

Beyond R's basic graphics system: lattice and ggplot2

Bernd Weiss
Research Institute for Sociology
University of Cologne
Germany

21/01/2011



Outline

Some preliminary work

The lattice package

The ggplot2 package

Acknowledgment, license and downloads

- ▶ This work was supported by a fellowship within the Postdoc-Programme of the German Academic Exchange Service (DAAD)(Grant D/10/43517).
- ▶ My presentation was created using Emacs' *org-mode* and *Babel: active code in Org-mode*. *Babel* is developed and maintained by Eric Schulte and Dan Davison who were extremely helpful in answering my questions or fixing bugs.
- ▶ Licensed under a Creative Commons [Attribution-NonCommercial-ShareAlike 3.0 Germany](#) license.
- ▶ Slides, dataset and R code can be downloaded from my github page: [xxx](#) (see "Downloads" button on the right-hand side).

Topic

Some preliminary work

The lattice package

The ggplot2 package

Loading packages and creating the data

```
1 library(lattice)
2 library(ggplot2)
3
4 ## Regression model:  $y = 1.2 + 0.3 \cdot x_1 + 0.4 \cdot x_2 + 0.9 \cdot x_1 \cdot x_2 + e$ 
5 set.seed(1)
6 x1 <- round(runif(1000, min = 1, max = 10), digits = 0)
7 x2 <- round(runif(1000, min = 1, max = 4), digits = 0)
8
9 y = 1.2 + 0.3*x1 + 0.4*x2 + 0.9*x1*x2 + rnorm(1000, 0, 1)
10 df <- data.frame(y = y, x1 = x1, x2 = x2,
11                 x2f = factor(x2, labels = c("a", "b", "c", "d")))
12
13 lm(y ~ x1 + x2 + x1*x2, data = df)
```

Call:

```
lm(formula = y ~ x1 + x2 + x1 * x2, data = df)
```

Coefficients:

(Intercept)	x1	x2	x1:x2
1.3046	0.2554	0.3738	0.9140

Topic

Some preliminary work

The lattice package

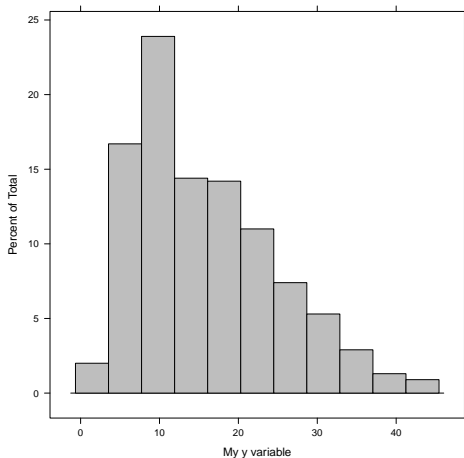
The ggplot2 package

A basic overview about the lattice package

- ▶ Needs to be loaded via `library(lattice)`
- ▶ Most lattice function use the formula interface (e.g., $y \sim x$).
- ▶ One strength on `lattice` is the ability to create multipanel plots (trellis graphs). Use the “conditional on” symbol `|` to create a multipanel plot (e.g., $y \sim x \mid g$).
- ▶ Sometimes, original data need to be prepared (e.g., for barcharts)
- ▶ Web resources:
 - ▶ [Website for “Lattice: Multivariate Data Visualization with R - Figures and Code” by Deepayan Sarkar](#)
 - ▶ [Using Lattice Graphics in R](#)
 - ▶ [An Introduction to R by Deepayan Sarkar](#)

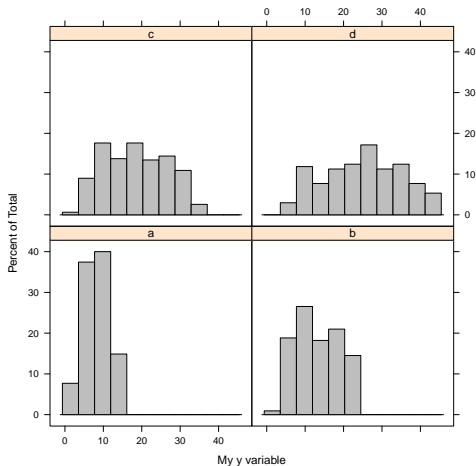
Histogram

```
1 histogram(~ y, data = df, xlab = "My y variable",  
2           col = "gray")
```



