Interactive Stepwise Discriminant Analysis *

Dimitar Vandev

ABSTRACT

`ldagui.m` is an interactive tool for linear and quadratic discriminant analysis. The reason for developing such a tool is the unconsistency of conventional statistical programs in following aspects:

- treating missing data;
- interaction with the user;
- testing the quality of obtained model.

*Supported by contracts:PRO-ENBIS: GTC1-2001-43031 and WINE DB: G6RD-CT-2001-00646

Interactive Stepwise Discriminant Analysis *

Dimitar Vandev

ABSTRACT

`ldagui.m` is an interactive tool for linear and quadratic discriminant analysis. The reason for developing such a tool is the unconsistency of conventional statistical programs in following aspects:

- treating missing data;
- interaction with the user;
- testing the quality of obtained model.

*Supported by contracts:PRO-ENBIS: GTC1-2001-43031 and WINE DB: G6RD-CT-2001-00646

Overview
File
Model
Diagnostics
Use
Algorithms
References
Home Page
Title Page
Go Back
Full Screen
Close
Quit

1. Overview

Discriminant analysis (DA) is a very popular tool in applied statistics.

1. Overview

Discriminant analysis (DA) is a very popular tool in applied statistics.

The program `Ldagui.m` is developed in the frame of MATLAB. It is used with the help of menus, shortcuts, listboxes and a slider.

1. Overview

Discriminant analysis (DA) is a very popular tool in applied statistics.

The program `Ldagui.m` is developed in the frame of MATLAB. It is used with the help of menus, shortcuts, listboxes and a slider.

- First we will shortly outline the mathematics behind DA.

1. Overview

Discriminant analysis (DA) is a very popular tool in applied statistics.

The program `Ldagui.m` is developed in the frame of MATLAB. It is used with the help of menus, shortcuts, listboxes and a slider.

- First we will shortly outline the mathematics behind DA.
- Then we will describe menus and shortcuts of the program.

1. Overview

Discriminant analysis (DA) is a very popular tool in applied statistics.

The program `Ldagui.m` is developed in the frame of MATLAB. It is used with the help of menus, shortcuts, listboxes and a slider.

- First we will shortly outline the mathematics behind DA.
- Then we will describe menus and shortcuts of the program.
- Finally a small demonstration will be done to illustrate other features of the program.

Let suppose we have observed two random variables:

1. continuous $\xi \in R^p$;

Let suppose we have observed two random variables:

1. continuous $\xi \in R^p$;
2. discrete (or categorical) η with values in $1, 2, \ldots, G$.

Let suppose we have observed two random variables:

1. continuous $\xi \in R^p$;
2. discrete (or categorical) η with values in $1, 2, \ldots, G$.
3. they have joined distribution (DA model):
   - $\mathbf{P}(\eta = g) = p(g)$
   - Conditional distribution of $\xi \in R^p$ given $\eta = g$ is described by the density $f(x, m(g), C(g))$.

Let suppose we have observed two random variables:

1. continuous $\xi \in R^p$;
2. discrete (or categorical) η with values in $1, 2, \ldots, G$.
3. they have joined distribution (DA model):
   - $\mathbf{P}(\eta = g) = p(g)$
   - Conditional distribution of $\xi \in R^p$ given $\eta = g$ is described by the density $f(x, m(g), C(g))$.

Here f is the density of Gauss distribution in R^p described by two parameters:

- mean - $m(g)$;
- covariance - $C(g)$,

1.1. Bayes formula

Suppose we know the parameters of this model:

1. The prior probabilities - $\{p(g)\}$;

1.1. Bayes formula

Suppose we know the parameters of this model:

1. The prior probabilities - $\{p(g)\}$;
2. Group means - $\{m(g)\}$;

1.1. Bayes formula

Suppose we know the parameters of this model:

1. The prior probabilities - $\{p(g)\}$;
2. Group means - $\{m(g)\}$;
3. Within group covariance matrices - $C(g)$;

That is, the set of numbers: $\{p_g, m_g, C(g), g = 1, 2, \ldots\ldots, G\}$

1.1. Bayes formula

Suppose we know the parameters of this model:

1. The prior probabilities - $\{p(g)\}$;
2. Group means - $\{m(g)\}$;
3. Within group covariance matrices - $C(g)$;

That is, the set of numbers: $\{p_g, m_g, C(g), g = 1, 2, \ldots\ldots, G\}$

Then according of the famous formula of Bayes we may write down the conditional probability of $\eta = g$ given x:

$$\mathbf{P}\left(\eta = g | \xi = g\right) = q(g|x) = c(x).p(g).f(x, m(g), C(g)),$$

