Tenfold Bootstrap as Resampling Method in Classification Problems

Borislava Vrigazova
Sofia University "St. Kliment Ohridski", Bulgaria

Abstract

In this research, we propose the bootstrap procedure as a method for train/test splitting in machine learning algorithms for classification. We show that this resampling method can be a reliable alternative to cross validation and repeated random test/train splitting algorithms. The bootstrap procedure optimizes the classifier's performance by improving its accuracy and classification scores and by reducing computational time significantly. We also show that ten iterations of the bootstrap procedure are enough to achieve better performance of the classification algorithm. With these findings, we propose a solution to the problem of how to reduce computing time in large datasets, while introducing a new practical application of the bootstrap procedure.

Keywords: the bootstrap, cross validation, repeated train/test splitting
JEL classification: C38, C52, C55

Acknowledgments: The author expresses her gratitude to prof. Ivan Ivanov.

Paper type: Research article
Received: Jul 2, 2020
Accepted: Jul 12, 2020

Introduction

Long computational time is a problem that often occurs in big datasets. Computationally exhaustive classification methods result in high accuracy but slow computing time. Computational time increases with the increase of the size of the dataset. Resampling methods like the tenfold cross validation, leave-one-out cross validation and repeated random train/test split perform validation of the model. The computational time of a classification method can increase or decrease Vrigazova and Ivanov (2020a), while keeping high accuracy, depending on the resampling method chosen. The aim of this paper is to propose the bootstrap procedure as a resampling method for classification, which can reduce computational time significantly, while preserving high accuracy.

Unlike previous research (Vrigazova & Ivanov, 2020a), we show that the tenfold bootstrap procedure (Vrigazova & Ivanov, 2020b) can achieve accuracy that is similar to other resampling methods but using training/test proportion of 20/80. In previous research (Vrigazova & Ivanov, 2020a) we showed that the tenfold bootstrap is competitive to other resampling methods in classification problems when having 70% of the observations as training test and 30% as test set. In this research, we show that the bootstrap procedure results in high accuracy and faster computing time even if the proportion for splitting into training and test set is not the standard one. We also show that regardless of the size of the test set, the bootstrap procedure produces high accuracy for a shorter period of time compared to other resampling methods like the tenfold cross validation, leave-one-out cross validation and repeated random train/test split. Thus, we propose a way to shorten time for classification, while preserving the accuracy of the model.

Literature review

The bootstrap was first introduced in 1979 by Efron (Efron, 1979). It has wide applications in various fields. For example, it can be used for inferring the unknown distribution of data, thus allowing confidence intervals to be built. One thousand iterations of the bootstrap can make data's distribution closer to the Gaussian distribution. As a result, the bootstrap is widely used in Monte Carlo simulations MacKinnon (2002). The bootstrap is also used in the random forest and for pruning decision trees (Breiman, 1996). In 1992, Breiman (Breiman, 1992) devised the little bootstrap procedure for applications as a resampling method in small datasets. Later, in 1995, he showed that the little bootstrap procedure can be used as a resampling method in data with fixed regressors (Breiman, 1995). He recommended the cross validation as a resampling technique in datasets with random regressors. In 2018 Vrigazova (Vrigazova, 2018) showed that the little bootstrap procedure (Breiman, 1992) can successfully be used for feature selection in panel data with fixed effects.

The bootstrap procedure has widely been used for estimating unknown distributions. Its properties as a resampling method have started to be more thoroughly researched lately. In 1997, Efron and Tibshirani tested the performance of the 0.632 + bootstrap procedure in machine learning methods for classification (k-nearest neighbor, logistic regression and decision tree) suggesting that the bootstrap can be an alternative to cross validation (Efron & Tibshirani, 1997). Since then few experiments have been made in this direction. The standard resampling procedure for splitting the dataset into training and test set in classification problems has been cross validation. Repeated random training/test split is also used as an alternative to cross validation.

