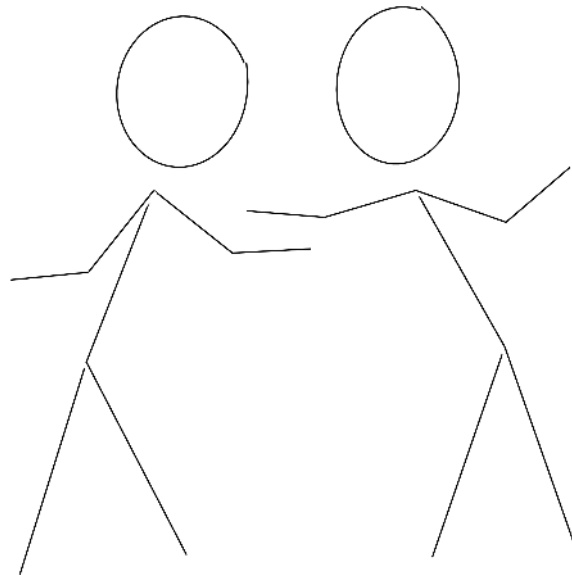


An introduction to the xkcd package

Emilio Torres-Manzanera
(torres@uniovi.es)

March 2014

Abstract



Contents

| | | |
|----------|---------------------------------|----------|
| 1 | Graph Gallery | 2 |
| 2 | The XKCD fonts | 2 |
| 2.1 | Installing fonts in R | 3 |
| 2.2 | Saving the graphs | 3 |
| 2.2.1 | png | 3 |
| 2.2.2 | pdf | 3 |

| | | |
|----------|------------------------------------|-----------|
| 3 | Installing xkcd | 4 |
| 4 | Axis | 4 |
| 5 | Cartoon characters | 5 |
| 5.1 | Facets | 6 |
| 5.2 | Angles of the xkcdman | 8 |
| 6 | Mother's day | 8 |
| 6.1 | Bar chart | 8 |
| 7 | Volunteers at Cáritas Spain | 10 |
| 7.1 | Scatterplot | 10 |
| 7.2 | Bar chart | 12 |
| 7.3 | Bar chart | 13 |
| 8 | References | 15 |

1 Graph Gallery

See more examples at <http://xkcd.r-forge.r-project.org/>.

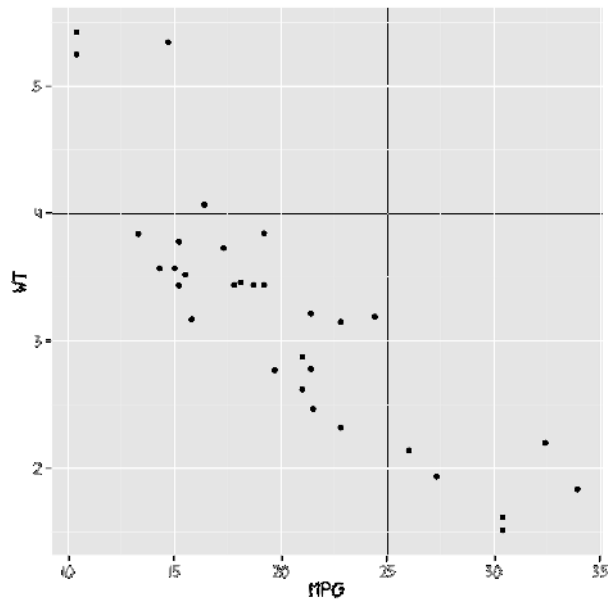
2 The XKCD fonts

The package `xkcd` uses the XKCD fonts. Therefore, an easy way to check whether this fonts are installed in the computer is typing the following code and comparing the graphs:

```

1 library(sysfonts)
2 library(ggplot2)
3 if( "xkcd.ttf" %in% font.files() ) {
4   font.add("xkcd", regular = "xkcd.ttf")
5   p <- ggplot() + geom_point(aes(x=mpg, y=wt), data=mtcars) +
6     theme(text = element_text(size = 16, family = "xkcd"))
7 } else {
8   warning("Not xkcd fonts installed!")
9   p <- ggplot() + geom_point(aes(x=mpg, y=wt), data=mtcars)
10 }
11 p

