R Note:

Install.packates(‘ggplot2’)

Install.packates(‘RColorBrewer’)

> library(ggplot2)

Warning message:

package ‘ggplot2’ was built under R version 3.4.1

> library(RColorBrewer)

Warning message:

package ‘RColorBrewer’ was built under R version 3.4.1

[www.stackoverflow.com](http://www.stackoverflow.com)

[www.statmethods.net](http://www.statmethods.net)

exercise 1:

getwd()



> summary(mtcars)

mpg cyl disp hp

Min. :10.40 Min. :4.000 Min. : 71.1 Min. : 52.0

1st Qu.:15.43 1st Qu.:4.000 1st Qu.:120.8 1st Qu.: 96.5

Median :19.20 Median :6.000 Median :196.3 Median :123.0

Mean :20.09 Mean :6.188 Mean :230.7 Mean :146.7

3rd Qu.:22.80 3rd Qu.:8.000 3rd Qu.:326.0 3rd Qu.:180.0

Max. :33.90 Max. :8.000 Max. :472.0 Max. :335.0

drat wt qsec vs

Min. :2.760 Min. :1.513 Min. :14.50 Min. :0.0000

1st Qu.:3.080 1st Qu.:2.581 1st Qu.:16.89 1st Qu.:0.0000

Median :3.695 Median :3.325 Median :17.71 Median :0.0000

Mean :3.597 Mean :3.217 Mean :17.85 Mean :0.4375

3rd Qu.:3.920 3rd Qu.:3.610 3rd Qu.:18.90 3rd Qu.:1.0000

Max. :4.930 Max. :5.424 Max. :22.90 Max. :1.0000

am gear carb

Min. :0.0000 Min. :3.000 Min. :1.000

1st Qu.:0.0000 1st Qu.:3.000 1st Qu.:2.000

Median :0.0000 Median :4.000 Median :2.000

Mean :0.4062 Mean :3.688 Mean :2.812

3rd Qu.:1.0000 3rd Qu.:4.000 3rd Qu.:4.000

Max. :1.0000 Max. :5.000 Max. :8.000

> subset(mtcars, mpg > 30 & hp > 100)

> subset(mtcars, mpg < 14 | disp > 390)

你需要安装并加载 knitr 包，以便使用 **KNIT HTML** 按钮。在 RStudio 控制台中运行以下命令，以安装并加载 knitr。

install.packages('knitr', dependencies = T)

library(knitr)

> mtcars$wt

[1] 2.620 2.875 2.320 3.215 3.440 3.460 3.570 3.190 3.150 3.440 3.440

[12] 4.070 3.730 3.780 5.250 5.424 5.345 2.200 1.615 1.835 2.465 3.520

[23] 3.435 3.840 3.845 1.935 2.140 1.513 3.170 2.770 3.570 2.780

> cond <- mtcars$wt < 3

> cond

[1] TRUE TRUE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE

[12] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE TRUE TRUE FALSE

[23] FALSE FALSE FALSE TRUE TRUE TRUE FALSE TRUE FALSE TRUE

> mtcars$weight\_class <- ifelse(cond, 'light', 'average')

> mtcars$weight\_class

[1] "light" "light" "light" "average" "average" "average" "average"

[8] "average" "average" "average" "average" "average" "average" "average"

[15] "average" "average" "average" "light" "light" "light" "light"

[22] "average" "average" "average" "average" "light" "light" "light"

[29] "average" "light" "average" "light"

> cond <- mtcars$wt > 3.5

> mtcars$weight\_class <- ifelse(cond, 'heavy', mtcars$weight\_class)

> mtcars$weight\_class

[1] "light" "light" "light" "average" "average" "average" "heavy"

[8] "average" "average" "average" "average" "heavy" "heavy" "heavy"

[15] "heavy" "heavy" "heavy" "light" "light" "light" "light"

[22] "heavy" "average" "heavy" "heavy" "light" "light" "light"

[29] "average" "light" "heavy" "light"



> setwd('C:/edwin/Document/Udacity R study note')

> getwd()

[1] "C:/edwin/Document/Udacity R study note"

> reddata <- read.csv('reddit.csv')

> str(reddata)

'data.frame': 32754 obs. of 14 variables:

$ id : int 1 2 3 4 5 6 7 8 9 10 ...

$ gender : int 0 0 1 0 1 0 0 0 0 0 ...

$ age.range : Factor w/ 7 levels "18-24","25-34",..: 2 2 1 2 2 2 2 1 3 2 ...

$ marital.status : Factor w/ 6 levels "Engaged","Forever Alone",..: NA NA NA NA NA 4 3 4 4 3 ...

$ employment.status: Factor w/ 6 levels "Employed full time",..: 1 1 2 2 1 1 1 4 1 2 ...

$ military.service : Factor w/ 2 levels "No","Yes": NA NA NA NA NA 1 1 1 1 1 ...

$ children : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...

$ education : Factor w/ 7 levels "Associate degree",..: 2 2 5 2 2 2 5 2 2 5 ...

$ country : Factor w/ 439 levels " Canada"," Canada eh",..: 394 394 394 394 394 394 125 394 394 125 ...

$ state : Factor w/ 53 levels "","Alabama","Alaska",..: 33 33 48 33 6 33 1 6 33 1 ...

$ income.range : Factor w/ 8 levels "$100,000 - $149,999",..: 2 2 8 2 7 2 NA 7 2 7 ...

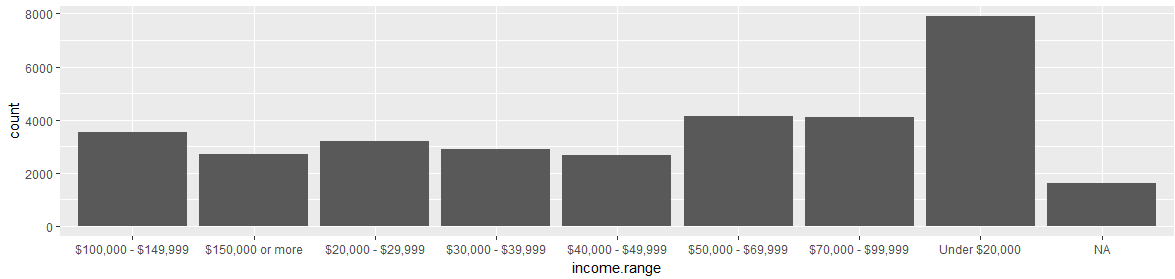
$ fav.reddit : Factor w/ 1834 levels "","'home' page (or front page if you prefer)",..: 720 691 1511 1528 188 691 1318 571 1629 1 ...

$ dog.cat : Factor w/ 3 levels "I like cats.",..: NA NA NA NA NA 2 2 2 1 1 ...

$ cheese : Factor w/ 11 levels "American","Brie",..: NA NA NA NA NA 3 3 1 10 7 ...

> qplot(data = reddata,x=age.range)

> qplot(data = reddata,x=income.range)



reddata$age.range <- ordered(reddata$age.range,levels=c("Under 18","18-24","25-34","35-44","45-54","65 or Above"))

> qplot(data = reddata,x=age.range)

Analyze Facebook data

> list.files()

[1] "~$R Note.docx" "EDA\_Course\_Materials" "EDA\_Course\_Materials.zip" "pseudo\_facebook.tsv" "R Note.docx"

[6] "reddit.csv"

>

>

> pf <- read.csv('pseudo\_facebook.tsv',sep='\t')

> str(pf)

'data.frame': 99003 obs. of 15 variables:

$ userid : int 2094382 1192601 2083884 1203168 1733186 1524765 1136133 1680361 1365174 1712567 ...

$ age : int 14 14 14 14 14 14 13 13 13 13 ...

$ dob\_day : int 19 2 16 25 4 1 14 4 1 2 ...

$ dob\_year : int 1999 1999 1999 1999 1999 1999 2000 2000 2000 2000 ...

$ dob\_month : int 11 11 11 12 12 12 1 1 1 2 ...

$ gender : Factor w/ 2 levels "female","male": 2 1 2 1 2 2 2 1 2 2 ...

$ tenure : int 266 6 13 93 82 15 12 0 81 171 ...

$ friend\_count : int 0 0 0 0 0 0 0 0 0 0 ...

$ friendships\_initiated: int 0 0 0 0 0 0 0 0 0 0 ...

$ likes : int 0 0 0 0 0 0 0 0 0 0 ...

$ likes\_received : int 0 0 0 0 0 0 0 0 0 0 ...

$ mobile\_likes : int 0 0 0 0 0 0 0 0 0 0 ...

$ mobile\_likes\_received: int 0 0 0 0 0 0 0 0 0 0 ...

$ www\_likes : int 0 0 0 0 0 0 0 0 0 0 ...

$ www\_likes\_received : int 0 0 0 0 0 0 0 0 0 0 ...

> names(pf)

[1] "userid" "age" "dob\_day" "dob\_year" "dob\_month"

[6] "gender" "tenure" "friend\_count" "friendships\_initiated" "likes"

[11] "likes\_received" "mobile\_likes" "mobile\_likes\_received" "www\_likes" "www\_likes\_received"

install.packages('ggthemes', dependencies = TRUE)   
library(ggthemes)

scale\_x\_discrete() 图层从 ggplot2 版本 2.0 开始被弃用。你可以改用 scale\_x\_continuous() 获得断点

分面的写法：

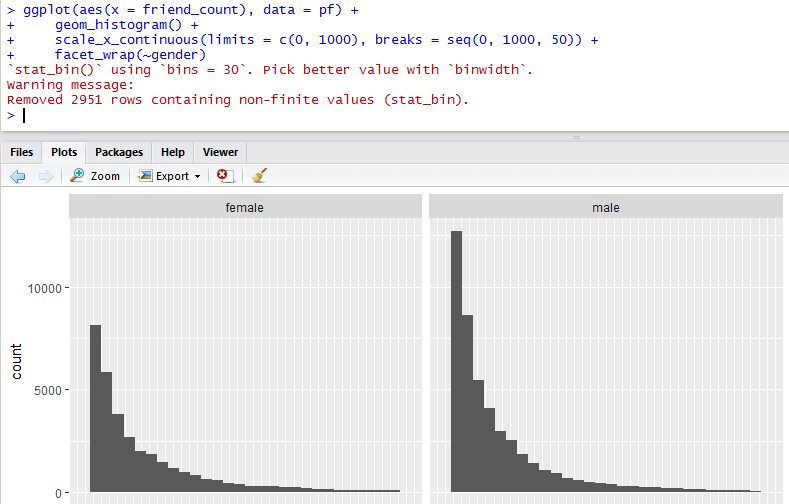
ggplot(data = pf, aes(x = dob\_day)) +   
  geom\_histogram(binwidth = 1) +   
  scale\_x\_continuous(breaks = 1:31) +   
  facet\_wrap(~dob\_month)

qplot(x=friend\_count,data=pf,xlim=c(0,3000))

