Multiple Kernel Learning for Efficient Conformal Predictions

Vineeth Balasubramanian, Shayok Chakraborty, Sethuraman Panchanathan
Center for Cognitive Ubiquitous Computing (CUbiC)
School of Computing, Informatics and Decision Systems Engineering
Arizona State University
Tempe AZ 85287
vineeth.nb@asu.edu, schakr10@asu.edu, panch@asu.edu

Abstract

The Conformal Predictions framework is a recent development in machine learning to associate reliable measures of confidence with results in classification and regression. This framework is founded on the principles of algorithmic randomness (Kolmogorov complexity), transductive inference and hypothesis testing. While the formulation of the framework guarantees validity, the efficiency of the framework depends greatly on the choice of the classifier and appropriate kernel functions or parameters. While this framework has extensive potential to be useful in several applications, the lack of efficiency can limit its usability. In this paper, we propose a novel Multiple Kernel Learning (MKL) methodology to maximize efficiency in the CP framework. This method is validated using the k-Nearest Neighbors classifier on a cardiac patient dataset, and our results show promise in using MKL to obtain efficient conformal predictors that can be practically useful.

1 Introduction

Reliable estimation of confidence remains a significant challenge as learning algorithms proliferate into challenging real-world pattern classification applications. In the last few years, Vovk, Shafer and Gammerman [1] have proposed a game-theoretic approach to confidence estimation called the Conformal Predictions (CP) framework, which has several desirable properties for potential use in various real-world applications. The theory of Conformal Predictions is based on the principles of algorithmic randomness, transductive inference and hypothesis testing. This theory is based on the relationship derived between transductive inference and the Kolmogorov complexity of an i.i.d. (identically independently distributed) sequence of data instances. One of the desirable features of this framework is the calibration of the obtained confidence values in an online setting. While probability/confidence values generated by the traditional aforementioned approaches can often be unreliable and difficult to interpret, the theory behind the CP framework guarantees that the confidence values obtained using this transductive inference framework manifest as the actual error frequencies in the online setting i.e. they are well-calibrated [2]. Further, this framework can be applied across all existing classification and regression methods, thus making it a very generalizable approach. The only assumption of the framework is that data should be i.i.d. or even just exchangeable, which can be obtained by permuting the data.

The CP framework is relatively recent, and more details of this framework, including the algorithm for Conformal Predictors in classification, are presented in an appendix for lack of space within the main body of this paper. We now describe the motivation and contributions of this work [1].

[1] As required by the workshop submission guidelines, we would like to mention that this work has also been accepted at the International Conference on Machine Learning Applications (December 2010), and is being submitted to this workshop since this work is of interest to this audience

-

2 Multiple Kernel Learning for Efficient Conformal Predictors

Without any loss in generality, we describe this work assuming a binary classification problem for the sake of simplicity and understanding. The CP framework [1] defines a *non-conformity measure* that quantifies the conformity of a data point to a particular class label for a given classifier. This non-conformity measure can be appropriately designed for any classifier under consideration.The non-conformity measure of a data point x_i for a k-Nearest Neighbor classifier is defined as:

$$\alpha_i^y = \frac{\sum_{j=1}^{k} D_{ij}^y}{\sum_{j=1}^{k} D_{ij}^{-y}} \tag{1}$$

where D_i^y denotes the list of sorted distances between a particular data point x_i and other data points with the same class label, say y. D_i^{-y} denotes the list of sorted distances between x_i and data points with any class label other than y. D_{ij}^y is the jth shortest distance in the list of sorted distances, D_i^y. In short, α_i^y measures the distance of the k nearest neighbors belonging to the class label y, against the k nearest neighbors from data points with other class labels. Given a new test data point, say x_{n+1}, a null hypothesis is assumed that x_{n+1} belongs to the class label, say, y_p. The non-conformity measures of all the data points in the system so far are re-computed assuming the null hypothesis is true. A p-value function is defined as:

$$p(\alpha_{n+1}^{y_p}) = \frac{count\left\{i : \alpha_i^{y_p} \geq \alpha_{n+1}^{y_p}\right\}}{m+1} \tag{2}$$

where $\alpha_{n+1}^{y_p}$ is the non-conformity measure of x_{n+1}, assuming it is assigned the class label y_p, and m is the total number of data instances. It is evident that the p-value is highest when all non-conformity measures of training data belonging to class y_p are higher than that of the new test point, x_{n+1}, which points out that x_{n+1} is *most conformal* to the class y_p. This process is repeated with the null hypothesis supporting each of the class labels, and a set of p-values corresponding to all the class labels is obtained. The output of the CP framework in a classification problem is a set of class labels based on a user-defined confidence level. The prediction region is given by Γ_ϵ, which contains all the class labels with a p-value greater than $1 - \epsilon$.

