MASS ESTIMATION BASED ON GENERATIVE CLASSIFIER

Alwin Pinakas J.
Head, Department of Electronics and Computer Systems,KG College of Arts and Science

Abstract:The aim to build generative classifiers is to estimate the joint probability p(x, y) indirectly, estimating the conditional likelihood p(x|y) and the prior probability p(y).To predict the likely class that maximizes the posterior probability p(y|x) using Bayes rule. The paper propose a generative classifier which estimates the joint distribution directly through a data modelling mechanism called mass estimation. The generative classifier makes prediction based on decision rule that maximizes mass, better than Bayes rule.

Index Terms - Mass distribution, Mass Estimation, Generative classifier.

I. INTRODUCTION

Classification is a data mining task that deals with assigning data instances described by a set of variables (x) to one of the predefined mutually exclusive categories (y).

Discriminative and generative classifiers are two distinct approaches to solve classification problems [10, 13]. Generative classifiers model the joint probability p(x, y) via Bayes rule. Discriminative classifiers, on the other hand, learn a direct mapping from x to y [10]. Classifers such as Naive Bayes (NB), Bayesian Belief Network (BayesNet), Aggregating One Dependence Estimators (AODE) are examples of generative classifiers; whereas, Artificial Neural Networks (ANN), Linear Logistic Regression (LLR), Support VectorMachines (SVM) are examples of discriminative classifiers. Building generative models require density estimators. Current density estimators such as kernel density estimator and k-nearest neighbour density estimator have a high time and space complexities. Thus, it is difficult to estimate p(x, y) directly to build generative models even with data sets that have a moderate number of dimensions and moderate data size.

Instead, the current generative approach focuses on estimating p(x|y) and p(y), and makes the final decision via Bayes rule. This approach encounters the same limitation of existing density estimators: p(x|y) cannot be estimated directly. However, surrogates of p(x|y) can be estimated efficiently provided some assumptions are made (e.g., attribute independence given the class.) Though this type of generative classifiers has been shown to perform well [7, 12, 8], the assumptions made are often violated in practice and can result in poor predictive accuracy.

Mass estimation [17, 16, 15] provides an alternative to density estimation for data modelling and it has been shown to work well in anomaly detection, information retrieval, clustering and regression. This paper is motivated to employ mass estimation to solve classification problems, in particular, by estimating joint distribution directly to build generative models. This is a more direct approach than the current approach to build generative models.

We propose a new type of generative classifier called MassCfier that exploits the notion of mass and mass distribution to estimate the joint distribution effectively. MassCfier has three distinctive characteristics compared to existing generative classifiers:

1. The joint distribution is estimated directly without estimating the likelihood p(x|y) and the prior probability p(y).
2. Its prediction decision is based on a maximum mass rule rather than Bayes rule.
3. It has sub-linear time complexity and constant space complexity; therefore, it scales better for very large database.

II. EXISTING GENERATIVE CLASSIFIERS

The existing generative classifiers estimate the conditional likelihood p(x|y) and the prior p(y) and use Bayes rule to make the final prediction.

$$y = arg_y^{max}\big(p(x|y)x\ p(y)\big) \qquad Equ(1)$$

Different generative classifiers estimate the conditional likelihood p(x|y) in different ways. We briefly describe three existing generative classifiers in this section.

2.1. Naive Bayes

Naive Bayes (NB) [2, 7] assumes class conditional independence and estimates the likelihood on each dimension separately. A typical structure of Naive Bayes is given in figure

The likelihood of x given class y is estimated as follows.

$$p(x|y) = \prod_{i=1}^{d} p(x_i|y) \qquad Equ\ (2)$$

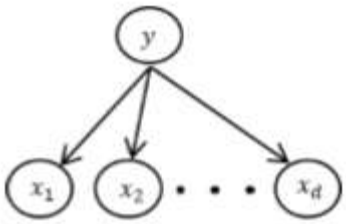

Figure 1: Structure of Naive Bayesian classifier where predictive attributes ($x_1, x_2, \ldots, x_d$) are conditionally independent given the class attribute y.

2.2. Bayesian Networks

Bayesian Networks (BayesNet) [5, 11] learns probabilistic relationships among the attributes including the class in the form of directed acyclic graph (DAG) from the training data. In a graph, edges represent conditional dependencies and nodes, which are not connected, are conditionally independent. At each node, conditional probabilities with respect to its parents are learned from the training data.

