data MDPI

Data Descriptor

Machine Learning Classification Workflow and Datasets for Ionospheric VLF Data Exclusion

Filip Arnaut *, Aleksandra Kolarski and Vladimir A. Srećković

Institute of Physics Belgrade, University of Belgrade, Pregrevica 118, 11080 Belgrade, Serbia;
aleksandra.kolarski@ipb.ac.rs (A.K.); vlada@ipb.ac.rs (V.A.S.)
* Correspondence: filip.arnaut@ipb.ac.rs

Abstract: Machine learning (ML) methods are commonly applied in the fields of extraterrestrial physics, space science, and plasma physics. In a prior publication, an ML classification technique, the Random Forest (RF) algorithm, was utilized to automatically identify and categorize erroneous signals, including instrument errors, noisy signals, outlier data points, and the impact of solar flares (SFs) on the ionosphere. This data communication includes the pre-processed dataset used in the aforementioned research, along with a workflow that utilizes the PyCaret library and a post-processing workflow. The code and data serve educational purposes in the interdisciplinary field of ML and ionospheric physics science, as well as being useful to other researchers for diverse objectives.

Dataset: https://zenodo.org/records/10433194 (accessed on 26 December 2023)

Dataset License: CC-BY 4.0

Keywords: machine learning; open data; ionospheric anomaly classification; open-source software

check for updates

Citation: Arnaut, F.; Kolarski, A.; Srećković, V.A. Machine Learning Classification Workflow and Datasets for Ionospheric VLF Data Exclusion. *Data* **2024**, *9*, 17. https://doi.org/10.3390/data9010017

Academic Editor: Giuseppe Ciaburro

Received: 26 December 2023
Revised: 15 January 2024
Accepted: 17 January 2024
Published: 18 January 2024

1. Summary

Numerous machine learning (ML) algorithms and pre-processing techniques have been made possible by rapid advancements in computer science, data science, and data analysis. It can be noted that it takes a lot of time and effort to manually verify, review, and exclude data from an ionospheric very-low-frequency (VLF) investigation during intense occurrences [1,2]. Nevertheless, ML classification methods can be used to automate this job. We evaluated the Random Forest (RF) algorithm [3] in our prior publication [4] with the purpose of automatically classifying erroneous ionospheric VLF amplitude data points during solar flare (SF) investigation/detection. These erroneous data points were categorized as representing SF events, instrumentation errors, or noisy signals. Due to its ease of use and simplicity (few hyperparameters to tune and a reduced likelihood of overfitting the model due to averaging/voting [5] and the law of large numbers [3]), the RF algorithm is considered a first choice for various ML tasks [6,7]. Consequently, it was a suitable selection for the given research purpose. However, as stated in the research paper [4], it is advantageous to extend the original dataset and test additional classification algorithms in order to possibly increase the predictive power of the algorithms.

This data report fulfills two objectives: firstly, it will catalog and provide a link to the data employed in this study, thereby making them accessible to a broader range of researchers, professionals, and others, and secondly, it will include a workflow that integrates the PyCaret library [8], enabling the comparison and testing of fifteen models in total (data and code available at Supplementary material). The chapter methods, i.e., the workflow description, will provide a comprehensive overview of the workflow utilized in conjunction with the data. A synopsis of the pre-processing steps performed during the construction of the original dataset is also available in [4].

2. Data Description

The datasets that were originally obtained and labeled (erroneous values were filtered out) for other research purposes in September and October of 2011 provided a favorable opportunity to evaluate ML classification on this type of data. The original and labeled samples were combined into a single data source, to which additional data (X-ray irradiance, transmitter and receiver data, and local receiver time) were added (Figure 1). The target variable was obtained from the labeled, i.e., filtered, dataset where each instance which was filtered out of the original dataset was annotated as 1 (anomalous data class) and the data that remained were annotated as 0 (normal data class). The feature extraction process was performed by analyzing statistical features of the VLF amplitude and X-ray irradiance signals. The statistical features utilized included rolling window statistics such as mean, standard deviation, and median, with three different window sizes (5, 20, and 180 min). Additionally, lagged signals and other measures such as rate of change, first- and second-order difference, etc., were also employed. Due to the imbalanced nature of the given ML task, random undersampling [9–11] was performed to balance the distribution of the target labels. As a result, the final dataset was generated. To obtain a more comprehensive explanation of the data pre-processing, refer to [4].

