Vector Fields

In [2]:
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
plt.rcParams["figure.figsize"] = (8,8)
plt.rcParams.update({'font.size': 14})
```

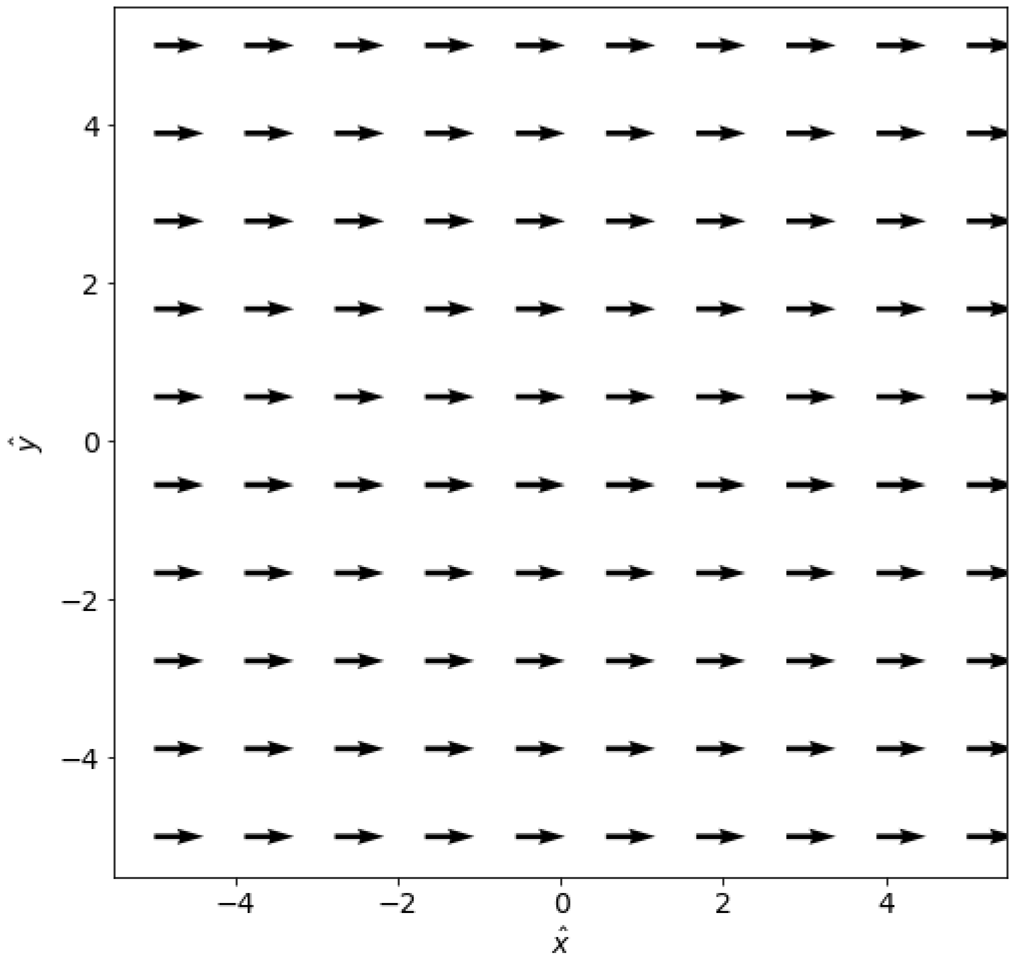
Some example vector fields

If we want to plot a vector field that has horizontal vectors (e.g., only point along $\hat{x}$) what would the components a and b of this vector field, $\vec{v}$ look like?

