REAL TIME PROPERTY INVESTIGATION IN SETS OF ALLOYS

Lorentz JÄNTSCHI, Technical University of Cluj-Napoca

Abstract

The present paper is focused on modeling of statistical data processing with applications in field of material science and engineering. A new method of data preprocessing is presented and applied on a set of 10 Ni–Mn–Ga ferromagnetic ordered shape memory alloys that are known to exhibit phonon softening and soft mode condensation into a premartensitic phase prior to the martensitic transformation itself.

The method allows to identify the correlations between data sets and to exploit them later in statistical study of alloys. An algorithm for computing data was implemented in preprocessed hypertext language (PHP) and a hypertext markup language interface for them was also realized and putted onto lejpt.utcluj.ro server at the address http://lejpt.utcluj.ro/~lori/research/alloys.

The program running for the set of alloys allow to identify groups of alloys properties and give qualitative measure of correlations between properties. Surfaces of property dependencies are also fitted.

Introduction

Many statistical procedures for processing data are available.[1] Most of them offer a voluble set of possibilities and variants, but which one to consider them? That is not a easy question and the frequent answer is: that is choice of analyst.[2,3] Data mining technology offer in this area of knowledge some answers, but not a complete answer.[4] By other hand, to interpret experiment results, data need to be well processed.[5] Structure investigations are frequently combined with statistical processing.[6] In most of cases, best results are obtained with specific procedures in contrast to general numeric algorithms.[7,8] Modeling of structure is benefit to property predictions.[9,10] Nonstandard statistical evaluation procedures then are helpful.[11]

The presented model make data preprocessing to a set of 10 Ni–Mn–Ga ferromagnetic ordered shape memory alloys that are known to exhibit phonon softening and soft mode condensation into a premartensitic phase prior to the martensitic transformation itself and is a extension added to the model presented in book [12].

Method

The logic scheme of data preprocessing is presented in figure 1.

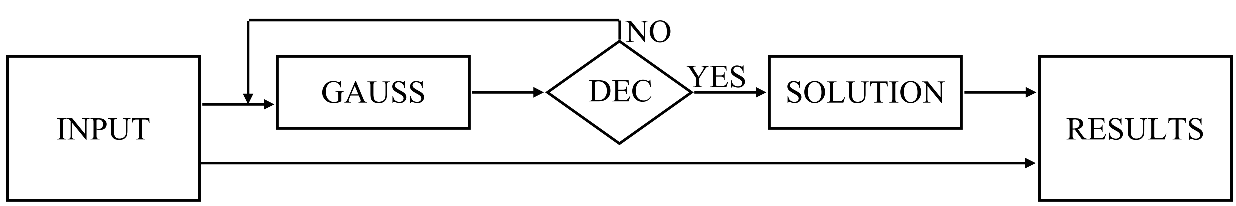

Figure 1. Data automat processing algorithm

The INPUT module read a text format data, process input data, split it into rows and columns and computes average means.

If name *n_rows* it assigned to number of rows, *n_cols* to number of columns, *data* to array of data, the output of module INPUT is computed by formulas:

$$M[i,j] = \frac{\sum_{k=1}^{n\_rows} data[k][i] \cdot data[k][j]}{n\_rows};\ M[0,j] = \frac{\sum_{k=1}^{n\_rows} data[k][i]}{n\_rows},\ 1 \le i, j \le n\_cols \quad (1)$$

Linear regression and PLS (partial least squares) are most used methods in statistical processing of data. Presented method uses them.

The output of INPUT module is used as input in GAUSS and RESULTS modules.

GAUSS module solves a linear system of equations in form:

$$\begin{cases} M[1,1]\cdot x_1 + \ldots + M[1,j]\cdot x_j + \ldots + M[1,n\_cols]\cdot x_{n\_cols} & = 1 \\ \ldots & \ldots \\ M[i,1]\cdot x_1 + \ldots + M[i,j]\cdot x_j + \ldots + M[i,n\_cols]\cdot x_{n\_cols} & = 1 \\ \ldots & \ldots \\ M[n\_cols,1]\cdot x_1 + \ldots + M[n\_cols,j]\cdot x_j + \ldots + M[n\_cols,n\_cols]\cdot x_{n\_cols} & = 1 \end{cases} \quad (2)$$

