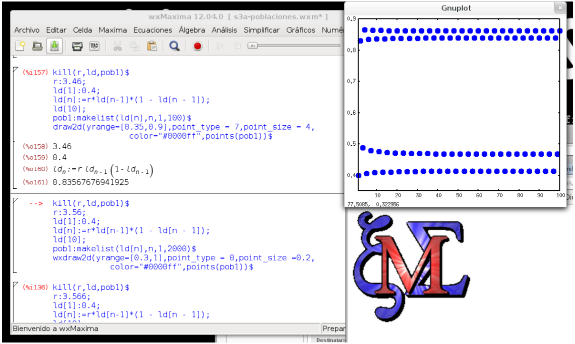
CÁLCULO DIFERENCIAL EN $\mathbb{R}^n$ con MAXIMA

Grado en Matemáticas

Renato Álvarez-Nodarse
Universidad de Sevilla

http://euler.us.es/~renato/clases.html

El software: MAXIMA



MAXIMA: programa simbólico/numérico con con licencia GNU/GPL accesibles por internet en

- http://maxima.sourceforge.net

Como intértprete de MAXIMA usaremos WXMAXIMA

Buscar el icono para comenzar

MAXIMA es un programa que funciona como una calculadora científica. Las operaciones aritméticas elementales son las habituales: $+$ suma, $-$ resta, $*$ multiplicación, $/$ división, ^ potencias:

```
(%i1) 2+2; 3-3; 2*3; 5/10; 3^3;
(%o1) 4
(%o2) 0
(%o3) 6
(%o4) 1/2
(%o5) 27
```

MAXIMA es un programa que funciona como una calculadora científica. Las operaciones aritméticas elementales son las habituales: $+$ suma, $-$ resta, $*$ multiplicación, $/$ división, ^ potencias:

```
(%i1) 2+2; 3-3; 2*3; 5/10; 3^3;
(%o1) 4
(%o2) 0
(%o3) 6
(%o4) 1/2
(%o5) 27
```

Dado que MAXIMA es un programa de cálculo *simbólico*, trabaja con variables definidas por letras:

```
(%i6) e;
(%o6) e
```

Las funciones (comandos) tienen el argumento (o los argumentos) que van entre paréntesis, como por ejemplo el comando `float(x)` que nos da como resultado el valor numérico de la variable x.

```
(%i7) float(e);
(%o7) e
(%i8) pi;
(%o8) pi
(%i9) float(pi);
(%o9) pi
```

por lo que su respuesta es la propia variable.
Para los valores numéricos de las constantes e o π MAXIMA usa

```
(%i10) %e;
(%o10) %e
(%i11) float(%e);
(%o11) 2.718281828459045
```

El orden de realización de las operaciones es el habitual. Así en la expresión

```
(%i18) (2+3^2)^3*(5+2^2);
(%o18) 11979
```

primero calcula las potencias dentro de cada paréntesis, luego las sumas, luego las potencias externas y finalmente la multiplicación.

El orden de realización de las operaciones es el habitual. Así en la expresión

```
(%i18) (2+3^2)^3*(5+2^2);
(%o18) 11979
```

primero calcula las potencias dentro de cada paréntesis, luego las sumas, luego las potencias externas y finalmente la multiplicación.

Para definir los valores de las variables se usan los dos puntos

```
(%i19) x:123; y:321; x*y; x/y; x-y;

(%i26) z=2;
(%i27) z;
(%i28) x;
(%i29) y;
(%i30) x*y;
(%i31) x+z;
```

También es posible definir funciones. Hay múltiples formas en función de lo queramos hacer.

```
(%i32) f(x):= x^2 -x + 1;
(%i33) f(%pi);
```

Nótese que a no ser que pidamos a MAXIMA que trabaje numéricamente, sigue usando cálculo simbólico

```
(%i35) float(f(%pi));
```

También es posible definir funciones. Hay múltiples formas en función de lo queramos hacer.

```
(%i32) f(x):= x^2 -x + 1;
(%i33) f(%pi);
```

Nótese que a no ser que pidamos a MAXIMA que trabaje numéricamente, sigue usando cálculo simbólico

```
(%i35) float(f(%pi));
```

Otro detalle interesante a tener en cuenta es que MAXIMA contiene una ayuda completa que puede ser invocada desde la línea de comandos. Para ello escribimos `??` delante del comando *desconocido*

```
(%i36) ??float;
```

Si no existe ningún comando con ese nombre la respuesta es `false`

```
(%i37) ??renato;
```

Otra de las opciones de MAXIMA es su potencia gráfica. Para hacer las gráficas MAXIMA usa GNUPLOT, un potente paquete gráfico GNU. El comando más sencillo de usar es `plot2d`.

```
(%i39) wxplot2d([f(x)], [x,-5,5]);
```

En conveniente asegurarse que variable `x` no tiene asignado ningún valor.

