实验10、Logistic回归和多项式回归分析

年级： 15级　　专业：生信　　学号：1513401013　　姓名：郑磊

编号　　一　　二　　三　　四　　总分　　评阅人

得分

软硬件平台：

1. 硬件平台：（硬件配置）i5，2.9HZ处理器，16G内存，64位操作系统

2. 系统平台：（操作系统及其版本号）Windows10 企业版

3. 软件平台：（软件系统及其版本号，若是在线分析平台，还需要提供URL地址）R3.4.1 ，Rstudio

一、目的要求：

1、加深对Logistic和多项式回归分析的理解；

2、从统计学角度来探索基因表达水平与不同病理类型样本的关联；

3、熟悉并掌握Logistic和多项式回归分析所涉及的R语言函数和脚本。

二、实验内容：

1、多次项曲线的模拟：

1.1、一元多次项模拟：

至少模拟2~5次项曲线，看看有没有什么规律？

```
dir="D:/RFile/实验十"
setwd(dir)
library(ggplot2)
library(gridExtra) #针对ggplot2的多图排版
#几次项？=》更改这个参数即可
k=2
```

```
#一共生成7组数据，每组100个数值
group=7; n=100
#自变量最大区间
x_min=-3; x_max=3
#系数区间
b_min=-8; b_max=8
#则因变量理论区间
y_min=y_max=0
for(j in 0:k){y_min = y_min + b_min*(x_max^j); y_max = y_max +
b_max*(x_max^j)}
#自变量波动区间
c_min=-1;c_max=1
#创建存储数据的data.frame，共3列，第一列group序号，第二列自变量x，第
三列因变量y
data<-data.frame(matrix(NA,group*n,3))
colnames(data)<-c("group","x","y")
#formula<-data.frame(matrix(NA,group,1)) #存放方程式
#根据设定参数进行数据模拟
for(i in 1:group)
{
   set.seed(i+runif(1,0,100))
   x<-runif(n,min=x_min,max=x_max)
```

```
    b<-round(runif(k+1,min=b_min,max=b_max))
    c<-runif(n,min=c_min,max=c_max)
    x<-x+c
    y<-data.frame(matrix(0,n,1))
    for(m in 0:k){ y <- y + b[m+1] *(x^m) }
    from = (i-1)*n+1; to = n*i
    data[from:to,1]=rep(i,n)
    data[from:to,2]=x
    data[from:to,3]=y
}
#绘制一元多次项模拟散点图+拟合曲线
g1=ggplot(data, aes(x=x, y=y, colour=group)) + geom_point()#以颜色梯度区分
data$group <- as.factor(data$group) #group列定义为因子
g2=ggplot(data, aes(x=x, y=y, colour=group)) + geom_point() #以不同颜色区分
g3= ggplot(data, aes(x=x, y=y, colour=group)) + geom_point() +
stat_smooth(method='lm', formula=y~poly(x,k)) #增加拟合曲线
g4= ggplot(data, aes(x=x, y=y, colour=group)) + geom_point() +
stat_smooth(method='lm', formula=y~poly(x,k)) +
theme(axis.title=element_text(face="bold",size=12), axis.text =
element_text(face="bold",color="blue", size=10)) #增加图片修饰

#注意4张图的区别，输出到一张图片上
```

```
png(file = "plot_y_x-k_ggplot_2.png")
grid.arrange(g1, g2, g3, g4, ncol=2)
dev.off()
```

2、基于基因表达水平（自变量xi）的样本类型的Logistic回归分析：

该环节需要大家提前准备好一个基因表达谱数据，如果没有，则有授课教师提供（gds4794）。尝试把不同样本作为因变量（y），几万个基因表达水平作为自变量（x1，2，...），进行探讨。

2.1、数据读取：

```
#加载本地的数据
gds4794 <- getGEO(filename='GDS4794.soft.gz')
#查看数据类型
mode(gds4794)
#查看注释信息
Meta(gds4794)
#查看列注释信息=》用来确定哪些列是肿瘤，哪些列是正常对照
Columns(gds4794)
#1：23是肺癌，24：65是正常组织
data<-Table(gds4794)
#查看数据表的列名
colnames(data)
#查看数据表行列数
ncol(data)
```

#[1] 67

nrow(data)

#[1] 54675

#前面两列是标题列，分别为探针id和基因名称

#3：25列是 lung cancer，26：67列是 normal

#第一列探针IDs定义为data的行标题

rownames(data)<-data[,1]

2.2、随机取样分析：

至少有3个基因的表达水平回归分析结果达到0.05的显著水平。

#随机抽取至少10行数据

n=10

#使用以下代码进行循环测试：齐方差、F检验，p>0.1；齐方差、F检验，双因素p<0.1，无交互作用；齐方差、F检验，双因素p无要求，p<0.1

#按行随机抽样【实验结果中需要记录】

row.names<-rownames(data)

sam.row.name <- sample(row.names,n,replace=F)

sam.row.name #查看抽中的数据行探针id

subdata<-data[sam.row.name,3:67] #提取抽样数据

#加上样本病理类型数据共n+1列

#初始化数据表

data2<-data.frame(matrix(NA,65, n+1))

#增加样本病理类型分类数据，肺癌=1，其他正常=0

```
data2[,1]<-c(rep(1,23),rep(0,42))
data2[,2:(n+1)]<-t(log(subdata)) #后面n列存放筛选出来的基因数据，注意矩阵行列转换
colnames(data2)<-c("y",paste("x",1:n,sep="")) #设定列标题y,x1,x2,...,x10
#以样本类型为因变量y，其他所有基因表达式水平为自变量x1,x2,...x10，进行总体回归分析
glm0<-glm(y~.,family=binomial(link='logit'),data=data2)
summary(glm0)
#向后逐步回归法
glm.step<-step(glm0,direction="backward")
summary(glm.step)
#绘制回归评估的4张图
png(file = "glm4.png")
par(mfrow=c(2,2))
plot(glm.step)
dev.off()
#car包里的influencePlot()函数能一次性同时检查离群点、高杠杆点、强影响点
library(car)
png("influencePlot.png")
influencePlot(glm.step,id.method = "identity", main="Influence Plot",sub="Circle size is proportional to Cook's distance")
dev.off()
```

#绘制subdata的热图

colnames(subdata)<-Columns(gds4794)$disease.state

png(file = "heatmap1.png")

heatmap(as.matrix(log(subdata)), Rowv = NA, Colv = NA)

dev.off()

2.3、差异表达基因分析：

#变量初始化，用来存放计算结果中的p.value和fold change值

p=NULL

fold.change=NULL

#R用Sys.time()可以查看当前系统时间

#程序开始时记录：

timestart<-Sys.time()

#基因表达谱遍历

for(i in 1:nrow(data))

{

  a <- unlist(data[i,3:25])

  b <- unlist(data[i,26:67])

  fold.change<-c(fold.change,mean(a,na.rm=TRUE)/mean(b,na.rm=TRUE))

  x<-t.test(a,b)

  p<-c(p,x$p.value)

}

#程序临结束时记录：

```
timeend<-Sys.time()

#程序运行时间：

timeend-timestart

#Time difference of 51.29762 secs

#data第一列探针名IDs作为p和fold.change的名称

names(p)<-data[,1]

names(fold.change)<-data[,1]

#设定阈值进行筛选

p_value = 0.01

up = 50 #lung cancer  上调2倍

down = 0.02 #lung cancer  下调2倍

#筛选

p2 <- p[p<p_value] #p值筛选

fc.up <- fold.change[fold.change>up] #上调基因

fc.down <- fold.change[fold.change<down] #下调基因

length(p2); length(fc.up); length(fc.down) #查看筛选结果

#交集计算

probes.up<-intersect(names(p2),names(fc.up)) #符合统计学显著性的上调基因

length(probes.up)

probes.down<-intersect(names(p2),names(fc.down)) #符合统计学显著性的下调
基因

length(probes.down)
```

