Knowl Inf Syst (2008) 14:299–326
DOI 10.1007/s10115-007-0095-1

REGULAR PAPER

Forecasting skewed biased stochastic ozone days: analyses, solutions and beyond

Kun Zhang · Wei Fan

Received: 22 March 2007 / Accepted: 28 April 2007 / Published online: 6 July 2007

Abstract Much work on skewed, stochastic, high dimensional, and biased datasets usually implicitly solve each problem separately. Recently, we have been approached by Texas Commission on Environmental Quality (TCEQ) to help them build highly accurate ozone level alarm forecasting models for the Houston area, where these technical difficulties come together in one single problem. Key characteristics of this problem that is challenging and interesting include: (1) the dataset is sparse (72 features, and 2 or 5% positives depending on the criteria of "ozone days"), (2) evolving over time from year to year, (3) limited in collected data size (7 years or around 2,500 data entries), (4) contains a large number of irrelevant features, (5) is biased in terms of "sample selection bias", and (6) the true model is stochastic as a function of measurable factors. Besides solving a difficult application problem, this dataset offers a unique opportunity to explore new and existing data mining techniques, and to provide experience, guidance and solution for similar problems. Our main technical focus addresses on how to estimate reliable probability given both sample selection bias and a large number of irrelevant features, and how to choose the most reliable decision threshold to predict the unknown future with different distribution. On the application side, the prediction accuracy of our chosen approach (bagging probabilistic decision trees and random decision trees) is 20% higher in recall (correctly detects 1–3 more ozone days, depending on the year) and 10% higher in precision (15–30 fewer false alarm days per year) than state-of-the-art methods used by air quality control scientists, and these results are significant for TCEQ. On the technical side of data mining, extensive empirical results demonstrate that, at least for this problem, and probably other problems with similar characteristics, these two straight-forward non-parametric methods can provide significantly more accurate and reliable solutions than a number of sophisticated and well-known algorithms, such as SVM and AdaBoost among many others.

K. Zhang
Department of Computer Science, Xavier University, New Orleans, LA, USA
e-mail: kzhang@xula.edu

W. Fan (✉)
IBM T.J.Watson Research, Hawthorne, NY, USA
e-mail: weifan@us.ibm.com

Keywords Sample selection bias · Probability estimation · Skewed distribution · Streaming · Random decision tree

1 Introduction and motivation

Ground ozone level depends on a sophisticated chemical and physical process as a function of many known and unknown factors, and stochastic in nature (i.e., with the same set of currently observable variables, the ozone level can differ from time to time). For many years, it has been an active topic for air quality study, an interdisciplinary field among atmospheric research, geochemistry and geophysics, since an ozone level above some well known threshold is rather harmful to human health, and affects other important parts of our daily life, such as farming, tourism etc. Therefore an accurate ozone alert forecasting system is necessary to issue warnings to the public before the ozone reaches a dangerous level. In air quality study, the estimation of ozone level uses known physical and chemistry reaction theories that attempt to explain the "true" mechanisms. There are several such theories around. As a result of these research, simulation systems, physical formulas and parametric models are created to calculate ozone level.

However, due to the difficulty of the problem and still limited knowledge about the true physical and chemical mechanism, existing approaches can only use a rather small number of parameters (≤ 10), and are still rather inaccurate and can be costly to build. However, it is a common belief among environmental scientists that a significant large number of other features currently never explored yet are very likely useful in building highly accurate ozone prediction model. Yet, little is known on exactly what these features are and how they actually interact in the formation of ozone. Information available is rather speculative in the sense that we roughly know a rather exhaustive set of features out there. Indeed, this candidate list contains over 60 features. As mentioned earlier, none of today's environmental science knows as of yet how to use them. This provides a wonderful opportunities for data mining. As discussed in Sect. 1.3, the combination of skewed distribution, large number of possibly irrelevant features, small training set size, feature selection bias, among others, is an interesting and challenging problem.

1.1 Seriousness of the ozone problem

Ground-level ozone O_3, the key ingredients of smog, is not emitted directly to the air like other air pollutants is formed as a result of a series of complex chemical reaction of Volatile Organic Compounds (VOCs) and or Nitrogen Oxides (NOx) in the presence of heat and sunlight. VOCs are emitted from a variety of sources, including motor vehicles, chemical plants, refineries, factories, and other industrial sources. Nitrogen oxides are emitted from motor vehicles, power plants, and other combustion devices such as off-road engines. Studies have shown that short term up to a few hours exposure to elevated ambient ozone can cause a number of health problems, such as asthma, chest pain and coughing [4]. The effects of long-term exposure is less established, although a few studies have associated long-term exposure to elevated ozone with decreases in lung function, exacerbation of existing asthma and causing new asthma [10]. Besides the effects on human health, elevated ozone level also has negative effects on vegetation and ecosystems, leading to reductions in agricultural and commercial forest yields, and increases plant susceptibility to disease, pests, and other environmental stresses. As part of the mandate of Clean Air Act, United States Environmental Protection Agency (EPA) established the National Ambient Air Quality Standards

(NAAQS) to regulate pollutants. In July 1997, the EPA announced a new 8-h 80 parts per billion (ppb) standard as an amendment to the previous 1-h 120 ppb standard. The new standard is for the protection against longer exposure to the ground-level ozone. The 8-h ozone is the average of the ozone concentration of the past 8 h. As a good case study, the Houston/Galveston/Brazoria (HGB) area of southeast Texas has been experiencing some of the highest ozone levels recently recorded in North America, out-pacing Los Angeles as the city with the most violation days in 1999 [8]. Recently, in 2004, the HGB area has a total of 52 days with measured 8-h ozone reaching the 80 ppb dangerous level. Currently the HGB is designated as a non-attainment area by EPA under the 8-h ozone standard. As the fourth largest city in the US, it is of great interest for Houston to achieve an attainment status before the EPA deadline in 2010. In the mean time, to monitor and forecast high ozone days before it actually happens remains the top priorities for Texas Commission on Environmental Quality (TCEQ).

1.2 State-of-the art in ozone forecasting

Ozone level forecasting has been an active area for environmental science and meteorology. There are mainly two family of methods, air dynamic and statistical models. The dynamic forecasting uses 3D air quality models to simulate the atmospheric processes that influence the formation, transport and dispersion of ozone. The statistical methods, on the other hand, find the empirical statistical correlation between ozone and atmospheric parameters such as wind, temperature, etc.

Two examples of dynamic models were used in the north of Spain [19]. The first model uses three modules for ozone forecasting. The mesoscale model (MASS) provides the initial condition to the non-local boundary layer model based on the transient turbulence scheme, while the third module is a photochemical box model (OZIPR) in Eulerian and Lagrangian modes and receives necessary information from the two previous modules. Quite different from the first model, the second forecast model, called MM5/UAM-V, is a grid model that predicts hourly 3D ozone concentration field. Both methods give good performance only for specific episode, but there is substantial computational cost in constructing these models and they are not portable to different locations, such as Houston. On the other hand, statistical forecasting is currently the most widely used method, primarily due to its low cost and competitive accuracy compared with the dynamic forecasting. Previously, various regression-based methods including regression trees, parametric regression equation, and artificial neural network (ANN), and others have been explored for specific datasets at different locations. However, Schlink et al. [21] conducted an inter-model comparison on 15 statistical techniques which were applied to ten data sets representing different meteorological and emission conditions throughout Europe. They found that none of the 15 techniques performs better than others in all aspects. Other non-regression based statistical methods explored previously include fuzzy logic [9,16], and Bayesian network [15].

1.3 Challenges as a data mining problem

In environmental science, the true model for ozone days are believed to be stochastic in nature. In other words, given all relevant features in the feature vector $\mathbf{x}_R$, the probability of an ozone day $y =$ "ozone day" conditional on $\mathbf{x}_R$ is non-trivial. Formally, $P(y = \text{“ozone day”}|\mathbf{x}_R) < 1$ and can be described as a density over $\mathbf{x}_R$. When the problem is stochastic, predictive mistakes and errors are inevitable.

The dataset, described in detail in Sect. 2.1, contains 2,500+ examples with 72 continuous features. Depending on the criterion for ozone days, either 2% or 5% of them are truly positive for the Houston area in the past 7 years. For data mining, it is a rather skewed and relatively sparse distribution. Small number of examples and large number of features increase statistical bias and variance even if we know the true stochastic model $P(y|\mathbf{x}_R)$'s exact form and use the training data to estimate parameters inside the model.

At the same time, only about 10 features among these 72 features have been verified by environmental scientists to be useful and relevant, and there is neither empirical nor theoretical information as of yet on the relevance of the other 60 features. However, air quality control scientists have been speculating for a long time that some of these features might be useful, but just haven't been able to either develop the theory or use simulations to justify their relevance. Part of our task is to test their possible relevancy using data mining techniques. This ought to be pursued with caution though, since the presence of a large number of features that are possibly irrelevant may seriously introduce overfitting problem.

By definition, the collected feature set $\mathbf{x}$ and relevant feature set $\mathbf{x}_R$ may not be the same, but are expected to have non-empty intercept $\mathbf{x}_R \cap \mathbf{x} = \mathbf{x}_r$, and $\mathbf{x}_r \neq \emptyset$. For convenience, define $\mathbf{x}_{ir}$ as the set of irrelevant features inside $\mathbf{x}$, or formally, $\mathbf{x} = (\mathbf{x}_r, \mathbf{x}_{ir})$. Since $\mathbf{x}_{ir}$ is independent from both y and $\mathbf{x}_r$, or formally, $P(y, \mathbf{x}_{ir}) = P(y)P(\mathbf{x}_{ir})$ and $P(\mathbf{x}_r, \mathbf{x}_{ir}) = P(\mathbf{x}_r)P(\mathbf{x}_{ir})$, then $P(y|\mathbf{x})$ is actually $P(y|\mathbf{x}_r)$. This is trivial since $P(y|\mathbf{x}) = P(y|\mathbf{x}_r, \mathbf{x}_{ir}) = \frac{P(y,\mathbf{x}_r,\mathbf{x}_{ir})}{P(\mathbf{x}_r,\mathbf{x}_{ir})} = \frac{P(y,\mathbf{x}_r)P(\mathbf{x}_{ir})}{P(\mathbf{x}_r)P(\mathbf{x}_{ir})} = P(y|\mathbf{x}_r)$. However the derivation $P(y|\mathbf{x}) = P(y|\mathbf{x}_r)$ is only true when the dataset is exhaustive, and this is clearly not the case for 7 years of ozone data. In fact, irrelevant features change the probability distribution represented in the data. For a normal day training example ($\mathbf{x}$, $y =$ "normal day"), irrelevant features tend to push the probability $P(y =$ "ozone day"$|\mathbf{x})$ down towards 0. Considering those irrelevant features, there is likely just one example with all these similar feature values (both $\mathbf{x}_r$ and $\mathbf{x}_{ir}$) that is a normal day. On the other hand, for ozone day example, irrelevant features are likely to push up the probability to 1. Intuitively, as more irrelevant features are introduced into the feature vector, the empirical probability conditional on irrelevant features tends to get closer to the two extreme cases, either 1 for ozone days or 0 for normal days.

