



Cuarto Taller de L^AT_EX

Taller Básico de TikZ

Manuel Merino

Creador de Aprendiendo L^AT_EX

Resumen

En este taller, el participante obtendrá las habilidades básicas para hacer uso del paquete `tikz` donde haremos uso de múltiples entornos que nos permitirán realizar diferentes tipos de gráficos de manera vectorial. Comenzaremos con el uso del comando `draw` y sus opcionales para que así el estudiante tenga múltiples herramientas de dibujo dependiendo de sus necesidades. Se realizará también gráficos de funciones en 2D con diferentes configuraciones dependiendo su finalidad, algunas gráficos estadísticos y una breve introducción a imágenes en 3D.

1. El paquete TikZ

Es importante dejar claros algunos puntos y sobre todo recordar los aspectos más básicos para un mayor dominio de este. Es esencial identificar y saber que Tikz es un lenguaje de programación de dibujo vectorial (Zavala, 2020).

1.1. Introducción al comando `draw`

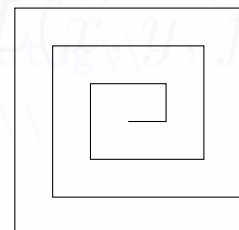
En esta sesión vamos a comenzar con los el comando `\draw` que nos ofrece TikZ, pero en lugar de una larga sección de instalación y configuración, esta parte comienza con recordatorios de algunos puntos vistos en el programa anterior. Se explicará todas las funciones básicas y algunas de las más avanzadas de Tikz, sin entrar en todos los detalles. Esta parte también contiene algunas pautas sobre cómo debe proceder al crear gráficos con TikZ (Tantau, 2020).

Recordando...

El uso del comando `\draw` nos permite crear segmentos mediante un sistema de coordenadas, veamos:

Código L^AT_EX:

```
\tikz \draw (0,0) -- (0,3) -- (3, 3) -- (3,0.5)-- (0.5, 0.5)
-- (0.5, 2.5) -- (2.5, 2.5) -- (2.5, 2) -- (2.5,1) -- (1, 1)
-- (1, 2) -- (2, 2) -- (2, 1.5) -- (1.5, 1.5);
```

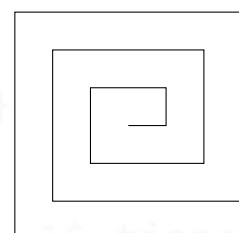


Comando `tikz`

Observamos el comportamiento del comando `draw` uniendo coordenadas mediante segmentos hasta lograr un lugar geometrico.

Código L^AT_EX:

```
\begin{tikzpicture}
\draw (0,0) -- (0,3) -- (3, 3) -- (3,0.5)-- (0.5, 0.5) --
(0.5, 2.5) -- (2.5, 2.5) -- (2.5, 2) -- (2.5,1) -- (1, 1)
-- (1, 2) -- (2, 2) -- (2, 1.5) -- (1.5, 1.5);
\end{tikzpicture}
```



Entorno `tikzpicture`

1.2. Opcionales de draw y más

Como mencionamos en el párrafo anterior, ya se vieron algunos de los opcionales que nos ofrece el comando `draw`, lo que daremos de manera resumida algunos aspectos.

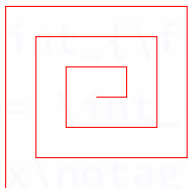
Opcional de color

El opcional de color solo se usa colocando el nombre del color que Tikz nos proporciona, pero sin olvidar que el usuario mismo podría crear su propio color.

Código L^AT_EX:

```
\begin{tikzpicture}
  \draw[red] (0,0) -- (0,3) -- (3, 3) -- (3,0.5) -- (0.5, 0.5) -- (0.5, 2.5) --
  (2.5, 2.5) -- (2.5, 2) -- (2.5,1) -- (1, 1) -- (1, 2) -- (2, 2) -- (2, 1.5) --
  (1.5, 1.5);
\end{tikzpicture}
```

Salida L^AT_EX:



Gracias al paquete `xcolor` tenemos el acceso a los siguientes colores [Kern \(2011\)](#):

red:		magenta:		gray:		pink:		purple:	
green:		yellow:		darkgray:		violet:		white:	
blue:		orange:		lime:		teal:		black:	
cyan:		lightgray:		brown:		olive:			

Una recomendación ya brindada, es acceder a la web [L^AT_EX Color Definitions](#), tomar el color más conveniente y que le agrade a usted, para finalmente copiar la definición del color mediante la escala `rgb` y escribirla en el preámbulo del documento. Se invita al lector a poner en práctica este ejercicio.

Opcional de grosor

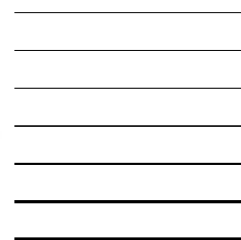
En este caso, los opcionales de grosor vistos son los siguientes:

- ultra thin
- thin
- thick
- ultra thick
- very thin
- semithick
- very thick

el cual se muestran desde el más fino hasta el más grueso. Veamos los ejemplos TikZ

Código L^AT_EX:

```
\draw[ultra thin] (2,1.5) -- (5,1.5);
\draw[very thin] (2,1) -- (5,1);
\draw[thin] (2,0.5) -- (5,0.5);
\draw[semithick] (2,0) -- (5,0);
\draw[thick] (2,-0.5) -- (5,-0.5);
\draw[very thick] (2,-1) -- (5,-1);
\draw[ultra thick] (2,-1.5) -- (5,-1.5);
```



Grosos

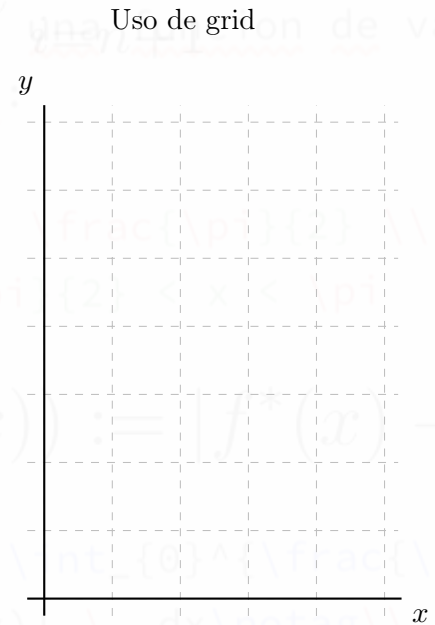
Opcional de tamaño de paso

Si bien es la primera vez que usaremos este opcional, no es algo muy complicado de entender. Cuando el usuario necesita realizar una malla de un sector rectangular, es necesario un tamaño de paso de esta grilla, y de esta manera es como funciona este opcional. Es necesario escribir `grid` en lugar de `rectangle`. Veamos un ejemplo.

Deseamos realizar un sistema de coordenadas, donde en el eje X tenemos los valores desde 0 hasta 5 y en el eje Y desde 0 hasta 7 con una grilla de color `lightgray`.

Código L^AT_EX:

```
\draw[lightgray, dashed] (-0.25, -0.255) grid
    (5.2,7.25);
\draw[thick] (5.25, 0) node[below right] {$x$} --
    (-0.25, 0);
\draw[thick] (0, 7.25) node[above left] {$y$} --
    (0, -0.25);
```

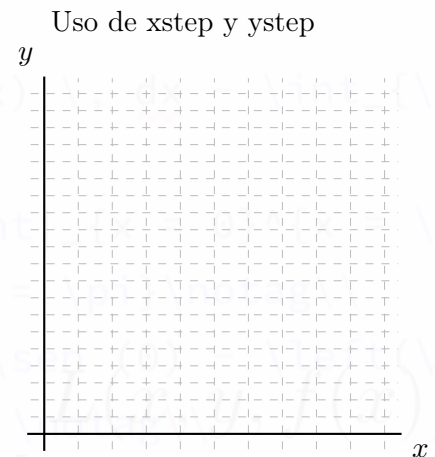


Esta forma de realizar una malla o grillado es más óptima que al hacer uso de alguna estructura de control repetitiva, como `foreach`. Lo que el usuario podría desear, es cambiar el tamaño de paso de esta grilla, tanto de manera proporcional, o de forma personalizada en los ejes x y y con los opcionales `step`, `xstep` y `ystep` respectivamente. Veamos un ejemplo.

Deseamos realizar un sistema de coordenadas, donde en el eje X tenemos los valores desde 0 hasta 5, con un tamaño de paso de 0.5 y en el eje Y desde 0 hasta 5 con un tamaño de paso de 0.25 y finalmente color `lightgray`.

Código L^AT_EX:

```
\draw[lightgray, dashed, xstep = 0.5, ystep = 0.25] (-0.25, -0.255) grid (5.2,5.25);
\draw[thick] (5.25, 0) node[below right] {$x$} -- (-0.25, 0);
\draw[thick] (0, 5.25) node[above left] {$y$} -- (0, -0.25);
```



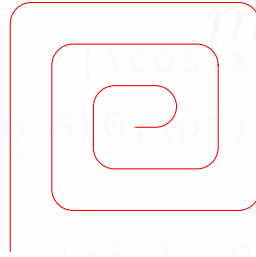
Se deja al lector a que ponga en práctica el uso de este opcional.

Opcional de esquinas redondeadas

En muchas ocasiones necesitamos dar un estilo distinto a las esquinas de los segmentos unidos por el comando `draw`, en este caso por ejemplo, con esquinas redondeadas. El opcional que nos ayudará a realizar este estilo es `rounded corners` (Tantau, 2020). Veamos un ejemplo para un mejor entendimiento

Código L^AT_EX:

```
\draw[red, rounded corners = 8pt] (0,0) -- (0,3) -- (3, 3) -- (3,0.5)-- (0.5, 0.5) --
(0.5, 2.5) -- (2.5, 2.5) -- (2.5, 2) -- (2.5,1) -- (1, 1) -- (1, 2) -- (2, 2) --
(2, 1.5) -- (1.5, 1.5);
```

Salida L^AT_EX:

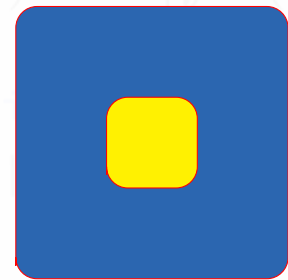
Uso de rounded corners

Opcional de relleno

El opcional de relleno es `fill` y nos permitirá llenar de un color la forma determinada con las coordenadas definidas. Es importante el uso del comando `cycle`.

