

VIEWS

International SAS Programmer Community



News, Hints, Tips and Information for SAS® Users

Issue 51, 3rd Quarter 2010

I don't like using excuses, but the lateness of this issue has broken my New Year's Resolution, and it was caused by trying to fit too much work into too little time. A warning that thorough planning is always a good idea! Even though it is late I still have another newsletter issue for you filled with SAS news, hints and tips that I hope you will find interesting. In this issue of *VIEWS News* there are articles from our regular contributors LeRoy Bessler and Murphy Choy, with articles on avoiding overlapping pie labels, the Gini Coefficient, and analysing the stock market with candlestick charts. Instead of the usual articles from Amadeus Software, David Shannon is giving you the chance to try out the Amadeus Friday Quiz. Finally, my regular Formats, Options, and Functions section has another collection of unusual and underused features of SAS software.

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.

Did You Know?

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

Avoiding the Perils of Pie Charts

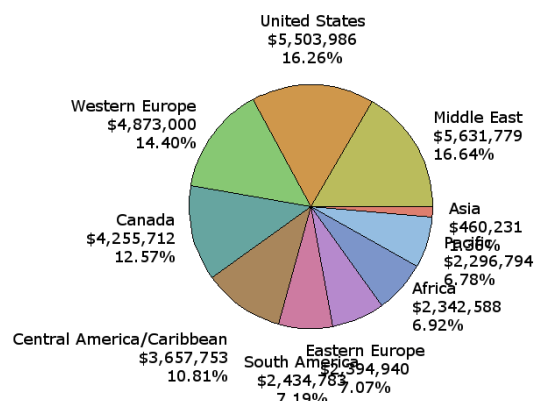
3D pie charts always misrepresent the relative size of shares of the whole, as I showed in *VIEWS News* issue 29. That peril is avoided easily by use of 2D pie charts.

The other common peril for pie charts, which has nothing to do with inherently wrong choice of tool, is the overlap of pie chart labels. SAS/GRAPH can detect the problem, and the SAS log then contains a list of eight possible circumventions. One of them is not to do a pie chart, which I realized long ago, and have addressed, e.g. in *VIEWS News* issue 45. One option suggested is to use a legend. Unfortunately, the SAS/GRAPH legend only displays the slice name, which does not guarantee that the slice

values and slice percentages are not vulnerable to label overlap. See my custom legend solution below (Case 2), as well as my way (Case 3) to further customize the legend, if that design suits your preference.

Case 1. Pie Chart Label Overlap Problem

Descending Pie Chart with Slice Name, Value, & Percent Shoes Sales By Region



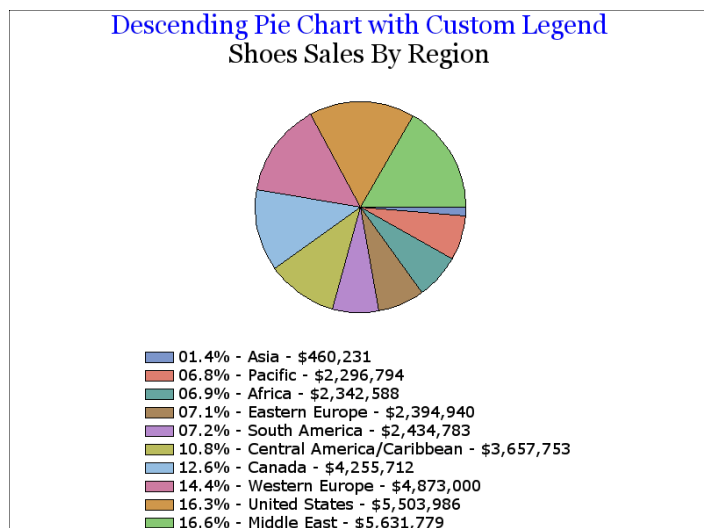
Here the code to create the pie chart above:

```
GOPTIONS RESET = ALL;
GOPTIONS DEVICE = PNG;
GOPTIONS BORDER;
GOPTIONS GSFNAME = anyone;
GOPTIONS HTEXT = 3 PCT FTEXT = 'Verdana';

TITLE1 HEIGHT = 5 PCT FONT = 'Georgia'
COLOR = CX0000FF
"Descending Pie Chart with Slice Name, Value,
& Percent"
;
TITLE2 HEIGHT = 5 PCT FONT = 'Georgia'
"Shoes Sales By Region"
;
FILENAME anyone "C:\PieChart1.PNG";

PROC GCHART DATA = sashelp.shoes;
  PIE region /
    SUMVAR = sales
    NOHEADING
    DESCENDING
    SLICE = OUTSIDE
    VALUE = OUTSIDE
    PERCENT = OUTSIDE
  ;
RUN;
QUIT;
```

Case 2. Custom Legend provides all information without overlap



NOTE: When the graph is shrunk to fit the width of the newsletter column, the colour samples in the legend become very small. When the sample area becomes very small, it can be difficult to distinguish one colour from another. An easy way to increase the area of the colour samples is to increase the length of the bar by use of the SHAPE parameter in the LEGEND statement. In this example, change 4 PCT to, say, 8 PCT or larger. At full size of the graph, there is no problem with 4 PCT.

