Unsupervised Learning project report

Aly Valiev
Student ID: 911861
University of Milano-Bicocca, Milan
Email: a.valiev@campus.unimib.it

***Abstract*—This project explores anomaly detection within a dataset containing mixed data types, specifically binary and continuous variables. We preprocess the data. Then four approaches were implemented: KPrototypes, an information gain approach with entropy calculation, a hybrid approach combining clustering for continuous data with entropy for binary data, and autoencoder. Distribution of anomaly scores was analyzed for all approaches. Additionally, we created an artificial dataset with labeled outliers and ran the models on it as well.**

1. Introduction and dataset description

Anomaly detection is important for finding unusual patterns in data. Mixed data types, with both continuous and binary attributes, make this task harder and need special methods to find anomalies accurately.

1.1. Dataset exploration

This dataset contains 7200 objects with 21 attributes. Six of them are continuous and 15 are binary. Continuous values lie in the range from 0 to 1. There is no information about attributes and how they are related to real life, so in this project, we treat the data just as arrays of numbers and don't make conclusions about their meaning

Figure 1 shows the distribution of binary attributes. As we can see, they are heavily imbalanced. For example, the binary dimension 14 has only one object whose value is 1 and 7199 zeros. If we look only at the binary columns, there are only 135 unique objects.

Figure 2 shows the distribution of continuous values on a log scale. Some outliers can be seen even visually.

Overall, the anomaly detection task on this dataset presents a significant challenge due to the imbalanced binary values, skewed distribution of continuous values, high dimensionality, and mixed data types.

In addition to the given dataset, we generated an artificial dataset similar to the original, but with known outliers. It is not used to directly evaluate the model, but can give us some understanding of the model's performance.

The dataset consists of 7,200 objects with a mix of continuous and binary attributes. The continuous attributes follow different statistical distributions, while the binary

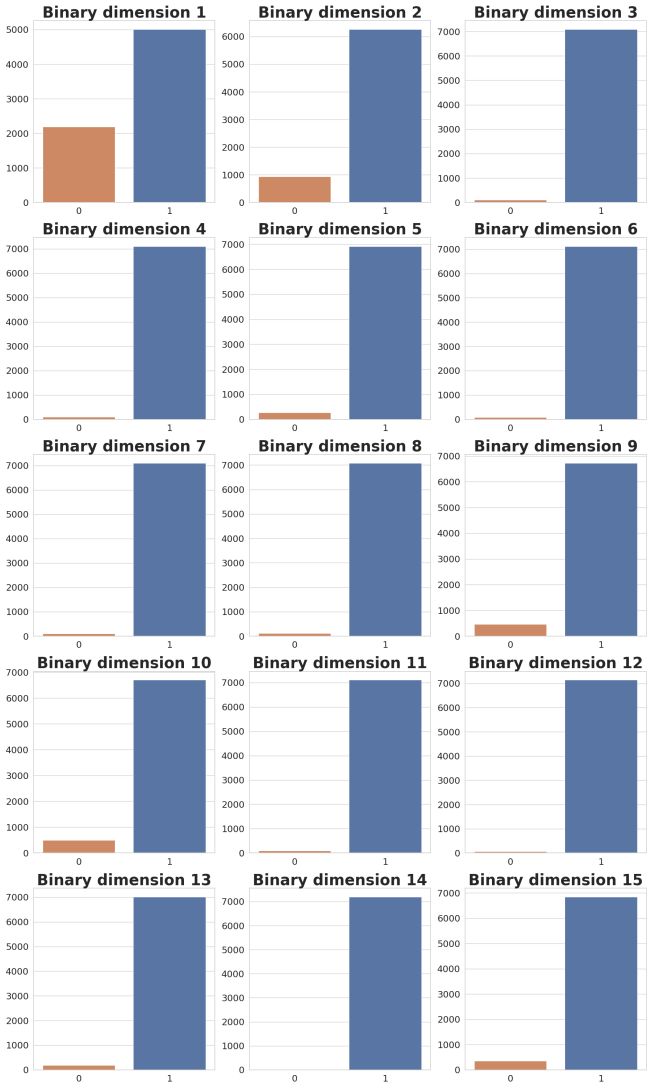


Figure 1: Distribution of binary values.

