Práctica 3: Ecuaciones e inecuaciones. Aplicaciones.

En esta práctica analizaremos algunas propiedades de las funciones, tales como el cáculo de raíces, máximos y mínimos e intervalos de crecimiento y decrecimiento, usando *Mathematica*. Utilizaremos los comandos para agilizar nuestro trabajo y para comprobar los resultados, teniendo presente los conocimientos teóricos adquiridos en la asignatura.

1. Resolución de ecuaciones algebraicas

A lo largo de la práctica plantearemos la resolución de tres tipos de problemas representados mediante ecuaciones algebraicas:

- Cálculo de raíces: $f(x) = 0$
- Puntos de corte entre dos funciones: $f(x) = g(x)$
- Posibles extremos relativos de $f(x)$: $f'(x) = 0$

Tanto para hallar las raíces o los posibles extremos relativos de una función como para calcular las abscisas de los puntos de corte entre dos funciones, necesitaremos utilizar los comandos que posee *Mathematica* para resolver ecuaciones: Solve, NSolve o FindRoot. Veamos, en primer lugar, la sintaxis y el funcionamiento de estos tres comandos.

El comando **Solve** de *Mathematica* permite resolver ecuaciones, ya sea una única ecuación o un sistema de ecuaciones, utilizando métodos algebraicos (siempre y cuando esto sea posible). Es importante señalar que las **ecuaciones** se introducen en *Mathematica* con un doble signo de igualdad:

expresión1 == expresion2

En el caso de una única ecuación, el comando Solve tiene la siguiente sintaxis:

Solve[ecuación, variable]

Ejemplo. Resuelve la ecuación $x^3 - 3x^2 + x - 3 = 0$.

```
In[1]:= Solve[x^3 - 3 x^2 + x - 3 == 0, x]
        resuelve

Out[1]= {{x → -i}, {x → i}, {x → 3}}
```

Si solo buscamos **soluciones reales**, añadiremos la opción Reals como argumento:

Solve[ecuación, variable, Reals]

Ejemplo. Resuelve la ecuación $e^x = 2$.

```
In[2]:= Solve[Exp[x] == 2, x, Reals]
        resue… exponencial     números

Out[2]= {{x → Log[2]}}
```

Ejemplo. Resuelve la ecuación $x^5 + 2x = 1$.

```
In[3]:= Solve[x^5 + 2 x == 1, x, Reals]
        resuelve                  números

Out[3]= {{x → 0.486…}}
```

Vemos, en este último ejemplo, que *Mathematica* no es capaz de encontrar soluciones exactas de esta ecuación. Aparece una aproximación de una solución real y, si pulsamos sobre el símbolo de la raíz, observamos que ha hecho un cálculo en función del comando **Root** pero no es capaz de obtener una solución explícita. Esto es natural por tratarse de una ecuación polinómica de quinto grado, para la que no existe una fórmula general de resolución.

En estos casos podemos usar el comando **NSolve**, que utiliza métodos numéricos de resolución de ecuaciones para encontrar aproximaciones decimales de las soluciones. La sintaxis del comando **NSolve** es la misma que la del comando **Solve**:

NSolve[ecuación, variable] o NSolve[ecuación, variable, Reals] (para soluciones reales)

Así, aplicando **NSolve** al ejemplo anterior, obtenemos

```
In[4]:= NSolve[x^5 + 2 x == 1, x, Reals]
        resolvedor numérico        números

Out[4]= {{x → 0.486389}}
```

Si queremos especificar un número de cifras significativas exactas, podemos escribir

NSolve[ecuación, variable, WorkingPrecision $->$ n]

donde n es el número de cifras:

```
In[5]:= NSolve[x^5 + 2 x == 1, x, Reals, WorkingPrecision → 10]
        resolvedor numérico        número·· precisión operativa

Out[5]= {{x → 0.4863890359}}
```