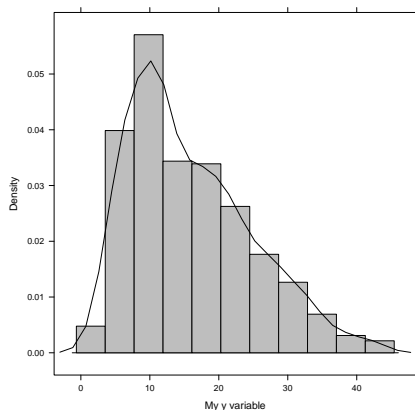
Histogram conditional on x2f

```
1 histogram(~ y | x2f, data = df,  
2           xlab = "My y variable", col = "gray")
```



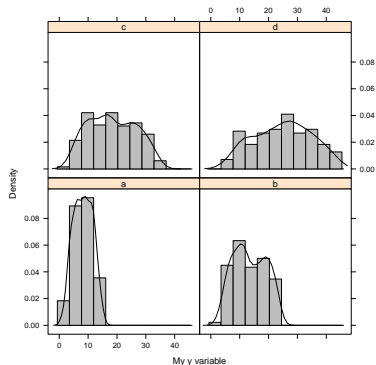
Histogram with superimposed density plot

```
1 histogram(~ y, data = df, xlab = "My y variable",  
2           col = "gray", type = "density",  
3           panel = function(...){  
4               panel.histogram(...);  
5               panel.densityplot(..., col.line = "black")  
6           }  
7 )
```



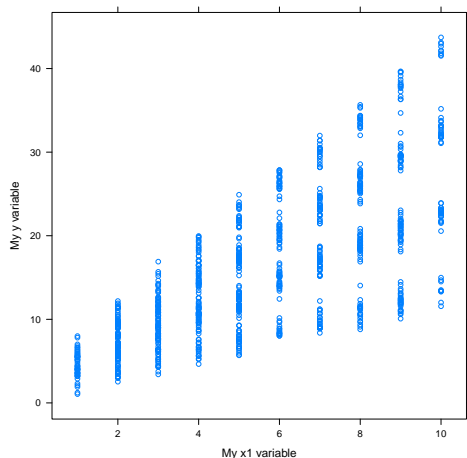
Histogram with superimposed density plot and conditional on x2f

```
1 histogram(~ y | x2f, data = df, xlab = "My y variable",
2           col = "gray", type = "density",
3           panel = function(...){
4             panel.histogram(...);
5             panel.densityplot(...,
6                               col.line = "black")
7           }
8 )
```



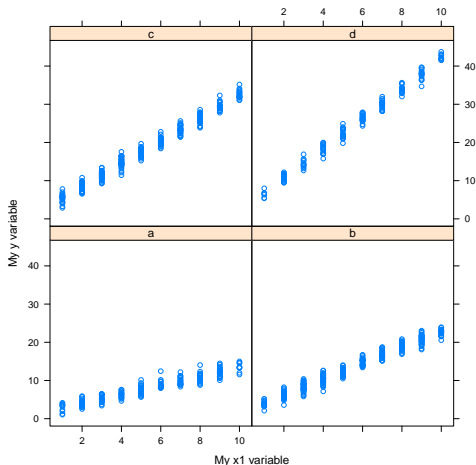
Scatter plot

```
1 xyplot(y ~ x1, data = df,  
2       xlab = "My x1 variable", ylab = "My y variable")
```