Otra de las opciones de MAXIMA es su potencia gráfica. Para hacer las gráficas MAXIMA usa GNUPLOT, un potente paquete gráfico GNU. El comando más sencillo de usar es `plot2d`.

```
(%i39) wxplot2d([f(x)], [x,-5,5]);
```

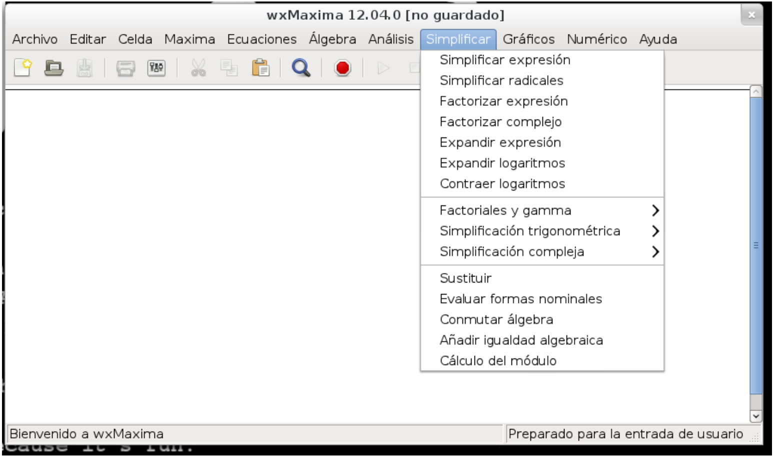
En conveniente asegurarse que variable `x` no tiene asignado ningún valor.
Podemos representar varias funciones

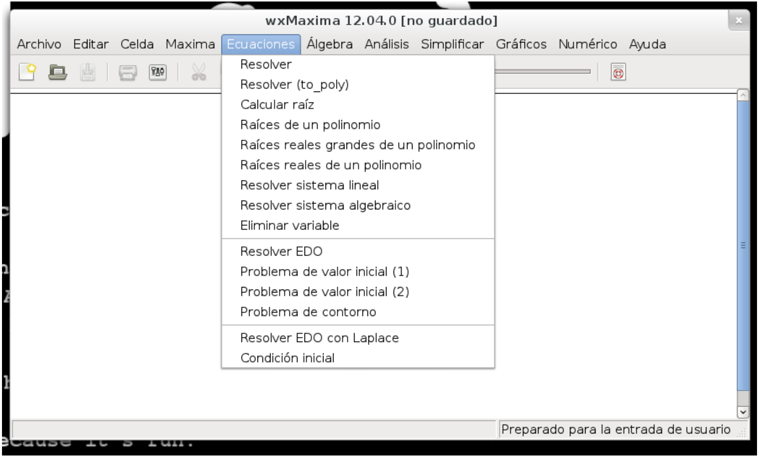
```
(%i41) plot2d([f(xx),2*sin(xx),atan(xx)], [xx,-2,2])$
```

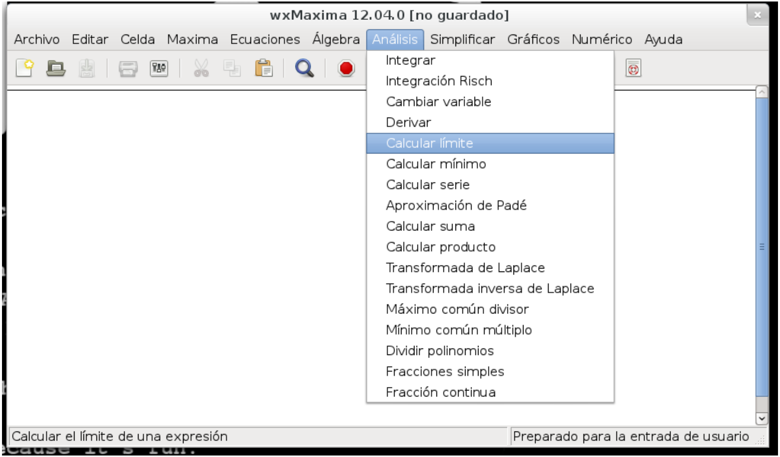
MAXIMA también dispone de una gran cantidad de funciones *elementales*, **exponencial**, **logaritmo**, **trigonométricas**, etc. Además las trata de forma simbólica.

```
(%i42) log(10);
(%i43) float(%);
(%i44) log(%e);
```

¿Qué más podemos hacer?


