PRÁCTICA 4:
Cálculo Diferencial de Funciones de varias variables

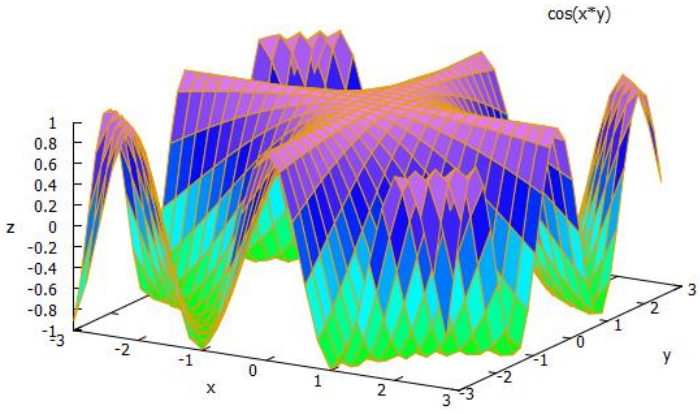
Gráficos en 3D

plot3d

Con *Maxima* se pueden representar funciones de dos variables de forma similar a como hemos representado funciones de una. La principal diferencia es que vamos a utilizar el comando `plot3d` en lugar de `plot2d`, pero igual que en el caso anterior, son obligatorios la función o funciones y el dominio que tiene que ser de la forma $[a,b]\times[c,d]$.

`plot3d(f(x,y),[x,a,b],[y,c,d])` gráfica de $f(x,y)$ en $[a,b]\times[c,d]$

```
(%i1)   plot3d(cos(x*y), [x,-3,3], [y,-3,3])$
```



⚠ **Observación** La gráfica que acabas de obtener se puede girar sin más que pinchar con el ratón sobre ella y deslizar el cursor.

Al comando `plot3d` se accede a través del menú **Gráficos→Gráficos 3D** o del botón Gráficos 3D. Después de esto aparece la ventana de la Figura con varios campos para rellenar:

Figura Gráficos 3D

a) Expresión. La función o funciones que vayamos a dibujar.
b) Variable. Hay dos campos para indicar el dominio de las dos variables.
c) Cuadrícula. Indica cuántas valores se toman de cada variable para representar la función. Cuanto mayor sea, más suave será la representación a costa de aumentar la cantidad de cálculos.
d) Formato. Igual que en `plot2d`, permite escoger qué programa se utiliza para representar la función. Se puede girar la gráfica en todos ellos salvo si escoges "en línea".
e) Opciones. Las comentamos a continuación.
f) Gráfico al archivo. Permite elegir un fichero donde se guardará la gráfica.

Quizá la mejor manera de ver el efecto de las opciones es repetir el dibujo anterior. La primera de ellas es "`set pm3d at b`" que dibuja la superficie usando una malla y en la base añade curvas de nivel (como si estuviéramos mirando la gráfica desde arriba)

La segunda hace varias cosas, "`set pm3d at s`" nos dibuja la superficie coloreada, "`unset surf` elimina la malla y "`unset colorbox`" elimina la barra que teníamos en la derecha con la explicación de los colores y su relación con la altura (el valor de la función)

La tercera, "`set pm3d map`", nos dibuja un mapa de curvas de nivel con alguna ayuda adicional dada por el color

La cuarta, "`set hidden3d`", sólo muestra la parte de la superficie que sería visible desde nuestro punto de vista. En otras palabras, hace la superficie sólida y no transparente.

Ten en cuenta que las opciones que tiene *Gnuplot* son casi infinitas y sólo estamos comentando algunas.

Gráficos con draw

Como ya vimos en funciones de una variable, otra opción para realizar gráficas es utilizar el módulo draw. Acuérdate de cargarlo

```
(%i2)    load(draw)$
0 errors, 0 warnings
```

