Gen_data_before_pass_module2

September 6, 2024

1 Tổng hợp phân tích về phân phối của các chiều của các tập dữ liệu train:valid:test với 1 nhãn cụ thể

1.1 Trên tập train

1.1.1 Với chiều dim0

KS-test for Normal distribution (Feature dim_0, Label 0): Statistic = 0.040042477669672616, p-value = 0.6373726490918339 Data seems to follow a Normal distribution (Feature dim_0, Label 0). Estimated parameters: Mean = 6.6730952667907015, Std = 1.7099229776136318

KS-test for t-Student distribution (Feature dim_0, Label 0): Statistic = 0.030657664203170176, p-value = 0.8995688032414052 Data seems to follow a t-Student distribution (Feature dim_0, Label 0). Estimated parameters: Shape = 14.09546348144989, Location = 6.682333440864278, Scale = 1.583714636860711

KS-test for Normal distribution (Feature dim_0, Label 1): Statistic = 0.047096066333404685, p-value = 0.3602864755533288 Data seems to follow a Normal distribution (Feature dim_0, Label 1). Estimated parameters: Mean = -3.4882447721150815, Std = 1.6740043543332306

KS-test for t-Student distribution (Feature dim_0, Label 1): Statistic = 0.024246225821293765, p-value = 0.9755120735841096 Data seems to follow a t-Student distribution (Feature dim_0, Label 1). Estimated parameters: Shape = 6.887969994494039, Location = -3.4826628121412195, Scale = 1.4173412983965794

KS-test for Normal distribution (Feature dim_0, Label 2): Statistic = 0.05564988238010782, p-value = 0.06542883258181875 Data seems to follow a Normal distribution (Feature dim_0, Label 2). Estimated parameters: Mean = -3.4285463575587602, Std = 1.5730853500789983

KS-test for t-Student distribution (Feature dim_0, Label 2): Statistic = 0.03221903723820441, p-value = 0.6106052178867643 Data seems to follow a t-Student distribution (Feature dim_0, Label 2). Estimated parameters: Shape = 6.741591270675432, Location = -3.49702852240671, Scale = 1.3222121093801271

1.1.2 Với chiều dim1

KS-test for Normal distribution (Feature dim_1, Label 0): Statistic = 0.03727908406410496, p-value = 0.7225536360879004 Data seems to follow a Normal distribution (Feature dim_1, Label 0). Estimated parameters: Mean = -3.0121014564787245, Std = 1.0196592716038102

KS-test for t-Student distribution (Feature dim_1, Label 0): Statistic = 0.037043221460074216, p-value = 0.7297091081381815 Data seems to follow a t-Student distribution (Feature dim_1, Label 0). Estimated parameters: Shape = 5950.827878483189, Location = -3.012115508088379, Scale = 1.0179750456734245

KS-test for Normal distribution (Feature dim_1, Label 1): Statistic = 0.056442767687892914, p-value = 0.1730994174592072 Data seems to follow a Normal distribution (Feature dim_1, Label 1). Estimated parameters: Mean = 3.400910885285133, Std = 1.554637383396044

KS-test for t-Student distribution (Feature dim_1, Label 1): Statistic = 0.034463030108380366, p-value = 0.7468607042291524 Data seems to follow a t-Student distribution (Feature dim_1, Label 1). Estimated parameters: Shape = 9.750214897481058, Location = 3.3563756179056035, Scale = 1.3858427392089983

KS-test for Normal distribution (Feature dim_1, Label 2): Statistic = 0.04520581150144809, p-value = 0.20808679195563928 Data seems to follow a Normal distribution (Feature dim_1, Label 2). Estimated parameters: Mean = -1.2972524945910733, Std = 1.638770436031492

KS-test for t-Student distribution (Feature dim_1, Label 2): Statistic = 0.045391540428530774, p-value = 0.20428334602700504 Data seems to follow a t-Student distribution (Feature dim_1, Label 2). Estimated parameters: Shape = 205760715754.7307, Location = -1.2972519730476502, Scale = 1.637269440760408

1.1.3 Với chiều dim2

KS-test for Normal distribution (Feature dim_2, Label 0): Statistic = 0.0648440762667804, p-value = 0.1124684731629576 Data seems to follow a Normal distribution (Feature dim_2, Label 0). Estimated parameters: Mean = -4.077170213004602, Std = 1.3730336842490314

KS-test for t-Student distribution (Feature dim_2, Label 0): Statistic = 0.064802765531351, p-value = 0.11287859428160196 Data seems to follow a t-Student distribution (Feature dim_2, Label 0). Estimated parameters: Shape = 6031.925947111136, Location = -4.077088001028676, Scale = 1.3707732893436253

KS-test for Normal distribution (Feature dim_2, Label 1): Statistic = 0.05373941101930846, p-value = 0.21710920823556878 Data seems to follow a Normal distribution (Feature dim_2, Label 1). Estimated parameters: Mean = -0.9183746229206758, Std = 1.4671256563338897

KS-test for t-Student distribution (Feature dim_2, Label 1): Statistic = 0.05386117651121236, p-value = 0.21495842173363422 Data seems to follow a t-Student distribution (Feature dim_2, Label

1). Estimated parameters: Shape = 32496280275.072308, Location = -0.918375499267911, Scale = 1.4651836200780632

KS-test for Normal distribution (Feature dim_2, Label 2): Statistic = 0.0499777040624142, p-value = 0.1263867167291649 Data seems to follow a Normal distribution (Feature dim_2, Label 2). Estimated parameters: Mean = 3.9749454849198353, Std = 1.9974776048174143

KS-test for t-Student distribution (Feature dim_2, Label 2): Statistic = 0.05017935126148748, p-value = 0.12361324019964848 Data seems to follow a t-Student distribution (Feature dim_2, Label 2). Estimated parameters: Shape = 777723636.4044157, Location = 3.974960695156831, Scale = 1.995645310828559

1.2 Trên tập valid

1.2.1 Với chiều dim0

KS-test for Normal distribution (Feature dim_0, Label 0): Statistic = 0.1082386296713711, p-value = 0.34291775907057015 Data seems to follow a Normal distribution (Feature dim_0, Label 0). Estimated parameters: Mean = 5.901460106174151, Std = 1.8331534638824876

KS-test for t-Student distribution (Feature dim_0, Label 0): Statistic = 0.10821789129821696, p-value = 0.3431413556923713 Data seems to follow a t-Student distribution (Feature dim_0, Label 0). Estimated parameters: Shape = 1849360219.1876264, Location = 5.9014627376083215, Scale = 1.8203761770184674

KS-test for Normal distribution (Feature dim_0, Label 1): Statistic = 0.07073411599309065, p-value = 0.7858012448424398 Data seems to follow a Normal distribution (Feature dim_0, Label 1). Estimated parameters: Mean = -3.1368570243134912, Std = 2.053941346776682

KS-test for t-Student distribution (Feature dim_0, Label 1): Statistic = 220.0577306260747692, p-value = 0.9356047314311468 Data seems to follow a t-Student distribution (Feature dim_0, Label 1). Estimated parameters: Shape = 6.485429842685162, Location = -3.2332248006121977, Scale = 1.6914875734252144

KS-test for Normal distribution (Feature dim_0, Label 2): Statistic = 0.07536065692684668, p-value = 0.4960663622025945 Data seems to follow a Normal distribution (Feature dim_0, Label 2). Estimated parameters: Mean = -3.2969148199782414, Std = 1.6415364599073772

KS-test for t-Student distribution (Feature dim_0, Label 2): Statistic = 0.059097383275878546, p-value = 0.7860129425189303 Data seems to follow a t-Student distribution (Feature dim_0, Label 2). Estimated parameters: Shape = 4.223346154580167, Location = -3.3912647182193707, Scale = 1.2250927860514396

1.2.2 Với chiều dim1

KS-test for Normal distribution (Feature dim_1, Label 0): Statistic = 0.09955412373543537, p-value = 0.444737758821241 Data seems to follow a Normal distribution (Feature dim_1, Label 0). Estimated parameters: Mean = -2.4497756492346525, Std = 1.3684680686210058

