0	H 1 H	G	0.75

A calculation yields that the group of three predictors has an accuracy of 91%. Recall from Section 3.1 that the single agent located at the central cell has an accuracy of 95%. The group is less accurate than the individual. This result occurs for two reasons. First, the agents located at w and y are not nearly as accurate as the agent located at the central cell. Second, their predictions are not independent of the central agent. If all three predictions were independent then the group of three would be correct approximately 94% of the time.

3.2 Computational Results

We now describe results from computational experiments on all 256 rules relying on automata having 20 sites and periodic boundary conditions. For each of the 256 rules, automata undergo a learning stage of 1000 steps. Automata were trained and tested on the prediction of their state, $K = 53$ steps in the future. (A prime number was chosen to avoid any periodicities that may affect prediction results. For good measure, $K = 10$, 20, 25, 40, and 100 were also tested and found to yield similar results in almost all cases.) Once the agents had been trained, we computed their accuracy during a testing phase consisting of 500 trials.

3.2.1 Predictability of Automaton by a Single Agent

We first show our findings for the accuracy of the single agent located at the central cell. Figure 1 shows a sorted distribution of this agent's accuracy.

Two features stand out. First, some rules can be predicted accurately 100% of the time while in other cases, learning does not help prediction at all (guessing randomly guarantees ability of 50%). Examples of the former would be rules 0 and 255 which map every initial state to all 0s and all 1s, respectively. These rules can be predicted perfectly. The majority of rules lie on a continuum of predictability. Though the graph reveals some minor discontinuities, the plot does not reveal a natural partition of the 256 rules into Wolfram's four classes. Therefore, the categories do not map neatly to predictability.

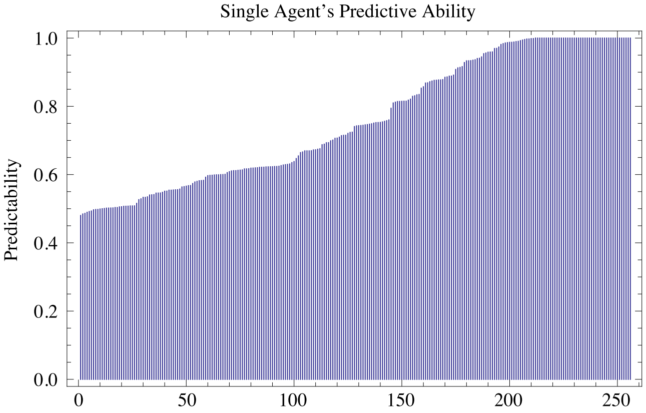


Figure 1. A single agent's ability to predict its future state given the 256 rules. Rule predictability can range from being no better than a fair coin flip to 100% accuracy depending on the dynamics of the rule. The *x* axis (rule number) does not correspond to Wolfram's numbering.

To see why the categories do not map, we return to Wolfram's classification [7] which classifies the rules as follows.

- *Class I.* Almost all initial conditions lead to exactly the same uniform final state.
- *Class II.* There are many different possible final states, but all of them consist just of a certain set of simple structures that either remain the same forever or repeat every few steps.
- *Class III.* Nearly all initial conditions end in a random or chaotic final state.
- *Class IV.* Final states involve a mixture of order and randomness. Simple structures move and interact in complex ways.

We give our classification of rules used in Appendix A. We have not found a complete listing elsewhere.

Figure 2 shows the sorted ability of the individual agent to make accurate predictions by class of rule. From this data we find that three of Wolfram's classes are informative of a rule's predictability while one is not. Class I (rules that converge to homogeneous steady states) are predictable with very high accuracy while the random and complex rules falling in classes III and IV are nearly impossible to accurately predict. For the intermediate class II rules, however, there is a large spectrum of ability. Some class II rules appear easy to predict while others fair worse than some class III rules. These results suggest that the regular patterns characterizing class II rules are not informative to a rule's predictability and that further classification refinement is needed for a better description.

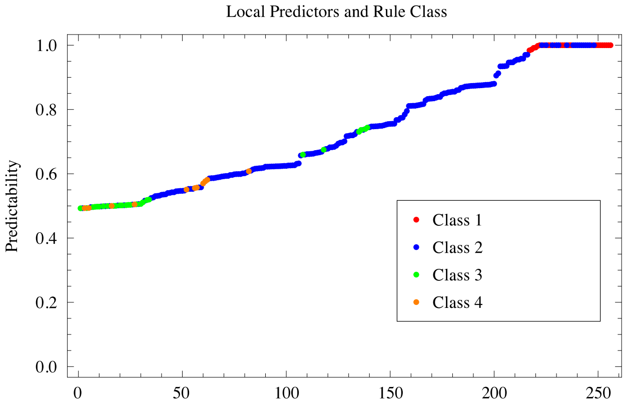


Figure 2. Using local predictors sorted according to fitness, we color code rules based on the class assigned by Wolfram. While classes I, III, and IV prove to be informative, class II rules show huge variation in their ability to be predicted.