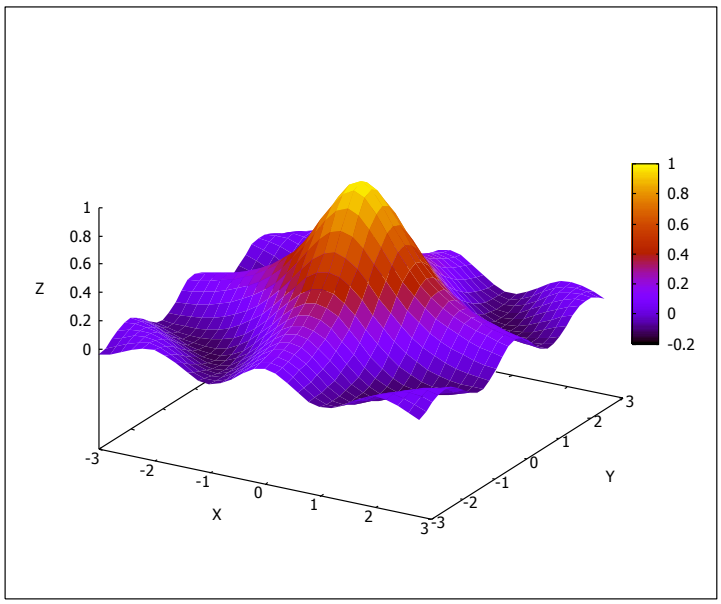
Y ahora ya podemos usar el comando draw3d

`draw3d(`*`opciones, objeto gráfico,...`*`)`	dibuja gráfico tres dimensional

De nuevo son muchas las opciones disponibles, y los tipos de objetos que pueden dibujarse, pero nosotros nos centraremos en explicit e implicit.

explicit: Usaremos explicit(*f* (*x*,*y*),*x*,*a*,*b*,*y*,*c*,*d*) para dibujar *f*(*x*,*y*) en [*a*,*b*]x[*c*,*d*].

```
(%i3)    draw3d(enhanced3d=true,surface_hide=true,
         explicit((1+x^2+y^2)^(-1)*cos(x*y),x,-3,3,y,-3,3))$
```



En el gráfico anterior hemos usado las opciones:

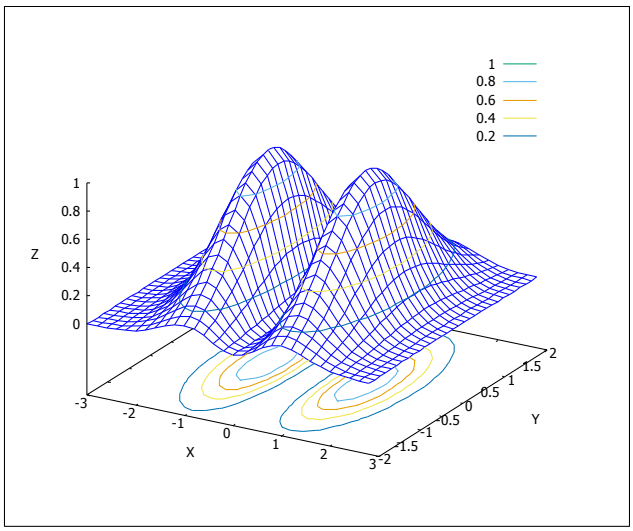
enhanced3d: si vale `true` se colorean las superficies en gráficos tridimensionales.

surface_hide: cuando vale `true` no se dibuja la parte no visible de las superficies.

También es interesante la opción **contour**.

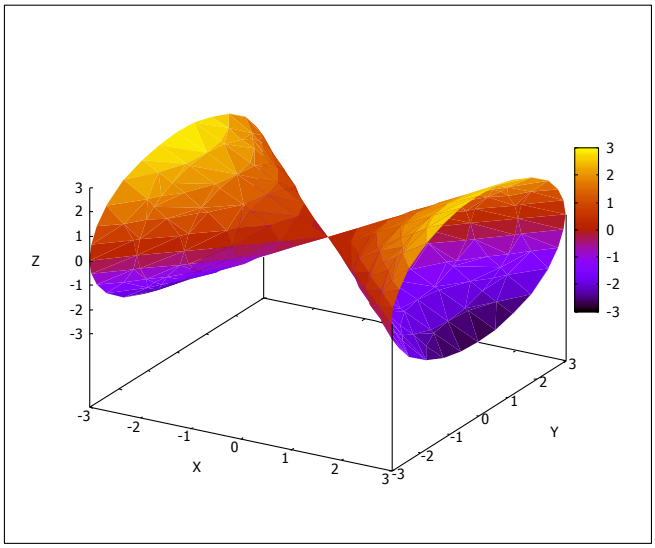
contour nos permite dibujar o no las curvas de nivel de una superficie. Por defecto no se muestran (`none`) pero también podemos dibujarlas sobre el plano XY con `base`
sobre la propia superficie con `surface`
las dos posibilidades anteriores se pueden usar a la vez con `both`
o, simplemente, podemos dibujar las curvas de nivel vistas desde arriba con `map`

```
(%i6)    draw3d(surface_hide=true,contour=both,
         explicit(x^2*exp(1-x^2-0.5*y^2),x,-3,3,y,-2,2))$
```



implicit: nos permite dibujar el lugar de los puntos que verifican una ecuación en tres dimensiones. Además de la ecuación debemos indicar los intervalos dónde pueden tomar valores las variables.

```
(%i9)    draw3d(enhanced3d=true,surface_hide=true,
         implicit(x^2-y^2=z^2,x,-3,3,y,-3,3,z,-3,3))$
```