Código L^AT_EX:

```
\draw[red, rounded corners = 8pt, fill = blue] (0, 0) --
(3, 0) -- (3, 3) -- (0, 3) -- (0, 0) -- cycle;
\draw[red, rounded corners = 8pt, fill = yellow] (1, 1)
-- (2, 1) -- (2, 2) -- (1, 2) -- (1, 1) -- cycle;
```



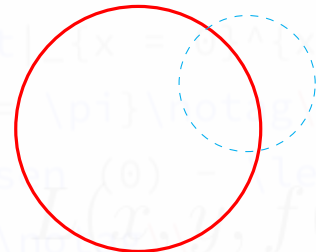
Uso de fill

Circunferencias

Matemáticamente sabemos que para definir una circunferencia en el plano cartesiano, debemos de tener el centro y el radio de ella, de esta manera, podemos representarla en el entorno `tikzpicture` con el comando `circle[radius = r]` luego de ubicar el centro con `\draw (h, k)`. Veamos un ejempllo.

Código L^AT_EX:

```
\draw[red, very thick] (0,0) circle[radius = 1.35];
\draw[cyan, dashed] (1.2,0.5) circle[radius = 0.75];
```



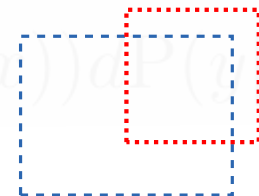
Uso de circle

Rectángulos

En esta ocasión, para definir un rectángulo con TikZ, necesitamos de dos de sus vértices, el cual, deben ser opuestos en su diagonal. Necesitaremos el comando `rectangle` entre los vertices. Veamos un ejemplo sencillo para un mejor entendimiento.

Código L^AT_EX:

```
\draw[azul, very thick, dashed] (0,0) rectangle (4, 3);
\draw[red, ultra thick, dotted] (2,1) rectangle
(4.5, 3.5);
```



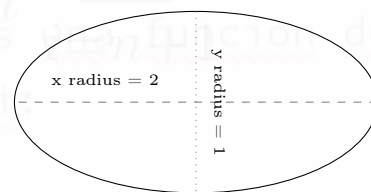
Uso de rectangle

Elipses

Para crear una elipse en TikZ, se necesita comando `ellipse` en el cual, se deberá de definir el centro de la elipse, la distancia del eje mayor y el menor (horizontal y/o vertical). Veremos un ejemplo de una elipse con centro en el vértice (0,0) y con eje mayor paralelo al eje x con una distancia de 2 y el eje menor, paralelo vertical con una distancia de 1.5

Código L^AT_EX:

```
\draw (0,0) ellipse[x radius = 2, y radius = 1];
```



Uso de ellipse

Arco de circunferencia

Para graficar un arco de circunferencia necesitamos el comando `arc` y definir 3 elementos, que son: el centro de la circunferencia, el ángulo en grados sexagesimales inicial, el ángulo en grados sexagesimales final y el radio de circunferencia. Como ejemplo, graficaremos un arco de circunferencia de centro (0,0), con ángulo inicial 0°, ángulo final 270° y radio 1.

Código L^AT_EX:

```
\draw (0,0) arc (0:270:1);
```



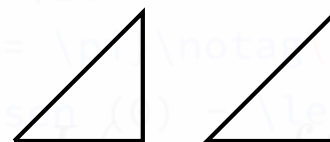
Uso de arc

Uso de ciclos

Cada vez que deseemos crear lugares geométricos cerrados, es decir, donde el vértice de inicio y el vértice de fin que definiremos con el comando `\draw` necesitaremos terminar la línea de código con `--cycle` en lugar de terminarlo con el vértice inicial. Veamos un ejemplo.

Código L^AT_EX:

```
\draw (0,0) -- (1,0) -- (1,1) -- (0,0);
\draw (1.5, 0) -- (2.5, 0) -- (2.5, 1) -- cycle;
```



Uso de cycle

Degradados con shape

Si deseamos graficar alguna forma con degradados de arriba hacia abajo, debemos usar ya no el comando `draw`, sino, el comando `shape` el cual necesitará como opcionales, los colores en donde se recorrerá el degradado vertical. Veamos un ejemplo

Código L^AT_EX:

```
\shade[top color=white, bottom color= azul] (0,0)
rectangle (2,1);
\shade[top color=white, bottom color= red] (2,0)
circle[radius = 0.5];
```



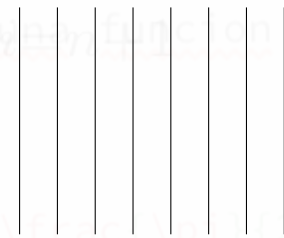
Uso de shape

Estructuras repetitivas foreach

En el mundo de la programación, existen formas para realizar bucles o iteraciones gracias a las estructuras de control repetitivas. En L^AT_EX podemos hacer uso de estos bucles gracias al comando `foreach`. Daremos un ejemplo sencillo para ver su uso.

Código L^AT_EX:

```
\foreach \x in {1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5}{
  \draw (\x, 0) -- (\x, 2);
}
```



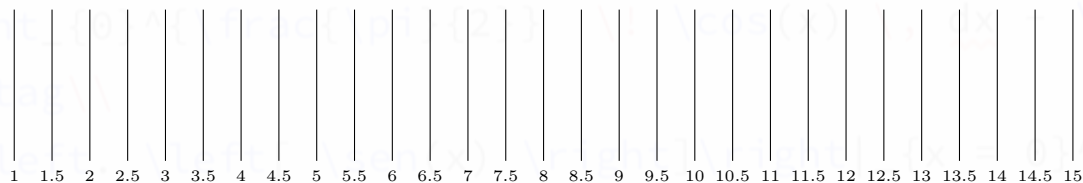
Uso de foreach

Si deseamos recorrer una cantidad grande de valores, no es necesario colocar todos, es decir, solo colocaremos el primer elemento, el segundo (indicando el tamaño de paso a seguir) y el último elemento. Veamos un ejemplo.

Código L^AT_EX:

```
\foreach \x in {1, 1.5, ..., 15}{
  \draw (\x, 0) -- (\x, 2);
  \node[below] at (\x, 0) {\tiny \x};
}
```

Salida L^AT_EX:



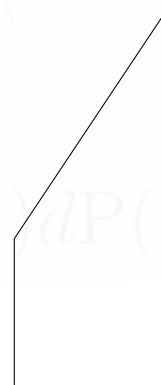
Uso de foreach

Definiendo coordenadas

La definición de coordenadas, muchas veces nos ahorran tiempo de codificación, ayudando al codificador un mayor fluidez en el código para el dibujo requerido. Para definir una coordenada, necesitamos el comando `\coordinate` seguido del nombre que se le dará a la coordenada, como por ejemplo (A), (B), (C), etc. Finalmente se se asigna la coordenada. Veamos una aplicación de la definición de coordenadas.

Código L^AT_EX:

```
\coordinate (A) at (0,0);
\coordinate (B) at (1,0);
\coordinate (C) at (1,2);
\coordinate (D) at (0,1);
\draw (A) -- (B) -- (C) -- (D) -- cycle;
```



Uso de coordinate

1.3. Recortando un camino

En muchas ocasiones el usuario necesita encapsular en diferentes formas nuestros gráficos creados con el entorno `tikzpicture`. Veamos un par de ejemplos:

Código L^AT_EX:

```
\begin{figure}[H]
  \centering
  \begin{minipage}{0.475\linewidth}
    \begin{tikzpicture}[scale = 1]
      \clip[draw] (-1.7, -1.5) rectangle (1.7, 1.5);
      \draw (0,0) ellipse[x radius = 2, y radius = 1];
      \draw[gray, dashed] (2,0) -- (-2,0);
      \draw[gray, dotted] (0, 1) -- (0, -1);
      \node at (-1, 0.25) {\tiny x radius = $2$};
      \node at (0.25, 0) {\rotatebox{-90}{\tiny y radius = $1$}};
    \end{tikzpicture}
    \caption{Uso de \texttt{rectangle}}
  \end{minipage}
  \hfill
  \begin{minipage}{0.475\linewidth}
    \begin{tikzpicture}[scale = 1]
      \clip[draw] (0,0) circle (1.5cm);
      \draw (0,0) ellipse[x radius = 2, y radius = 1];
      \draw[gray, dashed] (2,0) -- (-2,0);
      \draw[gray, dotted] (0, 1) -- (0, -1);
      \node at (-1, 0.25) {\tiny x radius = $2$};
      \node at (0.25, 0) {\rotatebox{-90}{\tiny y radius = $1$}};
    \end{tikzpicture}
    \caption{Uso de \texttt{circle}}
  \end{minipage}
\end{figure}
```

Salida L^AT_EX:

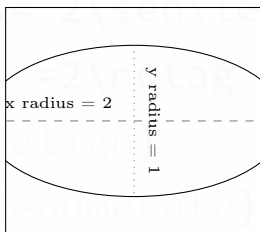


Figura 1: Uso de `rectangle`

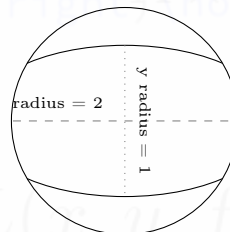


Figura 2: Uso de `circle`

1.4. Gráfica de funciones

Para realizar una gráfica de una función $f : A \subset \mathbb{R} \rightarrow \mathbb{R}$ donde $A = [a, b]$, $a < b$ y $y = f(x)$ tenemos que hacer uso del comando `plot` (`{\x},{funcion}`) donde `\x` es la variable independiente y `funcion` es la regla de correspondencia de la función a graficar. El dominio se debe definir como un opcional del comando `\draw` que sería de la forma `domain = a:b`.

