Decision Boundary Extraction of Classifiers

Ningyuan Gong\textsuperscript{a}, Zixiong Wang\textsuperscript{1}, Shuangmin Chen\textsuperscript{2*}, Guozhu Liu\textsuperscript{2} and Shiqing Xin\textsuperscript{1}

\textsuperscript{1}School of Computer Science and Technology, Shandong University at Qingdao, Qingdao, 266237, China
\textsuperscript{2}School of Information and Technology, Qingdao University of Science and Technology, Qingdao, 266061, China
\textsuperscript{a}201400301127@mail.sdu.edu.cn \textsuperscript{*}csmqq@163.com

Abstract. In the field of machine learning, explicitly extracting the decision boundary of a classifier not only helps to visualize the differences between different classifiers, but also provides a more convenient way to determine the class of a sample. We proposed a boundary extraction method that relies on the output result of the classifier but not on the classifier mechanism. We find the initial decision boundary based on Voronoi diagram, and then take the smoothness and simplicity as the driving goal, adaptively adjust the shape of the boundary until convergence. In order to verify the effectiveness and usefulness of the algorithm, the decision boundaries generated by four different classifiers, ANN, SVM, Random Forest, and ELM, were visualized on the four data sets, the classification accuracy is also analyzed based on the extracted decision boundaries. The test results show that the visualization results of decision boundaries and the classification accuracy based on explicit decision boundaries are highly consistent with the classification accuracy of the classifier, which can simulate the decision mechanism of the classifier well.

1. Introduction

Classification is an important problem in machine learning. Its goal is to determine the category of the new samples based on the characteristics of the known samples. There are two perspectives in the study of classifiers: mapping and division. For mapping, the classification model can be viewed as a mapping from the data space to the label set. For division, the classification model can be regarded as a set of decision boundaries that divide the data space into several decision regions, and the training process can be regarded as dividing the data space to obtain decision boundaries. This article aims to extract and visualize the classification decision boundaries with geometric meaning under the classification perspective of division. Visualizing classification boundary can not only help people visually judge the new sample category, but also provide a basis for analyzing the classifier. Simplicity as a driver, an extraction method for decision boundaries with high generalization ability based on Voronoi diagrams is proposed, and experiments are conducted on four data sets with SVM, RF, ANN and ELM classifiers. The experimental results show that, the decision boundary has high accuracy and strong generalization ability, which simulate the classification mechanism well, and the correct rate of separating the two types is highly consistent with the classification of the classifier.
2. Related work

Voronoi diagram. Voronoi diagram, also known as Dirichlet diagram or Tyson polygon\cite{1}, is composed of continuous polygons composed of vertical bisectors connecting straight lines of two neighboring points. There are many algorithms for generating Voronoi diagrams, divide and conquer algorithm\cite{7}, scan line algorithm\cite{8}, incremental method, etc. The lowest time complexity is the Delaunay triangulation algorithm\cite{9}. In this paper, the method of seeking the initial boundary is based on this idea to obtain the set of perpendicular lines between points of different categories.

Decision region and decision boundary. R is a region in the feature space. If all samples \( x \) in R are predicted as \( y_i \) by the classification model, then R is the decision region corresponding to \( y_i \) under the classification model. The decision boundary separates the two decision regions. Decision boundary divides the feature space into several parts\cite{2}. For points on the decision boundary, the probability of whether they belong to two different categories should be equal.

Decision boundary extraction method. The method can be divided as sampling and formal. The formal method is to obtain the analytical expression of the decision boundary based on the knowledge inside the classifier. However, the expression of many classification models is complicated, so it is difficult to get analytical expression. In addition, the formal method is not universal, for different classifiers, different analysis is required. However, the sampling method finds the sampling points on the decision boundary according to the training sample. This article is to use the sampling method to obtain the decision boundary, which is adaptive and is universal for different data sets and different classifiers.

3. Classification methods and preprocess

Our method is divided into the following steps: First, divide the data set into training set and test set to train different classifiers, and predict the category of the test set samples; according to the prediction results, use the Voronoi diagram method to obtain the initial boundary; Finally, with smoothness as the goal, the boundary is optimized under the premise of ensuring classification accuracy to obtain the final result. The algorithm steps are as follows:

Input. Two feature samples \( S = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \} \), \( x_i \in \mathbb{R}^2 \), \( y_i \in \{+1,-1\} \).

Output. Decision boundary \( E = \{ (v_1, v_2), (v_3, v_4), \ldots, (v_{n-1}, v_n) \} \), \( v_i \in \mathbb{R}^2 \).

Step1. After training prediction on the sample \( S \), a new discriminant result \( y_{\text{new}} \) is obtained.