attributes have varying imbalanced proportions. Six continuous attributes are the following:

- **cont_1**: Normal Distribution
- **cont_2**: Beta Distribution
- **cont_3**: Gamma Distribution

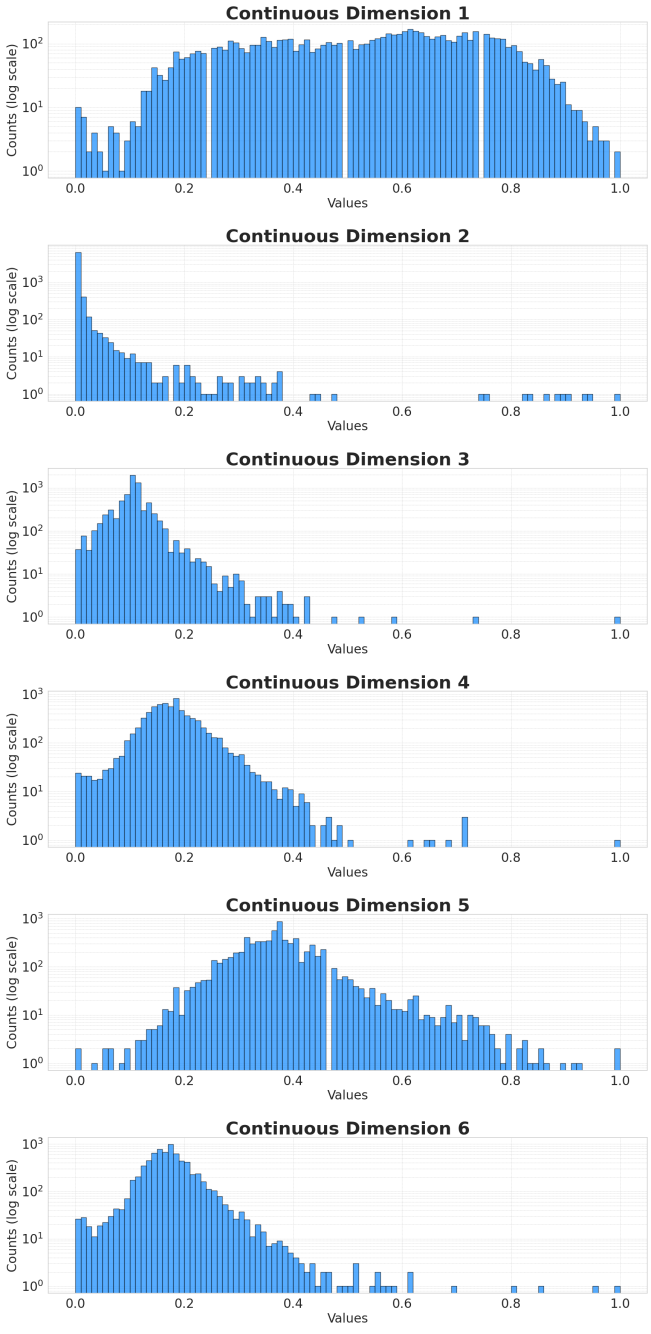


Figure 2: Distribution of continuous values.

- **cont_4**: Log-normal Distribution
- **cont_5**: Gamma Distribution (different from 3rd)
- **cont_6**: Exponential Distribution

All continuous data is normalized to a [0, 1] range for consistency and easier model training.

To test the robustness of the models, 5% of the objects (360) are turned into outliers. For these outliers, binary values are flipped, and Gaussian noise is added to the continuous attributes.

