Data characteristics that determine classifier performance

Christiaan van der Walt and Etienne Barnard

Human Language Technologies Research Group
Meraka Institute, Pretoria, South Africa
{cvdwalt,ebarnard}@meraka.org.za

Abstract

We study the relationship between the distribution of data, on the one hand, and classifier performance, on the other, for non-parametric classifiers. It is shown that predictable factors such as the available amount of training data (relative to the dimensionality of the feature space), the spatial variability of the effective average distance between data samples, and the type and amount of noise in the data set influence such classifiers to a significant degree. The methods developed here can be used to gain a detailed understanding of classifier design and selection.

1. Introduction

The quest to optimize the performance of trainable classifiers has a long and varied history. Soon after the design of the earliest parametric and linear classifiers, researchers found refinements (such as polynomial classifiers and the nearest-neighbour rule) that produced more accurate classification on comparable data sets. Hence, the quest for "the most accurate" classifier was initiated, and several generations of candidates for that title have been proposed: kernel functions, neural networks, support vector machines, etc.

In some ways, this activity has been extremely productive – we today have a wide range of classifiers that are employed in numerous applications, from credit scoring to speech processing, with great technical and commercial success. However, from another perspective, this entire enterprise can be considered a dismal failure: we still do not have single classifier that can reliably outperform all others on a given data set [1], and the process of classifier selection is still largely a process of trial and error.

This apparent contradiction would not be surprising in the context of purely parametric classifiers, since the accuracy of a particular parametric classifier on a given data set will clearly depend on the relationship between the classifier and the data. The concept of a single best parametric classifier is clearly not useful, and a trial-and-error process will generally be required to find the parametric form that best describes a given data set (although statistical tests may be employed to guide that search).

In the realm of non-parametric classifiers, however, there is less awareness of the need to harmonize the characteristics of data and classifiers. As we describe in Section 2, a few empirical studies have shown that the choice of optimal classifier does in fact depend on the data set employed, and some guidelines on classifier selection have been proposed. However, these guidelines do not provide much insight on the specific characteristics of the data that will determine the preference of classifier. To address that shortcoming, we focus on two pieces of conventional wisdom, which are often repeated in review papers [2] and text books [3]. The first wisdom is that discriminative classifiers tend to be more accurate than model-based classifiers at classification tasks (see, e.g. [3, p. 77]); the second is that k-nearest-neighbour (kNN) classifiers are almost always close to optimal in accuracy, for an appropriate choice of k (e.g. [2, p. 17]). A common subsidiary to the latter belief is that the best value of k can only be determined empirically. We therefore focus our attention on three specific topics:

- Do model-based classifiers substantially outperform discriminative classifiers under any circumstances?
- What attributes of classification data determine the optimal value of k in a kNN classifier?
- Are there specific circumstances that cause the kNN to underperform other classifiers substantially?

We have developed a methodology (summarized in Section 3) that uses artificial data sets to probe the interaction between classifiers and data sets. In Section 4 we show that these three questions can be answered using that methodology, and in Section 5 we discuss the implications of those findings.

2. Data sets and classifiers: previous work

Several comparative studies have been conducted to determine features in data that will predict classification performance. Tax and Duin [4] consider a one-class classification problem, 19 classifiers and 101 data sets were used. They define two features to characterise data sets, namely the effective sample size and the class overlap. The classifier disagreements for the data sets are calculated and data sets with the disagreement measures are mapped into a two-dimensional space. Datasets for which classifiers perform well and poorly are investigated. They find that the only variable that characterizes a data set well is the effective sample size (the ratio between the number of observations and variables in a data set).

Brazdil *et al*. [5] performed a comparative study based on the results of the StatLog Project [1]. The StatLog project compared 22 classifiers on more than 20 different data sets. The aim of [5] was to obtain a set of rules to predict classification performance of data sets. Statistical and information theoretic measures were used to extract features from data sets. These measures were used together with the classification results of the StatLog project to construct an expert system, named the Application Assistant, to predict the classification performance of various classifiers on a particular data set. The C4.5 algorithm [6] was used to construct rules from the given data. The classification results were considered one at a time by the C4.5 algorithm, until a

