Report on Creation and Use of Database for Coalition vs Board of Education

Jan de Leeuw Ph.D.
with assistance from
Jose Garcia
Vivian Lew
Stan Bentow

UCLA Statistics Program

January 3, 2012

Contents

Chapter 1

Introduction

UCLA Statistical Consulting was hired to construct a database from data provided by the Data Service Center in Wilmington. Throughout the project our major aim has been to preserve the integrity of the data. Thus we have made as few *coding decisions* as possible.

The process of creating the database consisted of various steps. Data arrived on tapes in the format of ADABAS[1] workfiles. The description of the process of loading, reading, and copying the tapes is described in Chapter 2. UCLA Statistical Consulting, as a rule, performs its analyses and formatting in SAS, the Statistical Analysis System. This means that files had to be translated into the appropriate SAS format. This step is described in more detail in Chapter 3. The next step was to merge the individual files into a large SAS database. This is described in Chapter 4. The final step in the construction process was to translate the SAS database back to ADABAS format. This is documented in Chapter 5.

UCLA Statistical Consulting also had to construct tables from the SAS database for the various experts involved in the case. The process of constructing the tables is described briefly in Chapter 6. We did not do any actual analyses, it was our job to make sure that the database, and the resulting tables, were a faithful representation of the files we received from DSC.

[1] Adabas is a proprietary format used by the database programs of Software AG. ADABAS software can also produce workfiles, which can be read by other software programs as well

Chapter 2

Tapes

2.1 State of Tapes

A total of 66 reel-to-reel tapes were received from the DSC. They are used and in poor condition. They have been assessed by the staff at the UCLA Office of Academic Computing to be at least five years old. They are marked by labels of at least three previous uses and in some instances they are marked as reinitialized and reissued. The tape heads on the Sun3 Server used to read the tapes were cleaned twice of the thick grime produced by the tapes. One tape in particular, 1988 Employee, quickly became useless producing an I/O error stating that the tape header was unreadable.

2.2 Files on Tapes

There were a number of complicating factors, which made efficient handling of the reading, translating, and writing stages difficult.

1. Files were written by IBM oriented database programs in IBM oriented formats. UCLA Statistical Consulting does most of its data analysis and manipulation on Unix systems, and there are many incompatibilities between the two operating systems that had to be resolved first.

2. Three files, “Grid”, “Textbook”, and “Transportation” were in the proprietary ADABAS Upload format and had to be translated into flat file format before using SAS on them. As a favor UC Santa Barbara translated one of the files, Textbook, for the Consulting Center.

3. One complication in the handling of the tapes was in the changing formats occuring on the tapes received. The method of handling arrays of variables depended on the format and which format was used, in some instances, was not always clear. The way arrays appeared were either "interleaved" or "blocked". That is to say either in the form

   $$x_1, y_1, z_1, x_2, y_2, z_2,$$

   or in the form

   $$x_1, x_2, y_1, y_2, z_1, z_2.$$

   Apparently the different places producing the tapes either used the ADABAS Input or ADABAS Output formats. These two formats have different array layouts and the documentation did not indicate this. This had to be discovered by trial and error, and resolving the resulting problems took some time.

4. Generally, the scope in years of the files produced problems in the construction of the data base. Again, solving these problems was time-consuming. There was a gradual transfer from V-SAM to ADABAS over the '80s. The splits occur 1982-1986, 1987-1994 for the Student files; 1982-1988, 1989-1994 for the Course files; 1982-1989, 1990-1994 for the Employee/Incumbent files.

5. The old field names for the '80s files had to be matched up with '90s names. Documentation for this was embedded in Natural programs which had to be requested from the DSC.

6. The Dropout Probability field, according to Joseph Saggione, has not been used in six years. It originates from dropout studies conducted in the mid '80s in the Christina District. He believes SPSS programs external to the DSC environment were used and the values were imported. Further investigation by year would have to be conducted to further pinpoint this fields origins.

Chapter 3

Translating to SAS

3.1 Tool Selection

For our database construction we have chosen to use SAS, the Statistical Analysis System. We could not use the ADABAS format in which the data were delivered, and in which the data had to be written. ADABAS is a commercial product, marketed by Software AG. Its formats are proprietary, and we do not own this software. The reasons for choosing SAS are

1. Able to manage very large amounts of data;

2. Able to translate IBM EBCDIC data to ASCII data[1] for use in UNIX;

3. Able to read IBM packed decimal format[2] (many of Wilmington's numeric variables were stored as packed decimal);

4. Able to store data in compressed SAS form and then able to write out EBCDIC files for use by Software AG;

5. SAS is also widely available on many platforms. In our case, it was available both on the IBM mainframe and the Sun Unix machines;

6. SAS has sufficient data handling capabilities which allow for merging of multiple files.

[1] EBCDIC and ASCII are two different encodings for the alphanumeric characters.

