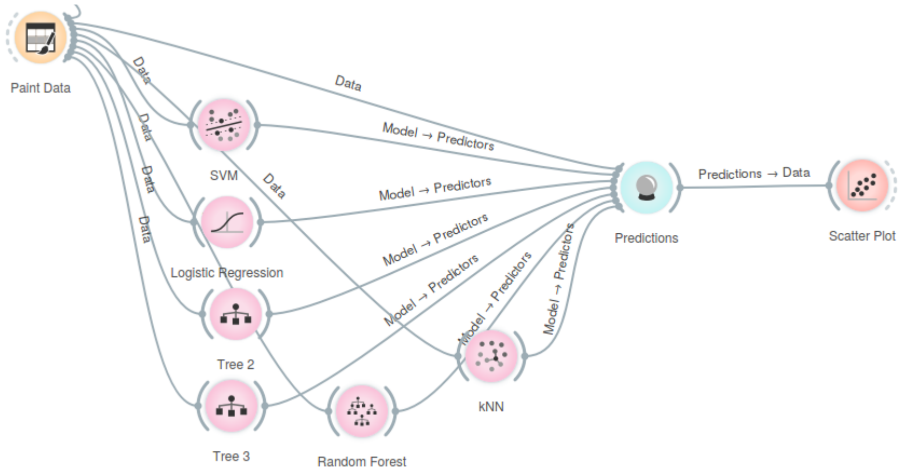
Homework 2: Classifiers and their Decision Boundaries

Jana Obšteter, **jana.obsteter@kis.si**

Problem: Different classifiers use different functions to draw the boundaries and different ways to identify classes of data. Some can draw only vertical and horizontal ones, others can draw only straight lines and some may transform data, draw a line in another dimension and transform data back to the original space. Therefore there are differences between their performance regarding the shape of the data. The aim of this homework was to determine the patterns and shapes of data that each classifier performs well and fails and try to identify the reasons for it.

Methods:



Different patterns of two classes of data was painted. The data widget was connected to each of the six classifiers widgets and to prediction widget. The concordance of predictions and true classes was inspected in a scatter plot.

Results: Here we show scatter plots of a set of data that each classifier performs well and poor on. The colours of the data mark the true classes (blue = class 1, red = class 2) and the shapes mark the predicted classes (circles = class 1, crosses = class 2).

1. Tree with the depth of one

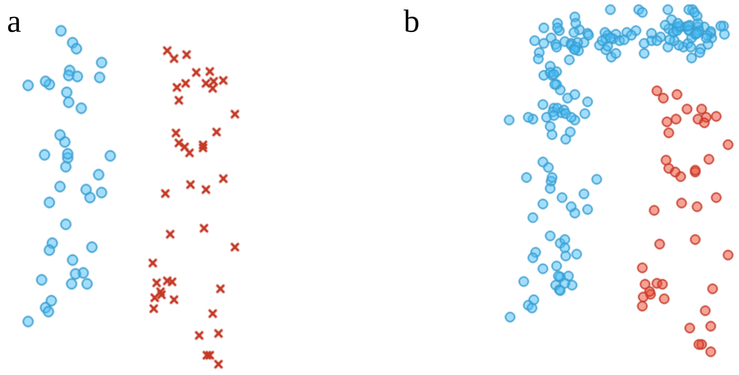


Figure 1: Performance of a tree classifier with a maximum depth of one. a) Good performance of the classifier. b) Poor performance of the classifier.

Trees can draw only horizontal and vertical lines between the data. Therefore they perform well with classifying the data that is vertically and horizontally defined. A tree with the depth of one performs well on data that can be separated with a single straight vertical or horizontal line (Figure 1a). All trees – and especially the ones with low maximal depth – fail with classifying the data that cannot be separated with a specified number (depth) of vertical or horizontal lines. A stump fail in classifying a simple pattern, such as the one in Figure 1b. The tree draws one vertical line – but because there are more blue then red data points on both side of the line, all data points are classified as blue.

2. Tree with depth of two

Similarly to the tree above this tree can separate data that can be separated with vertical and horizontal lines. Compared to the tree with depth of one this tree has more “power” in separating the data since it can draw more lines (two instead of one). The data in Figure 2a shows an example of data that can be separated with two horizontal and vertical lines – a tree with depth of three performs well while a stump fails. Tree fails on classifying the data that cannot be simply separated with vertical and horizontal lines (Figure 2b).

