2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Domain Adaptation Bounds for Multiple Expert Systems Under Concept Drift

Gregory Ditzler, Gail Rosen, and Robi Polikar

***Abstract*—The ability to learn incrementally from streaming data – either in an online or batch setting – is of crucial importance for a prediction algorithm to learn from environments that generate vast amounts of data, where it is impractical or simply unfeasible to store all historical data. On the other hand, learning from streaming data becomes increasingly difficult when the probability distribution generating the data stream evolves over time, which renders the classification model generated from previously seen data suboptimal or potentially useless. Ensemble systems that employ multiple classifiers may be used to mitigate this effect, but even in such cases some classifiers (experts) become less knowledgeable for predicting on different domains than others as the distribution drifts. Further complication results when labeled data from a prediction (target) domain is not immediately available; hence, causing prediction on the target domain to yield sub-optimal results. In this work, we provide upper bounds on the loss, which hold with high probability, of a multiple expert system trained in such a nonstationary environment with verification latency. Furthermore, we show why a single model selection strategy can lead to undesirable results when learning in such nonstationary streaming settings. We present our analytical results with experiments on simulated as well as real-world data sets, comparing several different ensemble approaches to a single model.**

I. Introduction & Learning Setting

Traditional learning algorithms often assume that the training data are sampled from the same probability distribution as test data; however, this stationarity assumption is violated in many practical settings [1], [2]. Concept drift is one such learning scenario where the training and testing probability distributions may be different. Concept drift is commonly found in many prediction scenarios such as spam filtering, electricity price forecasting, and financial forecasting. The problem of concept drift in data streams, also referred to as learning in nonstationary environments, has attracted much attention from the computational intelligence community due to increasing number of applications that generate such data, and multiple expert systems (MES) have been shown to be quite effective in such nonstationary settings [3]–[6]. MES, also called ensemble systems, are widely used in concept drift settings because of their inherent ability the provide a good balance for the *stability-plasticity dilemma* [7]: they are able to learn to forget old or irrelevant knowledge, and learn new relevant knowledge with the removal and addition of classifiers, respectively [3].

G. Ditzler and G. Rosen are with the Dept. of Electrical & Computer Engineering at Drexel University, Philadelphia, PA. They are supported by the National Science Foundation (NSF) CAREER award #0845827, NSF Award #1120622, and the Department of Energy Award #SC004335. Author email: gregory.ditzler@gmail.com, gailr@ece.drexel.edu

R. Polikar is with the Dept. of Electrical & Computer Engineering at Rowan University. He is supported by the NSF under Grant No ECCS-1310496. Author email: polikar@rowan.edu

Algorithms that address concept drift tend to be either *passive* or *active*[1]. Passive algorithms (e.g., Learn^{++}.NSE [9], and Learn^{++}.CDS [10]) simply assume that the environment is changing and are continuously taking some corrective action to adjust the prediction strategy. Active algorithms (e.g., Oza Bagging + `ADWIN` [11]) only make adjustments when drift is detected. Therefore, active approaches typically require a drift detection algorithm such as the Hellinger Distance Drift Detection Method (HDDDM) [12], or the Intersection of Confidence Intervals (ICI) rule [13], to be used in conjunction with a classifier. Some concept drift approaches try to use both passive and active techniques to aid learning in nonstationary environments [14].

One of the central issues we are concerned with when learning from data over time is the expected loss of a classifier on a target data set. Unfortunately, computing an exact quantity is typically infeasible, either because there is not enough data, or the labels are simply not available. Therefore, we must look at deriving bounds on the loss of a hypothesis that hold with probability $1-\delta$ [15], where $\delta \in (0,1)$. Such bounds can help us better understand, for example, how weights in a multiple expert system should be determined for efficient learning in nonstationary environments. A common weight for a classifier in an ensemble system is typically of the form:

$$w_k = \log \frac{1-\epsilon_k}{\epsilon_k} \tag{1}$$

where ϵ_k is some appropriate measure of loss for the kth classifier.

Our learning setting is as follows (summarized in Figure 1, with common mathematical notations described in Table I): data are sampled at each discrete-time t from a drifting probability distribution $\mathcal{D}_t$ to form a batch, $\mathcal{S}_t$. Given $\mathcal{S}_t$, an expert h_t (i.e., classifier) is learned from the data. We do not assume that the expert h_t is updated with future data when it becomes available. The classifier (i.e., form of the hypothesis) belongs to a hypothesis class $\mathcal{H}$ (e.g., $\mathcal{H}$ can be the set of all linear functions). Each expert has a weight $w_{k,t}$ (for expert k at time t) that is used to form a composite hypothesis, which is a convex combination of the individual expert hypotheses. In this work, we focus on binary prediction problems, although our results can easily be interpreted past this assumption. Experts in the ensemble make predictions on unlabeled data collected from $\mathcal{D}_{t+1}$, which is the target distribution, with $\mathcal{D}_{t+1} \neq \mathcal{D}_t$. At a later point in time, when labeled information about $\mathcal{D}_{t+1}$ becomes available, the expected loss, $\mathbb{E}_{t+1}[\ell(H, f_{t+1})]$ is measured, where ℓ is a convex loss computed on the

[1]Not to be confused with active learning. See [8].

