

# Predicting Cellular Automata

Jameson Toole  
Scott E. Page

SFI WORKING PAPER: 2010-09-020

SFI Working Papers contain accounts of scientific work of the author(s) and do not necessarily represent the views of the Santa Fe Institute. We accept papers intended for publication in peer-reviewed journals or proceedings volumes, but not papers that have already appeared in print. Except for papers by our external faculty, papers must be based on work done at SFI, inspired by an invited visit to or collaboration at SFI, or funded by an SFI grant.

©NOTICE: This working paper is included by permission of the contributing author(s) as a means to ensure timely distribution of the scholarly and technical work on a non-commercial basis. Copyright and all rights therein are maintained by the author(s). It is understood that all persons copying this information will adhere to the terms and constraints invoked by each author's copyright. These works may be reposted only with the explicit permission of the copyright holder.

[www.santafe.edu](http://www.santafe.edu)



**SANTA FE INSTITUTE**

# Predicting Cellular Automata

Jameson Toole

Massachusetts Institute of Technology, Cambridge, MA

Scott E Page

The University of Michigan, Ann Arbor, MI

September 17, 2010

## Abstract

We explore the ability of a locally informed individual agent to predict the future state of an automaton 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 the rules 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 of the values of future variables. Explanations of aggregate predictive success take two basic forms. Social science explanations tend to rely on a statistical framework in which independent errors cancel<sup>1</sup>. Computer science and statistical models rely on a logic built on diverse feature spaces (Hansen, L.K. and P. Salamon 1990). 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 (Hong and Page 2010). 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

---

<sup>1</sup>See Page 2008 for a survey. See also Al-Najjar 2003, Barwise 1997.

counter to experience. Some processes are very difficult to predict. Tetlock (2005) shows 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 ability of a single agent to small groups of agents to forecast accurately. 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 (Wolfram 2002). 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 automaton. This agent knows the initial state of the automaton and the states of the two neighboring automata. Its task is to predict the state of the automata in the center a fixed number of periods in the future. We then add other agents who also have local knowledge. Two of these agents are informed about neighboring automata, and two of these agents know the initial states of random automata. 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 fifty percent 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 automata, the rule can be predicted with one hundred percent accuracy with only local information. Contrast this to the rule "copy the state of the automata on the left." This rule produces a diagonal pattern, yet it cannot be predicted with local information. To know the state of an automata in one hundred periods requires knowing the initial state of the automata one hundred sites to the left.

## 2 The Model

We construct a string of binary cellular automata of length  $L$  with periodic boundary conditions (creating a torus) and random initial conditions (Wolfram 2002). Each automaton updates its state as a function of its state and the state of its two neighboring automata. Therefore, there exist 256 automata. For each of these, we test the ability to predict the state of an automaton  $K$  periods in the future, knowing only the initial state of the automaton 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 automaton as well as the states of the two neighboring automata.

In other words, this agent has the same information as does the automata 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 stages 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 automaton as well as the states of the two automata on its left and right. These three sites create set of eight possible initial states.

In the learning stage, the agent follows the following procedure: the agent notes the automata and its neighbors’ initial states, then keeps tallies of the automata’s state in period  $K$  (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 states of the automaton and its neighbors. The second and third columns correspond to the frequencies of states 0 and 1 in period  $K$ . The agent’s predictions, which correspond to the more likely outcome, appear in the rightmost column.

Initial State State	Outcome after $K$ Periods		Best Prediction
	0	1	
000	63	75	1
001	82	52	0
010	47	101	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$

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 automata to the left and right of the automata of the first agent. For ease of explanation, we refer to this as the *central automaton*. In these cases as well, the agents also look at the initial states of their two neighboring automata. However, these agents don’t predict the state of the automata on which they are centered but of the central automaton. To test the accuracy of the three predictors – the agent and it’s two neighbors - we rely on simple majority rule.

Finally, we also include agents who look at the initial state of two random automata as well as of the central automaton. The random automata chosen remain fixed throughout the learning stage. These agents’ predictive models consider the eight possible initial states for the three automata 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 rule's map the initial state into an predicted outcome. To asses the accuracy of these predictions, we create  $M$  random initial conditions. All  $L$  automata iterate for  $K$  periods according to whichever of the 256 rules is being studied. The state of the central automaton 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 preform the following steps:

- Step 1 : Create  $N$  random initial conditions and evolve automata  $K$  periods, 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 automata  $K$  periods.
- Step 4 : Compare predictions from the training stage to actual outcomes from testing and compute 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 automaton as well as for the group of three agents that includes the two agents on either side of the central automaton. We then examine all 256 rules computationally. Our analytic results provide a check on our computational analysis as well as provide 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 automaton appears random to our locally informed agents. We show why that's the case analytically.

