

# Graphs for Evidence Exhibits in Court

---

-Nick Jones

## Executive Summary

Next year I start working as an analyst at an economic consulting firm who provides expert witness testimony in court cases. One of the primary tasks of analysts at the firm is producing graphs to be used as individual evidence exhibits to aid the expert witness. Because they are to be used in court, and for standardization purposes, the graphs all must conform to fairly strict visual standards regarding fonts, line widths, custom footers, placement of legends and titles, and so forth. I have created a system to export data from a statistical package and automatically create graphs that conform to the firm's visual standards. Additionally, analysts using my final workbook can use it to automatically format graphs in other workbooks, so they don't even need to import the data into the workbook I have created. My project will allow me, and other analysts who use it, to spend more time in exploratory data analysis and less time formatting graphs.

## Implementation Documentation

I created a set of instructions for the user, on the Instructions Worksheet, the first worksheet in the workbook. A screen capture of them is shown below.

## Instructions for Creating Evidence Charts

To automatically create a chart that conforms with the visual standards, place your data on the "Data" worksheet, beginning in Cell A1, and use the buttons in the Evidence Charts tab. If you are creating a line graph, a scatter plot, or a bubble chart, place the variable that you want on the horizontal axis in column A, and the series that you want to graph in the adjacent columns. By default, line and scatter plots graph two series, and bubble charts graph one series. For a bubble chart, by default the program will look in column B for the series to be graphed, and in column C for the weights.

For categorical charts like column charts, bar charts, or pie charts, you will need to already have your data in summary form, meaning only one value per variable per category. By default, bar and column charts will look for your category in column A and will look for two series to graph in columns B and C. Pie charts will look for your category in column A and your values in column B.

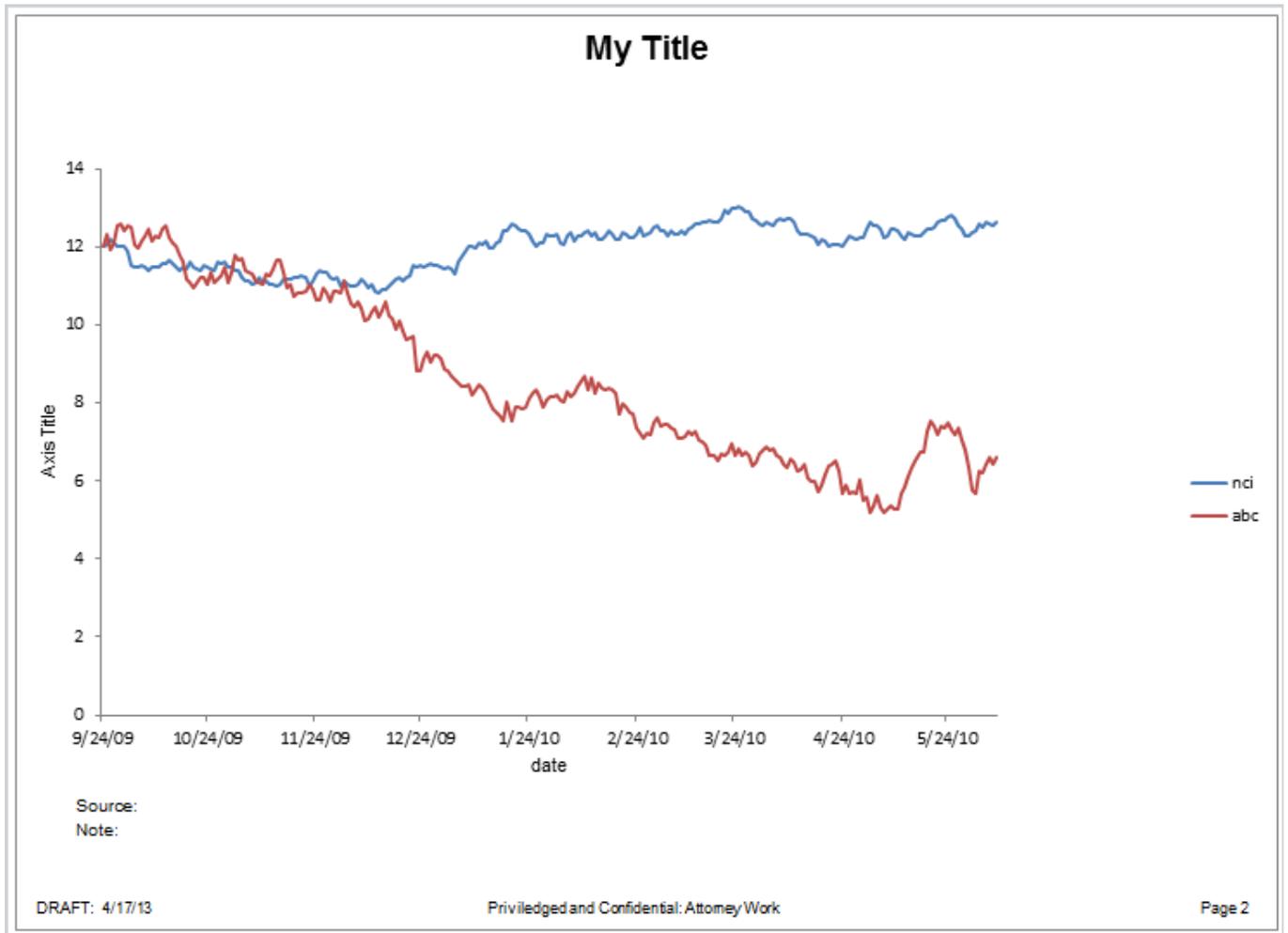
To format an already-existing chart in this or another workbook, first ensure that the chart is its own chart sheet, **not embedded in another worksheet as a chart object**. (To move a chart to its own sheet, right-click on the chart and select the Move Chart option.) Then, simply click on the "Select Chart to Format" button in the Format Existing Charts group in the Evidence Charts tab, select the chart you want to format from the drop down menu, and click the Format Chart button.

