```
>
```

The Gradient and Directional Derivatives

```
> restart:with(plots):with(plottools):with(VectorCalculus):
> setoptions3d(axes=NORMAL,labels=["x","y","z"],orientation=[20,
  70]);
> BasisFormat(false);
```

true

Two Dimensions

We begin by focusing on the function z = f(x, y) = $x^2 + 3\sin(y)$, the point (2, $\frac{\pi}{2}$, 7), and the vector **v** = <4,3>.

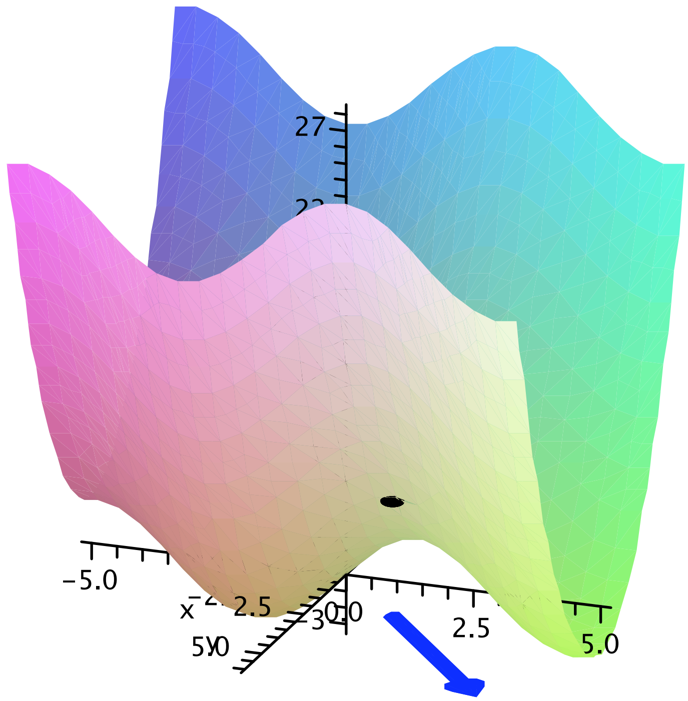
```
> z:=x^2+3*sin(y);
  v:=<4,3>;
```

$$z := x^2 + 3\sin(y)$$

$$v := \begin{bmatrix} 4 \\ 3 \end{bmatrix}$$

We plot the function z = f(x, y) = $x^2 + 3\sin(y)$, the point (2, $\frac{\pi}{2}$, 7), and the vector **v** = 4**i** + 3**j**.

```
> p0:=plot3d(z,x=-5..5,y=-5..5,style=patchnogrid):
  p1:=sphere([2,Pi/2,7],.2):
  p2:=PlotVector(RootedVector(root=[2,Pi/2,0],[4,3,0]),width=.4,
  head_width=.8,head_length=.8,color=blue):
  display(p0,p1,p2);
```



