An Educational-Assessment Database for the State of Tennessee

John F. Schneider¹, University of Tennessee

Abstract

The Education Improvement Act of the State of Tennessee mandates that the influence of school systems, schools, and teachers on measures of student learning be assessed. Students are tracked across years since the assessment is made on a gain basis, instead of the raw score or percentile basis. Although most students have the same identification information from year to year (e.g., name, Social Security Number, etc.), many do not. Even students who do not match contribute information to the analysis, and are included in the database.

The database was begun using data from the 1989-1990 school year with approximately 500,000 records. Currently, over 2,300,000 records are in the database. Beginning last year, data about teachers is being collected, such as subject taught, etc. Naturally, these two databases must link together by student.

When this project was started, many “experts” told us creating this database would be impossible. They must not have been aware of the capabilities of the SAS® System. The abundance of functions in the data step has overcome many obstacles present in the data. That, combined with the other capabilities and products available within the SAS System has allowed this project to proceed.

Introduction

The Education Improvement Act (EIA) enacted by the Legislature of the State of Tennessee took effect July 1, 1992. Part of the EIA mandates that the influence of school systems, schools, and teachers on measures of student learning be assessed. Since the EIA mandates that the assessment be made on the gain achieved, instead of their raw score, students must be followed across years. To the best of the knowledge of everyone involved in this project, this is the largest (and only) database of its kind in the country.

¹John F. Schneider, Data Analyst, Statistical and Computing Services, 225 Morgan Hall, Agricultural Experiment Station, The University of Tennessee, Knoxville, Tennessee, 37901-1071.
BITNET: PB99198@UTKVM1.BITNET
Internet: PB99198@UTKVM1.UTK.EDU (or) john@aesrs1.ag.utk.edu

I would like to thank Dr. W. L. Sanders and S. Paul Wright for reviewing this paper and their helpful suggestions.
Before this project was started on a state-wide basis, two years' worth of data for six "representative" counties in Tennessee were used as a pilot project. It was with this sample that we began to realize exactly what was involved. To make matters worse, this sample data set did not contain all of the problems which would later be encountered in the data.

Computer Resources Required

This project was started on a CMS system (IBM® 3081), but, with only two years data, it was beginning to outgrow the resources the university computing center was able to allocate. To merge the first two years, the process had to be done system-by-system, with the large systems being matched on Saturdays and Sundays. Then, when the analysis was run, the jobs kept getting bumped by the system, because they requested too much core and CPU time. It was possible to tell that they had indeed started, they would not complete. By the time the third year's worth of data was received, it was becoming apparent some other solution was necessary, both because more disk space was needed on the mainframe for the database, as well as more RAM for the analysis. Therefore, during the summer of 1992, the project was moved to an IBM RISC System/6000™ model 560 running AIX®. This has since been upgraded to a model 580, with 256 Mb RAM and 5 gigabytes disk space. With 4 years worth of data, the database occupies approximately three-quarters of a gigabyte of disk space. With 4 years worth of data, the database took less than 14 hours of real time, running version 6.07. The NONOTES options was in effect to reduce the size of the SAS log, therefore, the CPU time required for the matching is uncertain. Based on previous passes through the data, it was somewhere between 11 and 12 hours. Since the 1994 matching job will be run under version 6.09, the time should be cut by 25 to 30 percent, based on other benchmarks run on the machine. (The matching program is usually the only job in the system at the time.) For several years now the data for the whole state has been matched simultaneously by student.

Some Beneficial SAS Software Features

INDEXES: Indexes are used quite extensively. For example, after the matching has taken place, they enable one to quickly subset out data based on a particular school system, year, grade, or an individual student, etc.

THE DATA STEP: The data step in the SAS System provides many functions, both character and numerical. Also many of the statements in the data step, such as MERGE and SET (combined with BY), are also of great benefit, as much of the matching and combining algorithms have already been programmed. Similarly, Data Set options allow variables to be dropped, kept, or renamed "on the fly" without the need for a separate, intermediate data step to carry out the task.

MACROS: The SAS Macro language has been invaluable. This has allowed code to be conditionally executed, as well as repetitive code to be handled conveniently.

LABELS: On a personal note, prior to this project, I had heard many presentations at SUGI in which the authors had extolled the virtues of labels (both data set and variable). I had never quite believed them, because up to this point, all of my data sets had only a few variables, and the data sets were only used once, twice at the most. Now, with datasets containing many variables, and many data sets, labels have been invaluable, and I can see why other people have extolled their virtues, and now, I too, extoll their virtues.

FORMATS: User-defined formats have allowed one-digit or one-character codes to be stored in the database, therefore saving valuable space in the database, but yet it is transparently converted to a more complete description. For example, SEX is stored as a one-character variable, but this is mapped to a full description in print-out or when viewing the data with SAS/FSP. The user does not have to remember the different race codes. The DATE format comes in handy for birthdates.