$$\vec{v} = a\hat{x} + b\hat{y}$$

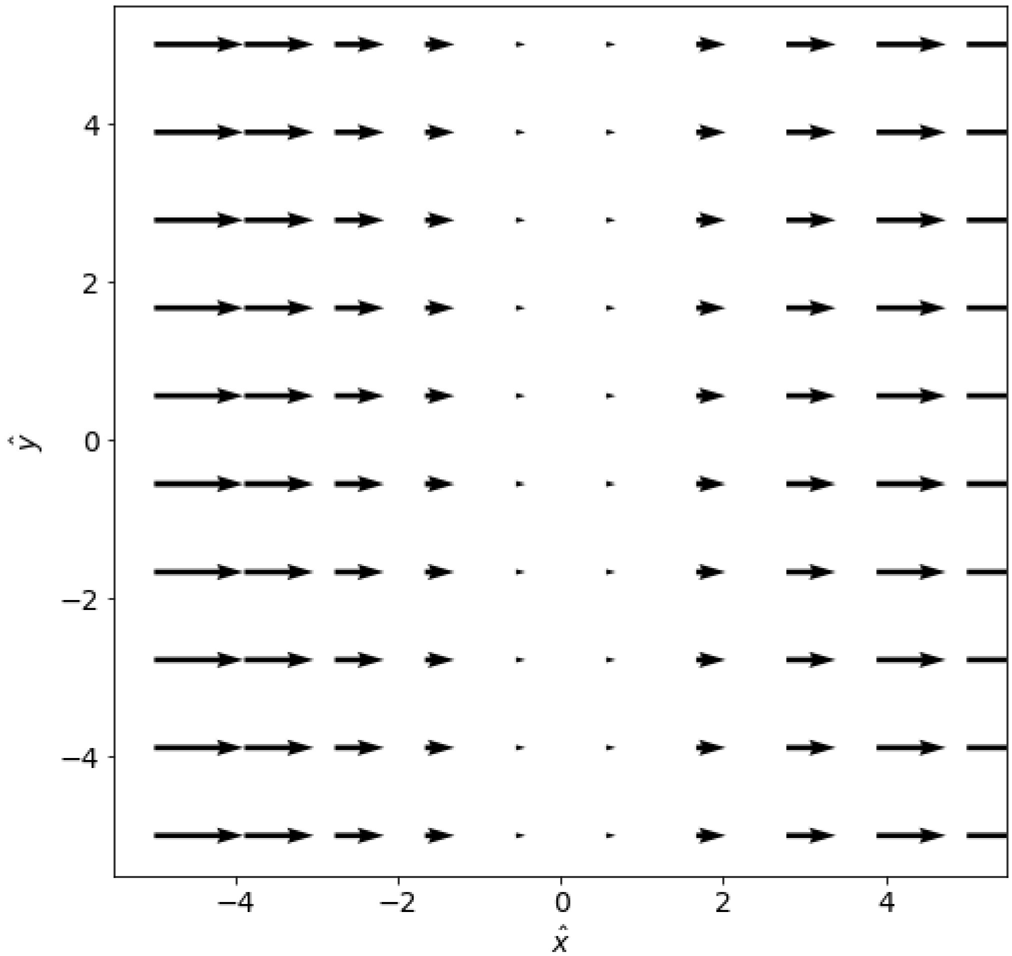
Let's give your answer a try.

In [3]:
```
#Define some cartesian x and y coordinates.
x,y = np.meshgrid(np.linspace(-5,5,10),np.linspace(-5,5,10))

vx = np.sqrt(x**2+y**2)/np.sqrt(x**2+y**2) #fill in an x length for the vector
vy = 0/np.sqrt(x**2+y**2) #fill in an y length for the vectors

plt.quiver(x,y,vx,vy)
plt.xlabel(r'$\hat{x}$')
plt.ylabel(r'$\hat{y}$')
plt.show()
```



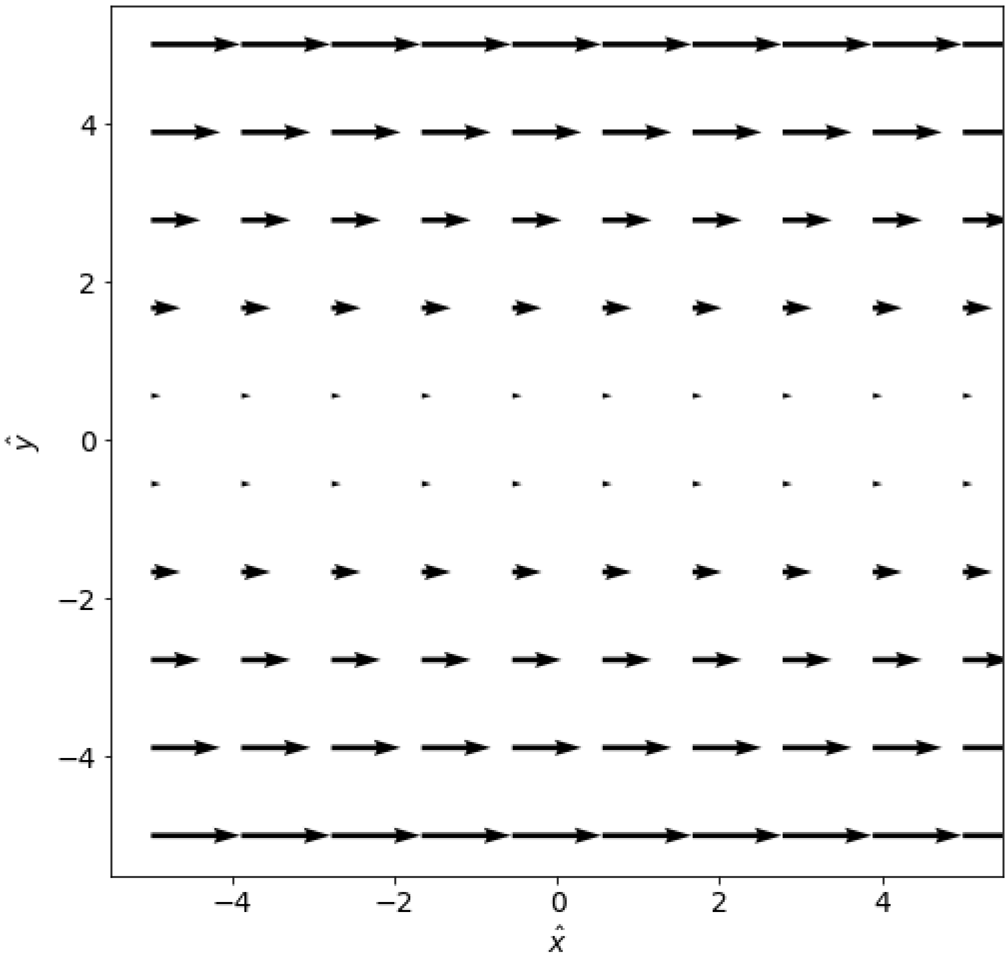
Suppose we want the length of the vectors to continue to point along $\hat{x}$, but we want the length to increase as the distance along the $\hat{x}$ axis?

