

VIEWS

International SAS Programmer Community



News, Hints, Tips and Information for SAS® Users

Issue 54, 2nd/3rd Quarter 2011

I know I said that I would try to hit all my deadlines in 2011, but, to be bluntly honest with you, I've never been so busy as I've been since the beginning of May! This newsletter will now be a combined May/August issue, mostly because my wife told me I should not try to do more than I was physically capable of completing. So, for the first time in the history of *VIEWS News*, I have opted to miss an issue, for which I feel a combination of guilt and relief. I will, however, still be bringing you a bigger than usual newsletter issue filled with SAS news, hints and tips that I hope you will find interesting. In this issue of *VIEWS News* I have articles from our regular contributors: LeRoy Bessler, Murphy Choy, and myself, with articles on charting long trends, protecting SAS programs with passwords, the Metaphone Algorithm and Anscombe's Quartet (you will never trust a statistician again! ☺). There is another regular article from Amadeus Software, David Shannon is giving you the chance to try out the latest Amadeus Friday Quiz. Finally, my regular Formats, Options, and Functions section has some new features in the new release, SAS 9.3.

If you would like to contribute an article, to re-visit and improve an existing article, or just discuss the possibility of doing so, please feel free to send an email to me at newsletter@views-uk.org. A list of subject suggestions for your articles can be found on the *VIEWS* web site, and anyone is very welcome to add to that list by sending emails to newsletter@views-uk.org with your own questions.

Philip R Holland (Newsletter Editor)

Membership of *VIEWS* is free and you can register for notification of all future *VIEWS* events by emailing us at membership@views-uk.org, making certain you include your name and postal address. If you would like to receive email notification whenever a new issue of the free quarterly *VIEWS News* is published on the web site, please remember to also include your email address, and please inform us when you change it.

Did You Know?

If you have a useful hint or tip, please send it to the Editor, and share it with the *VIEWS* membership.

Anscombe's Quartet

[SAS 9.2] I discovered these data sets during a presentation by Michael Blastland at the SAS Professionals Convention at SAS UK in Marlow in July this year (see the PhUSE News section below for my report on the event itself). Anscombe's quartet comprises four data sets that have identical simple statistical properties. Each dataset consists of eleven (x,y) points.

They were constructed in 1973 by the statistician F J Anscombe to demonstrate both the importance of graphing data before analysing it and the effect of outliers on statistical properties. The data sets can be read into SAS data sets using the following SAS code:

```
DATA anscombe;
  INFILE DATALINES DLM=' ' DSD TRUNCOVER;
  LENGTH dsset $3;
  INPUT x y @; dsset='I'; OUTPUT;
  INPUT x y @; dsset='II'; OUTPUT;
  INPUT x y @; dsset='III'; OUTPUT;
  INPUT x y; dsset='IV'; OUTPUT;
  DATALINES;
10.0,8.04,10.0,9.14,10.0,7.46,8.0,6.58
8.0,6.95,8.0,8.14,8.0,6.77,8.0,5.76
13.0,7.58,13.0,8.74,13.0,12.74,8.0,7.71
9.0,8.81,9.0,8.77,9.0,7.11,8.0,8.84
11.0,8.33,11.0,9.26,11.0,7.81,8.0,8.47
14.0,9.96,14.0,8.10,14.0,8.84,8.0,7.04
6.0,7.24,6.0,6.13,6.0,6.08,8.0,5.25
4.0,4.26,4.0,3.10,4.0,5.39,19.0,12.50
12.0,10.84,12.0,9.13,12.0,8.15,8.0,5.56
7.0,4.82,7.0,7.26,7.0,6.42,8.0,7.91
5.0,5.68,5.0,4.74,5.0,5.73,8.0,6.89
RUN;

PROC SORT DATA=anscombe;
  BY dsset x;
RUN;
```

The data sets have essentially the same statistical analysis:

```
dsset=I
Number of Observations Read      11

      Sum of      Mean
Source  DF  Squares  Square  F Value  Pr > F
Model    1   27.51000  27.51000  17.99   0.0022
Error    9   13.76269   1.52919
Corrected
  Total  10   41.27269

Root MSE=1.23660  R-Square=0.6665
Dependent Mean=7.50091  Adj R-Sq=0.6295
Coeff Var=16.48605

      Parameter Standard t
Variable DF  Estimate  Error  Value  Pr > |t|
Intercept 1    3.00009    1.12475    2.67   0.0257
x          1    0.50009    0.11791    4.24   0.0022
```

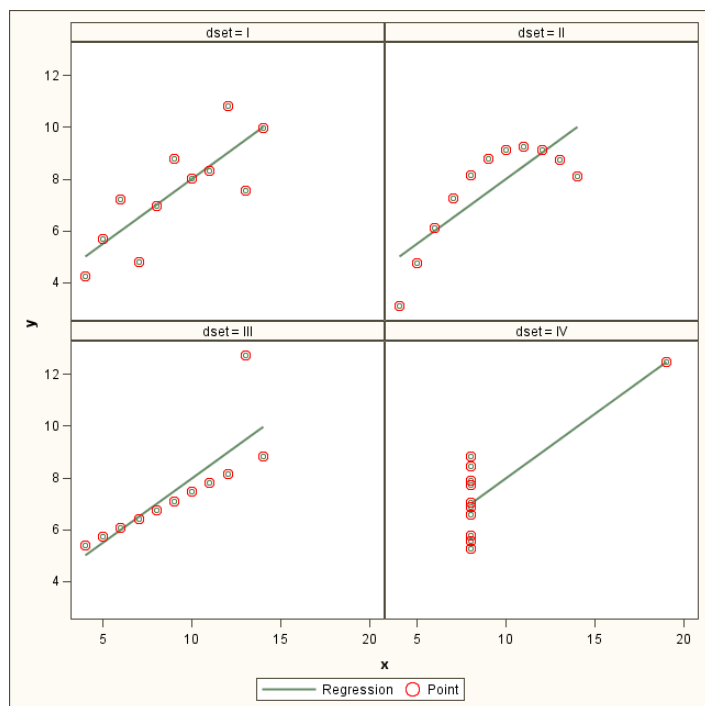
The statistical analysis above was generated using the following SAS code:

```
PROC REG DATA=anscombe RSQUARE;
  BY dsset;
  MODEL y=x;
RUN;
```

The following SGPANEL code generated the graphs, which plainly show that, although the statistical analysis of the 4 data

sets is almost identical, the graphs of the data sets are very different!

```
PROC SGPanel DATA=anscombe;
  PANELBY dset;
  REG X=x Y=y / LEGENDLABEL="Regression";
  SCATTER X=x Y=y / LEGENDLABEL="Point"
  MARKERATTRS=(COLOR=RED SIZE=10PT);
RUN;
```



Anscombe's quartet comprises four datasets that have identical simple statistical properties, yet appear very different when graphed. **Never accept numerical statistics alone, but request a graph too!**

Philip R Holland, Newsletter Editor

Data taken from the Wikipedia article about Anscombe's Quartet at en.wikipedia.org/wiki/Anscombe's_quartet.

How To Chart a Very Long Trend

A very long trend plot has many data points. If you want horizontal tick values at reasonable intervals, readability is difficult to achieve.

