Flow Cytometry-Based Classification in Cancer Research: A View on Feature Selection

S. Sakira Hassan[1], Pekka Ruusuvuori[2,3], Leena Latonen[3] and Heikki Huttunen[1]

[1]Department of Signal Processing, Tampere University of Technology, Tampere, Finland. [2]Pori Department, Tampere University of Technology, Pori, Finland. [3]BioMediTech, University of Tampere, Tampere, Finland.

Supplementary Issue: Statistical Systems Theory in Cancer Modeling, Diagnosis, and Therapy

ABSTRACT: In this paper, we study the problem of feature selection in cancer-related machine learning tasks. In particular, we study the accuracy and stability of different feature selection approaches within simplistic machine learning pipelines. Earlier studies have shown that for certain cases, the accuracy of detection can easily reach 100% given enough training data. Here, however, we concentrate on simplifying the classification models with and seek for feature selection approaches that are reliable even with extremely small sample sizes. We show that as much as 50% of features can be discarded without compromising the prediction accuracy. Moreover, we study the model selection problem among the ℓ_1 regularization path of logistic regression classifiers. To this aim, we compare a more traditional cross-validation approach with a recently proposed Bayesian error estimator.

KEYWORDS: AML, leukemia, flow cytometry, logistic regression, error estimation, model selection

SUPPLEMENT: Statistical Systems Theory in Cancer Modeling, Diagnosis, and Therapy

CITATION: Hassan et al. Flow Cytometry-Based Classification in Cancer Research: A View on Feature Selection. *Cancer Informatics* 2015:14(S5) 75–85 doi: 10.4137/CIN.S30795.

TYPE: Original Research

RECEIVED: November 18, 2015. **RESUBMITTED:** February 01, 2016. **ACCEPTED FOR PUBLICATION:** February 07, 2016.

ACADEMIC EDITOR: J. T. Efird, Editor in Chief

PEER REVIEW: Six peer reviewers contributed to the peer review report. Reviewers' reports totaled 1860 words, excluding any confidential comments to the academic editor.

FUNDING: Authors disclose no external funding sources.

COMPETING INTERESTS: Authors disclose no potential conflicts of interest.

CORRESPONDENCE: sakira.hassan@tut.fi

Introduction

Flow cytometry enables quantitative measurement of single-cell properties through visible and fluorescent light in a high-throughput manner. The measured signals include fluorescence emission and light scatter. Flow cytometry has been routinely used for detecting malignancies from blood samples.[1] Through technological advances, measuring the combination of fluorescent signals from several different channels has enabled the use of high-dimensional data for studies, such as cytometric fingerprinting[2] and large-scale analysis of cell types.[3]

In this paper, we study the analysis of flow cytometry data from the feature selection point of view. More specifically, flow cytometry is able to produce large quantities of partially redundant measurement data, and the selection of important quantities within the large body of measurements is of interest. Moreover, a typical scenario contains large quantities of measurement data but may be limited to only a few patients. Thus, an ideal method would distil only the essential parts of the measurements from each patient, while producing reliable and well generalizing results when only a small amount of individuals is available in the training data.

We will concentrate our attention on two particular sets of flow cytometry data. The first set originates from the acute myeloid leukemia (AML) prediction challenge of the DREAM initiative in 2013.[4] The competition attracted a number of teams, and as several researchers use the data as part of their work, the challenge data have become a standard benchmark within the field. For example, Aghaeepour et al.[4] presents a large pool of analysis approaches from the DREAM challenge. Among classification methods presented in the literature, there are several sophisticated machine learning approaches, such as learning vector quantization,[5] correlative matrix mapping, and relative entropy differences.[6] The strength of data-driven approaches relying on supervised classification is their ability to handle high-dimensional data without requiring prior knowledge of the biological application.

The DREAM AML data represent a relatively large-scale experiment consisting of altogether 179 patients. Although it is a small number for traditional machine learning problems, the number of patients is unusually large for a biological study. To this aim, we use another dataset that represents a more commonly encountered sample size of 16 samples extracted from a prostate cancer cell line. This dataset, with two different treatments and a low number of samples, presents a nontrivial but common challenge for prediction and related feature selection. More information on the two datasets is provided in Data section.

In our earlier work,[7] we presented a supervised classification pipeline based on linear discriminant analysis (LDA) and logistic regression (LR) classifiers. Briefly, the method first

transforms the measurement data into higher dimensional space by generating combined features with multiplications and divisions between measurements. Following this mapping into higher dimensional space, LDA is used for lowering the dimensionality into a single value per measurement. Then, empirical distribution functions (EDFs) are constructed from LDA results for AML-positive and AML-negative sample classes and compared to training EDFs of both classes. The comparison results in two similarity values per group of measurements, and these results are fed to the LR classifier for a final AML prediction result. Our approach, together with alternative well-performing approaches,[4,5] represents a relatively complicated pipeline of somewhat arbitrary computation steps. Thus, our interest is to simplify these pipelines into a simple collection of obvious features, while still retaining a good accuracy.

Our approach here is to use LR classifier applied to summarize features, which are the mean and standard deviation of the measurements instead of the complete data. This reduces the number of features used in classification and, subsequently, also the model complexity. An essential part of classifier design is error estimation, which guides model selection.[8] Our strategy for model selection is to apply the recently introduced Bayesian error estimator (BEE).[9] We compare BEE model selection with a traditional 10-fold cross-validation (CV-10) error estimation, as well as with Bayesian information criterion (BIC)-based model selection, and conclude that the proposed approach enables accurate prediction for flow cytometry data with fewer measurements and a less complex classifier model than those previously presented in the literature.

The rest of this paper is organized as follows. In Materials and Methods section, we describe the data and methods used in this study and briefly discuss how feature selection is commonly done in machine learning. Experimental Results section presents the results of our experiments with different model and feature selection criteria for the materials introduced in Materials and Methods section. Finally, in Conclusions section, we summarize the work and discuss the conclusions of the results.

Materials and Methods

In this section, we describe the datasets used in this paper. We also give a brief overview of modeling method for feature selection. In addition to this, we introduce the state-of-the-art Bayesian error estimator (BEE) for model parameter selection. Finally, we present an example where the performance of BEE is benchmarked against other model selection criteria.

Data. In this work, we study two datasets: A larger set with 179 samples and a smaller set with 16 samples. These two case studies represent different classification challenges in terms of both application and sample size.

AML dataset. The flow cytometry dataset for the AML experiment has been collected from the DREAM6-FlowCAP2 challenge, which was organized by the DREAM project and the FlowCAP initiative (DREAM challenge AML dataset can be accessed from Aghaeepour et al).[4] We use the training dataset that consists of flow cytometry measurements of 179 patients. Among them, 23 patients are AML positive and the remaining 156 patients are AML negative. The flow cytometry measurement for each patient corresponds to seven jointly measured groups (hereafter called tubes) of seven quantities with a total of 49 biomarker measurements per cell. The biomarkers are summarized in Table 1 and include *Forward Scatter* in linear scale (FS Lin), *Sideward Scatter* in logarithmic scale (SS Log), and five fluorescence intensities (FL1–FL5) in logarithmic scales. For calibration purposes, FS Lin, SS Log, and CD45-ECD were measured for all tubes and the other 28 biomarkers were measured only in one tube.

