Data Cleaning involving duplicate IDs and duplicate records

Lecture 8
July 7, 2003

Kate Cowles
374 SH
kcowles@stat.uiowa.edu

Duplicates

- may need to check for either duplicate ID codes or duplicate observations
- duplicate observations should just be eliminated
- duplicate IDs with different data values need to be resolved

Patients file: duplicate records added

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Gender</th>
<th>Date</th>
<th>Age</th>
<th>Zip Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>001M</td>
<td>11/11/1998</td>
<td>88140</td>
<td>80</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>002F</td>
<td>11/13/1998</td>
<td>84120</td>
<td>78</td>
<td>X0</td>
<td></td>
</tr>
<tr>
<td>003M</td>
<td>10/21/1998</td>
<td>68190</td>
<td>100</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>004F</td>
<td>01/01/1999</td>
<td>10120</td>
<td>5A</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>005F</td>
<td>07/07/1998</td>
<td>82148</td>
<td>84</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>007M</td>
<td>32/32/1998</td>
<td>88148102</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>011M</td>
<td>11/11/1998</td>
<td>90190</td>
<td>100</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>006F</td>
<td>07/08/1998</td>
<td>86240180</td>
<td>41</td>
<td></td>
<td></td>
</tr>
<tr>
<td>010F</td>
<td>10/10/1998</td>
<td>60120</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>011M</td>
<td>10/23/1998</td>
<td>68300</td>
<td>20</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>012M</td>
<td>12/98</td>
<td>60122</td>
<td>74</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>013F</td>
<td>23/23/1998</td>
<td>74108</td>
<td>64</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>014F</td>
<td>02/02/1999</td>
<td>22130</td>
<td>90</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>002F</td>
<td>11/13/1998</td>
<td>84120</td>
<td>78</td>
<td>X0</td>
<td></td>
</tr>
<tr>
<td>003M</td>
<td>12/12/1999</td>
<td>88112</td>
<td>74</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>015F</td>
<td>07/04/1999</td>
<td>82148</td>
<td>88</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>017F</td>
<td>05/05/1999</td>
<td>82148</td>
<td>84</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>019M</td>
<td>07/07/1999</td>
<td>58118</td>
<td>70</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>123M</td>
<td>12/12/1999</td>
<td>80</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>321F</td>
<td>000400200</td>
<td>51</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>020F</td>
<td>9/9/99</td>
<td>10</td>
<td>20</td>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td>022M</td>
<td>10/10/1999</td>
<td>48114</td>
<td>82</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>023F</td>
<td>12/12/1998</td>
<td>22</td>
<td>34</td>
<td>78</td>
<td>0</td>
</tr>
<tr>
<td>024F</td>
<td>09/09/1998</td>
<td>120</td>
<td>80</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>025F</td>
<td>10/10/1999</td>
<td>74102</td>
<td>68</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>027F</td>
<td>12/12/1999</td>
<td>166106</td>
<td>70</td>
<td></td>
<td></td>
</tr>
<tr>
<td>028F</td>
<td>28/28/1999</td>
<td>66150</td>
<td>90</td>
<td>30</td>
<td></td>
</tr>
</tbody>
</table>
Eliminating duplicates using PROC SORT

- **NODUPKEY** option on PROC SORT eliminates multiple observations where the BY variables have the same value.
- **OUT=** option is used to create a new data set, leaving the original data set unchanged.
- When **NODUPKEY** removes multiple observations, the only indication is in the NOTE in the SAS log.
- Looking for duplicate IDs with **NODUPKEY** in PROC SORT is useful only if the SAS log shows that NO duplicates were removed.
  - Otherwise, you need to see which IDs had duplicate data and the nature of the data.
- If **NODUPKEY** is used with more than one BY variable, only those observations with identical values on all the BY variables will be deleted.

