

Chapter 1

Getting Started

All data analyses begin with raw data in one form or another. In LISREL one can work with data in plain text (ASCII) form. But for most analysis with LISREL it is convenient to work with a LISREL **data system file** of the type **.lsf**. LISREL can import data from many formats such as SAS, SPSS, STATA, and EXCEL. LISREL can also import data in text format with spaces (*.dat or *.raw), commas (*.csv) or tab characters (*.txt) as delimiters between entries. The data is then stored as a LISREL data system file **.lsf**.

Section 1.1 describes how to create a LISREL data system file **.lsf** by data importation and how to set the attributes of the variables in the data, for example, whether a variable is categorical or continuous.

Sometimes the data can be split into several groups. Section 1.3 shows how to create different **.lsf** files for each group.

Section 1.6 shows how to define missing values in the **.lsf** file and some of the ways these can be handled.

It is often necessary to transform variables and/or construct new variables. Section 1.7 describes how this is done.

It should be emphasized that this chapter is *not* about modeling. It is about how to prepare the data for modeling. John Tukey once wrote “It is important to understand what you *can do* before you learn to measure how *well* you seem to have *done* it.” (Tukey, 1977). It is with this spirit in mind we suggest that a fair amount of data screening should be done before the data is submitted to modeling. This includes the understanding of the characteristics of the distribution of the variables and the distribution of missing values over variables and cases. Graphs play an important role in this and Section 1.2 describes some of the graphs that LISREL can produce.

1.1 Importing Data

This section illustrates how to import data and create a LISREL data system file **.lsf**. Most examples in later chapters of this book are based on LISREL data system files that have already been created.

To illustrate data importation we use a classic data originally published by Holzinger and Swineford (1939). They collected data on twenty-six psychological tests administered to seventh- and eighth-grade children in two schools in Chicago: the Pasteur School and the Grant-White School. The Pasteur

School had students whose parents had immigrated from Europe, mostly France and Germany. The students of the Grant-White School came from middle income American white families.

Nine of these tests are selected for this example. The nine tests are (with the original variable number in parenthesis):

VIS PERC Visual Perception (V1)

CUBES Cubes (V2)

LOZENGES Lozenges (V4)

PAR COMP Paragraph Comprehension (V6)

SEN COMP Sentence Completion (V7)

WORDMEAN Word meaning (V9)

ADDITION Addition (V10)

COUNTDOT Counting dots (V12)

S-C CAPS Straight-curved capitals (V13)

The nine test scores are preceded by four variables:

SCHOOL 0 for Pasteur School; 1 for Grant-White School

GENDER 0 for Boy; 1 for Girl

AGEYEAR Age in years

BIRTHMON Birthmonth: 1 = January, 2 = February ... 12 = December

The data file is an SPSS data file **hsschools.sav**. This file contains only the names of the variables and the data values. There are no missing values in this data set. Missing values are discussed in Section 1.6.

Using this data set we can learn

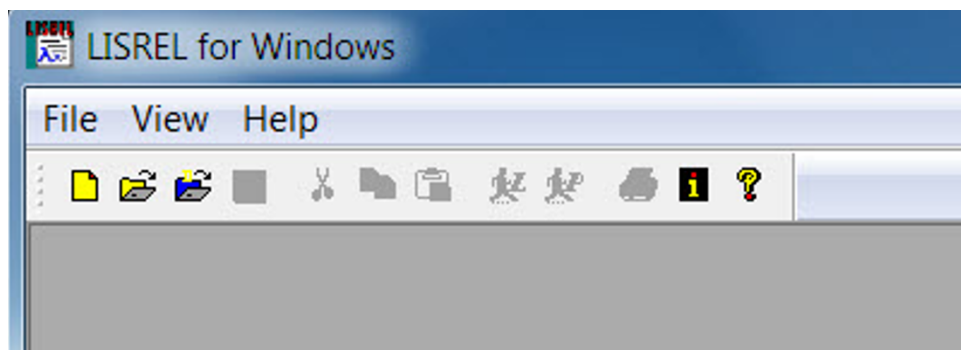
1. how to import data into a LISREL system data file **.lsf**
2. how to graph the data
3. how to divide the data into two groups (schools)

In later chapters we will use these data to illustrate

1. how to do exploratory factor analysis in one school
2. how to do confirmatory factor analysis in the other school
3. how to deal with non-normality
4. how to modify the model by means of path diagrams

5. how to interpret chi-square differences
6. how to estimate and test latent variable differences

When LISREL starts it opens with a screen like this:



Select **File** and then **Import Data**. In the **Files of Type** select **SPSS Data File (.sav)** and browse your computer until you find the file **hsschools.sav**. Double click on this file name and accept the suggested file name **hsschools.lsf**. LISREL will then open this which looks like this, showing the first 15 rows of data, *i.e.* the first 15 cases.

	SCHOOL	GENDER	AGEYEAR	BIRTHMON	VISPERC	CUBES	LOZENGES	PARCOMP	SENCOMP	WORDMEAN	ADDITION	COUNTDOT	SCCAPS
1	0.000	0.000	13.000	2.000	20.000	31.000	3.000	7.000	23.000	9.000	78.000	115.000	229.000
2	0.000	1.000	13.000	8.000	32.000	21.000	17.000	5.000	12.000	9.000	87.000	125.000	285.000
3	0.000	1.000	13.000	2.000	27.000	21.000	15.000	3.000	7.000	3.000	75.000	78.000	159.000
4	0.000	0.000	13.000	3.000	32.000	31.000	24.000	8.000	18.000	17.000	69.000	106.000	175.000
5	0.000	1.000	12.000	3.000	29.000	19.000	7.000	8.000	16.000	18.000	85.000	126.000	213.000
6	0.000	1.000	14.000	2.000	32.000	20.000	18.000	3.000	12.000	6.000	100.000	133.000	270.000
7	0.000	0.000	12.000	2.000	17.000	24.000	8.000	10.000	24.000	20.000	108.000	124.000	175.000
8	0.000	1.000	12.000	3.000	34.000	25.000	15.000	11.000	17.000	9.000	78.000	103.000	132.000
9	0.000	1.000	13.000	1.000	27.000	23.000	12.000	8.000	23.000	19.000	104.000	93.000	265.000
10	0.000	1.000	12.000	6.000	21.000	21.000	6.000	8.000	20.000	18.000	95.000	91.000	157.000
11	0.000	0.000	12.000	3.000	22.000	23.000	16.000	6.000	14.000	11.000	86.000	114.000	155.000
12	0.000	0.000	12.000	12.000	35.000	24.000	23.000	8.000	18.000	19.000	85.000	103.000	149.000
13	0.000	1.000	12.000	8.000	34.000	18.000	33.000	8.000	16.000	16.000	135.000	104.000	211.000
14	0.000	1.000	12.000	9.000	36.000	22.000	14.000	14.000	16.000	11.000	118.000	94.000	160.000
15	0.000	0.000	12.000	7.000	35.000	23.000	29.000	15.000	22.000	21.000	92.000	87.000	211.000

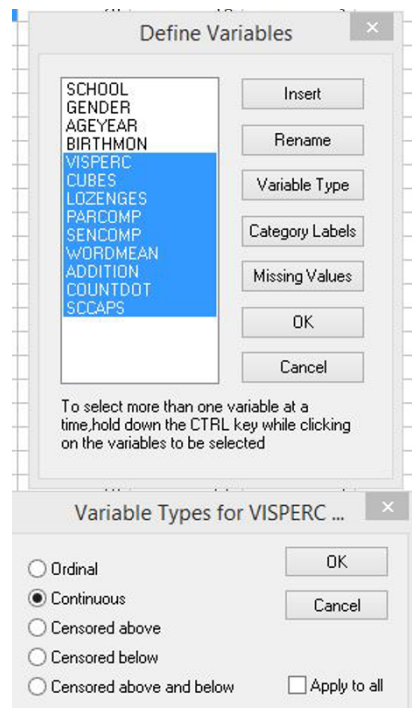
Variable Types

LISREL has essentially two types of variables called ordinal and continuous. The term ordinal variable is used here in the sense of a categorical variable, *i.e.*, a variable that has a only few distinct values (maximum 15), whereas continuous variables typically have a wide range of values¹. LISREL assumes that all variables are ordinal (categorical) by default. In this example the first four variables SCHOOL, GENDER, AGEYEAR and BIRTHMONTH are considered as ordinal (categorical) and the nine test scores VISPERC – SCCAPS as continuous. The latter variables must therefore be declared continuous in the **.lsf** file²

To do this, select **Data** in **LISREL System Data File Toolbar** when the **.lsf** file is displayed and select the **Define Variables** option. Then select the nine test score variables. Then click on **Variable Type**, select **Continuous** and click on the **OK** button twice and save the **.lsf** file as shown here.

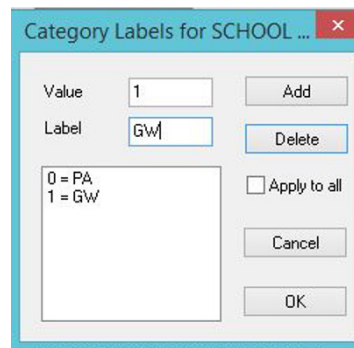
¹In Sections 7.4.1 and 7.4.2 in Chapter 7 we will learn about the more common use of ordinal variables where the observed variables are Likert scales with ordered categories

²There are also many situations where a binary variable should be treated as continuous. For example, in regression we may want to include a binary (dummy) variable as an independent variable. This variable must therefore be declared continuous in the **.lsf** file although it is categorical, see Chapter 2.



Category Labels

For the categorical variables it is convenient to define labels for the categories. These labels are limited to four characters. To do this, select **Data** in **LISREL System Data File Toolbar** when the **.lsf** file is displayed and select the **Define Variables** option. Then click on **Category Labels**, select the variable **SCHOOL**. Type 0 in the Value field and type PA in the Labels field and click on **Add**. Then type 1 in the Value field and type GW in the Labels field and click on **Add**. Then click **OK** button twice and save the **.lsf** file as shown here.



In the same way one can define category labels for **GENDER**.

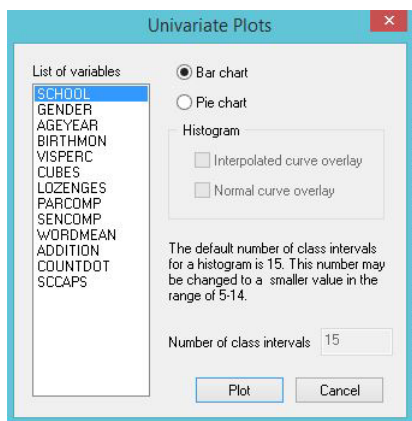
1.2 Graphs

There are three types of graphs available in **LISREL** and within each category there are several types of graphs depending on whether the variables in the graphs are ordinal or continuous.