In addition, the date of the ozone alarm cannot be ignored, since an inductive model trained from historical data will be used to predict ozone alarm in the future, and the number of "ozone days" varies from year to year in the Houston area. Considering the date, this problem can be formulated as either a "data stream" or "sample selection bias" problem. Evolving data stream is best described by changes in joint probability distribution $P(\mathbf{x}, y) = P(y|\mathbf{x})P(\mathbf{x})$. For ozone alarm forecasting, the physical law $P(y|\mathbf{x}_R)$ does not change over time, and as a result, neither $P(y|\mathbf{x})$ nor $P(y|\mathbf{x}_r)$ is expected to change. Observable values of $P(y|\mathbf{x})$ may change, however, this is due to limited number of labeled examples. Under evolving data stream framework, the only possible change is in feature vector probability distribution $P(\mathbf{x})$. In this sense, it is equivalent to "feature sample selection bias" as described below.

The dataset can be equally formulated as a "sample selection bias" problem [23]. Assume that $s = 1$ denotes that an example $(\mathbf{x}, y)$ is sampled from the universe of examples into the training set, and $s = 0$ denotes that $(\mathbf{x}, y)$ is not selected. Sample selection bias is best described by a dependency of $s = 1$ on feature vector $\mathbf{x}$ and class label y or $P(s = 1|\mathbf{x}, y)$. The sample selection bias is called a "feature bias", if it is explicitly dependent on feature vector $\mathbf{x}$ and conditionally independent from class label y or $P(s = 1|\mathbf{x}, y) = P(s = 1|\mathbf{x})$. Ozone day forecasting is an example of feature bias, since the training data set obviously is unlikely to contain too many "days" that are very similar to the future. Where there is sample selection bias, there are two closely-related challenges, (1) how to train an accurate model given sample selection bias, and (2) how to effectively use a model to predict the future with a different and yet unknown distribution.

1.4 Our contributions

Our work in this paper has made two main categories of contributions, one is a solution for ozone alarm forecasting that is more accurate than state of the art methods adopted by TCEQ, and the other is the experience to formulate the application as a data mining problem, the analysis to identify its unique combinations of technical challenges, as well as the process to search for the most suitable solutions. As discussed in [22], the latter contribution is a crucial step to make the data mining research meet practical requirements. Besides these two main contributions, the most important technical contributions are as follows:

- More accurate ozone forecasting system than current system used by TCEQ.
- Empirically show the relevance of about 60 features that environmental scientists have been speculating.
- How to estimate reliable probability from a dataset with a lot of possibly irrelevant features and very small number of examples.
- How to choose the best decision threshold to use a model trained from historical data to predict on new data under sample selection bias where neither feature vector nor true label is known beforehand.
- Empirically demonstrate that, comparing to a number of popular and more sophisticated algorithms, non-parametric probabilistic tree ensembles can offer more accurate and significant solutions to this problem.

2 Ozone alarm forecasting problem

We start by describing how the Houston area data is collected, the feature sets and ozone forecasting task.

2.1 Houston area dataset collection

We collected various meteorology and ozone data for the Houston, Galveston, and Brazoria (HGB) area. Seventy-two data attributes are extracted from several databases within two major federal data warehouse and one local database for air quality control. These are EPA Air Quality System (AQS) and National Climate Data Center (NCDC) [18] from the federal government as well as Continuous Ambient Monitoring Stations (CAMS) operated by TCEQ. The EPA AQS database is the national repository for information about airborne pollutants in the United States, and it provides the sensory information that is used in the regulatory feedback. Ozone exceeding above the National Air Quality Standard are based on the AQS data. The NCDC is the world's largest active archive of meteorological data. In addition, the CAMS database archives some detailed air parameters and local meteorological data not available in the federal database. Since each database contains many site stations, and there are several of these stations within the HGB area, we have chosen the information collected either at or closest to IAH-Bush International Airport for the study. This is because the weather station at IAH is the only one that records a wide range of hourly weather parameters in the HGB area. In addition, the archived datasets at IAH have the longest recording history, and usually the least missing value and erroneous readings. In summary, these 72 attributes contains various measures of air pollutant and meteorological information for the target area in our study.

Various air pollutant information from EPA AQS records and measures several sources of VOCs that contribute to the chemical reaction producing ozone in the target area. In addition, the CAMS Air database archives some additional air pollutants information not available from EPS AQS, and these include the amount of various pollutants such as carbon monoxide (CO), nitrogen dioxide (NO2), sulfur dioxide (SO2), nitric oxide (NOx), fine particular matters (PM10 and PM2.5), oxides of nitrogen, and hydrogen sulfide. When the recorded value for the same measurement is different between EPA AQS and CAMS, their average is used in our study.

Since the meteorology at both surface level and upper atmospheric level provides the physical condition that influences the formation, transport and dispersion of pollutants that contribute to ozone, meteorological data from both levels were obtained for modeling. NCDC Surface Airways (SA) database contains surface data such as relative humidity, ceiling height, sky cover, and etc. Other surface level features, such as temperature, ground wind speed and direction etc, are obtained from CAMS. For upper-air data, NCDC Radiosonde Data of North America dataset provides radiosonde observation (RAOB) record dated back to 1946. There are six RAOB stations in Texas. However, we use the data from the RAOB station in Lake Charles, Louisiana because it is the closest to Houston. About 15 variables are extracted from RAOB, including temperature (T), geopotential height (HT), dew point, wind speed and direction at the 850, 700, and 500 hPa levels. The variables are sampled twice a day and the average is used in the study. The wind speed and direction are converted to U (east–west) component and V (south–north) component. Additional variables that may affect the air quality include previous day pollutant level (for carry over effect) and day type (workday or non-workday, which affects NOX emissions). In summary, seven (1998–2004) years hourly ozone data, meteorological surface data and daily upper air meteorological data are collected in this study.

2.2 Current forecasting system

Since 1999, TCEQ started to issue ozone warning for public awareness for nine metropolitan areas within Texas, including the HGB area. TCEQ forecasts are primarily based on the Criteria method [4]. It is an expert-rule-based system developed over the years. Daily weather forecasts from National Weather Service (NWS) are fed into the Criteria model to predict if ozone levels will reach or exceed a target level for a particular area. The criteria for the HGB area is based on a rather small set of parameters, such as previous day ozone, maximum temperature and wind. The rules set are different for each month in the forecast ozone seasons, March to November for HGB area. In [12], researchers at TCEQ proposed a parametric ozone prediction model, called "local ozone peak model", based on monitored wind speed, temperature, and solar radiation, in conjunction with monitored estimates of upwind background levels. In this paper, we use the ozone peak model as baseline since it is the model currently promoted by TCEQ. This local ozone peak model uses the upwind ozone background level (default as 50 [4]), the maximum temperature in degrees F, the base temperature where net ozone production begins (50 F), the solar radiation total of the day, the wind speed near sunrise and midday. The emission factor can take values from 0 to 1. The following equation summarizes the parametric equation and the parameters involved.

$$O_3 = \text{Upwind} + \frac{Em\text{Factor} \times (T\max - T\text{b}) \times \text{SRd}}{W\text{Sa} \times 0.1 + W\text{Sp} \times 0.5 + 1} \tag{1}$$

in which,

- O_3 - Local ozone peak prediction
- Upwind - Upwind ozone background level
- EmFactor - Precursor emissions related factor
- Tmax - Maximum temperature in degrees F
- Tb - Base temperature where net ozone production begins (50 F)
- SRd - Solar radiation total for the day
- WSa - Wind speed near sunrise (using 09-12 UTC forecast mode)
- WSp - Wind speed mid-day (using 15-21 UTC forecast mode)

3 Addressing data mining challenges

We address those challenges as raised in Sect. 1.3.

3.1 Direct adaptations

Instead of predicting on class labels, it is well-known that a more effective approach on skewed and stochastic distribution is to directly estimate the probability distribution itself and then choose the best decision threshold to optimize on some given criteria, such as a compromise between precision and recall [14].

In general, they are two families of methods, either descriptive or generative, and sometimes called, non-parametric or parametric. Descriptive methods, such as decision trees, make rather loose assumption about the form of the true unknown probability distribution. Both the structure of the hypothesis and parameters within the hypothesis are estimated from the labeled training data. On the other hand, generative methods, such as naive Bayes and logistic regression, assume that the true conditional probability follows a "particular form". The formula is fixed and learning is to estimate the parameters used inside the formula with Maximum Likelihood Estimation. Since little is known about the true probability distribution of ozone days as a function of the large number of features, it is hard to choose the right generative methods, therefore descriptive methods are preferred.

3.2 Reliable probability estimation under irrelevant features

As discussed in Sect. 1.3, the conditional probability represented in the training data can be $P(y|\mathbf{x}_r)$ at its best if the dataset is exhaustively sampled. However, due to sample selection bias, it is in fact $P(y|\mathbf{x}_r, \mathbf{x}_{ir}, s = 1)$. As discussed earlier, the effect of irrelevant feature is to make the probability towards 0 or 1 or either under-estimate or over-estimate.

One way to solve this problem is to train multiple models, each of which are from a random feature subset. In other words, the model is built from $\mathbf{x}_r^s \subset \mathbf{x}_r$ and $\mathbf{x}_{ir}^s \subset \mathbf{x}_{ir}$. The effect to use a subset of $\mathbf{x}_r$ can make the probability less "sharp" (or away from 0 or 1), and the effect of $\mathbf{x}_{ir}^s$ is to make the probability sharper, but could be either in the correct or wrong directions. However, different models trained from different feature subset is unlikely to change the probability in the same direction unless these features are correlated. Without any knowledge about what features are relevant and which ones are not, the safest approach is to construct multiple models from different feature subset and average their predictions. This is because on average, the multiple model will not perform worse than any of the single models, an issue explained further below. Yet, a separate way to look at this issue is that variance in bias and variance decomposition can be reduced significantly with model averaging. There are many ways to train models from different feature subset. One of the simplest and somehow less ad hoc approach is to train multiple decision trees in different ways. Since there are 72 features and only 2,500 examples, a decision path can at most test 12 features ($2^{12} = 4,096$). Therefore, two different trees are very unlikely to consider the same 12 features out of 72, assuming all trees are uncorrelated in choosing features. Then, 50 trees can guarantee that all 72 features will be considered because $(1 - \frac{12}{72})^{50} \simeq 0.0001$.

Since the dependency on the irrelevant features $\mathbf{x}_{\mathrm{ir}}$ cannot be ignored, every single model θ constructed from the labeled training data is likely an inaccurate estimator of $P(y|\mathbf{x}_r)$. Let us denote the estimated probability by θ as $P(y|\mathbf{x}, \theta)$. We show that the expected error to estimate $P(y|\mathbf{x}_r)$ by averaging several models is no more than the expected error of any single model being averaged. This proof is adopted and modified from [3].