KS-test for t-Student distribution (Feature dim_1, Label 0): Statistic = 0.09012933222361286, p-value = 0.5711964716870412 Data seems to follow a t-Student distribution (Feature dim_1, Label 0). Estimated parameters: Shape = 9.157332138584566, Location = -2.5582845759493873, Scale = 1.2042946178217853

KS-test for Normal distribution (Feature dim_1, Label 1): Statistic = 0.08287027076238829, p-value = 0.6045053868092165 Data seems to follow a Normal distribution (Feature dim_1, Label 1). Estimated parameters: Mean = 2.869425572362947, Std = 1.6604617301233344

KS-test for t-Student distribution (Feature dim_1, Label 1): Statistic = 0.05427487404271514, p-value = 0.9603683939324354 Data seems to follow a t-Student distribution (Feature dim_1, Label 1). Estimated parameters: Shape = 10.881846314206019, Location = 2.9378852936504316, Scale = 1.494620210960159

KS-test for Normal distribution (Feature dim_1, Label 2): Statistic = 0.06472145956277509, p-value = 0.686478681684763 Data seems to follow a Normal distribution (Feature dim_1, Label 2). Estimated parameters: Mean = -0.8275960896235819, Std = 2.205532377103237

KS-test for t-Student distribution (Feature dim_1, Label 2): Statistic = 0.06563896145795223, p-value = 0.6697520395473657 28Data seems to follow a t-Student distribution (Feature dim_1, Label 2). Estimated parameters: Shape = 28166851689.583504, Location = -0.8275948799862657, Scale = 2.1960877379307933

1.2.3 Với chiều dim2

KS-test for Normal distribution (Feature dim_2, Label 0): Statistic = 0.05296224756871615, p-value = 0.9811973642218337 Data seems to follow a Normal distribution (Feature dim_2, Label 0). Estimated parameters: Mean = -3.78501249021954, Std = 1.2049539699806446

KS-test for t-Student distribution (Feature dim_2, Label 0): Statistic = 0.051772566530076336, p-value = 0.9850861739677995 Data seems to follow a t-Student distribution (Feature dim_2, Label 0). Estimated parameters: Shape = 503991144.07014275, Location = -3.785011835827479, Scale = 1.1965574961831855

KS-test for Normal distribution (Feature dim_2, Label 1): Statistic = 0.08271097473369815, p-value = 0.6069130437477042 Data seems to follow a Normal distribution (Feature dim_2, Label 1). Estimated parameters: Mean = -0.7532430911137734, Std = 1.6962860027793416

KS-test for t-Student distribution (Feature dim_2, Label 1): Statistic = 0.08399858167161356, p-value = 0.5875063314515516 Data seems to follow a t-Student distribution (Feature dim_2, Label

1). Estimated parameters: Shape = 90693296571.91873, Location = -0.7532428169012304, Scale = 1.6857823252660067

KS-test for Normal distribution (Feature dim_2, Label 2): Statistic = 0.07368451193400155, p-value = 0.5248803755227232 Data seems to follow a Normal distribution (Feature dim_2, Label 2). Estimated parameters: Mean = 3.3960575142986755, Std = 2.692087561822481

KS-test for t-Student distribution (Feature dim_2, Label 2): Statistic = 0.07451280732372995, p-value = 0.5105546664801632 Data seems to follow a t-Student distribution (Feature dim_2, Label 2). Estimated parameters: Shape = 41261535048.558044, Location = 3.396059125971778, Scale = 2.6805578397052976

1.3 Trên tập test

Cơ bản có giống đôi chút so với phân phối trên tập valid, có lúc gần phân phối của tập train 1 chút nhưng đa phần khá tương đồng với phân phối của tập valid.

Nhận xét quan trọng rút ra được là:

- Phân phối trên tập valid và test thì có xu hướng là các **giá trị trung bình** của các Label 0, Label 1, Label 2 gần về 0 hơn 1 chút.
- Ngoài ra thì phương sai có xu hướng lớn hơn đôi chút trên tập train.
- Với các nhãn Label 0, Label 1 với các dim0, dim1 thì xu hướng phân phối tStudent chiếm chủ đạo hơn (có thể lý do đến từ việc số bản ghi của chúng ít hơn).

Trên đây là phân tích ảnh hưởng phân phối của 3 chiều dim0, dim1, dim2 với 3 nhãn Label 0: B2, Label 1: B5, Label 2: B6. Có thể thấy 39 chiều từ dim0 đến dim38 thực sự có hiện tượng đa cộng tính lớn (có tương quan với nhau). Bởi vậy các vấn đề cần xử lý ở đây sẽ khá nhức đầu hơn 1 chút: - Vấn đề tạo thêm dữ liệu của tập train (tạo theo phân phối của tập train theo phân phối tStudent mà vẫn đảm bảo giá trị trung bình của phân phối gần 0 hơn 1 chút, giá trị phương sai lớn hơn 1 chút để có giống hơn với phân phối của tập valid và test). - Các cách triển khai để khắc phục tính đa cộng tính của 39 chiều này.

2 vấn đề trên đặt ra các công việc cần triển khai sau đây: - Bài toán 1: Tạo sinh thêm dữ liệu cho tập train mà đảm bảo sự tương đồng với phân phối dữ liệu của tập valid 1 chút. Tạo ra 2 phiên bản: một là chỉ dùng dữ liệu gốc, hai là dùng dữ liệu tăng cường để đánh giá cách nào cho hiệu quả tốt hơn - Bài toán 2: Thực hiện các phương pháp đánh giá ảnh hưởng của hiện tượng đa cộng tuyến so với 39 nhãn đặc trưng đầu vào. - Sử dụng 39 chiều này vào 1 ANN (39 dense, 97 dense, 3 dense) không quan tâm gì về đa cộng tuyến - Sử dụng 39 chiều này được giảm chiều phù hợp cho tác vụ phân loại (sử dụng PCA hoặc LDA) rồi mới cho vào 1 ANN phù hợp - Và 1 số cách khác hiện tại tôi chưa liệt kê ở đây.

2 Bài toán 1. Tạo sinh thêm dữ liệu

Sinh thêm dữ liệu 39 chiều dựa theo phân phối của từng Label với từng chiều từ dim0 đến dim38 theo phân phối tStudent và đảm bảo thêm 1 số cách để dữ liệu tăng cường thêm được giống phân phối của tập valid 1 chút (mean gần 0 hơn, var lớn hơn 1 chút) Dữ liệu tạo sinh ra được lưu lại dưới 1 file csv (tăng cường cho tập train thôi)

[ ]:
```python
import numpy as np

def augment_data(X):
    jitter = np.random.normal(0, 0.5, X.shape)
    # 0 và 0.5 là 2 đại lượng được ước lượng 1 cách xem xét kỹ lưỡng nhiều yếu␣
 ↪tố
    X_augmented = X + jitter
    return X_augmented

# Tăng cường dữ liệu lên gấp 20 lần
def augment_data_multiple_times(X_train, X_valid, y_train, y_valid,␣
 ↪num_repeats=20):
    if num_repeats % 4 != 0:
        raise ValueError("num_repeats must be a multiple of 4")
        # bởi vì đang cố gen dữ liệu theo phân phối của tập valid gấp 3 lần của␣
 ↪tập train

    # Tăng dữ liệu gấp 4 lần
    augmented_data_X = [X_train]
    augmented_data_Y = [y_train]
    for _ in range(int(3*70/15)):
        augmented_data_X.append(augment_data(X_valid))
        augmented_data_Y.append(y_valid)
    augmented_data_X = np.concatenate(augmented_data_X, axis=0)
    augmented_data_Y = np.concatenate(augmented_data_Y, axis=0)

    # Tăng dữ liệu gấp num_repeats // 4 lần
    X_augmented = [augmented_data_X]
    Y_augmented = [augmented_data_Y]
    for _ in range(num_repeats // 4 - 1):
        X_augmented.append(augment_data(augmented_data_X))
        Y_augmented.append(augmented_data_Y)
    X_augmented = np.concatenate(X_augmented, axis=0)
    Y_augmented = np.concatenate(Y_augmented, axis=0)

    return X_augmented, Y_augmented
```