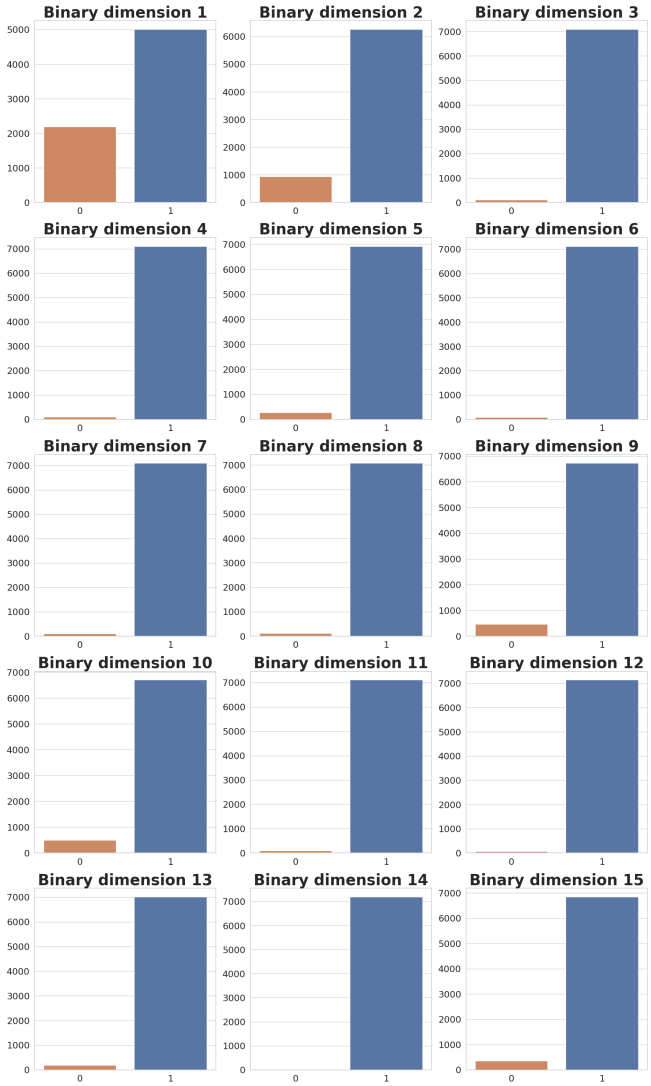


Figure 3: Distribution of binary values for a generated dataset.

Figures 3 and 4 show the distribution of values for this generated dataset.

2. Anomaly detection approaches and results

For the original dataset, we don't know the exact number of anomalies, so the threshold was chosen manually. We used the 99% threshold, so for each approach 72 anomalies were detected. For the generated dataset with 5% of outliers we tried to detect all anomalies (360 objects).

2.1. K prototypes approach

2.1.1. Model's description. The first approach is the K-Prototypes model, specifically designed to handle datasets containing both continuous and categorical (binary in our case) attributes. This model combines the strengths of the

