

VIEWS

International SAS Programmer Community



News, Hints, Tips and Information
for SAS® Users

Issue 47, 3rd Quarter 2009

Again in this issue of *VIEWS News* I have a wide range of articles from new and familiar sources. I would like to welcome a new *VIEWS* Consultant Priyanka Singh, who, with Murphy Choy, has written an article about heteroskedasticity, which I knew nothing about. Our regular contributors LeRoy Bessler, Murphy Choy and the Amadeus Software Consultancy Team have written articles on combining SAS graphs and tables in Excel spreadsheets, more text editors for SAS programmers, format rounding, and the final part of the series of articles on regular expressions. Finally, my regular Formats, Options, and Functions section returns with some more unusual features of SAS software.

I presented at the Joint PSI/PhUSE One-Day Event in Marlow in June, and attended the 2nd day of the SAS Professionals Convention, also in Marlow, in July, so you'll find my reports about these events, and news of events occurring in the next few months in the News, PhUSE News and Diary sections.

I'm hoping Murphy has started a series of articles about your favourite text editor for SAS programming. 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 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.

Did You Know?

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

Visual + Detail = Effective Communication: SAS/GRAPH + Table on One Excel Worksheet

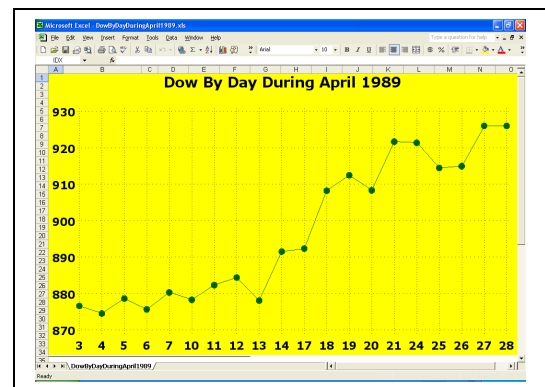
[SAS 9.2] Graphs facilitate and accelerate communication and decisions, but detail is needed for reliable communication and decisions. When data is prepared with SAS software, the ultimate user often insists on having it delivered via an Excel spreadsheet, with its next destination being a regrettable Excel graph.

In the previous issue of *VIEWS News*, I showed you how to package your results so that the user can view it as a maximally communication-effective web graph, but can toggle to a spreadsheet to inspect the supporting data as a unit, and manipulate it further non-graphically with Excel. For someone wanting data in Excel, but appreciating a better and finished graph, this is a useful deliverable.

Web deployment permits embedding hyperlinks in a graph, plus ALT text (also variously called pop-up text, flyover text, float over text, hover text, or a tool tip). A graph most likely to benefit from ALT text is a plot. Even with tick marks and/or reference/grid lines, it is impossible to know the precise y-value for a plot point. With ALT text, simply resting the mouse on a plot point displays its x- and y-values to any precision you choose to provide, in any format you like, and with whatever description you like for the x and y variables.

However, in this issue, I will show you how to package both the graph and the table on one Excel worksheet, if your user prefers everything in a single package and does not regard web-enabled pop-up plot point descriptions as useful. I will also explain important SAS/GRAPH version differences.

Here is the result using SAS 9.2:



Scrolling down, here is the PROC PRINT table in the same Excel worksheet:

Here is the code used:

```

DATA work.ToReport (KEEP = date snydjcm Day);
  LENGTH Day $2;
  SET sashelp.citiday (KEEP = date snydjcm);
  WHERE YEAR(date) = 1989 AND MONTH(date) = 4;
  Day = PUT(DAY(date), 2.);
RUN;

%LET Path = C:\VIEWS47;
%LET Title = Dow By Day During April 1989;
%LET BodyFileName = %SYSFUNC (COMPRESS(&Title));

ODS NORESULTS;
ODS LISTING CLOSE;

GOPTIONS RESET = ALL;
GOPTIONS DEVICE = GIF;
GOPTIONS CBACK = CXFFFF00;
  /* yellow background colour */
GOPTIONS FTEXT = 'Verdana/Bold' HTEXT = 4 PCT;
GOPTIONS XPIXELS = 980 YPIXELS = 576;
  /* Sized to fit inside an Excel 2000 window
  on a 1024 X 768 resolution screen.
  Inside an Excel 2003 window,
  the width is OK, but the height pushes
  the bottom of the graph too low. */

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

ODS TAGSETS.MSOFFICE2K
  PATH = "&Path" (URL = NONE)
  BODY = "&BodyFileName..xls"
  STYLE = Styles.Minimal;
/* (URL = NONE) is used to make the worksheet
and its embedded GIF file relocatable */

PROC GPLOT DATA = ToReport;
  TITLE FONT = 'Verdana/Bold' HEIGHT = 5 PCT
    "&Title";
  SYMBOL1 INTERPOL = JOIN LINE = 1 WIDTH = 1
    VALUE = DOT HEIGHT = 1.5
    COLOR = CX006600;
  AXIS1 LABEL = NONE MAJOR = NONE MINOR = NONE
    STYLE = 0;
  PLOT snydjcm * Day /
    NOFRAME VAXIS = AXIS1 HAXIS = AXIS1
    AUTOVREF LAUTOVREF = 33
    AUTOHREF LAUTOHREF = 33;
RUN;
QUIT;

GOPTIONS RESET = ALL;
  /* Prior GOPTIONS can affect PROC PRINT. */

TITLE1;

PROC PRINT DATA = ToReport LABEL;
  VAR date snydjcm;
  LABEL date = 'Date' snydjcm = 'Dow';
  FORMAT date WEEKDATX. snydjcm 5.;
RUN;

ODS TAGSETS.MSOFFICE2K CLOSE;

```

SAS 9.2 was used here. The results for 9.1.3 would be identical, except that the fonts would not be as thickly drawn, and these HEIGHT=1 plot symbols would be smaller.

In both SAS 9.2 and 9.1.3, omitting STYLE=Styles.Minimal (instead accepting Styles.Default) has several noteworthy effects. In both versions, it yields coloured area fills for the table, and seven useless empty rows between graph and table. If you omit GOPTIONS CBACK=CXFFFF00, which provides the yellow

background colour for the graph, in SAS 9.2 the graph has a light blue background, whereas in SAS 9.1.3 the graph has a white, i.e. no, background. The light blue background is due to the fact that SAS/GRAPH 9.2 uses a default graph style. All colours, fonts, symbols, and graph sizes are derived from whatever is the style, default or custom, currently in effect in your program or session. Procedure statement options and GOPTIONS can be used to override individual elements of that style. In 9.2, to revert to the simpler 9.1.3 environment, with no graph style, specify OPTIONS NOGSTYLE.

Also, in SAS 9.2 with Styles.Default the dotted reference lines are imperceptible. If you use solid reference lines by specifying LAUTOVREF=1 and LAUTOHREF=1, you will find that the lines are white (or very light grey). Thin black dotted lines are a better design choice if you want less conspicuous reference lines that are nevertheless visible. The colour of the reference lines can be controlled with CAUTOVREF and CAUTOHREF.

LeRoy Bessler, Assurant Health

Testing for heteroskedasticity in regression

Abstract

Heteroskedasticity is a term that is not commonly found in statistical textbooks and guides. Very often, SAS programmers and statisticians only find this term when they are looking a particular class of time series models called the GARCH and ARCH. These 2 classes of models has heteroskedasticity adjustment build into the model. However, occasionally, regression is used in time series and heteroskedasticity is not adjusted for in regression. In this article, a simple approach to testing for heteroskedasticity effect in regression models is highlighted and explained.

Introduction

In most models, there is usually an assumption concerning the structures of errors in the regression models. The error term is assumed to be normal and identically distributed. What this assumption is trying to convey is that all the error terms in the model are derived from a single normal distribution with mean of zero and variance being some arbitrary values. This condition is commonly known as homoskedasticity. However, occasionally this condition cannot be assumed for some particular cases. In the cases that this condition is violated, it is called heteroskedasticity.

What is heteroskedasticity?

