SURVIVAL ANALYSIS WITH COX'S PROPORTIONAL HAZARDS REGRESSION MODEL.

© M. URRUTIA AVISRROR (1994). Version 6.0 (English)

UNIVERSITY OF SALAMANCA - DEPARTAMENT OF UROLOGY
HOSPITAL UNIVERSITARIO DE SALAMANCA - CLINICAL RESEARCH UNIT
SALAMANCA - SPAIN

This version, compiled with Think Pascal ™, makes a full usage of the User's Interface of Apple's Macintosh Operating System.
To run the application you need at least a LC Mac or a higher model with a FPU (or the Pseudo-FP INIT) with a 8 bit Gray or Color Monitor. We firmly recommend to use System 7.xx and a minimum of 4 megabytes of RAM for maximum performance.
This statistical module allows the user to easily estimate survival functions from censored data and to assess the relationship between survival and a set of both continuous and categorical variables, referred to as covariates or prognostic factors in survival analysis. **Survival** can handle up to 30 variables and an unlimited number of cases, provided you have enough RAM memory in your system. The general performance of the program with virtual memory has not yet been assessed.
Modifications introduced respect to earlier versions of the program include:
-Use of both **fixed** and **time dependent** covariates.
-A **Text Editor** with all the possibilities of edition found in commercial modules.
-A **Macro Option**, which permits the on-line transformation, recoding and editing of data and to perform fully automated survival analysis.
-On Line **Tranformation** of categorical variables into Indicator variables or Deviation Contrast
variables.
-**Graphical plots** for all functions, which may be saved as Pict files for further processing, printed or converted to a photographic slide or transparency for projection.

THE OUTPUT OF THE PROGRAM INCLUDES

• PARAMETERS OF THE SURVIVAL MODEL •

- Estimated maximum likelihood coefficients for the variables in the model.
- Estimated standardized coefficients.
- Chi-Square Test (Wald) on the statistical significance of the estimated coefficients, with
exact probabilities.
- Odds-Ratio (exp. of coefficient).
- First Derivative of the objective function at convergence.
- Global Chi-Square Test.

- Likelihood Ratio Test.
- Wald Test.
- Estimated asymptotic covariance matrix.
- Basal Survival Function (Kaplan-Meier) and Survival Function for a given set of covariates. It also computes density function, hazard function and the standard deviation for the Kaplan-Meier estimate.

• GRAPHICAL OUTPUT (GRAY OR COLOR) •

- Basal (Kaplan-Meier) and Model survival function plots
- Log of -Log of Survival function. Provides a graphical test for proportionality.
- Residual Plots. Provides a graphical test for goodness of fit of data to model.
- I.C. (95% and 99%) plots of the cumulative survival function.
- Plot of Hazard, Density and -Log of Survival Function vs. Time and Log Time. It helps to assess the fitting of observed data to parametric models. A future version of the program will include the option for fitting data to different parametric models (Weibul, Exp, etc).

•• DESCRIPTION OF MENU COMMANDS ••

•• <u>FILE MENU</u> ••

• **NEW** Creates a new Text window than can be edited and saved as a Text File

• **OPEN** Opens a Text file for Editing

• **GET DATA.** You are presented a custom open dialog box with two options:

• **TEXT FILE.** Data to be analyzed is in a text file (may be exported from another application) where each column represents a variable and each row an observation or case. Column fields may be delimited by tabs or commas (,).

• **APPLICATION FILE**. Data is in a binary file created with the SAVE DATA OPTION as APPLICATION FILE of the FILE MENU.

• **CLOSE**. Closes the selected window. If the window is a Text window and you have made changes, you are requested to save the changes with a dialog box.

• **SAVE AS.** Save the content of Text Windows. If the selected window is a Graphical one, a dialog box with an error message is showed. To save graphs you should use the SAVE GRAPH PLOT from the GRAPH MENU

• **SAVE DATA .** Like the GET DATA option, you may save the data loaded in the active output window as a Tab delimited <u>Text File</u> or a Binary <u>Application File</u>. Applications files are saved with a .surv sufix (file.surv)

• **PAGE SETUP** . Shows a dialog box with the printing options for the selected printer.