wxMaxima 12.04.0 [no guardado]
Archivo Editar Celda Maxima Ecuaciones Álgebra Análisis Simplificar Gráficos Numérico Ayuda
Simplificar expresión
Simplificar radicales
Factorizar expresión
Factorizar complejo
Expandir expresión
Expandir logaritmos
Contraer logaritmos
Factoriales y gamma
Simplificación trigonométrica
Simplificación compleja
Sustituir
Evaluar formas nominales
Conmutar álgebra
Añadir igualdad algebraica
Cálculo del módulo
Bienvenido a wxMaxima
Preparado para la entrada de usuario


wxMaxima 12.04.0 [no guardado]
Archivo Editar Celda Maxima Ecuaciones Álgebra Análisis Simplificar Gráficos Numérico Ayuda
Resolver
Resolver (to_poly)
Calcular raíz
Raíces de un polinomio
Raíces reales grandes de un polinomio
Raíces reales de un polinomio
Resolver sistema lineal
Resolver sistema algebraico
Eliminar variable
Resolver EDO
Problema de valor inicial (1)
Problema de valor inicial (2)
Problema de contorno
Resolver EDO con Laplace
Condición inicial
Preparado para la entrada de usuario


wxMaxima 12.04.0 [no guardado]
Archivo Editar Celda Maxima Ecuaciones Álgebra Análisis Simplificar Gráficos Numérico Ayuda
Integrar
Integración Risch
Cambiar variable
Derivar
Calcular límite
Calcular mínimo
Calcular serie
Aproximación de Padé
Calcular suma
Calcular producto
Transformada de Laplace
Transformada inversa de Laplace
Máximo común divisor
Mínimo común múltiplo
Dividir polinomios
Fracciones simples
Fracción continua
Calcular el límite de una expresión
Preparado para la entrada de usuario

¡Y un sinfín de cosas más!

En la web de la asignatura buscar el apartado: Software Libre de Matemáticas

http://euler.us.es/~renato/clases.html#maxima

Calculando derivadas. El comando para calcular derivadas es `diff(f(x),x,k)` donde `f(x)` es la función a la que le vamos a calcular la derivada `k`-ésima respecto a la variable `x`:

```
(%i56) diff(sin(x^2+2),x);
(%o56) 2*x*cos(x^2+2)
(%i57) diff(x^x, x,3);
(%o57) x^(x-1)*(log(x)+(x-1)/x)+x^x*(log(x)+1)^3
           +2*x^(x-1)*(log(x)+1)
(%i58) diff(sin(x)^x, x);
(%o58) sin(x)^x*(log(sin(x))+(x*cos(x))/sin(x))
```

Calculando derivadas. El comando para calcular derivadas es `diff(f(x),x,k)` donde `f(x)` es la función a la que le vamos a calcular la derivada `k`-ésima respecto a la variable `x`:

```
(%i56) diff(sin(x^2+2),x);
(%o56) 2*x*cos(x^2+2)
(%i57) diff(x^x, x,3);
(%o57) x^(x-1)*(log(x)+(x-1)/x)+x^x*(log(x)+1)^3
           +2*x^(x-1)*(log(x)+1)
(%i58) diff(sin(x)^x, x);
(%o58) sin(x)^x*(log(sin(x))+(x*cos(x))/sin(x))
```

Veamos que podemos hacer en $\mathbb{R}^n$.

MAXIMA también dibuja gráficas 3D con el comando `plot3d` cuya sintaxis es

```
plot3d([fun1,fun2,...],[var1,ini,fin],[var2,ini,fin],...)
```

```
(%i62) kill(f,x,y)$
       f(x,y):= sin(x) + cos(y);
       wxplot3d(f(x,y), [x,-5,5], [y,-5,5])$
```

MAXIMA también dibuja gráficas 3D con el comando `plot3d` cuya sintaxis es

```
plot3d([fun1,fun2,...],[var1,ini,fin],[var2,ini,fin],...)
```

```
(%i62) kill(f,x,y)$
       f(x,y):= sin(x) + cos(y);
       wxplot3d(f(x,y), [x,-5,5], [y,-5,5])$
```

Para que MAXIMA reconozca el directorio local de trabajo (lo que es conveniente para importar y exportar ficheros) es conveniente definir la variable `file_search_maxima`.

La forma más sencilla de definirlo es mediante la opción "Añadir a ruta" en la pestaña "Maxima" del menú del programa.

Cálculo diferencial de funciones de varias variables

Problema: Sea la función $f : \mathbb{R}^2 \mapsto \mathbb{R}$, $f(x, y) = x^2 y^3$. Representarla gráficamente. Calcular sus derivadas parciales hasta orden dos. Calcular la derivada direccional en el punto $a = (1, 3)$ según el vector $v = (2, -1)$.

