SELF-CONFIDENCE MEASURES OF A DECISION SUPPORT SYSTEM BASED ON BAYESIAN NETWORKS

by

Marcin Kozniewski

M.Sc. Bialystok University of Technology,
Faculty of Computer Science, 2013

Submitted to the Graduate Faculty of
the School of Computing and Information
in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2019

UNIVERSITY OF PITTSBURGH

SCHOOL OF COMPUTING AND INFORMATION

This dissertation was presented

by

Marcin Kozniewski

It was defended on

April 11, 2019

and approved by

Marek J. Druzdzel, School of Computing and Information, University of Pittsburgh

Stephen C. Hirtle, School of Computing and Information, University of Pittsburgh

Paul W. Munro, School of Computing and Information, University of Pittsburgh

James F. Antaki, School of Biomedical Engineering, Cornell University

Dissertation Director: Marek J. Druzdzel, School of Computing and Information, University of Pittsburgh

SELF-CONFIDENCE MEASURES OF A DECISION SUPPORT SYSTEM BASED ON BAYESIAN NETWORKS

Marcin Kozniewski, PhD

University of Pittsburgh, 2019

A prominent formalism used in decision support is decision theory, which relies on probability theory to model uncertainty about unknown information. A decision support system relying on decision theory produces conditional probability as a response. The quality of a decision support system's response depends on three key factors: the amount of data available to train the model, the amount of information about the case at hand, and the adequacy of the system's model to the case at hand.

In this dissertation, I investigate different approaches to measuring the confidence of decision support systems based on Bayesian networks, addressing the three key factors mentioned above. Some of such confidence measures of the system response have been already proposed. I propose and discuss other measures based on analysis of joint probability distribution encoded by a Bayesian network.

The main contribution of this dissertation is the analysis of the discussed measures whether they provide useful information about the performance of a Bayesian network model. I start the analysis with an investigation of interactions among these measures. Then, I investigate whether confidence measures help us predict an erroneous response of a classifier based on Bayesian networks when applied to a particular case. Further, I conduct an experiment to check how confidence measures perform in combining the models' output in the ensemble of classifiers by weighting. Based on the findings, I conclude that the confidence measures may enrich the decision support system's output to serve as indicators for applicability of the model and its advice to a given case.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

This dissertation summarizes all the work I have done during my Ph.D. program in the Decision Systems Laboratory (DSL) at the University of Pittsburgh. I would like to use this section to thank several people whose support was influential on my work and development as a researcher.

I cannot express how thankful I am to my advisor, Marek Druzdzel, for guidance and all of the lessons on writing and doing research. He kept me motivated for all these years. I thank my dissertation committee members – Stephen Hirtle, Paul Munro, and James Antaki – for their patience, insightful comments, and feedback on my work. Also, I thank Clark Glymour and Stephen Hirtle for agreeing to participate in my comprehensive examination committee. I am grateful to the late Roger Flynn, who supported me in my early dissertation work.

I want to thank all the former and current DSL members I have met. My discussions with Adam in the early stage of my work helped me to discover new ideas. Tomek helped me to accommodate quickly in Pittsburgh when I arrived for the first time. Dmitriy let me join his research project. With Jidapa I discussed several research problems. She also introduced me to many restaurants. It was a pleasure to meet and talk to Martijn and Mark.

I want to thank the people from the Faculty of Computer Science at the Bialystok University of Technology for their positive attitude while I was working on my dissertation in Bialystok. I am grateful to the whole School of Computing and Information, people that work and study there. They all contribute to a wonderful community of friendly people working very hard and helping each other to push the limits. I would like to thank the people that I have been sharing the office with, especially, Evgeny for shared meals and for organizing the skiing trips, Xidao for shared free time, Saeed for investment suggestions.

I thank my parents, Edwin and Alicja, brother Bartosz and his family for supporting me throughout this time.

1.0 INTRODUCTION

Decision support systems (DSS) are used across many fields, such as, military, medicine, or health care. To enhance and support human judgment and decision making, different techniques and models have been applied. A prominent formalism used in decision support is decision theory, which relies on probability theory to model uncertainty about unknown information and utility to model user preferences. A key element of a probabilistic DSS is a model of the domain. To predict events based on observations and the model, probabilistic DSSs usually report posterior probability distributions conditional on these observations.

1.1 MOTIVATION

It is important to know how confident a system is about its analysis of a single instantiation of a problem (case). The confidence of a system relates to quality of its response advice for a specific case and depends on three key factors: (1) the amount of data available to train the model, (2) the amount of information about the case at hand, and (3) the adequacy of the system's model to the case at hand.

As we analyze the first factor, the amount of data used for training the model is always limited. This limitation has implications on the quality of the model. For example, there may be no or just a few records in the data that represent situations similar to the case at hand. Probability distribution learned from such dataset will be deficient.

The second factor plays a role when there is little information about the case at hand. When more information appears, the prediction can be more precise. For example, a physician may have problems to provide a specific diagnosis for a patient with just an information about patient's complaint and physical examination. After the physician orders medical tests and obtains their results, the prediction becomes more specific. Regardless of the case, providing more information will change and typically improve the prediction.

The third factor in the quality of the system advice is its competence in handling the case at hand. It may happen, that the model has been created for a different purpose or with different specialization. For example, a patient suffering from a heart problem is unlikely to get a good advice and help from a dentist.

To address the problem of uncertainty about the quality of a system's advice, probabilistic DSSs could provide some measure of confidence along with the posterior probability that they normally produce. I have encountered confusion about such measure and how it differs from the conditional probability distribution calculated by the system. The posterior probability of an outcome represents how well the given information lets us assert the truth of the outcome, assuming perfect correctness of the model and proper definition of the case query. However, the model is never perfect, due to, for example, the factors described above. Confidence measure will allow the user to quantify, how much he/she can trust the system's output. For example, when the system asserts that the probability of a cancer for a given patient is 6.83%, we would like to know the system's confidence in this probability. Confidence measures could be used to warn the user to take more action or to use the output of the system with caution. Such measures could be calculated for any case query at hand given the model.

1.2 CONTRIBUTIONS

In this dissertation, I focus on deriving measures of confidence that are calculated using the information already encoded in a Bayesian network model and possibly the data on which the model is based. I review existing measures of confidence about the output of a DSS. I propose an approach for describing the confidence about posterior probability distribution in anticipation of future observations and investigate two methods for obtaining it based on the joint probability distribution.

The research question that I focus on is:

> *Are the confidence measures helpful in predicting the performance of a Bayesian network model?*

By predicting the performance of a Bayesian network model, I mean predicting whether applying the model to a particular case may lead to a false answer. In case of success it will mean that the information carried by the measures is significant from the point of view of classification and we may consider it in applications.

To answer the question posed, I check *how the proposed measures relate to each other*. The values obtained for these measures may be dependent. If they are, it may indicate that some of the measures are redundant with respect to each other.

As the confidence measure is applied in practice, it is interesting *whether the proposed measures carry predictive information about performance of the classifier based on a Bayesian network*. I attempt to predict for which records in the testing dataset the classifier will make wrong predictions.

In the final part, I investigate *whether confidence measures are applicable to ensemble of Bayesian network classifiers*. Since we can capture the competence of the model/system, we may try to use that ability to integrate models that are hard to merge due to different definitions of variables. While building large DSSs, it may be easier to split the domain into pieces – separate models. Integration of multiple models into one reliable model is known to be a hard problem. Instead of merging multiple models, we may try to apply smaller models to the case at hand and use them in ensemble by integrating their outputs. The simplest approach to integrating outputs of a collection of models is to treat them equally. I propose to consider confidence measures over outputs of the individual models to derive the weights of each model.

Marcot (2012) provided quite extensive review of different metrics for evaluating uncertainty of Bayesian network models. He covered measures of models sensitivity, influence, complexity, prediction performance, and fitness to the data as general metrics of the model *a priori* to model application. He also paid some attention to the metrics of uncertainty in posterior probability distributions in model application. Van Allen et al. (2008) and Donald and Mengersen (2014) developed different techniques for obtaining error bars/confidence intervals for posterior probabilities based on data and the model.

Measures of conflict among observations, which also relate to rarity of the observations given the model, have been already introduced by: Habbema (1976), Jensen et al. (1990), and Laskey (1991). Most attention in the literature has been given to those measures that are computationally feasible, although some of intractable measures can be approximated.

1.3 OVERVIEW OF THE DISSERTATION

The remainder of this document is structured as follows. Chapter 2 introduces necessary notation and definitions. As the general performance of a Bayesian network is a subject of study in evaluation process, Chapter 3 overviews techniques for validation of Bayesian networks. Chapter 4 describes existing measures of confidence for particular case given the model. Chapter 5 elaborates on the surprise index (Habbema, 1976) and its approximation. Chapter 6 describes some methods for deriving variation intervals over posterior probabilities. Chapter 7 describes simulation of possible cases based on existing Bayesian network models and studies the relationships among the confidence measures calculated for those cases. Chapter 8 elaborates on predicting the erroneous response of a classifier based on a Bayesian network. Chapter 9 describes deployment of confidence measures in ensemble of classifiers based on Bayesian networks. Chapter 10 summarizes the work presented in this dissertation and outlines possible directions for future work.

2.0 PROBABILITY AND PROBABILISTIC DECISION SUPPORT SYSTEMS

In this Chapter, I present necessary definitions and notation in Section 2.1. In Section 2.2, I present concepts and definitions related to Bayesian networks, which are used to model uncertainty and decisions under uncertainty. In Section 2.3, I introduce useful terms that I will use throughout the document.

2.1 PRELIMINARIES

Throughout this document, I will use capital letters, e.g., X, to denote random variables. I will use bold font-face letters, e.g., $\mathbf{V}$, to denote sets. Let $\mathit{Val}(X)$ be a set of possible assignments of a random variable X. If the variable X is discrete, then $\mathit{Val}(X) = \{x_1, \ldots, x_{n_i}\}$. An observation of a variable X is an assignment out of its possible values $X = x_i$, which we will shorten to x_i.

Let $\mathbf{V} = \{V_1, \ldots, V_N\}$ be a set of variables, then the set of possible joint observations of variables in $\mathbf{V}$ is $\mathit{Val}(\mathbf{V}) = \{\{v_{i_1}, \ldots, v_{i_N}\} : v_{i_k} \in \mathit{Val}(V_{i_k})\}$. I will introduce some more specific sets of observations in Section 2.3.

Let $\mathcal{G}(\mathbf{V}, \mathbf{E})$ be an acyclic directed graph, where $\mathbf{V}$ is a set of vertices (nodes) and $\mathbf{E}$ is a set of pairs (V, W) representing directed edges between nodes $V, W \in \mathbf{V}$. Let $\mathrm{Pa}(V)$ be a set of nodes that are immediate predecessors (parents) of V. Let $\mathrm{Ch}(V)$ be a set of vertices that are immediate successors (children) of V.

2.2 BAYESIAN NETWORK

Even though random variables may be continuous, I will focus on models employing discrete random variables.

Definition 2.1. *A discrete* Bayesian network *(BN) (Pearl, 1988) is a pair* $(\mathcal{G}, \Theta)$*, where* $\mathcal{G}(\mathbf{V}, \mathbf{E})$ *consists of*

- $\mathbf{V} = \{V_1, \ldots, V_n\}$ *representing a set of random variables, each with a finite set of mutually exclusive states* $Val(V_i)$ *and*
- *a set of edges* $\mathbf{E}$ *that jointly model independencies among variables* $\mathbf{V}$*;*

Θ *is a set of parameters* $\{\theta_{v_{i,j}|c_k},\ v_{i,j} \in Val(V_i) \wedge c_k \in Val(Pa(V_i))\}$*, which define conditional probability distributions* $\Pr(V_i|Pa(V_i))$ *for each* V_i*.*

Parameters $\theta_{v_{i,\bullet}|\bullet}$ of the conditional probability distribution of a variable V_i can be organized in a conditional probability table (CPT) that describes conditional probability distributions over V_i for all combinations of assignments to $\mathrm{Pa}(V_i)$. Figure 1 shows the Asia model (Lauritzen and Spiegelhalter, 1988), which models the situation of a patient appearing in a clinic with dyspnea (shortness of breath). It consists of eight discrete random variables representing conditions (*Tuberculosis*, *Lung Cancer*, *Bronchitis*), historical data (*Visit to Asia*, *Smoking*), auxiliary variables (*Tuberculosis or Lung Cancer*), symptoms (*Dyspnea*) and examinations (*X-Ray Result*) that physician can perform. An edge between two variables (e.g., *Smoking* and *Lung Cancer*) denotes a direct influence between the two, usually interpreted as causal influence. Absence of an edge (e.g., between *Smoking* and *Visit to Asia*) means that there in no direct influence between variables, which does not exclude other, indirect, associations.

Compact definition of a BN allows us to retrieve the probability of any combination of assignments to all variables by its factorization

$$\Pr(\mathbf{V}) = \prod_{j=1}^{N} \Pr(V_i|\mathrm{Pa}(V_i)) \ , \tag{2.1}$$

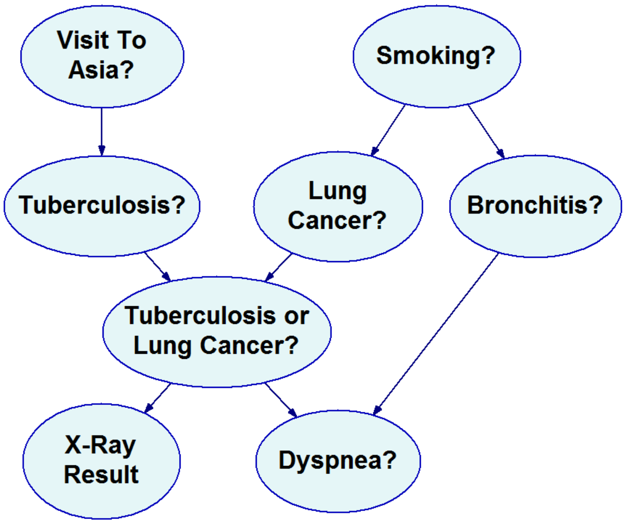


Figure 1: The Asia Bayesian network graph

which for a particular set of assignments turns into

$$\Pr(\{v_{j_1}, \ldots, v_{j_N}\}) = \prod_{i=1}^{n} \theta_{v_{ij}|c_k} \ . \tag{2.2}$$

2.3 TARGET VARIABLES, EVIDENCE, AND MARKOV BLANKETS

Let $\mathbf{T} \subset \mathbf{V}$ be a set of variables of interest (targets). Let the $\mathbf{S} \subset \mathbf{V}$ be all observable phenomena modeled by a BN, e.g., symptoms or patient history data in a medical decision support system. An evidence set $\mathbf{E}$ is a set of observations (assignments) ($\{v_{i_1,j_1}, \ldots, v_{i_k,j_k}\}$, where $\{V_{i_1}, \ldots, V_{i_k}\} = \mathbf{S}_O \subset \mathbf{S}$). A scenario $\mathbf{E}^* \supset \mathbf{E}$ is an evidence set that assigns outcomes to all variables in $\mathbf{S}$. A full scenario $\mathbf{E}_V \supset \mathbf{E}$ is a set of assignments of outcomes to all variables $\mathbf{V}$ modeled in the BN. We will denote by $\mathbf{S}_U$ the set of variables without associated assignment in $\mathbf{E}$ i.e., $\mathbf{S}_U = \mathbf{S} \setminus \mathbf{S}_O$. For example, in the Asia model, variables *Tuberculosis*, *Lung Cancer*, and *Bronchitis* compose the set of target variables $\mathbf{T}$. Variables *Visit to Asia*, *X-Ray Result*, *Dyspnea* and *Smoking* belong to the set $\mathbf{S}$ of observable phenomena. If we consider a patient with dyspnea, we have an evidence set consisting of one assignment $\mathbf{E} = \{dyspnea = present\}$. Based on this evidence set $\mathbf{E}$, we can calculate the posterior probability of the patient having tuberculosis $\Pr(Tuberculosis = present|\mathbf{E})$. Usually term *probabilistic inference* refers to calculations of posterior probabilities (Lauritzen and Spiegelhalter, 1988).

Information about conditional independence of observable variable and target variable helps us to simplify the computation of the posterior probability distribution. To simplify the algorithm presented later in Chapter 6, we will need to specify the necessary set of observable variables required to provide information about posterior marginal probability distribution, which is usually referred to as the *Markov blanket*.

Definition 2.2. *The* Markov blanket *of a variable $V_i \in \mathbf{V}$ is the set $\mathbf{M}(V_i) \subset \mathbf{V}$ consisting of variables that are parents $Pa(V_i)$, children $Ch(V_i)$, and other parents of its children $Pa(Ch(V_i))$, i.e.,*

$$\mathbf{M}(V_i) = (Pa(V_i) \cup Ch(V_i) \cup Pa(Ch(V_i))) \setminus \{V_i\} \; .$$

$\mathbf{M}(V_i)$ represents all variables such that, when observed, make V_i independent of the remainder of the variables in the network. For example, in Figure 1, $\mathbf{M}(Smoking) = \{$*Lung Cancer, Bronchitis*$\}$, as variables *Lung Cancer* and *Bronchitis* make *Smoking* independent of the rest of the network.

We can extend the definition of Markov blanket to sets of variables $\mathbf{A} \subset \mathbf{V}$. $\mathbf{M}(\mathbf{A})$ is a union of Markov blankets $\mathbf{M}(V_i)$ of each variable $V_i \in \mathbf{A}$ excluding V_i, i.e.,

$$\mathbf{M}(\mathbf{A}) = \left(\bigcup_{V_i \in \mathbf{A}} \mathbf{M}(V_i) \right) \setminus \mathbf{A} \; .$$

A Markov blanket $\mathbf{M}(V_i)$ may contain a variable V_j that is not observable (i.e., $V_j \in \mathbf{V} \setminus \mathbf{S}$), in which case V_j cannot be observed and, hence, cannot be used to screen V_i from the rest of the network. We extend the definition of Markov blanket $\mathbf{M}(V_i)$ to an extended Markov blanket.

Definition 2.3. *An* extended Markov blanket *$\mathbf{M}^*(V_i)$ is a set of observable variables that makes V_i independent from all the other observable variables.*

$\mathbf{M}^*(V_i)$ can be calculated recursively in the following way. We start with a set $\mathbf{C} = \{V_i\}$. We add all non-observable variables $V_j \in \mathbf{M}(\mathbf{C}) \cap (\mathbf{V} \setminus \mathbf{S})$ to $\mathbf{C}$. We repeat this procedure as long as $\mathbf{M}(C) \cap (\mathbf{V} \setminus \mathbf{S}) \neq \emptyset$, in which case $\mathbf{M}^*(V_i) = \mathbf{M}(\mathbf{C})$.

3.0 METHODS FOR EVALUATION OF BAYESIAN NETWORK MODELS

In this chapter, I describe some of the methods used to validate probabilistic DSS models. Most of these methods are well known in data mining, machine learning, and artificial intelligence community, as they are widely used for validation of models in supervised learning tasks. Methods presented in the following sections are performed to evaluate a model before its application. Remaining chapters of this dissertation focus on assessing the model's quality of prediction on individual cases.

3.1 DATA-BASED EVALUATION

Bayesian networks can be build from data or can be based on experts' knowledge of a domain. When using data, we infer both the structure and the parameters of the model in the learning process (Cooper and Herskovits, 1992). When building a model employing the expert's knowledge, we construct the graph by modeling known causal independencies and elicit the parameters from experts (Druzdzel and Van Der Gaag, 2000). We can also combine these two approaches by defining the structure of the model with the expert and populating the parameters using data, e.g., by means of the Expectation-Maximization (EM) algorithm (Dempster et al., 1977).

If we create a model just employing experts' knowledge, we want to check how accurate it is. To assess the accuracy of a model with a dataset, we apply the model to all the records in the dataset and compare the output against correct predictions specified along with other observations in the record of the dataset. As a result, we can report the number of times when the model correctly predicted the outcome. I will elaborate more on measures that can be reported within validation process later on in following subsections.

3.1.1 Validation and cross-validation

If we create any part of the model with records from the dataset, we want to check its performance on a different set of records. For that, we separate the dataset into two subsets: training and testing. We use the training part to learn the model (or just its parameters) and testing part for validation. We refer to this method as *holdout validation*.

The main drawback of holdout method is that it does not use all the records in the dataset for training. It is a major concern when we have just a handful of records in the dataset. If we can assume that all the records come from the same probability distribution, i.e., they are independent and identically distributed (i.i.d.), we can repeat the holdout method for different partitions of the dataset. In this situation we may use *k-fold cross-validation* method, which divides randomly the dataset into k mutually exclusive subsets (folds). Now we run holdout validation procedure k times using each fold for testing the model learned using the remainder of the dataset. We accumulate the results from different folds and report the summary. We may use different values of k. In literature, the term *leave-one-out cross-validation* (LOOCV) refers to the situation when $k = N$, where N is the number of records in the dataset.

The main drawback of the k-fold cross-validation is that it requires k runs of learning procedure. Increasing k sacrifices computation time for reliability of the validation result, if the learning procedure is computationally complex. k-fold cross-validation can not be used when the i.i.d. assumption is violated, e.g., when we deal with time-series data.

There are other methodologies of organizing the work-flow of the validation with the data, but they are beyond the scope of this document. The next section will pay some attention to combining expert-based and data-based validation. The remainder of this section describes some of the possible ways of summarizing the results of validation.

3.1.2 Prediction accuracy and other confusion matrix-based measures

Confusion matrix, as a report of performance of classification, represents how cases of each class were classified by the model. Let us consider a classification task with class variable that consists of k labels. A $k \times k$ confusion matrix A consist of elements a_{ij}, which represent how many records of class c_j were classified as the class c_i. Table 1 shows an example of a confusion matrix of a model. The summarized model classified 30 records associated with class c_1 correctly. Meanwhile, the model classified 10 records associated with class c_2 as class c_3.

Table 1: An example of a confusion matrix of a classifier

		records of class		
		c_1	c_2	c_3
classified as	c_1	30	12	0
	c_2	8	44	15
	c_3	5	10	76

From the confusion matrix, we can determine the accuracy of the classifier by calculating a fraction of a sum of elements on the diagonal and number of records in the testing dataset n (which is a sum of all elements in the matrix), i.e.,

$$accuracy = \frac{1}{n} \sum_{i=1}^{k} a_{ii} \ .$$

Error rate is complementary to accuracy, i.e., error = 1 − accuracy.

When we focus on one c_i class label (e.g., c_1), based on a confusion matrix, we can calculate *precision*, *recall* (also referred to as *sensitivity* or *true positive rate*), and *specificity* (*true negative rate*). Precision of classifying c_i is the proportion of records correctly classified as c_i (true positive) to all records classified as c_i, which is

$$precision(c_i) = \frac{a_{ii}}{\sum_{j=1}^{k} a_{ij}} \ .$$

Recall is a fraction of records correctly classified as c_i to all records associated with class c_i, i.e.,

$$recall(c_i) = \frac{a_{ii}}{\sum_{j=1}^{k} a_{ji}} .$$

Specificity is a fraction of records correctly classified as $c_j \neq c_i$ to all records associated with other classes than c_i, i.e.,

$$specificity(c_i) = \frac{\sum_{j,l=1,j,l\neq i}^{k} a_{jl}}{\sum_{j=1,j\neq i}^{k} \sum_{l=1}^{k} a_{lj}} .$$

Using precision and recall values, we can calculate F_1 score (F-score or F-measure) of a classifier, which is

$$F_1(c_i) = 2 \times \frac{precision(c_i) * recall(c_i)}{precision(c_i) + recall(c_i)}.$$

F_1 score is used to asses models for information retrieval task and classification tasks with class imbalance. For such problems, accuracy does not assess the performance of the model well, by advocating for models that favor the majority class.

There are other measures that can be reported for a classifier based on confusion matrix, but they are beyond the scope of this document.

3.1.3 ROC curve, area under the ROC curve

In a classification task, the main objective of a model is to discriminate records associated with each class. Many models produce a criterion value, which is used to determine the class for the case, e.g., in a probabilistic model the marginal probability of the class is used to determine how likely it is that the analyzed record belongs to a particular class. One way to present the ability of the model to discriminate the records of particular a class is the *Receiver Operating Characteristic* (ROC) curve (Bradley, 1997; Egan, 1975; Fawcett, 2006; Spackman, 1989). ROC curve is a plot of points $(x, y) \in [0, 1] \times [0, 1]$ in a Cartesian coordinate system representing pairs of $(1 - \mathit{specificity}, \mathit{sensitivity})$ or $(\mathit{false\ positive\ rate}, \mathit{true\ positive\ rate})$ obtained with all possible values of criterion threshold for the model. Figure 2 shows an example of an ROC curve. An error of a classifier is associated both with its false positive rate and its true positive rate. Thus, the closer the ROC curve passes near the point $(0, 1)$ (perfect sensitivity and specificity), the better the model is in discriminating the class. Similarly, the closer the curve passes near the diagonal line from $(0, 0)$ to $(1, 1)$, the weaker the classifier.

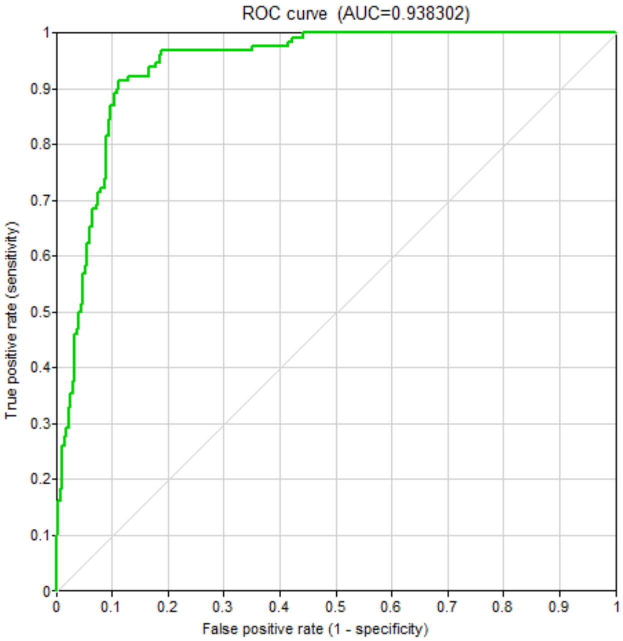


Figure 2: An example of an ROC curve generated with GeNIe software

Usually researchers report the *area under the ROC curve* (AUC) as a summary of the performance of the classifier in discriminating records associated with a chosen class. For a good classifier, the value of AUC will be close to one. One of the drawbacks of the AUC is that it does not describe the actual accuracy of the classifier.