1.1. Bayes formula

Suppose we know the parameters of this model:

1. The prior probabilities - $\{p(g)\}$;
2. Group means - $\{m(g)\}$;
3. Within group covariance matrices - $C(g)$;

That is, the set of numbers: $\{p_g, m_g, C(g), g = 1, 2, \ldots\ldots, G\}$

Then according of the famous formula of Bayes we may write down the conditional probability of $\eta = g$ given x:

$$\mathbf{P}\left(\eta = g|\xi = g\right) = q(g|x) = c(x).p(g).f(x, m(g), C(g)), \tag{1}$$

where c is a normalizing constant, such that $\sum q(g|x) = 1$.

1.1. Bayes formula

Suppose we know the parameters of this model:

1. The prior probabilities - $\{p(g)\}$;
2. Group means - $\{m(g)\}$;
3. Within group covariance matrices - $C(g)$;

That is, the set of numbers: $\{p_g, m_g, C(g), g = 1, 2, \ldots\ldots, G\}$

Then according of the famous formula of Bayes we may write down the conditional probability of $\eta = g$ given x:

$$\mathbf{P}\left(\eta = g|\xi = g\right) = q(g|x) = c(x).p(g).f(x, m(g), C(g)), \tag{1}$$

where c is a normalizing constant, such that $\sum q(g|x) = 1$.

We call this probability posterior and say that the observation x belongs to the group g with probability $q(g|x)$.

1.1. Bayes formula

Suppose we know the parameters of this model:

1. The prior probabilities - $\{p(g)\}$;
2. Group means - $\{m(g)\}$;
3. Within group covariance matrices - $C(g)$;

That is, the set of numbers: $\{p_g, m_g, C(g), g = 1, 2, \ldots\ldots, G\}$

Then according of the famous formula of Bayes we may write down the conditional probability of $\eta = g$ given x:

$$\mathbf{P}\left(\eta = g|\xi = g\right) = q(g|x) = c(x).p(g).f(x, m(g), C(g)), \tag{1}$$

where c is a normalizing constant, such that $\sum q(g|x) = 1$.

We call this probability posterior and say that the observation x belongs to the group g with probability $q(g|x)$.

According the maximum likelihood principle the classification rule should then be:

$$\widehat{g} = \underset{h}{argmax} : q(h). \tag{2}$$

According the maximum likelihood principle the classification rule should then be:

$$\widehat{g} = \underset{h}{argmax} : q(h). \tag{2}$$

1.2. Linear Discriminant Analysis

Suppose that within group covariance $C(g)$ are equal:

$$C(g) = C, \quad (g = 1, 2, \ldots, G)$$

1.2. Linear Discriminant Analysis

Suppose that within group covariance $C(g)$ are equal:

$$C(g) = C, \quad (g = 1, 2, \ldots, G) \qquad (3)$$

Then the maximum likelihood rule (2) becomes a set of inequalities:

$$p(\widehat{g}).f(x, m(\widehat{g}), C) \geq p(h).f(x, m(h), C), \quad (h = 1, 2, \ldots, G).$$

1.2. Linear Discriminant Analysis

Suppose that within group covariance $C(g)$ are equal:

$$C(g) = C, \quad (g = 1, 2, \ldots, G) \tag{3}$$

Then the maximum likelihood rule (2) becomes a set of inequalities:

$$p(\widehat{g}).f(x, m(\widehat{g}), C) \geq p(h).f(x, m(h), C), \quad (h = 1, 2, \ldots, G). \tag{4}$$

or (what is the same) to:

$$b(\widehat{g})'x + a(\widehat{g}) \geq b(h)'x + a(h), \quad (h = 1, 2, \ldots, G), \tag{5}$$

We decide that the observation x belongs to the group g, if for each h the inequality (5) holds:

$$L_g(x) \geq L_h(x), \quad (h = 1, 2, \ldots, G), \tag{6}$$

The functions L are called discriminant functions.

Overview
File
Model
Diagnostics
Use
Algorithms
References
Home Page
Title Page
Go Back
Full Screen
Close
Quit

The vector $b(g)$ and the number $a(g)$ in this case are calculated explicitly:

$$b(h) = m(h)'C^{-1}, \qquad a(h) = \log p(h) - m(h)'C^{-1}m(h). \tag{7}$$

This is why DA takes the name Linear - the discriminant functions are linear functions.

