On Development Classification Methods for Hidden Features Separation in Data

Valentina Petrovych[a], Vladislav Kuznetsov[a], Eduard Manziuk[b], Iurii Krak[a,c], Veda Kasianiuk[c], Olexander Barmak[b] and Anatoliy Kulias[a]

[a] *Glushkov Cybernetics Institute, Kyiv, 40, Glushkov ave., 03187, Ukraine*
[b] *National University of Khmelnytsky, 11 Institutes str., 29016, Ukraine*
[c] *Taras Shevchenko National University of Kyiv, Kyiv, 64/13, Volodymyrska str., 01601, Ukraine*

Abstract

This work discusses the comparison of hidden parameters in data based upon scientific datasets on three different topics: scientific text data, medical data and sound recordings. A set of problems was underscored, such as big data classification. The work proposes an approach to solve these problems; this can be made as follows: the properties of each given dataset are observed in feature space of reduced dimensionality and making use of decision boundary scaling from smaller dimension to the feature space of an original dimension. The representation obtained by using this approach gave a possibility to apply visual analysis of data and to obtain an efficient architecture of classifier using hidden properties of the data.

Keywords [1]

Texts analysis, data, hidden parameters, classification, feature extraction.

1. Introduction

Being scaled up by the feature space and dataset size, the data becomes more challenging for data scientists in order to process and give satisfying results; in particular, it becomes even more challenging when one has to scale and modify existing methods for data analysis, data classification and so all. These challenges occurred during a set of scientific works dedicated to data classification, in particular in [1,2]. This means, in fact, that every dataset needs a specific approach – to create a specific architecture of classifier in order to obtain the most plausible parameters and results as well.

Thus, in order to fulfill these requirements, the datasets become more detailed as well as the optimizing procedures – in order to get the best decision plane for classification. This approach needs much more resources than are available to data scientist, which tends to increase the computation costs in order to process the data in such way.

Taking in account the problems in analysis of such type of data, the research tasks are following:

- to create a few datasets which represent different research topics;
- to propose a feature representation for each data entry in each dataset;
- to study feature representation using dimensionality reduction, grouping of features and clustering;
- to obtain the locations of the biggest density in feature space for each dataset;
- to conduct tests on different algorithms for data classification on each dataset;
- to study the presence of hidden features' clusters in feature space;
- to assess the classification algorithms' stability using perturbations in data.

II International Scientific Symposium «Intelligent Solutions» IntSol-2021, September 28–30, 2021, Kyiv-Uzhhorod, Ukraine

EMAIL: filonval63@gmail.com (V. Petrovych); kuznetsowwlad@gmail.com (V. Kuznetsov); eduard.em.km@gmail.com (E.Manziuk); krak@univ.kiev.ua (I. Krak); veda.kasianiuk@gmail.com (V. Kasianiuk); barmakov@khnu.km.ua (O.Barmak); anatoly016@gmail.com (A.Kulias)
ORCID: 0000-0002-5982-8983(V. Petrovych); 0000-0002-1068-769X(V. Kuznetsov); 0000-0002-7310-2126 (E.Manziuk); 0000-0002-8043-0785 (I. Krak); 0000-0003-3268-303X (V.Kasianiuk); 0000-0003-0739-9678 (O.Barmak); 0000-0003-3715-1454 (A.Kulias)

2. Getting data and proposed methods

Let’s discuss the data dimensionality reduction task on three datasets dedicated to scientific texts analysis, miohram analysis (muscular contractions) and audible sound recordings’ analysis; each dataset has the following specifications:

- scientific texts – 100 000 data entries and 100 000 features: (TF-IDF) [3];
- miohram – 6 thousand data entries and 60 features (time segments);
- sound recordings had 2 thousand samples and 256 features (spectrogram density of FFT representation).

2.1. Linear systems’ synthesis for data dimensionality reduction and structured data recognition

In order to create a linear system [4, 5] one has to fulfil the requirements to data, such as linear independence of each feature one to another. It can be proven that the dataset answer the requirement above in case if it has a specific number of features and this quantity is equal to rank of the data matrix, and it’s greater or equal to the number of data entries. Thus, making a different number of features it can be said that there exists a specific set of features that maximizes the rank taking in account the hidden connection in data. As soon as this requirement is fulfilled, the dataset can be normalized by each feature span, for instance, making use of first momentum of the feature matrix. The next step in order to obtain linearly independent features is to calculate a covariance matrix; it gives a possibility to sort out the redundant data, which is in most cases is caused by its own hidden structure.