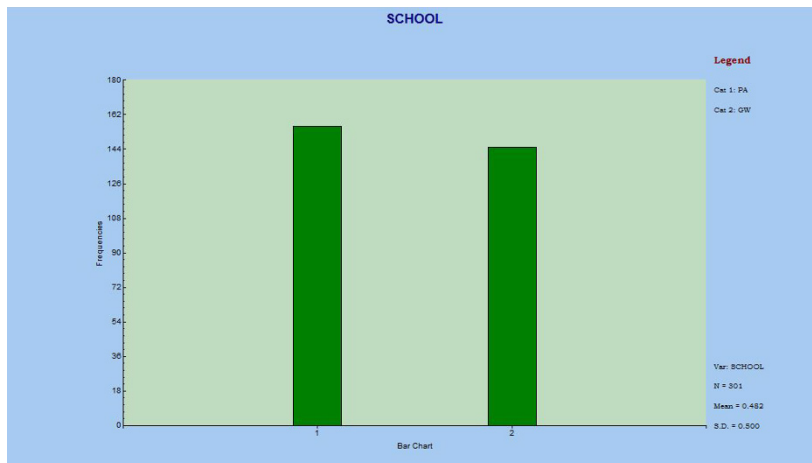
- Univariate Graphs
 - Bar Chart
 - Histogram
- Bivariate Graphs
 - 3D Bar Chart
 - Scatter Plot
 - Box and Whisker Plot
- Multivariate Graphs

We describe the univariate and bivariate graphs here.

To obtain a univariate graph of an ordinal variable, **leave all variables in the .lsf file unselected** and select Graphs in the **LISREL System Data File Toolbar**, then select the variable, for example SCHOOL, in the .lsf file, and select **Bar Chart** in the following screen

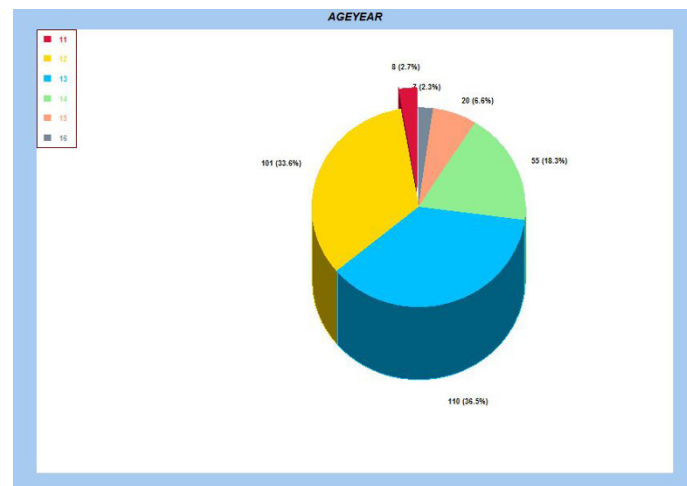


Click on **Plot**. This gives the following bar chart



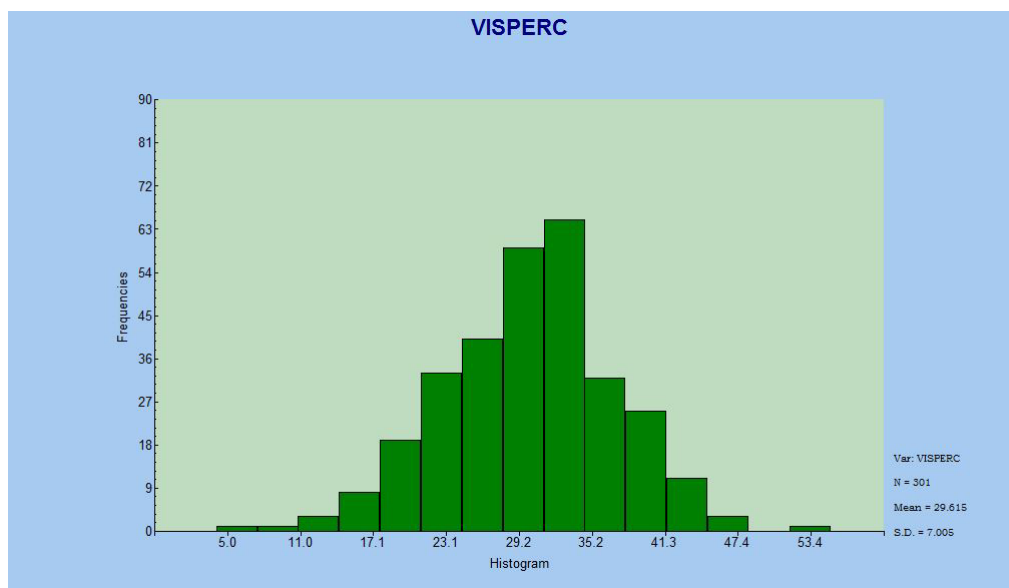
It is seen that the Pasteur school has approximately 156 students and the Grant-White school has approximately 145 students.

Alternatively, one can choose a pie chart in the **Univariate Plots** screen above. A pie chart of AGEYEAR, for example, gives the following pie chart

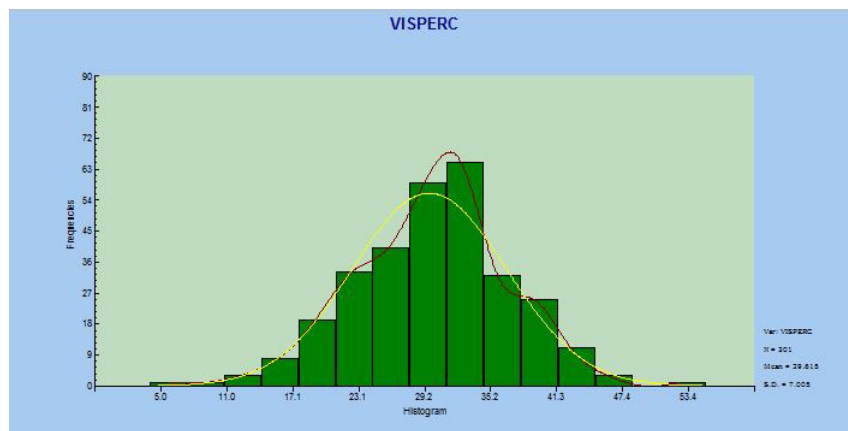


where we can see that most of the students are 12 or 13 years old and very few are 11 and 16 years.

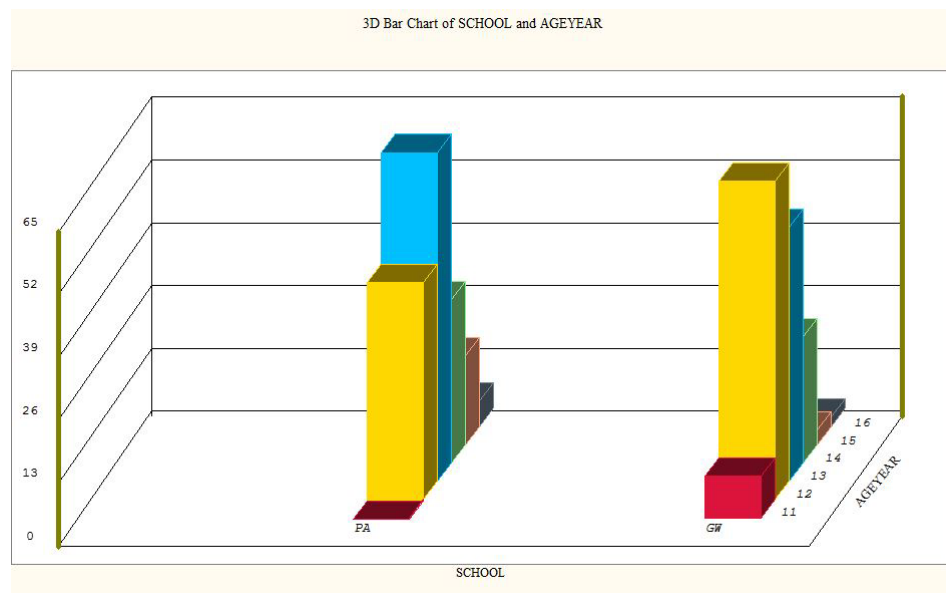
To obtain a univariate graph of a continuous variable, **do not** select the variable in the **.lsf** file first, but select **Graphs** in the **LISREL System Data File Toolbar**, then select the variable, for example **VISPERC**, and click on **Plot**. This gives the following **Histogram**.



This shows that the variable is slightly skewed on the right side but not remarkable so. Later we will learn how to measure and evaluate skewness. In the **Univariate Plots** screen one can also choose to obtain an overlaid fitted normal distribution and/or an overlaid curve fitted to the middle of each bar in the histogram, as shown here

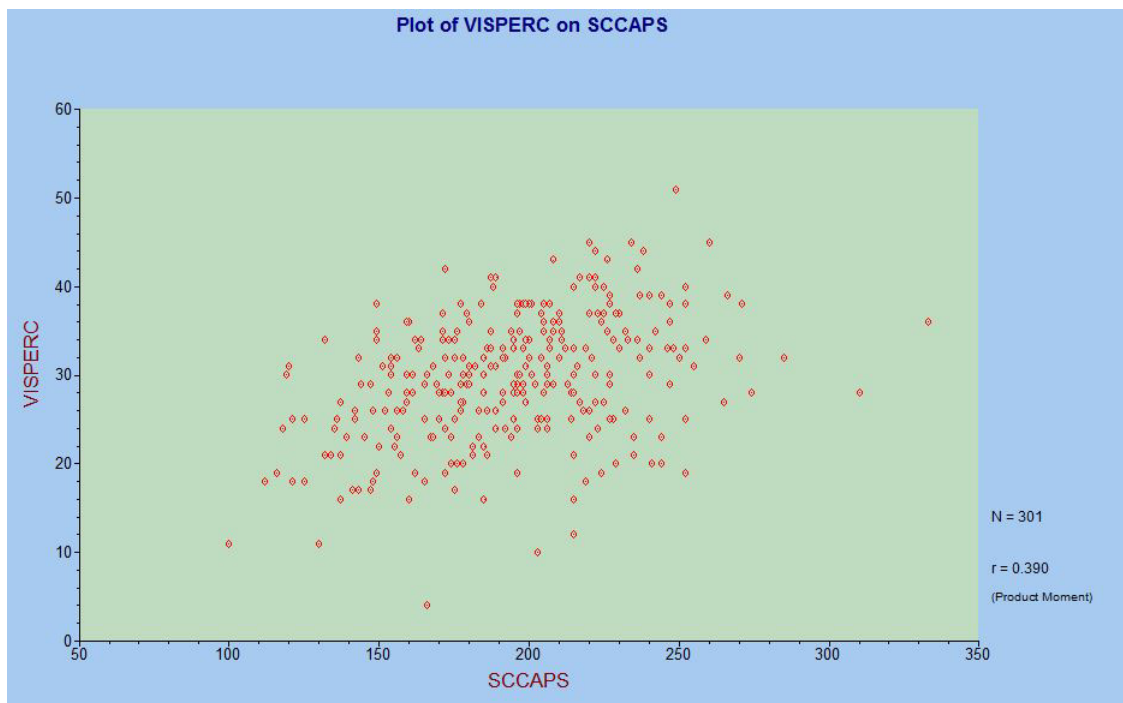


To obtain a bivariate plot with two categorical variables, select **Graphs** in the **LISREL System Data File Toolbar** and **Bivariate Graph** in the **Graphs** menu. Then select AGEYEAR as the independent variable (X variable) and SCHOOL as the dependent variable (Y variable). This gives a **3D Bar Chart** as follows



It is seen that most students in the Pasteur schools are 14 years old, whereas in the Grant-White school most students are 13 years old. In general, students in the Pasteur school tend to be older than the students in the Grant-White school.

