International Journal of Enhanced Research in Management & Computer Applications, ISSN: 2319-7471
Vol. 2 Issue 9, Nov.-Dec., 2013, pp: (5-13), Available online at: **www.erpublications.com**

A clustering method for seismic zone identification and spatial data mining

Soriful Hoque[1], Salim Istyaq[2], Mohammad Mushir Riaz[3]

Abstract: This paper shows how it made possible in geographical science to observe the seismic zone, clustering of highly sensitive earthquake zone and spatial data clustering during important geographical processes. This paper shows simple density based and K- Mean clustering technique. Density-Based clustering is done here using density estimation and by searching regions which are denser than a given threshold and to form clusters from these dense regions by using connectivity and density functions. Also we defined some optimal no of K locations for K-Mean clustering where the sum of the distance from every point to each of the K centers is minimized what is called global optimization. With this dataset it forms clusters using density estimation and K-Mean clustering. Also it correlates the clustering pattern by applying co-relation algorithm and proximity measure algorithm; hence it easily removes noisy data. This scheme can extract clusters efficiently with reduced number of comparisons.

Keywords: Clustering, co-relation, density based, K-Mean, proximity measure, spatial dataset, seismic zone.

I. INTRODUCTION

SEISMIC data collection refers to collecting huge spatial data for large geographical area. These data are placed in multidimensional array for analysis and formed desired pattern. As seismological data are multidimensional, they need to be stored and recovered by special techniques, more complex compared to those used for the traditional alphanumerical data. Under this point of view, spatial entities referred to temporal periods or temporal moments referred to layers of geographical information are under investigation within the frame of Database Management Systems. The explosive growth of spatial data and widespread use of spatial databases emphasize the need for the automated discovery of spatial knowledge. Spatial data mining is the process of discovering interesting and previously unknown, but potentially useful patterns from spatial databases. The complexity of spatial data and intrinsic spatial relationships limits the usefulness of conventional data mining techniques for extracting spatial patterns. Spatial data are the data related to objects that occupy space. A spatial database stores spatial objects represented by spatial data types and spatial relationships among such objects. Spatial data carries topological and/or distance information and it is often organized by spatial indexing structures and accessed by spatial access methods. These distinct features of a spatial database pose challenges and bring opportunities for mining information from spatial data. Spatial data mining, or knowledge discovery in spatial database, refers to the extraction of implicit knowledge, spatial relations, or other patterns not explicitly stored in spatial databases. Till a few years back, statistical spatial analysis had been the most common approach for analyzing spatial data. Statistical analysis is a well studied area and therefore there exist a large number of algorithms including various optimization techniques. It handles very well numerical data and usually comes up with realistic models of spatial phenomena. The major disadvantage of this approach is the assumption of statistical independence among the spatially distributed data. This causes problems as many spatial data are in fact interrelated, i.e., spatial objects are influenced by their neighboring objects. Kriging (interpolation technique) or regression models with spatially lagged forms of the dependent variables can be used to alleviate this problem to some extent. Statistical methods also do not work well with incomplete or inconclusive data. Another problem related to statistical spatial analysis is the expensive computation of the results. With the advent of data mining, various methods for discovering knowledge from large spatial databases have been proposed and many such methods can be developed to the different kind of datasets. Spatial Data Mining is a special kind of data mining. The main difference between data mining and spatial data mining is that in spatial data mining tasks we use not only non-spatial attributes (as it is usual in data mining in non-spatial data), but also spatial attributes. Spatial data mining is the process of discovering interesting and previously un-known, but potentially useful patterns from large spatial datasets. Extracting interesting and useful patterns from spatial datasets is more difficult than extracting the corresponding patterns from traditional numeric and categorical data due to the complexity of spatial data types, spatial relationships, and spatial autocorrelation. Specific features of geographical data that preclude the use of general purpose data mining algorithms are:

- Rich data types (e.g., extended spatial objects)
- Implicit spatial relationships among the variables
- Observations that are not independent, and
- Spatial autocorrelation among the features.