Cancer cell line dataset. As another case study, we use flow cytometry data from a small sample setting. The data come from a prostate cancer cell line 22Rv1 stained with propidium iodide for cell cycle analysis.[10] The cells are transfected with miRNAs (either control or miR-193b) and induced to proliferate by overexpression of cyclin D. The data consist of 16 samples, with 8 samples (without cyclin D overexpression) with relatively consistent cell cycle profile and 8 samples (overexpressing cyclin D) with an altered cell cycle profile, ie, induced cell cycle activity with an increase in cells in DNA synthesis phase. The samples for both classes include four repetitions of two treatments, which are considered here to represent the same class. Each sample contains 12 measured channels, consisting of two scatter measurements and four fluorescence channels, both as area and height measurements.

Table 1. List of seven tubes with biomarkers provided in DREAM6 AML prediction data.

			FL1 LOG	FL2 LOG	FL3 LOG	FL4 LOG	FL5 LOG
Tube 1	FS Lin	SS Log	IgG1-FITC	IgG1-PE	CD45-ECD	IgG1-PC5	IgG1-PC7
Tube 2	FS Lin	SS Log	Kappa-FIT	Lambda-PE	CD45-ECD	CD19-PC5	CD20-PC7
Tube 3	FS Lin	SS Log	CD7-FITC	CD4-PE	CD45-ECD	CD8-PC5	CD2-PC7
Tube 4	FS Lin	SS Log	CD15-FITC	CD13-PE	CD45-ECD	CD16-PC5	CD56-PC7
Tube 5	FS Lin	SS Log	CD14-FITC	CD11c-PE	CD45-ECD	CD64-PC5	CD33-PC7
Tube 6	FS Lin	SS Log	HLA-DR-FITC	CD117-PE	CD45-ECD	CD34-PC5	CD38-PC7
Tube 7	FS Lin	SS Log	CD5-FITC	CD19-PE	CD45-ECD	CD3-PC5	CD10-PC7

Feature extraction. Several feature extraction methods can be used to obtain meaningful features from raw flow cytometry measurements. For instance, among widely used feature extraction techniques are methods based on principal component analysis and histogram computation. Biehl et al proposed statistical divergences to extract features that include moments, median, and interquartile range.[5] The length of the feature vector was 186 in this case. Another well-performed model was based on multidimensional entropic distance-based features.[4,5] Manninen et al.[7] expanded the cell measurements of each tube to a higher dimensional space. Following this transformation, LDA is used to lower the dimensionality into a single value for each measurement. These previous studies are summarized in Table 2. Table 2 also tabulates the test accuracy in terms of the area under the receiver operating characteristics (ROC) curve (AUC) measure over a single train/test split, which should not be interpreted as a definitive measure of accuracy, as the split of the samples is just one instance of all possible splits.

Table 2. Studies based on feature extraction strategies for DREAM AML challenge dataset.

	ACCURACY	SIZE OF FEATURE VECTOR	BRIEF DESCRIPTION
Biehl et al.[5]	1.00	186	Extraction of features with moments, median and interquartile and learning vector quantization is used for prediction
Vilar et al.[4]	1.00	31	Extraction of features with entropies and histogram based classifier is used for prediction
Manninen et al.[7]	1.00	(# of events) x 84	Expand features to higher dimension and then mapping to 1-D using linear discriminant analysis; logistic regression is used for prediction
Our solution (this study)	0.9989	49	Extraction of feature vector from means of measurements and applying regularized logistic regression for prediction
Our solution (this study)	0.9992	98	Extraction of feature vector from means and standard deviation of measurements and applying regularized logistic regression for prediction

Note: The accuracy is measured in terms of the AUC of a single train/test split.

In this paper, we use one of the simplest feature extraction techniques that include only the mean and the standard deviation of the each measurement. For the first dataset, the length of this extracted feature vector is 98, comprising 49 mean values and 49 standard deviations. As seen in the experiments of Experimental Results section, these features are sufficient to separate the classes without compromising the prediction accuracy. We will consider two versions of these basic features: the first feature set contains only the 49 mean values of the measurements, while the second feature vector considers both mean values and standard deviations, with altogether 98 features. The same approach is used with the smaller dataset, thus producing two different experimental cases. Before training the classifiers, we normalized all features to the interval (0, 1).

LR and regularization. LR is a discriminative method for modeling the class conditional probability densities by the logistic function. Given an observation matrix $\mathbf{X} \in \mathbb{R}^{N \times P}$ with N observations, P features, and corresponding class labels $\mathbf{Y} \in 1,\ldots,C$, we define LR model for the binary classification as,

$$p(y = 1 \mid x) = \frac{\exp(\beta_0 + \boldsymbol{\beta}^T x)}{1 + \exp(\beta_0 + \boldsymbol{\beta}^T x)} \quad (1)$$

and

$$p(y = 0 \mid x) = 1 - p(y = 1 \mid x) = \frac{1}{1 + \exp(\beta_0 + \boldsymbol{\beta}^T x)}. \quad (2)$$

Here, $\boldsymbol{x}$ represents a feature vector in the feature space corresponding to class label $y \in \{0,1\}$, β_0 is the intercept, and $\boldsymbol{\beta}$ represents coefficients of the logistic model. We can determine the model parameters β_0 and $\boldsymbol{\beta}$ from the training dataset by solving the ℓ_1 penalized LR problem,

$$\arg\min_{\beta_0, \boldsymbol{\beta}} \sum_{i=1}^{N} -\log p(y_i \mid \mathrm{x}_i) + \lambda \| \boldsymbol{\beta} \|_1, \quad (3)$$

where $\lambda > 0$ is the regularization parameter. When the number of training data is not larger compared to the number of features, ie, $P \gg N$, regularization is used to solve the overfitting problem.[11] In regularization, an extra term, λ, is added, which controls the trade-off between the loss function and the size of the coefficients. More recently, in feature selection, ℓ_1-regularized LR has received much attention, as it yields a sparse solution that has relatively few nonzero coefficients.[12] This minimization task is analogous to *least absolute shrinkage and selection operator* (Lasso) algorithm proposed by Tibshirani.[13] In addition to this, several extensions of Lasso have also been developed, such as grouped Lasso,[14,15] Dantzig selector,[16] elastic net,[17] and graphical Lasso.[18] In this paper,

we use the GLMNET algorithm by Friedman et al.[19] that combines the ℓ_2 and ℓ_1 penalties:

$$\arg\min_{\beta_0,\boldsymbol{\beta}} \sum_{i=1}^{N} -\log p(y_i \mid \mathbf{x}_i) + \lambda\left(\alpha \|\boldsymbol{\beta}\|_1 + (1-\alpha)\|\boldsymbol{\beta}\|_2\right), \quad (4)$$

where $\lambda > 0$ and $\alpha \in [0,1]$. The parameter α is a compromise between the ℓ_1 and ℓ_2 penalties, thereby determining the type of regularization. On the other hand, the regularization parameter λ controls the amount of regularization. A very large λ will completely shrinks the coefficients to zero and may yield a null or empty model.

In general, the model parameters λ and α are selected using the CV approach.[20] The dataset is randomly split into K mutually exclusive subsets of approximately equal sized. In K-fold CV, the process is iterated k times. At the kth iteration, the Kth fold is retained as test set and the remaining K − 1 folds are used as training set to train the model. Each of the K-folds is tested exactly once. The test set assesses the quality of the trained model. Then, the K results are combined or averaged to produce a single estimation of the model. The most commonly used values for K are 5 and 10. In this experiment, we set the value of $\alpha = 1$ and CV-10 is used for the selection of the model parameter λ and assessment of the model. As the type of regularization is determined by α, setting $\alpha = 0$ provides ℓ_2 penalty that is useful in cases, where the features are mutually correlated. On the other hand, $\alpha = 1$ provides sparse solution with fewer coefficients and, in turn, this is suitable for implicit feature selection. We have also experimented with 5-fold CV, but the results do not improve significantly.

Bayesian error estimator. A Bayesian approach to error estimation was recently introduced in the context of discrete classifiers[21] and linear classifiers.[22] The Bayesian error estimator (BEE) estimates the classification error directly from the training set and has shown to improve both the accuracy and speed of the actual error estimate[21,22] compared to traditional counting-based approaches, such as CV. In our earlier papers, we have shown that BEE has improved the stability and speed of computation in the model selection context as well.[9,23] We will next briefly review the definition of BEE for a fixed linear two-class classifier specified by the parameters $\boldsymbol{\beta}$ and β_0.

The Bayesian error estimator for linear classification assumes that the samples from each class are independent and identically distributed Gaussian random variables. For the two classes, the parameters (mean and covariance) of the Gaussian model are denoted as $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}_1$ and the corresponding priors for the parameters are denoted as $p_0(\boldsymbol{\theta})$ and $p_1(\boldsymbol{\theta})$. Then, the posterior probability density functions (PDFs) of parameters for class $c \in \{0,1\}$ are given by the Bayes' rule:

$$p_c^*(\boldsymbol{\theta} \mid \mathbf{X}, \mathbf{y}) \propto p_c(\boldsymbol{\theta}) \prod_{i:y_i=c} p_c(\mathbf{x}_i \mid \boldsymbol{\theta}), \quad (5)$$

where $p_c(\mathbf{x}_i \mid \boldsymbol{\theta})$ is the Gaussian class conditional density of $c \in \{0,1\}$.

The Bayesian error estimator (BEE) is defined as the minimum mean squared estimator by minimizing the expectation between the error estimate and the true error. This quantity is composed of class-specific conditional expected errors balanced by the priors $p(c)$ for the two classes $c \in \{0,1\}$[22]:

$$\mathrm{BEE} \triangleq E[\varepsilon \mid \mathbf{X}, \mathbf{y}] = p(0)E[\varepsilon_0 \mid \mathbf{X}, \mathbf{y}] + p(1)E[\varepsilon_1 \mid \mathbf{X}, \mathbf{y}], \quad (6)$$

with the expected classification error of samples from class c given by

$$E[\varepsilon_c \mid \mathbf{X}, \mathbf{y}] = \int \varepsilon_c(\boldsymbol{\theta}) p_c(\boldsymbol{\theta} \mid \mathbf{X}, \mathbf{y})\, d\boldsymbol{\theta}, \quad (7)$$

where $\varepsilon_c(\boldsymbol{\theta})$ denotes the true classification error.

The integral of Equation (7) can be evaluated by assuming an inverse Wishart prior for the class conditional density:

$$p_c(\boldsymbol{\theta}) \propto \det(\boldsymbol{\Sigma}_c)^{-(\kappa+P+1)/2} \exp\left(\frac{1}{2}\mathrm{trace}\left(\mathbf{S}\boldsymbol{\Sigma}_c^{-1}\right)\right) \times \det(\boldsymbol{\Sigma}_c)^{-(1/2)} \exp\left(-(\upsilon/2)(\boldsymbol{\mu}_c - \mathbf{m})^T \boldsymbol{\Sigma}_c^{-1}(\boldsymbol{\mu}_c - \mathbf{m})\right), \quad (8)$$

where $\upsilon \in \mathbb{R}, \kappa \in \mathbb{R}, \mathbf{S} \in \mathbb{R}^{P \times P}$, and $\mathbf{m} \in \mathbb{R}$ are the hyperparameters of the Bayesian model and $\boldsymbol{\theta}_c = (\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$ is the parameter of the Gaussian distribution. Different choices of the values of hyperparameters lead to different error estimators, but we will concentrate on a specific choice shown to be successful in earlier works[9,22]: $\kappa = P + 2$, $\upsilon = 0.5$, $\mathbf{S} = \mathbf{I}$, and $\mathbf{m} = \mathbf{0}$. For the resulting simplified closed-form solution, refer to Ref.[9] Matlab and Python implementations of BEE are available for download (https://sites.google.com/site/bayesianerrorestimate/).

Model selection. Model selection is a critical aspect in classifier design. Moreover, most modern classifiers are tuned by a set of hyperparameters, whose selection has a substantial effect on the resulting accuracy as well. Thus, the selection of an appropriate model family and the associated hyperparameters requires an accurate measure for comparing the accuracies of the model candidates. In our work, we are primarily interested in the selection of the regularization parameter λ of an LR classifier. However, it is to be noted that the methodology applies to any linear classifier.

The prediction accuracy and selection of the best model can be quantified by error estimators. CV estimator is often used to select the best value of the model selection parameter λ along a regularization path. As an example, error curves for different values of λ are illustrated in Figure 1. For this purpose, we used the flow cytometry training data of 49 features and 179 observations. For an individual tube, each feature represents the average of the biomarker intensities. The error curves are estimated for different values of λ ranging from 10^{-9} to 10^{0}.

In the example of Figure 1 (left panel), a 5-fold CV procedure is repeated 100 times and each step includes five training iterations on partial data. The error curves obtained for 100 iterations of 5-fold CV illustrate the significant deviation

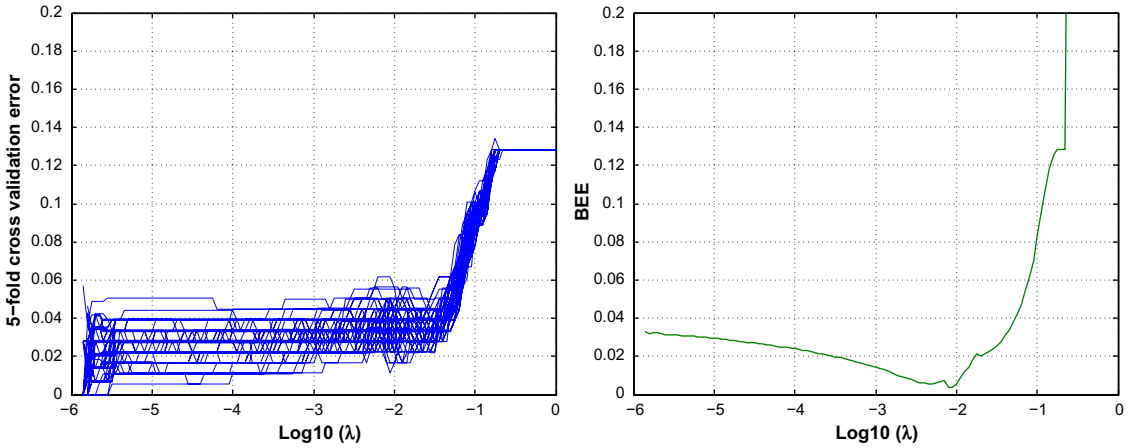

Figure 1. Left: Examples of regularization path error curves of 5-fold CV for our flow cytometry data with healthy and AML positive classes. Right: The corresponding BEE curve.

of the regularization paths from one iteration to another. The deviation is due to the randomness in splitting the training data into folds, which results in an individual error estimate for each split. Moreover, for a very small number of samples, such as 5 or 10, the split to validation and training sets for the K-fold CV estimator may not be appropriate. In fact, in this experiment, the K-fold CV approach fails to estimate the errors for smaller λ, as the number of samples split by CV is insufficient. On the other hand, Figure 1 (right panel) illustrates the error estimate of BEE, which is a single deterministic error curve. It is to be noted that the curve recognizes model overfitting (error estimate starts to increase for small regularization terms λ), although the error is estimated directly from the training set. No splitting or iterative resampling is required, which in turn accelerates the computation.

Experimental Results

In the following section, we present the experimental results. First, we demonstrate different model selection criteria to estimate the significant features. Then, we assess the performances of those methods in the AML classification case. Finally, we present the results for the second, small sample case.

Comparison of model selection criteria. Typical approach for the selection of model parameter is CV.[13] In this paper, we also consider Bayesian error estimator (see Bayesian Error Estimator section) and BIC[24] as alternative approaches to estimate the regularized parameter. In order to study the behavior of different parameter selection criteria, we first train the LR classifier with the training data along the decreasing sequence of regularization path with log10 (λ)$\in$ {0,–0.05,–0.1, –0.15,…, –8.90, –8.95, –9.00}. Then, again the whole training data are used to estimate the error rate for each λ. Finally, for each estimator, we select the model with λ value that achieves the minimum error rate. As resampling in CV-10 introduces randomness, in this case, we iterate 200 times and the result is averaged. The deterministic nature in BEE and BIC will produce the same result on the training data at each iteration.

The results are summarized in Table 3. For all methods, minimum error rates, AUC, and the number of selected features are estimated from the whole training data. It is to be noted that the reported AUC is computed from the training to emphasize that all feature sets are enough to partition the feature space into two categories perfectly. The test error is reported later.

The results indicate that the number of features selected by BEE method is lower compared to those of CV and BIC. For the first feature vector with size 49, BEE selects only 14 features as significant, while for the second feature vector with

Table 3. Parameter selection by different estimators: average number of selected features, λ, AUC, and their standard deviations with training data.

METHOD	FEATURE TYPE	NUMBER OF SELECTED FEATURES	SELECTED LOG10 (λ)	AUC (TRAINING)
CV-10	Mean	19.72 ± 2.41	–2.95 ± 2.30	0.9997 ± 0.0017
CV-10	Mean and std	23.91 ± 0.80	–4.14 ± 3.00	1 ± 0.0000
BEE	Mean	15 ± 0.00	–2.05 ± 0.00	0.9989 ± 0.0000
BEE	Mean and std	13 ± 0.00	–1.80 ± 0.00	0.9992 ± 0.0000
BIC	Mean	20 ± 0.00	–5.85 ± 0.00	1 ± 0.0000
BIC	Mean and std	24 ± 0.00	–5.70 ± 0.00	1 ± 0.0000

length 98, BEE selects only 12 features. Tables 4 and 5 list the selected features, ie, significant biomarkers along with the corresponding coefficient values. Due to the randomness in CV-10, we only present the results of one iteration as an illustration: there is a significant variation of the selected features depending on the chosen CV split. However, it is to be noted that the coefficients of BIC and BEE are not specific to this particular iteration, as they do not include the random split.

Performance assessment of the model selection criteria. The performances of the model selection methods are studied in the following section. The classification error is considered as the performance criterion, and both false positives (healthy control classified as AML) and false negatives (AML classified as healthy control) are counted with equal weight. The performance of the Bayesian error estimator is benchmarked against those of CV-10 and BIC for a different number of sample sizes. For this purpose, a randomly selected proportion of 10%, 15%, 20%–90%, and 95% is selected for training the classifier, while the remaining data are used for performance assessment. For each training sample, the experiment is executed 200 times by generating a new training set each time.

Classification errors. The error curves for different sample sizes are shown in Figure 2. The procedure is repeated 200 times for each training sample size, and the average of classification error is computed for each model selection criterion. With a very small number of training samples, such as 10% or 15% of the dataset, BEE provides improved accuracy over CV-10 and BIC (Fig. 2 left and right panels). For instance, with 10% training samples, the classification errors of the model selected by CV-10 are 7.56% (Fig. 2 left panel) and 7.41% (Fig. 2 right panel) higher than those of BEE. In case of BIC, the classification error is 7.81% higher than that of BEE (Fig. 2 left panel). As the number of training samples increases, for example, above 60%, the performance of BIC exceeds than that of BEE (Fig. 2 left panel). However, the performance of BIC is similar to that of BEE when more features are involved in the experiment (Fig. 2 right panel).

Table 4. The nonzero coefficients of features with mean.

TUBE	FEATURE	10-FOLD CV	BEE	BIC
	Constant	−13.38	−3.50	−13.54
Tube 1	FSLin	0.75	**0**	0.78
Tube 1	SSLog	−5.92	−0.73	−6.01
Tube 1	FL1:IgG1-FITC	−0.46	−0.19	−0.46
Tube 1	FL4:IgG1-PC5	−2.08	**0**	−2.13
Tube 1	FL5:IgG1-PC7	−3.07	−0.19	−3.14
Tube 2	FSLin	0.001	**0**	**0**
Tube 2	FL5:CD20-PC7	2.76	**0**	2.78
Tube 3	SSLog	**0**	−0.77	**0**
Tube 3	FL4:CD8-PC5	−1.94	−0.10	−1.97
Tube 4	FSLin	0.97	**0**	0.97
Tube 4	FL1:CD15-FITC	−4.77	**0**	−4.82
Tube 4	FL2:CD13-PE	3.44	0.21	3.48
Tube 4	FL4:CD16-PC5	**0**	−0.09	**0**
Tube 4	FL5:CD56-PC7	3.29	0.75	3.35
Tube 5	FSLin	2.35	**0**	2.40
Tube 5	FL2:CD11c-PE	−0.15	**0**	−0.16
Tube 5	FL3:CD45-ECD	−1.84	−0.02	−1.84
Tube 5	FL4:CD64-PC5	1.66	**0**	1.69
Tube 5	FL5:CD33-PC7	0.75	0.60	0.76
Tube 6	FL2:CD117-PE	4.82	0.89	4.88
Tube 6	FL4:CD34-PC5	6.88	0.72	6.99
Tube 6	FL5:CD38-PC7	**0**	0.41	**0**
Tube 7	FL1:CD5-FITC	−4.68	−0.19	−4.74

Note: The size of the feature sets is 49, of which the CV, BEE, and BIC select 20, 14, and 19 features, respectively.

Table 5. The nonzero coefficients of features with mean and standard deviation.

TUBE	FEATURE		10-FOLD CV	BEE	BIC
	Constant		−13.47	−3.27	−13.47
Tube 1	FSLin	Mean	0.027	**0**	0.027
Tube 1	SSLog	Mean	−4.49	−0.47	−4.49
Tube 1	FL1:IgG1-FITC	Std	−3.80	−0.22	−3.80
Tube 1	FL5:IgG1-PC7	Std	−1.60	−0.16	−1.60
Tube 2	FL5:CD20-PC7	Mean	0.50	**0**	0.50
Tube 3	SSLog	Mean	**0**	−0.29	**0**
Tube 3	FL5:CD2-PC7	Mean	−0.48	**0**	−0.48
Tube 3	FL5:CD2-PC7	Std	−1.21	**0**	−1.21
Tube 4	FSLin	Mean	0.05	**0**	0.05
Tube 4	FL1:CD15-FITC	Mean	−0.14	**0**	−0.14
Tube 4	FL2:CD13-PE	Mean	2.50	**0**	2.50
Tube 4	FL4:CD16-PC5	Mean	−1.32	**0**	−1.32
Tube 4	FL4:CD16-PC5	Std	**0**	−0.39	**0**
Tube 4	FL5:CD56-PC7	Std	6.07	0.45	6.07
Tube 5	FL1:CD14-FITC	Std	−0.004	**0**	−0.004
Tube 5	FL3:CD45-ECD	Mean	−2.51	**0**	−2.51
Tube 5	FL5:CD33-PC7	Mean	1.47	0.32	1.47
Tube 5	FL5:CD33-PC7	Std	−0.70	**0**	−0.70
Tube 6	SSLog	Std	**0**	−0.23	**0**
Tube 6	FL2:CD117-PE	Mean	3.19	0.46	3.19
Tube 6	FL2:CD117-PE	Std	2.19	**0**	2.19
Tube 6	FL4:CD34-PC5	Std	1.88	0.75	1.88
Tube 6	FL5:CD38-PC7	Mean	1.31	0.44	1.31
Tube 7	FSLin	Mean	1.39	**0**	1.39
Tube 7	FSLin	Std	**0**	−0.16	**0**
Tube 7	FL1:CD5-FITC	Std	−1.16	**0**	−1.16
Tube 7	FL5:CD10-PC7	Std	0.04	**0**	0.04

Note: The size of the feature sets is 98, of which the CV, BEE, and BIC select 23, 12, and 23 features, respectively.

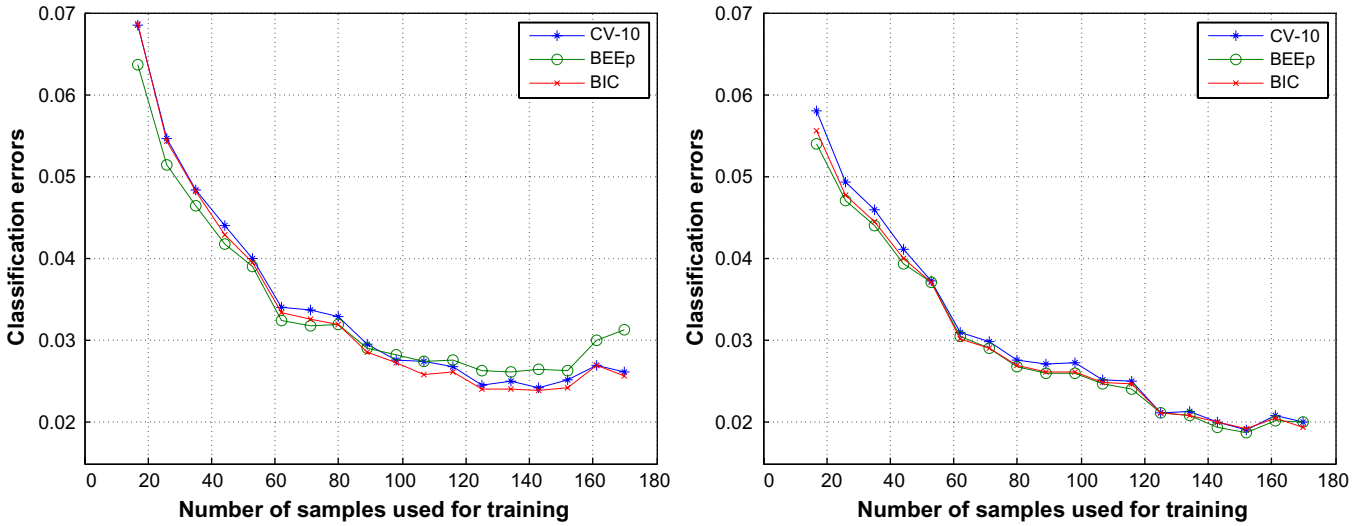

Figure 2. The average classification error curves for CV-10, BEE with proper prior (BEEp), and BIC.
Notes: Left: feature vector of mean values of measurements. Right: feature vector of mean and standard deviation of measurements.

Area under the ROC curve. In this section, we evaluate the performance in terms of AUC. Figure 3 illustrates the average of AUC for different training sample sizes. Here, the BEE method achieves improvement over the other methods. With small training samples, for instance, 10%, the average AUC of BEE is 1.11% (Fig. 3 left panel) and 1.30% (Fig. 3 right panel) higher than that of CV-10. As the number of training samples increases, CV-10 and BIC also converge toward the results of BEE; however, the BEE selected model consistently results in the highest AUC score. With the larger feature vector that includes the measurements of mean and standard deviation, the average AUC curves of BEE and BIC follow the similar pattern (Fig. 3 right panel).

Number of selected features. We further assess the performances of the estimators using feature selection criteria. At each iteration, we determine the total number of selected features that have nonzero values for a different number of training samples. Then, we compute the average and the variability (ie, standard deviation) of the selected features for different training samples. The results are illustrated in Figure 4. For BEE, the average number of selected features is lower in amount compared to those of CV and BIC (Fig. 4 top panel). For instance, with 95% training samples, BEE requires 36.49% and 33.89% less features than CV and BIC, respectively, for model prediction (Fig. 4 top-right panel). Moreover, the variability in selected features using BEE is also comparable (Fig. 4 bottom panel). The CV-10 has the worst performance. Although BIC shows that the deviation in feature selection at different iterations is smaller, the

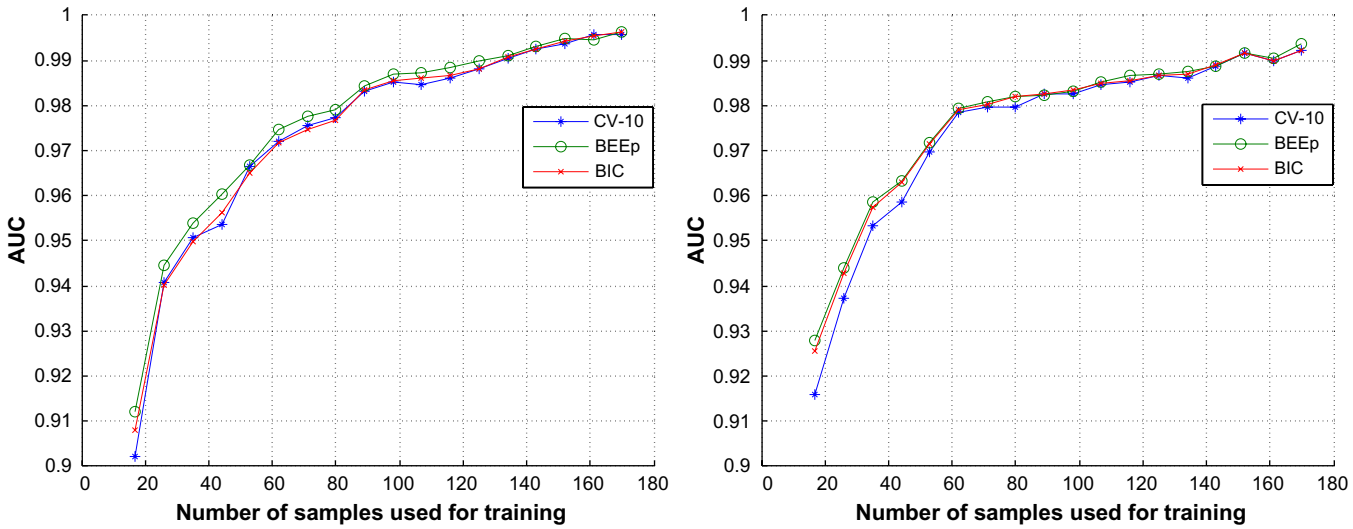

Figure 3. The average AUCs for CV-10, BEE with proper prior (BEEp), and BIC.
Notes: Left: feature vector of mean values of measurements. Right: feature vector of mean and standard deviation of measurements.

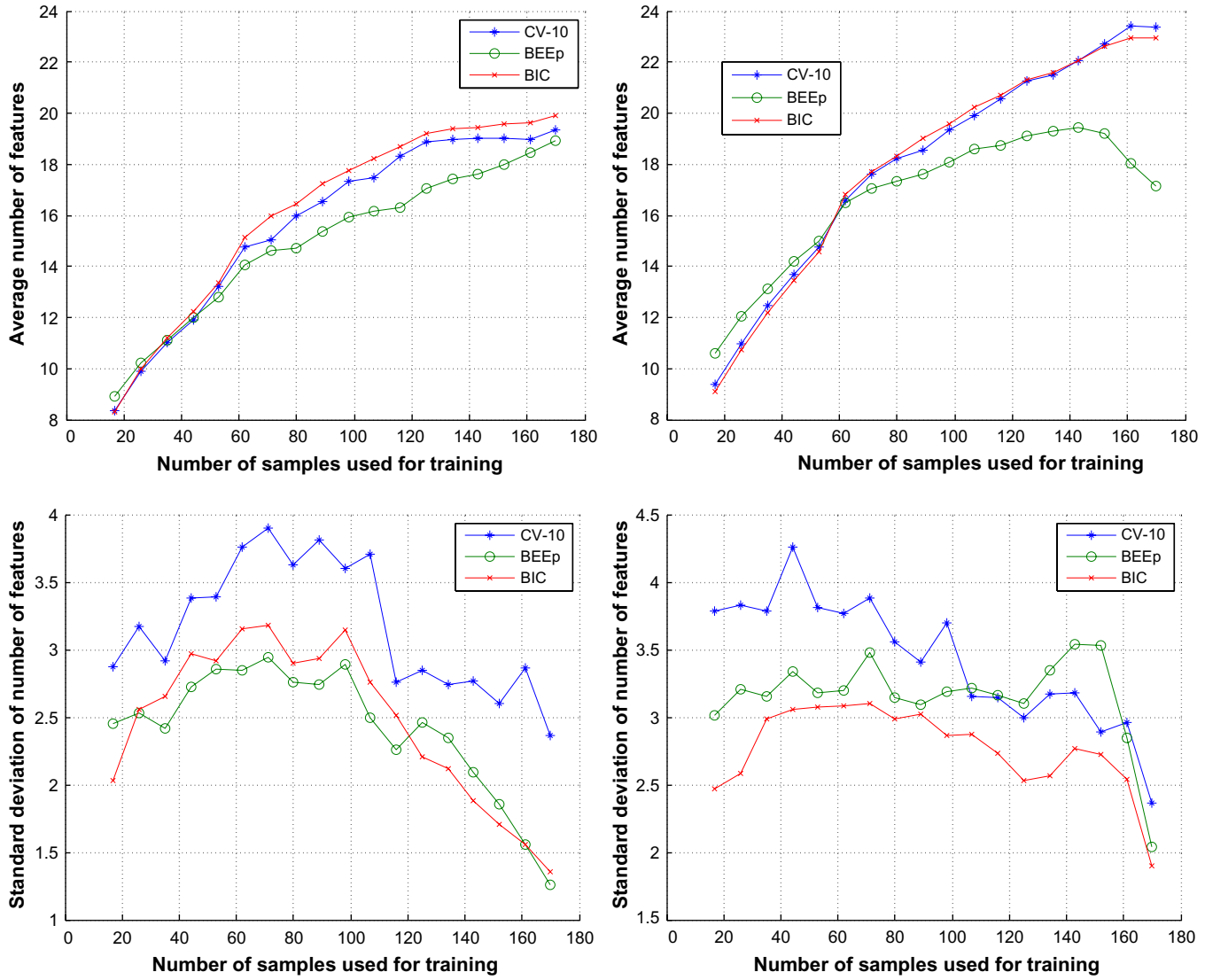

Figure 4. Comparisons of the number of selected features for CV-10, BEE with proper prior (BEEp), and BIC.
Notes: Left: feature vector of mean values of measurements. Right: feature vector of mean and standard deviation of measurements. Top: average number of selected features. Bottom: standard deviation of number of the selected features.

average number of selected features is higher than that of others (Fig. 4 top panel).

Similarity of the selected feature sets. Another performance measurement is the stability of selecting the same feature at different iterations. For this purpose, Sørensen–Dice coefficient[25] is used, which measures the degree of similarity between selected features of two different iterations. The ranges can vary from 0 to 1. The values closest to 1 indicate a high-degree of similarity.

For different training samples, we first determine which features are selected at each iteration. As the model selection process is repeated 200 times, we estimate the similarity as the mean dice coefficient for each of the 200!/(2! × (200 − 2)!) = 19,900 possible pairs of selected feature sets. The results are shown in Figure 5. In terms of stability, the performance of BEE is substantially better than those of the other methods, as the selected feature sets are most similar with that criterion – a significant issue when trying to understand the biological mechanisms behind the data. For example, with 60% training samples, the dice coefficient of BEE is 6.03% higher than that of CV (Fig. 5 left panel). On the other hand, with 90% training samples, the dice coefficient of BEE is 5.81% higher than that of CV and 4.90% higher than that of BIC (Fig. 5 right panel). Indeed, the dice coefficient is unfavorable for CV with small training samples: The dice coefficient is lowest among the alternatives, indicating that the selected feature sets with the CV criterion have high variability.

Small sample case with a cancer cell line. For further confidence on the presented method, we analyze data from a cancer cell line in a small sample setting. As described previously, we considered the classification accuracy, AUC measure, and the number of selected variables both with and without standard deviation features (Figs. 6–8). In this case,

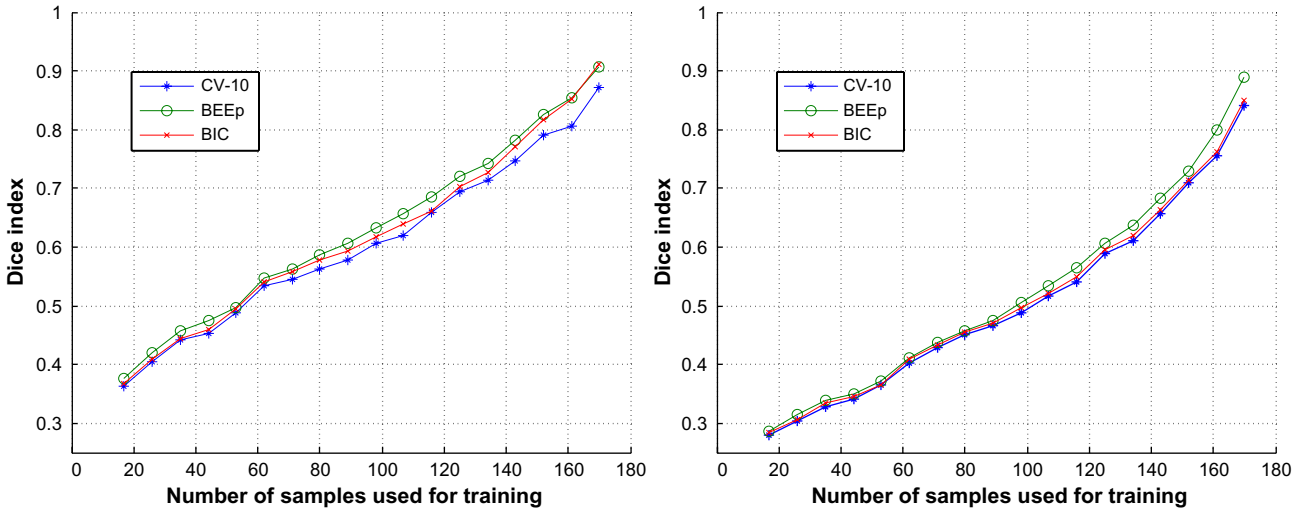

Figure 5. Comparison of the stability of selecting features for CV-10, BEEp, and BIC.
Notes: Left: feature vector of mean values of measurements. Right: feature vector of mean and standard deviation of measurements.

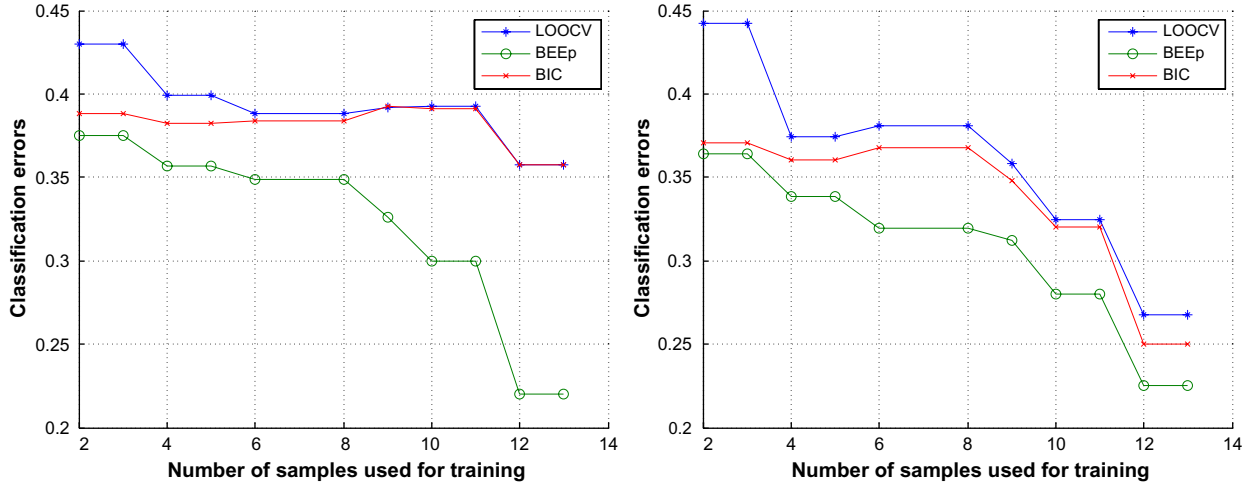

Figure 6. The average classification error curves for LOOCV, BEEp, and BIC.
Notes: Left: feature vector of mean values of measurements. Right: feature vector of mean and standard deviation of measurements.

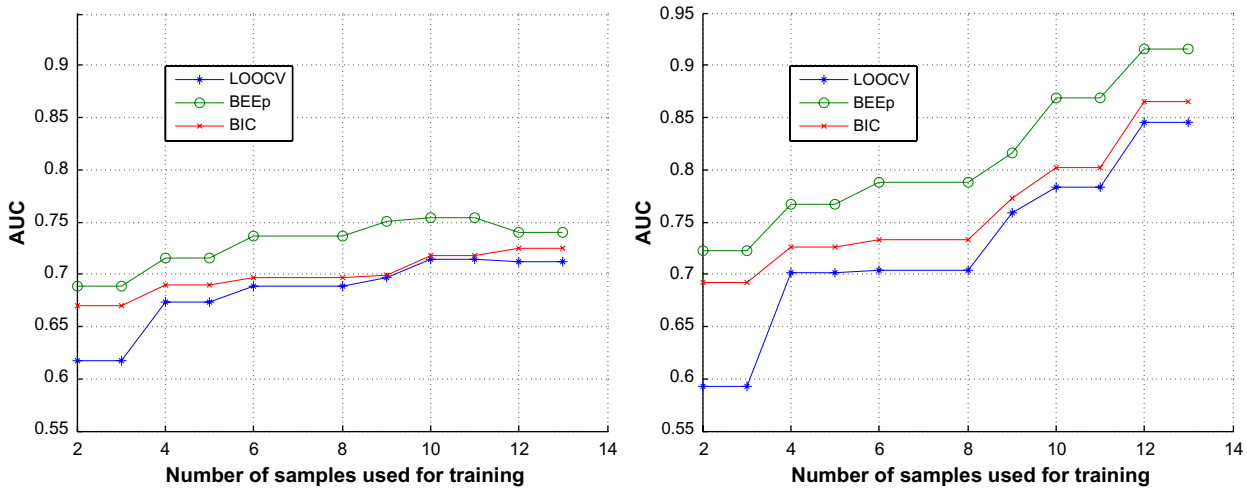

Figure 7. The average AUC curves for LOOCV, BEEp, and BIC.
Notes: Left: feature vector of mean values of measurements. Right: feature vector of mean and standard deviation of measurements.

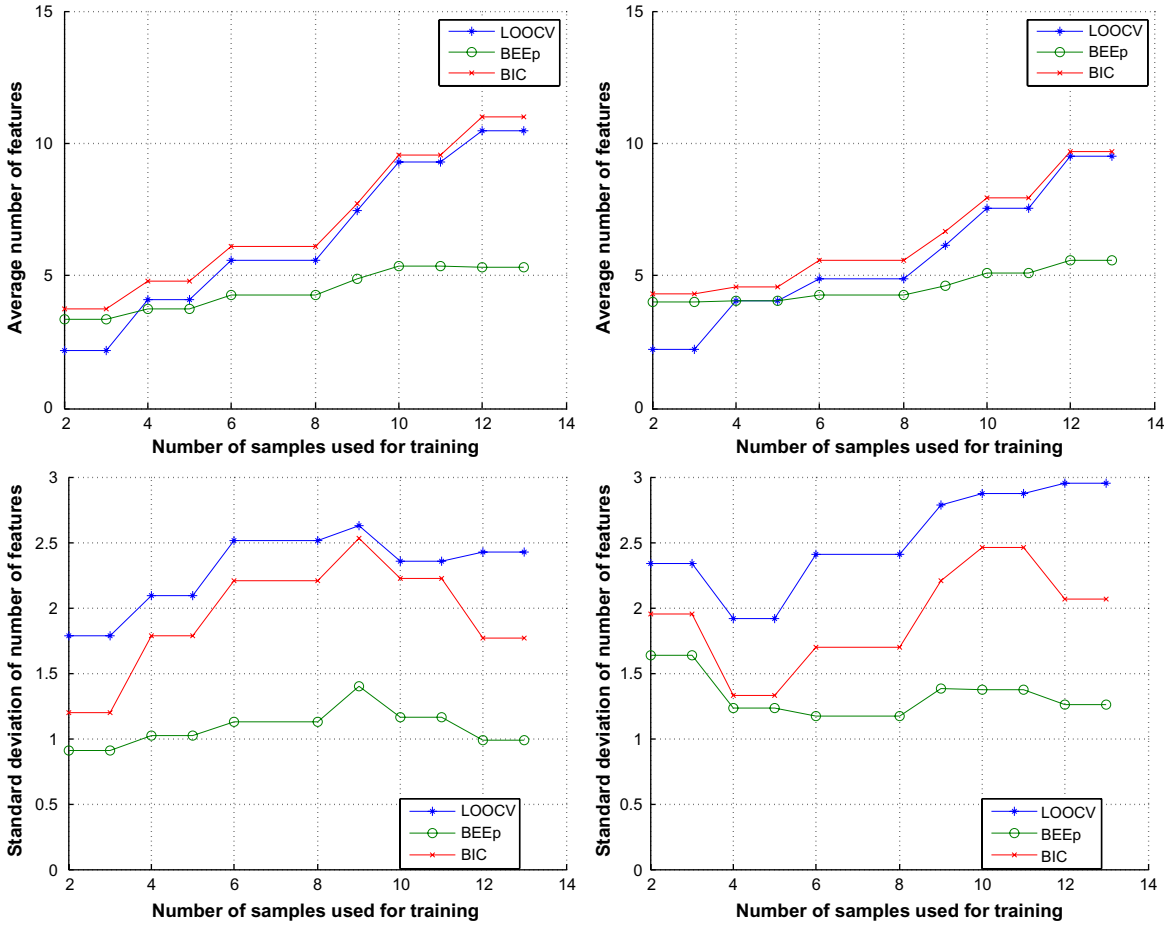

Figure 8. Comparisons of the number of selected features for LOOCV, BEEp, and BIC.
Notes: Left: feature vector of mean values of measurements. Right: feature vector of mean and standard deviation of measurements. Top: average number of selected features. Bottom: standard deviation of number of the selected features.

we split the dataset into 15%, 22%, 29%–78%, and 85% for training the classifier, while the remaining data are used for performance assessment.

As the sample size is minimal, we applied leave-one-out cross-validation (LOOCV) instead of the CV-10. The results by BEE are in general more accurate than those by BIC and LOOCV and also obtained with fewer parameters in the model. The case study with prostate cancer cell line shows that the presented method is able to efficiently classify between the treatments in a very small sample setting.

Conclusions

In this paper, we have studied the effect of feature selection classification of flow cytometry data. In particular, we considered using simplistic features instead of more complicated feature extraction pipelines widely seen in the literature. As a result, we were able to simplify and reduce the number of features without compromising the prediction accuracy. In addition to this, we considered the problem of feature selection in a small sample size setting. Such cases are not uncommon in biology, yet they have not received a lot of attention in scientific literature. In particular, the stability of the feature selection process varies a lot depending on the error measure used for model selection.

The Experimental Results section considered three different error metrics and compared them in terms of classification accuracy (measured by both classification error and AUC) and feature selection stability (measured by the number of features and the dice index between feature sets). As a result, the recently presented Bayesian error estimator (BEE) has a superior stability and an improved accuracy over the traditional counting-based approach, such as CV. The experiments show that BEE selects better classification models than the model selected by CV. In particular, the BEE is more effective compared to its alternatives when the number of training samples is relatively small.

Although in this study we concentrate only on flow cytometry data, we expect that the benefits of our approach – capability to deal with small sample settings and with high-dimensional data through reducing the number of features used for analysis – would make the method a good candidate also for other types of biomedical data. The effectiveness in model selection other types of data has already been demonstrated in Ref.[9]

Acknowledgment

We acknowledge the CSC – IT Center for Science Ltd., Finland, for the allocation of computational resources.

Author Contributions

SSH, PR, HH implemented the software and designed the experiments. LL acquired the data in the smaller dataset case. SSH wrote the manuscript. PR, HH, LL contributed to the writing and revising of the manuscript. All the authors reviewed and approved the final manuscript.

REFERENCES

1. Jennings CD, Foon KA. Recent advances in flow cytometry: application to the diagnosis of hematologic malignancy. *Blood*. 1997;90(8):2863–92.
2. Rogers WT, Moser AR, Holyst HA, et al. Cytometric fingerprinting: quantitative characterization of multivariate distributions. *Cytometry A*. 2008;73(5):430–41.
3. Bendall SC, Simonds EF, Qiu P, et al. Single-cell mass cytometry of differential immune and drug responses across a human hematopoietic continuum. *Science*. 2011;332(6030):687–96.
4. Aghaeepour N, Finak G, Consortium TF, et al. Critical assessment of automated flow cytometry data analysis techniques. *Nat Methods*. 2013;10:228–37.
5. Biehl M, Bunte K, Schneider P. Analysis of flow cytometry data by matrix relevance learning vector quantization. *PLoS One*. 2013;8(3):59401.
6. Vilar JM. Entropy of leukemia on multidimensional morphological and molecular landscapes. *Phys Rev X*. 2014;4(2):021038.
7. Manninen T, Huttunen H, Ruusuvuori P, Nykter M. Leukemia prediction using sparse logistic regression. *PLoS One*. 2013;8(8):72932.
8. Dougherty ER, Sima C, Hua J, Hanczar B, Braga-Neto UM. Performance of error estimators for classification. *Curr Bioinform*. 2010;5:53–67.
9. Huttunen H, Tohka J. Model selection for linear classifiers using Bayesian error estimation. *Pattern Recognit*. 2015;48:3739–48.
10. Kaukoniemi KM, Rauhala HE, Scaravilli M, et al. Epigenetically altered mir-193b targets cyclin d1 in prostate cancer. *Cancer Med*. 2015;4(9):1417–25.
11. Hastie T. The Elements of Statistical Learning Data Mining, Inference, and Prediction Vol. 2nd ed. New York: Springer; 2009:745.
12. Ng AY. Feature selection, L1 vs. L2 regularization, and rotational invariance. *Proceedings of the Twenty-First International Conference on Machine Learning (ICML)*. 2004:78–85.
13. Tibshirani R. Regression shrinkage and selection via the lasso. *J R Stat Soc Series B Stat Methodol*. 1996;58(1):267–88.
14. Yuan M, Lin Y. Model selection and estimation in regression with grouped variables. *J R Stat Soc Series B Stat Methodol*. 2006;68(1):49–67.
15. Meier L, Van De Geer S, Bühlmann P. The group lasso for logistic regression. *J R Stat Soc Series B Stat Methodol*. 2008;70(1):53–71.
16. Candes E, Tao T. The Dantzig selector: statistical estimation when p is much larger than n. *Ann Stat*. 2007;35(6):2313–51.
17. Zou H, Hastie T. Regularization and variable selection via the elastic net. *J R Stat Soc Series B Stat Methodol*. 2005;67(2):301–20.
18. Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*. 2008;9(3):432–41.
19. Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. *J Stat Softw*. 2010;33(1):1–22.
20. Efron B, Gong G. A leisurely look at the bootstrap, the jackknife, and cross-validation. *Am Stat*. 1983;37(1):36–48.
21. Dalton L, Dougherty ER. Bayesian minimum mean-square error estimation for classification error – part I: definition and the Bayesian MMSE error estimator for discrete classification. *IEEE Trans Signal Process*. 2011;59(1):115–29.
22. Dalton L, Dougherty ER. Bayesian minimum mean-square error estimation for classification error – part II: the Bayesian MMSE error estimator for linear classification of Gaussian distributions. *IEEE Trans Signal Process*. 2011;59(1):130–44.
23. Huttunen H, Manninen T, Tohka J. Bayesian error estimation and model selection in sparse logistic regression. *2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*. Southampton: IEEE; 2013:1–6.
24. Chen J, Chen Z. Extended Bayesian information criteria for model selection with large model spaces. *Biometrika*. 2008;95(3):759–71.
25. Dice LR. Measures of the amount of ecologic association between species. *Ecology*. 1945;26(3):297–302.