Expert Systems With Applications 79 (2017) 44–52

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Linear classifier design under heteroscedasticity in Linear Discriminant Analysis

Kojo Sarfo Gyamfi*, James Brusey, Andrew Hunt, Elena Gaura

Faculty of Engineering and Computing, Coventry University, Coventry, CV1 5FB, United Kingdom

ARTICLE INFO

Article history:
Received 12 November 2016
Revised 23 February 2017
Accepted 24 February 2017
Available online 24 February 2017

Keywords:
LDA
Heteroscedasticity
Bayes error
Linear classifier

ABSTRACT

Under normality and homoscedasticity assumptions, Linear Discriminant Analysis (LDA) is known to be optimal in terms of minimising the Bayes error for binary classification. In the heteroscedastic case, LDA is not guaranteed to minimise this error. Assuming heteroscedasticity, we derive a linear classifier, the Gaussian Linear Discriminant (GLD), that directly minimises the Bayes error for binary classification. In addition, we also propose a local neighbourhood search (LNS) algorithm to obtain a more robust classifier if the data is known to have a non-normal distribution. We evaluate the proposed classifiers on two artificial and ten real-world datasets that cut across a wide range of application areas including handwriting recognition, medical diagnosis and remote sensing, and then compare our algorithm against existing LDA approaches and other linear classifiers. The GLD is shown to outperform the original LDA procedure in terms of the classification accuracy under heteroscedasticity. While it compares favourably with other existing heteroscedastic LDA approaches, the GLD requires as much as 60 times lower training time on some datasets. Our comparison with the support vector machine (SVM) also shows that, the GLD, together with the LNS, requires as much as 150 times lower training time to achieve an equivalent classification accuracy on some of the datasets. Thus, our algorithms can provide a cheap and reliable option for classification in a lot of expert systems.

1. Introduction

In many applications one encounters the need to classify a given object under one of a number of distinct groups or classes based on a set of features known as the feature vector. A typical example is the task of classifying a machine part under one of a number of health states. Other applications that involve classification include face detection, object recognition, medical diagnosis, credit card fraud prediction and machine fault diagnosis.

A common treatment of such classification problems is to model the conditional density functions of the feature vector (Ng & Jordan, 2002). Then, the most likely class to which a feature vector belongs can be chosen as the class that maximises the a posteriori probability of the feature vector. This is known as the maximum a posteriori (MAP) decision rule.

Let K be the number of classes, $\mathcal{C}_k$ be the kth class, $\mathbf{x}$ be a feature vector and $\mathcal{D}_k$ be training samples belonging to the kth class ($k \in \{1, 2, \ldots, K\}$). The MAP decision rule for the classification task is then to choose the most likely class of $\mathbf{x}$, $\mathcal{C}^*(\mathbf{x})$ given as:

$$\mathcal{C}^*(\mathbf{x}) = \arg\max_{\mathcal{C}_k} p(\mathcal{C}_k|\mathbf{x}), \quad k \in \{1, 2, \ldots, K\} \tag{1}$$

We assume for the moment that there are only $K = 2$ classes, i.e. binary classification (we consider multi-class classification in a later section). Then, using Bayes' rule, the two posterior probabilities can be expressed as:

$$p(\mathcal{C}_k|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_k) \times p(\mathcal{C}_k)}{p(\mathbf{x})}, \quad k \in \{1, 2\} \tag{2}$$

It is often the case that the prior probabilities $p(\mathcal{C}_1)$ and $p(\mathcal{C}_2)$ are known, or else they may be estimable from the relative frequencies of $\mathcal{D}_1$ and $\mathcal{D}_2$ in $\mathcal{D}$ where $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2$. Let these priors be given by π_1 and π_2 respectively for class $\mathcal{C}_1$ and $\mathcal{C}_2$. Then, the likelihood ratio defined as:

$$\lambda(\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{C}_1)}{p(\mathbf{x}|\mathcal{C}_2)} \tag{3}$$

* Corresponding author.
E-mail addresses: gyamfik@uni.coventry.ac.uk (K.S. Gyamfi), j.brusey@coventry.ac.uk (J. Brusey), ab8187@coventry.ac.uk (A. Hunt), csx216@coventry.ac.uk (E. Gaura).

http://dx.doi.org/10.1016/j.eswa.2017.02.039

is compared against a threshold defined as $\tau = \pi_2/\pi_1$ so that one decides on class $\mathcal{C}_1$ if $\lambda(\mathbf{x}) \geq \tau$ and class $\mathcal{C}_2$ otherwise.