Límites y Continuidad

En los siguientes ejemplos veremos cómo usar el análisis gráfico para comprobar la continuidad de funciones de dos variables.

Comenzaremos utilizando el comando **plot3d** o **draw3d** para una primera visualización de su gráfica.

Observaremos las cercanías del origen. Acuérdate que puedes girar la gráfica para verla mejor. Esto nos da una primera impresión sobre la continuidad de la función en dicho punto.

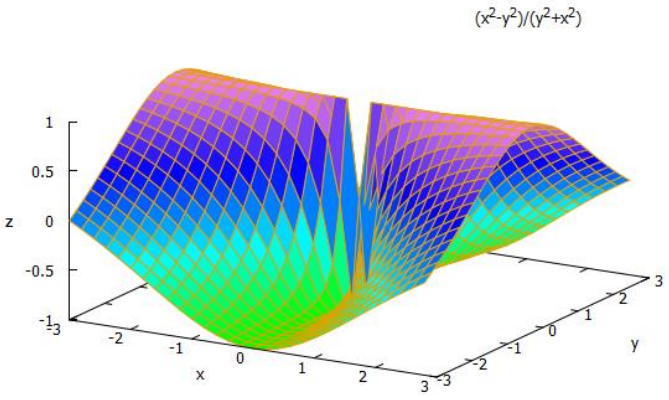
A continuación, mediante el comando **contour_plot** o la opción **contour** de **draw3d** representamos las curvas de nivel de la función. Recuerda que en una función continua las curvas de nivel nunca se cruzan.

Ejemplo. Estudiar la continuidad en el origen de la función f(x,y)=(x^2-y^2)/(x^2+y^2); f(0,0)=0.

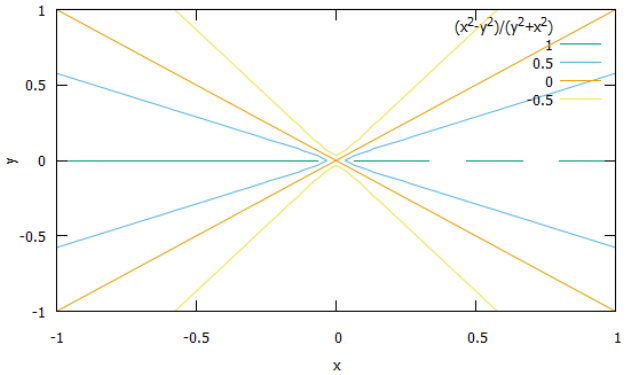
```
(%i10)  f(x,y):=(x^2-y^2)/(x^2+y^2);
```

$$(\%o10)\quad f(x,y):=\frac{x^2-y^2}{x^2+y^2}$$

```
(%i12)  plot3d(f(x,y),[x,-3,3],[y,-3,3])$
```




```
(%i14)  contour_plot(f(x,y),[x,-1,1],[y,-1,1])$
```



Derivadas parciales de una función

El primer objetivo será aprender a calcular derivadas parciales, gradientes, hessianos... de funciones de varias variables, y aplicar todo ello al cálculo de extremos relativos y condicionados.

Recordemos que el comando `diff` nos servía para calcular derivadas de funciones de una variable. Pues bien, también nos va a servir para calcular derivadas parciales de campos escalares de varias variables. Esto es

`diff(f(`$x_1, x_2 \cdots$`),`$x_1, n_1, x_2, n_2, \cdots$`)`	derivada parcial de la función $f(x_1, x_2, \cdots)$ respecto a x_1, n_1 veces, respecto a x_2, n_2 veces,...

