Classifier design given an uncertainty class of feature distributions via regularized maximum likelihood and the incorporation of biological pathway knowledge in steady-state phenotype classification

Mohammad Shahrokh Esfahani\textsuperscript{a,*}, Jason Knight\textsuperscript{a}, Amin Zollanvari\textsuperscript{a}, Byung-Jun Yoon\textsuperscript{a}, Edward R. Dougherty\textsuperscript{a,b}

\textsuperscript{a} Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA
\textsuperscript{b} The Computational Biology Division of the Translational Genomics Research Institute, Phoenix, AZ, USA

\textbf{A B S T R A C T}

Contemporary high-throughput technologies provide measurements of very large numbers of variables but often with very small sample sizes. This paper proposes an optimization-based paradigm for utilizing prior knowledge to design better performing classifiers when sample sizes are limited. We derive approximate expressions for the first and second moments of the true error rate of the proposed classifier under the assumption of two widely used models for the uncertainty classes: \textit{\epsilon}-contamination and \textit{p}-point classes. The applicability of the approximate expressions is discussed by defining the problem of finding optimal regularization parameters through minimizing the expected true error. Simulation results using the Zipf model show that the proposed paradigm yields improved classifiers that outperform traditional classifiers which use only training data. Our application of interest involves discrete gene regulatory networks possessing labeled steady-state distributions. Given prior operational knowledge of the process, our goal is to build a classifier that can accurately label future observations obtained in the steady state by utilizing both the available prior knowledge and the training data. We examine the proposed paradigm on networks containing NF-\textit{k}B pathways, where it shows significant improvement in classifier performance over the classical data-only approach to classifier design. Companion website: \url{http://gsp.tamu.edu/Publications/supplementary/shahrokh12a}.

Published by Elsevier Ltd.

1. Introduction

In recent years, phenotypic classification based on genomic data has confronted the pattern recognition community with very small samples. There can be tens of thousands of potential features (gene expressions) but the sample sizes tend to be small, typically under 100 and often less than 50. This makes classification problematic. A promising approach to alleviate the problem is the use of prior knowledge. For example, the usual procedure for classifier design is to apply a classification rule to a set of features and sample data with the result being a designed classifier that will be applied to the population (all future observations). Prior knowledge can play a role in deciding upon the nature of the data and the original list of features. Knowledge may also be used in choosing a classification rule based on the nature of physical characteristics. The salient point from our perspective herein is that, once the features, sampling procedure, and classification rule are decided upon, from that point on the typical classification rule proceeds without operational knowledge concerning the features. In particular, no assumptions are made regarding the feature-label distribution (population) from which the sample data have been drawn. It is in this regard that the classification procedure is said to be “model-free.” If knowledge concerning the feature-label distribution is available, then it can be used in classifier design.

A good bit of attention has been paid to the difficulty of error estimation in such circumstances. This has led to the incorporation of prior knowledge in error estimation, for instance, sample-size requirements based on an uncertainty class of feature-label distributions \cite{1} and minimum-mean-square-error (MMSE) estimation based on a prior distribution over an uncertainty class of feature-label distributions \cite{2,3}.

Here the issue is incorporation of prior knowledge into the design of the classifier itself, not the estimation of its error. A number of recent studies have proposed various methods that can enhance classifiers by incorporating prior knowledge. For example, to improve classification performance, several studies...
have proposed to interpret the gene expression data at the level of functional modules (i.e., pathways), instead of at the level of individual genes, by utilizing available pathway knowledge [4,5]. These pathway-based methods try to infer the activity level of a given pathway by analyzing the expression of its member genes, which is then used as a potential feature. These studies have shown that such “pathway markers” are generally more reproducible compared to “gene markers” and that they lead to better classification performance. Another example is the network-reducible compared to “gene markers” and that they lead to better shown that such “pathway markers” are generally more reproducibility compared to both gene and pathway markers. Considering that pathways are partial representations of the gene regulatory network and that the PPI network provides a skeleton of the biological network underlying cells, the aforementioned methods can be viewed as attempts to construct better classifiers by integrating partial network knowledge with measurement data. A Bayesian approach to using prior knowledge for classification has been taken by defining a prior distribution on an uncertainty class of feature-label distributions and deriving a classifier that is optimal with respect to the posterior distribution resulting from utilizing sample data in conjunction with the prior distribution [8,9]. This approach has been applied to classify the mammalian cell cycle as normal or mutated [10]. Although recent advances in pathway-based and network-based classification have demonstrated the potential for utilizing prior knowledge to improve genomic classification, currently available methods mostly rely on heuristics. In this paper, we propose a general paradigm for classification that incorporates prior knowledge along with the data in the context of an optimization procedure. We address optimal discrete classification where prior knowledge is restricted to an uncertainty class of feature distributions that absent a prior distribution on the uncertainty class, a problem that arises directly for biological classification using pathway information.

In our case, the application in mind is phenotype classification based on gene (or protein) expression measurements in the steady-state of a biological network. This “biomarker problem” is perhaps the most active area of research in genomics owing to the potential for disease diagnosis and prognosis. Rather than depending only on expression data, one can use classical genetic pathway information to provide prior knowledge and augment classifier design. The example laid out in this paper involves the following chain: [pathways] → [class of networks] → [class of steady-state distributions]. Prior knowledge in the form of a set of pathways constrains the possible behaviors of the dynamical system to an “uncertainty class” of networks consistent with the pathway information [11]. Each of these possesses a steady-state distribution, thereby yielding an uncertainty class of steady-state distributions. Fig. 1 shows an illustrative view of this process chain. Detailed description of this figure is given in Section 6. Hence, rather than assuming nothing is known about the feature-label distribution than what can be extracted from the data during classifier design, we can impose the constraint that the feature distribution belongs to the uncertainty class of steady-state distributions shown by a box in the middle of Fig. 1. Put simply, a classifier is designed based on the uncertainty class of steady-state distributions, denoted by \( \Pi^0 \) and \( \Pi^1 \) in Fig. 1, and the steady-state data.

We emphasize that while the particular application motivating our interest involves the generation of a steady-state uncertainty class from genetic pathway information, the theoretical content of this paper lies solely within classification theory. Classifier design assuming an uncertainty class of feature distributions. In line with that focus, we provide analytical characterization of the first and second moments of the true error for two well-known uncertainty models, \( \varepsilon \)-contamination and \( p \)-point uncertainty classes, under the assumption of

![Fig. 1. An illustrative example of the chain: [pathways] → [class of networks] → [class of steady-state distributions]. In this schematic view, an intermediate step is applied to construct a class of dynamical systems whose behaviors are consistent with the given pathways, for example, see the methods in [11,12]. Two uncertainty classes are shown by \( \Pi^0 \) and \( \Pi^1 \) for labels zero and one, respectively. These classes will be employed as the prior knowledge in the classifier design.](image-url)
stratified sampling. Characterization of these moments is basic to understand the behavior of a classification rule and has a long history in pattern recognition, most commonly with stratified sampling [1,13–29]. Recently, the issue of true-error moments has been addressed in the context of the joint distribution of the true and estimated error moments, in this case the most important moment being the second-order mixed moment between the true and estimated errors because this mixed moment is critical to characterize the accuracy of the error estimate [1,13,18,30].

The paper is organized in the following manner. In Section 2, we introduce our proposed paradigm. True error statistics for the stratified sampling case are derived in Section 3. Section 4 contains a brief discussion on the regularization parameter defined and used throughout the paper. Simulation results are shown in Sections 5 and 6 where we show the improvement of the designed classifier over the histogram rule in synthetic and biologically inspired cases. Finally, Section 7 contains concluding remarks.

We use the following notation and abbreviations. Boldface lower case letters denote column vectors. The cardinality of the set, $\mathcal{I}$, is denoted by $|\mathcal{I}|$. $\pi(k)$ and $\pi^2$ denote the $k$-th element and the transpose of the vector $\pi$, respectively. $P_r(A)$ denotes the probability of event $A$. The binomial distribution is shown by $bin(n,p)$. $bin(n,p) = x$ is used to denote the binomial random variable having value $x$. The trinomial distribution is shown by $trin(n,p_1,p_2)$. Thus,

$$
Pr(trin(n,p_1,p_2) = (x_1,x_2)) = \frac{n!}{x_1!x_2!(n-x_1-x_2)!} p_1^{x_1} p_2^{x_2} (1-p_1-p_2)^{n-x_1-x_2}.
$$

To show the comparison between two vectors, we use $\pi_1 \leq \pi_2$ meaning that the vector $\pi_1$ is element-wise less than or equal to $\pi_2$. The notation $E_k(x(k))$ is used to denote taking expectation of $g(x)$ with respect to the subscript $x$. The indicator function for the event $A$ is shown by $I_A$.

