R:Correlation and Its Visualization

徐爽
E-mail:henuxs@foxmail.com

2016 年 2 月 1 日

1 相关性

相关性（correlation）指两个随机变量或两组数据的统计关系。一般用相关系数来描述相关性的强弱，常记为 ρ 或 r。使用相关系数的时候必须明白两点，不可以滥用统计工具：第一，相关性是预测性关系，不是因果关系。比如，炎热的夏天用电量比较高。因为人们会使用空调降温，这两件事是因果关系，统计相关性不足以证实两者的因果关系。第二，不相关和独立有差别。不相关指 $r=0$，独立指 $Pr(X,Y)=Pr(X)Pr(Y)$。当相关系数定义为 Pearson 相关系数时：独立一定不相关，不相关不一定独立；不独立不一定相关，相关一定不独立。

下面介绍几种常用的相关系数。

1.1 Pearson Correlation Coefficient

Pearson 相关系数描述了两个变量或两组数据的线性相关性，常简称为（线性）相关系数，取值于 $[-1,1]$。两个随机变量的相关系数定义为

$$\begin{aligned}\rho(X,Y) &= \frac{COV(X,Y)}{D(X)D(Y)} \\ &= \frac{E[X-E(X))(Y-E(Y)]}{D(X)D(Y)} \\ &= \frac{E(XY)-E(X)E(Y)}{\sqrt{E(X^2)-[E(X)]^2}\sqrt{E(Y^2)-[E(Y)]^2}}.\end{aligned}$$

两组数据的相关系数定义为

$$r_{xy}=\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i-\bar{y})^2}}.$$

若 $r=\pm 1$，说明数据有完全相关性，即所有的数据点都位于同一条直线上。我们经常使用 t 检验来判断数据是否相关。样本 $(x_1,y_1),(x_2,y_2),\cdots,(x_n,y_n)$ 来自正态分布，相关性未知，令原假设 $H_0:r_{xy}=0$，备择假设 $H_1:r_{xy}\neq 0$。在原假设成立的条件下，t 统计量

$$t=r\sqrt{\frac{n-2}{1-r^2}}\sim t(n-2).$$

如果样本不服从正态分布，在样本量充分大时，t 统计量近似服从 t 分布。r 越大 t 统计量越大，越应该拒绝原假设。

1.2 Rank Correlation Coefficient

Spearman 相关系数是一种常用的秩相关系数，这里的秩表示数据的大小顺序。具体定义如下：假设有随机样本 $x_1, x_2, \cdots, x_n$，若对任意 i, j 满足 $x_i \neq x_j$，则第 i 小的样本的秩为 i；若存在 i, j 满足 $x_i = x_j$，则称这个样本是打结的 (tied)，先按照上面的定义计算秩，对于打结样本求平均即可。比如，不打结样本 5,4,8,9，它们的秩依次为 2,1,3,4。打结样本 5,5,1,1,1,7，先按照原先定义计算秩：4,5,1,2,3,6，再对打结样本求平均数，最终 5 的秩为 4.5，1 的秩为 2，7 的秩为 6。

假设数据 x_i 的秩为 a_i，数据 y_i 的秩为 b_i，则 x 的平均秩为 $\bar{a} = \sum_{i=1}^{n} a_i/n$，$x$ 的秩方差为 $s_x = \sum_{i=1}^{n}(a_i - \bar{a})^2/n$，类似地可以对 y 定义平均秩和秩方差。Spearman 相关系数为

$$\begin{aligned}\rho_{xy} &= \frac{\sum_{i=1}^{n}(a_i - \bar{a})(b_i - \bar{b})}{\sqrt{s_x s_y}} \\ &= 1 - \frac{6\sum_{i=1}^{n}(a_i - b_i)^2}{n^3 - n}.\end{aligned}$$

对于没有结或结不多的样本，Pearson 相关系数的 t 检验仍适用。

Kendall 相关系数是另外一种常用的秩相关系数。假设有 n 对观测值 $(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)$，如果 $(x_i - x_j)(y_i - y_j) > 0$，则称数对 (x_i, y_i) 和 (x_j, y_j) 是协同的，否则称为不协同。分别记协同对和不协同对的数量为 N_c 和 N_d，则 Kendall 相关系数为

$$\tau = \frac{N_c - N_d}{n(n-1)/2}.$$

2 在 R 中的相关系数和相关性检验

在 R 自带的程序包 stats 中，使用 cor() 可以计算相关系数。参数 method 控制计算相关系数的方法。

```
> data(trees)
> cor(trees,method="pearson")
Girth Height Volume
Girth 1.0000000 0.5192801 0.9671194
Height 0.5192801 1.0000000 0.5982497
Volume 0.9671194 0.5982497 1.0000000
```

```
> cor(trees,method="spearman")
Girth Height Volume
Girth 1.0000000 0.4408387 0.9547151
Height 0.4408387 1.0000000 0.5787101
Volume 0.9547151 0.5787101 1.0000000
```

```
> cor(trees,method="kendall")
Girth Height Volume
Girth 1.0000000 0.3168641 0.8302746
Height 0.3168641 1.0000000 0.4496306
Volume 0.8302746 0.4496306 1.0000000
```

相关性检验使用程序包 stats 中的 cor.test()，用法和 cor() 类似。参数 alternative = "two.sided", "less", "greater" 分别表示双侧检验、右侧检验、左侧检验。参数 conf.level 控制置信水平。

```
> cor.test(trees[,1],trees[,2],method="pearson")

Pearson's product-moment correlation

data: trees[, 1] and trees[, 2]
t = 3.2722, df = 29, p-value = 0.002758
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
0.2021327 0.7378538
sample estimates:
cor
0.5192801
```

程序包 Kendall 主要用于计算 Kendall 相关系数和 Mann-Kendall 趋势性检验，作者 A. I. McLeod。Kendall(x, y) 可以计算相关系数，并进行检验。如果数据没有打结，结果和 cor(x,y,method="kendall") 、cor.test(x,y,method="kendall") 一样。如果数据打结，相关系数仍相同，但是检验的 p 值不相同。

```
> x<-c(1.5,1.5,3,4,6,6,6,8,9.5,9.5,11,12)
> y<-c(2.5,2.5,7,4.5,1,4.5,6,11.5,11.5,8.5,8.5,10)
> cor.test(x,y,method="kendall")

Kendall's rank correlation tau

data: A and B
z = 1.8317, p-value = 0.067
alternative hypothesis: true tau is not equal to 0
sample estimates:
tau
0.4075438

警告信息：
In cor.test.default(A, B, method = "kendall") : 无法给连结计算精確 p 值
```

```
> install.packages("Kendall")
> library(Kendall)
> summary(Kendall(x,y))
Score = 34 , Var(Score) = 203.303
denominator = 61.49797
tau = 0.553, 2-sided pvalue =0.020645
```

程序包 pspearman 中的 spearman.test() 在不打结时提供了更精确的 p 值。cor.test(,method="spearman") 根据数据量的不同分别使用了 AS89 和 t-distribution 两种近似方法。在 spearman.test 中参数 approximation = c("exact", "AS89", "t-distribution") 可以选择近似的方法，默认是精确值。

```
> install.packages("pspearman")
> library(pspearman)
> x <- 1:10
> y <- c(5:1, 6, 10:7)
> out1 <- spearman.test(x, y)
> out2 <- spearman.test(x, y, approximation="AS89")
> out3 <- cor.test(x, y, method="spearman")
> out1$p.value # [1] 0.05443067 this is the exact value
[1] 0.05443067
> out2$p.value # [1] 0.05444507 approximation obtained from AS89
[1] 0.05444507
> out3$p.value # [1] 0.05444507 ditto
[1] 0.05444507
```

程序包 SuppDists 则给出了 Spearman、Kendall 相关系数的密度函数、分布函数、分位点函数以及随机数。下面的命令给出了样本数 10，相关系数为 0.95 的 p 值。

```
> install.packages("SuppDists")
> library(SuppDists)
> pSpearman(0.95,10)
> pSpearman(0.95,10)
[1] 0.9999755
> pKendall(0.95,10)
[1] 0.9999997
```

3 相关性的可视化

出了使用散点图矩阵可以表示数据的相关性，还可以对相关系数矩阵可视化。主要介绍一下函数：

Function	Package	Description
plotcorr	ellipse	以椭圆代表相关系数。
plotcov	pcaPP	用于两个相关系数矩阵的比较。
corrplot	corrplot	相关系数矩阵可视化专业户，推荐。
corrplot	arm	可被 ggcorr 完美代替。
ggcorr	GGally	实用性不强。
corrgram	corrgram	比 ggcorr 强一点。

4 plotcorr(ellipse)

4.1 基础

程序包 ellipse 主要提供了与椭圆置信区间相关的函数，作者 Duncan Murdoch and E. D. Chow。函数 plotcorr() 使用椭圆绘制相关性矩阵，作用于相关系数矩阵。这些椭圆的方向和形状代表了相关系数的大小。椭圆右偏 $\pi/4$ 表示正相关；左偏 $\pi/4$ 表示负相关。椭圆越扁则相关性越强，否则越弱。

```
install.packages("ellipse")
library(ellipse)
data(mtcars)
plotcorr(cor(mtcars))
```

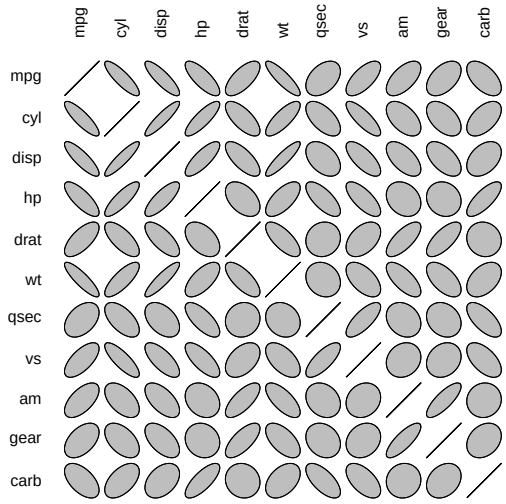


图 1: ellipse

4.2 颜色和上下三角形

仅凭形状和方向较难快速的识别相关性的正负和强弱，我们可以对不同的数值赋予不同的颜色辅助判断。

```
corr.mtcars <- cor(mtcars)
ord <- order(corr.mtcars[1,])
xc <- corr.mtcars[ord, ord]
colors <- c("#A50F15","#DE2D26","#FB6A4A","#FCAE91","#FEE5D9","white",
"#EFF3FF","#BDD7E7","#6BAED6","#3182BD","#08519C")
plotcorr(xc, col=colors[5*xc + 6])
```

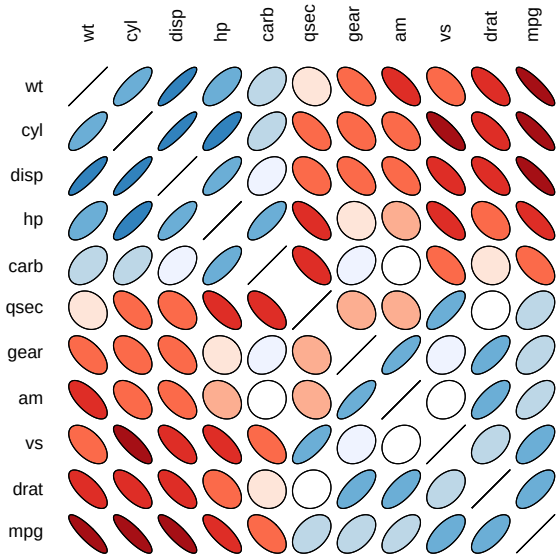


图 2: ellipseColor

参数 type 有 3 个取值："full","lower","upper"，分别表示绘制整个矩阵，仅绘制下三角，仅绘制上三角。diag 有 2 个取值：T,F，分别表示绘制对角线，不绘制对角线。

```
plotcorr(xc, col=colors[5*xc + 6], type = "upper")
plotcorr(xc, col=colors[5*xc + 6], type = "lower", diag = TRUE)
```

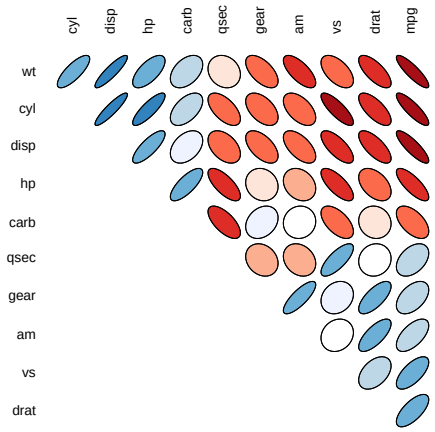


图 3: ellipseDiag1

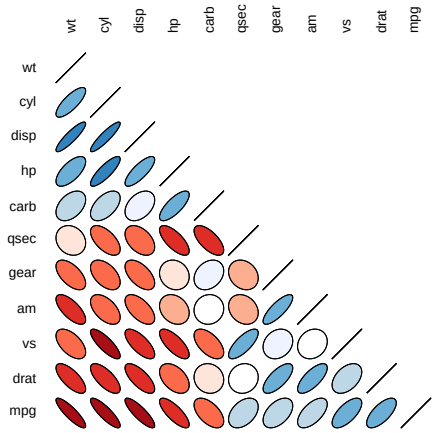


图 4: ellipseDiag2

4.3 数字

实际上，把相关矩阵列成表能给别人最充足的信息，加入参数 numbers=T 可以实现。cex 控制数字大小。

```
plotcorr(cor(mtcars[,7:11]),numbers = TRUE, type = "upper",cex=1)
```

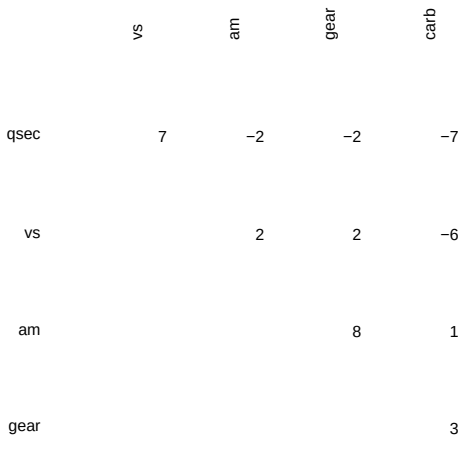


图 5: ellipseNum

Fig. 5并没有给出精确的数值，而是将原数据放大 10 倍。因为人们从一堆小数中判断大小是困难的，况且多数情况下，我们只需要用相关性分析图帮助我们判断线性相关性的存在趋势，并不需要具体数值。这种极简风格实用干练。

5 plotcov(pcaPP)

5.1 基础

程序包 pcaPP 的主要功能是 Robust PCA by Projection Pursuit，作者为 Peter Filzmoser, Heinrich Fritz 和 Klaudius Kalcher。函数 plotcov() 主要用于两个相关性矩阵的比较。当然也可以用于单个相关矩阵的可视化。上三角形使用椭圆，下三角形显示具体数值，风格和 Fig. 5类似。作用于相关系数矩阵

```
install.packages("pcaPP")
library(pcaPP)
data(Capm,package="Ecdat")
plotcov(cor(Capm),method1="correlation")
```