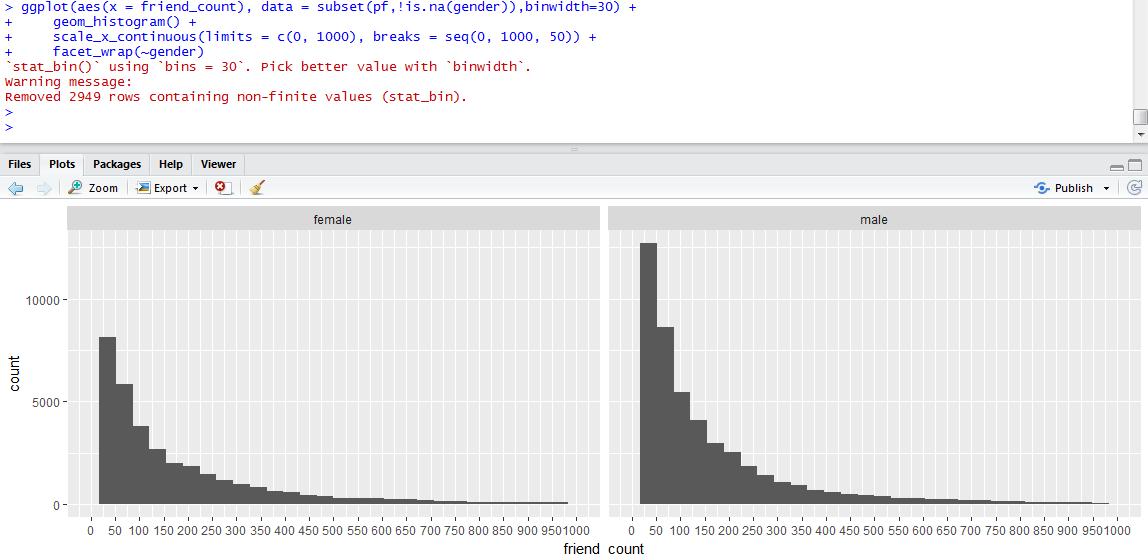
qplot(x=friend\_count,data=pf,binwidth=30)+

scale\_x\_continuous(limits = c(0, 1000),breaks= seq(0,1000,50))

ggplot(aes(x = friend\_count), data = subset(pf,!is.na(gender)),binwidth=30) +   
  geom\_histogram() +   
  scale\_x\_continuous(limits = c(0, 1000), breaks = seq(0, 1000, 50)) +   
  facet\_wrap(~gender)



去掉NA的数据



> table(pf$gender)

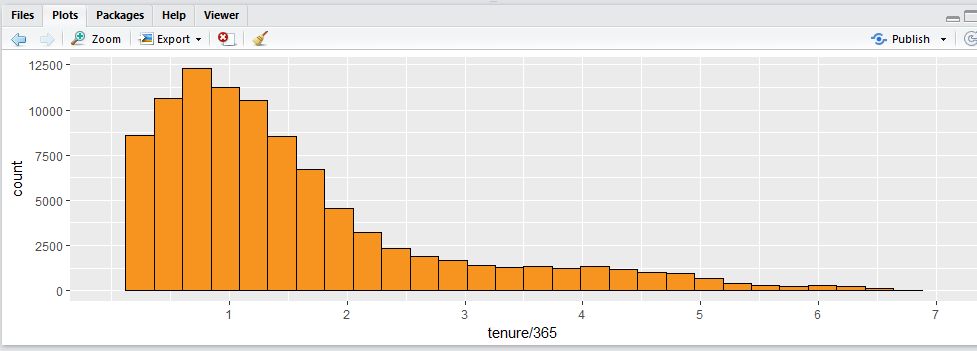
female male

40254 58574

ggplot(aes(x = tenure), data = pf) +   
   geom\_histogram(binwidth = 30, color = 'black', fill = '#099DD9')

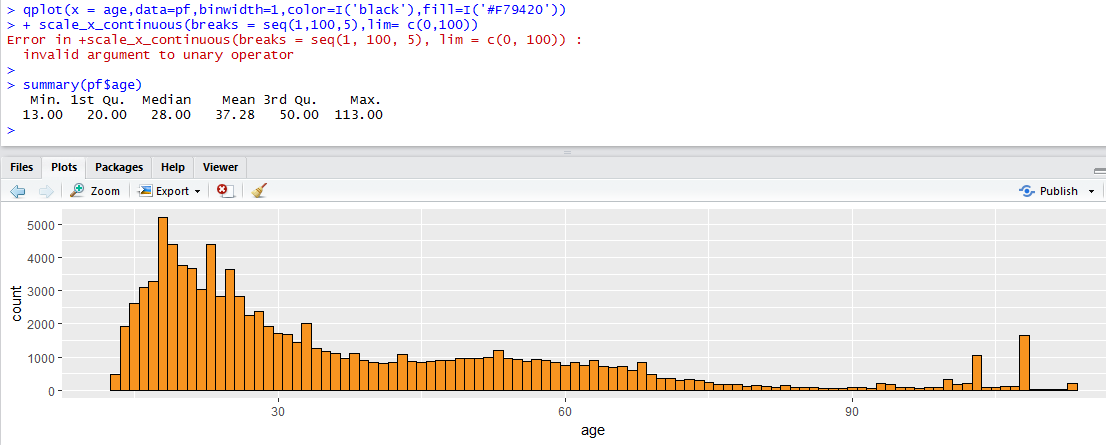
qplot(x = tenure/365,data=pf,color=I('black'),fill=I('#F79420'))

+ scale\_x\_continuous(breaks = seq(1,7,1),lim= c(0,7))



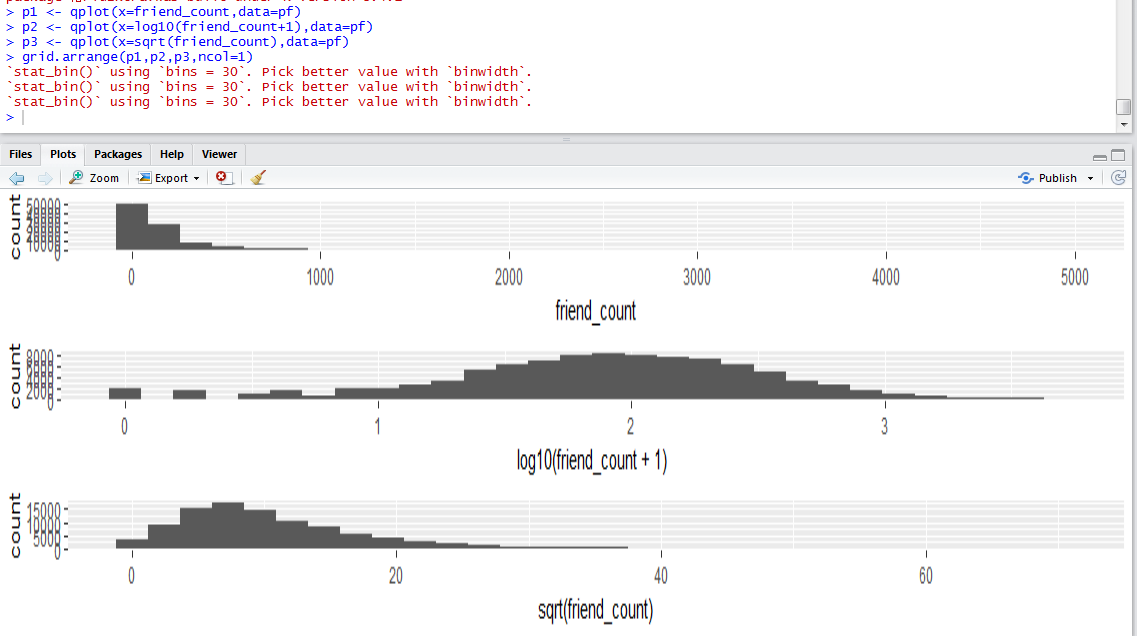
qplot(x = age,data=pf,binwidth=5,color=I('black'),fill=I('#F79420'))

+ scale\_x\_continuous(breaks = seq(1,100,5),lim= c(0,100))

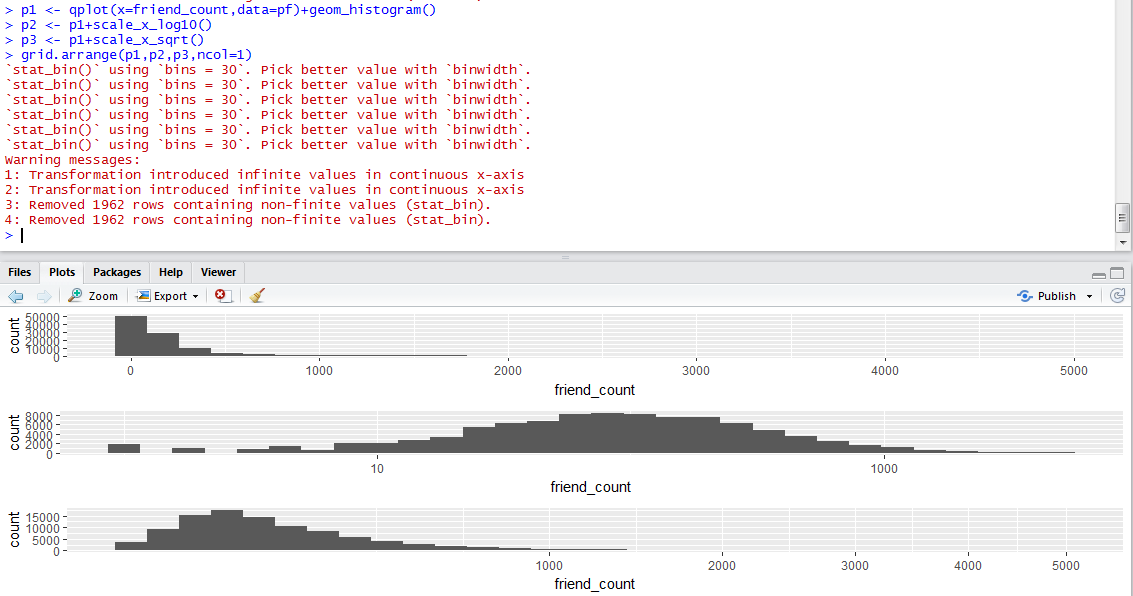


**显示三个图:**

install.packages('gridExtra')   
library(gridExtra)



**或者:**

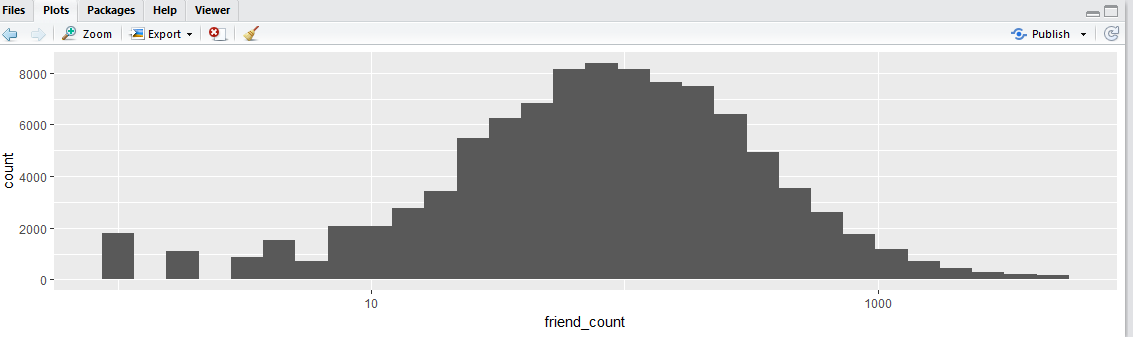


> logresult <- ggplot(aes(x=friend\_count),data=pf)+

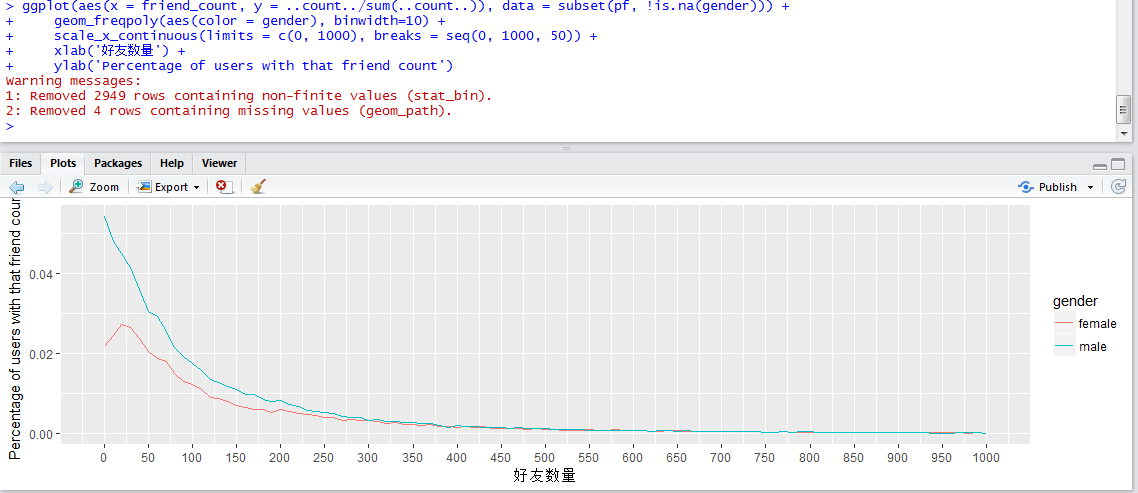
+ geom\_histogram()+

+ scale\_x\_log10()

> grid.arrange(logresult)



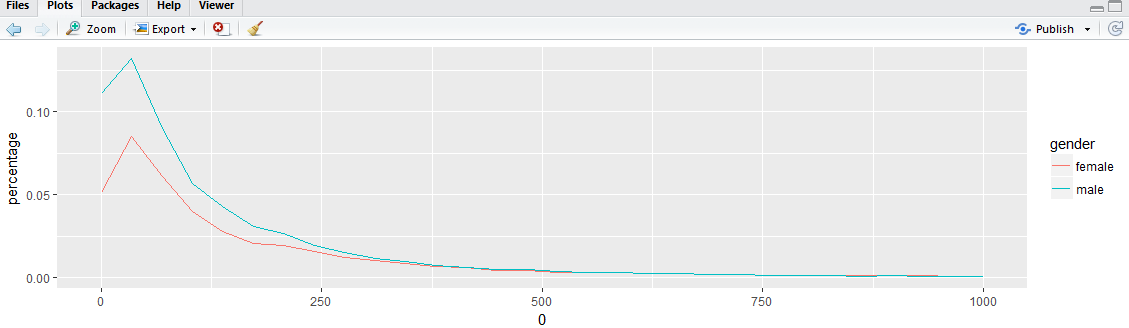
ggplot(aes(x = friend\_count, y = ..count../sum(..count..)), data = subset(pf, !is.na(gender))) +   
  geom\_freqpoly(aes(color = gender), binwidth=10) +   
  scale\_x\_continuous(limits = c(0, 1000), breaks = seq(0, 1000, 50)) +   
  xlab('好友数量') +   
  ylab('Percentage of users with that friend count')



> qplot(x=friend\_count,y=..count../sum(..count..),data=subset(pf,!is.na(gender)),binswidth=10,xlab='friend count',ylab = 'percentage',geom = 'freqpoly',color=gender)+

+ scale\_x\_continuous(seq(0,1000,50),lim=c(0,1000)

+ )



> by(pf$www\_likes,pf$gender,sum)

pf$gender: female

[1] 3507665

-----------------------------------------------------------------------------------------------------

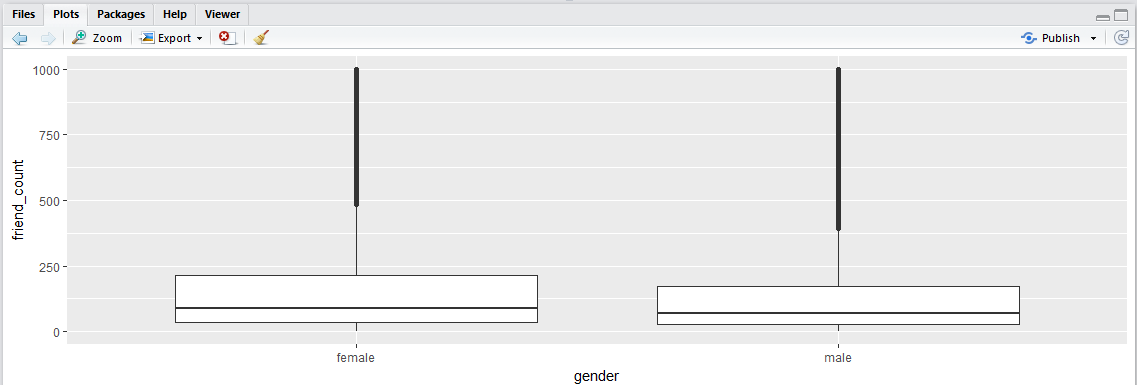
pf$gender: male

[1] 1430175

**Boxplot:**

qplot(x=gender,y=friend\_count,data=subset(pf,!is.na(gender)),geom = 'boxplot')

qplot(x=gender,y=friend\_count,data=subset(pf,!is.na(gender)),geom = 'boxplot',ylim = c(0,1000))



qplot(x=gender,y=friend\_count,data=subset(pf,!is.na(gender)),geom = 'boxplot')

+ scale\_y\_continuous(lim = c(0,1000))

> pf$mobile\_check\_in <- ifelse(pf$mobile\_likes > 0 ,1,0)

> pf$mobile\_check\_in <- factor(pf$mobile\_check\_in) 转换值

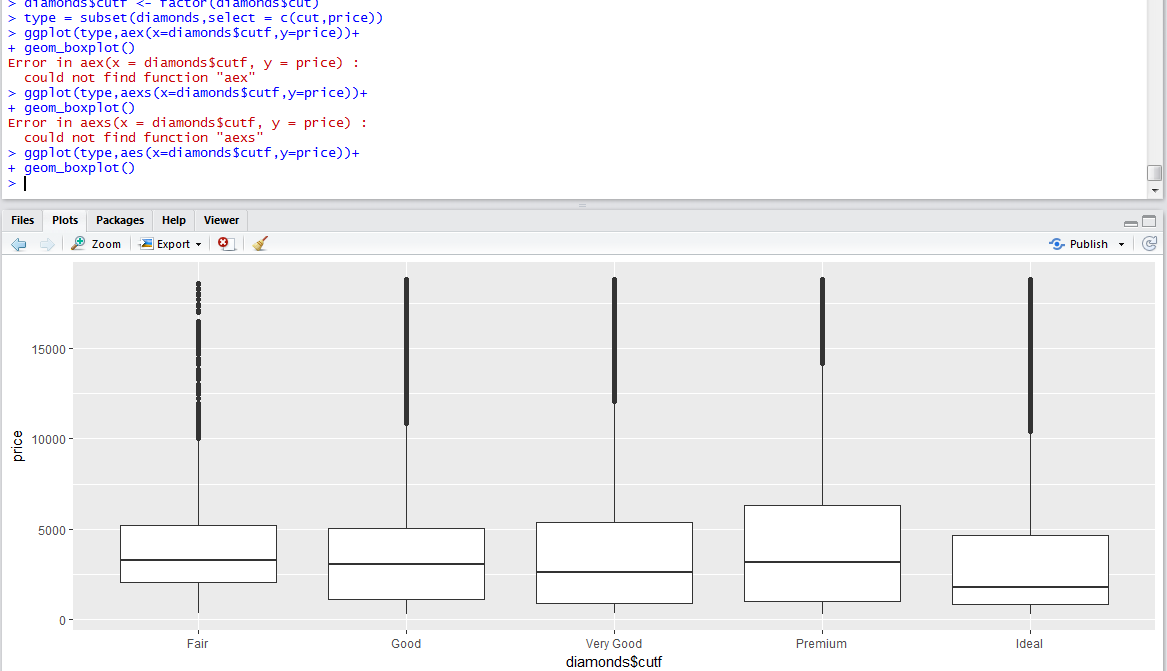
> summary(pf$mobile\_check\_in)

0 1

35056 63947

sum(pf$mobile\_check\_in == 1)/length(pf$mobile\_check\_in)

[1] 0.6459097



> by(diamonds$price,diamonds$cut,summary)

diamonds$cut: Fair

Min. 1st Qu. Median Mean 3rd Qu. Max.

337 2050 3282 4359 5206 18574

-------------------------------------------------------------------------------------------------------

diamonds$cut: Good

Min. 1st Qu. Median Mean 3rd Qu. Max.

327 1145 3050 3929 5028 18788

-------------------------------------------------------------------------------------------------------

diamonds$cut: Very Good

Min. 1st Qu. Median Mean 3rd Qu. Max.

336 912 2648 3982 5373 18818

-------------------------------------------------------------------------------------------------------

diamonds$cut: Premium

Min. 1st Qu. Median Mean 3rd Qu. Max.

326 1046 3185 4584 6296 18823

-------------------------------------------------------------------------------------------------------

diamonds$cut: Ideal

Min. 1st Qu. Median Mean 3rd Qu. Max.

326 878 1810 3458 4678 18806

**Subset 与四分位数：**

> subdata = subset(diamonds,diamonds$color==J)

Error in eval(e, x, parent.frame()) : object 'J' not found

> subdata = subset(diamonds,diamonds$color=="J")

> quantile(subset(subdata)$price)

0% 25% 50% 75% 100%

335.0 1860.5 4234.0 7695.0 18710.

> IQR(subset(diamonds, color=="J")$price)

[1] 5834.5

> IQR(subset(diamonds, color=="D")$price)

[1] 3302.5

散点图：  
散点图的等效 ggplot 语法：   
ggplot(aes(x = age, y = friend\_count), data = pf) +   
  geom\_point()

你还可以使用以下代码读入数据：   
read.delim('pseudo\_facebook.tsv')   
  
散点图的等效 ggplot 语法：   
ggplot(aes(x = age, y = friend\_count), data = pf) +   
  geom\_point()

Ggplot过度绘制：xlim分层

ggplot(aes(x=age,y=friend\_count),data=pf)+geom\_point(alpha=1/20)+xlim(13,90)

ggplot(aes(x=age,y=friend\_count),data=pf)+geom\_jitter(alpha=1/20)+xlim(13,90)

练习：

> getwd()

[1] "C:/Users/jiange7/Documents"

> setwd('C:/edwin/Document/Udacity R study note')

> facedata <- read.csv('pseudo\_facebook.tsv',sep = '\t')

> str(facedata)

'data.frame': 99003 obs. of 15 variables:

$ userid : int 2094382 1192601 2083884 1203168 1733186 1524765 1136133 1680361 1365174 1712567 ...

$ age : int 14 14 14 14 14 14 13 13 13 13 ...

$ dob\_day : int 19 2 16 25 4 1 14 4 1 2 ...

$ dob\_year : int 1999 1999 1999 1999 1999 1999 2000 2000 2000 2000 ...

$ dob\_month : int 11 11 11 12 12 12 1 1 1 2 ...

$ gender : Factor w/ 2 levels "female","male": 2 1 2 1 2 2 2 1 2 2 ...

$ tenure : int 266 6 13 93 82 15 12 0 81 171 ...

$ friend\_count : int 0 0 0 0 0 0 0 0 0 0 ...

$ friendships\_initiated: int 0 0 0 0 0 0 0 0 0 0 ...

$ likes : int 0 0 0 0 0 0 0 0 0 0 ...

$ likes\_received : int 0 0 0 0 0 0 0 0 0 0 ...

$ mobile\_likes : int 0 0 0 0 0 0 0 0 0 0 ...

$ mobile\_likes\_received: int 0 0 0 0 0 0 0 0 0 0 ...

$ www\_likes : int 0 0 0 0 0 0 0 0 0 0 ...

