

# Distributional Smoothing in Bayesian Fault Diagnosis

Stephyn G. W. Butcher, *Member, IEEE* and John W. Sheppard, *Fellow, IEEE*

**Abstract**—Previously, we demonstrated the potential value of constructing asset-specific models for fault diagnosis. We also examined the effects of using split probabilities where prior probabilities come from asset-specific statistics and likelihoods from fleet-wide statistics. In this paper, we build upon that work to examine the efficacy of smoothing probability distributions between asset-specific and fleet-wide distributions to improve diagnostic accuracy further. In the current experiments, we also add environmental differentiation to asset differentiation under an assumption that data is acquired in the context of online health monitoring. We hypothesize that overall diagnostic accuracy will be increased with the smoothing approach relative to a fleet-wide model or a set of asset-specific models. The hypothesis is largely supported by the results. Future work will concentrate on improving the smoothing mechanism and in the context of small data sets.

**Index Terms**—Diagnosis (fault), machine learning, Bayesian classifier, smoothing.

## I. INTRODUCTION

RECENT results exploring the merits of fleet-wide vs. asset-specific Bayesian diagnostic models suggest that circumstances can exist where using fleet-wide data in asset-specific models can yield significant improvements in overall diagnostic accuracy. These circumstances hinge largely on data heterogeneity, quantity, and noisiness and their effects on the estimates of the models' probability distributions [1], [2].

This paper reports on our most recent results applying distributional smoothing to probability estimates in Bayesian diagnostic models seeking to combine fleet-wide and asset-specific coverage. Other fields facing similar circumstances when using Bayesian approaches, e.g., natural language processing, apply smoothing to estimates of probability distributions. While most of these smoothing methods are not directly applicable to Bayesian diagnostic models, we present an alternative approach to distributional smoothing that is. We

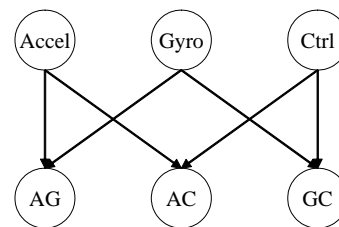


Fig. 1. Simple stability augmentation system BBN.

have found that our models using smoothed probability estimates can be more accurate over a wider variety of data quality and quantity than any of our other models.

The outline of the paper is as follows. The next two sections will describe the research problem and motivation as well as an approach to address the problem. The fourth discusses related work. In the fifth section, we explore the experimental design. The sixth and seventh sections will present and discuss the experimental results (including future work) respectively. We conclude in the final section.

## II. RESEARCH PROBLEM

Our research centers on learning Bayesian diagnostic models from test and maintenance data for an entire fleet. As an example of a diagnostic problem that uses a Bayesian network, suppose that we are considering the built-in test (BIT) from the stability augmentation system (SAS) of a helicopter. Stability augmentation systems provide stability control for the three axes of the aircraft, namely roll, pitch, and yaw. Without loss of generality, we will consider just the roll axis. In evaluating the performance of the roll stability control in the SAS, we might consider the health of at least three components: the roll control unit, the roll gyro, and an accelerometer. For our example, we assume that if the expected output of the control unit agrees with the actual, derived roll outputs from the accelerometer and roll gyro, then the system is functioning properly. On the other hand, if any two of these three elements disagree, a fault exists in one of the two units involved in the disagreement. This scenario can be represented with the Bayesian network shown in Fig. 1.

To interpret the elements of this network, Accel, Gyro, and Ctrl correspond to the diagnoses of whether the accelerometer, gyro, or control unit are faulty respectively. AG represents the observation associated with comparing the accelerometer output with the gyro output. AC compares the accelerometer output with the control output, and GC compares the gyro

Manuscript received October 31, 2007, revised March 25, 2008.

Based on "Asset-Specific Bayesian Diagnostics in Mixed Contexts," by S. G. W. Butcher and J. W. Sheppard, which appeared in the *Proceedings of IEEE AUTOTESTCON 2007*, © 2007, IEEE.

S. G. W. Butcher is with The Johns Hopkins University, Baltimore, MD 21218 USA (sbutche2@jhu.edu).

J. W. Sheppard is with The Johns Hopkins University, Baltimore, MD 21218 USA (phone: 410-516-4957; fax: 410-516-6134; e-mail: jsheppa2@jhu.edu).

output with the control output.

Using this network, suppose, we indicate that AC and AG both fail but CG passes. Logically, we would expect Accel to be faulty. Given a set of probability distributions for each of the nodes in the network, we might find posterior probabilities of {Accel: 0.539; Ctrl: 0.069, Gyro: 0.230; NF: 0.000}. Thus we would conclude from the tests that Accel is the most likely to have failed.

Bayesian networks such as the one described above are either constructed from domain knowledge or learned from actual test and maintenance data. In the latter case, one would accumulate data recorded over the current life of the system and derive the probability distributions from that data. Typically, such a data driven approach uses all of the data available to construct a model for a particular system. That is, they use data about the entire fleet of helicopters and construct fleet-wide models. As a practical matter, however, such data may be sufficiently heterogeneous that relevant diagnostic information is lost through aggregation. For example, specific helicopters may be from different production runs, they may have been exposed to different usage profiles, or they may have used different replacement parts. Similarly, the test equipment may have been operated in different environments or the test data may have been obtained from online health monitors which themselves may be affected by different usage profiles.

