

Imágenes gráficas: su contribución al aprendizaje de conceptos de cálculo para estudiantes de ciencias naturales. *Mi experiencia en la resolución de ejercicios utilizando Mathematica y Matlab para Windows.*

Palabras claves : Aprendizaje-Cálculo-Mathematica-Matlab.

Autor : Ing. Blanca I. Niel
Departamento de Matemática, U.N.S
Bahía Blanca
Av. Alem 1253

Domicilio particular : Chequén 115
Barrio Patagonia
Bahía Blanca
T. E : 860562

Imágenes gráficas: su contribución al aprendizaje de conceptos de cálculo para estudiantes de ciencias naturales. Mi experiencia en la resolución de ejercicios utilizando Mathematica y Matlab para Windows.

Resumen.

El interrogante que dio origen a esta experiencia fue cómo hacer que nuestros sentidos perciban la maravilla de un concepto matemático básico. La respuesta tradicional es utilizar el pizarrón de la manera más clara posible, como medio para representaciones gráficas. Una manera alternativa consiste en combinar el uso de una pizarra con el monitor color de una PC. De esta forma se resuelven múltiples ejercicios de Cálculo II aumentando la comprensión y visualización de los mismos por parte de los alumnos, quienes, además, logran generalizaciones y/o aplicaciones posteriores con mayor dinámica.

El presente trabajo contiene la resolución de algunos ejercicios, destacándose en cada caso el concepto que se pretende hacer *aprehender* y las ventajas de la utilización del Mathematica o del Matlab a tal efecto. Los tópicos aquí explicitados son parametrización de curvas diferenciables, parametrización de superficies uniformes, determinación de curvas por intersección de superficies, curvas de nivel, campo gradiente, puntos críticos de funciones diferenciables de dos variables y representación gráfica de problemas sencillos de extremos condicionados.

Palabras claves : Aprendizaje-Cálculo-Mathematica -Matlab.

1. Introducción.

La enseñanza de cálculo requiere en reiteradas oportunidades habilidad para graficar en el pizarrón, en la actualidad es posible soslayar eventuales limitaciones utilizando a tal fin los graficadores para PC, en cuya aplicación, se debe conseguir que las sentencias involucradas sean sencillas y exhiban el concepto matemático de manera explícita, garantizando así una mayor dinámica del aprendizaje. A continuación se desarrollan algunas de las ideas ya implementadas con alumnos de las carreras de Bioquímica, Lic. en Química y Geología de la U. N. S.

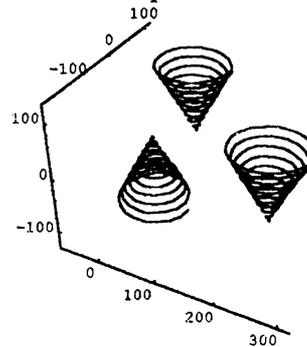
2. Desarrollo.

Las dos primeras imágenes gráficas que aparecen en el cálculo de tres variables son las líneas rectas y los planos. Sus expresiones en forma paramétrica,

$$\begin{aligned} x(t) &= x_0 + a_1 t & x(\lambda, \mu) &= p_0 + b_1 \lambda + c_1 \mu \\ y(t) &= y_0 + a_2 t & y(\lambda, \mu) &= p_1 + b_2 \lambda + c_2 \mu \\ z(t) &= z_0 + a_3 t & z(\lambda, \mu) &= p_2 + b_3 \lambda + c_3 \mu \end{aligned}$$

son los ejemplos más sencillos de curvas diferenciables y superficies uniformes diferenciables, respectivamente. La expresión paramétrica de una curva diferenciable es $\vec{x}(t) = \{x(t), y(t), z(t)\}$, y la expresión paramétrica de una superficie uniforme regular es $\vec{x}(\lambda, \mu) = ((x(\lambda, \mu), y(\lambda, \mu), z(\lambda, \mu)))$. Dada la expresión paramétrica de una curva $\in R^3$ y $\in C^\infty$ si se utiliza la sentencia **ParametricPlot3D** del Mathematica se grafica la misma de manera inmediata; inclusive se pueden graficar varias curvas simultáneamente para un determinado intervalo de variación del parámetro.

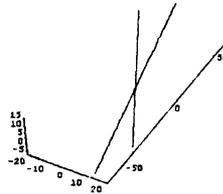
```
ParametricPlot3D[{
  {t Cos[t], t Sin[t], -2t},
  {t Cos[t]+50, t Sin[t]+50, 2t},
  {t Cos[t] + 250, t Sin[t]-100, 2t},
},
{t, 0, 20Pi},
PlotPoints->200,
Boxed->False
];
```



La expresión de la sentencia es didáctica como así también la manera de ingresar los datos. Si la curva fuera plana, la sentencia a escoger es **ParametricPlot**.

Dadas dos rectas que se intersectan ¿ Es posible graficarlas y obtener las coordenadas de su punto de intersección rápidamente ? La respuesta es afirmativa.

