American Journal of Software Engineering and Applications
2013; 2(2) : 32-39
Published online April 2, 2013 (http://www.sciencepublishinggroup.com/j/ajsea)
doi: 10.11648/j.ajsea.20130202.12

An approach to modeling domain-wide information, based on limited points' data – part I

John Charlery, Chris D. Smith

Dept. of Computer Science, Mathematics & Physics, Faculty of Science and Technology, University of the West Indies, Cave Hill Campus, Bridgetown, Barbados BB11000

Email address:
john.charlery@cavehill.uwi.edu (J. Charlery),chris.smith@mycavehill.uwi.edu (C. D. Smith)

To cite this article:
John Charlery, Chris D. Smith. An Approach to Modeling Domain-Wide Information, based on Limited Points' Data – Part I, *American Journal of Software Engineering and Applications.* Vol. 2, No. 2, 2013, pp. 32-39. doi: 10.11648/j.ajsea.20130202.12

Abstract: Predicting values at data points in a specified region when only a few values are known is a perennial problem and many approaches have been developed in response. Interpolation schemes provide some success and are the most widely used among the approaches. However, none of those schemes incorporates historical aspects in their formulae. This study presents an approach to interpolation, which utilizes the historical relationships existing between the data points in a region of interest. By combining the historical relationships with the interpolation equations, an algorithm for making predictions over an entire domain area, where data is known only for some random parts of that area, is presented. A performance analysis of the algorithm indicates that even when provided with less than ten percent of the domain's data, the algorithm outperforms the other popular interpolation algorithms when more than fifty percent of the domain's data is provided to them.

Keywords: Data Modeling, Interpolation, Data Prediction, Sparse Data Analysis

1. Introduction

There are numerous methods for making predictions based on sparse data. Some of these methods provide well-documented forms of interpolation between the known data points. Among the more popular approaches are methods such as Inverse Distance Weighting to a Power [1), Kriging [2, 3, 4, 5], Triangulation with Linear Interpolation [6], Natural Neighbour [7], and Minimum Curvature [8]. In should be noted that in this work, the term prediction and its other derivatives, are used with a purely statistical definition and is not meant to allude to a future event.

Although it would be very handy to predict the future behaviour of events, in this study, the concentration is to address the issue of predicting existing domain-wide information when the data required to draw such conclusions are significantly lacking. In setting up the premise, the strengths and weaknesses of the popular methods just mentioned, are drawn upon to formulate an algorithm which will address the stipulated intent.

The primary goal of this work therefore, is to develop an algorithm for making predictions over an entire domain area, where data is known only for some random parts of that area. In other words, to provide domain-wide information, based upon sparse and random point data. The fundamental approaches used by other authors such as Fisher [1], Stein [4], Lee [6], Laurent [7] and Smith [8] among others, have been to use physical rules and/or interpolations to produce the data values of the missing locations. This work seeks to go further to refine this approach by incorporating the historical data from all the available data points (or what will be hitherto referred to as "stations") in the domain area and to try to establish relationships in a pair wise fashion between all the stations. After these relationships are established, then a determination of the strengths or weaknesses of the paired stations are determined. Equations defining the relationships will then be created for each pair of data stations based on the historical data, so that it would be possible to predict the value of one station when given the value of the other station that makes up the pair.

Algorithms will be developed to determine which pairs of data stations would be selected to facilitate the global (or domain-wide) predictions. (We define a domain or universe, as the geographical area within whose physical boundaries that data value predictions will be made.) These algorithms would also use the strength factors of the selected pairs to make the most accurate (or strongest) predictions.

(Heretofore, the term 'global' will be used to represent coverage of the entire defined domain.)

In this first part of the research (Part I), we present the background and algorithms to the proposed methodology. Implementation of the methodology, through two very contrasting cases studied, is presented in the second part of the research (Part II).

2. Methodology

The foundations to the process of effecting the global predictions are therefore as follows:

1. Every station within the domain is paired to each other.
2. Each paired station is plotted on a Scatterplot diagram, to determine what type of relationship (if any) exists between the two stations.
3. The strength or weakness of the association between the stations is calibrated.
4. An equation is developed to compute the value of one station when the value of its pair is known.
5. As the stations' values become known, these known stations would be used as the basis for further predictions of the unknown stations. The prediction algorithms then incorporate this 'new' data to determine which predicted value is likely to be the most accurate for any station within the domain.

In order to accomplish this, the methodology employs the techniques of Scatterplots, Pearson's Correlation, Linear Regression and the Straight line equation.

