Cellular Automata and Neural Nets

Alex Ledger and Jacob Menick

Reed College

January 7, 2016

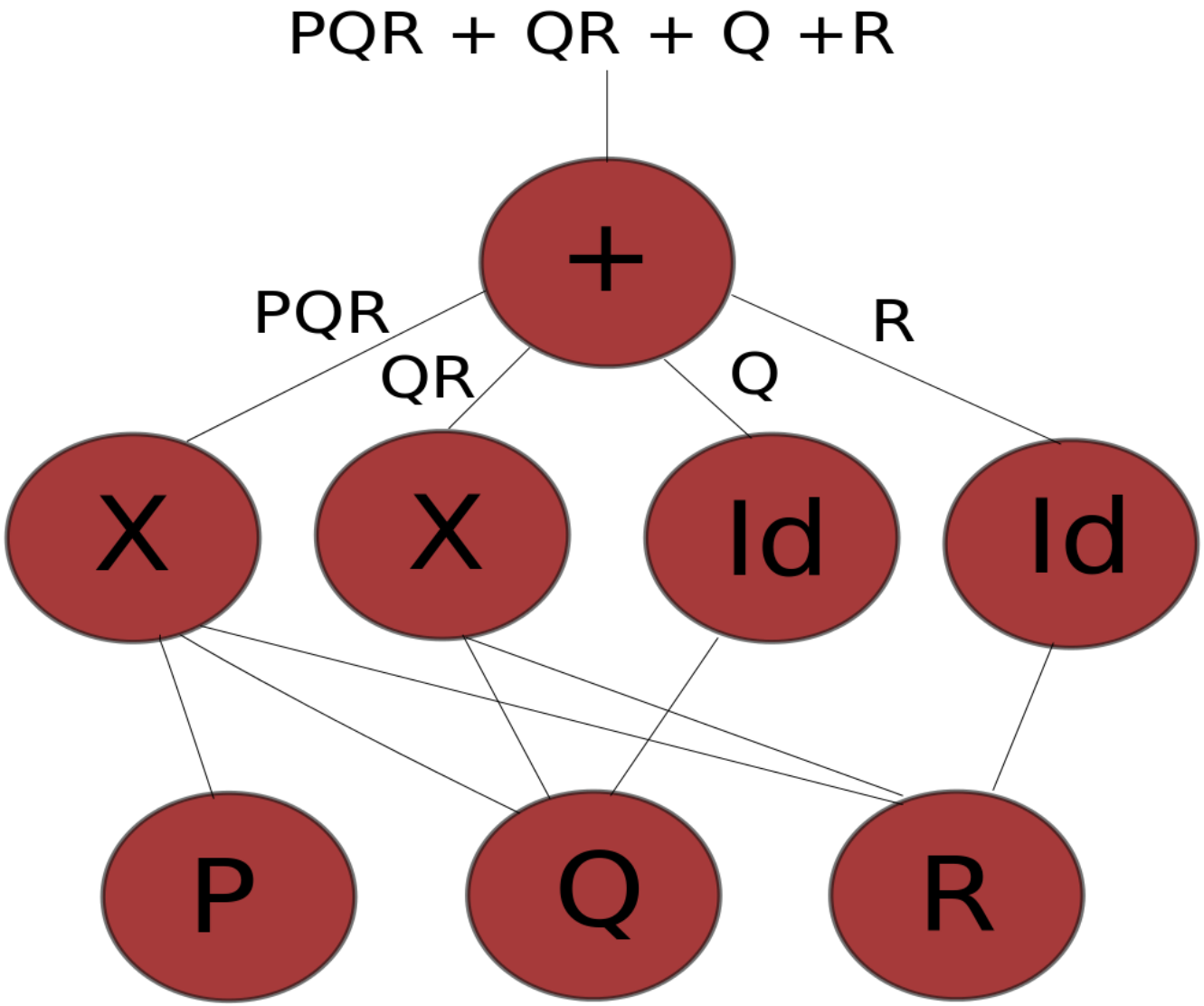
Rule 110

Recall rules for Rule 110:

Current Cell Pattern	111	110	101	100	011	010	001	000
New State for center cell	0	1	1	0	1	1	1	0