If the analyzed class is underrepresented in the data, ROC and AUC give an insight scaled to the problem. Sometimes classes are strongly unbalanced (there are just several records in the testing dataset associated with the class compared to a few thousands for remaining classes). In such situations, AUC re-scales the problem too much, and may be considered biased (Davis and Goadrich, 2006). Some researchers suggest the use of a *precision-recall* (PR) curve (e.g., Raghavan et al., 1989) instead. The construction procedure is similar, but instead of using pairs of $(1 - \textit{specificity}, \textit{sensitivity})$, we use pairs of $(\textit{recall}, \textit{precision})$. Figure 3 shows an example of PR curve. A curve usually starts at point $(0, 1)$ – a poor recall and perfect precision – and ends around $(1, 0)$ – a perfect recall and poor precision. The closer the curve approaches $(1, 1)$, the better the model. PR curve for a perfect model will connect points $(0, 1)$, $(1, 1)$, and $(1, 0)$ with straight lines.

3.1.4 Measures of precision of posterior probabilities: calibration curve and scoring rules

As the marginal posterior probability distribution is the vital part of a probabilistic graphical model, the creator of the model pays a lot of attention to precision of its estimates. To express the precision of the posterior probability of a class, we can use a plot of calibration curve or a scoring rule.

A *calibration curve* (also called a *reliability diagram* (DeGroot and Fienberg, 1983; Murphy and Winkler, 1977)) compares the marginal posterior probability estimates of a class calculated by the model against class' empirical frequency in the test dataset. Researchers in machine learning domain use these plots to show how well the model is calibrated to output the probabilities (e.g., obtaining probability estimates based on an SVM model output (e.g., Niculescu-Mizil and Caruana, 2005), calibration of probabilities of a naive Bayes classifier (e.g., Naeini et al., 2015)). “Calibration curve” as a term seems to be more broad

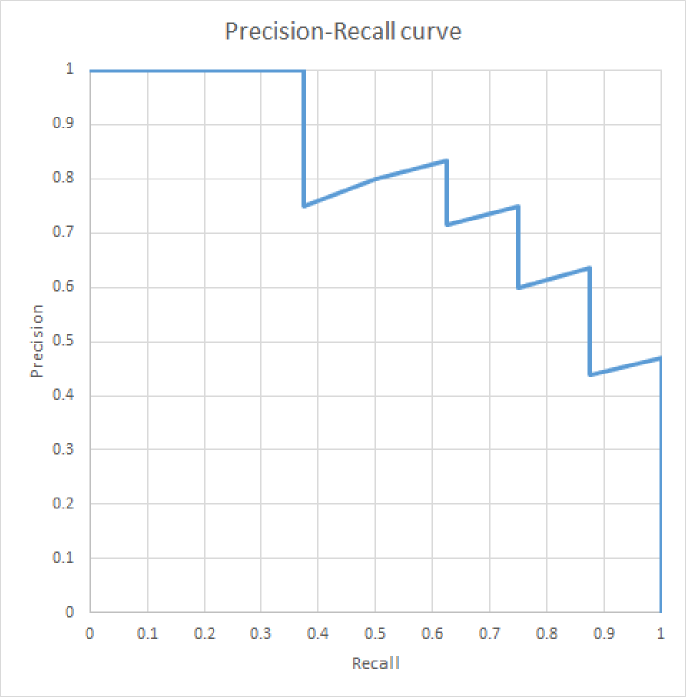


Figure 3: An example of a PR curve

and refers also to plots showing association between quantities that are not necessarily probabilities. The term “reliability diagram” seems to be more popular in terms of forecasting and probabilistic prediction, although researchers use the term “calibration curve” in this context as well (e.g., Dawid, 1982; Gould et al., 2007)). I encountered also terms “probability calibration curve,” “reliability plot,” and “confidence plot” referring to the same plot.

The calibration curve is a plot in a Cartesian coordinate system. The calibration curve is constructed with points representing posterior probabilities (horizontal axis) and the observed frequencies (vertical axis) in the data set. For each record in the testing dataset, we calculate the posterior probability estimate of the class by the model. Then we divide these records into bins using the calculated estimates. For each bin, we calculate the frequency of the true analyzed class value in the data. Then we create the plot using points (probability, frequency) corresponding to each bin. Another method for constructing a calibration curve is based on a moving average over a window instead of bins, which typically leads to a smoother plot. After calculating the posterior probability for each record in the testing dataset, we

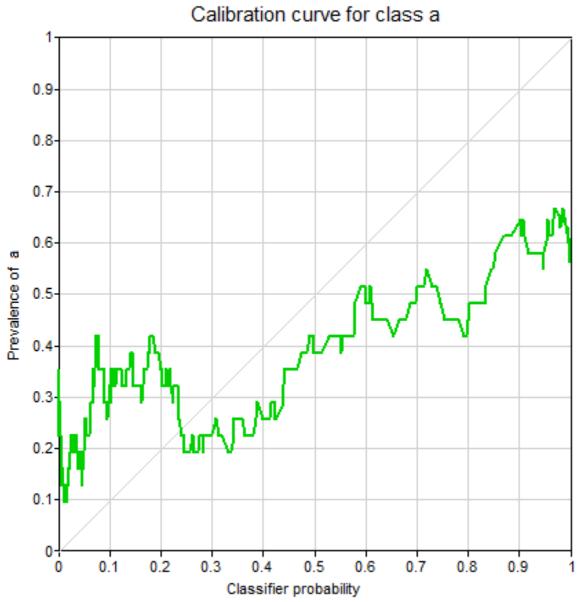


Figure 4: An example of a calibration curve generated with GeNIe software

use a window of fixed size which includes $2k+1$ records. We "slide" this window over sorted records and calculate the frequency of that class within the window. The calibration curve of a perfect model is a diagonal line connecting the points $(0,0)$ and $(1,1)$. Figure 4 shows an example of a calibration curve.

To express the overall error in estimates of posterior probability distribution obtained by the model we can use scoring functions, e.g., *Brier score* (Brier, 1950). Brier score has been originally defined as

$$\mathcal{BS} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\Pr(c_j|\mathbf{E}_i) - o_{ic_j}\right)^2 ,$$

where n is a number of records in the testing dataset, m is a number of classes, $\Pr(c_j|\mathbf{E}_i)$ is a conditional probability of class c_j calculated by the model, and

$$o_{ic_j} = \begin{cases} 1, & \text{if } i\text{th record belongs to class } c_j, \\ 0, & \text{otherwise.} \end{cases}$$

If we focus on just one class, the Brier score can be reduced to

$$\mathcal{BS}(c_j) = \frac{1}{n}\sum_{i=1}^{n}(\Pr(c_j|\mathbf{E}_i) - o_{ic_j})^2\,.$$

3.1.5 Measures of precision in modeling joint probability distributions

BN approximates the joint probability distribution over a set of variables. We would like to know how well the network reflect this distribution based on the data. One way of doing that is calculating likelihood of the data. To obtain likelihood of a BN model given the dataset, we need to calculate the probability of a case described by each record in the dataset. Then, the product of these probabilities is likelihood of the data given the BN model.

There are other scores usually applied in score-based methods for learning BNs (Cooper and Herskovits, 1992). Additionally to favoring well fitted models, these scores penalize models that are of a complex structure. Daly et al. (2011) provide an overview of most commonly used scoring functions for BN structure learning.

3.1.6 Confidence in model assessment

As the number of records in the dataset is always limited, we may want to express the uncertainty about the model assessment measures by constructing *bootstrap confidence intervals* (Efron and Tibshirani, 1993). To construct such $(1 - \alpha)$ confidence intervals, we pick multiple times a sample of N records from a dataset of N records without replacement. For each sample, we calculate the desired measure, e.g., accuracy. We sort obtained values ascending, then the confidence interval bounds are values of the indices $N \times \frac{\alpha}{2}$ and $N \times (1 - \frac{\alpha}{2})$.

We can perform similar calculations for plots generated in validation process (ROC curve and calibration curve). Instead of calculating one value for each sample, we derive an appropriate curve. Than we iterate over x values along X axis within the range $[0, 1]$. For each value x we have a corresponding set of values related to derived curves, which we use to construct a confidence interval. By integrating all of the intervals, we obtain a confidence region. For the ROC curve, we can perform the same procedure along Y axis.

Confidence intervals over assessment of a model may be an indicator of lack of significantly large testing dataset. Although it may be costly or time-consuming to obtain more data. In such situation we may look at the dynamics of change the confidence intervals and or regions as we apply gradually more data records in validation procedure (Kozniewski et al., 2016).

3.1.7 Validation with a self-sampled dataset

One of the concerns of a modeler may be misclassification of cases. Przytula et al. (2003) proposed to apply a simulation method, to obtain a confusion matrix. For each class (which can be modeled by separate variables) a sample of examples are generated from the model by means of Monte Carlo simulation (from conditional probability distribution $\Pr(\mathbf{S}|c_i)$). Next step is to classify each case in each sample. We may present the results as a confusion matrix in the form of a table, a 3-dimensional histogram plot, or a heat map. In this way we may identify problematic cases and try to improve the model.

3.2 EXPERT-BASED EVALUATION

When building a model with experts' knowledge, after initial model creation by an expert, the modeler typically performs several iterations of model adjustment, refinement, and calibration (e.g., Cypko et al., 2017; Druzdzel et al., 1999; Oniśko et al., 2000). After each iteration, experts evaluate the model or the modeler validates the model with a dataset. The techniques described above rely on the dataset. There are several techniques that can be used to evaluate the model with just the experts' help.

One of key methods that can be ran by experts is a *clarity test* (Howard, 1988). The main purpose of a clarity test is to answer the question whether all the variables (their names and outcomes) are well defined and describe clearly the phenomena that they are modeling.

Another method of validation of the model with an expert is scenario analysis. It involves analysis of either cases described by records in the dataset or assigning different outcomes to the variables and analyzing conditional probabilities of the remaining variables. As Druzdzel et al. (1999) point out, the conflicts between the model output and experts does not necessarily mean that there is a fault in the model.

Sensitivity analysis (Clemen, 1996; Coupé and Van Der Gaag, 2002; Kjærulff and van der Gaag, 2000; Laskey, 1995; Morgan et al., 1992, Chapter 5) focus on determining which parameters in the model influence the posterior probability distribution the most. The procedure employs small modification of parameters to determine the change in conditional probabilities of target variables. We are looking for these parameters that lead to largest changes in the output. Modeler has to pay more attention to eliciting these parameters to improve precision of the probability distributions over the target variables.

4.0 EXISTING MEASURES OF CONFIDENCE OF A SYSTEM FOR THE CASE AT HAND

In the previous chapter, I described some of the metrics that help to quantify the uncertainty about general model performance before its application in practice. As the model is applied, we want to know how certain the model is about the case at hand. In this chapter, I provide some of the existing metrics used to quantify this kind of confidence. Marcot (2012) reviewed some of such metrics.

4.1 CONFIDENCE MEASURES BASED ON POSTERIOR MARGINAL PROBABILITY DISTRIBUTION

A posterior marginal probability distribution $\Pr(V|\mathbf{E})$ expresses the uncertainty about the events modeled with variable V due to general uncertainty in the domain, i.e., uncertainty that is inherent to the problem. For example, if $\Pr(V|\mathbf{E}_1)$ consists of values $(0.25, 0.25, 0.25,$ $0.25)$ and $\Pr(V|\mathbf{E}_2)$ consists of values $(0.01, 0.04, 0.8, 0.15)$, it is clear that case $\mathbf{E}_2$ includes less uncertainty in V. Intuitively, the more asymmetry among the probabilities in a posterior distribution, the more information the distribution carries and less uncertain the output is.

Researchers use entropy of a distribution as a measure of uncertainty carried by the distribution. For a distribution $\Pr(V|\mathbf{E}) = (p_1, \ldots, p_N)$, entropy is defined as

$$H_{V|\mathbf{E}} = -\sum_{i=0, p_i \neq 0}^{N} p_i \ln p_i \ . \tag{4.1}$$

However, entropy is relative to the number of values in probability distribution. H gets its maximum value for uniform distribution of V, which is

$$\max_{\Pr(V)} H_V = \ln N \ .$$

H is zero as its minimum value for the distribution with one certain outcome (i.e., $\exists i, p_i = 1$). Marcot (2012) proposes to quantify unevenness in posterior probability with *posterior probability certainty index* (PPCI), which relates to complement of normalized entropy and is defined as follows.

Definition 4.1 (PPCI)**.** *Posterior probability certainty index is a complement of normalized entropy in the marginal posterior probability* $\Pr(V|\mathbf{E}) = (p_1, \ldots, p_N)$ *and can be expressed by*

$$PPCI_{V|\mathbf{E}} = (1 - \frac{H_{V|\mathbf{E}}}{\ln N}) \ , \tag{4.2}$$

where p_i *is the probability of the* i*th outcome of* V *in* $\Pr(V|\mathbf{E})$*.*

Thanks to taking complement of normalized entropy, *PPCI* ranges from zero to one, with zero representing the lowest certainty and one representing the highest certainty.

Another measure of unevenness can be *Gini impurity index* applied in decision tree learning.

Definition 4.2 (Gini impurity index)**.** *Gini impurity index can be calculated as*

$$I_G = 1 - \sum_{i=0}^{N} p_i^2 \ .$$

As I_G gives the lowest value for the distribution representing the most certain situation and the highest value is $(N-1)/N$, I will refer to its normalized complement

$$I'_G = 1 - \frac{I_G(N-1)}{N} \ . \tag{4.3}$$

Marcot (2012) proposes to use also *Gini coefficient* (Atkinson, 1970; Gastwirth, 1972) which is used to measure inequality in a distribution (e.g., in economics or ecology) and associated with the Lorenz curve, usually calculated for discretized continuous variables. He also proposes a *certainty envelope* to describe uncertainty about *PPCI* given the information about a subset of fixed values in the posterior distribution.

4.2 CONFIDENCE DUE TO IMPRECISE PARAMETERS

If the model has been build with a domain expert, the parameters elicited from the expert or learned from a small dataset may be imprecise. The precision of the model's parameters contribute to the precision of estimated posterior probability distribution. Sensitivity analysis (e.g., Kjærulff and van der Gaag, 2000; Laskey, 1995) focuses on studying the impact of the precision of parameters on the response of a DSS in the model building phase. I am more interested in the imprecision of posterior probabilities for a specific case query.

One way of expressing the imprecision of a system's response due to possible inaccuracy in the parameters are error bars (or confidence intervals) over values in the posterior probability distribution. Donald and Mengersen (2014) provided an overview of methods for constructing error bars. Some of the methods focus on estimating the parameters of the distribution over posterior probabilities (Van Allen et al., 2008). It is common to model uncertainty about parameters by Dirichlet prior distribution over parameters of the BN.

One way of deriving error bars for a query response based on Dirichlet distribution is a simulation based technique. Parameters of the prior distributions may be provided by the expert. Also, the prior distribution may be derived either form a training dataset or from a simulated dataset.

Throughout this document, I will use a simulation based method for deriving error bars over $\Pr(v_i|\mathbf{E})$. According to this method, we draw m times an assignment of parameters Θ_j from the prior distribution. For each Θ_j, we calculate the posterior probability $p_j = \Pr(v_i|\mathbf{E}, \Theta_j)$. We sort p_j values in the ascending order to get the ordered values p'_j. Than $1 - \alpha$ error bar bounds of $\Pr(v_i|\mathbf{E})$ are values of the indices $j_L = \frac{m\alpha}{2}$ and $j_H = m(1 - \frac{\alpha}{2})$.

As a measure of confidence of the query response due to parameter imprecision, we can provide the complement of the error bar width (CEBL), i.e.,

$$CEBL = 1 - EBL, \tag{4.4}$$

$$EBL = p_{j_H} - p_{j_L} \ . \tag{4.5}$$

4.3 RARITY-BASED CONFIDENCE MEASURES

The underlying motivation for measures presented in this sections is the problem of a model's self-awareness of its competence. Let us consider a patient who comes to a otolaryngologist and complains about headache. The otolaryngologist orders an X-ray to check paranasal sinuses for an inflammation and finds out that the sinuses are clear. She realizes that the problem may derive from outside of the domain of her expertise and refers the patient to a dentist to check his wisdom teeth instead. Just for the sake of argument, let us consider a decision support system that replaces the otolaryngologist's knowledge. As in case of the otolaryngologist, the model of system's knowledge is limited. The question of much interest is whether the system can realize by itself that the case at hand is outside of its domain of expertise.

One possible method to determine how the case fits the modeled domain is looking at the probability of observations of that case. Some cases have a low probability (e.g., less then 10^{10}) given the model, which means that they are atypical. In other words, knowledge represented by a Bayesian network may not cover the case – we deal with a very unlikely combination of observations, possibly even a conflict in the observations given the model (Jensen et al., 1990). A very low probability of a case suggests that the model is not suitable to perform a reasoning or the case at hand consists of noisy observations.

The probability of a case has one major disadvantage: it is relative to its domain. There may be many cases of the same probability in the same domain as the case at hand. Let us consider a sequence of 20 independent tosses of a coin. The probability of each sequence is the same and equal to $p = 2^{-20}$. For example, a result of 20 heads in a row has the same probability as the one of ten consecutive heads and then ten consecutive tails. There is no sequence of heads and tails that is less probable, which leads to the conclusion that all the sequences are common and none of them can be considered as rare.

As the probability of a case does not represent the confidence reliably, we need to introduce a measure of confidence that will take the joint probability distribution over the domain variables into account. There were several proposals of measures for conflict in the data.

We can use the joint probability of a case outcomes $\Pr(\mathbf{E})$ as a rarity measure. The problem is that the joint probability of case is relative to the set of variables that case is describing.

Laskey (1991) and Jensen et al. (1990) focus on determining conflicts among observations related to just one case, and propose conflict measures. One of the indicators of a conflict among observations is a very low probability of the case. They point out that the probability of observations may be associated with rarity of a case at hand, but may also indicate possible flaws in the model.

Another measure that refers to rarity of a case is the *surprise index* (Habbema, 1976), which can be defined as

Definition 4.3 (surprise index)**.** *Surprise index of a case* $\mathbf{E}$ *is the sum of probabilities of all cases less probable then* $\mathbf{E}$ *that are instantiations of* $Val(\mathbf{E})$*, i.e.,*

$$\mathcal{SI}(\mathbf{E}) = \sum_{\mathbf{E}_i \,:\, \Pr(\mathbf{E}_i) < \Pr(\mathbf{E})} \Pr(\mathbf{E}_i)\ . \tag{4.6}$$

Surprise index measures how typical the case is within its domain. If less probable cases cover just a small part of the probability space, our case is rare.

Exact calculation of the surprise index is intractable in practice, as it requires multiple calculations of $\Pr(\mathbf{E}_i)$. I will show two methods for approximating surprise index in Chapter 5.

Jensen et al. (1990) proposed an indicator of conflict in the observation set, which can be considered as a measure of confidence. The indicator of conflict compares the probability of case observations $\mathbf{E} = \{e_1, e_2, \ldots, e_k\}$ to the product of marginal probabilities of these observations, i.e.,

$$c_J(\mathbf{E}) = \log \frac{\prod\limits_{e_i} \Pr(e_i)}{\Pr(\mathbf{E})}\ . \tag{4.7}$$

As proposed by Jensen et al. (1990), the $c_J(\mathbf{E})$ has a positive value when the case $\mathbf{E}$ is rare.

Krüger and Hirschhäuser (2009) proposed measures of conflict in the set of observations from different sources of information. They propose to use a distance measure between, so called, likelihood vectors, which consist of conditional probabilities of events given the outcome of a class variable.

5.0 AN APPROXIMATION OF THE SURPRISE INDEX

In this chapter, I will focus on the concept of surprise index (mentioned in Section 4.3) along with its approximation.

Druzdzel (1994) proposed to analyze the distribution of logarithms of probabilities in JPD encoded by BN. Each value in JPD encoded by BN ($\mathcal{G}(\mathbf{V},\mathbf{E}),\mathbf{\Theta}$) refers to a probability of a full scenario. Let us consider an evidence set $\mathbf{E}_0$ which assigns values to subset $\mathbf{V}_\mathbf{E} \subset \mathbf{V}$. Let us consider a variable $X_{p\log p}$ that assigns a probability $\Pr(\mathbf{E})$ to outcome $\log\Pr(\mathbf{E})$, for all $\mathbf{E} \in \mathit{Val}(\mathbf{V}_\mathbf{E})$. It may happen that there are m cases of the same probability in JPD of $\mathbf{V}_\mathbf{E}$, then $\Pr_{p\log p}(\log\Pr(\mathbf{E})) = m\Pr(\mathbf{E})$. If we know the distribution of $X_{p\log p}$, the computation of the surprise index of $\mathbf{E}_0$ becomes straightforward. The surprise index of $\mathbf{E}_0$ reduces to calculation of the value of cumulative distribution function of $X_{p\log p}$. It is possible due to the fact that the logarithmic mapping is monotonic, thus it maintains the $<$ relation between values. For the case $\mathbf{E}_0$, we have

$$\mathcal{SI}(\mathbf{E}_0) = F_{X_{p\log p}}(\log(\Pr(\mathbf{E}_0)))\ .$$

In the following sections, I will show how JPD can be analyzed to approximate the $F_{X_{p\log p}}(x)$.

5.1 DISTRIBUTION OF PROBABILITIES IN JOINT PROBABILITY DISTRIBUTION

Druzdzel (1994) analyzed the distribution of logarithms of probabilities in JPD regardless of their relevance. Let us consider a random variable $X_{\log p}$ with logarithms of probabilities of full scenarios ($\log \Pr(\mathbf{E_V})$) as outcomes and associated equal probabilities $\Pr_{\log p}(\log \Pr(\mathbf{E_V}))$ $= \frac{1}{M}$, where $\mathbf{E_V} \in \mathit{Val}(\mathbf{V})$ and $M = |\,\mathit{Val}(\mathbf{V})|$. It may happen that there are m full scenarios of the same probability in JPD, then $\Pr_{\log p}(\log \Pr(\mathbf{E_V})) = \frac{m}{M}$.

Using factorization of probability (2.2), $\log \Pr(\mathbf{E_V})$ becomes a sum of logarithms of conditional probabilities, i.e.,

$$\log \Pr(\mathbf{E_V}) = \log \prod_{i=1}^{n} \theta_{v_{ij}|c_k} = \sum_{i=1}^{n} \log \theta_{v_{ij}|c_k} = \sum_{i=1}^{n} q_{ijk} \ , \tag{5.1}$$

where $q_{ijk} = \log \theta_{v_{ij}|c_k}$ is a logarithm of a conditional probability $\Pr(v_{j_i}|c_k)$, where c_k is kth joint assignment to parent variables of V_i. Druzdzel (1994) makes an argument that for a sufficiently sparse and sufficiently large BN we can apply central limit theorem to conclude that $X_{\log p}$ follows the normal distribution. Figure 5 shows a histogram of a sample of 1,000,000 values drawn from $X_{\log p}$ associated with HEPAR II model[1] for supporting diagnosis of liver disorders (Oniśko et al., 2001) consisting of 70 variables. We can see that the probabilities at the sample follow the normal distribution, at least approximately.

5.2 PARAMETERS OF THE LOGNORMAL DISTRIBUTION OF PROBABILITIES

It is possible to calculate the expected value and the variance of $X_{\log p}$ distribution for a given BN. Bouckaert et al. (1996) gave a set of formulas for calculating the exact values of $\mu_{\log p}$ and $\sigma_{\log p}$, which have some limitations. Due to commutativity of addition, the derivation of $\mu_{\log p}$ reduces to

$$\begin{aligned} \mu_{\log p} &= \frac{1}{M} \sum_{y=1}^{M} \sum_{i=1}^{n} q_{ij_y k_y} && (5.2) \\ &= \frac{1}{M} \sum_{i=1}^{n} \frac{M}{w_i} \sum_{z=1}^{w_i} q_{ij_z k_z} \\ &= \sum_{i=1}^{n} \frac{1}{w_i} \sum_{z=1}^{w_i} q_{ij_z k_z} \ , && (5.3) \end{aligned}$$

[1] Available through several public Bayesian network repositories.

