Novel Approach to Choosing Principal Components Number in Logistic Regression

Borislava Vrigazova
Sofia University, Bulgaria

Abstract

The confirmed approach to choosing the number of principal components for prediction models includes exploring the contribution of each principal component to the total variance of the target variable. A combination of possible important principal components can be chosen to explain a big part of the variance in the target. Sometimes several combinations of principal components should be explored to achieve the highest accuracy in classification. This research proposes a novel automatic way of deciding how many principal components should be retained to improve classification accuracy. We do that by combining principal components with the ANOVA selection. To improve the accuracy resulting from our automatic approach, we use the bootstrap procedure for model selection. We call this procedure the Bootstrapped-ANOVA PCA selection. Our results suggest that this procedure can automate the principal components selection and improve the accuracy of classification models, in our example, the logistic regression.

Keywords: ANOVA, PCA, Bootstrap, logistic regression
JEL classification: C38, C52, C63
Acknowledgments: This project was supported by Sofia University's research 2021 Programme

Paper type: Research article
Received: Mar 10, 2021
Accepted: Jun 23, 2021

DOI: 10.54820/PUCR5250

Introduction

Dimensionality reduction techniques are widely used in big datasets to decrease the size of the dataset. Feature selection methods are a dimensionality reduction technique that reduces the number of features in the dataset by keeping the most informative ones. Feature selection methods include lasso (Tibshirani, 1996), adaptive lasso (Zou, 2006), nonnegative garrote (Breiman, 1995), etc. Feature selection methods are also known as variable selection methods. They identify the variables of the biggest importance for the model and discard the rest.

Textbooks provide another type of dimensionality reduction technique – the principal component analysis (PCA) (James et al., 2013). The principal component analysis transforms the dataset from high-dimensional space to low-dimensional space (James et al., 2013). PCA finds variables that have linear correlation and transforms them into principal components. Each principal component is a linear combination of the original variables in the dataset. The criterion to perform dimensionality reduction is often by keeping the number of principal components that explain the biggest percentage of the variance in the target variable. Usually, the first, second, or third principal component is enough to build a model (James et al., 2013). Principal component analysis can be used either as an exploratory data technique or a dimensionality reduction technique for predictive modeling.

The principal component analysis consists of several steps. The first step is standardization, which gives equal weight to the initial variables. Second, the covariance matrix needs to be computed. The covariance matrix helps us identify which variables are highly correlated and contain noise. Third, we compute the eigenvectors and eigenvalues of the covariance matrix to extract the principal components (James et al., 2013). Principal components are new variables that contain some of the original variables. Principal components are uncorrelated and contain a linearly transformed combination of original features. The first several features usually contain the biggest portion of the information necessary to explain the variability in the target variable (James et al., 2013). The issue with the standard approach is that the research should choose among two or three options for the number of principal components. As James et al. (2013) outline, in some cases, the researcher may need to choose between the first three and the first four principal components, and the choice is made based on the researcher's experience and many other subjective factors. This is because the first three or four principal components may have almost undiscerning success in explaining a bigger percentage of the variance in Y. To solve this issue, we propose the ANOVA-Boot-PCA-LR model to choose the number of principal components in the logistic regression.

The standard approach (James et al., 2013) has been applied in many research papers, including recent ones (Salata et al., 2021). Some researchers, however, propose updated PCA algorithms to solve this issue. For instance, Pacheco (Pacheco et al., 2013) proposes several steps to perform variable selection using the PCA. Thus, he avoids the direct issue of what number of principal components to use. Our approach is similar to his as we use the feature selection method (the ANOVA model). Unlike him, we perform PCA selection rather than variable selection. New modifications of the PCA are developed to address the lack of an automatic algorithm for principal components selection. For example, Kim et al. (2011) propose an unweighted version of principal component analysis for variable selection. Unlike us, they modify the equation of the PCA to get more accurate results when choosing the number of principal components for variable selection. Prieto-Moreno et al. (2015) use the discriminant information contained on the principal components for their selection. This is a different approach for identifying the principal components analysis in academic literature. Rather than using the percentage of variance explained, Prieto-Moreno et al. (2015) introduce a "separability measure between multiple failures" to select the number of principal components. He tests his approach in the chemical industry. His approach, however, was devised to meet the needs of a specific industry but still improves the results from the standard PCA approach.