All of these circumstances may conspire to produce training data that contains a degree of inconsistency when aggregated. To the degree these heterogeneities exist, a model learned from aggregated data will be less accurate, and such a decrease in accuracy is predicted by machine learning theory [3]. Because a Bayesian diagnostic model is a type of classifier, the more closely the distribution of the training data matches the distribution of the target population, the more accurate the classifier will be [4]. In diagnostic terms, the more the maintenance and test data used to build the model reflect failure rates and test/diagnosis relationships the model will actually encounter when used in the field, the more accurate the diagnoses will be.

Thus the alternative—at the other extreme—is to build a model for each individual asset (a car, plane, or GPS unit with a specific serial number) under the assumption that each asset is *sui generis*, rather than one for the entire fleet of such assets. We realize that this assumption is equally unrealistic, and that is largely what motivates the research reported here.

We used the naïve Bayesian classifier for our learned diagnostic model in the reported experiments [5]. We decided on the naïve Bayes classifier because of its robustness, low computation complexity, and ease in learning. In addition, using such a simple model permits us to focus our attention on the affects of the smoothing approach without commingling with learning an appropriate model structure. The naïve Bayesian classifier is represented by the following equation:

$$D = \arg \max_{D_i \in \mathbf{D}} P(D_i) \prod_{j=1}^n P(o(T_j) | D_i) \quad (1)$$

where  $D_i$  is some diagnosis,  $P(D_i)$  is the prior probability estimate of a particular fault in the data set and  $P(o(T_j) | D_i)$  is the frequency of some discrete test outcome  $o(T_j)$ , e.g., PASS or FAIL, for some test  $T_i$ , considering only the particular diagnosis  $D_i$ , or the likelihood.<sup>1</sup> Thus the possible inconsistencies created by aggregation show up in the estimates of the priors  $P(D_i)$  and likelihoods  $P(o(T_j) | D_i)$ .

Because of the possibility that some likelihood estimates may be zero because of missing data, likelihood estimates are often calculated using the  $m$ -estimate [3]:

$$P(o(T_j) | D_j) = \frac{n_c + mp}{n + m} \quad (2)$$

where  $n_c$  is the number of instances in the data pairing particular values for  $o(T_i)$  and  $D_j$ ,  $n$  is the total number of instances in the data corresponding to diagnosis  $D_j$ ,  $p$  is a prior estimate for the probability, and  $m$  is the number of “virtual” examples in the data. This prevents the equation for the naïve Bayesian classifier from degenerating if any likelihood estimate is zero.

Note that the  $m$ -estimate modifies the likelihood estimate by adding in some fraction of a probability mass,  $p$ . That fraction is determined by some number of virtual examples,  $m$ . In many cases,  $p$  is simply chosen to be a small innocuous value sufficient to prevent the formula from zeroing out and  $m$  is often set to one. However, this need not be the case. If there is knowledge of  $p$ , or if  $p$  can be learned, then using that  $p$  should yield a more accurate estimate.

Based on the above observation on heterogeneous data, our original experiments investigated if a set of diagnostic models, each built with asset-specific data, would have a higher overall accuracy than a single diagnostic model built with aggregate data for the entire fleet. Our experiments used different quantities of synthesized data reflecting assets with different failure rates and varying levels of measurement noise<sup>2</sup> so that we could control the presence of heterogeneity. Our results showed that a set of asset-specific models could be more accurate than a single fleet-wide model but not always.

In our original experiments, we modeled asset differences by using various failure rates for their respective components. Usually failure rates are used to estimate  $P(D_i)$  in Equation 1. The current research adds the ability to capture usage patterns or environmental effects. By including usage patterns or environmental effects, we are able to capture different relationships between test outcomes and diagnoses within the data. Even with this change, the initial results were the same; a set of asset-specific models could be more accurate than a single fleet-wide model.

Although the results were consistent, the fact that some of the asset-specific models were less accurate than the fleet-wide model was problematic. Specifically, for a given quantity of data,  $N$ , as noise increased, the accuracy of the asset-specific models increased relative to the fleet-wide

<sup>1</sup> For a more in-depth discussion of Bayesian approaches to diagnostics, see some of our previous papers [1],[2],[8].

<sup>2</sup> Details of the type of noise used is discussed in Section V.

model. We attribute this to better estimates of the prior probabilities in the asset-specific models. Additionally, for a given level of noise in the data, the larger the data set size, the more accurate the asset-specific models were. We attribute this to better estimates of different likelihoods for the asset-specific models.

While the patterns in these results are interesting, the mixed results leave open the question of knowing when to use the fleet-wide model and when to use the asset-specific model. However, the results did point to an approach to achieve our goal. In the presence of noise, other things being equal, we need to use as much data as possible to average out that noise. This supports aggregating the data. However, in the presence of heterogeneity, other things being equal, we need to use as specific data as possible.

### III. DISTRIBUTIONAL SMOOTHING

Based on these observations, we hypothesize that a new model that uses smoothed distributional estimates for the likelihoods in the naïve Bayesian classifier will be more accurate than either model alone. The rationale for our smoothing approach is based on analysis of our prior experiments [1], [2]. In those experiments, we observed that the accuracy of the asset-specific vs. fleet-wide classifier depended heavily on the amount of noise in the data and the sample size, and this dependence was nonlinear. We also observed a strong dependence on the distribution of the priors from the asset-specific data regardless of noise and sample size. Therefore, we decided to focus on smoothing the likelihoods alone.

