Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 07-004

NORTHSTAR: A Parameter Estimation Method for the Spatial Autoregression Model

Mete Celik, Baris M. Kazar, Shashi Shekhar, Daniel Boley, and David J. Lilja

February 09, 2007

NORTHSTAR: A Parameter Estimation Method for the Spatial Autoregression Model

Mete Celik, Baris M. Kazar, Shashi Shekhar, Daniel Boley, David J. Lilja

Abstract

Parameter estimation method for the spatial autoregression model (SAR) is important because of the many application domains, such as regional economics, ecology, environmental management, public safety, transportation, public health, business, travel and tourism. However, it is computationally very expensive because of the need to compute the determinant of a large matrix due to Maximum Likelihood Theory. The limitation of previous studies is the need for numerous computations of the computationally expensive determinant term of the likelihood function. In this paper, we present a faster, scalable and **NO**vel p**R**ediction and estimation **T**ec**H**nique for the exact **S**pa**T**ial **A**uto **R**egression model solution (NORTHSTAR). We provide a proof of the correctness of this algorithm by showing the objective function to be unimodular. Analytical and experimental results show that the NORTHSTAR algorithm is computationally faster than the related approaches, because it reduces the number of evaluations of the determinant term in the likelihood function.

Index Terms

Spatial Autoregression Model, Spatial Autocorrelation, Spatial Data Mining, Spatial Databases, Maximum Likelihood Theory.

This work was partially supported by the Army High Performance Computing Research Center (AHPCRC) under the auspices of the Department of the Army, Army Research Laboratory (ARL) under contract number DAAD19-01-2-0014, and the NSF grant IIS-0534286. This work received additional support from the University of Minnesota Digital Technology Center and the Minnesota Supercomputing Institute.

M. Celik is with the CS Department, University of Minnesota, Minneapolis, MN, 55455. E-mail: mcelik@cs.umn.edu

B. M. Kazar is with the Oracle Corporation, Nashua, NH. E-mail:baris.kazar@oracle.com

S. Shekhar is with the CS Department, University of Minnesota, Minneapolis, MN, 55455. E-mail: shekhar@cs.umn.edu

D. Boley is with the CS Department, University of Minnesota, Minneapolis, MN, 55455. E-mail: boley@cs.umn.edu

D. J. Lilja is with the ECE Department, University of Minnesota, Minneapolis, MN, 55455. E-mail: lilja@ece.umn.edu

I. Introduction

Given a spatial framework, observations on a dependent variable, a set of explanatory variables, and neighborhood relationships (spatial dependencies) among the spatial data, SAR parameter estimation based on Maximum Likelihood Theory (ML) aims to find the optimum SAR model parameters by minimizing the likelihood function of the SAR model solution.

The massive sizes of geo-spatial datasets in many application domains make it important to develop scalable parameter estimation algorithms of the SAR model solutions for location prediction and classification. These application domains include regional economics [24], ecology [9], [40], environmental management [19], public safety [21], transportation [41], public health [43], business, travel and tourism [1], [39], [38]. For example, predicting the locations of the bird nests in a wetland is a location prediction problem. In this example dependent variable can be bird nest location and explanatory variables can be vegetation durability, water depth, vegetation distribution, etc. Initially classical prediction model, e.g., linear regression was used for this problem [29]. However, it yielded low prediction accuracy [29] because the autocorrelation in spatial data violates the *independently and identically distributed (i.i.d.)* assumption that underlies linear regression. SAR improved prediction accuracy in this problem [9], [40].

However, estimation of the SAR model parameters is computationally very expensive because of the need to compute the determinant of a large matrix in the likelihood function. The Maximum Likelihood function for SAR parameter estimation contains two terms, namely a determinant term and *SSE* term. The former involves computation of the determinant of a very large matrix, which is a well-known hard problem in numerical analysis. For example, the exact SAR model parameter estimation for a 10,000-point spatial problem can take tens of minutes on common desktop computers. Computation costs make it difficult to use SAR for many important spatial problems which involve millions of points. Because of the high cost of determinant computation, the use of the SAR model has been limited to small problem sizes, despite its promise to improve prediction and classification accuracy.

Previous approaches compute the determinant term of a large matrix of the SAR model solution repeatedly to determine the Maximum Likelihood values of SAR parameters, namely, an autocorrelation parameter and weights for explanatory variables [26], [32], [33], [34], [22], [37]. For example, they find the optimum spatial autocorrelation parameter using iterative search

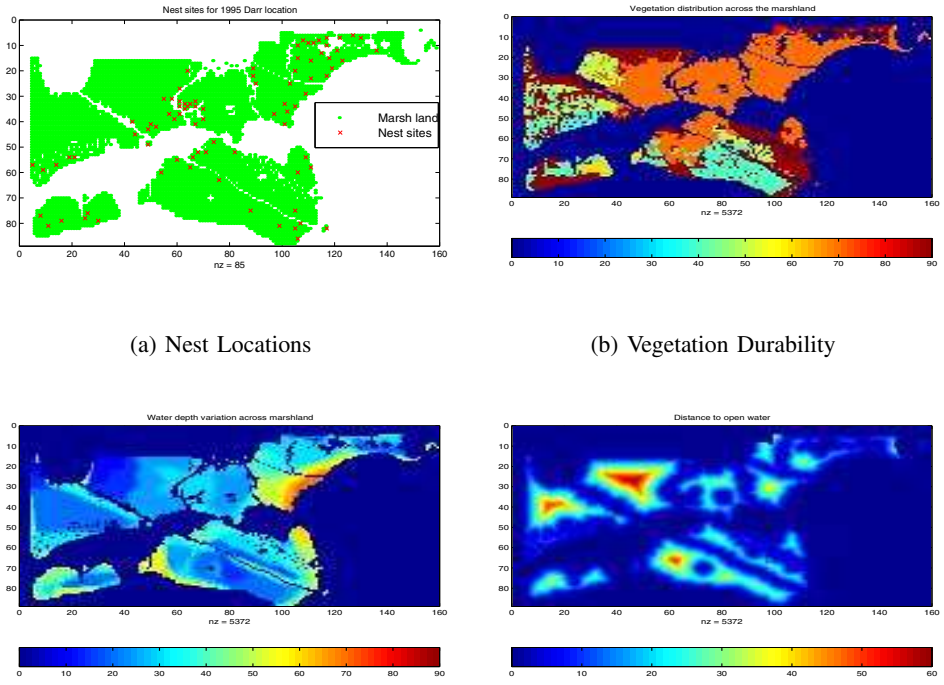
methods, (e.g., golden section search) in the interval of possible values (e.g., [0,1]).

In contrast, our approach yields a reduction in computation cost by reducing the number of determinant computations of a very large matrix. The key idea is to narrow the search interval by a cheap computation yielding an upper bound on the spatial autocorrelation parameter (Lemma 4.3). Recall that the ML-based SAR model solution contains two terms, a determinant term and an *SSE* term. Both terms involve the spatial autocorrelation parameter and both terms are unimodular in the autocorrelation function (Theorem 4.1). In addition, the location of the autocorrelation parameter that minimizes the *SSE* term of the ML function is an upper bound on the autocorrelation parameter that optimizes the likelihood function. This upper bound allows us to narrow the search interval and reduce the number of iterations and number of determinant evaluations of an iterative search to estimate the spatial autocorrelation parameter of the SAR model. Of course, there is a trade-off between the extra computations to determine the upper bound and the savings from reduced number of iterations. If the overhead of determining the upper bound is much smaller than the resulting savings, then our approach is computationally more efficient than the previous approaches.

The paper evaluates the proposed approach analytically and experimentally. Analytical and experimental results show that the proposed approach is computationally more efficient than the previous work. We analyzed that the evaluation of the *SSE* term of the ML function gives an upper bound on the autocorrelation parameter of the likelihood function by using SAR Unimodularity Theorem 4.1 and Lemma 4.3. Experimental results show that the computational cost of the proposed approach is usually smaller than the cost of related approaches. The experiments show that the proposed approach is computationally more efficient than the related approaches in terms of execution time and memory usage. In addition, when the value of the autocorrelation parameter decreases, the advantage of the proposed approach increases. It is also observed that determinant computation saving increases for the bigger neighborhood structures.

A. An Illustrative Application Domain

We now introduce an example which will be used throughout this paper to illustrate the different concepts in spatial data mining. We are given data about two wetlands, named Darr and Stubble, on the shores of Lake Erie in Ohio USA in order to *predict* the spatial distribution of a marsh-breeding bird, the red-winged blackbird (*Agelaius phoeniceus*). The data was collected



(a) Nest Locations

(b) Vegetation Durability

(c) Water Depth

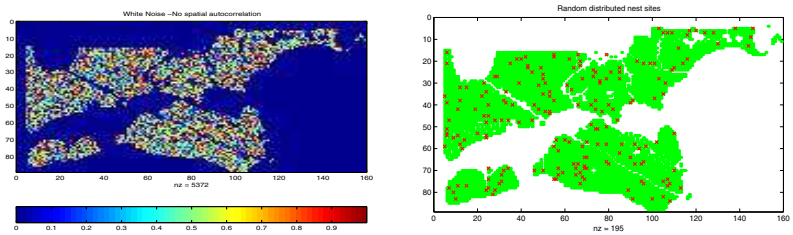
(d) Distance to Open Water

Fig. 1. (a) The geometry of the wetland and the locations of the nests, (b) The spatial distribution of *vegetation durability* over the marshland, (c) The spatial distribution of *water depth*, and (d) The spatial distribution of *distance to open water.*

from April to June in two successive years, 1995 and 1996 [29].

A uniform grid was imposed on the two wetlands and different types of measurements were recorded at each cell or pixel. In total, values of seven attributes were recorded at each cell. Domain knowledge is crucial in deciding which attributes are important and which are not. For example, *Vegetation Durability* was chosen over *Vegetation Species* because specialized knowledge about the bird-nesting habits of the red-winged blackbird suggested that the choice of nest location is more dependent on plant structure and plant resistance to wind and wave action than on the plant species.

For simplicity, we focus on three independent attributes, namely *Vegetation Durability*, *Distance to Open Water*, and *Water Depth*. The significance of these three variables was established using classical statistical analysis. The spatial distribution of these variables and the actual nest locations for the Darr wetland in 1995 are shown in Figure 1. These maps illustrate the following



(a) Pixel Property with Independent Identical Distribution

(b) Random Nest Locations

Fig. 2. Spatial distribution satisfying random distribution assumptions of classical regression

two important properties inherent in spatial data.

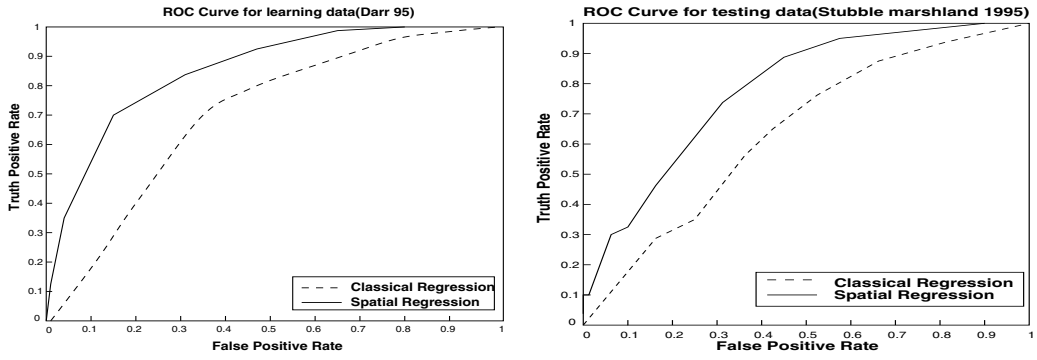
1) The value of attributes which are referenced by spatial location tend to vary gradually over space. While this may seem obvious, classical data mining techniques, either explicitly or implicitly, assume that the data is *independently* generated. For example, the maps in Figure 2 show the spatial distribution of attributes if they were independently generated. Ozesmi et al. has applied classical data mining techniques like logistic regression [29] and neural networks [28] to build spatial habitat models. Logistic regression was used because the dependent variable is binary (nest/no-nest) and the logistic function "squashes" the real line onto the unit-interval. The values in the unit-interval can then be interpreted as probabilities. The study concluded that with the use of logistic regression, the nests could be classified at a rate 24% better than random [28].
2) The spatial distributions of attributes sometimes have distinct local trends which contradict the global trends. This is seen most vividly in Figure 1(b), where the spatial distribution of *Vegetation Durability* is jagged in the western section of the wetland as compared to the overall impression of uniformity across the wetland. This property is called spatial heterogeneity.

Classification accuracy achieved by classical and spatial regression are compared on the test data. Receiver Operating Characteristic (ROC) [14] curves can be used to compare classification accuracy. ROC curves plot the relationship between the true positive rate (TPR) and the false positive rate (FPR). For each cut-off probability b, TPR(b) measures the ratio of the number

of sites where the nest is actually located and was predicted divided by the number of actual nest sites. The FPR measures the ratio of the number of sites where the nest was absent but predicted, divided by the number of sites where the nests were absent. The ROC curve is the locus of the pair (TPR(b), FPR(b)) for each cut-off probability. The higher the curve above the straight line TPR=FPR, the better the accuracy of the model.

Figure 3(a) illustrates the ROC curves for spatial autoregression regression (SAR) and classical regression models built using the real surveyed Darr95 learning data and Figure 3(b) displays the ROC curve for the real Stubble test data [39]. It is clear that using spatial regression resulted in better predictions at all cut-off probabilities relative to the classical regression model.