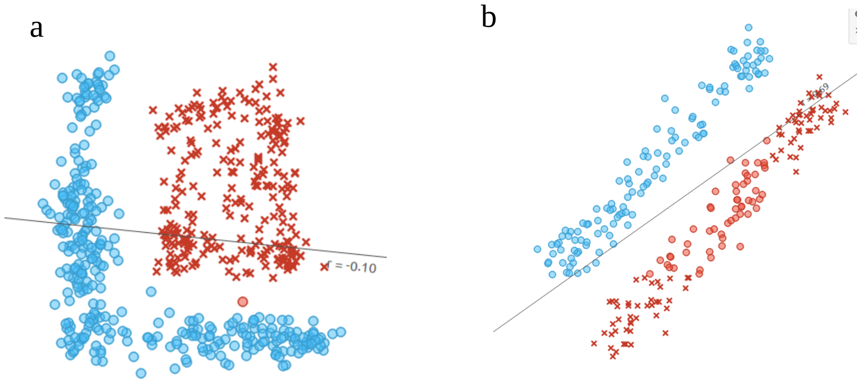


Figure 2: Performance of a tree classifier with a maximum depth of two. a) Good performance of the classifier. b) Poor performance of the classifier.

3. Logistic regression

Logistic regression draws one hyperplane in any direction to separate the data. Therefore it performs well when data can be separated with a simple line or hyperplane. A simple example is in Figure 3a – data can be separated with a diagonal line (hyperplane) - this is where trees fail. Logistic regression on the other hand fails when data cannot be separated with a flat hyperplane (line) – an example of such is in Figure 3b. Logistic regression also performs poor on dataset shown in Figure 2a where trees perform best.

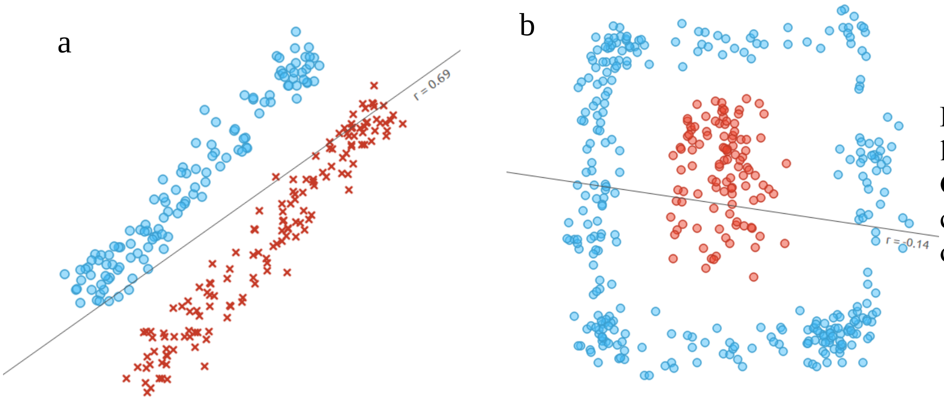


Figure 3: Performance of a logistic regression classifier. a) Good performance of the classifier. b) Poor performance of the classifier.

4. **Support vector machine with kernel basis functions**

Support vector machine with radial basis function uses kernel trick to transform data into higher dimensions where it is separable with a hyperplane. The radial basis function uses a Gaussian function with the gamma parameter determining its variance (gamma is technically the inverse of the standard deviation of the RBF kernel (Gaussian function)). SVM RBF can approximate any continuous function, even the complex patterns, due to the flexibility of the transformation (Figure 4a). However, it performs poorly when data of the same class is not continuous (Figure 4b) and can not be transformed in a way that the two classes are separable with a hyperplane in a higher dimension.

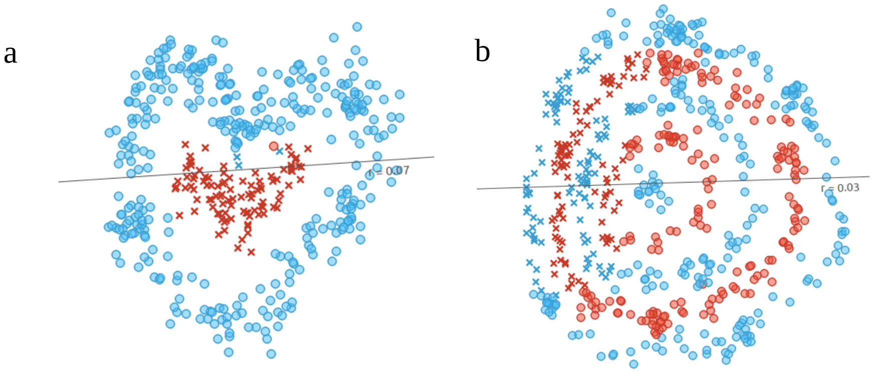


Figure 4: Performance of a support vector machine with radial basis kernel classifier. a) Good performance of the classifier. b) Poor performance of the classifier.