As stated in the instructions, the user begins by clicking on the Evidence Charts tab in the ribbon. The user has the option to three types of two way charts and three types of categorical charts. I chose these types of charts because they are the most common types of charts that are used as evidence exhibits. In addition to being able to create their own charts, the user can also select any existing chart this or any open workbook, and format it in according to the visual standards. However, this chart must be its own chart sheet; it cannot be embedded within another worksheet. This will not be problematic, since I include how to move a chart to its own chart sheet in the instructions, and more importantly, the firm has a policy of only using charts that are on their own chart sheets as evidence exhibits. A screen capture of the ribbon options is shown below. I made the icons for the Two Way Charts group myself in Photoshop, I found free icons with licenses allowing for commercial use without attribution for the Categorical Charts group, and used existing Microsoft icons for the Format Existing Charts group.



## Creating a New Chart in Conformity with the Visual Standards

Once the user has loaded the data that they wish to use for their chart into the Data worksheet, they just click on the type of chart that they want to make. I will illustrate this process with a Line Chart. After the user clicks on the Line Chart button, a chart is created, and a user form appears requesting further input. I will show the user form below, but first I will show the chart that is created if the user chooses not to specify anything in the user form. The new chart is created as a chart sheet, titled according to the pattern [Chart Type] [Number]. Since this is a line chart, and the first one we have made, the title of the chart sheet is Line Chart 1.



As stated previously, when the user clicks the Line Chart button, the above chart is created, and afterwards a user form appears on top of the chart, asking the user if they would like to specify more options. Doing it in this order allows the user to see what the chart currently looks like, by moving the form out of the way, and allows them to make sure that they fill in the user form correctly. The user

form is populated with the default values that the chart creates for things like axis titles, chart title, etc. The default title of the chart is My Title, and the horizontal axis title is taken from the data provided by the user. While the line graph only graphs two series by default, the user form allows the user to specify titles for up to four series, which they can add to the chart later. This option is also useful in formatting existing charts with more than two series. The default axis label for the vertical axis is just Axis Title. A screen capture of the user form is shown below.