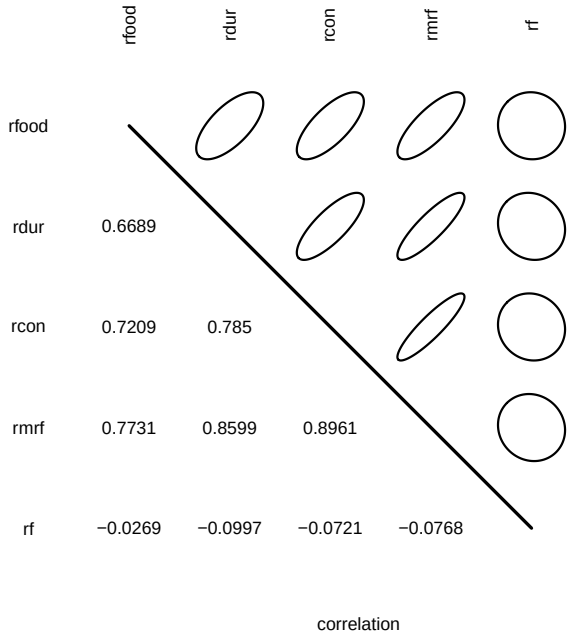


图 6: plotcov

5.2 2 个相关矩阵的比较

分别使用 PCAgrid 和 PCAproj 函数计算鲁棒协方差矩阵 (robust covariance matrix)，对比差异。

```
plotcov(covPCAproj(Capm),covPCAgrid(Capm))
```

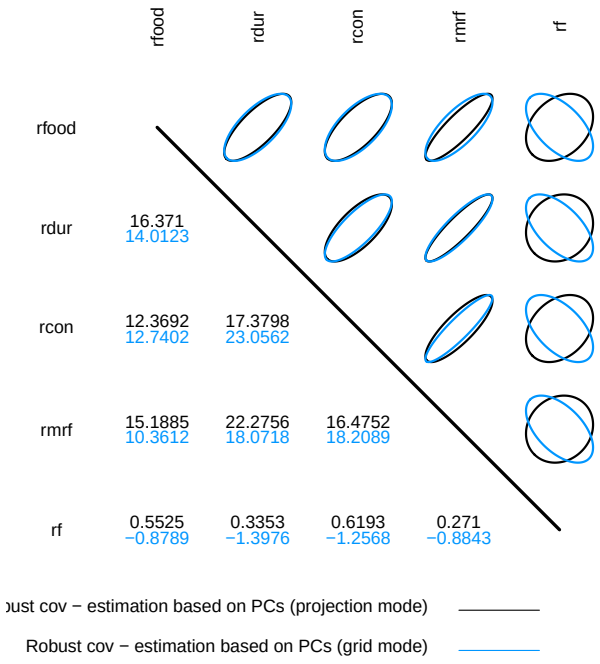


图 7: plotcovTwo

6 corrplot(corrplot)

6.1 基础

程序包 corrplot 专门用于相关性矩阵的可视化，作者魏太云。corrplot() 有着丰富的用法。默认用圆形代表相关系数，颜色表示相关性的强弱和正负，大小表示相关性的强弱。作用于相关系数矩阵

```
install.packages("corrplot")
library(corrplot)
data(mtcars)
M <- cor(mtcars)
corrplot(M)
```

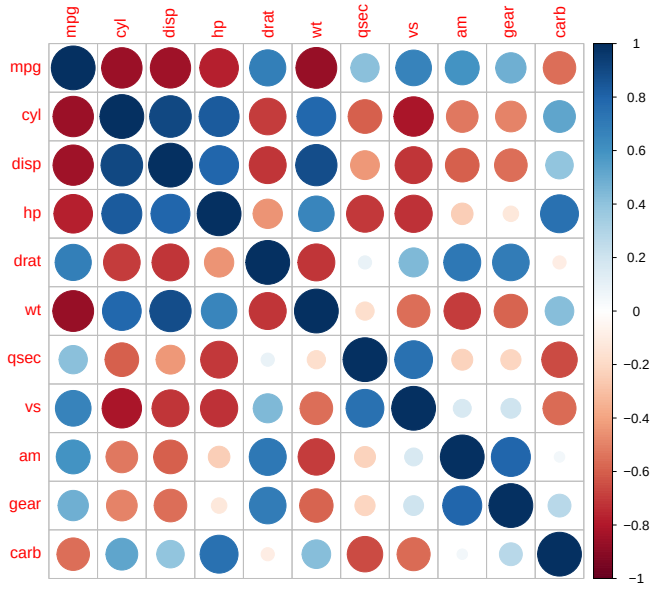


图 8: corrplot

另外，我们还可以在图上添加相关系数。

```
corrplot(M, addCoef.col="grey")
```

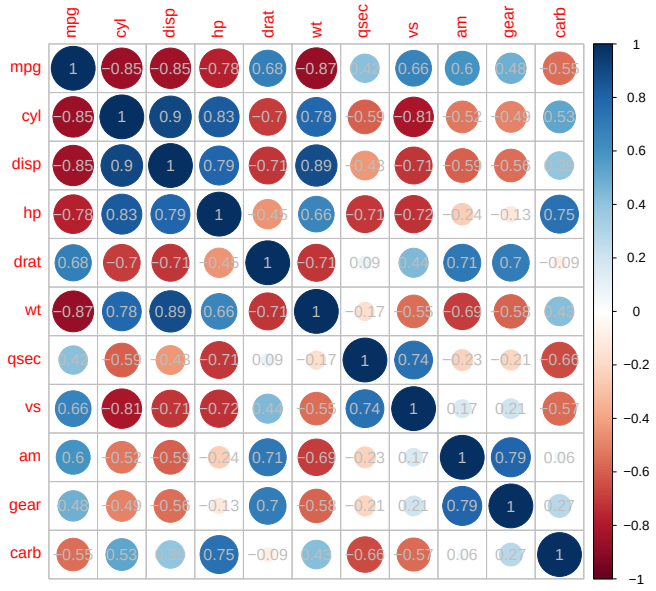


图 9: corrplotRC

6.2 顺序

Fig. 8存在的问题是颜色散乱，影响观看效果。corrplot() 提供了自动排序的功能。参数 order 共有 5 种排序方式：”original” 原始顺序，”AOE” 特征向量的角度，”FPC” 第一主成分，”hclust” 系统聚类，”alphabet” 字母顺序。对后四种方式进行对比：

```
corrplot(M, order =”AOE”); corrplot(M, order =”hclust”)
corrplot(M, order =”FPC”); corrplot(M, order =”alphabet”)
```