5. Random forest with 100 trees

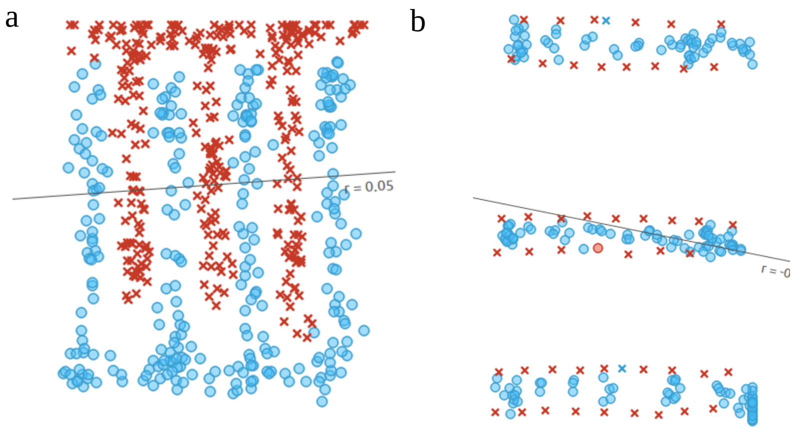


Figure 5: Performance of a random forest with 100 trees classifier. a and b) Good performance of the classifier.

Random forest is a powerful classifier since it is not limited with any functions or the shapes and boundaries of line and hyperplanes. It possesses all the advantages of trees with additional advantage of repeating the classification with variable selection. This way it can remove the “noise” variables from the data. Random forest performs well with many complex patterns, where other classifiers fail. For example, SVM RBF fails on pattern in 5a and nearest neighbor on pattern in 5b. I could not find any patterns where random forest would fail.

6.Nearest neighbor classifier (number of neighbors is 5)

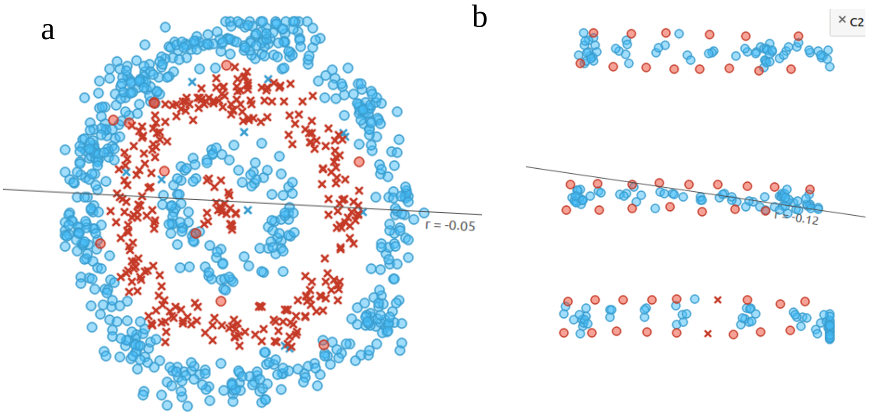


Figure 6: Performance of a nearest neighbor (k = 5) classifier. a) Good performance of the classifier. b) Poor performance of the classifier.

Nearest neighbor classification is not constrained by functions or hyperplanes or their shapes and complexity since it classifies the data based on the few nearest data points. This allows it to establish some very complex boundaries between the classes. The classifier performs well on many complex patterns of data, such as the one in Figure 6a. This is a data set where many other classifiers fail (trees, logistic regression, SVM RBF). However, there are instances where nearest neighbor fails – when the data points within one class are further apart then data points between two classes. This mean that the nearest neighbors are most probably the data points of the opposite class. An example of such is shown in Figure 6b. This problem is not insolvable, since the data could be separated with straight horizontal lines and could be classified with for example random forest.

Conclusion: Different classifiers use different methods, function and algorithms for drawing the boundaries between classes and identifying classes. Specific properties of each of the classifiers make it suitable for different data regarding number of observation, number of features, data distribution, available computational capacities ... Therefore we should first visually explore the data for any detectable patterns and further on test many classifiers, asses their performance with cross validation and use the most appropriate one.