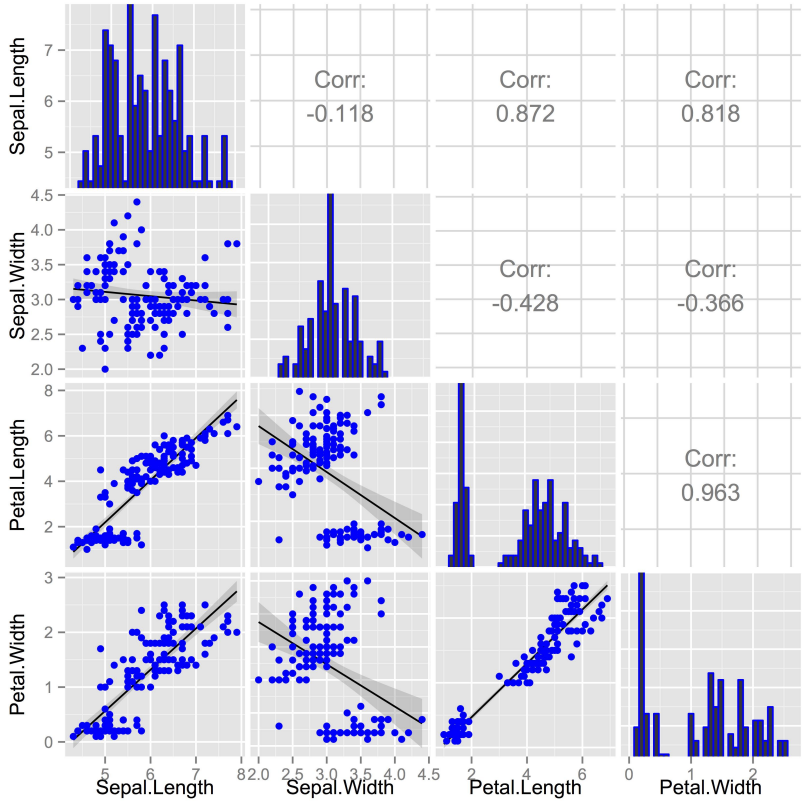
How to customize lines in ggpairs [GGally]

I have the following plot:



Generated with this code:

```
library("GGally")
data(iris)
ggpairs(iris[, 1:4], lower=list(continuous="smooth", params=c(colour="blue")),
  diag=list(continuous="bar", params=c(colour="blue")),
  upper=list(params=list(corSize=6)), axisLabels='show')
```

My questions are:

1. How can I change the correlation line to be `red` , now it's black.
2. And the correlation line is buried under the scatter plot. I want to put it on top. How can I do that?

r ggplot2 ggally

edited Jun 16 '15 at 18:59
jenesaisquoi
14.5k 3 13 41

asked Jun 16 '15 at 3:35
neversaint
8,548 41 136 223

You can control the transparency of the dots in the scatter plot by adding `alpha=0.3` to your lower list params. This will help focus the smooth lines more. – nehiljain Jun 16 '15 at 3:55

@nehiljain: No that won't do. When the scatter plot is dense the line will still be buried. I was thinking in this line . But don't know how to implement it in ggpairs. – neversaint Jun 16 '15 at 4:08

1 Answer

I hope there is an easier way to do this, but this is a sort of brute force approach. It does give you flexibility to easily customize the plots further however. The main point is using `putPlot` to put a `ggplot2` plot into the figure.