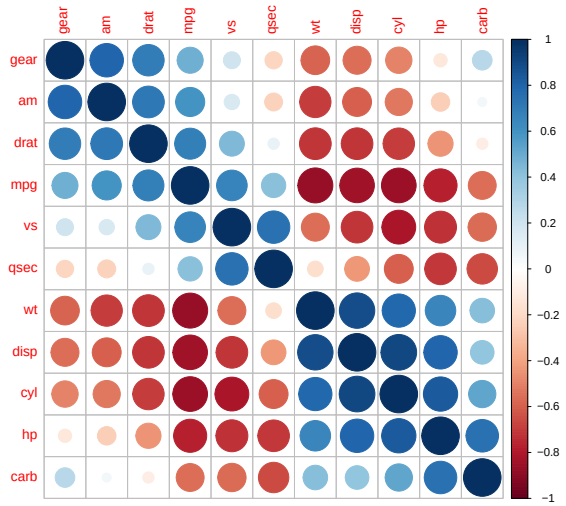


图 10: corrplotOrderAOE

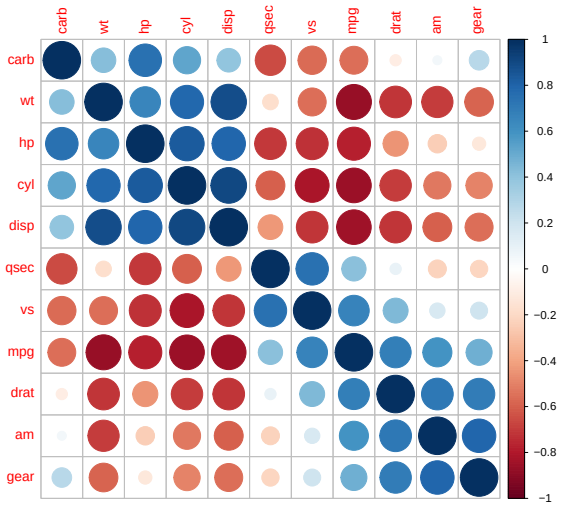


图 11: corrplotOrderhlcust

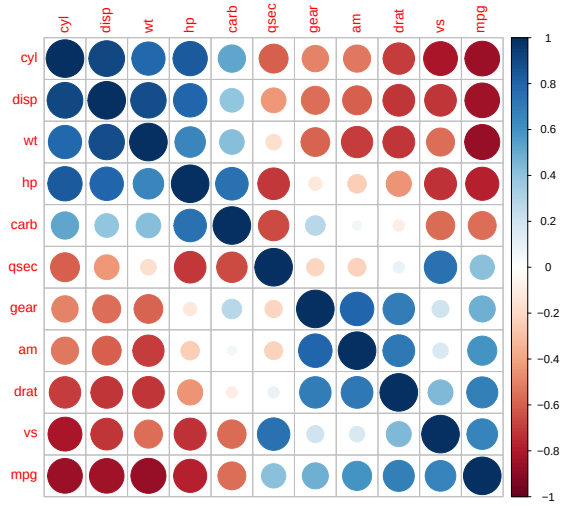


图 12: corrplotOrderFPC

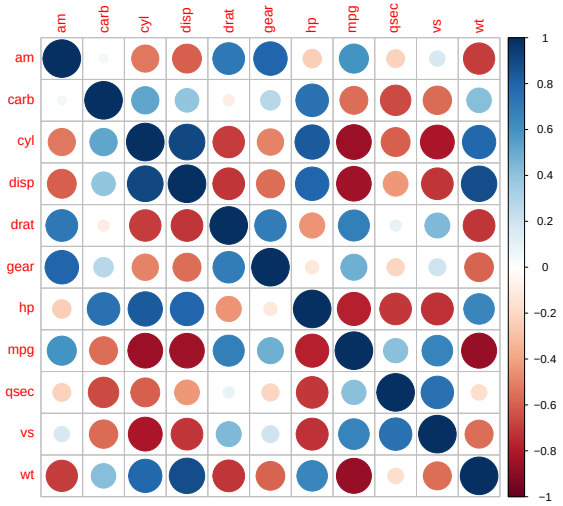


图 13: corrplotOrderalphabet

6.2.1 hclust

参数 addrect 的取值为整数 n，会根据系统聚类在图上添加 n 个方块。

参数 rect.col 可以控制方块的线条颜色。

参数 rect.lwd 控制方块的线条宽度。

参数 hclust.method 可以选择系统聚类的类间距离”ward”, ”single”, ”complete”, ”average”, ”mcquitty”, ”median” or ”centroid”。

```
corrplot(M, order=”hclust”, addrect = 2,
rect.lwd = 1)
corrplot(M, order=”hclust”, addrect = 3,
rect.col = ”red”)
corrplot(M, order=”hclust”, addrect = 4,
rect.col = ”blue”)
corrplot(M, order=”hclust”,
hclust.method=”ward”, addrect = 4)
```

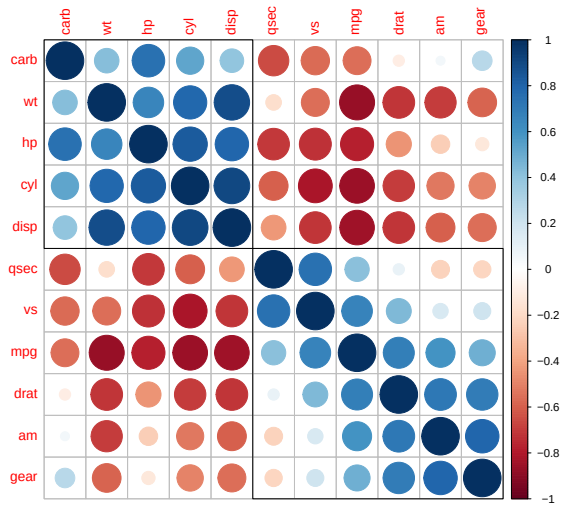


图 14: corrplotOrderhlcust1

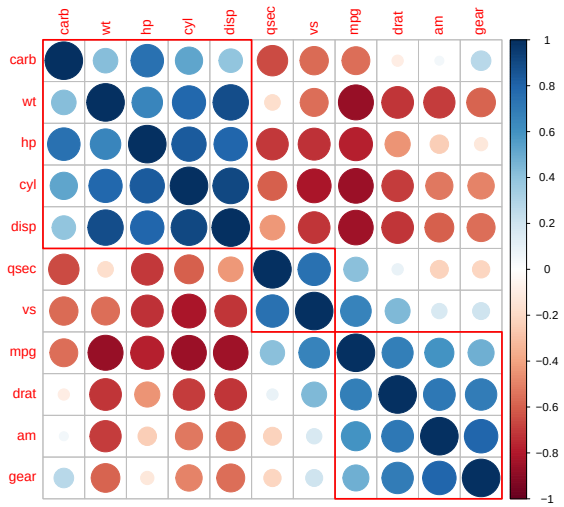


图 15: corrplotOrderhlcust2

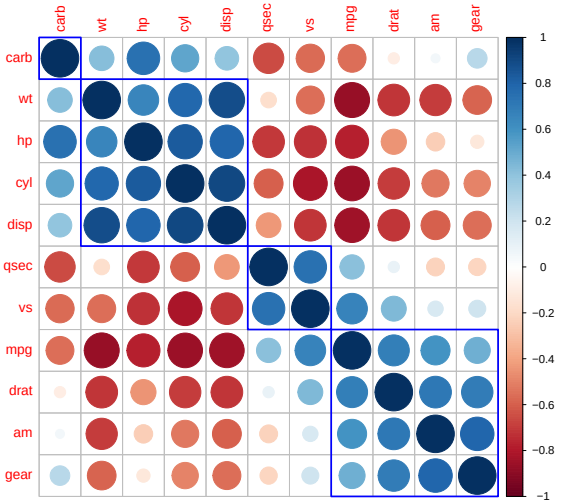


图 16: corrplotOrderhlcust3

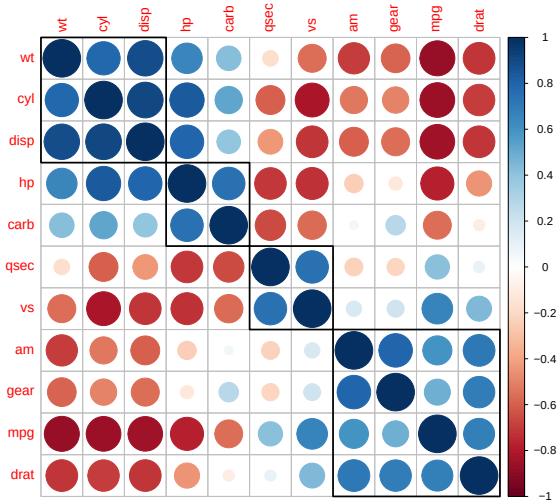


图 17: corrplotOrderhlcust4

6.3 颜色

我们可以根据自己的喜好设置图形的颜色。首先我们先预设几个调色板。

```
col1 <- colorRampPalette(c("#7F0000","red","#FF7F00","yellow","white",
"cyan", "#007FFF", "blue","#00007F"))
col2 <- colorRampPalette(c("#67001F", "#B2182B", "#D6604D", "#F4A582", "#FDDBC7",
"#FFFFFF", "#D1E5F0", "#92C5DE", "#4393C3", "#2166AC", "#053061"))
col3 <- colorRampPalette(c("red", "white", "blue"))
col4 <- colorRampPalette(c("#7F0000","red","#FF7F00","yellow","#7FFF7F",
"cyan", "#007FFF", "blue","#00007F"))
wb <- c("white","black")
par(ask = TRUE)
```

