Inverse Visualization In Data Mining

Evgenii Vityaev
Institute of Mathematics, Russian Academy of Science, Novosibirsk 630090, Russia Vityaev@math.nsc.ru
Boris Kovalerchuk
Department of Computer Science, Central Washington University, Ellensburg, WA, 98926-7520, USA
Borisk@cwu.edu

Abstract: Visualization is used in data mining for visual presentation of already discovered patterns and for discovering new patterns visually. Success in both tasks depends on the ability of presenting abstract patterns as simple visual patterns. Getting simple visualizations for complex abstract patterns is an especially challenging problem. A new approach called inverse visualization (IV) is suggested to address the problem of visualizing complex patterns. The approach is based on specially designed data preprocessing. Preprocessed data permit the discovery of abstract patterns that can be presented by simple visual patterns. Design of data preprocessing transformations is based on a transformation theorem proved in the paper. A mathematical formalism is derived from the Representative Measurement Theory [Suppes at al, 1990] . The possibility of solving inverse visualization tasks is illustrated on functional non-linear additive dependencies $\varphi(f(x,z))$. These dependencies are transformed into simple and intuitive visual patterns. The approach is called inverse visualization because it does not use data "as is" and does not follow a traditional sequence: discover pattern → visualize pattern. The new sequence is: convert data to visualizable form → discover patterns with predefined visualization.

Introduction

Visual data mining is a growing area of research and applications [Keim, 2001, Fayyad et al, 2001, Spence, 2001, Ware, 2000]. In this paper the visualization task for data mining (DM) is considered as presenting discovered patterns in a specific visual form. This visual form should be perceivable, understandable and interpretable in frames of a domain theory associated with data used for data mining.

We consider the **Inverse Visualization Task** (IVT) -- to find data transformations that permit to get simple and clear visualization of data and patterns. Success in this endeavor depends on properties of data transformations and data mining methods involved. Many DM practitioners share an opinion, that practically any DM method will discover meaningful patterns for "good" data but none of them or just few can produce meaningful patterns for "poor" data. One of the goals of IVT is to transform "poor" data into "good" data permitting a wide variety of DM methods to be used successfully to discover hidden patterns.

At this moment we are not attempting to define formally "poor" and "good" data, but are showing that in classical physics such transformations were successful for a long time in discovering patterns called now classical (fundamental) physical laws without formal definitions of "good" data. It is known, that the laws of classic physics are simple and thus the problem of their visualization is not very difficult.

Lessons learned from classical physics can help in DM in other domains where patterns do not appear to be simple. At first we need to understand what is the reason of simplicity of laws in classical physics. Are these reasons specific for physics or they can be expanded for domains such as finance, medicine, remote sensing, image analysis and so on?

The explanation in physics follows from two theories: the Representative Measurement Theory (RMT) [Krantz et al, 1990] and the Physical Structures Theory [Kulakov, 1971; Mikhailichenko, 1985]. In the measurement theory [Krantz et al, 1990, v.1] it is demonstrated, that a system of physical quantities and fundamental laws, are simple since they are obtained by a procedure of **simultaneous scaling** of variables involved in the laws. **Traditionally DM does not excise simultaneous scaling**. Note simultaneous scaling is different from data **normalization** procedures used in Neural Networks (NN) to speed up search [Rao, Rao, 1995] The typical normalization in NN transforms scales of each variable **independently** and non-linearly to another interval, such as [-1,1]. For instance, simultaneous scaling of variables x, y, z, can be able to transform these variables into new scales x',y',z' such, that the law has the simple linear form y'= x'+z'. In general laws of classical physics show that if all variables included in a law are scaled simultaneously then the law can have rather simple form.

The problem of **finding efficient simultaneous scaling** transformations was not posed and solved in the Representative Measurement Theory. This theory explains the simplicity effect but lacks a constructive way to reach it. On the other hand the representative measurement theory has much wider area of application than only physics. For instance, psychology has benefited significantly from RMT [1]. This observation raises a hope that simultaneous scaling will be beneficial in other areas too. This requires designing simultaneous scaling transformations.