Program 5-1 Demonstrating the **NODUPKEY** option of PROC SORT

```sas
options linesize = 72 ;
libname CLEAN '/space/kcowles/172/clean/' ;

PROC SORT DATA=CLEAN.PATIENTS OUT=SINGLE NODUPKEY;
   BY PATNO;
RUN;
```

```
PROC PRINT DATA=SINGLE;
   TITLE "Data Set SINGLE - Duplicated ID's Removed from PATIENTS";
   ID PATNO;
RUN;
```

Program 5-2 Demonstrating the **NODUP** option of PROC SORT

```sas
options linesize = 72 ;
libname CLEAN '/space/kcowles/172/clean/' ;

PROC SORT DATA=CLEAN.PATIENTS OUT=SINGLE NODUP;
   BY _ALL_;
RUN;
```

```
PROC PRINT DATA=SINGLE;
   TITLE 'SINGLE from NODUP' ;
RUN;
```

The **NODUP** option of PROC SORT

- **NODUP** also deletes duplicates, but only for two observations where ALL the variables have identical values.
- You have to sort by all of the variables to make this work correctly.
- Note use of \_ALL\_ as the by variable in the example for this purpose.

Program 5-3 Demonstrating the **NODUP** option of PROC SORT

```sas
options linesize = 72 ;
libname CLEAN '/space/kcowles/172/clean/' ;

PROC SORT DATA=CLEAN.PATIENTS OUT=SINGLE NODUP;
   BY _ALL_;
RUN;
```

```
PROC PRINT DATA=SINGLE;
   TITLE 'SINGLE from NODUP';
RUN;
```
Data step method for identifying and listing duplicate IDs

/***************************************************************************
Program 5-4 Identifying duplicate ID's
****************************************************************************/
PROC SORT DATA=CLEAN.PATIENTS OUT=TMP;
    BY PATNO;
RUN;
DATA DUP;
    SET TMP;
    IF FIRST.PATNO AND LAST.PATNO THEN DELETE;
RUN;
PROC PRINT DATA=DUP;
    TITLE "Listing of Duplicates from Data Set CLEAN.PATIENTS";
    ID PATNO;
RUN;

Listing of Duplicates from Data Set CLEAN.PATIENTS 4
22:25 Sunday, July 6, 2003

PATNO GENDER VISIT HR SBP DBP DX AE
002 F 11/13/1998 84 120 78 X 0
002 F 11/13/1998 84 120 78 X 0
003 X 10/21/1998 68 190 100 3 1
003 M 11/12/1999 58 112 74 0
006 06/15/1999 72 102 68 6 1
006 F 07/07/1999 82 148 84 1 0

A new dataset for illustration purposes

/***************************************************************************
Program 5-5 Creating the SAS data set, PATIENT2 (a data set containing multiple visits for each patient)
****************************************************************************/
DATA CLEAN.PATIENT2;
    INFILE "/space/kcowles/172/clean/PATIENT2.TXT" PAD;
    INPUT @1 PATNO $3. @4 VISIT MMDDYY10. @14 HR 3. @17 SBP 3. @20 DBP 3. ;
    FORMAT VISIT MMDDYY10.;
    RUN;
PROC PRINT DATA=CLEAN.PATIENT2 ;
    TITLE 'PATIENT2 DATA SET' ;
    RUN ;

Listing of Duplicates from Data Set CLEAN.PATIENTS 4
22:25 Sunday, July 6, 2003

PATNO GENDER VISIT HR SBP DBP DX AE
002 F 11/13/1998 84 120 78 X 0
002 F 11/13/1998 84 120 78 X 0
003 X 10/21/1998 68 190 100 3 1
003 M 11/12/1999 58 112 74 0
006 06/15/1999 72 102 68 6 1
006 F 07/07/1999 82 148 84 1 0
Detecting duplicates having same visit date for same patient

Program 5-6 Identifying patient ID’s with duplicate visit dates

PROC SORT DATA=CLEAN.PATIENTS OUT=TMP;
   BY PATNO VISIT;
RUN;

DATA DUP;
   SET TMP;
   BY PATNO VISIT;
   IF FIRST.VISIT AND LAST.VISIT THEN DELETE;
RUN;

PROC PRINT DATA=DUP;
   TITLE "Listing of Duplicates from Data Set CLEAN.PATIENTS";
   ID PATNO;
RUN;