Linear Discriminant Analysis (LDA) proceeds from here with two basic assumptions (Izenman, 2009, Chapter 8):

1. The conditional probabilities $p(\mathbf{x}|\mathcal{C}_1)$ and $p(\mathbf{x}|\mathcal{C}_2)$ have multivariate normal distributions.
2. The two classes have equal covariance matrices, an assumption known as homoscedasticity.

Let $\bar{\mathbf{x}}_1$, $\mathbf{\Sigma}_1$ be the mean and covariance matrix of $\mathcal{D}_1$ and $\bar{\mathbf{x}}_2$, $\mathbf{\Sigma}_2$ be the mean and covariance of $\mathcal{D}_2$ respectively. Then, for $k \in \{1, 2\}$,

$$p(\mathbf{x}|\mathcal{C}_k) = \frac{1}{\sqrt{(2\pi)^d \det(\mathbf{\Sigma}_k)}} \exp\left[-\frac{1}{2}(\mathbf{x} - \bar{\mathbf{x}}_k)^T \mathbf{\Sigma}_k^{-1} (\mathbf{x} - \bar{\mathbf{x}}_k) \right] \tag{4}$$

where d is the dimensionality of $\mathcal{X}$, which is the feature space of $\mathbf{x}$. Given the above definitions of the conditional probabilities, one may obtain a log-likelihood ratio given as:

$$\ln \lambda(\mathbf{x}) = \frac{1}{2} \ln \frac{\det \mathbf{\Sigma}_2}{\det \mathbf{\Sigma}_1} + \frac{1}{2}\left[(\mathbf{x}-\bar{\mathbf{x}}_2)^T \mathbf{\Sigma}_2^{-1} (\mathbf{x}-\bar{\mathbf{x}}_2) - (\mathbf{x}-\bar{\mathbf{x}}_1)^T \mathbf{\Sigma}_1^{-1} (\mathbf{x} - \bar{\mathbf{x}}_1) \right] \tag{5}$$

which is then compared against $\ln \tau$ so that $\mathcal{C}_1$ is chosen if $\ln \lambda(\mathbf{x}) \geq \ln \tau$, and $\mathcal{C}_2$ otherwise. Thus, the decision rule for classifying a vector $\mathbf{x}$ under class $\mathcal{C}_1$ can be rewritten as:

$$(\mathbf{x} - \bar{\mathbf{x}}_2)^T \mathbf{\Sigma}_2^{-1} (\mathbf{x} - \bar{\mathbf{x}}_2) - (\mathbf{x} - \bar{\mathbf{x}}_1)^T \mathbf{\Sigma}_1^{-1} (\mathbf{x} - \bar{\mathbf{x}}_1) \geq \ln \frac{\tau^2 \det \mathbf{\Sigma}_1}{\det \mathbf{\Sigma}_2} \tag{6}$$

In general, this result is a quadratic discriminant. However, a linear classifier is often desired for the following reasons:

1. A linear classifier is robust against noise since it tends not to overfit (Mika, Ratsch, Weston, Scholkopf, & Mullers, 1999).
2. A linear classifier has relatively shorter training and testing times (Yuan, Ho, & Lin, 2012).
3. Many linear classifiers allow for a transformation of the original feature space into a higher dimensional feature space using the kernel trick for better classification in the case of a non-linear decision boundary (Bishop, 2006, Chapter 6).

By calling on the assumption of homoscedasticity, i.e. $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \mathbf{\Sigma}_x$, the original quadratic discriminant given by (6) for classifying a given vector $\mathbf{x}$ decomposes into the following linear decision rule:

$$\mathbf{x}^T \mathbf{\Sigma}_x^{-1} (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2) \underset{\mathcal{C}_2}{\overset{\mathcal{C}_1}{\gtrless}} \ln \tau + \frac{1}{2}(\bar{\mathbf{x}}_1^T \mathbf{\Sigma}_x^{-1} \bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2^T \mathbf{\Sigma}_x^{-1} \bar{\mathbf{x}}_2) \tag{7}$$

Here, $\mathbf{\Sigma}_x^{-1}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)$ is a vector of weights denoted by $\mathbf{w}$ and $\ln \tau + \frac{1}{2}(\bar{\mathbf{x}}_1^T \mathbf{\Sigma}_x^{-1} \bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2^T \mathbf{\Sigma}_x^{-1} \bar{\mathbf{x}}_2)$ is a threshold denoted by w_0. This linear classifier is also known as Fishers Linear Discriminant. If only the weight vector $\mathbf{w}$ is required for dimensionality reduction, $\mathbf{w}$ may be obtained by maximising Fishers criterion (Fisher, 1936), given by:

$$S = \frac{\mathbf{w}^T (\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)^T \mathbf{w}}{\mathbf{w}^T \mathbf{\Sigma}_x \mathbf{w}} \tag{8}$$

where $\mathbf{\Sigma}_x = n_1 \mathbf{\Sigma}_1 + n_2 \mathbf{\Sigma}_2$ and n_1, n_2 are the cardinalities of $\mathcal{D}_1$ and $\mathcal{D}_2$ respectively.

