Comparative Study Between Ensemble Learning and Evolutionary Learning to solve the Higgs Bozon Detection

Michele PULVIRENTI, Marco RIVA

January 2025

Contents

1 Dataset

For this project the Higgs Boson dataset from Kaggle was used.
The dataset contains features derived from particle collision events to classify whether an event is a signal (Higgs boson) or background noise.

1.1 Data cleaning

Initially, some useless columns were removed (EventId, Weight, KaggleSet, KaggleWeight) and the Label column was encoded.

Then, since missing values are designated with the value of -999, **SimpleImputer** was used to fill this values using a **mean** strategy.

Data is also normalized using a **MinMaxScaler**.

We checked the class distribution of data and we noticed that is unbalanced.

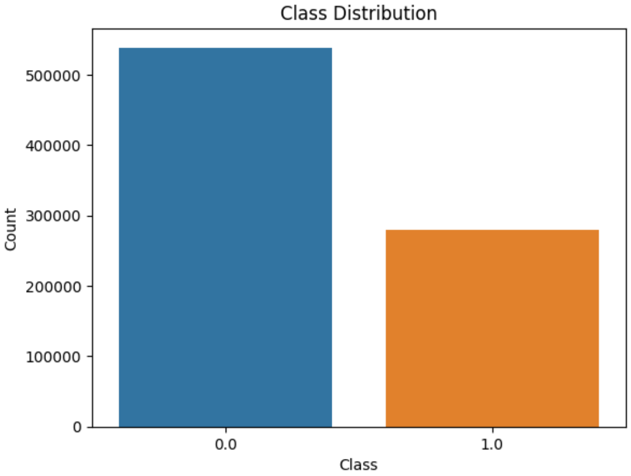

Figure 1: Class distribution

So, we undersampled the majority class and upsampled the minority class obtaining 301730 samples of each class in the training set.

2 Algorithm comparison

2.1 Bagging

As Bagging model, a simple **RandomForest** classifier was chosen.

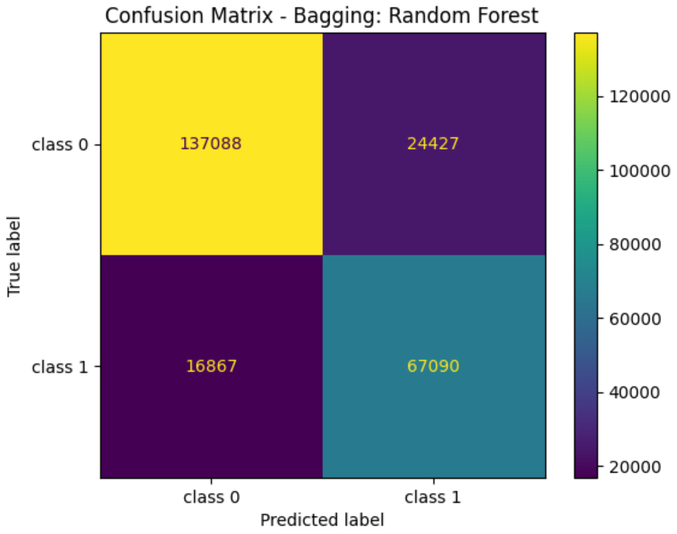

Figure 2: Confusion Matrix Random Forest

The model shows a strong ability to correctly identify class 0 (high TN) but has a significant number of false positives for class 1. The false negatives for class 1 are relatively lower, indicating a decent recall for class 1. Overall, the model performs well but could improve in reducing false positives for class 1.

2.2 Boosting

As Boosting model, a **XGBoost** classifier was chosen.
This classifier was trained on **cuda** to speed up computation.
Since the training time was very low, we had the opportunity to implement a **GridSearchCV** to find the best params for this classifier.