A veces *Mathematica* nos dice que no puede aproximar las soluciones de la ecuación con la instrucción **NSolve**. Se puede usar entonces una instrucción alternativa, **FindRoot**. Este comando aplica un método iterativo para aproximar numéricamente **una única** solución de la ecuación, por lo que si el problema tiene varias soluciones habrá que ejecutar la orden varias veces, modificando adecuadamente sus argumentos, para hallar todas las soluciones que buscamos. Su sintaxis admite dos variantes:

FindRoot[ecuación, {variable, x_0}] o **FindRoot**[ecuación, {variable, a, b}]

En el primer caso, x_0 es un valor inicial del que parte el método iterativo utilizado por **FindRoot** para aproximar la raíz. Este punto debe elegirse *suficientemente* cerca de la solución que queremos aproximar; mientras que en el segundo formato, a y b son, respectivamente, los extremos del intervalo [a, b] en el que se halla la solución que buscamos. Una representación gráfica previa[1] puede ser de gran utilidad a la hora de seleccionar tanto el punto inicial, x_0, como los extremos del intervalo [a, b].

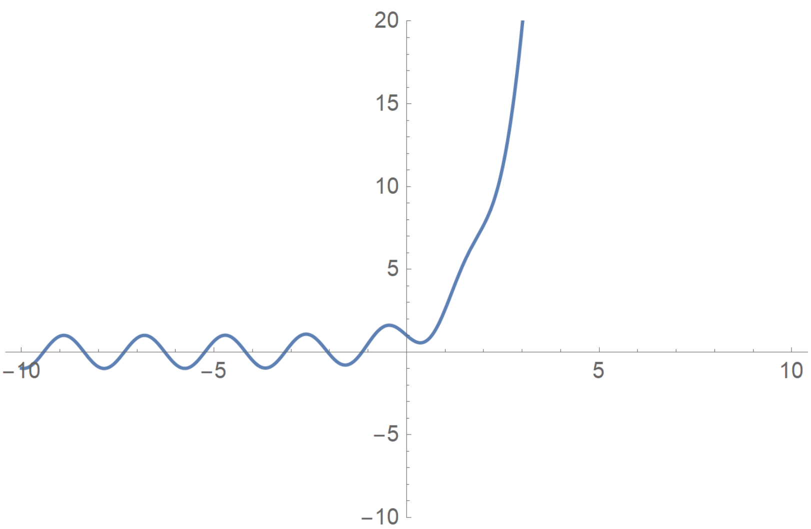
También en el comando **FindRoot** podemos especificar el número de cifras significativas exactas, añadiendo la opción WorkingPrecision $->n$.

Ejemplo. Observa gráficamente que la función $f(x) = e^x + \sin(3x)$ posee infinitas raíces reales. Aplica el comando **FindRoot** para aproximar la raíz más cercana al origen.

A partir de una gráfica adecuada podemos observar que la función posee infinitas raíces. En efecto,

```
In[6]:= Plot[Exp[x] - Sin[3 x], {x, -10, 10}, PlotRange -> {-10, 20}]
         repr··· exponen··· seno                 rango de representación
```

Out[6]=



Comprueba que no puedes resolverlo mediante el comando **NSolve**. Usaremos, por

[1]Recuerda que si te situás sobre la gráfica y eliges la opción *Obtener coordenadas* en el desplegable que aparece al pulsar el botón derecho del ratón, el cursor se convierte en una cruz y aparecen en pantalla las coordenadas del punto sobre el que se encuentra la cruz.

tanto, **FindRoot**. Eligiendo como punto inicial del método $x_0 = 0$, obtenemos:

```
In[7]:= FindRoot[Exp[x] - Sin[3 x] == 0, {x, 0}]
        encuentra··· exponen··· seno

Out[7]= {x → -3.1558}
```

Sin embargo, la solución que devuelve el método no es la que buscamos, ni tampoco la más cercana al punto de partida. Observando la gráfica, vemos que -1 podría ser un punto inicial más adecuado ya que está más próximo a la raíz pedida. En este caso,

```
In[8]:= FindRoot[Exp[x] - Sin[3 x] == 0, {x, -1}]
        encuentra··· exponen··· seno

Out[8]= {x → -1.15413}
```

que devuelve la aproximación que nos pedían.