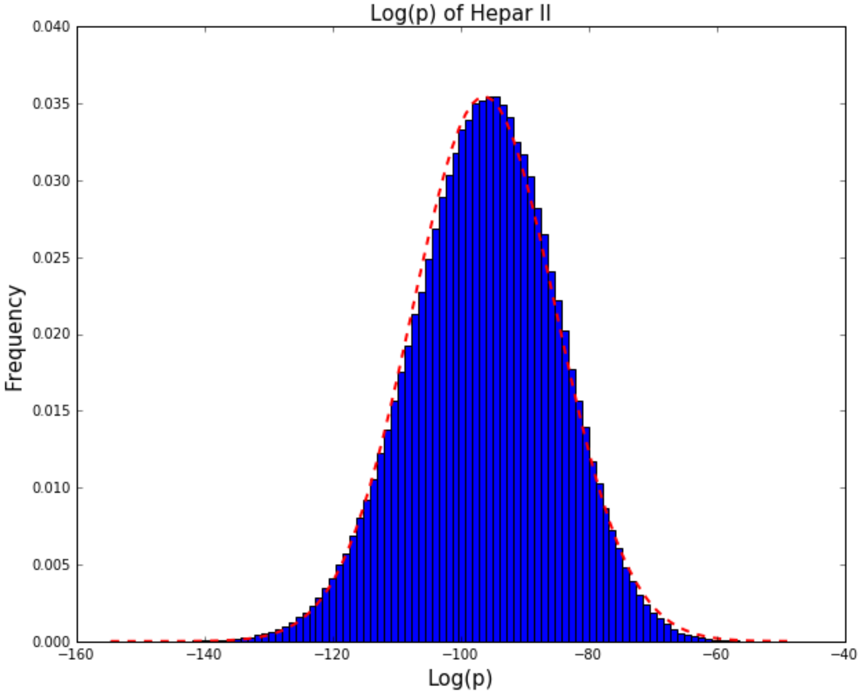


Figure 5: Histogram of a sample of 1,000,000 values from $X_{\log p}$ associated with the Hepar II model

where w_i is the size of the conditional probability table of variable V_i. Computation of variance is more complicated and is obtained based on the property $\mathrm{Var}(X) = \mathrm{E}(X^2) - \mathrm{E}^2(X)$.

$$\begin{aligned}
\sigma^2_{\log p} &= \frac{1}{M}\sum_{y=1}^{M} (\log \Pr(\mathbf{E_V}))^2 - \mu^2_{\log p} \\
&= \frac{1}{M}\sum_{y=1}^{M} \left(\sum_{i=1}^{n} q_{ij_yk_y}\right)^2 - \mu^2_{\log p} \\
&= \frac{1}{M}\sum_{y=1}^{M}\sum_{i=1}^{n} q^2_{ij_yk_y} + \frac{2}{M}\sum_{y=1}^{M}\left(\sum_{i=1}^{n-1} q_{ij_yk_y} \sum_{r=i+1}^{n} q_{rs_yt_y}\right) - \mu^2_{\log p} \; . \qquad (5.4)
\end{aligned}$$

We can simplify the first term in formula (5.4) using the same approach as in (5.2). The second term can be simplified to

$$\begin{aligned}
\sigma' &= \frac{2}{M}\sum_{y=1}^{M}\left(\sum_{i=1}^{n-1} q_{ij_yk_y} \sum_{r=i+1}^{n} q_{rs_yt_y}\right) \\
&= \sum_{i=1}^{n-1}\sum_{r=i+1}^{n} \frac{2}{M}\sum_{y=1}^{M} q_{ij_yk_y} q_{rs_yt_y} \\
&= \sum_{i=1}^{n-1}\sum_{r=i+1}^{n} \frac{2}{M}\sum_{y=1}^{w^*_{ir}} \frac{M}{w^*_{ir}} q_{ij_yk_y} q_{rs_yt_y} \\
&= \sum_{i=1}^{n-1}\sum_{r=i+1}^{n} \frac{2}{w^*_{ir}}\sum_{y=1}^{w^*_{ir}} q_{ij_yk_y} q_{rs_yt_y} \; , \qquad (5.5)
\end{aligned}$$

where w^*_{ir} is a number of distinct $q_{i\bullet\bullet}$ and $q_{r\bullet\bullet}$ pairs appearing together in the factorizations of a probability in JPD. If V_i and V_r have common parents or one is a parent of the other, which can be expressed as

$$(V_i \cup \mathrm{Pa}(V_i)) \cap (V_r \cup \mathrm{Pa}(V_r)) \neq \emptyset \; ,$$

the inner sum in (5.5) can be simplified to

$$
\begin{aligned}
\sigma'(i,r) &= \frac{2}{w_i w_r} \sum_{y=1}^{w_i w_r} q_{ij_y k_y} q_{rs_y t_y} \\
&= \frac{2}{w_i w_r} \sum_{y_1=1}^{w_i} q_{ij_{y_1} k_{y_1}} \sum_{y_2=1}^{w_r} q_{rs_{y_2} t_{y_2}} \\
&= 2 \left(\frac{1}{w_i} \sum_{y_1=1}^{w_i} q_{ij_{y_1} k_{y_1}} \right) \left(\frac{1}{w_r} \sum_{y_2=1}^{w_r} q_{rs_{y_2} t_{y_2}} \right) \\
&= 2\mu_i \mu_r \,,
\end{aligned}
$$

where μ_i is the average of logarithms of $\theta_{v_{i,\bullet}|\bullet}$ parameters associated with CPT of variable V_i.

Computational complexity of calculating $\mu_{\log p}$ is $O(nw)$, where n is number of variables in the BN and w is the size of the largest CPT in the BN. $\sigma_{\log p}$ can be computed in $O((nw)^2)$ time.

A major limitation of the formulas presented in this section is that we need to assume that none of the parameters Θ is zero (i.e., $\forall \theta_{v_{ij}|c_k} \neq 0$). When the number of zeros among parameters $\boldsymbol{\Theta}$ is small, we need to omit appropriate $q_{ij_z k_z}$ expressions and propagate necessary information to create a weight for expressions $q_{ij_z k_z}$ in other CPTs. Such solution complicates the calculation significantly.

One of the conclusions that Druzdzel (1994) makes is that the distribution of $X_{\log p}$ may get skewed as there is some skewness in some of the CPTs – some portion of the values are much lower then others, e.g., 0.000001.

If probabilities of full scenarios of a given BN are distributed lognormally, the same applies to cases $\mathbf{E}$ that consist of assignments of a sufficiently large subset of $\mathbf{V}$ associated with that BN. To calculate parameters of the distribution $X_{\log p}$ related to $\mathbf{V_E}$, we need to obtain a new network by marginalizing out unnecessary variables. It may happen that by marginalizing out a variable we introduce additional edges to maintain original dependencies. In this case, additional edges do not influence the conditions of the central limit theorem. The argument here is that by dropping a variable (taking the same network with one marginalized variable) we are preserving the dependencies among the remaining variables, which were weak enough to remove these edges from the initial network. Unfortunately, adding more edges leads to an exponential growth of CPTs, which may lead to intractable computation of parameters $\mu_{\log p}$ and $\sigma_{\log p}$.

5.3 DISTRIBUTION OF RELEVANT PROBABILITIES

Most of the individual probabilities in the JPD contribute just a little to probability mass of JPD. To express how much each probability value contributes to probability mass, we consider a variable $X_{p \log p}$ which assigns a probability $\Pr(\mathbf{E})$ to outcome $\log(\Pr(\mathbf{E}))$.

Druzdzel (1994) showed that if $X_{\log p}$ follows the normal distribution $\mathcal{N}(\mu_{\log p}, \sigma_{\log p})$, then $X_{p \log p}$ relates to the normal distribution $\mathcal{N}(\mu_{\log p} + \sigma^2_{\log p}, \sigma_{\log p})$. Because probabilities are fractions between zero and one, their logarithms cannot be larger than zero and we need to "cut" the right side of the distribution and normalize the remaining part that remains below zero.

Bouckaert et al. (1996) pointed out that the approximation with shifted distribution may become unreliable when $2\sigma^2_{\log p} > -\mu_{\log p}$. They do not provide any explanation of the formula. The problem lies in violation of at least two conditions. The first problem is that $X_{\log p}$ is discrete in its nature and has a finite number of outcomes. The second problem is that the distribution of $X_{p \log p}$ is an amplified tail of $X_{\log p}$. $X_{p \log p}$ represents the probabilities that are important and modeled with BN. It may happen that the largest probabilities in JPD are still very small due to characteristics of the modeled domain.

Zagorecki et al. (2015) proposed to calculate the exact values of $\mu_{p \log p}$ and $\sigma_{p \log p}$ of $X_{p \log p}$ directly from the BN. For many BNs, $X_{p \log p}$ is quite close to the normal distribution, despite violations of conditions mentioned above. This approximation does not necessarily have to be accurate especially for evidence sets of extremely small probabilities, e.g., $P(\mathbf{E}) < \mu_{p \log p} - 3\sigma_{p \log p}$.

5.4 SURPRISE INDEX APPROXIMATION BY MEANS OF SAMPLING

Mathematically speaking, the surprise index of a case is just a value of the cumulative distribution function (CDF) over values of probabilities of cases given the model. We can sample the values from $X_{p\log p}$ associated with the BN. Then we can approximate the surprise index by taking a fraction of cases that are less probable than $\mathbf{E}_0$.

When the value of the surprise index is small, we can improve the approximation employing an approximated tail of the normal distribution or by means of the extreme value theory. Castillo et al. (1998) presented one of the methods of approximating the left tail of $X_{p\log p}$ distribution by means of generalized Pareto distribution. Castillo et al. (2005) elaborates on methods for approximating the tails of distributions with generalized Pareto distribution.

If the surprise index is considered to be applied as a measure of confidence of a DSS, there may be no need to approximate accurately its small values. Small differences may be irrelevant for the end user of the system.

6.0 CONFIDENCE INTERVALS FOR POSTERIOR PROBABILITIES IN ANTICIPATION OF FUTURE OBSERVATIONS

The posterior probability distributions over variables of interest change as we gather observations about a case at hand. Each new observation introduces information that usually makes the probability estimate more case-specific and, hence, more precise. A user applying the model may want to know, how future observations will impact the model's result. For example, a physician investigating a case of a patient with a chest pain may consider running some clinical tests after gathering information about patient's medical history and listening to patient's lungs. A question of much interest is whether the probability of pneumonia can go up or down and by how much as we obtain the results of the clinical tests. In other words, how will the posterior probability of pneumonia change when we feed the model with more observations about the patient case at hand.

One way of representing the uncertainty about a calculated quantity (this is, in case of a BN model, a posterior probability) is a confidence interval, which utilizes the probability distribution over the predicted value. Given that a BN is a complete specification of the joint probability distribution over its variables, we have all the necessary information to derive such intervals.

Most of the literature on uncertainty in results of Bayesian network inference focuses on the impact of possible imprecision in parameters of the network. Such uncertainty can be captured by means of error bars or uncertainty intervals (e.g., work by Donald and Mengersen (2014) or Van Allen et al. (2008)). If the imprecision in parameters can be expressed by intervals, it can be propagated over the model to derive uncertainty intervals over results (Cano et al., 1993; Fagiuoli and Zaffalon, 1998). Uncertainty over results has also been a focus of sensitivity analysis, which amounts to studying the impact of small changes in individual model parameters on the result. For example, Laskey (1995) describes the derivation of error bars for probability assessment. Even though the question posed in this paper is useful and asked by users of probabilistic decision support systems, we have not found any literature analyzing the uncertainty intervals for posterior probabilities in anticipation of future observations.

In this chapter I present a method for deriving uncertainty (variation) intervals over posterior probabilities due to unknown observations about the case. The starting point for this work is a BN model, and we assume that both its structure and its parameters are correct. Because the distribution over possible values of posterior probabilities given different observations is not necessarily parametric, I propose to use an empirical distribution. The number of possible combinations of observations is typically too large to analyze. In such situation, we simulate the observations by means of a stochastic sampling method based on posterior probability distributions over unobserved variables.

The remainder of the chapter is structured as follows. Section 6.1 introduces notation and necessary definitions. Section 6.2 describes two simulation methods for deriving the variation intervals over posterior probabilities. demonstrates the behaviour of variation intervals and compares methods for obtaining them. Section 6.4 concludes the section with final remarks and discussion.

6.1 VARIATION INTERVALS OVER FUTURE PROBABILITIES

We are interested in anticipated changes in the posterior probability of a target variable due to possible future observations consistent with the evidence $\mathbf{E}$ at hand. Determining all possible future observations would require analyzing all possible scenarios $\mathbf{E}^* \supset \mathbf{E}$. Analyzing all these scenarios for a large model may be daunting. For example, the Hepar II model[1] for supporting diagnosis of liver disorders (Oniśko et al., 2001) consists of 70 variables of which 61 are observable. The size of the complete set of scenarios for Hepar II is over 3.78215×10^{21}.

[1] Available through several public Bayesian network repositories.

In such a case, we can derive a sample of scenarios as described below. For a given evidence set $\mathbf{E}$, we obtain possible future observations by stochastic simulation, i.e., we draw outcomes from the posterior probability distribution of each observable variable in $\mathbf{S}$ to obtain a possible scenario of observations $\mathbf{E}^*$. We can repeat the simulation to get a sample of possible scenarios $\{\mathbf{E}_1^*, \ldots, \mathbf{E}_s^*, \ldots, \mathbf{E}_N^*\}$. If we calculate the posterior probabilities of an outcome of a target variable given each scenario (e.g., $\Pr(Bronchitis = present|\mathbf{E}_s^*)$), we will obtain a sample of possible future probabilities of that outcome.

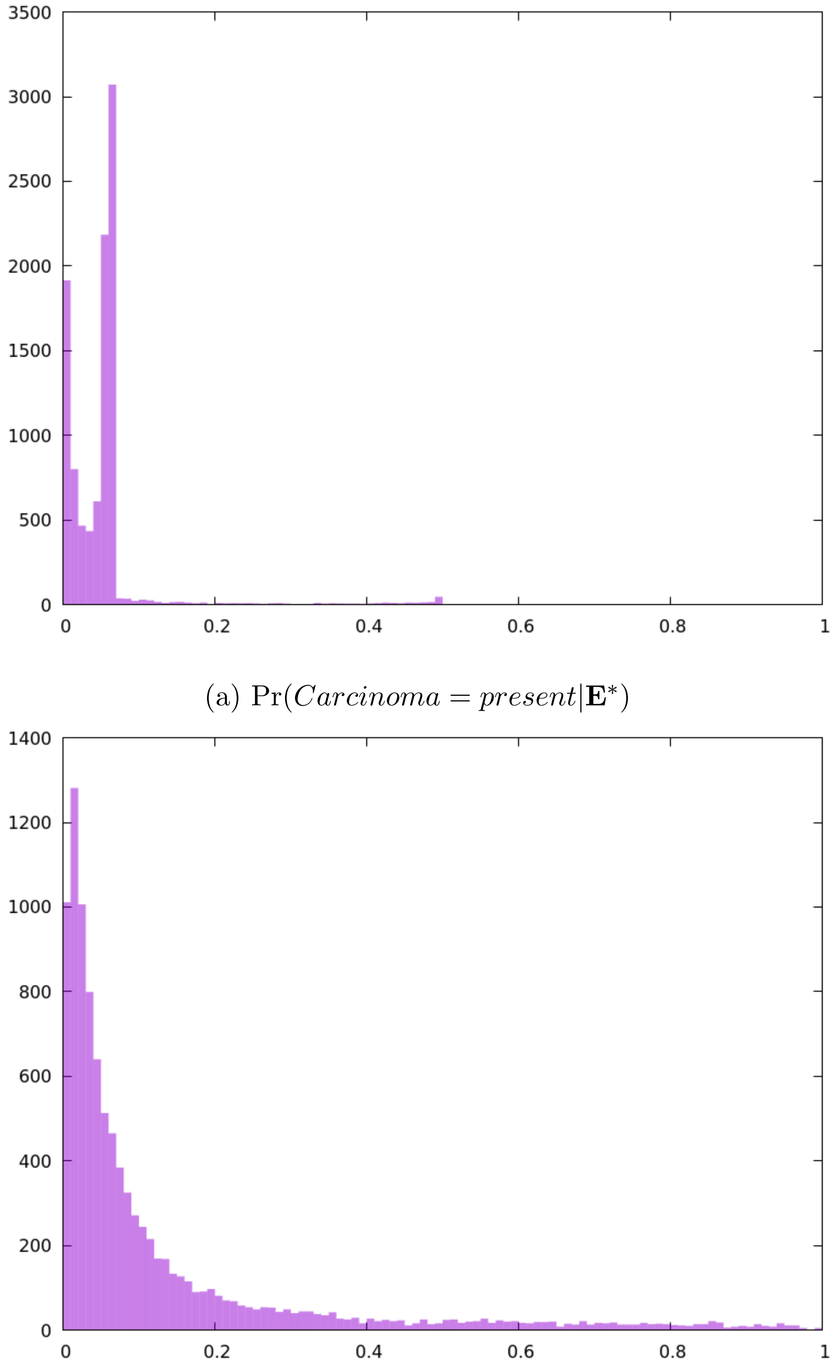
Figure 6 shows two histograms of posterior probability of assignments to two target variables in the Hepar II model, $\Pr(Carcinoma = present|\mathbf{E}^*)$ (a) and $\Pr(Chronic\,Hepatitis = active|\mathbf{E}^*)$ (b). Both histograms were generated by sampling (as described above) with the evidence set $\mathbf{E} = \{Hepatitis\,B\,Antigen = absent\}$.

Histograms such as those pictured in Figure 6 show typically a wide spread. For example, the values in the histogram (b) cover the entire range $(0, 1)$. It seems that reporting the range of possible values is, therefore, quite useless. Because both histograms show some central tendency, a trimmed range (for example, one showing 95% of all values) will be more informative. To this effect, we can trim the extreme 2.5% of sampled values at each end. The precise cut-off points can be interpreted as a numerical estimate of the 95% confidence interval over the current value of the target probability calculated by the model in the light of future observations.

6.2 CALCULATION OF THE VARIATION INTERVAL OVER FUTURE POSTERIOR PROBABILITIES

In this section, I formalize the procedure described in Section 6.1 by proposing two methods for sampling the possible posterior probabilities in anticipation of possible observations. The first method (Algorithm 1) is based on exhaustive instantiating of all observable variables. I follow this by an improved approach (Algorithm 2) that narrows down the number of sampled variables to the extended Markov blanket of a target variable.

Algorithm 1 iterates through the set of all observable variables to assign a value to each



(a) $\Pr(Carcinoma = present|\mathbf{E}^*)$

(b) $\Pr(Chronic\,Hepatitis = active|\mathbf{E}^*)$

Figure 6: Histograms representing samples of posterior probabilities values given one assignment to a variable in Hepar II model

unobserved variable (line 4). To draw an outcome for a variable, it calculates the posterior probability distribution over its outcomes given the evidence (line 5). Then, it samples an outcome from the calculated posterior probability distribution (line 6). Having outcomes assigned to all the observable variables, the algorithm calculates the posterior probability of the pursued outcome of the target variable, which amounts to one sample (lines 9-10). Based on the sample we derive a confidence interval over the posterior probability of the pursued outcome (line 12).

```
VISampleAllObservable
Input  : BN $(\mathcal{G}, \Theta)$, target variable $V_t$, target assignment $v_{t,j}$, evidence $\mathbf{E}$,
         unobserved variables $\mathbf{S}_U$, number of samples $N$, confidence level $1 - \alpha$
Output: Sample $H$ of possible probabilities $\Pr(v_{t,j}|\mathbf{E}^*)$, variation interval $(p_L, p_U)$
1  $H \leftarrow \emptyset$
2  for  $k = 1, \ldots, N$ do
3      $\mathbf{E}^* \leftarrow \mathbf{E}$
4      foreach  $V_i \in \mathbf{S}_U$ do
5          Calculate $\Pr(V_i|\mathbf{E}^*)$
6          Draw $v_{i,k} \sim \Pr(V_i|\mathbf{E}^*)$
7          $\mathbf{E}^* \leftarrow \mathbf{E}^* \cup \{v_{i,k}\}$
8      end
9      Calculate $\Pr(V_t|\mathbf{E}^*)$
10     $H \leftarrow (H, \Pr(v_{t,j}|\mathbf{E}^*))$
11 end
12 Construct $1 - \alpha$ variation interval $(p_L, p_U)$ using sample $H$
```

Algorithm 1: The algorithm for deriving the variation interval for posterior probability values by sampling the space of assignments of all unobserved variables

Each calculation of the marginal posterior probability distribution of a variable involves a call to a Bayesian network inference algorithm. Each derivation of the variation interval involves $O(N \times (|\mathbf{S}| - |\mathbf{E}|))$ calls of the inference algorithm, where N describes the number of samples, $|\mathbf{S}|$ is the number of observable variables, and $|\mathbf{E}|$ is the number of observations. Probabilistic inference is worst-case NP-hard (Cooper, 1990) and even with the fastest algorithm available may turn out to be too slow for interactive systems.

Generation of samples in Algorithm 1 can be improved by exploring independence between the target variable and other variables conditional on the target variable's Markov blanket. Because in practice not all model variables are observable, we use the concept of the extended Markov blanket, introduced in Chapter 2. Extended Markov blanket screens off the target variable given a minimal set of those variables that are observable. This mitigates the problem of multiple calls to Bayesian network inference algorithm by reducing the set of sampled variables to those in the extended Markov blanket of the target variable.

Algorithm 2 starts with determining the extended Markov blanket of the target variable (lines 1-10). In particular, we create two sets to store unprocessed ($\mathbf{A}$) and processed ($\mathbf{A}_D$) non-observable variables. After initialization (lines 1-3), we are recursively collecting variables from Markov blanket $\mathbf{M}(V_i)$ (lines 8-9) of a variable $V_i \in \mathbf{A}$ and moving V_i to the set $\mathbf{A}_D$ (lines 6-7). The remainder of the algorithm (lines 11-22) is similar to Algorithm 1, except for line 14, where we replaced $\mathbf{S}_U$ by $\mathbf{M}^*(V_t) \setminus \mathbf{S}_O$. As a result, Algorithm 2 involves $O(N \times (|\mathbf{M}^*(V_t) \setminus \mathbf{S}_O|))$ calls to the inference algorithm.

6.3 DEMONSTRATION AND EVALUATION OF THE PROPOSED METHOD

I applied our algorithms for calculating the 95% variation intervals over the posterior marginal probability of a target outcome to three practical Bayesian network models described below.

Hepar II is a Bayesian network model for diagnosis of liver disorders (Oniśko et al., 2001), available from several public Bayesian network repositories. Hepar II consists of 70 variables, arranged in three groups: patient history and risk factors (18 variables), diseases (9 target variables), and symptoms or test results (43 variables). Hepar II's graph models the causal structure of the domain. For our tests, we picked various target variables from among the nine disease variables.

Mortality90d is a Bayesian network model for forecasting mortality of patients 90 days after heart transplant (Kanwar et al., 2017). The structure of Mortality90d follows a Tree-

```
VISampleExtendedMarkovBlanket
Input  : BN $(\mathcal{G}, \Theta)$, target variable $V_t$, target assignment $v_{t,j}$, evidence $\mathbf{E}$, observable
         variables $\mathbf{S}$, number of samples $N$, confidence level $1 - \alpha$
Output: Sample $H$ of possible probabilities $\Pr(v_{t,j}|\mathbf{E}^*)$, variation interval $(p_L, p_U)$
$\mathbf{M}^*(V_t) \leftarrow \mathbf{M}(V_t) \cap \mathbf{S}$
$\mathbf{A} \leftarrow \mathbf{M}(V_t) \setminus \mathbf{S}$
$\mathbf{A}_D \leftarrow \emptyset$
while $\mathbf{A} \neq \emptyset$ do
    pick any $V_i$ from $\mathbf{A}$
    $\mathbf{A} \leftarrow \mathbf{A} \setminus \{V_i\}$
    $\mathbf{A}_D \leftarrow \mathbf{A}_D \cup \{V_i\}$
    $\mathbf{A} \leftarrow \mathbf{A} \cup (\mathbf{M}(V_i) \setminus (\mathbf{S} \cup \mathbf{A}_D))$
    $\mathbf{M}^*(V_t) \leftarrow \mathbf{M}^*(V_t) \cup (\mathbf{M}(V_i) \cap \mathbf{S})$
end
$H \leftarrow \emptyset$
for $k = 1, \ldots, N$ do
    $\mathbf{E}^* \leftarrow \mathbf{E}$
    foreach $V_i \in \mathbf{M}^*(V_t) \setminus \mathbf{S}_O$ do
        Calculate $\Pr(V_i|\mathbf{E}^*)$
        Draw $v_{i,k} \sim \Pr(V_i|\mathbf{E}^*)$
        $\mathbf{E}^* \leftarrow \mathbf{E}^* \cup \{v_{i,k}\}$
    end
    Calculate $\Pr(V_t|\mathbf{E}^*)$
    $H \leftarrow (H, \Pr(v_{t,j}|\mathbf{E}^*))$
end
Construct $1 - \alpha$ variation interval $(p_L, p_U)$ using sample $H$
```

Algorithm 2: The Algorithm for deriving the variation interval for posterior probability values by instantiating variables of the extended Markov blanket of the target variable.

augmented Naïve Bayes (TAN) model with one class variable representing *mortality* and 27 predictor variables. The TAN structure forces two types of edges: connecting *mortality* with all predictor variables and those forming a tree structure among all predictor variables. The Markov blanket of *mortality* consists of all predictor variables.

CPCS179 is a Bayesian network model created from the knowledge base of Computer-based Patient Case Simulation (CPCS) system (Pradhan et al., 1994). CPCS179 consists of 179 variables connected by 239 edges and, similarly to Hepar II, its graph follows the causal structure of the domain. We treat this model as an example of a sizable Bayesian network. We chose the following two variables as targets for our tests: *Alcoholic Hepatitis*, with one parent variable and 26 children variables, and *Cholestasis*, with one parent variable and 14 children variables. We treated the remaining variables as observable.

6.3.1 Examples of the derived variation intervals

To demonstrate the usefulness and practical behavior of the variation intervals over future observations, we performed several simulations of a diagnostic process using the Hepar II model (we used a handful of real patient cases from a data set used for learning the parameters of the Hepar II model). For each target variable V_t and an evidence set $\mathbf{E}_i$, we followed the following procedure:

1. From the set of unobserved variables, choose the variable that carries the most information measured by cross-entropy for target V_t given already observed values. This gave us a realistic order of observations during the diagnostic process: from the most to the least informative evidence.
2. Enter the observation from the evidence set $\mathbf{E}_i$ for the chosen variable into the model.
3. Calculate the posterior marginal probability distributions of the target variables.
4. Derive variation intervals for those probabilities.
5. Repeat all these steps until all observations belonging to evidence set $\mathbf{E}_i$ have been made.

Figure 7 shows eight examples of 95% variation intervals over the posterior probability of *Chronic Hepatitis* being persistent (a), *Chronic Hepatitis* being active for two different cases (b-c), *PBC* (primary biliary cirrhosis) (d) being present, *Toxic Hepatitis* being present (d), *Cirrhosis* being compensated for three different cases (f-h). There are 61 possible observations (referring to risk factors, symptoms, and test results in the Hepar II model) for each case and they are made individually from left to right. We used a fixed number of $N = 1,000$ samples in each experiment. The solid line running from left to right demonstrates the development of the probability of the target event in question as new observations are made. The area around the probability line shows the variation interval over the probability at each point in time. Please note that the variation intervals start by being very wide in the beginning, which corresponds to the situation when nothing about the patient is known. As more and more evidence is accumulated, the variation intervals narrow, to the point of becoming either a point probability (when all possible 61 observations have been made) or a fixed interval, when some of the observations have never been made in a patient's case.