Fortunately the theory of Physical Structures gives the answer for this problem for fundamental physical laws via constructive classification of all functional expressions of all possible fundamental physical laws [3]. All other functional expressions of physical laws, can be transformed to one of the classification types by a monotone transformation of all involved variables x, y, z... (using the procedure of simultaneous scaling of these variables). This result shows, that all physical laws can be described with exactness up to monotone transformations of variables, included in a law. This means that all laws can be enumerated in the classification from [3] with exactness up to an arbitrary monotone transformation All laws of this classification are simple and the problem of their visualization is simple too. All **complexity of visualization** of these laws is converted into design of **monotone transformations** of n-tuples of variables involved.

In this paper we workout this idea and transform data to get an elementary law of type y=x+z. Figure 2 shows original data and figure 3 shows transformed data visualized. The rest of this paper is structured as follows: definitions, the theorem, an example and discovering simultaneous scaling for the example. The example elaborates results presented in Figures 2 and 3.

Definitions

Let us define a class of functions F, which can be transformed to a linear function y = x+z by monotone transformations. There are many possible ways to define class F. It is convenient to assume that F is given through a system of axioms. Suppose that a data set D from a specific applied domain (e.g., finance) represent a set of triples x,y, z, where y = f(x,z) and function f is not known analytically. Function f is known only through tabulated values in D and possibly some meaningful (for the domain) properties in the form of axioms. We assume: (1) variables x,y,z are mapped to real numbers Re, (2) the order relation on Re is interpreted for variable y, and (3) equality relations on Re are interpreted for variables x and z.

We define class F of functions $f \in F$ on $X_f x Z_f$, $X_f \subset Re$, $X_f \neq \varnothing$, $Z_f \subset Re$, $Z_f \neq \varnothing$. Functions from F satisfy properties 1-5 of additive conjoint structure [1, v.1, p.256]:

1*. $\forall z1,z2, \exists x\ (f(x, z1) \geq f(x, z2) \Rightarrow \forall x'(f(x', z1) \geq f(x', z2)))$

2. $\forall x1,x2,x3,z1,z2,z3$

$$((f(x1,z2) \approx f(x2,z1))\&(f(x1,z3) \approx f(x3,z1)) \Rightarrow$$

$$(f(x2, z3) \approx f(x3,z2))$$

3. For any three of x1,x2,z1,z2 the fourth of them exists such that

$$f(x1,z2) = f(x2,z1)$$

4*. $\exists\ x1,x2, z\ (f(x1, z) \neq f(x2, z))$

5*. For any z1, z2 : $z1 \neq z2$, if a sequence x1,x2,…xi ... of elements of X_f is determined and satisfies the following properties:

$$\forall i\ xi < xmax$$

$$f(x1, z1) = f(x2, z2),$$

$$f(x2, z1) = f(x3, z2),$$

$$f(x3, z1) = f(x4, z2),$$

…………………..

$$f(xi, z1) = f(x(i+1), z2),$$

…………………..

then this sequence is finite.

Similar properties should be fulfilled for z by replacement of x by z and vice versa excluding properties marked with asterisks (*).

The universal and existence quantifiers apply to sets X_f, Z_f

Theorem

The theorem below is based on axioms 1-5 and is used for design of simultaneous scaling.

Theorem:

1. For any function $f \in F$ there are one-to-one functions φx, φz and monotone function φ such that

$$\varphi f(x, z) = \varphi x(x) + \varphi z(z),\ \langle x,z\rangle \in X_f x Z_f.$$

2. If $f \in F$, then the function

$$f'(x', z') = \varphi f(\varphi x x', \varphi z z'),$$

where φ is a strictly monotone function, and φx, φz are one-to-one functions from F.

The proof follows from the fact, that the system of axioms represents an additive conjoint structure [1, v.1]. Let function $f \in F$ on $X_f x Z_f$, satisfies the axioms 1-5. By virtue of the axiom 4 there are points on the plane <x0,z0>, <x1,z0> such that

$$f(x0, z0) \neq f(x1, z0)$$

(see Figure 1).