Heteroskedasticity is the condition in which the variance for the variables in a particular model does not derive from a single normal distribution, but varies with the effect of time. Thus the error terms may be increasing or decreasing with respect to time, occasionally, it is a function of time. This condition has a lot of repercussions on the regression models.

Why is it important?

The presence of heteroskedasticity has severe impacts on the estimation of both the coefficients and the standard deviation of the model. While the coefficients do not become biased, the fluctuating variance may lead to underestimation of standard errors of the variables. This underestimation may lead to non-significant variables becoming significant. Both effects could lead to a wrong model under certain circumstances and it is important to test for the presence of heteroskedasticity in the data.

What are the relevant statistical tests?

There are many tests that can be used to detect heteroskedasticity, but not all of them are found in SAS. Below is a list of tests useful for heteroskedasticity detection:

- White test (see tinyurl.com/kk3ze2)
- Levene test (see tinyurl.com/noy216)
- Breusch-Pagan test (see tinyurl.com/mw78rd)
- Brown-Forsythe test (see tinyurl.com/l2awxc)
- Goldfeld-Quandt test (see tinyurl.com/kwc5sc)
- Cook-Weisberg test
- Park test
- Glejser test
- Harrison-McCabe test

Options in SAS

In PROC REG, the test implemented is White test (1980). This test can be activated by including the ACOV and SPEC keywords in the model statements. The ACOV statements, as mentioned in the SAS help documentation, produces a consistent estimate of the covariance. The SPEC statement performs a model specification test, which is derived from the White test (1980).

Conclusions

Heteroskedasticity is something that modeller who works with time dependent data should take note and test for. This is to ensure that heteroskedasticity does not affect the results of the modelling.

Acknowledgements

The motivation behind this article is due to a post by Priyanka Singh in the SAS-L list with questions on heteroskedasticity. It is hoped that this article will help to improve awareness of this problem in other analysts. Many thanks to Michael Raitchel, Jack Hamilton, Kevin Viel, and not forgetting Arthur Tabachneck.

Murphy Choy, University College Dublin

Priyanka Singh, Indian Institute of Management Ahmedabad

PRX 7: Regular expression magic

[SAS 9] The PRX functions in SAS use Perl regular expressions. The full Perl regular expression syntax is documented at perldoc.perl.org/perlre.html. SAS have not implemented the whole of this, but they appear to have implemented quite a number of features not covered in the SAS documentation. All the features described in this article are currently working for me on Windows in SAS 9.1.3, but they cannot be guaranteed to work on all platforms or for all versions of SAS.

Perhaps the most useful of these features is *non-greedy repetition*. As we noted before, SAS will normally find the *longest* sub-string of the source that matches our regular expression, i.e. the matching is *greedy*. We can however use the “?” character to make our repeat specifiers non-greedy. For example:

```
DATA _NULL_;
  rx = PRXPARSE("/a.*?a/i");
  instr = "Abracadabra";
  CALL PRXSUBSTR(rx, instr, pos, len);
```

```
sub = SUBSTR(instr, pos, len);
PUT "Hey Presto! Matching string is ""
    sub +(-1) "".";
RUN;
```

This gives the following output:

```
Hey Presto! Matching string is "Abra".
```

Rather than “*”, we have used a repeat specifier of “*?”. This means “as few times as possible”. With “*” alone, the matching string would have been the longest one that began and ended in “a”: “Abracadabra”. In principle, all repeat specifiers can be modified in this way, e.g. “{2,}?” for “at least twice, but as few times as possible”, and even “?” for “at most once, and preferably not at all”.

Did you spot the deliberate mistake in that last paragraph? In fact, the matching string found would normally have been “acadabra” rather than “Abracadabra”, because normally matching is case-sensitive. But in the program above, we have used another undocumented feature: that “i” at the end makes the whole expression *case-insensitive*.

The two other undocumented features we are going to look at are both useful, ironically, for documenting your PRX-related code.

