Journal of Geomatics Vol 9 No. 2 October 2015

SpatialDM: An open source data mining plugin for QGIS

Pratyush Kar[1] and Sameer Saran[2]
[1]Computer Science, BITS Pilani
[2]IIRS (ISRO), Dehradun
Email: pratyush.kar@gmail.com, sameer@iirs.gov.in

(Received: Aug 04, 2015; in final form: Oct 13, 2015)

Abstract: Spatial data mining is the extraction of classes, clusters or relationships between data points in a spatial dataset. It has huge applications in GIS, image database exploration, etc. When the datasets become large it becomes infeasible to manually label the data. This paper presents a software plugin named SpatialDM which is a tool to run classification algorithms on spatial datasets. The plugin is designed for Quantum GIS (an open source Geographic Information System software). QGIS is a cross-platform software i.e. it runs on Linux, Mac OSX, Windows and Android. The plugin includes three types of data mining classifiers: Decision trees, AdaBoost and Random Forest and has been designed to run on both multi-band raster layers and comma separated values (CSV) files.

Keywords: Data mining, Decision tree, Raster data, QGIS, GIS, Geospatial data, Open source

1. Introduction

Due to the advent of a large number of remote sensing satellites huge amounts of georeferenced spatial data are being generated. In order to utilise this information, it needs to be labelled. Labelling such a huge volume of data requires a tremendous amount of manual effort and is not feasible for the rate at which new data are being generated. Manual interpretation and analysis of such a huge bulk of data is infeasible and difficult.

Data mining is the process by which a computer identifies patterns and relationships between variables in the data provided to it. Spatial data mining is the application of data mining algorithms to spatial datasets. Spatial data mining is one of the most important problems in Geographic Information System (GIS). For example, in 1854 John Snow used clusters of cholera deaths on maps of London to conclude that cholera was a water-borne disease (Johnson, 2006). The maps he developed enabled him to pinpoint the infected water pumps that were responsible for the spreading of the infection.

Data mining techniques can be categorized in the following types:-

- Classification
- Clustering
- Regression
- Anomaly detection
- Association rule mining, etc.

The tool presented in this paper deals with the classification of data points in a spatial dataset. The tool has been developed as a plugin for Quantum GIS (QGIS) which is an open source GIS software whose source code is available under the GNU General Public License.

Traditionally remote sensing data is classified with the help of a parametric classifier like a maximum likelihood classifier. But, its use assumes that the input data is normally distributed (Jensen, 1996). However, geospatial data not only contains the satellite spectrum bands but also contains some ancillary layers like elevation, slope, etc. These layers are not normally distributed hence it is very difficult to classify such data using this classifier (Saran et al., 2007). In the plugin presented here we have used classifiers based on Decision trees, because they are non-metric in nature.

The rest of the paper is organised as follows. Section 2 talks about the state of the art in spatial data mining and discusses the importance of the tool proposed here. Section 3 provides a brief overview of the tool. Section 4 gives an analysis of the plugin on some sample datasets. Section 5 concludes this paper and proposes some future work.

2. Related work

Quite an amount of research has been done to develop methods to apply data mining algorithms on spatial data. Miller and Han (2009) provide a nice introduction to geographic data mining and geographic knowledge discovery (GKD). Since spatial datasets are very large an efficient database system to handle these datasets is required. Ester et al. (1997) proposed a set of basic operations that must be supported by a spatial database system (SDBS). Aref and Samet (1991) discussed adding few spatial operations on top of existing DBMS so that it can handle spatial queries. Koperski et al. (1996) cite some challenging issues that are faced when data mining algorithms are applied to large spatial datasets. Since data obtained by satellite images is very huge and data mining algorithms have to scan through the dataset multiple number of times hence, spatial data mining algorithms have a huge

Figure 1: SpatialDM GUI: GUI and interface for training (L) Decision tree classifier (M) AdaBoost classifier (R) Random forest classifier

computation overhead. Wang et al. (1997) proposed a novel hierarchical grid based approach to reducing this overhead. Ng and Han (1994) propose two novel spatial data mining algorithms based on CLARANS a clustering method based on randomised search. Koperski and Han (1996) demonstrated that it is possible to efficiently extract interesting association rules in large spatial data bases.