2. Regularized maximum-likelihood

In this section, we propose an optimization paradigm for classifier design that utilizes both an uncertainty class (from prior knowledge) and the available training data. Let $\pi_{0,k}(y) = Pr(X=k|Y=y)$ be the true conditional distribution of the feature $X=k \in \{1,\ldots,b\}$ given the class label $y \in \{0,1\}$, and let $C_y = Pr(Y=y)$ be the prior distribution of the class label. We can build a classifier by first finding label conditional probabilities $\pi^y(k)$ that estimate the true probabilities $\pi_{0,k}(y)$ and then defining

$$
\psi(k) = \frac{1}{c_1 \pi^1(k) + c_2 \pi^0(k)} \begin{cases}
1 & \text{if } c_1 \pi^1(k) \geq c_2 \pi^0(k) \\
0 & \text{otherwise}
\end{cases}
$$

This can be viewed as using the “plug-in rule” in the Bayes classifier $\psi(k) = 1_{c_1 \pi^1(k) \geq c_2 \pi^0(k)}$. In the absence of prior knowledge, the label-conditional distribution $Pr(X=k|Y=y) = \pi_{0,k}(y)$ is estimated solely based on the training data by solving the following maximum-likelihood problem:

$$
\min_{\pi^y} \sum_{k=1}^{b} n_k^y \log \pi^y(k),
$$

where $n_k^y$ is the number of sample points at state $k$ with label $y$ and $e$ is the all-one column vector. The solution to (2) is

$$
\hat{\pi}^y_{data}(k) = \frac{u_k^y}{n_y},
$$

where $n_y$ is the number of sample points with label $y$.

We now assume that we have uncertainty classes, $\mathcal{P}_y = \{\pi_1^y, \pi_2^y, \ldots, \pi_b^y\}, y=0,1$, e.g., see Fig. 1, conveying the prior network knowledge of the label-$y$ conditional distribution, $\pi_{0,y}$. We adapt (2) to form the following weighted-sum optimization problem for the class labels $y=0,1$, which includes a term contributed by the uncertainty class:

$$
\min_{\pi^y} \sum_{k=1}^{b} \left(1- \lambda_y\right) n_k^y \log \pi^y(k) + \lambda_y \ell(\pi^y, \mathcal{P}_y).
$$

The regularization parameter $\lambda_y \in [0,1]$ reflects the uncertainty of the labeled training data compared to the total amount of uncertainty in our prior knowledge and $\ell : (\mathcal{P}_1 \times \mathcal{P}_2)^b \rightarrow [0,\infty)$, where $\mathcal{P}_y$ is the standard unit $(b-1)$-simplex and $\mathcal{P}^b$ is any uncertainty class containing $|\mathcal{P}||b|$-dimensional distributions, is a nonnegative function to measure the dissimilarity between a given $\pi^y$ and the uncertainty class.

If the objective function in (4) is a convex function, then the optimization problem can be solved efficiently. Since the log-likelihood function for the multinomial distribution is concave (i.e., the negative log-likelihood function for $\pi^y(k), k=1,\ldots,b$, given the sample, is convex), it is sufficient to use a convex function for $\ell$ (i.e., the regularizer term) in (4) to make it a convex programming problem. We use

$$
\ell(\pi^y, \mathcal{P}_y) = \frac{1}{|\mathcal{P}|} \sum_{i=1}^{|\mathcal{P}|} D(\pi^y || \pi^i),
$$

where $D(\pi^y || \pi^i) = \sum_{k=1}^{b} \pi^y(k) \log \pi^y(k) / \pi^i(k)$ is the Kullback-Leibler (information) distance (KL-distance).

Lemma 1 (RML Classifier). Suppose that the dissimilarity function $\ell$ is defined as (5). Then, the solution to the regularized maximum-likelihood (RML) problem in (4) is obtained bin-wise as

$$
\hat{\pi}^y_{RML}(k) = \frac{(1-\lambda_y) u_k^y + \lambda_y \pi_0(k)}{(1-\lambda_y) n_y + \lambda_y}, \quad y \in \{0,1\}, \forall k=1,\ldots,b,
$$

where $\bar{\pi}^y(k)$ is the probability of the $k$-th bin obtained from the average of $\pi_1, k=1,\ldots,b$ in the corresponding uncertainty class $\mathcal{P}_y, y \in \{0,1\}$. The corresponding RML classifier can be found by plugging $\hat{\pi}^0_{RML}$ and $\hat{\pi}^1_{RML}$ in Eq. (1).

Proof. Refer to Appendix A.

Consider the following two special cases:

1. Suppose the uncertainty in the information extracted from the training data is much less than that in the prior network knowledge. In the limiting case, $\lambda_y \to 0$ and

$$
\lim_{\lambda_y \to 0} \hat{\pi}^y_{RML}(k) = \frac{u_k^y}{n_y}, \quad \forall k=1,\ldots,b.
$$

This is consistent with our expectation: if there is infinite amount of training data (hence no uncertainty therein), the classifier can be perfectly estimated from the data.

2. Suppose we have very good prior network knowledge, so that the uncertainty in this knowledge is much smaller compared to that extracted from the data. In the limiting case, $\lambda_y \to 1$ and

$$
\lim_{\lambda_y \to 1} \hat{\pi}^y_{RML}(k) = \bar{\pi}^y(k), \quad \forall k=1,\ldots,b.
$$

If we have perfect knowledge of the steady-state distribution, then we do not need training data.
In this paper we consider two models having finite uncertainty classes:

2.1. \(\varepsilon\)-contamination uncertainty class

The \(\varepsilon\)-contamination class has been used for modeling uncertainty in a wide range of applications, including robust hypothesis testing [31], robust Wiener filter (uncertainty about the spectral density) [32,33], Bayesian robust optimal linear filter design [34], robust decision making problems [35], and minimax robust quickest change detection (with the application in intrusion detection in computer networks and security systems) [36]. In [32–34], the \(\varepsilon\)-contamination class contains all the power spectral densities (PSD) in the vicinity of the nominal PSD. In [31,36], the \(\varepsilon\)-contamination class contains all the probability densities in the vicinity of the nominal one.

Here, we use \(\varepsilon\)-contamination to model the uncertainty about the label-conditional probabilities. We define the \(\varepsilon\)-contamination class of multinomial distributions associated with each label as the class containing the distributions in the form of

\[
\pi' = (1-\varepsilon_p)\pi_{ac}^p + \varepsilon_p \pi; \quad y \in \{0,1\}
\]

where \(\varepsilon_p \in [0,1]\) is the degree of contamination and \(\pi\) is one of a finite number of randomly chosen densities from \(S_\pi\). Increasing \(\varepsilon_p\) corresponds to increasing the variance of prior knowledge about the true distribution. We assume a uniform distribution for the contamination part domain whose relative interior of the volume under the \((b-1)\)-simplex. Since our application of interest is related to steady-state classifiers, we assume that in the simplex the corners and axes have measure zero.

2.2. \(p\)-point uncertainty class

The \(p\)-point uncertainty class has been used to model uncertainty in rate distortion problems, detection problems, robust Wiener filter design, and robust non-stationary signal estimation [33,37–42]. In our application of interest, we often only know that the cell, in its steady state, spends a specific portion of time in a subset of states but know nothing about the details of these states individually. Hence, to model this prior knowledge, we can see the problem as a partitioning scenario: if we partition the state space, then the amount of time that the cell spends in each subset in the partition is known. Therefore, we can say that the label-conditional distributions belong to an uncertainty class of distributions satisfying the following constraints:

\[
\sum_{k=1}^b \pi(k|_{y \in S_y}) = \sum_{k=1}^b \pi_{ac}^k(k|_{y \in S_y}); \quad p = 1, \ldots, m_y,
\]

where \(\pi_{ac}^k\) is the actual steady-state distribution, \(S_y^1, \ldots, S_y^m\) form a partition of the state space denoted by \(p^y\), and \(\pi \in S_\pi\) is any density function.

We will use the following notation throughout the paper for the probability mass function cumulated in each partition:

\[
\sum_{k=1}^b \pi_{ac}^k(k|_{y \in S_y}) = \alpha_{p^y}; \quad p = 1, \ldots, m_y.
\]

Moreover, we define the following mapping from state space to the partition:

\[
P^y : \{1, \ldots, b\} \rightarrow \{1, \ldots, m_y\}; \quad y = 0,1.
\]

In the extreme case, \(m_y = 1\) means that we only know that the label-conditional probabilities for the bins sum up to 1, which corresponds to a minimal amount of prior knowledge. On the other hand, \(m_y = b\), i.e. \(S_y^p = 1\), for any \(p \in \{1, \ldots, m_y\}, y \in \{0,1\}\), means that we are certain about the label-conditional distributions, because we are given all bin probabilities—hence minimal variance in the uncertainty class (for more details refer to Section 1 of the supplementary materials on the companion website).

3. Moments for the true error

For a classifier \(\psi_n\) trained on the sample data \(S_n\), the probability of error is defined as \(\epsilon_{\text{data}} = \Pr(\psi_n(X) \neq Y|S_n)\). The overall performance of the classification rule can be evaluated by the expected classification error, \(E(\epsilon_{\text{data}}) = \mathbb{E}_n[\Pr(\psi_n(X) \neq Y|S_n)]\), over all samples of size \(n\). When prior knowledge (denoted by “uc” for uncertainty class) is incorporated into classifier design, we rewrite the probability of error as

\[
\epsilon_{\text{data, uc}} = \Pr(\psi_n(\hat{f}^0, \hat{f}^1)|X) \neq Y|S_n, \hat{f}^0, \hat{f}^1).
\]

In this section we provide analytical representation of the first and second moments for the error in the \(\varepsilon\)-contamination and \(p\)-point uncertainty models under stratified sampling, in which sampling is performed from classes 0 and 1 in accordance with their prior probabilities. Since we incorporate prior knowledge, the moments are computed relative to all samples of size \(n\) and the uncertainty-class space. They take the form

\[
E(\epsilon_{\text{RML}}) = E(\epsilon_{\text{data, uc}}) = E(\epsilon_{\text{RML}, uc}) = \mathbb{E}_n[\Pr(\psi_n(\hat{f}^0, \hat{f}^1)|X) \neq Y|S_n]|I^0, I^1, \hat{f}^0, \hat{f}^1).
\]

We derive tight approximations for these moments for \(\lambda_{uc} \in (0,1)\). The cases \(\lambda_{uc} \notin (0,1)\) can be handled with a slight modification to the proof.

**Theorem 1** (First-order moment of the true error: \(\varepsilon\)-contamination class). Suppose that the uncertainty classes, \(I^0\) and \(I^1\), come from \(\varepsilon_0\) and \(\varepsilon_1\)-contamination classes, respectively. Then, the first-order moment of the true-error of the RML classifier defined in Lemma 1 is given by

\[
E(\epsilon_{\text{RML}}) = c_0 \sum_{k=1}^b \pi_{ac}^0(k) \left( \sum_{l_0=0}^m \sum_{l_1=0}^m \Pr(\text{bin}(n_0, \pi_{ac}^0(k)) = l_0) \right) \\
+ c_1 \sum_{k=1}^b \pi_{ac}^1(k) \left( \sum_{l_1=0}^m \sum_{l_0=0}^m \Pr(\text{bin}(n_1, \pi_{ac}^1(k)) = l_1) \right)
\]

where the random variables \(\mathcal{X}^y_{\mathcal{Z}^y_{\lambda_{uc}}}, k = 1, \ldots, b, \forall p^y = 0, 1, y \in \{0,1\}\), approximately have the following probability mass function (pmf):

\[
\begin{align*}
\Pr(\mathcal{X}^y_{\mathcal{Z}^y_{\lambda_{uc}}} = 0) &= \Phi \left( -\frac{\mu_{\lambda_{uc}}^y}{\sigma_{\lambda_{uc}}^y} \right) \\
\Pr(\mathcal{X}^y_{\mathcal{Z}^y_{\lambda_{uc}}} = m) &= \Phi \left( \frac{m-\mu_{\lambda_{uc}}^y}{\sigma_{\lambda_{uc}}^y} \right) - \Phi \left( \frac{m-1-\mu_{\lambda_{uc}}^y}{\sigma_{\lambda_{uc}}^y} \right); \quad m = 1, \ldots, n_y \\
\Pr(\mathcal{X}^y_{\mathcal{Z}^y_{\lambda_{uc}}} = 0) &= 0; \quad m \geq n_y + 1
\end{align*}
\]

(16)

\(\Phi(\cdot)\) being the standard normal distribution. In Eq. (16) we have

\[
\mu_{\lambda_{uc}}^y = \gamma_y f_{\lambda_{uc}}^y (1-\varepsilon_p)\pi_{ac}^0(k) - (1-\varepsilon_p)\pi_{ac}^1(k) + \frac{\varepsilon_p \gamma_y f_{\lambda_{uc}}^y - \gamma_p \gamma_y}{b}
\]
\[ \sigma^2_{k,y} = \left( \frac{\alpha^2_{p,k}(b-1)}{b^2 |\mathcal{P}|(b+1)} + \frac{\alpha^2_{p,k}(b-1)}{b^2 |\mathcal{P}|(b+1)} \right) / g^2_y; \quad \forall k = 1, \ldots, b \]  

where \( \mathcal{P} \) denotes 1 – y and  
\[ g_y = (1-\lambda_y)\theta_l y(1 - \lambda_y) + \lambda_y \]  
\[ \sigma^2_y = \frac{\sigma^2_{k,y}}{1 - \lambda_y} \]  

**Proof.** Refer to Appendix B.

**Theorem 2** (First-order moment of the true error: p-point class). Let the uncertainty classes, \( \Pi_i^0 \) and \( \Pi_i^1 \), be modeled by the p-point model with partition probabilities \( \omega^0_k \) and \( \omega^1_k \) with \( p = 1, \ldots, m_y \) for labels 0 and 1, respectively. Then, the first-order moment of the true error of the RML classifier defined in Lemma 1 can be written as in Eq. (15) in which the random variables \( z_y^k, j = 1, \ldots, b \), for any \( y = 0, \ldots, n_y \), approximately have the pmf as defined in Eq. (16), whereas assuming the definitions in Eq. (18), we have  
\[ m_{k,0} = g_y \sigma^2_y + \sigma^2_{k,y} \]  
\[ \sigma^2_{k,y} = \left[ \frac{\alpha^2_{p,k}(b-1)}{b^2 |\mathcal{P}|(b+1)} \frac{(s^y_{p,k}(\mathbf{RML}) - 1)}{|\mathcal{P}|(b+1)} + \frac{\alpha^2_{p,k}(b-1)}{b^2 |\mathcal{P}|(b+1)} \frac{(s^y_{p,k}(\mathbf{RML}) - 1)}{|\mathcal{P}|(b+1)} \right] / g^2_y. \]  

**Proof.** Refer to Appendix B.

**Theorem 3** (Second-order moment of the true error). The second-order moment of the true-error of the RML classifier defined in Lemma 1 can be decomposed as  
\[ \text{E}(\hat{\sigma}^2_{\text{RML}}) = \text{E}_{\Pi_0, \Pi_1} \left[ c_0 \sum_{k=1}^{b} \left( \sigma_{p,k}^0(k) \right)^2 A^1 + c_1 \sum_{k=1}^{b} \left( \sigma_{p,k}^1(k) \right)^2 A^0 \right] + \text{E}_{\Pi_0, \Pi_1} \left[ c_0 \sum_{k=1}^{b} \sigma_{p,k}^0(k) \sigma_{p,k}^1(k) B^1 + c_1 \sum_{k=1}^{b} \sigma_{p,k}^1(k) \sigma_{p,k}^0(k) B^0 \right] + \text{E}_{\Pi_0, \Pi_1} \left[ c_{0c} c_1 \sum_{k=1}^{b} \sigma_{p,k}^0(k) \sigma_{p,k}^1(k) C^1 + c_{0c} c_1 \sum_{k=1}^{b} \sigma_{p,k}^1(k) \sigma_{p,k}^0(k) C^0 \right]. \]  

**Proof.** Refer to Appendix E.

**4. The regularization parameter**

The regularization parameter \( \lambda_y \) in (4) should be adjusted based on the relative uncertainty between the training data and the prior knowledge. We propose three approaches for tuning the regularization parameter.

**4.1. Minimizing the expected true error**

The optimal value of the regularization parameter, based on expected true error, can be found by solving the following optimization problem:

\[ \lambda^* = \arg \min_{0 < \lambda < 1} \text{E}(\hat{\sigma}^2_{\text{RML}}). \]  

**Proof.** Refer to Appendix C.

The joint distribution of \( \hat{\sigma}_{y^k}^0 \) and \( \hat{\sigma}_{y^k}^1 \) (similarly for \( \hat{\sigma}_{y^0}^1 \) and \( \hat{\sigma}_{y^1}^0 \)) and the joint distribution of \( \hat{\sigma}_{y^0}^0 \) and \( \hat{\sigma}_{y^1}^1 \) (similarly for \( \hat{\sigma}_{y^1}^0 \) and \( \hat{\sigma}_{y^0}^1 \)), which depend on the uncertainty classes, are given in Appendix D for c-contamination and p-point classes.

**4.2. SURE-tuning of regularization parameter**

One way to evaluate the performance of the estimator in Lemma 1 is to use the mean-squared error (MSE) of the estimator. In the problem of multiclass distribution estimation, the MSE can be expanded as follows:

\[ \text{MSE} = \mathbb{E} \left[ \sum_{k=1}^{b} \left( \hat{\sigma}_{y^k}^2(k) - \bar{\sigma}_{y^k}^2(k) \right)^2 \right]; \quad y = 0, 1. \]  

where we drop the subscript RML and instead use the regulariza-
**Proof.** Refer to Appendix E.
Minimizing the SURE-estimate of the MSE with respect to the regularization parameter $\lambda$ yields the following result for case of $n \geq 2$.

**Corollary 1** (SURE-optimal regularization parameter). The SURE-optimal regularization parameter of the estimator defined in Lemma 1 is given by

$$
\lambda_{\text{SURE}}^* = \begin{cases} \\
\lambda & 0 \leq \lambda \leq 1 \\
\text{otherwise}
\end{cases}
$$

(25)

in which we have

$$
\lambda = \frac{n[1 - \sum_{k=1}^{b}(u_k/n)^2]}{(n-1)[1 + \sum_{k=1}^{b}(\pi(k)\pi(k) - 2u_k/n)].}
$$

**Proof.** The corollary results from equating the derivative of (24) (with respect to $\lambda$) to zero, while considering the boundary of the feasible region of the $\lambda$ (the SURE estimate in Eq. (24) is continuous in $[0,1]$).