Based on the research of Efron and Tibshirani (Efron & Tibshirani, 1997), we raised the question if the bootstrap procedure can be used as a technique for splitting into training and test set and be a reliable alternative to cross validation. In a previous research (Vrigazova & Ivanov, 2020a, 2020b), we show that the bootstrap procedure is a reliable resampling procedure for ANOVA variable selection in the logistic regression, decision tree, k-nearest neighbour and the support vector machines when using 70/30 proportion for train/test split. In this research, we show that the tenfold bootstrap procedure can be alternative to other resampling methods without performing variable selection.

We show that the bootstrap procedure for classification methods provides high accuracy and accelerates computing time even if the train/test split proportion is 20/80. When using 70/30 splitting proportion, the bootstrap accelerates the performance of the classification methods compared to cross validation and repeated random train/test split and preserves the accuracy of the model. Using splitting proportion of 20/80 provides similar accuracy, while reducing computational time even more than using the bootstrap with 70/30 splitting proportion. Thus, we propose a novel way to further reduce computational time of classification methods applied to big datasets.

Next section describes the methodology we propose. Section 4 comments on the data used and the results from our proposed methodology. Sections 5 and 6 conclude and summarize possibilities for future research.

Methodology

We compare the performance of the logistic regression (Pampel, 2000) and the decision tree classifier (James et al., 2013) in terms of time, accuracy and error rate. We produced several types of experiments.

First, we splitted each dataset into training and test set using tenfold cross validation (Hoerl & Kennard, 1970). We used 70/30, 50/50, 30/70 and 20/80 as proportions for train/test split. We then fitted each classification method and calculated time, accuracy and error rate. We used the Python 3.7 function model_selection.cross_val_score() with the parameter cv fixed to 10 to perform the tenfold cross validation.