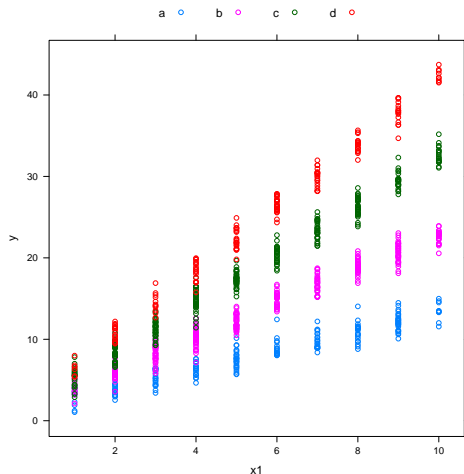
Multipanel scatter plot by x2f

```
1 xyplot(y ~ x1 | x2f, data = df,  
2         xlab = "My x1 variable", ylab = "My y variable")
```



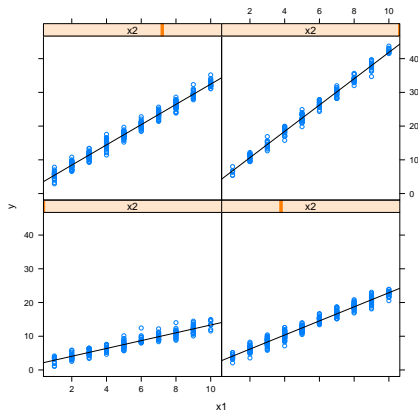
Scatter plot by grouping variable x2

```
1 xyplot(y ~ x1, groups = x2f, data = df,  
2       key = simpleKey(text = levels(df$x2f), columns = 4))
```



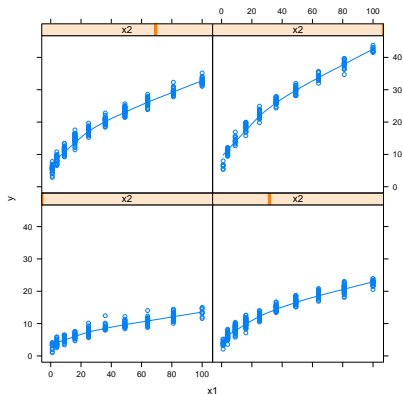
Scatter plot and linear regression line by grouping variable x2

```
1  xyplot(y ~ x1 | x2, data = df,  
2      panel = function(y, x1, ...){  
3      panel.xyplot(x1, y);  
4      panel.lmline(x1, y)  
5      })
```



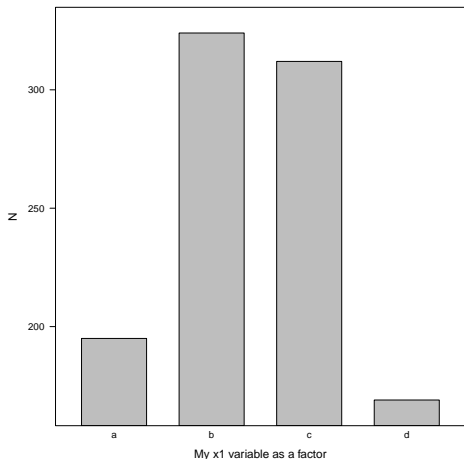
Scatter plot and LOESS line by grouping variable x2

```
1  xyplot(y ~ x1 | x2,  
2        data = data.frame(df[, -2], x1 = df$x1^2),  
3        panel = function(y, x1, ...){  
4          panel.xyplot(x1, y);  
5          panel.loess(x1, y)  
6        })
```



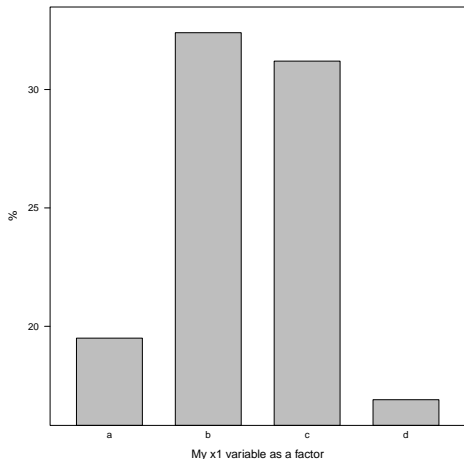
Bar chart (absolute frequencies)

```
1 barchart(Freq ~ Var1, data = data.frame(table(df$x2f)),  
2         xlab = "My x1 variable as a factor", ylab = "N",  
3         col = "gray")
```