### 3.1 Analytic Results for Rule 232: Majority Rule

In Rule 232, the automata looks at its state and the state of the two neighboring automata and matches the state of the majority. We denote the central automaton by  $x$  and the two neighboring automata 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 automaton and one of its neighbors are in the same state. In those cases, the state of the central automaton and that neighbor remain fixed in that state forever. In those cases, the predictive rule for the agent located at the central automaton 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 automaton 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  $\ell_i$  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  $\ell_3\ell_2\ell_1010r_1r_2r_3$ . Consider first the case where  $r_1 = 0$ . By convention, we let a question mark  $?$  denote an indeterminate state. The states of the automata iterate as follows

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

By symmetry, if  $\ell_1 = 0$ , the  $x$  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:

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

It follows then that if either  $r_2$  or  $\ell_2$  is in state 1 then the central automaton will be in state 1 in period  $K$ .

Given these calculations, we can determine the probability distribution over the state of the central automaton if it and its neighbors start in states 010. From above, unless  $r_1 = \ell_1 = 1$ , then  $x$  will be in state 0. Therefore, with probability  $\frac{3}{4}$ , it locks into state 0 in one period. With probability  $\frac{1}{4}$ , it does not lock into state 0. In those cases,  $r_1 = \ell_1 = 1$ . And, from above, with probability  $\frac{3}{4}$ ,  $x$  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:

$$Pr(x = 0 \mid wxy = 010) = \frac{3}{4} + \frac{1}{4} \frac{1}{4} [\frac{3}{4} + \frac{1}{4} \frac{1}{4} [\frac{3}{4} + \frac{1}{4} \frac{1}{4} \dots]]$$

This expression takes the form  $p + qp + q^2p^2 + q^3p^3 + \dots$ . 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.

### Rule 232: Optimal Predictions at $x$ and Accuracy

$wxy$	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 automaton. 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 automata 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 automaton, we need to include the the states for both  $y$  and  $r_1$ . We can write the initial states of these five automata as  $101yr_1$ . 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  $\ell_1wx = 101$  equals

$$Pr(x = 1 \mid \ell_1wx = 010) = \frac{1}{2} + \frac{1}{4} \frac{1}{4} [\frac{3}{4} + \frac{1}{4} \frac{1}{4} [\frac{3}{4} + \frac{1}{4} \frac{1}{4} \dots]]$$

which by a calculation similar to the one above 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 automaton 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 sixteen cases to consider. 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 zero.

### 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	1.0
00001	0	1.0	0 0 0	0	1.0
00010	0	1.0	0 0 0	0	1.0
00011	0	1.0	0 0 H	0	1.0
01000	0	1.0	0 0 0	0	1.0
01001	0	1.0	0 0 0	0	1.0
01010	1	0.2	0 1 0	0	0.8
01011	1	1.0	0 1 H	H	0.5
10000	0	1.0	0 0 0	0	1.0
10001	0	1.0	0 0 0	0	1.0
10010	0	1.0	0 0 0	0	1.0
10011	0	1.0	0 0 H	0	1.0
11000	0	1.0	H 0 0	0	1.0
11001	0	1.0	H 0 0	0	1.0
11010	1	1.0	H 1 0	H	0.5
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 above that the single agent located at the central automaton 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 automaton. 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 next describe results from computational experiments on all 256 rules relying on strings of automata having twenty sites and periodic boundary conditions. For each of the 256 rules,



automata undergo a learning stage of one thousand periods. Automata were trained and tested on the prediction of their state,  $K = 53$  periods in the future<sup>2</sup>. Once the agents had been trained, we computed their accuracy during a testing phase consisting of five hundred trials.

### 3.2.1 Predictability of Automata by a Single Agent

We first show our findings for the accuracy of the single agent located at the central automaton. Figure 1 shows a sorted distribution 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 rule 0 and rule 255 which map every initial state to all 0’s and all 1’s 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 don’t map neatly to predictability.

To see why not, we return to Wolfram’s classification (2002) which classifies automata 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.

In an appendix, we give the classification of rules that we used. We have not found a complete listing elsewhere (Appendix 6).

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 homogenous 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 suggests that the regular patterns that characterize Class II rules are not informative to a rule’s predictability and that further classification refinement is needed to better describe these automata.