II. PRE-PROCESSING OF SPATIAL DATA

Spatial data mining techniques have been widely applied to the data in many application domains. However, research on the preprocessing of spatial data has lagged behind. Hence, there is a need for preprocessing techniques for spatial data to deal with problems such as treatment of missing location information and imprecise location specifications, cleaning of spatial data, feature selection, and data transformation. Unique features of Spatial Data Mining that distinguishes spatial data mining from classical data mining in the following four categories:

Data input: The data inputs of spatial data mining are more complex than the inputs of classical data mining because they include extended objects such as points, lines, and polygons. The data inputs of spatial data mining have two distinct types of attributes: non-spatial attribute and spatial attribute. Non-spatial attributes are used to characterize non-spatial features of objects, such as name, population, and unemployment rate for a city. They are the same as the attributes used in the data inputs of classical data mining. Spatial attributes are used to define the spatial location and extent of spatial objects. The spatial attributes of a spatial object most often include information related to spatial locations, e.g., longitude, latitude and elevation, as well as shape. Relationships among non-spatial objects are explicit in data inputs, e.g., arithmetic relation, ordering, is instance of, subclass of, and membership of. In contrast, relationships among spatial objects are often implicit, such as overlap, intersect, and behind. One possible way to deal with implicit spatial relationships is to materialize the relationships into traditional data input columns and then apply classical data mining techniques. However, the materialization can result in loss of information. Another way to capture implicit spatial relationships is to develop models or techniques to incorporate spatial information into the spatial data mining process.

This clustering algorithm:

- provides a density based and K-mean cluster solution;
- it uses of proximity measures;
- faster processing due to simplified matching mechanism;
- capable of handling noisy datasets;

III. CLUSTERING TECHNIQUES

The goal of spatial clustering is to group co-related spatial data together. Co-related data indicates co-function and same seismic zone. Spatial data has certain special characteristics and is a challenging research problem. Here, we review a series of spatial data clustering algorithms.

3.1 K-Means:

K-means represents an attempt to define an optimal number of k locations where the sum of the distance from every point to each of the k centers is minimized what is called global optimization. In practice, (1) making initial guesses about the k locations and (2) local optimization for cluster locations in relation to the nearby points is implemented. Thus, two k-means procedures might not produce the same results, even if k is identical because of several underlying local optimization methods.

The k-means algorithm is built upon four basic operations:

- selection of the initial k means for k clusters,
- calculation of the dissimilarity between an object and the mean of a cluster,
- allocation of an object to the cluster whose mean is nearest to the object,
- Re-calculation of the mean of a cluster from the objects allocated to it so that the intra cluster dissimilarity is minimised. Except for the first operation, the other three operations are repeatedly performed in the algorithm until the algorithm converges (until no points change clusters). The essence of the algorithm is to minimise the cost function which is a function of dissimilarity measure between each observation with mean of cluster. Dissimilarity is usually modelled as Euclidean Distance in k-means. The cost function is as follows;

$$\text{Minimize} \sum_{j=1}^{N}\sum_{k=1}^{k} a_j d_{jk} Z_{jk}$$

Where

j , k denotes total number of observations and clusters
a_j denotes weight of observation j ,
d_{jk} denotes distance between observation j and centre of cluster k , and

Z_{jk}= { 1 if the observation j is in cluster k,
0 otherwise

3.2 Density Based Clustering: A Brief Review

- This work presents a density based clustering technique.
- It retains the regulation information which is also the main advantage of the clustering.
- It uses no proximity measures and is therefore free of the restrictions offered by them.
- Our approach improves the cluster quality by identifying sub-clusters within big clusters.
- It was compared with some well-known clustering algorithms and found to perform well in terms of the z-score cluster validity measure.

Works in two phases:

Phase 1

- Normalizing and discretizing the spatial dataset.
Normalization of the spatial dataset to have mean 0 and standard deviation 1. Expression data having low variance across conditions as well as data having more than 3-fold variation are filtered in this step.