Problema: Sea la función $f : \mathbb{R}^2 \mapsto \mathbb{R}$, $f(x, y) = x^2 y^3$.
Representarla gráficamente. Calcular sus derivadas parciales hasta orden dos. Calcular la derivada direccional en el punto $a = (1, 3)$ según el vector $v = (2, -1)$.
Primero cargamos el programa de representaciones gráficas `draw`

```
(%i1) load(draw)$
```

Luego definimos la función

```
(%i2) define(f(x,y),x^2*y^3);
(%o2) f(x,y):=x^2*y^3
```

Representamos a $f(x, y)$

```
(%i3) draw3d( xu_grid = 100, color = cyan,
explicit(f(x,y), x,-3,3,y,-3,3),
point_type=filled_circle, point_size = 3,
color = black, points([[0,0,0]]),
color = red, points([[1,2,f(1,2)]])  )$
```

Calculamos todas las **derivadas parciales** hasta orden dos

```
(%i4) diff(f(x,y),x); diff(f(x,y),y);
(%o4) 2*x*y^3
(%o5) 3*x^2*y^2
(%i6) diff(f(x,y),x,1,y,1); diff(f(x,y),y,1,x,1);
(%o6) 6*x*y^2
(%o7) 6*x*y^2
(%i8) diff(f(x,y),x,2); diff(f(x,y),y,2);
(%o8) 2*y^3
(%o9) 6*x^2*y
```

Calculamos todas las **derivadas parciales** hasta orden dos

```
(%i4) diff(f(x,y),x); diff(f(x,y),y);
(%o4) 2*x*y^3
(%o5) 3*x^2*y^2
(%i6) diff(f(x,y),x,1,y,1); diff(f(x,y),y,1,x,1);
(%o6) 6*x*y^2
(%o7) 6*x*y^2
(%i8) diff(f(x,y),x,2); diff(f(x,y),y,2);
(%o8) 2*y^3
(%o9) 6*x^2*y
```

Calculamos la **matriz jacobiana y la hessiana** con las órdenes

```
jacobian([f1,...,fn],[x1,...,xk])
hessian(f(x1,...xn),[x1,...,xk])
```

```
(%i10) jacobian([f(x,y)],[x,y]);
(%o10) matrix([ 2*x*y^3 , 3*x^2*y^2 ])
(%i11) hessian(f(x,y),[x,y]);
(%o11) matrix([ 2*y^3 , 6*x*y^2 ] , [ 6*x*y^2 , 6*x^2*y ])
```

Definimos la norma en $\mathbb{R}^2$

```
(%i12) define(norm2(vv),sqrt(sum(vv[k]^2,k,1,2)));
(%o12) norm2(vv):=sqrt(vv[2]^2+vv[1]^2)
```

Definimos la norma en $\mathbb{R}^2$

```
(%i12) define(norm2(vv),sqrt(sum(vv[k]^2,k,1,2)));
(%o12) norm2(vv):=sqrt(vv[2]^2+vv[1]^2)
```

Definimos el vector normalizado $u = v/\|v\|$

```
(%i13) v:[2,-1]$ vec:v/norm2(v);
(%o14) [2/sqrt(5),-1/sqrt(5)]
```

Definimos la norma en $\mathbb{R}^2$

```
(%i12) define(norm2(vv),sqrt(sum(vv[k]^2,k,1,2)));
(%o12) norm2(vv):=sqrt(vv[2]^2+vv[1]^2)
```

Definimos el vector normalizado $u = v/\|v\|$

```
(%i13) v:[2,-1]$ vec:v/norm2(v);
(%o14) [2/sqrt(5),-1/sqrt(5)]
```

Calculamos la matriz jacobiana de f

```
(%i15) gra:jacobian([f(x,y)],[x,y]);
(%o15) matrix([2*x*y^3,3*x^2*y^2])
```

Definimos la norma en $\mathbb{R}^2$

```
(%i12) define(norm2(vv),sqrt(sum(vv[k]^2,k,1,2)));
(%o12) norm2(vv):=sqrt(vv[2]^2+vv[1]^2)
```

Definimos el vector normalizado $u = v/\|v\|$

```
(%i13) v:[2,-1]$ vec:v/norm2(v);
(%o14) [2/sqrt(5),-1/sqrt(5)]
```

Calculamos la matriz jacobiana de f

```
(%i15) gra:jacobian([f(x,y)],[x,y]);
(%o15) matrix([2*x*y^3,3*x^2*y^2])
```

Calculamos la derivada direccional de f según u en el punto a; $D_u f(a) = \langle \nabla f(a), u \rangle$

```
(%i16) gra.vec;  ev(%,x=1,y=2);
(%o16) (4*x*y^3)/sqrt(5)-(3*x^2*y^2)/sqrt(5)
(%o17) 4*sqrt(5)
```

1. Calcula la derivada de f en a según la dirección v:

   1. $f(x,y) = xe^{xy}$; $a = (1,-1)$; $v = (1,1)$.
   2. $f(x,y,z) = (x/y)^z$; $a = (1,1,1)$; $v = (2,1,-1)$.
   3. $f(x,y) = (x^2+y^2)^x$; $a = (e,0)$; $v = (1,-1)$.

