CaseStudy3

December 8, 2024

1 Module 10: Unsupervised Learning

1.1 Case Study – 3

[4]:
```
import pandas as pd
from sklearn.cluster import AgglomerativeClustering
from sklearn.metrics import mean_squared_error
import scipy.cluster.hierarchy as shc
from matplotlib import pyplot as plt

# Ignore warnings for clean output
import warnings
warnings.filterwarnings("ignore")

# Step 1: Load and Explore the Dataset
data = pd.read_csv('zoo.csv')

# Check basic information
print("Dataset Info:")
print(data.info())
print("\nMissing Values Check:")
print(data.isnull().sum())

# Display the first 5 rows
print("\nFirst 5 Rows of the Dataset:")
print(data.head())

# Summary statistics
print("\nSummary Statistics:")
print(data.describe())

# 2.  Find out the unique number of high-level cclasses
# Check the unique high-level classes (class_type)
unique_classes = data['class_type'].nunique()
print(f"\nUnique High-Level Classes: {unique_classes}")

# Class distribution
class_counts = data['class_type'].value_counts()
```

```python
print("\nClass Distribution:")
print(class_counts)
```

```
Dataset Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 101 entries, 0 to 100
Data columns (total 18 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   animal_name  101 non-null    object
 1   hair         101 non-null    int64
 2   feathers     101 non-null    int64
 3   eggs         101 non-null    int64
 4   milk         101 non-null    int64
 5   airborne     101 non-null    int64
 6   aquatic      101 non-null    int64
 7   predator     101 non-null    int64
 8   toothed      101 non-null    int64
 9   backbone     101 non-null    int64
 10  breathes     101 non-null    int64
 11  venomous     101 non-null    int64
 12  fins         101 non-null    int64
 13  legs         101 non-null    int64
 14  tail         101 non-null    int64
 15  domestic     101 non-null    int64
 16  catsize      101 non-null    int64
 17  class_type   101 non-null    int64
dtypes: int64(17), object(1)
memory usage: 14.3+ KB
None

Missing Values Check:
animal_name    0
hair           0
feathers       0
eggs           0
milk           0
airborne       0
aquatic        0
predator       0
toothed        0
backbone       0
breathes       0
venomous       0
fins           0
legs           0
tail           0
domestic       0
```

```
catsize        0
class_type     0
dtype: int64

First 5 Rows of the Dataset:
  animal_name  hair  feathers  eggs  milk  airborne  aquatic  predator  \
0    aardvark     1         0     0     1         0        0         1
1    antelope     1         0     0     1         0        0         0
2        bass     0         0     1     0         0        1         1
3        bear     1         0     0     1         0        0         1
4        boar     1         0     0     1         0        0         1

   toothed  backbone  breathes  venomous  fins  legs  tail  domestic  catsize  \
0        1         1         1         0     0     4     0         0        1
1        1         1         1         0     0     4     1         0        1
2        1         1         0         0     1     0     1         0        0
3        1         1         1         0     0     4     0         0        1
4        1         1         1         0     0     4     1         0        1

   class_type
0           1
1           1
2           4
3           1
4           1

Summary Statistics:
             hair    feathers        eggs        milk    airborne     aquatic  \
count  101.000000  101.000000  101.000000  101.000000  101.000000  101.000000
mean     0.425743    0.198020    0.584158    0.405941    0.237624    0.356436
std      0.496921    0.400495    0.495325    0.493522    0.427750    0.481335
min      0.000000    0.000000    0.000000    0.000000    0.000000    0.000000
25%      0.000000    0.000000    0.000000    0.000000    0.000000    0.000000
50%      0.000000    0.000000    1.000000    0.000000    0.000000    0.000000
75%      1.000000    0.000000    1.000000    1.000000    0.000000    1.000000
max      1.000000    1.000000    1.000000    1.000000    1.000000    1.000000

         predator     toothed    backbone    breathes    venomous        fins  \
count  101.000000  101.000000  101.000000  101.000000  101.000000  101.000000
mean     0.554455    0.603960    0.821782    0.792079    0.079208    0.168317
std      0.499505    0.491512    0.384605    0.407844    0.271410    0.376013
min      0.000000    0.000000    0.000000    0.000000    0.000000    0.000000
25%      0.000000    0.000000    1.000000    1.000000    0.000000    0.000000
50%      1.000000    1.000000    1.000000    1.000000    0.000000    0.000000
75%      1.000000    1.000000    1.000000    1.000000    0.000000    0.000000
max      1.000000    1.000000    1.000000    1.000000    1.000000    1.000000

             legs        tail    domestic     catsize  class_type
```