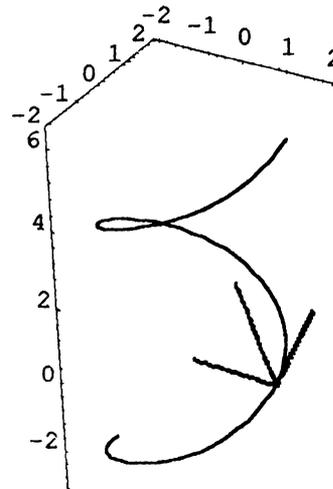
```
ParametricPlot3D[{
  {1+2*t, 1-4*t, 5-t},
  {4-t, -1+6*t, 4+t},
},
{t, -10, 10},
Boxed->False ];
```



```
Solve[{1+2t-4+v==0, 1-4t+1-6v==0, 5-t-4-v==0}, t, v]
{{t -> 2}}
```

Calcular la longitud de una determinada curva en el espacio, la expresión de su curvatura de flexión y de torsión, su vector tangente en un punto, los versores del Triedro Intrínseco, resultan evaluaciones directas y rápidas utilizando el Mathematica versión 2.2.3. A continuación se resuelve un caso particular.

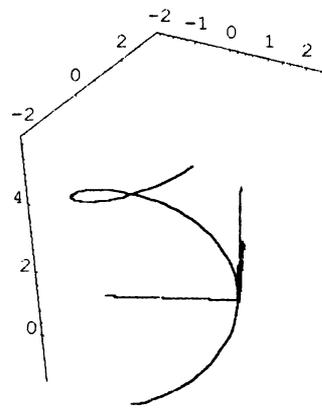
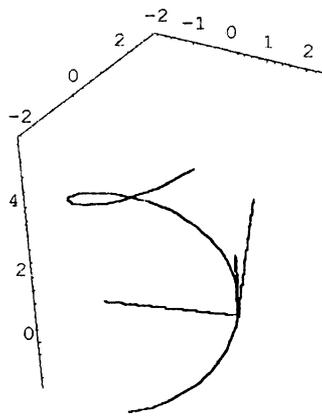
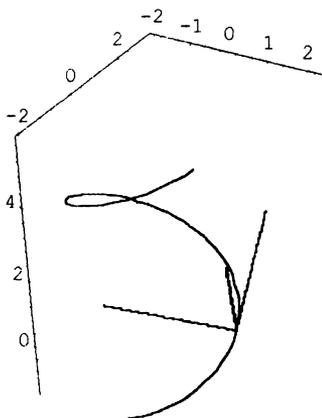
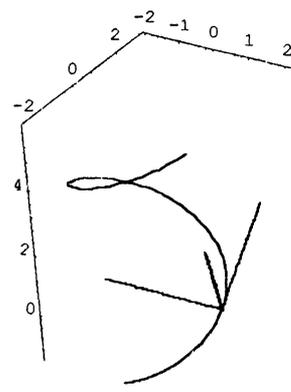
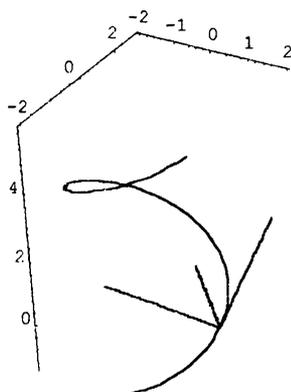
```
ParametricPlot3D[{
  {2 Cos[t], 2 Sin[t], t},
  {2, 4(t+Pi)/(3Pi*sqrt[5]),
  2(t+Pi)/(3Pi*sqrt[5])},
  {-2(t+Pi)/(3Pi)+2, 0, 0},
  {2, -2(t+Pi)/(3Pi),
  4(t+Pi)/(3Pi)},
{t, -Pi, 2Pi}, Boxed->False ]
```



```

Do[
ParametricPlot3D[
{
{o[0+n,t],p[0+n,t],q[0+n,t]},
{u[0+n,t],v[0+n,t],w[0+n,t]},
{x[0+n,t],y[0+n,t],z[0+n,t]},
{2Cos[t/Sqrt[5]],2Sin[t/Sqrt[5]],
t/Sqrt[5]}},
{t,-Pi,4Pi},
Boxed->False
],
{n,0,1,0.25}]

```



La animación anterior permite visualizar que el versor binormal se aleja del normal, confirmando el carácter dextrógiro de la curva.

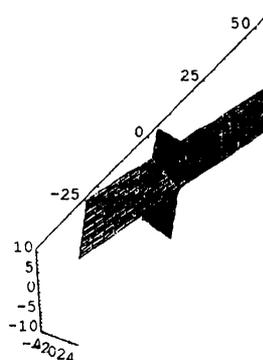
La ecuación paramétrica de una recta en el espacio puede surgir de resolver un sistema lineal de ecuaciones compatible sobredimensionado. Este problema de carácter algebraico se interpreta geoméricamente como la intersección de dos planos, cuyas expresiones explícitas o biparamétricas mantendrán la forma dada en la primer página. Luego, es posible visualizar este problema usando la sentencia ParametricPlot3D con argumentos dependientes de dos parámetros, por ahora relacionados de manera lineal entre sí, para cada coordenada del espacio.

