Technical Report: alg04-004
April 8,2004

Tyler Alumbaugh, talumbau@ncsa.uiuc.edu
Peter Bajcsy, pbajcsy@ncsa.uiuc.edu

Automated Learning Group
National Center for Supercomputing Applications
605 East Springfield Avenue, Champaign, IL 61820

Georeferencing, Feature Extraction and Modeling From Raster and Point Data Information

Abstract

In this report we present an overview of the GIS processing steps for predictive modeling from raster and point data and we describe the software tools developed for this purpose. The GIS processing steps include data georeferencing, visualization, feature extraction from raster data over locations defined by point data, and predictive modeling of (a) cross-feature dependencies by a correlation matrix and (b) spatial dependencies by a kriging technique. The described work leverages from our previous development of georeferencing capabilities and is driven by agricultural and environmental applications. This document outlines the theory and implementation of each processing step and serves as a guideline for other scientists trying to work with geo-registered raster and point data.

1. Introduction

In this work, we address the problems of pre-processing raster and point data information followed by two types of modeling techniques. The problem of pre-processing includes georeferencing raster and point data, visualization of both data types, and feature extraction from raster data over locations defined by point data. The problem of variable prediction or modeling is understood as the search for a mapping/relationship/function for predicting one or more features from a set of multi-variate measurements. In our work, the prediction mapping forms a model for either (1) cross-feature dependency by focusing only on generic features or (2) spatial/temporal dependency by processing a pair of spatial/temporal coordinates and associated feature information. Variable prediction problems can be found in many GIS applications, such as, hydrology, water quality research, environmental science, socio-economics modeling or atmospheric science.

A prediction of the cross-feature dependency typically does not consider any spatial or temporal information and its goal is to establish relationships among physical variables (or more precisely observations) of the same phenomenon. A few examples of such prediction modeling techniques would include cross-correlation matrices, least-square fit techniques with a built-in model, neural networks, or regression trees.

A prediction of the spatial/temporal dependency is used for modeling one or more variables at any spatial location and time from a spatially and/or temporally sparse set of measurements. This type of prediction usually involves spatial and/or temporal interpolation or extrapolation and its goal is to build a dense set of values while minimizing the cost of performing dense measurements. The underlying assumption for spatial and temporal predictions is the fact that most physical entities change their value continuously and can be approximated reasonably well with interpolation and extrapolation models. We would list the least-square polynomial, bilinear or B-spline models as the most popular interpolation and extrapolation prediction models.

From an application viewpoint, accurately predicting a variable, whether from another variable or from a spatially sparse set of its own values, is an incredibly useful ability. Hydrologists use spatial prediction techniques to model underground water sources [1]. Mining companies hope to accurately predict concentrations of a particular mineral when given only a small number of sample points [9]. The application drivers for our research and development came from agricultural and environmental domains while studying crop yield and water quality models.

The motivation for our modeling work comes from the fact that there exist feature-rich and feature-sparse data spaces, as well as, spatially/temporally dense and spatially/temporally sparse spaces. It is our objective to maximize our information gain in any given data space and hence we are looking for various modeling techniques to achieve our objective. In addition, the need for combining raster and point data in both agricultural and environmental application domains was eminent and the availability of flexible software tools is scarce.

In our work, we assume that in a feature-rich data space, we have a large number of known features throughout the space (e.g. elevation, average rainfall, etc.), as well as a variable (known only in some portion of the space) that we would like to predict. To be more specific, we assume that the variable we are interested in is specified in a DBF file [8]. The other known features are to be extracted from any number of image (raster) files from within a software package called I2K [4]. Thus, for modeling cross-feature dependencies, we implemented a software tool that allows users to perform feature extraction, insertion of data to the DBF file, computation and display of the correlation matrix of any number of variables.

Our work in a feature-sparse data space is based on a form of ordinary kriging [5]. We use only the given data of a variable to predict its value at other spatial points. Specifically, we use our given data set to compute the method-of-moments variogram estimator (or 'classical' variogram estimator). Next, we fit an isotropic model variogram to our estimator by a least squares approximation. Finally, we calculate our weights for the prediction of a particular value by solving a standard linear system using our modeled variogram.

2. Current Solutions

The ArcGIS software package contains a flexible Kriging tool. The Spatial Analyst tool used inside of the ArcMap program [2] gives the user the ability to analyze their point data in DBF form and form a raster image containing predictions of an attribute in an area around the point data. The dialog is shown in Figure 1.

