Using Maxima to write a LaTeX article

Richel Bilderbeek

January 5, 2015

Abstract

This article is created from within the CAS program Maxima and converted to PDF by using FOSS tools only, to assess the viability of this pipeline.As a testcase, an exercise of [1] (page 69) is used.

1 Introduction

LaTeX is commonly used for writing publishable scientific articles[2]. Algebraic manipulations can be done by a CAS, for example Maxima, Maple or Mathematica. Maxima is the only free and open-source program, and it is the oldest free and open-source computer algebra system, with development started in 1967 (as Macsyma) or 1982 (as MAXIMA). This article is an example of writing a LaTeX article within Maxima

Writing LaTeX is slower and introduces more errors then using Microsoft Word as a text editor [3]. In this article, however, the text of a document is generated. LaTeX can parse that text to create a document, like a Word .docx could do as well, with less markup (XML) added. Additionally, all tools in the current pipeline are FOSS, so anyone with an internet connection can access these without paying licensing costs.

Maxima its capabilities is tested by doing an exercise of [1] at page 69.

2 Exercise

symbol	description
I	Indentity matrix
L	Leslie matrix
M	Leslie matrix with λ subtracted at diagonal
x	population density vector
Z	Vector filled with zeroes
λ	population growth rate

Table 1: Definitions

The equations to solve are:

$$x\,L = x\,\lambda \tag{1}$$

$$x\,M = Z \tag{2}$$

Equation 1 equals equation 3.21 of [1] (or P2.4a, page 237, in [4]) Equation 2 equals equation 3.23 of [1] (for definitions see table 1 on page 1). The Leslie matrix, L, given in this exercise is:

$$L = \begin{pmatrix} 1.0 & 1.0 \\ 0.8 & 0.8 \end{pmatrix} \tag{3}$$

The Leslie matrix, L, used:

$$L = \begin{pmatrix} 1.0 & 1.0 \\ 0.8 & 0.8 \end{pmatrix} \tag{4}$$

Note that this matrix has 2 age classes. The simplifies equation 1 to:

$$\begin{pmatrix} 1.0\,x & 1.0\,x \\ 0.8\,x & 0.8\,x \end{pmatrix} = x\,\lambda \tag{5}$$

Solving equation 1 can be done with equation 6:

$$det(L - \lambda * I) = Z \tag{6}$$

Equation 6 equals equation 3.24 of [1] (also: P2.4b at page 237, or 10.3.2 at page 406, in [4]) , where I is the identity matrix:

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{7}$$

And Z is the vector of zeroes:

$$Z = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{8}$$

This simplifies equation 6 to:

$$det\left(\begin{pmatrix} 1.0 - \lambda & 1.0 \\ 0.8 & 0.8 - \lambda \end{pmatrix}\right) = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{9}$$

The determinant of that matrix (M), is:

$$det(\begin{pmatrix} 1.0 - \lambda & 1.0 \\ 0.8 & 0.8 - \lambda \end{pmatrix}) = (0.8 - \lambda)\ (1.0 - \lambda) - 0.8 \tag{10}$$

Solving $M = 0$, the λs found are:

$$\left[\lambda = \frac{9}{5}, \lambda = 0\right] \tag{11}$$

There is one stable population structure, $\lambda = 0$, which is denotes an extinct population. Here I focus on the more interesting value, where $\lambda = 9/5$. This lambda is called the dominant eigenvalue, which equals the ultimate population growth.