- Clustering the discretized normalized data.
Discretization is then performed on this normalized expression data where the regulation pattern, i.e. up- or down-regulation in each of the conditions for a particular spatial object plays an important role. While discretizing, following two cases will occur.

i. The discretized value of spatial object obi at condition, t_1 (i.e., the first condition)

$$\xi_{gi,t_1} = \begin{cases} 1 & if \ \ \xi_{gi,t_1} > 0 \\ 0 & if \ \ \xi_{gi,t_1} = 0 \\ -1 & if \ \ \xi_{gi,t_1} < 0 \end{cases}$$

ii. The discretized values of spatial object ob_i at conditions t_j (j = 1,..(T −1)) i.e., at the rest of the conditions (T − {t_1})

$$\xi_{gi,t_{j+1}} = \begin{cases} 1 & if \ \ \varepsilon_{gi,t_j} < \varepsilon_{gi,t_{j+1}} \\ 0 & if \ \ \varepsilon_{gi,t_j} = \varepsilon_{gi,t_{j+1}} \\ -1 & if \ \ \varepsilon_{gi,t_j} > \varepsilon_{gi,t_{j+1}} \end{cases}$$

where - ob_i, t_j is the discretized value of object obi at condition t_j (j = 1,..(T − 1)).

The expression value of spatial object obi at condition t_j is given by "ob_i,t_j . We see in the above computation that the first condition, t_1, is treated as a special case and it's discretized value is directly based on "obi,t_1 i.e., the expression value at condition t_1. For the rest of the conditions the discretized value is calculated by comparing its expression value with that of the previous value. This helps in finding whether the object is up- (1) or -down (-1) regulated at that particular condition. Each object will now have a regulation pattern (}) of 0, 1, and -1 across the conditions or time points.

IV. GENERALIZED DENSITY-BASED CLUSTERING

Clustering is the technique of grouping the objects of a database into meaningful subclasses (that is, clusters) so that the members of a cluster are as similar as possible whereas the members of different clusters differ as much as possible from each other. Applications of clustering in spatial databases are, e.g., the detection of seismic faults by grouping the entries of an earthquake catalog or the creation of thematic maps in geographic information systems by clustering feature vectors. The clustering algorithms can be supported by the database primitives if the clustering algorithm is based on a "local" cluster condition, i.e. if it constructs clusters by analyzing a restricted neighbourhood of the objects. Examples are the density-based clustering algorithm DBSCAN as well as its generalized version GDBSCAN which is discussed in the following. GDBSCAN (Generalized Density Based Spatial Clustering of Applications with Noise) relies on a density-based notion of clusters. The key idea of a density-based cluster is that for each point of a cluster its ε-neighbourhood for some given $\varepsilon > 0$ has to contain at least a minimum number of points, i.e. the "density" in the ε-neighbourhood of points has to exceed some threshold. "Density-based clusters" can be generalized to density-connected sets in the following way:

First, any notion of a neighbourhood can be used instead of an ε-neighbourhood if the definition of the neighbourhood is based on a binary predicate NPred which is symmetric and reflexive. Second, instead of simply counting the objects in a neighbourhood of an object, other measures to define an equivalent of the "cardinality" of that neighbourhood can be used as well. For that purpose we assume a predicate Min Weight which is defined for sets of objects and which is true for a neighbourhood if the neighbourhood has the minimum weight (e.g. a minimum cardinality as for density- based clusters). Whereas a distance-based neighbourhood is a natural notion of a neighbourhood for point objects, it may be more appropriate to use topological relations such as intersects or meets to cluster spatially extended objects such as a set of polygons of largely differing sizes. There are also specializations equivalent to simple forms of region growing, i.e. only local criteria for expanding a region can be defined by the weighted cardinality function.

For instance, the neighbourhood may be given simply by the neighbouring cells in a grid and the weighted cardinality function may be some aggregation of the non-spatial attribute values. While region growing algorithms are highly specialized to pixels, density-connected sets can be defined for any data types.