Here is the code used to create the pie chart above:

```
PROC SUMMARY NWAY DATA = sashelp.shoes;
  CLASS region;
  VAR sales;
  OUTPUT OUT = ShoeSalesByRegion SUM = Sales;
RUN;

PROC MEANS DATA = ShoeSalesByRegion
  NOPRINT SUM;
  VAR Sales;
  OUTPUT OUT = PieTotal SUM = TotalSales;
RUN;

/* concatenate Percent and Value with the
   original Slice Name */
DATA SliceNameWithPercentAndValue;
  LENGTH NameWithPercentAndValue $ 46;
  SET ShoeSalesByRegion;
  IF _N_ EQ 1 THEN SET PieTotal;
  Percent = (Sales / TotalSales) * 100;
  NameWithPercentAndValue =
    TRIM(LEFT(PUT(Percent, Z4.1))) ||
    '%' || ' - ' ||
    TRIM(LEFT(Region)) || ' - ' ||
    TRIM(LEFT(PUT(Sales, DOLLAR10.)));
RUN;

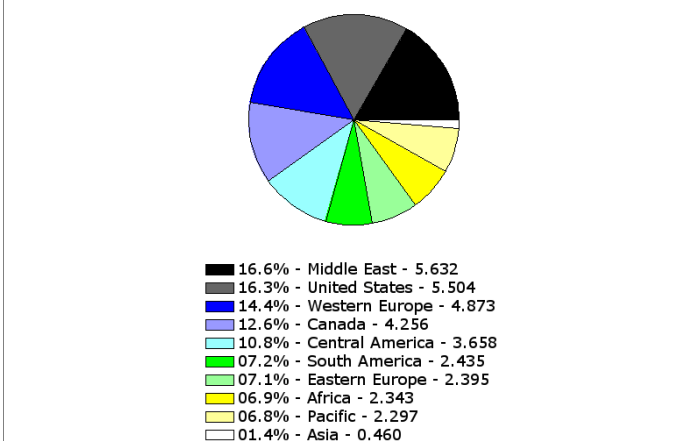
TITLE1 HEIGHT = 5 PCT FONT = 'Georgia'
  COLOR = CX0000FF
  "Descending Pie Chart with Custom Legend"
;
TITLE2 HEIGHT=5 PCT FONT='Georgia'
  "Shoes Sales By Region";
FILENAME anyname "C:\PieChart2.PNG";
LEGEND1 LABEL = NONE SHAPE = BAR(4 PCT, 2 PCT);
```

```
PROC GCHART
  DATA = SliceNameWithPercentAndValue;
  PIE NameWithPercentAndValue /
    SUMVAR = Sales
    NOHEADING
    DESCENDING
    LEGEND = LEGEND1
  ;
RUN;
QUIT;
```

If you want to order the legend entries from largest value to smallest value, and want to control slice area fill colour assignment, more pre-processing is required before the invocation of PROC GCHART, and macros must be used to dynamically create the PATTERN statements and the LEGEND statement. Also, for this ordering logic to succeed, the text in the legend entries must not exceed 32 characters. If this constraint is intolerable, see my solution in *VIEWS News* issue 45.

Case 3. Taking control of legend entry order and colour use

Shorten region names & values to order legend entries Shoes Sales By Region in Millions of Dollars