A continuación se resuelve el problema algebraicamente y luego se lo grafica.

```
Solve[{x-u==0,8-2x-4z+(1/3)-w/3+u/3==0,z-w==0}]
```

```

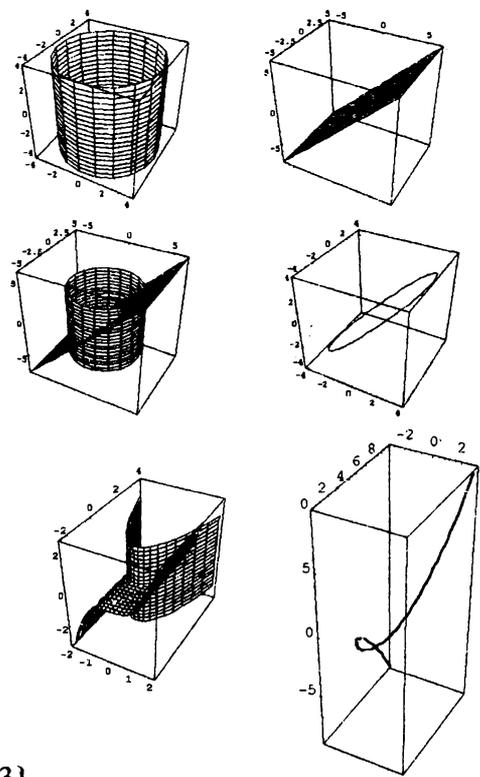
    13 z      13 z
    {{u -> 5 - ----, x -> 5 - ---- , w -> z } }
       5          5
    ParametricPlot3D[{
      {x,8-2x-4z,z},
      {x,-(1/3)+z/3-x/3,z}
    },
    {x,-4,4},{z,-10,10},
    Boxed->False,
    Shading->False
    ]
  
```



Desde esta idea se podría concebir la generación de ciertas curvas como intersección de dos superficies. Los dos ejemplos siguientes lo ilustran :

```

    p1=ParametricPlot3D[
      {4 Cos[t], 4 Sin[t], z},
      {t, 0, 2Pi}, {z,-4,4},
      Shading->False]
    p2=ParametricPlot3D[{x,t,x},
      {x,-2Pi,2Pi},{t,-5,5},
      Shading->False]
    Show[p1,p2]
    ParametricPlot3D[{4 Cos[t], 4 Sin[t],
      4 Cos[t]}, {t,0,2Pi}]
  
```



```

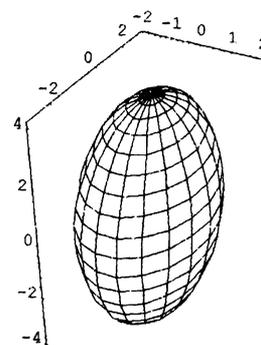
    ParametricPlot3D[{t,t^2,(1/3)t^3},{t,-3,3},
    Shading->False]
  
```

A partir de la expresión paramétrica del plano se está en condiciones de parametrizar ciertas superficies , dando por sentado que los parámetros independientes involucrados serán dos. Así se construyen las cuádricas normales, que se grafican a continuación.

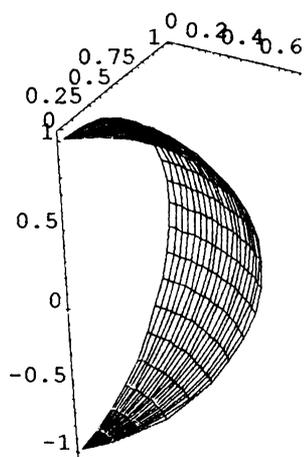
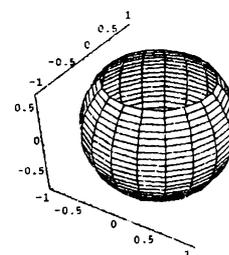
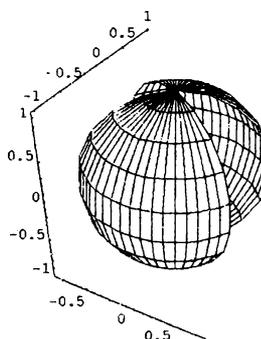
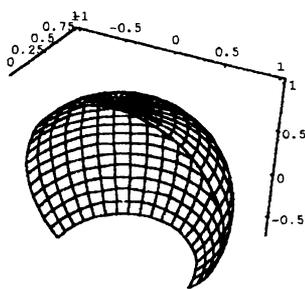
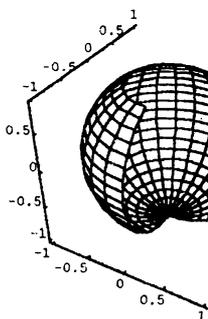
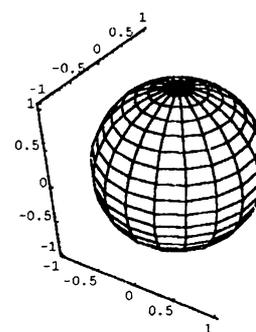
```
ParametricPlot3D[{2 Cos[u] Sin[v], 3 Sin[u] Sin[v],
4 Cos[v]}, {u, 0, 2 Pi}, {v, 0, Pi},
Shading->False,
Boxed->False]
```

El elipsoide pasa a ser una esfera, considerando las parametrizaciones para cada coordenada con el mismo valor de las constantes.

```
ParametricPlot3D[{Cos[u] Sin[v], Sin[u] Sin[v], Cos[v]},
{u, 0, 2 Pi}, {v, 0, Pi},
Boxed->False,
Shading->False]
```



