Super Learner Ensemble for Sound Classification using Spectral Features

Luana Gantert[1], Matteo Sammarco[2], Marcin Detyniecki[2], and Miguel Elias M. Campista[1]
[1]*GTA-PEE/COPPE-DEL/Poli – Universidade Federal do Rio de Janeiro (UFRJ)* – Rio de Janeiro, Brazil
[2]*AXA* – Paris, France
{gantert, miguel}@gta.ufrj.br, {matteo.sammarco, marcin.detyniecki}@axa.com

***Abstract*—Audio samples have emerged as a trend for monitoring and improving decision-making in smart cities, medical applications, and environmental event detections. This paper proposes a Super Learner ensemble application in two scenarios: to distinguish urban from domestic sounds, and detect abnormal samples in industrial machines. The Super Learner combines supervised classifiers to detect abnormal samples or determine a class of an event from spectral features extracted from original sounds. We study the impact on time processing and performance of varying the number of K-folds in the cross-validation step using the Environmental Sound Classification (ESC-50) and Malfunctioning Industrial Machine Investigation and Inspection (MIMII) datasets. The performance evaluation demonstrates that RF is the best classifier in the ESC-50 dataset and SVM in the MIMII dataset. However, the Super Learner reaches AUC and F1-Score values near the best algorithm in the majority of cases analyzed, representing the best tradeoff solution.**

***Index Terms*—Industrial Internet of Things, Machine Learning, Super Learner ensemble, Smart cities**

I. Introduction

Intelligent systems emerge to deal with challenges in different scenarios. In Industry 4.0 (I4.0) digital technologies are adopted to integrate manufacturing systems. The addition of digital technologies permits the management of industrial assets and logistic processes. Then, sustainable solutions can be applied to risk and safety management, product customization, waste reduction, and energy efficiency improvement [1], [2]. In this way, intelligent systems based on machine learning algorithms enable corrective or predictive maintenance to ensure that products and equipment are still in pre-established standard [3]–[5]. Nevertheless, these approaches have as their downside the high processing power and the large volume of data to train machine learning models.

Instead of spectrograms, spectral features can also be used to describe audio sounds enabling other machine learning classifiers to address the same detection task. Since these algorithms are numerous, a potential approach is the combination of them with the stacking ensemble method for performance improvement. Thus, Van der Lan *et al.* propose the stacking ensemble algorithm Super Learner, where multiple base learners are combined, and the optimal weights are selected with a meta-learner [6]. The Super Learner algorithm adopts K-folds cross-validation to train the base learners, store the predictions, and use these outputs as input to the meta-learner.

This paper proposes a Super Learner ensemble to conduct audio classification. The Super Learner combines supervised classifiers to detect events from spectral features extracted from sounds. To our knowledge, this is the first work to implement the method using sound spectral features. We analyze the performance of the Super Learner ensemble by combining five well-known base learners to solve classification tasks: AdaBoost (AB), Naive Bayes (NB), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Random Forest (RF). The idea is to construct a robust classifier to improve the performance or, at least, decrease the need to choose the best algorithm for each scenario. We conduct experiments using the Malfunctioning Industrial Machine Investigation and Inspection (MIMII) dataset [7] and Environmental Sound Classification (ESC-50) dataset [8]. The Synthetic Minority Oversampling technique (SMOTE) [9] is selected to circumvent the imbalanced data in the training set. Hence, the main contributions of this paper are summarized as follows:

- We propose a Super Learner approach using spectral features extracted from the original datasets.
- We adopt the SMOTE to deal with imbalanced training data using the Spectral Features extracted.
- We study the impact of changing the number of K-folds in the cross-validation step to train and fit the Super Learner.

This work is organized as follows: Section II discusses the related work. Section III presents our proposal using spectral features extracted and the base learners selected. Section IV presents the metrics computed, the applied methodology, and experimental results comparing the performance of the Super Learner with the base learners individually. Finally, Section V concludes this paper and draws future directions.

II. Related Work

We divide the related work into two categories. The first one discusses the deployment of multiple models for the classification in I4.0. The second one shows efforts using Super Learners in other scenarios.