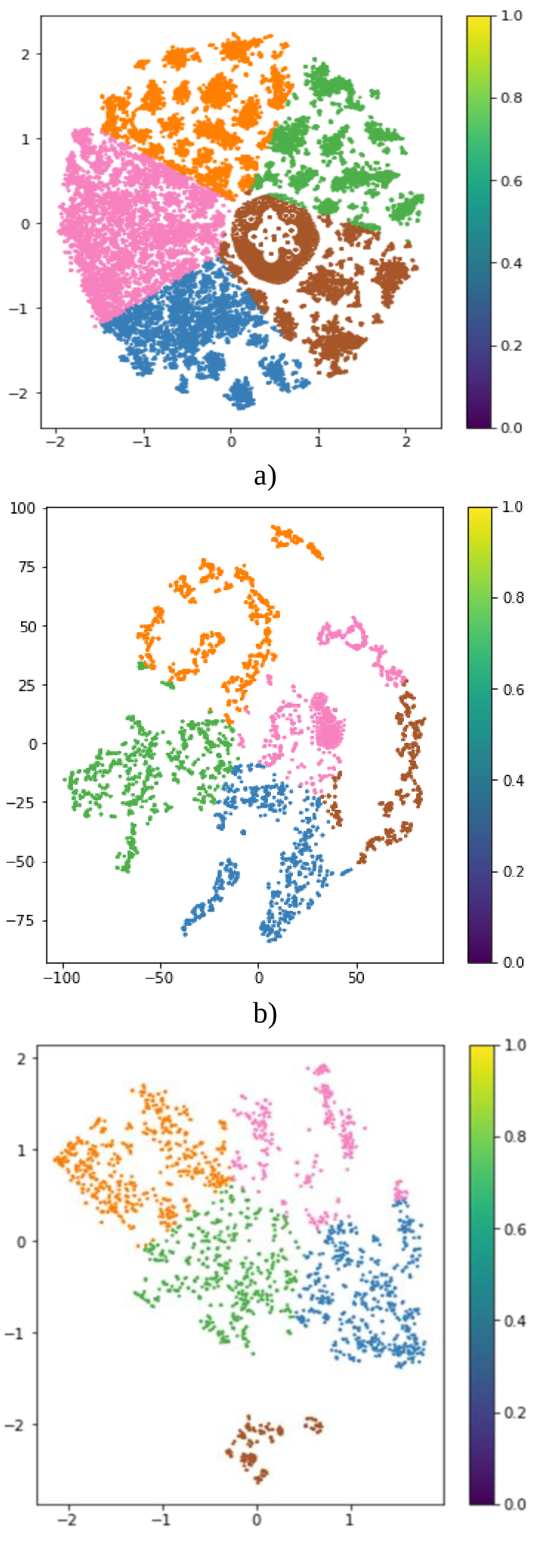
Since this process can be repeated many times using specific procedure, it can be described as an iterative procedure of sorting of connections in data by their significance taking in account the hidden internal structure of the data. In order to compress abundant features, we propose to apply the eigenvalues decomposition. The biggest advantage of this approach is a significant feature dimensionality reduction and, as a result, – reduction of computational time during each run of optimizing procedure in specific classification algorithms discussed below.

2.2. Clustering and grouping of features

In order to apply visual analysis we propose to apply methods for grouping of features, that gives us, in particular, an ability to visualize the data in 2 dimensional representation. It was made by T-stochastic grouping of nearest neighbors [6, 7] which allowed to reduce dimensions of the data to exactly 2 dimensions. This allowed representing feature vectors for each dataset studied by applying grouping of features and applying mean clustering methods [10] in order to aid the visual analysis and use these hypotheses in further experiments, discussed below.

2.3. Feature-space representation and hidden features of the data

According to our study on different datasets and types of the data, the data behave differently in different cases, depending on scattering of the data and distribution of points in feature-space. For instance, if the data has a great variance and representation of different types of classes in data, we observe the data representation using grouping of features show us areas of greater density, which can be treated as a hidden features, that, respectively, can be treated as classes of the data. In order to prove this statement we conducted a series of experiments on different datasets, in particular, in this work – on medical, audio/wave data as well as text data, which are presented on Fig.1.



a)

b)

c)

Figure 1: Data representation in feature space of lower dimensionality: scientific texts - a) medical data - b), audio/wave data - c)

As implies from the Fig. 1, each type of the data has its own properties, which are, based upon our considerations are connected with a number of features in original feature space and, in the same time, with a specific number of data entries in each dataset. Thus, the visual grouping of features and forming of data clusters in each dataset by its increasing both.

Since the data has hidden classes, they can be studied using different methods. For instance, the data points, that have big distance between each other can create a hypotheses in feature space of lower dimension and translate it to feature space of original or intermediate dimension

2.4. Data-driven classification of hidden features using mean clustering

Hidden features, the location of data points in feature space can be used either to propose an initial hypothesis to train other algorithms or apply so-called data-driven classification. This implies in following: the initial clusters are used as data point's labels and as a result, data now can be classified not by the a priori data labels, but as a posterior analysis, that was conducted using clustering of data points by some clustering method, for instance mean clustering [11, 12, 13, 14].