In [4]:
```
vx = np.abs(x) #fill in an x length for the vectors
vy = 0/np.sqrt(x**2+y**2) #fill in an y length for the vectors

plt.quiver(x,y,vx,vy)
plt.xlabel(r'$\hat{x}$')
plt.ylabel(r'$\hat{y}$')
plt.show()
```



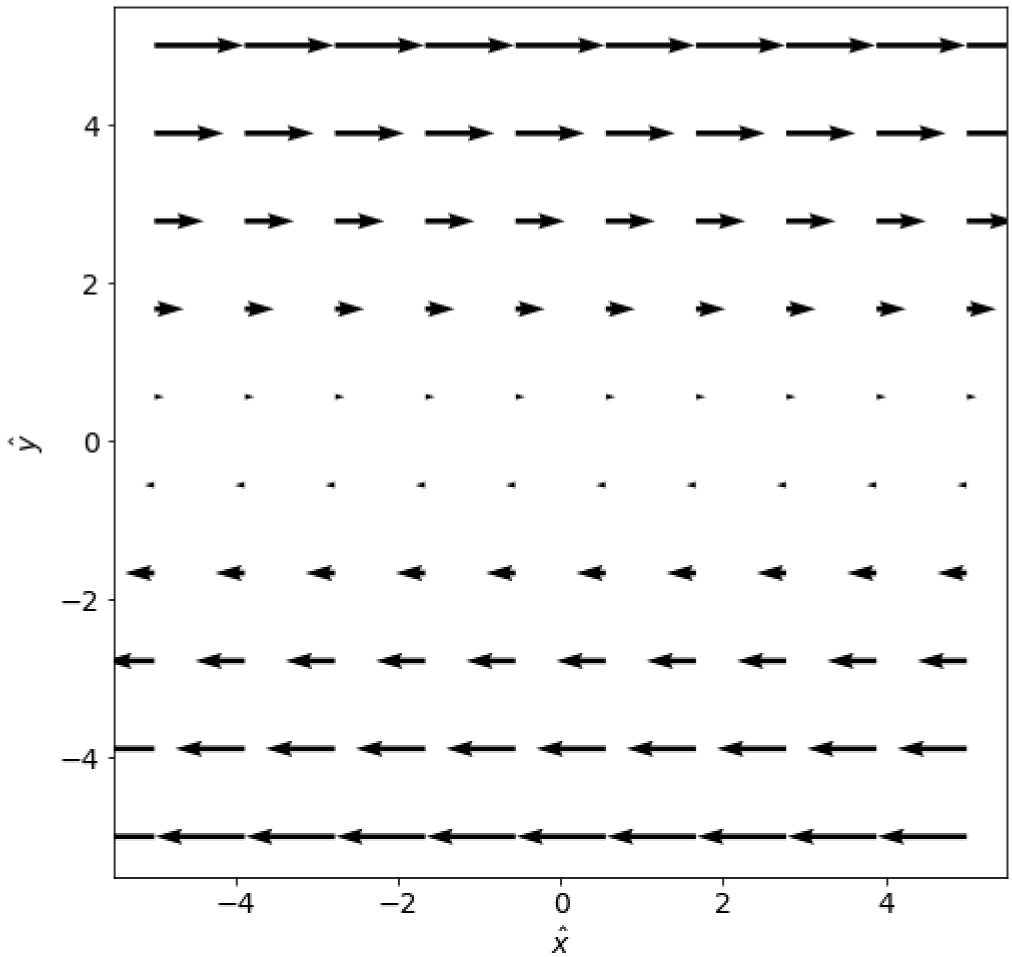
Now, suppose we want the vector lengths to point along $\hat{x}$ but increase in length based upon distance along the y-axis.

