The Stata Journal (2017)
17, Number 3, pp. 748–759

New graphic schemes for Stata: plotplain and plottig

Daniel Bischof
Department of Political Science
University of Zurich
Zurich, Switzerland
bischof@ipz.uzh.ch

Abstract. While Stata's computational capabilities have intensively increased over the last decade, the quality of its default graphic schemes is still a matter of debate among users. Some of the arguments speaking against Stata's default graphic design are subject to individual taste but others are not, for example, horizontal labeling, unnecessary background tinting, missing gridlines, and oversized markers. In this article, I present two new graphic schemes, `plotplain` and `plottig`, that attempt to address these concerns. These schemes provide users a set of 21 colors, of which 7 colors are distinguishable for people suffering from color blindness. I also give an introduction on how users can program their own graphic schemes.

Keywords: gr0070, plotplain, plottig, graphic scheme, colorblind

1 Introduction

Stata's computational possibilities have significantly expanded in the last decade. In contrast, less effort has been made to extend Stata's graphical possibilities by providing new graphic schemes. While StataCorp and Stata users have provided several small adaptations (for example, Juul [2003]), few ready-made and unique graphic schemes have recently been provided to users. For instance, Buchanan's (Forthcoming) `brewscheme` gives users the opportunity to adapt existing schemes. As such, `brewscheme` is a flexible, important advancement of Stata's graphical capabilities. With `brewscheme`, users can adapt figure designs themselves to fit their demands. While advanced Stata users will find `brewscheme` an excellent tool for adapting Stata's figure design to their liking, many less experienced users look for a ready-made graphic scheme that looks aesthetically attractive and addresses some concerns—outlined in detail below—about the design of Stata's default graphic schemes.

Thus, in contrast to Buchanan's work, I present two entirely new graphic schemes for Stata users. Users do not need to learn new code and syntax to use these graphic schemes but can use them directly after installation. I also give a short introduction to the code of Stata's graphic schemes to make the design of graphic schemes more accessible and potentially more attractive to a wider community of users.

The two graphic schemes, `plotplain` and `plottig`, are specifically designed for users who i) are interested in publishing figures in color, ii) want to make sure that colorblind persons can differentiate colors in graphics, and iii) want to adapt their graphics more closely to suggestions made in Tufte (2001) and Cleveland (1994) and specifically implemented in R's ggplot2 (Wickham 2009). Additionally, these new graphic schemes address some concerns with the default standard graphic schemes. Some of the limitations I raise in this article rest on personal taste, while other limitations have been raised elsewhere (Juul 2003; Buchanan Forthcoming). The latter issues are labeling, gridlines, background tinting, framing, and the colors used in figures (Tufte 2001; Cleveland 1994). Stata users often invest time and effort in writing excessive code to adapt figures (Buchanan Forthcoming), but users might want to spend less time writing code and spend more time thinking about how to present their data in a way that the story of their research stands out.

The first scheme I present is a minimalistic graphic scheme, plotplain, that addresses concerns about Stata's monographic scheme (for example, gridlines parallel to the y axis, horizontal labels on the y axis, and marker size). The second scheme, plottig, is mainly based on the insights from Cleveland (1994, 100–101) and largely replicates ggplot2, which was originally designed for the statistical software R (Wickham 2009).

While some issues discussed in the upcoming section have already been addressed by Juul's (2003) lean scheme, publication standards have changed significantly since Juul's (2003) publication. Today, many journals accept colored figures—especially those published online (for example, *PLOS ONE*). Thus, plotplain and plottig not only present a layout for figures that differs from the lean scheme but also come with two newly composed color palettes. The first palette ensures that each color is distinguishable for colorblind viewers. Color blindness is not equivalent to a loss of complete color vision; colorblind people tend to struggle differentiating only certain colors from each other (for example, red from green). It is a fairly common phenomenon; in a room of 250 people, more than 4% (>10 people) may be colorblind (http://jfly.iam.u-tokyo.ac.jp/color/). We should acknowledge this fact when creating our figures and ensure that we rely on colors that can be distinguished by colorblind viewers; we should also use different shapes for each color we draw. The second palette is inspired by ColorBrewer (Harrower and Brewer 2003; http://colorbrewer2.org) and provides a palette of 14 colors.