2. Calcula la matriz jacobiana de la función $f : \mathbb{R}^2 \mapsto \mathbb{R}^2$, y evalúala en $(x,y) = (\sqrt{\pi}, 1)$

$$f(x,y) = \left(\arctan\left(\frac{x+y}{1-xy}\right), \tan\left(\frac{x^2}{y}\right)\right);$$

3. Estudia el límite (rectas, parábolas, etc). ¿Es diferenciable en $(0,0)$?

$$\lim_{(x,y)\to(0,0)} \arctan\left(\frac{y^4+x^4}{y^2+x^2}\right)$$

Solución Ej. 3: Dibujando la función $g(x,y) = \arctan\left(\frac{y^4+x^4}{y^2+x^2}\right)$

```
draw3d( enhanced3d=false, xu_grid=100, color=green,
explicit(g(x,y), x,-3,3,y,-3,3),
point_type=filled_circle, point_size = 3,
color = black, points([[0,0,0]])  )$
```

Solución Ej. 3: Dibujando la función $g(x,y) = \arctan\left(\frac{y^4+x^4}{y^2+x^2}\right)$

```
draw3d( enhanced3d=false, xu_grid=100, color=green,
explicit(g(x,y), x,-3,3,y,-3,3),
point_type=filled_circle, point_size = 3,
color = black, points([[0,0,0]])  )$

draw3d (view = [111, 40], xu_grid=100, color=cyan,
explicit(g(x,y), x,-.3,.3,y,-.3,.3),
point_type=filled_circle, point_size = 3,
color = black, points([[0,0,0]])  )$
```

Solución Ej. 3: Dibujando la función $g(x,y) = \arctan\left(\frac{y^4+x^4}{y^2+x^2}\right)$

```
draw3d( enhanced3d=false, xu_grid=100, color=green,
explicit(g(x,y), x,-3,3,y,-3,3),
point_type=filled_circle, point_size = 3,
color = black, points([[0,0,0]])  )$

draw3d (view = [111, 40], xu_grid=100, color=cyan,
explicit(g(x,y), x,-.3,.3,y,-.3,.3),
point_type=filled_circle, point_size = 3,
color = black, points([[0,0,0]])  )$
```

"Investigando" gráficamente la diferenciabilidad

```
draw3d (view = [111, 40], xu_grid = 100, color = cyan,
explicit(g(x,y)/sqrt(x^2+y^2), x,-.3,.3,y,-.3,.3),
point_type=filled_circle, point_size = 3,
color = black,points([[0,0,0]])  )$
```

4. Investiga la diferenciabilidad de la función $f(x,y) = \dfrac{xy}{y^2+x^2}$

```
(%i77) f(x,y):=if x^2+y^2=0 then 0 else x*y/(y^2+x^2);
(%o77) f(x,y):=if x^2+y^2=0 then 0 else (x*y)/(y^2+x^2)
(%i78) f(0,0); f(2,1);
(%o78) 0
(%o79) 2/5
(%i80) draw3d ( xu_grid = 100, color = green,
explicit(f(x,y), x,-1,1,y,-1,1),
point_type=filled_circle, point_size = 3,
color = black, points([[0,0,0]])   )$
```

5. Idem para $f(x,y) = \dfrac{x^2y^2}{y^2+x^2}$ y $g(x,y) = \dfrac{y^4}{y^2+x^2}$.

Calcula además las derivadas parciales de orden 2.

Conviene reiniciar MAXIMA

Conviene reiniciar MAXIMA

```
(%i1) load(draw)$
```

Escojamos un ejemplo simple: $f(x,y) = 1 - (x^2 + y^2)$

```
 o(%i2) define(f(x,y),1-(x^2+y^2));
(%o2) f(x,y):=-y^2-x^2+1
(%i3) define(dxf(x,y),diff(f(x,y),x));
      define(dyf(x,y),diff(f(x,y),y));
(%o3) dxf(x,y):=-2*x
(%o4) dyf(x,y):=-2*y
```

Conviene reiniciar MAXIMA

```
(%i1) load(draw)$
```

Escojamos un ejemplo simple: $f(x, y) = 1 - (x^2 + y^2)$

```
 o(%i2) define(f(x,y),1-(x^2+y^2));
(%o2) f(x,y):=-y^2-x^2+1
(%i3) define(dxf(x,y),diff(f(x,y),x));
      define(dyf(x,y),diff(f(x,y),y));
(%o3) dxf(x,y):=-2*x
(%o4) dyf(x,y):=-2*y
```

Escribimos la ecuación del plano

```
(%i5) a:-1$ b:1$ xmin:a-1$ xmax:a+1$ ymin:b-1$ ymax:b+1$
      z=f(a,b)+dxf(a,b)*(x-a)+dyf(a,b)*(y-b);
define(plano(x,y), f(a,b)+dxf(a,b)*(x-a)+dyf(a,b)*(y-b));
(%o11) z=-2*(y-1)+2*(x+1)-1
(%o12) pla(x,y):=-2*(y-1)+2*(x+1)-1
```

