TestGraf
A Program for the Graphical Analysis of Multiple Choice Test and Questionnaire Data

J. O. Ramsay
McGill University

January 1, 1995

Acknowledgement: The development of TestGraf was supported by grant AP320 from the Natural Sciences and Engineering Research Council of Canada. The author would also like to express his appreciation to members of his Department for making available their data, to Raymond Baillargeon for his valuable editorial assistance, and to the users of TestGraf who have provided invaluable feedback.

The author can be contacted at
Department of Psychology
McGill University
1205 Dr. Penfield Ave.
Montreal, Quebec
Canada H3A 1B1
telephone (514) 398-6123
fax (514) 398-4896
electronic mail ramsay@psych.mcgill.ca

Contents

List of Figures

1 The Objectives of TESTGRAF

TestGraf is designed to aid the development, evaluation, and use of multiple choice examinations, psychological scales, questionnaires, and other types of data.

TestGraf was designed for use without any formal statistical knowledge. The essential aspects of each display were designed to be self-explanatory, although more statistically sophisticated users will also find information that they may find helpful. Most of the output from TestGraf is in graphical form, and the program was designed to be used interactively. Since the analyses that TestGraf performs take very little time on even personal computers of modest power, there will be only a limited need to produce hard or permanent copy of the displays that TestGraf produces, but nevertheless TestGraf does have hard copy capability.

The minimal data appropriate for a TestGraf analysis are characterized by:

- a set of examinees, respondents, cases, or other types of choice generators,
- a set of choice situations, such as items or questions on examinations or questionnaires, and
- a set of two or more options available for each choice situation.

A question or choice situation must be one of two types:

Multiple Choice Exam Item: Each question presents a list of possible answers, and the student is required to choose one. One of the answers is designated correct, and all other are considered by the examiner to be incorrect. Aside from correct versus incorrect, there is no particular ordering characterizing the answers.

Scale Item: Each possible answer associated with this type of question has a numerical weight that is assigned to it by the person constructing the item. This means that the answers are ordered in terms of the ordering of the weights attached to them. In this sense, multiple choice exam items are scale items where one option receives a weight of one and the others a weight of zero.

A set of questions can be mixtures of these two types of items. For example, an examination may contain both multiple choice items and open-ended short-answer questions. The latter have alternatives determined by the possible scores that the grader gives to the responses. Or, alternatively, a small subset of the multiple choice items, called a *testlet*, may be scored by counting the number of correct answers, and this count can be assigned to these items as a block, thereby treating them collectively as a single scale item.

TestGraf was conceived for use with tests or questionnaires where the responses of any individual are primarily determined by the amount or level of some single ability, characteristic, or trait. Or, alternatively, where the user of the program is interested in the extent to which a single proficiency or trait determines performance.

In the case of examinations, this manual will tend to refer to this presumed single dimension as *proficiency*, since the term ability has connotations that are often not entirely appropriate to exam performance. For psychological scales, containing mostly or entirely scale-type items, this dimension will be called the *trait* presumed to determine option choices.

TestGraf makes use of modern statistical methods to produce accurate estimates of examinee or respondent characteristics. For example, for examination data, TestGraf enables better estimates of examinee proficiency or ability by making use of the information provided by which wrong options were chosen for incorrectly answered items. Such estimates will tend to be much precise than the conventional estimates based only on number correct, and especially for examinees of low to medium proficiency. These more efficient estimates, which are still expressed in familiar terms as numbers correct, can be used to either replace or modify the classical number correct scores reported for examinees.

TestGraf also displays graphically what range of proficiency or trait values are reasonably consistent with the set of choices made by an individual. This means that TestGraf conveys the relative precision or level of confidence attached to the best estimate, so that one can assess how much information is provided by a respondent's data about her/his position on the proficiency or trait continuum in question.

Instructors or questionnaire developers will find TestGraf helpful for diagnosing problems with items, and for deciding whether to rewrite items in order to clear up ambiguous wording or to offer wrong options that are more plausible.

Although by default TestGraf is used to study the internal structure of a test or scale, TestGraf can also be used to study how individual items relate to scores on some entirely separate set of scores of measures on the examinees or respondents. For example, TestGraf might be used by an instructor to see how well test items relate to the final grade of examinees, which might be a composite of other tests as well as this one.

Finally, TestGraf can be used to graphically display the differences among two or more groups of examinees or respondents in terms of how they respond to items. Used in this way, TestGraf can, for example show whether there are systematic ways in which females and males respond to a particular question, or whether different ethnic or language groups respond to questions in different ways.

The next section discusses the installation of TestGraf. If you prefer to first see what TestGraf does, it might be preferable to skip this section and turn to Section 3.

2 Installation of TestGraf

In the following notes on installation, and throughout the documentation, commands which are to be typed into the personal computer are indicated in the following typewriter font:

`This is an example of typewriter font.`

2.1 System Requirements

This version of TestGraf is designed to run on microcomputer systems supporting the MS/DOS operating system. The following are minimum requirements:

- MS/DOS version 2 or subsequent versions, or systems such as OS/2 or Microsoft Windows supporting MS/DOS applications.
- 512K of RAM
- a hard disk with at least two megabytes of free space.
- while not required, a math coprocessor is strongly suggested
- one of the following graphics adaptors: CGA, EGA, VGA, or Hercules compatible graphics.

2.2 Installation

The distribution diskette contains executable versions for five programs:

1. TestGraf: This is the principal program for displaying the results.
2. TestSave: Processes a raw data file to set up the various files needed by TestGraf.
3. TestLasr: The special graphics commands produced by TestGraf are used to generate a file of PostScript commands for printing graphical displays using a PostScript laser printer.

4. TestComp: Results from two or more TestGraf analyses on subgroups of examinees or respondents are graphically compared to show any differences in the way the subgroups respond to items.

5. TestAnal: This is a version of TestGraf which only computes option characteristic curves, and does not display any graphs. This version is useful for analyzing very large files where more memory is required than TestGraf leaves available.

The diskette also contains a number of files of data in raw form, and files generated by processing the data with the setup program TestSave. Also, TestGraf requires a number of additional files to execute, and these are also on the diskette.

The first step should be to make a back-up copy of the diskette. Once the diskette is backed up, you should create a directory on your hard disk using a command such as

```
mkdir testgraf.dir
```

The next step is to copy the contents of the distribution diskette into the directory. For example, if the diskette is in drive `a:`, and you have created the directory `c:testgraf.dir` with the above command, this can be achieved by the command

```
xcopy a: c:testgraf.dir
```

Note that TestGraf requires a number of additional files besides `testgraf.exe` to run properly, so be sure to copy all the files on the diskette to the hard disk.

TestGraf uses a number of files, and you should ensure that `files=20` appears in the `config.sys` file.

2.3 Memory Limitations

TestGraf requests the memory it needs at the time it executes, and it is therefore difficult to specify exactly when a set of data will be too large for the program to analyze on a typical small computer. The largest set of data analyzed to date involved 18,500 examinees and 75 items, each with 6 options, so that most sets of test data will not cause any memory allocation problems. If the program does fail due to a lack of memory, one of the following messages may appear:

```
run-time error F6700: ALLOCATE
- heap space limit exceeded

Cannot execute TESTGRAF.EXE

Insufficient RAM memory available.
```

If this occurs, eliminate any memory resident or "terminate and stay resident" (TSR) programs on the system and attempt the analysis again. See Section 5 for further details.

2.4 Source Code

This distribution of TestGraf does not contain any source code, which is in C and FORTRAN. If the source code is required, it can be obtained on request.

2.5 Costs and Restrictions

All components of TestGraf and the other programs are provided either free of copyright restrictions or under license from the holder of the copyright. The latter case is noted where appropriate in the manual. However, it is unethical to distribute any part of the programs commercially or for profit.

A reimbursement of $35.00 US is requested of anyone receiving the distribution diskette and the manual from McGill University to cover the costs of reproduction and mailing. A cheque or money order should be made payable to McGill University.

2.6 Changes in This Version

The previous version of TestGraf and this manual displayed the date September 1, 1993; the current version is dated January 1, 1995.

Files produced by previous versions of the program are *not* compatible with this version. Any analyses of data processed by previous versions should begin with the setup program TestSave rather that with TestGraf.

The changes in this version relative to the previous version are:

1. TestGraf can now analyze data containing a mixture of multiple choice and scale items. The previous version required that all items be of one type or the other.

2. The `.itm` file describing each item now is much more readable, and contains more information about each item. This file is produced by the setup program TestSave but, like previous versions, can also be modified by any editor prior to a TestGraf analysis.

3. Each option for any question can be given a label.

4. Two new plots are provided by TestGraf: the estimated distribution of total test or scale scores, and a reliability function plot.

5. New plots are also provided by TestComp to aid comparisons among groups, and this program is extended in other ways.

6. The setup program TestSave has been extended to allow selection of a subgroup. This is useful in setting up the comparison of subgroups using TestComp.

7. The two plots used in the previous versions to display expected values and standard deviations of total score have been combined into a single plot.

8. The expected percentage correct score used to display multiple choice examination results has now been replaced by expected number correct.

9. Previous versions defaulted to 32 values of the display variable, and required that the number of values be a power of 2. This restriction was due to the use of the Fast Fourier Transform in previous versions. The FFT is no longer used, and any number up to 101 is now permitted. The default is 51.

10. The smoothing process used to compute option characteristic curves now uses only one phase rather than two, and will be seen to be substantially faster.

11. The separate program TestPwpl used in previous versions to produce versions of the TestGraf plots in an intermediate plotting language has been eliminated. This can now optionally be done by TestGraf itself.

12. A shortened version of TestGraf called TestAnal is included that only computes option characteristic curve values. This is useful for the analysis of files too large to be analysed by TestGraf.

TestGraf and its associated programs are always in a state of development and flux. They evolve in part as a consequence of comments and other feedback from users. The author would be very grateful to receive these at any time, and can be contacted by electronic mail at the address ramsay@psych.mcgill.ca, or by fax at (514) 398-4896.

3 Two Sample Analyses

3.1 The Introductory Psychology Data

In order to get a quick idea of what TestGraf is all about, you may want to try the following analysis. The data being analyzed in this tutorial come from a multiple choice examination given to 379 students in an introductory course in psychology in the Christmas exam period of 1989 at McGill. The test itself consisted of 100 multiple choice items, each having four response options.

This sample analysis presumes that the files needed by TestGraf have already been generated by running program TestSave on the raw data. Forget about this for now; the distribution diskette already contains these files; TestSave will be described in Section 5.

Here we go! Follow these steps, typing each command as indicated followed by the `Enter` or `Return` key. In the following descriptions of each step, the actual commands are in `typewriter` font.

First go to directory `testgraf.dir` or whatever directory you used to contain the files on the distribution diskette. For example, if you created `testgraf.dir` from the root directory, and are in that directory, the command `cd testgraf.dir` will put you into that directory.

Once in the appropriate directory, enter the command

```
tg psych101
```

You will now see the following screen display:

```
Introductory Psychology Exam           |First  line of title
                                       |Second line of title

379                 Number of examinees or respondents
100                 Number of items
51                  Number of plotting points for displays
2                   Number of principal components displayed
Expected Score      Display variable option (N, E, F, or O)
Number Correct      Type of scoring (MC items only)
0.33                Smoothing parameter value
                    Compute curve values
                    Input proficiency values
                    Only output .pwo plotting file
                    Control plotting limits for abscissa
Continue with TESTGRAF analysis
Quit
```

This is a menu that displays title lines, parameter values, and other information used by TestGraf. You could at this point change these values (except for the number of examinees and number of items), but there is no need to do so at this point. Simply press the `Enter` key to indicate that you do not wish to change anything, and are ready to proceed with the analysis.

After some status messages from TestGraf you will now see this menu displayed on the screen:

```
1 Probability of choice curves for each item
2 Expected total correct -- proficiency curve
3 Distribution of expected total scores
4 Information function for test
5 Relative credibility curves for each examinee
6 Factors of item curves
7 Maximum likelihood proficiency estimates
8 Tetrahedron display for each item
9 Quit
```

This lists the analysis options in TestGraf. Any option can be selected by:

- using the up and down arrow keys to move the highlighting to the desired option,
- using a mouse to move the cursor to the desired option and clicking,
- or typing the number or letter opposite the desired option followed by `Enter`

Once the desired option is highlighted, use the `Enter` key to activate the option. Double-clicking the mouse on the option has the same effect.

Now please press the `Enter` key to get the first display, the option characteristic curves.

You are now asked to indicate a number designating which item you wish to view. The following small menu is displayed in the upper right corner of the screen:

```
Next
Last
Again
Enter
Displ
PostS
Quit
```

Which item is displayed is determined by highlighting an option in this menu using one of the three techniques described above, followed by typing `Enter`.

- The `Next` option in the small window selects the next item following the current item displayed. If no current item is displayed, this selects item number 1, and the same is true if the last item is currently displayed.
- `Last` selects the previous item, or the last item if the current item is number 1.
- `Again` shows over again the item previously displayed.
- `Enter` opens up a small window. You then type into this window the number of the item that you wish to display.

- `Displ` opens up another small window, which indicates the type of variable that will be used for the horizontal variable or abscissa in the display.

- `Quit` terminates this sequence of displays and returns you to the main menu listing the TestGraf analysis options.

Let's begin with item number 4. Move the highlight bar to `Enter`, press the `Enter` key, type `4` into the window, and then type `Enter` again.

Question number 4 in this examination is, as are all other questions in this exam, of multiple choice type. It is

> *If the correlation coefficient between variable A and variable B is $r = -1$, one possible conclusion is that:*
>
> 1. *variable A and variable B are not related*
> 2. *A has a strong negative effect on B*
> 3. *B has a strong negative effect on A*
> 4. *as A increases B decreases*

You should now see the graph displayed in Figure 1.

