

瀑布图

庄闪闪

目录

1 简介	1
2 三维瀑布图	2
2.1 基础版本	2
2.2 添加第四个变量	4
3 行分面的带填充的曲线图	5
3.1 基础版本	5
3.2 加入第四个变量	6
4 峰峦图	8
4.1 基础版本	8
4.2 形状波动	9
4.3 加入颜色波动	10

1 简介

瀑布图（**waterfall plot**）用于展示拥有相同的 X 轴变量数据（如相同的时间序列）、不同的 Y 轴离散型变量（如不同的类别变量）和 Z 轴数值变量，可以清晰地展示不同变量之间的数据变化关系。

2 三维瀑布图

三维瀑布图可以看成是多数据系列三维面积图。

R 中 `plot3D` 包的 `polygon3D()` 函数和 `segments3D()` 函数可以绘制三维面积图, `lines3D()` 函数可以绘制三维曲线图, 所以, 综合这几个函数可以绘制三维瀑布图, 该代码, 数据来源R 语言书可视化之美。这是一本非常棒的 R 可视化书籍, 小编预计在年底进行一次抽奖送书活动, 进行期待。

2.1 基础版本

```
library(plot3D)
library(RColorBrewer)

mydata0<-read.csv("Facting_Data.csv",check.names =FALSE)

N<-ncol(mydata0)-1
mydata<-data.frame(x=numeric(),y=numeric(),variable=character())

for(i in 1:N){
  newdata<-data.frame(spline(mydata0[,1],mydata0[,i+1],n=300,method= "natural"))
  newdata$variable<-colnames(mydata0)[i+1]
  mydata<-rbind(mydata,newdata)
}

mydata$variable<-as.numeric(mydata$variable)
group<-unique(mydata$variable)
M<-length(group)

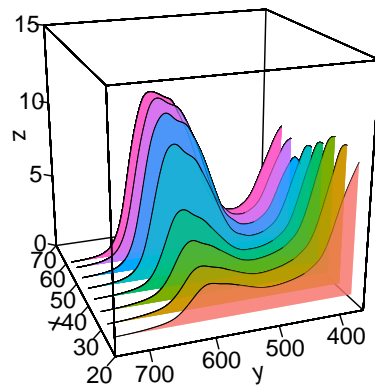
gg_color_hue <- function(n) {
  hues = seq(15, 375, length = n + 1)
  hcl(h = hues, l = 65, c = 100)[1:n]
}
```

```
colormap <- rev(gg_color_hue(M))#brewer.pal(M, 'RdYlGn')

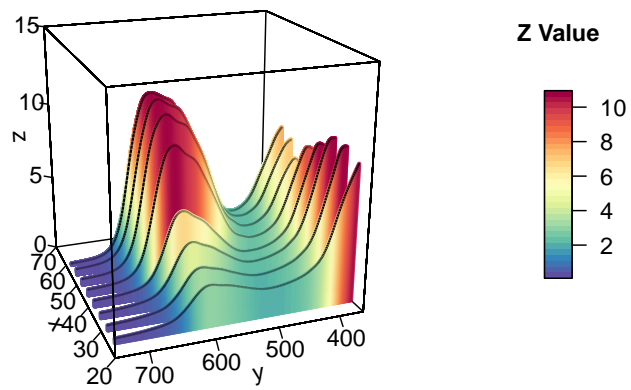
pmar <- par(mar = c(5.1, 4.1, 4.1, 6.1))

perspbox(z=as.vector(0),#add=TRUE,
         xlim=c(20,70),ylim=c(360,750),zlim=c(0,15),
         ticktype = "detailed",bty = "f",box = TRUE,
         colkey = FALSE,theta = -110, phi = 20, d=3)

for (i in 1:M){
  df0<-mydata[mydata$variable==group[i],]
  Ndf<-nrow(df0)
  df<-rbind(df0,c(df0$x[1],df0$y[Ndf],df0$variable[Ndf]))
  with(df,polygon3D(x=variable,y=x, z=y, alpha=0.6,
                   col=colormap[i],lwd = 3,add=TRUE,colkey = FALSE))
  with(df0,lines3D(x=variable,y=x, z=y,
                  lwd = 0.5,col="black",add=TRUE))
}
```



