Confidence Measures in Recognizing Handwritten Mathematical Symbols

Oleg Golubitsky and Stephen M. Watt

University of Western Ontario, London, Ontario, Canada n6a 5b7
http://publish.uwo.ca/~ogolubit
http://www.csd.uwo.ca/~watt

Abstract. Recent work on computer recognition of handwritten mathematical symbols has reached the state where geometric analysis of isolated characters can correctly identify individual characters about 96% of the time. This paper presents confidence measures for two classification methods applied to the recognition of handwritten mathematical symbols. We show how the distance to the nearest convex hull of nearest neighbors relates to the classification accuracy. For multi-classifiers based on support vector machine ensembles, we show how the outcomes of the binary classifiers can be combined into an overall confidence value.

1 Introduction

Recognition of handwritten mathematics is a substantially different problem from natural language text recognition. Because mathematical formulae use a larger variety symbols, which are better segmented, and because the applicability of dictionary-based classification methods is limited in the mathematical context, the problem of recognition of individual mathematical symbols is of special importance. In case of online recognition, it can be thought of as the problem of classification of parametric plane curves.

Previous work has proposed a model for classification of curves based on the representation of the curves in a finite-dimensional vector space by the coefficient vectors of their coordinate functions in an orthogonal functional basis. It has been shown that truncated Legendre-Sobolev series of order about 10 approximate most handwritten character curves to the extent that the approximation is visually indistinguishable from the original curve [1–3]. Furthermore, robust classification methods based on linear support vector machines and distance to the convex hull of nearest neighbors can be applied to this representation. These methods achieve a correct retrieval rate of over 96% for about 230 symbol classes and at least 9 training samples per class [4–6].

Our next goal is to incorporate individual symbol recognition into the classification of entire mathematical expressions. Earlier work [7] has shown that, depending on the mathematical area, statistically, there is a strong preference towards certain symbols or their combinations. This statistical information provides a measure of likelihood of a given symbol within its context, which can be

used to improve the classification results. In order to combine this information with the outcome of the individual symbol recognizer, the latter must have a similar format: namely, together with the suggested class or list of classes, it must supply confidence values associated to each choice. These values represent the likelihood that the choice made by the character recognizer is correct.

For nearest-neighbor-based classification, it is natural to use the distance to the nearest neighbor(s) to produce a confidence measure. In this paper, we show that the error rate increases with the distance following a cubic law, which becomes nearly quadratic for large distances. For support vector machines, the distance to the separating hyperplane can be used. For binary linear classifiers, we show that, independent of the choice of the class pair, the error rate decreases exponentially with the distance to the hyperplane. For an ensemble of binary linear classifiers, we give a formula to combine the confidence values of the individual binary classifiers into a final confidence value, which reflects the likelihood of correctness of the majority vote. Finally, we compare the nearest-neighbor-based and SVM-based confidence measures.

2 Representation and classification

Initially, handwritten symbols are usually represented as a sequence of points, which is sampled in real time by a digital pen. Given the sequences of X and Y coordinates of the points, we compute the moments of the coordinate functions, that is, approximations of the integrals $\int_0^T x(t)t^k\,dt$ and, similarly, for $y(t)$. From the moment integrals, we obtain the Legendre-Sobolev coefficients of the coordinate functions through a linear transformation of the moment vector [1–3]. By translating and normalizing the Legendre-Sobolev coefficient vector, we center and normalize the curve with respect to size. We obtain a representation of the symbol curve as a point in a 20–30 dimensional vector space, which is device-independent and invariant with respect to variations in the speed of writing. Then, vector-space-based classification techniques can be applied to this representation.

Among such techniques, linear support vector machines and the nearest convex hull of nearest neighbors have been considered. These techniques yield high correct retrieval rates (about 95–96%) and allow fast classification among multiple classes [4–6]. Moreover, as will be shown in the next two sections, the decisions produced by these classifiers can be accompanied by reliable confidence measures, without incurring any significant computational overhead.

3 Confidence of SVM classification

As classes of handwritten symbol curves are highly linearly separable [5], it is natural to apply linear support vector machines for classification. It has been observed previously [8] that the distance to the separating hyperplane can be used to produce a reliable measure of confidence in the classifier's outcome. Our

experiments with various pairs of handwritten symbol classes confirm that the error rates decrease exponentially with the distance to the separating hyperplane, see Fig. 1 (left). The thick line, which fits in the envelope of the frequency curves, is $y = 0.5\exp(-4.4x)$, which we take for the confidence measure of the binary linear classifier. Note that, when the distance to the hyperplane approaches zero, the error rate tends to 50%, which agrees with the intuition that points on the hyperplane should equally likely belong to either class.