These visual clues can be shown statistically. The following table gives the mean accuracy for the agent located at the central cell for each class of rules. The standard deviation is given in parentheses.

	Class I	Class II	Class III	Class IV
Agent at x	0.998 (0.004)	0.759 (0.180)	0.545 (0.0927)	0.554 (0.061)

Notice that complex rules are, on average, just as difficult to predict as chaotic rules for a single agent. Note also enormous variance in the predictability of the class II rules.

3.2.2 Individuals versus Groups

We next compare the ability of the single agent to that of small groups. Our main finding is that the small groups are not much more accurate. A statistical analysis, given in the following table (standard deviation in parentheses), shows no meaningful difference in accuracy for any of the classes. Were we to ramp up our sample sizes, we might gain statistical significance for some of these results, but the magnitude of the differences is small—most often much less than 1%.

Class I	Class II	Class III	Class IV
Agent at x			
0.998 (0.005)	0.733 (0.153)	0.551 (0.0923)	0.545 (0.040)
Agent at x Plus Local			
0.997 (0.006)	0.739 (0.153)	0.550 (0.0932)	0.543 (0.039)
Agent at x Plus Random			
0.998 (0.004)	0.720 (0.123)	0.550 (0.0920)	0.539 (0.037)
All Five Agents			
0.997 (0.005)	0.723 (0.132)	0.551 (0.0920)	0.547 (0.040)

This aggregate data demonstrates that, on average, adding predictors does not help. That is true even for the rules in classes II and IV. We found this to be rather surprising.

This aggregate data masks differences in the predictability of specific rules. Figure 3 displays the variance in prediction ability across all combinations of predictors. For most rules, we find that this variance is very low. In those cases where predictability does vary, different combinations of predictors give better predictability. Note that this has to be the case given that average accuracy is approximately the same for all combinations of predictors.

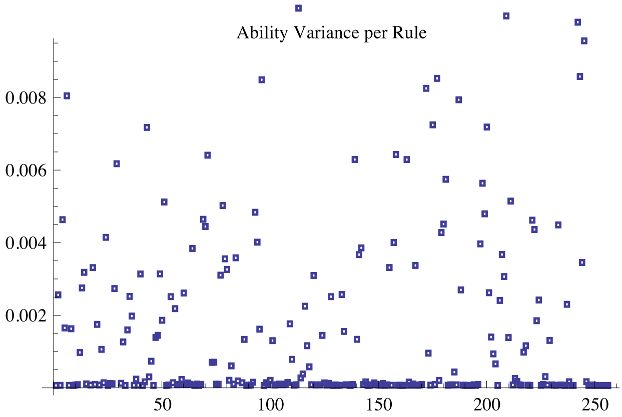


Figure 3. The variance in ability between four combinations of predictors reveals that for many rules, all predictors perform equally.

Detailed analysis of specific rules, such as the one we performed for rule 232, can reveal why adding local predictors for some rules increases or decreases predictability, but no general patterns exist. The data shows that adding local predictors, random predictors, or both over all rules does not help with overall predictability. This finding stands in sharp contrast to statistical results that show the value of adding more predictors (Figure 4).

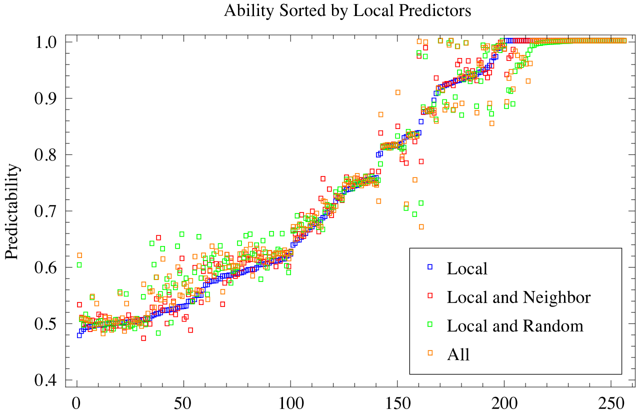


Figure 4. Various combinations of predictors are sorted by the ability of local predictors. While the x axis (rule number) does not correspond to Wolfram's numbering, all predictors can be easily compared to the use of only local predictors.