```



2.1 Installing fonts in R

The XKCD fonts are not installed in the system. You can use the package `sysfonts`, and the function `font.paths()` to check the current search path or add a new one, and use `font.files()` to list available font files in the search path.

```

1 library(sysfonts)
2 download.file("http://simonsoftware.se/other/xkcd.ttf", dest="xkcd.ttf", mode="wb")
3 font.paths()
4 system("mkdir ~/.fonts")
5 system("cp xkcd.ttf -t ~/.fonts")
6 font.files()
7 font.add("xkcd", regular = "xkcd.ttf")
8 font.families()

```

2.2 Saving the graphs

2.2.1 png

```

1 font.add("xkcd", regular = "xkcd.ttf")
2 p <- ggplot() + geom_point(aes(x=mpg, y=wt), data=mtcars) +
3   theme(text = element_text(size = 16, family = "xkcd"))
4 ggsave("fig.png", p)

```

2.2.2 pdf

Yixuan Qiu, author of the packages `sysfonts` and `showtext`, provides an example of saving pdf plots:

```

1 library(showtext)
2 font.add("xkcd", "xkcd.ttf")
3 pdf("showtext.pdf")
4 showtext.begin()
5 print(p)
6 showtext.end()
7 dev.off()

```

3 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

```
1 install.packages("xkcd",dependencies = TRUE)
```

Then, you may want to see the vignette and check the code:

```

1 help(package="xkcd")
2 vignette("xkcd-intro") # it opens the pdf
3 browseVignettes(package = "xkcd") # To browse the pdf, R and Rnw

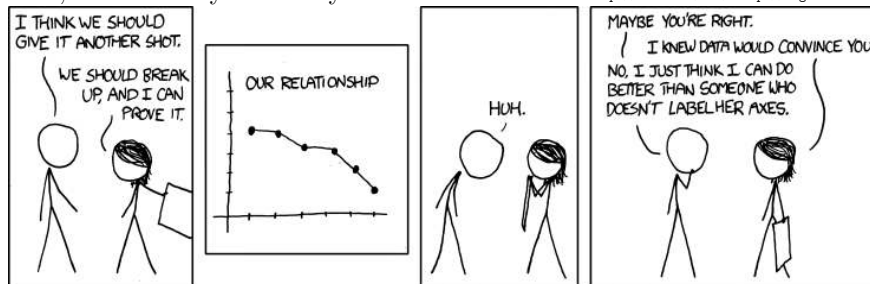
```

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

```
1 library(xkcd)
```

4 Axis

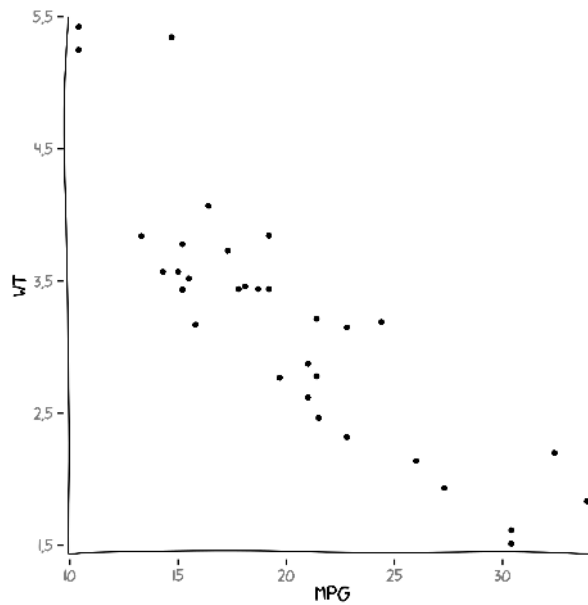
Man: No, I just think I can do better than someone who doesn't label her axes. Title text: And if you labeled your axes, I could tell you exactly how MUCH better. <http://xkcd.com/833/> <http://imgs.xkcd.com/comics/convincing.png>



```

1 xrange <- range(mtcars$mpg)
2 yrange <- range(mtcars$wt)
3 set.seed(123) # for reproducibility
4 p <- ggplot() + geom_point(aes(mpg, wt), data=mtcars) +
5   xkcdaxis(xrange,yrange)
6 p