Here are some comments on the graph:

- The four curves show the relation between the probability that an examinee will choose an option for item 4 and his proficiency measured in number correct terms.

- The abscissa or horizontal variable measures proficiency in familiar terms as the number of items that an examinee of that proficiency level would, on average, answer correctly.

- The solid curve with the cross-hatching shows the probability of choosing the correct option (in this case option number 4) as a function of proficiency. This is called the *option characteristic curve* for this option. Note that this probability is around 0.4 for examinees with proficiency scores of less than 50, and rises to over 0.9 for examinees with proficiency scores exceeding 80. This is an item of medium difficulty.

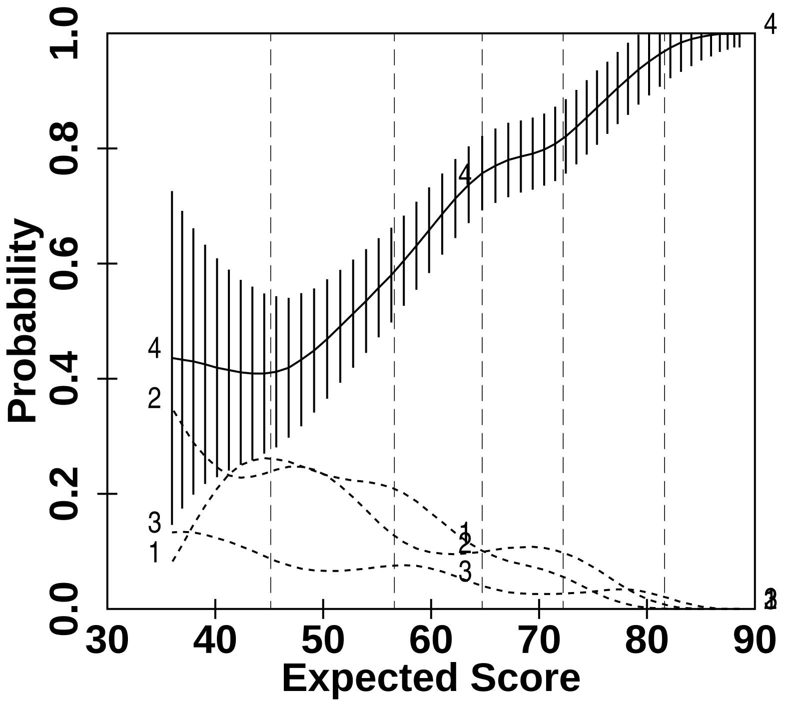

Figure 1: Option characteristic curves for item 4 for the Introductory Psychology Exam. Probability of choosing an option is plotted against proficiency. Vertical bars indicate 95% confidence limits for values of correct option curve. Vertical dashed lines indicate percentage points or quantiles for the distribution of number correct.

- The dashed curves show the probability-proficiency relation for each of the three incorrect options. We see that the very weakest examinees seem to prefer option 2 to the others, and that examinees at all proficiency levels are least likely to choose option 3.

- The cross-hatching indicates 95% confidence limits on the position of the correct answer option characteristic curve. We see that the position of the curve is not well defined for the lowest proficiency examinees, but that it is fairly well defined by the data for proficiency scores above 55 or so.

- The vertical dashed lines indicate the percentages of students whose actual numbers correct fell below various values. For example, we see that 50% of the students had scores of 65 or less. Also, 50% of the students had scores that fell between 57 and 73. Only 5% of the students had scores below 42, and this lack of data in this region explains why the option characteristic curves are not well defined for low-proficiency examinees.

- The right part of the display (not shown in Figure 1) contains the traditional measure of performance, the biserial correlation. This is the correlation between the 0-1 binary variable indicating whether or not the item was passed, and the total score on the text.

Now that we have seen the results for item 4, we can have a look at the results for item 5, which is as follows:

Mendel's laws state that:

1. *a gene can have more than one allele, and a recessive gene determines the trait in homozygous states*
2. *one member of each gene pair is produced by each parent, and different genes undergo segregation independently of one another*
3. *one member of each gene pair is produced by each pair and not all genes undergo segregation independently of each other*
4. *different genotypes may result in the same phenotype and a dominant gene determines the train in heterozygous states*

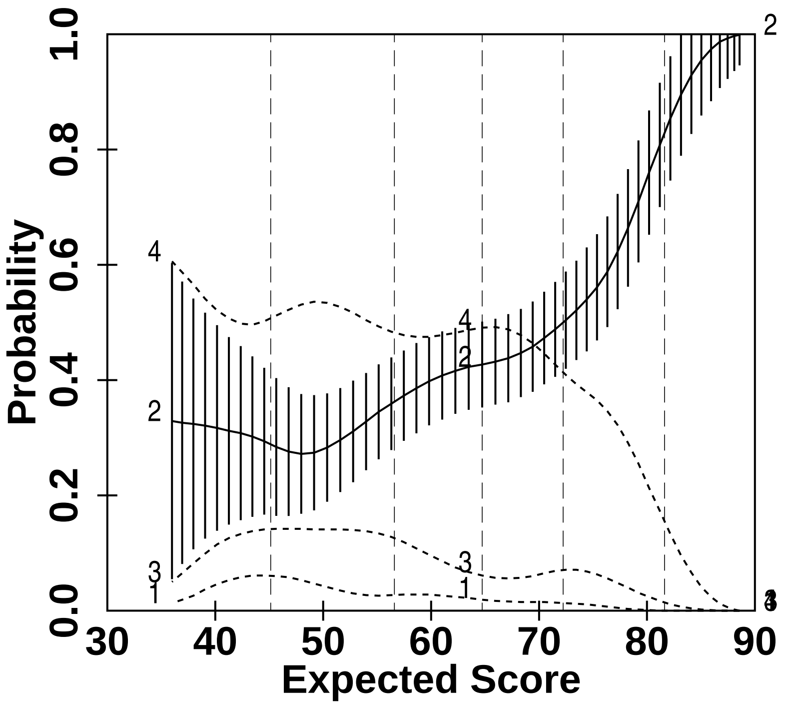

Figure 2: Option characteristic curves for item 5 for the Introductory Psychology Exam.

Let us advance to item 5 by typing `Enter` twice. The first keystroke opens up the small menu, and since `Next` is highlighted by default, the second keystroke moves us to the next item.

You should now see the display in Figure 2. It shows the same type of curves as you saw for item 4, but now for item 5. This is a more difficult item, however, since only examinees in the top 25% of the proficiency range have a high probability of getting the item right. We see that this is due to the attractiveness of incorrect option 4, which only the top students are able to recognize as wrong. Few students are interested in options 1 and 3.

There are 100 items, and you may want to view a few more by using the options in the small menu. Items 7, 80, 96, and 99 have severe problems, for example. Item 29 is probably the best item in the exam. When you are satisfied and wish to move on to other types of displays, open up the small menu if you have not already done so by typing `Enter`, highlight or select `Quit`, and type `Enter` again.

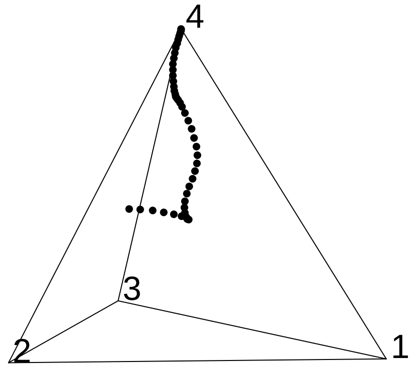

Figure 3: The probability tetrahedron for item 4 for the Introductory Psychology Exam.

This ends the series of item plots and returns you to the main menu listing TestGraf analysis options.

TestGraf complements the static item display that we have just viewed by a dynamic display that permits one to view interesting relations among the option choices at a glance. A dynamic display is one in which motion is used to allow the viewer to inspect three-dimensional relationships.

To see a dynamic display for item 4, highlight the main menu option called `Tetrahedron display for each item`. A simple way to highlight this menu option is to type `t`, but you can also use a mouse or the arrow keys. Once highlighted, type `Enter` to activate the display.

Once activated, you will again see the small menu in the upper right corner controlling which item will be displayed. If you highlight the `Enter` line, press the `Enter` key, type `4`, and then `Enter` again, the display for item 4 will appear.

You will see something similar to what is shown in Figure 3. Your first task is to make it move. Do this by pressing and holding down the space bar. The display shows a three-dimensional structure with a curve inside it.

This geometrical figure is a *regular tetrahedron.* It is the three-dimensional counterpart of an equilateral triangle; it has four instead of three vertices, and each of its four faces is an equilateral triangle.

Each vertex is labeled by an option number. The correct option (number 4 for item 4) appears at the top vertex.

Why a tetrahedron? The answer is that this display shows choices among four options. Thus there are four probabilities of choice to view simultaneously. But these probabilities are not free to vary arbitrarily; they must sum to one. The set of four-dimensional points which are nonnegative and sum to one define a regular tetrahedron, just as the set of three-dimensional points summing to one define an equilateral triangle, and the set of two-dimensional points of this nature define a straight line segment.

Even though this tetrahedron lives in 4-d space, it actually only takes up three dimensions, and we can therefore look at it inside 3-d space. Of course the computer screen is really only 2-d, but dynamic display of the projection of this figure on the screen allows our eyes and brains to reconstruct the 3-d relationships characterizing the figure.

Inside the tetrahedron is a track or curve constructed from a number of points. Each point corresponds to a level of proficiency, as displayed in the item plots that we just viewed. The lowest level of proficiency is identified by a small square, and the highest level of proficiency in this particular display is virtually identical with the top correct option vertex.

The closeness or proximity of a particular point to a given vertex indicates the probability that the corresponding option will be chosen by examinees or respondents at that proficiency level. Thus, the fact that the highest proficiency point is at the top vertex indicates simply that the most proficient students are choosing the right option with a probability of virtually one, and, consequently, the wrong options with almost zero probability.

It follows that if the curve is in generally close to a face, as is the case here, then the probability is near zero that the option identifying the vertex opposite the face will be chosen. Thus, hardly anyone is going for option 3 in this item. If a portion of the curve is near an edge, the implication is that examinees at these proficiency levels are confining their choices primarily to the two options at the ends of the edge, and avoiding the remaining two options.

The question mark in the center of the figure provides a point of reference. If a point is near this location, examinees at this level are choosing among all four options with nearly equal probability.

Additional interesting features of this dynamic display can be seen by looking at the 3-d graph for item 5. To get this, press the `Enter` key to end the item 4 display, and then `Enter` again to activate the next display.

If a portion of the curve is near an edge, the implication is that examinees at these proficiency levels are confining their choices primarily to the two options at the ends of the edge, and avoiding the remaining two options. We see this to be the case for almost the entire curve, indicating that all students are in effect choosing between options 2 (correct) and 4 (incorrect). This item could perhaps be improved by replacing options 1 and 3 with options that attract at least some students.

The spacing between points conveys more information. We see that for all except the highest proficiency levels the points are very closely packed, and near the center of the 2-4 edge. But beyond a fairly high proficiency level the spacing suddenly becomes large, and the curve moves rapidly in the direction of the correct answer. This implies that this is a very difficult item, with only the best students showing a strong tendency to choose the right answer. Of course, this is also obvious in Figure 2.

You might want to have a look at some more dynamic displays. Items 7, 95, and 96 are especially interesting. You might want to look at each of these before or after viewing the corresponding option characteristic curve plots produced by TestGraf analysis option 1.

We shall now have a look at some examinee displays. When you see the main option menu, type `5` or otherwise highlight the option `Relative credibility curves for each examinee`. Typing `Enter` will then select the examinee display sequence.

You will again see the small menu in the upper right corner that was used to control the item plots. Using this menu will now activate plots for examinees. We can look at the results for examinee 2 by highlighting the `Enter` option, typing `2`, and then `Enter`.

The display for examinee 2 is shown in Figure 4.

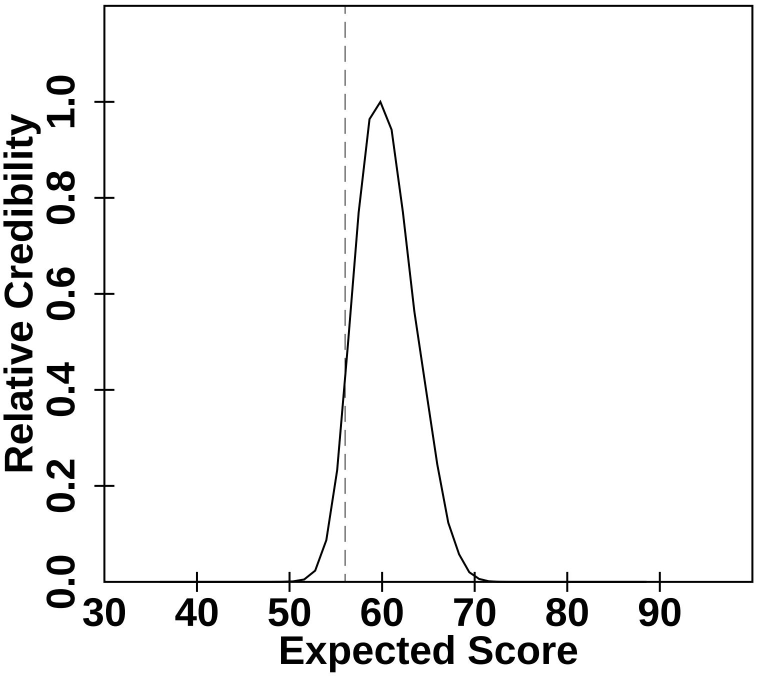

Figure 4: The relative credibility curve for examinee 2 for the Introductory Psychology Exam. The solid curve indicates the relative likelihood or credibility of possible proficiency values given this examinees choice of answers. The vertical dashed line indicates the number of correct answers for this examinee.

