Artificial Intelligence Research and Development
I. Sanz et al. (Eds.)

doi:10.3233/FAIA230671

A Bayesian Network Framework to Study Class Noise: Exploring the Filtering of Completely Random Noise

David MARTÍNEZ-GALICIA [a,1], Alejandro GUERRA-HERNÁNDEZ [a], Xavier LIMÓN [b], Nicandro CRUZ-RAMÍREZ [a] and Francisco GRIMALDO [c]

[a] *Inst. de Invest. en Inteligencia Artificial, Universidad Veracruzana, Xalapa, México*
[b] *Facultad de Estadística e Informática, Universidad Veracruzana, Xalapa, México*
[c] *Departament d'Informàtica, Universitat de València, València, España*

Abstract. Although the negative consequences of noise during induction have been widely studied, previous work often lacks the use of validated data to measure its impact. We propose a framework based on Bayesian Networks for modeling class noise and generating synthetic data sets where the kind and amount of class noise are under control. The benefits of the proposed approach are illustrated evaluating the filtering of noise completely at random in class labels when inducing decision trees. Unexpectedly, this kind of noise showed a low effect on accuracy and a low occurrence on real datasets. The framework and the methodology developed here seem promising to study other kinds of noise in class labels.

Keywords. Noise modeling, Bayesian Networks, Data generation, Noise filtering

1. Introduction

In supervised learning, noise refers to anything that obscures the relationship between attributes and class labels [1]. However, previous studies often use not validated data to measure its impact. A common approach while studying class corruption is to assume that datasets from repositories are clean, which may produced biased results [2]. We propose a Bayesian Network (BN) framework to model domains affected by class noise and provide a controlled experimental setting. We illustrate its benefits with a case study evaluating the effect of filtering completely random noise when inducing decision trees.

2. Modeling Class Noise

Frénay and Verleysen [3] pioneered the probabilistic approach to model class noise. Their modeling represents complex relationships in noisy scenarios using a domain distribution and a noise mechanism. However, it requires a complete description, which can cause problems when the parameters representing the full joint distribution of the

[1] Corresponding Author: Martínez-Galicia David, davidgalicia@outlook.es

domain grow exponentially. We propose a framework that extends their approach using BNs to reduce the number of parameters and generate experimental data with sampling algorithms. Three models arise considering certain statistical relationships, as shown in Figure 1.

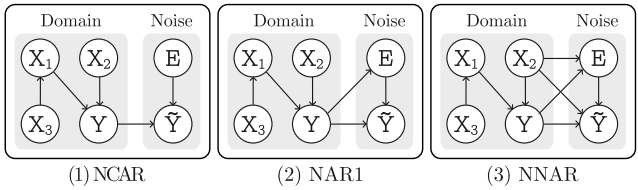

Figure 1. Models of class noise where X_i denotes the attributes, Y denotes the true class, E is a binary variable symbolizing the probability of a labeling error, and $\tilde{Y}$ denotes the observed class. *(1) Noisy Completely at Random:* errors are independent of other domain variables. *(2) Noisy at Random:* errors depends on the true class. *(3) Noisy Not at Random:* errors depends on a set of attributes and the true class.

Building the domain distribution is not trivial and requires prior knowledge. We recommend using reliable sources such as curated data or a domain expert to avoid spurious noise. Other suggestions for generating reliable data are: 1) defining the noise level to be less than 50% and, 2) in case the model was learned from curated data, restricting the generation of synthetic examples to those appearing in the original dataset.