Clearly, by including a spatial autocorrelation term, there is substantial and systematic improvement for all levels of cut-off probability on both the learning data (1995 Darr) and test data (1995 Stubble).



(a) ROC curves for learning

(b) ROC curves for testing

Fig. 3. (a) Comparison of the classical regression model with the spatial autoregression model on the Darr learning data. (b) Comparison of the models on the testing data.

B. Problem Statement

Given a spatial framework S for the underlying spatial graph G, and the attribute functions $f_{\mathbf{x}_k}$ over S, and the neighborhood relationship R, we can build the SAR model and find its parameters by minimizing the objective (log-likelihood) function as can be seen in (1).

$$\ell(\rho|\mathbf{y}) = \frac{-2}{n}\underbrace{\ln|\mathbf{I}-\rho\mathbf{W}|}_{log-det} + \underbrace{\ln((\mathbf{I}-\rho\mathbf{W})\mathbf{y})^T(\mathbf{I}-\mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1}\ \mathbf{x}^T)^T(\mathbf{I}-\mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1}\ \mathbf{x}^T)((\mathbf{I}-\rho\mathbf{W})\mathbf{y})}_{SSE} \quad (1)$$

The details of the derivation of the log-likelihood function for the ML-based SAR model is given in Appendix VI.

The problem of parameter estimation of the SAR model using Maximum Likelihood Theory (ML) is formally defined as follows:

Given:

- A spatial framework S consisting of sites $\{s_1, ..., s_n\}$ for an underlying geographic space G.
- A collection of explanatory functions $f_{\mathbf{x}_k} : S \rightarrow R^k$, $k = 1, ..., K$. R^k is the range of possible values for the explanatory functions.
- A dependent function $f_{\mathbf{y}} : S \rightarrow R^{\mathbf{y}}$.
- A family $\mathbf{F}$ (i.e., $\mathbf{y} = \rho\mathbf{W}\mathbf{y} + \mathbf{x}\beta + \epsilon$) of learning model functions mapping $R^1 \times ... \times R^K \rightarrow R^{\mathbf{y}}$.
- A neighborhood relationship on the spatial framework.

Find:

- The SAR scalar parameters ρ and the regression coefficient vector β

Objective:

- Minimizing the objective function, log-likelihood function $\ell(\rho|\mathbf{y})$ given in (1), of the ML-based SAR model solution.

Constraints:

- Geographic space S is a multi-dimensional Euclidean Space.
- The values of the explanatory functions, the $f_{\mathbf{x}_k}$'s and the response function $f_{\mathbf{y}}$ may not be independent with respect to those of nearby spatial sites, i.e., spatial autocorrelation exists.
- The domain R^k of explanatory functions is the one-dimensional domain of real numbers.
- The domain of the dependent variable, $R^{\mathbf{y}} = \{0, 1\}$.
- The SAR parameter ρ varies in the range $[0, 1)$.

- The error is normally distributed (Gaussian error), i.e., $\epsilon \sim N(0, \sigma^2 \mathbf{I})$ *IID*. In other words, the error is composed of normally distributed random numbers with unit standard deviation and zero mean.
- The neighborhood matrix $\mathbf{W}$ exhibits sparsity.

For the bird location prediction example, dependent variable **y** can be the locations of the nests. Explanatory variables **x** can be independent variables, namely observations of *Vegetation Durability*, *Water Depth*, and *Distance to Open Water*. Neighborhood matrix **W** represents the spatial dependencies of the neighboring locations. In the **W** matrix, neighboring locations are represented by 1s, and the rest of the matrix contains a value of zero.

TABLE I

THE NOTATION USED IN THIS STUDY

Variable	Definition	Variable	Definition
ρ	The spatial autoregression (autocorrelation) parameter	$\mathbf{I}$	Identity matrix
$\mathbf{y}$	n-by-1 vector of observations on the dependent variable	ϵ	n-by-1 vector of unobservable error
$\mathbf{x}$	n-by-k matrix of observations on the explanatory variable	b_l	Lower bandwith of the neighborhood matrix
$\mathbf{W}$	n-by-n neighborhood matrix that accounts for the spatial relationships (dependencie s) among the spatial data	b_u	Upper bandwith of the neighborhood matrix
k	Number of features	tol	Tolerance value
β	k-by-1 vector of regression coefficients	λ	Eigenvalue of a matrix
n	Problem size (number of observation points or pixels)	σ^2	The common variance of the error ϵ

C. Related Work

The Maximum Likelihood Theory (ML) is used in order to estimate the SAR model parameters. The ML function (log-likelihood) of the SAR model solution given in (1) is computed by calculating the maximum of the sum of the logarithm of the determinant (log-det term) of a

large matrix and a sum-of-squared errors (*SSE*) term [26], [32], [33], [34], [22], [37]. The ML-based SAR model solutions can be classified into two categories, exact SAR model solutions and approximate SAR model solutions, due to the strategy used to calculate the log-det term of a large matrix. This paper focuses on ML-based exact SAR model solutions.

To estimate the parameters of a ML-based SAR model solution, the log-likelihood function can be constructed, as shown in (1). The details of the derivation of the log-likelihood function for the ML-based SAR model is given in Appendix VI. The log-likelihood function of the ML-based SAR model solution basically contains two terms, namely, a log-det term and an *SSE* term as can be seen in (1). The estimation procedure involves computation of the logarithm of the determinant of (log-det) a large matrix, i.e. $(\mathbf{I} - \rho\mathbf{W})$. Computing the determinant of a matrix is very expensive.

In the literature, there are two ML-based exact SAR model solutions, an eigenvalue computation (EV) based solution [26] and a direct (straight) sparse log-det (SLD) based solution [30]. The EV-based SAR model solution uses dense data structures to find the determinant of a very large matrix. Because of the dense representation of the matrices in the EV-based approach, LU factorization of a large matrix requires O(n^3) operations, where n is the number of observations. LU factorization is used to compute determinant of the large matrix [13], [16]. This leads to high execution time and memory usage. In the SAR formulation, neighborhood matrix $\mathbf{W}$ is sparse. Pace and Barry proposed an SLD-based SAR model solution which uses sparse LU factorization using sparse data structures [30]. The number of operations of sparse LU factorization is O($2nb_ub_l$), where b_u and b_l correspond to the upper and lower bandwidths of the neighborhood matrix $\mathbf{W}$. Using sparse data structures drastically decreases the computation time and memory usage. However, even if sparse data structures are used, the computation of the computationally expensive log-det term of the log-likelihood function must be repeated in the parameter estimation process of the SAR model (Figure 4). As a result, ML-based exact SAR solutions in the literature exhibit high computational cost and thus are not scalable to large problem sizes.

In contrast, we limit the search space of the computationally expensive determinant computation of the log-likelihood function by finding an upper bound on the spatial autocorrelation parameter. First, we calculate the computationally efficient term (*SSE* term) of the log-likelihood function for finding an upper bound on the spatial autocorrelation parameter and then, we limit the

Input: $(\rho_{start}, tol, \mathbf{W})$

Output: $(\rho_{opt}, \beta_{opt}, \sigma^2_{opt})$

Step:

1. Step (i) {
2. ρ_{opt}=GSS($range = [0, 1]$,
3. $start = \{\rho_{start}\}$,
4. $f_{loglike} = \frac{-2}{n} \cdot \ln |\mathbf{I} - \rho\mathbf{W}| + SSE$)
5. Compute $(\beta_{opt}, \sigma^2_{opt})$ }
6. return $(\rho_{opt}, \beta_{opt}, \sigma^2_{opt})$

Fig. 4. The pseudocode of the EV-based and SLD-based SAR model solutions. The only difference of the related works is deciding whether to use sparse data structures or not. SLD-based solution uses sparse data structures and EV-based solution uses dense data structures. GSS stands for golden section search.

number of evaluations of the computationally expensive term (log-det term) of the log-likelihood function using this upper bound to find the optimum SAR model autocorrelation parameter. The proposed algorithm (NORTHSTAR) promises to reduce the computational cost and to scale to large problem sizes.

D. Contributions

Major contributions of this study include the following:

1) We developed a faster, scalable and **NO**vel p**R**ediction and estimation **T**ec**H**nique for the exact **S**pa**T**ial **A**uto**R**egression model solution (NORTHSTAR). In the first step of our approach, the SAR model parameters are estimated using the much less computationally complex sum-of-squared errors (*SSE*) term of the log-likelihood function. A second computationally more complex step is required only if the parameters obtained in the first step are not in the desired precision; in this case, the log-det term is embedded into the estimation process.
2) We analytically showed that the estimated SAR model parameter obtained after the first step can be used as an upper bound in the second step based on SAR Unimodularity Theorem 4.1 and Lemma 4.3, if the second step is necessary.

3) We experimentally showed that the proposed heuristic, NORTHSTAR, is computationally more efficient and scalable (in terms of memory usage and CPU time) than the previous work, i.e., the eigenvalue (EV) based and straight log-det (SLD) based approaches.

E. Outline of the Paper and Scope

The remainder of the paper is organized as follows: Section II presents the theory of the SAR model. The proposed approach, NORTHSTAR, for the SAR model solution is presented in section III. Section IV gives the analysis of the NORTHSTAR algorithm and section V presents experimental evaluations of the proposed algorithm. We conclude and summarize the paper with a discussion of future work in section VI.

This paper focuses on developing a new ML-based exact SAR model solution, NORTHSTAR.

II. Basic Concepts: Spatial Autoregression (SAR) Model

The SAR model [11], [2], also known in the literature as the spatial lag model [2] or mixed regressive model [31], is an extension of the linear regression model and is given in (2).

$$\mathbf{y} = \rho\mathbf{W}\mathbf{y} + \mathbf{x}\beta + \epsilon \tag{2}$$

In the equation, $\mathbf{y}$ is the n-by-1 vector of observations on the dependent variable, where n is the number of observation points; ρ is the spatial autoregression parameter; $\mathbf{W}$ is the n-by-n neighborhood matrix that accounts for the spatial relationships (dependencies) among the spatial data; $\mathbf{x}$ is the n-by-k matrix of observations on the explanatory variable, where k is the number of features; β is a k-by-1 vector of regression coefficients; and ϵ is an n-by-1 vector of unobservable error which is assumed to be generated from independent and identical standard normal distribution. Spatial autocorrelation term $\rho\mathbf{W}\mathbf{y}$ is added to the linear regression model in order to model the strength of the spatial dependencies among the elements of the dependent variable, $\mathbf{y}$. Data structures of the SAR equation can be seen in Figure 5. Construction of the neighborhood matrix $\mathbf{W}$ is discussed in Appendix I for regular and irregular grid spaces.

The solution procedure for the SAR equation is decided to be more complex than that for the linear regression equation because of the presence of the $\rho\mathbf{W}\mathbf{y}$ term on the right side of the equation. Also notice that the $\mathbf{W}$ matrix is quadratic in size relative to the size of the data

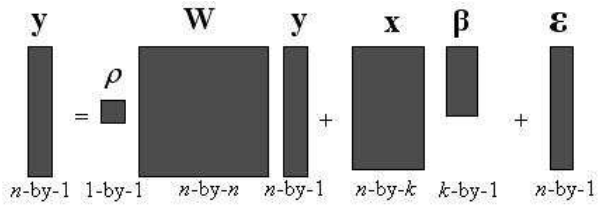


Fig. 5. Data structures of SAR Model

samples. Fortunately, very few entries of $\mathbf{W}$ are non-zero, and sparse matrix techniques are used, which exploit this fact, to speed up the solution process.

III. Proposed Approach

This section describes the new ML-based exact SAR model solution, NORTHSTAR, and then discusses the design decisions.

A. NORTHSTAR Algorithm

The NORTHSTAR algorithm aims to decrease the number of computations of the computationally expensive log-det term of the log-likelihood function which is given in (1) by finding an upper bound on the spatial autocorrelation parameter. In the first step of the algorithm, an upper bound on the spatial autocorrelation paremeter is estimated using a computationally more efficient *SSE* term of the log-likelihood function of the SAR model. In the second step, the computationally more expensive log-det term is embedded into the estimation process. The second step (of the NORTHSTAR algorithm) uses the upper bound on the spatial autocorrelation parameter, found in the first step, to narrow the search space and to decrease the number of determinant evaluations of a large matrix.

The pseudocode of the NORTHSTAR algorithm is given in Figure 6, where GSS stands for golden section search. Instead of the golden section search, which is not dependent to the derivative of the optimized function, a derivative-based search algorithm can be used for faster convergence to the optimal SAR parameter ρ, but it is necessary to compute the inverse of a large matrix $(\mathbf{I} - \rho\mathbf{W})$, which is as costly as the determinant computation of a large matrix.

Input: (ρ_{start},tol, $\mathbf{W}$)

Output: ($\rho_{opt}, \beta_{\mathbf{opt}}, \sigma^2_{opt}$)

Steps:

1. Step (i) {
2. $\rho_{init-est}$=GSS($range = [0, 1]$,
3. $start = \{\rho_{start}\}$,
4. $f_{loglike} =$ *SSE*) }
5. Step (ii) {
6. ρ_{opt}=GSS($range = [0, \rho_{init-est}]$,
7. $start = \{\rho_{init-est}\}$,
8. $f_{loglike} = \frac{-2}{n} \cdot \ln |\mathbf{I} - \rho\mathbf{W}| +$ *SSE*)
9. Compute $(\beta_{opt}, \sigma^2_{opt})$ }
10. return $(\rho_{opt}, \beta_{opt}, \sigma^2_{opt})$

Fig. 6. The NORTHSTAR algorithm.

