Hyperspectral Imaging: A comparative study of supervised learning algorithms

Mailson R. M. Guimarães[1], **Anne M. P. Canuto** [1], **Márcio Eduardo Kreutz** [1]

[1]Departamento de Informática e Matemática Aplicada
Universidade Federal do Rio Grande do Norte (UFRN) - Natal, RN - Brazil

mailson.rodrigues.095@ufrn.edu.br, anne.canuto@ufrn.br,

kreutz@dimap.ufrn.br

Abstract. *This study focuses on the hyperspectral image from the University of Pavia, performing various manipulations to derive new datasets and observe their impact on the classification results. The aim is to automate the pixel classification process using machine learning algorithms with different training and testing splits. Additionally, ensemble classifiers were implemented to improve accuracy. The results show that the Multilayer Perceptron (MLP) achieved the highest accuracy among the implemented methods, surpassing 85% and providing similar results to the ensemble classifiers. The original dataset (untouched) and the dataset reduced to 20 principal components using Principal Component Analysis (PCA) yielded the best results. It is worth noting that considering unlabeled pixels limited the accuracy of the implemented algorithms.*

Resumo. *O presente trabalho mostra um estudo em torno de uma base de dados de imagens hiperespectrais, realizando-se manipulações nesta para a derivação de novas bases de dados para que seja possível observar o comportamento dessas manipulações nos resultados. Além disso, é realizada a automatização do processo de classificação de pixels através da aplicação de algoritmos de aprendizado com diferentes razões de divisão entre as bases de teste e treinamento. Também foram implementados comitês de classificadores como forma de melhorar a acurácia. Os resultados mostram que a Multilayer Perceptron (MLP) é o método com maior acurácia dentre os implementados, atigindo valores superiores a 85% e fornecendo resultados semelhantes aos dos comitês. Dentre as bases analisadas, as que mostraram melhor resultado foi a base original (inalterada) e a que teve seus atributos reduzidos para 20 componentes principais através do algoritmo Principal Component Analysis (PCA). Além disso, devido à consideração dos pixels sem classe definida, obsevou-se uma limitação de acurácia para os algoritmos implementados.*

1. Introduction

Remote sensing is an essential field that involves acquiring data from a specific area, object, or phenomenon without direct physical contact [Lillesand et al. 2015]. In other words, the acquisition is done remotely through sensors. Within this context, the use of multispectral and hyperspectral imagery (HSI) stands out (Figure 1). The HSI is typically acquired using a spectrometer, which measures the amount of light emitted, reflected,

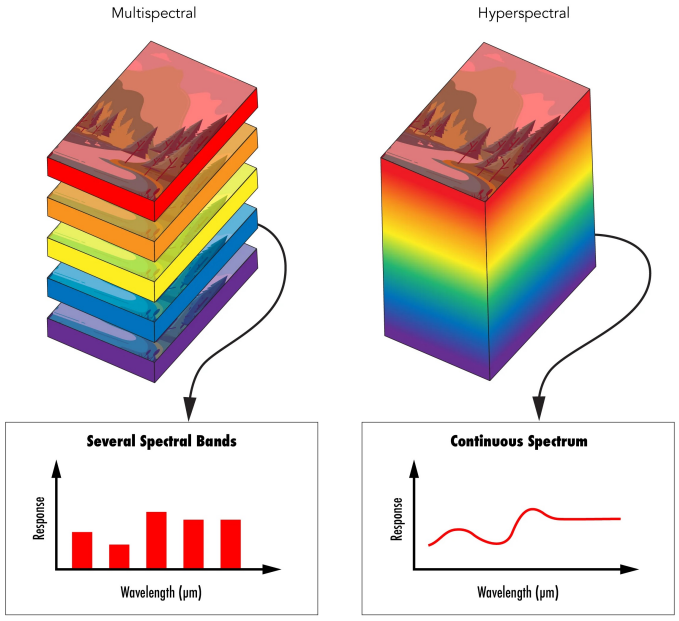

Figure 1. Multispectral and hyperspectral imaging (Adapted of [Edmund Optics 2023])

or transmitted by objects. These sensors typically operate between 400 and 2500nm, covering the visible light spectrum to the near-infrared range [Paoletti et al. 2019].