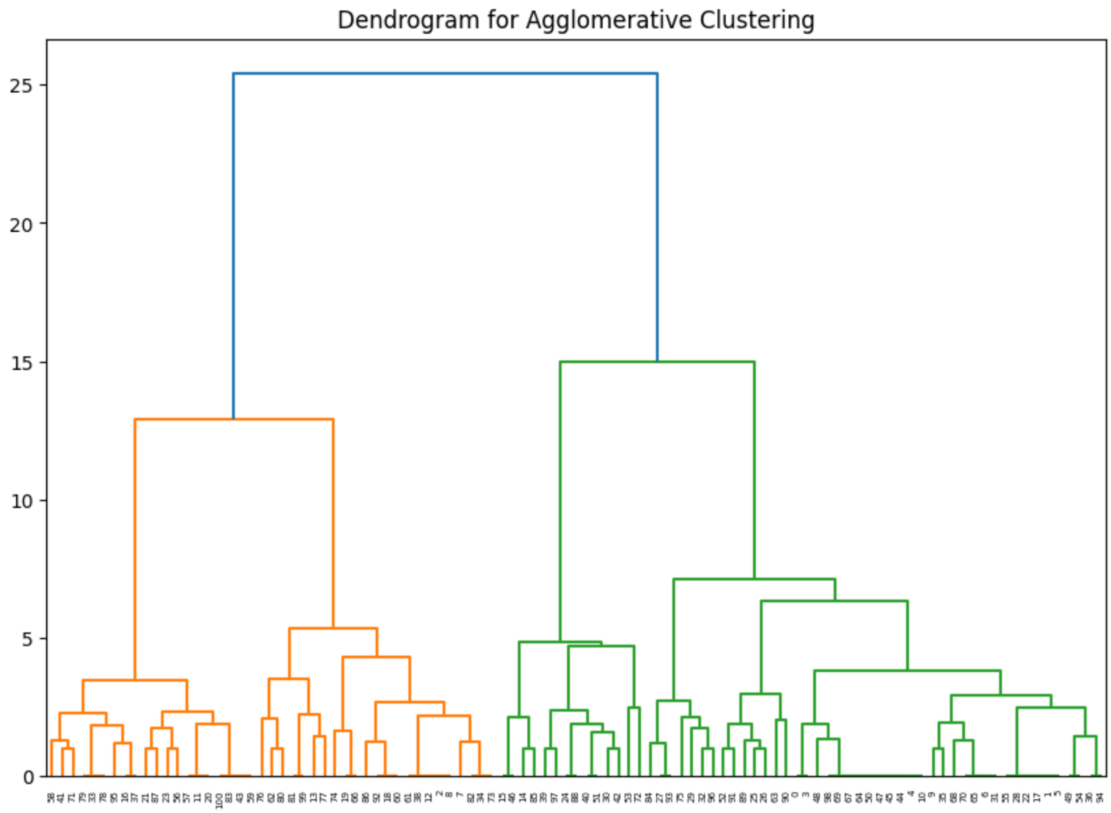
```
count  101.000000  101.000000  101.000000  101.000000  101.000000
mean     2.841584    0.742574    0.128713    0.435644    2.831683
std      2.033385    0.439397    0.336552    0.498314    2.102709
min      0.000000    0.000000    0.000000    0.000000    1.000000
25%      2.000000    0.000000    0.000000    0.000000    1.000000
50%      4.000000    1.000000    0.000000    0.000000    2.000000
75%      4.000000    1.000000    0.000000    1.000000    4.000000
max      8.000000    1.000000    1.000000    1.000000    7.000000

Unique High-Level Classes: 7

Class Distribution:
class_type
1    41
2    20
4    13
7    10
6     8
3     5
5     4
Name: count, dtype: int64
```