3. Case Study

We illustrate the benefits of adopting our approach with a case study that evaluates the efficiency of a new filter for class noise completely random. Class noise poses a challenge to estimate the reliability of class labels and it has the most detrimental effect on learning [4]. The Inconsistency Deletion Filter (IDF) identifies sets of examples with the same attribute values but different class and keeps only the ones belonging to the majority class, which are arguably the most likely to be clean. The case study has two parts:

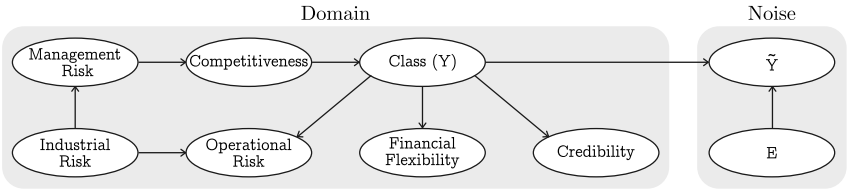

Figure 2. NCAR model for the QB domain.

1. Evaluating IDF on synthetic data. We define an NCAR model using an apparently clean domain, the Qualitative Bankruptcy (QB) dataset [5], see Figure 2. The model is fed with noise levels from 10% to 40% with increments of 10%. For each level, four datasets are generated with the following number of examples: 500, 2000, 5000, and 10000. A 10-fold cross-validation process is performed to obtain pairs of train and test sets. To evaluate the efficiency of IDF, three metrics are considered: the ratio of clean examples removed (ER1), the ratio of noisy examples not removed (ER2), and the per-

centage of conserved examples (CE) after filtering. ER1 and ER2 are known as errors in data cleansing [6]. To measure the impact on learning, decision trees are induced using J48 (with default parameters) [7] and the area under the ROC curve (AUC) is measured over clean and noisy test sets. Clean test sets are obtained by $P(E = True) = 0$, while noisy test sets are produced by a cross-validation process over noisy data.

2. Evaluating IDF on UCI data. We adopt seven UCI datasets, see Table 1. For continuous data, five methods from the discretization R package [8] were adopted: AMECA, CACC, CAIM, CHI2, and MDLP. Using various discretizers avoids their individual limitations for certain types of data or models and allows the description of their average performance. The methodology previously introduced is modified since the noisy examples in UCI datasets are unknown: 1) we use one type of test set obtained by a 10-folds cross-validation, which might be noisy; 2) ER1 and ER2 are not measured.

Table 1. Characteristics of the adopted datasets.

Dataset	Solar Flare	Nursery	Balance Scale	Breast Cancer	Diabetes	Ecoli	Semgent
Type	Disc.	Disc.	Disc.	Cont.	Cont.	Cont.	Cont.
Attributes	13	9	4	9	9	8	20
Class labels	5	5	3	2	2	8	7
Examples	323	12960	625	683	768	336	2310

4. Results and Discussion

Table 2 summarizes the efficiency of IDF on the synthetic datasets generated by our framework. Values of ER1 and ER2 closer to zero denote the best outcome, most of the clean data is conserved while noise is removed. Our filter seems to be quite effective in synthetic datasets with equal or less than 20% of noise. Increasing the noise level reduces its efficiency. However, this effect is mitigated in larger datasets. The remaining noisy examples in the filtered training set suggest that inconsistencies are a subset of noisy examples, i.e., there are noisy examples that do not produce inconsistent sets. Regarding the percentage of remaining examples, CE suggest that the proportion of deleted examples is closer to the noise level of a dataset, which is consistent with the low values of errors in data cleansing.

Table 2. Mean values of ER1, ER2 and CE. The size of the training sets represent 90% of the generated data.

Noise	10%			20%			30%			40%		
Size	ER1	ER2	CE	ER1	ER2	CE	ER1	ER2	CE	ER1	ER2	CE
450	1.6%	1.3%	89.9%	2.5%	2.2%	79.1%	7.0%	6.2%	66.6%	12.7%	12.3%	59.8%
1800	0.1%	0.0%	89.4%	0.4%	0.1%	79.3%	1.3%	0.9%	69.6%	6.8%	7.6%	60.2%
4500	0.0%	0.0%	89.6%	0.1%	0.0%	79.9%	1.1%	1.0%	70.1%	6.7%	5.6%	58.0%
9000	0.0%	0.0%	89.8%	0.0%	0.0%	79.9%	0.1%	0.0%	69.6%	1.6%	1.5%	59.4%