In a multi-class setting, we use a majority voting scheme, with each binary linear classifier casting one vote for the winning class in the pair. If more than one class gets the maximal number number of votes, the tie is broken randomly. The confidence values for the individual classifiers can be combined into an ensemble confidence value using the following observation. Each individual binary classifier makes a decision with a certain confidence, which approximates the likelihood of this decision being correct. On the other hand, with a certain probability, the decision is incorrect, in which case the vote will go to the opposite class. As a result, the winner of the election may lose enough votes, and another class gain enough votes, so that the outcome of the election changes. The probability of this event is the uncertainty of the ensemble classifier.

An exact computation of this uncertainty would incur exponential complexity. We therefore compute its approximation using the following assumption. Let C_1 be the class that has won the election, and let C_i be another class, for which we are going to compute the probability of winning the election instead of C_1. We assume that this different outcome can occur as a result of (some of) the following events:

1. The vote between C_1 and C_i is reversed.
2. C_1 loses a vote to another class C_j, $j \neq i$.
3. C_1 wins a vote from another class C_j, $j \neq i$.
4. C_i loses a vote to another class C_j, $j \neq 1$.
5. C_i wins a vote from another class C_j, $j \neq 1$.

In other words, we assume that the probability that C_1 or C_i wins/loses more than one vote from/to another class can be neglected.

Let ξ_{ij} be the probability that the vote between classes i and j is correct (approximated by the confidence value of the binary classifier between C_i and C_j). Then the probability that the vote between C_1 and C_i is reversed equals $1-\xi_{1i}$. If W_1 denotes the set of classes C_j, $j \neq i$, from which C_1 has won a vote, then $1 - \prod_{j \in W_1} \xi_{1j}$ is the probability of the second event in the above list. The probabilities of the remaining events are given by similar formulae.

Given the current numbers of votes collected by C_1 and C_i, we select those combinations of the events 1–5 that would result in C_i taking over C_1, and compute the sum of the probabilities of these combinations (for combinations that lead to a tie between C_1 and C_i, we divide the corresponding probability by 2). This sum, denoted η_{1i}, represents the probability that C_1 has wrongly defeated C_i in the election, because of possible errors made by the binary classifiers. Then, $\prod_{i \neq 1}(1 - \eta_{1i})$ is the probability that C_1 is the correct winner of the election.

The error rate versus the resulting measure of uncertainty (equal to one minus confidence) is shown in Fig. 1 (right). Apart from the uncertainty values that are very close to or very far from zero, we can see that the error rates are closely approximated by the uncertainties, the latter being slightly higher. This small difference is due to the fact that, in our setting, the classes may overlap, so more than one class can be considered as correct winner of the election.

4 Confidence of nearest neighbor classification

The distance to the convex hull of nearest neighbors [9] is the technique that has so far yielded the highest correct retrieval rates for classes of handwritten symbol curves [6]. Since this technique is much slower than SVM classification, we apply it only at the last stage, to distinguish among the top few classes that have received many votes. In each of the top S classes, we find k nearest neighbors to the test sample and compute the distance from the sample to their convex hull. The class with the closest convex hull is then chosen.

In Figure 2 (left), the dependence of the error rate on this distance is shown (computed for $S = 10$ and $k = 11$). However, the error bars, corresponding to the 95% confidence intervals, are too wide to allow a definite conclusion about the dependence of the error rate on the distance. The outcome is also influenced by the choice of the bins used to compute frequencies. This especially applies to distances near zero, where the error bars may cross an axis, rendering the corresponding points meaningless, as well as far away from zero, where few data points are available.

A more accurate estimate, which avoids the direct calculation of frequencies in subintervals, can be obtained as follows. Let $e(\rho)$ and $N(\rho)$ be the percentages of misclassified and all samples, respectively, whose distance to the nearest convex hull does not exceed ρ. These cumulative distributions are smooth functions, for which a good fit can be found in the family $f_{a,b,c,d}(t) = (at^b + c)^{-1} + d$. The values of the parameters that provide the lowest root mean square approximation errors are summarized in Table 4. Given the analytic formulae for $e(\rho)$ and $N(\rho)$, we can calculate the error rate as $e'(\rho)/N'(\rho)$. The graphs of this quotient, for dimensions 12, 16, 20, and 24, are shown in Figure 2 (right). The lowest curve (for dimension 24) models the direct error measurement shown in Figure 2 (left).