First, an “x” right at the end of your expression, where the “i” was in the last example, will convert it to an *extended regular expression*. This is less impressive than it sounds; all it means is that you can insert white space within it without altering the meaning. This can greatly improve readability, though. If you actually want a space as a meaningful part of the expression, you will need to escape it with “\”, in fact it might be clearer if you made a sub-expression of it: “(\)”.

Second, you can insert *comments* into your expressions, in the form of specially-formatted subexpressions thus: “(?# This is a comment.)”. In the Perl terminology, this is one form of *extended pattern*; there are others.

These two features enable us to lay out our regular expressions in ways that make them much easier to understand, for example:

```
rx_pc = PRXPARSE(
"/      (?# Expression for checking postcodes)
[A-Z](1,2) (?# Either 1 or 2 letters)
\d(1,2)    (?# Either 1 or 2 digits)
[A-Z]?    (?# One more letter, optional)
(\ )+     (?# 1 or more spaces)
\a        (?# A digit)
[A-Z](2)  (?# Exactly 2 letters)
/ix");
/* The whole thing is case-insensitive */
```

SAS is liable to warn that your quoted string is getting unusually long and that you may have unbalanced quotes, but the expression will still work. Note though that if you are defining expressions for use with PRXCHANGE, you cannot use either of these readability features on the “new string” side of the expression.

The Perl page noted above describes other features which are undocumented by SAS but still work, e.g. *local* case-insensitivity, lookahead, negative lookahead, but it also gives features that did not work in SAS when I last tried them, such as POSIX character classes, lookbehind, and negative lookbehind. You could explore.

That concludes this series of articles on the PRX functions. I hope I have convinced you that they are less frightening than their reputation!

Amadeus Software Consultancy Team

Rounding your formats

Creating reports that show monetary values is one of the most common tasks that any organisation needs to perform when it comes to prepare a report. SAS comes with multiple pre-defined formats and the ability to generate custom ones.

When it comes to create your own format, PICTURE formats are one of the most flexible methods a programmer can use to define how data should be presented. Two very useful options that are available when customising monetary values are ROUND and MULT.

The ROUND option will round the values prior to applying the format. And the MULT option allows the programmer to specify a multiplying factor that will also be applied prior to displaying the format.

The following code divides by a thousand and round values using a picture format:

```
PROC FORMAT;
  PICTURE fmtround (ROUND)
    low - high = '009k' (MULT = 0.001);
RUN;

DATA picturef;
  SET sashelp.shoes;
  format_sales = PUT(sales, fmtround.);
RUN;

PROC PRINT DATA = picturef NOBS;
  VAR product sales format_sales;
RUN;
```

And in the results we can see the unformatted and formatted values:

Round and multiplier picture format		
Product	Sales	format_sales
Boot	\$29,761	30k
Men's Casual	\$67,242	67k
Men's Dress	\$76,793	77k
Sandal	\$62,819	63k
Slipper	\$68,641	69k
Sport Shoe	\$1,690	2k

Amadeus Software Consultancy Team

Ask The Consultant

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

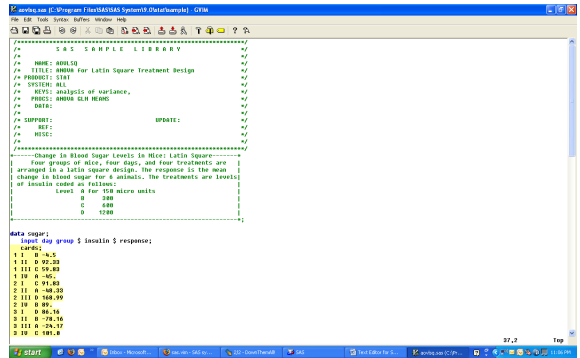
Text editors for SAS programmers - Part 2

Q: What text editors can be used to write SAS programs?

A: In this article, I will introduce some more text editors which are useful for various programmers utilizing different styles.

