Crime Hot-Spots Prediction Using Support Vector Machine

Keivan Kianmehr
Department of Computer Science
University of Calgary
Calgary, Alberta, CANADA
kiamehr@cpsc.ucalgary.ca

Reda Alhajj
Department of Computer Science
University of Calgary
Calgary, Alberta, CANADA
alhajj@cpsc.ucalgary.ca

Abstract

Location prediction is a special case of spatial data mining classification. For instance, in the public safety domain, it may be interesting to predict location(s) of crime hot spots. In this study, we present Support Vector Machine (SVM) based approach to predict the location as alternative to existing modeling approaches. SVM forms the new generation of machine learning techniques used to find optimal separability between classes within datasets. Experiments on two different spatial datasets show that SVMs gives reasonable results.

1 Introduction

Data mining employs algorithms and techniques from statistics, machine learning, artificial intelligence, databases and data warehousing, etc [1]. Data mining techniques have been successfully applied to analyze spatial data, i.e., data related to objects that occupy space. Spatial data carries topological and/or distance information; it is often organized by spatial index structures and accessed by spatial access methods. These distinct features of a spatial database bring challenges and opportunities for mining knowledge from spatial data [2]. Spatial Data mining is a subfield of data mining; it is a process that uses a variety of data analysis tools to discover spatial patterns and relationships in spatial data that may be used to make valid predictions [3, 4]. This has wide applications in Geographic Information Systems (GIS), remote sensing, image databases exploration, medical imaging, robot navigation, and other areas where spatial data are used. The main methods for spatial data analysis include: spatial association rules extraction, clustering, and classification.

Data objects stored in a database are identified by their attributes. Classification finds a set of rules which determine the class of each classified object according to its attributes. Objects with similar attribute values are classified into the same class. For instance, if the unemployment in a city is high and the population of the city is high then the crime rate in that city is high. Spatial classification deals with datasets that contain spatial objects. In spatial classification, attribute values of neighboring objects may also be relevant for the membership of objects in a certain group. This neighborhood factor which is called spatial autocorrelation plays a significant role in spatial data analysis and poses challenges in spatial data classification as well [5]. Traditional data mining methods cannot be effectively applied to spatial data classification since they do not consider spatial autocorrelations among objects. Therefore, there is the need to integrate the existing techniques into new approaches to the spatial data classification task.

Location prediction is a special case of spatial data classification in which we are interested to predict the location. For instance, we may be interested to predict the location of crime hot spots as important part of public safety. The output of the analysis can provide useful information to improve the activities for preventing and detecting safety and security problems. The availability of a location prediction inquiry system can be helpful for public safety experts.

A wide variety of research has considered the use of data mining techniques like Neural Network [6], logistic regression [7] and decision tree to extract patterns from spatial datasets to predict location. However, these previous studies have shown that traditional techniques do a poor job in predicting location task because they do not consider the spatial relations between spatial objects. Several techniques have been proposed to guarantee the spatial dependencies of objects. PLUMS by Shekhar and Chawla [8], is a method for supervised spatial data classification based on using map similarity measure. Spatial Autoregressive Regression technique has also been proposed by spatial statisticians [9].

This paper presents one-class support vector machines (SVMs) based approach [10, 11] to analyze and explore crime datasets. However, the framework we are providing here, can be applied to different domains. In our selected crime datasets, the location is described by Euclidian co-

1-4244-0212-3/06/$20.00/©2006 IEEE

ordinates or Latitude and Longitude. Also each location is identified by its crime rate and several related attributes. We develop a model to automatically classify the locations as either hotspot crime members or non-members. Our system accepts a predefined level of crime rate as input. Based on this value, the system will label a certain portion of the dataset as hotspot members and non-members. If a location's crime rate is above the predefined level of crime, it will be labeled as a member of hotspot crime class or positive sample, otherwise a non-member of hotspot crime or negative sample. The user specifies from the dataset a certain portion to be used as input to the system.

We are using two different approaches for choosing the certain portion of the dataset. After labeling the specified certain percentage of the dataset, the system uses the labeled portion as a training set for building a classifier by applying SVMs. Since we are classifying the locations into positive and negative samples, we can approach this task as a binary classification problem. The system automatically classifies a location with crime rate above (below) the predefined level as positive (negative).