In a binary classification problem, the output set predicted by the CP framework can contain zero, one or two (both) class labels. The performance of the CP framework is measured using two properties (as stated in [1]): *validity* and *efficiency*. These properties are captured by the following measures on a test set: (i) *number of errors*, when the output contains only one class label which is however incorrect; and (ii) *number of multiple predictions*, when the output set contains both class labels. Since the CP framework guarantees validity [2], the number of errors will always remain bounded by $1 - \epsilon$ (as illustrated in Figure 2 in Appendix). However, the efficiency of the framework lies in providing the maximum possible one-label prediction sets (at a given confidence level), since output sets with both labels in a binary classification problem do not provide any useful information to the end user. The efficiency can vary depending on the choice of a classifier, its parameters or kernel functions. This limitation can be a serious deterrent in real-world applications, since it may not be an easy task to identify the correct parameters for a practically useful conformal predictor. This motivates the need for a methodology that can minimize the number of multiple predictions, thus maximizing efficiency (while maintaining validity), given a particular classifier in the CP framework. This is the objective of this work.

From the definition of a non-conformity measure for the k-NN classifier (Equation 1), it is evident that we need to learn a kernel function, ϕ, such that for the projected data, $\phi(x)$: (i) the margin between the classes is maximized, and (ii) the variance inside each of the classes is minimized. Such a kernel feature space will ensure that the numerator of Equation 1 is low and the denominator is high, for a data point which is assigned the correct class label (and otherwise for an incorrect class label). The first criterion - maximizing the margin - can be achieved using a Support Vector Machine(SVM)-based approach to kernel learning, as used in earlier work [3, 4]. Similarly, the second criterion - minimizing intra-class variance - can be achieved by using a Linear Discriminant Analysis (LDA) approach, i.e. by minimizing the denominator of the Fisher discriminant criterion, $w^T S_w w$, where S_w is the within-scatter matrix. Hence, the combination of these two criteria will lead to a kernel function that can generate more efficient conformal predictors.

-

Combining the maximum-margin and minimum-variance criteria, we can define an objective function as follows (we assume a hard-margin formulation for convenience of explanation):

$$\min \frac{1}{2} \|w\|^2 + w^T S_w w = \frac{1}{2} w^T w + w^T S_w w = \frac{1}{2} w^T (I + 2S_w) w$$

More generally, this problem can be written as:

$$\min \frac{1}{2} w^T (\lambda S_w + I) w$$

subject to $y_i(w^T x_i + b) \geq 1 \; \forall i = 1, 2, \ldots, n$, and where $S_w = \sum_i \sum_x (x - m_i)^2$, the within-scatter matrix in Discriminant Analysis. λ is a parameter that can be set empirically to balance the SVM and LDA components of the objective function. Now, substituting $\Lambda = \lambda S_w + I$, we get:

$$\min \frac{1}{2} w^T \Lambda w$$

subject to $y_i(w^T x_i + b) \geq 1 \; \forall i = 1, 2, \ldots, n$. Applying KKT conditions, the dual problem is written as:

$$\max L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j x_i^T \Lambda^{-1} x_j \tag{3}$$

subject to $\sum_{i=1}^{n} \alpha_i y_i = 0, \alpha_i \geq 0 \; \forall i = 1, 2, \ldots, n$. This formulation can be shown to be equivalent to the following SVM formulation [5]:

$$\min \frac{1}{2} \|\hat{w}\|^2$$

such that $y_i(\hat{w}^T \hat{x}_i + b) \geq 1 \; \forall i = 1, 2, \ldots, n$ where $\hat{w} = \Lambda^{1/2} w$ and $\hat{x}_i = \Lambda^{-1/2} x_i \; \forall i = 1, 2, \ldots, n$. Evidently, this is the standard SVM formulation on the projected data points $\hat{x}_i$, and can be solved using existing SVM solving software. Xiong *et al.* [5] provided a method to compute $\Lambda^{1/2}$ and $\Lambda^{-1/2}$ using Singular Value Decomposition, which we have adopted in this work. Hence, the dual problem (Equation 3) to be solved can be rewritten as:

$$\max L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y_i y_j \alpha_i \alpha_j \hat{x}_i^T \hat{x}_j \tag{4}$$

subject to $\sum_{i=1}^{n} \alpha_i y_i = 0, \alpha_i \geq 0 \; \forall i = 1, 2, \ldots, n$. The optimization formulation to maximize efficiency in k-NN conformal predictors has thus been shown to be equivalent to a standard SVM problem with the projected data points.