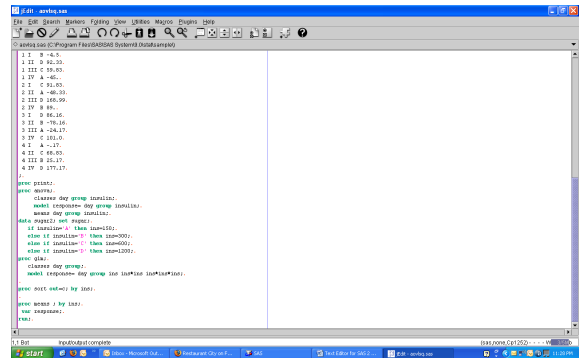
Vim

Vim is one of the first few original text editors with many capabilities and functions which caters for a wide variety of programming languages. It is first created by Bram Moolenaar and is a direct descendant of the original text editor Vi. Vim has two modes of inputs: insert mode and command mode. Like the original Vi, it is a modal program and allows for different approaches to code editing and writing. The complex functions in Vim enable users who are not used to mouse, or prefer direct keyboard utilization, to manoeuvre around with ease. It has one of the best syntax highlighter for SAS, amongst the text editors tested so far, and is the personal favourite of the author.



Jedit

Jedit is one of the few open source editors created using Java. It is a very compact editor with a small footprint on the computer. It is very efficient and has some highlighting for SAS files. The key feature of this editor is the folding ability, which can be useful at times. The automated indentation is also useful for certain scenarios.



Murphy Choy, University College Dublin

News

SAS Professionals Convention 2009

The first SAS Professionals Convention was held at the SAS UK offices in Marlow from 14-16 July as 3 Convention Days.

- 1st Day: Developing Your Career as a SAS Professional
- 2nd Day: Discovering SAS 9.2 Technologies
- 3rd Day: Solving Business Problems with SAS

I attended the 2nd Convention Day with 130 other delegates, sitting in the Enterprise Guide stream, but on the 1st Convention Day 100 had taken SAS Certification Exams, and on the 3rd

Convention Day there were 70 managers as delegates, so the Convention has been considered a great success.

The Enterprise Guide stream was attended by 22 delegates and looked at how the changes in Enterprise Guide 4.2 would help users. Alongside this stream there were 5 other streams: Data Integration, Analytics, Reporting, Architecture and JMP.

I'm hoping that this new conference format will be repeated next year, now that the SAS Forum UK is no longer going to be held in November.

Philip R Holland, Newsletter Editor

Joint PSI/PhUSE One-Day Event: Marlow

The Joint PSI/PhUSE One-Day Event in Marlow was held on 11 June with a theme of using new technologies in the pharmaceutical industry. Over 40 delegates enjoyed the varied presentations in the Library at Wittington House, which ranged from the use of SAS Drug Discovery to my own presentation about using Enterprise Guide to run suites of clinical trials programs, with topics in-between such as SAS/IML Studio, writing statistical add-ins for Enterprise Guide and the SAS Clinical Standards Tool.

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
- Amadeus is starting courses and seminars in Frankfurt am Main, Germany, later this year. Some of the seminars are listed in the Diary section below. A full list of courses and seminars can be found at: www.sascommunity.org/wiki/VIEWS_Events_Diary.
- More SAS hints and tips can be found on the SAS FAQ pages www.hollandnumerics.com/sasfaq and also on the Amadeus Software Tips pages at www.amadeus.co.uk/TipsList.aspx?ST=1
- There is a social networking groups for SAS users on Ning:
 - **SAS Professionals**, intended for SAS users in the UK, at www.sasprofessionals.net, but currently totalling more than 1,350 members worldwide.
- Phil Holland has also 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
- SAS users who use Twitter can now join a "twibe" at twibes.com/SAS, which filters your tweets down to those containing "SAS", "EnterpriseGuide" and "PROC".
- There is a new forum system for PhUSE members, including SAS-related streams at www.phenix-mtk.com/forums
- Phil Mason has a SAS tips newsletter, which can be requested by going to www.woodstreet.org.uk and then clicking on Tips. Previous tips can also be found on the web site.

<http://www.views-uk.org>

Formats, Options, and Functions

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

You may not have used the **VLABEL()** and **VLABELX()** functions before, but they return the label of a SAS variable as a text value. **VLABEL()** take a variable or array element as its argument, and **VLABELX()** requires a character expression containing the variable name. Both return the variable name if no label exists.