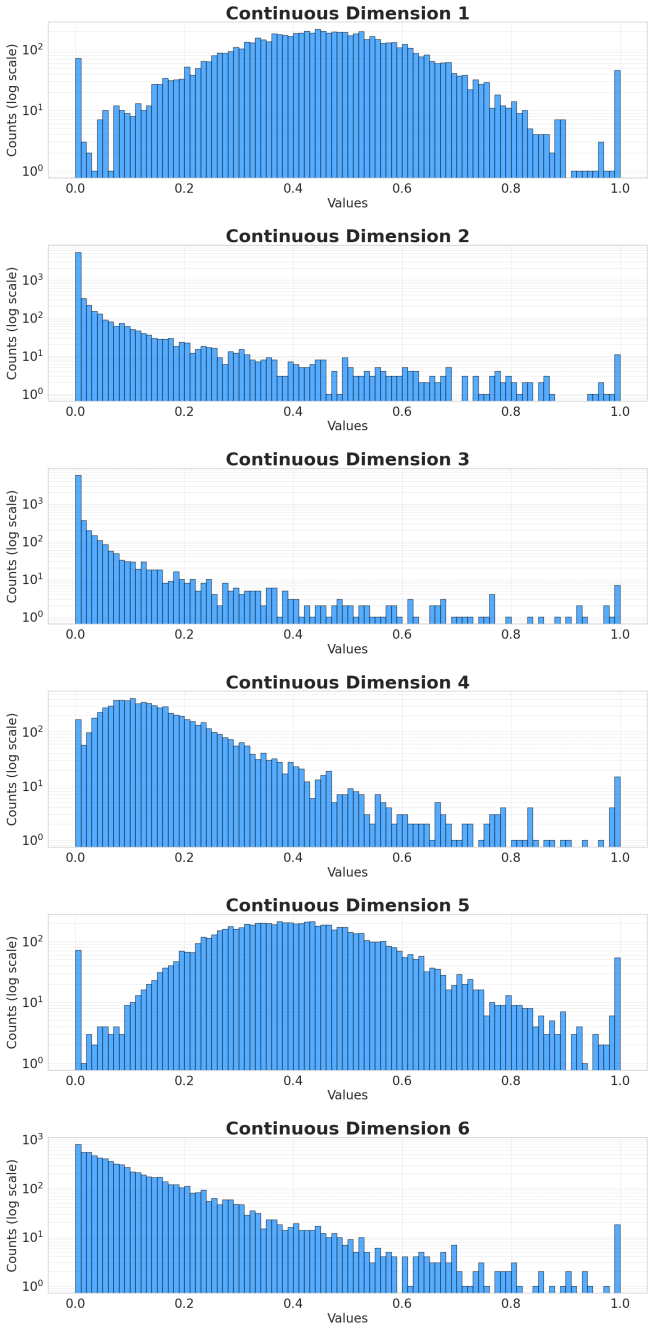


Figure 4: Distribution of continuous values for a generated dataset.

K-means and K-modes clustering algorithms, making it effective for mixed-type data.

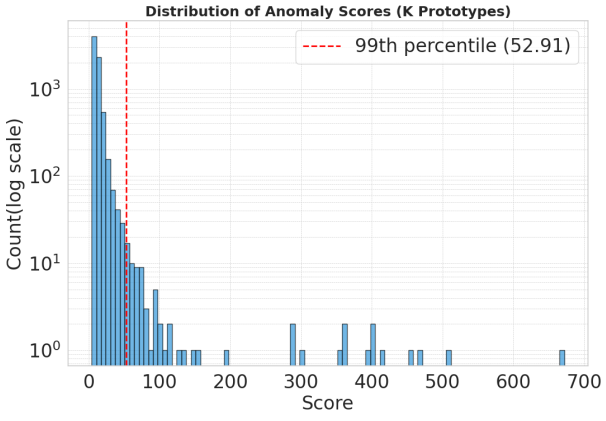
The K-Prototypes model works by minimizing a combined distance measure using Euclidean distance for continuous attributes and Matching distance for binary. The distances are calculated as the weighted average between the two. The weights are calculated by the model itself.

The model preprocesses the data by normalizing continuous attributes. The K-Prototypes algorithm is then applied to form the specified number of clusters. After clustering, the distance of each data point from its assigned cluster centroid is calculated. Points that significantly deviate from their cluster centroid are the anomalies detected by the model.

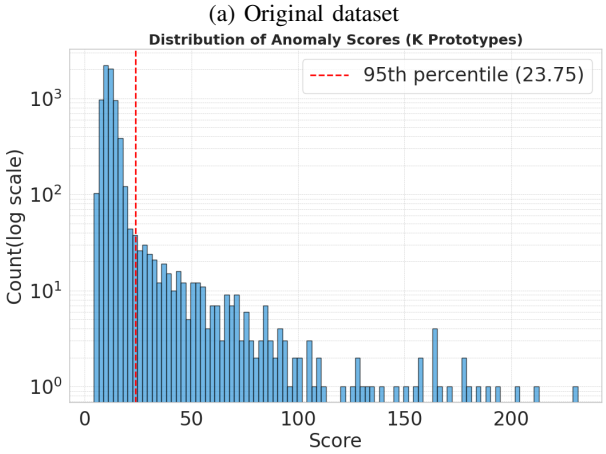
2.1.2. Results and discussion. Interestingly, all cluster centroids have identical binary values - only 1s, which are over-represented values (see 1). This means that the main influence on anomaly detection in this approach is from the continuous values.

Figure 5 shows the distribution of anomaly scores (in the case of K Prototypes approach - distances to cluster centroids) for the original and generated datasets. The y-axis is in log scale. the red dotted line represents the threshold after which data points are considered anomalies.