In this article, I first outline the key concerns and shortcomings of Stata's default graphic schemes. I then explain how plotplain and plottig attempt to address these concerns. In the final section, I give an introduction on how users can design their own graphic schemes.

2 Limitations of default Stata graphic schemes

I list all concerns I find with the default Stata schemes below. The critique is mainly based upon Cleveland's (1994) work. Cleveland's major advice and takeaway point is "Make the data stand out. Avoid superfluity" (Cleveland 1994, 24). I suggest that the default Stata figure design often comes with superfluous implementations. Therefore, the new graphic schemes attempt to avoid such “chartjunk” (Tufte 2001). The generic goal of these graphic schemes is to minimize the amount of chartjunk and maximize the information on the data we try to communicate to our audience. In some instances, readers might disagree with my opinion. To address these concerns, I outline two possibilities to adapt my graphic schemes in the last part of the article.

Figure 1 shows a scatterplot and figure 2 a lineplot, both with a frequently used Stata graphic scheme (`s2mono`).

```
. graph twoway
> (scatter nkill nevents if region==12 & nevents<=500 & nkill<=1000)
> (scatter nkill nevents if region==7  & nevents<=500 & nkill<=1000)
> (scatter nkill nevents if region==6  & nevents>=500 & nkill<=1000),
> scheme(s2mono) legend(label(1 Western Europe)
> label(2 North America) label(3 MENA)) title("s2mono")
```

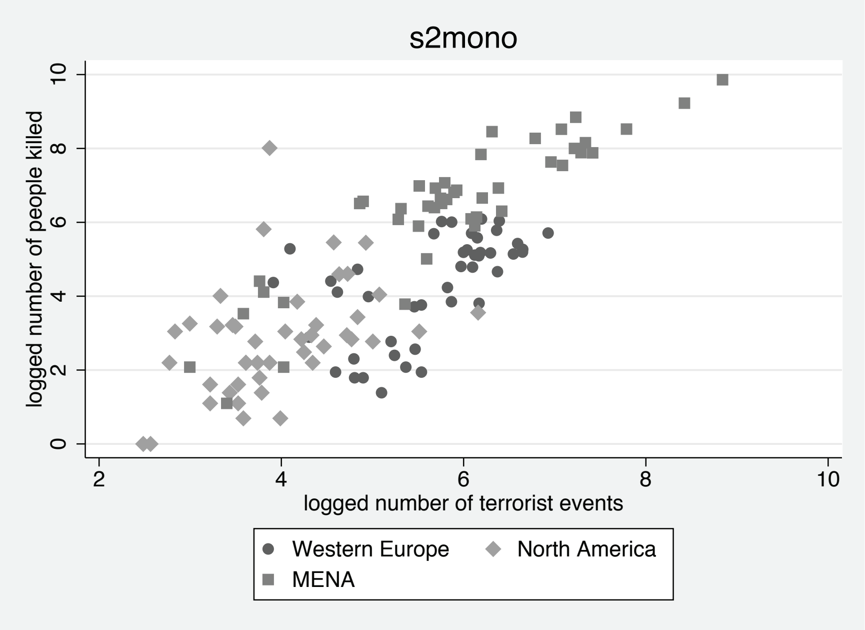

Figure 1. Limitations of `s2mono`: scatterplot

```
. graph twoway
> (line nevents iyear if region==12)
> (line nevents iyear if region==7) (line nevents iyear if region==6),
> scheme(s2mono)
> legend(order(1 "Western Europe" 2 "North America" 3 "MENA"))
> title("s2mono") xline(2003)
> text(8 2003  "Iraq war", place(e)) xlabel(1970(10)2015)
```