对 4 个调色板进行对比，Fig. 21是黑白色，cl.pos="n" 表示不绘制图例，outline=T 表示绘制轮廓线，否则负数的圆是没有显示的：

```
corrplot(M, order="AOE", col=col1(200))
corrplot(M, order="AOE", col=col2(200))
corrplot(M, order="AOE", col=col3(200))
corrplot(M, col = wb, order="AOE", outline=TRUE, cl.pos="n")
```

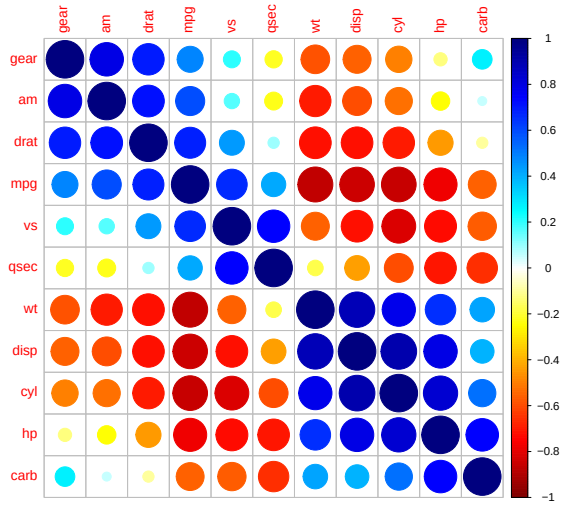


图 18: corrplotColor1

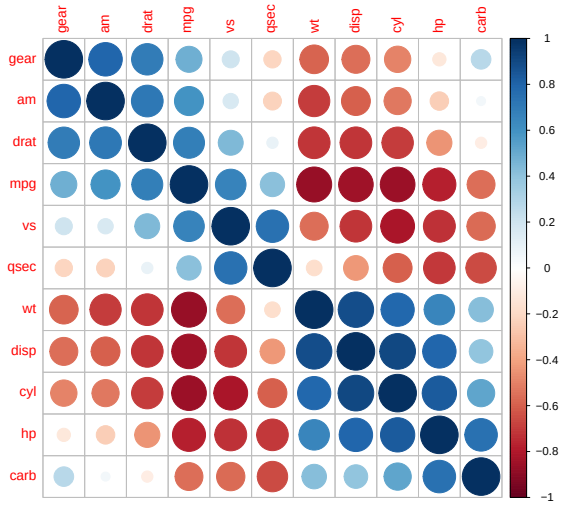


图 19: corrplotColor2

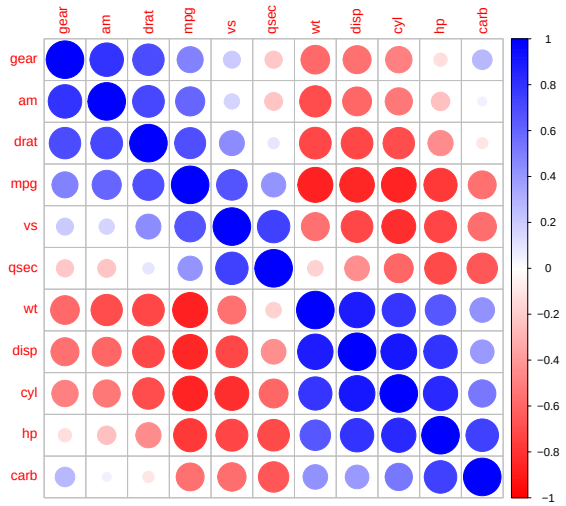


图 20: corrplotColor3

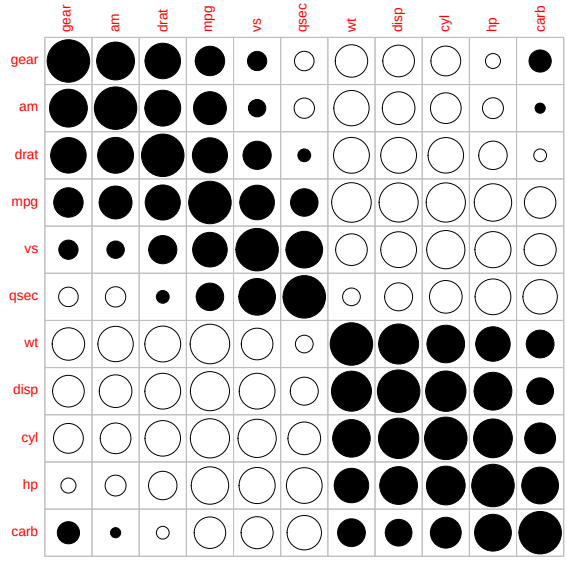


图 21: corrplotColorwb

使用参数 cl.length 控制图例的刻度数量。

```
corrplot(M, order="AOE", col=col2(20), cl.length=21)
corrplot(M, order="AOE", col=col2(20), cl.length=11)
```

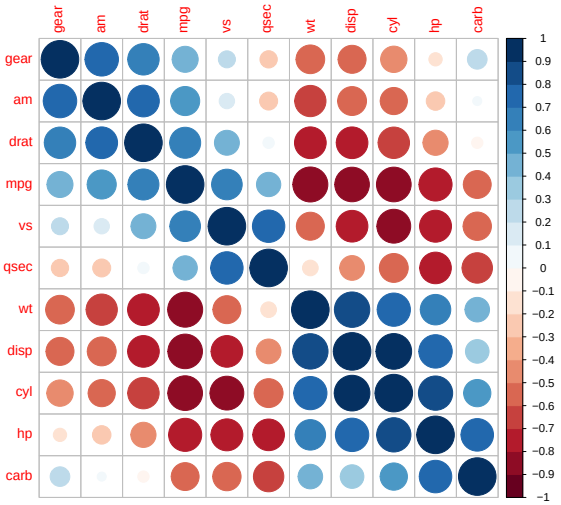


图 22: corrplotLabel1

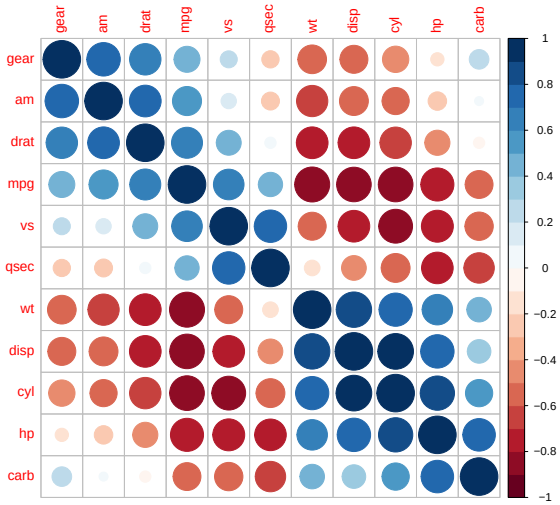


图 23: corrplotLabel2

使用 bg 控制背景颜色。

```
corrplot(M, order="AOE", col=col2(20), cl.length=21)
corrplot(M, order="AOE", col=col2(20), cl.length=11)
```