In [5]:
```
vx = np.abs(y) #fill in an x length for the vectors
vy = 0/np.sqrt(x**2+y**2) #fill in an y length for the vectors

plt.quiver(x,y,vx,vy)
plt.xlabel(r'$\hat{x}$')
plt.ylabel(r'$\hat{y}$')
plt.show()
```



Now, suppose we want the vector lengths to point along $\hat{x}$ but increase in length based upon distance along the y-axis and switch direction based upon whether y is positive or negative.

```
In [22]: vx = y #fill in an x length for the vectors
         vy = 0/np.sqrt(x**2+y**2) #fill in an y length for the vectors

         plt.quiver(x,y,vx,vy)
         plt.xlabel(r'$\hat{x}$')
         plt.ylabel(r'$\hat{y}$')
         #plt.hlines(y=0, xmin=-5, xmax=5, color='black',linestyles='dashed')
         #plt.vlines(x=0, ymin=-5, ymax=5, color='black',linestyles='dashed')
         plt.show()
```



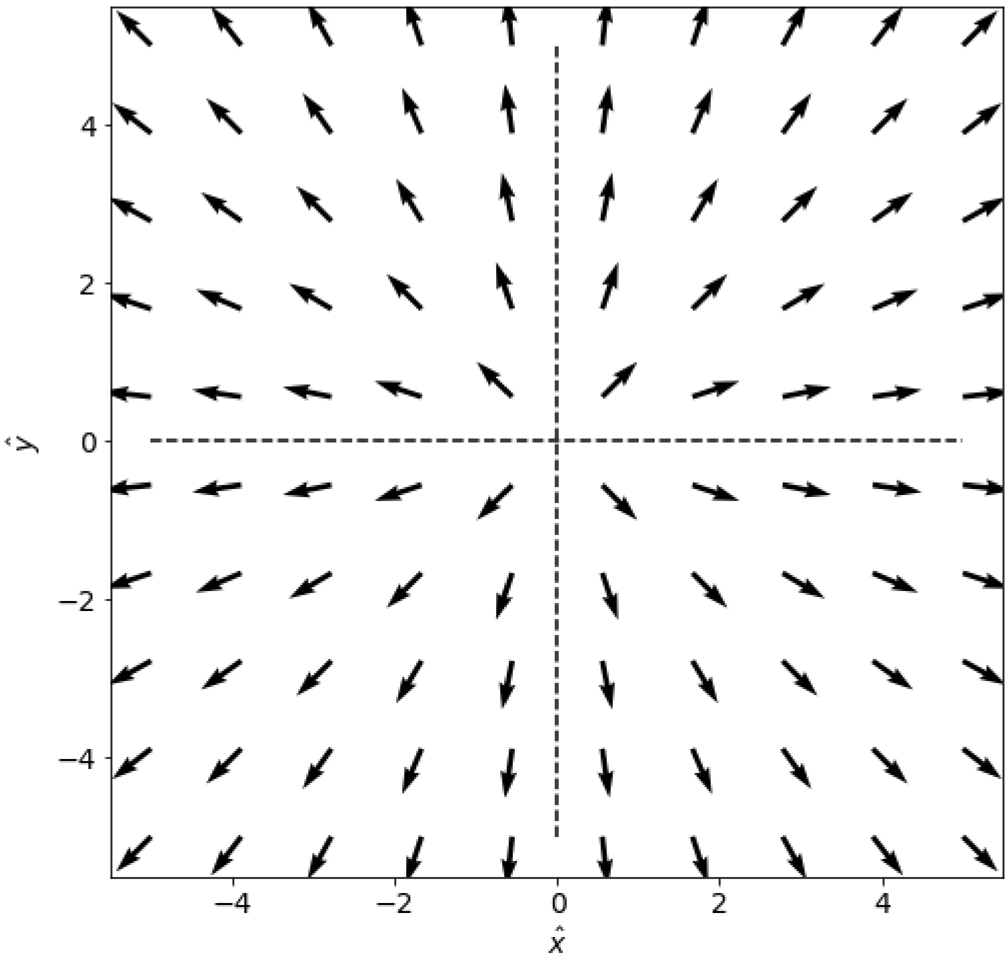
Radial Unit Vectors

We can plot radial unit vectors in 2D. We know that a radial vector points from the origin towards the edge of a circle and therefore has x and y component defined as follows.

$$\hat{r} = \cos\phi\hat{x} + \sin\phi\hat{y} \left|\hat{r}\right|_x = \frac{x}{\sqrt{x^2+y^2}} = \cos\phi \left|\hat{r}\right|_y = \frac{y}{\sqrt{x^2+y^2}} = \sin\phi$$

What is the range of values for $\left|\hat{r}\right|_x$ and $\left|\hat{r}\right|_y$?

