

Introducción a \LaTeX

Preparando documentos de alto nivel

ECEL Research Group

Abril 2021

¿Qué es \LaTeX y por qué aprender a usarlo?

- Es una herramienta para la preparación de documentos
- Contiene facilidades que otros programas no tienen
- *What you see is what you get* vs. *What you see is what you mean*
- Utilizado para la creación de artículos científicos, libros, presentaciones, etc.
- Documentos de alto nivel, con formatos consistentes y complicados

Nuestros objetivos

- 1 Breve historia
- 2 Preliminares
- 3 Empezando un documento
- 4 Manejo de Texto
- 5 Manejo de matemáticas
- 6 Gráficos
- 7 Beamer
- 8 Interfaz Overleaf
- 9 Manejo de Bibliografías
- 10 Integrando con otros programas

Breve historia de \LaTeX

- Creado por Leslie Lamport en 1984, cuando trabajaba en el SRI, como una herramienta para utilizar \TeX y escribir matemáticas avanzadas
- \TeX fue diseñado por Donald Knuth en 1978; un sistema de tipografía de alto nivel
- Para usar \LaTeX , se necesita una distribución \TeX así como un programa que tenga los macros cargados
- La última versión es $\text{\LaTeX} 2_{\epsilon}$, que reemplazó a $\text{\LaTeX} 2.09$



Localizando los caracteres más importantes

- Para invocar comandos, *backslash*: \
- Para señalar que estas ingresando inputs a un comando, llaves: { }
- Para hacer comentarios al script, porcentaje: %
- Para dar argumentos opcionales a comandos, corchetes: []
- Posiblemente es necesario cambiar el idioma o el *layout* del teclado para hacer coincidir los caracteres
- Para la creación y manipulación de tablas: &



Uso de \LaTeX en un computador

- Una distribución \TeX instalada
 - Mik \TeX para Windows
 - Mac \TeX para Mac
- Un editor de \LaTeX para escribir los scripts
 - \TeX maker
 - Sublime Text 3
- Un editor de \LaTeX online: Overleaf
 - No necesita ninguna preinstalación
 - Simple manejo de archivos del documento
 - Un editor con facilidades excelentes
 - Colaboración en tiempo real
 - Requiere internet
 - Tiene límites como software gratuito

- Aquí se especifican configuraciones para todo el documento, macros y paquetes
- Clases de documentos más comunes:
 - `article`
 - `book`
 - `presentation`
 - `report`
 - `beamer`
- Los comentarios se deben escribir comenzando con un signo de porcentaje (%)
- Los comandos se invocan normalmente con un *backslash*, parámetros obligatorios con llaves y parámetros opcionales con corchetes

Comenzando a escribir

- Algunos parámetros default para un documento `article`:
 - Tipo de letra tamaño 10, en libro es 11
 - Orientación vertical
 - 1 columna para el texto
- Para comenzar la segunda parte del archivo `LATEX`:
`\begin{document}`
- Todos los ambientes deben cerrarse con su respectivo `\end{}`
- El texto principal del documento está dentro del ambiente `document`
- Lo mínimo que necesita `LATEX` para construir un documento es la clase de archivo y la ubicación del texto del documento en el script
- Saltos de línea se deben hacer con doble *backslash*: `\\`
- Se deben utilizar comandos para los caracteres especiales

- Adiciones a \LaTeX , nuevos comandos
- Hace más simple algunos procesos que con \LaTeX base son complicados
- Paquetes fundamentales son inputenc con la opción `utf8` y babel con la opción de español (`spanish`)
- Otros paquetes básicos son:
 - geometry; márgenes
 - setspace; interlineado
 - lipsum; *dummy text*

- Es convención utilizar estructura para ordenar el documento
- Es la base para crear tablas de contenido
- Diferentes subniveles para crear secciones
- La tabla de contenidos se genera automáticamente con `\tableofcontents`
- Cuando existen cambios en la numeración, las tablas se actualizan solas
- Es posible editar la forma en la que aparecen los títulos
 - Mediante la configuración de babel se cambia los predeterminados de todo el documento (para el idioma utilizado diferente a inglés)

