An Empirical Study of Self/Non-self Discrimination in Binary Data with a Kernel Estimator

Thomas Stibor

Department of Computer Science
Darmstadt University of Technology
64289, Darmstadt, Germany
stibor@sec.informatik.tu-darmstadt.de

Abstract. Affinity functions play a major role within the artificial immune system (AIS) framework and crucially bias the performance of AIS algorithms. In the problem domain of self/non-self discrimination by means of negative selection, affinity functions such as the Hamming distance or the r-contiguous distance are frequently applied to measure distances in binary data. In recent years however, several limitations and problems with these distance measurements in negative selection have been identified. We propose to measure distances in binary data by means of probabilities which are modeled with a kernel estimator. Such a probabilistic model is preeminently applicable for the self/non-self discrimination problem. We underpin our proposal with an empirical study on artificially generated and real-world datasets.

1 Introduction

Self/non-self discrimination models are discussed intensively in immunology and also in the artificial immune system (AIS) community. In the field of AIS the negative selection is a popular, however also a controversial approach to discriminate self from non-self [1],[2]. The discrimination capability of negative selection is biased by the chosen shape space and the used affinity functions. In binary shape space (also called Hamming shape space) all immune components are represented as bit strings. The affinity between any two bit strings is measured with affinity functions such as the Hamming and r-contiguous distance. In recent years, however, research revealed that affinity functions used in negative selection induce manifold problems. The problems can be summarized as follows. Poor generalization capabilities, that is, the accurate self/non-self prediction of *unseen* bit strings [2]. Infeasible computational complexity of finding detectors [2]. To overcome these problems, it seems reasonable to look beyond the "classical" affinity functions proposed in the field of AIS.

The problem of self/non-self discrimination can be stated as follows. Given self data, that is, a sample $\mathcal{S}$ of bit strings which characterizes self:

– *Does an unseen bit string* $\mathbf{u}$ *belong to self?*

P.J. Bentley, D. Lee, and S. Jung (Eds.): ICARIS 2008, LNCS 5132, pp. 352–363, 2008.

This problem is usually tackled by using negative selection and corresponding affinity functions for binary data. Observe that this problem cannot be answered satisfyingly without giving a clear specification of self. In other words, the problem cannot be fitted in any machine learning framework.

By considering this problem from a statistical point of view, it can be equivalently formulated as follows:

– *Does* $\mathbf{u}$ *originate from the same probability distribution as bit strings in* $\mathcal{S}$*?*

This second question can be answered by assuming that $\mathcal{S}$ is i.i.d. generated by some unknown distribution which corresponds to self and that self data occurs concentrated. This leads to the problem of estimating the underlying probability distribution which generates $\mathcal{S}$ and finally to the rejection of data of low probability. Once the underlying probability distribution is properly modeled, membership queries, that is the first question, can be also answered.

In their seminal paper Kullback and Leibler stated [3]:

> *"We are also concerned with the statistical problem of discrimination by considering a measure of the "distance" or "divergence" between statistical populations in terms of our measure of information."*

By reviewing known problems in negative selection, it seems therefore reasonable to tackle the self/non-self discrimination problem by means of a statistical approach which will be discussed and empirically investigated in this paper. We structure the paper as follows: The kernel estimator method for binary data is explained in section 2. An experiment on artificially generated data is provided in section 2.1. The statistical discrimination function is presented in section 3. In section 4, an additional experiment is performed to explore whether regions where most of the self data is concentrated can be appropriately modeled. Results of detecting corrupted handwritten digits are presented in section 5. Conclusions and outlooks are provided in section 6.

2 Kernel Estimator for Binary Data

Kernel estimators belong to the class of non-parametric models and are well-known methods for estimating densities for continuous domains [4],[5]. For binary data, that is discrete data, kernel estimators such as Parzen Window or Nearest-Neighbor are not applicable due to their continuous nature. Aitchison and Aitken proposed a kernel estimator for binary data [6].