Con el fin de forzar el cálculo aproximado de la raíz que buscamos, conviene indicar un intervalo en el que está dicha raíz (y solo esa). En este caso, observamos, a partir de la gráfica, que la raíz pedida está en el intervalo $[-2, -1]$ y podemos obtenerla mediante:

```
In[9]:= FindRoot[Exp[x] - Sin[3 x] == 0, {x, -2, -1}]
        encuentra··· exponen··· seno

Out[9]= {x → -1.15413}
```

Además, podemos aumentar el número de decimales exactos de las aproximaciones. Por ejemplo, para aproximar la mayor de las raíces con 20 dígitos exactos, escribimos:

```
In[10]:= FindRoot[Exp[x] - Sin[3 x] == 0, {x, -2, -1}, WorkingPrecision → 20]
         encuentra··· exponen··· seno                    precisión operativa

Out[10]=
         {x → -1.1541326646068919041}
```

2. Resolución de inecuaciones

Las inecuaciones se plantean igual que las ecuaciones pero sustituyendo el signo == por $<$, $\leq$, $>$, $\geq$, según corresponda. Para resolver una inecuación del tipo

$$f(x) \leq g(x)$$

(o con $\geq$, $<$, $>$ en lugar de $\leq$) podemos emplear la instrucción **Reduce**, cuya sintaxis es

Reduce[inecuación, variable]

Ejemplo. Resuelve la inecuación $x^2 + 2x \geq 2$.

```
In[11]:= Reduce[x^2 + 2 x ≥ 2, x]
         reduce

Out[11]=
         x ≤ -1 - √3 || x ≥ -1 + √3
```

A la hora de interpretar los resultados debemos tener en cuenta que el símbolo `||` significa "o" (disyunción o unión de conjuntos), mientras que `&&` denotaría "y" (conjunción o intersección de conjuntos). Por lo tanto, la solución de la inecuación anterior será la unión de los intervalos:

$$\left]-\infty, -1-\sqrt{3}\right] \cup \left[-1+\sqrt{3}, +\infty\right[.$$

En ocasiones, **Reduce** no resuelve la inecuación si no se indica el conjunto en el que buscamos las soluciones. Para ello hay que incluir un tercer argumento, dominio:

Reduce[inecuación, variable, dominio]

Normalmente será Reals, puesto que buscaremos soluciones reales, aunque también podríamos elegir Integers o Complexes.

Ejemplo. Resuelve la siguiente inecuación:

$$\operatorname{arctg}(x^3 + x) < 0$$

Observa que, si no se especifica el dominio, el programa devuelve un mensaje de error:

```
In[12]:= Reduce[ArcTan[x^3 + x] < 0, x]
         reduce  arco tangente

         ••• Reduce: This system cannot be solved with the methods available to Reduce.
Out[12]=
         Reduce[ArcTan[x + x^3] < 0, x]
```

Este problema se puede evitar indicando que nos interesan solo las soluciones reales. Así,

```
In[13]:= Reduce[ArcTan[x^3 + x] < 0, x, Reals]
         reduce  arco tangente              números
Out[13]=
         x < 0
```

Por lo tanto, la solución es el intervalo $]-\infty, 0[$.

El comando **Reduce** también resuelve sistemas de inecuaciones, que se introducen entre llaves como primer parámetro del comando.