Let's simultaneously scale X, Z, and Y (y=f(x, z)) as follows:

- assign value 0 to x0 of the scale X; record it as x0 = 0;
- assign value 1 to x1;
- assign value 0 to z0 of scale Z;
- assign values f(x0,z0) = 0 and f(x1,z0) = 1 for function f (Y-axis).

By virtue of the axiom 3 for three elements x0,z0,x1 there exists fourth z1, such that

$$f(x0,z1) = f(x1,z0).$$

Let's link the points <x0,z1>, <x1,z0> as shown in Figure 1. Along this line the function has identical values. These values are the values of Y scale (which is not shown on the picture). It is easy to see, that these values of x, z, y satisfy the function x + z = y. We take a point <x1,z1> and assign value y = f(x1,z1) = 2 for this point. Next we again apply the axiom 3. At first we apply it to values x1,x0,z1 and receive x2 such that f(x1,z1) = f(x2,z0) and then we apply it to values x1,x0,z1 and receive z2, such that f(x0,z2) = f(x1,z1). After that we assign value y = f(x0,z2) = f(x1,z1) = f(x2,z0) = 2. Now we consider new points <x2,z1> and <x1,z2>. To make the given construction possible for all new points <x0,z3>, <x3,z0> it is necessary, that the values of the function would be identical f(x2,z1) = f(x1,z2) for points

<x2,z1> and <x1,z2>. The equality f(x2,z1) = f(x1,z2) follows from the axiom 2.

Figures 2 and 3 present such transformation. A surface on Figure 2 is transformed to a surface on Figure 3 by simultaneous rescaling of variables x, z, and y. It follows from the theorem, that if properties 1-5 take place for some variables x, y, z, then the function $f \in$ F can be converted to function y = x+z by rescaling variables. The visualization of rescaled data and function y = x+z is obvious (see Figure 3). The theorem has been proved by developing a constructive procedure, thus a rescaling algorithm can be extracted from the proof of the theorem and systems of axioms 1-5. Is it is shown in the proof that the procedure requires values of a function f on specific pairs of values <x,z> that are generated in the course of scaling. This condition is true for functions that satisfy the condition of the theorem, such as scaling of preference relations used in Decision Theory [4], but this is not a universally true condition for other tasks.

Test Example

Test example must satisfy several requirements to be really convincing:
(1) It should contain regularities (patterns) known in advance;
(2) These regularities should have at least a hypothetical meaningful interpretation in the domain;
(3) These regularities should not be obvious when data prescreened and visualized before rescaling (see Figure 2).

Table 1 contains data to meet these requirements. It is produced in the way described below:

- Attributes 1-3 and 5-9 are created by using a random number generator. Attributes 1-3 could model some basic independent indicators of a product manufacturing.
- Attribute 4 is a sum of the first two attributes.
- Attribute 10 is a target attribute, it is equal to some random monotone transformation F of attribute 4 minus attribute 3, that is Attr#10 = F(4-3) = F(1+2-3).
- Attributes 5-9 represent noise. They are random and unrelated to the target attribute #10. A hypothetical interpretation of regularity F(1+2-3) could be, for instance, productivity or production efficiency, revenue and so on.
- Possible interpretation of other attributes. Attribute #1 indicates initial capital (scaled from 0 to 10), Attribute #2 indicates quality of management (scaled also from 0-10) and attribute #3 indicates tax level (scaled from 0 to 10). Attributes #1 and #2 contribute positively to revenue and attribute #3 negatively.

A relatively complex monotone transformation is motivated by an intention:

- to solve a **realistic task**. In real-world tasks if there are any hidden pattern (regularity) they are usually disguised and significantly corrupted. Next experience shows that monotone regularities are common for many data mining tasks.
- to show **unique capabilities** of the simultaneous scaling method. Traditional methods that do not use simultaneous scaling can not discover a regularity corrupted by a random monotone transformation. The only way to do this to analyze all interpretable order relations for all attributes: $\le$1, $\le$2, $\le$3,..., $\le$10. These regularities are revealed by the simultaneous scaling method.
- to find **meaningful patterns**, regularities, functions. For instance, typically regression analysis produces functions that just interpolate data without a meaningful interpretation in the domain. In contrast the simultaneous scaling method produces meaningful regularities such as