[2] Packed decimals define a compact way of storing numerical data by using non-ASCII characters.

3.2 Procedure

We read the original data tapes from Wilmington at UCLA's IBM mainframe environment using SAS. The data was then output in SAS proprietary format for storage prior to the merge. Once the data was read, we selected the specific variables (e.g. race, sex) needed for analysis.

Thirteen files were created for years 1982-1994 from the Student Master tapes. Thirteen files were created for years 1982-1994 from the Student Course tapes. One file was created from the Student Retention dataset. Three files were created from the IOWA, SAT, and CTBS datasets respectively.

Seven files were created for years 1982-1989 from the Employee-Incumbent tapes. The 1988 Employee-Incumbent tape was not readable. Ten files were created for the 1990-1994 Employee and Incumbent files (the DSC had split the employee datasets into two files after 1989).

Once the data was completely read and all 47 files created, and checked for consistency in various ways, the merging began.

Chapter 4

Merging

4.1 Student Files

We were informed by the DSC that the variable known as "Tracking ID" was unique for each student and should be used as the variable which would link students across years in the Wilmington dataset. However, duplications were found for students who had transfered between schools in a given year. For example, in the 1994 Student-Master dataset, about 5% of the Tracking IDs are duplicates.

If we compare Tracking ID across datasets, we find

- IOWA less than 0.1% duplicates;
- SAT about 2.7% duplicates;
- RETAIN about 44% duplicates;
- CTBS about 0.1% duplicates each.

The merge we had to do required unique ID variables. To meet this requirement, we chose to keep the most recent addition to the database, for the Student-Master 1982-1994 files, the IOWA, SAT, and CTBS files. If a student occurs three times in a given file in a given year, we only use the last occurrence.

The Student-Course 1982-1994 datasets and the Retain dataset had to be handled differently because by definition they would have many duplicate Tracking IDs. The Student Course datasets, for example, were structured in such a way that each line of the dataset represent a single course that a student took in a year. Since students (in middle school and above) frequently enroll in many courses, student information was repeated.

To allow a merge between data from the Student-Master files and the Student Course files, the Student Course files were restructured in such a way that each line of the dataset represents a student followed by ALL of the course information for that student for that year. By structuring the data in this way, the variable count of the course data was increased but a merge was now possible, and analysis (in the sense of table construction) was now possible. Using SAS to tabulate the frequency of Tracking IDs in a dataset we discovered that a student could take as many as 20 courses in a single year. Thus, the restructured Student-Course file contains information on 1-20 courses per student per year.

The Student-Retain dataset was restructured in a similar way. A student could be in the Retain dataset as many as 7 times so the resulting dataset had up to 7 years of retain information for each student.

Once the Student-Course and Student- Retain datasets were restructured, all of the years and student files were merged. In the process of the merge, variables had to be renamed to reflect the particular calendar year the information was drawn from. Additionally, since the school districts radically restructured their database between 1982-1986 and 1987-1994, for the Student-Master datasets and between 1982-1988 and 1989-1994 for the Student-Course datasets, we had considerable difficulty insuring continuity between those two sets of years. Coding to correct for this problems was time-consuming.

The merged student 1982-1994 file was stored in SAS form which allowed us to both create EBCDIC flat files (for export to IBM machines, and later conversion to ADABAS), and create ASCII compatible datasets (for analysis on Unix machines).

This satisfied both our needs and those of DSC.

4.2 Employee-Incumbent File

The employee-incumbent merge required some preparatory restructuring, because of the number of job changes which occurred to the employees in any given year. Using SAS, we calculated that a given employee could change jobs as many as 10 times in a given year. Thus, the employee-incumbent datasets had a large number of repeated observations. Restructuring of the employee incumbent data was similar in logic of the restructuring of the Student-Course data discussed above.

Variables were renamed to reflect both the year the data was drawn from and the job number for the particular employee. For example, if a woman changed jobs 3 times in 1994, she might have 3 different job titles, work locations, salaries. All of this information was made unique for each job for each year, and then merged across years.