Listing of Duplicates from Data Set CLEAN.PATIENTS 6
08:41 Monday, July 7, 2003

PATNO VISIT HR SBP DBP
005 04/14/1998 72 118 74
005 04/14/1998 74 120 80

Using PROC FREQ to detect duplicate IDs

• can use PROC FREQ to count the number of observations for each value of patient ID variable

• use patient ID variable and OUT= option of TABLES statement to create a SAS data set containing the value of the PATNO variable and the frequency count

  – PROC FREQ uses the variable name COUNT to hold the frequency information

• after you have this information, you can use it to select the original duplicate observations from your data set

   /********************************************************************************
   Program 5-7 Using PROC FREQ and an output data set to identify duplicate
   ********************************************************************************/
   PROC FREQ DATA=CLEAN.PATIENTS NOPRINT;
      TABLES PATNO / OUT=DUP_NO(KEEP=PATNO COUNT
      WHERE=(COUNT GT 1));
   RUN;
   PROC SORT DATA=CLEAN.PATIENTS OUT=TMP;
      BY PATNO;
   RUN;
   PROC SORT DATA=DUP_NO;
      BY PATNO;
   RUN;
   DATA DUP;
      MERGE TMP DUP_NO(IN=YES_DUP DROP=COUNT);
         BY PATNO;
      IF YES_DUP;
   RUN;
   PROC PRINT DATA=DUP;
      TITLE "Listing of Data Set DUP";
   RUN;
More efficient program to accomplish similar task

/***************************************************************************
Program 5-8 Producing a list of duplicate patient numbers, using PROC FREQ
***************************************************************************/
PROC FREQ DATA=CLEAN.PATIENTS NOPRINT;
  TABLES PATNO / OUT=DUP_NO(KEEP=PATNO COUNT
WHERE=(COUNT GT 1));
RUN;

DATA_NULL_;  
TITLE "Patients with Duplicate Observations";
FILE PRINT;
SET DUP_NO;
  PUT "Patient number " PATNO "has " COUNT "observation(s).";
RUN;

  Patients with Duplicate Observations
  08:41 Monday, July 7, 2003
Patient number 002 has 2 observation(s).
Patient number 003 has 2 observation(s).
Patient number 006 has 2 observation(s).

Identifying subjects with ”n” observations each using data step

- sometimes we know how many observations there should be for each subject and we need to verify that the correct number are there in data set
- example program lists all patient IDs that do not have exactly two observations each

/***************************************************************************
Program 5-11 Using a Data Step to list all ID’s for subjects who do not have exactly two observations
***************************************************************************/
PROC SORT DATA=CLEAN.PATIENT2(KEEP=PATNO) OUT=TMP;
  BY PATNO;
RUN;

DATA_NULL_;  
TITLE "Patient ID’s for Patients with Other than Two Observations";
FILE PRINT;
SET TMP;
  BY PATNO;
  IF FIRST.PATNO THEN N = 1;
  ELSE N + 1;
  IF LAST.PATNO AND N NE 2 THEN PUT "Patient number " PATNO "has " N "observation(s).";
RUN;

  Patient ID’s for Patients with Other than Two Observations
  08:41 Monday, July 7, 2003
Patient number 002 has 3 observation(s).
Patient number 003 has 1 observation(s).
Patient number 006 has 1 observation(s).
Accomplishing same task using PROC FREQ

- usually easier to have a SAS procedure do the work than to code up a fancy DATA step

/***************************************************************************
 Program 5-12 Using PROC FREQ to list all ID’s for subjects who do not have exactly two observations
***************************************************************************/

PROC FREQ DATA=CLEAN.PATIENT2 NOPRINT;
  TABLES PATNO / OUT=DUP_NO(KEEP=PATNO COUNT WHERE=(COUNT NE 2));
RUN;

DATA _NULL_
TITLE "Patient ID’s for Patients with Other than Two Observations";
FILE PRINT;
SET DUP_NO;
PUT "Patient number " PATNO " has " COUNT " observation(s).";
RUN;

Patient ID’s for Patients with Other than Two Observations

Patient number 002 has 3 observation(s).
Patient number 003 has 1 observation(s).
Patient number 006 has 1 observation(s).