B. Design Decisions

The design decisions for the NORTHSTAR algorithm consist of choosing, the range of the SAR autocorrelation parameter ρ and neighborhood structure, (i.e., sparse vs. dense neighborhood matrix).

1) The Range of SAR Autocorrelation Parameter ρ: The range of the ρ parameter affects the performances of the SAR algorithms since it determines the search space of the algorithm. Lemmas 3.1 and 3.2 helps in the optimization of the SAR model parameters by ensuring that the SAR parameter ρ will be between a -1 and 1 interval , thereby reducing the search space of the SAR parameter ρ.

Lemma 3.1: Neighborhood matrix $\mathbf{W}$ has real eigenvalues, regardless of the neighborhood topology, as long as the neighborhood relation is symmetric (i.e. if i is a neighbor of j, then j is a neighbor of i).

Proof: Let $\mathbf{A}$ be the adjacency matrix for the neighborhood graph for a domain: $a_{ij} = 1$ if and only if nodes i and j are neighbors (i.e. are correlated). All other off-diagonal entries and all the diagonal entries are all zero.

The matrix $\mathbf{W}$ is obtained by scaling the rows of $\mathbf{A}$ so that the entries in each row of $\mathbf{W}$ add up to one. That means that $\mathbf{W} = \mathbf{DA}$ for some diagonal scaling matrix $\mathbf{D}$. The matrix $\mathbf{D}$ is not only diagonal, but all the diagonal entries are strictly positive (because every node has at least one neighbor). Hence the square root of $\mathbf{D}$ is well-defined, and so is the inverse of $\mathbf{D}$. So it can be seen that $\mathbf{W}$ is diagonally similar to a symmetric matrix and hence must have real eigenvalues. Specifically, $\mathbf{D}^{-1/2}\mathbf{W}\mathbf{D}^{+1/2} = \mathbf{D}^{+1/2}\mathbf{A}\mathbf{D}^{+1/2}$ and it is symmetric and similar to $\mathbf{W}$. ■

Lemma 3.2: The eigenvalues of the row-stochastic (i.e., row-normalized, row-standardized or Markov) neighborhood matrix $\mathbf{W}$ are in the range $[-1, +1]$ (see also Theorem 5.3 in §2.5 on page 49 of [4], §5.13.3 in [27]).

Proof: All eigenvalues of a row-stochastic matrix is bounded by 1 in absolute value by Perron-Frobenius theorem (please see page 32 of [4] and page 120 of [10]). ■

It is also possible to put bounds on SAR autocorrelation parameter ρ. In this study, we used one of the terms of the log-likelihood function of the SAR model to define an upper bound on the SAR autocorrelation parameter ρ.

2) Neighborhood Structure: Neighborhood matrices are used to model the spatial dependencies of given objects. Matrices can be constructed on regular or irregular grid spaces (Appendix I). Although it is possible to use neighborhood structures on irregular grid space on the NORTHSTAR algorithm, in this study, we used a two-dimensional regular grid space with a four-neighborhood. We also compared the performances of the algorithm for different neighborhood structures.

The use of sparse or dense representation of the neighborhood matrices affects the execution time very much since the dense LU factorization (decomposition) which is used to find the determinant of a matrix requires n^3 operations while the sparse version needs only $2nb_u b_l$ operations [13], [16], where b_u and b_l correspond to the upper and lower bandwidths of the neighborhood matrix $\mathbf{W}$ respectively. In this study, we used sparse representation of neighborhood matrices.

IV. Analysis of the NORTHSTAR Algorithm

A. Is NORTHSTAR correct?

In this section, we show that the SAR log-likelihood function, given in (1), is unimodular by developing SAR Unimodularity Theorem 4.1. We also show that SAR autocorrelation parameter

ρ ($\rho_{init-est}$), minimizing the *SSE* term is an upper bound of SAR autocorrelation parameter ρ minimizing the log-likelihood function by developing Lemma 4.3.

Lemmas 3.1 and 3.2, described in the previous section, helps reduce the search space of the SAR parameter ρ into the interval of [-1,+1]. Based on Lemmas 3.1 and 3.2, we describe two Lemmas (Lemma 4.1 and Lemma 4.2). These lemmas define a function $f(x)$ which has a form similar to that of the exponential of SAR log-likelihood function and prove that there is only one zero of such a function in the interval of (-1,+1). Then, using Lemma 4.1 and Lemma 4.2, a SAR Unimodularity Theorem is developed which shows that the SAR log-likelihood function has at most one zero within the interval of (-1,+1).

Lemma 4.1: Let

$$f(x) = \frac{p(x)}{q(x)^{n/2}},$$

where $p(x)$ is a polynomial of degree n with all real distinct zeroes $r_1 < \ldots < r_n$ outside the open interval $(-1, 1)$, and $q(x)$ is a polynomial of degree 2 that is positive for all real x. Then $f'(x)$ has exactly one zero in each open interval (r_i, r_{i+1}), $i = 1, \ldots, n-1$. In particular, $f'(x)$ has at most one zero in the interval $(-1, 1)$, and hence $f(x)$ is unimodular in $(-1, 1)$.

Proof: By a straightforward computation, the derivative of f is

$$f'(x) = \frac{p'q - \frac{n}{2}pq'}{q^{n/2+1}} = \frac{N(x)}{D(x)}.$$

$N(x)$ is a polynomial of degree at most $n+1$. For polynomials, $p(x) = \alpha_n x^n + \alpha_{n-1}x^{n-1} + \cdots + \alpha_0$, and $q(x) = \beta_2 x^2 + \beta_1 + \beta_0$ the polynomial $N(x) = p'q - \frac{n}{2}pq' = \gamma_{n+1}x^{n+1} + \gamma_n x^n + \cdots + \gamma_0$, we have that the leading coefficient is $\gamma_{n+1} = n\alpha_n \cdot \beta_2 - \alpha_n \cdot \frac{n}{2}2\beta_2 = 0$. Hence $N(x)$ is actually a polynomial of degree at most n.

Now we localize the n zeroes of the polynomial $N(x)$. At each point r_i, $N(r_i) = p'(r_i)q(r_i) \neq 0$ has the same sign as $p'(r_i) \neq 0$, for $i = 1, \ldots, n$. Since the zeroes of $p(x)$ are distinct, the signs of $p'(r_i)$ alternate, hence so do the signs of $N(r_i)$. Thus, polynomial $N(x)$ must have at least one zero in each open interval (r_i, r_{i+1}), for $i = 1, \ldots, n-1$. Call these zeroes s_i, $i = 1, \ldots, n-1$. This accounts for $n-1$ zeroes, leaving only one left. Denote this one left over zero s_*.

Because all the γ's are real, s_* must be real. Within each interval (r_i, r_{i+1}), there must be an odd number of zeroes due to the sign changes, but we have only one zero left over, so the extra zero s_* must satisfy either $s_* < r_1$ or $s_* > r_n$.

Since no r_i is in the interval $(-1,1)$, $f'(x)$ can have at most one zero within $(-1,1)$. ■

Next, we extend Lemma 4.1 by taking the zero eigenvalues of the neighborhood matrix $\mathbf{W}$ into account, which gives us Lemma 4.2.

Lemma 4.2: Let

$$f(x) = \frac{p(x)}{q(x)^{n/2}},$$

where $p(x)$ is a polynomial of degree $d \leq n$ with all real zeroes $r_1 \leq \ldots \leq r_d$ outside the open interval $(-1,1)$, and $q(x)$ is a polynomial of degree 2 that is positive for all real x. Then either $f(x)$ is unimodular in $(-1,1)$ or there is a unimodular function arbitrarily close to f.

Proof: Let ϵ be chosen arbitrarily such that $0 < \epsilon < \frac{1}{2}$, and let ϵ_j, $j = 1, 2, \ldots$ be arbitrarily distinct but small numbers such that $|\epsilon_j| < \epsilon$ for all j and $\epsilon_j > 0$ for $j > d$. Write $p(x) = \alpha_d(x - r_1)\cdots(x - r_d)$. Define the n-th degree polynomial $\tilde{p}(x) = \alpha_d(x - r_1 + \epsilon_1)\cdots(x - r_d + \epsilon_d)(1 - \epsilon_{d+1}x)\cdots(1 - \epsilon_n x)$ to be a "slight" perturbation of $p(x)$. Define $\tilde{f}(x) = \frac{\tilde{p}(x)}{q(x)^{n/2}}$. The function $\tilde{f}(x)$ satisfies Lemma 4.1, so is unimodular in $(-1,1)$.

By construction, $|\tilde{p}(x) - p(x)| < \epsilon \cdot |e(x)|$ where $e(x)$ is some polynomial of degree n in x, independent of ϵ (we are using the fact that $\epsilon < \frac{1}{2}$). Hence in the interval $(-1,1)$

$$|\tilde{f}(x) - f(x)| < \epsilon \cdot \frac{\max_{x\in(-1,1)} |e(x)|}{\min_{x\in(-1,1)} q(x)^{n/2}}.$$

■

Due to the Lemma 4.2, function $f(x)$ cannot have two distinct maxima in the open interval $(-1,1)$, because one would not be able to find a unimodular function arbitrarily close to it, unless function $f(x)$ is constant.

Theorem 4.1: *SAR Unimodularity Theorem:* The log-likelihood function $\ell(\rho|\mathbf{y})$ as a function of ρ is unimodular for $\rho \in (-1,1)$.

Proof: As the exponential of log-likelihood function $\ell(\rho|\mathbf{y})$ has the same form of function $f(x)$ defined in Lemma 4.2, it directly follows that SAR likelihood function $\ell(\rho|\mathbf{y})$ is unimodular.

■

Since the log-likelihood function $\ell(\rho|\mathbf{y})$ is unimodular, the golden section search algorithm always finds the global minimum of the log-likelihood function. Thus, we have an optimal parameter estimation for the ML-based SAR model solutions. We plotted the SAR log-likelihood function $\ell(\rho|\mathbf{y})$ in order to see its extrema for a problem size of 2500 in Figure 7. As can be

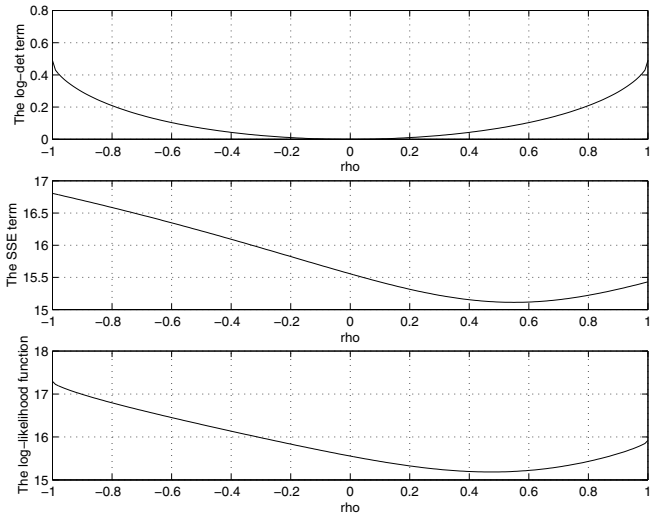


Fig. 7. The components of the log-likelihood function for a thematic class of satellite dataset with 2500 observation points and the log-likelihood function = log-det + *SSE*. "rho" stands for the SAR parameter ρ.

seen, the SAR log-likelihood function is unimodular and its components log-det and *SSE* terms are also unimodular.

Lemma 4.3: The initial estimate of the ρ parameter ($\rho_{init-est}$), calculated using the *SSE* term of the log-likelihood function $\ell(\rho|\mathbf{y})$, in the first step (step(i)) of NORTHSTAR is an upper limit on the location of autocorrelation parameter ρ optimizing the SAR log-likelihood function $\ell(\rho|\mathbf{y})$.

Proof: For most of the data mining problems of interest, spatial autoregression parameter ρ is in the interval [0,1], i.e. $0 < \rho < 1$.

Let us assume that functions *f1* and *f2* are unimodular, have minimas in the interval [-1,+1] and that the minima of *f1* is less than or equal to the minima of *f2* such that minima(*f1*) $\leq$ minima(*f2*). In that case, function *f1*+*f2* will also be unimodular and will have a minima between the minima of *f1* and *f2* such that minima(*f1*)$\leq$ minima(*f1*+*f2*) $\leq$ minima(*f2*).

Since the log-likelihood function $\ell(\rho|\mathbf{y})$ is unimodular and both the log-det and *SSE* terms are also unimodular, the minima of the log-likelihood function $\ell(\rho|\mathbf{y})$ is between the minimas of the log-det term and the *SSE* term. We need to prove that the minima of the log-det term is less than or equal to the minima of the *SSE* term and the minima of the *SSE* term is an upper bound for the log-likelihood function.

For a given $0 \leq \rho \leq 1$, and neighborhood matrix $\mathbf{W}$ with *K* symmetric pairs,

$$\text{log-det} = \ln |\mathbf{I} - \rho \mathbf{W}| = K \ln(1 - \rho^2) \tag{3}$$

To find the minima of the log-det term, we need to find the derivation of the log-det term and to set the derivation at 0.

$$\frac{d(\text{log-det})}{d(\rho)} = \frac{-2\rho K}{1 - \rho^2} \tag{4}$$

then $\rho = 0$, which is the minima of the log-det term.