A. Multiple models deployment

The MIMII dataset has abnormal and normal audio samples collected from industrial machines. This data encourages proposals for classifying the operational status of industrial equipment by extracting information from sounds. Tama *et al.* [10] extracted spectrograms and applied the convolutional neural networks EfficientNEt to image classification. These weak classifiers are combined in a weighted ensemble

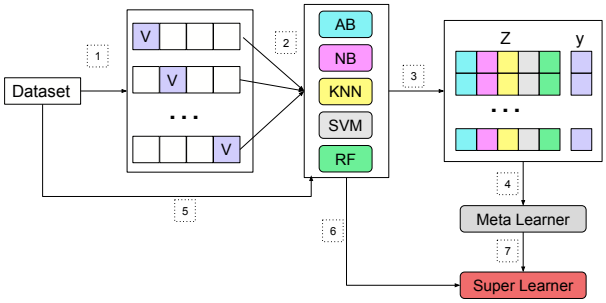


Fig. 1: The Super Learner ensemble method to combine machine learning algorithms, i.e., the base learners.

model to improve the overall performance. Gantert *et al.* [11] extracted spectral features from the original audio samples to save the converted dataset into CSV files avoiding the adoption of deep neural networks. Then, they could implement a binary classification using different supervised machine learning algorithms. The algorithms selected for reactive maintenance were SVM, RF, NB, and MLP. The results could achieve higher values of AUC than the baseline work based on deep autoencoders and Mel-spectrograms. Natesha and Guddeti [12], on the other hand, have selected and analyzed the classification adopting two distinctive spectral features as input. The system was implemented with fog nodes to reduce the response time compared with models running in cloud servers. The algorithms selected in their work were MLP, SVM, RF, AdaBoost, and Logistic Regression. These papers either ensemble similar models or evaluate the best individual performance. We combine different models in a single ensemble using a Super Learner instead.

B. Super Learner deployment

The ensemble methods combine algorithms to construct more powerful models. Hedeya *et al.* [13], for instance, adopt a Super Learner to improve the performance of deep neural networks in vehicle-type classifications from surveillance frames. The convolutional neural networks ResNet 50, Xception, and DenseNet were selected as base learners. Wei *et al.* [14] use a Super Learner to estimate the greenhouse emission in diesel-fueled vehicles. The Super Learner is compared with model systems adopted to predict the gas emission, decreasing the Root Mean Square Error (RMSE) by up to 50%. Hence, the deployment of Super Learners is often limited to scenarios different than audio classification using spectral features.

III. Super Learner Proposal for Audio Classification

The Super Learner ensemble proposed to conduct audio classification. Related work typically addresses audio classification through the individual evaluation of multiple machine learning models. We, instead, combine supervised classifiers using a Super Learner. The idea is to detect abnormal samples from spectral features from sounds of industrial machines and urban sounds from environmental sounds.

The Super Learner algorithm is based on K-fold cross-validation to determine the best combination of algorithms for predictions or classifications. Figure 1 shows the Super Learner functioning. In Step 1, the dataset is split into K blocks, and each block has K parts. One of the K parts is separated for validation while the remaining blocks are used to train the base learners. The base learners are the algorithms selected to construct the Super Learner. We selected the supervised algorithms as follows:

- **AdaBoost (AB):** This ensemble method fits the copies of classifiers, adjusting the weights in incorrect classifications. Then, the next classifier is enabled to solve the most challenging samples. In our proposal, we adopt the decision tree algorithm.
- **Naive Bayes (NB):** This probabilistic classifier is based on Bayes' theorem with a strong independence assumption. We adopted the Gaussian Naive Bayes, i.e., the classifier assumes that the classes follow a Gaussian distribution.
- **Support Vector Machine (SVM):** This classifier is based on data separation in classes using hyperplanes. These hyperplanes are chosen to maximize the distance between the nearest data.
- **K-nearest neighbor (KNN):** This classifier considers the number of the K nearest samples to label a new sample. The class is defined according to the majority class among the K-samples selected.
- **Random Forest (RF):** This classifier is an ensemble method, where multiple decision trees are used, and the class is the one with the majority of votes.

