Recently Written

Twitter

Categories

- The Julia Group
- WordPress

Archives

- August 2020
- April 2020
- February 2020
- January 2020
- December 2019
- October 2019
- September 2019
- July 2019
- May 2019
- April 2019
- January 2019
- December 2018
- November 2018
- October 2018
- August 2018
- July 2018
- April 2018
- February 2018
- January 2018
- December 2017
- November 2017
- October 2017
- September 2017
- August 2017
- July 2017
- June 2017
- May 2017
- April 2017
- March 2017
- February 2017
- January 2017
- December 2016
- November 2016
- October 2016
- September 2016

- August 2016
- July 2016
- June 2016
- May 2016
- April 2016
- March 2016
- February 2016
- January 2016
- December 2015
- November 2015
- October 2015
- September 2015
- August 2015
- July 2015
- June 2015
- May 2015
- April 2015
- March 2015
- February 2015
- January 2015
- December 2014
- November 2014
- October 2014
- September 2014
- August 2014
- July 2014
- June 2014
- May 2014
- April 2014
- March 2014
- February 2014
- January 2014
- December 2013
- November 2013
- October 2013
- September 2013
- August 2013
- July 2013
- June 2013
- May 2013
- April 2013

- March 2013
- February 2013
- January 2013
- December 2012
- November 2012
- October 2012
- September 2012
- August 2012
- July 2012
- June 2012
- May 2012
- April 2012
- March 2012
- February 2012
- January 2012
- December 2011
- November 2011
- October 2011
- September 2011
- August 2011
- July 2011
- June 2011
- May 2011
- April 2011
- March 2011
- February 2011
- January 2011
- December 2010
- November 2010
- October 2010
- September 2010
- August 2010
- July 2010
- June 2010
- May 2010
- April 2010
- March 2010
- February 2010
- January 2010
- December 2009
- November 2009

- October 2009
- September 2009
- August 2009
- July 2009
- June 2009
- May 2009
- April 2009
- March 2009
- February 2009
- January 2009
- December 2008
- October 2008
- September 2008
- August 2008
- July 2008
- June 2008
- May 2008
- April 2008
- March 2008
- February 2008
- January 2008

Admin

- Log in
- Wordpress
- XHTML

Search

search this site...

Mar

21

Random SAS tips: Mixed data types

March 21, 2014 | 1 Comment

This week I had one of those pain-in-the-ass problems. I had a test with 24 items but they were of mixed types. That is, for some the answer was multiple choice and for others it was numbers.

The data was received as an Excel file.

Now, I could have opened it with SAS Enterprise Guide and specified data types for each variable, but the problem is, I am going to get this particular data set over and over, so I want code I can write once and run every time.

As if that wasn't bad enough, the variables all had names like:

which_choice_is_the_same_as_the_

I wanted to rename these all to something sane like q1, q2 etc.

The first step was an option I don't think I've ever used before, oddly enough.

proc contents data= annoying varnum ;

Normally, SAS gives you the variables in a data set in alphabetic order when you do a PROC CONTENTS. The varnum option lists the variables in the order they appear on the data set. This was immensely helpful because it spared me going through the data trying to figure which was the first question, which was the second, and so on.

I just copied the variables in order after a RENAME statement and tacked on an =q1, q2, etc. like so

Data better ;

set annoying ;

rename

which_choice_is_the_same_as_the_ = q1
what_is_five_time_six__ = q2

etc.

proc contents data= better ;

I could have combined this with the previous step , but the fact is that unless the data set is really gigantic, the time that needs to be preserved is not computer processing time but my time, and this way was quicker because I didn't have to write out those ridiculous variable names and worry about the program failing because I used _ in the name instead of __ .

SAS does have a function to detect variable type, but that wouldn't really have helped me because I still need to write all of these variables into a single array of item1 – item24 for later use, and you cannot have mixed type arrays. So, I did this

```
data mo_better ;
set better ;
array qs{*} q2 – q6 q10 q12 q14 -q16 q19- q21 ;
array itemN {*} $12 item2 – item6 item10 item12 item14 – item16 item19 – item21 ;
array qsA {*} $12 q1 q7-q9 q11 q13 q17 q18 q22- q24 ;
array itemA {*} $12 item1 item7 – item9 item11 item13 item17 item18 item22- item24 ;
do i = 1 to dim(qs) ;
 itemN{i} = put(qs{i},12.) ;
end ;
do j = 1 to dim(qsa) ;
 itemA{j} = qsa{j} ;
end ;
drop i j q1 – q24 ;
```

I have 4 arrays. The first consists of the numeric variable type questions. I couldn't use _numeric_ to create an array of all numeric variables because there were others in the data set that were NOT test questions but were numeric and I did not want them in my array. I had to actually list each variable individually or in a range like q14-q16.

The next array is the one I am going to recode the variables into as character variables. Notice that character arrays need a $ and a length. The next two arrays are the character variables and the variables I'm going to copy them into. I could have just renamed the character variables in a RENAME statement and then changed the length in an ATTRIB statement but it would have taken more typing.

The DIM function is the dimension of the array, so it is going to loop through from 1 to however many variables in the array because I didn't feel like counting them.

The PUT function is going to put this numeric variable into a new character variable with the specified length. It changes the variable to character.

The next loop just puts all the character variables into other character variables with the names item1, item7, etc. Now I have variables that are all the same type and length, named item1 – item24 and I can do things with them like compare each student's response to each variable to the answer key, score it right or wrong and sum up the scored items, like this (1ANSWER is the first username)

```
Data in.pre_scored ;
set mo_better ;
by username ;
Array scored {24} sc1 – sc24 ;
Array items {24} $12 item1- item24 ;
Array ans{24} $12 ans1 – ans24 ;
if _n_ = 1 then do i = 1 to 24 ;
ans{i} = items{i} ;
end ;
else do i = 1 to 24 ;
if ans{i} = items{i} then scored{i} = 1 ;
else scored{i} = 0 ;
end ;
Retain ans1 – ans24 ;
total = sum(of sc1-sc24) ;
```

Since this is part of a two-year grant and I am going to receive these same test data sets many times, I am now finished with reading in and scoring the data for the next two years. After this, I just need to import the excel file and click run. I am happy.

Also curious, because I noted that this year's pre-test scores are 1.5 standard deviations higher than the previous year. I suspect this is because we have many more fifth-graders in this sample. So … with the scoring done automatically, I can now go on to interesting stuff.

If you want to check out the game these results came from, you can read about it here

Comments

Name (required)

Email (required)

Website

Speak your mind

Submit Comment

1 Comment so far

1. Random SAS tips on a snowy day : AnnMaria's Blog on March 31, 2014 1:06 pm

   […] Random SAS tips: Mixed data types […]

Blogroll

- Andrew Gelman's statistics blog - is far more interesting than the name
- Biological research made interesting
- Interesting economics blog
- Love Stats Blog - How can you not love a market research blog with a name like that?
- Me, twitter - Thoughts on stats
- SAS Blog for the rest of us - Not as funny as some, but twice as smart. If this is for the rest of us, who are those other people?
- Simply Statistics, simply interesting
- Tech News that Doesn't Suck
- The Endeavor -John D Cook - Another statistics blog

WP Themes

- Activate
- Blindigo
- Blue Zinfandel
- Eye Candy
- Detour
- Frequency
- Radiance

- Vertigo
- Vertigo Squared
- Wonderland