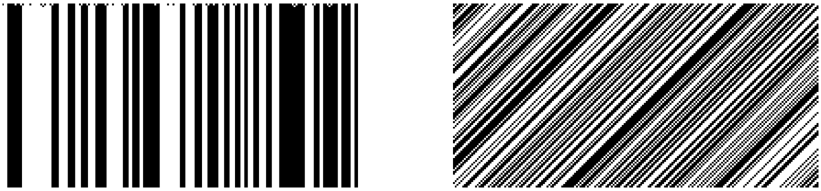
3.3 Class II Rules

We now present an explanation for the variation of the predictability of class II rules. We show that class II rules can be separated into two groups: those displaying vertical patterns in time, and those that are horizontal. The former are easy to predict. The latter are not.

Vertical temporal patterns form under rules where evolution can lock automata into stationary states, creating vertical stripes during their evolution (Figure 5). In contrast, some class II rules pass bits to the left or right, creating diagonal stripes in time. From the perspective of a single automaton, we will show that vertical patterns provide an opportunity to learn dynamics and make accurate predictions, while horizontal patterns make information gathering much more difficult. Finally, we show that accurately predicting the future given each of these patterns requires automata acquiring different types of information.

3.3.1 Rule 170: Pass to the Left

As shown in Figure 5, Rule 170 generates horizontal patterns in time. These horizontal patterns differ from vertical patterns in that no single cell locks into a stationary state. From an individual cell's point of view, vertical patterns correspond to a world that settles to a predictable equilibrium state. Horizontal rules on the other hand would seem random, as tomorrow may never be the same as today.



(a) Vertical Pattern - Rule 232

(b) Horizontal Pattern - Rule 170

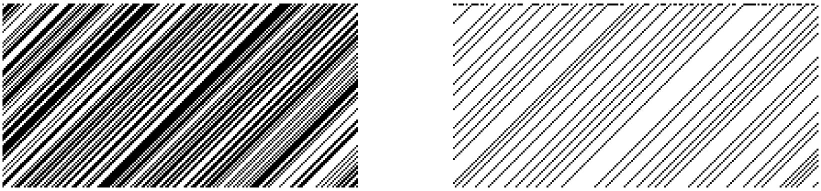
Figure 5. A pair of class II rules is shown. Rule 232 displays a vertical pattern where individual cells, starting from a random initial condition, lock into stationary states. Rule 170, by contrast, generates patterns that continually shift to the left, never settling into a stationary state.

This randomness makes a prediction based on an initial state difficult and often unsuccessful for rules that generate horizontal patterns. There is, however, some useful information in these patterns. While individual stationary states are not reached, the distribution of bits (the number of 0s and 1s) does become stationary in horizontal patterns.

We can see this by considering rule 170, informally named "pass to the left," as shown in the following table. This rule simply tells each cell to take on the state of the cell to their left in the next step. For any random initial state, half of the automata should be in state 0, with the other half in state 1. Under rule 170, these initial bits simply rotate around the torus.

Rule 170								
$w^t\ x^t\ y^t$	000	001	010	011	100	101	110	111
x^{t+1}	0	1	0	1	0	1	0	1

Though individual cells cycle from 0 to 1 as the pattern rotates, this rule preserves the distribution of bits. There are always the same number of 0s and 1s as in the random initial state. Other rules, though also displaying horizontal patterns, alter the distribution of bits, introducing more of one state. For example, rule 2 visibly results in patterns favoring more 0 bits as large strings of 1s flip to 0s (Figure 6).



(a) Rule 170

(b) Rule 2

Figure 6. Although horizontal patterns never result in individual stationary states, they do create different equilibrium distributions of bits.

Given rule 170, an agent trying to predict the central cell's state in step K learns nothing of value from the cell's initial condition. The agent should do no better than 50% accuracy.

Alternatively, consider rule 2, 00000010. Under this rule, there is only one initial condition (001) that can result in an "on" state in the next round. Because of this, the equilibrium distribution has many more 0s than 1s. Because automata are initialized randomly, this occurs with probability $1/8$. Thus, we expect $1/8$ to be the fraction of 1s in our equilibrium distribution. Knowing this, any cell will correctly predict its outcome 87.5% ($7/8$) of the time by always guessing 0.

We find near perfect agreement between these analytic results and those obtained through computation. For rule 170, we find individual cells can correctly guess their final state with accuracy 50 ± 1 %, while rule 2 allows accuracy of 87.5 ± 1 %.