Please Input Titles ✕

Graph Title

Vertical Axis Title

Horizontal Axis Title

Source

Note

Series 1 Name

Series 2 Name

Series 3 Name

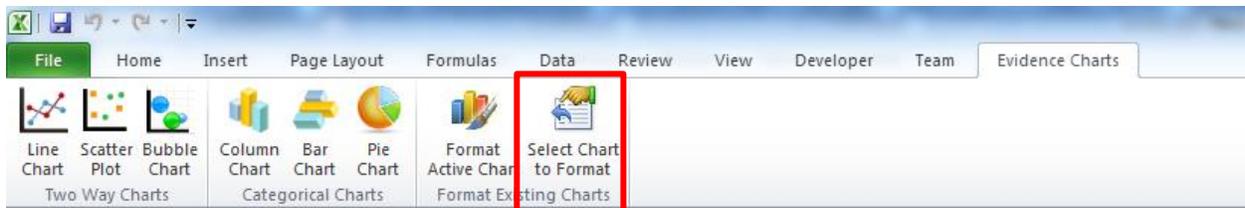
Series 4 Name

After the user fills in the form as desired and clicks the OK button, the changes are made to the chart. A representative chart is produced below.

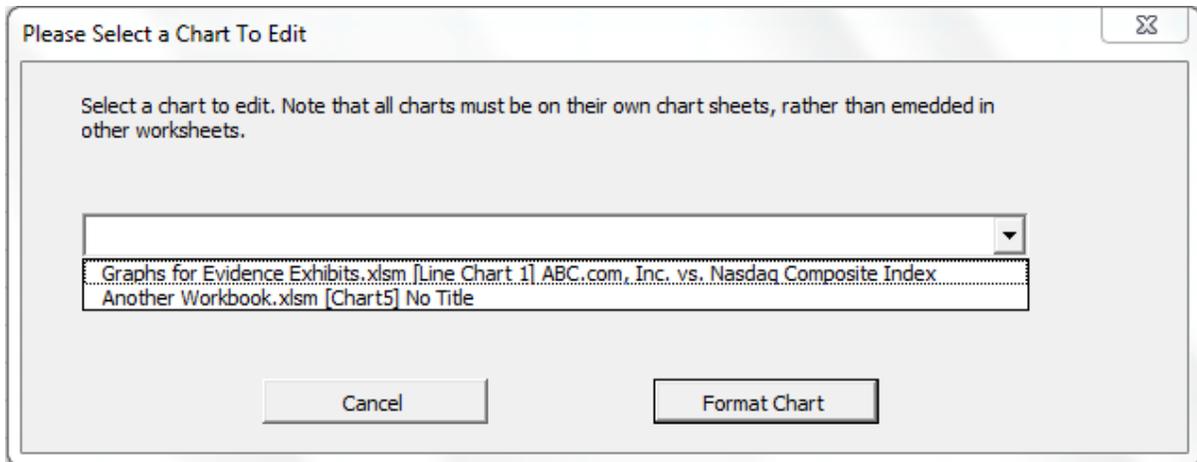


## Formatting an Existing Chart

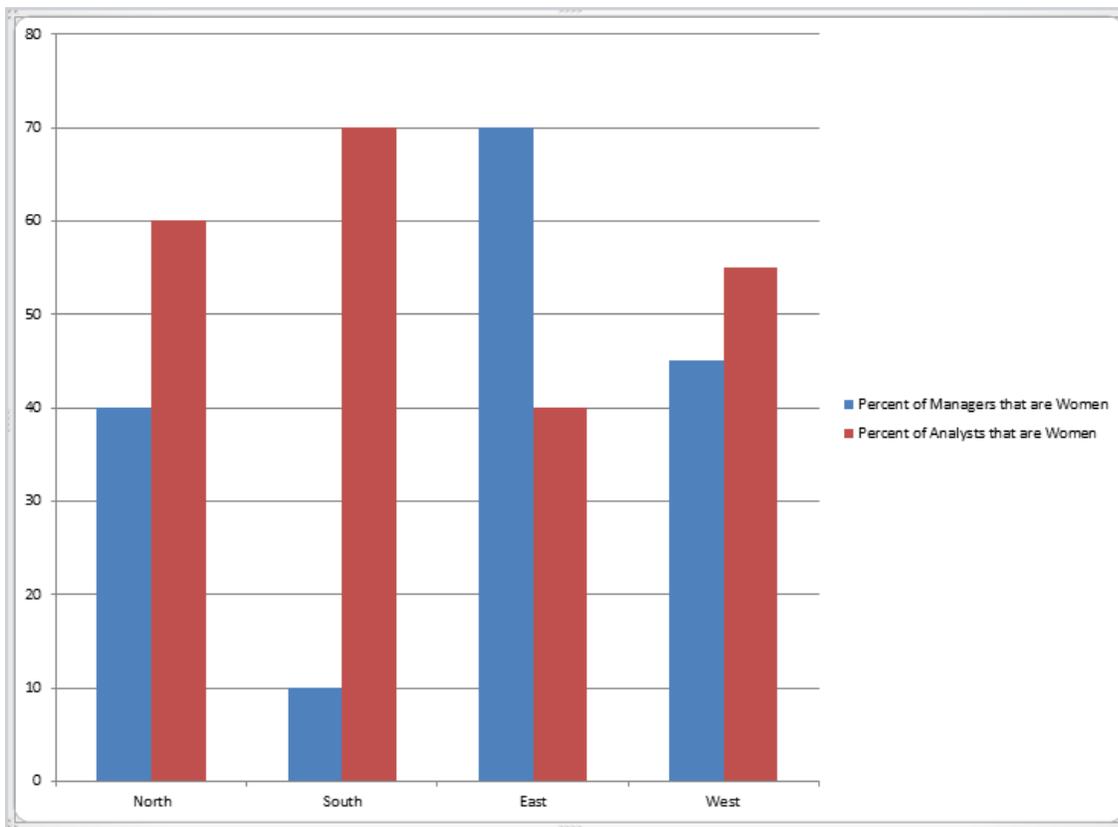
If the user already has a chart that they have made in this workbook or another workbook, they can format it using this workbook by clicking the Select Chart to Format button, as shown below.



