Identifying Outliers in the Data Using Variational Autoencoders

Problem Statement

The objective of this project is to identify outliers in a dataset using Variational Autoencoders (VAEs). The process follows these steps:

1. **Compressing the Data**: Using VAEs to reduce high-dimensional input data to a lower-dimensional latent space.

2. **Clustering in Latent Space**: Applying K-Means clustering to group similar data points.

3. **Outlier Detection**: Identifying anomalies based on clustering results:

    - **Boundary points** of large clusters.

    - **Points in small clusters**, which may represent outliers.

Input Data

The dataset consists of the following columns:

cov1, cov2, cov3, cov4, cov5, cov6, cov7, sal_pur_rat, igst_itc_tot_itc_rat, lib_igst_itc_rat

These features were used as inputs to the Variational Autoencoder model.

Methodology and Hyperparameter Tuning

We experimented with different hyperparameters to optimize the VAE model. The tested values were:

- **Latent dimensions**: [2, 3]

- **Hidden layer sizes**: [[32], [128, 32], [64], [128], [64,32], [128,64]]

- **Learning rates**: [0.001 ,0.002 ,0.003]
- **Batch sizes**: [16, 32]
- **Epochs**: [50, 100]

After evaluating different configurations, the best-performing hyperparameters were:

- **Latent dimensions**: 2 (data compressed into a 2D space).
- **Hidden layer sizes**: [128, 32] and [128,64].
- **Learning rates**: 0.001 and 0.002.
- **Batch sizes**: [16, 32].
- **Epochs**: 50.

Why These Hyperparameters Were Selected

1. Latent Dimensions: 2

- A latent dimension of **2** allows the VAE to compress the data while still preserving meaningful structures in the latent space.
- Since **clustering and visualization** are involved, a **2D latent space** helped in better **separation of clusters** and improved interpretability.
- A higher latent dimension (e.g., 3) led to **overfitting**, with fewer data points per dimension, reducing clustering quality.
- Since the data is now spread across 3 dimensions instead of 2, there are **fewer data points per dimension**, making it harder for K-Means to find **clear cluster boundaries**.

2. Hidden Layer Sizes: [128, 32] and [128, 64]

- **Deeper networks** (128 → 32 and 128 → 64) capture more complex features, improving the model's ability to learn expressive representations.
- **Avoids overfitting and underfitting**:
    - Too **large** networks (e.g., [256, 128]) resulted in overfitting.

    - Too **shallow** networks ([32]) resulted in underfitting.

Clustering and Outlier Detection

After reducing dimensionality using VAE, we performed K-Means clustering. The optimal number of clusters **(best_k)** was determined using the **Silhouette Score**.

Why we chose **Silhouette** over **Elbow** method?

- The dataset had **overlapping clusters** so the Elbow Method didn't show a clear "elbow."
- **Silhouette Score works better** in such cases because it considers both **compactness and separation of** clusters**.**
- We observed the formation of **2-3 small clusters**.
- Outliers were identified as:
    - **Boundary points** in large clusters.
    - **Points in smaller clusters**, which likely indicate anomalies.
- The number of outliers varied in the range of **80 to 150** across different iterations.

Results and Visualization

We conducted multiple iterations and consistently observed a stable pattern of outliers. The final clustering results and outlier distributions were plotted to visually validate the findings.