2.2 添加第四个变量



3 行分面的带填充的曲线图

使用分面图的可视化方法也可以展示瀑布图的数据信息，关于分面图可视化方法我已经在R 可视乎 | 分面一页多图介绍过。

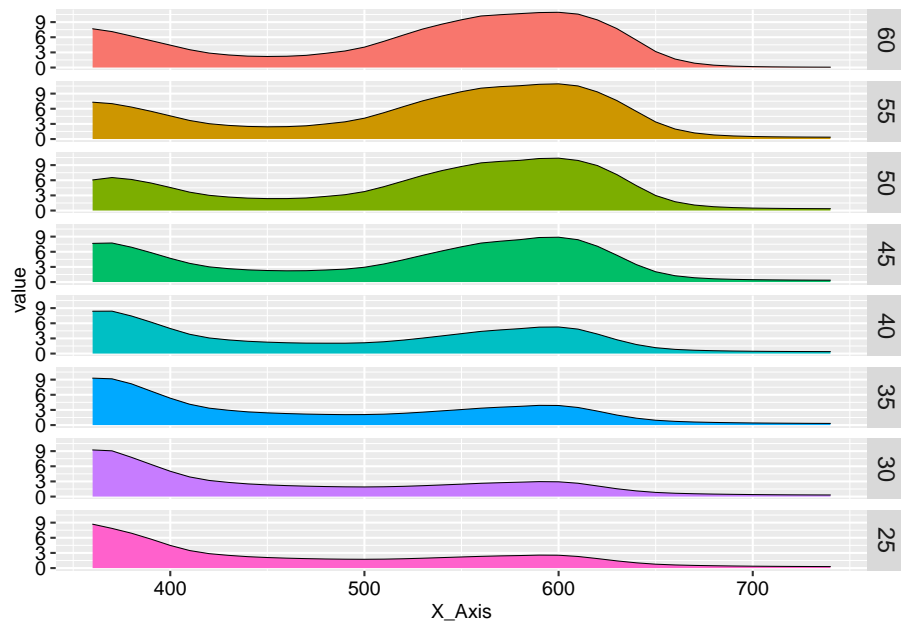
下面进行行分面的带填充的曲线图绘制，所有数据共用 X 轴坐标，每个数据类别拥有自己的 Y 轴坐标，数据类别显示在最右边。相对三维瀑布图，分面瀑布图可以更好地展示数据信息，避免不同类别之间数据重叠引起的遮挡问题，但是不能很直接地比较不同类别之间的数据差异。

3.1 基础版本

```
library(reshape2)
library(ggplot2)
mydata0<-read.csv("Facting_Data.csv",stringsAsFactors=FALSE)

colnames(mydata0)<-c("X_Axis",seq(60,25,-5))
mydata<-melt(mydata0,id.vars = "X_Axis")

ggplot(mydata,aes(X_Axis,value,fill=variable))+
  geom_area(color="black",size=0.25)+
  facet_grid(variable~.)+
  theme(
    text=element_text(size=15,face="plain",color="black"),
    axis.title=element_text(size=10,face="plain",color="black"),
    axis.text = element_text(size=10,face="plain",color="black"),
    legend.position="none"
  )
```



3.2 加入第四个变量

在上图的基础上将每个数据的 Z 变量进行颜色映射，这样有利于比较不同类别之间的数据差异。

```
library(RColorBrewer)
colormap <- colorRampPalette(rev(brewer.pal(11, 'Spectral')))(32)
mydata0<-read.csv("Facting_Data.csv",stringsAsFactors=FALSE)

N<-ncol(mydata0)-1

colnames(mydata0)<-c("X_Axis",seq(60,25,-5))

mydata<-data.frame(x=numeric(),y=numeric(),variable=character())

for (i in 1:N){
  newdata<-data.frame(spline(mydata0[,1],mydata0[,i+1],n=300,method= "natural"))
  newdata$variable<-colnames(mydata0)[i+1]
```

