Markov Chain Programming Functions for the TI-Nspire CX CAS Handheld and Associated Emulators.

Language: English
Version: 1.0
Date: 9/26/2018
Author: James O. Thompson
Email: jodaddy101@hotmail.com

Table of Contents

1. Introduction

The functions presented here are not intended to teach Markov chains. Instead they should complement a good introductory text on the subject. Such a text can be found in "Finite Mathematics" by Lial, Greenwell and Ritchey. This book is frequently used in a college course for students majoring in business, management, economics, etc.

Any student attempting to learn Markov chains without, at least, help with a modern calculator such as the TI-84 is sorely disadvantaged. Even with the help of the matrix features of the TI-84, the flood of trivial calculations obscures the beauty of Markov chains. Burdened with repetitive, redundant obscuring calculations such a disadvantaged student would be able to solve few of the many rich exercises at the end of each section of the book. With the help of the functions presented here the student should be able to solve many of the interesting problems posed at the end of each book section. That … and have time for other concurrent college courses as well.

The functions presented here are for the student equipped with a TI-Nspire CX CAS handheld or an emulator on a PC, iPad, etc. The "CAS" (Computer Algebra System) part is essential. The functions cannot be installed on the TI-Nspire without the CAS part.

The functions should be introduced piece meal through the course as aids to automate methods already taught from the book. The student should know the process being automated and use the automation only to avoid repetitive calculations not contributing to the learning process.

The functions are presented in four sets in Sections 4, 5 6 and 7 corresponding to the approach taken in "Finite Mathematics". Section 4 describes the augmented matrices to created Markov chains and state vectors along with two functions to create associated templates. Section 5 describes functions unique to regular Markov chains. Section 6 describes functions unique to absorbing Markov chains. Finally, Section 7 describes miscellaneous common functions.

2. Platforms

The functions presented here will execute on the TI-Nspire CX CAS handheld, the TI-Nspire CX CAS Student Software and associated emulators executing on the iPad.

3. Installation

All of these functions are bundled into a single file, "markov.tns". This file is to be installed in the "mylib" folder where it will join the other files (numtheory.tns and linalgcas.tns) from the default installation. The file can be transferred to the handheld using the process described for file transfers in the handheld owner's manual. On a PC the "mylib" subdirectory is in the User's Documents subdirectory which, unfortunately, Microsoft likes to hide. See this web reference on how to locate the User's Documents subdirectory.

After installing this file, you must "Refresh Libraries". On the handheld, press Doc, then choose the "Refresh Libraries" option. On the Student Software, choose the Tools menu, then the

“Refresh Libraries” option. Following this the functions may be accessed as the file name (markov) followed by the backslash (\) followed by the function name. e.g., “markov\about()”.

Following installation, the function source code may be inspected and modified as desired. Furthermore, it may be distributed freely and for free.

4. Template Creation Functions

Textbooks commonly illustrate Markov functions where the matrix columns and rows are unlabeled. This can be extremely confusing, especially when the column and row meanings are changed. The functions in this set work with labeled rows and columns and will shuffle the labels when appropriate. This and its value are seen in examples below.

a. Examples illustrating labeled rows and columns of Markov chains.
   Note the labeling for a Markov chain:

$$\textbf{chain} := \begin{bmatrix} state & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ s_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ s_2 & 1/4 & 1/2 & 0 & 1/4 & 0 & 0 \\ s_3 & 0 & 0 & 0 & 1 & 0 & 0 \\ s_4 & 1/16 & 1/4 & 1/8 & 1/4 & 1/4 & 1/16 \\ s_5 & 0 & 0 & 0 & 1/4 & 1/2 & 1/4 \\ s_6 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

   The normalize functions floats absorbing states to the top left. Note that column and row labels are still appropriate.

$$\textbf{markov\textbackslash normalize}\left(\textbf{chain}\right) \blacktriangleright \begin{bmatrix} state & s_1 & s_6 & s_2 & s_3 & s_4 & s_5 \\ s_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ s_6 & 0 & 1 & 0 & 0 & 0 & 0 \\ s_2 & \frac{1}{4} & 0 & \frac{1}{2} & 0 & \frac{1}{4} & 0 \\ s_3 & 0 & 0 & 0 & 0 & 1 & 0 \\ s_4 & \frac{1}{16} & \frac{1}{16} & \frac{1}{4} & \frac{1}{8} & \frac{1}{4} & \frac{1}{4} \\ s_5 & 0 & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{2} \end{bmatrix}$$

   Getting the Q and R subsets of the Markov chain maintain proper row and column labels.