Ejemplo. Resuelve el sistema de inecuaciones:

$$x^2 + 2x \geq 2, \;\; x - 4 \leq 0$$

Así

```
In[14]:= Reduce[{x^2 + 2 x ≥ 2, x - 4 ≤ 0}, x]
         reduce
Out[14]=
         x ≤ -1 - √3 || -1 + √3 ≤ x ≤ 4
```

por lo que la solución será

$$\left]-\infty, -1-\sqrt{3}\right] \cup \left[-1+\sqrt{3}, 4\right].$$

3. Cálculo de derivadas. Aplicaciones.

Para calcular la derivada de una función $f(x)$, previamente definida con *Mathematica*, disponemos de los siguientes comandos:

- **f´**[x] o **D**[f[x],x]: Calcula la derivada primera de f respecto de x, es decir, $f'(x)$.
- **f´´**[x] o bien **D**[f[x],{x,2}]: Halla la derivada segunda de f, esto es, $f''(x)$.
- **D**[f[x],{x,n}]: Calcula la derivada de orden n de f, es decir, $f^{(n)}(x)$.

Ejemplo. Halla la derivada primera de $f(x) = x^3 - 4x^2 + x + 6$.

```
In[15]:= f[x_] := x^3 - 4 x^2 + x + 6

In[16]:= f'[x]
Out[16]=
         1 - 8 x + 3 x^2

In[17]:= D[f[x], x]
         deriva
Out[17]=
         1 - 8 x + 3 x^2
```

Si queremos evaluar la derivada primera en $x = a$ podemos hacerlo de dos formas:

f´[a] o bien mediante **D**[f[x],x] $/.\ x \to a$

La misma regla se sigue para derivadas de órdenes superiores.

Ejemplo. Calcula $f'''(1)$, siendo $f(x) = e^{x^2} + \cos(x)$.

```
In[18]:= f[x_] := Exp[x^2] + Cos[x]
                   exponencial  coseno

In[19]:= f'''[1]

Out[19]=
         20 e + Sin[1]

In[20]:= D[f[x], {x, 3}] /. x → 1
         deriva

Out[20]=
         20 e + Sin[1]
```

Observa, sin embargo, que no es correcto usar **D[f[a],{x,n}]** ya que, en este caso, calcula la derivada de $f(a)$ respecto de x que, al ser constante, siempre valdrá 0. Compruébalo.

En general, se pueden estudiar algunas propiedades geométricas de las funciones analizando su derivada. Recuerda los siguientes resultados teóricos:

Crecimiento y decrecimiento

Si $f'(x) > 0$ en un intervalo $I \Rightarrow f$ es estrictamente creciente en I

Si $f'(x) < 0$ en un intervalo $I \Rightarrow f$ es estrictamente decreciente en I

Cálculo de extremos relativos

Si f es derivable y alcanza un extremo relativo en $x_0 \Rightarrow f'(x_0) = 0$

Por tanto, los posibles extremos relativos de una función se hallan resolviendo la ecuación[2] $f'(x) = 0$. Una vez calculados estos valores, para determinar si se trata de un máximo o un mínimo, hay que tener en cuenta que en un mínimo relativo la curva es cóncava y en un máximo relativo la curva es convexa, es decir,

Si $f'(x_0) = 0$ y $f''(x_0) > 0$, tenemos un mínimo en x_0

Si $f'(x_0) = 0$ y $f''(x_0) < 0$, tenemos un máximo en x_0.

Puesto que los posibles extremos relativos de una función derivable, $f(x)$, se obtienen resolviendo la ecuación $f'(x) = 0$ y los intervalos de crecimiento y decrecimiento de $f(x)$ se pueden hallar estudiando el signo de $f'(x)$, es decir, resolviendo la inecuación $f'(x) > 0$

[2]No siempre $f'(x) = 0$ implica la existencia de un extremo relativo

(f creciente) o $f'(x) < 0$ (f decreciente), podemos usar los comandos estudiados en los apartados anteriores con *Mathematica* para resolver este tipo de problemas.

Ejemplo. Calcula los máximos y mínimos relativos de $f(x) = x^4 + x^3 - 10x^2 + 8x$.

