Apuntes de Latex

Capitulo 5: Escribiendo matemáticas (continuación)

1 Fórmulas enmarcadas

Para enmarcar fórmulas, se puede utilizar el comando `\boxed{Objeto}`, que es una versión apropiada para modo matemático del comando `\fbox{Objeto}` (que se verá con más detalle más adelante). Por ejemplo:

$$\boxed{\int_{0}^{+\infty}\frac{\operatorname{sen} x}{x}dx = \frac{\pi}{2}}$$

$$\boxed{\int_{0}^{+\infty}\frac{\operatorname{sen} x}{x}dx} = \boxed{\frac{\pi}{2}}$$

se obtiene a partir de:

```
\[ \boxed{\int_{0}^{+\infty}
\frac{\sen x}{x}dx = \frac{\pi}{2}}\]

\[ \boxed{\int_{0}^{+\infty}
\frac{\sen x}{x}dx} = \boxed{\frac{\pi}{2}}\]
```

2 Negación de símbolos

En las tablas de símbolos, puede verse como en numerosos casos están disponibles las versiones de negación de diversos símbolos, por ejemplo:

$\neq$ `\neq` $\nless$ `\nless` $\ngtr$ `\ngtr` etc...

Sin embargo, la negación de *cualquier* símbolo puede obtenerse tambíen poniendo una barra inclinada sobre el mismo. Esto se consigue con el comando:

`\not`

situado justo antes del comando correpondiente al símbolo. Por ejemplo:

`\not\in` $\rightarrow \not\in$ `\not\subset` $\rightarrow \not\subset$

3 Fuentes en modo matemático

Se ha mencionado al comienzo que las letras que son parte de una fórmula aparecen en itálica. Para seleccionar un tipo de letra diferente se tienen los siguientes comandos:

- `\mathbb{Texto}` $\longrightarrow$ $\mathbb{A}\ \mathbb{B}\ \mathbb{C}\ \mathbb{D}\ \mathbb{E}$ (sólo mayúsculas)
- `\mathbf{Texto}` $\longrightarrow$ $\mathbf{A\ B\ C\ D\ E}$
- `\mathcal{Texto}` $\longrightarrow$ $\mathcal{A\ B\ C\ D\ E}$ (sólo mayúsculas)
- `\mathfrak{Texto}` $\longrightarrow$ $\mathfrak{A\ B\ C\ D\ E}$ (mayúsculas y minúsculas)
- `\mathit{Texto}` $\longrightarrow$ $\mathit{A\ B\ C\ D\ E}$
- `\mathnormal{Texto}` $\longrightarrow$ $\mathnormal{A\ B\ C\ D\ E}$ (similar a la itálica)
- `\mathrm{Texto}` $\longrightarrow$ $\mathrm{A\ B\ C\ D\ E}$
- `\mathsf{Texto}` $\longrightarrow$ $\mathsf{A\ B\ C\ D\ E}$
- `\mathtt{Texto}` $\longrightarrow$ $\mathtt{A\ B\ C\ D\ E}$

Podemos obtener otro tipos de letras caligráficas cargando el paquete mathrsfs. Con éste paquete cargado, el nuevo comando `\mathscr{Texto}` produce el siguiente conjunto de letras caligráficas: $\mathscr{A\ B\ C\ D\ E}$ (sólo mayúsculas).

Si en vez de cargar el paquete mathrsfs, cargamos el eucal con la opción `mathscr` (`\usepackage[mathscr]{eucal}`), el comando `\mathscr{Texto}` produce ahora letras tipo «Euler script» en lugar de caligráficas:

$\mathscr{A\ B\ C\ D\ E}$ (también sólo para mayúsculas)

Cuando tratamos con signos matemáticos en vez de con letras, los cambios de tipo de letra anteriores no siempre funcionan adecuadamente. Ésto es especialmente inconveniente cuando se desea poner en negrita una fórmula; por ejemplo, `$\mathbf{a+b=c}$` produce: $\mathbf{a+b=c}$, con los signos $+$ e $=$ no resaltados. Para estas situaciones se dispone del comando:

`\boldsymbol{Objeto}`

que proporciona símbolos en negrita:

`$\boldsymbol{a+b=c}$` $\longrightarrow$ $\boldsymbol{a+b=c}$

(además, puede verse que este comando sirve asimismo para obtener letras en negrita itálica, ya que `$\mathbf{Texto}` pone el texto en negrita romana).