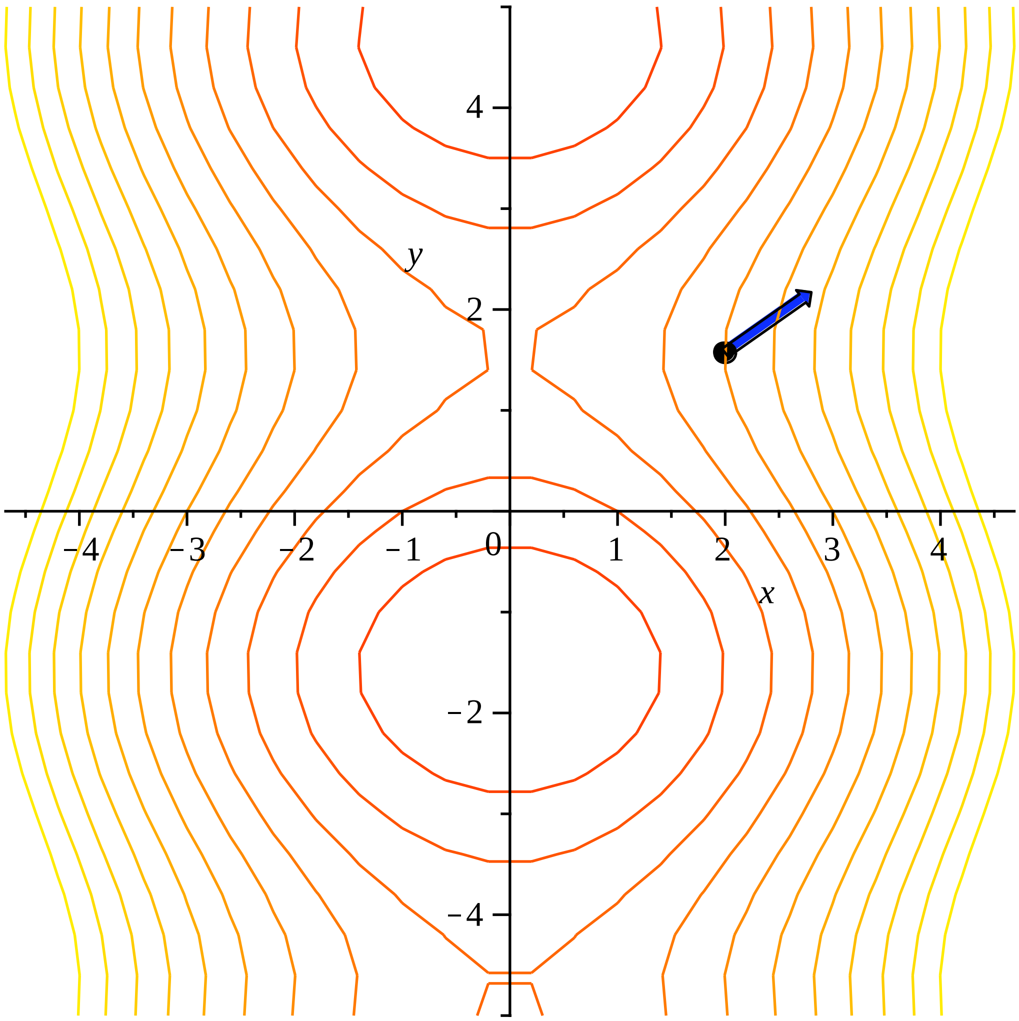
We divide **v** by its norm to transform it to a unit vector in the same direction.

```
> u:=v/Norm(v,2);
```

$$u := \begin{bmatrix} \frac{4}{5} \\ \frac{3}{5} \end{bmatrix}$$

We are interested in the rate of change of f at the point $(2, \frac{\pi}{2})$ in the direction of the unit vector **u**. We show that unit vector on the contourplot below that includes the contour z = 7 which goes through the point $(2, \frac{\pi}{2})$.

```
> p0:=contourplot(z,x=-5..5,y=-5..5,contours=[seq(3+2*i,i=-4..8)])
  :
  p1:=disk([2,Pi/2],.1):
  p2:=PlotVector(RootedVector(root=[2,Pi/2],[4/5,3/5]),width=.1,
  head_width=.2,head_length=.1,color=blue):
  display(p0,p1,p2);
```



We compute the directional derivative for a point (*x*, *y*) using DirectionalDiff. The third argument **[x, y]** is to indicate our coordinate system

```
> directderiv:=DirectionalDiff(z,u,[x,y]);
```

$$directderiv := \frac{8}{5}x + \frac{9}{5}\cos(y)$$

Now we evaluate it at the point $(2,\frac{\pi}{2})$.

```
> dd:=eval(directderiv,[x=2,y=Pi/2]);
```

$$dd := \frac{16}{5}$$

Now that we have the directional derivative at the point $(2, \frac{\pi}{2})$, we use the Gradient command to compute the gradient of f. The **[x,y]** again refers to our coordinates.

```
> gradf:=Gradient(z,[x,y]);
```