```



5 Cartoon characters

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

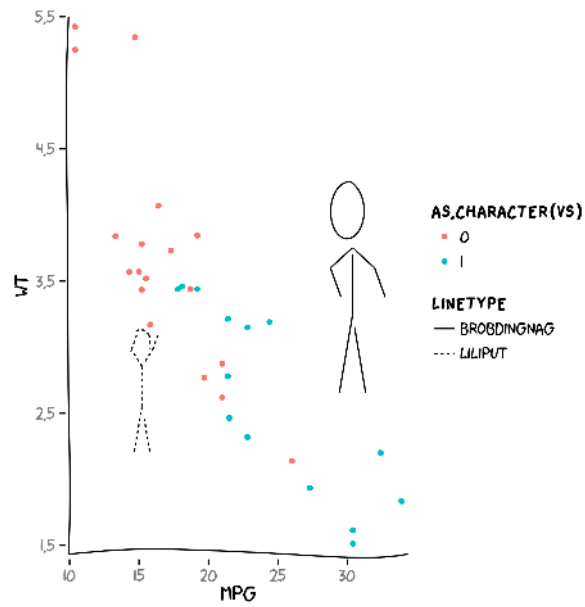
```

1 ratioxy <- diff(xrange)/diff(yrange)
2 mapping <- aes(x, y,
3               scale,
4               ratioxy,
5               angleofspine ,
6               anglerighthumerus,
7               anglelefthumerus,
8               anglerightradius,
9               angleleftradius,
10              anglerightleg,
11              angleleftleg,
12              angleofneck,
13              linetype=city)
14 dataman <- data.frame(x= c(15,30), y=c(3, 4),
15                       scale = c(0.3,0.51) ,
16                       ratioxy = ratioxy,
17                       angleofspine = -pi/2 ,
18                       anglerighthumerus = c(pi/4, -pi/6),
19                       anglelefthumerus = c(pi/2 + pi/4, pi +pi/6),
20                       anglerightradius = c(pi/3, -pi/3),
21                       angleleftradius = c(pi/3, -pi/3),
22                       anglerightleg = 3*pi/2 - pi / 12,
```

```

23     angleleftleg = 3*pi/2 + pi / 12 ,
24     angleofneck = runif(1, 3*pi/2-pi/10, 3*pi/2+pi/10),
25     city=c("Liliput","Brobdingnag"))
26 q <- ggplot() + geom_point(aes(mpg, wt, colour=as.character(vs)), data=mtcars) +
27     xkcdaxis(xrange,yrange) + xkcdman(mapping, dataman)
28 q