Sharifzadeh et al. (2017) propose a sparse PCA method called SSPCA for data pre-processing and dimensionality reduction. Their PCA version modifies the way eigenvectors and eigenvalues are computed. Their modification is applicable in large datasets where the noise coming from many variables should be reduced. Their modification represents another possible direction for improving the PCA and making it automatic. Unlike them, we do not introduce modifications in the eigenvectors and eigenvalues. Gajjar et al. (2017) propose a novel method to select non-zero loadings in sparse PCA. He proposes a genetic algorithm to identify the number of non-zero loadings instead of using eigenvalues and eigenvectors as in the classical version (James et al., 2013). This is also a modification of the PCA that makes the number of principal components automatically selected. But this algorithm is relatively complicated.

In 2021 Rahoma et al. (2021) propose a new way to estimate the loading factors. Their algorithm is similar to that in Gajjar et al. (2017) as they both work with the loading factors. Rahoma's algorithm differs from Gajaar in the bootstrap methods used to evaluate the distribution information of the elements of loading vectors in principal component analysis (Efron, 1979). The elements of loading vectors are then used to obtain a sparse loading structure for the loading vectors of the PCA. As a result of their experiments, Rahoma et al. (2021) propose two novel PCA algorithms – the Bootstrap SPCA and the Sparse IPCA, both based on the bootstrap. Although all these examples proposed have improved PCA, none offers an automatic algorithm for principal components selection.

Like Rahoma et al. (2021), our research examines the bootstrap procedure and its use in the principal component analysis. We propose a novel PCA method called the ANOVA-Bootstrap-PCA. Unlike Rahoma et al. (2021) and existing academic literature, we use the bootstrap procedure to split data into training and test set. We directly identify the number of principal components necessary for building a classification model using ANOVA and PCA transformation. Thus, we do not use the eigenvectors and eigenvalues to extract the important principal components. Instead, we propose an automatic algorithm for principal components' selection for classification models.

Methodology

Our methodology consists of two types of models – the classical PCA and the new approach we propose. We compare the results from the two approaches in terms of accuracy, precision, recall, and f_1-score. We call the classical PCA approach Classical Principal Component Analysis (Classical PCA). It is described in (James et al., 2013). This approach follows the steps usually recommended by machine learning books to determine the number of principal components (James et al., 2013; Prieto-Moreno et al., 2015).

Classical PCA

The topic of determining the number of principal components in the model is central to machine learning. The general approach (James et al., 2013) consists of several steps. We conduct our experiments in Python 3.6. To run the classical PCA, we use the built-in functions in scikitlearn in Python. These include LogisticRegression(), sklearm.decomposition.PCA() and sklearn.model_selection.kFold().

1. The input variables should be standardized to attribute equal weight to each variable.
2. Then the covariance matrix should be inspected to identify and remove highly correlated variables that contain noise.
3. The input variables can then be transformed into principal components. Each principal component is a linear combination of input variables. The principal component is a new variable. Each principal component is formed in a way that contains as much information as possible. The number of the principal components is equal to the number of the input variables. However, only a few principal components contain the most important information in the model. Identifying those leads to dimensionality reduction as only the important principal components participate in the final model. To compute the principal components, eigenvectors and eigenvalues are computed. They provide information about the percentage of variance explained by each principal component.
4. We use tenfold cross-validation to divide the input data into training and test set to evaluate logistic regression using the principal components instead of the original variables.
5. We explore what number of principal components leads to the highest variance explained. We choose that number of principal components and record the accuracy and classification scores of the model.