LDA is the optimal Bayes' classifier for binary classification if the normality and homoscedasticity assumptions hold (Hamsici & Martinez, 2008) (Izenman, 2009, Chapter 8). It demands only the computation of the dot product between $\mathbf{w}$ and $\mathbf{x}$, which is a relatively computationally inexpensive operation.

As a supervised learning algorithm, LDA is performed either for dimensionality reduction (usually followed by classification) (Barber, 2012, Chapter 16; Buturovic, 1994; Duin & Loog, 2004; Sengur, 2008), or directly for the purpose of statistical classification (Fukunaga, 2013, Chapter 4; Izenman, 2009; Mika et al., 1999). LDA has been applied to several problems such as medical diagnosis e.g. Coomans, Jonckheer, Massart, Broeckaert, and Blockx (1978); Polat, Güneş, and Arslan (2008); Sengur (2008); Sharma and Paliwal (2008), face and object recognition e.g. Chen, Liao, Ko, Lin, and Yu (2000); Liu, Chen, Tan, and Zhang (2007); Song, Zhang, Wang, Liu, and Tao (2007); Yu and Yang (2001) and credit card fraud prediction e.g. Mahmoudi and Duman (2015). The widespread use of LDA in these areas is not because the datasets necessarily satisfy the normality and homoscedasticity assumptions, but mainly due to the robustness of LDA against noise, being a linear model (Mika et al., 1999). Since the linear Support Vector Machine (SVM) can be quite expensive to train, especially for large values of K or n ($n = n_1 + n_2$), LDA is often relied upon (Hariharan, Malik, & Ramanan, 2012).

Yet, practical implementation of LDA is not without problems. Of note is the small sample size (SSS) problem that LDA faces with high-dimensional data and much smaller training data (Lu, Plataniotis, & Venetsanopoulos, 2003; Sharma & Paliwal, 2015). When $d \gg n$, the scatter matrix Σ_x is not invertible, as it is not full-rank. Since the decision rule as given by (7) requires the computation of the inverse of Σ_x, the singularity of Σ_x makes the solution infeasible. In works by, for example, Liu et al. (2007); Paliwal and Sharma (2012), this problem is overcome by taking the Moore–Penrose pseudo-inverse of the scatter matrix, rather than the ordinary matrix inverse. Sharma and Paliwal (2008) use a gradient descent approach where one starts from an initial solution of $\mathbf{w}$ and moves in the negative direction of the gradient of Fisher's criterion (8). This method avoids the computation of an inverse altogether. Another approach to solving the SSS problem involves adding a scalar multiple of the identity matrix to the scatter matrix to make the resulting matrix non-singular, a method known as regularised discriminant analysis (Friedman, 1989; Lu et al., 2003).

However, for a given dataset that does not satisfy the homoscedasticity or normality assumption, one would expect that modifications to the original LDA procedure accounting for these violations would yield an improved performance. One such modification, in the case of a non-normal distribution, is the mixture discriminant analysis (Hastie & Tibshirani, 1996; Ju, Kolaczyk, & Gopal, 2003; McLachlan, 2004) in which a non-normal distribution is modelled as a mixture of Gaussians. However, the parameters of the mixture components or even the number of mixture components, are usually not known a priori. Other non-parametric approaches to LDA that remove the normality assumption involve using local neighbourhood structures (Cai, He, Zhou, Han, & Bao, 2007; Fukunaga & Mantock, 1983; Li, Lin, & Tang, 2009) to construct a similarity matrix instead of the scatter matrix Σ_x used in LDA. However, these approaches aim at linear dimensionality reduction, rather than linear classification. Another modification, in the case of a non-linear decision boundary between $\mathcal{D}_1$ and $\mathcal{D}_2$, is the Kernel Fisher Discriminant (KFD) (Mika et al., 1999; Polat et al., 2008; Zhao, Sun, Yu, Liu, & Ye, 2009). KFD maps the original feature space $\mathcal{X}$ into some other space $\mathcal{Y}$ (usually higher dimensional) via the kernel trick (Mika et al., 1999). While the main utility of the kernel is to guarantee linear separability in the transformed space, the kernel may also be employed to transform non-normal data into one that is near-normal.

Our proposed method differs from the above approaches in that we primarily consider violation of the homoscedasticity assumption, and do not address the SSS problem. We seek to provide a linear approximation to the quadratic boundary given by (6) under heteroscedasticity without any kernel transformation; we note