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

Input Data sets $\mathcal{S}_t$ sampled from $\mathcal{D}_t$
for $t = 1, 2, \ldots$
1) Learn h_t from $\mathcal{S}_t$
2) Update weights $w_{k,t}$, where $k \in [t]$
3) Predict on unlabeled data sampled from $\mathcal{D}_{t+1}$
4) Receive labels for data in step (3) and measure the expected loss $\mathbb{E}_{t+1}[\ell(H, f_{t+1})]$

Fig. 1. MES algorithm for processing data incrementally in batches.

composite hypothesis H over the distribution at time $t + 1$, and f_{t+1} is the optimal labeling function for $\mathcal{D}_{t+1}$.

At time t the ensemble contains t different experts that can be used to predict on the target distribution, whose data labels are unknown at the time of prediction. One of the core problems with prediction in nonstationary environments is that t different training distributions could have been used to produce a classifier, any or all of which can be different than the target distribution. Therefore, an algorithm designer needs to determine if it is better to use a single expert, or a convex combination of all experts' decisions. One of the advantages that we can expect to gain from the ensemble is that the ensemble's decision can lead to a reduction in the error variance [16]–[19], while potentially providing a slightly larger bias than a single (best) model. The analysis of the bias/variance dilemma has been well studied in the setting of learning stationary problems (i.e., training/testing distributions are the same). In this contribution, we show that the selection of a single best expert can be problematic for classification of nonstationary data streams as well, which can be partially attributed to the larger error variance compared to that of an MES.

In our previous work, we demonstrated that MES, using a weighted majority vote, tended to provide a lower upper bound on a convex loss compared to the uniform weighting method [20]. Furthermore, the follow-the-leader approach, where only the classifier with the current best performance is used for prediction, was shown as reliable only when there is a small bias between the most recent labeled distribution and the target probability distribution (i.e., when $\mathcal{D}_t$ is very close to $\mathcal{D}_{t+1}$). This result was demonstrated using a simulation of a generic loss bound and empirically confirmed on several real-world data streams. In this work, we develop a more specific loss bound than the one used in [20] using mathematical tools similar to those used for domain adaptation.

The rest of this manuscript is organized as follows: Section II presents related work on domain adaptation as well as MES for handling concept drift, Section III presents our theoretical analysis of learning using MES in the presence of concept drift, Section IV evaluates several MES approaches on real-world data streams, and Section V draws conclusions and future work.

TABLE I
MATHEMATICAL NOTATIONS

symbol	meaning
t	unit of time
k	indices for experts through time t, $k \in \{1, \ldots, t\} := [t]$
h_t	classifier created at time t
f_t	true labeling function at time t
$w_{k,t}$	weight of the kth expert at time t
$\mathcal{H}$	hypothesis class
$\mathcal{D}_t$	probability distribution at time t from which data are sampled
$\mathcal{S}_t$	collection of data $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$ sampled from $\mathcal{D}_t$
m	cardinality of the data set $\mathcal{S}_t$
$\mathbb{E}_k[\cdot]$	statistical expectation on $\mathcal{D}_k$
$\mathbb{V}[\cdot]$	statistical variance
$\ell(\cdot, \cdot)$	loss function that is convex in the first argument

II. Related Work

A. Motivation & Background

There is certainly no shortage of multiple expert system implementations for learning from data streams experiencing concept drift (e.g., SEA [21], DWM [14], Learn^{++}.CDS [10], & Learn^{++}.NSE [9]). Data stream algorithms either implement an *online* or *incremental* learning strategy. In the online setting, one data instance is provided to the learning algorithm at a time. Typically, it is assumed that the class label is revealed to the algorithm only after a prediction has been made. Incremental learning processes a batch of data at each time stamp, and it is assumed that data are not retained after they have been used for learning. This work focuses on the incremental learning setting. However, recent work by Brzezinski and Stefanowski have experimented with the conversion of batch (i.e., incremental) MES approaches to online MES approaches [22].

Recall from our learning setting (Figure 1) that a sequence of classifiers $(h_1, \ldots, h_t)$ are learned from potentially different data distributions. The composite hypothesis, H, is a linear combination of the individual classifiers, and the classifiers may be making predictions on out-of-domain data, where the training and testing distributions may be different, which is similar to a domain adaptation setting. Therefore, it seems intuitive to examine how techniques from domain adaption can help us better understand prediction in such nonstationary learning settings.

B. Domain Adaptation & Incremental Learning

Ben-David et al. presented a comprehensive methodology for analyzing a classifier (hypothesis) learned across multiple domains [23]. In their work, they derived loss bounds for a single hypothesis trained on a source domain distribution ($\mathcal{D}_S$), and tested on a target domain distribution ($\mathcal{D}_T$) where $\mathcal{D}_S \neq \mathcal{D}_T$. In their analysis, they examine a hypothesis that was developed to minimize the loss across multiple domains as well as the simpler scenario where a hypothesis is learned on a single source and tested on a target domain. Their work did not consider the situation with multiple hypotheses trained across all domains, but the mathematical framework they developed easily lends itself to concept draft analysis. However, one such

upper bound they provide, as shown by (2), constitutes the starting point of our analysis described in this manuscript.

In our preliminary work, we examined the effect of different weighting mechanisms (e.g., uniform, weight majority, etc.) with a simple loss bound. However, the general bound we examined in [20] is not very informative if we are interested in understanding the roles that various types of concept drift play in forming an upper bound. For example, Žliobaitė derived a loss function for a simple linear classifier under sudden concept drift in [24]. However, their work does not provide bounds with high probability and the error equations only apply to linear Euclidean classifier under a pre-specified concept drift scenario. In this work, we do not make any explicit assumptions about the type of drift that occurs, nor do we limit the selection of classifier used in the MES.