$$\forall a,b\ ((a \le_1 b) \& (a \le_2 b) \Rightarrow a \le_{10} b)$$

The data (Table 1) encodes the following regularities by design:

$$\begin{aligned}
&\forall a,b(a >_3 b \ \&\ a \le_4 b \Rightarrow a \le_{10} b)\\
&\forall a,b(a \le_3 b \ \&\ a >_4 b \Rightarrow a >_{10} b)\\
&\forall a,b(a \le_1 b \ \&\ a \le_2 b \ \&\ a >_3 b \Rightarrow a \le_{10} b) \qquad (1)\\
&\forall a,b(a >_1 b \ \&\ a >_2 b \ \&\ a \le_3 b \Rightarrow a >_{10} b)
\end{aligned}$$

Discovering simultaneous scaling

The Discovery system [Kovalerchuk, Vityaev, 2000] can discover all monotone regularities including (1) actually encoded in Table 1 along with random noise. When (1) is discovered a simultaneous monotone rescalings of data can be arranged straightforward and simple visualization presented in Figure 3 will be generated.

Thus the major challenge is discovering monotone regularities. The Discovery system searches sequentially monotone regularities starting from simplest ones:

$$\forall a,b(a \le_i b \Rightarrow a >_{10} b),\ i = 1,...,9;$$

After testing them we discover a regularity

$$\forall a,b(a \le_4 b \Rightarrow a \le_{10} b).$$

with a statistical confidence level equal to 0.0001. This regularity is not in the list (1), although it is true for data from Table 1. Next the systems tests more complex regularities:

$$\forall a,b(a >_i b \ \&\ a \le_j b \Rightarrow a \le_{10} b),\ i,j = 1,...,9$$

and finds a regularity

$$\forall a,b(a >_3 b \ \&\ a \le_4 b \Rightarrow a \le_{10} b)$$

with a statistical confidence level equal to 0.025.

Similarly, another parametric set of hypothetical regularities is generated and tested to discover the second regularity in (1). Similarly to discover a regularity with all three variables in the if-part we substitute given attributes with parameters. For instance, for discovering

$$\forall a,b(a \le_1 b \ \&\ a \le_2 b \ \&\ a >_3 b \Rightarrow a \le_{10} b),$$

we generate a parametric set

$$\forall a,b(a \le_i b \ \&\ a \le_j b \ \&\ a >_k b \Rightarrow a \le_{10} b),\ i,j,k = 1,...,9$$

and test it. The test reveals a needed regularity with the level of confidence equal to 0.1 .

References

1. Krantz DH, Luce RD, Suppes P, Tversky A: Foundations of Measurement v.1-3, Acad. Press, NY, London. 1971, 1989, 1990.
2. Kulakov Yu.I. The One Principal Underlying Classical Physics, Soviet Physics – Doclagy, V.15, #7, Jan., 1971, 666-668.
3. Mikhailichenko G.G. Phenomenological and Group Symmetry in the Geometry of two Sets (Theory of Physical Structures), Soviet Math. Docl. 32(2), 1985, 371-374.
4. Keeney R.L., Raiffa H. Decisions with Multiple Objectives: preferences and value Tradeoffs. John Wiley& Sons, 1976
5. Keim D., Visual Exploration of Large Data Sets, Communication of ACM, vol. 44, N.8, 2001, pp. 39-44.
6. Kovalerchuk, B., Vityaev, E. Data Mining in Finance: Advances in Relational and Hybrid Methods, Kluwer Acad. Publ., Boston, 2000.
7. Rao H., Rao V., C++ Neural Networks and Fuzzy Logic, Hungry Minds, Inc, 1995
8. Spence R., Information Visualization, Addison-Wesley, 2001.
9. Information Visualization in Data Mining and Knowledge Discovery, Usama Fayyad, Georges Grinstein and Andreas Wierse, 2001, 300 p.
10. Ware C. Information Visualization: perception for Design, Acad. Press, 2000.
11. Information Visualization in Data Mining and Knowledge Discovery, Eds. U. Fayyad, G. Grinstein and A. Wierse, 2001, Morgan-Kaufman.

Table 1. Test data

1 1	**1 2**	**1 3**	**1 4**	**1 5**	**1 6**	**1 7**	**1 8**	**1 9**	**1 10**	**1**
1 08	1 02	1 08	1 10	1 05	1 09	1 00	1 04	1 01	1 53	1
1 06	1 04	1 00	1 10	1 01	1 01	1 08	1 01	1 05	1 79	1
1 04	1 01	1 03	1 05	1 00	1 03	1 08	1 07	1 01	1 53	1
1 08	1 00	1 09	1 08	1 09	1 01	1 05	1 00	1 00	1 30	1
1 07	1 08	1 04	1 15	1 05	1 00	1 02	1 08	1 06	1 83	1
1 09	1 01	1 08	1 10	1 09	1 07	1 00	1 03	1 04	1 53	1
1 09	1 05	1 05	1 14	1 03	1 03	1 09	1 07	1 07	1 75	1
1 05	1 08	1 02	1 13	1 07	1 01	1 06	1 06	1 08	1 83	1
1 01	1 09	1 04	1 10	1 00	1 01	1 04	1 07	1 01	1 64	1
1 01	1 02	1 03	1 03	1 04	1 00	1 03	1 02	1 07	1 31	1
1 07	1 05	1 05	1 12	1 06	1 08	1 00	1 02	1 04	1 66	1
1 02	1 01	1 06	1 03	1 03	1 08	1 01	1 07	1 02	1 17	1
1 04	1 06	1 05	1 10	1 06	1 06	1 07	1 05	1 04	1 61	1
1 01	1 05	1 08	1 06	1 04	1 09	1 00	1 09	1 05	1 24	1
1 07	1 03	1 09	1 10	1 01	1 04	1 05	1 08	1 02	1 51	1
1 02	1 06	1 09	1 08	1 09	1 05	1 03	1 05	1 02	1 30	1
1 01	1 08	1 09	1 09	1 06	1 06	1 03	1 08	1 02	1 31	1
1 09	1 03	1 07	1 12	1 03	1 09	1 01	1 02	1 01	1 61	1
1 00	1 04	1 04	1 04	1 04	1 02	1 00	1 03	1 04	1 31	1
1 09	1 01	1 05	1 10	1 05	1 04	1 02	1 06	1 01	1 61	1
1 08	1 07	1 06	1 15	1 01	1 02	1 07	1 01	1 01	1 75	1
1 07	1 09	1 06	1 16	1 09	1 06	1 07	1 03	1 01	1 79	1
1 08	1 09	1 02	1 17	1 01	1 05	1 06	1 03	1 09	1 96	1
1 08	1 02	1 07	1 10	1 09	1 04	1 01	1 01	1 06	1 56	1
1 00	1 01	1 07	1 01	1 06	1 09	1 00	1 06	1 06	1 09	1
1 07	1 07	1 01	1 14	1 05	1 00	1 05	1 02	1 05	1 91	1
1 02	1 09	1 07	1 11	1 02	1 01	1 01	1 05	1 03	1 58	1
1 05	1 05	1 01	1 10	1 02	1 04	1 05	1 08	1 08	1 75	1
1 01	1 07	1 00	1 08	1 06	1 03	1 06	1 09	1 07	1 72	1
1 04	1 06	1 00	1 10	1 00	1 09	1 08	1 02	1 07	1 79	1
1 00	1 06	1 01	1 06	1 03	1 07	1 08	1 06	1 01	1 61	1
1 08	1 04	1 04	1 12	1 02	1 06	1 06	1 04	1 06	1 72	1
1 03	1 04	1 06	1 07	1 07	1 02	1 01	1 07	1 06	1 51	1
1 02	1 07	1 08	1 09	1 03	1 03	1 08	1 04	1 04	1 51	1
1 06	1 05	1 09	1 11	1 07	1 02	1 04	1 05	1 05	1 53	1
1 09	1 03	1 09	1 12	1 08	1 05	1 08	1 06	1 00	1 56	1
1 01	1 06	1 02	1 07	1 00	1 07	1 06	1 00	1 08	1 61	1
1 06	1 00	1 00	1 06	1 00	1 02	1 03	1 07	1 05	1 64	1
1 02	1 07	1 09	1 09	1 00	1 01	1 06	1 09	1 08	1 31	1
1 04	1 08	1 02	1 12	1 00	1 09	1 05	1 01	1 05	1 79	1
1 01	1 04	1 03	1 05	1 07	1 09	1 06	1 06	1 07	1 53	1
1 03	1 08	1 09	1 11	1 05	1 06	1 02	1 04	1 02	1 53	1
1 09	1 01	1 05	1 10	1 01	1 08	1 00	1 06	1 03	1 61	1
1 08	1 06	1 06	1 14	1 06	1 02	1 06	1 07	1 02	1 72	1
1 00	1 08	1 09	1 08	1 09	1 07	1 02	1 01	1 01	1 30	1
1 04	1 04	1 03	1 08	1 09	1 02	1 08	1 02	1 05	1 61	1
1 09	1 07	1 08	1 16	1 04	1 04	1 05	1 00	1 09	1 72	1
1 06	1 04	1 07	1 10	1 04	1 07	1 04	1 09	1 06	1 56	1
1 05	1 02	1 04	1 07	1 07	1 07	1 07	1 07	1 03	1 56	1
1 02	1 02	1 09	1 04	1 03	1 06	1 02	1 05	1 02	1 12	1
1 01	1 07	1 08	1 08	1 05	1 00	1 06	1 05	1 06	1 31	1
1 09	1 01	1 08	1 10	1 03	1 06	1 05	1 05	1 04	1 53	1
1 08	1 01	1 05	1 09	1 05	1 02	1 07	1 03	1 01	1 58	1
1 02	1 07	1 07	1 09	1 08	1 00	1 04	1 08	1 09	1 53	1
1 03	1 05	1 06	1 08	1 05	1 06	1 02	1 04	1 02	1 53	1
1 05	1 05	1 01	1 10	1 08	1 07	1 09	1 01	1 03	1 75	1
1 03	1 05	1 07	1 08	1 08	1 03	1 03	1 06	1 07	1 51	1
1 04	1 05	1 04	1 09	1 08	1 06	1 04	1 08	1 01	1 61	1
1 02	1 09	1 04	1 11	1 04	1 07	1 04	1 00	1 01	1 66	1
1 04	1 01	1 09	1 05	1 09	1 00	1 02	1 08	1 05	1 14	1
1 09	1 01	1 05	1 10	1 04	1 00	1 07	1 09	1 02	1 61	1
1 05	1 08	1 04	1 13	1 08	1 06	1 02	1 05	1 08	1 75	1
1 05	1 08	1 06	1 13	1 06	1 00	1 05	1 09	1 09	1 66	1
1 09	1 02	1 05	1 11	1 02	1 01	1 09	1 07	1 02	1 64	1
1 05	1 08	1 08	1 13	1 03	1 06	1 02	1 01	1 07	1 61	1
1 03	1 07	1 09	1 10	1 02	1 01	1 06	1 04	1 07	1 51	1
1 08	1 04	1 08	1 12	1 08	1 02	1 04	1 09	1 00	1 58	1
1 07	1 00	1 08	1 07	1 06	1 00	1 08	1 01	1 05	1 30	1
1 04	1 01	1 06	1 05	1 09	1 02	1 08	1 09	1 07	1 30	1
1 08	1 06	1 08	1 16	1 04	1 08	1 08	1 02	1 08	1 72	1
1 01	1 05	1 05	1 06	1 02	1 03	1 07	1 06	1 08	1 51	1
1 01	1 03	1 09	1 04	1 02	1 04	1 05	1 06	1 04	1 12	1
1 07	1 03	1 00	1 10	1 06	1 03	1 07	1 06	1 06	1 79	1
1 09	1 00	1 04	1 09	1 03	1 08	1 00	1 00	1 05	1 61	1
1 05	1 07	1 04	1 12	1 01	1 06	1 08	1 06	1 08	1 72	1
1 05	1 01	1 09	1 06	1 01	1 04	1 08	1 08	1 04	1 17	1
1 03	1 00	1 05	1 03	1 07	1 06	1 05	1 04	1 01	1 24	1
1 07	1 07	1 07	1 14	1 07	1 09	1 07	1 05	1 07	1 66	1
1 00	1 02	1 09	1 02	1 05	1 08	1 05	1 01	1 08	1 05	1
1 03	1 04	1 04	1 07	1 00	1 09	1 09	1 07	1 08	1 56	1

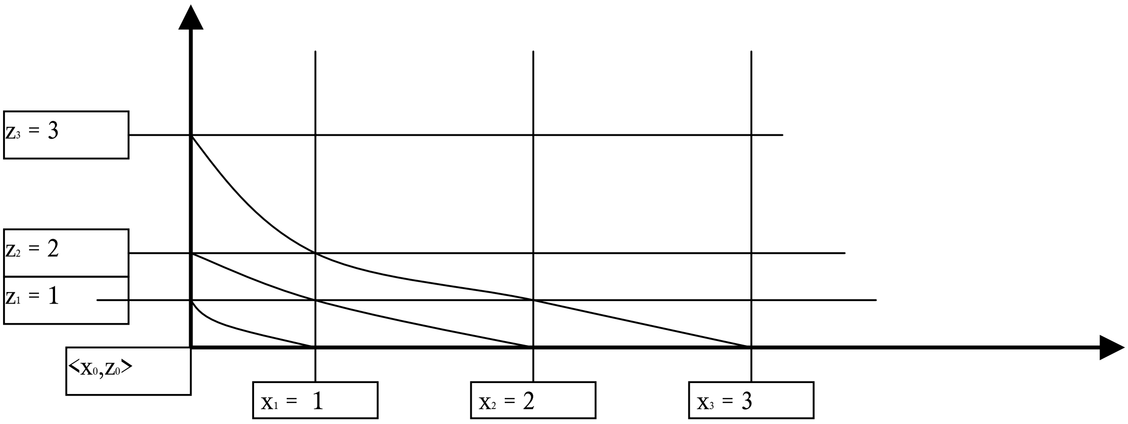

Figure 1

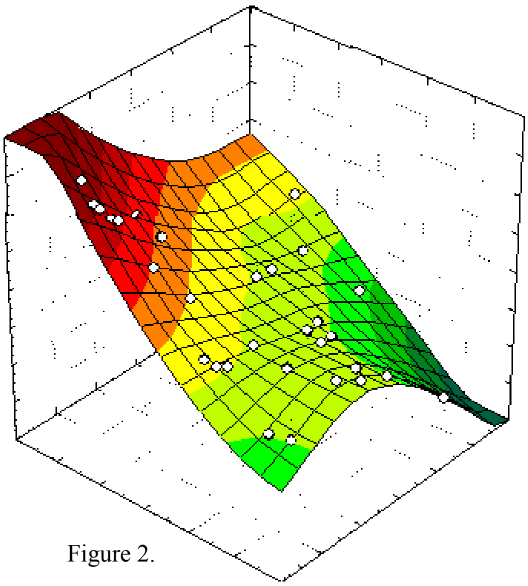

Figure 2.

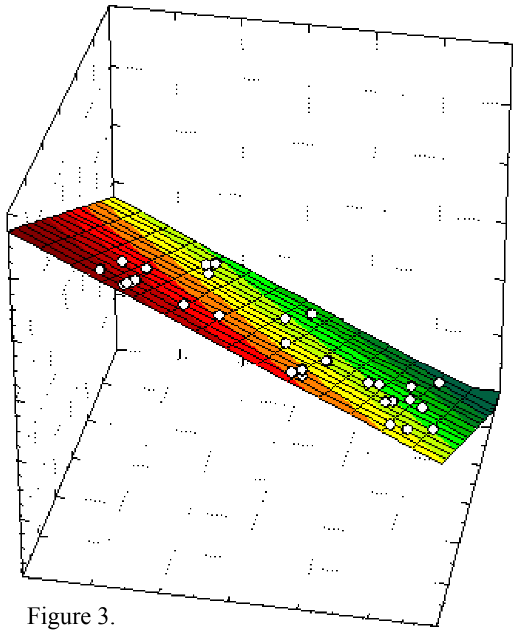

Figure 3.