Y dibujamos

```
(%i12) draw3d(color=blue,
explicit(f(x,y),x,xmin-1,xmax+1,y,ymin-1,ymax+1),
color=green, xu_grid = 100,
parametric_surface(x,y,pla(x,y),x,xmin,xmax,y,ymin,ymax),
point_type=filled_circle, point_size = 3,
color = black,points([[a,b,f(a,b)]]))$
```

Ejercicio: Encontrar los planos tangente en los puntos dados de las funciones siguientes y representarlos gráficamente:

1. $f(x,y) = \dfrac{x^2y^2}{y^2+x^2}$ en $a = (-1,1)$;
2. $f(x,y) = x^2/2 - y^2$ en $a = (2,1)$;
3. $f(x,y) = \arctan(x^2+y^2)$ en $a = (0,1)$;
4. $f(x,y) = (x^2+y^2)\sin(x^2+y^2)$ en $(0,0)$ y $(\sqrt{\pi},0)$.

Ejemplo: Calcular f_r y f_t, si $f(x,y) = x^2 + y^2$ y $x = r\mathrm{sen}\, t$, $y = r\cos t$.

Tenemos $f : \mathbb{R}^2 \mapsto \mathbb{R}$, $f(x,y) = x^2 + y^2$ y $g : \mathbb{R}^2 \mapsto \mathbb{R}^2$, $g(r,t) = [r\mathrm{sen}\, t, r\cos t] \Rightarrow h(r,t) = (f \circ g)(r,t) = r^2 \Rightarrow$ $Dh(r,t) = [2r, 0]$

Ejemplo: Calcular f_r y f_t, si $f(x,y) = x^2 + y^2$ y $x = r\text{sen}\, t$, $y = r\cos t$.

Tenemos $f : \mathbb{R}^2 \mapsto \mathbb{R}$, $f(x,y) = x^2 + y^2$ y $g : \mathbb{R}^2 \mapsto \mathbb{R}^2$, $g(r,t) = [r\text{sen}\, t, r\cos t] \Rightarrow h(r,t) = (f \circ g)(r,t) = r^2 \Rightarrow$ $Dh(r,t) = [2r, 0]$ Hagámoslo com MAXIMA.

Ejemplo: Calcular f_r y f_t, si $f(x,y) = x^2 + y^2$ y $x = r\text{sen}\,t$, $y = r\cos t$.

Tenemos $f : \mathbb{R}^2 \mapsto \mathbb{R}$, $f(x,y) = x^2 + y^2$ y $g : \mathbb{R}^2 \mapsto \mathbb{R}^2$, $g(r,t) = [r\text{sen}\,t, r\cos t] \Rightarrow h(r,t) = (f \circ g)(r,t) = r^2 \Rightarrow Dh(r,t) = [2r, 0]$ Hagámoslo com MAXIMA.

Definimos la función f, calculamos su matriz jacobiana $Df(a,x)$ y la evaluamos en la función interior $g(r,t)$

```
(%i1) define(f(x,y),x^2+y^2);
(%i2) jf:jacobian([f(x,y)],[x,y]);
(%i3) jfe:ev(jf,x=r*sin(t),y=r*cos(t));
```

Calculamos la matriz jacobiana de g, $Dg(r,t)$, y calculamos la matriz jacobiana de h, $Dh(r,t) = Df(x(r,t), y(r,t)) \cdot Dg(r,t)$

```
(%i4) jg:jacobian([r*sin(t),r*cos(t)],[r,t]);
(%i5) jfe.jg; trigsimp(%);
(%o7) matrix([2*r,0])
```

Calcular las derivadas parciales de las funciones compuestas:

a) $f(x,y) = \exp(xy)$; $x = u + v$, $y = u - v$.

b) $f(x,y) = x^2 - \log y$; $x = \log t$, $y = t^2$.

Calcular las derivadas parciales de las funciones compuestas:

a) $f(x,y) = \exp(xy)$; $x = u + v$, $y = u - v$.

b) $f(x,y) = x^2 - \log y$; $x = \log t$, $y = t^2$.

c) Sea $g : \mathbb{R}^3 \mapsto \mathbb{R}$ una función tal que $g(0,1,1) = 0$ y cuya matriz jacobiana en $a = (0,1,1)$ es $Dg(a) = (1,2,1)$. Sea $h : \mathbb{R} \mapsto \mathbb{R}^3$, $h(t) = (\arctan t^2, e^t, \cos t)$. Encontrar la matriz jacobiana de $h \circ g$ en $a = (0,1,1)$ y la de $g \circ h$ en $t = 0$.