We now briefly review relevant definitions and theorems from existing literature (primarily from [23] and [25]) that are used in our analysis in Section III.

Definition: The $\mathcal{H}\Delta\mathcal{H}$ distance, used in several recent works on domain adaptation and data stream analysis [23], [25], measures the maximum difference in expected loss between two hypotheses $h, h' \in \mathcal{H}$, when measured on distributions $\mathcal{D}_k$ and $\mathcal{D}_T$, where $\mathcal{D}_k$ indicates the distribution that generated the data at time k, and $\mathcal{D}_T$ is the target distribution. It is necessary to form a generalization of this distance that can be used with any loss function $\ell(\cdot,\cdot)$. The generalized $\mathcal{H}\Delta\mathcal{H}$ distance is given by

$$\begin{aligned} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_T,\mathcal{D}_k) &= 2\sup_{h,h'\in\mathcal{H}} |\mathbb{E}_T[\ell(h,h')] - \mathbb{E}_k[\ell(h,h')]| \\ &\geq 2|\mathbb{E}_T[\ell(h,h')] - \mathbb{E}_k[\ell(h,h')]| \end{aligned}$$

The quantity $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_T,\mathcal{D}_k)$ can be computed with unlabeled data $\mathcal{U}_T$ and $\mathcal{U}_k$, of size m sampled from $\mathcal{D}_T$ and $\mathcal{D}_k$, respectively, which can be used to establish the upper bound on $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_T,\mathcal{D}_k)$ as

$$d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_T,\mathcal{D}_k) \leq \hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_T,\mathcal{U}_k) + 8\sqrt{\frac{2\nu\log m + \log\frac{2}{\delta}}{m}}$$

which holds with probability $1-\delta$. Figure 2 shows the effect that the VC-confidence term, ν, plays in the upper bound for $\delta = 0.05$ when $\mathcal{H}$ is a class of linear functions. From this figure, we observe that – for the bounds to be meaningful – there needs to be a significant amount of data to force the VC-confidence term to be small.

Theorem (Expert Loss Bound) [23]: Let $\mathcal{H}$ be a hypothesis space of VC dimension ν. If $\mathcal{U}_S$ and $\mathcal{U}_T$ are unlabeled samples of cardinality m each, drawn from $\mathcal{D}_S$ and $\mathcal{D}_T$ respectively, then for any $\delta \in (0,1)$, with probability at least $1-\delta$, and for every $h \in \mathcal{H}$, the following inequality holds

$$\begin{aligned} \mathbb{E}_T\left[\ell(h,f_T)\right] \leq & \mathbb{E}_k\left[\ell(h,f)\right] + \lambda + \frac{1}{2}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_T,\mathcal{U}_S) \\ & + 4\sqrt{\frac{2\nu\log m + \log\frac{2}{\delta}}{m}} \end{aligned} \quad (2)$$

where λ is a measure of disagreement between f_S and f_T – more precisely, it is $\mathbb{E}_T[\ell(f_k,f_T)]$. The upper bound on an expert's loss on the target shows that the three primary terms are responsible for describing how an expert is expected to predict on the target distribution. The first term is the loss of the expert on the distribution it was trained on, which shows that if the expert had a high loss on the data it was trained on, we cannot expect it to perform well on a distribution that is different than the one used for training. The other two terms account for the differences in the data distributions (i.e., independent of what the labels of the data are), and the differences between the true labeling functions.

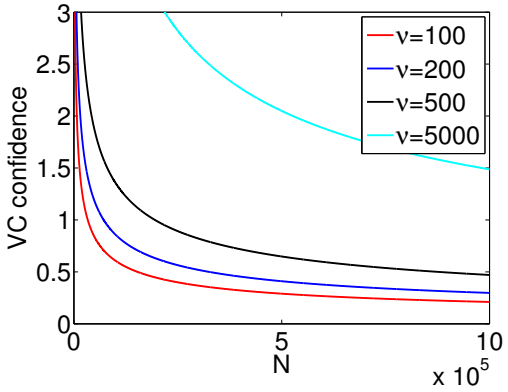


Fig. 2. VC confidence term for three linear classifiers with different VC-dimensions given by ν.

C. Concept Drift

One can view incremental learning with concept drift as a more elaborate, life-long, and time-dependent extension of domain adaptation. Recall that our objective is to minimize the loss on a distribution $\mathcal{D}_T$, when we are provided experts learned on $\mathcal{D}_1,\ldots,\mathcal{D}_t$. Traditionally, MES for concept drift only use information in $t \in [1, T-1]$ to modify parameters and make predictions provided by $\mathcal{D}_T$; however, this implementation ignores any knowledge on $\mathcal{D}_T$. Hence, more advanced methods are needed to handle this adaptation. In our previous work, we jointly used techniques from concept drift and semi-supervised/transductive learning to use $\mathcal{D}_T$ to adapt model parameters (such as expert weights) [26], [27]. More recently, Ruvolo and Eaton proposed ELLA, which uses methods from multi-task learning, to achieve what they refer to as *life-long learning* [28], [29].