In the version of Bayesian Networks we used, continuous valued attributes are discretised.

Aggregating One-Dependence Estimator

Aggregating One-Dependence Estimators (AODE) [18] allows conditional dependence with one 'privileged' attribute.Other attributes are conditionally independent given class label y. The conditional probability, with a privileged attribute xi, is computed as follows.

$$p(x|x_i,y) = \prod_{j=1}^{d} p(x_i|x_i,y) \qquad Equ\ (3)$$

As AODE is designed for discrete attributes, continuousvalued attributes are discretised. The conditional probability is computed as relative frequencies as in NB-Disc.

Each attribute gets a chance to be a privilege attribute once; hence, AODE builds d models and aggregates the decisions to make the final prediction.

III. MASS AND MASS-ESTIMATION

Ting et al [17] introduced the fundamental concept of mass as a base measure. The application of mass to solve various data mining problems such as regression, information retrieval, clustering, anomaly detection, and data stream are demonstrated in [17, 16, 14, 15]. Mass-based data mining methods often performed better than or at least as well as the state-of-the-art methods. The key advantages of mass- based methods are as follows:

1. Employ no distance measures and generally run faster.
2. Have average case sub linear time complexity and constant space complexity; hence, it can be applied to very large data sets.

In its simplest form, mass is the number of data instances in a bounded region. A mass base function is defined as follows

$$m(T(x)) = \begin{cases} m & \text{if } x \text{ is a region of } T(.) \\ 0 & \text{otherwise} \end{cases} \quad Equ\ (4)$$

where, T(•) is a function that subdivides the feature space of the given data set D into non-overlapping regions; and, m is the number of instances in a region of T(x) in which x falls into.

The estimated mass for an instance x is defined [16] as

$$mass(x) = \frac{1}{t}\sum_{i=1}^{t} m(T_i(x)) \quad Equ(5)$$

Mass estimation has been shown to be a good data modelling mechanism in [17, 16, 15]. Figure 4 shows the estimation of two overlapped clusters in one dimensional feature space using kernel density estimation (KDE) and mass estimation. It demonstrates that mass-based estimation is comparable to that of KDE.

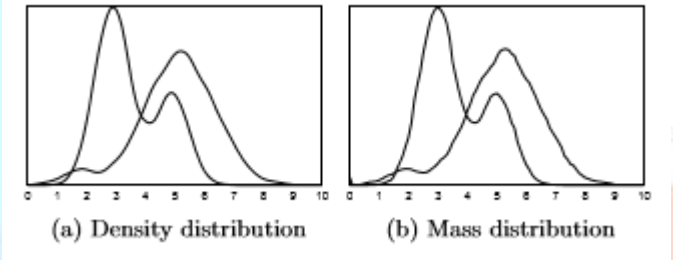

Figure 4: Density estimation of KDE vs. Mass estimation. Y-axis: a) density, b) mass. The parameters used for mass estimation are t = 100, 0 = 4096 and h = 5. These parameters are discussed in the following section. Parameter h for mass is equivalent to the bandwidth smoothing parameter for KDE. The bandwidth parameter was automatically selected in the case of KDE.

IV. GENARATIVE CLASSIFIER

MassCfier is a generative classifier that exploits the notion of mass and mass distribution. It estimates the mass joint distribution of x and y. The corresponding mass base function m(T(x), y) is defined as the count of instances in a region of T (x) that belong to class y.

$$m(T(x), y) = \begin{cases} m_y & \text{if } x \text{ is a region of } T(.) \\ 0 & \text{otherwise} \end{cases} \quad Equ\ (6)$$

where, m_y is the number of instances belonging to class y in a region of T(x).

The decision rules of existing generative classifiers are provided in Table 1.

Table 1: Decision rules of different existing generative classifiers and MassCfier.

Classifier	Decision Rule
Bayes Ne	$\arg\max_y \left(\prod_i^d p(\pi_i, y)p(x_i\|\pi_i, y)\right)$
AODE	$\arg\max_y \left(\prod_{i=1}^d p(\pi_i, y)p(x_i\|\pi_i, y)\right)$
MassCfier	$\arg\max_y (mass(x, y))$

Once the data distribution has been modelled using mass distribution, a simple decision rule based on maximum mass can be used to make a prediction in the classification context.

