***Results***

In our calculations we use four methods of Machine learning for estimation of the type of Fisher iris[[1]](#footnote-1). This data contains 150 observations with four attributes - petal length, petal width, sepal length, sepal width and species. Class variable have 3 unique values: [Iris setosa](https://en.wikipedia.org/wiki/Iris_setosa), [Iris virginica](https://en.wikipedia.org/wiki/Iris_virginica) and [Iris versicolor](https://en.wikipedia.org/wiki/Iris_versicolor). For estimation of the result we use four Machine learning algorithms:

1. Support vector machine[[2]](#footnote-2) with linear kernel
2. Decision tree[[3]](#footnote-3) with Gini index as measure.
3. Random forest[[4]](#footnote-4).
4. Gradient Boosting[[5]](#footnote-5).

For checking results of calculations we also use random splitting data to rain and test sets (80% + 20%). In the next table we can see results for all three methods. We use accuracy of the methods as main characteristic of goodness of method.

|  |  |  |
| --- | --- | --- |
| Method | Accuracy for train data | Accuracy for test data |
| Linear Support Vector Machine, | 97% | 100% |
| Decision tree | 99% | 100% |
| Random forest | 100% | 100% |
| Gradient Boosting | 100% | 100% |

Table 1. Accuracy of the four main methods

As we can see from table 1, all four methods give 100% accuracy for test dataset. This means perfect predict by all four choosing methods. Consider now all methods in details. Main aim in all next conclusions is choose best parameter of the model.

1. ***Support vector machine***. Results of all calculations for this method we can see in the Fig. 1. As we can see from this figure, accuracy is function from penalty parameter and optimal value of this parameter is

For this value accuracy of test and train sets is almost 100%.

1. ***Decision tree***. Results of calculations by decision tree we can see in the Fig. 2. As we can see accuracy is 99% for train data and 100% for test data. We use two parameters for choosing optimal model in DT – max depth and number of features. In result optimal parameters is

Also, last subplot in the Fig. 2 show importance of the features – most importance are features petal width (60% of total importance) and petal length (almost 40% of total importance). Two other features have almost 0% of importance.

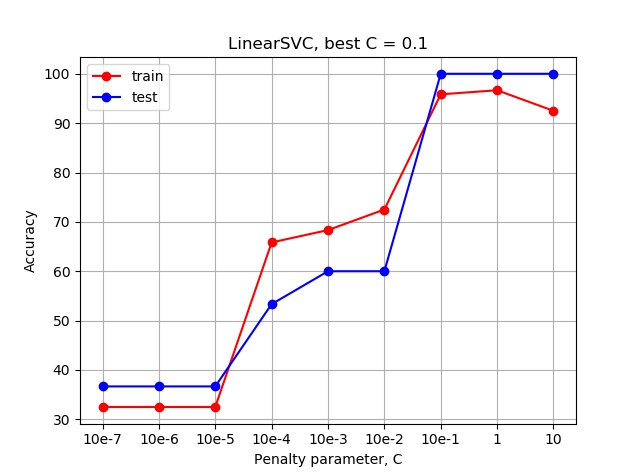


Figure 1. Accuracy vs Penalty parameter, for linear SVM

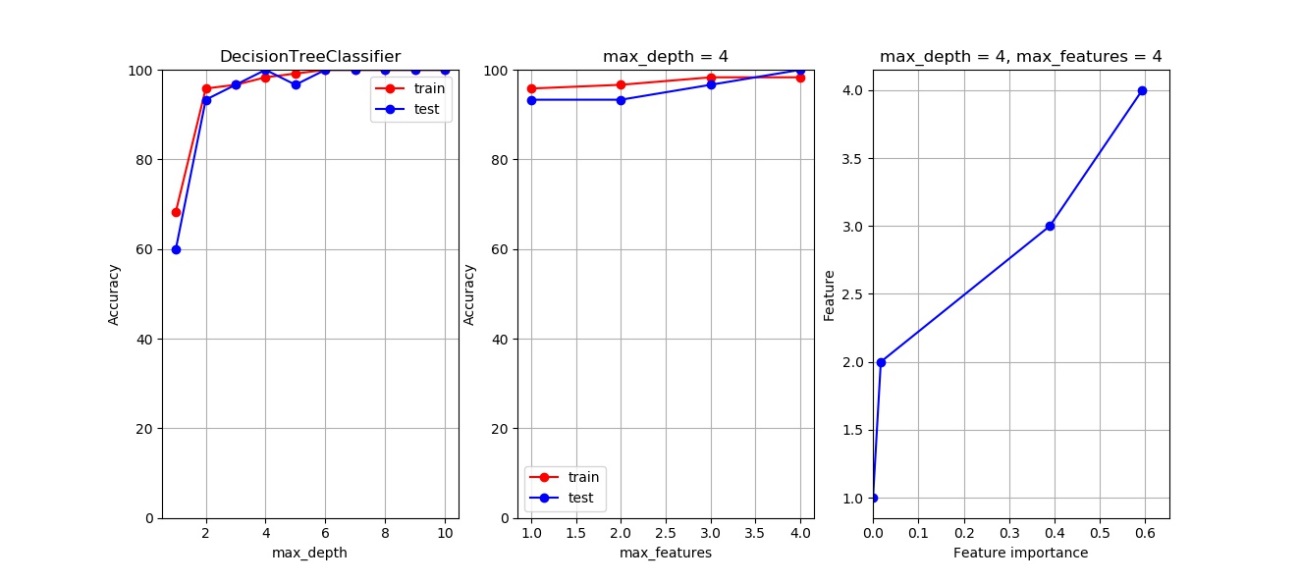


Figure 2. Results of calculations for Decision tree.