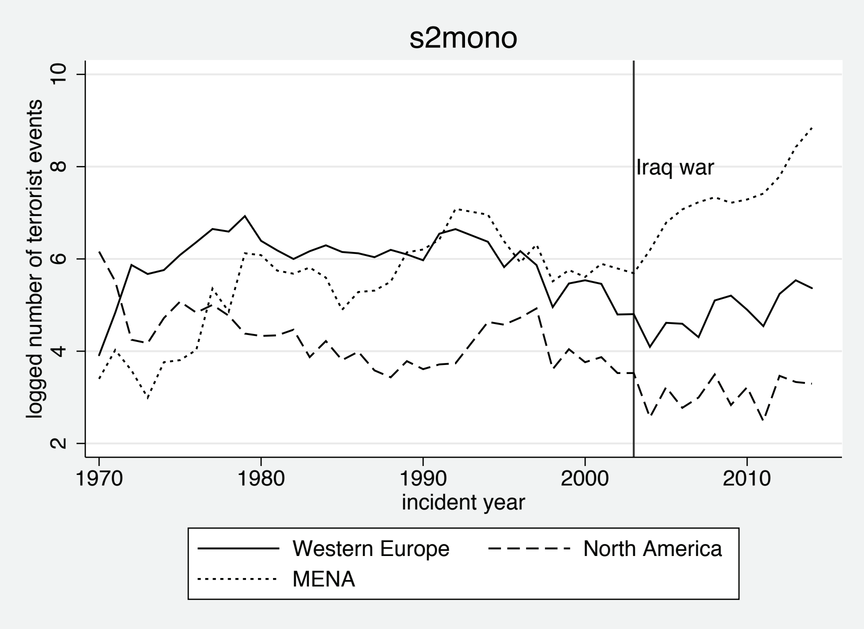

Figure 2. Limitations of `s2mono`: lineplot

The graphics in figures 1 and 2 are examples of the major concerns I want to raise. The following aspects can be improved:

1. The y-axis labels are vertical, making reading the axis inconvenient. Some scholars suggest that vertical labeling saves space. However, it also makes reading the y axis more cumbersome. Thus, many Stata users might rather consume more space to simplify reading the y axis.[1]

2. The background is unnecessarily tinted gray, which also results in the plot area being surrounded by a frame.

3. Markers and lines are often too thick. Thus, points in the scatterplot can end up being indistinguishable—not because the location of the points are equal but

[1] Some users might suggest that vertical labeling makes sense when users attempt to bring together several graphics into a single figure, which is valid, but rare. Thus, I suggest that using horizontal labels is a status quo that most users will prefer.

because the markers overlap because of their size. This especially pertains to scatterplots. In figure 1, several data points overlap because of their size, making it difficult for the reader to judge the location of each data point (see Robbins [2005, 63 and 165]; Cleveland [1994, 24–26]). This issue is worsened by an increase of subgroups in a plot.

4. For the same reasons, the symbols in the legend could be smaller; otherwise, they occasionally overlap.

5. Gridlines are drawn only parallel to the x axis. Gridlines are more important than intuition might suggest; in fact, human beings are limited in their selective attention. Our judgment of a position of a point or line on one dimension might lead us to misjudge its position on a different dimension (Cleveland 1994). Gridlines can help users overcome this issue and are also prominently used in Robbins (2005) and Tufte (2006). They are especially helpful if we want to judge whether two lines correlate (are parallel). Thus, it seems more reasonable that gridlines be drawn parallel to both the x and y axes (Robbins 2012). As suggested in Tufte (2001), gridlines could use a less dominant shape, such as thin dots, so that they will not distract readers.[2]

6. Overall, most colors used in Stata's default graphic schemes are not distinguishable for colorblind people. Smoothing the difference and fading the colors slightly should enhance legibility.

7. There is no obvious need to put a frame around the legend.

8. Legends should be placed closer to the data information to ease interpretation.[3] Placing legends beneath the figure forces the reader to pay attention to the area beneath the axis (Cleveland 1994, 100).

9. Axes use the same color as anything plotted in black. The axis should be more in the background and distinguishable from data points. However, this is a very pedantic point, and users might disagree on its usefulness. I will return to the colors of the axes later in this article.

In summary, a lot of chartjunk could be removed from Stata figures. As suggested earlier, Stata users might want to proceed and solve some of these concerns via syntax or use `brewscheme` to define their own designs. However, using graphic schemes that already address the points raised above might be a more efficient and attractive option for most users.

[2] Some users might disagree on using gridlines parallel to the x axis. These might be useful if we plot our data using a scatterplot, but they are less useful once we plot time-series data with a lineplot. To address this concern, I provide code in the final section to exclude the gridlines in case we would want to plot, for example, time-series data.