SVMs receive great consideration because of their attractive performance in a wide variety of application domains such as object recognition, speaker identification, face detection, handwriting recognition and text categorization [12]. Generally, SVM solves classification problems by learning from examples. As it is obvious from its name, the binary classification method requires negative and positive examples to establish a statistical relationship and to build a classifier model. However, in reality, many types of datasets suffer from lack of reliable negative examples. This is our main motivation to extend this problem to crime classification. SVM have been recently applied in several studies such as gene expression classification [13], text categorization [14] and text summarization [15].

To demonstrate the effectiveness and applicability of the proposed approach, we report in the paper test results on two datasets downloaded from the internet.

The rest of the paper is organized as follows. Section 2 covers the SVM information required for this study. Experiment methods are discussed in Section 3. The datasets utilized in the experiments are described in Section 4. Detailed explanation on the selected evaluation model, experiments and the results are provided in Section 5 and Section 6, respectively. Section 7 is summary and conclusions.

2 Support Vector Machine

SVMs perform classification, i.e., separate a given known set of +1,-1 labeled training data via a hyper-plane that is maximally distant from the positive samples and negative samples (Optimal Separating Hyper-plane as in Figure 1); plot the test data at the high dimensional space;

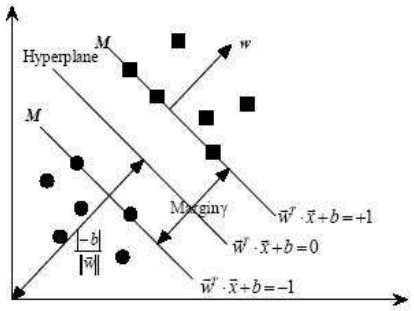

Figure 1. Definition of hyper-plane and margin: circular and square dots are samples of classes -1 and +1, respectively

and distinguish whether each data instance belongs to positive or negative according to the Optimal Separating Hyperplane.

For most real-world problems that seem not to be linearly separable, SVMs can work in combination with the kernel function technique Φ [11], which automatically realizes a non-linear mapping onto a feature space. The Optimal Separating Hyper-plane found by SVM is the feature space that corresponds to a non-linear decision boundary in the input space [11].

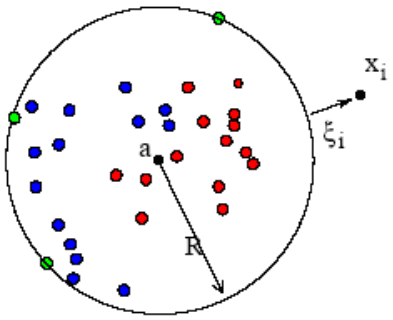

Figure 2. Hypersphere with the target data described by center a and radius R. Three objects on the boundary as the support vectors. One object x_i outside and has $\xi_i > 0$

2.1 One-Class SVM

There are two different approaches to one-class SVMs. The goal of the one-class SVM approach of Tax and Duin [17], which is called the Support Vector Data Description (SVDD), is to find a hypersphere that covers as many

training data points as possible, while keeping the radius of the hypersphere as small as possible. In other words, given l training data points, $x_i,\ (i = 1, 2, ..., l)$, find the smallest possible hypersphere to contain the training points in multi-dimensional space. As shown in Figure 2, small portion of outliers are allowed to exist using a slack variable (ξ_i) [18]:

$$\text{Min}(R^2) + \frac{1}{vl}\sum_i \xi_i \tag{1}$$

Subject to:

$$(x_i - c)^{\mathrm{T}}(x_i - c) \leq R^2 + \xi_i,\ \ \xi_i \geq 0, \quad \forall i \in [1, l]$$

where c and R are the center and radius of the sphere, respectively, T is the transpose, and $v \in (0, 1]$ is the tradeoff between volume of the sphere and the number of training data points rejected. When v is large, the volume of the sphere is small; so more training points will be rejected.