- The solid curve shows the relative likelihood or probability of this examinee's true proficiency level being at various values. For convenience of display, the curve has been made to have a maximum of 1.0, and is called the *relative credibility curve* for this examinee. It can be seen that, on the basis of the examinees option choices, wrong as well as right, it is very unlikely that his true proficiency is outside of the range 50 to 70. We can also note that the most likely value, where the curve reaches 1.0, is about 62%. This is called the *maximum likelihood estimate* of proficiency.

- the vertical dashed line running from 0.0 to 1.0 indicates the examinee's actual observed number of correct items, and this is also indicated as being 0.56 on the right portion of the display (not shown in Figure 4. Note, however, that the maximum likelihood estimate also takes account of whether the wrong answer options chosen are typical of more proficient examinees or not. In the case of this examinee, his wrong option choices suggested that his true proficiency is about 6 higher than his observed number correct.

Visit the other displays by selecting other options on the main options menu if you wish. We shall defer describing the other displays until the next section, however. When you are through, select option `Quit` in the main menu to terminate TestGraf.

3.2 Beck Depression Inventory Data

This second tutorial example aims to show how TestGraf can be used with a psychological scale to explore the performance of individual items.

The Beck Depression Inventory (Beck, Rush, Shaw & Emery, 1979) is a widely used set of 21 items developed to assess how depressed individuals are. The original scale was intended to be used with patients undergoing or considering treatment for depression, and hence likely to exhibit fairly significant symptoms of chronic depression. Since its inception, however, this scale has also been widely used by researchers in psychology to study a range of problems, and often with groups of subjects who are not clinically depressed.

In this example, the respondents are 242 McGill University students. The data were collected as part of a number of studies conducted by Prof. David

Zuroff. It is more than reasonable to wonder whether the scale consistently measures anything like depression for this population, who as a whole can be expected to be rather undepressed.

Associated with each option for all 21 items is a numerical weight, which in this case is one of the integers from 0 to 3. The order of the weights corresponds to the level of depression that is presumed to be appropriate to that option.

3.2.1 Setting Up the Data with TestSave

We shall run through this analysis from the very beginning, where we set up the data. The raw data are set up in a file named `bdi.dat`. The first 5 lines of data are as follows:

```
21  4
  0000       333333333333333333333
  1100       000000000011011010010
  1101       010000110000000000100
  1102       100100110111111111021
```

Obviously the first line of the data contains the number of items. It also indicates the number of characters of label used to identify each respondent.

The second line contains an answer key. If this were a multiple choice exam, the characters in this line would indicate the correct options. However, the key line also plays the role of telling TestSave how the data for each respondent is organized; it is a prototype for subsequent lines of actual data. The location of each response to be used is indicated by positioning one of the possible response characters in the appropriate position. In the case of scales, this character need only be one of the characters actually used to indicate a response; in this case we have chosen arbitrarily the response `3`, indicating the greatest level of depression, on this key line, but we could also have used the characters 0, 1, or 2.

The key line may also contain characters in positions containing identifying labels for respondents. The first four `0`'s play this role.

Once the data are set up in this way, we run the setup program TestSave. This program processes the raw data and sets up some files that TestGraf requires. This setup phase only needs to be done once, while we may want to run TestGraf many times, so it makes sense to relegate the initial processing of the raw data to a separate program.

TestSave is run by entering the following command

```
ts bdi
```

TestSave infers from the raw data various useful pieces of information. Depending on how the data are arranged, it may not get everything right, so TestSave asks you to confirm or change each of its proposals. It displays the first six lines of the data so that you will be able to remember how the data are arranged in the file.

Each time TestSave asks a question, a box opens up at the bottom of the screen. If the response is simply "yes" or "no", the box is at the lower right corner. One of the options is highlighted, and if this is the appropriate response, it is selected by simply pressing the `Enter` key. Otherwise, the other option is chosen by moving the highlight bar to the other bar using the mouse or the up and down arrow keys, followed by the `Enter` key. If the required response is longer, the box will occupy more columns as appropriate, and the response is entered into the box, followed by pressing the `Enter` key.

The questions asked by TestSave are:

1. Are one or more items to be used to select subgroups? The answer is "no" in this case, and pressing the `Enter` key moves to the next question.

2. The number of lines of data per respondent. It guesses 1, and asks you to either confirm that this number is correct, or to enter the appropriate values. TestSave has got it right, so you respond by pressing the `Enter` key to select the highlighted "yes" response.

3. The number of respondents. It calculates 242, which is correct, and you respond again by pressing `Enter`.

4. TestSave then tries to figure out the format or arrangement of the responses on each line. Once it has worked this out, it displays the first six lines of data, and then shows you the FORTRAN style format specification that it proposes to use. It gets it right for these data, so you respond with `Enter`.

5. TestSave then asks if the data are from a multiple choice exam. The answer in this case is "no", so you must now move the highlighting down to this response before pressing the `Enter` key. Or you can double-click on this bar with the mouse.

6. TestSave then proceeds to set up the required files that TestGraf will use. This takes only a few seconds.

7. TestSave has only one more question to ask. Most data contain some indication of when a respondent did not respond to a question. Non-response can always be treated as an explicit response, but by default TestSave and TestGraf assume that a blank character corresponds to an omitted response. This is correct for these data, so the TestSave analysis is completed by pressing once again the `Enter` key.

TestSave outputs information about each item in a file with the extension `.itm`. You might want to use an editor to examine the contents of the file `bdi.itm` produced by this analysis. You will see, for example, the frequencies with which each option is chosen.

3.2.2 The TestGraf Analysis

We are now ready to invoke TestGraf to look at the items.

We set TestGraf to work by typing

```
tg bdi
```

You might want to try your hand at changing the TestGraf information displayed in the initial menu by changing the two title lines. Move the highlighting bar to the first line and press `Enter`. You will now see a blank highlighted line displayed at the bottom of the display. If you type something like "Beck Depression Inventory Data", this appears in the blank field. Pressing `Enter` places this information in the first title line. The second title line can similarly be changed to something like "Nonclinical University Students".

Press `Enter` to begin the TestGraf analysis. Since the data are now being analyzed for the first time, you will see TestGraf report on its progress as it sets up results for each item. When it is ready, the menu listing the set of analysis options pops up.

We shall have a look at one of the best items in the scale (for this population, at least), number 11. It is as follows:

0 I am no more irritated by things than I ever am.

1 I am slightly more irritated now than usual.

2 I am quite annoyed or irritated a good deal of the time.

3 I feel irritated all of the time now.

We will use the item plotting option number 1 to have a look at item 11. Press `Enter` since this option is automatically highlighted.

You now observe the small control panel in the upper right corner of the screen. We want to move directly to item 11, so move the highlighting down to `Enter` and press the key `Enter`. At once a small window appears, and just type `11` followed by the `Enter` key.

What is now displayed is what is shown in Figure 5 below. The display consists of two plots. The first is similar to that shown in Figure 1 for the Psychology test. We see how the probabilities of choosing the four options are estimated to vary with score on the entire scale. As one would hope, only those with the smallest scale scores are choosing option 0; that is, a low depression scale score is associated with choosing the option claimed to go with the least level of depression. As the total depression score increases, respondents are estimated to be more likely to choose the next level option with a score of 1. Then the option with a score of 2 begins to take over, although it doesn't reach its peak. Few respondents ever choose the option scored as 3. This is not surprising, since the vertical dashed lines indicate that only 25% of the respondents achieve scores of 9 or above, out of a maximum score of 63. In fact, clinically depressed patients would usually have scores far higher than even those in this group scoring in the top 5%. University students are generally a pretty happy lot! It is very much to be expected, then, that even on a good item, few of these respondents will choose option indicating the most depression.

The right plot in Figure 5 shows how the average score on this item varies with the score on the entire test. Again, one is pleased to see that the average item score climbs consistently as the total test score increases. Of course, it does not approach the maximum value of 3, but this is only because none of the total scale scores come anywhere near their largest values, either. The

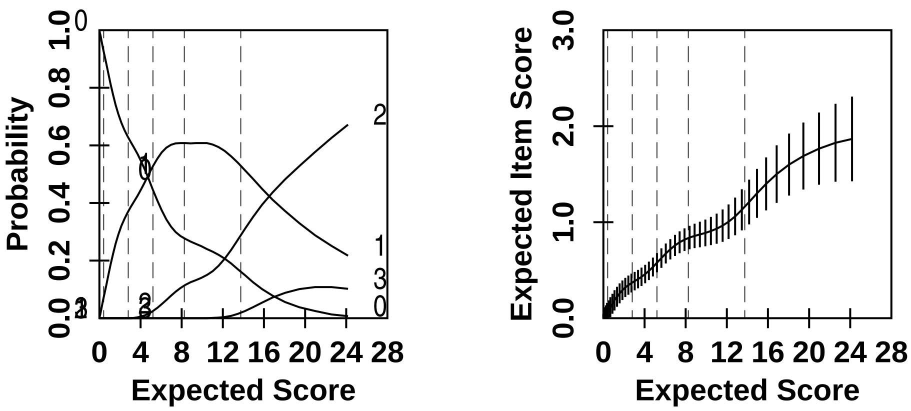

Figure 5: Item 11 from the Beck Depression Inventory Data. The left graph contains the four option characteristic curves. The right graph shows the expected item score as a function of the expected score on whole scale. Vertical bars on this curve indicate 95% confidence limits for the location of the curve. Vertical dashed lines in both figures show the quantiles of the distribution of observed scale scores.

data just do not offer any information about what would happen with total scale scores of, say, 30 or more.

The cross-hatching or bars on the expected item score curve in the second plot indicates estimated 95% confidence regions for the true curve. Note that the curve is well defined below a total score of about 10, put poorly defined for higher values. This is due to the fact that only about 20% of the respondents get scores in this higher range, and only about a dozen get scores above 15, so that there is very little data available for estimating the curve for higher scale scores.

Not all the items in this scale perform as well for this group of respondents. Look at item 18 on the screen (it is not shown here) by

1. pressing the `Enter` key to terminate the display,
2. moving the highlight in the control panel to `Enter`,
3. type `18` in the window, and
4. hit the `Enter` key to bring up the display.

Here we see a somewhat more "depressing" item. Virtually all respondents choose either options 0 or 1, and aside from those with the very lowest scale scores, there is little variation in the probabilities of choice. Consequently, we see in the right plot that the average score on the item does not vary substantially over the entire range of scale scores. This item is contributing little to the assessment of whatever the scale relates to for these subjects.

We return to item 11 (after perhaps some explorations of other items), but this time we use the dynamic 3-d display for another view of what is happening. Pressing the `Enter` key while the above plot is displayed returns us to the control panel, and we then move the highlight down to the `Quit` line and again press `Enter`. This returns us to the main menu, and we type `8` to bring up the dynamic display option. Using the same process as before, bring up the display for item 11 by using the `Enter` option.

One sees that the curve or track begins at the option labeled `0`. That is, those with the lowest scale scores choose this option with virtual certainty. The track moves in the direction of the `1` vertex as the scale score increases, and then changes direction for the higher observed scale scores to more toward vertex `2`. If the item were to continue its fine performance, but with respondents with larger scale depression scores than we have actually observed, we would expect the curve to change direction again, and finally terminate at or near the `3` vertex. In short, an excellent graded response scale item like this should do a "tour of the vertices". To reiterate, that the tour is not completed here is no fault of the item or of the scale; we just do not have enough truly unhappy respondents.

However, the Beck Depression Inventory does show excellent scale properties when data from a more clinical population are analyzed. This illustrates the point that the quality of the scale is partly a question of the scale being used on a population with an appropriate range of trait values.

In Section 8.7 other aspects of the scale are described.

4 A General Description of TestGraf

In this section we offer a general overview of TestGraf followed by a more detailed description of each of the displays that it produces.

The fundamental objective of TestGraf is to display the relationship between the probability that examinees or respondents choose various options for each item and the proficiency or trait level of the examinee making the choice. Thus, TestGraf estimates and works with the probability of choosing an option as a function of proficiency or trait level. This function or curve is called the *option characteristic curve*.

The displays in Figures 1 and 2 are the core of the program. Alternatively, the dynamic display shown in Figure 3 is a technique for presenting the joint behavior of all the option characteristic curves for a particular item.

Once these option characteristic curves are estimated, various other useful quantities can also be estimated based on these curves. These additional results include the examinee relative credibility curves, an example of which is in Figure 4, as well as the test information function, test characteristic function, standard error of total number correct score, and a principal components analysis of correct option item characteristic curves. For each examinee or respondent there is the possibility of estimating the proficiency or latent trait score by maximum likelihood, thereby producing a more efficient or accurate estimate that the traditional scoring methods, and also of displaying the likelihood function itself.

4.1 Notation

To aid further discussion, we introduce a little notation:

$i = 1, \ldots, n$: is the index for items, of which there are n.

$m = 1, \ldots, M_i$: the index for the options within item i, of which there are M_i.

$a = 1, \ldots, N$: the index for examinees or respondents, of which there are N.

θ: indicates a proficiency or trait level. This is an unobserved or *latent* variable.

$P_{im}(\theta)$: the fundamental notion of TestGraf. It is the function relating the probability of choosing option m for item i to proficiency level θ. One might think of $P_{im}(\theta)$ as the proportion of examinees with proficiencies at or near θ who choose this option for this item.

$\eta(\theta)$: a function of the unobserved variable θ is the test or scale score that examinees with proficiency or trait level θ will, on the average, have. It can be thought of as the average actual or observed score associated with examinees at trait level θ. It is presumed to be an increasing or monotonic function, meaning that it preserves the rank ordering of any collection of θ values. The main use of the values η in TestGraf is to define the horizontal or abscissa or "X" values used in the graphs produced by TestGraf.