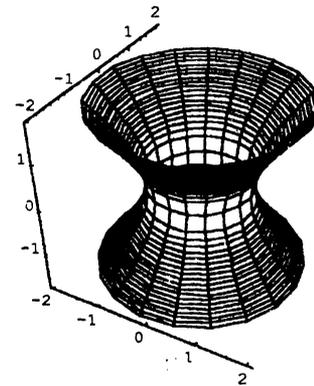
Estas imágenes, que se obtienen variando los ángulos, se utilizan para calcular áreas superficiales, volúmenes, para determinar el flujo de un campo vectorial a través de superficies orientables compuestas por trozos de superficies parametrizables, etc.



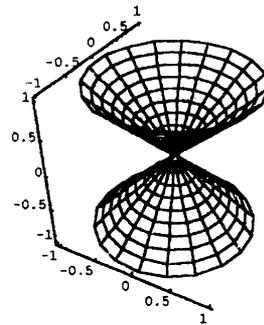
```
ParametricPlot3D[{Cos[u] Sin[v], Sin[u] Sin[v], Cos[v]},
{u, Pi/6, 5 Pi/6}, {v, 0, Pi}, Boxed->False,
Shading->False]
```

Luego de la presentación de los elipsoides, se continúa con las superficies cuadráticas y se presentan los hiperboloides de una y de dos hojas. Cabe mencionar que si se los considera como las superficies de nivel de la función de tres variables $f(x, y, z) = x^2 + y^2 - z^2$ (analizando las superficies de tipo $x^2 + y^2 - z^2 = c$, $c > 0$, $c < 0$, $c = 0$), se pasa desde hiperboloides de una hoja a conos y, por último, a hiperboloides de dos hojas.

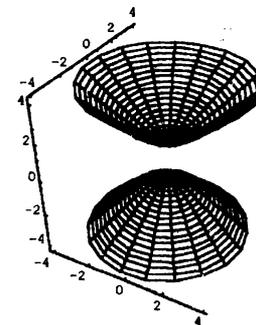
```
ParametricPlot3D[{
  {u Cos[v], u Sin[v], Sqrt[u^2-1]},
  {u Cos[v], u Sin[v], -Sqrt[u^2-1]},
},
{u,1,2}, {v,0,2Pi},
Boxed->False,
Shading->False]
```



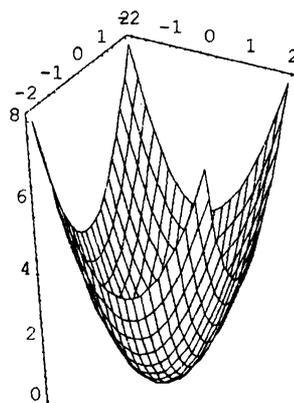
```
ParametricPlot3D[{
  {u Cos[v], u Sin[v], u},
  {u Cos[v], u Sin[v], u}
},
{u,-1,1}, {v,0,2Pi},
Boxed->False,
Shading->False]
```



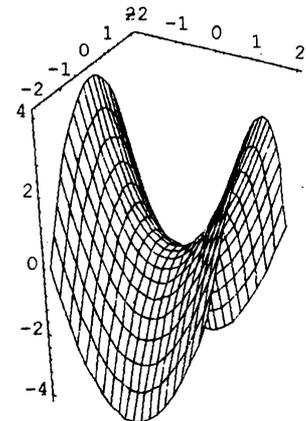
```
ParametricPlot3D[{
  {u Cos[v], u Sin[v], Sqrt[u^2+1]},
  {u Cos[v], u Sin[v], -Sqrt[u^2+1]},
},
{u,0,4}, {v,0,2Pi},
Boxed->False,
Shading->False]
```



```
ParametricPlot3D[{x, y, x^2+y^2},
{x,-2,2}, {y,-2,2},
Boxed->False,
Shading->False]
```

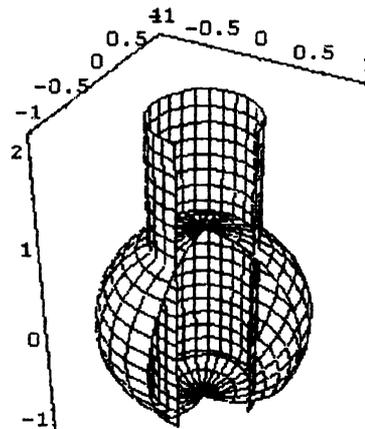


```
ParametricPlot3D[{x, y, x^2-y^2},
{x,-2,2}, {y,-2,2},
Boxed->False,
Shading->False]
```



Por último cabe destacar que dentro de las cuadráticas se incluyen los cilindros propiamente dichos y los cilindros parabólicos, que son graficables rápidamente. La siguiente figura muestra la parametrización de un típico cilindro circular superpuesto con una esfera unitaria.

```
ParametricPlot3D[{
  {Cos[u] Sin[v], Sin[u] Sin[v], Cos[v]},
  {Cos[u]/2, Sin[u]/2, v-1}
},
{u, 0, 3Pi/2}, {v, 0, Pi},
Boxed->False,
Shading->False]
```