---

<sup>2</sup>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 yield similar results in almost all cases.

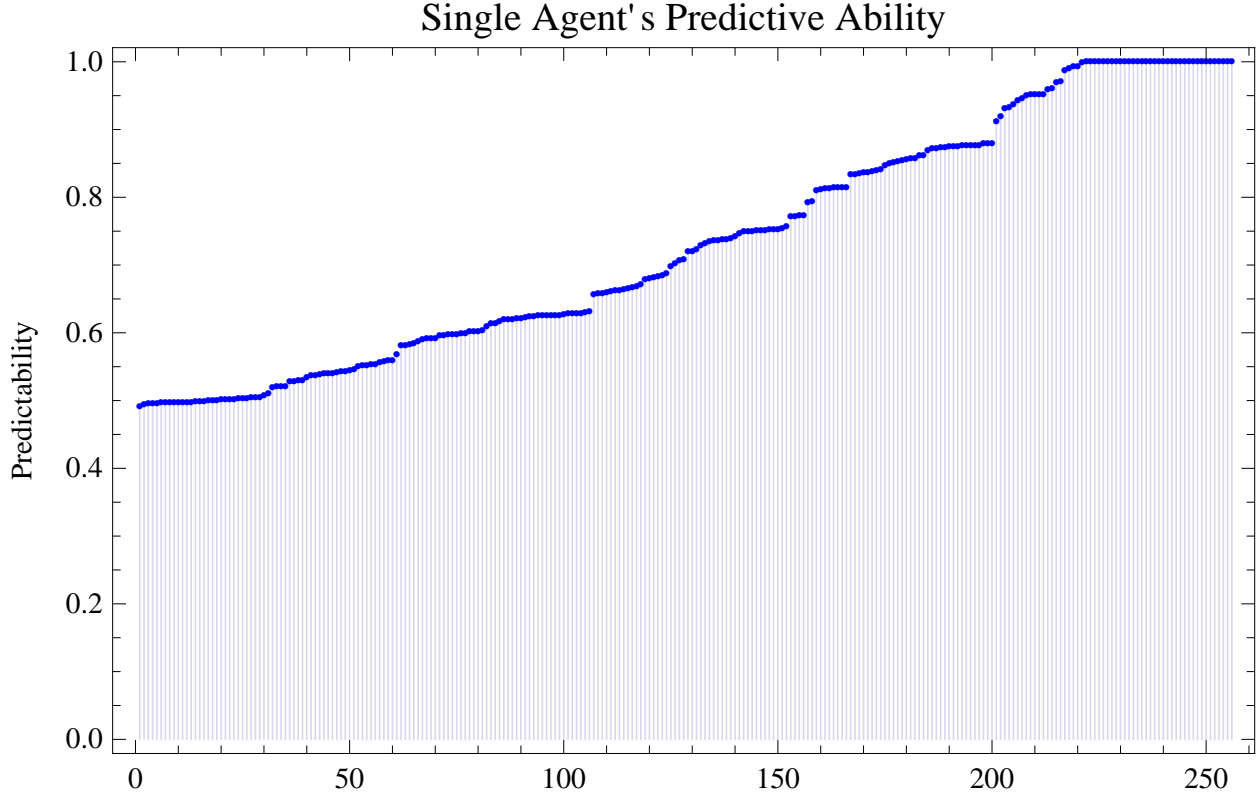


Figure 1: A single agents ability to predict its future state given the 256 rules. Rule predictability can range from being no better than a fair coin flip to %100 accuracy depending on the dynamics of the rule. The x-axis (Rule Number) does not correspond to Wolfram's numbering.

These visual intuition can be shown statistically. The table below gives the mean accuracy for the agent located at the central automaton for each class of rules as well as the standard deviation.

Accuracy (Std)	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)

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 vs. Groups

We next compare the ability of the single agent to that of small groups. Our main findings is that the small groups are not much more accurate. A statistical analysis shows no meaningful differences for any of the classes. Were we to ramp up our sample sizes, we might gain