Para ciertos símbolos especiales, el comando `\boldsymbol{Objeto}` puede no funcionar; por ejemplo, compárese:

`$\oint f$` $\longrightarrow$ $\oint f$

`$\boldsymbol{\oint f}$` $\longrightarrow$ $\boldsymbol{\oint f}$

El comando `\boldsymbol` no tiene efecto para la integral cerrada `\oint`. Es éstos casos, se utiliza el comando `\pmb{Objeto}` (poor man's bold), que proporciona a cada símbolo el aspecto de estar en negrita reescribiéndolo con pequeños desplazamientos:

`$\pmb{\oint f}$` $\longrightarrow \pmb{\oint f}$
(con una buena lupa puede verse que la letra 'f' negrita no es idéntica con `\boldsymbol` y con `\pmb`).

4 Puntos suspensivos

Para introducir puntos suspensivos en modo matemático tenemos una amplia colección de comandos. El comando `\dots` produce puntos suspensivos cuya unicación vertical obedece a determinadas reglas, según le sigan signos $+$, $-$, ó $\times$ (puntos centrados), una coma (puntos abajo), etc... Los comandos `\ldots` $\rightarrow \ldots$ y `\cdots` $\rightarrow \cdots$ producen *siempre* puntos abajo y centrados, respectivamente. Además, los comandos `\vdots` $\rightarrow \vdots$ y `\ddots` $\rightarrow \ddots$ producen puntos suspensivos verticales ó diagonales, que son útiles en la escritura de matrices.

Por otra parte, existe otra serie de comandos más especializados:

- `\dotsc` $\longrightarrow$ Para puntos separados por comas
- `\dotsb` $\longrightarrow$ Para puntos separados por operadores binarios ($+$, $-$, etc...)
- `\dotsm` $\longrightarrow$ Para puntos separados por multiplicaciones implícitas
- `\dotsi` $\longrightarrow$ Para puntos separados por signos integrales
- `\dotso` $\longrightarrow$ Otros puntos suspensivos

5 Operadores de tamaño variable: integrales, sumatorios,...

Los signos de integral y sumatorio se obtienen, respectivamente, con los comandos `\int` y `\sum`. El tamaño de éstos signos depende, al igual que ocurría con las fracciones, del modo matemático que se esté utilizando, texto ó resaltado. Por ejemplo:

```
Euler demostró que la serie
$\sum_{n=1}^\infty\frac{1}{n^2}$
converge, pero además que:
  \begin{equation*}
    \sum_{n=1}^\infty\frac{1}{n^2}=\frac{\pi^2}{6}
  \end{equation*}
```

Euler demostró que la serie $\sum_{n=1}^\infty \frac{1}{n^2}$ converge, pero además que:

$$\sum_{n=1}^\infty \frac{1}{n^2} = \frac{\pi^2}{6}$$

Se observa también que la posición de los subíndices del sumatorio cambia si estamos en modo texto (sub/superíndices a un lado) ó en modo resaltado (sub/superíndices debajo y encima). Éste comportamiento es común a casi todos los operadores de tamaño variable ($\sum$, $\prod$, $\coprod$, etc..) y tambíen a funciones (como la expresión para el límite: `\lim`).