Se parametriza y grafica el toro, $S^1 \times S^1$, como otro ejemplo de superficies orientables.

a=1.0;

b=0.25;

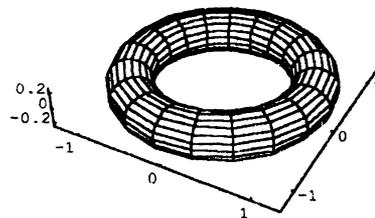
r[v_]:=a+b*Cos[v]

x[u_,v_]:=r[v] Cos[u]

y[u_,v_]:=r[v] Sin[u]

z[v_]:=b Sin[v]

```
ParametricPlot3D[{x[u,v],y[u,v],z[v]},
{u,0,2Pi},{v,0,2Pi},
Boxed->False,
Shading->False]
```



Un algoritmo que representa la superficie no orientable de una sola cara llamada Banda de Moebius se muestra a continuación.

a=1.0

b=0.5

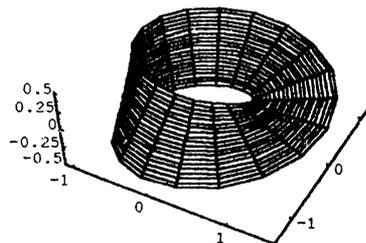
r[t_,v_]:=a+b*v*Cos[t/2]

x[t_,v_]:=r[t,v]*Cos[t]

y[t_,v_]:=r[t,v]*Sin[t]

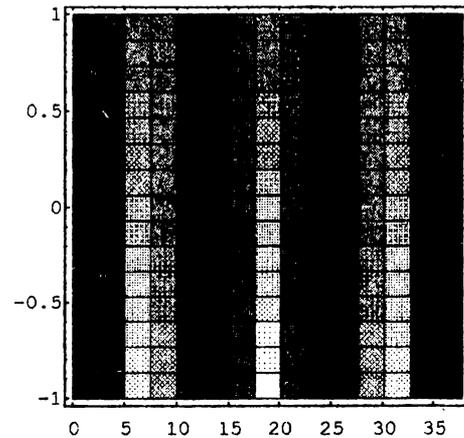
z[t_,v_]:=b*v*Sin[t/2]

```
ParametricPlot3D[{x[t,v],y[t,v],z[t,v]},
{t,0,2Pi},{v,-1,1},
Boxed->False,
Shading->False]
```



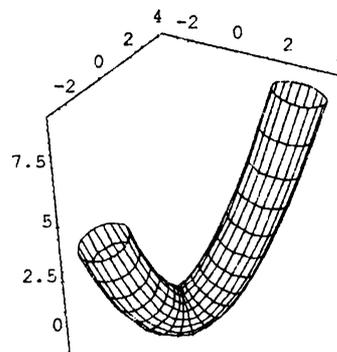
La característica topológica de la superficie de Moebius se puede considerar mediante la noción de vector normal a una superficie. Se verifica mediante una secuencia del Mathematica que dicho vector normal no vuelve a su posición original después de recorrerse la Banda de Moebius una vuelta completa.

```
Needs["LinearAlgebra`CrossProduct`"]
$ContextPath
{LinearAlgebra`CrossProduct`, Global`, System`}
Context[CrossProduct]
Global`
tant[t_,v_]:=D[{x[t,v],y[t,v],z[t,v]},t]
tav[t_,v_]:=D[{x[t,v],y[t,v],z[t,v]},v]
nz[t_,v_]:=Cross[tant[t,v],tav[t,v]][[3]]
nz[t,v]
      t
-0.125 v - 0.5 Cos[-] - 0.125 v Cos[t]
      2
DensityPlot[%,{t,0,12Pi},{v,-1,1}]
```



Ahora se da respuesta a la siguiente pregunta ¿Cómo generar una superficie tubular a partir de cualquier curva del espacio expresada en forma paramétrica ?

```
x[t_]:=t
y[t_]:=t
z[t_]:=t^2
dx[t_]:=D[x[t],t]
dy[t_]:=D[y[t],t]
dz[t_]:=D[z[t],t]
t[t_]={dx[t],dy[t],dz[t]}
n[t_]={dy[t],-dx[t],0}
Needs["LinearAlgebra`CrossProduct`"]
Context[CrossProduct]
w[t_]:=Cross[t[t],n[t]]
normn[t_]:=Sqrt[n[t].n[t]]
normw[t_]:=Sqrt[w[t].w[t]]
p[t_,u_]=(n[t]/normn[t])Cos[u]+
(w[t]/normw[t])Sin[u]
s[t_,u_]={x[t],y[t],z[t]}+p[t,u]
ParametricPlot3D[s[t,u],{t,-2,3},{u,0,2Pi},
  Boxed->False,
  Shading->False]
```



¿Qué hacer cuando las superficies cuadráticas no están en forma canónica o normal?
Las expresiones cuadráticas de tres variables más generales tienen la siguiente forma

$$f(x, y, z) = a_{11}x^2 + a_{22}y^2 + a_{33}z^2 + 2a_{12}xy + 2a_{13}xz + 2a_{23}yz + a_1x + a_2y + a_3z + k = 0$$

Se clasifica rápidamente una cuádrica utilizando conceptos algebraicos y el correspondiente empleo de los autovalores y autovectores. Para su cálculo se utiliza el Matlab. Para $A = \begin{bmatrix} 2 & -2 & -1 \\ -2 & 2 & 1 \\ -1 & 1 & 5 \end{bmatrix}$, $a = [10, 26, -2]$ y $k=0$, la expresión en las nuevas coordenadas son :

$3\tilde{y}^2 + 6\tilde{z}^2 - 11,31\tilde{x} + 19,63\tilde{y} - 16,33\tilde{z} = 0$ si se completa cuadrados respecto de \tilde{x} y de \tilde{z} se reconoce un paraboloides elíptico.

