Optimisation of Training Samples with KLE and Mutual Information

Denys Symonov[1], Oleksandr Palagin[1], Yehor Symonov[1] and Bohdan Zaika[1]

[1] *V.M. Glushkov Institute of Cybernetics of the National Academy of Sciences (NAS) of Ukraine, Akademika Glushkova Avenue 40, Kyiv, 03187, Ukraine*

Abstract

One of the key challenges in modern machine learning is reducing the dimensionality of the feature space in training samples while preserving essential information for classification and forecasting tasks. This study proposes a methodologically grounded approach that integrates the Kozachenko-Leonenko entropy (KLE) method with mutual information to enhance feature selection, thereby improving model accuracy and reducing computational complexity. A comparative analysis on the real-world dataset confirms the effectiveness of the proposed method in selecting informative features and improving classification performance.

Keywords

Kozachenko-Leonenko entropy (KLE) method, mutual information, machine learning, feature selection, dimensionality reduction, training sample

1. Introduction

The quality of machine learning models largely depends on the quality of training samples that are formed at the data preparation stage and directly affect the accuracy, generalisability and stability of the models. Effective data preparation involves data cleaning, transformation, selection of relevant features, and elimination of outliers. However, these processes are complex and require automation.

The concept of entropy is one of the most powerful mathematical tools that allows for such operations to be performed objectively and formally. Entropy methods allow to estimate the degree of uncertainty in training data, identify the most relevant features, and find optimal strategies for their processing.

The use of entropy for data processing is not a novel concept, but its relevance in modern machine learning tasks is only growing. One of the most well-known areas of its application is the discretisation of continuous features. For example, the Fayad-Irani method is based on entropy minimisation to determine the optimal partitioning thresholds, which allows obtaining compact and informative value intervals [1]. This technique is effective for improving trained models, which is confirmed by empirical studies [2, 3]. Such approaches allow not only to reduce the dimensionality of the feature space but also to improve the generalisation ability of the models.

Another important aspect is the selection of features based on entropy criteria. Methods such as information gain [4, 5] and Gini impurity [6-8] identify the extent to which each feature contributes to class recognition. This enables the elimination of redundant or insignificant features, thereby increasing the efficiency of classification algorithms. In particular, algorithms such as SelectKBest [9, 10] and Recursive Feature Elimination (RFE) [11, 12] are effectively used to select relevant features even in cases of large and unbalanced samples. Taking into account methods based on conditional and mutual information allows creating more flexible and adaptive feature selection strategies for further use in machine learning algorithms.

In addition to working with features, entropy is used to select the most informative training samples. In the context of active learning, one of the most common approaches is Entropy Sampling, where priority is given to samples for which the model has the highest uncertainty in predictions. This allows to significantly reduce the size of the training set without degrading the classification quality.

[1]CMIS-2025: The Eighth International Workshop on Computer Modeling and Intelligent Systems, May 5, 2025, Zaporizhzhia, Ukraine

denys.symonov@gmail.com (D. Symonov); palagin_a@ukr.net (O. Palagin); e.symonov@gmail.com (Y. Symonov); zaikabohdan5@gmail.com (B. Zaika)

0000-0002-6648-4736 (D. Symonov); 0000-0003-3223-1391 (O. Palagin); 0009-0008-2581-2001 (Y. Symonov); 0009-0001-9567-8361 (B. Zaika)

Studies in computer vision and text analytics confirm the effectiveness of this strategy [13, 14]. It should also be noted that entropy analysis helps to assess the balance of classes in the training set. A low value of the entropy of the class distribution signals a significant imbalance, which can negatively affect the performance and accuracy of the model. Diagnostic criteria based on entropy help to identify such problems in time and apply appropriate corrective strategies, such as sample rebalancing or weighting.

Therefore, the use of entropy methods in the processing and analysis of training samples opens up wide opportunities to improve the efficiency of machine learning models. These approaches not only optimise the structure of the sample but also improve its information content, which directly affects the accuracy and stability of the built models.

2. Problem statement

One of the key challenges of modern machine learning is to reduce the dimensionality of the feature space without significant loss of information required to solve forecasting or classification tasks. The growing amount of data used in models leads to an increase in computational complexity, model overtraining, and a decrease in their generalisation ability. As a result, it is important to develop effective methods for selecting informative features and reducing the dimensionality of the space while retaining relevant information. Traditional approaches, such as principal component analysis (PCA) [15] or linear discriminant analysis (LDA) [16], are effective only under certain assumptions about the data distribution. However, in the case of complex, non-linear relationships between features, these methods may not be effective. An alternative is entropy-based methods for assessing the information content of features, which do not require any prior assumptions about the data distribution.

Therefore, the problem statement is to develop a methodologically sound approach based on entropy methods, which will improve the quality of training samples and, consequently, the accuracy of machine learning models.

3. Goal and objectives of the study

The purpose of this study is to analyse and justify the effectiveness of using entropy method for processing and analysing training samples in order to improve the quality of machine learning models. To achieve this goal, the following objectives are considered.