Comenzamos definiendo la función con *Mathematica*:

```
In[21]:= f[x_] := x^4 + x^3 - 10 x^2 + 8 x
```

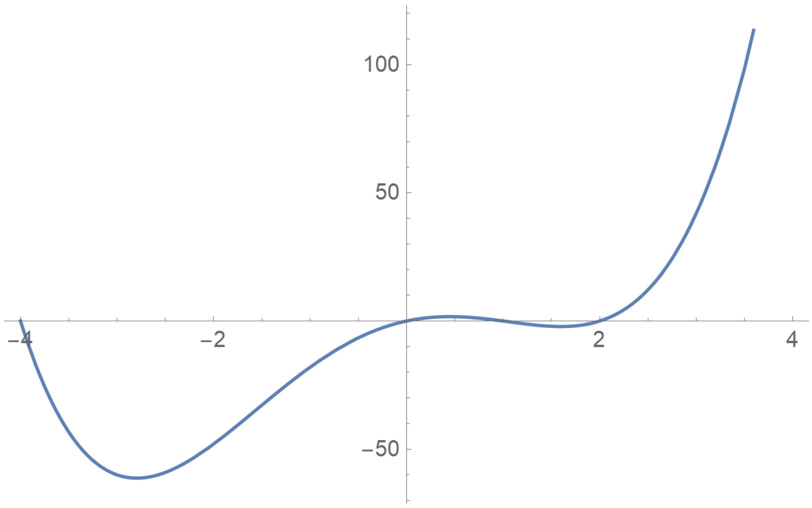
Para hallar los posibles extremos relativos resolvemos la ecuación $f'(x) = 0$:

```
In[22]:= Solve[f'[x] == 0, x]
         resuelve
Out[22]=
         {{x → -2.79...}, {x → 0.448...}, {x → 1.60...}}

In[23]:= N[%]
         valor numérico
Out[23]=
         {{x → -2.79496}, {x → 0.448119}, {x → 1.59684}}
```

A partir de una gráfica adecuada observamos que la función posee un máximo (en $x = 0.448119$) y dos mínimos relativos, que se corresponden con los otros dos valores obtenidos. En efecto,

```
In[24]:= Plot[f[x], {x, -4, 4}]
         representación gráfica
Out[24]=
```



Alternativamente, podríamos haber clasificado los extremos relativos evaluando el signo de la derivada segunda de f en dichos valores.

Por último, hallamos el valor de las ordenadas en los extremos relativos:

```
In[25]:= f[-2.79496]

Out[25]=
        -61.2871

In[26]:= f[0.448119]

Out[26]=
        1.70716

In[27]:= f[1.59684]

Out[27]=
        -2.1505
```

Y podemos concluir que $f(x)$ posee un máximo relativo, $(0.448119,\ 1.70716)$, y dos mínimos relativos, de coordenadas $(-2.79496,\ -61.2871)$ y $(1.59684,\ -2.1505)$, respectivamente.

El programa dispone de funciones específicas para encontrar los máximos y mínimos de una función, $f(x)$, previamente definida con *Mathematica*:

- **FindArgMin**[f[x], $\{x, x_0\}$]: Halla un mínimo de la función f cerca de x_0.
- **FindArgMax**[f[x], $\{x, x_0\}$]: Halla un máximo de la función f cerca de x_0.
- **FindMinimum**[f[x], $\{x, x_0\}$]: Devuelve un par de valores, donde el segundo es la abscisa del punto en el que la función alcanza un mínimo relativo cerca de x_0 y el primero es el valor de la función en dicho mínimo relativo.
- **FindMaximum**[f[x], $\{x, x_0\}$]: Devuelve un par de valores, donde el segundo es la abscisa del punto en el que la función alcanza un máximo relativo cerca de x_0 y el primero es el valor de la función en dicho máximo relativo.
- **Minimize**[f[x], x]: Devuelve, si la función tiene mínimo absoluto, un par de valores, siendo el segundo la abscisa del punto donde la función alcanza dicho mínimo absoluto y el primero es el valor del mínimo absoluto.
- **Maximize**[f[x], x]: Devuelve, si la función tiene máximo absoluto, un par de valores, siendo el segundo la abscisa del punto donde la función alcanza dicho máximo absoluto y el primero es el valor del máximo absoluto.

