Journal Homepage: - www.journalijar.com
INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)
ISSN NO. 2320-5407
Article DOI: 10.21474/IJAR01/4338
DOI URL: http://dx.doi.org/10.21474/IJAR01/4338

RESEARCH ARTICLE

SYMBOLIC CLASSIFICATION FOR MULTIVARIATE TIME SERIES.

Amanpreet Singh[1], Dashmeet Kaur Sethi[1], Karneet Singh[1], Lakshay Sharma[1] and Poonam Narang[1].

1. Btech (CSE), GTBIT, New Delhi.
2. Ast. Professor(CSE/IT), GTBIT, New Delhi.

Manuscript Info

Manuscript History

Received: 21 March 2017
Final Accepted: 24 April 2017
Published: May 2017

Key words:-
Multivariate Time Series, Classification, Tree, Random Forest, SVM.

Abstract

With emergence of various marketing strategies, there is a need of efficient ways to collaborate all the information pertaining to our domain and hence find useful trends, patterns and its associations.
To do the same we use multivariate time series, where we concatenate a number of time series pertaining to a single domain. Consumption and supply are such examples to study the trends and patterns of the market, we also need to classify the human resources to identify our potential customers. This project is a brief comparison of the classification algorithms such as random forest and Support vector machines applied on a multivariate time series, it focusses at comparing the error rate of the above stated algorithm for different sizes of dataset, so that one can efficiently choose an algorithm and classification when wanting to study a multivariate time series.

Introduction:-

Multivariate Time Series Classification is a supervised learning problem in which each instance is composed of more than one attributes. Data pertaining to multivariate time series can be easily found in the areas of medicine, finance and multimedia.

Multivariate time series studies and emphasis on the relationship among different time series, rather than the similarity among them. Another disadvantage faced by multivariate time series is that it is multidimensional.
Here, we provide a classifier for multivariate time series. It considers all the attributes of multivariate time series and their relationships simultaneously, rather than studying them separately. The supervised learning algorithm does not require predefined intervals or features.

Each multivariate time series is concatenated and each instance is labelled with a class label. With R denoting the number of terminal nodes the dictionary of the classifier contains.

R symbols because of the fact that each series has a time index attribute, we can segment each multivariate time series by time or by value of any it's attribute. Further, the set of instances from each multivariate time series is characterized by frequency over the R symbols.

Corresponding Author:- Amanpreet Singh.
Address:- Btech (CSE), GTBIT, New Delhi.

We have more than one trees in the model, the name ensemble. Each multivariate time series is represented by a collection of distributions and the concatenated vector becomes codebook, much is learned on its own from simple representation of raw data.

Algorithm:-

Random Forest:-

Random forest is an ensemble classification algorithm. By saying ensemble, we mean that it concatenates both the advantage and disadvantages of more than one classification algorithm. It uses a decision tree approach wherein the nodes of decision tree represent the test cases, the link of the decision tree represent corresponding outcomes of test cases and class labels are represented by the leaf nodes respectively. Random Forest uses an assembly of decision trees and to calculate the results, it might use the average or weighted average of all the decision trees.

First of all, it generates tree of every multivariate time series with R terminal nodes. At the next level of learning more trees are generated in the same way represented by Symbols. Symbolic representation has the same length as the time series. Tree based methods empower predictor nodes with high accuracy stability and ease of interpretations. we use tree bagging in random forest to reduce variance.

Random Forest involves sampling of input data with replacement called bootstrap sampling. Here, one third of data is used for training and rest is used for testing. Testing data is known as Out Of Bag samples. The error estimated in these samples are called as Out Of Bag Error

Support Vector Machine:-

Support Vector Machine is Machine Learning Algorithm used for classification. It involves construction of the hyperplane which is infinite high dimensional space used for separating the input data points into different classes. Each sample space on the corresponding site of the hyper plane represent the class. It is a non-probability binary linear classifier. Support Vector Machine can also be used for clustering that is Clustering SVM is an improvement of Traditional SVM which is needed to group similar items on basis of similarity and association and relationships. The distance of data points of corresponding hyperplane should be maximum to minimize the out of bag error

Project Working:-

Input the training dataset of multivariate time series:-

Each multivariate time series dataset is divided into two sections, the training dataset and the testing dataset. The training dataset is used to build the classifier and train it to further classify the new instances by random data points in space. The testing dataset is used to evaluate the classifier for its fault. It helps us to compute the out of bag error and the error rate corresponding to the different value of R

The dataset (which is a collection of multivariate time series) should be such that each multivariate time eries should be concatenated vertically where the first column represents the time series, to when it belongs.