$q = 1, \ldots, Q$: the index for values θ_q of the proficiency or $\eta(\theta_q)$ that will be displayed and used by TestGraf. There are Q of these, and by default TestGraf uses $Q = 51$.

x_a: a score or quantity that either may be input into TestGraf for examinee a, or computed from the data. For examinations, x_a is by default the number of items correctly answered, and in this case TestGraf computes these numbers internally. For scale or mixed types of exams or questionnaires, it is by default the total of the scale scores for each item. But x_a can also be any otherwise calculated score or measure for examinee a, and in this case is input to TestGraf for its use, and consequently overrides these default values.

w_{im}: a numerical weight to be attached to option m for item i for a scale type item.

y_{ima}: an *indicator variable* which takes on the values:

- 1 if examinee a chose option m for item i
- 0 if examinee a did not choose option m for item i

4.2 How TestGraf Works

The program estimates the option characteristic curves $P_{im}(\theta)$ by going through the following steps:

Step 1: A value or score x_a is associated with each examinee by one of the following methods:

1. computing the number of correct answers for each examinee for multiple choice items,
2. computing what is called the *formula score* for multiple choice items, which is the sum over items of the value 1 if the correct answer is chosen, or the value $-1/(M_i - 1)$ if the wrong answer is chosen. This is often called the "right minus a fraction of wrong" score, and is used mainly by testing agencies to correct for guessing.
3. computing the scale score for scales or mixed item types, which is the sum over items of the numerical weights associated with the options chosen,
4. reading in values from a file. These values can arise from any source, including
   - scores on a completely different test or scale,
   - scores arising from some specialized scoring procedure, or
   - proficiency or trait estimates produced by a previous TestGraf analysis of the same data.

Step 2: The examinees or respondents are sorted on the basis of the values or scores x_a, with ranks within tied values assigned randomly,

Step 3: The a^{th} examinee by order of size of x_a is assigned the a^{th} quantile of the standard normal distribution, z_a. This is the value such that the area under the standard normal density function to the left of this value is equal to $a/(N+1)$.

Step 4: For the m^{th} option for the i^{th} item the indicator values y_{ima} are computed. Here examinee index a refers to the a^{th} examinee by order of size of z_a rather than by the original order.

Step 5: The relationship $P_{im}(\theta)$ is estimated by *smoothing* the relationship between the 0-1 indicator variable values y_{ima} and the standard normal quantiles z_a. Smoothing is in effect a type of local averaging, in which for any proficiency or trait level θ the probability of choice $P_{im}(\theta)$ at that level is a weighted average of the values of y_{ima} for examinees or respondents with proficiency or trait levels close to θ.

The smoothing technique used is a kernel smoothing operation that uses what is called a "Gaussian kernel" with a smoothing parameter that is given a default value by TestGraf and that the user can modify. Further details of this smoothing process are deferred to Section 6.

At this point it is worth addressing a potential criticism of the way TestGraf works. One might be skeptical of the default choice made by TestGraf of ranking examinees or respondents according to test or scale score, since one of the central goals of a program like TestGraf is to replace these inefficient and even biassed indicators of latent trait value by something more statistically powerful and accurate.

Also, one might well wonder whether there is any point to a TestGraf analysis if the basis of the analysis is just test scores: "Why not just use the test scores themselves, as we have always done?" These are fair questions and deserve a careful answer. First, recall that it is not the score values themselves that are used, but only their ranks; if one were to input score ranks rather than scores, TestGraf would produce exactly the same results. Moreover, these ranks are then replaced by what amounts to a smooth and only moderate nonlinear transformation of rank.

Now, insofar as the ranks are based on inefficient indicators of proficiency such as number of correct items, they are themselves subject to considerable error variability. Here, however, is where the next phase, the smoothing step 5, plays a critical role. It turns out that even a substantial amount of error in the ranks or the quantile replacements has only a very small impact on the estimated curve values. This can be demonstrated both by mathematical analysis and by the repeated analysis of simulated data, where one knows the values of the curves being estimated.

Finally, TestGraf permits the iterative refinement of the values used in Step 1. TestGraf can compute more accurate maximum likelihood estimates of trait values, and these can be cycled back into the program as a undoubtably more efficient basis for ranking examinees. Even here, however, experience with many types of data has shown that only very small changes in estimated curve values result, and then only for extreme values of the trait. Moreover, these changes become rapidly negligible if the cycles or iterative refinements are continued, so two or three cycles is the maximum that will almost always be worthwhile. But for the vast majority of applications, no iterative refinement is really necessary, and the default choice of test or scale score for ranking examinees works fine.

4.3 Organization of a TestGraf Analysis

TestGraf cannot operate on a raw data file. First the setup program TestSave must be run with the raw data file as input. TestSave operates on this raw data file to produce a number of files that TestGraf needs. TestSave also calculates from the data information such as the number of items, the number of examinees or respondents, the format for the responses for any particular examinee, an appropriate value for the parameter controlling the smoothing process, and finally sets some options for TestGraf at some proposed values. The next section goes into greater detail on how the raw data file is to be prepared for analysis by TestSave. This setup phase is separated from the actual TestGraf analysis because one will usually want to run many TestGraf analyses on the same set of data, but this initial setup needs only be carried out once.

Once TestSave has been run and the needed files successfully set up, TestGraf can then be run. On the first analysis, it calculates the curve values $P_{im}(\theta_q)$ and stores these values in a file with a special name. To save time in future runs, these values can be input from this file rather than recalculated. Of course, if something critical is changed, such as the smoothing parameter or the number of curve evaluation points Q, then the curve values must be recalculated.

It is also possible to get a permanent or hard copy of the displays that TestGraf produces on a laser printer capable of processing the PostScript language. This permits a hard copy of the item displays.

In order to produce a file of PostScript commands, TestGraf first produces a file of higher level commands in an intermediate and easily interpretable graphics language. An auxiliary program, TestLasr, is then run to convert these intermediate commands into PostScript. The final step is to direct the file with PostScript commands to the laser printer.

There are two methods for producing the file containing the high level graphics commands to be processed by TestLasr.

1. The logical menu item labelled

   `Only output .pwo plotting file`

   in the opening menu when TestGraf is invoked can be selected. This puts TestGraf in noninteractive mode, and it runs through the displays generating the plotting commands rather than displaying results on the screen. These plotting commands are put in file `testgraf.pwo`.

2. The menu box displayed prior to each display in a TestGraf analysis has an option labeled `PostS`. Selecting this option results in TestGraf generating the graphics commands for this and subsequent displays automatically. This option is turned off the next time the `PostS` is selected. In other words, the `PostS` display option toggles graphic command output on and off.

Using either method, the result is a file with the fixed name `testgraf.pwo`. TestLasr converts the contents of this file to a file with the fixed name `testgraf.ps`. It is this latter file that is directed to a laser printer capable of handling the PostScript language.

It can sometimes be possible, using certain types of computers and printers, to obtain a hard copy of any display by using a screen capture process directed to a dot matrix or laser printer. One common way to do this is to set up the printer to emulate an IBM Proprinter II or other IBM graphics printer, execute the MS/DOS command `graphics` prior to running TestGraf, and then pressing `Shift PrtScr` when the desired screen display has been generated. If this doesn't work because your printer is not compatible with an IBM graphics printer, there do exist several commercial screen capture packages for various printing devices. A caution, however: these are memory-resident programs (called TSR's for short) which often use up a sizable portion of the 640K of RAM available on most MS/DOS machines, and you may have trouble running TestGraf at the same time.

Finally, in many applications there is a basis for grouping or segmenting a set of examinees of respondents. These subgroups may be defined by variables such as gender, language, ethnic, or racial group, or some combination of these. The data within each subgroup can be analyzed by TestGraf, and TestSave is designed to facilitate this. The results from the analyses of subgroups can then be compared graphically by using the program TestComp. This topic is taken up in detail in Section 8.

4.4 Details on Displays

4.4.1 The Variable Displayed as the Latent Trait

TestGraf does not use the numerical values of the scores x_a; it first replaces these by their ranks, and then replaces the ranks by the corresponding standard normal quantiles prior to smoothing. It is permissible to do this because one cannot measure a latent variable like proficiency or trait θ in the usual sense. Rather, one can only know its values to within any transformation which preserves rank order, called a *monotone transformation.* That is, we can also use some alternative value $\eta = g(\theta)$, so long as function g is rank order preserving. This issue is discussed in Ramsay (1991).

As a consequence, we are free to use any variable that we like in the plots as the independent or abscissa variable, so long as it preserves the order of the scores x_a used as a basis for the smoothing step. We call the proficiency or trait variable θ displayed in the plots the *display variable.*

TestGraf permits four types of proficiency or trait measures θ to be used as the independent variable in the plots:

Normal: (N) These are the standard normal quantiles used as a basis for smoothing in TESTGRAF, and are familiar to psychometricians as a means of quantifying the latent trait score. Users already familiar with most parametric item response models may be happy with this choice.

Expected Number Correct: (E) This means the expected or average number of correct items that an examinee at a particular proficiency level will achieve. If the data come from a psychological scale, expected scale score is used. This quantity is usually an order-preserving or monotone transformation of the standard normal quantile scores. TestGraf defaults to using this as the independent variable because it is easy for most users to interpret or assess relative to their experience with other tests.

The expression for this score is as follows for multiple choice examinations:

$$\eta(\theta) = \sum_{i=1}^{n} P_{iR}(\theta) \tag{1}$$

where $\eta_{iR}(\theta)$ is the probability of the correct response on item i at proficiency level θ. For scales or for mixtures of multiple choice and scale items, on the other hand, the expression is

$$\eta(\theta) = \sum_{i=1}^{n} \sum_{m=1}^{M_i} w_{im} P_{im}(\theta) \tag{2}$$

It can happen that this expected number correct or scale score fails to be completely increasing in θ, and in this case TestGraf warns the user that the resulting plots may be unsatisfactory. This event is rare, however, and tends to affect the plots only for extreme proficiency or trait levels. It is almost always curable by increasing the smoothing parameter by a small amount.

Formula: (F) This is the expected value of the formula scoring scheme, and is used for multiple choice exams only. The formula score is actually a kind of scale score in which correct options receive the numerical weight of 1 and incorrect options are assigned weights $-1/(M_i - 1)$, where M_i is the number of options for item i. This type of measure of proficiency is often used by testing corporations to attempt to control for the chances of getting an item right by straight guessing.

Observed: (O) TestGraf uses this quantity, denoted by x_a as a basis for ranking examinees. It can come from two sources: (a) by the actual number correct, formula score, or scale score on the test or scale, or (b) input from some other source. Using the observed score as the display variable can be especially useful in the latter case, where one often wants to plot the estimated curves as a function of these input scores. Note, however, that observed scores having large amounts of tied observations will yield rather ugly plots with sudden jumps in the curve. This is especially a problem with scores coming from tests

> or scales with a small number of items. If this option is used with the Beck Depression Inventory data on the distribution diskette, for example, which has only 21 items, the results are much less pleasing than they are with the Introductory Psychology test data which comes from 100 items.

TestGraf identifies these choices for the display variable by the letters `N, E, F,` and `O`, respectively. The user has the option of changing the display variable prior to any display, and in the display of the option characteristic curves, each display can be repeated with a new choice of display variable.

TestGraf also permits some control over how the display variable is plotted by permitting the user to specify the plotting range and the number of tik or hash marks. This can be especially useful in comparing results between different groups of examinees or respondents.

The number of evaluation points $\theta_q, q = 1, \ldots, Q$, for which the curve values are computed can be set as high as 101 in TestGraf. For most purposes the default of 51 gives sufficient accuracy, and higher values necessarily involve more computation. Unlike previous versions, this version no longer requires that Q be a power of 2.

For the normal display variable, the evaluation points are equally spaced and range from -2.5 to 2.5. Using fixed limits greatly facilitates the comparison of results between groups having different numbers of cases. Only 1.6%, or 16 in 1000 cases will fall outside of these limits, of which half will fall above 2.5 and half below -2.5. It would be a rare set of data large enough to yield useful information outside of this range.

The range and spacing of the other display variables, such as expected test or scale score, will vary from one set of data to another, since the distribution of these variables depends on the characteristics of the test.

4.4.2 Summary of Display Options

Once TestGraf has finished computing or setting up the analysis, it presents a menu of display options. Most of these are static, and one is dynamic. This menu of options is as follows:

```
1 Probability of choice curves for each item
2 Expected total correct -- proficiency curve
```

```
3 Distribution of expected total scores
4 Information function for test
5 Relative credibility curves for each examinee
6 Factors of item curves
7 Maximum likelihood proficiency estimates
8 Tetrahedron display for each item
9 Quit
```

A display can be selected from this menu by

- moving the highlight bar to the desired option using the up or down arrow keys or a mouse, followed by pressing the `Enter` key,
- double-clicking the mouse on the desired item,
- entering the number of the option followed by Enter, and
- entering the first character of the description (lower or upper case) followed by `Enter`.

The static displays are described briefly as follows, and in more detail afterwards.

Option characteristic curve plots: For each item the option characteristic curves $P_{im}(\theta)$ are displayed along with other useful information. Figures 1 and 2 in Section 3 are examples in the context of examination data. For scales where each option has a numerical weighting, this option also displays the expected item score as a function of trait level.

Expected and Standard Deviation of Total Score: There are two plots in this display:

1. The left display shows expected total score as a function of the display variable. For examination data expected total score is the expected number of items correctly answered as a function of proficiency. For scales it is the the expected total scale score as a function of trait level. If the display variable is expected score, defined by either expected number correct (1) or expected scale score (2), then the horizontal variable is also expected score, and

this plot will be a straight line and hence uninformative. This plot is useful along with the Normal display variable option, and especially if TestGraf issues a warning message to the effect that expected score is not a monotone transformation of the standard normal quantiles.

2. On the right appears the estimated standard error or sampling standard deviation of the total score as a function of proficiency or trait level. This can be denoted by $\sigma_T(\theta)$.