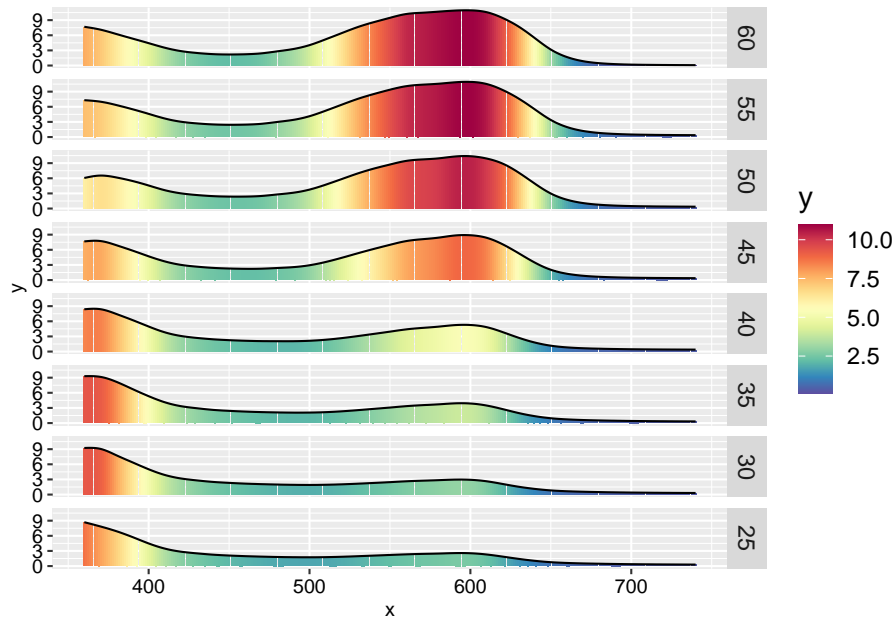
```

mydata<-rbind(mydata,newdata)
}

mydata$variable<-factor(mydata$variable,levels=seq(60,25,-5))

ggplot(mydata,aes(x,y,group=variable))+
  geom_bar(aes(fill=y),color=NA,size=0.25,stat="identity")+
  geom_line(color="black",size=0.5)+
  scale_fill_gradientn(colours=colormap)+
  facet_grid(variable~.)+
  theme(
    text=element_text(size=15,face="plain",color="black"),
    axis.title=element_text(size=10,face="plain",color="black"),
    axis.text = element_text(size=10,face="plain",color="black"),
    legend.position="right"
  )

