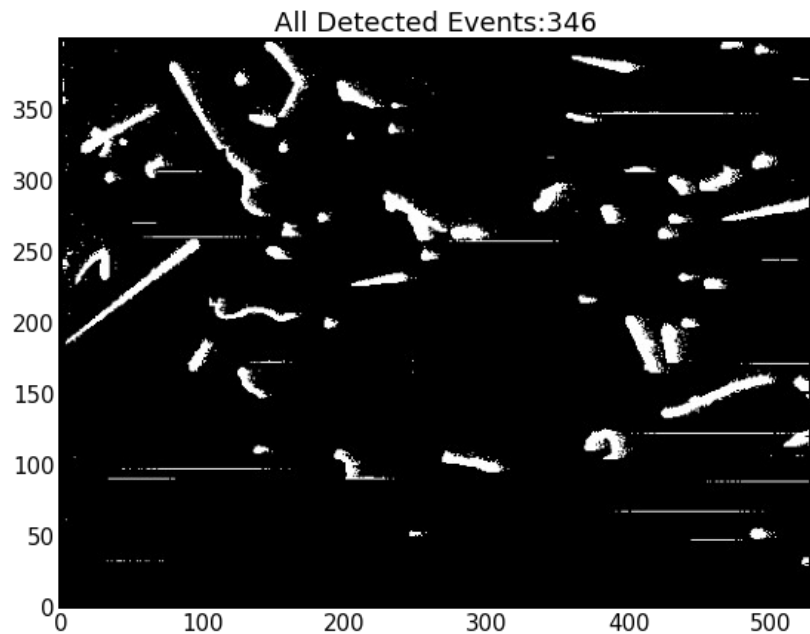


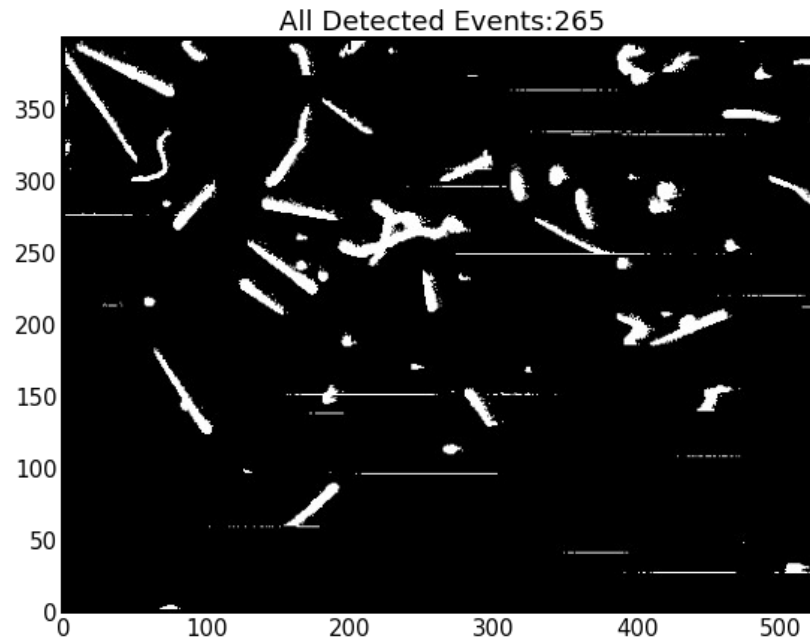
AVANCES DE TESIS

SEMANA 30/MAYO/2025

Para las imágenes del ICN se optó por utilizar 13σ y abajo se muestran ejemplos de como se visualizan una imagen típica para las extensiones 1 y 2.

