

Project 8

Gene Networks I (Boolean)

Outline

genestmdriver

STM = genestm(wire,rule)

(majority of the work is here)

b = d2b(r,C)

[convert decimal to binary]

d = b2d(bin)

[convert binary to decimal]

genestmdriver

1. Set wire and rule
2. Use biograph to visualize the gene net
3. call genestm on wire and rule (genestm outputs the STM diagram)
4. Use biograph to view the STM

Change a rule! (flip a bit)

5. Call genestm again on altered rules, biograph the STM

(Code given in notes!)

```
wire = [4 2 3; 5 3 2; 3 6 1; 5 4 6; 6 1 2; 3 5 6];
n = size(wire,1);
a = zeros(n+1);
for i=1:n
    a(wire(i,:),i) = 1;
    ids{i} = num2str(i);
end
ids{n+1} = 'Gene Net';
g = biograph(a,ids);
selfcon = find(diag(a)==1);
for i=1:length(selfcon)
    g.nodes(selfcon(i)).Shape = 'diamond';
end
view(g)
```

Toy example

Consider the network of 3 genes with
wire = [1 2 3; 2 3 1; 3 1 2] and
rule = [62; 62; 62]

1. act as biograph: draw the gene net.
(self-connecting nodes are diamonds!)

STM = genestm(wire,rule)

1. Build and display the rule table, “ruletab”

Before we go on, we'll review the conceptual mechanics of the project, to understand how ruletab works and why we need it.

Understanding the Model

- How do we get from current state of network (s) to next state of network (ns) using wire and rule?
- Wire tells us how the nodes are connected
- Rule tells us the next state given the current state

Wire

```
wire = [4 2 3; 5 3 2; 3 6 1; 5 4 6; 6 1 2; 3 5 6];
```

wire: (each row corresponds to one node and shows its connections)

Row 1	4	2	3	(Node 1 connected to nodes 4,2,3)
Row 2	5	3	2	(Node 2 connected to nodes 5,3,2)
Row 3	3	6	1	(Node 3 connected to nodes 3,6,1)
Row 4	5	4	6	(Node 4 connected to nodes 5,4,6)
Row 5	6	1	2	(Node 5 connected to nodes 6,1,2)
Row 6	3	5	6	(Node 6 connected to nodes 3,5,6)

State

Total number of states: 2^n (n =number of genes)

$$\mathbf{s} = (1 \ 0 \ 1 \ 0 \ 0 \ 1)$$

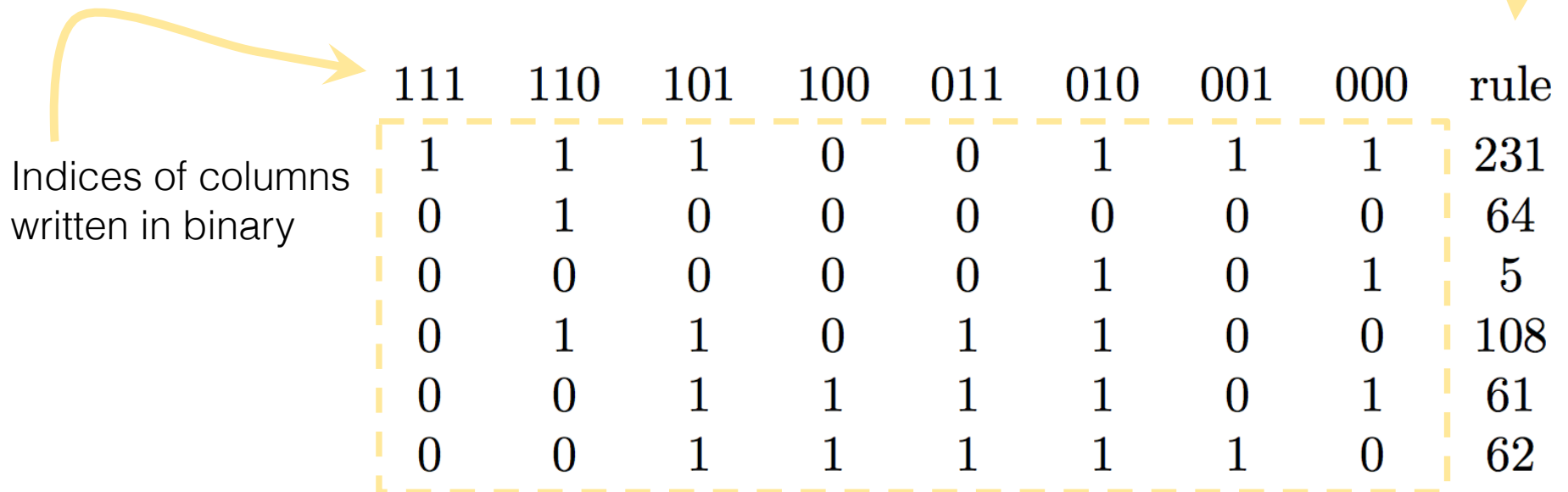
[1 means “on”, 0 means “off”]