$ www\_likes\_received : int 0 0 0 0 0 0 0 0 0 0 ...

ggplot(aes(x=age,y=friend\_count),data=facedata)+

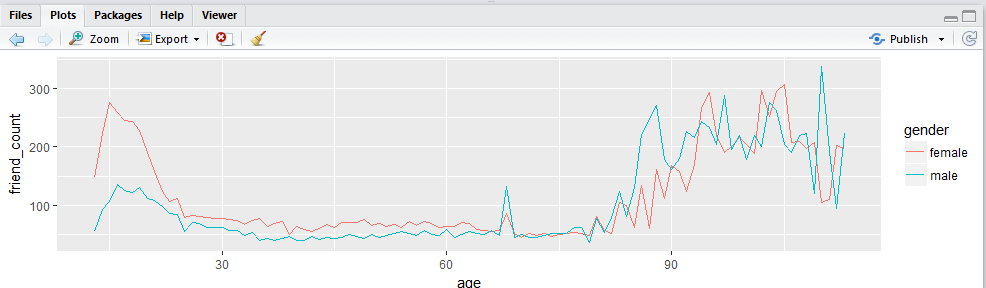
geom\_point(alpha=1/20,position=position\_jitter(h=0))+

xlim(13,90)+

coord\_trans(y = "sqrt")

ggplot(aes(x=age,y=friend\_count),data=subset(facedata,!is.na(gender)))+

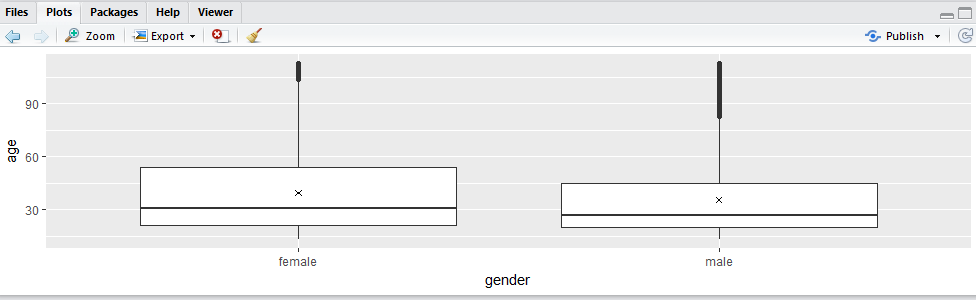
+ geom\_line(aes(color=gender),stat="summary",fun.y=median)



ggplot(aes(x=gender,y=age),data=subset(facedata,!is.na(gender)))+

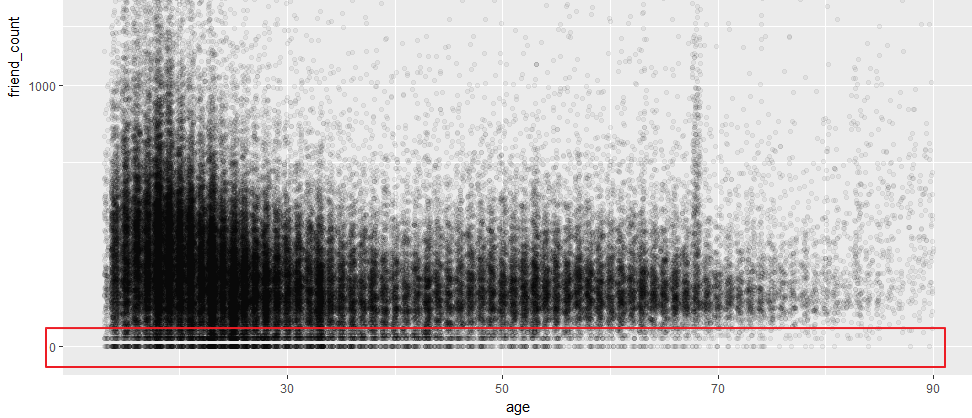
+ geom\_boxplot()+

+ stat\_summary(fun.y=mean,geom='point',shape=4)



重要：

h=0 使得增加噪声后，好友数真实值为0的，在增加了噪声后，变成负的。



Dplyr使用：

Group\_by(pf,age)

> age\_group <- group\_by(facedata,age)

> age\_group

# A tibble: 99,003 x 15

# Groups: age [101]

userid age dob\_day dob\_year dob\_month gender tenure friend\_count friendships\_initiated likes likes\_received

<int> <int> <int> <int> <int> <fctr> <int> <int> <int> <int> <int>

1 2094382 14 19 1999 11 male 266 0 0 0 0

2 1192601 14 2 1999 11 female 6 0 0 0 0

3 2083884 14 16 1999 11 male 13 0 0 0 0

4 1203168 14 25 1999 12 female 93 0 0 0 0

5 1733186 14 4 1999 12 male 82 0 0 0 0

6 1524765 14 1 1999 12 male 15 0 0 0 0

7 1136133 13 14 2000 1 male 12 0 0 0 0

8 1680361 13 4 2000 1 female 0 0 0 0 0

9 1365174 13 1 2000 1 male 81 0 0 0 0

10 1712567 13 2 2000 2 male 171 0 0 0 0

# ... with 98,993 more rows, and 4 more variables: mobile\_likes <int>, mobile\_likes\_received <int>,

# www\_likes <int>, www\_likes\_received <int>

> facedata.fc\_by\_age <- summarise(age\_group,friend\_count\_mean = mean(friend\_count),friend\_count\_media = median(friend\_count),n=n())

> head(facedata.fc\_by\_age)

# A tibble: 6 x 4

age friend\_count\_mean friend\_count\_media n

<int> <dbl> <dbl> <int>

1 13 164.7500 74.0 484

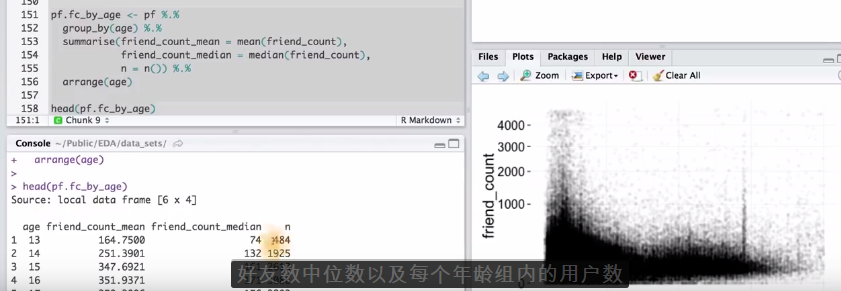
2 14 251.3901 132.0 1925

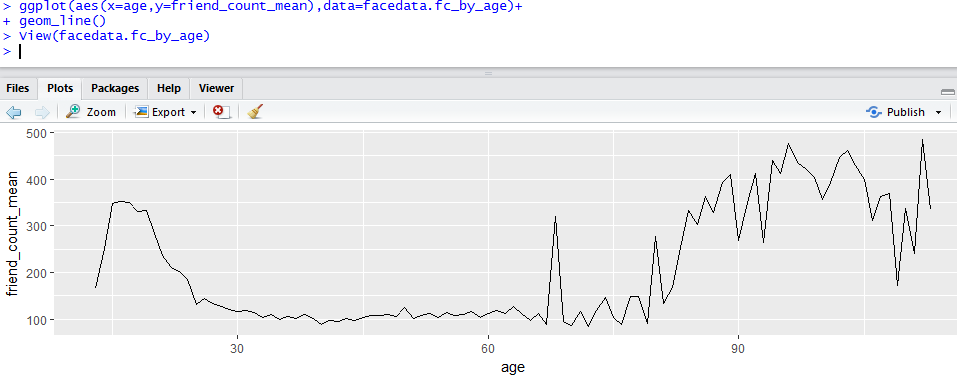
3 15 347.6921 161.0 2618

4 16 351.9371 171.5 3086

5 17 350.3006 156.0 3283

6 18 331.1663 162.0 5196





与原始数据叠加：

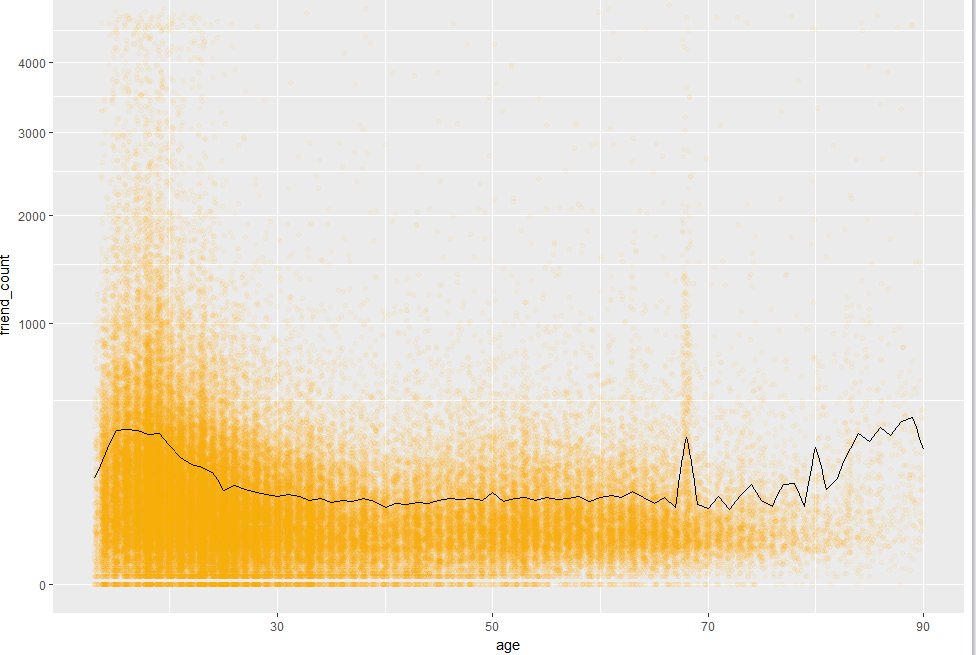
ggplot(aes(x=age,y=friend\_count),data=facedata)+

geom\_jitter(alpha=1/20,position=position\_jitter(h=0),color='orange')+

xlim(13,90)+

coord\_trans(y = "sqrt")+

geom\_line(stat='summary',fun.y=mean)



> ggplot(aes(x=age,y=friend\_count),data=facedata)+

+ coord\_cartesian(xlim = c(13, 90),ylim=c(0,1000))+

+ geom\_jitter(alpha=1/20,position=position\_jitter(h=0),color='orange')+

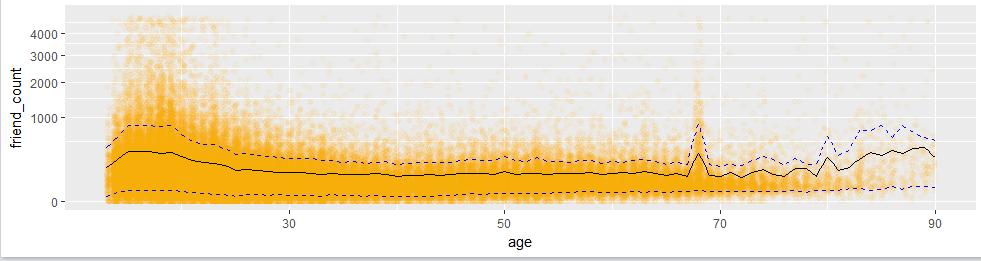
+ xlim(13,90)+

+ coord\_trans(y = "sqrt")+

+ geom\_line(stat='summary',fun.y=mean)+

+ geom\_line(stat='summary',fun.y=quantile, fun.args = list(probs = .1),linetype=2,color='blue')+

+ geom\_line(stat='summary',fun.y=quantile, fun.args = list(probs = .9),linetype=2,color='blue')



相关系数：

?cor.test函数：

> cor.test(pf$age,pf$friend\_count,method = 'pearson')

Pearson's product-moment correlation

data: pf$age and pf$friend\_count

t = -8.6268, df = 99001, p-value < 2.2e-16

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:

-0.03363072 -0.02118189

sample estimates:

cor

-0.02740737

这个相关系数很小，表明这两者之间相关性不大，一般在0.5到0.9之间，相关性比较大

> with(subset(pf,age<=70),cor.test(age,friend\_count)) 小于等于70岁的子集，相关性变大

Pearson's product-moment correlation

data: age and friend\_count

t = -52.592, df = 91029, p-value < 2.2e-16

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:

-0.1780220 -0.1654129

sample estimates:

cor

-0.1717245

强相关性：

> ggplot(aes(x=www\_likes\_received,y=likes\_received),data=pf)+

+ geom\_point()+

+ xlim(0,quantile(pf$www\_likes\_received,0.95))+

+ ylim(0,quantile(pf$likes\_received,0.95))+

+ geom\_smooth(method='lm',color='red')

