Anomaly Detection in Healthcare Provider Data

Bishal Jaysawal

Problem Statement

Healthcare fraud is a significant issue that diverts essential resources from patient care to fraudulent activities, contributing to rising healthcare costs. This problem is pervasive and complex, involving various forms of deceit by both healthcare providers and patients. The dataset provided includes detailed information about healthcare providers, their services, and Medicare-related financial data.

The primary goal is to employ unsupervised learning techniques to identify anomalies within this dataset, which could indicate potentially fraudulent activities. Detecting these anomalies will help in flagging suspicious providers or claims, thereby aiding in the reduction of healthcare fraud and ensuring that funds are appropriately allocated to genuine medical needs.

To achieve this, various variables such as provider details, service counts, and financial metrics will be analysed. By identifying unusual patterns or outliers in these variables, we aim to pinpoint areas where further investigation may be required to uncover fraudulent activities.

Dataset Details

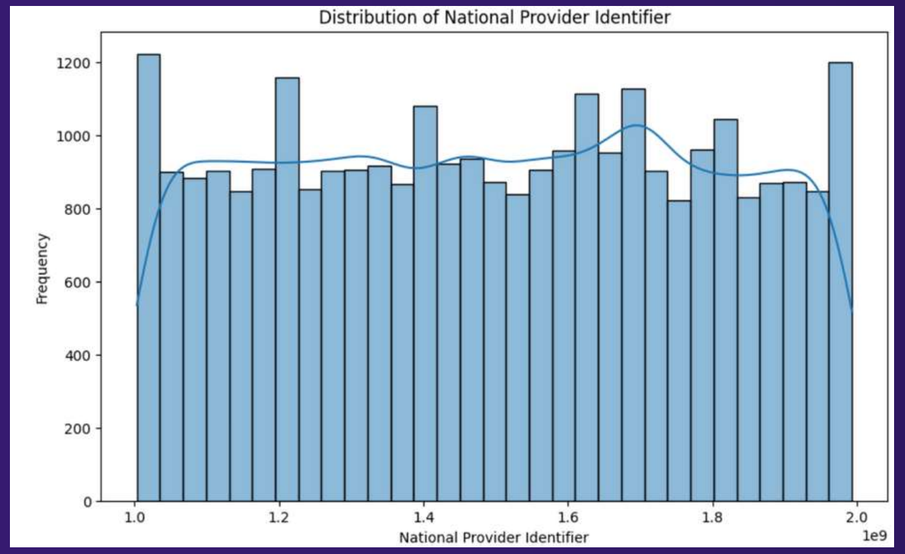
The dataset contains 100,000 entries of healthcare provider's insurance claims data, including categorical and numerical features. There were initially 100,000 rows and 27 columns.

Some Categorical Columns: National Provider Identifier, Last Name, First Name, Middle Initial, Credentials, Gender, Entity Type, Street Address 1, Street Address 2, City, State Code, Postal Code of the Provider, HCPCS Code, HCPCS Description, HCPCS Drug Indicator

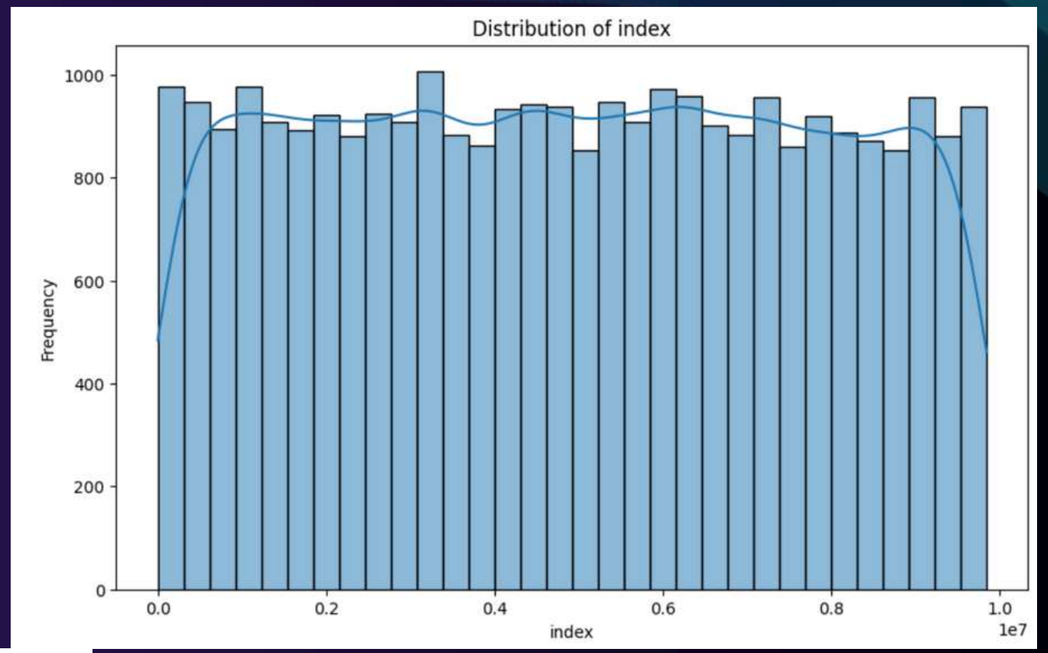
Numerical Columns: Number of Services, Number of Medicare Beneficiaries, Number of Distinct Medicare Beneficiary/Per Day Services, Average Medicare Allowed Amount, Average Submitted Charge Amount, Average Medicare Payment Amount, Average Medicare Standardized Amount

The final dataset after standardization consisted of 100000 rows and 11 columns

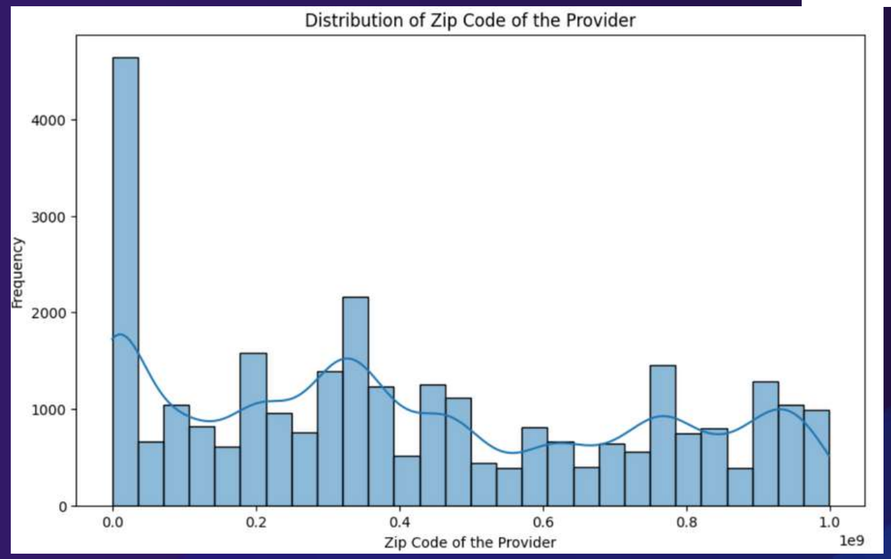
Exploratory Data Analysis (EDA)



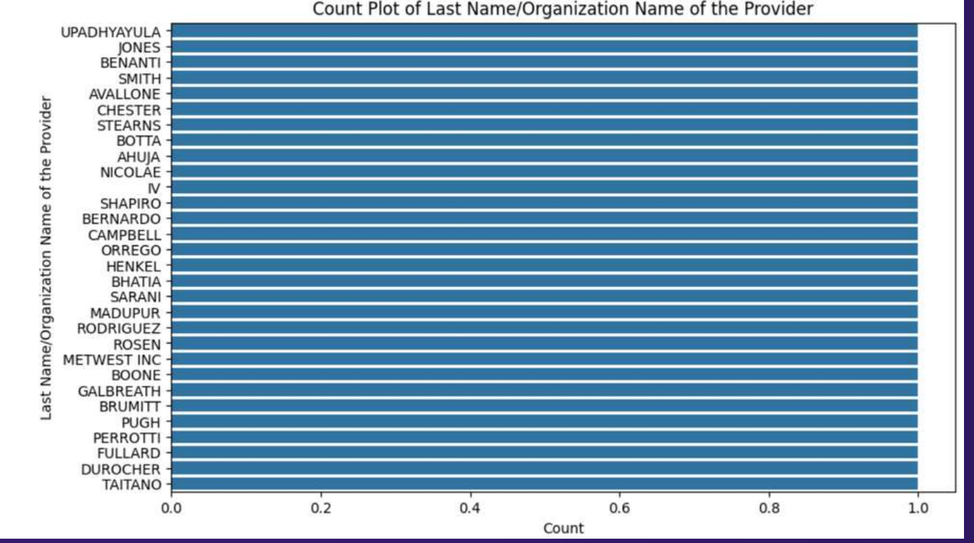
The histogram with a smooth blue line shows a uniform distribution of the index, indicating that the data is not skewed towards any particular value.



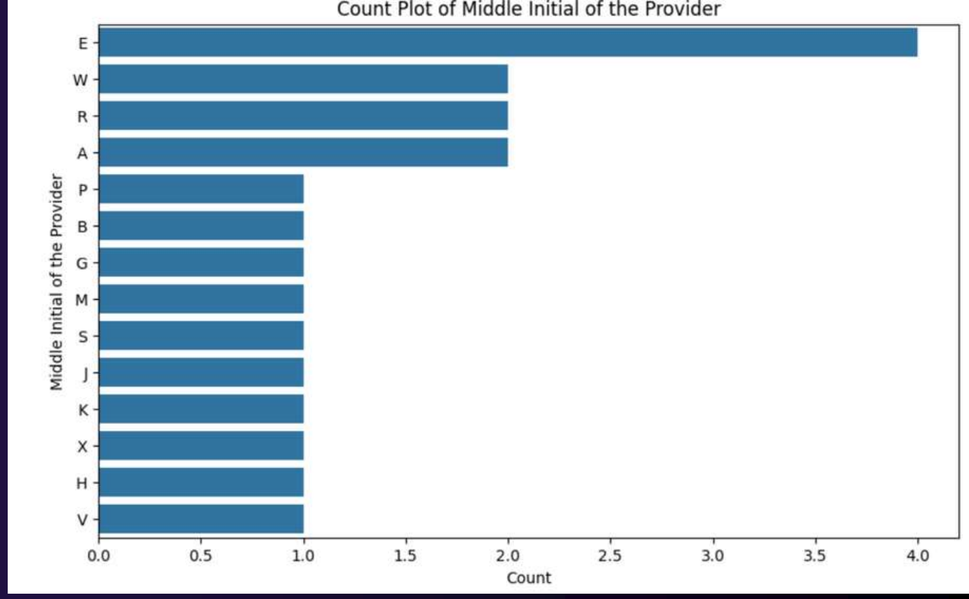
The graph shows the distribution of zip codes of the providers, with a peak around 0 and a few smaller peaks spread across the range.