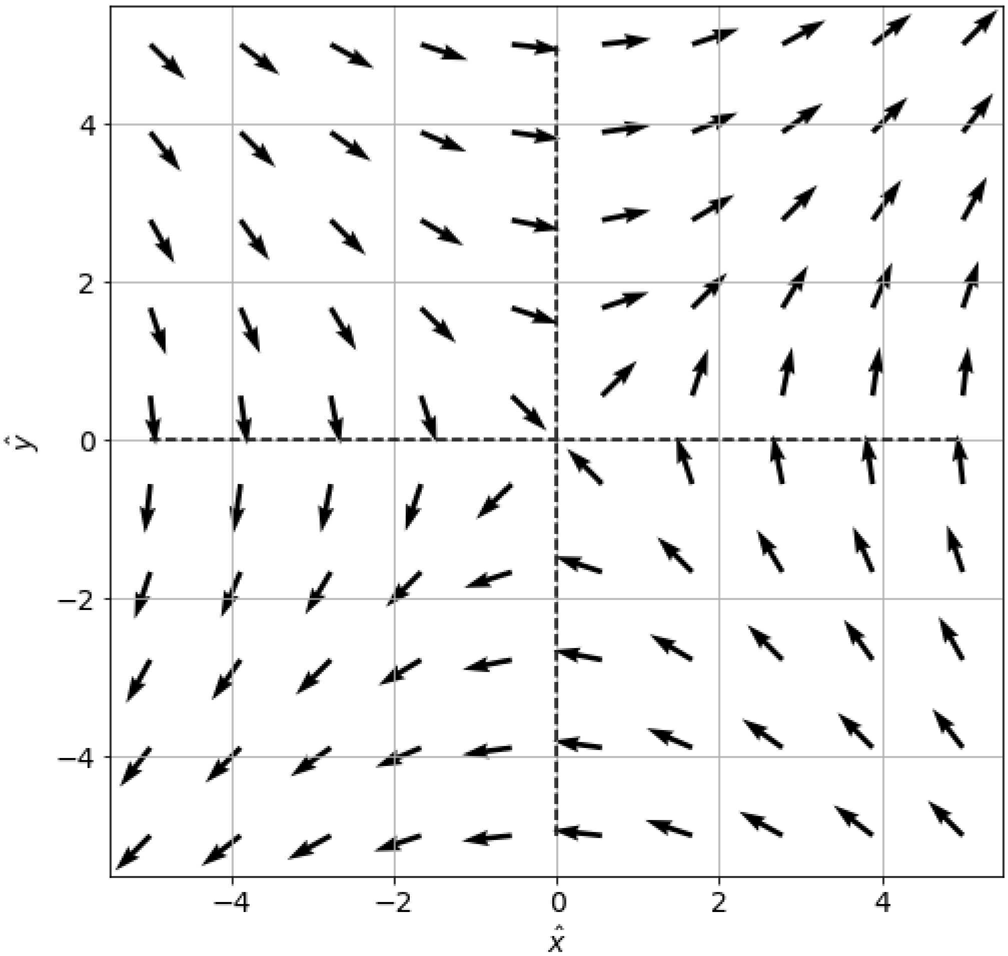
In [20]:
```python
xlen = x/np.sqrt(x**2 + y**2)
ylen = y/np.sqrt(x**2 + y**2)

plt.quiver(x,y,xlen,ylen)
plt.xlabel(r'$\hat{x}$')
plt.ylabel(r'$\hat{y}$')
plt.hlines(y=0, xmin=-5, xmax=5, color='black',linestyles='dashed')
plt.vlines(x=0, ymin=-5, ymax=5, color='black',linestyles='dashed')
plt.show()
```



A very slight change, has a dramatic effect on the vector field. For example, switch plotting such that x values are on the $\hat{y}$ axis and vice versa.

