A SAS PROGRAM FOR REPRODUCING THE RESULTS FROM THE ITEM PROCEDURE
Shu-Yeng Wong
Indiana Univ. - Purdue Univ. at Indianapolis

PROC ITEM was a SAS user supported program (Chilko & Hobbs, 1986) to perform the item analysis for multiple-choice tests. The ITEM procedure was written in PL/I and was included in the SUGI Supplemental Library which was available with Version 5 base SAS software. Also, it was only applicable under CMS and OS.

With the coming of SAS 6.06, PROC ITEM is no longer available. The transition from version 5 to version 6 creates a real problem for the users who have relied on this procedure to do item analysis (Rutherford & Wong, 1990). The solution to the problem is to write a SAS 6.06 program to do what PROC ITEM does. A SAS item analysis program has been designed to serve this purpose.

INPUT
A SAS dataset must be created before using the item analysis program. The first observation in the dataset is the answer key for each item while the rest are the actual responses made by the individual who took the test.

In order to associate the obtained score with the examinee, a string variable to signify the identification for each observation must be included in the SAS dataset. The name of this string variable used in the program is ID. Therefore, ID must be one of the variables existing in the SAS dataset.

Also included in the SAS dataset are the numeric variables representing the responses for the test items. The variables are named Q1, Q2, up to Qn, where n is the maximum number of items.

The names of variables used in the item analysis program are given above. In case different names were used in creating the SAS dataset, it will be necessary to rename them before they can be directly employed by the program.

Besides to prepare a SAS dataset, there are two IF-THEN statement in the item analysis program used to be modified. One IF-THEN statement assigns the number of test items to macro variable QNO, while the other assigns the name of the SAS dataset to macro variable DATASET.

OUTPUT
The item analysis program will produce five tables:

1. Score Report
The first table basically shows the scoring results over all examinees. The scores included in the table are as follows:
   a) the number of correct answers.
   b) the number of wrong answers.
   c) the number of omitted answers.
   d) the percentage score, which is the ratio of the number of correct items to the number of total items multiplied by 100.
   e) the t score, where the mean is 50 and the standard deviation is 10.
   f) the ranking, which shows the examinee's relative order among the others.

2. Table of Difficulty Level
The second table lists the difficulty level for each item. The difficulty level is the percentage of examinees who pass the item. A difficulty level of 75 indicates that seventy five percent of the observations answered the item correctly, while a difficulty level of 25 shows that only twenty five percent of the observations have the right answer. Obviously, an item with a difficulty level of 75 is easier than an item having a difficulty level of 25.

3. Percentile Table
The third table lists the percentiles from 1 through 99 which divide the distribution into 100 equal parts. A percentile is a score at or below which a given percentage of the observations is found. For example, say a student has a test score of 35, which is equivalent to the fiftieth percentile; this means that this student scored higher than 50 percent of the students in the test. In other words, fifty percent of all students taking the test have scored lower than or equal to 35. Because they are easy to understand, percentiles are commonly used in interpreting the test scores.

4. Table of Internal Consistency
The fourth table shows the coefficients of KR20 (Kuder-Richardson-20) and KR21 (Kuder-Richardson-21). KR-20 and KR-21 are measures of internal consistency which assume that all items measure the same general ability. The Kuder-Richardson formula gives values that vary between 0 and 1. A value approaching 1 indicates a high degree of internal consistency. KR-21 is a perfectly accurate estimate of KR-20 when all items have the same level of difficulty. While the items vary in difficulty, KR-21 always gives an underestimate of the internal consistency. The more the items vary in difficulty, the greater the underestimate which occurs.

5. Item Analysis Results
The fifth table presents the percentage of responses to each alternative for the upper
third and lower third groups. The difference between the two groups in terms of the percentage of response to the correct answer is the index of discrimination. This table also includes the point biserial correlation and its significance test.