The vector $b(g)$ and the number $a(g)$ in this case are calculated explicitly:

$$b(h) = m(h)'C^{-1}, \qquad a(h) = \log p(h) - m(h)'C^{-1}m(h). \tag{7}$$

This is why DA takes the name Linear - the discriminant functions are linear functions.

Overview
File
Model
Diagnostics
Use
Algorithms
References
Home Page
Title Page
Go Back
Full Screen
Close
Quit

1.3. Quadratic Discriminant Analysis

When the assumption (3):$C(g) = C$ is not appropriate, the corresponding functions become quadratic.

1.3. Quadratic Discriminant Analysis

When the assumption (3):$C(g) = C$ is not appropriate, the corresponding functions become quadratic.

If one has equal prior probabilities $p(h) = 1/G$, the solution of the classification problem (2) is equivalent to the minimization of so called Mahalanobis distances of the observation to the group means:

$$h(x, g) = (x - m(g))' C_g^{-1} (x - m(g))$$

1.3. Quadratic Discriminant Analysis

When the assumption (3):$C(g) = C$ is not appropriate, the corresponding functions become quadratic.

If one has equal prior probabilities $p(h) = 1/G$, the solution of the classification problem (2) is equivalent to the minimization of so called Mahalanobis distances of the observation to the group means:

$$h(x, g) = (x - m(g))' C_g^{-1} (x - m(g)) \tag{8}$$

One uses Mahalanobis distances (8) to classify the observation to the closest group:

$$\widehat{g} = \underset{h}{argmin}\, h(x, h).$$

In general however, the Bayes rule (1) is better.

1.4. Estimation

Let the training sample consists of vectors $(g_i, x_i), i = 1.2. \ldots, n$. Denote by $I(g)$ the set $\{i : g_i = g\}$ and let $n(g) = |I(g)|$. First the standard calculations - averages:

$$m(g) = \frac{1}{n(g)} \sum_{i \in I(g)} x_i.$$

1.4. Estimation

Let the training sample consists of vectors $(g_i, x_i), i = 1.2. \ldots, n$. Denote by $I(g)$ the set $\{i : g_i = g\}$ and let $n(g) = |I(g)|$. First the standard calculations - averages:

$$m(g) = \frac{1}{n(g)} \sum_{i \in I(g)} x_i. \tag{9}$$

Cross products:

$$SS(g) = \sum_{i \in I(g)} (x_i - m(g))(x_i - m(g))', \quad SS_{in} = \sum_{g \in G} SS(g)$$

1.4. Estimation

Let the training sample consists of vectors $(g_i, x_i), i = 1.2. \dots, n.$ Denote by $I(g)$ the set $\{i : g_i = g\}$ and let $n(g) = |I(g)|$. First the standard calculations - averages:

$$m(g) = \frac{1}{n(g)} \sum_{i \in I(g)} x_i. \tag{9}$$

Cross products:

$$SS(g) = \sum_{i \in I(g)} (x_i - m(g))(x_i - m(g))', \quad SS_{in} = \sum_{g \in G} SS(g) \tag{10}$$

Standard maximum likelihood estimates are:

$$C(g) = \frac{1}{n(g) - 1} SS(g), \quad C = \frac{1}{n - G} SS_{in}. \tag{11}$$

We propose to correct the within group covariance estimate considering instead the mixture:

$$C_g := (1-\alpha) * C + \alpha * C_g. \tag{12}$$

The parameter $0 \leq \alpha \leq 1$ is to be chosen in interactive way via slider. Such corrections are not new (see for example (Lauter, 159-168) in (Fedorov, 1992)).

We propose to correct the within group covariance estimate considering instead the mixture:

$$C_g := (1 - \alpha) * C + \alpha * C_g. \tag{12}$$

The parameter $0 \leq \alpha \leq 1$ is to be chosen in interactive way via slider. Such corrections are not new (see for example (Lauter, 159-168) in (Fedorov, 1992)).

