Classification-based event detection in ecological monitoring networks

Jayant Gupchup & Andreas Terzis
Computer Science Department, Johns Hopkins University, Baltimore Maryland 21210

Zhiliang Ma & Carey Priebe
Applied Mathematics & Statistics Department, Johns Hopkins University, Baltimore Maryland 21210

ABSTRACT: **Power-budgeting is a fundamental challenge in sensor networks today and the energy requirement of different sensing modalities is unevenly distributed. As a result, it is advisable to activate power-hungry sensors only during informative periods. Using low-power sensors, one can predict these informative periods due to strong correlations exhibited by environmental modalities. In this article, we consider an application of detecting "events" using classification based methods to increase the lifetime of the network. Specifically, we explore the problem of using low-power sensors to predict precipitation, which is one of the primary drivers of ecological activity. Such predictions can allow us to schedule the activation of expensive sensors (such as CO_2) when they are most informative. In order to achieve this trade-off between power and collecting informative data, we focus our efforts on predicting/classifying precipitation based on features extracted from inexpensive ambient temperature and barometric pressure modalities. Experimental results obtained from weather data collected over multiple years demonstrates that we can achieve accuracy towards 80% using these low-cost modalities and simple linear classifiers.**

1 INTRODUCTION

Environmental sciences require continuous collection of data at varying spatial and temporal resolutions to understand different processes. A number of research groups ([2], [3], [4]) have demonstrated the use of sensor networks to collect data at scientifically-relevant resolutions. Even though the technology has been employed extensively, these battery powered networks need to address a number of power-budgeting challenges. The power consumption of some commercially available sensors is listed in Table 1. In order to increase the lifetime of the network and maximize the collection of informative data, one must be judicious in the use of power-hungry sensors.

Data collected during events such as rain and snow are a subject of interest to many environmental studies, particularly to ones engaged in soil ecology. Precipitation events serve as major catalysts for ecological activity, and data gathered during these periods are crucial for understanding many ecological processes.

In this article, we explore the prediction of the onset and departure of precipitation events (i.e. rain, snow) in the context of an ecology monitoring sensor network [4]. To the best of our knowledge, most commercially available precipitation sensors (such as [5]) are power-hungry, and are not well-suited to be driven by small batteries (~19 *Ah*) that are employed in our target sensor networks applications. Thus, we need to predict the onset and departure of events using low-power modalities such as ambient temperature (*AT*) and barometric pressure (*BP*) that are "cheap" to sample. Figure 1 shows the signatures shown by *AT* and *BP* before the onset of rain and after its departure. We treat the event prediction problem as a two-class problem (Precipitation, No-precipitation) by using data obtained from *AT* and *BP*. The output of such a predictor-classifier can serve as the input to a scheduler, which schedules the sampling of a power-intensive CO_2 sensor that is most informative (from an ecology perspective) during such events.

We focus our attention on extracting a set of features that allows us to build a predictor-classifier using *AT* and *BP*. Principal component analysis (PCA) is employed to reduce dimensionality and select features for two major reasons: 1. Reducing computation and space overhead, and 2. Improving performance by avoiding the curse of dimensionality [11]. We compare the performance of various classifiers differing in complexity (linear to non-linear), execution and storage costs using these selected features. The performance of various classifiers using these features is evaluated using misclassification error

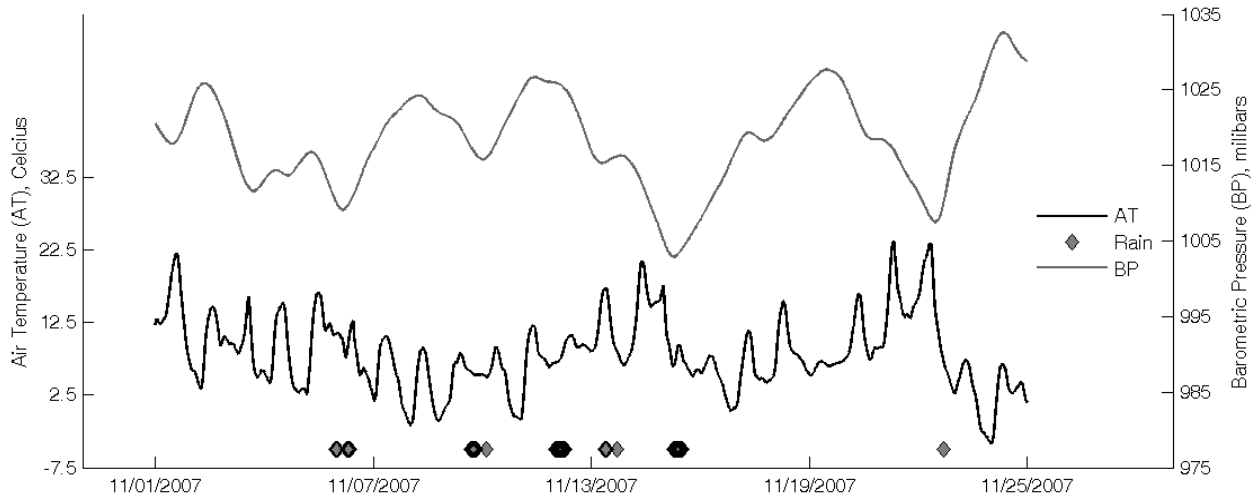

Figure 1 : An illustration of the typical signatures shown by Air temperature (*AT*) and Barometric pressure (*BP*) during the onset and departure of rain events. Note the drop in *BP* prior to the rain events and the deviation of AT from the diurnal pattern.

and the Brier metric [12]. Experimental results based on 6 years of meteorological data show that the accuracy of using *AT* and *BP* features in conjunction with simple linear classifiers is towards 80% and are comparable to the performance of non-linear classifiers built on the same set of features. We find that the misclassification error of *BP* is lower than *AT* by as much as 15%. The brier score decomposition allows us to understand and analyze the probability of prediction. We find that quality of the prediction on both ends of the probability spectrum is fairly good and trustworthy.