Here is the code used to create the pie chart above:

```
%MACRO PatternStatements;
  %DO i = 1 %TO 10;
    PATTERN&i. VALUE = PSOLID
      COLOR = &&SliceColor&i.
      REPEAT = 1
  ;
  %END;
%MEND PatternStatements;

%MACRO LegendEntries;
  %DO i = 1 %TO 10;
    "&&LegendEntry&i."
  %END;
%MEND LegendEntries;

DATA ColorList;
  /* Sequence colors from darkest to lightest,
     IF you want small slices to stand out. */
  pctseq = 1; COLOR = 'CX000000'; OUTPUT;
  pctseq = 2; COLOR = 'CX666666'; OUTPUT;
  pctseq = 3; COLOR = 'CX0000FF'; OUTPUT;
  pctseq = 4; COLOR = 'CX9999FF'; OUTPUT;
  pctseq = 5; COLOR = 'CX99FFFF'; OUTPUT;
  pctseq = 6; COLOR = 'CX00FF00'; OUTPUT;
  pctseq = 7; COLOR = 'CX99FF99'; OUTPUT;
  pctseq = 8; COLOR = 'CXFFFF00'; OUTPUT;
```

```

pctseq = 9; COLOR = 'CXFFFF99'; OUTPUT;
pctseq = 10; COLOR = 'CXFFFFFF'; OUTPUT;
RUN;

/* concatenate Percent and Shorter Value with a
   Shortened Slice Name */
DATA SliceNameWithPercentAndValue;
  LENGTH NameWithPercentAndValue $ 31;
  SET ShoeSalesByRegion;
  IF _N_ EQ 1 THEN SET PieTotal;
  Percent = (Sales / TotalSales) * 100;
  NameWithPercentAndValue =
    TRIM(LEFT(PUT(Percent,Z4.1)))
    || '%' || ' - ' ||
    TRIM(LEFT(SUBSTR(region,1,15))) || ' - ' ||
    TRIM(LEFT(PUT(Sales / 1E6, 5.3)));
RUN;

PROC SORT DATA = SliceNameWithPercentAndValue;
  BY DESCENDING Percent Region;
  /* Order slices from largest to smallest. */
RUN;

DATA ToSort;
  SET SliceNameWithPercentAndValue;
  pctseq = _N_; /* add a key to MERGE this data
                 with the ColorList */
  CALL SYMPUT('LegendEntry' || TRIM(LEFT(_N_)),
    TRIM(LEFT(NameWithPercentAndValue)));
RUN;

PROC SORT DATA = ToSort;
  BY pctseq;
RUN;

DATA SliceWithColor (DROP = pctseq);
  MERGE ToSort ColorList;
  BY pctseq;
RUN;

PROC SORT DATA = SliceWithColor;
  BY NameWithPercentAndValue;
  /* because SAS/GRAPH will always apply your
     PATTERN statements in the sort order of
     the SLICE name text */
RUN;

DATA _NULL_;
  SET SliceWithColor;
  CALL SYMPUT('SliceColor' || TRIM(LEFT(_N_)),
    TRIM(LEFT(color)));
RUN;

TITLE1 HEIGHT = 5 PCT FONT = 'Georgia'
  COLOR = CX0000FF
  "Shorten region names & values to order
  legend entries"
  ;
TITLE2 HEIGHT = 5 PCT FONT = 'Georgia'
  "Shoes Sales By Region in Millions of
  Dollars"
  ;
LEGEND1 ORDER = (%LegendEntries) LABEL = NONE
  ACROSS = 1 SHAPE = BAR(4 PCT, 2 PCT)
  ;
%PatternStatements;
FILENAME anyname "C:\PieChart3.PNG";

```

```

PROC GCHART
  DATA = SliceNameWithPercentAndValue;
  PIE NameWithPercentAndValue /
    SUMVAR = Sales
    NOHEADING
    DESCENDING
    LEGEND=LEGEND1
  ;
RUN;
QUIT;

```

LeRoy Bessler

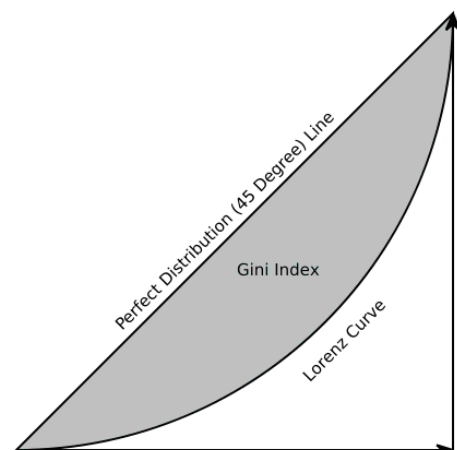
Gini Coefficient

The Gini Coefficient is one of the common statistics used in data mining and modelling. However, there exists no in-built SAS procedures to construct this statistics, and it is not often reported by the procedures themselves. While Gini Coefficients are produced in SAS products such as SAS Enterprise Miner, it is not found in Base SAS and SAS/STAT. In this article, I will explain the concepts behind the Gini Coefficient and the methodology behind its calculation. At the end of the article, I will also introduce a simple macro to calculate the Gini Coefficient.