Aunque estos comandos pueden ser de gran utilidad para encontrar máximos o mínimos, no siempre funcionan bien, por lo que recomendamos su aplicación solo como apoyo o confirmación de los resultados que obtienes usando directamente las definiciones teóricas correspondientes.

Para finalizar la práctica, utilizaremos los comandos y procedimientos descritos en los apartados anteriores para determinar algunas propiedades de una función, con ayuda de *Mathematica*.

Ejemplo. Dada la función racional:

$$f(x) = \frac{3x^3 - x^2 - 6x + 3}{x^2 + x - 1}$$

- Represéntala gráficamente y halla su dominio.
- Calculas sus raíces exactas y/o aproximadas.
- Halla los puntos de corte de $f(x)$ con la recta $y = 3x$.
- Encuentra, usando el comando **Reduce**, los puntos en los que $f(x)$ se halla por debajo de $3x$.
- Determina las coordenadas de los extremos relativos de $f(x)$ e indica sus intervalos de crecimiento y decrecimiento.

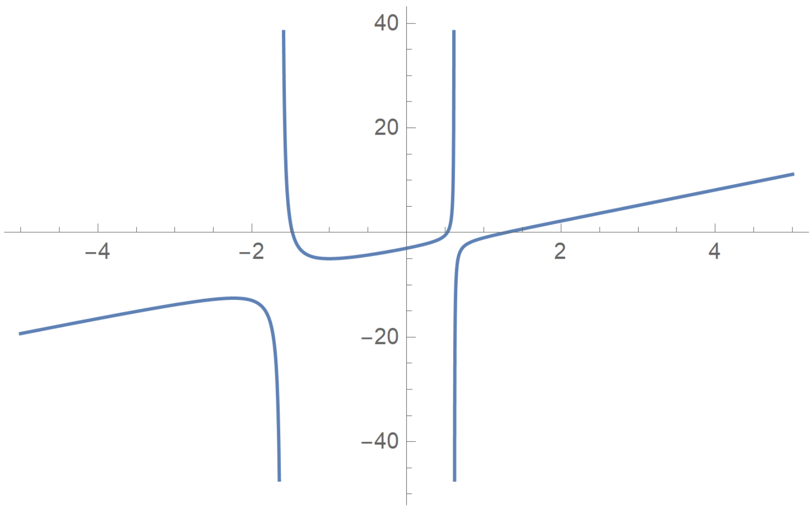
Comencemos por definir $f(x)$ con *Mathematica*:

```
In[28]:= f[x_] := (3 x^3 - x^2 - 6 x + 3) / (x^2 + x - 1)
```

A continuación, mostramos su represención gráfica:

```
In[29]:= Plot[f[x], {x, -5, 5}]
         representación gráfica
```

Out[29]=



Observa que se trata de una función racional, por lo que su dominio serán todos los número reales excepto las raíces del denominador, que podemos hallar mediante **Solve**:

```
In[30]:= Solve[x^2 + x - 1 == 0, x]
         resuelve

Out[30]=
         {{x → 1/2 (-1 - √5)}, {x → 1/2 (-1 + √5)}}
```

Por tanto, el dominio de $f(x)$ será

$$\mathbb{R} \sim \left\{ -\frac{\sqrt{5}}{2} - \frac{1}{2}, \frac{\sqrt{5}}{2} - \frac{1}{2} \right\}$$

Además,

$$x = -\frac{\sqrt{5}}{2} - \frac{1}{2} \quad \text{y} \quad x = \frac{\sqrt{5}}{2} - \frac{1}{2}$$

son asíntotas verticales de $f(x)$, como se aprecia en la gráfica.

