International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Survey on Gaussian Estimation Based Decision Trees for Data Streams Mining

Poonam M Jagdale[1], Devendra P Gadekar[2]

[1]Pune University, Pune, Maharashtra, India
[2]Assistant Professor, Pune University, Pune, Maharashtra, India

Abstract: ***In the literature, Hoeffding tree algorithm was projected, for mining data streams decision trees became one of the most popular tools. Determine the best attribute to split the considered node is the key point of constructing the decision tree. Existing system presented the numerous methods to solve this problem. But they are either time consuming such as, in the MacDiarmid tree algorithm or justified wrongly by mathematically such as, in the Hoeffding tree algorithm. The selection of best attribute in the considered node with the help of finite data sample is similar as it would be in the case of the entire data stream with the high probability set by the user is make sure by the this method. In this paper we are presenting some efficient research approaches suggested by numerous scholars.***

Keywords: Decision tree, Data stream, Gaussian approximation, Information gain.

1. Introduction

Now days, in the data mining community mining data stream become an extremely demanding task [1-8]. Data stream is of infinite size, it is different the static data set. In the system, data elements arrive incessantly with high rates. Furthermore, in time the idea of data can developed which is also called as concept drift [9-13]. Due to this reason data streams cannot be applied directly by data mining algorithms. In this paper we discuss one of the data mining techniques that are classification task [14-18]. Labeling the unclassified data and learning from the training data set are the two steps calm in classification procedure. The n numbers of elements s_j, j = 1, ….., n are present in the training data set S which is characterized by D attributes a^1,…, a^D. As well, each data element is assigned by one of the K classes. The values from the corresponding set A^i are taken by the each attribute ai, $i \in \{1, \dots, D\}$. Hence, the training data element s_j can be articulated in the form

$s_j = ([v_j^1, \dots, v_j^D], k_j)$, $v^i_j \in A^I$, $k_j \in \{1, \dots, K\}$, (1)

Where v^i_j is denoted as a value of attribute a^i for data element s_j. To construct the classifier which is used to labels the unclassified data elements is done by using the training data set.

In CVFDT ALGORITHM FOR MINING OF DATA STREAMS, system implementation is based on the decision tree of CVFDT Algorithm to address the inequalities projected in stream mining. The planned work also uses the network data streams to examine the attacks using splitting attribute with gain values. The builder tree can be used for classification of new observation. It gives better performance than the Hoeffding trees.

The CART Decision Tree for Mining Data Streams proposes a new algorithm, which is based on the commonly known CART algorithm. The most important task in constructing decision trees for data streams is to determine the best attribute to make a split in the considered node. To solve this problem they apply the Gaussian approximation. The presented algorithm allows obtaining high accuracy of classification, with a short processing time. The main result of this paper is the theorem showing that the best attribute computed in considered node according to the available data sample is the same, with some high probability, as the attribute derived from the whole data stream.

Here, this paper projected a classification method multitude for the static data like decision trees [14], k-nearest neighbors [6], [23] or neural networks [14]. In this paper the most effective classifier based on decision trees discussed. In the decision tree contain nodes, branches and leaves which are used for taking the decision. The nodes which are not end nodes or not terminal have some attribute a^i. There are two types of tree binary or nonbinary. The decision tree may be binary or nonbinary. In binary tree nodes split into two children nodes and in nonbinary tree the node has number of children's so there is number of elements of set A^i. With the help of branches the children nodes and parent nodes connected to the each other. In the binary tree, value of some subset of A^i is assigned to the each branch and in the nonbnary tree; the attribute a^i is allocated to the every branch and it make sense when the attribute get nominal values. For the growth of tree, the training set is divided into subsets on the basis of attribute values allocated to the branches and it is propel towards the equivalent children nodes. Leaves are also called as end nodes are terminal nodes which are used to detect the decision in decision trees. It is also used to allocate a class to the unclassified data elements. Selecting the most excellent attribute to split the considered node is a key point in building the decision tree. In the majority of the projected algorithm, the selection is based on some contamination gauge of the data set. The impurity of the data set before the split and weighted impurity of the resulting subsets are calculated for all the probable dividers of the node. Split measure function is the difference of these values. Considered node is assigned by the best attribute which is nothing but an attribute which gives the highest value of this function. Impurity measure is taken as information entropy in the ID3 algorithm. The matching split-measure function is also called as entropy reduction or the information gain. The ID3 algorithm supports the attributes with big domain of probable values in the case of non-binary trees. It is the main disadvantage of

the ID3 algorithm. This problem is solved by using C4.5 algorithm [25].