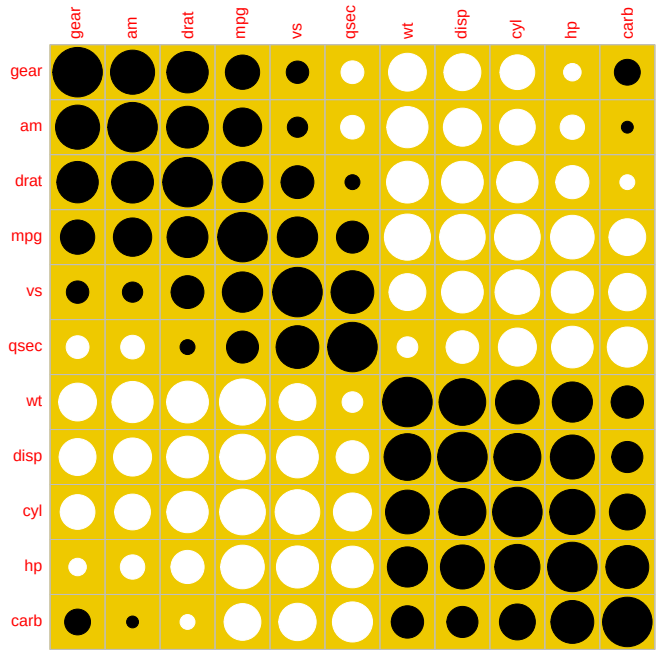


图 24: corrplotBackGround

6.4 形状与表达

使用 method 控制矩阵中出现的元素。取值有”circle” (default), ”square”, ”ellipse”, ”number”, ”pie”, ”shade” and ”color”。默认使用圆代表相关系数。

square 绘制矩形，ellipse 绘制椭圆，用法和功能与 circle 相同。

number 显示相关系数（到小数点后两位），数字颜色为黑色。

pie 绘制饼图，饼图一部分涂色另一部分留白，颜色越深，涂色面积越大，相关性越强。

shade，color，square 都是基于矩形：color 相当于热图，shade 和 square 相当于在热图的基础上进行了修改，shade 对负相关系数的矩形加上了斜线，square 改变矩形大小以表示相关系数的强弱。

```
corrplot(M, method=”square”, col=col2(200),order = ”AOE”)
corrplot(M, method=”ellipse”, col=col1(200),order = ”AOE”)
corrplot(M, method=”number”, col=”black”, cl.pos=”n”)
corrplot(M, method=”pie”, order = ”AOE”)
corrplot(M, method=”shade”, col=col3(20),order = ”AOE”)
corrplot(M, method=”color”,order = ”AOE”)
```

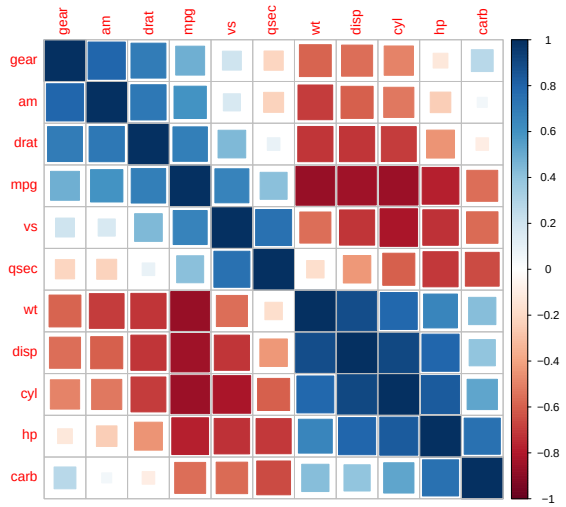


图 25: corrplotSquare

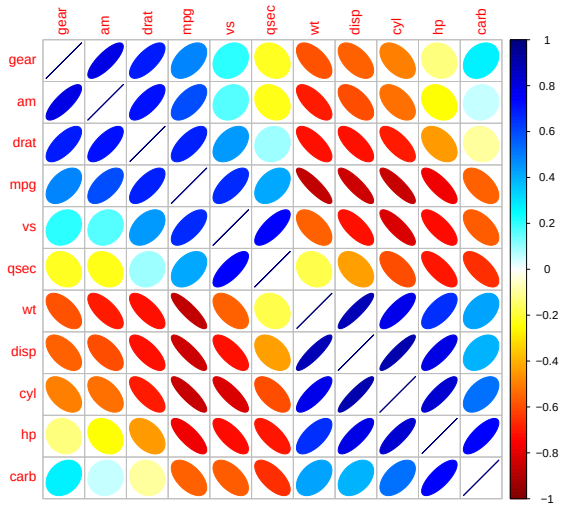
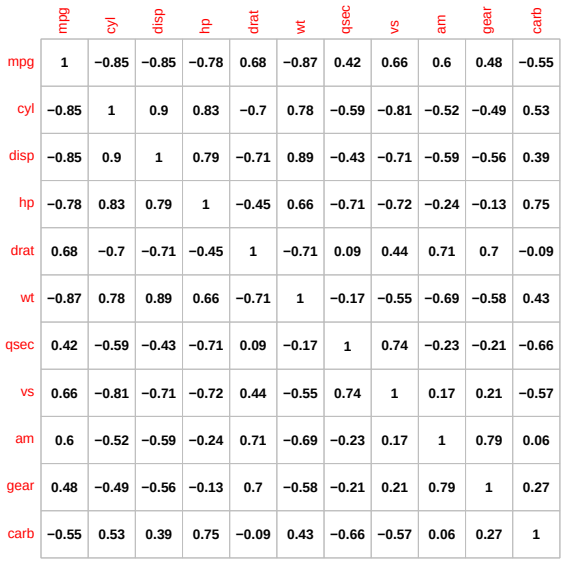


图 26: corrplotEllipse



	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
mpg	1	−0.85	−0.85	−0.78	0.68	−0.87	0.42	0.66	0.6	0.48	−0.55
cyl	−0.85	1	0.9	0.83	−0.7	0.78	−0.59	−0.81	−0.52	−0.49	0.53
disp	−0.85	0.9	1	0.79	−0.71	0.89	−0.43	−0.71	−0.59	−0.56	0.39
hp	−0.78	0.83	0.79	1	−0.45	0.66	−0.71	−0.72	−0.24	−0.13	0.75
drat	0.68	−0.7	−0.71	−0.45	1	−0.71	0.09	0.44	0.71	0.7	−0.09
wt	−0.87	0.78	0.89	0.66	−0.71	1	−0.17	−0.55	−0.69	−0.58	0.43
qsec	0.42	−0.59	−0.43	−0.71	0.09	−0.17	1	0.74	−0.23	−0.21	−0.66
vs	0.66	−0.81	−0.71	−0.72	0.44	−0.55	0.74	1	0.17	0.21	−0.57
am	0.6	−0.52	−0.59	−0.24	0.71	−0.69	−0.23	0.17	1	0.79	0.06
gear	0.48	−0.49	−0.56	−0.13	0.7	−0.58	−0.21	0.21	0.79	1	0.27
carb	−0.55	0.53	0.39	0.75	−0.09	0.43	−0.66	−0.57	0.06	0.27	1