2.5、混合上调和下调基因进行Logistic回归分析：

probes<-union(probes.up,probes.down) #合并合统计学显著性的上调和下调基因

#上述过程合并进行

#probes <- intersect(names(p2),union(names(fc.up),names(fc.down)))

length(probes)

subdata2<-data[probes,3:67] #从原始基因表达谱数据表中提取筛选出来的基因数据

rownames(subdata2)<-probes #设定探针IDs为行标题

nrow(subdata2)

#如果筛选的基因数量过多，接下来则无法进行下去

#加上样本病理类型数据共17列

data3<-data.frame(matrix(NA,65, 17)) #初始化数据表

data3[,1]<-c(rep(1,23),rep(0,42)) #增加样本病理类型分类数据，肺癌=1，其他正常=0

data3[,2:17]<-t(log(subdata2)) #后面16列存放筛选出来的基因数据，注意矩阵行列转换

colnames(data3)<-c("y",paste("x",1:16,sep="")) #设定列标题

#以样本类型为因变量y，其他所有基因表达式水平为自变量x1,x2,...，进行总体回归分析

glm0<-glm(y~.,family=binomial(link='logit'),data=data3)

summary(glm0)

```
glm.step<-step(glm0,direction="backward")
summary(glm.step)
png(file = "lec11_ICU_glm.png")
par(mfrow=c(2,2))
plot(glm.step)
dev.off()
#car包里的influencePlot()函数能一次性同时检查离群点、高杠杆点、强影响点。
library(car)
png("influencePlot.png")
influencePlot(glm.step,id.method = "identity", main="Influence Plot",sub="Circle size is proportional to Cook's distance")
dev.off()
#绘制subdata2的热图
colnames(subdata2)<-Columns(gds4794)$disease.state
png(file = "heatmap.png")
heatmap(as.matrix(log(subdata2)), Rowv = NA, Colv = NA)
dev.off()
```

3、多项式回归分析：

探索经纬度与温度变化的关系。

3.1、数据读取与可视化：

```
file="US_Temperatures_Data"
```

```
data<-read.table(file,head=T,sep="\t")
colnames(data)
a<-max(data$JanTemp) - min(data$JanTemp) + 1 #设定颜色梯度区间
png(file = "plot_y_x_t_scatter.png")
cPal <- colorRampPalette(c('green','red'))
Cols <- cPal(a)[as.numeric(cut(data$JanTemp,breaks = a))]
plot(data$Long,data$Lat,pch = 20,col = Cols,cex=2)
dev.off()
```

3.2、局部多项式回归拟合探索：

在R语言中进行局部多项式回归拟合是利用loess函数

LOESS的优势是并不需要确定具体的函数形式，而是让数据自己来说话，其缺点在于需要大量的数据和运算能力。LOESS作为一种平滑技术，其目的是为了探寻响应变量和预测变量之间的关系，所以LOESS更被看作一种数据探索法，而不是作为最终的结论。

用loess来建立模型时重要的两个参数是span和degree，span表示数据子集的获取范围，取值越大则数据子集越多，曲线越为平滑。degree表示局部回归中的阶数，1表示线性回归，2表示二次回归（默认），也可以取0，此时曲线退化为简单移动平均线。这里我们设span取0.4和0.8，从下图可见取值0.8的蓝色线条较为平滑。

（1） JanTemp~Lat拟合

```
model1=loess(JanTemp~Lat,data=data,span=0.4)
summary(model1)
png(file = "plot_T_Lat_loess.png")
```

```
plot(data$JanTemp~data$Lat,pch = 20,col = Cols,cex=2)

lines(data$Lat,model1$fit,col='red',lty=2,lwd=2)

dev.off()
```

（2） JanTemp~Long拟合

```
#JanTemp~Long拟合

model2=loess(JanTemp~Long,data=data,span=0.8)

summary(model2)

png(file = "plot_T_Long_loess.png")

plot(data$JanTemp~data$Long,pch = 20,col = Cols,cex=2)

lines(data$Long,model2$fit,col='red',lty=2,lwd=2)

dev.off()
```

3.3、二元线性回归分析：

```
#二元线性回归的探索

lm.line<-lm(JanTemp~Lat+Long,data=data)

summary(lm.line)

png(file = "plot_y_x_t_lm.png")

par(mfrow=c(2,2))

plot(lm.line)

dev.off()
```

3.4、多项式回归分析：

```
#Lat为线性，Long为三次项

model <- lm(JanTemp ~ Lat + poly(Long,3),data=data)

summary(model)
```

#模型参数的置信区间

confint(model, level=0.95)

#拟合VS残差图,如果这是一个拟合效果比较不错的模型，应该看不到任何一种模型的特征

png(file = "plot_T_Lat_Long_model_residuals.png")

par(mfrow=c(2,2))

plot(model)

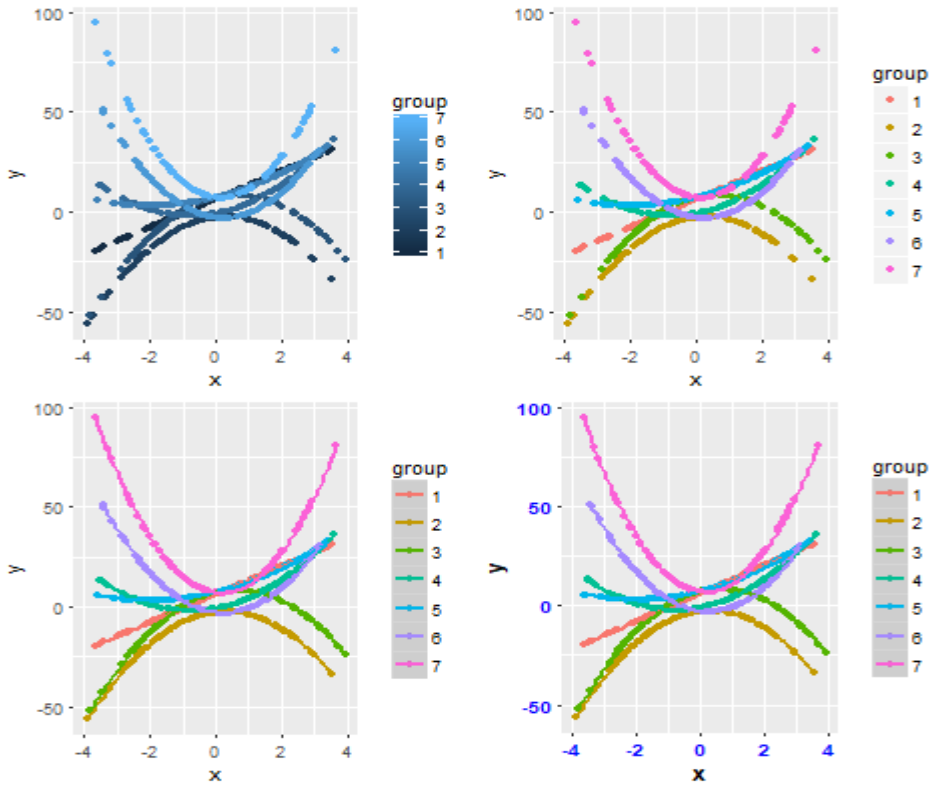
plot(fitted(model),residuals(model))

dev.off()

