Facial Expression Classification using Kernel Principal Component Analysis and Support Vector Machines

J. Ghent
Computer Vision and Imaging Laboratory
Department of Computer Science
NUI Maynooth
jghent@cs.nuim.ie

J. Reilly
Computer Vision and Imaging Laboratory
Department of Computer Science
NUI Maynooth
jreilly@cs.nuim.ie

J. McDonald
Computer Vision and Imaging Laboratory
Department of Computer Science
NUI Maynooth
johnmcd@cs.nuim.ie

Abstract

This paper details a novel procedure for accurately classifying lower facial expressions. A shape model is developed based on an anatomical analysis of facial expression called the *Facial Action Coding System* (FACS). This model analyzes the movement in shape due to the formation of a specific expression. We apply *Kernel Principal Component Analysis* (KPCA) to the shapes in the training set and classify new unseen expressions by using *Support Vector Machines* (SVMs). We further analyse our model by attaching a probability measure to the outputs.

Keywords: Facial expression classification, *Kernel Principal Component Analysis* (KPCA), *Support Vector Machines* (SVMs).

1 Introduction

Of all the human senses, vision is the most informative with the majority of activity in our brain being concerned with visual processing. One of the most interesting and difficult visual processing tasks is facial image analysis. Of major importance here is the classification of facial expressions.

This paper details a technique that enables a computer to classify specific changes to the shape of a mouth. The approach taken employs psychological tools, computer vision techniques, and machine learning algorithms. We construct a training set such that every image in the training set depicts desired expressions. These expressions are anatomically analysed using a system for measuring expression called the *Facial Action Coding System* (FACS) [Ekman et al., 1978]. The expression we classify in this paper are all lower facial expressions and therefore only the shape of the mouth is analysed. A model is developed based on the shape of each mouth in the training set using KPCA.

We use the outputs of the shape model to train an SVM to classify an observed expression. SVMs are a new generation of learning system based on recent advances in statistical learning theory [Campbell, 2002, ScholKopf and Smola, 2002, Rogers et al., 2004, Rogers, 2004]. SVMs deliver state-of-the-art performance in real-world applications such as text categorisation, hand-written character recognition, image classification and bioinformatics [Guyon, 2006]. SVMs are based on a combination of techniques. A principal idea behind SVMs is the *kernel trick*, where data is transformed into a high-dimensional space making linear discriminant functions practical. SVMs also use the idea of *large margin classifiers*. This ensures the hyperplane is positioned in an optimal location seperating the two classes.

Considerable literature on facial expression classification exists. Techniques range from template based methods [Lyons et al., 1999], to neural network based methods [Er et al., 2002], or a combination of the two [Pantic and Rothkrantz, 2000, Ghent and McDonald, 2005b]. However, perhaps the most substantial work in this area has been done by Bartlett *et al.* Bartlett proposes a technique which combines Gabor wavelets and SVMs to classify *Action Units* (AUs) with 93.3% accuracy [Bartlett et al., 2004, Bartlett et al., 2003]. Again in [Littlewort et al., 2004], Littlewort and Bartlett propose a similar technique which classifies AUs with 97% accuracy. In [Littlewort-Ford et al., 2001], Bartlett used SVMs again to successfully distinguish between genuine and fake smiles.

More relevantly, Bartlett employed linear SVMs with PCA to classify facial actions with 75% accuracy [Littlewort et al., 2006] and concluded that there existed an incompatibility between PCA and SVMs for facial expression classification [Bartlett et al., 2005]. This approach performed PCA on Gabor wavelets of images from the training set prior to applying the SVM. We hypothesised that the non-linear nature of facial expression prohibited higher classification accuracy in [Bartlett et al., 2005] using this method. With this in mind we propose a technique that uses KPCA in conjunction with SVMs to classify facial expressions.

The rest of this paper is structured as follows: Section 2 documents our approach, section 3 details experiments and results, and we conclude with some final remarks. This paper extends our previous work detailed in [Ghent and McDonald, 2005b, Ghent and McDonald, 2005a] and [Ghent, 2005].