Using such labels as labels for classification algorithms, for now it is possible to compare their efficiency and most important, whether the features as well as studied algorithms are stable (what will be discussed later in this paper). According to series of experiment, various methods were tested:

- bayesian networks,
- feedworward neural networks;
- support vector machine classifiers with linear and radial basis decision function,
- decision trees, in particular: adaptive and extreme gradient boosting classifiers.

In most cases the achieved accuracy was in average 85%, depending on the type of the dataset. In order to test the repeatability of these results as well as to figure out worst case scenario on particular data, we decided to test the algorithm stability using perturbations in data.

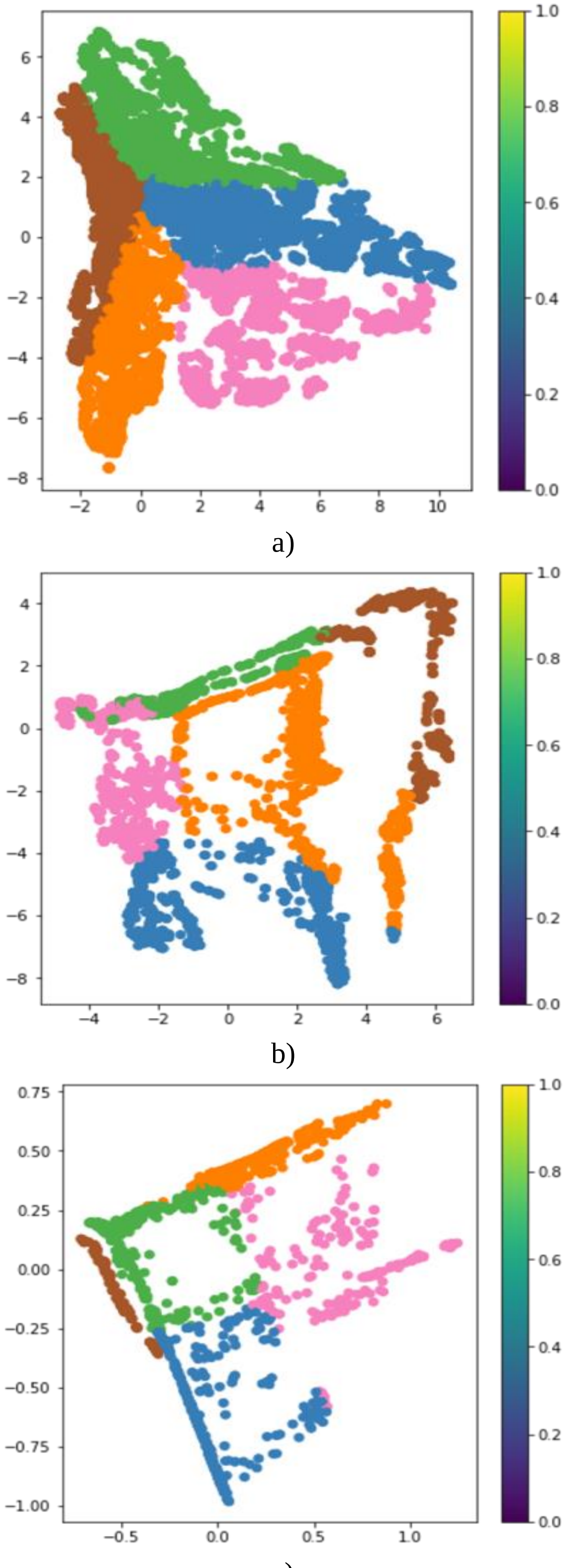
2.5. Testing algorithm stability to feature matrix perturbations

In order to compare different algorithms, we conducted an additional experiment. This experiment was carried out as follows: for each dataset has been applied a specific type of perturbations, that translated the feature representation from one feature-space to another, keeping the relative orientation of centroids of data in place; it was made by applying a variation autoencoders [8, 9] encoding procedure to data of reduced dimension. All this was possible due to autoencoders properties – the datasets are keeping integrity, at the same time decreasing the mean square error; thus, it was used to perform such task.

Let's discuss in details about the architecture used in this experiment. In most cases, the autoencoder is used to compress data and because of that it has a bottleneck structure, where hidden layers have smaller dimensions than the dimensions of an input data. In contrary, we propose to encode the input dimension using the opposite – hidden layer of bigger dimension and the output dimension of the same dimension as the original one. Using this approach we can compare side-by-side the representation in both cases and also to compare the decision boundary, using means of visual analysis. In order to do so, we applied an encoding transformation for each dataset and made a visualization of encoding dimension (Fig. 2).