In Step 2, the models are trained and validated until all K parts of the blocks have been reserved as the validation set. Step 3 in Figure 1 builds a matrix with the Z outputs of the base learners. The predictions are stored with the real value y of the samples. In Step 4, the matrix becomes the input of the meta learner. The meta learner is an algorithm trained with Z to find the optimal weights of base learners to achieve the value y. We selected in this work the decision tree as meta learner. In Step 5, the base algorithms are trained with the entire dataset. Step 6 represents the outputs obtained in the previous step. Finally, in Step 7, the Super Learner is built by applying the weights selected by the meta learner. This is the final step for building the Super Learner.

Once the spectrograms can be replaced by the features extracted from the original audio datasets to describe the sounds, image recognition algorithms can also be replaced by algorithms with less computational requirements.

Considering the results in the related work [11], we select the five spectral features presented as follows:

- **Mel-Frequency Cepstral Coefficients (MFCC)**: the MFCCs can phonetically describe the instantaneous amplitude of an oscillating signal. In this way, the coefficients are widely used in speech recognition and are computed as the inverse discrete cosine transform (DCT) of the cepstrum power at each Mel frequency. Thus, the

MFCC is defined as follows:

$$MFCC = \sqrt{\frac{2}{M_{mfcc}}} \sum_{m=1}^{M_{mfcc}} \log(\tilde{X}_m(t)) \cos\left(\frac{c(n-\frac{1}{2})}{M_{mfcc}}\right), \quad (1)$$

where M_{mfcc} is the number of Mel frequency bands (in our case, $M_{mfcc} = 20$), $\tilde{X}_m(t)$ is the signal energy in the m^{th} Mel frequency band, and $c \in [1 - -M_{mfcc}]$ is the index of the cepstrum coefficient.

- **Spectral Centroid (SC)**: the magnitude of the spectrum center of gravity. This feature is associated with the subjective brightness idea of sound intensity, where signals with higher frequencies are perceived as clearer. The SC is the average of frequencies present in the signal weighted by their amplitudes:

$$SC = \frac{\sum_{k=1}^{M} f(k)S(k)}{\sum_{k=1}^{M} S(k)}, \quad (2)$$

where the $S(k)$ is the spectral magnitude at frequency bin k.

- **Spectral Bandwidth (SB)**: spectral bandwidth calculated considering the spectral centroid SC as follows:

$$SB = \sqrt{\sum_{k=1}^{M} S(k)(f(k) - SC)^2}. \quad (3)$$

- **Spectral roll-off (SR)**: the frequency below which 85% of the spectral magnitude is satisfying the following relation:

$$\sum_{k=1}^{SR} S(k) = 0.85 \sum_{k=1}^{M} S(k). \quad (4)$$

- **Zero Crossing Rate (ZCR)**: The ZCR counts the number of times an audio signal waveform crosses zero, and it is directly computed from the time domain as:

$$ZCR = \frac{1}{2N} \sum_{t=0}^{N-2} |sgn(x(t+1)) - sgn(x(t))|, \quad (5)$$

where $sgn(x(t)) = \begin{cases} 1, & \text{if } x(t) \geq 0, \\ -1, & \text{if } x(t) < 0. \end{cases}$

The original sample rate is preserved while the value of each feature is the mean obtained from the original audio.

IV. Performance Evaluation

We use the ESC-50 and MIMII datasets to evaluate the Super Learner approach to classify sound samples. The ESC-50 dataset contains 2000 audio samples with 5 seconds duration of 50 categories. We can group the categories into 5 classes: Animals, Natural soundscapes and water sounds, Human non-speech sounds, Domestic sounds, and Urban sounds. We select the Domestic and Urban sounds to obtain a binary classification and adopt the same evaluation metrics used in the MIMII dataset.

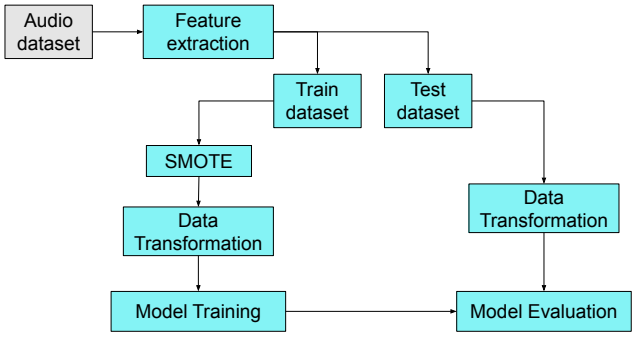


Fig. 2: Description of the system implemented to train and evaluate the Super Learner and the base learners.