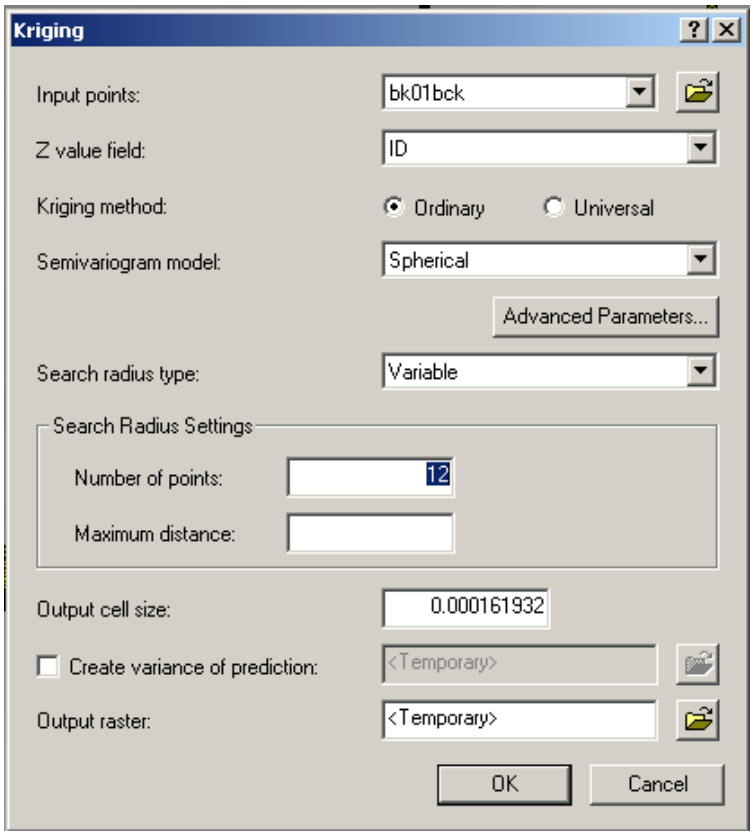

Figure 1: ArcMap Dialog.

The available semivariogram models are as follows: spherical, circular, exponential, Gaussian, and linear. The "Z value field" drop-down list allows the user to choose which column of the DBF file contains the attribute they would like to predict. As we explain in Section 4.2.2, it was a difficult task to find a proper semivariogram for the particular data set coming from our agricultural application domain and, indeed, the semivariogram options here do not provide the model we eventually used. Thus, the results obtained from this tool were consistent with our earlier kriging attempts. Namely, the prediction raster images produce by Spatial Analyst contained large negative numbers, which should not be possible in this dataset with a statistically valid semivariogram. In Section 4.2, we detail our approach and give references to the work that motivated our choice of semivariogram.

3. Preprocessing of Raster and Point Data Information for Modeling

The dialog box shown is the interface by which to access all of the developed software. A brief overview of its components follows:

Load DBF - This is the first button the user must click to begin using the tool. An 'Open File' dialog box launches where the user can select the DBF file for analysis.

ShowDBF – The loaded DBF file is read into a standard table structure used in [6]. Clicking this button launches a TableViewer, allowing the user to inspect the DBF.

ShowGeoPts – This button allows the user to view any locations indicated in the DBF file that are within the spatial boundaries of the image in the main frame.

Extract – Here the user can extract the value of every pixel in the main image, which coincides with a point in the loaded DBF file.

Insert2DBF – This will insert a column into the DBF table containing the values of the extracted feature.

SaveDBF – A button to write out the modified DBF.

ListFeatures – Lists the column number and title of every column in the DBF

Correlate – This button brings up a dialog where the user can choose which features (columns of the DBF file) to correlate. After the features are chosen, a coefficient matrix is calculated.

ShowCorrelate – Here the results of the correlation calculation are displayed.