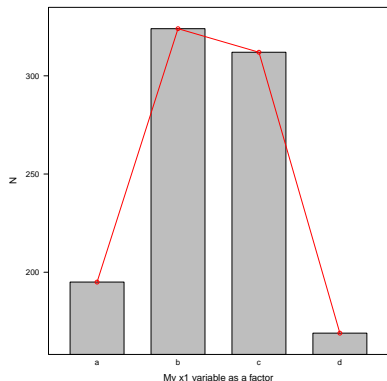
Bar chart (percentages)

```
1 barchart(Freq ~ Var1,  
2           data = data.frame(100 * prop.table(table(df$x2f))),  
3           xlab = "My x1 variable as a factor", ylab = "%",  
4           col = "gray")
```



Bar chart with superimposed line plot

```
1 barchart(Freq ~ Var1, data = data.frame(table(df$x2f)),  
2         xlab = "My x1 variable as a factor", ylab = "N",  
3         col = "gray",  
4         panel = function(...){  
5             panel.barchart(...);  
6             panel.xyplot(..., type = "b", col.symbol = "red",  
7                           col.line = "red");  
8         })
```



Topic

Some preliminary work

The lattice package

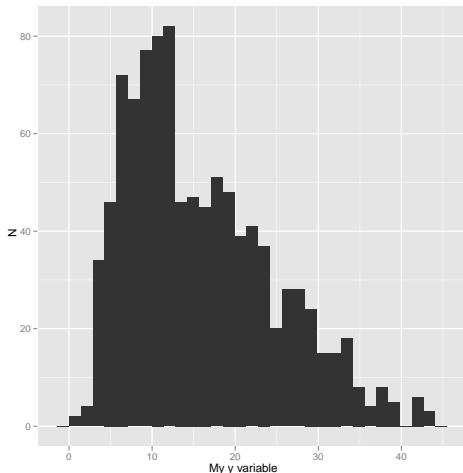
The ggplot2 package

A basic overview about the ggplot2 package

- ▶ Needs to be loaded via `library(ggplot2)`
- ▶ Is based on The Grammar of Graphics by Leland Wilkinson
- ▶ Sometimes difficult to “tweak” plots which do not follow the GoG (and Hadley Wickham’s implementation of the GoG)
- ▶ Often, original data need to be modified (e.g., aggregated for barcharts)
- ▶ Web resources:
 - ▶ [Hadley Wickham’s website for ggplot2](#) (this website is simply awesome; he also has written a ggplot2 related book)
 - ▶ [Wiki for ggplot2: Elegant graphics for data analysis](#) (ultimate resource when it comes to fine tuning)
 - ▶ [The blog Learning R offers a lot of examples of how ggplot2 works](#)
 - ▶ Visualizing Data with R and ggplot2 (video w/ slides) (website: www.drewconway.com/zia/?p=1637)

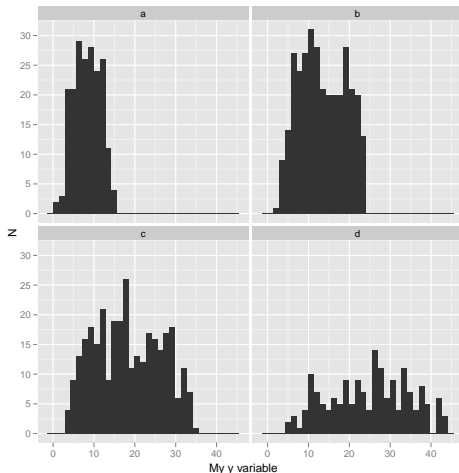
Histogram

```
1 ggplot(aes(x = y), data = df) + geom_histogram() +  
2   xlab("My y variable") + ylab("N")
```