The MIMII dataset contains information from four machine types: pumps, fans, slide rails, and valves. For each machine type, samples from four individual machines were collected, resulting in a total of 16 for different equipment. Besides, original audio signals were mixed with industrial noise to compose three levels of Signal to Noise Ratio (SNR): 6 dB, 0 dB, and −6 dB. The dataset consists of 18,019 audio samples per SNR with 10 seconds duration. The proportion of abnormal samples varies between 11.5% to 26.58% per machine type.

We evaluate the binary classification considering imbalanced data in the test set. We use the AUC-ROC and the F1-Score metrics to assess the proposal performance. In this case, we do not use the accuracy metric as this could hide a biased classifier. The AUC-ROC plots the True Positive Rate (TPR) against the False Positive Rate (FPR). The resulting value falls between 0 to 1, with higher values indicating better performance.

$$Precision = \frac{TP}{TP + FP} \quad (6)$$

$$Recall = \frac{TP}{TP + FN} \quad (7)$$

$$F1 - Score = \frac{2 \times precision \times recall}{precision + recall} \quad (8)$$

We also analyze the classifiers using the F1-Score. This metric considers precision and recall in Equations 6 and 7, respectively. The True Positive (TP) is the number of correct classifications of class 1, the True Negative (TN) is the number of correct classifications of class 0, and the False Negative (FN) is the number of incorrect classifications of class 1. Whereas lower precision values indicate more incidence of false positives, lower recall values indicate more incidence of false negatives. The F1-Score is a harmonic mean, aiming to find a balance between precision and recall (Equation 8).

A. Methodology

We conduct experiments using the Librosa library [15] to extract the spectral features from the original sounds, the Scikit-learn to implement and evaluate the algorithms [16], and the ML-Ensemble to build the Super Learner [17]. Figure 2

TABLE I: Hyperparameters values for base learners.

Algorithm	Hyperparameter	Value
AB	n_estimators	50
KNN	n_neighbors	5
SVM	C	1
	kernel	rbf
	gamma	scale
RF	n_estimators	100
	criterion	Gini
	max_features	sqrt
	min_samples_split	2
	min_samples_leaf	1

shows the methodology of this paper. After the spectral features extraction, the dataset is split into train and test sets. We consider the detection of urban sounds from ESC-50 and abnormal sounds from the MIMII dataset as the interest tasks, setting these classes as class 1 in the binary classification.

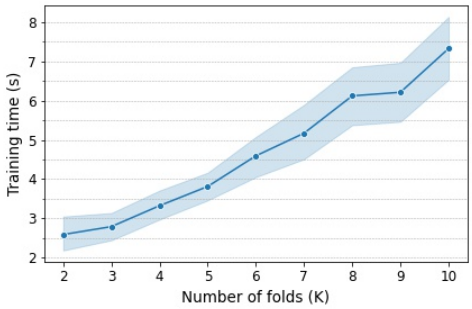


Fig. 3: Time per number of K-folds in cross-validation during Super Learner model training in ESC-50 dataset.

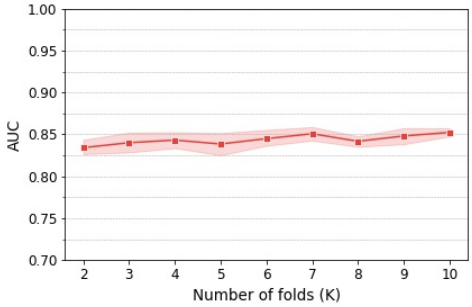
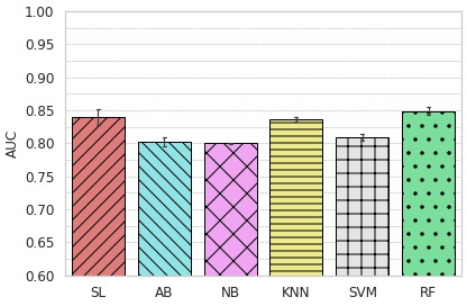
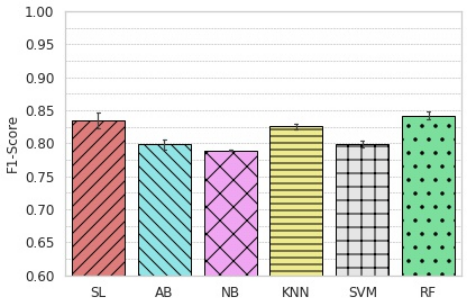


Fig. 4: The AUC in the test set of the Super Learner model *vs.* the number of K-Folds in the training step for the ESC-50 dataset.