• **PRINT.** Prints the content of the active Text Window or Plot Window. Notice that plots of graphical windows saved as PICT files may be edited with different graphic programs that accept this format (MacDraw, SuperPaint, Canvas, etc.).

• **HELP.** Shows the Help window.

• **QUIT.** Closes the application. If you have modified your text windows,

you are requested to save the changes with a dialog box before quitting.

•• EDIT MENU ••

Needs no further explanation if you are a Mac User (for PC users converted to the faith, read any Macintosh Manual)

•• MODEL MENU ••

• **DEFINE MODEL.** When you select this submenu item a dialog box (the big one of Fig. 1) is shown, where you must select the different options of the test. If you don't have an active output window with loaded data, you are warned with an error message.On the right side of the dialog box there's a set of different **CONTROL PARAMETERS:**

•*Status*. Type here the column number of the data file where the status variable is located. Defaults to the first column.

• *Time*. Type here the column number of the data file where survival times are located. Defaults to the second column. We recommend to use days as the basic Time Unit in your data file.

• *CutPoint*. If you want to make an analysis dividing the total data in two groups by a given cut-point value of a variable type this value followed by the column number where the variable is located in your data file (VAR box).

• *Group*. For stratified analysis, type the maximum number of strata followed by the column number where the group variable is located in your data file (VAR box).

• *Fail Code*. You should usually code the status of your observations as 1 for failed and as 0 for censored. If you have different code numbers for this variable, type the number value corresponding to the end of the event to be analyzed. Internally, the program transforms this value to 1's and the other values to 0,s.

• *Iterations*. Maximum number of iterations for the Newton-Raphson algorithm. Defaults to 15. In most cases, 4 or 5 iterations are enough to achieve convergence. If no convergence is achivied you should carefully check your data for strong colineality between selected variables in the model, which may result in a singular or nearly singular matrix of second partial derivatives of the likelihood function.

• *Precision*.. Level of precision wanted in the estimation of the coefficients of the Model. Defaults to 0.000001

•On the left side of the dialog Box there are two boxes. The one on the left, lists in a scrolling window (as VAR1, VAR2, etc.) the variables in the data file that were loaded with the GET DATA Command. To Select the variables that should be included in a given model, double click with the mouse on the corresponding variable. The selected variable will be shown on the box to the right. If you want to tell the program that the selected variable should be treated as a TIME DEPENDENT covariate, press first the OPTION key on the Keyboard. The variable shows with an asterisk (*) on its rightmost side. You may combine in a model both fixed and time dependent covariates. If you are not happy about the variables you selected, you may clear the variable's box by selecting with the mouse the CLEAR button. If you don't select any variable a **Kaplan-Meier** product-limit estimate will be computed. This estimator is used as an arbitrary basal function to assess the effect of the covariates included in the survival model upon death rates.

• Other options in the dialog box control: a) the survival *Time Units (days, weeks, month, years, user)*; b) the output of different *Survival and Hazard* functions; c)*Graphical Plots* and; d)*Labeling* of the selected variables. Notice that if you choose the time unit to be in weeks, months or years, it is supposed that survival times have been recorded in days for any subsequent transformation. Otherwise you hould use your original time unit with the DAY or USER box selected in the TIME UNITS pop-up menu.

• You may also enter the value of the coefficients for estimation of the survival function -provided they have been previously estimated- by checking the *User Coefficients* radio button. Internal estimation of MLE's is ommited.

• Select the *User Covariate Pattern* to estimate the survival function for a given covariate set. When you check this option, a dialog box appears requesting the values for the variables included in the model.

. •*FIT to EXPONENTIAL DATA* : This option will be available in a future version.

• ESTIMATE

This options starts the computation of the different parameters and functions for the model. Progression of the Newton-Raphson procedure for estimation of MLE's is shown on the active output window. If you have not previously selected the DEFINE MODEL option you're warned with a message.

After computation, a plot window with a graphical output of survival function is shown if you checked that option. If you push the mouse button inside the graph content, a cross cursor and a box with the probability of survival against time is shown at the lower left corner of the window. If you press simultaneously the Option Key this values remain as a part of the graph plot; otherwise it is cleared after you release the mouse button.