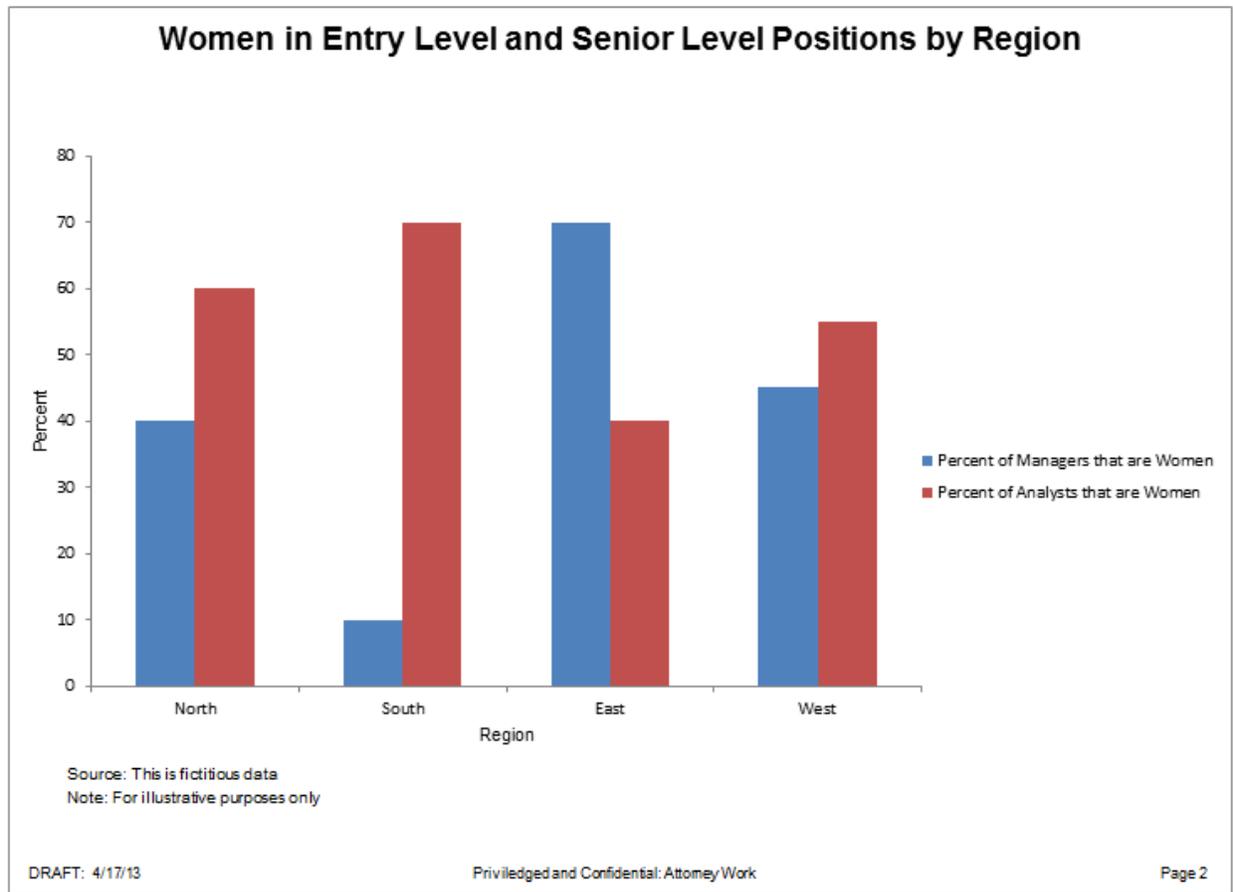
Once the user clicks on the button, another user form appears which allows them to select the chart that they would like to format. The chart must be its own chart sheet; it cannot be embedded in another worksheet. The combo box in the user form is populated with all of the chart sheets in all of open workbooks, and it lists them by the workbook name, the chart sheet name, and the chart title. If the chart has no title, then the combo box displays “No Title”, as shown below.