This optimization problem can be solved by the Lagrangian:

$$L(R, \xi, c, a_i, \beta_i) = \quad R^2 + \tfrac{1}{vl}\sum_{i=1}^{l}\zeta_i - \sum_{i=1}^{l} a_i\{R^2 + \zeta_i - (x_i^2 - 2cx_i + c^2)\} - \sum_{i=1}^{l}\beta_i\xi_i \tag{2}$$

where $a_i \geq 0$ and $\beta_i \geq 0$. Setting to *zero* the partial derivative of L with respect to R, a_i, and c, we get:

$$\sum_{i=1}^{l} a_i = 1 \tag{3}$$

$$0 \leq a_i \leq \frac{1}{vl} \tag{4}$$

$$c = \sum_{i=1}^{l} a_i x_i \tag{5}$$

Substituting Eqs. (3)-(5) in Eq. (2) gives the dual problem:

$$\min_a \sum\nolimits_{i,j} a_i a_j (x_i.x_j) - \sum\nolimits_i a_i (x_i.x_j) \tag{6}$$

Subject to:

$$0 \leq a_i \leq \tfrac{1}{vl}, \qquad \sum_{i=1}^{l} a_i = 1$$

Calculating the distance between a test point (x) and the center C of the hypersphere determines whether (x) is inside the sphere or not. By using the following inequality, the position of the test point can be identified:

$$(x.x) - 2\sum_i a_i (x.x_i) + \sum a_i a_j (x_i.x_j) \leq R^2 \tag{7}$$

In reality the data points are not always spherically distributed. So, different types of kernel functions $K(x_i, x_j)$ can be used in order to build more flexible model that can deal with non-linear data. Kernel functions and their usage in building powerful classifiers are described in Section 2.2.

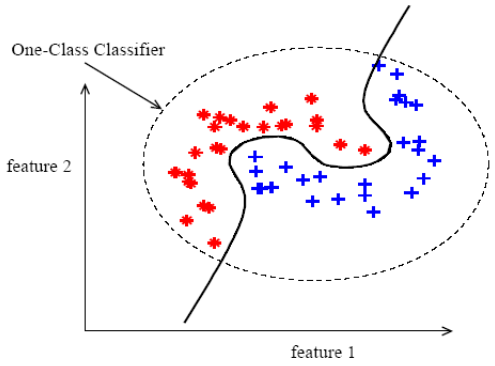

Figure 3. The solid line is the conventional classifier to distinguish between positives and negatives, while the dashed line describes the dataset.

The second approach in one-class SVMs was proposed by Scholkopf et al [16]. Their approach is to construct a hyperplane that is maximally distant from the origin with all data lying on one side from the origin as shown in Figure 3. In other words, given training set $x_1, \ldots, x_l \in R^N$, where x_i is a feature vector, it is required to estimate a function that takes the value +1 in a small region capturing most of the data points, and -1 elsewhere [14]. Formally, the function is written as:

$$f(x) = \begin{cases} +1 & \text{if } x \in S \\ -1 & \text{if } x \in \bar{S} \end{cases} \tag{8}$$

where S and $\bar{S}$ are simple subsets of the input space and its complement, respectively. Let $\Phi : \Re^N \rightarrow F$ be a nonlinear mapping that maps the training data from $\Re^N$ to a feature space F. To separate the data set from the origin, solve the following primal optimization problem [15]:

$$\text{Minimize} \quad V(\mathrm{w}, \xi, \rho) = \frac{1}{2}\|w\|^2 + \frac{1}{vl}\sum_{i=1}^{l}\xi_i - \rho \tag{9}$$

Subject to:

$$(\mathrm{w}.\Phi(\mathrm{x}_i)) \geq \rho - \xi_i, \xi_i \geq 0,$$

where $v \in (0, 1)$ is a parameter for controlling the tradeoff between the number of outliers and model complexity, and ρ is the margin. Using the following decision function, a label can be assigned to a new given data point (x) for classification task:

$$f(x) = sgn(w.\Phi(x_i) - \rho) \tag{10}$$

Again same as the former approach, introducing Lagrange multipliers α_i and using the Kuhn-Tucker condition, the derivatives with respect to the primal variables are set to zero to get:

$$w = \sum_i \alpha_i \Phi(x_i) \tag{11}$$

where only a subset of points x_i that are closest to the hyperplane have nonzero values α_i. These points are called support vectors. Instead of solving the primal optimization problem directly, the following dual problem can be considered:

$$\text{Maximize} \quad w(\alpha) = \frac{1}{2} \sum\nolimits_{i,j} \alpha_i \alpha_j K(x_i, x_j) \tag{12}$$