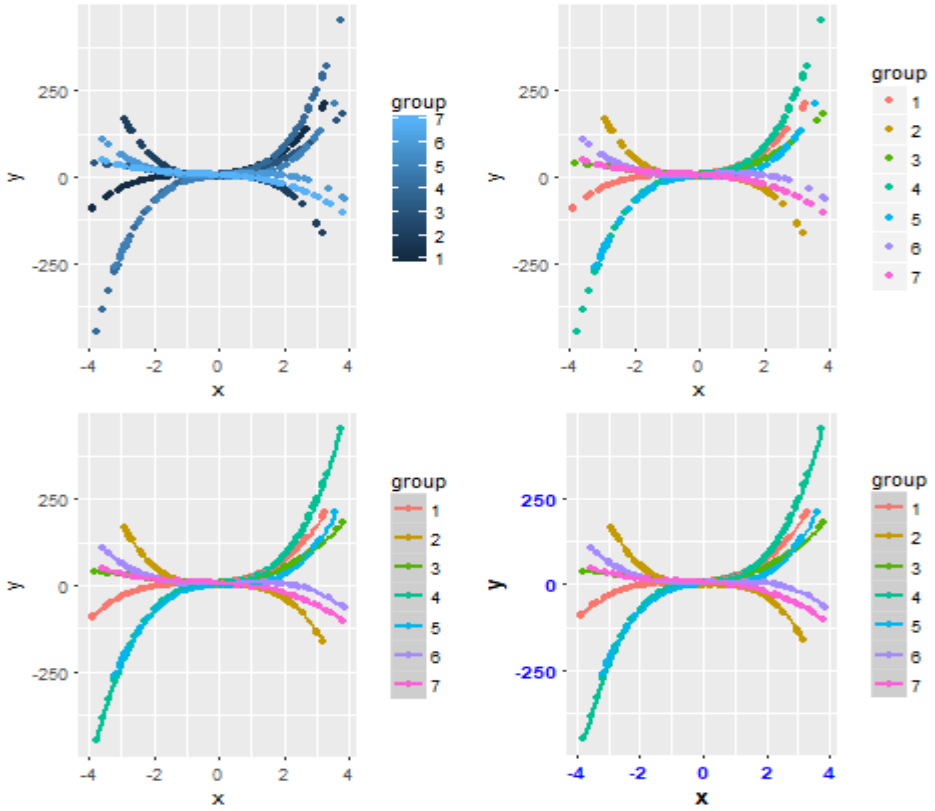
三、实验结果：

1.1

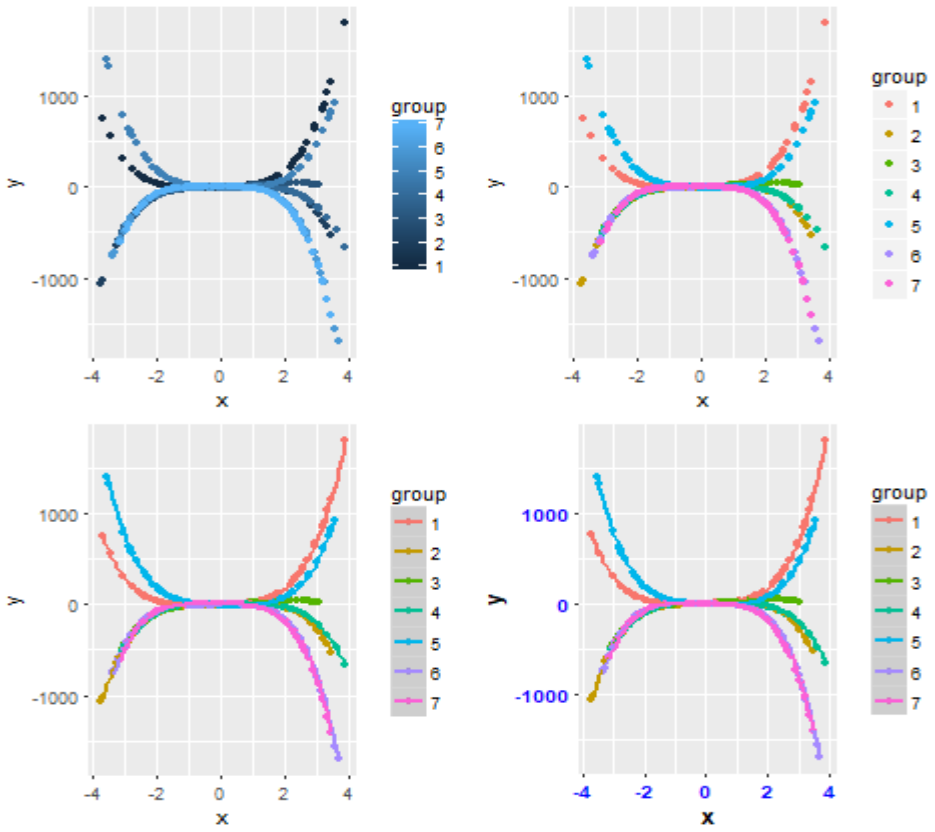
二次项：



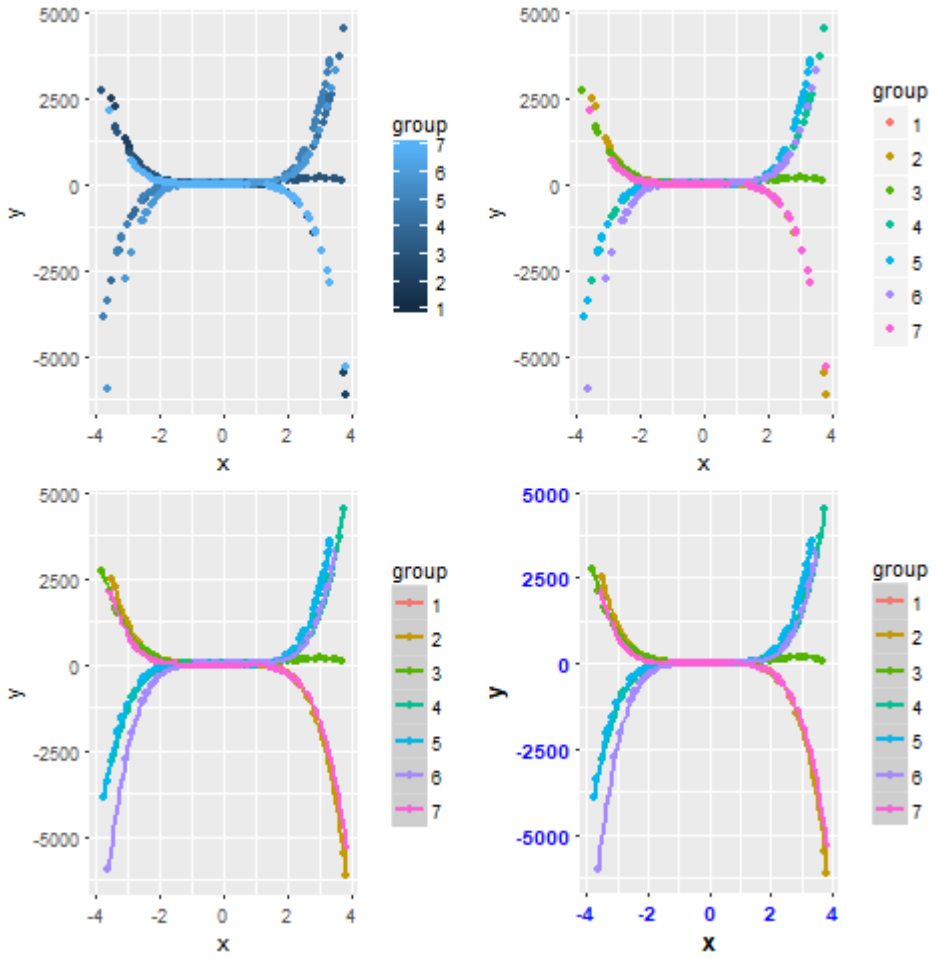
三次项：


group
1
2
3
4
5
6
7
y
x
250
0
-250
-4
-2
0
2
4

四次项：


group
1
2
3
4
5
6
7
y
x
1000
0
-1000
-4
-2
0
2
4

五次项：



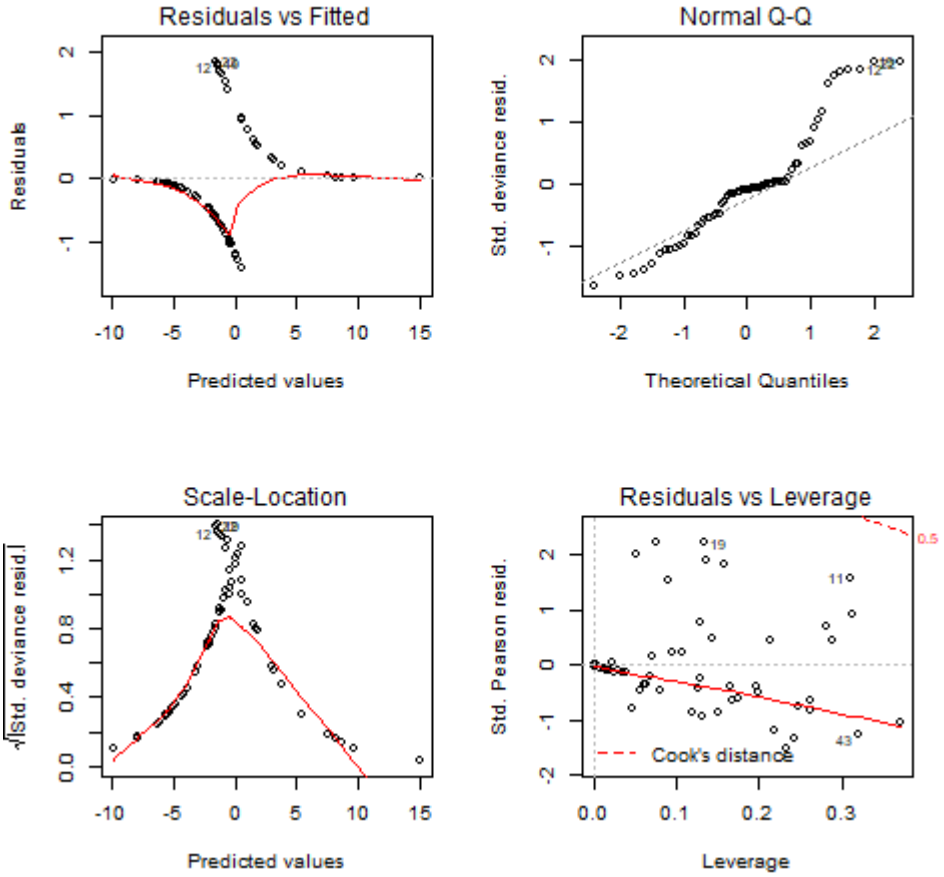
2.2

```
> sam.row.name #查看抽中的数据行探针 id
 [1] "206814_at"   "221165_s_at" "1561164_at"  "215663_at"   "204262_
s_at"
 [6] "204133_at"   "238503_at"   "223459_s_at" "237285_at"   "1570196
_at"
> summary(glm0)

Call:
glm(formula = y ~ ., family = binomial(link = "logit"), data = data2)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.59499  -0.54581  -0.06736   0.08030   2.21008

Coefficients:
```

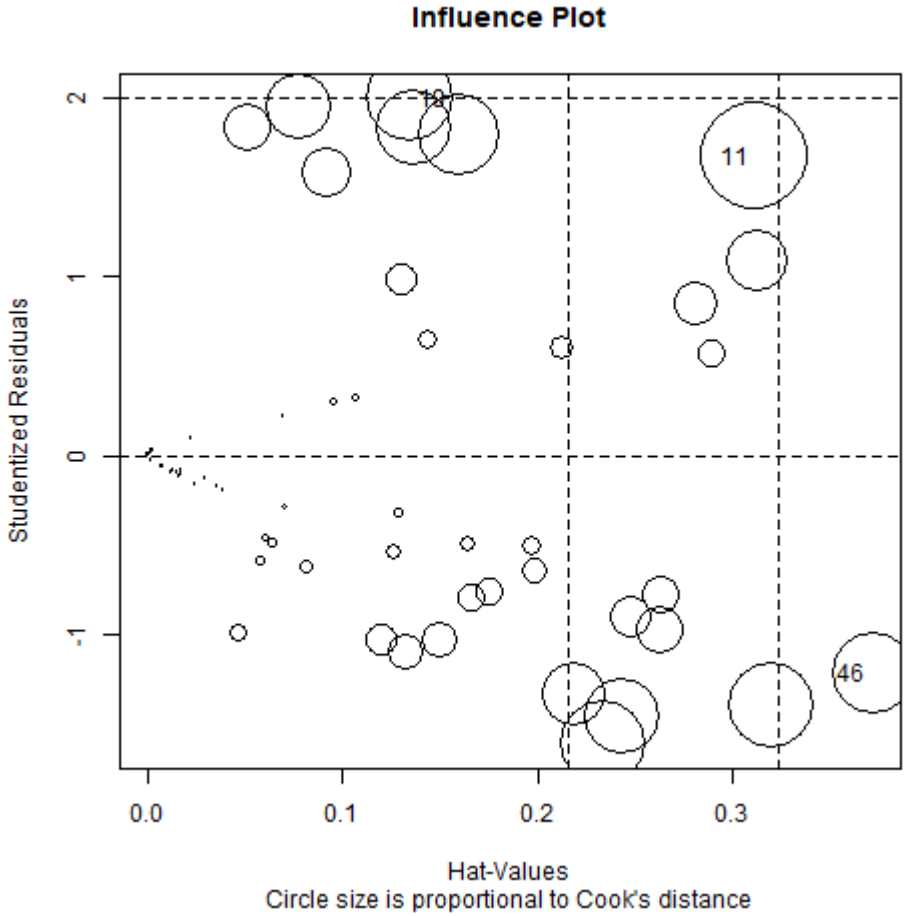
```
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  28.8084    11.3334   2.542   0.0110 *
x1           -2.3585     0.9316  -2.532   0.0114 *
x2            0.4322     0.5999   0.720   0.4713
x3           -1.0963     0.5354  -2.048   0.0406 *
x4           -1.7792     0.7738  -2.299   0.0215 *
x5           -2.5481     1.2007  -2.122   0.0338 *
x6           -1.5590     1.0564  -1.476   0.1400
x7           -1.9033     1.0824  -1.758   0.0787 .
x8           -0.1450     0.6846  -0.212   0.8322
x9            0.4893     0.4631   1.057   0.2907
x10           0.3227     0.4298   0.751   0.4528
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1
 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 84.473  on 64  degrees of freedom
Residual deviance: 38.788  on 54  degrees of freedom
AIC: 60.788

Number of Fisher Scoring iterations: 7
> summary(glm.step)

Call:
glm(formula = y ~ x1 + x3 + x4 + x5 + x6 + x7, family = binomial(link
 = "logit"),
    data = data2)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.43363  -0.58136  -0.09042   0.09105   1.86312

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  28.4434     9.3170   3.053  0.00227 **
```

```
x1           -2.0871     0.7765  -2.688  0.00719 **
x3           -0.8778     0.4703  -1.867  0.06197 .
x4           -1.4976     0.6194  -2.418  0.01561 *
x5           -2.7271     1.1079  -2.461  0.01384 *
x6           -1.4368     0.9142  -1.572  0.11602
x7           -1.5511     0.8132  -1.907  0.05647 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1
 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 84.473  on 64  degrees of freedom
Residual deviance: 40.950  on 58  degrees of freedom
AIC: 54.95

Number of Fisher Scoring iterations: 7
```