Step2. Use the Voronoi method to find the initial boundary curve \( E_{\text{initial}} \) between different types of samples and the point set \( V_{\text{initial}} \) on the curve.

Step3. Adjust the position of \( V_{\text{initial}} \) to get the optimized and smooth decision boundary \( E \).

We experiment on two-feature data set \( S = \{ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \} \), \( x_i \in \mathbb{R}^2 \), \( y_i \in \{+1,-1\} \). Four classification methods are used, including SVM, RF, ANN and ELM.

3.1. Artificial Neural Network

Artificial Neural Network simulates the activity of neurons with a mathematical model, and builds an information processing system based on mimicking the structure and function of brain neural networks. Since McCulloch and Pitts\cite{5} proposed the first neuron-like computing model, artificial neural networks has been developed a lot\cite{10}, the most commonly used method is the back propagation algorithm ANN model\cite{11}. The topology includes the input layer, hidden layer and output layer.

3.2. Support Vector Machine

Support Vector Machine (SVM) is a machine learning algorithm proposed by Vapnik\cite{3}\cite{4} for classification and regression tasks. The basic idea is to solve the largest hyperplane that can divide samples\cite{12}. For nonlinear classification Problem, it can be transformed into a linear classification problem in a high-dimensional space through a nonlinear transformation\cite{13}, and a kernel function is used to represent the inner product between two instances. The kernel function is:
The corresponding classification decision function is:

\[ f(x) = \text{sign} \left( \sum_{i=1}^{N} \alpha_i^* y_i \exp \left( -\frac{||x - z_i^2||}{2\sigma^2} \right) + b^* \right) \]  

(2)

3.3. Random forest

Random Forest RF is a machine learning algorithm that integrates multiple decision trees based on the idea of integrated learning\(^{[14]}\). The bagging technology in integrated learning is used in the training of multiple decision trees, and the final result passes through all decision trees obtained by voting.

3.4. Extreme Learning Machine

Extreme Learning Machine is a learning algorithm for solving a single hidden layer feedforward neural network\(^{[6]}\). The weights of the ELM hidden layer are randomly generated and not need to update, so the calculation speed is fast\(^{[15]}\). The linear solution obtained by the least square method is used as the weight of the output layer, making the algorithm generalization performance good.

Under the experimental settings, the training accuracy has reached more than 87% (Table 1).

| Data set | ANN     | SVM     | RF      | ELM     |
|----------|---------|---------|---------|---------|
| banana   | 90.06%  | 90.62%  | 95.37%  | 87.26%  |
| bank     | 92.12%  | 93.44%  | 94.03%  | 92.86%  |
| wilt     | 92.26%  | 91.68%  | 95.54%  | 92.09%  |
| iris     | 100%    | 99%     | 95.25%  | 100%    |

After training process, we obtained the classification model to get the predicted label \(y_{\text{new}}\) on test set, which may be different from the original category \(y\). Taking the banana data set as an example, the data under the \(y_{\text{new}}\) have a clearer boundary (Figure 1). We use \(S'=((x_1,y_{\text{new}(1)}),(x_2,y_{\text{new}(2)}),...,(x_n,y_{\text{new}(n)}))\) as the next input to extract the initial boundary.

4. Decision boundary extraction and optimization

Based on the prediction results, we use the Voronoi method to obtain the initial boundary and adjust the position of the boundary driven by smoothness and simplicity, according to the constraints that meet the classifier mechanism until the objective function converges.

4.1. Extract initial classification boundary

The decision boundary must intersect with two types of two points’ connections in different types\(^{[2]}\). We first select the set of middle vertical lines of two-point connections in two different types. This principle is as same as Voronoi diagram. Our method first perform a Delaunay split on the point set \(S'\). When solving the Voronoi diagram for each edge \(e_j\) of the Delaunay split, the principles are used:
If $y_{\text{new}(i)} \neq y_{\text{new}(j)}$, that is, $x_i, x_j$ are points belonging to different categories, the medial perpendicular line obtained by dual processing is a part of the initial decision boundary.

If $y_{\text{new}(i)} = y_{\text{new}(j)}$, $x_i, x_j$ are points belonging to the same category, the classification boundary does not cross the edge $e_{ij}$, so it is not processed.

After using Voronoi method, we obtain the initial decision boundary $E_{\text{initial}} = \{(p_1, p_2), (p_3, p_4), \ldots, (p_{n-1}, p_n)\}$, $p_i \in R^2$. Sometimes, because the classification result of the classifier itself will produce an overfitting situation, there is no clear boundary, resulting in the initial boundary with many some small loops (Figure 2). To facilitate subsequent optimization, we first remove the redundant small loops, leaving only the complete curves.