Calculemos como ejemplo $\frac{\partial^3 f}{\partial^2 x \partial y}$ y $\frac{\partial^2 f}{\partial x \partial y}$, donde $f(x,y) = \text{sen}(x)\cos(y^2)$:

```
(%i15)  f(x,y):=sin(x)*cos(y^2)$
(%i17)  diff(f(x,y),x,2,y,1);
        diff(f(x,y),x,1,y,1);
(%o16)  2 sin(x) y sin(y^2)
(%o17)  -2 cos(x) y sin(y^2)
```

Con las derivadas parciales de primer orden construimos el vector gradiente ($\nabla f(a)$) de un campo escalar en un punto, y la matriz jacobiana ($Jf(a)$) de un campo vectorial en un punto. Para ambos cálculos utilizaremos el mismo comando: `jacobian(f,x)`. Por ejemplo:

jacobian

```
(%i19)  jacobian([f(x,y)],[x,y]);
(%o19)  [cos(x) cos(y^2)  -2 sin(x) y sin(y^2)]
```

Ahora bien, si queremos calcular la matriz hessiana de un campo escalar, utilizaremos el comando `hessian` de la forma siguiente:

hessian

```
(%i18)  hessian(f(x,y),[x,y]);
0 errors, 0 warnings
(%o18)  [ -sin(x) cos(y^2)          -2 cos(x) y sin(y^2)          ]
        [                                                         ]
        [-2 cos(x) y sin(y^2)  -2 sin(x) sin(y^2) - 4 sin(x) y^2 cos(y^2) ]
```

Recuerda que las derivadas cruzadas de segundo orden coinciden para funciones "suficientemente buenas". Además, hay que comentar que, si estamos calculando el gradiente (o matriz jacobiana) de un campo escalar, mientras en el comando `hessian` la función se escribe $f(x,y)$, en el comando `jacobian` la función hay que escribirla entre corchetes. La razón es que este comando sirve tanto para campos escalares como vectoriales.

Extremos relativos

El método para encontrar extremos relativos de funciones de varias variables suficientemente derivables consiste en buscar primero los puntos críticos, es decir, puntos donde se anula el gradiente, y después estudiar la matriz hessiana en esos puntos. Los resultados que conocemos nos aseguran que todos los puntos extremos de una función están entre los puntos críticos, con lo que una vez calculados estos nos dedicaremos a estudiar la matriz hessiana en ellos.

Para comprobar qué tipo de puntos son los puntos críticos definimos la matriz hessiana, H.
Si |H|>0 y $\partial_{x,x} f[x, y]$<0 en ese punto crítico hay un máximo.
Si |H|>0 y $\partial_{x,x} f[x, y]$>0 en ese punto crítico hay un mínimo.
Si |H|<0 en es punto crítico no hay máximo ni mínimo.
Si |H|=0 puede existir o no extremo, en este caso haría falta realizar un estudio más detallado.

Ejemplo Calculemos los extremos relativos de la función $f(x,y) = x^3+3xy^2-15x-12y$. Primero definimos la función, y calculamos su gradiente y su matriz hessiana.

```
(%i20)  f(x,y):=x^3+3*x*y^2-15*x-12*y$
(%i21)  j:jacobian([f(x,y)],[x,y]);
(j)     matrix([3*y^2+3*x^2-15,6*x*y-12])
(%i22)  h:hessian(f(x,y),[x,y]);
(h)     matrix([6*x,6*y],[6*y,6*x])
```

Para calcular los puntos críticos podemos irnos a **Ecuaciones→Resolver sistema algebraico** e ir rellenando los datos que nos van pidiendo. De esta forma entra en acción el comando `algsys` para resolver sistemas de ecuaciones. A la hora de llamar a las ecuaciones, lo haremos de la siguiente forma:

```
(%i23)  realonly:true$
(%i24)  pcrit:algsys([j[1,1],j[1,2]],[x,y]);
(pcrit) [[x=2,y=1],[x=1,y=2],[x=-1,y=-2],[x=-2,y=-1]]
```

donde j[1,1] es el primer elemento del gradiente, es decir $\frac{\partial f}{\partial x}$ y j[1,2] es el segundo, es decir $\frac{\partial f}{\partial y}$. Observad que el resultado es una lista de listas de puntos; lista que además hemos llamado `pcrit` para luego poder acudir a ella a la hora de evaluar la matriz hessiana en los puntos críticos obtenidos.

