Line/Scatter Examples

Dave Lorenz

September 9, 2013

These examples demonstrate the basic high-level line/scatter plot functions in the USGSwsGraphs package. All of the examples use randomly generated sets of data. **NOTE:** to use any of these functions, you must first call a function to set up the graphics environment like `setPage` or `setPDF`, but these are not included here to use the graphics tools in `Sweave`.

```
> # Load the USGSwsGraphs package
> library(USGSwsGraphs)
> # Generate random samples for the examples.
> set.seed(2576)
> X <- runif(33)
> Y <- runif(33)
> Z <- rep(c("A", "B", "C"), 11)
> XD <- seq(as.Date("2010-01-05"), as.Date("2010-12-15"), length.out=33)
```

1 Scatter Plot

The `xyPlot` function plots paired x- and y-coordinate data. As of version 0.7, there are methods for factor and numeric x-coordinate data and numeric y-coordinate data. This example plots only numeric data.

```
> # setSweave is a specialized function that sets up the graphics page for
> # Sweave scripts. It should be replaced by a call to setPage or setPDF
> # in a regular script.
> setSweave("lsplot01", 6 ,6)
> xyPlot(X, Y, Plot=list(color="darkblue"))
> # Required call to close PDF output graphics
> graphics.off()
```

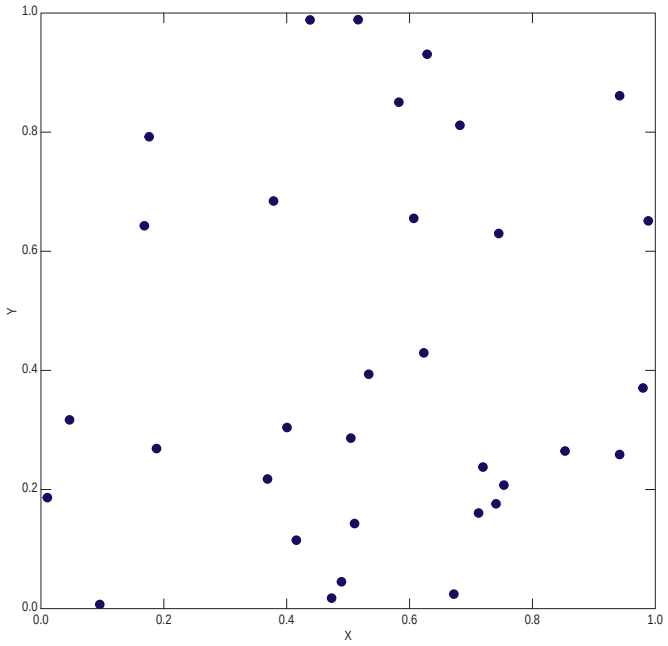

Figure 1. A simple scatter plot

2 Date/Time Plot

The `timePlot` function plots paired x- and y-coordinate data. As of version 0.7, there are methods for Date, POSIXt, numeric, and integer x-coordinate data and numeric y-coordinate data. This example plots a sequence of numeric data along a sequence of Date data.

```
> # setSweave is a specialized function that sets up the graphics page for
> # Sweave scripts. It should be replaced by a call to setPage or setPDF
> # in a regular script.
> setSweave("lsplot02", 6 ,6)
> timePlot(XD, Y, Plot=list(color="darkblue"))
> # Required call to close PDF output graphics
> graphics.off()
```

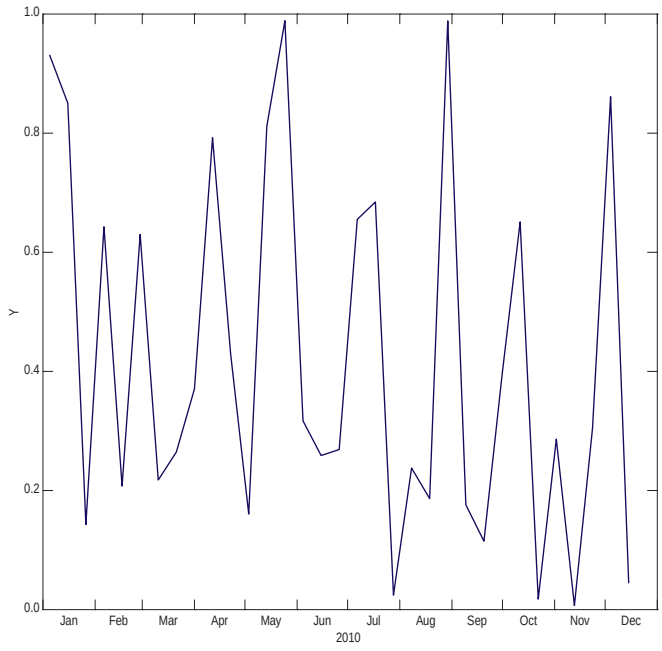

Figure 2. A time-series plot

3 Color Plot

The `colorPlot` function plots paired x- and y-coordinate data, with symbol color controlled by another variable. As of version 0.7, there is only one method, for numeric x and y data. The color variable can be type character or factor, numeric, or integer. If the type is character or factor, then the values can be color names (see the documentation for colors), or group names that are converted to bright colors. If the type is numeric, then the colors are created using the color, groups, and ramp components of the Plot argument. If the type is integer, the the color component of Plot should be "Index" to set the color to the index color of the palette (see the docuementation for palette).

```
> # setSweave is a specialized function that sets up the graphics page for
> # Sweave scripts. It should be replaced by a call to setPage or setPDF
> # in a regular script.
> setSweave("lsplot03", 6 ,6)
> # Accept the default colors for groups.
> AA.pl <- colorPlot(X, Y, color=Z)
> addExplanation(AA.pl, where="ul", title="")
> # Required call to close PDF output graphics
> graphics.off()
```

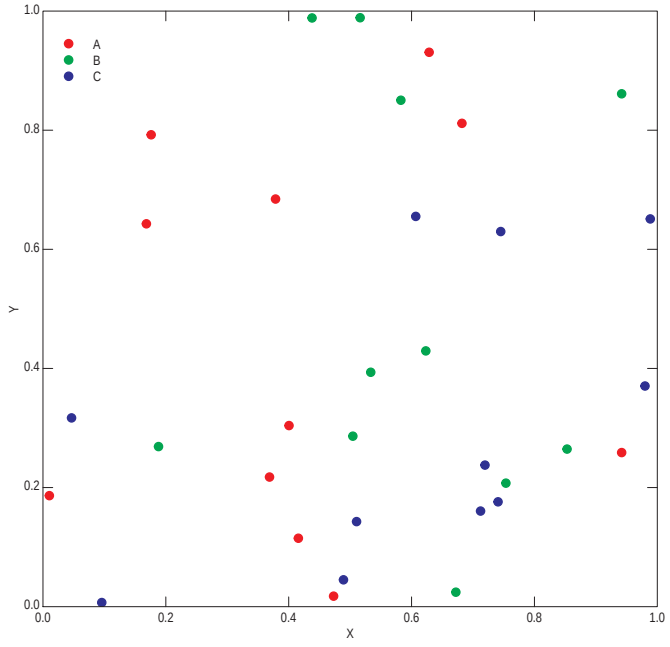

Figure 3. A color plot