If answer of algorithm solving is *undetermined system* and null variable is x_{n_cols} then GAUSS module solve determined system of *n_cols* order given by equation (3):

$$\begin{cases} M[1,1]\cdot x_1 + \ldots + M[1,j]\cdot x_j + \ldots + M[1,n\_cols-1]\cdot x_{n\_cols-1} & = M[0,1] \\ \ldots & \ldots \\ M[i,1]\cdot x_1 + \ldots + M[i,j]\cdot x_j + \ldots + M[i,n\_cols-1]\cdot x_{n\_cols-1} & = M[0,j] \\ \ldots & \ldots \\ M[n\_cols-1,1]\cdot x_1 + M[n\_cols-1,j]\cdot x_j + \ldots + M[n\_cols-1,n\_cols-1]\cdot x_{n\_cols-1} & = M[0,n\_cols-1] \end{cases} \quad (3)$$

If answer of algorithm solving is *undetermined system* and null variable is different form x_{n_cols} then GAUSS module pass extended system matrix to DEC module.

If input in DEC module is *undetermined system* this it extract null row and column corresponding to the null variable (figure 2) and the resulting matrix of (n_cols-1)× n_cols dimension is passed again to GAUSS module.

$$\begin{array}{cc} & \begin{array}{ccccc} & \text{null column} & & & \\ & \downarrow & & & \end{array} \\ & \begin{bmatrix} \ldots & 0 & \ldots & M[1,n\_cols] & 1 \\ \ldots & 0 & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & 1 \\ \ldots & 0. & \ldots & \ldots & \ldots \\ \ldots & 0 & \ldots & M[n\_cols,n\_cols] & 1 \end{bmatrix} \\ \text{null row} \rightarrow & \end{array}$$

Figure 2. Processing data in DEC module

When system is solved a unique solution is found. Then, System extended matrix contain at column n_cols the coefficients of regression equation:

$$a_1 \cdot x_1 + \ldots + a_i \cdot x_i + \ldots + a_{n\_cols} \cdot x_{n\_cols} + a_{n\_cols+1} = 0 \quad (4)$$

where the coefficients a_{n_cols+1} and a_{n_cols+1} result different by case of reducing order of system (equation 2 or 3). Thus, if is applied equation 2, then:

$$a_{n\_cols+1} = 1 \quad (5)$$

else (is applied equation 3):

$$a_{n\_cols} = 1,\ a_{n\_cols+1} = 0 \quad (6)$$

At end of module SOLUTION it result an implicit linear regression equation between given variables through his values in columns (equation 4). Equation 4 can be exploited to obtain explicit linear regression equations for each variable that has no null coefficient a_i:

$$x_i = \left(\frac{a_1}{-a_i}\right)\cdot x_1 + \ldots + \left(\frac{a_{i-1}}{-a_i}\right)\cdot x_{i-1} + \left(\frac{a_{i+1}}{-a_i}\right)\cdot x_{i+1} + \ldots + \left(\frac{a_{n\_cols}}{-a_i}\right)\cdot x_{n\_cols} + \left(\frac{a_{n\_cols+1}}{-a_i}\right) \quad (7)$$

Sum of residues can be now evaluated:

$$S_i = \left(\frac{a_{n\_cols+1}}{a_i} + \sum_{j=1}^{n\_cols}\frac{a_j}{a_i}\cdot x_i\right)^2 \quad (8)$$

To compare one equation to another, a order value is required. Let to explicit this. If x_1 values (data[1] from input) are percents expressed in values from 0 to 100 and x_2 is premartensitic temperature transformation expressed in K with values from 100 to 600, then also sum of residues are expressed square of same measurement units. To make independence of measurement unit and measure order, values S_i are divided with own sum of squares of variable measurements (M[i,i] from INPUT module, equation 1). Final equation, with substitution x_i = data[k,i], $1 \le k \le$ n_rows and summing is:

$$Q_i = \sum_{k=1}^{n\_rows}\left(\frac{a_{n\_cols+1}}{a_i} + \sum_{j=1}^{n\_cols}\frac{a_j}{a_i}\cdot \mathrm{data}[k,i]\right)^2 / M[i,i] \quad (9)$$

and express relative residues of variable x_i when variable x_i is assumed to be dependent of independent variables x_1, …, x_{i-1}, x_{i+1}, x_{cols}. Note that the dependence and independence statistical concept is hard to prove in practical situations, but will see later, can be decelerated.

Algorithm and Implementation

The implementation of algorithm is relative simple, if are used a flexible language processing. In terms of programming, portability of resulted program can be a problem. As example, if we are chose to implement the algorithm in Visual Basic, the execution of the program is restricted to Windows machines. If Perl is our choice, a Unix-based machine is necessary to run program. Even if we chouse to implement the program in C language, we will have serious difficulties to compile the programs on machines running with different operating systems.

Other questions require an answer: We want a server based application or client based application? We want a server side application or a client side application?

As example, a client side application can have disadvantage of execution on client, and dependence of processing speed by power of client machine. If we prefer this variant, a java script or visual basic script is our programming language.

Most benefit to portability and execution speed seems to be a PHP (post processed hypertext) variant of implementation. A PHP script can be putted on any server or client with PHP processor and executed from them trough http server (Apache, Squid, …) and client (Internet Explorer, Netscape).

Another advantage of PHP using is the possibility to link our algorithm with a materials database (d-Base, Interbase, MySQL, PostGres format) and input data can be then loaded from them.

As conclusion, a PHP implementation is our choice.

A graphical interface was built in html with a TEXTAREA for input data and an INPUT SUBMIT button for submitting data to the server. The server is a Free BSD Unix based server with an Apache web server running on. The server is hosted in educational network of Technical University of Cluj-Napoca with address 193.226.7.140 and name lejpt.east.utcluj.ro. Five alias names are also available for them: www.lejpt.utcluj.ro, lejpt.utcluj.ro, www.ljs.utcluj.ro, ljs.utcluj.ro, lori.east.utcluj.ro

With PHP technology, was build a routine for pseudo domain names, that redirect client to different pages, depending on his input of domain name in client http browser.

The program build have 14 subroutines and a main program, specified in Table 1:

Table 1. Module Specification

Module declaration	Specification
function do_means(&$data,&$mean,$n_rows,$n_cols)	make all (xi, xi*xj) means
function af_mt($titlu,&$mean,$n_r,$n_c)	display any matrix with an title
function ch_ln($l1,$l2,&$cc,$r)	Gauss linear algebra method, change two lines in system extended matrix
function mx_rw($cl,&$cc,$r)	Gauss linear algebra method, best line for zeroes in system extended matrix
function im_ln($nr,$rw,&$cc,$r)	Gauss linear algebra method, make unitary element in system extended matrix
function ze_sd($e,&$cc,$r)	Gauss linear algebra method, make subdiag. zeroes in system extended matrix
function ze_pd(&$cc,$r)	Gauss linear algebra method, make supdiag. zeroes in system extended matrix
function rd_gs(&$cc,$r)	Gauss linear algebra iterative algorithm
function cine_e_nul(&$mean,$n_cols)	find bad variable in an undetermined system
function elimin($pe_cine,&$din_cine,$n_rows,$n_cols)	remove column of bad variable in an undetermined system
function ec_reg(&$mean,&$nule,$n_cols,&$ecuatia,$origin)	build coefficients from tables of solutions for given system
function sum_r(&$d,&$ecuatia,$r,$by)	compute sums of residues from original data and equation of regression
function af_ec(&$ecuatia)	format and display regression equation
function n_to_s($nr)	format and display a real number
main program	solve system and display results

Results and discussion

A set of Ni–Mn–Ga ferromagnetic ordered shape memory alloys are used for investigation.[13]

Table 2. Processed Data

Column	Property	Measurement unit
1	Alloy State (Poly- or Single-crystalline alloy)	1, -1 (PC, SC respectively)
2	e/a	Electron/atom ratio
3	Concentration of Ni	%
4	Concentration of Mn	%
5	Concentration of Ga	%
*6	T_1 (rows 1-7), T_M' (rows 8-10)	K
7	T_M, premartensitic temperature transformation	K

*temperatures: T_1= martensitic transformation; T_M'=intermartensitic transformation in Group III alloys.

Table 3. Input data values (outputted by PHP program)

0	1	2	3	4	5	6	7
1	1	7.35	49.6	21.9	28.5	4.2	178
2	1	7.36	47.6	25.7	26.7	4.2	152
3	-1	7.45	49.7	24.3	26.0	183	218
4	1	7.50	50.9	23.4	25.7	113	224
5	-1	7.51	49.2	26.6	24.2	184	240
6	1	7.56	47.7	30.5	21.8	227	240
7	1	7.57	51.1	24.9	24.0	197	248
8	-1	7.78	53.1	26.6	20.3	417	379
9	-1	7.83	51.2	31.1	17.7	443	415
10	-1	7.91	59.0	19.4	21.6	633	517

Program computes and output the regression equations. These equations are listed in following table:

Table 4. Output equations by unary variable coefficient (outputted by PHP program)

1	$+x_1*1.00-x_2*1.39*10^3-x_3*2.27*10^{10}-x_4*2.27*10^{10}-x_5*2.27*10^{10}-x_6*5.47*10^{-3}+x_7*0.15+2.27*10^{12}=0$
2	$-x_1*7.17*10^{-4}+x_2*1.00+x_3*1.63*10^7+x_4*1.63*10^7+x_5*1.63*10^7+x_6*3.92*10^{-6}-x_7*1.09*10^{-4}-1.63*10^9=0$
3	$-x_1*4.39*10^{-11}+x_2*6.12*10^{-8}+x_3*1.00+x_4*1.00+x_5*1.00+x_6*2.40*10^{-13}-x_7*6.71*10^{-12}-1.00*10^2=0$
4	$-x_1*4.39*10^{-11}+x_2*6.12*10^{-8}+x_3*0.99+x_4*1.00+x_5*1.00+x_6*2.40*10^{-13}-x_7*6.71*10^{-12}-1.00*10^2=0$
5	$-x_1*4.39*10^{-11}+x_2*6.12*10^{-8}+x_3*0.99+x_4*0.99+x_5*1.00+x_6*2.40*10^{-13}-x_7*6.71*10^{-12}-1.00*10^2=0$
6	$-x_1*1.82*10^2+x_2*2.54*10^5+x_3*4.15*10^{12}+x_4*4.15*10^{12}+x_5*4.15*10^{12}+x_6*1.00-x_7*2.79*10^1-4.15*10^{14}=0$
7	$+x_1*6.54-x_2*9.11*10^3-x_3*1.48*10^{11}-x_4*1.48*10^{11}-x_5*1.48*10^{11}-x_6*3.58*10^{-2}+x_7*1.00+1.48*10^{13}=0$
8	$+x_1*4.39*10^{-13}-x_2*6.12*10^{-10}-x_3*9.99*10^{-3}-x_4*9.99*10^{-3}-x_5*9.99*10^{-3}-x_6*2.40*10^{-15}+x_7*6.71*10^{-14}+1.00=0$

Sum of residues Q are presented in table 5.

Table 5. Output sums of residues in same order as in Table 4 (outputted by PHP program)

1	1.91E+01	3	1.41E-23	5	6.46E-23	7	8.94E-03
2	1.71E-07	4	5.58E-23	6	6.75E+00	8	3.68E-24

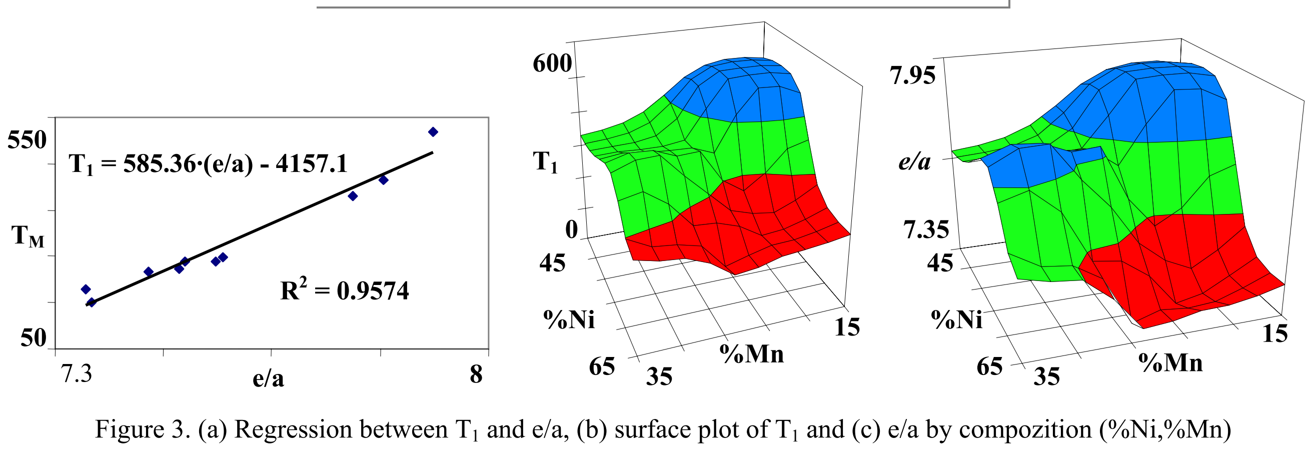

Figure 3. (a) Regression between T_1 and e/a, (b) surface plot of T_1 and (c) e/a by compozition (%Ni,%Mn)

Conclusions

Looking to the output sums of residues from table 5, is easy to observe now that the properties type of alloy, and his martensitic, intermartensitic and premartenistic temperatures are interrelated having same order of sum residues in global equation, that is also expected conclusion. Very small same order of sum residues for concentrations suggest a strong interrelation between them, that is also true, because %Ni+%Mn+%Ga = 100. This conclusion lead to consider the 3D plots fitted in figure 3 (b and c) of electron/atom ratio and T_1 temperature dependencies by concentration (%Ni,%Mn). The figure 3a prove good correlation between T_1 and e/a.

Acknowledgments

Author is grateful to the rector of Technical University of Cluj-Napoca, prof. dr. ing. Gheorghe Lazea for his policy on promoting information technology and also to the university staff for support in internet connection of lejpt web server.

Useful support was also benefit from Romanian Ministry of Education for finance funding of MEC/CNCSIS contract 468/"A"/2002 and 281/"A"/2002.

References

[1] Horea Nascu, Lorentz Jäntschi, Teodor Hodisan, Claudia Cimpoiu, Gabriela Câmpan, Some Applications of Statistics in Analytical Chemistry, Reviews in Analytical Chemistry, Freud Publishing House, XVIII(6), 1999, 409-456.

[2] Lorentz Jäntschi, Mihaela Unguresan, MathLab. Statistical Applications, International Conference on Quality Control, Automation and Robotics, May 23-25 2002, Cluj-Napoca, Romania, work published in volume "AQTR Theta 13" (ISBN 973-9357-10-3) at p. 194-199.

[3] Lorentz Jäntschi, Horea Iustin Nascu, Numerical Description of Titration (MathCad application), International Conference on Quality Control, Automation and Robotics, May 23-25 2002, Cluj-Napoca, Romania, work published in volume "AQTR Theta 13" (ISBN 973-9357-10-3) at p. 259-262.

[4] Lorentz Jäntschi, Mihaela Unguresan, Parallel processing of data. C++ Applications, Oradea University Annals, Mathematics Fascicle, 2001, VIII, 105-112, ISSN 1221-1265.

[5] Liana Marta, Ion Grigore Deac, Ion Fruth, Maria Zaharescu, Lorentz Jäntschi, Superconducting Materials: Comparision Between Coprecipitation and Solid State Phase Preparation, International Conference on Science of Materials, August 20-22, 2000, Cluj-Napoca, Romania, work published in volume 2 (ISBN 973-686-058-2) at p. 627-632.

[6] Liana Marta, Ion Grigore Deac, Ion Fruth, Maria Zaharescu, Lorentz Jäntschi, Superconducting Materials at High Temperature: Thermal Treatment and Pb Addtition, International Conference on Science of Materials, August 20-22, 2000, Cluj-Napoca, Romania, work published in volume 2 (ISBN 973-686-058-2) at p. 633-636.

[7] Claudia Cimpoiu, Lorentz Jäntschi, Teodor Hodisan, A New Method for Mobile Phase Optimization in High-Performance Thin-Layer Chromatography (HPTLC), Journal of Planar Chromatography - Modern TLC, Springer, 11(May/June), 1998, 191-194, ISSN 0933-4173.

[8] Claudia Cimpoiu, Lorentz Jäntschi, Teodor Hodisan, A New Mathematical Model for the Optimization of the Mobile Phase Composition in HPTLC and the Comparision with Other Models, Journal of Liquid Chromatography and Related Technologies, Dekker, 22(10), 1999, 1429-1441, ISSN 1082-6076.

[9] Mircea Diudea, Lorentz Jäntschi, Ante Graovac, Cluj Fragmental Indices, Math/Chem/Comp'98 (The Thirteenth Dubrovnik International Course & Conference on the Interfaces among Mathematics, Chemistry and Computer Sciences), June 22-27, 1998.

[10] Lorentz Jäntschi, Gabriel Katona, Mircea Diudea, Modeling Molecular Properties by Cluj Indices, Commun. Math. Comput. Chem. (MATCH), 2000, 41, 151-188, ISSN 0340-6253.

[11] Costel Sârbu, Lorentz Jäntschi, Statistic Validation and Evaluation of Analytical Methods by Comparative Studies. I. Validation of Analytical Methods using Regression Analysis (in Romanian), Revista de Chimie, Bucuresti, 49(1), 1998, 19-24, ISSN 0034-7752.

[12] Mircea Diudea, Ivan Gutman, Lorentz Jäntschi, Molecular Topology, Nova Science, Huntington, New York, 2001, 332 p., ISBN 1-56072-957-0.

[13] Chernenko V.A., Pons J., Seguý´ C., Cesari E., Premartensitic phenomena and other phase transformations in Ni–Mn–Ga alloys studied by dynamical mechanical analysis and electron diffraction, Acta Materialia, 50 (2002), 53–60, Elsevier, USA.