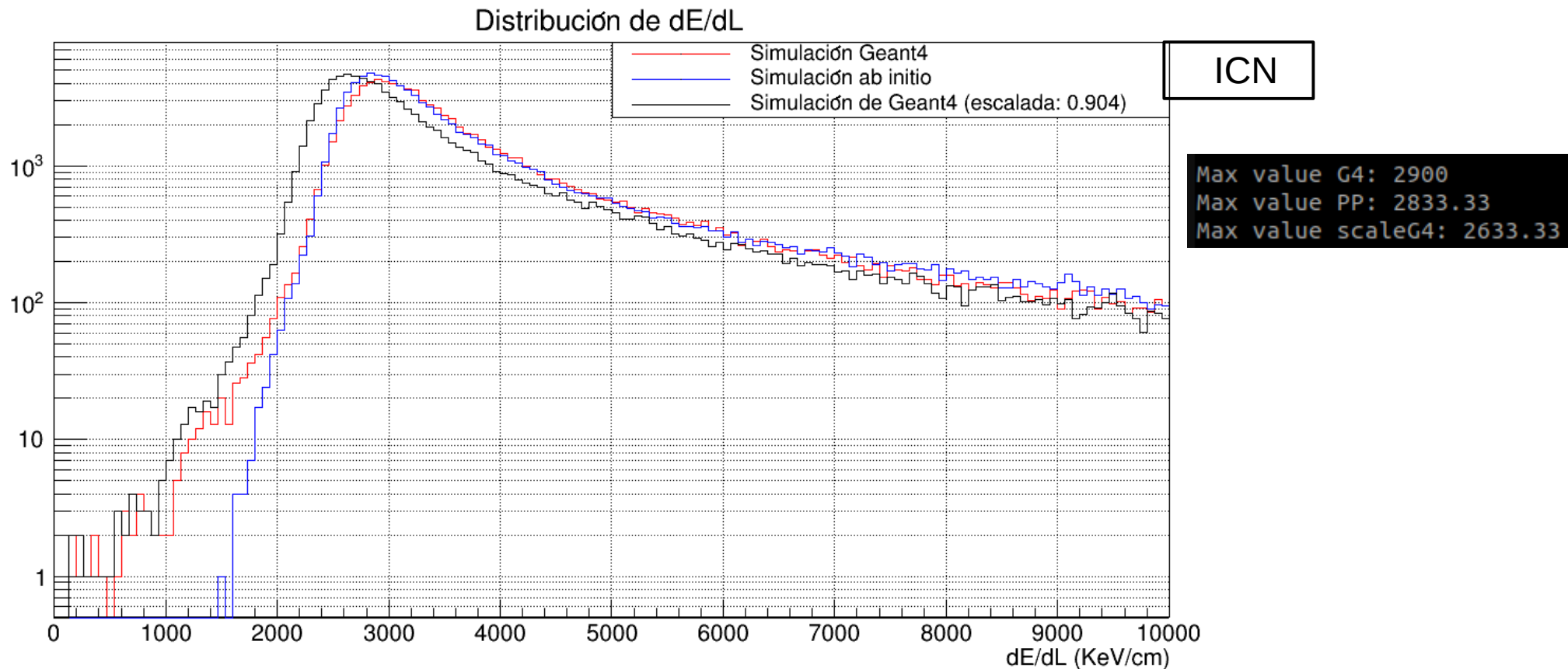


Extensión 1

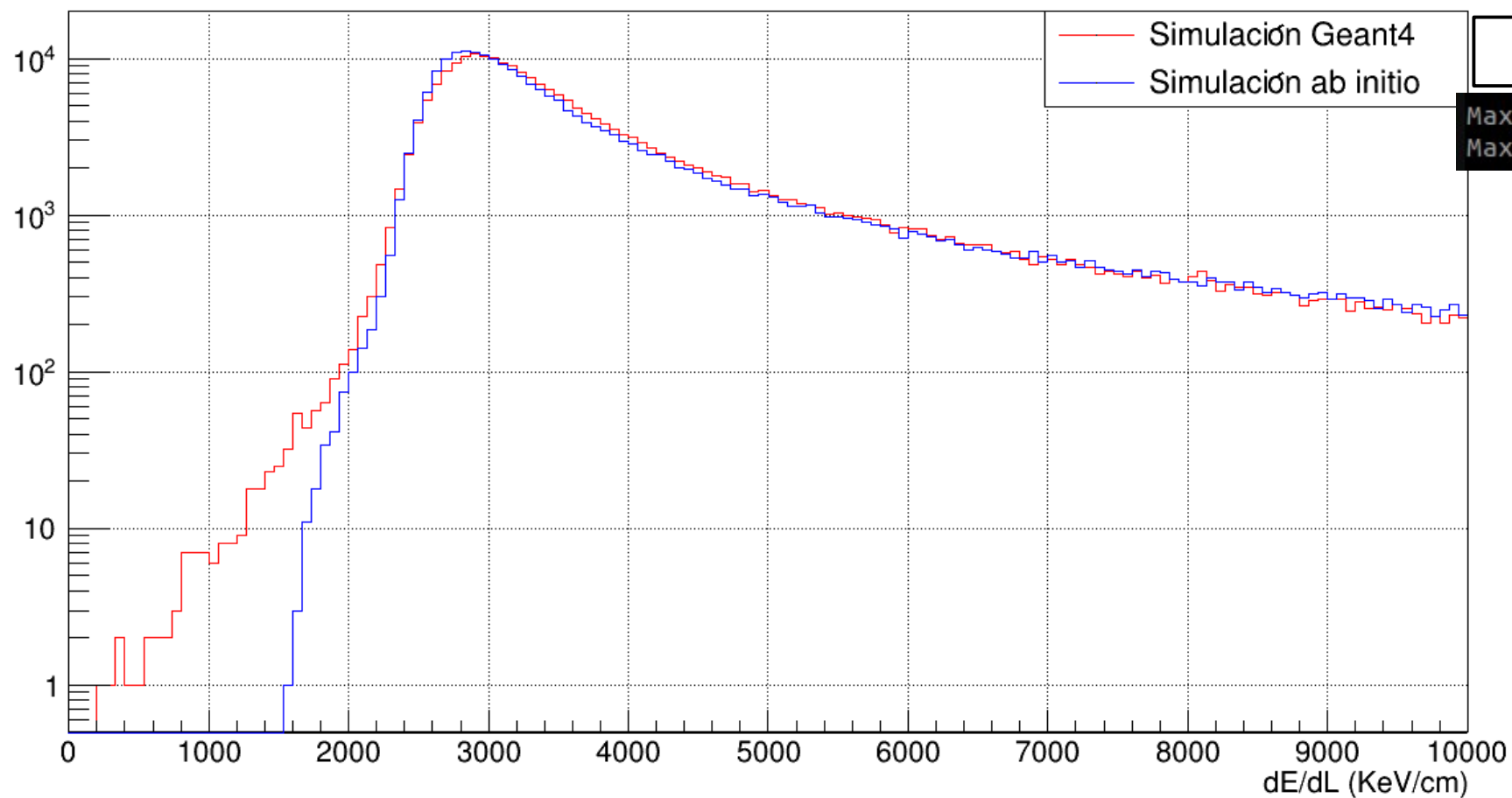


Extensión 2

Se obtuvo la distribución de dE/dL para ambas simulaciones las cuales se muestran abajo, claramente no son del todo iguales.



