538 Duck Pond

Xavier Weisenreder

January 20, 2020

Riddler Problem

https://fivethirtyeight.com/features/can-you-track-the-delirious-ducks/

After a long night of frivolous quackery, two delirious ducks are having a difficult time finding each other in their pond. The pond happens to contain a 3×3 grid of rocks.

Every minute, each duck randomly swims, independently of the other duck, from one rock to a neighboring rock in the 3×3 grid — up, down, left or right, but not diagonally. So if a duck is at the middle rock, it will next swim to one of the four side rocks with probability 1/4. From a side rock, it will swim to one of the two adjacent corner rocks or back to the middle rock, each with probability 1/3. And from a corner rock, it will swim to one of the two adjacent side rocks with probability 1/2.

If the ducks both start at the middle rock, then on average, how long will it take until they're at the same rock again? (Of course, there's a 1/4 chance that they'll swim in the same direction after the first minute, in which case it would only take one minute for them to be at the same rock again. But it could take much longer, if they happen to keep missing each other.)

Extra credit: What if there are three or more ducks? If they all start in the middle rock, on average, how long will it take until they are all at the same rock again?

Markov Chain Graph

Graph Relations Were Calculated Manually

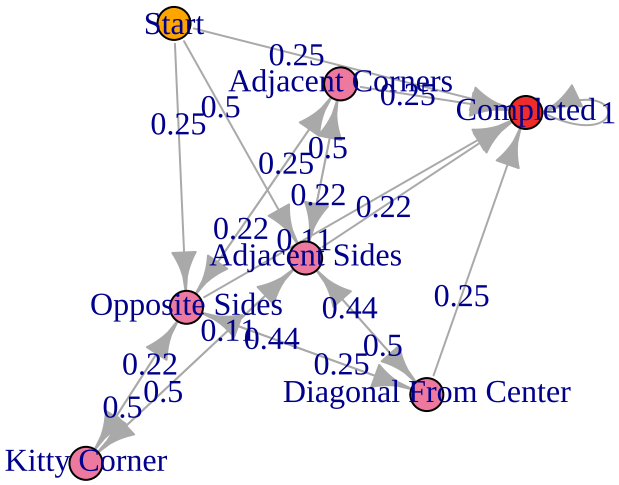
```
library(markovchain)
```

```
## Package:  markovchain
## Version:  0.8.2
## Date:     2020-01-10
## BugReport: http://github.com/spedygiorgio/markovchain/issues
```

```
full.graph.matrix <- t(matrix(c(0, 1/2, 1/4, 0, 0, 0, 1/4, 0, 0, 0, 1/9,
                                2/9, 4/9, 2/9, 0, 0, 0, 2/9, 2/9, 4/9, 1/9, 0,
                                1/2, 1/2, 0, 0, 0, 0, 0, 1/2, 1/4, 0, 0, 0,
                                1/4, 0, 1/2, 1/4, 0, 0, 0, 1/4, 0, 0, 0, 0, 0, 0, 1),
                              nrow=7, ncol=7))
duck.markov <- new('markovchain',
              transitionMatrix = full.graph.matrix,
              states = c('Start','Adjacent Sides','Opposite Sides', 'Kitty Corner',
                         'Adjacent Corners', 'Diagonal From Center', 'Completed'))
full.graph.matrix
```

```
##      [,1] [,2] [,3]      [,4]      [,5]      [,6]      [,7]
## [1,]    0  0.5 0.25 0.0000000 0.0000000 0.0000000 0.2500000
## [2,]    0  0.0 0.00 0.1111111 0.2222222 0.4444444 0.2222222
## [3,]    0  0.0 0.00 0.2222222 0.2222222 0.4444444 0.1111111
## [4,]    0  0.5 0.50 0.0000000 0.0000000 0.0000000 0.0000000
## [5,]    0  0.5 0.25 0.0000000 0.0000000 0.0000000 0.2500000
## [6,]    0  0.5 0.25 0.0000000 0.0000000 0.0000000 0.2500000
## [7,]    0  0.0 0.00 0.0000000 0.0000000 0.0000000 1.0000000
```

```
plot(duck.markov)
```



Calculating Average Steps Based on Node

```
library(MASS)
matrix.calculating.absorption <- full.graph.matrix[1:6, 1:6]
minus.iden <- diag(6)-matrix.calculating.absorption
ginv(minus.iden) %*% matrix(c(1,1,1,1,1,1), nrow=6)
```

```
##          [,1]
## [1,] 4.905405
## [2,] 4.972973
## [3,] 5.675676
## [4,] 6.324324
## [5,] 4.905405
## [6,] 4.905405
```