Subject to:

$$0 \leq \alpha_i \leq \tfrac{1}{vl}, \sum_i \alpha_i = 1$$

From Eqs. (10) and (11), the decision function can be eventually written as:

$$f(x) = sgn\left(\sum_i \alpha_i K(x_i, x_j) - \rho\right) \tag{13}$$

$K(x_i, x_j) = (\Phi(x_i), \Phi(x_j))$ are the kernels (the dot products between mapped pairs of input points). Kernel functions allow more general decision functions when the data points are not linearly separable.

2.2 Kernel Functions

The idea of the kernel function is to enable operations to be performed in the feature space rather than the potentially high dimensional input space. Hence, the inner product does not need to be evaluated in the feature space. This provides a way of addressing the curse of dimensionality. However, the computation is still critically dependent upon the number of training patterns, and to provide a good data distribution for a high dimensional problem will generally require a large training set [19].

The kernel theory is based upon Reproducing Kernel Hilbert Spaces (RKHS) [20, 21]. An inner product in the feature space has an equivalent kernel in the input space, $K(x, x') = (\phi(x), \phi(x'))$, provided certain conditions hold. If K is a symmetric positive definite function, which satisfies Mercer's Conditions,

$$K(x, x') = \sum_m^\infty a_m \phi_m(x) \phi_m, a_m \geq 0 \tag{14}$$

$$\int \int K(x, x') g(x) g(x') dx dx' > 0, g \in L_2 \tag{15}$$

then the kernel represents a legitimate inner product in the feature space. There are many kinds of valid functions that satisfy Mercer's conditions. Since two types of these functions are often used for classification problems; polynomial and Gaussian kernels; here we limit ourselves to these two kernel functions.

A polynomial mapping is a popular method for nonlinear modeling:

$$K(x, x') = \langle x, x' \rangle^d \tag{16}$$

$$K(x, x') = (\langle x, x' \rangle + 1)^d \tag{17}$$

The second kernel is usually preferable as it avoids problems with the Hessian becoming zero.

Radial basis functions have received significant attention, most commonly with a Gaussian of the form:

$$K(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) \tag{18}$$

Classical techniques utilizing radial basis functions employ some method of determining a subset of centers. Typically, a method of clustering is first employed to select a subset of centers. An attractive feature of the SVM is that this selection is implicit, with each support vector contributing one local Gaussian function, centered at that data point.

3 Experiment Methods

In order to select a certain representative portion of the crime datasets to be used as the training set by the system, we experiment the following approach: for a given percentage of the data and a predefined level of crime rate, we select a subset of the crime dataset to label; and then based on the predefined level of crime rate, we specify a class label to each data point in the selected set. The data points which have the crime rate above the predefined rate are positive or members of hotspot class and data points with crime rate below the predefined rate are negative or non-members of hotspot class. Then this labeled data set will be used as the training set in SVM classification. To select a given percentage of the data to be labeled, we use the k-median clustering algorithm. Then, we compare the result when the same percentage of the data is selected randomly.

3.1 K-Mean Clustering Algorithm

K-means clustering is a partitioning method that partitions the data points of the input dataset into k clusters. Each data point in the dataset is treated by k-means algorithm as an object having a location in space. The algorithms finds a partition in which objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible. Each cluster is defined by its data members and its centroid. The centroid for each

cluster is the point to which the sum of distances from all objects in that cluster is minimized.

In our experiment, the idea of using k-means clustering is to consider the data points within a certain distance from the centroids of the k clusters as representative data points of the clusters. The size of k should be chosen in a way by the system such that a certain total percentage of the data set falls into a desired distance from a closest cluster center. By gathering a small set of data points from each cluster based on the above procedure, we will have a portion of the whole data set ready to be labeled as the training set. Points having their crime rate above the predefined level of crime are labeled as members of hotspot class and points having crime rate less than the predefined level are labeled as non-members of hotspot class.

To compare the clustering approach and random approach for choosing a portion of the data set as the training set, we devise the following experimental plans:

1. **Random Selection for Labeling + One-Class SVM:** we randomly select a given percentage of the data points as a small representative portion of the crime dataset to be labeled. Then, the one-class SVM algorithm uses the output labeled set as the training set to build the classifier.