In most cases, we expect the lack of stationary states for individual cells to impede predictive ability. Many of the equilibrium distributions of horizontal rules are complex and arise from many non-trivial initial states. For this reason we expect class II rules that generate horizontal patterns to have relatively low predictability compared to rules generating vertical patterns. Figure 7 confirms our expectations.

Finally, we note that in cases with horizontal patterns, each cell's neighbors are in the same situation and thus cannot provide any useful information to help with prediction. We find that rules with horizontal patterns display the same levels of predictability regardless of the specific combination of predictors (e.g., neighbor, random, or both), whereas for vertical patterns, neighbors may provide some information, good or bad.

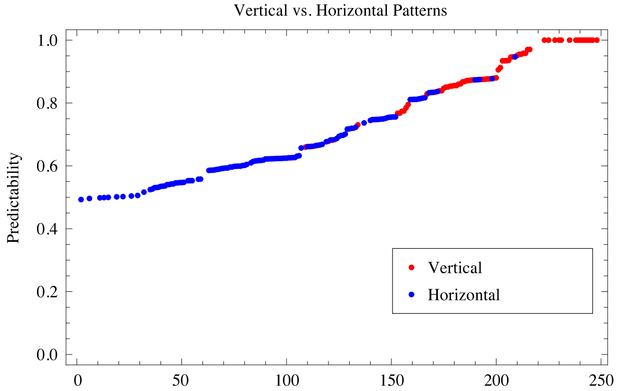


Figure 7. Rules are sorted based on predictability fulfilling our expectations that rules generating vertical patterns are more easily predicted using inductive reasoning than horizontal patterns.

4. Discussion

In this project, we tested whether an individual agent could predict the future state of a dynamic process using local information. We considered a classic set of 256 dynamic processes that have been categorized according to the type of dynamics they create. We then compared an individual agent to small groups of agents who had slightly different local information. These agents used predictive models that they created inductively. During a training period, our agents observed outcomes K steps in the future as well as initial states. The accuracy of their resulting predictive models was then calculated during a testing phase.

We found three main results. First, classifications of cellular automata rules based on the nature of the dynamics that they produce corresponds only weakly to their predictability by locally informed agents of the type we construct. We found predictability lies on a continuum from difficult to trivial. This itself is not surprising. What does seem surprising is that some of the processes that cannot be predicted are ordered. Moreover, it is these ordered rules that range in their predictability, and not the rules that produce complex fractal patterns. Through more careful examination of these rules, we found those that generate stationary patterns in time are, on average, more predictable than those that generate stationary distributions, but patterns that are periodic in time.

Second, we found that small groups of agents are not much better than individuals. This is true even though the additional agents had diverse local information and constructed their models independently. This finding suggests that the large literature on collective predictions might benefit from a deeper engagement into complexity in general and Wolfram's rules in particular.

Third, we found that ordered rules can take two forms. They can produce horizontal patterns or they can produce vertical patterns. The latter produce future states based on current states of local cells, so they can be predicted with some accuracy. The former produce future states based on current states of nonlocal cells. Therefore, they cannot be predicted by a locally informed agent. This insight shows why the complexity of a pattern does not correspond neatly to its predictability.

Many social processes are complex. Outcomes emerge from interactions between locally informed agents following rules. In this paper, we have seen that those outcomes may be difficult to predict for both individuals and small groups. Whether larger groups can leverage their diversity of information to make accurate predictions is an open question that is worth exploring.

Acknowledgments

We would like to thank the Santa Fe Institute (SFI) community for its help and accommodations during this research process. This material is based upon work supported under a National Science Foundation Graduate Research Fellowship, the National Science Foundation, and the Army Office of Research.

Appendix A.

		Pattern
Rule	Class	(0/1 Hor/Ver)
0	1	-
1	2	0
2	2	1
3	2	1
4	2	0
5	2	0
6	2	1
7	2	1
8	1	-
9	2	1
10	2	1
11	2	1
12	2	0
13	2	0
14	2	1
15	2	1
16	2	1
17	2	1
18	-	-
19	2	0
20	2	1
21	2	1
22	-	-
23	2	0
24	2	1
25	2	1
26	2	1
27	2	1
28	2	0
29	2	0
30	-	-
31	2	1
32	1	-
33	2	0
34	2	1
35	2	1
36	2	0
37	2	0
38	2	1
39	2	1
40	1	-
41	4	-
42	2	1
43	2	1
44	2	0
45	-	-
46	2	1
47	2	1
48	2	1
49	2	1
50	2	0
51	2	0
52	2	1
53	2	1
54	4	-
55	2	0
56	2	1
57	2	1
58	2	1
59	2	1
60	-	-
61	2	1
62	2	1
63	2	1
64	1	-
65	2	1
66	2	1
67	2	1
68	2	0
69	2	0
70	2	0
71	2	0
72	2	0
73	2	1
74	2	1
75	-	-
76	2	0
77	2	0
78	2	0
79	2	0
80	2	1
81	2	1

