



City Research Online

City St George's, University of London

Citation: Zhu, R. & Xue, J-H. (2017). On the orthogonal distance to class subspaces for high-dimensional data classification. *Information Sciences*, 417, pp. 262-273. doi: 10.1016/j.ins.2017.07.019

This is the accepted version of the paper.

This version of the publication may differ from the final published version. To cite this item please consult the publisher's version.

Permanent repository link: <https://openaccess.city.ac.uk/id/eprint/20734/>

Link to published version: <https://doi.org/10.1016/j.ins.2017.07.019>

Copyright and Reuse: Copyright and Moral Rights remain with the author(s) and/or copyright holders. Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge, unless otherwise indicated, provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way. For full details of reuse please refer to [City Research Online policy](#).

On the orthogonal distance to class subspaces for high-dimensional data classification

Rui Zhu^a, Jing-Hao Xue^{a,*}

^a*Department of Statistical Science, University College London, London WC1E 6BT, UK*

Abstract

The orthogonal distance from an instance to the subspace of a class is a key metric for pattern classification by the class subspace-based methods. There is a close relationship between the orthogonal distance and the residual standard deviation of a test instance from the class subspace. In this paper, we shall show that an established and widely-used relationship, between the residual standard deviation and the sum of squares of the residual PC scores, is not precise, and thus can lead to incorrect results, for the inference of high-dimensional data which nowadays are common in practice.

Keywords: Classification, high-dimensional data, orthogonal distance, principal component analysis (PCA), soft independent modelling of class analogy (SIMCA).

1. Introduction

2 In class subspace-based classification methods, a subspace is first learned
3 in the training phase for each class separately from its training data. Then in

*Corresponding author. Tel.: +44-20-7679-1863; Fax: +44-20-3108-3105

Email addresses: r.zhu.12@ucl.ac.uk (Rui Zhu), jinghao.xue@ucl.ac.uk (Jing-Hao Xue)

4 the test phase, these learned class subspaces are utilised to predict the label
5 of a new test instance, by comparing the distances from the test instance to
6 the class subspaces, in terms of certain distance metrics. For example, in a
7 widely-used classifier for spectral data called soft independent modelling of
8 class analogy (SIMCA) [28], principal component (PC) subspaces are learned
9 for individual classes. Similar to SIMCA, another popular PCA-based clas-
10 sification approach has been extensively adopted in process control in engi-
11 neering, such as fault detection and diagnosis [20, 16, 15, 25]. Besides classi-
12 fication methods, some clustering methods also aim to seek low-dimensional
13 subspaces for better clustering results [13, 23, 22].

14 In the above two classification approaches, associated with the PC sub-
15 spaces, two distance metrics (or statistics) are often adopted to achieve pat-
16 tern classification [3, 17, 18, 20, 16, 15, 25, 29]: 1) the orthogonal distance
17 (OD), also known as the Q-statistic or the squared prediction error, i.e. the
18 squared orthogonal Euclidean distance from a test instance to a PC subspace;
19 and 2) the score distance (SD), also known as the Hotelling’s T^2 statistic,
20 i.e. the squared Mahalanobis distance from the projection of a test instance
21 to the centre of a PC subspace [17]. The distributions of OD and SD have
22 also been studied extensively, in order to find a proper acceptance area for
23 classification; recent work includes [17], [18], [19], [30] and [21]. Also in
24 recent years, a linear combination of these two distances is often used to
25 classify a test instance: the test instance is assigned to the class with the
26 minimum value of the linear combination [3].

27 There is a close relationship between the OD (from a test instance to a
28 class subspace) and the residual standard deviation of the test instance to

29 the class subspace. Moreover, Maesschalck et al. [9] show that the residual
 30 standard deviation based on the residual matrix can be equivalently calcu-
 31 lated from using the residual PC scores based on the PC score matrix. This
 32 work has been cited over a hundred times, including methodological develop-
 33 ments [4, 10, 8], reviews [24, 14] and applications [5, 2, 6, 27, 7]. The recent
 34 work studying the distributions of OD and SD [17, 18, 19] also adopted the
 35 formulae in [9] following [10].

36 However in this paper, we shall point out that the relationship presented
 37 in [9], between the residual standard deviation and the sum of squares of the
 38 residual PC scores, is *not* precise for the inference of high-dimensional data.

39 To distinguish the training and test scenarios, we shall establish the no-
 40 tation of two ODs, respectively, as follows.

- 41 1. The OD $v^{k,l}$ from the *training* instance l to the subspace of class k
 42 that was learned from all training instances. It is closely related to
 43 the residual standard deviation $s^{k,0}$ of class k , which will be defined in
 44 Section 2.1.
- 45 2. The OD $v^{k,new}$ from the new *test* instance to the subspace of class k .
 46 It is closely related to the residual standard deviation $s^{k,new}$ of the new
 47 test instance to class k , which will be defined in Section 2.2.

48 In short, the difference between $v^{k,l}$ and $v^{k,new}$ is that $v^{k,l}$ is the OD for the
 49 training instance while $v^{k,new}$ is the OD for the test instance.

50 The contributions of this paper are as follows. First, although Maess-
 51 chalck et al. [9] establish formulae for $s^{k,0}$ and $s^{k,new}$ using the residual PC
 52 scores, we shall show that their formula for $s^{k,new}$ is only precise when the
 53 training data of class k have more instances than predictor features, i.e. when

54 the number of instances (denoted by n_k) is larger than the number of features
 55 (denoted by p). In other words, we shall show that, when the training data
 56 of class k are high-dimensional (i.e. $n_k \leq p$, also called “large p , small n ” in
 57 the statistical literature), the calculation of $s^{k,new}$ in [9] is not precise.

58 Second, because of the above results, we shall point out that, for high-
 59 dimensional data, although the OD $v^{k,l}$ can be accurately calculated by fol-
 60 lowing the (precise) formula of the residual standard deviation $s^{k,0}$ in [9],
 61 the OD $v^{k,new}$ cannot be accurately calculated by following the (imprecise)
 62 formulae of the residual standard deviation $s^{k,new}$ in [9]. Consequently, in-
 63 ference results of the studies that calculated the ODs for high-dimensional
 64 data using the formulae in [9] can be imprecise.

65 Because nowadays high-dimensional data are commonly present in pattern-
 66 recognition tasks, it is of great interest to practitioners to point out the im-
 67 precise calculation of the ODs for high-dimensional data if we follow the
 68 formulae in [9], as well as to suggest that the formulae in [28] should be
 69 adopted in this “large p , small n ” paradigm.

70 **2. The calculations of OD in [9]**

71 The following calculations are all for class k . The subscripts p , q and r
 72 denote the number of columns in matrices \mathbf{U} , \mathbf{D} , \mathbf{V} and \mathbf{T} ; for example, \mathbf{V}_p
 73 indicates that there are p columns in matrix \mathbf{V}_p of class k .

74 *2.1. The training phase of class k*

75 Suppose $\mathbf{X} \in \mathbb{R}^{n_k \times p}$ is the training set of class k , in which there are n_k
 76 training instances (or say training samples) and each instance is represented
 77 by a p -dimensional data vector. To build the PC subspace of class k , we

78 apply the reduced singular value decomposition (SVD) to the column-centred
 79 training set $\mathbf{X}_{(c)}$:

$$\mathbf{X}_{(c)} = \mathbf{U}_q \mathbf{D}_q (\mathbf{V}_q)^T, \quad (1)$$

80 where $\mathbf{U}_q \in \mathbb{R}^{n_k \times q}$ and $\mathbf{V}_q \in \mathbb{R}^{p \times q}$ are the two matrices containing left and
 81 right singular vectors as columns, respectively, and $\mathbf{D}_q \in \mathbb{R}^{q \times q}$ is a diagonal
 82 matrix with singular values $\{\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_q \geq 0\}$. The parameter
 83 $q \leq \min(p, n_k - 1)$ is the rank of $\mathbf{X}_{(c)}$.

84 In PCA, the rows of $\mathbf{T}_q = \mathbf{U}_q \mathbf{D}_q \in \mathbb{R}^{n_k \times q}$ are known as PC scores and
 85 the columns of \mathbf{V}_q are known as PCs. Suppose the first r ($r \leq q$) PCs are
 86 selected to build the PC subspace for class k , then

$$\mathbf{X}_{(c)} = \mathbf{T}_r (\mathbf{V}_r)^T + \mathbf{E}, \quad (2)$$

87 where $\mathbf{T}_r \in \mathbb{R}^{n_k \times r}$; $\mathbf{V}_r \in \mathbb{R}^{p \times r}$; and $\mathbf{E} \in \mathbb{R}^{n_k \times p}$ is the training residual matrix
 88 of class k .

89 In [9], the residual standard deviation of class k is expressed in two forms:

$$s^{k,0} = \sqrt{\frac{1}{\text{DoF}^{k,0}} \sum_{l=1}^{n_k} \sum_{j=1}^p (e_{lj})^2} = \sqrt{\frac{1}{\text{DoF}^{k,0}} \sum_{l=1}^{n_k} \sum_{i=r+1}^q (t_{li})^2} \quad (3)$$

90 where $\text{DoF}^{k,0} = (q - r)(n_k - r - 1)$, e_{lj} is the (l, j) -entry of residual matrix
 91 \mathbf{E} representing the residual of the l th instance for the j th variable, and t_{li} is
 92 the (l, i) -entry of score matrix \mathbf{T}_q representing the score of the l th instance
 93 for the i th PC.

94 The OD from the l th training instance to the subspace of class k , $v^{k,l}$, is

95 originally defined as $\sum_{j=1}^p (e_{lj})^2$. Thus $\sum_{l=1}^{n_k} v^{k,l}$ is proportional to $(s^{k,0})^2$,

$$\sum_{l=1}^{n_k} v^{k,l} = (s^{k,0})^2 (q - r)(n_k - r - 1). \quad (4)$$

96 In [9], it follows from (3) that $\sum_{l=1}^{n_k} v^{k,l}$ can be calculated as

$$\sum_{l=1}^{n_k} v^{k,l} = \sum_{l=1}^{n_k} \sum_{i=r+1}^q (t_i)^2. \quad (5)$$

97 2.2. The test phase for class k

98 In the test (prediction) phase, to decide whether a new instance \mathbf{x}^{new}
 99 belongs to class k or not, \mathbf{x}^{new} is first centred by using the means of the
 100 variables of the training data \mathbf{X} of class k , and the result is denoted by $\mathbf{x}_{(c)}^{k,new}$.
 101 Then projecting $\mathbf{x}_{(c)}^{k,new}$ to the PC subspace of class k with the selected r PCs,
 102 we can obtain

$$\mathbf{x}_{(c)}^{k,new} = \mathbf{t}_r^{k,new} (\mathbf{V}_r)^T + \mathbf{e}^{k,new}, \quad (6)$$

103 where $\mathbf{t}_r^{k,new} \in \mathbb{R}^{1 \times r}$ and $\mathbf{e}^{k,new} \in \mathbb{R}^{1 \times p}$ are two vectors of the PC score and
 104 the residual, respectively, of the new instance when it is fitted to the subspace
 105 of class k .

106 In [9], the residual standard deviation of the new instance is also expressed
 107 in two forms:

$$s^{k,new} = \sqrt{\frac{1}{\text{DoF}^{k,new}} \sum_{j=1}^p (e_j^{k,new})^2} = \sqrt{\frac{1}{\text{DoF}^{k,new}} \sum_{i=r+1}^q (t_i^{k,new})^2}, \quad (7)$$

108 where $\text{DoF}^{k,new} = (q - r)$, $e_j^{k,new}$ and $t_i^{k,new}$ denote the j th element of the

109 residual vector $\mathbf{e}^{k,new}$ and the i th element of the PC score vector $\mathbf{t}_r^{k,new}$,
 110 respectively.

111 The OD from the new instance to the subspace of class k , $v^{k,new}$, is
 112 originally defined as $\sum_{j=1}^p (e_j^{k,new})^2$. Thus $v^{k,new}$ is proportional to $(s^{k,new})^2$,

$$v^{k,new} = (s^{k,new})^2(q - r). \quad (8)$$

113 In [9], it follows from (7) that $v^{k,new}$ can be written as

$$v^{k,new} = \sum_{i=r+1}^q (t_i^{k,new})^2. \quad (9)$$

114 To determine the class of \mathbf{x}^{new} , the residual standard deviation $s^{k,new}$
 115 of \mathbf{x}^{new} is compared to the residual standard deviation $s^{k,0}$ of the training
 116 instances of class k [9]. The F -test statistic used in [9] to determine whether
 117 the two residual variances are significantly different is expressed as

$$F^{k,new} = \frac{(s^{k,new})^2}{(s^{k,0})^2} = \frac{\sum_{i=r+1}^q (t_i^{k,new})^2 (n_k - r - 1)}{\sum_{l=1}^{n_k} \sum_{i=r+1}^q (t_{li})^2}. \quad (10)$$

118 3. Discussion of $v^{k,l}$ and $v^{k,new}$

119 The calculations for $v^{k,0}$ and $v^{k,new}$ in [9] use formulae (5) and (9), respec-
 120 tively. We shall show that, while formula (5) is correct for both the cases of
 121 $n_k > p$ and $n_k \leq p$, formula (9) is only valid when $n_k > p$.

122 3.1. $v^{k,l}$

123 The OD $v^{k,l}$ is originally defined on the basis of the residual matrix \mathbf{E} .
 124 The calculation of $v^{k,l}$ in (5), which was defined in [9], is on the basis of the

125 PC score matrix \mathbf{T}_r . This is due to the relationship that

$$\sum_{l=1}^{n_k} \sum_{j=1}^p (e_{lj})^2 = \sum_{l=1}^{n_k} \sum_{i=r+1}^q (t_{li})^2 . \quad (11)$$

126 This relationship is true for both the cases of $n_k > p$ and $n_k \leq p$, as we shall
 127 show in the following two subsections, respectively.

128 *3.1.1. $n_k > p$*

129 When $n_k > p$, we have $q = p$ (assume that no feature is a linear com-
 130 bination of others), and thus $\mathbf{V}_q \in \mathbb{R}^{p \times p}$ is a square matrix. It follows that
 131 $\mathbf{V}_q(\mathbf{V}_q)^T = (\mathbf{V}_q)^T\mathbf{V}_q = \mathbf{I}_p$.

Let $\mathbf{x}_{(c)}^l \in \mathbb{R}^{1 \times p}$ denote the l -th training instance in class k , i.e. the l -th
 row of $\mathbf{X}_{(c)}$. For every $\mathbf{x}_{(c)}^l$ ($l = 1, \dots, n_k$), we have $\mathbf{x}_{(c)}^l = \mathbf{x}_{(c)}^l \mathbf{V}_q(\mathbf{V}_q)^T$ and

$$\begin{aligned} \sum_{j=1}^p (e_{lj})^2 &= \|\mathbf{x}_{(c)}^l - \mathbf{x}_{(c)}^l \mathbf{V}_r(\mathbf{V}_r)^T\|_2^2 \\ &= \|\mathbf{x}_{(c)}^l \mathbf{V}_q(\mathbf{V}_q)^T - \mathbf{x}_{(c)}^l \mathbf{V}_r(\mathbf{V}_r)^T\|_2^2 \\ &= \|\mathbf{t}_q^l(\mathbf{V}_q)^T - \mathbf{t}_r^l(\mathbf{V}_r)^T\|_2^2 \\ &= \sum_{i=r+1}^q (t_{li})^2 , \end{aligned} \quad (12)$$

132 where $\|\cdot\|_2$ denotes the Euclidean norm of a vector, and \mathbf{t}_q^l and \mathbf{t}_r^l are the
 133 l th row of \mathbf{T}_q and \mathbf{T}_r , respectively. Therefore (11) and thus (5) are correct
 134 when $n_k > p$.

135 3.1.2. $n_k \leq p$

136 When $n_k \leq p$, we have $q = \text{rank}(\mathbf{X}_{(c)}) \leq n_k - 1 < p$, and thus $\mathbf{V}_q \in \mathbb{R}^{p \times q}$
 137 is not square. It follows that $(\mathbf{V}_q)^T \mathbf{V}_q = \mathbf{I}_q$ but $\mathbf{V}_q(\mathbf{V}_q)^T \neq \mathbf{I}_p$.

138 Suppose we apply the full SVD to $\mathbf{X}_{(c)}$:

$$\mathbf{X}_{(c)} = \mathbf{U}_{n_k} \hat{\mathbf{D}}_p (\mathbf{V}_p)^T, \quad (13)$$

139 where $\mathbf{U}_{n_k} \in \mathbb{R}^{n_k \times n_k}$ and $\mathbf{V}_p \in \mathbb{R}^{p \times p}$ denote the two matrices containing n_k
 140 left and p right singular vectors as columns, respectively, and $\hat{\mathbf{D}}_p \in \mathbb{R}^{n_k \times p}$
 141 is a matrix with singular values $\{\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{n_k-1} \geq \lambda_{n_k} = 0\}$ on the
 142 main diagonal.

143 To make the explanation more clear, we expand $\hat{\mathbf{D}}_p \in \mathbb{R}^{n_k \times p}$ to a square
 144 matrix $\mathbf{D}_p \in \mathbb{R}^{p \times p}$ by adding zeros because the singular values associated
 145 with the last $(p - q)$ PCs are zeros when $n_k \leq p$. Matrix $\mathbf{U}_{n_k} \in \mathbb{R}^{n_k \times n_k}$ is
 146 also expanded to $\mathbf{U}_p \in \mathbb{R}^{n_k \times p}$ using $(p - n_k)$ unit-length column vectors that
 147 are randomly calculated to be orthogonal to the previous column vectors.
 148 Thus we have

$$\mathbf{X}_{(c)} = \mathbf{U}_{n_k} \hat{\mathbf{D}}_p (\mathbf{V}_p)^T = \mathbf{U}_p \mathbf{D}_p (\mathbf{V}_p)^T, \quad (14)$$

149 where $\mathbf{U}_p \in \mathbb{R}^{n_k \times p}$ and $\mathbf{V}_p \in \mathbb{R}^{p \times p}$ denote the matrices containing p left and
 150 p right singular vectors, respectively, and $\mathbf{D}_p \in \mathbb{R}^{p \times p}$ is a diagonal matrix
 151 with singular values $\{\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_q \geq \lambda_{q+1} = \dots = \lambda_p = 0\}$. Since
 152 $\mathbf{V}_p \in \mathbb{R}^{p \times p}$ is square, we have $\mathbf{V}_p (\mathbf{V}_p)^T = (\mathbf{V}_p)^T \mathbf{V}_p = \mathbf{I}_p$.

153 Let $\mathbf{T}_p = \mathbf{U}_p \mathbf{D}_p \in \mathbb{R}^{n_k \times p}$ denote the PC scores. Let t_{li} denote the (l, i) -
 154 entry of score matrix \mathbf{T}_p representing the score of the l th instance for the i th
 155 PC.

Let \mathbf{m}^l denote the residual from using the first q PCs to reconstruct $\mathbf{x}_{(c)}^l$: $\mathbf{m}^l = \mathbf{x}_{(c)}^l - \mathbf{x}_{(c)}^l \mathbf{V}_q (\mathbf{V}_q)^T$. We calculate the sum of squares of the residuals in \mathbf{m}^l for the l -th instance:

$$\begin{aligned} \|\mathbf{m}^l\|_2^2 &= \|\mathbf{x}_{(c)}^l - \mathbf{x}_{(c)}^l \mathbf{V}_q (\mathbf{V}_q)^T\|_2^2 \\ &= \|\mathbf{x}_{(c)}^l \mathbf{V}_p (\mathbf{V}_p)^T - \mathbf{x}_{(c)}^l \mathbf{V}_q (\mathbf{V}_q)^T\|_2^2 \\ &= \|\mathbf{t}_p^l (\mathbf{V}_p)^T - \mathbf{t}_q^l (\mathbf{V}_q)^T\|_2^2. \end{aligned} \quad (15)$$

156 The sum of $\|\mathbf{m}^l\|_2^2$ for all n_k training instances is

$$\sum_{l=1}^{n_k} \|\mathbf{m}^l\|_2^2 = \sum_{l=1}^{n_k} \sum_{i=q+1}^p (t_{li})^2 = \sum_{i=q+1}^p (\lambda_i)^2. \quad (16)$$

157 The second equation in (16) can be shown as follows. $\mathbf{X}_{(c)} = \mathbf{U}_p \mathbf{D}_p (\mathbf{V}_p)^T \Rightarrow$
 158 $(\mathbf{U}_p)^T \mathbf{X}_{(c)} \mathbf{V}_p = \mathbf{D}_p \Rightarrow (\mathbf{U}_p)^T \mathbf{T}_p = \mathbf{D}_p$. For the i th singular value λ_i in \mathbf{D}_p ,
 159 we have $(\lambda_i)^2 = (\mathbf{u}_i^T \mathbf{t}_i)^2 = \mathbf{t}_i^T \mathbf{u}_i \mathbf{u}_i^T \mathbf{t}_i = \mathbf{t}_i^T \mathbf{t}_i = \sum_{l=1}^{n_k} (t_{li})^2$, where \mathbf{u}_i and \mathbf{t}_i are
 160 the i th columns of \mathbf{U}_p and \mathbf{T}_p , respectively.

161 Since the last $(p - q)$ singular values are zeros, $\sum_{l=1}^{n_k} \|\mathbf{m}^l\|_2^2 = 0$. Because
 162 each term in the sum $\sum_{l=1}^{n_k} \|\mathbf{m}^l\|_2^2$ is nonnegative, $\|\mathbf{m}^l\|_2^2 = 0$ for all l ($l =$
 163 $1, \dots, n_k$). Thus we have $\mathbf{x}_{(c)}^l = \mathbf{x}_{(c)}^l \mathbf{V}_q (\mathbf{V}_q)^T$, which means that the first q
 164 PCs can perfectly reconstruct the training instances in class k . Using the
 165 same proof as in (12), we can show that (11) and thus (5) are also true for
 166 $n_k \leq p$.

167 Therefore, $v^{k,l}$ can be correctly calculated by using (5) for both the cases
 168 of $n_k > p$ and $n_k \leq p$.

169 3.2. $v^{k,new}$

170 The OD $v^{k,new}$ is originally defined in terms of the residual vector $\mathbf{e}^{k,new}$ [28],
 171 while following [9] $v^{k,new}$ is formulated in (9) by using the PC score $t_r^{k,new}$ of
 172 the new sample. We shall show that the formula (9) is valid when $n_k > p$
 173 but not valid when $n_k \leq p$, in the following two subsections, respectively.

174 3.2.1. $n_k > p$

175 When $n_k > p$, we have $q = p$, and thus $\mathbf{V}_q \in \mathbb{R}^{p \times p}$ is a square matrix. As
 176 before, $\mathbf{V}_q(\mathbf{V}_q)^T = (\mathbf{V}_q)^T \mathbf{V}_q = \mathbf{I}_p$. Since $\mathbf{x}_{(c)}^{k,new} = \mathbf{x}_{(c)}^{k,new} \mathbf{V}_q(\mathbf{V}_q)^T$, we have

$$\sum_{j=1}^p (e_j^{k,new})^2 = \sum_{i=r+1}^q (t_i^{k,new})^2. \quad (17)$$

177 Using a proof similar to (12) by replacing $\mathbf{x}_{(c)}^l$ with $\mathbf{x}_{(c)}^{k,new}$, we can readily
 178 show that (17) and thus (9) are correct for $n_k > p$.

179 3.2.2. $n_k \leq p$

180 When $n_k \leq p$, we have $q = \text{rank}(\mathbf{X}_{(c)}) < p$, and thus $\mathbf{V}_q \in \mathbb{R}^{p \times q}$ is not
 181 square. Again, it follows that $(\mathbf{V}_q)^T \mathbf{V}_q = \mathbf{I}_q$ but $\mathbf{V}_q(\mathbf{V}_q)^T \neq \mathbf{I}_p$.

Let $\mathbf{m}^{k,new}$ denote the residual from using the q PC vectors to reconstruct
 $\mathbf{x}_{(c)}^{k,new}$: $\mathbf{m}^{k,new} = \mathbf{x}_{(c)}^{k,new} - \mathbf{x}_{(c)}^{k,new} \mathbf{V}_q(\mathbf{V}_q)^T$. We calculate the sum of squares

of the residuals in $\mathbf{m}^{k,new}$:

$$\begin{aligned}
\|\mathbf{m}^{k,new}\|_2^2 &= \|\mathbf{x}_{(c)}^{k,new} - \mathbf{x}_{(c)}^{k,new} \mathbf{V}_q (\mathbf{V}_q)^T\|_2^2 \\
&= \|\mathbf{x}_{(c)}^{k,new} \mathbf{V}_p (\mathbf{V}_p)^T - \mathbf{x}_{(c)}^{k,new} \mathbf{V}_q (\mathbf{V}_q)^T\|_2^2 \\
&= \|\mathbf{t}_p^{k,new} (\mathbf{V}_p)^T - \mathbf{t}_q^{k,new} (\mathbf{V}_q)^T\|_2^2 \\
&= \sum_{i=q+1}^p (t_i^{k,new})^2, \tag{18}
\end{aligned}$$

182 where $\|\cdot\|_2$ denotes the Euclidean norm of a vector.

183 However, unlike the case for the training data, $\sum_{i=q+1}^p (t_i^{k,new})^2$ is not
184 necessarily equal to zero for a p -dimensional test instance. Thus $\mathbf{x}_{(c)}^{k,new} \neq$
185 $\mathbf{x}_{(c)}^{k,new} \mathbf{V}_q (\mathbf{V}_q)^T$, which means that the new test instance cannot be perfectly
186 reconstructed by the first q PC vectors.

Hence, if we rewrite

$$\begin{aligned}
\mathbf{x}_{(c)}^{k,new} &= \mathbf{x}_{(c)}^{k,new} \mathbf{V}_q (\mathbf{V}_q)^T + \mathbf{m}^{k,new} \\
&= \mathbf{x}_{(c)}^{k,new} \mathbf{V}_r (\mathbf{V}_r)^T + (\mathbf{x}_{(c)}^{k,new} \mathbf{V}_q (\mathbf{V}_q)^T - \mathbf{x}_{(c)}^{k,new} \mathbf{V}_r (\mathbf{V}_r)^T) + \mathbf{m}^{k,new}, \tag{19}
\end{aligned}$$

we have

$$\begin{aligned}
\mathbf{e}^{k,new} &= (\mathbf{x}_{(c)}^{k,new} \mathbf{V}_q (\mathbf{V}_q)^T - \mathbf{x}_{(c)}^{k,new} \mathbf{V}_r (\mathbf{V}_r)^T) + \mathbf{m}^{k,new} \\
&= (\mathbf{t}_q^{k,new} (\mathbf{V}_q)^T - \mathbf{t}_r^{k,new} (\mathbf{V}_r)^T) + (\mathbf{t}_p^{k,new} (\mathbf{V}_p)^T - \mathbf{t}_q^{k,new} (\mathbf{V}_q)^T) \\
&= \mathbf{t}_p^{k,new} (\mathbf{V}_p)^T - \mathbf{t}_r^{k,new} (\mathbf{V}_r)^T \tag{20}
\end{aligned}$$

and

$$\begin{aligned}
\sum_{j=1}^p (e_j^{k,new})^2 &= \|\mathbf{e}^{k,new}\|_2^2 \\
&= \|\mathbf{t}_p^{k,new}(\mathbf{V}_p)^T - \mathbf{t}_r^{k,new}(\mathbf{V}_r)^T\|_2^2 \\
&= \sum_{i=r+1}^p (t_i^{k,new})^2 \\
&= \sum_{i=r+1}^q (t_i^{k,new})^2 + \sum_{i=q+1}^p (t_i^{k,new})^2. \tag{21}
\end{aligned}$$

187 Comparing (21) with (17), we can find an additional term $\sum_{i=q+1}^p (t_i^{k,new})^2$ in
188 (21), and this term may not be zero. It follows that (17) and thus (9) are
189 not valid when $n_k \leq p$.

190 When $n_k \leq p$, $\sum_{i=q+1}^p (t_i^{k,new})^2$ is hard to estimate because the last $(p-q)$
191 PCs are randomly calculated by satisfying the orthogonal condition. Never-
192 theless, it can be harmful to the classification of the new instance of high-
193 dimensional “large p , small n ” data, if we use (9) to calculate $v^{k,new}$ which
194 omits $\sum_{i=q+1}^p (t_i^{k,new})^2$, because the decision making for classification is based
195 on $v^{k,new}$.

196 4. Experiments

197 In the following experiments, take SIMCA as an example: we compare
198 the SIMCA with the OD defined originally in [28] (denoted by SIMCA) and
199 the SIMCA with the OD calculated by following [9] (denoted by SIMCA-D),
200 evaluating them on both simulated and real datasets. We aim to show that
201 the additional term $\sum_{i=q+1}^p (t_i^{k,new})^2$ can be important for classifying high-

202 dimensional data. To simplify the experiment settings, we discuss the effect
 203 of $\sum_{i=q+1}^p (t_i^{k,new})^2$ on two-class classification in the experiments. The effect
 204 of $\sum_{i=q+1}^p (t_i^{k,new})^2$ on multi-class classification can be readily extended.

205 4.1. Classification rule

206 New test instances can be classified by following the classification rule of
 207 the robust SIMCA (RSIMCA) [3], which is a linear combination of the OD
 208 and the SD of a new test instance (Here our notations of OD and SD are
 209 both for *squared* distances). That is, a new test instance is classified to the
 210 class with the minimum value of

$$\gamma \frac{\text{OD}^k}{c_{\text{OD}}^k} + (1 - \gamma) \frac{\text{SD}^k}{c_{\text{SD}}^k}, \quad (22)$$

211 where $\text{OD}^k = v^{k,new}$; $\text{SD}^k = (\mathbf{t}_r^{k,new})^T \mathbf{\Lambda}_r^{-1} \mathbf{t}_r^{k,new}$, in which $\mathbf{\Lambda}_r$ is the diagonal
 212 matrix of the r largest eigenvalues for the PC subspace; $c_{\text{SD}}^k = \chi_{r;0.975}^2$; and
 213 $c_{\text{OD}}^k = (\hat{\mu} + \hat{\sigma} z_{0.975})^3$, in which $\hat{\mu}$ and $\hat{\sigma}$ are the mean and the standard
 214 deviation of the square roots of $v^{k,l}$.

215 Since OD^k is the only term that is different between SIMCA and SIMCA-
 216 D, the value of the second term in (22) does not affect the difference between
 217 SIMCA and SIMCA-D. We force the value of the second term in (22) to zero
 218 by setting $\gamma = 1$, to simplify the experiments.

219 4.2. Validation criterion

220 We use the overall misclassification percentage (MP) as the validation
 221 criterion following the experiments in [3]. We use the one-assignment-rule
 222 suggested in [3], i.e. a test sample is assigned to one of the known classes

223 with the smallest F -value, to simplify the calculation of the MP and obtain
 224 unambiguous final results. The MP is defined as

$$\text{MP} = \sum_{k=1}^K n_k^t / N^t, \quad (23)$$

225 where n_k^t denotes the the number of wrongly assigned test samples in class
 226 k and N^t denotes the total number of test samples.

227 4.3. Datasets

228 4.3.1. Simulated datasets

229 Simulated datasets are generated by following the experiments in [18].
 230 Assume that a sample vector \mathbf{x} is the sum of two independent normal random
 231 components:

$$\mathbf{x} = \boldsymbol{\delta} + \boldsymbol{\epsilon}, \quad (24)$$

232 where

$$\boldsymbol{\delta} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \text{ and } \boldsymbol{\epsilon} \sim N(\mathbf{0}, \sigma^2 \mathbf{I}). \quad (25)$$

233 Based on the above assumption, the samples of the two classes are drawn
 234 from $N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1 + \sigma_1^2 \mathbf{I})$ and $N(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2 + \sigma_2^2 \mathbf{I})$, respectively.

235 Two sets of parameters, simulation A and simulation B, are devised to
 236 show the following two situations, respectively: 1) $\sum_{i=q+1}^p (t_i^{k,new})^2$ is not
 237 important for classification; and 2) $\sum_{i=q+1}^p (t_i^{k,new})^2$ may be important for
 238 classification. The details of the two simulation settings are summarised in
 239 Table 1.

240 For each simulation setting, we generate 20 datasets with different n_k/p
 241 ratios to explore the difference between SIMCA and SIMCA-D with respect

Table 1: Simulation settings. Notation: K , number of classes; D , number of datasets; n_k , number of samples in each class

	Simulation A	Simulation B
$\boldsymbol{\mu}_1$	$\mathbf{0}_p$	$\mathbf{0}_p$
$\boldsymbol{\mu}_2$	$(10, \mathbf{0}_{p-1}^T)^T$	$(10, \mathbf{0}_{p-1}^T)^T$
$\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$	$\begin{bmatrix} 5000 & 0.1 & 0.1 & \cdots & 0.1 \\ 0.1 & 0.1 & 0.1 & \cdots & 0.1 \\ 0.1 & 0.1 & 0.1 & \cdots & 0.1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0.1 & 0.1 & 0.1 & \cdots & 0.1 \end{bmatrix}_{p \times p}$	$\begin{bmatrix} 0.1 & 0.1 & 0.1 & \cdots & 0.1 \\ 0.1 & 5000 & 0.1 & \cdots & 0.1 \\ 0.1 & 0.1 & 0.1 & \cdots & 0.1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0.1 & 0.1 & 0.1 & \cdots & 0.1 \end{bmatrix}_{p \times p}$
$\sigma_1^2 = \sigma_2^2$	0.1	0.1
K	2	2
D	20	20
n_k	50	50

242 to p . In each dataset, 50 samples are generated for each class, from which 25
243 samples are selected as the training set and the rest as the test set, i.e. n_1
244 and n_2 are fixed to 25 for all the datasets. The 20 n_k/p ratios are 1.5, 1, 0.7,
245 0.5, 0.3, 0.1, 0.09, 0.08, 0.07, 0.06, 0.05, 0.04, 0.03, 0.02, 0.01, 0.009, 0.008,
246 0.007, 0.006 and 0.005; and the corresponding p 's are 17, 25, 36, 50, 83, 250,
247 278, 313, 417, 500, 625, 833, 1250, 2500, 2778, 3125, 3571, 4167 and 5000.
248 Among these settings, $n_k/p = 1.5$ (i.e. $p = 17$) indicates a low-dimensional
249 dataset while other ratios indicate high-dimensional datasets.

250 It is clear in Table 1 that the only difference between simulation A and
251 simulation B is the values of $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$, which determines the importance
252 of $\sum_{i=q+1}^p (t_i^{k,new})^2$ for classification. In both simulations, the first dimen-
253 sions of the feature vectors contain major discriminative information since
254 $\mu_{11} = 0$ and $\mu_{21} = 10$, while other dimensions contain little discriminative
255 information since $\mu_{1i} = \mu_{2i} = 0$ ($i \neq 1$). Therefore, the variance of the
256 first dimension determines how the discriminative information between two

257 classes is distributed to the PCs. The discriminative information left in the
 258 residuals for classification is determined by the discriminative information in
 259 the first few PCs used in the class subspace.

260 If the first dimension has the largest variance and the discriminative in-
 261 formation is concentrated on the first PC which is definitely used in the class
 262 subspace, i.e. $(\Sigma_1)_{11} = (\Sigma_2)_{11} = 5000$ in simulation A, then $\sum_{j=1}^p (e_j^{k,new})^2$
 263 is not very discriminative (or say unimportant for classification) and so is
 264 $\sum_{i=q+1}^p (t_i^{k,new})^2$. In contrast, if the first dimension has a small variance and
 265 contributes randomly to the PCs, i.e. $(\Sigma_1)_{11} = (\Sigma_2)_{11} = 0.1$ in simulation B,
 266 then the discriminative information may not be concentrated on the first few
 267 PCs that are used in the class subspace. In this case, $\sum_{j=1}^p (e_j^{k,new})^2$ can be
 268 discriminative (or say important for classification) and so be $\sum_{i=q+1}^p (t_i^{k,new})^2$.

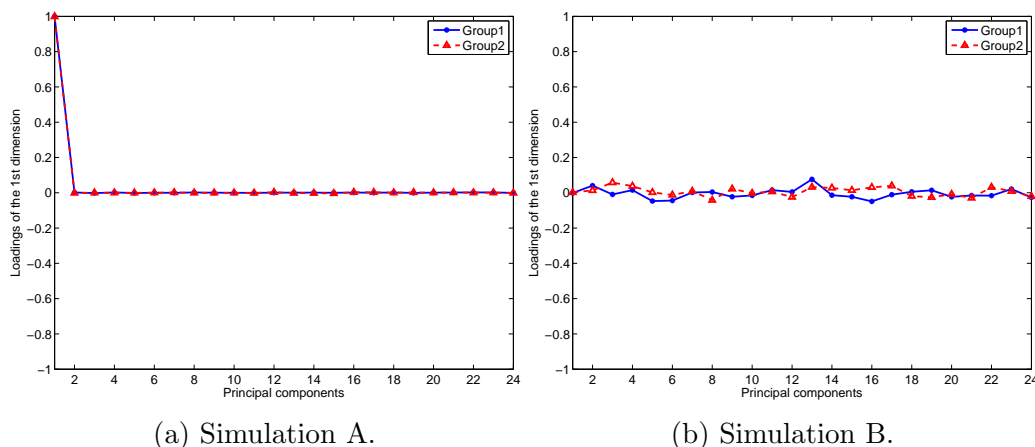


Figure 1: The loading plots of the first dimension.

269 Here we show an example to demonstrate the above argument. Two
 270 datasets with $p = 1250$ are generated. Applying PCA separately to the two
 271 classes of each dataset, we obtain the PCs for each class. We record the

272 first entries of all the PCs in each class, i.e. $\mathbf{V}_q(1, :)$, and plot them against
273 the PCs sorted in decreasing order of singular values, as shown in Figure 1
274 for simulation A and simulation B, respectively. These loadings indicate the
275 contributions of the first dimensions of the feature vectors to the PCs.

276 In simulation A, the absolute loadings of the first PC are close to one while
277 those of other PCs are close to zeros, which indicates that the discriminative
278 information between the two classes is concentrated on the the first PC.
279 Since the first PC is definitely used to build the class subspace, $\sum_{j=1}^p (e_j^{k,new})^2$
280 contains little discriminative information from the first dimension. Thus, as
281 a part of $\sum_{j=1}^p (e_j^{k,new})^2$, $\sum_{i=q+1}^p (t_i^{k,new})^2$ is not important for classification.

282 In simulation B, the loadings are distributed randomly around zero, which
283 indicates that the discriminative information is spread over all PCs. There-
284 fore, $\sum_{j=1}^p (e_j^{k,new})^2$ may contain discriminative information important for
285 classification and so be $\sum_{i=q+1}^p (t_i^{k,new})^2$.

286 4.3.2. Real datasets

287 A low-dimensional dataset (the iris data) and three high-dimensional
288 datasets (the Phenyl data, the meat data and the fat data) are used in
289 the experiments.

290 The iris dataset [12] contains 150 samples with three classes: each class
291 contains 50 samples. Each sample is described by four features.

292 The Phenyl dataset is provided in the R package, ‘chemometrics’. The
293 dataset consists of 600 mass spectrum of chemical components, with 300
294 compounds contain the phenyl substructure and 300 compounds do not con-
295 tain the substructure. Each spectra contains 658 mass spectral features. We
296 randomly select 100 samples from the Phenyl dataset for our experiments,

297 with 50 contain the phenyl substructure and 50 do not contain the structure.

298 The meat dataset [1] consists of 108 spectra of meat spectra measured at
299 1051 wavelengths, with 55 chicken samples and 54 turkey samples.

300 The fat dataset [11] consists of 193 spectra of finely chopped meat, with
301 122 meat samples of less than 20% fat and 71 samples of larger than 20%
302 fat. Each spectrum is measured at 100 wavelengths.

303 4.4. *Experiment settings*

304 For the iris data and the Phenyl data, we randomly select 25 samples
305 from each class to generate the training set. For the meat data, we randomly
306 select 27 chicken samples and 27 turkey samples for training. For the fat
307 data, we randomly select 35 samples of less than 20% fat and 35 samples
308 of larger than 20% fat for training. The remaining samples of each dataset
309 generate the test set.

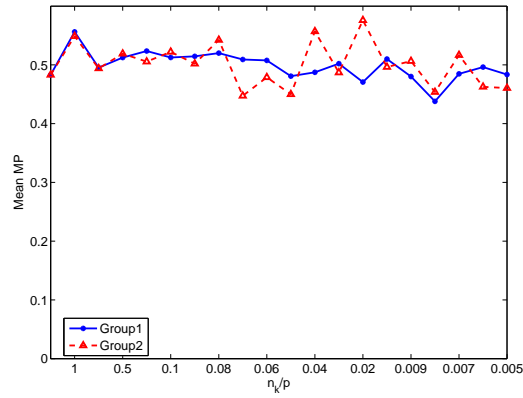
310 We repeat this procedure 100 times and perform the two methods, SIMCA
311 and SIMCA-D, on each training-test split.

312 In both methods, the number of PCs are chosen using the criterion that
313 the variance explained is more than 85% for all classes. Thus the numbers
314 of PCs, r , are the same for the two methods.

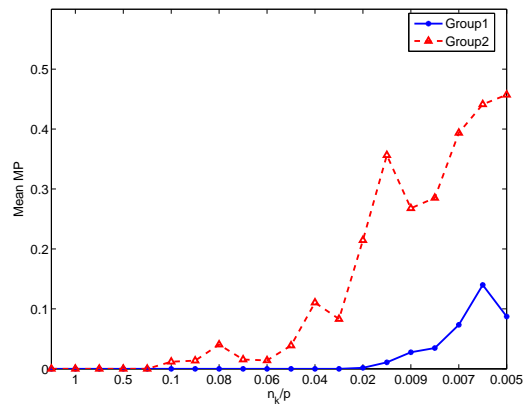
315 4.5. *Results*

316 4.5.1. *Simulated datasets*

317 To explore the effect of the n_k/p ratio on the performances of SIMCA
318 and SIMCA-D, we plot the the mean MP against the n_k/p ratio in Figure 2
319 for simulation A and simulation B, respectively. It is clear that the mean
320 MPs of SIMCA and SIMCA-D are the same when $n_k/p = 1.5$, i.e. in the



(a) Simulation A.



(b) Simulation B.

Figure 2: The plots of mean MP against n_k/p .

321 low-dimensional situation, in each of the simulation settings, as indicated by
 322 the leftmost points in each panel of Figure 2.

323 However, the relative performances of SIMCA and SIMCA-D are different
 324 for the two simulations when $n_k/p \leq 1$, i.e. in the high-dimensional situation.

325 In simulation A, the mean MPs of the two methods are similar for all n_k/p
 326 ratios, as shown in Figure 2a. This indicates that ignoring $\sum_{i=q+1}^p (t_i^{k,new})^2$
 327 in the calculation of the OD does not affect the classification results in this
 328 simulation, because in this case $\sum_{i=q+1}^p (t_i^{k,new})^2$ is not important for classifi-
 329 cation. In addition, since the residuals are not discriminative, the mean MP
 330 varies around 0.5.

331 In simulation B, the difference between the mean MPs of the two methods
 332 becomes larger as n_k/p becomes smaller (i.e. when the data are higher di-
 333 mensional), as shown in Figure 2b. Since in this simulation the first few PCs
 334 used in class subspaces contain little discriminative information, the residual
 335 $\sum_{j=1}^p (e_j^{k,new})^2$ is important for classification. SIMCA performs pretty well
 336 for almost all the n_k/p ratios because $\sum_{j=1}^p (e_j^{k,new})^2$ captures the discrimi-
 337 native information for classification. In contrast, SIMCA-D, which only uses
 338 $\sum_{i=r+1}^q (t_i^{k,new})^2$ for classification and ignores $\sum_{i=q+1}^p (t_i^{k,new})^2$, cannot capture
 339 the discriminative information in $\sum_{i=q+1}^p (t_i^{k,new})^2$ and can be suboptimal in
 340 classification, especially when n_k/p is small (i.e. when the data dimension is
 341 high). For example, the mean MP of SIMCA-D worsens to around 0.4 when
 342 n_k/p decreases to 0.008.

343 In addition for simulation B, we show an example of how $\sum_{i=q+1}^p (t_i^{k,new})^2$
 344 affects the classification performance using the Coomans' plots. Figure 3
 345 shows the Coomans' plots of the test samples on one training-test split of

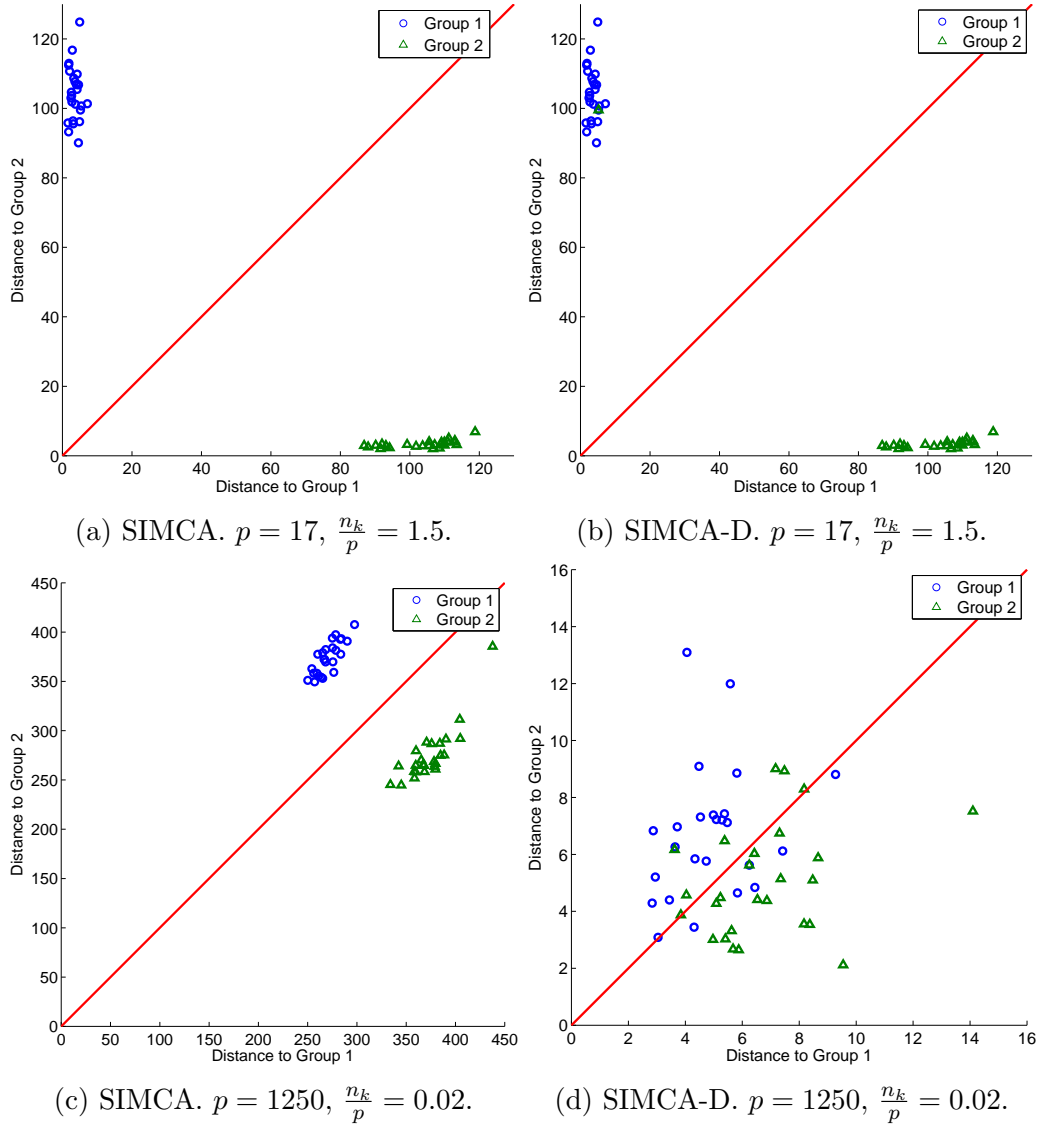


Figure 3: Coomans' plots.

346 each simulated dataset. The Coomans' plot [26] shows the orthogonal dis-
 347 tance from the test samples to two class subspaces at the same time. In
 348 our experiments, the horizontal and vertical axes denote the ODs to Group
 349 1 and Group 2, respectively. In Figure 3, the red reference line divides the
 350 Coomans' plot into two parts: in the upper triangular part, the distance to
 351 Group 1 is smaller than that to Group 2; in the lower triangular part, it is
 352 the other way around.

353 Since SIMCA and SIMCA-D have the same q and r , the Coomans' plots
 354 reflect the difference between the ODs of these two methods.

355 When $n_k/p = 1.5$ (i.e. low-dimensional), the Coomans' plots of the two
 356 methods are the same. When $n_k/p = 0.02$ (i.e. high-dimensional), the Coomans'
 357 plots of the two methods are different. We observe large differences between
 358 the values of ODs in Figure 3c and Figure 3d, which indicates that the value
 359 of $\sum_{i=q+1}^p (t_i^{k,new})^2$ is large. Including $\sum_{i=q+1}^p (t_i^{k,new})^2$ can perfectly separate
 360 the two groups as shown in Figure 3c; however, omitting $\sum_{i=q+1}^p (t_i^{k,new})^2$ re-
 361 sults in a mixture of the two groups as shown in Figure 3d. This indicates
 362 that the additional term $\sum_{i=q+1}^p (t_i^{k,new})^2$ is important for classification in this
 363 high-dimensional simulated dataset.

364 4.5.2. Real datasets

365 Figure 4 shows the box plots of the MP for the real datasets. In the high-
 366 dimensional Phenyl data and the high-dimensional meat data, SIMCA-D
 367 provides worse classification performance than the original SIMCA. However,
 368 in the high-dimensional fat data, SIMCA-D and SIMCA provides the same
 369 classification results. The results suggest that SIMCA-D can provide worse
 370 classification results than SIMCA for some high-dimensional real datasets. In

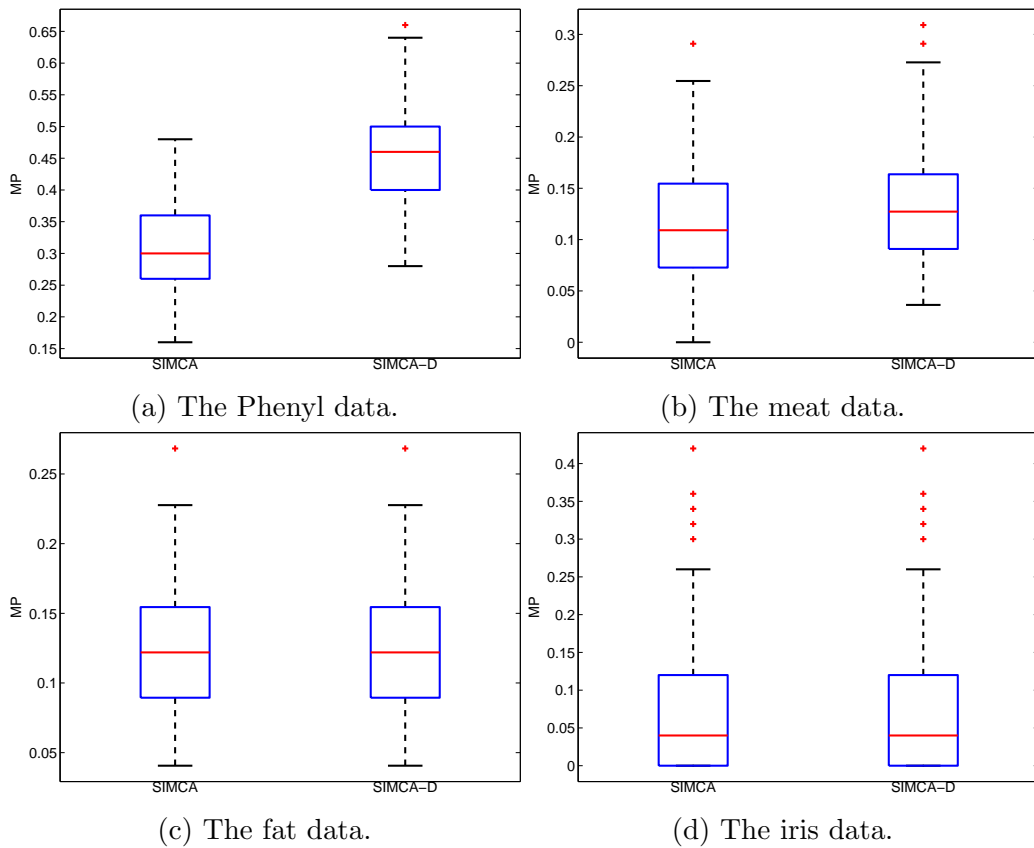


Figure 4: The box plots of the MP for the real datasets.

371 the low-dimensional iris dataset, the two methods provide the same results.
372 This pattern for the real datasets is consistent with that for the simulated
373 datasets.

374 **5. Conclusion**

375 We have investigated the formulae in [9] of calculating two ODs, $v^{k,l}$ and
376 $v^{k,new}$. We have shown that the formula for $v^{k,new}$ in [9] is not valid for high-
377 dimensional data (i.e. when $n_k \leq p$). The experiments on both the simulated
378 datasets and the real datasets have confirmed that the formula following [9]
379 can result in worse classification performance than the original one in [28].
380 Therefore, we suggest that the original formulae in [28] for calculating the
381 ODs, rather than the formulae in [9], should be used for the classification
382 of high-dimensional data which have more features than samples (i.e. when
383 $n_k \leq p$).

384 **Acknowledgment**

385 The authors would like to thank the reviewers for their constructive com-
386 ments.

387 **References**

- 388 [1] T. Arnalds, J. McElhinney, T. Fearn, G. Downey, A hierarchical dis-
389 criminant analysis for species identification in raw meat by visible and
390 near infrared spectroscopy, *Journal of Near Infrared Spectroscopy* 12 (3)
391 (2004) 183–188.

- 392 [2] S. Bicciato, A. Luchini, C. Di Bello, PCA disjoint models for multiclass
393 cancer analysis using gene expression data, *Bioinformatics* 19 (5) (2003)
394 571–578.
- 395 [3] K. V. Branden, M. Hubert, Robust classification in high dimensions
396 based on the SIMCA method, *Chemometrics and Intelligent Laboratory*
397 *Systems* 79 (1) (2005) 10–21.
- 398 [4] A. Candolfi, R. De Maesschalck, D. Jouan-Rimbaud, P. Hailey, D. Mas-
399 sart, The influence of data pre-processing in the pattern recognition of
400 excipients near-infrared spectra, *Journal of Pharmaceutical and Biomed-*
401 *ical Analysis* 21 (1) (1999) 115–132.
- 402 [5] A. Candolfi, R. De Maesschalck, D. Massart, P. Hailey, A. Harring-
403 ton, Identification of pharmaceutical excipients using NIR spectroscopy
404 and SIMCA, *Journal of Pharmaceutical and Biomedical Analysis* 19 (6)
405 (1999) 923–935.
- 406 [6] Q. Chen, J. Zhao, H. Zhang, X. Wang, Feasibility study on qualita-
407 tive and quantitative analysis in tea by near infrared spectroscopy with
408 multivariate calibration, *Analytica Chimica Acta* 572 (1) (2006) 77–84.
- 409 [7] N. C. da Silva, M. F. Pimentel, R. S. Honorato, M. Talhavini, A. O. Mal-
410 daner, F. A. Honorato, Classification of Brazilian and foreign gasolines
411 adulterated with alcohol using infrared spectroscopy, *Forensic science*
412 *international* 253 (2015) 33–42.
- 413 [8] M. Daszykowski, K. Kaczmarek, I. Stanimirova, Y. Vander Heyden,

- 414 B. Walczak, Robust SIMCA-bounding influence of outliers, *Chemomet-*
415 *rics and Intelligent Laboratory Systems* 87 (1) (2007) 95–103.
- 416 [9] R. De Maesschalck, A. Candolfi, D. Massart, S. Heuerding, Decision
417 criteria for soft independent modelling of class analogy applied to near
418 infrared data, *Chemometrics and Intelligent Laboratory Systems* 47 (1)
419 (1999) 65–77.
- 420 [10] R. De Maesschalck, D. Jouan-Rimbaud, D. L. Massart, The Maha-
421 lanobis distance, *Chemometrics and Intelligent Laboratory Systems*
422 50 (1) (2000) 1–18.
- 423 [11] F. Ferraty, P. Vieu, *Nonparametric Functional Data Analysis: Theory*
424 *and Practice*, Springer Science & Business Media, 2006.
- 425 [12] R. A. Fisher, The use of multiple measurements in taxonomic problems,
426 *Annals of Eugenics* 7 (2) (1936) 179–188.
- 427 [13] L. Jiao, F. Shang, F. Wang, Y. Liu, Fast semi-supervised clustering
428 with enhanced spectral embedding, *Pattern Recognition* 45 (12) (2012)
429 4358–4369.
- 430 [14] N. Kumar, A. Bansal, G. Sarma, R. K. Rawal, *Chemometrics tools used*
431 *in analytical chemistry: An overview*, *Talanta* 123 (2014) 186–199.
- 432 [15] B. Mnassri, E. M. EI Adel, B. Ananou, M. Ouladsine, Fault detection
433 and diagnosis based on PCA and a new contribution plot, *IFAC Pro-*
434 *ceedings Volumes* 42 (8) (2009) 834–839.

- 435 [16] B. Mnassri, M. Ouladsine, et al., Reconstruction-based contribution ap-
436 proaches for improved fault diagnosis using principal component analy-
437 sis, *Journal of Process Control* 33 (2015) 60–76.
- 438 [17] A. L. Pomerantsev, Acceptance areas for multivariate classification de-
439 rived by projection methods, *Journal of Chemometrics* 22 (11-12) (2008)
440 601–609.
- 441 [18] A. L. Pomerantsev, O. Y. Rodionova, Concept and role of extreme ob-
442 jects in PCA/SIMCA, *Journal of Chemometrics* 28 (5) (2014) 429–438.
- 443 [19] A. L. Pomerantsev, O. Y. Rodionova, On the type II error in SIMCA
444 method, *Journal of Chemometrics* 28 (6) (2014) 518–522.
- 445 [20] M. Rafferty, X. Liu, D. M. Lavery, S. McLoone, Real-time multiple
446 event detection and classification using moving window PCA, *IEEE*
447 *Transactions on Smart Grid* 7 (5) (2016) 2537–2548.
- 448 [21] O. Y. Rodionova, P. Oliveri, A. L. Pomerantsev, Rigorous and compli-
449 ant approaches to one-class classification, *Chemometrics and Intelligent*
450 *Laboratory Systems* 159 (2016) 89–96.
- 451 [22] R. Shang, Z. Zhang, L. Jiao, C. Liu, Y. Li, Self-representation based
452 dual-graph regularized feature selection clustering, *Neurocomputing* 171
453 (2016) 1242–1253.
- 454 [23] R. Shang, Z. Zhang, L. Jiao, W. Wang, S. Yang, Global discriminative-
455 based nonnegative spectral clustering, *Pattern Recognition* 55 (2016)
456 172–182.

- 457 [24] V. Uríčková, J. Sádecká, Determination of geographical origin of alco-
458 holic beverages using ultraviolet, visible and infrared spectroscopy: A
459 review, *Spectrochimica Acta Part A: Molecular and Biomolecular Spec-*
460 *troscopy* 148 (2015) 131–137.
- 461 [25] P. Van den Kerkhof, J. Vanlaer, G. Gins, J. F. Van Impe, Analysis
462 of smearing-out in contribution plot based fault isolation for statistical
463 process control, *Chemical Engineering Science* 104 (2013) 285–293.
- 464 [26] B. G. Vandeginste, D. L. Massart, *Handbook of Chemometrics and*
465 *Qualimetrics*, Elsevier Science, 1998.
- 466 [27] E. E. Waddell, M. R. Williams, M. E. Sigman, Progress toward the
467 determination of correct classification rates in fire debris analysis II:
468 utilizing soft independent modeling of class analogy (SIMCA), *Journal*
469 *of Forensic Sciences* 59 (4) (2014) 927–935.
- 470 [28] S. Wold, Pattern recognition by means of disjoint principal components
471 models, *Pattern Recognition* 8 (3) (1976) 127–139.
- 472 [29] R. Zhu, K. Fukui, J.-H. Xue, Building a discriminatively ordered sub-
473 space on the generating matrix to classify high-dimensional spectral
474 data, *Information Sciences* 382-383 (2017) 1–14.
- 475 [30] Y. Zontov, O. Y. Rodionova, S. Kucheryavskiy, A. Pomerantsev, DD-
476 SIMCA: A MATLAB GUI tool for data driven SIMCA approach,
477 *Chemometrics and Intelligent Laboratory Systems* 167 (2017) 23–28.