An introduction to the xkcd package

Emilio Torres Manzanera
(torres@uniovi.es)

May 2013

Contents

1 Installing xkcd

The xkcd homepage is located at `http://xkcd.r-forge.r-project.org`. From within R, you can install the latest version of xkcd by typing `install.packages(picante, dependencies=TRUE)`. Typing `help(functionName)` will display documentation for any function in the package.

Once the package has been installed, it can be loaded by typing:

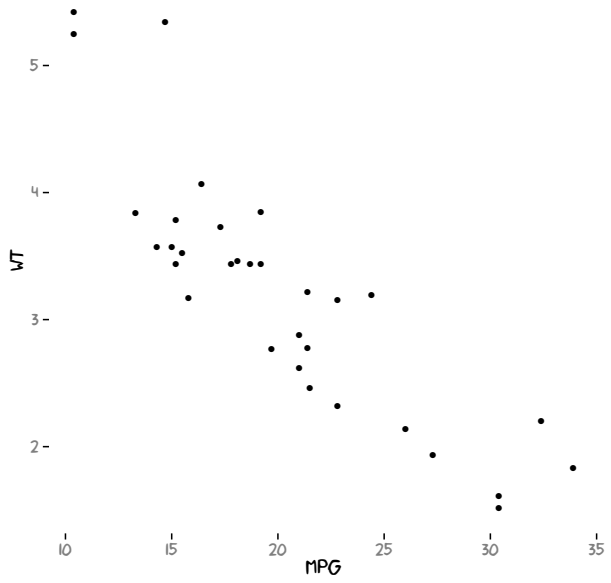
```
> library(xkcd)
```

Check if the fonts are loaded:

```
> fonts()
```

```
[1] "xkcd"
```

```
> ggplot() + geom_point(aes(mpg, wt), data=mtcars) + theme_xkcd()
```



Installing the xkcd fonts

If the xkcd fonts are not installed in the system, you must install them. See `extrafont` for further details `https://github.com/wch/extrafont`:

- Option I: Borrowed from `http://fibosworld.wordpress.com/2013/02/17/change-fonts-in-ggplot2-and-create-`

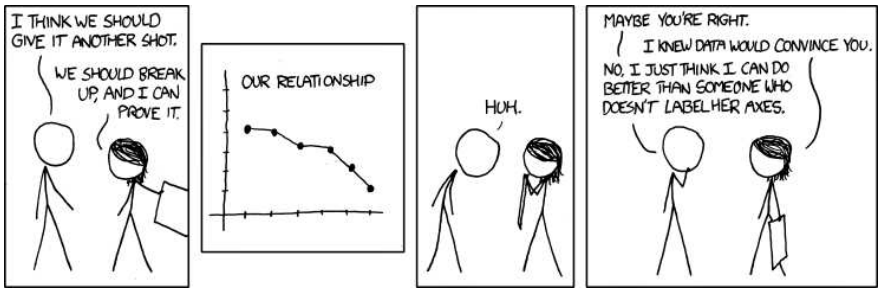
```
> ## Borrowed from
> ## fibosworld 2013. Change fonts in ggplot2, and create xkcd style graphs
> library(extrafont)
> if(! "xkcd" %in% fonts()) {

   xkcdFontURL <- "http://simonsoftware.se/other/xkcd.ttf"
   download.file(xkcdFontURL,dest="xkcd.ttf")
   font_import(".")   ## because we downloaded to working directory
   loadfonts()
 }
```

- Option II. The Option I does not work for me (on a Linux machine). I installed the fonts in this way:

```
> download.file("http://simonsoftware.se/other/xkcd.ttf", dest="xkcd.ttf")
> system("mkdir ~/.fonts")
> system("cp xkcd.tff -t ~/.fonts")
> library(extrafont)
> font_import()
> loadfonts()
```