III. Analysis

In this section, we first overview the prediction setting with multiple expert systems, and show a general bound on loss before describing our analysis for obtaining a more informative bound. In what follows, $\mathbf{x} \in \mathbb{R}^D$ is a data vector, $f_T(\mathbf{x})$ is a target labeling function (e.g., $f_T \in \{-1,+1\}$ for binary prediction problems), $\mathbb{E}_T$ denotes a statistical expectation over $\mathcal{D}_T$, and $h_t \in \mathcal{H}$ is an expert (or hypothesis) learned at time t from the hypothesis class $\mathcal{H}$. As shown in Figure 1, we assume a new hypothesis is generated on each data set $\mathcal{S}_t$.

A. An Indecisive Bound

Each expert in the ensemble is responsible for providing a prediction on an unlabeled vector $\mathbf{x}$. As mentioned previously, the kth expert has a weight at time t denoted by $w_{k,t}$ such that $\sum_k w_{k,t} = 1$. The composite hypothesis H is the ensemble decision given by:

$$H(\mathbf{x}) = \sum_{k=1}^{t} w_{k,t} h_k(\mathbf{x})$$

where $\mathbf{x}$ is sampled from $\mathcal{D}_t$. For brevity, we use $H(\mathbf{x}) = H$ and $h_k(\mathbf{x}) = h_k$ whenever the meaning is clear from the context. The loss $\ell(\cdot, \cdot)$ is a function that is: (a) a measure of the quality of the hypothesis, and (b) convex in the first argument. The loss of the composite hypothesis can be written as:

$$\ell(H, f_T) = \ell\left(\sum_{k=1}^{t} w_{k,t} h_k, f_T\right) \leq \sum_{k=1}^{t} w_{k,t} \ell(h_k, f_T)$$

where the last step is Jensen's inequality. This inequality shows that the loss of the MES can be upper bounded by a convex combination of the loss of each expert with respect to the target function f_T. Continuing the analysis, the statistical expectation of the inequality yields:

$$\begin{aligned} \mathbb{E}_T[\ell(H, f_T)] &\leq \mathbb{E}_T\left[\sum_{k=1}^{t} w_{k,t} \ell(h_k, f_T)\right] \\ &= \sum_{k=1}^{t} w_{k,t} \mathbb{E}_T[\ell(h_k, f_T)] \end{aligned} \tag{3}$$

which holds because of the linearity of expectations. This result should not come as a surprise, and is a well known starting point for analysis of MES. However, in our situation the above bound is too vague and uninformative. Specifically, the latter term, $\mathbb{E}_T[\ell(h_k, f_T)]$, is unfortunate for a couple of reasons: (i) the statistical expectation is computed over $\mathcal{D}_T$, which is generally not provided, and (ii) it is a function of f_T, which is not available. If f_T were known, we could use f_T and $\mathcal{D}_T$ jointly to form a new hypothesis, which would then make the problem easier to address, but much less interesting. In Section III-B we discuss how to resolve these concerns.

B. An Interpretable Bound

Our method in this section is an application of the work presented by Ben-David et al. for domain adaption [23]; however, we make a few generalizations. Recall that our goal is to decompose $\mathbb{E}_T[\ell(h_k, f_T)]$ into terms that can be estimated and more easily interpreted. Ben-David's theory of domain adaptation allows us to use their upper bound of an expert's loss in the concept drift setting.

Theorem (MES Loss Bound): The expected loss of a MES on the target distribution, $\mathcal{D}_T$, is bounded above with probability $1 - \delta$ by,

$$\begin{aligned} \mathbb{E}_T\left[\ell(H, f_T)\right] \leq \sum_{k=1}^{t} w_{k,t} \Bigg(& \mathbb{E}_k\left[\ell(h_k, f_k)\right] \\ & + \lambda_{T,k} + \frac{1}{2} \hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_T, \mathcal{U}_k) \\ & + 4\sqrt{\frac{2\nu \log m + \log \frac{2}{\delta}}{m}} \Bigg) \end{aligned} \tag{4}$$

where $\lambda_{T,k}$ is a measure of disagreement between f_k and f_T, $\mathcal{U}_T$ ($\mathcal{U}_k$) is an unlabeled data sample from the target (training) distribution, and $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_T, \mathcal{U}_k)$ is a measure of divergence between $\mathcal{D}_T$ and $\mathcal{D}_k$.

Proof: Using the result of (3), we follow the procedure described in [23] by applying the triangle inequality of loss, and adding/subtracting the expected loss of the kth classifier on $\mathcal{D}_k$.

$$\begin{aligned} \mathbb{E}_T[\ell(h_k, f_T)] &\leq \mathbb{E}_T[\ell(h_k, f_k)] + \mathbb{E}_T[\ell(f_T, f_k)] \\ &= \mathbb{E}_T[\ell(h_k, f_k)] + \mathbb{E}_T[\ell(f_T, f_k)] \\ &\quad + \mathbb{E}_k[\ell(h_k, f_k)] - \mathbb{E}_k[\ell(h_k, f_k)] \\ &\leq \mathbb{E}_k[\ell(h_k, f_k)] + \mathbb{E}_T[\ell(f_T, f_k)] \\ &\quad + |\mathbb{E}_T[\ell(h_k, f_k)] - \mathbb{E}_k[\ell(h_k, f_k)]| \\ &\leq \mathbb{E}_k[\ell(h_k, f_k)] + \lambda_{T,k} + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_T, \mathcal{D}_k) \end{aligned}$$

where $\lambda_{T,k} = \mathbb{E}_T[\ell(f_T, f_k)]$. Combining the above inequality with (3) gives (4). □