PROGRAMMING CONCERNS

The item analysis program consists of six procedures. Each procedure is written as a macro program to perform a specific task. A list of the macros and their functions follows:

1. Macro READIN

READIN is the first and only mandatory macro program to run. It takes a SAS dataset provided by user as input. The structure of this dataset is mentioned in the above INPUT section. From this dataset, Macro READIN will create three datasets which can be utilized by the following procedures:

   a) Dataset ASUB is the most fundamental one which consists of each individual's responses to the test items and his/her computed statistics.

   b) The second dataset, DIFF, contains the variables DIFF1 through DIFFn which preserve the values of the difficulty levels for items from 1 through n. This dataset will be used in the macro program DIFF to produce the difficulty level table.

   c) The third dataset, MAXMIN, is used in the macro program SCORING to produce the score report. The variables contained in this dataset are some basic calculations obtained from the scoring process; these are the numbers of right, wrong, and omitted items. Identification and percent correct are also included.

2. Macro SCORING

SCORING is used to report the test results for each observation in the dataset. It takes dataset MAXMIN, which is created from macro program READIN as input, and then applies Proc Standard to obtain the t-score (m=50, sd=10) and applies Proc Rank to obtain the ranking and the percentile rank for the test score. After all the related scores are gathered, Proc Print is used to display the score report.

3. Macro DIFF

Macro DIFF will show the difficulty level for each test item. It takes dataset DIFF, which is created from macro program READIN, as input. Dataset DIFF consists of only one observation with n variables, where n is the number of test items. The variables included in the dataset DIFF are actually the difficulty level for each item. In order to rank the difficulty level, Proc Transpose is used first to transpose the dataset DIFF and then Proc Rank is applied to do the ranking. The difficulty level for each item and its related ranking are listed by the Proc Print.

4. Macro PCTILE

PCTILE takes advantage of the new feature in Proc Univariate in version 6 to create an output dataset which stores the percentiles from 1 through 99. A data step then sets up a display format to show the contents of this dataset.

5. Macro KR20

Macro KR20 is the routine to compute the internal consistency for the test. The coefficients of KR-20 and KR-21 are the measures of the internal consistency.

Macro KR20 takes the dataset ASUB as input. First, it uses Proc Means to create an output dataset which includes number of cases, mean, standard deviation, maximum and minimum scores. Thereafter, a data step uses this output dataset to calculate the related statistics and write the report.

6. Macros UPGROG, POINT, and ITMOUT

Three separate macro programs, UPGROG, POINT, and ITMOUT, are combined to reproduce the item analysis output from Proc Item.

UPGROG forms two groups first, the upper third and the lower third groups. The upper third group includes the observations with their scores higher than or equal to the percentile 67, while the lower third group has the observations with scores lower than or equal to the percentile 33. With the new enhancement from version 6, Proc Univariate is capable of writing the percentiles in an output dataset for use, just like Proc Pctl did in the SUGI Supplemental Library. This new feature is really an advantage for our programming purpose.

Within the UPGROG, there is another macro program, called LISTPCT. This program will compute the percentage of responses to each alternative, for the upper third and the lower third group separately.

POINT calculates the point biserial correlation for each alternative and its significance test. For the computation of point biserial correlation within each alternative, five datasets (MAXMIN1 through MAXMIN5) consisting of descriptive statistics (mean, standard deviation, number of cases) for each alternative for all the items are created by using Proc Means. All these datasets are then integrated in a data step to compute the point biserial correlation, t-test, and p-value.

ITMOUT takes the datasets created by the
previously executed macro programs, UPLGGRP and POINT, to produce a table exactly like the one produced from Proc Item in order not to confuse the old users.

REFERENCES


NOTES

SAS is a registered trademark of SAS Institute Inc., Cary, NC, USA.