$g:\mathbb{R}^3 \mapsto \mathbb{R}$ es tal que $g(0,1,1)=0$ y $Dg(0,1,1)=(1,2,1)$.
$h:\mathbb{R} \mapsto \mathbb{R}^3$ está definida por $h(t)=(\arctan t^2, e^t, \cos t)$.
Definimos la matriz jacobiana de $h(t)$

```
define(jach(t),jacobian([atan(t^2),exp(t),cos(t)],[t]));
```

y la matriz jacobiana de g en $a=(0,1,1)$

```
(%i9) jacg:[1,2,1];
```

Usamos que

$$D(h\circ g)(t)=Dh(y)\Big|_{y=g(0,1,1)}\cdot Dg(0,1,1), \quad D(g\circ h)(t)=Dg(y)\Big|_{y=h(0)}\cdot Dh(0)$$

```
(%i10) jachg:jach(0).jacg;
(%i11) jacgh:jacg.jach(0);
```

Utilizar la fórmula de Taylor (de orden 2) para desarrollar las siguientes funciones en los puntos indicados hasta orden dos:

$$f(x,y) = y^2/x^3 \qquad \text{en} \quad (1,-1)$$

$$g(x,y) = \log(x^2 + 2 * y^2) \qquad \text{en} \quad (1,1)$$

$$h(x,y) = (x^2 + 3y^2)\exp(1 - x^2 - y^2) \qquad \text{en} \quad (0,0)$$

$$l(x,y) = x^3 + y^2 + xy^2 \qquad \text{en} \quad (1,2)$$

$$q(x,y,z) = e^{a(x+y+z)} \qquad \text{en} \quad (0,0,0)$$

Solución usar el comando `taylor`

```
taylor(function , variables , vector, orden)
```

donde `funcion` es la función a desarrollar, `variables` son las variables (escritas como una lista), `vector` es el punto alrededor del cual hacemos el desarrollo hasta orden definido por `orden`.

Por ejemplo

```
(%i7) define(f(x,y),x*exp(y)+y*sin(2*x));
(%o7) f(x,y):=x*%e^y+sin(2*x)*y
(%i8) taylor (f(x,y) , [x, y], [0,0], 2)$  expand(%);
(%o9) 3*x*y+x
(%i10) taylor (f(x,y) , [x, y], [0,0], 5)$ expand(%);
(%o11) (x*y^4)/24+(x*y^3)/6+(x*y^2)/2-(4*x^3*y)/3+3*x*y+x
```

Hay otras variantes de este comando cuya salida no es exactamente el polinomio de Taylor del orden dado.

Ejercicio: Dibuja cada una de las funciones anteriores junto a su polinomio de Taylor de orden 2.

Ejemplo: Estudiar los extremos de $f(x, y) = (x + y)e^{-x^2-y^2}$.

Ejemplo: Estudiar los extremos de $f(x,y) = (x+y)e^{-x^2-y^2}$.

Comenzamos definiendo la función y encontrando los puntos críticos:

```
(%i1) define(f(x,y),(x+y)*exp(-x^2-y^2))
eq1:diff(f(x,y),x);
(%i3) eq2:diff(f(x,y),y);
(%i4)  sol:solve([eq1,eq2],[x,y]);
(%o4) [[x=1/2,y=1/2],[x=−1/2,y=−1/2]]
```

Un dibujo nos muestra que efectivamente tenemos máximo y mínimo locales (el segundo es un dibujo d econtorno):

```
(%i5) plot3d(f(x,y),[x,-1.1,1.1],[y,-1.1,1.1]);
(%i6) plot3d (f(x,y), [x,-2,2],[y,-2,2], [zlabel,""],
[mesh_lines_color,false], [elevation,0], [azimuth,0],
color_bar, [grid,80,80],[ztics,false],[color_bar_tics,1]);
```

Vamos a comprobarlo con el hessiano

```
(%i7) hes:hessian(f(x,y),[x,y]);
(%i8) h1:ev(hes,sol[1]);
(%o8) matrix([-3/sqrt(%e),-1/sqrt(%e)],
[-1/sqrt(%e),-3/sqrt(%e)])
(%i9) eigenvalues(h1);determinant(h1);
(%o9) [[-4/sqrt(%e),-2/sqrt(%e)],[1,1]]
(%o10) 8*%e^-1
```

De lo anterior se sigue que en $(1/2, 1/2)$ hay un máximo ya que la matriz hessiana es definida negativa. Finalmente repetimos el cálculo para el punto $(-1/2, -1/2)$

```
(%i11) h2:ev(hes,sol[2]);
(%o11) matrix([3/sqrt(%e),1/sqrt(%e)],
[1/sqrt(%e),3/sqrt(%e)])
(%i12) eigenvalues(h2);determinant(h2);
(%o12) [[2/sqrt(%e),4/sqrt(%e)],[1,1]]
(%o13) 8*%e^-1
```