V. EXPERIMENTS

In this section, we compare the performance of MassCfier with existing generative classifiers Naive Bayes with density estimation through Gaussian Distribution assumption (NB-GD), Naive Bayes with Kernel Density Estimator (NB- KDE), Naive Bayes with Discretisation (NB-Disc), Bayesian Networks (BayesNet), and Aggregating One-dependence Estimators (AODE).

We have implemented the proposed method using the WEKA platform [6, 19] which has all of the existing generative classifiers. The data sets used are from UCI Machine Learning Repository [4] unless stated otherwise.

All the experiments were conducted as single thread jobs processed at 2.27 GHz on a Linux cluster using a node with 40 GB memory.

All the algorithms were executed with default parameters except BayesNet. For BayesNet, the parameter 'max number of parents' was set to 100 to enforce no restriction on the number of parents that a node can have in the network; and the parameter 'initialise as Naive Bayes' was set to 'false' to initialise an empty network structure. The rest of the parameters were set to defaults. The default settings for MassCfier were t = 100, 0 = 4096 and h = |"log2(0)"|. Where there are less than 4096 instances in each class, the entire data set were used to construct the trees.

We compared the performance of proposed methods with the existing generative classifiers on 18 data sets with different sizes, dimensions, number of classes and class distributions. The properties of the data sets are provided in Table 2.

Table 2: Data sets used to compare the performance of MassCfier with other existing generative classifiers.

Data set	datasize	^dimensions	#classes
CoverType	581012	10	7
MiniBooNE	129596	50	2
OneBig	68000	20	10
Shuttle	58000	8	7
Wave	20000	2	2
RingCurve	20000	2	2
Letters	20000	16	26
Magic04	19020	10	2
Mammography	11183	6	2
Pendigits	10992	16	10
Wine	6497	11	2
Satellite	6435	36	7
OpticalDigits	5620	62	10
PageBlocks	5473	10	5
RobotNavigation	5456	24	4
Waveform	5000	21	3
ImageSegments	2310	19	7
SteelPlateFaults	1941	25	7

Out of 18 data sets used, OneBig, Wave and RingCurve are synthetic and the rest are real data sets. Wave and RingCurve are two dimensional data sets, which are subsets of RingCurve-Wave-TriGaussian data set, shown in Appendix A, each having two classes with 10000 data instances in each class. The OneBig data set [9] has 20 attributes, 9 clusters and 10000 noise instances randomly distributed in the feature space. Noise in the data set are treated as a separate class; hence, it has 10 classes.

Overall Comparison, Classification Accuracy, The experimental results, in terms of classification accuracies, are show in Table 3. Compared with existing generative classifiers, the result showed that MassCfier yielded better or at least competitive classification accuracies in most of the data sets. A statistical test based on two standard errors was performed to examine whether the difference is significant. The win:loss:draw counts of MassCfier over existing generative classifiers are reported in Table 4. A win or loss is counted if the difference is significant; otherwise, it is a draw.

Table 3: Classification accuracies (%) on different data sets over a 10-fold cross validation for MassCfier and existing generative classifiers: AODE, BayesNet, NB-KDE, NB-GD and NB-Disc. Figures marked with * and 1 represent significant win and loss respectively, of MassCfier with respect to AODE based on a two-standard-error significance test.

Data Set	Mass	AO	Bayes
CoverType	79.16*	72.89	87.54
MiniBooNE	90.90*	89.58	90.19
OneBig	100*	99.69	99.99
Shuttle	99.89*	99.85	99.92
Wave	99.99*	78.50	78.27
RingCurve	100*	99.98	99.96
Letters	96.67*	88.81	86.76
Magic04	84.58*	83.00	83.36
Mammography	98.59	98.42	98.48
Pendigits	99.45*	97.84	96.56
Wine	99.32	99.29	99.20
Satellite	91.41*	89.26	83.29
OpticalDigits	98.40*	97.03	96.16
PageBlocks	96.311	97.37	96.25
RobotNavigation	91.511	94.13	94.85
Waveform	84.481	86.48	82.32
ImageSegments	96.97*	95.76	95.50
SteelPlateFaults	74.19	75.32	74.03
Avg. Accuracy	93.43	91.29	91.26

MassCfier had 12 wins, 3 losses and 3 draws when compared to AODE, and 11 wins, 2 losses and 5 draws in comparison to BayesNet. Similarly, it had 18 wins over NB-KDE and NB-GD; and 17 wins and 1 draw over NB-Disc.

