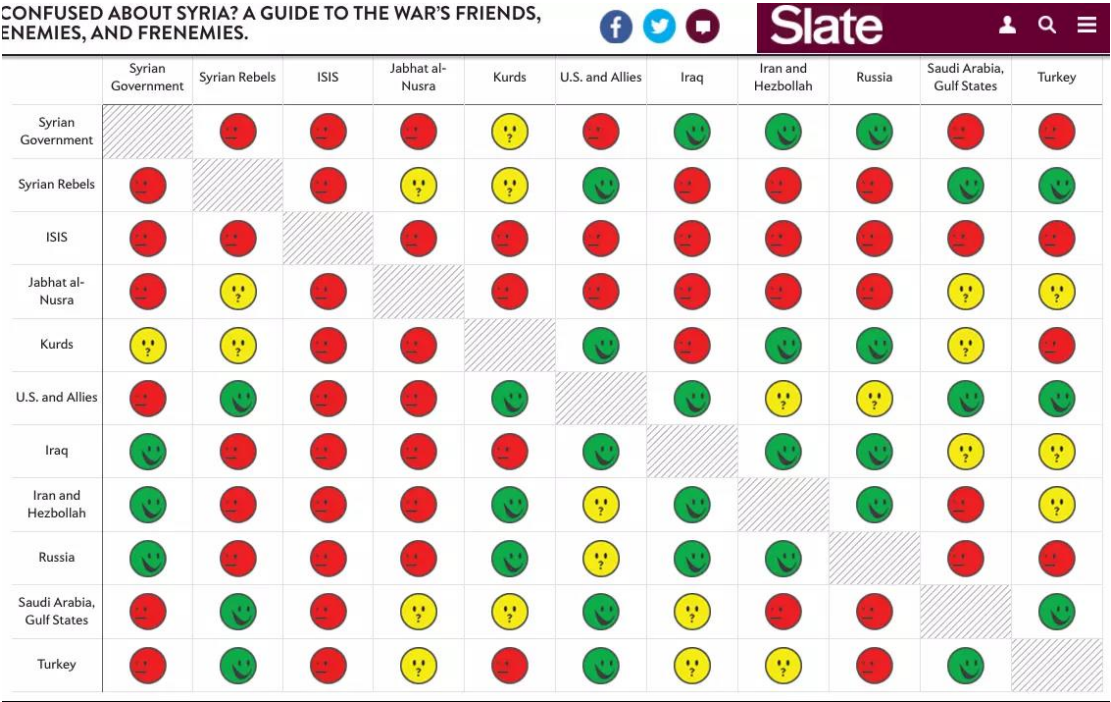


# 可视化优秀案例



图：在叙利亚，谁和谁在战斗

许多不同的种族、团体之间的关系复杂又难以理解，尤其是在叙利亚地区，当有 11 个这样相互独立的团体存在的时候，有的团体之间是合作，有的是敌意。在 11\*10 的相互关系中，通过表格的形式和熟悉的视觉表达，将数据转化成了一种简单而又易于理解的形式。图中的数据来源于各个团体之间的调查问卷、图表是一种变体的相关系数矩阵图。

# 数据可视化

## 数据整理

```
## {r warning=FALSE}
#setwd("f://大数据//大数据统计基础//第一次作业")
data = haven::read_dta("demographic_background.dta")
#出生年
age = as.numeric(data$ba002_1)
#最高学历
edu = plyr::revalue(as.character(data$bd001),c('1' = 'No formal education illiterate',
                                                '2' = 'Did not finish primary school ',
                                                '3' = 'Sishu','4' = 'Elementary school',
                                                '5' = 'Middle school','6' = 'High school',
                                                '7' = 'Vocational school',
                                                '8' = 'Two/Three Year College / Associate degree',
                                                '9' = 'Four Year College / Bachelors degree',
                                                '10' = 'Post-graduate, Masters degree',
                                                '11' = 'Post-graduate, Ph.D.'))

edu = as.factor(edu)
#婚姻状况
marriage = plyr::revalue(as.character(data$be001),
                          c('1' = 'Married with spouse present',
                            '2' = 'Married but not living with spouse temporarily',
                            '3' = 'Separated',
                            '4' = 'Divorced',
                            '5' = 'Widowed',
                            '6' = 'Never married'))

mydata = data.frame(age,edu,marriage)
colnames(mydata) = c('出生年份','教育程度','婚姻状况')
```