6.3.2 Computation time

To compare the computation time of the two proposed algorithms, for each of the three models we generated 100 test records containing values of the observable variables. We used a version of probabilistic logic sampling (Henrion, 1988), making sure that 50% of all values are missing at random. For each record in the generated data sets, we derived 95% confidence interval of posterior probability of one target variable (randomly chosen among targets in the model), using both Algorithm 1 and Algorithm 2. We ran our tests on a computer with Intel® Core™ i5-5200U CPU @ 2.20GHz processor, 8GiB memory, 32KiB/256KiB/3MiB processor cache, running Ubuntu Linux 16.04.1 LTS x86-64 distribution. The implementation used SMILE (BayesFusion, LLC, 2019) Bayesian network software library.

Figure 8 shows box plots representing time spent by each of the algorithms. For the Mortality90d model (tree augmented naïve Bayes), derivation of confidence intervals takes similar amount of time. This is understandable given that the Markov blanket of the target variable in a TAN model consists of all remaining variables and Algorithm 2 practically deteriorates into Algorithm 1. For both, the Hepar II and CPCS179 models, Algorithm 2 is much faster ($p < 10^{-57}$ for Hepar II model and $p < 10^{-115}$ for CPCS179 model), as it takes advantage of the extended Markov blankets of the target variables. In all three cases,

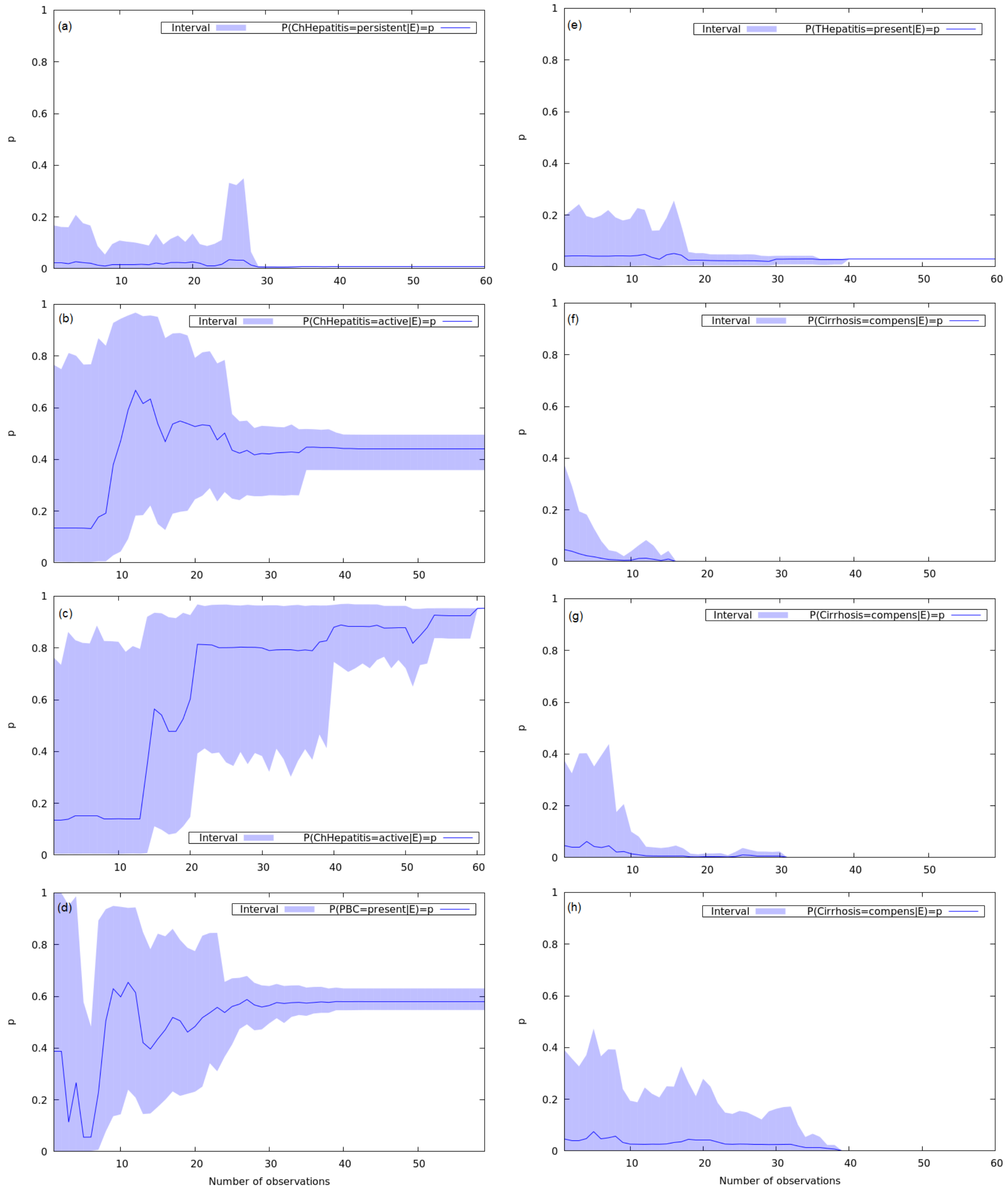


Figure 7: Examples of 95% variation intervals over the posterior probability of *Chronic Hepatitis* being persistent (a), *Chronic Hepatitis* being active (b-c), *PBC* (primary biliary cirrhosis) being present (d), *Toxic Hepatitis* being present (e), end *Cirrhosis* being compensated (f-h) in the Hepar II model

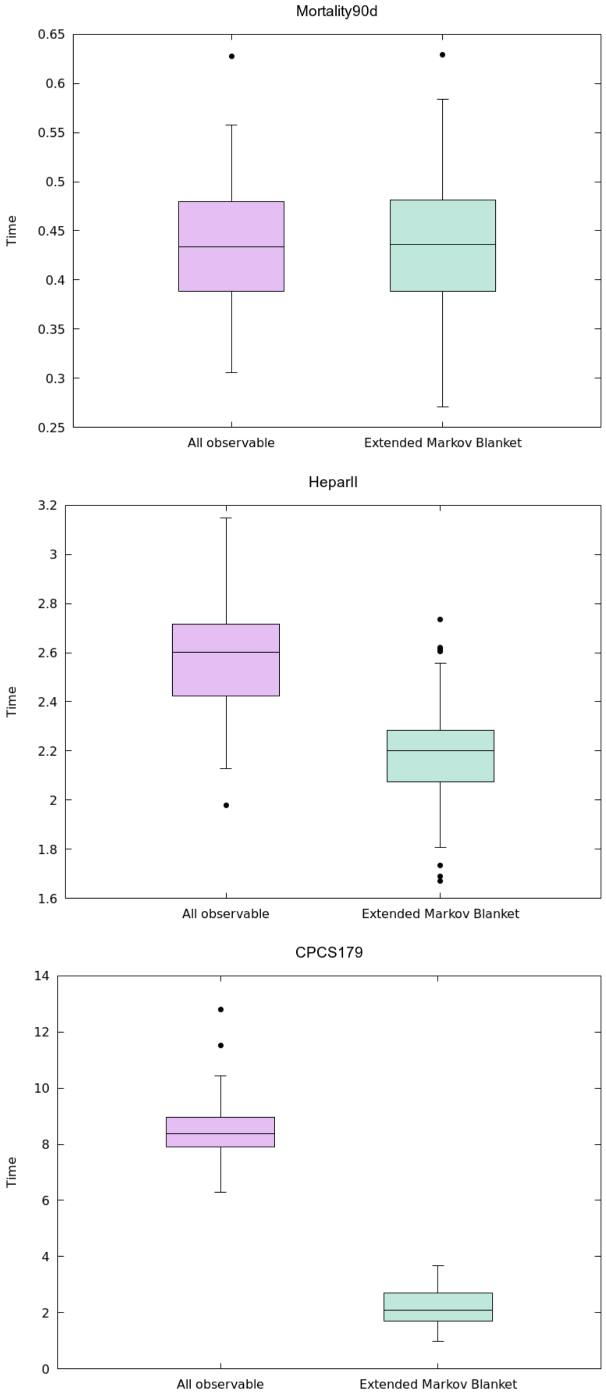


Figure 8: Box plots comparing computation times of confidence intervals for posterior probabilities with both versions of the algorithm (measured in seconds).

the absolute computation time seems acceptable from the point of view of an interactive user interface.

6.4 CONFIDENCE MEASURE IN ANTICIPATION OF FUTURE OBSERVATIONS

Similarly to error bars, we can take the length of the constructed variation interval and treat it as a measure of confidence for a given case.

As we apply the model to cases in classification, we use a decision rule that typically assigns a particular class to a case, when its posterior probability exceeds some threshold p_0. In such situation, we can perform similar analysis of possible future values of posterior probability and determine the probability of changing the decision (fraction of sampled posterior probabilities on the same side of threshold p_0 as current posterior probability).

7.0 HOW DIFFERENT CONFIDENCE MEASURES RELATE TO EACH OTHER

Different measures of confidence may provide similar information about the case at hand, even though they are attributed to different sources of uncertainty. For example, while dealing with a rare case, our model may not be well defined for that case. In such situation, the posterior probability distribution may be vaguely defined, which can be observed in the form of wide error bars.

In this chapter, I look into the relationships among the measures presented in this dissertation. For several models I run a simulation of many possible cases, for which I calculate the confidence measures. I present the results in form of scatter plots involving various confidence measures.

This chapter is organized as follows. Section 7.1 introduces the models used and describes the simulation procedure. Section 7.2 presents the most interesting observations about the measures. I conclude this chapter with Section 7.3, where I summarize and make additional comments on the observations I made in Section 7.2.

7.1 SIMULATION SETUP

In the simulation, I used seven BN models: five models from BayesFusion Model Repository,[1] one model (Mortality90d) used in prediction of mortality in CORA system (Kanwar et al., 2017), and one model (HV) created based on the dataset containing information about votes in United States House of Representatives (Schlimmer, 1987). For each model, I identified

[1] https://repo.bayesfusion.com/

variables of interest (target variables). Models Hepar II, HV, and Mortality90d include each only one target variable. The node associated with target variable variable in the structure of these models is a predecessor for each of the other nodes as the model follows augmented naïve Bayes (ANB) structure. Table 2 presents all models used in this simulation with some statistics and target variables identified for the purpose of the simulation.

For each model, I generated a set of 8,000 cases by means of probabilistic logic sampling (Henrion, 1988). For each case, I removed the information about variables at various rates $m \in \{0.0, 0.2, 0.4, 0.6\}$ to simulate missing values in the data. I also distorted observations randomly at various rates $w \in \{0.0, 0.1, 0.2, 0.4\}$ to simulate erroneous information in the data. As a result, I obtained 500 cases for each pair of m and w values.

For each case $\mathbf{E}$, the target variable (associated with the model) V_i and its value v_{ij}, I calculated the surprise index $\mathcal{SI}(\mathbf{E})$, the posterior probability $p = \Pr(V_i = v_{ij}|\mathbf{E})$, the length of the error bar over the posterior probability $\Pr(V_i = v_{ij}|\mathbf{E})$, and the length of the variation interval over the posterior probability $\Pr(V_i = v_{ij}|\mathbf{E})$.

7.2 OBSERVATIONS

Figures 9 through 13 show the results of the experiment. The first observation that I made based on the plot presenting the length of intervals against the posterior probability p. Both, variation intervals and error bars tend to get tighter as p approaches the values of zero or one. I present an example of this pattern in Figure 9. In some of the plots, I got a similar pattern, but truncated, which is a result of the fact that a posterior probability of an event may not get the value close enough to either of the ends of the interval $[0, 1]$, which is the feature of the modeled variable. Thus, I produced scatter plots of the length of variation interval and error bar against $d(p)$, where

$$d(p) = min(p, 1 - p) ,$$

which is the distance between the value $p = \Pr(V_i = v_{ij}|\mathbf{E})$ and the closer end of the interval $[0, 1]$. Figure 10 presents several examples in logarithmic scale. Reviewing these plots, we

Table 2: Models used in the simulations

Model	# nodes	# arcs	# targets	target variables
Alarm (Beinlich et al., 1989)	37	46	8	Anaphylaxis, Intubation, KinkedTube, Disconnect, Hypovolemia, LVFailure, InsuffAnesth, PulmEmbolus
BarleyFungalDisease (Kristensen and Rasmussen, 2002)	15	19	3	gt25, lt22, udbrsv
BarleyMain (Kristensen and Rasmussen, 2002)	48	84	6	bgbyg, tkv, ksort, spndx, udb, protein
BarleyWeed (Kristensen and Rasmussen, 2002)	15	24	2	weed, udbr
Hepar II (Oniśko et al., 2000)	70	123	9	THepatitis, ChHepatitis, PBC, fibrosis, Steatosis, Cirrhosis, Hyperbilirubinemia, RHepatitis, carcinoma
HV (Tree Augmented Naïve Bayes learnt from data (Schlimmer, 1987))	17	31	1	Party
Mortality90d (Kanwar et al., 2017)	27	51	1	DEAD_90d

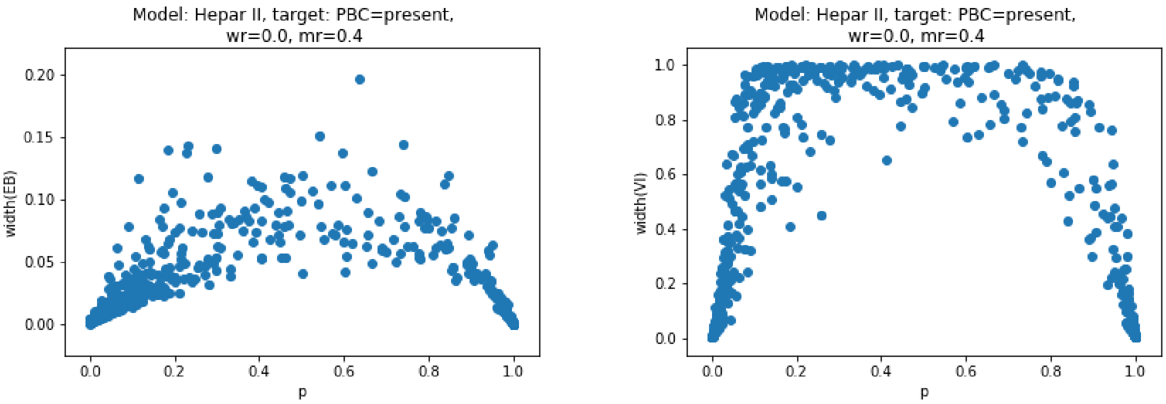


Figure 9: An example of two scatter plots showing how the interval length depends on the posterior probability $p = \Pr(PBC = present|\mathbf{E})$. The left plot shows the relationship between p and the length of the error bars over the value of p. The right plot shows the relationship between p and the length of the variation interval over the value of p

can observe an exponential relationship between these values $d(p)$ and the lengths of the intervals. The length of the error bar seems to depend exponentially on $d(p)$ as well. The first of these patterns seem to be stronger. For some target values in some models, I observed that $d(p)$ interacts in this way with both length of the variation interval and length of the error bar, there was also same relationship between two latter values as well (as exponential relationship is transitive).

I accumulated all of the cases for various possible missing information rates and possible data corruption rates. Figure 11 shows a portion of scatter plots of surprise index against length of error bar for various models. These plots suggest that for many variables, error bars get wide very often, when we deal with a case that has a low surprise index. Please note that it is not true for all of the modeled variables. For example, for the model BarleyFungalDisease, we can find cases that have quite high surprise index while the error bars over $\Pr(\mathit{gt25} = \mathrm{x0\_85\_\_}|\mathbf{E})$ are wide (the bottom right plot in the Figure 11).

Figure 12 shows a portion of scatter plots of surprise index against length of variation interval for various models. I could not find a common pattern besides the observation that there are many different cases with wide range of both values. It seems to be common that for cases for which we get extreme lengths of variation intervals we can have various values of surprise index. It is worth noting that for some of the variables there are values, for which posterior probability is within limited range tighter than $[0, 1]$. For example, the length of variation intervals over $p = \Pr(\mathit{weed} = \text{x100\_150}|\mathbf{E})$ in BarleyWeed model does not exceed 0.25, which means that the posterior probability p takes values in that range.

Figure 13 shows a portion of scatter plots of length of variation interval against length of error bar for various models. Some of the models suggest a similar pattern that I found analyzing plots of lengths of intervals against posterior probability p – error bars seem to be tighter as we get the length of the variation interval close to extreme values. The situation when both values are close to zero is exactly the situation when both of them are in relationship with p.

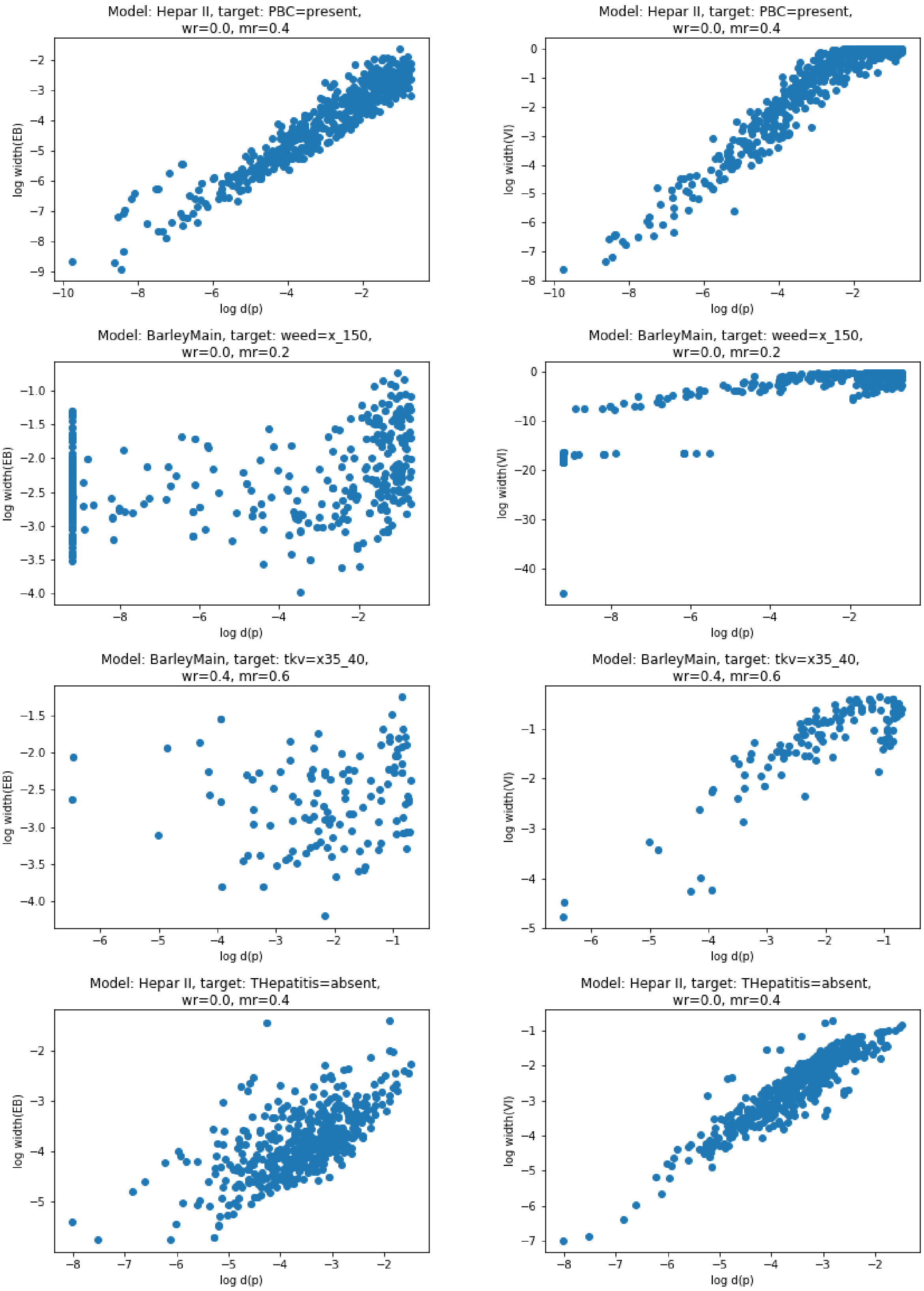
7.3 CONCLUSIONS

There seem to be an exponential relationship among lengths of variation intervals and error bars, and the posterior probability (its distance from the ends of $[0, 1]$ range). I could find plenty of examples, where lengths of variation intervals and error bars are dependent on the posterior probability value $p = \Pr(V_i = v_{ij}|\mathbf{E})$, although lengths of intervals are not perfectly explained by probability probability p, especially when the value of p is far from the ends of the interval $[0, 1]$.

For some models, wide error bars yield rarity of the case at hand. We can observe it as the error bars get wide only when the surprise index approaches zero for some models. Small surprise index, as mentioned earlier in Chapter 4, means that either we deal with a rare case or with inconsistent information describing the case. This phenomena is strongly visible for models of TAN structure.

All of the measures discussed in this chapter may enhance the output of a DSS in a useful way, also when all of them are provided to the user. Surprise index seems to be less dependent on the other measures. It suggests that providing surprise index alongside with the output of the model may improve the understanding of the situation that the user deals with, even having error bars or variation intervals provided already.

None of the relationships could be seen clearly for all of the models. It means that the creator of a DSS may consider all of the measures, but should test them all on the model before the deployment of the system. It may happen that the relationship between two of the measures may be strong for some models. In such situation these measures may be redundant.



Model: Hepar II, target: PBC=present, wr=0.0, mr=0.4

Model: Hepar II, target: PBC=present, wr=0.0, mr=0.4

Model: BarleyMain, target: weed=x_150, wr=0.0, mr=0.2

Model: BarleyMain, target: weed=x_150, wr=0.0, mr=0.2

Model: BarleyMain, target: tkv=x35_40, wr=0.4, mr=0.6

Model: BarleyMain, target: tkv=x35_40, wr=0.4, mr=0.6

Model: Hepar II, target: THepatitis=absent, wr=0.0, mr=0.4

Model: Hepar II, target: THepatitis=absent, wr=0.0, mr=0.4

Figure 10: Scatter plots showing how the variation interval (left column) and the error bar (right column) depends on the posterior probability $d(p)$ interact in log-log scale

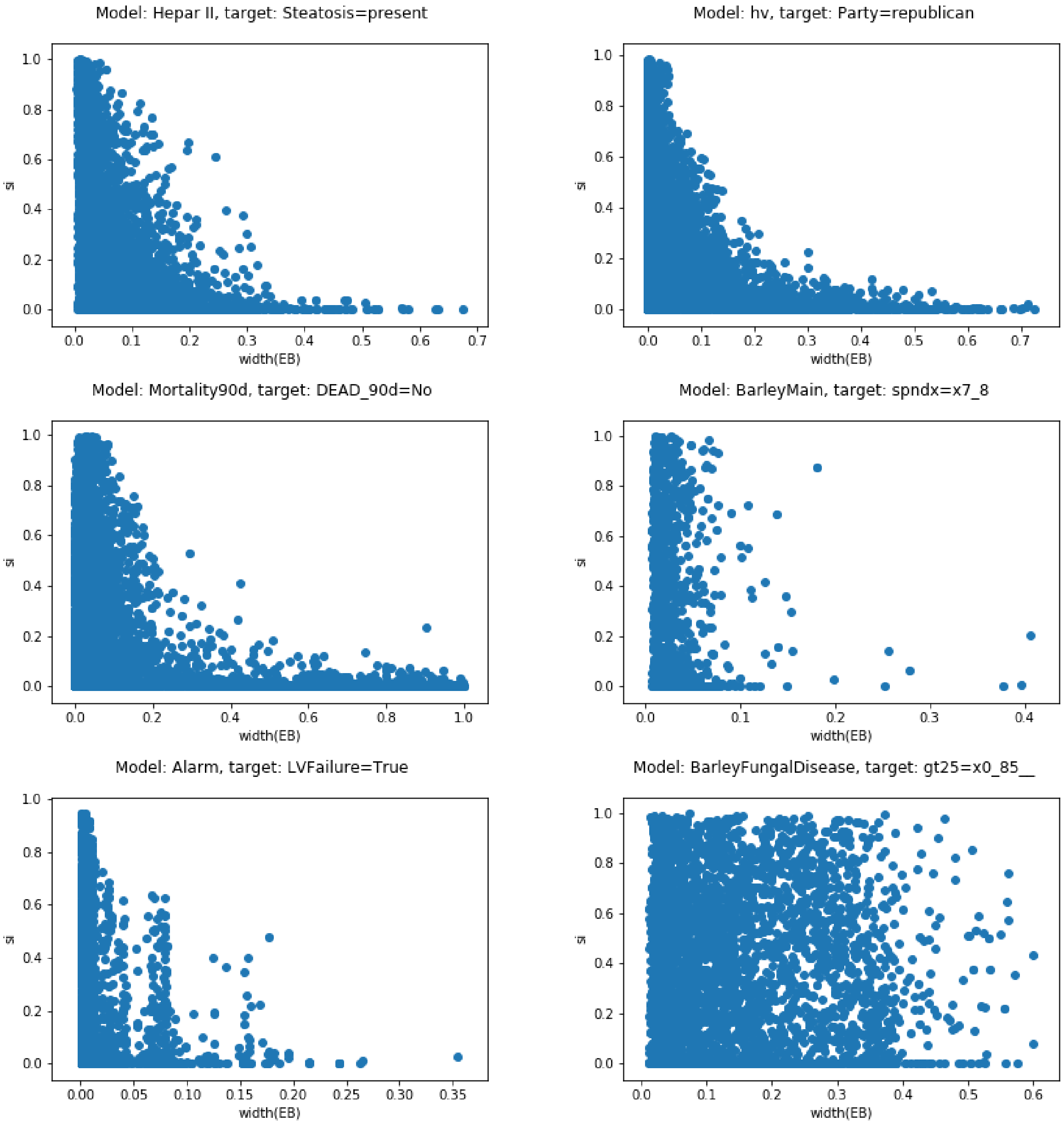


Figure 11: Scatter plots presenting the surprise index against the length of the error bar in various models and variables