The rest of the paper is organized as follows. In Section 2 we introduce background information and survey similar bodies of work. In Section 3, we formulate the problem and provide a summary of the methods and metrics used to evaluate the solution. In Section 4, we present the dataset used for our study. The extraction of features is outlined in Section 5. In Section 6, we present our results, and finally, in Section 7, we conclude.

2 BACKGROUND

Precipitation events are known to be the dominant catalysts responsible for increase in ecological activity in the soil. The period during precipitation, and a short duration after precipitation stops, is known to be the critical period for major ecological activity. Data collected during this period can address a number of questions of interest to soil ecologists. Soil CO_2 and other gas data are important to address matters related to soil respiration and exchange of gases in the carbon cycle. Sensor networks allow us to capture this information remotely in a non-invasive, low-cost, continuous fashion.

Table 1: Power consumption of some commercially available sensors

Sensor Type	**Power Consumption**
Barometric Pressure [6]	< 36 µW
Humidity Temperature [7]	80 µW
Soil Moisture [8]	19.6 mW
CO2 (Research) [9]	26 mW
Precipitation [5]	< 30 mW
CO2 (Product) [10]	< 4 W

In typical environmental monitoring networks, each mote is powered by a small battery (typically ~ 19 *Ah*). Currently, the power consumed by commercially available gas sensors is significant [10]. A number of research groups (e.g. [13]) are working towards enabling low power gas sensing technologies that are envisioned to be used by sensor network applications such as the life under your feet [4]. So et al. describe their work in demonstrating a technology they refer to as LaserSPECks [9]. The goal of this work was to develop a proof-of-concept prototype that demonstrates how this low cost gas sensing technology can be interfaced with typical motes (e.g. Telos) and be used in sensor network applications. Their early prototype (LaserSPECk v1.0) suggests that one can interface a Telos mote with this technology, and the power drawn by the CO_2 sensor is as low as 26 mW. However, we note that this technology is not yet commercially available, and despite reduced energy requirements, applications would still need to apply selective (or adaptive) sampling strategies to balance the power budget to increase network lifetime.

2.1 *Related work*

A number of bodies of work have focused their efforts on detecting events in sensor network applications. Bulk of the efforts has involved heuristic based approaches in an effort to minimize the power. Work done by Abadi et al. is one such example of a heuristic-based in-network approach. Their system declares an event when certain pre-specified conditions are met [14]. Gupchup et al. provide an offline PCA-based method to detect events in environmental networks using ambient temperature and soil temperature modalities [15]. Obst et al. demonstrate the use of an offline echo state network to detect anomalies in monitoring gas concentrations in underground coal mines. They conclude that echo state networks are more effective in modelling dynamical systems and they outperform Bayesian network based method [16]. Chang et al. build on this work and demonstrate the feasibility of implementing an echo state network on a mote-class device [17]. Furthermore, they unify fault detection and event detection under a general framework. Most of the prior work, with the exception of [17], has focused on offline event detection. It is worth mentioning that the definition of an event differs from system to system. Our work is primarily targeting environmental monitoring application in which the detection of rain events can enable the system to make more meaningful decisions. For example, this work might prove useful for agriculture and water-management agencies that require predicting precipitation activity in remote locations in a low-cost, low-power fashion.

Prediction of precipitation can be a complex and challenging task. Modern systems have tremendous quantities of global spatial patterns and computing resources available to them to predict the onset of rain. An example of one such system is the service run by the Hydrometeorological Prediction Center [18]. Mears et al. provide an algorithm that can detect rain using features extracted from wind speeds [19]. Providing an exhaustive list of the work being done in this field is extremely difficult, and hence, we only provide some major ones for completeness. Typical environmental monitoring sensor networks operate in remote locations under harsh conditions. In most practical settings they have limited or no connectivity to the Internet. Consequently, they cannot make use of such prediction services. Under these restrictions, they need to predict precipitation using data and variables that are collected by the network locally.

3 PROBLEM DESCRIPTION

In this section, we describe the formalism involved in setting up the prediction problem as a 2-class classification problem.

3.1 *Precipitation Prediction: 2 class problem*

Let us denote the actual precipitation class-label at a given time instant, t+1, by the random variable $\boldsymbol{Y}_t$. Let $\boldsymbol{X}_{t-m}$ denote a d-dimensional feature vector obtained using measurements collected between time instants $(t - m)$ and t. Then, $(\boldsymbol{X}_{t-m}, \boldsymbol{Y}_t)$ denotes a pair of random variables such that $\boldsymbol{X}_{t-m} : \boldsymbol{\Omega} \rightarrow \mathfrak{R}^d$ and $\boldsymbol{Y}_t$ is given by $\boldsymbol{Y}_t = \boldsymbol{I}_{\{\text{precipitation = true, at time instant } t+1\}}$.

