

Identification of electrons from B meson decays at the CMS experiment

A. BELVEDERE⁽¹⁾(²)

⁽¹⁾ *INFN, Sezione di Roma I - Rome, Italy*

⁽²⁾ *Dipartimento di Fisica, Sapienza Università di Roma - Rome, Italy*

received 31 January 2022

Summary. — A new electron reconstruction algorithm has been developed at the CMS experiment to test the lepton flavour universality in B meson decays after the discrepancy with the Standard Model highlighted by the LHCb experiment. In this report, the development and the test of the performance of the identification algorithm for this new type of electrons are presented.

1. – Introduction

In latest years, lepton flavour universality has been extensively tested using processes involving B mesons. One of the most interesting results is the measurement by the LHCb experiment [1] of the R_K ratio,

$$(1) \quad R_K \equiv \frac{\Gamma(B^\pm \rightarrow K^\pm \mu^+ \mu^-)}{\Gamma(B^\pm \rightarrow K^\pm e^+ e^-)},$$

that shows a deviation of 3.1σ with respect to the Standard Model expectation of $R_K = 1.00 \pm 0.01$.

Unlike the LHCb, the CMS experiment is designed to identify particles with high transverse momentum ($p_t > 15$ GeV), while final state particles produced by B meson decays have very low p_t . For this reason, a new technique to collect a high number of B meson decays (B-parking [2]) and a new algorithm, called low-pt, to increase the reconstruction efficiency of low transverse momentum electrons have been developed. The low-pt algorithm requires looser conditions than the standard electron reconstruction algorithm. In this way, the number of low p_t electrons reconstructed increases, however also the mistag rate increases and therefore an identification algorithm is necessary to ensure the purity of the sample.

2. – Electron identification algorithm

To develop the low-pt electron identification algorithm, a Machine Learning ensemble method called XGBoost [3] is used. This classifier exploits ensembles of decision trees to distinguish between categories of particles, in this case between particles correctly reconstructed as electrons and fake electrons. The XGBoost algorithm is trained on a Monte Carlo simulation of the decay $B^\pm \rightarrow J/\psi(e^+e^-)K^\pm$ because this process has the same final state as the one involved in the measurement of R_K (eq. (1)).

The particles reconstructed as electrons are divided into two categories depending on the ΔR distance ($\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$) between the reconstructed electrons and the electrons at the generator level produced by the decay of a J/Ψ :

- Signal: $\Delta R < 0.03$
- Background: $\Delta R > 0.1$

These two sets of simulated data compose the dataset used to develop the electron identification algorithm. The dataset is then divided into three independent sub-samples to train and measure the performance of the algorithm:

- *Train set*: set of data used to train the algorithm.
- *Validation set*: set of data used to test the performance of the algorithm while searching for the best hyperparameters and features.
- *Test set*: set of data used to test the algorithm performance when the best set of features and hyperparameters had been already chosen.

The test set represents the 20% of the entire dataset, while the remaining part is further divided: 60% train set and 40% validation set.

2.1. Algorithm training. – The goal of the training phase is to obtain a good discriminating power by keeping the algorithm as simple as possible. Indeed, a too complex algorithm could lead to overfitting, *i.e.*, very good performance during the training phase but poor discriminating power on new data.

The best set of features is chosen starting from a set of 33 features considering both the correlation and the importance of each feature with respect of the others. First of all, the most correlated features are removed and then the behaviour of the AUC score, a measure of the classifier discriminating power, as a function of the number of features is considered. Eventually, the set of features that turns out to be the most reasonable compromise between good performance and a not too complex algorithm is the one with the best 16 features. Reducing the number of features is crucial to maintain the algorithm simple and also to reduce the risk of discrepancies between data and Monte Carlo. The hyperparameters tuned during the training phase were the number of decision trees and the maximum depth of each decision tree. Different possible values of each hyperparameter are tested to find the best working point and eventually a reasonable compromise between performance and simplicity is:

- number of trees: 447
- max depth: 11

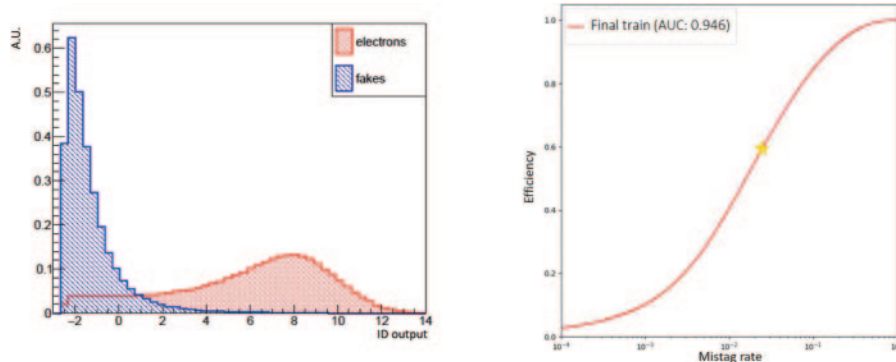


Fig. 1. – Left: signal and background distribution of the variable that represents the identification algorithm output. Right: behaviour of the efficiency as a function of the mistag rate; the yellow star identifies the working point used for the R_K analysis.