Although a large amount of research has been done on this topic, tools to run data mining algorithms on spatial data are lacking in most GIS software. The plugin presented here is created for the purpose of research and education in the field of spatial data mining.

3. SpatialDM: An overview

SpatialDM is a tool designed as a plugin for QGIS software. QGIS was chosen as the base software for the tool due to various reasons. First, because it is open source and cross-platform software, the source code can be changed very easily and it can run on any operating system. Secondly, the raster functionality in QGIS is implemented using GDAL (Geospatial Data Abstraction Library) hence it supports a large number of raster data formats for example TIFF, Erdas Imagine, GRASS Raster Format etc. (for a detailed list visit http://www.gdal.org/formats list.html). Lastly, QGIS has a built-in plugin system which enables developers to expand the functionality of QGIS using python plugins. Through the plugin system, the developer can use the QGIS core libraries along with other user-defined modules to create a new plugin.

3.1 GUI

The GUI of the plugin is made using the PyQt library which is a Python binding for Qt C++ framework. The plugin is designed to accept data in the form of both a comma separated values (CSV) file and multi-band raster layers. In CSV file format, the first row of the file denotes the attribute names while remaining rows denote the data points. For the multi-band raster format, each band of the raster layer denotes a unique attribute. For example, if a point is selected on the raster layer then the values of the bands at that point will be the attribute values of the selected data point. The plugin provides an option to select how the user wants to input the training data. In both the formats, the plugin provides an option to choose the target band (the outcome that we would like to predict). There is also an option to partition the input dataset into two partitions namely the training partition and the test partition. The classifier is trained using the training data partition. The test data partition is used to evaluate the generalization error of the classifier. After the training of the classifier, the plugin displays the test set accuracy, the training set accuracy and the confusion matrix. The current implementation only supports numerical data types (integer or double), support for string type datasets is not available. Fig. 1 depicts the GUI of the SpatialDM plugin in QGIS.

3.2 Algorithms

Currently, three data mining classifiers have been implemented in the SpatialDM plugin:-

- Decision tree classifier
- AdaBoost classifier
- Random Forest classifier

These classifiers have been implemented with the help of scikit-learn which is a machine learning library for Python (Pedregosa et al., 2011). Scikit-learn is open source and is available under the BSD license. It has been developed using standard scientific libraries for Python such as scipy and numpy and does not have external dependencies like R or Shogun. This makes it easier for developers to add additional algorithms to the existing framework.

3.2.1 Decision tree classifier: Decision trees are one of the most powerful and common classifiers in the arsenal of a data scientist. It is a supervised learning method that classifies data points based on a set of rules which are expressed in the form of a tree. Decision tree learns these rules on the basis of information gain which is the decrement in the node impurity. SpatialDM implements two different measures of node impurity namely the GINI index and the entropy.

$$GINI(t) = 1 - \sum[p(j|t)]^2 \quad (1)$$

GINI index can be represented as Eq. 1. Here $p(j|t)$ is the relative probability of class j at node t.

$$Entropy(t) = -\sum p(j|t)\log(p(j|t)) \quad (2)$$

Eq. 2 represents the entropy at node t. Here $p(j|t)$ is the relative probability of class j at node t.

$$GAIN_{split} = GINI(p) - \left(\sum_{i=1}^{k} \frac{n_i}{n} GINI(i)\right) \quad (3)$$

$$GAIN_{split} = Entropy(p) - \left(\sum_{i=1}^{k} \frac{n_i}{n} Entropy(i)\right) \quad (4)$$

Eq. 3 and 4 represent the information gain when GINI index and entropy is used as node impurity measure respectively. Here ***p*** denotes the parent node which is being split into ***k*** partitions. n_i is the number of records in partition ***i*** and ***n*** is the total number of records in the parent node. SpatialDM uses CART algorithm (Breiman et al., 1984) to build the decision trees hence it only supports numerical variables.