The Algorithm

- The clustering process starts with an arbitrary spatial data object ob_i and searches the neighborhood of it to check if it is core.
- If ob_i is not core then the process restarts with another unclassified object.
- If ob_i is a core object, then clustering proceeds with finding all reachable object from ob_i.
- All reachable object are assigned the same sub cluster id as ob_i . From the neighbors of ob_i, if any object satisfies the core object condition, sub cluster expansion proceeds with that object.
- The process continues till no more object can be assigned to the sub cluster.
- The process then restarts with another unclassified object and starts forming the next sub cluster.
- The clustering process continues till no more object can be assigned sub cluster id.
- Once all sub clusters have been assigned, the process groups all sub-clusters as well as genes having no sub cluster id but having the same regulation pattern into the same cluster and assign them the same cluster id.
- All unclassified object are now termed as noise spatial data.

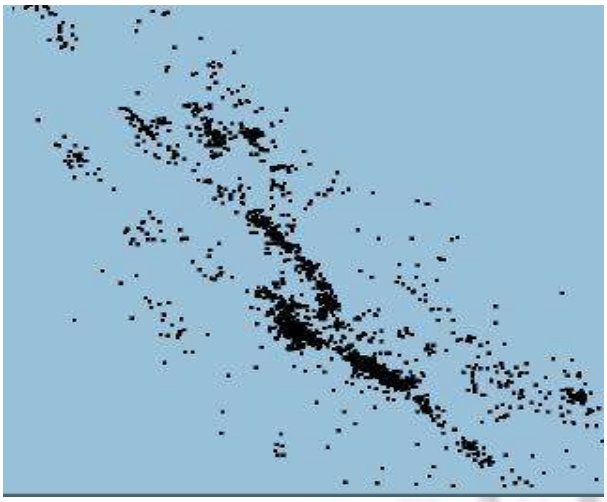

Fig.1: Spatial Seismic Data

Fig. 2: Sample Spatial Data for density based clustering

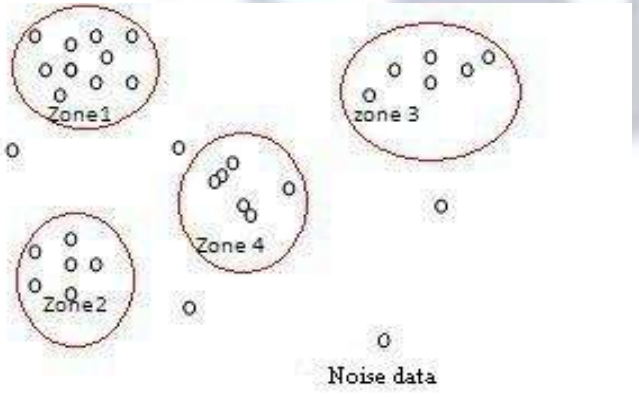

Fig. 3: Seismic zone clustering

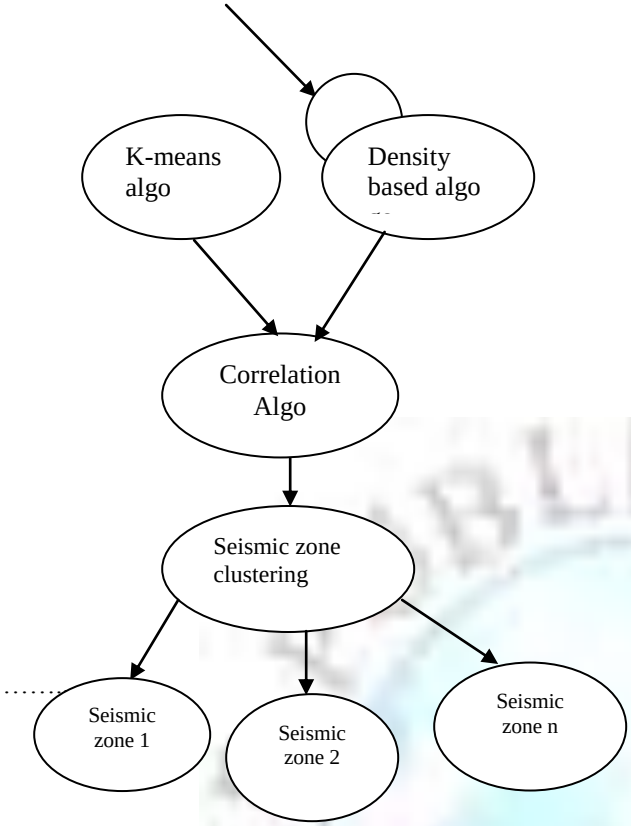

Fig. 4: Systematic Structure of Spatial Data mining

V. PROXIMITY MEASUREMENT FOR SPATIAL DATA

Proximity measurement measures the similarity (or distance) between two data objects. Gene expression data objects, no matter genes or samples, can be formalized as numerical vectors.

Euclidean distance is one of the most commonly-used methods to measure the distance between two data objects. The **Euclidean distance** between points **p** and **q** is the length of the line segment connecting them. $(\overline{pq})$

In Cartesian coordinates, if **p** = (p_1, p_2,..., p_n)

and **q** = (q_1, q_2,..., q_n) are two points in Euclidean n-space, then the distance from **p** to **q**, or from **q** to **p** is given by:

$$d(p,q) = d(q,p) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + ... + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

The position of a point in a Euclidean n-space is a Euclidean vector. So, **p** and **q** are Euclidean vectors, starting from the origin of the space, and their tips indicate two points. The **Euclidean norm**, or **Euclidean length**, or **magnitude** of a vector measures the length of the vector:

$$\|p\| = \sqrt{p_1^2 + p_2^2} + ... + p_n^2 = \sqrt{p.p}$$

where the last equation involves the dot product. A vector can be described as a directed line segment from the origin of the Euclidean space (vector tail), to a point in that space (vector tip). If we consider that its length is actually the distance from its tail to its tip, it becomes clear that the Euclidean norm of a vector is just a special case of Euclidean distance: the Euclidean distance between its tail and its tip.