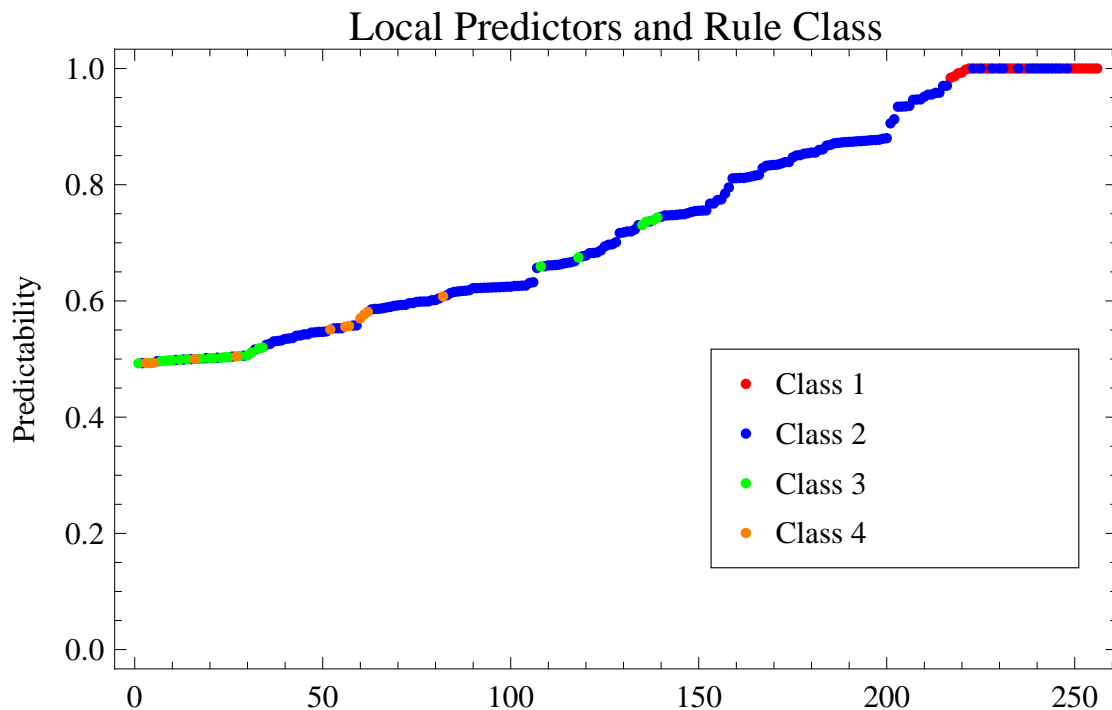


Figure 2: Using local predictors sorted according to fitness, we color-code rules based on the class assigned by Wolfram. While Classes I, III, and IV prove to be informative, Class II rules show huge variation in their ability to be predicted.

statistical significance of some of these results, but the magnitude of the differences is small – most often much less than 1%.

Accuracy (Std.)	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 (0.092)	0.539 (0.037)
All Five Agents	0.997 (0.005)	0.723 (0.132)	0.551 (0.092)	0.547 (0.040)

These aggregate data demonstrate that on average adding predictors does not help. That’s true even for the Class II rules and the Class IV rules. We found this to be rather surprising.

These aggregate data mask 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.