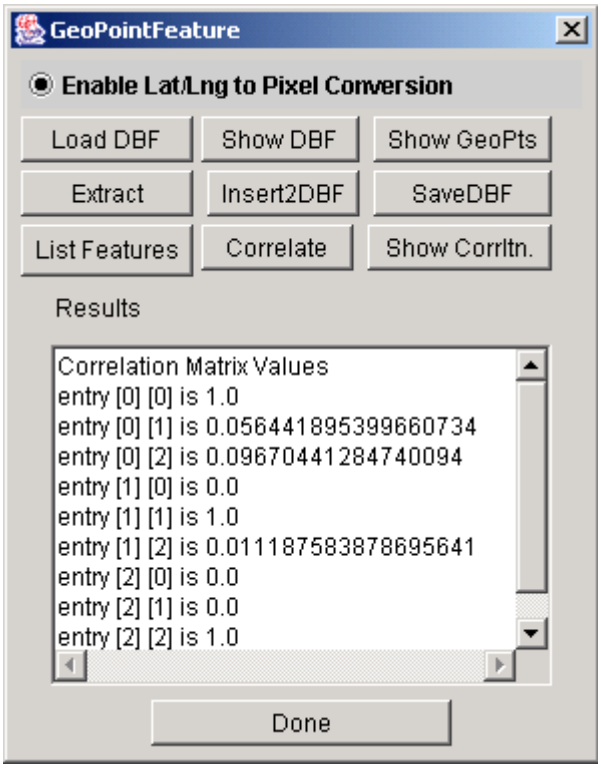

Figure 2:GeoPoint Dialog

3.1 Georeferencing Raster and Point Data

This pre-processing step is based on two assumptions. First, an image (raster data) is loaded with its georeferencing information. Our current software supports several geographic projections, including the Lambert Azimuthal Equal Area projection and the Universal Transverse Mercator projections for all zones in the Northern Hemisphere. The loading and display of images can be achieved using the main frame of the I2K application (see I2K documentation at [4]). Second, a table of point data in a DBF file format (version 4.0) with multiple rows and columns has to contain (a) column heading with the two required fields defined as “LATITUDE” and “ LONGITUDE”, and (b) two columns with latitude and longitude information in each row, where all other columns correspond to measured variables at that particular geographic location. We do not support currently files with six columns defining latitude and longitude in degree, minutes and seconds. The loading of tabular data in a DBF file format can be executed from the GeoPoint dialog shown in Figure 2 by selecting “Load DBF”. After loading the DBF file, the user can see either the tabular information by clicking “Show DBF” or the overlaid geographic location of all points by clicking “ShowGeoPts”. The spatial information about each point in latitude and longitude is converted to pixel information in row and column by using the GeoConvert object. This object, through calls to its

LatLng2ColumnRow method, allows us to associate a pixel location with each point in our data set.

3.2 Feature Extraction

The goal of this pre-processing step is to perform feature extraction from raster data over locations defined by point data. Extracting a feature from a given image is accomplished through the use of the "Extract" button. For each point in the loaded DBF file, the raster value of its associated pixel is copied to an array. This array is inserted into the tabular information by clicking "Insert2DBF". Finally, the extracted information can be saved into a new DBF file by clicking "SaveDBF". The user can then load a different image, and repeat the process. This ability is particularly useful for extracting features from multiple images. Additionally, the user can see if the loaded DBF points represent locations seen within the image by clicking the "Show GeoPts" button.

4. Multi-Dimensional Multi-Variate Modeling

4.1 Modeling Cross-Feature Dependencies

We have implemented a computation of the correlation matrix as defined in [7], Pattern classification, p 614. The correlation value for every pair of features indicates the statistical dependency of features and can be utilized for predicting the outcome of complex relationships. For example, if two features are not correlated (in other words the two features (random variables) are statistically independent) then the expected value E of the product of two features could be computed by multiplying the expected values of each feature $\mathrm{E}(v_1 * v_2) = \mathrm{E}(v_1) * \mathrm{E}(v_2)$. The expected value and standard deviation would be used as statistical descriptors of each feature.

By clicking on the Correlate button in GeoPoint Dialog (see Figure 2), the user can calculate the correlation coefficient for any number of features. It is assumed that the features are all scalar values. As of yet, the tool does not support any method for dealing with nominal features.

The correlation coefficients for the selected features appear in the text area of the GeoPoint Dialog below the buttons. Additionally, the coefficient matrix is formed and displayed for the user. This matrix follows the convention of a bright color entry indicating high correlation and increasingly dark colors for increasingly uncorrelated features. This correlation matrix is obviously symmetric, so only the lower diagonal is shown. See Section 5.1 for figures and examples.

4.2 Modeling Spatial Dependencies

In this section, we focus on modeling spatial dependencies by a kriging technique. The problem of spatial modeling can be described as a spatial prediction problem from a

given set of 2D points and their variable values inside a geographic domain. The goal is to use these data points to predict the value of the variable throughout the relevant spatial domain.

Kriging is a topic covered in many textbooks [5], and our description of kriging gives only a brief summary of the concepts relevant for this work. The most important tool is the variogram, or more correctly the semivariogram, denoted as γ. This function takes a vector value representing the difference between two locations in the data, usually called a 'lag.' Its value for a given lag represents how much the feature can be expected to vary over that lag distance and direction. This is a tremendously powerful tool, as a data set may vary much in one direction (say the north-south direction) and hardly at all in another (say the east-west direction). A semivariogram that takes into account these differences in direction is called an anisotropic semivariogram. Another type of semivariogram, called an isotropic semivariogram, is a function of only the magnitude of the lag distance. As one would imagine, this is significantly easier to implement. For this work, we restricted our search for semivariograms to several isotropic models.