The second column represents the observation number for that time series and the third column represents the corresponding class labels respectively

Random Forest Algorithm is applied on concatenated dataset and decision trees with R, terminal nodes are build:-

Each Instance of multivariate time series is assigned to a terminal node of tree.

The Trees built are put together as a collection in codebook and the codebook acts as a classifier:-

New data instances are classified on the classifier, trained on this codebook.

Note: During Codebook generation, the input feature selection is expected to handle dimensional input.

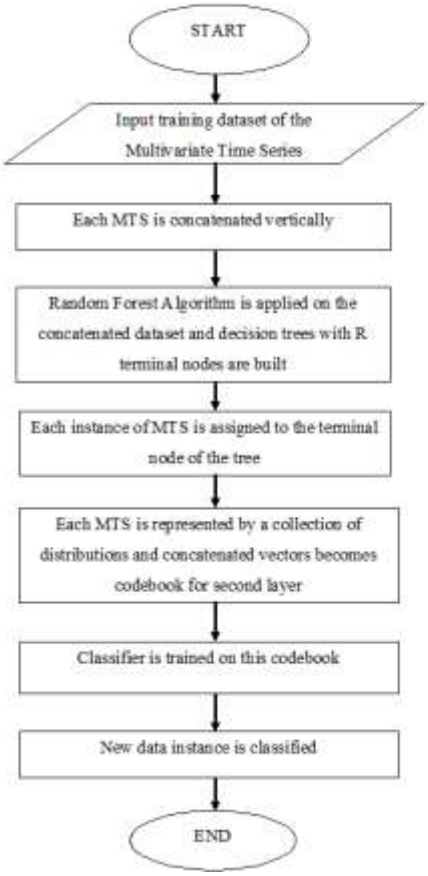

Figure 1:- Methodology of Project.

Experimental Results:-

Comparison Graphs:-

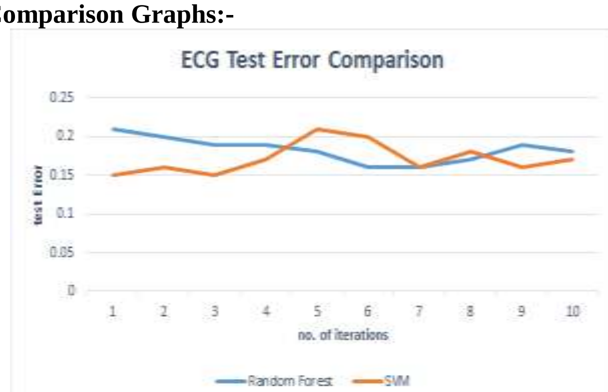

Figure 2:- Comparison Graph for ECG Test Error.

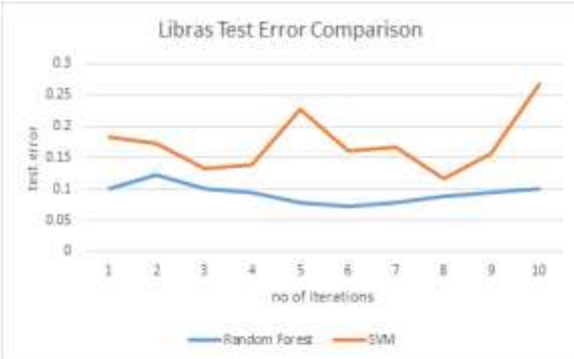

Figure 3:- Comparison Graph for Libras Test Error.

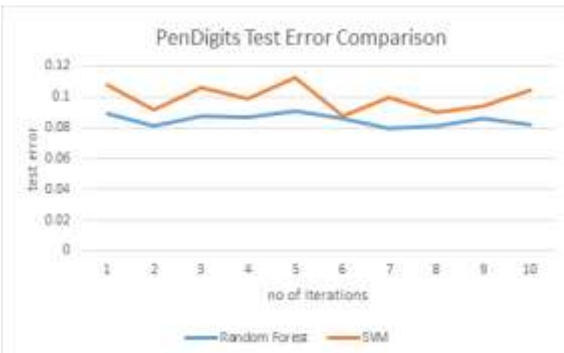

Figure 4:- Comparison Graph for Pen Digit Test Error.