If you press the shift key while pushing the mouse button, a green cross sign may be placed in the graphic plot at the window coordinates where the mouse button is released. It servs to mark some points of interest in the graph, e.g., the median of survival.

• CLEAR WINDOW

Clears the content of the output window. We recommend the use of this option after each analysis is completed, since texts longer than 4000 character slows down the process. First save the numerical results of the previous TEST with the SAVE AS option of the FILE MENU.

• FORMAT.

Shows a dialog box where you can edit the length and the number of digits after the decimal point for both numerical output and data listing.

• OUTPUT OPTIONS.

This is a pop-up hierarchical menu with three different options.

• SAVE MODEL. Saves the Kaplan-Meier estimate and the estimated coefficients of the Model that will be used with the SURVIVAL CHART option.
• SAVE BASAL FUNCTIONS. Saves all Baseline Survival Functions.
• SAVE MODEL FUNCTIONS. Saves all Survival Functions for a given user-defined survival model.

• TRANSFORM.

This option internally transforms categorical variables into either Indicator variables or variables coded according to the Deviation Contrast scheme. The category of reference is by the default the first one (1) .Should you want another category as reference, put the category number in the text box before double-clicking on the corresponding variable in the variable's window. After transformation the new values and Var numbers of the transformed variables and the new Var number of the remaining variables are listed in the output window.

• LIST .

List the variables of the active data file in the active output window.

• SURVIVAL CHART.

This option graphically estimates the probability of survival for a patient based on: a) a set of coefficients; b) an user-defined basal survival function and; c) a given covariate pattern. You are requested to enter the value for the different covariates . The program first loads the basal survival function and the value of the coefficients of a previously saved survival model. An error message is shown if no model has been saved or of you have any application Data or Plot window opened.

•TEST OF HYPOTHESIS.

Performs a Wald Test for a subset of variables in the model. It is used to assess the statistical significance of a group o variables while keeping constant the effect of the remaining variables, e.g., when you codify a categorical variable in k-1 Indicator variables and want to assess the joint effect of the different categories for that variable.

•• GRAPH MENU••

• WHITE BACKGROUND. Toggles the background color of the Plot window
to white . Foreground colors are changed accordingly. The active

background color
is shown with a check mark.
• WHITE BACKGROUND. Toggles the background color of the Plot window
to black . Foreground colors are changed accordingly. The active background color

is shown with a check mark.

• SAVE GRAPH PLOT. Saves the active Plot Window in a standard PICT II format.

• SEPARATE STRATA. Should you perform a stratified analysis, this option shows the survival curve for each strata in a different plot window.

• RESIDUALS PLOT. Plot of Cumulative Hazard function of Residuals against residual itself. Used to assess the goodness of fit of data to model.

• INTERVALS PLOT [95% and 99%]. Plots of 95% and 99% confidence interval limits for the Kaplan-Meier product-limit estimate.

• LOG AND LIN FUNCTIONS PLOT. Currently used to asses the fit of your survivl data to parametric models.This option genarate plots of Hazard, cumulative Hazard and -Log of Survival function against Time and log Time. See Cox's 'Analysis of Survival Data' book for further references.

At the end of the Graph Menu items all plots windows created by the program are listed are listed. You may select any graph plot and make it the active window with the mouse.

•• MACRO MENU ••

This option permits the use of standard PASCAL procedures to internally compute the value of Time Dependent Covariates and to transform, select, delete or recode the data loaded with the GET DATA OPTION of the FILE MENU.

With version 6.0 a new set of macro commands have been added to allow the user to perform automated survival analysis. Examples for the use of these commands within a standard Pascal procedure can be found in the TestMacro.macro file in the Macros folder.

a) Standard Pascal Commands:

-Procedure
-Begin...end
-For ...To...Do
-Repeat Until
-While...Do
-If...Then...Else
-Var

b) Operators:

{, },+, -, *, /, DIV, MOD, :=, =, >=, <=, < >, =

c) Logical Operators:

-AND
-OR
-NOT

d) Pascal Types:

-Integer; internally stored in extended precision
-Real; internally stored in extended precision
-Boolean: true,false

e) Pascal Standard Functions

-trunc(x)	converts real (x) to integer with truncation
-round(x)	converts x to integer with rounding
-random	generates a random number between 0 and 1
-odd(x)	returns true if integer x is odd
-abs(x)	absolute value of x
-sqr(x);	square of x
-sqrt(x)	square root of x
-sin(x)	sine of x

-cos(x) cosine of x

-arctan(x) arctangent of x
-ln(x) natural logarithm of x
-exp(x) exponential of x

f) Custom Commands: This command allows the user to perform fully automated survival analysis and to transform or recode the variables in the active data array. See
the examples in the MacroTest Macro File to learn how to use this commands.