$$gradf := \begin{bmatrix} 2x \\ 3\cos(y) \end{bmatrix}$$

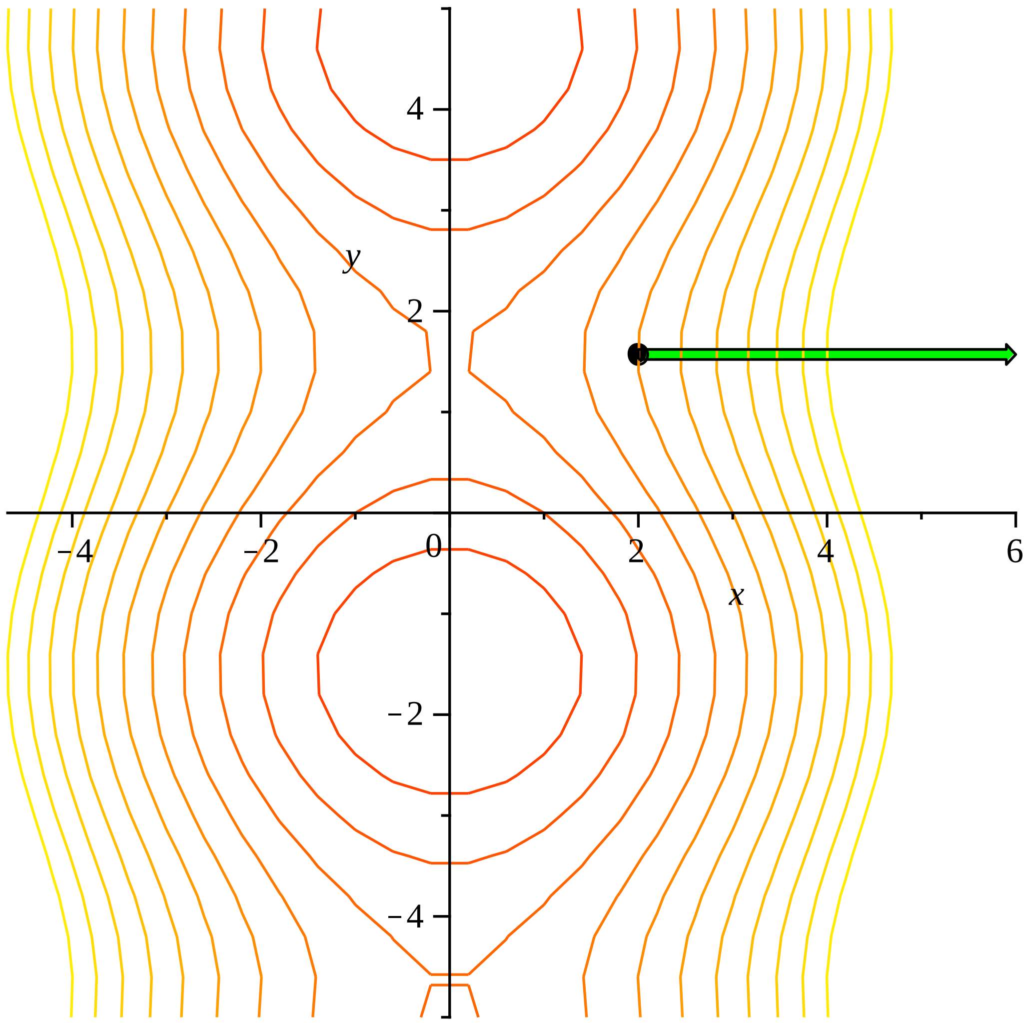
We evaluate the gradient at the point (2, Pi/2).

```
> gf:=eval(gradf,[x=2,y=Pi/2]);
```