Veamos algunos ejemplos:

Código L^AT_EX:

```
%\definecolor{azul}{rgb}{0.17, 0.40, 0.69}
%\definecolor{background}{gray}{0.90}
\begin{tikzpicture}[scale = 2, samples = 60]
  \draw[background, fill = background] (-pi, -1) rectangle (pi, 1);
  \draw[thick, white, dashed, xstep = 0.25*pi, ystep = 0.5] (-pi, -1) grid (pi, 1);
  \draw[<-, thick] (3.5, 0) node [below right] {\x$} -- (-3.5, 0);
  \draw[<-, thick] (0, 1.5) node [above left] {\y$} -- (0, -1.5);
  \draw[red, thick, domain = -pi:pi] plot({\x},{sin(2*\x r)});
  \draw[azul, thick, domain = -pi:pi] plot({\x},{cos(2*\x r)});
  \node[below left] at (pi/4, 0) {\frac{\pi}{4}};
  \node[below left] at (pi/2, 0) {\frac{\pi}{2}};
  \node[below left] at (3*pi/4, 0) {\frac{3\pi}{4}};
  \node[below left] at (pi, 0) {\pi};
  \node[below left] at (-pi/4, 0) {-\frac{\pi}{4}};
  \node[below left] at (-pi/2, 0) {-\frac{\pi}{2}};
  \node[below left] at (-3*pi/4, 0) {-\frac{3\pi}{4}};
  \node[below left] at (-pi, 0) {-\pi};
  \node[below left] at (0, 1) {\footnotesize $1$};
  \node[below left] at (0, 0.5) {\footnotesize $0.5$};
  \node[below left] at (0, -0.5) {\footnotesize $-0.5$};
  \node[below left] at (0, -1) {\footnotesize $-1$};
\end{tikzpicture}
```

Salida L^AT_EX:

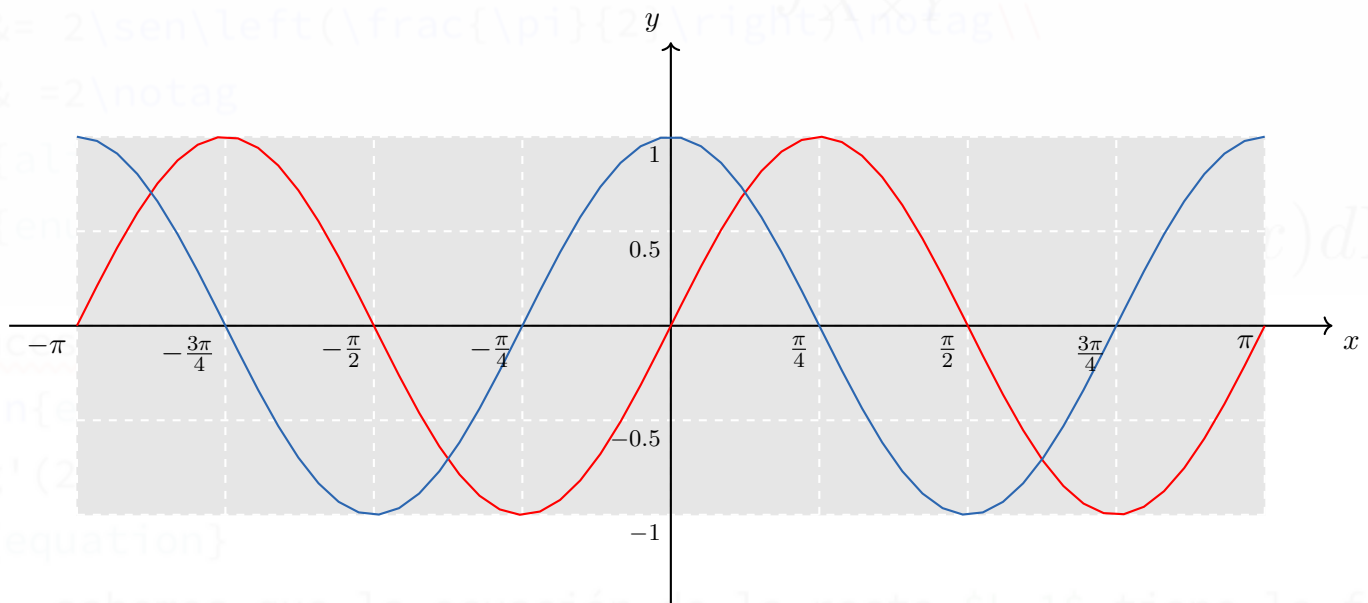


Figura 3: Gráficas de las funciones $\sin(2x)$ y $\cos(2x)$ con TikZ

Código L^AT_EX:

```

\begin{tikzpicture}[scale = 1.4]
  \draw[fill = background, background] (-4.5, -4) rectangle (4.5, 4);
  \draw[white, thick, xstep = pi/2, dashed] (-4.5, -4) grid (4.5, 4);
  \draw[<-] (4.85, 0) node [below right] {$x$} -- (-4.85, 0);
  \draw[<-] (0, 4.5) node[above left] {$y$} -- (0, -4.5);
  \draw[verde, thick, domain = -3*pi/2+0.25:-pi/2-0.25] plot ({\x},{tan(\x r)});
  \draw[verde, thick, domain = -pi/2+0.25:pi/2-0.25] plot ({\x},{tan(\x r)});
  \draw[verde, thick, domain = pi/2+0.25:3*pi/2-0.25] plot ({\x},{tan(\x r)});
  \draw[red, dashed] (pi/2, -4) -- (pi/2, 4);
  \draw[red, dashed] (-pi/2, -4) -- (-pi/2, 4);
  \node[below left] at (pi/2, 0) {$\frac{\pi}{2}$};
  \node[below left] at (pi, 0) {$\pi$};
  \node[below left] at (3*pi/2, 0) {$\frac{3\pi}{2}$};
  \node[below left] at (-pi/2, 0) {$-\frac{\pi}{2}$};
  \node[below left] at (-pi, 0) {$-\pi$};
  \node[below left] at (-3*pi/2, 0) {$-\frac{3\pi}{2}$};
  \foreach \x in {-4, -3, ..., 4}{
    \node[below left] at (0, \x) {$\x$};
  }
\end{tikzpicture}

```

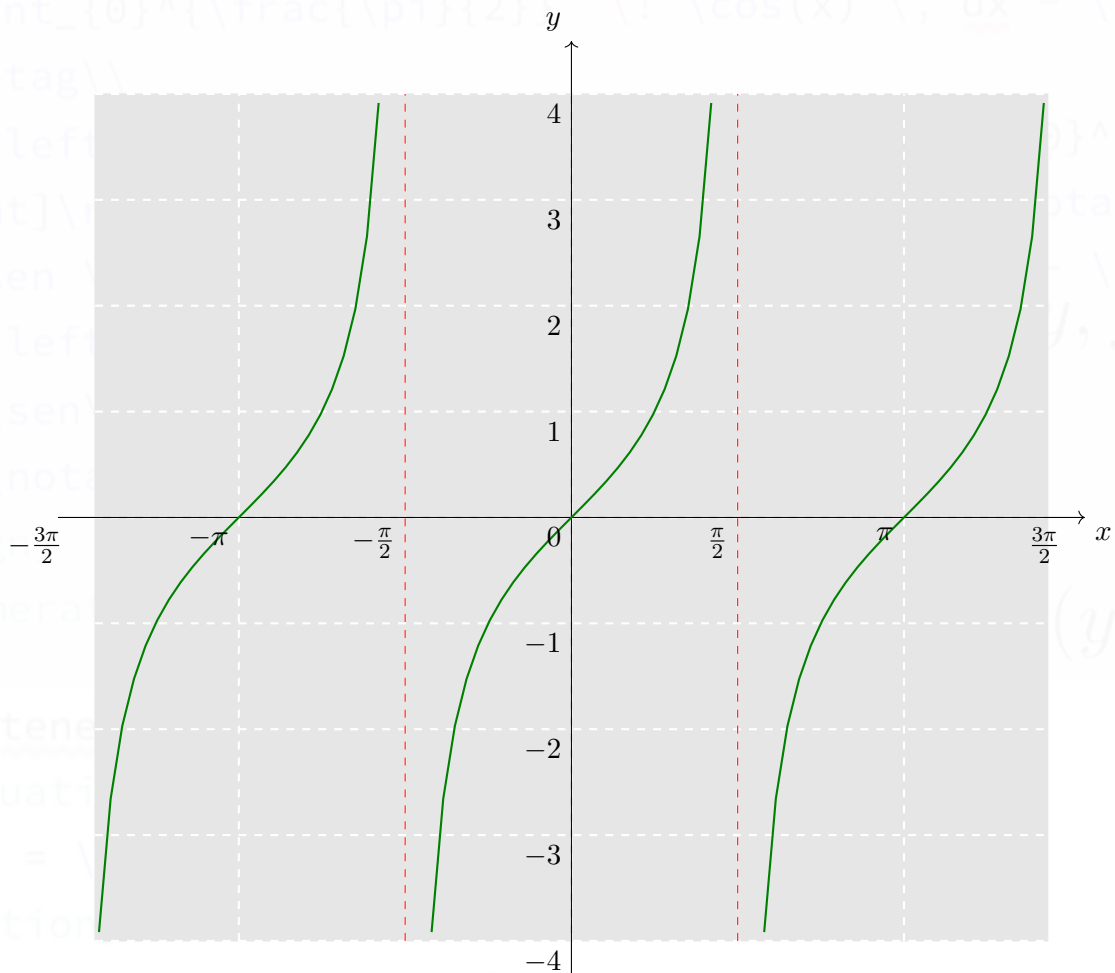
Salida L^AT_EX:

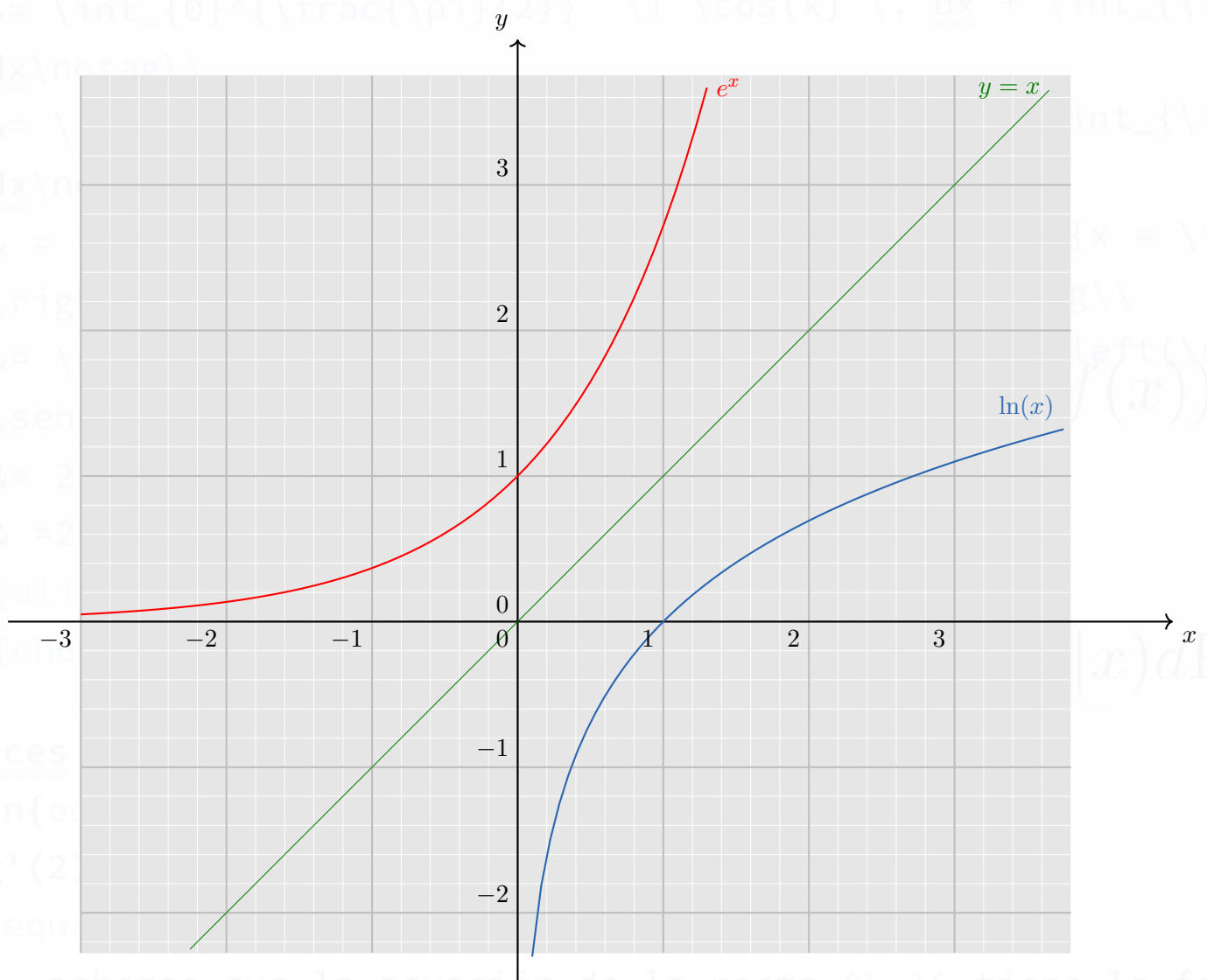
Figura 4: Gráfica de la función $\tan(x)$ en el dominio $(-\frac{3\pi}{2}, -\frac{\pi}{2}) \cup (-\frac{\pi}{2}, \frac{\pi}{2}) \cup (\frac{\pi}{2}, \frac{3\pi}{2})$

Código L^AT_EX:

```

\begin{tikzpicture}[scale = 2, samples = 60]
  \draw[background, fill = background] (-3, -2.275) rectangle (3.8, 3.75);
  \draw[ultra thin, white, xstep = 0.2, ystep = 0.2] (-3, -2.275) grid (3.8, 3.75);
  \draw[thick, lightgray, xstep = 1, ystep = 1] (-3, -2.275) grid (3.8, 3.75);
  \draw[<-, thick] (4.5, 0) node [below right] {$x$} -- (-3.5, 0);
  \draw[<-, thick] (0, 4) node [above left] {$y$} -- (0, -2.5);
  \draw[red, thick, domain = -3:1.3] plot({\x},{exp(\x)}) node[right] {$e^x$} ;
  \draw[azul, thick, domain = 0.1:3.75] plot({\x},{ln(\x)}) node[above left]
    {$\ln(x)$};
  \draw[verde] (-2.25, -2.25) -- (3.65,3.65) node[left] {$y=x$};
  \foreach \x in {-3, -2, ..., 3}{
    \node[below left] at (\x, 0) {$\x$};
  }
  \foreach \y in {-2, -1, ..., 3}{
    \node[above left] at (0, \y) {$\y$};
  }
\end{tikzpicture}

```

Salida L^AT_EX:Figura 5: Gráficas de las funciones $\ln(x)$ y e^x con TikZ

Mostraremos una lista de funciones que podemos usar en el entorno `tikzpicture` como hemos visto en los ejemplos anteriores:

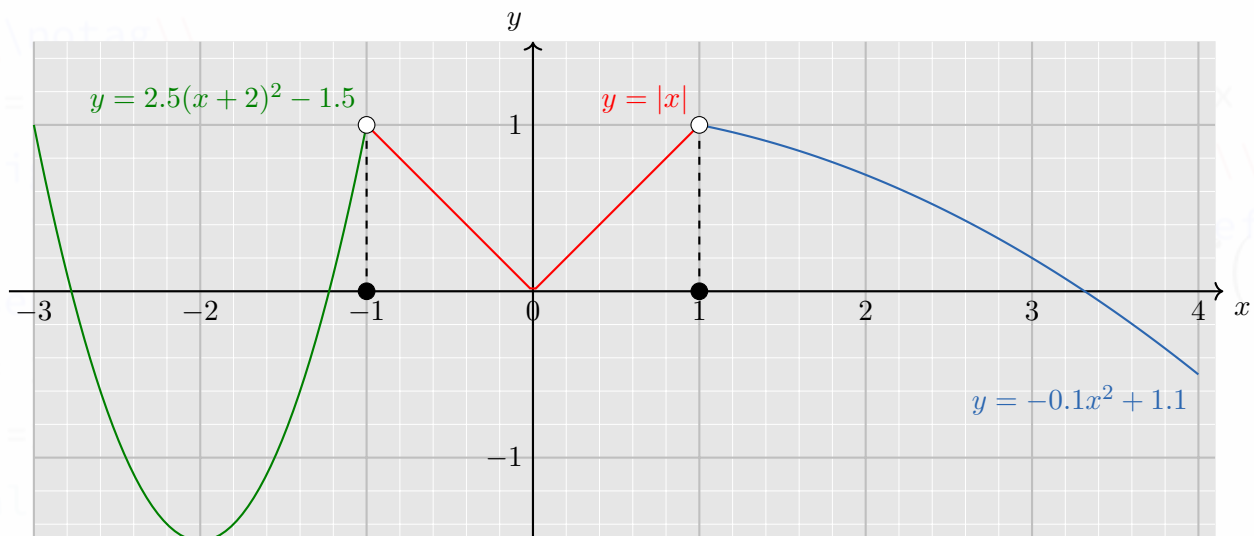
Tabla 1: Lista de algunas funciones matemáticas

Función matemática	Entrada L ^A T _E X	Función matemática	Entrada L ^A T _E X
$x!$	<code>factorial(\x)</code>	\sqrt{x}	<code>sqrt(\x)</code>
x^y	<code>pow(\x,y)</code>	e^x	<code>exp(\x)</code>
$\ln(x)$	<code>ln(\x)</code>	$\log(x)$	<code>log10(\x)</code>
$\log_2(x)$	<code>log2(\x)</code>	$ x $	<code>abs(\x)</code>
$x \bmod y$	<code>mod(\x,y)</code>	$\sin(x)$	<code>sin(\x r)</code>
$\cos(x)$	<code>cos(\x r)</code>	$\tan(x)$	<code>tan(\x r)</code>
$\cot(x)$	<code>cot(\x r)</code>	$\sec(x)$	<code>sec(\x r)</code>
$\arcsen(x)$	<code>rad(asin(\x))</code>	$\arccos(x)$	<code>rad(acos(\x))</code>

Se deja al lector como retos, graficar las siguientes funciones.

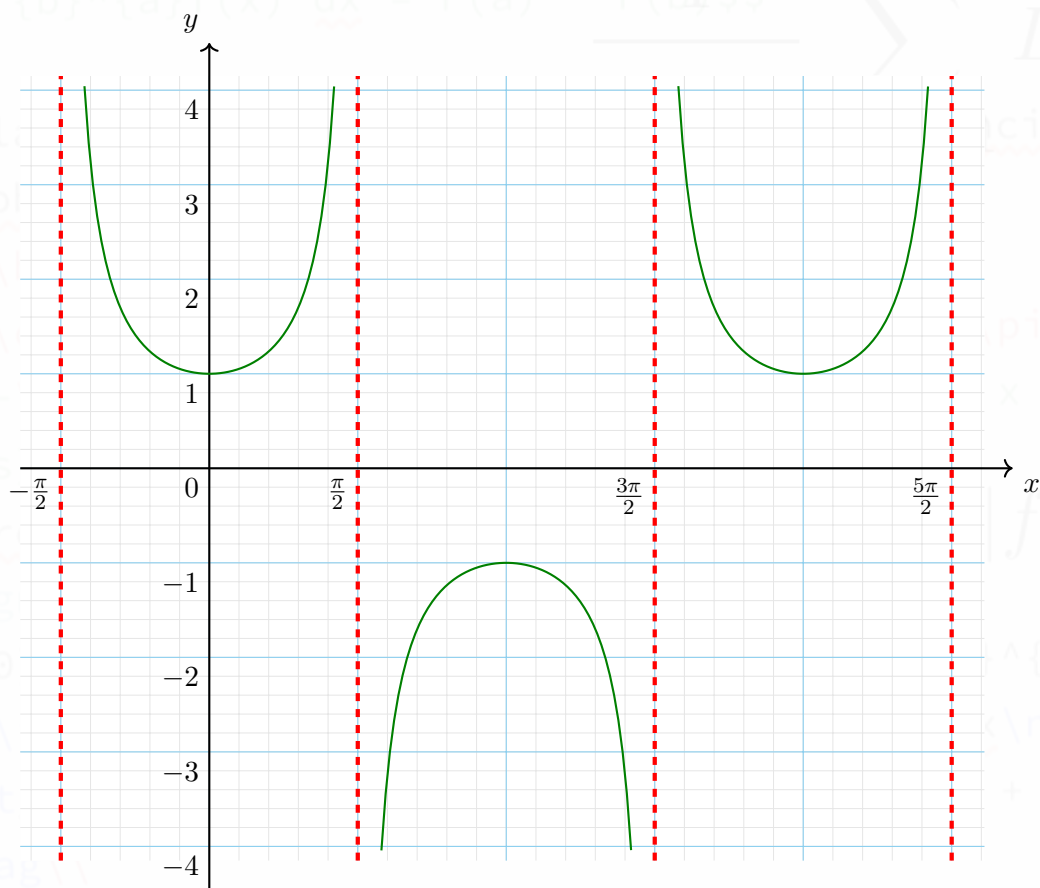
- Grafique:

$$f(x) = \begin{cases} 2.5(x+2)^2 - 1.5 & , -3 < x < -1 \\ 0 & , x = -1 \\ |x| & , -1 < x < 1 \\ 0 & , x = 1 \\ -0.1x^2 + 1.1 & ; 1 < x < 4 \end{cases}$$