Multispectral images have a discrete spectrum, meaning that a scene is represented through values of limited wavelengths. In contrast, hyperspectral images have a ”continuous” spectrum, as they contain significantly more wavelengths than multispectral images, although still finite.

A hyperspectral image forms a hyperspectral data cube (Figure 1), where dimensions x and y correspond to the image's pixel resolution, and the depth λ corresponds to the acquired wavelengths. In this scope, there is a need for automating the process of classifying pixels in the image based on the spectrum that makes up each pixel. For example, in a given HSI, pixels may represent water, shadows, asphalt, and other materials, making manual classification time-consuming, even in low-resolution images.

Automating this process is treating the image as a dataset and applying Machine Learning (ML) techniques. ML is an area that gained increasing prominence and has a wide range of applications. In the specific problem, the ML algorithms should be able to classify each pixel appropriately based on its wavelength spectrum.

The work is organized in such a way that Section 2 presents related and current works regarding the studied theme, while Section 3 introduces the methodology, including the target dataset and preprocessing. Section 4 presents the results of the supervised algorithms, ensemble techniques, confusion matrix analysis, and statistical tests. Finally, Section 5 presents the conclusions of this work.

2. State of the art

The state-of-the-art works cited in this study use the same dataset under analysis for comparison purposes. Therefore, the accuracy results displayed are from the HSI dataset from

the University of Pavia.

Among the found references, [Sildir et al. 2020] can be mentioned, which introduces a technique applied to neural networks for large datasets to find an optimal architecture for the network, including removing connections between neurons. The maximum accuracy obtained for the Pavia dataset was 99.53% for a fully connected neural network and 92.10% for a network with optimal structure.

In [Akbari 2020], the hyperspectral image undergoes a series of transformations (Wavelet Transform, Gabor Transform, Mean, Entropy, and Contrast) to extract spatial and texture features. Such features are combined with the image spectrum for classification purposes using Multilayer Perceptron (MLP) and Support Vector Machine (SVM) algorithms. In the end, the classification is combined with the result of a hierarchical segmentation algorithm to the final classification. [Akbari 2020] achieved an overall accuracy of 93.3% with their method.

The author [Chen et al. 2021] presents the classification technique for HSIs using Kernel Extreme Learning Machine (KELM), called PLG-KELM. Similar to [Akbari 2020], spatial and texture feature extraction is performed on the image to construct the classification model. The total accuracy achieved by the technique for the target dataset was 99.05% with a training set of 10%.

In [Tan et al. 2019], a parallel Gaussian-Bernoulli Restricted Boltzmann Machine (GBRBM) is introduced, where the HSI undergoes multiple GBRBM layers for feature extraction. These features are then fed into a Logistic Regression model for pixel classification. For the Pavia dataset, the authors achieved a maximum accuracy of 92.20% with the presented method.

Este paper difere dos mencionados acima por considerar pixels sem rótulo na criação dos modelos matemáticoa dw

This paper differs from the ones mentioned above by considering unlabeled pixels in creating mathematical classification models. When applying the model trained disregarding the unlabeled pixels in a real context, after acquiring a new hyperspectral image, all pixels would only be classified according to the labeled pixels. This is invalid since the pixels of interest comprise a small part of the image. Datasets with pre-sorted pixels often require a human expert [Paoletti et al. 2019].

3. Experimental Methodology

This Section presents the characteristics and class distribution of the target dataset and the preprocessing that branches the initial dataset into three reduced ones. The primary tool used for carrying out all the processes presented in this work is the Python programming language, machine learning, and data manipulation support libraries.

3.1. Dataset

The target dataset consists of a scene acquired at the University of Pavia, Italy, using the Reflective Optics System Imaging Spectrometer (ROSIS) sensor. The hyperspectral image has dimensions of 610x340 pixels and 103 spectral bands. Each pixel value in the hyperspectral cube ranges from 0 to 8000. Since the objective is the classification of each pixel, the image is then converted into a table where each row corresponds to a

pixel (instance), and the columns correspond to the values in each of the 103 bands of the image.