[3] Note that the legend should also rarely be placed within the plot area. It is only a reasonable approach if the user can ensure that such a placement does not distract from the information of the data. Therefore, Cleveland (1994, 100) cautions to "[a]void putting notes, keys, and markers in the data region".

3 plotplain and plottig

plotplain and plottig provide a graphical design that aims to address the key points discussed above. Figures 3 and 4 plot the exact same data and use the same Stata code outlined above but with the plotplain scheme.[4] Figure 5 does the same with the plottig scheme. The former belongs to the s1 scheme family, the latter to the s2 family.

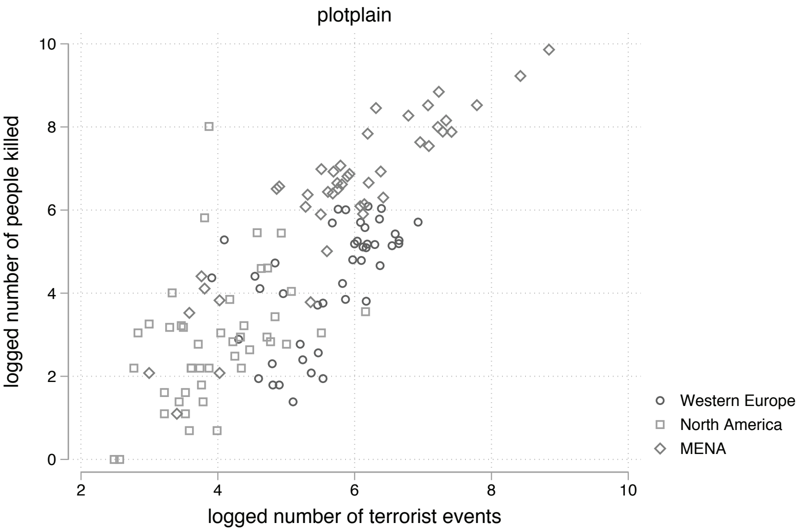

Figure 3. Advantages of plotplain: scatterplot

[4] Consult the author's webpage or try the examples yourself to see the plotplain scheme used to display the figures in color.

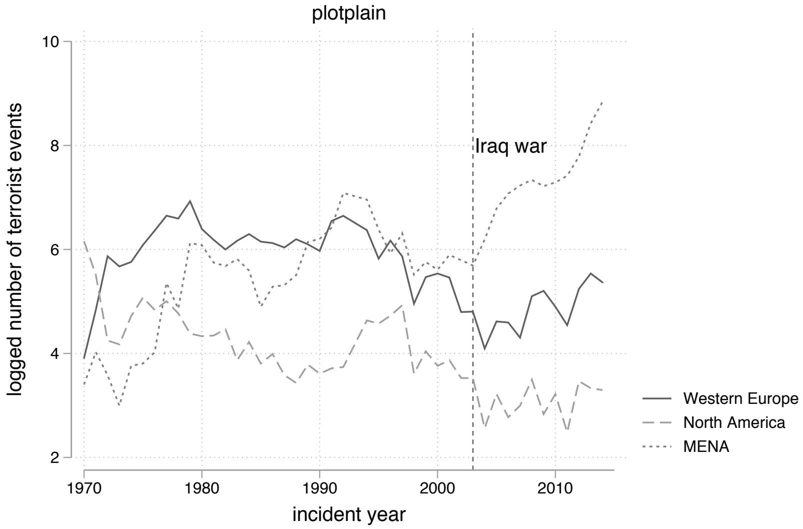

Figure 4. Advantages of `plotplain`: lineplot

These graphic schemes contain the following improvements:

1. Labels are all horizontal.
2. Markers and lines are thinner. One can clearly tell where data points overlap and where they do not overlap.
3. Legends are placed to the right of the plot area, rather than beneath the plot area.
4. Symbols in the legend are smaller.
5. Frames demarcating legends are omitted.
6. There is no background color.
7. Axes have a gray color scheme, putting them more into the background of the plot.
8. Gridlines exist for both axes and are marked by thin gray dots.
9. Plot areas are not surrounded by any frames.