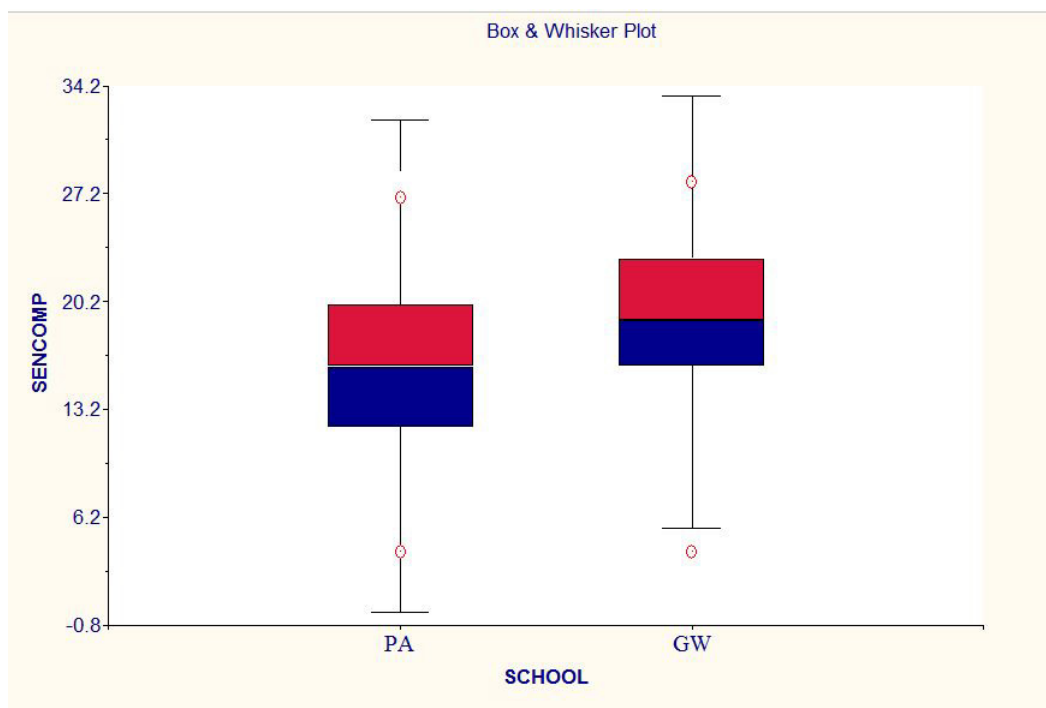
To obtain a bivariate plot with two continuous variables, select **Graphs** in the **LISREL System Data File Toolbar** and **Bivariate Graph** in the **Graphs** menu. Then select VISPERC as the dependent variable, *i.e.*, the variable on the vertical axis, and SCCAPS as the independent variable, *i.e.*, the variable on the horizontal axis. This gives a **Scatter Plot** as follows



From this scatter plot it seems that there is no clear relationship between VISPERC and SCCAPS. It will be difficult to predict one variable from the other. The correlation is given in the lower right side of the plot as $r = 0.390$ based on a sample size of 301. This is Pearson correlation or product moment correlations. Later we will learn to evaluate the statistical significance of this and other types of correlations.

In many examples the so called Box and Whisker plot is very useful because it gives much detailed information about the conditional distribution of a continuous variable for given values of a categorical variable.

To obtain a bivariate plot with one continuous variable, here SENCOMP, and a categorical variable, here SCHOOL, select **Graphs** in the **LISREL System Data File Toolbar** and **Bivariate Graph** in the **Graphs** menu. Then select SENCOMP as the dependent variable, *i.e.*, the variable on the vertical axis, and SCHOOL as the independent variable, *i.e.*, the variable on the horizontal axis. This gives a **Box & Whisker Plot** as follows



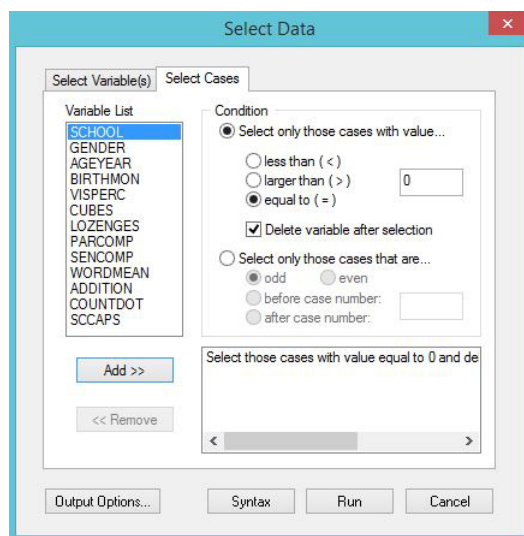
The box indicates the area between lower and upper quartile so that 25% of the observations are below the bottom border of the box and 25% of the observations are above the top border of the box. The border line between the two colored areas in the box is the mean. If the areas above and below the mean are of different sizes, this indicates that the distribution of the observations is skewed. The two unfilled circles are the largest and the smallest value in the data. The whiskers are located $\pm 1.5 \times IQR$, where IQR is the interquartile range, *i.e.* the distance between the upper and lower quartile which is the length of the box. Observations that are outside of the whiskers are sometimes called outliers.

The most striking feature of the Box and Whisker plot for SENCOMP is that the students in the Grant-White school have higher scores than the students in the Pasteur school. In particular, the mean score is larger for the Grant-White school than the Pasteur school. In Chapter 2 we will learn how to test the statistical significance of this difference.

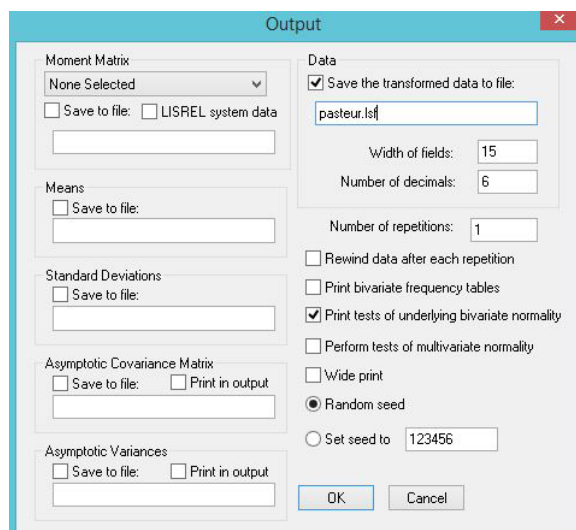
1.3 Splitting the Data into Two Groups

In view of what we already know about the two schools it may be a good idea to split the data into two groups, one for the Pasteur school and one for the Grant-White school, rather than treating the two schools as one homogeneous group. This section illustrates how this can be done.

With the `hsschools.lsf` displayed select **Data** in the **LISREL System Data File Toolbar** and **Select Variables/Cases**. Then select **Select Cases**. Tick **Select Only Those Cases with** and tick **Equal To** and type 0 in the numeric field. Since all values in the selected school will have 0 value on the variable SCHOOL, this variable may well be deleted after selection. Therefore tick the box to do this. Then click on **Add**. The screen then looks like this:



Then select **Output Options**. In the **Output** box, make sure it says **None Selected** under **Moment Matrix**. Tick the box **Save the transformed data to file** and fill in **pasteur.lsf** in the text box:



Click **OK**. Then click on **Run**. This produces a data file **pasteur.lsf**, where the first 15 rows of data looks like this:

	GENDER	AGEYEAR	BIRTHMON	VISPERC	CUBES	LOZENGES	PARCOMP	SENCOMP	WORDMEAN	ADDITION	COUNTDOT	SCCAPS
1	0.	13.	2.	20.	31.	3.	7.	23.	9.	78.	115.	229.
2	1.	13.	8.	32.	21.	17.	5.	12.	9.	87.	125.	285.
3	1.	13.	2.	27.	21.	15.	3.	7.	3.	75.	78.	159.
4	0.	13.	3.	32.	31.	24.	8.	18.	17.	69.	106.	175.
5	1.	12.	3.	29.	19.	7.	8.	16.	18.	85.	126.	213.
6	1.	14.	2.	32.	20.	18.	3.	12.	6.	100.	133.	270.
7	0.	12.	2.	17.	24.	8.	10.	24.	20.	108.	124.	175.
8	1.	12.	3.	34.	25.	15.	11.	17.	9.	78.	103.	132.
9	1.	13.	1.	27.	23.	12.	8.	23.	19.	104.	93.	265.
10	1.	12.	6.	21.	21.	6.	8.	20.	18.	95.	91.	157.
11	0.	12.	3.	22.	23.	16.	6.	14.	11.	86.	114.	155.
12	0.	12.	12.	35.	24.	23.	8.	18.	19.	85.	103.	149.
13	1.	12.	8.	34.	18.	33.	8.	16.	16.	135.	104.	211.
14	1.	12.	9.	36.	22.	14.	14.	16.	11.	118.	94.	160.
15	0.	12.	7.	35.	23.	29.	15.	22.	21.	92.	87.	211.

Follow the same procedure to obtain a data file **grantwhite.lsf**, where the first 15 rows of data look like this:

	GENDER	AGEYEAR	BIRTHMON	VISPERC	CUBES	OZENGES	PARCOMP	SENCOMP	/ORDMEAN	ADDITION	/OUNTD01	SCCAPS
1	0	13	1	23	19	4	10	17	10	69	82	156
2	1	11	11	33	22	17	8	17	10	65	98	195
3	0	12	7	34	24	22	11	19	19	50	86	228
4	0	11	12	29	23	9	9	19	11	114	103	144
5	0	12	6	16	25	10	8	25	24	112	122	160
6	1	12	7	30	25	20	10	23	18	94	113	201
7	1	12	9	36	33	36	17	25	41	129	139	333
8	1	11	12	28	25	9	10	18	11	96	95	174
9	1	12	6	30	25	11	11	21	8	103	114	197
10	1	12	6	20	25	6	9	21	16	89	101	178
11	0	12	1	27	26	6	10	16	13	88	107	137
12	0	12	11	32	21	8	1	7	11	103	136	154
13	0	12	10	38	31	12	10	11	14	83	108	201
14	1	12	9	17	21	6	5	10	10	99	87	147
15	0	12	10	34	28	24	14	22	26	49	84	171

These data files will be used in several later chapters in this book.

1.4 Introduction to LISREL Syntaxes

LISREL comes with several program modules, the most important ones being PRELIS and LISREL. The LISREL module is used for structural equation modeling (SEM), including confirmatory factor analysis for continuous and ordinal variables, models for relationships between latent variables, multiple group analysis, and general covariance structures. For most other purposes, one can use the PRELIS module.

PRELIS reads raw data and can be used for

- Data screening
- Data exploration
- Data summarization
- Computation of
 - Covariance matrix
 - Correlation matrix
 - Moment matrix
 - Augmented moment matrix
 - Asymptotic covariance matrix

With LISREL 9 it is not necessary to use PRELIS to compute these matrices as PRELIS is called from within LISREL for these purposes. PRELIS can also be used to create a LISREL data system file `.lsf`.

PRELIS is also used for various statistical analysis such as

- Multiple univariate and multivariate regression
- Logistic and probit regression
- Censored regression

described in Chapter 2, for principle components analysis described in Chapter 5 and for exploratory factor analysis described in Chapter 6.