Si repetimos lo anterior para la función $f(x,y) = (xy)e^{x^2+y^2}$ es fácil comprobar que tiene un punto silla en $(0,0)$.

```
(%i1)  remfunction(f)$ define(f(x,y),(x*y)*exp(x^2+y^2));
(%o2) f(x,y):=x*y*%e^(y^2+x^2)
(%i3) eq1:diff(f(x,y),x); eq2:diff(f(x,y),y);
(%i5) sol:solve([eq1,eq2],[x,y]);
(%i6) plot3d(f(x,y),[x,-1.1,1.1],[y,-1.1,1.1]);
(%i7) wxplot3d (f(x,y),[x,-1,1],[y,-1,1], [zlabel,""],
[mesh_lines_color,false], [elevation,0], [azimuth,0],
color_bar,[grid,80,80],[ztics,false],[color_bar_tics,5]);
(%i8) hessian(f(x,y),[x,y]);
(%i9) h:ev(%,sol[1]);
(%o9) matrix([0,1],[1,0])
(%i10) eigenvalues(h);
(%o10) [[−1,1],[1,1]]
```

- Como ejercicio calcula los extremos de las funciones

$$f(x,y) = (x^2 + y^2)\exp(x^2 - y^2)$$

$$g(x,y) = (x^2 + 3y^2)\exp(1 - x^2 - y^2)$$

$$h(x,y) = x^3y^3 - (x^4 + y^4) + xy$$

$$l(x,y) = (1 + e^y)\cos(x) - ye^y$$

Para el caso de **extremos condicionados** el procedimiento es análogo pero se usa la función de Lagrange. Determinar el tipo de punto crítico requiere un poco más de cuidado. Como ejemplo resolveremos el siguiente problema:

•Encontrar los puntos de mayor y menor distancia al origen de la curva $x^2 + 3xy + y^2 - 4 = 0$.

La función a minimizar/maximizar es $f(x, y) = x^2 + y^2$

Para el caso de **extremos condicionados** el procedimiento es análogo pero se usa la función de Lagrange. Determinar el tipo de punto crítico requiere un poco más de cuidado. Como ejemplo resolveremos el siguiente problema:

•Encontrar los puntos de mayor y menor distancia al origen de la curva $x^2 + 3xy + y^2 - 4 = 0$.

La función a minimizar/maximizar es $f(x, y) = x^2 + y^2$

Hay dos maneras:

1. Usando la ecuación de la curva para despejar una de las variables, digamos y y sustitirla en f lo que nos conduce a un problema de extremo de una variable independiente.

2. Usando el método de los multiplicadores de Lagrange. Para este ejemplo la función de Lagrange es:

$$L(x, y, \lambda) = x^2 + y^2 + \lambda(x^2 + 3xy + y^2 - 4).$$

Para el caso de **extremos condicionados** el procedimiento es análogo pero se usa la función de Lagrange. Determinar el tipo de punto crítico requiere un poco más de cuidado. Como ejemplo resolveremos el siguiente problema:

•Encontrar los puntos de mayor y menor distancia al origen de la curva $x^2 + 3xy + y^2 - 4 = 0$.

La función a minimizar/maximizar es $f(x, y) = x^2 + y^2$

Hay dos maneras:

1. Usando la ecuación de la curva para despejar una de las variables, digamos y y sustitirla en f lo que nos conduce a un problema de extremo de una variable independiente.

2. Usando el método de los multiplicadores de Lagrange. Para este ejemplo la función de Lagrange es:

$$L(x, y, \lambda) = x^2 + y^2 + \lambda(x^2 + 3xy + y^2 - 4).$$

Vamos a resolverlo con Maxima usando ambos métodos.

- Encontrar los extremos de la función

$f(x, y) = x^2 + y^2 - x - y + 1$ definida sobre la región $x^2 + y^2 \leq 2$.

En este caso hay dividir el problema en dos: El primero un problema de extremos libres (interior de la región) y el segundo de extremos condicionados con la condición $x^2 + y^2 = 2$.

Hagámoslo con MAXIMA.

•Encontrar los extremos de la función

$f(x, y) = x^2 + y^2 - x - y + 1$ definida sobre la región $x^2 + y^2 \leq 2$.

En este caso hay dividir el problema en dos: El primero un problema de extremos libres (interior de la región) y el segundo de extremos condicionados con la condición $x^2 + y^2 = 2$.

Hagámoslo con MAXIMA.

•Ejercicio. Para la misma función del ejemplo anterior encontrar los extremos cuando la región es $x^2 + y^2 \leq 2$, $y \leq 1$.

•Como ejercicio resolver los siguientes problemas:

1. Encuentra los puntos de mayor y menor distancia al origen de la curva $y^2 - 8y + x^2 - 6x - 75 = 0$.

2. Calcular los extremos de $f(x, y) = x + y$ sobre la elipse $x^2 + 4y^2 = 1$.

3. Encontrar los puntos más cercanos y alejados a $(0, 0, 3)$ de la superficie $x^2 + 2y^2 + 4z^2 - 9 = 0$.

4. Estudiar los extremos de la función $f(x, y, z) = x^2 + y^2 + z^2 - 2x + 4y - 3$ en la región $x^2 + y^2 + z^2 \leq 9$, $x \geq 0$.