2 Proposed Methodology

Measuring facial expressions is a non-trivial task as everyone's face is unique. Several methods have been proposed, however, the technique we use must measure expression consistently independent of identity. In this paper we use the *Facial Action Coding System* (FACS), which measures expression by the movement of muscles in the face. This system is based on an anatomical analysis of facial expressions. A movement of a muscle or in some cases a group of muscles is known as an *Action Unit* (AU). All expressions can be described using the AUs defined by the FACS. The FACS allows us to subdivide our training data into subsets where the variation in each subset is precisely characterised. This provides the basis for accurate classification of expression independent of subject. We use FACS coded images to build a statistical model of shape based on point distribution.

2.1 KPCA

We label every image in the training set with a set of landmark points. These points are located around key areas such as the eyes, nose, mouth and eyebrows. The mean shape of the face is calculated and every image is aligned to the mean shape using *Generalised Procrustes Alignment* (GPA)[Gower, 1975]. This technique aligns two shapes with respect to translation, rotation and scale by minimising the weighted sum of the squared distances between the corresponding landmark points. The aligned landmark points are analysed using *Kernel Principal Component Analysis* (KPCA). This technique is similar to standard PCA except the data is projected into a higher dimensional feature space prior to performing eigenvector decomposition.

We project the data into feature space through the use of the *kernel trick*. This *kernel trick* permits the computation of dot products in high dimensional *feature spaces*, using functions defined on pairs of input patterns.

More specifically, mapping from one space to a higher dimensional space involves a mapping from $\mathbf{x}_i \longrightarrow \phi(\mathbf{x}_i)$, however, with an appropriate choice of kernel there exists a mapping ϕ such that

$$(\phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)) = K(\mathbf{x}_i, \mathbf{x}_j). \tag{1}$$

This means that the inner products of the feature space can be calculated without computing $\phi(x)$ directly. This allows us to work in an extremely high dimensional feature space. The choice of kernel is still a matter of debate, however, in this paper we use a *Gaussian* kernel. The Gaussian kernel is defined as

$$K(\mathbf{x}_i, \mathbf{x}_j) = e^{-(\mathbf{x}_i-\mathbf{x}_j)^T(\mathbf{x}_i-\mathbf{x}_j)/2\sigma^2}. \tag{2}$$

The difference between KPCA and PCA is illustrated in Figure 1. It can be seen from this figure that the first principal component clusters the data using KPCA while standard PCA illustrates the most significant mode of variation.

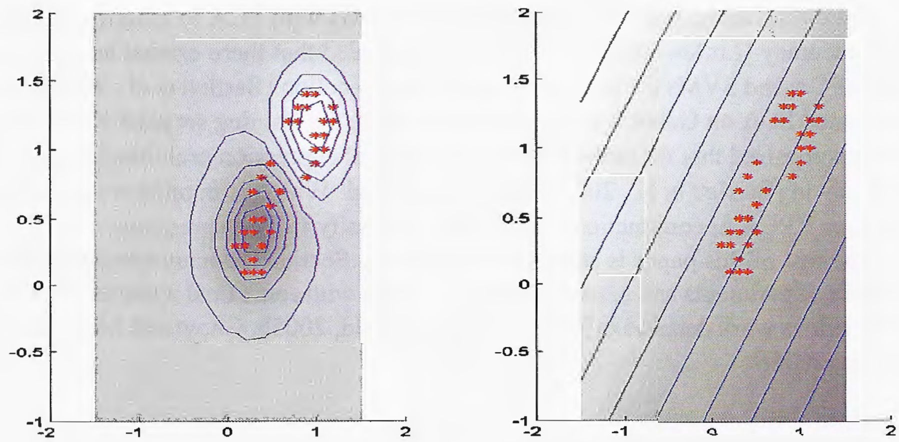

Figure 1: *An comparison using sample data between KCPA and PCA. KPCA is shown to the left of this figure and PCA is shown on the right.*

2.2 Support Vector Machines

The SVM algorithm can be separated into two distinct procedures, the *kernel method*, which we have already discussed, and the *base algorithm*. Suppose we have a dataset $(x_1, y_1), ..., (x_m, y_m) \in \mathbf{X} \times \{\pm 1\}$ where $\mathbf{X}$ is some space from which the x_i have been sampled. We can construct a dual Lagrangian of the form

$$W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \cdot \mathbf{x}_j) \tag{3}$$

which are subject to the constraints

$$\alpha_i \geq 0 \quad \forall i \qquad \mathit{and} \qquad \sum_{i=1}^{m} \alpha_i y_i = 0. \tag{4}$$

