

MATweave: Integración de código MATLAB/Octave en L^AT_EX*

Mario Mañana Canteli
mananam@unican.es
Dpto. de Ingeniería Eléctrica y Energética
Universidad de Cantabria

07/02/2017

Abstract

MATweave define un procedimiento para integrar código MATLAB/Octave dentro de documentos L^AT_EX. El objetivo final es facilitar a los usuarios de MATLAB/Octave la generación integral de investigación reproducible.

1 Introducción

MATweave proporciona un método simple para la generación de informes que integran datos y código utilizando L^AT_EX como lenguaje de edición, y puede considerarse como un sistema básico para la generación de documentos basados en el paradigma de la investigación reproducible. El objetivo final de este conjunto de herramientas es obtener un entregable que integre documentación, datos y código que pueda, además, ser reutilizado por otros usuarios. MATweave está inspirado en Sweave [1], diseñado originalmente para combinar programas en R y L^AT_EX.

1.1 Procedimiento

La idea que subyace detrás de MATweave es utilizar códigos específicos para indicar el inicio y fin del fragmento de código dentro del fichero L^AT_EX.

1. El primer procedimiento es utilizar códigos de comentario tipo bloque, introducidos en Octave 3.2 and MATLAB R14.

Los comentarios tipo bloque en MATLAB/Octave se abren con `%{` y se cierran con `%}` permitiendo incluir código dentro:

```
%{  
Esto es un comentario!  
%}  
for i = 1:10  
    % Este codigo esta fuera del comentario.  
    disp(i)  
end
```

El ‘truco’ consiste en que un bloque de comentario en MATLAB/Octave es un comentario en L^AT_EX (debido a que comienza con el carácter `%`), pero *no* un bloque de comentario.

* Adaptado de Neil D. Lawrence

```
%{
En MATLAB/Octave esto sería un comentario, pero en \LaTeX sí se compilaría.
Esto significa que es posible escribir código \LaTeX, como por ejemplo
 $\tau = 2\pi$ , dentro de un script MATLAB.
Ahora debemos ser capaces de escribir MATLAB/Octave dentro de un
fichero \LaTeX.
%}
```

2. El segundo ‘truco’ es incluir el paquete `verbatim` y utilizarlo para definir un nuevo entorno MATLAB/Octave utilizando los comandos siguientes:

```
\newenvironment{matlab}{\comment}{\endcomment}
\newenvironment{octave}{\comment}{\endcomment}
\newenvironment{matlabv}{\verbatim}{\endverbatim}
\newenvironment{octavev}{\verbatim}{\endverbatim}
```

Esto permitirá incluir código MATLAB/Octave que no será leído por \LaTeX utilizando: `\begin{octave}` ... `\end{octave}`, así como código que será mostrado en un entorno `verbatim` utilizando `\begin{octavev}` ... `\end{octavev}`. Por supuesto es posible hacer esto mismo utilizando el entorno estándar `verbatim`. Sin embargo, utilizando un nuevo entorno es posible definir los fragmentos de código que se muestran y/o se ejecutan.

1.2 Ejemplo

El código siguiente se ha desarrollado utilizando el entorno `matlabv` definido anteriormente. El código fuente de este documento comienza con un código de bloque abierto MATLAB/Octave `%{`. Dicho código se cierra al comiendo de un bloque MATLAB/Octave, de forma que será compilado por dichos programas.

```
%}
% El código MATLAB/Octave comienza aqui
tau = 2*pi;
x = linspace(-3, 3, 100)';
y = 1/sqrt(tau)*exp(-0.5*x.^2);
plot(x, y, 'r-');
set(gca, 'fontname', fontName, 'fontsize', fontSize);
print -dpdf myGaussian.pdf
% El código MATLAB/Octave finaliza aqui
%{
```

La combinación de dos bloques significa que el código fuente de este documento puede ser compilado en \LaTeX o en MATLAB/Octave.

2 Ejecutando MATweave

El procedimiento para generar el documento se resume a continuación:

- Ejecutar el comando ‘`source fichero.tex`’ en Matlab/Octave.
- Ejecutar el comando ‘`pdflatex fichero.tex`’ en la línea de comandos del sistema operativo. Si el documento incluye referencias bibliográficas es necesario realizar una secuencia:
 - ‘`pdflatex fichero.tex`’
 - ‘`bibtex fichero.tex`’

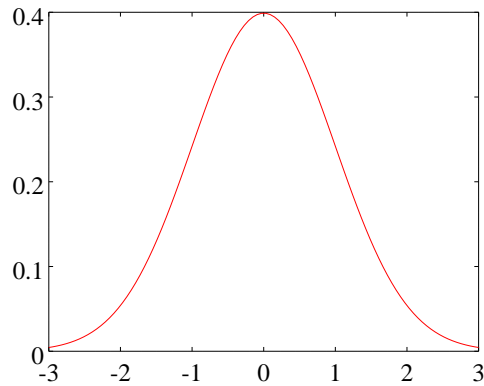


Figure 1: Curva de densidad de probabilidad Gausiana.