-BACKCOLOR (*'white', 'black'*) sets the background color for graphical plots. You must put one of the arguments in the list above; otherwise you'll get an error message. the Type is STRING. Use 'negro' and 'blanco' for the Spanish version of the program.

-BEEP; beeps for one second.

-COMPUTE. This command is used to recode or transform variables in a speedy fashion. It
has the following syntax's:

COMPUTE
statement ;
statement ;
etc. ;
END;

See the examples in the MacroTests macro file to learn how to use this command.

-COUNTIF(*Boolean expresion*) count the number of cases in the active data array that match the conditions specified in the Boolean argument. Use the rDATA[ ] token to access the variables in the active data array. The result of the count is shown in the active Output Window.
Example:

COUNTIF((rData[1] > 50) **and** (rData[2] =3));

-COVARIATES (*number, number* ....) set covariate values for the current survival test. The Type is REAL.

-CREATE (*'filename', Var, Var, Var* ...) . Creates a new Filename *'Filename'* with the variables specified in the argument list. Example CREATE ('MyNewFile', 1,2,3,7,8);.

-FUNCTION . This command has the following syntax's:

FUNCTION
statement ;
statement ;
etc. ;
END;

It must be used only to compute the value of the variables at each failure point when you select Time Dependent Analysis. See examples in the MacroTime1 and MacroTime2 macro files to learn how to use this command.

-ESTIMATE; starts estimation of Mle's and survival functions. It has no arguments. You must always specify the column number of the status variable and time variable with SETSTATUS and SETTIME before issuing this command; otherwise you'll get an error message.

-EXIT; terminates macro execution.

-GET (*Rows, Columns, nVars*); gets the number of rows, number of columns and the number of selected vars of the active data array. The Type is INTEGER;

-GETDATA (*'filename'*) loads data form a binary file or a tab delimited text file. If the data file is in a different folder or volume you must specify the full path to access the file. If not, a dialog box is shown. The TYPE is STRING.

-GRAPHPLOT (*true, false*) ; this command enables (true) or disables (false) graphical Plots. The Type is BOOLEAN.

-LIST (*length, decimals*); Updates the output windowand lists data with the format given by the two arguments. The Type is INTEGER

-MACRO; command needed to identify a Macro procedure. The text in quotes after the word MACRO is added as a new command in the Macro Menu.

-MAKEBINS (*VarIn,VarOut,Option,Bin1,Bin2,Bin3 ..... Bin_{Max}*). This command recodes a numerical variable into categories. *VarIn* is the source column number. *VarOut* specifies the column number where to put the newly coded variable. If *VarOut* equals to *VarIn ,* all data in the source column will be overwritten. *Option* is a Boolean variable; when is set to **true** the first category of the new coded variable corresponds to those values of the numerical variables that are less than the value specified in Bin1 and the last category to those values that are equal or greater than the value given in the last Bin. THE MAX NUMBER OF BINS IS 30.

Example 1: MAKEBINS (3,3,**true**, 40,50,60,70) will recode the variable in column 3 and put the result in the same column as follows:

Value	Category Nº
< 40	1
40 - 49	2
50 - 59	3

60 - 69	4
70 >	5

Example 2: MAKEBINS(3,4,**false**,40,50,60,70) will codify the variable in column 3 an put the result in column 4 as follows:

Value	Category Nº
40 - 49	1
50 - 59	2
60 - 69	3

-NEWVARS (*number, default*) . This command appends to the right of the active data array new *number* columns with the *default* value. It is used to create a dummy variable to store the transformed values of other variables. Example NEWVARS(2, 0);

-OMITIF (*Boolean argument*) ; excludes from the active data array the cases that meet the conditions specified in the Boolean argument. Use the rDATA[ ] token to access the variables in the active data array.