```

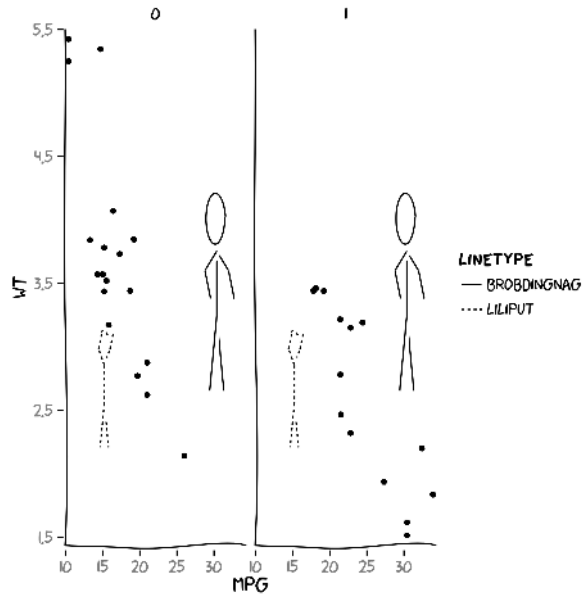


5.1 Facets

```

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

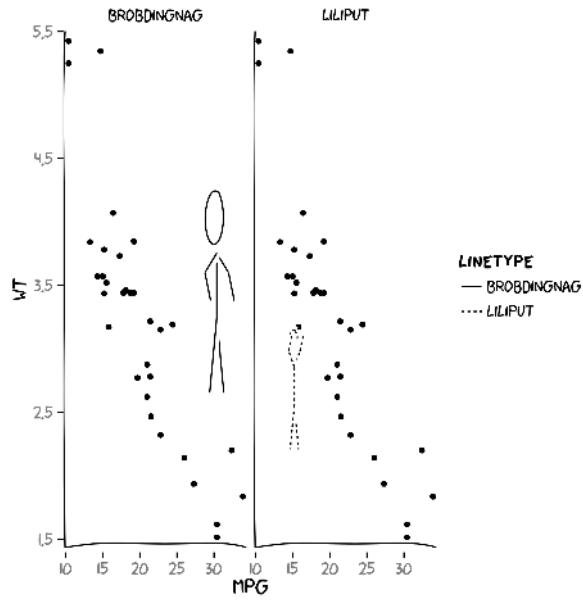
```



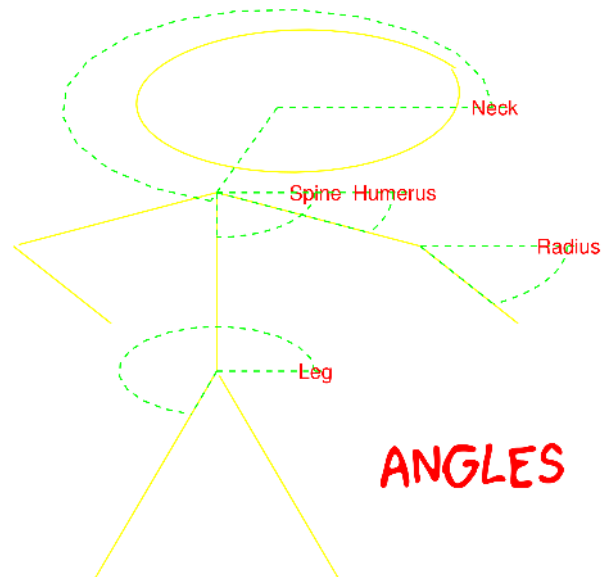
```

1 ggplot() + geom_point(aes(mpg, wt), data=mtcars) +
2   xkcdaxis(xrange,yrange) + xkcdman(mapping, dataman) +
3   facet_grid(.~city)