We cast the prediction problem as a 2-class classification problem with the additional constraint that for any given time, t, the class-prediction, $\hat{Y}_t$ must only depend on the observed values, $\boldsymbol{X}_{t-m}$ for $m \leq t$. The problem effectively translates to designing a classifier that takes as input, features that have been observed until the current time instant t, and predicts whether or not precipitation will be observed at time t+1. In the context of sensor networks, the classifier must possess the following desired properties: (1) low misclassification rate; (2) low computational, storage and communication costs. Typical sensor network contains many motes (or nodes). Potentially, these motes could collaborate to obtain a better prediction at the cost of communicating among themselves. However, in this article we assume that each mote makes a prediction independently, and hence, do not consider the collaboration aspect of this problem. We reserve this as a subject of future work.

3.2 *Classification*

For any classifier $\boldsymbol{g} : \mathfrak{R}^d \rightarrow \{0, 1\}$. The performance of $\boldsymbol{g}$ is measured by $\text{Prob}\{\boldsymbol{g(X)} \neq \boldsymbol{Y}\}$ where $\boldsymbol{X}:\boldsymbol{\Omega} \rightarrow \mathfrak{R}^d$ and $\boldsymbol{Y}:\boldsymbol{\Omega} \rightarrow \{0, 1\}$. Using notation borrowed from [20], let us define $\boldsymbol{L}:\boldsymbol{L(g)} = \text{Prob}\{\boldsymbol{g(X)} \neq \boldsymbol{Y}\}$. A good classifier has a low misclassification probability. The "optimal classifier" (a.k.a Bayes classifier), $\boldsymbol{g^*}$, is one that has the minimum probability of misclassification. Mathematically speaking, $\boldsymbol{L(g^*)} \leq \boldsymbol{L(g)}$. This gives us a lower bound on the performance of any classifier g. From here on, we refer to $\boldsymbol{L(g^*)}$ as $\boldsymbol{L^*}$. The Bayes optimal classifier for a 2-class problem is given by :

$$\boldsymbol{g^*(x)} = \begin{cases} 1 \text{ if } \text{Prob}\{\boldsymbol{Y} = 1 | \boldsymbol{X} = x\} > 1/2, \\ 0 \text{ otherwise.} \end{cases}$$

Computing the joint probability distribution of $\mathbf{X}'_i$s and $\mathbf{Y}'_i$ s is not practical. As a result, $\boldsymbol{g^*}$ is difficult to obtain. However, theoretical results such as one furnished by Stone [21] shows that for all distributions and for some $k : k \rightarrow \infty, k/n \rightarrow 0$, the probability of error of the k-NN (given by $\boldsymbol{L}_{k-NN}$) classifier converges to $\boldsymbol{L^*}$, where k is the number of neighbours and n is the number of labelled vectors. Furthermore, Stones' work [21] results in the power-

ful inequality, $\boldsymbol{L^* \leq L_{NN} \leq 2L^*}$, where $\boldsymbol{L_{NN}}$ is defined as the performance of the 1-NN as $n \rightarrow \infty$. For a more complete treatment on classifiers, the authors strongly urge the reader to refer to [20]. These theoretical results provide us some intuition regarding how far we are from the optimal classifier.

3.3 *Classification spectrum*

A number of factors are responsible for the selection of a classifier for a classification problem. Data characteristics, memory and computing constraints are some of the major factors that govern this choice. Given our problem structure, we are interested in comparing the efficiency of a range of classifiers without losing sight of the complexity involved in the classification. We consider five classifiers ranging from linear ones (e.g. naive Bayes) to non-linear ones (*k*-NN). To this effect, the classifiers used in this study are: (1) Naive Bayes (NB); (2) Fisher's linear discriminant (LDA); (3) Support Vector Machines (SVM); (4) Random Forests (RF) [22], and (5) *k*-NN [23] . In support of the fact that as $lim_{k/n \rightarrow, n \rightarrow \infty}\ \boldsymbol{L_{k-NN} = L^*}$, we evaluate $\boldsymbol{L_{k-NN}}$ by varying *k*.

3.4 *Another metric: Brier Score*

Many classifiers provide us with the probability associated with the class prediction. The brier score is commonly used to evaluate probability forecasts [12]. The brier score is the mean squared difference between the predicted probabilities and the observed event (represented as {0, 1}). Typically, the brier score is computed by stratifying the probability forecasts into bins. Let *n* events be distributed in *m* non overlapping bins based on the probability prediction. If e_{kj} represents the indicator of *j*th event falling in the *k*th bin , and p_{kj} represents the probability prediction for e_{kj} , then the brier score *(BS)* is given by

$$BS = \frac{1}{n} \sum_{k=1}^{m} \sum_{j=1}^{nk} (p_{kj} - e_{kj})^2$$

BS can be decomposed into five components as shown by Stephenson et al [24]. *BS* can be rewritten as the following:

$$\begin{aligned} BS = & \frac{1}{n} \sum_{k=1}^{m} n_k (\bar{p}_k - \bar{e}_k)^2 \\ & - \frac{1}{n} \sum_{k=1}^{m} n_k (\bar{e}_k - \bar{e}_k)^2 \\ & + \bar{e}(1-\bar{e}) \\ & + \frac{1}{n} \sum_{k=1}^{m} \sum_{j=1}^{nk} (p_{kj} - \bar{p}_k)^2 \\ & - \frac{2}{n} \sum_{k=1}^{m} \sum_{j=1}^{nk} (e_{kj} - \bar{e}_k)(p_{kj} - \bar{p}_k) \end{aligned}$$

The bar symbol represents the mean operator.