$$\hat{v} = \sin\phi\hat{x} + \cos\phi\hat{y} \left|\hat{v}\right|_x = \frac{y}{\sqrt{x^2+y^2}} = \sin\phi \left|\hat{v}\right|_y = \frac{x}{\sqrt{x^2+y^2}} = \cos\phi$$

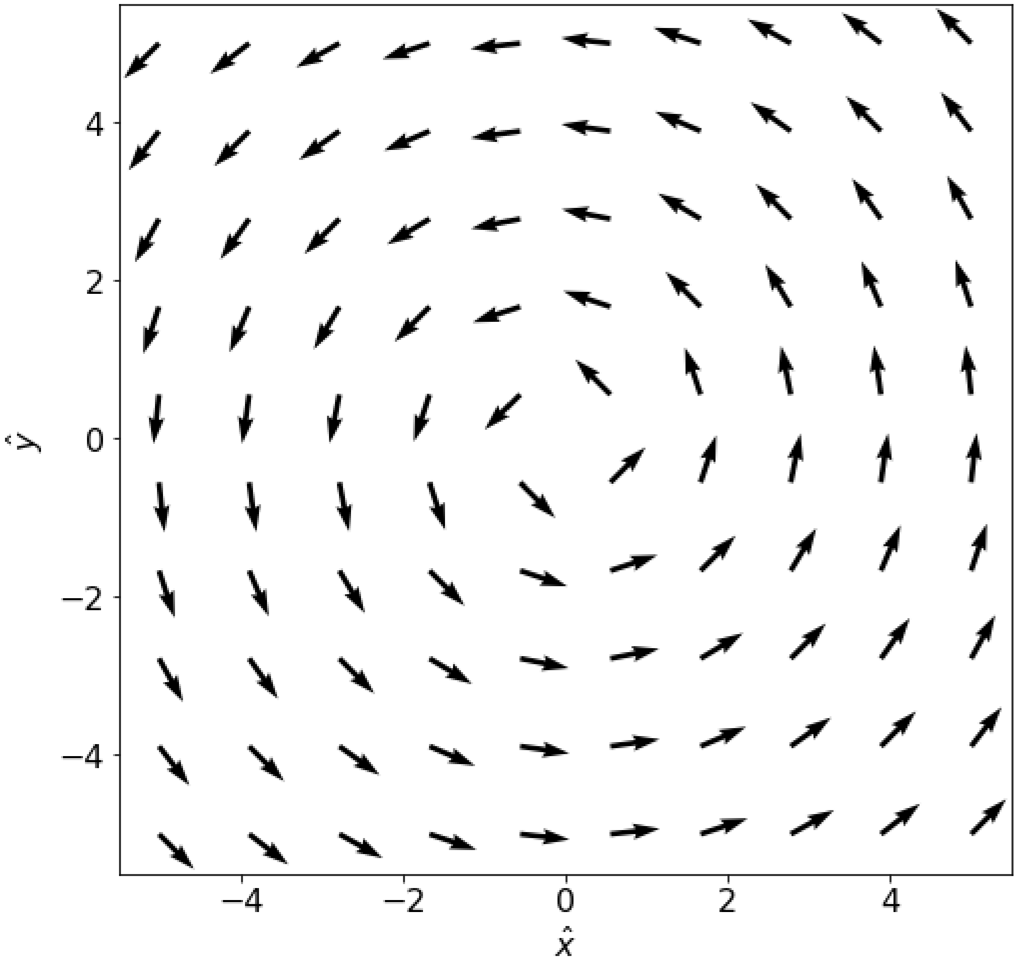
In [23]:
```python
plt.quiver(x,y,ylen,xlen)
plt.xlabel(r'$\hat{x}$')
plt.ylabel(r'$\hat{y}$')
plt.grid(True)
#plt.hlines(y=0, xmin=-5, xmax=5, color='black',linestyles='dashed')
#plt.vlines(x=0, ymin=-5, ymax=5, color='black',linestyles='dashed')
plt.show()
```



Or plot $-y$ values on the $\hat{x}$ axis with x values on the $\hat{y}$ axis.

$$\hat{v} = -\sin\phi\hat{x} + \cos\phi\hat{y} \left|\hat{v}\right|_x = \frac{-y}{\sqrt{x^2+y^2}} = -\sin\phi \left|\hat{v}\right|_y = \frac{x}{\sqrt{x^2+y^2}} = \cos\phi$$