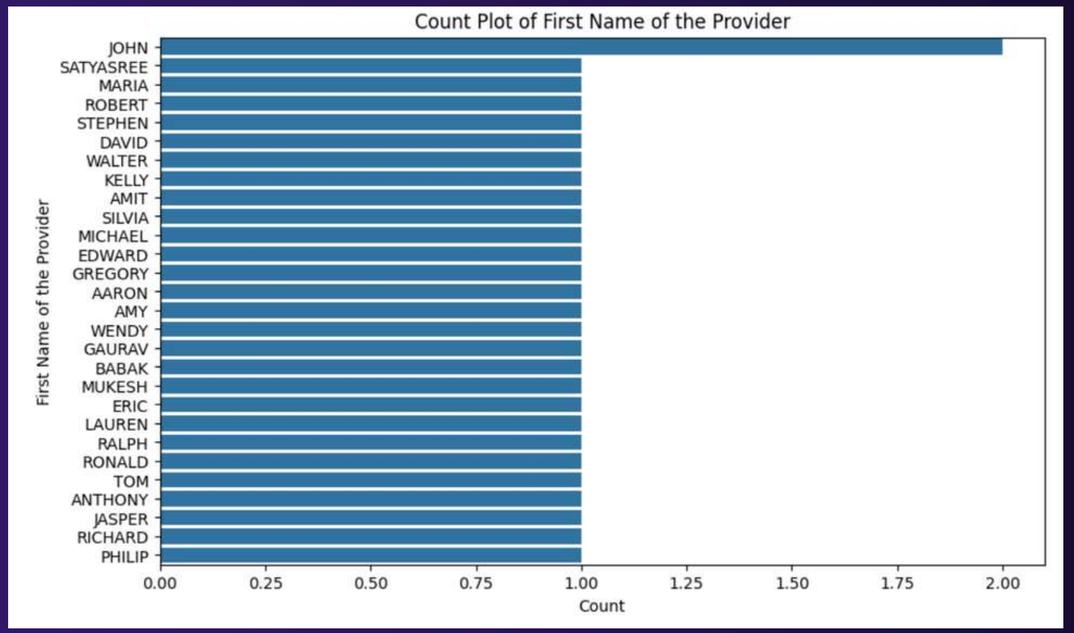
The plot shows the distribution of National Provider Identifier with a histogram and an overlaid density plot.



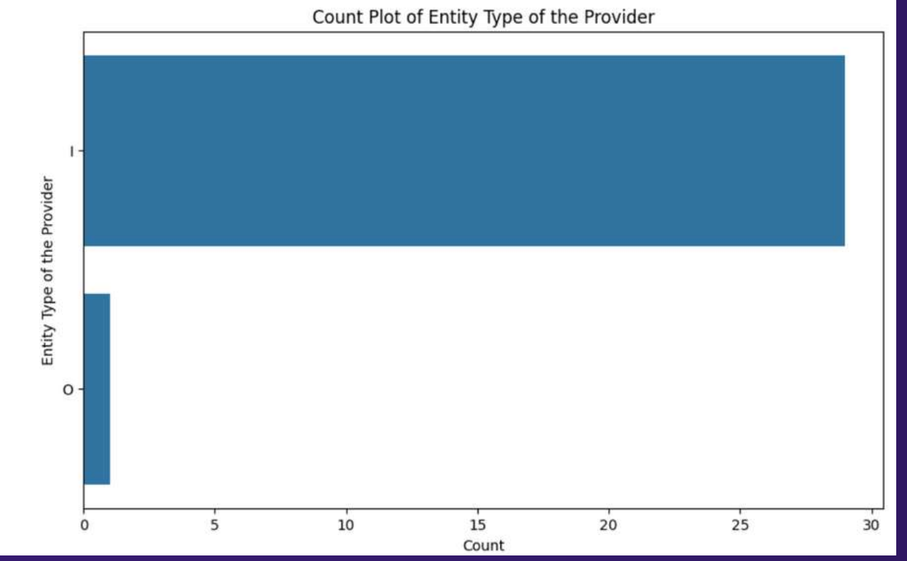
This plot shows the frequency of the first names of the providers.



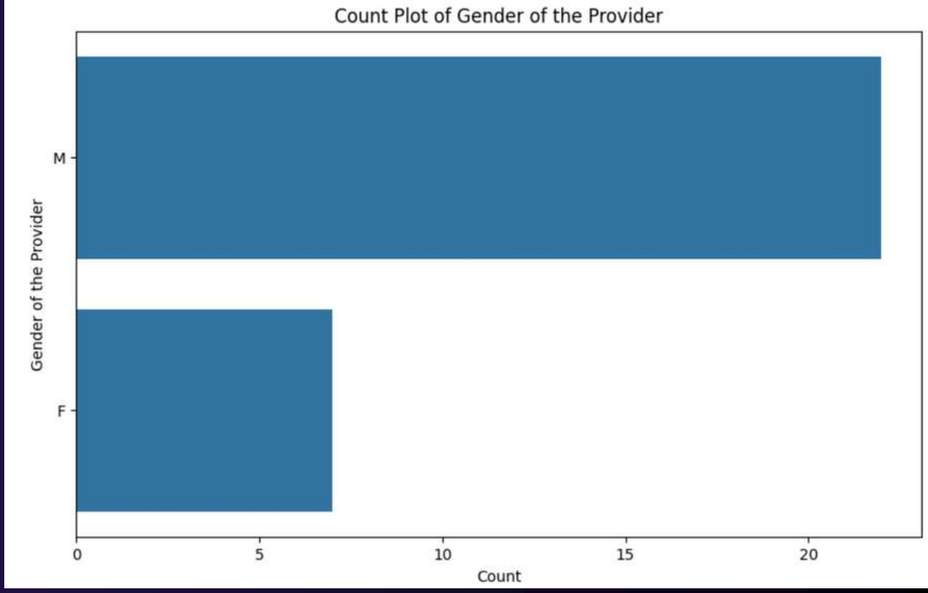
This is a count plot of the number of times each last name/organization name appears in the dataset. The y-axis is the last name/organization name and the x-axis is the count.



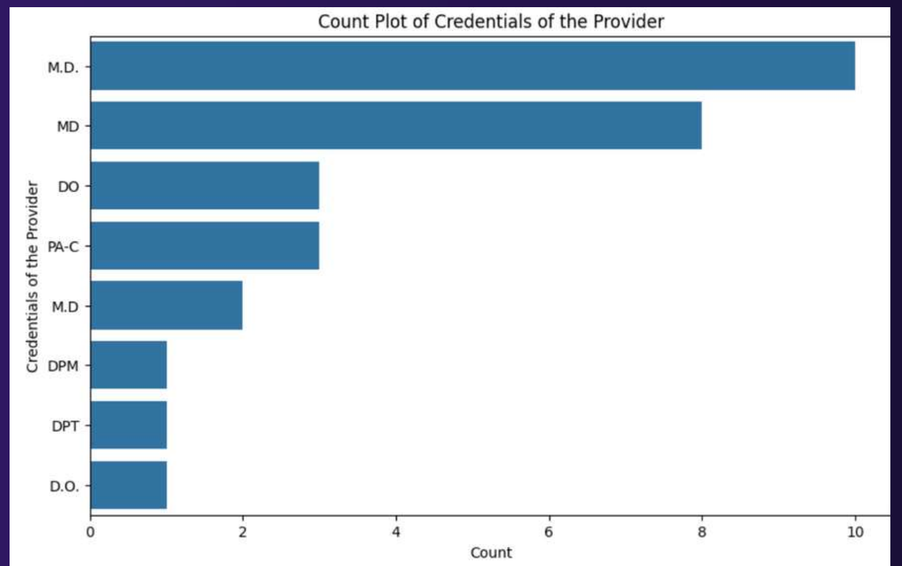
This plot shows the count of each middle initial of the providers.



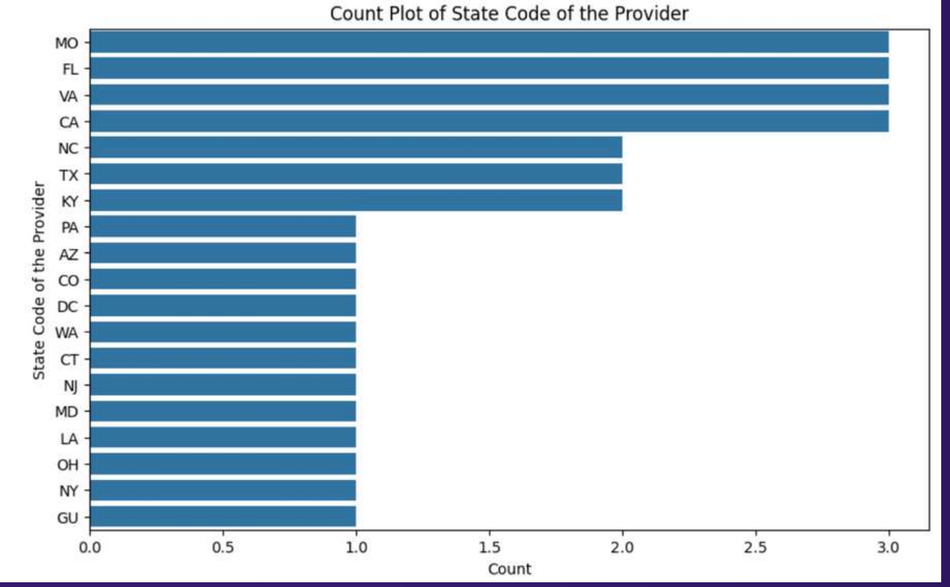
The chart shows the distribution of entity types of providers. The most common entity type appears to be 'I', and there are very few providers of type 'O'.



