Improving Bayesian Learning Using Public Knowledge

Farid Seifi[1], Chris Drummond[2], Nathalie Japkowicz[1] and Stan Matwin[1,3]

[1] School of Information Technology and Engineering
University of Ottawa
Ottawa Ontario Canada, K1N 6N5
fseif050@uottawa.ca,nat@site.uottawa.ca,stan@site.uottawa.ca
[2] Institute for Information Technology,
National Research Council Canada,
Ottawa, Ontario, Canada, K1A 0R6
Chris.Drummond@nrc-cnrc.gc.ca
[3] Institute for Computer Science, Polish Academy of Sciences, Warsaw, Poland

Abstract. Both intensional and extensional background knowledge have previously been used in inductive problems to complement the training set used for a task. In this research, we propose to explore the usefulness, for inductive learning, of a new kind of intensional background knowledge: the inter-relationships or conditional probability distributions between subsets of attributes. Such information could be mined from publicly available knowledge sources but including only some of the attributes involved in the inductive task at hand. The purpose of our work is to show how this information can be useful in inductive tasks, and under what circumstances. We will consider injection of background knowledge into Bayesian Networks and explore its effectiveness on training sets of different sizes. We show that this additional knowledge not only improves the estimate of classification accuracy it also reduces the variance in the accuracy of the model.

Key words: Bayesian Networks, Public Knowledge, Classification.

1 Introduction

While standard machine learning acquires knowledge from instances of the learning problem, there has always been interest in a more cognitively plausible scenario in which learning - besides the training instances - utilizes also background knowledge relevant for the problem. In many inductive problems, the training set, which is a set of labeled samples, could be complemented using intensional or extensional background knowledge in order to improve the learning performance [11, 19]. In Inductive Logic Programming, intensional background knowledge is provided in the form of a theory expressed in logical form. In Semi-Supervised Learning, the extensional background knowledge is provided in the form of unlabeled data.

In this research, we propose to explore a different type of intensional background knowledge. In many domains, there exist publicly available very large, and related, data sets, for example from demographics and statistical surveys. This sort of information is ubiquitous: it is published by many national governments [2, 3, 5]; international organizations [1, 6, 4]; and private companies. Such data sets may not have exactly the same attributes as the data set we are studying. However, using an intensionalising process [13], we can derive intensional background knowledge, in the form of distributions, from this extensional background knowledge, given as collections of instances. A question that we consider here is whether it is possible to use such information to improve the performance of learning methods in machine learning problems.

Let us consider a simple medical example. Suppose we are learning from data a model for the prediction of heart attacks in patients. The data used in the inductive learning of this model may include attributes describing sleep disturbance, as a disease outcome, and stress, as a disease, but does not include enough instances to relate these attributes in a statistically significant way. There exists, independently of the data used in model building, a large medical survey that describes quantitatively sleep disturbance in patients who experience cardiac problems or stress. This set could be used in learning a better predictive model, capturing the important relationship between sleep disturbance, stress, and a heart attack, if we can integrate the data from the medical study with the data we are using in learning the predictive model.

The big challenge in this research is how such background knowledge can be integrated with the existing data sets. Bayesian learning is a natural candidate as it draws on distributional data for its assessment of the probabilities of an instance belonging to different classes of the concept. In Bayesian Networks the attribute inter-relationships are encoded into a network structure. We propose here to replace parts of this structure, some of the conditional probability distributions, with more accurate alternatives, which are available as background knowledge contained in large public data sets, e.g. statistical surveys.

The paper is structured as follows: In Section 2, Bayesian learning is reviewed with a simple example. Section 3 discusses how background knowledge is added to the network. In Section 4 experimental results are provided. Section 5 contains discussion and future work.