$$gf := \begin{bmatrix} 4 \\ 0 \end{bmatrix}$$

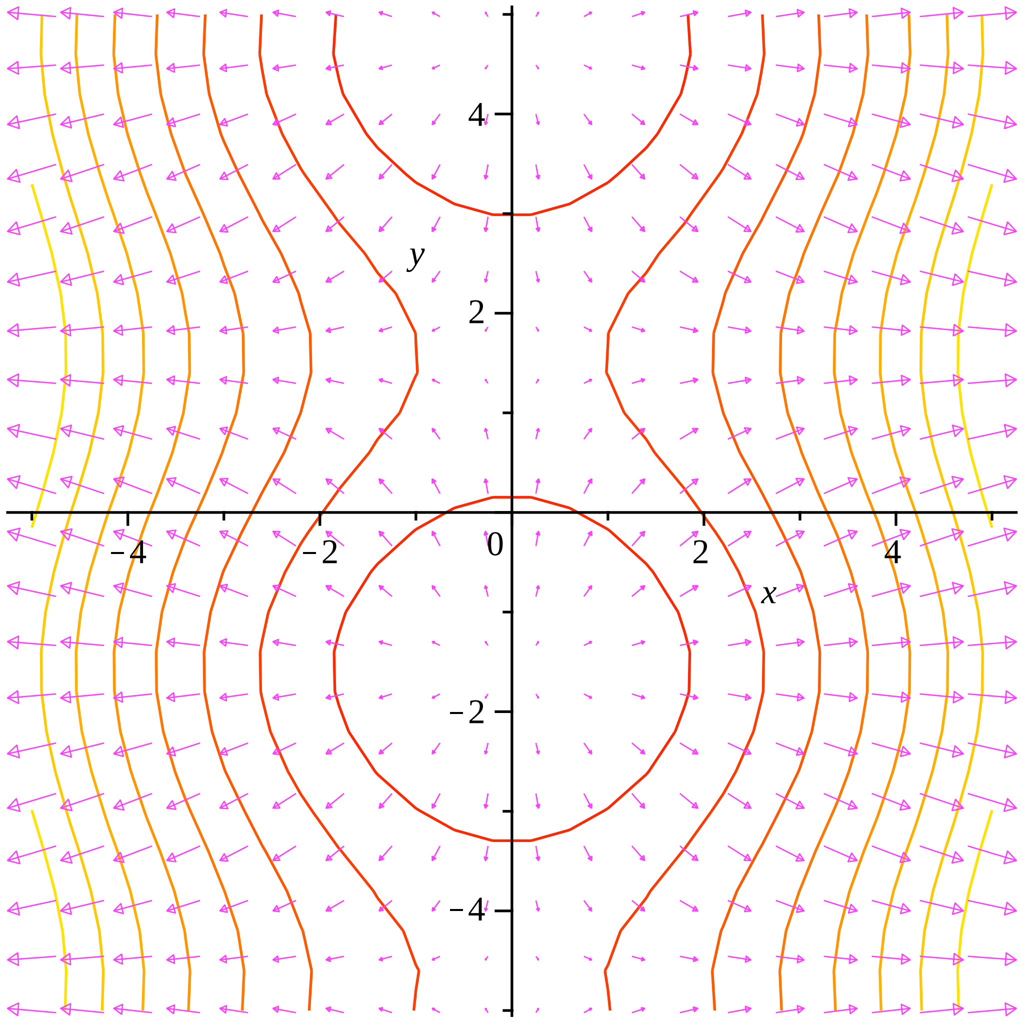
We plot the gradient vector on our contourplot.

```
> p0:=contourplot(z,x=-5..5,y=-5..5,contours=[seq(3+2*i,i=-4..8)])
  :
  p1:=disk([2,Pi/2],.1):
  p2:=PlotVector(RootedVector(root=[2,Pi/2],[4,0]),width=.1,
  head_width=.2,head_length=.1,color=green):
  display(p0,p1,p2);
```



We see that the gradient vector at the point $(2, \frac{\pi}{2})$ is orthogonal to the contour through the point. We next overlay a gradientplot over the contourplot. This places arrows in the direction of the gradient at a selection of points, with arrow lengths indicating relative magnitude.

```
> p0:=contourplot(z,x=-5..5,y=-5..5):
  p1:=gradplot( z, x=-5..5,y=-5..5,arrows=slim,color=magenta):
  display(p0,p1);
```



We see that all gradients seem to be orthogonal to the various contours, and that their magnitudes are greater where the contours are more closely bunched, indicating a more rapid rate of change.