```



5.2 Angles of the xkcdman



6 Mother's day

6.1 Bar chart

```
1 mommy <- read.table(sep=" ",text ="
2 8 100
3 9 0
4 10 0
5 11 0
6 12 0
7 13 0
8 14 100
9 15 100
10 16 500
11 17 420
12 18 75
13 19 50
14 20 100
15 21 40
16 22 0
17 ")
18 names(mommy) <- c("hour","number")
19 data <- mommy
20 data$xmin <- data$hour - 0.25
```



```

21 data$xmax <- data$xmin + 1
22 data$ymin <- 0
23 data$ymax <- data$number
24 xrange <- range(8, 24)
25 yrange <- range(min(data$ymin) + 10 , max(data$ymax) + 200)
26 ratioxy <- diff(xrange)/diff(yrange)
27 timelabel <- function(text,x,y) {
28   if( "xkcd" %in% font.families()){
29     te1 <- annotate("text", x=x, y = y + 65, label=text, size = 6,family ="xkcd")
30   } else {
31     te1 <- annotate("text", x=x, y = y + 65, label=text, size = 6)}
32   list(te1,
33     xkcdline(aes(xbegin=xbegin, ybegin= ybegin, xend=xend,yend=yend),
34       data.frame(xbegin=x,ybegin= y + 50, xend=x,yend=y), xjitteramount = 0.5))
35   }
36 n <- 1800
37 set.seed(123)
38 x <- runif(n, xrange[1],xrange[2] )
39 y <- runif(n, yrange[1],yrange[2] )
40 inside <- unlist(lapply(1:n, function(i) any(data$xmin <= x[i] & x[i] < data$xmax &
41     data$ymin <= y[i] & y[i] < data$ymax)))
42 x <- x[inside]
43 y <- y[inside]
44 nman <- length(x)
45 sizer <- round(runif(nman, 1, 10),0)
46 angler <- runif(nman, -10,10)
47 if( "xkcd" %in% font.families()){
48 p <- ggplot() +
49   geom_text(aes(x,y,label="Mummy",angle=angler,hjust=0, vjust=0),
50     family="xkcd",size=sizer,alpha=0.3) +
51   xkcdaxis(xrange,yrange) +
52   annotate("text", x=16, y = 650,
53     label="Happy Mother's day", size = 16,family ="xkcd") +
54   xlab("daily schedule") +
55   ylab("Number of times mothers are called on by their children") +
56   timelabel("Wake up", 9, 125) + timelabel("School", 12.5, 90) +
57   timelabel("Lunch", 15, 130) +
58   timelabel("Homework", 18, 525) +
59   timelabel("Bath", 21, 110) +
60   timelabel("zzz", 23.5, 60)
61 } else {
62 p <- ggplot() +
63   geom_text(aes(x,y,label="Mummy",angle=angler,hjust=0, vjust=0),
64     size=sizer,alpha=0.3) +
65   xkcdaxis(xrange,yrange) +
66   annotate("text", x=16, y = 650,
67     label="Happy Mother's day", size = 16) +
68   xlab("daily schedule") +
69   ylab("Number of times mothers are called on by their children") +

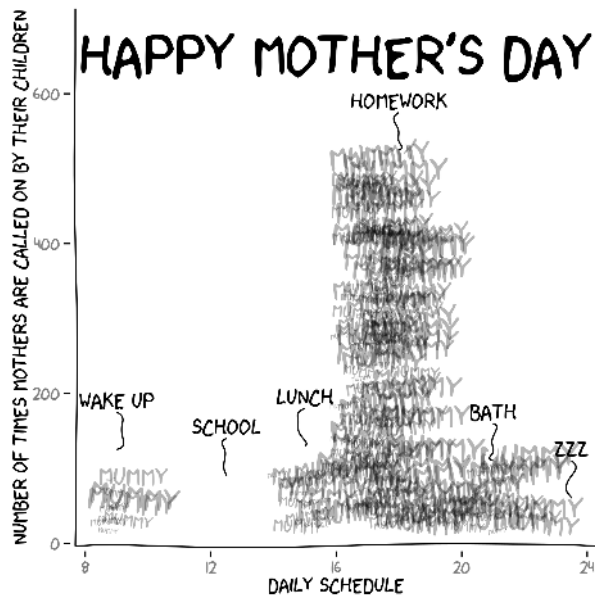
```

```

70 timelabel("Wake up", 9, 125) + timelabel("School", 12.5, 90) +
71 timelabel("Lunch", 15, 130) +
72 timelabel("Homework", 18, 525) +
73 timelabel("Bath", 21, 110) +
74 timelabel("zzz", 23.5, 60)}

```

75 p



7 Volunteers at Cáritas Spain

7.1 Scatterplot

```

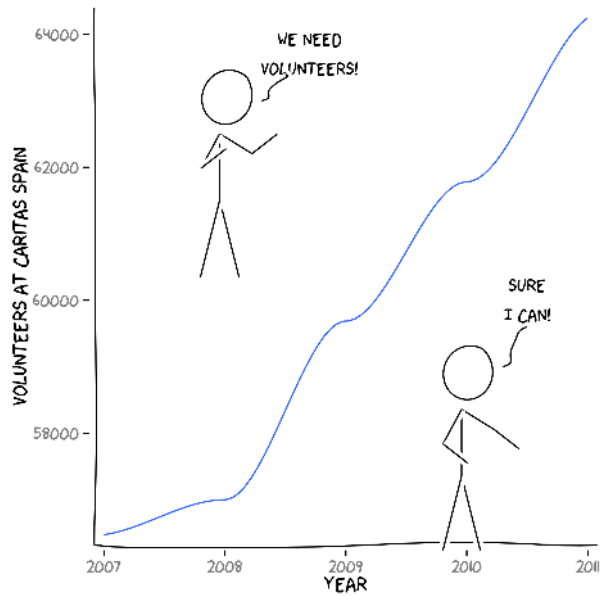
1 volunteers <- data.frame(year=c(2007:2011), number=c(56470, 56998, 59686, 61783, 64251))
2 xrange <- range(volunteers$year)
3 yrange <- range(volunteers$number)
4 ratioxy <- diff(xrange) / diff(yrange)
5 mapping <- aes(x, y,
6               scale,
7               ratioxy,
8               angleofspine,
9               anglerighthumerus,
10              anglelefthumerus,
11              anglerightradius,
12              angleleftradius,
13              anglerightleg,
14              angleleftleg,
15              angleofneck)

