3. From decision trees to decision forest.

A decision forest is a set of several decision trees. These trees can be formed by various methods (or by one method, but with various parameters of work), by different sub-samples of observations over one and the same phenomenon, by use of different characteristics. Such many-sided consideration of a problem, as a rule, gives the improvement of quality of forecasting and a better understanding of laws of the researched phenomenon.

Let us consider a set of trees and an observation *x*. Each tree gives a forecast for *x*. How to find the general (collective) decision for the predicted variable *Y*?

The simplest way to obtain the collective forecast, which gives the given decision forest, is voting method (PR problem) or method of averaging (RA problem).

Using a voting method, a class attributed to observation *x* is a class which the majority of trees prefer. In the regression analysis problem, the predicted value is a mean of forecasts of all trees. We will consider, for example, a set of regression trees, which are shown at figure 18. Consider observation $x = (3,4,8)$. The collective decision will be equal to $Y(x)=(10.2+6.3+11.2)/3=9.233$.

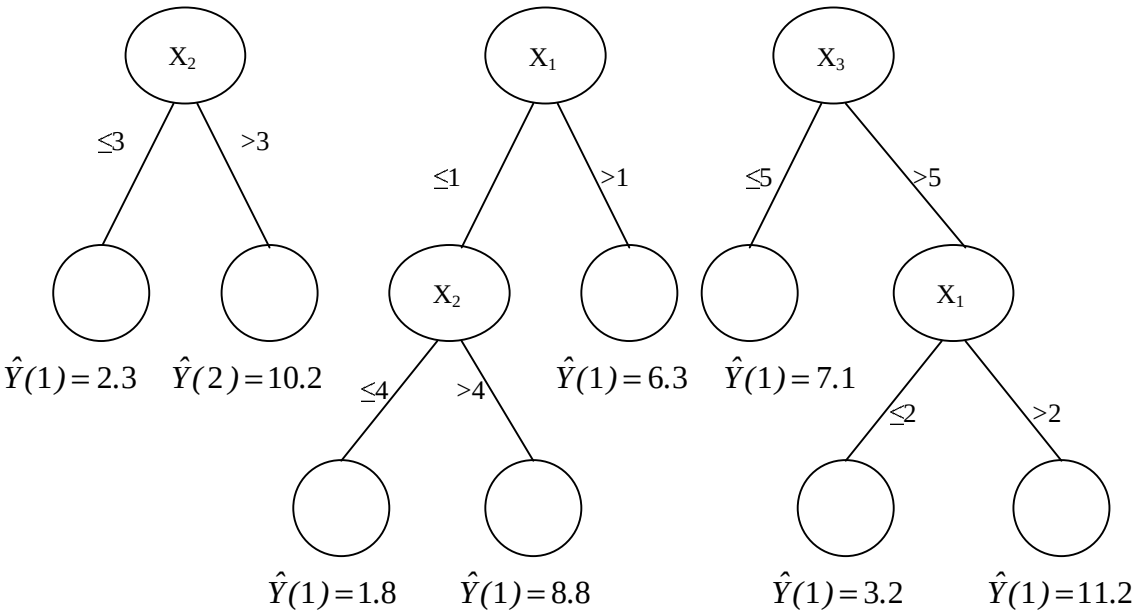

Fig. 18

Next to simple averaging or voting with equal contributions of each vote, it is possible to use a procedure in which the number of mistakes accomplished by each tree on training sample, is taken into account. The fewer mistakes, the greater weight the appropriate voice has.

For estimating the quality of the constructed decision forest, it is possible to use similar ways, which were used for an individual tree. Thus, both control sample and one-leave-out or cross-validation can be used.

Let us consider some ways of construction of a decision forest more detailed.

3.1 Consecutive exception of characteristics.

The given method consists of several stages. On each stage, we build a tree with the help of the full set of observations, but we use a different set of characteristics. On the first stage, all available characteristics are used. The characteristic, which corresponds to a root of the constructed tree, is the most informative, because firstly the further movement in a tree depends on it, and secondly, when branching, the full set of observations is used. On the next stage, when the second tree is forming, all characteristics are used, except for the above mentioned. It is done with the purpose to receive the variant of a tree which is the most distinguished from the previous, i.e. to find out the new set of laws. On the following stages, the characteristics appropriate to the roots of already constructed trees are consecutively excluded. Since the most informative characteristics are excluded, the quality of the trees, as a rule, only become worse from stage to stage.

The algorithm finishes work as soon as the given number of trees will be constructed, or the parameter of quality will reach the given minimal allowable value.

3.2 Using of various sub-samples.

The basic idea of the given method is to use various parts of initial training sample for tree constructions (for the construction of each tree the whole set of characteristics is used). Thus, the quality of the final (collective) decisions, as a rule, is improved. This property can be explained by fact, that the forecasts, given by random (unstable) laws on various sub-samples, at the end have no deciding vote. At the same time, really steady laws on various sub-samples are confirmed only.

Let us consider three methods of sub-samples forming: *bootstrap aggregation method, L-fold method and boosting method.*

In *the bootstrap aggregation method (‘bagging’)* for construction of the new tree, the sub-sample is formed by random independent selection of objects from initial sample. The probability of selection is identical to each object. The volume of sub-sample is set beforehand (for example, 70 % from initial). After the construction of a tree by way of analysis of given sub-sample, the selected observations return into initial sample and the process repeats the given number of times. Thus, each object can repeatedly get into analyzed sub-sample, but also some objects of initial sample can never be included into analyzed sub-samples.

To achieve guaranteed inclusion, it is possible to use *L-fold method*, which uses the same principle as the method of *L*-fold cross-validation described above (paragraph 2.8). The sample divided by case on *L* parts of approximately equal size, then each part is orderedly thrown out, and the rests are united in sub-sample, which is used to construct the next decision tree.

The boosting method is based on the following adaptation idea. At the beginning, the first decision tree is constructed on all objects of initial sample. As a rule, for a part of objects the forecast given by a tree will differ from observed values. During construction of the second tree, more attention is given to those objects which have a bigger error, with the purpose to reduce it. The constructed tree also will give an error for some objects, and the third tree should be constructed to reduce this error. The given procedure repeats the given number of times or until the error is bigger than the given acceptable value.

Errors can be seen the following way. A probability of choosing an object from initial sample is attributed to it. During the construction of the first tree, as well as in the bootstrap aggregation procedure, value of this probability is the same for all objects. On the following stages, the probability of the selection of each object changes. In PR problem, incorrectly classified objects

receive an increment of probability on the given size. In RA problem, an increment of probability is proportional to the square of an error. Thus, for the construction of the next tree, a sub-sample of the given size is formed. The objects are chosen according to the current distribution of probabilities.

As researches show, the last described method allows to reduce an error of the collective forecast more than others do.