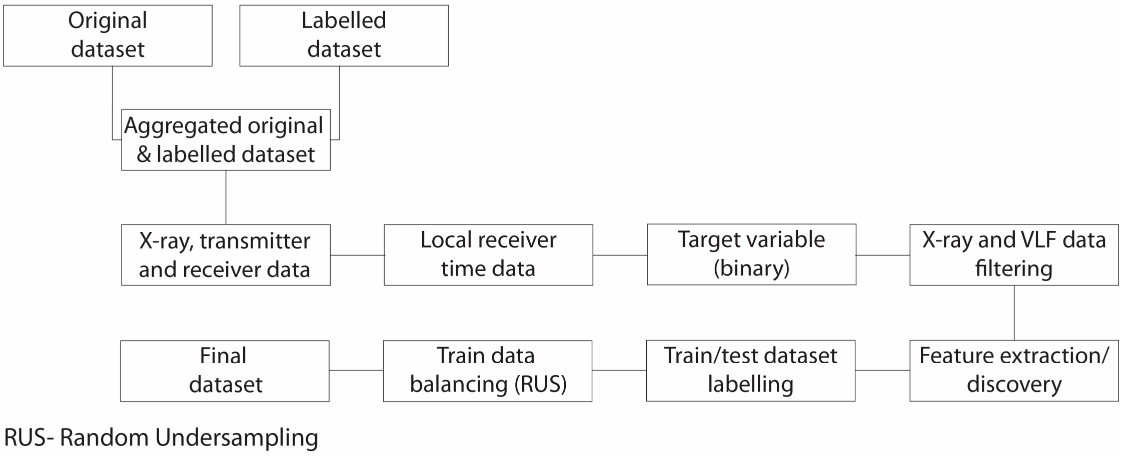
Original dataset
Labelled dataset
Aggregated original & labelled dataset
X-ray, transmitter and receiver data
Local receiver time data
Target variable (binary)
X-ray and VLF data filtering
Final dataset
Train data balancing (RUS)
Train/test dataset labelling
Feature extraction/ discovery
RUS- Random Undersampling

Figure 1. Data pre-processing workflow (modified after [4]).

The dataset was divided into two separate sets: the training dataset and the test dataset. Both of these sets have been pre-processed and are provided as links. The workflow described below can be readily applied to these pre-processed versions of the dataset.

3. Methods (Workflow Description)

Prior to executing the code, the user must specify the input variables, which include the training and test datasets, as well as the visualization range. The initial stage of the workflow involves ML modeling, where the PyCaret library employs the training dataset to conduct a comparison among 15 ML algorithms (Figure 2 and Table 1). After conducting the comparison, the model with the best evaluation metrics and statistics is selected as the overall best model. This model is then used for the hyperparameter tuning process to further optimize the model. The last step involves employing the optimized model to make predictions on the given test dataset. This process generates an output file that includes the predictions made by the most effective and fine-tuned ML algorithm, along with the features and target variables. The post-processing workflow can be summarized in four steps: decoding the data from each transmitter and receiver, separating the individual transmitter–receiver pairs due to the presence of 19 pairs in the test dataset, computing per-class evaluation metrics for each pair, and finally visualizing the true and predicted data labels using the specified input range at the start of the workflow.