Further details of the construction of this equation can be found in [Ghent, 2005]. The solution to Equation 3 is a set of α values which are used in the decision function

$$f(\mathbf{z}) = sign\left(\sum_{i=1}^{m} y_i \alpha_i (\mathbf{x}_i \cdot \mathbf{z}) + b\right) \tag{5}$$

here $\mathbf{z}$ is an input and b is the bias. The resulting α_i values that are non-zero correspond to the support vectors. If $\alpha_i = 0$ then these points make no contribution to the decision function. The value of each α_i also carries information about the importance of particular datapoints in the training set. This insight can be utilised to deal with outliers and erroneous datapoints [Campbell, 2002]. Imposing a box constraint on the α's can limit the effect of outlying input data. The box constraint is given by

$$C \geq \alpha_i \geq 0 \tag{6}$$

where C is known as the soft margin parameter. The value of C is set using a standard optimisation approach, details can be found in [Ghent, 2005].

2.3 Measure of Confidence

It is possible to extract probabilities from SVM outputs which can be used as a post processing tool for classification problems. An SVM has a confidence measure which is inherent in the technique. The further a test point is from the separating hyperplane the greater the degree of confidence should be in the classification of that point. This distance can be mapped to a probability using a technique devised by Platt [Platt, 1999]. We use a parametric model to fit the posterior probability $P(y = 1|f)$ directly. The parametric function is

$$P(y = 1|f) = \frac{1}{1 + exp(Af + B)}. \tag{7}$$

A and B can be found from the training set by minimising the negative log likelihood function

$$min \left[-\sum_i t_i log(p_i) + (1 - t_i) log(1 - p_i) \right] \tag{8}$$

where p_i is (7) evaluated at f_i (the real value output of input i). This is minimised using a Levenberg-Marquardt algorithm. Once the sigmoid is found using the training set we can calculate the probability an unseen shape has of belonging to the class in question.

3 Experiments and Results

In this paper we classify AU's associated with the mouth. We classify four expressions; AU20+AU25, AU12, AU10+20+25 and AU25+AU27. The effect each of these AUs have on the mouth is illustrated in Table 1.

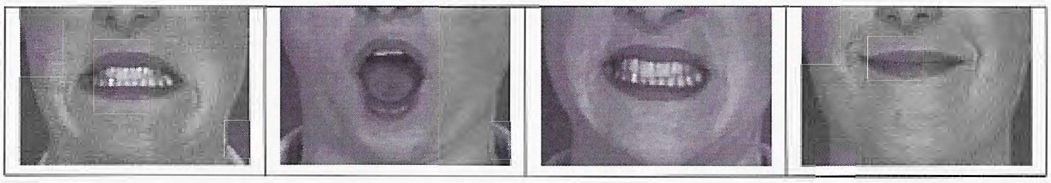

Table 1: *This table illustrates the effect of portraying four different expressions. The AUs portrayed, from left to right are; AU20+25, AU25+27, AU10+20+25, and AU12,*

We calculate our shape space by performing KPCA on the training data, as outlined in Section 2.1. The training data consists of just one subject performing the four desired AUs we wish to classify. We project new unseen data to be classified into the shape space and use these outputs as inputs to the SVM classifier. As there exists four expressions to be separated, the one-against all approach yields four separate SVM classifiers. This approach requires, at most, four evaluations to acquire a result. The results from the one-against-all approach are detailed in Table 2.

In Table 2 NTs is the number of test shapes, C is the soft margin variable, σ is the kernel parameter, and Ts is the percentage of correctly classified test data.

AU	NTs	C	σ	Ts
A-v-all	116	0.2	0.1	82.756
B-v-all	116	0.9	0.5	91.3794
C-v-all	147	0.1	0.2	93.8776
D-v-all	147	0.1	0.2	93.1972
Average				90.01

Table 2: *This table details the results from a one-against-all approach to classifying four multiple AU expressions. In the table above A = AU20+AU25, B = AU25+AU27, C = AU10+AU20+AU25 and D = AU12.*

It can be seen from Table 2 that a one-against-all SVM classifies four primary facial expression with an average accuracy of 90.01%. This is an encouraging result for three main reasons. Firstly, there exists a large amount of variance in the training set which would complicate the separation task. Secondly, the test data is completely unseen from the expression space i.e the test data was not used in calculating the expression space. This means that there exists enough variance in the expression space to accurately describe unseen shapes of individuals. And thirdly, there is significant overlap between the expressions we wish to classify, for example, it can be seen from Table 1 that two of the expression are extremely similar, this makes the separation task significantly more difficult.