图 27: corrplotNum

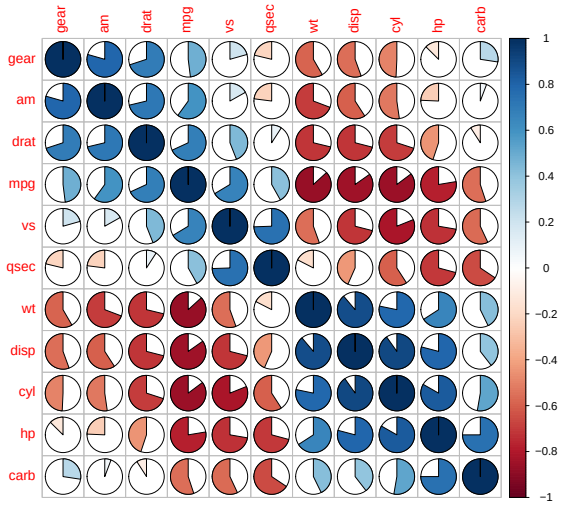


图 28: corrplotPie

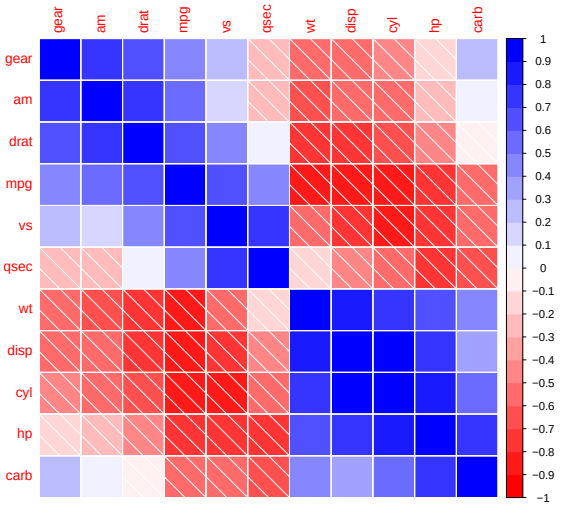


图 29: corrplotShade

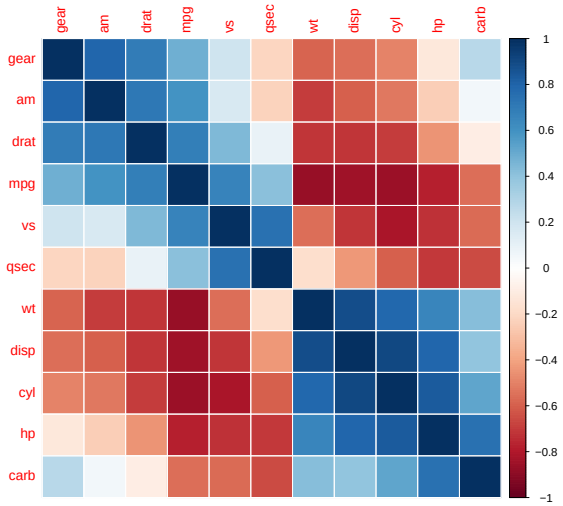


图 30: corrplotColor

6.5 上下对角线

corrplot() 提供了许多形状供我们选择，控制参数 type 可以在上三角绘制一种形状，在下三角绘制另一种形状。

比如下面的代码，在第一条命令中，type=”upper” 表示在上三角形中绘制图形，tl.pos=”d” 表示把变量名称放在对角线上。在第二条命令中，add=TRUE 表示继续在上一张图上绘制本条命令，diag=FALSE 表示

不在对角线上添加元素，tl.pos=”n” 表示不显示变量名称，cl.pos=”n” 不显示图例。

```
corrplot(M,order=”AOE”,type=”upper”,tl.pos=”d”)
corrplot(M,add=TRUE, type=”lower”, method=”number”,order=”AOE”,
diag=FALSE,tl.pos=”n”, cl.pos=”n”)
```

图 31: corrplotDiag1

这种方法比较麻烦，作者还提供了另一个函数，能够简单地实现上述命令。参数 lower，upper 控制上下对角线的形状，diag=”u” 表示对角线与上保持一致，”l” 与下保持一致，”n” 显示变量名称。

```
corrplot(M,order=”AOE”,type=”upper”,tl.pos=”d”)
corrplot(M,add=TRUE, type=”lower”, method=”number”,order=”AOE”,
diag=FALSE,tl.pos=”n”, cl.pos=”n”)
```

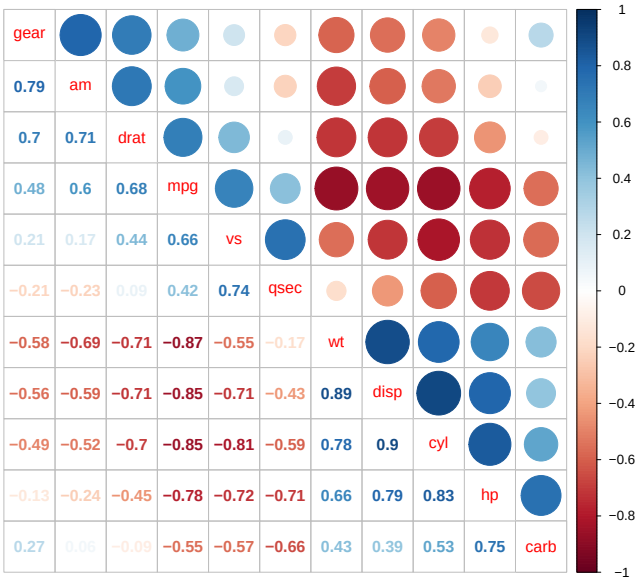


图 32: corrplotDiag2

6.6 标签与图例

参数 tl.cex 控制标签的大小。参数 tl.col 控制标签的颜色。参数 srt 控制标签倾斜的角度。

参数 tl.pos 控制变量名称的位置，”lt” 表示左部和顶部，”d” 表示在对角线上，”n” 表示不显示。

```
corrplot(M, order="AOE", tl.srt=45, tl.col="black", tl.cex=1.5)
corrplot(M, order="AOE")
```

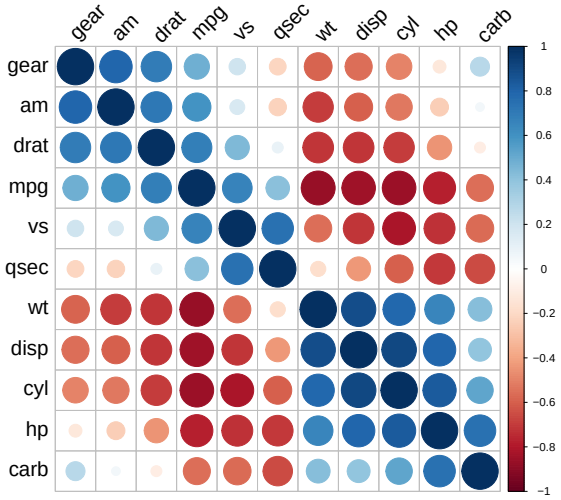


图 33: corrplotDiag3

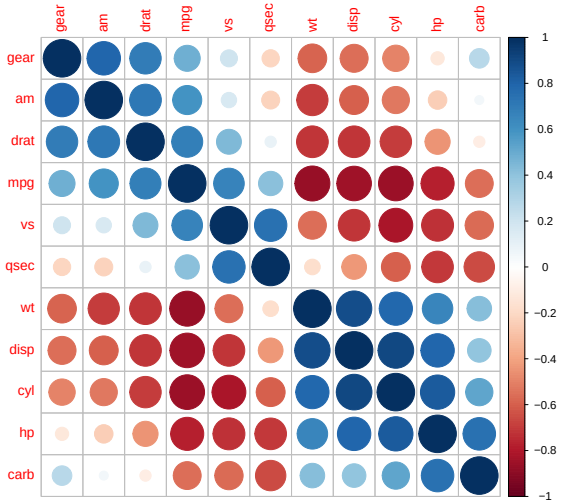


图 34: corrplotDiag4

参数 cl.ratio 控制图条的宽度。参数 cl.align 控制刻度相对于图条的位置，取值有”l” 左对齐，”c” 居中，”r” 右对齐。参数 cl.pos 控制图例的位置，取值有”r” 右边，”b” 下边（仅当下三角形时），”n” 不显示。

6.7 corrplot(arm)

程序包 arm 主要用于回归和层次模型，作者 Andrew Gelman, Yu-Sung Su et al.。这个包里也有 corrplot()，用于绘制相关矩阵。

如果你使用 library()，先加载了程序包 arm，又加载了程序包 corrplot，R 会提示你:

```
The following object is masked from ‘package:arm’:

corrplot
```

所以在使用时必须加以区分，corrplot::corrplot() 和 arm::corrplot() 分别表示两个包的函数。GGally 中的 ggcorr() 是其拓展，两者的思想是一样的，这里就不详细介绍 arm::corrplot()

```
install.packages(”arm”)
library(arm)
arm::corrplot(mtcars,color=T,abs=F)
```