In order to perform ordinary kriging, we had to first make decisions on the type of variogram we wanted to use and how we were going to calculate this variogram to use in our predictions. In the interest of simplicity, we chose a variogram that is isotropic. That is, we process only the magnitude of the lag vector, but not its direction. This means our variogram is a scalar function of a scalar variable, which makes its estimation much easier.

4.2.1 Variogram Estimation

The notation for this section follows closely with [5]. The variable we are trying to predict is collectively denoted as **Z**. Our set of locations is $\{s_1,...,s_n\}$. The value of our variable at a location s_i is $Z(s_i)$. The prediction of our variable at a location s_0 is denoted as $p(Z,s_0)$.

Method-of Moments Estimator

To begin to understand the characteristics of the variogram associated with a data set, it is a general practice to estimate it by some means. The approach taken in this work is to use the standard method-of-moments variogram estimator, also known as the 'classical' variogram estimator. The formula for the 'classical' variogram estimator is:

$$2\hat{\gamma}(h_k) = \frac{1}{|N(h_k)|}\sum_{N(h_k)} (Z(s_i) - Z(s_j))^2, \;\; h \in R^d$$

where the set of pairs, $N(h_k)$, is defined as:

$$N(h_k) \equiv \left\{(s_i, s_j) : Dist(s_i, s_j) = h_k : i, j = 1,..., n\right\}$$

Throughout, we will usually denote $\hat{\gamma}_k = \hat{\gamma}(h_k)$ as the estimator at a distance h_k. From a mathematical viewpoint, the set $N(h_k)$ is well defined. However, from a computational viewpoint, the size of the set $N(h_k)$ for a given h_k may be quite small or even zero unless we process extremely large data sets. Thus, we must use reasonable values of h_k and a small tolerance Δ_t before computing parameters of the classical estimator. The issue related to populating the set $N(h_k)$ for a chosen h_k is described next.

Populating $N(h_k)$: To find reasonable values of h_k, we iterate through some portion of our data and record the distances between a particular point and several of its neighbors. The location of each of the points is given in latitude and longitude. If the distances are computed for all point pairs then this computation requires $\frac{n \cdot (n-1)}{2}$ distance calculations. For the distance metric, we briefly investigated the feasibility of approximating the distances between any two points in meters using a geodetic ellipsoid, but it seemed to work well to just use the standard Euclidean distance. Obviously, the Earth is not a Euclidean surface so we introduce some error here, but the approximation is certainly valid for sufficiently small areas in which we are usually interested. We find the minimum and maximum distance between points in the sample of data and use those values as the minimum and maximum values of h for our estimator. By deciding on a rather arbitrary number of total points for the estimator (in this case, K=10), we were able to choose the rest of the values of h_k for our estimator and then calculate each $\hat{\gamma}_k$.

To populate $N(h_k)$ for each of our values of h_k, we could not insist that the distances between two points be exactly h_k. The distance between any two points was stored as a double precision floating point number, so it is unlikely that this distance would agree with any value of distance h_k up to its highest precision. Thus, a tolerance value should be chosen and refined as needed. There is no concrete rule for the size of each set $N(h_k)$, which we denote as *P*. Following the advice of Cressie [5], our desire is to have $P \geq 30$ pairs of points such that the distance between each pair falls within the tolerance for each value of h_k.

$$\left\| Dist(s_i - s_j) - h_k \right\| < \Delta_t$$

First, we compute distances for a small subset of points (perhaps 10-25% of the data) and then compute the minimum and maximum distance values. Second, we partition the interval of [min, max] into K-1 sub-intervals as it is illustrated in Figure 3. The partition should, in general, be evenly spaced throughout [min, max], but fine-tuning may be necessary to achieve the desired number of pairs for each lag distance.

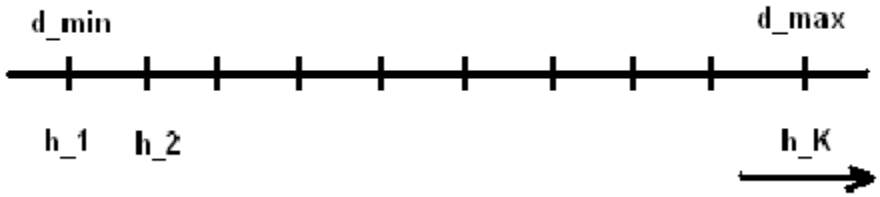

Figure 3: Finding K distance values.