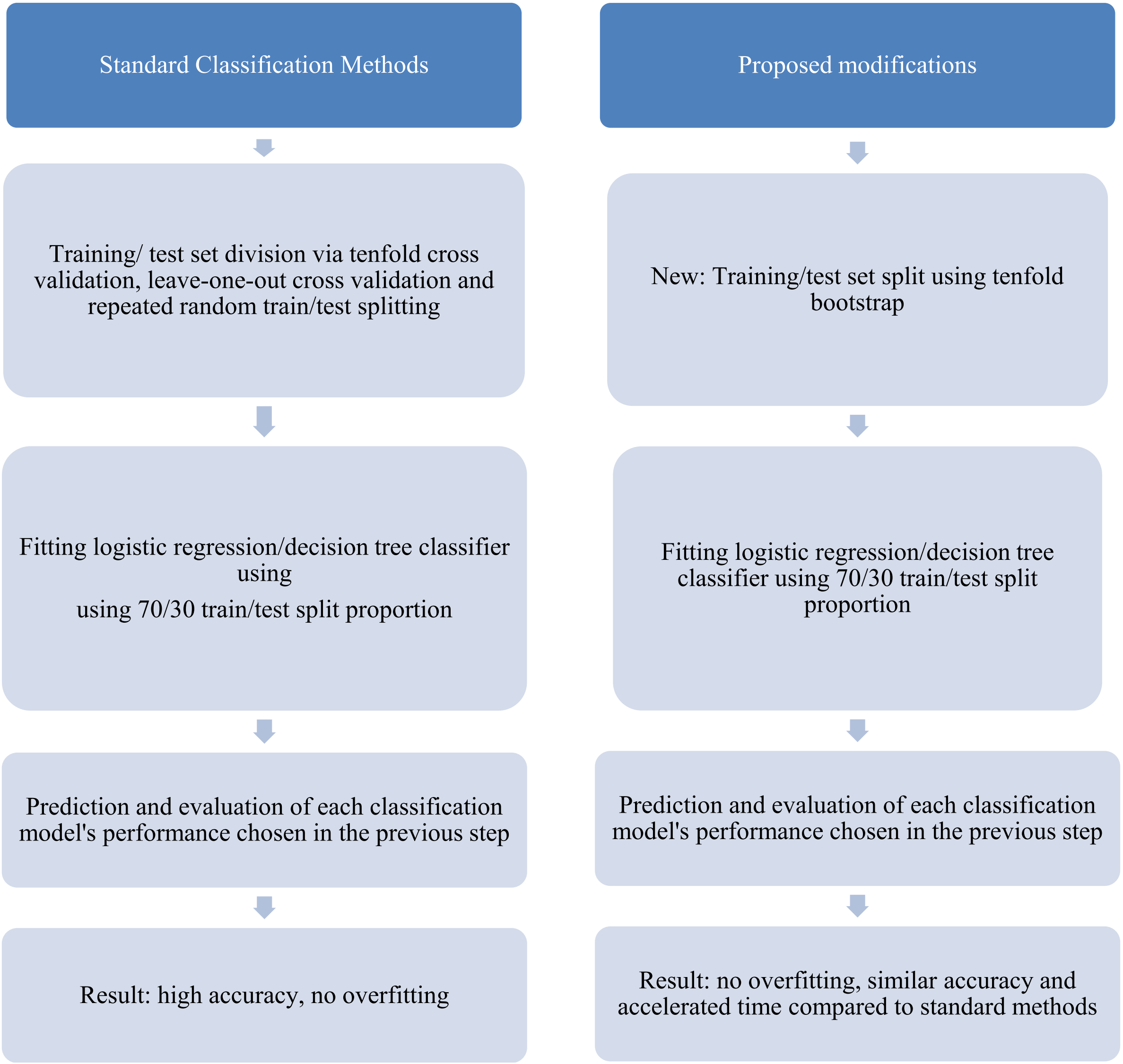
We also used leave-one-out cross validation (Wong, 2015) as alternative to tenfold cross validation. We use the same train/test split proportions as in the tenfold cross validation. To run the leave-one-out cross validation, we use the function model_selection.LeavePOut(p=1) in Python by fixing the parameter p to 1. We apply the leave-one-out cross validation to the three classification methods.

As a third resampling alternative, we apply the repeated random train/test split (Krstajic et al., 2014) to the logistic regression, decision tree classifier and the k-nearest neighbour. The function ShuffleSplit() can be used to randomly and repeatedly divide the dataset into training and test set. We fixed the parameter n_splits to 10 and the random_state parameter to 7 to be able to replicate the results.

We also ran the tenfold bootstrap (Vrigazova & Ivanov, 2019) procedure as alternative to the three resampling methods. We introduced the bootstrap procedure for classification problems in (Vrigazova & Ivanov, 2019). In this research, we used 70/30, 50/50, 30/70 and 20/80 splitting proportions to apply to the bootstrap in the logistic regression and decision tree classifier.

Because of our experiments, we propose applying the tenfold bootstrap procedure with train/test split proportion of 70/30. This proportion combined with the tenfold bootstrap procedure resulted in high accuracy and much faster computing time. Figure 1 summarizes the standard approach and the novel approach in this study.

Figure 1
Standard vs proposed resampling methods


Standard Classification Methods
Proposed modifications
Training/ test set division via tenfold cross validation, leave-one-out cross validation and repeated random train/test splitting
New: Training/test set split using tenfold bootstrap
Fitting logistic regression/decision tree classifier using
using 70/30 train/test split proportion
Fitting logistic regression/decision tree classifier using 70/30 train/test split proportion
Prediction and evaluation of each classification model's performance chosen in the previous step
Prediction and evaluation of each classification model's performance chosen in the previous step
Result: high accuracy, no overfitting
Result: no overfitting, similar accuracy and accelerated time compared to standard methods

Source: Author's presentation

To compare the performance of each model, we use time, accuracy and error rate as defined in (Vrigazova & Ivanov, 2020a). We summarize our results in the next section.