In that case, the minima of the *SSE* term will be greater than or equal to the actual ρ value and 0 (which is the minima of the log-det term) and also it gives the upper bound on the ρ value, such that minima(*SSE*) $\geq$ minima($\ell(\rho|\mathbf{y})$) $\geq$ minima(log-det term).

Figure 7 shows the SAR autocorrelation parameter ρ values which are minimizing the log-det term, *SSE* term and log-likelihood function $\ell(\rho|\mathbf{y})$. It can be seen that both the log-det and *SSE* terms are unimodular in the open interval (-1,+1). Thus, the minima of the likelihood function $\ell(\rho|\mathbf{y})$ is between the minima of the log-det term and the minima of the *SSE* term. It can also be seen from Figure 7 that the value of the minima of the log-det term is less that the value of the minima of the *SSE* term. This observation shows that the minima of the *SSE* term is an upper bound of the minima of the sum of the *SSE* and the log-det term (The sum of both the terms is the log-likelihood function $\ell(\rho|\mathbf{y})$ as given in (1)). ■

B. How Computationally Expensive is the Proposed Algorithm-NORTHSTAR?

In this section, we show that the magnitude of the log-det term of the SAR log-likelihood function, given in (1), is very small with respect to the magnitude of the *SSE* term of the SAR log-likelihood function $\ell(\rho|\mathbf{y})$ by developing the Relative Magnitude Observation (Observation 4.1). The algorithms in the previous studies calculate the optimum SAR parameters using the log-det and *SSE* terms in the estimation procedure at the same time. It is observed that if the magnitude of the *SSE* term is bigger enough than the magnitude of the log-det term, it is possible that the effect of the log-det term in the calculations will be dominated by the magnitude of the *SSE* term, especially when the problem size is big. In such cases, the *SSE* term itself may be

enough to find optimal ρ parameter and there may be no need to include the log-det term in the SAR parameter estimation process. The NORTHSTAR algorithm is designed based on this observation and described in Lemma 4.4. In the first step of the NORTHSTAR, the *SSE* term is used to find the optimal ρ parameter; if the desired precision of the optimal ρ parameter is not enough, a second step is required. In the second step, the log-det term is included in the estimation procedure and optimal ρ parameter, found in the fist step, is used as an upper bound on optimum SAR autoroccelation parameter ρ.

Observation 4.1: *Relative Magnitude Observation:* The log-det term of the SAR log-likelihood function $\ell(\rho|\mathbf{y})$, given in (1), is very small in magnitude with respect to the magnitude of the *SSE* term of the log-likelihood function.

Proof: The magnitude (absolute value) of the log-det term of the log-likelihood function given in (5) is a function of the ρ parameter and $\rho < 1$.

$$\text{abs}(\text{log-det term}) = \text{ abs}\left(\frac{-2}{n}\ln|\mathbf{I}-\rho\mathbf{W}|\right) = \text{ abs}\left(\frac{-2}{n}\sum_{i=1}^{n}\ln(1-\rho\lambda_i)\right) \leq \text{abs}\left(-2\ln(1+\rho)\right) \tag{5}$$

It can be seen that the magnitude of the log-det term is determined by the value of the ρ value. In the extreme case, ρ can be maximum 1 and the value of equation given in (5) can be approximately 1.4.

In contrast, the magnitude (absolute value) of the *SSE* term of the log-likelihood function $\ell(\rho|\mathbf{y})$ is a function of problem size n, values of the eigenvalues and dependent vector $\mathbf{y}$ as can be seen in (6).

$$\begin{aligned}\text{abs}(\text{SSE term}) = \text{ abs}\left(\ln((\mathbf{I}-\rho\mathbf{W})\mathbf{y})^T(\mathbf{I}-\mathbf{M})^T\ (\mathbf{I}-\mathbf{M})\ ((\mathbf{I}-\rho\mathbf{W})\mathbf{y})\right) \\ = \text{abs}\left(\ln(\mathbf{y}^T\mathbf{A}\ \mathbf{y})\right)\ = n*\ \text{E}[\lambda_i(y_1^2+...+y_n^2)]\end{aligned} \tag{6}$$

where $\mathbf{M} = \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T$, $\mathbf{A} = ((\mathbf{I}-\rho\mathbf{W})^T(\mathbf{I}-\mathbf{M})^T(\mathbf{I}-\mathbf{M})(\mathbf{I}-\rho\mathbf{W})$. E[.] represents the expected value, which is the average of the all eigenvalues in (6).

It can be seen that the magnitude of the *SSE* term is much bigger than the magnitude of the log-det term (i.e. (*SSE* term) $\gg$ (log-det term)), especially when the problem size is big and the norm of the dependent vector $\mathbf{y}$ is big.

Details of the proof can be seen in Appendix V. ■

Table II shows the magnitudes of the log-det and *SSE* terms at the optimal ρ value where log-likelihood function $\ell(\rho|\mathbf{y})$ is minimum for different neighborhood structures.

TABLE II

THE MAGNITUDES OF THE LOG-DET AND *SSE* TERMS AT THE OPTIMAL ρ VALUE WHERE THE LOG-LIKELIHOOD FUNCTION IS MINIMUM

Problem Size(n)	Neighborhood	ρ_{opt}	abs(Log-likelihood)	abs(*SSE*)	abs(log-det)
2500	4-N	0.467	15.185	15.125	0.061
2500	8-N	0.430	15.267	15.238	0.028

Lemma 4.4: Let the θ ratio be the ratio of the magnitude of the log-det term and the *SSE* term.

$$\theta = \frac{abs(\max_{\rho}(\text{log-det term}))}{abs(\min_{|\rho|<1}(SSE \text{ term}))}$$

If the θ ratio is small enough, there is no need to include the log-det term in the estimation procedure of the NORTHSTAR algorithm. Recall that the θ ratio will be small when the problem size gets bigger and the norm of $\mathbf{y}$ vector is big.

Proof: The Relative Magnitude Observation (Observation 4.1) proves that the magnitude of the *SSE* term is much bigger than the magnitude of the log-det term. In the worst case, ρ will be close to 1 and the absolute value of the log-det term will be close to 1.4. In contrast, the absolute value of the *SSE* term will increase with the increasing problem size and increasing magnitude of the norm of vector $\mathbf{y}$. In this case, θ will be small enough and the effect of the log-det term will be small, or even negligible, on the log-likelihood function calculation. Because of this property of the θ ratio, it can be used as a stopping criteria of the NORTHSTAR algorithm and may eliminate the need for numerous computations of the determinant of a large matrix. ■

This leads to our NORTHSTAR heuristic, which can be defined as follows:

- (i) $\rho_{init-est}$ = value of approximate ρ (ignoring log-det term)
- (ii) ρ_{opt} = value of ρ approximated in the second step of NORTHSTAR In this step $\rho_{init-est}$ is used as an upper bound.

The cost of the NORTHSTAR algorithm is dominated by the sparse LU factorization operation which is used to calculate determinant of $(\mathbf{I}-\rho\mathbf{W})$. The cost of it will be $(j-m)(2nb_ub_l)+9n^2+$

$2j-3$, where m is the savings from the log-det computation when there is no stopping criteria. When Lemma 4.4 the applied to the algorithm, the cost of the algorithm will be dependent on the function of the θ value, such that $(j-f(\theta))(2nb_ub_l)+9n^2+2j-3$. It should be noted that the $f(\theta)$ value will be close to the m value for small problem sizes. In contrast, $f(\theta)$ will be close to j for big problem sizes because of the huge savings from the log-det computation (such that $f(\theta) \gg m$). The parameters b_u and b_l correspond to the upper bandwidth and lower bandwidth of the neighborhood matrix $\mathbf{W}$ respectively. The parameter $(j-m)$ is the number of log-det computations for the NORTHSTAR algorithm. Next, we compare the cost of NORTHSTAR with the related approaches.

C. How Does NORTHSTAR Cost Compare with Related Approaches?

This section presents the cost-modeling of the exact SAR model solutions. The total computational complexity (the operation counts) of our NORTHSTAR heuristic is listed in Figure 8 and it should be noted that $j \ll n$. The eigenvalue computation based SAR model solution cannot go beyond problem sizes of $10K$ due to memory constraints. The parameters b_u and b_l correspond to the upper and lower bandwidths of the neighborhood matrix $\mathbf{W}$ respectively. The first terms of the NORTHSTAR and SLD cost functions ($(j-f(\theta))(2nb_ub_l)$ and $j(2nb_ub_l)+9n^2+j$), respectively) are the costs of the sparse LU factorization which needs to be calculated for each ρ until it reaches its optimum value ρ_{opt}. The rest of the cost functions of the NORTHSTAR and SLD algorithms are the costs of sparse matrix-vector multiplication of the GSS algorithm. The first term of the cost function of the EV is dense LU factorization and the rest is the dense matrix-vector multiplication of the GSS algorithm. The EV-based approach is more expensive than the others because of the dense LU factorization (n^3 operation). The NORTHSTAR algorithm is more efficient than the SLD algorithm, since it decreases the number of computations of sparse LU factorization.$f(\theta)$ represents the savings from the log-det computation.

Thus, for large problem sizes, NORTHSTAR is much more computationally efficient than the SLD and EV-based approaches.

V. Experimental Evaluation

We compared the NORTHSTAR algorithm with the EV-based, and SLD-based solutions using real and synthetic datasets to estimate SAR model parameters. It should be noted that the

Problem Size	NORTHSTAR	EV-based Approach	SLD-based Approach
n	$(j - f(\theta))(2nb_u b_l) + 9n^2 + 2j - 3$	$\frac{2}{3}n^3 + 529n^2 + j$	$j(2nb_u b_l) + 9n^2 + j$

Fig. 8. The total computational complexity (the operation counts) of our NORTHSTAR heuristics with respect to the previous exact SAR model implementations. The variable $f(\theta)$ is the number of savings from log-det computations. "SLD" stands for the straight log-det approach and "EV" stands for the eigenvalue approach.

algorithms give the same SAR parameter estimates since all of them are the exact SAR model solution.

A. Experimental Design and System Setup

We conducted experiments to answer the following questions:

- What is the execution time and memory usage of the proposed algorithm?
- What is the effect of the value of the SAR parameter ρ on the NORTHSTAR algorithm? What is the behavior of the NORTHSTAR algorithm for varying ρ parameters? How does the precision of the predicted ρ parameter affect the savings from log-det?
- What is the behavior of the NORTHSTAR algorithm for different problem sizes?
- What is the behavior of the NORTHSTAR algorithm for different neighborhood structures?
- What is the effect of the θ ratio over log-det savings?

The control parameters for the experiments are summarized in Table III. Notable solutions for the SAR model have been implemented in Matlab [25]. The system setup of the experiments is shown in Figure 9.

B. Datasets

1) Real datasets: We used real datasets from ecology and satellite remote-sensing image data in order to evaluate the SAR model solutions.

The ecology data is used to predict bird nest locations, as explained in Section I.A.

The satellite remote-sensing data is used for thematic classification. The study site encompasses Carlton County, Minnesota, which is approximately 20 miles southwest of Duluth, Minnesota. The region is predominantly forested, composed mostly of upland hardwoods and low-land conifers. There is a scattering of agriculture throughout. The topography is relatively flat, with

TABLE III

THE EXPERIMENTAL DESIGN

Factor Name	**Parameter Domain**
Problem Size (n)	100, 400, 900, 1600, 2500, 10,000, 160,000 and 1,000,000 observation points
Neighborhood Structure	2-D with 4-neighbors, 8-neighbors, and 12-neighbors
Candidates	Eigenvalue Based Approach, Straight Log-det Based Approach, and NORTHSTAR
SAR Parameter ρ	[0,1)
Optimization	Non-derivative Based Optimization
Dataset	Real and Synthetic Datasets
Hardware	IBM Regatta and IBM Netfinity Linux Cluster

the exception of the eastern portion of the county containing the St. Louis River. Wetlands, both forested and non-forested, are common throughout the area. The largest city in the area is Cloquet, a town of about 10,000. For this study we used a spring Landsat 7 scene, taken May 31, 2000. This scene was clipped to the Carlton County boundaries, which resulted in an image of size 1343 lines by 2043 pixels and 6-bands. Out of this we took a subset image of 1200 by 1800 to eliminate boundary zero-valued pixels. This translates to a $\mathbf{W}$ matrix of size 2.1 million x 2.1 million (2.1M x 2.1M) points. The observed variable $\mathbf{x}$ is a matrix of size 2.1M by 6. We chose nine thematic classes for the classification.

In thematic classification, the goal is to categorize the pixels of satellite images into classes (e.g., water, urban, rural, forest,...) based upon the values of the "spectral signatures" recorded by receivers on board the satellite. The problem of thematic classification has deep spatial connections because in most instances, pixels, which are neighbors on the image, belong to the same class. Thus satellite images naturally exhibit high spatial autocorrelation if pixel sizes are smaller than the size of spatial features.

2) Synthetic dataset generation: Synthetic datasets were generated using standard normal distribution with unit standard deviation and zero mean for different problem sizes, such as n=100, 400, 900, 1600, 2500 and for different ρ parameters, such as ρ=0.1, 0.2, 0.3, ..., 0.9. Observation variable n-by-k $\mathbf{x}$ and unobservable error n-by-1 ϵ were generated using standard

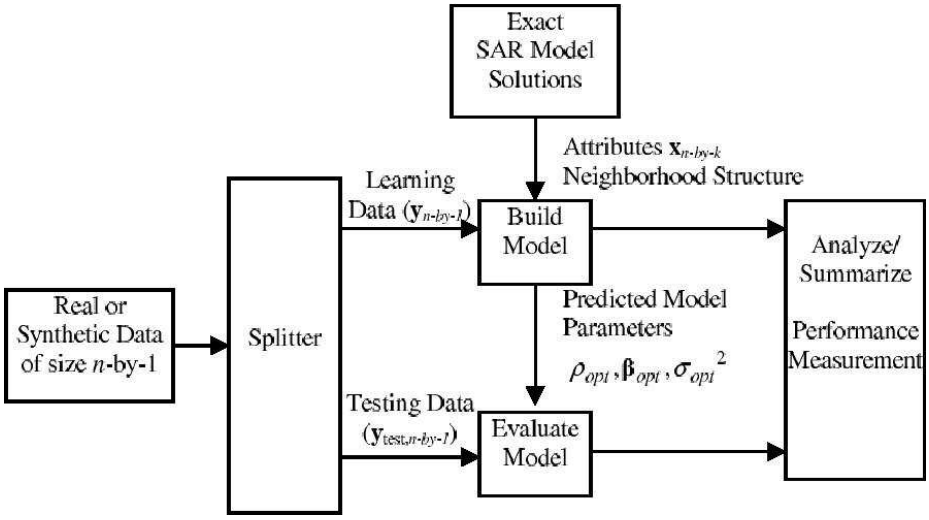


Fig. 9. The flow diagram for experiments

normal distribution for k=5 explanatory variables. Regression coefficients k-by-1 β vector were taken as one vector. Using these parameters, dependent variable $\mathbf{y}$ was generated for different problem sizes and ρ parameters.

C. Experimental Results

1) Scalability and memory usage of the algorithms: We compared the execution time and memory usage of the EV-based, SLD-based, and NORTHSTAR algorithms for different problem sizes using the real dataset.

Results showed that the NORTHSTAR algorithm is faster than EV-based and SLD-based approaches, especially when the problem size is increased, because of the log-det savings of the NORTHSTAR algorithm (Table IV).

The memory usage of the SLD-based and NORTHSTAR algorithms is very low due to the sparse representation of the neighborhood matrix $\mathbf{W}$ as a sparse matrix (Table IV). However, this is not possible for the EV-based approach since it has to use the dense representation of the matrix. Results showed that NORTHSTAR is the most scalable algorithm among the exact SAR model solutions, when execution time and memory usage are considered.

Since the execution time and memory usage of the EV-based approach is too high, we compared only the SLD-based and NORTHSTAR algorithms in the rest of the experiments.

TABLE IV

THE EXECUTION TIME AND MEMORY USAGE

Problem Size(n)	Execution Time		
	EV-based App.	SLD-based App.	NORTHSTAR
400x400 (160,000)	Intractable	32 minutes	24 minutes
1000x1000 (1,000,000)	Intractable	72 hours	45 hours
Problem Size(n)	Memory (MB)		
	Exact EV	Exact SLD	NORTHSTAR
50x50 (2,500)	50	1	1
100x100 (10,000)	2400	4.5	4.5
400x400 (160,000)	$\sim 6.14 * 10^5$	70	70
1000x1000 (1,000,000)	$\sim 8 * 10^6$	450	450

2) Effect of the SAR parameter ρ: We conducted experiments to characterize the behavior of the NORTHSTAR algorithm for varying desired precision and varying values of the ρ parameter.

Effect of the desired precision of the ρ parameter: We examined the effect of the desired precision of the ρ parameter for problem size 2500 of the SLD and NORTHSTAR algorithms. We used the real dataset (e.g., satellite remote-sensing dataset) and an optimum SAR autocorrelation parameter ρ of $4.7293 * 10^{-1}$ for this dataset. In the stage (ii) of the NORTHSTAR algorithm set the starting point for searching optimal ρ value as $\frac{(2*\rho_{init-cst})}{3}$. It is observed that when the precision decreased, the savings from the log-det computation of the NORTHSTAR increases (Figure 10).

Effect of value of ρ parameter: We conducted experiments to determine the behavior of the NORTHSTAR algorithm and compared the log-det savings of the SLD-based and NORTHSTAR algorithms for various ρ parameters. Synthetic datasets were produced using standard normal distribution for different ρ parameters for the fixed problem size 2500. For each ρ parameter, experiments were conducted 10 times and the number of log-det savings in Figure 11 shows the average of these 10 runs. Results showed that if the value of the spatial autocorrelation parameter ρ is low, NORTHSTAR algorithm outperforms SLD-based approach and if it is high, there may be no significant savings from the log-det computation (Figure 11). It is a fact that when the ρ parameter is close to 1 (which is the theoretical upper bound), the upper bound on

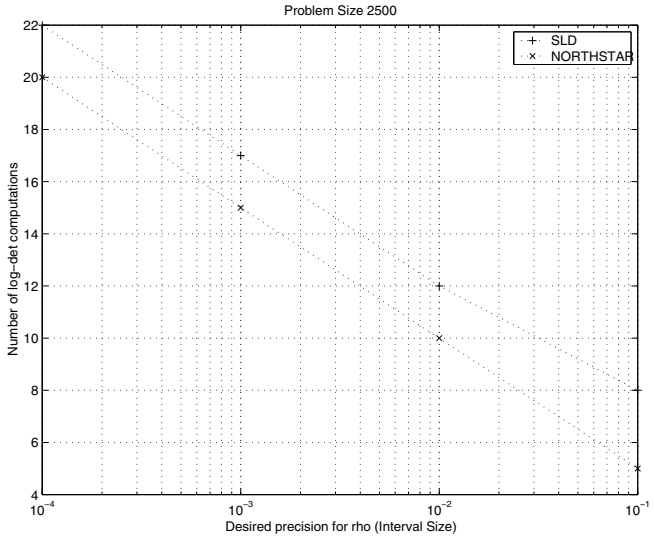


Fig. 10. The savings from log-det computation. "rho" stands for the SAR parameter ρ

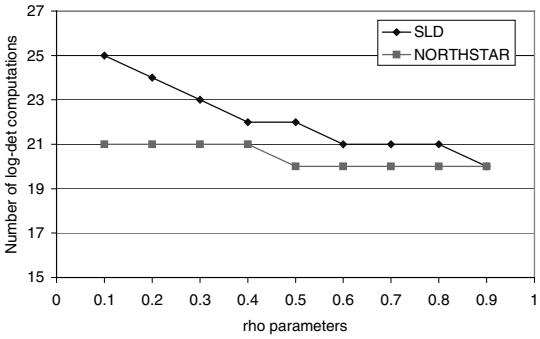


Fig. 11. Effect of value of ρ parameter

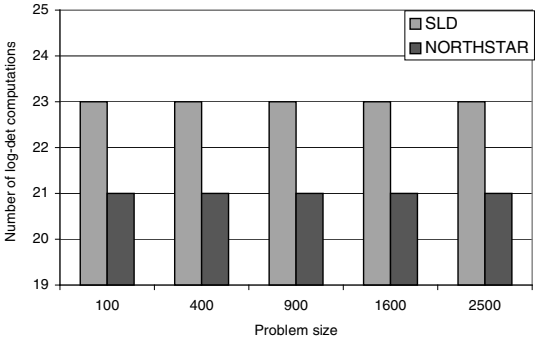


Fig. 12. Effect of problem size

the SAR autocorrelation parameter found in the first step of the NORTHSTAR will be close to 1 (Figure 11) and will have no significant effect to limit the search space of the second step of the NORTHSTAR algorithm. In such a case, NORTHSTAR will behave like the SLD-based approach. For SLD-based approach, the bound of the ρ parameter is [0,1]. For the NORTHSTAR algorithm the bound of the ρ parameter in stage(i) is [0,1] and in stage(ii) [0,ρ_{init}] (ρ_{init} is the result of the stage(i)).

3) Effect of problem size: : We conducted experiments to see the behavior of the NORTHSTAR algorithm for different problem sizes and compared it with SLD-based approach algorithm. In the experiments, synthetic dataset are used for problem sizes 100, 400, 800, 1600, and

			N2			
		NW	N	NE		
	W2	W		E	E2	
		SW	S	SE		
			N2			

Fig. 13. 2-D regular grid space and neighborhoods of the center cell

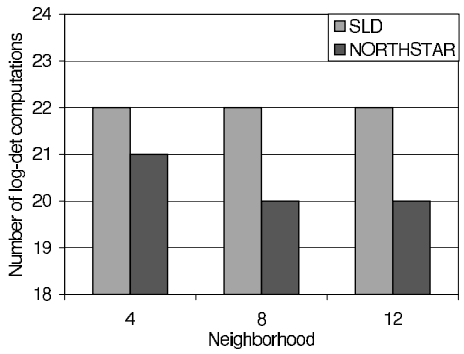


Fig. 14. Effect of different neighborhood structures

2500. Datasets are produced using standard normal distribution for a fixed SAR autocorrelation parameter ρ such that 0.3. Experiments showed that NORTHSTAR outperforms SLD-based approach and the number of log-det computation is constant with the increasing problem size for SLD-based approach and NORTHSTAR (Figure 12). Since each log-det computation will be costly, constant saving will be valuable, especially, with the increasing problem size.

4) Effect of neighborhood structure: : We conducted experiments using different neighborhood structures to determine behavior of NORTHSTAR algorithm. Real dataset is used for problem size 2500. 4-neighbors, 8-neighbors, and 12-neighbors structures are used for 2-D regular grid space (Figure 13). For the 2-D regular grid space, the 4-neighbors of a cell (center cell in Figure 13) can be defined as the cells which are found on four direction of it, such as the North (N), South (S), East (E), and West (W). For the 2-D regular grid space, the 8-neighbors of a cell (center cell in Figure 13) can be defined as the cell which are found on eight directions of it, such as N, NW, W, SW, S, SE, E, and NE. For the 2-D regular grid space, the 12-neighbors include second North (N2), second West (W2), second (E2), and second South (S2) cells in addition to the 8-neighbors cells.

Experiments showed that when the neighborhood structure increased number of log-det computation of SLD does not change, although number of the log-det computation of the NORTHSTAR algorithm decreases.

5) Effect of θ Ratio: Experiments showed that if the magnitude of the log-det term is less than or equal to $\frac{1}{60}$ of the magnitude of the *SSE* term computed in step(i) of NORTHSTAR, then

the ρ value that our NORTHSTAR heuristic finds in its step (i) is within the ± 0.1 range of the optimal ρ value. In other words, if the desired precision is ± 0.1, one would not need to run the second step of NORTHSTAR, where we compute the computationally expensive log-det term.

VI. Conclusions and Future Work

Linear regression is one of the best-known classical data mining techniques. However, it makes the assumption of independent identical distribution (*i.i.d.*) in learning data samples, which does not work well for geo-spatial data, which is often characterized by spatial autocorrelation. In the SAR model, spatial dependencies within data are taken care of by the autocorrelation term, and the linear regression model thus becomes a spatial autoregression model. Incorporating the autocorrelation term enables better prediction accuracy. However, computational complexity increases due to the need for computing the determinant of a large matrix $(\mathbf{I} - \rho\mathbf{W})$.

The related work computes determinant term of a large matrix of SAR model solution repeatedly to determine optimum values of SAR parameters, namely, autocorrelation parameter and weights for explanatory variables.

Since the determinant computation of a large matrix is computationally expensive, we developed a faster, scalable and novel prediction and estimation technique for an exact SAR model solution (NORTHSTAR) which is based on sparse LU decomposition and aims to recude the number of determinant computation of a very large matrix . This yields to reduce computation cost of the SAR parameter estimation process. The key idea is narrow the search interval by a cheap computation yielding an upper bound on the spatial autocorrelation parameter (Lemma 4.3). In the paper we proved that both terms of the ML-based SAR model solution (determinant term and *SSE* term) are unimodular and contains SAR autocorrelation term (Theorem 4.1). In addition, the location of autocorrelation parameter minimizing the *SSE* term of ML function is an upper bound on the location of autocorrelation parameter optimizing the likelihood function (Lemma 4.3). This upper bound allows us to narrow the search interval and reduce the number of iterations and number of determinant evaluations of an iterative search to estimate the spatial autocorrelation parameter of SAR model.

Analytical and Experimental evaluations showed that our approach is computationally more efficient than related work. We analyzed that the evaluation of the *SSE* term of ML function gives an upper bound on autocorrelation parameter of the likelihood function by using SAR

Unimodularity Theorem 4.1 and Lemma 4.3. Experimental results show that the computational cost of proposed approach is usually smaller than the of related approaches. The experiments show that proposed approach is computationally more efficient than related approaches in terms of execution time and memory usage. The experiments show that when the value of autocorrelation parameter gets smaller, the advantage of the proposed approach increases. It is also observed that determinant computation saving increases for the bigger neighborhood structures.

We have two main items for future work:

1) Regarding bounds on the parameter ρ, we will investigate ways to eliminate computing some of the eigenvalues to reach very high problem sizes with the eigenvalue approach.
2) We plan to reduce the bandwidth of the $\mathbf{W}$ matrix for big neighborhood structures. This will help us obtain a more efficient solution for all eigenvalues using sparse matrix algebra.