4.2.2 Selecting and Fitting the Model

Given a set of estimated values $\hat{\gamma}_k = \hat{\gamma}(h_k),\ k = 1,...,K,\ K = 10$, the next step is to choose a continuous model variogram and fit the model's parameters to the estimated values. There is great freedom of choice for this step, and the success of the kriging process is mostly determined by the choice of semivariogram model. A semivariogram must have certain properties to be statistically valid (see [5] for details). For example, a valid semivariogram must always give nonnegative answers for positive values of a lag distance *h*. For our proof-of-concept kriging model (see Section 5.3), we chose a simplified power model for the semivariogram:

$$\gamma(h) = a^2 \|h\|^{\alpha}$$

By design, this semivariogram will always be positive.

4.2.3 'Linear' Least Squares Fitting to Power Model

To fit data to our model, we seek values for parameters *a* and α that predict $\hat{\gamma}_k = \hat{\gamma}(h_k),\ k = 1,...,K,\ K = 10$ as close as possible. We choose to fit our model in a linear least squares fashion, which we can do with some manipulation of our original formula. Note that for some *h*, we would like our model to be such that

$$\hat{\gamma}(h) = a^2 \|h\|^{\alpha}$$

$$\ln(\hat{\gamma}(h)) = \ln(a^2 \|h\|^{\alpha})$$

$$\ln(\hat{\gamma}(h)) = 2\ln(a) + \alpha \ln(\|h\|)$$

Writing this equation in vector form yields:

$$\begin{bmatrix}\ln(\|h\|) & 1\end{bmatrix}\cdot\begin{bmatrix}\alpha \\ 2\ln(a)\end{bmatrix} = \ln(\hat{\gamma}(h))$$

Now, we find values for α and 2ln(a) that minimize the two-norm of the residual of the following linear system:

$$\begin{bmatrix}\ln(\|h_1\|), 1 \\ \vdots \\ \ln(\|h_{10}\|), 1\end{bmatrix}\cdot\begin{bmatrix}\alpha \\ 2\ln(a)\end{bmatrix} = \begin{bmatrix}\ln(\hat{\gamma}(h_1)) \\ \vdots \\ \ln(\hat{\gamma}(h_{10}))\end{bmatrix}$$

Using QR factorization, we solve for this α and 2ln(*a*). This is our valid semivariogram used to perform our predictions.

4.2.4 Predicting a Value

Setting up Linear System

Following closely with Cressie [5], Sec. 3.2, we seek to solve the following linear system:

$$\Gamma\lambda = \gamma_0$$

where Γ, γ_0, and λ are as follows (recalling that s_0 is our value to predict):

Γ : an $n \times n$ matrix whose (i^{th}, j^{th}) entry is $\gamma(s_i - s_j)$

γ_0 : an $n \times 1$ column vector whose i^{th} entry is $\gamma(s_0 - s_i)$

λ : an $n \times 1$ column vector where each entry is a weight to be used in the final prediction

Solving Linear System

For some data sets, it may be too costly to use the entire data set in the prediction process (i.e. $n = |Z(s)|$). For such cases, it may be more computationally feasible to sample the data throughout the space and use this sample to form the matrix Γ (and corresponding vectors γ_0, λ). After solving the system, we end up with a column vector of weights, λ. We multiply the transpose of this vector by a column vector of our *n* points (however they are chosen) as follows:

$$p(Z, s_0) = \lambda' \cdot Z, \quad Z = \begin{bmatrix} Z(s_1) \\ \vdots \\ Z(s_n) \end{bmatrix}$$

This yields the prediction, $p(Z, s_0)$.

5. Experimental Results

5.1 Input Data

For this work, we used a DBF file containing data on approximately 53,000 latitude/longitude points on a farm in central Illinois. The data was collected by harvesting machines equipped with GPS systems in August 2003. As the machines drove through the fields, they were able to record, among other things, the volume of crop harvested at that location (hereafter referred to as 'yield'). We are interested mostly in the yield data, which was only taken for about 18,000 of the 53,000 points.

We also worked with a portion of the National Elevation Dataset for the state of Illinois. Specifically, we selected a sub-area of this data that includes the land from where the yield data was collected. We used previously developed tools in I2K to verify that we sub-area did indeed represent the same land area as in the DBF file.

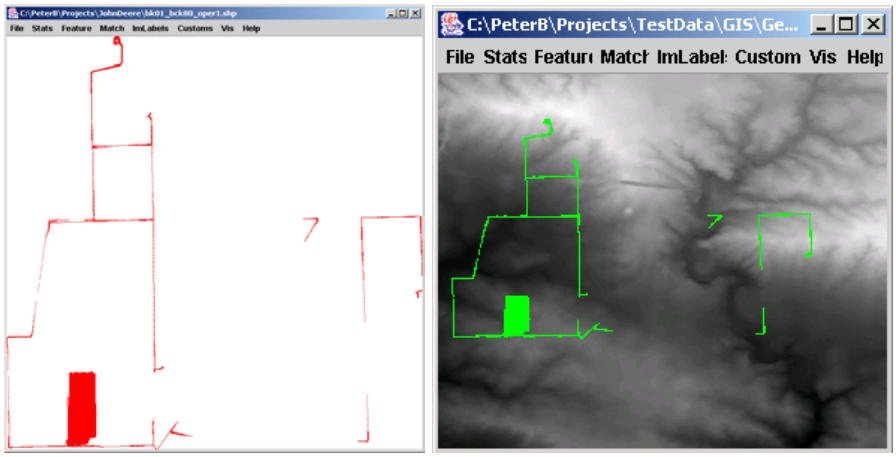

Figure 4: Visualization of shape information (left) and overlaid shape and raster data (right). The raster data is a digital elevation map from USGS.

5.2 Modeling Cross-Feature Dependency Using Correlation

As a simple example, we show the correlation matrix for several numerical features of the DBF file as well as the extracted elevation data, which we add to the DBF. The user follows the process described in Section 3, and chooses to correlate the features of latitude, longitude, moisture, dry_yield, and elevation (the extracted field). Clicking the Show GeoPts button shows our location in the image (Figure 5).

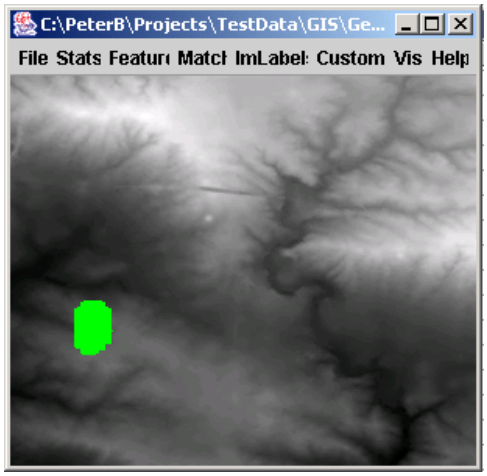

Figure 5: Visualization of selected point data for extracting DEM features.

DBF Table

	MOISTURE	STATUS	PASS	FIELD	CROP	VARIETY	DRY_YIELD	column_13	DIS
9213	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	161.2339	240.14295959472656	
9214	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	172.7686	240.14295959472656	
9215	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	174.559	240.14295959472656	
9216	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	169.1786	240.14295959472656	
9217	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	173.6661	240.14295959472656	
9218	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	179.4998	240.14295959472656	
9219	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	166.4861	240.14295959472656	
9220	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	171.4223	239.9219512939453	
9221	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	177.2888	239.9219512939453	
9222	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	162.7792	239.9219512939453	
9223	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	171.8477	239.9219512939453	
9224	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	171.8477	239.9219512939453	
9225	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	169.5806	239.9219512939453	
9226	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	161.4189	239.9219512939453	
9227	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	171.8477	239.9219512939453	
9228	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	162.3258	239.9219512939453	
9229	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	170.9408	239.9219512939453	
9230	21.19	1	3	F1: Becker ...	Corn	Pioneer 32...	168.3406	239.9219512939453	

Figure 6: Extracted DEM features are inserted into the original tabular information under the column heading "column_13".

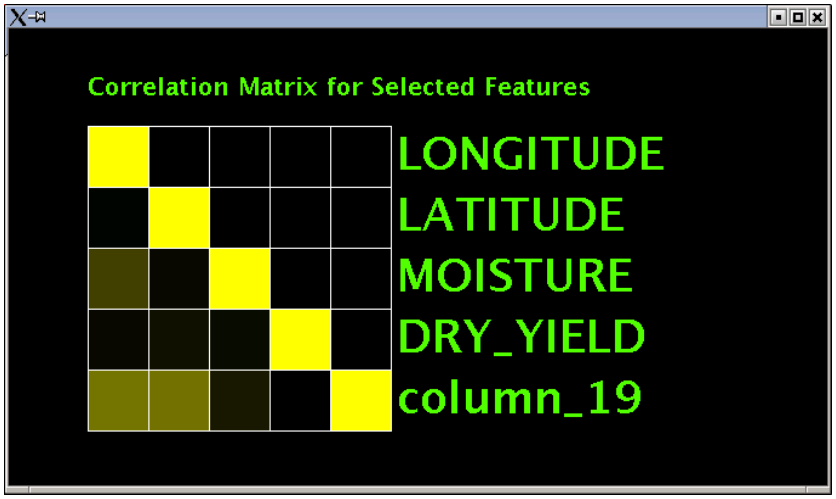

Figure 7: Correlation matrix for three selected variables.