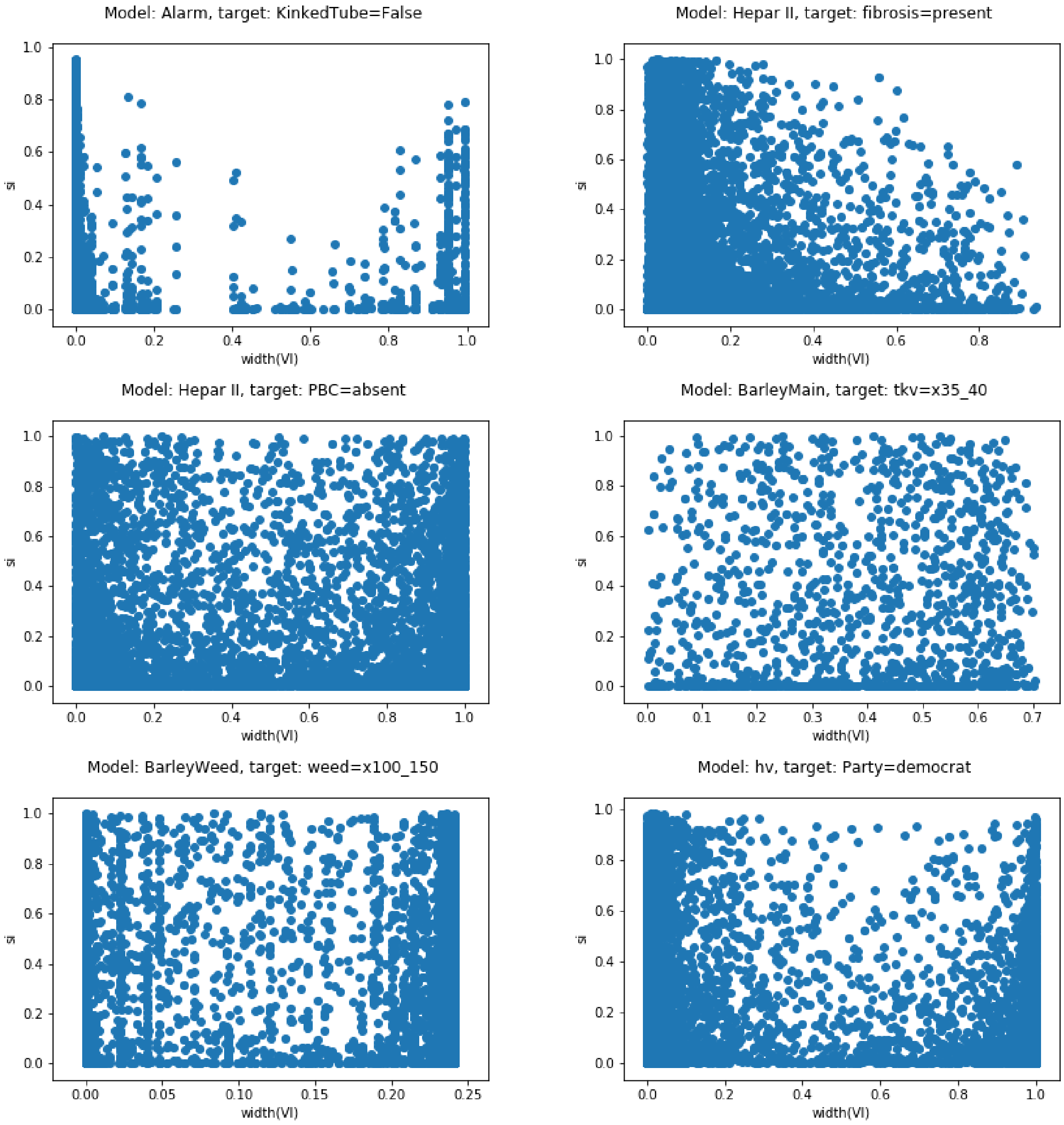


Figure 12: Scatter plots presenting the surprise index against the length of the variation interval in various models and variables

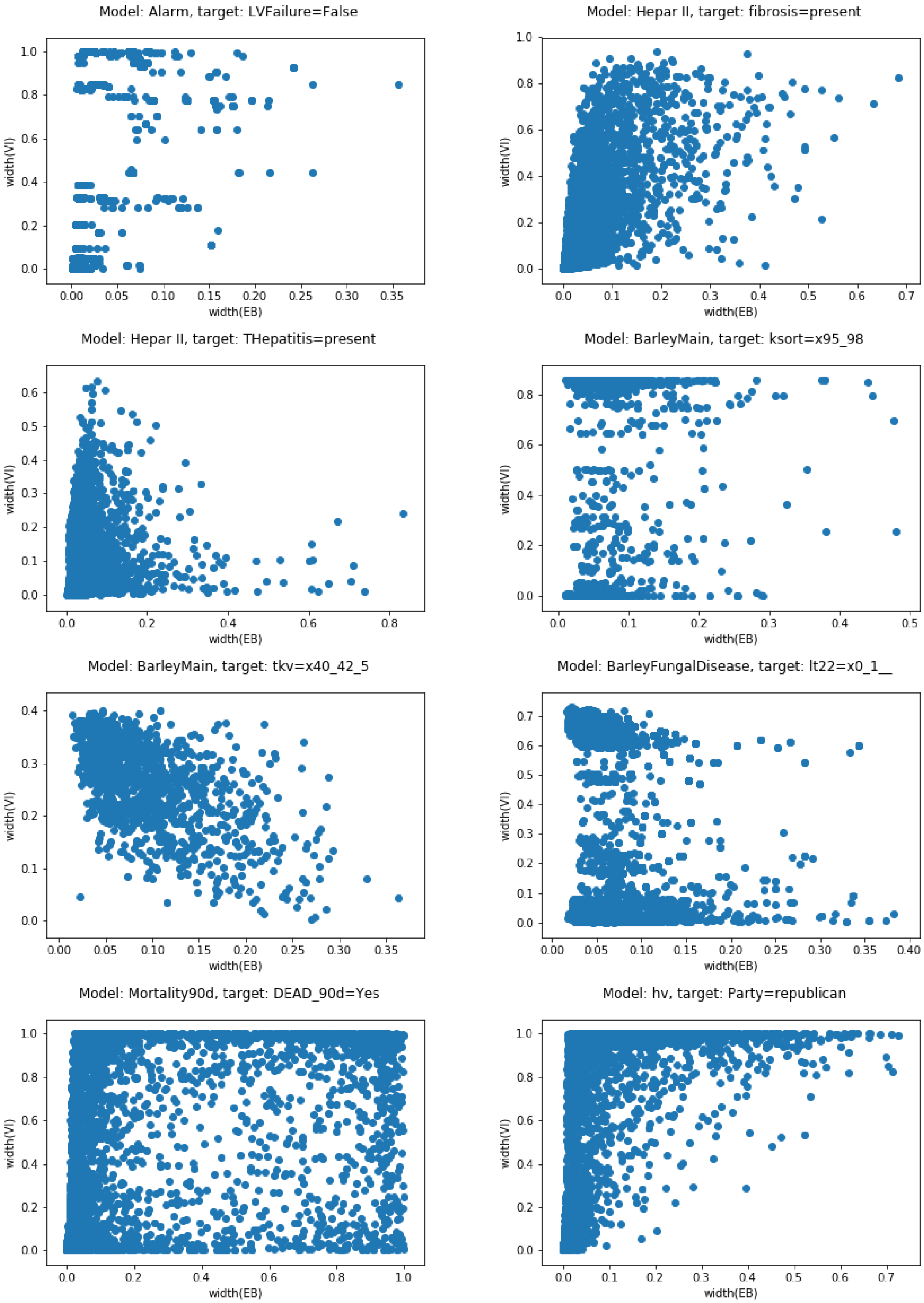


Figure 13: Scatter plots showing how the surprise index relates to size of the variation interval in various models and variables

8.0 PREDICTING PERFORMANCE OF A CLASSIFIER BY MEANS OF CONFIDENCE MEASURES

The purpose of using a confidence measure is to describe how much we can trust the output of a DSS. We trust the output of a system, when it gives correct answers in particular cases. The confidence measure should help to determine whether the output of the DSS is correct. To evaluate the measure, we can check whether it helps to predict an erroneous output of the DSS. In other words, we want to discriminate those cases (data points) that are problematic for the DSS by means of the confidence measure.

One form of a DSS is a classifier that assigns a label to the case provided in the input. For example, we can construct a classifier that is labeling patients with high risk of readmission to the hospital. In case of classification, problematic cases are those for which classifier gives an erroneous answer.

In this chapter, I check whether confidence measures presented in this dissertation help with predicting the correctness of the classification of a particular case with Bayesian network models.

This chapter is organized as follows. Section 8.1 describes the setup of the experiment, i.e., an overview of datasets and model types I used, followed by the explanation of the experiment. Section 8.2 shows obtained results. Section 8.3 presents conclusions from the performed experiment.

8.1 EXPERIMENT SETUP

8.1.1 Models used in the experiment

A BN model, in which the variable relating to the class is represented by one of its nodes, can be used as a classifier. During the process of classification, we calculate the posterior probability distribution over the class variable. Then we use a decision criterion applied to that distribution to choose a label. In my experiment, I choose the label that is the most probable given the case.

I used three types of classifiers based on Bayesian networks: naïve Bayes (NB), augmented naïve Bayes (ANB), and tree-augmented naïve Bayes (TAN). All models were created with the SMILE library.

Naïve Bayes is a simple model based on the assumption that all of the feature variables are independent of each other given the class variable. It means that in presence of information about value of the class variable, information about one feature variable does not have any influence on the distribution of other feature variables. As a result, a naïve Bayes network consists of the class variable, feature variables, and edges connecting the class variable with all the feature variables. A naïve Bayes classifier, besides being simple in calculation, tends to over-fit the data (that it was learned from) less than other models. It leads to better generalization of the model and performs quite well when applied to new cases.

We can use the augmented naïve Bayes model, when the assumption employed in the naïve Bayes is strongly violated. If the size of the dataset is sufficient to infer dependencies among the feature variables, especially in the presence of the information about value of the class variable, we can augment the naïve Bayes structure with additional edges. We can apply the Bayesian search algorithm (Cooper and Herskovits, 1992) to determine additional edges.

A BN with too many edges may over-fit the data. To prevent this from happening, it is possible to add restrictions on the number of edges added to naïve Bayes structure during the process of learning a model form data. We can augment the naïve Bayes structure with a spanning tree of nodes associated with feature variables. As a result we have a model where every node associated with feature variable has at most two parents.

To learn the models from the data, I used the default settings of the SMILE library (BayesFusion, LLC, 2019). The only exception was the augmented naïve Bayes model for *cover type* dataset, which was too complex to perform the experiments (it took too much time), when created with default settings. I put restrictions on the structure of the model. The algorithm was trying to fit the model, where each feature variable node had at most four (eight on default) direct predecessors (parents). I also lowered link probabilities from default 0.1 to 0.05 and prior link probability from 0.001 to 0.0005.

8.1.2 Datasets used in the experiments

In the experiments presented in this chapter, I used a collection of datasets selected from the UCI Machine Learning Repository (Dheeru and Karra Taniskidou, 2017). My choice was guided by following factors:

- the dataset had to be curated for classification task,
- the dataset should be available in a form of tabular data,
- the dataset had to consist of over ten attributes,
- and the dataset had to consist of vast number of records (at least 300).

Table 3 lists the datasets selected by means of above criteria along with their characteristics (discussed balow). I ran a simple preprocessing step for each of the dataset.

Because all of the measures described in this dissertation are meant for discrete BN models, I discretized all attributes that could be characterized as continuous with a simple rule into three values: *low*, *medium*, and *high*. I replaced all the values in the lower quartile by *low*, all the values in the upper quartile by *high*, and the remaining values in the interquartile range by *medium*. In these cases, where either the lower quartile or the upper quartile was dominated by one value, I had to adjust the cut points for the discretization manually. Sometimes it happened that the vast number of values were zero. In such cases, I discretized the attribute either into two values (e.g., *zero* and *positive*) or into three values (e.g., *zero*, *low*, and *high*).

I treated missing values in three ways: (1) I removed the records with missing values, if the number of such records for a given attribute was very small (less than eight); (2) if in the given column of the data (for a given attribute) there was a vast number of missing values (more than 40), I filled them with a special value *missing*; (3) in all other cases I filled the missing value with the most frequent value.

For two datasets, I modified the set of records. I created a sub-dataset of the *cover type* dataset by picking 5% of the records by random. The *adult merged* is a dataset consisting of both training and testing subsets available for *adult* dataset in the repository.

8.1.3 Procedure

For each dataset, I build a model and test it with the hold-out validation principle. I add a column to the testing dataset which consists of the result of the comparison of the predicted value for the class variable with the true value from the dataset. Then, I augment the testing dataset with columns consisting of calculated confidence measures for each row. Then, I consider confidence measures and the dataset with confidence measures and a column of values indicating whether the model assigned a correct class label to the record (case). I treat each confidence measure as a predictor of poor classifier performance. Evaluation of the measures is based on the area under the ROC curve for classifier performance prediction. Smith and Gal (2018) used a similar approach to evaluate measures of uncertainty for predicting adversarial attacks on systems based on neural networks. Figure 14 shows the basic procedure performed for each testing dataset.

In addition to standard settings of validation of classifiers, I wanted to check how well the measures will perform in the presence of missing and corrupted information. We deal with missing information when the values of some features are missing in the case. I simulated such phenomena by dropping information from testing dataset with predefined rate, i.e., each value was not entered into the model with probability 0.0 (no missing information), 0.2, and 0.5.

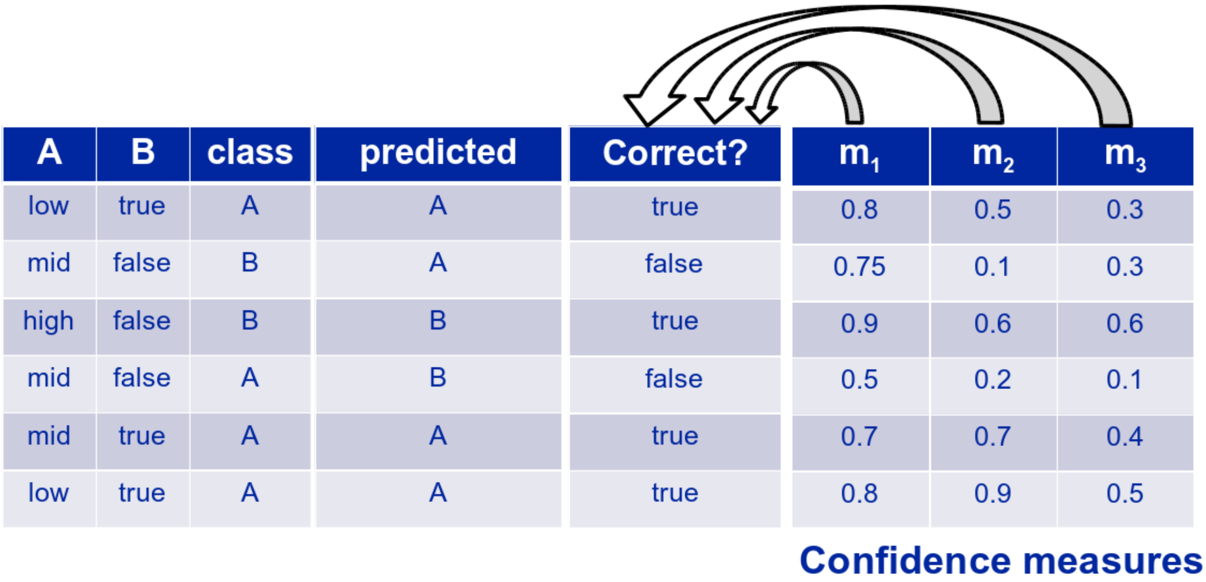
I simulated corrupted information by changing randomly selected values with predefined rate, i.e., each value was considered to be replaced with probability 0.0 (no corrupted in-

Table 3: Datasets used in the experiments

Dataset	# attr.	# rec.	Processing comments	Credit
thoracic surgery	17	470		(Zięba et al., 2014)
messidor (Diabetic Retinopathy)	19	1151		(Antal and Hajdu, 2014)
wine red	12	1599		(Cortez et al., 2009)
adult merged	14	48842	original training and testing datasets have been combined into one dataset	(Kohavi, 1996)
house votes	17	435		(Schlimmer, 1987)
cervical cancer	36	858		(Fernandes et al., 2017)
wine white	12	4898		(Cortez et al., 2009)
ionosphere	34	351		(Sigillito et al., 1989)
cover type	55	29050	20 times less instances compared to original dataset	(Blackard, 1998)
bands	34	533		(Evans and Fisher, 1994)
dermatology	35	366		(Güvenir et al., 1998)

formation), 0.2, and 0.4. When it was considered for change, I replaced it with one of the possible values of the feature by giving all of the values an equal chance. So it could happen that the value was not changed at all due to drawing the original value, e.g., for a binary variable, there was 0.5 probability that the value will not change. Thus the actual corrupted rate is lower than the predefined one.

I modified the hold-out validation procedure as described above. I divided the dataset



A	B	class	predicted	Correct?	m_1	m_2	m_3
low	true	A	A	true	0.8	0.5	0.3
mid	false	B	A	false	0.75	0.1	0.3
high	false	B	B	true	0.9	0.6	0.6
mid	false	A	B	false	0.5	0.2	0.1
mid	true	A	A	true	0.7	0.7	0.4
low	true	A	A	true	0.8	0.9	0.5

Figure 14: The goal of the experiment is to check whether the measures help predicting correctness of the classifier

into training and testing datasets at random in proportion 2:1.

I ran the whole modified hold-out validation procedure seven times for all combinations of model types, datasets, missing rates (0.0, 0.2, and 0.5), and corrupted data rates (0.0, 0.2, and 0.4). For each run, I calculated the area under the ROC curve corresponding to the ability of the confidence measure to discriminate mistakes in classification by the model.

8.2 RESULTS

I present the results in the form of box plots that accumulate the results over all datasets. Each box plot represents 77 runs (11 datasets, 7 runs per each dataset) on a particular model with a setup of missing values rate m and data corruption rate w. The green line represents a value 0.5 of AUC score. If the AUC is greater than 0.5, it means that the measure was helpful in predicting the correctness of classification. More detailed results for each dataset can be found in the Appendix.

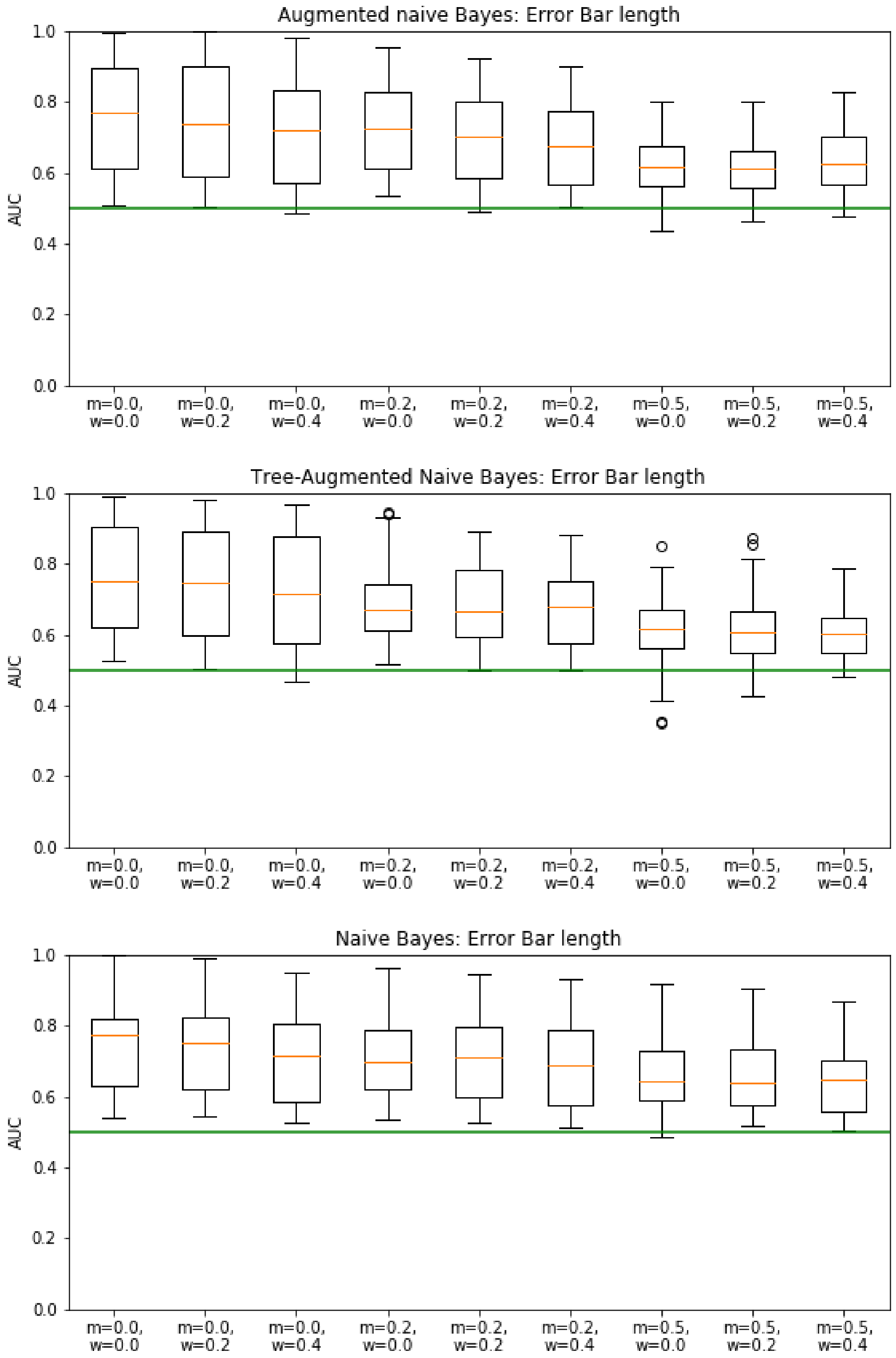


Figure 15: Area under the ROC for predicting the correctness of classification using length of the error bar

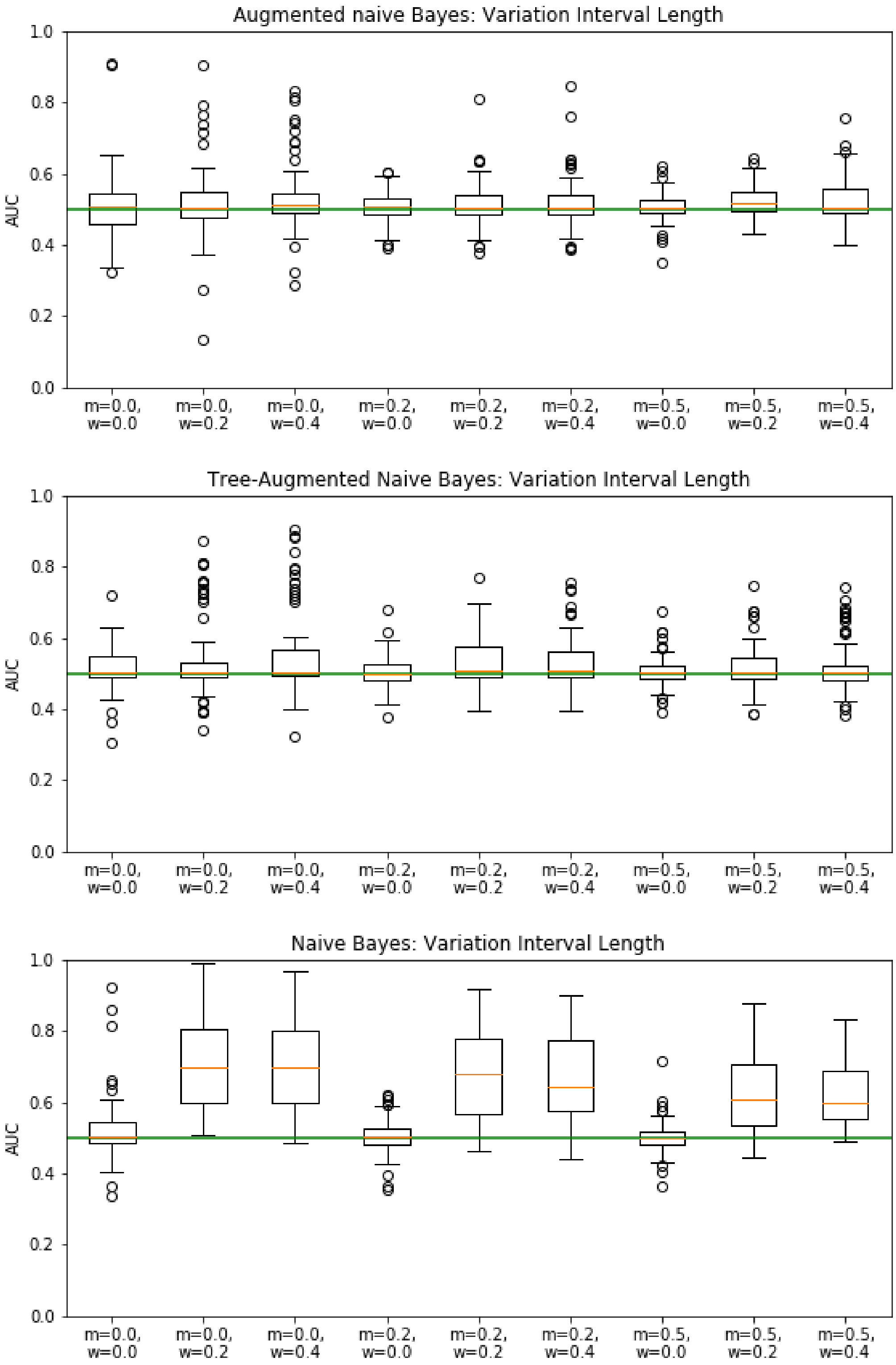


Figure 16: Area under the ROC for predicting the correctness of classification using length of the variation interval

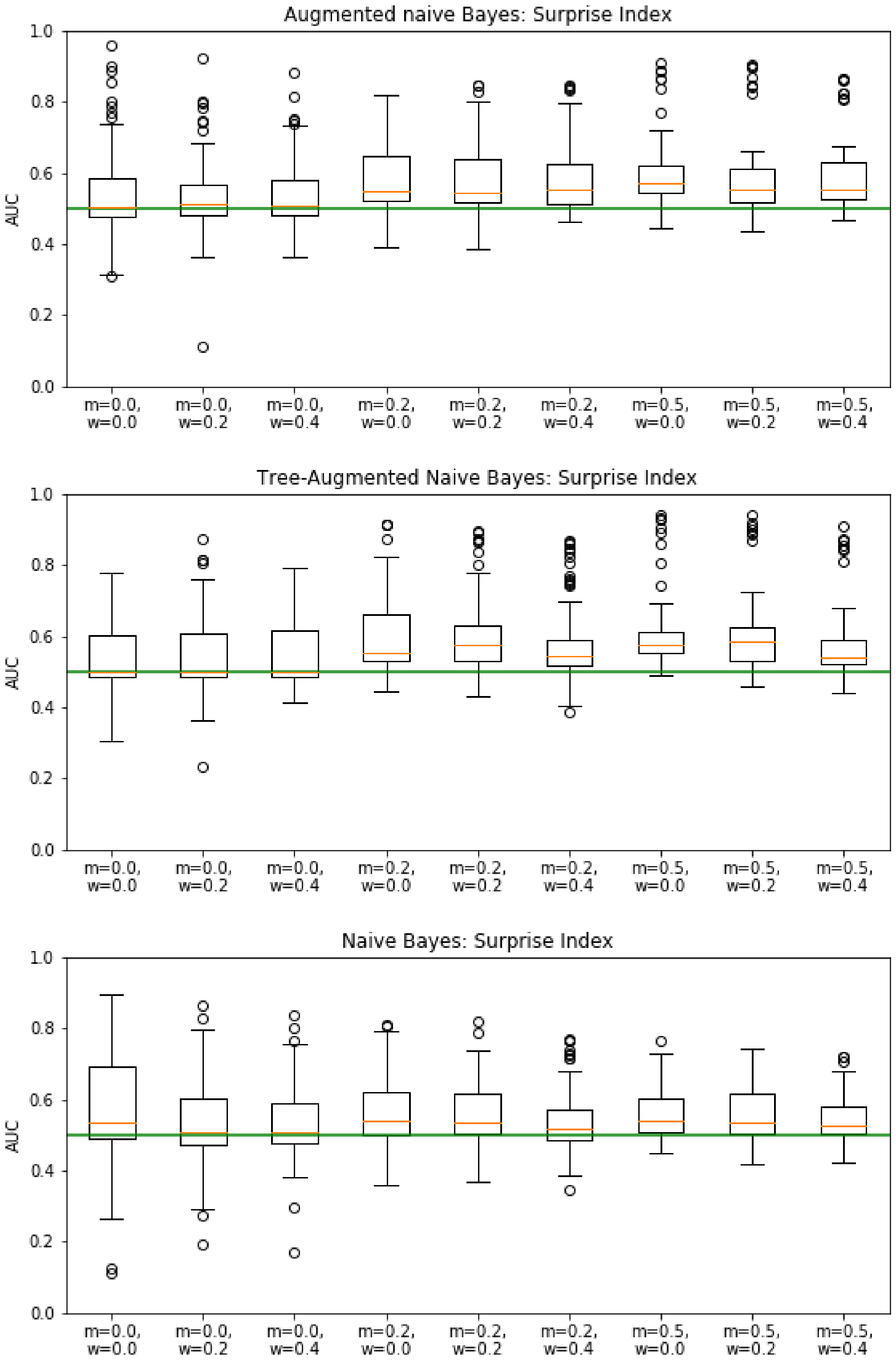


Figure 17: Area under the ROC for predicting the correctness classification using the surprise index

Figure 15 shows aggregated results in form of box plots for the complement of the error bar length ($CEBL$) applied to all three types of models. For the error bar length, we can see a very good performance for all types of the model with different configurations, as in most of the runs, AUC is greater than 0.5. Usually the AUC exceeded 0.55 (see tables in the Appendix). The performance drops slightly as we provide less information to the system about the case to classify (introduce more missing values). We can see slightly better performance of the complement of the error bar length when applied to augmented naïve Bayes in absence of missing information.

Figure 16 shows aggregated results in form of box plots for the complement of variation interval length ($CVIL$) organized in the same fashion. We can observe rather poor performance compared to the results obtained by applying the complement of the error bar length measure. $CVIL$ performs quite well for several datasets when all the information is provided in the input and the data are corrupted to some degree. A very good performance can be observed when the complement of variation interval length is applied to naïve Bayes in the presence of corrupted information.

Figure 17 shows aggregated results in form of box plots for the surprise index ($\mathcal{SI}$) organized in the same way. In general, surprise index seems to perform better than $CVIL$, but worse than $CEBL$. For some datasets surprise index performed very well for all different types of settings. The more missing values were present in the cases, the better surprise index was performing for particular datasets. Unfortunately, for some datasets, surprise index was introducing misleading information in predicting erroneous output.

I compared the resulting AUC values for these measures with Wicoxon signed-rank test. $CEBL$ is better than surprise index in predicting the erroneous classification with p value less than 10^{-5} (for most of the datasets and among all of the models types). In my experiment, surprise index turns out to perform better than $CVIL$ for most of the datasets with tree-augmented naïve Bayes (TAN) and augmented naïve Bayes classifiers (ANB) with p value less than 10^{-5}. For naïve Bayes, surprise index was significantly better just for three datasets.

8.3 DISCUSSION AND CONCLUSIONS

Complement of the error bar length helps the most in predicting erroneous output of a classifier. One of the reasons is that it is closely related to the posterior probability distribution (as has been shown in Chapter 7).

Surprise index gave good performance, especially in the presence of missing values. It is worth mentioning that surprise index does not depend on the posterior probability distribution of the target as much as other measures, like complement of error bars length.

The complement of variation interval length seemed to perform the worst. The best result I obtained for naïve Bayes in presence of wrong information. It means that $CVIL$ is good at detecting inconsistent information when applied to naïve Bayes models. It has to be noted that $CVIL$ performed very well for some of the datasets when applied to models of more complex structure than NB.

Furthermore AUC score obtained for $CVIL$ seems to get slightly worse as we fit the model more to the data (ANB against NB). Although, as presented earlier in this document, variation intervals tend to get tight quite quickly in the presence of most significant information about the case. In the experiment, the information provided to the classifier was either full or containing missing information at random. When a DSS is applied in a way where the user is entering information about the case piece by piece, starting from the most significant one, the measure based on variation intervals could be more useful in predicting performance of the system.

Both measures, complement of the error bar length and surprise index, seem to perform well in various configurations and can be used in applications.

If any of these measures could be applied to the system using a Bayesian network model, the modeler needs to determine the proper threshold for the measure to indicate possible erroneous output.

8.4 FURTHER WORK

As I showed in Chapter 7, the size of the error bar over posterior probability of a label is dependant on the value in question. We can check how much information the error bar introduces over the posterior probability based on already done experiments and obtained results.

Significance of the feature value in the information about the case can be measured with cross entropy. It could be further investigated whether for a few pieces of information about the case chosen based on the cross entropy, the variation interval length helps to predict erroneous output of the classifier.

9.0 APPLYING CONFIDENCE MEASURES TO ENSEMBLES OF CLASSIFIERS

It may happen that institutions that own data, do not want to share them due to various reasons. For example, hospitals may not share data due to privacy protection: they need to protect the information about the patients. As the data cannot be shared, each of the institutions may prepare their own version of the model and share it. Other parties may use these models separately, combine them into one model, or try to use as an ensemble.

An intuitive way of merging these models would be a combination of the available models into one model. One of problems is that different modelers may choose a different subset of the variables to build a model for a similar purpose. In such a situation, they may have troubles with generalizing their models into one model applicable by all institutions. Another problem is that there are many ways of handling continuous variables. For example, each modeler may choose different discretization of continuous variables.

In case of Bayesian network models, even if the modelers agree to one standard of handling the data and the variables used in the model, there is still a problem of determining the structure of the network that would utilize all of the knowledge encoded in the models. Despite some attempts to develop a method for combining Bayesian networks (e.g., Feng et al., 2014), the problem remains difficult.

As merging of models may be problematic, an alternative approach is to use all the created models as an ensemble. A possible approach is to take an average of outputs of these models. We can weight the particular outputs with overall accuracy of particular models. If the domain (e.g., a subset of variables) modeled by BNs differs, we may want to use surprise index (or another confidence measure) to weight each of the model's output for a given case.

Ensembles of classifiers and regression functions are well studied (Rokach, 2010). By averaging multiple regression functions, we decrease the variance of the output and increase the bias (Geman et al., 1992), which contribute to the error of prediction. One method to find an optimal bias and variance resulting in high accuracy is Bagging (Breiman, 1996). This method uses an ensemble of classifiers learned from bootstrapped samples of the training dataset. In the problem described above we do not have one dataset defined in a feature space, but a set of various datasets that may cover different parts of the same feature space.

Usually, ensembles of classifiers, that employ simple aggregating methods, get the best results when individual models in the ensemble are diverse. If the models are strongly diverse and good in classification in their domains, we expect to get better performance when we favor proper classifiers that are good in particular cases. If we can determine which classifier is the best for the case at hand we can give it more credit by assigning higher weight in the ensemble. Many methods have been developed for the aggregation of the ensemble of classifiers (Cruz et al., 2018) that could be categorized as trainable, non-trainable, and dynamic weighting. Usually, these methods rely on the outputs of all classifiers in the ensemble and the level of competence induced from the accuracy of the classifier in the region close to the case at hand (e.g., Woloszynski and Kurzynski, 2011). Some studies utilize credal intervals (which may refer to similar imprecision of the output as error bars) to assess the robustness of graphical probabilistic models in an ensemble (e.g., Conaty et al., 2018). I have not found any paper that would explore using surprise index for assessing the competence of models in ensembles. The confidence measures I present in this paper could be applied as measures of competence. For each case and model in the ensemble, I calculate the confidence measures and use their combination as weight in averaging of the outputs of the models.

Another additional stage in the creation of an ensemble of classifiers is a selection of a subset of the models. Such selection could be done either in a static (Perrone and Cooper, 1992) or in a dynamic way, dependent on the case at hand. The dynamic model selection could be made based on the methods for describing the competence of the model as discussed above. I omit this stage and assess the usefulness of confidence measures in weighting classifiers in the ensemble.

In this chapter, I show the results of the experiment where I simulate the situation of various modelers building their own model for the purpose of classification. I simulate the separate domains by taking different subsets of the original learning dataset, which is a method used in an ensemble of classifiers to create diverse classifiers. I check whether employing confidence measures in ensembles of classifiers based on Bayesian networks improves classification accuracy. Section 9.1 outlines the setup of the experiment. Section 9.2 presents the results of the experiment. Section 9.3 concludes this chapter with final remarks on improving the classification accuracy in ensembles of classifiers by means of confidence measures.

9.1 EXPERIMENT SETUP

This experiment had three phases: learning the models, applying the models to the testing dataset, and evaluation of different methods to combine the posterior probability distributions of the models. I used the same datasets as in the experiment presented in Chapter 8.

In the first phase, I divided each dataset into two datasets: learning dataset (consisting of 66% of all datapoints) and testing dataset. For each learning dataset, I created 13 datasets which where constructed by removing randomly $(1 - s_c)$ columns and dropping randomly $(1-s_r)$ rows, for fixed s_c and s_r respectively. Sometimes I had to remove additional columns that becomes constant after dropping some of the datapoints. For each of these smaller datasets, I learned a model using the SMILE library.

I applied all of the models to cases represented by rows in the corresponding testing dataset. For each case, I used $1 - m_r$ of values of features to simulate missing information. For each case, I calculated the surprise index, posterior probability distribution of the class variable, error bars, and variation intervals.

All the tests I ran were based on two types of models: naïve Bayes (NB) and tree-augmented naïve Bayes (TAN). I ran the tests with different values of s_c, s_r, and m_r, which was $s_c \in \{0.5, 0.6\}$, $s_r \in \{0.7, 0.8\}$, and $m_r \in \{0.0, 0.5\}$.

At the end I applied different approaches in combining outputs of models. A baseline method I used, was an arithmetic mean of the probabilities for respective value of the class variable across the posterior probability distributions of the models

$$p_{avg}(i) = \frac{1}{n}\sum_{j=1}^{n} \Pr(C = c_i|\mathbf{E}, \mathcal{B}_j), \tag{9.1}$$

where $\mathcal{B}_j = (\mathcal{G}_j, \Theta_j)$ is the j-th model. I compared it with weighted mean

$$p_{w_k}(i) = \sum_{j=1}^{n} w_k(i,j) \Pr(C = c_i|\mathbf{E}, \mathcal{B}_j), \tag{9.2}$$

where w_k is a weight based on surprise index ($\mathcal{SI}$), complement of error bar length ($CEBL$), and complement of variation interval length ($CVIL$). I considered seven different weights

$$\begin{aligned}
w_1(i,j) =& \mathcal{SI}(\mathbf{E}, \mathcal{B}_j), \\
w_2(i,j) =& \qquad\qquad CEBL(C = c_i|\mathbf{E}, \mathcal{B}_j), \\
w_3(i,j) =& \mathcal{SI}(\mathbf{E}, \mathcal{B}_j) \cdot CEBL(C = c_i|\mathbf{E}, \mathcal{B}_j), \\
w_4(i,j) =& \qquad\qquad CVIL(C = c_i|\mathbf{E}, \mathcal{B}_j), \\
w_5(i,j) =& \mathcal{SI}(\mathbf{E}, \mathcal{B}_j) \cdot CVIL(C = c_i|\mathbf{E}, \mathcal{B}_j), \\
w_6(i,j) =& \qquad\qquad CEBL(C = c_i|\mathbf{E}, \mathcal{B}_j) \cdot CVIL(C = c_i|\mathbf{E}, \mathcal{B}_j), \\
w_7(i,j) =& \mathcal{SI}(\mathbf{E}, \mathcal{B}_j) \cdot CEBL(C = c_i|\mathbf{E}, \mathcal{B}_j) \cdot CVIL(C = c_i|\mathbf{E}, \mathcal{B}_j).
\end{aligned}$$

As I had a set of 13 models, I considered 11 consecutive subsets of the models ranging from 3 models to 13 models and compared classification accuracy obtained with p_{w_k} to simple p_{avg}.

9.2 RESULTS

It has to be mentioned, that usually when the ensembles of classifiers are created, the modeler tries with different numbers of classifiers with different settings. The modeler stops building the models when she finds a settings that gives optimal or satisfying result. Nevertheless, in my experiments I tried the methods of combining model's outputs on different settings blindly.

This is why the main point of reference that I used in comparing the approaches of combining outputs of the models was the fraction of ensembles that got better results by using the weighting approach under consideration to simple averaging of the posterior probability distributions with p_{avg}.

Table 4: Aggregated results for ensembles of naïve Bayes (NB) and tree-augmented naïve Bayes (TAN) with and without missing values in the input. Each value in the table represents the fraction of 484 ensembles that got better accuracy in classification with respective p_{w_k} compared to baseline p_{avg}

p_{w_k}	measures employed	NB	NB (missing)	TAN	TAN (missing)
p_{w_1}	$\mathcal{SI}$	0.194	0.364	0.176	0.550
p_{w_2}	$CEBL$	0.227	0.335	0.202	0.405
p_{w_3}	$\mathcal{SI}, CEBL$	0.215	0.397	0.192	0.510
p_{w_4}	$CVIL$	0.000	0.093	0.000	0.010
p_{w_5}	$\mathcal{SI}, CVIL$	0.194	0.368	0.176	0.550
p_{w_6}	$CEBL, CVIL$	0.227	0.351	0.202	0.405
p_{w_7}	$\mathcal{SI}, CEBL, CVIL$	0.215	0.401	0.192	0.510

Table 4 presents aggregated results for ensembles of naïve Bayes (NB) and tree-augmented naïve Bayes (TAN) with and without missing data. We can observe that it is most likely to obtain an improvement in accuracy utilizing confidence measures when we deal with a case with missing information in the input of the system (50% missing values). Additionally, when dealing with missing information it is more likely to get improvement on ensembles based on tree-augmented naïve Bayes. When there is no missing information in the output, it is more likely to get improvement in ensembles based on naïve Bayes.

Utilizing complement of the variation interval length ($CVIL$) does not introduce any improvement when there is no missing information in the output. It is due to the nature of the variation intervals. They shrink to a point as we have all of the information that the posterior probability distribution of the class variable depends on. $CVIL$ makes it slightly more likely to improve accuracy when building ensembles of naïve Bayes classifiers.

Table 5: Results for ensembles of naïve Bayes (NB) and tree-augmented naïve Bayes (TAN) without missing values in the input. Each value in the table represents the fraction of 44 ensembles that got better accuracy in classification with respective p_{w_k} compared to baseline p_{avg}

models	NB			TAN		
dataset	$\mathcal{SI}$	$CEBL$	$\mathcal{SI}, CEBL$	$\mathcal{SI}$	$CEBL$	$\mathcal{SI}, CEBL$
adult merged	0.0227	0.0227	0.0000	**0.3409**	0.0000	0.0000
bands	0.2500	**0.3182**	0.3182	0.2727	0.2955	**0.3864**
cover type	0.0000	**0.4091**	0.0227	**0.1364**	0.0682	0.0000
dermatology	0.0227	0.1591	**0.1818**	0.0227	**0.2045**	**0.2045**
house votes	**0.1364**	0.0909	**0.1364**	0.1136	**0.2500**	0.1818
ionosphere	**0.7727**	0.0682	0.6818	0.0455	**0.1136**	**0.1136**
messidor	0.0682	0.0455	0.0682	0.2955	0.3182	**0.3409**
cervical cancer	0.4091	0.2955	**0.4318**	0.0000	0.0909	**0.1364**
thoracic surgery	0.0909	0.0909	**0.1818**	0.0227	**0.1136**	0.0909
wine red	0.1364	**0.4318**	0.1364	0.3864	0.3864	0.3864
wine white	0.2273	**0.5682**	0.2045	0.2955	**0.3864**	0.2727

Table 5 presents more detailed results with respect to datasets used in the experiments. These results correspond to ensembles of NB and TAN without missing values in the input. I dropped the cases in which I applied $CVIL$ as it did not improve the classification accuracy (no missing information means that $CVIL = 1$). For only three datasets for ensembles of NB models, the integration of the measures increased the fraction of improved classification accuracy. We can observe the same for ensembles of TAN models. For ensembles of TAN models, $CEBL$ improved the accuracy more often then $\mathcal{SI}$ individually. For ensembles of

NB models, each measure worked better for different datasets.

Figure 18 presents histograms of improvement of classification (difference between accuracies) of combining model outputs with weighting and simple averaging, corresponding to ensembles of NB and TAN without missing values in the input. We can see that usually the improvement is close to zero. By employing surprise index we can get as much as 0.04 of improvement. For ensembles of TAN models, usually, the improvement does not exceed 0.01.

Table 6 shows more detailed results with respect to datasets used in the experiments that correspond to ensembles of NB models with missing values in the input. We can see that for different datasets different set of confidence measures used was most likely to lead to an improvement. Combining the model outputs utilizing surprise index improved accuracy frequently. For four datasets, it was among best methods (we can say the same about the method employing just $CEBL$) in terms of frequency of improving classification. The method using the product of all three measures was among the best for five datasets. In several cases, utilizing $CVIL$ increased frequency of improving classification.

Table 7 shows more detailed results with respect to datasets used in the experiments that correspond to ensembles of TAN models with missing values in the input. The method utilizing surprise index seems to be among the best for most datasets.

Figure 18 presents histograms of improvement of classification of combining models outputs with weighting and simple averaging, corresponding to ensembles of NB and TAN with missing values in the input. We can see that usually the improvement is close to zero as well. For ensembles of TAN models, the histograms look skewed. They also reveal that huge improvements are possible (e.g., cervical cancer dataset). By employing both $CEBL, CVIL$ (p_{w_6}) for ensembles of TAN models, we could get an improvement as high as 0.14. For ensembles of TAN models, usually, the improvement did not exceed 0.03.

9.3 CONCLUSIONS AND REMARKS

The presented experiment attempted to utilize the confidence measures in just one particular way – by constructing weights based on product of the measures. It seem to be possible to improve the accuracy of the classification with ensemble of the models based on Bayesian networks by adding weights to outputs of different models based on confidence measures presented in this dissertation, although it is not the only way to utilize them in combining model outputs. It is very easy to construct different formulas for aggregating model outputs, which leaves plenty of room for future work.

The most advantage from confidence measures, both in frequency of improvement and its quality, I could get when I was using them to combine models in ensemble of TAN in presence of missing information. It may be due to the fact that TAN models seem to fit the data better than NB and that partial information about the case at hand gives is propagated in the structure of a TAN model.

In the results of the experiment presented in this chapter, I could observe high accuracy of classification by taking simple average of the posterior probability distributions of the models in the ensemble. Based on the results obtained in the experiment, this method gives a proper stability for accuracy of the classification. The accuracy was usually close to or exceeding the accuracy of the best single model in the ensemble.

Utilization of variation intervals in presence of all information about a case does not make any sense. But it does not contribute that much compared to other measures when we deal with missing information either. It still seems to be useful in combining of models in ensemble. Poor performance of variation intervals may be due to the same fact as it was not useful in predicting the performance of the classifier, i.e., the information missing at random does not give a proper room for variation intervals to show their full potential as information is not missing at random in real problems.

In future work confidence measures may be considered in dynamic model selection in the ensemble of classifiers based on Bayesian networks.

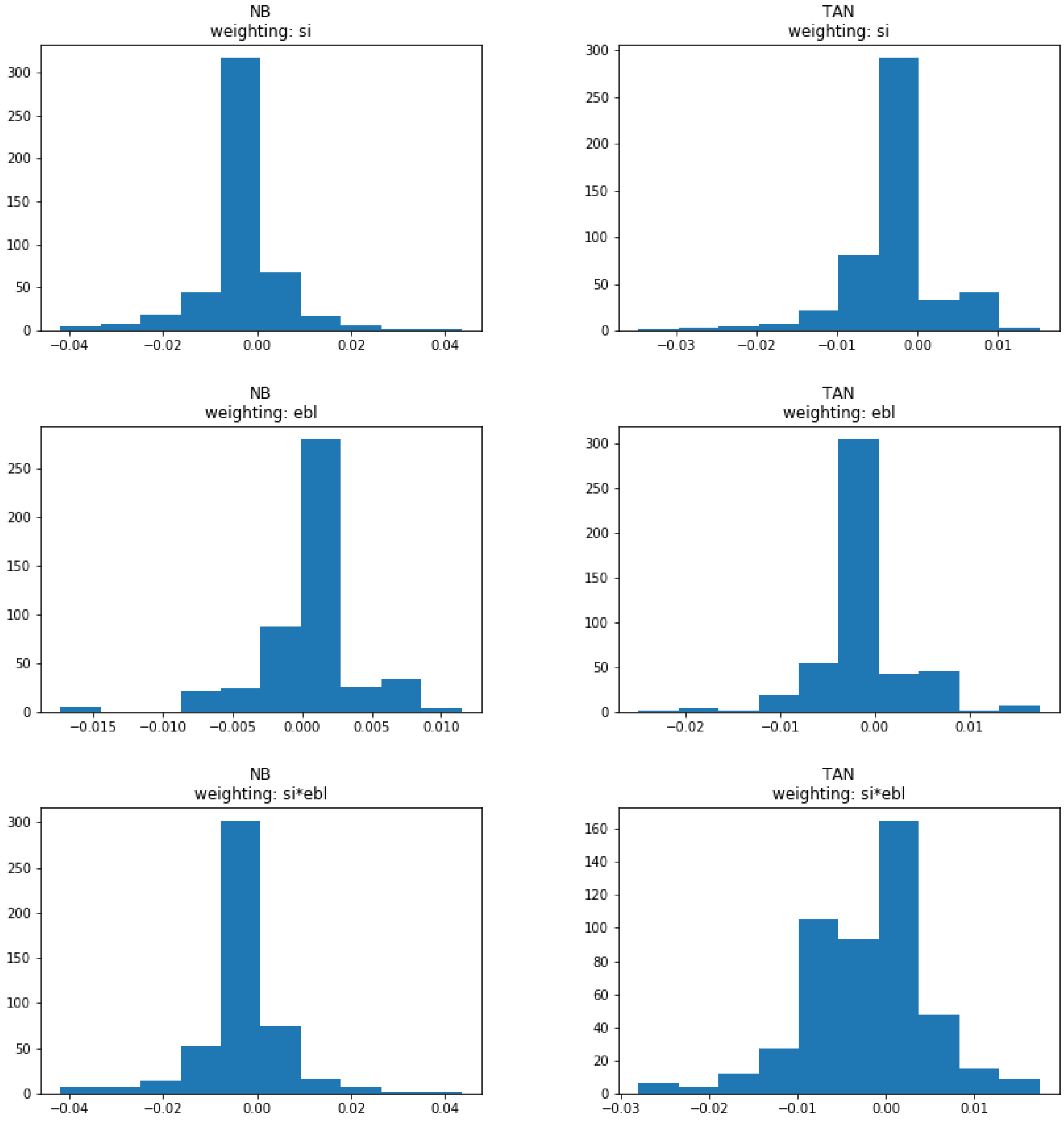


Figure 18: Histograms of improvements for ensembles of naïve Bayes (NB) and tree-augmented naïve Bayes (TAN) without missing values in the input. Each value contributing to histogram represents the improvement in accuracy of classification with respective p_{w_k} compared to baseline p_{avg}. Please, note that the median is placed close to zero, so we get quite good improvements when we observe a skewed histogram with tail on the right

Table 6: Results for ensembles of naïve Bayes (NB) with missing values in the input. Each value in the table represents the fraction of 44 ensembles that got better accuracy in classification with respective p_{w_k} compared to baseline p_{avg}. The maximum value for each dataset is presented in bold

	p_{w_1}	p_{w_2}	p_{w_3}	p_{w_4}	p_{w_6}	p_{w_5}	p_{w_7}
dataset	si	ebl	si,ebl	vil	ebl,vil	si,vil	si,ebl,vil
adult merged	0.2955	**0.5227**	0.3636	0.0909	0.5000	0.2955	0.3636
bands	0.5455	0.3409	0.6364	0.3636	0.5227	0.6591	**0.7045**
cover type	**0.8864**	0.6364	0.5682	0.0909	0.6364	**0.8864**	0.5682
dermatology	0.0455	0.4773	**0.5227**	0.2045	0.5000	0.0227	**0.5227**
house votes	**0.6136**	0.1364	0.5682	0.0000	0.1364	0.5909	0.5682
ionosphere	0.5682	0.2955	0.5909	0.0227	0.2955	0.5682	**0.6136**
messidor	**0.4091**	**0.4091**	0.3636	0.0000	0.3864	**0.4091**	0.3636
cervical cancer	0.3409	0.1136	**0.3864**	0.0000	0.1136	0.3409	**0.3864**
thoracic surgery	**0.0227**	**0.0227**	**0.0227**	0.0000	**0.0227**	**0.0227**	**0.0227**
wine red	0.0000	0.3864	0.0455	0.2500	**0.4091**	0.0000	0.0000
wine white	0.2727	**0.3409**	0.2955	0.0000	**0.3409**	0.2500	0.2955