Formally,

$$\mathrm{MA}(\mathbf{x}) = \frac{1}{K}\sum_k P(y|\mathbf{x}, \theta_k) = E_{P(\theta)}(P(y|\mathbf{x}, \theta)) \tag{2}$$

The MSE for a single model θ_k is simply the expected difference between the true and estimated probabilities squared.

$$\begin{aligned}\mathrm{Error}_{\theta_k} &= \Sigma_{\mathbf{x},y} P(\mathbf{x}, y)(P(y|\mathbf{x}_r) - P(y|\mathbf{x}, \theta_k))^2 \\ &= E_{P(\mathbf{x},y)}[P(y|\mathbf{x}_r)^2 - 2P(y|\mathbf{x}_r)P(y|\mathbf{x}, \theta_k) + P(y|\mathbf{x}, \theta_k)^2] \qquad (3)\end{aligned}$$

In the above equation, the sum over which joint distribution, either the unbiased joint distribution $P(\mathbf{x}, y)$ or possibly biased distribution of the next year $P(\mathbf{x}, y, s = \text{next year})$, is insignificant. The *expected* MSE, if the model is chosen at random from a model space Θ, is then the same as Eq. 3 except that there is an additional term $P(\theta)$ in the expectation.

$$\begin{aligned}\mathrm{Error}_{\mathrm{SingleModel}} &= \Sigma_{\theta_k} \Sigma_{\mathbf{x},y} P(\mathbf{x}, y) \times (P(y|\mathbf{x}_r) - P(y|\mathbf{x}, \theta_k))^2 \\ &= E_{P(\theta),P(\mathbf{x},y)}[P(y|\mathbf{x}_r)^2 - 2P(y|\mathbf{x}_r)P(y|\mathbf{x}, \theta_k) + P(y|\mathbf{x}, \theta_k)^2]\end{aligned}$$

If we were to take T models at random from the model space and average their predictions, then the expected performance would be

$$\begin{aligned}\mathrm{Err}_{\mathrm{MA}} &= \Sigma_{\mathbf{x},y} P(\mathbf{x}, y)(P(y|\mathbf{x}_r) - E_{P(\theta)}[P(y|\mathbf{x}, \theta)])^2 \qquad (4)\\ &= E_{P(\mathbf{x},y)}[P(y|\mathbf{x}_r)^2 - 2P(y|\mathbf{x}_r)E_{P(\theta)}[P(y|\mathbf{x}, \theta)] + E_{P(\theta)}[P(y|\mathbf{x}, \theta)]^2] \\ &\le E_{P(\mathbf{x},y)}[P(y|\mathbf{x}_r)^2 - 2P(y|\mathbf{x}_r)E_{P(\theta)}[P(y|\mathbf{x}, \theta)] + E_{P(\theta)}[P(y|\mathbf{x}, \theta)^2]] \\ &\le \mathrm{Error}_{\mathrm{SingleModel}} \text{as } E[f(x)]^2 \le E[f(x)^2] \qquad (5)\end{aligned}$$

Therefore, on average, conditional probability averaging would perform no worse than a single model, if many repeat experiments comparing conditional probability averaging against a single model were conducted.

3.3 Predicting on the future data with feature selection bias

When there is feature sample selection bias and the testing data is completely withheld from the training process, in other words, both feature vector values and class label are not known in advance,[1] an algorithm's performance estimated from the procedure described below and illustrated in Fig. 1 is expected to be very similar to its actual performance on the testing data.

[1] This is a different situation from [5]. In [5], the feature vector is known but not the class label. In our case, neither the feature vector nor the class label is known.

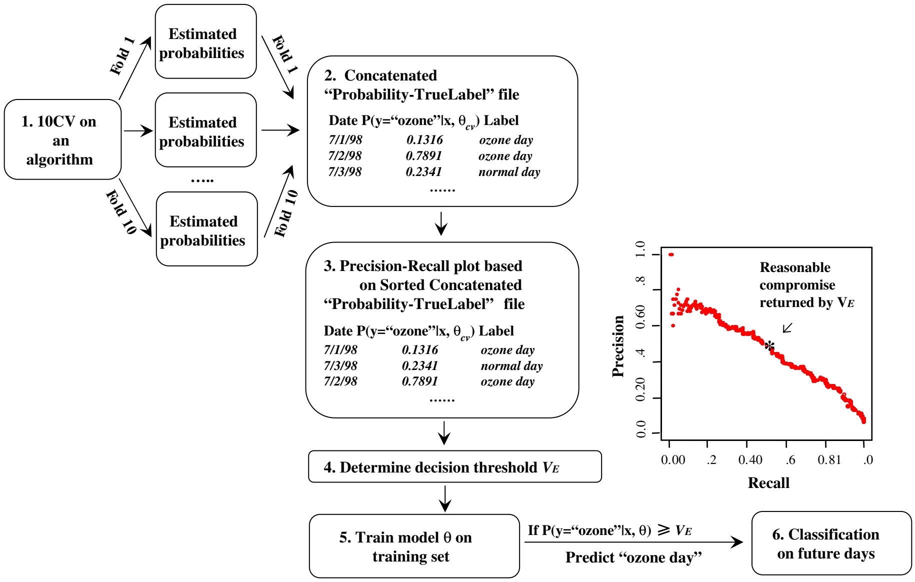

1. 10CV on an algorithm
Fold 1
Fold 10
Estimated probabilities
Estimated probabilities
.....
Estimated probabilities
Fold 1
Fold 10
2. Concatenated "Probability-TrueLabel" file
Date P(y="ozone"|x, θ_cv) Label
7/1/98 0.1316 ozone day
7/2/98 0.7891 ozone day
7/3/98 0.2341 normal day
......
3. Precision-Recall plot based on Sorted Concatenated "Probability-TrueLabel" file
Date P(y="ozone"|x, θ_cv) Label
7/1/98 0.1316 ozone day
7/3/98 0.2341 normal day
7/2/98 0.7891 ozone day
......
4. Determine decision threshold VE
5. Train model θ on training set
If P(y="ozone"|x, θ) ⩾ VE
Predict "ozone day"
6. Classification on future days
Precision
0.0
0.20
.4
0.60
.8
1.0
0.00
.2
0.40
.6
0.81
.0
Recall
Reasonable compromise returned by VE

Fig. 1 The cross-validation based procedure for decision threshold determination and prediction when there is feature sample selection bias

1. Use ten-fold cross-validation on an algorithm to be considered. Assume the obtained model is θ_{cv}, the estimated probability can be represented as $P(y = \text{"ozone"}|\mathbf{x}, \theta_{cv})$ in Fig. 1.
2. Concatenate the estimated probability values from ten separate testing files into a single "probability-true label" file. As shown in step 2 of Fig. 1, each entry in the file corresponds to one example, and it includes the estimated probability using ten-fold CV and the true class label for that example. For instance, an entry would look like (0.7891, "ozone day") or (0.2341, "normal day").
3. Plot either precision-recall or ROC plot by choosing "unique" values of probabilities to be "decision threshold" and applying the decision threshold on the "probability-true label" file to compute a pair of precision and recall to be plotted. An obtained example precision-recall curve is demonstrated in Fig. 1. A straight-forward implementation for this step is as follows:

    (a) Sort the "probability-true label" file into one with increasing estimated probability.
    (b) Choose each "unique" estimated probability as "decision threshold", one by one from the sorted file. Let v be this chosen decision threshold.
    (c) If the estimated probability for an example in the probability-true label file is more than a threshold v, predict it as an "ozone day", otherwise, predict it as a "normal day". The choice of v is discussed next.
    (d) For the sorted "probability-true label" file, use examples whose estimated probability equal to v as the "borderline". Then, every entry below this borderline will be labeled as "normal day" and every entry above and on this border line will be labeled as "ozone day".
    (e) Calculate the paired precision and recall corresponding to v, then plot recall and decision threshold v on the x-axis, and resulting precision on the y-axis. This can be either on the same plot or two separate plots.

4. By reading from the precision-threshold plot obtained above, choose a "decision threshold" v_E that returns a reasonable compromise between recall and precision. Typically, when v_E is high, precision is also high, but recall is low. This is especially true for the ozone day forecasting since the true model is stochastic. Assume that as a result of v_E, the recall and precision are r and p. Ideally, there should be many adjacent points with similar recall and precision values that are close to r and p. As explained below, this can effectively prevents "surprises". Therefore, it is safe to choose a value as the "decision threshold" v_E if it can return the reasonable precision-recall compromise denoted by asterisk on the precision-recall curve in Fig. 1.
5. Train a new model θ using all available examples.
6. Use θ to classify the future days. To be specific, the model predicts "ozone days" if the estimated probability by θ is more than v_E. Formally, predict "ozone day" if $P(y =$ "ozone"$|\mathbf{x}, \theta) \geq v_E$.

We next argue that the above process will return reasonable results for unseen future where neither feature vectors nor their true labels are known in advance. First, the precision-recall plot is constructed by cross-validation, and is still expected to be a reasonable estimate on the future testing data even under feature bias. Since neither the feature vector nor the prior ozone day probability for the coming year is known exactly, a better estimate would be quite hard to obtain. The cross-validation process already implicitly takes feature bias into account. Every test set in tenfold cross-validation contains a rather small number of examples considering that the problem contains 72 features and 2,500 + total number of examples. In other words, every test set is feature-biased, but just could be biased differently from the days in the coming year. Nonetheless, when this process is repeated ten times in cross-validation, the precision-recall plot is an average performance evaluated over ten different feature bias distributions. Unless the feature distribution of the coming year is so different from the training data, such as due to catastrophic weather, the precision-recall plot is expected to a reasonable prediction for the coming year.

Each point or a precision-recall point, on the plot is estimated from cross-validation. When these points appear rather continuous in the precision-recall plot, and not many empty spaces in between adjacent points, the chance for surprises when using model θ with decision threshold v_E on the future days is little. The simple reason is that a continuous curve has more "facts" instead of "speculations" than a discontinuous or broken plot. Therefore, as shown in Fig. 2, the algorithm which can generate continuous curve (M_a) is preferred over the one whose obtained curve is broken (M_b). In addition, one common held misconception about precision-recall curve as well as ROC plot is that they are infinitely continuous. This is not the case in reality. The simple understanding is that validation data is limited in size and many learning techniques have limited number of "unique" predictions. The line segment connecting adjacent points in precision-recall plot (or ROC plot) are not supported by any data. The connection is simply made for a visual line effect. For a chosen v_E as described above, if the adjacent precision-recall points are rather far away from the reasonable compromise returned by v_E, the chance of performance difference on future days is likely to be increased. For the future year's data, both the recall and precision given decision threshold v_E are unlikely exactly the same as r and p, as one would otherwise read from the plot. For reassurance, it is preferable to have an idea on the approximate value if the estimate is indeed off. Obviously, a continuous precision-recall plot can provide this information. On the contrary, for a broken or discontinuous plot, one could only speculate and assume that the "visual lines" are close to reality. For this reason, when choosing v_E, one should also

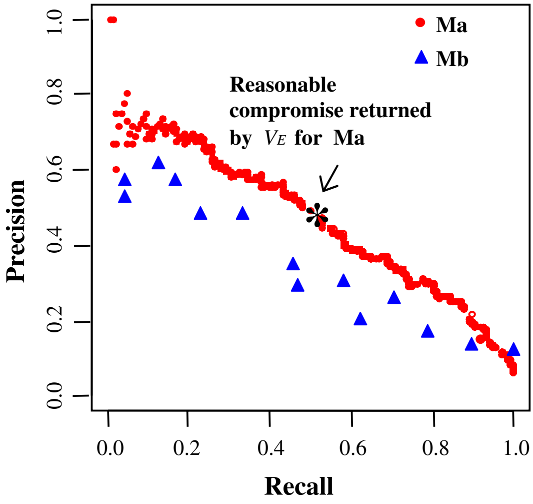


Fig. 2 Precision-recall curves: continuous versus broken, illustrating that, when the determined decision threshold(v_E) is applied to future unseen examples, the algorithm which can generate continuous curve (M_a) will receive less surprise than that whose obtained curve is broken (M_b). Please refer to Sect. 3.3 for explanation

consider adjacent points. When two algorithms construct similar shaped precision-recall plot, one should prefer the algorithm with a more "continuous" plot for the same reason.

