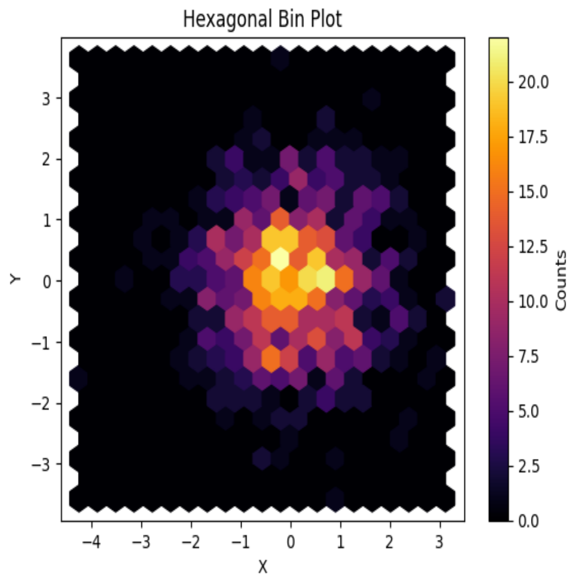
HEXAGONAL BINNING'S: PLT.HEXBIN()

- Hexagonal binning is a technique used in data visualization to represent the distribution of points on a two-dimensional scatterplot.
- It involves dividing the plot into a grid of hexagons and counting the number of points that fall within each hexagon.
- This technique is particularly useful when dealing with a large number of data points that might overlap, making it difficult to discern patterns or trends.
- By using hexagonal bins, we can aggregate the data and create a more informative and visually appealing representation of the distribution.

1. To create a hexagonal binning plot in Python, we can use the **hexbin()** function from the matplotlib library. Here's an example code snippet that demonstrates how to create a hexagonal binning plot

```
import matplotlib.pyplot as plt
import numpy as np

# Generate some random data
x = np.random.randn(1000)
y = np.random.randn(1000)

# Create a hexagonal binning plot
plt.hexbin(x, y, gridsize=20, cmap='inferno')

# Add a colorbar
cb = plt.colorbar()
cb.set_label('Counts')

# Add axis labels and a title
plt.xlabel('X')
plt.ylabel('Y')
plt.title('Hexagonal Bin Plot')

# Display the plot
plt.show()
```

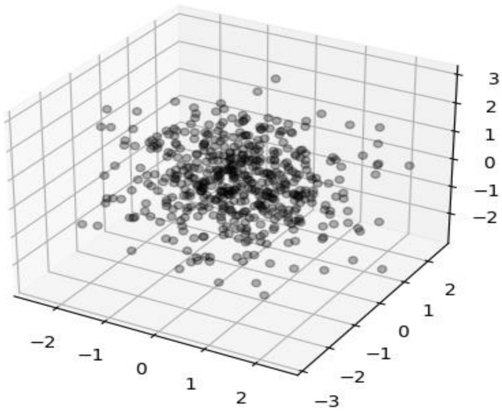


- ➢ In this example, we first generate some random data for the x and y variables. We then use the **hexbin**() function to create a hexagonal binning plot with a grid size of 20 and a colormap of 'inferno'.
- ➢ We add a color bar to the plot and label it with the word 'Counts'. We also add axis labels and a title to the plot. Finally, we display the plot using the **show**() function.
- ➢ The resulting plot will show the distribution of points in a two-dimensional scatterplot, with the hexagons indicating the density of points in different regions of the plot. The color of each hexagon represents the number of points that fall within it.

Kernel Density Estimation:

- Kernel density estimation (KDE) is a non-parametric method used to estimate the probability density function of a random variable.
- In other words, KDE is a way to estimate the probability distribution of a continuous random variable based on a set of observed data
- The basic idea behind KDE is to estimate the density of a variable by smoothing the observations.
- The bandwidth of the kernel controls the degree of smoothing. A small bandwidth results in a very peaked estimate with a lot of detail, while a large bandwidth produces a smoother estimate with less detail.
- KDE is a popular method in statistics and machine learning, and is used in a variety of applications, such as image processing, signal processing, and financial modeling. It is also used in exploratory data analysis to visualize the distribution of data and identify potential outliers.

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import gaussian_kde
from mpl_toolkits.mplot3d import Axes3D

# Generate sample data
np.random.seed(0)
x = np.random.normal(size=500)
y = np.random.normal(size=500)
z = np.random.normal(size=500)

# Fit the 3D KDE model
xyz = np.vstack([x,y,z])
kde = gaussian_kde(xyz)

# Create a grid of points to evaluate the KDE model
xmin, ymin, zmin = xyz.min(axis=1)
xmax, ymax, zmax = xyz.max(axis=1)
xi, yi, zi = np.mgrid[xmin:xmax:50j, ymin:ymax:50j, zmin:zmax:50j]
coords = np.vstack([xi.ravel(), yi.ravel(), zi.ravel()])
density = kde(coords).reshape(xi.shape)

# Plot the 3D KDE estimate
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.scatter(x, y, z, c='k', s=20, alpha=0.3)
ax.set_xlim3d(xmin, xmax)
ax.set_ylim3d(ymin, ymax)
ax.set_zlim3d(zmin, zmax)
ax.plot_surface(xi, yi, zi, rstride=1, cstride=1, alpha=0.3, facecolors=plt.cm.jet(density))
ax.set_xlabel('X Label')
ax.set_ylabel('Y Label')
ax.set_zlabel('Z Label')

plt.show()
```



- This code generates three arrays of random data, fits a 3D KDE model using **gaussian_kde**, and creates a grid of points to evaluate the density of the model
- It then uses **plot_surface** to plot the 3D KDE estimate on a scatter plot of the data.
- The resulting plot shows a smooth, continuous estimate of the probability density function in three dimensions.

Questions:

1. Explain the Hexagonal Binnings?

2.with Example describe the plt.hexbin() Method?

3. what is KDE and explain with the Example?