Therefore, there is initially a dataset with 207,400 instances, 103 attributes, and a class attribute corresponding to the pixel classification. However, an initial instance reduction was performed by applying a scaling factor of 70% to the image, resulting in a size of 427x238 pixels. This process reduced the total instances to 101,626, representing a reduction of approximately 50% in the dataset size.

The dataset consists of 9 classes, plus one unlabeled class, which corresponds to most of the image and contains non-relevant information for classification. The distribution of classes in the initial image and after reduction can be seen in Table 1. Indeed, the consideration of unlabeled pixels makes the dataset even more imbalanced.

Table 1. Dataset class distribution

Classes	Samples	
	Initial	**Reduced**
Unlabeled	166,624	77,680
Asphalt	6,631	5,187
Meadows	18,649	10,252
Gravel	2,099	1,695
Trees	3,064	1,178
Painted metal sheets	1,345	810
Bare soil	5,029	2,809
Bitumen	1,330	884
Self-Blocking Bricks	3,682	955
Shadows	947	176

3.2. Preprocessing

The dataset also underwent a preprocessing step, branched into three separate datasets. For the first dataset, a reduction of instances from the majority class (unlabeled) was performed, where the number of instances was reduced by 90% through randomly selected samples. This step resulted in a reduction from 77,680 instances to 7,768 instances.

For the second dataset, it was assumed that neighboring bands would have similar results in data acquisition, implying a particular dependence relationship among the attributes. A correlation matrix was computed among the 103 attributes to verify this assumption, resulting in Figure 2. It can be observed that there is a high correlation across most of the dataset, except for a dark region starting from band 68, which marks the transition between the visible light spectrum and the near-infrared region. In this region, there is no correlation with the previous bands.

Thus, regions of high correlation between attributes were defined, and the average value among the attributes within each region was calculated to create a single attribute in the new dataset. For example, between bands 1 and 37, there is a high correlation, so they were summarized into a single attribute. Band 0 and the range between 68 and 76 were kept unchanged due to a weak correlation with other bands. Overall, the dataset was reduced to only 14 attributes.

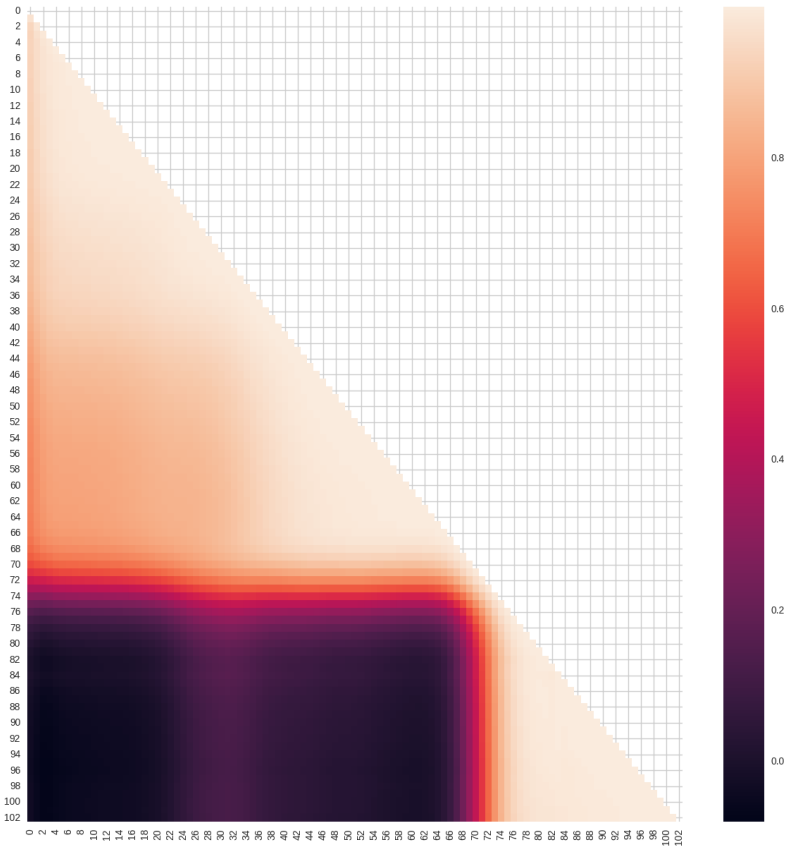

Figure 2. Correlation matrix among attributes of the dataset

Finally, the third dataset was obtained by applying the Principal Componente Anslysis (PCA) algorithm, reducing the number of attributes to 20. This value was chosen based on previous tests with ML algorithms, where an improvement was observed in algorithms like Naïve Bayes. At the same time, there was no considerable degradation in performance for other algorithms. Table 2 shows the number of instances and attributes for each dataset studied in this work.