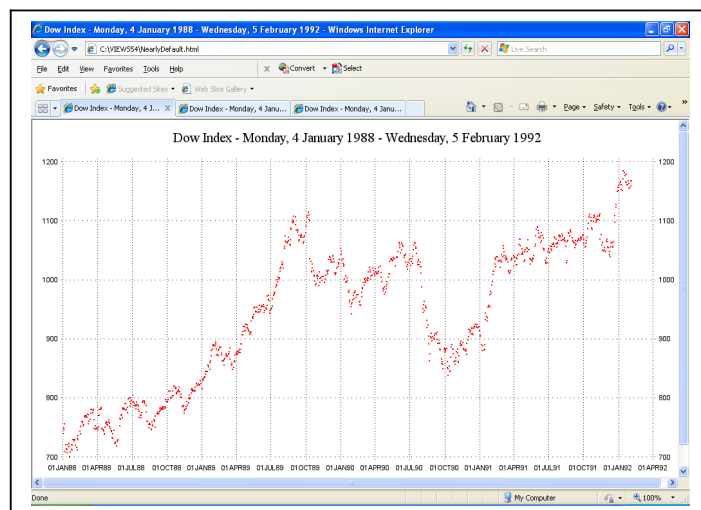
NOTE: The images here will be artificially hard to read due to shrinkage required for publication. Since I am using data set SASHELP.CITIDAY, the reader can run the same code as an exercise, display the results at monitor size, and proceed to experiment further with the code as desired.

Any trend chart is best delivered with HTML since the viewer can then access pop-up text (ALT text, in HTML parlance, or otherwise called hover text, flyover text, float-over text, mouse-over text, tool tip, and probably one or more terms that I have yet to encounter). A graph accelerates or facilitates decision or inference, but precise numbers are needed for a reliable decision or inference. Any trend with numerous points cannot be annotated without collisions between labels and possibly between labels and the line. To use a lookup table for a y-value, the viewer needs to know what is the precise x-value of interest. ALT text avoids the problem. The programmer can be as precise as

desired with the values, format the text as desired, and even include a special comment relevant to a data point.

But a graph viewer would still like to know approximately what the x-value is at any point along the trend line without having to constantly check the ALT text with the mouse.

SASHELP.CITIDAY covers 1034 market-trading days at the New York Stock Exchange, spanning 4+ years. Here is the nearly default result from SAS/GRAPH:



I call this nearly default because I made some font selections, turned on reference lines with

```
PLOT ... / AUTOHREF LHREF=33 AUTOVREF LVREF=33
```

and turned off unneeded axis paraphernalia with

```
AXIS1 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0;
```

I also adjusted the image size with

```
GOPTIONS XPIXELS=993 YPIXELS=528;
```

to maximize the use of the available live space in Internet Explorer 8 on Windows XP Pro on a laptop screen with resolution 1024x768, which was a very popular setting, before the days of wide screen laptops. Those wide screens are not designed for the dimensions of a presentation LED projection screen (which has width and height in the proportion 4:3), but for viewing movies.

The default result is horizontal axis tick mark values at the first day of every quarter, 01JAN88, 01APR88, . . . , 01APR92. The actual date range is 4 January 1988 to 5 February 1992.

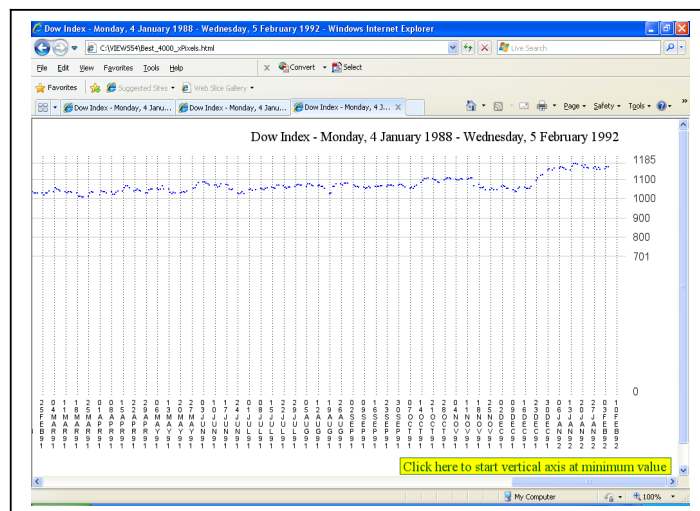
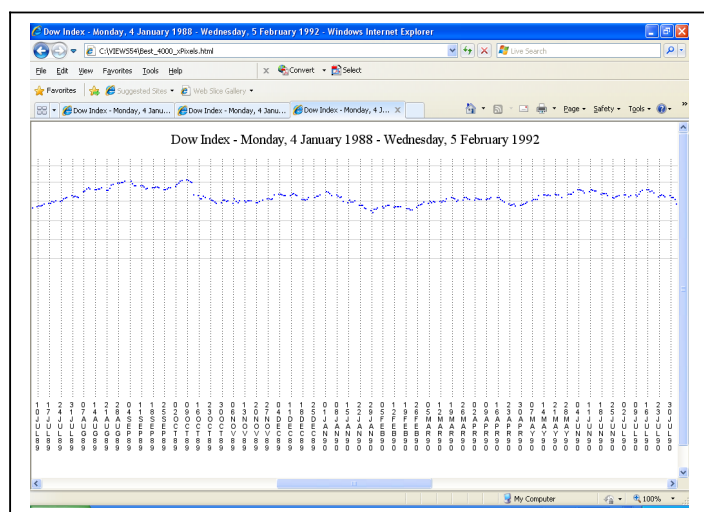
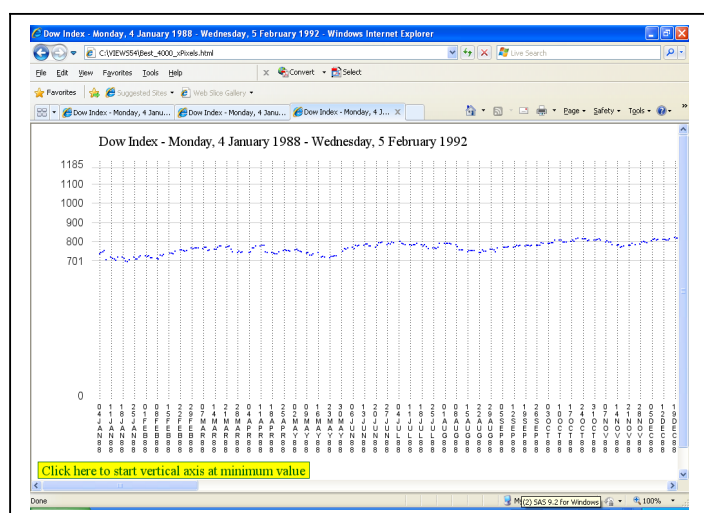
Though I usually object to horizontal scrolling, this seemed like a situation where widening the image could get more tick value detail without coding to try to force it. When I used GOPTIONS XPIXELS=4000, the only change in the result was horizontal axis tick mark values at the first day of every second month, 01JAN88, 01MAR88, . . . , 01MAR92. Increasing the width to 10000 pixels, 20000 pixels, and even 40000 pixels yielded that same result.