We reserved 70% of the dataset as the train set and the remaining data as the test set. To circumvent the imbalance between classes, we use the SMOTE in the train set until both classes have an equal number of samples. This technique randomly selects a sample from the minority class and their N-nearest neighbors. Then, a neighbor is chosen randomly, and a synthetic sample is created between both samples in the feature space. We use N equal to 5 in our experiments. In the Data Transformation step, the standardized function from



(a) AUC.



(b) F1-Score.

Fig. 5: AUC and F1-Score comparing Super Learners and the base learners for ESC-50 dataset.

the Scikit-learn library is applied to the samples to remove the mean and scaling to unit variance.

In the Model Training step, the base learners and the Super Learner are trained, and the optimal weights of the Super Learner are obtained. The test set is used to evaluate the performance of the models by using the selected metrics previously explained.

The hyperparameters used for base learners are in Table I. We adopted the default defined in the Scikit-learn library since hyperparameter tuning is out of the scope of this paper.

B. Results

We run our experiments in Google Colab with Intel(R) Xeon(R) CPU @ 2.20GHz and 12GB of RAM. We first analyze the increase in training time with the number of K

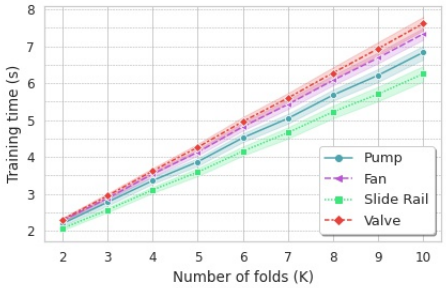
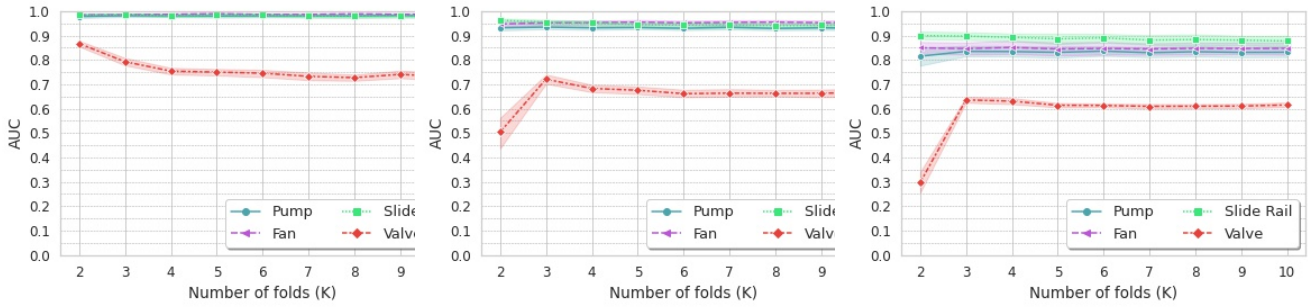


Fig. 6: Time per number of K-folds in cross-validation during Super Learner model training for the MIMII dataset.



(a) SNR = 6 dB. (b) SNR = 0 dB. (c) SNR = −6 dB.

Fig. 7: The AUC in the test set of the Super Learner model *vs.* the number of K-Folds in the training step for the MIMII dataset.

folds to build the Super Learner model. Figures 3 and 6 shows that the increase of time needed to train the model is approximately linear in both datasets. Even though this is an offline approach, reducing this time can impact online systems and systems that run updates from time to time.