The main goal of the C4.5 algorithm is to bring in the split information function which punishes the attributes with large domains. In the C4.5 algorithm split-measure function projected as a ratio of the information gain and the split information. The Gini index is one more impurity measure value the consideration which is used in the CART algorithm [5]. The CART algorithm planned to developed binary tress. That's why it is applied to the data with nominal attribute values and also the numerical data values.

The above algorithms cannot be applied directly to the data streams and it is intended for static data set. The stream is of infinite size due to this reason establishment of the best attribute in each node is the dominant problem. To be familiar with if the best attribute calculated from this data set is also the best attribute for the whole data stream, with some fixed probability $1-\delta$. Referring to the two papers which constitute the "state of the art" in this subjects (see Section 2), the major and original result of this paper can be summarized as follows:

- Based on MacDiarmid's inequality [22], Rutkowski et al. [24], recently projected a method. Very large number of data elements n in the considered node needs for the selection of best attributes. To decrease the value of n radically contrasting with [24], with the similar prospect $1-\delta$, the projected method sustained the theorem 1. From the application of the substitute mathematical tool, this important dissimilarity occurs. They permissible to get much improved results.
- Based on the multivariate delta method, one more method is projected [18]. Though the idea was promising, the result was wrong and not appropriate to the problem of building decision trees for data streams. To determine the best attribute in a node which ensure the highest value of the split-measure function with significantly high probability is done by statistical method is projected in this paper. The properties of the normal distribution and Taylor's theorem are also used in this method [19].

2. Background

A. ID3 Algorithm

The ID3 algorithm is a background of our method. To create nonbinary trees, it is at first projected but it is easily distorted into the binary mode. These methods adopted by the binary as well as nonbinary trees but we focus on the binary case for the requirement of this paper. This algorithm initiate with single node that is root node. Exacting subset of the training data set is procedure in each created node throughout the learning process. The node is tagged as a leaf and the split is not made if the all elements of the set are of the similar class or select the best attribute to split amongst the obtainable attributes in the considered node. The set of attribute values A^i is divided into two disjoint subsets A^i_L and A^i_R ($A^i = A^i_L \cup A_R{}^i$) for every obtainable attribute. The divider is symbolized additional only by $A_L{}^i$. The complementary subset $A_R{}^i$ is automatically determined by selection of $A_L{}^i$.

In the ID3 algorithm split-measure function used as a maximizes information gain is a difference between the entropy and the weighted entropy. The maximizes value of the information gain is selected from the all likely partition of the set. For the subset of the training data set, the partition information gain is also called the optimal partition of number of element set. It is used to generate subset and this value called as an information gain of subset for attribute. One of the highest values of information gain is selected from the obtainable attributes in the node. The node split into two children nodes where the index of nodes created in the entire tree. The following two circumstances happened if the considered node is not split. They are,

1. All elements from the subset are from the same class.
2. Only one element is present in the list of available attributes in the node.

The problem of the concept drift is used as a part of the CVFDT algorithm [17] in this paper. It also replaces the Hoeffding's bound which is used incorrectly in the CVFDT algorithm. The thought of CVFDT algorithm published initially by Domingo's and Hulten in 2001 is correct, though these authors incorrectly used the Hoeffding's bound in their paper.

B. C4.5 Algorithm

C4.5 is an algorithm used to generate a decision tree developed by Ross Quinlan. It is the extension of ID3 algorithm that accounts for unavailable values, continuous attribute value ranges, pruning of decision trees, rule derivation, and so on. It is also refer as a statistical classifier. In non binary case the ID3 algorithm favors the attributes with large domain of possible values. To cope up with this problem C4.5 algorithm is used. In the C4.5 algorithm the ratio of the information gain and the split information is projected as the split measure function.