Though it is impractical to use 1034 horizontal tick mark values for this 4+ year trading-day time span, a tick-mark value for the starting Monday of every week is an adequate solution. I can tell you that, on a wide screen laptop with resolution set to 1280x800, and corresponding GOPTIONS XPIXELS=1242, it is easy to read weekly tick mark values (with HEIGHT=1.5 PCT)

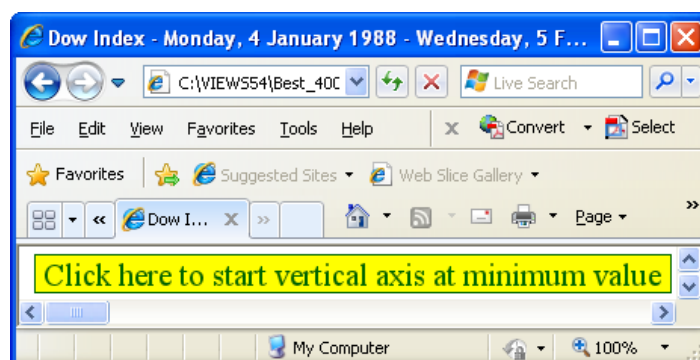
for a time period as long as 26 months, and significantly longer with a smaller font height.

Here I am going to assume a traditional monitor shape, and will use a width of 4000 pixels. A pair of plots is my recommended solution. Actually, just one of them is my preference, but I provide a button to allow the viewer to toggle between a plot that starts the vertical axis at zero and a plot that maximizes use of the available vertical space. In a business, organizational, or institutional performance-tracking context, NOT starting the vertical axis at zero artificially magnifies what might really be insignificant rises or declines, and can inspire needless elation or dismay about changes that are not really that bad. The significance of a change is the actual numeric AMOUNT of change (unless the change passes some threshold, or meets or fails some goal value), NOT the visual size of the change on a plot. Recall the maxim above: *“Graphs for quick, easy inference, but precise numbers for reliable inference”*.

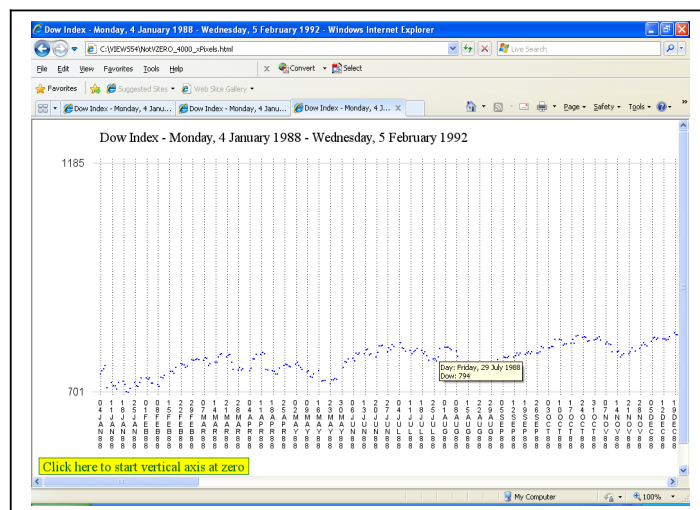
Below are two ways to chart a very long trend. For the first option, a screen print of the left end (start) of the trend, the centre of the trend, and the right end (stop) of the trend are shown. There is a “push button” link to the companion/alternative chart. The button is below the left and right axes. It could have been repeated in the middle of the chart, just as the title of the chart appears at the left, middle, and right areas of the chart. Below the screen images is a blow-up of the push button.



Squashing the browser window down to focus on a push button:



Below is just one view of the companion equally wide plot, using the vertical-axis space differently. I chose to use the minimum and maximum y-values as the only tick marks. The precise y-values are available in the ALT text, as demonstrated with the pop-up box displayed. One could choose to use ORDER=700 to 1200 by 100 instead.



Creating weekly tick marks on every Monday, or on any other specific day of the week, requires pre-analysis of the data dates. Using the ORDER parameter on the AXIS statement in this situation requires starting on a Monday and ending on a Monday, and using an increment (for “BY N”, where N is the tick-mark increment in SAS date days) of 7 days. If the first data date is not a Monday, then the axis must start on the last Monday before the start data date. If the last data date is not a Monday, then the axis

must end on the first Monday after the last data date. The start and end SAS dates for ORDER are determined in a DATA step and passed to an AXIS statement as symbolic (or macro) variables. Since I like to take control of the vertical axis, the minimum and maximum values of the y-variable are also determined with preprocessing and passed to an AXIS statement.

Here is the code used to create the pair of graphs:

```
DATA WORK.DowByDay;
  LENGTH HTMLvar $ 512;
  SET sashelp.citiday
    (KEEP=date snydjcm WHERE=(snydjcm NE .));
  snydjcm=ROUND(snydjcm,1);
  HTMLvar = "alt=" || "Day: " ||
    TRIM(LEFT(PUT(Date,weekdatx32.))) ||
    '0D'X || "Dow: " ||
    TRIM(LEFT(PUT(snydjcm,4.))) || " ";
RUN;

PROC MEANS DATA=work.DowByDay MIN MAX NOPRINT;
  VAR date;
  OUTPUT OUT=work.StartEndDates
    MIN=Day1 MAX=DayLast;
RUN;

DATA _NULL_;
  SET work.StartEndDates;
  CALL SYMPUT('DateRange',
    TRIM(LEFT(PUT(Day1,weekdatx32.))) ||
    ' - ' ||
    TRIM(LEFT(PUT(DayLast,weekdatx32.))));
  * Want to start on a Monday not later than
  the first day of data *;
  DayOfWeek1 = WEEKDAY(Day1);
  IF DayOfWeek1 GT 2
    THEN Day1 = Day1 - (DayOfWeek1 - 2);
  ELSE IF DayOfWeek1 EQ 1
    THEN Day1 = Day1 - 6;
  CALL SYMPUT('StartDateTick',Day1);
  * Want to end on a Monday not earlier than
  the last day of data *;
  DayOfWeekLast = WEEKDAY(DayLast);
  IF DayOfWeekLast GT 2
    THEN
      DayLast = DayLast + (9 - DayOfWeekLast);
    ELSE IF DayOfWeekLast EQ 1
      THEN DayLast = DayLast + 1;
  CALL SYMPUT('EndDateTick',DayLast);
RUN;

PROC MEANS DATA=work.DowByDay MIN MAX NOPRINT;
  VAR snydjcm;
  OUTPUT OUT=work.MinMaxY MIN=MinY MAX=MaxY;
RUN;

DATA _NULL_;
  SET MinMaxY;
  CALL SYMPUT('MinY',TRIM(LEFT(MinY)));
  CALL SYMPUT('MaxY',TRIM(LEFT(MaxY)));
  CALL SYMPUT('Yrange',MaxY - MinY);
RUN;

ODS NORESULTS; /* do not open in SAS session */
ODS LISTING CLOSE;

GOPTIONS RESET=all;
GOPTIONS FTEXT="Arial"; /* for axis values */
GOPTIONS XPixels=4000 YPixels=528;
```

```
SYMBOL1 COLOR=CX0000FF HEIGHT=0.5 FONT=ZAPF
  VALUE='E2'X;
SYMBOL2 COLOR=CX0000FF HEIGHT=0.5 FONT=ZAPF
  VALUE='E2'X;

TITLE1 FONT='Times New Roman' HEIGHT=4 PCT
  JUSTIFY=LEFT
  "
  Dow Index - &DateRange"
  JUSTIFY=CENTER
  "Dow Index - &DateRange"
  JUSTIFY=RIGHT
  "Dow Index - &DateRange";
/* NOTE: MOVE= could instead be used to achieve
  the shifts from the margins above. */

AXIS1 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0
  ORDER = &StartDateTick to &EndDateTick by 7
  VALUE=(HEIGHT=2PCT);
AXIS2 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0
  ORDER = 0 to 1200 by 100 VALUE=(
    HEIGHT=3PCT '0' ' ' ' ' ' ' ' ' ' '
    "&MinY" '800' '900' '1000' '1100' "&MaxY");
AXIS3 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0
  ORDER = 0 to 1200 by 100 VALUE=(JUSTIFY=LEFT
    HEIGHT=3PCT '0' ' ' ' ' ' ' ' '
    "&MinY" '800' '900' '1000' '1100' "&MaxY");

FOOTNOTE1 FONT='Times New Roman' HEIGHT=4 PCT
  COLOR=CX006600
  JUSTIFY=LEFT
  BOX=1 BSPACE=0 IN BCOLOR=CXFFFF00
  LINK='NotVZERO_4000_xPixels.html'
  ' Click here for vertical axis at min value '
  JUSTIFY=RIGHT
  BOX=1 BSPACE=0 IN BCOLOR=CXFFFF00
  LINK='NotVZERO_4000_xPixels.html'
  ' Click here for vertical axis at min value ';

PROC CATALOG CAT=work.gseg KILL; RUN; QUIT;

ODS HTML PATH="C:\VIEWS54" (URL=NONE)
  STYLE=Minimal GTITLE GFOOTNOTE
  BODY="Best_4000_xPixels.html"
  (TITLE="Dow Index - &DateRange");

PROC GPLOT DATA=work.DowByDay;
  PLOT snydjcm*date / HTML=HTMLvar
    NAME='Best4000' DESCRIPTION=' '
    HAXIS=AXIS1 VAXIS=AXIS2
    VREF = (&MinY 800 900 1000 1100 &MaxY)
    CVREF=CXCCCCC LVREF=1
    AUTOHREF CHREF=CX000000 LHREF=33;
  PLOT2 snydjcm*date / VAXIS=AXIS3;
  FORMAT date date7.;
RUN; QUIT;

ODS HTML CLOSE;

AXIS2 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0
  ORDER = &MinY to &MaxY by &Yrange
  VALUE=(HEIGHT=3PCT);
AXIS3 LABEL=NONE MAJOR=NONE MINOR=NONE STYLE=0
  ORDER = &MinY to &MaxY by &Yrange
  VALUE=(HEIGHT=3PCT JUSTIFY=LEFT);

FOOTNOTE1 FONT='Times New Roman' HEIGHT=4 PCT
  COLOR=CX006600
  JUSTIFY=LEFT
  BOX=1 BSPACE=0 IN BCOLOR=CXFFFF00
  LINK='Best_4000_xPixels.html'
  ' Click here to start vertical axis at zero ';
```

```

JUSTIFY=RIGHT
BOX=1 BSPACE=0 IN BCOLOR=CXFFFF00
LINK='Best_4000_xPixels.html'
' Click here to start vertical axis at zero ' ;

PROC CATALOG CAT=work.gseg KILL; RUN; QUIT;

ODS HTML PATH="C:\VIEWS54" (URL=NONE)
STYLE=Minimal GTITLE GFOOTNOTE
BODY="NotVZERO_4000_xPixels.html"
(TITLE="Dow Index - &DateRange");

PROC GPLOT DATA=work.DowByDay;
PLOT snydjcm*date / HTML=HTMLvar
NAME='NotVZERO' DESCRIPTION=' '
HAXIS=AXIS1 VAXIS=AXIS2
AUTOVREF CVREF=CXCCCCCX LVREF=1
AUTOHREF CHREF=CX000000 LHREF=33;
PLOT2 snydjcm*date / VAXIS=AXIS3;
FORMAT date date7.;
RUN; QUIT;

ODS HTML CLOSE;
ODS LISTING;

