

1. Visual Inspection of Data Samples Visualizing random samples of digits to check clarity, noise level, and visual diversity. python CopyEdit

```
import matplotlib.pyplot as plt
import numpy as np # Display 25 images from training set
plt.figure(figsize=(6,6))
for i in range(25):
    plt.subplot(5,5,i+1)
    plt.imshow(x_train[i].reshape(28, 28), cmap='gray')
plt.axis('off')
plt.suptitle("Sample Handwritten Digits")
plt.tight_layout()
plt.show()
```

2. Distribution of Digit Classes Checks if all digit classes (0–9) are uniformly represented. python CopyEdit

```
import seaborn as sns
import pandas as pd
digit_labels = np.argmax(y_train, axis=1)
sns.countplot(x=digit_labels)
plt.title("Digit Class Distribution")
plt.xlabel("Digit")
plt.ylabel("Count")
plt.show()
```

3. Pixel Intensity Histogram Checks brightness distribution; useful for normalization strategies. python CopyEdit

```
plt.figure(figsize=(10,4))
plt.hist(x_train.flatten(), bins=50, color='blue', alpha=0.7)
plt.title("Histogram of Pixel Intensities")
plt.xlabel("Pixel Value")
plt.ylabel("Frequency")
plt.show()
```

4. Mean Image per Digit (Optional) Average image for each digit to understand typical structure. python CopyEdit

```
mean_images = np.zeros((10, 28, 28))
for i in range(10):
    mean_images[i] = np.mean(x_train[np.argmax(y_train, axis=1) == i], axis=0)
plt.figure(figsize=(12,4))
for i in range(10):
    plt.subplot(2,5,i+1)
    plt.imshow(mean_images[i], cmap='gray')
    plt.title(f"Digit: {i}")
plt.axis('off')
plt.suptitle("Mean Images per Digit")
plt.tight_layout()
plt.show()
```