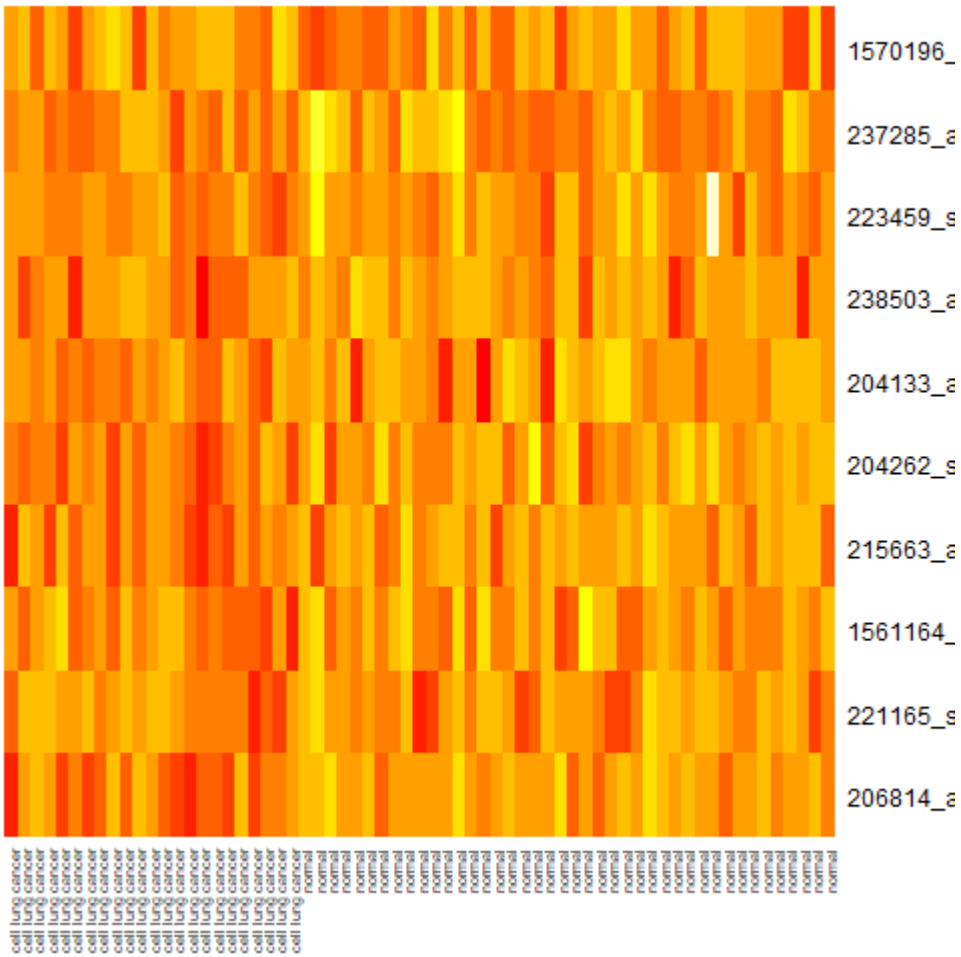
Glm4:



influencePlot


Influence Plot
Studentized Residuals
Hat-Values
Circle size is proportional to Cook's distance
11
46
2
1
0
-1
0.0
0.1
0.2
0.3

Heatmap



2.5

```
> summary(glm0)

Call:
glm(formula = y ~ ., family = binomial(link = "logit"), data = data3)

Deviance Residuals:
       Min          1Q      Median          3Q         Max
-1.620e-05  -2.110e-08  -2.110e-08   2.110e-08   1.708e-05

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  6.992e+01  2.860e+05       0        1
x1           6.600e+00  7.486e+04       0        1
x2           4.463e+00  3.681e+04       0        1
x3          -8.040e+00  5.971e+04       0        1
```

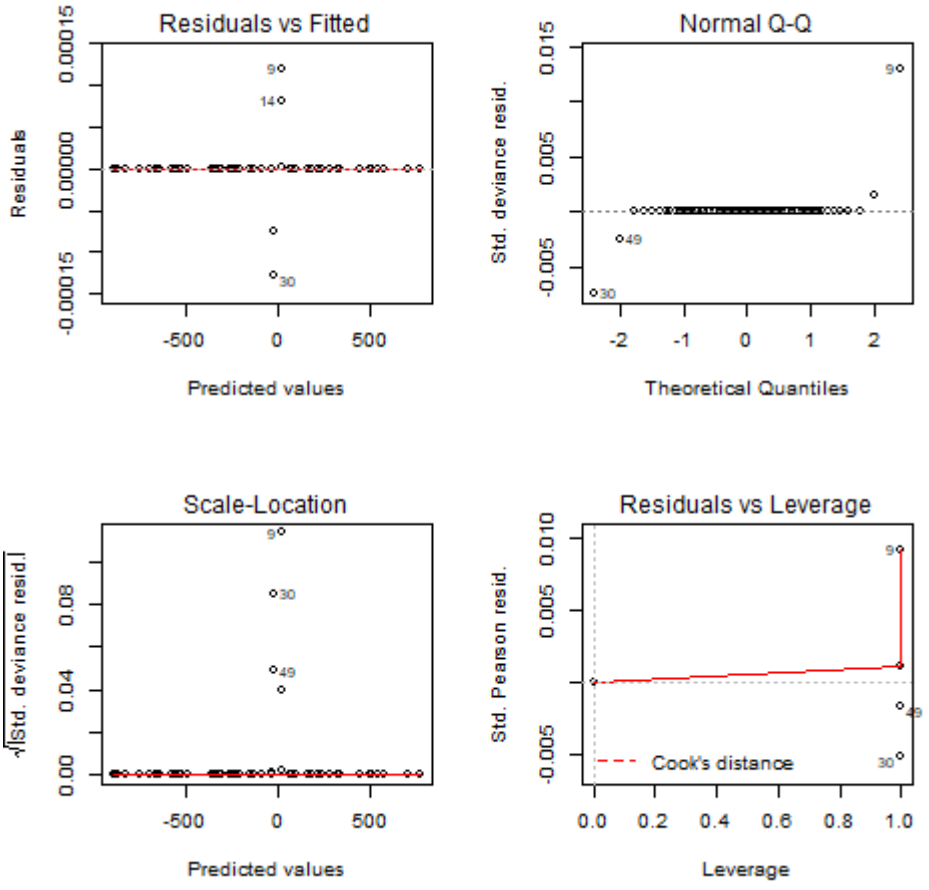
```
x4           1.810e+00  5.052e+04       0        1
x5           9.990e+00  9.227e+04       0        1
x6          -1.491e-01  1.221e+05       0        1
x7           3.566e+00  8.888e+04       0        1
x8          -1.010e+01  6.846e+04       0        1
x9          -9.567e+00  1.025e+05       0        1
x10         -4.819e+00  8.878e+04       0        1
x11          6.841e-01  6.159e+04       0        1
x12         -2.107e+01  1.383e+05       0        1
x13          1.160e+00  1.058e+05       0        1
x14         -2.037e-01  9.773e+04       0        1
x15         -7.983e+00  1.137e+05       0        1
x16          7.027e+00  5.500e+04       0        1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 8.4473e+01  on 64  degrees of freedom
Residual deviance: 1.7436e-09  on 48  degrees of freedom
AIC: 34

Number of Fisher Scoring iterations: 25
> summary(glm.step)

Call:
glm(formula = y ~ x2 + x7 + x12, family = binomial(link = "logit"),
    data = data3)

Deviance Residuals:
       Min          1Q      Median          3Q         Max
-1.290e-04  -2.100e-08  -2.100e-08   2.100e-08   1.206e-04

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    563.86  122833.56   0.005    0.996
x2              39.29    8983.90   0.004    0.997
x7              67.44   15496.15   0.004    0.997
x12           -222.10   48861.92  -0.005    0.996
```

(Dispersion parameter for binomial family taken to be 1)

