

Actividad 8: Iniciando cálculo simbólico con Maxima

Reyna Cornelio

15 de Abril 2016

El objetivo de esta actividad es familiarizarse con el entorno de WxMáxima. Para ello se tomaron algunas funciones del manual de Jay Kerns, dentro de las cuales se modificaron las funciones, los colores de las gráficas.

A continuación se muestran las funciones tomadas del manual, así también se muestran las modificaciones que se realizaron.

2. Geometría en tres dimensiones

2.1 Vectores y Álgebra Lineal

El funcionamiento de Maxima en esta sección es definir vectores para posteriormente realizar las operaciones deseadas.

Ejemplo;

2.2 Líneas, Planos y Superficies cuadráticas

2.3 Funciones Vectoriales

2.4 Longitud de arco y Curvaturas

3. Funciones de Varias Variables

3.1 Derivadas Parciales

Para definir una función es necesario definirla con una letra mayúscula.

G: $1/x^{-2} / y^3$;

$$\frac{x^2}{y^3}$$

Para que Maxima realice la derivada de la función antes definida, es necesario poner el comando que se muestra a continuación. Si se tienen mas funciones, se requiere de definirla dentro de los parentesis.

`diff(G);`

$$\frac{2x \operatorname{del}(x)}{y^3} - \frac{3x^2 \operatorname{del}(y)}{y^4}$$

Si se quiere saber el valor en un punto determinado, se define de la siguiente manera:

`diff(G, x, 1, y, 1, x, 5);`

0

3.2 Aproximación Lineal y Diferenciales

Para esta sección, es necesario definir la función, en este caso una de dos variables a la cual se le hará una aproximación.

`f(x,y) :=exp(x^6)* cos(y);`

`f(x,y) := exp(x^6) cos(y)`

La aproximación que se realizará para la función es una serie de potencias con Taylor.

`taylor(f(x,y), [x,y], [5,2], 5);`

$$\begin{aligned} & \cos(2) e^{15625} + (18750 \cos(2) e^{15625} (x-5) - \sin(2) e^{15625} (y-2)) + \frac{351581250 \cos(2) e^{15625} (x-5)^2 - 37500 \sin(2) e^{15625} (y-2)^2}{2} \\ & (6592851577500 \cos(2) e^{15625} (x-5)^3 - 1054743750 \sin(2) e^{15625} (y-2) (x-5)^2 - \\ & 56250 \cos(2) e^{15625} (y-2)^2 (x-5) + \sin(2) e^{15625} (y-2)^3) / 6 + (123635744367196500 \cos(2) e^{15625} (x-5)^4 - \\ & 26371406310000 \sin(2) e^{15625} (y-2) (x-5)^3 - 2109487500 \cos(2) e^{15625} (y-2)^2 (x-5)^2 + \\ & 75000 \sin(2) e^{15625} (y-2)^3 (x-5) + \cos(2) e^{15625} (y-2)^4) / 24 + (2318664702396234378600 \cos(2) e^{15625} (x-5)^5 - \\ & 618178721835982500 \sin(2) e^{15625} (y-2) (x-5)^4 - 65928515775000 \cos(2) e^{15625} (y-2)^2 (x-5)^3 + \\ & 3515812500 \sin(2) e^{15625} (y-2)^3 (x-5)^2 + 93750 \cos(2) e^{15625} (y-2)^4 (x-5) - \\ & \sin(2) e^{15625} (y-2)^5) / 120 + \dots \end{aligned}$$

3.3 Regla de la Cadena y Diferenciación Implícita

Definimos la función:

$$F: 3x^{1/2}yz + 2x^2y^3z^4 - 7/3x - xz;$$

$$2x^2y^3z^4 + \frac{3xyz}{2} - xz - \frac{7x}{3}$$

Realizamos la derivada parcial de cada variable, para ello se define la función y la variable a la que se hará dicha derivada:

$$F_x: \text{diff}(F, x);$$

$$4xy^3z^4 + \frac{3yz}{2} - z - \frac{7}{3}$$

$$F_y: \text{diff}(F, y);$$

$$6x^2y^2z^4 + \frac{3xz}{2}$$

$$F_z: \text{diff}(F, z);$$

$$8x^2y^3z^3 + \frac{3xy}{2} - x$$

3.4 Derivadas Diferenciales y Gradiente

Definimos la función:

$$f(x,y) := \exp(x^6) \cos(y);$$

$$f(x,y) := \exp(x^6) \cos(y)$$

Cambiamos al sistema de coordenadas adecuado. Para ello ponemos el siguiente comando.