– ‘pdflatex fichero.tex’

Como se ha comentado, el fichero \LaTeX puede ser ejecutado en Octave con el comando `source`. Como ejercicio, guardar este documento como `myexample.tex` y ejecutar `source myexample.tex` en la línea de comandos de Octave. En MATLAB el procedimiento es un poco más complicado.

La salida del código incluye figuras en formato `.pdf`. La primera de las figuras se muestra en la Figura 1.

El código MATLAB/Octave puede ser mostrado u oculto en el documento según convenga en cada momento atendiendo a la necesidad de documentar un procedimiento de cálculo e incluso por motivaciones de depuración del código. A modo de ejemplo, el histograma mostrado en la Figura 2 ha sido codificado utilizando el entorno `\begin{matlab} ... \end{matlab}`.

Otra posibilidad que puede mejorar la visualización de código es utilizar el paquete `listings`. Otra posibilidad para mejorar la automatización del documento es utilizar MATLAB/Octave para generar un fichero `.tex` con la tabla que se pretenda visualizar en el documento \LaTeX .

```
%}
% Código para generar una tabla con números aleatorios.
rows = 3;
cols = 4;
numSigFigs = 3;
resultMatrix = randn(3, 4);
fid = fopen('results.tex', 'w');
for i = 1:rows
    for j = 1:cols
        fprintf(fid, ['$' num2str(resultMatrix(i, j), numSigFigs) '$']);
        if j < cols
```

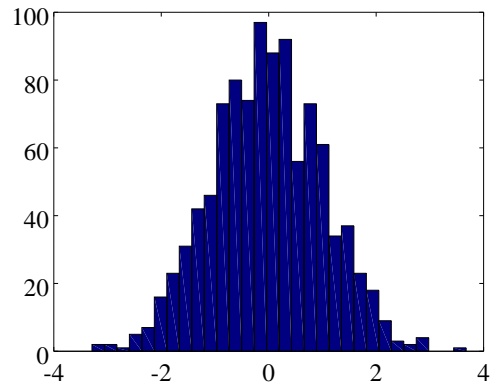


Figure 2: Histograma con 1000 muestras de una distribución Gaussiana.

Table 1: Ejemplo de una tabla de números aleatorios con MATLAB/Octave.

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>
-0.442	1.59	0.302	-1.5
-0.418	2.39	-0.123	0.632
-1.46	1.08	0.128	-2.35

```

        fprintf(fid, ' & ');
    end
end
if i < rows
    fprintf(fid, '\\\\\\\\n');
end
end
fclose(fid);
%{

```

El fichero de resultados con la tabla puede ser incorporado al documento mediante el código `\input{results.tex}` y mostrado como aparece en la Tabla 1.

2.1 Consejos adicionales

Se considera una buena medida comenzar el fichero \LaTeX con un conjunto de comandos que limpien la memoria, definan las rutas necesarias, borren las figuras, etc.

2.2 Nota para usuarios de Beamer

Aquellos usuarios de MATweave que generen presentaciones utilizando Beamer deben utilizar la opción `[fragile]` en aquellas transparencias que contengan el entorno MATLAB/Octave. Sin dicha opción Beamer no será capaz de manejar los entornos `verbatim` o `comment`.

Se necesita una versión MATLAB R14 o superior y Octave 3.2 o superior. Para compilar el fichero en Linux:

```
octave --eval source\ myexample.tex
```

o

```
matlab < myexample.tex
```

y después,

```
pdflatex myexample
```

3 Conclusiones

Se ha mostrado que la integración de código MATLAB/Octave con \LaTeX puede realizarse mediante procedimientos sencillos. El procedimiento no es tan elegante como la pareja Sweave con R, donde tanto las variables como las gráficas se integran en el texto de forma natural.

El objetivo de MATweave es facilitar la generación de documentos reproducibles.

Nota

Este documento fue generado utilizando MATweav utilizando Octave version 4.0.0 en una máquina i686-w64-mingw32. El documento fue generado el 07/02/2017.

References

- [1] Friedrich Leisch. Sweave: Dynamic generation of statistical reports using literate data analysis. In Wolfgang Härdle and Bernd Rönz, editors, *Compstat 2002 — Proceedings in Computational Statistics*, pages 575–580. Physica Verlag, Heidelberg, 2002. ISBN 3-7908-1517-9.