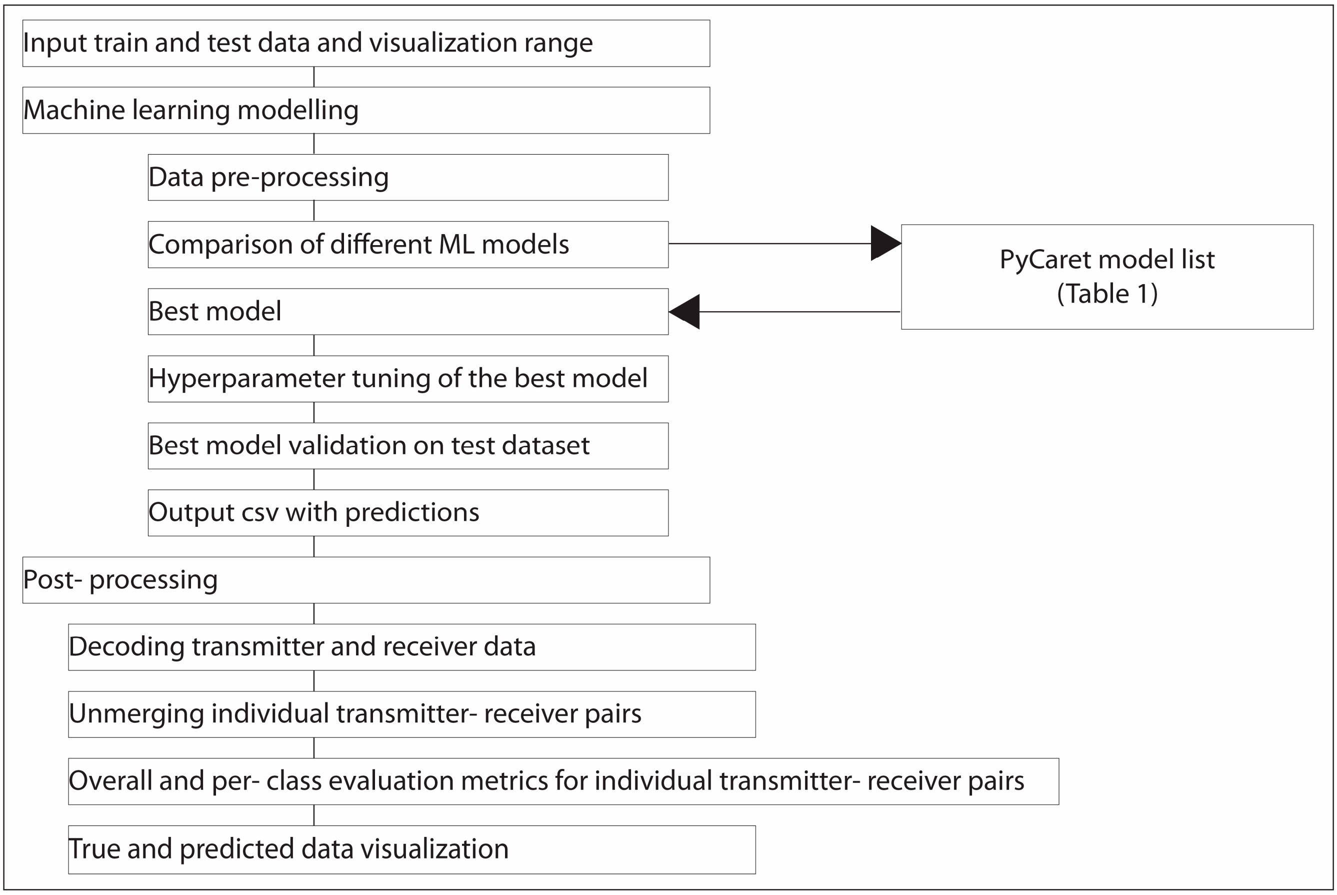
Input train and test data and visualization range
Machine learning modelling
Data pre-processing
Comparison of different ML models
PyCaret model list (Table 1)
Best model
Hyperparameter tuning of the best model
Best model validation on test dataset
Output csv with predictions
Post- processing
Decoding transmitter and receiver data
Unmerging individual transmitter- receiver pairs
Overall and per- class evaluation metrics for individual transmitter- receiver pairs
True and predicted data visualization

Figure 2. Workflow for ML modeling and post-processing.