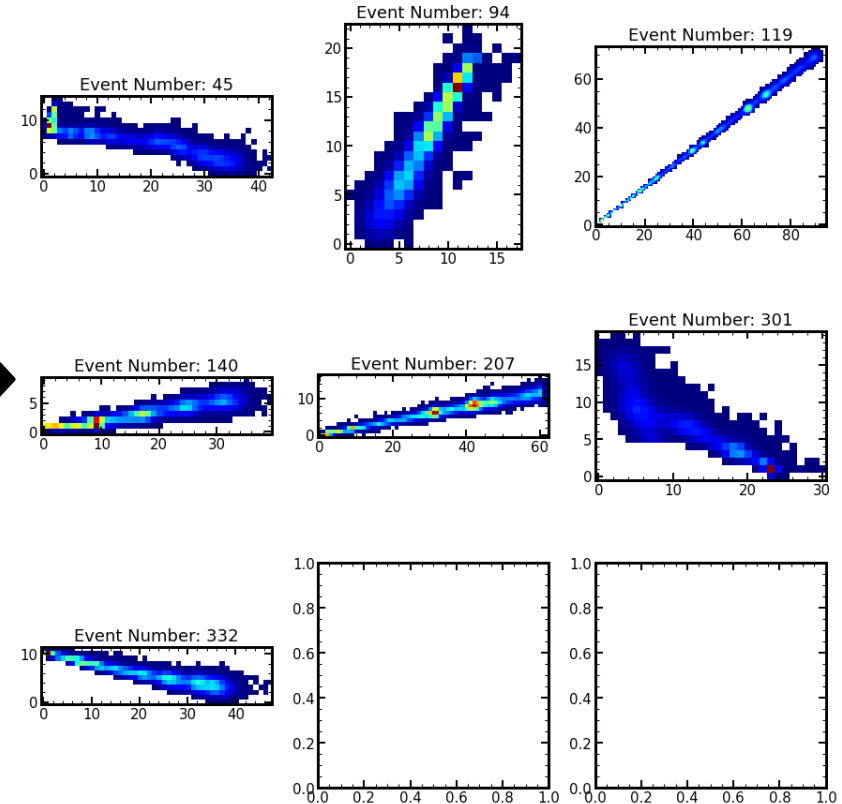
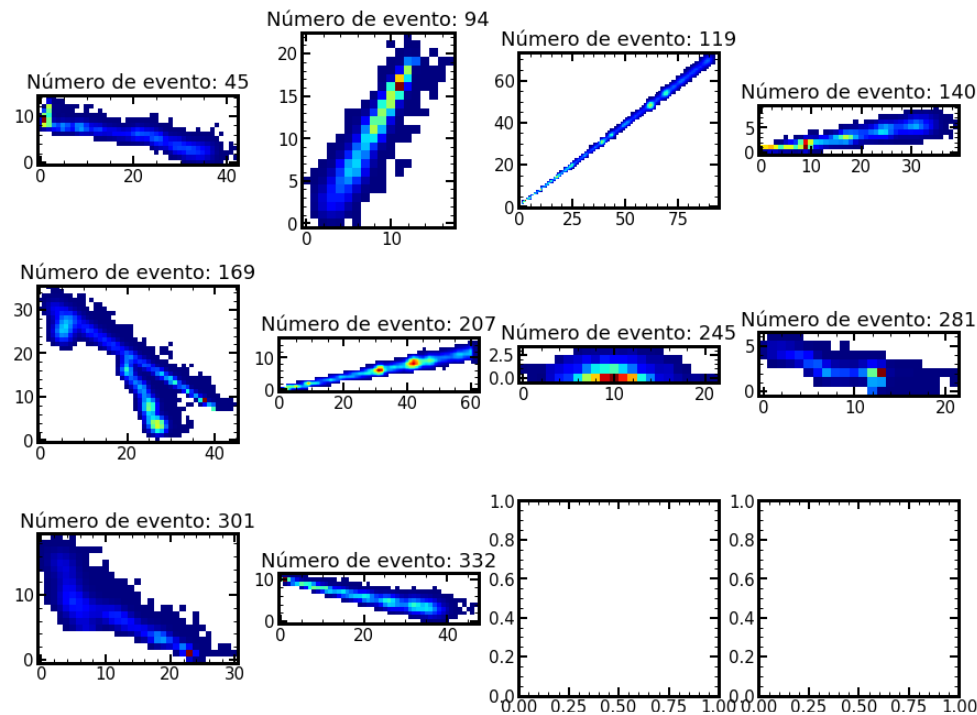
Distribución de dE/dL