C. ESC-50 dataset

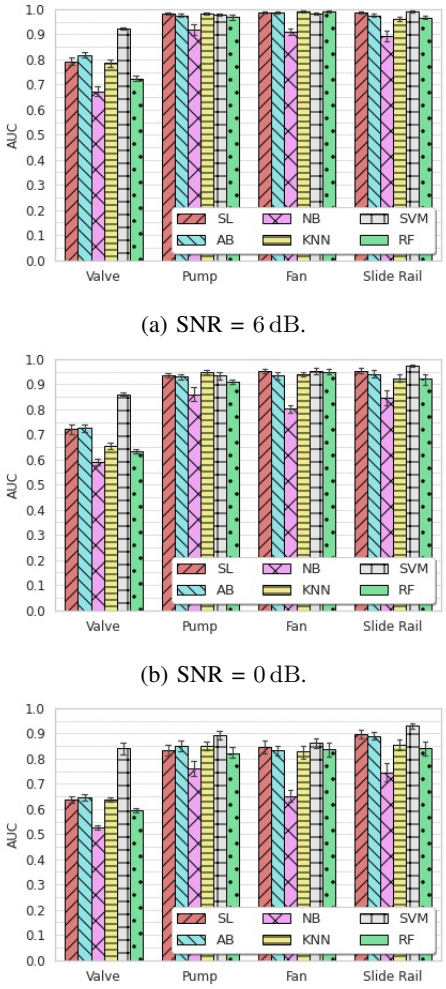
Figure 3 evidences the impact of the K in the training time and the Figure 4 the impact on the AUC in the test set. We consider the AUC improvement when $K = 10$ low relative to the increase in the training time. Thus, we adopt $K = 3$ in next evaluations.

The error bars in the Figure 5 show the 95% confidence interval. In Figure 5a, the AUC obtained by the Super Learner (SL) in the test set is compared with the base learners. The SL reaches a value near 0.85, being higher majority base learners as AB, NB, and SVM. In Figure 5b the F1-Score shows similar result. In both cases, RF is the best choice to classify environmental sounds.

D. MIMII dataset

We analyze the impact on the AUC in the test set as shown in Figure 7 to confirm the choice of $K = 3$ as in the ESC-50 dataset experiments. Since the valves, represented by the red dashed line, are highlighted by the lower value, we prioritized this equipment to choose the value of K. However, in the most challenging SNR, SNR=-6 dB, in Figures 7b and 7c the value obtained with $K = 3$ is higher, being approximately 0.75 and 0.6 simultaneously.

Figure 8 shows the AUC obtained by the Super Learner and the base learners for the types of equipment in the different SNRs. The NB, in this context, is used as a baseline as this is a weak classifier. With SNR = 6 dB, except for the valves, we observe that the SL is near the maximum value for the metric. This is expected since most algorithms obtain similar performance. For valves, while SL reaches 0.8, the SVM is greater than 0.9. Nevertheless, the performance of the ensemble method exceeds the majority of the base learners selected. This behaviour can also be observed for valves and slide rail in SNR = 0 dB in Figure 8b, and valve, fan, and slide rail for SNR = -6 dB in Figure 8c.



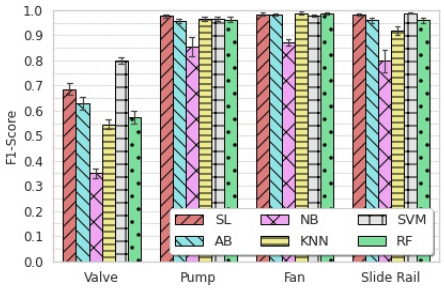
(a) SNR = 6 dB.

(b) SNR = 0 dB.

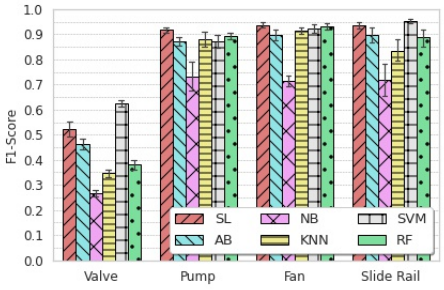
(c) SNR = −6 dB.

Fig. 8: AUC comparing Super Learners and the base learners for MIMII dataset.

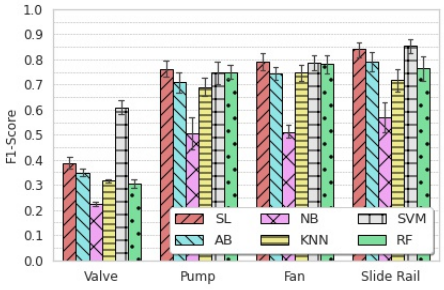
We analyse the F1-Score in Figure 9. The SL is the best algorithm for pumps in all the SNRs in Figures 9a, 9b, and 9c



(a) SNR = 6 dB.