Besides these, the plugin also implements few parameters to tweak the final tree obtained. The user can set the maximum depth of the tree. By default, the tree continues to expand until the leaf obtained is pure. The user can also change the minimum number of samples in an internal node that can trigger a split. The total number of leaf nodes can also be set. If so set, the plugin will ignore the maximum depth setting. Once the classifier is built the plugin saves the tree in DOT format (graph description language) in the plugin directory. This DOT file can be opened using any graph visualization tool such as Graphviz (an open source graph visualization software from AT&T Research).

3.2.2 AdaBoost classifier: In AdaBoost classifiers, a large number of classifiers are trained and the consensus of these classifiers is the final result of the classifier. In AdaBoost, a large number of classifiers are built iteratively. After every iteration, the weights of the records that were misclassified are increased so that the next classifier pays more attention to them. SpatialDM implements two different boosting algorithms SAMME and SAMME.R. Besides this, a maximum number of classifiers can be adjusted and the learning rate (the contribution of each classifier) of the algorithm can also be tweaked.

3.2.3 Random Forest classifier: Random Forest is a type of ensemble classifier that generates a large number of decision trees and the consensus of all these trees is the result of the classifier. This reduces the variance of the classifier and reduces the classifier's tendency to over-fit. The Random Forest tab of the SpatialDM plugin provides options similar to those provided in the Decision tree tab. Moreover, it provides an option to create the trees simultaneously on multiple cores.

4. Sample data

In this section, the usage of the SpatialDM plugin on a test dataset and the training and test data accuracies of the three classifiers are illustrated. For this analysis a multi-band satellite dataset has been used. The dataset has 121 data points and 7 geospatial data layers. The 7 geospatial data layers have been listed in order in Table 1. The sample data is divided into 4 classes namely agriculture, forest, sand, and water.

Table 1: Data bands

Geospatial layer	Type
Slope	Continuous
Aspect	Continuous
Digital Elevation Model	Continuous
Green Band	Continuous
Red Band	Continuous
NIR Band (Near Infrared)	Continuous
Target Class	Nominal

Table 2 displays the parameters that were used to train the three classifiers. Here min_samples_split denotes the minimum number of records required for an internal node to be split. min_samples_leaf represents the minimum number of records required to be at the leaf node, max_depth is the maximum allowed depth of the decision tree, n_estimators is the maximum number of classifiers present in the AdaBoost algorithm.

Table 2: SpatialDM plugin parameters

Parameter	Value
Partition (%)	80
Criterion	entropy
min_samples_split	4
min_samples_leaf	2
max_depth	5
n_estimators	20
learning_rate	0.65
Boosting algorithm	SAMME.R

Table 3 summarizes the accuracy and the time required for training each of the classifiers. The plugin has been tested on a Mac Laptop (2.3 GHz Intel i7 Processor, 16 GB 1600 MHz DDR3 memory). The decision tree generated by the plugin is displayed in Fig. 2.

Table 3: Classifier performance

Classifier Name	Accuracy (%)	
	Training set	Test set
Decision tree	92.71	87.50
AdaBoost	92.67	83.33
Random Forest	96.88	91.67

The plugin was also tested on a larger dataset (with around 2500 records), the results of which have been displayed in Table 4.