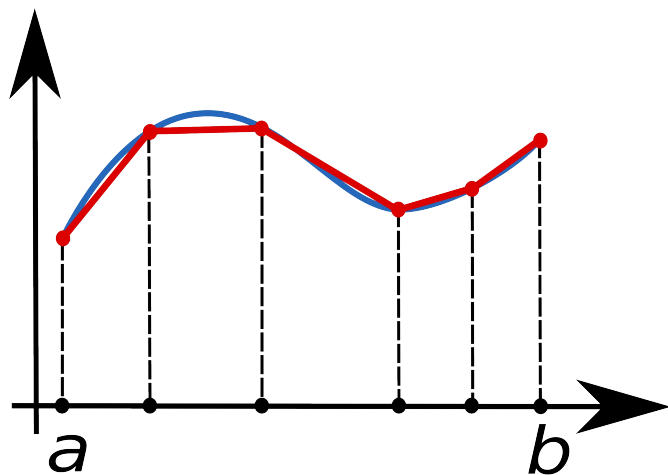
There are many statistical measures used by data miners and statistical modellers in their line of duty. Examples of these statistics include KS-Statistics, AIC, BIC, Mallow's CP, Loss Matrix and Misclassification Matrix. Some modellers also make use of graphical measures such as ROC curves and Lorenz curves. However, one of the more popular statistics, the Gini coefficient, is not often included in SAS procedure results. In the following sections, we will go through the history of this statistics, the calculation method and its uses.

The Gini Coefficient was invented by the great Italian statistician Corrado Gini in 1912 and published in the paper "Variability and Mutability". Ever since its introduction, it has been widely used in economics as the main measure of income inequality. Its usefulness in evaluating the spread of data for certain categories has led to its application in data mining and modelling. While it is gaining adoption steadily, it is still uncommon in most statistical modelling results and is rarely given to modellers by their statistical software.

The Gini Coefficient can be easily derived from two commonly used graphical methods, Lorenz curve and ROC curve. Let us look at the Lorenz curve in the diagram below. (Extracted from Wikipedia, 2009).



The Gini Coefficient (labelled Gini Index) is usually defined as the ratio of the areas on the diagram. Assuming the area between the line of perfect equality and Lorenz curve is A, and the area under the Lorenz curve to the axis is B, then the Gini Coefficient is computed as $A/(A+B)$. As $A+B = 0.5$, the Gini Coefficient, $G = A/(0.5) = 2A = 1-2B$. To measure the area under the Lorenz curve, the trapezoid rule is used. Below is an example diagram of trapezoid rule and the formula for calculation of area (Extracted from Wikipedia, 2009).



$$\int_a^b f(x) dx \approx (b-a) \frac{f(a) + f(b)}{2}.$$

Assuming that we are only computing for a small increment in the x-axis, the above rule will hold and allow us to use a numerical method to estimate the area under the curve. This approach lays the foundation for the Gini Coefficient macro given in the appendix of this paper. A detailed programming walk-through will not be done for the macro.

The Gini Coefficient is one of the few statistics that is not available in many base SAS procedures. However, a simple macro will allow us to compute this simple and beautiful statistic:

```

/*****
GINI COEFFICIENT MACRO
-----
GINI COEFFICIENT IS ONE OF THE MANY STATISTICS
THAT IS BEING USED TO MEASURE THE EFFECTIVENESS
OF THE MODEL. IN THIS SIMPLE MACRO, THE GINI
COEFFICIENT FOR A SCORED DATASET IS CALCULATED.
-----
INPUT          DESCRIPTION
-----
input          Input data set
yvar           Prediction or dependent variable
xvar           Score or independent variable
*****/

/*****
MAIN MACRO
*****/

%MACRO gini(input=, xvar=, yvar=);

```

```

/*****
DATA REDUCTION PROCESS
-----
IN THIS PARTICULAR STEP, THE TOTAL AMOUNT OF
DATA IS REDUCED TO JUST THE TWO VARIABLES WHICH
ARE NEEDED IN THE CALCULATION OF THE GINI
COEFFICIENT. THIS STEP ALSO HELPS TO REDUCE THE
AMOUNT OF PROCESSING TIME SUBSEQUENT TO THIS
STEP. A FILTER STATEMENT REMOVES ALL THE
INDETERMINATES OR NON-0,1 OBSERVATIONS.
*****/

DATA temp;
    SET &input. (WHERE = (&yvar. IN (0,1))
                    KEEP = &yvar. &xvar.);

RUN;

/*****
SUMMARIZATION PROCESS
-----
IN THIS PARTICULAR STEP, THE VARIABLES ARE
SUMMARIZED FOR GINI CALCULATIONS. THE DATA ARE
FIRST SUMMARIZED BY THE X VARIABLE AND THEN THE
OVERALL COUNTS AND SUMS ARE CALCULATED.
*****/

PROC SQL;
    CREATE TABLE summary_table AS
        SELECT &xvar.
              ,SUM(&yvar.) AS bad
              ,COUNT(&yvar.) AS total
        FROM   temp
        GROUP BY &xvar.
    ;

    SELECT SUM(&yvar) AS bad
          ,COUNT(&yvar.) AS total
    INTO    :totalbad
          ,:total
    FROM    temp
    ;

QUIT;

/*****
SUMMARIZATION PROCESS
-----
IN THIS STEP, THE CUMULATIVE PERCENTAGES ARE
CALCULATED AND GINI VALUES ARE THEN DERIVED
FROM THESE CALCULATION. THE GINI VALUES ARE
EVALUATED USING THE CLASSIC TRAPEZOID RULE
WHICH IS A NUMERICAL INTEGRATION METHOD. THE
APPROACH IS SIMILAR TO MANY CURRENT SOFTWARE'S
APPROACH.
*****/

DATA summary_table;
    SET summary_table;
    /*****
    RETAINING THE VALUES TO FORM A CUMULATIVE
    BEHAVIOUR
    *****/
    RETAIN cumbad 0;
    RETAIN cumtotal 0;
    /*****
    CALCULATION OF THE CUMULATIVE SUMS
    *****/
    cumbad = bad + cumbad;
    cumtotal = total + cumtotal;
    /*****
    CALCULATION OF THE CUMULATIVE PERCENTAGES
    *****/
    cumbadpct = cumbad / &totalbad.;
    cumtotalpct = cumtotal / &total.;