```

With the example of the code above, the reader could instead create a package of web pages and images that break the trend period into segments to be viewed separately, using push-button links on each web page to select the discrete window of the total time period to be viewed next.

NOTE: The unnatural choice of `FONT=ZAPF VALUE='E2'X` on the `SYMBOL` statements prevents ODS from creating the HTML code in an inefficient manner to provide the ALT text. With most common choices of SAS/GRAPH-selectable markers, if you try to open multiple very dense plots concurrently in Internet Explorer 7, the browser session will get fouled up, as might other application windows concurrently open.

LeRoy Bessler

Ask The Consultant

This part of VIEWS News is where you can get your technical questions answered. Send your questions to the Editor.

Protecting SAS Programs with SAS Data

Q: Is there a way to password-protect SAS programs?

A: Securing SAS programs can be a tough job as there is no existing mechanism in SAS to protect SAS programs from being edited. Occasionally, such protection is needed to prevent unauthorized users from accessing critical pieces of programs for personal gain. At the same time, by securing these programs, we can prevent any accidental changes made to the programs which might have a severe impact to their operation. The techniques here can also be used for commercialization of macros and user interfaces which will be very useful for developers.

Securing SAS programs has been a very important and crucial work in development for SAS. Very often, users have a tendency to extract pieces of programs which they think could be useful. Sometimes, these programs are not very important but in other cases, they may be part of a proprietary implementation of some models for the company. Protecting programs which are part of the operation process is also very important as any changes to the program may result in a disruption of operation. In light of this, it is extremely important for us to protect the SAS programs from

being abused or accidentally changed by others. In this article, I will discuss using data protection in SAS to protect the programs as well as encryption and decryption of the program for subsequent use.

A SAS data set has an interesting option that allows it to be password protected and thus preventing access to the data itself. This option is very often used in a data warehouse where data can be seen by all and access control is very loose. The SAS password can be easily assigned during the Data Step where it is given as part of the DATA statement:

```

DATA <data output> (PW = <assigned password>);
SET <data input> ;
RUN;

```

Once the above step has been run, the SAS data will require the password to start reading it or writing from it. According to the SAS documentation, if the password is lost, it will require a lot of resources from the SAS Institute to decrypt the password. Thus assuming that the password protection from SAS is very strong, this would provide us with an excellent solution to encrypting and protecting SAS macros.

Reading SAS Macros into SAS data sets is very easy and can be achieved using a simple INFILE statement with a large single-variable input. Below is an example of how this can be done:

```

DATA <SAS Macro Data Name>
(PW = <assigned password>);
LENGTH sourcecode $32676;
INFILE <SAS Macro Location> LRECL=32676
TRUNCOVER;
RUN;

```

Once the macro is in data form, it can be protected further by using data masking or encryption techniques to make it difficult for people to read, even if they manage to crack the password.

Reading in the macro is a very simple step, where a FILE statement is used to create a text file which contains the macro, and then reading it using an %INCLUDE statement. Below is an example:

```

DATA _NULL_;
SET <SAS Macro data> (PW = <password>);
FILE PRINT <location>;
RUN;

%INCLUDE <location>;

```

Once the macro has been compiled, it is now available for use.

Since the statement that reads in the code creates a SAS file, ensure that no-one else can access it. It is important to delete the resulting SAS file. To achieve that, we can send a command to the operating system to delete it. Below is an example of that command for Windows:

```
X "del <location>";
```

With the file removed, all traces of the macro have been erased except the data set containing the macro.