2 Learning and classification using Bayesian Networks

In a Bayesian network [17, 16], there is a structure which encodes a set of conditional independence assumptions between attributes; a node is conditionally independent of its non-descendants given its parents. Also, there are conditional probability distributions capturing each attribute's dependency on others, typically represented by multi-dimensional tables. Together, these define the joint probability distribution of the attributes and class. With such a distribution, we can use Bayes rule to do inference, i.e. determine the probability of some unobserved variable. There exist many different ways of building Bayesian net-

works from training data [7]. We used the software package *BN predictor* [9, 10] to build the network and used a maximum likelihood estimator (frequency counts) to construct the tables. To build these tables, we need the number of training samples for each permutation of attribute value that is involved in the conditional probabilities. These are normalized to give probabilities.

As a simple medical example, consider the diabetes diagnosis problem [12], whose network is given in figure 1. In the following equations, the terms A, N, M, I, G, D represent Age, Number of pregnancy, Mass, Insulin, Glucose and Diabetes, respectively. We need the posterior probability of the class Diabetes $P(D|A, N, M, I, G)$given the other attributes. From Bayes rule we can rewrite this as in equation 1.

$$P(D|A, N, M, I, G) = \frac{P(A, N, M, I, G, D)}{P(A, N, M, I, G)} \tag{1}$$

To classify a new example $S = (A_i, N_i, M_i, I_i, G_i)$, given by values for all attributes except the class diabetes, we chose the class with the highest posterior probability, as in equation 2

$$\underset{D\in(Yes,No)}{Argmax}\ P(D|A_i, N_i, M_i, I_i, G_i) = \underset{D\in(Yes,No)}{Argmax}\ P(A_i, N_i, M_i, I_i, G_i, D) \tag{2}$$

Using a Bayesian network, we can construct the joint probability by simply multiplying a few independent terms as in equation 3. From figure 1, the arrows indicate which conditional probabilities must represented by tables. For example, there are 3 arrows going from the mass, age and # pregnacies nodes to that of the class. This accounts for the $P(D|M, A, N)$term in the equation. There is a single arrow going from diabetes to insulin accounting for the term $P(I|D)$. Continuing this process for every arrow will produce equation 3.

$$P(A, N, M, I, G, D) = P(A)P(N)(M|A, N)P(D|M, A, N)P(I|D)P(G|I, D) \tag{3}$$

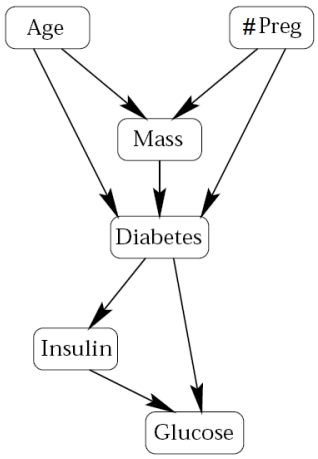

Fig. 1. A Bayesian network for diabetes diagnosis

The probabilities in equation 3 which do not include a term for class D , like $P(M|A,N)$, have no effect on the result of equation 2. For a single sample from the test set, all the attributes except the class are defined. Thus any terms in the joint probability that do not include a term for the class are identical. They can be safely ignored for classification.

3 Injecting Public Knowledge into a Network

Normally, we obtain the conditional probability distributions which we use in Bayesian Network inference from the training set. If we do not have enough training data samples, our estimates of the true distribution will be poor and the result will not be an accurate classifier. These distributions are independent from each other, so it could be possible to improve the performance even by replacement of a few of them with accurate alternatives, which we could find from statistical surveys.

We propose improving Bayesian networks by replacing some of the conditional probability distributions - represented in the form of tables and corresponding to the edges of the network - with their accurate alternatives which are available as background knowledge.

For example, in the Bayesian Network for diabetes diagnosis presented in section 2, suppose we have an accurate distribution of insulin given the diabetes, $P(I|D)$, from a large demographic survey. If we use this accurate distribution, instead of the one which is extracted from the limited training set, together with other distributions, which are extracted for other nodes of the network from small training set, and we apply them in formula 1, the performance of the Bayesian Network should improve.