markov\getqorr(chain,"Q") ▸

$$\begin{bmatrix} state & s_2 & s_3 & s_4 & s_5 \\ s_2 & \frac{1}{2} & 0 & \frac{1}{4} & 0 \\ s_3 & 0 & 0 & 1 & 0 \\ s_4 & \frac{1}{4} & \frac{1}{8} & \frac{1}{4} & \frac{1}{4} \\ s_5 & 0 & 0 & \frac{1}{4} & \frac{1}{2} \end{bmatrix}$$

markov\getqorr(chain,"R") ▸

$$\begin{bmatrix} state & s_1 & s_6 \\ s_2 & \frac{1}{4} & 0 \\ s_3 & 0 & 0 \\ s_4 & \frac{1}{16} & \frac{1}{16} \\ s_5 & 0 & \frac{1}{4} \end{bmatrix}$$

The fundamental and trend matrices are again appropriately labeled.

$$\mathbf{markov\backslash getfundamental}(\mathbf{chain}) \rightarrow \begin{bmatrix} state & s_2 & s_3 & s_4 & s_5 \\ s_2 & \frac{8}{3} & \frac{1}{6} & \frac{4}{3} & \frac{2}{3} \\ s_3 & \frac{4}{3} & \frac{4}{3} & \frac{8}{3} & \frac{4}{3} \\ s_4 & \frac{4}{3} & \frac{1}{3} & \frac{8}{3} & \frac{4}{3} \\ s_5 & \frac{2}{3} & \frac{1}{6} & \frac{4}{3} & \frac{8}{3} \end{bmatrix}$$

$$\mathbf{markov\backslash gettrend}(\mathbf{chain}) \rightarrow \begin{bmatrix} state & s_1 & s_6 \\ s_2 & \frac{3}{4} & \frac{1}{4} \\ s_3 & \frac{1}{2} & \frac{1}{2} \\ s_4 & \frac{1}{2} & \frac{1}{2} \\ s_5 & \frac{1}{4} & \frac{3}{4} \end{bmatrix}$$

b. Markov\newtransitions(numStates)
This function will generate a template markov chain with labeled rows and columns as follows. This template may be copied and pasted into another statement to edit and assign to a variable or do a computation as follows:

$$\textbf{markov\newtransitions}(6) \blacktriangleright \begin{bmatrix} state & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ s_1 & ? & ? & ? & ? & ? & ? \\ s_2 & ? & ? & ? & ? & ? & ? \\ s_3 & ? & ? & ? & ? & ? & ? \\ s_4 & ? & ? & ? & ? & ? & ? \\ s_5 & ? & ? & ? & ? & ? & ? \\ s_6 & ? & ? & ? & ? & ? & ? \end{bmatrix}$$

$$\textbf{chain}:=\begin{bmatrix} state & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ s_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ s_2 & 1/4 & 1/2 & 0 & 1/4 & 0 & 0 \\ s_3 & 0 & 0 & 0 & 1 & 0 & 0 \\ s_4 & 1/16 & 1/4 & 1/8 & 1/4 & 1/4 & 1/16 \\ s_5 & 0 & 0 & 0 & 1/4 & 1/2 & 1/4 \\ s_6 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

In addition to editing the internal Markov matrix, the row and column labels may be edited using anything that is meaningful and legal in a matrix cell. Typically, the labels will be symbolic variable names adhering to the legal syntax for the TI-Nspire calculator. These names are not used in any way other than labeling. i.e., no calculations, etc.

C. Markov\newstates(numStates)
This function will generate a state vector with labeled columns as follows. This template may be copied and pasted into another statement to edit and assign to a variable or do a computation as follows:

$$\textbf{markov\newstates}(6) \blacktriangleright \begin{bmatrix} state & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ \_ & ? & ? & ? & ? & ? & ? \end{bmatrix}$$

$$\textbf{nowstates}:=\begin{bmatrix} state & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ \_ & 0.1 & 0.2 & 0.2 & 0.3 & 0.1 & 0.1 \end{bmatrix}$$

In addition to editing the internal Markov state probabilities, the column labels may be edited using anything that is meaningful and legal in a matrix cell. Typically, the labels will be symbolic variable names adhering to the legal syntax for the TI-Nspire calculator and mirrowing the associated Markov chain column labels. These names are not used in any way other than labeling. i.e., no calculations, etc.