Table 2. Number of instances and attributes in the datasets

	Instances	**Attributes**
Original dataset	101,626	103
Reduced dataset 1	41,626	103
Reduced dataset 2	101,626	13
Reduced dataset 3	101,626	20

4. Results and discussion

This section presents the accuracy results obtained from the application of supervised algorithms: k-NN, Decision Tree (DT), Naïve Bayes (NB), and Multilayer Perceptron (MLP), as well as a comparison among them. Results related to classifier ensembles, such as Bagging, Boosting, Random Forest (RF), and Stacking, will also be presented. Some results will be analyzed using confusion matrices, and statistical tests will be performed to compare the results.

4.1. Supervised learning

In addition to the algorithms mentioned above and the four datasets under study, different splits between training and test sets were used: 70/30, 80/20, and 90/10. Cross-validation (CV) with 10-fold was also employed.

Exhaustive tests were performed to find the parameters that resulted in the highest accuracy of the methods (except NB). For k-NN, it was found that a value of k=13 reached a tipping point where accuracy started to decline. Analogously to k-NN, the DT was tested with different values of the maximum depth parameter, and the best precision was obtained with 7. For the MLP, all parameters were kept fixed, and the number of neurons in the hidden layer, where a quantity of 72, was obtained for greater accuracy. This parameter was then fixed, and the number of iterations varied between 100, 1000, and 5000, with 1000 iterations being the best result. With the two parameters previously defined, the initial learning rate varied between the values 0.1, 0.01, and 0.001, the latter being the best result.

Table 3 shows the average accuracy results for each method and dataset. The values were calculated across all possible splits between training and test sets.

Table 3. Average accuracy for each dataset and classifiers

Dataset	k-NN	Árvore de decisão	Naïve Bayes	MLP
Original dataset	79.35	77.06	18.21	84.54
Reduced dataset 1	62.20	53.43	39.46	69.57
Reduced dataset 2	76.47	76.95	24.42	81.41
Reduced dataset 3	76.55	78.05	60.86	84.79
Average	73.64	71.37	35.74	80.07

The original dataset showed an accuracy of around 80% for the analyzed methods, except for Naïve Bayes, indicating that, as mentioned earlier, there is dependence among the attributes in the dataset. On the other hand, the first reduced dataset showed the worst results overall, except for NB, which doubled its accuracy compared to the original dataset. This conclusion suggests that the removal of unlabeled samples had a positive impact on the Naïve Bayes algorithm.

The second reduced dataset, obtained through the correlation matrix, showed a deterioration of results compared to the original dataset, with only NB showing improvement. However, its performance remains low. The third dataset exhibited the best result, particularly for Naïve, confirming that using the principal components of the attributes partially eliminated the dependence among them.

The MLP method achieved the best results, with an accuracy of 84.54% and 84.79% on the original and the third reduced datasets, respectively. Regarding the datasets, the third also showed the best result due to the significant improvement in the performance of Naïve Bayes.

4.2. Confusion matrix analysis

In addition to accuracy analysis, a confusion matrix was also calculated. For this purpose, the MLP classifier from Section 4.1 with a 70/30 split was chosen for both the original

and reduced dataset 1 (reduction of instances). Figure 3 shows the confusion matrix for the test split of the original dataset. This split has 30,488 samples, and the class number corresponds to the rows of Table 1. It is noticeable that accuracy analysis masks the proper classification behavior of each class. Despite an accuracy of 83.68%, many instances are misclassified as belonging to the unlabeled class. However, due to the imbalanced nature of the dataset, the overall accuracy remains high. The misclassification error occurs due to the diverse spectra range the unlabeled class encompasses. Some spectra may resemble those of other classes leading to misclassifications by the MLP.