- $f(p, q, r) = q + r + qr + pqr$
- $f(p, q, r) = (q \wedge (\bar{p})) \vee (q \oplus r)$
- We constructed a neural network (on paper) that calculates one iteration of Rule 110:
  - Input: target cell's 3 parents,
  - Output: whether cell is 0 or 1

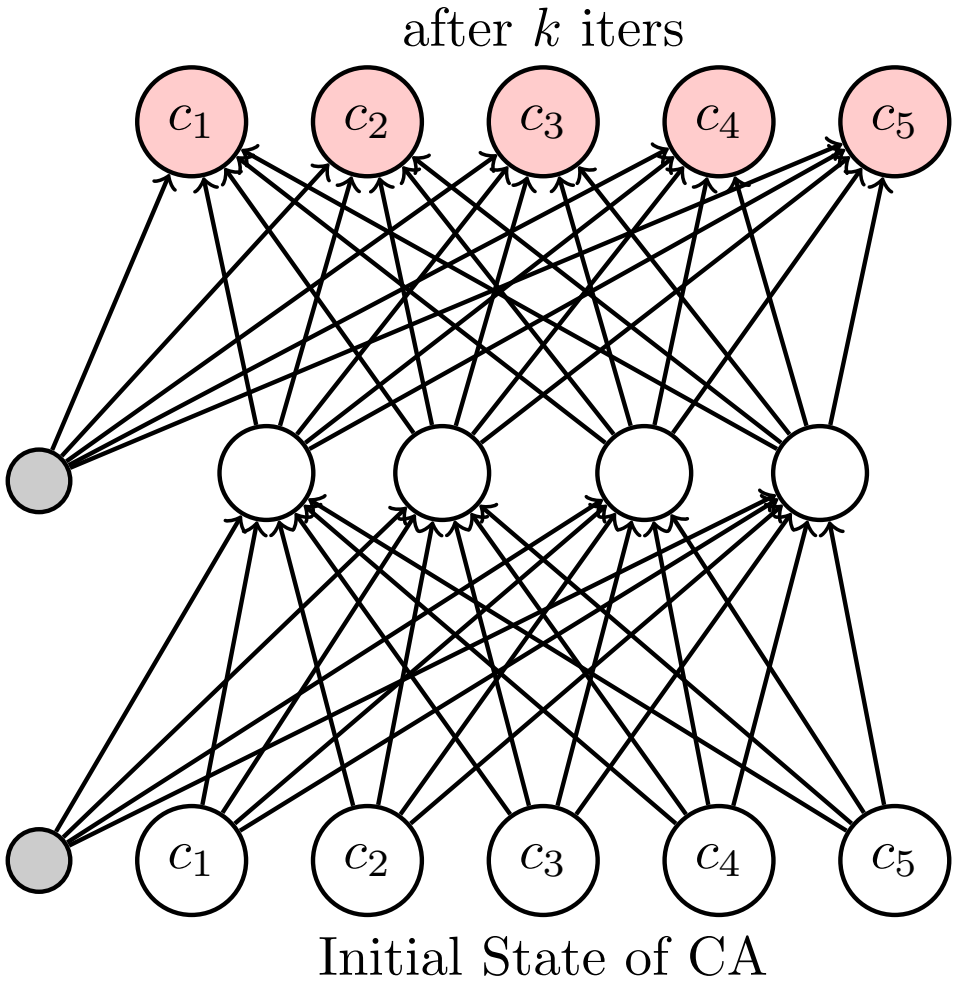
Neural Net Computing Rule 110



Our Thinking

- We decided to see if a neural network could efficiently compute rule 110
- Given an initial configuration (the first row)
- Can a neural net find the kth row *without computing every row*?
- Can a neural net *predict* the kth row given the first row?
- So we made some neural nets and tested it.