The distance between points **p** and **q** may have a direction (e.g. from **p** to **q**), so it may be represented by another vector, given by

$$q - p = (q_1 - p_1, q_2 - p_2 ....., q_n - p_n)$$

In a three-dimensional space (n=3), this is an arrow from **p** to **q**, which can be also regarded as the position of **q** relative to **p**. It may be also called a displacement vector if **p** and **q** represent two positions of the same point at two successive instants of time.

The distance between points **p** and **q** may have a direction (e.g. from **p** to **q**), so it may be represented by another vector, given by

$$q - p = (q_1 - p_1, q_2 - p_2 ....., q_n - p_n)$$

In a three-dimensional space (n=3), this is an arrow from **p** to **q**, which can be also regarded as the position of **q** relative to **p**. It may be also called a displacement vector if **p** and **q** represent two positions of the same point at two successive instants of time.

The Euclidean distance between **p** and **q** is just the Euclidean length of this distance (or displacement) vector:

$$|q - p| = \sqrt{(q - p.(q - p))}$$

which is equivalent to equation 1, and also to:

$$\|q - p\| = \sqrt{\|p\|^2 + \|q\|^2 - 2p.q}$$

However, for gene expression data, the overall shapes of gene expression patterns (or profiles) are of greater interest than the individual magnitudes of each feature. Euclidean distance does not score well for shifting or scaled patterns. To address this problem, each object vector is standardized with zero mean and variance one before calculating the distance.

Pearson's correlation coefficient between two variables is defined as the covariance of the two variables divided by the product of their standard deviations.

Pearson's correlation coefficient when applied to a population is commonly represented by the Greek letter ρ (rho) and may be referred to as the population correlation coefficient or the population Pearson correlation coefficient. The formula for ρ is:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

Pearson's correlation coefficient when applied to a sample is commonly represented by the letter r and may be referred to as the sample correlation coefficient or the sample Pearson correlation coefficient. We can obtain a formula for r by substituting estimates of the covariance's and variances based on a sample into the formula above. That formula for r is:

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

An equivalent expression gives the correlation coefficient as the mean of the products of the standard scores. Based on a sample of paired data (X_i, Y_i), the sample Pearson correlation coefficient is

$$r = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{X_i - \bar{X}}{S_x}\right)\left(\frac{X_i - \bar{X}}{S_y}\right)$$

Where $\frac{X_i - \bar{X}}{S_X}$, $\bar{X}$ and S_x are the standard score, sample mean, and sample standard deviation, respectively.