1. Analyse the capabilities of the Kozachenko-Leonenko entropy (KLE) method for assessing the informativeness of features in N-dimensional space.
2. Development of a combined approach using KLE and mutual information for feature selection to improve the quality of models and reduce the need for resources to solve classification or prediction problems.

The accomplishment of these objectives will contribute to the development of a methodological framework for machine learning and provide an effective approach to the preparation of training samples, which in turn will increase the accuracy and generalisation of models.

4. A combined approach using KLE and mutual information for feature selection

This section presents a novel approach to feature selection that combines KLE entropy and mutual information. By integrating these two methods, the proposed approach aims to improve the evaluation of feature relationships, decrease dimensionality of training sample and, as a result, enhance both classification accuracy and generalization performance. The first part of the section discusses the use of the KLE method in N-dimensional space for preparing the training sample, while the second part demonstrates how KLE and mutual information are combined to perform feature selection.

4.1. KLE method in N-dimensional space

The KLE method is an effective approach for estimating differential entropy in N-dimensional space. This nonparametric method, unlike its parametric counterparts, does not require any prior assumptions about the data distribution and works well even with complex, nonlinear distributions [17].

Let's assume that the task of training sample preparation involves the following.

1. Remove or reduce the influence of noisy data (outliers);
2. Select or transform a subset of features in such a way as to ensure the highest informativeness with respect to the output variable;
3. Ensure satisfactory accuracy of the machine learning model, even under conditions of incomplete information.

Denote the given dataset by $X=\{(x_i, y_i)\}, i=\overline{1,N}$, where $x_i \in R^N$ is a feature vector and y_i is the target variable ($y_i \in R$ for regression, $y_i \in \{C_1, C_2, \ldots, C_k\}$ for classification). The KLE method estimates the differential entropy of the feature space X, which is useful for analysing the informativeness of features and their relationship with the target variable Y.

The differential entropy of a random variable $X \in R^N$ is defined as

$$H(X)=-\int_{R^N} f_X(x)\log f_X(x)dx, \quad (1)$$

where $f_X(x)$ is the probability density of the feature distribution.

The KLE algorithm for N-dimensional space is as follows. First, for each point x_i the distance to its k-th nearest neighbour is found (for example, with the Euclidean distance):

$$\rho_k(x_i)=\min_k\left\{\rho \vee \left|\{x_j \in X : \|x_j-x_i\| \le \rho\}\right| \ge k+1\right\}, \quad (2)$$

where $\rho_k(x_i)$is the radius containing k+1 points, including the point x_i itself; x_j is the point for which the k-th nearest neighbour is searched; $\|x_j-x_i\|$ is the distance between points x_i and x_j according to the selected distance metric; ρ is the value of the radius, which changes until the minimum value is found that satisfies the condition; k is the number of nearest neighbours that are taken into account.

The number of k nearest neighbours, which is taken into account when estimating the distribution of the noise component, can be calculated with the pseudocode from the Figure 1.

$k \leftarrow 1, k_{opt} \leftarrow 1, J_{min} \leftarrow J(1), Q = [J(1)]$
for $k = 2, \ldots, N$:
    calculate $J(k)$;
    $Q \leftarrow Q \cup \{J(k)\}$;
    if $|Q| > L$ **then** Q.pop_front();
    if $J(k) < J_{min}$ **then** $k_{opt} \leftarrow k, J_{min} \leftarrow J(k)$;
    if $\forall r \in Q: |r - J_{min}| < \varepsilon$ **then** break;
    if $J(Q_{first}) - J(Q_{last}) > -\varepsilon$ **then** break;
return k_{opt}.

Figure 1: Pseudocode for calculating the number of k nearest neighbours

The pseudocode in Figure 1 implements an adaptive method for finding the optimal value of parameter k, using a window of size L to track the dynamics of the evaluation function $J(k)$ changes and introducing a stopping condition based on the stability or deterioration of the function value.

The evaluation function $J(k)$ can be calculated with the equation

$$J(k)=\frac{1}{B}\sum_{b=1}^{B}\left[H_{KLE,k}(X_b)-\overline{H}_k\right]^2, \quad (3)$$

where $H_{KLE,k}(X_b)$ is the KLE entropy with fixed k (e.g., $k = 1$) on the sample X_b; $\overline{H}_k$ is the average KLE entropy with fixed k on the sample X_b; B is the number of samples (e.g., bootstrap samples or cross-validation samples); X_b is one of the B samples.

Next, the volume of a unit sphere in N-dimensional space is determined by the chosen norm. For example, for the Euclidean norm, the volume of a unit sphere

$$V_N = \frac{\pi^{\frac{N}{2}}}{\Gamma\left(\frac{N}{2}+1\right)}, \tag{4}$$

where $\Gamma(\cdot)$ is the gamma function.

The last step is to calculate the KLE-entropy estimate

$$H_{KLE} = \Psi(M) - \Psi(k) + \log(V_N) + \frac{N}{M}\sum_{i=1}^{M} \ln \rho_k(x_i) + \gamma, \tag{5}$$

where $\Psi(\cdot)$ is the digamma function, $\Psi(x+1) = \Psi(x) + \frac{1}{x}$; M is the number of sample points; $\gamma \approx 0.5772$ is the Euler-Mascheroni constant.