ACKNOWLEDGMENT

The authors would like to thank the members of the Spatial Database Group, ARCTiC Labs Group, Professor Vladimir Cherkassky, Mingyu Guo, Dr. Birali Runesha and Dr. Shuxia Zhang at Minnesota Supercomputing Institute for valuable discussions. The authors thank Kim Koffolt for helping improve the readability of this paper.

REFERENCES

[1] P. Albert and L. McShane. A generalized Estimating Equations Approach for Spatially Correlated Binary Data: Applications to the Analysis of Neuroimaging Data. *Biometrics (Publisher: Washington, Biometric Society, etc.)*, 51:627–638, 1995.

[2] L. Anselin. *Spatial Econometrics: Methods and Models*. Kluwer Academic Publishers, Dorddrecht, 1988.

[3] F. Bavaud. Models for spatial weights: A systematic look. *Geographical Analysis*, 30:153–171, 1998.

[4] A. Berman and R. Plemmons. *Nonnegative Matrices in the Mathematical Sciences*. Computer Science and Applied Mathematics, 1979.

[5] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society, B*, 36:192–225, 1974.

[6] J. Besag. Statistical analysis of nonlattice data. *The Statistician*, 24:179–195, 1975.

[7] E. Borghers and P. Wessa. Scientific resources. *http://www.xycoon.com/matrix-algebra.htm*.

[8] S. Boyd. Lecture notes for ee263: Introduction to linear dynamical systems- lecture 13: Symmetric matrices, quadratic forms,matrix norm and svd. *Stanford University*, 2004.

[9] S. Chawla, S. Shekhar, W. Wu, and U. Ozesmi. Modeling spatial dependencies for mining geospatial data. *1st SIAM International Conference on Data Mining*, 2001.

[10] D. Cox and H. Miller. *The Theory of stochastic processes*. Methuen, London, 1965.

[11] N. Cressie. *Statistics for Spatial Data*. Wiley, New York, 1993.

[12] R. Davidson and J. MacKinnon. *Estimation and Inference in Econometrics*. Oxford University Press, New York, 1993.

[13] J. W. Demmel. *Applied Numerical Linear Algebra*. SIAM, 1997.

[14] J. Egan. *Signal Detection Theory and ROC Analysis*. Academic Press, New York, 1975.

[15] J. Freund and R. Walpole. *Mathematical Statistics*. Prentice Hall, 1980.

[16] G. Golub and C. V. Loan. *Matrix Computations*. Johns Hopkins University Press, 1996.

[17] D. Griffith. *Advanced Spatial Statistics*. Kluwer Academic Publishers, 1998.

[18] F. Hayashi. *Econometrics*. Princeton University Press, 2000.

[19] M. Hohn and L. G. A.E. Liebhold. A Geostatistical model for Forecasting the Spatial Dynamics of Defoliation caused by the Gypsy Moth, Lymantria dispar (Lepidoptera:Lymantriidae). *Environmental Entomology (Publisher: Entomological Society of America)*, 22:1066–1075, 1993.

[20] R. Horn and C. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[21] Isaaks, Edward, and M. Srivastava. *Applied Geostatistics*. Oxford University Press, Oxford, 1989.

[22] B. Kazar, S. Shekhar, D. Lilja, and D. Boley. A parallel formulation of the spatial auto-regression model for mining large geo-spatial datasets. *SIAM International Conf. on Data Mining Workshop on High Performance and Distributed Mining (HPDM2004)*, April 2004.

[23] B. Klinkenberg. Geography 471: Applied gis: Using your knowledge. *http://www.geog.ubc.ca/courses/geog471/notes/*.

[24] P. Krugman. *Development, geography, and economic theory*. MIT Press, Cambridge, MA, 1995.

[25] J. LeSage. Econometrics toolbox for matlab. *http://www.spatial-econometrics.com/*.

[26] B. Li. Implementing spatial statistics on parallel computers. *Practical Handbook of Spatial Statistics, CRC Press*, pages 107–148, 1996.

[27] M. Marcus and H. Minc. *A Survey of Matrix Theory and Matrix Inequalities*. Dover, New York, 1992.

[28] S. Ozesmi and U. Ozesmi. An Artificial neural network approach to spatial habitat modeling with interspecific interaction. *Ecological Modelling (Publisher: Elsevier Science B. V.)*, (116):15–31, 1999.

[29] U. Ozesmi and W. Mitsch. A spatial habitat model for the Marsh-breeding red-winged black-bird(agelaius phoeniceus l.) In coastal lake Erie wetlands. *Ecological Modelling (Publisher: Elsevier Science B. V.)*, (101):139–152, 1997.

[30] R. Pace and R. Barry. Quick computation of spatial auto-regressive estimators. *Geo-graphical Analysis*, 29:232–246, 1997.

[31] R. Pace and R. Barry. Simulating mixed regressive spatially autoregressive estimators. *Computational Statistics*, 13:397–418, 1998.

[32] R. Pace and J. LeSage. Closed-form maximum likelihood estimates for spatial problems (mess). *http://www.spatial-statistics.com*, 2000.

[33] R. Pace and J. LeSage. Semiparametric maximum likelihood estimates of spatial dependence. *Geographical Analysis*, 34(1):76–90, 2002.

[34] R. Pace and J. LeSage. Simple bounds for difficult spatial likelihood problems. *http://www.spatial-statistics.com*, 2003.

[35] R. Pace and J. LeSage. Spatial auto-regressive local estimation (sale). *Spatial Statistics and Spatial Econometrics, ed. by Art Getis*, 2003.

[36] R. Pace and J. LeSage. Chebyshev approximation of log-determinant of spatial weight matrices. *Computational Statistics and Data Analysis*, 2004.

[37] R. Pace and J. LeSage. Closed-form maximum likelihood estimates of spatial auto-regressive models: the double bounded likelihood estimator (dble). *Geographical Analysis*, Forthcoming.

[38] R.J.Haining. *Spatial Data Analysis in the Social and Environmental Sciences*. Cambridge University Press, Cambridge, U.K., 1989.

[39] S. Shekhar and S. Chawla. *Spatial Databases: A Tour*. Prentice Hall, 2003.

[40] S. Shekhar, P. Schrater, R. Raju, and W. Wu. Spatial contextual classification and prediction models for mining geospatial data. *IEEE Transactions on Multimedia*, 4(2):174–188, 2002.

[41] S. Shekhar, T. Yang, and P. Hancock. An Intelligent Vehicle Highway Information Management System. *Intl Jr. on Microcomputers in Civil Engineering (Publisher:Blackwell Publishers*, 8, 1993.

[42] R. van der Kruk. A general spatial arma model: Theory and application. *ERSA (Europian Regional Science Association) Conference*, pages 110–131, 2002.

[43] Y. Yasui and S. Lele. A Regression Method for Spatial Disease Rates: An Estimating Function Approach. *Journal of the American Statistical Association*, 94:21–32, 1997.

APPENDIX I
CONSTRUCTING NEIGHBORHOOD MATRIX

A. Constructing Neighborhood Matrix ($\mathbf{W}$) on Regular (Uniform) Grid Space.

Several previous studies have shown that modeling of spatial dependency during the prediction and classification process improves overall prediction and classification accuracy. Spatial dependency can be defined by the relationship between spatially adjacent pixels in a small neighborhood. The spatial relationship among locations in a spatial framework is often modeled via a neighborhood (contiguity) matrix. A simple neighborhood matrix may represent the neighborhood relationship defined using adjacency, Euclidean distance, etc. Example definitions of neighborhood using adjacency include 2-neighborhood on 1-dimensional grid space and 4-neighborhood, 8-neighborhood, 12-neighborhood and so on neighborhood on 2-dimensional grid space. This structure is also known as regular square tessellation 1-dimensional and 2-dimensional planar surface partitioning [17]. One can use Moran's I index in order to see whether there is significant spatial dependency in the given dataset (attributes). Appendix II summarizes Moran's I index computation.

The rows (neighboring values) of neighborhood matrix $\mathbf{W}$ sum to 1, which means that $\mathbf{W}$ is row-standardized i.e., row-normalized or row-stochastic. A non-zero entry in the j^{th} column of the i^{th} row indicates that the j^{th} observation will be used to adjust the prediction of the i^{th} row where i is not equal to j.

To form the row-normalized neighborhood matrix $\mathbf{W}$, first a non-normalized neighborhood matrix $\mathbf{C}$ formed by putting putting "1"s for neighborhoods of $(i,j)^{th}$ pixel of the spatial framework and by putting zeros for the rest of the entries. Then, non-zero row elements of $\mathbf{C}$ matrix are divided by each row sum of it. The algebraic equivalent of this definition can be formulized as $\mathbf{W} = \mathbf{D}^{-1}\mathbf{C}$ where $\mathbf{D}$ is a diagonal matrix whose diagonal elements contains row sums of matrix $\mathbf{C}$ and the rest of the elements of the $\mathbf{D}$ matrix is zero, that is $d_{ii} = \sum_{i=1}^{n} c_{ij}$ and $d_{ij} = 0$. In other words, $\mathbf{W}$ matrix is formed by dividing non-zero elements of $\mathbf{C}$ by corresponding diagonal element of $\mathbf{D}$.

Next, illustration of the neighborhood matrix formation on 4-by-4 regular grid space using 4-neighborhood spatial relationships is discussed.

1) Illustration of the Neighborhood Matrix Formation on a 4-by-4 Regular Grid Space: Spatial dependency can be defined by the relationships among spatially adjacent pixels in a small neighborhood within a spatial framework that is a regular grid space. Given a gridded spatial framework, the 4-neighborhood assumes that a pair of locations influence each other if they share an edge.

For the 4-neighborhood case, the neighbors of the $(i,j)^{th}$ pixel of the regular grid are the pixels which are found NORTH, SOUTH, EAST, and WEST side of it as shown in Figure 15.

$$neighbors(i,j) = \begin{cases} (i-1,j) & 2 \geq i \geq \phi,\ 1 \geq j \geq q \text{ NORTH} \\ (i,j+1) & 1 \geq i \geq \phi,\ 1 \geq j \geq q-1 \text{ EAST} \\ (i+1,j) & 2 \geq i \geq \phi-1,\ 1 \geq j \geq q \text{ SOUTH} \\ (i,j-1) & 1 \geq i \geq \phi,\ 2 \geq j \geq q \text{ WEST} \end{cases}$$

Fig. 15. The four neighbors of the $(i,j)^{th}$ pixel on the regular grid.

Using this 4-neighborhood definition the non-row-normalized spatial neighborhood matrix $\mathbf{C}$ of the 4-by-4 spatial framework given in Figure 16 can be formed as shown in Figure 17(a). For example, the neighbors of pixel 6 of the spatial framework is represented in the 6th row of the non-row-normalized neighborhood matrix $\mathbf{C}$ (Figure 17(a)) and the neighbors of the other pixels are represented in that fashion. The 2nd, 5th, 7th, and 10th columns of the 6th row contains value "1" since the neighbors of the pixel 6 are pixels 2 (NORTH), 5 (EAST), 7 (WEST), and 10 (SOUTH) in the spatial framework.

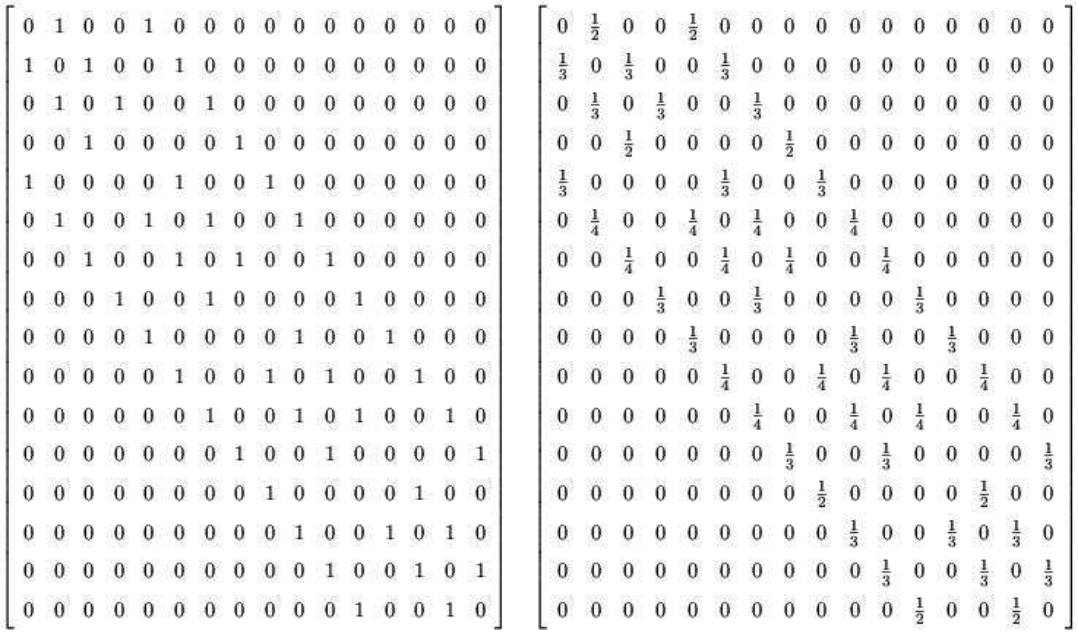
The row-normalized neighborhood matrix $\mathbf{W}$ (Figure 17(b)) is formed by dividing neighboring values by the row sums of the $\mathbf{C}$ matrix. For example, the 6th row of the $\mathbf{C}$ is divided by 4 which is row sum of it.