This book presents multivariate statistical analysis as it can be done using simple syntax files. For this purpose, there are three syntaxes: **PRELIS** syntax, **SIMPLIS** syntax and **LISREL** syntax. These different syntaxes are introduced as we move along. For now, we give a brief introduction to the **PRELIS** syntax and demonstrate how one can do all the data descriptions in the previous section by a few simple **PRELIS** syntax commands.

The starting point is the data file **hsschools.lsf** created in Section 1.1. Here we show how the files **pasteur.lsf** and **grantwhite.lsf** can be obtained for each school and at the same time get information about the important characteristics of the data in simple tabular form.

To generate the file **pasteur.lsf**, use the following **PRELIS** syntax file **hsschools1.prl**:

```
!Creating a Data File for Pasteur School
SY=hsschools.lsf
SD SCHOOL = 0
OU RA=pasteur.lsf
```

The first line is a title line. One can include one or more title lines before the first line, *i.e.*, before the **SY** line. Such title lines are optional but they can be used to describe the data and the problem. The title can have any text but to avoid conflicts with real command lines it is recommended to begin with an exclamation mark (!).

The first real **PRELIS** syntax line is the **SY** line which is short for **System File**. This tells **LISREL** to read the data file **hsschools.lsf**. This line can also be written as

System File **hsschools.lsf**

The **SD** line is the **Select Delete Cases** command which tells **LISREL** to select the cases with **SCHOOL = 0**, *i.e.*, the Pasteur School sample, and delete the variable **SCHOOL** after selection. The last line is the **Output** command which must always be the last line in a **PRELIS** syntax file. The **RA=pasteur.lsf** on the **OU** line tells **LISREL** to save the selected data in the file **pasteur.lsf**.

To run the file **hsschools1.prl** click the **P (Run PRELIS)** button. This generates an output file **hsschools1.out** with some sections of output described as follows.

Total Sample Size(N) = 156

Univariate Distributions for Ordinal Variables

GENDER	Frequency	Percentage
Boy	74	47.4
Girl	82	52.6

AGEYEAR	Frequency	Percentage
12	41	26.3
13	62	39.7
14	31	19.9
15	17	10.9
16	5	3.2

BIRTHMON	Frequency	Percentage
1	9	5.8
2	17	10.9
3	15	9.6
4	13	8.3
5	16	10.3
6	13	8.3
7	8	5.1
8	15	9.6
9	16	10.3
10	8	5.1
11	11	7.1
12	15	9.6

This reveals that

- There are 156 students in the Pasteur School.
- There are 74 boys (47.4%) and 82 girls (52.6%).
- Most of the students are 13 years old (39.7%) but some are 12 and a few are 16 years old.
- There are 17 students born in February, for example.

The next section of the output gives information about the continuous variables, *i.e.*, the nine test scores VISPERC - SCCAPS:

Univariate Summary Statistics for Continuous Variables

Variable	Mean	St. Dev.	Skewness	Kurtosis	Minimum	Freq.	Maximum	Freq.
VISPERC	29.647	7.110	-0.378	0.736	4.000	1	45.000	2
CUBES	23.936	4.921	0.695	0.255	14.000	1	37.000	3
LOZENGES	19.897	9.311	0.156	-1.074	2.000	1	36.000	5
PARCOMP	8.468	3.457	0.221	-0.052	0.000	1	18.000	1
SENCOMP	15.981	5.244	-0.132	-0.839	4.000	1	27.000	1
WORDMEAN	13.455	6.932	1.009	2.034	1.000	2	43.000	1
ADDITION	101.942	24.958	0.303	-0.395	47.000	1	171.000	1
COUNTDOT	111.263	19.577	0.362	0.091	70.000	1	166.000	1
SCCAPS	195.038	35.708	0.226	0.194	100.000	1	310.000	1

As shown this gives the mean, standard deviation, skewness, kurtosis, minimum, and maximum and the frequency of the minimum and maximum. It is seen that the mean, standard deviation, minimum and maximum varies considerably across the tests. This is a reflection of the fact that the tests have varying number of items.

The skewness and kurtosis are used to evaluate the degree of non-normality in the variables which may be important in the evaluation of the fit of models as explained in later chapters. In the output, there is also a section of tests of skewness of kurtosis which looks like this:

Test of Univariate Normality for Continuous Variables

Variable	Skewness		Kurtosis		Skewness and Kurtosis	
	Z-Score	P-Value	Z-Score	P-Value	Chi-Square	P-Value
VISPERC	-1.934	0.053	1.682	0.093	6.568	0.037
CUBES	3.360	0.001	0.782	0.434	11.900	0.003
LOZENGES	0.818	0.414	-5.966	0.000	36.257	0.000
PARCOMP	1.153	0.249	0.018	0.985	1.329	0.514
SENCOMP	-0.691	0.489	-3.585	0.000	13.332	0.001
WORDMEAN	4.555	0.000	3.207	0.001	31.028	0.000
ADDITION	1.567	0.117	-1.131	0.258	3.736	0.154
COUNTDOT	1.856	0.064	0.398	0.691	3.601	0.165
SCCAPS	1.176	0.240	0.645	0.519	1.798	0.407

The measures of skewness and kurtosis and their tests are explained in the Appendix (see Chapter 12). In practice, one only needs to examine the *P*-values given in this table. *P*-values which are very small, less than 0.05, say, indicate that there is some degree of non-normality. It is seen that CUBES and WORDMEAN are skewed variables, LOZENGES and SENCOMP have too small kurtosis and WORDMEAN has too high kurtosis. Probably WORDMEAN is the most non-normal variable since it is both skewed and kurtotic. In Chapter 7 we will learn about the consequences of non-normality and about ways to deal with this problem.

To obtain the data file for the Grant White School, use the following PRELIS input file **hss-schools2.prl**:

```
!Creating a Data File for Grant White School
SY=hsschools.lsf
SD SCHOOL = 1
OU RA=grantwhite.lsf
```

Syntax for Reading the lsf file

In general, suppose the data file is called **data.lsf**. This name is not case sensitive, so one can use upper case **DATA.LSF** as well.

As we have already seen the PRELIS syntax for reading an **.lsf** file is

```
System File data.lsf
```

This can be shortened to

```
SY data.lsf
```

In SIMPLIS syntax, the command for reading an **.lsf** file is

```
Raw Data from File data.lsf
```

All the words `Raw Data from File` are needed.

In LISREL syntax the command for reading the `.lsf` file is

```
RA=data.lsf
```

More details about PRELIS, SIMPLIS, and LISREL syntax are given later in the book and in documents that come with the LISREL program.

1.5 Estimating Covariance or Correlation Matrices

To estimate a covariance matrix put `MA=CM` on the `OU` line in `hsschools1.prl` or `hsschools2.prl`. This is not useful in this case since there is no meaningful interpretation of the covariances between the categorical variables `GENDER`, `AGEYEAR`, and `BIRTHMON` and the test scores `VISPERC` - `SCCAPS`. However, one can select any subset of variables and estimate the covariance matrix for these variables. For example, to estimate the covariance matrix for the nine test scores for the Grant White School, add the line

```
SE VISPERC-SCCAPS
```

in `hsschools2.prl` and add `MA=CM` on the `OU` line.

The output file gives the covariance matrix in the following form

Covariance Matrix

	VISPERC	CUBES	LOZENGES	PARCOMP	SENCOMP	WORDMEAN
	-----	-----	-----	-----	-----	-----
VISPERC	47.801					
CUBES	10.013	19.758				
LOZENGES	25.798	15.417	69.172			
PARCOMP	7.973	3.421	9.207	11.393		
SENCOMP	9.936	3.296	11.092	11.277	21.616	
WORDMEAN	17.425	6.876	22.954	19.167	25.321	63.163
ADDITION	17.132	7.015	14.763	16.766	28.069	33.768
COUNTDOT	44.651	15.675	41.659	7.357	19.311	20.213
SCCAPS	124.657	40.803	114.763	39.309	61.230	79.993

Covariance Matrix

	ADDITION	COUNTDOT	SCCAPS
	-----	-----	-----
ADDITION	565.593		
COUNTDOT	293.126	440.792	
SCCAPS	368.436	410.823	1371.618

The output file also gives certain characteristics of covariance matrix:

Total Variance = 2610.906 Generalized Variance = 0.106203D+17

Largest Eigenvalue = 1734.725 Smallest Eigenvalue = 3.665

Condition Number = 21.756

In this book a sample covariance matrix will always be denoted **S**. The total variance is the sum of the diagonal elements of **S** and the generalized variance is the determinant of **S** which equals the product of all the eigenvalues of **S**. The largest and smallest eigenvalues of **S** are also given. These quantities are useful in principal components analysis, see Chapter 5. The condition number is the square root of the ratio of the largest and smallest eigenvalue. A large condition number indicates multicollinearity in the data. If the condition number is larger than 30, LISREL gives a warning. This might indicate that one or more variables are linear or nearly linear combinations of other variables. If this is intentional, the warning may be ignored.

The means and standard deviations of the variables are also given in the output file as follows

Means

VISPERC	CUBES	LOZENGES	PARCOMP	SENCOMP	WORDMEAN
-----	-----	-----	-----	-----	-----
29.579	24.800	15.966	9.952	18.848	17.283

Means

ADDITION	COUNTDOT	SCCAPS
-----	-----	-----
90.179	109.766	191.779

Standard Deviations

VISPERC	CUBES	LOZENGES	PARCOMP	SENCOMP	WORDMEAN
-----	-----	-----	-----	-----	-----
6.914	4.445	8.317	3.375	4.649	7.947

Standard Deviations

ADDITION	COUNTDOT	SCCAPS
-----	-----	-----
23.782	20.995	37.035

To estimate the correlation matrix instead of the covariance matrix put **MA=KM** on the **OU** line instead. Since the nine test scores are declared as continuous variables in the **.lsf** file, the correlations will be product-moment correlations, also called Pearson correlations. There are also several other types of correlations that can be estimated, such as polychoric, polyserial, Spearman-rank, and Kendall-tau correlations but these are only relevant when some or all variables are ordinal.

The correlation matrix for the Grant White School, as obtained in the output file is:

Correlation Matrix

	VISPERC	CUBES	LOZENGES	PARCOMP	SENCOMP	WORDMEAN
	-----	-----	-----	-----	-----	-----
VISPERC	1.000					
CUBES	0.326	1.000				
LOZENGES	0.449	0.417	1.000			
PARCOMP	0.342	0.228	0.328	1.000		
SENCOMP	0.309	0.159	0.287	0.719	1.000	
WORDMEAN	0.317	0.195	0.347	0.714	0.685	1.000
ADDITION	0.104	0.066	0.075	0.209	0.254	0.179
COUNTDOT	0.308	0.168	0.239	0.104	0.198	0.121
SCCAPS	0.487	0.248	0.373	0.314	0.356	0.272

Correlation Matrix

	ADDITION	COUNTDOT	SCCAPS
	-----	-----	-----
ADDITION	1.000		
COUNTDOT	0.587	1.000	
SCCAPS	0.418	0.528	1.000

One can also save the covariance or correlation matrix in a file. For example, to estimate the covariance matrix for the nine test score for the Pasteur School, add both `MA=CM` and `cm=pasteur_npv.cm` on the `OU` line in **hsschools1.prl**. The saved covariance matrix **pasteur_npv.cm** is

```
0.50552D+02  0.97127D+01  0.24215D+02  0.29957D+02  0.15355D+02  0.86686D+02
0.10289D+02  0.10689D+01  0.39773D+01  0.11954D+02  0.11400D+02  0.22955D+01
0.21400D+01  0.13041D+02  0.27503D+02  0.21032D+02  0.54681D+01  0.12518D+02
0.15960D+02  0.26318D+02  0.48056D+02  0.64118D+01 -0.18875D+02 -0.45737D+01
0.22330D+02  0.15825D+02  0.35504D+02  0.62288D+03  0.19951D+02  0.31008D+01
0.24414D+02  0.88762D+01  0.12199D+02  0.28976D+02  0.19727D+03  0.38325D+03
0.76336D+02  0.31693D+02  0.96256D+02  0.16866D+02  0.30207D+02  0.47221D+02
0.23936D+03  0.25538D+03  0.12751D+04
```

This gives the elements below and in the diagonal of \mathbf{S} as one long string of numbers, where each number is given in scientific notation to get as much accuracy as possible. Users rarely need to look at these types of files.