Results

To perform our research we use three fully available datasets. These are the monica, food and adult datasets. The can be downloaded at www.kaggle.com. The monica dataset is the smallest one, containing 6,367 observations and 11 independent variables. The dependent variable is called 'outcome". The food dataset contains 23,971 observations and 5 independent variables, with the 'sex' variable being the dependent one. The last dataset is the adult dataset with 45,222 observations and 11 independent variables. The dependent variable is 'income'. We chose the datasets to be increasing in size so that we can observe the performance of the resampling methods in large datasets. We did not apply preliminary transformations on the input variables.

Table 1 presents the results from the resampling methods applied to the logistic regression.

Table 1
Logistic regression results

Dataset	Train/test ratio	Resampling method	Accuracy	Error rate	Time (s)
monica	70/30				
		10-fold cross validation	87.8	12.2	1.84
		LOO	87.8	12.2	105.56
		Random train/test split	87.9	12.1	0.05
		10-fold bootstrap	87.8	12.2	0.02
monica	50/50				
		10-fold cross validation	87.7	12.3	0.14
		LOO	87.7	12.3	44.70
		Random train/test split	87.9	12.1	0.05
		10-fold bootstrap	87.4	12.6	0.01
monica	30/70				
		10-fold cross validation	87.9	12.1	0.09
		LOO	87.9	12.1	18.32
		Random train/test split	88.0	12.0	0.14
		10-fold bootstrap	87.5	12.5	0.01
monica	20/80				
		10-fold cross validation	87.8	12.2	0.05
		LOO	88.0	12.0	7.68
		Random train/test split	87.4	12.6	0.04
		10-fold bootstrap	87.5	12.5	0.01
food	70/30				
		10-fold cross validation	86.2	13.8	0.83
		LOO	86.2	13.8	306.52
		Random train/test split	86.4	13.6	0.05
		10-fold bootstrap	86.1	13.9	0.03
food	50/50				
		10-fold cross validation	86.2	13.8	0.10
		LOO	86.2	13.8	145.48
		Random train/test split	85.8	14.2	0.15
		10-fold bootstrap	86.1	13.9	0.02
food	30/70				
		10-fold cross validation	86.3	13.7	0.07
		LOO	86.3	13.7	55.77
		Random train/test split	86.0	14.0	0.04
		10-fold bootstrap	86.0	14.0	0.01
food	20/80				
		10-fold cross validation	86.1	13.9	0.06
		LOO	86.1	13.9	28.24
		Random train/test split	86.0	14.0	0.04
		10-fold bootstrap	86.0	14.0	0.01
adult	70/30				
		10-fold cross validation	79.8	20.2	1.78
		LOO	79.7	20.3	6440.27
		Random train/test split	79.1	20.9	0.23
		10-fold bootstrap	79.1	20.9	0.23
adult	50/50				

		10-fold cross validation	79.7	20.3	0.99
		LOO	79.7	20.3	3029.14
		Random train/test split	79.0	21.0	0.19
		10-fold bootstrap	79.2	20.8	0.12
adult	30/70				
		10-fold cross validation	79.5	20.5	0.41
		LOO	79.6	20.4	659.80
		Random train/test split	79.1	20.9	0.14
		10-fold bootstrap	79.2	20.8	0.07
adult	20/80				
		10-fold cross validation	79.2	20.8	0.30
		LOO	79.2	20.8	273.80
		Random train/test split	79.1	20.9	0.10
		10-fold bootstrap	79.3	20.7	0.06

Source: Author's calculations

Table 1 shows that the slowest resampling method is the leave-one-out cross validation (LOO). Regardless of the size of the dataset and the splitting proportion, the leave-one-out cross validation was between 18 and 6440 times slower than the rest of the resampling methods. Despite this, it produced accuracy and error rate similar to the tenfold cross validation. Its computational disadvantage makes it rarely used in large datasets. The tenfold cross validation is faster than the leave-one-out cross validation but slower than the random train/test split and the tenfold bootstrap.