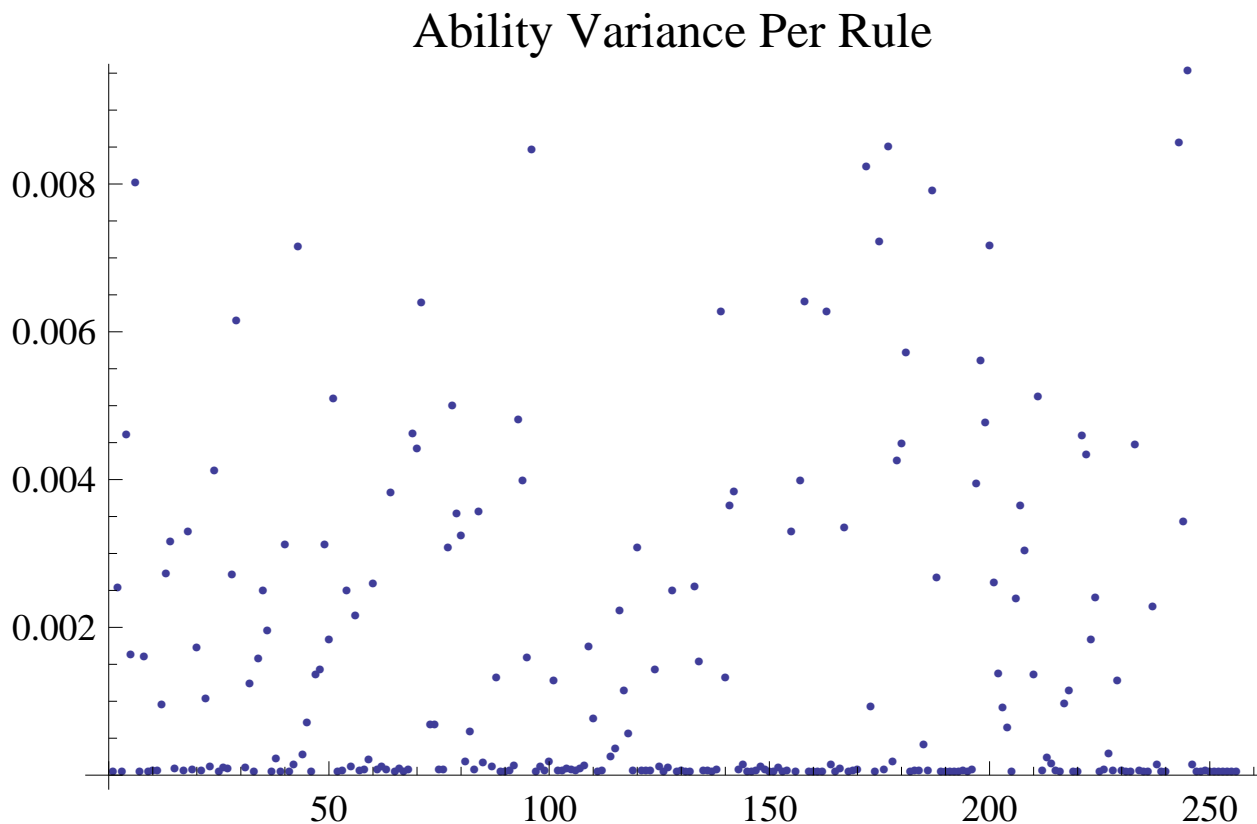


Figure 3: The variance in ability between four combinations of predictors reveals that for many rules, all predictors preform equally.

Detailed analysis of specific rules, such as the one we performed form Rule 232 can reveal why for some rules adding local predictors increases or decreases predictability, but there exist no general pattern. The data show that over all rules adding local predictors, random predictors, or both does not help with overall predictability. This finding stands in sharp contrast to statistical results which show the value of adding more 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 in automata 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 automata, we will show that vertical patterns provide an opportunity to learn dynamics and make accurate predictions, while horizontal patterns makes information gathering much more difficult. Finally, we show that accurately predicting the future given