4.2. Combining KLE with mutual information

The mutual information is a powerful tool for detecting non-linear dependencies between features and the target variable, facilitating the construction of a dataset that is both balanced and information-rich [18]. Mutual Information measures the extent to which information about $X_b = \{(x_i, y_i)\}, i = \overline{1, N}$ helps to determine Y. If the values of X_b are conditionally independent, then mutual information will be

$$I(X_b, Y) = \sum_{j=1}^{N} I(x_{ij}, y_i), \tag{6}$$

provided that

$$p(x_{i1}, x_{i2}, \ldots, x_{i} \vee y_i) = \prod_{j=1}^{n} p(x_{ij} \vee y_i). \tag{7}$$

Accordingly, Mutual Information for x_i is defined as:

$$I(x_i \vee y_i) = \sum_{i=1}^{N} \sum_{y_i \in Y} p(x_i, y_i) \log \frac{p(x_i, y_i)}{p(x_i)\, p(y_i)}, \tag{8}$$

where $p(x_i, y_i)$ is the joint probability of events x_i and y_i; $p(x_i)$, $p(y_i)$ are the corresponding marginal probabilities of x_i and y_i.

Based on the integral Mutual Information scores given on the sample from the dataset X_b, it is possible to eliminate insignificant features, thus the sample for model training will be of the form

$$X_T = \{X_b \vee I(x_i \vee y_i) \geq \alpha\}, \tag{9}$$

where $\alpha -$ is the cut-off threshold.

This approach makes it possible to reduce the dimensionality of the feature space while maintaining a high level of relevance to the target variable.

If the model performance at the validation stage is insufficient, it is possible to add an algorithm for iteratively adjusting the α threshold or implementing the combined criterion $\phi(X_T)$, which aims to maximise the ratio by changing the α parameter and the structure of X_T:

$$\phi(X_T) = Score_{CV}(X_T) - \lambda |X_T|, \tag{10}$$

Where $Score_{CV}(X_T)$ is the average model quality score based on the cross-validation; $|X_T|$ is the number of selected features; $\lambda \geq 0$ is the penalty factor.

To sum up, the integration of KLE entropy and Mutual Information methods allows to significantly reduce the dimensionality of the initial feature set, while maintaining sufficient information potential for efficient model training. This increases not only the performance of machine learning algorithms, but also their stability and interpretability in real classification and regression tasks.

5. Comparative analysis of modelling results

One of the important stages of developing effective machine learning models is validation of the results on real data set, which allows to objectively assess the impact of preprocessing on the accuracy and stability of classification. To verify the quality of the proposed entropy methods, a comparative analysis of classification results using pre-processed and unprocessed training samples was applied.

5.1. Dataset

To evaluate the effectiveness of entropy-based methods for processing training samples to improve the quality of machine learning models, the Gas Sensor Array Low-Concentration dataset [19] is used. Table 1 shows a snapshot of the Gas Sensor Array Low-Concentration dataset. The full dataset contains 90 gas samples collected by 10 semiconductor sensors. The studied gases include ethanol, acetone, toluene, ethyl acetate, isopropanol and n-hexane at three concentrations: 50 ppb, 100 ppb and 200 ppb. For each gas and concentration combination, five samples were collected to provide a variety of data for modelling. Each sample consists of 9000 data points representing the sensors' response to the gas. Each sensor generates 900 data points, allowing for detailed analysis of their response to different gases and concentrations. The data was collected in three stages: baseline (5 minutes), gas injection (10 minutes), and purification (15 minutes) with a sampling rate of 1 Hz.

Table 1
Data: "Gas Sensor Array Low-Concentration"

Col1	Col2	Col3	Col4	Col5	Col6	Col7	...	Col9001	Col9002
ethanol	100ppb	0.3565	0.3345	0.3575	0.333	0.3565	...	3.9325	3.9315
ethanol	100ppb	0.3525	0.3305	0.3525	0.332	0.355	...	3.918	3.9205
ethanol	100ppb	0.355	0.3345	0.354	0.332	0.355	...	3.8825	3.8815
...	...	...	...	...	...	...	...	...	...
hexane	50ppb	0.3955	0.3805	0.394	0.3745	0.394	...	3.8995	3.8995

The presence of data for several types of gases, concentrations, and time phases (see Table 1) makes it possible to form a representative training set for building classification models in real-world conditions. Such a sample is optimal for testing the effectiveness of feature space reduction methods, in particular those based on entropy and mutual information.

5.2. Description of the experiments

The base machine learning model is the ensemble method Random Forest, where multiple independent decision trees are combined to enhance accuracy and stability. The implementation of machine learning algorithms and data analysis is conducted in Python, utilising libraries such as NumPy, pandas, matplotlib, scikit-learn, time, and psutil. These libraries support tasks including classification, dataset splitting (train_test_split), learning curve analysis, model training with RandomForestClassifier, performance evaluation metrics, t-SNE, PCA, and resource and execution time monitoring.