Accuracy results are not presented because of space limitations. However, in most cases, IDF preserves the levels of AUC. Experiments were performed on clean and noisy test sets. When facing clean test sets the induced decision trees seem robust to noise (AUCs between 1.0 and 0.97), perhaps because of the low complexity of the QB domain. Noisy tests sets exhibits a linear degradation, e.g., AUCs near 0.9 for 10% noise, decreasing to 0.6 in tests with 40% noise.

Table 3 shows the results over UCI datasets. Even though the kind and amount of noise affecting these datasets are unknown, it is possible to make assumptions about their noise type. Balance Scale and Nursery do not seem to include NCAR given the lack of data reductions. However, Balance Scale could probably have other types of noise that affect the performance of induced decision trees. Poor reductions and high AUCs on Breast Cancer and Segment suggest a low percentage of noise, possibly NCAR. Finally, Solar Flare, Diabetes, and Ecoli seem to include NCAR since our filter reduces up to 15% of data. Indeed, these observations imply NCAR noise is not frequent in UCI data.

Table 3. Mean values of conserved examples and AUC on UCI datasets. For continuous sets, results are calculated as an average of all discretized versions. J48 stands for the case where no filter was applied.

	CE	AUC			CE	AUC	
Data	IDF	J48	IDF	Data	IDF	J48	IDF
Balance Scale	100.0%	68.5	68.5	Breast Cancer	98.8%	96.5	96.4
Solar Flare	85.9%	71.9	71.5	Diabetes	84.4%	75.0	70.8
Nursery	100.0%	99.5	99.5	Ecoli	93.3%	79.4	78.5
				Segment	99.8%	93.1	93.1

5. Conclusions and Future Work

Our BN framework allows the compressed description of noisy domains and provides tools to build a controlled experimental setting. These benefits help to assess the efficiency of thetechniques proposed to cope with noise since the kind and amount of noisy examples are known. Our case study shows the significant filtering achieved by IDF, although it does not improve the accuracy of decision trees. These observations are relevant because they contradict some results from the literature. First, according to our experiments, class noise seems not as harmful as suggested when assessed on clean test sets. Second, filtering synthetic data does not lead to better predictive models, perhaps because of the low complexity of the QB domain. The last observation, derived from our experiments on real datasets, is that completely random class noise does not seem to be usual in the UCI repository. These observations would not be possible without using our framework. Future work includes investigating other types of class noise (NAR1, NAR2, and NNAR) and exploring the application of IDF to inductive algorithms that do not use noise-handling methods, e.g., naive Bayes.

References

[1] Hickey, R. Noise Modelling and Evaluating Learning from Examples. *Artif. Intell.*. **82** (1996).
[2] Zhu, X. & Wu, X. Class Noise vs. Attribute Noise: A Quantitative Study. *Artif. Intell. Rev.*. **22** (2004).
[3] Frénay, B. & Verleysen, M. Classification in the Presence of Label Noise: A Survey. *IEEE Trans. Neural Networks Learn. Syst.* **25** (2014).
[4] Quinlan, J. Learning from noisy data. *Proc. 1983 Int. Machine Learning Workshop*. (1983).
[5] Kim, M. & Han, I. The discovery of experts' decision rules from qualitative bankruptcy data using genetic algorithms. *Expert Syst. Appl.*. **25** (2003).
[6] Brodley, C. & Friedl, M. Identifying Mislabeled Training Data. *CoRR*. (2011)
[7] Witten, I., Frank, E. & Hall, M. Data mining: practical machine learning tools and techniques, 3rd Ed. *Morgan Kaufmann, Elsevier* (2011).
[8] Kim, H. discretization: Data Preprocessing, Discretization for Classification. (2022), R package v. 1.1.