4 Experiments

The ideal process to test this approach is to use real data along with some statistical surveys which could provide us with accurate conditional probability distributions that occur in our network. But if we are to experimentally investigate the usefulness of our approach, we need to be sure that the distributions that we use as background knowledge are accurately representing the real distribution of our data. Otherwise, they may negatively impact our results. In order to avoid such situations we simulate the problem as explained below.

Due to the imperfections of Bayesian Network constructors, it is probable that the extracted network for a data set contains some incorrect attribute dependencies. If we replace a conditional probability distribution, which is extracted based on such relations, the result may not be a significantly better classifier. Such situations would also negatively bias any conclusions about the reliability and usefulness of our method. In order to avoid this problem, related to a lack of sufficient real-world knowledge, in our second experiment we use an artificial data set for which we know the correct attribute independencies.

4.1 Experimental setup

For the purpose of our experiments, we have made some simplifying assumptions about the sizes of the datasets used. These assumptions are parameters of the experiments and can be easily changed. In particular, we assume that a large data set, which is a representative sample of the "huge" dataset (the whole universe of interest), exists. We model our approach by using a large real data set (or a large generated artificial data set) to supply us with highly accurate conditional probability distributions for the attributes involved in a Bayesian Network classifier, trained on a much smaller sample of a huge dataset. In this manner we simulate having the relevant information available from statistical surveys. For this purpose a real or artificial data set with 20,000 instances, which represents the universe of interest, is considered a huge data set. In each experiment we sample a large data set from this huge data set, containing 50%, 10,000 samples. Some of the conditional probability distributions are extracted using this large data set. Since these distributions are extracted from a large sets of instances, they are similar to what is available from statistical surveys. A part of the huge data set, 10%, 2,000 samples, is held out as testing set in each experiment. A very small subset of the huge data set is sampled as the training set. In these experiments 0.5%, 100 instances, of the huge data set are sampled as a training set. In many real world learning problems we only have such small training data sets. Using this small training set we build all conditional probability distributions, which are not very accurate, for all the nodes in the Bayesian Network.

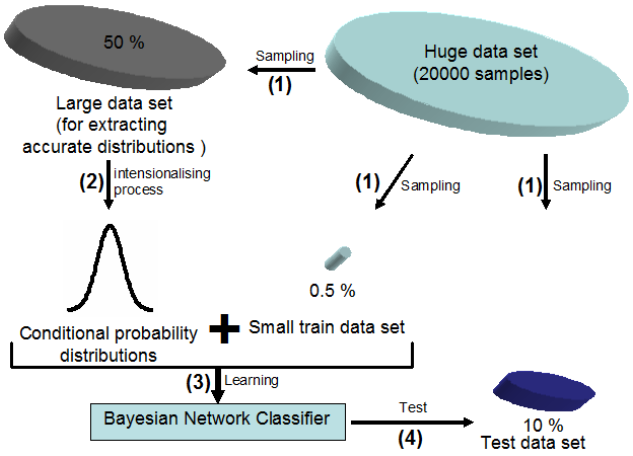

Fig. 2. Experimental setup.

What we want to show is the effect of replacing these inaccurate conditional probability distributions, extracted from the small training set, with accurate alternatives, which in real problems might be available, for example from statistical

surveys. For concluding that the replacement of a selected set of distributions makes a better classifier we run a set of experiments. In each experiment the large data set as well as the training and testing sets are sampled again from the huge data set. Then the classifier is trained using the small training set. More specifically, potentially inaccurate conditional probability distributions are built from the training set. Instead of using statistical surveys to extract accurate distributions, we use the distributions which were obtained from the large data set. Then we replace the selected set of distributions with accurate alternatives and compute the performance of the new modified classifier. We run several experiments with the same replacements and then we use paired t-test to see whether these sets of replacements make a significantly better classifier or not. Our experiments show that replacing more distributions results in a more accurate classifier unless a distribution is not extracted based on correct attribute dependencies. The Letter data set from the UCI machine learning repository [8] is used as the real data set. In addition, an artificial data set from the heart attack domain is used in a second experiment.