The “classical” way of doing structural equation modeling with LISREL was to use a two-step procedure as follows.

Step 1 Use PRELIS to estimate and save the covariance matrix to a file.

Step 2 Read the covariance matrix into LISREL and estimate the model.

Although this two-step procedure can still be used, it is no longer necessary in LISREL 9 since LISREL can read the data file and estimate the model in one step.

All the procedures described in this section can be done with one single PRELIS syntax file using so called stacked input, see file **hsschools3.prl**:

```
!Creating a Data File for Pasteur School
SY=hsschools.lsf
SD SCHOOL = 0
SE VISPERC - SCCAPS
OU MA=CM RA=pasteur_npv.lsf cm=pasteur_npv.cm
!Creating a Data File for Grant White School
SY=hsschools.lsf
SD SCHOOL = 1
SE VISPERC - SCCAPS
OU MA=CM RA=grantwhite_npv.lsf cm=grantwhite_npv.cm
```

This file will split the data into two schools and save the **.lsf** for the nine test scores for each school as well as the covariance matrices for each school.

1.6 Missing Values

Missing values and incomplete data are almost unavoidable in the social, behavioral, medical and most other areas of investigation. In particular, missing values are very common in longitudinal data and repeated measurements data, where individuals are followed and measured repeatedly over several points in time.

One can distinguish between three types of incomplete data:

- Unit nonresponse, for example, a person does not respond at all to an item in a questionnaire.
- Subject attrition, for example, when a person falls out of a sample after some time in a longitudinal follow-up study.
- Item nonresponse, for example, a person respond to some but not all items in a questionnaire.

The literature, *e.g.*, Schafer (1997) distinguishes between three mechanisms of nonresponse.

MCAR Missing completely at random

MAR Missing at random

MNAR Missing not at random

Let z_{ij} be any element on the data matrix. Informally, one can define these concepts as

MCAR $Pr(z_{ij} = \text{missing})$ does not depend on any variable in the data.

MAR $Pr(z_{ij} = \text{missing})$ may depend on other variables in the data but not on z_{ij} . Example: A missing value of a person's income may depend on his/her age and education but not on his/her actual income.

MNAR $Pr(z_{ij} = \text{missing})$ depends on z_{ij} . Example: In a questionnaire people with higher income tend not to report their income.

LISREL has several ways of dealing with missing values:

1. Listwise deletion
2. Pairwise deletion
3. Imputation by matching
4. Multiple imputation
 - EM
 - MCMC
5. Full Information Maximum Likelihood (FIML)

Of these methods the first three are *ad hoc* procedures whereas the last two are based on probability models for missingness. As a consequence, the *ad hoc* methods may lead to biased estimates under MAR and can only be recommended under MCAR.

Listwise deletion means that all cases with missing values are deleted. This leads to a complete data matrix with no missing values which is used to estimate the model. This procedure can lead to a large loss of information in that the resulting sample size is much smaller than the original. Listwise deletion can give biased, inconsistent, and inefficient estimates under MAR. It should only be used under MCAR.

Pairwise deletion means that means and variances are estimated using all available data for each variable and covariances are estimated using all available data for each pair of variables. The means, variances and covariances are then combined to form a mean vector and a covariance matrix which are used to estimate the model. While some efficiency is obtained compared to listwise deletion, it is difficult to specify a sample size N to be used in the estimation of the model, since the variances and covariances are all based on different sample sizes and there is no guarantee that the covariance matrix will be positive definite which is required by the maximum likelihood method. Although pairwise deletion is available in LISREL, it is not recommended. Its best use is for data screening for then it gives the most complete information about the missing values in the data.

Imputation means that real values are substituted for the missing values. Various *ad.hoc.* procedures for imputation have been suggested in the literature. One such is imputation by matching which is available in LISREL. It is based on the idea that individuals who have similar values on a set of matching variables may also be similar on a variable with missing values. This will work well if the matching variables are good predictors of the variable with missing values.

Methods 4 and 5 are both based on the assumption of multivariate normality and missingness under MAR. Method 4 uses multiple imputation methods to generate a complete data matrix. The multiple imputation procedure implemented in LISREL is described in details in Schafer (1997) and uses the EM algorithm and the method of generating random draws from probability distributions via Markov chains (MCMC). Formulas are given in Section 16.1.11. The EM algorithm generates one single complete data matrix whereas the MCMC method generates several complete data matrices and uses the average of these. As a consequence, the MCMC method is more reliable than the EM algorithm. In both cases, the complete data matrix can be used to estimate the mean vector and the covariance matrix of the observed variables which can be used to estimate the model. However, in LISREL 9 it is not necessary to do these steps separately as they are done automatically as will be described in what follows.

Method 5 is based on the following idea. If the variables have a multivariate normal distribution all subsets of the variables also have that distribution. So the likelihood function for the observed values can be evaluated for each observation without using any missing values. Formulas are given in Section 16.1.10.

Method 5 is the recommended method for dealing with the problem of missing data but it can only be used together with a specified model. So we have to postpone the discussion of this method until later chapters.

To illustrate the concepts of missing values and some ways to deal with this problem, we consider another example and data set.

A medical doctor offered all his patients diagnosed with prostate cancer a treatment aimed at reducing the cancer activity in the prostate. The severity of prostate cancer is often assessed by a plasma component known as prostate specific antigen (PSA), an enzyme that is elevated in the presence of prostate cancer. The PSA level was measured regularly every three months. The data contains five repeated measurements of PSA. The age of the patient is also included in the data. Not every patient accepted the offer at the first visit to the doctor and several patients chose to enter the program after the first occasion. Some patients, who accepted the initial offer, are absent at some later occasions for various reasons. Thus, there are missing values in the data.

The data file is **psavar.lsf** where the first 16 rows of data are shown here

	PSA0	PSA3	PSA6	PSA9	PSA12	Age
1	30.400	28.000	26.900	25.200	19.600	69.000
2	27.800	26.700	20.500	18.700	18.800	58.000
3	26.600	21.800	17.800	17.900	14.500	53.000
4	24.800	24.500	20.200	19.800	18.800	61.000
5	33.700	30.300	25.400	27.300	20.100	63.000
6	26.500	24.600	20.900	-9.000	18.900	49.000
7	26.200	24.400	21.800	22.200	18.400	63.000
8	24.800	19.500	18.000	16.100	12.500	49.000
9	28.400	-9.000	22.500	19.400	22.900	63.000
10	26.100	-9.000	23.300	22.000	14.600	56.000
11	28.800	31.300	-9.000	23.100	22.800	68.000
12	29.800	-9.000	25.600	24.500	21.000	67.000
13	22.900	23.900	-9.000	19.400	15.600	47.000
14	30.100	27.700	25.700	20.400	20.800	56.000
15	26.500	-9.000	-9.000	20.000	17.400	57.000

The value -9 is used here to indicate a missing value. This is called a missing value code. One can have one or more missing value code(s). The missing code(s) must be specified in the **.lsf** file. If this is not done the missing value code(s) are treated as real values in the data which is likely to give very wrong results.

To specify the missing value code in the **.lsf** file, proceed as follows. Select **Data** in the **LISREL System Data File Toolbar**, select **Define Variable** and click on **Missing Values**. In the section **Global Missing Value** type -9 in the numerical field:

The screenshot shows a dialog box titled "Missing Values for PSA0 ...". It has two radio buttons: "No missing values" (selected) and "Missing values". Below these are fields for "Low" and "High" values. There is an "Apply to all" checkbox. In the "Global missing value" section, the value "-9" is entered in the numerical field. At the bottom, under "Deletion methods", the "Listwise" radio button is selected, and the "Pairwise" radio button is unselected. "OK" and "Cancel" buttons are at the top right.

Now click **OK** twice and save the **.lsf** file.

In this kind of data it is almost inevitable that there are missing values. For example, a patient may be on vacation or ill or unable to come to the doctor for any reason at some occasion or a patient may die and therefore will not come to the doctor after a certain occasion. For example, a quick perusal of the data shows that

- Patients 9 and 10 are missing at 3 months
- Patient 15 is missing at 3 and 6 months
- Patient 16 is missing at 0, 3, and 12 months

For this data we illustrate listwise deletion, imputation by matching, imputation by EM, and imputation by MCMC and in each case we show how one can get a new **.lsf** file. While these procedures can all be done by using the **Graphical User's Interface**, we will use simple PRELIS syntax files here

Listwise Deletion

To generate an **.lsf** file for the listwise sample, use the following PRELIS syntax file **psavar1.prl**:

```
!Creating psavar_listwise.lsf
SY=psavar.LSF
OU ra=psavar_listwise.lsf
```

This just reads the data and produces the **.lsf** file for the listwise sample. Since listwise deletion is the default in PRELIS when there are missing values in the data, nothing needs to be specified in the PRELIS syntax file.

Some sections of the output file are shown and discussed here

Number of Missing Values per Variable

PSA0	PSA3	PSA6	PSA9	PSA12	Age
-----	-----	-----	-----	-----	-----
17	14	13	12	11	0

This shows that there are most (17) missing values at the first visit to the doctor and fewest (11) missing at 12 months. Age is recorded for all patients.

Distribution of Missing Values

Total Sample Size(N) = 100

Number of Missing Values	0	1	2	3
Number of Cases	46	43	9	2

This shows that there are only 46 patients with complete data on all occasions. So the listwise sample has only 46 cases, i.e., Since the sample size is 100, 54 cases have been deleted. It is also seen that 43 patients have 1 missing value, 9 have 2 missing values, and 2 have 3 missing

values. Thus, no patient have 4 or 5 missing values. The **.lsf file** for the listwise sample is **psavar_listwise.lsf**. This has complete data on 46 patients.

Imputation by Matching

It is not likely that data is MCAR. More likely, the mechanism is MAR since the probability of missingness may depend on Age. So one idea is to use imputation by matching and use Age as a matching variable, see **psavar2.prl**:

```
!Creating psavar_machimputed.lsf
SY=psavar.LSF
IM (PSA0 - PSA12) (Age) XN
OU ra=psavar_machimputed.lsf
```

Here the line

```
IM (PSA0 - PSA12) (Age) XN
```

means impute the missing values on the variables PSA0 - PSA12 by matching on the variable Age. The XN option tells LISREL to list all successful imputations. The general form of IM command is

```
IM Ivarlist Mvarlist VR=n XN XL
```

where **Ivarlist** is a set of variables whose missing values should be imputed and **Mvarlist** is a set of matching variables. **VR**, **XN**, and **XL** are explained below.