[ ]:
```
!pwd
```

```
/mnt/DataSamsung/project/Research_ThyroidFNA_ClassAI/phase2_280824/notebooks/exp
lore/before_pass_module2/create_more_data
```

[ ]:
```python
import pandas as pd

# Đọc dữ liệu từ CSV
data_dir = '../../../../data/processed/'
```

```python
train_df = pd.read_csv(data_dir + 'train_features.csv').
 ↪drop(columns=['image_path'])
valid_df = pd.read_csv(data_dir + 'valid_features.csv').
 ↪drop(columns=['image_path'])
# test_df = pd.read_csv(data_dir + 'test_features.csv').
 ↪drop(columns=['image_path'])

# Xem cấu trúc của DataFrame
print('Train DataFrame:')
print(train_df.head(3))
print('Valid DataFrame:')
print(valid_df.head(3))
# print('Test DataFrame:')
# print(test_df.head(3))
```

```
Train DataFrame:
   label     dim_0     dim_1     dim_2     dim_3     dim_4     dim_5  \
0      2  1.518592 -2.122206  1.063359  1.083591 -3.078244  1.832143
1      2 -3.997844 -2.013066  5.606269 -2.221863  0.908517  0.995050
2      2 -4.144322 -3.335181  7.014940 -2.851661 -1.254374  4.090703

      dim_6     dim_7     dim_8  …    dim_29    dim_30    dim_31    dim_32  \
0 -3.617894  1.835063  0.739837  … -0.450482  0.230301  0.322319 -0.361697
1 -4.953275  1.075495  3.199515  …  2.695855 -4.411282 -1.748895  5.605376
2 -3.888599 -2.345787  6.091836  …  6.469357 -5.430499 -2.013651  6.954945

     dim_33    dim_34    dim_35    dim_36    dim_37    dim_38
0 -1.096648  1.402599  0.442939 -3.928780  2.343227  1.505659
1 -6.375985 -0.828269  6.179725 -4.965612  0.590781  3.893331
2 -3.220028 -1.721707  5.068684 -4.063029 -1.196603  5.176606

[3 rows x 40 columns]
Valid DataFrame:
   label     dim_0     dim_1     dim_2     dim_3     dim_4     dim_5  \
0      2 -4.308372 -0.436520  3.545043 -2.595636 -1.076509  3.019734
1      2 -2.273990 -2.976918  4.630661 -5.481691  1.987185  3.077199
2      2 -4.070328 -3.232069  6.819434 -3.002290 -1.391492  3.774137

      dim_6     dim_7     dim_8  …    dim_29    dim_30    dim_31    dim_32  \
0 -3.748788 -0.974676  4.619431  …  4.596820 -4.209521 -0.469024  4.025341
1 -4.994466  1.433963  2.813911  …  3.553807 -3.890979 -0.693126  4.631985
2 -2.197132 -2.203136  3.730880  …  2.323358 -3.455380 -3.436366  6.039357

     dim_33    dim_34    dim_35    dim_36    dim_37    dim_38
0 -3.474844 -1.487661  5.017033 -1.916778 -0.383969  2.663312
1 -2.152809 -0.549343  3.090957 -1.015712 -1.852066  2.541131
2 -3.923026 -2.357094  5.158929 -4.472951 -3.406128  6.673154
```

```
[3 rows x 40 columns]
```

[ ]:
```
X_train = train_df.drop(columns=['label']).values
y_train = train_df['label'].values
X_valid = valid_df.drop(columns=['label']).values
y_valid = valid_df['label'].values

print(type(X_train), X_train.shape)
print(type(y_train), y_train.shape)
print(type(X_valid), X_valid.shape)
print(type(y_valid), y_valid.shape)
```

```
<class 'numpy.ndarray'> (1261, 39)
<class 'numpy.ndarray'> (1261,)
<class 'numpy.ndarray'> (270, 39)
<class 'numpy.ndarray'> (270,)
```

[ ]:
```
# Tăng cường dữ liệu
X_train_augmented_np, y_train_augmented_np =␣
↪augment_data_multiple_times(X_train, X_valid, y_train, y_valid,␣
↪num_repeats=20)

print(type(X_train_augmented_np), X_train_augmented_np.shape)
print(type(y_train_augmented_np), y_train_augmented_np.shape)
```

```
<class 'numpy.ndarray'> (25205, 39)
<class 'numpy.ndarray'> (25205,)
```

[ ]:
```
# Lưu dữ liệu tăng cường vào file CSV
augmented_data_dir = '../../../../data/augmented/'
train_augmented_df = pd.DataFrame(data=X_train_augmented_np, columns=train_df.
↪columns.drop('label'))
train_augmented_df.insert(0, 'label', y_train_augmented_np)
train_augmented_df.to_csv(augmented_data_dir + 'train_augmented_features.csv',␣
↪index=False)
```