```

```

/*****
CALCULATION OF GINI BASICS
*****/
xdiff = cumtotalpct - LAG(cumtotalpct);
ydiff = cumbadpct + LAG(cumbadpct);
gini = 0.5 * (xdiff * ydiff);
RUN;

/*****
PRINTING OF GINI COEFFICIENT
-----
THIS STEP EVALUATES THE FINAL GINI VALUE.
*****/

PROC SQL;
  SELECT (2 * SUM(gini) - 1) AS gini
  FROM   summary_table
  ;
QUIT;

/*****/
%MEND;

/*****/
MACRO CALLING
*****/
%gini(input=scorecard
      ,xvar=score
      ,yvar=yesno
      );

```

Murphy Choy, University College Dublin

Ask The Consultant

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

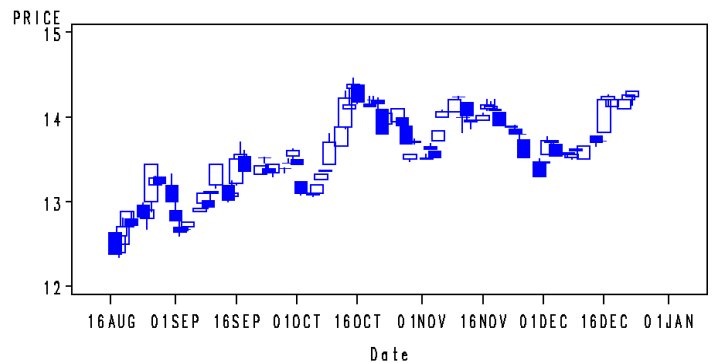
Simple Introduction to Candlestick Charts

Q: How can you analyse the stock market with SAS?

A: Technical analysis has been often touted as one of the key tools that is widely used in stock trading, commodities trading and even currency trading. It is widely cited as the tool of choice for many famous investors who have made their millions (maybe even billions!!) through this simple approach. Unlike its direct rival both in terms of theory and application, fundamental analysis, it has more varieties in terms of tools. In this particular paper, we will introduce users to the candlestick chart in SAS and how to use them to analyse the stock market.

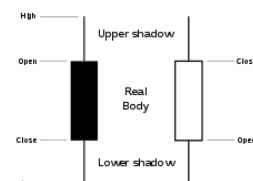
Candlestick charts have been an integral part of the technical analysis ever since they were adopted from the world of rice trading. They were first said to be developed by a rice trader by the name of Homma Munehisa in the mid-1700. Due to the presentation approach, they were easy to understand and key information can be extracted for any period of time. As a testament to the simplicity of the chart, below is an example of a candlestick chart created from SAS using a SAS macro incorporating some elements from a SAS/GRAPH example.

CANDLE STICK PLOTS



From the chart, we can see distinct number of blue and white bars as well as the overall trend. A quick glance at the chart indicates a fluctuating trend. At the same time, one can easily conclude that there are roughly equal numbers of positive and negative bar. We will go into the details of the various kinds of bars in the chart that will help us with stock trading. First, let us understand the anatomy of the bars.

Any candlestick will have several key elements in them. The most important elements are High, Low, Close and Open. As shown in the chart below, the bar consists of four main pieces of information, namely the high, open, close and low. Below is an example of how the bars were constructed. If the close is higher than the open, the box will be empty and vice versa.



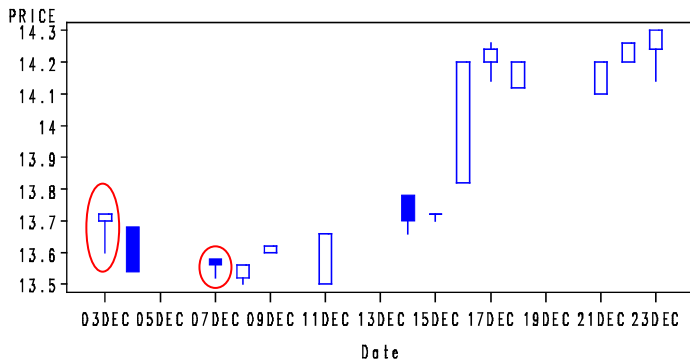
Due to the large number of possible combinations of the information, there are many different types of bar create in which some have importance in that they are signals for subsequent behaviour. In this subsequent section, we will explain one of the most important combinations in the candlestick chart analysis: the hammer and the hanging man.