The imputation procedure is as follows. Let y_1, y_2, \dots, y_p denote the variables in **Ivarlist** and let x_1, x_2, \dots, x_q denote the variables in **Mvarlist**. To begin, let us assume that there is only a single variable y in **Ivarlist** whose missing values are to be imputed and that y is not included in **Mvarlist**. Let z_1, z_2, \dots, z_q be the standardized x_1, x_2, \dots, x_q , *i.e.*, for each case c

$$z_{cj} = (x_{cj} - \bar{x}_j) / s_j \quad j = 1, 2, \dots, q,$$

where \bar{x}_j and s_j are the estimated mean and standard deviation of x_j . These are estimated from all complete data on x_j .

The imputation procedure is as follows.

1. Find the first case a with a missing value on y and no missing values on x_1, x_2, \dots, x_q . If no such case exists, imputation of y is impossible. Otherwise, proceed to impute the value y_a as follows.
2. Find *all* cases b which have no missing value on y and no missing values on x_1, x_2, \dots, x_q , and which minimizes

$$\sum_{j=1}^q (z_{bj} - z_{aj})^2. \quad (1.1)$$

3. Two cases will occur

- If there is a single case b satisfying 2, y_a is replaced by y_b .

- Otherwise, if there are $n > 1$ matching cases b with the same minimum value of (1.1), denote their y -values by $y_1^{(m)}, y_2^{(m)}, \dots, y_n^{(m)}$. Let

$$\bar{y}_m = (1/n) \sum_{i=1}^n y_i^{(m)}, \quad s_m^2 = [1/(n-1)] \sum_{i=1}^n (y_i^{(m)} - \bar{y}_m)^2,$$

be the mean and variance of the y -values of the matching cases. Then, imputation will be done only if

$$\frac{s_m^2}{s_y^2} < v, \quad (1.2)$$

where s_y^2 is the total variance of y estimated from all complete data on y , and v is the value **VR** specified on the **MI** command. This may be interpreted to mean that the matching cases predict the missing value with a reliability of at least $1 - v$. The default value of **VR** is **VR=.5**, *i.e.*, $v = .5$. Larger values than this are not recommended. Smaller values may be used if one requires high precision in the imputation. For each value imputed, **LISREL** gives the value of the variance ratio and the number of cases on which s_m^2 is based.

If condition (1.2) is satisfied, then y_a is replaced with the mean \bar{y}_m if y is continuous or censored, or with the value on the scale of y closest to \bar{y}_m if y is ordinal. Otherwise, no imputation is done and y_a is left as a missing value.

4. This procedure is repeated for the next case a for which y_a is missing, and so on, until all possible missing values on y have been imputed.

If **Ivarlist** contains several variables, they will be imputed in the order they are listed. This is of no consequence if no variables in **Ivarlist** is included in **Mvarlist**. Ideally, **Ivarlist** contains the variables with missing values and **Mvarlist** contains variables without missing values.

This procedure has the advantage that it does not make any distributional assumption unlike Methods 3 - 5 which depend on the assumption of multivariate normality. Furthermore, it has the advantage that it gives the same results under linear transformation of the matching variables. Thus, if age is a matching variable, age can be in years or months, or represented by the year of birth, and the resulting imputed data will be the same in each case. Another advantage is that the results of the imputation will be the same regardless of the order of cases in the data. A disadvantage is that it does not always impute all the missing values, so one would typically use listwise deletion after imputation.

Imputation of missing values should be done with utmost care and control, since missing values will be replaced by other values that will be treated as real observed values. If possible, use matching variables which are *not* to be used in the **LISREL** modeling. Otherwise, if the matching variables are included in the **LISREL** model, it is likely that the imputation will affect the result of analysis. This should be checked by comparing with the result obtained without imputation.

For each variable to be imputed, the output lists all the cases with missing values. If imputation is successful, it gives the value imputed, the number of matching cases **NM** and the variance ratio **VR**. If the imputation is not successful, it gives the reason for the failure. This can be that no matching case was found or that the variance ratio was too large. The **XN** option on the **IM** command will make **LISREL** list only successful imputations, and the **XL** option makes **LISREL** skip the entire listing of cases.

The output always gives the number of missing values per variable, both before and after imputation. Here we show some sections of the output file **psavar2.out**:

Number of Missing Values per Variable

PSA0	PSA3	PSA6	PSA9	PSA12	Age
-----	-----	-----	-----	-----	-----
17	14	13	12	11	0

This is the number of missing values per variable before imputation. After this comes the listing of the successful imputations:

Imputations for PSA0

Case	16 imputed	with value	33.200 (Variance Ratio = 0.000), NM=	1
Case	33 imputed	with value	29.800 (Variance Ratio = 0.000), NM=	1
Case	67 imputed	with value	38 (Variance Ratio = 0.016), NM=	2
Case	68 imputed	with value	33.400 (Variance Ratio = 0.365), NM=	4
Case	73 imputed	with value	26.800 (Variance Ratio = 0.000), NM=	1
Case	83 imputed	with value	29.150 (Variance Ratio = 0.327), NM=	4
Case	99 imputed	with value	38 (Variance Ratio = 0.008), NM=	3

Imputations for PSA3

Case	12 imputed	with value	31.500 (Variance Ratio = 0.000), NM=	1
Case	15 imputed	with value	37.350 (Variance Ratio = 0.212), NM=	2
Case	16 imputed	with value	31.200 (Variance Ratio = 0.000), NM=	1
Case	32 imputed	with value	35.900 (Variance Ratio = 0.000), NM=	1
Case	34 imputed	with value	22.800 (Variance Ratio = 0.000), NM=	3
Case	99 imputed	with value	35.867 (Variance Ratio = 0.055), NM=	3

Imputations for PSA6

Case	11 imputed	with value	35.900 (Variance Ratio = 0.000), NM=	1
Case	15 imputed	with value	34.600 (Variance Ratio = 0.000), NM=	1
Case	20 imputed	with value	27.600 (Variance Ratio = 0.000), NM=	1
Case	32 imputed	with value	33.900 (Variance Ratio = 0.000), NM=	1
Case	44 imputed	with value	22.167 (Variance Ratio = 0.085), NM=	3
Case	68 imputed	with value	30.800 (Variance Ratio = 0.261), NM=	4
Case	74 imputed	with value	32.133 (Variance Ratio = 0.185), NM=	3
Case	89 imputed	with value	34.600 (Variance Ratio = 0.000), NM=	2
Case	95 imputed	with value	29 (Variance Ratio = 0.000), NM=	1

Imputations for PSA9

Case	6 imputed	with value	17.500 (Variance Ratio = 0.096), NM=	2
Case	22 imputed	with value	28.400 (Variance Ratio = 0.000), NM=	1
Case	28 imputed	with value	19.833 (Variance Ratio = 0.250), NM=	3
Case	30 imputed	with value	33.575 (Variance Ratio = 0.006), NM=	4
Case	60 imputed	with value	15.400 (Variance Ratio = 0.000), NM=	1
Case	70 imputed	with value	29.400 (Variance Ratio = 0.000), NM=	1
Case	100 imputed	with value	25.100 (Variance Ratio = 0.000), NM=	1

Imputations for PSA12

Case	16	imputed	with value	24.750 (Variance Ratio = 0.026), NM=	4
Case	19	imputed	with value	24.325 (Variance Ratio = 0.420), NM=	4
Case	23	imputed	with value	24.750 (Variance Ratio = 0.000), NM=	1
Case	35	imputed	with value	26.400 (Variance Ratio = 0.086), NM=	3
Case	51	imputed	with value	24.750 (Variance Ratio = 0.027), NM=	4
Case	69	imputed	with value	9.600 (Variance Ratio = 0.000), NM=	1
Case	84	imputed	with value	24.750 (Variance Ratio = 0.000), NM=	1

After these imputations the number of missing variables per variable is

Number of Missing Values per Variable After Imputation

PSA0	PSA3	PSA6	PSA9	PSA12	Age
-----	-----	-----	-----	-----	-----
10	8	4	5	4	0

It is seen that the number of missing values has been considerably reduced for each variable.

The distribution of missing values after imputation is now

Distribution of Missing Values

Total Sample Size(N) = 100

Number of Missing Values	0	1	2	3
Number of Cases	76	18	5	1

It is seen that after the imputation there are 76 cases without missing values. So the listwise sample size after imputation will be 76. More details about the patterns of missing values are obtained in the missing data map:

Missing Data Map

Frequency	PerCent	Pattern
76	76.0	0 0 0 0 0 0
5	5.0	1 0 0 0 0 0
6	6.0	0 1 0 0 0 0
1	1.0	1 1 0 0 0 0
2	2.0	0 0 1 0 0 0
1	1.0	1 0 1 0 0 0
2	2.0	0 0 0 1 0 0
1	1.0	1 0 0 1 0 0
1	1.0	1 1 0 1 0 0
1	1.0	0 0 1 1 0 0
3	3.0	0 0 0 0 1 0
1	1.0	1 0 0 0 1 0

The missing data map gives all possible patterns of missing data and their sample frequencies in absolute and percentage form. Each column under **Pattern** corresponds to a variable. A 0 means a complete data and a 1 means a missing data.

In this example, although we know from the distribution of missing values that 18 cases have only 1 missing value, we don't know on which variable they are missing. But the missing data map tells us that 5 are missing on PSA0, 6 are missing on PSA3, 2 are missing on PSA6, 2 are missing on PSA9, and 3 are missing on PSA12, thus $5 + 6 + 2 + 2 + 3 = 18$. The last column is the variable Age which has no missing values.

Multiple Imputation by EM

Another way to impute missing values is multiple imputation using the EM algorithm. This will impute all missing values and create a complete data matrix. To do this kind of imputation, use the PRELIS syntax file **psavar3.prl**:

```
!Creating psavar_eminputed.lsf
SY=psavar.LSF
EM CC = 0.00001 IT = 25 TC = 0
OU ra=psavar_eminputed.lsf
```

The third line is the command for doing multiple imputation by EM. Here CC is a convergence criterion with default value 0.00001, IT is the maximum number allowed (default = 200), and TC is a constant which specifies how cases with missing values on *all* variables should be treated. TC = 0 means that their values will be replaced by means, TC = 1 means that these cases will be deleted, and TC = 2 means that their values will be left as missing values. Although each of the parameters CC, IT, and TC have default values, they must nevertheless be specified, *i.e.*, an empty EM line will not work.

The resulting imputed data file is **psavar__eminputed.lsf** which has complete data on all 100 cases.

To perform multiple imputation with the MCMC method instead, just replace the line, see file **psavar4.prl**

```
EM CC = 0.00001 IT = 25 TC = 0
```

with

```
MC CC = 0.00001 IT = 25 TC = 0
```

1.7 Data Management

Often the initial data is not such that it can be analyzed directly. One can add or delete cases in the **.lsf** file directly and one can also delete variables. This section describes how one can add new variables which are functions of other variables.

We illustrate this with some data that will be analyzed in Chapter 3.

In 1951, all British doctors were sent a brief questionnaire about whether they smoked tobacco. Ten years later the number of deaths from coronary heart disease for smokers and non-smokers was

recorded. The original data come from a study by Sir Richard Doll. A table of the number of deaths and the number of person-years of observation in different age groups was published by Breslow and Day (1987, p. 112). A similar table was given by Dobson and Barnett (2008, p. 168). This is given here in Table 1.1.

