

Documentación para **EA3examen** y herramientas relacionadas

26 de marzo de 2020

Índice

1. Introducción	1
2. Modus operandi	2
3. Macrodefiniciones	3
3.1. Locuciones frecuentes	3
3.2. Notación matemática	3
4. Exámenes para Moodle (= e-Gela)	7
4.1. Ex-novo	7
4.2. A partir de exámenes pre-existentes	9
A. Descripción de operaciones que realiza moodleiza.	11
B. Licencias	12

1. Introducción

EA3examen es un conjunto de plantillas, macros en \LaTeX y funciones en R y en ELISP para Emacs que buscan facilitar la producción de exámenes multilingües con múltiples modelos en cada idioma. Pueden utilizarse sólo partes, como las definiciones en \LaTeX , que pueden emplearse en cualquier documento, examen o no, en uno o en varios idiomas.

La producción de exámenes hace uso del paquete **examdesign**¹.

¹Dsponible en CTAN, <http://mirror.ox.ac.uk/sites/ctan.org/macros/latex/contrib/examdesign/>.

2. Modus operandi

El flujo de trabajo se recoge esquemáticamente en la Figura 1, pág. 3. El usuario prepara el fichero maestro alternativamente en formato `.Rnw` o `.tex`. El primer formato se emplea cuando hay material que requiere ser procesado por R (cálculos, tablas estadísticas, gráficas. etc.).

Tal material se inserta en un fichero nombrado con extensión `.Rnw` en bloques tales como

```
«»=  
library(MASS)  
@
```

en que los delimitadores `«»=` y `@` señalan fragmentos de código que se ejecuta e inserta automáticamente en un nuevo fichero en formato `LATEX` standard. Las variantes idiomáticas, por su parte, se codifican señalando las líneas en cada idioma con un prefijo, del modo siguiente:

```
ES- Estadística y probabilidad...  
EN- Statistics and probability...  
EU- Estatistika eta probabilitatea...
```

Estas líneas se copian tal cual al fichero `.tex` al procesar el fichero `.Rnw`. Algunas funciones en Emacs ayudan a editar el texto variante.

En la etapa posterior, se generan las versiones para los idiomas que se hayan configurado. Un único golpe de tecla en Emacs genera a partir del documento precursor tantos otros como idiomas, con nombres iguales al del precursor prefijados por el idioma (`EN.fichero.tex`, `ES.fichero.tex`, etc.). Éstos son tratados como ficheros `LATEX` ordinarios que son, dando lugar cada uno al documento final en un idioma y el formato que se haya escogido, normalmente PDF.

EA3examen requiere una instalación razonablemente completa de `LATEX` incluyendo el paquete `examdesign` y los siguientes ficheros, con expresión de los lugares donde deben instalarse:

Fichero	Ubicación	En Linux, típicamente:
EA3examen.sty	Formatos <code>L^AT_EX</code> ,	<code>/usr/local/share/texmf/tex/latex</code>
EA3examen.el	Ficheros <code>ELisp</code> ,	<code>/usr/local/share/emacs/site-lisp</code>
EA3plantilla.Rnw	Ficheros <code>ELisp</code> ,	<code>/usr/local/share/emacs/site-lisp</code>

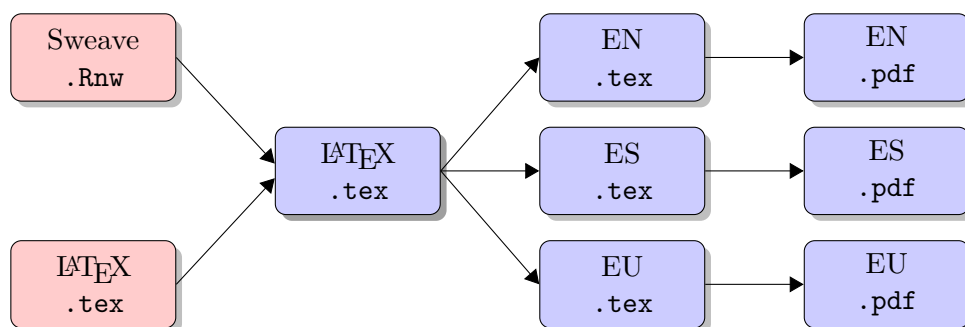
La versión más actual de los ficheros que componen **EA3examen** puede obtenerse de GitHub², y clonarse en cualquier máquina equipada con Git³ mediante el mandato:

```
git clone git@github.com:FernandoTusell/EA3examen.git
```

²Ver <https://github.com/FernandoTusell/EA3examen>. Código abierto, licencia GPL.