-RECODE (*Var,OldValue,NewValue, [Default Value]*) . This comand easily recodes the *OldValue* of the variable number *Var* with NewValue. *Default Value* is optional; if you specify it, the variables with values different to OldValue take its value. Is the same as if you would use the following Pascal statement within a procedure :

'**If** *OldValue* **then** *NewValue* **else** *DefaultValue*'.

You may use either the name of a previously defined variable with the column number you want to transform, the column number itself or a matrix definition rVar[column number]. Example: The following commands will render the same result

```
RECODE(1:1:2;5:1:2:-1);

grade:= 3; stage:= 5;
RECODE(grade:1:2;stage:1:2:-1);

RECODE(rVar[3]:1:2;rVar[5]:1:2:-1);
```

You may use colons or commas as delimiters of the arguments. Use a semicolon as delimtier when you recode different variables with a single command

-RESET; this command set different flags and constants needed for the survival algorithm. It has no arguments and **should be** the first command in a procedure that use macro commands.

-SAVEDATA(*'text' , 'binary'*) saves data in the active data array as a tab delimited text file (*text*) or as a binary file (*binary*). You must put one of the arguments in the list above; otherwise you'll get an error message. Use 'text' and 'binario' for the Spanish version of the program. the Type is STRING.

-SAVEMODEL; Saves the parameters of the current estimated model for further use with the Survival Chart option of the Survival Menu. It has no arguments.

-SAVETEST(*'filename'*) . Saves in the file *'filename'* the numerical results of the current Test in the active Output Window.

-SELECTIF (*Boolean argument*) ; selects from the active data array the cases that meet the conditions specified in the Boolean argument. As with the OMITIF command, use the rDATA[i] token to access the variables in the active data array.

-SELECTVAR(*number, number*, ...); this command selects the variables to be included in the regression model. *Number* specifies the column number of the data file where variables are located. The Type is INTEGER.

-SET (*Rows,Columns,nVars*); sets the new number of Rows (cases), Columns (variables) and the number of selected vars (nVars), if any, for the active data array or the array containing the selected variables for a Test. The Type is INTEGER.

-SETCUTPOINT (*number, varnumber*) sets cut point value (number). *Varnumber* is the column number of the variable. The Type is REAL for the first argument and INTEGER for the second argument.

-SETFAILCODE (*number*) sets the code identifying the terminal event. The Type is INTEGER

-SETITER (*number*) sets number of max iterations. The Type is INTEGER;

-SETLABEL(*'string',string'* ....) puts labels to the selected variables. The Type is STRING

-SETOUTPUT(*'all', 'none', 'basal', model'*) selects numerical output to be shown in the active output window. You must put one of the arguments in the list above; otherwise you'll get an error message. The Type is STRING. Use 'todas', "ninguna', 'basal' and 'modelo' for the Spanish version of the program.

-SETPRECISION(*number*) sets the level of precision for estimation of the coefficients of the regression model. The Type is REAL.

-SETSTATUS(*number*); sets the column number with the status variable in the data array. The Type is INTEGER.

-SETSTRATA (*number, stratavar*) sets the number of strata and the column number of the strata variable (*stratavar*) to perform a stratified analysis . The Type is INTEGER.

-SETTIME(*number*); sets the column number with the survival times in the data array. The Type is INTEGER.

-SETUNIT(*'days', 'weeks', 'months', 'years', 'user'*) selects Time unit for numerical and graphical output. You must put one of the arguments in the list above; otherwise you'll get an error message. the Type is STRING. use 'dias', 'semanas', 'meses', 'años' and' usuario' for the Spanish version of the program.

-WRITE(*'message', number or variable*) writes the message with an optional number or the value of the variable to the active Output Window.
Example:

```
TimeAverage:= 560; (you may compute the value)
WRITE( 'The average survival time is', TimeAverage);
```

g) CUSTOM ARRAYS:

Six different arrays are internally defined for data manipulation;

1) rVAR[k] ; vector of length MaxVars (30). Used to temporarily store the value of the kth selected variable for a given test. You must use this token to set or get the VAR value within the COMPUTE ,FUNCTION, SELECTI, OMITIF, RECODE and COUNTIF custom procedures.