In [8]:
```
plt.quiver(x,y,-ylen,xlen)
plt.xlabel(r'$\hat{x}$')
plt.ylabel(r'$\hat{y}$')
plt.show()
```



The definition of $\hat{\phi}$ is

$$\hat{\phi} = -\sin\phi\hat{x} + \cos\phi\hat{y}$$

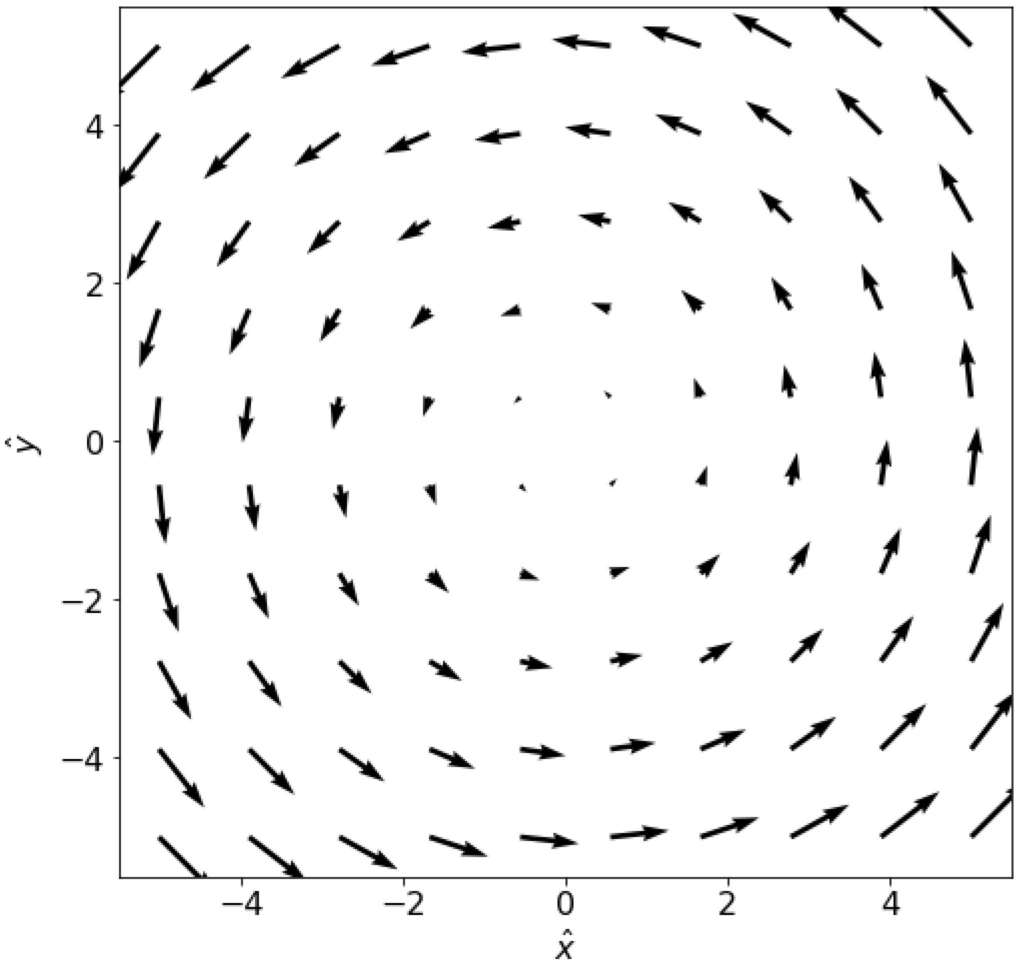
Therefore, in the last example, we took a radial vector

$$\hat{r} = \cos\phi\hat{x} + \sin\phi\hat{y}$$

and changed it to an angular one. It is still a unit vector field because the components are normalized. Make this rotational vector field have vectors that increase in length based on their radial distance from the origin.

$$\hat{v} = \hat{\phi}$$

In [9]:
```
r = np.sqrt(x**2+y**2)
plt.quiver(x,y,-r*ylen,r*xlen)