We next plot a field of vectors that show the direction of the various contours. At each point in the plane, a direction vector is $[1, \frac{dy}{dx}]$, or equivalently (having the same direction), [dx, dy]. We begin by computing the implicit derivative $\frac{dy}{dx}$ for the equation $z = 0$ by using the command implicitdiff.

```
> imp:=implicitdiff(z=0,y,x);
```

$$imp := -\frac{2}{3}\,\frac{x}{\cos(y)}$$

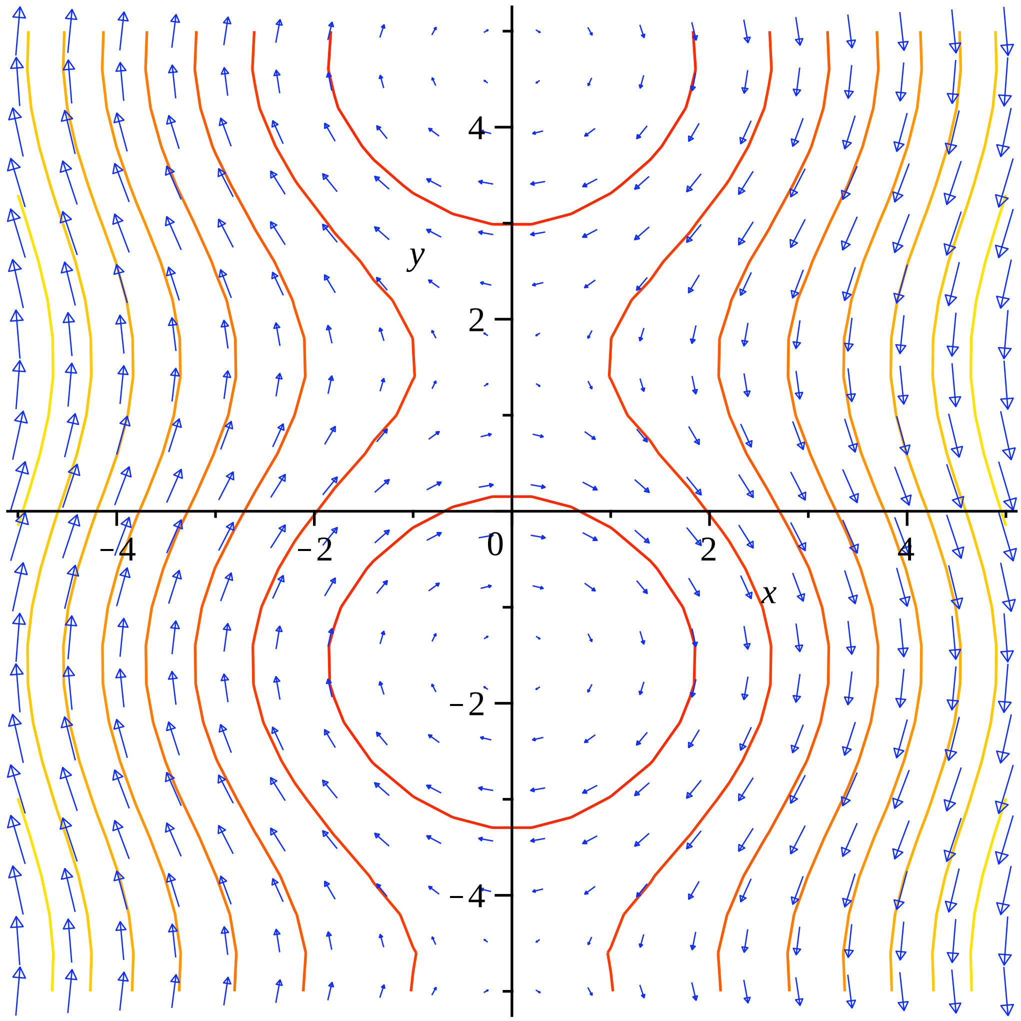
We use the VectorField command to form the vector field.

```
> vf:=VectorField(<denom(imp),numer(imp)>,'cartesian'[x,y]);
```