4.2 Experiments with the Letter data set

In these experiments we will see the effect of replacing the conditional probability distributions on a real data set. The objective of classifiers on the letter data set is to identify each of a large number of black-and-white rectangular pixel displays as one of the 26 capital letters in the English alphabet. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli. Each stimulus was converted into 16 primitive numerical attributes. All these numerical attributes contain statistical moments and edge counts, in the box containing the letter, which were then scaled to fit into a range of integer values from 0 through 15. For example y-bar is the mean of y of 'on' pixels in the box; *xy2br* is the mean of $x \times y \times y$; and *xegvy* is the correlation of x-edge with y. See [14], where the dataset was introduced, for details.

In these experiments we converted 26 classes to two classes by dividing the letters into two groups of the first and the second 13 letters in the English alphabet. This binarization of the letter recognition task makes it hard, as there are no obvious differences between the letters in the first and second half of the alphabet. Using the Bayesian Network constructor package discussed in [10], the network, which is shown in figure 3, is extracted. In this network all 16 attributes, which are nodes of the network, are shown as rectangles. Conditional dependencies are represented by arrows. Arrows represent the parent-child relationship, with a parent in the start and a child in the end of an arrow. Each node in the network is conditionally dependent on its parents. For Bayesian inference we only need to extract the conditional probability distribution of the nodes which have at least one incoming arrow from the class node. In this network we need to extract these distributions for nodes *y-bar*, *xegvy* and *xy2br*.

Figure 4 shows an example of how the difference between the accuracy of the unmodified model and of the modified model - in which the conditional proba-

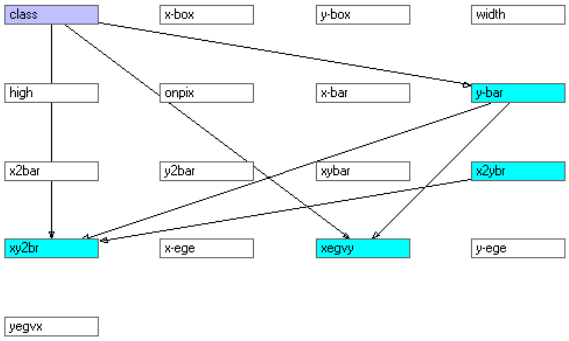

Fig. 3. Bayesian Network extracted for Letter data set.

bility distribution of attribute *xy2br* is replaced with the accurate alternative - changes with respect to the size of the training set. The greatest difference between the accuracy of both the modified and unmodified models appears when the size of the training set is very small. In order to focus on the effectiveness of our approach, we use small training sets, 0.5% of huge data set.

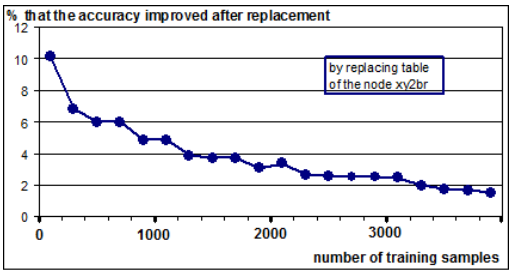

Fig. 4. Accuracy against size of training set for real data.

Next we want to show that replacements of the conditional probability distributions with accurate alternatives make significantly better classifiers. We run several experiments, as explained in the experimental setup subsection, all with the same replacements. The accuracies of these experiments on the modified models are compared with unmodified ones, using the paired t-test, to show that those specific replacements made a significantly better classifier. Figure 5 illustrates an example of the normal distributions for the accuracies of the unmodified and modified models, in which the conditional probability distribution of attribute *xy2br* is replaced with the accurate alternative. The bold curve represents the normal distribution for the accuracies of the unmodified model in different experiments. In all cases, as in the example of figure 5, the variance of the accuracies of the modified model is smaller than that of the unmodified

model. This means that when we replace a conditional probability distribution in a Bayesian Network with an accurate alternative, the new model tends to be more robust when sampling new data sets for training and testing.