It has few base cases such as

- If all the samples in the list belong to the same class then it simply creates a leaf node for the decision tree saying to select that class.
- C4.5 creates a decision node higher up the tree using the predictable value of the class if not any of the features provide any information gain. C4.5 again creates a decision node higher up the tree using the expected value if Instance of previously-unseen class encountered. The pseudo code of C4.5 algorithm is,
  1. Check for base cases
  2. For each attribute *a* find the normalized information gain ratio from splitting on *a*
  3. Let *a_best* be the attribute with the highest normalized information gain.
  4. Create a decision node that split on *a_best*.
  5. Recurse on the sublists obtained by splitting on *a_best*, and add those nodes as children of *node.*

3. Related Works

It is very difficult to adapt the ID3 algorithm or any decision trees based on algorithm to data stream. The equivalent subsets of training data set incessantly cultivate due to this

reason it is tricky to approximation the values of split-measure function in each node. On the basis of infinite training data set hypothetically the information gain values is intended in the data stream case. But it is impossible that's why these values predictable from the obtainable data sample in the considered node. Due to this only with some possibility, one can make a decision which attribute is the most excellent. Here this paper converse the few efforts to solve this problem. They are,

1. G. Hulten and P. Domingos work on "Hoeffdings trees" [7] for data mining streams. It was resulting from the Hoeffding's bound [15], which states that with probability $1-\delta$ the accurate mean of a random variable of range R does not vary from the predictable mean, after n observations, by more than

$$\epsilon_H = \sqrt{\frac{R2 \ln 1/\delta}{2n}} \quad (2)$$

To solve the difficulty of selecting the attribute according to which the split should be made Hoeffding's bound is not a sufficient tool. It is a correct tool only for numerical data, which does not of necessity have to be met become aware of by the Rutkowski et.al [24]. The split measures like information gain and Gini index form is the second problem. Both measures are uses the elements frequency and cannot be expressed as a sum of elements.

2. One more method for finding the best attribute was projected in the work [18]. One exacting node will be considered for the expediency of the following text situation. Hence, the node index q will be absent in all notations introduced before. Let G_x and G_y be the values of information gain for attributes a_x and a_y calculated using a data sample in a considered node. Such amounts are random variables, whereas g_x and g_y are their predictable values, correspondingly. Jin and Agrawal [18] observe that the value G_x can be approximated by a normal distribution

$$G_x \rightarrow N\left(g_x, \frac{T2_x}{n}\right) (3)$$

Here number of elements in the sample is denoted as a n and variance of this distribution is denoted as a T^2_x /n. the difference of $G_x - G_y$ calculated by the normal distribution

$$G_x - G_y \rightarrow N\left(g_x - g_y, \frac{T2_{x+T2_y}}{n}\right) (4)$$

To make a decision is there attribute a^x gives higher value of information gain than attribute a^y, based on distribution (4), the suitable statistical test projected by the authors. Multivariate delta method is used to give good reason for the estimate.

Only two-class problem for binary trees are considered by the author. For example for selecting attribute, p=3,

- If the i^{th} element passes through the left branch then X_{1i} = 1 otherwise it is 0.
- If the i^{th} element is from the first class and X_{1i} =1 then X_{2i} = 1 and if the i^{th} element is from the second class and X1i = 1then X_{2i} =0 and
- If the i^{th} element is from the first class and X_{3i} =1 then X_{1i} = 0 and if the i^{th} element is from the second class and X_{3i} = 0then X_{1i} =0. Hence, g is the functions of three variables, they are
- P_L is a variable denoted as a fraction of elements transitory through the left branch
- P_{1L} is a variable denoted as a from the first class, the fraction of elements transitory through the left branch and
- P_{1R} is a variable denoted as from the second class, the fraction of elements transitory through the left branch.

3. The correcting the mathematical foundations of Hoeffding's trees worked by the Rutkowski et al. in [29]. Selecting the best attribute to make split in the node is extremely hard task and to solve this problem it is projected a McDiarmid's inequality.

A. The Gaussian Decision Trees Algorithm

Here, we projected a Gaussian decision tree algorithm which is the modification of the Hoeffding tree algorithm projected in [7]. The algorithm initiates with a single root also called leaf and the input parameters are initialized. The statistics of elements collected in the root are initialized which enough to compute all the essential values is come into view in the part of pseudocode. In the main loop of the algorithm, using the current tree it gets data or element from the stream and sorts it into a leaf. All statistics and majority class in leaf is updated. After that it ensures is there any class which is dominated to the other classes. It is also known as preprinting condition. The information gain values are calculated for each attribute if there is not establish any prepruning condition. After that the determination of the best attribute and second best attribute takes place. Then they calculate the value and verify that obtained values are enough or not to make a decision whether the split should be made or not. The leaf is replaced by a node with the attribute allocate to it if the answer is positive. At last the algorithm returns were a new data element from the stream is taken.