2. **Clustering-based Selection for Labeling + One-Class SVM:** k-means is used to select a given percentage of the data points to be labeled. Then, the one-class SVM algorithm uses the output labeled set as the training set to build the classifier.

3. **Complete Data Set + One-Class SVM:** after choosing a certain percentage of the dataset for labeling by random selection or clustering technique, we label the rest of the dataset as negative samples and add them to the training set. Then, we pass the complete labeled dataset to one-class SVM as the training set.

4 Datasets

To test the different approaches used by our model, we downloaded two published crime datasets from the internet. The datasets consist of the crime rate and related variables for each data point. The location of each data point is described in the dataset by Euclidian coordinates or Latitude and Longitude. The datasets were downloaded from [http://www.terraseer.com/]. In this section, we will provide short description for each dataset. For further information, please, refer to [http://www.terraseer.com/].

The first dataset is a small crime dataset [22] that records crime rate and 20 related variables in 49 neighborhoods in Columbus Ohio, USA (See Figure 4a and 4b). The problem is to distinguish between members and non-members of crime hotspot class. Hotspot crime locations are places which have the crime rate above the predefined crime level.

The second dataset [23] records 78 counties surrounding St. Louis, MO homicides rate and related variables (see Figure 4c and 4d). The problem here is to distinguish between members and non-members of homicide hotspot class. Hotspot homicide locations are places which have the homicide rate above the predefined level of homicide.

5 Model Evaluation

Basically, n-fold cross validation is a method in which the data is randomly divided into n disjoint groups [24]. For example, suppose the data is divided into ten groups. The first group is set aside for testing and the other nine are put together for model building. The model built on the 90% group is then used to predict the group that was set aside. This process is repeated a total of 10 times as each group in turn is set aside. Finally, a model is built using all the data. The mean of the 10 independent error rate predictions is used as the error rate for this final model. In our study, a five-fold cross validation method has been used to estimate the accuracy of the classification model.

6 Experiment and Result

For our experiments we used Personal Computer with Intel P4 2.4GHZ CPU and 1GB memory. The experiments were carried out by using a Matlab interface of LIBSVM [25] in Matlab 7. LIBSVM is a library for SVM classification and regression. The predefined level of crime depends on the knowledge of domain expert and is usually specified by crime experts. In this experiment, we assume that the data points of our datasets follow a normal distribution (Gaussian distribution) so that the optimal average value of the crime C is halfway between $C_{\min}$ and $C_{\max}$.

According to the definition of the Gaussian distribution C_{average} and $C_{\max}$ are the mean ("average") and the mean incremented by standard deviation ("variability"), respectively. In our first set of experiments, we considered C_{average} as the predefined level of crime rate, i.e., if the crime rate of a sample location is above C_{average}, then that location is a member of hotspot class; otherwise it would be a non-member of the hotspot class. In the second set of the experiments, $C_{\max}$ was considered as the predefined value for hotspots crime rate.

We selected 20% of the datasets to be labeled and used as training data in the SVM algorithm. In our experiments, we also take into account the case in which we assign the remaining 80% unlabeled data to negative class or hotspot non-member class, and give the complete labeled data set to the SVM to be trained. Then, we compare the result with the