The first, second, and third term represent reliability (REL), resolution (RES), and observational uncertainty (UNC) respectively. Within-bin variance (WBV) and within-bin covariance (WBC) between forecasts and observations are given by the fourth and fifth term. Typically, WBV and WBC are small and can be ignored. An effective predictor function will have a low brier score. REL represents the extent to which the prediction matches the observed events. RES captures the difference between overall sample event frequency and the observed frequency for each bin. Note that the RES term is negative so a large value leads to a reduction in the brier score. The UNC term does not depend on the prediction. It is purely a function of the sample observations and represents the uncertainty in the labelled samples.

4 DATASET

Weather data recorded by a NOAA weather station is used for our experiments. The Jug Bay (JB) weather station is located at the Anne Arundel county of Maryland [25]. *AT*, *BP* and precipitation data recorded by the JB station from January 2003 to October 2008 (1811 days) were used for the purpose of this study. The sampling interval of this weather station is 15 minutes.

To our knowledge, there is no standard definition of an "event", and as a result, we specify our definition here. Non-zero precipitation measurements that occur within a window of 10 hours from each other are clustered together and considered to be a part of the same event. Precipitation events that are less than 5 mm of cumulative rainfall were not considered as they were considered to be insignificant. Furthermore, in order to establish data independence, we did not consider events whose start time was less than 48 hours from the end time of the previous event. Using this definition of an event, 124 rain events were known to occur for the JB location. The 24-hr period after the end of the event is extracted and is tagged as no-event. In addition to these vectors, a few no-events periods are selected at random and a 24-hr period is extracted from these periods. The total number of no-event periods in this dataset is 135. Next, we describe the process of preprocessing the *AT* and *BP* data that is used in the analysis.

4.1 *Data Preprocessing*

The data is first smoothed to mitigate the effect of sharp transients caused due to instrument errors or

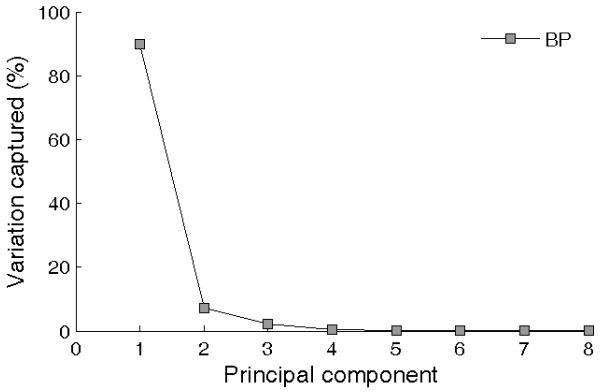

Figure 2 (a) : Scree Plot for the *BP* vectors.

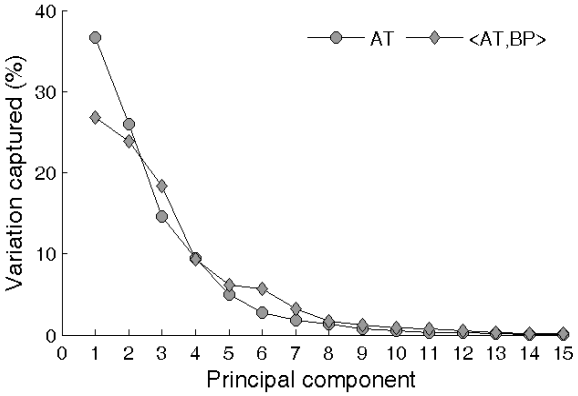

Figure 2 (b) : Scree Plot for the *AT* and <*AT,BP*> vectors.

Figure 2 : Percentage of variation captured by the principal components for the *BP*, *AT* and <*AT,BP*> vectors.

other reasons. The temperature and pressure modalities are both smoothed using a simple moving average filter with a width of 1 hour. Next, we begin to remove the effects of well-known priors in our data. *AT* shows strong trends, diurnal and annual patterns, whereas pressure does not show these periodic components. The temperature data is de-trended by first creating a smooth moving average signal with a window of 24 hours, and then subtracting this smooth signal from the original signal. The resulting signal, *S*, contains the annual and diurnal components, but does not contain the low frequency trend signal. In order to remove these periodic components, a daily profile is created for each day using the detrended series, *S*. This profile is created by pooling together data across consecutive days that share the same minute of day (MoD) value. In practise, we allow a window of one week around the day being considered. Then, for each 15 minute measurement of the day being considered, the profile MoD value is obtained by averaging across the days that fall within the window and share the same MoD value. As an example, let T_d^m represent the profile value for the mth minute for day *d*. This is obtained by taking the mean of seven values i.e. < S_{d-3}^m... S_{d+3}^m >. We then subtract the daily profile from the *S* series to obtain the residual series referred to as *A*. *BP* does not exhibit well behaved seasonal components. We do not remove the trend and seasonal components as done with air temperature.