The evaluation metrics employed for the workflow consist of the confusion matrix, as well as the true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values for each class. Furthermore, the workflow presents accuracy, precision, and F1-score values for each class as well. The overall metrics are also displayed as an output of the workflow. However, due to the highly imbalanced ML task in question, greater emphasis should be given to per-class evaluation metrics. Furthermore, a comprehensive overview of all evaluation metrics can be found in [4]. However, for the purpose of this brief data descriptor, we will provide a short definition of the F1-score. The F1-score is calculated as the harmonic mean of the true positive (TP) rate, also known as recall, and the precision parameter. When evaluating imbalanced ML tasks, the F1-score is typically preferred over accuracy [12,13].

The ML workflow's results are displayed in Figure 3b. All three panels represent the signal obtained from the NAA-Walsenburg transmitter–receiver pair. The signal's duration in Figure 3 spans 600 min, beginning on 19 October 2011 at 14:37 UT and concluding on 20 October 2011 at 1:37 UT. In addition, the workflow also provides the evaluation metrics for each transmitter–receiver pair, specifically the F1-score. In the given example shown in Figure 3b, the anomalous data class has an F1-score of 0.65, while the normal data class has a score of 0.96. This results in a total F1-score of 0.93.

A comparison of the outcomes produced by the workflow integrating the PyCaret library and the transmitter–receiver pair utilized by [4] employing the RF algorithm reveals a distinction. The PyCaret algorithm ascertains that the Extra Trees Classifier (ET) is the optimal overall model for the given task. The comparison of the outputs clearly illustrates that, at least for the instance depicted in Figure 3, the ET classifier is more suitable. However, additional investigation is required to ascertain the specific conditions that require a certain model.

Table 1. Models in the PyCaret library. For more details please see [3,14–23].

#	Model Name	Python Function	More Information at:
1	Logistic Regression	lr	[14]
2	Ridge Classifier	ridge	[14]
3	Linear Discriminant Analysis	lda	[15]
4	Random Forest Classifier	rf	[3]
5	Naive Bayes	nb	[14]
6	Gradient boosting Classifier	gba	[16]
7	Adaboost Classifier	ada	[17]
8	Extra Trees Classifier	et	[18]
9	Quadratic Discriminant Analysis	qda	[19]
10	Light Gradient Boosting Machine	lightgbm	[20]
11	K Neighbors Classifier	knn	[21]
12	Decision Tree Classifier	dt	[22]
13	Extreme Gradient Boosting	xgboost	[23]
14	Dummy Classifier	dummy	[15]
15	SVM Linear Kernel	svm	[14]