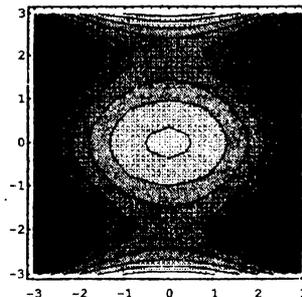
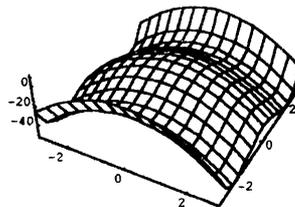
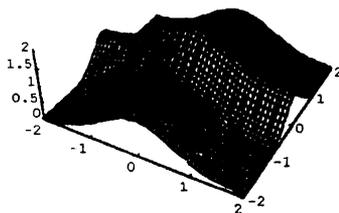
El reconocimiento de las cuádricas mediante técnicas algebraicas tiene su aplicación en la determinación de los extremos de funciones diferenciables, si se le asocia el concepto de campo Gradiente y curvas de nivel dicha caracterización es graficable. Se utiliza el Matlab pues mediante las sentencia **gradient** y **quiver** en primera instancia calcula y luego grafica el campo vectorial gradiente.

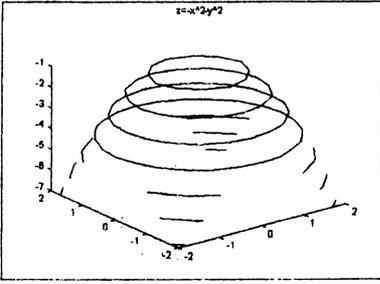
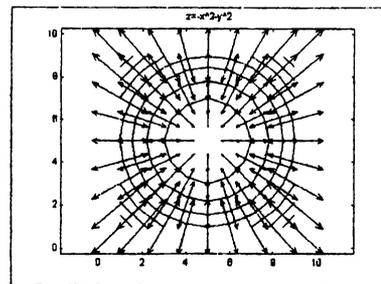
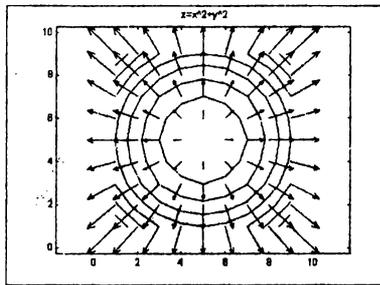
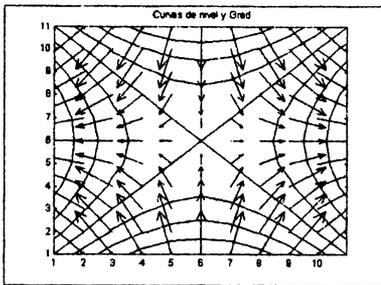
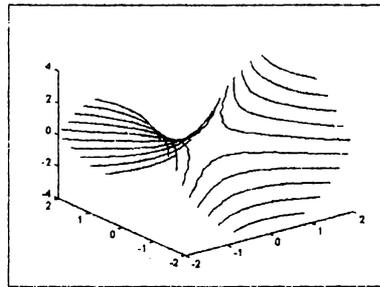
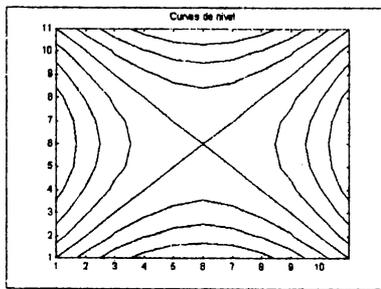
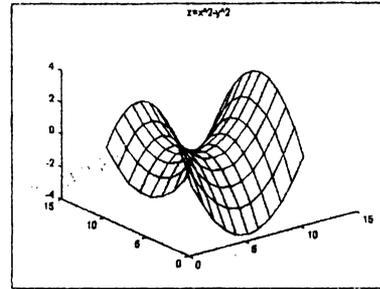
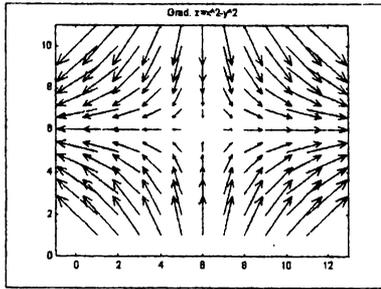
```
[x,y]=meshdom(-2:.4:2,-2:.4:2); z=x.^2-y.^2; [px,py]=gradient(z,.4,.4);
» axis('square'); » quiver(px,py,3); title('Grad. z=x^2-y^2'); » mesh(z), title('z=x^2-y^2');
» contour(z), title('Curvas de nivel'); » contour3(x,y,z,10); » quiver(px,py,3); hold on;
» contour(z), title('Curvas de nivel y Grad').
```