5 FEATURE EXTRACTION

We begin extracting features corresponding to the event and no-event periods by making use of well known processes and priors for *AT* and *BP*. Figure 1 illustrates the typical diurnal cycle shown by the *AT* modality. This bell-shaped pattern is the most dominant foreground signal and is brought about by the day-night cycle. Well-behaved (or typical) days tend to adhere well to this dominant foreground signal whereas "event days" tend to deviate from it. One can also observe the falling *BP* trend prior to the precipitation event.

The residual series, *A*, captures the deviation of the *AT* signal from the expected daily pattern. For each instance of the event class, the 24-hour period prior to the start of the event is extracted from the *A* series. These vectors consist of the *AT* vectors corresponding to the event class. Similarly, for each instance of the no-event class, the 24 hour period after the end of the event is extracted from the *A* series to form the no-event *AT* vectors. We note that the deviation of the signal of an event day also depends on the time of the day. Since events start and end at different times of the day, these vectors are normalized by cyclically shifting them by an amount equal to the MoD value of the event start or end time.

The onset of precipitation is marked by a sharp drop in *BP* (Figure 1). Specifically, a consistent drop in barometric pressure for 12 hours indicates rough weather (high chances of a rain event), whereas a sharp rise in barometric pressure generally indicates a period of fair (event free) weather. The *BP* event vectors are obtained by extracting the 12 hour signal prior to the event start time after removing the baseline signal (mean). Similarly, the *BP* no-event vectors are obtained by extracting the 12 hour signal after the event ends.

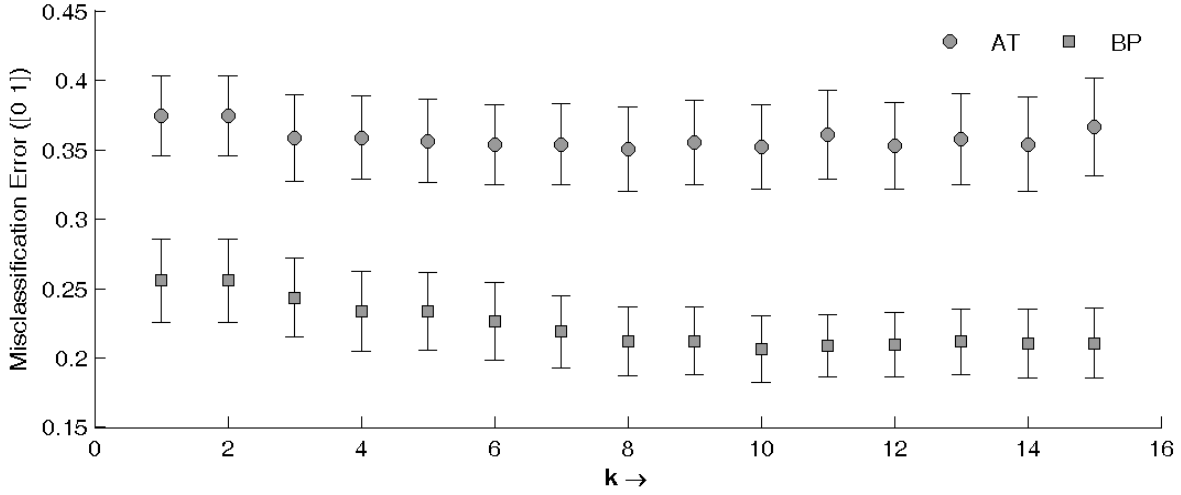

Figure 3 : Performance of k-NN classifiers on *AT* and *BP* features. The bars indicate one standard deviation.

In this study, we also investigate the effect of combining *AT* and *BP* signals on the classification. The *AT* and *BP* corresponding to each event and no-event class are combined to represent <*AT,BP*> vectors.

5.1 *A. Dimensionality Reduction*

The *AT* and *BP* vectors are basically transformed time signals. As a result, the dimensions are strongly correlated to each other. In order to reduce the dimensions and remove the correlations, we employ the well-known principal component analysis method to reduce dimensionality and capture orthogonal directions along which the variation is maximum. The columns of <*AT,BP*> are normalized such that their standard deviation is 1. Since the sampling rate is 15 minutes, the original dimensions of the *BP*, *AT* and <*AT,BP*> vectors are 48 (12 hours), 96 (24 hours) and 144 respectively. The percentage of variation captured by the first few principal components for each set is shown in Figure 2. Based on the scree plots, we selected to keep 5, 8 and 10 components for the *BP, AT* and <*AT,BP*> vectors. The original data are projected onto these principal components to form the feature vectors.

6 RESULTS

We begin by describing the methodology used to evaluate the performance of various classifiers on different modalities (Section 6.1)[1]. In order to establish an estimate of L^*, we evaluate the performance of using the *k*-nn classifier for varying values of *k* on all three vector sets i.e. *AT, BP* and <*AT,BP*> (Section 6.2). In Section 6.3, the performance of various classifiers is presented, and finally, in Section 6.4, we analyse the performance of the three sets using the brier metric.