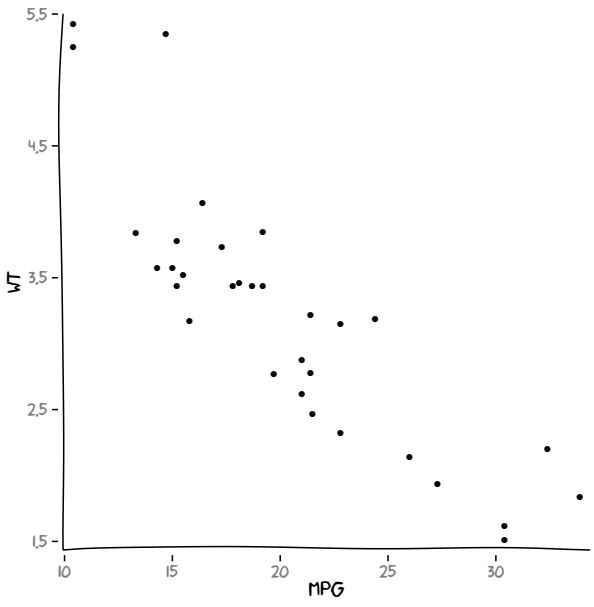
2 Axis

Man: No, I just think I can do better than someone who doesn't label her axes. Title text: And if you labeled your axes, I could tell you exactly how MUCH better.



http://xkcd.com/833/ http://imgs.xkcd.com/comics/convincing.png

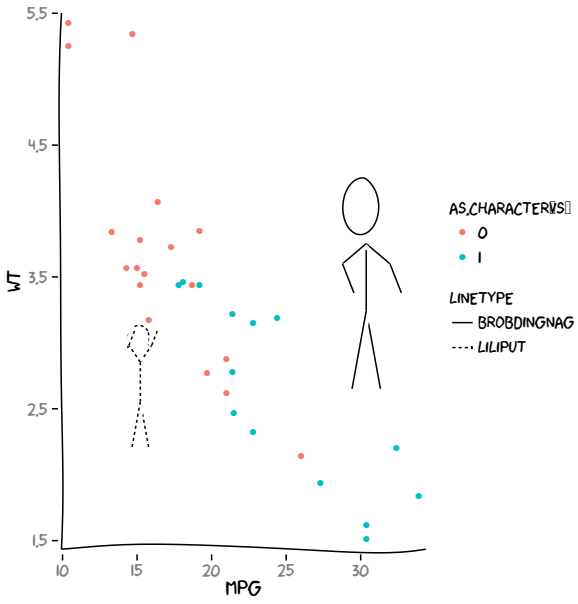
```
> xrange <- range(mtcars$mpg)
> yrange <- range(mtcars$wt)
> set.seed(123) # for reproducibility
> p <- ggplot() + geom_point(aes(mpg, wt), data=mtcars) + xkcdaxis(xrange,yrange)
> p
```