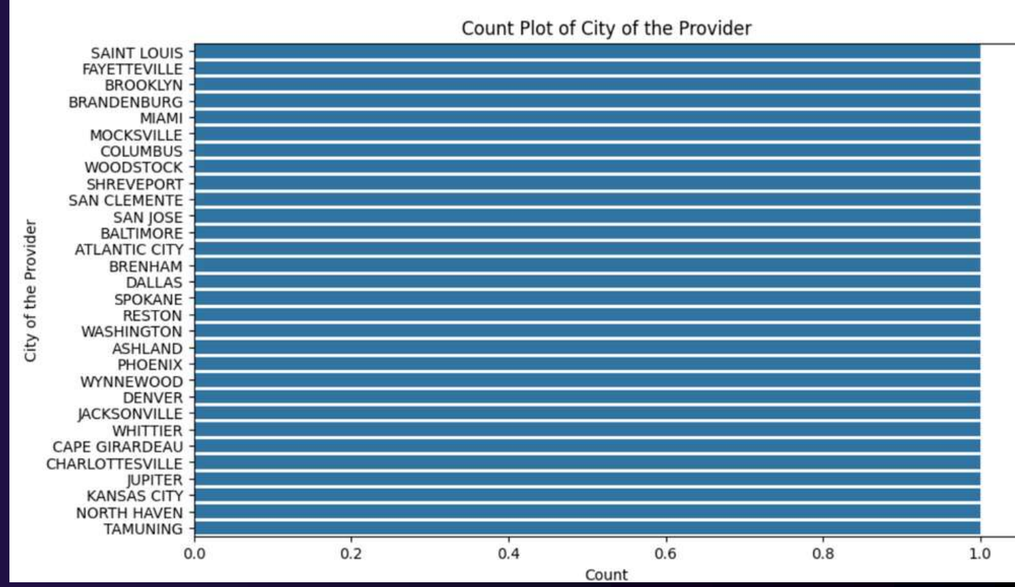
This plot shows the count of providers with each credential.



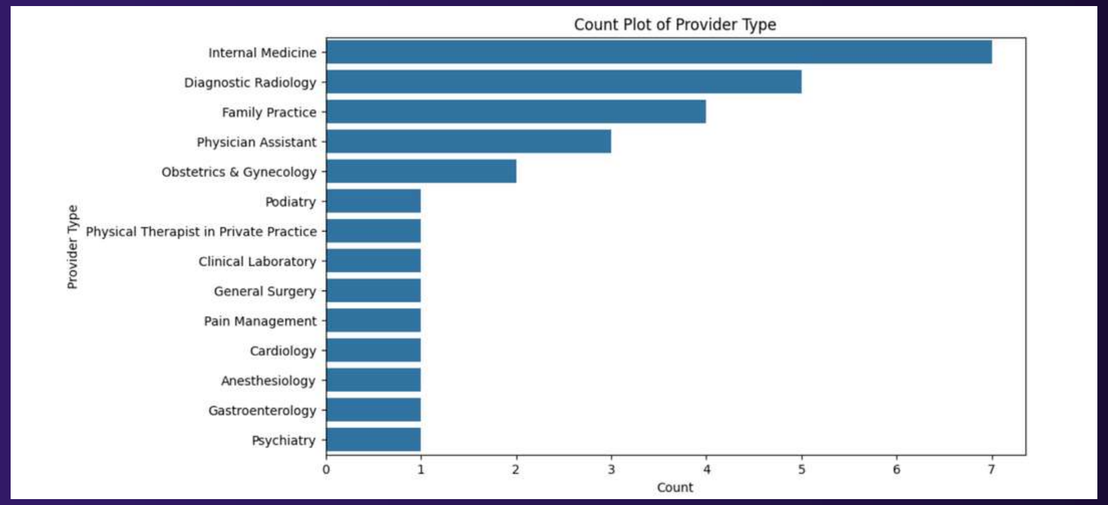
The graph shows the count of male and female providers, with male providers being the dominant gender.



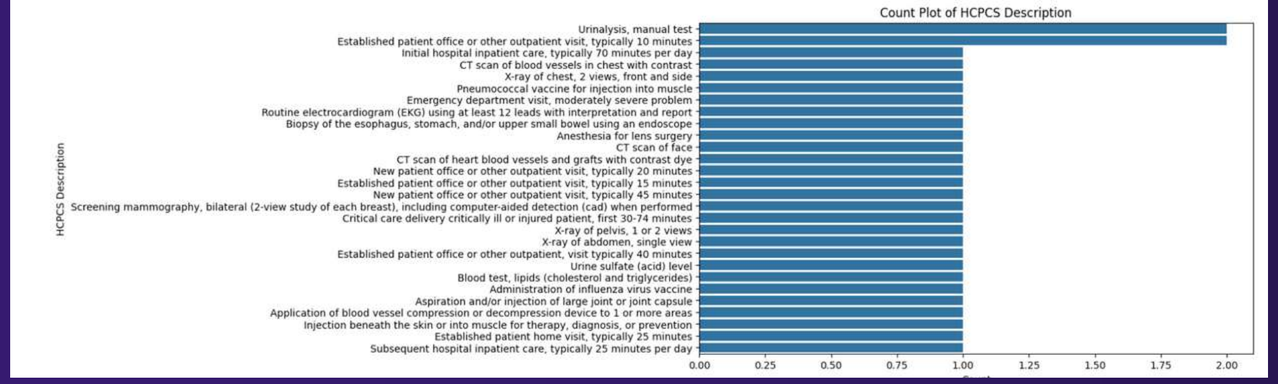
The plot shows the count of each type of provider.



The plot shows the frequency of different US states in a dataset, with MO being the most frequent and GU being the least frequent.

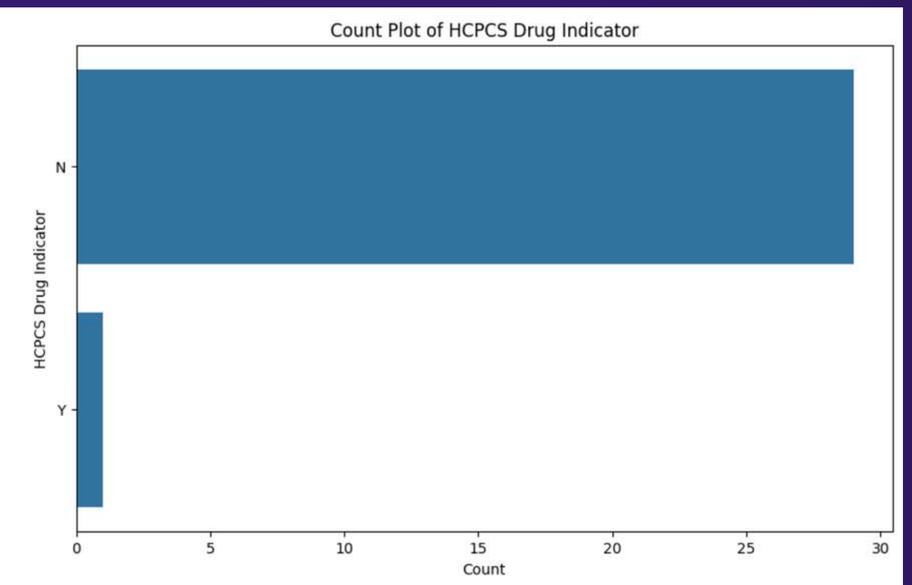


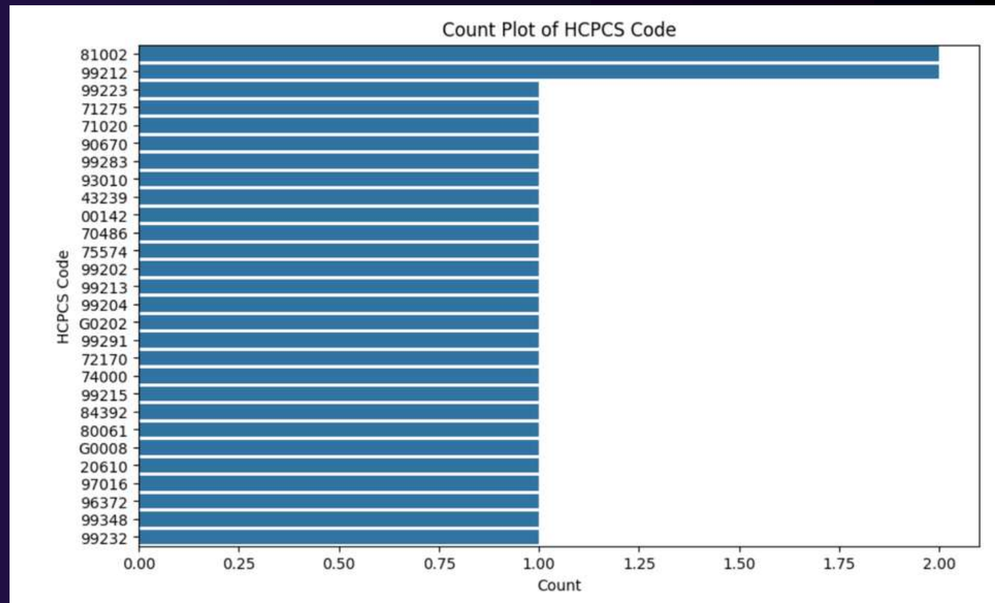
This plot shows the counts for each of the cities of the providers.

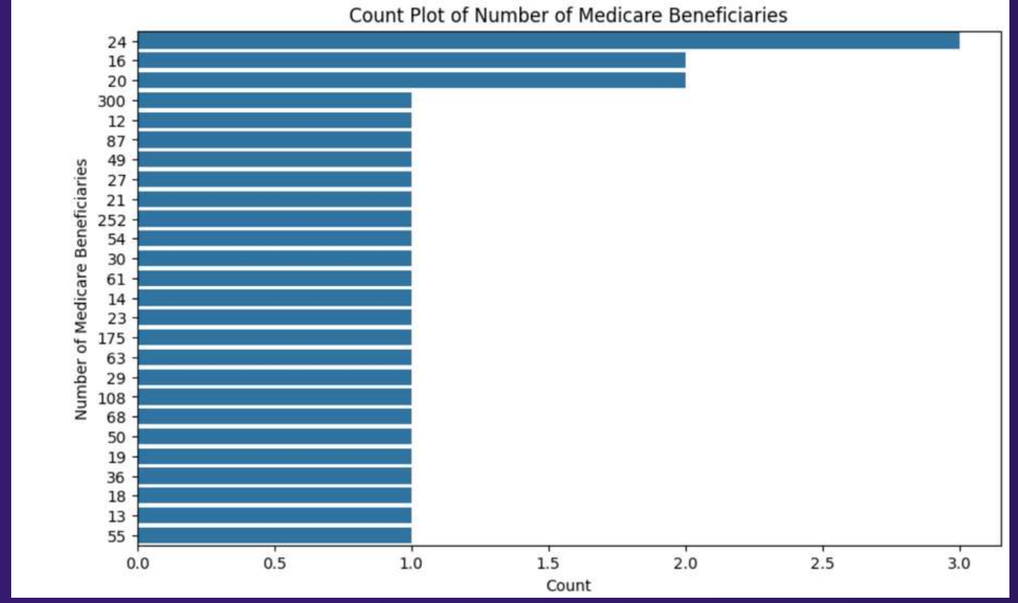


The chart shows the count of different medical procedures described in the HCPCS code.