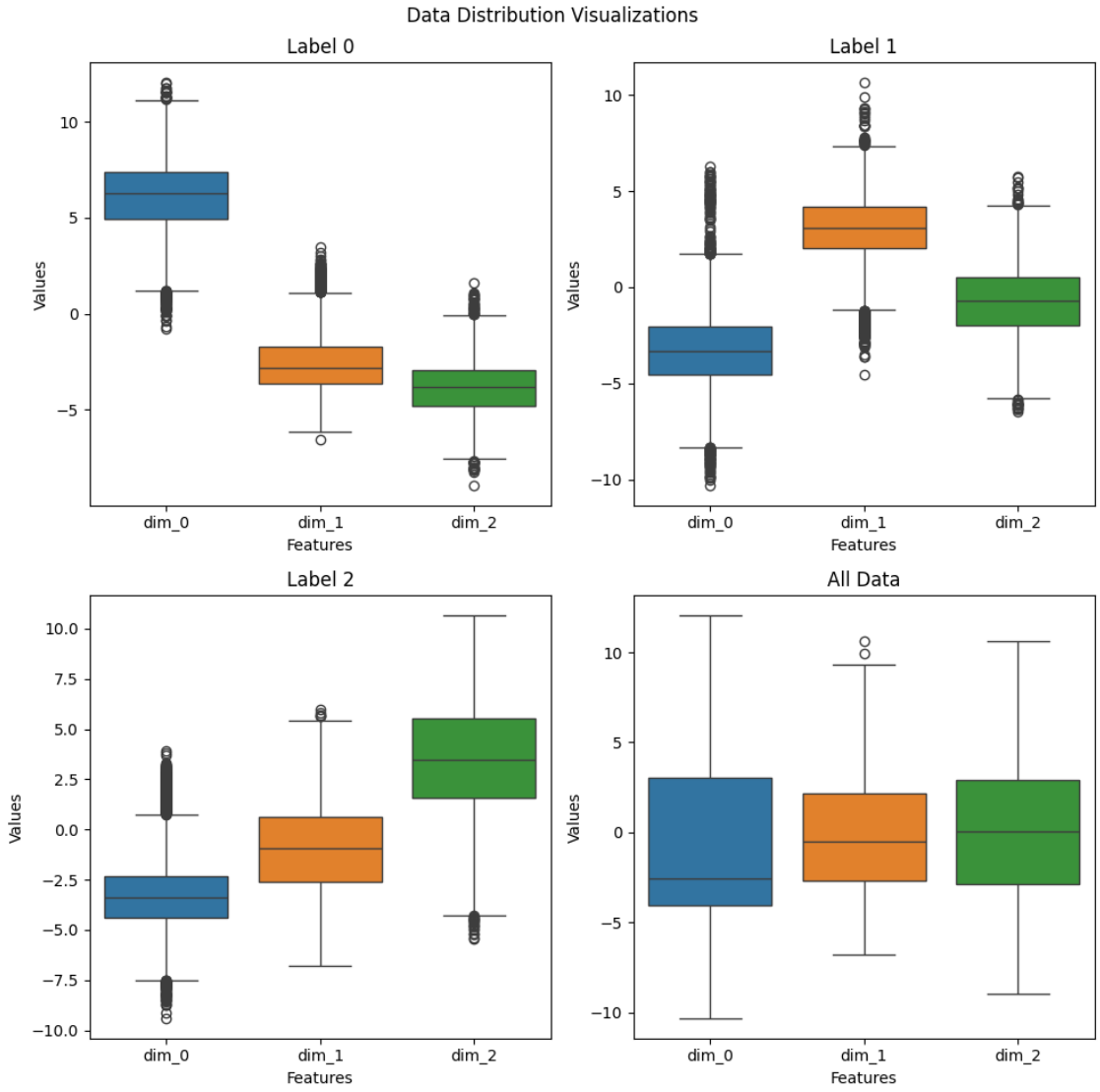
3 Trực quan hóa dữ liệu vừa tạo ra

[ ]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Sử dụng box plot
def visualize_distribution(features, labels):
    # Create a figure with 4 subplots
    fig, axs = plt.subplots(2, 2, figsize=(10, 10))
    fig.suptitle("Data Distribution Visualizations")
```

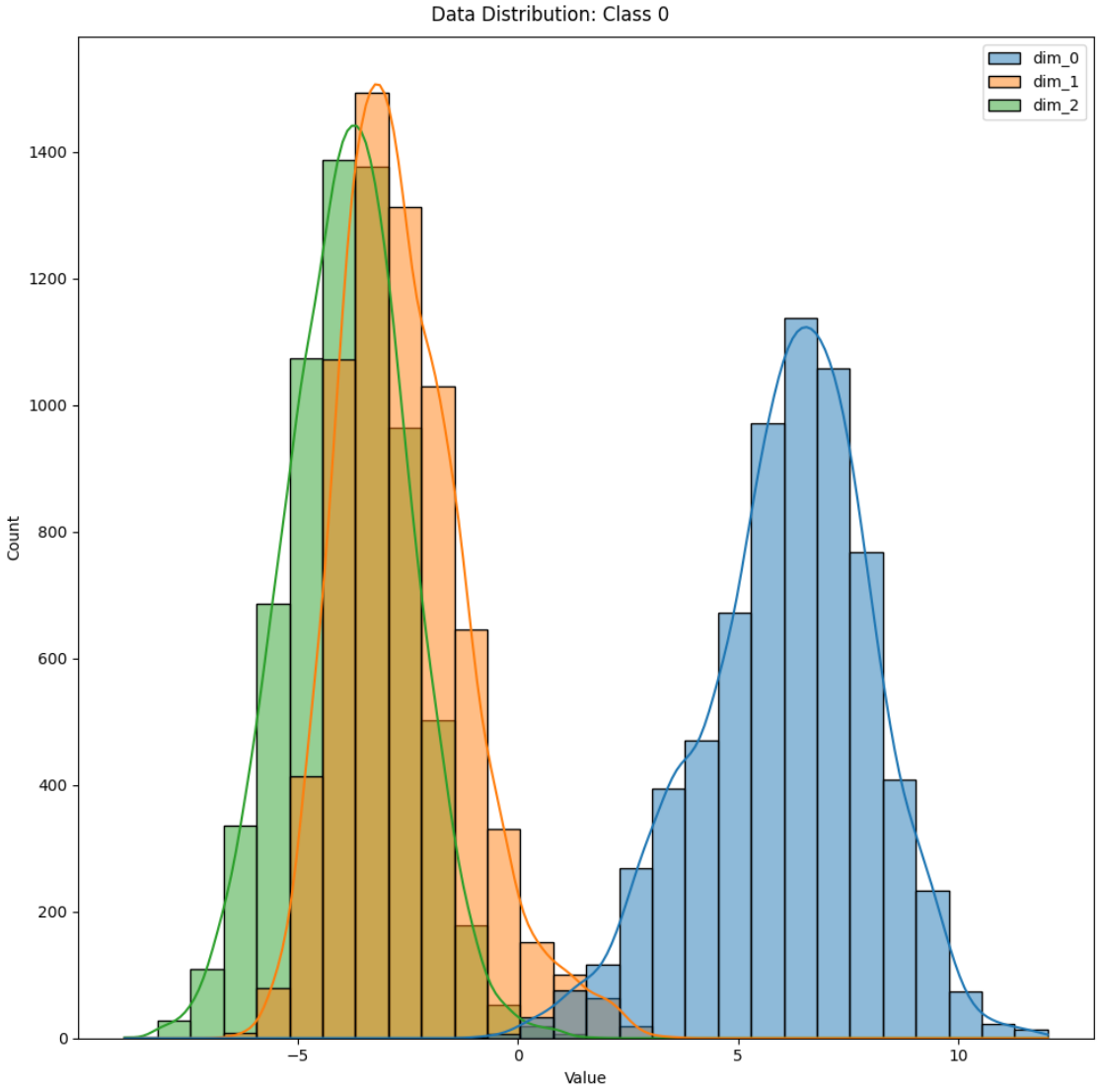
```
    # Plot for all data
    sns.boxplot(data=features, ax=axs[1, 1])
    axs[1, 1].set_title("All Data")
    axs[1, 1].set_xlabel("Features")
    axs[1, 1].set_ylabel("Values")

    # Plot for each label
    for i, label in enumerate([0, 1, 2]):
        label_data = features[labels == label]
        sns.boxplot(data=label_data, ax=axs[i // 2, i % 2])
        axs[i // 2, i % 2].set_title(f"Label {label}")
        axs[i // 2, i % 2].set_xlabel("Features")
        axs[i // 2, i % 2].set_ylabel("Values")

    plt.tight_layout()
    plt.show()


# Sử dụng histogram
def visualize_distribution_histogram(features, labels):
    # Create 4 separate figures
    num_features = features.shape[1]
    for i in range(4):
        if num_features == 3:
            plt.figure(figsize=(10, 10))
        else:
            plt.figure(figsize=(15, 15))

        if i == 3:
            title = "All Data"
            data_to_plot = features
        else:
            title = f"Class {i}"
            data_to_plot = features[labels == i]

        plt.suptitle(f"Data Distribution: {title}")

        sns.histplot(data_to_plot, kde=True, binwidth=0.75)
        # if num_features == 3:
        #     sns.histplot(data_to_plot, kde=True, binwidth=1)
        # else:
        #     sns.histplot(data_to_plot, kde=True, binwidth=2)
        plt.xlabel("Value")
        plt.ylabel("Count")