3 Cartoon characters

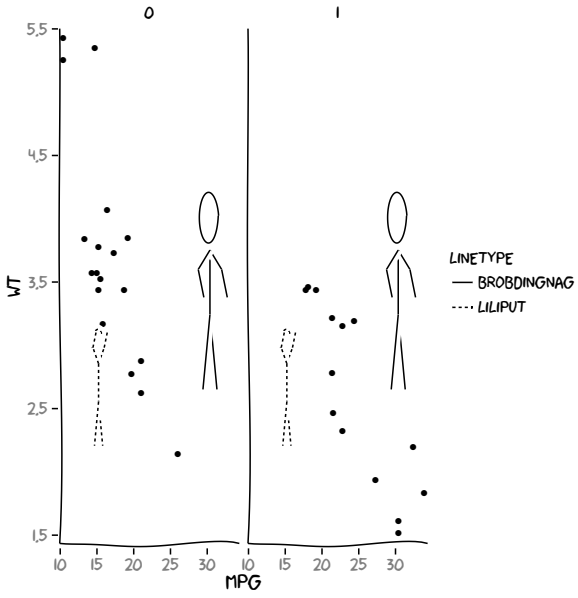
To include cartoon characters in the graph, use the `xkcdman` function.

```
> ratioxy <- diff(xrange)/diff(yrange)
> mapping <- aes(x,  y,
                 scale,
                 ratioxy,
                 angleofspine ,
                 anglerighthumerus,
                 anglelefthumerus,
                 anglerightradius,
                 angleleftradius,
                 anglerightleg,
                 angleleftleg,
                 angleofneck,
                 linetype=city)
> dataman <- data.frame(x= c(15,30), y=c(3, 4),
                        scale = c(0.3,0.51) ,
                        ratioxy = ratioxy,
                        angleofspine =  -pi/2  ,
                        anglerighthumerus = c(pi/4, -pi/6),
                        anglelefthumerus = c(pi/2 + pi/4, pi +pi/6),
                        anglerightradius = c(pi/3, -pi/3),
                        angleleftradius = c(pi/3, -pi/3),
                        anglerightleg = 3*pi/2  - pi / 12,
                        angleleftleg = 3*pi/2  + pi / 12 ,
                        angleofneck = runif(1, 3*pi/2-pi/10, 3*pi/2+pi/10),
                        city=c("Liliput","Brobdingnag"))
> q <- ggplot() + geom_point(aes(mpg, wt, colour=as.character(vs)), data=mtcars) +
   xkcdaxis(xrange,yrange) + xkcdman(mapping, dataman)
> q
```



3.1 Facets

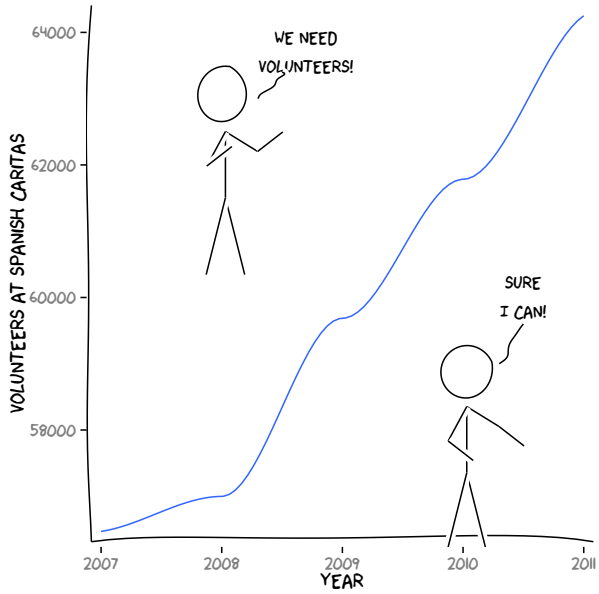
```
> ggplot() + geom_point(aes(mpg, wt), data=mtcars) +
   xkcdaxis(xrange,yrange) + xkcdman(mapping, dataman) +
  facet_grid(.~vs)
```