Similar to Lanckriet's formulation [4], Equation 4 can be rewritten as an MKL problem:

$$\min_{p \in P} \max_{\alpha \in Q} \alpha^T e - \frac{1}{2} (\alpha \circ y)^T (\sum_{i=1}^{m} p_i K_i)(\alpha \circ y)$$

where $P = \left\{ p \in \mathbb{R}^m : p^T e = 1, 0 \leq p \leq 1 \right\}$ denotes the set of kernel weights, $Q = \left\{ \alpha \in \mathbb{R}^n : \alpha^T y = 0, \alpha \geq 0 \right\}$ is the set of SVM dual variables, e is a vector all ones, $\{K_i\}, i = 1, 2, \ldots, m$ is a group of base kernel matrices that are defined on the projected data, $\hat{x}_i$, and $\circ$ denotes the vector dot product. Several ways of solving this optimization problem have been proposed in earlier work such as Semi-Definite Programming, Quadratically Constrained Quadratic Programming, Semi-Infinite Linear Programming and Subgradient Descent. In this work, we adopted the more recent Extended Level Set method proposed by Xu *et al.* [3] which was shown to be more efficient than other optimization methods for MKL. More details of this method can be found in [3].

3 Results and Conclusions

To study the performance and generalizability of the proposed method, we carried out experiments on three binary datasets (with different number of dimensions and instances): 2 datasets from the UCI Machine Learning repository, and a challenging cardiac patient dataset. We focused on datasets from the healthcare domain, since reliable confidence measures are extremely valuable for machine learning algorithms to be successfully applied in this domain. However, for lack of space, we present the results obtained on the real-world cardiac decision support problem in this paper. Data consisted of 2312 patient cases who had a Drug Eluting Stent procedure performed during the period 2003-07, and who followed up with the cardiac facility during the 12 months following the procedure. The

-
complications considered for this model included: Stent Thrombosis and Restenosis, which manifest as chest pain, myocardial infarction and sometimes even death. The objective of using this dataset was to build a classification model which can predict the risk of complications that a cardiac patient may face in the first one year following a stent procedure. This is treated as a binary classification problem which predicts the onset of complications, or otherwise. 75% of each of the datasets was randomly permuted (to meet the exchangeability requirements of the CP framework) and used for training, while the remaining portion of the dataset was used for testing. Further, the experiment was repeated 5 different times to remove any randomness bias. Please refer to [6] for more details on the dataset, patient attributes and other relevant details.

By definition of the CP framework, the validity property is always satisfied, i.e. the number of errors are always bounded by the confidence threshold. Hence, we focus on studying the results related to the efficiency of the CP framework. The proposed MKL approach was compared against the plain k-NN classifier (with different values of k) and kernel k-NN classifier with varying kernel functions and parameters. The results are presented in Table 1. Note that the best representative results were selected and presented for each of the considered classifiers. The number of multiple predictions is the number of test data points that the k-NN conformal predictor provided both class labels as an output. *A lower number of multiple predictions at all possible confidence levels is most desirable*.

Classifier	Parameters	Number of Multiple Predictions at Confidence Level (Total: 578 test points)				
		70%	80%	90%	95%	99%
k-NN	k=3	0	0	1	181	522
k-NN	k=15	0	0	0	193	500
kernel k-NN	k=3, Gaussian kernel, Spread=10	0	0	1	184	512
kernel k-NN	k=3, Polynomial kernel, Degree=3	0	0	2	176	517
Proposed MKL	k=3,5,10; Mixture of Polynomial kernels	0	0	0	141	461
Proposed MKL	k=3,5,10; Mixture of Gaussian kernels	0	0	0	137	470

Table 1: Results obtained on the Cardiac Patient dataset. Note that the number of multiple predictions are clearly the least when using the proposed MKL approach, even at high confidence levels.

The results obtained with the proposed MKL approach for efficiency maximization are significantly better than the best possible results obtained with the other classifiers (which were obtained after long trials of varying parameter values). Similar results were also observed with other datasets from the UCI Machine Learning repository. It can be observed that the number of multiple predictions are very low at lower confidence levels. This is because the CP framework allows a higher number of errors at lower confidence levels, thereby reducing the number of multiple predictions. Hence, it is rather most desirable to obtain low number of multiple predictions at very high confidence levels. Note that when the number of multiple predictions is high, the classifier is providing results with both class labels, thereby serving no purpose to the physician in prognosing or diagnosing the patient. The proposed approach reduces this number significantly to provide more useful results to the end user. While we validated our approach using k-NN, this methodology can be adapted to any other classifier, depending on the definition of the non-conformity measure for the classifier.