The dataset is initially divided into training and test sets, followed by model training based on a predefined target vector. To assess sensitivity to missing features, a mechanism is employed that retains only a fixed number of significant features, replacing the remaining ones with mean values computed from the training set. Classification quality is evaluated using accuracy metrics, ROC AUC, MAE, and MSE, while the model's performance dependency on training set size is examined through a learning curve analysis.

For feature space analysis and dimensionality reduction, t-SNE (a non-linear projection) and PCA (a linear projection onto principal components) are applied. In addition to classification performance, computational efficiency and resource consumption are assessed by measuring execution time and CPU load.

All computational experiments presented in the paper were conducted on a laptop equipped with an Intel Core i7-13620H processor (13th generation, 10 cores: 6 performance and 4 efficiency cores, base frequency 2.40 GHz) and 16 GB of RAM. The system operates on a 64-bit Windows operating

system with x64 architecture. Parallel computations were automatically handled through CPU multithreading using libraries such as scikit-learn, NumPy, and joblib, which support task parallelization via the n_jobs parameter. GPU acceleration was not employed, as the main computations involved tabular data processing and ensemble modeling (Random Forest), which are efficiently executed on modern CPUs.

5.3. Results of the experiments

The results of model testing presented in this section demonstrate a comparative analysis of the effectiveness of classification methods under conditions of incomplete input data and different approaches to feature preprocessing. Particular attention is paid to the quality of classification, stability of models, their ability to generalise, and computational efficiency.

Figure 2 a) (left) shows the effect of the available features on the classification accuracy in the absence of training set preprocessing (¬KLE). There is a gradual increase in classification accuracy with the number of available features, but this increase is non-linear and has some fluctuations. The initial accuracy values are low, and the maximum value does not reach one, which indicates a significant loss of information. These results indicate that even with an increase in available features, the classifier cannot achieve 'perfect' accuracy due to the influence of noisy or unrepresentative data. In Figure 2 b) (right), where the training set was pre-processed using the Kozachenko-Leonenko entropy (KLE) method, a much faster increase in classification accuracy is observed. With a small number of features, the accuracy values are almost the same as in the first graph, but after reaching a certain threshold (approximately at 7 features), the accuracy increases sharply and approaches one. This indicates a significant improvement in classification quality due to data preprocessing, which likely eliminated the influence of noisy or irrelevant features, making the model more robust to incomplete data.

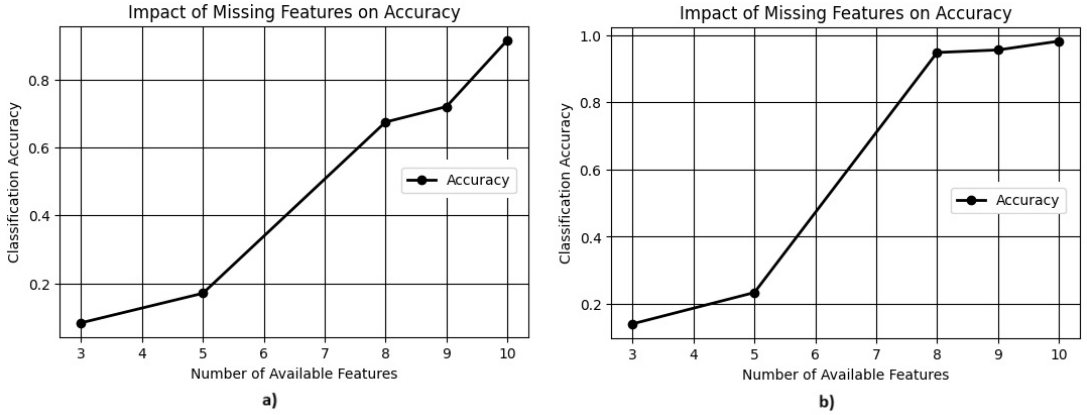

Figure 2: Incomplete data and classification accuracy

Figure 3 a) (left) shows that for the ¬KLE model, the training accuracy remains relatively stable as the training sample size increases, while the cross-validation accuracy gradually increases but remains below the training accuracy. This may indicate a certain level of overfitting, as the model demonstrates higher accuracy on training data than on cross-validation data. The difference between the two curves indicates the presence of noise and uneven distribution of information in the training sample. In Figure 3 b) (right), the KLE model shows a much better balance between training and cross-validation accuracy. Already with relatively small amounts of data, the model achieves high accuracy, and the difference between the two curves is much smaller, indicating better model generalisation and reduced overfitting. This confirms the effectiveness of pre-processing, which reduces the influence of irrelevant or noisy features and improves the quality of training.

In Figure 4 a) (left), there is significant chaos and high density of points for the t-SNE test of the ¬KLE model, indicating a weak structure in the data. The classes overlap significantly, which can make classification difficult. Such a distribution indicates the presence of noise and irrelevant information in the features, which can reduce the accuracy of the model and its ability to generalise patterns in the data. In Figure 4 b) (right), a more structured distribution of points is observed for the KLE model. Clusters are clearer, indicating improved differentiation between classes. This confirms the effectiveness of pre-processing in reducing noise and identifying hidden patterns in the data.