We follow this approach to identify the principal components following the classical procedure in the textbooks (James et al., 2013).

We also test a new approach to determine the number of important principal components. We test both algorithms with logistic regression.

Proposed approach: ANOVA-Boot-PCA-LR

We call the novel approach we propose ANOVA-Boot- PCA. As we test the model with the logistic regression will denote the new approach as **ANOVA-Boot-PCA-LR**. It consists of the following steps.

1. We standardize the input data.
2. We apply PCA transformation to the standardized data.
3. We normalize the PCA transformed data between 0 and 1 to avoid negative values in the principal components.
4. We divide the input space into percentiles – 10,20,30, 40, 50,60,70,80,90 and 100. This is necessary for the tenfold bootstrap procedure.
5. For each percentile, we divide the data into training and test set in proportion 70/30 using the tenfold bootstrap described in Vrigazova (2020).
6. For each percentile, we perform ANOVA. We evaluate ten logistic test regressions for each percentile of principal components and average their accuracy and classification scores. We choose the percentile of principal components that results in the highest accuracy.
7. We then compare the number of principal components chosen, the accuracy and classification scores from the ANOVA-Boot-PCA-LR, and the classical approach.

To run the new approach we propose, we use a script that we build. For running ANOVA, PCA, and the logistic regression, we use existing functions in Python (LogisticRegression(), sklearm.decomposition.PCA() and sklearn.Pipeline (anova)). At the same time, we create a script for running the tenfold bootstrap. The tenfold bootstrap we use in step 5, and its software realization in Python 3.6 can be found in our previous study (Vrigazova, 2020).

Datasets

Table 1 shows the size of the datasets we work with. We denote the number of observations by N, the number of variables by p. The target variable is marked by Y. Table 2 presents the results of the classical PCA on all three datasets. For example, the classical PCA approach identifies the first four or five principal components as the most informative in the glass dataset. The first four principal components explain 83% of the variance in y at an accuracy rate of 54%. At the same time, the first five principal components can explain 91% of the variance of y at an accuracy rate of 53%.

Table 1
Dataset size

Dataset	N	p	Y
Glass dataset	175	9	Type
Leafshape dataset	286	7	arch
Wells dataset	3020	4	association

Source: Author's calculations

Table 2
PCA: all three datasets

	% of var explained		
	Glass dataset	Leafshape dataset	Wells dataset
PC 1	30.2%	67%	30.1%
PC 2	28.1%	18%	28.8%
PC 3	14.0%	7%	24.3%
PC 4	10.4%	4%	16.8%
PC 5	8.7%	2%	
PC 6	4.6%	1%	
PC 7	3.2%	0%	
PC 8	0.7%		
PC 9	0.0%		

Source: Author's calculations

Results

Glass dataset

The glass dataset contains nine features. Applying the classical PCA approach described in the previous section, we identify the first four principal components as the most informative ones. Table 3 contains information about the percentage of variance explained by the accuracy and the error rate at each level of principal components. We define the error rate as a 1-accuracy rate.

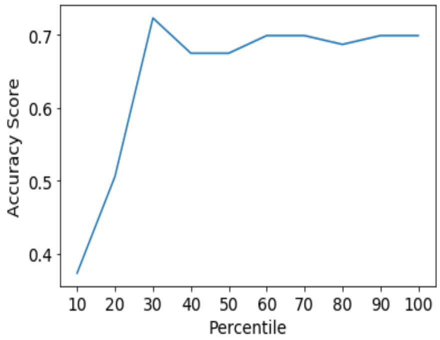
Table 3
PCA: the glass dataset

Classical approach	var explained	accuracy	error rate
PC1+PC2	58.3%	39%	61%
PC1+PC2+PC3	72%	51%	49%
PC1+PC2+PC3+PC4	83%	54%	46%
PC1+PC2+PC3+PC4+PC5	91%	53%	47%

Source: Author's calculations

Figure 1 illustrates the outcome from the ANOVA-Boot-PCA-LR we propose. Our algorithm identifies the first three principal components as the most appropriate ones, resulting in a 72% accuracy score. In contrast, the ANOVA-Boot-PCA-LR would result in an accuracy of 67% if we choose the first four or five principal components. While the classical approach chooses four or five principal components at an accuracy of 53%-54%, our approach identifies 3 principal components and achieves 72% accuracy. This result is important as it demonstrates that our algorithm not only performs an automatic selection of principal components but it can also lead to improved accuracy results.