Fixing the uncertainty class, as $n \to \infty$, we obtain

$$
\lim_{n \to \infty} \lambda_{\text{SURE}}^* = \frac{1 - \|\pi_{uc}\|^2}{1 - \|\pi_{uc}\|^2 + \|\pi_{uc} - \pi_{uc}^*\|^2},
$$

(26)

in which $\|\cdot\|_2$ denotes the $\ell_2$-norm of vector $\pi$.

To illustrate the effects of different sample sizes and different amounts of uncertainty on $\lambda_{\text{SURE}}^*$, we have run a simulation assuming an $\epsilon$-contamination uncertainty class and that the actual distribution follows a Zipf model with parameter $a=1$ (a detailed description of the Zipf model will be provided in Section 5). We observe the behavior of $\lambda_{\text{SURE}}^* = E[I|E[\lambda_{\text{SURE}}^* | I]]$ using Monte-Carlo expectation over 4000 training datasets (for each fixed sample size) and 500 uncertainty classes. We consider different values for $\epsilon \in [0,1]$ and sample size $n$. Fig. 2 shows the 3-D figure with $n$ as the $x$-axis and $\epsilon$ as the $y$-axis. As $\epsilon \to 1$ (uncertainty is increased), for a fixed sample size, $\lambda_{\text{SURE}}^*$ decreases as in Eq. (26).

### 4.3. A heuristic approach

Although one can use a stochastic algorithm to solve (22) (which is not necessarily guaranteed to achieve the global minimum), or use the result in **Corollary 1**, we can take a heuristic approach for specifying $\lambda_{Y}^*$. Suppose $|I^0|$ and $|I^1|$ are the sizes of the uncertainty classes for labels 0 and 1, respectively. Proceeding heuristically and denoting the $i$th distribution with label $y$ as $\pi_{i,y}^*$, we form a network-based estimate, $\hat{\pi}_{uc}^* = \overline{\pi_{i,y}}$, by averaging the $\pi_{i,y}^*, i=1,\ldots,|I^y|$. A data-based estimate, $\hat{\pi}_{uc}^{\text{data}}$, is obtained from (3). Under this setting, we can estimate the relative uncertainty by

$$
\lambda_{Y}^* = \frac{\text{Var}(\hat{\pi}_{uc}^{\text{data}})}{\text{Var}(\hat{\pi}_{uc}^{\text{data}}) + \text{Var}(\hat{\pi}_{uc}^*)},
$$

(27)

where

$$
\text{Var}(\hat{\pi}_{uc}^{\text{data}}) = \frac{b}{\sum_{k=1}^{b}n_{uc}(k)(1 - n_{uc}^*(k))}
$$

(28)

In (28), the variance of the training data is independent of the uncertainty class model and can therefore be analytically computed by

$$
\text{Var}(\hat{\pi}_{uc}^{\text{data}}) = \frac{b}{\sum_{k=1}^{b}n_{uc}(k)(1 - n_{uc}^*(k))}
$$

(29)

The variance of the uncertainty class depends on the underlying model of the uncertainty class. We obtain

$$
\text{Var}(\hat{\pi}_{uc}^{\text{data}}) = \frac{\sigma^2_{uc}(b-1)}{(b+1)b^2},
$$

(30)

for a $\epsilon$-contamination class and

$$
\text{Var}(\hat{\pi}_{uc}^{\text{data}}) = \frac{m_{uc}}{\sum_{p=1}^{m_{uc}}(1 - |s_p|^2)(|s_p|^2 - 1)}
$$

(31)

for a $p$-point uncertainty class (Refer to **Section 1** of the supplementary materials on the companion website).

### 5. Numerical experiments

In this section, we evaluate the performance of the classifiers designed using the proposed optimization paradigm. Let $\epsilon_{\text{RML}}$ denote the error of the RML classifier designed via (4) using the estimated probabilities given in **Lemma 1**. Let $\epsilon_{\text{hist}}$ denote the error of the traditional histogram rule obtained by designing the classifier as in (1) using the data-based estimate $\hat{\pi}_{uc}^{\text{data}}$ given in (3). The exact expression for $E(\epsilon_{\text{hist}})$ is given in [18].

![Fig. 2](image-url) Illustrating the expected value of $\lambda_{\text{SURE}}^*$ for different amount of uncertainty and sample sizes. The result is for $\epsilon$-contamination classes. The uncertainty class size, $|I|$ is set to 50.
We use both the approximation in (15) and Monte Carlo simulations for assessing \( \epsilon_{\text{RML}} \). In the Monte-Carlo estimation, based on the given assumption for the structure of the uncertainty classes, we generate \( T \) pairs of uncertainty classes denoted by \((I_{ml}^{0}, I_{ml}^{1})\), \( l=1, \ldots , T \). Then for each pair, based on the given model for the true distributions \( \pi_{uc}^{y} \), \( y=0,1 \), we generate \( M \) sample sets with size \( n \) denoted by \( S_{2m}^{n} \), \( m=1, \ldots , M \). For each sample \( S_{2m}^{n} \), we estimate the conditional probabilities using Lemma 1. The estimates \( \hat{\pi}_{uc}^{y}(k) \) are then used to construct the classifier, as defined in (1). The error of the classifier designed using \( S_{2m}^{n} \) (i.e., \( m \)th sample set generated for the \( l \)th pair) is then computed analytically using the actual distribution \( \pi_{uc}^{y} \) which was used to generate the sample. We denote this error by \( \epsilon_{\text{RML}}^{l,m} \). The first- and the second-order moments of the true error are approximated by

\[
E(\epsilon_{\text{RML}}^{l,m}) \approx \frac{1}{MT} \sum_{l=1}^{T} \sum_{m=1}^{M} \epsilon_{\text{RML}}^{l,m},
\]

\[
E(\epsilon_{\text{RML}}^{l,m})^{2} \approx \frac{1}{MT} \sum_{l=1}^{T} \sum_{m=1}^{M} (\epsilon_{\text{RML}}^{l,m})^{2},
\]

via Monte-Carlo simulation. We estimate the variances, \( \text{Var}(\hat{\pi}_{\text{data}}^{y}) \) and \( \text{Var}(\hat{\pi}_{\text{uc}}^{y}) \) in (27) as

\[
\text{Var}(\hat{\pi}_{\text{data}}^{y}) = \frac{b}{n} \sum_{k=1}^{b} \left( \frac{1}{n} - \frac{\hat{U}_{k}^{y}}{n} \right),
\]

\[
\text{Var}(\hat{\pi}_{\text{uc}}^{y}) = \frac{1}{|I_{ml}^{0}|-1} \sum_{k=1}^{|I_{ml}^{0}|} \left( \pi_{uc}^{y}(k) - \hat{\pi}_{uc}^{y}(k) \right)^{2}.
\]

5.1. Performance assessment using a Zipf model

We first assume that the true label-conditional distributions (i.e., \( \pi_{uc}^{y}, y=0,1 \)) follow a Zipf model

\[
\pi_{uc}^{y}(k) = \frac{\zeta}{k^{\alpha}}, \quad \pi_{uc}^{y}(k) = \pi_{uc}^{0}(b-k+1),
\]

where \( \zeta \) is a normalizing constant. The Zipf distribution, originally introduced by G.K. Zipf to model the frequency of words in common text [46], is a well-known power-law discrete distribution, encountered in many applications. In particular, it has been used as a model to study the moments of error estimators for discrete classifiers [18]. As \( \alpha \to 0 \), both conditional distributions (\( y \in \{0,1\} \)) tend to become uniform. Hence the classification problem becomes more difficult, resulting in a larger Bayes error. We assume \( \pi_{uc}^{y} = 0.5; y \in \{0,1\} \) are known. We consider \( b=8 \) (which corresponds to the number of states in a three-gene Boolean network when modeling genomic regulatory networks [47]). We evaluate the proposed framework under two different scenarios. First, we examine the accuracy of our approximate expressions by comparing them with the Monte-Carlo simulation while one has access to the exact regularization parameters defined by applying (29)-(31). The motivation is to test the accuracy of our approximation when the regularization parameters are found off-line, independent of the given sample data. In the second scenario, we assume that one has to estimate the regularization parameters based on the given data and uncertainty classes using Eqs. (34) and (35). Depending on the underlying assumption for the uncertainty classes, for each size \( n \) and each set of model parameters (e.g., \( \epsilon_{0}, \epsilon_{1} \), or partitions in the \( p \)-point class), we generate \( T=1000 \) different pairs of uncertainty classes, \((I_{ml}^{0}, I_{ml}^{1})\), \( l=1, \ldots , 1000 \), for which we generate \( M=2000 \) samples, \( S_{2m}^{n} \), \( m=1, \ldots , 2000 \), for estimating the first- and the second-order moments of the true error, \( E(\epsilon_{\text{RML}}^{l,m}) \) and \( E(\epsilon_{\text{RML}}^{l,m})^{2} \). For the approximate second-order moments, where there are double-integrals, we use the adaptive Simpson algorithm for approximating the integral values. Some results for the various

Fig. 3. The first- and second-moment of the true error of the RML classifier and Histogram rule with \( \epsilon \)-contamination uncertainty classes with size, \( |I_{ml}^{0}| = |I_{ml}^{1}| = 250 \). Steady-state distributions with \( b=2^{7} \) states are considered. In (a) and (c) the regularization parameters, are exact as in (29) and (30). In (b) and (d), they are estimated as in (34) and (35). (a) Expected true error with exact \( \pi_{uc}^{y} \) and \( \lambda_{1} \). (b) Expected true error with data-dependent \( \pi_{uc}^{y} \) and \( \lambda_{1} \). (c) Second-order moment with exact \( \pi_{uc}^{y} \) and \( \lambda_{1} \). (d) Second-order moment with data-dependent \( \pi_{uc}^{y} \) and \( \lambda_{1} \).
experiments are shown in Fig. 3 for, c-contamination, b=8, and uncertainty class size |P| = 250 for y = 0.1 (more results, including those for p-point uncertainty, are shown in Section 2 of the supplementary materials on the companion website). In the figure, the Bayes error corresponding to the optimal classifier is denoted as \( \epsilon_{\text{Bayes}} \).