$$\text{load}(\text{vect});$$

$$C: /PROGRA2/MAXIMA1.0-2/share/maxima/5.28.0-2/share/vector/vect.mac$$

Para el factor de escala:

$$\text{scalefactors}([x,y]);$$

done

A continuación nos encontramos el gradiente:

```

gdf: grad(f(x,y));

grad  $\left( e^{x^6} \cos(y) \right)$ 

ev(express(gdf), diff);

 $[6 x^5 e^{x^6} \cos(y), -e^{x^6} \sin(y)]$ 

define(gdf(x,y),

gdf(x,y) :=  $[6 x^5 e^{x^6} \cos(y), -e^{x^6} \sin(y)]$ 

define(gdf(x,y), %);

gdf(x,y) :=  $[6 x^5 e^{x^6} \cos(y), -e^{x^6} \sin(y)]$ 

float(%);

gdf(x,y) :=  $[6.0 x^5 2.718281828459045^{x^6} \cos(y), -1.0 2.718281828459045^{x^6} \sin(y)]$ 

```

3.5 Optimización y Extremos Locales

Para presentar los puntos críticos tenemos que resolver el sistema de ecuaciones:

```

f(x,y) := x^3 + 1/2 * y^2 - 3 * x * y;

```

$f(x,y) := x^3 + \frac{1}{2}y^2 + (-3)xy$ hora vamos a hallar el primer orden derivadas parciales de f , establecen iguales a cero y resolver para los valores de (x,y) . Hacemos esto con la función de resolver, lo que supone que las expresiones se igualan a cero por defecto. Tenga en cuenta que resolver NDS fi todas las soluciones reales y complejas; sólo nos interesan las soluciones de bienes valiosos, sin embargo, y sin ignorar el resto.

```

fx : diff(f(x,y), x);

 $3x^2 - 3y$ 

fy : diff(f(x,y), y);

```

$$y - 3x$$

Obtenemos el mapa de los contornos:

```
draw3d(implicit(f(x,y), x, -2, 2, y, -2, 2), contour = map);
```

```
draw3d(implicit((y^2/2 - 3*x*y + x^3, x, -2, 2, y, -2, 2), contour = map))
```

Obtenemos el hessiano:

```
H: hessian(f(x,y), [x,y]);
```

$$\begin{pmatrix} 6x & -3 \\ -3 & 1 \end{pmatrix}$$

obtenemos el determinante

```
determinant(H);
```

$$6x - 9$$

3.6 Multiplicadores de Lagrange

Encontrar los valores extremos de la función:

```
f(x,y) := x^2 + y^2;
```

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

```
g: 3*x^2 + y^2;
```

$$y^2 + 3x^2$$

Creamos el sistema de ecuaciones que queremos resolver:

```
eq1: diff(f(x,y), x) = h * diff(g, x);
```

$$2x = 6hx$$

```
eq1: diff(f(x,y), x) = h * diff(g, x);
```

$$2x = 6hx$$

```
eq2: diff(f(x,y), y) = h * diff(g, y);
```

$$2y = 2hy$$

```
eq3: g = 3;
```

$$y^2 + 3x^2 = 3$$

4. Integración Múltiple

4.1 Integrales Dobles

Definimos la función

$$f(x, y) := 5x^7 - \frac{2}{5}xy + 2y;$$

$f(x, y) := 5x^7 - \frac{2}{5}xy + 2y$ Para que maxima realice la doble integración ponemos le siguiente comando:

```
integrate(integrate(f(x,y), y), x);
```

$$-\frac{x^2 y^2}{10} + x y^2 + \frac{5 x^8 y}{8}$$

Si queremos un valor exacto se agrega los puntos en los que se quiere que sea evaluado.

```
integrate(integrate(f(x,y), y, x^1/3, 7 - x), x, 0, 1);
```

$$\frac{15091}{360}$$

4.2 Integración en Coordenadas Polares

Podemos integrar en coordenadas polares de la forma obvia. Hacemos simplemente la sustitución $x = R \cos \theta$ $y = R \sin \theta$, entonces no se olvide de multiplicar el integrando por r .

```
[x,y]: [1/2*r * cos(theta), r * sin(2*theta)];
```

$$\left[\frac{r \cos(\theta)}{2}, r \sin(2\theta) \right]$$

```
integrate(integrate(f(x,y) * r, r, 0, 2*cos(theta)), theta, -%pi/2, %pi/2);
```

$$\frac{3575 \pi}{8192}$$

4.3 Integrales Triples

Para las integrales triples, el procedimiento es similar al de la integración doble, sólo se le agraga un integrate más.

```
f(x,y,z) := y/z; (%o1);
```

$$f(x, y, z) := \frac{y}{z}$$

```
[x,y,z] : [9*r*cos(2*theta), r*sin(1/2*theta), z];
```

$$[9r \cos(2\theta), r \sin\left(\frac{\theta}{2}\right), z]$$

```
integrate(integrate(integrate(f(x,y,z)*r, z,1,3), r,0,2), theta,0,3*%pi);
```

$$\frac{16 \log(3)}{3}$$

4.4 Integrales en Coordenadas Cilíndricas y Esféricas

Estas integrales se calculan igual que las integrales triples ordinarias, excepto multiplicamos el integrando por r (en coordenadas cilíndricas) o ρ^2 (en coordenadas esféricas).

```
f(x,y,z) := y/z; (%o1);
```

$$f(x, y, z) := \frac{y}{z}$$

```
[x,y,z] : [9*r*cos(2*theta), r*sin(1/2*theta), z];
```

$$[9r \cos(2\theta), r \sin\left(\frac{\theta}{2}\right), z]$$

```
integrate(integrate(integrate(f(x,y,z)*r, z,1,3), r,0,2), theta,0,3*%pi);
```

$$\frac{16 \log(3)}{3}$$

4.5 Cambio de Variable

Para más general transformaciones $x = x(u, v)$ y $y = y(u, v)$ que pueden utilizar la función jacobiana

```
f(x,y) := 3*x + 9*y;
f(x,y) := 3 x + 9 y
[x,y]: [u^3 - v^4, 5 * u * v];
[u^3 - v^4, 5 u v]
J: jacobian([x,y], [u,v]);
(3 u^2 -4 v^3)
(5 v 5 u)
J: determinant(J);
20 v^4 + 15 u^3
integrate(integrate(f(x,y) * J, u,1,2), v,3,4);
- 70104845
84
```