Figure 1
The ANOVA-Boot-PCA-LR on the glass dataset: Varying the percentile of principal components selected



Source: Author's calculations

The leafshape dataset

Tables 4 present the results from the classical PCA approach on the leafshape dataset.

Table 4
PCA classical approach: the leafshape dataset

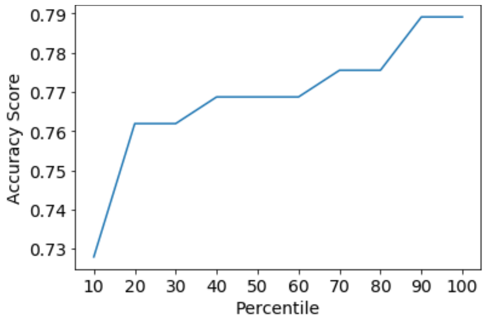
	var explained	accuracy	error rate
PC1+PC2	85.0%	77%	23%
PC1+PC2+PC3	92.3%	75%	25%

Source: Author's calculations

The leafshape dataset is a similar case to the glass dataset. The classical approach shows that we must choose between the first two and three principal components to continue our analysis. The first three principal components achieve an accuracy rate of 75%, explaining 92% of the variance, while the first two explain 85% of the variance in y and have an accuracy rate of 77%. Similarly, the number of principal components that will be chosen depends on the researcher. The first two principal components contribute the most to the variance explained, so we pick the first two principal components due to the classical approach. An important note should be made that more principal components can be chosen depending on the research purpose. However, the classical approach does not provide a direct answer to the question:" In which case do we need the first few principal components and in which do we need more principal components?".

Figure 2 shows the results from the ANOVA-Boot-PCA-LR we propose on the leafshape dataset. The figure illustrates that we will get an accuracy of 77% if we pick either the first 3, 4, or 5 principal components. However, as figure 2 illustrates, the ANOVA-Boot-PCA-LR achieves the highest accuracy of 78.9% when using 6 and 7 principal components. As a result, from the ANOVA-Boot-PCA-LR, we select the first 6 principal components to use in the logistic regression model. Thus, we propose an automatic way to answer the question, "In which case do we need the first few principal components and in which do we need more principal components?".

Figure 2
The ANOVA-Boot-PCA-LR on the leafshape dataset: Varying the percentile of principal components selected



Source: Author's calculations

In the case of the glass dataset, our algorithm identifies the first 3 principal components as the most important ones; in the case of the leafshape dataset, we use 6 out of 7 principal components. Those conclusions differ from the classical approach, where we have to manually pick the right number of features. The well's dataset has also confirmed this finding.

Wells dataset

Table 5 summarizes the results on the wells dataset from the classical PCA approach.

Table 5
PCA classical approach: the wells dataset

	var explained	accuracy	error rate
PC1+PC2	58.9%	58%	42%
PC1+PC2+PC3	83.2%	57%	43%
PC1+PC2+PC3+PC4	100.0%	57%	43%

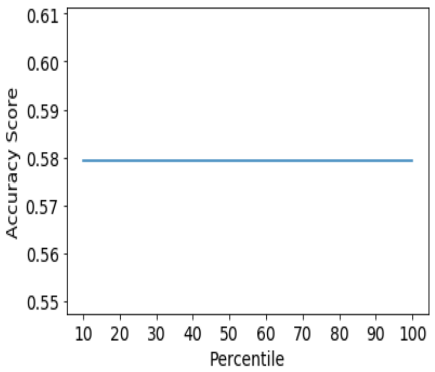
Source: Author's calculations

According to table 5, the researcher should choose the first two, three, or four principal components. The dataset contains 4 principal components. Table 5 shows that the accuracy rate does not change significantly whether we choose 2,3, or 4 principal components. So, again the researcher should decide manually which principal components to choose. However, we would get similar accuracy regardless of the number of principal components chosen in this case.