Observamos también que $f(x)$ tiene tres raíces reales, ya que la gráfica corta al eje OX en tres puntos (nótese que aunque solo visualizamos la gráfica en el intervalo $[-5,\ 5]$ no puede haber más cortes puesto que la función del numerador es un polinomio de grado 3). Para hallar las raíces exactas debemos resolver la ecuación $f(x) = 0$. Así,

```
In[31]:= Solve[f[x] == 0, x]
         resuelve

Out[31]=
         {{x → -1.48...}, {x → 0.527...}, {x → 1.28...}}
```

que no devuelve el resultado exacto, a pesar de que las ecuaciones polinómicas de tercer grado (numerador) sí se pueden resolver de manera exacta. Para hallar una aproximación numérica de las raíces usaremos **NSolve** (también puedes aproximar el resultado anterior con **N**[%]):

```
In[32]:= NSolve[f[x] == 0, x]
         resolvedor numérico

Out[32]=
         {{x → -1.47785}, {x → 1.28432}, {x → 0.52686}}
```

Alternativamente, podemos aproximar las raíces mediante **FindRoot**, teniendo en cuenta que debemos ejecutarlo tantas veces como raíces buscamos, ya que solo devuelve una aproximación cada vez. Así, podemos hallar una aproximación a la raíz de $f(x)$, escogiendo como punto inicial $x_0 = 1$, mediante:

```
In[33]:= FindRoot[f[x] == 0, {x, 1}]
         encuentra raíz
Out[33]=
         {x → 1.28432}
```

Como ya hemos comentado, en funciones con más de una raíz, no es posible saber qué solución proporcionará el método cuando se usa la sintaxis de **FindRoot** con x_0. En efecto, observa que incluso, aunque el punto elegido esté cerca de una raíz, puede aproximar otra más alejada:

```
In[34]:= FindRoot[f[x] == 0, {x, -0.5}]
         encuentra raíz
Out[34]=
         {x → 1.28432}
```

Con el fin de forzar el cálculo aproximado de la solución que buscamos, conviene indicar un intervalo en el que está dicha raíz (y solo esa). Para ello, usaremos el segundo formato del comando **FindRoot**, ayudándonos previamente de una representación gráfica adecuada para seleccionar los extremos del intervalo que debemos introducir. Utilizando esta orden podemos aproximar las tres raíces de $f(x)$ mediante:

```
In[35]:= FindRoot[f[x] == 0, {x, -1.5, -1}]
         encuentra raíz
Out[35]=
         {x → -1.47785}

In[36]:= FindRoot[f[x] == 0, {x, 0, 0.57}]
         encuentra raíz
Out[36]=
         {x → 0.52686}

In[37]:= FindRoot[f[x] == 0, {x, 1, 2}]
         encuentra raíz
Out[37]=
         {x → 1.28432}
```

Respecto al siguiente apartado, las abscisas de los puntos de corte entre la función $f(x)$ y la recta $y = 3x$ se obtienen resolviendo la ecuación:

$$f(x) = 3x$$

Usando *Mathematica*:

```
In[38]:= Solve[f[x] == 3 x, x]
         resuelve
Out[38]=
```

$$\left\{\left\{x \to \frac{1}{8}\left(-3-\sqrt{57}\right)\right\}, \left\{x \to \frac{1}{8}\left(-3+\sqrt{57}\right)\right\}\right\}$$

Para averiguar en qué regiones el gráfico de $y = f(x)$ se halla por debajo de la recta $y = 3x$, basta utilizar la secuencia

```
In[39]:= Reduce[f[x] < 3 x, x]
         reduce
Out[39]=
```

$$x < \frac{1}{2}\left(-1-\sqrt{5}\right) \;||\; \frac{1}{8}\left(-3-\sqrt{57}\right) < x < \frac{1}{8}\left(-3+\sqrt{57}\right) \;||\; x > \frac{1}{2}\left(-1+\sqrt{5}\right)$$