4 Probabilistic tree models

The probabilistic decision tree models are suitable for the skewed ozone modeling and forecasting because they output "semi-continuous" probabilities and a decision-threshold can be chosen to find the most satisfactory compromise between recall and precision. There are two types of probability estimation trees — a single tree estimator and an ensemble of trees. Obviously, the single tree is better for comprehensibility. However, it only approximates the true unknown model from one particular unstable model representation [11] and may not have high accuracy. The ensemble is preferred when accurate probability estimation is desired. C4.5 and C4.4 [20] are representatives of single tree, and bagging trees and random decision tree [6] are examples of tree ensembles. It is important to point out that bagging trees in our study average over posterior probability, but seminal work on bagging uses voting of class labels.

C4.4 A straightforward method to estimate class membership probability is to use the class frequency at the corresponding leaf. In our case, this is achieved by dividing the number of high ozone days by total number of days from the training data that are sorted into the classifying leaf node. It has been noted (Provost and Domingos 2003) that frequency-based estimates of class-membership probability are not always accurate and statistically reliable. One reason is that the tree-growing algorithm searches for pure leaves, and tends to produce unrealistically high probability estimates, especially when the leaves cover few training examples. These estimates can be smoothed to mitigate these problems. As one of the simplest forms of smoothing, Laplace correction incorporates an equal prior for each class. Various experiments have showed that smoothing techniques can generally improve performance [7,20]. Importantly, Provost et al [20] pointed out that many heuristics for improving classification accuracy and minimizing tree size actually are biased "against" estimating accurate probabilities. For example, pruning via error reduction is blamed as the culprit. In summary, C4.4 is a variation of C4.5 by replacing frequency estimation with Laplace correction and turning off pruning and collapsing.

Random decision trees Random decision tree [6] is an ensemble of individual trees, each of which is constructed "randomly". To build each tree, a feature is randomly selected from remaining features on which the data is split on. Along a decision path, a discrete feature can be selected only once; however, continuous features can be used multiple times, but each time with a different, randomly chosen splitting value. Features from the training data are used to construct tree structures, and the data values themselves are used to update the class probabilities recorded in the leaf nodes. For a testing example, each tree produces a class probability. Probabilities from all the trees in the ensemble are averaged to generate the overall class probability estimate. Normally, 30 trees is sufficient, and up to 50 trees may be necessary when the distribution is skewed. On average, the depth of a random tree is about half of the number of the features, whereby distinct features are most likely to be selected to maximize diversity for the ensemble.

5 Experimental studies

Probabilistic decision trees are inductive methods that are constructed from labeled training examples and are affected by both sample selection bias (feature bias in our case) and amount of training data. However, parametric model used by TCEQ is not trained, but simply calculates the ozone level using some features for the day and is therefore not affected by either sample selection bias or amount of training data. Given these differences, both "exhaustive" cross-validation tests that ignore the time stamp of the data (therefore alleviating problem of sample selection bias and number of training examples), and incremental tests that build models from previous years and test on subsequent years are necessary. In particular, cross-validation tests are important from algorithmic point of view, since it will demonstrate the levels of accuracy that can be achieved when there are reasonable number of training examples and not really biased. On the other hand, incremental tests can help reveal the limit of each algorithm when the number of examples are limited and, most important of all, simulate the actual deployment of each method in practice.

5.1 Cross-validation experiments

In tenfold cross-validation experiments, the time stamp of each day is omitted, and the 7 years worth of ozone data is completely shuffled prior to generating training and testing pairs. Seven decision tree algorithms are applied on the data for both 1- and 8-h peak detection. These include baggingc4.5, baggingc4.4, random decision tree with half depth rdh, and full depth rdf, C4.4 as well as C4.5 with and without pruning. Each tree model computes a posterior probability $P(y = \text{“ozone day”}|\mathbf{x}, \theta)$. This notation explicitly specifies its dependency on the feature vector $\mathbf{x}$ for each day as well as some decision tree model θ. It is important to understand that the dependency on θ cannot be ignored since each model estimates probability based on its trained model and it may not be the true probability $P(y = \text{“ozone day”}|\mathbf{x})$. We choose a subjective decision threshold v_E, and whenever $P(y = \text{“ozone day”}|\mathbf{x}, \theta) \geq v_E$, we issue an alert. Obviously, with different values of v_E, different recall and precision will result. Normally with decreasing v_E, recall increases but precision tends to decrease. We say that model θ_a is preferred over θ_b, if the precision of θ_a is consistently higher than θ_b under the same recall numbers. It is important to observe how the recall and precision are correlated for each chosen algorithm. The resulting recall and precision results are plotted in a "recall-precision" chart with x-axis as the recall and y-axis as the precision. This is similar to ROC, but is more straightforward for meteorologists.

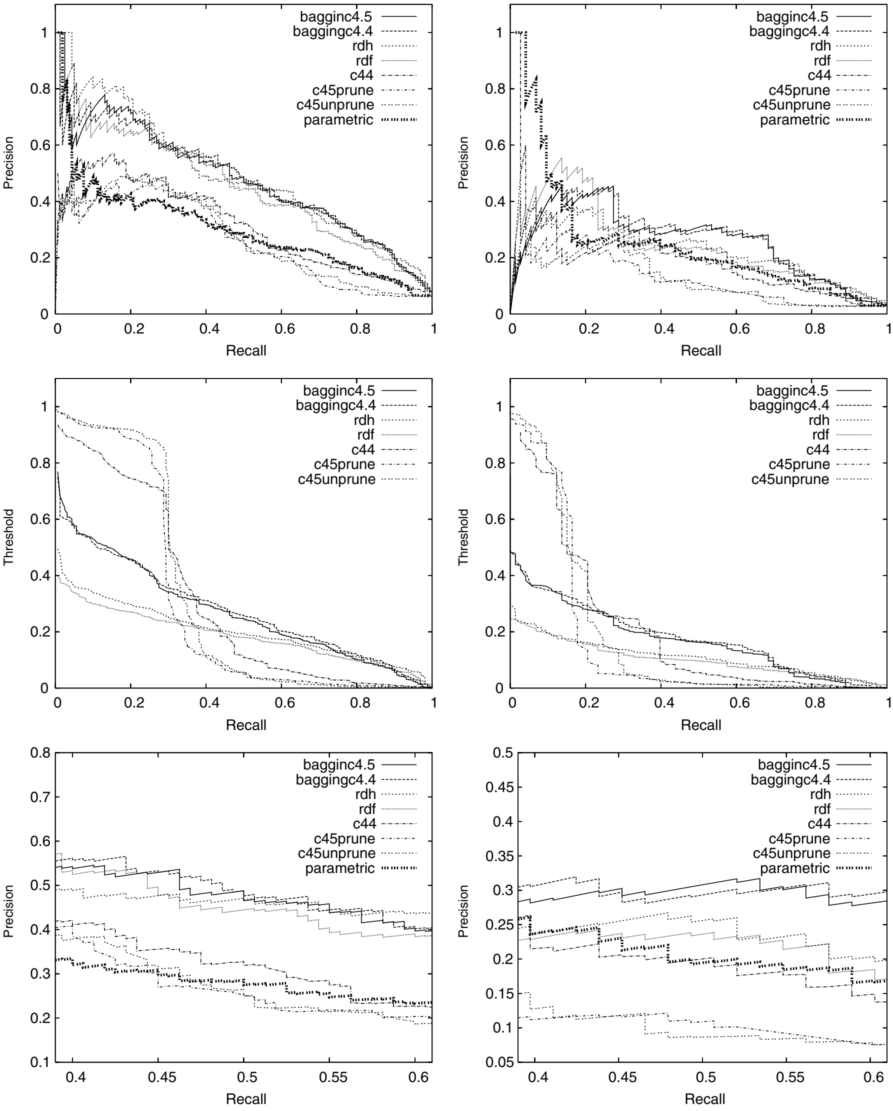


Fig. 3 Ozone alert prediction. *Left* 8-h; *Right* 1-h

Each probabilistic decision tree algorithm does not generate exactly "continuous" probabilities, but "semi-continuous" estimates. For example, the number of "unique" probabilities generated by C4.4 and C4.5 cannot be more than the number of leaf nodes. For bagging and random decision trees, the number of unique probabilities cannot be more than the multiple of leaf nodes of all trees in the ensemble. In reality, each tree is correlated in certain degree, the actual number on a given test set is expected to be smaller than this upper bound. In order to compute an exhaustive recall-precision plot, we choose the unique probabilities (as a result of a model tested on the test set) as the decision thresholds. The results by concatenating tenfolds together are shown in Fig. 3. For both 8- and 1-h forecast, we present the results

Table 1 Precision-recall coverage CV=10

8-Hour				1-Hour			
Bagging c4.4	Bagging c4.5	rdf	rdh	Bagging c4.4	Bagging c4.5	rdf	rdh
0.475	0.477	0.453	0.495	0.262	0.254	0.238	0.223
c4.4	c4.5 prune	c4.5 unprune	Parametric	c4.4	c4.5 prune	c4.5 unprune	Parametric
0.316	0.27	0.258	0.301	0.2	0.123	0.12	0.252

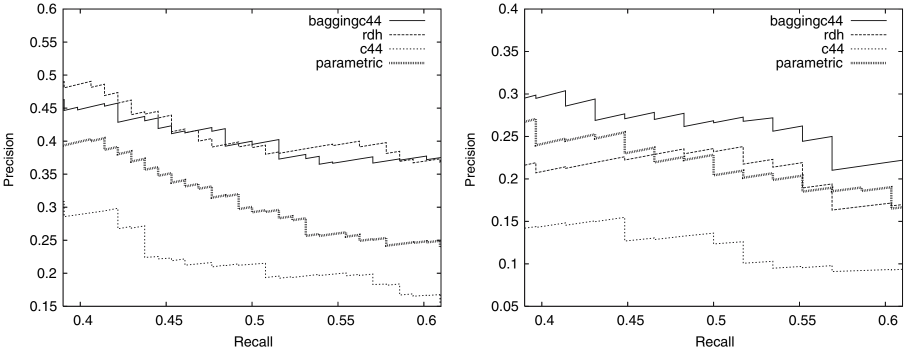
in full recall range, between 0.4 and 0.6 where both the recall and precision numbers are useful to cover most alerts with decent number of false alarms, as well as the unique decision thresholds for each decision tree. The scale on precision or y-axis is adjusted accordingly to emphasize the difference in results among different models. In all four plots, the baseline "parametric" model is the "darkest" curve. Clearly, the precision obtained by the ensemble approaches (baggingc4.5, baggingc4.4, rdh, and rdf) for the same recall numbers are nearly all higher than the baseline parametric approach, and consistently higher in the useful critical recall range of [0.4, 0.6], for both 1-h standard and 8-h standard. Importantly, the higher precision obtained by trees are particularly obvious for the newly enacted 8-h long duration standard. As shown in the left plots of Fig. 3, each of the four ensemble methods has achieved twice as much precision as the parametric model, and single tree method C4.4 has obtained higher precision for recall range [0.4, 0.6]. The difference among ensemble methods appears to be "twisted" and insignificant in the full recall range. In the critical range, rdh and baggingc4.4 appear to have achieved slightly higher precision than the other methods. For the 1-h exposure standard, as shown in the right plots of Fig. 3, in decreasing order of precision for recall range [0.4,0.6], the top performers are approximately, baggingc4.4, baggingc4.5, rdh, rdf, followed by the parametric model.