final set of rules were constructed. All the rules had a confidence measure to indicate the usefulness of a rule. Only 22 classification errors were used per classifier due to the fact that they used the StatLog project results. The rules that were generated by the expert system were not very meaningful due to a lack of training data – it is easy to find counterexamples to the conclusions reached in [5]. For example, one of the basic rules in [5] is that the Linear Discriminative classifier will perform well (with a confidence or information score of 0.247) if the number of samples in the data set is less or equal to 1000. All the rules with an information score of more than 0.2 are considered useful. This is clearly not a rule that will hold in general: only the size of the data set is considered, and factors such as the dimensionality and number of classes in the data set is ignored. The linear classifier used in StatLog is a model based classifier; generally the performance of the Linear classifier improves as the number of observations increases. This rule is thus valid for Statlog specifically, but does not generalize to other data sets.

It is therefore fair to say that limited understanding on the relationship between data properties and classifier performance is currently available, and it is against that background that we present our methods and results below.

3.   Methods and data

In order to experiment with the relationship between data and classifiers, we have generated several series of artificial data, and experimented with both model-based and discriminative classifiers. 10-fold cross-validation is used to evaluate and compare the performance of the classifiers on the different data sets.

3.1. Artificial data generation

Multivariate Gaussian distributions are used to generate artificial data sets. We use d single variable Gaussian distributions

$$p(x) = \frac{1}{(2\pi\sigma^2)^{\frac{1}{2}}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} \qquad (1)$$

to generate n samples per distribution. Note that μ is the mean and σ^2 is the variance of (1). This result in a $d \times n$ matrix $\mathbf{X}$. The d single dimensional variables are expanded to a multivariate Gaussian distribution by using

$$\mathbf{Y} = \mathbf{Ax} + \mathbf{B}, \qquad (2)$$

where $\mathbf{B}$ is the d-dimensional mean of the distribution repeated n times and $\mathbf{AA}^T$ is the covariance, $\mathbf{\Sigma}$, of the multivariate distribution. The consequent multivariable distribution may be written as

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}} |\mathbf{\Sigma}|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mathbf{\mu})^T \mathbf{\Sigma}^{-1}(\mathbf{x}-\mathbf{\mu})\right\} \qquad (3)$$

where $\mathbf{X}$ is a d-component column vector, $\mathbf{\mu}$ is a d-component mean vector, $\mathbf{\Sigma}$ is the d-by-d covariance matrix,

$(\mathbf{x}-\mathbf{\mu})^T$ is the transpose of $(\mathbf{x}-\mathbf{\mu})$, $\mathbf{\Sigma}^{-1}$ is the inverse of $\mathbf{\Sigma}$, and $|\mathbf{\Sigma}|$ is the determinant of $\mathbf{\Sigma}$.

A weighted mixture of multivariate Gaussian distributions of the form in (3) are used to generate class conditional probability density functions for each class in a data set.

Data sets were generated for three different experiments explained later in this section.

3.1.1.   Correlation of variables

Multivariate artificial data with class conditional probability density functions of the form given in (3) are generated for correlated and uncorrelated variables. A full covariace matrix $\mathbf{AA}^T$ is used for correlated variables. Diagonal $\mathbf{A}$ matrices are used to generate uncorrelated data. This results in diagonal covariance matrices $\mathbf{AA}^T$.

3.1.2.   Standard deviation

Data sets were generated with different standard deviations (SDs) for the class distributions. SD was introduced into the data by multiplying all the elements in the $\mathbf{A}$ matrix with the SD value.

3.1.3.   Noise

Two forms of noise are relevant in classification problems: *input* noise affects the class-conditional density functions, and can be adjusted by changing $\mathbf{A}$ and $\mathbf{B}$ in (2). *Output* noise is implemented by changing the class labels of the observations in the original data set. For reasons that will become clear, we are more interested in output noise; for this case, the percentage noise is measured by the percentage of class labels that have been changed.

3.2. Classifiers

Two model-based and four discriminative classifiers are used in this study. The model based classifiers are the Naïve Bayes classfier (NB) [6] and the Gaussian (Gauss) classifier. The discriminative classifiers are the Decision Tree classifier (DT) [7], the k-nearest-neighbour (kNN) classifier [8], the multi-layer perceptron (MLP) and support vector machines (SVMs) [9].