Given sample $\mathcal{S} = \{\mathbf{x}_t\}_{t=1}^{N}$ from $\{0,1\}^l$ and kernel function

$$K_h(\mathbf{x}|\mathbf{y}) = \begin{cases} h^{l-d(\mathbf{x},\mathbf{y})}(1-h)^{d(\mathbf{x},\mathbf{y})} & \text{for} \quad \frac{1}{2} \leq h < 1 \\ \begin{cases} 1 & (\mathbf{x} = \mathbf{y}) \\ 0 & (\mathbf{x} \neq \mathbf{y}) \end{cases} & \text{for} \quad h = 1 \end{cases} \tag{1}$$

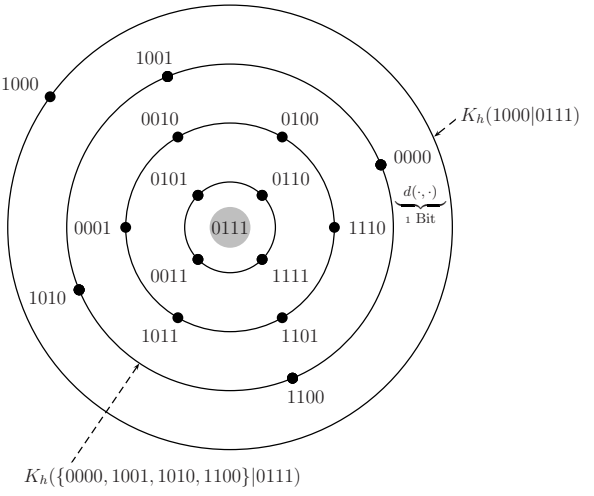


Fig. 1. Coherence between kernel function $K_h(\cdot|\cdot)$ and Hamming distance $d(\cdot,\cdot)$. The Hamming distance from 0111 to all bit strings sitting on the same ring is related to the probability mass function $K_h(\cdot|0111)$. Note that the Hamming distance is increasing from center 0111 to bit strings sitting on the outer rings at one bit per ring.

where

$$d(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y}) \equiv \sum_{i=1}^{l} x_i \text{ XOR } y_i$$

is the Hamming distance, and h the bandwidth parameter. The true underlying probability distribution which corresponds to sample $\mathcal{S}$ can be estimated by:

$$\widehat{P}(\mathbf{x}|\mathcal{S}) = \frac{1}{N}\sum_{i=1}^{N} K_h(\mathbf{x}|\mathbf{x}_i). \tag{2}$$

The kernel function $K_h(\mathbf{x}|\mathbf{y})$ is a probability mass function and is related to the Hamming distance between $\mathbf{x}$ and $\mathbf{y}$ (see Fig 1). Loosely speaking, the smaller the Hamming distance the larger the probability. Analogous to continuous kernel estimators, the bandwidth parameter h in (2) controls the *smoothness*, i.e. the influence of the surrounding bit strings. The smallest bandwidth $h = 1/2$ gives the uniform distribution $\widehat{P}(\mathbf{x}|\mathcal{S}) = (1/2)^l$ for all $\mathbf{x} \in \{0,1\}^l$, whereas the largest bandwidth $h = 1$ gives the distribution of the relative frequencies.

To find an appropriate value of bandwidth parameter h such that consistency properties are obeyed, Aitchison and Aitken proposed to maximize:

$$\prod_{i=1}^{N} \widehat{P}(\mathbf{x}_i \,|\, \mathcal{S} \setminus \{\mathbf{x}_i\}) \tag{3}$$

where $\mathcal{S}\setminus\{\mathbf{x}_i\}$ denotes sample $\mathcal{S}$ with excluded bit string $\mathbf{x}_i$ (leave-one-out method).