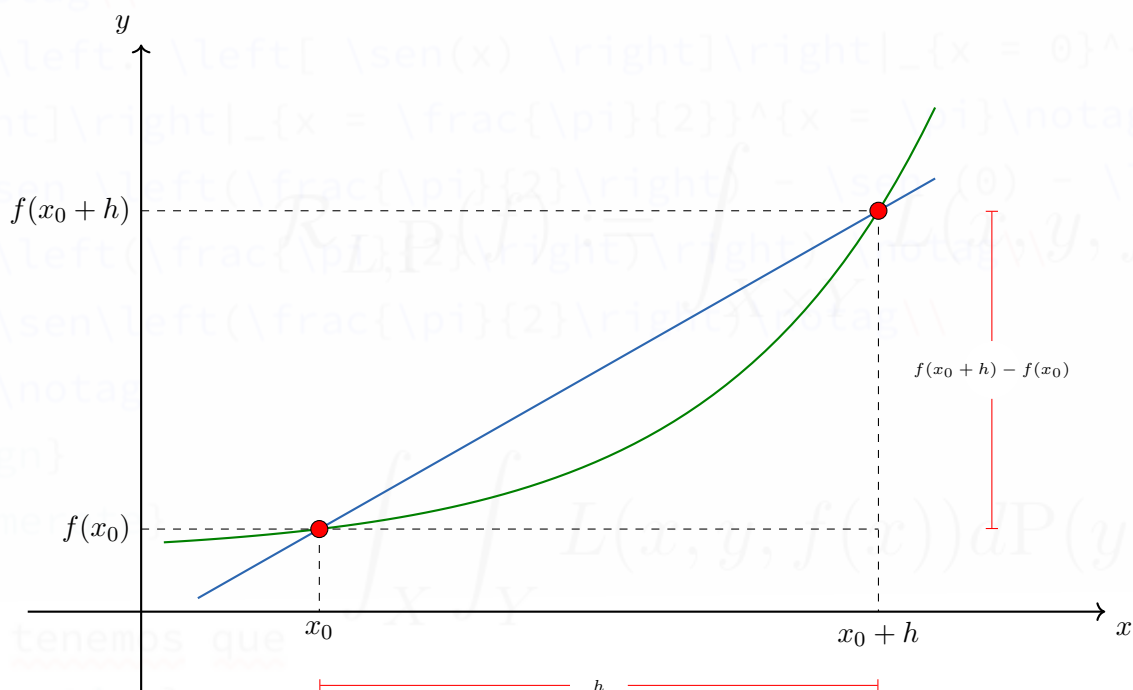


- Grafique

$$f(x) = \sec(x), x \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right) \cup \left(\frac{\pi}{2}, \frac{3\pi}{2}\right) \cup \left(\frac{3\pi}{2}, \frac{5\pi}{2}\right).$$



- Replique. Tome en cuenta que la función es $0.1e^{0.525x} + 0.5$



1.5. Gráficas con el entorno axis

Haremos uso del paquete `pgfplots` y la configuración `\pgfplotsset{compat=newest}`, podemos hacer uso de un entorno dentro del entorno `tikzpicture` y realizar algunas modificaciones a nuestro gusto. Veamos unos ejemplos:

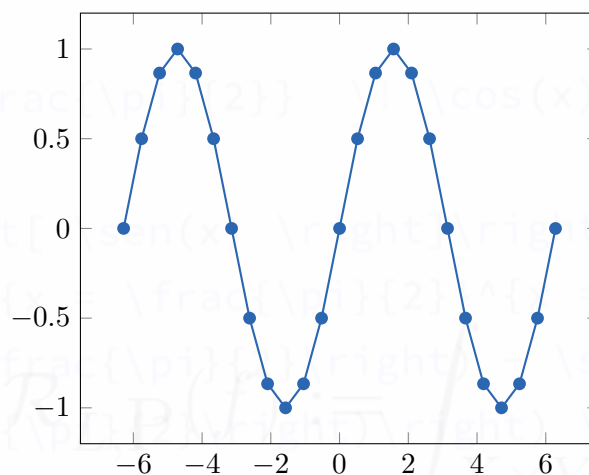
Graficaremos una función $f(x) = \sin(x)$ con el dominio en $[-2\pi, 2\pi]$ para observar los resultados y el proceso que podemos obtener con el entorno `axis`.

- Conociendo el entorno `axis`

Código L^AT_EX:

```
\begin{tikzpicture}
  \begin{axis}
    \addplot[mark = *, azul, thick, domain = -2*pi:2*pi]{sin(deg(x))};
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:

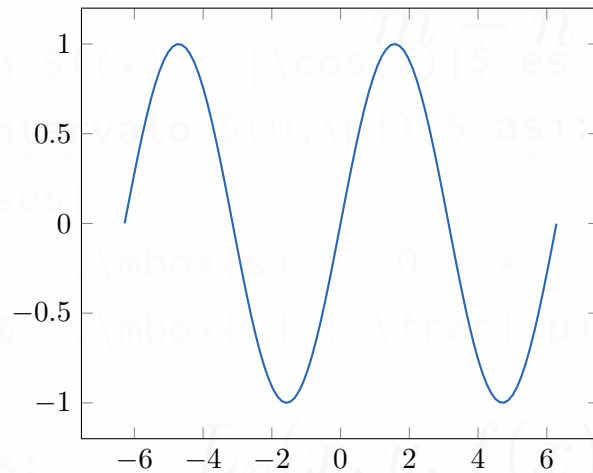


- Sin marcadores y determinando el número de muestras:

Código L^AT_EX:

```
\begin{tikzpicture}[samples = 100]
  \begin{axis}
    \addplot[mark = none, azul, thick, domain = -2*pi:2*pi]{sin(deg(x))};
  \end{axis}
\end{tikzpicture}
```


Salida L^AT_EX:

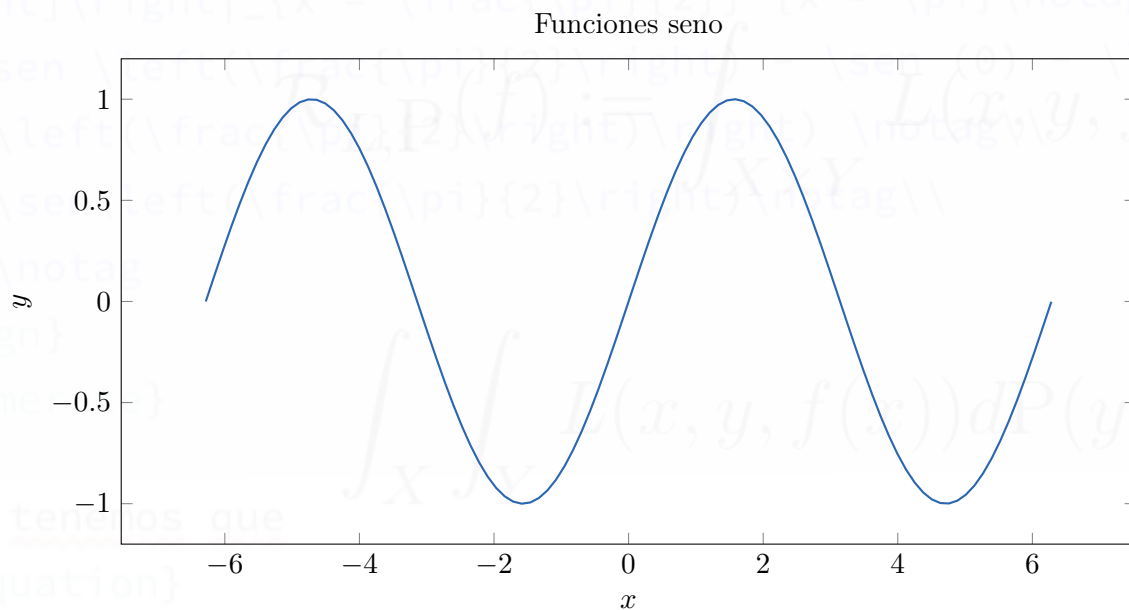


- Colocando un título, variando las dimensiones de la gráfica y etiquetas de los ejes:

Código L^AT_EX:

```
\begin{tikzpicture}[samples = 100]
  \begin{axis}[title=Funciones seno, height=4cm, width=15cm, xlabel=$x$,
    ylabel=$y$]
    \addplot[mark = none, azul, thick, domain = -2*pi:2*pi]{sin(deg(x))};
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:

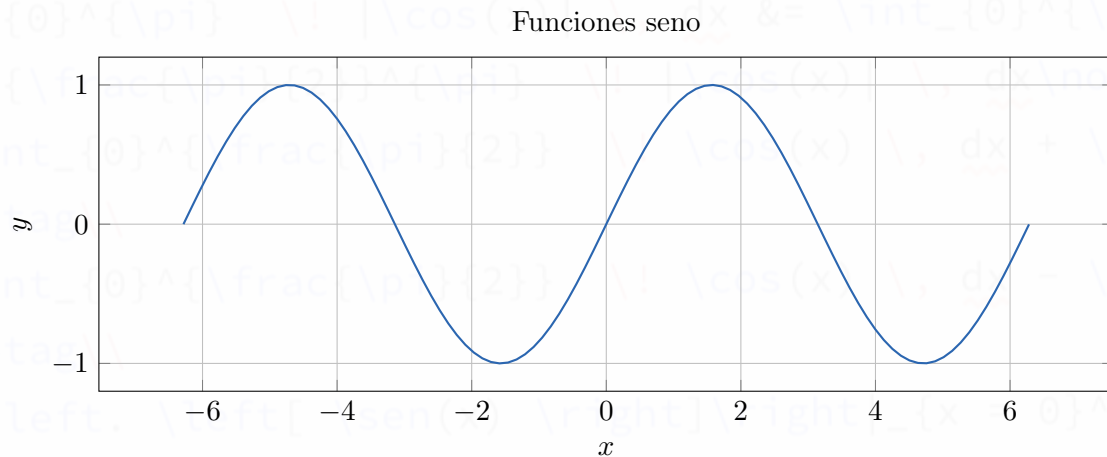


- Agregando una cuadrícula de fondo.