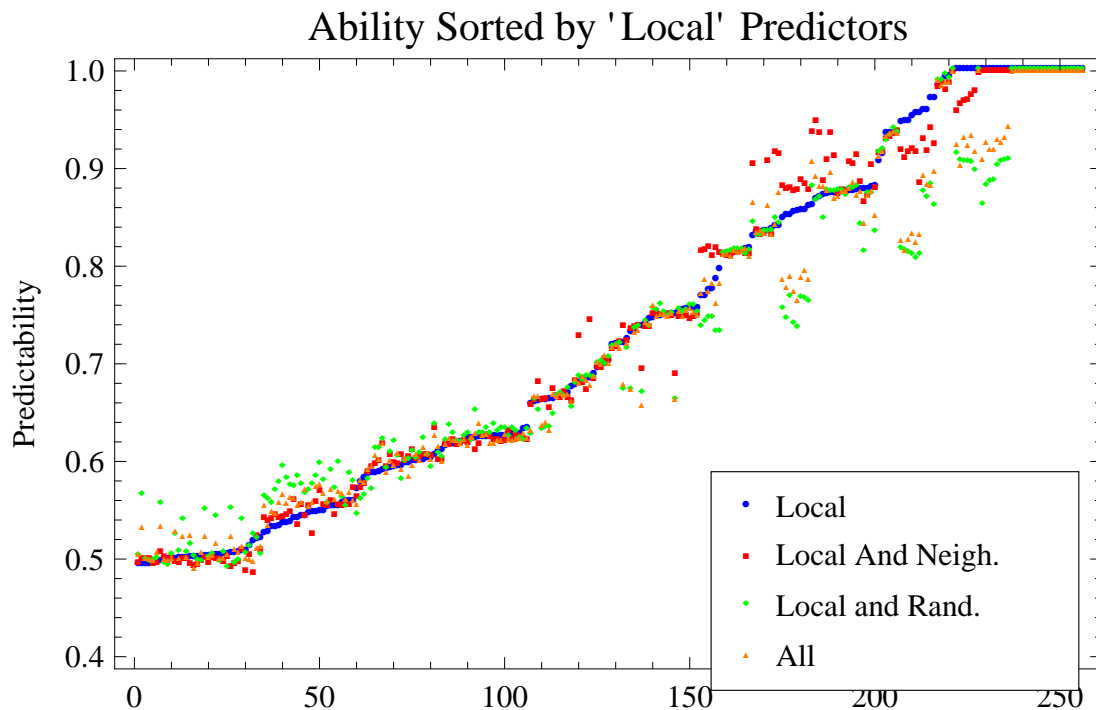


Figure 4: Various combinations of predictors are sorted by the ability of Local predictors. While the x-axis (Rule Number) does not correspond to Wolfram's numbering, all predictors can be easily compared to the use of only Local predictors.

each of these patterns requires automata acquiring different types information.

### 3.3.1 Rule 170: Pass to the Left

As shown above in Figure 5, Rule 170 generates horizontal patterns in time. These horizontal patterns differ from vertical patterns in that no single automata locks into a stationary state. From an individual automata'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.

This randomness makes prediction based on a 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.

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

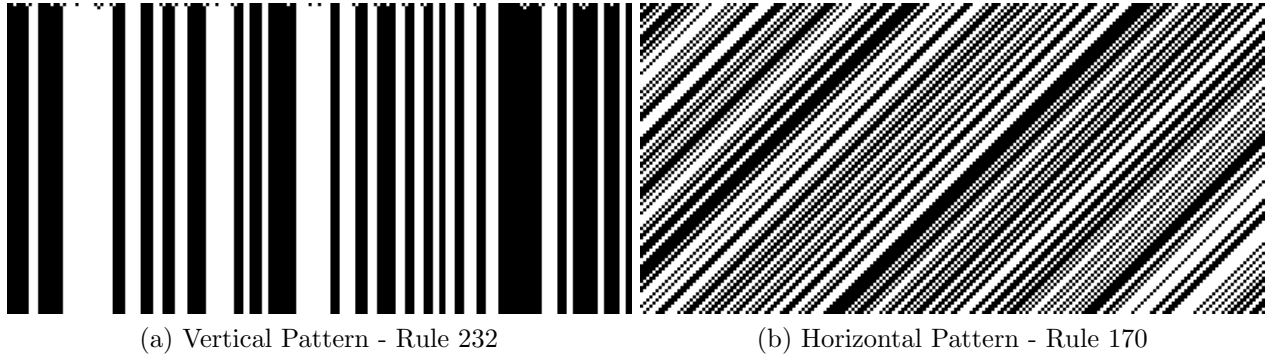


Figure 5: A pair of Class 2 rules is shown. Rule 232 displays a vertical pattern where individual automata, starting from a random initial conditions, lock into stationary states. Rule 170, by contrast, generates patterns that continually shift to the left, never settling into a stationary state.

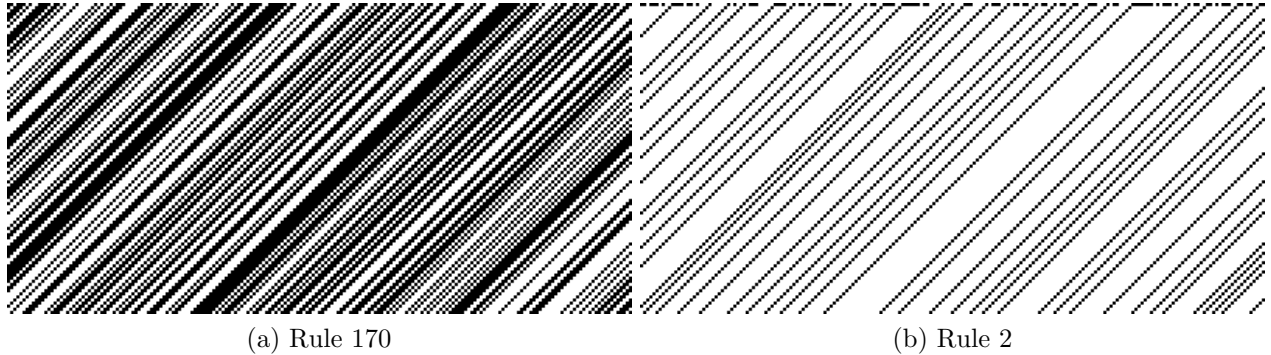


Figure 6: Although horizontal patterns never result in individual stationary states, they do create different equilibrium distributions of bits.