The starting point for distributional smoothing is Equation 2 above for the  $m$ -estimate. When estimating the asset-specific likelihood, we use asset-specific data to calculate  $n_c$  and  $n$  and the *fleet-wide likelihood* as the value for  $p$ . When we estimate the fleet-wide likelihood, we use fleet-wide data to calculate  $n_c$  and  $n$  and a small value for  $p$ . We use a formula for  $m$  consistent with the following form when determining the asset-specific likelihood:

$$m = \frac{k}{f(o(T_i), D_j)g(n)} + 1 \quad (3)$$

where  $k$  is a user defined parameter to control how much weight goes towards the asset-specific estimate versus the fleet-wide estimate,  $f(o(T_i), D_j)$  is a function that relates the asset-specific and aggregate distributions for the  $(T_i, D_j)$  pairs, and  $g(n)$  is a function of the data set size for a particular diagnosis  $D_j$ . The resulting function will cause  $m$  to decrease as noise and amount of data increases.

This formula represents a generalization of the result provided in [8]. Consequently, several options now exist for  $f(o(T_i), D_j)$ . If we observe that we are dealing with two different probability distributions where the asset-specific distribution tends to be the “preferred,” we can apply a function of the divergence between the distributions for  $P(o(T_i) | D_j)$ , such as Kullback-Leibler divergence:

$$D_{KL}(P_{asset} \parallel P_{agg}) = \sum_i P_{asset}(o(T_i) | D_j) \lg \frac{P_{asset}(o(T_i) | D_j)}{P_{agg}(o(T_i) | D_j)} \quad (4)$$

or the chi-squared statistic:

$$\chi^2 = \sum_i \frac{(P_{asset}(o(T_i) | D_j) - P_{agg}(o(T_i) | D_j))^2}{P_{agg}(o(T_i) | D_j)} \quad (5)$$

where  $P_{agg}$  is the aggregate probability distribution and  $P_{asset}$  is the asset-specific probability distribution. A simple measure inspired by prior experiments was the conditional variance of the aggregate distribution:

$$Var(o(T_i) | D_j) = E[(o(T_i) - E[o(T_i) | D_j])^2 | D_j] \quad (6)$$

Similarly, several options exist for  $g(n)$ , ranging from  $g(n) = n$  to some polynomial or exponential function of  $n$ . In our experiments, we found that  $f(o(T_i), D_j) = Var(o(T_i) | D_j)$  and  $g(n) = n^{1/q}$  worked well.

Before we describe the experimental design used to test our hypothesis, we’ll look at some of the related work in Bayesian diagnosis and distributional smoothing.

### IV. RELATED WORK

The idea of applying Bayesian methods in general and Bayesian networks in particular to diagnosis is not new. Early Bayesian methods involved manually constructing models as an alternative to rule-based expert systems [9]. Perhaps the best known Bayesian network method is the “Quick Medical Reference-Decision Theoretic” model [10]. The QMR-DT model was a “bi-partite” network where diagnoses were root nodes and tests/observations were leaf nodes. Diagnoses were connected directly to tests. This model is similar to the naïve Bayes model (Equation 1) in that a naïve Bayes network is also bipartite. QMR-DT, however, does not employ the conditional independence assumption.

More recently, Lerner, *et al.*, applied Bayesian networks to perform fault diagnosis in dynamical systems [11]. Their approach made use of a hybrid dynamic Bayesian network (DBN) [12] to represent the dynamics of the system. This approach is similar to the factorial hidden Markov model approach used by Singh *et al* [13]. In their approach, factorial HMMs were used to incorporate historical information for purposes of multiple fault diagnosis.

Although Bayesian techniques are used in many fields of computer science, the  $N$ -gram models used in natural language processing (NLP) use smoothing techniques that bear some resemblance to the distributional smoothing technique for Bayesian diagnostics described in this paper [14]. One use of the  $N$ -gram technique is to classify text. For each type of text, a separate  $N$ -gram model is trained on texts of that type, for example, astronomy articles for one model and astrology articles for another. The typical  $N$ -gram sets  $N=3$  and is called a “trigram.” A trigram is the condition probability of a third word given the first and second words. The trigram model is built by calculating all of the trigrams from the training texts,

$P(w_3 | w_1, w_2)$ ,  $P(w_4 | w_2, w_3)$ , etc. It is, essentially, a second-order Markov chain. Using these models, we can then find the product of those probabilities for a text we want to classify and determine which one has the higher probability. In this case, whether the new text is astronomy or astrology. This simple approach uses unsmoothed maximum likelihood estimates for the trigrams.

The problem arises in NLP that no matter how many texts are used in training, it is always possible that a new text will have something slightly different than anything seen before. This is the same problem that the  $m$ -estimate is designed to handle in general Bayesian classification. The NLP response is to use smoothing or a related technique called backoff (and sometimes both). A variety of smoothing methods have been introduced for  $N$ -gram models over the years including Add- $\lambda$  (which is similar to the  $m$ -estimate) [14], Witten-Bell discounting [15], and Good-Turing discounting [16].