        plt.tight_layout()
```

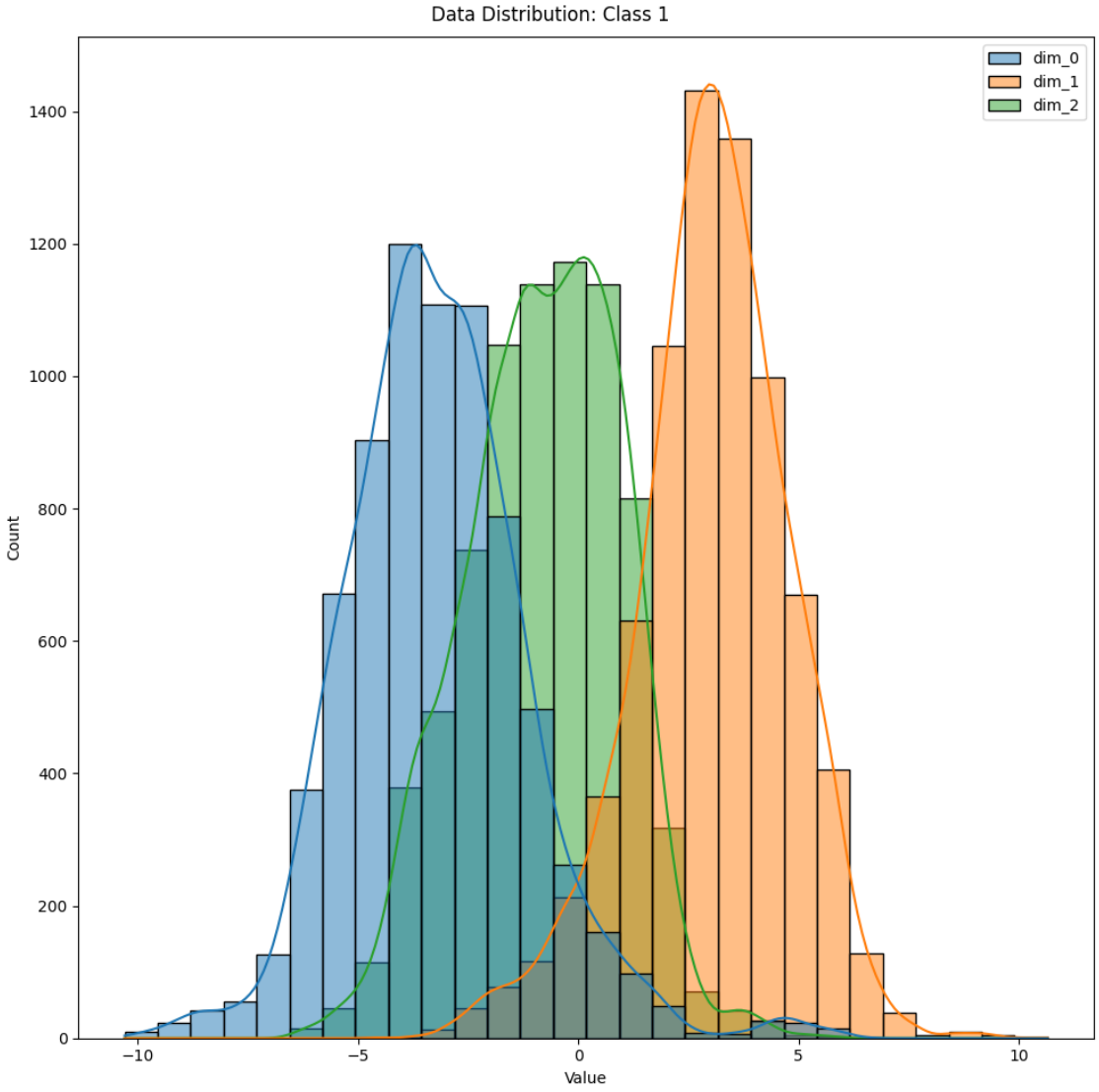
```
        plt.show()
```

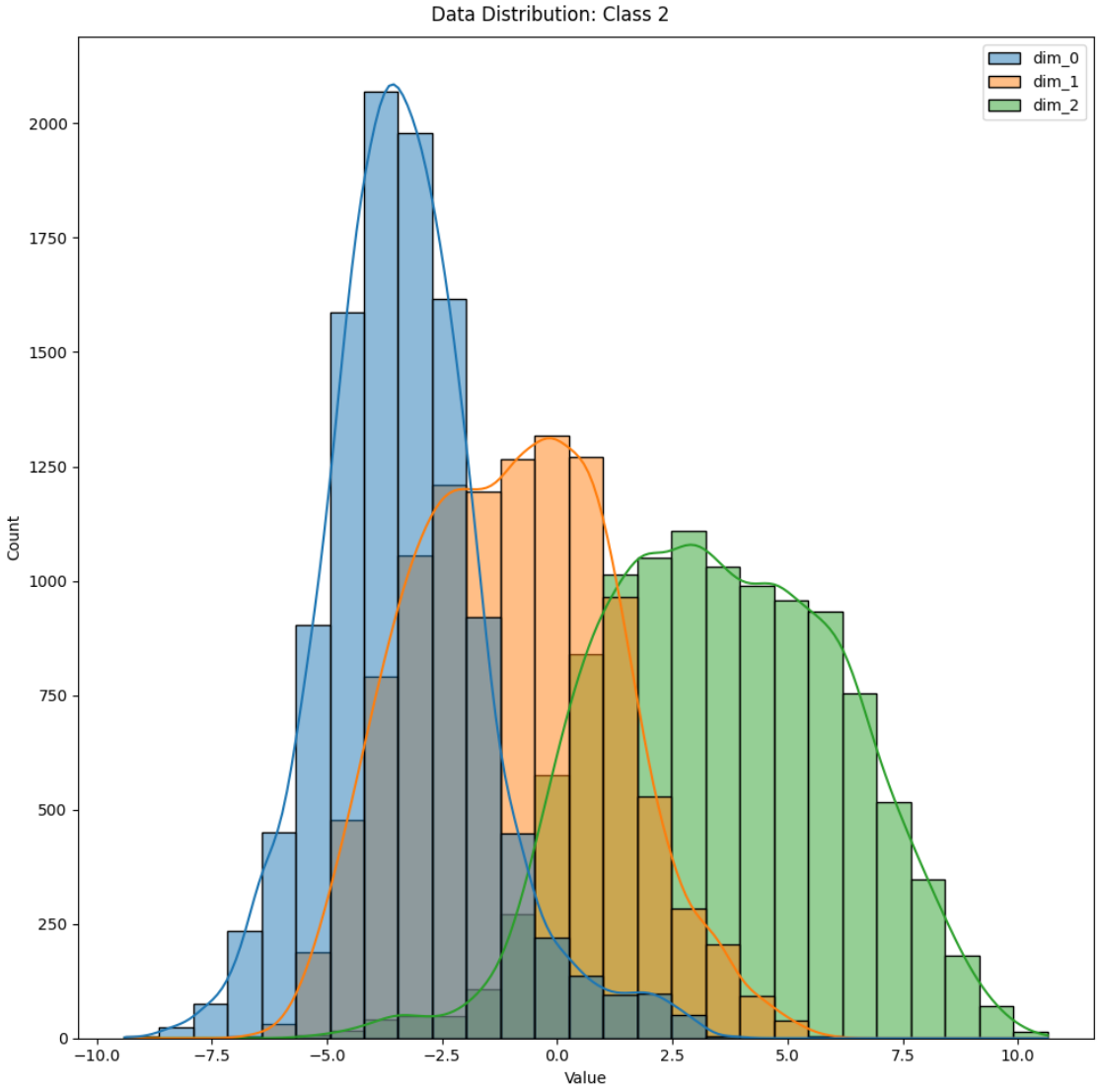
[ ]:
```
visualize_distribution(train_augmented_df.iloc[:, 1:4], train_augmented_df.
↪iloc[:, 0])
```

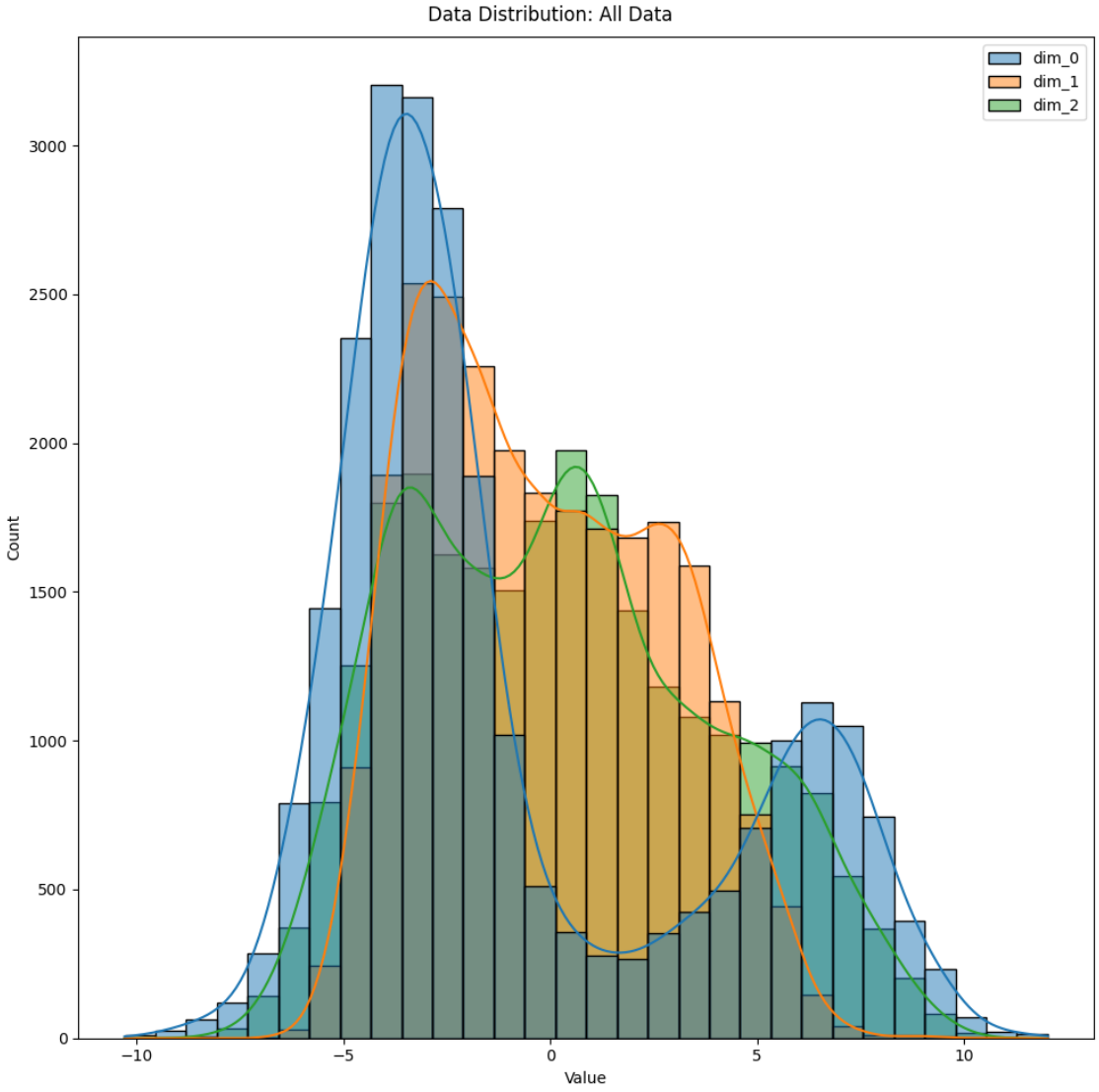


[ ]:
```