Unfortunately, there exists no human classification baseline data of these AU's to compare our systems performance with. However, Bartlett has shown that naive human subjects classify single AU's with an accuracy of 77.9% while expert FACS coders classify AU's with an accuracy of 94.1% [Bartlett et al., 2003]. Naive subjects were provided with a guide sheet of the AU's which depicted examples of each AU and were also provided with written descriptions of each AU. Furthermore, alternative techniques for classifying multiple AUs achieve results of 83.34% [Abboud et al., 2004] and 86.0% [Michel and Kaliouby, 2003].

We extend our approach by incorporating a confidence measure associated with each new unseen shape. This information can be used to provide a measure of how confident we are that an unseen input belongs to a particular class. For example, in Figure 2 we input shapes into a probability function designed to recognise AU20+AU25. Each subject's expressions range from neutral to AU20+AU25. At neutral, represented by 1 on the *x-axis*, there is a low probability of the shape belonging to class AU20+AU25, however, once the expression is formed the likelihood of that shape belonging to class AU20+AU25 increases significantly. This probability measure makes no inference as to the intensity of the expression in question.

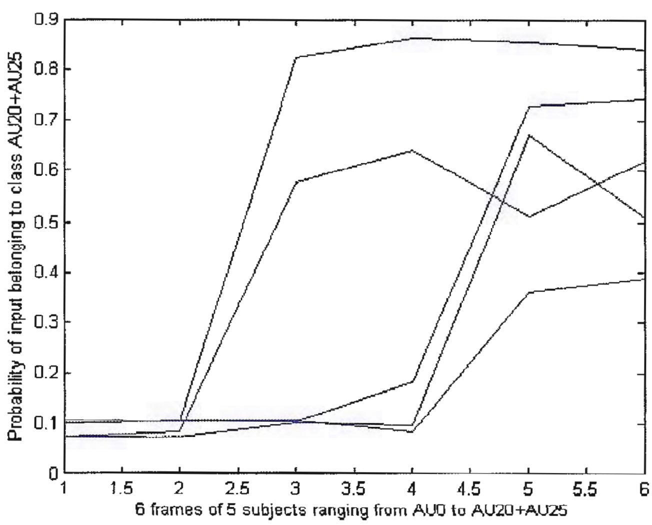

Figure 2: *The probability of a sequence of shapes belonging to one class*

This property of our approach is emphasized in Figure 3, here, 18 shapes are passed into four probability functions, each designed to measure the likelihood of an input belonging to a particular class. The input to this experiment was 18 shapes ranging from low intensity AU12 to high intensity AU12. As the diagram shows, the probability of the inputs belonging to class AU12 is greater than the probability of the same inputs belonging to any other class. It should also be noted that there is no significant difference between the likelihood of a low intensity example of AU12 and the likelihood of a high intensity example of AU12. The reason for this is that a high intensity expression is not necessarily at a greater distance from the separating hyperplane in an SVM.

