Grasas y Aceites
Vol. 53. Fasc. 1 (2002), 115-127

Chemometrics: From Classical to Genetic Algorithms

By **Riccardo Leardi**

Department of Pharmaceutical and Food Chemistry and Technology, University of Genova, via Brigata Salerno (ponte), I-16147 Genova, Italy. E-mail: riclea@dictfa.unige.it

CONTENTS

RESUMEN

Quimiometria: De los algoritmos clásicos a los genéticos.

En este artículo se muestran los aspectos fundamentales de la Quimiometria por medio de una revisión rápida de las técnicas más relevantes para mostrar los datos, modelar y calibrar. Se describen dos técnicas emergentes como los algoritmos genéticos y las redes neuronales. El objetivo del articulo es que la comunidad cientifica tome conciencia de la gran superioridad del análisis multivariante sobre el análisis univariante. No se describen los detalles matemáticos y algorítmicos porque el articulo está dirigido a problemas genéricos en los que la Quimiometría puede ser aplicada con éxito dentro del campo de la Química Analítica.

PALABRAS-CLAVE: Análisis multivariante - Calibración – Clasificación – Modelos – Presentación de datos - Quimiometría.

SUMMARY

Chemometrics: From classical to genetic algorithms.

In this paper the fundamentals of Chemometrics are presented, by means of a quick overview of the most relevant techniques for data display, classification, modeling and calibration. Two emerging techniques such as Genetic Algorithms and Artificial Neural Networks will also be presented. Goal of the paper is to make people aware of the great superiority of multivariate analysis over the commonly used univariate approach. Mathematical and algorithmical details are not presented, since the paper is mainly focused on the general problems to which Chemometrics can be successfully applied in the field of Food Chemistry.

KEY-WORDS: Calibration - Chemometrics - Classification – Data display – Modeling – Multivariate analysis.

1. INTRODUCTION

I am well aware that many of the readers of this book are not familiar with chemometrics, and that a relevant percentage among them have never even heard about this “new” science (it is quite funny that it is still considered as a “new” science, when the Chemometrics Society has been funded 30 years ago and the most basic algorithms date back to the beginning of the century …). I also know very well that some among the readers are quite frightened by everything involving mathematical computations higher than a square root or statistical tests more complex than a t test.

Therefore, the goal I set for myself in writing this contribution is simply that of being read and understood by the majority of the readers of this Journal; I will be completely satisfied if some of them, after having read it, would say: “Chemometrics is easy and powerful indeed, and from now on I will always think in multivariate way”.

Of course, to accomplish this goal in the reduced space of a chapter I must try to highlight the attractive sides of chemometrics, without giving too much relevance to the algorithms. Therefore, except for Principal Component Analysis, that is the basis of multivariate techniques, I will always try to show the intuitive aspects of each technique.

The reader interested in a deeper knowledge of chemometrics will find at the end of this chapter a short list of books and web sites that can be used as textbooks.

First of all, what is Chemometrics? According to the definition of the Chemometrics Society, it is “the chemical discipline that uses mathematical and statistical methods to design or select optimal procedures and experiments, and to provide maximum chemical information by analyzing chemical data”.

One of the major mistakes people do about chemometrics is thinking that to use it one has to be a very good mathematician and to know the mathematical details of the algorithms he is using. From the definition itself, it is clear instead that a chemometrician is a chemist (the word “chemical” appears three times) who can use mathematical and statistical methods.

If we want to draw a parallel with everyday life, how many among us do really know in detail how do a TV set, a telephone, a car or a washing machine work? Anyway, everybody watches TV programs, makes phone calls, drives a car and starts a washing machine. Of course, what is important is that people know what each instrument is made for and that

nobody tries to watch inside a telephone, or to drive a TV set, or to speak inside a washing machine or to do the laundry in a car…

People can deal with chemometrics at different levels, according to their knowledge and to the time they want to (or can) give to chemometrics. Roughly speaking, I can divide them into four levels (from the highest to the lowest):

level 1: full-time chemometricians developing new algorithms;

level 2: full-time chemometricians applying chemometrics to problems of other people;

level 3: part-time chemometricians able to solve their simplest problems by applying basic chemometrics and giving their most complex problems to level 2 chemometricians for being solved;

level 4: people who, though not knowing how to use chemometrics, are anyway well aware of its potential and give their problems to level 2 or level 3 chemometricians for being solved.

Of course, the required knowledge of the algorithms decreases when going down to the scale: while at level 1 a detailed knowledge and high mathematical skills are required, at levels 2 and 3 it is important to know the principles on which the techniques are based, in such a way that they can be applied in an appropriate way; at level 4 what one has to know is that chemometrics and chemometricians do exist and that they can solve a lot of problems …

It has also to be considered that the great majority of the real problems can be solved by applying one of the basic techniques, whose understanding, at least from an intuitive point of view, is relatively easy and does not require high-level mathematical skills.

2. DATA COLLECTION

Chemometrics works on data matrices. This means that on each sample a certain number of variables have been measured (in the "chemometrical jargon" we say that each object is described by *p* variables). Although some techniques can allow to deal with a limited amount of missing values, a chemometrical data set must be thought of as a spreadsheet in which all the cells are full:

	var. 1	var. 2	var. 3	var. 4	var. 5	var. 6	var. 7	...	var. p
obj. 1									
obj. 2									
obj. 3									
obj. 4									
obj. 5									
obj. 6									
.....									
obj. n									

Sometimes, instead, if data are gathered without having any specific project, it happens that the result is a "sparse" matrix, in which not all the cells contain a value. In that case, if the percentage of missing data is quite high, the whole data set is not suitable for a multivariate analysis; as a consequence, the variables and/or the objects with the lowest number of data must be removed, and therefore a huge amount of experimental effort can be lost.

All the chemometrical software allows the import of data from ASCII files or from spreadsheets. It is therefore suggested to organise the data from the beginning in matrix form, in such a way that the import can be performed in a single step. If, on the contrary, the data are spread in several files or sheets (e.g., one file for each sample or for each variable), then the import procedure would be much longer and more cumbersome.

3. DATA DISPLAY

Human mind can get much more information when looking at plots than at numbers. This is easily shown by taking into account at first the sequence of numbers reported in Table I, and then the plot in Figure 1.