The tenfold bootstrap proved to be the fastest resampling method. Its computational advantage was significant. For instance, the adult dataset (70/30) was classified by the LOO in 6440 seconds, while the bootstrap did that in 0.23 seconds. The tenfold cross validation led to the output from the logistic regression in 1.78 seconds, while the random train/test split produced results similar to the bootstrap. The two produced accuracy of 79.1%, while the cross validation – 79.8%. However, the accuracy of the bootstrap is stable regardless of the splitting proportion, similarly to the random train/test split. Unlike them, the cross validation's accuracy fell from 79.8% to 79.2%. Therefore, possible overfitting can be present in the cross validation.

Accuracy did not change so drastically with reducing the training set. All resampling methods provided error rate between 13.6% and 14%. The bootstrap resulted in highest accuracy of 86.1% (70/30), while the tenfold cross validation – 86.2% (70/30). The random train/test split resulted in accuracy of 86.4% (70/30). However, when the train/test random split was applied with 50/50 splitting proportion, its accuracy dropped to 85.8%. The 30/70 proportion lead to increased accuracy (86.3%) from the tenfold cross validation. Changing the splitting proportion did not lead to significant changes in the logistic regression's error rate but significantly accelerated computing time. It ran 306 times faster than the leave-one-out cross validation and 27 times faster than the tenfold cross validation.

Splitting the dataset into 70/30 proportion led to 87.8% accuracy from the cross validation and the bootstrap. The exception was the leave-one-out cross validation that produced accuracy of 87.9%. When using smaller training set, the random train/test split resulted in 88% accuracy, while the other methods had a slight increase. However, the bootstrap procedure was the fastest.

We consider the bootstrap procedure as suitable for train/test set split for the logistic regression in large dataset as it provided similar results to the tenfold cross validation that did not change much with the decreasing of the size of the training set. We recommend using the 70/30 proportion to preserve accuracy similar to the tenfold cross validation, while reducing computational time.

Similar observations can be made for the decision tree classifier. Table 2 summarizes its performance.

Table 2
Resampling methods for the Decision Tree Classifier

dataset	train/test ratio	resampling method	accuracy	error rate	time
monica	70/30				
		10-fold cross validation	80.7	19.3	0.08
		LOO	80.8	19.2	40.92
		Random train/test split	81.3	18.7	0.03
		10-fold bootstrap	80.5	19.5	0.01
monica	50/50				
		10-fold cross validation	80.9	19.1	0.07
		LOO	81.7	18.3	20.40
		Random train/test split	80.5	19.5	0.03
		10-fold bootstrap	80.6	19.4	0.01
monica	30/70				
		10-fold cross validation	81.5	18.5	0.05
		LOO	82.1	17.9	8.11
		Random train/test split	80.5	19.5	0.03
		10-fold bootstrap	80.5	19.5	0.01
monica	20/80				
		10-fold cross validation	81.2	18.8	4.20
		LOO	80.1	19.9	0.02
		Random train/test split	80.0	20.0	0.01
		10-fold bootstrap	80.5	19.5	0.01
food	70/30				
		10-fold cross validation	83.5	16.5	0.67
		LOO	83.6	16.4	1383.83
		Random train/test split	83.9	16.1	0.14
		10-fold bootstrap	83.7	16.3	0.09
food	50/50				
		10-fold cross validation	83.5	16.5	0.48
		LOO	83.5	16.5	635.69
		Random train/test split	83.9	16.1	0.10
		10-fold bootstrap	83.7	16.3	0.06
food	30/70				
		10-fold cross validation	83.7	16.3	0.26
		LOO	83.2	16.8	211.96
		Random train/test split	83.7	16.3	0.07
		10-fold bootstrap	83.5	16.5	0.04
food	20/80				
		10-fold cross validation	83.7	16.3	0.17
		LOO	83.6	16.4	100.17
		Random train/test split	83.8	16.2	0.05
		10-fold bootstrap	83.5	16.5	0.03
adult	70/30				
		10-fold cross validation	80.9	19.1	1.65
		LOO	80.6	19.4	4815.19
		Random train/test split	79.6	20.4	0.25
		10-fold bootstrap	80.4	19.6	0.17
adult	50/50				
		10-fold cross validation	80.5	19.5	0.86