ACKNOWLEDGEMENTS

Special thanks to Phil Kizer of Integrated Technologies at IUPUI for his assistance in developing this paper.
APPENDIX

/* Purpose: scoring and item analysis to reproduce the results */
/* from ITEM procedure */
/* Author: Shin-Tung Yung */
/* Integrated Technologies */
/* Indiana Univ. - Purdue Univ. at Indianapolis */

/* --- Initialization --- */
/* 1. assign the number of item to macro variable QNO */
/* 2. assign the name of a SAS dataset to macro variable DATASET */

DATA ASUB (DROP=TOTAL PX PQ WONG OHIT)
  temporary datasets which subject to the changes.
  DATA ASUB(DROP=TOTAL PX PQ WONG OHIT)
  DIFF(ITEMSCORE=TOTAL SCORE ITEM SCORE TOTAL ITEM SCORE WONG DIFF)
  TOTAL(ITEMTOTAL=TOTAL ITEM TOTAL TOTAL ITEM TOTAL)
  OHIT=(ITEMOHIT=ITEM OHIT OHIT ITEM OHIT)
  WONG=(ITEMWONG=ITEM WONG WONG ITEM WONG)
  TOTAL(ITEMTOTAL=ITEM TOTAL ITEM TOTAL TOTAL ITEM TOTAL)

PROC FORMAT DATA=DIFF OUT=DIFF* DIFF)
  TITLE 'DIFFICULTY REPORT'; TITLE4 'DIFFICULTY REPORT';

/* create a dataset which includes the t-score, rank, percentile rank */
/* print the score report */
/* print the difficulty level report */
/* obtain the rankings for the difficulty level */
/* list the difficulty level for each item */
/* create output dataset which contains the percentiles 1 thru 99 */
/* procedure to print out the percentiles from 1 thru 99 */
/* procedure to print out the percentiles from 1 thru 99 */

/* temporary datasets which subject to the changes. */

/* create a dataset which includes the t-score, rank, percentile rank */
/* print the score report */
/* print the difficulty level report */
/* obtain the rankings for the difficulty level */
/* list the difficulty level for each item */
/* create output dataset which contains the percentiles 1 thru 99 */
/* procedure to print out the percentiles from 1 thru 99 */
**MACRO USER**

// prepare the statistics for the computation of EX-20 */
PROC MEANS DATA=AUTO N MEAN STD MIN MAX NMISS=M; 
VAR SCORE; 
OUTPUT OUT=VARINFOR NOCANS=0 MAXVAR=NAME=SCORE STD=STD; 
TITLE; 
/* calculate the EX-20, EX-11 and Spearman-Brown prophecy */
DATA NULL; SET MAXVAR; 
EX11=SUM(SCORE); (1-EX11)/(1-EX11); EX11=SUM(SCORE); (1-EX11)/(1-EX11); EX12=SUM(SCORE); (1-EX12)/(1-EX12); 
STD=STD*STD; 
END; 
TITLE; 
/* defines the number of test statistics */
PROC PRINT; 
/* creates lower third and upper third groups, UPLOI and UPLO2 */
DATA NULL; SET MAXVAR; 
LISTPCT(UPLOI); XLISTPCT(UPLO2); 
TITLE; 
PRC THE GROUPS 
OUTPUT; 
/* calculates the 0-20, 0-21 and scale-Brown prophecy */
CALL EXPERT; 
TITLE; 
PROC PRINT; 
/* creates upper and lower third groups, UPLO1 and UPLO2 */
DATA NULL; SET MAXVAR; 
LISTPCT(UPLO1); XLISTPCT(UPLO2); 
TITLE; 
PRC THE GROUPS 
OUTPUT; 
/* calculate the percentage choosing each alternative */
MACRO LISTPCT(7); 
DATA AFI; SET AFI; 
NO=AFI+1; T1=AFI; ITEM=61; ANSWER=061; OUTPUT; 
END; 
PROC SORT; 
BY ITEM; 
PROC FREQ DATA=AFI; 
TITLE; 
**MACRO LISTPCT**

**DATA MANAGER**

// Compute the point biserial correlation **/**
END N=1; STD 5; 
DATA MAXVAR; SET AFI (KEEP=Expanded отделы (масштаб, VAR=AFI)); 
STD=STD/10; 
END; 
PROC MEANS DATA=MAXVAR N MEAN STD MIN MAX NMISS=M; 
VAR SCORE; 
OUTPUT OUT=VARINFO; 
MEAN=SCORE; (1<=AFI<=10); MAXVAR; NMISS=M; 
STD=SCORE; (*<=AFI<=10); MAXVAR; NMISS=M; 
END; /* DO NL */