- Como una alternativa al común `\maketitle`
- Crea una página sin ningún formato (número o pie de página, encabezado si se ha hecho)
- Aquí se puede utilizar los comandos de tamaño de letra y formato para crear una portada
- Los ambientes de alineado especiales son claves para una buena portada
- Mediante `\vspace` se puede espaciar con perfecta exactitud cada título en la portada
- Se puede seguir un formato específico de un trabajo dado por autoridades

- Utilizando símbolos especiales o la creación de un ambiente, se avisa a \LaTeX que estamos escribiendo matemáticas
- Las capacidades tipográficas son muy avanzadas y se puede escribir de lo más simple a expresiones ultra complejas

Ejemplo 1: Función Objetivo de la estimación MCO

El estimador de Mínimos Cuadrados Ordinarios para un modelo de regresión múltiple (para dos regresores) minimiza la expresión

$$\sum_{i=1}^n \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \hat{\beta}_2 x_{i2} \right)^2$$

- Utilizar Word para este tipo de expresiones puede ser rápido e intuitivo para expresiones relativamente simples
- Sin embargo, para expresiones largas, o automatizar muchas expresiones se vuelve poco eficiente
- Alinear las ecuaciones entre ellas es prácticamente imposible, y puede generarse un problema, como movimientos en imágenes

Ejemplo 2: Ecuaciones simultáneas para la minimización MCO

$$\frac{\partial G}{\partial \hat{\beta}_0} = \frac{\partial}{\partial \hat{\beta}_0} \left[\sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i - \hat{\beta}_2 z_i)^2 \right] = \sum_{i=1}^n \left[\frac{\partial}{\partial \hat{\beta}_0} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i - \hat{\beta}_2 z_i)^2 \right] = 0$$

$$\frac{\partial G}{\partial \hat{\beta}_1} = \frac{\partial}{\partial \hat{\beta}_1} \left[\sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i - \hat{\beta}_2 z_i)^2 \right] = \sum_{i=1}^n \left[\frac{\partial}{\partial \hat{\beta}_1} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i - \hat{\beta}_2 z_i)^2 \right] = 0$$

$$\frac{\partial G}{\partial \hat{\beta}_2} = \frac{\partial}{\partial \hat{\beta}_2} \left[\sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i - \hat{\beta}_2 z_i)^2 \right] = \sum_{i=1}^n \left[\frac{\partial}{\partial \hat{\beta}_2} (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i - \hat{\beta}_2 z_i)^2 \right] = 0$$

- El modo matemático, señalado por $\$$ $\$$ es la forma más directa de escribir matemáticas
- Automáticamente pone en un tipo de cursiva al texto, ignora espacios y permite comandos especiales
- Necesitaremos ubicar en nuestro teclado los siguientes caracteres:
 - Signo de dólar: $\$$
 - Acento circunflejo (sombbrero): \wedge
 - Guión bajo: $_$
 - “Et” (Ampersand) : $\&$

- Mediante el uso de ambientes, se puede obviar los signos de dólar
- Se puede numerar y hacer *cross-reference* a las ecuaciones, aplica también a tablas e imágenes
- La notación más avanzada suele tener problemas de visualización, normalmente se arregla con `\displaystyle`
- La alineación se hace mediante el operador “&”, en ambientes que lo permitan
- Matrices funcionan de la misma manera que el ambiente `tabular`
- Se puede utilizar otro software para facilitar la utilización de matemáticas, pero no es infalible
 - Codecogs
 - Mathpix Snip

El paquete graphicx

- Permite la utilización de imágenes en el directorio del script
- Para Overleaf, se debe subir las fotos al proyecto online
- Se maneja el tamaño de la imagen con parámetros opcionales
- El ambiente `figure` permite el posicionamiento de la imagen dentro del documento
- El paquete `float` permite una opción especial de posicionamiento