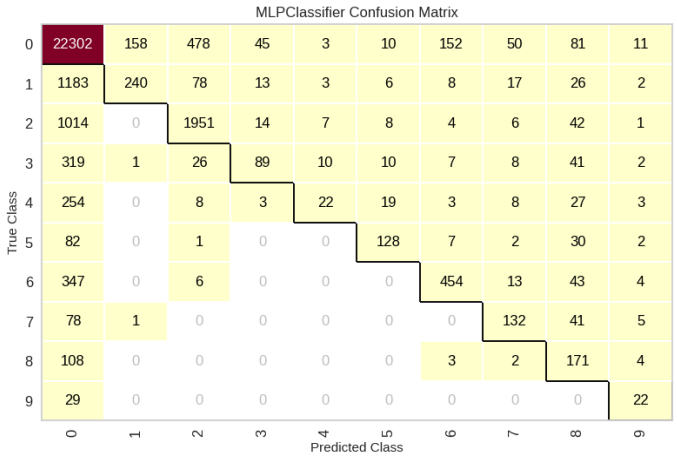

Figure 3. Confusion matrix for the test split of the original dataset

Observing the columns of Figure 3, it is possible to calculate the precision of the classification for each class individually, indicating how many of the predicted pixels of a particular class were correct. Due to the number of samples, the majority class exhibited the highest precision (83.72%), while class 8 had the lowest precision (34.06%).

Concerning the dataset with reduced instances, the confusion matrix (Figure 4) shows that even with the reduction of instances, there are still misclassifications due to class 0 (unlabeled). The remaining classes showed increased precision compared to the original dataset, except for classes 0, 4, and 9. Additionally, by analyzing the rows of the table, it is possible to calculate the recall considering the true positives and false negatives, indicating how many of the pixels in a class were correctly classified. The reduction of instances resulted in an increase in recall for all classes except unlabeled.

In summary, the reduction of instances in the results presented in Figures 3 and 4 maintained approximately the same average accuracy (0.60), while the average recall increased from 0.46 to 0.64.

4.3. Ensemble learning

Ensemble classifiers were also implemented to try to improve the results obtained from the supervised methods. These ensembles were applied to the original and PCA-reduced datasets, as they achieved better results in Section 4.1. The training and test set split was also fixed to reduce the analysis space. The chosen split was 70/30, as it showed the best accuracy result, which occurred on the original dataset with the MLP classifier, reaching 85,50%.

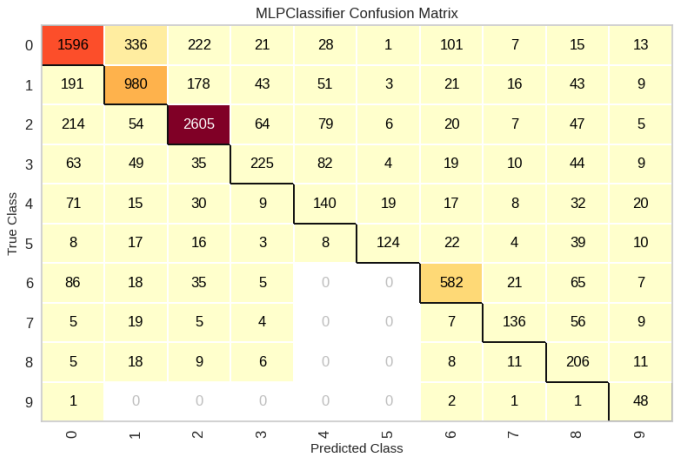

Figure 4. Confusion matrix for the test split of the reduced dataset 1

The Bagging ensemble method used the same classifiers as the 4.1. Each ensemble comprised 10 and 20 classifiers of each type, and the *max_feature*, which represents the number of features sampled from the train set to train each classifier, was also varied. Table 4 shows the accuracy results obtained for the original dataset with the given split, using four possible values of *max_feature*: 1.0, 0.8, 0.5, and 0.3.