From this theorem, we see that the upper bound on the loss is comprised of three primary components. That is, the upper bound on the loss on expert k, namely $\mathbb{E}_T[\ell(h_k, f_T)]$, at time $t = T$ can be expressed as

$$\begin{aligned} \mathbb{E}_T[\ell(h_k, f_T)] \leq \text{training loss} + \text{disagreement of } f_k \text{ and } f_T \\ + \text{divergence of } \mathcal{D}_k \text{ and } \mathcal{D}_T \end{aligned} \tag{5}$$

This shows that the bounded loss of the MES is comprised of expert k's loss on its training data, the disagreement of the labeling functions f_T and f_k, and the divergence of the probability distributions $\mathcal{D}_T$ and $\mathcal{D}_k$. Writing the upper bound on loss as in (5) allows us to accommodate for all different types of drift that can occur. That is, the drift can be characterized by changes in $\mathcal{D}_T$ or $\mathcal{D}_k$ (*virtual drift*), or characterized by changes in f_T and f_k (*real drift*).

Theorem (Tightest Upper Bound on Expected Loss) The tightest bound for a MES is the one that sets $w_{tk} = 1$ for the kth predictor that has the smallest

$$\hat{\epsilon}_k = \mathbb{E}_k[\ell(h_k, f_k)] + \lambda_{T,k} + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_T, \mathcal{D}_k)$$

and all other weights set to zero adhering to a convex combination.

Proof: Let $[\hat{\epsilon}_{\min}, \hat{\epsilon}_{\max}] \subset \mathbb{R}_+$ form a convex set. To show this set is convex, let $\hat{\epsilon}_1, \hat{\epsilon}_2 \in [\hat{\epsilon}_{\min}, \hat{\epsilon}_{\max}]$ be two points in the interval over the convex set with extreme points $\hat{\epsilon}_{\min}$ and

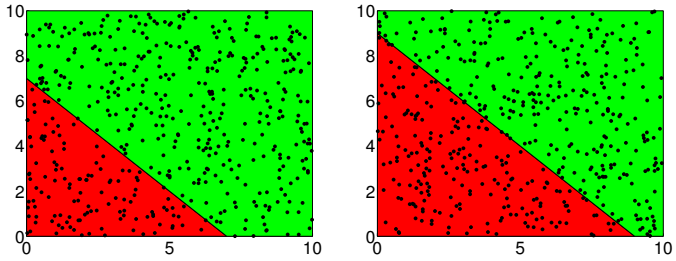


Fig. 3. Visualization of the SEA decision boundary for the first two shifts in the hyperplane. The term $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_1, \mathcal{U}_2)$ would be unaffected by the shift; however, $\lambda_{T,k}$ would experience a dramatic change due the change in the labeling functions.

$\hat{\epsilon}_{\max}$, where $\hat{\epsilon}_1 < \hat{\epsilon}_2$. Then for any $w \in [0, 1]$,

$$\begin{aligned}\hat{\epsilon}_{\min} \leq \hat{\epsilon}_1 &= (1-w)\hat{\epsilon}_1 + w\hat{\epsilon}_1 \\ &< (1-w)\hat{\epsilon}_1 + w\hat{\epsilon}_2 < (1-w)\hat{\epsilon}_2 + w\hat{\epsilon}_2 \\ &\leq \hat{\epsilon}_{\max}\end{aligned}$$

thus the minimum is attained thus when $w = 0$, and all other $w_{k,t} = 0$ for $k \in \{[t] \backslash 1\}$. □

While the latter theorem reveals a powerful insight about how the weights should be selected such that the tightest upper bound is obtained, it requires information that is not available at the time of prediction. Namely, the $\lambda_{T,k}$ term cannot be determined without some level of access to f_T, which, of course, is an unrealistic expectation. However, setting the weights of the follow–the–leader at time t does not guarantee that the non-zero weight corresponds to the smallest $\hat{\epsilon}_k$, because $\lambda_{T,k}$ and $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_T, \mathcal{D}_k)$ are not available at the time the weights are calculated.

SEA Problem: The SEA data stream is one of the most widely used synthetic benchmarks for concept drift algorithms. In this section, we discuss how the terms in the upper bound in (4) apply to this classical data stream problem. The SEA data set is characterized by long periods without change followed by abrupt changes in the decision boundary at three different distinct time stamps [21]. The data set also includes 10% class noise added into the labels (i.e., the class labels are randomly flipped to the other class for 10% of the data instances). In the SEA problem, the decision boundary undergoes an abrupt change between two consecutive time stamps as shown in Figure 3. The two features are uniform random variables in the $[0, 10]$ interval. The green and red regions – that change over time – indicate the regions in feature space that represent the two classes. The examination of the SEA data set with this bound is interesting because only two of the three terms in (4) that (approximately) contribute to the bound. The divergence of $\mathcal{D}_t$ and $\mathcal{D}_{t+1}$ does not apply since the distribution of $p(\mathbf{x})$ does not change over time. This is because the feature vectors are distributed as uniform random variables through all time stamps. The remaining terms, $\mathbb{E}_k[\ell(h_k, f_k)]$ and $\lambda_{T,k}$, both play a role in the upper bound. The classifier's loss at time k, denoted by $\mathbb{E}_k[\ell(h_k, f_k)]$, is expected to be nonzero due to the class noise added into the SEA data stream. The labeling function divergence term $\lambda_{T,k}$ is the area of disagreement between the two decision boundaries between the left and right graphics in Figure 3.