→ Node 1 is on, node 2 is off, etc.

Rule → ruletab

Rule= [231; 64; 5; 108; 61; 62]

Ruletab is simply these rules turned into binary...one row per rule.



A diagram illustrating the conversion of rule indices to binary. A yellow arrow points from the text 'Indices of columns written in binary' to the first column of the table, which contains the binary representations of the indices: 111, 110, 101, 100, 011, 010, 001, and 000. Another yellow arrow points from the text 'Rule= [231; 64; 5; 108; 61; 62]' to the 'rule' column of the table, which contains the corresponding rule values: 231, 64, 5, 108, 61, and 62. The table is enclosed in a dashed yellow border.

	111	110	101	100	011	010	001	000	rule
	1	1	1	0	0	1	1	1	231
	0	1	0	0	0	0	0	0	64
	0	0	0	0	0	1	0	1	5
	0	1	1	0	1	1	0	0	108
	0	0	1	1	1	1	0	1	61
	0	0	1	1	1	1	1	0	62

Do not hardcode ruletab!!! Use d2b to generate.

STM = genestm(wire,rule)

1. Build and display the rule table, “ruletab”
2. Calculate the next states (ns) and build the STM

Once we have wire, state, and have ruletab, how do we use them to find the next state (ns) from the current state (s)?

Next State

- For each node in (s):
 - Look at **wire**, to which other nodes it that node connected?
 - Look at **state**, what is the state of those other nodes? (It will be some combination of 1s and 0s)
 - Convert this combination of 1s and 0s into decimal notation, it is the index of the column you need to look at in rulemat
 - Look down the column until you arrive at the row corresponding to your node of interest. If a 1 lies in that spot, $ns(\text{that node})=1$. If 0, $ns(\text{that node})=0$.
 - Build a vector of 1s and 0s, for the on/off state of each node. This is your new state vector! To find the “number” of the next state, turn this from binary to decimal.

Next State (ns) \rightarrow STM

The STM is an adjacency matrix for states.

What does it mean for two states to be adjacent?

Next State (ns) \rightarrow STM

The STM is an adjacency matrix for states.

What does it mean for two states to be adjacent?

\rightarrow If we see the first state transition to the next!

What does the matrix look like/how do we build it?

Next State (ns) \rightarrow STM

The STM is an adjacency matrix for states.

What does it mean for two states to be adjacent?

\rightarrow If we see the first state transition to the next!

What does the matrix look like/how do we build it?

\rightarrow Where i is the index of the current state, and j is the number of the next state, converted from binary to decimal, place a 1 in the (i,j) slot of STM! But be careful. Matlab indexes rows and columns starting from “1” while our gene net indexes them starting from “0.”

Quiz Question 1

Given the state transition matrix (STM) of a Boolean gene network, if state 8 transitions to state 10, then what entry of STM should equal 1 to signify this transition?

- ☐ A. $STM(10,8)$
- ☐ B. $STM(8,10)$
- ☐ C. $STM(9,11)$
- ☐ D. $STM(11,9)$

STM = genestm(wire,rule)

1. Build and display the rule table, “ruletab”
2. Calculate the next states (ns) and build the STM

```
for i=1:2^n
    s = d2b(i-1,n);
    ns = next state (using s, wire and rulemat)
    j = b2d(ns);    your binary to decimal converter
    STM(i,j+1) = 1;
end
```


d2b and b2d

Quiz question 5: translate [1 0 1 0 1 0] into base 10 (decimal)

- ☐ A.42
- ☐ B.32
- ☐ C.22

The coding of b2d follows logically from this question (just sum the correct powers of two, multiplying the others by 0).

I leave coding d2b to you.

Hint from notes: `b(C-floor(log2(r)))=1, reset r, repeat.`