[5]:
```python
# Step 2: Extract Intermediate Features and Target
# Removing non-numerical column 'animal_name' and keeping only the first 16␣
↪features
features = data.iloc[:, 1:17]  # Columns 1 to 16 are the intermediate features
target = data['class_type']    # The actual high-level class
```

[6]:
```python
# Step 3: Perform Agglomerative Clustering
# Visualize the dendrogram to understand the clustering structure
plt.figure(figsize=(10, 7))
plt.title("Dendrogram for Agglomerative Clustering")
dendrogram = shc.dendrogram(shc.linkage(features, method='ward'))
plt.show()
```



[10]:
```
# Perform Agglomerative Clustering
n_clusters = unique_classes  # Set number of clusters to the number of unique classes
cluster = AgglomerativeClustering(n_clusters=n_clusters, metric='euclidean', linkage='ward')
predicted_classes = cluster.fit_predict(features)

print(predicted_classes)

# Step 4: Compute Mean Squared Error
# Compare predicted cluster labels with actual class labels
mse = mean_squared_error(target, predicted_classes)
print(f"\nMean Squared Error between Actual and Predicted Classes: {mse}")
```

```
[1 1 2 1 1 1 1 2 2 1 1 3 2 6 0 0 3 1 2 2 3 3 1 3 0 5 5 4 1 4 0 1 4 3 2 1 1
 3 2 0 0 3 0 3 1 1 0 1 1 1 1 0 5 0 1 1 3 3 3 3 2 2 6 5 1 1 2 1 1 1 1 3 0 2
 2 4 6 6 3 3 6 6 2 3 4 0 2 3 0 5 5 5 2 4 1 3 4 0 1 6 3]

Mean Squared Error between Actual and Predicted Classes: 7.673267326732673
```

[11]:
```python
# Animals in Each Cluster:

# Add the predicted cluster labels to the original dataset
data['predicted_cluster'] = predicted_classes

# Group animals by their predicted cluster
clusters = data.groupby('predicted_cluster')['animal_name'].apply(list)

# Print the animals in each cluster
print("\nAnimals in Each Cluster:")
for cluster_id, animals in clusters.items():
    print(f"Cluster {cluster_id}: {animals}")
```

```
Animals in Each Cluster:
Cluster 0: ['crab', 'crayfish', 'flea', 'gnat', 'honeybee', 'housefly',
'ladybird', 'lobster', 'moth', 'octopus', 'scorpion', 'starfish', 'termite',
'wasp']
Cluster 1: ['aardvark', 'antelope', 'bear', 'boar', 'buffalo', 'calf', 'cavy',
'cheetah', 'deer', 'elephant', 'giraffe', 'goat', 'hamster', 'hare', 'leopard',
'lion', 'lynx', 'mink', 'mole', 'mongoose', 'opossum', 'oryx', 'polecat',
'pony', 'puma', 'pussycat', 'raccoon', 'reindeer', 'vole', 'wolf']
Cluster 2: ['bass', 'carp', 'catfish', 'chub', 'dogfish', 'dolphin', 'haddock',
'herring', 'pike', 'piranha', 'porpoise', 'seahorse', 'seal', 'sole',
'stingray', 'tuna']
Cluster 3: ['chicken', 'crow', 'dove', 'duck', 'flamingo', 'gull', 'hawk',
'kiwi', 'lark', 'ostrich', 'parakeet', 'penguin', 'pheasant', 'rhea', 'skimmer',
'skua', 'sparrow', 'swan', 'vulture', 'wren']
Cluster 4: ['fruitbat', 'girl', 'gorilla', 'sealion', 'squirrel', 'vampire',
'wallaby']
Cluster 5: ['frog', 'frog', 'newt', 'platypus', 'toad', 'tortoise', 'tuatara']
Cluster 6: ['clam', 'pitviper', 'seasnake', 'seawasp', 'slowworm', 'slug',
'worm']
```