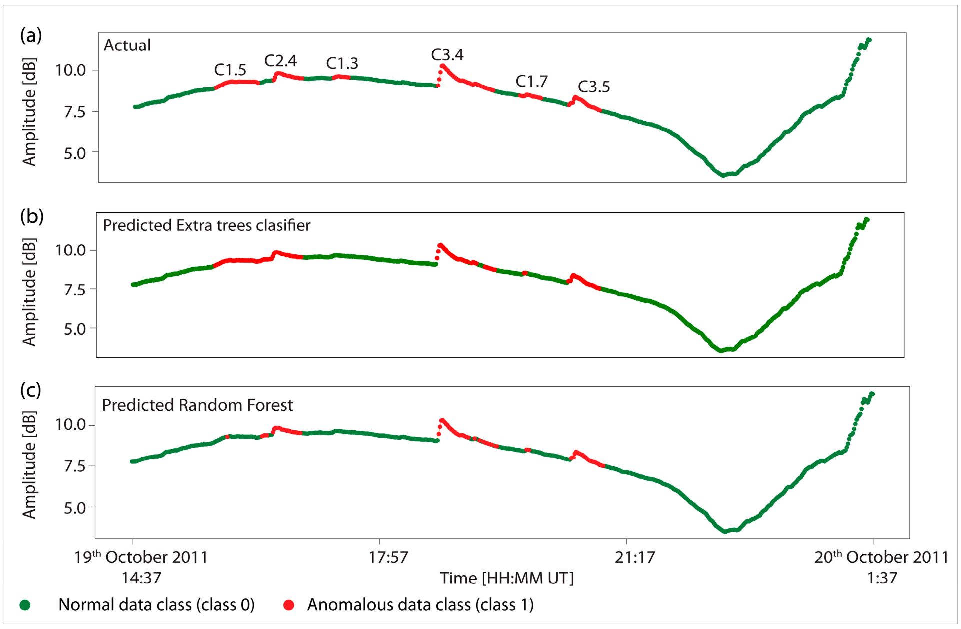

Figure 3. (**a**) Visualization of actual class labels for the NAA_Walsenburg transmitter–receiver pair from 19 October 2011 14:37 to 20 October 2011 01:37, obtained from [4]; (**b**) predictions made by the Extra Trees Classifier from the PyCaret library for the same time period; (**c**) predictions made by the Random Forest Classifier from the same time period, obtained from [4].

In addition to conducting additional research to identify the most suitable model for different scenarios, it is crucial to undertake a comprehensive data acquisition endeavor to further enhance the predictive capabilities of a model. Additional data collection would allow the model to acquire observations from a wider array of events and varying degrees of potential noise levels. For example, data from different time periods within one or a couple of solar cycles can be utilized, etc. Acquiring this level of detailed data requires the collaboration of a larger team of researchers to label and verify the data in a semimanual manner. Following this undertaking, the model has the potential to be significantly enhanced and additional solutions can be devised to cater to a larger research community, for instance, the creation of standalone software with a user-friendly interface that can be utilized by a diverse group of researchers for data VLF pre-processing.

4. User Notes

Researchers may benefit from this database and method/analysis. The code and data serve educational purposes in the interdisciplinary field of ML and ionospheric physics science, as well as being useful to other researchers for diverse objectives. The benefits of these data are the fact that manual data labeling is a laborious undertaking that requires the time of a few researchers. This dataset fulfills the function of being a publicly accessible, annotated dataset that can be employed by researchers to experiment with various research perspectives, thereby conserving both time and labor. The study of space weather (which includes VLF ionospheric research) is highly significant for the broader community (for more information see [24]), as it offers a valuable understanding of the sun–earth connection [25]. This research can enhance our comprehension of space weather phenomena, which in turn will have significant implications for navigation systems, telecommunications, and other related fields.

Supplementary Materials: The following supporting information can be downloaded at https://zenodo.org/records/10433194 (accessed on 26 December 2023).

Author Contributions: Conceptualization, F.A.; formal analysis, F.A., A.K. and V.A.S.; visualization, F.A.; writing—original draft, F.A.; writing—review and editing, V.A.S., A.K. and F.A. All authors contributed equally to this work. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the Institute of Physics Belgrade through a grant by the Ministry of Science, Technological Development and Innovation of the Republic of Serbia. This article/publication is based upon work from COST Action CA22162—A TRANSDISCIPLINARY NETWORK TO BRIDGE CLIMATE SCIENCE AND IMPACTS ON SOCIETY (FutureMed)—supported by COST (European Cooperation in Science and Technology).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All the data and the workflow are available at https://zenodo.org/records/10433194.

Acknowledgments: The WALDO database (https://waldo.world, accessed on 1 October 2023) provides VLF data. It is run collaboratively by the University of Colorado Denver and the Georgia Institute of Technology, utilizing data gathered from Stanford University and these two institutions. It has been made possible by numerous grants from the Department of Defense, NASA, and the NSF.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. McRae, W.M.; Thomson, N.R. VLF Phase and Amplitude: Daytime Ionospheric Parameters. *J. Atmos. Sol.-Terr. Phys.* **2000**, *62*, 609–618. [CrossRef]
2. Šulić, D.M.; Srećković, V.A.; Mihajlov, A.A. A Study of VLF Signals Variations Associated with the Changes of Ionization Level in the D-Region in Consequence of Solar Conditions. *Adv. Space Res.* **2016**, *57*, 1029–1043. [CrossRef]
3. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
4. Arnaut, F.; Kolarski, A.; Srećković, V.A. Random Forest Classification and Ionospheric Response to Solar Flares: Analysis and Validation. *Universe* **2023**, *9*, 436. [CrossRef]
5. Cutler, D.R.; Edwards, T.C.; Beard, K.H.; Cutler, A.; Hess, K.T.; Gibson, J.; Lawler, J.J. Random Forests for Classification in Ecology. *Ecology* **2007**, *88*, 2783–2792. [CrossRef] [PubMed]
6. Hatwell, J.; Gaber, M.M.; Azad, R.M.A. CHIRPS: Explaining Random Forest Classification. *Artif. Intell. Rev.* **2020**, *53*, 5747–5788. [CrossRef]