We can see this by considering Rule 170, informally named “Pass to the Left”. This rule simply tells each automata to take on the state of the cell to their left in the next period. For any random initial state, half of the automata should be in the state 0, with the other half in state 1. Under Rule 170, these initial bits simply rotate around the torus.

Though individual automata 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 the 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).

Given rule 170, an agent trying to predict the central automaton’s state in period  $K$  learns nothing of value from the automata’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 next round. Because of this, the equilibrium

distribution has many more 0s than 1s. Because automata are initialized randomly, this occurs with probability  $\frac{1}{8}$ . Thus we expect  $\frac{1}{8}$  to be the fraction of 1s in our equilibrium distribution. Knowing this, any automata will correctly predict it's outcome 87.5% ( $\frac{7}{8}$ ) 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 automata can correctly guess their final state with accuracy  $50 \pm 1\%$ , while Rule 2 allows accuracy of  $87.5 \pm 1\%$ .

In most cases, we expect the lack of stationary states for individual automata 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 predicability compared to rules generating vertical patterns. Figure 7 confirms our expectations.

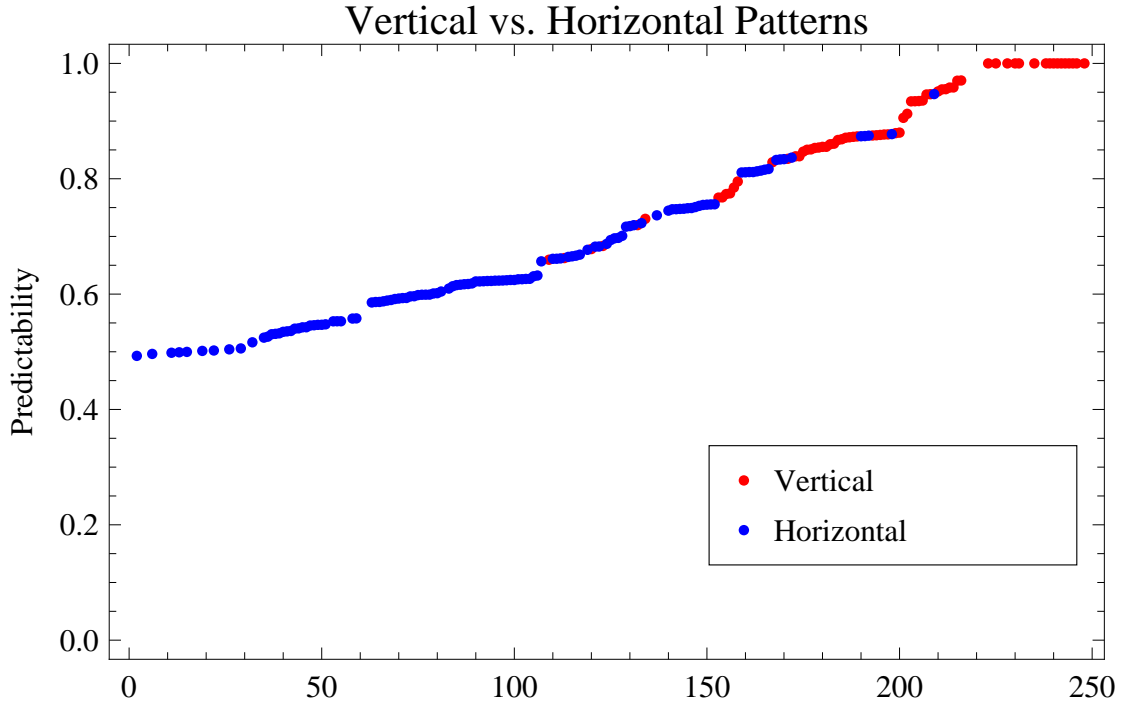


Figure 7: Rules are sorted based on predictability fulfilling our expectations that rules generating vertical patterns be more easily predicted using inductive reasoning than horizontal patterns

Finally, we note that in cases with horizontal patterns, each automata'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 (neighbor, random, or both), where as for vertical patterns, neighbors may provide some information, good or bad.

## 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  $K$  periods in the future as well as initial states. The accuracy of their resulting predictive models was then calculated during a testing phase.

We find three main results. First, classifications of cellular automata 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, and not rules that produce complex, fractal patterns, range in predictability. 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 automata, so they can be predicted with some accuracy. The former produce future states based on current states of non local automata. 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 local informed rule following agents. In this paper, we've seen that those outcomes may be difficult to predict for both individuals and for small groups. Whether larger groups can leverage their diversity of information to make accurate predictions is an open question that's worth exploring.