One way to evaluate the anomaly detection models in the case of unsupervised learning task is to look at the distribution of anomaly scores. Typically the majority of anomaly scores should be relatively low. There may be one or more secondary peaks in the distribution as one moves to the right, but these secondary peaks should only contain a relatively small fraction of the points [1].



(a) Original dataset



(b) Generated dataset

Figure 5: Distribution of K Prototypes anomaly scores (distances to cluster centroids) for original and generated datasets.

2.2. Information based approach

2.2.1. Model's description. The second approach is Information Gain Anomaly Detection, which identifies anomalies based on the concept of information gain. This approach is based on calculating the entropy of the dataset and identifying anomalies as data points that introduce huge uncertainty or disorder influencing the entropy more than normal data points. The entropy is computed separately for continuous and binary attributes and then combined.

To detect anomalies, the model calculates the entropy of the dataset after removing each individual data point. By observing the change in entropy when each point is removed, the model identifies points that cause high entropy changes, indicating they are anomalies. In this case, the difference in entropy is the anomaly score.

2.2.2. Results and discussion. Figure 6 shows the distribution of anomaly scores (in the case of the Information-based approach - entropy change for each object) for the original and generated datasets. Unlike the K Prototypes distributions, the information-based approach performs seemingly well for the generated dataset but poorly for the original. The scores are too densely packed around low values.

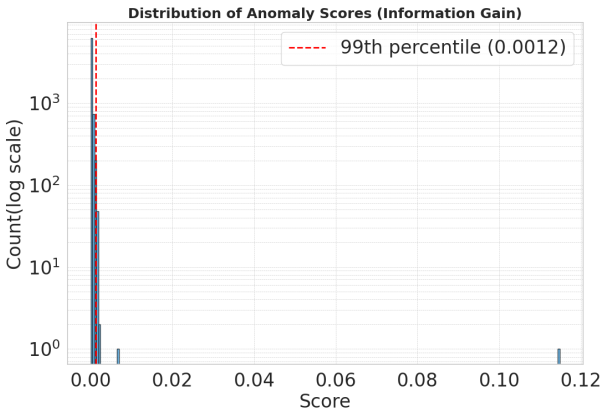
The main drawback of this approach is that entropy is not well-defined for continuous variables. By separating continuous values into bins, we approximate the differential entropy, which extends Shannon entropy to continuous variables, replacing the sum with an integral. However, it is important to note that discrete and differential entropies are not equivalent and do not necessarily operate on the same scale [2]. This distinction must be considered when applying entropy-based methods to mixed data types.

2.3. Hybrid approach combining clustering for continuous data with entropy for binary data

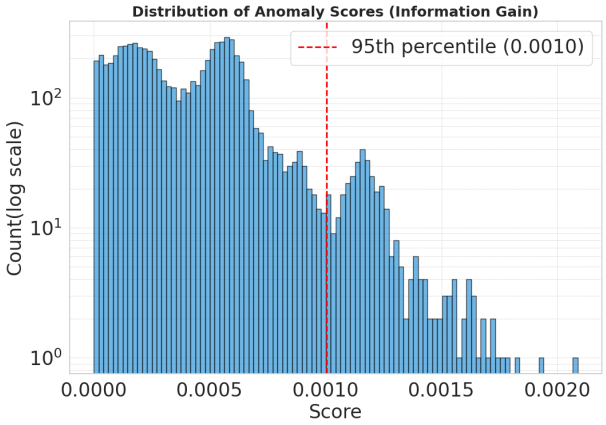
2.3.1. Model's description. To overcome the disadvantages of previous approaches, the hybrid one was implemented. As we discussed earlier, clustering in the K Prototypes was more effective for continuous values and the information gain approach is more natural for binary data. For these reasons, the hybrid method uses clustering for continuous data and information gain for binary data to identify anomalies effectively.

Continuous attributes are normalized. The KMeans algorithm is applied to the continuous attributes. The distance of each data point from its assigned cluster centroid is then calculated. These distances serve as the anomaly scores for the continuous attributes, with larger distances indicating potential anomalies.