We use the algorithm proposed in [48] for generating the contaminating distribution generated uniformly under a unit-simplex. Fig. 3(a) shows the results for the first scenario. Three cases are considered for the pair: \((\epsilon_0,\epsilon_1)\): (0.3,0.9), (0.4,0.6), and (0.1,0.95). The expected true error for the proposed scheme is smaller than that of the histogram rule in all cases. Moreover, the results from the Monte-Carlo simulations are very close to those obtained from our approximations in [15], shown by “Approx” in the legends of plots. The expected true error for the case (0.4,0.6) is significantly smaller than the others for small sample sizes. This is due to the reliable prior knowledge compared to other cases, for small samples. However, when the sample size increases, (0.1,0.95) outperforms (0.4,0.6). Owing to a small contamination degree \( \epsilon_0 \) in (0.1,0.95), the proposed RML framework provides a good estimate of \( \pi_{y^*} \) for any sample size. Furthermore, by increasing the sample size, we achieve a better estimate of \( \pi_{y^*} \), making the designed classifier perform close to the optimal classifier. Therefore, it outperforms (0.4,0.6), which has less accurate estimates of the two conditional distributions for these sample sizes.

Fig. 3(b) shows the results for the data-dependent regularization parameter, where one can see that our approximation and the Monte-Carlo simulations are slightly different. This happens only for small sample sizes, owing to have a poor estimate of \( \lambda_y \), \( y = 0.1 \), defined in (29) and (30). Fig. 3(c) and (d) corresponds to Fig. 3(a) and (b) for the second-order moment, respectively.

6. Performance assessment using networks containing NF-κB pathways

While the theoretical development of the paper pertains to uncertainty classes of distributions for classification, as stated at the outset, our original motivation for the theory comes from our desire to apply prior pathway knowledge in biological network steady-state classification.

In this section, we use prior pathway knowledge and an associated cellular context in order to improve the performance of a classifier which discriminates between biologically relevant states of a biological system. More specifically, a biological system can be modeled by a discrete, dynamical system that is subject to external stimuli and behaves according to interactions amongst its constituent components. These interactions between components are often referred to as pathways and are time invariant in most biological processes. It is instead the varying cellular context that activates or deactivates pathways in order for a cell to respond to the demands of life. For many classification problems of interest and this example here, these pathways will be identical in each class and it is the cellular context of available nutrients, signaling proteins, or other agents that are of interest. However, the general method can be used with differing pathways if the goal is to discriminate against such things as the presence of mutations, separate organisms, or cancer. In all of these examples, we would expect the two classes to have different pathways through differing genetics.

To set up the classification example, we use a single set of pathways describing our biological system of interest, and choose two different cellular contexts which describe the biological phenomena we are interested in classifying. Then for each (context, pathways) tuple we generate an intermediate class of dynamical systems that have behavior described by the biological pathways under this context. These classes represent all possible dynamical systems that can behave according to the constraints of the pathways and cellular context. Each dynamical system in these two classes possesses a unique steady-state distribution, and we can therefore obtain two classes of steady state distributions from our two tuples of (context, pathways).

6.1. The NF-κB system

Nuclear factor-κB (NF-κB) is a family of transcription factors that control the expression of over 100 genes. Its primary role is in the immune system as a central regulator of inflammation. This makes it important in cancer research as inflammation contributes to the reduction of apoptosis and increased angiogenesis in the tumor microenvironment [49].

Biologically the NF-κB transcription factor can be activated through several parallel signaling pathways. In this paper we use a model containing three stimulating external inputs which are shaded in Fig. 4. When a bacterial infection occurs, the lipopolysaccharide (LPS) molecule present in the cell wall of the bacteria binds to TLR4 receptors in immune cell membranes and initiates a strong NF-κB response [50]. Tumor necrosis factor \( \alpha \) (TNF \( \alpha \)) is a cytokine produced primarily by macrophages to induce an endogenous inflammatory response by binding to the TNFR receptor. And finally, NF-κB responses can be initiated through the ‘alternative pathway’ with the lymphotoxin \( \beta \) receptor (LT\( \beta \)R). Once activated, each of these inputs initiates a downstream signaling cascade activating the NF-κB system. As there is no feedback from the system back onto these three external

![Fig. 4.](image-url)
signaling molecules, their states are constant once chosen and help to determine the behavior of the other nine genes.

6.2 NF-κB classification

In a biological system, we are often unable to directly measure or quantify the cellular context which controls the behavior of some cells of interest. We consider such a scenario as a classification problem. Given two possible cellular contexts and some data samples of the nine proteins whose behaviors are constrained by the context, determine which context the samples were taken from. In Fig. 5 we graphically depict the two contexts (or classes) in the three such classification problems (or configurations). The presence of an input indicates activation, absence indicates inactivation, and a shaded input indicates the input may either be active or inactive.

Qualitatively the three configurations in Fig. 5 can be described in the following manner: configuration 1 considers an endogenous macrophage induced inflammatory insult in class 0 versus inflammation as a result of bacteria and the response of immune cells in class 1 [50]. Configuration 2 considers an inflammatory insult resulting from bacteria and immune cells in class 0 versus an endogenous inflammatory insult arising from many types of immune cells signaling in class 1. Configuration 3 compares inflammation resulting from a bacterial infection (either in the early stage with no immune cells present or late stage after immune cells have arrived) in class 0 versus an inflammatory injury with immune cells present (possibly resulting from a bacterial infection in class 1).

In these three configurations we measure the ability of the classifier to distinguish the underlying context for an inflammatory injury with immune cells present (possibly resulting from a bacterial infection in class 1).

In the NF-κB model, there are only three uncertainties that arise from the pathways. These determine the parameterization of the uncertainty class via the vector \( \theta \in [0,1]^6 \). Choosing \( \theta \) gives a single well-defined Markov chain from the uncertainty class. For a small example see the companion website (Section 3 of the Supplementary materials) and for more details we refer to [12,51]. For the true network, we choose a network from [12]. It is at the center of the parameter space, \( \theta_{\text{nc}} = (0.5,0.5,0.5) \). From the standpoint of classification this network is unknown; it is chosen here to generate samples. A priori we only know that the true network exists inside our uncertainty class.

6.4 Results

To utilize this modeling technique with the proposed RML framework we define two uncertainty classes of models for each configuration by fixing the inputs according to Fig. 5. Since the RML framework requires finite uncertainty classes, we discretize the continuous [0,1] space as explained in the companion website. Then, adding a perturbation probability \( p = 10^{-3} \) in our simulations to each network, we obtain a class of ergodic irreducible Markov chains and, accordingly, a class of steady-state distributions [47]. The perturbation probability for the true model is set to \( p = 10^{-5} \). We generate data from the true network in each class. These two datasets along with the two uncertainty classes allow us to compare the RML classification framework with the classical histogram rule. Fig. 6 shows the results for the histogram-rule and proposed method for different configurations. In configuration 3, the error of the classifier briefly increases as a function of the sample size at the beginning. The regularization parameter is set according to Corollary 1, denoted by \( \lambda_{\text{SURE}} \). Both the histogram and RML classifiers converge to the Bayes errors as \( n \to \infty \). In all cases, the RML approach outperforms the histogram-rule, illustrating the benefit of prior knowledge, if available.

6.4.1 Comparison to MAP

Designing the RML classifier begins with the assumption of having finite uncertainty classes of feature distributions, in the absence of a prior distribution governing these classes, i.e., no prioritization of any uncertainty class member in favor of the others. Nonetheless, one would still solve the maximum posteriori (MAP) to find the most likely multimodal distribution existing in the uncertainty class and build the “plug-in rule” classifier according to Eq. (1). Hence, using the log-likelihood function in Eq. (2), we define the MAP

![Fig. 6. Performance comparison between the Histogram-rule and the RML framework. The x-axis shows the number of samples n, with n = n_0 + n_1, n_0 = n_1. We have \( \epsilon_{\text{Bayes}} = 0.193 \), \( \epsilon_{\text{Bayes}} = 0.299 \), and \( \epsilon_{\text{Bayes}} = 0.371 \) for configurations 1, 2, and 3, respectively.](image-url)
distribution as

\[
\hat{\pi}_{\text{MAP}}^y = \arg \max_{\pi \in \Pi^y} \sum_{k=1}^b u_k^y \log \pi^y(k).
\]  

(37)

Thereafter, we define the MAP classifier by plugging the estimates \(\hat{\pi}_{\text{MAP}}^y\) in Eq. (1). In Fig. 7, we compare performance of the RML given in Lemma 1 with that of MAP given in Eq. (37) by plotting the difference between the corresponding expected true errors, i.e., \(E_{\text{RML}}[\text{error}]-E_{\text{MAP}}[\text{error}]\) as a function of sample size for the three configurations considered in Fig. 6. Fig. 7 illustrates that for configurations 1 and 2 the RML classifier performs always better than the MAP. For category 3, the MAP classifier performs better than the RML in some range, but then, the RML classifier outperforms the MAP after increasing the sample size.