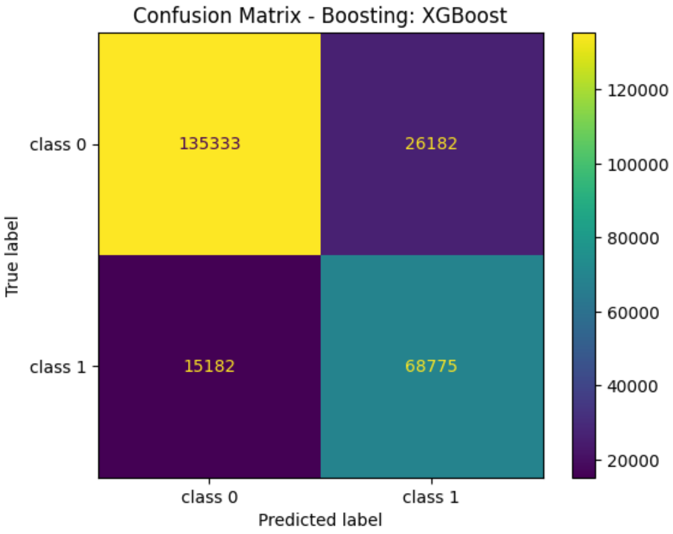

Figure 3: Confusion Matrix XGBoost

The model demonstrates a strong ability to correctly identify class 0 (high TN) and a good number of true positives for class 1. However, there is a notable number of false positives for class 1, indicating that the model may be overly predicting class 1 in some cases. The false negatives for class 1 are relatively lower, suggesting a decent recall for class 1. Overall, the model performs well but could benefit from reducing false positives to improve precision for class 1.

2.3 DEAP Algorithm

Due to the high computation time of the DEAP classifier, each model was trained for only 5 generations.

This led to poor model performance because it needs many iterations to get better.

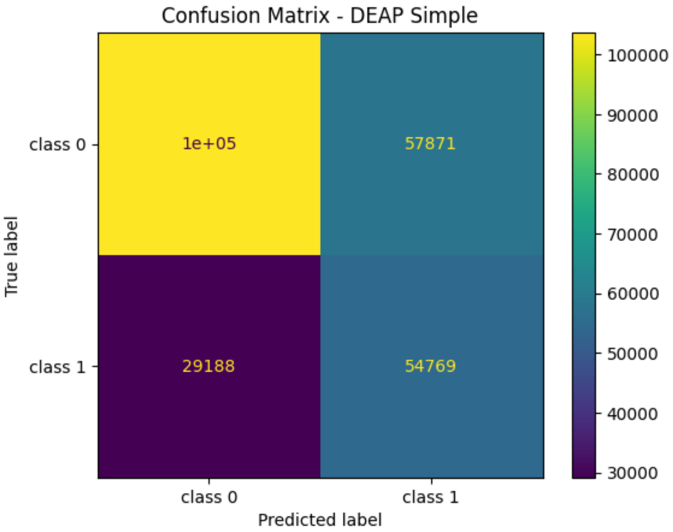

Figure 4: Confusion Matrix DEAP Simple

The model shows a significant number of false positives, indicating that it frequently misclassifies class 0 instances as class 1. This suggests a lower precision for class 1. The number of false negatives is also relatively high, which affects the recall for class 1. Overall, the model has room for improvement, particularly in reducing false positives and false negatives to enhance both precision and recall for class 1.

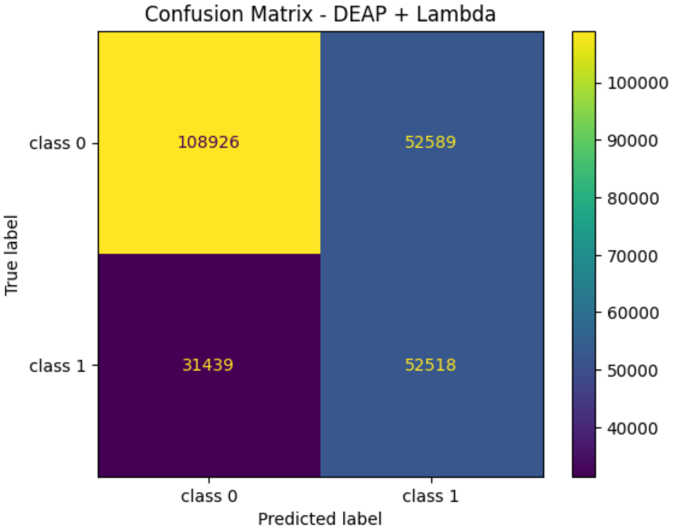

Figure 5: Confusion Matrix DEAP Mu Plus Lambda

The model demonstrates a balanced number of true positives and true negatives, but there is a significant number of false positives and false negatives. This indicates that the model struggles with both precision and recall, particularly in distinguishing between the two classes. Improving the model's ability to correctly classify instances, especially reducing false positives, would enhance its overall performance.