Essentially, backoff is a technique employing the strategy that says if a trigram equals zero, use the bigram. If the bigram equals zero, use the unigram. Finally if the unigram is zero, back off to a uniform probability [17]. While our approach has much more in common with smoothing, to a certain degree our proposed technique does “backoff” from the asset-specific estimate to the fleet-wide estimate during the smoothing process. The similarity ends there, because it can also move in the other direction as well (i.e., from fleet-wide to asset-specific).

As an alternative to creating a single, smoothed model, many have suggested using ensembles of models or combining models through averaging to improve prediction accuracy. Ensemble methods seek to improve accuracy by combining recommendations from multiple classifiers [18]. Ensemble methods vary widely and include, for example, examples bagging, boosting, and mixtures of experts.

Bagging normally involves the creation a set of classifiers by using bootstrapping to resample the available data. Boosting involves creating successive classifiers trained on the mistakes of the previous classifier. Both approaches have been used in classifiers used for diagnostics [19], [20]. Mixtures of experts create a meta-classifier that combines the results of simpler classifiers and have been successfully used with Bayesian approaches to classification [21], [22].

An alternative ensemble-based approach involves generating several models and combining their predictions through model averaging. Madigan and York describe an approach to Bayesian model averaging where they generate a baseline model and then generate alternative models using a Markov chain Monte Carlo approach [23]. Meila and Jaakola discuss approaches to performing exact model averaging over tree-based Bayesian networks [24], and Dash and Cooper showed how to perform exact model averaging over naïve Bayesian classifiers [25]. All of the ensemble methods differ from ours in that they construct multiple models and combine their predictions. Model averaging, specifically, differs from our approach, not only in the averaging over multiple models but by training over the same data set. We derive and smooth estimates of probabilities within a single model using data

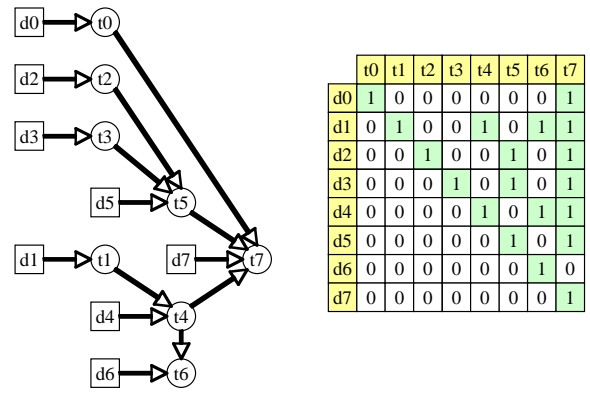


Fig. 2. Sample logic model and  $D$ -matrix for experimental evaluation.

drawn from different populations.

## V. EXPERIMENTAL DESIGN

To test our hypothesis, we generated synthetic data that was known to reflect asset-specific heterogeneity both in terms of failure rates, test outcomes, and diagnoses. We started with known and consistent diagnostic relationships modeled using a  $D$ -Matrix [26]. The particular  $D$ -Matrix represented eight tests over eight possible component failures. Because each row in the  $D$ -Matrix represents a test signature or pattern of passing and failing tests and a corresponding diagnosis, rows were repeatedly extracted from the matrix in proportion to different failure distributions to create the data. These different failure distributions are one kind of asset-specific variation that may be present in fleet-wide data and are eventually used in the Bayesian diagnostic model to estimate the prior probability distributions. The logic model and  $D$ -Matrix for this idealized system used for data generation can be seen in Fig. 2.

On the other hand, the test signatures themselves end up represented in the Bayesian diagnostic model as estimates of the likelihoods. These likelihoods have two important characteristics. First, they represent a transformation of the test outcome and diagnosis relationships in the  $D$ -Matrix rows to probabilistic representations. This reduction is advantageous when the relationships need to be learned from (possibly noisy) test data. Second, the likelihoods in the naïve Bayesian classifier represent an assumption about the conditional independence between tests given the diagnoses. Despite the assumption not generally holding in practice, the naïve Bayesian classifier consistently performs well. In fact, a naïve Bayesian classifier can learn the diagnostic model represented by a  $D$ -Matrix with 100% accuracy as long as no diagnosis is present more than once in the  $D$ -Matrix [5], [27].

In order to introduce a degree of asset-specific variability in the fleet-wide data with respect to the likelihoods, we must sample from slightly different  $D$ -Matrices. We start with the baseline  $D$ -Matrix and create small changes. For example, perhaps a certain test always passes or always fails, or a certain test always fails for a particular diagnosis. The rationale for these changes is that usage profiles, extreme environmental effects, or variations in online testing

conditions may have caused the real world diagnostic relationships to diverge from the baseline.

To simulate the different conditions affecting diagnostic performance, we used five different failure distributions and five different  $D$ -Matrices creating a total of 25 different combinations of specific assets. Examples of the failure distributions include uniform probability and “one bad actor” distributions where one component was significantly more likely to fail than the others. Examples of the different  $D$ -Matrices—reflecting different usage profiles or environmental conditions—included the baseline  $D$ -Matrix and  $D$ -Matrices where one test always failed and another where one test always passed. All variations are described fully in [8].