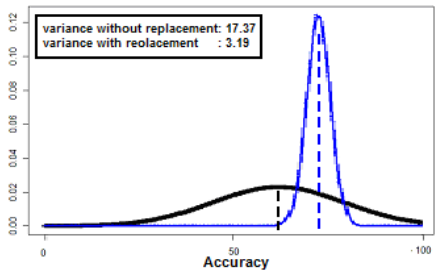

Fig. 5. Distribution of accuracies for unmodified (bold curve) and modified model.

We have tested the effect of replacement of different permutations of conditional probability distributions in the letter data set. In the extracted network for the letter data set we only have 3 conditional probability distributions, so the number of permutations is 3! = 6. Table 1 shows the results for different sets of distribution replacements. For each set of replacements it tells whether the modified model is significantly better than the unmodified one or not. These results are obtained using the paired t-test with 95% confidence interval.

Table 1. Accuracy for different replacements in the Bayesian Network on the letter data set as well as the results of the t-test for 20 different experiments.

experiment	**no change**	**y-bar**	**xy2br**	**xegvy**	**y-bar xy2br**	**y-bar xegvy**	**Xy2br xegvy**	**y-bar xy2br xegvy**
Average of the accuracy	61.7	66.1	72.3	61.4	72.9	64.3	71.4	72.7
$Varianceofaccuracy$		4.36	10.62	-0.338	11.163	2.583	9.663	11.015
$T-Testresult:$		**ESS**	**ESS**	**NSS**	**ESS**	**VSS**	**ESS**	**ESS**

* ESS- extremely statistically significant * VSS- Very statistically significant * NSS- Not statistically significant * SS- Statistically Significant

Replacing the conditional probability distribution of *y-bar* or *xy2br* leads to a significantly better classifier. But, when we replace *xegvy* with the accurate alternative we obtain a less accurate classifier. One reason is that, according to attribute evaluators (such as information gain and chi square), this attribute has less effect on the results of classification than the other two. The attribute evaluators ranked attributes *xy2br*, *y-bar* and *xegvy*, which are nodes of our network, as 1, 2 and 3, respectively. The effectiveness of replacement of the

conditional probability distribution of an attribute is directly related to the correctness of all its conditional dependencies. Therefore, another reason for this negative result is that the conditional dependencies of attribute *xegvy* may not extracted correctly. This problem is investigated in subsequent experiments and also discussed in more detail in the discussion section.

4.3 Experiments with the artificial data set

In these experiments, we want to show the effect of the accuracy of the extracted conditional dependencies on the accuracy of modified classifiers using our approach. The data set, used in these experiments, is generated using a heart attack data generator (which we designed ourselves) which generates data samples with 21 different attributes including the class. The objective of classifiers on this data set is heart attack diagnosis. The attributes are partitioned

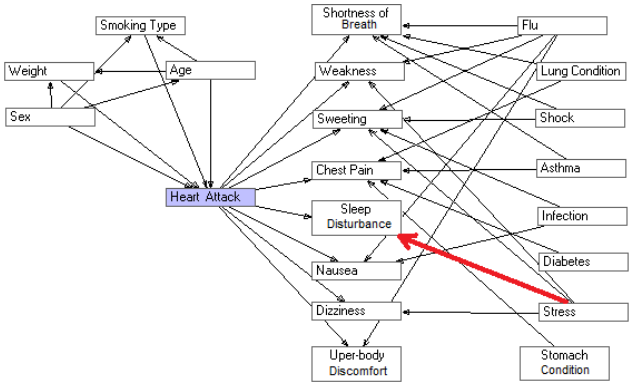

Fig. 6. Bayesian Network extracted for Heart attack data set.

into 4 groups: conditions, concept, outcomes and contexts. The data generator uses conditional probability distributions between attributes which are extracted from statistical surveys and the medical science literature. The data generator is used to generate a data set of 20,000 samples. The class attribute, heart attack, could be positive or negative.