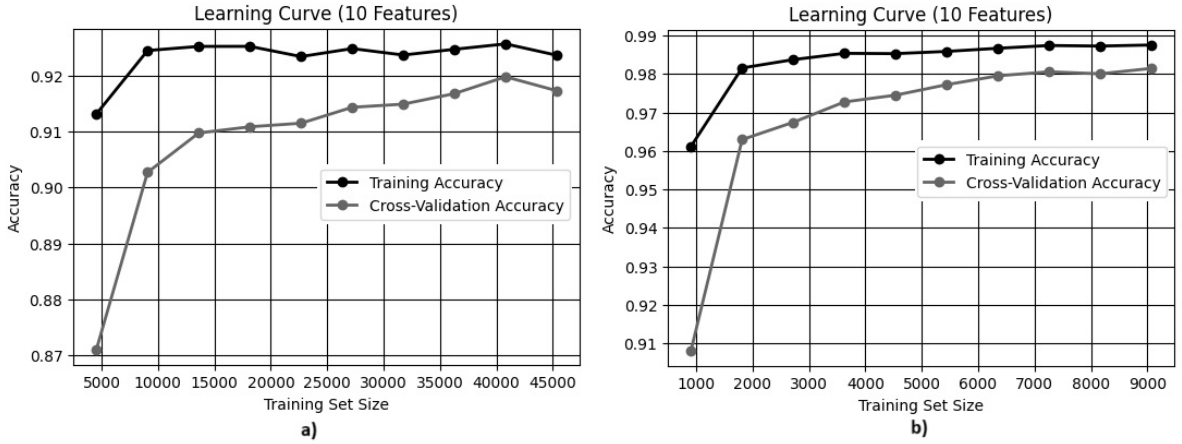

Figure 3: Comparison of learning curves of "¬KLE" and "KLE" methods

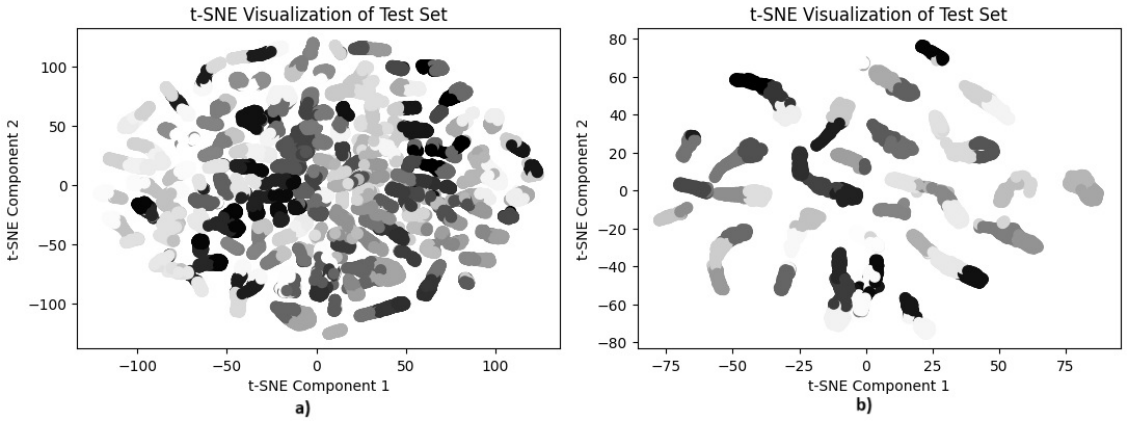

Figure 4: t-SNE analysis of the test set with different feature processing

Figure 5 a) (left) for the ¬KLE model shows that the data are unevenly distributed and have some clusters, but the structure remains blurred. The classes overlap to a large extent, which can make classification difficult, as there is no clear boundary between the groups. Such a distribution indicates that the original features contain a significant amount of noise or irrelevant information, which reduces the quality of model training. In Figure 5 b) (right), the KLE model shows a clearer separation between the groups, the data looks more clustered and has distinct directions in the principal component space. This indicates effective noise removal and improved differentiation between classes, which can improve classification accuracy.

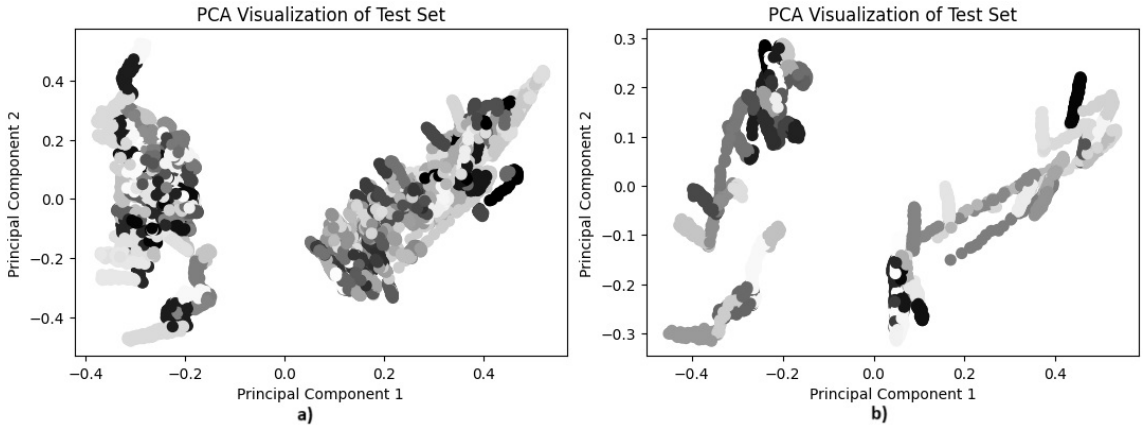

Figure 5: PCA analysis of the test set with different data processing

Figure 6 a) (top row) shows that without preprocessing, the prediction time and the CPU usage increases from 0.45 sec (3 features) to 0.67 sec (10 features), indicating the high computational complexity of the model. In Figure 6 b) (bottom row), after processing with the Kozachenko-Leonenko