In addition, as a summary of each curve in the precision-recall plot, we measure the "coverage" area or integration under each curve, similar to AUC in ROC, as a single number to compare different models. The normalized area for each modeling technique over the full range is summarized in Table 1. Since a high coverage implies better performance, the order of performance among different models is in consensus with the visual observation.

As shown in the recall-threshold plots in Fig. 3, the threshold curves for the three single tree methods, C4.5prune, C4.5unprune, and C4.4, all exhibit an s-shape, and are either quite high (close to 1.0) or quite low (close to 0.0). This is due to the fact that single trees tends to construct pure nodes that are mostly ozone days or normal days. It appears that the Laplace correction employed by C4.4 can make the curve "flatter", but it is limited. The threshold curves of random decision trees are consistently lower than those of bagging, and this is due to bagging's use of information gain to choose feature which results in "purer" nodes than random trees.

5.2 Incremental study

Incremental study is necessary, since in reality, the dataset is not collected all at once, but on a day-by-day basis. We have chosen three algorithms for this study, baggingc4.4, random decision tree with half depth, and C4.4, as they are the best performers for ensemble methods or single trees in the cross-validation tests.



Eight-Hour				One-Hour			
baggingc4.4	rdh	c4.4	parametric	baggingc4.4	rdh	c4.4	parametric
0.079	0.081	0.045	0.061	0.051	0.041	0.022	0.042

Fig. 4 Monthly test precision-recall plot and coverage with recall=[0.4,0.6]. *Left* 8-h; *Right* 1-h

Month by month exhaustive test In a month-by-month test, we incrementally include the previous month's data to train a new model in order to make predictions for the coming month. We start with the whole year's data in 1998, the incremental tests start on January 1999, and terminates in December 2004. For each test (1-h or 8-h), there is a total of 72 of training and test pairs. To compare with cross-validation results, we concatenate the estimated probabilities from 72 tests into a single file, choose unique probability values as decision thresholds to plot recall-precision curves for recall range [0.4,0.6] and its corresponding coverage measurement, as shown in Fig. 4. Comparing with exhaustive cross-validation test results in Fig. 3, the parametric model's performance is very similar but not the same, since it is not tested on the first year's data. The performance of baggingc4.5 and rdh is slightly worse than the exhaustive cross-validation tests since the training sets are significantly smaller, i.e., incrementally from 12 to 83 months, as compared to fixed training data of $\frac{84}{10} \times 9 = 75.6$ months. However, they are still significantly better than the parametric model as summarized by the coverage measures in Fig. 4. To be specific, for 8-h test, rdh and baggingc4.4 achieve precision level 10–20% higher than the parametric model for different recall values. For the 1-h test, baggingc4.4's precision is consistently about 5% higher, but rdh and parametric model are very similar in precision. Compared with the two ensemble approaches, C4.4's performance appears to be significantly worse than the exhaustive test, and consistently worse than the parametric model used by TCEQ. The reason is that single decision tree methods are sensitive to the amount of training data. When the amount is less, single tree's error due to variance increment becomes quite large. However, ensemble methods like bagging and rdt can effectively reduce variance even if the training set size is small. This has been recently demonstrated in [24].

Annual test The annual incremental test simulates the realistic scenario that meteorologists would use the probabilistic model. The reason is that the number of ozone days per year for both 1- and 8-h peak is rather skewed and mostly accumulated during the summer months. Incremental learning "finer" than 1 year increment is unlikely to include meaningful number of examples that could otherwise improve the model trained previously. For annual incremental learning, we start by building models from 1998s data to predict on 1999, incrementally assimilate the previous year's data and reconstruct the model to predict on the coming year, till 2004, for a total of 6 pairs of tests.

Table 2 Incremental annual test for 8-h and 1-h

	Bagging c4.4			rdh			c4.4			Parametric
8-h Annual										
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	80
R	0.308	**0.554**	0.429	0.384	**0.608**	0.488	0.378	0.514	0.382	0.568
P	0.425	**0.348**	0.416	0.350	**0.323**	0.358	0.199	0.162	0.183	0.227
1-h Annual										
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	120
R	0.382	**0.692**	**0.522**	0.392	0.545	**0.501**	0.316	0.540	0.340	0.476
P	0.197	**0.155**	**0.185**	0.174	0.121	**0.154**	0.125	0.115	0.110	0.158

T threshold; *R* recall; *P* precision

Like parametric model, we need to "fix a decision threshold" to make prediction on a daily basis. The threshold for the parametric model used by meteorologists is 120 for 1-h peak and 80 for 8-h peak, and the resulting recall is between 40 and 60%. We are interested in estimating a decision threshold for each algorithm to obtain similar levels of recall, and then comparing the resulting precisions (the higher the better). For this reason, tenfold cross validation is applied on the "annual" training set to determine the thresholds used for the next year. Three thresholds are selected for this purpose, and they are the decision thresholds for recall = 0.4, recall = 0.6, and the average of these two thresholds. For example, on December 31, 2000, the training data contains 1998, 1999 and 2000. In order to decide the decision threshold for the year 2001, we use tenfold cross validation on the training data of 1998, 1999 and 2000. We concatenate all ten result files generated from cross-validation together, and respectively find out the decision threshold for recall to be 0.4, 0.6 and the average of these two thresholds. This procedure is repeated for each of the three chosen decision tree algorithms, and a different set of decision thresholds is selected for each model. We then use the respective model with the chosen thresholds to predict the year 2001, and report the corresponding precision and recall. Since the collected data is from 1998 to 2004, this procedure is repeated six times. The results averaged for 6 runs are shown in Table 2. The "threshold" for decision trees is marked under "0.40", "0.60" and "avg". These are not the actual thresholds chosen for each method. It just indicates the thresholds chosen from cross-validation to obtain recall to be 0.4, 0.6 and the average of these two thresholds. We bold-font a result if both of recall and precision are higher or rather close to the parametric model. The advantages of baggingc4.4 and rdh over the parametric model are obvious. For the 8-h alert, comparing rdh (recall = 0.608, precision = 0.323) and parametric (recall = 0.568 and precision = 0.227), assuming that each year has about 25 8-h ozone alert days, the result means that rdt can correctly detect 1 more day but issue 16 days fewer false alarms. For meteorologists, this is significant.

Ozone month only annual test One important observation is that high ozone days only happen during warm days and never occur during winter time. For this reason, it is a reasonable conjecture that those months without any ozone days are unlikely to help to construct a model to detect ozone days for summer months. We design "an ozone month only annual test". The difference from the previously described annual test is that we only use the previous years' "ozone month" data to make predictions for current year's ozone month data. We define ozone month as the month which includes at least a day whose ozone level is higher than or equal to 80 for 8-h peak and 120 for 1-h peak. The total number of months of out

Table 3 Ozone month only annual test: 8-h

	Bagging c4.4			rdh			c4.4			Parametric
T	0.70	0.80	Avg	0.70	0.80	Avg	0.70	0.80	Avg	80
R	**0.654**	**0.728**	**0.696**	**0.662**	**0.770**	**0.718**	0.689	0.762	0.695	0.568
P	**0.329**	**0.270**	**0.293**	**0.301**	**0.246**	**0.271**	0.173	0.157	0.168	0.237

T threshold; *R* recall; *P* precision

$12 \times 7 = 84$ months has 46 months with 8-h peak and 33 months with 1-h peak. When the percentage of ozone days increases or the percentage of positives increases, it becomes a slight different problem. As a result, the decision tree algorithms construct "conceptually" different models as compared to previous experiments, and the recall/precision as a function of decision threshold over different models also changes. Using cross-validation test, we have found that with recall between 0.7 and 0.8, the precision by decision trees are still higher than the parametric model for the same year. For this reason, in the ozone month annual test, the decision threshold chosen for the next year are the ones that obtain recall=0.7, 0.8, and the average of the two thresholds. The result is shown in Table 3. It is important to understand that the recall for parametric model (not the tree model) is the same as previous annual test, however, the precision is slightly higher. This is because the none-ozone months have been taken out. In Table 3, when the decision tree's methods achieve both higher recall and higher precision, the results are highlighted in bold-fonts. Obviously, for each choice of decision thresholds, both baggingc4.4 and rdt have achieved 10–20% higher recall with up to 10% higher precision.

6 Extensive comparison with other learners

Besides solving a practically difficult problem, one important task of this paper is to demonstrate that straight-forward non-parametric method, particularly bagging decision trees and random decision trees, can provide significantly more accurate solutions than some other more sophisticated methods to this ozone problem and potentially other problems with similar characteristics (such as skewed distribution, stochastic, possibly many irrelevant features and sample selection bias). In particular, we plan to show that, as compared to some more sophisticated methods discussed below, random decision tree and bagging decision trees can really provide reliable, expected or "non-surprisingly different" results when the testing data follows a different distribution due to sample selection bias in the future. For these reasons, we have included following methods in our study since they are the most obvious choices that many people would consider: logistic regression (LR), CFT or confusion factor tree [13], SVM [2] with both linear and RBF kernels, naive Bayes (NB), as well as AdaBoost C4.5 and NB in both re-sampling and re-weighting implementations.

Logistic regression (LR): Typically, logistic regression is described as

$$P(y = \text{“ozone day”}|\mathbf{x}) = \frac{1}{1 + \exp(\beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n)}$$

In other words, it assumes that the true distribution $P(y|\mathbf{x})$ follows a particular parametric form based on sigmoid function. Obviously, the success of LR is dependent on the appropriateness of "sigmoid" to match the true known distribution.

Naive bayes (NB): As a "naive" simplification of the Bayes rule, the naive Bayes algorithm assumes that all features x_i's are mutually independent, given the class variable y, or $P(x_i x_j|y) = P(x_i|y)P(x_j|y)$. Consequently, naive Bayes (Mitchell Tom 1997) assigns a test example $\mathbf{x} = (x_1, \ldots, x_k)$ to the class y with the highest $P(y|x_1, \ldots, x_k) = P(y) \prod P(x_i|y)$. In spite of its naive design and apparently over-simplified assumptions, naive Bayes classifiers often work well in many complex real-world situations.

Support vector machines (SVM): Given m examples, the soft-margin SVM seeks the solution to the following optimization problem

$$\min_{w,b,\epsilon} \left(\frac{1}{2} w^T w + C \sum_{1}^{m} \epsilon_i \right)$$

subject to

$$y_i \cdot (w^T \theta(\mathbf{x}_i) + b) \geq 1 - \epsilon_i$$

In which, $K(\mathbf{x}_i, \mathbf{x}_j) = \theta(\mathbf{x}_i) \cdot \theta(\mathbf{x}_j)$ is known as the "kernel function", ϵ's are "slack variables" that measure prediction errors on the training data and C is the penalty parameter of the error term. Both w's and b are variables that define the hyperplane. When the training examples are on the wrong side of the computed hyperplane, the corresponding ϵ for that example is greater than 1, thus $\sum \epsilon_i$ is an upper bound of the training errors.

We excluded original SVM (i.e. hard-margin SVM) from the comparison since the soft-margin version is more robust to cope with potentially inseparable or noisy examples. These examples are typically ubiquitous and inevitable for real-world scientific datasets, such as this ozone level prediction problem. In our experiment, software package LibSVM (Chang and Lin 2001) is used for soft-margin SVM implementation. The libSVM package has several kernel functions available, and we have chosen to use the radial basis functions (RBF) $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \cdot \|\mathbf{x}_i - \mathbf{x}_j\|)$ and $\gamma > 0$, as well as the linear kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \cdot \mathbf{x}_j$. To ensure the best performance of the obtained SVM model, we first linearly scale each attribute to the range [−1,+1], then tenfold CV is applied on the given training set to estimate the best values for γ and C before the resulting model is used for posterior probability estimation on the "unseen" test set.