## 5 Acknowledgements

We would like to thank the 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.



## References

- [1] Al-Najjar, N., R. Casadesus-Masanell and E. Ozdenoren (2003) “Probabilistic Representation of Complexity”, *Journal of Economic Theory* 111 (1), 49 - 87.
- [2] Aragones, E., I. Gilboa, A. Postlewaite, and D. Schmeidler (2005) “Fact-Free Learning”, *The American Economic Review* 95 (5), 1355 - 1368.
- [3] Barwise and Seligman, (1997) *Information Flow: The Logic of Distributed Systems* Cambridge Tracts In Theoretical Computer Science, Cambridge University Press, New York.
- [4] Billingsley, P. (1995) *Probability and Measure (3rd Edition)* Wiley-Interscience
- [5] Caplan, Bryan (2007) *The Myth of the Rational Voter: Why Democracies Choose Bad Policies* Princeton University Press.
- [6] Fryer, R. and M. Jackson (2008 ), “A Categorical Model of Cognition and Biased Decision-Making”, *Contributions in Theoretical Economics, B.E. Press*
- [7] Holland, J. and J. Miller (1991) “Artificial Agents in Economic Theory”, *The American Economic Review Papers and Proceedings* 81, 365 - 370.
- [8] Gilboa, I., and D. Schmeidler, (1995) Case-Based Decision Theory, *The Quarterly Journal of Economics*, 110, 605-639.
- [9] Hansen, L.K. and P. Salamon, (1990) “Neural network ensembles,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:10, 993-1001.
- [10] Holland, J.H. , K. Holyoak, R E Nisbett and P. Thagard. (1989) *Induction: Processes of Inference, Learning, and Discovery* MIT Press.
- [11] Hong L. and S. Page (2001) “Problem Solving by Heterogeneous Agents”, *Journal of Economic Theory* 97, 123 - 163.
- [12] Hong L. and S. Page (2008) “On the Possibility of Collective Wisdom” working paper
- [13] Hong, L and S.E. Page (2009) “Interpreted and Generated Signals” *Journal of Economic Theory*, 5: 2174-2196
- [14] Judd, K. (1997) ”Computational Economics and Economic Theory: Complements or Substitutes?” *Journal of Economic Dynamics and Control*.
- [15] Judd, K. and S. Page (2004) “Computational Public Economics”, *Journal of Public Economic Theory* forthcoming.
- [16] Klemperer, P. (2004) *Auctions: Theory and Practice* Princeton University Press.

- [17] Ladha, K. (1992) "The Condorcet Jury Theorem, Free Speech, and Correlated Votes", *American Journal of Political Science* 36 (3), 617 - 634.
- [18] Milgrom, P. and R. Weber (1982) "A Theory of Auctions and Competitive Bidding", *Econometrica* 50 (5), 1089 - 1122.
- [19] Nisbett, R. (2003) *The Geography of Thought: How Asians and Westerners Think Differently...and Why* Free Press, New York.
- [20] Page, S. (2007) *The Difference: How the Power of Diversity Creates Better Firms, Schools, Groups, and Societies* Princeton University Press.
- [21] Pearl, Judea (2000) *Causality* New York: Oxford University Press.
- [22] Stinchcombe, A. (1990) *Information and Organizations* California Series on Social Choice and Political Economy I University of California Press.
- [23] Tesfatsion, L. (1997) "How Economists Can Get A-Life" in *The Economy as a Complex Evolving System II* W. Brian Arthur, Steven Durlauf, and David Lane eds. pp 533-565. Addison Wesley, Reading, MA.
- [24] Tetlock, P. (2005) *Expert Political Judgment: How Good is it? How Can we Know?* Princeton University Press. Princeton, NJ.
- [25] Valiant, L.G. (1984) "A Theory of the Learnable" *Communications of the ACM*, 17(11), 1134-1142.
- [26] Von Hayek, F. (1945) "The Use of Knowledge in Society," *American Economic Review*, 4 pp 519-530.
- [27] Wellman, MP, A Greenwald, P. Stone, and PR Wurman (2003) "The 2001 Trading Agent Competition" *Electronic Markets* 13(1).
- [28] Wolfram, Stephen. (2002) "A New Kind of Science." *Wolfram Media*.

## 6 Appendix

Rule	Class	Pattern (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
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
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
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
246	2	1
247	2	1
248	1	-
249	1	-
250	1	-
251	1	-
252	1	-
253	1	-
254	1	-
255	1	-