5. ## Regular Markov Chain Functions

This section contains the functions appropriate to a regular Markov chain.

a. Markov\isRegular(mat)

Is mat a regular Markov chain? This function will determine whether a Markov chain is regular. It is defined to be regular if some power of the chain contains all positive, non-zero cells.

$$\textbf{markov\textbackslash isregular}\left(\begin{bmatrix} state & s_1 & s_2 \\ s_1 & 0 & 1 \\ s_2 & 0.2 & 0.8 \end{bmatrix}\right) \blacktriangleright \text{true}$$

$$\textbf{markov\textbackslash isregular}\left(\begin{bmatrix} state & s_1 & s_2 & s_3 \\ s_1 & 0.4 & 0.2 & 0.4 \\ s_2 & 0 & 1 & 0 \\ s_3 & 0.6 & 0.3 & 0.1 \end{bmatrix}\right) \blacktriangleright \text{false}$$

b. Markov\getfixed(mat)

This function will compute the equilibrium state vector for a regular Markov chain.

$$\textbf{weight}:=\begin{bmatrix} state & thin & normal & overweight \\ thin & 0.3 & 0.5 & 0.2 \\ normal & 0.2 & 0.6 & 0.2 \\ overweight & 0.1 & 0.5 & 0.4 \end{bmatrix}$$

$$\textbf{markov\textbackslash getfixed}(\textbf{weight}) \blacktriangleright \begin{bmatrix} state & thin & normal & overwe \\ \_ & 0.194444 & 0.555556 & 0.2 \end{bmatrix}$$

6. ## Absorbing Markov Chain Functions

This section contains the functions appropriate to an absorbing Markov chain.

a. Markov\isabsorbing(mat)

This function will determine whether a Markov chain is absorbing. It is defined to an absorbing chain if all non-absorbing states have access to an absorbing state, possibly in multiple steps.

$$\textbf{markov\textbackslash isabsorbing}\left(\begin{bmatrix} state & thin & normal & overweight \\ thin & 0.3 & 0.5 & 0.2 \\ normal & 0.2 & 0.6 & 0.2 \\ overweight & 0.1 & 0.5 & 0.4 \end{bmatrix}\right) \blacktriangleright$$

$$\text{markov\textbackslash isabsorbing}\left(\begin{bmatrix} state & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ s_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ s_2 & 1/4 & 1/2 & 0 & 1/4 & 0 & 0 \\ s_3 & 0 & 0 & 0 & 1 & 0 & 0 \\ s_4 & 1/16 & 1/4 & 1/8 & 1/4 & 1/4 & 1/16 \\ s_5 & 0 & 0 & 0 & 1/4 & 1/2 & 1/4 \\ s_6 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}\right) \blacktriangleright \text{tru}$$

b. Markov\normalize(mat)
   Normalize the absorbing Markov chain. This will return the Markov chain with the absorbing states floated to the top left of the chain. Row and column labeling will be

adjusted accordingly.

$$\textbf{markov\textbackslash normalize}\begin{bmatrix} state & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ s_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ s_2 & 1/4 & 1/2 & 0 & 1/4 & 0 & 0 \\ s_3 & 0 & 0 & 0 & 1 & 0 & 0 \\ s_4 & 1/16 & 1/4 & 1/8 & 1/4 & 1/4 & 1/16 \\ s_5 & 0 & 0 & 0 & 1/4 & 1/2 & 1/4 \\ s_6 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\blacktriangleright \begin{bmatrix} state & s_1 & s_6 & s_2 & s_3 & s_4 & s_5 \\ s_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ s_6 & 0 & 1 & 0 & 0 & 0 & 0 \\ s_2 & \frac{1}{4} & 0 & \frac{1}{2} & 0 & \frac{1}{4} & 0 \\ s_3 & 0 & 0 & 0 & 0 & 1 & 0 \\ s_4 & \frac{1}{16} & \frac{1}{16} & \frac{1}{4} & \frac{1}{8} & \frac{1}{4} & \frac{1}{4} \\ s_5 & 0 & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{2} \end{bmatrix}$$

c. Markov\getFundamental(mat)
This function will compute the fundamental matrix for an absorbing Markov chain. The cells of this matrix will contain the number of times each non-absorbing state is expected to visit each non-absorbing state before finally entering an absorbing state.

Row and column labels are adjusted appropriately.

$$\textbf{markov\textbackslash getfundamental}\begin{bmatrix} state & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ s_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ s_2 & 1/4 & 1/2 & 0 & 1/4 & 0 & 0 \\ s_3 & 0 & 0 & 0 & 1 & 0 & 0 \\ s_4 & 1/16 & 1/4 & 1/8 & 1/4 & 1/4 & 1/16 \\ s_5 & 0 & 0 & 0 & 1/4 & 1/2 & 1/4 \\ s_6 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\blacktriangleright \begin{bmatrix} state & s_2 & s_3 & s_4 & s_5 \\ s_2 & \frac{8}{3} & \frac{1}{6} & \frac{4}{3} & \frac{2}{3} \\ s_3 & \frac{4}{3} & \frac{4}{3} & \frac{8}{3} & \frac{4}{3} \\ s_4 & \frac{4}{3} & \frac{1}{3} & \frac{8}{3} & \frac{4}{3} \\ s_5 & \frac{2}{3} & \frac{1}{6} & \frac{4}{3} & \frac{8}{3} \end{bmatrix}$$

d. Markov\getTrend(mat)
   This function will compute the long-term trend for an absorbing Markov chain. The cells of this matrix will note the probability that each non-absorbing state will eventually

enter each absorbing state. The row and column labels are adjusted appropriately.