Once the user has selected a char that they would like to format, they click the Format Chart button on the user form to change the selected chart according to the visual standards. The code uses conditional logic, according to the chart type of the selected chart, to change only those elements that are appropriate for that type of chart. For example, if the selected chart is a pie chart, the program does not format the horizontal or vertical axes, because by charts do not have horizontal or vertical axes. An example chart that a user might like to format is shown below.



Once the user clicks the Format Chart button, changes are made to the chart, and the input user form appears, just like when the user creates their own charts using the program. The fields on the user form are populated with the values already present in the chart, such as title, horizontal axis label, etc. Once the user has filled in all of the desired fields in the user form, they click the OK button, and the final changes are made. The final formatted chart is shown below.



If the user does not want to select their chart from the drop down list in the user form, they can instead just click on the chart they want to format, and click the Format Active Chart button. If the user wants to format a chart from another workbook, however, they must use the user form.

## Learning and Conceptual Difficulties Encountered

I would say that there were two main “lessons” about VBA that I learned from doing this project: the usefulness of object variables, and the usefulness of using recordings as a jumping off point. Both of these things were stressed by Professor Allen in class, and I believed them when he said them, but doing this project made me learn them in a much deeper, more experiential sense.

I found object variables to be so useful in writing code, because of the Intellisense help that they give you, as well as the ability to pass them as argument to different functions. I used object variables in the homework and projects when we were supposed to, but I had never really faced a VBA challenge of

this magnitude prior to doing this project, so the use of object variables became less of fulfilling a requirement and more of a means of survival.

The macro recorder provided another means of survival as I carried out this project. Working with charts in VBA is difficult, because there are so many objects involved in charts, with their attendant methods and properties. I looked online for help, as well as in the object browser, but many times the way that I located the necessary objects and properties was simply recording myself making a change on a chart. Professor Allen said at the beginning of class that sometimes people who already have a programming background are at a disadvantage in this class, because they are used to coding everything themselves, so they don't every use the macro recorder. I would say that this was more or less true of me, up until the time I did this project. I really had to rely on the macro recorder to quickly find what I needed to change, and my VBA skills improved accordingly.

One final thing that I learned was the use of keyboard shortcuts to navigate in the VBE. Pushing Control + Up and Control + Down to navigate quickly between different subroutines was tremendously helpful. Also, pushing Control + R to move the focus to the Project Window and quickly change modules was another helpful shortcut, because I kept my own code in a separate module from my recorded code.

As far as elements that I wanted to include couldn't, there was only one thing that I ended up leaving out, which was exporting data directly from Stata, a statistical package, into my Excel workbook. I left this part out because, contrary to what I believed when I wrote my project proposal, Stata cannot replace a worksheet on a .xslm file; it can only replace a worksheet on a .xls or .xlsx file. Also, since this is a VBA class and not a Stata class, I decided to just have the user put their data directly onto the Data worksheet in my workbook.

## Assistance

The only person who assisted me on my project was Professor Gove Allen. He helped me figure out how to populate a user form with all of the charts on all open workbooks, and took me through the code as well. He helped me debug the code and make sure it was running correctly. This was tremendously useful for me, because I was considering dropping that functionality out of my project, but in the end I think that it proved to be its most useful feature.