5 Comparison of confidence measures

A good confidence measure should yield a high value for most correctly classified samples and a low value for most misclassified samples. Let X be the set of all samples, and let X^+ and X^- be the subsets of correctly classified and misclassified samples, respectively. As a measure of quality of a confidence measure $f(x)$ on X, we propose the function

$$q(\xi) = \left(\#\{x \in X^+ \mid f(x) \geq \xi\} + \#\{x \in X^- \mid f(x) \leq \xi\}\right) / \#X,$$

where ξ ranges over all possible confidence values, that is, over the interval $[0, 1]$. When deciding between the outputs of the character recognizer and another independent classifier (such as the statistical character predictor described in the introduction), we will always choose the more confident one (it is easy to show that this choice is optimal). Then, if ξ is the confidence of the character predictor, then the greater $q(\xi)$, the more likely we will make a correct choice.

The qualities of the two proposed confidence measures are shown in Fig. 3 (left), and their difference in Fig. 3 (right). We can see that the SVM confidence measure is better at accepting correct classification results and should be used for high confidence values, while the confidence measure based on the distance to the convex hull of nearest neighbors is better at rejecting incorrect results and should be used for low confidence values. The dividing line between the two is at about 96%, which is the mean correct retrieval rate.

6 Combining prediction and recognition

When human readers interpret a handwritten mathematical formula, they recognize some symbols and infer the others from context. A handwriting recognition system can also use both approaches in order to achieve high retrieval rates. One way to take into account a symbol's context is by looking at the frequencies of the n-grams involving it and neighboring symbols. This approach assumes that at least some neighbors have been recognized with a high confidence and that there are only a few high-frequency choices for the symbol under consideration. In such a setting, we would have several choices proposed by the n-gram predictor, with a probability associated to each choice. It would be convenient to have the character recognizer's output in a similar format. Then, assuming that these two classifiers are independent (indeed, their decisions are based on very different considerations), we can combine their outputs by maximizing the posterior probability. This implies that we choose the class for which the product of the probabilities associated by the character recognizer and the n-gram predictor is maximal. We may also let the value of n (the size of n-grams) vary in order to maximize this product.

Using the confidence values presented in this paper, we can obtain a distribution on the set of all classes as follows. Let the confidence value be the probability associated to the winning class (denote it p_1). Then, discard the winning class from consideration and repeat the classification process. Associate to the new winner the resulting confidence value, multiplied by $(1 - p_1)$. Our experiments show that these probabilities decrease very rapidly. In fact it takes on average 7 and at most 26 iterations for the probabilities to become less than 10^{-10}.

Moreover, in the case of SVM classification, we do not need to collect all the votes again, but instead we discard only the ones that involve the winning class and recalculate the ensemble confidence values. This will incur only a mild computational overhead. Indeed, let

$$W_i = \{j \mid C_i \text{ won the vote } C_i - C_j\}, \quad L_i = \{j \mid C_i \text{ lost the vote } C_i - C_j\},$$

where i and j range over the indices of classes still under consideration. Assume that the products $\prod_{j \in W_i} \xi_{ij}$, $\prod_{j \in L_i} \xi_{ij}$ have been computed. When the winning class is discarded, exactly one element will be removed from W_i or L_i, for each i; call it j_i. Then the above products for the set of remaining classes can be obtained using a single division by ξ_{ij_i}. Using the new values of the products, the probabilities of events 1–5 in Section 3 can be obtained in time proportional to the number of classes. Since the computation of ξ_{ij} is quadratic in the number of classes, and since the number of times we need to discard the winner and calculate the new confidence values is small, the complexity of computing the probabilities is by an order of magnitude lower than the complexity of computing the initial confidence values.

In the case of distance-based classification, very little additional computation is needed to obtain the probabilities, since the confidence values for all classes are derived directly from the distances.

7 Conclusions

We have derived confidence measures for two classifiers, one based on SVM and one based on nearest neighbor geometry. We have demonstrated quantitatively that the SVM ensemble confidence measure performs better than the distance to the convex hull of nearest neighbors at samples classified with high confidence. Future work will be to combine the character recognizer with statistical frequency data using the proposed confidence measures.

