Examen de Prácticas de Laboratorio de Cálculo

Grupo B: PL-10

Resolver los siguientes ejercicios utilizando `Matlab` *indicando* ***los comandos utilizados para resolverlos****,* ***las respuestas obtenidas*** *y* ***las representaciones gráficas que se piden****. Adjuntar un archivo PDF en la tarea del Campus Virtual creada para esta prueba, identificándolo con vuestro nombre y apellidos.*

Corrección

1. **Dada la función:**

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

a) **(1 Punto) Calcular los límites:** $\lim\limits_{x\to+\infty} f(x)$ **y** $\lim\limits_{x\to-\infty} f(x)$**.**

Se introduce la función en `Matlab`, se muestra por pantalla utilizando el comando `pretty` (para comprobar que se ha informado a `Matlab` de la función correcta, ¡no faltan paréntesis!) y se ejecutan los comandos para calcular los límites pedidos:

```
syms x
f(x)=x/(1+x^2);
pretty(f)
limit(f(x),x,+Inf)
limit(f(x),x,-Inf)
```

Se obtiene:

```
   x
------
 2
x  + 1


ans =

0


ans =

0
```

Esto es, el valor de cada uno de los límites es 0.

b) **(1 Punto) Obtener los intervalos de monotonía de la función (en dónde es creciente y decreciente), proporcionando los máximos y mínimos que posea (sean relativos o absolutos).**

Se calcula la derivada (primera) de la función y los números reales en los que se anula:

```
fprima(x)=diff(f,x);
double(solve(fprima))
```

```
ans =

    -1
     1
```

El dominio de la función es $\mathbb{R}$, puesto que el denominador que aparece en la definición de la función no se anula en ningún número real ($x^2 + 1 \geq 1$), por tanto, no se debe añadir ningún otro *valor conflictivo* para estudiar la monotonía de la función. Evaluamos la función en cada una de las tres regiones de los números reales que definen -1 y 1; se toman (por ejemplo) los valores -2, 0 y 2.

```
fprima([-2,0,2])
```

```
ans =

[-3/25, 1, -3/25]
```

Por todo lo obtenido, se deduce que la función es decreciente ($f'(x) \leq 0$) sobre los intervalos $(-\infty, -1) \cup (1, +\infty)$ y creciente ($f'(x) \geq 0$) sobre el intervalo $(-1, 1)$. Así, la función posee un mínimo en $x = -1$ y un máximo en $x = 1$.

c) **(1 Punto) Obtener los intervalos de curvatura de la función (en dónde es cóncava y convexa), proporcionando los puntos de inflexión que posea.**

Se calcula la derivada segunda de la función y los números reales (en formato numérico, puesto que eran presentados simbólicamente) en los que se anula:

```
fsegunda(x)=diff(f,x,2);
double(solve(fsegunda))
```

```
ans =

          0
     1.7321
    -1.7321
```

Se calculan las imágenes de la función en cada una de las cuatro regiones delimitadas por los valores obtenidos (por ejemplo, en -2, -1, 1 y 2).

```
fsegunda([-2,-1,1,2])
```

```
ans =

[-4/125, 1/2, -1/2, 4/125]
```

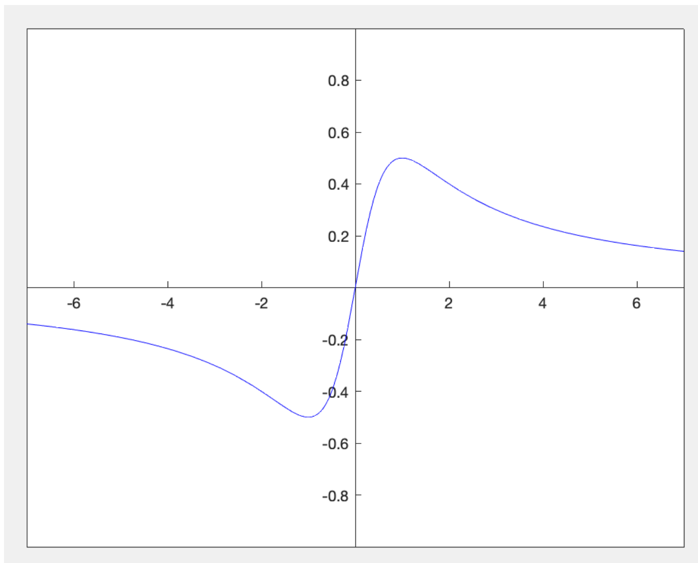
La función es cóncava (convexa para Set) en los intervalos $(-1.7321, 0) \cup (1.7321, +\infty)$ y convexa (cóncava para Set) en los intervalos $(-\infty, -1.7321) \cup (0, 1.7321)$. Por tanto, la función posee tres puntos de inflexión situados en $x = -1.7321$, $x = 0$ y $x = 1.7321$.