Confusion factor tree (CFT): Aiming at improving the AUC value, Ling et al propose the "confusion factor tree"(CFT) by assigning a fixed confusion factor s to each internal node of the tree. This value measures the probability of errors altering the value of a tested attribute at a decision node due to noise introduced in data collection. Therefore, the probabilities from all other leaves as well as the probability of the leaf into which an example falls will contribute to the final probability estimate of the example. The contribution is determined by the number of unequal attribute values along the path leading to the leaf. Thus, the final class probability estimate for $\mathbf{x}$ is a weighted average of the contributions from all of the leaves in a tree:

$$P(y|\mathbf{x}) = \frac{P_i \times s^q}{\sum s^q}$$

where P_i is the probability at leaf i and q is the number of unequal attribute values. As suggested in [13], with Laplace correction and the confusion factor set to 0.3, the pruned variation is used in the experiment.

AdaBoost NB and C4.5: AdaBoost is the most representative boosting method. It trains multiple classifiers with the same inductive algorithm (such as decision tree or naive Bayes) in different iterations. At each iteration, it increases the weight of those misclassified examples while reducing the weight of those correctly classified, according to the following formula:

$$D_{t+1}(i) = \frac{D_t(i)\exp(-\alpha_t y_i h_t(\mathbf{x}_i))}{Z_t}$$

where $D_{t+1}(i)$ is the weight for $(\mathbf{x}, y)_i$ at iteration $t+1$, $h_t(\mathbf{x})$ is the hypothesis constructed at iteration t, and Z is a normalization factor $Z_t = \sum D_t(i)$. The final hypothesis is a weighted voting ensemble, $H(\mathbf{x}) = \text{sign}(\sum \alpha_j h_j(\mathbf{x}))$. It has been proven that in order to minimize the training error, one ought to choose α_t at each iteration to minimize the normalization factor Z_t. There are two implementations of AdaBoost, one uses re-sampling to simulate the change of weights and the other directly uses re-weighting. In our experiments, we have used both implementations on top of C4.5 and NB.

6.1 Expectation: cross validation results

Since the annual incremental test most closely simulates the real-world scenario in which the constructed models would be used in practice, all the algorithms are compared against each other using the same experimental and tenfold CV threshold determination procedure as described in “Annual Test” of Sect. 5.2. There are several important points that one wishes to find out. First, it is useful to find out if any of these methods can significantly improve the accuracy of straight-forward applications of baggingc4.4 and rdh. Second due to sample selection bias that the past years may not follow the same distribution as future years, it is also important to know if one could reliably choose a decision threshold based on past data with a targeted recall and precision results in mind, then receive similar performance when applied on future year’s unseen data.

The results by concatenating tenfolds of the first 6 years (1998–2003) are plotted in Figs. 5 and 6, for 8-h and 1-h standard, respectively. In each figure, ensemble methods and single models are plotted separately. In order to show their relative performance clearly, algorithms are approximately ranked from top to bottom in the legend, based on the precision values within the full recall range. For example, on the right plot of Fig. 5, among all legends, SVMRBF is at the top while NB is at the bottom, this corresponds to their relative

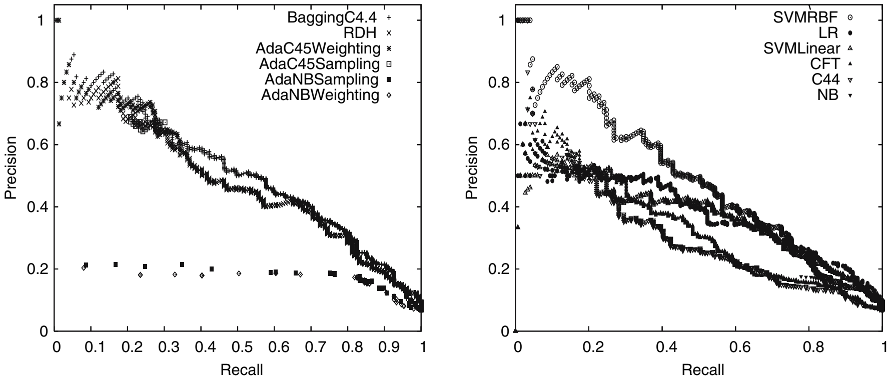


Fig. 5 Ten-fold CV on the first 6 years’ data. 8-h: ensemble vs. single

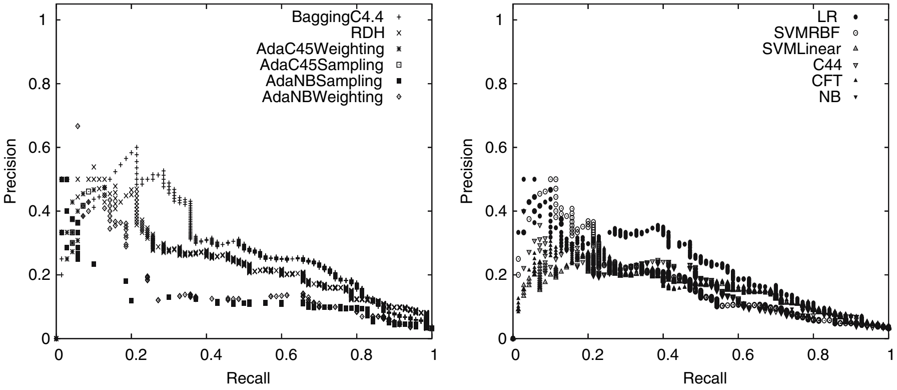


Fig. 6 Ten-fold CV on the first 6 years' data. 1-h: ensemble vs. single

performance measured in precision. To better visualize each algorithm's performance in the precision-recall space, we "decompose" Fig. 5 into Fig. 7, and Fig. 6 into Fig. 8 by plotting one or two models separately.

Results for ensembles Obviously, as shown in the left plots of Figs. 5 and 6, for both datasets, baggingc4.4 and rdh consistently achieve much higher precision scores than other ensemble algorithms, particularly, adaNB. The precision-recall curves of adaNB with both re-sampling (adaNBs) and re-weighting (adaNBw) are rather flat, with precision values around 0.195 for the 8-h set and 0.145 for the 1-h set. Low precision levels (and subsequent high false positive rates) for the entire recall range could limit the practicality of AdaBoost NB. In addition, as demonstrated in Figs. 7 and 8, the precision-recall curves of baggingc4.4 and rdh are significantly more continuous than those of AdaBoost ensembles, and this is true for both C4.5 and NB. For the 8-h criterion, almost all of the precision-recall points of adaC45 with re-weighting(adaC45w) or re-sampling(adaC45s) are "crowded" within the low recall range [0.1, 0.35]. Further, its resultant recall range "shrinks" down to [0, 0.1] for the one hour criterion. Obviously, recall at these low levels can potentially harm the usefulness of AdaBoost C4.5 on this problem. For AdaBoost NB, there are altogether 146 points in their 8-h plots (both re-sampling and re-weighting implementation). As these 146 discrete points are spread out within the [0, 1] range, one can choose any decision threshold/desirable result in between two adjacent points, since in reality there are no data and no information between adjacent points.

Results for single models For 8-h set, as shown in the right plot of Fig. 5, SVMRBF has the advantage over other single algorithms. Its precision-recall curve achieved by concatenating tenfolds of the training set are rather close to those of baggingc4.4 and rdh, as shown in Fig. 9. For 1-h criterion, as summarized in Fig. 6, LR clearly outperforms all other single models. Its precision-recall curve obtained by concatenating tenfolds of the training set appears visually "twistest" among those of baggingc4.4 and rdh, as demonstrated in the right plot of Fig. 9. At the same time, the differences of SVMs, CFT and C4.4 are insignificant and hard to distinguish. In addition, except for naive Bayes, as shown in Figs. 7 and 8, the precision-recall curves of SVMs, LR, CFT and C4.4 are rather continuous within the full recall range. For naive Bayes, in its 1-h plot as shown in the bottom of Fig. 8, there are 149 points scattered within the rightmost area with recall from 0.75 to 0.9 and precision from 0.07 to 0.1. This

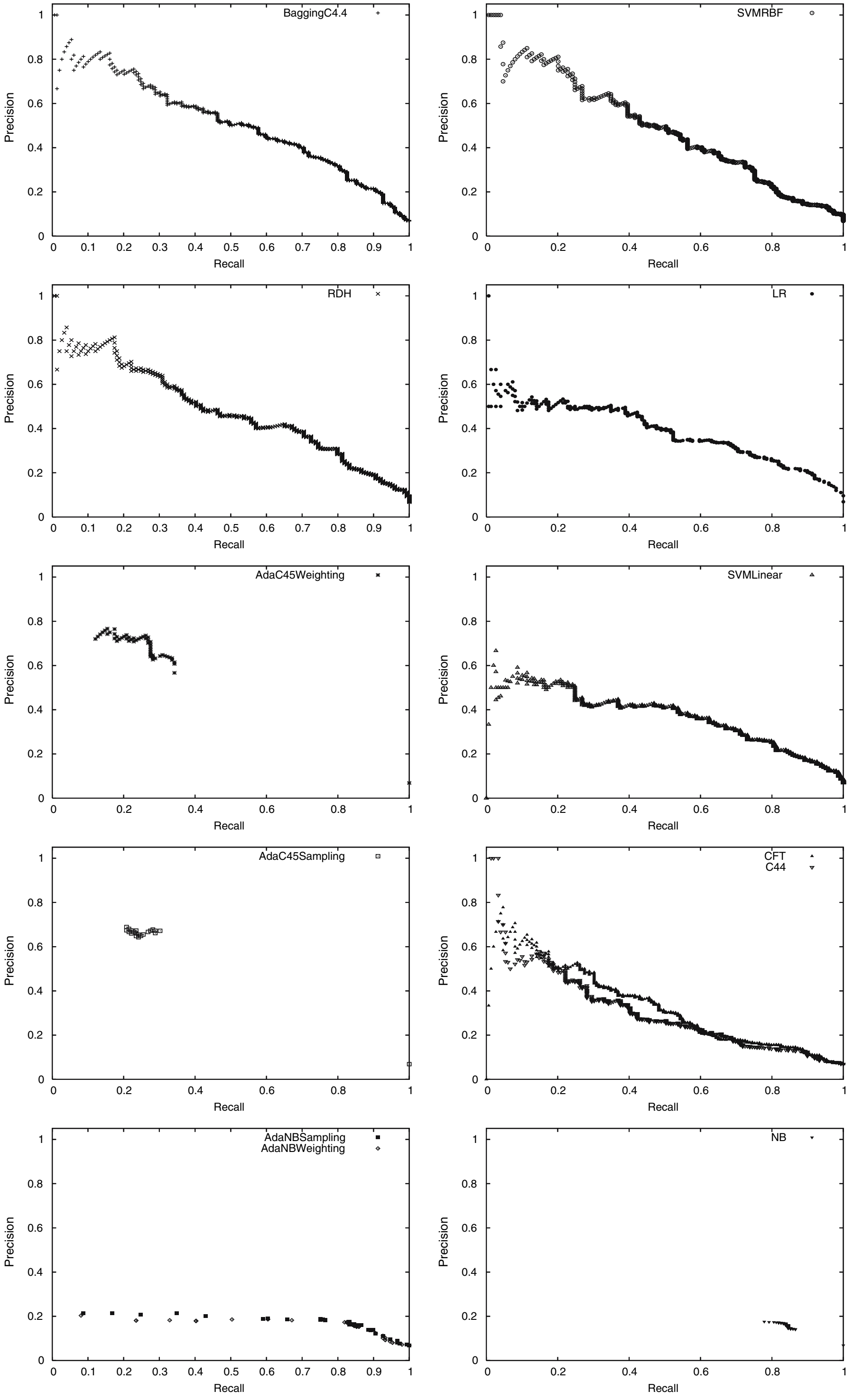


Fig. 7 Tenfold CV on the first 6 years' data: 8-h. *Left* each ensemble; *Right* each single