3. Established Interpolation Methods

Over the years, many methods have been developed for the interpolation of unknown values based on known ones which lie in the region of interest. There are advantages and disadvantages to each method. The strengths and weaknesses of the different approaches depend on the nature of the data that is being examined and hence the accuracy of the different methods varies depending on the dataset being used. Following is a list and brief description of some of the more popular interpolation methods that are used extensively today.

3.1. Inverse Distance Weighting to a Power

The Inverse Distance Weighting to a power method is implemented through different strategies. In this work the method proposed by Shepard [9] is used to implement the Inverse Distance Weighting (IDW). IDW is a simple method of assigning values to unknown points by using values from known points. The influence on an unknown point is inversely proportional to the distance between that unknown point and a point that is known.

3.2. Kriging

Sierra [10] describes Kriging as a modified linear regression technique that estimates a value at a point by assuming that the value is spatially related to the known values in a neighborhood near that point. Kriging computes the value for the unknown data point using a weighted linear sum of known data values. The weights are chosen to minimize the estimation error variance and to eliminate any bias in the sampling. Unlike other techniques for scalar values, kriging bases its estimates upon a dynamic neighborhood point configuration and treats those points as regionalized variables instead of random variables. Regionalized variables assume the existence of a region of influence on the data.

3.3. Triangulation with Linear Interpolation

As Lee [6] showed, one of the most common triangulation method is the Delauney triangulation. In triangulation, the known data points have lines drawn between them in such a way so as to create a network of triangles. When these triangles are formed, no triangle's edges are intersected by any other triangle. The result is a patchwork of triangular faces over the extent of the data region. These triangles look somewhat like a tiled mosaic over the region of interest.

Each triangle may have several data points with unknown values within it. The values of the unknown points which lie in the triangle is determine by using a weighted average of the vertices of that specific triangle. Each triangle creates a data plane, which has tilt and elevation so the value of the known points would depend on where it is located in the triangle.

Triangulation with linear interpolation works best when the known points are evenly distributed over the entire region of interest. However, it should be noted that unknown points, which lie outside of the network of triangles, would not have values predicted for them.

3.4. Natural Neighbour

Natural Neighbour is a weighted average technique that is based on the Voronoi tessellation. One definition of the Voronoi tessellation given by Parag [11] is "the partitioning of a plane with n points into n convex polygons such that each polygon contains exactly one point and every point in a given polygon is closer to its central point than to any other". The Voronoi tessellation polygon network is constructed from the triangulation of the data points.

The vertices of the Voronoi Tessellation polygons, correspond to the centroids of the circumcircles of the triangles connecting the known data points. A circumcircle is a triangle's circumscribed circle. In other words, it is the unique circle that passes through each of the triangle's three vertices.

The way natural neighbour interpolation works to get an unknown value for a point x, is by inserting the point x into it's actual location in the defined network. After point x is inserted, the Voronoi Tessellation polygon network must be rearrange to accommodate this new point, since the new point must have its own Voronoi Tessellation polygon. This

accommodation is facilitated by shrinking the polygons, which surround the new polygon. The value of x is then determined from the proportion of the intersections of the original polygons surrounding the newly inserted one.

3.5. Minimum Curvature

Smith [8] describes Minimum Curvature as a technique, which generates an interpolated surface that is like a thin elastic plate, which passes through each of the known data points with a minimum amount of bending. Minimum curvature tries to produce the smoothest possible surface while attempting to honour the known data as closely as possible.

In this method a two-dimensional cubic spline function is applied to fit a smooth surface to the set of input data values. The computation requires a number of iterations to adjust the surface so that the final result has a minimum amount of curvature.

4. The Algorithm

The prediction algorithm is a technique, which is used to make the best global prediction based on a subset of data stations, while taking into consideration the historical relationships, which exist between the various data stations in the region of interest.

The first step in the technique is to establish the historical relationships. This is achieved by having the data stations organized in a pair-wise fashion, where each station is paired with all other stations in the domain area. When the existing historical data from the paired stations are plotted on the scatter plot, there are four possible simple scenarios:

1. One station's values are high while the other station's values are low.
2. One station's values are low while the other station's values are high.
3. The values between the stations alternates from being higher and lower.
4. Both stations have the same values.

In each case a linear or quasi-linear relationship can be modeled between the pairs. The predictions for a station could then be obtained using those linear equations for stations where data are available.

As an example, if there are n stations in the domain area, then there will be a total of n(n-1) paired combinations. In other words, given a set of stations A, B and C, then the pairs would be AB, AC, BA, BC, CA and CB. All of these paired names are then stored in the primary key field of individual records in a lookup file. The records in the lookup file also store the correlation between the two stations, the distance between the two stations, the equation which defines the relationship between the two stations as well as some other data which will be discussed in the subsequent sections.

The algorithm implements a lookup file to store the information dynamically derived, which define the relationships existing between the stations. The lookup file is employed as follows:

Given the value of a station, say station A, the algorithm searches the lookup file for all the records that have its primary field value starting with the station name A (i.e. AB, AC, as in the example provided above). From the selected records, it is now possible to predict the values for the other stations in the pair, based on the value for the known station (station A in this case), since there would exist in the selected records an equation which defines their statistical relationship with the known station (station A) respectively. When another station's value becomes available (e.g. station B), it then becomes possible to predict the value of the other stations, not only from the previously known stations (e.g. station A) but also from the new station's record (Station B) in the lookup file. In all likelihood, it is quite possible that the different predictions for a station, whose data value is not known, by stations with known values, would not be perfectly identical. In such a case, the precedence portion of the algorithm, which forms the secondary nucleus of this work, would then have to decide which value among the lot would be the best prediction.

The proposed algorithm has basically three processes, which work together in order to have a successful and acceptable prediction outcome. The processes are made up of:

- Input data acceptance routine
- Paired automation routine
- Precedence selection routine

A brief description of the processes, along with the various pseudocode, are presented in Sections 4.1 to 4.4.

4.1. Input Data Acceptance Routine

The historical data for the stations within the specified domain are passed to the prediction algorithm through this module. The prediction algorithm uses the historical data values of the stations during common historical instances throughout the region of interest. These historical data values represent "snapshots" of the domain during those specified instances. To get the data organized in the required manner, the data can either be entered manually into the system, one station at a time, or more appropriately, can be accepted through the use of an input file.

For the purposes of providing an implementation example, the format of Microsoft Excel file is being used for the input file. In this file, the first column contains the names of the stations; the next series of columns hold the historical values for the stations (each column holds one value for the station for each time instance), and the last two columns to the extreme right of the data values holds the X and Y coordinates respectively of the station, within the domain.

The pseudocode for the data acceptance routine, when using an input file, is as follows:

```
create "points" file with fields (stations x, y)
open input file (possibly in Microsoft Excel file format)
```

```
for each station in the input file do
insert station names and coordinates into "points" file
create file of current station name with field (value)
insert station's data values into newly created file of the station name
endfor
```

The preceding pseudocode essentially creates a file name "points" which stores the names and coordinates of all the stations. It also creates a file named after each station where the historical values of each station are stored in their respective file. After the data have been organized in this way then the next step is to develop the paired automation stage.

4.2. Paired Automation Routine

After the historical data values for all the stations in the region of interest have been entered into the system, the algorithm proceeds to organize the data stations in a pair wise fashion. This type of organization is crucial to facilitate the prediction processes, because it is from these paired relationships that the unknown stations values can then be predicted. These predictions are made possible because each station is paired with all other stations in the region of interest. On pairing the stations, a predictive equation is developed which allows one station's value, in the pair of stations, to be used to predict the value of the other station in the pair. Therefore, it becomes possible to make a prediction for all the stations when the value of only one station is known. Needless to say, predictions made of all the stations when only one station is known would provide little more than a scaled representation of the mean spatial statistical distribution over the region of interest and would therefore be considerable less accurate than when many stations are known and used in the prediction process. As the stations are paired together, various statistical calculations are preformed with the goal of extracting and saving the statistical data in a master lookup file, which can then be referenced to guide the decision process.

The development and establishment of the master lookup file is a significant underpin to the proposed algorithm. The data stored in this file allow the algorithm to calculate various predicted values for the unknown stations based on the stations that have known values and also facilitate the determination of which predicted value is the best one based on the selected criteria in the precedence portion of the algorithm. In updating the master lookup file, several fields must be addressed. These fields hold the following information:

- Paired stations identification routine
- Relationship formula between the paired stations
- Correlation between the paired stations
- Distance between the paired stations
- Statistical Deviation between the paired stations

4.3. Paired Stations Identification Routine

In this component of the algorithm, the name of each of the station is combined with all the other station names. For convenience, the compound name is created by placing a "%" character between the two names. These concatenated names are then inserted into the master lookup table. The following pseudocode illustrates the process:

```
create array1
create array2
store all the station names in array1
store all the station names in array2
for each name in array1 do
for each name in array2 do
concatenate array1 station name to array2 station name
insert concatenated names into master lookup file
endfor
endfor
```

4.4. Relationship Formula Between the Paired Stations

This component of the algorithm determines the linear relationship formula between each pair of stations. The formula takes the form of the linear equation:

$$y = a \pm bx$$

where a and b are known and x and y are unknown. Therefore, the objective of this routine is to determine the values of a and b respectively. This is achieved by using the linear regression model on the historical data. (Kleinbaun et al [12] provide a detailed description on the technique). The routine queries the master lookup file, and uses the concatenated stations' names to access the necessary files in order to get the historical data to be used in the regression model. The following pseudocode illustrates how it is done:

```
open master lookup file
for each record in the master lookup file do
get names of each station in the paired name
assign first part of paired name to station1
assign second part of paired name to station2
open station1 file
open station2 file
store sum of station1 values to variable sx
store sum of station2 values to variable sy
store sum of (station1 values * station2 values) to variable sxy
store sum of (station1 values * station1 values) to variable sxsq
store record count of either station1 or station2 to variable rec
store (sy * sxsq) – (sx * sxy) to variable numeratorA
store (rec * sxsq) – (sx * sx) to variable denominatorA
if denominatorA is not zero then
store (numeratorA / denominatorA) to variable A
endif
store (rec * sxy) – (sx * sy) to variable numeratorB
store (rec * sxsq) – (sx * sx) to variable denominatorB
```

if denominatorB is not zero then
store (numeratorB / denominatorB) to variable B
endif
insert variable A and B's values in master lookup file for pair
endfor

4.5. Correlation between the Paired Stations

This routine calculates the strength of the linear relationship of the historical data for each pair of stations. This is achieved by using the established measure of correlation known as Pearson Product Moment Correlation. (For more details on the technique, refer to [1, 12, 13, 14, 15]. The routine queries the master lookup file, and uses the concatenated stations' names to access the necessary files in order to get the historical data to use in the Pearson's Correlation formula. The following pseudocode demonstrates how the correlation is obtained:

open master lookup file
for each record in the master lookup file do
get names of each station in the paired name
assign first part of paired name to station1
assign second part of paired name to station2
open station1 file
open station2 file
store sum of station1 values to variable sx
store sum of station2 values to variable sy
store sum of (station1 values * station2 values) to variable sxy
store sum of (station1 values * station1 values) to variable sxsq
store sum of (station2 values * station2 values) to variable sysq
store record count of either station1 or station2 to variable rec
store (rec * sxy) – (sx * sy) to variable corT
store the square root of (rec * sxsq)-(sx * sx) to variable corB1
store the square root of (rec * sysq)-(sy * sy) to variable corB2
store (corB1*corB2) to variable corB
if corB is not zero then
store corT/corB to variable correlation
endif
insert variable correlation's value in master lookup file
go to next record in the master lookup file repeat above steps
endfor

4.6. Distance between the Paired Stations

In this section of the algorithm, the Euclidean distance between the paired stations in the master lookup file is calculated. This distance is required by some of the precedence algorithms which will be presented subsequently. As with the other routines, this routine also queries the master lookup file to get the names of the paired stations. It then searches for the names in the points file created by the data acceptance routine. When the stations are found in the points file, the X-Y coordinates of each station are used to calculate the relative distance between them. The following pseudocode demonstrates the procedure:

open master lookup file
for each record in the master lookup file do
get names of each station in the paired name
assign first part of name to station1
assign second part of name to station2
open points file
search for station1
store coordinate in variables X1 and Y1
search for station2
store coordinates in variables X2 and Y2
store square of X1-X2 to variable num1
store square of Y1-Y2 to variable num2
store square-root of num1+num2 to variable distance
insert variable distance value in master lookup file
endfor

4.7. Deviation Value between the Paired Stations

In this routine the deviation value among the paired stations is calculated. We define the deviation value as the absolute difference between the values predicted for a station by another station that it is paired with and the actual value observed at that particular station. This deviation value is required by one of the precedence routines which will be described later in Section 5.1. The lookup file is searched and as each record is encountered, the file associated with the paired stations is opened and the values of the first station are plugged into the associated relationship formula for the calculation of the second stations' value. The absolute difference between calculated values and the actual observed values for the second station are tallied and inserted into the lookup file for that pair. This approach is chosen to allow faster execution of the precedence portion of the algorithm, in that, the deviation calculation would not have to be computed every time the deviation value is required. The steps are illustrated in the following pseudocode:

open master lookup file
for each record in the master lookup file
get names of stations making up pair
get relationship formula
open file of the paired stations
for each record in the paired stations file do
place the first station value in relationship formula
store absolute difference between output from formula and actual
value of second station into variable deviation
insert deviation value into lookup file for the current pair
endfor
endfor

5. Precedence Selection Routines

Having each station paired with all other stations, it therefore means that each station will have a predicted value based on the station it is paired with. In other words, if there are ***n*** stations in the region of interest then there can potentially be as many as ***n-1*** different predicted values for a station. Therefore, the precedence selection is the aspect of the algorithm, which allows for the setting of criteria by which one predicted value is chosen as the best value for the unknown station. For this algorithm, four criteria settings have been developed to drive the precedence selection process. These criteria settings are:

- Least Deviating Function
- Shortest Distance
- Moving Average
- Greatest Correlation

The Greatest Correlation (GC) criterion is being set as the default precedence setting for the algorithm.

5.1. Least Deviating Function

The Least Deviating Function method searches both the list of known stations and the list of stations to be predicted. As it searches, it concatenates the known stations name with the names of the stations to be predicted and these concatenated names are searched for in the lookup file. When the record is found, the deviation value is temporarily stored. The process of searching and temporarily storing the deviation value is continued until all the known stations are compared to each other and the lowest deviation value recorded. The record which has the lowest deviation values is selected as the record to make the prediction for the unknown station. The pseudocode below demonstrates the steps involved:

```
create variable mindev
initialize mindev to 0
open list of known stations
for each record in the list of known stations do
 store name of station to variable station1
open list of unknown stations
for each record in the list of unknown stations do
store name of station to variable station2
concatenate names of station1 and station2 to variable station1-2
open master lookup file
search for station1-2
if found then
 store deviation value to variable statdev
if mindev is equal to 0
store statdev to mindev
else
if statdev is less than mindev
store statdev to mindev
store equation to variable leastdev
endif
endif
endif
endfor
update unknown stations with prediction from leastdev record
endfor
```

5.2. Shortest Distance

The Shortest Distance method is similar to the Least Deviating Function. However, instead of searching the lookup file for the record with the least deviation value, it searches for the pair of stations that has the shortest relative distance between them. The pseudocode to this process is given below:

```
create variable leastdis
open list of known stations
for each record in the list of known stations do
 store name of station to variable station1
open list of unknown stations
for each record in the list of unknown stations do
store name of station to variable station2
concatenate names of station1 and station2 to variable station1-2
open master lookup file
search for station1-2
if found then
store distance value to variable dis
if this is the first search of the lookup file
store dis to leastdis
else
if dis is greater than leastdis
store dis to leastdis
store equation to variable less
endif
endif
endif
endfor
update unknown stations with prediction from less record
endfor
```

5.3. Moving Average

With the Moving Average method, the value of an unknown station is determined by taking the mean average of the predicted values for a number of known stations that are closest to the unknown station. In order to be meaningful, the number of known stations must be at the very least equal to two. The actual number of stations used in the routine can be chosen arbitrarily. If the total number of stations is small (less than 50 for example) then it would be expected that a small number of known stations would be selected to be used in the routine (say 3 to 8 for example). However, if the total number of stations is large (say, greater than 150) then a larger number of averaging stations could be selected.

The Moving Average routine systematically moves to all the unknown stations in the region of interest. As it moves to the unknown stations, the closest neighboring stations

with known data values within the specified number, are selected and the mean average of their predictions for that unknown station is calculated. The pseudocode for this routine is as follows:

```
initialize variable sum to 0
initialize variable average to 0
get the number of stations to used in the average from user
count number of known stations to variable kcount
open the list of unknown station
for each record in the list of unknown stations do
for item going from 1 to kcount do
concatenate known station name with station to be predicted
open lookup file
locate concatenated name
insert information into temporary file
endfor
open temporary file
sort file by distance in ascending order
for record from 1 to number of stations used in average do
concatenate known station name with station to be predicted
open lookup file
locate concatenated name
calculate predicted value
add predicted value to sum
store sum divided by number of stations used in prediction into
average
endfor
insert average into prediction value for current unknown stations
endfor
```

5.4. Greatest Correlation

The Greatest Correlation routine is similar to the Least Deviating Function presented previously in Section 5.1. However, instead of searching the lookup file for the record with the least deviation value, it searches for the pair of stations that has the highest correlation value between them. The pseudocode to the process is as follows:

```
create variable maxcorr
open list of known stations
for each record in the list of known stations do
store name of station to variable station1
open list of unknown stations
for each record in the list of unknown stations do
store name of station to variable station2
concatenate names of station1 and station2 to variable station1-2
open master lookup file
search for station1-2
if found then
store correlation value to variable corr
if this is the first search of the lookup file
store corr to maxcorr
else
if corr is greater than maxcorr
store corr to maxcorr
store equation to variable maxequ
endif
endif
endif
endfor
update unknown stations with prediction from maxequ record
endfor
```

6. Conclusion

The primary objective of this work was to develop an approach to the prediction of unknown stations values when some randomly selected stations values are known. This was achieved by the implementation of an algorithm that established a historical statistical relationship between pairs of all the data stations in the region of interest. The use of a historical relationship in the prediction of unknown data stations is a "step-away" from the established interpolations methods such as Kriging, Inverse Distance and Minimum Curvature (to name a few). These established methods do not factor in any historical relationships in their formulae.

A secondary objective was to compare the results of some of the more popular interpolation methods with the proposed algorithm. This comparison is implemented through the detailed examinations of two cases studied in Part II of this paper. In these studies, it will be shown that the image distribution maps for both an objective and subjective dataset, the proposed methodology generally out-performs the other popular interpolation analyses to generate predictions for the entire domain.

References

[1] N. I. Fisher, T. Lewis, B. J. J. Embleton, Statistical Analysis of Spherical Data, Cambridge University Press, 1987.

[2] D. Dorsel, T. La Breche, Kriging. <http://ewr.cee.vt.edu/environmental/teach/smprimer/kriging/kriging.html>, January 2009.

[3] R. V. Jesus, Kriging: An Accompanied Example in IDRISI, GIS Centrum, University of Lund for Oresund, Summer University, 2003.

[4] M.L. Stein, Interpolation of Spatial Data: Some Theory for Kriging, Springer, New York, 1999.

[5] W.C.M. Van Beers, J. P.C. Kleijnen, Kriging Interpolation in Simulation : A Survey, in: R .G. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters (Eds.), Proceedings of the 2004 Winter Simulation Conference,Washington, DC, 2004, pp. 113-121.

[6] D. T. Lee, B.J. Schachter, Two Algorithms for Constructing a Delaunay Triangulation. International Journal of

Computer and information Sciences, Vol. 9, 1980, pp. 219-242.

[7] P-J. Laurent, Wavelets, Images, and Surface Fitting, in: A. Le Mehaute (Ed.), A.K Peters Ltd., 1994.

[8] W. H. F. Smith, P. Wessel, Gridding with Continuous Curvature Splines in Tension, Geophysics, 55, 1990.

[9] D. Shepard, A two-dimensional interpolation function for irregularly spaced data. Proceedings of the 23rd ACM National Conference (128), 1968, pp.517-524.

[10] R. Sierra, Rigid Registration. <http://www.rsierra.com/DA/node10.html#SECTION00103 0000000000000000>, May 2009.

[11] J. Parag, Class Presentation. <http://arcib.dowling.edu/~JainP/Research1/slide2.html>, May 2009.

[12] D. Kleinbaum, L. Kupper, K. Muller, Applied Regression Analysis and other Multivariable Method, Duxbury Press, 1987.

[13] Wasson J. Statistics in Educational Research - An Internet Based Course. <http://www.mnstate.edu/wasson/ed602pearsoncorr.htm>, April 2009.

[14] J. Deacon, Correlation, and regression analysis for curve fitting. <http://www.biology.ed.ac.uk/research/groups/jdeacon/statistics/tress11.html >, January 2009.

[15] R.J. Rummel, Understanding Correlation. <http://www.mega.nu:8080/ampp/rummel/uc.htm>, December 2009.

[16] E. Yudkowsky, An Intuitive Explanation of Bayesian Reasoning, <http://yudkowsky.net/bayes/bayes.html>, May 2009.

[17] P. E. Gill, W. Murray, Algorithms for the solution of the nonlinear least-squares problem. SIAM Journal of Numeral Analysis, 15 [5], 1978, pp. 977-992.

[18] W. R. Greco, M. T. Hakala. Evaluation of methods for estimating the dissociation constant of tight binding enzyme inhibitors, Journal of Biological Chemistry, (254), 1979, pp.12104-12109.

[19] D. F. Symancyk, Visualizing Gaussian Elimination, <http://ola4.aacc.edu/dfsymancyk/vgetalk/VGEtalkexpanded.html>, May 2006.