[14]:
```python
# Analyze Features Driving the Clustering
# To identify the features that most likely influence the clustering:
#Calculate the mean values of features within each cluster.
#Compare these means to the overall dataset or other clusters to see
 ↪distinguishing characteristics.

# Exclude non-numeric columns before calculating the mean
numeric_data = data.drop(columns=['animal_name', 'class_type'])  # Drop
 ↪non-numeric columns

# Calculate the mean of each feature per cluster
cluster_features = numeric_data.groupby(data['predicted_cluster']).mean()
```

```python
print("\nMean Values of Features per Cluster:")
print(cluster_features)

# Identify significant features for each cluster
print("\nKey Features Driving Each Cluster:")
for cluster_id, feature_means in cluster_features.iterrows():
    top_features = feature_means.sort_values(ascending=False).head(5)  # Top 5␣
↪features
    print(f"Cluster {cluster_id}:")
    for feature, value in top_features.items():
        print(f"  - {feature}: {value}")
```

```
Mean Values of Features per Cluster:
                       hair  feathers      eggs      milk  airborne   aquatic  \
predicted_cluster
0                  0.285714       0.0  0.928571  0.000000  0.428571  0.357143
1                  1.000000       0.0  0.000000  1.000000  0.000000  0.033333
2                  0.062500       0.0  0.812500  0.187500  0.000000  1.000000
3                  0.000000       1.0  1.000000  0.000000  0.800000  0.300000
4                  1.000000       0.0  0.000000  1.000000  0.285714  0.142857
5                  0.142857       0.0  1.000000  0.142857  0.000000  0.714286
6                  0.000000       0.0  0.857143  0.000000  0.000000  0.285714

                   predator   toothed  backbone  breathes  venomous      fins  \
predicted_cluster
0                  0.500000  0.000000  0.000000  0.642857  0.214286  0.000000
1                  0.533333  1.000000  1.000000  1.000000  0.000000  0.000000
2                  0.750000  1.000000  1.000000  0.187500  0.062500  1.000000
3                  0.450000  0.000000  1.000000  1.000000  0.000000  0.000000
4                  0.285714  1.000000  1.000000  1.000000  0.000000  0.142857
5                  0.714286  0.714286  1.000000  1.000000  0.142857  0.000000
6                  0.714286  0.428571  0.428571  0.571429  0.428571  0.000000

                       legs      tail  domestic   catsize  predicted_cluster
predicted_cluster
0                  6.071429  0.071429  0.071429  0.071429                0.0
1                  4.000000  0.900000  0.233333  0.800000                1.0
2                  0.000000  0.937500  0.062500  0.437500                2.0
3                  2.000000  1.000000  0.150000  0.300000                3.0
4                  2.000000  0.714286  0.142857  0.571429                4.0
5                  4.000000  0.571429  0.000000  0.285714                5.0
6                  0.000000  0.428571  0.000000  0.000000                6.0

Key Features Driving Each Cluster:
Cluster 0:
  - legs: 6.071428571428571
  - eggs: 0.9285714285714286
```

```
  - breathes: 0.6428571428571429
  - predator: 0.5
  - airborne: 0.42857142857142855
Cluster 1:
  - legs: 4.0
  - hair: 1.0
  - milk: 1.0
  - toothed: 1.0
  - backbone: 1.0
Cluster 2:
  - predicted_cluster: 2.0
  - fins: 1.0
  - toothed: 1.0
  - aquatic: 1.0
  - backbone: 1.0
Cluster 3:
  - predicted_cluster: 3.0
  - legs: 2.0
  - breathes: 1.0
  - feathers: 1.0
  - eggs: 1.0
Cluster 4:
  - predicted_cluster: 4.0
  - legs: 2.0
  - hair: 1.0
  - milk: 1.0
  - backbone: 1.0
Cluster 5:
  - predicted_cluster: 5.0
  - legs: 4.0
  - breathes: 1.0
  - eggs: 1.0
  - backbone: 1.0
Cluster 6:
  - predicted_cluster: 6.0
  - eggs: 0.8571428571428571
  - predator: 0.7142857142857143
  - breathes: 0.5714285714285714
  - venomous: 0.42857142857142855
```

[ ]:

```
```