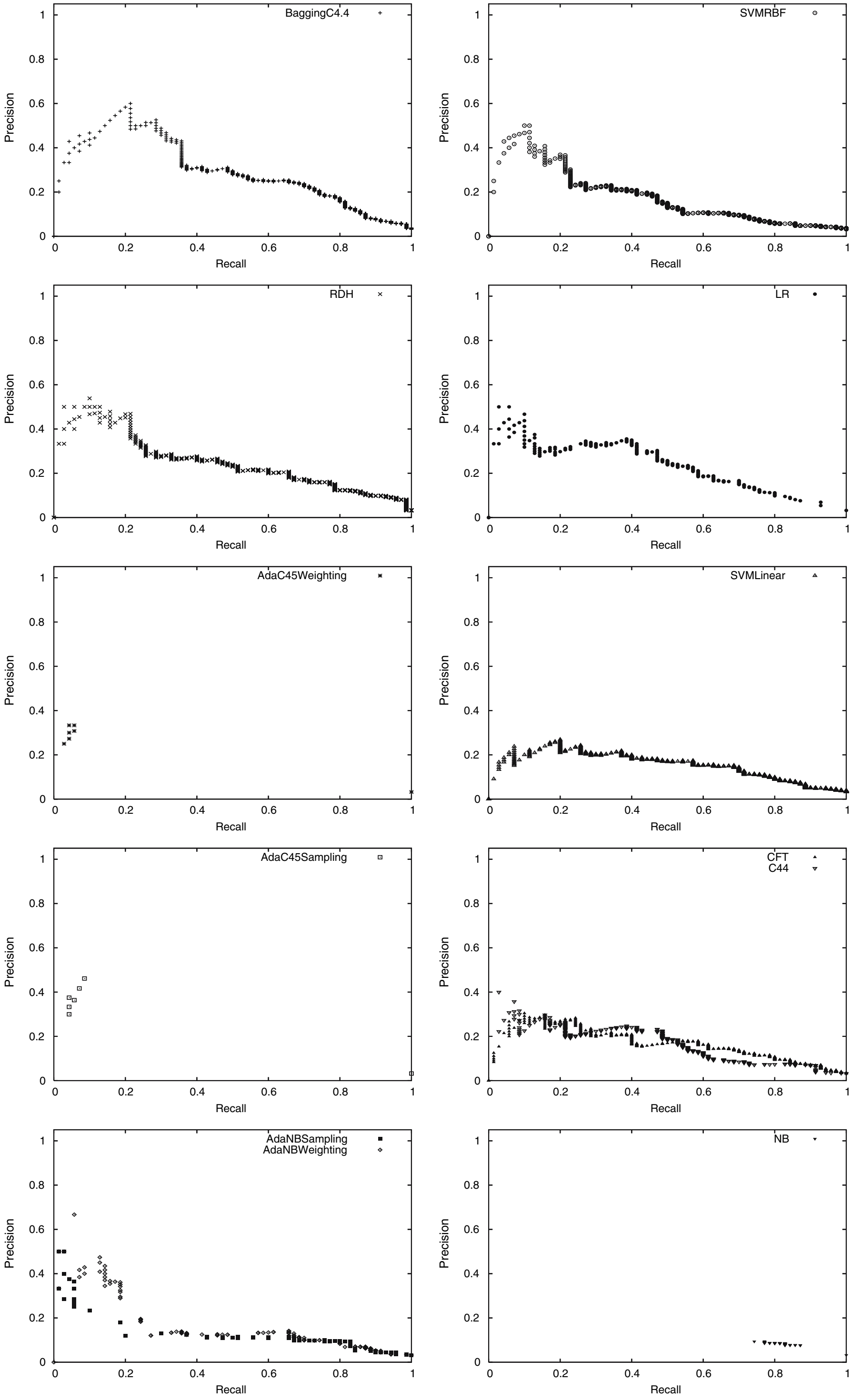


Fig. 8 Tenfold CV on the first 6 years' data: 1-h. *Left* each ensemble; *Right* each single

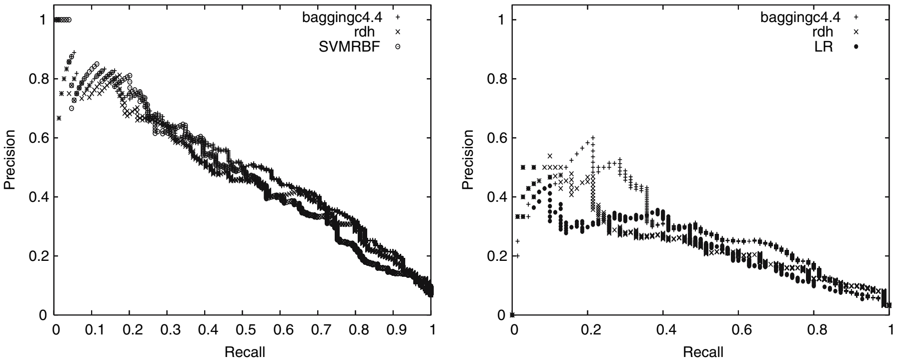


Fig. 9 Tenfold CV on the first 6 year's data. *Left* 8-h: SVMRBF versus baggingc4.4; *Right* 1-h: LR versus baggingc4.4 and rdh

"clustered" or "skewed" predictive result and low precision levels may potentially prevent NB's practicality.

6.2 Surprises: actual performance on future data

The above results demonstrate the performance difference among different learners when "cross-validated" on old training data. One would be very interested to find out how these learners actually perform when trained on old data but tested on future unseen data. In particular, one need to find out how these cross-validation results and selected decision thresholds to choose targeted precision-recall can actually "spell" the future.

Practical Recall Range [0.4, 0.6] Averaged over 6 years, the actual performances of these 12 learning algorithms on future data are summarized in Tables 4 and 5. Please note that there are no values (na) available for either adaC45 or NB since they have no precision-recall points within this recall range ([0.4,0.6]) in tenfold CV-based threshold determination. This can be seen from Figs. 7 and 8. Thus we exclude NB and adaC45 from the following discussion.

To identify the practicality for each method, we use the results of expert-based parametric model as the baseline reference. We highlight a method in bold-fonts if both its recall and precision are higher than or rather close to those of parametric model. Obviously, for both datasets, as compared to those algorithms just introduced, algorithms, baggingc4.4 and rdh are still the only learners whose overall performances, in terms of both recall and precision, outperform or are rather comparable to those of the parametric model. None of these algorithms, including fine-tuned SVMs, classical logistic regression and sophisticated AdaBoost, can achieve more decent pairs of recall and precision than these two probabilistic tree ensembles. In addition, for re-weighting and re-sampling implementations of adaNB, their predicted recall and precision values on both datasets are quite similar to their performances on the "cross-validated" training sets. With recalls in the targeted range, the mean value of predicted precision of both implementations is 0.167 for 8 h set, and 0.13 for 1-h forecast.

A couple of important points can be summarized from the above observations. First, at least among all of the inductive learning algorithms we have tested, bagging probabilistic trees and random decision trees offer the most significant and rather straight-forward solutions to this problem. Second, based on the decision thresholds selected through the "cross-validated" historical data, ensemble learners, such as these two probabilistic tree ensembles

Table 4 Incremental annual test for 8-h with all learners

	Bagging c4.4			rdh			adaNBs			Parametric
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	80
R	0.308	**0.554**	0.429	0.384	**0.608**	0.488	0.601	0.606	0.606	0.568
P	0.425	**0.348**	0.416	0.350	**0.323**	0.358	0.152	0.150	0.154	0.227
	adaNBw			SVM-RBF			SVM-Linear			Parametric
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	80
R	0.456	0.506	0.506	0.325	0.513	0.391	0.274	0.503	0.343	0.568
P	0.187	0.177	0.177	0.374	0.380	0.520	0.324	0.268	0.254	0.227
	c4.4			CFT			LR			Parametric
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	80
R	0.378	0.514	0.382	0.354	0.703	0.538	0.291	0.505	0.378	0.568
P	0.199	0.162	0.183	0.234	0.202	0.235	0.283	0.228	0.273	0.227
	adaC45s			adaC45w			NB			Parametric
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	80
R	na	na	na	na	na	na	na	na	na	0.568
P	na	na	na	na	na	na	na	na	na	0.227

T threshold; *R* recall; *P* precision

and AdaBoost implementations, can really provide the reliable and expected results for the future unseen examples. This is particularly true when the distributions of training and test sets do not match with each other.

In contrast, for single models, their performances on the future unseen data are quite different from what they have achieved on the historical training sets. This is typically the case for both SVMRBF and LR. For example, as showed in Fig. 9, the precision-recall curve of SVMRBF is indistinguishable from those of baggingc4.4 and rdh on 8-h training set. However, its best recall value is 15.6% lower than that of rdh for the same set forecast even if the obtained precision is quite decent. To thoroughly examine the efficacy of these three learners, we have run an additional test in which decision thresholds of SVMRBF are chosen through the "unseen" test data, and then applied on the same test examples for prediction. In a way, this provide the best possible results that could be obtained using the SVMRBF model. For 8-h forecast, the precision value achieved at recall to be 0.6 is 0.278. Comparing to rdh(recall $=$ 0.608, precision $=$ 0.323) and baggingc4.4(recall $=$ 0.554, precision $=$ 0.348), whose decision thresholds are selected from the "cross-validated" historical training data, this result provides extra empirical evidence to demonstrate the strength of these two probabilistic tree ensembles. Similar analysis is applied to logistic regression on the 1-h dataset.