![Figure 2. overfitting loops in RF classifier](image)

### 4.2. Boundary optimization

The initial boundary is too tortuous (Figure 3), and the points on the initial boundary we obtained $V_{\text{initial}} = \{p_1, p_2, p_3, \ldots, p_n\}$, $p_i \in R^2$ is the set of midpoints, however, the real decision boundary of the actual classifier does not necessarily cross the midpoint of the edge $e_{ij}$. So it is necessary to optimize the initial boundary, that is, to adjust the positions of $V_{\text{initial}}$.

![Figure 3. tortuous initial boundary Einitial](image)

For $V_{\text{initial}} = \{p_1, p_2, p_3, \ldots, p_n\}$, $Y^1_p$ and $Y^2_p$ represent the probability that point $p$ belongs to the first category and the second category. For the points on the target decision boundary, the two probability weights should be equal. Thus, the optimization goal is to smooth the curve under the constraint of $Y^1_p = Y^2_p$. That is, the distance between each pair of adjacent points $p_i, p_{i+1}$ should be as small as possible, so the optimization goal is expressed as:

$$\text{min} \sum_{i=1}^{n} \|p_i - p_{i+1}\|^2, \text{s.t. } Y^1_p = Y^2_p (i = 1, 2, \ldots, n, i \neq j)$$

(3)

However, since $Y^1_p$ and $Y^2_p$ are the results of classification model, there is no definable correlation function between the sample coordinates. Therefore, we express the optimization target energy function as two items, the smoothness part is defined as:

$$W_1 = \sum_{i=1}^{n} \|p_i - p_{i+1}\|^2$$

(4)

And we give it a weight of $C_1$. And the constraint part are transformed into:

$$W_2 = \sum_{i=1}^{n} \|Y^1_p - Y^2_p\|^2$$

(5)

Give it the weight of $C_2$, the optimization goal becomes:

$$\min (C_1W_1 + C_2W_2)$$

(6)
We get the best results when set both $C_1$ and $C_2$ as 1.0. After optimization, while the decision boundary becomes smooth, the values of $Y^1_p$ and $Y^2_p$ also change from the original values with large differences to only slight differences, which are nearly equal (Table 2).

| Initial | Optimized |
|---------|-----------|
| $Y^1_p$ | 0.1997 0.2006 0.2276 0.2180 | 0.5199 0.5218 0.5208 0.5211 |
| $Y^2_p$ | 0.6602 0.6593 0.6511 0.6759 | 0.5267 0.5230 0.5208 0.5191 |

Table 2. probability of the boundary points

For some classifiers such as ANN, the trained model will automatically estimate the values of $Y^1_p$ and $Y^2_p$. However, for classifiers such as SVM or RF which do not generate probability values, the method we adopt is to take each initial point $p$ as the center and make a circle with radius $r=0.05$. Then we make an average sampling to get $x$ points on the circle and predict their labels. For point $p$, the number of points belonging to the first category and the second category are $x_1$ and $x_2$, then the probability that $p$ corresponds to two categories can be expressed as:

$$Y^1_p = \frac{x_1}{x}, \quad Y^2_p = \frac{x_2}{x}$$

We use the fmincon optimization function in Matlab\cite{16}. It will stop when the optimal solution is found. It automatically reaches convergence when iterating 50-100 times. Figure 4 shows the process of optimizing while $k$ is the iteration times. The shape of the first few iterations of the curve changes greatly, as the iteration progresses, the shape of the curve is gradually approaching smoothness, and the range of change is getting smaller and smaller until it converges.

![Figure 4. The optimization process, k = 1, 10 and when convergence.](image)

After optimization, $V_{\text{initial}} = \{p_0, p_1, \ldots, p_n\}$ is adjusted to $V = \{v_0, v_1, \ldots, v_m\}$, and the final decision boundary $E = \{(v_1, v_2), (v_2, v_3), \ldots, (v_{m-1}, v_m)\}$, $v_i \in V$. Compared with the final decision boundary $E$ and the initial boundary $E_{\text{initial}}$, the result not only satisfy the goal of smoothness and simple (Figure 5), it better simulates the prediction mechanism of the classifier to separates samples in different categories.

![Figure 5. Decision boundary after optimization.](image)

5. Experimental results and analysis
In order to compare the accuracy of the method of this article under different classifiers, in the experiment we used PC with Windows 10 system and test four classifiers on four data sets, using
Matlab and C++ programming. We complete the experiment on four datasets: banana, iris, banknote and wilt, all of which are classic machine learning datasets from UCI (Figure 6).