4.3 Merging Student and Employee Files

A final merge, involving the employees and students, could (unfortunately) not take place. The reasons are simple. In the Student-Course file there is a "Teacher ID" variable. In the Master-Schedule file there is also a "Teacher ID" variable, and a social security number. We had hoped to use the social security number to match the master schedule to the employee-incumbent information, and then use Teacher ID to match teachers back to students. Unfortunately, the employee social security number on the master schedule file was absent.

The only way to match students to teachers at this point would be a string match on teacher surname. We have experience with string matches in similar situations, and this is a very time-consuming and delicate process. There has not been enough time available to date to do the string match.

Chapter 5

Translating to ADABAS upload

5.1 Production

Flat files were sent to Software A.G.'s Right Sizing Center in Reston, VA. With David del Rio's assistance we wrote two ADACMP programs to convert the flat EBCDIC files to ADABAS compressed files. The flat files contain the following (also compare Appendix B.)

Student/Master 157,695 observations; 9,233 fields; 28,506 bytes/record. Compression was roughly 67% producing an upload file of 1.48 gigabytes.

Employee/Incumbent 31,199 records; 16,632 fields; 81,307 bytes/record. Compression was about 35% producing an upload file of 1.65 gigabytes.

5.2 Some DSC Queries

Subsequently, DSC had some questions about the conversion process, and the files it produced. We give the reasons here for the choices. In none of these cases, however, was the integrity of the database compromised, i.e. all the information in the original EBCDIC version of the file is still available in the ADABAS version. Quotes are from a letter of the defendant to Thomas Henderson.

5.2.1 Alpha

Query:

> Software AG changed all of the fields to alpha. If we were to use this file all numeric fields would have to be converted back.

There was a problem with the numeric fields. SAS handles empty numeric fields differently than does ADABAS. To elaborate on this point, the SAS convention for writing an empty numeric field of length, say n, is to write $n-1$ spaces and a decimal point at the nth digit. ADABAS can not handle spaces in its numeric fields. To make the appropriate conversion Mr. del Rio would have had to have translated these empty fields into all zeroes. This presented two problems. First, empty fields and legitimately zeroed fields would have been grouped together losing information in the process. Second, it would have required another day of programming and computer time by Software AG to do the translation.

Mr. del Rio informed us that he discussed these issues with Mr. Saggione. Mr. Saggione understood the situation, and reported that it would be allright with the DSC to leave the fields as alpha.

5.2.2 Names

Query:

> In every listing we have seen, no DSC data field names used.

The documentation for the student file, sent on November 1st and 2nd, included the comment fields used by the DSC to link SAS' eight character names with their data field names. These comment fields were taken from the DSC's field description list verbatim. It was thought their meaning was clearer and the association with the data fields easy to make since they can be found together on the same line in the DSC's field description listing.

For the employee file,we put the data field name as a comment next to the SAS variable associated with it in the SAS programs that we wrote.

5.2.3 Record size

Query:

> Our disk devices can only handle 7,164 characters.

A pseudo-block size was created due to the unusually large size of the ADABAS files. The block size of a usual file is roughly 3300 characters (bytes) long. Both files required block sizes of 27644 bytes. This requires that the DSC write special programs to read in files with this block size. Again, no action was taken until the matter was resolved between Software AG and the DSC. Mr. del Rio informed us that he brought this to Mr. Saggione's attention, and according to Mr. del Rio Software AG was given the go ahead by the DSC to produce such block sizes.

5.2.4 Fields

Query:

> ADABAS, on any platform ... can only handle 926 (fields).

This was the most obvious concern to us during the ADABAS translation process. The student file contains 9,233 fields; the employee file 16,632. At the time, given the circumstances, there appeared to be no other way to handle the number of fields but to group them together. This was done judiciously. First, arrays fields were grouped together, then fields that made logical sense to be together such as Grid-X and Grid-Y.

These grouped fields can be accessed used the Natural language command "redefine". It is not the most elegant way to do things in Natural but easily done, according to Mr. del Rio. This command was used for this very purpose by the DSC to convert their V-SAM files into ADABAS files before.

Chapter 6

Analysis

6.1 Goal of Analysis

The analysis consists of providing expert witnesses with tables. The tables depicted the racial composition of districts and schools with regard to outcomes of interest. These tables provide the actual number and percentages of students who fall within each of these categories. In addition, the tables include marginal (or conditional) percentages.