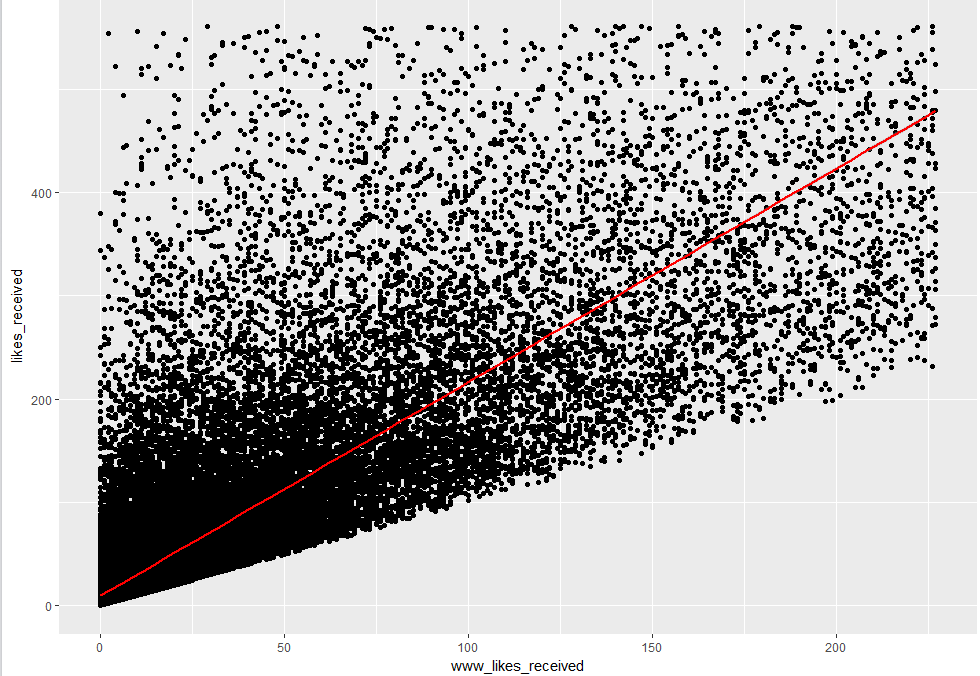
Alr3：

install.packages('alr3')

library(‘alr3’)

> ggplot(aes(x=Month,y=Temp),data=Mitchell)+

+ geom\_point()



在 ggplot 2.0 中，你需要使用 scale\_x\_continuous() 层，而不是 scale\_x\_discrete()。因为月是一个数字变量。

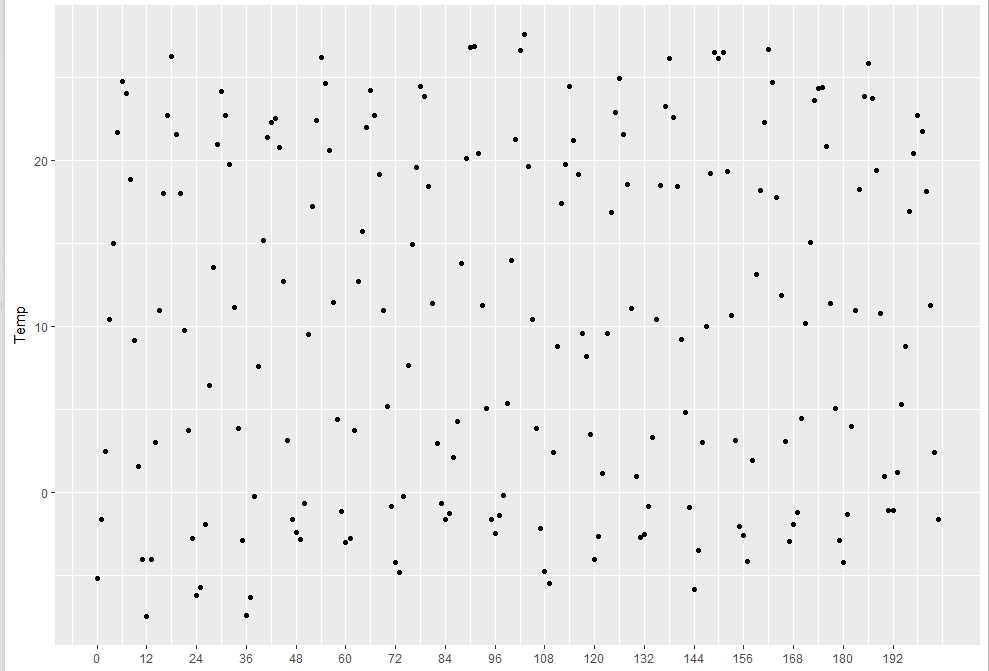
range(Mitchell$Month)

[1] 0 203

> ggplot(aes(x=Month,y=Temp),data=Mitchell)+

geom\_point()+

scale\_x\_continuous(breaks=seq(0,203,12))



ggplot(aes(x=Month,y=Temp),data=Mitchell)+

geom\_point()+

scale\_x\_discrete(breaks=seq(0,203,11))

要理解该数据，你还可以将每一年的数据相互叠加，构建出一个清晰的常规正弦曲线图。你可以在代码中使用 R 的[模数](http://en.wikipedia.org/wiki/Modular_arithmetic" \t "_blank)运算符 %% 来这样做。尝试运行以下代码！

ggplot(aes(x=(Month%%12),y=Temp),data=Mitchell)+   
  geom\_point()

还可以使用其他的关联度量检测这一点。energy 包中的 dcor.ttest() 函数对两个变量的独立性执行非参数检验。尽管 Mitchell 土壤数据集太过粗糙，以至于无法识别“月份”和“温度”之间的显著依赖性。我们可以通过其他例子了解 dcor.ttest 和 cor.test 的区别，例如：

x <- seq(0, 4\*pi, pi/20)  
y <- cos(x)  
qplot(x = x, y = y)  
dcor.ttest(x, y)

消除年龄的噪声，给实际年龄加上一个月龄：

pf$age\_with\_months <-pf$age + (1 - pf$dob\_month / 12)

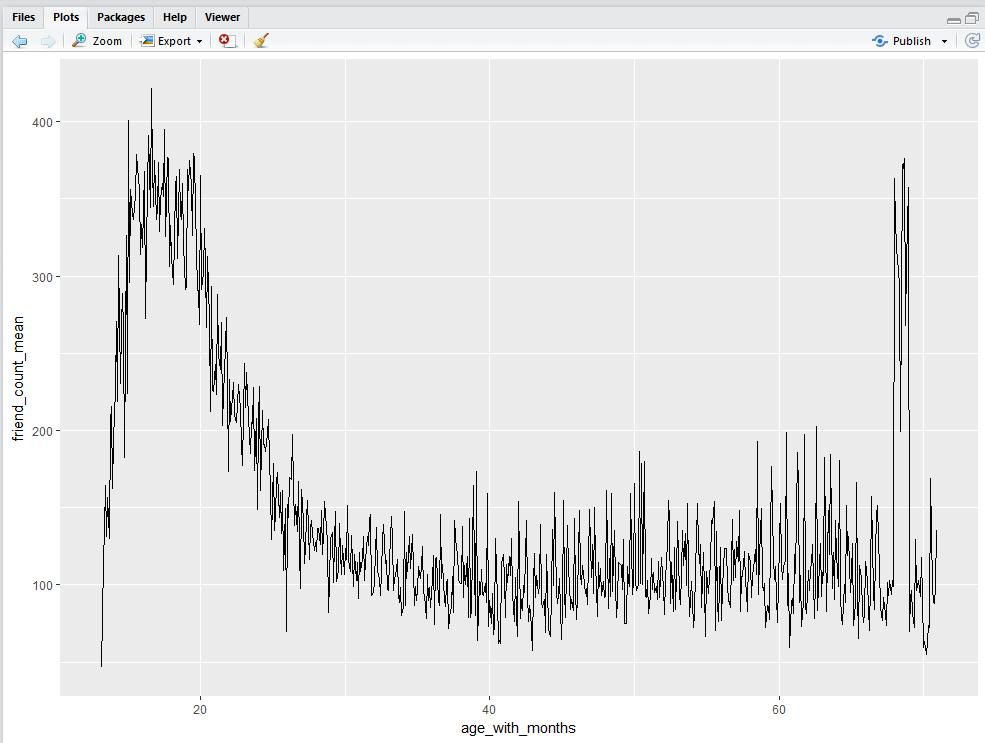
age\_with\_months\_groups <- group\_by(pf,age\_with\_months)

pf.fc\_by\_age\_months <- summarise(age\_with\_months\_groups,friend\_count\_mean = mean(friend\_count),friend\_count\_media = median(friend\_count),n=n())

pf.fc\_by\_age\_months <- arrange(pf.fc\_by\_age\_months,age\_with\_months)

> ggplot(aes(x=age\_with\_months,y=friend\_count\_mean),data=subset(pf.fc\_by\_age\_months,age\_with\_months<71)+

+ geom\_line()



三种参数的比较及绘图：

> library(dplyr)

> library(ggplot2)

> age\_group <- group\_by(pf,age)

> pf.fc\_by\_age <- summarise(age\_group,friend\_count\_mean = mean(friend\_count),friend\_count\_media = median(friend\_count),n=n())

> p1 <- ggplot(aes(x=age,y=friend\_count\_mean),data = subset(pf.fc\_by\_age,age<71))+

+ geom\_line()+

+ geom\_smooth()

> age\_group <- group\_by(pf,age)

> pf.fc\_by\_age <- summarise(age\_group,friend\_count\_mean = mean(friend\_count),friend\_count\_media = median(friend\_count),n=n())

> p1 <- ggplot(aes(x=age,y=friend\_count\_mean),data = subset(pf.fc\_by\_age,age<71))+

+ geom\_line()+

+ geom\_smooth()

> p2 <- ggplot(aes(x=age\_with\_months,y=friend\_count\_mean),data=subset(pf.fc\_by\_age\_months,age\_with\_months<71))+

+ geom\_line()+

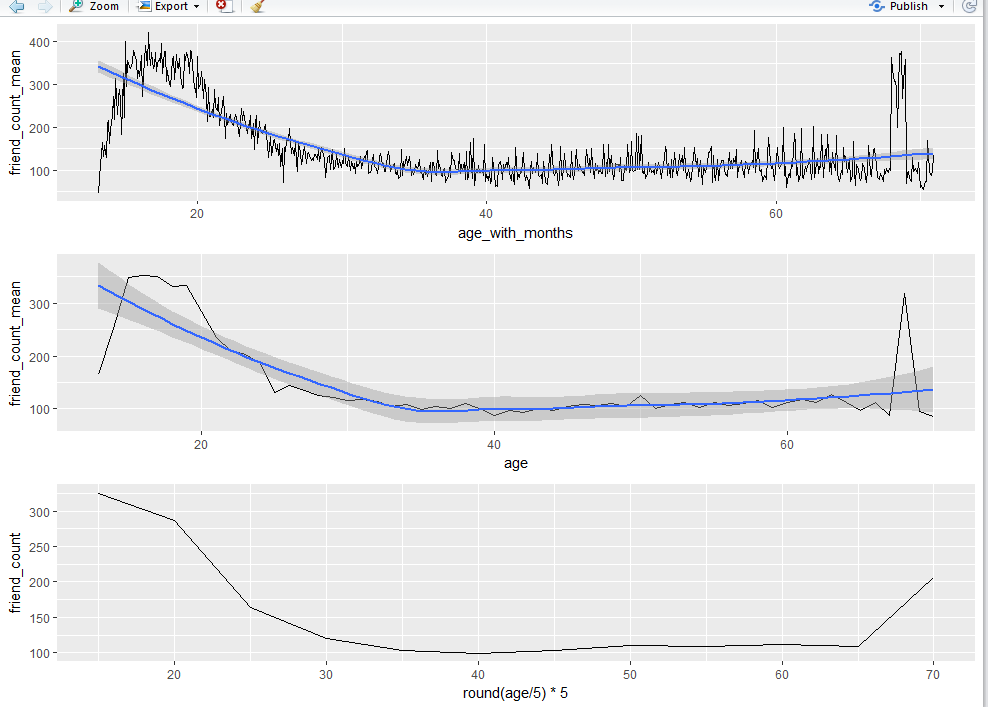
+ geom\_smooth()