For the binary attributes, the model computes the information gain using an entropy calculation. The idea is the same as in the previous approach. The continuous and binary anomaly scores are combined using a weighted average. Before combining, both sets of scores are normalized using Min-Max normalization to ensure they are on a comparable scale.



(a) Original dataset



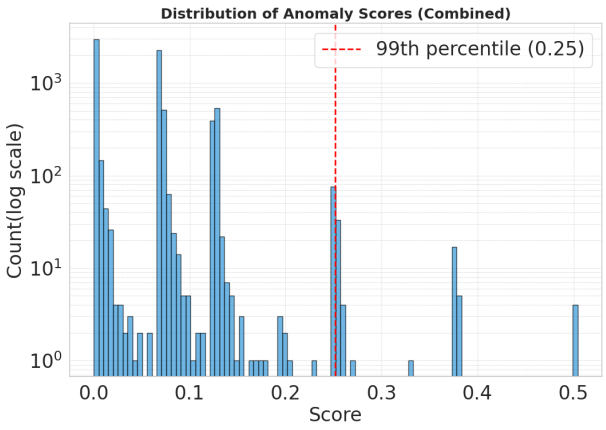
(b) Generated dataset

Figure 6: Distribution of Information based approach anomaly scores (entropy change) for original and generated datasets.

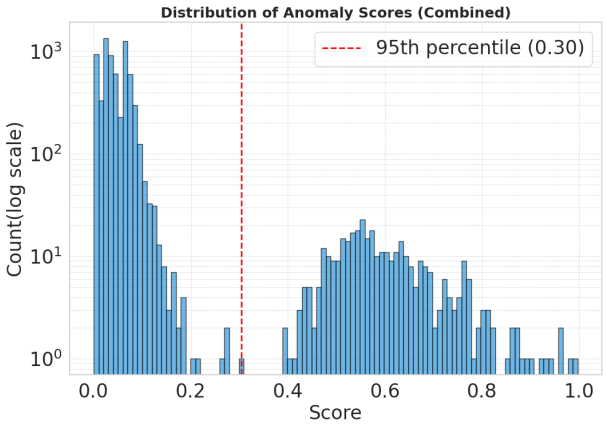
2.3.2. Results and discussion. Figure 7 shows the distribution of anomaly scores (in the case of the Hybrid approach - weighted average of the entropy gain on binary data and distances to cluster centroids on continuous) for the original and generated datasets. As we can see, there are secondary peaks after the threshold with a significant number of objects.

2.4. Autoencoder approach

2.4.1. Model's description. The core idea of using autoencoders for anomaly detection is based on their architecture, which consists of two main components: the encoder and the decoder. The encoder reduces the dimensionality of the data while preserving its essential features. The decoder then reconstructs the original data from this compressed representation. The network is trained to minimize the difference between the input data and the reconstructed output. Once trained, the autoencoder can be used to detect anomalies by evaluating the reconstruction error based on the assumption that normal data points will have a lower reconstruction error than anomalies.