The NB, DT, kNN, MLP and SVM classifiers are all implementations of the machine learning package Weka [10]. The Gaussian classifier is a Matlab implementation available at [11].

The kNN uses a LinearNN nearest neighbour search algorithm with an Euclidean distance metric [8]. The optimal k value is determined by performing 10-fold cross-validation. An optimal k value between 1 and 10 is used for Experiments 1 and 3. Experiment 2 uses optimal k values between 1 and 20.

The SVM uses C-Support Vector classification with a radial basis function kernel. The optimal cost parameters, C, and gamma parameters, g, are determined by performing 10-fold cross validation. g values in the range $[10^{-7}, 10^6]$ and C values in the range $[10^{-3}, 10^3]$ are considered.

A single hidden-layer back-propagation MLP is used. The optimal number of nodes in the hidden layer is determined by 10-fold cross-validation. An optimal number of hidden nodes between 2 and 10 are used.

3.3. Experimental design

The three research questions introduced in Section 1 were studied through the design of targeted data sets for three experimental conditions, as described below. Note that all the experiments were repeated ten times on ten different data sets (with the same properties) to reduce the effect of variability in the results.

3.3.1.   Experiment 1

Experiment 1 uses artificial data sets with Gaussian distributed classes to illustrate where the Gaussian classifier and the Naïve Bayes classifier outperform discriminative classifiers.

   The method of data generation explained in this section is used. Artificial data sets for correlated and uncorrelated variables are generated. Data sets with three classes are used. The number of samples per class in each data set ranges from 20 to 100. All the data sets have ten variables. The SDs for each data set ranges from 1 to 25, and the class means are manually chosen to give well-separated means in the hypercube where each variable ranges from -5 to 5.

   The purpose of this experiment is to generate data sets with models that fit the Gaussian classifier and the Naïve Bayes classifier assumptions well. The Gaussian classifier assumes data with Gaussian distributions and dependent or correlated variables, whereas the naïve Bayesian classifier assumes independent variables of a particular one-dimensional distribution (for simplicity, we have employed Gaussian distributions for those cases as well). The number of samples per class is varied to probe for cases where the model-based assumption is optimally useful.

3.3.2.   Experiment 2

Experiment 2 uses artificial data sets with Gaussian distributed data and added output noise to illustrate the effect of output noise on the optimal value of k in the kNN classifier. The effect of output noise on the ratio between the error rate of the optimal kNN classifier and the error rate of the 1NN classifier is also illustrated.

   Two and ten dimensional correlated data sets with noise fractions ranging from 5-25 % are generated. All the data sets have three classes and the number of samples per class range from 20 to 100. The standard deviations of the distributions are varied from 1 to 25 to illustrate the effect of the SD on the optimal k values.

   The error rates for the optimal kNN classifier and the 1NN classifier are calculated for all the data sets and are compared.

3.3.3.   Experiment 3

Experiment 3 uses two-dimensional Gaussian distributed data with different SDs in the x and y directions. These data sets are used to illustrate the effect of the constant distance metric used by the kNN classifier throughout the entire variable space.

   Data sets are generated where the number of classes in a data set range from 2 to 4. The number of samples per class ranges from 20 to 100.

   The 10-fold cross-validation error rates for the model-based and discriminative classifiers are calculated and compared to the error rates of the optimal kNN classifier.

4.   Results

The results of the three experiments explained in Section 3 are summarized in this section. Throughout our discussion, high dimensional data is defined as data with a small number of samples in each class per dimension.

4.1. Experiment 1 results

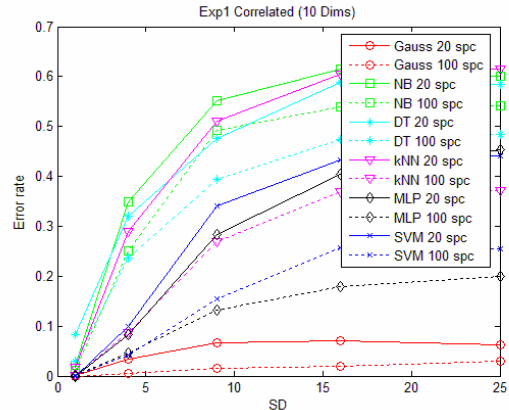The classification results from experiment 1 are given in Figures 1 and 2.