1.5. Selecting variables

The standard Fisher approach was to maximize the between group variance or (what is the same) to minimize common within group variance:

$$SS = \sum_{g \in G} \sum_{i \in I(g)} (x_i - m(g))(x_i - m(g))'.$$

1.5. Selecting variables

The standard Fisher approach was to maximize the between group variance or (what is the same) to minimize common within group variance:

$$SS = \sum_{g \in G} \sum_{i \in I(g)} (x_i - m(g))(x_i - m(g))'.$$

One may use trace or determinant to find corresponding variables. Now in all programs the so called Wilks lambda is used:

$$\Lambda = \frac{\det(SS_{in})}{\det(SS_{total})}.$$

1.5. Selecting variables

The standard Fisher approach was to maximize the between group variance or (what is the same) to minimize common within group variance:

$$SS = \sum_{g \in G} \sum_{i \in I(g)} (x_i - m(g))(x_i - m(g))'.$$

One may use trace or determinant to find corresponding variables. Now in all programs the so called Wilks lambda is used:

$$\Lambda = \frac{\det(SS_{in})}{\det(SS_{total})}.$$

It is easy to calculate (see (Jennrich, 1977)) and convenient to update when new variable is to be chosen.

2. File

Now we will go trough the menus of the program.

2. File

Now we will go trough the menus of the program.

The File drop down menu may be used separately in order to fill the missing data with within group means.

2.1. Open Data

The program loads a data (.csv) file.

The program will ask you to supply one variable to be used for classification. You should decide. Otherwise the use of `Ldagui.m` is impossible.

2.1. Open Data

The program loads a data (.csv) file.

The program will ask you to supply one variable to be used for classification. You should decide. Otherwise the use of `Ldagui.m` is impossible.

Comma separated values

These (.csv) files are common for many applications. They are easily exported and imported by Excel and Statistica programs.

2.1. Open Data

The program loads a data (.csv) file.

The program will ask you to supply one variable to be used for classification. You should decide. Otherwise the use of `Ldagui.m` is impossible.

Comma separated values

These (.csv) files are common for many applications. They are easily exported and imported by Excel and Statistica programs.

`Ldagui.m` assumes that:

- the first row contains strings for variable names;
- the first column contains strings for case names.

All other fields should contain numbers (or be empty for missing values).

Categorical variables

The categorical variables should have consecutive positive integer values. When exporting from Statistica you should say integers instead of text values.

Categorical variables

The categorical variables should have consecutive positive integer values. When exporting from Statistica you should say integers instead of text values.

Any information about text values they may have in Statistica is loss and moreover their values in `Ldagui.m` are changed to first natural numbers: 1,2,3...

2.2. Filling Missing Data

You will be asked to supply selection variable. This is not obligatory. Then you should supply obligatory classification variable.

2.2. Filling Missing Data

You will be asked to supply selection variable. This is not obligatory. Then you should supply obligatory classification variable.

Both classification and selection variables will be used in the algorithm for filling missing data.

2.2. Filling Missing Data

You will be asked to supply selection variable. This is not obligatory. Then you should supply obligatory classification variable.

Both classification and selection variables will be used in the algorithm for filling missing data.

Filling Missing Data is done automatically by LDAgui upon reading of .csv data. They are replaced by within group means. These means are formed by each combination of values of classification and selection variables.

2.3. Save Data

Saves the data file in a form of comma separated file for later import in Excel or Statistica.

2.3. Save Data

Saves the data file in a form of comma separated file for later import in Excel or Statistica.

2.4. Exit to MATLAB

Saves the MATLAB workspace in tempmodel.mat for later examination and use.

2.3. Save Data

Saves the data file in a form of comma separated file for later import in Excel or Statistica.

2.4. Exit to MATLAB

Saves the MATLAB workspace in tempmodel.mat for later examination and use.

2.5. Quit MATLAB

3. Model

Under model we understand:

1. the training sample (with no missing values);

3. Model

Under model we understand:

1. the training sample (with no missing values);
2. a subset of cases having fixed value of the selection variable (if any);

3. Model

Under model we understand:

1. the training sample (with no missing values);
2. a subset of cases having fixed value of the selection variable (if any);
3. a subset of variables chosen for predictors (may be empty);

3. Model

Under model we understand:

1. the training sample (with no missing values);
2. a subset of cases having fixed value of the selection variable (if any);
3. a subset of variables chosen for predictors (may be empty);
4. fixed value of the parameter α (12) of nonlinearity.

3. Model

Under model we understand:

1. the training sample (with no missing values);
2. a subset of cases having fixed value of the selection variable (if any);
3. a subset of variables chosen for predictors (may be empty);
4. fixed value of the parameter α (12) of nonlinearity.
5. the estimated parameters of DA model.

3. Model

Under model we understand:

1. the training sample (with no missing values);
2. a subset of cases having fixed value of the selection variable (if any);
3. a subset of variables chosen for predictors (may be empty);
4. fixed value of the parameter α (12) of nonlinearity.
5. the estimated parameters of DA model.

3.1. Build Model

Performs all preliminary calculations for an empty model with no selection variable taken into account. To activate this option click on the Selection Listbox.

Overview
File
Model
Diagnostics
Use
Algorithms
References
Home Page
Title Page
Go Back
Full Screen
Close
Quit

3.2. Load Model

Loads previously saved model (or workspace).

3.2. Load Model

Loads previously saved model (or workspace).

3.3. Save Model

Save the current model with data, names, selected groups, predictors, etc. for later use. In fact the current workspace of MATLAB is saved.

3.4. Print Results

The following results are printed in the MATLAB command window:

- File name - the name of file (data or model) you have loaded recently;
- Model name - the name of corresponding value of selection variable if any;
- Number of cases in training sample.

3.4. Print Results

The following results are printed in the MATLAB command window:

- File name - the name of file (data or model) you have loaded recently;
- Model name - the name of corresponding value of selection variable if any;
- Number of cases in training sample.
- Variables in model with their $f-$ and $p-$ values ordered;

3.4. Print Results

The following results are printed in the MATLAB command window:

- File name - the name of file (data or model) you have loaded recently;
- Model name - the name of corresponding value of selection variable if any;
- Number of cases in training sample.
- Variables in model with their $f-$ and $p-$ values ordered;
- Value of parameter α responsible for nonlinearity;

3.4. Print Results

The following results are printed in the MATLAB command window:

- File name - the name of file (data or model) you have loaded recently;
- Model name - the name of corresponding value of selection variable if any;
- Number of cases in training sample.
- Variables in model with their $f-$ and $p-$ values ordered;
- Value of parameter α responsible for nonlinearity;
- Value and p-value of Wilks Λ;

3.4. Print Results

The following results are printed in the MATLAB command window:

- File name - the name of file (data or model) you have loaded recently;
- Model name - the name of corresponding value of selection variable if any;
- Number of cases in training sample.
- Variables in model with their $f-$ and $p-$ values ordered;
- Value of parameter α responsible for nonlinearity;
- Value and p-value of Wilks Λ;
- Results of the classification of the training sample - number of errors;

3.4. Print Results

The following results are printed in the MATLAB command window:

- File name - the name of file (data or model) you have loaded recently;
- Model name - the name of corresponding value of selection variable if any;
- Number of cases in training sample.
- Variables in model with their $f-$ and $p-$ values ordered;
- Value of parameter α responsible for nonlinearity;
- Value and p-value of Wilks Λ;
- Results of the classification of the training sample - number of errors;
- Cases classified with probability below .8;

3.4. Print Results

The following results are printed in the MATLAB command window:

- File name - the name of file (data or model) you have loaded recently;
- Model name - the name of corresponding value of selection variable if any;
- Number of cases in training sample.
- Variables in model with their $f-$ and $p-$ values ordered;
- Value of parameter α responsible for nonlinearity;
- Value and p-value of Wilks Λ;
- Results of the classification of the training sample - number of errors;
- Cases classified with probability below .8;
- Estimated power of the model by groups.

3.4. Print Results

The following results are printed in the MATLAB command window:

- File name - the name of file (data or model) you have loaded recently;
- Model name - the name of corresponding value of selection variable if any;
- Number of cases in training sample.
- Variables in model with their $f-$ and $p-$ values ordered;
- Value of parameter α responsible for nonlinearity;
- Value and p-value of Wilks Λ;
- Results of the classification of the training sample - number of errors;
- Cases classified with probability below .8;
- Estimated power of the model by groups.

Finally, a huge sample with 6000 observations per group is produced according estimated within group means and covariance matrices. The sample is classified and results reported on the MATLAB command window. This may be considered as an estimate of the theoretical power of the model.

3.5. Clear Model

Clears any information for the model. You should start with Build model step.

Finally, a huge sample with 6000 observations per group is produced according estimated within group means and covariance matrices. The sample is classified and results reported on the MATLAB command window. This may be considered as an estimate of the theoretical power of the model.

3.5. Clear Model

Clears any information for the model. You should start with Build model step.

4. Diagnostics

The tools proposed for making adequate decision are:

- Test - (Ctrl-t) - produces a test random sample;

4. Diagnostics

The tools proposed for making adequate decision are:

- Test - (Ctrl-t) - produces a test random sample;
- Leave-One-Out - checks the model against deleting each of observations;

4. Diagnostics

The tools proposed for making adequate decision are:

- Test - (Ctrl-t) - produces a test random sample;
- Leave-One-Out - checks the model against deleting each of observations;
- Plot - (Ctrl-g) - makes two plots over canonical variables.

4. Diagnostics

The tools proposed for making adequate decision are:

- Test - (Ctrl-t) - produces a test random sample;
- Leave-One-Out - checks the model against deleting each of observations;
- Plot - (Ctrl-g) - makes two plots over canonical variables.

4.1. Test

A small sample with 100 observations per group is produced according estimated within group means and covariance matrices. The sample is classified and results reported on the MATLAB command window. This may be considered as an estimate of the power of the model. One may repeat this step in order to be sure or use print menu with larger sample 3.4.

4. Diagnostics

The tools proposed for making adequate decision are:

- Test - (Ctrl-t) - produces a test random sample;
- Leave-One-Out - checks the model against deleting each of observations;
- Plot - (Ctrl-g) - makes two plots over canonical variables.

4.1. Test

A small sample with 100 observations per group is produced according estimated within group means and covariance matrices. The sample is classified and results reported on the MATLAB command window. This may be considered as an estimate of the power of the model. One may repeat this step in order to be sure or use print menu with larger sample 3.4.

4.2. Leave-One-Out

A standard procedure is performed:

1. For each observation in the training sample a model with the same variables is build but without this particular observation.
2. The training sample is classified with this new model and classification errors counted.
3. The errors for all observations are summarized and reported.

4.3. Plot

Second (upper plot) and third canonical variables are plotted against the first (on horizontal axes). The training sample is plotted with different colors for the groups

4.2. Leave-One-Out

A standard procedure is performed:

1. For each observation in the training sample a model with the same variables is build but without this particular observation.
2. The training sample is classified with this new model and classification errors counted.
3. The errors for all observations are summarized and reported.

4.3. Plot

Second (upper plot) and third canonical variables are plotted against the first (on horizontal axes). The training sample is plotted with different colors for the groups

5. Use

5.1. Load sample

A standard data (.csv) file is loaded which should not contain missing values in the columns used for recognition. Columns to use should have the same variable names.

5. Use

5.1. Load sample

A standard data (.csv) file is loaded which should not contain missing values in the columns used for recognition. Columns to use should have the same variable names.

5.2. Print results

Results of classification are printed.

5. Use

5.1. Load sample

A standard data (.csv) file is loaded which should not contain missing values in the columns used for recognition. Columns to use should have the same variable names.

5.2. Print results

Results of classification are printed.

5.3. Save sample

The sample is saved in a data (.csv) file with resulting classification in the first column.

6. Algorithms

The calculations are based on the paper of (Jennrich, 1977) in the classical collection of (Einslein, Ralston et al., 1977) being in the foundations of the package BMDP(see (Dixon, 1981)).

6. Algorithms

The calculations are based on the paper of (Jennrich, 1977) in the classical collection of (Einslein, Ralston et al., 1977) being in the foundations of the package BMDP(see (Dixon, 1981)).

References

DIXON, J. (ed.) (1981). *BMDP Statistical Software - 81*. University of California, Los Angeles.

EINSLEIN, K., RALSTON, A., ET AL. (eds.) (1977). *Statistical Methods for Digital Computers*. John Wiley & Sons, New York.

FEDOROV, V. (ed.) (1992). *Model oriented data analysis: a survey of recent methods*. Physica-Verlag, Heidelberg.

JENNRICH, R. I. (1977). Stepwise discriminant analysis. In: (Einslein, Ralston et al., 1977), pp. 76–95.

LAUTER, H. (159-168). Bootstrap and estimation of nonlinear parameters. In: (Fedorov, 1992).

References

DIXON, J. (ed.) (1981). *BMDP Statistical Software - 81*. University of California, Los Angeles.

EINSLEIN, K., RALSTON, A., ET AL. (eds.) (1977). *Statistical Methods for Digital Computers*. John Wiley & Sons, New York.

FEDOROV, V. (ed.) (1992). *Model oriented data analysis: a survey of recent methods*. Physica-Verlag, Heidelberg.

JENNRICH, R. I. (1977). Stepwise discriminant analysis. In: (Einslein, Ralston et al., 1977), pp. 76–95.

LAUTER, H. (159-168). Bootstrap and estimation of nonlinear parameters. In: (Fedorov, 1992).