Distribution of expected total scores: This display shows the distribution of total scores in terms of a smooth curve called a *probability density function* indicating the relative probability that various score values will occur.

Test information function: The amount of information in the test about proficiency, denoted by $I(\theta)$. The standard error of a maximum likelihood estimation of trait score is also plotted, and if the display variable is `N`, a reliability function is displayed.

Examinee credibility plots: For each examinee or respondent, the likelihood or relative credibility of the true proficiency or trait level being at value θ is shown. This offers not only an indication of the best estimate of proficiency or trait level, but also provides an indication of how precisely this is estimated on the basis of the responses given by this particular examinee or respondent.

Principal components of item characteristic curves: A principal components analysis of the shapes of the correct-answer characteristic curves is carried out, and the curves themselves are plotted at the positions defined by their principal components scores.

An additional option causes TestGraf to put to an output file the maximum likelihood estimate of proficiency or trait level for each examinee. A maximum likelihood estimate is in effect the value of θ which yields the highest value of the relative credibility curve shown in option 5 above. It makes use of all the available information in the data to position an examinee or respondent along the $\theta-$axis.

We now discuss each of these displays in greater detail.

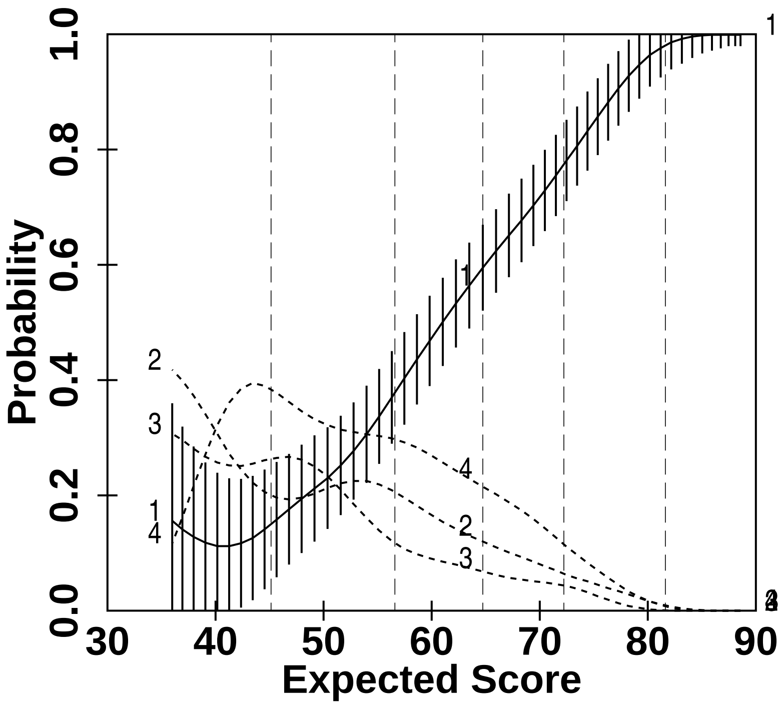

Figure 6: Option characteristic curves for item 29 in the Introductory Psychology Examination.

4.4.3 Option Characteristic Curve Plots

The plot differs slightly depending on whether the data are for a multiple choice item or for a scale item. The plot in Figure 6 is for one of the better items in the Introductory Psychology exam. In this plot the default display variable, expected number correct, is used as the independent variable.

For multiple choice items the plot has the following elements:

1. The solid curve shows the probability that an examinee of ability θ will choose the correct option. This is often referred to as the *item characteristic curve*, especially when the data consist only of whether the item was correctly answered or not. We see in Figure 6 that that the least proficient examinees have only about a 0.15 probability of getting this item right, and that only the very strongest examinees get it right with certainty. A good item has a correct-option characteristic curve which increases rapidly over some range of proficiencies, and this

item has fairly steep slope over the entire range.

2. The vertical solid lines indicate estimated 95% confidence limits for the value of the correct option curve at particular points. These are referred to as *pointwise confidence limits*, to distinguish them from confidence limits for the entire curve. Notice that these limits are broader for very low numbers correct because there are few examinees with proficiencies in this range, and hence there is not as much data defining the curve location as there is for more typical proficiency values. Over most of the range of θ, however, a sample size as large as 379 defines the curve value fairly well, that is to say within about ± 0.1.

3. The dashed curves indicate the probabilities of choosing each of the incorrect options. We note here, for example, that option 2 is the favorite wrong answer (called a *distractor*) for the weakest examinees, but that option 4 is the strongest competitor of the correct answer for most proficiency values. Ideally all distractors should be chosen by some range of examinees, and this is the case here.

4. The fact that this and the other option characteristic curves are only defined over the range of about 36 to 88 in Figure 1 reflects the fact that these are the smallest and largest expected numbers correct for the standard normal quantiles used for these data. In other words, the most proficient examinee in this sample for this particular test is only expected to get 88 out of 100 right on the average. Why not 100? Well, a perusal of the item displays reveals a number of items for which even the best examinees have a probability of choosing the correct option that is substantially less than one (items 7, 80, 96 and 99, for example.) Consequently, we can expect that even the best examinees will get about 12 items wrong on the average.

5. Also shown on this plot are vertical dashed lines indicating various *quantiles* of the distribution of actual or observed number correct. Thus, for example, only 5% of the examinees obtained less than 45 correct, and the median number correct (the 50% line) is at 65. Only 5% of the examinees exceeded 83, so that the bulk of the observed numbers correct for this exam are concentrated in the relatively narrow range of 55 to 75. These quantile markers do not change from one

item display to another, since they are characteristics of the test as a whole.

6. It is possible to label each item with something other than a number, such as a short character string. By default the item number is also used as a label. Note, however, that the item number is, properly speaking, the index of the item among those actually analyzed, but it often happens that the label does not correspond to the item number on the original exam. This can occur, for example, when the examiner decides to omit a particular item from the analysis.

7. The biserial correlation is a traditional measure of item performance. It has serious limitations, however, and can be high for items which are relatively uninformative, especially where the probability of correct response for weak examinees (called somewhat inaccurately the *guessing level*) is high. Values of the biserial correlation exceeding 0.4 can be regarded as high, those between 0.3 and 0.4 as not too bad, and values less than 0.2 as fairly poor. The biserial correlation for item 29, not shown in Figure 6, but shown on the screen, is 0.54.

8. When the display variable is normal, three other values are also displayed. The are three indices, a, b and c, derived from a parametric model for item characteristic curves very widely used in the psychometric literature (Lord & Novick, 1968; Lord, 1981). This model is often called the *three-parameter logistic model.* It has the form

$$P(\theta) = c + (1 - c)\frac{\exp[1.7a(\theta - b)]}{1 + \exp[1.7a(\theta - b)]} \ . \tag{3}$$

We may also put it in the form of the *log odds ratio:*

$$\log \frac{P(\theta) - c}{1 - P(\theta)} = 1.7a(\theta - b).$$

In this model the parameters play the following roles:

- c is the lowest value that the curve attains. Although it is often called the *item guessing parameter*, in fact even the weakest examinees seldom guess at random among options, in which case the probability of a correct response would only be $1/M$, M being the number of options. For item 29 this value is 0.14, indicating a rather low probability of success for the weakest examinees.

- a is a slope parameter determining how rapidly the curve rises from its lowest value, c, to 1.0. Values of a of 1.0 or more are considered very good. It is referred to as the *item discrimination parameter.* Its value for item 29 is 1.00.
- b is the proficiency value at which the curve is midway between the lowest value c and 1.0, and thus measures the difficulty of the item. Values of b much in excess of 1.0 indicate rather difficult items, while values much below -1.0 indicate a fairly easy item. This is the *item difficulty parameter.* The difficult parameter of item 29 is -0.03, suggesting a difficulty level in the middle of the proficiency range.

Estimates of these parameters can be viewed as useful numerical summaries of important aspects of the correct option characteristic curve. In TestGraf these estimates are fairly crude, and if one is primarily interested in estimating these parameters, TestGraf should not be seen as a substitute for a more serious analysis of the data using the logistic model by programs such as LOGIST (Wingersky, Patrick, & Lord, 1988) or BILOG (Mislevy & Bock, 1982). In fact, the whole point of TestGraf is to replace numerical summaries by the curves themselves.

The item display differs in some respects if it is of scale type, for which numerical weights are available for each option. For scales, two plots are produced for each item: the first or left plot contains the option characteristic curves presented in the same way as for examination data. However, confidence bands are not displayed for any option, and there is no display of the descriptive statistics on the right of the display. An example is given in Figure 5 in the previous section.

The second or right plot for scales consists of the expected scale score for this item as a function of trait level. This plot appears on the right of the screen. For an item contributing strongly to the total scale score, this function will vary strongly over trait level, and in the appropriate direction. If, on the other hand, option choices have little to do with the trait being assessed by the scale, the expected scale score will show little systematic variation with trait level, or will show variation that is inconsistent with the trait. Confidence limits are displayed for this curve.

4.4.4 Test Characteristic and Score Standard Deviation Curves

For examination data this plot shows the expected number of correct items as a function of proficiency scores. When expected number correct is the display variable, the relationship is simply a straight line connecting number correct to number correct, and therefore of little interest. If you want to see what the transformation of the standard normal quantiles θ_a that generates expected number correct, then select the standard normal quantiles as the display variable. The notation $\eta(\theta)$ is often used for this function, and the equations defining this value are (1) and (2).

Figure 7 shows the test characteristic function for the introductory psychology data. Note that the relationship has least slope for extreme values of the standard normal quantiles, but is nearly linear elsewhere. This is fairly typical, and cases of nonmonotonicity usually involve a failure to be strictly increasing in the extremes. The problem can often be cured by increasing the bandwidth or smoothing parameter slightly.

When the data being analyzed arise from a questionnaire or scale with numerically weighted options, this option displays the expected total scale score as a function of proficiency. As above, the relationship will only be nonlinear and interesting if the curves are displayed as a function of normal quantiles.

In the context of examination data, the number of correctly answered items for examinees having a particular proficiency value θ is a random variable, in part because different examinees or even the same examinee on different occasions cannot be expected to make exactly the same choices. The standard deviation of these values is therefore also a function of θ, denoted by $\sigma_T(\theta)$, and indicates the standard deviation of this conventional measure of examinee proficiency.

Figure 7 indicates that the standard error is greatest for examinees at around the 50 mark, where it is about 4.5 items out of 100. This translates into confidence limits of about 41 and 59 for a examinee getting 50 items correct, using the values about 2 standard errors up and down. The standard error is about 3.5 items for examinees at the 80 mark, corresponding to

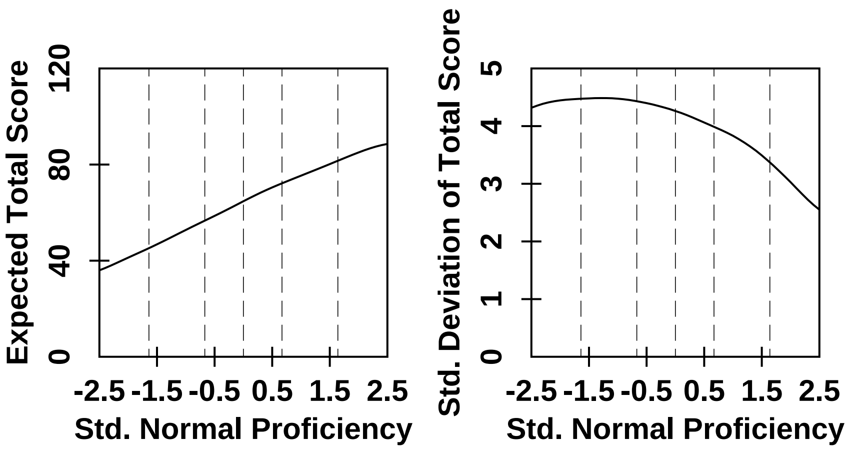

Figure 7: The test characteristic function for the Introductory Psychology Data appears in the left plot, and the right plot contains the corresponding standard error of the function value. The test characteristic function is the expected number of correctly answered items as a function of proficiency or trait value.

confidence limits of 73 and 87.

For scale data, the standard error curve refers to the total scale score.

4.4.5 Probability Density for Total Score

This display supplements the previous display by indicating the distribution of total test or scale scores. The probability density function specifies how probable scores are by the height of the function, and the best known example of a density function is the famous normal density, the "bell" curve. Of course, actual distributions are seldom as neat as the normal, so this display is designed to show characteristics of the actual distribution of scores.

Figure 8 gives the density of scores for the Introductory Psychology exam. We see that scores in the 60 to 70 range are the most probable, and that probability trails off more gradually below this region than above. This is termed *negative skewness*, and is a consequence of the exam having relatively more easy items than hard ones. We also see that the probability goes to near zero well below the upper limit of 100, indicating that there are a substantial number of items in the test that even the best examinees cannot be expected to get right.