Rule	Class	Pattern (0/1 Hor/Ver)
82	2	1
83	2	1
84	2	1
85	2	1
86	-	-
87	2	1
88	2	1
89	-	-
90	-	-
91	2	0
92	2	0
93	2	0
94	2	0
95	2	0
96	1	-
97	2	1
98	2	1
99	2	1
100	2	0
101	-	-
102	-	-
103	2	1
104	2	0
105	-	-
106	4	-
107	2	1
108	2	0
109	2	1
110	4	-
111	2	1
112	2	1
113	2	1
114	2	1
115	2	1
116	2	1
117	2	1
118	2	1
119	2	1
120	4	-
121	2	1
122	-	-
123	2	0
124	4	-
125	2	1
126	-	-
127	2	0
128	1	-
129	-	-
130	2	1
131	2	1
132	2	0
133	2	0
134	2	1
135	-	-
136	1	-
137	4	-
138	2	1
139	2	1
140	2	0
141	2	0
142	2	1
143	2	1
144	2	1
145	2	1
146	-	-
147	4	-
148	2	1
149	-	-
150	-	-
151	-	-
152	2	1
153	-	-
154	2	1
155	2	1
156	2	0
157	2	0
158	2	1
159	2	1
160	1	-
161	-	-
162	2	1
163	2	1

		Pattern
Rule	Class	(0/1 Hor/Ver)
164	2	0
165	-	-
166	2	1
167	2	1
168	1	-
169	4	-
170	2	0
171	2	1
172	2	0
173	2	1
174	2	1
175	2	1
176	2	1
177	2	1
178	2	1
179	2	0
180	2	1
181	2	1
182	-	-
183	-	-
184	2	1
185	2	1
186	2	1
187	2	1
188	2	1
189	2	1
190	2	1
191	2	1
192	1	-
193	4	-
194	2	1
195	-	-
196	2	0
197	2	0
198	2	0
199	2	0
200	2	0
201	2	0
202	2	0
203	2	0
204	2	0

		Pattern
Rule	Class	(0/1 Hor/Ver)
205	2	0
206	2	0
207	2	0
208	2	1
209	2	1
210	2	1
211	2	1
212	2	1
213	2	1
214	2	1
215	2	1
216	2	0
217	2	0
218	2	0
219	2	0
220	2	0
221	2	0
222	2	0
223	2	0
224	1	-
225	4	-
226	2	1
227	2	1
228	2	0
229	2	1
230	2	1
231	2	1
232	2	0
233	2	0
234	1	-
235	1	-
236	2	0
237	2	0
238	1	-
239	1	-
240	2	1
241	2	1
242	2	1
243	2	1
244	2	1
245	2	1

Rule	Class	Pattern (0/1 Hor/Ver)
246	2	1
247	2	1
248	1	-
249	1	-
250	1	-
251	1	-
252	1	-
253	1	-
254	1	-
255	1	-

References

[1] S. E. Page, *The Difference: How the Power of Diversity Creates Better Groups, Firms, Schools, and Societies*, Princeton, NJ: Princeton University Press, 2007.

[2] N. Al-Najjar, R. Casadesus-Masanell, and E. Ozdenoren, “Probabilistic Representation of Complexity,” *Journal of Economic Theory*, **111**(1), 2003 pp. 49–87. doi:10.1016/S0022-0531(03)00075-9.

[3] J. Barwise and J. Seligman, *Information Flow: The Logic of Distributed Systems*, New York: Cambridge University Press, 1997.

[4] L. K. Hansen and P. Salamon, “Neural Network Ensembles,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**(10), 1990 pp. 993–1001. doi:10.1109/34.58871.

[5] L. Hong and S. E. Page, “Interpreted and Generated Signals,” *Journal of Economic Theory*, **5**, 2009 pp. 2174–2196. doi:10.1016/j.jet.2009.01.006.

[6] P. E. Tetlock, *Expert Political Judgment: How Good Is It? How Can We Know?*, Princeton, NJ: Princeton University Press, 2005.

[7] S. Wolfram, *A New Kind of Science*, Champaign, IL: Wolfram Media, Inc., 2002.