SQL VIEWS: Although not yet implemented, these will allow the various data sets to "appear" as one massive data set to people using the data, even though the data is coming from multiple files.

Match Rate

The first year, it was easy to calculate the match-rate, i.e., the percent merged. One just had to take the total number matched and divide by the total number of records. It was well over 90 percent. In succeeding years, this calculation is not as easy to make. Instead, one must take the number not matched, subtract out the number which can not match, and divide by the total number of records. The match rate has remained substantially the same in subsequent years. Table 1 gives the number of students who have one, two, three and four years worth of data (number of records).

<table>
<thead>
<tr>
<th>Number Of Records per Student</th>
<th>Number of Students</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>282,004</td>
</tr>
<tr>
<td>2</td>
<td>182,538</td>
</tr>
<tr>
<td>3</td>
<td>212,033</td>
</tr>
<tr>
<td>4</td>
<td>256,060</td>
</tr>
<tr>
<td>Total Students</td>
<td>932,635</td>
</tr>
</tbody>
</table>

Of the 282,004 students with only one record, approximately 160,000 have not had a chance to match.
e.g., 10th graders the first year, first or second graders this last year, etc. This leaves about 122,000 (or about 13 percent) that have not matched. There are many reasons, also, why these 13 percent have not matched - they indeed have lived in Tennessee only one year; they have lived in Tennessee two years, but only took the test one year because they were sick the second year; since test papers without names or scale scores are not matched, they could have taken the test two (or more years) but their record was only usable one year, etc.

Organisation of the Database

Early versions of the database contained two files - a set of "matched" records and a set of "unmatched" records. Hence, to match a new year's worth of data, both the "matched" file and the "unmatched" file had to be checked. Then, the unmatched file had to be recreated each year, as some children who were previously "unmatched" were now "matched." This hindrance has been corrected, and now all student records are kept in one large master file. No flags are attached to a record to indicate whether a record is matched or unmatched, as this does not influence subsequent matches in later years, and the analysis software handles this internally in its own way.

The following items are contained in the database. Although, at the present time, all student scores are kept in one large file, this will be broken down into multiple files, each with one year's worth of data.

The Student File: The Student-Score file was begun using data from the 1989-1990 school year with approximately 500,000 student records, and about 600,000 are added each year. With the 1992-1993 test data, over 2,300,000 records are in the database. (See Table 2 to see how many records are present in each grade/year in the database.) Each record in the database contains information only on one year's test data per student. This observation contains basic identification information such as name, birth date, Social Security Number, sex, race, school code, grade, year, and school system code. It also contains several fields for special education information, and finally the 16 scale-scores the student earned on that year's test. A unique ID is assigned to each child, and this follows a student every year, and in every file in the database.

The Teacher File: Beginning with the 1992-1993 school year, data about teachers is being collected. Teachers completed forms essentially listing all students they taught, the subject(s) they taught for each student, the percentage of time the student was in class, and of the time the student was in class, the percentage of time they taught the child. Naturally, these two databases must link together by student.

School Name File: School information, which can get rather detailed, is stored in another file, and is linked to the student file via system/school numbers.

Source of Data

All data comes from the scan forms filled in by either the students or teachers. All student data comes from the scan forms filled in by the students when they take the TCAP tests. The teachers themselves fill in the data for the teacher file, but as attendance information is required, these forms must wait until the end of the school year to be completed.

Observed Problems in the Data

Although most students have the same identification information from year to year (e.g. name, birth date, social security number, etc.), not all do. Several reasons exist for this.

Name: The first two years, very few students included suffixes, such as Jr., III, etc. in their names. Beginning in 1992, they started including these. Also, one year, several systems did not have the students code a middle initial, while the rest of the years those same systems did code in middle initial.

Social Security Number: Since, by law, students can not be forced to give their SSN, some do refuse. Some school systems use their own internal ID instead of social security number.

Duplicates: Although the testing time-window is small, school systems do have some lee-way as to when they administer the test. About 10 - 12 students a year will start (and maybe complete) the test in one system, move to another system and retake all or part of the test.

Others: One year, one school system, merely sampled its students instead of administering the test to everyone. Although not a data matching problem per se, it has caused an undue number of mismatched records. Similarly, one system is located near a large military base. The year of Operation Desert Storm, many students moved away to live with other family members.

One school in Tennessee straddles the county line and so is legitimately one school in two separate school systems. Students from each system feed into this school, and in turn feed back into a school in the system in which they came from. The school keeps its records as if it were two schools (one in each system) but in reality, it is one school.