2) rDATA [k**]** ; vector of length MaxVars (30). Used to temporarily store the value of the kth variable in the active data array. You must use this token to set or get the VAR value within the above mentioned custom procedures.

3) gVAR[offset]; the same as **rVAR** but it must be used within a macro or standard Pascal procedure.

4) gDATA[offset**]** ;the same as **rDATA** but it must be used within a macro or standard Pascal procedure.

As Total Data and Selected Data are stored in a one-column array of length (rows * columns) see the examples in the Survival Macros file to learn how to access the ith row of the kth column through the gVAR and gDATA tokens. The Type of these arrays is REAL.

5) rMEAN [k]; vector of REAL Type and MaxVars(30) length which stores the vector of average values for the kth selected variable of the ith individual. Currently used with the Time Dependent option.

6) TEMP [k]; Vector of REAL Type and MaxVars(30) length which stores the code of the kth selected variable for the model of the ith individual or case. This value is = 1 for Fixed Covariates and = 2 for Time Dependent Covariates. You may use **TEMP[k]** for the temporarly storage of any value or variable within a custom or a standard Pascal procedure.

In Version 5.0.2 of the program I gloomy wrote the following paragraph:

"When you use Time Dependent analysis you should get first a big pot of coffee and relax because it takes time to get the final results due to the iterative access to the macro function at each fail point (sorry, but **nobody** is perfect. If you know how to speed up the process, **please** let me know)."

Well, with version 6.0 forget about the coffee because with the use of the custom FUNCTION procedure, things have been speed-up considerably and it takes only a few more seconds to perform a Time Dependent Analysis respect to an analysis using fixed covariates. Carefully read the examples in the TimeMacro 1 and TimeMacro 2 macro files to learn how to use this option. By the way, I fixed the problem by myself and, believe me, it took me hours to find the best solution.

• COMMANDS IN THE MACRO MENU

• EDIT MACRO

Opens a dialog box to select the macro file you want to edit. If you press the Cancel button an empty Macro Window is shown. Use the SAVE command of the FILE MENU to save changes. You're also requested to save changes when you close the window.

• INSERT MACRO

Inserts a New Macro definition at the end of the MACRO MENU. If a macro file has different macro definitions every macro is listed as a different item of the MACRO MENU. To RUN a macro just select its name with the mouse from the MACRO MENU.

To use macros with the Time Dependent Option you should INSERT a macro file with a single macro definition before running the Test; otherwise an error message is shown. (see example)

• DELETE MACRO

Deletes a previously inserted macro from the MACRO MENU.

• RUN MACRO

Executes the macro defined in the active Macro Window or the first macro listed in the MACRO MENU.

•• WINDOW MENU••

-MACRO COMMAND shows a dialog box with a text edit field where you can enter any of the Custom Commands listed before. Use a semmicolon as delimiter if you use different commands in a single string. The max length of the string must be 255 characters. The program adds internally a dummy macro definition so you don't need to open a Macro window to execute the command.

Every time you open a Data Window or a Text Window, its name is appended to the **window menu** from where you can select it. Useful to easily access a given window when you have multiple windows in you desktop.

SURVIVAL IS PUBLIC DOMAIN, SO FEEL FREE TO GIVE A COPY OF THE PROGRAM TO ANY PERSON CONCERNED WITH SURVIVAL ANALYSIS

¡¡ PLEASE REPORT ANY BUG, PROBLEM OR COMMENT TO THE FOLLOWING ADDRESSES !!

1) MANUEL URRUTIA AVISRROR, M.D. P.h. D.
Hospital Universitario de Salamanca
Unidad de Investigación
Paseo San Vicente s/n
37007 SALAMANCA - ESPAÑA

2) MANUEL URRUTIA AVISRROR, M.D. P.h. D.
Calle Perdices 53
Urbanización Valdelagua
37191 SALAMANCA - ESPAÑA