By combining all the above steps, we can have a macro that seeks to protect the other macros.

```

/*****
MACRO DATA PROTECTION PROGRAM
-----
THIS IS A MACRO CREATED TO READ IN MACRO FILES
TO CREATE SAS DATASETS WHICH ARE PASSWORD
PROTECTED AND ALLOWS FOR SUBSEQUENT DESCRIPTION
FOR USE. THIS MACRO ASSUMES THAT THE SAS DATA
SET PROTECTION ABILITIES ARE MUCH STRONGER THAN
THE TRADITIONAL APPROACH.
-----
INPUT      DESCRIPTION
-----
SAS_MACRO  THE LOCATION OF THE SAS
            MACRO/LOCATION OF SAS MACRO DATA
OUTPUT     THE DATASET FOR SAS MACRO/OUTPUT
            LOCATION OF THE MACRO PASSWORD
PASSWORD   USED TO ENCRYPT AND DECRYPT MODE
            1 IS ENCRYPTION MODE AND
            2 IS DECRYPTION MODE
-----
MAIN MACRO
*****/

%MACRO macro_protector(sas_macro=
                      ,output=
                      ,password=
                      ,mode=);

/*MODE 1 : ENCRYPTION PROCESS*/
%IF &mode. = 1 %THEN %DO;

/*DATA STEP TO ENCRYPT THE SAS MACRO*/
DATA &output.(ENCRYPT=YES PW=&password.);
  LENGTH text $500;
  INFILE "&sas_macro." TRUNCOVER;
  INPUT text $ 1-500;
RUN;
%END;

/*MODE 2 : DECRYPTION PROCESS*/
%IF &mode. = 2 %THEN %DO;

/*SAS TO DECRYPT THE DATASTEP*/
DATA _NULL_;
  SET &sas_macro.(PW=&password.);
  FILE "&output.";
  PUT text;
RUN;

/*STEP TO INCLUDE THE MACRO FOR
  SUBSEQUENT USE*/
%INCLUDE "&output.";

/*PREPARING TO SEND A OS COMMAND*/
OPTIONS NOXWAIT;

/*DELETE THE SAS MACRO TO PREVENT ANYONE
  TRYING TO READ IT*/
X "del &output.";
%END;
%MEND;

/*****
MACRO CALLING
*****/

%macro_protector(sas_macro=testmacro.sas
                ,output=pswd.sas
                ,password=SECRET
                ,mode=1);

```

This approach can be used to store SAS macros in such a way that prevent access is prevented. It is possible to combine the approach with other SAS products such as SAS/AF and SAS/FSP to make the protection more complete.

Murphy Choy, University College Dublin

Name De-duplication - Metaphone Algorithm

Q: What algorithm can be used to remove duplicate names in data?

A: Data de-duplication has always been a very important aspect of data warehousing, especially in customer relationship management. This is particularly important in places where personal identification is difficult and requires analysis to be done on the address and name of the customer to identify individual customers. Very often, these forms of de-duplication require the use of a phonetics and addressing algorithm which breaks down the information into usable and comparable forms. In this article, I will demonstrate how names can be analysed using the Metaphone Algorithm and de-duplication applied.

In many databases especially those in the customer-relation management area, there are many duplicated records of customers where they may have registered for multiple accounts or purchased at multiple stores of a company which does not have a centralized customer database system. To have a unified view of the customers, it is very important for the company to de-duplicate the customer name lists and consolidate information about the customer. There are many approaches such as using phonetic algorithms in conjunction with addressing systems to handle this situation as well as the use of fuzzy logic to identify customers. In this article, I will discuss the Metaphone Algorithm which is a phonetic algorithm that is often applied to names for name de-duplication.

The Metaphone Algorithm was first developed by Lawrence Phillips in 1990 to counter the inherent deficiencies in the original Soundex algorithm. The Metaphone Algorithm has a similar semantic and phonetic replacement process, which allows it to remove unnecessary differences, making spelling differences nearly the same. Below is the original algorithm as described by Lawrence Phillips.

```

METAPHONE USES THE FOLLOWING TRANSFORMATION
RULES: DOUBLED LETTERS EXCEPT "C" -> DROP
2ND LETTER. VOWELS ARE ONLY KEPT WHEN
THEY ARE THE FIRST LETTER.
-----
B -> B    UNLESS AT THE END OF A WORD AFTER "M"
          AS IN "DUMB"
C -> X    (SH) IF -CIA- OR -CH-
          S    IF -CI-, -CE- OR -CY-
          K    OTHERWISE, INCLUDING -SCH-
D -> J    IF IN -DGE-, -DGY- OR -DGI-
          T    OTHERWISE
F -> F
G -> -    SILENT IF IN -GH- AND NOT AT END OR
          BEFORE A VOWEL IN -GN- OR -GNED-
          (ALSO SEE DGE ETC. ABOVE)
          J    IF BEFORE I OR E OR Y IF NOT DOUBLE GG
          K    OTHERWISE
H -> -    SILENT IF AFTER VOWEL AND NO VOWEL
          FOLLOWS
          H    OTHERWISE
J -> J

```

```

K -> -      SILENT IF AFTER "C"
      K      OTHERWISE
L -> L
M -> M
N -> N
P -> F      IF BEFORE "H"
      P      OTHERWISE
Q -> K
R -> R
S -> X      (SH) IF BEFORE "H" OR IN -SIO- OR
            -SIA-
      S      OTHERWISE
T -> X      (SH) IF -TIA- OR -TIO-
      O      (TH) IF BEFORE "H"
      -      SILENT IF IN -TCH-
      T      OTHERWISE
V -> F
W -> -      SILENT IF NOT FOLLOWED BY A VOWEL
      W      IF FOLLOWED BY A VOWEL
X -> KS
Y -> -      SILENT IF NOT FOLLOWED BY A VOWEL
      Y      IF FOLLOWED BY A VOWEL
Z -> S

```

```

-----
INITIAL KN-, GN- PN, AE- OR WR-
      -> DROP FIRST LETTER
INITIAL X-
      -> CHANGE TO "S"
INITIAL WH-
      -> CHANGE TO "W"

```

So how is it being used for name de-duplication? Let us use a few examples of names to demonstrate.

In the English naming system, there are usually a variety of ways to spell the same name, and while they are different semantically, they are phonetically the same. Sometimes, the name is not filled in by the customer, but instead by a telemarketer or sales person, resulting in a slightly different spelling from what it should have been. Below are examples of the name pairs which are similar to each other.

<u>Original Form</u>	<u>Metaphone</u>
Ian	IN
Iain	IN
Vallis	FLS
Fallis	FLS
Davis	TFS
Davies	TFS

From the list we can see that names which are similar to one another in phonetics are assigned similar structures. This allows us to start filtering names within the system and de-duplicating the names which are similar in nature.

The Metaphone Algorithm is not the most perfect algorithm. It has several shortcomings which make it only marginally stronger than Soundex. Most phonetic algorithms are limited to the English language and perform poorly in other languages. The other issue is that phonetic algorithms are heuristic in nature and can perform poorly in a lot of cases. It is also comparatively more tedious to use a phonetic algorithm compared to fuzzy logic.