Por lo tanto, la solución de la inecuación $f(x) < 3x$ es la unión de los intervalos:

$$\left]-\infty, -\frac{\sqrt{5}}{2}-\frac{1}{2}\right[\;\cup\; \left]-\frac{\sqrt{57}}{8}-\frac{3}{8}, \frac{\sqrt{57}}{8}-\frac{3}{8}\right[\;\cup\; \left]\frac{\sqrt{5}}{2}-\frac{1}{2}, +\infty\right[$$

Así pues, $f(x)$ se halla por debajo de $3x$ entre los puntos de intersección entre ambas, $x = -\frac{\sqrt{57}}{8} - \frac{3}{8}$ y $x = \frac{\sqrt{57}}{8} - \frac{3}{8}$, a la izquierda de su asíntota $x = -\frac{\sqrt{5}}{2} - \frac{1}{2}$ y a la derecha de su asíntota $x = \frac{\sqrt{5}}{2} - \frac{1}{2}$. Puedes comprobarlo mediante un gráfico adecuado, representando ambas funciones, la recta $y = 3x$ junto a $y = f(x)$, superpuestas.

Como aplicación del cálculo de derivadas podemos determinar las coordenadas de los puntos donde la función $f(x)$ alcanza el máximo y el mínimo relativo que se aprecian en la gráfica. Puesto que buscamos soluciones de la ecuación $f'(x) = 0$, para hallar las abscisas de estos puntos bastará con aplicar **Solve** para resolver la ecuación correspondiente:

```
In[40]:= Solve[f'[x] == 0, x]
         resuelve
Out[40]=
{{x → -1}, {x → -2.24...}, {x → 0.620... - 0.248... i}, {x → 0.620... + 0.248... i}}
```

que, tras aproximar el resultado, resulta

```
In[41]:= N[%]
         valor numérico
Out[41]=
{{x → -1.}, {x → -2.24059}, {x → 0.620294 - 0.248087 i}, {x → 0.620294 + 0.248087 i}}
```

De las dos[3] soluciones reales que obtenemos

$$x = -1 \qquad \text{y} \qquad x \approx -2.24059$$

resulta evidente que $x = -1$ es la que corresponde[4] al mínimo relativo y la otra al máximo relativo. El mínimo relativo se alcanza, pues, en el punto de coordenadas

$$(-1, f(-1)) = (-1, -5),$$

mientras que el máximo relativo se alcanza en (compruébalo)

$$(-2.24059, f(-2.24059)) \approx (-2.24059, -12.5427).$$

Por último, podemos hallar los intervalos de crecimiento de esta función, aplicando la secuencia **Reduce** para resolver la inecuación

$$\mathtt{f}'(\mathtt{x}) > 0$$

```
In[42]:= Reduce[f'[x] > 0, x]
         reduce
Out[42]=
```

$$x < -2.24\ldots \;||\; -1 < x < \frac{1}{2}\left(-1+\sqrt{5}\right) \;||\; x > \frac{1}{2}\left(-1+\sqrt{5}\right)$$

y obtendrás, después de aproximar la expresión,

```
In[43]:= N[%]
         valor numérico
Out[43]=
         x < -2.24059 || -1. < x < 0.618034 || x > 0.618034
```

Comprueba el resultado observando la gráfica. Análogamente, podrías hallar los intervalos de decrecimiento resolviendo la inecuación $f'(x) < 0$.

[3] $f'(x) = 0$ tiene cuatro soluciones pero las dos complejas no tienen interpretación gráfica.
[4] La prueba analítica de tal hecho es que $f'(-1) = 0 \quad \text{y} \quad f''(-1) > 0$.