En el caso de que queramos, *para el modo texto*, cambiar la posición de los sub/superíndices (para que aparezcan abajo/arriba), podemos utilizar el comando `\limits` inmediatamente a continuación del comando de operador de tamaño variable (`\sum`, `\proc`, etc...). Por contra, si deseamos que, *en modo resaltado*, los sub/superíndices aparezcan a un lado, se debe utilizar el comando `\nolimits`; por ejemplo:

```
Euler demostró que la serie $\sum\limits_{n=1}^\infty\frac{1}{n^2}$
converge, pero además que:
  \begin{equation*}
    \sum\nolimits_{n=1}^\infty\frac{1}{n^2}=\frac{\pi^2}{6}
  \end{equation*}
```

Euler demostró que la serie $\sum\limits_{n=1}^\infty \frac{1}{n^2}$ converge, pero además que:

$$\sum\nolimits_{n=1}^\infty \frac{1}{n^2} = \frac{\pi^2}{6}$$

Todo lo anterior es válido para cualquier signo de tamaño variable, excepto los de tipo integral; para éstos, tanto en modo texto como en resaltado, los límites se colocan a un lado:

```
Así, $\lim\limits_{x\to\infty}\int_0^x\frac{\sin x}{x}\,\mathrm{d}x
=\frac{\pi}{2}$ y por definición,
\begin{equation*}
  \int_0^\infty\frac{\sin x}{x}\,\mathrm{d}x=\frac{\pi}{2}
\end{equation*}
```

Así, $\lim\limits_{x\to\infty}\int_0^x\frac{\sin x}{x}\,\mathrm{d}x=\frac{\pi}{2}$ y por definición,

$$\int_0^\infty\frac{\sin x}{x}\,\mathrm{d}x=\frac{\pi}{2}$$

Si queremos límites abajo/arriba, el comando `\limits` cambia su ubicación.

Finalmente, además de los operadores de tamaño variable predefinidos, existe la posibilidad de asignar a cualquier expresión el comportamiento de un operador de tamaño variable con el comando `\mathop{Expresión}`; por ejemplo, definiendo:

```
\newcommand*{\prueba}{\mathop{\textrm{A}}}
```

(con lo cual asignamos al comando `\prueba` el operador `\mathop{\textrm{A}}`, para abreviar), escribiendo:

```
Esto es el operador $\prueba_{3x}^{4y}$ con subíndices en modo texto

\[ \text{y esto lo mismo en modo resaltado:} \quad \prueba_{3x}^{4y} \]
```

obtenemos:

Esto es el operador A_{3x}^{4y} con subíndices en modo texto

$$\text{y esto lo mismo en modo resaltado} \quad \mathop{\mathrm{A}}_{3x}^{4y}$$

Si deseamos que el nuevo operador cambie su tamaño según el tipo de fórmula, y su situación en la misma, se puede incluir como argumento del comando `\mathop` el comando:

```
\mathchoice{Texto.display}{Texto.text}{Texto.script}{Texto.scriptscript}
```

donde en cada uno de los argumentos definimos cómo se va a representar el operador en modo displaystyle, textstyle, scriptstyle y scriptscriptstyle, respectivamente. Véase el siguiente ejemplo:

```
\newcommand*{\pruebaa}{\mathop{\mathchoice{\textrm{\huge $\Delta$}}%
{\Delta}{\Delta}{\Delta}}}
```

con lo cual de: `$\pruebaa^{\pruebaa^{\pruebaa}}$` resultaría: $\Delta^{\Delta^{\Delta}}$
y de: `\[ \pruebaa_3^4 \]` resultaría:

$$\mathop{\Large\Delta}_{3}^{4}$$

6 Varias líneas de subíndices

En caso de que se quiera colocar varias líneas de subíndices, se puede utilizar el comando `\substack`, como en el siguiente ejemplo:

$$p_k(x) = \prod_{\substack{i=1 \\ i\neq k}}^{n} \left(\frac{x - t_i}{t_k - t_i} \right)$$

se obtiene con:

```
\begin{equation*}
  p_k(x)=\prod_{\substack{i=1\\i\ne k}}^n
    \left(\frac{x-t_i}{t_k-t_i}\right)
\end{equation*}
```