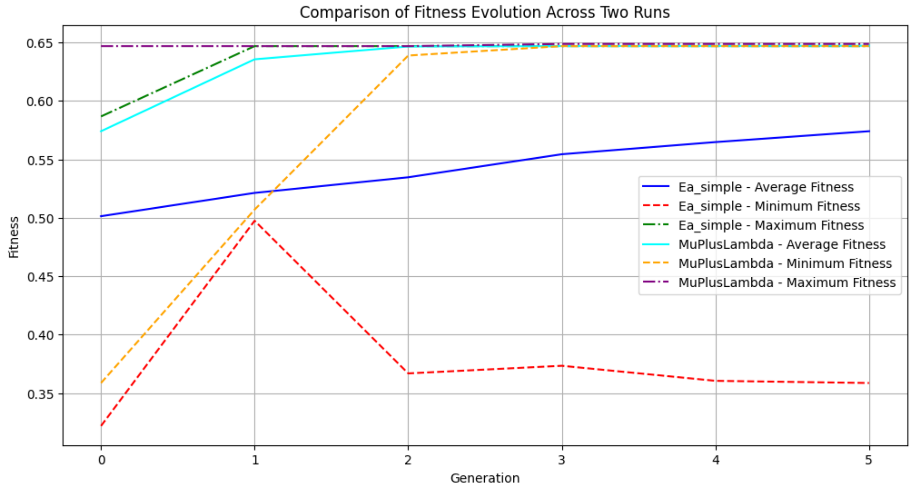

Figure 6: DEAP fitness

The fact that MuPlusLambda values are getting closer to 0.65 suggests that this algorithm is converging towards a higher fitness value, which is generally good as it indicates improvement and optimization over generations.

However, the closer values might also suggest less diversity in the population, potentially leading to premature convergence.

2.4 Performances comparison

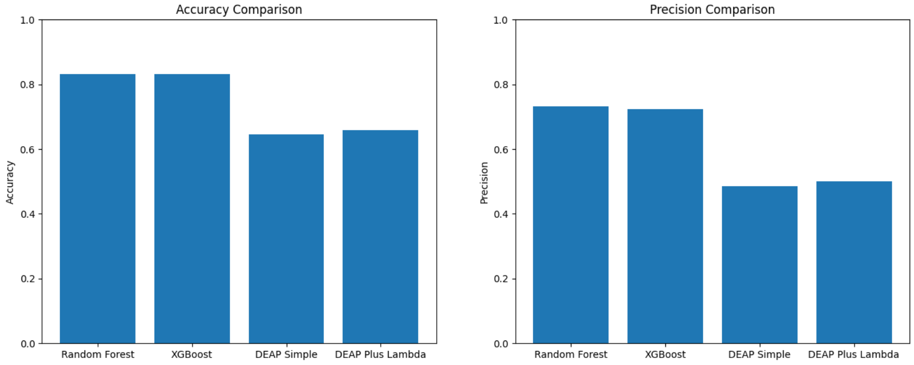

Figure 7: Accuracy and precision

The Random Forest method achieved the highest accuracy at 0.8318, closely followed by XGBoost with 0.8315. Both perform similarly in terms of accuracy.

Random Forest achieved the highest precision (0.7331), followed by XGBoost with a precision of 0.7243.
DEAP Plus Lambda and DEAP Simple had lower precision (around 0.499-0.486), indicating that they may have more false positives.

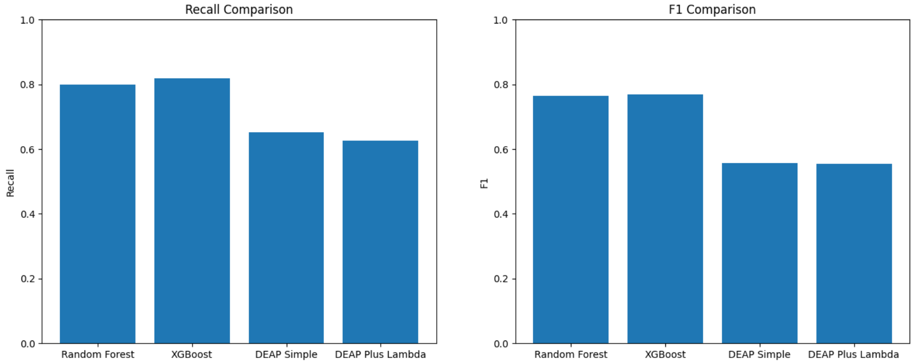

Figure 8: Recall and F1

XGBoost achieved the highest F1-score at 0.7688, while Random Forest was slightly lower at 0.7647. So, XGBoost gives a more balanced result when considering both precision and recall.