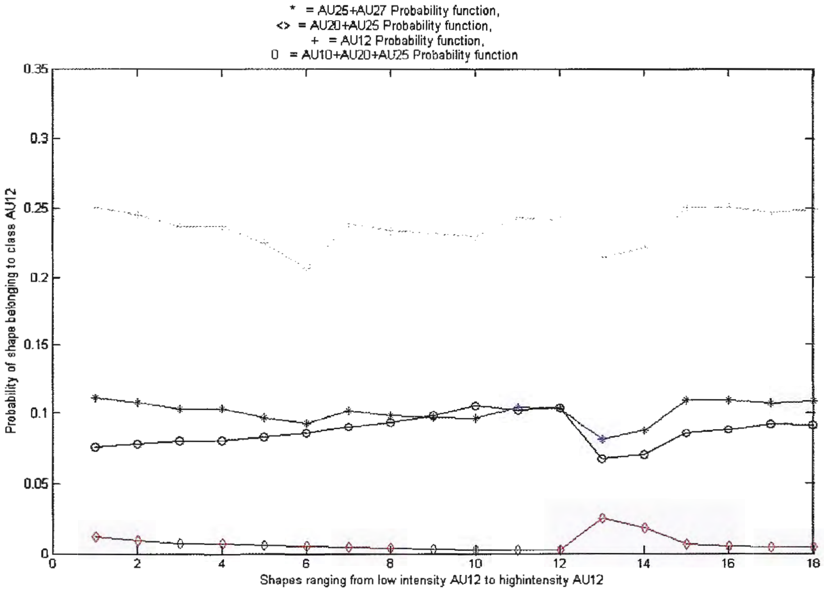

Figure 3: *The probability 18 inputs ranging from low intensity AU12 to high intensity AU12 belonging to a specific class.*

As can be seen from Figure 4, the confidence measure can also be used to aid in the classification process. This figure shows the probability of an input belonging to class AU20+AU25. The input data in this experiment is a sequence of extreme examples of AU20+AU25 as shown by several subjects. It should also be noted some of the inputs are classified as belonging to class AU12. This attribute again suggests that the most extreme expression is not necessarily going to return the highest probability of an input belonging to a particular class.

4 Conclusion

The accurate classification of facial expressions is a growing problem within several domains. The solution described in this paper takes a multidisciplinary approach drawing together psychological tools, statistical models and machine learning techniques. We first built a shape model that was based on an anatomical analysis of facial expression (FACS). The FACS provided us with a universal method of analyzing facial expression and allowed for the classification of facial expressions independent of subject (age, sex, skin, colour, etc.).

The shape model was calculated by using KPCA to lower the dimensionality of the problem. A one-against-all SVM was used to classify four expressions (AU20+AU25, AU25+AU27, AU10+AU20+AU25 and AU12). A one-against-all SVM classified multiple AU's with an average of 90% accuracy. A Gaussian kernel was used in each SVM and the value of the Gaussian (σ) and the soft margin parameter (C) were calculated using cross validation. Finally the data was further analysed by extracting probabilities from the outputs of the SVM's and establishing a confidence measure.

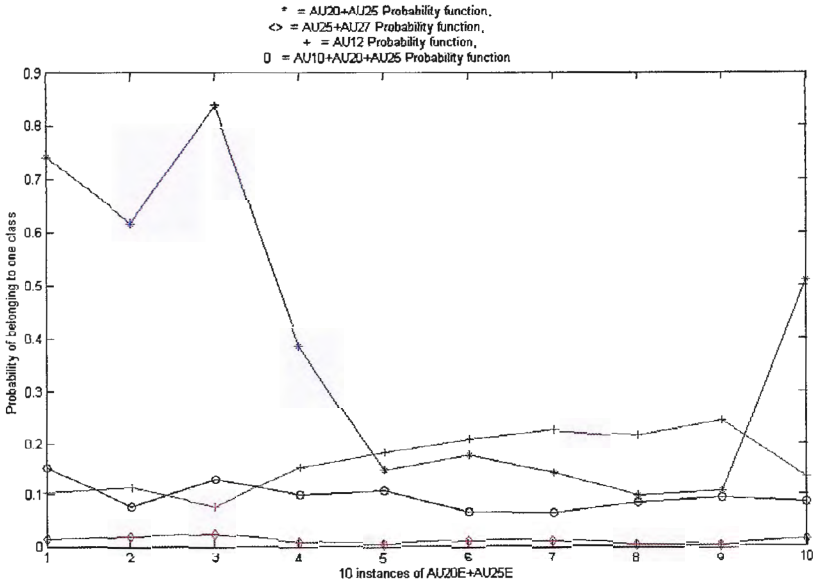

Figure 4: *The probability of 10 inputs belonging to a specific class. The inputs in this experiment all represent shapes portraying extreme examples of AU20+AU25.*

4.1 Acknowledgements

This publication has emanated from research conducted with the financial support of Science Foundation Ireland.