³Ver <https://git-scm.com/>.

Figura 1: Proceso de generación de documentos con EA3examen. El usuario genera uno de los dos ficheros en color salmón, mientras que los ficheros en color lavanda se generan automáticamente.



3. Macrodefiniciones

3.1. Locuciones frecuentes

Son expresiones en modo texto que se insertan a veces en una fórmula o entre fórmulas, y constituyen los únicos elementos que requerirían traducción. El uso de estos macros permite utilizar una única expresión para todos los idiomas. Son frecuentes “sí”, “no”, “cierto”, “falso” y la opción `\af` en las preguntas de elección múltiple, para proporcionar una respuesta correcta cuando ninguna más lo es.

3.2. Notación matemática

Constantes. Constantes de uso frecuente como parte de expresiones.

Estimadores. El macro `\est` no está destinado a ser invocado directamente, aunque puede serlo. Sirve como base a `\emv` y similares, que tienen variantes idiomáticas.

La macro `\est` y sus derivados están previstos para emplearse sobre expresiones simples, sobre las que pueda ponerse fácilmente un acento circunflejo. En ocasiones habrá que recurrir a dejar parte de la expresión que se estima fuera del argumento. Véase por ejemplo la diferencia entre estas dos formas de escribir el estimador máximo verosímil de σ^2 :

El estimador `\emv{\sigma^2}...` \Rightarrow El estimador $\hat{\sigma}_{MV}^2 \dots$
 El estimador `\emv{\sigma}^2...` \Rightarrow El estimador $\hat{\sigma}_{MV}^2 \dots$

Cuadro 1: Locuciones frecuentes

Macro	Variantes idiomáticas		
	Español	Euskera	Inglés
<code>\af</code>	Las demás respuestas son falsas.	Beste erantzunak faltsuak dira.	All other answers are false
<code>\casicero</code>	Prácticamente cero	Praktikoki zero	Zero for all practical purposes
<code>\cierto</code>	Cierto	Egiazkoa	True
<code>\falso</code>	Falso	Faltsua	False
<code>\nunca</code>	Nunca	Inoiz ez	Never
<code>\ii</code>	No hay información suficiente para responder	Ez daukagu informazio nahikorik erantzuteko	There is not enough information to answer
<code>\no</code>	No	Ez	No
<code>\Sesgo</code>	Sesgo	Alborapena	Bias
<code>\si</code>	Sí	Bai	Yes
<code>\sif</code>	si	bada	if
<code>\Sif</code>	Si	Bada	If
<code>\Sifthen{a}{b}</code>	Si a entonces b	Bada a orduan b	If a then b
<code>\Var</code>	Var	Bar	Var
<code>\Y</code>	y	eta	and

Cuadro 2: Constantes frecuentes

Macro	Variantes idiomáticas		
	Español	Euskera	Inglés
<code>\oh</code>	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
<code>\rdpi</code>	$\sqrt{2\pi}$	$\sqrt{2\pi}$	$\sqrt{2\pi}$

Funciones y expresiones.

Como argumento de `ecm` se pueden insertar cualesquiera estimadores de los definidos en la subsección 3.2. Por ejemplo, podemos escribir:

El `\ecm{\emv{\theta}}` es...

El $\text{ECM}(\hat{\theta}_{MV})$ es...

`\ecm{\emv{\pi}} = \ecm{\emom{\pi}}`

$\text{ECM}(\hat{\pi}_{MV}) = \text{ECM}(\hat{\pi}_M)$

`\ecm{\hat{\pi}} = \Sesgo(\hat{\pi}) + \Var(\hat{\pi})`

$\text{ECM}(\hat{\pi}) = \text{Sesgo}(\hat{\pi}) + \text{Var}(\hat{\pi})$

Cuadro 3: Estimadores frecuentes

Macro	Variantes idiomáticas		
	Español	Euskera	Inglés
<code>\est{a}{b}</code>	\hat{a}_b	\hat{a}_b	\hat{a}_b
<code>\env{\theta}</code>	$\hat{\theta}_{MV}$	$\hat{\theta}_{EH}$	$\hat{\theta}_{MLE}$
<code>\emom{\theta}</code>	$\hat{\theta}_M$	$\hat{\theta}_M$	$\hat{\theta}_M$

Límites, convergencias, distribuciones asintóticas Un macro general, `\anylim{texto}` sitúa “texto” sobre una flecha hacia la derecha. No está destinado a ser invocado directamente, aunque puede serlo. Hay especializaciones para límites en distribución, probabilidad y casi seguros (o “con probabilidad unitaria”) que tienen variantes idiomáticas..