Observad también que cómo únicamente nos interesan las raíces reales, hemos incluido la orden **realonly.** Calcularíamos ahora la matriz hessiana en cada uno de los puntos críticos para clasificarlos. Por ejemplo en el primero obtenemos:

```
(%i25)  pcrit[1];
(%o25)  [x=2,y=1]
(%i26)  h,pcrit[1];
(%o26)  matrix([12,6],[6,12])
(%i27)  determinant(%);
(%o27)  108
```

Por tanto, en el punto (2,1) la función *f* presenta un mínimo relativo. Así mismo, en (-2,-1) hay un máximo relativo:

```
(%i28)  pcrit[4];
(%o28)  [x=-2,y=-1]
(%i29)  h,pcrit[4];
(%o29)  matrix([-12,-6],[-6,-12])
(%i30)  determinant(%);
(%o30)  108
```

Mientras que en los puntos (1,2) y (-1,-2) la función *f* presenta puntos de silla.

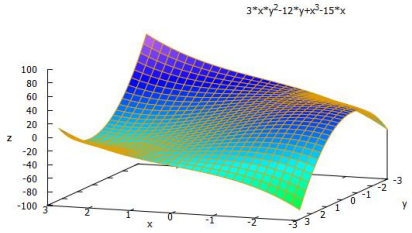
```
(%i31)  pcrit[2];
(%o31)  [x=1,y=2]

(%i32)  h,pcrit[2];
(%o32)  matrix([6,12],[12,6])

(%i33)  determinant(%);
(%o33)  -108
```

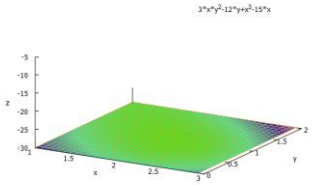
La gráfica de la función nos da también una idea de los extremos:

```
(%i34)  plot3d(f(x,y), [x,-3,3], [y,-3,3])$
```

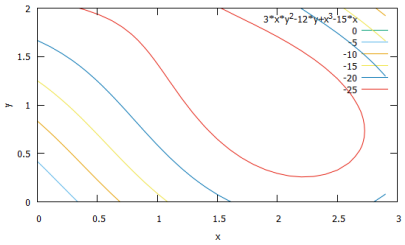


Y las curvas de nivel, centradas en el punto crítico (2,1), nos muestran claramente la presencia en ese punto de un mínimo:

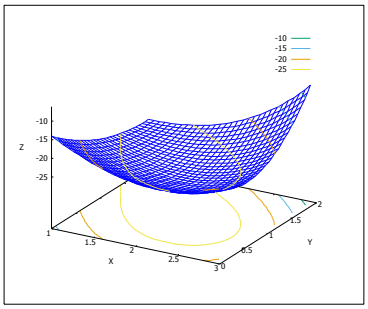
```
(%i40)  plot3d(f(x,y), [x,1,3], [y,0,2], [plot_format,gnuplot],
         [gnuplot_postamble, "set pm3d at b"])$
```



```
(%i36)  contour_plot(f(x,y),[x,0,3],[y,0,2])$
```




```
(%i41)  draw3d(surface_hide=true,contour=both,
         explicit(f(x,y),x,1,3,y,0,2))$
```



Prueba a hacer lo mismo con el resto de los puntos críticos, y prueba también distintas gráficas de la función centradas en dichos puntos.

Ejemplo Calculemos los extremos relativos de $g(x,y) = (x^2+3y^2)\,e^{1-x^2-y^2}$. Comenzamos definiendo g:

```
(%i1)  kill(all)$
(%i1)  g(x,y):=(x^2+3*y^2)*exp(1-x^2-y^2)$
```

Calculamos ahora los puntos críticos como en el ejemplo anterior y tenemos:

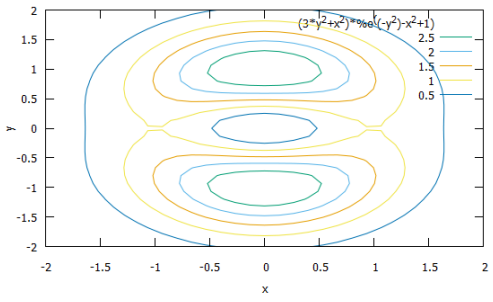
```
(%i2)  j:jacobian([g(x,y)],[x,y]);
(j)    [2 x %e^(-y^2-x^2+1)-2 x (3 y^2+x^2) %e^(-y^2-x^2+1)  6 y %e^(-y^2-x^2+1)-2 y (3 y^2+x^2) %e^(-y^2-x^2+1)]
(%i3)  realonly:true$
(%i4)  pcrit:algsys([j[1,1],j[1,2]],[x,y]);
(pcrit) [[x=0,y=0],[x=-1,y=0],[x=1,y=0],[x=0,y=-1],[x=0,y=1]]
```