Table 4. Acurracy results of Bagging for the original dataset

max_feature=1.0			
Classifier	**10**	**20**	**Mean**
AD	81.74	82.62	82.18
k-NN	81.22	81.37	81.30
NB	18.12	18.34	18.23
MLP	84.42	84.29	84.36
Mean	66.38	66.66	

(a)

max_feature=0.8			
Classifier	**10**	**20**	**Mean**
AD	82.12	82.80	82.46
k-NN	81.48	81.66	81.57
NB	18.47	18.51	18.49
MLP	84.25	84.43	84.34
Mean	66.58	66.85	

(b)

max_feature=0.5			
Classifier	**10**	**20**	**Mean**
AD	81.95	82.51	82.23
k-NN	81.74	81.82	81.78
NB	19.00	18.86	18.93
MLP	83.95	84.18	84.07
Mean	66.66	66.84	

(c)

max_feature=0.3			
Classifier	**10**	**20**	**Mean**
AD	82.10	82.69	82.40
k-NN	81.72	82.02	81.87
NB	20.14	19.85	20.00
MLP	83.75	83.54	83.65
Mean	66.93	67.03	

(d)

It can be observed that increasing the number of estimators led to an increase in accuracy, except for MLP in (a) and (d) and NB in (c). However, the increase was not significant, as indicated by the average across the rows of the tables, where the maximum increase was 0.27% in (a).

The process was repeated for the reduced dataset 3; the results can be seen in Table 5. It is possible to observe a considerable improvement in NB, as seen in Section 4.1. Additionally, compared to the previous table, it can be noted that for the values of

0.5 and 0.3 of *max_feature*, the reduced dataset showed lower accuracy results. For the other values, the accuracy was similar. The best average accuracy result occurred in the reduced dataset with the Bagging of MLPs, yielding a value of 85.02%.

Table 5. Acurracy results of Bagging for the reduced dataset 3

max_feature=1.0			
Classifier	**10**	**20**	**Mean**
AD	82.31	83.00	82.66
k-NN	80.56	80.61	80.59
NB	61.45	60.85	61.15
MLP	84.91	85.13	85.02
Mean	77.31	77.40	

(a)

max_feature=0.8			
Classifier	**10**	**20**	**Mean**
AD	82.09	82.31	82.20
k-NN	80.58	80.62	80.60
NB	67.46	68.45	67.96
MLP	84.72	84.51	84.62
Mean	78.71	78.97	

(b)

max_feature=0.5			
Classifier	**10**	**20**	**Mean**
AD	78.90	80.10	79.50
k-NN	79.20	78.95	79.08
NB	76.47	74.93	75.70
MLP	80.86	81.07	80.97
Mean	78.86	78.76	

(c)

max_feature=0.3			
Classifier	**10**	**20**	**Mean**
AD	77.64	77.27	77.46
k-NN	77.66	77.24	77.45
NB	76.39	76.65	76.52
MLP	77.43	77.08	77.26
Mean	77.28	77.06	

(d)

For Boosting, only the classifiers AD and NB were used, varying the number of classifiers between 10 and 20, similar to Bagging. The average results are shown in Table 6, where a reduction in the accuracy of the decision tree compared to the Bagging results can be observed. Increasing the number of classifiers did not impact the accuracy of AD, while for NB, there was an increase. However, the accuracy of NB did not surpass the best result obtained in the previous ensemble method.

Table 6. Accuracy results of Boosting for (a) the original dataset and (b) reduced dataset 3

Original dataset 70/30			
Classifier	**10**	**20**	**Mean**
AD	74.35	74.19	74.27
NB	56.87	62.59	59.73
Mean	65.61	68.39	

(a)

Reduced dataset 3 70/30			
Classifier	**10**	**20**	**Mean**
AD	74.59	74.52	74.56
NB	59.32	73.19	66.26
Mean	66.96	73.86	

(b)

Random Forest was used with variations in the metric: Gini, Entropy, and Log-loss, as well as the maximum depth of the tree set to 10 and 100. Table 7 shows the

results, where it can be observed that increasing the maximum depth of the tree leads to an increase in accuracy for both the original and reduced datasets. On the other hand, varying the metric did not result in a significant variation in accuracy. The results of Random Forest were comparable to or even surpassed those of the two ensemble methods analyzed previously, except for MLP in Bagging.