According to Fig. 2, we can acknowledge, that the encoding dimension decreases variance and bias in data and, since, it becomes denser, as well as the mean square error becomes smaller, than in original image. Because the autoencoder creates non-affine transformations in order to create new representation of data, it is very important to figure out how it affects the algorithm stability, because the visual analysis shows that the data is separable. Based upon series of tests on three datasets (Fig. 2) as well as preliminary study on datasets of smaller sizes, the classification algorithms shown us different behavior and since – different stability to perturbations.

In most cases the error rate was in average 15 percent for five class classification (92% for the audio data; 68% for the text data and 70% for the medical data); the best results were obtained on decision tree methods – extreme gradient boost and adaptive boosting. We suppose, that this was caused by very narrow decision boundary, and thus it decreased the algorithm stability and overall needed to perform more iteration in order to achieve same results.



a)

b)

c)

Figure 2: Feature representation in encoding dimension of an autoencoder: scientific texts - a), medical data - b), audio/wave data - c)

The disturbance of the input data will affect the convergence of the classifier learning algorithm. The perturbation in the original matrix and the transformation matrix in the feature latent space is the entropy value. Entropy can be determined indirectly by the energy reduction of the characteristic mean square error vector of the sample matrix. Variational autoencoders minimize the mean quadratic error (MSE) between data elements. MSE is also an expression of data magnitude. Therefore, the value of the variance change can be related to the nature of the energy decrease of its source and the self-number of the transformation matrix. Therefore, knowing the magnitude of variance change, we can estimate the number of vectors responsible for the information part of the data being studied. The advantage of latent representation is that it reduces the distance between data classes (and separation bands), which ultimately affects the number of iterations of the optimization algorithm and the convergence of the classification algorithm. In order to study performance on this data we decided to run an additional experiment for multiclass classification

2.6. Testing algorithm convergence on different number of classes

Most important feature of a dataset and classification algorithm used on it is ability to achieve certain amount of determination of results. It can be made in different ways – using different chunks of the data, splitting the dataset to a test and a validation dataset, as well as running multiple iterations of splitting of the dataset and running the optimizing procedure. In contrary to this approach, it can be proven that in different situations the datasets can be scaled up to achieve a certain level of a confidence as well as accuracy and number of errors. Just because the dataset has the upper limit since it is prepared – labeled, preprocessed and formed in feature matrix, sometimes it is important to assess the desired size of the dataset, using prior knowledge about dataset performance as well as it's features – number of features and data points.

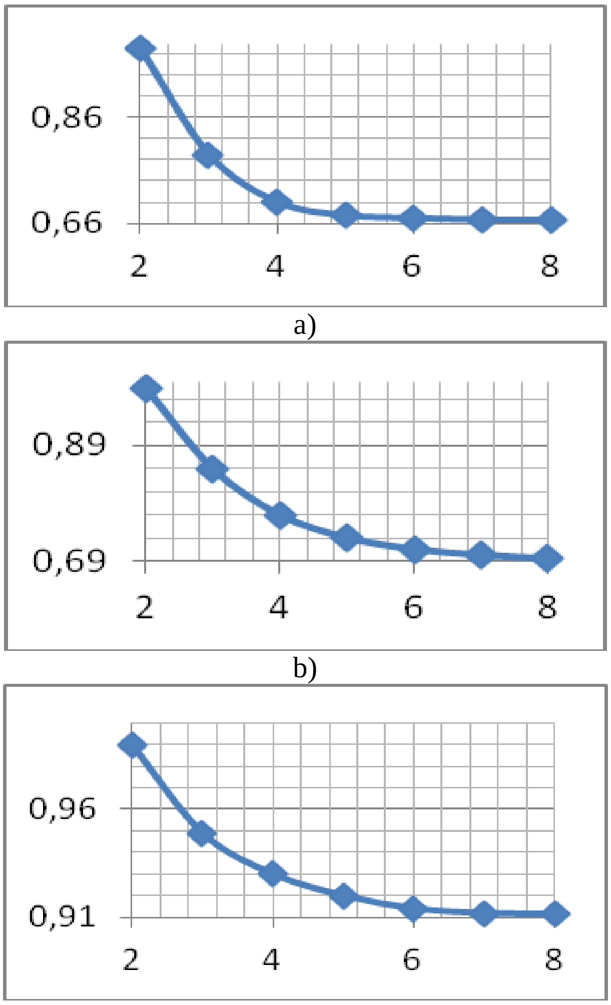
According to this approach we conducted a series of experiments that were targeting to best and worse-case scenario varying the number of classes and testing algorithm performance. We studied the overall performance in between 2 and 8 class separation using boosting algorithms. According the test runs, we built an approximation curve that gives an assumption about behavior of a particular dataset in general. In Fig. 3 we represented the theoretical curves based upon recognition rate on a test dataset. Using a visual analysis approach on these accuracy curves one can say that each dataset has an upper and a lower limit of accuracy depending on a number of classes and number of data points (which, in reality, is lower that our initial assumption). Since, for each given dataset it is possible to calculate an approximate value of regression coefficients (Fig.3), we can assume the size of the dataset that uses same features and gives similar performance as the dataset used as a standard one. For instance, the regression hypothesis of an audio recordings' dataset (Fig. 3 c)) can be scaled to hypothesis about medical data (Fig. 3 b)) which can give an estimate about magnitude and the need of data to get a desired accuracy.