C. Empirical Analysis of the Variance in the Upper Bound on Expected Loss

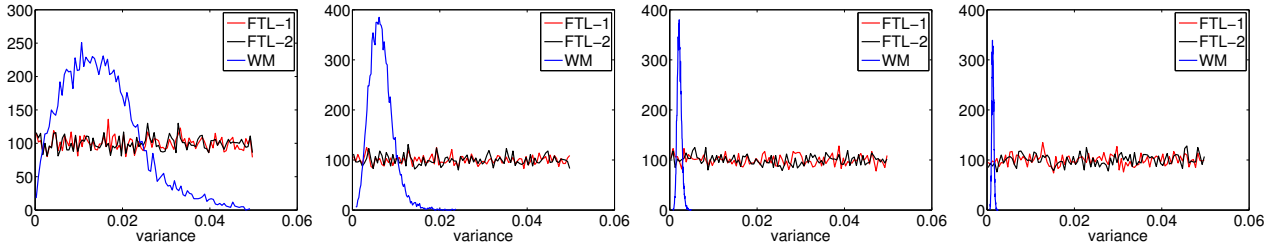
In the main body of the work, we were able to apply mathematical methods from domain adaptation, and convex sets to determine an upper bound on the expected loss (i.e., error) of an MES that has experts trained on different probability distributions. Furthermore, we also presented a theorem that demonstrates the ability of the follow–the–leader model to provide the tightest upper bound on the expected loss – so long as the leader is "informed" about f_T and f_k, which is generally not the case. However, if we are using n instances to compute the upper bound, we know that there will be an expected value of the bound as well as a variance to the bound's calculation. This point was demonstrated by the simulation of bound given by (3) [20]. The interpretation of the results in [20] goes along with our intuition about the bias/variance dilemma in machine learning, and we can further interpret them in this section.

Let us consider that $\mathbb{E}_k[\ell(h_k, f_k)]$ is computed from a data set of n instances and that $d_{\mathcal{H}\Delta\mathcal{H}}$ is computed using a finite sample as well. Then, the quantity $\hat{\epsilon}_k$ has an affiliated mean (i.e., the bias) and variance. The error variance of the upper bound becomes

$$\mathbb{V}\left[\sum_{k=1}^{t} w_{k,t}\hat{\epsilon}_k\right] = \sum_{k=1}^{t} w_{k,t}^2 \mathbb{V}[\hat{\epsilon}_k] \tag{6}$$

where $\sum w_{k,t} = 1$, $w_{k,t} \geq 0$, and $\mathbb{V}[X]$ denotes the variance of a random variable X. For the follow–the–leader, only one $w_{k,t}$ is non-zero (for a fixed t), which begs the question: how likely is it that $\sum_{k=1}^{t} w_{k,t}^2 \mathbb{V}[\hat{\epsilon}_k] < \mathbb{V}[\hat{\epsilon}_{k^*}]$ for k^* being the leader and $w_{k,t}$ are formed by a weighted majority vote, i.e., the error variance of the ensemble is less than that of the classifier with the lowest error. Note that we are not claiming $\sum_{k=1}^{t} w_{k,t}^2 \mathbb{V}[\hat{\epsilon}_k] < \mathbb{V}[\hat{\epsilon}_{k^*}]$ – this inequality does not hold uniformly. However, a legitimate question to ask is how probable is it that the inequality would hold. To address this question, we use a simulation. The simulation described below, suggests that the variance is indeed lower – on average – for the weighted majority algorithm over the follow–the–leader in a nonstationary setting, where the best expert cannot be precisely identified for the target distribution.

Let the error variance of the upper bound for each expert, $\mathbb{V}[\hat{\epsilon}_k]$, be a uniform random variable in the interval [0,0.05], and the weights, for weighted majority vote, are sampled from a probability distribution on the $[0, 1]$ interval. The weights are then normalized to assure they form a valid probability distribution. As is typically the case, indices in the weight vector with a large value represent experts with a small loss. Then $\mathbb{V}[\hat{\epsilon}_{k^*}]$ and $\sum_{k=1}^{t} w_{k,t}^2 \mathbb{V}[\hat{\epsilon}_k]$ are calculated. We use two methods to determine the "leader" for time $t + 1$ in this simulation: (i) uniformly choose an expert at random as the leader, or (ii) sample an index from the distribution of expert weights. The former case assumes that all experts are equally likely to be identified as being the best performing expert at



(a) $t = 2$ (b) $t = 5$ (c) $t = 15$ (d) $t = 25$

Fig. 4. Histogram of the error variance for the MES upper bound on loss for the follow-the-leader (FTL), and the weighted majority vote (WM). The number of experts (i.e., time stamps) is varied from 2 to 25. The variance of the weight majority vote produces a lower variance on the estimate of the target loss.

time $t + 1$, which implies that there is no prior knowledge being used to identify the best expert. The latter case selects the experts with a larger weight with higher probability than experts with a smaller weight; hence, some prior knowledge is used because experts with larger weights reflect those that have a smaller loss. We refer to these two variants as FTL-1 and FTL-2, respectively. This process of simulating error variances and weights of the individual experts, and computing $\mathbb{V}[\hat{\epsilon}_{k^*}]$ and $\sum_{k=1}^{t} w_{k,t}^2 \mathbb{V}[\hat{\epsilon}_k]$ is performed over 10,000 trials.