Figure 7 demonstrates nicely how the user can immediately discern any relationship between the selected features. Numerical values are also printed in the dialog box (see Figure 2). The significant correlation calculations are as follows: corr(moisture, dry_yield)= 0.056441895399660734, corr(moisture, elevation)= 0.09670441284740094, and corr(elevation, dry_yield)= 0.011187583878695641, corr(longitude, moisture) = 0.24967296821023308, corr(longitude, elevation) = 0.4619475583873144, corr(latitude, elevation) = 0.44534181231501924. The brighter colors in the lower left of our correlation matrix can be explained by the fact that the elevation would be somewhat correlated to position in a relatively flat farm field with a gentle slope. Our conclusion is that none of the correlated features by themselves would be a good predictor for the yield data.

5.3 Modeling Spatial Dependency Using Spatial Kriging

Here we present a 'proof-of-concept' implementation of kriging using the yield data. As in Section 4.2, we begin with the semivariogram estimator. Based on our analysis for M=1000, P=30, and $\triangle_t = 5 * 10^{-6}$, the values of h_k were chosen were as follows:

h_k k = 1, …, 10	$\hat{\gamma}_k = \hat{\gamma}(h_k)$
h_1 = .0002	234.633333333333
h_2 = .0008	863.233333333333
h_3 = .0014	1986.8
h_4 = .002	1304.56666666667
h_5 = .0025	3944.93333333333
h_6 = .003	10031.1666666667

$h_7 = .0035$	16754.8666666667
$h_8 = .004$	19175.1
$h_9 = .0045$	16880.1
$h_{10} = .005$	18168.9333333333

Using the techniques described in Section 4.2.1, we used the crop yield data to construct the classical semivariogram estimator, $\hat{\gamma}(h)$, show in Figure 8:

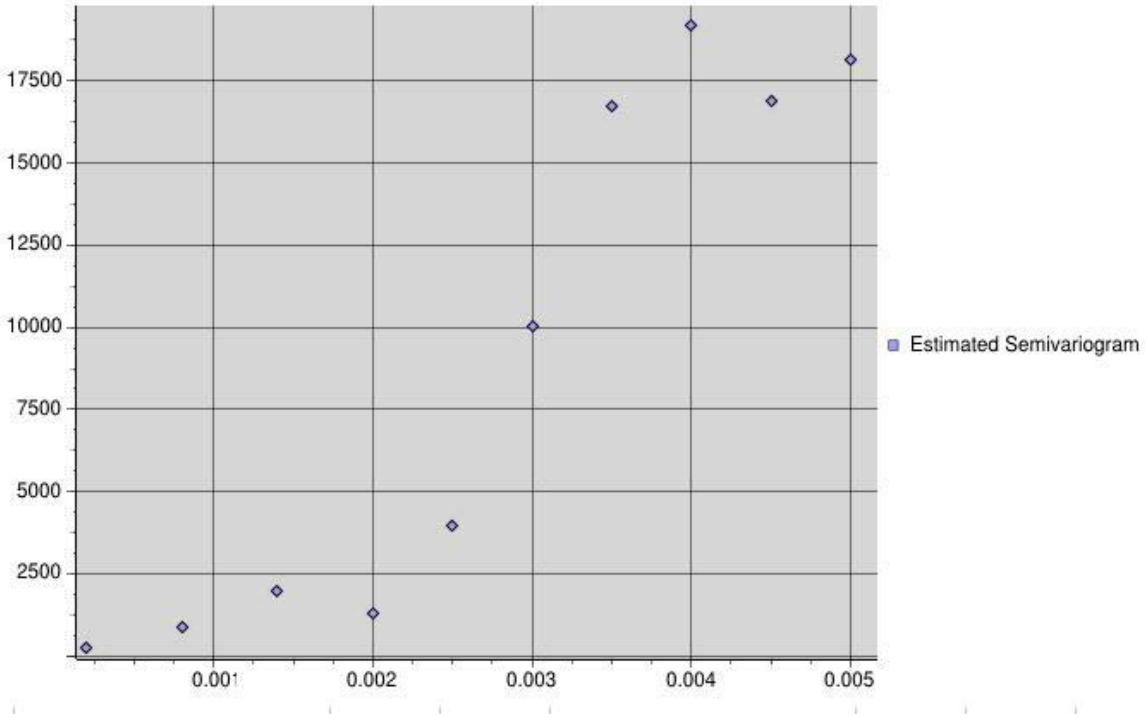

Figure 8 The classical semivariogram estimator for crop yield data

It turned out to be difficult to choose a good model to fit this estimator. For many of the models we tried, fitting the parameters to the data (using a least squares method) resulted in a variogram that did not obey one of essential properties of a variogram. Specifically, we would often end up with a semivariogram that would give a negative value for extremely small values of h. Using the model described in Section 4.2 solved these problems. Actually, others have already suggested using this type of semivariogram for agriculture yield data. Cressie [5] points readers to Whittle [10], but Whittle develops his covariance models initially in [11], using an isotropic model as well.

As stated above, our data consists of nearly 18,000 separate locations and yield values, so it would be quite costly to solve a linear system of that size. Further, it seems that the density of this data set greatly affected the conditioning of the linear system, making it nearly singular when nearly all of the data was used. The sensitivity was so great that our linear solver was unable to provide a solution. We eventually found that by using 600 of

the points distributed throughout our space (n = 600), the resulting linear system was well-conditioned and fairly cheap to solve.

To test the functionality of the prediction method, we predicted the yield values for a few points already in our data set. Thus, we could compare the actual values with what our prediction gave. Here is a table of our arbitrarily chosen points, their actual yield values and the predicted values:

Latitude	Longitude	Actual Yield	Predicted Yield
40.408634	89.074655	135	150.01
40.413923	89.074691	147	128.85
40411642	89.074255	180	158.48
40.410427	89.075334	172	155.83
40.411624	89.074609	180	161.18

Our initial results indicate that the prediction method does indeed yield sensible values, but our model parameters might need to be adjusted to obtain better accuracy.

6. Summary and Future Work

We have developed the necessary infrastructure in I2K to support predictive modeling for both cross-feature dependent variables and spatially dependent variables. Leveraging our early georeferencing work [3], I2K can now be a powerful tool for exploring the relationships among multiple types of spatial data. Additionally, we have built a proof-of-concept kriging tool that shows the potential for the software to predict unknown values of spatially dependent geographic data.

With regard to the correlation analysis, the tools developed stop short of predicting the spatial variable based on the correlation results. Further tools could be developed which make predictions based on highly correlated features. In addition, we could incorporate domain knowledge of nominal features (vegetation type, for example) into the predicting process. For the more statistical work, a more automated system of estimating and modeling the semivariogram will be needed to make the process easier for the user. However, we must first assess the accuracy of the prediction by a cross validation process.

References

[1] Ahmed, S. and de Marsily, G. (1987). Comparison of geostatistical methods for estimating transmissivity using data on transmissivity and specific capacity. *Water Resources Research.* V. 23.

[2] ArcGIS by ESRI. ESRI web site: http://www.esri.com
[3] Bajcsy, P. and Alumbaugh, Tyler, 2003.. Georeferencing Maps With Contours. *Journal proceedings of 7th World Multiconference on Systemics, Cybernetics, and Informatics. Vol X.* July 27-30, 2003. Orlando, Florida
[4] Bajcsy, P. et. al., 2004. Image To Knowledge (I2K), Software Overview and Documentation, Automated Learning Group(ALG) at the National Center for Supercomputing Applications (NCSA), University of Illinois at Urbana-Champaign, IL, URL: http://alg.ncsa.uiuc.edu/tools/docs/i2k/manual/index.html
[5] Cressie, Noel A.C. *Statistics for Spatial Data*. Wiley & Sons, Inc. 1993
[6] Data2Knowledge (D2K). Software Documenation at http://alg.ncsa.uiuc.edu/do/tools/d2k
[7] Duda, R., P. Hart and D. Stork, 2001. *Pattern Classification,* Second Edition, Wiley-Interscience.
[8] ESRI, 1998. ESRI Shapefile Technical Description. URL: http://www.esri.com/library/whitepaper/pdfs/shapefile.pdf
[9] Krige, D.G. (1951). A statistical approach to some basic mine valuation problems on the Witwatersrand. Journal of the Chemical, Metallurgical and Mining Society of South Africa, V. 52.
[10] Whittle, P. "Topographic Correlation, power-law covariance functions, and diffusion." *Biometrika*. V. 49, 305-314
[11] Whittle, P. "On the Variation of Yield Variance with Plot Size. *Biometrika*. V. 43, 337-343.