As noted previously, a naïve Bayesian classifier can learn training data directly derived from a  $D$ -Matrix with 100% accuracy. Therefore, to introduce a degree of realism into the data, various levels of noise were added. This was accomplished by converting an expected pass or fail result of each test from the test signature into two real values, one for pass and one for fail. Each real value was then perturbed by independently and identically distributed Gaussian white noise with zero mean of different variances. If a passing value fell below a certain threshold it was converted into a false positive or failure; otherwise, it was kept as a pass. Similarly, if a failing value fell above that threshold it was converted into a false negative or pass; otherwise, it was kept as a failure.

When generating the data, noise was introduced by varying the standard deviation of the real values from 0.0 to 0.1 in 0.01 increments. Different data set sizes for each asset were also generated ranging from 25 to 5,000 observations. For each noise level and data set size, three naïve Bayesian diagnostic models were constructed: a fleet-wide model, a set of asset-specific models and a set of combined models using distributional smoothing. Because the fleet-wide model aggregates all of the available data, each fleet-wide model is trained using datasets with size  $MN$ , where  $M = 25$  is the number of assets.

We ran 30 trials for each experiment ( $N$  and noise level combination). Each trial used 66% of the data to train and 34% of the data to test the model. New data was generated for each trial. Results were averaged over the trials, and a two-tailed difference of means test ( $t$ -test) was used for all comparisons with a significance level of 0.05. All random selection was stratified first by asset (if necessary) and then by diagnosis. The  $m$ -estimate was set with  $p = 0.001\%$  and  $m = 1$  in all cases except in the combined model where distributional smoothing was used to estimate the likelihoods. In that case, the special  $m$  and  $p$  were used from Equation 3. For the combined model’s estimate of the smoothed likelihoods, the user defined parameters  $k$  and  $q$  were set to 100 and 1.2 respectively. Choosing a diagnosis at random broke all classification ties. For a more in-depth explanation of the experimental design, see our previous papers [1], [2], [8].

## VI. RESULTS

Our experimental results are presented in TABLE I through TABLE IV. In each we compare the differential performance of asset-specific models or the smoothed models with the fleet-

TABLE I. NUMBER OF ASSET-SPECIFIC MODELS AS GOOD AS THE FLEET-WIDE MODEL (OUT OF 25).

$N$	Noise (Standard Deviation)										
	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
25	25	25	24	25	25	20	15	15	13	11	10
50	25	25	25	25	25	16	11	8	5	8	8
100	25	24	25	25	25	17	7	7	9	12	16
250	25	25	25	25	25	7	2	10	16	25	21
500	25	25	25	25	25	6	17	20	25	25	23
1000	25	25	25	25	8	14	23	23	24	25	23
2500	25	25	25	25	10	24	25	25	25	25	23
5000	25	25	25	25	19	25	25	25	25	25	23

TABLE II. NUMBER OF SMOOTHED MODELS AS GOOD AS THE FLEET-WIDE MODEL (OUT OF 25).

$N$	Noise (Standard Deviation)										
	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
25	25	25	24	25	25	25	25	25	25	25	25
50	22	22	22	25	22	23	25	25	25	25	25
100	25	25	25	25	25	25	25	25	25	25	25
250	25	25	25	25	25	25	25	25	25	25	25
500	25	25	25	25	25	25	25	25	25	25	25
1000	25	25	25	25	25	25	25	25	25	25	25
2500	25	25	25	25	25	25	25	25	25	25	25
5000	25	25	25	25	25	25	25	25	25	25	25

wide model. We look both at the case where the set of models is “just as good as” the fleet-wide model ( $t$ -statistic  $\geq -1.96$ ) and when the set of models is “better than” the fleet-wide model ( $t$ -statistic  $> 1.96$ ). With five failure distributions and five usage profiles, there are 25 different possible models.

TABLE I shows results for the asset-specific model compared to the fleet-wide model. This pattern is what originally inspired this research when we were investigating the accuracy of asset-specific models. When the noise level is zero, all of the asset-specific models are at least as good as the fleet-wide model (with a few random hits here and there). This trend continues until about noise level 0.04 when some of the asset-specific models begin to lose accuracy relative to the fleet-wide model. As noise increases, the drop off in accuracy occurs at smaller and smaller  $N$  but also returns with smaller and smaller  $N$ . For example, at noise level 0.05, accuracy drops off steadily from an initial value of 20 but begins to rebound at  $N = 1000$ . On the other hand, with noise level 0.08, the drop off starts with 13 but begins to rebound with  $N = 250$ .

The results in TABLE I are the frame of reference for the remainder of the experiments. Ideally, we want all of the cells with values less than 25 to be 25 when using the fleet-wide

TABLE III. NUMBER OF ASSET-SPECIFIC MODELS BETTER THAN THE FLEET-WIDE MODEL (OUT OF 25).

N	Noise (Standard Deviation)										
	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
25	1	1	2	2	1	2	1	1	2	2	3
50	1	1	1	1	1	1	2	1	0	0	0
100	1	1	1	1	2	1	1	1	1	1	2
250	1	3	1	1	2	1	1	4	5	10	12
500	1	1	1	1	1	1	3	8	11	17	19
1000	1	1	1	5	1	1	13	15	15	21	24
2500	1	1	2	2	2	15	18	16	17	23	25
5000	1	1	1	2	4	17	21	19	19	24	25

TABLE V. NUMBER OF SMOOTHED MODELS BETTER THAN THE FLEET-WIDE MODEL (OUT OF 25).