References

[Abboud et al., 2004] Abboud, B., Davoine, F., and Dang, M. (2004). Facial expression recognition and synthesis based on an appearance model. *Signal Processing: Image Communication, ELSEVIER.*

[Bartlett et al., 2005] Bartlett, M. S., Littlewort, G., Frank, M., lainscsek, C., Fasel, I., and Movellan, J. (2005). Recognising facial expression: Machine learning and application to spontaneous behaviour. *IEEE International conference on computer vision and pattern recognition.*

[Bartlett et al., 2004] Bartlett, M. S., Littlewort, G., Lainscsek, C., Fasel, I., and Movellan, J. (2004). Machine learning methods for fully automatic recognition of facial expressions and facial actions. *IEEE International conference on systems, man and cybernetics*, pages 592–597.

[Bartlett et al., 2003] Bartlett, M. S., Movellan, J., Littlewort, G., Braathen, B., Frank, M. G., and Sejnowski, T. J. (2003). Towards automatic recognition of spontaneous facial actions. *In Paul Ekman, editor, what the face reveals.* Oxford University Press.

[Campbell, 2002] Campbell, C. (2002). kernel methods: A survey of current techniques. *Neurocomputing*, 48:63–84.

[Ekman et al., 1978] Ekman, P., Friesen, W., and Hager, J. (1978). Facial action coding system. *Consulting Psychologists Press.*

[Er et al., 2002] Er, M. J., Wu, S., Lu, J., and Toh, H. L. (2002). Face recognition with radial basis function neural networks. *IEEE transactions on neural networks*, 13(3):697–710.

[Ghent, 2005] Ghent, J. (2005). *A Computational Model of Facial Expression.* PhD thesis, National University of Ireland Maynooth, Co. kildare, Ireland.

[Ghent and McDonald, 2005a] Ghent, J. and McDonald, J. (2005a). Facial expression classification using a one-against-all support vector machine. *Proceedings of the Irish machine vision and image processing conference.*

[Ghent and McDonald, 2005b] Ghent, J. and McDonald, J. (2005b). Holistic facial expression classification. *Opto-Ireland.*

[Gower, 1975] Gower, J. C. (1975). Generalised procrustes analysis. *Psychometrika*, 40:33–50.

[Guyon, 2006] Guyon, I. (2006). Svm application list. http://www.clopinet.com/isabelle/Projects/SVM/applist.html.

[Littlewort et al., 2004] Littlewort, G., Bartlett, M. S., andJ. Chenu, I. F., Kanda, T., Ishiguro, H., and Movellan, J. (2004). Towards social robots: automatic evaluation of human-robot interaction by face detection and expression classification. *Advances in Neural Information Processing Systems*, 16:1563–1570.

[Littlewort et al., 2006] Littlewort, G., Bartlett, M. S., Fasel, I., Susskind, J., and Movellan, J. (2006). Dynamics of facial expression extracted automatically from video. *Computer vision and Image understanding.*

[Littlewort-Ford et al., 2001] Littlewort-Ford, G., Bartlett, M. S., and Movellan, J. R. (2001). Are your eyes smiling? detecting genuine smiles with support vector machines and gabor wavelets. *Proceedings of the 8th annual joint symposium on neural computation.*

[Lyons et al., 1999] Lyons, M. J., Budyek, J., and Akamatsu, S. (1999). Automatic classification of single facial expressions. *IEEE transactions on pattern analysis and machine intelligence*, 21(12):1357–1362.

[Michel and Kaliouby, 2003] Michel, P. and Kaliouby, R. E. (2003). Real time facial expression recognition in video using support vector machines. *Proceedings of HCI International Conference.*

[Pantic and Rothkrantz, 2000] Pantic, M. and Rothkrantz, L. J. M. (2000). Automatic analysis of facial expressions: the sate of the art. *IEEE transactions on pattern analysis and machine learning*, 22(12).

[Platt, 1999] Platt, J. C. (1999). Probabilistic outputs for support vector machines and comparisons to regularised likelihood methods. *In Alexander J. Smola, Peter Bartlett, Scholkopf Bernhard, and Dale Schuurmans, editors, Advances in large margin Classifiers.*

[Rogers, 2004] Rogers, S. (2004). *Machine Learning Techniques for Microarray Analysis.* Faculty of engineering mathematics, University of Bristol.

[Rogers et al., 2004] Rogers, S., Williams, R. D., and Campbell, C. (2004). *BioInformatics with computational intelligence paradigms*, chapter Class prediction with Microarray Datasets. Springer-Verlag.

[ScholKopf and Smola, 2002] ScholKopf, B. and Smola, A. J. (2002). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* MIT Press.