The first part of this experiment consists of extracting the Bayesian Network or in other words the conditional independence between attributes. Since we know the exact conditional dependencies, used for data generation, we have manually defined the "true" network. But the relation between stress and sleep disturbance in this network, which is shown with the bold arrow in figure 6, is omitted on purpose, while in the data generation process sleep disturbance is conditionally dependant on stress. This is done in order to find out what the effect of replacement of a conditional probability distribution, which is extracted based on wrong conditional dependencies in the network, would be. Figure 6 shows the Bayesian Network which is used in this experiment.

Again, to focus the effect of replacement of conditional probability distributions, we use a very small training data set. Figure 7 shows the difference in the accuracy of the unmodified model with the modified one—in which the upper body discomfort distribution (*Upbd*) is replaced by a more accurate distribution—as a function of different training set sizes.

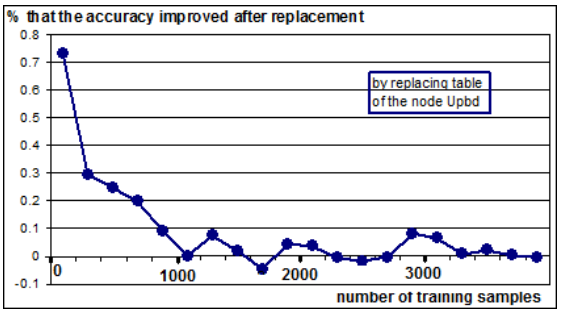

Fig. 7. Accuracy against size of training set for artificial data.

Now we want to show that the replacement of conditional probability distributions, which are in the form of tables, makes significantly better classifiers. For this purpose we run several experiments as explained in the experiment setup subsection. For each individual replacement, for example changing table Chest Pain (*Chestp*), the accuracies of different experiments on the modified model are compared with the unmodified ones, using paired t-test, to find out whether the modified model is significantly better than the unmodified one or not. Table 2 contains the results for different distribution replacements. For each replacement, we mentioned whether the result is significantly better or not. These results are obtained using the paired t-test with 95% confidence interval. This table contains the results of replacing just one conditional probability distribution in each test. Each column has a header which indicates the name of the attribute whose conditional distribution is replaced. The modified models in six different cases out of eight are extremely statistically significant. In one case, it is statistically significant. In another case, when the conditional dependence of the sleep disturbance attribute is removed, it is not statistically significant. Incomplete dependencies of this attribute on others were intentionally used here to show what the effect of using incomplete or wrong dependencies would be.

The results of these experiments again show that the variance in the accuracy of the modified model is smaller than the variance in the accuracy of the unmodified model.

5 Discussion and Future Work

Replacement of conditional probability distributions of attributes which are extracted according to wrong or incomplete dependencies or have a weak relation

Table 2. Accuracy for different replacements in the Bayesian Network on the heart attack data set as well as the results of the t-test for 20 different experiments.

experiment	**no change**	**Chstp**	**Upbd**	**Shrtb**	**Swetg**	**Dizns**	**Nasa**	**Slepd**	**Wkns**
Average of the accuracy	89.5	89.9	90.4	90.2	90.1	90.2	89.9	89.6	90.4
$Varianceofaccuracy$	0.445	0.945	0.6625	0.64	0.725	0.45	0.1025	0.925	
$T-Testresult:$		**SS**	**ESS**	**ESS**	**ESS**	**ESS**	**ESS**	**NSS**	**ESS**

with the class, may impact negatively on the classification result. Suppose that we replace such a distribution with an accurate alternative and that we use the replaced distribution along with other attribute distributions to classify a given instance. If, using accurate conditional probability distribution, the two conditional probabilities of belonging to each class are close to each other, the effect of replacing the distribution on the joint probabilities is weak. But when conditional probabilities obtained from replaced distributions of such faulty attributes are far apart, they have a larger impact on the Bayesian classification results, and since the conditional distributions involving these attributes are incorrect, there is a negative impact on the results.