2.2. Algorithm performance. – The algorithm performance is tested on the test set. The variable that represents the output of the algorithm shows high discriminating power (fig. 1). The curve that represents the behaviour of the efficiency as a function of the mistag rate is shown in fig. 1, the x axis is in logarithmic scale to make the value of the mistag rate visible, that corresponds to an efficiency of the 60%. The final AUC score, obtained by computing the area under the same curve but in linear scale, is equal to 0.946, while it would be equal to 1 for a perfect algorithm.

3. – Performance on data

The algorithm performance is tested on data by analyzing the process $B^\pm \rightarrow J/\psi(e^+e^-)K^\pm$. A cut based selection is studied on a Monte Carlo simulation of the same process to select this decay from the B-parking dataset. The variables that enter in the cut based selection are: the invariant mass and the transverse momentum computed combining the three candidate particles ($K^\pm e^+ e^-$), the secondary vertex probability and the cosine of the angle between the B^\pm meson line of flight and the vectorial sum of the momentum of the three tracks.

The Monte Carlo truth is used to separate the signal and the background component of the variable that represents the identification algorithm output in the simulated data.

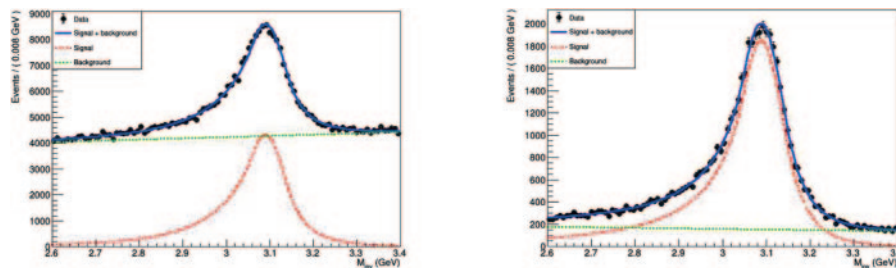


Fig. 2. – Fit to the invariant mass distribution of the selected e^+e^- candidates. On the left, no cut on the variable that represents the identification algorithm output is applied, while on the right the same variable is required to be greater than 7.

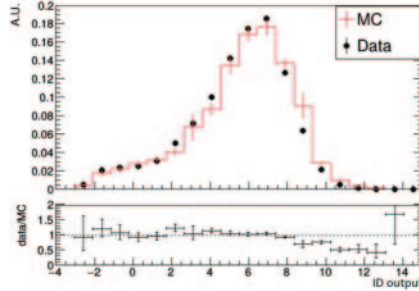


Fig. 3. – Data-Monte Carlo comparison for the signal component of the variable that represents the identification algorithm output.

The selection of the signal component in data is performed using the SPlot technique [4]. This technique leverages the knowledge of the functional form of signal and background of a given variable to separate the two contributions also in any other variable of the same dataset. The variable used is the electron-positron invariant mass; the signal component is parameterized through a double Crystal Ball, while the background is parameterized through an exponential function (fig. 2). To test the discriminating power of the algorithm, a cut on the identification algorithm output is applied (fig. 2). The significance $\frac{S}{\sqrt{S+B}}$, where S and B are the number of signal and background events estimated from the data, improves from 137 ± 5 to 179 ± 9 when the cut is applied.

The data-Monte Carlo comparison (fig. 3) shows a good agreement even if there is a trend for the high value of the identification variable that needs to be further investigated.

4. – Conclusion

A new algorithm for low p_t electron reconstruction has been developed to test lepton flavour universality in B meson decays at the CMS experiment. Modern Machine Learning techniques have been adopted to significantly improve the identification of electrons with $p_t < 15$ GeV. The final results show a good data-Monte Carlo agreement and high discriminating power between electrons and fakes.

REFERENCES

- [1] AAIJ R. *et al.*, arXiv:2103.11769 (2021).
- [2] BAINBRIDGE R., *EPJ Web of Conferences*, **245** (2020) 01025.
- [3] CHEN T. and GUESTRIN C., *Xgboost: A scalable tree boosting system*, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Association for Computing Machinery) 2016, pp. 785–794.
- [4] PIVK M. *et al.*, *Nucl. Instrum. Methods Phys. Res. Sect. A*, **555** (2005) 356.