(b) SNR = 0 dB.



(c) SNR = −6 dB.

Fig. 9: F1-Score comparing Super Learners and the base learners for MIMII dataset.

reaching values higher than 0.75 in the lower SNR, and for fan in the last two SNRs. Nevertheless, the SL increases the AUC value obtained in most of the algorithms. Since in some scenarios the accuracy of SVM is much higher than that of other algorithms, SL can increase the correct classifications of the other base learners.

V. Conclusion

In this paper, we propose the use of a Super Learner ensemble method to sound events classification. The performance was compared with the performance of the base learners selected by application in related works. Our analysis shows Super Learner reached values for AUC and F1-Score near the best algorithm in the majority of results analyzed. Thus, the method can be applied in other scenarios without previous comparisons between algorithms candidates.

In future works, we will investigate the impact of tunning the hyperparameters in base learners and change the meta learner.

Acknowledgment

This work was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. It was also supported by CNPq, FAPERJ Grants E-26/211.144/2019 and E-26/202.689/2018, and FAPESP Grant 15/24494-8.

References

[1] S. Wang, Y. Liang, W. Li, and X. Cai, "Big data enabled intelligent immune system for energy efficient manufacturing management," *Journal of cleaner production*, vol. 195, pp. 507–520, 2018.

[2] M. Ghobakhloo, "Industry 4.0, digitization, and opportunities for sustainability," *Journal of cleaner production*, vol. 252, p. 119869, 2020.

[3] G. Aceto, V. Persico, and A. Pescapé, "A survey on information and communication technologies for industry 4.0: State-of-the-art, taxonomies, perspectives, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3467–3501, 2019.

[4] L. Li, K. Ota, and M. Dong, "Deep learning for smart industry: Efficient manufacture inspection system with fog computing," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4665–4673, 2018.

[5] K. Bochie, M. S. Gilbert, L. Gantert, M. S. Barbosa, D. S. Medeiros, and M. E. M. Campista, "A survey on deep learning for challenged networks: Applications and trends," *Journal of Network and Computer Applications*, vol. 194, p. 103213, 2021.

[6] M. J. Van der Laan, E. C. Polley, and A. E. Hubbard, "Super learner," *Statistical applications in genetics and molecular biology*, vol. 6, no. 1, 2007.

[7] H. Purohit, R. Tanabe, T. Ichige, T. Endo, Y. Nikaido, K. Suefusa, and Y. Kawaguchi, "MIMII Dataset: Sound dataset for malfunctioning industrial machine investigation and inspection," in *4th Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE 2019)*, Oct. 2019, pp. 209–213.

[8] K. J. Piczak, "Esc: Dataset for environmental sound classification," in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 1015–1018.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[10] B. A. Tama, M. Vania, I. Kim, and S. Lim, "An efficientnet-based weighted ensemble model for industrial machine malfunction detection using acoustic signals," *IEEE Access*, vol. 10, pp. 34 625–34 636, 2022.

[11] L. Gantert, M. Sammarco, M. Detyniecki, and M. E. M. Campista, "A supervised approach for corrective maintenance using spectral features from industrial sounds," in *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*. IEEE, 2021, pp. 723–728.

[12] B. Natesha and R. M. R. Guddeti, "Fog-based intelligent machine malfunction monitoring system for industry 4.0," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 7923–7932, 2021.

[13] M. A. Hedeya, A. H. Eid, and R. F. Abdel-Kader, "A super-learner ensemble of deep networks for vehicle-type classification," *IEEE Access*, vol. 8, pp. 98 266–98 280, 2020.

[14] N. Wei, Q. Zhang, Y. Zhang, J. Jin, J. Chang, Z. Yang, C. Ma, Z. Jia, C. Ren, L. Wu *et al.*, "Super-learner model realizes the transient prediction of co2 and nox of diesel trucks: Model development, evaluation and interpretation," *Environment International*, vol. 158, p. 106977, 2022.

[15] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, vol. 8. Citeseer, 2015, pp. 18–25.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[17] S. Flennerhag, "mlens documentation," 2017, accessed at https://buildmedia.readthedocs.org/media/pdf/mlens/dev/mlens.pdf.