Note that (3) can lead to numerical instabilities for large sample sizes. To avoid such a problem, one can also maximize the corresponding log-likelihood value:

$$\sum_{i=1}^{N} \log \widehat{P}(\mathbf{x}_i \,|\, \mathcal{S} \setminus \{\mathbf{x}_i\}). \tag{4}$$

It is worthwhile to notice that by maximizing (3), (4) respectively, one mutually minimizes the Kullback-Leibler divergence [3]:

$$\sum_{i=1}^{N} G(\mathbf{x}_i) \log \left(\frac{G(\mathbf{x}_i)}{\widehat{P}(\mathbf{x}_i|\mathcal{S})} \right). \tag{5}$$

The Kullback-Leibler divergence can be considered as a closeness measure between the true underlying probability distribution $G(\mathbf{x})$ and the estimated distribution $\widehat{P}(\mathbf{x}|\mathcal{S})$. The smaller the value of (5), the more "similar" are the true and estimated probability distribution.

2.1 Experiment on Data Generated by Mixture of Multivariate Bernoulli Distributions

For creating binary self data, it is helpful to use a *generative model* such that samples can be generated from the true underlying distribution which is specified by some parameters. A multivariate Bernoulli distribution is a generative model and fulfills this criterion. To be more precise, the distribution is specified by parameter vector $\boldsymbol{\Theta} \in [0,1]^l$ and takes binary values $x_i = 1$ with probability Θ_i and $x_i = 0$ with the complementary probability $1 - \Theta_i$, for $i = 1, \ldots, l$. It therefore has probability mass function:

$$P(\mathbf{x}|\boldsymbol{\Theta}) = \prod_{i=1}^{l} \Theta_i^{x_i} \, (1 - \Theta_i)^{1-x_i}. \tag{6}$$

To model higher order correlations in the generated samples, it is necessary to combine mixtures of multivariate Bernoulli distributions:

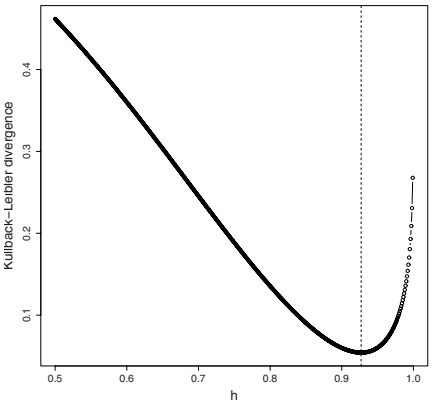
$$P(\mathbf{x}|\overline{\boldsymbol{\Theta}}, \boldsymbol{\alpha}) = \sum_{m=1}^{M} \alpha_m \, P(\mathbf{x}|\boldsymbol{\Theta}_m), \tag{7}$$

where the mixture proportion $\boldsymbol{\alpha} \in \mathbb{R}^M$ has to obey the convex combination $\sum_{m=1}^{M} \alpha_m = 1$ with $\alpha_m \geq 0$ and $\overline{\boldsymbol{\Theta}}$ is composed of $(\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2, \ldots, \boldsymbol{\Theta}_M)$.

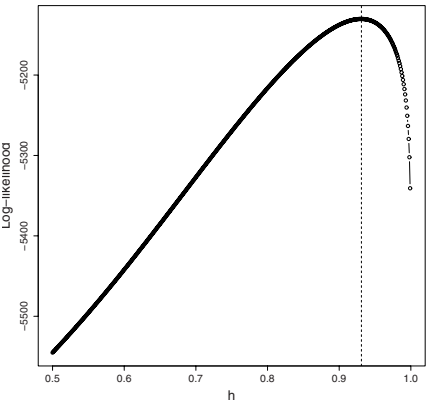
In this experiment we specified $M = 3$ mixtures of multivariate Bernoulli distributions with following parameters:

$$\boldsymbol{\alpha} := \begin{bmatrix} \frac{1}{9} \\ \frac{3}{9} \\ \frac{5}{9} \end{bmatrix}, \qquad \overline{\boldsymbol{\Theta}} = \begin{bmatrix} \boldsymbol{\Theta}_1 \\ \boldsymbol{\Theta}_2 \\ \boldsymbol{\Theta}_3 \end{bmatrix} := \begin{bmatrix} \frac{1}{10} & \frac{4}{5} & \frac{3}{5} & \frac{1}{5} & \frac{1}{5} & \frac{7}{10} & \frac{3}{10} & \frac{1}{10} \\ \frac{1}{2} & \frac{1}{5} & \frac{2}{5} & \frac{7}{10} & \frac{4}{5} & \frac{3}{5} & \frac{1}{10} & \frac{2}{5} \\ \frac{7}{10} & \frac{1}{10} & \frac{3}{10} & \frac{1}{5} & \frac{1}{2} & \frac{7}{10} & \frac{1}{2} & \frac{3}{5} \end{bmatrix},$$