plt.xlabel(r'$\hat{x}$')
plt.ylabel(r'$\hat{y}$')
plt.show()
```



How would you write this vector field in cylindrical coordinate notation?

$$\vec{v} = r\hat{\phi}$$

Plot the vector field

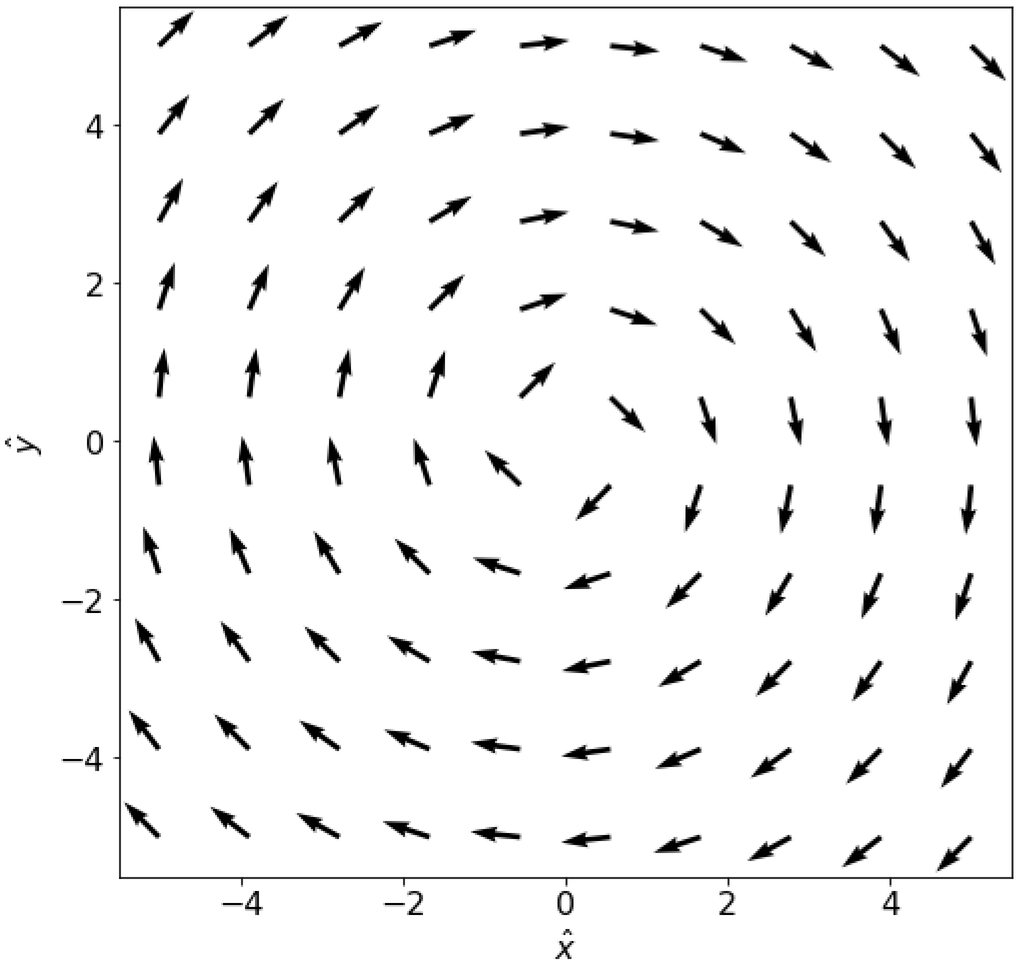
$$\hat{v} = -\hat{\phi}$$

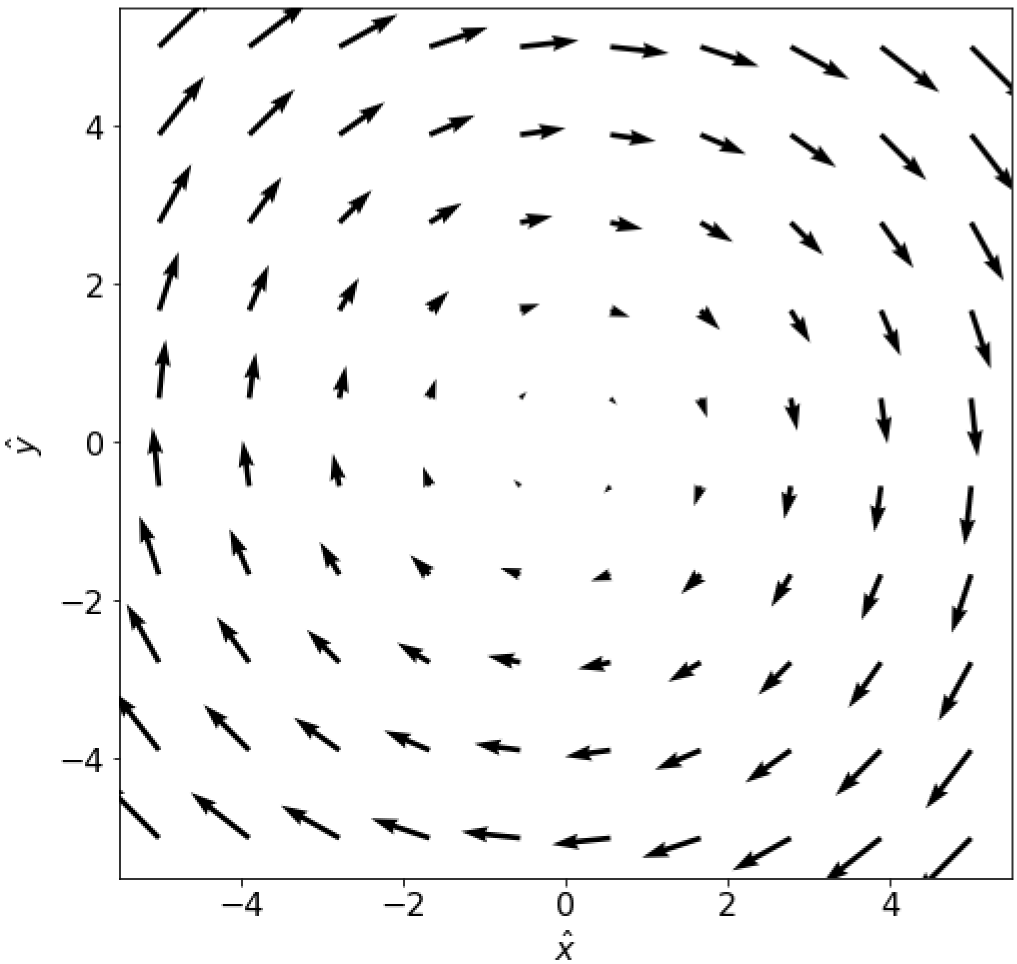
and then plot the vector field

$$\vec{v} = -r\hat{\phi}$$

Describe the change in the vector field. Include in your description the cartesian coordinate changes made from the original radial vector.

In [10]:
```
plt.quiver(x,y,ylen,-xlen)
plt.xlabel(r'$\hat{x}$')
plt.ylabel(r'$\hat{y}$')
plt.show()

plt.quiver(x,y,r*ylen,-r*xlen)
plt.xlabel(r'$\hat{x}$')
plt.ylabel(r'$\hat{y}$')
plt.show()
```





In [ ]: