Robust Linear Bayes Classifier for Microarray Gene Expression Data Analysis

Md. Mushfiqur Rahman1
Md. Alam Hossin2
Abu Shaleh Mahmud3
Md. Nurul Haque Mollah4

Dept. of Bioinformatics, College of Life Sciences
Zhejiang University, Zijingang Campus,
Hangzhou 310058, China.
mahirat@gmail.com

Abstract— Classifying tissue samples or genes into different groups is a central task of bioinformatics research. Different classification (supervised and unsupervised) methods are available. However, these existing methods produce low accuracy results if the data set is high-dimensional or contaminated by outliers. Microarray gene expression data contains high level of noise. These dimensionality issues and the noise in the dataset, substantially influence classification accuracy. To noise resistance and accurately classify of tissue samples or genes, a suitable method is necessary to achieve good performance. In this paper, we proposed highly robust Linear Bayes classification rule for supervised classification in gene expression studies.

Index Terms— Bayes classifier, Gaussian distribution, minimum β-divergence method, Robustness and Gene-expression data.

I. INTRODUCTION

Tumor/cancer classification based on gene expression data or gene classification based on gene expression on normal cell and cancer cell playing the significant role in cancer research [9–11,20,23,24,27]. It is a technique that successfully used to explore the relationship among the genes and the tissue samples [17]. Gene expression data analysis using classifications are considered Golub et al., Alon et al, Stollem et al., Ben-Dor et al., Veen et al [15, 16, 20, 25,27] and other researchers to elucidate unknown gene function, phenotypic outcomes, disease diagnosis such as tumor or cancer types, analysis of prognosis and treatment outcomes, clinical drug analysis and so on. In the literature, there are several classifier such as Bayes classifier, support vector machines, Fisher linear discriminant analysis, k-Nearest Neighbor classifier, classification tree, Bagging, boosting, among others has been applied in gene-expression data analysis studies [5,11,16,18,19,22].

II. METHODS

Linear Bayes Classifier

Suppose we have a training gene expression samples

\[
\begin{align*}
[x_1(x), & \ldots, x_n(x)],
\end{align*}
\]

obtained from a gene expression density function for \(k = 1 \ldots m\). Our goal is to classify a new gene expression sample \(x = (x_1, x_2, \ldots, x_n)\) into one of \(m\) gene expression population group \(\Pi_k\) based on the training sample. The pdf of the new gene expression sample \(x\) can be defined as the mixture distributions as follows-

\[
f(x; \theta) = \sum_{k=1}^{m} q_k f_k(x; \theta).
\]

(1)

Where, \(q_k\) is the mixing proportion of \(x\) belongs to population \(\Pi_k\) such that \(\sum_{k=1}^{m} q_k = 1\). Then the posterior pdf \(x\) belongs to \(\Pi_k\) is given by-

\[
g_k(x; \theta; \Pi_k) = \frac{q_k f_k(x; \theta)}{\sum_{k=1}^{m} q_k f_k(x; \theta)}
\]

(2)

For the Bayes classifier, the space of all observations is divided into \(m\) mutually exclusive and exhaustive regions\(\Delta_k\) \((k = 1 \ldots m)\). Then define region \(\Delta_k\) for classifying new gene expression sample \(x\) to one of \(m\) populations as follows-

\[
\Delta_k: q_k f_k(x; \theta) > q_j f_j(x; \theta), j = 1, 2, \ldots, m, \quad (j \neq k)
\]

(3)

This minimizes the expected cost of misclassification [1]. For simplicity we assumed that cost and mixing proportion is equal for each population and the gene expression distributions are multivariate Gaussian with identical covariance matrices, then the Bayes classifier is linear [14] and is given by-

\[
y^*(x) = x^T \beta - c
\]

(4)

And, \(x^T \beta\) and the covariance matrix \(\Sigma\) are strongly influenced by outliers. It is obvious that the Linear Bayes classifier dependent on the feature vector, mean vectors and covariance matrices those are estimated by the non-robust maximum likelihood estimates based on training sample. Therefore, traditional Linear Bayes mechanism may produce misleading results in presence of outliers in the training or test or in both gene expression datasets. To improve the results, Mollah and Matirah, 2014 [13] proposed a robust Bayes classification rule using minimum β-divergence method [4,8].

Robust Linear Bayes classification Rule:

The minimum β-divergence estimators \(\hat{\mu}_k^\beta\) and \(\hat{\Sigma}_k^\beta\) for the mean vector \(\mu_k\) and the covariance matrix \(\Sigma_k\) respectively are obtained iteratively as follows-

\[
\mu_{k+1} = \frac{\sum_{l=1}^{m} \phi_{1,\beta,\mu_l} (x_l - \mu_l^\beta) \mu_l}{\sum_{l=1}^{m} \phi_{1,\beta,\mu_l} (x_l - \mu_l^\beta)}
\]

(5)

and,

\[
\Sigma_{k+1} = \frac{\sum_{l=1}^{m} \phi_{1,\beta,\mu_l} (x_l - \mu_l^\beta) (x_l - \mu_l^\beta)^T \Sigma_l}{\sum_{l=1}^{m} \phi_{1,\beta,\mu_l} (x_l - \mu_l^\beta) ^2}
\]

(6)

where, \(\phi_{1,\beta,\mu} (x_l - \mu^\beta)\) is the mixing proportion of \(\mu^\beta\) and \(\Sigma\) respectively. The standard estimates of the mean \(\hat{\mu}^\beta\) and the covariance matrix \(\hat{\Sigma}^\beta\) is used during iteration. If \(\beta\) tends to 0, then Moore-Penrose generalized inverse of \(\hat{\Sigma}^\beta\) is used during iteration. However, the \(\beta\) weight function plays the significant role for robustification of Linear Bayes classifier as discussed below.

Step-1: First, calculate β-weight for the new gene expression feature vector \(x\) using the β-weight function: \(W_{\beta, x}(x) = \phi_{1,\beta,\mu_l} (x_l - \mu_l^\beta)\).

\[
W_{\beta, x}(x) = \exp \left(-\frac{\beta}{2} (x - \mu^\beta)^T \Sigma^{-\beta} (x - \mu^\beta) \right)
\]

(7)

Then and construct a criteria to test whether the feature vector is contaminated or not as follows-

\[
W_{\beta, x}(x) = \begin{cases}
> \epsilon, & \text{if } x \text{ is not contaminated} \\
< \epsilon, & \text{if } x \text{ is contaminated}
\end{cases}
\]

(8)

where, \(\epsilon = (1 - \eta) \min_{\pi l, \pi j, \pi k} W_{\pi l}(x_l) + \max_{\pi l, \pi j, \pi k} W_{\pi j}(x_j)\), with heuristically \(\eta = 0.15\), where, \(D\) is the gene expression dataset including the new test sample \(x\). It was also used in [8] for choosing the threshold value. If the test sample is not contaminated by outliers, we compute any one of classifications regions defined in (5) using the minimum β-divergence estimators \(\hat{\mu}_k^\beta, \hat{\Sigma}_k^\beta\) of \(\{\mu_k, \Sigma\}\) where, \(\tilde{y}_k\) is computed using equation (6). If test sample \(x\) is contaminated by outliers, we classify it by replacing its contaminated components by their mean components as discussed in the following step.

Step-2: For contaminated test gene expression sample \(x\) we calculate the absolute difference between the contaminated \(x\) and each training gene expression mean vector as \(d_{k\pi} = \max_{\pi l, \pi j, \pi k} |x_l - \mu_l^\beta|\).

\[
d_{k\pi} = \begin{cases}
> \epsilon, & \text{if } x \text{ is not contaminated} \\
= \infty, & \text{if } x \text{ is contaminated}
\end{cases}
\]

(9)

where, \(\epsilon = (1 - \eta) \min_{\pi l, \pi j, \pi k} W_{\pi l}(x_l) + \max_{\pi l, \pi j, \pi k} W_{\pi j}(x_j)\), with heuristically \(\eta = 0.15\), where, \(D\) is the gene expression dataset including the new test sample \(x\). It was also used in [8] for choosing the threshold value. If the test sample is not contaminated by outliers, we compute any one of classifications regions defined in (5) using the minimum β-divergence estimators \(\hat{\mu}_k^\beta, \hat{\Sigma}_k^\beta\) of \(\{\mu_k, \Sigma\}\) where, \(\tilde{y}_k\) is computed using equation (6). If test sample \(x\) is contaminated by outliers, we classify it by replacing its contaminated components by their mean components as discussed in the following step.

Step-2: For contaminated test gene expression sample \(x\) we calculate the absolute difference between the contaminated \(x\) and each training gene expression mean vector as \(d_{k\pi} = \max_{\pi l, \pi j, \pi k} |x_l - \mu_l^\beta|\).

\[
d_{k\pi} = \begin{cases}
> \epsilon, & \text{if } x \text{ is not contaminated} \\
= \infty, & \text{if } x \text{ is contaminated}
\end{cases}
\]

(9)
Simulated Gene Expression Data Set

In here we are generated microarray data using a model displayed in Fig. 1 and this model generates microarray gene expression datasets with two levels corresponding to two sets of differentially expressed genes. We randomly added Gaussian noise N (0, $\sigma^2$) with each expression of each gene. Also, [12] have used this model for generating simulated microarray data sets. In Fig.1, columns represent the gene expression positive and negative control of patients and rows represent the gene group. In this simulation study we have generated gene expression datasets with two levels corresponding to two sets of differentially expressed genes where, $d$ is the difference among the expression. Using this model (fig.1) we generated several training data sets and test gene sets with setting different values of the parameters $d$ and $\sigma^2$ with different number of samples ($n=100$).

and some or all components of the unclassified contaminated $x$ corresponding to $d_{ij_1}, d_{ij_2}, ..., d_{ij_p}$ are assumed to be corrupted by outliers. Then we update $x$ by replacing its corrupted components with the corresponding mean components from the mean vector $\bar{x}_{ji}$ of $k$-th population. Let $x^*$ be the updated vector of the contaminated data vector $x$. Then we use $x^*$ instead of $x$ to compute any one of classifications regions defined in (3) for classifying $x^*$ etc.

III. GENE EXPRESSION DATA ANALYSIS

Real Gene Expression Data Set

Head and neck cancer data (HNC): We have analyzed the publicly available microarray data in the study of head and neck cancer, where RNA was extracted from 22 paired samples of HNSCC and normal tissue from the same donors and hybridized to the Affymetrix U95A chip [6]. This data consist of 12625 cellular RNA transcripts contain tumor and normal tissues from 22 patients with histologically confirmed HNSCC.

Colon Data:

Gene expression data from the microarray experiments of colon tissue samples of [15] gives the expression levels of 2,000 genes for 62 samples of which 40 tumor tissues and 22 normal tissues. Although, originally 6,000 gene expression data were measured on Affymetrix human. Among these 6,000 genes 4,000 genes were removed considering the reliability of measured values in the measured expression levels . The measured expression values of 2,000 genes are publicly available at colonCA library in Bioconductor [30] and in http://microarray.princeton.edu/oncology/affydata/index.html.

Gene Selection: Dimensionality reduction of microarray gene expression data has been performed by many authors, as for example [16,18,20] among others. In our paper, we applied 7 gene selection methods in order to improved classification performance for real gene expression data analysis as Empirical Bayes (EB), significance analysis of microarrays (SAM) [29], t-statistic, linear models for microarray data (Limma) [13], GaGa, [28], Bridge [21] and \beta-Empirical Bayes (\beta-EB) [3]. These methods were applied in the datasets and firstly we found little agreement in gene lists produced by each of the methods. The t-test, Limma, and SAM detect DE genes based on p-values while, the EB approach, GaGa, Bridge and the \beta-EB approach detect DE genes based on posterior probabilities. To detect the DE genes in here we consider p-value< 0.05 and the posterior probability (>0.9). Table 1 and Table 2 shows the DE gene lists for two microarray dataset produced by the each of the methods. Also each method obtained same 239 and 24 DE genes for HNC and colon cancer datasets respectively. Then we ordering the DE genes according to its p-value and posterior probability and then selected top 20 genes for each datasets to evaluate the class prediction efficiency of each classification methods.

A complete workflow for real data analysis is shown in figure 2.In fig. 2 the most highly differential genes were selected from 2 gene expression datasets using 7 feature selection approaches. In each case genes were selected and 6 classifiers are performed and then misclassification error rate for each classifier was recorded using LOCV. For the performance of our proposed method, we assume that the original data have zero level of noise. Therefore, we added some noise to the original DE gene set in the same way described as before.

IV. RESULT AND DISCUSSION

Simulation results:

Exploring the performance of our proposed method we calculated misclassification rate (MR) for each of the comparative classification method. Therefore, we generated 1,000 simulated microarray datasets for computing MR and added 20% outliers for data
contamination. Basically, in this paper we compared our proposed classifier with the traditional Linear Bayes classifier, Support Vector Machine (SVM), \( k \)-Nearest Neighbor (KNN) and Boosting (Adaboost and Logitboost) method. Fig. 3 shows the box plot of the MR of each classifier for simulated (with and without outliers) dataset, generated with the same setting i.e., \( n = 100; d = 2; \sigma = 3; g = 60 \).

Table 1: Misclassification rate (MR) for Head and Neck cancer dataset of top 20 different sets of differentially expressed genes of the different classifier and our proposed classifier.

<table>
<thead>
<tr>
<th>Methods</th>
<th>DE genes</th>
<th>Linear Bayes</th>
<th>SVM</th>
<th>KNN</th>
<th>Adaboost</th>
<th>LogitBoost</th>
<th>t-Linear Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>EB</td>
<td>102</td>
<td>0.226</td>
<td>0.274</td>
<td>0.293</td>
<td>0.387</td>
<td>0.321</td>
<td>0.216</td>
</tr>
<tr>
<td>SAM</td>
<td>342</td>
<td>0.258</td>
<td>0.371</td>
<td>0.354</td>
<td>0.348</td>
<td>0.225</td>
<td>0.207</td>
</tr>
<tr>
<td>t-EB</td>
<td>611</td>
<td>0.209</td>
<td>0.174</td>
<td>0.219</td>
<td>0.370</td>
<td>0.348</td>
<td>0.207</td>
</tr>
<tr>
<td>LIM</td>
<td>343</td>
<td>0.177</td>
<td>0.374</td>
<td>0.292</td>
<td>0.328</td>
<td>0.127</td>
<td>0.207</td>
</tr>
<tr>
<td>GAGA</td>
<td>279</td>
<td>0.211</td>
<td>0.238</td>
<td>0.329</td>
<td>0.241</td>
<td>0.207</td>
<td>0.127</td>
</tr>
<tr>
<td>BRIDGE</td>
<td>275</td>
<td>0.216</td>
<td>0.273</td>
<td>0.340</td>
<td>0.293</td>
<td>0.252</td>
<td>0.127</td>
</tr>
<tr>
<td>t-EB</td>
<td>125</td>
<td>0.193</td>
<td>0.203</td>
<td>0.249</td>
<td>0.272</td>
<td>0.203</td>
<td>0.193</td>
</tr>
</tbody>
</table>

Table 3: Misclassification rate (MR) of the top 20 same differential expressed (DE) genes obtained in different differential expressed calculation method of different classifier including our proposed method.

<table>
<thead>
<tr>
<th>Data set</th>
<th>DE genes</th>
<th>Linear Bayes</th>
<th>SVM</th>
<th>KNN</th>
<th>Adaboost</th>
<th>LogitBoost</th>
<th>t-Linear Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>HNC</td>
<td>230</td>
<td>0.045</td>
<td>0.055</td>
<td>0.113</td>
<td>0.068</td>
<td>0.055</td>
<td>0.045</td>
</tr>
<tr>
<td>Colon</td>
<td>24</td>
<td>0.068</td>
<td>0.120</td>
<td>0.112</td>
<td>0.152</td>
<td>0.056</td>
<td>0.056</td>
</tr>
<tr>
<td>HNC</td>
<td>230</td>
<td>0.106</td>
<td>0.089</td>
<td>0.083</td>
<td>0.110</td>
<td>0.065</td>
<td>0.057</td>
</tr>
<tr>
<td>Colon</td>
<td>24</td>
<td>0.088</td>
<td>0.275</td>
<td>0.226</td>
<td>0.217</td>
<td>0.205</td>
<td>0.205</td>
</tr>
</tbody>
</table>

Table 4: To investigate the performance of robust Linear Bayes classifier 1,000 contaminated simulated gene expressions datasets average results (with increasing number of sample \( n \)) are shown below:

<table>
<thead>
<tr>
<th>Methods</th>
<th>DE genes</th>
<th>Linear Bayes</th>
<th>SVM</th>
<th>KNN</th>
<th>Adaboost</th>
<th>LogitBoost</th>
<th>t-Linear Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>EB</td>
<td>102</td>
<td>0.226</td>
<td>0.274</td>
<td>0.293</td>
<td>0.387</td>
<td>0.321</td>
<td>0.216</td>
</tr>
<tr>
<td>SAM</td>
<td>342</td>
<td>0.258</td>
<td>0.371</td>
<td>0.354</td>
<td>0.348</td>
<td>0.225</td>
<td>0.207</td>
</tr>
<tr>
<td>t-EB</td>
<td>611</td>
<td>0.209</td>
<td>0.174</td>
<td>0.219</td>
<td>0.370</td>
<td>0.348</td>
<td>0.207</td>
</tr>
<tr>
<td>LIM</td>
<td>343</td>
<td>0.177</td>
<td>0.374</td>
<td>0.292</td>
<td>0.328</td>
<td>0.127</td>
<td>0.207</td>
</tr>
<tr>
<td>GAGA</td>
<td>279</td>
<td>0.211</td>
<td>0.238</td>
<td>0.329</td>
<td>0.241</td>
<td>0.207</td>
<td>0.127</td>
</tr>
<tr>
<td>BRIDGE</td>
<td>275</td>
<td>0.216</td>
<td>0.273</td>
<td>0.340</td>
<td>0.293</td>
<td>0.252</td>
<td>0.127</td>
</tr>
<tr>
<td>t-EB</td>
<td>125</td>
<td>0.193</td>
<td>0.203</td>
<td>0.249</td>
<td>0.272</td>
<td>0.203</td>
<td>0.193</td>
</tr>
</tbody>
</table>

Table 5: To investigate the performance of robust Linear Bayes classifier 1,000 contaminated simulated gene expressions datasets average results (with increasing difference \( d \)) are shown below:

<table>
<thead>
<tr>
<th>Methods</th>
<th>DE genes</th>
<th>Linear Bayes</th>
<th>SVM</th>
<th>KNN</th>
<th>Adaboost</th>
<th>LogitBoost</th>
<th>t-Linear Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>EB</td>
<td>102</td>
<td>0.226</td>
<td>0.274</td>
<td>0.293</td>
<td>0.387</td>
<td>0.321</td>
<td>0.216</td>
</tr>
<tr>
<td>SAM</td>
<td>342</td>
<td>0.258</td>
<td>0.371</td>
<td>0.354</td>
<td>0.348</td>
<td>0.225</td>
<td>0.207</td>
</tr>
<tr>
<td>t-EB</td>
<td>611</td>
<td>0.209</td>
<td>0.174</td>
<td>0.219</td>
<td>0.370</td>
<td>0.348</td>
<td>0.207</td>
</tr>
<tr>
<td>LIM</td>
<td>343</td>
<td>0.177</td>
<td>0.374</td>
<td>0.292</td>
<td>0.328</td>
<td>0.127</td>
<td>0.207</td>
</tr>
<tr>
<td>GAGA</td>
<td>279</td>
<td>0.211</td>
<td>0.238</td>
<td>0.329</td>
<td>0.241</td>
<td>0.207</td>
<td>0.127</td>
</tr>
<tr>
<td>BRIDGE</td>
<td>275</td>
<td>0.216</td>
<td>0.273</td>
<td>0.340</td>
<td>0.293</td>
<td>0.252</td>
<td>0.127</td>
</tr>
<tr>
<td>t-EB</td>
<td>125</td>
<td>0.193</td>
<td>0.203</td>
<td>0.249</td>
<td>0.272</td>
<td>0.203</td>
<td>0.193</td>
</tr>
</tbody>
</table>

Table 6: To investigate the performance of robust Linear Bayes classifier 1,000 contaminated simulated gene expressions datasets average results (with increasing variance \( \sigma \)) are shown below:

<table>
<thead>
<tr>
<th>Methods</th>
<th>DE genes</th>
<th>Linear Bayes</th>
<th>SVM</th>
<th>KNN</th>
<th>Adaboost</th>
<th>LogitBoost</th>
<th>t-Linear Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>EB</td>
<td>102</td>
<td>0.226</td>
<td>0.274</td>
<td>0.293</td>
<td>0.387</td>
<td>0.321</td>
<td>0.216</td>
</tr>
<tr>
<td>SAM</td>
<td>342</td>
<td>0.258</td>
<td>0.371</td>
<td>0.354</td>
<td>0.348</td>
<td>0.225</td>
<td>0.207</td>
</tr>
<tr>
<td>t-EB</td>
<td>611</td>
<td>0.209</td>
<td>0.174</td>
<td>0.219</td>
<td>0.370</td>
<td>0.348</td>
<td>0.207</td>
</tr>
<tr>
<td>LIM</td>
<td>343</td>
<td>0.177</td>
<td>0.374</td>
<td>0.292</td>
<td>0.328</td>
<td>0.127</td>
<td>0.207</td>
</tr>
<tr>
<td>GAGA</td>
<td>279</td>
<td>0.211</td>
<td>0.238</td>
<td>0.329</td>
<td>0.241</td>
<td>0.207</td>
<td>0.127</td>
</tr>
<tr>
<td>BRIDGE</td>
<td>275</td>
<td>0.216</td>
<td>0.273</td>
<td>0.340</td>
<td>0.293</td>
<td>0.252</td>
<td>0.127</td>
</tr>
<tr>
<td>t-EB</td>
<td>125</td>
<td>0.193</td>
<td>0.203</td>
<td>0.249</td>
<td>0.272</td>
<td>0.203</td>
<td>0.193</td>
</tr>
</tbody>
</table>
others. For the gene set obtained by GaGa and r-EB approaches the classifier SVM, KNN and Logitboost produced smaller MR in the colon cancer original dataset. Again, for the performance of our proposed method we added only 10% outliers and as other classifier which produced smaller MR in the colon cancer original dataset.

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions.

REFERENCES