d) **(1 Punto) Representar la función en color azul sobre el intervalo $[-7, 7]$, con los valores de las imágenes comprendidas entre $[-1, 1]$ y verficar los resultados obtenidos analíticamente.**

Se ejecuta el comando `fplot` con los parámetros apropiados para representarla sobre el intervalo de abscisas exigido por el enunciado y en color azul. Además, se colocan los ejes de coordenadas de manera que se corten en el origen $(0, 0)$ y se define también el rango de las imágenes en el intervalo $[-1, 1]$:

```
fplot(f,[-7,7],'b')
ax=gca;
ax.XAxisLocation='Origin';
ax.YAxisLocation='Origin';
axis([-7,7,-1,1])
```

Se muestra el resultado obtenido:



Todo cuadra con los resultados obtenidos en los apartados anteriores.

2. *a*) **(1 Punto) Calcular el *Polinomio de Taylor* de grado 3 asociado a la función alrededor del punto $x = 0$.**

Se pide a `Matlab` el *Polinomio de Taylor* con los parámetros adecuados para hacerlo: alrededor del punto $x = 0$ y con grado $n = 3$, en general, tendría cuatro sumandos (de ahí 'order', 4). La salida se muestra con el comando `pretty`.

```
p3(x)=taylor(f,x,0,'order',4);
pretty(p3)

    3
- x   + x
```

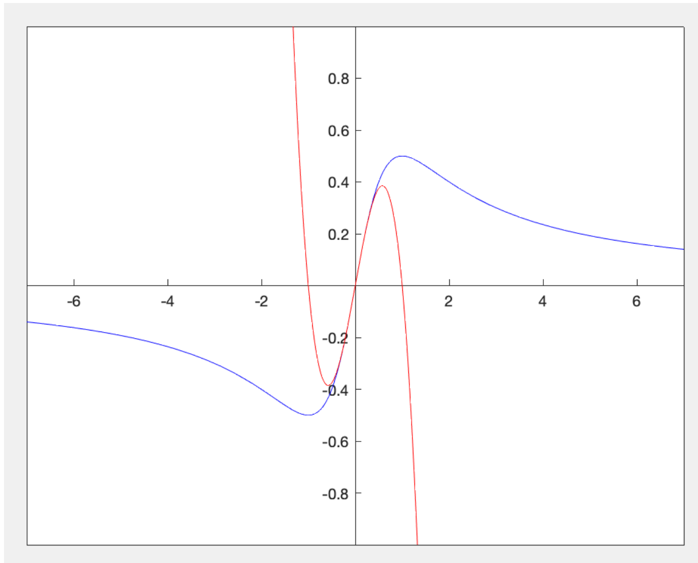
Esto es, el *Polnomio de Taylor* pedido es $p_3(x) = -x^3 + x$.

b) **(1 Punto) Representar el *Polinomio de Taylor* de grado 3 en color rojo junto con la función sobre el intervalo $[-7, 7]$.**

Se mantiene la representación de la función (lo exige el ejercicio) y se representa el polinomio en color rojo.

```
hold on
fplot(p3,[-7,7],'r')
```

Véase la figura que se obtiene:



c) **(1 Punto) Utilizar el polinomio anterior para aproximar el valor que toma la función en $x = 0.6$ y calcular el error cometido en la aproximación anterior. Expresar ambos valores numéricamente (no simbólicamente).**

Se realizan los cálculos directamente y se muestran los resultados.

```
aprox1=double(p3(0.6))
error1=double(f(0.6)-p3(0.6))
```

```
aprox =

    0.3840


error =

    0.0572
```

d) **(1 Punto) Calcular el *Polinomio de Taylor* de grado 5 asociado a la función alrededor del punto $x = 2$. Representar <u>exclusivamente</u> este *Polinomio de Taylor* de grado 5 junto con la función sobre el intervalo $[-1, 6]$.**