7. Conclusion

We have proposed a novel classifier design paradigm that allows us to design enhanced classifiers by incorporating available prior knowledge of the process generating the observation data. As shown in our simulations, such knowledge can significantly improve the performance of the designed classifier, especially, when the sample size is small. Having laid the theoretical groundwork for enhancing steady-state classifier design via the use of prior process knowledge, our plan is to apply the methodology to develop better biomedical classifiers in the presence of partial knowledge of the underlying genetic regulatory network. More generally, given the ubiquity of large feature sets and relatively small sample sizes now common in many disciplines, including medicine, material science, environmental science, and transportation, there will no doubt be an increasing number of methods proposed for using prior knowledge in classifier design. We believe it is important to provide analytic performance characterization of the classifiers on standard models, as we have done in this paper, so that their behaviors can be understood.

Appendix A. Proof of Lemma 1

Plugging (5) in (4), we obtain

\[
\hat{\pi}_{\text{RML}}^y = \arg \min \left(1-\lambda_y\right) \sum_{k=1}^b u_k^y \log \pi^y(k) + \frac{\lambda_y}{n} \sum_{k=1}^b \left| \pi^0(k) \log \frac{\pi^0(k)}{\pi^y(k)} \right|
\]

\[
= \arg \min \left(1-\lambda_y\right) \sum_{k=1}^b u_k^y \log \pi^y(k) + \frac{\lambda_y}{n} \sum_{k=1}^b \left| \pi^0(k) \log \frac{\pi^0(k)}{\pi^y(k)} \right|
\]

The solution to this problem can be obtained using a Lagrangian multiplier similar to (2), which leads to the label conditional probabilities in (6).

Appendix B. Proof of Theorems 1 and 2

In this appendix, we prove Theorems 1 and 2 for \(y=0\). The case \(y=1\) can be handled similarly. Let the inner expectation in (13), \(E_{\text{E}}[\Pr(\psi_n,\Pi^y,\Pi^0(X) \neq Y|S_n)]\), be denoted by \(\text{EX}_1\). Then

\[
\text{EX}_1 = E_{\text{E}} \left[ \sum_{k=1}^b \Pr(X=k|Y=0)\Pi_{\psi_n,\Pi^y,\Pi^0} = 1 + \Pr(X=k|Y=1)\Pi_{\psi_n,\Pi^y,\Pi^0} = 0 \right]
\]

\[
= \tilde{c}_0 \sum_{k=1}^b \pi^0(k) \Pr(\lambda_1^0 \mu_{10}^0 + \lambda_1^0 \pi^0(k) \geq \lambda_0 (1-\lambda_0) \mu_{10}^0 + \lambda_0 \pi^0(k))
\]

\[
+ \tilde{c}_1 \sum_{k=1}^b \pi^0(k) \Pr(\lambda_1^0 \mu_{10}^0 + \lambda_0 \pi^0(k) \geq \lambda_1 (1-\lambda_1) \mu_{10}^0 + \lambda_1 \pi^0(k))
\]

\[
= \tilde{c}_0 \sum_{k=1}^b \pi^0(k) \Pr(\lambda_1^0 \mu_{10}^0 + \lambda_1^0 \pi^0(k) \geq \lambda_0 (1-\lambda_0) \mu_{10}^0 + \lambda_0 \pi^0(k))
\]

\[
+ \tilde{c}_1 \sum_{k=1}^b \pi^0(k) \Pr(\lambda_1^0 \mu_{10}^0 + \lambda_0 \pi^0(k) \geq \lambda_1 (1-\lambda_1) \mu_{10}^0 + \lambda_1 \pi^0(k))
\]

in which we apply \(\tilde{c}_0 = n_y/n; y=0\). We denote the average distribution by \(\pi_{\psi,\Pi^y,\Pi^0} = \pi\), where \(\pi\) is the average of contaminating distributions. Now, for \(y=0\), define

\[
g_y := (1-\lambda_y) \mu_{10}^0 (1-\lambda_{1-y} - \lambda_1-y)
\]

\[
\tilde{c}_y := \frac{g_y}{1-\lambda_y}
\]

\[
p_y(k) := \tilde{c}_y \pi_y(k)
\]

Eq. (B.1) can be written as

\[
\text{EX}_1 = \sum_{k=1}^b \left[ \Pr(X=k|Y=0)\tilde{c}_0 \Pr(g_1 \mu_{10}^0 + p_1(k) \geq g_0 \mu_{10}^0 + p_0(k))
\]

\[
+ \Pr(X=k|Y=1)\tilde{c}_1 \Pr(g_0 \mu_{10}^0 + p_0(k) > g_1 \mu_{10}^0 + p_1(k)) \right]
\]

\[
= \sum_{k=1}^b \pi^0(k) \left[ \sum_{k=0}^{b} \left( \sum_{m=0}^{n_1} \left( \pi^0_{m,k} \right) \left( \mu_{10}^0 \right)^m \left( \lambda_1 \right)^{n_1-m} n_0 \right) \right]
\]

\[
\times \left( \frac{n_0}{m} \right)
\]

\[
+ \sum_{k=1}^b \pi^0(k) \left[ \sum_{k=0}^{b} \left( \sum_{m=0}^{n_1} \left( \pi^0_{m,k} \right) \left( \mu_{10}^0 \right)^m \left( \lambda_1 \right)^{n_1-m} n_0 \right) \right]
\]

\[
\times \left( \frac{n_0}{m} \right)
\]

\[
\times \left( \frac{n_0}{m} \right)
\]

where

\[
\tilde{c}_0 = \max \left\{ \frac{g_0 + p_0(k) - p_1(k)}{g_1} + 1 \right\}
\]

Conflict of interest statement

None declared.
\[ \varepsilon^{-1} \sum_{k=0}^{1} = \max \left\{ 0, \left[ \frac{g_1 \varepsilon_1 + p_1(k) - p_0(k)}{g_0} \right] + 1 \right\}. \tag{B.5} \]

In (B.4), we have two random variables \( \sum_{k=0}^{1} \) and \( \sum_{k=1}^{2} \), depending on the uncertainty classes \( \Pi^0 \) and \( \Pi^1 \), respectively. We present the distributions of these random variables for the uncertainty class models described in Section 2 in the following subsections:

### B.1. \( \varepsilon \)-contamination class

We first show that the contaminating part \( \pi(k) \) in (9) has a Beta distribution \( B(1, b-1) \), where \( b \) is the number of states. Suppose that the contaminating distribution comes from a uniform distribution on a \( (b-1) \)-simplex. Thus, as \( \Delta x \to 0 \),

\[ \Pr(\pi x < \Delta x) < x + \Delta x / 2. \tag{B.8} \]

where \( \text{Vol}(\cdot) \) denotes volume under the specified argument and \( S_{b-1} \) and \( S_{b-2} \) are the unit \( (b-1) \)-simplex and \( (b-2) \)-simplex with corners on \( 1-x \), respectively. (B.6) can be written as a density function according to

\[ f_{\pi|(k)}(x) = (b-1)(1-x)^{b-2}, \quad x \in (0, 1) \tag{B.7} \]

which is a Beta distribution with parameters \( 1 \) and \( b-1 \) whose mean and variance are \( 1/b \) and \( (b-1)/(b^2(b+1)) \), respectively. Using the Edgeworth expansion to approximate the cumulative density function of \( \pi(k) \), we obtain

\[ \Pr(\pi(k) < x) = P_{\varepsilon}(x) + R_{\varepsilon} \tag{B.9} \]

where \( z = \sqrt{\Pi^0 / (x-1/b) / \sqrt{(b-1)/b^2(b+1)}} \), and we have

\[ R_{\varepsilon}(\Pi^0) = \lim_{n \to \infty} \sum_{k=0}^{n} \exp(c \Pi^0 k) \times 0 \geq 0. \tag{B.10} \]

In (B.9), according to the Edgeworth expansion, we have

\[ R_{\varepsilon}(\Pi^0) = O((\Pi^0)^{\varepsilon / 2 - 1}). \tag{B.10} \]

Considering (B.9) and (B.10), one can conclude that \( R_{\varepsilon}(\Pi^0) \to 0 \) for large enough uncertainty classes. Therefore, for large uncertainty classes, we will have \( \pi(k) \) approximately have \( (\Pi^0 - 1/b) / \sqrt{(b-1)/b^2(b+1)} \to N(0, 1/\Pi^0) \). Hence, considering the last equation of (Eq. B.2), we get the following result:

\[ p_0(k) \sim N \left( \frac{\varepsilon_0}{B_0}, \frac{\varepsilon_1^2}{b^2(b+1)} \right) \tag{B.11} \]

\[ p_1(k) \sim N \left( \frac{\varepsilon_1}{B_1}, \frac{\varepsilon_1^2}{b^2(b+1)} \right) \tag{B.11} \]

Thus, since \( p_0(k) \) and \( p_1(k) \) are independent random variables, we get

\[ \sum_{k=0}^{1} p_0(k) - p_1(k) \sim N(\mu_{\varepsilon_1}^0, \sigma_{\varepsilon_1}^2) \tag{17} \]

where \( \mu_{\varepsilon_1}^0, \sigma_{\varepsilon_1}^2 \) are defined in (17). It is now straightforward to find the distribution of \( \sum_{k=0}^{1} \) (and similarly \( \sum_{k=1}^{2} \)) using Eq. (B.5).