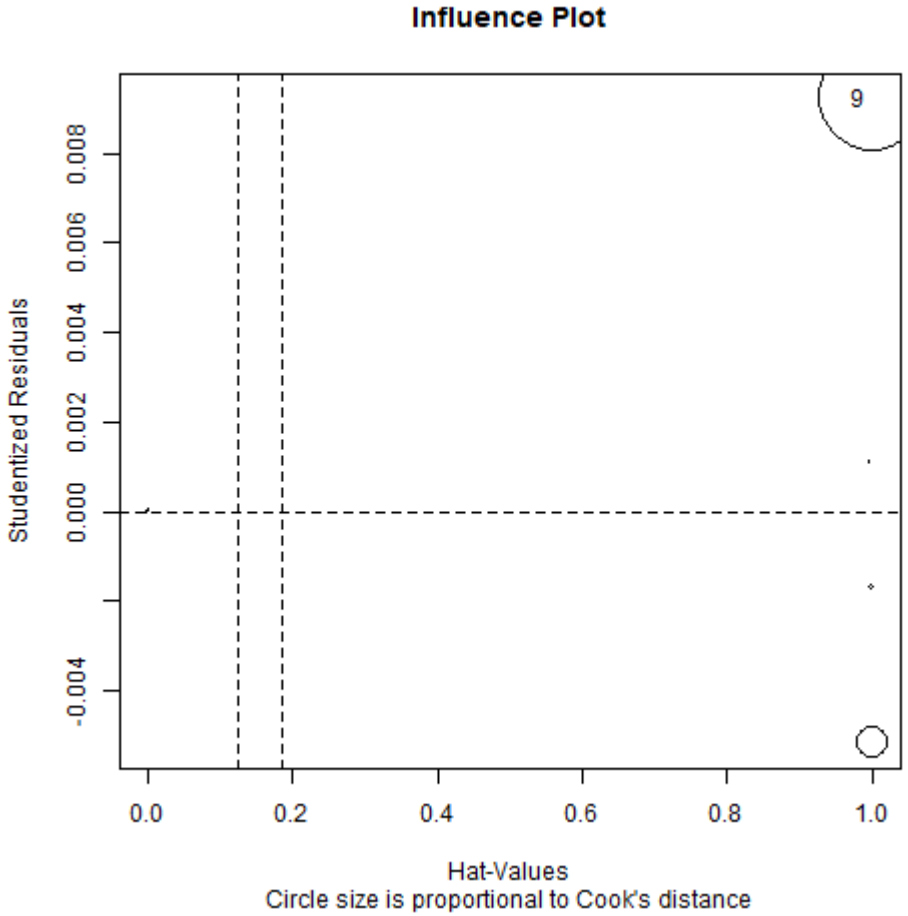
    Null deviance: 8.4473e+01 on 64 degrees of freedom
Residual deviance: 4.3524e-08 on 61 degrees of freedom
AIC: 8

Number of Fisher Scoring iterations: 25

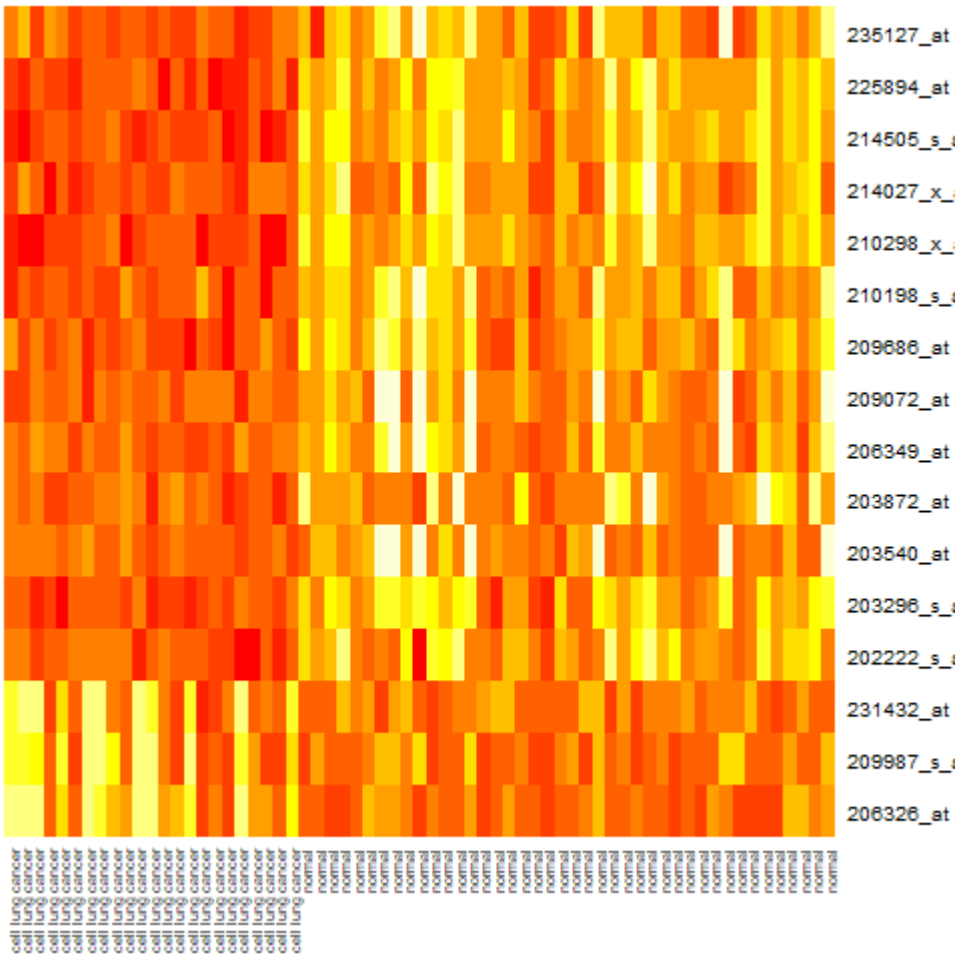
lec11_ICU_glm.png :



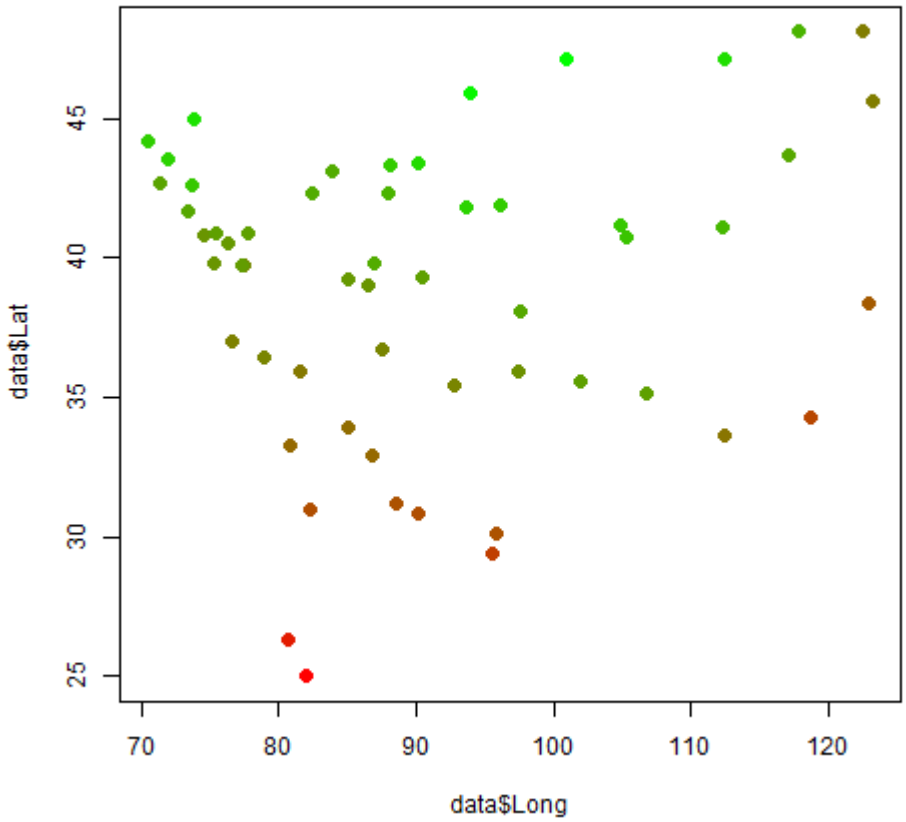
influencePlot.png


Influence Plot
9
Studentized Residuals
0.008
0.006
0.004
0.002
0.000
-0.004
0.0
0.2
0.4
0.6
0.8
1.0
Hat-Values
Circle size is proportional to Cook's distance

heatmap.png


235127_at
225894_at
214505_s_
214027_x_
210298_x_
210198_s_
209686_at
209072_at
206349_at
203872_at
203540_at
203296_s_
202222_s_
231432_at
209987_s_
206326_at
cell lung cancer
normal