Both schemes are available in colorblind-friendly versions. While `plotplain` relies only on black-white shading and different shapes of markers and lines, `plotplainblind` adds seven colors to the scheme—all distinguishable for colorblind people. Another advantage of the `plotplainblind` scheme is that colors are drawn only if users specify more than two subgroups to be plotted. Single graphs are still black, and graphs running across two subgroups are differentiated by black and gray. Any number of subgroups beyond two are then plotted in seven colors that can be distinguished by colorblind people.

In contrast to `plotplain`, `plottig` relies on the idea that using a colored background within the plot area contributes to the legibility of figures.

As can be seen in figure 5, this decision enhances the quality of the figures extensively. Markers and especially colors show a higher legibility than `plotplain` figures. Thus, the colored plot area is not chartjunk but emphasizes the differences between markers and colors. Furthermore, `plottig` provides a new 14-color palette, while `plottigblind` uses the exact same colors used in `plotplainblind`.

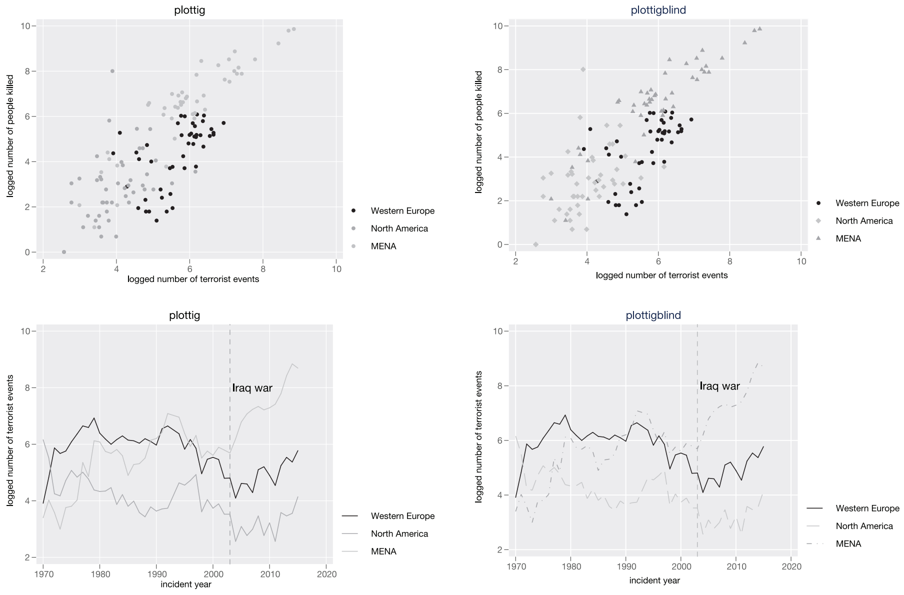

Figure 5. `plottig` and `plottigblind`

Like the `lean` scheme, both schemes also assign more space to the plot area than the default Stata schemes (Juul 2003, 300) because of the adjusted placement of the legend. Again, this ensures that Stata users will maximize the information on their data in figures and minimize the presence and protrusion of chartjunk.

Figure 6 plots these colors using a bar plot. Once users install the schemes, these colors will be available in Stata. Colors can be used in syntax commands just like any other color (for example, `lcolor(vermillion)`), and they might be used in the graph editor by selecting the relevant color from the color palette. The command names of all colors are shown as labels in figure 6.