and denote the true underlying distribution as $G(\mathbf{x}) \equiv P(\mathbf{x}|\overline{\boldsymbol{\Theta}}, \boldsymbol{\alpha})$.



(a) The dotted line denotes the value of h where the smallest Kullback-Leibler divergence value between the true probability distribution $G(\mathbf{x})$ and kernel estimated probability distribution $\widehat{P}(\mathbf{x}|\mathcal{S})$ can be found.



(b) The dotted line denotes the value of h where the largest log-likelihood value of $\widehat{P}(\mathbf{x}|\mathcal{S})$ can be found.

Fig. 2. Coherence between kernel parameter h and Kullback-Leibler divergence (left), and log-likelihood evaluation by means of the leave-one-out method (right). The value of h which maximizes (4) corresponds to the smallest Kullback-Leibler divergence value.

In non-parametric models *no* parametrized distribution has to be fitted in the samples; therefore, one has to determine only the suitable bandwidth parameter h. In this experiment the parameter h is run from 1/2 to 1. The corresponding value of (4) as well as the Kullback-Leibler divergence between $G(\mathbf{x})$ and $\widehat{P}(\mathbf{x}|\mathcal{S})$ are depicted in Figure 2.

One can observe that by maximizing (4) one mutually minimizes the Kullback-Leibler divergence between true the probability distribution and the kernel estimated. To say it the other way around, given a sample $\mathcal{S}$ which characterizes self and bandwidth parameter h which maximizes (4). One can model the underlying probability distribution which corresponds to $\mathcal{S}$ and hence is able to discriminate self from non-self by means of probabilities. Note that the Hamming distance is still used as a measurement, however expressed in terms of weighted kernel estimated probabilities. This allows the modeling of smooth discrimination boundaries, whereas the plain Hamming distance does not offer such degrees of smoothness (see [7]).

3 Statistical Discrimination in Binary Data

Let $\mathcal{S}$ be a sample which characterizes self and h the bandwidth parameter which is found such that (4) is maximized. A probabilistic discrimination function for the self/non-self problem[1] can be defined as follows:

[1] In the field of machine learning this equivalent problem is termed outlier detection or novelty detection.

$$\mathfrak{D}(\mathbf{x}, t) = \begin{cases} \widehat{P}(\mathbf{x}|\mathcal{S}) \geq t, & \text{self} \\ \text{otherwise}, & \text{non-self} \end{cases} \tag{8}$$

where $\mathbf{x}$ is the to classified bit strings and t some threshold. By specifying a value for t, one obtains enclosed decision region(s) such that most of the support of the distribution is captured. In other words, if $\mathbf{x}$ is within the region(s) where most of the self data is concentrated, then $\mathbf{x}$ belongs to self otherwise it belongs to non-self. It is worthwhile to mention that discrimination function $\mathfrak{D}$ can be extended to a multi-class decision function by assigning $\mathbf{x}$ to that class where the corresponding class-conditional probability is largest.

4 Experiment on Data Generated by Mixture of Gaussian Distributions

Due to the fact that mixtures of multivariate Bernoulli distributions are hardly to visualize, a second experiment is performed. In this experiment we explore whether regions, where most of the self data is concentrated, can be appropriately modeled. Therefore, self data is generated by a mixture of 2-dim. Gaussian distributions with different mean vectors and covariance matrices and consists of 5000 data points. The generated self data is visualized in Figure 3(a), the corresponding density image is depicted in Figure 3(b).

One can see in Figure 3(a) that self data is concentrated in regions of high density. This coincidence with our assumption and leads to the problem of finding regions where most of the self data is concentrated.