The absolute value of both the sample and population Pearson correlation coefficients are less than or equal to 1. Correlations equal to 1 or -1 correspond to data points lying exactly on a line (in the case of the sample correlation), or to a bivariate distribution entirely supported on a line (in the case of the population correlation). The Pearson correlation coefficient is symmetric: corr(X,Y) = corr(Y,X).

A key mathematical property of the Pearson correlation coefficient is that it is invariant (up to a sign) to separate changes in location and scale in the two variables. That is, we may transform X to a + bX and transform Y to c + dY, where a, b, c, and d are constants, without changing the correlation coefficient (this fact holds for both the population and sample Pearson correlation coefficients).

The Pearson correlation can be expressed in terms of uncensored moments.

Since $\mu_X = E(X)$, $\sigma_X^2 = E[(X - E(X))^2] = E(X^2) - E^2(X)$ and likewise for Y, and since

$E[(X - E(X))(Y - E(Y))] = E(XY) - E(X)E(Y)$, the correlation can also be written as

$$\rho_{X,Y} = \frac{E(XY) - E(X)E(Y)}{\sqrt{E(X^2) - (E(X))^2}\sqrt{E(Y^2) - (E(Y))^2}}$$

Alternative formulae for the sample Pearson correlation coefficient are also available:

$$r_{xy} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{(n-1)S_x S_y} = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}}$$

The above formula suggests a convenient single-pass algorithm for calculating sample correlations, but, depending on the numbers involved, it can sometimes **be numerically unstable.**

VI. CONCLUSION

The objective of this paper is to build a program that generates an application menu for the user. The system developed is able to meet all the basic requirements. There is always a room for improvement in any software, however efficient the system may be. The important thing is that the system should be flexible enough for future modifications. The system has been factored into different modules to make system adapt to the further changes. Every effort has been made to cover all user requirements and make it user friendly. This work presents a density based clustering approach which finds seismic zone of highly correlated spatial data within a cluster. This clustering does not require the number of clusters priory and the clusters obtained have been found satisfactory on visual inspection and also based on z-score for two real datasets. Work is going on for establishing the effectiveness of seismic zone clustering over more real-life datasets.

ACKNOWLEDGMENT

Although it is simple technique to find seismic zone and spatial data clustering, it is tedious job to implement it practically because we need raw spatial dataset. So we thank our colleague and lab staff who helped us in this regards. Also we have taken guidance from some reference papers published regarding gene based clustering and books regarding spatial data mining. The ability to help and patience to exercise diligence and provide support is a quality admonished by very few. Any job in this world, however trivial or tough can not be accomplished without the assistance of the others. We would hereby take the opportunity to express our indebtedness to people who have helped us to accomplish this task.

REFERENCES

[1] Allard D. and Fraley C.:"Non Parametric Maximum Likelihood Estimation of Features in Spatial Point Process Using Voronoi Tessellation", Journal of the American Statistical Association, to appear in December 1997.[Available at http://www.stat.washington.edu/tech.reports/tr293R.ps].
[2] Beckmann N., Kriegel H.-P., Schneider R., Seeger B.: 'TheR*-tree: An Efficient and Robust Access Method for Points and Rectangles', Proc. ACM SIGMOD Int. Conf. on Management of Data, Atlantic City, NJ, 1990, pp. 322-331.
[3] Banfield J. D. and Raftery A. E.: "Model based Gaussian and non-Gaussian clustering", Biometrics 49, September 1993, pp. 803-821.
[4] Byers S. and Raftery A. E.: "Nearest Neighbor Clutter Removal for Estimating Features in Spatial Point Processes", Technical Report No. 305, Department of Statistics, University of Washington. [Available at http://www.stat.washington.edu/ tech.reports/tr295.ps]
[5] Devore J. L.: 'Probability and Statistics for Engineering and the Sciences', Duxbury Press, 1991.
[6] Ester M., Kriegel H.-P., Sander J., Xu X.: "A Density- Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", Proc. 2cnd Int. Conf. on Knowledge Discovery and Data Mining, Portland, Oregon, 1996, AAAI Press, 1996.