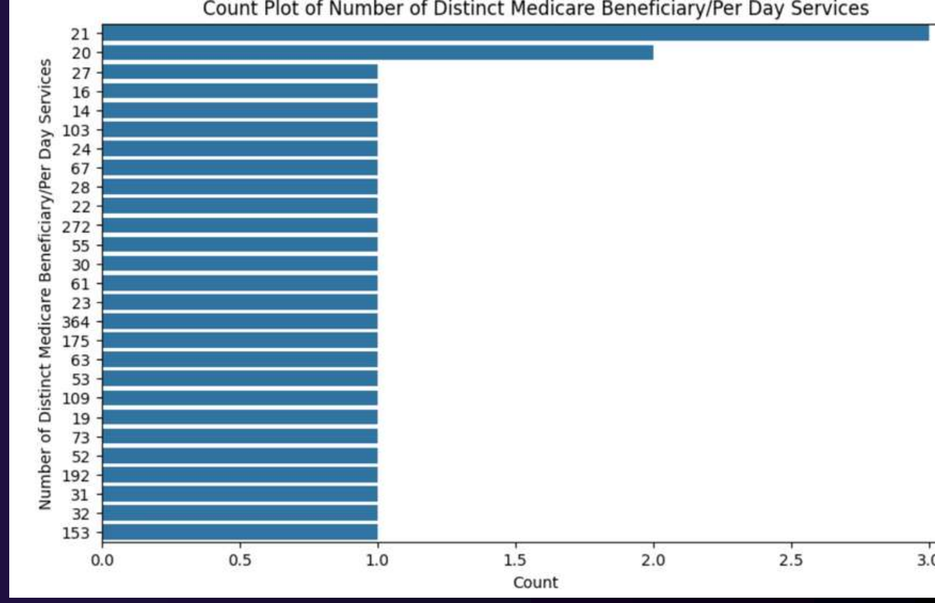
The graph shows that almost all the HCPCS codes are not drug related.



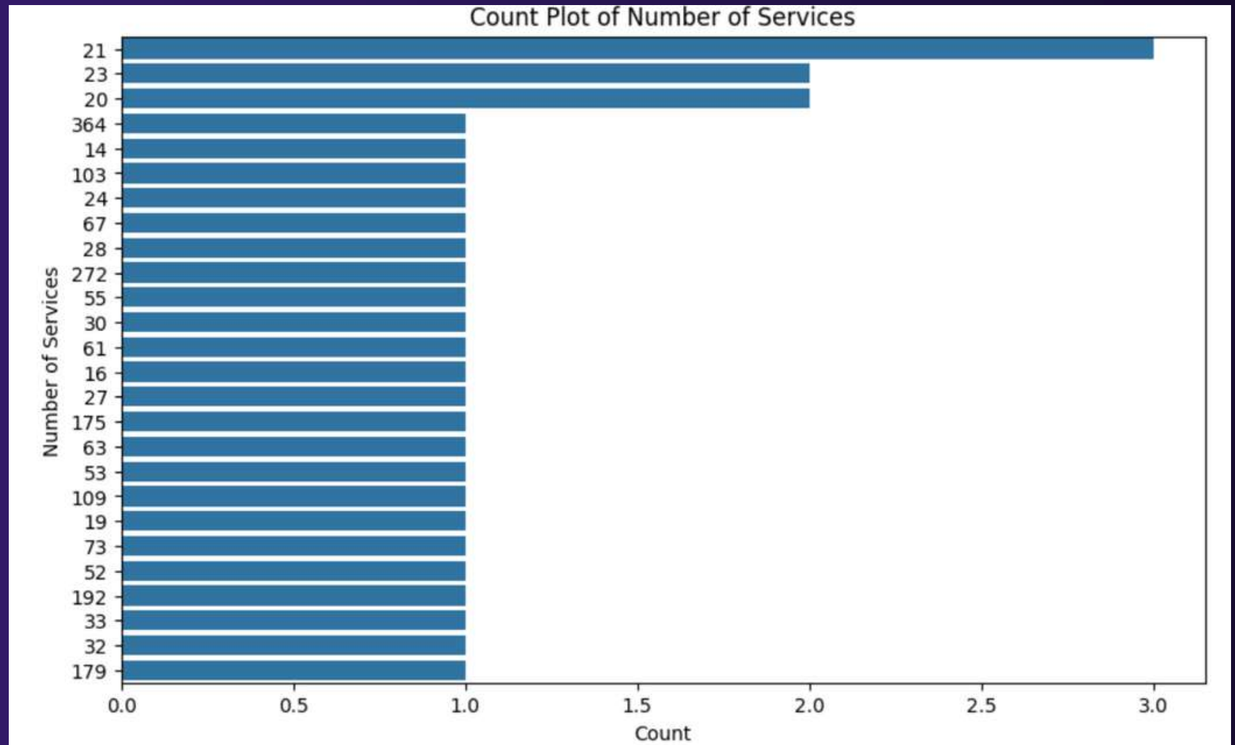




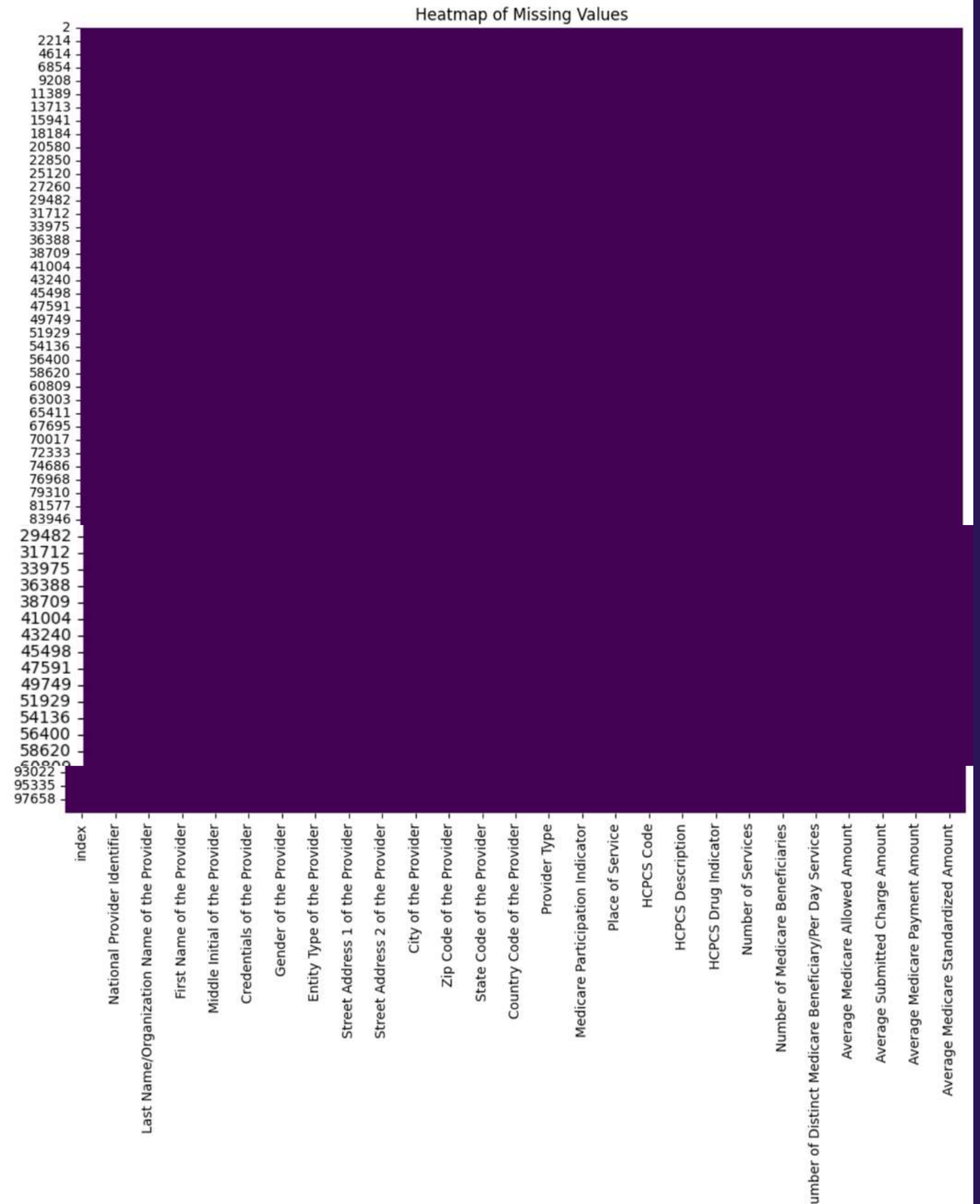
The plot shows the number of times each number of Medicare beneficiaries appears in the data.



The graph shows the count of each number of services provided.



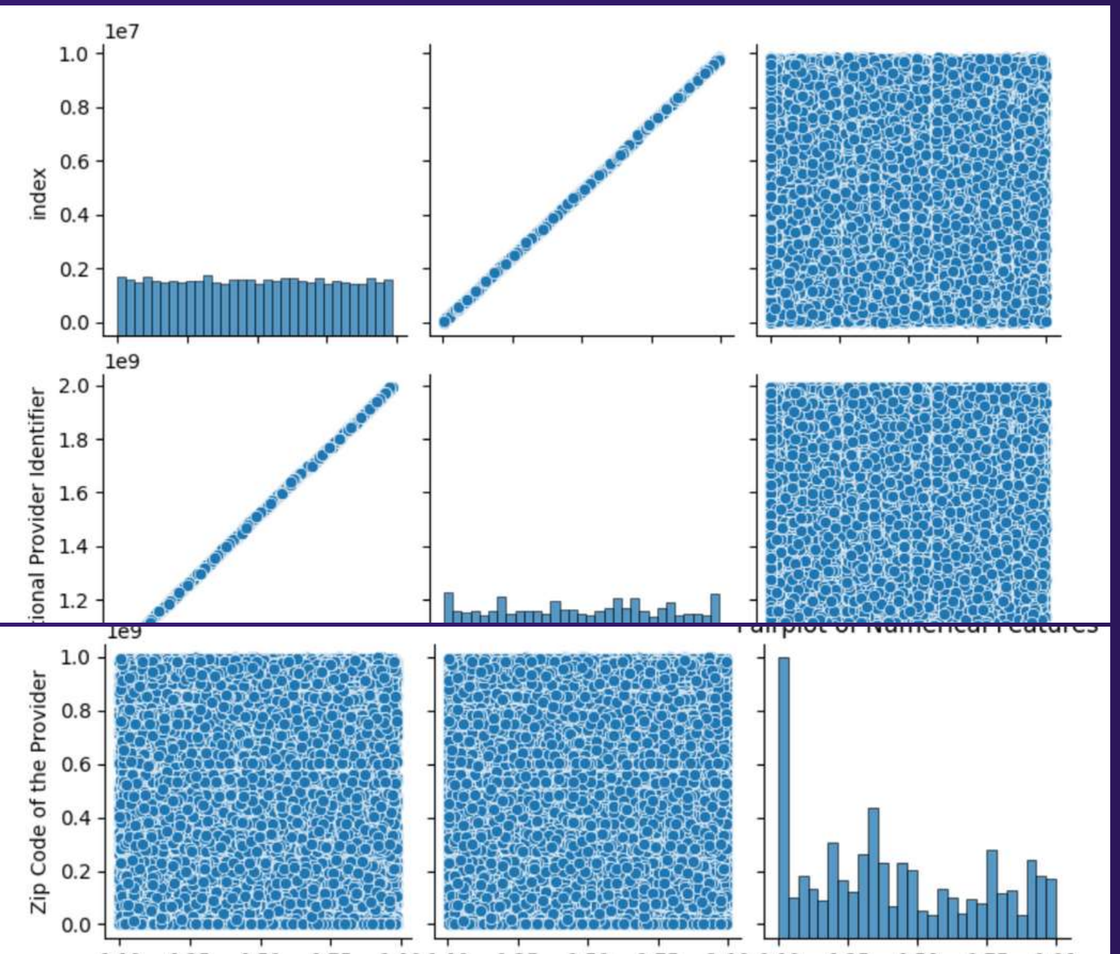
This graph shows how many days have a given number of distinct Medicare beneficiaries.