B. Constructing the Neighborhood Matrix $\mathbf{W}$ *on Irregular Grid Space*

Spatial statistics requires some means of specifying the spatial dependence among observations [17]. The neighborhood matrix i.e., $\mathbf{W}$, spatial weight matrix fulfills this role for lattice models [5], [6] and can be formed on both regular and irregular grid. This section shows a way to form the neighborhood matrix on the irregular grid space which is based on Delaunay triangulation

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

Fig. 16. The spatial framework which is 4-by-4 where rows may or may not be equal to columns.



(a) (b)

Fig. 17. (a) The 4×4-by-4×4 non-row-normalized neighborhood matrix $\mathbf{C}$ with 4 nearest neighbors. (b) The row-normalized version i.e. $\mathbf{W}$ which is also 4×4-by-4×4. The product 4×4 is equal to 16, the problem size n.

algorithm [34], [35]. [36] describes another method of forming the neighborhood matrix on the irregular grid which is based on nearest neighbors. One specification of the spatial weight matrix begins by forming the binary adjacency matrix $\mathbf{N}$ where $N_{ij} = 1$ when observation j is a neighbor to observation i ($i \neq j$). The neighborhood can be defined using computationally very expensive Delaunay triangulation algorithm [25]. These elements may be further weighted to give closer neighbors higher weights and incorporate whatever spatial information the user desires. By itself, $\mathbf{N}$ is usually asymmetric. To insure symmetry, we can rely on the transformation $\mathbf{C} = \left(\mathbf{N} + \mathbf{N}^T\right)/2$. The rest of forming neighborhood matrix on irregular grid follows the same

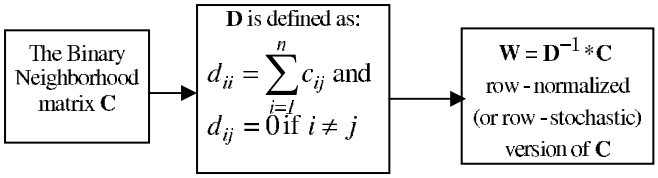


Fig. 18. Summary of formation of $\mathbf{W}$ matrix.

procedure discussed in the preceding section. Users often re-weight the adjacency matrix to create a row-normalized i.e., row-stochastic matrix or a matrix similar to a row-stochastic matrix. This can be accomplished in the following way: Let $\mathbf{D}$ represent a diagonal matrix whose i^{th} diagonal entry is the row-sum of the i^{th} row of matrix $\mathbf{C}$. The matrix $\mathbf{W} = \mathbf{D}^{-1/2}\mathbf{D}^{-1/2}\mathbf{C} = \mathbf{D}^{-1}\mathbf{C}$ is row-stochastic where $\mathbf{D}^{-1/2}$ is a diagonal such that its i^{th} entry is the inverse of the square root of the i^{th} row of matrix $\mathbf{C}$. Note that the eigenvalues of the matrix $\mathbf{W}$ do not exceed 1 in absolute value, and the maximum eigenvalue equals 1 via the properties of row-stochastic matrices (see Lemmas 3.2 and 3.1 and Figure 18 in this study, §5.13.3 in [27]).

From a statistical perspective, one can view $\mathbf{W}$ as a spatial averaging operator. Given the vector $\mathbf{y}$, the row-stochastic normalization i.e., $\mathbf{W}\mathbf{y}$ results in a form of local average or smoothing of $\mathbf{y}$. In this context, one can view elements in the rows of $\mathbf{W}$ as the coefficients of a linear filter. (See [3], [34], [35], [36] for more information on spatial weight matrices.)

APPENDIX II

MORAN'S I INDEX: QUANTIFYING THE AUTOCORRELATION IN DATASETS

Spatial autocorrelation analysis tests whether the observed value of a variable at one locality is independent of the values of the variable at neighboring localities. If a dependence exists, the variable is said to exhibit spatial autocorrelation. Spatial autocorrelation measures the level of interdependence between the variables, and the nature and strength of that interdependence. It may be classified as either positive or negative. In a positive case all similar values appear together, while a negative spatial autocorrelation has dissimilar values appearing in close association [23].

$$I = \frac{n}{S_0} \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} (w_{ij}(x_i - \bar{x})(x_j - \bar{x}))}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \tag{7}$$

The term S_0 is equal to $\sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}$ and n is the problem size.

APPENDIX III
SIMPLE OVERVIEW OF LOG-LIKELIHOOD THEORY

We simply define the likelihood function in this section. Let $\mathbf{Y} = (Y_1, Y_2, ..., Y_n)$ be a random vector. and define a statistical model $\{f_{\mathbf{Y}}(\mathbf{y}|\theta) : \theta \in \Theta\}$ which is parameterized by $\theta = (\theta_1, ..., \theta_n)$, the parameter vector in the parameter space Θ. The likelihood function is the mapping defined as $L : \Theta \to [0, 1] \subset \mathbb{R}$ given in (8).

$$L(\theta|\mathbf{y}) = f_{\mathbf{Y}}(\mathbf{y}|\theta) \tag{8}$$

In other words, the likelihood function is functionally the same in form as a probability density function (pdf). However, the emphasis is changed from $\mathbf{y}$ to θ. The pdf is a function of the $\mathbf{y}$'s while holding the parameters θ's constant, L is a function of the parameters θ's, while holding the $\mathbf{y}$'s constant. We can abbreviate $L(\theta|\mathbf{y})$ to $L(\theta)$. The parameter vector $\hat{\theta}$ such that $L(\hat{\theta}) \geq L(\theta)$ for all $\theta \in \Theta$ is called maximum likelihood estimate, or MLE, of θ. Many of the density functions are exponential in nature, therefore it is easier to compute the MLE of a likelihood function L by finding the maximum of the natural log of L, known as the log-likelihood function defined in in (9) due to the monotonicity of the log function. Finding maximum of a function is carried by taking the first derivative of that function and finding the values of parameters which equate the derivative to zero.

$$\ell(\theta|\mathbf{y}) = \ln(L(\theta|\mathbf{y})) \tag{9}$$

APPENDIX IV
BASIC LINEAR ALGEBRA FACTS

This appendix section presents the basic linear algebra equalities [7] used in our proofs.

- A real n-by-n matrix $\mathbf{A}$ is called a Markov matrix, or row-stochastic matrix if:
  1) $a_{ij} \geq 0$ for $1 \leq i, j \leq n$;
  2) $\sum_{j=1}^{n} a_{ij} = 1$ for $1 \leq i \leq n$.
- If $\mathbf{A}$ is a Markov matrix, then $\mathbf{A}\mathbf{J}_n = \mathbf{J}_n$ where $\mathbf{J}_n = [1, ..., 1]^T$. So, 1 is always an eigenvalue of a Markov matrix.
- If $\mathbf{A}$ and $\mathbf{B}$ are Markov matrices, then $\mathbf{AB}$ is also a Markov matrix.
- $tr(\mathbf{A}) = \sum_{i=1}^{n} a_{ii}$
- $\sum_{i=1}^{n} \lambda_i = tr(\mathbf{A})$ and $\prod_{i=1}^{n} \lambda_i = |\mathbf{A}|$
- If:
  1) $\mathbf{x}^T\mathbf{A}\mathbf{x} > 0$ then $\forall \mathbf{x} \neq 0$ the matrix $\mathbf{A}$ is called positive definite.
  2) $\mathbf{x}^T\mathbf{A}\mathbf{x} \geq 0$ then $\forall \mathbf{x}$ the matrix $\mathbf{A}$ is called positive semi-definite matrix.
- All positive definite matrices are non-singular.
- Eigenvectors of positive semi-definite matrices are non-negative.
- If $\mathbf{A}^2 = \mathbf{A}$, then matrix $\mathbf{A}$ is idempotent.
- All idempotent matrices are positive semi-definite with non-negative diagonal elements since $\mathbf{A}^T\mathbf{A} = \mathbf{A}\mathbf{A}^T = \mathbf{A} = \mathbf{A}^2$. Then, $\mathbf{x}^T\mathbf{A}\mathbf{x} = (\mathbf{A}\mathbf{x})^T\mathbf{A}\mathbf{x}$ which is just a sum of squares of the elements of $\mathbf{A}\mathbf{x}$.
- If the square of an idempotent matrix $\mathbf{A}$ is non-singular, then that matrix is the identity matrix. Because: $\mathbf{A}^2 = \mathbf{A}$ then $\mathbf{A}^{-1}\mathbf{A}^2 = \mathbf{A}^{-1}\mathbf{A}$ which means $\mathbf{A} = \mathbf{I}$.
- Eigenvalue of an idempotent matrix is either 0 or 1.
- If $\mathbf{A}$ is positive definite (or positive semi-definite) matrix and $\mathbf{B}$ is non-singular matrix i.e., its determinant is not zero then $\mathbf{B}^T\mathbf{A}\mathbf{B}$ is also positive definite (or positive semi-definite) matrix [7].

APPENDIX V
PROOF OF OBSERVATION 4.1

Proof: The following are the facts used in this proof:

1) Eigenvalues of a Markov matrix are in the range $[-1..1]$.
2) Maximum eigenvalue of a Markov matrix is 1.
3) $\text{abs}(\ln(1+xy)) < \text{abs}(\ln(1+x))$, if x and y are between 0 and 1.
4) $\text{abs}(\ln(1+\epsilon)) > \text{abs}(\ln(1-\epsilon))$ for small number $0 < \epsilon < 1$
5) $0 < \rho < 1$ for almost all spatial data mining problems of interest.

Now let us use these facts to rewrite $\frac{-2}{n}\sum_{i=1}^{n}\ln(1-\rho\lambda_i)$. Let's assume that we have a sequence of sorted n eigenvalues in the descending order and have l positive eigenvalues and $n-l-1$ negative eigenvalues. For instance, for 4-neighborhood structure l is $n/2$ and for 8-neighborhood case l is less than $n/2$.

$$\begin{aligned}
\text{abs}\left(\frac{-2}{n}\sum_{i=1}^{n}\ln(1-\rho\lambda_i)\right) &= \text{abs}\left(\frac{-2}{n}\left(\sum_{i=1}^{l}\ln(1-\rho\ \text{abs}(\lambda_i)) + \sum_{i=l+1}^{n}\ln(1+\rho\ \text{abs}(\lambda_i))\right)\right) \\
&\leq \text{abs}\left(\frac{-2}{n}\sum_{i=1}^{n}\ln(1+\rho\ \text{abs}(\lambda_i))\right) \text{ due to 4} \\
&\leq \text{abs}\left(\frac{-2}{n}\sum_{i=1}^{n}\ln(1+\rho)\right) \text{ due to 3} \\
&\leq \text{abs}\left(\frac{-2n}{n}\ln(1+\rho)\right) \\
&\leq \text{abs}\left(-2\ln(1+\rho)\right) \text{ due to 5} \qquad (10)
\end{aligned}$$

The quantity given in (10) is small as long as $\rho < 1$. Now, let's work on the *SSE* term and show that (*SSE*) $\gg$ (log-det).

We will review eigenvectors of symmetric matrices [8] since our matrix $(\mathbf{I}-\rho\mathbf{W})^T(\mathbf{I}-\mathbf{M})(\mathbf{I}-\rho\mathbf{W})$ is a symmetric matrix which we represent by matrix $\mathbf{A}$.

Suppose $\mathbf{A} \in \mathbf{R}^{nxn}$ is symmetric, i.e., $\mathbf{A} = \mathbf{A}^T$. It is the **fact** that the eigenvalues of the matrix $\mathbf{A}$ are real. To see this, suppose $\mathbf{Av} = \lambda\mathbf{v}$, where $\mathbf{v} \neq \mathbf{0}$. Then, $\bar{\mathbf{v}}^T\mathbf{Av} = \lambda\bar{\mathbf{v}}^T\mathbf{v} = \lambda\sum_{i=1}^{n}(\text{abs}\ (v_i))^2$ But also $\bar{\mathbf{v}}^T\mathbf{Av} = \bar{\mathbf{A}\mathbf{v}}^T\mathbf{v} = \bar{\lambda}\sum_{i=1}^{n}(\text{abs}\ (v_i))^2$. So, we have $\lambda = \bar{\lambda}$, i.e., $\lambda \in \mathbf{R}$ and we can safely assume that $\mathbf{v} \in \mathbf{R}^n$.