6.1 *Methodology*

In order to evaluate various classifiers, we need to split the data in two sets: train and test. In this study, We create multiple train sets by re-sampling with replacement. This is done to obtain a distribution for the performance of various classifiers using the 259 vectors available to us. For each *i*, the train set, T_i, is created by sampling 66.6% of the vectors uniformly at random with replacement. In this case, the size of train set is 172 vectors. We then create a list, E_i, of all the vectors that are not contained in *Ti*. The E_i vectors comprise the test set for this group. We created 100 such groups i.e. *i* was varied from 1 to 100. Beleites et al. refer to this method as out-of-bootstrap error estimation [28]. We note that this method is preferred over the leave-one-out cross-validation method because we are interested in obtaining a distribution of the performance of various classifiers. Furthermore, this method is a similar to cross-validation as our test set and train set are disjoint for every group.

[1] We made use of the Matlab implementation provided by Franc et al for evaluating NB, LDA, SVM and *k*-NN [26]. The Matlab port provided by Abhishek Jaiantilal was used for evaluating Random Forests [27]

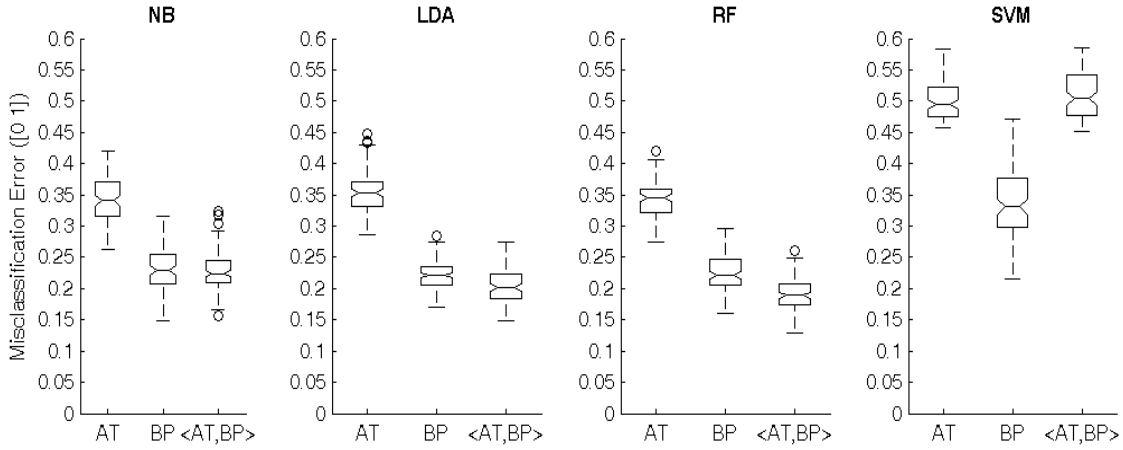

Figure 4 : Performance of classifiers using *AT*, *BP* and <*AT,BP*> features.

6.2 *Performance on k-NN classifiers*

The performance of the *k*-NN classifier was significantly lower when using the *AT* features compared to the *BP* features, as Figure 3 indicates. Using the t-test, no statistically significant differences were found between the *k*-NN classifiers using *BP* and <*AT,BP*> features at the 5% significance level, and hence, we do not report those results here. The NN performances for *AT* and *BP* were found to be 0.38 and 0.26 respectively. These finite sample performance results represent a weak upper bound for $\boldsymbol{L_{NN}}$. Stone's theorem implies that $\boldsymbol{L^* \leq L_{NN}/2}$, and consequently, 0.19 and 0.13 represent a weak upper bound of $\boldsymbol{L^*}$ using the *AT* and *BP* features respectively. Even with these estimates, we see that there is room for improvement. We also find that the performance does not change significantly beyond $k = 2$ and $k = 8$ for *AT* and *BP* respectively.

6.3 *Performance using other classifiers*

Figure 4 presents the performance of the NB, LDA, RF and SVM classifiers. RF using <*AT,BP*> achieves the best performance, whereas the performance of the SVM classifier trained using *AT* was found to be the worst[2]. There was no statistically significant difference between the median performance between *BP* and <*AT,BP*> for the NB and LDA classifiers. In other words, the 95% confidence intervals (given by the notches of the boxplot) for the median performance of the NB and LDA classifiers overlap. A somewhat surprising and interesting result was to find that the performance of simple linear classifiers is comparable to non-linear and computationally expensive classifiers. Furthermore, for linear classifiers, there is no statistically significant difference in performance using *AT* and <*AT,BP*> features. Once again, we use the confidence interval of the median to establish statistical significance. Lastly, one notes that none of the classifiers achieve a performance that is close to the estimate of $\boldsymbol{L^*}$ established using Stones theorem.

6.4 *Evaluation using the brier metric*

The brier score (BS) is computed as described in Section 3.4. It is useful in capturing the reliability (REL) and resolution (RES) of the probability forecasts. We used 10 probability bins (step size of 0.1) to compute the brier score. The probability outcomes of the LDA classifier are used to compute the BS for a given (train, test) group (cf. Sec.6.1). We chose the LDA classifier as it performs almost as well as RF and the probability of precipitation (PoP) estimation with LDA is significantly easier. Specifically, for each (train, test) group, one BS is obtained. Table 2 shows the median decomposed scores, and the BS obtained by combining the decomposed values. As mentioned in Section 3.4, the BS is a negatively oriented score. In other words, a low REL value and a high RES value are desirable.