Note that the domain of (2) is $\{0,1\}^l$. We therefore use the mapping from $\mathbb{R}^2 \rightarrow \{0,1\}^l$ proposed in [8]. That is, the data is min-max normalization to $[0,1]^2$ and discretized to bit strings of length $l = 16$

$$\underbrace{b_1, b_2, \ldots, b_8}_{b_x}, \underbrace{b_9, b_{10}, \ldots, b_{16}}_{b_y},$$

where the first 8 bits encode the integer x-value

$$i_x := \lceil 255 \cdot x + 0.5 \rceil$$

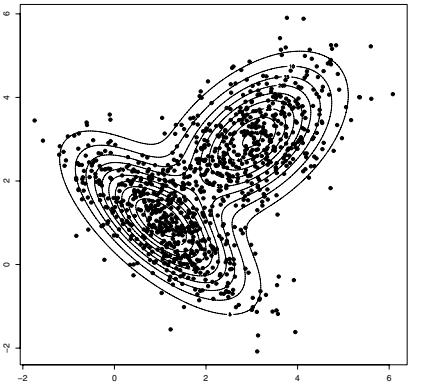
and the last 8 bits the integer y-value

$$i_y := \lceil 255 \cdot y + 0.5 \rceil,$$

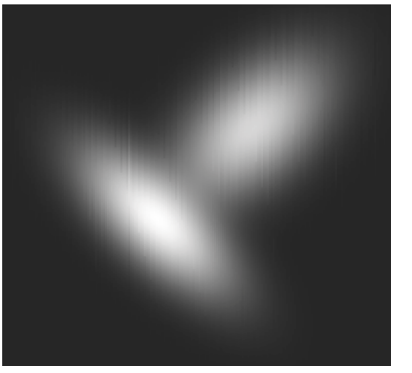
that is,

$$\begin{aligned} [0,1]^2 &\rightarrow (i_x, i_y) \in (1, \ldots, 256) \times (1, \ldots, 256) \\ &\rightarrow (b_x, b_y) \in \{0,1\}^8 \times \{0,1\}^8. \end{aligned}$$

By means of the leave-one-out method bandwidth parameter $h = 0.909$ is determined. The corresponding density image is depicted in Figure 4(b), where each

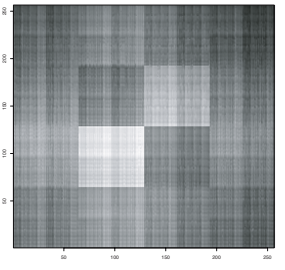


(a) Self data is generated by a mixture of two multivariate Gaussian distributions with different mean vectors and covariance matrices.

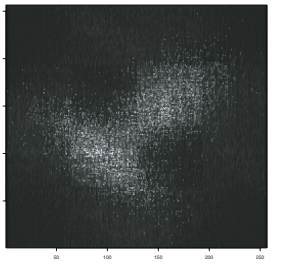


(b) Density image of the underlying distributions. Self data is concentrated in regions of high probability (light regions).

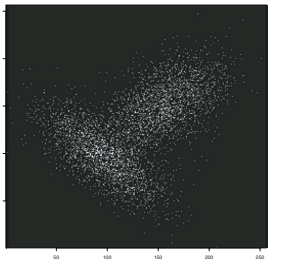
Fig. 3. Self data is sampled from a mixture of multivariate Gaussian distributions



(a) Bandwidth value $h = 0.55$ results in a too underfitted model.



(b) For $h = 0.909$ a proper generalization is obtained.



(c) A too overfitted model $h = 1$ results in a poor generalization.

Fig. 4. Coherence between different bandwidth values and estimated models

pixel in the 256×256 grid represents a bit string of length $l = 16$. The color corresponds to the probability $\widehat{P}(\mathbf{x}|\mathcal{S})$. For the sake of comparison, two additional density images of bandwidth value $h = 0.55$ and $h = 1$ are depicted (see Fig. 4(a), 4(c)). One can observe that the true underlying distribution can be closely approximated if an appropriate value of h is determined. For a too oversmoothed bandwidth value $h = 0.55$ the resulting model is underfitted, whereas for $h = 1$ the model is overfitted. For $h = 0.909$ the probability distribution is appropriately modeled, thus good generalization is obtained.