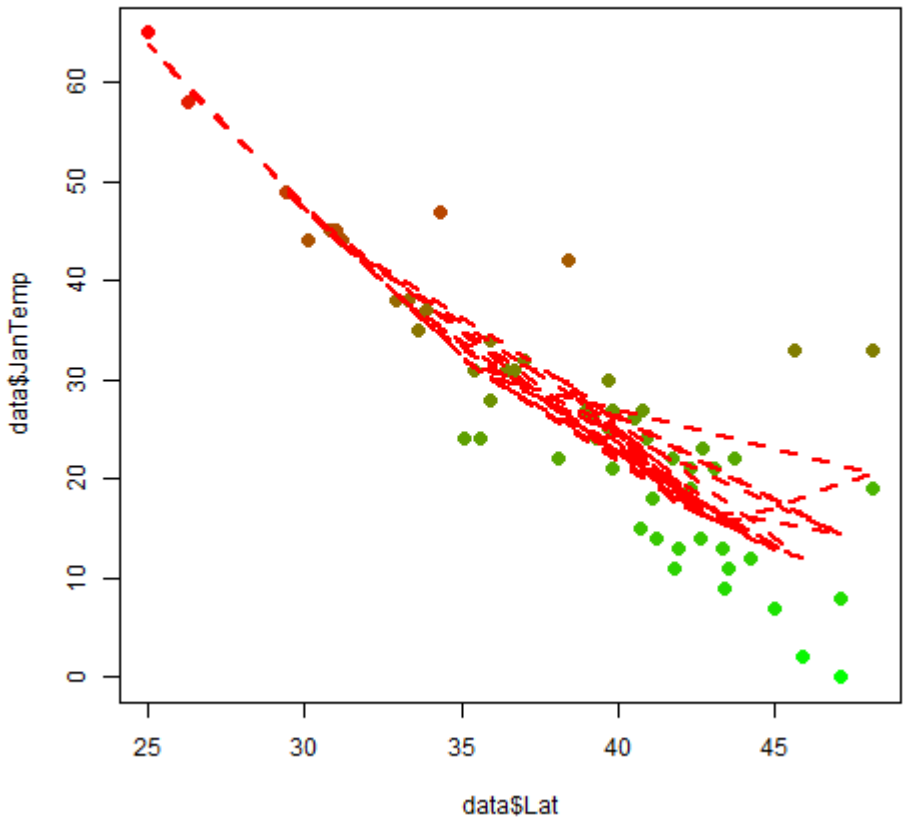
3.1



3.2 (1)

```
> summary(model1)
Call:
loess(formula = JanTemp ~ Lat, data = data, span = 0.4)

Number of Observations: 56
Equivalent Number of Parameters: 8.64
Residual Standard Error: 6.613
Trace of smoother matrix: 9.55  (exact)

Control settings:
  span     :  0.4
  degree   :  2
  family   :  gaussian
  surface  :  interpolate        cell = 0.2
  normalize:  TRUE
```

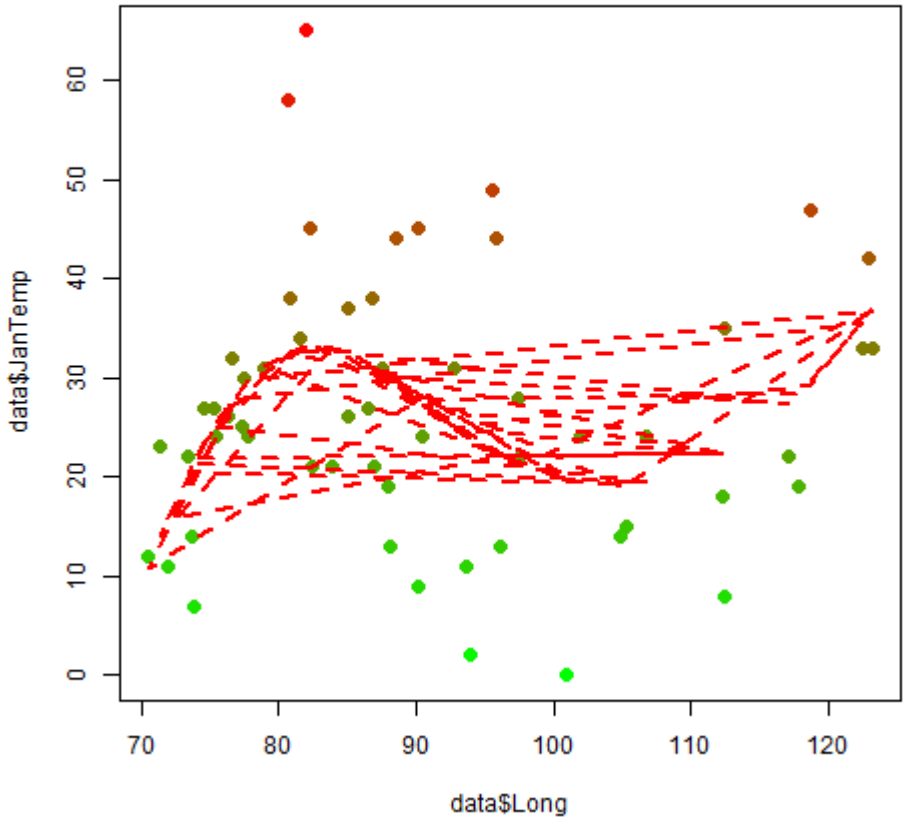
```
 parametric:   FALSE
drop.square:   FALSE
```



3.2 (2)

```
> summary(model2)
Call:
loess(formula = JanTemp ~ Long, data = data, span = 0.8)

Number of Observations: 56
Equivalent Number of Parameters: 4.33
Residual Standard Error: 12.08
Trace of smoother matrix: 4.73  (exact)

Control settings:
  span     :  0.8
```

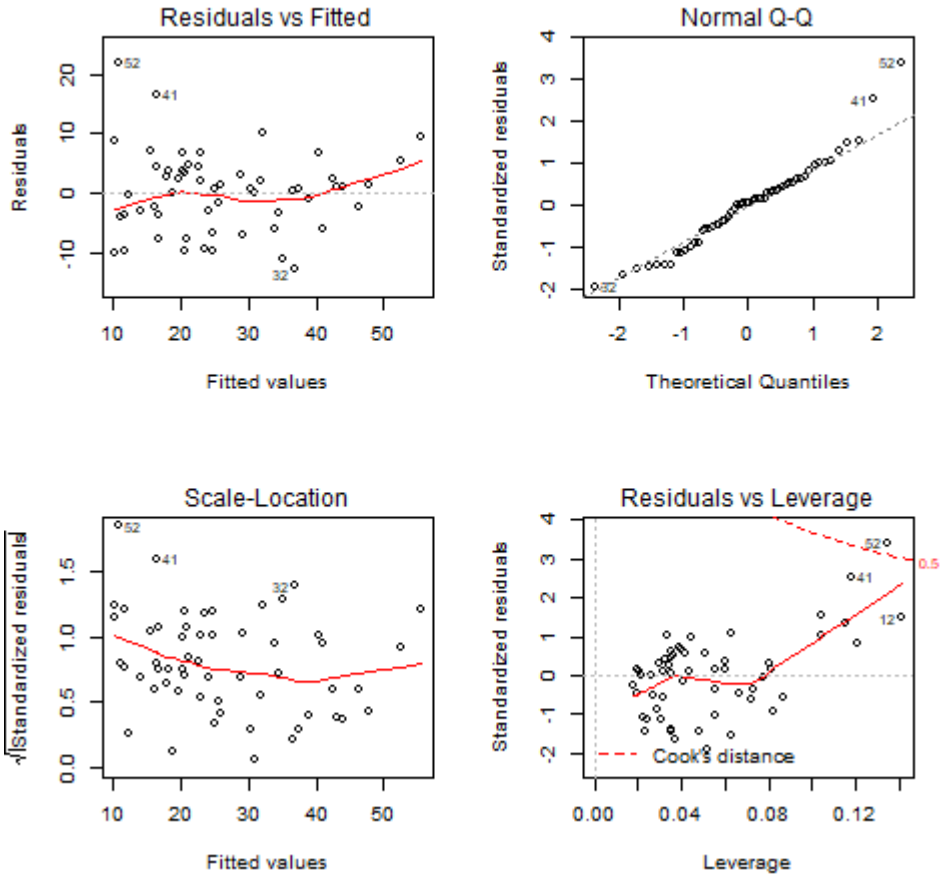
```
  degree   :  2
  family   :  gaussian
  surface  :  interpolate        cell = 0.2
  normalize:  TRUE
 parametric:  FALSE
drop.square:  FALSE
```