1. ***Random forest***. Result of calculations for random forest we can see in the Fig. 3. This method give 100% of accuracy for train and test sets. From first figure we can notice, that this method give almost same result for all possible number of features, hence each number of features can be chosen as optimal for this algorithm:

Also, this method have different values of features importance than decision tree. In this method most importance are also features petal width (37% of total importance) and petal length (almost 28% of total importance) and sepal length (23% of total importance).

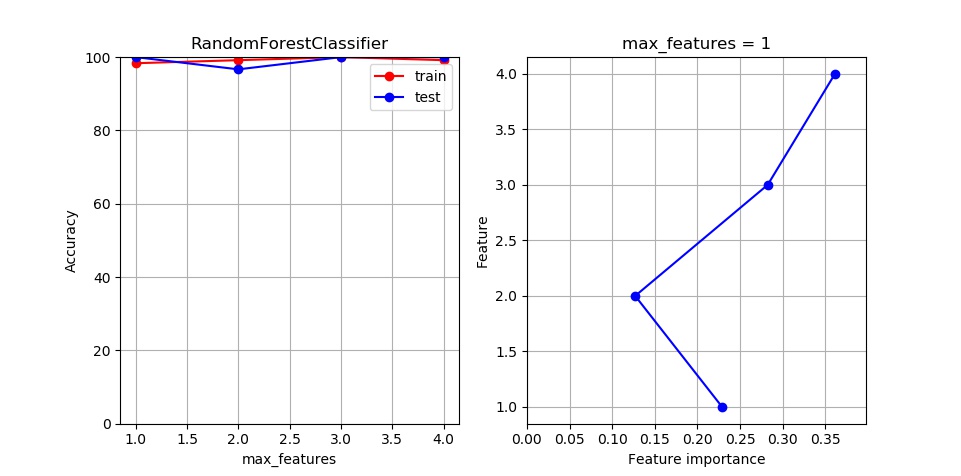


Figure 3. Results of calculations for Random forest.

1. ***Gradient Boosting***. Result of calculations for random forest we can see in the Fig. 4. This method give 100% of accuracy for train and test sets. From first figure we can notice, that this method give almost same result for all possible number of depth, hence each number of features can be chosen as optimal for this algorithm:

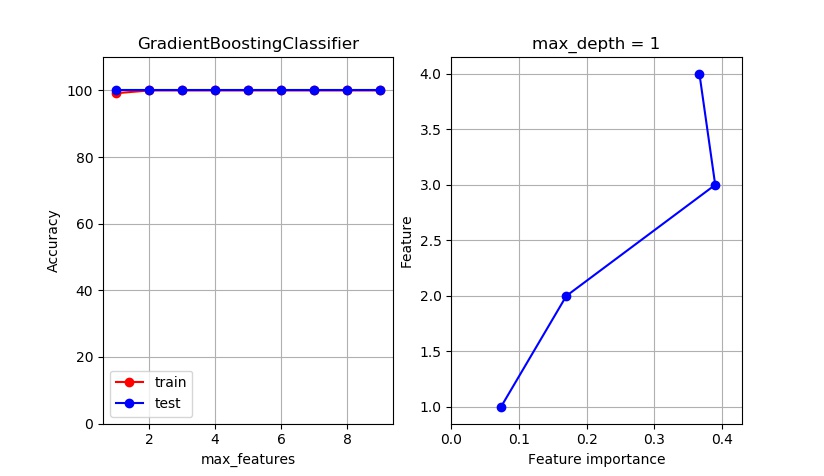


Figure 4. Results of calculations for Gradient Boosting.

Also this method have some different result about importance of variables. Most importance variables for this method are petal length (39% of total importance) and petal width (almost 36% of total importance). Two other features have importance less than 20%.

1. <https://en.wikipedia.org/wiki/Iris_flower_data_set> [↑](#footnote-ref-1)
2. <https://en.wikipedia.org/wiki/Support-vector_machine> [↑](#footnote-ref-2)
3. <https://en.wikipedia.org/wiki/Decision_tree> [↑](#footnote-ref-3)
4. <https://en.wikipedia.org/wiki/Random_forest> [↑](#footnote-ref-4)
5. <https://en.wikipedia.org/wiki/Gradient_boosting> [↑](#footnote-ref-5)