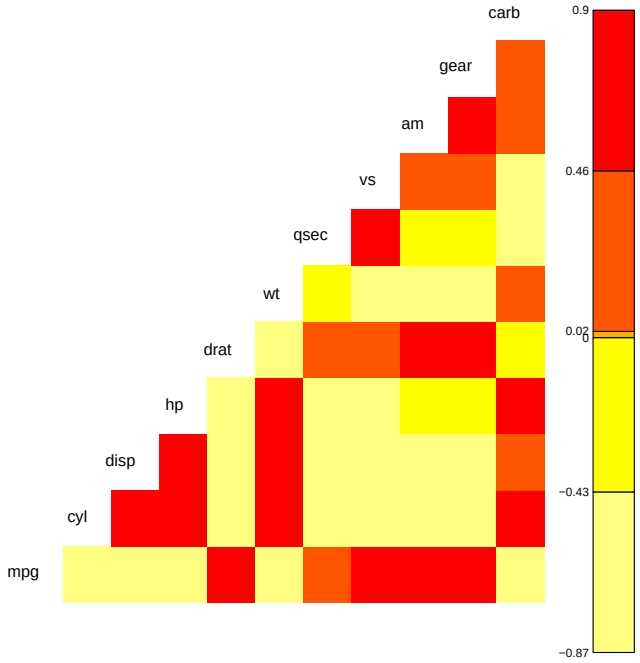


图 35: armcorrplot

7 ggcorr(GGally)

ggcorr 是 GGally 包中的相关系数矩阵可视化函数。功能介于 corrplot 与其他函数之间。

```
install.packages("GGally")
library(GGally)
nba <- read.csv("http://datasets.flowingdata.com/ppg2008.csv")
ggcorr(nba[, -1])
```

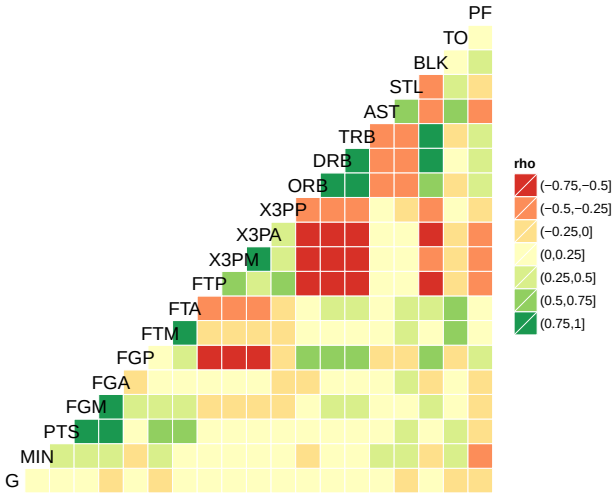


图 36: ggcorr1

使用 label=T 可以添加相关系数，label_alpha 表示相关系数半透明，cex 调整变量名称的大小。

```
ggcorr(nba[, -1], label = TRUE, label_alpha = TRUE,cex=3)
```

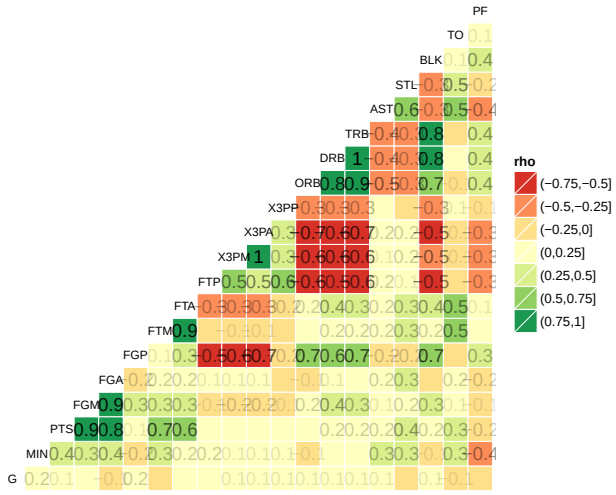


图 37: ggcorr2

geom 用于调整元素的形状，默认为”tile” 矩形，还可以有”circle” 圆形。使用圆形时，圆的大小和颜色表示相关系数的大小和强弱。max_size 控制圆最大的尺寸。angle 控制变量的角度。hjust 调整变量与元素的距离。palette 调色板，默认”RdYlGn”。

```
ggcorr(nba[, -1], geom = ”circle”, max_size = 6,
size = 3, hjust = 0.75, angle = -45, palette = ”PuOr” )
```

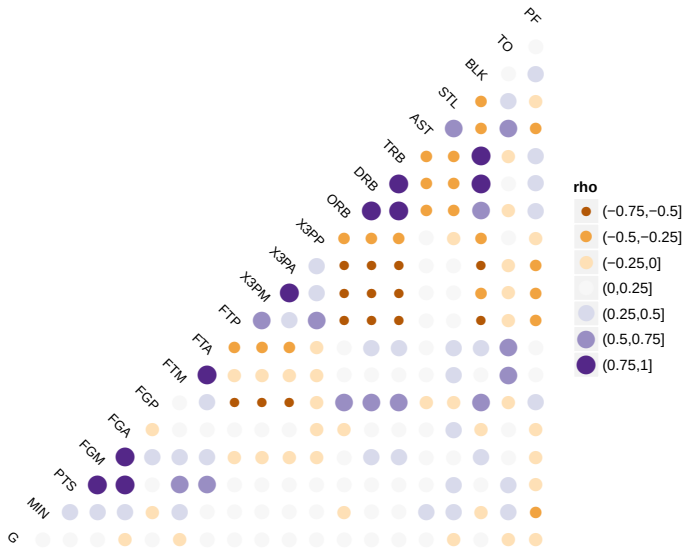


图 38: ggcorr3

8 corrgram(corrgram)

程序包 corrgram 主要功能是绘制相关性矩阵和散点图，作者 Kevin Wright。它的用法更像散点图矩阵。lower.panel，upper.panel，diag.panel 控制要绘制的图形，前两个参数的取值有 panel.pts, panel.pie, panel.shade, panel.bar, panel.ellipse, panel.conf。而 diag.panel 的取值有 panel.txt, panel.minmax, panel.density。参数 order 可以进行排序，取值有”PCA”,”GW”,”HC”,”OLO”。

```
install.packages(”corrgram”)
library(corrgram)
corrgram(iris,lower.panel=panel.shade,
upper.panel=panel.pie,text.panel=panel.txt)
```

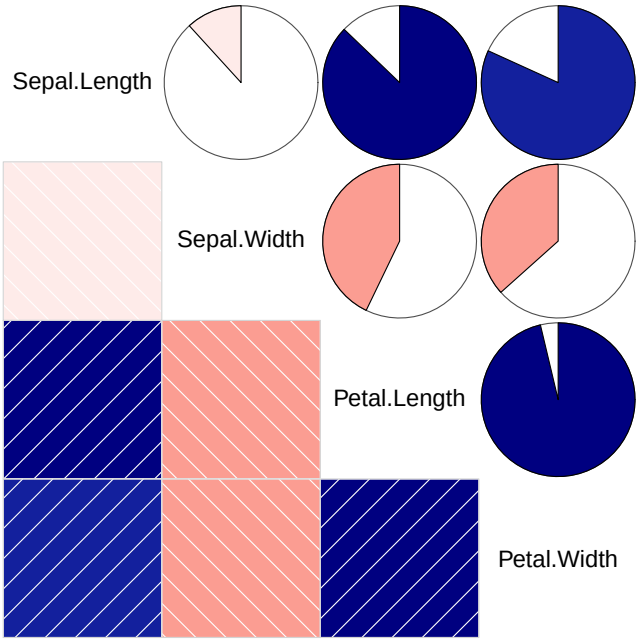


图 39: corrgram

参考文献

[1] Wikipedia：Pearson product-moment correlation coefficient

[2] Wikipedia：Rank correlation

[3] Wikipedia：Spearman's rank correlation coefficient

[4] Wikipedia：Ranking

[5] Wikipedia：Kendall rank correlation coefficient

[6] 数据铺子，豆瓣：如何在 R 中画出高效美观的相关性分析图 (二)

[7] 数据铺子，豆瓣：如何在 R 中画出高效美观的相关性分析图 (三)