$$vf := \begin{bmatrix} 3\cos(y) \\ -2\,x \end{bmatrix}$$

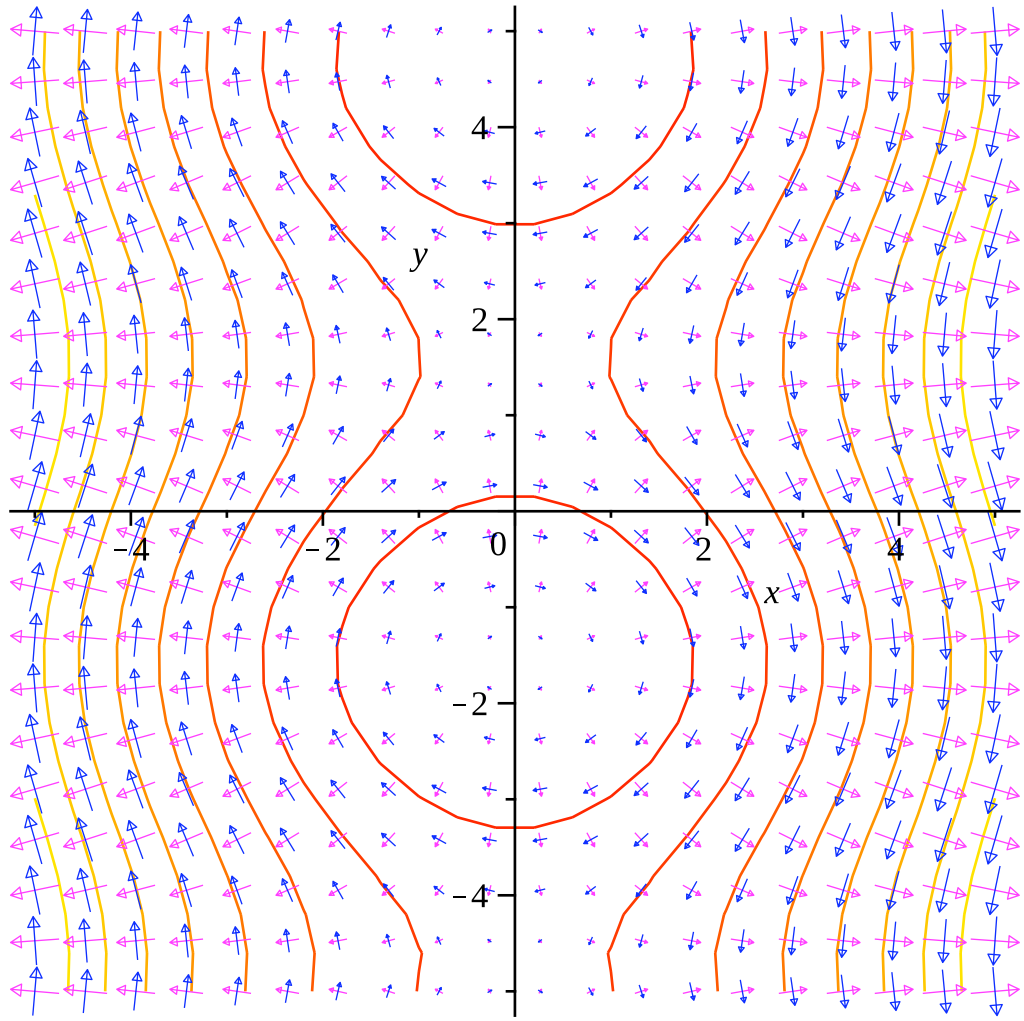
Then we use the fieldplot command to plot the field of vectors along with the contourplot.

```
> p0:=contourplot(z,x=-5..5,y=-5..5):
  p2:=fieldplot(vf,x=-5..5,y=-5..5,arrows=slim,color=blue):
  display(p0,p2);
```



Adding in the field of gradient vectors, we see even more clearly that they are orthogonal to the direction of the contours.

```
> display(p0,p1,p2);
```



To verify what we see, we take the dot product of the gradient with the direction vector at each point. We need the evalVF command to evaluate the vector fields at the points (x,y).

```
> dp:=DotProduct(evalVF(gradf,<x,y>),evalVF(vf,<x,y>));
```

$$dp := 0$$

Finally, we compute the directional derivative again, this time taking the dot product of the gradient at the point $(2, \frac{\pi}{2})$ with the unit direction vector.

```
> directderiv:=DotProduct(evalVF(gf,<x,y>),u);
```

$$directderiv := \frac{16}{5}$$

```
>
```

Three Dimensions

```
> restart:with(plots):with(plottools):with(VectorCalculus):
> setoptions3d(axes=NORMAL,labels=["x","y","z"],orientation=[20,
  70]);
> BasisFormat(false);
```

$$true$$

We begin by focusing on the function w = f(x,y,z) = $z \ln(x^2 + y^2)$, the point (-1, 1, 0), and the direction vector **v** = <2,3,5>.