References

[1] V. Vovk, A. Gammerman, and G. Shafer, *Algorithmic Learning in a Random World*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005.

[2] V. Vovk, "On-line confidence machines are well-calibrated," in *43rd Symposium on Foundations of Computer Science*, Washington, DC, USA, 2002, pp. 187–196.

[3] Z. Xu, R. Jin, I. King, and M. Lyu, "An extended level method for efficient multiple kernel learning," in *Neural Information Processing Systems (NIPS)*, 2009.

[4] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan, "Learning the kernel matrix with semidefinite programming," *J. Mach. Learn. Res.*, pp. 27–72, 2004.

[5] T. Xiong and V. Cherkassky, "A combined SVM and LDA approach for classification," in *Neural Networks, 2005. IJCNN '05. Proceedings. 2005 IEEE International Joint Conference on*, vol. 3, 2005, pp. 1455–1459.

[6] R. Gouripeddi, V. Balasubramanian, S. Panchanathan, J. Harris, A. Bhaskaran, and R. Siegel, "Predicting risk of complications following a drug eluting stent procedure: A SVM approach for imbalanced data," in *22nd IEEE International Symposium on Computer-Based Medical Systems*, 2009, pp. 1–7.

-

4 Appendix

4.1 Background: Theory of Conformal Predictions

Consider the set of labeled data instances to be represented as the sequence $Z = ((x_1, y_1), (x_2, y_2), ..., (x_n, y_n))$, where x_i is a data instance, and y_i is the corresponding class label. If $l(Z)$ is the length of this sequence, and $C(Z)$ is its Kolmogorov complexity (the length of the minimal description of Z using a universal description language), then:

$$\delta(Z) = l(Z) - C(Z) \tag{5}$$

where $\delta(Z)$ is called the *randomness deficiency* of the sequence Z. Intuitively, Equation 5 states that lower the value of $\delta(Z)$, the higher is the randomness of the sequence. As a corollary, if there was a new data instance x_{n+1}, and we were to predict its label based on the available labeled data Z, the confidence in the prediction would be low, if the sequence Z was highly random i.e. $\delta(Z)$ was low.

Evidently, the challenge is the computation of the randomness deficiency, $\delta(Z)$, of a given sequence Z. This is achieved using the Martin-Lof test for randomness, which can be summarized as a function $t : Z^* \rightarrow \mathbb{N}$ (the set of natural numbers with 0 and ∞), such that $\forall n \in \mathbb{N}, m \in \mathbb{N}, P \in P_n$:

$$P\{z \in Z^n : t(z) \geq m\} \leq 2^{-m} \tag{6}$$

where P_n is the set of computable probability distributions. Equation 6 can also be written as:

$$P\{z \in Z^n : t(z) \in [m, \infty)\} \leq 2^{-m} \tag{7}$$

Now, if we use the transformation $f(x) = 2^{-x}$, Equation 7 can in turn be written in terms of a new function $t'(z)$:

$$P\left\{z \in Z^n : t'(z) \in \left(0, 2^{-m}\right]\right\} \leq 2^{-m} \tag{8}$$

Hence, a function $t' : Z^* \rightarrow (0, 2^{-m}]$ is a Martin-Lof test for randomness if $\forall m, n \in \mathbb{N}$, the following holds true:

$$P\left\{z \in Z^n : t'(z) \leq 2^{-m}\right\} \leq 2^{-m} \tag{9}$$

If 2^{-m} is substituted for a constant, say r, and r is restricted to the interval $[0, 1]$, Equation 9 is equivalent to the definition of a p-value typically used in statistics for hypothesis testing. Given a null hypothesis H_0 and a test statistic, p-value is defined as the probability of obtaining a result at least as extreme as the one that was actually observed, assuming that the null hypothesis is true. In other words, the p-value provides a measure of the extent to which the observed data supports or disproves the null hypothesis.