7. Bartz-Beielstein, T.; Chandrasekaran, S.; Rehbach, F.; Zaefferer, M. Case Study I: Tuning Random Forest (Ranger). In *Hyperparameter Tuning for Machine and Deep Learning with R*; Springer Nature: Berlin/Heidelberg, Germany, 2023; pp. 187–220. [CrossRef]
8. Ali, M. PyCaret: An Open Source, Low-Code Machine Learning Library in Python. PyCaret Version 1.0.0. 2020. Available online: https://www.pycaret.org (accessed on 1 October 2023).
9. Batista, G.E.A.P.A.; Prati, R.C.; Monard, M.C. A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. *ACM SIGKDD Explor. Newsl.* **2004**, *6*, 20–29. [CrossRef]
10. Hasanin, T.; Khoshgoftaar, T. The Effects of Random Undersampling with Simulated Class Imbalance for Big Data. In Proceedings of the 2018 IEEE International Conference on Information Reuse and Integration (IRI), Salt Lake City, UT, USA, 6–9 July 2018. [CrossRef]
11. Saripuddin, M.; Suliman, A.; Syarmila Sameon, S.; Jorgensen, B.N. Random Undersampling on Imbalance Time Series Data for Anomaly Detection. In Proceedings of the 4th International Conference on Machine Learning and Machine Intelligence, Hangzhou, China, 17–19 September 2021. [CrossRef]
12. Hossin, M.; Sulaimani, M.N. A Review on Evaluation Metrics for Data Classification Evaluations. *Int. J. Data Min. Knowl. Manag. Process* **2015**, *5*, 1–11. [CrossRef]
13. Joshi, M.V. On Evaluating Performance of Classifiers for Rare Classes. In Proceedings of the 2002 IEEE International Conference on Data Mining, Maebashi City, Japan, 9–12 December 2002. [CrossRef]
14. Bonaccorso, G. *Machine Learning Algorithms*; Packt Publishing Ltd.: Birmingham, UK, 2017; pp. 1–368.
15. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
16. Friedman, J.H. Greedy Function Approximation: A Gradient Boosting Machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
17. Freund, Y.; Schapire, R.E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
18. Geurts, P.; Ernst, D.; Wehenkel, L. Extremely Randomized Trees. *Mach. Learn.* **2006**, *63*, 3–42. [CrossRef]
19. Friedman, J.H. Regularized Discriminant Analysis. *J. Am. Stat. Assoc.* **1989**, *84*, 165–175. [CrossRef]
20. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems*; The MIT Press: Cambridge, MA, USA, 2017; Volume 30.
21. Cover, T.; Hart, P. Nearest Neighbor Pattern Classification. *IEEE Trans. Inf. Theory.* **1967**, *13*, 21–27. [CrossRef]
22. Quinlan, J.R. Induction of Decision Trees. *Mach. Learn.* **1986**, *1*, 81–106. [CrossRef]
23. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. Available online: https://arxiv.org/abs/1603.02754v3 (accessed on 14 October 2023).
24. Hapgood, M. Societal and Economic Importance of Space Weather. In *Machine Learning Techniques for Space Weather*; Elsevier: Amsterdam, The Netherlands, 2018; pp. 3–26. [CrossRef]
25. Kolarski, A.; Srećković, V.A.; Arnaut, F. Low Intensity Solar Flares' Impact: Numerical Modeling. *Contrib. Astron. Obs. Skaln. Pleso.* **2023**, *53*, 176–187. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.