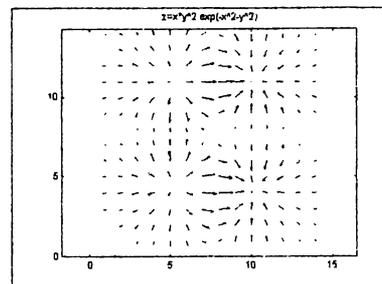
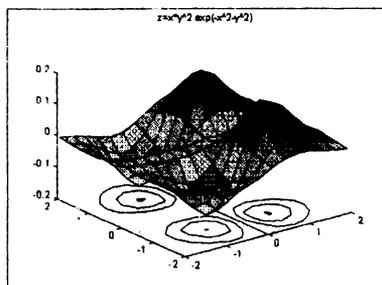
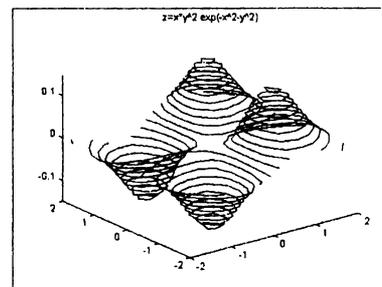
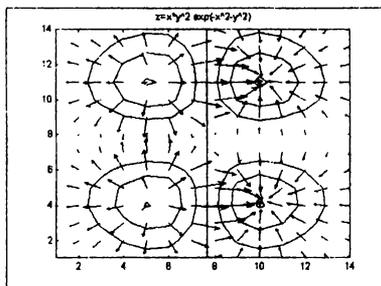
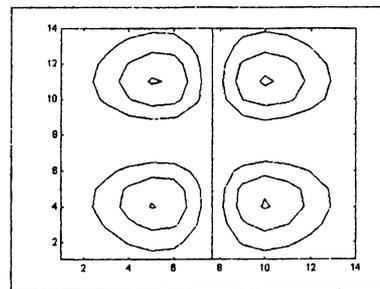
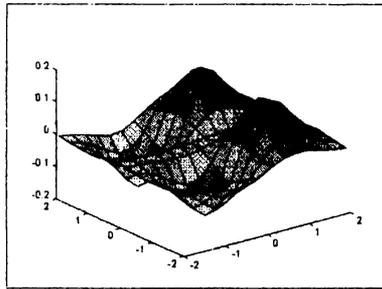
Las sentencias anteriores corresponden a $z = x^2 - y^2$, las gráficas de las funciones $z = x^2 + y^2$ y $z = -x^2 - y^2$ se obtienen variando el tamaño de la grilla de puntos a 0.5 y la longitud de las flechas de los vectores gradientes a 2.

El gradiente en cada punto del dominio de una función diferenciable apunta en el sentido del mayor crecimiento de la misma, el gradiente es el vector nulo en (0,0) para las tres funciones, las imágenes de la página diez muestran las diferencias de comportamiento de cada una, la silla no tiene extremo, el paraboloides positivo, posee un mínimo y en el paraboloides negativo un máximo; en la silla los vectores gradiente en un entorno del (0,0), y en el eje x, se ajejan del (0,0), en el eje y, los vectores gradiente convergen en él, mientras que en $z = x^2 + y^2$ los vectores apuntan en direcciones radiales todos alejándose del origen, indicando que el (0,0) es un mínimo de la función, en el último caso todos los vectores en forma radial convergen al (0,0), es decir están indicando que éste es el mayor valor de la función. Para visualizar un problema con mayor número de puntos críticos, se grafica la función $f(x, y) = xy^2 \exp(-x^2 - y^2)$ en la página once.

En las gráficas siguientes ¿Cuál de las dos superficies se corresponde con las curvas de nivel ?