This is a heatmap of missing values, which is a graphical representation of the missing data points in a dataset. The darker the color, the more missing values there are in that particular column or row.

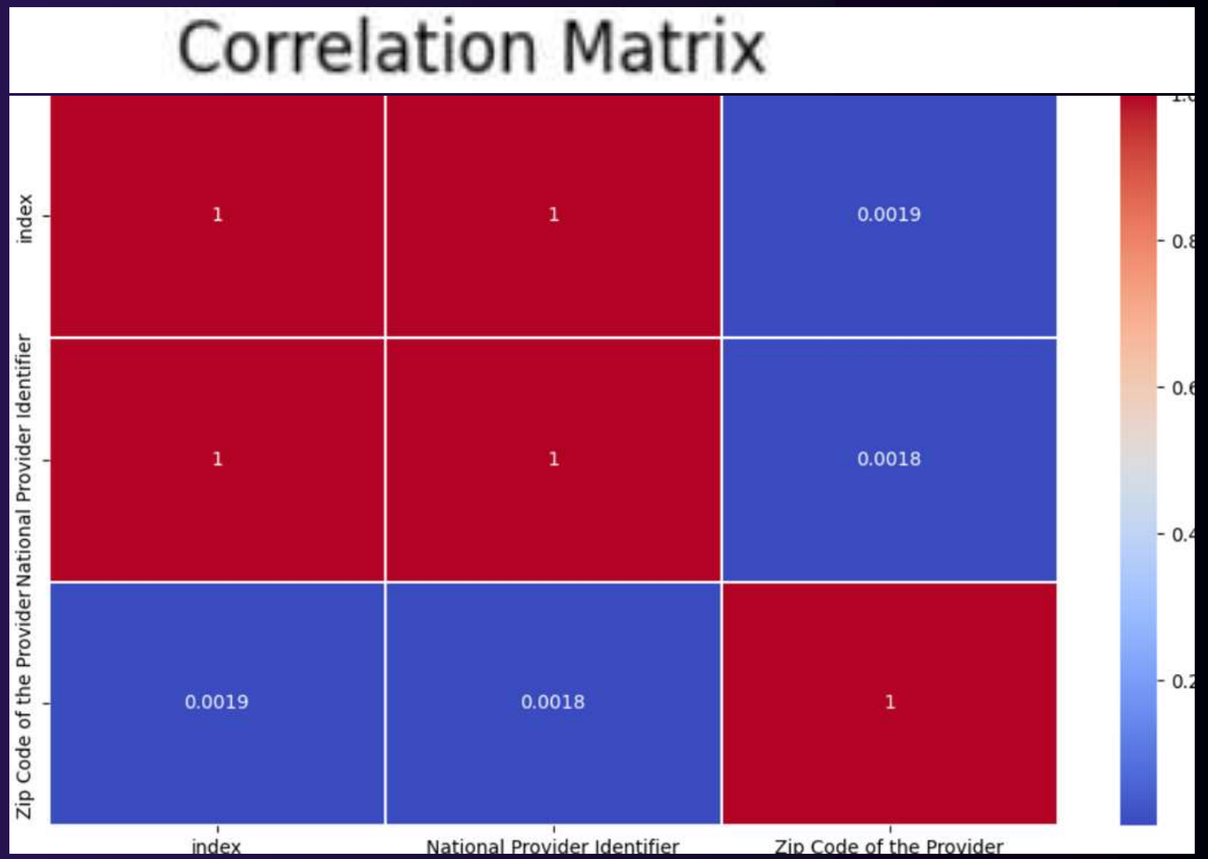
visualize relationships between numerical features

The image visualizes relationships between numerical features in a dataset, showing correlations and distributions. Scatter plots reveal positive correlations between some features, such as National Provider Identifier and Zip Code of the Provider. The plot helps identify relationships and patterns in the data.

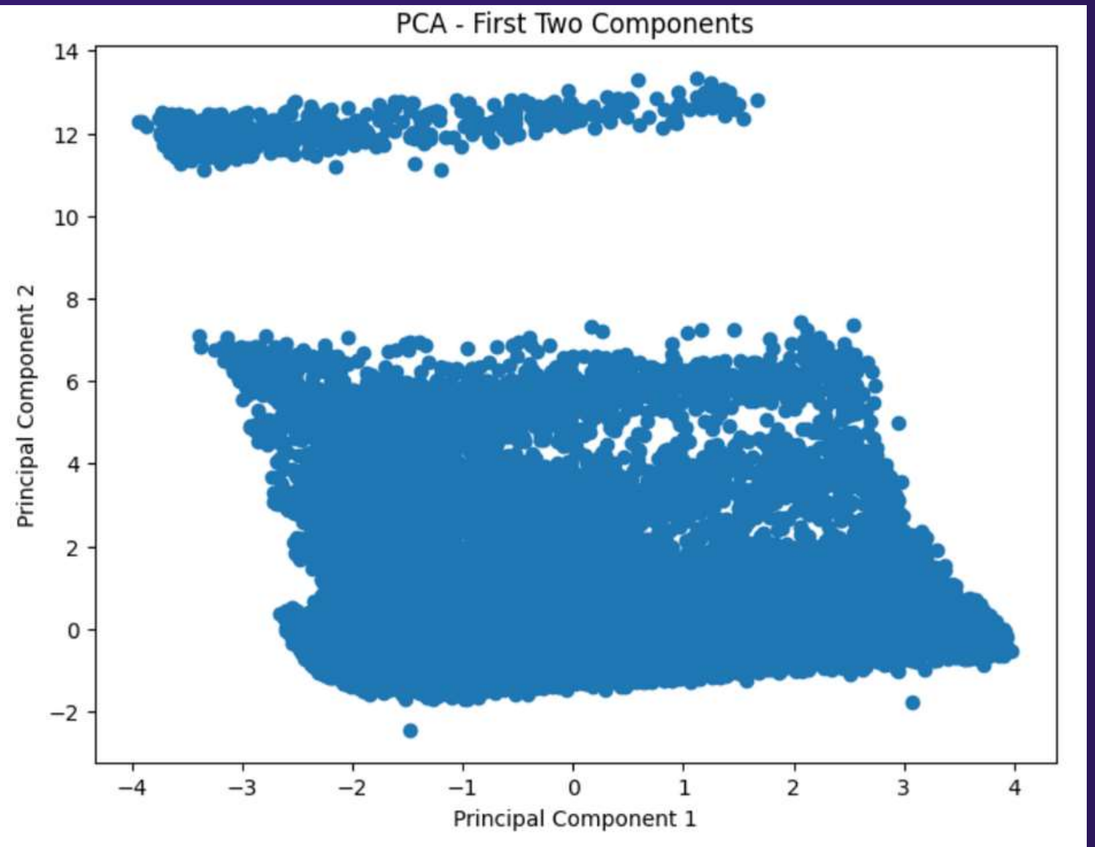


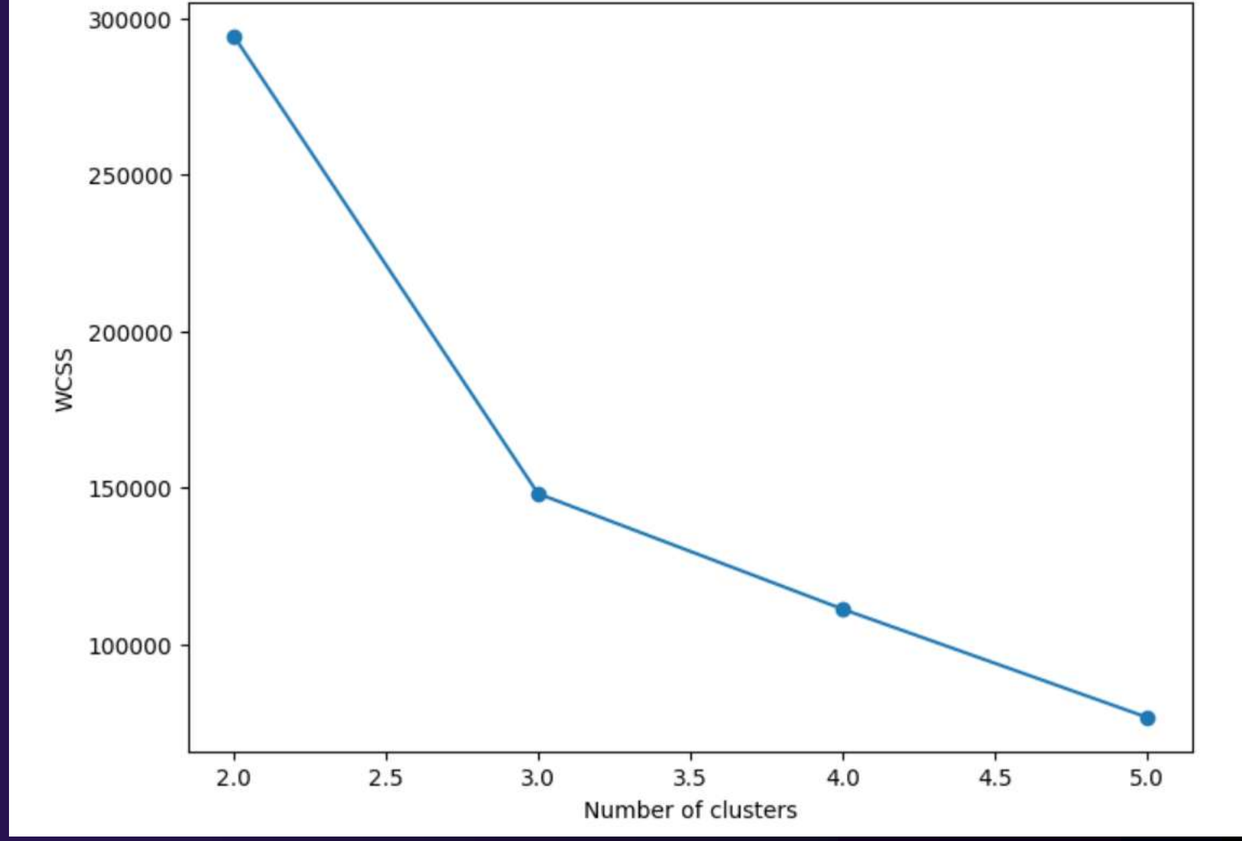
Correlation matrix for numerical features

The correlation matrix shows the correlation between numerical features in the dataset. It indicates that there is a very weak positive correlation between "Zip Code of the Provider" and "National Provider Identifier" features.