4 Volunteers at Cáritas Spain

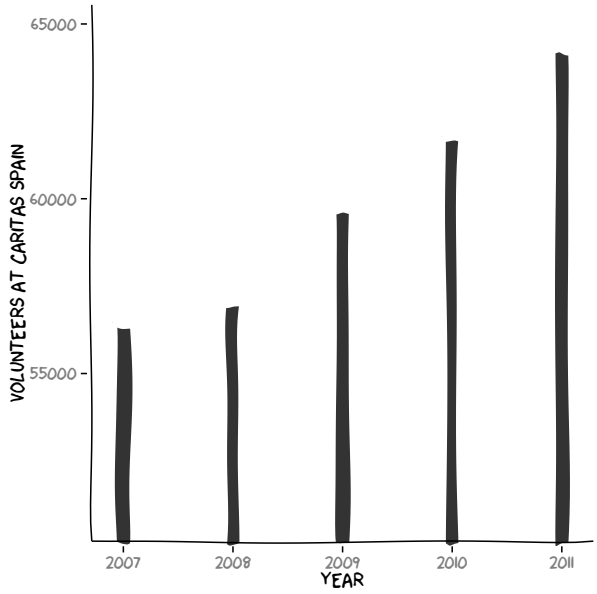
```
> volunteers <- data.frame(year=c(2007:2011), number=c(56470, 56998, 59686, 61783, 64251))
> xrange <- range(volunteers$year)
> yrange <- range(volunteers$number)
> ratioxy <-  diff(xrange) / diff(yrange)
> mapping <- aes(x,  y,
                 scale,
                 ratioxy,
                 angleofspine ,
                 anglerighthumerus,
                 anglelefthumerus,
                 anglerightradius,
                 angleleftradius,
                 anglerightleg,
                 angleleftleg,
                 angleofneck)
> dataman <- data.frame( x= c(2008,2010), y=c(63000, 58850),
                       scale = 1000 ,
                       ratioxy = ratioxy,
                       angleofspine =  -pi/2  ,
                       anglerighthumerus = c(-pi/6, -pi/6),
                       anglelefthumerus = c(-pi/2 - pi/6, -pi/2 - pi/6),
                       anglerightradius = c(pi/5, -pi/5),
```

```
                    angleleftradius = c(pi/5, -pi/5),
                    angleleftleg = 3*pi/2  + pi / 12 ,
                    anglerightleg = 3*pi/2  - pi / 12,
                    angleofneck = runif(1, 3*pi/2-pi/10, 3*pi/2+pi/10))
> datalines <- data.frame(x=c(2008.3,2010.5),y=c(63000,59600),
                          xend=c(2008.5,2010.3), yend=c(63400,59000))
> p <- ggplot() + geom_smooth(mapping=aes(x=year, y =number), data =volunteers,method="loess")
> p + xkcdaxis(xrange,yrange) +
   ylab("Volunteers at Spanish Caritas") +
   xkcdman(mapping, dataman) +
   annotate("text", x=2008.7, y = 63700, label = "We Need\nVolunteers!", family="xkcd" ) +
   annotate("text", x=2010.5, y = 60000, label = "Sure\nI can!", family="xkcd" ) +
   xkcdline(aes(x,y,xend,yend),datalines, xjitteramount = 0.12)
```