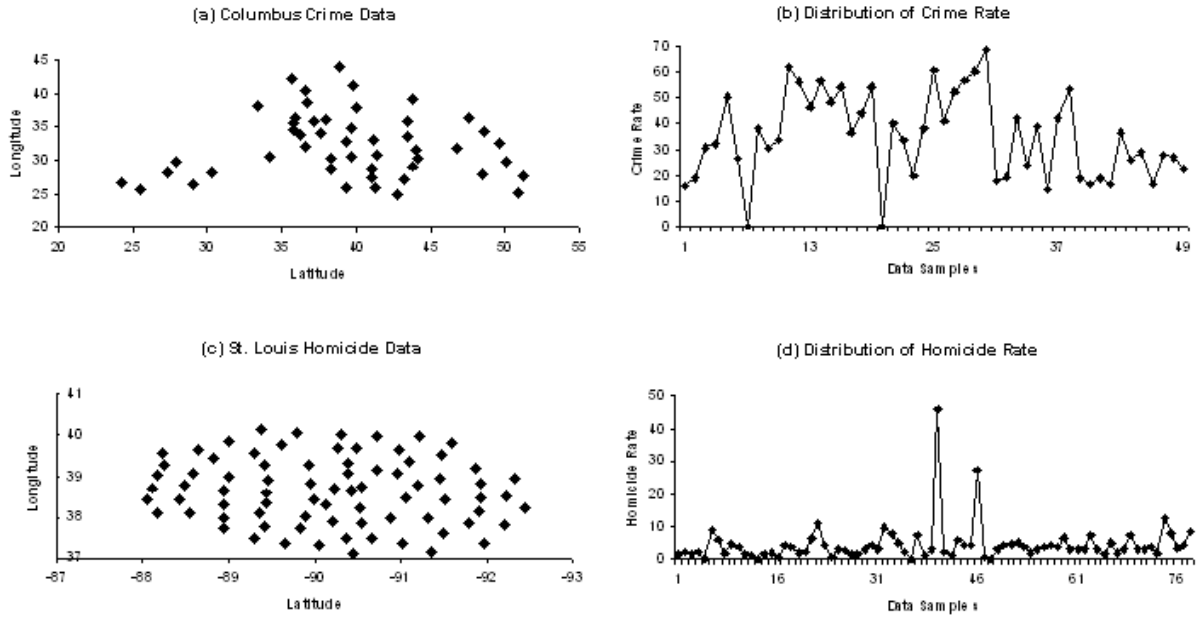

Figure 4. (a) Crime data and (b) its rate distribution; (c) Homicide data and (d) its rate distribution.

case where we ignore the remaining 80% unlabeled data. In our first approach to data selection, which is random data selection, we selected 20% of the dataset randomly. In order to get a more consistent result, we performed this experiment 20 times.

In the clustering-based data selection approach, we applied k-mean clustering algorithm to the datasets first. This second approach of data selection chooses the data more wisely than the random selection. After labeling the data, we used the result set as input to the SVM algorithm. In one-class SVMs, we applied different kernel functions to see how it would influence the classification accuracy. The results for four different experiments are shown in Tables 1-4. As it can be seen in each experiment, first we evaluate the effect of using different kernel function for one-class SVM technique. Here we have chosen Linear, Polynomial and Gaussian kernel functions. We applied the default values of LIBSVM for Polynomial and Gaussian. We also changed the parameter in a range to see whether the result will be improved or not. We also performed each step 20 times and presented the average of 20 runs as the final result.

All the results from different techniques are expressed as the percentages of correctly classified samples on the test data. The values of C are set to the mean (μ) and the mean plus the standard deviation ($\mu + s$) calculated for each experimented dataset.

In Table 1, the result is displayed when 20% of the dataset is selected randomly and labeled. This portion of data is passed to SVM as the training set. Then, we run the SVM classifier with different kernel functions on the data set; Gaussian kernel performs better. Based on our knowledge from the Columbus dataset, when we set the predefined crime rate to mean value, number of positive samples will be more than the case when we set the predefined value C to mean plus standard deviation. Therefore, we can say that one-class SVM performs better when we have a small training set with more positive samples.

Table 1. 20% of the dataset selected randomly and labeled.

Data Set	C	One-Class SVM		
		Linear	Polynomial	Gaussian
Columbus	35.13	63.00	60.50	68.50
	51.86	58.50	57.50	62.50
St. Louis	4.57	53.13	53.44	49.69
	10.58	50.63	50.94	48.44

In Table 2, the result is displayed when 20% of the dataset is selected randomly and labeled, and the rest is labeled as negative samples and added to the training set. The difference between this experiment and the one shown in Table 1 is that we have increased the size of the training set by adding more negative samples. As it can be seen in Table 2, one-class SVM will result in a noticeably decreasing

Table 2. 20% of the dataset selected randomly and labeled; the rest 80% labeled as negative samples and added to the training set.

Data Set	C	One-Class SVM		
		Linear	Polynomial	Gaussian
Columbus	35.13	52.45	51.43	57.04
	51.86	51.22	51.53	55.10
St. Louis	4.57	49.94	49.62	38.65
	10.58	50.39	50.06	38.46

classification performance as we have a larger training set with more negative samples.