entropy method, the prediction time increases only from 0.077 sec to 0.096 sec, while CPU Time stabilises at 0.094 sec after 6 features. This confirms the effectiveness of the processing in reducing computational costs and improving performance.

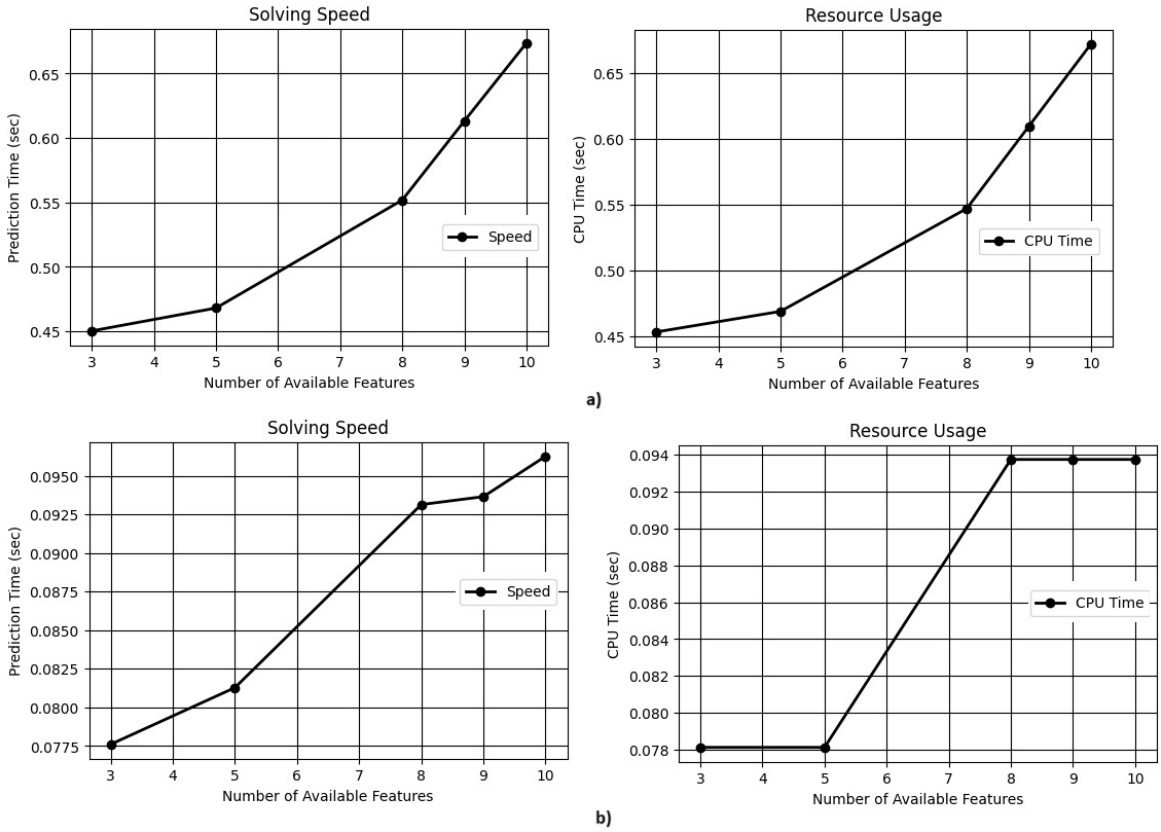

Figure 6: Dependence of performance and resources on training set pre-processing

Table 2 shows that without pre-processing (¬KLE), all metrics deteriorate sharply as the number of features decreases: AUC-ROC drops from 0.999 (10/10 features) to 0.747 (3/10 features), MAE and MSE increase significantly, and Log Loss increases from 0.525 to 5.427, indicating a loss of model stability. This indicates that without preprocessing, the model becomes very sensitive to a decrease in the number of features, which impairs its ability to generalise patterns. On the other hand, with KLE, the classification accuracy remains consistently high even with incomplete information. For example, the AUC-ROC changes less sharply (from 1.000 to 0.805), and the MSE and Log Loss remain at lower levels than in the case of ¬KLE. This shows that entropy processing improves model generalisability and reduces the impact of missing features, making the algorithm more robust to incomplete data.