N	Noise (Standard Deviation)										
	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
25	1	1	1	1	1	1	1	3	5	6	8
50	1	1	1	1	1	1	2	4	3	8	11
100	1	1	1	1	1	2	2	4	10	10	11
250	1	1	1	1	1	4	9	11	10	13	16
500	1	1	1	1	3	10	15	10	12	18	20
1000	1	1	1	1	5	15	16	16	15	21	23
2500	1	1	1	1	13	18	18	18	17	23	24
5000	1	1	1	1	16	17	20	19	19	24	25

model would be an improvement. However, rather than actually using the fleet-wide model, we performed distributional smoothing by including the fleet-wide data in the estimate of the likelihoods for the asset-specific models, resulting in a set of smoothed models.

As TABLE II shows, the set of smoothed models was able to achieve the desired result. Except for a run at  $N = 50$ , all 25 of the smoothed models are at least as good as the fleet-wide model. While these results are encouraging, they must be tempered with the realization that ultimately we want the emphasis to be on the “*or better*” part of the “as good or better.” After all, the fleet-wide model is always as good as itself. TABLE III shows the results of comparing the accuracy of asset-specific models against the fleet-wide model.

These results are typical of those we have found in previous research. Just as TABLE I shows that there are some asset-specific models that are less accurate than the fleet-wide model, there are some asset-specific models that are better. For example, at  $N = 250$  and noise level 0.07, TABLE III shows that four of the asset-specific models are better than the fleet-wide model. Referring back to the same cell in TABLE I, we can see that 10 were “as good or better.” Thus overall, for

TABLE IV. AVERAGE PERCENT INCREASE IN ACCURACY BETWEEN THE SET OF SMOOTHED MODELS AND THE FLEET-WIDE MODEL.

N	Noise (Standard Deviation)										
	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
25	1.8	1.8	1.8	1.8	1.9	2.3	3.2	4.1	5.8	6.7	9.2
50	0.5	0.5	0.5	0.6	0.6	1.0	2.0	2.5	3.6	6.5	8.1
100	1.1	1.1	1.1	1.1	1.2	1.7	2.0	3.1	4.6	5.5	6.7
250	1.3	1.3	1.3	1.2	1.4	2.1	2.0	3.5	4.8	6.1	8.1
500	1.3	1.3	1.3	1.3	1.4	2.0	3.1	3.7	4.8	6.5	8.9
1000	1.3	1.3	1.3	1.3	1.5	2.1	3.1	4.1	4.8	6.8	9.3
2500	1.3	1.3	1.3	1.3	1.5	2.2	3.2	4.3	5.0	6.8	9.7
5000	1.3	1.3	1.3	1.3	1.5	2.2	3.3	4.3	4.9	7.1	10.0

that cell, four asset-specific models were better, six were the same and a full 15 were worse than the fleet-wide model in terms of accuracy. Note, however, that when  $N$  is large and the data is noisy, the asset-specific models are not just as good as the fleet-wide model, they are generally all better.

There is also another surprising and subtle result shown in TABLE III. Even when there is little or no noise, there are asset-specific models that do better than the fleet-wide model. This is not usually the case if the asset data is homogeneous. We therefore believe that the heterogeneities we introduced into the data and that may exist in actual test data can lead to the learning of inconsistent models.

The results for the smoothed models versus the fleet-wide model are shown in TABLE V. Compared to the asset-specific models, there are some substantial gains over the fleet-wide model in terms of accuracy. This is especially true once the noise level reaches 0.04 and beyond. It should be noted that where the gains are similar, such as the case when with large  $N$  in the high noise area of the table, the asset-specific models also have some individuals that are worse than the fleet-wide model whereas this is not the case for the smoothed models.

TABLE IV shows our measure of the gains to be had from creating a set of smoothed probability models. Specifically, TABLE IV shows the percent increase in accuracy, on average, over the fleet-wide classifier for the set of smoothed probability models. The gains are modest at low noise levels, which is to be expected. However, they are nearly 10% at the highest noise levels.

One of the areas we left for future research in our previous paper [8] was a sensitivity analysis of our results for different values of the user defined parameters  $k$  and  $q$  in our implementation of Equation 3. In the results just presented, we used of  $k = 100$  and  $q = 1.2$ . An initial sensitivity analysis examined values that were 50% smaller and 50% larger for each while keeping the other parameter constant at the value used above. In all cases, we looked to see how varying  $k$  and  $q$  affected the number of smoothed models better than the fleet-wide model.

The results showed that when we lowered  $k$  to 50, the total number of smoothed models better than the fleet-wide model

dropped by four out of 660. Increasing  $k$  to 150 led to a net gain of only 18 out of 660. On the other hand, when  $q$  was decreased to 0.6, the net loss was 104 out of 660, but when  $q$  was increased to 1.8, the net gain was only 18 out of 660. This suggests that the parameter influencing the contribution of  $N$  to  $m$  (i.e.,  $q$ ) is very important to overall accuracy. Even so, our settings still demonstrated strong performance, even without the more exhaustive evaluation of parameter values.