Table 7. Accuracy results of Random Forest for (a) the original dataset and (b) reduced dataset 3

Original dataset 70/30			
Metrics	**10**	**100**	**Mean**
Gini	81.89	83.37	82.63
Entropy	81.73	83.45	82.59
Log-loss	81.98	83.40	82.69
Mean	81.87	83.41	

(a)

Reduced dataset 3 70/30			
Metrics	**10**	**100**	**Mean**
Gini	81.78	82.88	82.33
Entropy	81.99	83.34	82.67
Log-loss	82.13	83.52	82.83
Mean	81.97	83.25	

(b)

Finally, Stacking was implemented, with the analysis conducted using 10 and 20 classifiers, where half were MLPs, and the other half were k-NNs. According to Table 8, the results obtained slightly outperformed the results obtained by MLP in Tables 4 and 5, making it the best ensemble among those applied.

Table 8. Accuracy results of Stacking for (a) the original dataset and (b) reduced dataset 3

Original dataset 70/30	
Nº of Classifiers	**Mean Accuracy**
10	84.84
20	84.64

(a)

Reduced dataset 3 70/30	
Nº of Classifiers	**Mean Accuracy**
10	85.13
20	85.24

(b)

4.4. Statistical tests

Another way to compare the obtained results is through statistical tests. For the first test, the division methods of the datasets were analyzed. For each possible division, 36 observations were made (total accuracy results for each division). Then the Friedman Test

was then performed with a 95% confidence level, resulting in a p-value of 3.36×10^{-7}. The null hypothesis is rejected since this value is less than 0.05, indicating statistical differences among the samples (divisions). In order to perform the pairwise comparison, Wilcoxon Tests were performed, forming the matrix shown in Figure 5. The matrix's main diagonal should be ignored in this analysis, and the light colors indicate no statistical similarity between the pairs. Therefore, it can be concluded that there is only statistical similarity between the 90/10 division and the 10-fold division, with a p-value of 0.94.

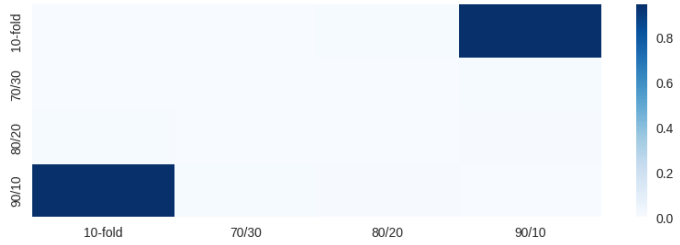

Figure 5. Heatmap graph comparing the divisions of the datasets using the Wilcoxon Test

The comparison among the accuracy results of the supervised algorithms was also performed using the Friedman Test, comparing all four datasets with all four divisions, considering the best parameters. This comparison resulted in 16 observations for each classifier. With a 95% confidence level, the obtained p-value was 3.69×10^{-10}. The heatmap in Figure 6 supports this result, as, despite the dark color between k-NN and AD, the p-value is in the order of magnitude of 0.002, indicating a statistically significant difference between the classifiers.

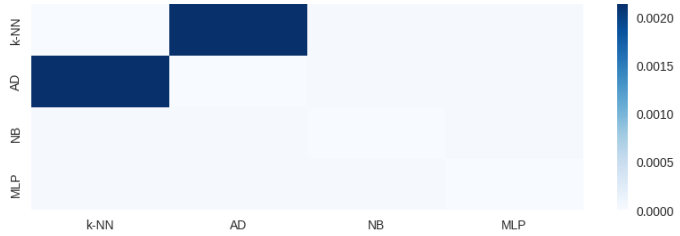

Figure 6. Heatmap graph comparing the classifiers accuracy using the Wilcoxon Test

Finally, for comparing the ensembles to Bagging, the observations considered were the accuracies of MLP with max_feature=1.0 for both the 10 and 20 classifiers and all four bases, resulting in 8 observations. Similarly, for Boosting, the results of the Decision Tree were considered, and for Random Forest, the entropy metric was used. For Stacking, all results were included. The same tests were applied with the same confidence level. The obtained p-value was 5.93×10^{-5}, and the heatmap in Figure 7 shows statistical similarity only between Stacking and Bagging, with a p-value of 0.55.