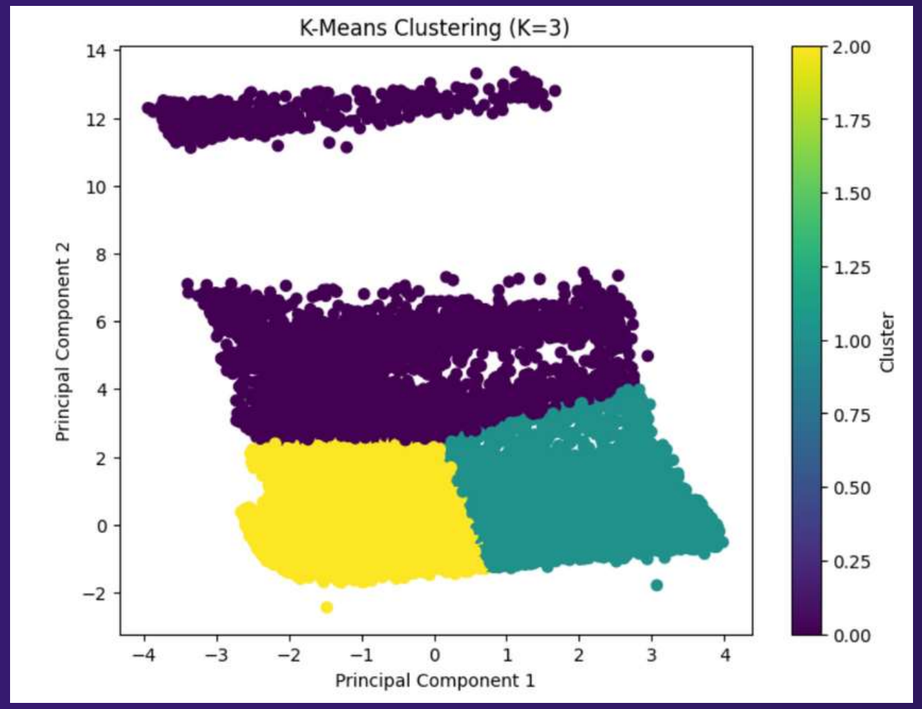


The image shows a scatter plot of the first two principal components of a dataset, with two distinct clusters. The plot helps understand the relationships between variables and identify important features in the data.



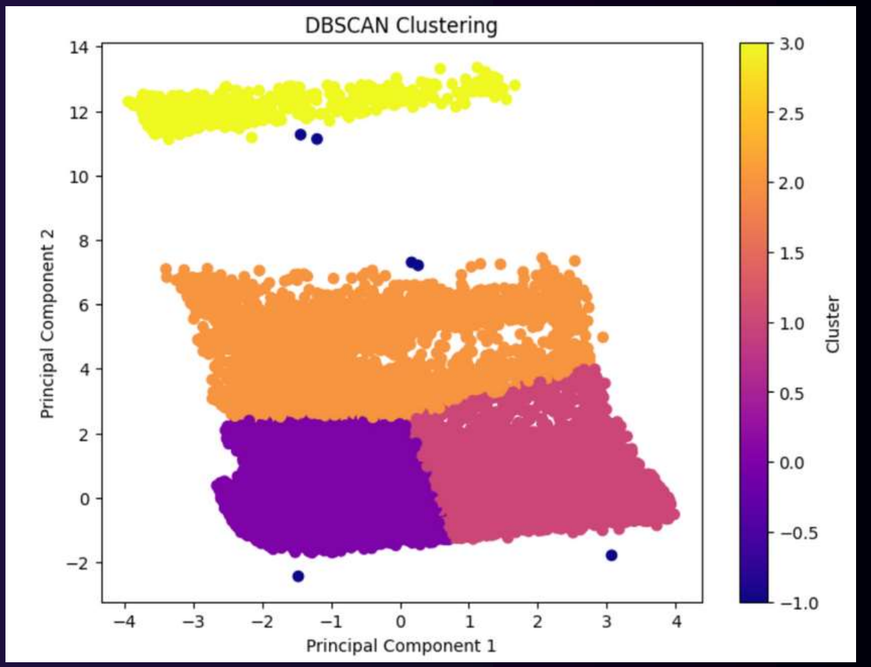


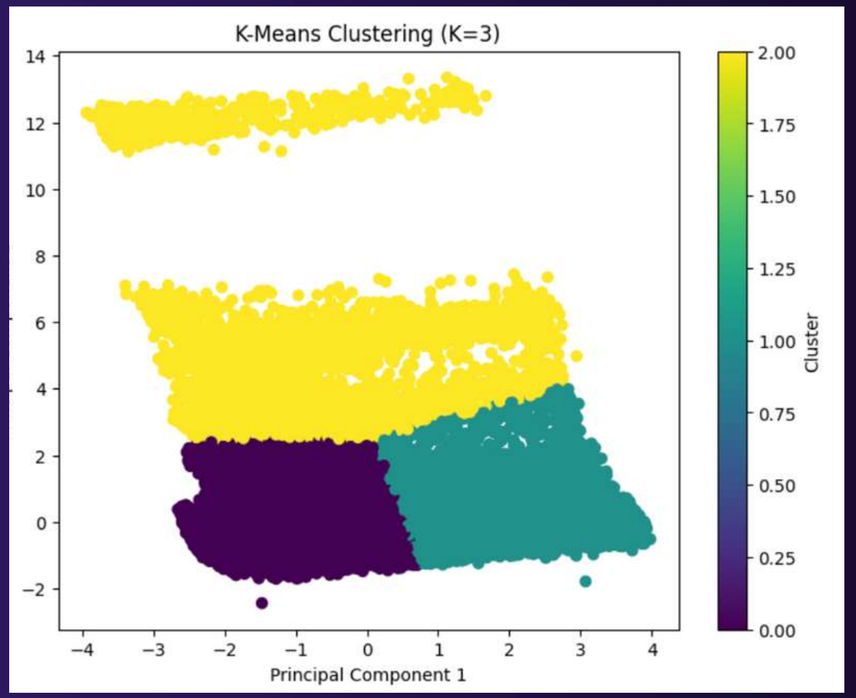
The plot shows the Elbow method to determine the optimal number of clusters (k) for a dataset. The optimal k is where the decrease in WCSS (Within-Cluster Sum of Squares) becomes less significant, which appears to be at k = 3 in this case.



The plot shows a K-Means clustering with K=3. The data points are colored according to their cluster, which are three groups of data that have been identified based on their similarity.

The image shows a scatter plot of data points clustered using the K-Means algorithm with K=3. The data has been reduced to two dimensions using principal component analysis. The three clusters are color-coded. The plot suggests that the clusters are well separated in the feature space.

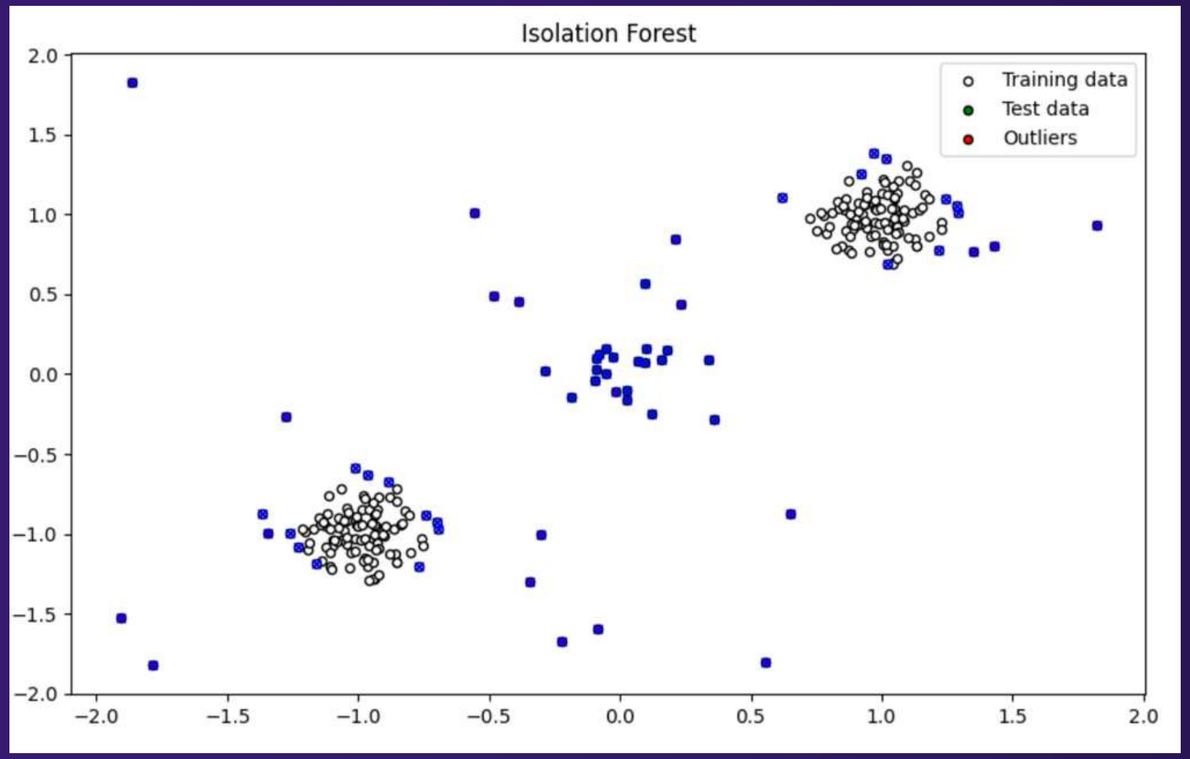




The image shows the result of applying DBSCAN clustering to a dataset, with 4 clusters identified in a 2D space. The clusters are well-separated, with noise points represented in dark blue.