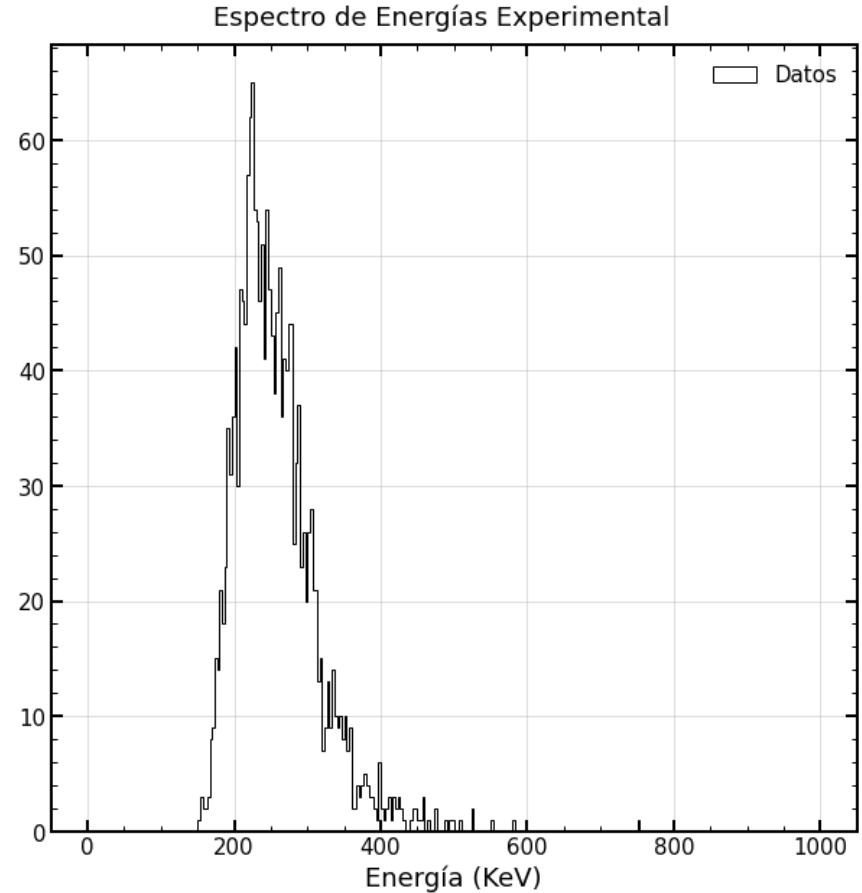
CONNIE

Max value G4: 2900
Max value PP: 2833.33

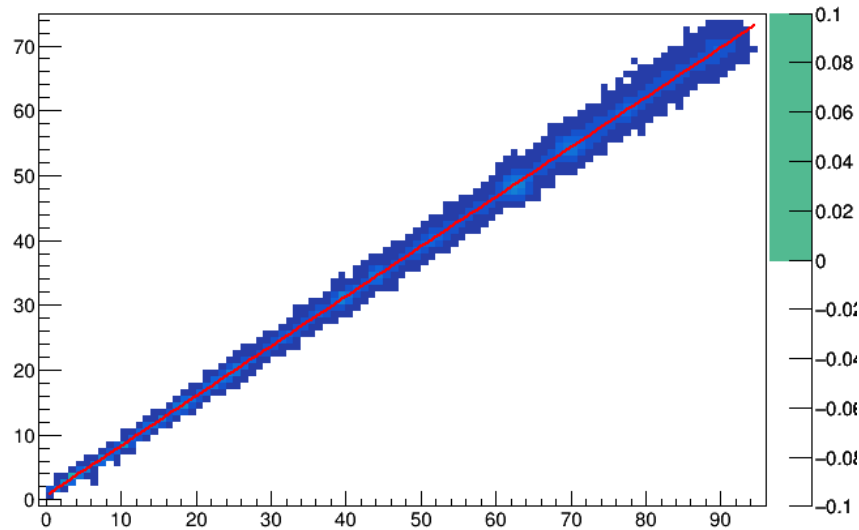
Tomando como referencia el espectro de dE/dL se añadió como parámetro al filtro de muones considerando solo valores entre 2000 y 3500 KeV/cm. Abajo se muestra el cambio en los muones detectados al implementar esto.



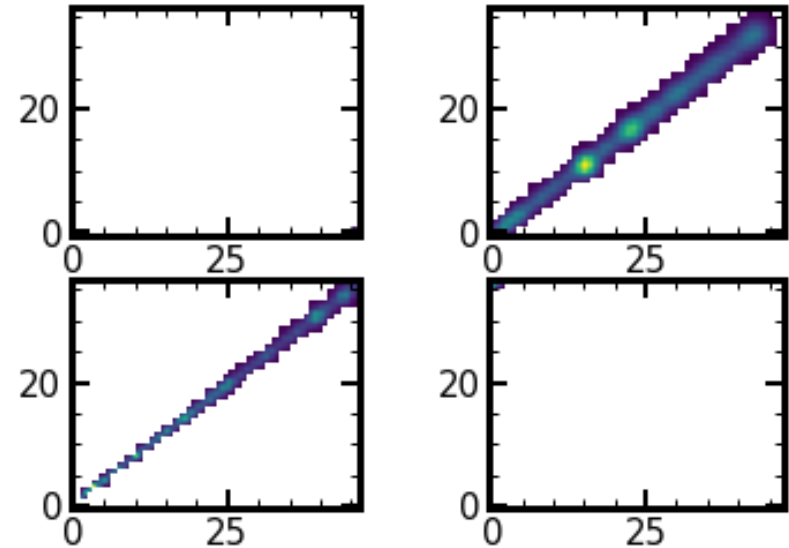
Sin embargo hacer esto también modificó el espectro de energías el cual se muestra abajo ya que la estructura de plateau desapareció. Tal vez modificando ese rango de aceptación podría mejorar el filtro y a la vez mantener el espectro original, por este motivo no se implementó este nuevo parámetro para CONNIE



Se implementó el nuevo método para determinar el ángulo ϕ de los muones realizando un ajuste lineal al cluster. Para determinar cual es la entrada del muon el cluster se divide en secciones y se cuenta el número de píxeles vacíos, así donde haya un mayor número de píxeles vacíos será la zona donde entró el muon y desde ahí se medirá el ángulo ϕ .