5. Cálculo Vectorial

5.1 Campos Vectoriales

```
coord: setify(makelist(k,k,-4,4));
points2d: listify(cartesian_product(coord ,coord)); vf2d(x,y):= vector([x,y],[cos
-4, -3, -2, -1, 0, 1, 2, 3, 4
[[-4, -4], [-4, -3], [-4, -2], [-4, -1], [-4, 0], [-4, 1], [-4, 2], [-4, 3], [-4, 4], [-3, -4], [-3, -3], [-3, -2], [-3, -1], [-3, 0], [-3, 1], [-3, 2], [-3, 3], [-3, 4], [-2, -4], [-2, -3], [-2, -2], [-2, -1], [-2, 0], [-2, 1], [-2, 2], [-2, 3], [-2, 4], [-1, -4], [-1, -3], [-1, -2], [-1, -1], [-1, 0], [-1, 1], [-1, 2], [-1, 3], [-1, 4], [0, -4], [0, -3], [0, -2], [0, -1], [0, 0], [0, 1], [0, 2], [0, 3], [0, 4], [1, -4], [1, -3], [1, -2], [1, -1], [1, 0], [1, 1], [1, 2], [1, 3], [1, 4], [2, -4], [2, -3], [2, -2], [2, -1], [2, 0], [2, 1], [2, 2], [2, 3], [2, 4], [3, -4], [3, -3], [3, -2], [3, -1], [3, 0], [3, 1], [3, 2], [3, 3], [3, 4], [4, -4], [4, -3], [4, -2], [4, -1], [4, 0], [4, 1], [4, 2], [4, 3], [4, 4]]);
vf2d(x,y) := vector([x,y],[cos(y),x])
vector([-4, -4], [cos(4), -2/3]), vector([-4, -3], [cos(3), -2/3]), vector([-4, -2], [cos(2), -2/3]), vector([-4, -1], [cos(1), -2/3]), vector([-4, 0], [cos(0), -2/3]), vector([-4, 1], [cos(1), -2/3]), vector([-4, 2], [cos(2), -2/3]), vector([-4, 3], [cos(3), -2/3]), vector([-4, 4], [cos(4), -2/3]), vector([-3, -4], [cos(4), -2/3]), vector([-3, -3], [cos(3), -2/3]), vector([-3, -2], [cos(2), -2/3]), vector([-3, -1], [cos(1), -2/3]), vector([-3, 0], [cos(0), -2/3]), vector([-3, 1], [cos(1), -2/3]), vector([-3, 2], [cos(2), -2/3]), vector([-3, 3], [cos(3), -2/3]), vector([-3, 4], [cos(4), -2/3]), vector([-2, -4], [cos(4), -2/3]), vector([-2, -3], [cos(3), -2/3]), vector([-2, -2], [cos(2), -2/3]), vector([-2, -1], [cos(1), -2/3]), vector([-2, 0], [cos(0), -2/3]), vector([-2, 1], [cos(1), -2/3]), vector([-2, 2], [cos(2), -2/3]), vector([-2, 3], [cos(3), -2/3]), vector([-2, 4], [cos(4), -2/3]), vector([-1, -4], [cos(4), -2/3]), vector([-1, -3], [cos(3), -2/3]), vector([-1, -2], [cos(2), -2/3]), vector([-1, -1], [cos(1), -2/3]), vector([-1, 0], [cos(0), -2/3]), vector([-1, 1], [cos(1), -2/3]), vector([-1, 2], [cos(2), -2/3]), vector([-1, 3], [cos(3), -2/3]), vector([-1, 4], [cos(4), -2/3]), vector([0, -4], [cos(4), -2/3]), vector([0, -3], [cos(3), -2/3]), vector([0, -2], [cos(2), -2/3]), vector([0, -1], [cos(1), -2/3]), vector([0, 0], [cos(0), -2/3]), vector([0, 1], [cos(1), -2/3]), vector([0, 2], [cos(2), -2/3]), vector([0, 3], [cos(3), -2/3]), vector([0, 4], [cos(4), -2/3]), vector([1, -4], [cos(4), -2/3]), vector([1, -3], [cos(3), -2/3]), vector([1, -2], [cos(2), -2/3]), vector([1, -1], [cos(1), -2/3]), vector([1, 0], [cos(0), -2/3]), vector([1, 1], [cos(1), -2/3]), vector([1, 2], [cos(2), -2/3]), vector([1, 3], [cos(3), -2/3]), vector([1, 4], [cos(4), -2/3]), vector([2, -4], [cos(4), -2/3]), vector([2, -3], [cos(3), -2/3]), vector([2, -2], [cos(2), -2/3]), vector([2, -1], [cos(1), -2/3]), vector([2, 0], [cos(0), -2/3]), vector([2, 1], [cos(1), -2/3]), vector([2, 2], [cos(2), -2/3]), vector([2, 3], [cos(3), -2/3]), vector([2, 4], [cos(4), -2/3]), vector([3, -4], [cos(4), -2/3]), vector([3, -3], [cos(3), -2/3]), vector([3, -2], [cos(2), -2/3]), vector([3, -1], [cos(1), -2/3]), vector([3, 0], [cos(0), -2/3]), vector([3, 1], [cos(1), -2/3]), vector([3, 2], [cos(2), -2/3]), vector([3, 3], [cos(3), -2/3]), vector([3, 4], [cos(4), -2/3]), vector([4, -4], [cos(4), -2/3]), vector([4, -3], [cos(3), -2/3]), vector([4, -2], [cos(2), -2/3]), vector([4, -1], [cos(1), -2/3]), vector([4, 0], [cos(0), -2/3]), vector([4, 1], [cos(1), -2/3]), vector([4, 2], [cos(2), -2/3]), vector([4, 3], [cos(3), -2/3]), vector([4, 4], [cos(4), -2/3])
