ISSN 1392–124X (print), ISSN 2335–884X (online) INFORMATION TECHNOLOGY AND CONTROL, 2016, T. 45, Nr. 4

Combined Classification Error Rate Estimator for The Fisher Linear Classifier

Mindaugas Gvardinskas

Department of System Analysis, Vytautas Magnus University
Vileikos St. 8, LT-44404 Kaunas, Lithuania
e-mail: m.gvardinskas@if.vdu.lt

crossref http://dx.doi.org/10.5755/j01.itc.45.4.14268

Abstract. Classification error rate estimation is one of the most important issues in machine learning and pattern recognition. This problem has been studied by many researchers and a number of error estimators have been proposed. However, theoretical analysis and empirical experiments show that most of these error estimation techniques are biased. One way to correct this bias is to use a linear combination of two different error rate estimators. In this paper we propose a new combined classification error rate estimator designed specially for the Fisher linear classifier. Experiments with real world and synthetic data sets show that resubstitution, leave-one-out, repeated 10-fold cross-validation, repeated 2-fold cross-validation, basic bootstrap, 0.632 bootstrap, zero bootstrap, D-method, DS-method and M-method are outperformed by the proposed combined error rate estimator (in terms of root-mean-square error).

Keywords: Error estimation; Classification; Resubstitution; Cross-validation; Bootstrap.

1. Introduction

Supervised machine learning is an important research area with many practical applications ranging from credit card fraud detection to image and language recognition [16, 32, 33, 34]. One key aspect of supervised learning is the evaluation of the induced classifier by means of any score or evaluation function. The most popular evaluation function is classification error which can be defined as the ratio between the number of incorrectly classified instances and the total number of instances. However, in most real-world situations, the true classification error of a classifier is unknown. Moreover, it can not be exactly calculated because the underlying probability distribution is unknown. So, it must be estimated from the given data. The problem of classification error rate estimation has been studied by many researchers and a number of error estimators have been proposed [5, 14, 17, 19, 20, 29, 30]. However, theoretical analysis and empirical experiments show that most of these error estimation techniques are biased [4, 8, 18, 20]. One way to deal with the problem of biased classification error rate estimation is to combine two different error rate estimators. A combined error rate estimator is an estimator of the form [28, 31]:

$$\hat{\varepsilon}_N = \omega\hat{\varepsilon}_N^{(1)} + (1-\omega)\hat{\varepsilon}_N^{(2)}, \qquad (1)$$

where $\hat{\varepsilon}_N^{(1)}$ and $\hat{\varepsilon}_N^{(2)}$ are error estimators, N is the training set size and $0 \le \omega \le 1$. If $\hat{\varepsilon}_N^{(1)}$ and $\hat{\varepsilon}_N^{(2)}$ are biased high (true classification error is overestimated) and low (true classification error is underestimated), respectively, then weight ω can be chosen so that the bias of estimator (1) is minimized. However, such an estimator requires prior knowledge about the underlying probability distribution, classification rule and sample size in order to derive the optimal weight ω [28]. In most cases, this information is unavailable and therefore, empirically chosen suboptimal weights are used [12, 13, 24, 28, 31].

In this paper we propose a new combined classification error rate estimator designed specially for the Fisher linear classifier. Contrary to most other combined estimators, the new method uses theoretically calculated fixed weight ω that is asymptotically optimal.

This paper is organized as follows. Section 2 presents basic definitions used throughout the paper, Fisher linear classifier, which is the basis of our method and also, most common classification error rate estimation techniques that are used as baseline methods in this study. The new error estimation method is introduced in section 3. Section 4 presents the results of our simulation study. The concluding remarks are in section 5.

2. Methods investigated

2.1. Basic definitions

Consider two category classification problem where class label $y \in \{0,\ 1\}$, feature vector $\mathbf{x} \in R^n$ and a classifier is a function f: $R^n \to \{0,\ 1\}$. An induction algorithm builds a classifier from a set of $N = N_1 + N_2$ independent observations $D_N = \{(\mathbf{x}_1, y_1), \ldots,\ (\mathbf{x}_N, y_N)\}$ drawn from some distribution *T*. Formaly, it is a mapping *g*: $\{R^n \times \{0,\ 1\}\}^N \times R^n \to \{0,\ 1\}$. Here N_1 is the number of observations from the first class and N_2 is the number of observations from the second class. The performance of a classifier is measured by conditional probability of misclassification (conditional PMC):

$$\varepsilon_N = P(g(D_N, \mathbf{x}) \neq y) \qquad (2)$$

Some authors call it true classification error rate or actual error rate [4, 27]. This error is conditioned on one particular training set D_N and induction algorithm *g*. In most real world pattern recognition problems conditional PMC is unknown, therefore an error estimator $\hat{\varepsilon}_N$ is used.

2.2. Fisher linear classifier

Fisher linear classifier is a well known classification method which is widely used in many fields, including medical diagnosis [10], robotics [7] and computer vision [6]. This classification rule can realize linear least squares and single layer perceptron classifiers [9, 25], also it bears strong connection with support vector machine classification technique [15]. Fisher classifier can be written as a linear discriminant function:

$$j(x) = \mathbf{V}^T \mathbf{x} + v_0 \qquad (3)$$

where

$$\mathbf{V} = \hat{\mathbf{\Sigma}}^{-1}(\hat{\mathbf{M}}_1 - \hat{\mathbf{M}}_2), \qquad (4)$$

$$v_0 = -\frac{1}{2}(\hat{\mathbf{M}}_1 + \hat{\mathbf{M}}_2)^T \mathbf{V} + \ln \frac{P_1}{P_2} \qquad (5)$$

here $\hat{\mathbf{\Sigma}}$ is a sample estimate of a common covariance matrix and $\hat{\mathbf{\Sigma}}^{-1}$ is inverse of $\hat{\mathbf{\Sigma}}$, $\hat{\mathbf{M}}_1$ and $\hat{\mathbf{M}}_2$ are the estimates of class mean vectors, P_1 and P_2 are class prior probabilities. A new pattern $\boldsymbol{x}$ is classified according to the sign of the discriminant function *j*.

2.3. Resubstitution

The resubstitution method is the simplest example of the class of nonparametric estimators. In this method, the whole data set is used as the training set and then reused as the test set. The resubstitution estimated error is defined as

$$\hat{\varepsilon}_N^{(R)} = \frac{1}{N} \sum_{i=1}^{N} | g(D_N, \mathbf{x}_i) - y_i | \qquad (6)$$

This method is known to have high bias, but low variance [4, 27].

2.4. Cross-validation

In *k*-fold cross-validation, the data set is randomly partitioned into *k* subsets of approximately equal size. Each subset is used as a test set and the remaining *k*-1 subsets are used as the training set. The cross-validation error estimate is defined as

$$\hat{\varepsilon}_N^{(CV)} = \frac{1}{N} \sum_{i=1}^{k} \sum_{j=1 \wedge (\mathbf{x}_j, y_j) \in D_i}^{N} | g(D_N \setminus D_i, \mathbf{x}_j) - y_j | \quad (7)$$

where D_i is the *i*-th fold of the data set D_N, *k* is the number of folds and *N* is the size of D_N. The tradeoff between bias and variance in cross-validation depends on *k* [18]. Cross-validation with small *k* values (5-2) typically have lower variance than cross-validation with large *k* values (10-*N*), however, estimators with small *k* values are more biased.

2.5. Bootstrap

Basic bootstrap estimator tries to correct the bias of resubstitution estimator. This bias can be expressed as $b = E[\varepsilon_N] - E[\hat{\varepsilon}_N^{(R)}]$. Since *b* is not known, it must be estimated from the given data set D_N. The estimation procedure is as follows. First, a bootstrap sample is formed by sampling *N* data points uniformly and with replacement from the original data set. Then, an induction algorithm is trained on the bootstrap sample and tested on the original data set D_N. These steps are repeated *r* times and the estimate of *b* is calculated as

$$\hat{b} = \frac{1}{r} \sum_{i=1}^{r} (\hat{\varepsilon}_N - \hat{\varepsilon}_N^{(R)^*}) \qquad (8)$$

where $\hat{\varepsilon}_N^{(R)^*}$ is resubstitution error on the bootstrap data and $\hat{\varepsilon}_N$ is conditional error estimate obtained by testing the classifier on the original data set D_N. The bootstrap estimate of the conditional error rate is given by [11]

$$\hat{\varepsilon}_N^{(B)} = \hat{\varepsilon}_N^{(R)} + \hat{b} \qquad (9)$$

where $\hat{\varepsilon}_N^{(R)}$ is resubstitution error on the original data set D_N. There are many variants of this basic bootstrap estimator. The one, which in various empirical studies has shown good performance is called the 0.632 bootstrap. Similar to basic bootstrap estimator, this method tries to correct the bias of zero bootstrap by doing a weighted average of resubstitution and zero bootstrap estimators [4]. The 0.632 bootstrap estimated error is defined as [12, 13]

$$\hat{\varepsilon}_N^{(0.632B)} = 0.632 \cdot \hat{\varepsilon}_N^{0B} + 0.368 \cdot \hat{\varepsilon}_N^{(R)} \qquad (10)$$

where $\hat{\varepsilon}_N^{(0B)}$ is zero bootstrap estimate

$$\hat{\varepsilon}_N^{(0B)} = \frac{1}{r}\sum_{i=1}^{r} \sum_{j=1 \wedge (\mathbf{x}_j, y_j) \in D_N \setminus D_{Bi}}^{N} | g(D_{Bi}, \mathbf{x}_j) - y_j | \qquad (11)$$

here r is the number of bootstrap samples and D_{Bi} is the i-th bootstrap sample.

2.6. D-method

It is the first parametric classification error rate estimator, proposed in statistical pattern recognition literature. For the homoscedastic (equal covariance matrices) normal model for two classes with equal prior probabilities, the D estimator is given by [14]

$$\hat{\varepsilon}_N^{(D)} = \Phi\left\{-\frac{\hat{\delta}}{2}\right\} \qquad (12)$$

where Φ is a standard Gaussian cumulative distribution function and $\hat{\delta} = \sqrt{(\hat{\mathbf{M}}_1 - \hat{\mathbf{M}}_2)^T \hat{\boldsymbol{\Sigma}}^{-1} (\hat{\mathbf{M}}_1 - \hat{\mathbf{M}}_2)}$ is an estimate of Mahalanobis distance. This method is known to have low variance but large bias [20].

2.7. DS-method

An estimate of the Mahalanobis distance $\hat{\delta}$ used in D estimator overestimates true Mahalanobis distance and this increases the bias of the above mentioned parametric error rate estimator. An unbiased estimator of the Mahalanobis distance δ is given by [20]

$$\hat{\delta}_{DS} = \sqrt{\frac{N-n-3}{N-2}\hat{\delta}} \qquad (13)$$

and the DS estimator can be expressed as

$$\hat{\varepsilon}_N^{(DS)} = \Phi\left\{-\frac{\hat{\delta}_{DS}}{2}\right\}. \qquad (14)$$

2.8. M-method

McLachlan proposed another parametric classification error rate estimator which assumes multivariate normality [23]. This estimator can be expressed as

$$\hat{\varepsilon}_{N_1}^{(M)} = \Phi\left\{-\frac{\hat{\delta}}{2}\right\} + \phi\{\frac{\hat{\delta}}{2}\}(a_1 + a_2 + a_3 + a_4 + a_5) \qquad (15)$$

where

$$a_1 = \frac{(n-1)}{\hat{\delta} N_1} \qquad (16)$$

$$a_2 = \frac{\hat{\delta}(4(4n-1) - \hat{\delta}^2)}{32(N-2)} \qquad (17)$$

$$a_3 = \frac{(n-1)(n-2)}{4\hat{\delta} N_1^2} \qquad (18)$$

$$a_4 = \frac{(n-1)(-\hat{\delta}^3 + 8(2n+1)\hat{\delta} + \frac{16}{\hat{\delta}})}{64 N_1 (N-2)} \qquad (19)$$

$$a_5 = \left(3\hat{\delta}^6 - 4(24n+7)\hat{\delta}^4 + 16(48n^2 - 48n - 53)\hat{\delta}^2 + 192(-8n+15)\right)\frac{\hat{\delta}}{12288\ (N-2)^2} \qquad (20)$$

here ϕ is standart normal density function. Note that by interchanging N_1 and N_2 in the expressions above for the first group, the corresponding error rate estimator for the second group is obtained.

2.9. Performance of error estimators

Commonly used performance measures of an error estimator $\hat{\varepsilon}_N$ are bias, deviation variance and root-mean-square error (RMS) [4, 8]. However, in most cases, the derivation of exact analytical expressions for the above mentioned performance measures is rather complicated, therefore, in practice, bias, deviation variance and RMS are calculated approximately, by using the following expressions:

$$Bias[\hat{\varepsilon}_N] \approx \frac{1}{MC}\sum_{i=1}^{MC}(\hat{\varepsilon}_{N(i)} - \varepsilon_{N(i)}) \qquad (21)$$

$$Var_{dev}[\hat{\varepsilon}_N] \approx \frac{1}{MC}\sum_{i=1}^{MC}(\hat{\varepsilon}_{N(i)} - \varepsilon_{N(i)})^2 - (\frac{1}{MC}\sum_{i=1}^{MC}(\hat{\varepsilon}_{N(i)} - \varepsilon_{N(i)}))^2 \qquad (22)$$

$$RMS[\hat{\varepsilon}_N] \approx \sqrt{\frac{1}{MC}\sum_{i=1}^{MC}(\hat{\varepsilon}_{N(i)} - \varepsilon_{N(i)})^2} \qquad (23)$$

where MC is the number of samples created by Monte-Carlo simulation, $\hat{\varepsilon}_{N(i)}$ is i-th estimate of conditional PMC based on the i-th sample and $\varepsilon_{N(i)}$ is i-th conditional PMC.

The bias measures whether, on average, the estimator overestimates or underestimates true conditional PMC, while deviation variance measures the variability of the estimator. Finally, root-mean-square error combines both, bias and the deviation variance into a single metric.

3. Proposed method

Unbiased combined classification error rate estimator can be expressed as

$$Bias[\hat{\varepsilon}_N] = \omega\ E[\hat{\varepsilon}_N^{(1)}] + (1-\omega)\ E[\hat{\varepsilon}_N^{(2)}] - E[\varepsilon_N] = 0\ . \qquad (24)$$

Now, assume that estimator $\hat{\varepsilon}_N^{(1)}$ is repeated k-fold cross-validation. In each run, repeated k-fold cross-validation uses $N^* = N - N/k$ vectors for classifier training, therefore we can write that $E[\hat{\varepsilon}_N^{(1)}] \approx E[\varepsilon_{N^*}]$. Also, suppose that estimator $\hat{\varepsilon}_N^{(2)}$ is resubstitution and it uses N vectors to estimate resubstitution error.

Now we can write that $E[\hat{\varepsilon}_N^{(2)}] = E[\varepsilon_N^R]$ and equation (24) can be rewritten as

$$Bias[\hat{\varepsilon}_N] \approx \omega\ E[\varepsilon_{N^*}] + (1-\omega)E[\varepsilon_N^R] - E[\varepsilon_N] \approx 0\ . \tag{25}$$

From (25) we have that

$$\omega \approx \frac{E[\varepsilon_N] - E[\varepsilon_N^R]}{E[\varepsilon_{N^*}] - E[\varepsilon_N^R]}. \tag{26}$$

Now, suppose that the following preconditions are met:

1. classifier deals with two multivariate Gaussian pattern classes;
2. the covariance matrix is the same for all classes;
3. class prior probabilities are equal;
4. the training set has the same number of patterns from each class;
5. Mahalanobis distance is constant;
6. the dimensionality n is fixed and very large;
7. both values, N, $N^* \to \infty$.

Then expected error of the Fisher linear classifier can be expressed as [25, 26]

$$E[\varepsilon_N] \approx \Phi\left\{-\frac{\delta}{2}\frac{1}{\sqrt{T_M T_\Sigma}}\right\} \tag{27}$$

and expected resubstitution error can be expressed as

$$E[\varepsilon_N^R] \approx \Phi\left\{-\frac{\delta}{2}\sqrt{T_M T_\Sigma}\right\} \tag{28}$$

where $T_M = 1 + \dfrac{4n}{\delta^2 N}$, $T_\Sigma = 1 + \dfrac{n}{N-n}$.

Finally, from (27) and (28) we get that the weight is

$$\omega \approx \lim_{\frac{n}{N}\to 0} \frac{E[\varepsilon_N] - E[\varepsilon_N^R]}{E[\varepsilon_{N^*}] - E[\varepsilon_N^R]} \approx \frac{2}{1+\dfrac{N}{N^*}}. \tag{29}$$

The derivation of expression (29) is based on the Taylor series expansion of $E[\varepsilon_N]$, $E[\varepsilon_{N^*}^R]$ and $E[\varepsilon_{N^*}]$. The proposed combined classification error rate estimator (PCE) is defined as:

$$\hat{\varepsilon}_N^{(PCE)} = \frac{1}{r}\sum_{i=1}^{r}(\omega \cdot \hat{\varepsilon}_N^{(CV)} + (1-\omega)\cdot \hat{\varepsilon}_N^{(R)}) \tag{30}$$

where r is the number of repetitions.

4. Simulation study

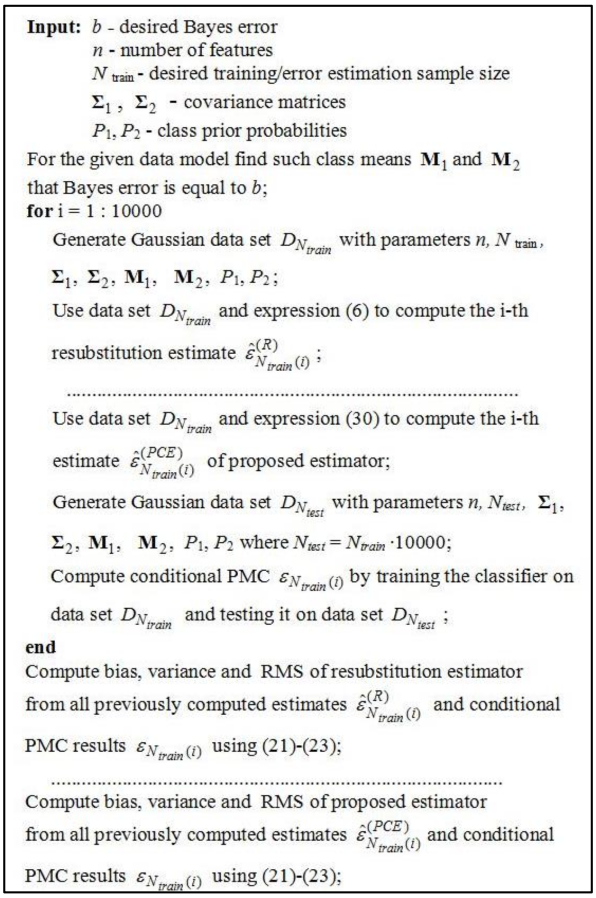
4.1. Experimental setup

Our experiments consist of two parts: synthetic experiments with Gaussian data and experiments with real world data sets. The error estimators studied are resubstitution (resub), leave-one-out (loo), repeated 10-fold cross-validation (rcv10), repeated 2-fold cross-validation (rcv2), basic bootstrap (bboot) 0.632 bootstrap (b0632), zero bootstrap (zboot), D-method (D), DS-method (DS), M-method (M) and proposed combined estimator that uses repeated 2-fold cross-validation and resubstitution as component estimators. To get a fair comparison, we made the number of classifiers built for each estimator equal, i.e. 200 (except resubstitution and leave-one-out estimators where the number of induced classifiers is fixed and cannot be changed). Therefore, in 0.632 bootstrap, zero bootstrap and basic bootstrap, the number of runs (r) is set to 200, in proposed combined estimator and repeated 2-fold cross-validation the number of runs is set to 100, in repeated 10-fold cross-validation the number of runs is set to 20. Finally, to make the simulations more realistic, we used small to moderate sample sizes (20-200).

4.2. Synthetic data

We use four data models to generate sample points. Data model 1 is two-class Gaussian data model with equally likely classes, common covariance matrix and class means located at $\mathbf{M}_1 = (m,\ m,\dots,\ m)^T$ and $\mathbf{M}_2 = (-m,\ -m,\dots,-m)^T$. The elements of the common covariance matrix are equal to 0.1, except the main diagonal, where elements are equal to 1. Data model 2 is similar to model 1. The only difference is that different class prior probabilities are used. In model 2, we use $P_1 = 0.7$, $P_2 = 0.3$. Data models 3 and 4 are similar to models 1 and 2, except that different covariance matrices $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ are used. The elements of main diagonal of $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ are set to 1, non diagonal elements of $\mathbf{\Sigma}_1$ are set to 0.2 and non diagonal elements of $\mathbf{\Sigma}_2$ are set to 0.4. For each data model, we choose four values of m such that Bayes error is from 0.05 to 0.20. In each of these 16 cases, 10000 independent samples of size N = 60 are generated (in all cases n = 20). A pseudo-code for synthetic simulations is presented in Algorithm 1.

Experimental results for data models 1-4 are displayed in Tables 1-4. The best bias, variance and RMS are marked in bold font for easier reading of the presented tables. Also, to better visualize the obtained results, RMS values from Tables 1-4 are additionally provided in Fig. 1-4. Our experiments show that the least biased error estimation method is leave-one-out while repeated 10-fold cross-validation, M-method, zero bootstrap, 0.632 bootstrap and proposed combined estimator are moderately biased. The most biased error estimation methods are resubstitution, D-method, DS-method, basic bootstrap and repeated 2-fold cross-validation. Also, we can see that the proposed method works well in correcting the bias of repeated 2-fold cross-validation. The situation with other bias correcting classification error rate estimators is similar. However, there is one exception: when the Bayes error is high, 0.632 bootstrap fails in correcting the bias of zero bootstrap. The experiments also show that repeated 10-fold cross-validation, leave-one-out, M-method, zero bootstrap and basic bootstrap are more variable than the proposed combined estimator, repeated 2-fold cross-validation, 0.632 bootstrap, D-method, DS-method and resubstitution. Also, we can see that the proposed estimator outperforms resubstitution, D-method, DS-method, basic bootstrap, zero bootstrap and 2-fold cross-validation (in RMS sense). However, the situation with other error estimators is different. When $\varepsilon_{Bayes} = 0.05$, repeated 10-fold cross-validation , leave-one-out, M-method and 0.632 bootstrap are better than proposed combined estimator and when $\varepsilon_{Bayes} \geq 0.10$, the proposed method outperforms the above mentioned methods.



```
Input: $b$ - desired Bayes error
       $n$ - number of features
       $N_{\text{train}}$ - desired training/error estimation sample size
       $\mathbf{\Sigma}_1$ , $\mathbf{\Sigma}_2$ - covariance matrices
       $P_1, P_2$ - class prior probabilities
For the given data model find such class means $\mathbf{M}_1$ and $\mathbf{M}_2$
that Bayes error is equal to $b$;
for i = 1 : 10000
    Generate Gaussian data set $D_{N_{train}}$ with parameters $n, N_{\text{train}}$,
    $\mathbf{\Sigma}_1, \mathbf{\Sigma}_2, \mathbf{M}_1, \mathbf{M}_2, P_1, P_2$;
    Use data set $D_{N_{train}}$ and expression (6) to compute the i-th
    resubstitution estimate $\hat{\varepsilon}^{(R)}_{N_{train}(i)}$ ;
    ......................................................................................
    Use data set $D_{N_{train}}$ and expression (30) to compute the i-th
    estimate $\hat{\varepsilon}^{(PCE)}_{N_{train}(i)}$ of proposed estimator;
    Generate Gaussian data set $D_{N_{test}}$ with parameters $n, N_{test}$, $\mathbf{\Sigma}_1$,
    $\mathbf{\Sigma}_2, \mathbf{M}_1, \mathbf{M}_2, P_1, P_2$ where $N_{test} = N_{train} \cdot 10000$;
    Compute conditional PMC $\varepsilon_{N_{train}(i)}$ by training the classifier on
    data set $D_{N_{train}}$ and testing it on data set $D_{N_{test}}$ ;
end
Compute bias, variance and RMS of resubstitution estimator
from all previously computed estimates $\hat{\varepsilon}^{(R)}_{N_{train}(i)}$ and conditional
PMC results $\varepsilon_{N_{train}(i)}$ using (21)-(23);
......................................................................................
Compute bias, variance and RMS of proposed estimator
from all previously computed estimates $\hat{\varepsilon}^{(PCE)}_{N_{train}(i)}$ and conditional
PMC results $\varepsilon_{N_{train}(i)}$ using (21)-(23);
```

Algorithm 1: synthetic simulation scheme

Table 1. Simulation results, data model 1

		resub	loo	rcv10	b0632	**Proposed combined estimator**	bboot	D	DS	M	rcv2	zboot
Bayes error 0.05	Bias	-0.0875	**0.0012**	0.0077	-0.0045	0.0380	0.0613	-0.0854	-0.0579	-0.0205	0.1009	0.0501
	Variance	$7.3 \cdot 10^{-4}$	0.0025	0.0022	0.0012	0.0014	0.0029	$\mathbf{5.7 \cdot 10^{-4}}$	$8.6 \cdot 10^{-4}$	0.0019	0.0022	0.0022
	RMS	0.0916	0.0497	0.0477	**0.0353**	0.0535	0.0817	0.0887	0.0649	0.0484	0.1110	0.0687
Bayes error 0.10	Bias	-0.1283	**-0.0018**	0.0089	-0.0228	0.0240	0.0784	-0.1256	-0.0848	-0.0155	0.1003	0.0532
	Variance	0.0014	0.0037	0.0033	0.0018	0.0020	0.0043	**0.0011**	0.0015	0.0033	0.0026	0.0029
	RMS	0.1336	0.0606	0.0578	**0.0478**	0.0503	0.1021	0.1298	0.0930	0.0593	0.1123	0.0760
Bayes error 0.15	Bias	-0.1572	**0.0022**	0.0091	-0.0398	0.0090	0.0902	-0.1544	-0.1062	-0.0097	0.0923	0.0512
	Variance	0.0020	0.0045	0.0040	0.0021	0.0023	0.0051	**0.0015**	0.0019	0.0041	0.0027	0.0033
	RMS	0.1635	0.0675	0.0638	0.0611	**0.0486**	0.1152	0.1593	0.1146	0.0649	0.1059	0.0771
Bayes error 0.20	Bias	-0.1794	**0.0024**	0.0085	-0.0557	-0.0061	0.0985	-0.1770	-0.1252	-0.0046	0.0807	0.0466
	Variance	0.0025	0.0053	0.0045	0.0024	0.0024	0.0056	**0.0019**	0.0021	0.0047	0.0027	0.0035
	RMS	0.1862	0.0728	0.0677	0.0740	**0.0499**	0.1237	0.1823	0.1332	0.0685	0.0960	0.0753

Table 2. Simulation results, data model 2

		resub	loo	rcv10	b0632	**Proposed combined estimator**	bboot	D	DS	M	rcv2	zboot
Bayes error 0.05	Bias	-0.0912	**0.0020**	0.0083	-0.0067	0.0354	0.0623	-0.0891	-0.0604	-0.0201	0.0989	0.0499
	Variance	7.9·10-4	0.0026	0.0023	0.0012	0.0014	0.0031	**6.1·10-4**	9·10-4	0.0021	0.0021	0.0023
	RMS	0.0954	0.0506	0.0486	**0.0359**	0.0518	0.0833	0.0925	0.0675	0.0497	0.1092	0.0689
Bayes error 0.10	Bias	-0.1343	**0.0029**	0.0091	-0.0279	0.0182	0.0789	-0.1315	-0.0892	-0.0144	0.0947	0.0507
	Variance	0.0015	0.0039	0.0034	0.0018	0.0020	0.0045	**0.0012**	0.0015	0.0035	0.0025	0.0030
	RMS	0.1398	0.0623	0.0591	0.0507	**0.0478**	0.1033	0.1359	0.0974	0.0609	0.1070	0.0743
Bayes error 0.15	Bias	-0.1646	0.0029	0.0085	-0.0478	**4.6·10-5**	0.0902	-0.1619	-0.1125	-0.0080	0.0825	0.0460
	Variance	0.0022	0.0048	0.0042	0.0021	0.0023	0.0053	**0.0017**	0.0019	0.0044	0.0026	0.0033
	RMS	0.1710	0.0693	0.0650	0.0665	**0.0475**	0.1159	0.1670	0.1208	0.0667	0.0967	0.0734
Bayes error 0.20	Bias	-0.1895	0.0034	0.0071	-0.0681	-0.0200	0.0974	-0.1872	-0.1349	**-0.0021**	0.0651	0.0376
	Variance	0.0027	0.0057	0.0047	0.0023	0.0024	0.0057	**0.0021**	0.0021	0.0050	0.0025	0.0033
	RMS	0.1964	0.0752	0.0689	0.0835	**0.0526**	0.1232	0.1926	0.1426	0.0709	0.0819	0.0690

Table 3. Simulation results, data model 3

		resub	loo	rcv10	b0632	**Proposed combined estimator**	bboot	D	DS	M	rcv2	zboot
Bayes error 0.05	Bias	-0.0948	**0.0016**	0.0085	-0.0063	0.0383	0.0675	-0.0931	-0.0624	-0.0183	0.1049	0.0535
	Variance	$8.7\cdot10^{-4}$	0.0027	0.0024	0.0013	0.0015	0.0032	**$6.8\cdot10^{-4}$**	$9.9\cdot10^{-4}$	0.0022	0.0023	0.0024
	RMS	0.0993	0.0519	0.0498	**0.0372**	0.0549	0.0883	0.0967	0.0699	0.0506	0.1152	0.0724
Bayes error 0.10	Bias	-0.1387	**0.0022**	0.0095	-0.0275	0.0208	0.0851	-0.1361	-0.0919	-0.0120	0.1006	0.0549
	Variance	0.0016	0.0040	0.0035	0.0019	0.0021	0.0046	**0.0013**	0.0016	0.0036	0.0026	0.0031
	RMS	0.1445	0.0634	0.0600	0.0517	**0.0501**	0.1088	0.1406	0.1004	0.0610	0.1128	0.0780
Bayes error 0.15	Bias	-0.1707	0.0026	0.0093	-0.0481	**0.0014**	0.0965	-0.1678	-0.1170	-0.0061	0.0875	0.0502
	Variance	0.0023	0.0049	0.0042	0.0023	0.0024	0.0053	**0.0018**	0.0020	0.0044	0.0026	0.0034
	RMS	0.1773	0.0697	0.0655	0.0676	**0.0486**	0.1210	0.1729	0.1252	0.0663	0.1015	0.0767
Bayes error 0.20	Bias	-0.1961	0.0029	0.0082	-0.0683	-0.0191	0.1034	-0.1935	-0.1409	**$-9\cdot10^{-4}$**	0.0696	0.0419
	Variance	0.0028	0.0055	0.0047	0.0024	0.0025	0.0057	**0.0021**	0.0021	0.0049	0.0025	0.0034
	RMS	0.2031	0.0743	0.0690	0.0842	**0.0530**	0.1280	0.1988	0.1481	0.0702	0.0859	0.0721

Table 4. Simulation results, data model 4

		resub	loo	rcv10	b0632	**Proposed combined estimator**	bboot	D	DS	M	rcv2	zboot
Bayes error 0.05	Bias	-0.0919	**0.0022**	0.0088	-0.0061	0.0371	0.0655	-0.0899	-0.0597	-0.0159	0.1018	0.0520
	Variance	$8.3\cdot10^{-4}$	0.0026	0.0024	0.0013	0.0015	0.0032	**$6.6\cdot10^{-4}$**	$9.6\cdot10^{-4}$	0.0023	0.0022	0.0023
	RMS	0.0964	0.0515	0.0495	**0.0366**	0.0535	0.0865	0.0936	0.0673	0.0501	0.1122	0.0710
Bayes error 0.10	Bias	-0.1405	**0.0030**	0.0098	-0.0304	0.0170	0.0851	-0.1372	-0.0921	-0.0065	0.0961	0.0527
	Variance	0.0018	0.0041	0.0036	0.0019	0.0021	0.0048	**0.0014**	0.0017	0.0039	0.0026	0.0031
	RMS	0.1466	0.0643	0.0608	0.0535	**0.0488**	0.1095	0.1421	0.1008	0.0627	0.1087	0.0766
Bayes error 0.15	Bias	-0.1754	0.0033	0.0086	-0.0556	-0.0067	0.0981	-0.1727	-0.1212	**0.0028**	0.0780	0.0452
	Variance	0.0025	0.0051	0.0044	0.0023	0.0024	0.0055	**0.0019**	0.0021	0.0048	0.0026	0.0034
	RMS	0.1825	0.0717	0.0666	0.0735	**0.0493**	0.1231	0.1783	0.1295	0.0697	0.0931	0.0735
Bayes error 0.20	Bias	-0.2066	**0.0031**	0.0060	-0.0836	-0.0356	0.1038	-0.2045	-0.1528	0.0110	0.0501	0.0308
	Variance	0.0031	0.0059	0.0048	0.0024	0.0024	0.0057	0.0023	**0.0021**	0.0055	0.0023	0.0032
	RMS	0.2139	0.0768	0.0697	0.0968	**0.0604**	0.1283	0.2100	0.1596	0.0752	0.0695	0.0647

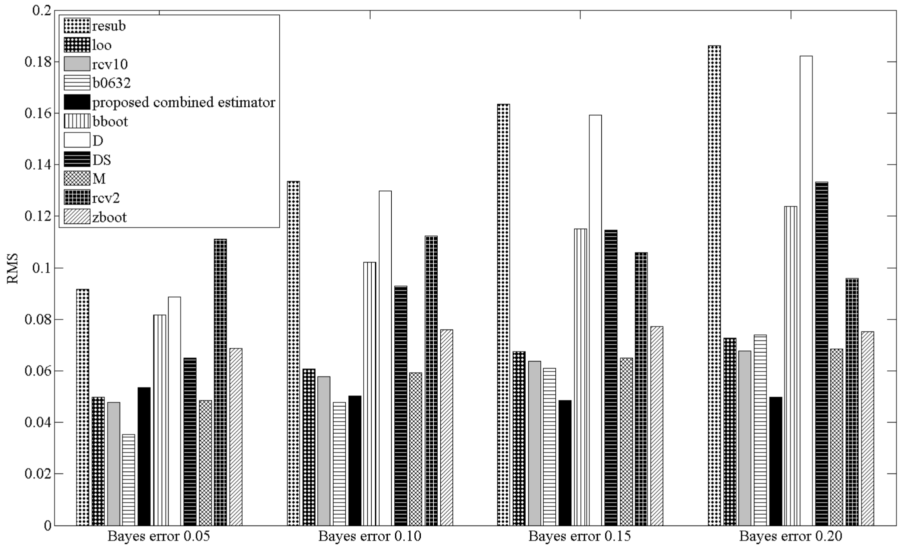


Figure 1. RMS results, data model 1

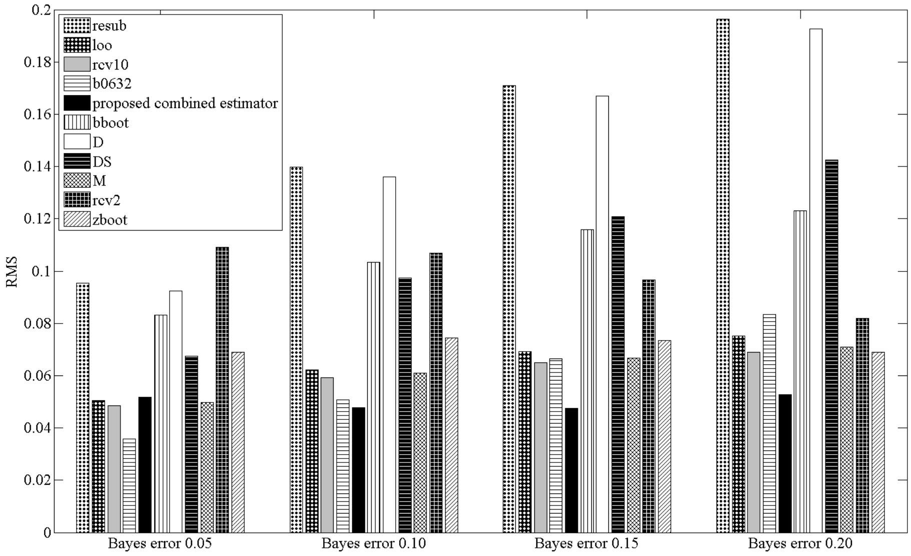


Figure 2. RMS results, data model 2

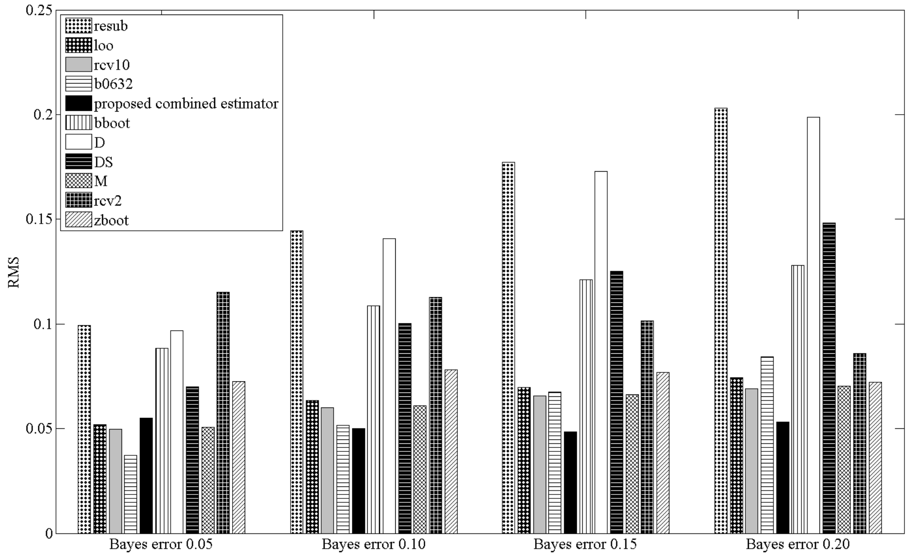


Figure 3. RMS results, data model 3

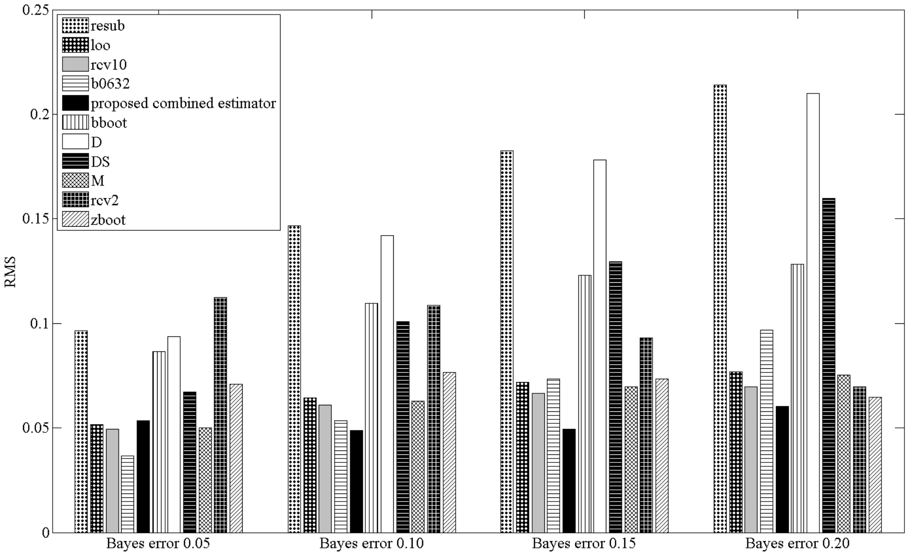


Figure 4. RMS results, data model 4

4.3. Real data

In most real world scenarios true class-conditional densities are unknown. As a result, there is no way to generate new independent data points or very large test sets. Therefore, in all real data experiments approximately independent data vectors and relatively small test sets are used. A pseudo-code for real data simulations is presented below:

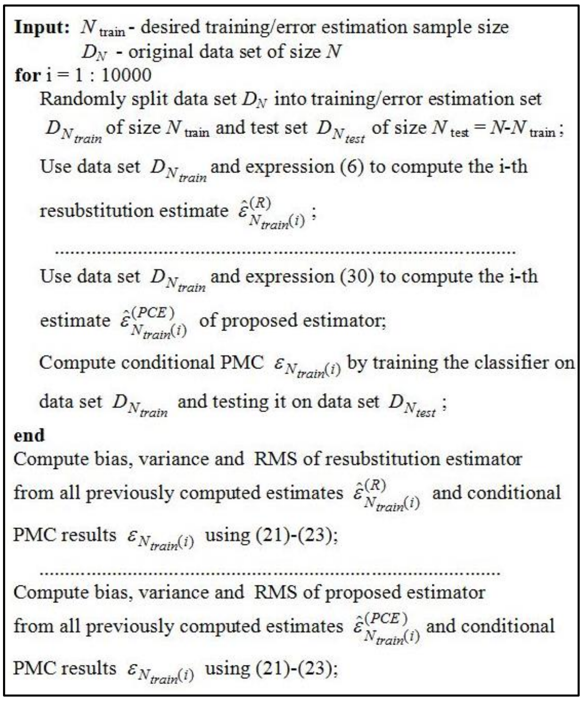
Real world experiments were conducted on eight non Gaussian data sets (according to Mardia's and Henze-Zirkler tests) from the UCI machine learning repository:

Magic Gamma Telescope data set [2]. This data set is generated to simulate registration of high energy gamma particles in a ground-based atmospheric Cherenkov gamma telescope. The data set contains 19020 instances, of which 12332 are classified as signal and 6688 are classified as background. Each instance has 10 features. The size of the training/error estimation data set is 32.

Pima Indian Diabetes data [2]. It consists of 768 instances that are diabetes positive (268) or diabetes negative (500). The number of features is 8. Training/ error estimation sample size is set to 32.

MiniBoone particle identification data [2]. This data set is taken from the MiniBoone experiment and is used to distinguish electron neutrinos from muon neutrinos. The database is composed of 130064 instances of which 36499 are electron neutrinos and 93565 are muon neutrinos. The number of features is 50. Training/error estimation sample size is 200.

Skin data set [3]. This data set is collected by randomly sampling B,G,R values from face images of various age groups, race groups, and genders obtained from FERET database and PAL database. Skin data set is composed of 245057 instances of which 50859 are skin instances and 194198 are non-skin instances. The number of features is 3. Training/error estimation sample size is set to 20.



```
Input: $N_{\text{train}}$ - desired training/error estimation sample size
       $D_N$ - original data set of size $N$
for i = 1 : 10000
    Randomly split data set $D_N$ into training/error estimation set
    $D_{N_{train}}$ of size $N_{\text{train}}$ and test set $D_{N_{test}}$ of size $N_{\text{test}} = N\text{-}N_{\text{train}}$;
    Use data set $D_{N_{train}}$ and expression (6) to compute the i-th
    resubstitution estimate $\hat{\varepsilon}^{(R)}_{N_{train}(i)}$;
    ................................................................................
    Use data set $D_{N_{train}}$ and expression (30) to compute the i-th
    estimate $\hat{\varepsilon}^{(PCE)}_{N_{train}(i)}$ of proposed estimator;
    Compute conditional PMC $\varepsilon_{N_{train}(i)}$ by training the classifier on
    data set $D_{N_{train}}$ and testing it on data set $D_{N_{test}}$;
end
Compute bias, variance and RMS of resubstitution estimator
from all previously computed estimates $\hat{\varepsilon}^{(R)}_{N_{train}(i)}$ and conditional
PMC results $\varepsilon_{N_{train}(i)}$ using (21)-(23);
    ................................................................................
Compute bias, variance and RMS of proposed estimator
from all previously computed estimates $\hat{\varepsilon}^{(PCE)}_{N_{train}(i)}$ and conditional
PMC results $\varepsilon_{N_{train}(i)}$ using (21)-(23);
```

Algorithm 2: simulation scheme

Climate Model Simulation Crashes data set [21]. This data set is composed of 540 instances and each instance is represented by 18 climate model input parameter values. The goal is to predict simulation outcomes (fail or succeed) from the input parameters. There are 46 instances classified as simulation crashes and 494 instances that are classified as not simulation crashes. Training/error estimation sample size is set to 50.

Diabetic Retinopathy data [1]. This data set contains 1151 instances, of which 540 are classified as diabetic retinopathy and 611 are classified as not diabetic retinopathy. The number of features is 19 and training/error estimation sample size is set to 50.

QSAR biodegradation database [22]. This data set describes 1055 molecules that are ready biodegradable (356) or not ready biodegradable (699). The original data set contains 41 features, however, in order to avoid non-invertible singular covariance matrix, the number of features is reduced to 20 (linear correlation based feature selection). Training/error estimation sample size is 40.

Spambase data set [2]. This database is composed of 4601 instances of which 1813 are classified as spam and 2788 are classified as non-spam. The number of features is 20 (after feature selection). Training/error estimation sample size is set to 80.

Table 5 displays experimental results based on real world data sets. The best bias, the best variance and the best RMS is marked in bold font for easier reading of the presented tables. Also, to better visualize the obtained results, RMS values from Table 5 are additionally provided in Fig. 5-6. The experiments show that leave-one-out, repeated 10-fold cross-validation, M-method and the proposed combined estimator generally have smaller bias than resubstitution, D-method, DS-method, basic bootstrap, 0.632 bootstrap, zero bootstrap and repeated 2-fold cross-validation. Also, we can see that DS-method corrects the bias of D-method, basic bootstrap corrects the bias of resubstitution and proposed combined estimator corrects the bias of repeated 2-fold cross-validation. The only one exception is 0.632 bootstrap method which fails in correcting the bias of zero bootstrap estimator. The experiments also show that proposed combined estimator, repeated 2-fold cross-validation, 0.632 bootstrap, D-method, DS-method and resubstitution are less variable than repeated 10-fold cross-validation, leave-one-out, M-method, zero bootstrap and basic bootstrap. Also, we can see that the proposed estimator generally outperforms other error estimation techniques (in RMS sense) and the only exception is Skin data set, where 0.632 bootstrap performs better.

Table 5. real data results

		Resub	Loo	rcv10	b0632	**Proposed combined estimator**	bboot	D	DS	M	rcv2	zboot
Magic Gamma Telescope	Bias	-0.1709	0.0071	0.0075	-0.0566	-0.0144	0.0975	-0.1700	-0.1169	**$-3\cdot10^{-4}$**	0.0640	0.0422
	Variance	0.0046	0.0097	0.0084	0.0043	0.0046	0.0102	**0.0039**	0.0042	0.0101	0.0052	0.0063
	RMS	0.1839	0.0988	0.0917	0.0868	**0.0696**	0.1403	0.1812	0.1337	0.1006	0.0965	0.0896
Pima Indian Diabetes	Bias	-0.1496	0.0076	0.0072	-0.0546	-0.0203	0.0825	-0.1475	-0.1054	**-0.0046**	0.0445	0.0307
	Variance	0.0055	0.0099	0.0085	0.0048	0.0052	0.0103	0.0042	**0.0041**	0.0087	0.0055	0.0065
	RMS	0.1671	0.0999	0.0926	0.0882	**0.0747**	0.1307	0.1610	0.1234	0.0934	0.0868	0.0864
Mini Boone	Bias	-0.0835	0.0012	0.0058	-0.0190	0.0110	0.0554	-0.0897	-0.0604	**-0.0001**	0.0583	0.0321
	Variance	$4.1\cdot10^{-4}$	$8.4\cdot10^{-4}$	$7.4\cdot10^{-4}$	$4.7\cdot10^{-4}$	$5.5\cdot10^{-4}$	0.0010	**$3.3\cdot10^{-4}$**	$4.2\cdot10^{-4}$	$8.7\cdot10^{-4}$	$7.1\cdot10^{-4}$	$7.2\cdot10^{-4}$
	RMS	0.0860	0.0292	0.0280	0.0289	**0.0260**	0.0640	0.0915	0.0638	0.0296	0.0641	0.0419
Skin	Bias	-0.0348	0.0046	0.0065	-0.0079	**$-2.1\cdot10^{-4}$**	0.0246	-0.0258	-0.0114	-0.0019	0.0171	0.0139
	Variance	**0.0029**	0.0046	0.0046	0.0030	0.0034	0.0060	0.0049	0.0064	0.0097	0.0042	0.0042
	RMS	0.0642	0.0683	0.0682	**0.0551**	0.0587	0.0813	0.0746	0.0811	0.0983	0.0668	0.0663
Climate Model Simulation Crashes	Bias	-0.1401	0.0071	0.0022	-0.0352	**$-3.82\cdot10^{-4}$**	0.0566	-0.1360	-0.0940	-0.0255	0.0697	0.0372
	Variance	0.0019	0.0049	0.0039	0.0020	0.0021	0.0043	**0.0016**	0.0020	0.0045	0.0025	0.0029
	RMS	0.1467	0.0706	0.0624	0.0572	**0.0456**	0.0867	0.1419	0.1041	0.0714	0.0855	0.0657
Diabetic Retinopathy	Bias	-0.1239	**0.0028**	0.0042	-0.0461	-0.0161	0.0664	-0.1167	-0.0885	0.0266	0.0380	0.0238
	Variance	0.0037	0.0063	0.0054	0.0034	0.0035	0.0060	0.0026	**0.0024**	0.0042	0.0037	0.0043
	RMS	0.1380	0.0794	0.0734	0.0744	**0.0610**	0.1019	0.1274	0.1014	0.0702	0.0721	0.0695
QSAR Biodegradation	Bias	-0.1388	**0.0031**	0.0047	-0.0467	-0.0113	0.0803	-0.1323	-0.0924	0.0258	0.0529	0.0339
	Variance	0.0045	0.0079	0.0069	0.0042	0.0043	0.0081	0.0035	**0.0034**	0.0068	0.0048	0.0055
	RMS	0.1541	0.0891	0.0831	0.0797	**0.0668**	0.1207	0.1451	0.1092	0.0866	0.0869	0.0817
Spambase	Bias	-0.0900	**$1.7\cdot10^{-4}$**	0.0045	-0.0210	0.0086	0.0571	-0.0987	-0.0675	-0.0088	0.0579	0.0330
	Variance	**0.0012**	0.0023	0.0020	0.0012	0.0013	0.0024	0.0012	0.0015	0.0029	0.0015	0.0017
	RMS	0.0962	0.0478	0.0450	0.0409	**0.0372**	0.0750	0.1045	0.0778	0.0550	0.0700	0.0529

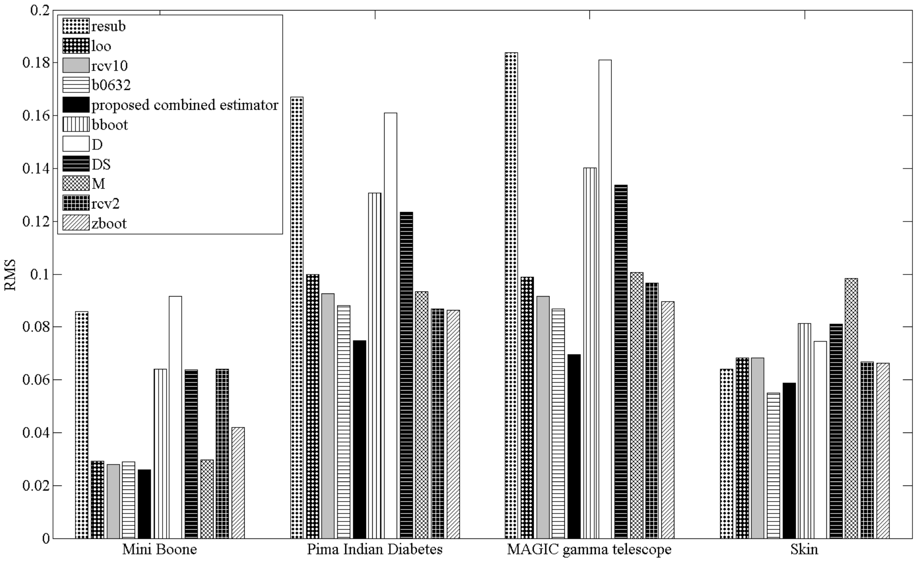


Figure 5. RMS results, Mini Boone, Pima Indian Diabetes, Magic Gamma Telescope and Skin data sets

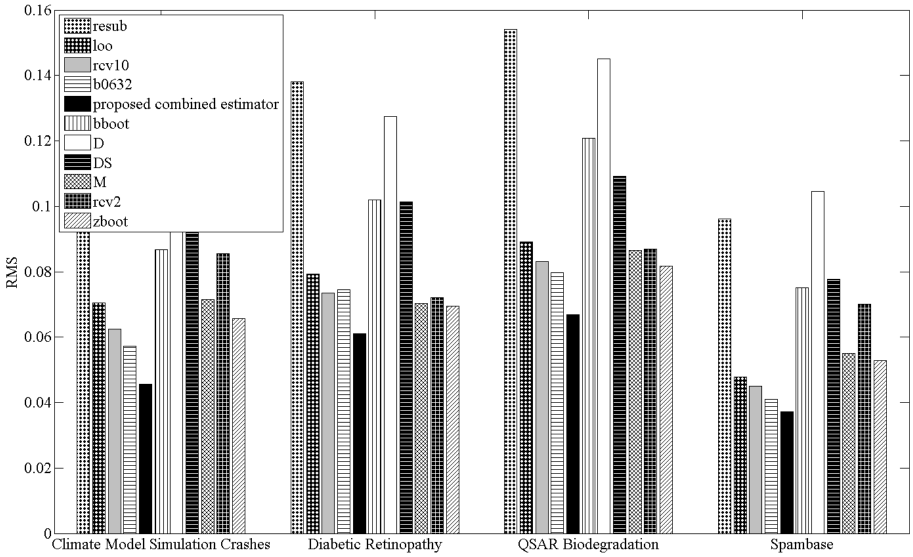


Figure 6. RMS results, Climate Model Simulation Crashes, Diabetic Retinopathy, QSAR Biodegradation and Spambase data sets

5. Conclusion

In this paper we have proposed a new classification error rate estimator designed specially for the Fisher linear classifier. The proposed method approximates unbiased combined classification error rate estimator by using fixed weight that is calculated from asymptotic approximations of expected classification and resubstitution errors of the Fisher linear classifier. This weight is computed by assuming that the classifier deals with two multivariate Gaussian pattern classes (1), all classes share the same covariance matrix (2), class prior probabilities are equal (3), the training set has the same number of patterns from each class (4) Mahalanobis distance is constant (5), dimensionality is fixed and very large (6) and sample size approaches infinity (7). When these assumptions are violated, proposed method may fail to correct the bias of cross validation. However, experimental results show that bias correction works well, even when some of the above preconditions are not met. Although proposed weight ω does not minimize RMS of the combined error rate estimator, it allows to construct estimator that has better RMS than most other error estimation techniques.

References

[1] **B. Antal, A. Hajdu**. An ensemble-based system for automatic screening of diabetic retinopathy. *Knowledge-Based Systems*, 2014, Vol. 60, 20-27.

[2] **K. Bache, M Lichman.** UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: Uni-

versity of California, School of Information and Computer Science, 2015.

[3] **R. B. Bhatt, G. Sharma, A. Dhall, S. Chaudhury**. Efficient skin region segmentation using low complexity fuzzy decision tree model. In: *Proceedings of the sixth IEEE India International Conference*, 2009, pp. 1-4.

[4] **U. Braga-Neto, E. Dougherty**. Is cross-validation valid for small sample microarray classification? *Bioinformatics*, 2004, Vol. 20, No. 3, 374-380.

[5] **V. L. Brailovskij.** An object recognition algorithm with many parameters and its applications. *Engineering Cybernetics*, 1964, Vol. 2, 22-30.

[6] **M. Breukelen, R. P. V. Duin, D. M. J. Tax, J. E. Hartog.** Handwritten digit recognition by combined classifiers. *Kybernetika,* 1998, Vol. 34, No. 4, 381-386.

[7] **Y. Chen, H. Wang, J. Zhang, G. Garty, N. Simaan, Y. L. Yao, D. J. Brenner**. Automated recognition of robotic manipulation failures in high-throughput biodosimetry tool. *Expert Systems with Applications*, 2012, Vol. 39, No. 10, 9602-9611.

[8] **E. Dougherty, C. Sima., J. Hua., B. Hanczar, U. Braga-Neto.** Performance of error estimators for classification. *Current Bioinformatics*, 2010, Vol. 5, No. 1, 53-67.

[9] **R. Duda, P. Hart, D. Stork.** Pattern classification. *Wiley*, 2000.

[10] **S. Dudoit, J. Fridlyand, T. P. Speed.** Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 2002, Vol. 97, Issue 457, 77–87.

[11] **B. Efron.** Bootstrap Methods: Another look at the jackknife. *Annals of Statistics*, 1979, Vol. 7, No. 1, 1-26.

[12] **B. Efron.** Estimating the error rate of a prediction rule: improvement on cross-validation. *Journal of the American Statistical Association*, 1983, Vol. 78, No. 382, 316–331.

[13] **B. Efron, R. Tibshirani.** Improvements on cross-validation: The .632+ bootstrap method. *Journal of the American Statistical Association*, 1997, Vol. 92, No. 438, 548-560.

[14] **R. Fisher.** The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 1936, Vol. 7, 179–188.

[15] **T. Gestel, J. Suykens, B. Baesens, S. Viaene, J. Vanthienen, G. Dedene, B. Moor, J. Vandewalle.** Benchmarking Least Squares Support Vector Machine Classifiers. *Machine Learning,* 2004, Vol. 54, No. 1, 5-32.

[16] **G. Guodong, S. Li, C. Kapluk.** Face recognition by support vector machines. In: *Proceedings of the fourth IEEE International Conference on Automatic Face and Gesture Recognition,* 2000, pp. 196–201.

[17] **M. Gvardinskas.** Weighted Classification Error Rate Estimator for the Euclidean Distance Classifier. In: *Proceedings of the 21st International Conference on Information and Software Technologies* (*ICIST*), 2015, pp. 343-355.

[18] **R. Kohavi.** A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 1995, pp. 1137-1143.

[19] **W. J. Krzanowski, D. J. Hand.** Assessing error rate estimators: the leaving-one-out reconsidered. *Australian Journal of Statistics*, 1997, Vol. 39, No. 1, 1997, 35-46.

[20] **P. Lachenbruch, R. Mickey.** Estimation of error rates in discriminant analysis. *Technometrics*, 1968, Vol. 10, No. 1, 1-11.

[21] **D. D. Lucas, R. Klein, J. Tannahill, D. Ivanova, S. Brandon, D. Domyancic, Y. Zhang**. Failure analysis of parameter-induced simulation crashes in climate models. *Geoscientific Model Development*, 2013, Vol. 6, No. 4, 1157-1171.

[22] **K. Mansouri, T. Ringsted, D. Ballabio, R. Todeschini, V. Consonni.** Quantitative Structure - Activity Relationship models for ready biodegradability of chemicals. *Journal of Chemical Information and Modeling*, 2013, Vol. 53, No. 4, 867-878.

[23] **G. J. McLachlan.** An asymptotic unbiased technique for estimating the error rates in discriminant analysis. *Biometrics*, 1974, Vol. 30, 239-249.

[24] **G. J. McLachlan.** A note on the choice of a weighting function to give an efficient method for estimating the probability of misclassification. *Pattern Recognition*, 1977, Vol. 9, 147-149.

[25] **S. Raudys.** Statistical and Neural Classifiers. An Integrated Approach to Design. *Springer-Verlag*, London, 2001.

[26] **S. Raudys, D. M. Young.** Results in statistical discriminant analysis: A review of the former Soviet Union literature. *Journal of Multivariate Analysis*, 2004, Vol. 89, No. 1, 1–35.

[27] **R. A. Schiavo, D. J. Hand.** Ten more years of error rate Research. *International Statistical Review,* 2000, Vol. 68, No. 3, 295–310.

[28] **C. Sima, E. Dougherty.** Optimal convex error estimators for classification. *Pattern Recognition*, 2006, Vol. 39, No. 9, 1763-1780.

[29] **C. Smith.** Some examples of discrimination. *Annals of Eugenics*, 1947, Vol. 18, 272–282.

[30] **M. Stone.** Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society*, 1974, Vol. 36, No. 2, 111–147.

[31] **G. Toussaint, P. Sharpe.** An efficient method for estimating the probability of misclassification applied to a problem in medical diagnosis. *Computers in Biology and Medicine*, 1975, Vol. 4, 269–278.

[32] **V. Uloza, A. Verikas, M. Bacauskiene, A. Gelzinis, R. Pribuisiene, M. Kaseta, V. Saferis.** Categorizing Normal and Pathalogical Voices: Automated and Perceptual Categorization. *Journal of Voice,* 2011, Vol. 25, No. 6, 700-708.

[33] **A. Verikas, A. Gelzinis, M. Bacauskiene, M. Hallander, V. Uloza, M. Kaseta.** Combining image, voice, and the patient's questionnaire data to categorize laryngeal disorders. *Artificial Intelligence in Medicine*, 2010, Vol. 49, No. 1, 43-50.

[34] **S. Viaene, R. Derrig, G. Dedene**. A Case Study of Applying Boosting Naive Bayes to Claim Fraud Diagnosis. *IEEE Transactions on Knowledge and Data Engineering*, 2004, Vol. 16, No. 5, 612-620.

Received October 2015.