The process described above is one of the many ways that individuals can approach the problem. However, SAS has a product called Dataflux which deals exclusively with the problem of addressing and name de-duplication. For more information, please look up Dataflux in the SAS websites for more details.

```

/*****
TRANSFORMATIONS
*****/

%MACRO metaphone(input = NAMES, var = NAME);
/*****
DATA &input.;
SET &input.;
meta_&var. = UPCASE(&var.);

/*****
EXCEPTION CODING SECTION
*****/
/*HANDLING INITIALS SUCH AS 'KN', 'GN', 'PN',
'AE', 'WR'*/
IF SUBSTR(meta_&var.,1,2)
IN ('KN', 'GN', 'PN', 'AE', 'WR') THEN DO;
meta_&var. = SUBSTR(meta_&var.,2);
END;
/*HANDLING SITUATION WHERE THE FIRST LETTER
IS X*/
ELSE IF SUBSTR(meta_&var.,1,1) IN ('X')
THEN SUBSTR(meta_&var.,1,1) = 'S';
/*HANDLING SITUATION WHERE WH IS THE FIRST
LETTER*/
ELSE IF SUBSTR(meta_&var.,1,2) IN ('WH')
THEN DO;
SUBSTR(meta_&var.,1,2) = 'W';
meta_&var. = COMPRESS(meta_&var.);
END;

/*****
DOUBLE SUBSTITUTION
*****/
/*EXTRACTING THE LENGTH OF THE STRING*/
length_meta = LENGTH(meta_&var.);
/*DO FOR EACH CHARACTER OF THE STRING FOR
DOUBLE CHARACTERS*/
DO I = 1 TO length_meta - 1;
IF SUBSTR(meta_&var.,I,1) =
SUBSTR(meta_&var.,I+1,1) AND
SUBSTR(meta_&var.,I,1) ^= 'C' THEN DO;
SUBSTR(meta_&var.,I,2) =
SUBSTR(meta_&var.,I,1);
meta_&var. = COMPRESS(meta_&var.);
END;
END;

/*****
B -> B UNLESS AT THE END OF A WORD AFTER "M"
AS IN "DUMB"
*****/
/*SILENT SOUND CASE FOR B AFTER M*/
IF INDEX(meta_&var.,"B") = length_meta AND
SUBSTR(meta_&var.,length_meta-1,1) = "M"
THEN DO;
SUBSTR(meta_&var.,length_meta,1) = "";
meta_&var. = COMPRESS(meta_&var.);
END;

/*****
C -> X (SH) IF -CIA- OR -CH-
      S IF -CI-, -CE- OR -CY-
      K OTHERWISE, INCLUDING -SCH-
*****/
IF INDEX(meta_&var.,"C") > 0 THEN DO;
IF SUBSTR(meta_&var.,
INDEX(meta_&var.,"C")+1,1)
IN ("E","I","Y") THEN SUBSTR(meta_&var.,
INDEX(meta_&var.,"C"),2) = "S";

```

```

ELSE IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "C")+1,1) = "H"
    THEN SUBSTR(meta_&var.,
    INDEX(meta_&var., "C"),2) = "X";
ELSE IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "C")+1,2) = "IA"
    THEN SUBSTR(meta_&var.,
    INDEX(meta_&var., "C"),3) = "X";
ELSE SUBSTR(meta_&var.,
    INDEX(meta_&var., "C"),1) = "K";
meta_&var. = COMPRESS(meta_&var.);
END;

/*****
D -> J IF IN -DGE-, -DGY- OR -DGI-
T OTHERWISE
*****/
IF INDEX(meta_&var., "D") > 0 THEN DO;
IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "D")+1,2)
IN ("GE", "GY", "GI") THEN
SUBSTR(meta_&var.,
    INDEX(meta_&var., "D"),3) = "J";
ELSE SUBSTR(meta_&var.,
    INDEX(meta_&var., "D"),1) = "T";
meta_&var. = COMPRESS(meta_&var.);
END;

/*****
G -> - SILENT IF IN -GH- AND NOT AT END OR
BEFORE A VOWEL IN -GN- OR -GNED-
(ALSO SEE DGE ETC. ABOVE)
J IF BEFORE I OR E OR Y IF NOT DOUBLE GG
K OTHERWISE
*****/
IF INDEX(meta_&var., "G") > 0 THEN DO;
IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "G")+1,1) = "H" AND
INDEX(meta_&var., "G")+2
= LENGTH(meta_&var.)
THEN SUBSTR(meta_&var.,
    INDEX(meta_&var., "G"),2) = "";
ELSE IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "G")+1,1) = "N" AND
SUBSTR(meta_&var.,
    INDEX(meta_&var., "G")+2,1)
IN ("A", "E", "I", "O", "U") THEN
SUBSTR(meta_&var.,
    INDEX(meta_&var., "G"),2) = "";
ELSE IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "G")+1,3) = "NED" AND
SUBSTR(meta_&var.,
    INDEX(meta_&var., "G")+4,1)
IN ("A", "E", "I", "O", "U") THEN
SUBSTR(meta_&var.,
    INDEX(meta_&var., "G"),4) = "";
ELSE IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "G")+1,1)
IN ("Y", "E", "I") THEN
SUBSTR(meta_&var.,
    INDEX(meta_&var., "G"),1) = "J";
ELSE SUBSTR(meta_&var.,
    INDEX(meta_&var., "G"),1) = "K";
meta_&var. = COMPRESS(meta_&var.);
END;

```

```

/*****
H -> - SILENT IF AFTER VOWEL AND NO VOWEL
FOLLOWS
H OTHERWISE
*****/
IF INDEX(meta_&var., "H") > 1 THEN DO;
IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "H")-1,1)
IN ("A", "E", "I", "O", "U") AND
SUBSTR(meta_&var.,
    INDEX(meta_&var., "H")+1,1)
NOT IN ("A", "E", "I", "O", "U") THEN
SUBSTR(meta_&var.,
    INDEX(meta_&var., "H"),1) = "";
meta_&var. = COMPRESS(meta_&var.);
END;

/*****
Q -> K
*****/
IF INDEX(meta_&var., "Q") > 0 THEN
SUBSTR(meta_&var.,
    INDEX(meta_&var., "Q"),1) = "K";
meta_&var. = COMPRESS(meta_&var.);

/*****
K -> - SILENT IF AFTER "C"
K OTHERWISE
*****/
IF INDEX(meta_&var., "K") > 1 AND
SUBSTR(meta_&var.,
    INDEX(meta_&var., "K")-1,1) = "C" THEN DO;
SUBSTR(meta_&var.,
    INDEX(meta_&var., "K"),1) = "";
meta_&var. = COMPRESS(meta_&var.);
END;

/*****
P -> F IF BEFORE "H"
P OTHERWISE
*****/
IF INDEX(meta_&var., "P") > 1 AND
SUBSTR(meta_&var.,
    INDEX(meta_&var., "P")-1,1) = "H" THEN DO;
SUBSTR(meta_&var.,
    INDEX(meta_&var., "P"),1) = "F";
meta_&var. = COMPRESS(meta_&var.);
END;

/*****
X -> KS
*****/
IF INDEX(meta_&var., "X") > 0 THEN
SUBSTR(meta_&var.,
    INDEX(meta_&var., "X"),1) = "KS";
meta_&var. = COMPRESS(meta_&var.);

/*****
Z -> S
*****/
IF INDEX(meta_&var., "Z") > 0 THEN
SUBSTR(meta_&var.,
    INDEX(meta_&var., "Z"),1) = "S";
meta_&var. = COMPRESS(meta_&var.);