		LOO	80.5	19.5	2566.74
		Random train/test split	79.8	20.2	0.19
		10-fold bootstrap	80.4	19.6	0.12
adult	30/70				
		10-fold cross validation	80.8	19.2	0.49
		LOO	80.7	19.3	858.83
		Random train/test split	79.6	20.4	0.12
		10-fold bootstrap	80.2	19.8	0.08
adult	20/80				
		10-fold cross validation	79.8	20.2	0.33
		LOO	79.8	20.2	339.88
		Random train/test split	79.0	21.0	0.10
		10-fold bootstrap	80.0	20.0	0.06

Source: Author's calculations

The bootstrap optimizes the performance of the decision tree classifier as well. The bootstrap produced the output from the decision tree classifier (70/30) in 0.17 seconds on the adult dataset, while the tenfold cross validation in 1.65 seconds. As table 2 shows the bootstrap resulted in accuracy and error rate, similar to those from the other resampling methods. However, the computational time was much faster. In some cases, the bootstrap decreased the error rate of the model. Like the logistic regression, the decision tree classifier suffered loss of accuracy after decreasing the size of the training set. We recommend using the bootstrap procedure with splitting ratio of 70/30. It is important to be noted that the datasets did not have any preliminary transformations. In previous research (Vrigazova & Ivanov, 2020a) we show that if the input data have been standardized and variable selection is performed, the bootstrap produces higher accuracy than other resampling methods. This is also valid for the logistic regression.

We also showed (Vrigazova & Ivanov, 2020a) that the bootstrap needs ten iterations to produce these results. Increasing the number of the iterations produces the same accuracy but increases computational time. The computational advantage of the bootstrap becomes obvious with the increase of the dataset. The bootstrap produced similar accuracy regardless of the splitting proportion. The cross validation methods and the random train/test split varied in accuracy depending on the splitting ratio. We believe the bootstrap can also be applied with other splitting proportions like those that the ones presented in this research.

The rest of the resampling method, however, suffer from loss of accuracy when changing the splitting ratio from 70/30 to 30/70 or 20/80. This result is confirmed by another research we made (Vrigazova & Ivanov, 2020a). There we show that the support vector machines classifier with tenfold bootstrap and 30/70 splitting ratio can produce similar accuracy to that produced from the tenfold cross validation with ratio 70/30. The advantage is the computing time. As tables 1 and 2 show, this finding holds for the logistic regression and the decision tree classifier. However, when applied to untransformed data without variable selection, the bootstrap can be used with 50/50 splitting ration instead of 30/70.

This is an important finding as the bootstrap can additionally decrease computing time by applying smaller size of the training set but preserve accuracy. The other resampling methods suffer from fluctuations, so changing the splitting ratio affects the error rate. The computing time reduced but accuracy as well. Another important finding is that untransformed data are much more sensitive to the splitting ration than transformed data. This affects the accuracy of the classification method regardless of the resampling method used. The bootstrap is affected by non-transformed data the least.

Conclusion

In this paper, we present new application of the bootstrap for resampling method in classification method. Despite its computational advantage, several points need consideration. First, the bootstrap works better with transformed data, where the accuracy can be boosted as well. Second, the characteristics of the dataset can also affect the performance of the bootstrap. As a result, the bootstrap can be suitable for one dataset and unsuitable for another. We recommend comparison between the bootstrap and the cross validation to identify the most suitable one for the dataset.