References

1. Char, B., Watt, S.M.: Representing and Characterizing Handwritten Mathematical Symbols through Succinct Functional Approximation. Proc. Intl. Conf. on Docum. Anal. and Rec. (ICDAR) (2007) 1198–1202.
2. Golubitsky, O., Watt, S.M.: Online Stroke Modeling for Handwriting Recognition. Proc. 18th Intl. Conf. on Comp. Sci. and Soft. Eng. (CASCON) (2008) 72–80.
3. Golubitsky, O., Watt, S.M.: Online Computation of Similarity between Handwritten Characters. Proc. Document Recognition and Retrieval (DRR XVI) (2009) C1–C10.
4. Golubitsky, O., Watt, S.M.: Online Recognition of Multi-Stroke Symbols with Orthogonal Series. Accepted to ICDAR (2009).
5. Golubitsky, O., Watt, S.M.: Improved Character Recognition through Subclassing and Runoff Elections. Ontario Research Center for Computer Algebra Tecnical Report TR-09-01.
6. Golubitsky, O., Watt, S.M.: Tie Breaking for Curve Multiclassifiers. Ontario Research Center for Computer Algebra Tecnical Report TR-09-02.
7. Watt, S.M.: Mathematical Document Classification via Symbol Frequency Analysis. Proc. Towards Digital Mathematics Library (DML 2008) 29–40.
8. Li, M., Sethi, I.: Confidence-Based Classifier Design. Pattern Recognition **39** (7) (2006) 1230–1240.
9. Vincent, P., Bengio, Y.: K-local Hyperplane and Convex Distance Nearest Neighbor Algorithms. Adv. in Neural Inform. Proc. Systems, The MIT Press, 2002. 985–992.

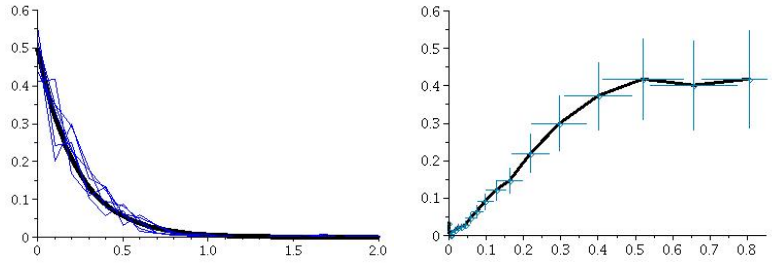

Fig. 1. *Left:* Error rate *vs* distance to hyperplane. Thin curves are for different class pairs, thick curve for exponential fit. *Right:* Ensemble uncertainty (horizontal/vertical error bars correspond to 95th percentiles of the normal/Bernoulli distributions.)

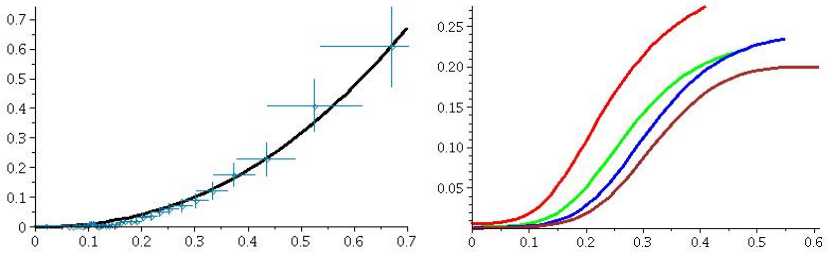

Fig. 2. Error rate *vs* distance to the nearest convex hull of nearest neighbors.

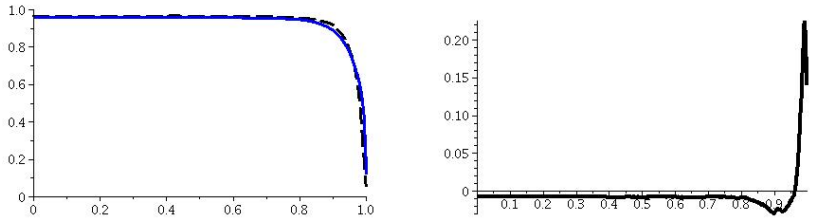

Fig. 3. *Left:* Qualities of confidence measures. Solid is SVM, dashed is convex hull of nearest neighbors. *Right:* Their difference.

Dim	a_e	b_e	c_e	d_e	error	a_N	b_N	c_N	d_N	error
12	0.18	-2.98	22.5	-0.00023	0.00021	0.0010	-3.00	1.0037	0.024	0.0038
14	0.21	-3.31	28.9	-0.00004	0.00013	0.0012	-3.08	1.0043	0.022	0.0030
16	0.29	-3.39	33.3	-0.00014	0.00013	0.0013	-3.16	1.0058	0.022	0.0027
18	0.39	-3.49	35.3	-0.00014	0.00012	0.0014	-3.22	1.0065	0.021	0.0025
20	0.42	-3.63	36.4	-0.00014	0.00016	0.0015	-3.29	1.0074	0.021	0.0027
22	0.44	-3.77	37.5	-0.00014	0.00012	0.0017	-3.33	1.0078	0.021	0.0026
24	0.40	-3.95	38.9	-0.00010	0.00011	0.0019	-3.36	1.0079	0.021	0.0026

Fig. 4. Parameters of the best fits to cumulative distributions.