Figure 1: Classification results of correlated ten dimensional data sets.

Fig.1 shows that the Gaussian classifier achieves the lowest error rate over all the correlated data sets in this experiment. This result is not surprising in itself since the data are in fact normally distributed. The interesting results are contained in (1) the extent to which the discriminative classifiers underperform the appropriate model-based classifier, and (2) the dependence of this underperformance on factors such as data overlap and the size of the training set.

   We see that all the discriminative classifiers perform considerably worse than the model-based classifier in the ten dimensional space. It is thus not safe to assume that discriminative classifiers will perform comparable to a Gaussian classifier on correlated high dimensional data with a small number of samples per class.
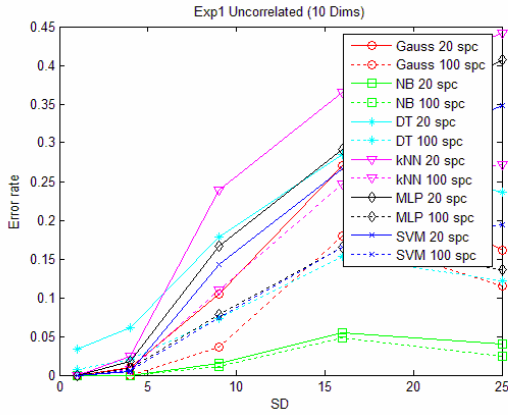
Figure 2: Classification results of uncorrelated ten dimensional data sets.

Fig. 2 shows that the NB classifier has the lowest error rate over all the uncorrelated data sets used in this experiment, whereas all the other classifiers had substantial error rates for at least some experimental conditions. It is thus not safe to assume that discriminative classifiers will perform comparable to a Naïve Bayes classifier on uncorrelated high dimensional data with a small number of samples per class.

4.2. Experiment 2 results
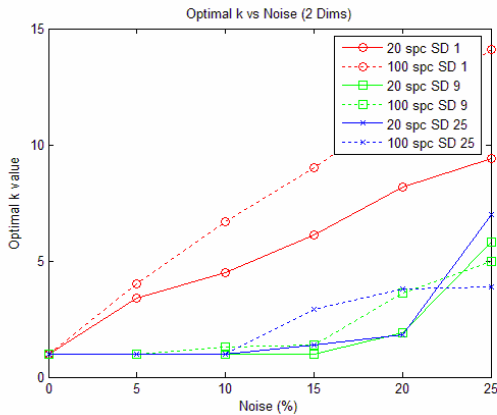
The results of experiment 2 are given in Figures 3-6.

Figure 3: Classification results of correlated noisy two dimensional data sets.

Fig. 3 shows that the optimal value of k for the kNN classifier increases monotonically as the (output) noise in the data increases, whereas the optimal k value seems to decrease (though not as predictably) when the SD increases. At first glance these results seem contradictory, since the SD can also be viewed as a form of noise – specifically, input noise. However, these results are actually consistent, and provide an important hint on the choice of k: whereas increasing SD does create increasing overlap of the class distributions, that overlap tends to lie at the edges of these distributions. Output noise, on the other hand, permeates the entire feature space – hence, a larger k value is required to properly smooth over these samples as the noise percentage increases.
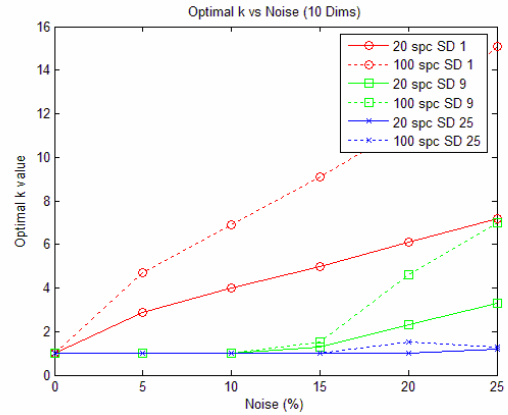

Figure 4: Classification results of correlated noisy ten dimensional data sets.