Los paquetes tikz y pgfplots

- Para hacer gráficos directamente en \LaTeX como alternativa a usar imágenes
- El paquete tikz es la base de la construcción de gráficos, pero usarlo por sí solo puede ser algo complicado
- Utiliza una sintaxis diferente a la de \LaTeX común, usa puntos, paréntesis, comas, etc.
- El paquete pgfplots trabaja a partir de tikz, es más directo
- Se puede graficar funciones, a partir de coordenadas, y leyendo archivos

- Una clase de documento \LaTeX con opciones especiales para hacer presentaciones con diapositivas
- Hay otras formas, pero es la más simple de usar
- Se basa en *frames*, que no necesariamente es igual a diapositivas de PowerPoint
- Un *frame* puede tener diferentes “diapositivas”
- Se maneja en base a temas (estilos) y colores de los mismos
- Tiene desventajas en el uso de animación, se hace en base al avance de diapositivas (no funciona como video)

- Se puede hacer documentos colaborativos, mediante links
- Cuenta con un control de versiones y un chat entre colaboradores
- Se pueden descargar y subir los archivos \TeX para utilizarlos offline
- Se pueden hacer proyectos multi-archivo mediante el paquete subfiles
- Se construye un archivo principal, con un preámbulo que llama a todos los paquetes que se usan en todos los archivos
- Los otros archivos tienen una clase de documento especial, y también compilan por sí solos
- El *subfile* usa el preámbulo del documento principal

El paquete biblatex

- Una alternativa a natbib o bibtex
- Utiliza un archivo externo, que puede ser generado desde una página web de citación o desde programas como Citavi, Mendeley, Zotero
- Con un “índice”, se puede llamar a la fuente al documento y citarla con varios comandos del paquete
 - `\parencite`
 - `\textcite`
- Admite diferentes estilos:
 - APA
 - MLA
 - Chicago
 - Harvard

- Algunos documentos de Word pueden requerir matemáticas avanzadas y no podemos pasarlos a \LaTeX :
 - Tablas que se actualizan con Excel
 - Se necesitan comentarios de alguien que no usa \LaTeX
 - Formatos obligatorios para algunos trabajos
- Se puede utilizar el editor de ecuaciones, pero hemos visto que no es eficiente
- En lugar de ello, podemos escribir código de \LaTeX base, con una opción especial en Word
- No es demasiado avanzado, porque no podemos usar paquetes

- El análisis estadístico avanzado normalmente requiere escribir un reporte en algún procesador de texto
- Puede ser complicado exportar los resultados de ese análisis al documento
- Con Rstudio, se puede simplificar el proceso con el paquete `stargazer`
- Es la opción más simple y rápida; produce resultados de alto nivel para diferentes elementos de R
- Existen herramientas más avanzadas, pero requieren más conocimiento de Rstudio
 - `knitr`
 - `sweave`

- Se puede exportar a Excel las tablas que se generan mediante comandos de Stata
- Se selecciona los resultados deseados, click derecho y copiar tabla
- Eso se pega directamente a Excel, y luego se puede utilizar un *table generator*
- Para las tablas de regresiones, se utiliza el paquete `outreg2`
- Instalar en Stata mediante `ssc install outreg2`
- Realizar la regresión e inmediatamente poner `outreg2 using filename, append tex`
- Se puede exportar como Excel o también como archivo TEX

- *Typos* son el error más común, por ende compilar constantemente es la mejor prevención para esos errores
- Tener proyectos *multi-file* suele limitar las posibles localizaciones de los errores en documentos largos
- La definición incorrecta de ambientes también es un error común, pero Overleaf casi elimina esa posibilidad con su predictor
- Los errores más difíciles de corregir son cuando L^AT_EX entiende algo diferente a lo que pretendíamos (no es exactamente un error)
- Pegar el log de errores en Google o formular/buscar una pregunta T_EX Exchange es la mejor manera de arreglarlos
- Los errores gramaticales/ortográficos pueden pasar desapercibidos, siempre pedir a alguien ser tu *proofreader*

***YOUR
TRAINING
IS NOW
COMPLETE***