```

```

16 dataman <- data.frame( x= c(2008,2010), y=c(63000, 58850),
17                       scale = 1000 ,
18                       ratioxy = ratioxy,
19                       angleofspine = -pi/2 ,
20                       anglerighthumerus = c(-pi/6, -pi/6),
21                       anglelefthumerus = c(-pi/2 - pi/6, -pi/2 - pi/6),
22                       anglerightradius = c(pi/5, -pi/5),
23                       angleleftradius = c(pi/5, -pi/5),
24                       angleleftleg = 3*pi/2 + pi / 12 ,
25                       anglerightleg = 3*pi/2 - pi / 12,
26                       angleofneck = runif(1, 3*pi/2-pi/10, 3*pi/2+pi/10))
27 datalines <- data.frame(xbegin=c(2008.3,2010.5),ybegin=c(63000,59600),
28                        xend=c(2008.5,2010.3), yend=c(63400,59000))
29 p <- ggplot() + geom_smooth(mapping=aes(x=year, y =number), data =volunteers,method="loess")
30 if( "xkcd" %in% font.families()){
31 p + xkcdaxis(xrange,yrange) +
32   ylab("Volunteers at Caritas Spain") +
33   xkcdman(mapping, dataman) +
34   annotate("text", x=2008.7, y = 63700, label = "We Need\nVolunteers!", family="xkcd" ) +
35   annotate("text", x=2010.5, y = 60000, label = "Sure\nI can!", family="xkcd" ) +
36   xkcdline(aes(xbegin=xbegin,ybegin=ybegin,xend=xend,yend=yend),datalines, xjitteramount = 0.12)
37 } else {
38 p + xkcdaxis(xrange,yrange) +
39   ylab("Volunteers at Caritas Spain") +
40   xkcdman(mapping, dataman) +
41   annotate("text", x=2008.7, y = 63700, label = "We Need\nVolunteers!") +
42   annotate("text", x=2010.5, y = 60000, label = "Sure\nI can!") +
43   xkcdline(aes(xbegin=xbegin,ybegin=ybegin,xend=xend,yend=yend),datalines, xjitteramount = 0.12)
44 }

```

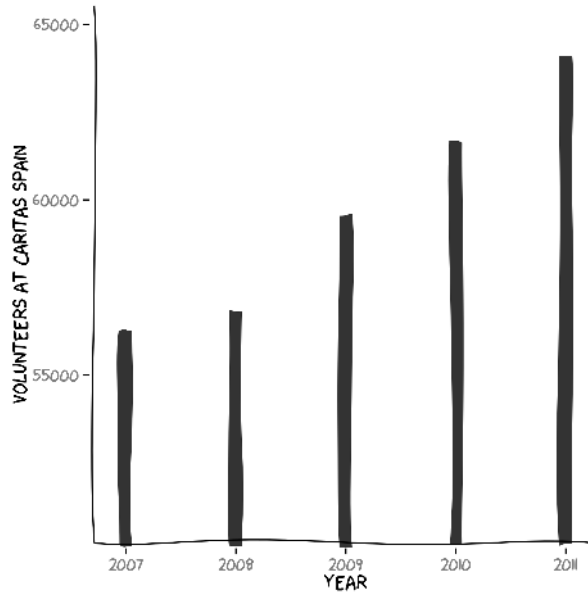


7.2 Bar chart

```

1 data <- volunteers
2 data$xmin <- data$year - 0.1
3 data$xmax <- data$year + 0.1
4 data$ymin <- 50000
5 data$ymax <- data$number
6 xrange <- range(min(data$xmin)-0.1, max(data$xmax) + 0.1)
7 yrange <- range(min(data$ymin)+500, max(data$ymax) + 1000)
8 mapping <- aes(xmin=xmin,ymin=ymin,xmax=xmax,ymax=ymax)
9 p <- ggplot() + xkcdrect(mapping,data) +
10   xkcdaxis(xrange,yrange) +
11   xlab("Year") + ylab("Volunteers at Caritas Spain")
12 p