[7] Ester M., Kriegel H.-P., Xu X.: "Knowledge Discovery in Large Spatial Databases: Focusing Techniques for Efficient Class Identification", Proc. 4th Int. Symp. On Large Spatial Databases, Portland, ME, 1995, in: Lecture Notes in Computer Science, Vol. 951, Springer, 1995, pp.67-82.
[8] Fayyad U. M.,.J., Piatetsky-Shapiro G., Smyth P.: "From Data Mining to Knowledge Discovery: An Overview", in: Advances in Knowledge Discovery and Data Mining, AAAI Press, Menlo Park, 1996, pp. 1 - 34.
[9] Gueting R. H.: "An Introduction to Spatial Database Systems", in: The VLDB Journal, Vol. 3, No. 4, October 1994, pp.357-399.
[10] Kaufman L., Rousseeuw P. J.: "Finding Groups in Data: An Introduction to Cluster Analysis", John Wiley & Sons,1990.
[11] McKenzie M., Miller R., and Uhrhammer R.: "Bulletin of the Seismographic Stations", University of California, Berkeley. Vol. 53, No. 1-2.
[12] Muise R. and Smith C.: "Nonparametric minefield detection and localization", Technical Report CSS-TM- 591-91, Naval Surface Warfare Center, Coastal Systems Station.
[13] Ng R. T., Han J.: "Efficient and Effective Clustering Methods for Spatial Data Mining", Proc. 20th Int. Conf. on Very Large Data Bases, Santiago, Chile, 1994, pp. 144-155.
[14] Zhang T., Ramakrishnan R., Linvy M.: "BIRCH: An Efficient Data Clustering Method for Very Large
[15] Databases", Proc. ACM SIGMOD Int. Conf. on
[16] Agarwal, P., & Skupin, A. (2008). Self-organising maps: Applications in geographic information science. Chichester: Wiley.
[17] Agrawal, R., Imielinski, T., Swami, A. (1993). Mining association rules between sets of items in large databases. In ACM SIGMOD international conference on management of data (pp. 207–216). 17.Andrienko, G., & Andrienko, N. (1999). Data mining with C4.5 and interactive cartographic visualization. In N. W. G. T. Paton (Ed.), User interfaces to data intensive systems (pp. 162–165). 18.Los Alamitos, CA: IEEE Computer Society. Anselin, L. (1999). Interactive techniques and exploratory spatial data analysis. In P.A. Longley, M. F. Goodchild, D. J. Maguire, & D. W. Rhind (Eds.), Geographical information systems–principles and technical issues (pp. 253–266). New York, NY:
[18] John Wiley & Sons, Inc..
[19] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lan-der, and T.R. Golub. Interpreting patterns of gene expression with self-organizing
[20] maps: Methods and application to hematopoietic differentiation. In Proceedings of National Academy of Sciences, volume 96(6), pages 2907–2912, USA, 1999.
[21] J. Dopazo and JM. Carazo. Phylogenetic reconstruction using an unsupervised neural network that adopts the topology of a phylogenetic tree. J Mol Eval, 44:226–233, 1997.
[22] A. Bhattacharya and R. De. Divisive correlation clustering algorithm (dcca) for grouping of genes: detecting varying patterns in expression profiles. Bioinformatics, 24(11):1359–1366, 2008.
[23] R. J. Cho, M. Campbell, E. Winzeler, L Steinmetz, et al. A genome-wide tran-scriptional analysis of the mitotic cell cycle. Mol. Cell, 2(1):6573, 1998.
[24] F. Gibbons and F. Roth. Judging the quality of gene expression based clustering methods using gene annotation. Genome Research, 12:1574–1581, 2002.