Se observa que todas las líneas de subíndices aparecen centradas. Si se quiere justificarlas a la izquierda, se debe utilizar el entorno:

```
\begin{subarray}{l}
Líneas
\end{subrray}
```

como por ejemplo:

```
\[ \sum_{\begin{subarray}{l} 1\leq i\leq 100\\
      i<j<8\end{subarray}} P(i,j) \]
```

$$\boxed{\sum_{\begin{subarray}{l} 1\leq i\leq 100\\ i<j<8\end{subarray}} P(i,j)}$$

7 Integrales múltiples

Para utilizar integrales múltiples, el reiterar signos de integración con `\int` resulta inadecuado, ya que el espaciado entre signos no es correcto. La forma correcta es utilizar los comandos `\iint` (int. dobles), `\iiint` (int. triples) e `\iiiint` (int. cuádruples). El comando `\idotsint` produce la abreviatura con puntos suspensivos apropiada para integrales múltiples n-dimensionales:

```
\[ \iint f(x,y) dxdy \qquad \iiint f(x,y,z) dxdydz \qquad
\idotsint_M dx_1\dots dx_n \]
```

$$\boxed{\iint f(x,y) dxdy \qquad \iiint f(x,y,z) dxdydz \qquad \idotsint_M dx_1\dots dx_n}$$

Finalmente, `\oint` produce el símbolo de integral cerrada $\oint$ (véanse las tablas en "The comprehensive LaTeX symbol list" para versiones más complejas del símbolo integral).

8 Nombres de funciones

La forma correcta de escribir una función genérica, como por ejemplo, $f(x)$, es en itálica, la forma estándar en modo matemático. Sin embargo, existen funciones especiales como cos, lim, log, etc..., que poseen un nombre específico para designarlas. Estas funciones se escriben habitualmente en fuente de tipo “roman”. Existen las siguientes funciones disponibles, que se obtienen a través del comando `\NombreFunción`

`\arccos`	`\cos`	`\csc`	`\exp`	`\ker`	`\limsup`	`\min`	`\sinh`
`\arcsin`	`\cosh`	`\deg`	`\gcd`	`\lg`	`\ln`	`\Pr`	`\sup`
`\arctan`	`\cot`	`\det`	`\hom`	`\lim`	`\log`	`\sec`	`\tan`
`\arg`	`\coth`	`\dim`	`\inf`	`\liminf`	`\max`	`\sin`	`\tanh`

Además, si tenemos el paquete babel con la opción spanish cargado, se pueden utilizar los nombres castellanizados de algunas funciones:

sen	`\sen`	arcsen	`\arcsen`	tg	`\tg`	arctg	`\arctg`

Algunos de éstos comandos (`\lim`, `\det`, ...) funcionan de forma similar a los símbolos $\sum$, $\prod$, etc..., y al igual que éstos, puede cambiarse su comportamiento con el comando `\limits`:

```
Esto es un ejemplo de fórmula tipo texto $\lim_{x \rightarrow \infty} 3x+1$
\[ \text{y esto de resaltada:} \quad \lim_{x \rightarrow \infty} 3x+1  \]
cambiando con el comando limits: \ $\lim\limits_{x \rightarrow \infty} 3x+1$
```

Esto es un ejemplo de fórmula tipo texto $\lim_{x \rightarrow \infty} 3x+1$

$$\text{y esto de resaltada:} \quad \lim_{x \rightarrow \infty} 3x+1$$

cambiando con el comando limits tenemos: $\lim\limits_{x \rightarrow \infty} 3x+1$ (en modo texto)

Con los comandos:

```
\DeclareMathOperator{\NombreComando}{SímboloFunción}
\DeclareMathOperator*{\NombreComando}{SímboloFunción}
```

podemos definir nuevas funciones, donde `\NombreComando` denota el nombre del nuevo comando, y `SímboloFunción` cómo será escrita la nueva función (por defecto, en tipo de letra romana). La versión con asterisco se utiliza cuando se quiere que los límites del operador se comporten como en los comandos `\sum` ó `\lim`.