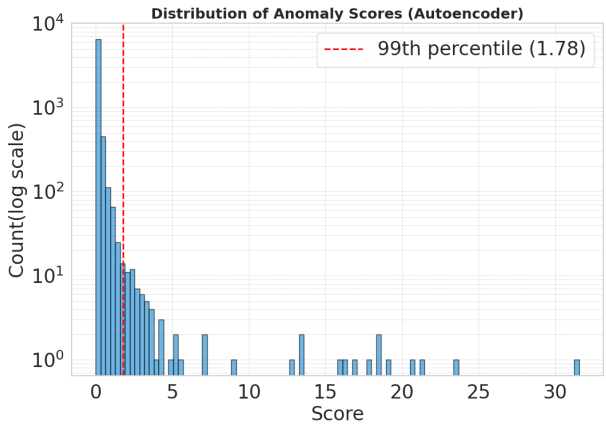


(a) Original dataset

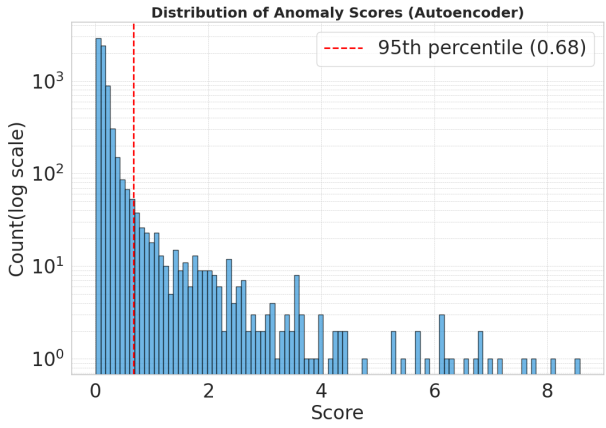


(b) Generated dataset

Figure 7: Distribution of the Hybrid approach anomaly scores (entropy change and distances) for original and generated datasets.



(a) Original dataset



(b) Generated dataset

Figure 8: Distribution of the Autoencoder approach anomaly scores (reconstruction error) for original and generated datasets.

2.4.2. Architecture. The encoder compresses the input data into a lower-dimensional representation. It consists of three linear layers, ReLU activation functions, and dropout layers to prevent overfitting.

The decoder is a neural network with three linear layers, ReLU activations, and dropout layers. A sigmoid activation function is applied at the end to put values between 0 and 1.

2.4.3. Results and discussion. Figure 8 shows the distribution of anomaly scores (in the case of the Autoencoder approach - reconstruction error) for the original and generated datasets. The distribution for the original dataset seems even more reliable than the distribution for the generated one.

Typically, Autoencoders are trained on 'normal' data and only after new data points are shown to the model. In our case, however, we don't have labels, so we train and run the model on the same data.

Overall, K Prototypes and Autoencoder are the most promising based on their distributions

3. Choosing the best model

For the generated dataset, we know which data points are outliers. All models were evaluated and results can be seen below.

- **Correctly detected outliers by K prototypes approach:** 295 / 360
- **Correctly detected outliers by Information based approach:** 140 / 360
- **Correctly detected outliers by Hybrid of approach:** 360 / 360
- **Correctly detected outliers by Autoencoder:** 261 / 360

Surprisingly, all outliers were detected using the Hybrid model. This may be a consequence of the way we built the outliers in the generated dataset. Looking at the scores distribution (see 7) all the outliers are distinctly separated from the normal data with a noticeable gap in scores between these two groups.

Another interesting metric can be the number of anomalies that are predicted by different models. First, let's see this in the case of the generated dataset:

Combination	Common Indices (max 360)
All Methods	107
KPrototypes and InfoBased	121
KPrototypes and Hybrid	295
KPrototypes and Autoencoder	319
InfoBased and Hybrid	140
InfoBased and Autoencoder	118
Hybrid and Autoencoder	261
KPrototypes, InfoBased, Hybrid	116
KPrototypes, InfoBased, Autoencoder	112
KPrototypes, Hybrid, Autoencoder	256
InfoBased, Hybrid, Autoencoder	111

TABLE 1: Common Indices Among Different Approaches for the Generated dataset

Overall, all approaches except the Information-Based work well on the generated dataset.

Now let's look at the results for the original dataset. All models defined 72 outliers (99% threshold). Below, we can see the similarity of the predictions.

Combination	Common Indices (max 72)
All Methods	0
KPrototypes and InfoBased	3
KPrototypes and Hybrid	2
KPrototypes and Autoencoder	61
InfoBased and Hybrid	29
InfoBased and Autoencoder	4
Hybrid and Autoencoder	3
KPrototypes, InfoBased, Hybrid	0
KPrototypes, InfoBased, Autoencoder	3
KPrototypes, Hybrid, Autoencoder	2
InfoBased, Hybrid, Autoencoder	1

TABLE 2: Common Indices Among Different Approaches for the Original dataset

K Prototypes and Autoencoder have the highest number of common detected anomalies. As mentioned before, they also were the most promising based on the distribution of anomaly scores. Since these two models are significantly different, we may assume that the common predictions are most likely the true anomalies. The Hybrid and InfoBased approaches performed very poorly.