Put λ in M, this results in:

$$M = \begin{pmatrix} -0.8 & 1.0 \\ 0.8 & -1.0 \end{pmatrix} \tag{12}$$

Now we can solve the x of equation 2 (equals 3.23 from [1]), which was this:

$$x\,M = Z \tag{13}$$

This equation is unsolvable, unless we assign a value to an element of x. Here, I put 1.0 as the initial value of x its first element. (it will be rescaled later):

$$x = \begin{pmatrix} 1.0 \\ x_2 \end{pmatrix} \tag{14}$$

Putting this x in equation 2:

$$M * x = \begin{pmatrix} 1.0\,x_2 - 0.8 \\ 0.8 - 1.0\,x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{15}$$

As our matrix has two rows, there are two equations that can be solved: Solving the upper, results in $x2 = 4/5$. This results in an x of:

$$x = \begin{pmatrix} 1.0 \\ \frac{4}{5} \end{pmatrix} \tag{16}$$

x must be rescaled so that its sum equals 1.0. x its current sum is 1.8, so dividing all elements by it, results in an x of:

$$x = \begin{pmatrix} 0.55555555555555 \\ 0.44444444444444 \end{pmatrix} \tag{17}$$

3 Conclusion

For this Leslie matrix:

$$\begin{pmatrix} 1.0 & 1.0 \\ 0.8 & 0.8 \end{pmatrix} \tag{18}$$

The dominant eigenvalue, λ, is:

$$\lambda = 9/5 \tag{19}$$

The stable population size distribution is:

$$x = \begin{pmatrix} 0.55555555555555 \\ 0.44444444444444 \end{pmatrix} \tag{20}$$

4 Discussion

Writing LaTeX within Maxima can be done, but it is a bit cumbersome: Maxima does not know LaTeX syntax and just creates contextless strings, which might not be compilable by LaTeX. However, because the script does create a .tex file, this file can be inspected easily with a LaTeX tool like texmaker.

References

[1] Case, Ted J. 2000 An illustrated guide to theoretical ecology.

[2] Gaudeul, A. 2006 Do Open Source Developers Respond to Competition?: The (La)TeX Case Study. Available at SSRN: http://ssrn.com/abstract=908946 or http://dx.doi.org/10.2139/ssrn.908946

[3] Knauff, M. and Nejasmic, J. December 19, 2014 An Efficiency Comparison of Document Preparation Systems Used in Academic Research and Development. PLoS ONE 9(12): e115069. doi: 10.1371/journal.pone.0115069

[4] Otto, Sarah P. and Day, T. 2007 A biologist's guide to mathematical modeling in ecology and evolution. ISBN-13: 978-0-691-12344-8

A Script file

```
#!/bin/bash
maxima_input_file="case_2000_69.txt"
tex_output_file="case_2000_69_output.tex"

if [ -e $tex_output_file ]
then
  rm $tex_output_file
fi

maxima -b $maxima_input_file
pdflatex $tex_output_file
pdflatex $tex_output_file
```