The hammer, or hanging man, is a simple name that suggests a particular type of candlestick bar which has the following characteristics.

- The (thick) real body is at the upper end of the recent price range, whether it is solid or hollow is unimportant.
- A long, lower shadow (thin line) should be twice the height of the real body.
- It should have no upper shadow or a very short one.

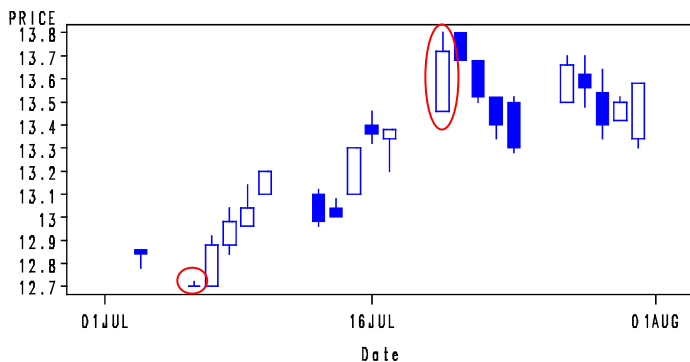
This particular type of bar with an extremely long shadow indicates a major reversal and the effects is even more profound when the shadow is extremely long. Typically when this type of bar is found, it hails the beginning of the reversal of trends regardless of its colour. From the following graphs, we can observe that the trend does hold true.

CANDLE STICK PLOTS



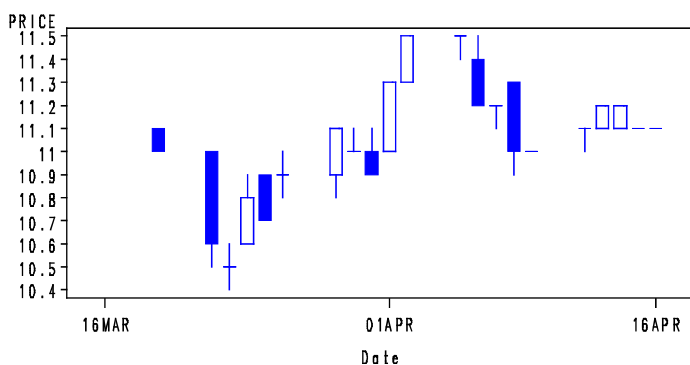
From the chart above, we can observe that the first hammer has resulted in a downward trend and using the same logic, we can also observe that the second hammer herald the beginning of the rising trend.

CANDLE STICK PLOTS



From the chart above, we can also observe that the 2 hammers started the upward and downward trend in this case. Again the hammer has been seen to be a useful indicator for detecting major reversal of trends.

CANDLE STICK PLOTS



There are many different type of bars as shown in the diagram above that have many different meanings and it is beyond the scope of this article. Interested readers should refer to more specialized books which contain more interesting guidelines than the one above.

```

/*****
CANDLESTICK CHART
*****/

%MACRO candle(input=, noofdays=);

/*****
DATA PROCESSING
*****/

PROC SORT DATA = &input.;
  BY DESCENDING date;
RUN;

/*****
THE FOLLOWING SECTION IS PART OF THE EXAMPLE
GIVEN IN THE SAS SAMPLE SITE
*****/

DATA test;
  SET &input. (OBS = &noofdays.);
RUN;

DATA candles;
  LENGTH COLOR FUNCTION STYLE $8;
  RETAIN XSYS YSYS '2' COLOR 'BLUE' SIZE 1;
  SET test;
  IF close > open THEN STYLE = 'EMPTY';
  ELSE STYLE = 'SOLID';

  /* DRAW THE OPEN/CLOSE BOX */
  FUNCTION = 'MOVE';
  XSYS = '2'; YSYS = '2';
  X = date; Y = close;
  OUTPUT;

  FUNCTION = 'POLY';
  XSYS = '7'; YSYS = '2';
  X = +1; Y = close;
  OUTPUT;

  FUNCTION = 'POLYCONT';
  Y = open;
  OUTPUT;

  FUNCTION = 'POLYCONT';
  X = -1; Y = open;
  OUTPUT;

  FUNCTION = 'POLYCONT';
  Y = close;
  OUTPUT;

  FUNCTION = 'POLYCONT';
  X = +0.5; Y = close;
  OUTPUT;

  /* DRAW HIGH TO CLOSE AND LOW TO OPEN */
  IF close > open THEN DO;
    FUNCTION = 'MOVE';
    XSYS = '2'; YSYS = '2';
    X = date; Y = close;
    OUTPUT;

    FUNCTION = 'DRAW';
    XSYS = '2'; YSYS = '2';
    X = date; Y = high;
    OUTPUT;

    FUNCTION = 'MOVE';
    XSYS = '2'; YSYS = '2';
    X = date; Y = open;
    OUTPUT;
  