It is the most important fact that there is a set of orthonormal eigenvectors of matrix $\mathbf{A}$ i.e., $\mathbf{q_1}, ..., \mathbf{q_n}$ such that $\mathbf{Aq_i} = \lambda_i\mathbf{q_i}, \mathbf{q_i}^T\mathbf{q_j} = \delta_{ij}$ which is zero and each eigenvector has unit length

such that $\mathbf{q_i}^T\mathbf{q_i} = 1$ or equivalently norm of each eigenvector is 1 $||\mathbf{q_i}|| = 1$. In matrix form: there is an orthogonal $\mathbf{Q}$ such that:

$$\mathbf{Q}^{-1}\mathbf{A}\mathbf{Q} = \mathbf{Q}^T\mathbf{A}\mathbf{Q} = \Lambda \tag{11}$$

Hence we can express $\mathbf{A}$ as:

$$\mathbf{A} = \mathbf{Q}^{-1}\Lambda\mathbf{Q} = \sum_{i=1}^{n} \lambda_i \mathbf{q_i}\mathbf{q_i}^T \tag{12}$$

Suppose the eigenvalues of $\mathbf{A}$ are sorted so $\lambda_1 \geq ... \geq \lambda_n$. Please note that $\mathbf{A}$ represents $(\mathbf{I} - \rho\mathbf{W})^T(\mathbf{I} - \mathbf{M})(\mathbf{I} - \rho\mathbf{W})$. Thus,

$$\begin{aligned}
\text{abs } \left(\mathbf{y}^T\mathbf{A}\mathbf{y}\right) &= \text{abs } \left(\mathbf{y}^T\mathbf{Q}^T\Lambda\mathbf{Q}\mathbf{y}\right) \\
&= \text{abs } \left((\mathbf{Q}^T\mathbf{y})^T\Lambda(\mathbf{Q}^T\mathbf{y})\right) \\
&= \text{abs } \left(\sum_{i=1}^{n} \lambda_i(\mathbf{q_i}^T\mathbf{y})^2\right) \\
&\leq \text{abs } \left(\sum_{i=1}^{n} \lambda_i\mathbf{y}^T\mathbf{y}\right) \\
&= n * \underbrace{\mathrm{E}[\lambda_i\mathbf{y}^T\mathbf{y}]}_{\text{per element SE}} \\
&= n * \mathrm{E}[\lambda_i(y_1^2 + ... + y_n^2)]
\end{aligned} \tag{13}$$

where E[.] represents expected value (i.e. average of eigenvalues). In the equation, all eigenvalues greater symmetric positive semi-definite matrix are positive and y_i^2 's of an order of magnitude which makes the natural logarithm of this end-result much greater than the log-det term. ■

APPENDIX VI
DERIVATION OF THE ML (LOG-LIKELIHOOD) FUNCTION

Ordinary least squares are not appropriate to solve for the models given in (2). One way to solve is to use the ML theory procedure. In probability, there are essentially two classes of problems: the first is to generate a data sample given a probability distribution and the second is to estimate the parameters of a probability distribution given data. Obviously in our case, we are dealing with the latter problem. This derivation not only shows the link between the need for eigenvalue computation and the SAR model parameter fitting but also explains how the SAR model works and can be interpreted as an execution trace of the solution for the SAR model. The end-result will be the log-likelihood function that is used in the optimization of SAR model parameter estimate ρ. We presented a simple overview of log-likelihood theory in Appendix III.

We begin the derivation by choosing a SAR model that is given in (2). We can explicitly write SAR model using its matrix-vector form as follows:

$$y_t = (\mathbf{I} - \rho\mathbf{W})^{-1}(x_{t1}\beta_1 + x_{t2}\beta_2 + ... + x_{tk}\beta_k + \epsilon_t) \tag{14}$$

where $t = 1, ..., n$ is the index for n successive observations. Let us assume that the disturbances or error ϵ_t is distributed normally, independently and identically with mean $E(\epsilon) = 0$ and variance σ^2. The set of n such equations can be compiled as equation (2). Let us assume that the disturbances ϵ_t, which are the elements of the vector $\epsilon = [\epsilon_1, ..., \epsilon_t, ..., \epsilon_n]$ and are distributed independently and identically according to a normal distribution as given in (15). Let's call the matrix $(\mathbf{I} - \rho\mathbf{W})$ as matrix $\mathbf{A}$ to simplify the expressions. Please note that $\epsilon_t = (\mathbf{A}y_t - x_{t.}\beta)$.

$$N(\epsilon_t; 0, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-1}{2\sigma^2}\ (\mathbf{A}y_t - x_{t.}\beta)\right) \tag{15}$$

If the vector ϵ has a multi-variate normal distribution just like in our case, the normal distribution is then defined as in (16) with a covariance matrix defined as $\Sigma = \sigma^2\mathbf{I}$. Please note that $|\Sigma|^{\frac{-1}{2}} = \sigma^n$, $\Sigma^{-1} = \frac{1}{\sigma^2}\mathbf{I}$ and $|\Sigma| = |\sigma^2\mathbf{I}| = \sigma^{2n}$.

$$\begin{aligned} N(\epsilon_t; 0, \Sigma^2) &= (2\pi)^{\frac{-n}{2}}|\Sigma|^{\frac{-1}{2}}\ \exp\left(\frac{-1}{2}\epsilon_t^T\Sigma^{-1}\epsilon_t\right) \\ &= (2\pi)^{\frac{-n}{2}}|\Sigma|^{\frac{-1}{2}}\ \exp\left(\frac{-1}{2}(\mathbf{A}y_t - x_{t.}\beta)^T\ \Sigma^{-1}(\mathbf{A}y_t - x_{t.}\beta)\right) \end{aligned} \tag{16}$$

Then, taking the $x_{t.}$ vectors which forms the rectangular matrix $\mathbf{x}$ of size n-by-k as data, the observations y_t (where $t = 1, ..., n$) have density functions $N(y_t; (\mathbf{A}y_t - x_{t.}\beta), \sigma^2)$ which are of the same form as those of the disturbances, and the likelihood function of β and σ^2, based on sample is given in (17) [18]. Thus, the prediction of the SAR model solution heavily depends on the quality of the normally distributed random numbers generated.

$$\begin{aligned} L(\theta|\mathbf{y}) &= L((\rho, \beta, \sigma^2)|(y_t, x_{t.}, \mathbf{W})) = \prod_{t=1}^{n} N(y_t; (\mathbf{A}y_t - x_{t.}\beta), \sigma^2) \\ &= N(\epsilon; 0, \Sigma^2)|d\epsilon/d\mathbf{y}| \\ &= (2\pi)^{\frac{-n}{2}} |\Sigma|^{\frac{-1}{2}} \exp\left(\frac{-1}{2}(\mathbf{A}\mathbf{y} - \mathbf{x}\beta)^T \Sigma^{-1}(\mathbf{A}\mathbf{y} - \mathbf{x}\beta)\right) |d\epsilon/d\mathbf{y}| \\ &= (2\pi\sigma^2)^{\frac{-n}{2}} \exp\left(\frac{-1}{2}(\mathbf{A}\mathbf{y} - \mathbf{x}\beta)^T \Sigma^{-1}(\mathbf{A}\mathbf{y} - \mathbf{x}\beta)\right) |d\epsilon/d\mathbf{y}| \end{aligned} \tag{17}$$

The *Jacobian* term $|d\epsilon/d\mathbf{y}|$ [12], [15] needs to be calculated out in order to find the probability density function of the variable $\mathbf{y}$, which is given in (18). Please note that $\epsilon = (\mathbf{A}\mathbf{y} - \mathbf{x}\beta)$ and the term $\Sigma^{\frac{-1}{2}}(\mathbf{A}\mathbf{y} - \beta)$ is also known as the vector of homoskedastic random disturbances [2], [42]. The Jacobian term is equal to the identity matrix $\mathbf{I}$ in classical linear regression model [2]. The need for the Jacobian term is formally stated and proved by Theorem 7.1 (Theorem 6.1 in this paper) on pages 232-233 of [15]. We provide the theorem and proof for the reader's convenience by converting to our notation.

$$|d\epsilon/d\mathbf{y}| = |\mathbf{A}| \tag{18}$$

Theorem 6.1: Let $N(\epsilon; 0, \Sigma^2)$ be the value of the probability density of the continuous random variable ϵ at ϵ_t. Since the function given by $\mathbf{y} = \mathbf{A}^{-1}\mathbf{x}\beta + \mathbf{A}^{-1}\epsilon$ is differentiable and either increasing or decreasing for all values within the range of ϵ for which $N(\epsilon; 0, \Sigma^2) \neq 0$, then for these values of ϵ, the equation $\mathbf{y} = \mathbf{A}^{-1}\mathbf{x}\beta + \mathbf{A}^{-1}\epsilon$ can be uniquely solved for ϵ to give $\epsilon = \mathbf{A}\mathbf{y} - \mathbf{x}\beta$ and the probability density of $\mathbf{y}$ is given by:

$$L(\theta|\mathbf{y}) = N(\epsilon; 0, \Sigma^2)|d\epsilon/d\mathbf{y}| \text{ provided } \mathbf{A}^{-1}\mathbf{x}\beta + \mathbf{A}^{-1}\epsilon \neq 0 \tag{19}$$

Elsewhere, $L(\theta|\mathbf{y}) = 0$.

Proof: The proof can be found on pages 233-235 of [15]. ■

$$L(\theta|\mathbf{y}) = (2\pi\sigma^2)^{\frac{-n}{2}} \ \exp\left(\frac{-1}{2\sigma^2}(\mathbf{Ay} - \mathbf{x}\beta)^T \ (\mathbf{Ay} - \mathbf{x}\beta)\right) \ |\mathbf{A}| \tag{20}$$

$L(\theta|\mathbf{y})$ given in (20) will henceforth be referred to as the "likelihood function of the SAR model". It is a probability distribution but now interpreted as a distribution of parameters which have to be calculated as noted in the Appendix III. Since the log function is monotonic and the log-likelihood function is unimodular (See SAR Unimodularity Theorem 4.1), we can then equivalently minimize the log-likelihood function, which has a simpler form and can handle large numbers. This is because the logarithm is advantageous, since $\ln(ABC) = \ln(A)+\ln(B)+\ln(C)$. After taking the natural logarithm of equation given in (20), we get the log-likelihood function given in (21).

$$\ell(\theta|\mathbf{y}) = \ln L(\theta|\mathbf{y}) = -\frac{n}{2}\ln(2\pi) - \frac{n}{2}\ln(\sigma^2) \ - \frac{1}{2\sigma^2}(\mathbf{Ay} - \mathbf{x}\beta)^T \ (\mathbf{Ay} - \mathbf{x}\beta) \ + \ln|\mathbf{A}| \tag{21}$$

The MLE estimators given in (22a) and (22b) are obtained by setting $\frac{\partial \ell(\theta|\mathbf{y})}{\partial \beta}$ and $\frac{\partial \ell(\theta|\mathbf{y})}{\partial \sigma^2}$ to zero respectively.

$$\hat{\beta} = (\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T \ \mathbf{Ay} \tag{22a}$$

$$\hat{\sigma}^2 = (\mathbf{Ay})^T(\mathbf{I} - \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1} \ \mathbf{x}^T)^T(\mathbf{I} - \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1} \ \mathbf{x}^T)(\mathbf{Ay})/n \tag{22b}$$

Replacing $\hat{\beta}$ given in (22a) with β given in (21) and $\hat{\sigma}^2$ given in (22b) with σ^2 given in (21) lead to equation given in (23) for the log-likelihood function (i.e. the logarithm of the ML function) to be optimized for ρ.

$$\ell(\rho|\mathbf{y}) = \underbrace{\ln|\mathbf{A}|}_{log-det} - \underbrace{\frac{n}{2}\ln(2\pi) - \frac{n}{2}\ln\left\{\frac{1}{n}\right\} - \frac{1}{2n}}_{constants} -$$
$$\frac{n}{2}\ln\underbrace{(\mathbf{Ay})^T(\mathbf{I} - \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1} \ \mathbf{x}^T)^T(\mathbf{I} - \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1} \ \mathbf{x}^T)(\mathbf{Ay})}_{SSE} \tag{23}$$

The first term in (23), i.e., the log-det, is nothing but the logarithm of the sum of a collection of scalar values including all of the eigenvalues of the neighborhood matrix $\mathbf{W}$ as given in (24).

$$|\mathbf{I} - \rho\mathbf{W}| = \prod_{i=1}^{n}(1 - \rho\lambda_i) \rightarrow \ \ln|\mathbf{I} - \rho\mathbf{W}| = \sum_{i=1}^{n}\ln(1 - \rho\lambda_i) \tag{24}$$

Hence, the final form of the log-likelihood function is given in (25) after ignoring constant terms in (23) and then multiplying the resulting equation with the constant $\frac{-2}{n}$.

$$\ell(\rho|\mathbf{y}) = \frac{-2}{n}\ln|\mathbf{A}| + \ \ln(\mathbf{A}\mathbf{y})^T(\mathbf{I} - \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1}\ \mathbf{x}^T)^T(\mathbf{I} - \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1}\ \mathbf{x}^T)(\mathbf{A}\mathbf{y}) \tag{25}$$

Therefore, the log-likelihood function $\ell(\rho|\mathbf{y})$ is optimized using single-variable optimization routine Golden Section Search to find the best estimates for ρ. Once the estimate for ρ is found, both β and σ^2 can be computed. Finally, the predicted variable ($\mathbf{y}$ vectors or thematic classes) can be computed using equation (26).

$$\mathbf{y} = (\mathbf{I} - \rho\mathbf{W})^{-1}(\mathbf{x}\beta + \epsilon) \tag{26}$$

Equation (26) needs a matrix inversion algorithm in order to get the predicted observed dependent variable $\mathbf{y}$. For small problem sizes, one can use exact matrix inversion algorithms; however, for large problem sizes (e.g., > 10K) one can use *geometric series expansion* to compute the inverse matrix in (26) as stated by Lemma 6.1.

Lemma 6.1:

$$(\mathbf{I} - \rho\mathbf{W})^{-1} = \sum_{i=0}^{\infty}(\rho\mathbf{W})^i$$

Proof: Since $\|\mathbf{W}\| \leq 1$ and $|\rho| < 1$, we have that $\|\rho\mathbf{W}\| < 1$. We then apply Lemma 2.3.3 on page 58 of [16]. (Please also see page 301 of [20]). ■

In practice, we truncate the sum to at most 30 terms, fewer if ρ is bounded away from 1.