Figure 3 demonstrates the outcome of our proposed algorithm. It shows that we would achieve an accuracy rate of 58% regardless of the number of principal components we use in the logistic regression. This finding confirms the result from the classical approach. However, the ANOVA-Boot-PCA-LR automatically performs the PCA analysis, providing direct guidelines on the number of principal components that would produce the highest accuracy in the logistic regression. As a result, we can select a small number of principal components from our algorithm without losing accuracy. This is an important advantage of our algorithm in the case of datasets with a large number of principal components.

Figure 3

The ANOVA-Boot-PCA-LR on the wells dataset: Varying the percentile of principal components selected



Source: Author's calculations

Classification scores

During our research, we also calculate classification scores like precision, recall, and f_1-score. As table 6 below shows, the classification scores from the two algorithms are very similar for the glass dataset. The ANOVA-Boot-PCA-LR does not lead to a decrease in the classification scores on the glass dataset. The classification metrics from both algorithms are similar on the glass dataset. The new approach we propose improves the precision and recall for classes 1 and 2.

Table 6

Classification scores of the classical PCA approach vs. the ANOVA-Boot-PCA-LR: The glass dataset

Classical approach	**PC1+PC2+PC3+PC4**			
Class	Precision	Recall	F1-score	Support
1	0.55	0.74	0.63	70
2	0.62	0.42	0.50	76
7	0.90	0.90	0.90	29
Average	0.64	0.63	0.62	175
ANOVA-Boot-PCA-LR (3 principal components)				
Class	Precision	Recall	F1-score	Support
1	0.69	0.74	0.71	34
2	0.69	0.68	0.68	37
7	0.91	0.83	0.87	12
Average total	0.73	0.72	0.72	83

Source: Author's calculations

Table 7 shows the results of the leafshape dataset. In the case of the leafshape dataset, the new algorithm predicts better the precision of class 1 and the recall of class 0. At the same time, the classical PCA predicts better the recall of class 1 and precision of class 0. Overall, both algorithms have the same precision, while the classical approach provides slightly better recall and f-1 score measures. Table 8 shows the results on the wells dataset. As table 8 shows, classification metrics on the wells dataset are similar.

Table 7
Classification scores of the classical PCA approach vs. the ANOVA-Boot-PCA-LR: The leafshape dataset

Classical approach	**PC1+PC2**			
Class	Precision	Recall	F1-score	Support
0	0.84	0.90	0.87	192
1	0.76	0.66	0.70	94
Average	0.81	0.82	0.81	286
ANOVA-Boot-PCA-LR (6 principal components)				
Class	Precision	Recall	F1-score	Support
0	0.75	0.99	0.85	99
1	0.94	0.31	0.47	48
Average total	0.81	0.77	0.73	147

Source: Author's calculation

Table 8
Classification scores of the classical PCA approach vs. the ANOVA-Boot-PCA-LR: The wells dataset

Classical approach	**PC1+PC2+PC3**			
Class	Precision	Recall	F1-score	Support
0	0.58	1.00	0.73	1743
1	0.20	0.00	0.00	1277
Average	0.42	0.58	0.42	3020
ANOVA-Boot-PCA-LR (4 principal components)				
Class	Precision	Recall	F1-score	Support
0	0.57	1.00	0.72	843
1	0.00	0.00	0.00	647
Average total	0.32	0.57	0.41	1490

Source: Author's calculations

As the results for all three datasets indicate, the classification scores from both algorithms are similar. More experiments may be conducted with the leafshape dataset to improve the results from the ANOVA-Boot-PCA-LR. Nevertheless, our proposed algorithm provides an automatic way to select the number of principal components to avoid the manual steps. The criteria we propose for choosing the number of principal components is the highest accuracy rate.