```

```

FUNCTION = 'DRAW';
XSYS = '2'; YSYS = '2';
X = date; Y = low;
OUTPUT;
END;

/* DRAW HIGH TO OPEN AND LOW TO CLOSE */
IF close LE open THEN DO;
  FUNCTION = 'MOVE';
  XSYS = '2'; YSYS = '2';
  X = date; Y = open;
  OUTPUT;

  FUNCTION = 'DRAW';
  XSYS = '2'; YSYS = '2';
  X = date; Y = high;
  OUTPUT;

  FUNCTION = 'MOVE';
  XSYS = '2'; YSYS = '2';
  X = date; Y = close;
  OUTPUT;

  FUNCTION = 'DRAW';
  XSYS = '2'; YSYS = '2';
  X = date; Y = low;
  OUTPUT;
END;
RUN;

/*****/

SYMBOL1 I = NONE V = NONE R = 2;
AXIS1 MINOR = NONE OFFSET = (5, 5) PCT;
AXIS2 LABEL = ('PRICE') MINOR = NONE;
TITLE1 'CANDLE STICK PLOTS';
GOPTIONS DEVICE = JPEG;

/* PROC Gplot IS USED AS A 'HOLDING' AREA FOR
THE ANNOTATE, PLUS, Gplot GENERATES THE AXES.
OVERLAY PLOT REQUIRED SO THAT THE VERTICAL AXIS
HAS THE COMPLETE RANGE OF THE DATA,
I.E. HIGH TO LOW. */

PROC Gplot DATA = test;
  PLOT (high low) * date /
    OVERLAY HAXIS = AXIS1 VAXIS = AXIS2
    ANNO = candles
  ;
  FORMAT date DATE5.;
RUN;
QUIT;

/*****/

%MEND;

/*****/

%candle(input=market
, noofdays=30
);

```

SAS provides users with a simple and effective way of analysing the financial markets with the help of candlestick chart.

Murphy Choy, University College Dublin

Friday Quiz from Amadeus Software

The Friday Quiz originated from the Technical staff at Amadeus Software. Most Friday's we would share a recent problem that was encountered (and solved) to see what approaches each other would use. 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 other's code.

Birthday Promotion

Consider yourself a SAS Programmer who works in a team that execute marketing campaigns with SAS. You wish to create a personalised message to your recipients if their birthday falls within active campaign dates.

Here is a program that generates two SAS tables:

- CAMPAIGNS (three fictitious campaigns)
- POPULATION (50 fictitious customers and dates of birth).

```

DATA campaigns;
  LENGTH cid 8 name $30;
  INPUT cid name $ 3-16
        campaign_start DATE9.
        campaign_end: DATE9.;
  FORMAT campaign_: DATE9.;
  DATALINES;
1 New Holidays 01JAN2009 31MAR2009
2 January Sales 26DEC2008 31JAN2009
3 New Product 01FEB2009 24APR2009
RUN;

DATA population (KEEP=customer_uid
                  date_of_birth );
  ATTRIB customer_uid LENGTH=8
        LABEL="Unique Customer ID" FORMAT=Z6.
        date_of_birth LENGTH=8
        LABEL="Date of Birth" FORMAT=DATE9.;
  DO customer_uid = 1 TO 50;
    dd = INTNX('DAY', '31DEC2008'D,
              FLOOR((RANUNI(256) * 365) + 1));
    year = FLOOR((RANUNI(512) * 70) + 1921);
    date_of_birth = MDY(MONTH(dd), DAY(dd),
                      year);

    OUTPUT;
  END;
RUN;

```

Write a SAS program that:

- Merges the two tables creating one row for each member of the population, for each campaign.
- Creates a flag indicating whether the customers birthday (not date of birth!) falls between the campaign start and end dates.

Solutions can be obtained after 31 October 2010 by request from fq@amadeus.co.uk, 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.

The **FINFO()** function returns information about an external file reference, e.g. last modified date, size and access permissions. This platform-dependent function originated in SCL and requires the data from a number of other SCL functions: **FOPEN()**, **FOPTNUM()** and **FOPTNAME()**. Sample code can be found in the SAS online documentation at is.gd/gmb3Q.

[SAS 9] The **COMPRESS=** data set option specifies how observations are compressed in a new output SAS data set. Accepted option values are **NO**, **YES** (or **CHAR**) and **BINARY**.