Table 7: Results for ensembles of tree-augmented naïve Bayes (TAN) with missing values in the input. Each value in the table represents the fraction of 44 ensembles that got better accuracy in classification with respective p_{w_k} compared to baseline p_{avg}. The maximum value for each dataset is presented in bold

	p_{w_1}	p_{w_2}	p_{w_3}	p_{w_4}	p_{w_6}	p_{w_5}	p_{w_7}
dataset	si	ebl	si,ebl	vil	ebl,vil	si,vil	si,ebl,vil
adult merged	0.2500	**0.3182**	0.1364	0.0000	**0.3182**	0.2500	0.1364
bands	**0.5682**	0.3864	**0.5682**	0.0000	0.3864	**0.5682**	**0.5682**
cover type	**0.7727**	0.3636	0.2955	0.0000	0.3636	**0.7727**	0.2955
dermatology	0.3636	0.5227	**0.6136**	0.0000	0.5227	0.3636	**0.6136**
house votes	0.6364	0.4091	**0.7273**	0.0000	0.4091	0.6364	**0.7273**
ionosphere	0.8636	0.3636	**0.9545**	0.0000	0.3636	0.8636	**0.9545**
messidor	**0.4545**	0.2727	0.4318	0.0000	0.2727	**0.4545**	0.4318
cervical cancer	**1.0000**	0.9545	**1.0000**	0.1136	0.9545	**1.0000**	**1.0000**
thoracic surgery	**0.0909**	**0.0909**	0.0682	0.0000	**0.0909**	**0.0909**	0.0682
wine red	**0.4091**	0.2500	0.2727	0.0000	0.2500	**0.4091**	0.2727
wine white	**0.6364**	0.5227	0.5455	0.0000	0.5227	**0.6364**	0.5455

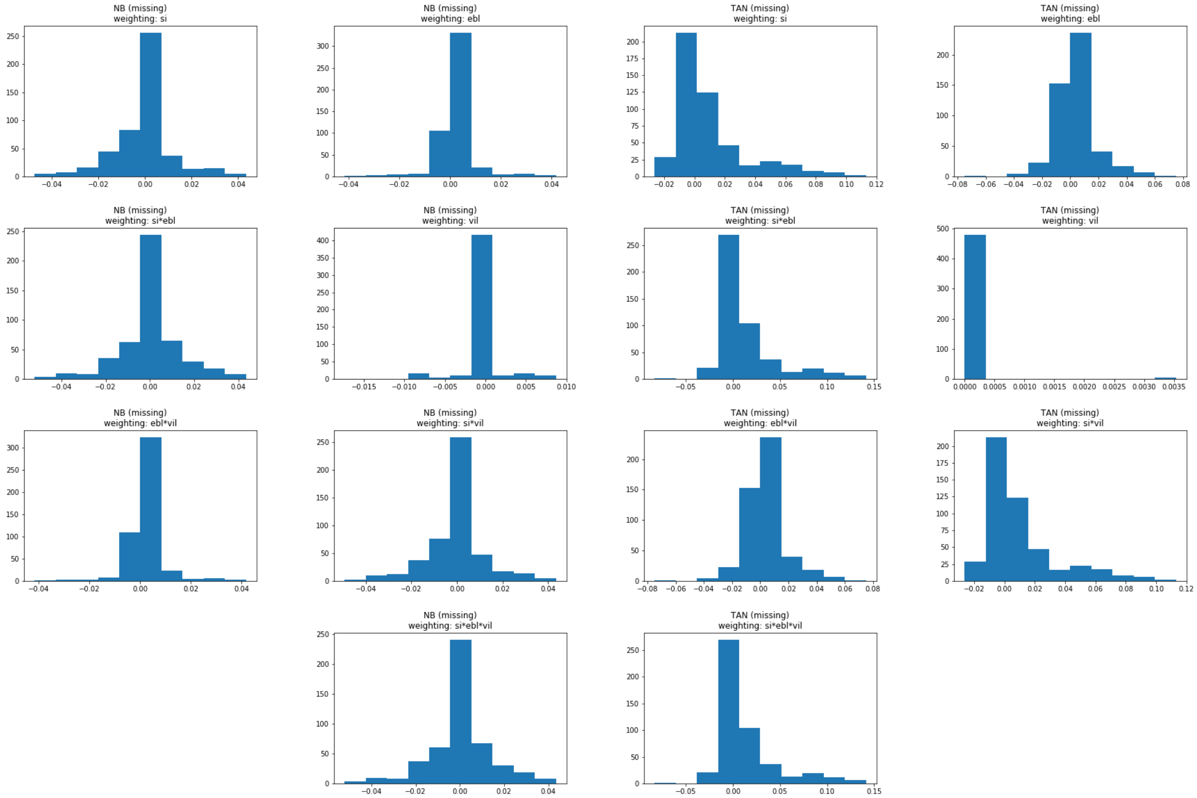


Figure 19: Histograms of improvements for ensembles of naïve Bayes (NB) (left) and tree-augmented naïve Bayes (TAN) (right) without missing values in the input. Each value contributing to histogram represents the improvement in accuracy of classification with respective p_{w_k} compared to baseline p_{avg}. Please, note that the median is placed close to zero, so we get quite good improvements when we observe a skewed histogram with tail on the right

10.0 CONCLUSIONS AND POSSIBLE DIRECTIONS FOR FURTHER WORK

10.1 CONCLUSIONS

Many practitioners applying DSSs ask how much they may trust the output of the system that they are using. This is why the quantification of the competence of the DSS is crucial.

In this dissertation, I have reviewed and evaluated several measures of confidence of a DSS based on Bayesian networks. I focused on three measures of confidence in the system output that are specific to the use case of a Bayesian network model: (1) surprise index, (2) the length of the error bar, and (3) the length of the variation interval. I investigated their performance in indicating the erroneous output of DSS. Each of these three measures proved to be useful and may be considered in applications of Bayesian network models.

Most widely known among these three measures of confidence in probabilistic models used in practice is error bar. It captures the uncertainty about the posterior probability of an event of our concern due to the imprecision of parameters in the model.

The surprise index as a confidence measure is considered to be intractable in the calculation. I proposed to use an approximation of it which is tractable. Surprise index represents well the rarity and conflicting evidence of a specific case in applications of Bayesian network model.

I proposed another measure based on variation intervals, which I have not found anywhere else in the literature. In the presence of missing information, variation intervals capture how the posterior probability of an event may change as we provide more information about the case to the system.

I performed a series of experiments to demonstrate how these three measures work and how they relate to each other. The first observation that I made is that for many models there is an exponential relationship between the posterior probability of an event and the error bar over that probability value, although, this relationship does not explain the length of the error bars to the full extent. I also observed such relationship for variation intervals as well. Nevertheless, it was not that ubiquitous among all of the models that I have analyzed. In general, low surprise index indicated rarity of the case at hand. Additionally, I observed that for some models (especially for tree-augmented naïve Bayes classifiers) wide error bars might indicate the rarity of the case at hand as well.

All of the measures carry a predictive information about the possible erroneous classification of a case at hand with Bayesian network models. The best measure to predict a faulty assignment of a class label was the length of the error bar. It performed very well across three model types and all datasets used in the experiment. The surprise index is also a good predictor of erroneous classification, although it is worse than the length of the error bar.

The length of the variation interval appeared to be the worst in predicting faulty assignment of the class labels. I observed a good performance for naïve Bayes classifier. In the experiments I performed, I was considering only cases with missing information at random with models of a specific structure. In practice, information is not missing at random. Variation intervals seem to be insightful when calculated as the information about the case is entered into the system piece by piece. Those intervals get tighter very quickly as we get more certain about the posterior probability about the event of our concern. That is why variation intervals are more useful in practical applications of Bayesian networks, where the user sets up observations about the case to the model manually.

When considering a problem of inaccessibility of the data used for training several models for predicting the same class variable, many methods for aggregation of models' outputs known from ensembles of classifiers are not suitable. This is why I considered confidence measures as means of assessing the competence of each model. When applied to ensembles of classifiers as weights of particular model's outputs, the confidence measures help to increase the accuracy of classification. The best results at that matter I obtained for tree-augmented naïve Bayes in the presence of missing values. Both error bar length and surprise index proofed to complement each other in that task.

10.2 FUTURE WORK

In the future, I would like to check how the length of variation interval performs in predicting erroneous output of a DSS when we provide information about the case to the system gradually, in the order starting from the most to the least valuable information from the point of view of class prediction. It would be a proper simulation of what a user of DSS does with the information provided to the system about the case at hand. As presented in examples Chapter 6, the variation intervals may get tight very quickly. The experiment that I have presented in Chapter 8 should give better results for the measure based on variation intervals with such setup.

To evaluate the performance of confidence measures in combining the outputs of the models while applying them in ensembles of classifiers I used a simple product formula (i.e., I created the overall weight for each model based on a product of measures). When just one of the measures gets a low value (we are not confident about the output of the system for the specific model), the weight based on the product of confidence measures drops significantly. I want to investigate other approaches to integrating confidence measures for a given model. For example, we can develop the weight for the given model based on an average of the measures. Particular measures would less influence it. Further, we may generalize that approach and consider a linear combination of the confidence measures as weights. Another problem arising in such situation is finding proper coefficients for the measures.

It would be beneficial to investigate why and when surprise index and error bar length perform well in improving the accuracy of an ensemble of classifiers based on Bayesian networks.

APPENDIX

DETAILED RESULTS

The results are presented in tables, where each table represents a pair of the model type and the measure. Each table presents the means and standard deviations (in the parenthesis) of area under the ROC curve from seven runs for all the combinations of missing data rate and corrupted data rate. The values are emphasized if at least six out of the seven runs gave AUC over 0.5 as an output. Additionally, the values are underlined if at least six out of the seven runs gave 0.55 as an output.

Tables with results are organized as presented in Table 8

Table 8: The organization of the tables presenting the results of the experiment

model type \ measure	error bar length	variation interval length	surprise index
naive Bayes	Table 9	Table 10	Table 11
tree-augmented naive Bayes	Table 12	Table 13	Table 14
augmented naive Bayes	Table 15	Table 16	Table 17

Table 9: Area under the ROC for predicting the correctness of the naive Bayes classification using length of the error bar

Missing rate	0		
Corrupted rate	0	0.2	0.4
adult merged	**0.808 (0.002)**	**0.788 (0.005)**	**0.764 (0.004)**
bands	**0.729 (0.042)**	**0.724 (0.038)**	**0.710 (0.032)**
cover type	**0.634 (0.006)**	**0.607 (0.006)**	**0.568 (0.004)**
dermatology	**0.974 (0.021)**	**0.946 (0.038)**	**0.925 (0.022)**
house votes	**0.885 (0.026)**	**0.866 (0.037)**	**0.850 (0.030)**
ionosphere	**0.808 (0.031)**	**0.813 (0.028)**	**0.759 (0.057)**
messidor	**0.626 (0.019)**	**0.638 (0.016)**	**0.602 (0.024)**
cervical cancer	**0.830 (0.034)**	**0.799 (0.093)**	**0.861 (0.037)**
thoracic surgery	**0.762 (0.020)**	**0.740 (0.062)**	**0.675 (0.036)**
wine red	**0.616 (0.021)**	**0.620 (0.023)**	**0.577 (0.020)**
wine white	**0.559 (0.014)**	**0.557 (0.014)**	**0.538 (0.009)**

Missing rate	0.2		
Corrupted rate	0	0.2	0.4
adult merged	**0.778 (0.004)**	**0.761 (0.006)**	**0.738 (0.005)**
bands	**0.705 (0.030)**	**0.699 (0.030)**	**0.652 (0.034)**
cover type	**0.637 (0.004)**	**0.627 (0.002)**	**0.597 (0.003)**
dermatology	**0.936 (0.026)**	**0.875 (0.029)**	**0.868 (0.035)**
house votes	**0.812 (0.041)**	**0.825 (0.047)**	**0.804 (0.065)**
ionosphere	**0.734 (0.052)**	**0.748 (0.042)**	**0.754 (0.039)**
messidor	**0.616 (0.021)**	**0.575 (0.030)**	**0.605 (0.038)**
cervical cancer	**0.834 (0.075)**	**0.869 (0.034)**	**0.909 (0.024)**
thoracic surgery	**0.689 (0.039)**	**0.664 (0.053)**	**0.638 (0.057)**
wine red	**0.608 (0.024)**	**0.594 (0.014)**	**0.565 (0.016)**
wine white	**0.561 (0.016)**	**0.559 (0.009)**	**0.537 (0.012)**

Missing rate	0.5		
Corrupted rate	0	0.2	0.4
adult merged	**0.722 (0.004)**	**0.701 (0.006)**	**0.681 (0.004)**
bands	**0.640 (0.019)**	**0.623 (0.022)**	**0.634 (0.034)**
cover type	**0.593 (0.004)**	**0.583 (0.003)**	**0.563 (0.006)**
dermatology	**0.767 (0.027)**	**0.786 (0.040)**	**0.741 (0.036)**
house votes	**0.761 (0.071)**	**0.747 (0.046)**	**0.715 (0.039)**
ionosphere	**0.701 (0.038)**	**0.677 (0.043)**	**0.666 (0.049)**
messidor	**0.589 (0.035)**	**0.570 (0.028)**	**0.549 (0.015)**
cervical cancer	**0.823 (0.099)**	**0.838 (0.045)**	**0.832 (0.023)**
thoracic surgery	**0.605 (0.062)**	**0.640 (0.079)**	**0.640 (0.084)**
wine red	**0.591 (0.019)**	**0.562 (0.011)**	**0.544 (0.029)**
wine white	**0.547 (0.014)**	**0.542 (0.007)**	**0.529 (0.016)**

Table 10: Area under the ROC for predicting the correctness of the naive Bayes classification using length of the variation interval

Missing rate	0		
Corrupted rate	0	0.2	0.4
adult merged	0.501 (0.004)	**0.758 (0.005)**	**0.774 (0.004)**
bands	0.499 (0.049)	**0.703 (0.048)**	**0.701 (0.031)**
cover type	0.500 (0.006)	**0.599 (0.005)**	**0.601 (0.008)**
dermatology	0.673 (0.214)	**0.942 (0.046)**	**0.907 (0.046)**
house votes	0.505 (0.082)	**0.836 (0.056)**	**0.862 (0.036)**
ionosphere	0.516 (0.062)	**0.814 (0.028)**	**0.751 (0.063)**
messidor	**0.517 (0.024)**	**0.624 (0.026)**	**0.603 (0.022)**
cervical cancer	0.527 (0.089)	**0.780 (0.086)**	**0.823 (0.051)**
thoracic surgery	0.545 (0.057)	**0.646 (0.052)**	**0.609 (0.019)**
wine red	0.495 (0.028)	**0.543 (0.020)**	**0.555 (0.013)**
wine white	0.495 (0.019)	**0.522 (0.015)**	0.502 (0.012)
Missing rate	0.2		
Corrupted rate	0	0.2	0.4
adult merged	0.500 (0.005)	**0.732 (0.005)**	**0.741 (0.005)**
bands	0.478 (0.020)	**0.677 (0.033)**	**0.610 (0.027)**
cover type	0.495 (0.004)	**0.579 (0.008)**	**0.584 (0.004)**
dermatology	0.496 (0.076)	**0.880 (0.044)**	**0.854 (0.037)**
house votes	**0.563 (0.021)**	**0.820 (0.056)**	**0.805 (0.068)**
ionosphere	**0.535 (0.043)**	**0.728 (0.039)**	**0.744 (0.039)**
messidor	0.488 (0.031)	**0.570 (0.020)**	**0.591 (0.042)**
cervical cancer	0.487 (0.105)	**0.832 (0.038)**	**0.849 (0.038)**
thoracic surgery	0.532 (0.041)	**0.569 (0.061)**	**0.586 (0.064)**
wine red	0.501 (0.012)	**0.531 (0.017)**	**0.528 (0.016)**
wine white	0.497 (0.011)	0.514 (0.015)	**0.517 (0.014)**
Missing rate	0.5		
Corrupted rate	0	0.2	0.4
adult merged	0.501 (0.004)	**0.679 (0.005)**	**0.685 (0.003)**
bands	0.504 (0.028)	**0.593 (0.028)**	**0.577 (0.045)**
cover type	0.500 (0.004)	**0.548 (0.006)**	**0.559 (0.004)**
dermatology	0.510 (0.046)	**0.766 (0.058)**	**0.712 (0.068)**
house votes	0.470 (0.055)	**0.727 (0.040)**	**0.714 (0.034)**
ionosphere	0.494 (0.046)	**0.660 (0.058)**	**0.632 (0.044)**
messidor	0.509 (0.024)	**0.554 (0.039)**	**0.556 (0.025)**
cervical cancer	0.496 (0.095)	**0.806 (0.046)**	**0.807 (0.023)**
thoracic surgery	0.521 (0.060)	**0.565 (0.074)**	**0.574 (0.035)**
wine red	0.487 (0.023)	0.503 (0.027)	**0.537 (0.023)**
wine white	0.508 (0.014)	**0.510 (0.014)**	**0.520 (0.014)**

Table 11: Area under the ROC for predicting the correctness of the naive Bayes classification using the surprise index

Missing rate	0		
Corrupted rate	0	0.2	0.4
adult merged	0.491 (0.002)	0.484 (0.006)	0.481 (0.003)
bands	0.456 (0.026)	0.423 (0.027)	0.468 (0.034)
cover type	0.507 (0.006)	0.497 (0.003)	0.490 (0.006)
dermatology	0.395 (0.262)	0.330 (0.089)	0.359 (0.089)
house votes	**0.808 (0.044)**	**0.782 (0.049)**	**0.754 (0.047)**
ionosphere	**0.583 (0.047)**	**0.558 (0.039)**	**0.541 (0.065)**
messidor	0.438 (0.029)	0.439 (0.021)	0.424 (0.017)
cervical cancer	**0.754 (0.040)**	**0.732 (0.046)**	**0.705 (0.028)**
thoracic surgery	**0.681 (0.067)**	**0.617 (0.071)**	**0.613 (0.069)**
wine red	**0.554 (0.024)**	**0.552 (0.018)**	**0.537 (0.030)**
wine white	**0.540 (0.006)**	**0.528 (0.014)**	**0.520 (0.012)**
Missing rate	0.2		
Corrupted rate	0	0.2	0.4
adult merged	0.494 (0.008)	0.490 (0.005)	0.485 (0.005)
bands	0.488 (0.036)	0.490 (0.043)	0.480 (0.038)
cover type	**0.538 (0.006)**	**0.517 (0.003)**	**0.510 (0.007)**
dermatology	**0.572 (0.107)**	**0.561 (0.093)**	0.494 (0.102)
house votes	**0.753 (0.059)**	**0.736 (0.054)**	**0.679 (0.112)**
ionosphere	0.530 (0.052)	**0.570 (0.045)**	**0.535 (0.035)**
messidor	0.439 (0.017)	0.428 (0.017)	0.457 (0.015)
cervical cancer	**0.694 (0.044)**	**0.698 (0.023)**	**0.696 (0.048)**
thoracic surgery	**0.603 (0.048)**	**0.588 (0.063)**	**0.573 (0.036)**
wine red	**0.561 (0.020)**	**0.549 (0.026)**	**0.533 (0.025)**
wine white	**0.531 (0.016)**	**0.528 (0.010)**	**0.517 (0.019)**
Missing rate	0.5		
Corrupted rate	0	0.2	0.4
adult merged	0.502 (0.004)	0.498 (0.006)	0.494 (0.005)
bands	0.504 (0.031)	0.506 (0.046)	0.517 (0.026)
cover type	**0.538 (0.005)**	**0.531 (0.007)**	**0.520 (0.005)**
dermatology	**0.587 (0.042)**	**0.583 (0.058)**	0.540 (0.041)
house votes	**0.673 (0.054)**	**0.672 (0.034)**	**0.657 (0.038)**
ionosphere	**0.587 (0.048)**	**0.593 (0.030)**	**0.571 (0.017)**
messidor	0.475 (0.024)	0.479 (0.018)	0.458 (0.033)
cervical cancer	**0.603 (0.084)**	**0.654 (0.044)**	**0.596 (0.070)**
thoracic surgery	**0.606 (0.048)**	**0.533 (0.058)**	**0.576 (0.069)**
wine red	**0.553 (0.013)**	**0.522 (0.018)**	**0.529 (0.024)**
wine white	**0.524 (0.010)**	**0.513 (0.010)**	**0.514 (0.008)**

Table 12: Area under the ROC for predicting the correctness of the tree-augmented naive Bayes classification using length of the error bar

Missing rate	0		
Corrupted rate	0	0.2	0.4
adult merged	**0.779 (0.003)**	**0.755 (0.005)**	**0.719 (0.004)**
bands	**0.743 (0.022)**	**0.726 (0.039)**	**0.718 (0.044)**
cover type	**0.629 (0.005)**	**0.604 (0.007)**	**0.569 (0.008)**
dermatology	**0.953 (0.032)**	**0.954 (0.016)**	**0.920 (0.027)**
house votes	**0.924 (0.022)**	**0.901 (0.035)**	**0.904 (0.029)**
ionosphere	**0.860 (0.044)**	**0.824 (0.032)**	**0.802 (0.039)**
messidor	**0.585 (0.031)**	**0.603 (0.016)**	**0.569 (0.028)**
cervical cancer	**0.920 (0.031)**	**0.944 (0.025)**	**0.929 (0.025)**
thoracic surgery	**0.663 (0.049)**	**0.691 (0.082)**	**0.660 (0.042)**
wine red	**0.607 (0.024)**	**0.594 (0.023)**	**0.552 (0.017)**
wine white	**0.538 (0.007)**	**0.518 (0.014)**	0.495 (0.017)

Missing rate	0.2		
Corrupted rate	0	0.2	0.4
adult merged	**0.722 (0.003)**	**0.708 (0.007)**	**0.682 (0.003)**
bands	**0.654 (0.049)**	**0.678 (0.051)**	**0.677 (0.043)**
cover type	**0.637 (0.004)**	**0.627 (0.002)**	**0.597 (0.003)**
dermatology	**0.903 (0.041)**	**0.864 (0.016)**	**0.836 (0.026)**
house votes	**0.874 (0.026)**	**0.829 (0.037)**	**0.812 (0.041)**
ionosphere	**0.755 (0.026)**	**0.816 (0.020)**	**0.735 (0.047)**
messidor	**0.558 (0.039)**	**0.559 (0.032)**	**0.561 (0.018)**
cervical cancer	**0.628 (0.045)**	**0.645 (0.070)**	**0.711 (0.037)**
thoracic surgery	**0.696 (0.049)**	**0.653 (0.053)**	**0.701 (0.045)**
wine red	**0.619 (0.021)**	**0.586 (0.020)**	**0.560 (0.019)**
wine white	**0.562 (0.013)**	**0.538 (0.007)**	**0.509 (0.007)**

Missing rate	0.5		
Corrupted rate	0	0.2	0.4
adult merged	**0.665 (0.010)**	**0.656 (0.007)**	**0.640 (0.004)**
bands	**0.612 (0.041)**	**0.560 (0.029)**	**0.575 (0.034)**
cover type	**0.622 (0.006)**	**0.623 (0.007)**	**0.606 (0.007)**
dermatology	**0.768 (0.050)**	**0.749 (0.047)**	**0.747 (0.028)**
house votes	**0.751 (0.025)**	**0.763 (0.062)**	**0.683 (0.043)**
ionosphere	**0.628 (0.070)**	**0.573 (0.030)**	**0.613 (0.054)**
messidor	**0.519 (0.034)**	0.517 (0.043)	**0.527 (0.023)**
cervical cancer	0.409 (0.038)	0.509 (0.040)	**0.559 (0.038)**
thoracic surgery	**0.622 (0.074)**	**0.678 (0.058)**	**0.670 (0.069)**
wine red	**0.592 (0.021)**	**0.597 (0.014)**	**0.557 (0.009)**
wine white	**0.582 (0.016)**	**0.547 (0.016)**	**0.537 (0.014)**

Table 13: Area under the ROC for predicting the correctness of the tree-augmented naive Bayes classification using length of the variation interval

Missing rate	0		
Corrupted rate	0	0.2	0.4
adult merged	0.497 (0.005)	0.499 (0.004)	0.502 (0.002)
bands	0.537 (0.052)	0.492 (0.027)	0.511 (0.052)
cover type	0.496 (0.005)	0.498 (0.009)	0.502 (0.005)
dermatology	0.532 (0.051)	0.435 (0.061)	0.462 (0.090)
house votes	0.499 (0.073)	**0.768 (0.049)**	**0.773 (0.062)**
ionosphere	0.567 (0.093)	0.504 (0.098)	0.502 (0.045)
messidor	0.513 (0.041)	0.509 (0.012)	0.493 (0.026)
cervical cancer	0.485 (0.087)	**0.708 (0.114)**	**0.797 (0.069)**
thoracic surgery	0.505 (0.071)	0.505 (0.059)	0.519 (0.042)
wine red	0.510 (0.023)	0.504 (0.031)	0.508 (0.030)
wine white	0.502 (0.018)	**0.509 (0.005)**	0.499 (0.016)
Missing rate	0.2		
Corrupted rate	0	0.2	0.4
adult merged	0.500 (0.005)	0.501 (0.006)	0.498 (0.004)
bands	0.489 (0.053)	0.502 (0.054)	0.494 (0.069)
cover type	0.497 (0.009)	0.502 (0.005)	0.497 (0.004)
dermatology	0.506 (0.072)	0.523 (0.070)	0.494 (0.048)
house votes	0.534 (0.067)	**0.637 (0.038)**	**0.668 (0.072)**
ionosphere	0.529 (0.046)	0.474 (0.058)	0.507 (0.035)
messidor	0.490 (0.033)	0.496 (0.035)	0.496 (0.025)
cervical cancer	0.508 (0.050)	**0.681 (0.043)**	**0.638 (0.033)**
thoracic surgery	0.477 (0.044)	0.560 (0.059)	0.533 (0.073)
wine red	0.501 (0.027)	0.496 (0.021)	0.490 (0.029)
wine white	0.497 (0.012)	0.496 (0.011)	**0.511 (0.008)**
Missing rate	0.5		
Corrupted rate	0	0.2	0.4
adult merged	**0.504 (0.005)**	0.499 (0.004)	0.500 (0.004)
bands	0.523 (0.048)	0.490 (0.051)	0.499 (0.052)
cover type	0.500 (0.007)	0.502 (0.005)	0.499 (0.006)
dermatology	0.510 (0.063)	0.500 (0.039)	0.483 (0.055)
house votes	0.503 (0.042)	**0.585 (0.045)**	**0.607 (0.068)**
ionosphere	0.520 (0.046)	0.484 (0.061)	0.458 (0.041)
messidor	**0.522 (0.025)**	**0.514 (0.023)**	0.510 (0.034)
cervical cancer	0.496 (0.080)	**0.641 (0.062)**	**0.665 (0.048)**
thoracic surgery	0.475 (0.046)	0.463 (0.069)	0.518 (0.070)
wine red	0.502 (0.019)	0.507 (0.031)	0.491 (0.022)
wine white	0.499 (0.009)	0.513 (0.023)	0.502 (0.011)