2.7. Engineering of the dataset using data augmentation

As we said above, according to our hypotheses on test data, it is possible to generate new instances of the data in order to increase accuracy and test. In order to do so, we must define an area of data points feature space, where are the data points can be present and to generate new points having same distribution with slight changes that are focused on balancing of classes. It can be done via different techniques which are described in papers [15, 16, 17, 18, 19]. In our case, we decided to test these methods in some test data with imbalanced classes and big classification error on one specific class (Table 1). At first glance, we ran an AdaBoost classifier on data of reduced dimension with changes and, then we ran a classifier on an augmented data, using data over-sampling and generating necessary data points. According to our test runs, we got an increase of overall accuracy – either in-sample or out-of sample (Table 2).

2.8. Detection of anomalies and distortions in the data

An important point in conducting experimental tests for classification and clustering of data is to assess the feasibility of the necessary procedures on the resulting data set. During data acquisition, distortions of individual data elements and, in some cases, perturbations in the group of data points may occur in such a way that the overall representation of the data in space of lower dimension will be also distorted by such data elements. That is why before testing the classification algorithms, the characteristics of the data themselves were also evaluated, namely the relative location of the hidden features of the data elements in the space of reduced dimension (n=2) and the possibility of simultaneous visual analysis [20,21].



a)

b)

c)

Figure 3: Approximation of accuracy curves for a test data for datasets: text data - a); medical data - b); audio recordings c). Bottom axis is for number of classes, left one is for accuracy value

Table 1
Classification rate on test dataset

precision	recall	f1-score	support	
Class 1	0.59	1.00	0.74	366
Class 2	0.98	0.95	0.96	454
Class 3	0.90	0.64	0.75	332
Class 4	1.00	0.59	0.74	333
accuracy			0.81	1485
macro avg	0.87	0.79	0.80	1485
weighted avg	0.87	0.81	0.81	1485

This can be done in several ways – by calculating standard deviation and mean for each of the data classes and determining the appropriate intervals for variance of the elements of each class or using special algorithms and procedures that perform it automatically. As an example, the search for data elements in the feature space of dimension 2 using linear regression and random sample consensus of for three given data sets. [22,23]. In Fig. 4 we show the regressions for each subset of studied data in the representation of T-SNE features, using following resressions: linear regression and random sample consensus (RANSAC) as follows.

Table 2
Classification rate on augmented dataset

precision	recall	f1-score	support	
Class 1	0.81	0.97	0.88	1313
Class 2	0.92	0.96	0.94	1239
Class 3	0.95	0.76	0.85	1274
Class 4	0.98	0.93	0.96	1254
accuracy			0.91	5080
macro avg	0.92	0.91	0.91	5080
weighted avg	0.91	0.91	0.91	5080

Performing the construction of each regression, we can determine one or more major axes in the data (assuming that the available elements in the hidden feature space perform a shift of the centroids relative to the origin of data graph). Data elements for a single class or an entire data set (as shown in fig. 2) that appear to have a specific distance from the principal axis (or axes) that is more than, for example, 3 standard deviations for a given data set, may indicate possible classification problems .

That is why it is important either to carry out the procedure of retraining data with the rejection of distorted elements (if possible and plausible in such conditions), or (if this is not possible) to correct data by introducing appropriate distortions that compensate existing distortions in data. In following paragraph of the article, we discuss the compensation of distortions in detail.