Código L^AT_EX:

```
\begin{tikzpicture}[samples = 100]
  \begin{axis}[
    title=Funciones seno,
    height=6cm, width=15cm,
    xlabel=$x$, ylabel=$y$,
    grid = both]
    \addplot[mark = none, azul, thick, domain = -2*pi:2*pi]{sin(deg(x))};
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:

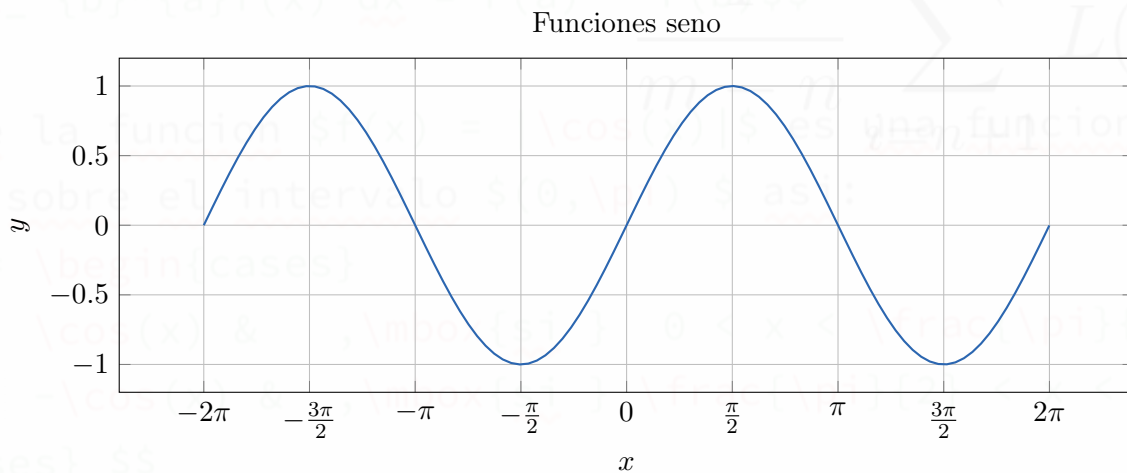


- Cambiando el tamaño de paso de la cuadrícula de fondo.

Código L^AT_EX:

```
\begin{tikzpicture}[samples = 100]
  \begin{axis}[
    title=Funciones seno,
    height=6cm, width=15cm,
    xlabel=$x$, ylabel=$y$,
    grid = both,
    ytick distance = 0.5,
    xtick = {-6.283185, -4.712389, -3.141592, -1.570796, 0,
             1.570796, 3.141592, 4.712389, 6.283185},
    xticklabels={-2 \pi$, $-\frac{3\pi}{2}$, $-\pi$, $-\frac{\pi}{2}$, $0$,
                 $\frac{\pi}{2}$, $\pi$, $\frac{3\pi}{2}$, $2\pi$}]
    \addplot[mark = none, azul, thick, domain = -2*pi:2*pi]{sin(deg(x))};
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:

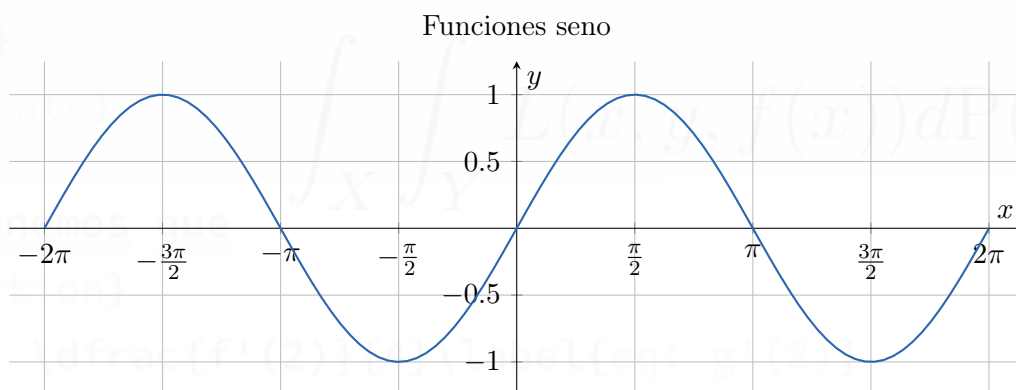


- Cambiando el estilo de los ejes y los valores máximos y mínimos:

Código L^AT_EX:

```
\begin{tikzpicture}[samples = 100]
  \begin{axis}[
    title=Funciones seno,
    height=6cm, width=15cm,
    xlabel=$x$, ylabel=$y$,
    grid = both,
    axis y line=center,
    axis x line=middle,
    ytick distance = 0.5,
    xtick = {-6.283185, -4.712389, -3.141592, -1.570796, 0,
      1.570796, 3.141592, 4.712389, 6.283185},
    xticklabels={$-2 \pi$, $-\frac{3\pi}{2}$, $-\pi$, $-\frac{\pi}{2}$, $0$,
      $\frac{\pi}{2}$, $\pi$, $\frac{3\pi}{2}$, $2\pi$}]
    \addplot[mark = none, azul, thick, domain = -2*pi:2*pi]{sin(deg(x))};
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:

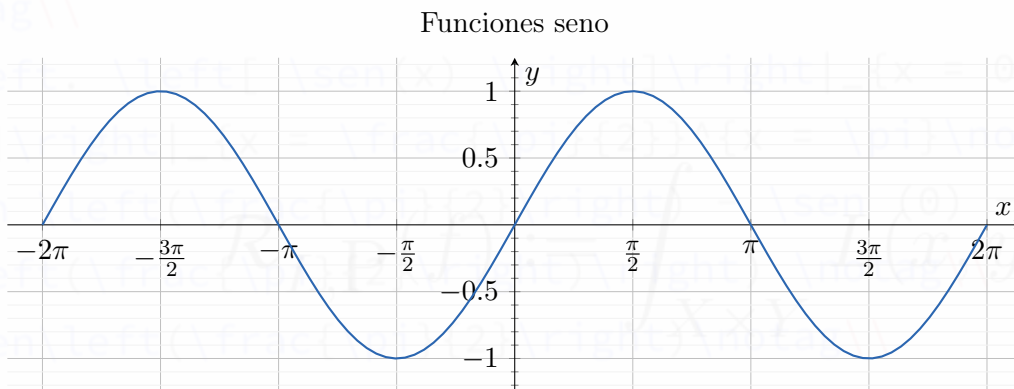


- Aumentango la cuadrícula

Código L^AT_EX:

```
\begin{tikzpicture}[samples = 100]
  \begin{axis}[
    title=Funciones seno,
    height=6cm, width=15cm,
    xlabel=$x$, ylabel=$y$,
    grid = both,
    axis y line=center,
    axis x line=middle,
    grid = both,
    minor tick num=4,
    grid style={line width=.1pt, draw=gray!10},
    major grid style={line width=0.25pt, draw=gray!50},
    ytick distance = 0.5,
    xtick = {-6.283185, -4.712389, -3.141592, -1.570796, 0,
             1.570796, 3.141592, 4.712389, 6.283185},
    xticklabels={$-2 \pi$, $-\frac{3\pi}{2}$, $-\pi$, $-\frac{\pi}{2}$, $0$,
                 $\frac{\pi}{2}$, $\pi$, $\frac{3\pi}{2}$, $2\pi$}]
    \addplot[mark = none, azul, thick, domain = -2*pi:2*pi]{sin(deg(x))};
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:



- Agregando una función y colocando una leyenda.

Código L^AT_EX:

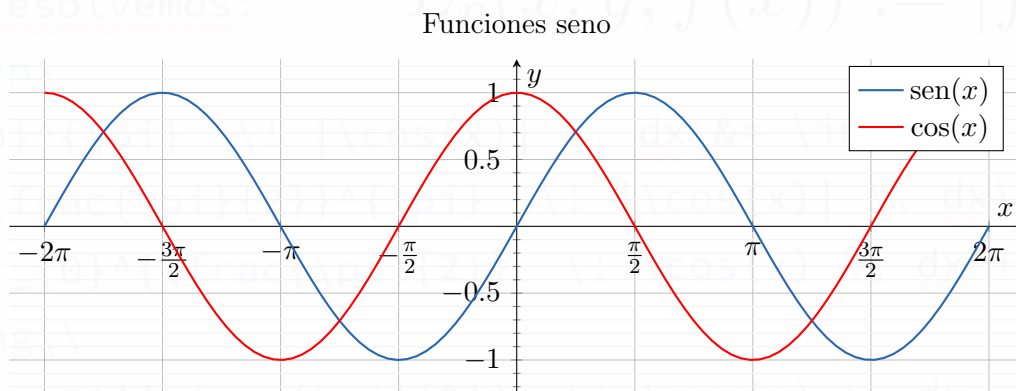
```
\begin{tikzpicture}[samples = 100]
  \begin{axis}[
    title=Funciones seno,
    height=6cm, width=15cm,
    xlabel=$x$, ylabel=$y$,
    grid = both,
    axis y line=center,
    axis x line=middle,
    grid = both,
    minor tick num=4,
    grid style={line width=.1pt, draw=gray!10},
```

```

major grid style={line width=0.25pt, draw=gray!50},
ytick distance = 0.5,
xtick = {-6.283185, -4.712389, -3.141592, -1.570796, 0,
         1.570796, 3.141592, 4.712389, 6.283185},
xticklabels={-2 \pi$, $-\frac{3\pi}{2}$, $-\pi$, $-\frac{\pi}{2}$, $0$,
             $\frac{\pi}{2}$, $\pi$, $\frac{3\pi}{2}$, $2\pi$}
\addplot[mark = none, azul, thick, domain = -2*pi:2*pi]{sin(deg(x))};
\addplot[mark = none, red, thick, domain = -2*pi:2*pi]{cos(deg(x))};
\legend{$\sen(x)$, $\cos(x)$};
\end{axis}
\end{tikzpicture}

```

Salida L^AT_EX:



1.6. Gráficos Estadísticos

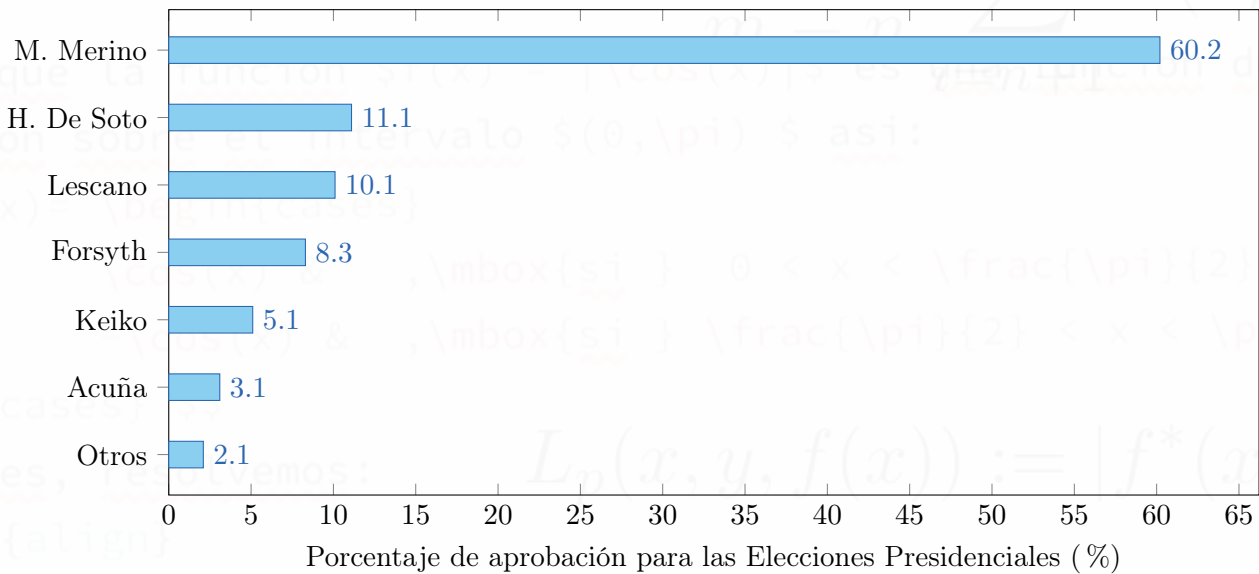
Gráficos de Barras

Código L^AT_EX:

```

\begin{tikzpicture}
\begin{axis}[
xbar, xmin=0,
width=16cm, height=8cm,%, enlarge y limits=0.2,
xlabel={Porcentaje de aprobación para las Elecciones Presidenciales (\%)},
symbolic y coords={Otros, Acuña, Keiko, Forsyth, Lescano, H. De Soto,
M. Merino},
nodes near coords,
nodes near coords align={horizontal},
]
\addplot[azul, fill = babyblue] coordinates {
(2.1,Otros)
(3.1,Acuña)
(5.1,Keiko)
(8.4,Forsyth)
(10.0,Lescano)
(11.1,H. De Soto)
(60.2,M. Merino) };
\end{axis}
\end{tikzpicture}

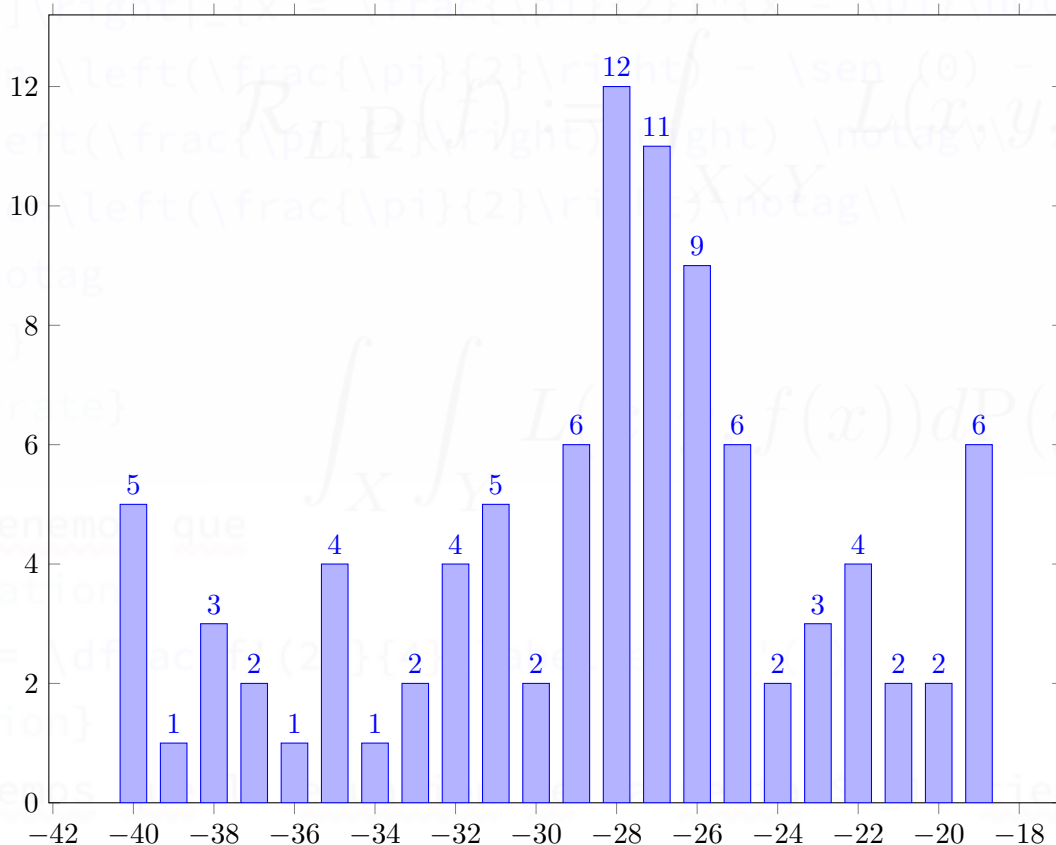
```


Salida L^AT_EX:

Extrayendo los datos de un documento `datos_barras.txt` y realizando el gráfico de barras:

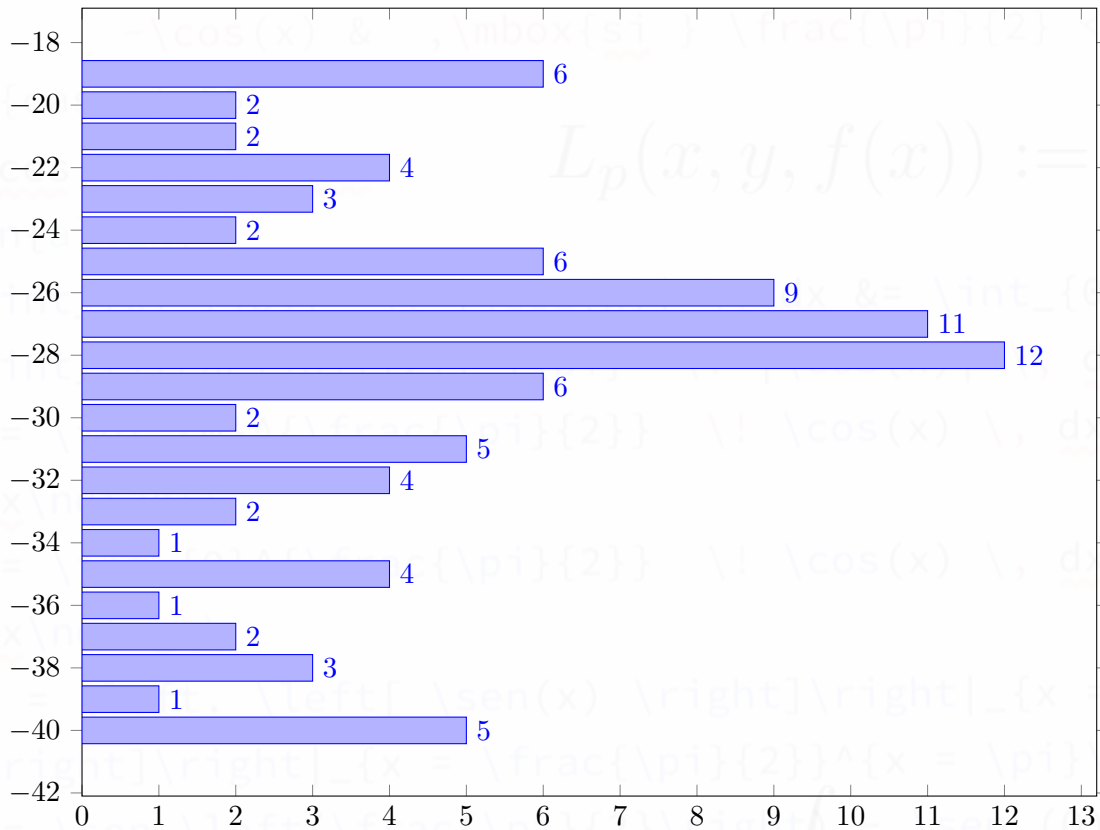
Código L^AT_EX:

```
\begin{tikzpicture}
  \begin{axis}[ybar, height=12cm,width=15cm, nodes near coords, ymin = 0]
    \addplot table[x index=0, y index=1] {datos_barras.txt};
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:

Código L^AT_EX:

```
\begin{tikzpicture}
  \begin{axis}[xbar, height=12cm,width=15cm, nodes near coords, xmin = 0]
    \addplot table[y index=0, x index=1] {datos_barras.txt};
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:**Gráfico de dispersión con Clases****Código L^AT_EX:**

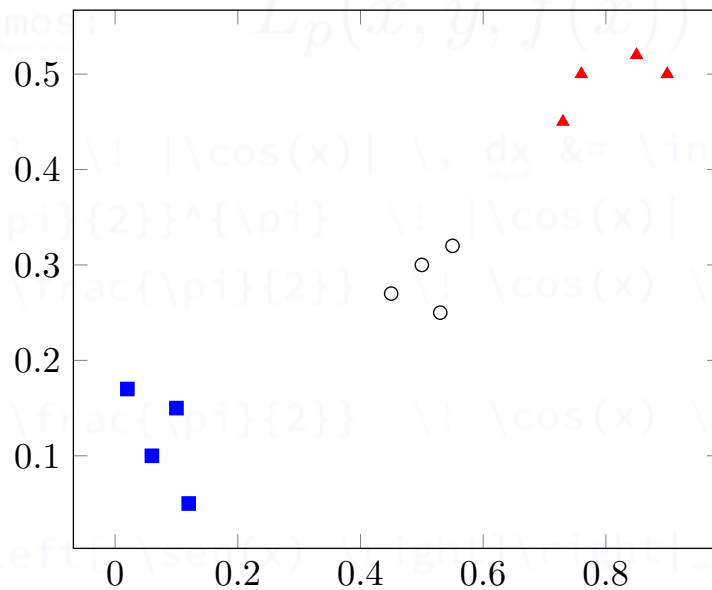
```
\begin{tikzpicture}
  \begin{axis}[%
    scatter/classes={%
      a={mark=square*,blue},%
      b={mark=triangle*,red},%
      c={mark=o,draw=black}}]
    \addplot[
      scatter,
      only marks,%
      scatter src=explicit symbolic]%
    table[meta=label] {
      x      y      label
      0.1    0.15   a
      0.45   0.27   c
      0.02    0.17   a
      0.06    0.1    a
    }
  \end{axis}
\end{tikzpicture}
```

```

0.9 0.5 b
0.5 0.3 c
0.85 0.52 b
0.12 0.05 a
0.73 0.45 b
0.53 0.25 c
0.76 0.5 b
0.55 0.32 c
};
\end{axis}
\end{tikzpicture}