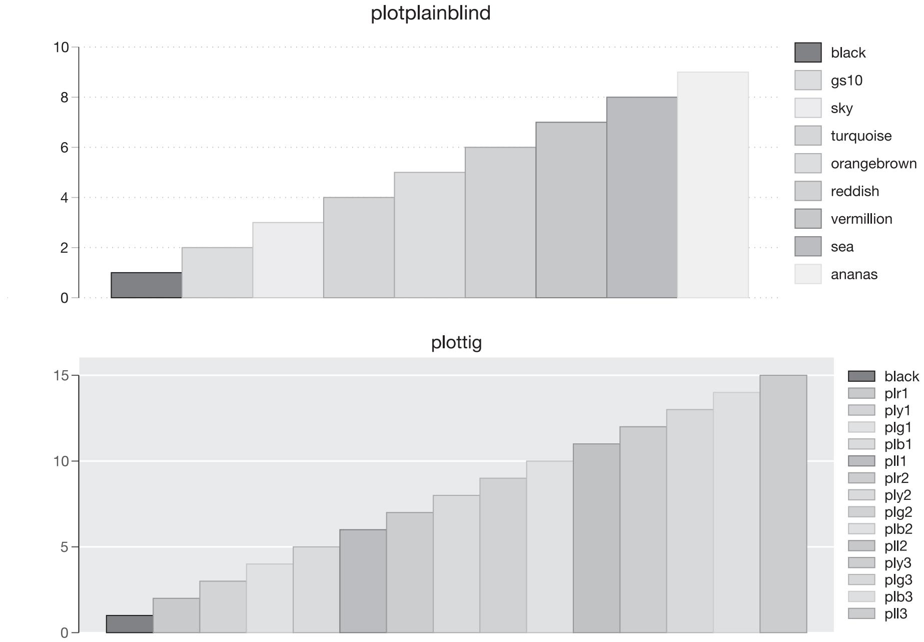

Figure 6. The colors of `plotplainblind` and `plottig`

To use the graphic schemes presented here, users might want to download the files from the *Stata Journal* website and draw a line graph using the color "vermillion" by typing

```
. set scheme plottig, permanently
. twoway (line x y, lcolor(vermillion))
```

3.1 Customizing schemes

A major concern of some advanced users is the adaption or creation of their own graphic schemes like the ones in this article. Besides some information provided in Mitchell (2012, 379–382), there is not much guidance on how to design a graphic scheme from scratch or how to amend an existing one.

The programming language of graphic schemes does not follow the standard Stata syntax. I present some of the key characteristics of a scheme below. First, you can create schemes with a simple text editor (`.txt` file). However, users have to ensure that the suffix of their scheme files is `.scheme` instead of `.txt`. Commands within a scheme always follow a three-step logic:

```
characteristic object option
```

A general characteristic (such as the color) of a respective object (such as the background) is assigned a specific option (such as `white`):

```
color background  white
```

The example above is taken from the `plottig` scheme and assigns a white background to the scheme. Thus, if we wanted to change the background color, we would just need to exchange `white` in the code above with, for example, `black`. Below is an overview of the most important options in the `plottig` scheme:

- *clockdir* controls the position of an object.
- *color* controls the color of an object.
- *linepattern* controls the style of a line to be drawn in a figure (for example, `shortdash`).
- *linestyle* controls the style of a line to be drawn outside a figure (for example, `shortdash` for an axis).
- *linewidth* controls the size of a line.
- *gsize* controls the size of an object.
- *margin* controls the margins between objects.
- *numticks_g* controls the number of ticks in an axis.
- *symbol* controls the style of a symbol (for example, `circle` for scatterplots).
- *symbolsize* controls the size of a symbol.
- *tickstyle* controls the style of ticks (for example, ticks for labels).
- `yesno` options control certain options by simply adding `yes` or `no` after a command (for example, `yesno draw_major_grid yes` controls gridlines for the `plottig` scheme).

In general, I would recommend that users interested in creating their own graphic scheme get started with an existing graphic scheme. This way, users can see the actual code and output of the scheme before adapting it to their liking. For instance, some users might entirely disagree with my suggestion to use gray for the axes in the `plotplain` scheme. To change the colors of the axes, users can simply copy and paste the original `plotplain` scheme file in their Stata applications directory. Then they can open a copy of the scheme and search for the following command:

```
color axisline     gs10
```

Then users might want to change `gs10` into `black` by deleting the `gs10` and writing instead

```
color axisline     black
```

After users save the newly created scheme file and reset the general scheme option to use their own scheme, all axes should appear in black. Finally, users might still want to adapt the graphic schemes presented here by using code. For instance, users might want to fade out gridlines entirely. Even though I do not recommend doing this in all applications, there are reasons for excluding them in some applications. While I regard them as a must in scatterplots to ensure readers' orientations and especially to ensure a simple comparison between the positions of data points, gridlines can be less useful if a single time trend is plotted across time. Thus, if users want to use the `plottig` scheme and prohibit the appearance of grids, they might want to use the following command:

```
. lpoly nkill iyear if region==12,
> title("Polynomial fit of terrorist casualties in Western Europe")
> note("") xlabel(, nogrid) ci legend(off)
```