5 Handwritten Digit Recognition Experiment

Recognizing handwritten digits is a challenging real-world problem in the field of machine learning. In this experiment, we focus on the problem of outlier detection, that is, the detection of digits which are corrupted. In the language of self/non-self discrimination, self of each digit is modeled as shown in section 2 and corrupted digits are detected by means of decision function (8).

To obtain meaningful results regarding the robustness of the kernel estimator method, experiments on two popular handwritten digits datasets (USPS and MNIST database) are performed.

The USPS database[2] contains handwritten digits scanned from envelopes by the U.S. Postal Service. The digits are size-normalized in a 16×16 fixed image of gray color values in the range $[-1, 1]$. The database consists of 7291 training examples and 2007 testing samples which are partitioned in digit sets 0 to 9 (see Table 1).

Table 1. Number of digits in training and testing set in the USPS database

digit	0	1	2	3	4	5	6	7	8	9
training set	1194	1005	731	658	652	556	664	645	542	644
testing set	359	264	198	166	200	160	170	147	166	177

The USPS database contains a number of corrupted digits, which not even humans can correctly classify (human error rate 2.5%) and therefore is a challenging benchmark. However, the database is also criticized due to their noisy nature [9].

The MNIST database[3] contains also handwritten digits. However if one compares the two databases, then one can observe that the MNIST database has cleaner digits thus becomes the state of the art benchmark database in recent years. The digits in the MNIST database are centered and size-normalized in a 28×28 fixed-size image of gray color values $\{0, 1, \ldots, 255\}^{784}$. The MNIST database consists of 60000 training examples and 10000 testing samples which are partitioned in digit sets 0 to 9 (see Table 2).

Table 2. Number of digits in training and testing set in the MNIST database

digit	0	1	2	3	4	5	6	7	8	9
training set	5923	6742	5958	6131	5842	5421	5918	6265	5851	5949
testing set	980	1135	1032	1010	982	892	958	1028	974	1009

To obtain comparative results between the two databases, digits in the USPS database are min-max normalized from $[-1, 1]$ to gray color values $\{0, 1, \ldots, 255\}^{256}$. Both databases are finally binarized by means of:

[2] Available at: `http://www-stat.stanford.edu/~tibs/ElemStatLearn/datasets/`

[3] Available at: `http://yann.lecun.com/exdb/mnist/index.html`

$$\mathfrak{B}(\mathbf{z}, t_{\mathtt{bw}}) = \begin{cases} z_i \leq t_{\mathtt{bw}}\,, & 0 \\ \text{otherwise}\,, & 1 \end{cases} \tag{9}$$

where threshold $t_{\mathtt{bw}} = 128$ is chosen and $\mathbf{z} \in \{0, 1, \ldots, 255\}^{256}$ (USPS database), $\mathbf{z} \in \{0, 1, \ldots, 255\}^{784}$ (MNIST database), respectively.

The bandwidth value h of each digit class for both training sets is determined by means of the leave-one-out method and results in:

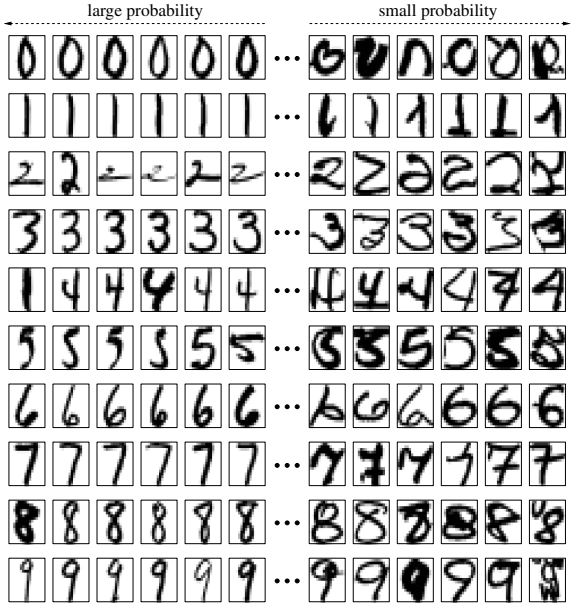
	digit	0	1	2	3	4	5	6	7	8	9
USPS	h	0.917	0.99	0.871	0.888	0.906	0.877	0.92	0.938	0.889	0.93
MNIST	h	0.94	0.984	0.928	0.935	0.945	0.936	0.946	0.956	0.929	0.95