2.9. Classification of text samples using binary classification methods

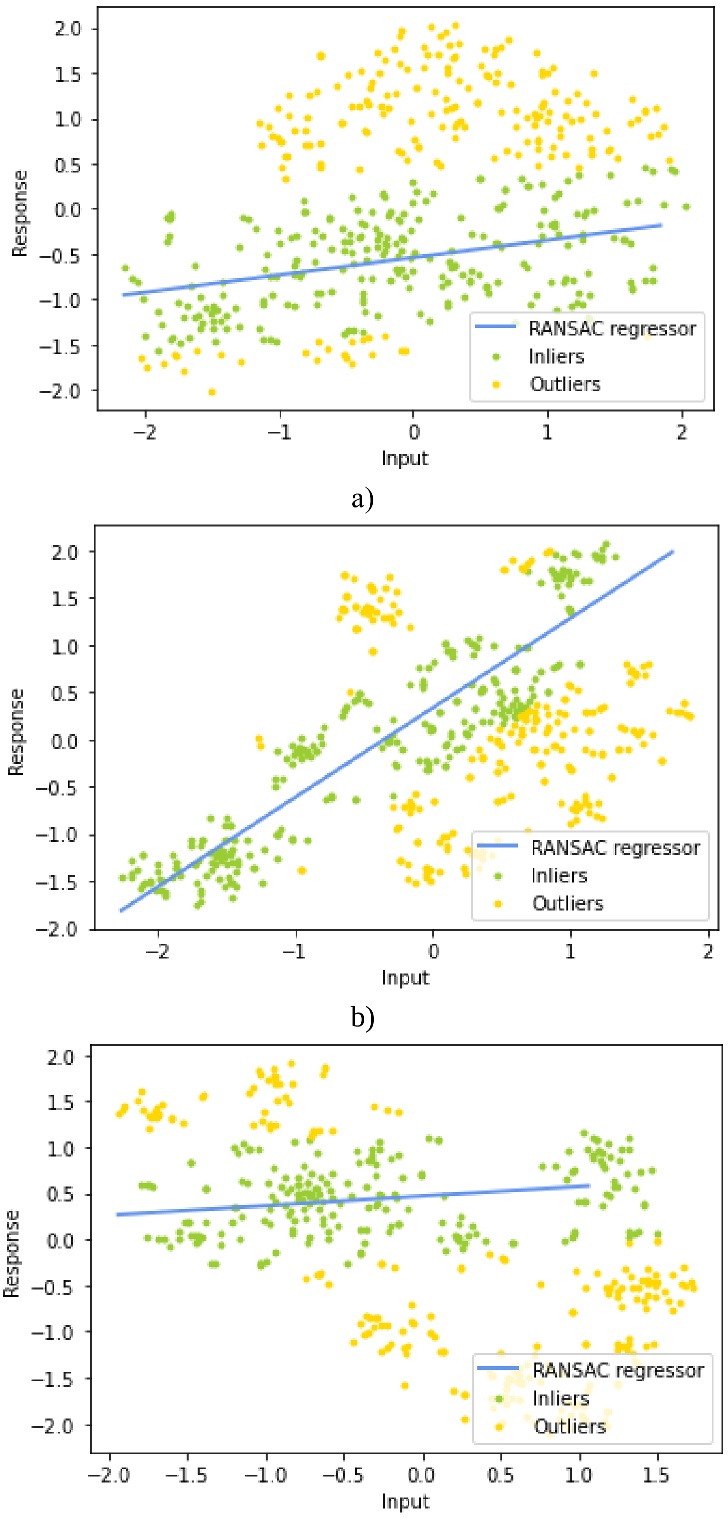
When classifying data of reduced dimension, it is important to preserve the balance of classes and data distribution when conducting binary classification. This is achieved in two ways: by engineering data classes by generating additional data samples that have the same distribution as the original data set (see section 2.7) or by balancing data samples for each member of the data class under study. In the general case, balancing is possible by dividing the class representatives into batches, or by building an ensemble of binary classifiers that compare the two data classes with the number of data representatives of the compared power.

On the example of several binary classifiers, this approach was used to classify the t-stochastic representation of feature vectors of scientific texts obtained by methods of intellectual data processing [24, 25]. To test this approach, three samples of texts on three different topics were formed, which were analyzed by the following methods of linear classification: random forests, decision trees, nearest neighbors, support vector machines (linear hypothesis), Bayesian classifier, single-layer neural network. The learning results of the algorithms are illustrated in Table 3.

Table 3
Precision, recall, score and support for Bayesian classifier on given datasets

Dataset	Precision	recall	f1-score	support
Emotion	0.94	0.87	0.90	138
Gesture	0.82	0.94	0.87	163
NLP	0.95	0.87	0.91	159
accuracy	0.90	0.89	0.89	460
macro avg	0.90	0.89	0.89	460
weighted avg	0.90	0.89	0.89	460

Based on the experiment, it was shown that the proposed approach gives a stable and reliable results with accuracy around 87% based on proposed set of scientific texts. The recognition rate, w.r.t. errors of the 1st and 2nd kind (see Fig. 4) can be increased by using decreasing affinity of the texts and using additional criterions. For instance taking in account the authors of the text, it can may eliminate biasing of the data by these hidden features, so this needs additional research.



a)

b)

c)

Figure 4: Main axes and data points in the vicinity of centroids (inliers) and outside them (outliers)

2.10. Evaluation of the effectiveness of binary classification using clustering

One of the problems existing within binary classification – the presence of data elements that are far from the centroids and data axes (see Figure 4), which causes potential problems with decreasing the decision boundary and, as a consequence, increasing the number of errors of type I and II.

In order to evaluate this task, a number of clustering methods were used to indirectly determine data labels. Given a certain distance between the centroids in binary clustering, the achievement of the desired separation band between data classes, it can be stated (with some restrictions) that clustering methods should figure out the positions of data elements from two distinct classes (in clustering of 1 cluster against 1 another), taking into account the variance of the data. By determining the elements of data clusters, we can evaluate error rates in classification such data with certain data classes [26].