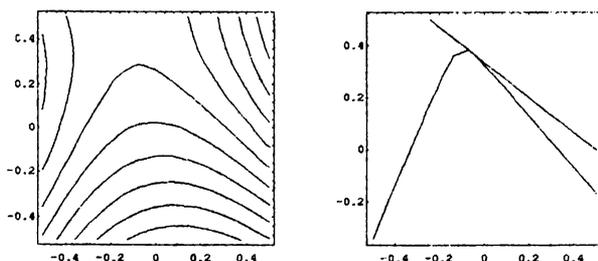




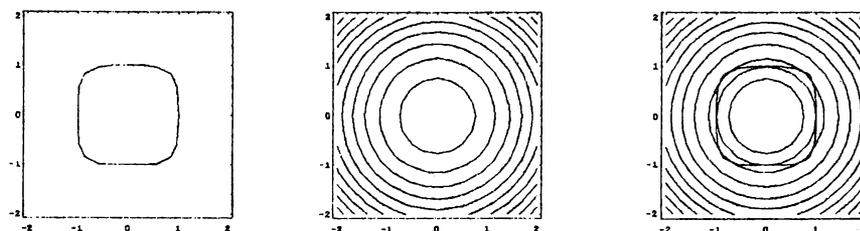
```
[x,y]=meshgrid(-2:3:2);
» m=x.*y;
» n=m.*y;
» z=n.*exp(-x.^2-y.^2);
» surf(x,y,z)
» contour(z);
» hold on;
[px,py]=gradient(z,3,3);
```

```
hold on;
quiver(px,py,2);title('z=x*y^2
exp(-x^2-y^2)');
» hold off; contour3(x,y,z);
title('z=x*y^2 exp(-x^2-y^2)');
» hold off; contour3(x,y,z,20);
title('z=x*y^2 exp(-x^2-y^2)')
» hold off; surfc(x,y,z);
title('z=x*y^2 exp(-x^2-y^2)');
» hold off; quiver(px,py);
title('z=x*y^2 exp(-x^2-y^2)');
```

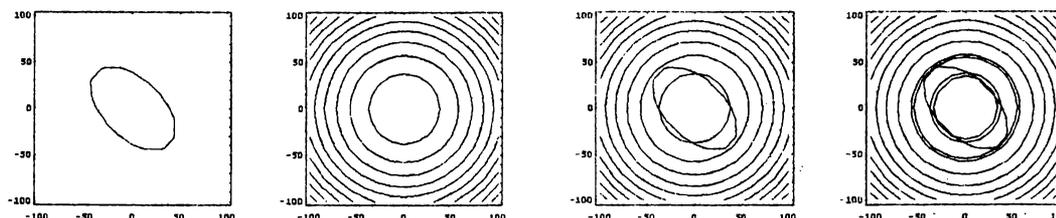
Se desea extremar la función $f(x,y) = 2x^2 + xy - y^2 + y$ bajo la restricción $2x + 3y - 1 = 0$. Se resuelve por el Método de los Multiplicadores de Lagrange y se adjunta la correspondiente representación gráfica.



Se resuelve graficamente el siguiente problema ¿ Encontrar los puntos que minimizan y maximizan la distancia al origen que están ubicados sobre $x^4 + y^4 = 1$?

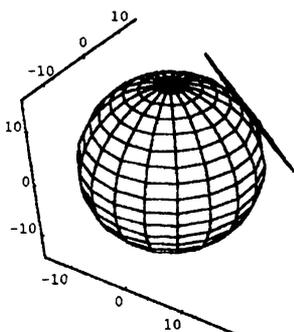


La siguiente secuencia gráfica se puede interpretar como un problema análogo al anterior de minimización de la función distancia con una restricción cuadrática no canónica.



¿Cuál es el punto del plano $x + y + z - 25 = 0$ que está más cerca del origen ?

Rta: $[25/3, 25/3, 25/3]$



3. Conclusiones.

- 1.- Este trabajo presenta evidencias claras de la valiosa ayuda que significan estos softwares para la enseñanza. En particular, la estructura de determinadas sentencias involucran en sí mismas una reafirmación del concepto a ser aprehendido.
- 2.- Se consigue una verdadera amalgama entre conceptos teóricos y su visualización gráfica.
- 3.- La posibilidad de cambiar los parámetros rápidamente y de obtener inmediata reposición de imágenes gráficas aceleran la dinámica del aprendizaje, ya que el mismo estudiante a partir de un ejemplo comienza a resolver problemas análogos.
- 4.- Respecto a la elección de uno u otro software, considero que es conveniente dejar que el alumno trabaje con aquél que conozca; de lo contrario sugerirle aquellos con los que trabaja el docente.
- 5.- En la secuencia de ejercitación aquí propuesta, tanto el Mathematica como el Matlab son excelentes, la elección de uno u otro para resolver los problemas está asociada directamente con la sencillez de las sentencias específicas involucradas en los mismos.
- 6.- En lo que respecta a la ejercitación propuesta existen muchos conceptos dentro del llamado cálculo básico, que se pueden tratar con estas herramientas y que no están desarrollados en el presente trabajo, por ejemplo las técnicas numéricas elementales para determinar un mínimo irrestricto de funciones de varias variables, desde el Método del Gradiente al Método de Broyden-Fletcher-Goldfarb-Shanno (Toolbox, Optimization del Matlab); tópicos introductorios de Ecuaciones Diferenciales, desde el Algoritmo de Euler a un Runge Kutta, también aplicaciones de la teoría de campos vectoriales, etc.

4. Bibliografía

- 1.-Exploring Mathematics with Mathematica . Theodore W. Gray and Jerry Glynn.
- 2.-Mathematica for the Sciences. Richard E. Crandall.
- 3.-Matlab. High-Performance Numeric Computation and Visualization Software. Reference Guide. The Math Works, Inc.
- 4.-Optimization TOOLBOX For Use with Matlab. Andrew Grace. The Math Works, Inc.
- 5.-Mathematica . A System for Doing Mathematics by Computer. Stephen Wolfram. Addison-Wesley Publishing Company, inc.
6. Análisis Matemático -Tomo II- J. Rey Pastor, P. Pi Calleja y C. A. Trejo. Kapelus.
7. Advanced Calculus of Several Variables. C.H. Edwards, Jr. Academic Press.
8. Numerical Analysis An Introduction. Lars Eldén. Linde Wittmeyer-Koch. Academic Press.