Table 2
Quality metrics of models

Number of input parameters	AUC-ROC		MAE		MSE		Log Loss		Macro-F1	
	¬KLE*	KLE	¬KLE	KLE	¬KLE	KLE	¬KLE	KLE	¬KLE	KLE
10/10	0.999	1.000	0.486	0.029	14.35	0.05	0.525	0.097	0.91	0.98
9/10	0.997	0.999	1.589	0.282	40.16	8.46	1.038	0.407	0.69	0.95
8/10	0.996	0.999	2.180	0.637	53.80	23.88	1.421	0.776	0.64	0.94
5/10	0.847	0.899	11.118	7.641	244.37	157.74	4.827	7.058	0.11	0.14
3/10	0.747	0.805	12.877	13.915	302.93	427.91	5.427	12.562	0.05	0.06

** - Machine learning algorithm without using KLE for preparing the initial sample*

The results in Table 2 demonstrate that data preprocessing using KLE not only improves classification accuracy, but also ensures the stability of the metrics while reducing the amount of input information. This approach is effective both in terms of model quality and computational performance.

6. Conclusion

The primary goal of this study was to develop and substantiate an effective method for optimization of training sample, based on entropy theory, combining the Kozachenko-Leonenko entropy (KLE) and mutual information. The declared objectives included analysing the potential of KLE in N-dimensional feature space and constructing a hybrid approach for feature selection to enhance model quality and reduce computational cost. The findings fully reflect the achievement of these objectives.

The proposed method offers a non-parametric evaluation of differential entropy, capable of detecting noise and selecting informative features without relying on prior distributional assumptions. The integration with mutual information enables identification of features most relevant to the target variable, contributing to the creation of a compact yet expressive feature space.

Empirical validation on the Gas Sensor Array Low-Concentration dataset confirmed the practical effectiveness of the method with the following results.

1. The AUC-ROC metric under preprocessing with KLE remained high even with partial data (1.000 with the full feature set; 0.805 with only 3 out of 10 features), whereas in the unprocessed baseline it dropped to 0.747.
2. The Mean Squared Error (MSE) remained low (ranging from 0.05 to 427.91 depending on the number of features) for the proposed method, indicating improved noise resilience.
3. The Macro-F1 score remained consistently higher (ranging from 0.98 to 0.06 for KLE vs. 0.91 to 0.05 for the baseline) under feature removal scenarios.
4. Prediction time decreased from 0.67 seconds (baseline) to 0.096 seconds (with KLE) for 10 features, demonstrating enhanced computational efficiency.
5. Visualisation techniques such as t-SNE and PCA further confirmed improved class separability and reduced noise.

The analysis shows that the use of KLE entropy allows for an objective assessment of the informativeness of features, reducing their number without losing relevance, which significantly increases the accuracy and stability of models. The use of mutual information in combination with KLE facilitates the selection of the most significant features, which minimises the influence of noise factors and allows optimising the feature space for training. The results also show a significant reduction in model overfitting and computational costs by removing redundant information.

Therefore, this research presents a theoretically grounded and empirically validated approach to entropy-based preprocessing. The alignment between the initially defined objectives and the achieved results has been demonstrated through both qualitative and quantitative analysis. This work provides a foundation for the further integration of entropy-driven techniques into advanced machine learning pipelines, particularly in domains characterised by complex or imbalanced datasets.

7. Directions for further research

The use of entropy criteria in combination with deep learning methods can significantly improve the quality of training samples, especially in high-dimensional spaces. In particular, a promising area is the adaptation of the KLE method to analyse the relationship between features in deep neural networks, which will not only reduce the feature space but also determine their informativeness in the context of multilevel data representations.

Special attention should be paid to the integration of entropy-based approaches with active learning methods, which will allow for dynamic sample adjustment in the process of model training. The use of strategies similar to Entropy Sampling will allow optimising the balance of classes and selecting the most informative examples for training. Further development of such approaches may include the creation of adaptive algorithms that combine estimates of differential entropy and mutual information to optimise the learning process in real time. This will not only reduce computational costs, but also improve the generalisation capability of the models, ensuring their stability even in circumstances of high variability in input data.

Acknowledgements

The work was supported by the state budget research project “Develop methods for modelling the processes of targeted management of complex multi-component information systems for various

purposes" (state registration number 0123U100754) of the V.M. Glushkov Institute of Cybernetics of the National Academy of Sciences (NAS) of Ukraine.

Declaration on Generative AI

During the preparation of this work, the authors used Grammarly in order to: Grammar and spelling check. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] Q. Chen, M. Huang, H. Wang, A feature discretization method for classification of high-resolution remote sensing images in coastal areas, IEEE Transactions on Geoscience and Remote Sensing 59 (2021) 8584–8598. doi: 10.1109/TGRS.2020.3016526.

[2] S. Ma, J. Zhai, Big data decision tree for continuous-valued attributes based on unbalanced cut points, Journal of Big Data 10 (2023) 135. doi: 10.1186/s40537-023-00816-2.

[3] A. Suppa, F. Asci, G. Saggio, P. Di Leo, Z. Zarezadeh, G. Ferrazzano, G. Costantini, Voice analysis with machine learning: one step closer to an objective diagnosis of essential tremor, Movement Disorders 36 (2021) 1401–1410. doi: 10.1002/mds.28508.

