Paper 102-2012

Mining and Merging DATAMONITOR and WRDS Databases with SAS®

N. Yaraghi, R. Kishore, SUNY at Buffalo, Buffalo, NY, United States

R. Chen, Ball State University, Muncie, IN, United States

ABSTRACT

In this paper, we show how a DATAMONITOR database can be grouped to public and private companies based on ticker-symbol databases of NASDAQ, AMEX, and NYSE. Moreover, we show EVENTUS software output is cleaned and merged with the public companies databases.

INTRODUCTION

DATAMONITOR data base is a famous and reliable source of data in management field. For example, it provides precise and reliable information about different aspects of IT contracts in US companies over a specific period of time. This data base can be of huge interest to Information Systems researchers for investigating different research interests such as the effect of IT contracts on stock prices, etc.

However, DATAMONITOR has its own shortcomings and sometimes should be merged with other data bases to construct a complete data base. In previous example, data from stock market prices should be added to the data base to enable researchers to conduct an event study. It is not easy; First of all, private companies which their stock data is not available should be eliminated. Since there is no identifier about the status of the company, this should be done by using other data bases and some SAS tricks. Second, the available stock market data should be merged with the existing companies correctly. EVENTUS is the software which is usually used for event study research. When EVENTUS is accessed over a server, its output needs special cleansing before merging with any data set. We show how it can be cleaned and properly merged with public companies subset of DATAMONITOR database.

SUBSETTING DATAMONITOR

Financial data, such as stock market prices is only publicly available for public companies. However, there is no identifier in DATAMONITOR to help us selecting public companies. Since DATAMONITOR databases are generally very big, there is an absolute need in automating the process of extracting data of public companies from this data set.

Comparing this data base with easily available NYSE, NASDAQ and AMEX database of public companies is a smart way to do this task, however, since the only common value between these two databases is the company names, which is of string type, and may not always be exactly the same, this becomes a little tricky. If we simply want to choose the companies with exact same names, we will lose a lot of valid observations. An approach to solving this problem is to do iterative merging. Consider the following examples.

There are 20 observations in DATAMONITOR data base; some of the companies in this data set are private and not listed in the NYSE data base. If we simply merge these two data sets, we will miss some observations in which the name of the companies are slightly different due to differences in data entry methods. Companies such as “Hartford Financial Services Gr” will be incorrectly removed from the data base.

By a set of iterative merges, we will start with the companies which have the exact same name in both databases, set them aside for later and continue with merging the rest of data base with NYSE database while we have truncated the name of the companies by 1 character. By Repeating this code for several times, and finally combining all the subsets that you had set aside in previous merges you can make sure that you have extracted all the public companies in your dataset. The following code is shown for two iterations.

	Company	Contract Date
1	Huntsman Corporation	02112001
2	IAC/InterActiveCorp	04022003
3	Johnson & Johnson	04062005
4	Johnson & Johnson	11232005
5	Johnson & Johnson	11282005
6	KB Home	04052000
7	KeyCorp	09082007
8	Morgan Stanley	04262005
9	7-Eleven	03152001
10	Zurich Financial Services	11282005
11	Harman International Industrie	08072008
12	Hartford Financial Services Gr	03042001
13	Hartford Financial Services Gr	03072010
14	Hartford Financial Services Gr	04092009
15	Partner Communications Company	03052001
16	Partner Communications Company	09112008
17	Yorkshire Water	11122009
18	Partner Communications Company	12112001
19	MyTravel	04182009
20	Yorkshire Water	04222006

Table 1: DATAMONITOR data base

	Company Name	Ticker
1	Huntsman Corporation	HUN
2	IAC/InterActiveCorp	IACI
3	Johnson & Johnson	JNJ
4	KB Home	KBH
5	KeyCorp	KEY
6	Morgan Stanley	MS
7	Harman International Industries	HAR
8	Hartford Financial Services Group	HIG
9	Partner Communications Company	PTNR

Table 2: NYSE ticker symbols data base

```
data nyse;
length company $20;
set nyse;
run;

data DATAMONITOR;
length company $20;
set DATAMONITOR;
companyfullname=company;
run;

data similar1 others1;
merge DATAMONITOR (in=inIW)
       nyse(in=inticker);
       by company;
if inIW and inticker then output similar1;
else if inIW and not inticker then output others1;
run;

data nyse;
length company $19;
set nyse;
run;

data others;
length company $19;
set others;
run;

data similar2 others2;
merge others1(in=inIW)
       nyse(in=inticker);
       by company;
if inIW and inticker then output similar2;
```

```
else if inIW and not inticker then output others2;
run;

data allsimilar;
set similar1 similar2;
run;
```

Note that we cannot reduce the number of characters at first, since there are so many different companies which the first few characters of their names are similar. If we do the merging based on reduced names at first, we are incorrectly considering different companies as the same. For example if we have two companies of "`Yorkshire Electricity`" and "`Yorkshire Water`" and do the merging only based on the first 9 characters, we will end up with a wrong data set. The trick in this code is that it first starts with 20 characters, checks if there is any company with the name of "`Yorkshire Electricity`" is existing in the NYSE data set, if there were, it puts it in `similar` data base and excludes it from the second round of comparisons which is with the first 19 characters. This process continues and thus the number of companies which are searched for is continuously reducing. At the end of the process, all of the data bases which contain unique companies in NYSE data bases are combined with each other.