Table 3. 20% of the dataset selected by clustering and labeled.

Data Set	C	One-Class SVM		
		Linear	Polynomial	Gaussian
Columbus	35.13	63.89	61.11	69.44
	51.86	78.33	76.11	65.55
St. Louis	4.57	53.44	54.38	50.94
	10.58	51.25	51.56	24.38

In Table 3, the result is displayed when 20% of the dataset is selected by clustering and labeled. Then this portion of data is passed to SVM as the training set and we run the SVM classifier with different kernel functions on the data set. Compared to Table 1, one-class SVM performs much better when we use clustering instead of random selection. As a matter of fact, by clustering technique we choose the data more wisely than we do in random selection. The size of k will be chosen by the system such that we will be able to select 20% of the original data set from the different clusters. This means we will select 20% of the dataset within a certain distance from the closest center of each cluster and then we put them all together and label them as the training set. Same as our first experiment, we see that one-class SVM with Linear and Gaussian kernels performs better when C is set to the mean value of crime rate, which means we have a small training set with more positive samples.

Based on the results from our previous experiments, we can predict the expected result from our last experiment where 20% of the dataset is selected by clustering and labeled and the rest 80% is labeled as negative sample and added to the training set. The result is displayed in Table 4. One-class SVM performs better compared to Table 2, since we are using clustering instead of random selection. However, it will do a poor job compare to Table 3, where we have a smaller dataset with more positive sample than we do in Table 4.

To sum up, one-class SVMs is attractive for two main

Table 4. 20% of the dataset selected by clustering and labeled; the rest 80% labeled as negative samples and added to the training set.

Data Set	C	One-Class SVM		
		Linear	Polynomial	Gaussian
Columbus	35.13	53.47	53.06	57.35
	51.86	53.37	53.37	57.76
St. Louis	4.57	51.47	51.09	39.23
	10.58	50.51	50.64	39.31

advantages. First, one-class SVMs construct a hyperplane that is maximally distant from the origin with all data lying on the opposite side from the origin. This helps to find an optimal hypersphere which contains all or most of the training points which belong to the positive samples. Furthermore, using different types of kernel functions gives an ability to one-class SVMs so that it represents various data distribution shapes in feature space (e.g., spherical shapes or very irregular shapes). Second, one-class SVMs make no assumption on the probability density of the data. This is a useful benefit when the data do not follow any probability distribution (such as a normal distribution), or insufficient data are available to test the distribution [18].

7 Summary and Conclusions

In this study, we provided an inquiry system which can be used as a general framework in different domains and by domain experts in order to customize the spatial data classification task. In our framework, we focused on some special types of spatial datasets like the ones we used for our experiments. As a case study in the area of public safety, we concentrated on the performance of one-class SVMs for predicting the hotspot crime location when a predefined level of crime rate and a percentage for selecting a portion of that are given. We applied two different approaches for data selection, first we chose the certain portion of the data randomly and in the second approach we applied k-means clustering algorithm in order to make a more wise selection. Then we labeled the selected portion of the data set as members and non-members of crime hotspot class based on the predefined level of crime rate. We also studied the case when the rest of the dataset were labeled as non-members samples and added to the training set. Our experiments show that one-class SVM gives reasonable result when we choose appropriate parameters for the algorithm. Based on our different experiences shown in the result tables, we can conclude that SVMs form an appropriate approach to hotspot crime prediction, while k-means clustering algorithm is useful for data selection and by using the rest of the dataset as non-member samples. This moti-

vates us to investigate the same problem by using two-class SVMs; we expect to achieve much better results. Our plan is to compare one-class and two-class SVMs with the other approaches to hotspot crime prediction.

References

[1] M.H. Dunham, *Data Mining: Introductory and Advanced Topics*, Prentice Hall, 2002.

[2] W. Lu, J. Han, and B. C. Ooi, "Discovery of General Knowledge in Large Spatial Databases," *Proc. of Far East Workshop on Geographic Information Systems,* pp.275-289, Singapore, June 1993.

[3] K. Koperski, J. Adhikary, and J. Han, "Spatial Data Mining: Progress and challenges," *Proc. of ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery,* pp.55-70, Montreal, 1996.