The histograms of the error variance for the upper bound over the 10,000 trials are presented in Figure 4. The results for $t = \{2, 5, 15, 25\}$ (i.e., the number of experts in the ensemble) are shown. Of particular interest is the observation that the error variance for the bound decreases as classifiers are added to the ensemble that is combined using weighted majority voting. Recall that the benefit of ensemble of experts (classifiers) is that the ensemble can result in lower error variance as the ensemble size increases. On the other hand using a model selection method, such as FTL that chooses only one best expert, cannot provide consistently lower error variances than a weighted majority vote. In fact, for $t = 25$, the weighted majority vote ensemble had a smaller error variance than the FTL model in 9,739 out of 10,000 trials performed.

IV. Evaluation on Real-World Data Streams

To demonstrate the effectiveness of the ensemble approaches – and hence the difficulty of using the follow–the–leader (FTL) algorithm with concept drift – we present average classification error of FTL, simple majority vote (SMV), weighted majority vote (WMV), and Learn^{++}.NSE [9] on several synthetic and real-world data streams. All MES use CART as their base classifier and the ensemble size is limited to 25 CART models before age-based pruning is applied [30]. Pruning is applied to limit the computational resources for the larger data streams. The error is reported as the average of the individual time stamp errors. We used the chess [31], electricity pricing [32], NOAA weather [9], and Spam data sets [33] to carry out our experiments. The batch size (m) and number of time stamps (t) for each data set is indicated in Table II, and the data streams are evaluated using the test-then-train scheme. In the test-then-train setting, the MES begins by testing on a data set whose labels are not available to the algorithm. Then, the labels become available, and the data set is used for training at the next time stamp. The code used to implement the algorithms presented in this section can be found at http://github.com/gditzler/IncrementalLearning.

TABLE II
Average classification error of four MES approaches on data streams with concept drift. The number of time stamps is indicated by t and batch size by m.

	t	m	**FTL**	**SMV**	**WMV**	**Learn^{++}.NSE**
sea	200	250	20.35	13.37	13.17	13.22
chess	15	35	38.43	33.74	36.6	37.48
elec2	220	125	25.49	23.79	23.52	22.14
noaa	151	120	35.77	24.21	26.14	23.16
spam	46	100	17.57	8.85	9.52	10.04

Table II presents the average classification error of the four ensemble classifiers. From this table, we observe that the FTL method is outperformed by the other MES on nearly all data streams evaluated. We attribute the superiority of MES approaches over FTL to the variance of the estimated loss and unpredictability of $\lambda_{T,k}$. Furthermore, without a proper method of extracting $\lambda_{T,k}$, we should not expect FTL to perform well given (4), the variance simulation in section III-C, and the loss bound simulations we have in [20]. The accuracy for the different MES approaches on the SEA data stream – monitored over the duration of the experiment – is shown in Figure 5. From this figure, we observe that the FTL approach to learning in nonstationary environment is not able to adapt to the changing environments, while the MES approaches have no difficulty in tracking the drifting distribution.

V. Conclusion

One of the primary issues with using a multiple expert system (MES) in a nonstationary environment is the selection of an appropriate set of weights for each of the experts. The selection of expert weights are chosen such that the error of the MES is expected to be arbitrarily small on a target data set. Thus it is important to understand the various terms that are involved in computing the upper bound on the MES loss. In this contribution we presented a formal loss bound for a MES learning from a stream of data drawn from a nonstationary environment, also known as learning in the presence concept drift. The analysis has shown that the components controlling the bound can be related to dif-

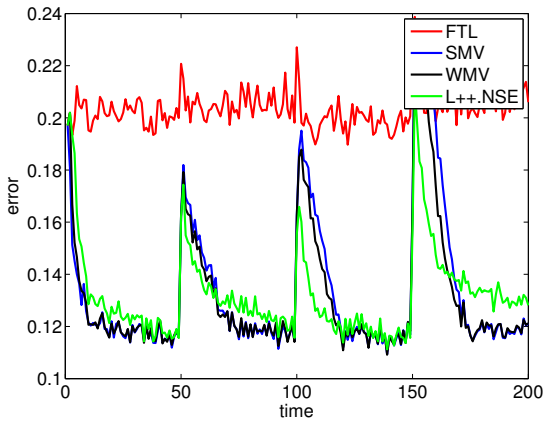


Fig. 5. Classification error of four ensemble methods on the classical SEA data stream.

ferent types of concept drift, such as real and virtual drift. Furthermore, through a simulation we have demonstrated that using a single classifier method, such as follow-the-leader, can lead to high variance in the estimate of the upper bound. The variance can be reduced by adding experts to the ensemble of classifiers and giving each expert nonzero weights. We tested four MES implementations on synthetic and real-world data streams, where we showed that the weighted MES approach to classification in nonstationary environments provides lower classification error rates than relying on a single model.

Our future work includes developing a framework for prediction in nonstationary environments that use the three terms in the MES loss bound to improve the classification error of the ensemble. The improvements to the state-of-the-art requires estimating the divergence of the distributions and labeling functions. Our prior work on semi-supervised learning provides a promising start to the implementation of such a MES approach for nonstationary environments [26], [27].