3) email Internet: urrutia@gugu.usal.es

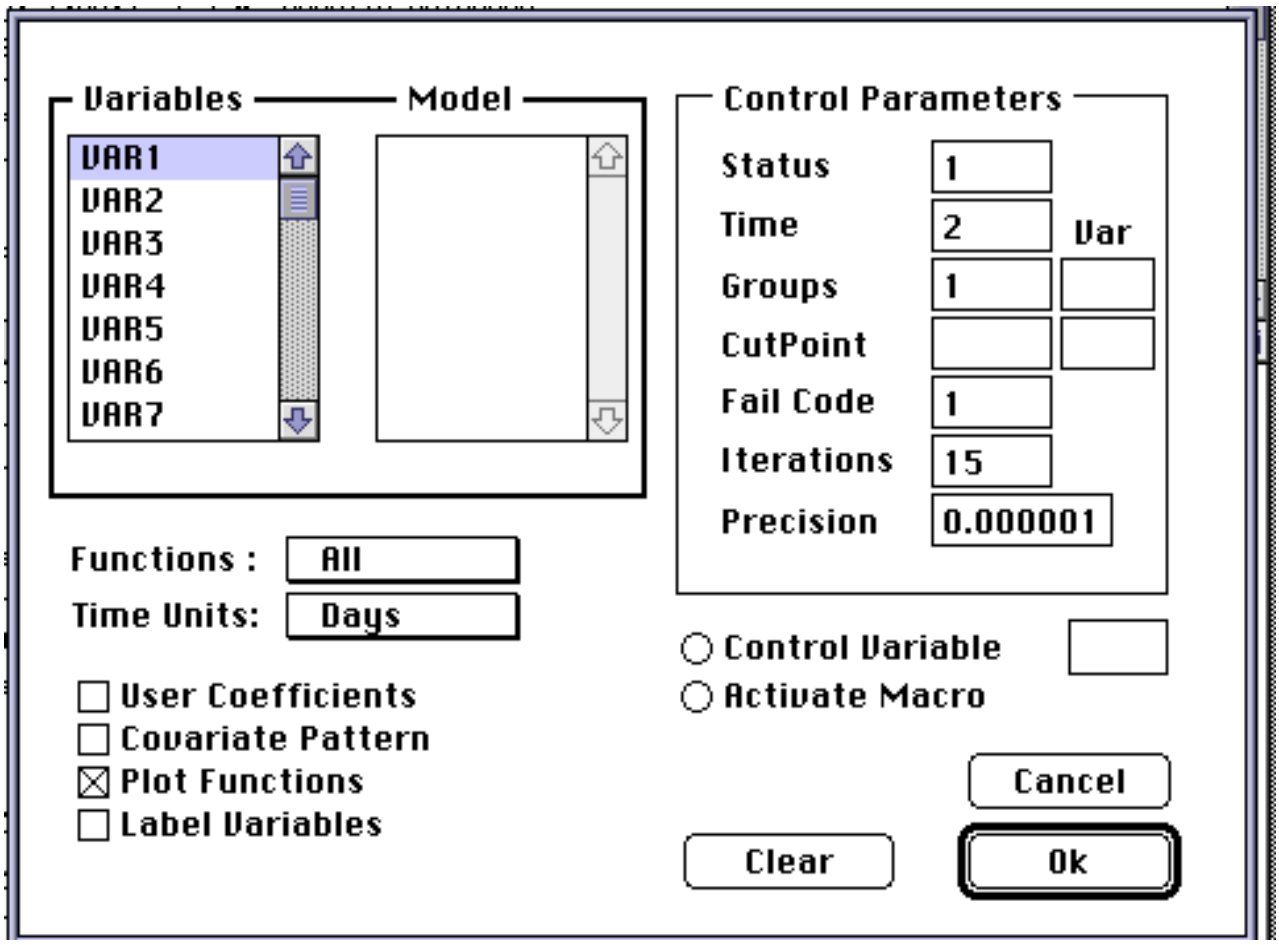
Variables
Model
VAR1
VAR2
VAR3
VAR4
VAR5
VAR6
VAR7
Control Parameters
Status
1
Time
2
Var
Groups
1
CutPoint
Fail Code
1
Iterations
15
Precision
0.000001
Functions :
All
Time Units:
Days
Control Variable
Activate Macro
User Coefficients
Covariate Pattern
Plot Functions
Label Variables
Cancel
Clear
Ok

Figure 1