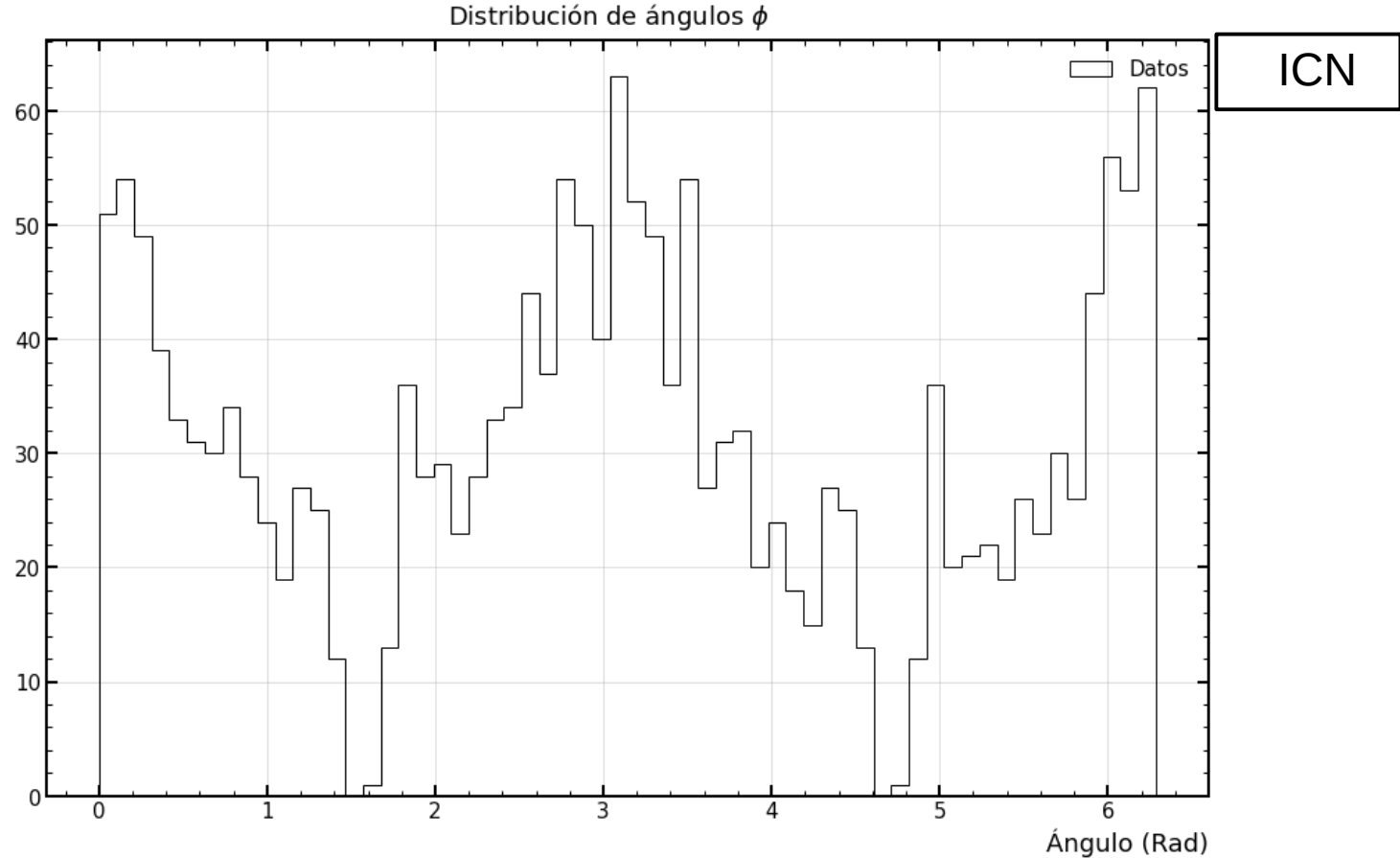


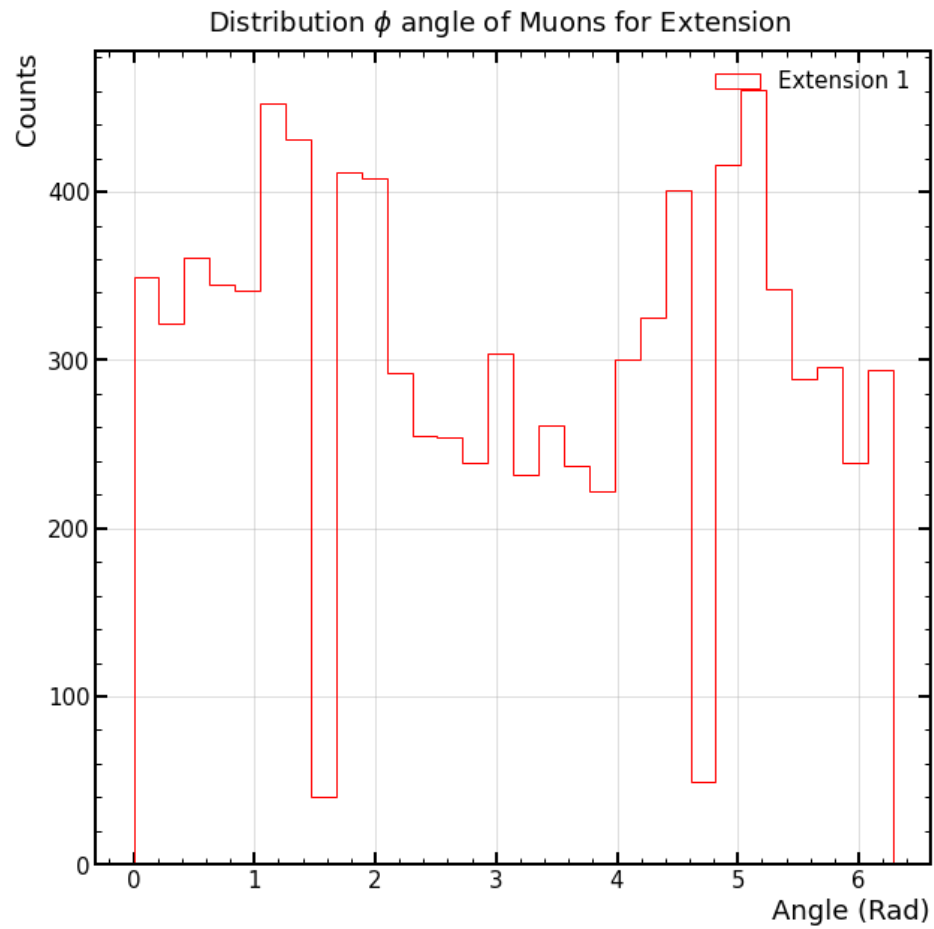
```
Shape of image (X, Y): 95 74
Ordenada: 0.5132114741545486
Pendiente: 0.767891635143501
Probabilidad: 0.02817908350779794
Chi square: 28.724549833139765
```



```
pendiente positiva
El muon está en el sector 1
El ángulo phi es: 0.6548537668942694 rad o 37.52035704128547°