We need to select one model for the task. We chose the Autoencoder. Based on reconstruction error, we estimated probabilities of each object to be an anomaly. The error was normalized. Then, if it is above the threshold, we set the probability to 1; otherwise, we scale it.

4. Further improvements

There are several ideas for improvement of each model.

For K Prototypes, the number of clusters can significantly impact the performance. Additionally, scaling the continuous values can make a big difference. When we tested the K Prototypes without scaling, leaving the values in the range from 0 to 1, it performed worse. Trying different scaling methods may be helpful.

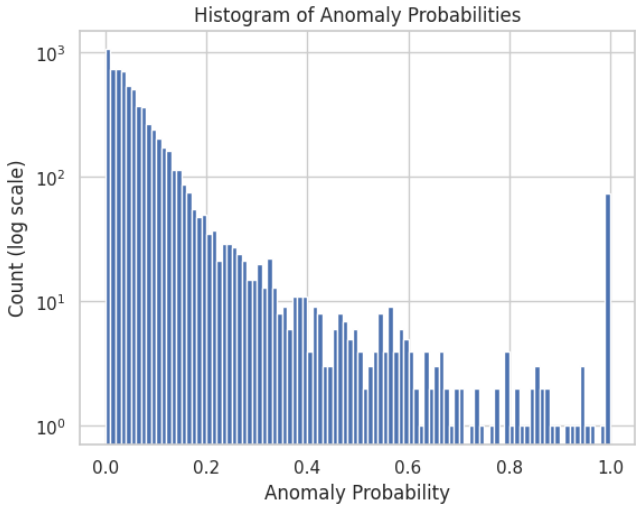


Figure 9: Distribution of anomaly probabilities (log scale).

For InfoBased, we could explore more advanced binning techniques or smooth approximations to better estimate the differential entropy. Another idea is to use kernel density estimation to approximate the continuous distributions more accurately. Additionally, we can experiment with entropy normalization for more equal influence from discrete and differential entropy.

For the hybrid approach, calculating the weights of binary and continuous contributions rather than defining them manually (e.g., 0.5 for both) could improve the performance. We can implement ideas similar to the ones used in K Prototypes, where during clustering the 'gamma' constant is calculated to define the weights of categorical data.

For the Autoencoder, further tuning of hyperparameters may be helpful. Additionally, switching to a variational autoencoder (VAE) could provide better results by adding probabilistic elements into the model, which can improve its ability to detect anomalies. Exploring different activation functions, learning rates, and batch sizes could also enhance the model's performance.

5. Conclusion

This project explored anomaly detection within a dataset containing mixed data types, specifically binary and continuous variables. We implemented and evaluated four approaches: KPrototypes, an information gain approach with entropy calculation, a hybrid approach combining clustering for continuous data with entropy for binary data, and an Autoencoder. The distribution of anomaly scores was analyzed for all approaches, and an artificial dataset with labeled outliers was also used for evaluation purposes.

Our results show that each method has unique strengths and limitations. The KPrototypes approach and Autoen-

coder demonstrated the most promising results based on the distribution of anomaly scores and their ability to detect anomalies in both the original and generated datasets. The hybrid approach successfully identified all artificial outliers, possibly due to the construction of the outliers in the generated dataset.

While KPrototypes and Autoencoder approaches show strong potential, there is room for improvement in each method. Future work should focus on optimizing hyperparameters, scaling techniques, and integrating more advanced methods to achieve better results.

Statement

The authors declare that this report is entirely original and does not contain any plagiarism. The research explained in this report was conducted by the authors themselves, and all the sources have been cited in the Reference Section. None of the content was generated using automated language models.

References

[1] F. Stella, "Anomaly Detection: Additional Algorithms," Lecture, University of Milano-Bicocca, 2024.

[2] G. Carcassi, C. A. Aidala, and J. Barbour, "Variability as a better characterization of Shannon entropy," *European Journal of Physics*, vol. 42, no. 4, p. 045102, May 2021. [Online]. Available: https://dx.doi.org/10.1088/1361-6404/abe361