```

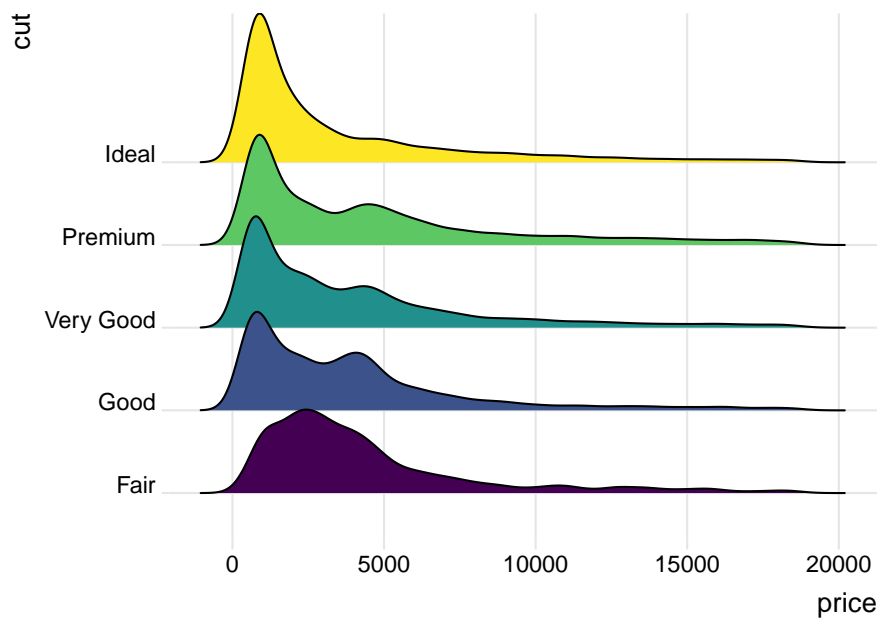


4 峰峦图

使用峰峦图也可以很好地展示瀑布图的数据信息，如图 4-7-3 所示。图 4-7-3 可以看成是在图 4-7-2(b) 的基础上将 Y 轴坐标移除，并缩小数据类别之间的距离，这样可以有效地缩小图表的占有面积，同时可以很好地展示数据的完整信息，包括不同类别之间的数据差异比较。

4.1 基础版本

```
# library
library(ggribes)
library(ggplot2)
ggplot(diamonds, aes(x = price, y = cut, fill = cut)) +
  geom_density_ridges() +
  theme_ridges() +
  theme(legend.position = "none")
```

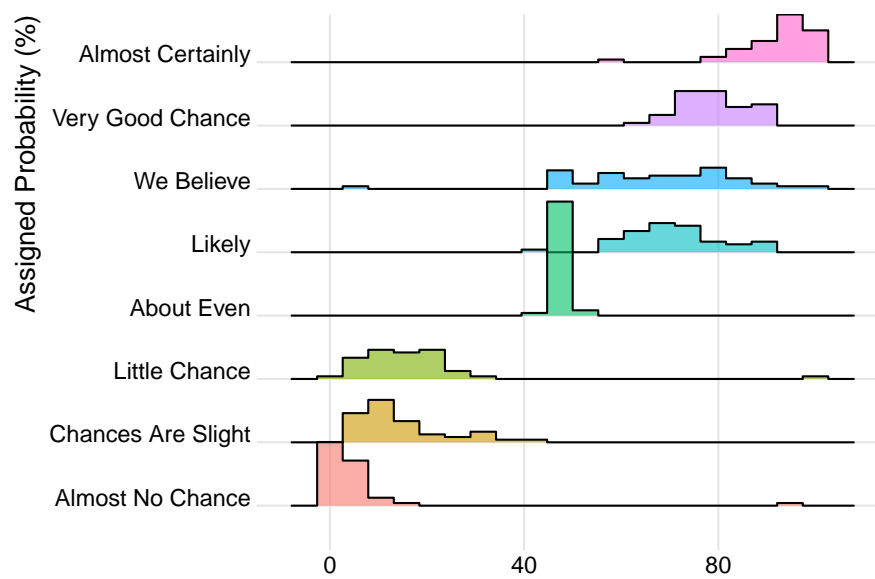


4.2 形状波动

```
# library
library(ggribes)
library(ggplot2)
library(dplyr)
library(tidyr)
library(forcats)

# Load dataset from github
data <- read.table("https://raw.githubusercontent.com/zonination/perceptions/master/probability.txt")
data <- data %>%
  gather(key="text", value="value") %>%
  mutate(text = gsub("\\\\.", " ",text)) %>%
  mutate(value = round(as.numeric(value),0)) %>%
  filter(text %in% c("Almost Certainly","Very Good Chance","We Believe","Likely","Above All Doubt"))

# Plot
data %>%
  mutate(text = fct_reorder(text, value)) %>%
  ggplot( aes(y=text, x=value, fill=text)) +
  geom_density_ridges(alpha=0.6, stat="binline", bins=20) +
  theme_ridges() +
  theme(
    legend.position="none",
    panel.spacing = unit(0.1, "lines"),
    strip.text.x = element_text(size = 8)
  ) +
  xlab("") +
  ylab("Assigned Probability (%)")
```



4.3 加入颜色波动