## 画图部分

## 前期准备

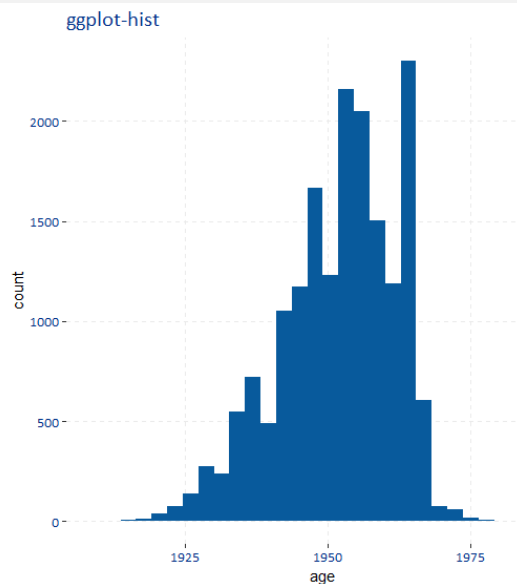
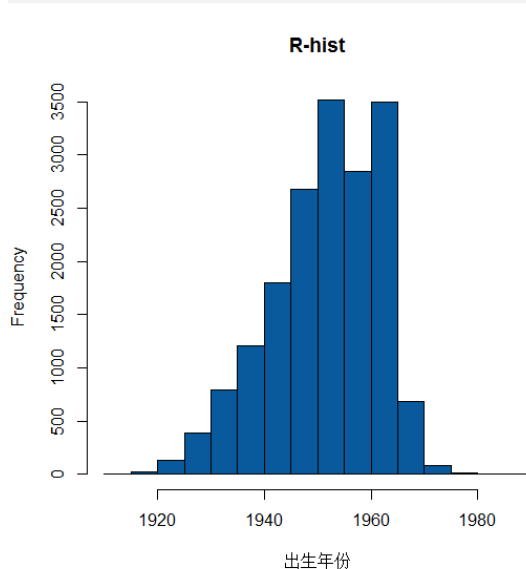
```
## {r warning=FALSE}
library(ggplot2)
library(grid)
library(gridBase)
#事先定制好要加图形的形状、颜色、主题等
mytheme<-theme_bw()+theme(legend.position="top",
                           panel.border=element_blank(),
                           panel.grid.major=element_line(linetype="dashed"),
                           panel.grid.minor=element_blank(),
                           plot.title=element_text(size=15,
                                                       colour="#003087"),
                           legend.text=element_text(size=9,colour="#003087"),
                           legend.key=element_blank(),
                           axis.text=element_text(size=10,colour="#003087"),
                           strip.text=element_text(size=12,colour="#EF0808"),
                           strip.background=element_blank())

)
mycolour_1<-"#085A9C"
```

## 直方图

```
{r warning=FALSE}

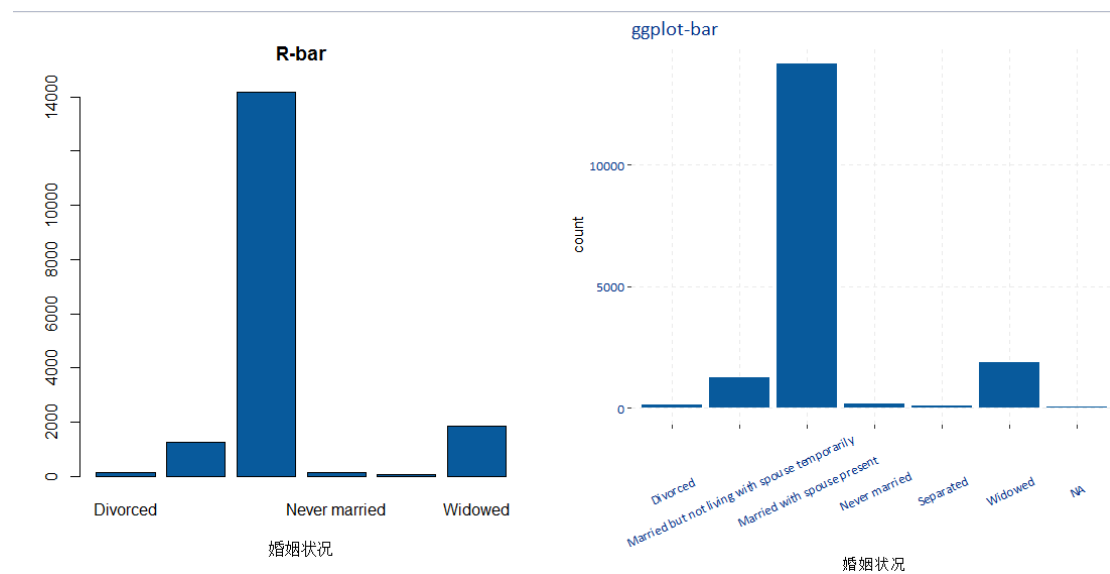
plot.new()
pic1 = grid.layout(nrow=1,ncol=2)
pic1.1 = viewport(layout.pos.col = 1, layout.pos.row = 1)
pic1.2 = viewport(layout.pos.col = 2, layout.pos.row = 1)
#第一张直方图
pushViewport(viewport(layout=pic1))
pushViewport(pic1.1)
par(new = T , fig = gridFIG())
p1 = hist(mydata$出生年份 ,col="#085A9C",main = 'R-hist', xlab = '出生年份')
popViewport()
#第二张直方图
pushViewport(pic1.2)
p2 = ggplot(mydata,aes(x=age))+geom_histogram(fill = "#085A9C") + mytheme +
  ggtitle('ggplot-hist')
print(p2, newpage = FALSE)
popViewport(1)
```