visualize_distribution_histogram(train_augmented_df.iloc[:, 1:4],␣
↪train_augmented_df.iloc[:, 0])
```

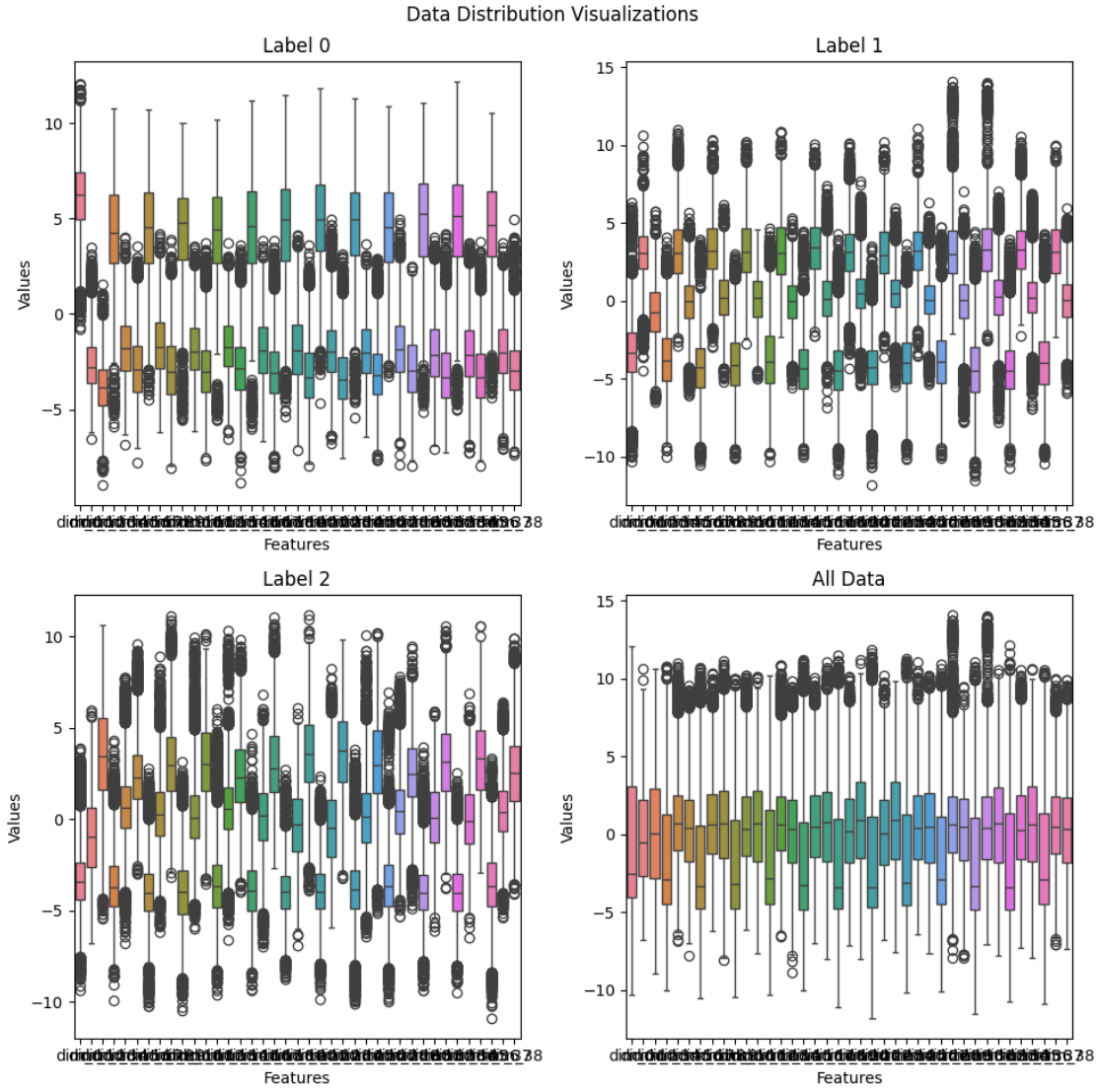

Data Distribution: Class 0
dim_0
dim_1
dim_2
Count
Value
0
200
400
600
800
1000
1200
1400
−5
0
5
10


Data Distribution: Class 1
dim_0
dim_1
dim_2
Count
Value
0
200
400
600
800