Higher recall range [0.6, 0.8] As shown in Table 4, for the 8-h forecast, the recall value of SVMRBF achieved at targeted recall to be 0.6 is actually 0.513, which is 14.5% lower

Table 5 Incremental annual test for 1-h with all learners

	Bagging									Para-
	c4.4			rdh			adaNBs			metric
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	120
R	0.382	**0.692**	**0.522**	0.392	0.545	**0.501**	0.607	0.821	0.738	0.476
P	0.197	**0.155**	**0.185**	0.174	0.121	**0.154**	0.137	0.120	0.134	0.158
				SVM			SVM			Parametric
	adaNBw			-RBF			-Linear			
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	120
R	0.361	0.538	0.409	0.309	0.606	0.389	0.257	0.339	0.296	0.476
P	0.150	0.111	0.118	0.122	0.089	0.105	0.141	0.104	0.114	0.158
										Parametric
	c4.4			CFT			LR			
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	120
R	0.316	0.540	0.340	0.420	0.689	0.552	0.121	0.388	0.255	0.476
P	0.125	0.115	0.110	0.158	0.094	0.109	0.073	0.070	0.108	0.158
										Parametric
	adaC45s			adaC45w			NB			
T	0.40	0.60	Avg	0.40	0.60	Avg	0.40	0.60	Avg	120
R	na	na	na	na	na	na	na	na	na	0.476
P	na	na	na	na	na	na	na	na	na	0.158

T threshold; *R* recall; *P* precision

than the recall in mind. However, at the same time, the corresponding precision value is quite decent. This indicates that the standard useful recall range [0.4, 0.6] could cause the predicted recall of SVMRBF to be underestimated. Therefore, it would be reasonable to think that using a lower threshold to get a higher recall range might result in overall better precision-recall pairs for SVMRBF on the future. If so, it would be worthwhile to further investigate how this change could "challenge" other learners, particularly, baggingc4.4, rdh, LR as well as AdaBoost NB. For this reason, we use [0.6, 0.8] as the higher recall range, and the obtained results for above algorithms averaged over 6 runs are summarized in Table 6. Similarly, the results of a method are highlighted in bold-fonts if its both recall and precision are higher than those of parametric model. For the 8-h forecast, interestingly, baggingc4.4 and rdh continue to maintain their leading places, as compared to other models. As expected, SVMRBF achieves better results than the parametric method. However, its overall performances are still less than those of baggingc4.4 and rdh. For example, comparing rdh (recall = 0.717, precision = 0.294) and SVMRBF (recall = 0.633, precision = 0.282), still assuming that each year has about 25 8-h ozone alert days, the result means that rdh can correctly detect 2 more days but issue 3 days fewer false alarms than SVMRBF. For the 1-h forecast, baggingc4.4 is the only learner whose performance is comparable to that of parametric model. On the other hand, higher recall range obviously is not a good choice for LR and AdaBoost NB. Their precisions are more seriously impaired as a consequence.

Table 6 Incremental annual test for 8-h and 1-h with recall range [0.6, 0.8]

	Bagging c4.4			rdh			adaNBs			Parametric
8-h Annual										
T	0.60	0.80	Avg	0.60	0.80	Avg	0.60	0.80	Avg	80
R	0.560	**0.760**	**0.640**	**0.608**	**0.760**	**0.717**	0.706	0.818	0.744	0.568
P	0.346	**0.239**	**0.296**	**0.350**	**0.244**	**0.294**	0.165	0.152	0.164	0.237
	adaNBw			SVM -RBF			LR			Parametric
T	0.60	0.80	Avg	0.60	0.80	Avg	0.60	0.80	Avg	80
R	0.781	0.850	0.806	0.502	**0.633**	**0.587**	0.500	0.637	0.588	0.568
P	0.151	0.142	0.143	0.296	**0.282**	**0.263**	0.208	0.152	0.191	0.237
	Bagging c4.4			rdh			adaNBs			Parametric
1-h Annual										
T	0.60	0.80	Avg	0.60	0.80	Avg	0.60	0.80	Avg	120
R	**0.668**	0.725	0.713	0.545	0.772	0.645	0.800	0.857	0.809	0.476
P	**0.164**	0.093	0.132	0.124	0.095	0.115	0.126	0.072	0.082	0.158
	adaNBw			SVM -RBF			LR			Parametric
T	0.60	0.80	Avg	0.60	0.80	Avg	0.60	0.80	Avg	120
R	0.459	0.842	0.747	0.606	0.758	0.715	0.259	0.467	0.379	0.476
P	0.116	0.072	0.097	0.090	0.084	0.088	0.046	0.026	0.042	0.158

T threshold; *R* recall; *P* precision

7 Conclusion

Our work provides a data mining based solution to forecast ozone days for the Houston area, as well as experiences and guideline to solve problems with similar characteristics. Comparing to the existing method adopted by Texas Commission on Environmental Quality, our method is more accurate, with 20% higher recall and 10% higher precision.

On the practical side, ozone level forecast is one of the most important and difficult problems for air quality control. It is well established that ozone level above certain threshold is dangerous for human health and inadversely affects other parts of our daily life. Traditional approaches to forecast ozone alert relies on "air dynamics" that simulates the physical and chemical process that generate ozone. It is well known that these methods consume high computational power and the solution is not portable from one scenario to another. At the same time, the prediction accuracy is still far from being desirable. On the other hand, regression-based methods (regression trees, neural network and parametric regression) have shown limited success to forecast ozone level. To the best of our knowledge, this paper is the first attempt that use inductive learning technique to issue ozone level alert. Our choice learners are probabilistic decision trees, a number of more sophisticated and well-known algorithms, as well as a base-line parametric model developed by experts in ozone level prediction. Using seven recent years of data, rather exhaustive cross-validation experiments as well as incremental experiments, we have demonstrated that, (1) inductive learning, particularly probabilistic tree ensembles, can significantly improve an expert-based parametric model. To be specific, in the annual incremental test, baggingc4.4 and random decision tree can achieve 10–20% higher recall and up to 10% higher precision than the parametric model. (2) for this problem, and possibly other problems with similar characteristics, these two straight-forward non-parametric methods can provide more accurate and reliable solutions than a number of sophisticated and well-known algorithms, including SVMRBF, SVMLinear, AdaBoost C4.5, AdaBoost NB, Logistic Regression, NB, and a number of decision tree variations. Though our choice of inductive learners are non-exhaustive, this paper has shown that inductive learning can be a method of choice for ozone level forecast, and ensemble-based probability trees provide better forecasts (higher recall and precision) than existing approaches.

For data mining research, besides the procedure to analyze and formulate the problem and look for the most appropriate modeling technique, we have shown that in general, model averaging of posterior probability estimators trained from random subset of feature vectors can effectively approximate the true probability when there are (1) a lot of irrelevant features and (2) feature sample selection bias. For stochastic problems under sample selection bias, we have provided a cross-validation based procedure and guide on how to choose the most appropriate decision threshold as a compromise between precision and recall, and in the same time, avoid "surprises" when applied on biased testing data where neither the feature vector nor the prior class distribution is known. We have also empirically shown that, ensemble learners, including probabilistic tree ensembles and AdaBoost variations, are expected to receive "fewer surprises" and more similar results than single models when the cross-validated decision thresholds are used to "spell" the future.

References

1. Chang CC, Lin CJ (2001) LIBSVM: a library for support vector machines. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm
2. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297
3. Davidson I, Fan W (2003) When efficient model averaging out-performs boosting and bagging. In: Proceedings of the 10th European conference on principles and practice of knowledge discovery in databases. Springer, Berlin, pp. 478–486
4. EPA (1999) Guideline for developing an ozone forecasting program. EPA-454/R-99-009
5. Fan W, Davidson I (2006) ReverseTesting: an efficient framework to select amongst classifiers under sample selection bias. In: Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining. Philadelphia
6. Fan W, Wang H, Yu P, Ma S (2003) Is random model better? On its accuracy and efficiency. In: Proceedings of the 3rd IEEE international conference on data mining
7. Ferri C, Flach P, Hernndez J (2003) Decision trees for ranking: effect of new smoothing methods, new splitting criteria and simple pruning methods. Technical report, UPV(DSIC 2003)
8. Forswall CD, Higgins KE (2006) Clean air act implementation in Houston: an historical perspective, 1970–2005 Technical report, Rice University, Environmental and Energy Systems Institute, Shell Center for Sustainability
9. Ghiaus C (2005) Linear fuzzy-discriminant analysis applied to forecast ozone concentration classes in sea-breeze regime. Atmos Environ 39(26):4691–4702
10. Janssen N, Sanderson E (2004) Air-pollution exposure assessment. http://airnet.iras.uu.nl
11. Kim Y, Kim J (2006) Convex Hull ensemble machine for regression and classification. Knowledge Info Sys 6(6):645–663
12. Lambeth B (2006) Ozone maximum model forecast version. In: Proceedings of the national air quality conference, San Antonio
13. Ling CX, Yan J (2003) Decision tree with better ranking. In: The Proceedings of the 20th international conference on machine learning

14. Mamitsuka H (2006) Query-learning-based iterative feature-subset selection for learning from high-dimensional data sets. Knowl Info Syst 9(1):91–108
15. McMillan N, Bortnicka S, Irwinb M, Berlinerc LM (2005) A hierarchical Bayesian model to estimate and forecast ozone through space and time. Atmos Environ 39(8):1373–1382
16. Mintz R, Young B, Svrcek W (2005) Fuzzy logic modeling of surface ozone concentrations. Comput Chem Eng 29(10):2049–2059
17. Mitchell T (1997) Machine learning. McGraw Hill
18. NCDC (2000) http://www.ncdc.noaa.gov/oa/ncdc.html
19. Ortega S, Soler MR, Beneito J, Pino D (2004) Evaluating of two ozone air quality modeling systems. Atmos Chem Phys Discussi 4:1855–1885, European Geosciences Union
20. Provost F, Domingos P (2003) Tree induction for probability-based rankings. Mach Learn 52(3):199–215
21. Schlink U, Dorlingb S, Pelikanc E, Nunnarid G, Cawleye G, Junninenf H, Greigg A, Foxallb R, Ebenc K, Chattertonb T, Vondracekc J, Richtera M, Dostalc M, Bertuccod L, Kolehmainenf M, Doyleb M (2003) A rigorous inter-comparison of ground-level ozone predictions. Atmos Environ 37:3237–3253
22. Wu X, Yu P, Piatetsky-Shapiro G, Cercone N, Lin TY, Kotagiri R, Wah BW (2003) Data mining: how research meets practical development?. Knowl Info Syst 5(2):248–261
23. Zadrozny B (2004) Learning and evaluating classifiers under sample selection bias. In: Proceedings of the 21st international conference on machine learning. Morgan Kaufmann, Sanfransisco
24. Zhang K, Xu Z, Peng J, Buckles B (2005) Learning through changes: an empirical study of dynamic behaviors of probability estimation trees. In: Proceedings of the 5th IEEE international conference on data mining

Authors Biography

Kun Zhang is an assistant professor at the Department of Computer Science, Xavier University of Louisiana. She received her PhD in Computer Science from Tulane University in 2006. Her main research interests include probabilistic tree models, cost-sensitive learning, sample selection bias, imbalanced datasets, and applications of data mining techniques to tackle difficult problems in medical, business and scientific fields. Her co-authored paper won the best application paper award at ICDM'06.

Wei Fan received his PhD in Computer Science from Columbia University in 2000. He published more than 50 papers in top data mining, machine learning and database conferences, such as KDD, SDM, ICDM, SIGMOD, VLDB, ICDE, AAAI, etc. Dr. Fan has served as Senior PC and PC of several prestigious conferences in the area including KDD'07/06/05, ICDM'06/05/04/03, SDM'07/06/05/04, CIKM'07/06, ECML/PKDD'07'06, ICDE'04, AAAI'07, PAKDD'07, DASFAA'05, EDBT'04, etc. His main research interests and experiences are in risk analysis, high performance computing, extremely skewed distribution, cost-sensitive learning, data streams, ensemble methods, and commercial data mining systems. He is particularly interested in simple, unconventional, but effective methods to solve difficult problems. His thesis work on intrusion detection has been licensed by a start-up company since 2001. His co-authored paper in ICDM'06 that uses "Randomized Decision Tree" to predict skewed ozone days won the best application paper award. His co-authored paper in KDD'97 on distributed learning system "JAM" won the runner-up best research paper award. He won IBM invention achievement awards in 2002, 2003, 2004, 2005 and 2006.