One solution for such conditional probability distributions, which we propose as future work, could be to assign a weight for each attribute based on its real effect on classification. Then, during classification, the difference between the values which belong to each attribute's conditional probability distribution could be smoothened based on the weight of the attribute which it belongs to. A similar approach has been used to improve the accuracy of Naive Bayes by weakening its attribute independence assumptions in Lazy Bayesian Rules [20], Tree Augmented Naive Bayes [15] and Averaged One-Dependence Estimators [18]. If we consider a small weight for problematic attributes of this kind, their effect on classification results would be reduced and therefore better results would be obtained. This solution would require a good strategy to measure these weights.

Another result that we experienced in these tests was that the variance of the accuracy of any modified classifier is smaller than the variance of unmodified model. This means that the modified classifiers tend to be more robust with respect to learning and testing with different data sets sampled from the same domain. Testing the result of the modified classifiers on different types of drifts in the data sets and finding which of the modified and unmodified models are more robust in case of different types of changes in the data, such as concept drifts or population drifts, is also proposed as future work.

6 Conclusion

In this study we propose a practical method for improving Bayesian classifiers by using background knowledge from large, publicly available datasets existing

independently of the training data set. We present a method which manipulates the Bayesian Network's conditional probability distributions, given in the form of tables, based on background knowledge. The idea is tested on a real and an artificial data set. The results show that such changes produce significantly better classifiers than normal Bayesian Network classifiers.

References

1. European Commission. http://epp.eurostat.ec.europa.eu/portal/page/portal/eurostat/home/.
2. Statistics Canada. http://www.statcan.gc.ca/start-debut-eng.html.
3. UK National Statistics. http://www.statistics.gov.uk/hub/index.html.
4. United Nations. http://www.un.org/en/development/progareas/statistics.shtml.
5. USA Government. http://www.usa.gov/Topics/Reference_Shelf/Data.shtml.
6. World Health Organization. http://www.who.int/whosis/whostat/en/.
7. S. Acid, L. M. de Campos, J. M. Fernndez-Luna, S. Rodrguez, J. M. Rodrguez, and J. L. Salcedo. A comparison of learning algorithms for bayesian networks: a case study based on data from an emergency medical service. *Artificial Intelligence in Medicine*, (30):215–232, 2004.
8. C. Blake and C. Merz. UCI repository of machine learning databases. Univ. of California at Irvine. http://www.ics.uci.edu/~mlearn/MLRepository.html.
9. J. Cheng. BN powerpredictor. http://webdocs.cs.ualberta.ca/~jcheng/bnsoft.htm.
10. J. Cheng and R. Greiner. Learning bayesian belief network classifiers: algorithms and system. *LNCS*, 2056:141–151, 2001.
11. P. Clark and S. Matwin. Learning domain theories using abstract beckground knowledge. In P. Brazdil, editor, *ECML*, volume 667 of *Lecture Notes in Computer Science*, pages 360–365. Springer, 1993.
12. T. G. Dietterich. Machine learning research: Four current directions. *The AI Magazine*, 18(4):97–136, 1998.
13. P. A. Flach. From extensional to intensional knowledge: Inductive logic programming techniques and their application to deductive databases. *Transactions and Change in Logic Databases, LNCS*, 1472:356–387, 1998.
14. P. W. Frey and D. J. Slate. Letter recognition using Holland-style adaptive classifiers. *Machine Learning*, 6(2):161–182, 1991.
15. N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine learning*, 29(2):131–163, 1997.
16. D. Heckerman. A tutorial on learning with bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, 1995.
17. T. M. Mitchell. *Machine learning.* McGraw Hill, 1997.
18. G. I. Webb, J. R. Boughton, and Z. Wang. Not so naive bayes: Aggregating one-dependence estimators. *Machine learning*, 58(1):5–24, 2005.
19. P. Wu and T. G. Dietterich. Improving svm accuracy by training on auxiliary data sources. In C. E. Brodley, editor, *ICML*, volume 69 of *ACM International Conference Proceeding Series*. ACM, 2004.
20. Z. Zheng and G. I. Webb. Lazy learning of bayesian rules. *Machine learning*, 41(1):53–84, 2000.