5.1 Results

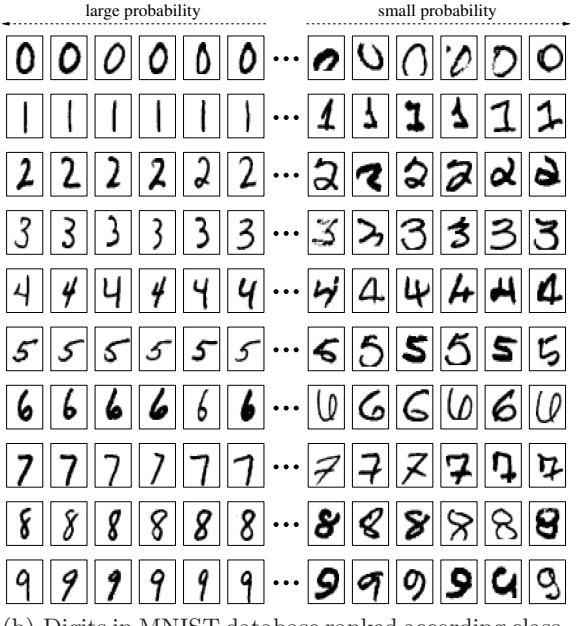
Both testing sets contain no information regarding the magnitude of corruption of the digits. As a result, it is difficult to obtain meaningful outlier detection results. Due to such difficulties, the digits of each class are ranked. To be more precise, the digits of each class are ranked in descending order regarding their class-conditional probabilities (see Fig. 5). One can see that corrupted digits have small class-conditional probabilities and hence can be recognized as outliers by decision function $\mathfrak{D}$ with regard to some threshold value t. Furthermore, one can observe that some less corrupted digits ("7") which are written according to the European standard have small probabilities. This is an undesirable result and is caused by the fact that the training set contains an underrepresented amount of those digits. This problem can be addressed by tuning the corresponding bandwidth parameter towards more smoothness. Moreover, one can observe that in the USPS database the mislabeled digit 1 has a large estimated probability and thus can not be detected as an outlier.

Table 3. State of the art classification results on testing sets USPS and MNIST. For a detailed overview see [9], pp. 219 and pp. 341.

Database	Classifier	Error rate (%)
USPS	Linear SVM	8.9
	Relevance Vector Machine	5.1
	Hard margin SVM	4.6
	SVM	4.0
	Hyperplane on KPCA features	4.0
	Kernel Fisher Discriminant	3.7
	Virtual SVM	3.2
	Virtual SVM, local kernel	3.0
MNIST	Linear classifier	8.4
	3-Nearest-Neighbor	2.4
	⋮	⋮
	Virtual SVM with 8 VSVs per SV	0.6
	Virtual SVM with 12 VSVs per SV	0.6



(a) Digits in USPS database ranked according class-conditional probabilities in descending order.



(b) Digits in MNIST database ranked according class-conditional probabilities in descending order.

Fig. 5. First six digits of each class (testing set) ranked according to the largest, smallest class-conditional probability, respectively. One can see that corrupted digits have smaller probabilities compared to "clean" digits having larger probabilities.