1000
1200
1400
−10
−5
0
5
10

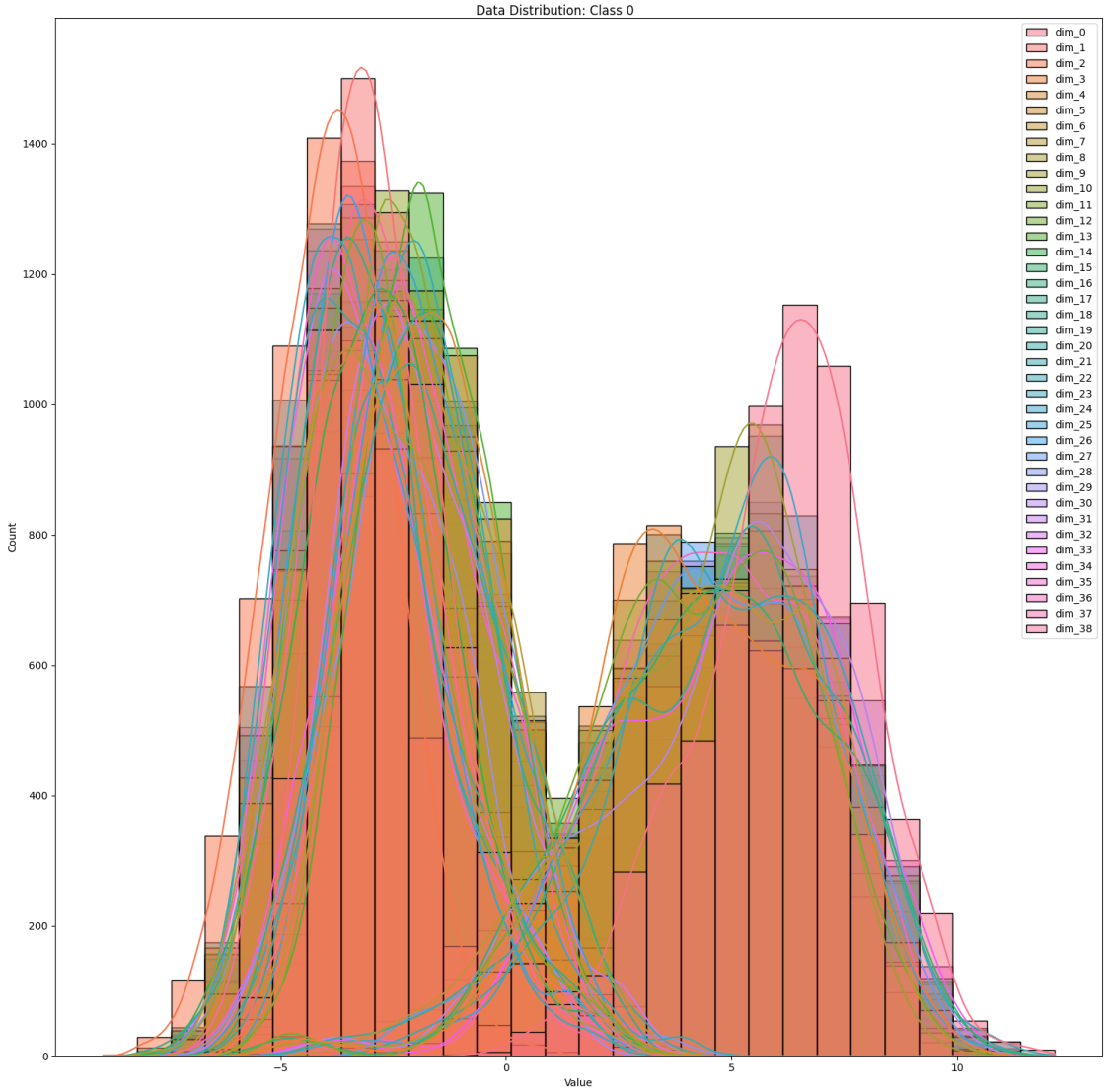

Data Distribution: Class 2
dim_0
dim_1
dim_2
Count
Value
0
250
500
750
1000
1250
1500
1750
2000
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0

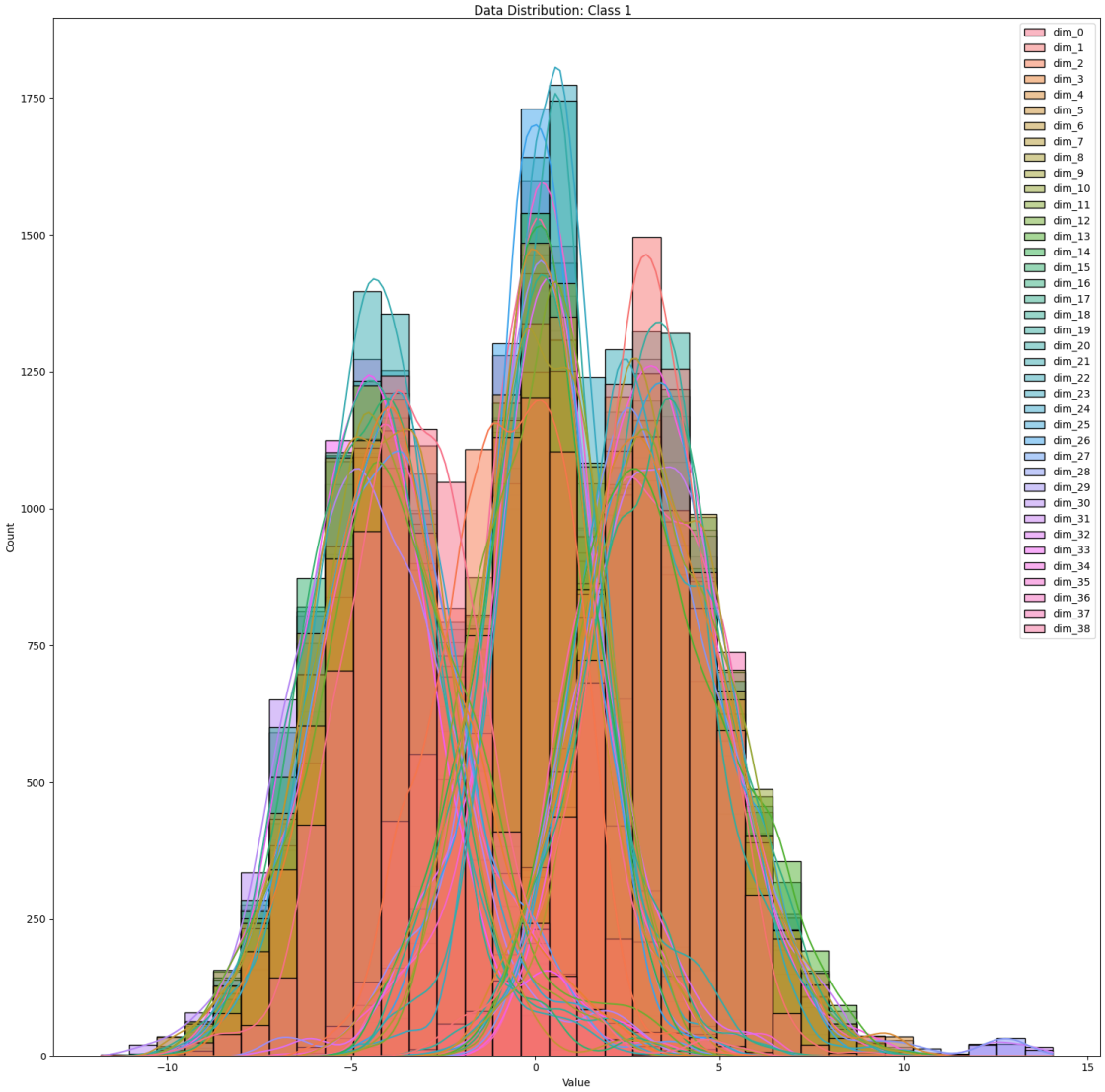


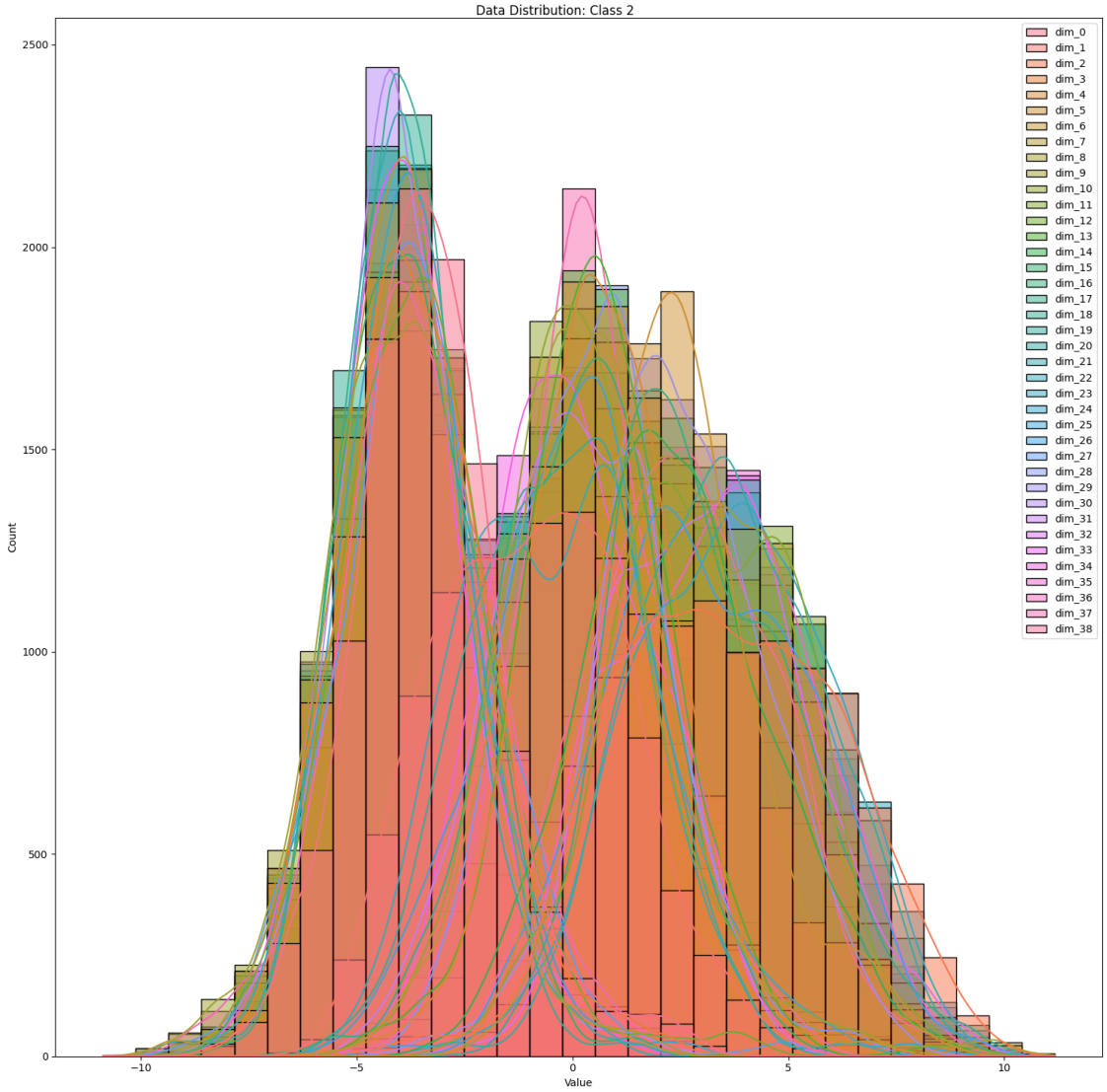