The initial positions of clusters obtained by clustering methods can be used later for visual analysis and for classification using the obtained cluster labels to assess the effectiveness of classification methods in the presence of distortions and perturbations in the internal data structure [27, 28].

To test this approach, clustering was performed using several clusering methods, such as K-means, DBSCAN, CLARANS, DENCLUE, Birch using implementation written on Python language. Using this data set and this implementation, the K-means method had best efficiency and execution time.

Classification algorithms were trained on the obtained cluster locations, namely the support vector machines with linear and nonlinear hypothesis, single-layer neural network, Bayesian classifier, decision trees and other related methods. As a result of visual analysis of the hypotheses of classifiers in comparison with the hypotheses of clustering methods, the efficiency of linear classification methods for weakly separarable data classes was evaluated. Based on series of tests, the method of extreme gradient boosting gave in overall the best efficiency (about 99%) to time ratio.

2.11. Testing algorithm decision stability on two-case scenario: noisy vs denoised data

Let consider one possible approach to testing the reliability of our hypotheses, which involves introducing perturbations into the initial data set. The difference is that the introduction of perturbations, instead of approach discussed above, is performed in a space of original dimension (in this case it differs in some cases by several orders of magnitude from the input) and, accordingly, transformed space, which is converted by data dimension reduction and then by grouping features. may differ significantly in the presence or absence of noise in the initial data. So as the perturbations are affecting the hidden space in such manner that they are less visible in the space of lower dimension as well.

For this purpose, two additional experimental data sets were prepared, based on the sampling of myogram samples, but with the implementation of certain transformations. In the first case, a data filtering method was used, which cuts off existing noise and data elements by the standard deviation. Thus, data fragments that had a significant deviation (anomalies in the data) were discarded, and existing fragments in the temporal data were replaced by approximate values. In the second case, a data encoder was built that uses a noise-canceling auto encoder to convert data from the original increased-dimensional space to the reduced-dimensional space and subsequently to the increased-dimensional space, which allowed to generate a new data set with partially lost noise information.

The obtained data sets were compared similarly to the previous stages - namely with the use of transformations of data dimension and grouping of features. This allowed to save the settings of algorithms – their architecture, hyper parameters and other features as in experiments without the use of data noise and ensure comparability of results and, as a consequence, the transformed space of reduced dimension in all three cases (using two methods of data noise and without).

In order to achieve similar results, the same classification algorithms were applied: the gradient boosting algorithms, decision trees algorithms as well as a Bayesian classifier. As a result of the experiment, the following was found. The application of the proposed filtering algorithm allowed to increase the recognition efficiency for the worst case scenario by 30% and on average by 20% compared to the results shown in table 1. This may show that correct usage of data filtering as well as data augmentation and engineering is plausible approach to increase overall classification algorithm performance without creation of totally new dataset. The use of a deep auto encoder also had its own characteristics. In the presence of data without noise, the efficiency of classification decreased, which is caused by a change in the value of the variance and, accordingly, a decrease in the distance between the data classes. In turn, in the absence of data filtering, the autocoder made it possible to increase the efficiency of classification methods in the worst case by 17% and on average by 10%. This allows us to conclude that the stability of decision-making directly depends on the quality of pre-processing of input data and the potential for retraining of the algorithm (the case of using an auto encoder with data filtering methods), which in this case converts useful information.

3. Conclusion

In this work we applied classification of data using hidden features of the data and mean clustering to build initial hypotheses. These techniques are suitable for visual analysis and most important to build hypotheses about the data, that are stable, both in the original dimension and transformed.

We also suggested an approach to scale the methods on datasets of bigger scale and different origin using hidden properties of the data and data augmentation and engineering.

The deep auto encoder has shown high noise elimination efficiency with and without the application of the proposed method of data noise reduction, which indicates the potential possibility of its use to indicate the presence of noise and anomalies in the data.

In further research, we propose to investigate the architecture of the deep auto encoder, which simulates the proposed structure of the classification algorithm in order to assess the possibility of further improving the efficiency of the classification algorithm. In order to investigate further the deep learning methods, we may consider other architectures of neural networks such as convolution, recurrent and others.

4. References

[1] N. Firoozeh, A. Nazarenko, F. Alizon, B. Daille. Keyword extraction: Issues and methods. Natural Language Engineering, 26(3) (2020): 259-291. doi:10.1017/S1351324919000457

[2] J. Ventura, J. Silva. (2007). New techniques for relevant word ranking and extraction. In: Neves J., Santos M.F., Machado J.M. (eds) Progress in Artificial Intelligence. EPIA 2007. Lecture Notes in Computer Science, 4874. Springer, (2007), pp.691–702. https://doi.org/10.1007/978-3-540-77002-2

[3] M. Ortuno, P. Carpena, P. Bernaola, E. Munoz, A.M. Somoza. Keyword detection in natural languages and DNA. Europhys. Lett, 57 (5) (2002): 759-764.