Table 1.1: Deaths from Coronary Heart Disease among British Doctors. Originally published in Dobson and Barnett (2008, Table 9.1). Published with kind permission of © Taylor and Francis Group, LLC 2008. All Rights Reserved

Age Group	Smokers		Non-smokers	
	Deaths	Person-years	Deaths	Person-years
35–44	32	52407	2	18790
45–54	104	43248	12	10673
55–64	206	28612	28	5710
65–74	186	12663	28	2585
75–84	102	5317	31	1462

For the analysis with LISREL in Section 3.3.1 in Chapter 3, we need some additional variables. Begin by typing a text file like this, **corheart1.dat**, say

```
SMOKE AGECAT DEATHS PYEARS
1      1      32    52407
1      2     104    43248
1      3     206    28612
1      4     186    12633
1      5     102     5317
0      1       2    18790
0      2      12    10673
0      3      28     5710
0      4      28     2585
0      5      31     1462
```

where **SMOKE** is a dummy variable equal to 1 for smokers and equal to 0 for non-smokers, **AGECAT** is coded 1, 2, 3, 4, 5 and treated as a continuous variable, **DEATHS** is the number of deaths, and **PYEARS** is the number of person-years. All numbers are taken from Table 1.1.

For the analysis in this example we need to do a number of simple calculations with the data in this **dat** file. There are at least three different ways to do these calculations:

1. Since this is a very small data set, one can do these calculations by using a hand calculator or by using an external spread sheet, such as Microsoft Excel, and then import the results to a LISREL system file **.lsf** as described in Section 1.1.
2. One can use PRELIS syntax.
3. One can use LISREL's **Compute Dialog Box** in the graphical users interface, see the file **Graphical Users Interface.pdf** that comes with the program.

We illustrate each of these alternatives in turn.

1: Calculation by Hand

The file **corheart1.dat** is appended by four new variables and called **corheart2.dat**:

SMOKE	AGECAT	DEATHS	PYEARS	DTHRATE	AGESQ	SMKAGE	LNPYEARS
1	1	32	52407	0.0006116	1	1	10.866796
1	2	104	43248	0.0024047	4	2	10.674706
1	3	206	28612	0.0071998	9	3	10.261581
1	4	186	12633	0.0147233	16	4	9.444068
1	5	102	5317	0.0191837	25	5	8.578665
0	1	2	18790	0.0001064	1	0	9.841080
0	2	12	10673	0.0011243	4	0	9.275473
0	3	28	5710	0.0049037	9	0	8.649974
0	4	28	2585	0.0108317	16	0	7.857481
0	5	31	1462	0.0212038	25	0	7.287560

where **DTHRATE** is the death rate equal to the number of deaths divided by the number of person-years, **AGESQ** is **AGECAT** squared, **SMKAGE** is the product of **SMOKE** and **AGECAT**, and **LNPYEARS** is the natural logarithm of **PYEARS**.

To import **corheart2.dat** to a LISREL system file select **File** \Rightarrow **Import Data**, enter 8 as the **Number of Variables**, tick the box **Variable names at top of file**, and click **OK**. This gives a LISREL system file **corheart2.lsf** which looks like this (to see six decimals, select **Edit** \Rightarrow **Format**, enter 12 as Width and 6 as Number of Decimals):

	SMOKE	AGECAT	DEATHS	PYEARS	DTHRATE	AGESQ	SMKAGE	LNPYEARS
1	1.000000	1.000000	32.000000	52407.000000	0.000612	1.000000	1.000000	10.866796
2	1.000000	2.000000	104.000000	43248.000000	0.002405	4.000000	2.000000	10.674706
3	1.000000	3.000000	206.000000	28612.000000	0.007200	9.000000	3.000000	10.261581
4	1.000000	4.000000	186.000000	12633.000000	0.014723	16.000000	4.000000	9.444068
5	1.000000	5.000000	102.000000	5317.000000	0.019184	25.000000	5.000000	8.578665
6	0.000000	1.000000	2.000000	18790.000000	0.000106	1.000000	0.000000	9.841080
7	0.000000	2.000000	12.000000	10673.000000	0.001124	4.000000	0.000000	9.275473
8	0.000000	3.000000	28.000000	5710.000000	0.004904	9.000000	0.000000	8.649974
9	0.000000	4.000000	28.000000	2585.000000	0.010832	16.000000	0.000000	7.857481
10	0.000000	5.000000	31.000000	1462.000000	0.021204	25.000000	0.000000	7.287560

2: Using PRELIS Syntax

If the data is large, it is better and easier to do these calculations using PRELIS syntax instead. The following PRELIS syntax file, see file **corheart2.prl**, will produce a file **corheart2.lsf** which is identically the same as the one generated previously

```
!Creating the .lsf file by PRELIS syntax
da ni=4
ra=corheart1.dat lf
new DTHRATE=DEATHS*PYEARS**-1
new AGESQ=AGECAT**2
```

```
new SMKAGE=SMOKE*AGECAT
new LNPYEARS=PYEARS
log LNPYEARS
co all
ou ra=corheart2.lsf
```

This is a slightly different PRELIS syntax file, compared with those previously used. This is because it reads a text file **.dat** instead of a **.lsf** file. So we must tell LISREL first how many variables there are. This is done by the line

```
da ni=4
```

The line

```
ra=corheart1.dat lf
```

tells LISREL to read the data file **corheart1.dat**. Note that this is an **ra** line and not an **sy** line. The keyword **lf** tells LISREL that the names of the of the variables are in the first line of the **.dat** file. **new** is the command for computing a new variable. So

```
new DTHRATE=DEATHS*PYEARS**-1
new AGESQ=AGECAT**2
new SMKAGE=SMOKE*AGECAT
```

will compute the new variables **DTHRATE**, **AGESQ**, and **SMKAGE**. The logarithm transformation needs a special explanation. First we make a copy of **PYEARS** and call it **LNPYEARS**. Then we compute the log of **LNPYEARS**.

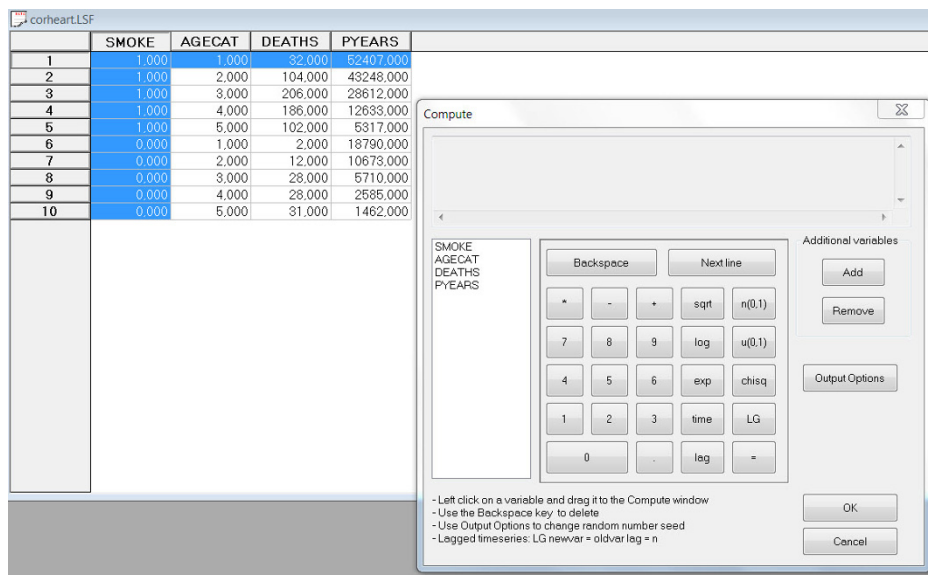
3: Using the Compute Dialog Box

No matter how large the data is one can use LISREL's **Compute Dialog Box** to do the calculations.

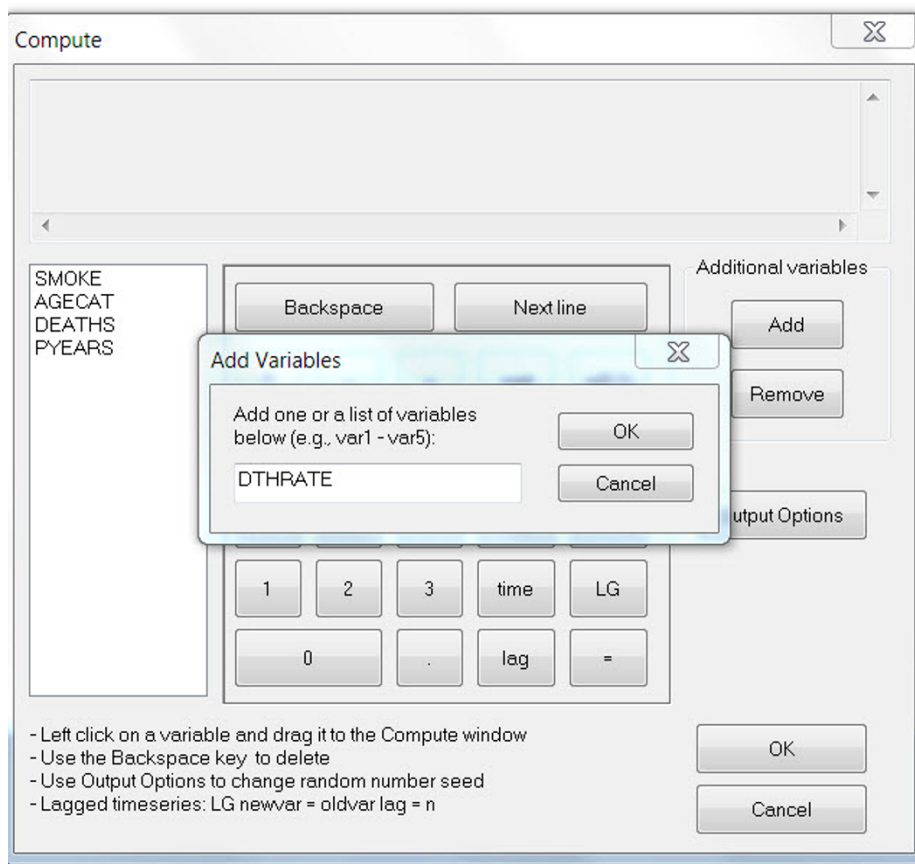
First import **corheart1.dat** to obtain the following **.lsf** file **corheart.lsf**:

	SMOKE	AGECAT	DEATHS	PYEARS
1	1.000	1.000	32.000	52407.000
2	1.000	2.000	104.000	43248.000
3	1.000	3.000	206.000	28612.000
4	1.000	4.000	186.000	12633.000
5	1.000	5.000	102.000	5317.000
6	0.000	1.000	2.000	18790.000
7	0.000	2.000	12.000	10673.000
8	0.000	3.000	28.000	5710.000
9	0.000	4.000	28.000	2585.000
10	0.000	5.000	31.000	1462.000

