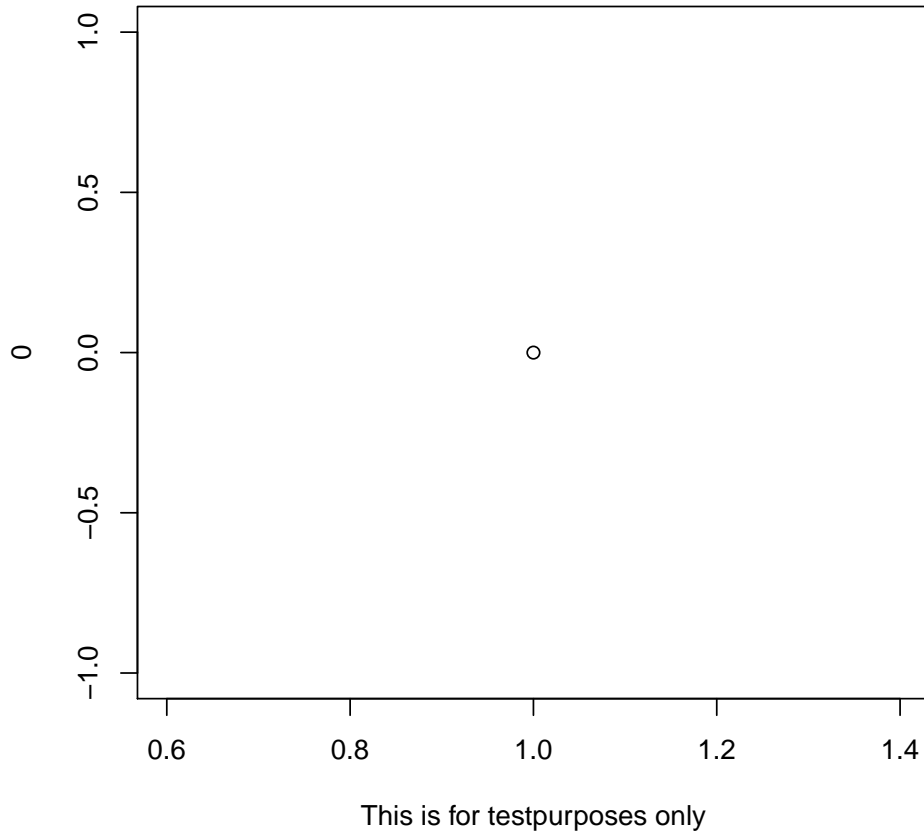


Test SplomT

Christian W. Hoffmann

September 24, 2012


```
> plot(0,xlab="This is for testpurposes only")
```



```
> SplomT <- function (data, mainL = deparse(substitute(data)), xlabL = "",
+   hist = "h", adjust = 1, hist.col = trellis.par.get("strip.background")$col[5], cex.diag
+   stopifnot (hist %in% c("h", "d", "b"))
+   data <- data.frame(data)
+   mxnam <- max(nchar(names(data)))
+   lnam <- ncol(data)
+   ce <- 100*cex.diag*get.gpar()$cex/lnam
+   cexd <- ce/mxnam
+   cexn <- ce/5
+   print(splom(~data, as.matrix = TRUE, main = mainL, xlab = xlabL,
+     upper.panel = function(x, y, breaks = NULL, ...) {
+       minS <- 0.05
+       ccr <- cor(x, y, use = "complete.obs")
+       ccq <- sqrt(max(abs(ccr), minS))
+       if (is.na(ccr)) {ccr <- 0; ccq <- sqrt(minS)}
+       grid.text(round(ccr, 2), gp = gpar(cex = cexn*ccq))
+     },
+     lower.panel = function(x, y, ...) {
+       options(show.error.messages = FALSE)
+       try(panel.xyplot(x, y, type = c("p", "smooth"), col.line = colYonX,
+         pch = 3, cex = 1.5/dim(data)[2], ...))
+       lo <- try(loess.smooth(y, x, ...))
+     }
+   )
+ }
```

```

+     if (!inherits(lo,"try-error")) panel.lines(lo$y, lo$x, col.line = colXonY, ...)
+     options(show.error.messages = TRUE)
+   },
+   diag.panel = function(x, varname, limits, ...) {
+     d <- density(x[!is.na(x)])
+     yrng <- range(d$y)
+     ylim <- yrng + 0.07 * c(-1, 1) * diff(yrng)
+     xlim <- current.panel.limits()$xlim
+     pushViewport(viewport(xscale = xlim, yscale = ylim))
+     if (hist %in% c("h", "b")) {
+       panel.histogram(x[!is.na(x)], breaks = NULL, col = hist.col, type = "density", ...)
+     }
+     if (hist %in% c("d", "b")) {
+       llines(d)
+     }
+     grid.text(varname, y=unit(h.diag,"npc"), gp = gpar(cex = cexd))
+     popViewport()
+   }, varnames = abbreviate(names(data)), pscales = 0 )
+ )
+ } ## end SplomT 2012-09-24, 16:25
> nr <- 100; nc <- 8;
> data <- as.data.frame(matrix(rnorm(nr*nc),nrow=nr,ncol=nc))
> data[,nc] <- data[,nc-2] + 0.3*data[,nc-1] #generate higher correlations
> data[,nc-1] <- data[,nc-1] + 0.9*data[,nc]
> colnames(data)<-paste("vw",letters[1:nc],sep="")
> SplomT(data,mainL="",hist="d",cex.diag=0.6,hist.col="green")

```