B Maxima file

```
/* Maxima batch file */

/* Load libraries */
load("stringproc")$

/* Input filename */
bash_filename:"case_2000_69.sh"$
maxima_filename:"case_2000_69.txt"$ /* this file */

/* Output filenames */
tex_filename:"case_2000_69_output.tex"$

/* Write results to TeX file */
stream: openw(tex_filename)$
printf(stream,"\\documentclass{article}~%")$
printf(stream,"~%")$
printf(stream,"\\usepackage{listings}~%")$
printf(stream,"\\usepackage{graphicx}~%")$
printf(stream,"~%")$
printf(stream,"\\title{Using Maxima to write a \\LaTeX~~
    article}~%")$
printf(stream,"\\author{Richel Bilderbeek}~%")$
printf(stream,"\\date{\\today}~%")$
printf(stream,"~%")$
printf(stream,"\\begin{document}~%")$
printf(stream,"~%")$
printf(stream,"\\maketitle~%")$
printf(stream,"~%")$
printf(stream,"\\begin{abstract}~%")$
printf(stream,"This article is created from within the
    CAS program Maxima~%")$
printf(stream,"and converted to PDF by using FOSS tools
    only, to assess the viability of this pipeline.")$
printf(stream,"As a testcase, an exercise of \\cite{
    case2000} (page 69) is used.~%")$
printf(stream,"\\end{abstract}~%")$
printf(stream,"~%")$
printf(stream,"\\section{Introduction}~%")$
printf(stream,"~%")$
printf(stream,"\\LaTeX~~is commonly used for writing
    publishable scientific articles\\cite{gaudeul2006}.~%"
    )$
```

```
printf(stream,"Algebraic manipulations can be done by a
   CAS, for example Maxima, Maple or Mathematica.~%")$
printf(stream,"Maxima is the only free and open-source
   program, and it is the oldest free and open-source
   computer algebra system, with development started in
   1967 (as Macsyma) or 1982 (as MAXIMA).~%")$
printf(stream,"This article is an example of writing a \\
   LaTeX~~ article within Maxima~%")$
printf(stream,"~%")$
printf(stream,"Writing \\LaTeX~~ is slower and introduces
    more errors ~%")$
printf(stream,"then using Microsoft Word as a text editor
    \\cite{knauff&nejasmic2014}.~%")$
printf(stream,"In this article, however, the text of a
   document is generated. \\LaTeX~~ can~%")$
printf(stream,"parse that text to create a document, like
    a Word .docx could do as well,~%")$
printf(stream,"with less markup (XML) added. Additionally
   , all tools in the current pipeline~%")$
printf(stream,"are FOSS, so anyone with an internet
   connection can access these without~%")$
printf(stream,"paying licensing costs.~%")$
printf(stream,"~%")$
printf(stream,"Maxima its capabilities is tested by doing
    an exercise of \\cite{case2000} at page 69.~%")$
printf(stream,"~%")$
printf(stream,"\\section{Exercise}~%")$
printf(stream,"~%")$
printf(stream,"\\begin{table}[here]~%")$
printf(stream,"  \\centering~%")$
printf(stream,"  \\begin{tabular}{ | r | l | }~%")$
printf(stream,"    \\hline~%")$
printf(stream,"    symbol & description \\\\~%")$
printf(stream,"    \\hline~%")$
printf(stream,"    $I$ & Indentity matrix \\\\~%")$
printf(stream,"    $L$ & Leslie matrix \\\\~%")$
printf(stream,"    $M$ & Leslie matrix with $\\lambda$
   subtracted at diagonal \\\\~%")$
printf(stream,"    $x$ & population density vector \\\\~%
   ")$
printf(stream,"    $Z$ & Vector filled with zeroes \\\\~%
   ")$
printf(stream,"    $\\lambda$ & population growth rate
   \\\\~%")$
printf(stream,"    \\hline~%")$
printf(stream,"  \\end{tabular}~%")$
```

```
printf(stream,"  \\caption{Definitions}~%")$
printf(stream,"  \\label{table:table_definition}~%")$
printf(stream,"\\end{table}~%")$
printf(stream,"~%")$
printf(stream,"The equations to solve are:~%")$

Eq3_21(L,x,lambda) := L * x = lambda * x;

printf(stream,"\\begin{equation}~%")$
printf(stream,tex1(Eq3_21(L,x,lambda)))$ /* Use tex1
    instead of tex(...,false) for non $$-enclosed output
    */
printf(stream,"\\label{eq:eq3_21}~%")$
printf(stream,"\\end{equation}~%")$

Eq3_23(M,x,Z) := M * x = Z;
Eq3_23_str:tex1(Eq3_23(M,x,Z));

printf(stream,"\\begin{equation}~%")$
printf(stream,tex1(Eq3_23(M,x,Z)))$ /* Use tex1 instead
    of tex(...,false) for non $$-enclosed output */
printf(stream,"\\label{eq:eq3_23}~%")$
printf(stream,"\\end{equation}~%")$

printf(stream,"~%")$
printf(stream,"Equation \\ref{eq:eq3_21} equals equation
    3.21 of \\cite{case2000} (or P2.4a, page 237, in \\
    cite{otto&day2007})~%")$
printf(stream,"Equation \\ref{eq:eq3_23} equals equation
    3.23 of \\cite{case2000} ~%")$
printf(stream,"(for definitions see table \\ref{table:
    table_definition} on page \\pageref{table:
    table_definition}).~%")$
printf(stream,"The Leslie matrix, L,  given in this
    exercise is:~%")$
printf(stream,"~%")$

L_original:matrix([1.0,1.0],[0.8,0.8]);

printf(stream,"\\begin{equation}~%")$
printf(stream,"L=")$
printf(stream,tex1(L_original))$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$

L:L_original;
```

```
/*L:matrix([0.1,2.0],[0.1,0.8]); */
/* L:matrix([0.1,1.0,0.0],[0.8,0.8,0.0],[0.8,0.8,0.0]);
   */

printf(stream,"The Leslie matrix, L, used:~%")$
printf(stream,"~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,"L=")$
printf(stream,tex1(L))$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$

n_age_classes:matrix_size(L)[1];

printf(stream,"~%")$
printf(stream,"Note that this matrix has ")$
printf(stream,string(n_age_classes))$
printf(stream," age classes.~%")$
printf(stream,"The simplifies equation \\ref{eq:eq3_21}
   to:~%")$
printf(stream,"~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,tex1(Eq3_21(L,x,lambda)))$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$
printf(stream,"Solving equation \\ref{eq:eq3_21} can be
   done with equation \\ref{eq:eq3_24}:~%")$
printf(stream,"~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,"det(L - \\lambda*I) = Z~%")$
printf(stream,"\\label{eq:eq3_24}~%")$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$

I:ident(n_age_classes);
Z:zeromatrix(n_age_classes,1);

printf(stream,"Equation \\ref{eq:eq3_24} equals equation
   3.24 of \\cite{case2000} (also: P2.4b at page 237, or
   10.3.2 at page 406, in \\cite{otto&day2007}) ,~%")$
printf(stream,"where I is the identity matrix:~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,"I=")$
printf(stream,tex1(I))$
printf(stream,"\\end{equation}~%")$
printf(stream,"And Z is the vector of zeroes:~%")$
```

```
printf(stream,"\\begin{equation}~%")$
printf(stream,"Z=")$
printf(stream,tex1(Z))$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$

Eq3_24(L,lambda,I,Z) := det(L - lambda * I) = Z;

printf(stream,"This simplifies equation \\ref{eq:eq3_24}
    to:~%")$
printf(stream,"~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,tex1(Eq3_24(L,lambda,I,Z)))$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$

/* Great that M is in Eq3_24_b, but I cannot get it out,
    so I do it manually*/
M:copymatrix(L);
D: diagmatrix (n_age_classes, lambda);
M:M-D;

printf(stream,"~%")$
printf(stream,"The determinant of that matrix ($M$), is
    :~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,"det(")$
printf(stream,tex1(M))$
printf(stream,")=")$
printf(stream,tex1(determinant(M)))$
printf(stream,"~%")$printf(stream,"\\end{equation}~%")$


stable_lambdas : solve(determinant(M) = 0,lambda);

printf(stream,"~%")$
printf(stream,"Solving $M=0$, the $\\lambda$s found are
    :~%")$
printf(stream,"~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,tex1(stable_lambdas))$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$
printf(stream,"There is one stable population structure,
    $\\lambda=0$, which is~%")$
printf(stream,"denotes an extinct population.~%")$
```

```
lambda:rhs(stable_lambdas[1]);

printf(stream,"Here I focus on the more interesting value
    ,~%")$
printf(stream,"where $\\lambda=")$
printf(stream,string(lambda))$ /* Don't forget the string
     function */
printf(stream,"$.~%")$
printf(stream,"This lambda is called the dominant
    eigenvalue, which equals the ultimate population
    growth.~%")$
printf(stream,"~%")$

M:''(M); /* Filling it in */

printf(stream,"Put $\\lambda$ in M, this results in:~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,"M=")$
printf(stream,tex1(M))$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$


printf(stream,"Now we can solve the $x$ of equation \\ref
    {eq:eq3_23} (equals 3.23 from \\cite{case2000}),~%")$
printf(stream,"which was this:~%")$
printf(stream,"~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,Eq3_23_str)$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$
printf(stream,"This equation is unsolvable, unless we
    assign a value to an element of $x$.~%")$
printf(stream,"Here, I put $1.0$ as the initial value of
    $x$ its first element.~%")$
printf(stream,"(it will be rescaled later):~%")$
printf(stream,"~%")$


x:transpose(matrix([1.0,x2]));

if n_age_classes=3
then
  x:transpose(matrix([1.0,x2,x3]))
;
```

```
if n_age_classes=4
then
  printf(stream,"NOT SUPPORTED 4 AGE CLASSES YET!~%")
;


printf(stream,"\\begin{equation}~%")$
printf(stream,"x = ")$
printf(stream,tex1(x))$
printf(stream,"\\end{equation}~%")$

EqSolve(M,x,Z):= M.x = Z;

printf(stream,"Putting this $x$ in equation \\ref{eq:
    eq3_23}:~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,"M * x = ")$
printf(stream,tex1(lhs(EqSolve(M,x,Z))))$
printf(stream,"= ")$
printf(stream,tex1(Z))$
printf(stream,"\\end{equation}~%")$
printf(stream,"As our matrix has two rows, there are two
    equations that can be solved:~%")$

q:''(M.x)[1,1];
x2:rhs(solve(q)[1]);

printf(stream,"Solving the upper, results in $x2=")$
printf(stream,string(x2))$
printf(stream,"$.~%")$
printf(stream,"This results in an $x$ of:~%")$

x;
x:''(x);

printf(stream,"\\begin{equation}~%")$
printf(stream,"x = ")$
printf(stream,tex1(x))$
printf(stream,"\\end{equation}~%")$
printf(stream,"$x$ must be rescaled so that its sum
    equals $1.0$.~%")$

sz: matrix_size(x);
my_sum: sum(sum(x[i,j],i,1,sz[1]),j,1,sz[2]);
```

```
printf(stream,"$x$ its current sum is $")$
printf(stream,string(my_sum))$
printf(stream,"$, so dividing all elements by it, results
    in an $x$ of:~%")$

x:x/my_sum;
x;

printf(stream,"\\begin{equation}~%")$
printf(stream,"x = ")$
printf(stream,tex1(x))$
printf(stream,"\\end{equation}~%")$

/* Conclusion */

printf(stream,"\\section{Conclusion}~%")$
printf(stream,"~%")$
printf(stream,"For this Leslie matrix:~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,tex1(L))$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$
printf(stream,"The dominant eigenvalue, $\\lambda$, is:~%
   ")$
printf(stream,"\\\\\\\\~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,"\\lambda = ")$
printf(stream,string(lambda))$ /* Don't forget the string
    function */
printf(stream,"\\end{equation}~%")$
printf(stream,"\\\\~%")$
printf(stream,"~%")$
printf(stream,"The stable population size distribution is
   :~%")$
printf(stream,"~%")$
printf(stream,"\\begin{equation}~%")$
printf(stream,"x = ")$
printf(stream,tex1(x))$
printf(stream,"\\end{equation}~%")$
printf(stream,"~%")$

/* Discussion */

printf(stream,"\\section{Discussion}~%")$
printf(stream,"~%")$
```

```
printf(stream,"Writing \\LaTeX~~within Maxima can be done
    , but it is a bit cumbersome:~%")$
printf(stream,"Maxima does not know \\LaTeX~~syntax and
    just creates contextless strings,~%")$
printf(stream,"which might not be compilable by \\LaTeX
    .~%")$
printf(stream,"However, because the script does create a
    .tex file,~%")$
printf(stream,"this file can be inspected easily with a
    \\LaTeX~~tool like texmaker.~%")$
printf(stream,"~%")$

/*
Why can't it be solved simpler by stating
X: population vector at t
Y: population vector at t+1

and solving when Y has the same relative proportions than
    X

?
*/
/*
kill(X);
kill(Y);
L:matrix([1.0,1.0],[0.8,0.8]);
Eq1(X,Y,L) := Y = L.X;
Eq2(X,Y) := X[1] / (X[1] + X[2]) = Y[1] / (Y[1] + Y[2]);
Eq3(X,Y) := X[2] / (X[1] + X[2]) = Y[2] / (Y[1] + Y[2]);
X:transpose(matrix([xa,xb]));
Y:transpose(matrix([ya,yb]));
*/



/* Bibliography */

printf(stream,"\\begin{thebibliography}{9}~%")$
printf(stream,"~%")$
printf(stream,"\\bibitem{case2000}~%")$
printf(stream,"  Case, Ted J.~%")$
printf(stream,"  2000~%")$
printf(stream,"  An illustrated guide to theoretical
    ecology.~%")$
printf(stream,"~%")$
printf(stream,"\\bibitem{gaudeul2006}~%")$
```

```
printf(stream,"  Gaudeul, A.~%")$
printf(stream,"  2006~%")$
printf(stream,"  Do Open Source Developers Respond to
    Competition?: The (La)TeX Case Study.~%")$
printf(stream,"  Available at SSRN: http://ssrn.com/
    abstract=908946 or http://dx.doi.org/10.2139/ssrn
    .908946~%")$
printf(stream,"~%")$
printf(stream,"\\bibitem{knauff&nejasmic2014}~%")$
printf(stream,"  Knauff, M. and Nejasmic, J.~%")$
printf(stream,"  December 19, 2014~%")$
printf(stream,"  An Efficiency Comparison of Document
    Preparation Systems Used in Academic Research and
    Development.~%")$
printf(stream,"  PLoS ONE 9(12): e115069. doi: 10.1371/
    journal.pone.0115069~%")$
printf(stream,"~%")$
printf(stream,"\\bibitem{otto&day2007}~%")$
printf(stream,"  Otto, Sarah P. and Day, T.~%")$
printf(stream,"  2007~%")$
printf(stream,"  A biologist's guide to mathematical
    modeling in ecology and evolution.~%")$
printf(stream,"  ISBN-13: 978-0-691-12344-8~%")$
printf(stream,"~%")$




printf(stream,"\\end{thebibliography}~%")$
printf(stream,"~%")$

/* Appendix */

printf(stream,"\\appendix~%")$
printf(stream,"~%")$
printf(stream,"\\section{Script file}~%")$
printf(stream,"~%")$
printf(stream,"\\lstinputlisting[language=C++,
    showstringspaces=false,breaklines=true,frame=single]{"
    )$
printf(stream,bash_filename)$
printf(stream,"}~%")$
printf(stream,"~%")$
printf(stream,"\\section{Maxima file}~%")$
printf(stream,"~%")$
```

```
printf(stream,"\\lstinputlisting[language=C++,
    showstringspaces=false,breaklines=true,frame=single]{"
    )$
printf(stream,maxima_filename)$
printf(stream,"}~%")$
printf(stream,"~%")$
printf(stream,"\\section{\\LaTeX~~file}~%")$
printf(stream,"~%")$
printf(stream,"\\lstinputlisting[language=tex,
    showstringspaces=false,breaklines=true,frame=single]{"
    )$
printf(stream,tex_filename)$
printf(stream,"}~%")$
printf(stream,"~%")$
printf(stream,"\\end{document}~%")$
close(stream)$
```

C LaTeX file

```
\documentclass{article}

\usepackage{listings}
\usepackage{graphicx}

\title{Using Maxima to write a \LaTeX~ article}
\author{Richel Bilderbeek}
\date{\today}

\begin{document}

\maketitle

\begin{abstract}
This article is created from within the CAS program
    Maxima
and converted to PDF by using FOSS tools only, to assess
    the viability of this pipeline.As a testcase, an
    exercise of \cite{case2000} (page 69) is used.
\end{abstract}

\section{Introduction}

\LaTeX~is commonly used for writing publishable
    scientific articles\cite{gaudeul2006}.
```

```
Algebraic manipulations can be done by a CAS, for example
    Maxima, Maple or Mathematica.
Maxima is the only free and open-source program, and it
    is the oldest free and open-source computer algebra
    system, with development started in 1967 (as Macsyma)
    or 1982 (as MAXIMA).
This article is an example of writing a \LaTeX~ article
    within Maxima

Writing \LaTeX~ is slower and introduces more errors
then using Microsoft Word as a text editor \cite{knauff&
    nejasmic2014}.
In this article, however, the text of a document is
    generated. \LaTeX~ can
parse that text to create a document, like a Word .docx
    could do as well,
with less markup (XML) added. Additionally, all tools in
    the current pipeline
are FOSS, so anyone with an internet connection can
    access these without
paying licensing costs.

Maxima its capabilities is tested by doing an exercise of
    \cite{case2000} at page 69.

\section{Exercise}

\begin{table}[here]
  \centering
  \begin{tabular}{ | r | l | }
    \hline
    symbol & description \\
    \hline
    $I$ & Indentity matrix \\
    $L$ & Leslie matrix \\
    $M$ & Leslie matrix with $\lambda$ subtracted at
        diagonal \\
    $x$ & population density vector \\
    $Z$ & Vector filled with zeroes \\
    $\lambda$ & population growth rate \\
    \hline
  \end{tabular}
  \caption{Definitions}
  \label{table:table_definition}
\end{table}
```

```
The equations to solve are:
\begin{equation}
x\,L=x\,\lambda\label{eq:eq3_21}
\end{equation}
\begin{equation}
x\,M=Z\label{eq:eq3_23}
\end{equation}

Equation \ref{eq:eq3_21} equals equation 3.21 of \cite{
    case2000} (or P2.4a, page 237, in \cite{otto&day2007})
Equation \ref{eq:eq3_23} equals equation 3.23 of \cite{
    case2000}
(for definitions see table \ref{table:table_definition}
    on page \pageref{table:table_definition}).
The Leslie matrix, L,  given in this exercise is:

\begin{equation}
L=\pmatrix{1.0&1.0\cr 0.8&0.8\cr }\end{equation}

The Leslie matrix, L, used:

\begin{equation}
L=\pmatrix{1.0&1.0\cr 0.8&0.8\cr }\end{equation}


Note that this matrix has 2 age classes.
The simplifies equation \ref{eq:eq3_21} to:

\begin{equation}
\pmatrix{1.0\,x&1.0\,x\cr 0.8\,x&0.8\,x\cr }=x\,\lambda\
    end{equation}

Solving equation \ref{eq:eq3_21} can be done with
    equation \ref{eq:eq3_24}:

\begin{equation}
det(L - \lambda*I) = Z
\label{eq:eq3_24}
\end{equation}

Equation \ref{eq:eq3_24} equals equation 3.24 of \cite{
    case2000} (also: P2.4b at page 237, or 10.3.2 at page
    406, in \cite{otto&day2007}) ,
where I is the identity matrix:
\begin{equation}
I=\pmatrix{1&0\cr 0&1\cr }\end{equation}
```

```
And Z is the vector of zeroes:
\begin{equation}
Z=\pmatrix{0\cr 0\cr }\end{equation}

This simplifies equation \ref{eq:eq3_24} to:

\begin{equation}
{\it det}\left(\pmatrix{1.0-\lambda&1.0\cr 0.8&0.8-\
    lambda\cr }\right)=\pmatrix{0\cr 0\cr }\end{equation}


The determinant of that matrix ($M$), is:
\begin{equation}
det(\pmatrix{1.0-\lambda&1.0\cr 0.8&0.8-\lambda\cr })=\
    left(0.8-\lambda\right)\,,\left(1.0-\lambda\right)-0.8
\end{equation}

Solving $M=0$, the $\lambda$s found are:

\begin{equation}
\left[ \lambda={{9}\over{5}} , \lambda=0 \right] \end{
    equation}

There is one stable population structure, $\lambda=0$,
    which is
denotes an extinct population.
Here I focus on the more interesting value,
where $\lambda=9/5$.
This lambda is called the dominant eigenvalue, which
    equals the ultimate population growth.

Put $\lambda$ in M, this results in:
\begin{equation}
M=\pmatrix{-0.8&1.0\cr 0.8&-1.0\cr }\end{equation}

Now we can solve the $x$ of equation \ref{eq:eq3_23} (
    equals 3.23 from \cite{case2000}),
which was this:

\begin{equation}
x\,M=Z\end{equation}

This equation is unsolvable, unless we assign a value to
    an element of $x$.
Here, I put $1.0$ as the initial value of $x$ its first
    element.
```

```
(it will be rescaled later):

\begin{equation}
x = \pmatrix{1.0\cr {\it x_2}\cr }\end{equation}
Putting this $x$ in equation \ref{eq:eq3_23}:
\begin{equation}
M * x = \pmatrix{1.0\,{\it x_2}-0.8\cr 0.8-1.0\,{\it x
    _2}\cr }= \pmatrix{0\cr 0\cr }\end{equation}
As our matrix has two rows, there are two equations that
    can be solved:
Solving the upper, results in $x2=4/5$.
This results in an $x$ of:
\begin{equation}
x = \pmatrix{1.0\cr {{4}\over{5}}\cr }\end{equation}
$x$ must be rescaled so that its sum equals $1.0$.
$x$ its current sum is $1.8$, so dividing all elements by
     it, results in an $x$ of:
\begin{equation}
x = \pmatrix{0.55555555555555\cr 0.44444444444444\cr }\
    end{equation}
\section{Conclusion}

For this Leslie matrix:
\begin{equation}
\pmatrix{1.0&1.0\cr 0.8&0.8\cr }\end{equation}

The dominant eigenvalue, $\lambda$, is:
\\\\
\begin{equation}
\lambda = 9/5\end{equation}
\\

The stable population size distribution is:

\begin{equation}
x = \pmatrix{0.55555555555555\cr 0.44444444444444\cr }\
    end{equation}

\section{Discussion}

Writing \LaTeX~within Maxima can be done, but it is a bit
     cumbersome:
Maxima does not know \LaTeX~syntax and just creates
    contextless strings,
which might not be compilable by \LaTeX.
However, because the script does create a .tex file,
```

```
this file can be inspected easily with a \LaTeX~tool like
    texmaker.

\begin{thebibliography}{9}

\bibitem{case2000}
  Case, Ted J.
  2000
  An illustrated guide to theoretical ecology.

\bibitem{gaudeul2006}
  Gaudeul, A.
  2006
  Do Open Source Developers Respond to Competition?: The
      (La)TeX Case Study.
  Available at SSRN: http://ssrn.com/abstract=908946 or
      http://dx.doi.org/10.2139/ssrn.908946

\bibitem{knauff&nejasmic2014}
  Knauff, M. and Nejasmic, J.
  December 19, 2014
  An Efficiency Comparison of Document Preparation
      Systems Used in Academic Research and Development.
  PLoS ONE 9(12): e115069. doi: 10.1371/journal.pone
      .0115069

\bibitem{otto&day2007}
  Otto, Sarah P. and Day, T.
  2007
  A biologist's guide to mathematical modeling in ecology
       and evolution.
  ISBN-13: 978-0-691-12344-8

\end{thebibliography}

\appendix

\section{Script file}

\lstinputlisting[language=C++,showstringspaces=false,
    breaklines=true,frame=single]{case_2000_69.sh}

\section{Maxima file}

\lstinputlisting[language=C++,showstringspaces=false,
    breaklines=true,frame=single]{case_2000_69.txt}
```

```
\section{\LaTeX~file}

\lstinputlisting[language=tex,showstringspaces=false,
    breaklines=true,frame=single]{case_2000_69_output.tex}

\end{document}
```