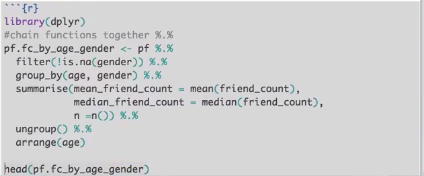
> p3 <- ggplot(aes(x=round(age/5)\*5,y=friend\_count),data=subset(pf,age<71))+

+ geom\_line(stat='summary',fun.y=mean)

> library(gridExtra)

> grid.arrange(p2,p1,p3,ncol=1





**重要事项！请注意，在较新版本的 dplyr (0.3.x+) 中，语法 %.% 已被弃用且替换为 %>%。课程视频中使用-的是 %.%，但这会产生警告消息。如果你在回答中使用链接运算符，你应该使用 %>%**

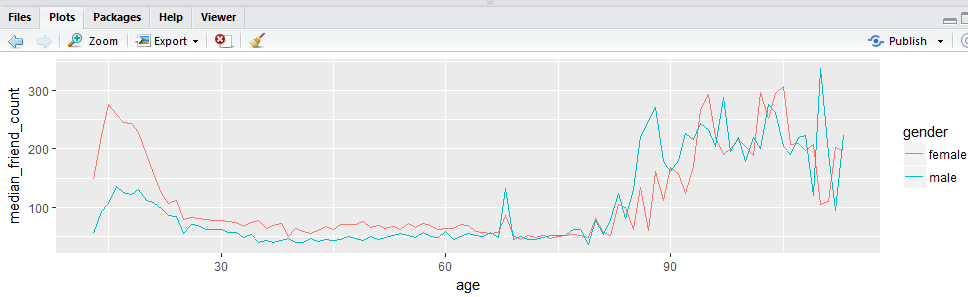
**Group\_by,多个变量的使用：**

age\_gen\_group <- group\_by(facedata,age,gender)

facedata.frm\_age\_gen <- summarise(age\_gen\_group,mean\_friend\_count=mean(friend\_count),median\_friend\_count=median(friend\_count),n=n())

ggplot(aes(x=age,y=median\_friend\_count),data=subset(facedata.frm\_age\_gen,!is.na(gender)))+

+ geom\_line(aes(color=gender))



如果你在进行此练习之前使用了 plyr 包中的 count() 函数，则需要运行以下命令卸载 plyr 包。   
detach("package:plyr", unload=TRUE)

0.4.0 版本的 dplyr 在汇总层上使用中间值函数时有一个错误，具体取决于被汇总的数据性质。你可能需要将数据转换为数值型（浮点型），以获得期望的结果，比如：median(as.numeric(var))。

> library(plyr)

------------------------------------------------------------------------------------------------------------------

You have loaded plyr after dplyr - this is likely to cause problems.

If you need functions from both plyr and dplyr, please load plyr first, then dplyr:

library(plyr); library(dplyr)

------------------------------------------------------------------------------------------------------------------

Attaching package: 憄lyr?

The following objects are masked from 憄ackage:dplyr?

arrange, count, desc, failwith, id, mutate, rename, summarise, summarize

Warning message:

package 憄lyr?was built under R version 3.4.1

> count(diamonds$volume == 0)

x freq

1 FALSE 53920

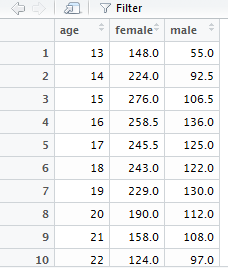
2 TRUE 20

数据重新组合，长格式转成宽格式

install.packages('reshape2')

> library(reshape2)

facedata.frm\_age\_gen\_wide <- dcast(subset(facedata.frm\_age\_gen,!is.na(gender)),age~gender,value.var = 'median\_friend\_count')



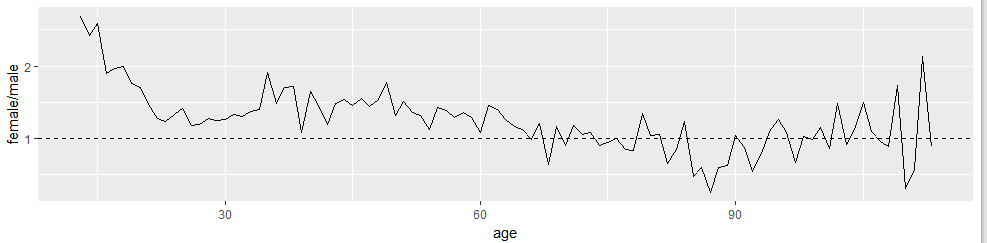
# The linetype parameter can take the values 0-6:

# 0 = blank, 1 = solid, 2 = dashed # 3 = dotted, 4 = dotdash, 5 = longdash # 6 = twodash

> ggplot(aes(x=age,y=female/male),data=facedata.frm\_age\_gen\_wide)+

+ geom\_line()+

+ geom\_hline(yintercept = 1,linetype='dashed')



Floor取值：

> facedata$year\_joined <- floor(2014 - facedata$tenure/365)

> table(facedata$year\_joined)

2005 2006 2007 2008 2009 2010 2011 2012 2013 2014

9 15 581 1507 4557 5448 9860 33366 43588 70

> summary(facedata$year\_joined)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

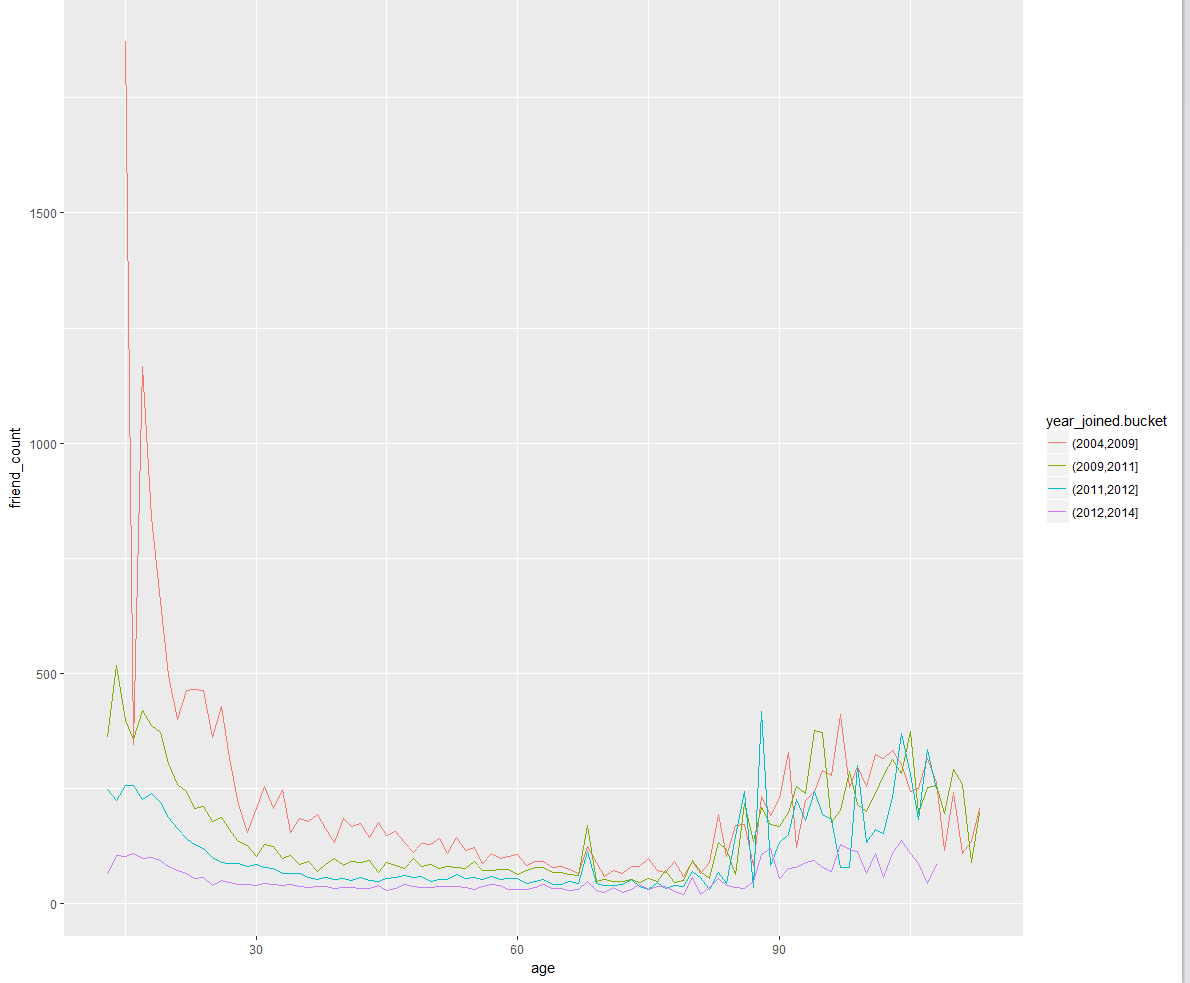
2005 2012 2012 2012 2013 2014 2

切割一个变量：

facedata$year\_joined.bucket <- cut(facedata$year\_joined,c(2004,2009,2011,2012,2014))

ggplot(aes(x = age, y = friend\_count),data = subset(facedata, !is.na(year\_joined.bucket)))+

geom\_line(aes(color = year\_joined.bucket), stat = 'summary', fun.y = median)



增加总均值：

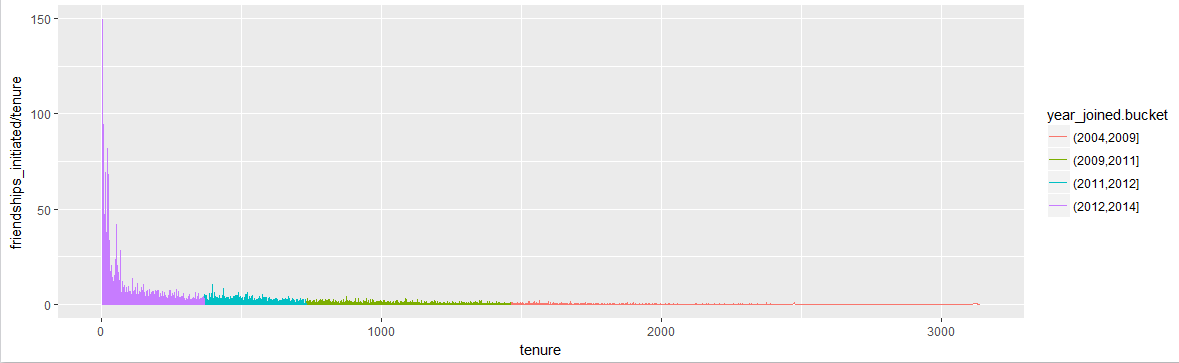
> ggplot(aes(x = age, y = friend\_count),data = subset(facedata, !is.na(year\_joined.bucket)))+

+ geom\_line(aes(color = year\_joined.bucket), stat = 'summary', fun.y = mean)+

+ geom\_line(stat='summary',fun.y=mean)

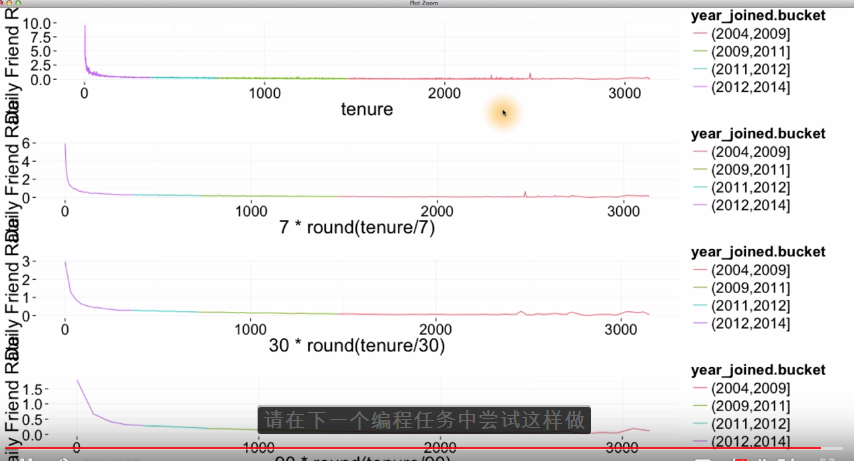
使用的时间越长，建立新的友谊越少

ggplot(aes(x=tenure, y=friendships\_initiated/tenure),data= subset(facedata,tenure>=1))+geom\_line(aes(color=year\_joined.bucket))