[SAS 9.2] The **MDYAMPMw.** informat reads datetime values in the form **mm-dd-yy<yy> hh:mm:ss.ss AM|PM**, where a special character, such as a hyphen (-), period (.), slash (/), or colon (:) separates the month, day, and year; and the year can be either 2 or 4 digits.

DKROCOND= can be used to control the level of error detection for output data sets during the processing of **DROP=**, **KEEP=**, and **RENAME=** data set options and the corresponding DATA step statements. **DKRCOND=** is the equivalent system option for input data sets. These options can take the following values: **WARN**, **WARNING**, **NOWARN**, **NOWARNING** or **ERROR**.

PhUSE News

PhUSE Conference in Basel, Switzerland

This year's PhUSE conference in Basel, from 19-21 October, is approaching quickly. More than 150 abstracts were submitted to the paper portal, and gave the conference committee a challenging, but also satisfying, job to select the best in order to make sure the conference will be of the highest quality! Finally 114 presentations were selected, with 11 streams and a University Day.

As announced previously, a PhUSE Management Stream and PhUSE Discussion Clubs will be organized to encourage discussions and debates. PhUSE Management Stream, which is growing on the success of previous year's presentations and subsequent discussions. The topics covered this year are:

- Lean / Six Sigma.... The 5 Y's
- Does a Sponsor Clinical Trial Reporting System fit into a Sponsor / CRO alliance?
- Juggling balls and resource management strategies from a CRO perspective
- Future Skills & emerging markets
- Considerations for Process development
- Managing and working effectively in a globally dispersed team across different time zones

PhUSE Discussion Clubs, which are new this year. Through a series of interviews and feedback sessions PhUSE has decided on the three topics below, which are considered critical to our profession:

- Open Source Code
- CDISC
- Accelerating Clinical Development

Throughout Tuesday, one of the conference rooms will focus on each of these three topics. There will be a short scene-setting presentation, followed by an open discussion covering the topic.

http://www.sascommunity.org/wiki/VIEWS_News

We are sure you agree that along with the vast number of presentations, PhUSE 2009 is promising to be an excellent opportunity to both contribute and learn more about our industry. If not done so already, you can register for the conference at www.phuse.eu/Registration2009.aspx

If you need any assistance, please contact the PhUSE office at office@phuse.eu.

We look forward to seeing you in Basel.

Benjamin Szilagyi (Conference Chair)

Heidi Curinckx (Conference Co-Chair)

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 2009

24 **SAS Programming Tips and Tricks**, Oxfordshire

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

October 2009

19-20 **Best Practices in SAS Statistical Programming in Regulatory Submission**, Hoofddorp, The Netherlands

These courses are intended for anyone directly or indirectly responsible for the creation, content or validation of summary tables, data lists and graphs used to support research, drug or medical device efficacy and safety in a regulatory submission. See tinyurl.com/5ayrmq for more details of the courses from Sunil Gupta.

18 **PhUSE Pre-conference Training**, Basel, Switzerland

See www.phuse.eu for more details.

19-21 **PhUSE Conference**, Basel, Switzerland

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.

28 **New and Enhanced Features of SAS 9.2**, Oxfordshire

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

November 2009

2 **SAS Programming Tips and Tricks**, Frankfurt am Main, Germany

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

19 **From Data to Knowledge: A Journey into SAS9**, Oxfordshire

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

December 2009

1 **SAS Programming Tips and Tricks**, Oxfordshire

See the Amadeus Software web site www.amadeus.co.uk for further 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

Amadeus Software

Email: info@amadeus.co.uk
Tel: +44-(0)1993-878287
Web: www.amadeus.co.uk

LeRoy Bessler Assurant Health

Email: le_roy_bessler@wi.rr.com

Murphy Choy University College Dublin

Email: goladin@gmail.com

Priyanka Singh Indian Institute of Management Ahmedabad

Email: priyankas@iimahd.ernet.in

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.