### B.2. \( \mu \)-point class

From the mapping defined in (12), we know that state \( k \) belongs to \( p_0^{\mu_k} \) and \( p_1^{\mu_k} \) under labels zero and one, respectively.

Considering class \( \Pi^0 \), similar to (B.6), one can show that

\[ p_{\mu_k}(k) = \frac{\varepsilon_0^{\mu_k} - 1}{\varepsilon_0^{\mu_k}} \left( \frac{1 - x}{\varepsilon_0^{\mu_k}} \right)^{\varepsilon_0^{\mu_k} - 1}, \quad x \in (0, \varepsilon_0^{\mu_k}) \tag{B.12} \]

which is equivalent to the random variable \( \omega_{\mu_k} \sim \text{Beta}(1, \varepsilon_0^{\mu_k} - 1) \). Therefore, similar to (B.11), we obtain

\[ p_0(k) \sim N \left( \frac{\varepsilon_0^{\mu_k}}{2}, \frac{\varepsilon_1^2}{b^2(b+1)} \right) \tag{B.13} \]

from which we obtain \( g_{\varepsilon_0^0} + p_0(k) - p_1(k)/g_1 \sim N(\mu_{\varepsilon_1}^0, \sigma_{\varepsilon_1}^2) \), whereas

\[ \mu_{\varepsilon_1}^0 = \frac{g_0 + 1}{g_1} \tag{17} \]

\[ \sigma_{\varepsilon_1}^2 = \frac{1}{g_1} \tag{17} \]

Now, one can find the distribution of \( \sum_{k=0}^{1} \) according to (B.5). The distribution of \( \sum_{k=1}^{2} \) can be found similarly. Afterwards, we obtain Eq. (15).

### Appendix C. Proof of Theorem 3

The second-order moment of the true error of the RML classifier can be written as

\[ E(\varepsilon_{\text{RML}}) = E_{\Pi^0} \text{Pr}(\psi_{\varepsilon,\Pi^0}(X) \neq \varepsilon | S_0)^2 \Pi^0, \Pi^1 \). \tag{C.1} \]

For simplicity, we drop the subscript of \( \psi_{\varepsilon,\Pi^0}(X) \), noting that the classifier depends on \( S_0 \) and \( \Pi^0, \Pi^1 \). The proof has two parts shown in two appendices. First, we take the expectation with respect to the training data, \( S_0 \). Later, we will see that the dependency of the second-order moment on the uncertainty classes manifests itself in the indices of the dependencies (found from combinatorial parts). In the next section, then find the distribution of those indices, knowing that the randomness comes from the uncertainty classes. Let us start the proof by expanding Eq. (C.1).

\[ E(\varepsilon_{\text{RML}}) = E_{\Pi^0} \text{Pr}(\psi_{\varepsilon,\Pi^0}(X) \neq \varepsilon | S_0)^2 \Pi^0, \Pi^1 \). \tag{C.1} \]

\[ + \varepsilon_2^2 \sum_{k=1}^{1} \text{Pr}(\psi_{\varepsilon,\Pi^0}(X) \neq \varepsilon | S_0)^2 \Pi^0, \Pi^1 \). \tag{C.2} \]
In (C.2), parts $A_0$ and $A_1$ can be found similarly as in Appendix 1. In the following, whenever we sum over $t_1', t_2', y \in \{0,1\}$ we implicitly consider $t_1', t_2' \geq 0$ and $t_1' + t_2' \leq n_y$. Furthermore, for any pair of $(t_1', t_2') \geq 0$ with $t_1' + t_2' \leq n_y$, we have

$$Pr(u_{k_0} = t_1', u_{k_0} = t_2') = Pr(trin(n_y, \pi_{k_0}^0, (k_1, n_{k_0}^0, (k_2))) = (t_1', t_2')).$$

Hence, for the $B^1$, we may write

$$B^1 = E_{\psi \mid \psi(X = k_1) = 1 \mid \psi(X = k_2) = 0} = Pr(\psi(X = k_1) = 1, \psi(X = k_2) = 1)$$

$$= \sum_{i_1, i_2} Pr(g_1 u_{k_1} + p_1(k_1) \geq g_0 u_{k_0} + p_0(k_1), g_1 u_{k_2} + p_1(k_2) < g_0 u_{k_0} + p_0(k_2))$$

$$= \sum_{i_1, i_2} Pr(g_1 u_{k_1} + p_1(k_1) \geq g_0 u_{k_0} + p_0(k_1), g_1 u_{k_2} + p_1(k_2) < g_0 u_{k_0} + p_0(k_2))$$

$$\times Pr(u_{k_0} = t_1', u_{k_0} = t_2') = \sum_{i_1, i_2} Pr(u_{k_1} = t_1', u_{k_2} = t_2') Pr(u_{k_0} = t_1', u_{k_0} = t_2')$$

$$= \sum_{i_1, i_2} \left[ \sum_{i_1, i_2} Pr(u_{k_1} = t_1', u_{k_2} = t_2') \right]$$

(C.3)

Similarly, we can get

$$B^0 = \left[ \sum_{i_1, i_2} \sum_{i_1, i_2} Pr(u_{k_1} = t_1', u_{k_2} = t_2') Pr(u_{k_0} = t_1', u_{k_0} = t_2') \right].$$

(C.4)

Next, we can obtain $C^1$

$$C^1 = E_{\psi \mid \psi(X = k_1) = 1 \mid \psi(X = k_2) = 0} = Pr(\psi(X = k_1) = 1, \psi(X = k_2) = 1)$$

$$= \sum_{i_1, i_2} Pr(g_1 u_{k_1} + p_1(k_1) \geq g_0 u_{k_0} + p_0(k_1), g_1 u_{k_2} + p_1(k_2) < g_0 u_{k_0} + p_0(k_2))$$

$$= \sum_{i_1, i_2} Pr(g_1 u_{k_1} + p_1(k_1) \geq g_0 u_{k_0} + p_0(k_1), g_1 u_{k_2} + p_1(k_2) < g_0 u_{k_0} + p_0(k_2))$$

$$\times Pr(u_{k_0} = t_1', u_{k_0} = t_2') = \sum_{i_1, i_2} Pr(u_{k_1} = t_1', u_{k_2} = t_2') Pr(u_{k_0} = t_1', u_{k_0} = t_2')$$

$$= \sum_{i_1, i_2} \left[ \sum_{i_1, i_2} Pr(u_{k_1} = t_1', u_{k_2} = t_2') \right]$$

(C.5)

Similarly, we obtain

$$C^0 = \sum_{i_1, i_2} \left[ \sum_{i_1, i_2} \sum_{i_1, i_2} Pr(u_{k_0} = t_1', u_{k_0} = t_2') \right] Pr(u_{k_1} = t_1', u_{k_2} = t_2')$$

(C.6)

In (C.5) and (C.6), we have

$$w^0_{k_0, l_1} = \min \left\{ \frac{g_0 l_0 + p_0(k) - p_1(k)}{g_1} - 1, l_1 \right\}$$

$$w^1_{k_1, l_1} = \min \left\{ \frac{g_1 l_1 + p_1(k) - p_0(k)}{g_0} - 1, n_0 \right\}. \quad (C.7)$$

In order to take the last expectation in (C.2) with respect to the uncertainty classes, we need to find the joint distribution of $\pi^0_{k_0, l_1}$ and $\pi^1_{k_1, l_1}$ (similarly for $\pi^0_{k_1, l_1}$ and $\pi^1_{k_1, l_1}$), and the joint distribution between $\pi^{0}_{k_0, l_1}$ and $\pi^{0}_{k_1, l_1}$ (similarly for $\pi^{1}_{k_0, l_1}$ and $\pi^{1}_{k_1, l_1}$). These distributions are found in Appendix D.

### Appendix D. Joint distributions

To find the joint distribution of $\pi^0_{k_1, l_1}$ and $\pi^0_{k_1, l_1}$, we need to approximate the joint distribution of $(p_0(k_1), p_0(k_2))$ defined in Eq. (B.2). We do this by a (zero-order) Edgeworth expansion. Thus, similar to the single variate case in (B.11), for the multivariate case we have $(p_0(k_1), p_0(k_2)) \sim N(\mu_{k_1, k_2}, \Sigma_{k_1, k_2})$, whereas we find the parameters for different uncertainty classes in the following subsections.