Our Neural Net Architecture

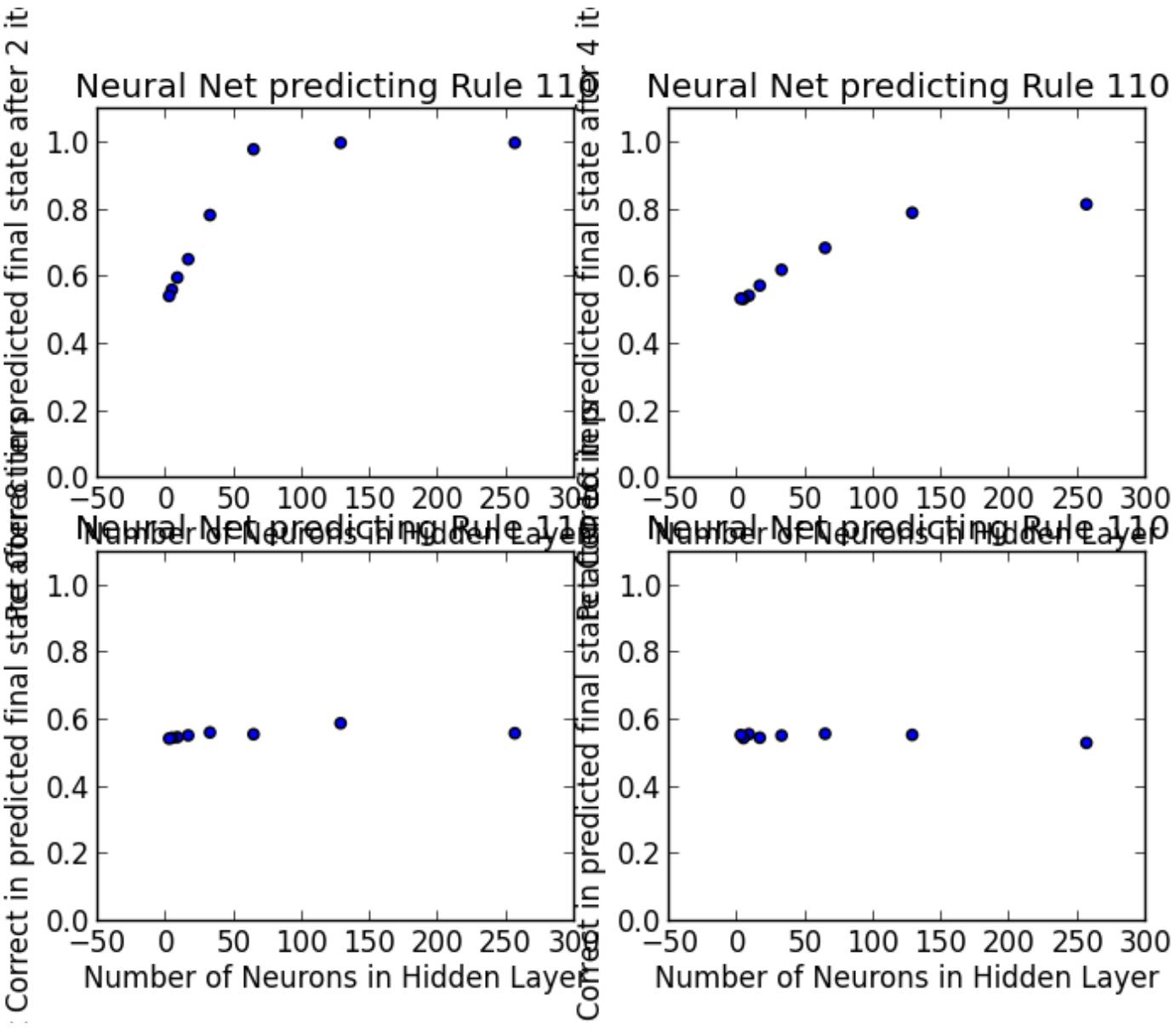


Hypothesis

- The neural nets will not be able to accurately predict the kth row.
- Differences in initial conditions propagate outward in unpredictable ways.
  $\Rightarrow$ Hard for NN to generalize

Results 1

Results 2


Neural Net predicting Rule 110
Neural Net predicting Rule 110
Neural Net predicting Rule 110
Neural Net predicting Rule 110
Correct in predicted final state after 2 iterations
Correct in predicted final state after 4 iterations
Number of Neurons in Hidden Layer
Number of Neurons in Hidden Layer
Number of Neurons in Hidden Layer
Number of Neurons in Hidden Layer
0.0
0.2
0.4
0.6
0.8
1.0
−50
0
50
100
150
200
250
300

Conclusions

- We can build a neural net to exactly model an iteration of Rule 110
- A neural net with a single hidden layer **can** predict rule 110 for a small number of iterations, but only trivially. Just give it enough neurons in the hidden layer to model it.
- Doesn't appear to need many layers
- Incompressibility of Class 4 Automata?