XGBoost achieved the highest recall (0.8192), followed by Random Forest (0.7991). Both methods had similar recall values, indicating better sensitivity than specificity compared to DEAP methods.
The recall of DEAP Simple (0.6523) and DEAP Plus Lambda (0.6255) was noticeably lower.

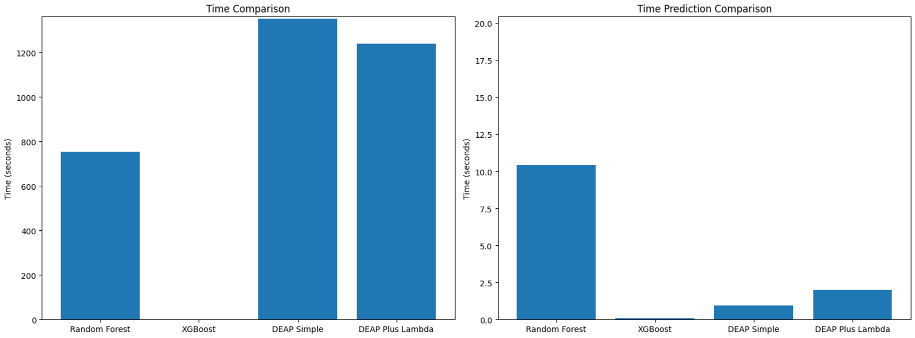

Figure 9: Time training and time prediction

DEAP Simple is the most computationally intensive, taking 1351.4 seconds for training, followed by DEAP Plus Lambda with 1239.2 seconds and Random-Forest with 753.6 seconds.

XGBoost is highly efficient, taking only 1.99 seconds for training.

Evolutionary Learning (DEAP methods) requires significantly more time compared to Ensemble Learning (Random Forest and XGBoost). Both DEAP Simple and DEAP Plus Lambda had much longer training times, indicating a high computational cost in generating and evolving models.
Ensemble methods, especially XGBoost, offer faster training and prediction times, making them more efficient computationally.

Random Forest took significantly longer time for predictions at 10.45 seconds, followed by DEAP Plus Lambda (2.01 seconds) and DEAP Simple (0.96 seconds).
XGBoost remains the fastest on predictions as well at 0.10 seconds

3 Model analysis

3.1 Interpretability

Both Random Forest and XGBoost can be more difficult to interpret compared to traditional models due to their reliance on multiple decision trees or boosted trees, though techniques like SHAP or feature importance can help in understanding them.

DEAP methods may yield more complex solutions that are harder to interpret because they evolve models through genetic algorithms. However, they may also generate simpler solutions in some cases, depending on the parameters of the evolutionary process.

3.2 Complexity

DEAP Simple and DEAP Plus Lambda likely produce models with higher complexity, as these are often evolved through genetic algorithms, resulting in varied and complex solutions based on the learning and evolutionary process.

Random Forest and XGBoost typically have simpler and more structured models compared to evolutionary methods. However, their complexity comes from the ensemble of trees, and they may have reduced interpretability compared to simpler algorithms.

3.3 Advantages and disadvantages

3.3.1 Random Forest

Rando Forest has strong performance with high accuracy, good handling of overfitting by averaging across multiple trees, high precision.

Training and prediction times are longer compared to XGBoost. The model may still suffer from reduced interpretability due to the ensemble of many decision trees.

3.3.2 XGBoost

XGBoost has high accuracy and F1-score, faster training and testing times, robust performance even with large datasets.

Like Random Forest, XGBoost can be harder to interpret. Though faster than Random Forest, it may still require significant computational resources to be fine-tuned.

3.3.3 DEAP Simple

Evolutionary learning could result in novel solutions with good adaptability to specific datasets, flexible in terms of model architecture.

Training time and computational cost are high. The models are harder to interpret, and precision and recall tend to be lower compared to ensemble methods.

3.3.4 DEAP Plus Lambda

Similar advantages to DEAP Simple, with potential for generating varied and adaptable models through evolution.

High computational cost, relatively low precision and recall, and the complexity of evolved models reduces interpretability.

4 Values resume

Method	**Accuracy**	**Precision**	**Recall**	**F1 Score**	**Training Time (s)**	**Prediction Time (s)**
Random Forest	0.8318	0.7331	0.7991	0.7647	753.59	10.45
XGBoost	0.8315	0.7243	0.8192	0.7688	1.99	0.10
DEAP Simple	0.6453	0.4862	0.6523	0.5572	1351.37	0.96
DEAP Plus Lambda	0.6577	0.4997	0.6255	0.5556	1239.17	2.01