$$\textbf{chain} \rightarrow \begin{bmatrix} state & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ s_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ s_2 & \frac{1}{4} & \frac{1}{2} & 0 & \frac{1}{4} & 0 & 0 \\ s_3 & 0 & 0 & 0 & 1 & 0 & 0 \\ s_4 & \frac{1}{16} & \frac{1}{4} & \frac{1}{8} & \frac{1}{4} & \frac{1}{4} & \frac{1}{16} \\ s_5 & 0 & 0 & 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ s_6 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\textbf{markov\backslash gettrend}(\textbf{chain}) \rightarrow \begin{bmatrix} state & s_1 & s_6 \\ s_2 & \frac{3}{4} & \frac{1}{4} \\ s_3 & \frac{1}{2} & \frac{1}{2} \\ s_4 & \frac{1}{2} & \frac{1}{2} \\ s_5 & \frac{1}{4} & \frac{3}{4} \end{bmatrix}$$

7. Miscellaneous Functions

This section contains miscellaneous functions appropriate to both regular and absorbing Markov chains.

a. Markov\about()

This displays information on the development of the set of functions.

```
markov\about() → [ "Markov Chain Programming Functions "
                   "Language: English "
                   "Version: 1.0 "
                   "Date: 9/26/2018 "
                   "Author: James O. Thompson "
                   "Email: jodaddy101@hotmail.com " ]
```

b. Markov\dopowers(mat,power)

Since the Markov chain matrices have row and column headers, regular arithmetic matrix syntax cannot be used. This function will extract the inner matrix and raise it to the specified power and return the result with appropriate row and column labels.

$$\textbf{markov\textbackslash dopowers}\left(\begin{bmatrix} state & S_1 & S_2 & S_3 & S_4 & S_5 & S_6 \\ S_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ S_2 & 1/4 & 1/2 & 0 & 1/4 & 0 & 0 \\ S_3 & 0 & 0 & 0 & 1 & 0 & 0 \\ S_4 & 1/16 & 1/4 & 1/8 & 1/4 & 1/4 & 1/16 \\ S_5 & 0 & 0 & 0 & 1/4 & 1/2 & 1/4 \\ S_6 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},3\right)$$

$$\begin{bmatrix} state & S_1 & S_2 & S_3 & S_4 & S_5 & S_6 \\ S_1 & 1 & 0 & 0 & 0 & 0 & 0 \\ S_2 & \frac{123}{256} & \frac{13}{64} & \frac{3}{128} & \frac{11}{64} & \frac{5}{64} & \frac{11}{256} \\ S_3 & \frac{9}{64} & \frac{3}{16} & \frac{1}{32} & \frac{5}{16} & \frac{3}{16} & \frac{9}{64} \\ S_4 & \frac{53}{256} & \frac{11}{64} & \frac{5}{128} & \frac{13}{64} & \frac{11}{64} & \frac{53}{256} \\ S_5 & \frac{11}{256} & \frac{5}{64} & \frac{3}{128} & \frac{11}{64} & \frac{13}{64} & \frac{123}{256} \\ S_6 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

C. Markov\doTransitions(vector,mat)
Since the Markov chain matrices have row and column headers, regular matrix operations cannot be performed. This function will extract the inner matrices and perform a (vector * matrix) transition operation returning the result with appropriate row and column labels.

weight:=

state	*w1*	*w2*	*w3*
w1	0.3	0.5	0.2
w2	0.2	0.6	0.2
w3	0.1	0.5	0.4

fixed:=**markov\getfixed** (**weight**) ▸

state	*w1*	*w2*	*w3*
_	0.19	0.56	0.25

markov\dotransitions (**fixed**,**weight**) ▸

state	*w1*	*w2*	*w3*
_	0.19	0.56	0.25

Helper Functions

These functions are not for the end user and are in the library as private functions. Because they are not for the end user they are not documented further. However, they are commented enough to facilitate usage to an experienced programmer.

d. addSubscript
e. colSwap
f. getMsg
g. getQorR
h. isNear
i. matInsert