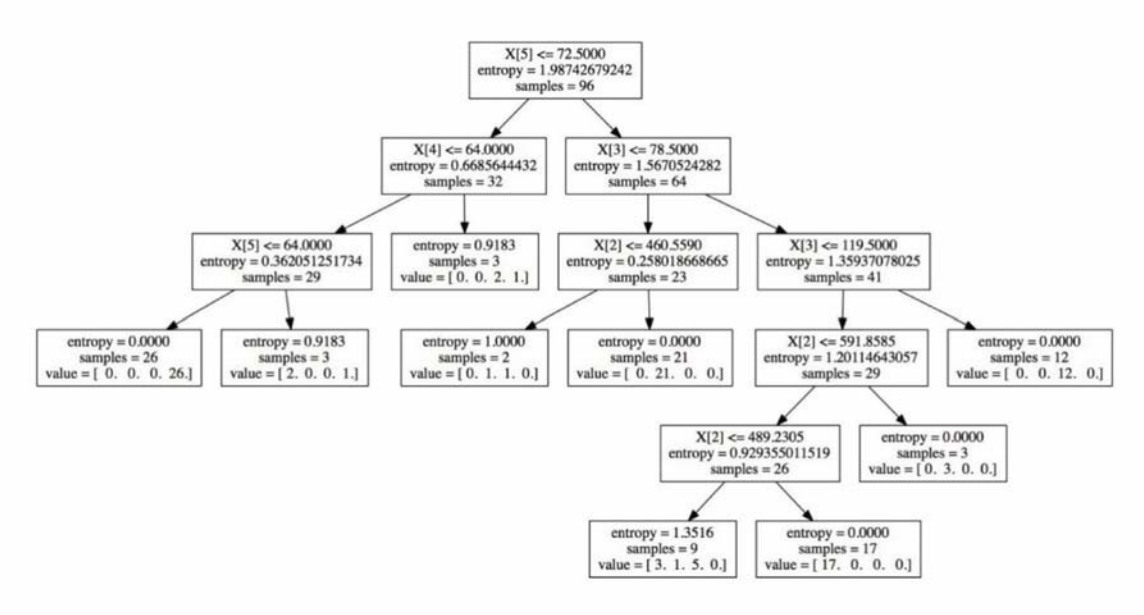

Figure 2: Decision tree generated using the SpatialDM plugin

Table 4: Classifier performance on large dataset

Classifier Name	Accuracy (%)		Training time (s)
	Training Set	Test set	
Decision tree	94.82	92.03	0.012
AdaBoost	89.59	88.04	0.131
Random Forest	95.73	92.53	0.021

5. Conclusion

Decision trees can be trained very efficiently and many fast algorithms to generate an optimal tree exist. However, decision trees fail to provide reasonable generalization errors when the test conditions involve multiple attributes at a time. Decision trees also face data fragmentation issues because a lesser amount of data is present as we go down the tree. AdaBoost classifiers and Random Forest classifier, being ensemble classifiers, are robust against these issues.

This plugin has been tested on QGIS version 2.8.2-Wein on Mac OSX but, is compatible with earlier versions of QGIS and works on other platforms as well. The source code of the plugin and the sample datasets used in the previous section are available online at https://github.com/p-kar/SpatialDM. Further versions of the plugin will include clustering and association rule mining algorithms for spatial data.

Acknowledgement

The authors would like to thank Mr. Anand Maurya for his help in the implementation of this plugin.

References

Aref, W.G. and H. Samet (1991). Extending a DBMS with spatial operations. Proceedings of the 2nd International Symposium on Advances in Spatial Databases.

Breiman, L., J.H. Friedman, R.A. Olshen and C.J. Stone (1984). Classification and regression trees. Wadsworth Inc.

Ester, M., Hans-Peter Kriegel and J. Sander (1997). Spatial data mining: A database approach. Proceedings of the 5th International Symposium on Large Spatial Databases.

Jensen, J.R. (1996). Introductory digital image processing: A remote sensing perspective, 2nd Edition. Upper Saddle River: Prentice Hall.

Johnson, S. (2006). The ghost map: The story of London's most terrifying epidemic – and how it changed science, cities, and the modern world. Riverhead Books, New York.

Koperski, K., J. Adhikary and J. Han (1996). Spatial data mining: Progress and challenges survey paper. SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery.

Koperski, K. and J. Han (1996). Discovery of spatial association rules in geographic information databases. Proceedings of the 4th International Symposium on Advances in Spatial Databases.

Miller, H.J. and J. Han (2009). Geographic data mining and knowledge discovery: An overview. CRC Press.

Ng, R.T. and J. Han (1994). Efficient and effective clustering methods for spatial data mining. Proceedings of the 20th International Conference on Very Large Data Bases.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay (2011). Scikit-learn: Machine learning in Python. The Journal of Machine Learning Research 12, pp. 2825-2830.

Saran, S., A. Bharti, G. Sterk and P.L.N. Raju (2007). Comparing and optimising land use classification in a Himalayan area using parametric and non parametric approaches. Journal of Geomatics, Vol. 1, No. 1, pp. 30-38.

Wang, W., J. Yang and R. Muntz (1997). STING: A statistical information grid approach to spatial data mining. Proceedings of the 23rd International Conference on Very Large Data Bases.