4. Conclusion

With the application of decision trees, data mining streams subjects measured in this paper. Selecting the best attribute to split the considered node is the key point in building the decision tree. This is solved by projecting the new method were if the best attribute determined for the current set of data elements in the node is also the best according to the entire stream. It is based on the properties of the normal distribution and Taylor's Theroms. Also C4.5 algorithm is use for building the decision tree which overcomes the problem of ID3 algorithm. It is also necessary mathematically. We also projected a GDT that is Gaussian Decision Tree algorithm. This algorithm radically outperforms the McDiarmid tree algorithm in the field of time expenditure. The GDT algorithm is able to give acceptable accuracies in data streams classification problems is shows by the numerical simulations.

References

[1] C. Aggarwal, Data Streams: Models and Algorithms. Springer, 2007.
[2] A. Bifet and R. Kirkby, "Data Stream Mining a Practical Approach," technical report, Univ. of Waikato, 2009.

[3] W. Fan, Y. Huang, H. Wang, and P.S. Yu, "Active Mining of Data Streams," Proc. SIAM Int'l Conf. Data Mining (SDM '04), 2004.
[4] M.M Gaber, A. Zaslavsky, and S. Krishnaswamy, "Mining Data Streams: A Review," ACM SIGMOD Record, vol. 34, no. 2, pp. 18-26, June 2005.
[5] J. Gama, R. Fernandes, and R. Rocha, "Decision Trees for Mining Data Streams," Intelligent Data Analysis, vol. 10, no. 1, pp. 23-45, Mar. 2006.
[6] J. Gao, W. Fan, and J. Hang, "On Appropriate Assumptions to Mine Data Streams: Analysis and Practice," Proc. IEEE Int'l Conf. Data Mining (ICDM '07), Oct. 2007.
[7] B. Pfahringer, G. Holmes, and R. Kirkby, "New Options for Hoeffding Trees," Proc. 20th Australian Joint Conf. Advances in Artificial Intelligence (AI '07), pp. 90-99, 2007.
[8] A. Bifet, Adaptive Stream Mining: Pattern Learning and Mining from Evolving Data Streams. IOS Press, 2010
[9] A. Bifet, G. Holmes, G. Pfahringer, R. Kirkby, and R. Gavalda, "New Ensemble Methods for Evolving Data Streams," Proc. 15th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD '09), June/July 2009.
[10] C. Franke, "Adaptivity in Data Stream Mining," PhD dissertation, Univ. of California, 2009.
[11] G. Hulten, L. Spencer, and P. Domingos, "Mining Time-Changing Data Streams," Proc. Seventh ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 97-106, 2001.
[12] J. Liu, X. Li, and W. Hong, "Ambiguous Decision Trees for Mining Concept-Drifting Data Streams," Pattern Recognition Letters, vol. 30, no. 15, pp. 1347-1355, Nov. 2009.
[13] A. Tsymbal, "The Problem of Concept Drift: Definitions and Related Work," Technical Report TCD-CS-2004-15, Computer Science Dept., Trinity College Dublin, Apr. 2004
[14] L .Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone, Classification and Regression Trees. Chapman and Hall, 1993.
[15] J. Han and M. Kamber, Data Mining: Concepts and Techniques, second ed., Elsevier, 2006.
[16] D.T. Larose, Discovering Knowledge in Data: An Introduction to Data Mining.Wiley & Sons, Inc., 2005.
[17] L. Rutkowski, Computational Intelligence: Methods and Techniques. Springer-Verlag, 2008
[18] I.H. Witten, E. Frank, and G. Holmes, Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufman, 2005.
[19] C. McDiarmid, "On the Method of Bounded Differences," Surveys in Combinatorics, J. Siemons, ed., pp. 148-188, Cambridge Univ. Press, 1989.
[20] P. Domingos and G. Hulten, "Mining High-Speed Data Streams," Proc. Sixth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 71-80, 2000.
[21] G. Hulten, L. Spencer, and P. Domingos, "Mining Time-Changing Data Streams," Proc. Seventh ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, pp. 97-106, 2001.
[22] R. Jin and G. Agrawal, "Efficient Decision Tree Construction on Streaming Data," Proc. Ninth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, 2003.
[23] L. Wasserman, All of Statistics: A Concise Course in Statistical Inference. Springer, 2005.
[24] L. Rutkowski, L. Pietruczuk, P. Duda, and M. Jaworski, "Decision Trees for Mining Data Streams Based on the McDiarmid's Bound," IEEE Trans. Knowledge and Data Eng., vol. 25, no. 6, pp. 1272-1279, 2013.
[25] J.R. Quinlan, C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993.