## 柱状图

```
""""柱状图"""
{r warning=FALSE}

plot.new()
pic1 = grid.layout(nrow=1,ncol=2)
pic1.1 = viewport(layout.pos.col = 1, layout.pos.row = 1)
pic1.2 = viewport(layout.pos.col = 2, layout.pos.row = 1)
#第一张柱状图
pushViewport(viewport(layout=pic1))
pushViewport(pic1.1)
par(new = T , fig = gridFIG())
p1 = barplot(table(mydata$婚姻状况 ), col = "#085A9C", main = 'R-bar', xlab = '婚姻状况')
popViewport()
#第二张柱状图
pushViewport(pic1.2)
p2 = ggplot(mydata,aes(x=婚姻状况 )) +geom_bar(fill = "#085A9C") + mytheme +
  ggtitle('ggplot-bar')+theme(axis.text.x = element_text(angle=25, vjust=0.5))
print(p2, newpage = FALSE)
popViewport(1)
```

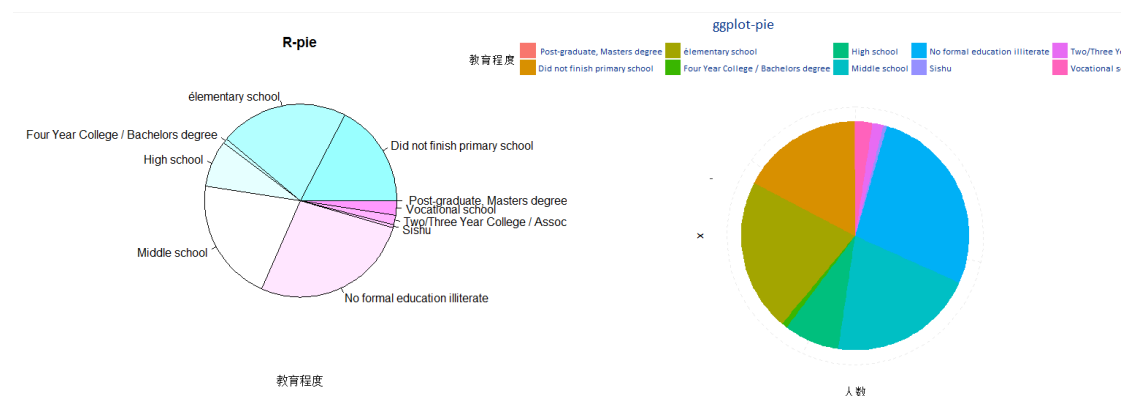


## 饼图

```

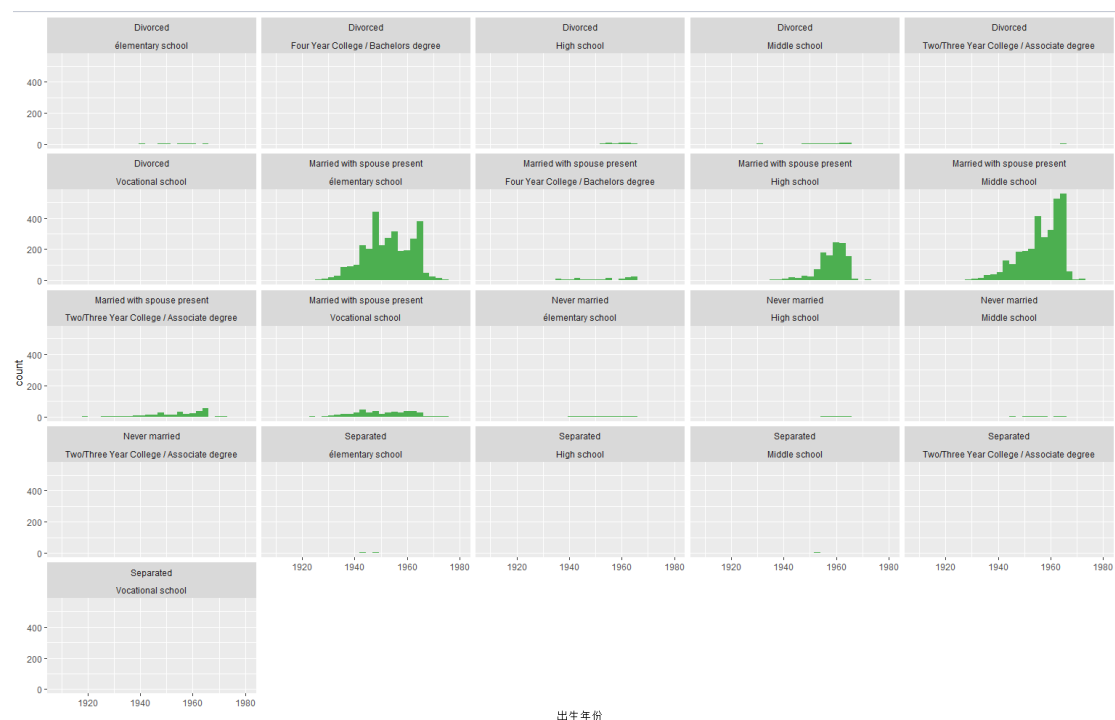
{r warning=FALSE}
mydata_pie = as.data.frame(table(mydata$教育程度))
colnames(mydata_pie)=c("教育程度","人数")
plot.new()
pic1 = grid.layout(nrow=1,ncol=2)
pic1.1 = viewport(layout.pos.col = 1, layout.pos.row = 1)
pic1.2 = viewport(layout.pos.col = 2, layout.pos.row = 1)
#第一张饼图
pushViewport(viewport(layout=pic1))
pushViewport(pic1.1)
par(new = T, fig = gridFIG())
p1 = pie(table(mydata$教育程度),main = 'R-pie',xlab='教育程度',col =
cm.colors(dplyr::n_distinct(mydata$教育程度)))
popViewport()
#第二张饼图