9 El comando \{smash}

El comando `\smash[Argumento]{Objeto}` anula la altura ó profundidad (si `Argumento` toma el valor t ó b, respectivamente) de `Objeto`, lo cual, por ejemplo, puede ser útil para hacer más consistente el aspecto de radicales adyacentes de distinto tamaño:

```
\[ x:=(1/\sqrt{x_{i_j}})\sqrt{1+y} \qquad
x:=(1/\sqrt{\smash[b]{x_{i_j}}})\sqrt{1+y} \]
```

$$x:=(1/\sqrt{x_{i_j}})\sqrt{1+y} \qquad x:=(1/\sqrt{\smash[b]{x_{i_j}}})\sqrt{1+y}$$

10 Diagramas conmutativos con el paquete amscd

El paquete `amscd` proporciona el entorno `CD` para escribir diagramas conmutativos rectangulares simples (sin flechas en diagonal). Para diagramas más complejos, se pueden utilizar paquetes más potentes (diagram, xypic, kuvio ó pstricks). Debe tenerse en cuenta que el entorno CD sólo funciona *en modo matemático*. Además, parece tener problemas de incompatibilidad con la opción `spanish` de `babel` (el libro de Latex da una solución, pero parece no funcionar, estamos trabajando en averiguar que es lo que falla). Su sintaxis es:

```
\begin{CD}
Linea superior \\
Linea central \\
Linea inferior
\end{CD}
```

En las líneas superior e inferior deben escribirse los nodos del diagrama, así como los comandos siguientes, que se utilizan para trazar flechas **horizontales**:

```
@>>>      @>Encima>Debajo>
@<<<      @<Encima<Debajo<
@=
```

donde los de las dos primeras filas escriben flechas orientadas, y el último un símbolo de igualdad horizontal. Las versiones a la izquierda representan flechas simples, y las de la derecha pueden usarse para añadir objetos encima/debajo de las flechas. En la línea central se deben utilizar los comandos para escribir flechas `verticales`:

```
@AAA      @AEncimaADebajoA
@VVV      @VEncimaVDebajoV
@|
```

que poseen un significado análogo; la "A" denota flecha hacia arriba, la "V" flecha hacia abajo, y el símbolo "|" igualdad vertical. Véase el siguiente ejemplo:

$$\begin{CD} A @>j>> B @>>c> C \\ @VVV @AAhA @| \\ D @= E @<<p< F \end{CD}$$

11 Definiendo nuevos comandos

Una característica general de LaTeX que lo hace muy interesante, es la capacidad de definir nuevos comandos. Por ejemplo, imaginemos que la expresión $(x_1, x_2, \dots, x_n)$ aparece frecuentemente en nuestro documento. Podemos entonces definir:

```
\newcommand{\vect}{(x_1,x_2,\dots,x_n)}
```

con lo cual, cada vez que escribamos `$\vect$` (el nombre del nuevo comando) se imprimirá $(x_1, x_2, \dots, x_n)$. Todos los nuevos comandos conviene situarlos *en el preámbulo.*

La redefinición de comandos admite *argumentos variables.* Si por ejemplo escribimos:

```
\newcommand{\vect}[1]{(#1_1,#1_2,\dots,#1_n)}
```

(añadiendo un argumento, que se sustituye en la fórmula con `"#1"`), escribiendo `$\vect{x}$` obtendríamos $(x_1, x_2, \dots, x_n)$, con `$\vect{a}$` se tendría $(a_1, a_2, \dots, a_n)$, etc...

Añadiendo más argumentos, podemos obtener construcciones más complejas, por ejemplo, definiendo:

```
\newcommand{\vect}[2]{(#1_1,#1_2,\dots,#1_#2)}
```

`$\vect{x}{n}$` daría como resultado $(x_1, x_2, \dots, x_n)$ mientras que con `$\vect{a}{p}$` se obtendría $(a_1, a_2, \dots, a_p)$.