#### D.1. $\omega$-contamination class

From the definition of the joint probability distribution, for $x_1, x_2 > 0, x_1 + x_2 \leq 1$, we have

$$Pr(\pi(k_1) = x_1, \pi(k_2) = x_2)$$

$$= \lim_{\Delta x_1 \to 0, \Delta x_2 \to 0} Pr \left( \frac{\pi(k_1) - x_1}{\Delta x_1}, \frac{\pi(k_2) - x_2}{\Delta x_2} < \frac{\Delta x_1}{2} \right)$$

$$= \lim_{\Delta x_1 \to 0, \Delta x_2 \to 0} \frac{Vol(S^{(b-3)}_{-}\pi(k_1), \pi(k_2))}{\Delta x_1 \Delta x_2}$$

$$= \frac{(b-1)(b-2)(1-x_1-x_2)^{b-3}}{b^5}.$$  

(D.1)

Since we are going to use the zero-order Edgeworth expansion, we only need to find the mean vector and the covariance matrix of these random variables. The variances are already found in the previous section of the Appendix. Therefore, we only find the covariance between these variables. Specifically,

$$Cov(\pi(k_1), \pi(k_2)) = E[\pi(k_1) \pi(k_2)] - E[\pi(k_1)]E[\pi(k_2)]$$

$$= \int_0^1 \int_0^{1-x_1} x_1 x_2 (b-1)(b-2)(1-x_1-x_2)^{b-3} d x_2 d x_1 - 1$$

$$= \frac{-1}{b^5} (b+1),$$

(D.2)

where in (D.2) we used integration by parts. Hence, considering our definitions in (B.2) for $p_0(k_1)$ and $p_0(k_2)$, we obtain the following for the normal distribution statistics:

$$\mu_{k_1, k_2}^0 = \begin{bmatrix} x_0 (\frac{2}{b} + (1-x_0) \pi_{k_0}^0(k_1)) \\ x_0 (\frac{2}{b} + (1-x_0) \pi_{k_0}^0(k_2)) \end{bmatrix},$$

(D.3)

$$\Sigma_{k_1, k_2}^0 = \begin{bmatrix} \frac{2}{b^2} \frac{1}{b+1} \frac{1}{b+1} \pi_{k_0}^0(k_1) & \frac{2}{b^2} \frac{1}{b+1} \frac{1}{b+1} \pi_{k_0}^0(k_2) \\ \frac{2}{b^2} \frac{1}{b+1} \frac{1}{b+1} \pi_{k_0}^0(k_1) & \frac{2}{b^2} \frac{1}{b+1} \frac{1}{b+1} \pi_{k_0}^0(k_2) \end{bmatrix}.$$  

(D.4)

Similarly, we can write for the joint distribution of $(p_1(k_1), p_1(k_2))$.

#### D.2. $p$-point class

Since we have partitions in this model, we need to know whether two states belong to the same partition or not. First, suppose that $p^0(k_1) \neq p^0(k_2)$. Then,

$$Pr(\pi(k_1) = x_1, \pi(k_2) = x_2) = Pr(\pi(k_1) = x_1) Pr(\pi(k_2) = x_2),$$

(D.5)

from which we get

$$\mu_{k_1, k_2}^0 = \begin{bmatrix} \frac{\mu_{p_0(k_1)}^0}{p_0^0(k_1)} \\ \frac{\mu_{p_0(k_2)}^0}{p_0^0(k_2)} \end{bmatrix}.$$  

(D.6)
Table D1
Defined parameters.

\[
\begin{align*}
\theta_{l,t}^0 &= g_l(m_{1,t} - g_0 t) \\
\epsilon_{l,t} &= g_l(m_{1,t} + g_0 t) \\
\eta_{l,t}^0 &= g_l(m_{1,t} - g_0 t) \\
\eta_{l,t} &= g_l(m_{1,t} + g_0 t) \\
\eta_{l,k}^0 &= g_l(m_{1,k} - g_0 t) \\
\eta_{l,k} &= g_l(m_{1,k} + g_0 t) \\
\eta_{l,k}^0 &= g_l(m_{1,k} - g_0 t) \\
\eta_{l,k} &= g_l(m_{1,k} + g_0 t) \\
\end{align*}
\]

Now, suppose \( P^0(k_1) = P^0(k_2) = m_{k_1,k_2} \). Then

\[
\text{Cov}[^{10}(\pi(k_1),\pi(k_2))] = \frac{-(\omega_{m_{k_1,k_2}})^2}{|\omega_{m_{k_1,k_2}}|^2 (|\omega_{m_{k_1,k_2}}| + 1)^2}, \tag{D.8}
\]

and we have

\[
\begin{align*}
\mu_{k_1,k_2}^0 &= \left[ \begin{array}{c}
\frac{\omega_{m_{k_1,k_2}}}{\omega_{m_{k_1,k_2}}^2} \\
\frac{\omega_{m_{k_1,k_2}}}{\omega_{m_{k_1,k_2}}^2}
\end{array} \right] \\
\end{align*}
\]

In the following, \( P_{l,k}(k_1) = x, p_{l,k}(k_2) = y \) and \( P_{l,k}(k_1) = x, p_{l,k}(k_2) = y \) will be denoted by \( F_{l,k}(x,y) \) and \( F_{l,k}(x,y) \), respectively. Now, we start by computing the pmf of \((k_1,t,k_2,t_2)\). After quite some computation we obtain

\[
\begin{align*}
\text{Pr}[^{01}(k_1,t_1) = m_{1,k_1,t_1}] &= m_{1,k_1,t_1} \\
\text{Pr}[^{10}(k_1,t) = m_{1,k_1,t}] &= m_{1,k_1,t} \\
\end{align*}
\]

Furthermore, we have

\[
\text{Pr}[^{00}(k_1) = m_{1,k_1}] = m_{1,k_1} \\
\text{Pr}[^{11}(k_1) = m_{1,k_1}] = m_{1,k_1}
\]

Table D1 shows the parameters used in Eqs. (D.11)–(D.14).

Appendix E. Proof of Lemma 2

Although we took a standard approach to find the unbiased estimator of the MSE, in this part, for simplicity, we only show that \( \text{E}(\text{MSE}) = \text{MSE} \) (it is sufficient for the proof), where MSE can be expanded as follows:

\[
\text{MSE} = E \left( \sum_{k=1}^{b} [\pi_j(k) - \pi_{\text{ac}}(k)]^2 \right) = \sum_{k=1}^{b} E[\pi_j^2(k) + \pi_{\text{ac}}^2(k) - 2\pi_j(k)\pi_{\text{ac}}(k)]
\]

The first and the second terms in the right summation do not need any manipulation. Therefore, in the remainder of the proof, we focus on the last term in the right summation. Using the definitions of \( \delta_j \) and \( \theta_j(k) \), and the fact that \( \text{E}(u_j) = n\pi_{\text{ac}}(k) \), we have

\[
E \left( \sum_{k=1}^{b} \frac{\pi_j(k)\pi_{\text{ac}}(k)}{n} \right) = \sum_{k=1}^{b} (\delta_j n\pi_{\text{ac}}(k) + \theta_j(k))\pi_{\text{ac}}(k)
\]

Now, we return to the MSE in Lemma 2 and take the expectation of the last term in the summation (the term multiplied by 2).
We obtain

\[ E \left[ \sum_{i=1}^{n} \frac{\hat{p}_i}{n} - \frac{1}{n} \sum_{i=1}^{n} \hat{p}_i \right] ^2 = \sum_{i=1}^{n} \left( \frac{\hat{p}_i}{n} - \frac{1}{n} \right)^2 = \sum_{i=1}^{n} \frac{b_i}{n} - \frac{1}{n} \sum_{i=1}^{n} \frac{b_i}{n} \]

(E.1)

in which we used the terms for the first and second moments of the multinomial distribution. Some simplification completes the proof.

Appendix F. Supplementary data

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.patcog.2013.02.017.

References


[23] G. McLachlan, Asymptotic results for discriminant analysis when the initial samples are misclassified, Technometrics 14 (1972) 415–422.


M. Shahrokh Esfahani et al. / Pattern Recognition 46 (2013) 2783–2797
Amin Zollanvari received Ph.D. in electrical and computer engineering from Texas A&M University, College Station, TX, in 2010. He held a postdoctoral position in Harvard Medical School and Brigham and Women’s Hospital, Boston MA, from 2010 to 2012. He is currently an assistant research scientist in Electrical and Computer Engineering Department in Texas A&M University, College Station, TX. His research interests include statistical pattern recognition and genomic signal processing.

Byung-Jun Yoon received the B.S.E. (summa cum laude) degree from Seoul National University (SNU), Seoul, Korea, in 1998, and the M.S. and Ph.D. degrees from California Institute of Technology (Caltech), Pasadena, CA, in 2002 and 2007, respectively, all in electrical engineering. In 2008, he joined the Department of Electrical and Computer Engineering at Texas A&M University, College Station, TX, where he is currently an assistant professor. His main research interests include genomic signal processing, bioinformatics, and computational biology.

Edward R. Dougherty is a Professor in the Department of Electrical and Computer Engineering at Texas A&M University in College Station, TX, where he holds the Robert M. Kennedy’26 Chair in Electrical Engineering and is Director of the Genomic Signal Processing Laboratory. He is also co-Director of the Computational Biology Division of the Translational Genomics Research Institute in Phoenix, AZ. He holds a Ph.D. in mathematics from Rutgers University and has been awarded the Doctor Honoris Causa by the Tampere University of Technology in Finland. He is a fellow of both IEEE and SPIE, and has received the SPIE President’s Award.