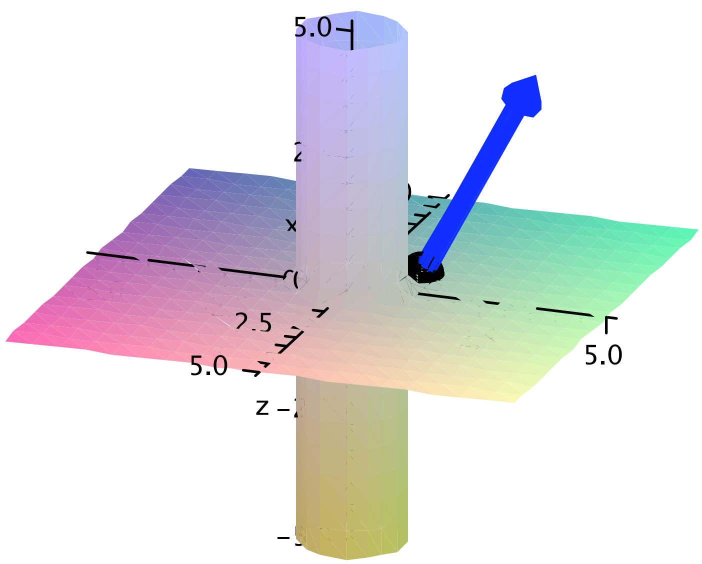
```
> w:=z*ln(x^2+y^2);
  v:=<2,3,5>;
```

$$w := z \ln\left(x^2 + y^2\right)$$

$$v := \begin{bmatrix} 2 \\ 3 \\ 5 \end{bmatrix}$$

We plot the level surface w = f(x,y,z) = 0 along with the point and direction vector.

```
> p0:=implicitplot3d(w=0,x=-5..5,y=-5..5,z=-5..5,style=
  patchnogrid,numpoints=5000):
  p1:=sphere([-1,1,0],.4):
  p2:=PlotVector(RootedVector(root=[-1,1,0],[2,3,5]),width=.4,
  head_width=.8,head_length=.8,color=blue):
  display(p0,p1,p2);
```



We transform the direction vector to a unit vector.

```
> u:=v/Norm(v,2);
```

$$u := \begin{bmatrix} \frac{1}{19}\sqrt{38} \\ \frac{3}{38}\sqrt{38} \\ \frac{5}{38}\sqrt{38} \end{bmatrix}$$

We are interested in the rate of change of f at the point (-1, 1, 0) in the direction of the unit vector **u**. We compute the directional derivative.

```
> directderiv:=DirectionalDiff(w,u,[x,y,z]);
```

$$directderiv := \frac{2}{19}\,\frac{z\,x\sqrt{38}}{x^2+y^2} + \frac{3}{19}\,\frac{z\,y\sqrt{38}}{x^2+y^2} + \frac{5}{38}\ln\left(x^2+y^2\right)\sqrt{38}$$

Now we evaluate it at the point (-1,1,0).

```
> dd:=eval(directderiv,[x=-1,y=1,z=0]);
```

$$dd := \frac{5}{38}\ln(2)\sqrt{38}$$

Now that we have the directional derivative at the point (-1, 1, 0), we use the Gradient command to compute the gradient of f. The [x,y,z] refers to our coordinates.

```
> gradf:=Gradient(w,[x,y,z]);
```

$$gradf := \begin{bmatrix} \frac{2\,z\,x}{x^2+y^2} \\ \frac{2\,z\,y}{x^2+y^2} \\ \ln\left(x^2+y^2\right) \end{bmatrix}$$