```

```

/*****
S -> X   (SH) IF BEFORE "H" OR IN -SIO- OR
        -SIA-
        S   OTHERWISE
*****/
IF INDEX(meta_&var., "S") > 0 THEN DO;
  IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "S")+1,1) = "H" THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "S"),2) = "X";
  ELSE IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "S")+1,2)
    IN ("IO", "IA") THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "S"),3) = "X";
  meta_&var. = COMPRESS(meta_&var.);
END;

/*****
T -> X   (SH) IF -TIA- OR -TIO-
        0   (TH) IF BEFORE "H"
        -   SILENT IF IN -TCH-
        T   OTHERWISE
*****/
IF INDEX(meta_&var., "T") > 0 THEN DO;
  IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "T")+1,1) = "H" THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "T"),2) = "0";
  ELSE IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "T")+1,2)
    IN ("IO", "IA") THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "T"),3) = "X";
  ELSE IF SUBSTR(meta_&var.,
    INDEX(meta_&var., "T")+1,2) = "CH" THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "T"),3) = "";
  meta_&var. = COMPRESS(meta_&var.);
END;

/*****
V -> F
*****/
IF INDEX(meta_&var., "V") > 0 THEN
  SUBSTR(meta_&var.,
    INDEX(meta_&var., "V"),1) = "F";
  meta_&var. = COMPRESS(meta_&var.);

/*****
W -> -   SILENT IF NOT FOLLOWED BY A VOWEL
        W   IF FOLLOWED BY A VOWEL
*****/
IF INDEX(meta_&var., "W") > 0 AND
  SUBSTR(meta_&var.,
    INDEX(meta_&var., "W")+1,1)
  NOT IN ("A", "E", "I", "O", "U") THEN DO;
  SUBSTR(meta_&var.,
    INDEX(meta_&var., "W"),1) = "";
  meta_&var. = COMPRESS(meta_&var.);
END;

```

```

/*****
Y -> -   SILENT IF NOT FOLLOWED BY A VOWEL
        Y   IF FOLLOWED BY A VOWEL
*****/
IF INDEX(meta_&var., "Y") > 0 AND
  SUBSTR(meta_&var.,
    INDEX(meta_&var., "Y")+1,1)
  NOT IN ("A", "E", "I", "O", "U") THEN DO;
  SUBSTR(meta_&var.,
    INDEX(meta_&var., "Y"),1) = "";
  meta_&var. = COMPRESS(meta_&var.);
END;

/*****
REMOVAL OF VOWEL TERMS
*****/
DO I = 1 TO 5;
  IF INDEX(meta_&var., "A") > 0 THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "A"),1) = "";
  IF INDEX(meta_&var., "E") > 0 THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "E"),1) = "";
  IF INDEX(meta_&var., "I") > 0 THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "I"),1) = "";
  IF INDEX(meta_&var., "O") > 0 THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "O"),1) = "";
  IF INDEX(meta_&var., "U") > 0 THEN
    SUBSTR(meta_&var.,
      INDEX(meta_&var., "U"),1) = "";
  meta_&var. = COMPRESS(meta_&var.);
END;
RUN;

/*****
%MEND;
*****/
%metaphone(input=NAME, var=NAME);

```

Murphy Choy, University College Dublin

Friday Quiz from Amadeus Software

The Friday Quiz originated from the Technical staff at Amadeus Software. Most Fridays we would share a recent problem that was encountered (and solved) to compare the approaches that were used. The enthusiasm for thrashing around programming ideas was such that we were encouraged to share the Friday Quiz with the general public.

Each quarter we will publish a new challenge for SAS Programmers to solve. Quizzes vary in style, but are designed not to require more than a few minutes thought. We encourage readers to share their solutions (anonymous or credited) and we will publish a selection at the end of each quarter.

We hope our quizzes serve as a pleasant distraction; but mostly we hope you learn new skills and techniques by sharing each others' code.

The Leopard that Changed its Baseline Spots

The summer Friday Quiz from Amadeus is based upon an internal training exercise that we used to train our new Consultants. We have provided the SAS program:

```

DATA leopard (INDEX=(subday=(subject day)))
  randomised (KEEP=subject treatment);
ATTRIB subject LABEL="Unique Subject ID"
  treatment LABEL="Randomised Treatment Group"
  day LABEL="Day Relative to First Dose (First
    Dose is Day 1)"
  spots LABEL="Count of Leopard Spots";
DO subject = 1 TO 20;
  treatment = RANTBL(731973,.6,.4);
  DO day = -7,-3,-1,0,1,2,3,5,10,20,40,100;
    spots= CEIL(100+(_n*treatment*50)) +
      INT(NORMAL(813501)*20);
    IF RANTBL(227460,.9,.1)=2 THEN spots=.;
    IF subject IN (1:4,6,7,9:14,16:20)
      THEN OUTPUT leopard;
    IF day = 0 THEN OUTPUT randomised;
  END;
END;
RUN;

```

Rangers in Kruger Park were concerned to notice that some of their leopards were losing their spots. They consulted a vet who observed that this was very unusual. Since there was no standard treatment, the vet suggested they try an experimental treatment on one group of the leopards, and a placebo treatment on the others.

The affected leopards were randomly assigned to one of the two treatments.

Over the course of a week, the park rangers rounded up the leopards, sedated them and counted their spots to establish a baseline measurement. Some leopards were counted more than once over this period. On Sunday, the park rangers collected all the leopards together and administered the treatments.

The baseline count was defined as the last non-missing count prior to treatment (days up to and including Day 0). The leopards' spots were counted again on Days 1, 2, 3, 5, 10, 20, 40 and 100 (relative to first day of dosing).

You should write a SAS program presenting a report containing the change from baseline of number of spots for each Day collection occurred, by randomised treatment. The changes should be summarised by presenting:

- n (the number of non-missing observations);
- Mean;
- Standard Deviation;
- Minimum;
- Maximum.

Additionally, the percentage change from baseline should be calculated and summarised with the same statistics.

Furthermore, it turns out leopards are quite nimble on their paws and consequently tricky to catch. Despite a number of leopards initially being randomised, not all leopards were caught when counting took place. The Park Rangers would like to see included within the report, the actual number randomised into each group.

The code accompanying this quiz generates two tables:

- LEOPARDS: Contains subject, day, spot count and the treatment group for each leopard. The grain of this table is one row per subject with data, per day.

- RANDOMISED: Contains subject and treatment. The grain of this table is one row per randomised subject.

If you wish us to consider your solution for publication, send it to fq@amadeus.co.uk by the 30 September 2011 and state if you are happy for Amadeus to publish your name along with the code. Solutions can be obtained after that date by request from the same email address, or by visiting the Friday Quiz web page www.amadeus.co.uk/fq.

David Shannon, Amadeus Software

Formats, Options, and Functions

This section is devoted to the description of useful, or unusual, SAS Formats, Options, and Functions.

[SAS 9.3] The SYSEXIST() function returns an indication of the existence of an operating environment variable.

[SAS 9.3] The AGREE= option has been added to PROC FREQ. When the AGREE= option is used and ODS Graphics is switched on, PROC FREQ will produce agreement plots.

[SAS 9.3] The HHMMSSw. informat can read times in the form "hhmmss" or "hh:mm:ss".

[SAS 9.3] Not a format, option or function change, but still worth including here: the ODS Graphics Editor, the ODS Graphics Designer and the ODS Graphics procedures have been moved from SAS/GRAPH to Base SAS in SAS 9.3!

PhUSE News

2011 Single Day Events

Earlier in 2011 I presented my 2010 PhUSE Conference paper on SAS 9.2 Graph Templates at 3 PhUSE Single Day Events (SDE) in Brussels, Frankfurt and Uxbridge (near London). Each day was very enjoyable, and, in spite of the similar topics, also very different.

The Brussels SDE was held on 28 April in the Business & Decision office in the suburbs of the city. I worked for 18 months in Belgium when I went freelance, so the city and culture (i.e. the beer and the food!) is very familiar to me, and I look forward to any opportunity to return there. All the presentations on the day were in English, a mixture of programming, CDISC and clinical topics were covered, and the international audience present picked up on all the jokes in the Oracle Health coding slides, which I was to see at 2 more SDEs!