Fig. 4 shows that in a high-dimensional feature space with high SD, the optimal k values are close to 1. This might be because, for large k, the contributing samples may be so far away from the sample as to be meaningless. Fig. 4 also shows that the optimal k value continues to increase reasonably monotonically with the noise percentage in 10 dimensions both for high and low overlap.
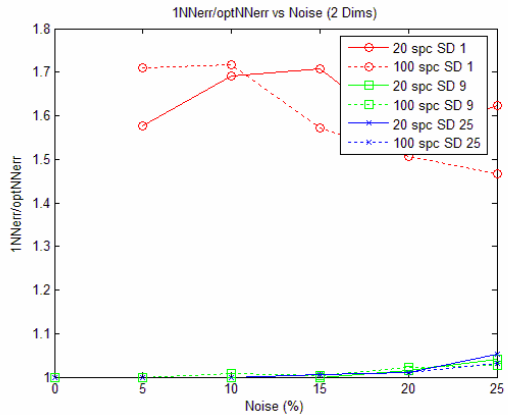

Figure 5: Classification results of correlated noisy two dimensional data sets.

How significant are the differences between the accuracies obtained with the various values for k? Figs. 5 and 6 show the error rates of the 1NN classifier divided by those of the optimal kNN classifier for each of the cases corresponding to Figs. 3 and 4. In the vast majority of cases, for a SD of 1, the 1NN classifier has more than 1.5 times the error rate of the optimal classifier, indicating that these differences are indeed significant.
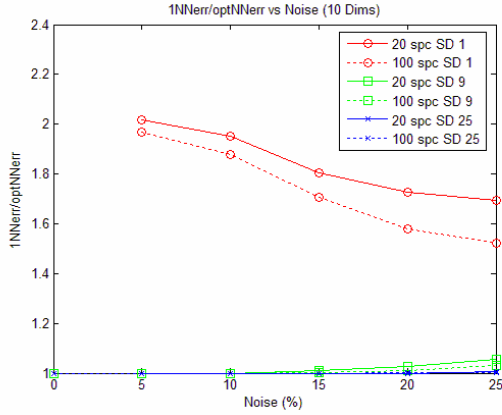
Figure 6: Classification results of correlated noisy ten dimensional data sets.

4.3. Experiment 3 results

The results of experiment 3 are summarized in Tables 1 and 2. The highest error rate for each data set is in bold.

Table 1: Classification results of model-based and DT classifiers

Classes	SPC	NB	Gauss	DT
2	20	0.03	0.07	0.0625
	40	0.0275	0.05	0.04375
	60	0.028333	0.037045	0.041667
	80	0.026875	0.033417	0.03375
	100	0.0285	0.036342	0.0375
3	20	0.031667	0.057667	0.056667
	40	0.030833	0.050909	0.040833
	60	0.024444	0.042222	0.035
	80	0.025	0.043279	0.030833
	100	0.026	0.040943	0.029333
4	20	0.02875	0.070179	0.06625
	40	0.03	0.053083	0.04125
	60	0.02625	0.043261	0.04125
	80	0.02625	0.043689	0.04125
	100	0.02625	0.044705	0.03975

Table 2: Classification results of remaining discriminative classifiers

Classes	SPC	kNN	MLP	SVM
2	20	**0.07**	0.0575	0.045
	40	**0.05**	0.04125	0.0375
	60	**0.045833**	0.0375	0.036667
	80	**0.04125**	0.029375	0.030625
	100	**0.042**	0.033	0.031
3	20	**0.14667**	0.055	0.035
	40	**0.1**	0.038333	0.026667
	60	**0.073333**	0.035556	0.025556
	80	**0.06375**	0.032917	0.025417
	100	**0.056667**	0.034667	0.027333
4	20	**0.155**	0.05875	0.045
	40	**0.10625**	0.049375	0.03375
	60	**0.090417**	0.04875	0.032917
	80	**0.08125**	0.045625	0.031563
	100	**0.07375**	0.04475	0.037

Two dimensional data sets with uncorrelated class-conditional densities were used. The SD of the classes in the data sets were different in the x and y directions. Figure 7 is a scatter plot of the four-class data set with 100 samples per class that was used.

Fig. 7 shows that the classes marked with 'x','o' and '+' all have the same SD in the vertical (*y*) direction but have different SD's in the horizontal (*x*) direction. The class marked by '.' has a very large SD in the *x* direction and a very small SD in the *y* direction.