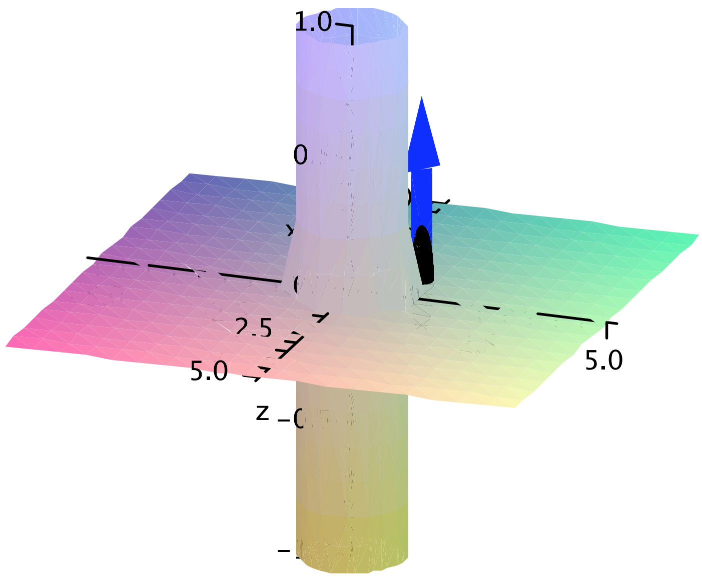
We evaluate the gradient at the point (-1, 1, 0).

```
> gf:=eval(gradf,[x=-1,y=1,z=0]);
```

$$gf := \begin{bmatrix} 0 \\ 0 \\ \ln(2) \end{bmatrix}$$

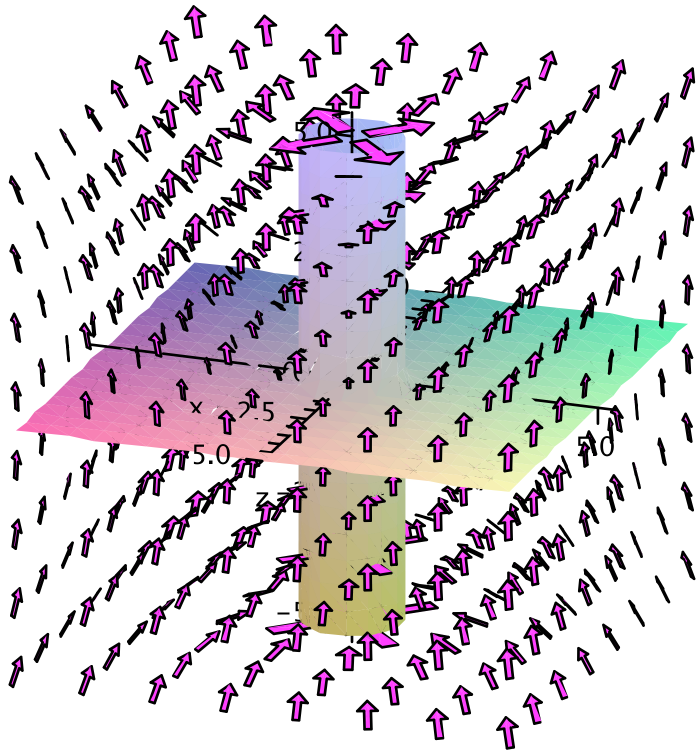
We plot the gradient vector on our level surface

```
> p0:=implicitplot3d(w=0,x=-5..5,y=-5..5,z=-5..5,style=
  patchnogrid,numpoints=5000):
  p1:=sphere([-1,1,0],.2):
  p2:=arrow([-1,1,0],gf,.4,.8,.4,color=blue):
  display(p0,p1,p2,view=-1..1);
```



We see that the gradient vector is orthogonal to the level surface through the point. We next overlay a gradientplot over the level surface. This places arrows in the direction of the gradient at a selection of points, with arrow lengths indicating relative magnitude.

```
> p1:=gradplot3d( w, x=-5..5,y=-5..5,z=-5..5,arrows=THICK,color=
  magenta):
  display(p0,p1);
```



We see that the gradients with base on our level surface seem to be orthogonal to it. Finally, we compute the directional derivative again, this time taking the dot product of the gradient at the point (-1, 1, 0) with the unit direction vector.

```
> directderiv:=DotProduct(evalVF(gf,<x,y,z>),u);
```

$$directderiv := \frac{5}{38} \ln(2) \sqrt{38}$$

```
>
>
```