Table 14: Area under the ROC for predicting the correctness of the tree-augmented naive Bayes classification using the surprise index

Missing rate	0		
Corrupted rate	0	0.2	0.4
adult merged	0.494 (0.005)	0.491 (0.006)	0.492 (0.004)
bands	**0.568 (0.056)**	0.514 (0.057)	0.516 (0.063)
cover type	0.490 (0.004)	0.489 (0.005)	0.482 (0.010)
dermatology	0.580 (0.144)	0.479 (0.140)	0.571 (0.099)
house votes	**0.751 (0.027)**	**0.744 (0.073)**	**0.740 (0.044)**
ionosphere	0.415 (0.039)	0.473 (0.039)	0.472 (0.038)
messidor	0.468 (0.026)	0.471 (0.023)	0.448 (0.026)
cervical cancer	0.481 (0.082)	**0.702 (0.091)**	**0.682 (0.037)**
thoracic surgery	**0.670 (0.035)**	**0.636 (0.040)**	**0.614 (0.065)**
wine red	0.496 (0.013)	0.501 (0.027)	0.505 (0.015)
wine white	0.501 (0.011)	**0.517 (0.020)**	0.495 (0.016)
Missing rate	0.2		
Corrupted rate	0	0.2	0.4
adult merged	**0.533 (0.011)**	**0.523 (0.010)**	**0.515 (0.010)**
bands	0.521 (0.020)	0.552 (0.071)	0.524 (0.031)
cover type	**0.595 (0.003)**	**0.577 (0.003)**	**0.552 (0.006)**
dermatology	**0.605 (0.085)**	**0.607 (0.056)**	0.482 (0.061)
house votes	**0.716 (0.035)**	**0.694 (0.066)**	**0.721 (0.049)**
ionosphere	**0.550 (0.037)**	0.536 (0.076)	0.529 (0.051)
messidor	0.510 (0.037)	0.509 (0.042)	0.507 (0.029)
cervical cancer	**0.854 (0.062)**	**0.861 (0.030)**	**0.843 (0.021)**
thoracic surgery	**0.689 (0.063)**	**0.615 (0.053)**	**0.647 (0.069)**
wine red	**0.549 (0.014)**	**0.536 (0.013)**	0.525 (0.023)
wine white	**0.536 (0.009)**	**0.537 (0.009)**	**0.525 (0.010)**
Missing rate	0.5		
Corrupted rate	0	0.2	0.4
adult merged	**0.566 (0.012)**	**0.544 (0.011)**	**0.526 (0.007)**
bands	**0.559 (0.034)**	**0.532 (0.048)**	0.502 (0.036)
cover type	**0.614 (0.005)**	**0.602 (0.007)**	**0.582 (0.004)**
dermatology	**0.581 (0.041)**	**0.596 (0.049)**	0.545 (0.062)
house votes	**0.624 (0.071)**	**0.656 (0.046)**	**0.591 (0.065)**
ionosphere	**0.580 (0.058)**	**0.572 (0.062)**	0.550 (0.060)
messidor	0.507 (0.022)	0.496 (0.018)	**0.521 (0.023)**
cervical cancer	**0.894 (0.044)**	**0.901 (0.021)**	**0.858 (0.028)**
thoracic surgery	**0.609 (0.046)**	**0.633 (0.037)**	**0.592 (0.045)**
wine red	**0.568 (0.021)**	**0.561 (0.016)**	**0.528 (0.019)**
wine white	**0.571 (0.013)**	**0.530 (0.016)**	**0.527 (0.011)**

Table 15: Area under the ROC for predicting the correctness of the augmented naive Bayes classification using length of the error bar

Missing rate	0		
Corrupted rate	0	0.2	0.4
adult merged	**0.790 (0.009)**	**0.757 (0.017)**	**0.721 (0.007)**
bands	**0.723 (0.048)**	**0.718 (0.034)**	**0.698 (0.027)**
cover type	**0.615 (0.004)**	**0.595 (0.006)**	**0.567 (0.001)**
dermatology	**0.969 (0.022)**	**0.943 (0.041)**	**0.904 (0.049)**
house votes	**0.903 (0.026)**	**0.882 (0.049)**	**0.841 (0.061)**
ionosphere	**0.864 (0.054)**	**0.812 (0.075)**	**0.809 (0.033)**
messidor	**0.593 (0.043)**	**0.582 (0.033)**	**0.558 (0.033)**
cervical cancer	**0.931 (0.037)**	**0.962 (0.014)**	**0.902 (0.064)**
thoracic surgery	**0.676 (0.061)**	**0.661 (0.045)**	**0.667 (0.048)**
wine red	**0.599 (0.015)**	**0.576 (0.014)**	**0.552 (0.018)**
wine white	**0.538 (0.009)**	**0.524 (0.009)**	0.506 (0.013)
Missing rate	0.2		
Corrupted rate	0	0.2	0.4
adult merged	**0.737 (0.014)**	**0.720 (0.009)**	**0.693 (0.009)**
bands	**0.691 (0.034)**	**0.653 (0.031)**	**0.640 (0.051)**
cover type	**0.620 (0.015)**	**0.609 (0.008)**	**0.600 (0.004)**
dermatology	**0.888 (0.040)**	**0.865 (0.037)**	**0.846 (0.045)**
house votes	**0.863 (0.026)**	**0.823 (0.047)**	**0.777 (0.040)**
ionosphere	**0.789 (0.042)**	**0.731 (0.051)**	**0.745 (0.041)**
messidor	**0.549 (0.013)**	**0.546 (0.030)**	**0.559 (0.015)**
cervical cancer	**0.839 (0.038)**	**0.841 (0.051)**	**0.816 (0.068)**
thoracic surgery	**0.682 (0.054)**	**0.655 (0.068)**	**0.638 (0.032)**
wine red	**0.621 (0.023)**	**0.596 (0.018)**	**0.576 (0.027)**
wine white	**0.560 (0.009)**	**0.542 (0.014)**	**0.520 (0.008)**
Missing rate	0.5		
Corrupted rate	0	0.2	0.4
adult merged	**0.671 (0.005)**	**0.655 (0.005)**	**0.639 (0.008)**
bands	**0.584 (0.026)**	**0.545 (0.043)**	**0.579 (0.036)**
cover type	**0.611 (0.012)**	**0.603 (0.008)**	**0.597 (0.004)**
dermatology	**0.740 (0.043)**	**0.737 (0.062)**	**0.750 (0.043)**
house votes	**0.746 (0.029)**	**0.696 (0.050)**	**0.707 (0.032)**
ionosphere	**0.577 (0.035)**	**0.586 (0.056)**	**0.629 (0.059)**
messidor	**0.529 (0.021)**	**0.544 (0.027)**	0.533 (0.038)
cervical cancer	**0.606 (0.098)**	**0.691 (0.063)**	**0.754 (0.049)**
thoracic surgery	**0.635 (0.033)**	**0.604 (0.057)**	**0.632 (0.061)**
wine red	**0.613 (0.029)**	**0.603 (0.026)**	**0.562 (0.026)**
wine white	**0.564 (0.017)**	**0.551 (0.011)**	**0.537 (0.017)**

Table 16: Area under the ROC for predicting the correctness of the augmented naive Bayes classification using length of the variation interval

Missing rate	0		
Corrupted rate	0	0.2	0.4
adult merged	0.501 (0.006)	0.498 (0.005)	0.500 (0.006)
bands	0.503 (0.079)	0.490 (0.046)	0.498 (0.032)
cover type	0.497 (0.006)	0.498 (0.004)	0.503 (0.002)
dermatology	0.596 (0.224)	0.467 (0.168)	0.517 (0.127)
house votes	0.511 (0.077)	**0.613 (0.163)**	**0.617 (0.089)**
ionosphere	0.504 (0.112)	0.501 (0.061)	0.440 (0.072)
messidor	0.509 (0.031)	0.496 (0.022)	**0.525 (0.027)**
cervical cancer	0.487 (0.079)	0.603 (0.146)	**0.710 (0.117)**
thoracic surgery	0.481 (0.055)	0.517 (0.076)	0.511 (0.047)
wine red	**0.510 (0.011)**	0.505 (0.022)	0.502 (0.021)
wine white	0.504 (0.018)	0.497 (0.015)	0.509 (0.015)
Missing rate	0.2		
Corrupted rate	0	0.2	0.4
adult merged	0.502 (0.005)	0.500 (0.004)	**0.501 (0.006)**
bands	0.511 (0.052)	0.494 (0.055)	0.497 (0.036)
cover type	0.499 (0.005)	0.500 (0.003)	0.497 (0.002)
dermatology	0.521 (0.045)	0.513 (0.053)	0.508 (0.077)
house votes	0.501 (0.081)	0.557 (0.124)	**0.633 (0.124)**
ionosphere	0.479 (0.066)	0.519 (0.036)	0.465 (0.066)
messidor	0.506 (0.031)	0.506 (0.029)	0.502 (0.018)
cervical cancer	0.517 (0.018)	**0.583 (0.046)**	**0.577 (0.034)**
thoracic surgery	0.505 (0.035)	0.459 (0.071)	0.485 (0.045)
wine red	0.514 (0.013)	0.498 (0.015)	0.498 (0.028)
wine white	0.492 (0.015)	0.506 (0.019)	0.505 (0.013)
Missing rate	0.5		
Corrupted rate	0	0.2	0.4
adult merged	0.499 (0.004)	0.500 (0.006)	0.498 (0.004)
bands	**0.519 (0.018)**	0.525 (0.034)	0.517 (0.037)
cover type	0.501 (0.007)	0.500 (0.005)	0.502 (0.003)
dermatology	**0.542 (0.031)**	0.513 (0.055)	0.495 (0.031)
house votes	0.477 (0.069)	**0.559 (0.044)**	**0.581 (0.052)**
ionosphere	0.531 (0.051)	0.497 (0.043)	0.515 (0.070)
messidor	0.505 (0.022)	0.498 (0.012)	0.504 (0.037)
cervical cancer	0.512 (0.021)	**0.580 (0.043)**	**0.638 (0.055)**
thoracic surgery	0.489 (0.056)	**0.541 (0.051)**	0.509 (0.076)
wine red	0.499 (0.015)	0.503 (0.024)	0.499 (0.021)
wine white	0.498 (0.010)	0.511 (0.012)	0.503 (0.014)

Table 17: Area under the ROC for predicting the correctness of the augmented naive Bayes classification using the surprise index

Missing rate	0		
Corrupted rate	0	0.2	0.4
adult merged	0.495 (0.016)	0.489 (0.008)	0.483 (0.007)
bands	0.518 (0.039)	0.522 (0.050)	0.494 (0.053)
cover type	0.495 (0.006)	0.499 (0.006)	0.492 (0.005)
dermatology	0.694 (0.226)	0.480 (0.185)	0.469 (0.059)
house votes	**0.770 (0.060)**	**0.756 (0.094)**	**0.690 (0.072)**
ionosphere	0.439 (0.087)	0.468 (0.053)	0.521 (0.082)
messidor	0.473 (0.031)	0.466 (0.035)	0.445 (0.023)
cervical cancer	0.498 (0.095)	0.564 (0.103)	**0.691 (0.119)**
thoracic surgery	**0.614 (0.067)**	**0.633 (0.051)**	**0.601 (0.049)**
wine red	0.485 (0.014)	0.507 (0.017)	0.503 (0.021)
wine white	0.504 (0.008)	0.507 (0.014)	0.502 (0.013)

Missing rate	0.2		
Corrupted rate	0	0.2	0.4
adult merged	**0.518 (0.018)**	0.507 (0.012)	**0.508 (0.007)**
bands	0.515 (0.057)	0.511 (0.025)	0.514 (0.040)
cover type	**0.602 (0.016)**	**0.574 (0.013)**	**0.546 (0.014)**
dermatology	**0.592 (0.092)**	**0.555 (0.042)**	**0.611 (0.031)**
house votes	**0.727 (0.046)**	**0.723 (0.022)**	**0.677 (0.053)**
ionosphere	**0.532 (0.031)**	0.544 (0.088)	**0.572 (0.047)**
messidor	0.496 (0.016)	**0.511 (0.025)**	0.502 (0.029)
cervical cancer	**0.755 (0.064)**	**0.796 (0.045)**	**0.814 (0.029)**
thoracic surgery	**0.654 (0.059)**	**0.642 (0.030)**	**0.593 (0.050)**
wine red	**0.555 (0.026)**	**0.541 (0.013)**	**0.545 (0.017)**
wine white	**0.541 (0.011)**	**0.534 (0.016)**	**0.520 (0.011)**

Missing rate	0.5		
Corrupted rate	0	0.2	0.4
adult merged	**0.544 (0.021)**	**0.527 (0.017)**	**0.512 (0.011)**
bands	**0.525 (0.040)**	0.516 (0.050)	0.529 (0.034)
cover type	**0.632 (0.008)**	**0.593 (0.013)**	**0.578 (0.018)**
dermatology	**0.621 (0.041)**	**0.562 (0.062)**	**0.572 (0.050)**
house votes	**0.653 (0.038)**	**0.636 (0.017)**	**0.625 (0.037)**
ionosphere	**0.553 (0.025)**	0.543 (0.052)	**0.582 (0.034)**
messidor	**0.534 (0.027)**	0.511 (0.022)	0.505 (0.018)
cervical cancer	**0.860 (0.043)**	**0.869 (0.031)**	**0.836 (0.024)**
thoracic surgery	**0.585 (0.031)**	**0.573 (0.049)**	**0.631 (0.026)**
wine red	**0.573 (0.023)**	**0.563 (0.036)**	**0.547 (0.025)**
wine white	**0.557 (0.018)**	**0.545 (0.016)**	**0.526 (0.009)**

BIBLIOGRAPHY

Antal, B. and Hajdu, A. (2014). An ensemble-based system for automatic screening of diabetic retinopathy. *Knowledge-Based Systems*, 60:20–27.

Atkinson, A. B. (1970). On the measurement of inequality. *Journal of Economic Theory*, 2(3):244–263.

BayesFusion, LLC (2019). SMILE Programmer's Manual.

Beinlich, I. A., Suermondt, H. J., Chavez, R. M., and Cooper, G. F. (1989). *The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks.* Springer.

Blackard, J. (1998). Comparison of neural networks and discriminant analysis in predicting forest cover types. *Ph.D. dessertation. Department of Forest Science, Colorado State University.*

Bouckaert, R. R., Castillo, E., and Gutiérrez, J. (1996). A modified simulation scheme for inference in Bayesian networks. *International Journal of Approximate Reasoning*, 14(1):55–80.

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78:1.

Cano, J., Delgado, M., and Moral, S. (1993). An axiomatic framework for propagating uncertainty in directed acyclic networks. *International Journal of Approximate Reasoning*, 8(4):253–280.

Castillo, E., Gutierrez, J. M., and Hadi, A. S. (1998). Improving search-based inference in Bayesian networks. In *Proceedings of the Eleventh International Florida Artificial*

Intelligence Research Society Conference, pages 405–409. AAAI Press.

Castillo, E., Hadi, A. S., Balakrishnan, N., and Sarabia, J.-M. (2005). *Extreme value and related models with applications in engineering and science*. Wiley-Interscience Hoboken, NJ, USA.

Clemen, R. T. (1996). *Making Hard Decisions: An Introduction to Decision Analysis*. Duxbury Press.

Conaty, D., Del Rincon, J. M., and De Campos, C. P. (2018). Cascading sum-product networks using robustness. In *International Conference on Probabilistic Graphical Models*, pages 73–84.

Cooper, G. F. (1990). The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 42:393–405.

Cooper, G. F. and Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9(4):309–347.

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553.

Coupé, V. M. and Van Der Gaag, L. C. (2002). Properties of sensitivity analysis of Bayesian belief networks. *Annals of Mathematics and Artificial Intelligence*, 36(4):323–356.

Cruz, R. M., Sabourin, R., and Cavalcanti, G. D. (2018). Dynamic classifier selection: Recent advances and perspectives. *Information Fusion*, 41:195–216.

Cypko, M. A., Stoehr, M., Kozniewski, M., Druzdzel, M. J., Dietz, A., Berliner, L., and Lemke, H. U. (2017). Validation workflow for a clinical Bayesian network model in multidisciplinary decision making in head and neck oncology treatment. *International Journal of Computer Assisted Radiology and Surgery*, 12(11):1959–1970.

Daly, R., Shen, Q., and Aitken, S. (2011). Learning Bayesian networks: approaches and issues. *The Knowledge Engineering Review*, 26(2):99–157.

Davis, J. and Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 233–240. ACM.

Dawid, A. P. (1982). The well-calibrated Bayesian. *Journal of the American Statistical Association*, 77(379):605–610.

DeGroot, M. H. and Fienberg, S. E. (1983). The comparison and evaluation of forecasters. *The Statistician*, pages 12–22.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (methodological)*, 39:1–38.

Dheeru, D. and Karra Taniskidou, E. (2017). UCI machine learning repository.

Donald, M. R. and Mengersen, K. L. (2014). Methods for constructing uncertainty intervals for queries of Bayesian nets. *Australian & New Zealand Journal of Statistics*, 56(4):407–427.

Druzdzel, M. J. (1994). Some properties of joint probability distributions. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, pages 187–194. Morgan Kaufmann Publishers Inc.

Druzdzel, M. J., Onisko, A., Schwartz, D., Dowling, J. N., and Wasyluk, H. (1999). Knowledge engineering for very large decision-analytic medical models. In *Proceedings of the AMIA Symposium*, page 1049. American Medical Informatics Association.

Druzdzel, M. J. and Van Der Gaag, L. C. (2000). Building probabilistic networks: "Where do the numbers come from?". *IEEE Transactions on knowledge and data engineering*, 12(4):481–486.

Efron, B. and Tibshirani, R. J. (1993). *An Introduction to the Bootstrap.* Chapman and Hall.

Egan, J. P. (1975). *Signal detection theory and ROC analysis.* Academic Press.

Evans, B. and Fisher, D. (1994). Overcoming process delays with decision tree induction. *IEEE Expert: Intelligent Systems and Their Applications*, 9(1):60–66.

Fagiuoli, E. and Zaffalon, M. (1998). 2U: An exact interval propagation algorithm for polytrees with binary variables. *Artificial Intelligence*, 106(1):77–107.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874.

Feng, G., Zhang, J.-D., and Liao, S. S. (2014). A novel method for combining bayesian networks, theoretical analysis, and its applications. *Pattern Recognition*, 47(5):2057–2069.

Fernandes, K., Cardoso, J. S., and Fernandes, J. (2017). Transfer learning with partial observability applied to cervical cancer screening. In *Iberian conference on pattern recognition and image analysis*, pages 243–250. Springer.

Gastwirth, J. L. (1972). The estimation of the Lorenz curve and Gini index. *The review of economics and statistics*, pages 306–316.

Geman, S., Bienenstock, E., and Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, 4(1):1–58.

Gould, M. K., Ananth, L., and Barnett, P. G. (2007). A clinical model to estimate the pretest probability of lung cancer in patients with solitary pulmonary nodules. *Chest*, 131(2):383–388.

Güvenir, H. A., Demiröz, G., and Ilter, N. (1998). Learning differential diagnosis of erythemato-squamous diseases using voting feature intervals. *Artificial intelligence in medicine*, 13(3):147–165.

Habbema, J. (1976). Models for diagnosis and detection of combinations of diseases. *Decision Making and Medical Care*, pages 399–411.

Henrion, M. (1988). Propagation of uncertainty by probabilistic logic sampling in Bayes networks. In *Uncertainty in Artificial Intelligence*, volume 2, pages 149–164.

Howard, R. A. (1988). Decision analysis: practice and promise. *Management Science*, 34(6):679–695.

Jensen, F. V., Chamberlain, B., Nordahl, T., and Jensen, F. (1990). Analysis in HUGIN of data conflict. In *Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence*, pages 519–528. Elsevier Science Inc.

Kanwar, M. K., Lohmueller, L. C., Kormos, R. L., Loghmanpour, N. A., Benza, R. L., Mentz, R. J., Bailey, S. H., Murali, S., and Antaki, J. F. (2017). Low accuracy of the HeartMate risk score for predicting mortality using the INTERMACS registry data. *ASAIO Journal*, 63(3):251–256.

Kjærulff, U. and van der Gaag, L. C. (2000). Making sensitivity analysis computationally efficient. In *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, pages 317–325. Morgan Kaufmann Publishers Inc.

Kohavi, R. (1996). Scaling up the accuracy of naive-Bayes classifiers: a decision-tree hybrid. In *Second International Conference on Knowledge Discovery and Data Mining*, pages 202–207.

Kozniewski, M., Cypko, M. A., and Druzdzel, M. (2016). How reliable is a measure of model reliability? Bootstrap confidence intervals over validation results. *Advances in Computer Science Research.*

Kristensen, K. and Rasmussen, I. A. (2002). The use of a bayesian network in the design of a decision support system for growing malting barley without use of pesticides. *Computers and Electronics in Agriculture*, 33(3):197–217.

Krüger, M. and Hirschhäuser, D. (2009). Source conflicts in Bayesian identification. In *GI Jahrestagung*, pages 2485–2490.

Laskey, K. B. (1991). Conflict and surprise: Heuristics for model revision. In *Proceedings of the Seventh conference on Uncertainty in Artificial Intelligence*, pages 197–204. Morgan Kaufmann Publishers Inc.

Laskey, K. B. (1995). Sensitivity analysis for probability assessments in Bayesian networks. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(6):901–909.

Lauritzen, S. L. and Spiegelhalter, D. J. (1988). Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 157–224.

Marcot, B. G. (2012). Metrics for evaluating performance and uncertainty of Bayesian network models. *Ecological Modelling*, 230:50–62.

Morgan, M. G., Henrion, M., and Small, M. (1992). *Uncertainty: a guide to dealing with uncertainty in quantitative risk and policy analysis.* Cambridge University Press.

Murphy, A. H. and Winkler, R. L. (1977). Reliability of subjective probability forecasts of precipitation and temperature. *Applied Statistics*, pages 41–47.

Naeini, M. P., Cooper, G. F., and Hauskrecht, M. (2015). Obtaining well calibrated probabilities using Bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence.*

Niculescu-Mizil, A. and Caruana, R. (2005). Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 625–632. ACM.

Oniśko, A., Druzdzel, M. J., and Wasyluk, H. (2000). Extension of the Hepar II model to multiple-disorder diagnosis. *Intelligent Information Systems* , pages 303–313.

Oniśko, A., Druzdzel, M. J., and Wasyluk, H. (2001). Learning Bayesian network parameters from small data sets: Application of Noisy-OR gates. *International Journal of Approximate Reasoning*, 27(2):165–182.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Perrone, M. P. and Cooper, L. N. (1992). When networks disagree: Ensemble methods for hybrid neural networks. Technical report, Brown University, Providence RI, Institute for Brain and Neural Systems.

Pradhan, M., Provan, G., Middleton, B., and Henrion, M. (1994). Knowledge engineering for large belief networks. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, pages 484–490. Morgan Kaufmann Publishers Inc.

Przytula, K. W., Dash, D., and Thompson, D. (2003). Evaluation of Bayesian networks used for diagnostics. In *2003 IEEE Aerospace Conference Proceedings*, volume 7, pages 3177–3187. IEEE.

Raghavan, V., Bollmann, P., and Jung, G. S. (1989). A critical investigation of recall and precision as measures of retrieval system performance. *ACM Transactions on Information Systems (TOIS)*, 7(3):205–229.

Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review*, 33(1-2):1–39.

Schlimmer, J. (1987). Concept acquisition through representational adjustment. *Doctoral dissertation (TR-87-19) Department of Information and Computer Science University of California.*

Sigillito, V. G., Wing, S. P., Hutton, L. V., and Baker, K. B. (1989). Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, 10(3):262–266.

Smith, L. and Gal, Y. (2018). Understanding measures of uncertainty for adversarial example detection. In *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence*, pages 560–569.

Spackman, K. A. (1989). Signal detection theory: Valuable tools for evaluating inductive learning. In *Proceedings of the Sixth International Workshop on Machine Learning*, pages 160–163. Elsevier.

Van Allen, T., Singh, A., Greiner, R., and Hooper, P. (2008). Quantifying the uncertainty of a belief net response: Bayesian error-bars for belief net inference. *Artificial Intelligence*, 172(4-5):483–513.

Woloszynski, T. and Kurzynski, M. (2011). A probabilistic model of classifier competence for dynamic ensemble selection. *Pattern Recognition*, 44(10-11):2656–2668.

Zagorecki, A., Kozniewski, M., and Druzdzel, M. J. (2015). An approximation of surprise index as a measure of confidence. In *AAAI Fall Symposium-Technical Report*, pages 39–41.

Zięba, M., Tomczak, J. M., Lubicz, M., and Świątek, J. (2014). Boosted SVM for extracting rules from imbalanced data in application to prediction of the post-operative life expectancy in the lung cancer patients. *Applied Software Computing*, 14:99–108.