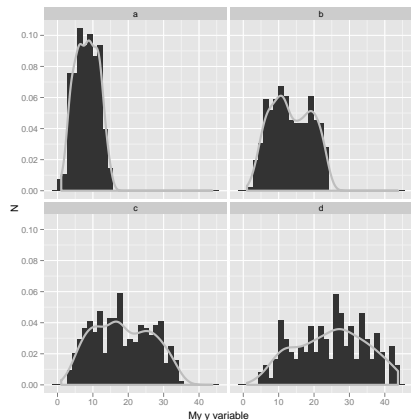
Histogram conditional on x2f

```
1 ggplot(aes(x = y), data = df) + geom_histogram() +  
2   xlab("My y variable") + ylab("N") + facet_wrap(~x2f)
```



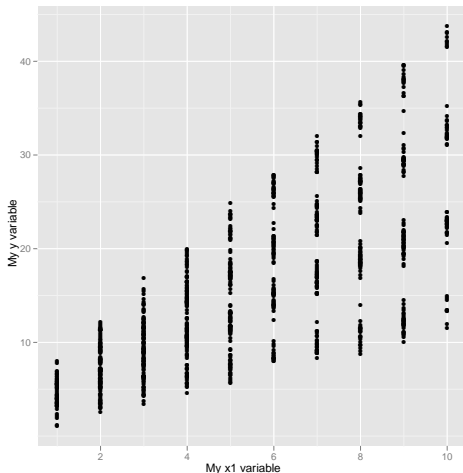
Histogram with superimposed density plot and conditional on x2f

```
1 ggplot(aes(x = y), data = df) +  
2   geom_histogram(aes(y = ..density..)) +  
3   geom_density(colour = "grey", size = 1.2) +  
4   xlab("My y variable") + ylab("N") +  
5   facet_wrap(~x2f)
```



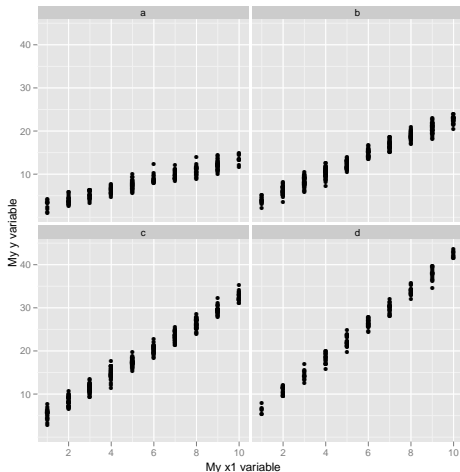
Scatter plot

```
1 ggplot(aes(x = x1, y = y), data = df) + geom_point() +  
2   xlab("My x1 variable") + ylab("My y variable")
```



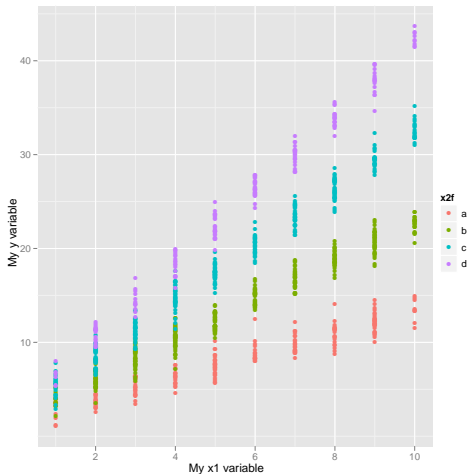
Multipanel scatter plot by x2f

```
1 ggplot(aes(x = x1, y = y), data = df) + geom_point() +  
2   xlab("My x1 variable") + ylab("My y variable") +  
3   facet_wrap(~ x2f)
```



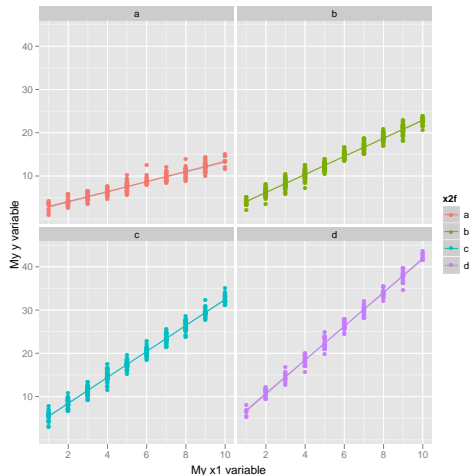
Scatter plot by grouping variable x2f

```
1 ggplot(aes(x = x1, y = y, colour = x2f), data = df) +  
2   geom_point() + xlab("My x1 variable") +  
3   ylab("My y variable")
```