We note that this methodology can be applied to any classifier which computes a probability of classification. For this purpose, we chose the LDA classifier because it performs almost as well as RF and the probability of precipitation using LDA is significantly easier.

[2] A number of different parameter settings were explored for the RF and SVM classifiers. The best results for RF were obtained by using the default settings provided by the RF library. The Gaussian kernel was found to provide the best results for the SVM classifier. In the interest of space, we do not report the results of using other settings.

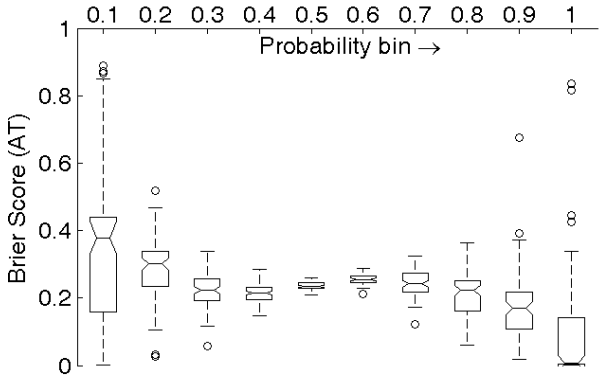

Figure 5(a) : Brier score for *AT* at different probability bins.

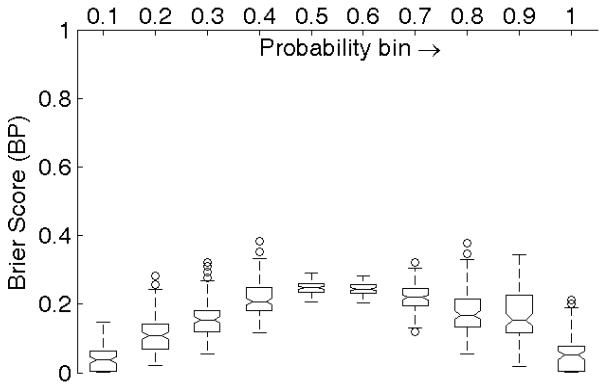

Figure 5 (b) Brier score for *BP* at different probability bin.

Table 2: Brier score and its decomposition for AT, BP and <AT,BP>

Type/ Score	BS	REL	RES	UNC	WBV	WBC
AT	0.229	0.023	0.042	0.249	0.0007	0.0012
BP	0.152	0.011	0.106	0.249	0.0008	0.0020
<AT, BP>	0.141	0.013	0.12	0.249	0.0008	0.0018

We find that the brier score of *<AT,BP>* and *BP* is significantly better than that of *AT*. Another useful property of the brier metric is that it allows us to access the probability forecasts across different bins. The brier scores corresponding to each probability bin for *AT* and *BP* are shown in Figure 5. We find that the scores are low at both ends of the probability spectrum for *BP* (Figure 5(b)). This implies that the predictions can be trusted more when the class-conditional probabilities provided by the LDA classifier is either high or low. This trust information can be an input in deciding which predictions to trust and which ones to ignore in a meaningful way. In comparison to *BP*, *AT* does not show low brier scores when the probability of prediction is low.

7 CONCLUSION

In this article, we explored the localized prediction of precipitation using ambient temperature and barometric pressure modalities. The goal of this work is to demonstrate that one can use the information present in variables that are "cheap" to sample to balance the conservation of power and collection of informative data for energy-constrained, ecological monitoring networks. Towards this goal, we used long term data obtained from a weather station to emulate the collection of data by a typical sensor node. We extracted features using domain knowledge and well known priors for these two modalities. Classifiers ranging from linear to non-linear were used for the purpose of "predicting/classifying" an event or a lack of it. Their performance was evaluated using misclassification error and the brier metric.

Our analysis demonstrates that one can use a very small set of features to obtain classification that is significantly better than chance. We find that simple classifiers based on these features are able to achieve accuracy up to 80%. The performance of linear classifiers such as naive Bayes and LDA are comparable to non-linear classifiers such as Random Forests and *k*-NN. We found that the barometric pressure modality is significantly more informative than ambient temperature in predicting events. Commercially available pressure sensors consume very little energy (cf. Sec. I) and they can be used effectively to detect the onset and departure of precipitation. Moreover, we note that most sensor motes have on-board temperature sensors, resulting in no additional hardware cost. The brier metric provides us with a way to analyse probability forecasts, and understand the performance of the linear classifiers in the probability spectrum. These results imply that one can design a fairly accurate light-weight predictor/classifier that is capable of running on the restricted environment of a mote.

Intuitively, the temperature and pressure signatures are more prominent when the magnitude of the event is larger. A shortcoming of our study is that it does not consider the magnitude of the event or inter-arrival time of events for predictions. In the future, we will investigate these aspects in our analysis and implement this system on the mote environment.