También existe la posibilidad de definir comandos con argumentos optativos, que toman un determinado valor por defecto. Por ejemplo, construyamos:

```
\newcommand{\nuevovector}[2][x]{(#1_1,#1_2,\dots,#1_#2)}
```

donde la "x" entre paréntesis es el valor por defecto del argumento opcional (siempre el número 1). Entonces, escribiendo: `$\nuevovector{n}$` ó `$\nuevovector{p}$` obtendríamos $(x_1, x_2, \dots, x_n)$ y $(x_1, x_2, \dots, x_p)$ respectivamente, mientras que añadiendo un argumento optativo cambiaríamos su valor por defecto de "x":
`$\nuevovector[a]{n}$` $\longrightarrow (a_1, a_2, \dots, a_n)$.

12 Ejercicios

1.- Escribir un comando dependiente de dos argumentos que calcule la derivada parcial respecto de una función f respecto a una variable x, de forma que se puedan elegir tanto f como x:

$$\frac{\partial f}{\partial x}$$

2.-

$$\left(\prod_{j=1}^{n} \hat{x}_j\right) H_c = \frac{1}{2} \hat{k}_{ij} \det \widehat{\mathbf{K}}(i|i), \qquad i = 1, \dots, n.$$

3.-

$$\det \mathbf{B} = \sum_{l=0}^{n} \sum_{I_l \subseteq n} \prod_{i \in I_l} (b_{ii} - \lambda_i) \det \mathbf{B}^{(\lambda)}(I_l | I_l),$$

4.-

$$\mathbf{K}(t, t_1, \dots, t_n) = \begin{pmatrix} D_1 t & -a_{12} t_2 & \dots & -a_{1n} t_n \\ -a_{21} t_1 & D_2 t & \dots & -a_{2n} t_n \\ \hdotsfor{4} \\ -a_{n1} t_1 & -a_{n2} t_2 & \dots & D_n t \end{pmatrix}$$

5.-

$$\begin{aligned} H_c =& \frac{n_1!\,n_2!\,n_3!}{n_1+n_2+n_3} \sum_i \left[\binom{n_1}{i} \binom{n_2}{n_3-n_1+i} \binom{n_3}{n_3-n_2+i} \right. \\ & \left. + \binom{n_1-1}{i} \binom{n_2-1}{n_3-n_1+i} \binom{n_3-1}{n_3-n_2+i} \right] \end{aligned}$$

6.-

$$\lim_{v \to 0} \frac{H(z+v) - H(z) - BH(z)v}{\|v\|} = 0$$

7.-

$$\Delta_0 \ln \psi_0(r) = \left(\frac{d^2}{dr^2} + \frac{1}{r} \frac{d}{dr} \right) \ln \psi_0(r)$$

8.- Construir con el entorno `align`:

$$\begin{aligned} \gamma_x(t) &= (\cos tu + \operatorname{sen} tx, v), && (1) \\ \gamma_y(t) &= (u, \cos tv + \operatorname{sen} ty), && (2) \\ \gamma_z(t) &= \left(\cos tu + \frac{\alpha}{\beta} \operatorname{sen} tv, -\frac{\beta}{\alpha} \operatorname{sen} tu + \cos tv \right). && (3) \end{aligned}$$

9.- Construir con el entorno `alignat`:

$$\begin{alignedat}{4} V_i &= v_i - q_i v_j, &\qquad X_i &= x_i - q_i x_j, &\qquad U_i &= u_i, &\qquad &\text{para } i \neq j; \\ V_j &= v_j, & X_j &= x_j, & U_j u_j &+ \sum_{i \neq j} q_i u_i. & & \qquad (4) \end{alignedat}$$

10.- Construir:

$$\begin{aligned} Jv(B) = \int_{B \cap S_v} (v^+ - v^-) \otimes \nu_v \, d\mathcal{H}_{n-1} &= \int_{B \cap S_v} (f(u^+) - f(u^-)) \otimes \nu_u \, d\mathcal{H}_{n-1} \\ &= \int_{B \cap S_u} (f(u^+) - f(u^-)) \otimes \nu_u \, d\mathcal{H}_{n-1} \end{aligned} \qquad (5)$$