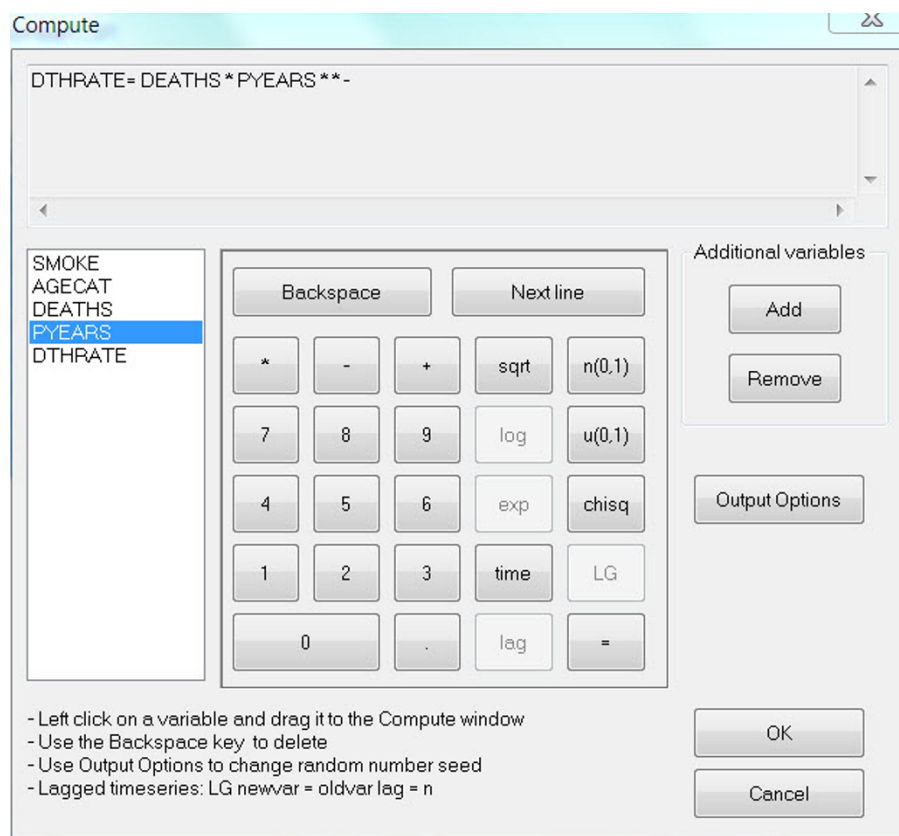
Select **Compute** in the **Transformation** menu:



Click on **Add** and type DTHRATE:



Click **OK**. This will add the variable **DTHRATE** in the list of variables on the left side of the compute screen. To compute the values of this variable drag the variable **DTHRATE** to the compute screen and use the symbols in the calculator as shown (the death rate is defined the number of deaths divided by the number of person-years, but since no division sign is available in the calculator, we multiply by the inverse of **PYEARS** instead as indicated by **PYEARS**-1**):

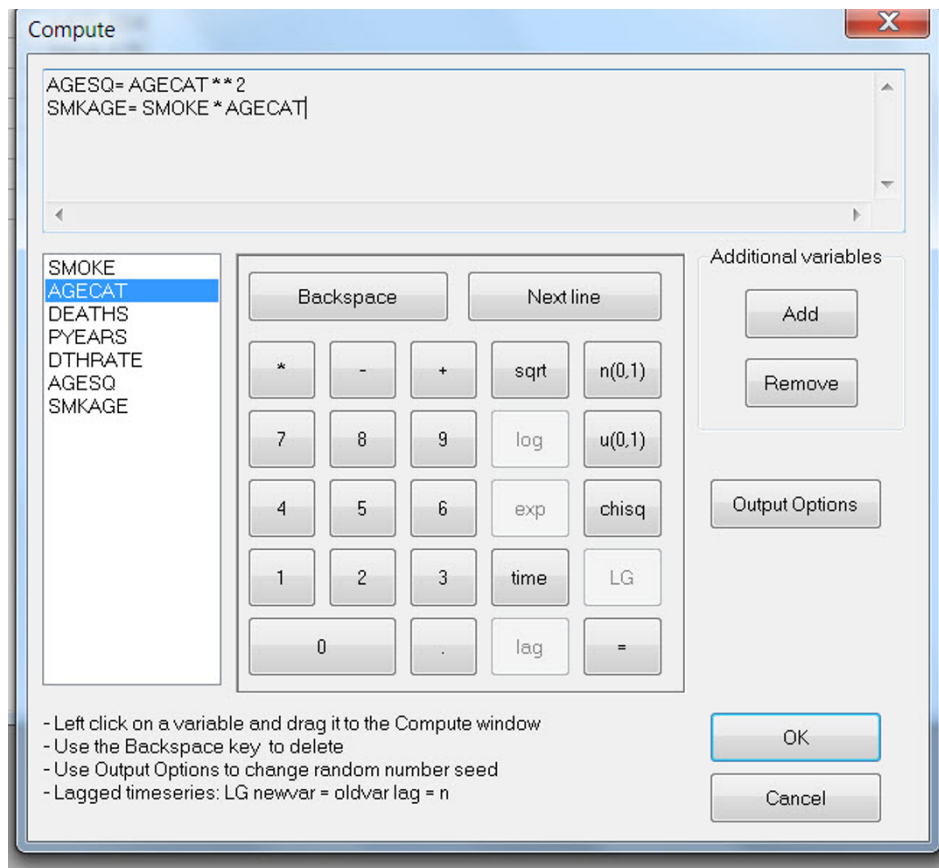


Then click **OK** and the **.lsf** file now looks like

	SMOKE	AGECAT	DEATHS	PYEARS	DTHRATE
1	1.000000	1.000000	32.000000	52407.000000	0.000611
2	1.000000	2.000000	104.000000	43248.000000	0.002405
3	1.000000	3.000000	206.000000	28612.000000	0.007200
4	1.000000	4.000000	186.000000	12633.000000	0.014723
5	1.000000	5.000000	102.000000	5317.000000	0.019184
6	0.000000	1.000000	2.000000	18790.000000	0.000106
7	0.000000	2.000000	12.000000	10673.000000	0.001124
8	0.000000	3.000000	28.000000	5710.000000	0.004904
9	0.000000	4.000000	28.000000	2585.000000	0.010832
	0.000000	5.000000	31.000000	1462.000000	0.021204

Note that the death rate increases with age for both smokers and non-smokers and the death rate is higher for smokers than non-smokers for all age groups except the age category 5 (the 75-84 years group), where the reverse is true.

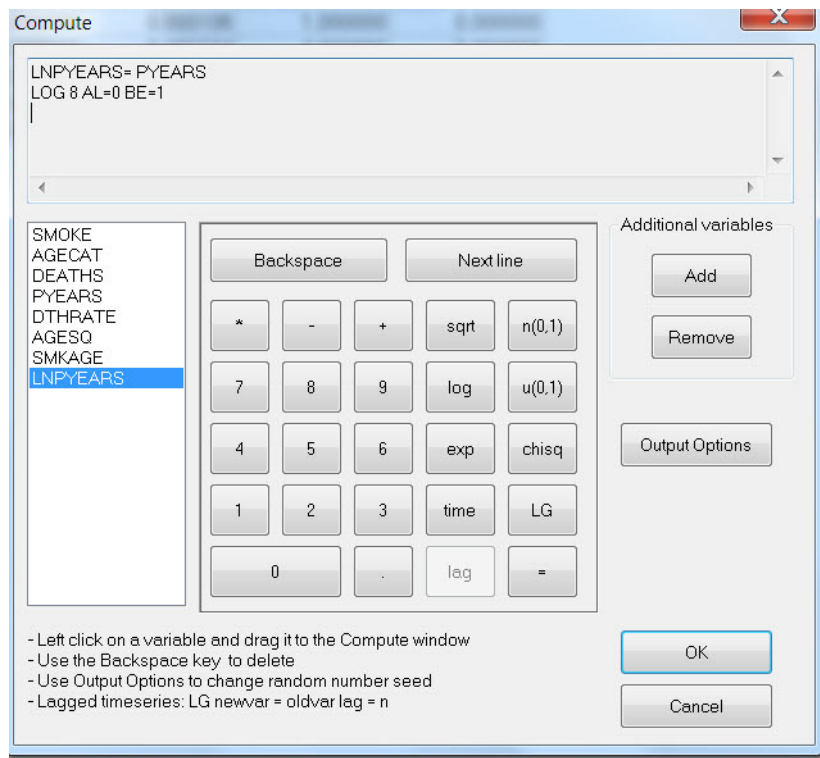
In the GLM model we therefore include two more variables, **AGESQ** as the square of **AGECAT** and **SMKAGE** as the product of **SMOKE** and **AGECAT**. To compute the values of these variables, we use the compute screen again:



After clicking **OK** in the compute screen, the new **.lsf** file looks like this:

	SMOKE	AGECAT	DEATHS	PYEARS	DTHRATE	AGESQ	SMKAGE
1	1.000000	1.000000	32.000000	52407.000000	0.000611	1.000000	1.000000
2	1.000000	2.000000	104.000000	43248.000000	0.002405	4.000000	2.000000
3	1.000000	3.000000	206.000000	28612.000000	0.007200	9.000000	3.000000
4	1.000000	4.000000	186.000000	12633.000000	0.014723	16.000000	4.000000
5	1.000000	5.000000	102.000000	5317.000000	0.019184	25.000000	5.000000
6	0.000000	1.000000	2.000000	18790.000000	0.000106	1.000000	0.000000
7	0.000000	2.000000	12.000000	10673.000000	0.001124	4.000000	0.000000
8	0.000000	3.000000	28.000000	5710.000000	0.004904	9.000000	0.000000
9	0.000000	4.000000	28.000000	2585.000000	0.010832	16.000000	0.000000
10	0.000000	5.000000	31.000000	1462.000000	0.021204	25.000000	0.000000

To perform the estimation of the GLM model, we must also compute the **offset** variable $\ln(n_i)$, i.e., the logarithm of **PYEARS**. To do this is somewhat counterintuitive. To keep the original variable **PYEARS**, click **Add** in the compute screen and type **LNPYEARS**, generate a line **LNPYEARS=PYEARS**, select the variable **LNPYEARS** in the list of variables on the left and click on **LOG**. The compute screen now looks like this:



After clicking **OK** in the compute screen, the new **.lsf** file looks like this:

	SMOKE	AGECAT	DEATHS	PYEARS	DTHRATE	AGESQ	SMKAGE	LNPYEARS
1	1.000000	1.000000	32.000000	52407.000000	0.000611	1.000000	1.000000	10.866796
2	1.000000	2.000000	104.000000	43248.000000	0.002405	4.000000	2.000000	10.674706
3	1.000000	3.000000	206.000000	28612.000000	0.007200	9.000000	3.000000	10.261581
4	1.000000	4.000000	186.000000	12633.000000	0.014723	16.000000	4.000000	9.444068
5	1.000000	5.000000	102.000000	5317.000000	0.019184	25.000000	5.000000	8.578665
6	0.000000	1.000000	2.000000	18790.000000	0.000106	1.000000	0.000000	9.841080
7	0.000000	2.000000	12.000000	10673.000000	0.001124	4.000000	0.000000	9.275473
8	0.000000	3.000000	28.000000	5710.000000	0.004904	9.000000	0.000000	8.649974
9	0.000000	4.000000	28.000000	2585.000000	0.010832	16.000000	0.000000	7.857481
10	0.000000	5.000000	31.000000	1462.000000	0.021204	25.000000	0.000000	7.287560

Multivariate Analysis with LISREL

Jöreskog, K.G.; Olsson, U.H.; Wallentin, F.Y.

2016, XV, 557 p. 155 illus., 89 illus. in color., Hardcover

ISBN: 978-3-319-33152-2