The following week I was presenting in Frankfurt on 4 May. The Frankfurt SDE is always very friendly and I know a lot of attendees there from presenting there before. Apart from presentations by Stephen Bamford and myself, all the presentations here were in German, but with English slides, so I had very little difficulty following them. I did actually start my own presentation in German, but only to explain how awful my spoken German was, and then reverted back to English! Sadly the jokes in the Oracle Health coding slides were missed.

The Uxbridge SDE should have been held the week after the Frankfurt SDE, but was, from a personal perspective, thankfully rescheduled to 11 June. At first sight the agenda seemed to be a random selection of PhUSE presentations, but as the day went on it became clearer that the presentations were linked together in a

network of common ideas, e.g. graphics templates, planning, adverse event coding and programming quality. Each of these topics were covered in 2 or more presentations. Oh yes, and the mostly English audience got all of the jokes in the Oracle Health slides!

Philip R Holland, Newsletter Editor

2011 PhUSE Conference in Brighton

The 2011 PhUSE conference is scheduled for 9-12 October 2011 in Brighton, UK. The city is recognized as a leading and dynamic conference city, both nationally and internationally, combining a seaside location with a vibrant city atmosphere. Brighton is under an hour by rail from London and thirty minutes from London Gatwick Airport, and, in the city itself, everything is within walking distance.

The fantastic venue is without doubt a major draw, but ultimately it is the quality of the conference itself that will determine its success. The conference theme is 'Open Standards and Global Collaboration'.

We are again able to offer pre-conference training on Sunday 8 October 2011. This year we are pleased to announce that there will be two 'class room' based training sessions on 'Advanced Enterprise Guide' and a 'CDISC Workshop', and there will also be the opportunity to become SAS Certified.

Several 'Discussion Clubs' will be run at the conference which will provide the ideal occasion to share knowledge and views and to discuss and debate experiences in a number of topics including, standardisation and validation. There will also be an opportunity to contribute to a 'live' interactive Wiki where specific content will be discussed whilst the Wiki is being updated online in real-time.

For further information, please see the PhUSE website, www.phuse.eu.

PhUSE Committee

News

SAS Professionals Convention in Marlow

This year's SAS Professionals Convention was held over 3 days from 12-14 July at SAS UK in Marlow, and the weather was glorious every day.

The first day was devoted to SAS certification training in the morning and the exams themselves in the afternoon, with a record of 136 taking the exams on the day with a 70% pass rate, breaking the previous record of 110 set at the 2010 Convention here.

The second day was focused on industry topics. I went to BBC journalist Michael Blastland's presentation on the media's love of reporting misleading statistics (e.g. Anscombe's Quartet above). However, the rest of the day I spent meeting and talking with friends and SAS contacts, as I was less interested in the business presentations than the technical presentations scheduled for the third day. The Convention is intended as a networking event, so my second day was spent doing just that.

The third day was much more technical, with 2 big presentations for everyone in the morning, and 6 technology streams in the afternoon. Each 2-hour long stream was repeated,

so that delegates could attend more than one, so I went to the Architecture stream, where I heard about SAS Grid Architecture, and the SAS Enterprise Guide stream, where I heard about Enterprise Guide 5.1.

I'll probably only go to 2 days of the Convention next year, as my certification will now last until SAS 10 comes out, but I'm already looking forward to it!

Philip R Holland, Newsletter Editor

SAS Global Forum in Las Vegas

Las Vegas is bright lights, loud music and casinos, and, as the conference was held in Caesars Palace from 4-7 April, you had to cope with all the distractions! I was there as co-chair of the Reporting & Information Visualization section which ran from Tuesday morning to Thursday lunchtime, so, unfortunately, I didn't get to see that much of the rest of the conference, although I can say that it was vast, as I managed to lose my way to my own section several times! Fortunately the quality of the presentations in my section, when I did find it, was very good.

At next year's SAS Global Forum in Orlando I'll be a section chair of 2 sections, but I've been promised a maximum of 3 sessions, instead of the 5 sessions I had this year, so I'm hoping to be able to see a bit more of the rest of the SAS world!

Philip R Holland, Newsletter Editor

In Brief

- Back issues of *VIEWS News* are available from the *VIEWS* web site, and also from the *VIEWS News* page at www.sascommunity.org/wiki/VIEWS_News
- Andrew Ratcliffe's has a blog at notecolon.info.
- More SAS hints and tips can be found on the SAS FAQ pages www.hollandnumerics.com/sasfaq, on the Amadeus Software Tips pages at www.amadeus.co.uk/tips and in their free podcasts at: www.amadeus.co.uk/sas-technical-services/podcasts.
- There is a social networking group for SAS users on Ning called **SAS Professionals**, originally intended for SAS users in the UK, at www.sasprofessionals.net, but currently totalling over 5,300 members worldwide.
- Phil Holland has started a group page on Squidoo for other pages related to SAS, and would like to make it a central resource for SAS hints, tips, links and books at www.squidoo.com/groups/SAS_users
- Phil Holland has also created the SAS Contractors group on LinkedIn specifically for independent SAS consultants working in the UK and Europe. Requests to join the group should be made at: www.linkedin.com/groups?gid=2481512 and the associated SAS Contractor Resources page, which is open to all, can be found at: www.squidoo.com/sas_contractors
He also manages the SAS Professional Forum group on LinkedIn, which currently has over 11,300 members. Everyone is welcome to join, but discussions are restricted to SAS-related topics: www.linkedin.com/groups?gid=70702

- SAS users who use Twitter can now join a “twibe” at [twibes.com/SAS](https://twitter.com/SAS), which filters your tweets down to those containing “SAS”, “EnterpriseGuide” or “PROC”.

Diary

Are you organising an event that would be of interest to the VIEWS readership? Let us know as we are interested in all non-profit making events related to SAS.

September 2011

6 Using the SAS 9.2 Statistical Graphics Procedures, webinar

See the Amadeus web site www.amadeus.co.uk/webinars for further details.

October 2011

9-12 PhUSE Conference, Brighton

The Pharmaceutical Users Software Exchange (PhUSE) conference is the premier programming event for the pharmaceutical industry in Europe. See www.phuse.eu for more details.

11 SAS Programming Tips and Tricks for Base SAS, webinar

See the Amadeus web site www.amadeus.co.uk/webinars for further details.

April 2012

22-25 SAS Global Forum, Orlando, FL, USA

The SAS Global Forum (SGF, formerly called SUGI) is held each year and focuses on the technical aspects of SAS software. See www.sasglobalforum.org for more details.

All VIEWS event information will be posted and/or emailed to registered members of VIEWS, and will include an event application form. Please send your queries about any VIEWS events to event@views-uk.org, and don't forget to look at the web site for the latest news.

Contacts

LeRoy Bessler

Email: le_roy_bessler@wi.rr.com

Murphy Choy

University College
Dublin

Email: goladin@gmail.com

David Shannon

Amadeus Software

Email: fq@amadeus.co.uk

Web: www.amadeus.co.uk

PhUSE

Email: office@phuse.eu

Web: www.phuse.eu

SAS UK

Tel: +44-(0)1628-486933

Web: www.sas.com/uk

VIEWS

Web Site

Web: www.views-uk.org

Chairman

Email: chair@views-uk.org

Newsletter Editor

Email: newsletter@views-uk.org

Membership secretary

Email: membership@views-uk.org

VIEWS News is published quarterly by VIEWS International SAS Programmer Community. VIEWS is a not-for-profit organisation. Our Mission Statement and Rules can be viewed on our web site together with a list of current committee members.