5. Final considerations

In the context of hyperspectral images, the classification process is of paramount importance for accelerating result generation. The present work analyzed the HSI dataset from the University of Pavia, applying transformations and machine learning algorithms.

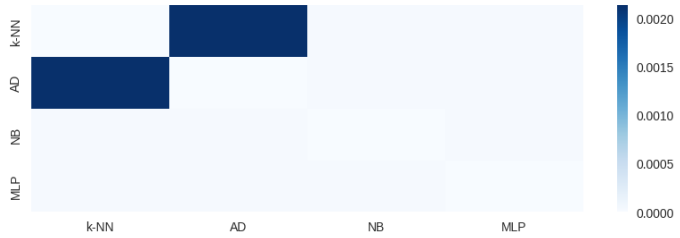

Figure 7. Heatmap graph comparing the ensembles accuracy using the Wilcoxon Test

Reducing the number of instances in the first reduced dataset resulted in lower accuracy, indicating that there were better choices than randomly removing samples from the majority class (unlabeled). However, the confusion matrix analysis showed an improvement in recall compared to the original dataset. The second reduced dataset, obtained by reducing the number of attributes described in Section 3.2, showed promising results. However, the reduced dataset using the PCA algorithm yielded even better results, similar to the original dataset.

In terms of supervised methods, the MLP achieved the best results with accuracies ranging from 84% to 85%. Furthermore, the Friedman test indicated that the supervised classifiers do not exhibit statistical similarity. The use of ensembles did not lead to a significant increase in accuracy compared to individual classifiers. Additionally, the statistical test conducted among the ensembles showed that only Stacking and Bagging exhibited statistical similarity.

The limitation in achieving maximum accuracy with supervised techniques and ensembles and the inability to perform clustering can be attributed to the consideration of unlabeled pixels. Previous works discussed in Section 2 do not explicitly consider background pixels, resulting in a wide variety of scattered samples in the feature space that overlap with the classes of interest. The limitation of the maximum accuracy achieved by supervised techniques and ensembles can be attributed to the consideration of unlabeled pixels, which are not considered in the works discussed in Section 2. This results in a wide variety of samples scattered in the instance space, which can overlap with the target classes of interest.

For future work, one approach could simplify the classification problem to a binary problem between the unlabeled pixels and the pixels of interest. A model capable of accurately differentiating between these two classes can be developed by doing so. Subsequently, one can apply the models mentioned in Section 2, or even develop a new model, to classify the pixels correctly. Additionally, studying the confusion matrices can provide insights into the performance of the classifiers for each class. Additionally, it would be beneficial to simplify the classification problem by treating it as a binary classification between unlabeled pixels and the other classes. This approach could yield better accuracy in distinguishing between the unlabeled pixels and the target classes. Additionally, one could apply the models mentioned in Section 2 or even develop new models specifically designed to classify the target pixels accurately.

References

Akbari, D. (2020). A novel method for spectral-spatial classification of hyperspectral images with a high spatial resolution. *Arabian Journal of Geosciences*, 13(23).

Chen, H., Miao, F., Chen, Y., Xiong, Y., and Chen, T. (2021). A hyperspectral image classification method using multifeature vectors and optimized kelm. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:2781–2795.

Edmund Optics (c2023). Hyperspectral and multispectral imaging. `https://www.edmundoptics.com/knowledge-center/application-notes/imaging/hyperspectral-and-multispectral-imaging/`. Accessed: 2023-06-02.

Lillesand, T., Kiefer, R., and Chipman, J. (2015). *Remote Sensing and Image Interpretation, 7th Edition*. Wiley.

Paoletti, M., Haut, J., Plaza, J., and Plaza, A. (2019). Deep learning classifiers for hyperspectral imaging: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 158:279–317.

Sildir, H., Aydin, E., and Kavzoglu, T. (2020). Design of feedforward neural networks in the classification of hyperspectral imagery using superstructural optimization. *Remote Sensing*, 12(6).

Tan, K., Wu, F., Du, Q., Du, P., and Chen, Y. (2019). A parallel gaussian–bernoulli restricted boltzmann machine for mining area classification with hyperspectral imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(2):627–636.