El macro `\dist{texto}` sitúa “texto” sobre el símbolo habitualmente utilizado para “se distribuye como”. Especializaciones son `\asin` ($= \text{\dist{a}}$) para indicar distribución asintótica y `\diid` ($= \text{\dist{\i id}}$) para indicar “independientes y con idéntica distribución”.

Cuadro 4: Funciones y expresiones frecuentes

Macro	Variantes idiomáticas		
	Español	Euskera	Inglés
<code>\f{x}</code>	$f_X(x)$	$f_X(x)$	$f_X(x)$
<code>\F{x}</code>	$F_X(x)$	$F_X(x)$	$F_X(x)$
<code>\FisherInf{\theta}</code>	$E\left(\frac{\partial \ln f(X,\theta)}{\partial \theta}\right)^2$	$E\left(\frac{\partial \ln f(X,\theta)}{\partial \theta}\right)^2$	$E\left(\frac{\partial \ln f(X,\theta)}{\partial \theta}\right)^2$
<code>\charf{x}</code>	$\psi_X(u)$	$\psi_X(u)$	$\psi_X(u)$
<code>\momg{x}</code>	$\varphi_X(u)$	$\varphi_X(u)$	$\varphi_X(u)$
<code>\ecm{\hat{\theta}}</code>	ECM($\hat{\theta}$)	BEK($\hat{\theta}$)	MSE($\hat{\theta}$)
<code>\puntuacion{1}{5}</code>	(1 puntos, 5 minutos)	(1 puntu, 5 minutu)	(1 points, 5 minutes)

Cuadro 5: Límites, convergencias y distribuciones asintóticas

Macro	Variantes idiomáticas		
	Español	Euskera	Inglés
<code>\anylim{texto}</code>	$\xrightarrow{\text{texto}}$	$\xrightarrow{\text{texto}}$	$\xrightarrow{\text{texto}}$
<code>\dlim</code>	\xrightarrow{d}	\xrightarrow{b}	\xrightarrow{d}
<code>\plim</code>	\xrightarrow{p}	\xrightarrow{p}	\xrightarrow{p}
<code>\cslim</code>	$\xrightarrow{c.s.}$	\xrightarrow{u}	$\xrightarrow{a.s.}$
<code>\dist{xxx}</code>	$\underset{\sim}{xxx}$	$\underset{\sim}{xxx}$	$\underset{\sim}{xxx}$
<code>\asint</code>	$\underset{\sim}{a}$	$\underset{\sim}{a}$	$\underset{\sim}{a}$
<code>\diid</code>	$\underset{\sim}{iid}$	$\underset{\sim}{ibb}$	$\underset{\sim}{iid}$

4. Exámenes para Moodle (= e-Gela)

4.1. Ex-novo

Si se puede partir desde cero, la forma más simple que he encontrado consiste en emplear un formato distinto a `examdesign` junto con el paquete `moodle`⁴. El uso es muy simple. Para preguntas de elección múltiple, emplearíamos algo como:

```
\documentclass[12pt]{article}
\usepackage{moodle}
\begin{document}
\begin{quiz}{My first quiz}
  \begin{multi}[points=3]{A first derivative}
    What is the first derivative of  $x^3$ ?
    \item  $\frac{1}{4} x^4 + C$ 
    \item*  $3x^2$ 
    \item  $51$ 
  \end{multi}
\end{quiz}
\end{document}
```

Al procesar un fichero como el precedente (que guardaremos con extensión `.tex`) mediante

```
pdflatex examen.tex
```

obtenemos un fichero en `examen.pdf` con las soluciones marcadas (que puede servirnos para verificar que todo está bien) y, además, un fichero de nombre `examen-moodle.xml` en XML con la apariencia siguiente:

⁴Ver <http://tug.ctan.org/macros/latex/contrib/moodle/moodle.pdf>. Paquete disponible en CTAN.

```

<?xml version="1.0" encoding="UTF-8"?>
<quiz>
<question type="category">
  <category>
    <text>$module$/My first quiz</text>
  </category>
</question>
<question type="multichoice">
  <name>
    <text>A first derivative</text>
  </name>
  <questiontext format="html">
    <text><![CDATA[<p>What is the first
      derivative of  $(x^3)$ ? </p>]]></text>
  </questiontext>
  <defaultgrade>3</defaultgrade>
  <generalfeedback format="html">
    <text/></generalfeedback>
  <penalty>0.1000000</penalty>
  <hidden>0</hidden>
  <single>true</single>
  <shuffleanswers>1</shuffleanswers>
  <answernumbering>abc</answernumbering>
  <answer fraction="0" format="html">
    <text><![CDATA[<p><math>\frac{1}{4}</math>
       $x^4+C$ </p>]]></text>
  </answer>
  <answer fraction="100" format="html">
    <text><![CDATA[<p><math>(3x^2)</math></p>]]></text>
  </answer>
  <answer fraction="0" format="html">
    <text><![CDATA[<p><math>(51)</math></p>]]></text>
  </answer>
</question>
</quiz>

```

Este fichero, que puede contener muchas cuestiones, es el que puede importarse en Moodle empleando la opción de importación en XML.

Podríamos emplear un procedimiento análogo al seguido en las secciones precedentes para generar ficheros en distintos lenguajes; pero no podemos emplear **examdesign** ni los macros en **EA3examen**, porque Moodle no los interpretará.

4.2. A partir de exámenes pre-existentes

Si se desea aprovechar un examen pre-existente en el formato `examdesign` que por bastantes años hemos venido empleando, puede hacerse mediante el procedimiento que se bosqueja a continuación.

Este procedimiento tiene la ventaja de generar *a la vez* formularios de examen sobre papel para administrar un examen convencional y fuente en XML para importar a un banco de preguntas en Moodle. Tiene no obstante muchos inconvenientes:

1. Emplea herramientas de software auxiliares que pueden no existir o ser difíciles de instalar o sustituir en Windows⁵.
2. No se pueden emplear preguntas relacionadas en bloques, compartiendo un enunciado común. En Moodle cada pregunta es independiente. Puede recurrirse a copiar el enunciado común en todas ellas, pero no se puede garantizar que se presenten al estudiante en orden.
3. Si se emplean macros en las cuestiones para recoger variaciones lingüísticas, hay que reemplazarlos antes de dar el paso de L^AT_EX a XML.
4. Es preciso emplear codificación de caracteres UTF-8; Moodle no admitirá otra cosa.
5. No podremos emplear la ventaja de incluir figuras generadas “al vuelo” mediante la inclusión de código R en el fichero `.Rnw`, ni tampoco código. En particular, el símbolo `<` no será utilizable en una expresión como

`a <- 4`

porque en XML los caracteres `<` y `>` tienen uso especial.

6. Normalmente será preciso retocar el resultado manualmente. Es un proceso frágil.

⁵Aunque son standard en Debian Linux, por ejemplo.

Si, pese a todo, se desea seguir esta vía, el modo de hacerlo es el siguiente:

1. Generar, siguiendo el procedimiento descrito en la Sección 2, versiones `.tex` en todos los idiomas requeridos. En lo que sigue, se describe el modo de actuar con el fichero `EN.examen.tex`; debe hacerse lo propio con cada uno de los otros idiomas.

Este examen no debe contener bloques de cuestiones o figuras.

2. Procesar el fichero `EN.examen.tex` corriendo sobre él el programa `moodleiza`:

```
moodleiza EN.examen.tex
```

Esto produce en el directorio de trabajo varios nuevos ficheros, de entre los cuales `EN.examen-clean.xml` es el que nos interesa. Está en formato XML y permite, si carece de errores, su importación directa a Moodle⁶.

3. Verificar la existencia de errores antes de intentar la importación a Moodle. Un modo de hacerlo es invocar Emacs sobre el fichero `EN.examen-clean.xml` y en el menú XML pinchar **Next error** para ver los errores que puedan existir. En la medida de lo posible, estos errores se deben solucionar “aguas arriba”, en el fichero `.Rnw` si es posible, de manera que queden solucionados para todas las versiones que se generen a partir del mismo.

El programa `moodleiza` invoca otros varios, que deben estar instalados y hace implícitamente supuestos sobre la fuente. En particular, las respuestas deben estar en el modo en que las inserta Emacs cuando pulsamos F4:

```
\choice{
  Texto respuesta
}
```

de forma que `moodleiza` pueda detectar inambiguamente el comienzo y el final. Causaría un error algo como:

```
\choice{
  Texto respuesta
} \choice{
  Otra respuesta
}
```

a pesar de que ello no originaría problemas al producir el examen en L^AT_EX.