![Figure 6. Four data set in experiment.](image)

We first used several classifiers to train and predict the dataset. The prediction accuracy of the classification results of several classifiers (Table 1) shows that the RF random forest method has a higher classification accuracy for data sets with larger sample sizes. For the data set $S'$ under the classification prediction result $y_{new}$, ANN, SVM and ELM have produced more intuitive boundaries. However, the RF training results are still overfitting which makes the extracted decision boundary’s ability to simulate the decision mechanism of the classifiers is lower than that of the other three methods (Table 3), and our method of extracting decision boundaries has a simulate accuracy rate of more than 99% for the classification results of ANN, ELM, and SVM classifiers, shows that the decision boundary accurately simulate the classifier to predict the sample category.

| Data set | ANN | SVM | RF  | ELM  |
|----------|-----|-----|-----|------|
| banana  | 99.81% | 99.95% | 96.75% | 99.56% |
| bank    | 99.92% | 99.13% | 97.55% | 99.24% |
| wilt    | 99.91% | 100%  | 98.08% | 100%  |
| iris    | 96.07% | 100%  | 97.38% | 100%  |

We put the decision boundary we extracted on the original data set $S$ to instead the classifiers to see the prediction accuracy of classification (Table 4). Compared with the classification prediction results (Table 2), the accuracy rate on the four data sets has been improved to a certain extent.

| Data set | ANN | SVM | RF  | ELM  |
|----------|-----|-----|-----|------|
| banana  | 90.15% | 90.63% | 95.38% | 87.32% |
| bank    | 92.16% | 93.47% | 94.05% | 92.91% |
| wilt    | 92.28% | 91.70% | 95.58% | 92.12% |
| iris    | 100%  | 100%  | 96.50% | 100%  |

6. Conclusion

In this paper, for the problem of machine learning classification, a decision boundary extraction method with high generalization ability based on Voronoi diagram is proposed, and the decision boundaries of four classifiers are compared through experiments. The decision boundary is consistent with mechanism of the classifier, the result of separating data is highly matches the prediction accuracy of the classifier itself. Decision boundaries can be used in many applications like disease monitoring, geological exploration, which can more intuitively and conveniently simulate the
classifier to make predictions. In the future, we can apply the extraction of decision boundaries to high-dimensional data to extract the decision interface, making it more widely used.

References
[1] Okabe A, Boots B, Sugihara K, et al. Spatial Tessellations: Concepts and Applications of Voronoi Diagrams[J]. The College Mathematics Journal, 2000, 43(1)
[2] Chulhee Lee D A L. FEATURE EXTRACTION BASED ON DECISION BOUNDARIES[J]. Pattern Analysis & Machine Intelligence IEEE Transactions on, 1991, 15(4):388-400.
[3] Cortes, C.; Vapnik, V. Support-vector networks. Machine Learning 1995, 20, 273-297.
[4] Vapnik, V. The Nature of Statistical Learning Theory; Berlin: Springer, 1995.
[5] McCulloch, W.S., Pitts, W. A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics 5, 115–133 (1943).
[6] Huang, G. B., Zhu, Q. Y., & Siew, C. K. (2006). Extreme learning machine: theory and applications. Neurocomputing, 70(1-3), 489-501
[7] Yap, C.K. An O(n logn) algorithm for the voronoi diagram of a set of simple curve segments. Discrete Comput Geom 2, 365–393 (1987).
[8] Fortune S. A sweepline algorithm for Voronoi diagrams[C]// SCG '86: Proceedings of the second annual symposium on Computational geometry. ACM, 1986.
[9] Fortune, S. Voronoi diagrams and Delaunay triangulations. Computing in Euclidean Geometry, pp. 193–233 (1992)
[10] Widrow B, Lehr M A. 30 years of adaptive neural networks: perceptron, Madaline, and back propagation[J]. Proceedings of the IEEE, 1990, 78(9):1415-1442.
[11] Rumelhart D E, Hinton G E, Williams R J. Learning Representations by Back Propagating Errors[J]. Nature, 1986, 323(6088):533-536.
[12] Alex J, Smola, Bernhard Scho lkopf. A Tutorial on Support Vector Regression[J]. Statistics and Computing, 2004, 14(3):199-222.
[13] V. D S A. Advanced support vector machines and kernel methods[J]. Neurocomputing, 2003, 55(1/2): p.5-20.
[14] Breiman L, Breiman L, Cutler R A. Random Forests Machine Learning[J]. journal of clinical microbiology, 2001, 2:199-228.
[15] Huang G B, Wang D H, Lan Y. Extreme learning machines: a survey[J]. International Journal of Machine Learning & Cybernetics, 2011, 2(2):107-122.
[16] Spellucci P. A new technique for inconsistent QP problems in the SQP method[J]. Mathematical Methods of Operations Research, 1998, 47(3):355-400.