REFERENCES

[1] M. Sugiyama and M. Kawanabe, "Machine learning in nonstationary environments." The MIT Press, 2012.

[2] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Computing Surveys*, 2013.

[3] L. I. Kuncheva, "Classifier ensembles for changing environments," in *Proceedings 5th International Workshop on Multiple Classifier Systems*, pp. 1–15, 2004.

[4] L. L. Minku and X. Yao, "DDD: A new ensemble approach for dealing with concept drift," *IEEE Transactions on Knowledge Discovery and Data Engineering*, vol. 24, no. 4, pp. 619–633, 2012.

[5] J. Gao, W. Fan, J. Han, and P. S. Yu, "A general framework for mining concept-drifting data streams with skewed distributions," in *Proceedings of the 7th SIAM International Conference on Data Mining*, pp. 203–208, 2007.

[6] J. Gao, B. Ding, W. Fan, J. Han, and P. S. Yu, "Classifying data streams with skewed class distributions and concept drifts," *IEEE Internet Computing*, vol. 12, no. 6, pp. 37–49, 2008.

[7] S. Grossberg, "Nonlinear neural networks: Principles, mechanisms, and architectures," *Neural Networks*, vol. 1, no. 1, pp. 17–61, 1988.

[8] B. Settles, "Active learning literature survey," Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2010.

[9] R. Elwell and R. Polikar, "Incremental learning of concept drift in nonstationary environments," *IEEE Transactions on Neural Networks*, vol. 22, no. 10, pp. 1517–1531, 2011.

[10] G. Ditzler and R. Polikar, "Incremental learning of concept drift from streaming imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2283–2301, 2013.

[11] A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavalda, "New ensemble methods for evolving data streams," in *Knowledge and Data Discovery*, 2009.

[12] G. Ditzler and R. Polikar, "Hellinger distance based drift detection for nonstationary environments," in *IEEE Symposium on Computational Intelligence in Dynamic and Uncertain Environments*, pp. 41–48, 2011.

[13] C. Alippi and M. Roveri, "Change detection tests using the ICI rule," in *International Joint Conference on Neural Networks*, pp. 1190–1196, 2010.

[14] J. Kolter and M. Maloof, "Dynamic weighted majority: An ensemble method for drifting concepts," *Journal of Machine Learning Research*, vol. 8, pp. 2755–2790, 2007.

[15] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer-Verilag, 2nd ed., 1999.

[16] S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural Computation*, vol. 4, pp. 1–58, 1992.

[17] G. Brown, J. Wyatt, and P. Tiňo, "Managing diversity in regression ensembles," *Journal of Machine Learning Research*, vol. 6, pp. 1621–1650, 2005.

[18] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. John Wiley & Sons, Inc., 2004.

[19] G. Brown, *Encyclopedia of Machine Learning*, ch. Ensemble Learning. Springer Press, 2010.

[20] G. Ditzler, G. Rosen, and R. Polikar, "Discounted expert weighting for concept drift," in *IEEE Symposium on Computational Intelligence in Dynamic and Uncertain Environments*, pp. 61–67, 2013.

[21] W. N. Street and Y. Kim, "A streaming ensemble algorithm (SEA) for large scale classification," in *Proceedings to the 7th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 377–382, 2001.

[22] D. Brzezinski and J. Stefanowski, "From block-based ensembles to online learners in changing data streams: If- and how-to," in *ECML PKDD, Workshop on Instant Interactive Data Mining*, 2012.

[23] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman Vaughan, "A theory of learning from different domains," *Machine Learning*, vol. 79, pp. 151–175, 2010.

[24] I. Žliobaitė, "Expected classification error of the euclidean linear classifier under sudden concept drift," in *International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 2, pp. 29–33, 2008.

[25] D. Kifer, S. Ben-David, and J. Gehrke, "Detecting change in data streams," in *Proceedings of the 30th VLDB Conference*, pp. 180–191, 2004.

[26] G. Ditzler, G. Rosen, and R. Polikar, "A transductive learning algorithm for nonstationary environments," in *International Joint Conference on Neural Networks*, pp. 945–952, 2012.

[27] G. Ditzler and R. Polikar, "Semi-supervised learning in nonstationary environments," in *International Joint Conference on Neural Networks*, pp. 2471–2478, 2011.

[28] P. Ruvolo and E. Eaton, "ELLA: An efficient lifelong learning algorithm," in *International Conference on Machine Learning*, 2013.

[29] P. Ruvolo and E. Eaton, "Scalable lifelong learning with active task selection," in *AAAI Conference on Artificial Intelligence*, 2013.

[30] R. Elwell and R. Polikar, "Incremental learning in nonstationary environments with controlled forgetting," in *International Joint Conference on Neural Networks*, pp. 771–778, 2009.

[31] I. Žliobaitė, "Change with Delayed Labeling: when is it detectable?," in *IEEE International Conference on Data Mining Workshops*, 2010.

[32] M. Harries, "Splice-2 comparative evaluation: Electricity pricing," tech. rep., The University of South Wales, 1999.

[33] S. Delany, P. Cunningham, and A. Tsymbal, "A comparison of ensemble and case-base maintenance techniques for handling concept drift in spam filtering," in *International Conference on Artificial Intelligence*, pp. 340–345, 2006.