It is very clear that, also in a very simple data set like this one (just 10 samples, and only 1 variable) the information obtained by looking at the plot is superior and much more easily available than the information one can get by analysing the raw numbers. From the plot, it is very evident that the samples are clustered into two groups of the same size, the one at higher values being much tighter than the one at low values; much more time and effort is required when we want to get the same information from the table.

Let us now take into account a more complex data set as the one reported in Table II, in which each

Table I
Ten samples described by one variable

Sample	1	2	3	4	5	6	7	8	9	10
Value	25.3	22.1	25.5	25.6	19.4	25.7	20.2	21.3	25.9	21.8

Figure 1
Scatter plot of the data in Table I.

Table II
Twenty samples described by two variables

sample	var. 1	var. 2
1	21.2	32.5
2	16.2	21.0
3	13.1	21.7
4	11.6	21.3
5	20.8	29.9
6	10.4	20.6
7	19.5	26.8
8	9.8	25.2
9	15.2	31.2
10	12.0	26.0
11	17.6	28.5
12	24.0	30.0
13	17.8	33.1
14	15.0	24.0
15	11.0	24.2
16	24.8	25.3
17	12.8	23.3
18	26.5	30.6
19	22.9	27.5
20	9.7	22.8

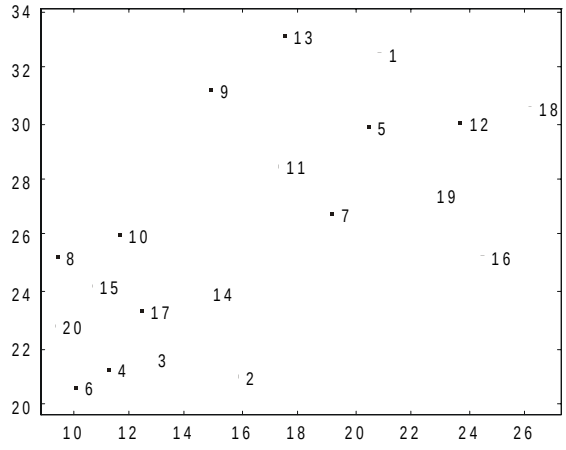

Figure 2
Scatter plot of the data in Table II.

object is described by two variables. The same data are plotted in Figure 2.

This bivariate data set, beyond showing once more that a plot is much more easily handled by the human brain than a data table, demonstrates that, when dealing with more than one variable, the analysis of just one variable at a time can lead to wrong results.

In this data set we have 20 samples, supposed to belong to the same population. When looking at the plot, we realise that we are in a situation very similar to what we found with the univariate data set: the samples are split into two clusters of the same size, with the objects of the first one more tightly grouped than the objects of the second one. This conclusion cannot be reached when looking at one variable at a time, since none of the two variables is able to discriminate between the two groups.

If we had a data set with three variables it would still be possible to visualise the whole information by a tridimensional scatter plot, in which the co-ordinates of each object are the values of the variables. But what to do if the variables are more than three?

What we need is therefore a technique allowing to visualise by simple bi- or tri-dimensional scatter plots the majority of the information contained in a highly dimensional data set. This technique is the Principal Component Analysis (PCA), one of the simplest and most used methods of multivariate analysis. PCA is very important especially in the preliminary steps of an elaboration, when one wants to perform an exploratory analysis in order to have an overview of the data.

It is rather common to have to deal with large data tables, in which, for instance, a series of samples is described by a number (p) of chemico-physical parameters. Examples of such data sets can be samples of olive oils from different origins described by their content in fatty acids and sterols, or samples of wines described by FTIR spectra. It is easy to realise how, especially in spectral data sets, p can be really very high (>1000); in such cases it would be impossible to obtain valuable information without the help of multivariate techniques.

From a geometrical point of view, we can consider a p-dimensional space, in which each dimension is associated to one of the variables. In this space each sample (object) has co-ordinates corresponding to the values of the variables describing it. Since it is impossible to visualise all the information at once, one should stay content with the analysis of several bi- or three-dimensional plots, each of them showing a different part of the global information.

It is also evident that not all possible combinations of two or three variables will give the same quality of information; for instance, if some variables are very highly correlated, then the information brought by each of them would be almost the same.

If two variables are perfectly correlated, then one of them can be discarded, losing no information at all; in this way, the dimensionality of our space will be reduced from p to p-1. If two variables are very highly correlated, then the elimination of one of them would produce only a slight loss of information, while the dimensionality of the space would be reduced to p-1. So, one can deduce that the information contained in the “lost” p-th dimension was well below the average of the information contained in the other dimensions.

It is quite apparent now that not all the dimensions have the same importance, and that, owing to the correlations among the variables, the "real" dimensionality of our data matrix is somehow lower than *p*. Therefore, it would be very valuable to have a technique capable of concentrating in a few variables, and therefore in a few dimensions, the bulk of our information.

This is exactly what is performed by PCA: it reduces the dimensionality of the data and extracts the most relevant part of the information, placing into the last dimensions the non-structured information, i.e. the noise; according to these two characteristics, the information contained in very complex data matrices can be visualised in just one or a few plots.

From the mathematical point of view, the goal of PCA is to obtain, from *p* variables (X_1, X_2, …, X_p), *p* linear combinations having two important features: to be uncorrelated and to be ordered according to the explained variance (i.e., to the information they contain).

The lack of correlation among the linear combinations is very important, since it means that each of them describes different "aspects" of the original data. As a consequence, the examination of a limited number of linear combinations (generally the first two or three) allows us to obtain a good representation of the studied data set.

From a geometrical point of view, what is performed by PCA corresponds to look for the direction that, in the *p*-dimensional space of the original variables, brings the greatest possible amount of information (i.e., explains the greatest variance). Once the first direction is identified, the second one is looked for: it will be the direction explaining the greatest part of the residual variance, under the constraint of being orthogonal to the first one. This process goes on until the *p*-th direction has been found.

These new directions can be considered as the axes of a new orthogonal system, obtained after a simple rotation of the original axes. While in the original system each direction (i.e., each variable) brings with it, at least in theory, 1/*p* of total information, in the new system the information is concentrated in the first directions, and decreases progressively so that in the last ones no information, but only noise, can be found.

The global dimensionality of the system is always that of the original data (*p*), but, since the last dimensions explain only a very small part of the information, they can be neglected and one can take into account only the first dimensions (the "significant components"). The projection of the objects in this space of reduced dimensionality retains almost all the information, which can now be analysed also in a visual way, by bi- or three-dimensional plots.

These new directions, linear combinations of the original ones, are the Principal Components (PC) (or eigenvectors).

With a mathematical notation, we can write:

$$\text{var}(Z_1) > \text{var}(Z_2) > \ldots > \text{var}(Z_p)$$

where $\text{var}(Z_i)$ is the variance explained by component *i*.

Furthermore, since a simple rotation has been performed, the total variance is the same in the two systems of axes:

$$\Sigma\, \text{var}(X_i) = \Sigma\, \text{var}(Z_i)$$

The first PC is formed by the linear combination

$$Z_1 = a_{11}X_1 + a_{12}X_2 + \ldots + a_{1p}X_p$$

explaining the greatest variance, under the condition that

$$\sum a_{1i}^2 = 1$$

This last condition notwithstanding, the variance of Z_1 could be made greater simply by increasing one of the values of a.

The second PC

$$Z_2 = a_{21}X_1 + a_{22}X_2 + \ldots + a_{2p}X_p$$

is the one having $\text{var}(Z_2)$ as large as possible, under the conditions that

$$\sum a_{2i}^2 = 1$$

and that

$$\sum a_{1i}\, a_{2i} = 0$$

(this last condition assures the orthogonality of components 1 and 2).

The lower order components are computed in the same way, always under the two conditions previously reported.

From a mathematical point of view, PCA is solved by finding the eigenvalues of the variance-covariance matrix; they correspond to the variance explained by the corresponding principal component. Since the sum of the eigenvalues corresponds to the sum of the diagonal elements (trace) of the variance-covariance matrix, and the latter corresponds to the total variance, one has the confirmation that the variance explained by the principal components is the same explained by the original data.

It is now interesting to locate each object into this new reference space. The co-ordinate on the first PC is computed simply by substituting into equation $Z_1 = a_{11}X_1 + a_{12}X_2 + \ldots + a_{1p}X_p$ the terms X_i with the values of the corresponding original variables. The co-ordinates on the other principal components are then computed in the same way.

These co-ordinates are named scores, while the constants a_{ij} are named loadings. By taking into account the loadings of the variables on the different principal components, it is very easy to understand the importance of each single variable in constituting each PC; a high absolute value means that the variable under examination plays an important role for the component, while a low absolute value means that it has a very limited importance.

If a loading has positive sign, it means that the objects with a high value of the corresponding variable have high scores on that component; if the sign is negative, then the objects with low values of the variables will have high scores.

As already mentioned, after a PCA the information is mainly concentrated on the first components. As a consequence of that, a plot of the scores of the objects on the first components allows the direct visualisation of the global information in a very efficient way; it is now very easy to detect similarity between objects (similar objects have a very similar position in the space) or the presence of outliers (they are very far from all other objects) or the existence of clusters.

Taking into account at the same time scores and loadings it is also possible to interpret very easily the differences among objects or groups of objects, since it is very immediate to understand which are the variables giving the greatest contribution to the phenomenon under study. Now, let us see the application of PCA to a real data set. Seven variables describing the protein composition have been measured on 23 samples of peas, of different cultivars. 15 samples were from smooth pea cultivars, while 8 samples were from wrinkled pea cultivars. The data are reported in Table III.

It could be interesting to check whether the protein composition of the smooth peas is different from that of the wrinkled peas. When looking separately at each of the seven variables, it can be seen that none of them completely separates the two categories. Therefore, one could say that, though some variables are on average higher in one category (e.g., the vicilin/legumin ratios are higher in the wrinkled peas), it is not possible to discriminate

Table III

Protein composition of peas (Gueguen 1988) (reduced data set). (a) 1 = smooth pea cultivars; 2 = wrinkled pea cultivars; (b) Laurell's technique; (c) ultracentrifugation

Object	Category (a)	Protein	Non-prot. material	Albumin	Globulin	Insoluble Prot. Fract.	Vicilin/ legumin (b)	Vicilin/ legumin (c)
1	1	219	20.7	24.3	55.7	20.0	2.2	2.0
2	1	273	30.2	12.3	61.0	26.6	1.3	1.5
3	1	255	17.8	19.3	53.8	26.9	1.5	2.0
4	1	262	30.2	13.1	63.2	23.5	1.6	2.3
5	1	242	20.8	20.8	52.6	26.5	0.8	1.3
6	1	235	16.1	23.2	60.8	16.0	0.8	1.4
7	1	272	14.9	17.9	62.1	19.9	0.8	1.3
8	1	235	24.5	25.1	59.6	14.9	0.8	1.4
9	1	225	22.0	25.0	58.8	16.1	1.9	1.8
10	1	195	20.0	15.1	58.6	26.2	2.1	2.1
11	1	181	18.7	16.1	65.4	18.4	2.7	3.2
12	1	236	16.6	20.0	57.0	23.0	1.2	1.6
13	1	261	22.1	19.2	63.7	17.0	1.3	1.6
14	1	244	21.9	19.6	65.0	22.2	1.8	1.9
15	1	239	32.1	27.9	58.0	14.1	1.6	1.6
16	2	263	19.8	21.9	59.4	18.6	2.5	2.5
17	2	263	20.3	22.8	60.3	16.8	2.9	2.8
18	2	309	18.5	24.6	58.5	16.8	2.2	2.5
19	2	241	16.7	24.0	58.6	17.3	2.5	3.7
20	2	241	19.3	24.6	55.6	19.7	3.2	3.2
21	2	292	21.3	20.0	54.6	25.3	2.0	3.0
22	2	287	21.2	21.5	54.7	23.7	4.3	3.3
23	2	278	20.0	23.1	55.6	21.3	2.5	4.7

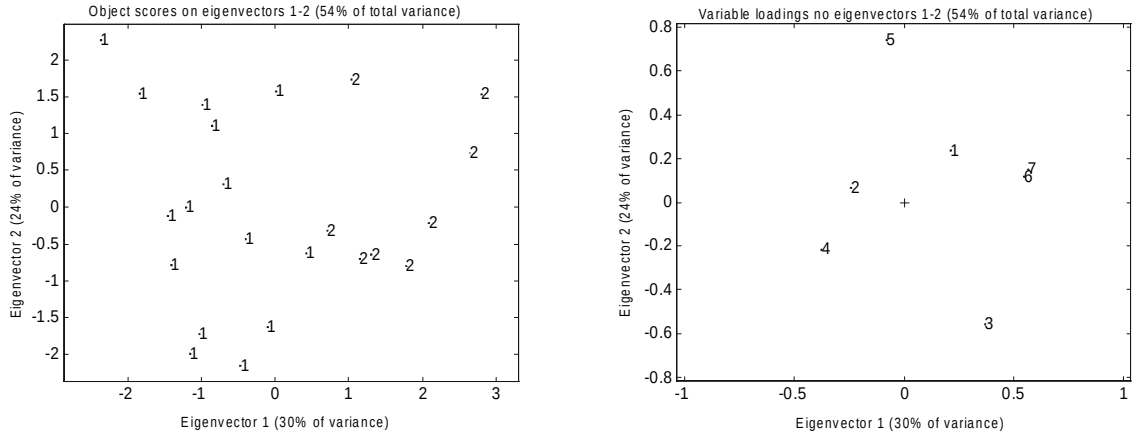

Figure 3
PCA of the data of Table III. On the left, the score plot of the objects (coded according to the category number), on the right the loading plot of the variables (coded according to the order in Table III).

between smooth and wrinkled peas. As a consequence, one could look for different (and possible more expensive to be determined) variables.

After a PCA (Figure 3), it is instead evident that the information present in the seven variables is sufficient to clearly discriminate the two categories. Once more, it has to be pointed out that taking into account all the variables at the same time gives much more information than just looking at one variable at a time.

Now, let us go one step back and let us try to understand how this result has been obtained. At first, since the variables have different magnitudes and different variances, a normalisation has to be performed, in such a way that each variable will have the same importance. Autoscaling is the most frequently used normalisation: it subtracts from each variable the mean value, and divides the result by the standard deviation of that variable. After that, each variable will have mean = 0 and variance = 1. Table IV shows the data after autoscaling.

The results of PCA are such that PC1 explains 30.3% of the total variance and PC2 23.6%. This means that the PC1-PC2 plots shown in Figure 3 explain 53.9% of total variance. Table V shows the loadings of the variables on PC1 and PC2. From it, the loading plot in Figure 3 is obtained.

From the score plot in Figure 3 it can be seen that PC1 perfectly separates the two categories. By looking at the loading plot and at Table V it is possible to know which are the variables mainly contributing to PC1 (and therefore to the separation). Variables 6 and 7 (the two vicilin/legumin ratios) have the loadings with the highest absolute values, both being positive. This means that these ratios are higher in the wrinkled peas (the objects of category 2, being on the right side of the score plot, have higher scores on PC1) than in the smooth peas. Also albumin and globulin have high absolute value of their loadings on PC1, though having opposite sign (positive for albumin, negative for globulin). This means that

Table IV
Autoscaled data

Protein	Non-prot. material	Albumin	Globulin	Insoluble Prot. Fract.	Vicilin/ legumin (b)	Vicilin/ legumin (c)
-1.040	-0.094	0.837	-0.871	-0.115	0.304	-0.326
0.777	2.042	-2.144	0.614	1.495	-0.727	-0.887
0.171	-0.746	-0.405	-1.403	1.569	-0.498	-0.326
0.407	2.042	-1.946	1.230	0.739	-0.383	0.010
-0.266	-0.071	-0.032	-1.739	1.471	-1.300	-1.111
-0.502	-1.128	0.564	0.558	-1.090	-1.300	-0.999
0.743	-1.398	-0.753	0.922	-0.139	-1.300	-1.111
-0.502	0.760	1.036	0.222	-1.359	-1.300	-0.999
-0.838	0.198	1.011	-0.002	-1.066	-0.040	-0.551
-1.847	-0.251	-1.449	-0.058	1.398	0.189	-0.214
-2.318	-0.543	-1.200	1.846	-0.505	0.876	1.018
-0.468	-1.015	-0.231	-0.507	0.617	-0.842	-0.775
0.373	0.221	-0.430	1.370	-0.846	-0.727	-0.775
-0.199	0.176	-0.331	1.734	0.422	-0.154	-0.439
-0.367	2.469	1.732	-0.227	-1.554	-0.383	-0.775
0.440	-0.296	0.241	0.166	-0.456	0.647	0.234
0.440	-0.184	0.465	0.418	-0.895	1.105	0.570
1.988	-0.588	0.912	-0.086	-0.895	0.304	0.234
-0.300	-0.993	0.763	-0.058	-0.773	0.647	1.579
-0.300	-0.409	0.912	-0.899	-0.188	1.449	1.018
1.416	0.041	-0.231	-1.179	1.178	0.075	0.794
1.248	0.019	0.142	-1.151	0.788	2.709	1.130
0.945	-0.251	0.539	-0.899	0.203	0.647	2.699

Table V
Loadings of the variables on PC1 and PC2

	Protein	Non-prot. material	Albumin	Globulin	Insoluble Prot. Fract.	Vicilin/ legumin (b)	Vicilin/ legumin (c)
PC1	0.214	-0.239	0.370	-0.372	-0.080	0.546	0.563
PC2	0.237	0.066	-0.557	-0.219	0.739	0.115	0.151

Table VI
Scores of the objects on PC1 and PC2

Object	Category	Score on PC1	Score on PC2
1	1	0.425	-0.627
2	1	-2.358	2.264
3	1	0.006	1.576
4	1	-1.841	1.548
5	1	-0.858	1.100
6	1	-1.023	-1.735
7	1	-1.453	-0.119
8	1	-1.153	-1.998
9	1	-0.099	-1.623
10	1	-0.978	1.386
11	1	-0.404	-0.439
12	1	-0.700	0.304
13	1	-1.408	-0.784
14	1	-1.217	-0.004
15	1	-0.466	-2.148
16	2	0.714	-0.312
17	2	1.151	-0.705
18	2	1.304	-0.647
19	2	1.781	-0.806
20	2	2.085	-0.226
21	2	1.040	1.724
22	2	2.797	1.535
23	2	2.653	0.737

wrinkled peas have higher content of albumin and lower content of globulin. Table VI reports the scores of the objects on PC1 and PC2.

As previously shown, the scores of an object are computed by multiplying the loadings of each variables by the value of the variable. As an example, let us compute the score of sample 1 on PC1 (since the autoscaled data have been used, these are the values that must be taken into account):

$$0.214*(-1.040) + (-0.239)*(-0.094) + 0.370*0.837 + (-0.372)*(-0.871) + (-0.080)*(-0.115) + 0.546*0.304 + 0.563*(-0.326) = 0.425$$

4. CLASSIFICATION

In the previous section we could verify that the smooth and the wrinkled peas are indeed well separated in the multivariate space of the variables. Therefore, we can say that we have two really different classes. Let us suppose we now get some smashed peas (so that we can not see if they are smooth or wrinkled) and we want to know which is their class. After having performed the chemical analyses, we can add these data to the previous data set, run a PCA and see where the new samples are placed. This will be fine if the new samples fall inside one of the clouds of points corresponding to a category, but what if they fall in a somehow intermediate position? How can we say with "reasonable certainty" that the new samples are from a smooth or from a wrinkled pea? We know that PCA is a very powerful technique for data display, but we realise that we need something different if we want to classify new samples. What we want is a technique producing some "decision rules" discriminating among the possible categories. While PCA is an "unsupervised" technique, the classification methods are "supervised" techniques, since they must be told to which category each of the objects belongs.

The most commonly used classification techniques are Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA). They define a set of delimiters (according to the number of categories under study), in such a way that the multivariate space of the objects is divided in as many subspaces as the number of categories, and that each point of the space belongs to one and only one subspace. Rather than describing in detail the algorithms behind these techniques, I will focus on the critical points of a classification.

As I said earlier, the classification techniques use objects belonging to the different categories to define boundaries delimiting regions of the space. The final goal is to apply these classification rules to new objects that will be classified into one of the existing categories.

The performance of the technique can be expressed as classification ability and prediction ability. The difference between "classification" and "prediction", though quite subtle at a first glance, is instead very important and its underestimation can lead to very bitter deceptions. The classification ability is the capability of assigning to the correct category the same objects that have been used to build the classification rules, while the prediction ability is the capability of assigning to the correct category objects that have not been used to build the classification rules. Since the final goal is the classification of new samples, it has to be clear that the predictive ability is by far the most important score to be looked at.

The results of a classification method can be expressed in several ways. The most synthetic one is the percentage of correct classifications (or predictions. Note: in the following, only the term "classification" will be used, but it has to be understood as "classification or prediction"). This can be obtained as the number of correct classifications (independently of the category) divided by the total number of objects, or as the average of the performance of the model over all the categories. The two results are very similar when the size of all the categories is very similar, but can be very different if the size is quite different. Let us consider

Table VII
Example of the performance of a classification technique

Category #	Objects	Correct class.	% Correct class.
1	112	105	93.8
2	87	86	98.9
3	21	10	47.6
total	220	201	91.4/80.1

Table VIII
Example of a classification matrix

Category	1	2	3
1	105	0	7
2	1	86	0
3	11	0	10

the case shown in Table VII. The very poor performance of category 3, by far the smallest one, almost does not affect the classification rate computed on the global number of classification, while it produces a much lower result if the classification rate is computed as the average of the three categories.

A more complete and detailed overview of the performance of the method can be obtained by using the classification matrix, by which also the categories to which the wrongly classified objects are assigned can be known (in many cases the cost of an error can be quite different according to the category the sample is assigned to). In it, each row corresponds to the true category and each column to the category to which the sample has been assigned. Going on with the previous example, a possible classification matrix is the shown in Table VIII.

From it, it can be seen that the 112 objects of category 1 were classified in the following way: 105 correctly to category 1, none to category 2 and 7 to category 3. In the same way, it can be deduced that all the objects of category 3 that were not correctly classified have been assigned to category 1. Therefore, it is easy to conclude that category 2 is well defined and that the classification of its objects gives no problems at all, while categories 1 and 3 are quite overlapping; as a consequence, to have a perfect classification more efforts must be done to better separate categories 1 and 3. All these information cannot be obtained from just the percentage of correct classifications.

If overfitting occurs, then the prediction ability will be much worse than the classification ability. To avoid it, it is very important that the sample size is adequate to the problem and to the technique. A general rule is that the number of objects should be more than 5 times (anyway, no less than 3 times) the number of parameters to be estimated. LDA works on a pooled variance-covariance matrix: this means that the total number of objects should be at least 5 times the number of variables. QDA computes a variance-covariance matrix for each category; this makes it a more powerful method than LDA, but this also means that each category should have a number of objects at least 5 times higher than the number of variables. This is a good example of how the more complex, and therefore “better” methods, sometimes can not be used in a safe way because their requirements do not correspond to the characteristics of the data set.

5. MODELLING

In classification, the space is divided into as many subspaces as categories, and each point belongs to one and only one category. This means that the samples that will be predicted by such methods must belong to one of the categories that have been used to build the models; if not, they will anyway be assigned to one of them. To make this concept clearer, let us suppose to use a classification technique to discriminate between water and wine. Of course, this discrimination is very easy, and each sample of water will be correctly assigned to the category “water” and each sample of wine will be correctly assigned to the category “wine”. But what happens with a sample of orange squash? It will be assigned either to the category “water” (if variables such as alcohol are taken into account) or to the category “wine” (if variables such as colour are considered).

The classification techniques are therefore not able to define a new sample as being “something different” from all the categories of the training set. This is instead the main feature of the modelling techniques. Though several techniques are used for modelling purpose, UNEQ (one of the modelling versions of QDA) and SIMCA (Soft Independent Model of Class Analogy) are the most used.

While in classification every point of the space belongs to one and only one category, with these techniques the models (one for each category) can overlap and leave some regions of the space unassigned. This means that every point of the space can belong to one category (the sample has been recognised as a sample of that class), to more than one category (the sample has such characteristics that it could be a sample of more than one class) or to none of the categories (the sample has been considered as being different from all the classes).

Of course, the “ideal” performance of such a method would be not only to correctly classify all the samples in their category (as in the case of a

classification technique), but also that the models of each category could be able to accept all the sample of that category and to reject all the samples of the other categories.

The results of a modelling technique are expressed the same way as in classification, plus two very important parameters: specificity and sensitivity. For category *c*, its specificity (how much the model rejects the objects of different categories) is the percentage of the objects of categories different from *c* that have been rejected by the model, while its sensitivity (how much the model accepts the objects of the same category) is the percentage of the objects of category *c* that have been accepted by the model.

While the classification techniques need at least two categories, the modelling techniques can be applied also when only one category is present. In this case the technique detects if the new sample can be considered as a typical sample of that category or not. This can be very useful in case of Protected Denomination of Origin products, to verify whether a sample, declared as having been produced in a well defined region, has indeed the characteristics typical of the samples produced in that region.

The application of a multivariate analysis will reduce very much the possibility of frauds. While an "expert" can adulterate a product in such a way that all the variables, independently considered, still stay in the accepted range, it is almost impossible to adulterate a product in such a way that its multivariate "pattern" is still accepted by the model of the original product, unless the amount of the adulterant is so small that it becomes no more profitable from the economic point of view.

6. CALIBRATION

Let us imagine to have a set of wine samples and that on each of them the FTIR spectrum is measured, together with some variables such as alcohol content, pH or total acidity. Of course, the chemical analysis will require much more time than a simple spectral measurement. It would therefore be very useful to find a relationship between each of the chemical variables and the spectrum. This relationship, after having been established and validated, will be used to predict the content of the chemical variables. It is easy to understand how much time (and money) this will save, since in a few minutes it will be possible to have the same results previously obtained by a whole set of chemical analyses.

Generally speaking, we can say that multivariate calibration finds relationships between one or more response variables y and a vector of predictor variables x. As the previous example should have shown, the final goal of multivariate calibration is not just to "describe" the relationship between the x and the y variables in the set of samples on which the relationship has been computed, but to find a real practical application on samples that in a following time will have the x variables measured.

The model is a linear polynomial ($y = b_0 + b_1x_1 + b_2x_2 + \ldots + b_Kx_K + f$), where b_0 is an offset, the b_K ($k = 1, \ldots, K$) are regression coefficients and f is a residual.

The "traditional" method of calculating b, the vector of regression coefficients, is ordinary least squares (OLS). This method has anyway two major limitations, that make it inapplicable to many data sets:

- it can not handle more variables than objects;
- it is sensitive to collinear variables.

It can be easily seen that both these limitations do not allow to apply OLS to spectral data sets, where the samples are described by a very a high number of highly collinear variables. If one wants anyway to use OLS to such data, the only way to do it is to reduce the number of variables and their collinearity through a suitable feature selection (see later).

When describing the PCA, it has been noticed that the components are orthogonal (i.e., uncorrelated) and that the dimensionality of the resulting space (i.e., the number of significant components) is much lower than the dimensionality of the original space. Therefore, it can be seen that both the aforementioned limitations have been overcome. As a consequence, it is possible to apply OLS to the scores originated by PCA. This technique is Principal Component Regression (PCR).

It has anyway to be considered that the Principal Components are computed by taking into account only the x variables, without considering at all the y variable(s), and are ranked according to the explained variance of the "x world". This means that it can happen that the first PC has little or no relevance in explaining the response we are interested to. This can be easily understood by considering that, even when we have several responses, the PC's to which the responses have to be regressed will be the same.

Nowadays, the most favoured regression technique is Partial Least Squares Regression (PLS, or PLSR). As it happens with PCR, PLS is based on components (or "latent variables"). The PLS components are anyway computed by taking into account both the x and the y variables, and therefore they are slightly rotated versions of the Principal Components. As a consequence, the order by which they are ranked corresponds to the importance in the modelling of the response. A further difference with OLS and PCR is that, while the former must work on each response variable separately, PLS can be applied to multiple responses at the same time.

Being both PCR and PLS based on latent variables, a very critical point is the number of components that have to be retained. Though we know that information is "concentrated" in the first components and that the last components explain just noise, it is not always an easy task to detect the correct number of components (i.e., when information finishes and noise begins). Selecting a lower number of components would mean to remove some useful information (underfitting), while selecting a higher number of components would mean to incorporate some noise (overfitting).

Before applying the results of a calibration, it is very important to look for the presence of outliers. Three major types of outliers can be detected: outliers in the x-space (samples for which the x-variables are very different from that of the rest of the samples; they can be found by looking at a PCA of the x-variables), outliers in the y-space (samples for which the y-variable is very different from those of the rest of the samples; they can be found by looking at a histogram of the y-variable) and samples for which the calibration model is not valid (they can be found by looking at a histogram of the residuals).

The goodness of a calibration can be summarised by two values: the percent of variance explained by the model and the Root Mean Square Error in Calibration (RMSEC). The former, being a "normalised" value, gives a first idea about how much of the variance of the data set is "captured" by the model; the latter, being an absolute value to be interpreted the same way as a standard deviation is, gives information about the magnitude of the error.

As already described in the classification section and as pointed out at the beginning of this section, the goal of a calibration is essentially not to describe the relationship between the response and the x-variables of the samples on which the calibration is computed (training, or calibration, set), but to apply it to future samples on which only the cheaper x-variables will be measured. In this case too, the model must be validated by using a set of samples different from those that have been used to compute the model (validation, or test, set). The responses of the objects of the test set will be computed by applying the model obtained by the training set and then compared with their "true" response. From these values the percent of variance explained in prediction and the Root Mean Square Error in Prediction (RMSEP) can be computed. Provided that the objects forming the two sets have been selected flawlessly, these values give the real performance of the model on new samples.

7. FEATURE SELECTION

Usually, not all the variables of a data set bring useful and non-redundant information. Therefore, a variable (or feature) selection can be highly beneficial, since by it the following results are obtained:

- removal of noise and improvement of the performance;
- reduction of the number of variables to be measured and simplification of the model.

The removal of noisy variables should always be looked for. Though **some methods can give good results also with a moderate amount of noise disturbing the information, it is clear that their performance will increase when this noise is removed. So, feature selection is now widely applied also for those techniques (PLS and PCR) that in the beginning were considered to be almost insensitive to noise.**

While the noise reduction is a common goal for any data set, the relevance of the reduction of the number of variables in the final model depends very much on the kind of data constituting the data set, and a very wide range of situations are possible. Let's consider the extreme conditions:

- each variable requires a separate analysis
- all the variables are obtained by the same analysis (e.g., chromatographic and spectroscopic data).

In the first case, each variable not selected means a reduction in terms of costs and/or analysis time. The variable selection should therefore always be made on a cost/benefit basis, looking for the subset of variables leading to the best compromise between performance of the model and cost of the analyses. This means that, in presence of groups of useful but highly correlated (and therefore redundant) variables, only one variable per group should be retained. With such data sets, it is also possible that a subset of variables giving a slightly worse result is preferred, if the reduction in performance is widely compensated by a reduction in costs or time.

In the second case, the number of retained variables has no effect on the analysis cost, and the presence of useful and correlated variables improves the stability of the model.

Intermediate cases can happen, in which "blocks" of variables are present. As an example, take the case of olive oil samples, on each of which the following analyses have been run: a titration for acidity, the analysis of peroxides, a UV spectroscopy for ΔK, a GC for sterols and another GC for fatty acids. In such a situation, it is not the final number of variables that counts, but the number of analyses one can save.

The only possible way to be sure that "the best" set of variables has been picked up is the "all-models" techniques, by which all the possible combinations are tested. Since, with k variables, the number of possible combinations is 2^k-1, it is easy to

understand that this approach cannot be used unless the number of variables is really very low (e.g., with 30 variables more than 10^9 combinations should be tested).

The simplest (but least effective) way of performing a feature selection is to operate on a "univariate" basis, by retaining those variables having the greatest discriminating power (in case of a classification) or the greatest correlation with the response (in case of a calibration). By doing that, each variable is taken into account by itself, without considering how its information "integrates" with the information brought by the other (selected or unselected) variables. As a result, if several highly correlated variables are "good", they are all selected, without taking into account that, owing to their correlation, the information is highly redundant and therefore at least some of them can be removed without any decrease in the performance. On the other side, those variables are not taken into account that, though not giving by themselves a significant information, become very important when their information is integrated with that of other variables.

An improvement is brought by the "sequential" approaches. They select the best variable and then the best pair formed by the first and second and so on in a forward or backward progression. A more sophisticated approach applies a look back from the progression to reassess previous selections. The problem with these approaches is that only a very small part of the experimental domain is explored and that the number of models to be tested becomes very high in case of highly dimensional data sets, such as spectral data sets. For instance, with 1000 wavelengths, 1000 models are needed for the first cycle (selection or removal of the first variable), 999 for the second cycle, 998 for the third cycle, and so on.

More "multivariate" methods of variable selection, especially suited for PLS applied to spectral data, are currently available. Among them, we can cite Interactive Variable Selection (Lindgren 1994), Uninformative Variable Elimination (Centner 1996), Iterative Predictor Weighting PLS (Forina 1999) **and Interval PLS (Nørgaard 2000).**

8. GENETIC ALGORITHMS

Genetic Algorithms are a general optimization technique, that has found good applicability in many fields, especially when the problem is so complex that it cannot be tackled with the "standard" techniques. In Chemometrics it has been applied especially in feature selection (Leardi 2000). GA try to simulate the evolution of a species according to the Darwinian theory. Each experimental condition (in this case, each model) is treated as an individual, whose "performance" (in the case of a feature selection for a calibration problem, it can be the explained variance) is treated as its "fitness". Through operators simulating the fights among individuals (the best ones have a greatest probability of mating and thus spreading their genome), the mating among individuals (with the consequent "birth" of "offspring" having a genome that is derived by both the parents) and the occurrence of mutations, the GA result in a pattern of search that, by mixing "logical" and "random" features, allows a much more complete search of complex experimental domains.

Genetic Algorithms have been proposed by Holland around 1960, but only since 1990, owing to the development of computer speed, it was possible to apply them to real problems with acceptable computing time. The basic idea is a computer simulation of what happens in nature, and the first problem concerns the coding of the information in such a way that it can be processed by a computer.

We can say that the fitness to the environment is a function of the genetic material of the individual, the same way as the result of an experiment is a function of the experimental conditions. Therefore, we can state that experimental conditions correspond to what in life is genetic material. We can also say that genetic material is defined by genes, the same way as an experimental condition is defined by the values of the variables relevant to that experiment. As a consequence, variables correspond to genes.

At a lower level, we know that the information contained in each gene is coded by a sequence of bases: since there are four different bases, each gene can be considered as a word of variable length, written by using a four-letter alphabet. The same way, the binary code can be used to transform the value of a variable into a word of variable length, written in bits (two-letter alphabet, 0 and 1).

Therefore, we have the following correspondences:

genetic material (chromosome) = experimental conditions

gene = variable

base = bit

As a consequence, each experimental condition can be coded by a sequence of 0's and 1's.

In the case of feature selection, for instance, the coding is very simple: each chromosome has as many genes as the number of variables, and each gene is made by a single bit, being 0 if the variable is not present in the model, and 1 if the variable is present.

According to the evolution theory, a species increases its fitness to the environment because, throughout very many generations, the genetic material of its individuals becomes better and better. This depends on the fact that the "bad" individuals do not survive and that the best individuals have a greater probability of passing their genetic material

to the following generation. Beyond this “logical” development, mutations allow to explore new “experimental conditions”; usually mutations generate bad results (e.g., severe pathologies), but it can happen that random changes of a base produce a better genetic material.

Several genetic algorithms have been developed. Though they can be very different, all of them have three fundamental steps: creation of an original population, reproductions, mutations. Since the detailed description of the algorithm would surely be beyond the scope of this paper, only a very concise overview of these steps will be given.

The first step is the creation of the original population. After having set the population size (usually between 20 and 500 chromosomes), for each bit of each gene of each individual a random number is drawn, determining whether that bit will be 0 or 1. At the end of this procedure, the sequence of bits of each chromosome will be decoded to the real variables and the associated response will be measured.

The pairs of chromosomes that will mate and that will originate the offspring that will form the next generation can be now selected. Also in this case, a drawing will take place, but it will be “biased” in such a way that the probability associated to each chromosome for being selected will be a function of its fitness.

From each pair two new chromosomes will originate, whose genetic material will be derived by the genetic material of the “parents”. For each gene, the genetic material of the two parents will be passed to the two offsprings, with a random drawing determining which offspring will “inherit” the gene of which “parent”. While this step simply redistributes to the new population already existing genes, the mutation operator makes possible to flip some bits from 0 to 1 (and vice versa), by that allowing to test variable values that never happened before or to get out from deadlock situations.

After reproductions and mutations, the fitness of the chromosomes of the new generation is computed and the process continues with the formation of new pairs, until a stop criterion is met.

9. ARTIFICIAL NEURAL NETWORKS

Artificial Neural Networks (ANN) try to mimic the behavior of the nervous system to solve practical computational problems. As in life, the structural unit of ANN is the neuron. The input signals are passed to the neuron body, where they are weighted and summed, then they are transformed, by passing through the transfer function into the output of the neuron. The propagation of the signal is determined by the connections between the neurons and by their associated weights. The appropriate setting of the weights is essential for the proper functioning of the network. Finding the proper weight setting is achieved in the training phase.

The neurons are usually organized into three different layers: the input layer contains as many neurons as input variables, the hidden layer contains a variable number of neurons and the output layer contains as many neurons as output variables. All units from one layer are connected to all units of the following layer. The network receives the input signals through the input layer. Information is passed to the hidden layer and finally to the output layer that produces the response.

In the training phase the weights are initially set randomly. A training set with a number of objects with known output pattern is presented to the network, and during the training session the weights are progressively adapted according to the learning rule. The weight updates are based on the difference between the actual and the desired output of the network, and the weight updating can be done after each training example that is offered to the network or after all training examples have been seen once. The process is then repeated for all training examples (at each iteration their sequence is randomized, to avoid bias) until a stop criterion is reached. Usually several iterations (50 to 5000) are required.

The main limitation in applying ANN to real data sets is strictly connected with the architecture of the ANN itself. For an ANN having i input neurons, h hidden neurons and o output neurons the number of weights to be optimized is equal to $hi + ho$. Having 30 input variables (30 input neurons), 3 hidden neurons and just one output neuron (e.g., if we want to classify the geographical origin of olive oils from their chemical composition), 93 weights must be optimized. If we apply here too what has been previously said about the minimum objects/variables ratio required not to have an overfitting model, we can see that almost 500 samples (anyway, no less than 300 samples) are required.

Apart from these objects, a different set has to be prepared for monitoring the predictive ability, in order to avoid the phenomenon of overtraining. This happens when too many iterations are used for training the network, since after a certain number of iterations the noise present in the training set is also modeled by the network; as a consequence, the network starts loosing its ability to predict.

A third set of objects has also to be taken into account for testing the predictive ability of the network on an independent set. It has in fact to be remembered that a prediction is correct only if the samples to be predicted have never been seen during the different steps leading to the final model.

Therefore, it is very easy to realize that ANN, though a very powerful technique, can be very

seldom applied in a correct way. Unfortunately, many people are not at all aware of this strong limitation, and it can happen to read about ANN that have been trained with a number of samples much smaller than what should be required. As a consequence, despite a very good performance on the training set (due to overfitting), these ANN will show very poor results when applied to external data sets.

REFERENCES

Centner, V., Massart, D.L., de Noord, O.E., de Jong, S., Vandeginste, B.M., Sterna, C. (1996). Elimination of uninformative variables for multivariate calibration. *Anal. Chem.*, **68**, 3851-3858.

Forina, M., Casolino, C., Pizarro Millán, C. (1999). Iterative predictor weighting (IPW) PLS: A technique for the elimination of useless predictors in regression problems. *J. Chemometrics*, **13**, 165-184.

Gueguen, J., Barbot, J. (1988). Quantitative and qualitative variability of pea (*Pisum sativum L.*) protein composition. *J. Sci. Food Agric.*, **42**, 209-224.

Leardi, R. (2000). Application of genetic algorithm-PLS for feature selection in spectral data sets. *Journal of Chemometrics*, **14**, 643-655.

Lindgren, F., Geladi, P., Rännar, S., Wold, S. (1994). Interactive Variable Selection (IVS) for PLS. 1. Theory and algorithms. *J. Chemom.*, **8**, 349-363.

Nørgaard, L., Saudland, A., Wagner, J., Nielsen, J.P., Munck, L., Engelsen, S.B. (2000). Interval partial least-squares regression (iPLS): A comparative chemometric study with an example from near-infrared spectroscopy - *Applied Spectroscopy*, **54**, 413-419.

COMPLEMENTARY BIBLIOGRAPHY SUGGESTED

Beebe, K.R., Pell, R.J., Seasholtz, M.B. (1998). *Chemometrics: A Practical Guide*, Wiley & Sons, NewYork.

Brereton, R.G. (1994). Multivariate Pattern Recognition in Chemometrics. In: *Data Handling in Science and Technology*, vol 9. Elsevier, Amsterdam.

Manly, B.F.J. (1986). *Multivariate Statistical Methods. A Primer*. Chapman and Hall, London.

Martens, H., Naes, T. (1991). Multivariate Calibration. Wiley & Sons, NewYork.

Massart, D.L., Vandeginste, B.G.M., Deming, S.N., Michotte, Y., Kaufman, L. (1990). Chemometrics: A Textbook. In: *Data Handling in Science and Technology, vol 12*. Elsevier, Amsterdam.

Massart, D.L., Vandeginste, B.G.M., Buydens, L.M.C., De Jong, S., Lewi, P.J., Smeyers-Verbeke, J. (1997). Handbook of Chemometrics and Qualimetrics. Part A. In: *Data Handling in Science and Technology, vol.20A*, Elsevier, Amsterdam.

Massart, D.L., Vandeginste, B.G.M., Buydens, L.M.C., De Jong, S., Lewi, P.J., Smeyers-Verbeke, J. (1998). Handbook of Chemometrics and Qualimetrics. Part B. In: *Data Handling in Science and Technology, vol.20A;* Elsevier, Amsterdam.

Meloun, M., Militky, J., Forina, M. (1992). Chemometrics for Analytical Chemistry. In: *PC- aided Statistical Data Analysis - Volumen 1.* Ellis Horwood, Chichester.

Meloun, M., Militky, J., Forina, M. (1994). Chemometrics for Analytical Chemistry. In: *PC-Aided Regression and Related Methods - Volumen 2*. Ellis Horwood, Hemel Hempstead (UK).

Sharaf, M.A., Illman, D.L., Kowalski, B.R. (1986). Chemometrics**.** In: *Chemical Analysis, a Series of Monographs on Analytical Chemistry and its Applications, vol. 82***.** Wiley & Sons, New York,. In, P.J. Elving and J.D. Winefordner (Eds.).

WEB SITES

http://ull.chemistry.uakron.edu/chemometrics/
http://www.acc.umu.se/~tnkjtg/chemometrics/index.html
http://www.statsoft.com/textbook/stathome.html