```



7.3 Bar chart

```

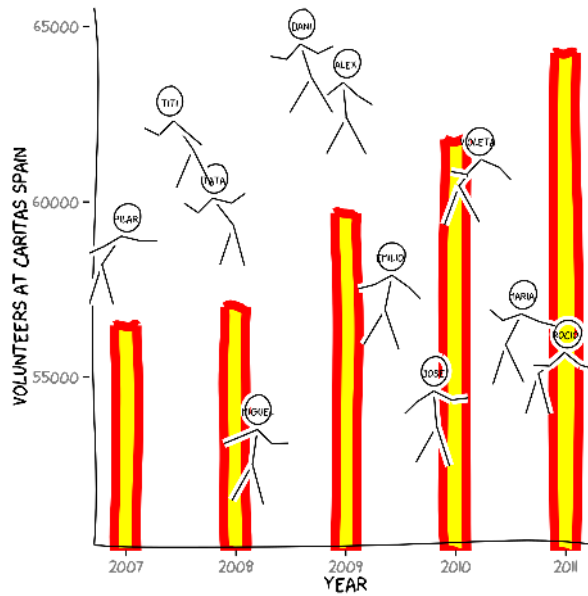
1 data <- volunteers
2 data$xmin <- data$year - 0.1
3 data$xmax <- data$year + 0.1
4 data$ymin <- 50000
5 data$ymax <- data$number
6 xrange <- range(min(data$xmin) - 0.1, max(data$xmax) + 0.1)
7 yrange <- range(min(data$ymin) + 500, max(data$ymax) + 1000)
8 ratioxy <- diff(xrange)/diff(yrange)
9 plotmen <- function(x,y, scale,ratioxy,...){
10   mapping <- aes(x, y,
11                 scale,
12                 ratioxy,
13                 angleofspine,
14                 anglerighthumerus,
15                 anglelefthumerus,
16                 anglerightradius,
17                 angleleftradius,
18                 anglerightleg,
19                 angleleftleg,
20                 angleofneck)
21   n <- length(x)
22   data <- data.frame(x=x,
23                     y=y,
24                     scale = scale,

```

```

25         ratioxy = ratioxy,
26         angleofspine = runif(n, - pi/2 - pi/3, -pi/2 + pi/3),
27         anglerighthumerus = runif(n, -pi/6- pi/10, - pi/6 + pi/10),
28         anglelefthumerus = runif(n, pi + pi/6 -pi/10, pi + pi/6 + pi/10),
29         anglerightradius = runif(n, -pi/4, pi/4),
30         angleleftradius = runif(n, pi -pi/4, pi + pi/4),
31         anglerightleg = runif(n, 3* pi/2 + pi/12 , 3* pi/2 + pi/12 + pi/10),
32         angleleftleg = runif(n, 3* pi/2 - pi/12 - pi/10, 3* pi/2 - pi/12 ),
33         angleofneck = runif(n, -pi/2-pi/10, -pi/2 + pi/10))
34     xkcdman(mapping,data,...)
35 }
36 volun <- c("Miguel","Jose","Rocio","Maria","Emilio",
37           "Pilar","Tata","Violeta","Titi","Alex","Dani")
38 positionx <- seq(2007,2011, length.out=length(volun))
39 set.seed(123)
40 positionx <- positionx[sample(1:length(volun),length(volun))]
41 positiony <- seq(54000,65000,length.out = length(volun))
42 a <- ggplot() +
43     xkcdrect(mapping,data,fill="yellow",colour="red") +
44     xkcdaxis(xrange,yrange) +
45     xlab("Year") + ylab("Volunteers at Caritas Spain")
46 b <- a + plotmen(positionx, positiony,1000, ratioxy)
47 if( "xkcd" %in% font.families()){
48 c <- b + annotate("text",
49                 x= positionx, y= positiony,
50                 label=volun, family="xkcd",size=3)
51 } else {
52 c <- b + annotate("text",
53                 x= positionx, y= positiony,
54                 label=volun,size=3)
55 }
56 c

```



8 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>

Yixuan Qiu, 2014, sysfonts, <http://cran.r-project.org/web/packages/sysfonts/index.html>

Yixuan Qiu, 2014, showtext, <http://cran.r-project.org/web/packages/showtext/index.html>