[4] K. Son, S. W. Lee, W. Yoon, K. H. Hyun, CreativeSearch: Proactive design exploration system with Bayesian information gain and information entropy, Automation in Construction (2022). doi: 10.1016/j.autcon.2022.104502.

[5] G. S. Reddy, S. Chittineni, Entropy based C4.5-SHO algorithm with information gain optimization in data mining, PeerJ Computer Science 7 (2021). doi: 10.7717/peerj-cs.424.

[6] Z. Wang, G. Deng, H. Xu, Group feature screening based on Gini impurity for ultrahigh-dimensional multi-classification, AIMS Mathematics (2023). doi: 10.3934/math.2023216.

[7] P. K. Mali, H. Motiyani, Q. Sameed, A. Mehta, Hyper spectral image clustering and local feature selection using Gini impurity, in: Proceedings of the 7th International Conference on Trends in Electronics and Informatics (ICOEI), 2023, pp. 1629–1634. doi: 10.1109/ICOEI56765.2023.10125605.

[8] R. A. Disha, S. Waheed, Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based Weighted Random Forest (GIWRF) feature selection technique, Cybersecurity 5 (2022). doi: 10.1186/s42400-021-00103-8.

[9] M. Maftoun, J. H. Joloudari, O. Zare, M. Khademi, A. Atashi, M. A. Nematollahi, R. Alizadehsani, J. M. Górriz, Improving prediction of mortality in ICU via fusion of SelectKBest with SMOTE method and Extra Tree classifier, in: J. M. Ferrández Vicente, M. Val Calvo, H. Adeli (Eds.), Artificial Intelligence for Neuroscience and Emotional Systems, vol. 14674, Lecture Notes in Computer Science, Springer, Cham, 2024. doi:10.1007/978-3-031-61140-7_7.

[10] M. Jamei, M. Ali, H. Afzaal, M. Karbasi, A. Malik, A. A. Farooque, Z. Haydar, Q. U. Zaman, Accurate monitoring of micronutrients in tilled potato soils of eastern Canada: Application of an explainable inspired-adaptive boosting framework coupled with SelectKBest, Comput. Electron. Agric. 216 (2024) 108479. doi: 10.1016/j.compag.2023.108479.

[11] S. Kollem, C. Sirigiri, and S. Peddakrishna, A novel hybrid deep CNN model for breast cancer classification using Lipschitz-based image augmentation and recursive feature elimination, Biomed. Signal Process. Control. 95 (2024) 106406, doi: 10.1016/j.bspc.2024.106406.

[12] M. Awad and S. Fraihat, Recursive Feature Elimination with Cross-Validation with Decision Tree: Feature Selection Method for Machine Learning-Based Intrusion Detection Systems, J. Sens. Actuator Netw. 12(5) (2023) 67, doi:10.3390/jsan12050067.

[13] S.-l. Li, G. Zheng, H. Wang, T. Yao, Y. Chen, S. Ding, and X. Li, Entropy-driven Sampling and Training Scheme for Conditional Diffusion Generation, in European Conference on Computer Vision, 2022, doi:10.48550/arXiv.2206.11474.

[14] W. Zhang, H. Guo, A. Le, J. Yang, J. Liu, Z. Li, T. Zheng, S. Xu, R. Zang, L. Zheng, and B. Zhang, Lemur: Log Parsing with Entropy Sampling and Chain-of-Thought Merging, ArXiv, 2024, doi:10.48550/arXiv.2402.18205.

[15] S. Ali, S. Verma, M. B. Agarwal, R. Islam, M. Mehrotra, R. K. Deolia, J. Kumar, S. Singh, A. A. Mohammadi, D. Raj, M. K. Gupta, P. Dang, and M. Fattahi, "Groundwater quality assessment

using water quality index and principal component analysis in the Achnera block, Agra district, Uttar Pradesh, Northern India", Scientific Reports 14 (2024), doi:10.1038/s41598-024-56056-8.
[16] S. Zhao, B. Zhang, J. Yang, J. Zhou, and Y. Xu, Linear discriminant analysis, Nature Reviews Methods Primers 4(1) (2024) 70, doi:10.1038/s43586-024-00346-y.
[17] W. Xu, S. Zhu, Q. Li, X. Chen, and X. Zhou, Star uniform selection algorithm based on maximizing Kozachenko-Leonnko entropy, in Proceedings of the Fourth International Conference on Artificial Intelligence and Electromechanical Automation (AIEA 2023), F. Wen, C. Zhao, and Y. Chen, Eds., vol. 12709, International Society for Optics and Photonics (SPIE), 2023, p. 1270936, doi:10.1117/12.2684884.
[18] H. Gong, Y. Li, J. Zhang, B. Zhang, and X. Wang, A new filter feature selection algorithm for classification task by ensembling Pearson correlation coefficient and mutual information, Engineering Applications of Artificial Intelligence 131 (2024) 107865, doi: 10.1016/j.engappai.2024.107865.
[19] F. Tian, L. Zhao, S. Deng, Gas sensor array low-concentration [Dataset], UCI Machine Learning Repository, 2023. doi:10.24432/C5CK6F.