The proposed algorithm has advantages that can be applied to large datasets. Selecting the number of principal components in datasets with many principal components can be a tedious and time-consuming manual process. As a result, we propose the ANOVA-Boot-PCA-LR algorithm to select principal components in the logistic regression automatically.

Discussion

The algorithm we propose is a novel approach to selecting the number of principal components for classification. The ANOVA-Boot-PCA-LR algorithm provides a fast and effective way to select the number of principal components and retain/improve the model's accuracy.

This algorithm needs further research, though. Academic literature confirms the application of the classical PCA either as a self-sufficient data exploratory or as an additional model combined with prediction and classification models. The algorithm we propose can be used in combination with other predictive and classification models. In this research, we examine its use with logistic regression. So, deeper research into other classification and predictive models should be done. Also, we do not propose an automatic algorithm when the principal component analysis is used as a stand-alone method to explore data. The algorithm can further be developed to apply to a wide range of datasets, for example, improve the results on the leafshape dataset.

Another limitation of our research is the value of the random iterator (the seed parameter) in Python that allows for the repetition of the results. We do not investigate how this value affects our results, and we do not recommend a concrete value for the seed parameter. This value may be specific for each dataset, and it can interact with the classification model used. As this is the subject of further research, we do not examine this matter in our research. This paper aims to propose a new automatic algorithm to select the number of principal components in the logistic regression. We call this algorithm the ANOVA-Boot-PCA-LR.

Conclusion

In conclusion, we developed a simple algorithm for automatically detecting the number of principal components used in the logistic regression. The advantages of our algorithm include simplicity as it is based on existing algorithms, is easy to interpret, and provides more efficient results in terms of accuracy.

Many authors of machine learning textbooks recommend detecting the number of principal components based on the percentage of variance explained, but this algorithm is not automatic. It is an issue when the percentage of variance explained identifies two possible principal components, and their accuracy is close. Many authors try to solve this problem by modifying parts of the equation of the PCA (for instance, eigenvectors, eigenvalues, etc.) or by using variable selection with PCA (Salata et al., 2021; Pacheco et al., 2013, Kim et al., 2011). The classical approach recommends manually deciding which number is most appropriate. However, our approach fixes this issue by conducting principal component selection rather than variable selection. It performs an automatic selection of the number of principal components to be used in the logistic regression.

This advantage of our algorithm is important in big datasets where the number of principal components can be big. The traditional manual selection of principal components can give several alternatives, and choosing between them can be hard and time-consuming. For instance, if the traditional theory outlines the first five, six, seven, or eight principal components as possible options, the researcher needs a criterion to choose among them. In many cases, the criteria are subjective. On the other hand, the ANOVA-Boot-PCA-LR algorithm we propose gives only one number of principal components, removing the subjectiveness from manually choosing the number of principal components. This chooses principal components faster.

The approach we propose has several limitations that need to be further researched. First, more experiments with larger datasets should be conducted to demonstrate the advantage of our algorithm better. Datasets should include a large number of variables. Our results in this research show that the ANOVA-Boot-PCA-LR gives similar results to the classical approach. For example, suppose the classical approach identifies two options – the first three and four principal components. In that case, our algorithm identifies as the most appropriate only one of the options provided by the classical approach. Researching with a bigger dataset would test the hypothesis that our algorithm provides automatic, reliable principal component selection in large datasets.

Second, deeper research into the seed value should be conducted to discover whether it affects the performance of the proposed algorithm. The seed value in Python controls for the reproducibility of the results. Deeper research should be conducted into how this value affects the output of our algorithm. Does it change the number of principal components selected? How and why if that is the case.
Third, deeper research is necessary about whether changing the values of the logistic regression parameters would affect the number of principal components chosen and whether it stays close to that outlined from the classical approach. This would allow a deeper understanding of the automatic nature of our algorithm and its applications to large datasets.