Se construye el *Polinomio de Taylor* pedido (llamado `p5(x)`):

```
p5(x)=taylor(f,x,1,'order',6);
fplot(p5,[-1,6],'m')
```

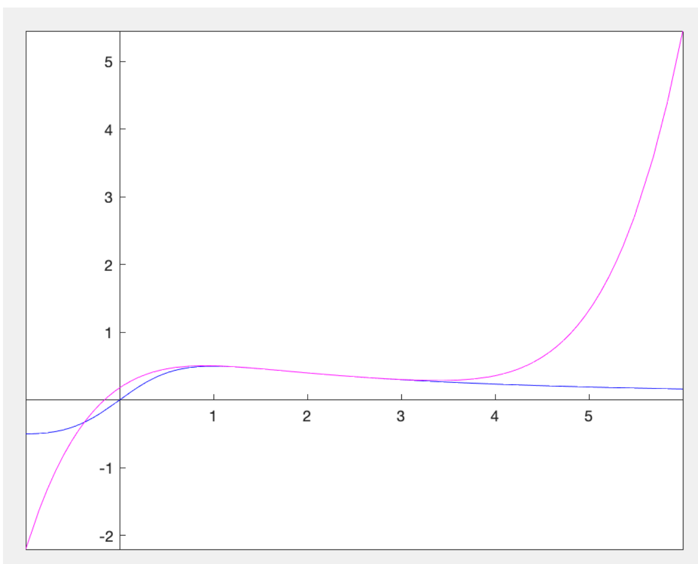
```
             2                          3                     4                         5
2 (x - 2)     3 x    7 (x - 2)     38 (x - 2)     117 (x - 2)     16
---------- - --- + ---------- - ----------- + ------------ + --
    125        25        625           3125            15625        25
```

El polinomio pedido es (por visualizarlo mejor):

$$p_5(x) = \frac{117}{15625}(x-2)^5 - \frac{38}{3125}(x-2)^4 + \frac{7}{625}(x-2)^3 + \frac{2}{125}(x-2)^2 - \frac{3}{25}x + \frac{16}{25}$$

Dado que la representación gráfica debe contener, **exclusivamente**, lo que se pide, antes de nada se elimina todo lo dibujado previamente. Se repinta la función, se recolocan los ejes en el origen y, finalmente, se representa `p5(x)` junto con la función. Se han escogido los colores azul para la función (como siempre) y magenta para el polinomio.

```
hold off
fplot(f,[-1,6],'b')
ax=gca;
ax.XAxisLocation='Origin';
ax.YAxisLocation='Origin';
hold on
fplot(p5,[-1,6],'m')
```



3. **(1 Punto) Calcular el área que encierra la función $f(x)$ sobre la parte positiva del eje de abscisas.**

Para calcular el área que encierra la función sobre la parte positiva del eje de abscisas $(0, +\infty)$, sólo es necesario el siguiente comando:

```
area1=int(f,x,0,+Inf)
```

Y se obtiene:

```
area1 =

   Inf
```

4. **(1 Punto) Calcular el área encerrada por el *Polinomio de Taylor* de grado 3 calculado en el apartado *a*) del ejercicio anterior con el eje de abscisas.**

   Observando la segunda representación gráfica, se detecta simetría respecto al origen de coordenadas (el polinomio $p_3(x) - x^3 + x$ es una función impar); claramente los cortes con el eje de abscisas se producen en $x = -1$, $x = 0$ y $x = 1$. Por esto, el área pedida es el doble de la que encierra sobre el intervalo $[0, 1]$. No obstante, resolviendo con `Matlab`:

```
solve(p3)
```

```
-1
 0
 1
```

   Y así:

```
area2=double(2*int(p3,x,0,1))
```

   Para obtener:

```
area2 =

    0.5000
```

La siguiente cuestión no se pide, es más, se va utilizar el comando `area` que no se incluía en los contenidos del examen (a pesar de que está en el guion cuarto de las Prácticas de Laboratorio). Es con carácter aclaratorio. Se introduce:

```
fplot(p3,[-1.5,1.5],'r')
ax=gca;
ax.XAxisLocation='Origin';
ax.YAxisLocation='Origin';
axis equal
hold on
absc=linspace(-1,1,50);
imag=p3(absc);
area(absc,imag)
```

Para obtener una representación gráfica del área pedida: