101 Applications of Multivariate Statistical Procedures: PRINCOMP, CLUSTER, DISCRIM in SAS® 9.2

Keh-Dong Shiang, City of Hope National Medical Center, Duarte, CA

ABSTRACT

This paper briefly introduces multivariate statistical analysis techniques, including procedures PROC PRINCOMP, PROC CLUSTER, and PROC DISCRIM in SAS® version 9.2, which enable users to investigate relationships among multiple variables in the data. Two illustrative sports examples will be offered and demonstrated. Please note that this work is intended for the most basic, beginner-intermediate level audience, who has no familiarity or experience with cluster, discriminant, and principal component analyses.

INTRODUCTION

Multivariate statistical analyses are designed to handle many independent variables (IVs) and several dependent variables (DVs) all interrelated with one another to a certain degree. The multivariate analysis procedures are used to investigate relationships among variables without designating some as independent and others as dependent. Broadly speaking, Multivariate analysis techniques include factor analysis, cluster analysis (CA), discriminant analysis (DA), principal component analysis (PCA), and multiple regression analysis. In general, users would ask questions:

- What kind of correlations exists between the variables that characterize our studies cases?
- Which of our studied cases have similar multivariate profiles and can therefore be grouped together?

Clustering systems assign objects into groups (or called clusters) so that objects from the same cluster are more similar to each other than objects from different clusters. Cluster analysis contains many diverse methodologies for exploring structure within complex data contents, which is a technique for classifying data cases into distinct groups on the basis of similarity across variables. Usually, in biomedicine, this means that we are interested in clustering groups of patients or genes. So, in a sense it's the opposite of factor analysis: instead of forming groups of variables based on a number of patients' responses to those variables, we instead group patients based on their responses to several variables.

The objective in cluster analysis is to group "similar" observations together when the underlying structure is unknown. More clearly, it groups the observed data cases into clusters such that elements within a cluster have a high degree of "natural correlation" among themselves while the clusters are relatively distinct from one another. As a result,

1. All elements in each group are homogeneous with respect to certain characteristics; that is, observations in each group are similar to each other.
2. Each group after clustering should be different from other groups with respect to the characteristics; that is, observations of one group should be sharply distinguished from the observation of other groups.

Simply stated, intra-cluster distances are minimized, and the inter-cluster distances are maximized.

DA method is used to find a set of linear combinations of the variables, whose values are as close as possible within groups and as far apart as possible between groups. It tests for significant differences among groups and can also be used to predict group membership as well. Briefly, discriminant function attempts to establish whether a set of variables can be used to distinguish between among groups. A well-known method, Fisher's Linear Discriminant Analysis (LDA), is usually applied to compute a linear predictor from several sets of normally distributed data to allow for classification of new observations.

It is commonly known that PCA is a multivariate procedure, which rotates the data that maximum variabilities are projected onto the axes. Essentially, a set of correlated variables is transformed into a set of uncorrelated variables, which are ordered by reducing variability. The uncorrelated variables are linear combinations of the original variables, and the last of these variables can be removed with minimum loss of real data. In fact, it is a data reduction technique designed to determine the modes of variation of a multivariate random variable in high dimensions.

It is widely thought that searching the complex data for a structure of natural grouping is a critical exploratory technique. The most important techniques for data classification are CA and DA. This is the reason why we introduced these two methods. A simple introduction of the basic theory in Classification and DA, as well as PCA will be given in the following.

Simple Introduction to Cluster Analysis (CA)

"Cluster Analysis" (CA) is generally performed through a variety of methods, which use some measurements of distance between data points as a basis for creating groups. Typically, this distance is the standard Euclidian distance, i.e. a straight line in two geometric dimensions. Data points with the smallest distances between them are grouped together. If the analysis works, distinct groups or clusters will be distinctive.

CA is a classification method that is used to arrange a set of studied cases into clusters. It is a class of statistical techniques that can be applied to raw data that exhibit "natural" groupings. CA sorts through the data and groups them into clusters. By definition, cluster is a group of relatively homogeneous cases or observations. Objects in a cluster are similar to each other. They are also dissimilar to objects outside the cluster, particularly objects in other clusters.

Mathematically noted, associated with each object is a set of *M* measurements which form the feature vector,

$$X = (X_1, X_2, X_3, \cdots, X_M)$$

The case vector X_i belongs to a feature space *X*. The objective is to identify groups, or clusters, of similar objects on the basis of a set of case vectors,

$$X_1 = (x_1, x_2, x_3, \cdots, x_N), \ldots, X_M = (x_1, x_2, x_3, \cdots, x_N)$$

As a practical example in cancer clinical study, an accurate sub-categorization of tumor types through gene expression profiling requires analytical technique like clustering that estimates the number of categories or clusters. Specifically, clustering is the assignment of a set of observations into subsets (or called groups or clusters) so that observations in the same cluster are similar in some sense. Clustering is also a method of unsupervised learning, and a common technique for statistical data analysis used in many fields, including machine learning in engineering and science, data mining, pattern recognition, image analysis, biology, medicine, agriculture, social sciences, sports, and bioinformatics. In biomedical field, DNA microarray technology has been extensively applied to monitor the expression levels of thousands of genes during important biological processes and across collections of related but limited samples (or phenotypes). Elucidating the patterns hidden in gene expression data does offer a tremendous opportunity for an enhanced understanding of functional genomics.

Mathematically, the Euclidean geometric distance between data points **p** and **q** is the length of the line segment $\overline{pq}$. In Cartesian coordinates, if **p** = (p_1, p_2, p_3, …, p_n) and **q** = (q_1, q_2, q_3, …, q_n) are two data points in Euclidean space, then the distance from **p** to **q** is given by:

$$\mathrm{d(p, q)} = \sqrt{(\mathrm{p}_1 - q_1)^2 + (p_2 - q_2)^2 + (p_3 - q_3)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

which is the sum of the squared differences between values for two cases (variables) on all variables (cases). Here, a simple concept of variables' Cosine vectors is introduced. The cosine of the angle between two variable vectors is identical to their correlation coefficient.

$$Similarity(p,q) = \frac{S(p,q)}{S(p^2) \cdot S(q^2)}$$

The dot product (or inner product) of two vectors **p** and **q** is denoted:

$$\vec{p} \cdot \vec{q} = | \vec{p} \, || \, \vec{q} \, | \cos\theta$$

where θ is the measure of the angle between **p** and **q**. Thus, the value of angle (**p**, **q**) is defined as follows:

$$angle(\vec{p},\vec{q}) = \cos^{-1}(\vec{p},\vec{q}) = \cos^{-1}(\frac{\vec{p} \cdot \vec{q}}{\sqrt{(\vec{p} \cdot \vec{p}) \cdot (\vec{q} \cdot \vec{q})}})$$

Distances between points can be calculated by using an extension of Pythagorus, which are also defined as Euclidean geometric distances. It is likely that the patterns between the above two ratios seem to be similar. In other words, the angle is given by arc-cosine of the dot product of the two normalized vectors and the similarity measure can be interpreted as the inverse of cosine function if the data point objects are represented as length-normalized vectors.

It is obvious that these measurements of 'dissimilarity' can be extended to more than two dimensions without any difficulty. Fortunately, calculation tools and methods have been provided in SAS cluster procedure. The SAS METHOD options control the clustering method used for the clustering. There are 11 methods that can be specified. Method examples are AVERAGE, CENTROID, COMPLETE, DENSITY, EML, MEDIAN, SIMPLE and WARD. The method selected in our example is the AVERAGE, which bases clustering decisions on the average distance (linkage) between points or clusters. Some other possibilities include CENTROID which uses the distance between the geometric centers of the clusters, MEDIAN which is similar to average, but based on median values, and SIMPLE which uses a nearest neighbor approach. The computed clusters will be saved in a dataset called TREE for plotting purposes.

Basic Theory of Classification and Discriminant Analysis (DA)

Classification aims to assign observations (also called features, attributes, variables or measurements) into classes. To assist the process of classification, a rule (or classifier) can be established to assign a new observation into a class. In other words, previously known information is used to define the classifier, and classification deals with assigning a new data point to a class separated by a boundary. Linear classification provides a mathematical formula to predict a binary result. This result is typically represented as a true/false (positive/negative, 1/0) Boolean value. To make this prediction more clearly, a linear formula is proposed to fit the given data. The linear form is computed over the inputs, and its result is compared against a basis constant. Depending on the result of the comparison, a either true or false value will be determined. In terms of equation, this can be expressed as the discriminator:

$$c_1 x_1 + c_2 x_2 + c_3 x_3 + \cdots + c_p x_p > x_0$$

where c_1, c_2, ... are the variables corresponding to one observation and x_1, x_2, ... together with x_0 are the solution vector plus basis constant.

Practically, observations often come in natural "groups" or "classes". For example in a clinical study, a set of symptoms (observations) is associated to a disease (group or class), and another set of symptoms is related to another disease.

Two-Group Discrimination Illustration

The two-group discriminant problem deals with discrimination between two predefined groups and is the fundamental problem addressed by discriminant analysis. A two-group discriminant problem assumes that there are two well-defined populations, group G_1 and group G_2. The focus of discriminant analysis is the determination of a numerical rule or discriminant function that can be used to distinguish between two populations using the measured *j* variables or attributes. A linear discriminant function can be expressed by

$$y_i = c_1 x_{i1} + c_2 x_{i2} + c_3 x_{i3} + \cdots + c_j x_{ij}$$

where c_j is the weight assigned to variable *j*, x_{ij} is the value of the *j*th variable for the *i*th individual, and y_i is the discriminate value for the *i*th individual.

The goal of any discriminant analysis model is to estimate parameters c_j and boundary (or cut-off) value as well as to minimize the number of misclassifications. In a discriminant analysis model, the weighting parameters c_j and boundary (or cut-off) value are determined from a set of observations for which their group membership is known.

Fisher linear Discriminant Function

Assuming there are two classes, A and B, *p* variables, and *m* and *n* observation for each class.

Class A	Class B
$(x_{11}^1, x_{12}^1, x_{13}^1, \cdots, x_{1p}^1)$	$(x_{11}^2, x_{12}^2, x_{13}^2, \cdots, x_{1p}^2)$
$(x_{21}^1, x_{22}^1, x_{23}^1, \cdots, x_{2p}^1)$	$(x_{21}^2, x_{22}^2, x_{23}^2, \cdots, x_{2p}^2)$
$(x_{31}^1, x_{32}^1, x_{33}^1, \cdots, x_{3p}^1)$	$(x_{31}^2, x_{32}^2, x_{33}^2, \cdots, x_{3p}^2)$
$(\cdots\cdots\cdots\cdots)$	$(\cdots\cdots\cdots\cdots)$
$(x_{m1}^1, x_{m2}^1, x_{m3}^1, \cdots, x_{mp}^1)$	$(x_{n1}^2, x_{n2}^2, x_{n3}^2, \cdots, x_{np}^2)$

Supposing a discriminant function is defined as

$$y_i^{class} = c_1 x_{i1}^{class} + c_2 x_{i2}^{class} + c_3 x_{i3}^{class} + \cdots + c_p x_{ip}^{class}$$

where the weighting parameter vector $\vec{c} = (c_1, c_2, c_3, \cdots, c_p)$. Then, the individual functions corresponding to group 1 (or class A) and group 2 (or class B) are written as

$y_1^1 = c_1 x_{11}^1 + c_2 x_{12}^1 + c_3 x_{13}^1 + \cdots + c_p x_{1p}^1$	$y_1^2 = c_1 x_{11}^2 + c_2 x_{12}^2 + c_3 x_{13}^2 + \cdots + c_p x_{1p}^2$
$y_2^1 = c_1 x_{21}^1 + c_2 x_{22}^1 + c_3 x_{23}^1 + \cdots + c_p x_{2p}^1$	$y_2^2 = c_1 x_{21}^2 + c_2 x_{22}^2 + c_3 x_{23}^2 + \cdots + c_p x_{2p}^2$
$y_3^1 = c_1 x_{31}^1 + c_2 x_{32}^1 + c_3 x_{33}^1 + \cdots + c_p x_{3p}^1$	$y_3^2 = c_1 x_{31}^2 + c_2 x_{32}^2 + c_3 x_{33}^2 + \cdots + c_p x_{3p}^2$
……	……
$y_m^1 = c_1 x_{m1}^1 + c_2 x_{m2}^1 + c_3 x_{m3}^1 + \cdots + c_p x_{mp}^1$	$y_n^2 = c_1 x_{n1}^2 + c_2 x_{n2}^2 + c_3 x_{n3}^2 + \cdots + c_p x_{np}^2$

The mean discriminant values are simply expressed as

$$\bar{y}^1 = \frac{1}{m} \cdot \sum_{i=1}^{m} y_i^1 \qquad \bar{y}^2 = \frac{1}{n} \cdot \sum_{i=1}^{n} y_i^2$$

To clearly distinguish the difference between group 1 and group 2, these two groups' observations, which represented by two distinct discriminant values, have a large squared between-group difference and a small sum of squared within-group differences:

The larger the $(\bar{y}^1 - \bar{y}^2)^2$ value and the smaller the $\sum_{i=1}^{m}(y_i^1 - \bar{y}^1)^2 + \sum_{i=1}^{n}(y_i^2 - \bar{y}^2)^2$ value, the better the discriminant result will be yielded. Thus, the best predicted discriminant result will depend upon the maximized the *L* ratio function:

$$L(c_1, c_2, c_3, \cdots) = \frac{(\bar{y}^1 - \bar{y}^2)^2}{\sum_{i=1}^{m}(y_i^1 - \bar{y}^1)^2 + \sum_{i=1}^{n}(y_i^2 - \bar{y}^2)^2}$$

In calculus, the partial derivatives with respect to each weight parameter will be used to locate maximum L ratio.

$$\frac{\partial L(c_1, c_2, c_3, \cdots)}{\partial c_i} = 0 \qquad \text{where } i = 1,2,3,\cdots, p$$

As the entire derivation is too complex and tedious, a partial formulation is briefly introduced in the following. Assume that ywo groups' data are represented as matrix form:

$$W^1 = \begin{bmatrix} x_{11}^1 & x_{12}^1 & x_{13}^1 & \cdots & x_{1p}^1 \\ x_{21}^1 & x_{22}^1 & x_{23}^1 & \cdots & x_{2p}^1 \\ x_{31}^1 & x_{32}^1 & x_{33}^1 & \cdots & x_{3p}^1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{m1}^1 & x_{m2}^1 & x_{m3}^1 & \cdots & x_{mp}^1 \end{bmatrix} \qquad W^2 = \begin{bmatrix} x_{11}^2 & x_{12}^2 & x_{13}^2 & \cdots & x_{1p}^2 \\ x_{21}^2 & x_{22}^2 & x_{23}^2 & \cdots & x_{2p}^2 \\ x_{31}^2 & x_{32}^2 & x_{33}^2 & \cdots & x_{3p}^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{n1}^2 & x_{n2}^2 & x_{n3}^2 & \cdots & x_{np}^2 \end{bmatrix}$$

Column means for group 1 and group 2 are written as

$$\bar{x}_j^1 = \frac{1}{m} \cdot \sum_{i=1}^{m} x_{ij}^1 \qquad \bar{x}_j^2 = \frac{1}{n} \cdot \sum_{i=1}^{n} x_{ij}^2$$

where $j = 1,2,3,\cdots, p$ Then, the new transformed matrix A and matrix B are expressed as follows.

$$A = W^1 - \text{Matrix 1 consisting of Column Means} = \begin{bmatrix} x_{11}^1 - \bar{x}_1^1 & x_{12}^1 - \bar{x}_2^1 & x_{13}^1 - \bar{x}_3^1 & \cdots & x_{1p}^1 - \bar{x}_p^1 \\ x_{21}^1 - \bar{x}_1^1 & x_{22}^1 - \bar{x}_2^1 & x_{23}^1 - \bar{x}_3^1 & \cdots & x_{2p}^1 - \bar{x}_p^1 \\ x_{31}^1 - \bar{x}_1^1 & x_{32}^1 - \bar{x}_2^1 & x_{33}^1 - \bar{x}_3^1 & \cdots & x_{3p}^1 - \bar{x}_p^1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{m1}^1 - \bar{x}_1^1 & x_{m2}^1 - \bar{x}_2^1 & x_{m3}^1 - \bar{x}_3^1 & \cdots & x_{mp}^1 - \bar{x}_p^1 \end{bmatrix}$$

$$B = W^2 - \text{Matrix 2 consisting of Column Means} = \begin{bmatrix} x_{11}^2 - \bar{x}_1^2 & x_{12}^2 - \bar{x}_2^2 & x_{13}^2 - \bar{x}_3^2 & \cdots & x_{1p}^2 - \bar{x}_p^2 \\ x_{21}^2 - \bar{x}_1^2 & x_{22}^2 - \bar{x}_2^2 & x_{23}^2 - \bar{x}_3^2 & \cdots & x_{2p}^2 - \bar{x}_p^2 \\ x_{31}^2 - \bar{x}_1^2 & x_{32}^2 - \bar{x}_2^2 & x_{33}^2 - \bar{x}_3^2 & \cdots & x_{3p}^2 - \bar{x}_p^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{m1}^2 - \bar{x}_1^2 & x_{m2}^2 - \bar{x}_2^2 & x_{m3}^2 - \bar{x}_3^2 & \cdots & x_{mp}^2 - \bar{x}_p^2 \end{bmatrix}$$

A new *S* matrix is then formed through simple matrix algebra:

$$S_1 = A'A \qquad S_2 = B'B \qquad S = S_1 + S_2$$

The best estimate of the discriminant function's weighting parameters $(c_1, c_2, c_3, \cdots, c_p)$ can be obtained by solving the system equations below:

$$S \cdot \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ .. \\ c_p \end{pmatrix} = \begin{bmatrix} \bar{x}_1^1 - \bar{x}_1^2 \\ \bar{x}_2^1 - \bar{x}_2^2 \\ \bar{x}_3^1 - \bar{x}_3^2 \\ \cdots \\ \bar{x}_p^1 - \bar{x}_p^2 \end{bmatrix} \quad \Rightarrow \quad \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ .. \\ c_p \end{pmatrix} = S^{-1} \cdot \begin{bmatrix} \bar{x}_1^1 - \bar{x}_1^2 \\ \bar{x}_2^1 - \bar{x}_2^2 \\ \bar{x}_3^1 - \bar{x}_3^2 \\ \cdots \\ \bar{x}_p^1 - \bar{x}_p^2 \end{bmatrix}$$

By substituting this set of parameter, the discriminant function is yielded:

$$y = c_1 x_1 + c_2 x_2 + c_3 x_3 + \cdots + c_p x_p$$

In SAS practical running, if a discriminant analysis will be performed, it is suggested that trying to run PROC STEPDISC before executing PROC DISCRIM. The reason of doing so is to find out which variables yield largest difference among groups and then those insignificant variables can be ignored in discriminant analysis.

Basics of Principal Component Analysis (PCA)

Principle Components Analysis (PCA) is a data reduction technique designed to determine the modes of variation of a multivariate random variable in high dimensions. Briefly, it may be considered as a tool for discovering structures in multivariate data, in particular for the purpose of reducing the dimensionality. PCA takes the cloud of data points, rotates and projects it onto a space of lower dimension, selecting the directions in the data space with maximum variability, or equivalently high information. PCA method is used to find the linear combinations of vectors that maximize the variation. Obviously, the single linear combination that maximizes the variance will be the first eigenvector of the covariance matrix.

The starting point for PCA is a set of observations in a multi-dimensional space, which is defined by our measurement channels. For example, if we are measuring m variables, the space will be m-dimensional. The purpose of PCA is to introduce a new set of m orthogonal axes in such a way that our original data will demonstrate the highest variance on the principal axis #1, the second highest variance on principal axis #2, and so on, with the least variance being shown on principal aces #m. These axes are referred to as principal component axes, or simply as principal components. It can also be viewed as a rotation of the existing axes to new positions in the space defined by the original variables. In this new rotation, there will be no correlation between the new variables defined by the rotation. The first new variable contains the maximum amount of variation; the second new variable contains the maximum amount of variation unexplained by the first and orthogonal to the first new variable.

Each principal component is a linear combination of the original variables. As the principal components are orthogonal to each other, they do not contain redundant information. Theoretically, the number of principal components is equal to the number of the original variables. However, ideally the variance of the original data can be explained by the first few principal components and the rest can be ignored. If this is the case, using only few principal components will reduce a large multi-dimensional set of data into data along a few coordinates. It should be noted that principal components analysis would only help us if there were dependencies among the measured variables. If all variables were independent from each other, then we would not gain any reduction of complexity by doing a principal component analysis.

Please note that the eigenvalues are strongly connected with the principal components analysis of the predictor variables. The principal components of the predictors are a set of new variables that are linear combinations of the original predictors. These components have two important properties: they are not correlated with each other; and each has maximum variance. The principal components provide idealized predictor variables that still retain all of the same information as the original variables. The variances of these new variables are called eigenvalues. The larger the eigenvalue, the more important is the associated principal component in representing the information in the predictors is. As an eigenvalue approaches zero, the presence of a near collinearity among the original predictors is indicated.

Suppose that we observe a n-dimensional vector of random variables $Z = (Z_1, Z_2, Z_3, \cdots, Z_n)'$ with $Z_i = (Y_i, X_i)$, where Y_i is 1×1 vector and X_i is $n \times 1$ vector. Here, $Z_1, Z_2, Z_3, \cdots, Z_n$ are iid N(0,1) random variables. Then, the density function of Z (Y_i given X_i) can be written as

$$f(Z) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}} \exp[-\frac{1}{2} \cdot z_i^2] = (\frac{1}{2\pi})^{n/2} \exp[-\frac{1}{2} \cdot \sum_{i=1}^{n} z_i^2] = (\frac{1}{2\pi})^{n/2} \exp[-\frac{1}{2} \cdot Z'Z]$$

Because Z_i have mean 0 and variance 1, and are uncorrelated, the mean and covariance matrix of Z are

$$E[Z] = 0 \qquad \text{and} \qquad COV[Z\} = I_n$$

where I_n denotes the identity matrix of order n.

For the general case, suppose $\sum$ is an $n \times n$, symmetric and positive semi-definite matrix. Then, from linear algebra theory, we can always decompose $\sum$ as

$$\sum = \Gamma' \Lambda \Gamma$$

where Λ is the diagonal matrix $\Lambda = diag(\lambda_1, \lambda_2, \lambda_3, \cdots, \lambda_n)$, $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \cdots \geq \lambda_n \geq 0$ are the eigenvalues of $\sum$, and the columns of Γ', $v_1, v_2, v_3, \cdots, v_n$, are the corresponding eigenvectors. This decomposition is called the spectral decomposition of $\sum$. The matrix Γ is orthogonal, i.e., $\Gamma^{-1} = \Gamma'$, and $\Gamma\Gamma' = I$. We can write the spectral decomposition in another way, as

$$\Sigma = \Gamma' \Lambda \Gamma = \sum_{i=1}^{n} \lambda_i v_i v_i^{'}$$

Formally, PCA is a well-established method of reducing the dimensionality in multi-channel data, which is equivalent to the Singular Value Decomposition (SVD) of the data matrix as

$$Y = USV^t$$

with the constraints that $V^tV = U^tU = I$, where *I* is the identity matrix. The orthogonal matrix *V* is the matrix of loadings or projections of the original variables onto the components, where one row stands for one

of the original variables and one column for one of the new factors. S is a diagonal matrix with the so-called singular values on the diagonal. In general, the subjects' score matrix is obtained as US. Let $\{r_i\}$, $\{c_j\}$ be the row and column vectors of the matrix X, respectively. Singular columns $\{u_i\}$ form an orthonormal basis for the column vector space, and singular rows $\{v_i\}$ form an orthonormal basis for the row vector space. The first k SVD components provide the best rank k approximation of the data matrix X, where $k \le r = rank(X)$. If X is column centered at 0 (i.e., column means are zeros), PCA is the factorization of X^tX. The major difference between PCA and SVD: PCA is the factorization of X^tX (covariance matrix), and SVD is the factorization of X (original data matrix).

Exploratory Data Analysis in SAS

Example 1: Fitness data analysis using PROC CLUSTER and PROC DISCRIM

The dataset provides an example consisting of one Fitness category variable and two quantitative measurement variables.

```
PROC PRINT DATA=Fitness;
VAR FitnessAbility EnergyConsumption SugarConsumption;
RUN;

PROC CLUSTER DATA=Fitness METHOD=AVERAGE OUTTREE=TreeData;
VAR EnergyConsumption SugarConsumption;
PROC TREE DATA=TreeData;
RUN;
```

The Fitness dataset is shown in the output below:

Obs	Fitness Ability	Energy Consumption	Sugar Consumption
1	Squat	26.38	60.29
2	Somersault	25.82	55.43
3	Lift weight	22.73	72.94
4	Pull up	22.52	54.58
5	Curl	18.46	60.12
6	Bench press	16.78	72.82
7	Leg extension	20.12	53.25
8	Squat jump	24.51	78.35
9	Calf raise	17.85	48.56
10	Sit up	19.37	71.74

The SAS output for PROC CLUSTER is displayed in the following. As you can clearly see, the Average Linkage method is specified. This first section lists the eigenvalues of the covariance matrix.

The CLUSTER Procedure
Average Linkage Cluster Analysis

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	106.057425	94.376384	0.9008	0.9008
2	11.681041		0.0992	1.0000

Root-Mean-Square Total-Sample Standard Deviation 7.672629

Root-Mean-Square Distance Between Observations 15.34526

The second section gives us the clustering “history”, which startis with the smallest distance (normalized RMS distance). The first line shows a cluster, #9, was created and formed using observations 4 and 7; therefore the frequency was 2. Its calculated distance was 0.1788, which was the smallest. Similar clusters #8 and #7 were generated from single observations 6 and 10, as well as 1 and 2. Then, at cluster #6, observation 3 was added to cluster #8 (from observations 6 and 10), so its frequency count was 3. This process continues until all observations are included, and frequency count is reaching the total observation number. Although this clustering process might be interesting, it could be intuitively difficult for users to follow on this section when viewing it at the first time.

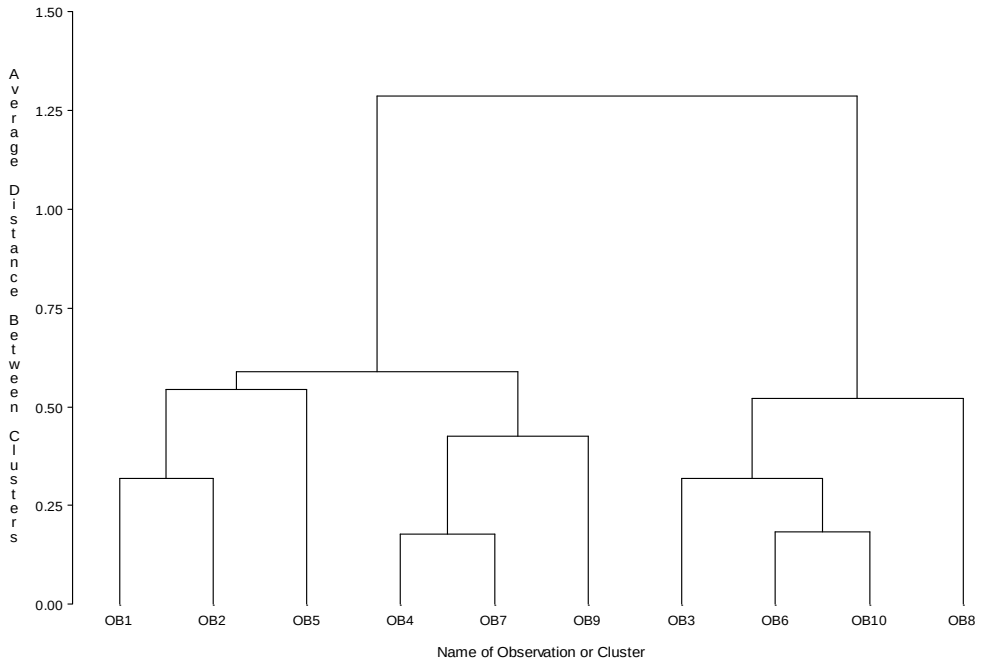
Cluster History

NCL	--Clusters Joined---		FREQ	Norm RMS Dist	Time
9	OB4	OB7	2	0.1788	
8	OB6	OB10	2	0.1829	
7	OB1	OB2	2	0.3188	
6	OB3	CL8	3	0.3197	
5	CL9	OB9	3	0.4253	
4	CL6	OB8	4	0.5225	
3	CL7	OB5	3	0.5431	
2	CL3	CL5	6	0.5898	
1	CL2	CL4	10	1.2866	

However, for easy visualization and understanding, clustering history is often transformed to a graphical plot, referred to as tree diagram or dendogram. In SAS, there is a procedure to generate such plot named PROC TREE. This procedure uses the output dataset from PROC CLUSTER (with optional parameter defined as `OUTTREE=TreeData`). The code is simply written:

```
PROC TREE DATA=TreeData;
```

PROC TREE has a number of options and statements available to fine tune the tree plot by modifying its labeling and shape, which is left to the interested users to read. The default tree plot is given below:



There are two large clustering groups in this tree diagram. The group #1 one the left has 6 observation (obs1, obs2, obs5, obs4, obs7, and obs9), and the group #2 on the right includes 4 observations (obs3, obs6, obs10, and obs8). Within these two large groups (clusters), there are several subgroups (sub-clusters) corresponding to the next levels' classifications, and these subgroups can even be further broken down into the deeper (detailed) sub-subgroups.

In clustering analysis, the goal is to use the raw data to define unknown groups. In contrast, discriminant analysis will be applied to classify and identify if the predefined grouping has been correctly describing each group or not. Here, in order to distinguish the difference between machine learning modules, two different data learning processes are introduced below:

Supervised versus unsupervised learning

Assign studied cases or objects to classes on the basis of measurements made on these cases.

Unsupervised learning
The classes are unknown, which need to be "discovered" from the data. For instances, cluster analysis; class discovery; unsupervised machine learning; unsupervised pattern recognition.

Supervised learning
The classes are predefined, and the task is to verify if the classification from a set of labeled objects is appropriate or not. This classified information is then used to identify future observations. For examples, classification; discriminant analysis; class prediction; supervised machine learning; supervised pattern recognition.

The demonstrated data is displayed through SAS output:

```
PROC PRINT DATA=Fitness;
VAR FitnessAbility EnergyConsumption SugarConsumption ClusterGrouping;
RUN;
```

The SAS code to perform this analysis is listed below:

```
PROC DISCRIM ANOVA CANONICAL DATA=Fitness POOL=YES;
CLASS ClusterGrouping;
VAR EnergyConsumption SugarConsumption;
PRIORS EQUAL;
RUN;
```

The CLASS statement lists the variable that represents the known groups or classes, and the VAR statement specifies the quantitative variables to be included in the analysis. The PRIORS statement is applied to specify the anticipated prior probabilities that observation belong to each group. "EQUAL" prior probabilities provide no preference for any class, and "PROPORTIONAL" priors use probabilities equal to the proportion of observations in each class of the dataset (i.e., proportional to the sample sizes in each group). If there is no information or previous knowledge about the actual data distribution among groups, the EQUAL option is recommended and most appropriate to use. Thus in our case, the prior Probabilities are defined as "EQUAL", which means the probabilities are equal for each class, i.e. probability = 0.5 (same as "PRIORS 0.5 0.5"). However, this default prior probability of classification can be changed through the PRIORS statement whenever you like.

The sample study's data are listed below:

Obs	Fitness Ability	Energy Consumption	Sugar Consumption	Cluster Grouping
1	Squat	26.38	60.29	1
2	Somersault	25.82	55.43	1
3	Lift weight	22.73	72.94	2
4	Pull up	22.52	54.58	1
5	Curl	18.46	60.12	1
6	Bench press	16.78	72.82	2
7	Leg extension	20.12	53.25	1
8	Squat jump	24.51	78.35	2
9	Calf raise	17.85	48.56	1
10	Sit up	19.37	71.74	2

The first section displays summary information on the number of observations, variables, and classes.

The DISCRIM Procedure

Total Sample Size	10	DF Total	9
Variables	2	DF Within Classes	8
Classes	2	DF Between Classes	1

Number of Observations Read	10
Number of Observations Used	10

The grouping information, frequency count for each group, proportion and prior probability of each class are also given.

Class Level Information

Cluster Grouping	Variable Name	Frequency	Weight	Proportion	Prior Probability
1	_1	6	6.0000	0.600000	0.500000
2	_2	4	4.0000	0.400000	0.500000

Pooled Covariance Matrix Information

Covariance Matrix Rank	Natural Log of the Determinant of the Covariance Matrix
2	4.98706

Pairwise Generalized Squared Distances Between Groups

$$D^2(i|j) = (\bar{X}_i - \bar{X}_j)' \, COV^{-1} \, (\bar{X}_i - \bar{X}_j)$$

Generalized Squared Distance to ClusterGrouping

From ClusterGrouping	1	2
1	0	32.34093
2	32.34093	0

Univariate Test Statistics

F Statistics, Num DF=1, Den DF=8

Variable	Label	Total Standard Deviation	Pooled Standard Deviation	Between Standard Deviation	R-Square	R-Square / (1-RSq)
Energy Consumption	Energy Consumption	3.4214	3.5864	0.7003	0.0233	0.0238
Sugar Consumption	Sugar Consumption	10.2972	3.9498	12.8801	0.8692	6.6461

Variable	F Value	Pr > F
Energy Consumption	0.19	0.6739
Sugar Consumption	53.17	<.0001

Average R-Square

Unweighted	0.4462458
Weighted by Variance	0.7851093

Canonical Discriminant Analysis

	Canonical Correlation	Adjusted Canonical Correlation	Approximate Standard Error	Squared Canonical Correlation
1	0.952135	0.951626	0.031146	0.906562

Eigenvalues of Inv(E)*H = CanRsq/(1-CanRsq)

	Eigenvalue	Difference	Proportion	Cumulative
1	9.7023		1.0000	1.0000

Test of H0: The canonical correlations in the current row and all that follow are zero

	Likelihood Ratio	Approximate F Value	Num DF	Den DF	Pr > F
1	0.09343803	33.96	2	7	0.0002

NOTE: The F statistic is exact.

Total Canonical Structure

Variable	Label	Can1
EnergyConsumption	EnergyConsumption	-0.160237
SugarConsumption	SugarConsumption	0.979185

Between Canonical Structure

Variable	Label	Can1
EnergyConsumption	EnergyConsumption	-1.000000
SugarConsumption	SugarConsumption	1.000000

Pooled Within Canonical Structure

Variable	Label	Can1
EnergyConsumption	EnergyConsumption	-0.049561
SugarConsumption	SugarConsumption	0.827651

Total-Sample Standardized Canonical Coefficients

Variable	Label	Can1
EnergyConsumption	EnergyConsumption	-0.626618053
SugarConsumption	SugarConsumption	3.047360629

Pooled Within-Class Standardized Canonical Coefficients

Variable	Label	Can1
EnergyConsumption	EnergyConsumption	-0.656848037
SugarConsumption	SugarConsumption	1.168906049

Raw Canonical Coefficients

Variable	Label	Can1
EnergyConsumption	EnergyConsumption	-.1831475166
SugarConsumption	SugarConsumption	0.2959403315

Class Means on Canonical Variables

ClusterGrouping	Can1
1	-2.274763598
2	3.412145397

Linear Discriminant Function

$$\text{Constant} = -.5\,\bar{X}_j' \,COV^{-1}\, \bar{X}_j \qquad \text{Coefficient Vector} = COV^{-1}\, \bar{X}_j$$

Then, next section provides the coefficients for the linear discriminant function of each class. These are the linear combinations of the responses that “define” each clustered group. The constant or intercept terms are also included, which can be mathematically expressed below:

$$Function1 = -99.231 - 0.454 \times EnergyConsumption + 3.763 \times SugarConsumption$$

$$Function2 = -185.825 - 1.495 \times EnergyConsumption + 5.446 \times SugarConsumption$$

Linear Discriminant Function for ClusterGrouping

Variable	Label	1	2
Constant		-99.23144	-185.82524
EnergyConsumption	EnergyConsumption	-0.45393	-1.49547
SugarConsumption	SugarConsumption	3.76339	5.44637

Classification Summary for Calibration Data: WORK.FITNESS
Resubstitution Summary using Linear Discriminant Function

Generalized Squared Distance Function

$$D_j^2(X) = (X-\bar{X}_j)' \, COV^{-1} \, (X-\bar{X}_j)$$

Posterior Probability of Membership in Each ClusterGrouping

$$Pr(j|X) = \exp(-.5\, D_j^2(X)) \,/\, SUM_k \exp(-.5\, D_k^2(X))$$

The next section lists an error matrix giving the number of observations correctly classified and also misclassified. These are calculated by simply running the data set through the discriminant function to see how they get classified. Here, we see that the percentage rates of classification error for both the 6 observations in group 1 and 4 observations in group 2 are all equal to 0.00% This indicates that both groups do not have any misclassified observations. In addition, the overall error rate is also computed to be 0.00%, which reflects that the clustering we performed earlier on all observed consumption records are correctly classified

Number of Observations and Percent Classified into ClusterGrouping

From ClusterGrouping	1	2	Total
1	6	0	6
	100.00	0.00	100.00
2	0	4	4
	0.00	100.00	100.00
Total	6	4	10
	60.00	40.00	100.00
Priors	0.5	0.5	

Error Count Estimates for ClusterGrouping

	1	2	Total
Rate	0.0000	0.0000	0.0000
Priors	0.5000	0.5000	

Discriminant analysis is a powerful method for classifying new and unknown data. If a new dataset (i.e., the test dataset) is available, the output from the previous run (using previously specified training dataset) can be used to classify the new dataset using the code:

```
PROC DISCRIM ANOVA CANONICAL DATA=Fitness OUTSTAT=FitnessStat LIST CROSSVALIDATE
POOL=YES;
CLASS ClusterGrouping;
VAR EnergyConsumption SugarConsumption;
PRIORS EQUAL;
RUN;
```

In the first statement, the "OUTSTAT=" option will store the calibration information in a new dataset that is ready to be used for classifying future observations. The "LIST" option displays the re-substitution classification results for each observation. The "CROSSVALIDATE" option lists cross validation error-rate estimates. The default "POOL=YES" assumes equal variances among groups.

```
PROC PRINT DATA=FitnessTest;
VAR FitnessAbility EnergyConsumption SugarConsumption;
RUN;
```

This procedure outputs the testing dataset in the following:

Obs	Fitness Ability	Energy Consumption	Sugar Consumption
1	Squat	24.54	50.29
2	Bench press	19.32	68.37
3	Leg extension	21.65	54.98
4	Sit up	20.59	69.83

The second "**PROC DISCRIM**" procedure applies the early-calculated calibration information stored in FitnessStat dataset to classify a test dataset FitnessTest. The TESTLIST option displays the classification results for each observation in the test dataset.

```
PROC DISCRIM DATA=FitnessStat TESTDATA=FitnessTest TESTOUT=FitnessOut TESTLIST;
CLASS ClusterGrouping;
VAR EnergyConsumption SugarConsumption;
RUN;
```

Results of the DISCRIM procedure are listed in the following:

The DISCRIM Procedure
Classification Results for Test Data: WORK.FITNESSTEST
Classification Results using Linear Discriminant Function

Generalized Squared Distance Function

$$D_j^2(X) = (X-\bar{X}_j)' \, COV^{-1} \, (X-\bar{X}_j)$$

Posterior Probability of Membership in Each ClusterGrouping

$$Pr(j|X) = \exp(-.5\, D_j^2(X)) \,/\, SUM_k \exp(-.5\, D_k^2(X))$$

Posterior Probability of Membership in ClusterGrouping

Obs	From ClusterGrouping	Classified into Cluster Grouping	1	2
1	2	1 *	1.0000	0.0000
2	2	2	0.0002	0.9998
3	1	1	1.0000	0.0000
4	2	2	0.0001	0.9999

* Misclassified observation

Classification Summary for Test Data: WORK.FITNESSTEST
Classification Summary using Linear Discriminant Function

Generalized Squared Distance Function

$$D_j^2(X) = (X-\bar{X}_j)' \, COV^{-1} \, (X-\bar{X}_j)$$

Posterior Probability of Membership in Each ClusterGrouping

$$Pr(j|X) = \exp(-.5\, D_j^2(X)) \,/\, SUM_k \exp(-.5\, D_k^2(X))$$

Observation Profile for Test Data

Number of Observations Read	4
Number of Observations Used	4

The next section lists an error matrix representing the number of observations that are correctly classified and misclassified. Based on the discriminant functions estimated from the training dataset running earlier, the observations in test dataset can be classified. As you can clearly see from the table below, one pre-defined group 2's observation was misclassified to group 1, and the other two pre-defined group 2's observations were classified correctly. Therefore, it yields a 33.33% rate of error. In addition, the only one observation from pre-defined group 1 was correctly classified (as group 1).

Number of Observations and Percent Classified into ClusterGrouping

From ClusterGrouping	1	2	Total
1	1 100.00	0 0.00	1 100.00
2	1 33.33	2 66.67	3 100.00
Total	2 50.00	2 50.00	4 100.00
Priors	0.5	0.5	

Error Count Estimates for ClusterGrouping

	1	2	Total
Rate	0.0000	0.3333	0.1667
Priors	0.5000	0.5000	

Example 2: Olympic Heptahlon data analysis using PROC PRINCOMP

A heptathlon is a track and field athletics combined events contest made up of seven sports events. In Olympics women heptathlon competition, the women's heptathlon rules are exactly the same as the men's decathlon rules except that the heptathlon consists of seven events, also held on two consecutive days. The first day's events, in order, include the 100-meter hurdles, the high jump, shot put and the 200-meter run. The second day's events, in order, include the long jump, the javelin throw and an 800-meter run. For your reference, this sports data and statistical analysis can be referenced at web site:

http://cran.r-project.org/web/packages/HSAUR2/vignettes/Ch_principal_components_analysis.pdf

In our second demonstration example, the clustering algorithm is applied to 1988 Olympic Heptathlon data to produce characterizations of groups of the leading athletes.

PCA is a linear transformation which is usually applied on highly correlated multidimensional data. In general, the basic procedures of the transformation are:

1. organizing a dataset in a matrix form
2. data centering by subtracting the dimensional means from themselves, so each of the dimensions in the dataset has zero mean
3. compute the covariance matrix (i.e., non-standardized PCA) or the correlation matrix (i.e., standardized PCA, also known as "scaling")
4. compute either the eigenvectors and eigenvalues of the covariance (or the correlation) matrix, or the SVD of the data matrix
5. sort variances in decreasing order (i.e., decreasing eigenvalues)
6. project original data to get predicted PC scored vectors

The score sorted Heptathlon data can be formatted and displayed with the SAS **PROC SQL** code as follows:

```
PROC SQL;
CREATE TABLE Mylib.Heptathlon AS
SELECT Name AS Names LABEL='Names' FORMAT=$20.,
hurdles AS Hurdles LABEL='Hurdles' FORMAT=6.2,
highjump AS Highjump LABEL='Highjump' FORMAT=6.2,
shot AS Shot LABEL='Shot' FORMAT=6.2,
run200m AS Run200m LABEL='Run200m' FORMAT=6.2,
longjump AS Longjump LABEL='Longjump' FORMAT=6.2,
javelin AS Javelin LABEL='Javelin' FORMAT=6.2,
run800m AS Run800m LABEL='Run800m' FORMAT=6.2,
score AS Score LABEL='Score' FORMAT=6.
FROM WORK.Heptathlon
ORDER BY Score DESCENDING;
QUIT;

PROC PRINT DATA=Mylib.Heptathlon;
RUN;
```

The individual and total scores are displayed in the following SAS output.

Obs	Names	Hurdles	Highjump	Shot
1	Joyner-Kersee (USA)	3.73	1.86	15.80
2	John (GDR)	3.57	1.80	16.23
3	Behmer (GDR)	3.22	1.83	14.20
4	Choubenkova (URS)	2.91	1.74	14.76
5	Sablovskaite (URS)	2.81	1.80	15.23
6	Schulz (GDR)	2.67	1.83	13.50
7	Fleming (AUS)	3.04	1.80	12.88
8	Greiner (USA)	2.87	1.80	14.13
9	Bouraga (URS)	3.17	1.77	12.62
10	Lajbnerova (CZE)	2.79	1.83	14.28
11	Wijnsma (HOL)	2.67	1.86	13.01
12	Dimitrova (BUL)	3.18	1.80	12.88
13	Scheider (SWI)	2.57	1.86	11.58
14	Braun (FRG)	2.71	1.83	13.16
15	Ruotsalainen (FIN)	2.63	1.80	12.32
16	Yuping (CHN)	2.49	1.86	14.21
17	Hagger (GB)	2.95	1.80	12.75
18	Brown (USA)	2.35	1.83	12.69
19	Mulliner (GB)	2.03	1.71	12.68
20	Hautenauve (BEL)	2.38	1.77	11.81
21	Kytola (FIN)	2.11	1.77	11.66
22	Geremias (BRA)	2.19	1.71	12.95
23	Hui-Ing (TAI)	1.57	1.68	10.00
24	Jeong-Mi (KOR)	1.89	1.71	10.83

Obs	Run200m	Longjump	Javelin	Run800m	Score
1	4.05	7.27	45.66	34.92	7291
2	2.96	6.71	42.56	37.31	6897
3	3.51	6.68	44.54	39.23	6858
4	2.68	6.32	47.46	35.53	6540
5	2.69	6.25	42.78	31.19	6540
6	1.96	6.33	42.82	37.64	6411
7	3.02	6.37	40.28	30.89	6351
8	2.13	6.47	38.00	29.78	6297
9	3.02	6.28	39.06	28.69	6252
10	1.75	6.11	42.20	27.38	6252
11	1.58	6.34	37.86	31.94	6205
12	3.02	6.37	40.28	30.89	6171
13	1.74	6.05	47.50	28.50	6137
14	1.83	6.12	44.58	20.61	6109
15	2.00	6.08	45.44	26.37	6101
16	1.61	6.40	38.60	16.76	6087
17	1.14	6.34	35.76	24.95	5975
18	1.78	6.13	44.34	17.00	5972
19	1.69	6.10	37.76	25.41	5746
20	1.00	5.99	35.68	29.53	5734
21	0.92	5.75	39.48	30.08	5686
22	1.11	5.50	39.64	19.41	5508
23	1.38	5.47	39.14	26.13	5290
24	0.00	5.50	39.26	24.26	5289

The SAS code for **PRINCOMP** procedure on the Heptathlon data is illustrated below:

```
PROC PRINCOMP DATA=Mylib.Heptathlon COV OUT=Mylib.HeptathlonOut
OUTSTAT=Mylib.HeptathlonStat N=5;
      VAR Hurdles Highjump Shot Run200m Longjump Javelin Run800m;
RUN;
```

Here, the OUT statement is used to generate a dataset, HeptathlonOut, which contains the calculated PCA scores.

```
PROC PRINT DATA=Mylib.HeptathlonStat;
RUN;
```

Computed output results from **PRINCOMP** procedure are listed in the following. The first section lists the number of observations and variables used along with the simple summarized means and standard deviations for each variable.

The PRINCOMP Procedure

Observations	24
Variables	7

Simple Statistics

	Hurdles	Highjump	Shot	Run200m
Mean	2.687500000	1.793750000	13.17333333	2.023750000
StD	0.514563978	0.052321124	1.49714995	0.936769716

Simple Statistics

	Longjump	Javelin	Run800m
Mean	6.205416667	41.27833333	28.51666667
StD	0.401659375	3.46870690	6.14724800

The second section reports the covariance matrix for the seven quantitative variables. However, the Pearson correlation matrix would be used as the default method if it's not specified. If the correlation matrix is used, the variables will be standardized and the total variance will be equal to the number of variables used in the analysis. This is because each standardized variable will have 'variance equals 1' values. If the covariance matrix is used, the variables will remain in their original metric. Please note that in this illustration, the PCA run is not rigorously defined and tested; however, it is used for demonstration purpose only.

Covariance Matrix

		Hurdles	Highjump	Shot
Hurdles	Hurdles	0.26477609	0.01566196	0.59063913
Highjump	Highjump	0.01566196	0.00273750	0.03640000
Shot	Shot	0.59063913	0.03640000	2.24145797
Run200m	Run200m	0.40010109	0.01915924	0.93886957
Longjump	Longjump	0.18380978	0.01392663	0.47147681
Javelin	Javelin	0.59343043	0.06317174	1.78143188
Run800m	Run800m	1.76750435	0.04899565	3.75765942

Covariance Matrix

	Run200m	Longjump	Javelin	Run800m
Hurdles	0.40010109	0.18380978	0.59343043	1.76750435
Highjump	0.01915924	0.01392663	0.06317174	0.04899565
Shot	0.93886957	0.47147681	1.78143188	3.75765942
Run200m	0.87753750	0.30500489	1.52979783	3.30074783
Longjump	0.30500489	0.16133025	0.39997464	1.29227971
Javelin	1.52979783	0.39997464	12.03192754	5.45729855
Run800m	3.30074783	1.29227971	5.45729855	37.78865797

Total Variance 53.368424819

The last section of the output provides the eigenvalues and eigenvectors (or loadings) for each axis. This calculated evidence reflects that the first two axes will describe the majority of the variability among these seven variables. Therefore, we "retain" or interpret only PCA axes 1 and 2. Please note that some other methods for determining the number of axes to retain are to use only those axes with eigenvalues greater than 1.0.

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	39.8213674	28.7087460	0.7462	0.7462
2	11.1126213	9.1093028	0.2082	0.9544
3	2.0033185	1.6549203	0.0375	0.9919
4	0.3483982	0.2879550	0.0065	0.9985
5	0.0604433		0.0011	0.9996

Eigenvalues can be interpreted as the variances of the principal components. The first component accounts for 74.62% of the variance, and the first two components account for 95.44% of the variance. The difference column provides the difference between the eigenvalues. It can be easily computed and illustrated as, for instance: 39.82 – 11.11 = 28.71. This column allows users to view how quickly the eigenvalues are decreasing. The proportion column is the proportion of the total variance that each principal component accounts for. The last Cumulative column is the sum of the proportion column.

Eigenvectors

		Prin1	Prin2	Prin3	Prin4	Prin5
Hurdles	Hurdles	0.048991	0.025317	0.219769	0.258721	0.770392
Highjump	Highjump	0.001689	0.005045	0.014128	0.007663	0.113562
Shot	Shot	0.109842	0.109469	0.884290	-.415090	-.143844
Run200m	Run200m	0.093345	0.082934	0.309956	0.850113	-.403820
Longjump	Longjump	0.035796	0.016070	0.184285	0.183911	0.457550
Javelin	Javelin	0.203545	0.964065	-.162681	-.046438	0.021370
Run800m	Run800m	0.966492	-.225372	-.114165	-.045089	-.005347

The eigenvectors indicate the relative importance of each variable within the individual axis. The relative importance is determined based on the absolute magnitude of the eigenvector coefficients or loadings. Deciding which eigenvector coefficients are "large" is really a subjective decision, and there are no definitely clear rules for selecting coefficients to judge the importance. In this printed output, PCA axis #1 seems to have large loadings on the Run800m and Javelin variables. Thus, axis #1, which accounts for 74.62% of the variability, appears to be associated with these two variables. The second axis has large loadings on the Javelin and Run800m variables. Please note about the second axis that the loading values for these two variables, Javelin and Run800m, are going to the opposite direction. This indicates the negative correlation between Javelin and Run800m. Based on these findings, surprisingly but interestingly, the Run800m and Javelin events seem to play the dominant role in this 1988 Olympic Heptathlon game.

CONCLUSION

The SAS procedures, PROC CLUSTER and PROC TREE, are all intuitive and powerful in separating heterogeneous data. The presented results indicate that multivariate discriminant analysis is effective when training dataset is calibrated and then its discriminant functions can be applied to the test dataset.

Furthermore, the purpose of principal component analysis, PROC PRINCOMP, is to derive a small number of independent linear combinations (i.e., principal components) of a set of variables that retain as much of the information in the original variables as possible. Its goal is to squeeze a high-dimensional distribution into a smaller-dimensional space that has most of the variation. PCA can be used to eliminate the redundant or dependent channels in some instances; as a result, it will yield a compact and optimal description of the dataset.

ACKNOWLEDGMENTS

The author would like to take this opportunity to acknowledge all of the guidance and support given to him by Drs. Ravi Bhatia and Jeff Longmate at City of Hope National Medical Center.

REFERENCES

Identifying Plant Species: A Botanical Analysis Using PROC DISCRIM, Robert G. Downer and Philip E. Hyatt, which is available on-line at http://www2.sas.com/proceedings/sugi27/p199-27.pdf

Application of Proc Discrim and Proc Logistic in Credit Risk Modeling, Jin Li, which is available on-line at http://www2.sas.com/proceedings/sugi31/081-31.pdf

Principal Component Analysis vs. Exploratory Factor Analysis, Diana D. Suhr, University of Northern Colorado, which is available on-line at: http://www2.sas.com/proceedings/sugi30/203-30.pdf.

A Handbook of Statistical Analyses Using R, 2nd Edition, Brian S. Everitt and Torsten Hothorn, which is available on-line at:
http://cran.r-project.org/web/packages/HSAUR2/vignettes/Ch_principal_components_analysis.pdf

RECOMMENDED READING

SAS 9.2 Documentation (SAS/STAT(R) 9.2 User's Guide, Second Edition):

The CLUSTER Procedure
http://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#/documentation/cdl/en/statug/63033/HTML/default/cluster_toc.htm

The DISCRIM Procedure
http://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#/documentation/cdl/en/statug/63033/HTML/default/discrim_toc.htm

The PRINCOMP Procedure
http://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#/documentation/cdl/en/statug/63033/HTML/default/princomp_toc.htm

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Keh-Dong Shiang, Ph.D.
Division of Biostatistics, Department of Information Sciences
Division of Hematopoietic Stem Cell and Leukemia Research
City of Hope National Medical Center
1500 East Duarte Road
Duarte, CA 91010-3000
Phone: (626)256-HOPE(4673) Ext. 65768
Fax: (626)471-7106
E-mail: kshiang@coh.org
Web: www.coh.org