减少噪声：

Tenure=30\*round(tenure/30)噪声最小



ggplot(aes(x = 7 \* round(tenure / 7), y = friendships\_initiated / tenure),

data = subset(pf, tenure > 0)) +

geom\_line(aes(color = year\_joined.bucket),

stat = "summary",

fun.y = mean)

可以改为用geom\_smooth来变的光滑，去除噪声

> ggplot(aes(x=tenure, y=friendships\_initiated/tenure),data= subset(facedata,tenure>=1))+

geom\_smooth(aes(color=year\_joined.bucket))

酸奶数据集：

> table(yo$price)

20 24.96 33.04 33.2 33.28 33.36 33.52 39.04 44 45.04 48.96 49.52 49.6 50 55.04 58.96 62 63.04 65.04 68.96

2 11 54 1 1 22 1 234 21 11 81 1 1 205 6 303 15 2 799 609

> summary(yo)

obs id time strawberry blueberry pina.colada plain mixed.berry

Min. : 1.0 Min. :2100081 Min. : 9662 Min. : 0.0000 Min. : 0.0000 Min. : 0.0000 Min. :0.0000 Min. :0.0000

1st Qu.: 696.5 1st Qu.:2114348 1st Qu.: 9843 1st Qu.: 0.0000 1st Qu.: 0.0000 1st Qu.: 0.0000 1st Qu.:0.0000 1st Qu.:0.0000

Median :1369.5 Median :2126532 Median :10045 Median : 0.0000 Median : 0.0000 Median : 0.0000 Median :0.0000 Median :0.0000

Mean :1367.8 Mean :2128592 Mean :10050 Mean : 0.6492 Mean : 0.3571 Mean : 0.3584 Mean :0.2176 Mean :0.3887

3rd Qu.:2044.2 3rd Qu.:2141549 3rd Qu.:10255 3rd Qu.: 1.0000 3rd Qu.: 0.0000 3rd Qu.: 0.0000 3rd Qu.:0.0000 3rd Qu.:0.0000

Max. :2743.0 Max. :2170639 Max. :10459 Max. :11.0000 Max. :12.0000 Max. :10.0000 Max. :6.0000 Max. :8.0000

price

Min. :20.00

1st Qu.:50.00

Median :65.04

Mean :59.25

3rd Qu.:68.96

Max. :68.96

> length(unique(yo$price))

[1] 20

数据transform

> yo <- transform(yo, all.purchase=strawberry+blueberry+pina.colada+plain+mixed.berry)

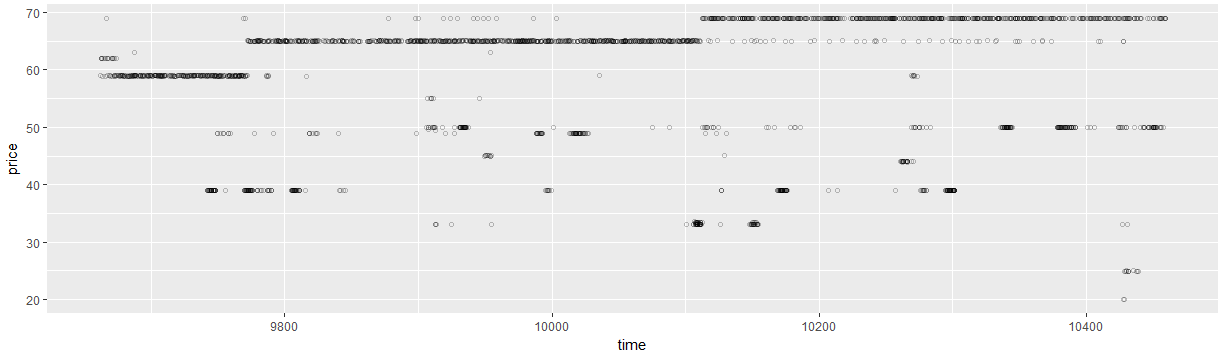
> summary(yo$all.purchase)

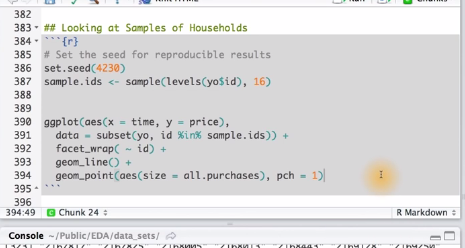
Min. 1st Qu. Median Mean 3rd Qu. Max.

1.000 1.000 2.000 1.971 2.000 21.000

ggplot(aes(x=time,y=price),data=yo,color='orange')+

geom\_jitter(alpha=1/4,shape=21)





> sample.ids <-sample(yo$id,16)

> sample.ids

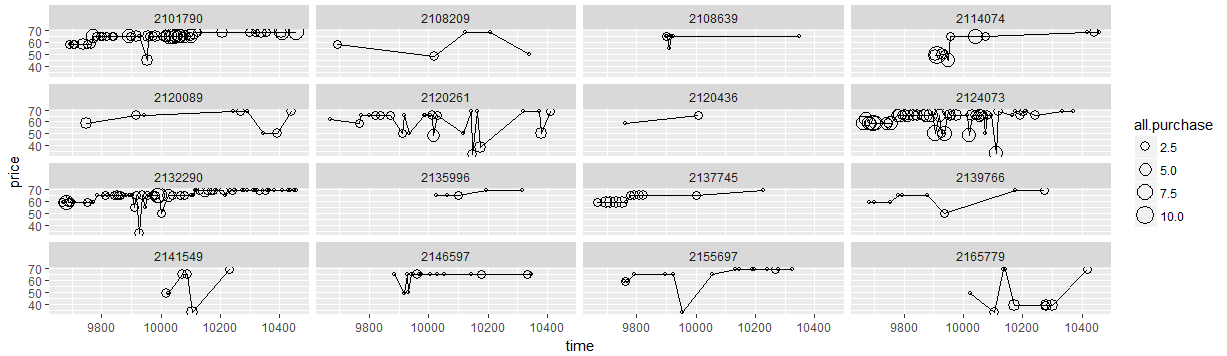
[1] 2139766 2120089 2132290 2124073 2135996 2120261 2165779 2146597 2155697 2108639 2137745 2141549 2120436 2101790 2108209 2114074

> ggplot(aes(x=time,y=price),data=subset(yo,id %in% sample.ids))+

+ facet\_wrap(~id)+

+ geom\_line()+

+ geom\_point(aes(size=all.purchase),pch=1)



散点图矩阵：

install.packages('GGally')

> set.seed(1895)

> pf\_subset <- facedata[,c(2:15)]

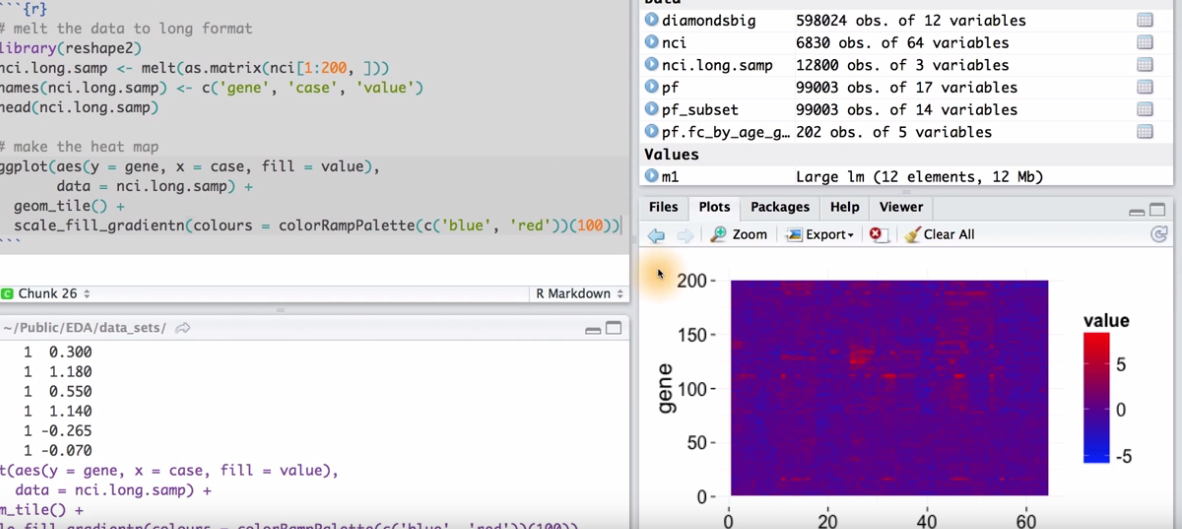
> names(pf\_subset)

[1] "age" "dob\_day" "dob\_year" "dob\_month" "gender"

[6] "tenure" "friend\_count" "friendships\_initiated" "likes" "likes\_received"

[11] "mobile\_likes" "mobile\_likes\_received" "www\_likes" "www\_likes\_received"

> ggpairs(pf\_subset[sample.int(nrow(pf\_subset),1000),])



omit the top 1% of diamond volumes from the plot.

在 xlim 和 ylim 中使用 quantile() 函数，以忽略每个变量前 1% 的值。

ggplot(aes(x=x\*y\*z,y=price),data=diamonds)+

geom\_point(aes(color=clarity))+

scale\_y\_log10()+

xlab('Volume')+

xlim(0,quantile(diamonds$x\*diamonds$y\*diamonds$z,0.99))

探索diamonds数据集：

1 价格与克拉的关系，可以看出是个非线性的关系

ggplot(aes(x=carat,y=price),data=diamonds)+

stat\_smooth(method="lm")+

scale\_x\_continuous(lim=c(0,quantile(diamonds$carat,0.99)))+

scale\_y\_continuous(lim=c(0,quantile(diamonds$price,0.99)))+

geom\_point(fill=I("#F79420"),color=I("black"),shape=21)

2 install.packages("GGally")

install.packages("scales")

install.packages("MASS")

install.packages("car")

install.packages("lattice")

install.packages("memisc")

install.packages("reshap2")

install.packages("dplyr")

> library(GGally)

> library(scales)