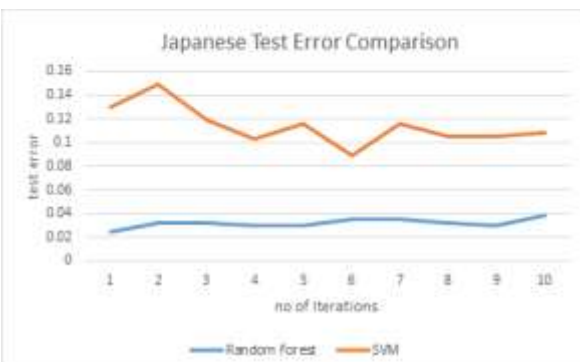

Figure 5:- Comparison Graph for Japanese Test Error

Conclusion Tables

```
"multivariate"
"Rep  R    Jins  Jts   OOB(E)  Test(E)"
" 1   20   50    100   0.120   0.160"
" 2   20   50    100   0.170   0.160"
" 3   100  50    150   0.090   0.190"
" 4   100  100   150   0.100   0.170"
" 5   100  50    100   0.080   0.180"
" 6   100  100   150   0.080   0.190"
" 7   20   100   100   0.130   0.160"
" 8   50   20    100   0.140   0.170"
" 9   20   50    200   0.100   0.150"
" 10  50   20    100   0.120   0.170"
"RESULTS SUMMARY OVER 10 REPLICATIONS"
"Average test error rate: 0.170"
"Min and Max test error rates: 0.150 and 0.190"
```

Figure 6:- Conclusion Table for ECG Dataset.

```
[1] "multivariate"
[1] "Rep   R   Jins   Jts   OOB(E)   Test(E)"
[1] " 1   100   50   150   0.022   0.030"
[1] " 2   50   50   200   0.026   0.041"
[1] " 3   100   100   200   0.019   0.024"
[1] " 4   100   20   150   0.015   0.032"
[1] " 5   100   20   100   0.026   0.035"
[1] " 6   100   50   100   0.030   0.030"
[1] " 7   100   100   100   0.022   0.035"
[1] " 8   100   50   150   0.019   0.030"
[1] " 9   100   20   150   0.019   0.014"
[1] " 10   100   20   200   0.019   0.027"
[1] "RESULTS SUMMARY OVER 10 REPLICATIONS"
[1] "Average test error rate: 0.030"
[1] "Min and Max test error rates: 0.014 and 0.041"
```

Figure 7:- Conclusion Table for ECG Dataset.

```
[1] "multivariate"
[1] "Rep   R   Jins Test(E)"
[1] " 1   20   100  0.150"
[1] " 2   20   50  0.150"
[1] " 3   100   20  0.180"
[1] " 4   100   20  0.160"
[1] " 5   100   100  0.200"
[1] " 6   50   20  0.160"
[1] " 7   100   50  0.190"
[1] " 8   50   50  0.160"
[1] " 9   20   50  0.170"
[1] " 10   100   100  0.180"
[1] "RESULTS SUMMARY OVER 10 REPLICATIONS"
[1] "Average test error rate: 0.170"
[1] "Min and Max test error rates: 0.150 and 0.200"
>
```

Figure 8:- Conclusion Table for ECG Dataset (SVM)

```
[1] "multivariate"
[1] "Rep   R   Jins Test(E)"
[1] " 1   100   50  0.124"
[1] " 2   100   50  0.124"
[1] " 3   100   100  0.108"
[1] " 4   100   50  0.116"
[1] " 5   50   20  0.065"
[1] " 6   100   50  0.111"
[1] " 7   100   50  0.103"
[1] " 8   100   50  0.154"
[1] " 9   50   100  0.062"
[1] " 10   100   50  0.111"
[1] "RESULTS SUMMARY OVER 10 REPLICATIONS"
[1] "Average test error rate: 0.108"
[1] "Min and Max test error rates: 0.062 and 0.154"
```

Figure 9:- Conclusion Table for Japanese Vowel Dataset (SVM).