Fourth, experiments with other classification methods are necessary to test the feasibility of the ANOVA-Boot-PCA-LR to other classification models. This would expand the practical applications of the model we propose and show its universal nature, outlining in what cases it can and cannot be applied to other classification methods. Despite all possible directions for improving our research, we believe our research provides the first step to an efficient and fast automatic algorithm for principal components selection in classification problems. Thus, we can avoid the manual process and achieve better accuracy, a novel approach in academic literature.

References

1. Breiman, L. (1995), "Better Subset Regression Using the Nonnegative Garrote", Technometrics, Vol. 37 No. 4, pp. 373-384.
2. Efron, B. (1979), "Bootstrap methods: another look at the jackknife", The Annals of Statistics, Vol. 7 No. 1, pp. 1-26.
3. Gajjar, S., Kulahci, M., Palazoglu, A. (2017), "Selection of non-zero loadings in sparse principal component analysis", Chemometrics and Intelligent Laboratory Systems, Vol. 162, pp. 160-171.
4. James, G., Witten, D., Hastie, T., Tibshirani, R. (2013). An Introduction to Statistical Learning, Springer, New York.
5. Kim, S., Rattakorn, P. (2011), "Unsupervised feature selection using weighted principal components", Expert Systems with Applications, Vol. 38 No. 5, pp. 5704-5710.
6. Pacheco, J., Casado, S., Porras, S. (2011), "Exact methods for variable selection in principal component analysis: Guide functions and pre-selection", Computational Statistics & Data Analysis, Vol. 57 No. 1, pp. 95-111.
7. Prieto-Moreno, A., Llanes-Santiago, O., García-Moreno, E. (2015), "Principal components selection for dimensionality reduction using discriminant information applied to fault diagnosis", Journal of Process Control, Vol. 33, pp. 14-24.
8. Rahoma, A., Imtiaz, S., Ahmed, S. (2021), "Sparse principal component analysis using bootstrap method", Chemical Engineering Science, Vol. 246, paper no. 116890.
9. Salata, S., Grillenzoni, C. (2021), "A spatial evaluation of multifunctional Ecosystem Service networks using Principal Component Analysis: A case of study in Turin, Italy", Ecological Indicators, Vol. 127, pp. 1-13.
10. Sharifzadeh, S., Ghodsi, A., Clemmensen, L., Ersboll, B. (2017), "Sparse supervised principal component analysis (SSPCA) for dimension reduction and variable selection", Engineering Applications of Artificial Intelligence, Vol. 65, pp. 168-177.

11. Tibshirani, R. (1996), "Regression Shrinkage and Selection via the Lasso", Journal of the Royal Statistical Society, Series B (Methodological), Vol. 58 No. 1, pp. 267-288.
12. Vrigazova, B. (2020), "Tenfold Bootstrap as Resampling Method in Classification Problems", in Proceedings of the ENTRENOVA-ENTerprise REsearch InNOVAtion Conference, virtual conference, pp. 74-83.
13. Zou, H. (2006), "The adaptive lasso and its oracle properties", Journal of the American statistical association, Vol. 101 No. 476, pp. 1418-1429.

About the author

Borislava Vrigazova is currently a Ph.D. candidate in Data Science at Sofia University, Bulgaria. She obtained a master's degree in Statistics, financial econometrics, and actuarial studies in 2017 after a bachelor's degree in Economics at the same university. Her research areas include practical applications of machine learning algorithms for prediction and boosted performance. Also, she is an expert in applying big data techniques to small datasets in the field of economics as an alternative to traditional econometrics theory. She challenges traditional econometric modeling techniques to find connections among variables from institutional economics by combining feature selection methods and big data prediction models. As a result, new applications of machine learning techniques to economic data appear. The author can be contacted at **vrigazova@uni-sofia.bg**