[4] B. Das, S. Chakraborty. An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation. 2018. arXiv preprint arXiv:1806.06

[5] M. Labbé, L.I. Martínez-Merino, A.M. Rodríguez-Chía. Mixed Integer Linear Programming for Feature Selection in Support Vector Machine. Discrete Applied Mathematics, 261. Elsevier, (2019), pp.276-304. ff10.1016/j.dam.2018.10.025f.

[6] B. Heap, M. Bain, W. Wobcke, A. Krzywicki, S. Schmeidl. Word Vector Enrichment of Low Frequency Words in the Bag-of-Words Model for Short Text Multi-class Classification Problems, 2017. arXiv:1709.05778

[7] Y. Krak, O.Barmak, O. Mazurets. The practical implementation of the information technology for automated definition of semantic terms sets in the content of educational material. CEUR WS, Vol. 2139, (2018):245-254. DOI:10.15407/pp2018.02.245

[8] S. Robertson. Understanding inverse document frequency: On theoretical arguments for IDF, Journal of Documentation. 60 (5) (2004): 503–520.

[9] A. Aizawa. An information-theoretic perspective of tf-idf measures, Information Processing and Management. 39 (1) (2003): 45–65.

[10] M. Farouk. Measuring Sentences Similarity: A Survey. Indian Journal of Science and Technology, 12(25) (2019): 1-11. DOI: 10.17485/ijst/2019/v12i25/143977

[11] W.H. Gomaa, A. A. Fahmy. A survey of text similarity approaches. International Journal of Computer Applications, 68(13) (2013): 13-18.

[12] A.V. Barmak, Y.V. Krak, E.A. Manziuk, V.S. Kasianiuk. Information technology of separating hyperplanes synthesis for linear classifiers. Journal of Automation and Information Sciences, 51(5) (2019): 54-64. doi: 10.1615/JAutomatInfScien.v51.i5.50

[13] Iu.V. Krak, G.I. Kudin, A.I. Kulyas. Multidimensional scaling by means of pseudoinverse operations. Cybernetics and Systems Analysis, 55(1) (2019): 22-29. doi: 10.1007/s10559-019-00108-9

[14] E.L. Shimomoto, L.S. Souza, B.B. Gatto, K. Fukui. Text classification based on word subspace with term frequency. 2018. arXiv:1806.03125v1

[15] Iu.V. Krak, O.V. Barmak, S.O. Romanyshyn. The method of generalized grammar structure for text to gesture computer-aided translation, Cybernetics and Systems Analysis, 50(1) (2014): 116-123. doi: 10.1007/s10559-014-9598-4

[16] S. Bird, E. Klein, E. Loper. Natural Language Processing with Python. O'Reilly Media, 2009

[17] J. Perkins. Python Text Processing with NLTK 2.0 Cookbook. Packt Publishing, 2010.

[18] T. Mikolov, K. Chen, G. Corrado, J. Dean. Efficient Estimation of Word Representations in Vector Space. 2013. arXiv:1301.3781.

[19] A. Globerson, G. Chechik, F. Pereira, N. Tishby. Euclidean Embedding of Co-occurrence Data, Journal of Machine Learning Research, 8 (2007): 2265-2295.

[20] L. Van der Maaten, G. Hinton. Visualizing Data using t-SNE, Journal of Machine Learning Research, 9 (2008): 2579-2605.

[21] T. Mikolov. Distributed representations of words and phrases and their compositionality, Advances in Neural Information Processing Systems. 2013. arXiv:1310.4546.

[22] M.A. Fischler, R.C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography, Comm. of the ACM, 24(6) (1981): 381-395. https://doi.org/10.1145/358669.358692.

[23] A. Hast, A. Nysjö, A. Marchetti. Optimal RANSAC – Towards a Repeatable Algorithm for Finding the Optimal Set, Journal of WSCG, 21(1)(2013): 21-30.

[24] Survey of Text Mining I: Clustering, Classification, and Retrieval. Ed. by M. W. Berry. 2004, . https://www.springer.com/gp/book/9780387955636.

[25] Emerging Technologies of Text Mining: Techniques and Applications. Ed. by H. A. Do Prado, E. Ferneda. IGI Global, 2007.

[26] G.E. Hinton, R.R. Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks, Science, 313(5786) (2006):504-507. doi: 10.1126/science.1127647.

[27] E.A. Manziuk, A.V. Barmak, Y.V. Krak, V.S. Kasianiuk. Definition of information core for documents classification, J. Autom. Inf. Sci. 50(4) (2018): 25-34.

[28] S. Ioffe, C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. 2015. arXiv:1502.03167[cs].