Conclusion:-

Represents a MTS is a challenge for many methods. The algorithm that we used, does not required predefined time interval and features. In this method all attributes of MTS are considered simultaneously during a supervised process. So relationship between the individual attributes are taken into account. The early representation of raw data and first differences is quite simple conceptually and operationally. But a RF can detect interactions in the space S of time index and time value and this is used to generate a codebook. The codebook is processed with a second RF where now the implicit feature selection is exploited to handle the high-dimensional input. The constituent properties yield an approach quite different from current methods. Moreover, MTS with nominal and missing values are handled efficiently with tree learners. Ensemble learners that scale well with large number of attributes and long time series make SMTS computationally efficient. Our results and experiments demonstrate the effectiveness of the used approach in terms of accuracy for MTS as compared to other algorithms. Although not explored here, the proposed representation can be used for similarity analysis, and tasks such as clustering.

Acknowledgment:-
We take this opportunity to thank our mentors, professors and teachers whose kind and generous support helped us in keeping up with this project. We express our sincere gratitude to our guide “Ms. Poonam Narang” without whose guidance, the ongoing proceedings of the project would have been a mammoth task. Lastly, as no human endeavor is fully perfect, suggestions are further invited.

References:-
1. Kudo M, Toyama J, ShimboM(1999) Multidimensional curve classification using passing-through regions. Pattern Recognition Letters 20(1113)
2. Chan, K. & Fu, A. W. (1999). Efficient Time Series Matching by Wavelets. In proceedings of the 15th IEEE Int'l Conference on Data Engineering. Sydney, Australia, Mar 23-26. pp 126-133.
3. Faloutsos, C., Ranganathan, M., & Manolopoulos, Y. (1994). Fast Subsequence Matching in Time-Series Databases. In proceedings the ACM SIGMOD Int’l Conference on Management of Data. May 24-27, Minneapolis, MN. pp 419-429.
4. Keogh, E., Chakrabarti, K., Pazzani, M. & Mehrotra, S. (2001). Locally Adaptive Dimensionality Reduction for Indexing Large Time Series Databases. In proceedings of ACM SIGMOD Conference on Management of Data. Santa Barbara, CA, May 21-24. pp 151-162.
5. Larsen, R. J. & Marx, M. L. (1986). An Introduction to Mathematical Statistics and Its Applications. Prentice Hall, Englewood, Cliffs, N.J. 2nd Edition.
6. Durbin, R., Eddy, S., Krogh, A. & Mitchison, G. (1998). Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University Press.
7. Akl A, Valaee S (2010) Accelerometer-based gesture recognition via dynamic time warping, affinity propagation, compressive sensing. In Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on, pp 2270–2273, March
8. Orsenigo C, Vercellis C (2010) Combining discrete svm and fixed cardinality warping distances for multivariate time series classification. Pattern recognition
9. Bankó Z, Abonyi J (2012) Correlation based dynamic time warping of multivariate time series. Expert Systems with Applications
10. Weng X, Shen J (2008) Classification of multivariate time series using locality preserving projections. Knowledge-Based Systems 21(7):581–587
11. McGovern A, Rosendahl D, Brown R, Droegemeier K (2011) Identifying predictive multi-dimensional time series motifs: an application to severe weather prediction. Data Mining and Knowledge Discovery 22:232–258
12. Breiman L, Friedman J, Olshen R, Stone C (1984) Classification and Regression Trees. Wadsworth, Belmont, MA
13. Moosmann F, Nowak E, Jurie F (2008) Randomized clustering forests for image classification. IEEE Transactions on Pattern Analysis and Machine Intelligence
14. Ordonez P, Armstrong T, Oates T, Fackler J (2011) Using modified multivariate bag-of-words models to classify physiological data. In Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops, ICDMW ’11, pages 534–539, Washington, DC, USA, IEEE Computer Society.
15. Lin J, Khade R, Li Y (2012) Rotation-invariant similarity in time series using bag-of-patterns representation. Journal of Intelligent Information Systems,
16. Weng X, Shen J (2008) Classification of multivariate time series using locality preserving projections. Knowledge-Based Systems.
17. Li C, Khan L, Prabhakaran B (2006) Real-time classification of variable length multi-attribute motions. Knowledge and Information Systems.
18. Li C, Khan L, Prabhakaran B (2007) Feature selection for classification of variable length multiattribute motions. In Multimedia Data Mining and Knowledge Discovery.
19. Bankó Z, Abonyi J (2012) Correlation based dynamic time warping of multivariate time series. Expert Systems with Applications.
20. Breiman L (2001) Random forests. Machine Learning.