```

Con esto se obtuvo el espectro de ángulos ϕ de todas las imágenes. Abajo se muestra el del ICN y en la siguiente diapositiva el de CONNIE.



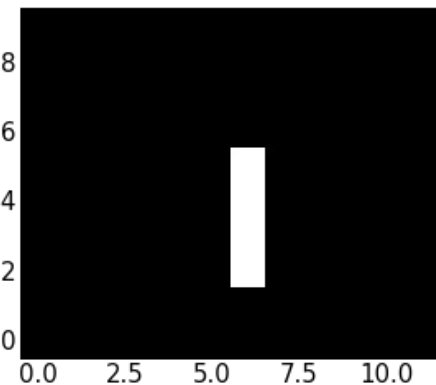
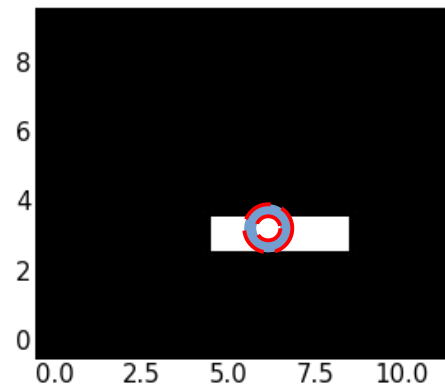
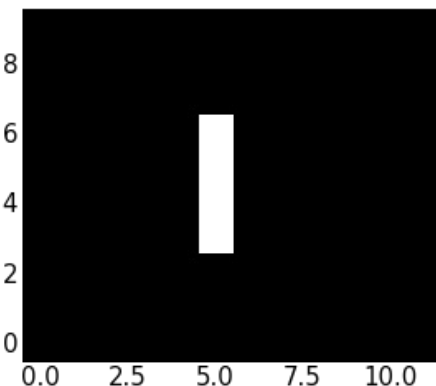
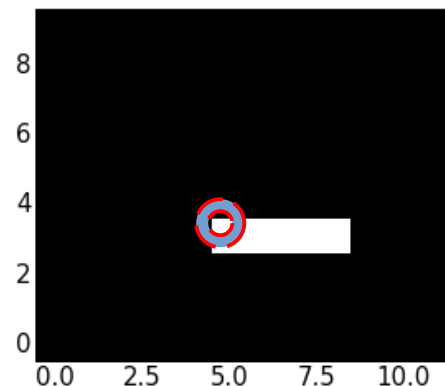