[4] K. Koperski and J. Han, "Discovery of Spatial Association Rules in Geographic Information Databases," *Proc. of the international Symp. on Large Spatial Databases,* pp.47-66, Portland, Maine, 1995.

[5] S. Shekhar, et al, "Trend in Spatial Data Mining, to appear in Data Mining: Next Generation Challenges and Future Directions," in H. Kargupta, A. Joshi, K. Sivakumar, Y. Yesha (eds.), AAAI/MIT Press, 2003.

[6] S. Ozesmi and U. Ozesmi, "An Artificial Neural Network Approach to Spatial Habitat Modeling with Interspecific Interaction," *Ecological Modeling,* (116):15-31, 1999.

[7] U. Ozesmi and W. Mitsch, "A Spatial Habitat Model for Marsh-Breeding Red-Winged Black Bird," *Ecological Modeling,* (101):139-152, 1997.

[8] S. Chawla, S. Shekhar, W. Wu, "Predicting Locations Using Map Similarity (PLUMS): A Framework for Spatial Data Mining," *Proc. of ACM International Conference on Knowledge Discovery and Data Mining,* Boston, MA, 2000.

[9] J. Lesage, "Regression Analysis of Spatial Data," *The Journal of Regional Analysis and Policy,* Vol.27, No.2, pp.83-94, 1997.

[10] B. E. Boser, I. M. Guyon and V. Vapnik, "A Training Algorithm for Optimum Margin Classifiers," *Proc. of the Annual Workshop on Computational Learning Theory,* Pittsburgh. ACM, 1992.

[11] V. Vapnik, *Statistical Learning Theory,* John Wiley, NY, p.732, 1998.

[12] N. Cristianini and J. S. Taylor, *An Introduction to Support Vector Machines,* Cambridge University Press, 2000.

[13] A. Kowalczyk and B. Raskutti, "One Class SVM for Yeast Regulation Prediction," *SIGKDD Explorations,* Vol.4, pp.99-100, 2002.

[14] L. M. Manevitz and M. Yousef, "One-Class SVMs for Document Classifcation," *Journal of Machine Learning,* 2, pp.139-154, 2001.

[15] C. Kruengkrai and C. Jaruskulchai, "Using One-Class SVMs for Relevant Sentence Extraction," *Proc. of the International Symposium on Communications and Information Technologies,* 2003.

[16] B. Scholkopf, et al, "Estimating The Support of a High Dimensional Distribution," *Technical report,* Microsoft Research, MSRTR9987, 1999.

[17] D. M. Tax and R. P. Duin, "Outliers and Data Descriptions," *Proc. of the Annual Conference of the Advanced School for Computing and Imaging,* 2001.

[18] Q. Gou, M. Kelley and C. H. Graham, "One-Class Support Vector Machines for Predicting Distribution of Sudden Oak Death in California," *Ecological Modeling,* 182:75-90, 2005.

[19] V.N. Vapnik, *The Nature of Statistical Learning Theory,* Second Edition, Springer, New York, 1999.

[20] N. Aronszajn, "Theory of Reproducing Kernels," *Trans. Am. Math. Soc.,* Vol.68, pp.337-404, 1950.

[21] G. Wahba, "Spline Models for Observational Data," *Proc. of SIAM CBMS-NSF Regional Conference Series in Applied Mathematics,* v. 59, 1990.

[22] L. Anselin, *Spatial Econometrics: Methods and Models,* Dordrecht: Kluwer Academic, Table 12.1 p. 189, 1998.

[23] S. Messner, et al, "The Spatial Patterning of County Homicide Rates: An Application of Exploratory Spatial Data Analysis," *Journal of Quantitative Criminology,* Vol.15, No.4, pp.423-450, 1999.

[24] H. A. Edelstein, *Introduction to Data Mining and Knowledge Discovery,* Third Edition, Potomac, MD: Two Crows Corp, 1999.

[25] C. C. Chang and C. J. Lin, "LIBSVM: A Library for Support Vector Machines," 2001. URL: http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[26] J. P. LeSage, "MATLAB Toolbox for Spatial Econometrics," 1999. URL: http://www.spatial-econometrics.com.