# Determining the importance of features in a random forest classification

## Random Forest feature importance with Gini Impurity Measure

[Random Forest Feature Importance Computed in 3 Ways with Python | MLJAR](https://mljar.com/blog/feature-importance-in-random-forest/)

*This biggest advantage of this method is a speed of computation - all needed values are computed during the Radom Forest training. The drawbacks of the method is to tendency to prefer (select as important) numerical features and categorical features with high cardinality. What is more, in the case of correlated features it can select one of the feature and neglect the importance of the second one (which can lead to wrong conclusions)*

## Permutation Based Feature Importance

[Random Forest Feature Importance Computed in 3 Ways with Python | MLJAR](https://mljar.com/blog/feature-importance-in-random-forest/)

*The permutation based importance is computationally expensive. The permutation based method can have problem with highly-correlated features, it can report them as unimportant.*

[Feature Selection with the Boruta Package (researchgate.net)](https://www.researchgate.net/profile/Witold-Rudnicki/publication/280138095_Feature_Selection_with_Boruta_Package/links/02bfe51407cd7a64a3000000/Feature-Selection-with-Boruta-Package.pdf)

## Feature Importance computed with SHAP values

[Random Forest Feature Importance Computed in 3 Ways with Python | MLJAR](https://mljar.com/blog/feature-importance-in-random-forest/)

*It is using the Shapley values from game theory to estimate the how does each feature contribute to the prediction.*