```
[ ]: visualize_distribution(train_augmented_df.iloc[:, 1:], train_augmented_df.iloc[:
     ↪, 0])
```

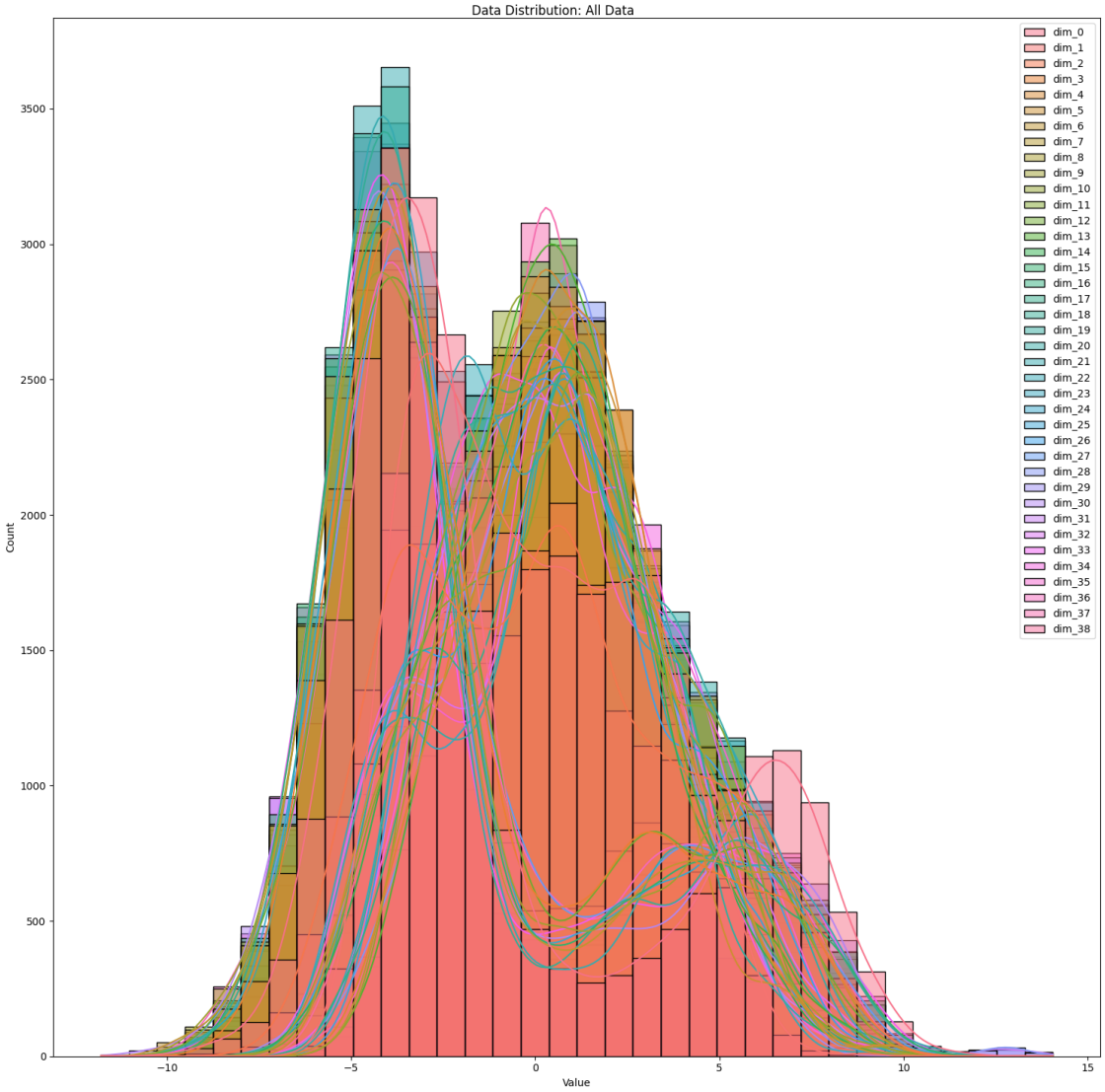


[ ]:
```
visualize_distribution_histogram(train_augmented_df.iloc[:, 1:],␣
↪train_augmented_df.iloc[:, 0])
```


Data Distribution: Class 0
Count
Value
0
200
400
600
800
1000
1200
1400
−5
0
5
10
dim_0
dim_1
dim_2
dim_3
dim_4
dim_5
dim_6
dim_7
dim_8
dim_9
dim_10
dim_11
dim_12
dim_13
dim_14
dim_15
dim_16
dim_17
dim_18
dim_19
dim_20
dim_21
dim_22
dim_23
dim_24
dim_25
dim_26
dim_27
dim_28
dim_29
dim_30
dim_31
dim_32
dim_33
dim_34
dim_35
dim_36
dim_37
dim_38


Data Distribution: Class 1
Count
Value
0
250
500
750
1000
1250
1500
1750
−10
−5
0
5
10
15
dim_0
dim_1
dim_2
dim_3
dim_4
dim_5
dim_6
dim_7
dim_8
dim_9
dim_10
dim_11
dim_12
dim_13
dim_14
dim_15
dim_16
dim_17
dim_18
dim_19
dim_20
dim_21
dim_22
dim_23
dim_24
dim_25
dim_26
dim_27
dim_28
dim_29
dim_30
dim_31
dim_32
dim_33
dim_34
dim_35
dim_36
dim_37
dim_38


Data Distribution: Class 2
Count
Value
2500
2000
1500
1000
500
0
−10
−5
0
5
10
dim_0
dim_1
dim_2
dim_3
dim_4
dim_5
dim_6
dim_7
dim_8
dim_9
dim_10
dim_11
dim_12
dim_13
dim_14
dim_15
dim_16
dim_17
dim_18
dim_19
dim_20
dim_21
dim_22
dim_23
dim_24
dim_25
dim_26
dim_27
dim_28
dim_29
dim_30
dim_31
dim_32
dim_33
dim_34
dim_35
dim_36
dim_37
dim_38


Data Distribution: All Data
Count
Value
0
500
1000
1500
2000
2500
3000
3500
−10
−5
0
5
10
15
dim_0
dim_1
dim_2
dim_3
dim_4
dim_5
dim_6
dim_7
dim_8
dim_9
dim_10
dim_11
dim_12
dim_13
dim_14
dim_15
dim_16
dim_17
dim_18
dim_19
dim_20
dim_21
dim_22
dim_23
dim_24
dim_25
dim_26
dim_27
dim_28
dim_29
dim_30
dim_31
dim_32
dim_33
dim_34
dim_35
dim_36
dim_37
dim_38