Solutions: As Ricciuti (1993) points out, about the only way to find mistakes in the database has been to print discrepancies and check them against each other, and "laborious hand checking" of the lists. This has been the way most of these mistakes have been handled, where they can be corrected using SAS/FISP software. Some of these errors can be corrected automatically with programming. Prior to matching, it takes one person between ten and fifteen days to edit one year's worth of data. The data has been getting progressively more accurate each year.
Early on, it was suggested that the answer forms be "pre-slugger," i.e., where a computer fills in the spaces, and the students do not have to fill anything in but answers. However, the reply was that this was not possible.

Likewise, it would have been convenient to have the teacher ID portions of the teacher forms be preslugged. Again, the answer was that this was not possible because the schools did not know how many forms each teacher would need, so they might print too many or may not print enough.

Other Considerations

It is considered more of a "sin" to not match two people than to incorrectly match them.

An over-riding concern is that a student's information remain confidential.

PROC SQL, was tried for the merge process. Unfortunately, PROC SQL causes memory errors when trying to have it match one database containing 800,000 records (the number of unique students in the master in Spring 1993) and another containing 600,000 records (the number to be added). This was under version 6.07 of the SAS System. This error may have been fixed, but has not been tested by us.

The Future

We are anxiously looking forward to SAS Institute coming out with SAS/ACCESS® for Informix (announced last year). Informix will have its advantages for this project, as does the SAS System. Its biggest advantage to us will be that the data analysis software will be able to read the Informix data base directly. Currently, the data is stored in SAS data sets. However, for analysis, it must be exported to ASCII files. The use of Informix will allow the retention of SAS software advantages while eliminating the disadvantage. (And I don't have to rewrite my programs.) Also, there will not have to be multiple copies of the data on disk - the main purpose of SAS/ACCESS software.

Conclusions

The many and varied features of the SAS System have overcome the score of data problems. It can be used to manipulate data files, whether that data is stored in SAS data sets, or data stored in other formats with SAS/ACCESS software. The SAS programming language has also been a much friendlier and easier environment to work in, than for example, C or FORTRAN. It combines, in one package, a "database program," and a programming language for manipulating the database as well as the capability of performing analysis of the data and writing reports.

The SAS System also provides the tools for editing the data prior to merging, in addition to supplying the means of finding the data which needs editing.

References


<table>
<thead>
<tr>
<th>Grade</th>
<th>1990</th>
<th>1991</th>
<th>1992</th>
<th>1993</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td>23,190</td>
<td>23,845</td>
<td>24,669</td>
<td>23,486</td>
<td>95,190</td>
</tr>
<tr>
<td>1</td>
<td>54,489</td>
<td>62,953</td>
<td>60,796</td>
<td>61,684</td>
<td>239,922</td>
</tr>
<tr>
<td>2</td>
<td>57,354</td>
<td>63,147</td>
<td>63,835</td>
<td>65,068</td>
<td>249,404</td>
</tr>
<tr>
<td>3</td>
<td>56,651</td>
<td>64,266</td>
<td>63,203</td>
<td>63,551</td>
<td>247,671</td>
</tr>
<tr>
<td>4</td>
<td>55,769</td>
<td>63,484</td>
<td>64,366</td>
<td>63,347</td>
<td>246,966</td>
</tr>
<tr>
<td>5</td>
<td>55,996</td>
<td>62,220</td>
<td>63,784</td>
<td>64,324</td>
<td>246,324</td>
</tr>
<tr>
<td>6</td>
<td>55,059</td>
<td>62,323</td>
<td>62,527</td>
<td>63,818</td>
<td>243,727</td>
</tr>
<tr>
<td>7</td>
<td>56,191</td>
<td>62,259</td>
<td>63,892</td>
<td>64,213</td>
<td>246,555</td>
</tr>
<tr>
<td>8</td>
<td>56,657</td>
<td>60,287</td>
<td>60,282</td>
<td>61,845</td>
<td>239,071</td>
</tr>
<tr>
<td>9</td>
<td>4,912</td>
<td>5,984</td>
<td>8,454</td>
<td>7,723</td>
<td>27,073</td>
</tr>
<tr>
<td>10</td>
<td>52,030</td>
<td>52,222</td>
<td>51,225</td>
<td>54,029</td>
<td>209,506</td>
</tr>
<tr>
<td>11</td>
<td>2,079</td>
<td>2,926</td>
<td>3,497</td>
<td>3,075</td>
<td>11,577</td>
</tr>
<tr>
<td>12</td>
<td>807</td>
<td>872</td>
<td>1,681</td>
<td>1,073</td>
<td>4,433</td>
</tr>
<tr>
<td>Total</td>
<td>531,184</td>
<td>586,788</td>
<td>592,211</td>
<td>597,236</td>
<td>2,307,419</td>
</tr>
</tbody>
</table>

SAS, SAS/ACCESS, and SAS/FSP are registered trademarks of SAS Institute Inc., in the USA and other countries. IBM and AIX are registered trademarks of International Business Machines Corporation. IBM Risc System/6000 is a trademark of International Business Machines Corporation. © Indicates USA registration.