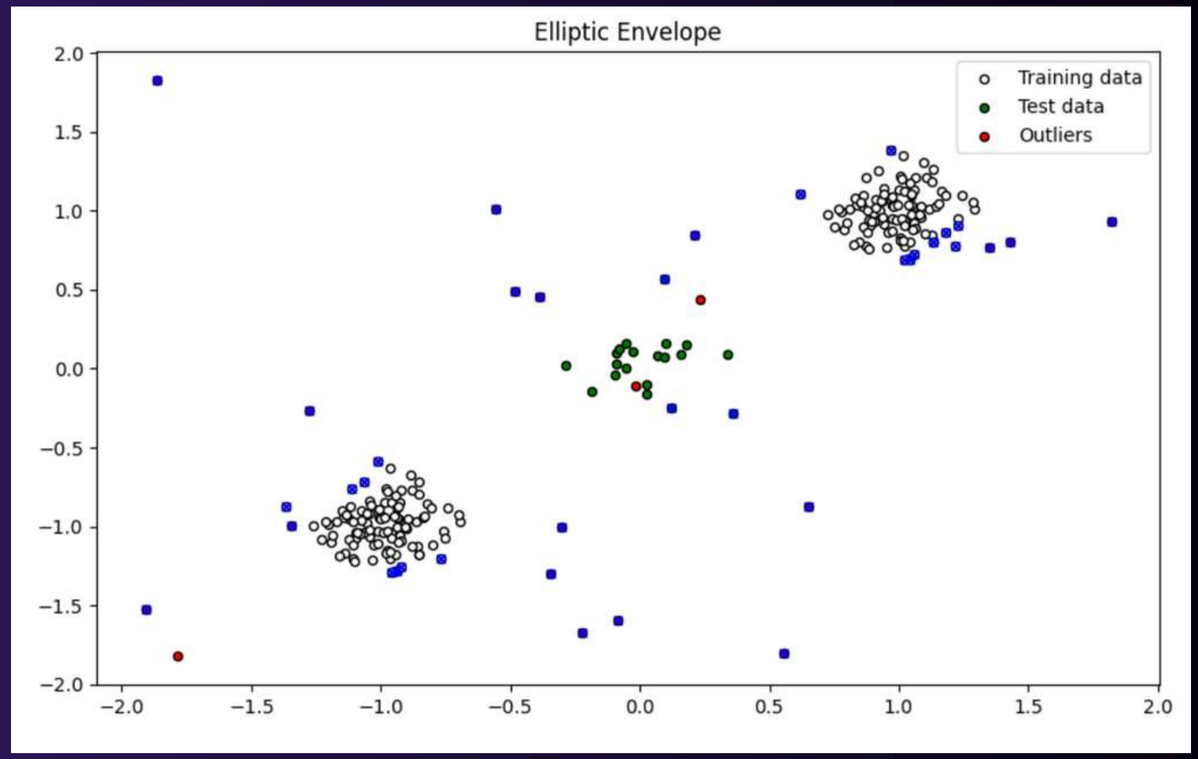
Isolation Forest

This code snippet demonstrates using the Isolation Forest algorithm to detect anomalies in a synthetic dataset. First, a dataset is generated with training, test, and outlier data, combining points clustered around specific centers and uniformly distributed outliers. The data is then standardized using StandardScaler to ensure equal feature contribution. The Isolation Forest model is trained on the standardized training data and subsequently used to predict anomalies in the training, test, and outlier data. Anomalies are labeled as -1, while normal points are labeled as 1. Finally, the results are visualized with a plot that displays the training data in white, test data in green, outliers in red, and detected anomalies marked with blue 'x's. This visualization helps illustrate how effectively the model identifies anomalies across different datasets.



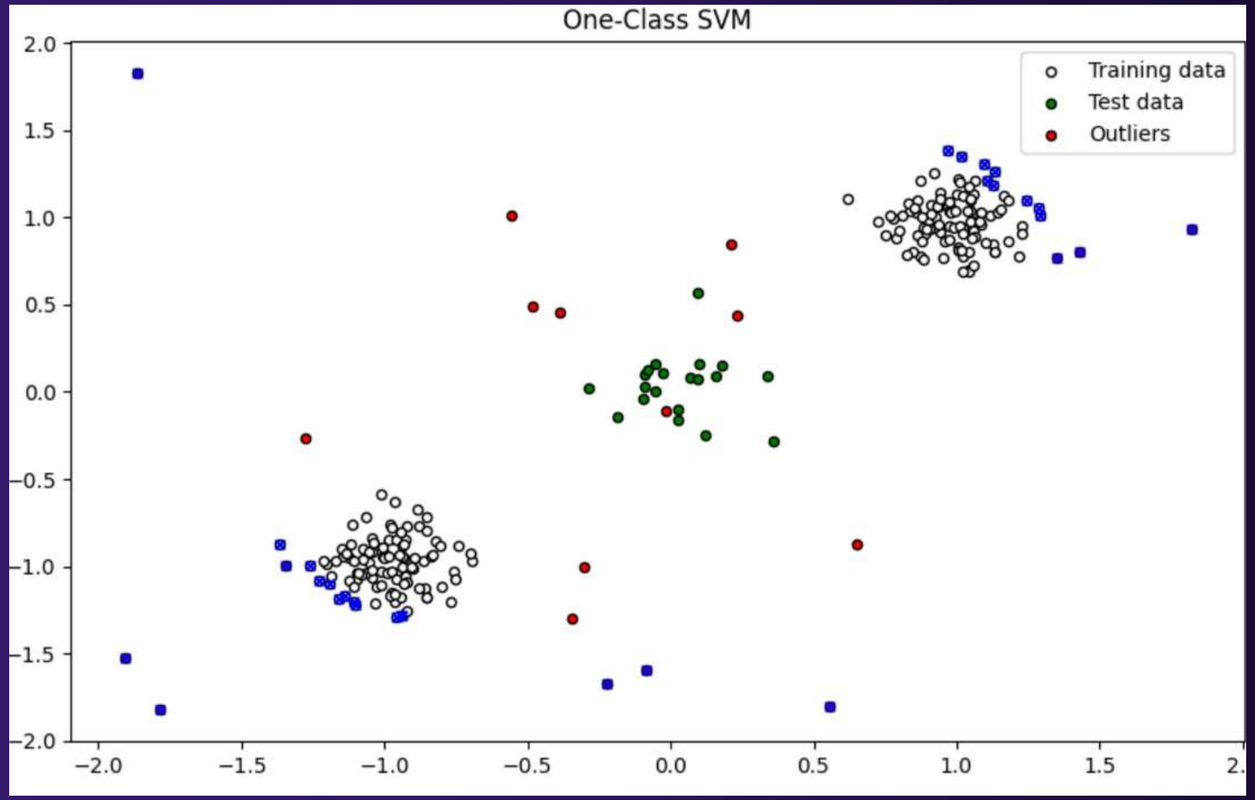
Elliptic Envelope

This code demonstrates using the Elliptic Envelope algorithm to detect anomalies in a synthetic dataset. A dataset is generated with training, test, and outlier data, which is then standardized using StandardScaler to ensure consistent feature scaling. The Elliptic Envelope model is trained on the standardized training data and predicts anomalies in the training, test, and outlier datasets. Anomalies are labeled as -1, while normal points are labeled as 1. The results are visualized in a plot, where training data points are shown in white, test data in green, outliers in red, and detected anomalies marked with blue 'x's. This visualization illustrates the model's effectiveness in identifying anomalies across different data sets.



One-Class SVM

This code snippet demonstrates using the One-Class SVM algorithm to detect anomalies in a synthetic dataset. It starts by generating a dataset with training, test, and outlier data. The data is then standardized using StandardScaler to ensure consistent scaling. A One-Class SVM model is trained on the standardized training data and used to predict anomalies in the training, test, and outlier data. The anomalies are labeled as -1, while normal points are labeled as 1. Finally, the results are visualized in a plot where the training data points are shown in white, test data in green, outliers in red, and detected anomalies marked with blue 'x's. This visualization helps to illustrate how well the model identifies anomalies across different datasets.



Deep Learning (DL) Results

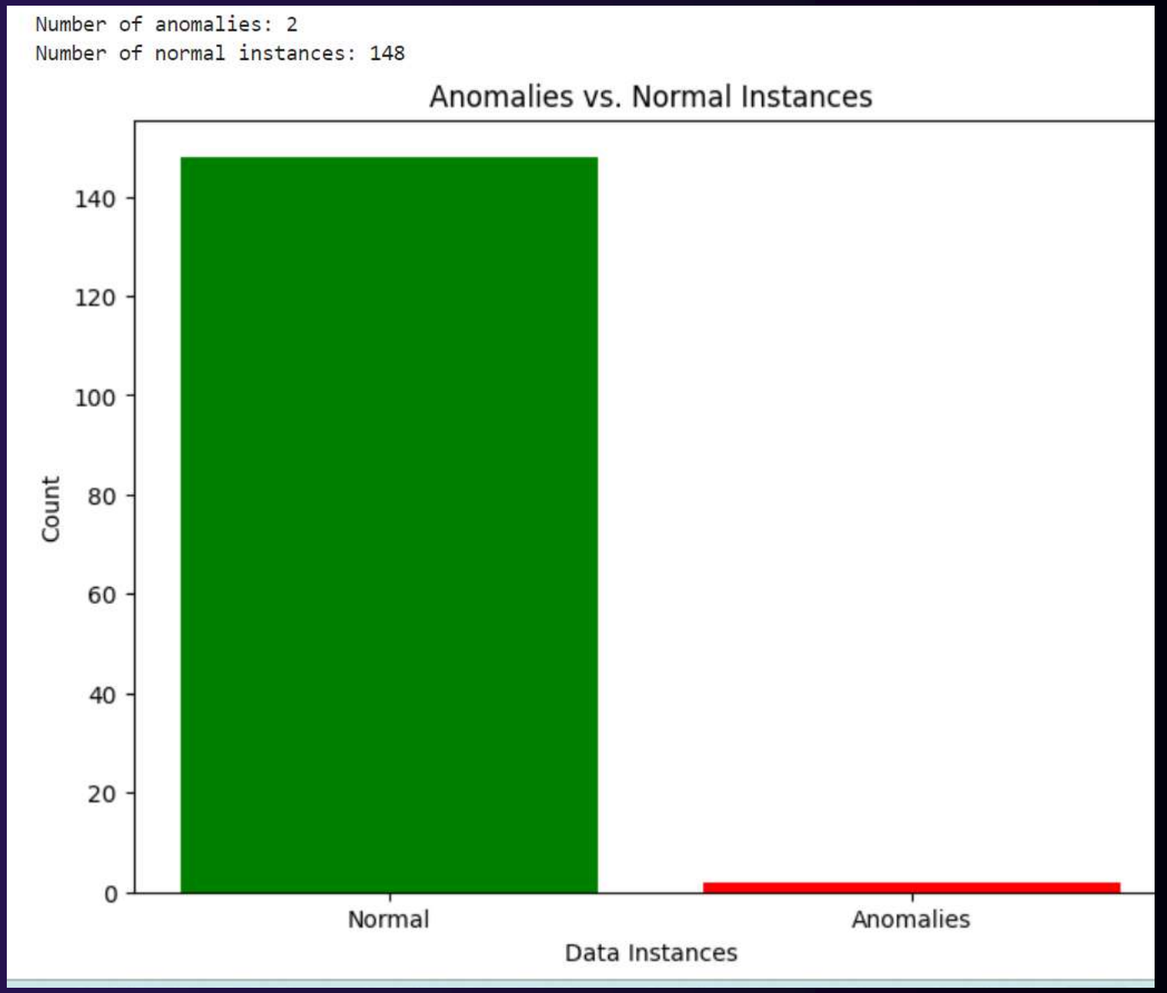
Model: "functional_9"