pushViewport(pic1.2)
p2 = ggplot(mydata_pie,aes(x="",y = 人数,fill=教育程度))+geom_bar(stat= "identity",width = 1)
+ mytheme + coord_polar(theta = "y")+
  ggtitle('ggplot-pie')+theme(axis.text.x = element_blank())
print(p2, newpage = FALSE)
popViewport(1)

```



## 分面图

```
#### 4.4.4
library(ggplot2)
mydata_new = mydata[mydata$教育程度 %in% c('Élementary school','Middle school','High school','Vocational school','Two/Three Year College / Associate degree','Four Year College / Bachelors degree') & mydata$婚姻状况 %in% c('Married with spouse present','Separated','Divorced','Never married'),]
ggplot(mydata_new, aes(x = 出生年份)) + geom_histogram(fill = "#4CAF50") + facet_wrap(~婚姻状况+教育程度)
```



## 总结

ggplot2 和 R 系统自带的画图函数从底层的统计来看是相同的，但 ggplot2 有更多参数可以调整，可以使图表有更多的变化。

相同处：

都可以进行常见的统计图表的绘制，均能客观反映数据的基本特征，保证了信息传递的正确性。

不同处：

基础绘图默认主题为空白底，无网格，ggplot2 为灰底白色网格，且在图形主题上，ggplot2 的变化更多一点，可以更个性化地定制主题。基础函数更简单，ggplot2 相比较为复杂，上手难度较大。基础函数只适合画单一的统计图表，ggplot2 可以根据图层叠加的原理画多个

统计图的叠加。

## 多元数据可视化

可用相关系数矩阵图的形式将多张图表按照矩阵布局进行组合。每一个小的图表表示了两个变量之间的相互关系，连续变量和连续变量间呈现为散点图，连续变量和类别形变量呈现箱型图，类别形变量可类别型变量为百分比堆叠图。

```
{r warning=FALSE}
library('GGally')
month = as.numeric(data$bb008_2)
mydata$出生月份 = month
ggpairs(mydata)
```