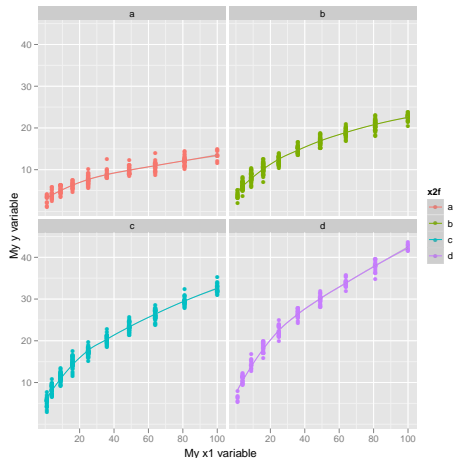
Scatter plot and linear regression line by x2f

```
1 ggplot(aes(x = x1, y = y, colour = x2f), data = df) +  
2   geom_point() + xlab("My x1 variable") +  
3   ylab("My y variable") + facet_wrap(~x2f) +  
4   geom_smooth(method = "lm")
```



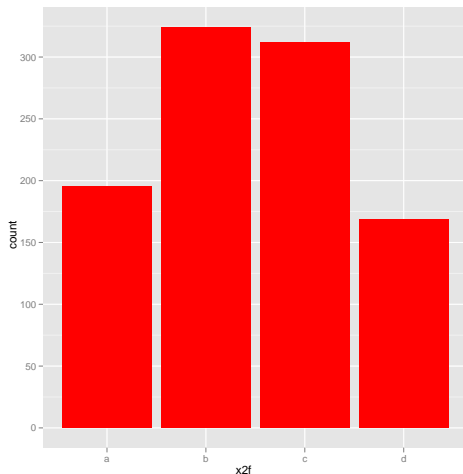
Scatter plot and LOESS line by x2

```
1 ggplot(aes(x = x1, y = y, colour = x2f),  
2       data = data.frame(df[, -2], x1 = df$x1^2)) +  
3       geom_point() + xlab("My x1 variable") +  
4       ylab("My y variable") + facet_wrap(~x2f) +  
5       geom_smooth(method = "loess")
```



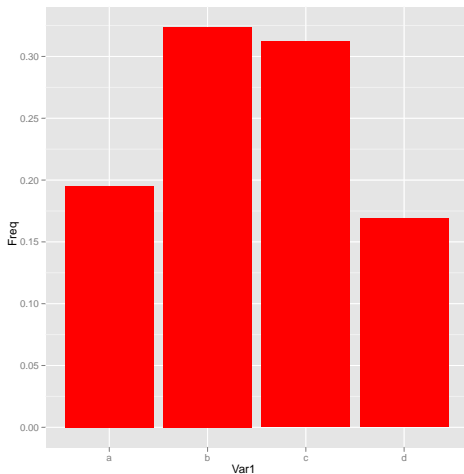
Bar chart (absolute frequencies)

```
1 ggplot(aes(x = x2f), data = df) + geom_bar(fill = "red")
```



Bar chart (percentages)

```
1 tmp <- data.frame(prop.table(table(df$x2f)))  
2 ggplot(aes(x = Var1, y = Freq), data = tmp) +  
3   geom_bar(fill = "red")
```



Box plot, jittered scatterplot, LOESS line (slightly useless...)

```
1 ggplot(aes(x = x2f, y = y), data = df) +  
2   geom_jitter() + geom_boxplot(alpha = 0.8) +  
3   stat_smooth(aes(x = as.numeric(x2f), y = y)),  
4               data = df, method = "loess",  
5               level = 0.90) +  
6   geom_hline(yintercept = mean(df$y),  
7             col = "green", size = 1.2)
```