In terms of the overall classification error rate[4], the following results are obtained on the testing sets: USPS database 7.47 % and MNIST database 3.92 %. Compared to the state of the art classification results (see Table 3) our achieved results are limited competitive. However, one has to mention that the best achieved classification results are obtained with highly tuned classifiers which are invariant regarding translation and rotation. Furthermore, we used binary features rather than gray color values from $\{0, 1, \ldots, 255\}$ and therefore utilized a poorer feature representation due to the operation on binary data. On the other hand one should mention that kernel based estimation methods suffer of high computational complexity. This results from the fact that each bit string is used to evaluate term (2). However there exist different techniques for reducing the computational complexity of kernel based estimation methods (e.g. [10],[11]). These techniques can be also applied to reduce the computational complexity of term (2). Additional improvements regarding the detection accuracy could be obtained by applying different binarization techniques.

6 Conclusion

Self/non-self discrimination in binary data is a challenging problem in the field of AIS. It has been tackled with negative selection and affinity functions such as the Hamming and the r-contiguous distance. Research results in recent years, however, revealed manifold problems in negative selection with regard to the generalization capability, and with regard to the computational complexity. We proposed to model *self* by means of a statistical approach, namely by estimating the underlying probability distribution which corresponds to self with a kernel estimator. The proposed method was motivated by the fact that the self/non-self discrimination problem can be clearly specified from a statistical view point. Such a statistical method is far from any immune-inspired paradigms, however, overcomes known problems in the immune-inspired negative selection method. From our point of view it is worthwhile to introduce such a statistically founded method in the field of AIS. It allows us to consider problems formulated in the field of AIS from a mathematically founded perspective, rather than by biologically motivated arguments. Observe that in the early days the term "neural network" was motivated towards modelling networks of real neurons in the brain. Nowadays:

> *"The perspective of statistical pattern recognition, however, offers a much more direct and principled route to many of the same concepts."* [Neural Networks for Pattern Recognition, C. M. Bishop]

Acknowledgment

The author thanks Erin Gardner for her valuable suggestions and comments.

[4] Recall: discrimination function $\mathfrak{D}$ can be extended to a multi-class decision function by assigning $\mathbf{x}$ to that class where the corresponding class-conditional probability is largest.

References

1. Forrest, S., Perelson, A.S., Allen, L., Cherukuri, R.: Self-nonself discrimination in a computer. In: Proceedings of the Symposium on Research in Security and Privacy, pp. 202–212. IEEE Computer Society Press, Los Alamitos (1994)
2. Stibor, T.: On the Appropriateness of Negative Selection for Anomaly Detection and Network Intrusion Detection. PhD thesis, Darmstadt University of Technology (2006)
3. Kullback, S., Leibler, R.A.: On information and sufficiency. The Annals of Mathematical Statistics 22(1), 79–86 (1951)
4. Duda, R., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley-Interscience, Chichester (2001)
5. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Heidelberg (2006)
6. Aitchison, J., Aitken, C.G.G.: Multivariate binary discrimination by the kernel method. Biometrika 63(3), 413–420 (1976)
7. Stibor, T.: Discriminating self from non-self with finite mixtures of multivariate bernoulli distributions. In: Proceedings of Genetic and Evolutionary Computation Conference – GECCO. ACM Press, New York (to appear, 2008)
8. González, F., Dasgupta, D., Gómez, J.: The effect of binary matching rules in negative selection. In: Cantú-Paz, E., Foster, J.A., Deb, K., Davis, L., Roy, R., O'Reilly, U.-M., Beyer, H.-G., Kendall, G., Wilson, S.W., Harman, M., Wegener, J., Dasgupta, D., Potter, M.A., Schultz, A., Dowsland, K.A., Jonoska, N., Miller, J., Standish, R.K. (eds.) GECCO 2003. LNCS, vol. 2723, pp. 195–206. Springer, Heidelberg (2003)
9. Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge (2001)
10. Giromali, M., He, C.: Probability density estimation from optimally condensed data samples. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(10), 1253–1264 (2003)
11. Fukunaga, K., Hayes, R.R.: The reduced parzen window classifier. IEEE Transaction on Pattern Analysis and Machine Intelligence 11(4), 423–425 (1989)