The results are shown in data set "all similar";

	Company Name	Ticker	Contract Date
1	Huntsman Corporation	HUN	02112001
2	IAC/InterActiveCorp	IACI	04022003
3	Johnson & Johnson	JNJ	04062005
4	Johnson & Johnson	JNJ	11232005
5	Johnson & Johnson	JNJ	11282005
6	KB Home	KBH	04052000
7	KeyCorp	KEY	09082007
7	Morgan Stanley	MS	04262005
8	Harman International Industries	HAR	08072008
9	Hartford Financial Services Group	HIG	03042001
10	Hartford Financial Services Group	HIG	03072010
11	Hartford Financial Services Group	HIG	04092009
12	Partner Communications Company	PTNR	03052001
13	Partner Communications Company	PTNR	09112008

Table 3: "ALL SIMILAR" data base

ADDING CUSIP CODES

CUSIP codes are essential for downloading correct data from WRDS data base. You can download a complete database of CUSIP directory which is available online and then merge the CUSIP codes with the company's existing in your data set.

```
data CUSIPavailable others;
merge CUSIP(in=inIW)
      allsimilar (in=inticker);
      by ticker;
if inIW and inticker then output CUSIPavailable;
else if inIW and not inticker then output others;
run;
```

The following table shows the results

	Company Name	Ticker	Contract Date	CUSIP
1	Huntsman Corporation	HUN	02112001	447011107
2	IAC/InterActiveCorp	IACI	04022003	44919P508
3	Johnson & Johnson	JNJ	04062005	478160104
4	Johnson & Johnson	JNJ	11232005	478160104
5	Johnson & Johnson	JNJ	11282005	478160104

6	KB Home	KBH	04052000	48666K109
7	KeyCorp	KEY	09082007	493267108
8	Morgan Stanley	MS	04262005	617446448
9	Harman International Industries	HAR	08072008	413086109
10	Hartford Financial Services Group	HIG	03042001	416515104
11	Hartford Financial Services Group	HIG	03072010	416515104
12	Hartford Financial Services Group	HIG	04092009	416515104
13	Partner Communications Company	PTNR	03052001	70211M109
14	Partner Communications Company	PTNR	09112008	70211M109

Table 4: "CUSIPavailable" data base

COLLECTING FINANCIAL DATA THROUGH EVENTUS

With the CUSIP and contract dates at hand, we can collect Cumulative Abnormal Returns (CAR) on stock prices for the specific list of companies in our data set around a specified event window and based on a specified estimation window. The following is the scrip code which collects CAR values in (-1, 1) and (0,2) event window based on a (-120,-1) estimation window in S&P500 benchmark. For further details please refer to EVENTUS manual.

```
FILENAME REQUEST CUSIPavailable.txt';
EVENTUS;
WINDOWS (-1,1) (0,2);
REQUEST CUSIPERM SP500 EST=-1 ESTLEN=120 AUTODATE;
EVTSTUDY MAR OVERLAP FILEWIN='SP500-120-1.data';
```

CLEANING THE EVENTUS OUTPUT

The former script is used to collect data from WRDS database over a server. After running the script through any Telnet/SSH client such as puTTY, three files will be created on the server, namely data file (.DATA extension), log file (.LOG extension) and list file (.LST extension). The log file simply shows how EVENTUS has processed the data. The data file, contains CAR values for each CUSIP code. But there is no date for observations. The dates are included in the list file. To be able to use these outputs, one should first check the list files for the observations which there were no CAR value reported. These observations are listed in the beginning of the list file. The output 1 is a screenshot of the list file.

The CUSIP codes, with "`Unavailable; missing PERMNO`" are the ones which EVENTUS has not returned any CAR values.
The data file, has much more observation than the input data because of these different event windows or estimation windows which has been requested in the script. We should first subset the data file into separate data sets which represent unique properties. The following code subsets the data set into two data sets in which each of them contains CAR values for a specific event window.

```
Data SP_one_one SP_zero_two;
set cusip_data;
if window="(-1,+1)" then output SP_one_one;
else output SP_zero_two;
run;
```

One should delete the observations with `missing PERMNO` in list file and then merge the remaining observations based on common CUSIP with the reported observations in the subset data files. Although data file, does not report the data for which the CAR value is calculated for, but it reports the CAR values in ascending order of dates, which is the same order in list file.
Luckily merging the list and data files based on CUSP code, provides a complete and correct data set of companies with CUSIP, event dates and relative CAR values.

```
*
    *     17133Q50    Unavailable; missing PERMNO          11/19/2003
*
    *     92839U20    Unavailable; missing PERMNO          12/02/2003
*
    *     30033R30    Unavailable; missing PERMNO          03/22/2004
*
    *     17133Q50    Unavailable; missing PERMNO          08/04/2004
*
    *     17133Q50    Unavailable; missing PERMNO          08/11/2004
*
    *     10553M10    Unavailable; missing PERMNO          11/11/2004
*
    *     10553M10    Unavailable; missing PERMNO          08/21/2006
*
    *     10553M10    Unavailable; missing PERMNO          01/17/2008
*
    *     10553M10    Unavailable; missing PERMNO          05/07/2008
*
    *     17133Q50    Unavailable; missing PERMNO          05/30/2008
*
    *     17133Q50    Unavailable; missing PERMNO          10/09/2008
*
    *     17133Q50    Unavailable; missing PERMNO          12/10/2008
*
10107     59491810    MICROSOFT CORP                       10/23/2001
201
10107     59491810    MICROSOFT CORP                       08/14/2002
201
10107     59491810    MICROSOFT CORP                       04/30/2003
201
10107     59491810    MICROSOFT CORP                       05/23/2005
201
```

Output1:EVENTUS output

RECOMMENDED READINGS

EVENTUS manual : http://www.eventstudy.com/Eventus-and-SAS.htm

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Niam Yaraghi
Enterprise: SUNY at Buffalo
Address: 304 Jacobs Management Center,
City, State ZIP: Buffalo, NY 14260-4000
Work Phone: (716)645-5256
E-mail: niamyara@buffalo.edu