3.3

```
> summary(lm.line)

Call:
lm(formula = JanTemp ~ Lat + Long, data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-12.9983  -3.8957   0.5577   3.7330  22.0113
```

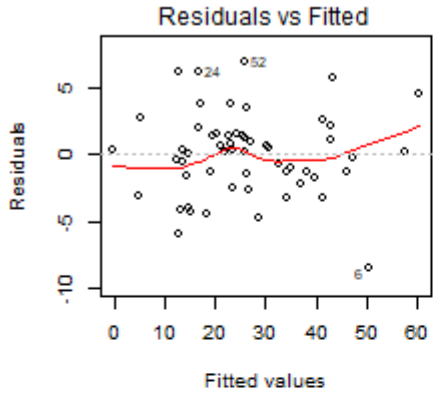
```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 98.64523    8.32708  11.846   <2e-16 ***
Lat         -2.16355    0.17570 -12.314   <2e-16 ***
Long         0.13396    0.06314   2.122   0.0386 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1
 ‘ ’ 1

Residual standard error: 6.935 on 53 degrees of freedom
Multiple R-squared:  0.7411,    Adjusted R-squared:  0.7314
F-statistic: 75.88 on 2 and 53 DF,  p-value: 2.792e-16
```

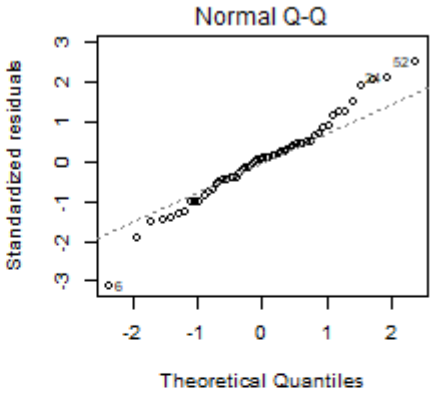


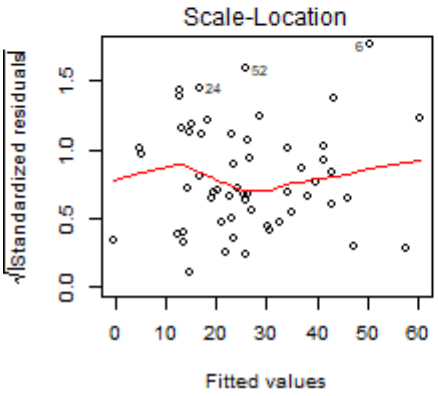
3.4

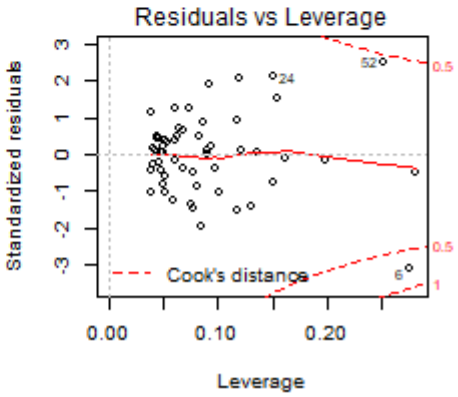
```
> summary(model)

Call:
```

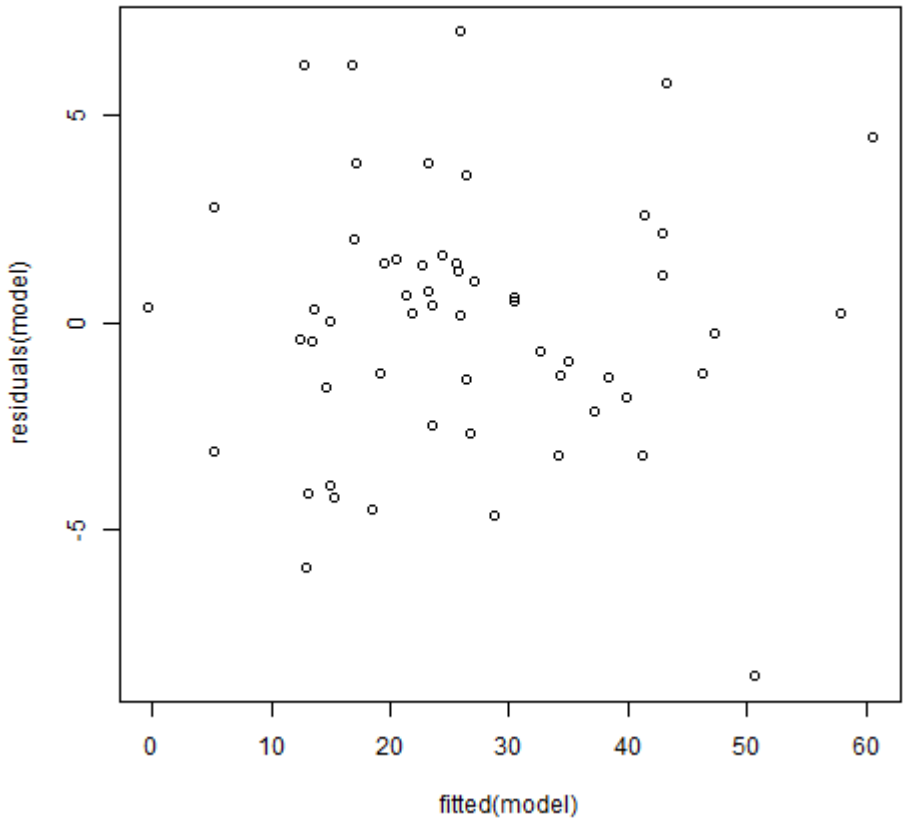
```
lm(formula = JanTemp ~ Lat + poly(Long, 3), data = data)

Residuals:
   Min     1Q Median     3Q    Max
-8.569 -1.624  0.218  1.472  7.039

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    118.39739    3.53301  33.512  < 2e-16 ***
Lat             -2.35772    0.08998 -26.202  < 2e-16 ***
poly(Long, 3)1  15.99052    3.26685   4.895 1.03e-05 ***
poly(Long, 3)2  36.26524    3.47734  10.429 3.02e-14 ***
poly(Long, 3)3  27.59874    3.30506   8.350 4.13e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1
 ‘ ’ 1

Residual standard error: 3.225 on 51 degrees of freedom
Multiple R-squared:  0.9461,   Adjusted R-squared:  0.9419
F-statistic: 223.9 on 4 and 51 DF,  p-value: < 2.2e-16
> confint(model, level=0.95)
                    2.5 %     97.5 %
(Intercept)    111.304571 125.490206
Lat             -2.538370  -2.177071
poly(Long, 3)1   9.432048  22.548985
poly(Long, 3)2  29.284192  43.246297
poly(Long, 3)3  20.963563  34.233918
```


Residuals vs Fitted
Residuals
Fitted values
24
52
6


Normal Q-Q
Standardized residuals
Theoretical Quantiles
52
24
6


Scale-Location
√|Standardized residuals|
Fitted values
6
52
24


Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
52
24
6
0.5
0.5
1



四、讨论：

1.1，一元多次项模拟结果接近于其对应的函数曲线，如一元二次项结果就是抛物线。

2.6

通过随机筛选的数据，其应当是有显著性差异的，但不论从回归图还是热图都不明显，3.5热图很明显，但回归没什么意义，因此可以看出Logistic回归不适用于这种数据。

3.5

二元线性回归从数据和图看来，其主要影响因素是纬度，经度虽然有一点贡献，但相比纬度还是很小，其结果也较为符合统计学意义，但可以看出细微趋

势，不是特别好，QQ图偏离对角线较多，也不是很好。而下面的方法得到的R值为0.94相比于上面的0.74要好得多，也客观准确的定义了经度的贡献。残差图也较为均匀。QQ图接近于对角线。因此后者的拟合度更好。