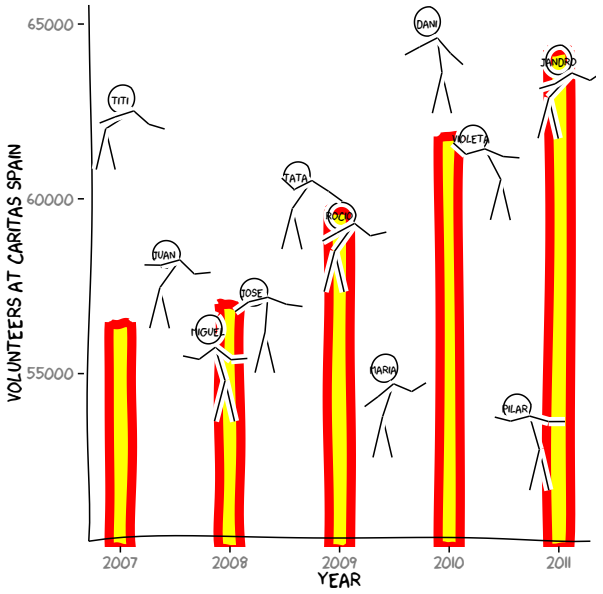

```
> data <- volunteers
> data$xmin <- data$year - 0.1
> data$xmax <- data$year + 0.1
> data$ymin <- 50000
> data$ymax <- data$number
> xrange <- range(min(data$xmin)-0.1, max(data$xmax) + 0.1)
> yrange <- range(min(data$ymin)+500, max(data$ymax) + 1000)
> mapping <- aes(xmin=xmin,ymin=ymin,xmax=xmax,ymax=ymax)
> p <- ggplot() + xkcdrect(mapping,data) +
   xkcdaxis(xrange,yrange) +
   xlab("Year") + ylab("Volunteers at Caritas Spain")
```

```
> p
```




```
> data <- volunteers
> data$xmin <- data$year - 0.1
> data$xmax <- data$year + 0.1
> data$ymin <- 50000
> data$ymax <- data$number
> xrange <- range(min(data$xmin) - 0.1, max(data$xmax) + 0.1)
> yrange <- range(min(data$ymin) +500 , max(data$ymax) + 1000)
> ratioxy <- diff(xrange)/diff(yrange)
> plotvolunteers <- function(x,y,scale,ratioxy) {
   p <- NULL
   mapping <- aes(x,  y,
                  scale,
                  ratioxy,
                  angleofspine ,
                  anglerighthumerus,
                  anglelefthumerus,
                  anglerightradius,
                  angleleftradius,
                  anglerightleg,
                  angleleftleg,
                  angleofneck)
   for( i in 1:length(x)) {
     data <- data.frame(x=x[i],
```

```
                        y=y[i],
                        scale = scale,
                        ratioxy = ratioxy,
                        angleofspine = runif(1, - pi/2 - pi/3, - pi/3 ),
                        anglerighthumerus = runif(1, -pi/6- pi/10, - pi/6 + pi/10),
                        anglelefthumerus = runif(1, pi + pi/6 -pi/10, pi + pi/6 + pi/10),
                        anglerightradius =  runif(1, -pi/4, pi/4),
                        angleleftradius =  runif(1, pi -pi/4, pi + pi/4),
                        anglerightleg = runif(1,  3* pi/2 - pi/12 - pi/180, 3* pi/2 - pi/12 + pi/180),
                        angleleftleg = runif(1, 3* pi/2 + pi/12 - pi/180, 3* pi/2 + pi/12 - pi/180),
                        angleofneck = runif(1, -pi/2-pi/10, -pi/ + pi/10))
     q <- xkcdman(mapping,data)
    p <- c(p,q)
   }
   p
 }
> volun <- c("Pilar","Maria","Miguel","Jose","Juan","Rocio","Tata","Violeta","Titi","Jandro","Dani")
> positionx <- seq(2007,2011, length.out=length(volun))
> positionx <- positionx[sample(1:length(volun),length(volun))]
> positiony <- seq(54000,65000,length.out = length(volun))
> a <- ggplot() + xkcdrect(mapping,data,fill="yellow",colour="red") +
   xkcdaxis(xrange,yrange) +
   xlab("Year") + ylab("Volunteers at Caritas Spain")
> b <- a + plotvolunteers(positionx, positiony,1000, ratioxy)
> c <- b + annotate("text", x= positionx, y= positiony, label=volun, family="xkcd",size=3)
> c
>
```



5 Saving the graphs

5.1 png

```
> png("myfigure.png")
> print(p)
> dev.off()
```

5.2 pdf

Remember to embed the fonts!

```
> ## Borrowed from
> ## fibosworld 2013. Change fonts in ggplot2, and create xkcd style graphs
> ## \url{http://fibosworld.wordpress.com/2013/02/17/change-fonts-in-ggplot2-and-create-xkcd-style-grapl
>
> ggsave("font_ggplot.pdf", plot=p,  width=12, height=4)
> ## needed for Windows - make sure YOU have the correct path for your machine:
> ## Sys.setenv(R_GSCMD = "C:\\Program Files (x86)\\gs\\gs9.06\\bin\\gswin32c.exe")
> embed_fonts("font_ggplot.pdf")
```

6 References

Hadley Wickham 2012. ggplot2 `http://ggplot2.org/`

Randall Munroe. A webcomic of romance, sarcasm, math, and language http://xkcd.com/

Various Authors 2012. How can we make xkcd style graphs in R? http://stackoverflow.com/questions/12675147/how-can-we-make-xkcd-style-graphs-in-r

fibosworld 2013. Change fonts in ggplot2, and create xkcd style graphs http://fibosworld.wordpress.com/2013/02/17/change-fonts-in-ggplot2-and-create-xkcd-style-graphs/

Winston Chang. extrafont https://github.com/wch/extrafont