draw2d(color = pink, vector([-4, -4], [cos(4), -2/3]), vector([-4, -3], [cos(3), -2/3]), vector([-4, -2], [cos(2), -2/3]), vector([-4, -1], [cos(1), -2/3]), vector([-4, 0], [cos(0), -2/3]), vector([-4, 1], [cos(1), -2/3]), vector([-4, 2], [cos(2), -2/3]), vector([-4, 3], [cos(3), -2/3]), vector([-4, 4], [cos(4), -2/3]), vector([-3, -4], [cos(4), -2/3]), vector([-3, -3], [cos(3), -2/3]), vector([-3, -2], [cos(2), -2/3]), vector([-3, -1], [cos(1), -2/3]), vector([-3, 0], [cos(0), -2/3]), vector([-3, 1], [cos(1), -2/3]), vector([-3, 2], [cos(2), -2/3]), vector([-3, 3], [cos(3), -2/3]), vector([-3, 4], [cos(4), -2/3]), vector([-2, -4], [cos(4), -2/3]), vector([-2, -3], [cos(3), -2/3]), vector([-2, -2], [cos(2), -2/3]), vector([-2, -1], [cos(1), -2/3]), vector([-2, 0], [cos(0), -2/3]), vector([-2, 1], [cos(1), -2/3]), vector([-2, 2], [cos(2), -2/3]), vector([-2, 3], [cos(3), -2/3]), vector([-2, 4], [cos(4), -2/3]), vector([-1, -4], [cos(4), -2/3]), vector([-1, -3], [cos(3), -2/3]), vector([-1, -2], [cos(2), -2/3]), vector([-1, -1], [cos(1), -2/3]), vector([-1, 0], [cos(0), -2/3]), vector([-1, 1], [cos(1), -2/3]), vector([-1, 2], [cos(2), -2/3]), vector([-1, 3], [cos(3), -2/3]), vector([-1, 4], [cos(4), -2/3]), vector([0, -4], [cos(4), -2/3]), vector([0, -3], [cos(3), -2/3]), vector([0, -2], [cos(2), -2/3]), vector([0, -1], [cos(1), -2/3]), vector([0, 0], [cos(0), -2/3]), vector([0, 1], [cos(1), -2/3]), vector([0, 2], [cos(2), -2/3]), vector([0, 3], [cos(3), -2/3]), vector([0, 4], [cos(4), -2/3]), vector([1, -4], [cos(4), -2/3]), vector([1, -3], [cos(3), -2/3]), vector([1, -2], [cos(2), -2/3]), vector([1, -1], [cos(1), -2/3]), vector([1, 0], [cos(0), -2/3]), vector([1, 1], [cos(1), -2/3]), vector([1, 2], [cos(2), -2/3]), vector([1, 3], [cos(3), -2/3]), vector([1, 4], [cos(4), -2/3]), vector([2, -4], [cos(4), -2/3]), vector([2, -3], [cos(3), -2/3]), vector([2, -2], [cos(2), -2/3]), vector([2, -1], [cos(1), -2/3]), vector([2, 0], [cos(0), -2/3]), vector([2, 1], [cos(1), -2/3]), vector([2, 2], [cos(2), -2/3]), vector([2, 3], [cos(3), -2/3]), vector([2, 4], [cos(4), -2/3]), vector([3, -4], [cos(4), -2/3]), vector([3, -3], [cos(3), -2/3]), vector([3, -2], [cos(2), -2/3]), vector([3, -1], [cos(1), -2/3]), vector([3, 0], [cos(0), -2/3]), vector([3, 1], [cos(1), -2/3]), vector([3, 2], [cos(2), -2/3]), vector([3, 3], [cos(3), -2/3]), vector([3, 4], [cos(4), -2/3]), vector([4, -4], [cos(4), -2/3]), vector([4, -3], [cos(3), -2/3]), vector([4, -2], [cos(2), -2/3]), vector([4, -1], [cos(1), -2/3]), vector([4, 0], [cos(0), -2/3]), vector([4, 1], [cos(1), -2/3]), vector([4, 2], [cos(2), -2/3]), vector([4, 3], [cos(3), -2/3]), vector([4, 4], [cos(4), -2/3])
```

El entorno de trabajo de WxMaxima es sencillo, sin embargo, al momento de realizar los gráficos marcaba una serie de errores los cuales generalmente tenían que ver con el rango con el que se estaba trabajando. Me resultó mucho más sencillo al trabajar con campos vectoriales ya que sólo se define la función además de que ya cuenta con un comando determinado.