[SAS 9.2] The **VMSZnw.d** informat reads, and the format of the same name writes, VMS and MicroFocus COBOL zoned numeric data.

News

Amadeus on iTunes

Amadeus are delighted to announce a new series of free tutorial podcasts; you can subscribe to them via iTunes or, alternatively, view them on the Amadeus web site:

www.amadeus.co.uk/podcasts

The first 4 titles are:

- Using the Macro IN Operator.
- Using PROC SCAPROC to Analyse the Inputs and Outputs of SAS Programs.
- Controlling SAS System Options with GETOPTION.
- Improving Performance with the SASFILE Statement.

Podcasts are free to view and will vary in length, starting with short 5-minute tips through to longer episodes covering a diversity of topics and potentially as part of a longer series of tips.

Also events to highlight:

- Webinar: Using SAS to Create Microsoft Excel Reports
- 5th November.
- Webinar: SAS Programming Tips and Tricks for Base SAS
- 25th November.
- Webinar: Descriptive Statistics with Base SAS
- 3rd December.

Amadeus Consultancy Team

PhUSE News

Single Day Events

These are one of the main benefits of membership and the board of directors are keen to ensure that the whole membership benefits as much as possible. This September we held our first SDE outside Europe with a very exciting SDE in Boston, where Matt Soukup from the FDA was presenting.

We are looking at holding other non-EU SDEs in India, China and Japan and are planning for the 2011 events.

Industry Standards

Ben Szilagyi has been leading the PhUSE initiative to create, document and share programming standards and open source code. This work is very much in its infancy and the detailed planning at this early stage will help the project deliver over the coming years.

e-learning

Leo Valmas has been leading the charge to get e-learning up and running for the membership. The first modules will be live very shortly, but Leo and his team are looking for more material. Please contact Leo at leo.valmas@phuse.eu if you would like to be involved.

Pharmaceutical Programming Journal

Dirk Spruck, the Editor of the journal is doing a sterling job in driving this project forward, but needs additional quality material. If you would like to discuss a future article, please contact Dirk via email at dirk.spruck@phuse.eu.

Vacant PhUSE Director Position

At last year's AGM we documented a process for filling the PhUSE Director Positions. We have had an open position (Regional Director, Germany) and unfortunately, following the process has resulted in no nominations. If you would like to find out more about this exciting position, please contact Stephen Bamford at stephen.bamford@phuse.eu.

2011 Planning

An advanced team has already started preparations for PhUSE 2011. Next year the conference will be held in the UK in Brighton from 9-12 October, there will be SDEs in London, Basel, Copenhagen, Frankfurt and Brussels, and provisionally in Philadelphia, San Francisco and Boston too. There will be more information during the next few months at www.phuse.eu.

Finally we should occasionally look behind us to see the journey that we have travelled. Since the first conference there have now been over 500 quality presentations and PhUSE bridges a knowledge gap that was present in the industry.

If you have any comments or thoughts about PhUSE, or if you would like to be more involved, please contact me via e-mail at stephen.bamford@phuse.eu.

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 excellent email newsletter "NOTE:" has been converted into a blog at notecolon.info.
- 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/tips.

- There is a social networking groups for SAS users on Ning called **SAS Professionals**, originally intended for SAS users in the UK, at **www.sasprofessionals.net**, but currently totalling more than 3,400 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
- SAS users who use Twitter can now join a “twibe” at **twibes.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.

November 2010

5 **Using SAS to Create Microsoft Excel Reports**, webinar
See the Amadeus Software web site

www.amadeus.co.uk/webinars.aspx for further details.

25 **SAS Programming Tips and Tricks for Base SAS**,
webinar

See the Amadeus Software web site

www.amadeus.co.uk/webinars.aspx for further details.

December 2010

3 **Descriptive Statistics with Base SAS**, webinar

See the Amadeus Software web site

www.amadeus.co.uk/webinars.aspx for further details.

March/April 2011

tba **PhUSE Single Day Event**, Basel, Switzerland

See the PhUSE web site **www.phuse.eu** for further details.

tba **PhUSE Single Day Event**, Brussels, Belgium

See the PhUSE web site **www.phuse.eu** for further details.

tba **PhUSE Single Day Event**, Copenhagen, Denmark

See the PhUSE web site **www.phuse.eu** for further details.

tba **PhUSE Single Day Event**, Frankfurt, Germany

See the PhUSE web site **www.phuse.eu** for further details.

April 2011

4-7 **SAS Global Forum**, Las Vegas, NV, 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.

May 2011

tba **PhUSE Single Day Event**, London

See the PhUSE web site **www.phuse.eu** 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.

*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

Email: **goladin@gmail.com**

University College Dublin

David Shannon

Email: **fq@amadeus.co.uk**

Amadeus Software

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.