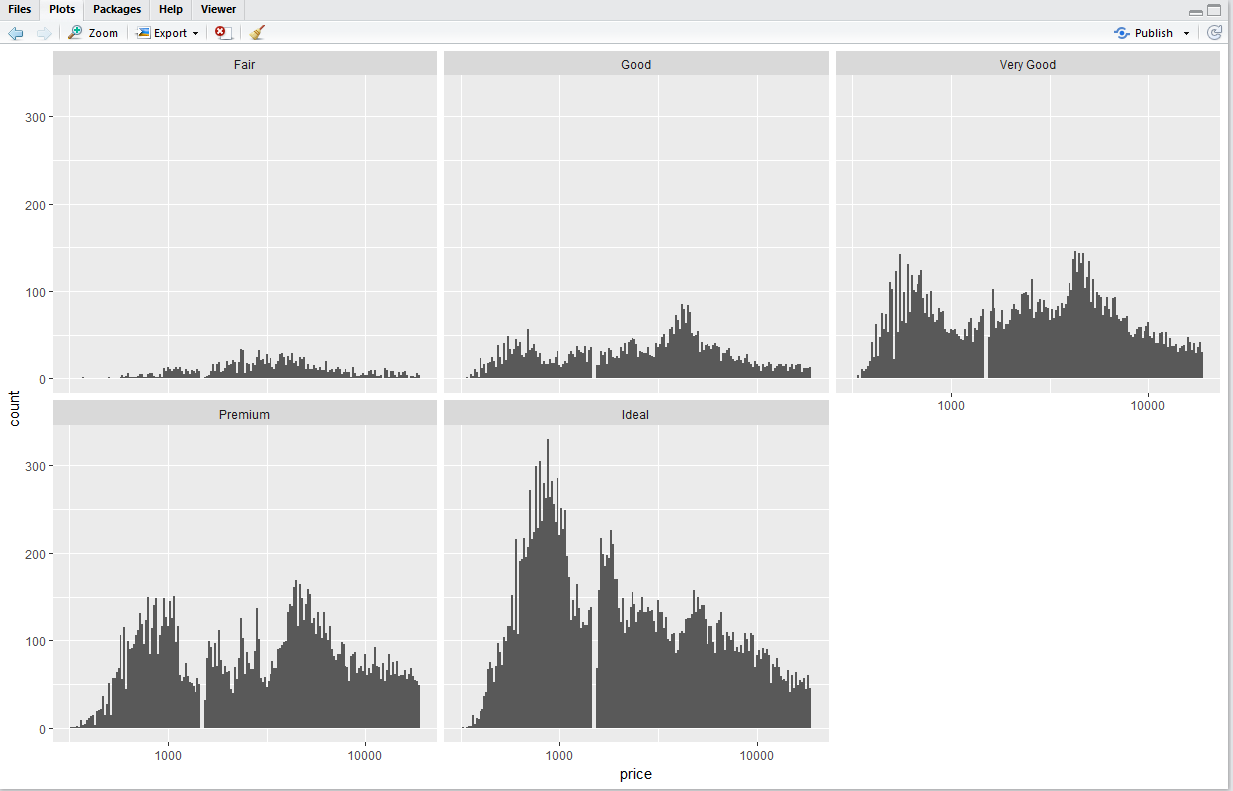
> library(memisc)

3 > ggplot(aes(x=price),data=diamonds)+

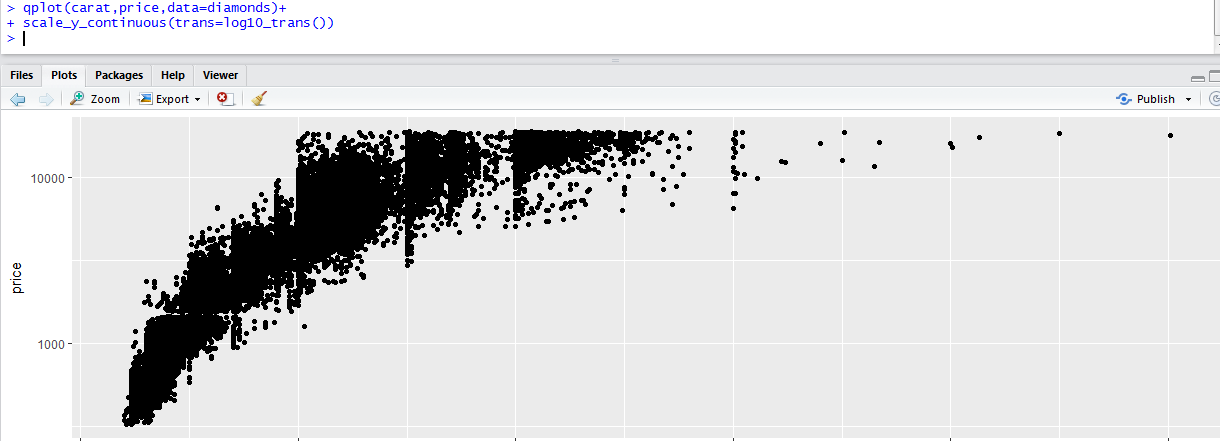
+ facet\_wrap(~ cut)+

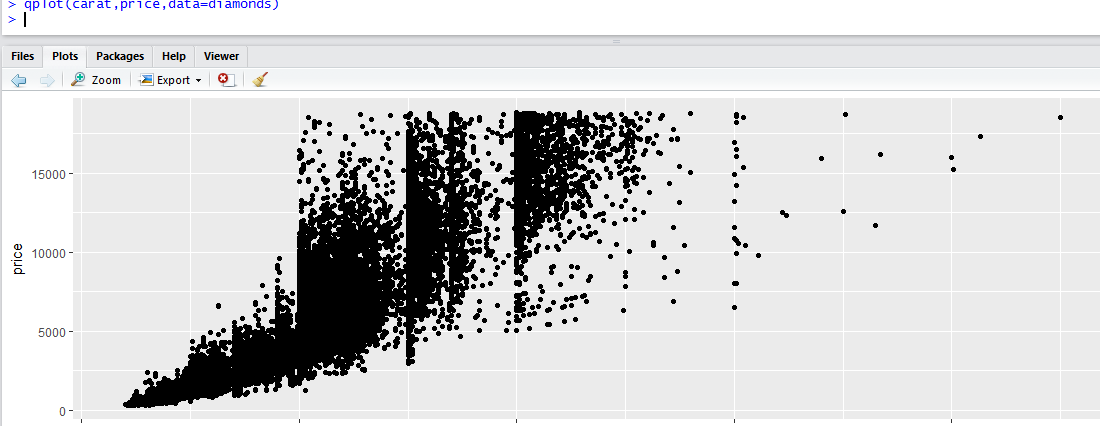
+ geom\_histogram(binwidth=0.01)+

+ scale\_x\_log10()

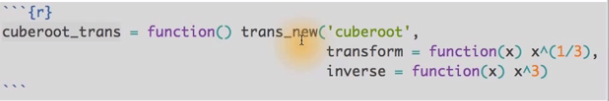


在对数标度上，价格在克拉大小和价格的高端离散度小。





立方根转换函数：



ggplot(aes(carat, price), data = diamonds) +

geom\_point() +

**geom\_point(alpha=1/20,position=’jitter’,size=0.75)**

scale\_x\_continuous(trans = cuberoot\_trans(), limits = c(0.2, 3),

breaks = c(0.2, 0.5, 1, 2, 3)) +

scale\_y\_continuous(trans = log10\_trans(), limits = c(350, 15000),

breaks = c(350, 1000, 5000, 10000, 15000)) +

ggtitle('Price (log10) by Cube-Root of Carat')

过度绘制的处理：

> head(sort(table(diamonds$carat),decreasing = T))

0.3 0.31 1.01 0.7 0.32 1

2604 2249 2242 1981 1840 1558

> head(sort(table(diamonds$price),decreasing = T))

605 802 625 828 776 698

132 127 126 125 124 121

重复的值很多，会导致过度叠加产生过度的绘制，可以用透明度或者抖动来处理

ggplot(aes(x=age,y=friend\_count),data=facedata)+

**geom\_point(alpha=1/20,position=’jitter’,size=0.75)+**

xlim(13,90)+

coord\_trans(y = "sqrt")

安装并加载 RColorBrewer 包。   
install.packages('RColorBrewer', dependencies = TRUE)   
library(RColorBrewer)

价格与克拉之间的关系进行调整，按照净度显示不同的颜色

ggplot(aes(x = carat, y = price,colour=clarity), data = diamonds) +

geom\_point(alpha = 0.5, size = 1, position = 'jitter') +

scale\_color\_brewer(type = 'div',

guide = guide\_legend(title = 'Clarity', reverse = T,

override.aes = list(alpha = 1, size = 2)))**+**

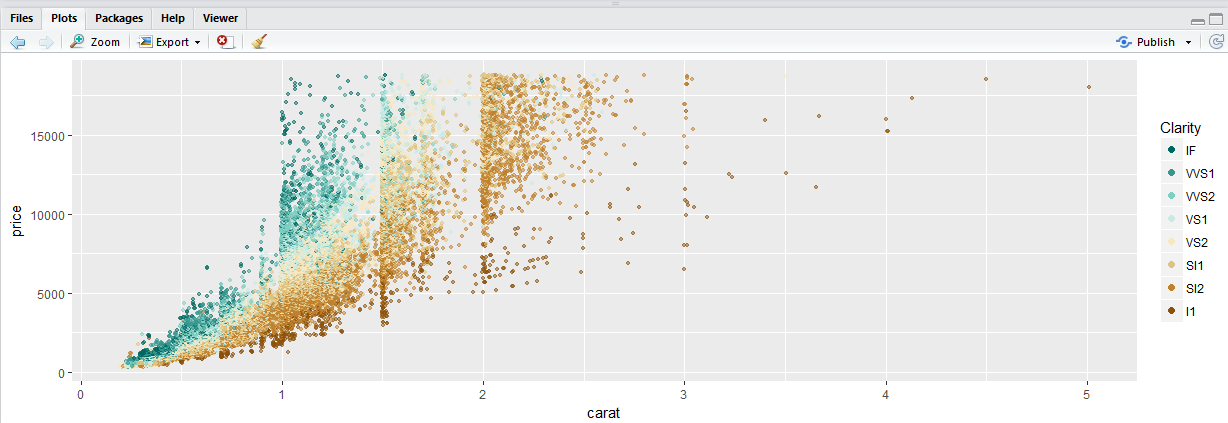
scale\_x\_continuous(trans = cuberoot\_trans(), limits = c(0.2, 3),

breaks = c(0.2, 0.5, 1, 2, 3)) +

scale\_y\_continuous(trans = log10\_trans(), limits = c(350, 15000),

breaks = c(350, 1000, 5000, 10000, 15000)) +

ggtitle('Price (log10) by Cube-Root of Carat')



这样的分类不确定合不合理？

clari\_group <- group\_by(diamonds,carat,clarity)

ggplot(aes(x = carat, y = price), data = diamonds) +

geom\_point(aes(color=clarity),alpha = 0.5, size = 1, position = 'jitter')

R的线性模型：

library(lattice)

library(MASS)

library(memisc)



1 获取更好的数据集：

> library('RCurl')

> library('bitops')

> diamondurl= getBinaryURL("https://raw.github.com/solomorm/diamonds-data/master/BigDiamonds,Rda")

load(rawConnection(diamondurl))

或者：

git clone https://github.com/SolomonMg/diamonds-data

> setwd('C:/edwin/Document/Udacity R study note/diamonds-data')

> load("BigDiamonds.Rda")

2 设置模型：

diamondsbig$log\_price = log(diamondsbig$price)

m1 <- lm(log\_price ~ I(carat^(1/3)),data = diamondsbig[diamondsbig$price<10000 & diamondsbig$cert == "GIA",])

m2 <- update(m1, ~. + carat)

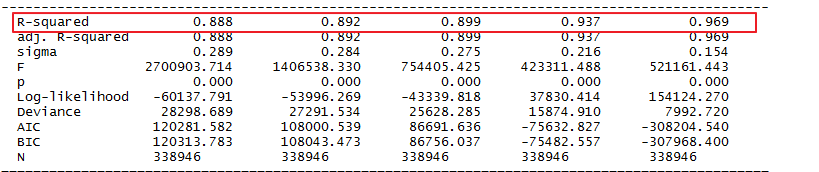
m3 <- update(m2, ~. + cut)

m4 <- update(m3, ~. + color)

m5 <- update(m4, ~. + clarity)

3 计算模型的好坏

mtable(m1,m2,m3,m4,m5)



4 预测：

> thisdiamond = data.frame(carat=1.00, cut='V.Good', color= 'I', clarity ='VS1')

> modelEstimate = predict(m5,newdata=thisdiamond,interval = "prediction",level=.95)

> exp(modelEstimate)

fit lwr upr

1 5040.436 3730.34 6810.638

dat = data.frame(m4$model, m4$residuals)   
  
with(dat, sd(m4.residuals))   
  
with(subset(dat, carat > .9 & carat < 1.1), sd(m4.residuals))   
  
dat$resid <- as.numeric(dat$m4.residuals)  
ggplot(aes(y = resid, x = round(carat, 2)), data = dat) +   
  geom\_line(stat = "summary", fun.y = sd)