To translate this theory to pattern classification problems, Vovk *et al.* [1] defined the concept of a *non-conformity measure* for a given classifier, as a measure that quantifies the conformity of a data point to a particular class label. This non-conformity measure can be appropriately designed for any classifier under consideration, thereby allowing the concept to be generalized to different kinds of pattern classification problems. To illustrate this idea, the non-conformity measure of a data point x_i for a k-Nearest Neighbor classifier is defined as:

$$\alpha_i^y = \frac{\sum_{j=1}^{k} D_{ij}^y}{\sum_{j=1}^{k} D_{ij}^{-y}} \tag{10}$$

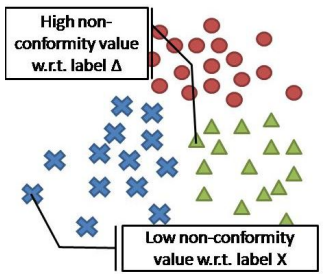

Figure 1: An illustration of the non-conformity measure defined for k-NN

where D_i^y denotes the list of sorted distances between a particular data point x_i and other data points with the same class label, say y. D_i^{-y} denotes the list of sorted distances between x_i and data points with any class label other than y. D_{ij}^y is the jth shortest distance in the list of sorted distances, D_i^y. In short, α_i^y measures the distance of the k nearest neighbors belonging to the class label y, against the k nearest neighbors from data points with other class labels (Figure 1). Note that the higher the value of α_i^y, the more non-conformal the data point is with respect to the current class label i.e. the probability of it belonging to other classes is high.

Given a new test data point, say x_{n+1}, a null hypothesis is assumed that x_{n+1} belongs to the class label, say, y_p. The non-conformity measures of all the data points in the system so far are recomputed assuming the null hypothesis is true. A p-value function (which satisfies the Martin-Lof

test definition in Equation 9) is defined as:

$$p(\alpha_{n+1}^{y_p}) = \frac{count\left\{i : \alpha_i^{y_p} \geq \alpha_{n+1}^{y_p}\right\}}{m+1} \tag{11}$$

where $\alpha_{n+1}^{y_p}$ is the non-conformity measure of x_{n+1}, assuming it is assigned the class label y_p, and m is the total number of data instances. In simple terms, Equation 11 states that the p-value of a data instance belonging to a particular label is the normalized count of the data instances that have a higher non-conformity score than the current data instance, x_{n+1}. It is evident that the p-value is highest when all non-conformity measures of training data belonging to class y_p are higher than that of the new test point, x_{n+1}, which points out that x_{n+1} is *most conformal* to the class y_p. This process is repeated with the null hypothesis supporting each of the class labels, and a set of p-values corresponding to all the class labels is obtained. As mentioned earlier, these p-values satisfy the modified Martin-Lof test in Equation 9.

The output of the CP framework in a classification problem is a set of class labels (or a region/interval for regression) based on a user-defined confidence level. The prediction region is given by Γ_ϵ, which contains all the class labels with a p-value greater than $1-\epsilon$. These prediction regions, Γ_ϵ, are *conformal* i.e. the confidence threshold, $1-\epsilon$ directly translates to the frequency of errors, ϵ in the online setting [2]. For more details about this framework and on how non-conformity measures can be defined for other classifiers, please refer to [1].

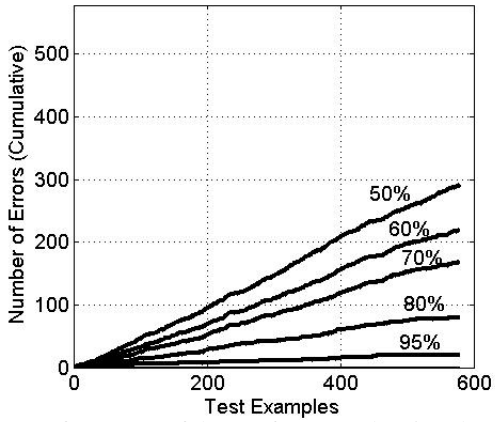

Figure 2: Illustration of the performance of the CP framework using the cardiac patient dataset. Note the validity of the framework, i.e. the errors are calibrated in each of the specified confidence levels. For example, at a 80% confidence level, the number of errors will always be lesser than 20% of the total number of test examples.

Algorithm 1 Conformal Predictors for Classification

Require: Training set $T = \{(x_1, y_1), ..., (x_n, y_n)\}$, $x_i \in X$, number of classes M, $y_i \in Y = y_1, y_2, \ldots, y_M$, classifier Ξ

1: Get new unlabeled example x_{n+1}.
2: **for** all class labels, y_i, where $i = 1, \ldots, M$ **do**
3: Assign label y_i to x_{n+1}.
4: Update the classifier Ξ, with $T \cup \{x_{n+1}, y_i\}$.
5: Compute non-conformity measure value, $\alpha_{n+1}^{y_i}$ to compute the p-value, P_i, w.r.t. class y_i (Equation 2) using the conformal predictions framework.
6: **end for**
7: Output the conformal prediction regions $\Gamma_{1-\epsilon} = \{y_i : P^{y_i} > \epsilon, y_i \in Y\}$, where $1-\epsilon$ is the confidence level.