We believe that the bootstrap procedure can be used as alternative to cross validation in some cases. The advantages of the procedure include decreased computational time and stable accuracy that does not depend on the splitting ratio. However, the characteristics of the dataset and the preliminary data transformations may affect the outcome from the bootstrap.

Moreover, the advantages of the bootstrap procedure are more visible when applied to ANOVA variable selection procedure. Without variable selection, the bootstrap procedure can be more suitable for one dataset more than for another. One possible way to make the bootstrap procedure suitable for more datasets without dimensionality reduction is by standardizing the variable or using 70/30 train/test splitting proportion.

Despite its disadvantages, the bootstrap procedure can be a powerful tool to reduce computational time in large datasets. Additional research can be made on how to further improve the accuracy of classification models resulting from nontransformed, nonreduced bootstrapped classification.

References

1. Breiman, L. (1992), "The little bootstrap and other methods for dimensionality selection in regression: x-fixed prediction error", Journal of American Statistical Association, Vol. 87, No. 419, pp. 738-754.
2. Breiman, L. (1995), "Better subset regression using the nonnegative garrote", Technometrics, Vol. 37, No. 4, pp. 373-384.
3. Breiman, L. (1996), "Bagging predictors", Machine Learning, Vol. 24, No. 2, pp. 123-140.
4. Efron, B. (1979), "Bootstrap methods: another look at the jackknife", The Annals of Statistics, Vol. 7, No. 1, pp. 1-26.
5. Efron, B., Tibshirani, R., (1997), "Improvements on cross-validation: the .632+ bootstrap method", Journal of the American Statistical Association, Vol. 92, No. 438, pp. 548-560.
6. Hoerl, A. E., Kennard, R. W. (1970), "Ridge regression: applications to nonorthogonal Problems", Technometrics, Vol. 12, No. 1, pp. 69-82.
7. James, G., Witten, D., Hastie, T., Tibshirani, R. (2013), An introduction to statistical learning, Springer, New York.
8. Krstajic, D., Buturovic, L. J., Leahy, D. E., Thomas, S. (2014), "Cross-validation pitfalls when selecting and assessing regression and classification models", Cheminformatics, vol. 6, No. 1, 10.
9. MacKinnon, J. G. (2002), "Bootstrap inference in econometrics", The Canadian Journal of Economics, Vol. 35, No. 4, pp. 615-645.
10. Pampel, F. (2000), "Logistic regression: a primer", Sage University Papers Series on Quantitative Applications in the Social Sciences, 07-132, Sage, Thousand Oaks.

11. Vrigazova, B. (2018), "Nonnegative garrote as a variable selection method in panel data", International Journal of Computer Science and Information Security, vol. 16, No. 1, pp. 95-106.
12. Vrigazova, B., Ivanov, I. (2019), "Optimization of the ANOVA procedure for support vector machines", International Journal of Recent Technology and Engineering, Vol. 8, No. 4, pp. 5160-5165.
13. Vrigazova, B., Ivanov, I. (2020a), "The bootstrap procedure in classification problems", International Journal of Data Mining, Modelling and Management, Vol. 12, in press.
14. Vrigazova, B., Ivanov, I. (2020b), "Tenfold bootstrap procedure for support vector machines", Computer Science, Vol. 21, No. 2, pp. 241-257.
15. Wong, T. T. (2015), "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation", Pattern Recognition, Vol. 48, No. 9, pp. 2839-2846.

About the authors

Borislava is currently a PhD candidate in Data science at Sofia University, Bulgaria. She obtained a master's degree in Statistics, financial econometrics and actuarial studies in 2015, after a bachelor's degree in Economics at the same university. Her research areas include practical applications of machine learning algorithms for prediction, and how their performance can be boosted. Also, applications of big data techniques to small datasets in the field of economics as alternative to traditional econometrics theory. She challenges traditional econometric modelling techniques used 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. Author can be contacted at **vrigazova@uni-sofia.bg.**