```

Salida L^AT_EX:



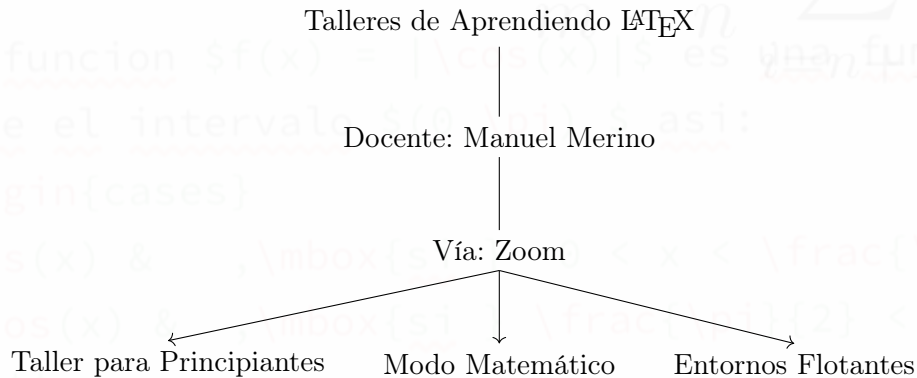
1.7. Esquemas o gráficos

Código L^AT_EX:

```

\begin{tikzpicture}[node distance = 1.5cm, auto]
  \node(via){Vía: Zoom};
  \node[above of=via](docente){Docente: Manuel Merino};
  \node[above of=docente](talleres){Talleres de Aprendiendo \LaTeX};
  \node[below of= via](mate){Modo Matemático};
  \node[left=0.5cm of mate](principiantes){Taller para Principiantes};
  \node[right=0.5cm of mate](float){Entornos Flotantes};
  %lineas
  \draw (talleres.south) -- (docente.north) (docente.south) -- (via.north);
  \draw[<-] (principiantes.north) -- (via.south);
  \draw[<-] (mate.north) -- (via.south);
  \draw[<-] (float.north) -- (via.south);
\end{tikzpicture}

```

Salida L^AT_EX:Código L^AT_EX:

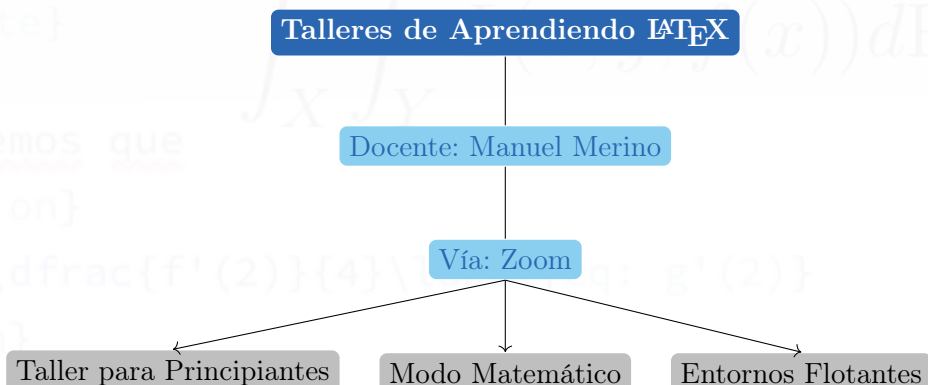
```

\begin{tikzpicture}[node distance = 1.5cm, auto]
  \node[azul, thick, rectangle, fill=babyblue, rounded corners=0.1cm](via)
    {Vía: Zoom};
  \node[azul, thick, rectangle, fill=babyblue, rounded corners=0.1cm,
    above of=via](docente){Docente: Manuel Merino};
  \node[azul, thick, rectangle, fill=azul, text = white, font = \bf,
    rounded corners=0.1cm, above of=docente](talleres)
    {Talleres de Aprendiendo \LaTeX};
  \node[black, thick, rectangle, fill=lightgray, rounded corners=0.1cm,
    below of= via](mate){Modo Matemático};
  \node[black, thick, rectangle, fill=lightgray, rounded corners=0.1cm,
    left=0.5cm of mate](principiantes){Taller para Principiantes};
  \node[black, thick, rectangle, fill=lightgray, rounded corners=0.1cm,
    right=0.5cm of mate](float){Entornos Flotantes};

  \draw (talleres.south) -- (docente.north) (docente.south) -- (via.north);

  \draw[<-] (principiantes.north) -- (via.south);
  \draw[<-] (mate.north) -- (via.south);
  \draw[<-] (float.north) -- (via.south);
\end{tikzpicture}

```

Salida L^AT_EX:

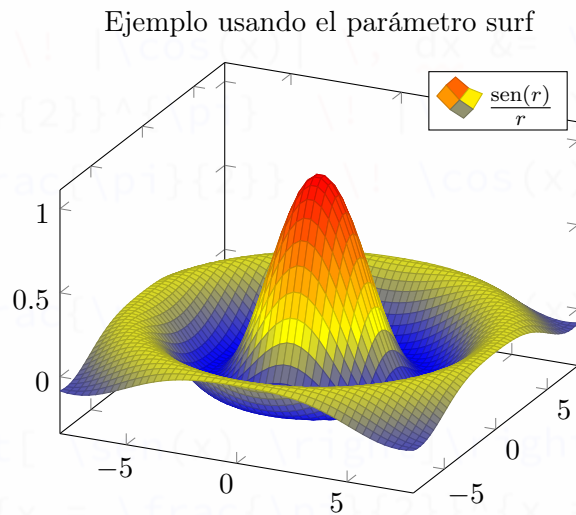
2. Introducción a las imágenes 3D

En esta sección haremos uso del comando `addplot3` para introducir las imágenes tridimensionales. Veamos un ejemplo

Código L^AT_EX:

```
\begin{tikzpicture}
  \begin{axis}[title=Ejemplo usando el parámetro surf]
    \addplot3[surf, samples=50, domain=-8:8] {\sin(deg(sqrt(x^2+y^2)))/sqrt(x^2+y^2)};
    \legend{\frac{\sin(r)}{r}}
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:

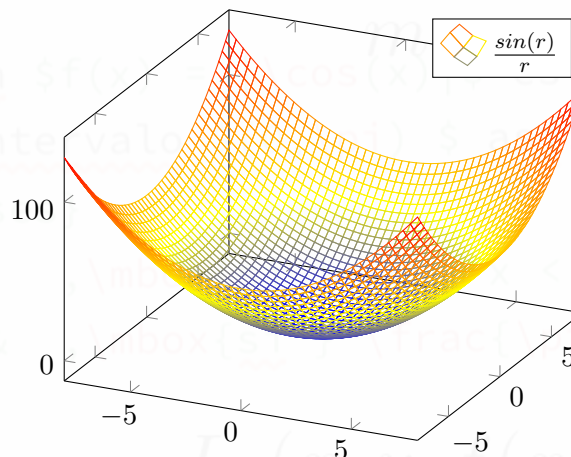


Código L^AT_EX:

```
\begin{tikzpicture}
  \begin{axis}[title=Ejemplo usando el parámetro surf]
    \addplot3[mesh,samples=50,domain=-8:8]{x^2 + y^2};
    \legend{$x^2 + y^2$}
  \end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:

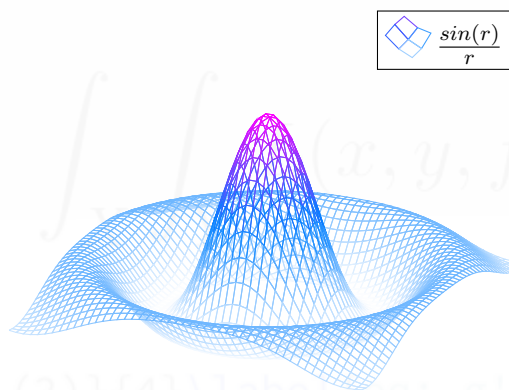
Ejemplo usando el parámetro surf

**Código L^AT_EX:**

```
\begin{tikzpicture}
\begin{axis}[
title=Ejemplo usando el parámetro mesh,
hide axis,
colormap/cool,
]
\addplot3[
mesh,
samples=50,
domain=-8:8,
]
{sin(deg(sqrt(x^2+y^2)))/sqrt(x^2+y^2)};
\addlegendentry{$\frac{\sin(r)}{r}$}
\end{axis}
\end{tikzpicture}
```

Salida L^AT_EX:

Ejemplo usando el parámetro mesh



Referencias

Kern, U. (2011). *Extending L^AT_EX's color facilities: the xcolor package*. Germany: Institut für Theoretische Informatik.

Tantau, T. (2020). *The TikZ and PGF Packages. Manual for version 3.1.5b*. Germany: Institut für Theoretische Informatik.

Zavala, R. (2020). *L^AT_EXnomicón*. México: Quiah Editorial.

Entonces, resolvemos:

$$L_p(x, y, f(x)) := |f^*(x) - f(x)|^p$$

$$\int_0^{\pi} |\cos(x)| \, dx = \int_0^{\frac{\pi}{2}} |\cos(x)| \, dx + \int_{\frac{\pi}{2}}^{\pi} |\cos(x)| \, dx$$

$$= \int_0^{\frac{\pi}{2}} \cos(x) \, dx + \int_{\frac{\pi}{2}}^{\pi} -\cos(x) \, dx$$

$$= \left[\sin(x) \right]_0^{\frac{\pi}{2}} - \left[\sin(x) \right]_{\frac{\pi}{2}}^{\pi}$$

$$= \sin\left(\frac{\pi}{2}\right) - \sin(0) - \left(\sin(\pi) - \sin\left(\frac{\pi}{2}\right)\right)$$

$$= 2\sin\left(\frac{\pi}{2}\right)$$

$$= 2$$

Entonces tenemos que

$$g'(2) = \frac{f'(2)}{4} \quad \text{\label{eq: g'(2)}}$$

Luego, sabemos que la ecuación de la recta L_1 tiene la forma

$$g'(2)(x-2) \quad \text{\label{eq: l1}}$$