4.4.6 Test Information, Trait Standard Error, and Reliability Functions

The test information function, denoted by $I(\theta)$, indicates the amount of information in the test about proficiency at various proficiency levels. It is a sum of *item information functions*,

$$I(\theta) = \sum_{i=1}^{n} I_i(\theta) \ , \tag{4}$$

and an item information function $I_i(\theta)$ indicates the amount of information about the trait score that is provided by a single item. The item information function, in turn, is defined as

$$I_i(\theta) = \sum_{m=1}^{M_i} \frac{(dP_{im}/d\theta)^2}{P_{im}(\theta)} \ . \tag{5}$$

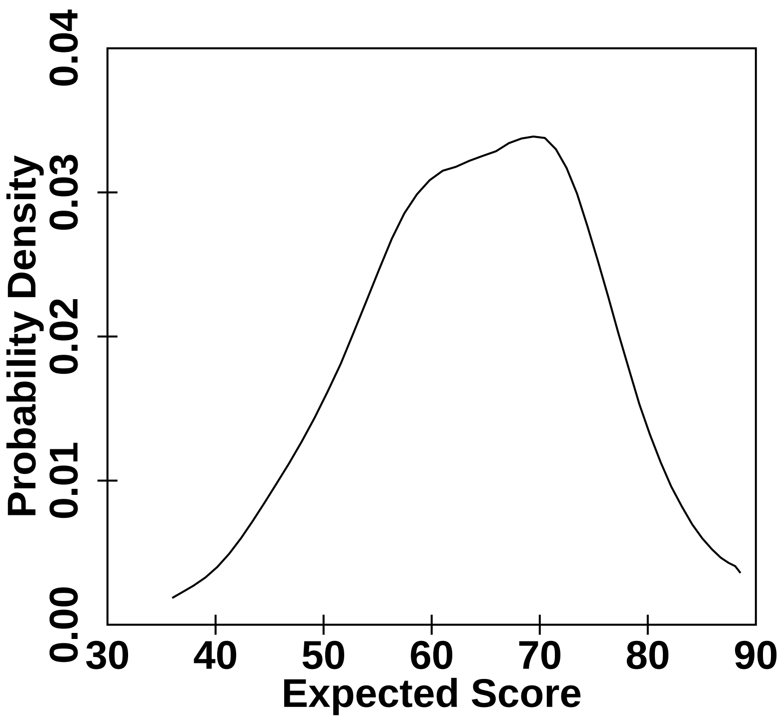

Figure 8: The probability density function for the expected number correct for the Introductory Psychology exam.

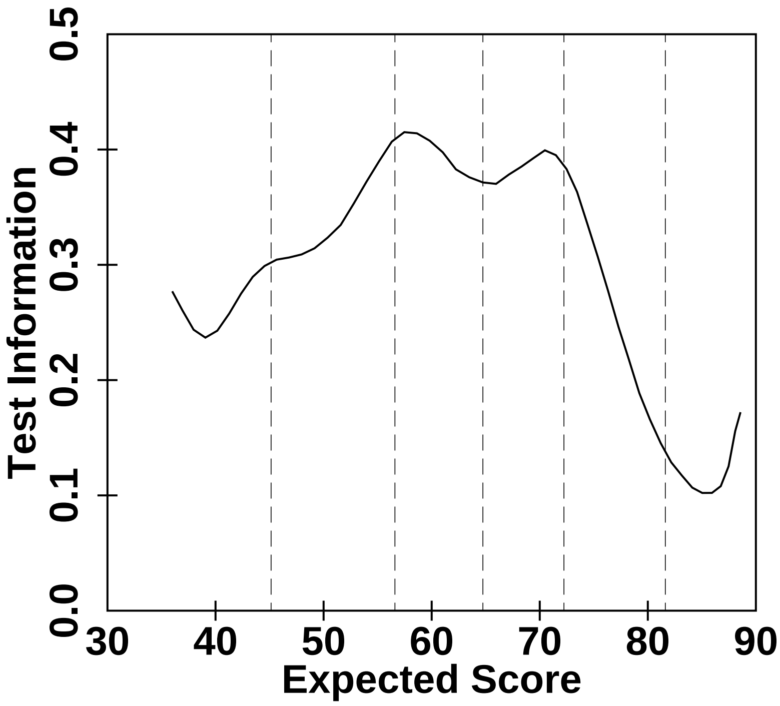

Figure 9: The test information function for the Introductory Psychology Exam displayed as a function of expected number correct.

In the case of dichotomous or right/wrong scored items, this equation simplifies to

$$I_i(\theta) = \frac{(dP_i/d\theta)^2}{P_i(\theta)[1 - P_i(\theta)]} \tag{6}$$

where $P_i(\theta)$ is the value of the option characteristic curve for the correct option.

From these definitions we can see that an item is informative for proficiency or trait value θ if the slope is large for one or more options, as measured by the size of the derivative $dP_{im}/d\theta$. Thus, the information function is related to the concept of *item discrimination* used in connection with parametric item response models. Figure 9 shows the test information function for the Introductory Psychology test as a function of expected number correct.

For this test we see that the test is most informative for low proficiency examinees performing at around the 50 items correct level. This is primarily due to the fact that the test contains a very large number of easy items, which tend to convey information only for low proficiency levels. The test information falls off beyond 70 since it lacks many highly discriminating items for high proficiency examinees.

The sharp increases at the extremes of proficiency should be ignored; they are an artifact of the transformation $\eta(\theta)$ from standard normal quantiles to expected number correct. This transformation usually has near zero slope for extreme values of θ, and since estimating $I(\eta)$ requires dividing $I(\theta)$ by $(d\eta/d\theta)^2$, $I(\eta)$ typically blows up. Actually, this is not unreasonable, since examinees with very high values of θ can be expected to get nearly all items right, so that the test gives nearly infinite information about their true number correct, which is nearly 100. A similar argument holds for examinees at the lowest extreme of θ.

Test information functions $I(\theta)$ are probably easier to interpret if viewed using standard normal quantiles as the display option. This option also facilitates comparing these curves with similar curves computed from parametric models. Figure 10 shows the same information function displayed in Figure 9, but relative to standard normal quantiles. We see in this plot more clearly that this test is most informative for examinees of lower proficiency ranges.

When comparing the information curves for two tests or scales having differing number of items, it is more useful to divide $I(\theta)$ by the number of items, giving the *average information per item*, $n^{-1}I(\theta)$. When plotted against standard normal quantiles, the value of the average information for multiple choice examinations will tend to be less than 1 for any value of θ, and if it approaches unity, the test can be considered as unusually powerful, at least for those values of proficiency. We see by dividing the information by 100 in Figure 10 that the average information peaks at about 0.6 for examinees scoring at a bit below the mean, gives some information even for the weakest examinees, but drops to near zero for examinees in the top 5%. This is, again, because there are very few really difficult items in the test that are at the same time highly discriminating.

It is typical of scales that the information function is less peaked, so that the power of the scale to assess trait level is distributed more evenly over the range of values of θ. Figure 11 indicates the scale information function for the Beck Depression Inventory for a clinical sample. These data are discussed further in Section 8.

One of the most important applications of $I(\theta)$ is to estimate the standard

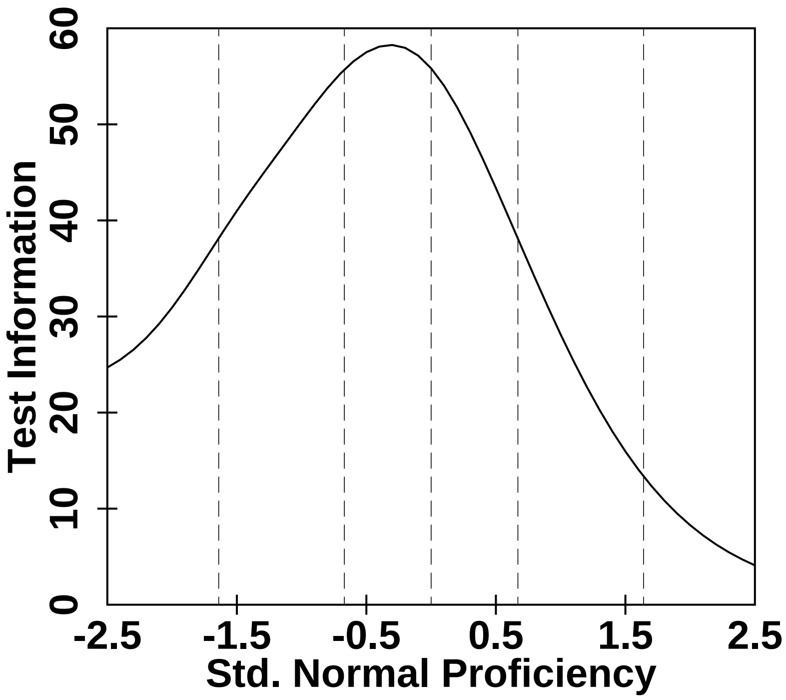

Figure 10: The test information function for the Introductory Psychology Exam displayed as a function of standard normal quantiles.

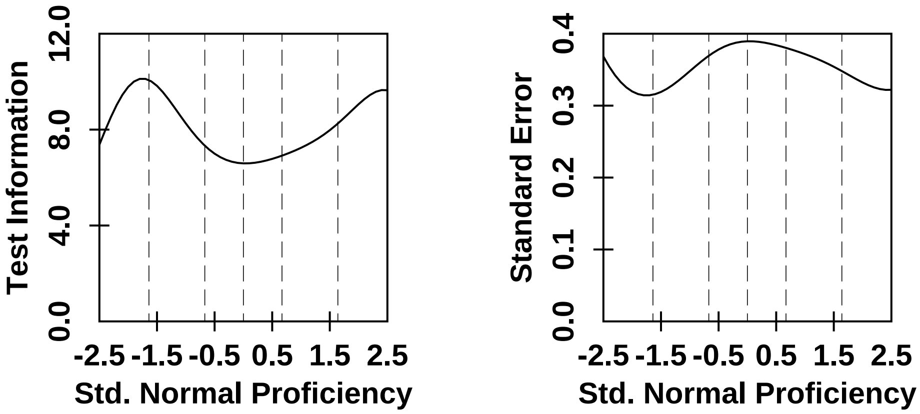

Figure 11: The scale information function for the Beck Depression Inventory for a clinical sample.

error of an efficient estimate of θ itself, an efficient estimate being one which makes best use of the information in the test. This standard error is defined in terms of $I(\theta)$ as follows

$$\sigma(\theta) = 1/\sqrt{I(\theta)}\ . \tag{7}$$

From this relation, we see that the larger the value of $I(\theta)$, the more precisely θ will be estimated, and, indeed, the information function is essentially a measure of the *precision* with which θ can be estimated from a set of responses to the test items.

Figure 12 displays the standard error function for the Introductory Psychology data, and we see standard error is about 1.7 for proficiencies in the range 55 to 75, which includes the central half of the examinees. But the standard error shoots up to nearly 3 for examinees scoring at around 85 due to the poor quality of information in this test for these stronger examinees.

Another useful function derived from the information function is the *reliability function*. In classical test theory, the reliability of a test is defined as

$$\rho^2 = \frac{\sigma_\tau^2}{\sigma_\tau^2 + \sigma_e^2}$$

where σ_τ^2 is the variance of the true scores and σ_e^2 is the variance of errors in estimating true scores averaged across the whole test. In modern test theory, true score τ can be interpreted as the latent trait value θ. When θ has a standard normal distribution, this means simply that $\sigma_\tau^2 = 1$. Above we noted that the error variance with respect to θ is given by $1/I(\theta)$, and this leads to the reliability function

$$\rho^2(\theta) = \frac{1}{1 + 1/I(\theta)} \tag{8}$$

The big difference is now that error variance is estimated *locally*, meaning that it is allowed to vary with levels of θ. Thus, we have replaced the scalar concept of reliability used in classical test theory with a function indicating the repeatability or stability of a test for various values of θ.

Figure 13 shows the reliability function for the Introductory Psychology exam. We see that reliability for this long test is excellent, being around 0.98, for students scoring at around 50, where the test is the most powerful. But the reliability falls off quickly as ability increases because of the lack of high quality difficult items.

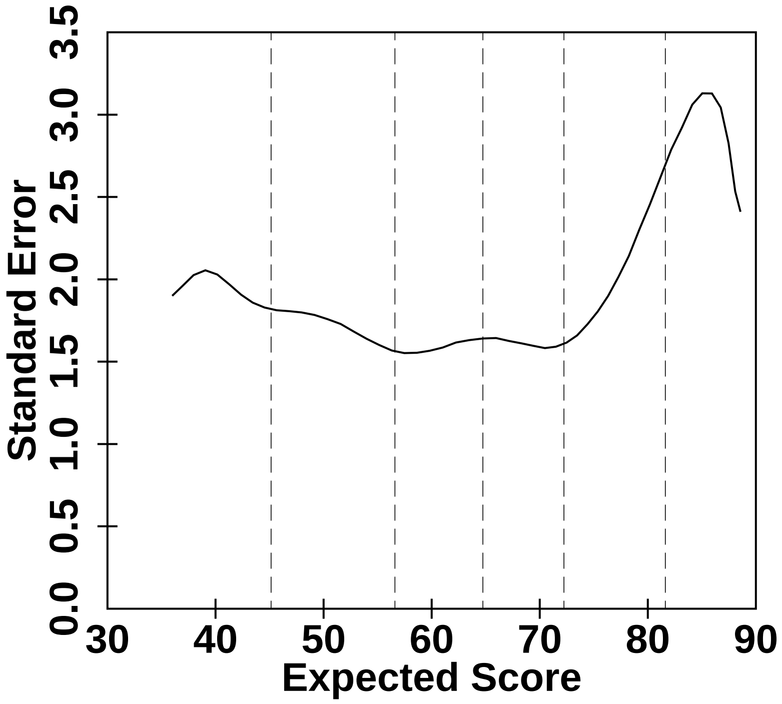

Figure 12: The standard error of proficiency estimation for the Introductory Psychology Exam. This is defined as the reciprocal of the square root of the test information function.