Tables 1 and 2 show the high error rates of the kNN classifier. The tables also show that the classification results become worse, compared to the other classifiers, when the number of classes increases.
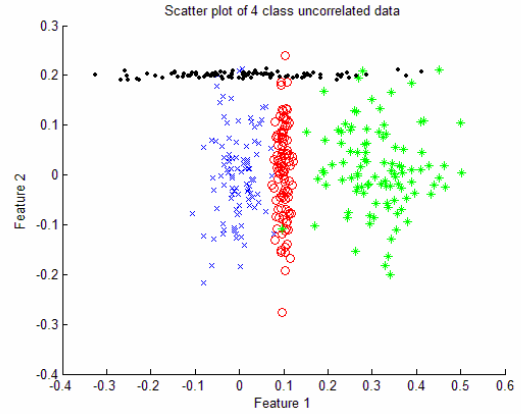

Figure 7: Scatter plot of 4 class 100 samples/class 2 dimensional data set.

The kNN uses a constant distance metric throughout the entire variable space. Due to the different SDs of the classes in different directions, the distance metric should ideally adapt to each sub-space covered by a different class. Because the distance metric remains constant, samples are misclassified in the regions where different classes overlap or are close together.

5. Conclusions

We have studied two examples where "conventional wisdom" is concretely shown to be incorrect – namely, classification problems where model-based classifiers outperform several discriminative classifiers by a wide margin, and other classification problems where kNN classifiers, even with optimized k, perform poorly in comparison with the other classifiers studied.

These examples contain a number of lessons relevant to the particular classifiers studied here, and also suggest some more general conclusions regarding classifiers. Thus, we have seen that kNN classifiers are best employed in cases where the "natural" metric is fairly constant throughout feature space, and that the optimal value for *k* depends on the effective output noise, rather than the input noise (which produces a different form of class overlap). We have also seen that model-based classifiers are a viable alternative to discriminative classifiers when the amount of training data is

severely limited (relative to the dimensionality of the feature space), and the parametric form of the assumed model is a sufficiently good fit for the actual data distribution.

More generally, our results show how certain properties of the data (e.g. spatial variability of the natural metric) can influence even non-parametric classifiers in much the same way that the parametric fit can influence the performance of parametric classifiers.

It remains an intriguing challenge to fully describe these relationships between data characteristics and classifier behaviour, and to develop algorithms that automatically select classifiers and parameters appropriate for a given set or subset of data.

6. Acknowledgements

7. References

[1] D. Michie and D.J Spiegelhalter and C.C. Taylor, "Machine learning, neural and statistical classification," Hemel Hempstead: Ellis Horwood Limited, 1994.

[2] A.K. Jain, R.P.W. Duin and J. Mao, "Statistical Pattern Recognition: A Review", *in IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 1, pp. 4-37, 2000.

[3] A.R. Webb, "Statistical pattern recognition," 2nd ed., New York: John Wiley, 2002.

[4] D.M.J. Tax and R.P.W. Duin, "Characterizing one-class datasets," *in Proceedings of the Sixteenth Annual Symposium of the Pattern Recognition Association of South Africa*, pp.21-26, 2005.

[5] P. Brazdil and J. Gama and B. Henery, "Characterizing the applicability of classification algorithms using meta-level learning," *in Proceedings of the European Conference on Machine Learning*, pp. 83-102, 1994.

[6] G.H. John and P. Langley, "Estimating Continuous Distributions in Bayesian Classifiers," *in Eleventh Conference on Uncertainty in Artificial Intelligence*, San Mateo, pp. 338-345, 1995.

[7] R. Quinlan, "C4.5: Programs for Machine Learning," Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[8] D. Aha and D. Kibler, "Instance-based learning algorithms," *in Machine Learning*, Vol. 6, pp.37-66, 1991.

[9] C. Chang and C. Lin, "LIBSVM: a library for support vector machines", 2001. [Online] Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm

[10] I.H. Witten and E. Frank, "Data Mining: Practical machine learning tools and techniques," 2nd Edition, Morgan Kaufmann, San Francisco, 2005. [Online] Available: http://www.cs.waikato.ac.nz/~ml/weka/

[11] Yu Hen Hu, "Maximum likelihood classifier with Gaussian mixture distribution", 2001. [Online] Available: http://homepages.cae.wisc.edu/~ece539/matlab/