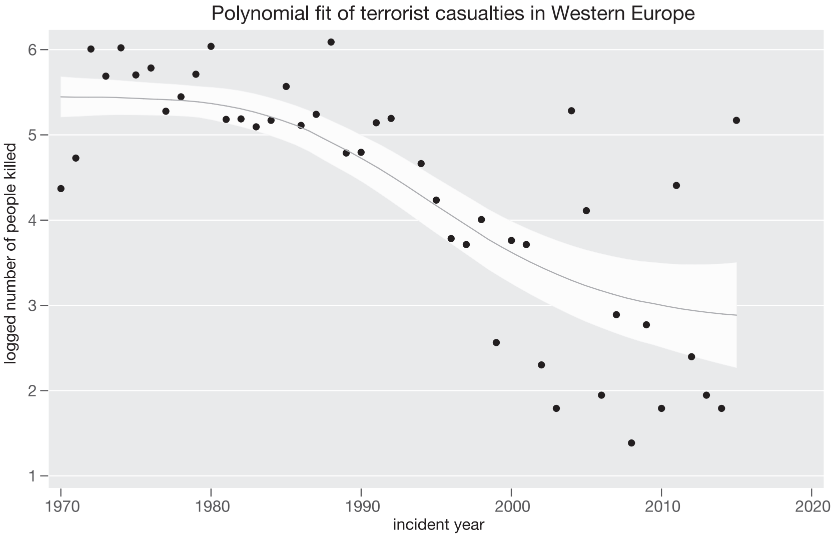

Figure 7. Getting rid of grids

As shown in figure 7, by specifying **xlabel(, nogrid)**, users omit the gridlines parallel to the y axis (more detail on gridlines can be found in Cox [2009]).

Of course, this short application just scratches the surface of possibilities. As seen in the graphic schemes introduced here, many options and changes can be implemented by adapting already existing graphic schemes.

4 Conclusion

In this article, I presented two new graphic schemes, `plotplain` and `plottig`, to the Stata user community. In doing so, I attempted not only to provide users with a set of new alternatives to produce figures in Stata but also to address some common concerns about Stata's default graphic schemes (for example, horizontal labeling, background tinting, size of markers, placement and style of legends, missing gridlines, and colors not distinguishable for colorblind people). Finally, I provided users with an introduction on how to design graphic schemes for Stata.

As a final disclaimer, I do not believe that the graphic schemes provided here are perfect for each user and each application. Some of the concerns raised are also subject to individual taste, while others are not. I hope that this introduction motivates users to design their own graphic schemes and to share them with the Stata user community.

5 References

Buchanan, W. R. Forthcoming. The brewscheme toolkit for data visualization in Stata. *Stata Journal*.

Cleveland, W. S. 1994. *The Elements of Graphing Data*. Rev. ed. Summit, NJ: Hobart.

Cox, N. J. 2009. Stata tip 82: Grounds for grids on graphs. *Stata Journal* 9: 648–651.

Harrower, M., and C. A. Brewer. 2003. ColorBrewer.org: An online tool for selecting colour schemes for maps. *Cartographic Journal* 40: 27–37.

Juul, S. 2003. Lean mainstream schemes for Stata 8 graphics. *Stata Journal* 3: 295–301.

Mitchell, M. N. 2012. *A Visual Guide to Stata Graphics*. 3rd ed. College Station, TX: Stata Press.

Robbins, N. B. 2005. *Creating More Effective Graphs*. Hoboken, NJ: Wiley.

———. 2012. Are grid lines useful or chartjunk? http://www.forbes.com/sites/naomirobbins/2012/02/22/are-grid-lines-useful-or-chartjunk/.

Tufte, E. R. 2001. *The Visual Display of Quantitative Information*. 2nd ed. Cheshire, CT: Graphics Press.

———. 2006. *Beautiful Evidence*. Cheshire, CT: Graphics Press.

Wickham, H. 2009. *ggplot2: Elegant Graphics for Data Analysis*. New York: Springer.

About the author

Daniel Bischof is a senior researcher (Oberassistent) at the University of Zurich, Switzerland. His main research interest is comparative politics. He studies party competition, government formation, political representation, public opinion formation, and the causes and consequences of political upheaval.