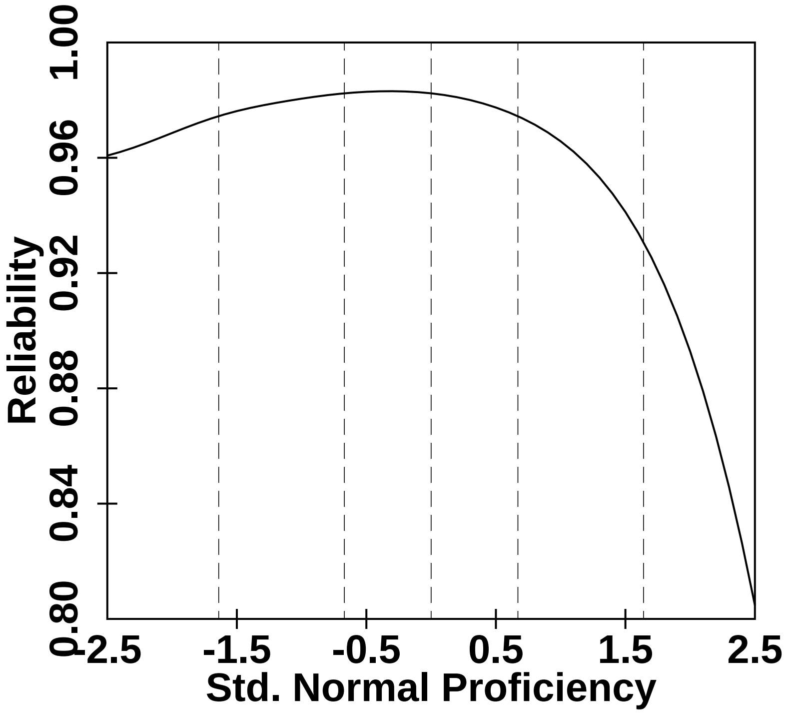

Figure 13: The reliability function for the Introductory Psychology data.

4.4.7 Examinee or Respondent Plots

The display for a typical examinee is shown in Figure 4. The curve plotted shows how likely it is that a specific examinee has a actual or true number correct score given his pattern of responses on this test and given that the option characteristic curves are as indicated. The curve has been normalized to have a maximum value of 1.0, and is called the *relative credibility curve.* The proficiency value for which it reaches its maximum is called the *maximum likelihood estimate* of proficiency for this examinee, denoted by $\hat{\theta}_a$. This estimate is based not only on how many items were answered correctly, but also on

- whether the items answered correctly were difficult or easy,
- whether the items answered incorrectly were difficult or easy,
- whether the correctly answered items where of high quality or not, and
- whether the options chosen for incorrectly answered items were typical of stronger or weaker examinees.

This, the maximum likelihood estimate makes use of much more information than the conventional number-right score, and will tend to be a more accurate estimate of proficiency.

The screen display also shows the observed number correct as a vertical dashed line, and the numerical value is given at the right of the display, but this has been removed in Figure 4. When there is a substantial difference between the point of maximum curve value and this line, it is probable that the pattern of option choices for incorrectly-answered items gave important additional information about proficiency. This can be seen in Figure 4.

The width of the base of the curve indicates how precisely the proficiency of the examinee is estimated. For some examinees the relative credibility curve will be much wider than for others, and can on occasion also be bimodal. This indicates an response pattern giving a mixed message: the examinee passed some tough items, which therefore indicate high proficiency, and at the same time failed some easy items, suggesting lower proficiency. This can happen when the examinee knows some part of the material well and another part poorly. The curve rightly reflects the resulting ambiguity about the examinee's true proficiency.

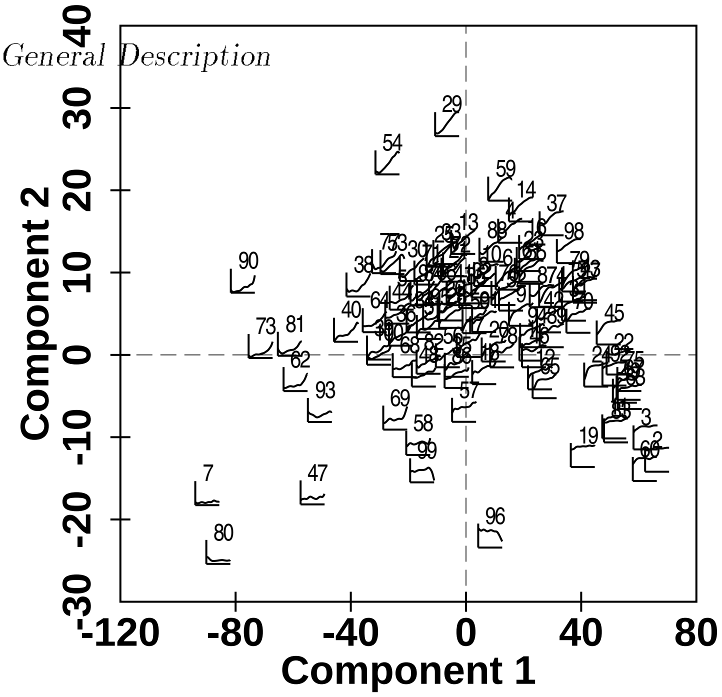

Figure 14: Principal component plot for the Introductory Psychology Exam. The correct option characteristic curves are positioned according the first two principal component scores resulting from a functional principal components analysis of these curves.

When the data are scale responses, the independent variable is trait level rather than proficiency, but otherwise the display is the same.

4.4.8 Principal Component Plots

The purpose of this plot, shown in Figure 14, is to display all of the correct-option characteristic curves or all of the expected item scores simultaneously so as to show relationships among them. This is done by a principal components analysis of the values of the curves at each point of curve evaluation $\theta_q, q = 1, \ldots, Q$. Prior to the analysis, the average curve is calculated across items, and subtracted from each item characteristic curve. In other words, the principal components analysis is carried out on the *centered* item characteristic curves.

In this analysis, evaluation points correspond to the role played by variables in conventional applications of principal components analysis, and items play the role of subjects or replications.

The results of the principal components analysis are:

1. principal component loadings or coefficients which indicate dominant types of variation among the curves,

2. eigenvalues which indicate how much variation each type accounts for, and

3. principal component scores, indicating for any specific item how much of each type of variation is to be found in that curve.

The display in Figure 14 shows the actual item characteristic curves plotted according to their first two principal component scores. An examination of these curves shows that:

- the first principal component, plotted as the horizontal axis, represents item easiness since the easiest items are to be found on the right and the most difficult on the left,

- the second principal component, the vertical axis, corresponds to item discriminability since items low on the plot tend to be have shallow of low slope, while those high in the plot have high slope,

- actually, though, the items distribute themselves along an inverted "U" or horseshoe, and discriminability is more nearly indicated by whether the item falls above or below this curve,

- while the large majority of items have moderate slope and are of medium difficulty, there is a tail to the lower right of items which are extremely easy and therefore necessarily have shallow slope,

- there is a corresponding tail to the lower left of items having very low curve values for all abilities and hence also failing to discriminate,

- items 29 and 54 stand out from the pack as being unusually discriminating,

- items 7 and 80 have near zero discriminability, and are extremely difficult, and
- finally, items 96 and 99 are outliers in that they actually have negative discriminability.

Thus, the principal components plot tends to be a useful overall summary of the composition of the test, at least as far as its correct-answer curves go. The display in Figure 14 is fairly typical of most academic tests, and it is also usual to have only two dominant principal components, reflecting item difficulty and discrimination, respectively. It is possible, however, to display up to five principal components.

4.4.9 Dynamic Probability Tetrahedron Display

This display, an example of which is shown in Figure 3, uses motion to facilitate the viewing of characteristics of a three-dimensional display.

First, a few words of motivation. TestGraf works with what is known as a *latent trait* model for test and questionnaire data. This model postulates an unobserved one-dimensional continuum of some sort which determines how probabilities of choice vary across examinees. Each examinee is assumed to have some level θ_a on this continuum, and his/her probabilities of choosing among options are a function $P_{im}(\theta_a)$ of this value.

However, θ is not like a conventional independent variable that is used in familiar statistical models like regression analysis. Rather, it's values can only be inferred from the data since they are not observed directly. Consequently, once a set of values have been inferred or set up, they can then be transformed by any arbitrary order-preserving or monotone transformation. This is because there is always a corresponding transformation of the functions P_{im} which will exactly compensate for any such change in θ.

What this means in practice is that it is only the *values* of the functions P_{im} that are determined from the data, while by contrast only the rank order of their arguments are known. It is for this reason that TestGraf can take an arbitrary decision to replace any computed or observed values x_a by the quantiles of the standard normal distribution. It is also the reason why the user of TestGraf has a choice of variables for the horizontal axis or abscissa in the displays that TestGraf produces. Indeed, the two choices that are offered could be expanded at will to any type of variable, subject only to the restriction that the ranking of examinees or respondents be preserved.

Thus, one would like a display of the variation in the probability values P_{im} across examinees that tends to hide the role of the argument or domain variable θ. This is precisely the purpose of the dynamic display option.

The probabilities of choices among the options for a particular item i must sum to one. There are M_i of these options, so that in principle the probabilities of choice for a specific examinee/respondent and item define a point in a space of dimension M_i. In fact, however, the coordinates of this point are not free to vary: they must both sum to one, and be nonnegative. Consequently, the space of all possible probabilities defines a subspace of dimension $M_i - 1$. For example, for two choices, the possible probabilities satisfying $P_1 + P_2 = 1$ must fall on a line segment on the plane joining the two extreme probabilities (1,0) and (0,1), and hence define a one-dimensional subspace in the plane. Similarly, the space of possible probabilities among three choices defines an equilateral triangle of dimension two with vertices at the extreme probabilities (1,0,0), (0,1,0), and (0,0,1). And the probabilities of choices among four options define a regular tetrahedron as shown in Figure 3 of dimensionality three. In general probabilities for M choices define an $M - 1$ dimensional subspace called the *probability simplex.*

Thus, restricting our attention to four choice situations, the probabilities of choice for each examinee define a point within this three-dimensional structure. Each vertex of this tetrahedron, which has four edges and four faces, corresponds to an extreme situation in which all the probability is loaded on a single choice. For points in the interior, or within a face or edge, the closer the point is positioned to a vertex, the higher the probability that the corresponding option will be chosen.

Because TestGraf assumes that the variation among probabilities of choice is determined by variation in a single latent variable, the probabilities estimated by the program must fall along a single curve or track within this tetrahedron. TestGraf allows us to view this curve directly.

TestGraf only displays four-choice situations in this way. If there are more than four choices actually available, it computes the four choices which are the most active in the sense of having the largest collective probability of being chosen, and then computes *conditional probabilities* conditioning on choice among only these four. This seldom loses any real information since experience tends to show that in a very wide range of situations people tend to eliminate all but a few options.

Movement of the display on the screen is essential to see its three dimensional structure. The relative motion of parts of a three-dimensional visual field relative to each other is called *motion parallax*, and is the principal cue to depth that we use in ordinary life.

TestGraf provides a variety options for controlling the motion. They are:

- Pressing and holding down the space bar causes rotation about the vertical axis. For most viewing purposes this is just what is needed.
- The speed of the motion can be decreased by pressing and releasing the `G` key. Press it many times to produce a large decrease in speed. This can be useful if a very large number of points are involved, or if the display screen has a very long decay time (as is true for currently available overhead projector tablets).
- The speed increases each time the `F` key is pressed.
- The `Home`, `End`, `PgUp`, `PgDn`, `Ins`, and `Del` keys all produce rotation around different axes and in different directions.
- The arrow keys displace or translate the display.
- The `+` and `-` keys expand and shrink the display, respectively.
- The numeric keypad `5` key returns the display to the size and orientation with which it is first displayed.

TestGraf displays a question mark at the point in the center of the tetrahedron, corresponding to completely equal probabilities of choice among the options.

A point is plotted for each level of proficiency or trait used in the regular two-dimensional plots. The point corresponding to the lowest level of proficiency is plotted with a small square to mark it off from the others.

Here are some things to look for in these displays:

- A basic requirement of a reasonable test item is that the track terminates at or near the correct answer. Similarly, a reasonable scale item should have a track tending in the direction of the response given the largest numerical weight. Have a look at item 96 of the Introductory Psychology test to see a failure.

- The total variation or length of the track is very important. The individuals with the lowest proficiency or trait levels should be far from those with the highest. Items 4 and 5 in the Introductory Psychology test are examples. By contrast, very easy test items (such as item 3) have very short tracks concentrated close to the correct answer. The highly convoluted track for item 7 which never gets close to the correct answer suggests great confusion among the students. For numerically weighted scale items, we should look for the track to do a tour of the vertices in the appropriate order.

- How close is the curve to a face or an edge? Points on faces imply virtually zero probability of choosing the option associated with the opposite vertex, and points on edges imply that two options have been effectively eliminated from consideration.

- The relative spacing of the points indicates the speed with which probabilities of choice are changing. See the contrast between items 4 and 5, for example, among the best students.

- Sudden changes in direction can also be indicative of an abrupt change in knowledge state, understanding, or psychological condition. Item 95 of the psychology test shows that the weakest students are busy eliminating one of the wrong answers rather than moving directly to the correct answer. Only when this option is removed from consideration does the track move in the appropriate direction.

5 How to Run TestGraf

In this section further details are provided on how to run TestGraf, as well as the setup program TestSave.

For most purposes the user will not need to know anything beyond a rudimentary acquaintance with the MS/DOS operating system or an equivalent. The programs TestSave and TestGraf have built-in capabilities for allowing users to choose among options for analysis and display. However, the files used by these programs can also be edited separately by any editor, such as the editor EDLIN provided with MS/DOS, versions up to 4.0, or with EDIT in MS/DOS version 5.0, to permit various modifications of the processing.

The user of TestGraf should be familiar with the MS/DOS file naming convention. Each file name consists of one or two parts:

- *filename*: up to 8 characters in length
- *extension*: up to 3 characters following a period

While the second extension part is optional, it is wise to use an extension to every file name to indicate the nature of the file. TestGraf, TestSave, TestLasr and TestComp assume that all file names have an extension, and use this extension to identify the nature of the file. These programs use the *filename* component to identify the data being analyzed. Both parts of the file name may be in either upper or lower case.

We first have a look at how to set up the raw data file for processing by TestSave.

5.1 Running TestSave

TestGraf cannot operate directly on the raw data file. Rather, one must first run the setup program, TestSave. Separating the data setup from the actual analysis has the advantage that the fairly lengthy processing required to prepare the raw data for processing by TestGraf need not be repeated each time TestGraf is run. Since one may wish to use TestGraf many times on the same set of data, this results in an important saving of time.