⁶El modo de realizarla sed detalla en https://docs.moodle.org/38/en/Moodle_XML_format.

A. Descripción de operaciones que realiza moodleiza.

(Esta Sección debiera ser innecesaria, pero saber las transformaciones que realiza `moodleiza` puede ayudar a diagnosticar problemas cuando algo va mal, o incluso realizar tales transformaciones manualmente de una en una, para ver dónde se produce el fallo.)

Las operaciones que es preciso realizar sobre el fichero `EN.examen.tex` son las siguientes:

1. Expandir todos los macros que hayamos empleado para introducir variantes lingüísticas, de forma que una línea tal como:

```
$X_n\plim a$
```

quede reemplazada (en la versión inglesa) por

```
$X_n {\buildrel p \over\longrightarrow} a$
```

Esto es necesario porque `\plim` (y otros muchos) son macros privados nuestros, que Moodle no reconocerá.

Para obtener esta expansión de macros necesitamos disponer de sus definiciones (las llamaremos `EN.macros-private.sty`; ha de existir un tal fichero por idioma, y necesariamente el nombre ha de terminar en `-private.sty`)⁷. Se inserta en la cabecera de `EN.examen.tex` una línea:

```
\usepackage{EN.macros-private}
```

y se elimina `\usepackage{EA3examen}` (pues las definiciones alternativas de los mismos macros, mutuamente se machacan). Esta sustitución la hace `moodleiza` invocando el editor de flujo `sed`⁸.

Para ficheros fuente en otros idiomas se reemplaza la línea de macros que corresponda (como `ES.macros-private.sty` o `EU.macros-private.sty`). La detección del lenguaje del fichero se efectúa examinando la configuración del paquete `babel` (en particular, la línea `\selectlanguage{<idioma>}`).

A continuación se ejecuta:

```
de-macro EN.examen
```

⁷Este fichero ha de estar creado. No puede obtenerse directamente de `EA3examen.sty`, como podría imaginarse: en `EN.macros-private.sty` y análogos en otros idiomas, los macros han de estar definidos mediante `newcommand` o `renewcommand`; en `EA3examen.sty` han de estar definidos mediante `def`.

⁸Presente en toda instalación Unix/Linux.

Obsérvese que no se incluye la extensión `.tex`. El ejecutable `de-macro` es un programa en Python⁹. El output de `de-macro` es un fichero de nombre `EN.examen-clean.tex`.

2. Convertir el fichero `EN.examen-clean.tex` a codificación UTF-8 si la original era otra (supondremos en lo que sigue que era ISO-8859-1, también conocida como `latin1`).

En Linux, esto puede hacerse mediante el mandato:

```
recode latin1..utf8 EN.examen-clean.tex
```

que **sobreescribe** el fichero original¹⁰.

3. Correr el programa `exam2moodle.pl` sobre el fichero `EN.examen-clean.tex` obtenido en el paso precedente¹¹.

```
exam2mooddle.pl EN.examen.tex
```

Dicho programa cambia el formato de las cuestiones, del propio de `examdesign` al formato XML que Moodle requiere para importación a su banco de cuestiones. Dejará en el directorio en el que estamos trabajando un nuevo fichero `EN.examen-clean.xml`. Este fichero es el que se importa por Moodle, comprobando primero¹² la inexistencia de errores sintácticos, en presencia de los cuales la importación fallaría.

B. Licencias

Todo el software mencionado es propiedad de sus autores, licenciados en diferentes términos. `moodleiza` y `exam2moodle.pl` son de fuente libre bajo licencia GPL, y pueden obtenerse del repositorio GitHub mencionado más arriba.

⁹Por tanto, debe existir un interprete de Python en la máquina en que trabajamos, algo habitual en máquinas Linux; además, ha de ser la versión 2. `de-macro` procede de <https://ctan.org/pkg/de-macro>

¹⁰`recode` es un programa habitual en instalaciones Linux. En Debian/Ubuntu se encuentra en el paquete del mismo nombre.

¹¹`exam2moodle.pl` es un programa muy simple escrito en Perl. Debe existir un intérprete de Perl en la máquina, algo standard en máquinas Linux.

¹²Con Emacs, por ejemplo.