6.2 Extracting Information from Database

The first task in creating these tables is extracting the relevant information from the student and employee files. This is a time consuming task, basically because the files are very large. In our environment, it requires planning in efficiently handling the (networked) computing resources, dividing the jobs over different machines, dividing output over different printers, and so on. The “largeness” of the files affects the analysis in many ways.

1. There are many distinct variables for each year;

2. Each variable consists of many fields (for example, the variable ”teacher certification” has 100 fields within a year);

3. There are many years of data (from 1982 to 1994). For example, the variable ”extra-curricular activity” had 13 years of information, and each year had 20 extra-curricular activities per student per year.

Thus, extracting the relevant variables was a nontrivial task.

6.3 Conversion and Running of Tables

Once the variables were extracted, programs were written to convert the the files into a usable format. The programs also checked for consistency of the fields. This required quite a lot of “looping”, since each field within each variable had to be checked for valid values. Given the size of the arrays (13 years, dozens of variables, and up to 100 fields within each variable), this procedure again used up a lot of computer resources.

Once the variables are extracted and converted, the final procedure is analyzing the data using SAS. The final file is then used to create the tables requested by the experts.

Appendix A

List of Files

Item	*Description*	*Tape*	*Airbill*
1	File 125 Stu-Iowa	5026	2046380674
2	File 82 SAT	5034	
3a	File 81 CTBS	5024	
3b	File 81 CTBS	5025	
4	File 52 Emp/Inc	5016	
5	File 52 Emp/Inc	5018	
6	File 52 Emp/Inc	5017	
7	File 52 Emp/Inc	5020	
8	File 52 Emp/Inc	5021	
9	File 52 Emp/Inc	5019	
10	File 52 Emp/Inc	5022	
11	File 52 Emp/Inc	5023	

Item	*Description*	*Tape*	*Airbill*
12	File 103 Grid	5004	2046380685
13	File 130 Stu-Textbook	5003	
14	File 31 MLI	5002	
15	File 119 Stu Planning	5001	
16	File 110 Master Schedule	5033	
17	File 52 Employee	5014	
18	File 53 Incumbent	5013	
19	File 52 Employee	5012	
20	File 53 Incumbent	5011	
21	File 52 Employee	5010	
22	File 53 Incumbent	5009	
23	File 53 Incumbent	5015	
24	File 52 Employee	5008	
25	File 53 Incumbent	5007	
26	File 52 Employee	5006	
31	File 100 Student	5064	2046379694
32	File 112 Course	5055	
33	File 112 Course	5056	
34	File 112 Course	5057	
35	File 112 Course	5058	
36	File 100 Student	5065	
37	File 112 Course	5060	
38	File 112 Course	5059	
39	File 100 Student	5066	
40	File 112 Course	5061	
41	File 112 Course	5062	

Item	*Description*	*Tape*	*Airbill*
42	File 105 Transportation	5031	2046379672
43	File Suspensions	5030	
44	File Suspensions	5029	
45	File Suspensions	5028	
46	File Suspensions	5027	
47	File 3187 Chapter-1	5032	
48	File 201 Chapter-1	5032	
49	File 201 Chapter-1	5032	
50	File 116 Stu-Retain	5005	
61	File 112 Stu Course	5044	2046379683
62	File 100 Stu Masters	5043	
63	File 112 Stu Course	5042	
64	File 100 Stu Masters	5041	
65	File 112 Stu Course	5040	
66	File 100 Stu Masters	5039	
67	File 112 Stu Course	5038	
68	File 100 Stu Masters	5037	
69	File 112 Stu Course	5036	
70	File 100 Stu Masters	5035	
71	File 112 Stu Course	5045	
72	File 100 Stu Masters	5046	
73	File 100 Stu Masters	5047	
74	File 112 Stu Course	5048	
75	File 100 Stu Masters	5049	
76	File 112 Stu Course	5050	
77	File 100 Stu Masters	5051	
78	File 112 Stu Course	5053	
79	File 112 Stu Course	5052	
80	File 100 Stu Masters	5063	
81	File 112 Stu Course	5054	

Appendix B

File size

This summarizes the size of the database, in gigabytes[1].

	EBCDIC Format	*ADABAS Format*
Student File	4.5 GB	1.5 GB
Employee File	2.5 GB	1.65 GB

[1] A gigabyte is roughly one million characters