Layer (type)	Output Shape	Param #
input_layer_18 (InputLayer)	(None, 4)	0
dense_142 (Dense)	(None, 64)	320
dense_143 (Dense)	(None, 32)	2,080
dense_144 (Dense)	(None, 16)	528
dense_145 (Dense)	(None, 11)	187
dense_146 (Dense)	(None, 16)	192
dense_147 (Dense)	(None, 32)	544
dense_148 (Dense)	(None, 64)	2,112
dense_149 (Dense)	(None, 4)	260

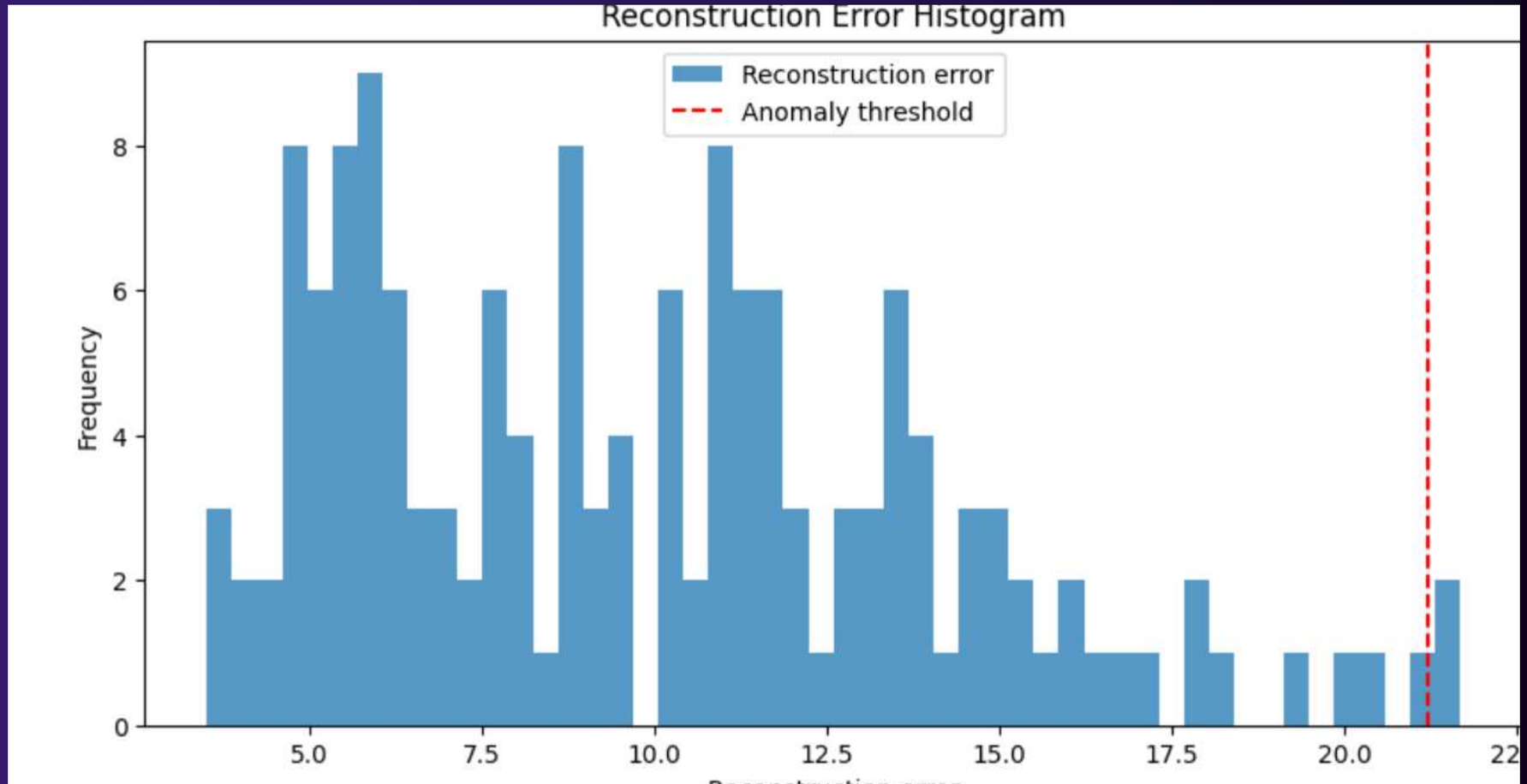
Total params: 6,223 (24.31 KB)
Trainable params: 6,223 (24.31 KB)
Non-trainable params: 0 (0.00 B)

```
Epoch 1/100
4/4 ━━━━━━━━━━━━━━━━━━━━ 2s 39ms/step - loss: 11.6308 - val_loss: 17.7327
```

```
Number of anomalies: 2
Number of normal instances: 148
```

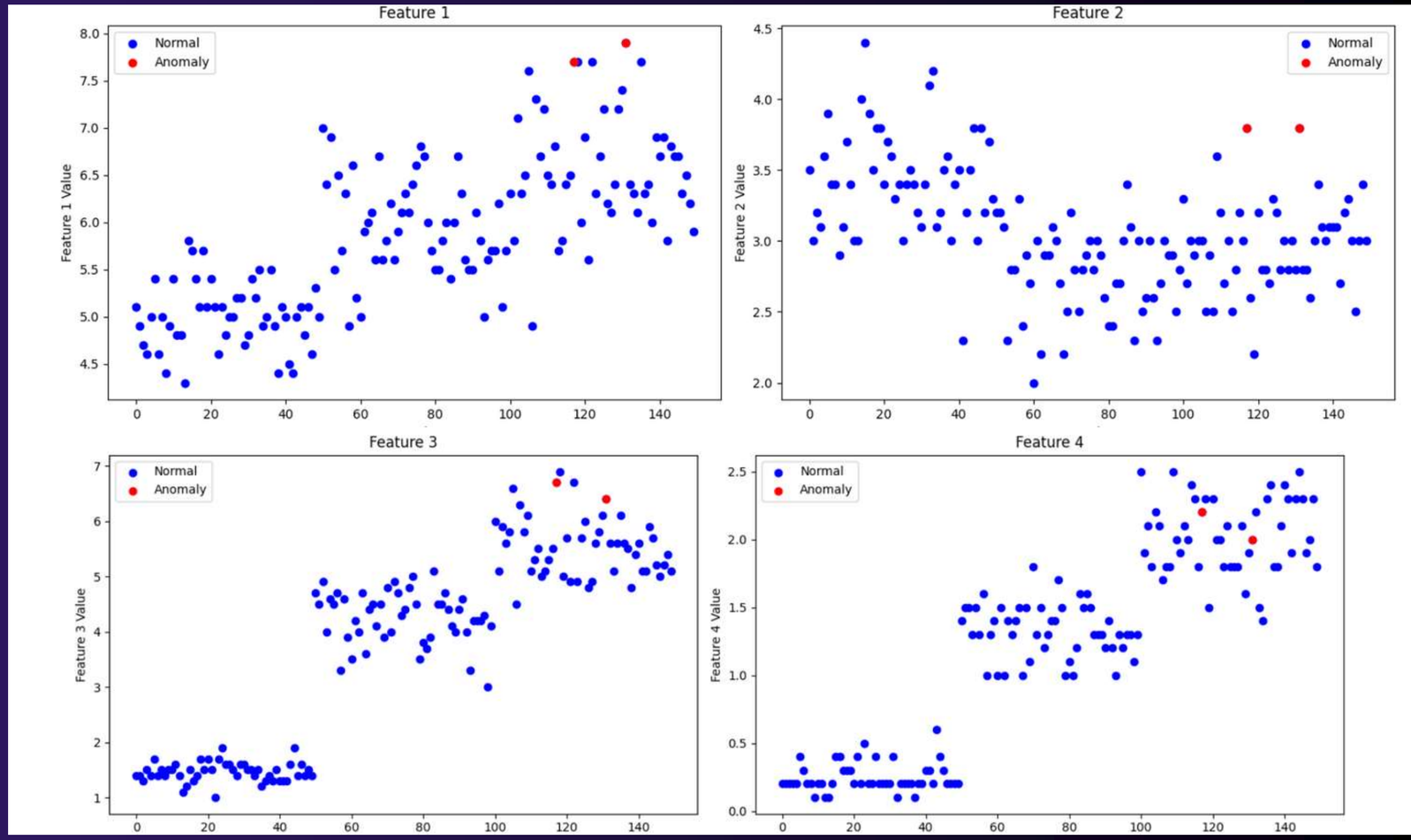


The histogram shows the distribution of reconstruction errors. The reconstruction error is a measure of how well the autoencoder can reconstruct the original data. The higher the reconstruction error, the less accurate the reconstruction. The red dashed line represents the anomaly threshold. The data points that are above the threshold are considered anomalous. In this case, the threshold is set to 21. It is possible that the anomalies in this dataset will correspond to data points that have a reconstruction error greater than 21.



Scatter Plot

The image shows a scatter plot of four features. The blue dots represent the normal data points, and the red dots represent the anomalous data points. It seems like feature 1 and feature 3 are most sensitive to anomalies. The feature 2 and feature 4 have very few anomalies. The anomalies are more spread out in features 1 and 3 than in features 2 and 4.



THANK YOU