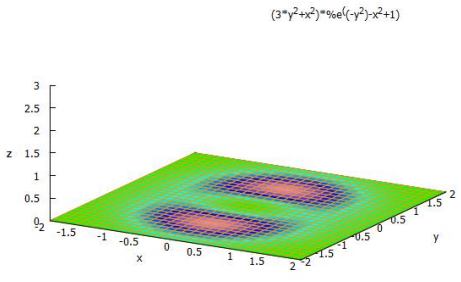
y vamos a calcular la matriz hessiana y a clasificarla en cada uno de los puntos críticos encontrados:

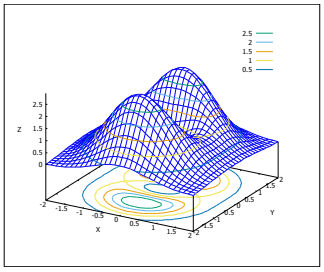
```
(%i5)  h:hessian(g(x,y),[x,y])$
(%i7)  h,pcrit[1];
       determinant(%);
(%o6)  matrix([2 %e, 0], [0, 6 %e])
(%o7)  12 %e^2
```

Así, en el punto $(0,0)$ tenemos un mínimo y, razonando de la misma forma en los otros cuatro puntos, dos puntos de máximo en $(0,-1)$ y $(0,1)$, así como dos puntos de silla en $(-1,0)$ y $(1,0)$. Podemos observar esto en la gráfica de la función, mejor aún si a la vez dibujamos las curvas de nivel:

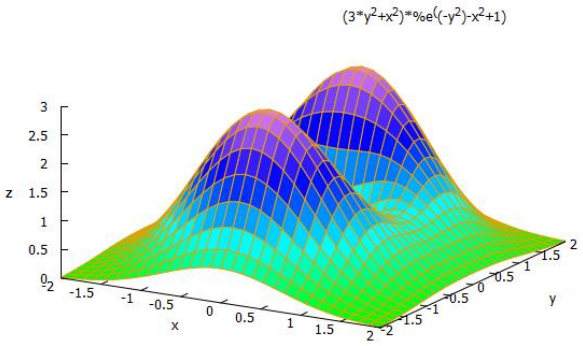
```
(%i8)  plot3d(g(x,y), [x,-2,2], [y,-2,2])$
(%i9)  contour_plot(g(x,y),[x,-2,2],[y,-2,2])$
(%i11) plot3d(g(x,y), [x,-2,2], [y,-2,2], [plot_format,gnuplot],
         [gnuplot_postamble, "set pm3d at b"])$
(%i12) draw3d(surface_hide=true,contour=both,
       explicit(g(x,y),x,-2,2,y,-2,2))$
```







O la propia gráfica de la función:



EJERCICIOS PROPUESTOS

1) Representar gráficamente y estudiar la continuidad en el origen de la función:

$$f(x,y = \frac{xy^2}{x^3+y^3} \quad si\ (x,y) \neq (0,0);\ f(0,0) = 0$$

$$f(x,y = \frac{x^2y}{\sqrt{x^2+y^2}} \quad si\ (x,y) \neq (0,0);\ f(0,0) = 0$$

$$f(x,y = \frac{sen(x^2+y^2)}{x^2+y^2} \quad si\ (x,y) \neq (0,0);\ f(0,0) = 1$$

$$f(x,y = \frac{y(x^2-y^2)}{x^2+y^2} \quad si\ (x,y) \neq (0,0);\ f(0,0) = 0$$

5)

a) Representar gráficamente y hallar los máximos y mínimos relativos de la función:

$$f(x,y) = x^4 + 9y^2 - 3x^2 - 6xy$$

b) Representar gráficamente las curvas de nivel en un entorno de los máximos o mínimos hallados, para comprobar los resultados obtenidos.

6)

a) Representar gráficamente y hallar los máximos y mínimos relativos de la función:

$$f(x,y) = x^5 - 8x^3 + 25x^2 - 32x - 6xy + 9y^2$$

b) Representar gráficamente las curvas de nivel en un entorno de los máximos o mínimos hallados, para comprobar los resultados obtenidos.

7)

a) Representar gráficamente y hallar los máximos y mínimos relativos de la función:

$$f(x,y) = x^4 + y^4 + 2x^2y^2$$

b) Representar gráficamente las curvas de nivel en un entorno de los máximos o mínimos hallados, para comprobar los resultados obtenidos.