## VII. DISCUSSION

In prior research, we examined the role of asset-specific models in improving diagnostic accuracy for a fleet of assets. While we found that asset-specific models could improve diagnostic accuracy, this was not always the case. Subsequently we started to investigate ways that we might achieve the accuracy of both approaches in a single model. We realized that this would involve boosting the accuracy of the asset-specific model when the fleet-wide model was more accurate and boosting the accuracy of the fleet-wide model when the asset-specific model was more accurate. This investigation led to the idea of smoothing the distribution estimates using both fleet-wide data and asset-specific data. Furthermore, we sought to make the smoothing endogenous by making the weighting factor a function of the characteristics, data quantity and noise, that we observed to affect model accuracy.

The results in TABLE II, TABLE V, and TABLE IV support our hypothesis. We were able to get the desired effect by using distributional smoothing. With the results of TABLE II, we showed that the smoothed models were not worse than the fleet-wide model—unlike the unsmoothed asset-specific models. In TABLE V, we showed that there were generally more smoothed models that were better than the fleet-wide model than asset-specific models that were better. Even when this was not the case, the number of smoothed models that were better was supported by the fact that none were worse. Finally, TABLE IV showed that the actual increase in accuracy, although data set and noise level dependence could be substantial.

We plan to concentrate future research on four areas. First, the impact of the user-defined parameters of the formula for  $m$  should be examined further. Not only might this improve the overall accuracy but also accuracy in cases of smaller  $N$ . In addition, examining various values of  $k$  and  $q$  on alternative data sets and models might provide more general insight into the range of their impact on model accuracy.

Second, our version of Equation 3 was derived from empirical observation. In this paper, we generalized the equation to use a generic measure of distributional variation and suggested a couple alternative specific measures. In future research, we would like to examine the effects of those specific measures on accuracy.

Third, smaller values of  $N$  may be a special case warranting additional study. Specifically, we believe we need to investigate methods of leveraging small data sets to extract more information from them since, for many real-world systems, large amounts of training data may not be available.

Finally, since real systems are more complex than the artificial system studied here, we expect the test and maintenance data to contain nonlinearities. As we pointed out previously, the naïve Bayes classifier is a linear discriminant. This would suggest abandoning naïve Bayes in favor of learning more complicated network structures. In previous work [7], we investigated the use of tree-augmented naïve Bayes (TAN) in diagnostics [26]. This would be a good starting point.

## VIII. SUMMARY

Based on our research on asset-specific models for Bayesian diagnostics, we discovered that while heterogeneities in the data might support the use of asset-specific models, they were not always more accurate than a fleet-wide model. As a result, we introduced a technique for distributional smoothing that used both asset-specific and fleet-wide data. We hypothesized that this technique could achieve the best accuracies of both models. In order to test the hypothesis, we constructed a data set that emphasized the types of data heterogeneities, namely different failure distribution rates and test signatures, that could exist in a fleet-wide test and maintenance data set and would warrant asset-specific models. Our results showed that we could indeed improve accuracy by using the smoothing technique in a combined model.

## ACKNOWLEDGEMENTS

The authors would like to thank Mark Kaufman and Craig MacDougall from the US Navy for their input and inspiration in helping us formulate our approach. We also thank the reviewers of IEEE AUTOTESTCON and of this special issue for their helpful comments. This work was supported, in part, by contracts from the US Army and the US Navy.

## REFERENCES

- [1] Butcher S., and Sheppard, J., "Improving Diagnostic Accuracy by Blending Probabilities: Some Initial Experiments," *Proceedings of the 18th International Workshop on Principles of Diagnosis (DX-07)*, Nashville, TN, May 2007.
- [2] Butcher S., Sheppard J., Kaufman M., Ha, H. and MacDougall, C. "Experiments in Bayesian Diagnostics with IUID-Enabled Data" *IEEE AUTOTESTCON 2006 Conference Record*, Anaheim, CA: September 2006., pp 605-614.
- [3] Mitchell, T. *Machine Learning*, New York: The McGraw-Hill Companies, 1997.
- [4] Langley, P., Iba W., and Thompson, K., "An Analysis of Bayesian Classifiers," *Proceedings of the Tenth National Conference on Artificial Intelligence*, San Mateo, CA: AAAI Press, 1992, pp. 223–228.
- [5] Duda, R., Hart, P., and Stork, D., *Pattern Classification*, New York: John Wiley & Sons, 2001.
- [6] Sheppard, J. and Kaufman, M., "A Bayesian Approach to Diagnosis and Prognosis Using Built In Test," *IEEE Transactions on Instrumentation and Measurement*, Special Section on Built-In Test, Vol. 54, No. 3, June 2005, pp. 1003–1018
- [7] Sheppard, J., Butcher, S., Kaufman, M., and MacDougall, C., "Not-So-Naïve Bayesian Networks and Unique Identification in Developing Advanced Diagnostics," *Proceedings of the IEEE Aerospace Conference*, New York: IEEE Press, March 2006
- [8] Butcher, S. and Sheppard, J. "Asset-Specific Bayesian Diagnostics in Mixed Contexts" *IEEE AUTOTESTCON 2007 Conference Record*, Baltimore, MD, 2007.