Moreover, TestSave also defines various critical pieces of information about the data such as whether items are of multiple choice exam or of scale type.

5.1.1 Setting Up the Raw Data File

The first task prior to an invocation of TestSave is to set up the raw data file correctly. This will not usually involve a great deal of effort, but this phase does require the use of an editor such as the `edlin` or `edit` programs in DOS.

TestSave requires that the data be set up in a particular way in order to determine obvious features of the data such as the number of examinees or respondents, the number of items, the number of lines of data per examinee or respondent, and the organization of the data within these lines. Otherwise TestSave is used interactively.

The raw data file on which TestSave operates must have the extension `.dat`. An example would be `psych101.dat`.

TestSave prompts for confirmation of various critical aspects of the data that it estimates, such as the number of lines of data per case and the FORTRAN format specification for reading in the data. The user can either accept this estimated or default value, or change the value in question. If the data are set up properly, the user has usually only to confirm the decisions made by TestSave prior to the program proceeding to carry out its job.

A typical setup is shown in Figure 5.1.1, and has the following components:

- The first line contains the number of items to be analyzed and the the number of characters of label identifying examinees. The number of label characters cannot exceed 8. These values are in free format, meaning that they can appear anywhere on the line separated by one or more blanks or by a comma. If there are no labels for examinees or respondents, the number 0 must nevertheless appear after the number of items on this first line.

- The next line or lines contains a key or prototypical response. This is required for two purposes:

  - in the context of examinations, it indicates which options are to be considered correct.

  - TestSave also uses this key line or lines to generate an estimated FORTRAN format specification for the data. For this reason, the

```
100   7
0000000    142422124334112313422212141313ll . . .
           324321334412223232112342443141 42 . . .
8600001    142422224334112123422242141143ll . . .
           344322321312123112123342443144 42 . . .
8600002    142142424334112313412242341343ll . . .
           244221344442323121124242113444 42 . . .
                         .
```

Figure 15: The setup of the raw data file `psych101.dat` for the Introductory Psychology data. The first line contains the number of items and the number of characters of examinee label. The second two lines contain the exam key, or the correct option indices. These lines also serve as a template indicating how the data are to be input. Each subsequent pair of lines contains the actual responses of an examinee, as well as the examinee's label.

key line or lines should be set up to indicate the typical organization of the response data for actual examinees or respondents.

The first characters in the key line indicate the location of the examinee label, if any. For the Introductory Psychology data the key label is 7 zeros followed by 4 blanks,

```
"0000000    "
```

The seven label characters for the Introductory Psychology data are followed by 60 characters designating the correct options. Because there are more than 60 items, the key is carried on to the next line after an initial 11 blank spaces. This arrangement is identical to that used in the lines containing the actual responses of examinees, and thus acts as a template.

- Subsequent lines for the introductory psychology exam file contain the actual responses for each examinee in exactly the same format as for the key. That is, the key is just a set of responses for an examinee getting all items correct. The first line for each examinee contains a series of up to 7 characters which identify the examinee . The next line for the same examinee begins with 11 blanks, followed by the correct option choices for the remaining 40 items.

Columns in the key line(s) not used for label information or for response information should be left blank.

The setup for the Beck Depression Inventory data was given in Section 3.

5.1.2 Subgroup Selection

The first question asked by TestSave is whether one or more items are to be used as a basis for selecting a subgroup, rather than being used as part of the test. Subgroup analysis is taken up in detail in Section 8, and the remaining discussion of TestSave in this section will assume that this is *not* the case, or that all items will be analyzed. In this situation, the default option of "no" is accepted.

5.1.3 Identifying the Data Format

The first task faced by TestSave is to use the key line(s) to determine how many lines there are per examinee or respondent, so that it can count their number. TestSave will estimate this number based on the number of key lines it needs to read to process label and response characters. TestSave then asks you if this is the correct number of lines before proceeding. For some types of data setup, this value may be too low. It can happen, for example, that the file contains much other information besides responses to items to be analyzed, and, therefore, there are more lines per case than you need to contain the label and key characters. Or, in the case of the McGill exam scoring system, the scoring software uses four lines per examinee no matter how many items there are. But remember that the key lines must have *exactly* the same number of lines, even if blank, as each case uses; the key lines must be a replica or prototype for the lines containing each respondent's data. You can change the number of lines estimated by TestSave if necessary to the correct value when TestSave asks if its estimate of the number of lines per case is correct.

TestSave analyzes these key lines to construct a FORTRAN style format specification. It then prompts you to ask whether its computed format is what you wish, and you can then change it if you want to. If you are not familiar with FORTRAN format specifications, you can probably figure out what to do by noting the following rules for setting up FORTRAN format specifications:

- FORTRAN format specifications are contained within parentheses
- `nx` means "skip n columns", so that `4x` causes the program to skip four columns before reading the next character
- `ma1` means "read m single characters", so that `6a1` causes the program to read the next six characters. Note that blanks are considered characters, and are usually used to indicate omitted responses. The program always expects to read single characters, so that the "1" after "a" can be dropped.
- `/` means go to the next line

For example, the format specification proposed by TestSave for the psychology data is

```
(7a,4x,60a1/11x,40a1)
```

This translates to "read 7 characters (of label), skip 4 columns, read the next 60 characters (responses for the first 60 items), go to the next line, skip 11 columns, and read 40 characters (responses for the last 40 items)." The format `(7a,4x,60a/11x,40a)` would also work.

TestSave has only a limited capacity to construct a format specification, and will have trouble with complex formats, especially if a lot of skipping of spaces is required. If the format specification that TestSave comes up with is incorrect in some way, you can choose to enter your own format specification. Choosing the "no" option opens up a box in which you can enter up to 78 characters of format specification. Don't forget that the format specification begins with "(" and ends with ")".

It can happen that the format is so complex that it cannot fit within the 80 spaces on a single line. For example, if every 6th character is to be used in a multi-line record, TestSave would set up a whole string of `5x,1a1,` sequences that might go beyond 80 characters. If this happens, you will either need to consult a reference on Fortran format specification, or edit the raw data file to remove unwanted characters.

The subsequent lines of the raw data file will contain the actual responses of the respondents or examinees. Be careful here: if your file contains characters in the columns specified in the key lines that you did not intend to indicate an examinee response, TestSave will nevertheless assume that these characters are an actual response. As indicated below, you can reserve a special character to indicate a missing response, although these are usually just left blank. Check your file for any characters besides those identifying options or missing responses. Unwanted characters should be edited out of the `.dat` file before processing by TestGraf. Also, check the final lines of your file; it can happen that unwanted lines have been added along the way. In fact, scan the entire file to check that everything is in order; there is no substitute for checking your data! TestSave will assist you by displaying all the characters that it has identified as indicating responses; scan this display to be sure that there are no unwanted characters.

In summary, the things that you must satisfy in setting up your raw data are:

1. The first line must contain the number of items to be analyzed and the number of characters of label. Use 0 if there are no labels.

2. The next line(s) must contain the exam key. This is the template for reading in actual labels and responses. Label characters, if any, must come first.

3. Subsequent line(s) must contain the responses of examinees in the same pattern as used for the key.

5.1.4 The Omit Character

It is common for examinees or respondents to fail to produce a response for one or more items. This can be indicated in various ways in the raw data file. Leaving this character blank, for example, is a commonly used procedure. Or one may choose some other character such as "*" as a placeholder for omitted responses.

Unless TestSave is told that this omit character is to indicate an omitted response, it will assume that this is a response like any other, and generate a corresponding item option. Now this may be the right thing to do, especially if there are large numbers of omitted responses, as would happen if the test were too long for most examinees to finish.

But TestGraf does handle omitted responses differently than designated responses. The difference between an omitted response and an actual response is important in estimating the option characteristic curves. Omitted responses are bypassed in estimating these curves, butactual responses have their own curves associated with them. On the other hand, an option characteristic curve is generated for every actual designated response, so that if you choose to designate omitted responses with some character and choose not to tell TestSave that this character is to be treated as an omit, an option characteristic curve will be computed for this response, just like the others. This may, of course, be what is desired.

By default TestSave assumes that a blank indicates an omitted response. The last question asked by TestSave is whether this is to be so, and if not, the user can enter the appropriate character. But note that if a nonblank character is used, a blank is then treated as an actual response, and will have its own curve associated with it.

It is important to preprocess the raw data file so that only one character, blank or otherwise, indicates an omitted response. Also to be sure that only the desired characters appear; it is very common in raw data files to find extraneous characters that the user did not know were there.

5.1.5 Identifying Item Types

TestSave asks for a default item type, and all items are set up by TestSave to be this type. Normally exams consist of items that are all of the multiple choice type, and scales are comprised of items all of which are of the scale type, and in this case no additional work needs to be done. However, if the data come from exams or questionnaires where these types are mixed, then an editor must be used after the TestSave run to modify the file with the extension `.itm`, described in detail below. For mixed item types, the default choice would naturally be that of the majority of items.

5.1.6 Command Line Arguments for TestSave

To start TestSave, you type `ts` or `testsave` followed by 0, 1, or 2 arguments. These arguments are used to identify the raw data file, and to construct the names of the files produced by TestSave. They define the main part of the file name preceding the period or dot, called the *stem* of the file name. Consequently, these arguments must be from 1 to 8 characters in length, and the characters must be letters or numbers. They must also be separated by one or more blanks.

Here's what happens depending on the number of arguments:

- no arguments, or you just type `ts`. TestSave prompts you to enter string of from 1 to 8 characters to construct files names. It opens up a window in which you enter these characters.

- one argument, or you type `ts filename`, where "filename" is just an example of a possible set of characters. The argument is used both to identify the raw data file, which is named filename.dat, and to construct file names for the output files. For example, TestSave produces an output file with the extension `.itm`, and its complete name would be `filename.itm`.

- two arguments, or you type, for example, `ts filenam1 filenam2`. The raw data file must be named filenam1.dat, and the output file names will have main parts or stems filenam2. The two-argument invocation of TestSave is useful when multiple subgroups are being selected from the same raw data file.

5.1.7 The Parameter or `.par` File

The next task for TestSave is to set up a file containing all of the information that TestGraf needs to process the files that TestSave produces. TestSave sets up a file with the extension `.par` to set various default values. You could modify this file after a TestSave run with any editor if you want to change the default values that TestSave uses. However, TestGraf will also permit you to modify any of these values when you run it, so that there is no real need to modify this file after the TestSave run.

Figure 5.1.7 displays the `.par` file `psych101.par` set by TestSave for the introductory psychology data.

```
Run
Introductory Psychology Exam           |First  line of title
                                       |Second line of title
379                 Number of examinees or respondents
100                 Number of test or scale items
51                  Number of plotting points for displays
2                   Number of principal components displayed
Expected Score      Display variable option (N, E, F, or O)
Number Correct      Type of scoring (MC exams only) (N or F)
0.33                Bandwidth or smoothing parameter value
                    Compute curve values
                    Input proficiency or trait  values
                    Only output .pwo plotting file
                    Control plotting limits for abscissa
psych101.exo              File for data in examinee order
psych101.ito              File for data in item order
psych101.out              File for output of results
psych101.prb              File for curve values
psych101.itm              File for item info
psych101.lab              File for examinee labels
psych101.abi              File for input proficiency/trait values
                    Omit character
```

Figure 16: Setup of the TestGraf parameter data file `psych101.par` for the Introductory Psychology data.

5.1.8 The Item or `.itm` File

TestSave also produces a file that contains information about each item. This file may well require modification prior to further processing by TestGraf. The information recorded for each item consists of:

1. a label of up to 8 characters for the item,

2. the type of the item (multiple choice or scale),

3. in the case of multiple choice options only, the index of the correct option,

4. the number of options for the item, and

5. information for each option, consisting of:

   (a) the character associated with the option,

   (b) a label of up to 8 characters for the option,

   (c) a numerical weight for the option,

   (d) the frequency with which the option was chosen, and

   (e) the frequency with which the item was omitted.

This file is designed to be modified by the user, and the beginning of the file contains some instructions about how the file is to be modified. It is a good idea to at least look at the contents at this time to ensure that the information has been set up appropriately by TestSave, and it can also be helpful to print this file as a record of item characteristics, such as the frequency with which options are chosen.

Some of the modifications that are typical for the `.itm` file are

- specifying a label for an item,

- changing the type of the item from the default choice indicated in the TestSave run,

- specifying a label for each option for a specific item,

- modifying the numerical weights generated by TestSave, and

- adding options even though TestSave did not detect anyone choosing an option.

The initial section and the lines associated with the first two items for some Beck Depression Inventory data are given in Figure 5.1.8. Notice the labels "Sad" and "Future" given for these items, as well as that they are specified as scale type. The correct option is indicated as "None" because scale type items do not have correct options.

5.1.9 Other Files Set Up by TestSave

Finally, TestSave sets up various files that are needed by TestGraf. These are described briefly as follows:

`.exo` **file:** contains the examinee choices organized by examinees, so that TestGraf can input any examinee's responses as a single batch. This is a binary direct access file, and cannot be edited.

`.ito` **file:** contains the examinee choices organized by items, so that TestGraf can input all the choices for any item as a single batch. This is a binary direct access file, and cannot be edited.

`.lab` **file:** contains labels of up to 8 characters for examinees.

`.abi` **file:** contains proficiency values for each examinee in the event that these are to be input by TestGraf rather than using number correct as a proficiency measure. The first line of the file must contain a FORTRAN format specification, and subsequent lines contain the proficiency values in this format, one per examinee. If maximum likelihood estimates of proficiency or latent trait score are requested in a TestGraf run, these are output to this file. See Section 9 for a discussion of how to use this `.abi` file in the iterative refinement of trait scores.

You might want to try TestSave now for the introductory psychology data by typing

```
ts psych101
```