8 ACKNOWLEDGMENTS

We would like to thank Alex Szalay (Physics and Astronomy, JHU) for useful discussions and feedback that significantly improved the quality of the article. We thank Farzeena Lakdawala for proofreading and helping with the document formatting. Finally, we thank the anonymous referees for their insightful comments and feedback.

REFERENCES

[1] J. Polastre, R. Szewczyk, and D. Culler, "Telos: Enabling Ultra-Low Power Wireless Research," in IPSN/SPOTS 2005.

[2] G. Tolle, J. Polastre, R. Szewczyk, N. Turner, K. Tu, P. Buonadonna, S. Burgess, D. Gay, W. Hong, T. Dawson, and D. Culler, A Macroscope in the Redwoods," in SenSys 2005.

[3] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in ACM International Workshop on Wireless Sensor Networks and Applications 2002.

[4] A. Terzis, R. Musaloiu-E., J. Cogan, K. Szlavecz, A. Szalay, J. Gray, S. Ozer, M. Liang, J. Gupchup, and R. Burns, "Wireless Sensor Networks for Soil Science," International Journal on Sensor Networks.

[5] Vaisala Industrial Instruments, "All Weather Precipitation Gauge VRG101," Available at http://www.vaisala.com/weather/productsvrg101.html.

[6] Bosch Sensortec, "BMP085 Digital, barometric pressure sensor, Available at http://www.bosch-sensortec.com/content/language1/downloads/BMP085 Flyer Rev.0.2 March2008.pdf.

[7] Sensirion Inc., "SHT11 - Digital Humidity Sensor (RH & T) specifications," Available at http://www.sensirion.com/en/01humidity sensors/02 humidity sensor sht11.htm.

[8] Decagon Devices, "EC-5 Soil Moisture Sensor ," Available at http://www.decagon.com/ag research/soil/ec5.php.

[9] S. So, F. Koushanfar, A. Kosterev, and F. Tittel, "Laserspecks: Laser spectroscopic trace-gas sensor networks - sensor integration and applications," in IPSN 2007

[10] Vaisala Industrial Instruments, "CARBOCAP Carbon Dioxide Transmitter Series GMT220 specifications," Available at http://www.vaisala.com/instruments/products/gmt220.html.

[11] C. M. Bishop, Neural Networks for Pattern Recognition, 1st ed.Oxford University Press, USA, January 1996.

[12] G. W. Brier, "Verification of forecasts expressed in terms of probability," Monthly Weather Review, vol. 78, no. 1, pp. 1–3, 1950.

[13] Mid-InfraRed Technologies for Health and the Environment, Available at http://www.mirthecenter.org/.

[14] D. J. Abadi, S. Madden, and W. Lindner, "Reed: Robust, efficient filtering and event detection in sensor networks," in VLDB 2005.

[15] J. Gupchup, A. Terzis, R. Burns, and A. Szalay, "Model-Based Event Detection in Wireless Sensor Networks," in Workshop for Data Sharing and Interoperability on the World Wide Web 2007.

[16] O. Obst, X. R. Wang, and M. Prokopenko, "Using echo state networks for anomaly detection in underground coal mines," in IPSN 2008

[17] M. Chang, A. Terzis, and P. Bonnet, "Mote-based online anomaly detection using echo state networks," in DCOSS 2009.

[18] National Oceanic and Atmospheric Administration, "Hydrometeorological prediction center," Available at http://www.hpc.ncep.noaa.gov/.

[19] C. Mears, D. Smith, and F. Wentz, "Detecting rain with quikscat," Geoscience and Remote Sensing Symposium, 2000. Proceedings. IGARSS 2000. IEEE 2000 International, vol. 3, pp. 1235–1237, vol.3, 2000.

[20] L. Devroye, L. Gy¨orfi, and G. Lugosi, A Probabilistic Theory of Pattern Recognition (Stochastic Modelling and Applied Probability). Springer, February 1997.

[21] C. J. Stone, "Consistent nonparametric regression," The Annals of Statistics, vol. 5, pp. 595–620, 1977.

[22] L. Breiman, "Random forests," in Machine Learning, vol. 45, 2001, pp. 5–32.

[23] R. Duda, P. Hart, and D. Stork, Pattern Classification. Wiley, 2001.

[24] D. B. Stephenson, C. A. S. Coelho, and I. T. Jolliffe, "Two extra components in the brier score decomposition," Weather and Forecasting, vol. 23, pp. 752–757, 2008.

[25] National Estuarine Research Reserve, "Jug Bay (cbmjbmet)," Available at http://cdmo.baruch.sc.edu/QueryPages/anychart.cfm.

[26] V. Franc and V. Hlavac, "Statistical pattern recognition toolbox for Matlab," Center for Machine Perception, K13133 FEE Czech Technical University, Prague, Czech Republic, Research Report CTU–CMP–2004–08, June 2004.

[27] Abhishek Jaiantilal, "Random Forest (Regression, Classification and Clustering) implementation for MATLAB (and Standalone)," Available at http://code.google.com/p/randomforest-matlab/.

[28] C. Beleites, R. Baumgartner, C. Bowman, R. Somorjai, G. Steiner, R. Salzer, and M. G. Sowa, "Variance reduction in estimating classification error using sparse datasets," Chemometrics and Intelligent Laboratory Systems, vol. 79, no. 1-2, pp. 91–100, October 2005.