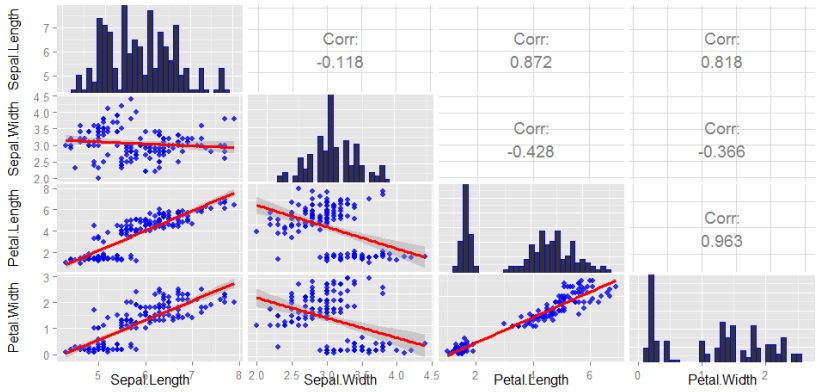
```
library(ggplot2)

## First create combinations of variables and extract those for the lower matrix
cols <- expand.grid(names(iris)[1:4], names(iris)[1:3])
cols <- cols[c(2:4, 7:8, 12),]  # indices will be in column major order

## These parameters are applied to each plot we create
pars <- list(geom_point(alpha=0.8, color="blue"),
             geom_smooth(method="lm", color="red", lwd=1.1))

## Create the plots (dont need the lower plots in the ggpairs call)
plots <- apply(cols, 1, function(cols)
    ggplot(iris[,cols], aes_string(x=cols[2], y=cols[1])) + pars)
gg <- ggpairs(iris[, 1:4],
              diag=list(continuous="bar", params=c(colour="blue")),
              upper=list(params=list(corSize=6)), axisLabels='show')

## Now add the new plots to the figure using putPlot
colFromRight <- c(2:4, 3:4, 4)
colFromLeft <- rep(c(1, 2, 3), times=c(3,2,1))
for (i in seq_along(plots))
    gg <- putPlot(gg, plots[[i]], colFromRight[i], colFromLeft[i])
gg
```




```
## If you want the slope of your lines to correspond to the
## correlation, you can scale your variables
scaled <- as.data.frame(scale(iris[,1:4]))
fit <- lm(Sepal.Length ~ Sepal.Width, data=scaled)
coef(fit)[2]
# Sepal.Length
#  -0.1175698

## This corresponds to Sepal.Length ~ Sepal.Width upper panel
```

Edit

To generalize to a function that takes any column indices and makes the same plot

```
## colInds is indices of columns in data.frame
.ggpairs <- function(colInds, data=iris) {
    n <- length(colInds)
    cols <- expand.grid(names(data)[colInds], names(data)[colInds])
    cInds <- unlist(mapply(function(a, b, c) a*n+b:c, 0:max(0,n-2), 2:n, rep(n, n-1)))
    cols <- cols[cInds,]  # indices will be in column major order

    ## These parameters are applied to each plot we create
    pars <- list(geom_point(alpha=0.8, color="blue"),
                 geom_smooth(method="lm", color="red", lwd=1.1))

    ## Create the plots (dont need the lower plots in the ggpairs call)
    plots <- apply(cols, 1, function(cols)
        ggplot(data[,cols], aes_string(x=cols[2], y=cols[1])) + pars)
    gg <- ggpairs(data[, colInds],
                  diag=list(continuous="bar", params=c(colour="blue")),
                  upper=list(params=list(corSize=6)), axisLabels='show')

    rowFromTop <- unlist(mapply(`:`, 2:n, rep(n, n-1)))
    colFromLeft <- rep(1:(n-1), times=(n-1):1)
    for (i in seq_along(plots))
        gg <- putPlot(gg, plots[[i]], rowFromTop[i], colFromLeft[i])
    return( gg )
}

## Example
.ggpairs(c(1, 3))
```

edited Jun 16 '15 at 7:04

answered Jun 16 '15 at 4:49

jenesaisquoi
14.5k 3 13 41

@LegalizeIt: How can I generalize your code into a function so that it can take data with any number of columns? (i.e. Now you assume 4 columns) – neversaint Jun 16 '15 at 6:20

1 It shouldn't be too hard, the indices just need to be generalized, I can give it a shot – jenesaisquoi Jun 16 '15 at 6:28

@LegalizeIt: `colFromRight` do you mean `rowFromTop` ? – neversaint Jun 16 '15 at 6:39

1 You could try out that function – jenesaisquoi Jun 16 '15 at 7:05

1 glad it helped! – jenesaisquoi Jun 16 '15 at 7:28