- [9] Kleiter, G., "Bayesian Diagnosis in Expert Systems," *Artificial Intelligence*, 54:1–32, 1992.
- [10] Shwe, M. and Cooper, G. "An Empirical Analysis of a Likelihood-Weighting Simulation on a Large, Multiply-Connected Medical Belief Network," *Computers and Biomedical Research*, 24, 453–475, 1991.
- [11] Lerner, U., Parr, R., Koller, D., and Biswas, G. "Bayesian Fault Diagnosis in Dynamical Systems," *Proceedings of the 17<sup>th</sup> National Conference on Artificial Intelligence (AAAI)*, 2000, pp. 531–537.
- [12] Murphy, K., *Dynamic Bayesian Networks: Representation, Inference, and Learning*, PhD Dissertation, Department of Computer Science, University of California, Berkeley, 2002.
- [13] Singh, S., Choi, K., Kodali, A., Pattipati, K., Sheppard, J., Namburu, S., Chigusa, S., Prokhorov, D., and Qiao, L. "Dynamic Multiple Fault Diagnosis and Solution Techniques," *Proceedings of the International Workshop on Principles of Fault Diagnosis (DX-07)*, Nashville, TN, 2007, pp. 383–390.
- [14] Jurafsky, D. and Martin, J. H. *Speech and Language Processing*. Upper Saddle River, NJ: Prentice-Hall, 2000.
- [15] Witten, I. H. and Bell, T. C. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory* 37(4), 1991, 1085-1094.
- [16] Good, I. J. The population frequency of species and the estimation of population parameters. *Biometrika*, 40, 1953, 237-264.
- [17] Katz, S. M. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(3), 1987, 400-401.
- [18] Polikar, R. "Ensemble Based Systems in Decision Making," *IEEE Circuits and Systems Magazine*, 3rd Quarter 2006, 21-54.
- [19] Hu, Z.-H., Li, Y.-G., Cai, Y.-Z., and Xu, X.-M. "An Empirical Comparison of Ensemble Classification Algorithms with Support Vector Machines," *Proceedings of the Third International Conference on Machine Learning and Cybernetics*, Shanghai, China, August 2004, 3520-3523.
- [20] Li, Y., Cai, Y.-Z., Yin, R.-P., and Xu, X.-M. "Fault Diagnosis on Support Vector Ensemble," *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics*, Guangzhou, China, August 2005, 3309-3314.
- [21] Bishop, C. M. and Svensen, M. "Bayesian Hierarchical Mixtures of Experts," *Uncertainty in Artificial Intelligence: Proceedings of the Nineteenth Conference*, 2003.
- [22] Titsias, M. K. and Likas, A. "Mixture of Experts Classification Using a Hierarchical Mixture Model," *Neural Computation* 14, 2000, 2221-2244.
- [23] Madigan, D. and York, J., "Bayesian Graphical Models for Discrete Data," *International Statistical Review*, 63, 215–232, 1995.
- [24] Meila, M. and Jaakola, T., "Tractable Bayesian Learning of Tree Belief Networks," *Proceedings of the 16<sup>th</sup> Conference on Uncertainty in Artificial Intelligence*, San Francisco: Morgan Kaufman Publishers, pp. 380–388, 2000.
- [25] Dash, D. and Cooper, G., "Exact Model Averaging with Naïve Bayesian Classifiers," *Proceedings of the International Conference on Machine Learning*, San Francisco: Morgan Kaufman Publishers, pp. 91–98, 2002.
- [26] Simpson, W. and Sheppard, J., *System Test and Diagnosis*, Norwell, MA: Kluwer Academic Publishers, 1994.
- [27] Sheppard, J. and Butcher S., "A Formal Analysis of Fault Diagnosis with D-Matrices," *Journal of Electronic Testing: Theory and Applications*, 23, 309–322, 2007.
- [28] Friedman, N., Geiger, D., and Goldszmidt, M., "Bayesian Network Classifiers," *Machine Learning*, 29, 1997, 131–163.

**Stephyn G. W. Butcher** (M'04) is currently pursuing his PhD in computer science at the Whiting School of Engineering, The Johns Hopkins University.



He has served as a Lecturer in economics and Grader in computer science. He received his BA in economics from the California State University, Sacramento, his MA in economics from The American University, Washington, DC, and his MS in computer science from Johns Hopkins. His research interests are mainly in machine learning and include Bayesian networks and evolutionary computation.

**John W. Sheppard** (M'86–SM'97–F'07) is an Assistant Research Professor in the Department of Computer Science, Johns Hopkins University. Dr. Sheppard started his



career as a research computer scientist for ARINC and attained the rank of Fellow. His research interests include algorithms for diagnostic and prognostic reasoning, machine learning and data mining in temporal systems, and reinforcement learning. Dr. Sheppard holds a BS in computer science from Southern Methodist University and an MS and PhD in computer science from Johns Hopkins University. He has over 100 publications in artificial intelligence and diagnostics, including an authored book and an edited book. He is a senior member of the IEEE and currently serves as Vice Chair of the IEEE Standards Coordinating Committee 20 (SCC20) on Test and Diagnosis for Electronic Systems, Secretary and Past Chair of the Diagnostic and Maintenance Control subcommittee of SCC20, and Official Liaison of the Computer Society Standards Activities Board to SCC20.