CONNIE

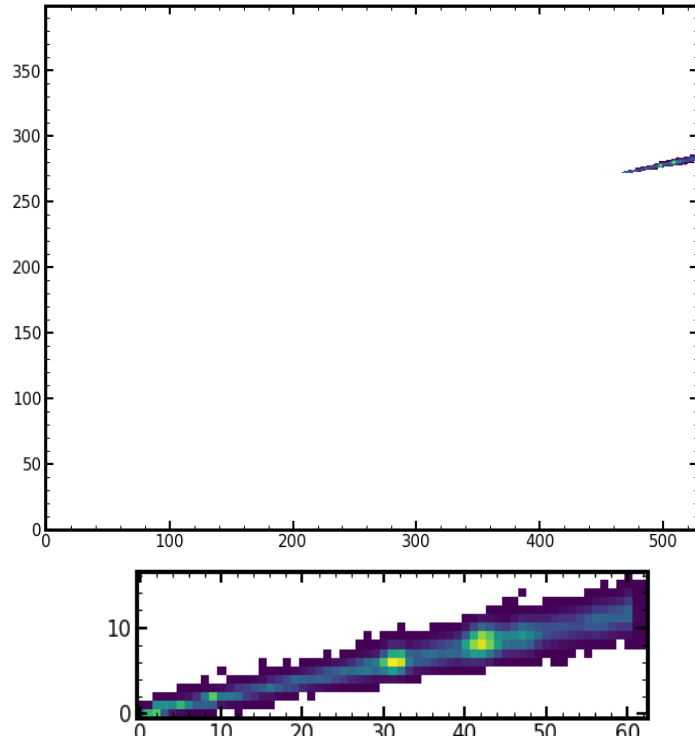
También se logró realizar la rotación de muones usando una matriz de rotaciones, abajo se muestra la función que se utiliza para realizarlo y un ejemplo con un cluster artificial. Las rotaciones se pueden hacer tomando cualquier punto de referencia.

```
def pixel_rot(x_bin, x0, y_bin, y0, theta):  
    diff_x = x_bin - x0  
    diff_y = y_bin - y0  
  
    new_x = diff_x * np.cos(theta) - diff_y * np.sin(theta) + x0  
    new_y = diff_x * np.sin(theta) + diff_y * np.cos(theta) + y0  
  
    return int(np.around(new_x, 0)), int(np.around(new_y, 0))
```

```
flag_firstbin = False  
for x_bin in range(0, example.shape[1]):  
    for y_bin in range(0, example.shape[0]):  
        if example[y_bin][x_bin] > 0 and not flag_firstbin:  
            x0, y0 = x_bin, y_bin  
            nx, ny = pixel_rot(x_bin=x_bin, x0=x0, y_bin=y_bin, y0=y0, theta= np.pi/2)  
            example_copy[ny][nx] = example[y_bin][x_bin]  
            flag_firstbin = True  
            # print(example[y_bin][x_bin])  
  
        elif example[y_bin][x_bin] > 0 and flag_firstbin:  
            nx, ny = pixel_rot(x_bin=x_bin, x0=x0, y_bin=y_bin, y0=y0, theta= np.pi/2)  
            example_copy[ny][nx] = example[y_bin][x_bin]  
            flag_firstbin = True
```



Se realizó la rotación de un muon considerando el tamaño de toda la imagen original y se tomó como referencia el píxel inicial, además se utilizó el ángulo ϕ que se le calculó antes. Aun se debe mejorar el código para saber cual es el punto de referencia que se tomará en todos, con esto se puede aumentar la estadística para el modelo de difusión.



Rotación