VI. CONCLUSION

In this research, we proposed a new type of generative classifier exploiting the notion of mass called MassCfier. Unlike existing generative classifiers based on Bayes rule, MassCfier has the following distinctive characteristics.

1. MassCfier estimates the joint distribution directly in multi-dimensional space
2. MassCfier utilises a new decision rule based on maximum mass rather than Bayes rule.

It has sub-linear time complexity and constant space complexity.

Empirical results show that MassCfier is better or at least competitive in terms of classification accuracy when compared to the existing generative classifiers. MassCfier empowers generative classifiers to be more powerful and flexible with no assumption and improved time complexity.

One direction for future work is to explore a non-grid based

implementation for mass estimation that eliminates the weaknesses of grid based implementation to deal with high-dimensional problems.

.REFERENCES

[1.] J. Catlett. On changing continuous attributes into ordered discrete attributes. In Proceedings of the European Working Session on Learning, Porto, Portugal, pages 164-178, 1991.
[2.] R. O. Duda and P. E. Hart. Pattern Classification and, Scene Analysis. New York: Wiley, 1973.
[3.] U. M. Fayyad and K. B. Irani. Multi-interval discretization of continuous valued attributes for classification learning. In Proceedings of 14th International Joint Conference on Artificial Intelligence, pages 1034-1040, 1995.
[4.] Frank and A. Asuncion. UCI machine learning repository. http://archive.ics.uci.edu/ml, 2010. University of California, Irvine, School of Information and Computer Sciences.
[5.] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. Machine Learning, 29:131-163, 1997.M. Hall, E. Frank, G. Holmes, B. Pfahringer,
[6.] P. Reutemann, and I. H. Witten. The weka data mining software: An update. SIGKDD Explorations, 2009. Volume 11, Issue 1.
[7.] P. Langley, W. Iba, and K. Thompson. An analysis of bayesian classifiers. In Proceedings of the Tenth National Conference on Artificial Intelligence, pages 399-406,1992.

[8.] P. Langley and G. H. John. Estimating continuous distribution in bayesian classifiers. In Proceedings of Eleventh conference on uncertainty in artificial intelligence, 1995.

[9.] Nanopoulos, Y. Theodoridis, and Y. Manolopoulos. Indexed-based density biased sampling for clustering applications. IEEE Transaction on Data and Knowledge Engineering, 57(1):37-63, 2006.

[10.] Y. Ng and M. I. Jordan. On discriminative vs. generative classffiers: A comparison of logistic regression and naive bayes. In NIPS, pages 841-848, 2002.

[11.] J. Pearl. Bayesian networks: A model of self-activated memory for evedential reasoning. In Proceedings of the Seventh Annual Conference of the Cognitive Science Society, pages 329-334, 1885.

[12.] Rish. An empirical study of the naive bayes classifier. In IJCAI Workshop on Empirical Methods in Artificial Intelligence, 2001.

[13.] Y. D. Rubinstein and T. Hastie. Discriminative vs informative learning. In Proceedings of the Third International Conference on Knowledge and Data Mining, pages 461-464, 1997.

[14.] S. C. Tan, K. M. Ting, and F. T. Liu. Fast anamoly detection for streaming data. In Proceedings of IJCAI, pages 1151-1156, 2011.

[15.] M. Ting, T. Washio, J. R. Wells, and T. Liu. Density estimation based on mass. In Proceedings of IEEE International Conference on Data Mining, pages 715-724, 2011.

[16.] M. Ting and J. R. Wells. Multi-dimensional mass estimation and mass-based clustering. In Proceedings of IEEE ICDM, pages 511-520, 2010.

[17.] M. Ting, G.-T. Zhou, F. T. Liu, and S. C. Tan. Mass estimation and its applications. In Proceedings of ACM SIGKDD, pages 989-998, 2010.

[18.] G. I. Webb, J. R. Boughton, and Z. Wang. Aggregating one-dependence estimators. Machine Learning, 58:5-24, 2005.

[19.] H. Witten and E. Frank. Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Second Edition. Morgan Kaufmann, 2005.