Lecture6
计数数据分析

刘晨辉
邮箱：chenhuiliu@hnu.edu.cn
办公室：土木楼A422

2023.04.04

目录

1. 泊松分布
2. 泊松回归
3. 广义线性模型

1. 泊松分布(Poisson Distribution)

变量分类：

- 定量变量(quantitative)：有具体数值含义
- 定性变量(qualitative, categorical)：分类数据，两分类与多分类

有些数据既不是传统的定量数据，也不是定性数据。比如交通事故的死亡人数，每小时共享单车被租用的次数，它有什么特点?

- 非负
- 整数

这类数据，我们称之为计数数据(Count Data)。用线性回归模型分析是否合适?

```
```{r libraries}
library(ggplot2)
library(ISLR2)
philly<- read.csv("Philly.csv")
data("Bikeshare")

```
```

```
```{r Check the data}
### 1. 查看数据
## 查看Injury分布
print(table(philly$Injury))

```
```

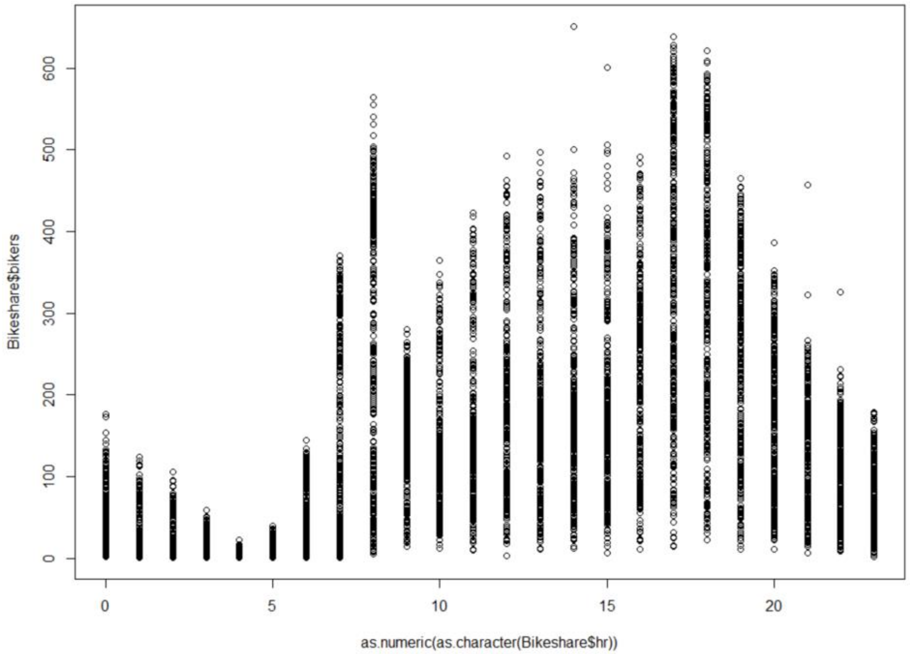
1. 泊松分布(Poisson Distribution)

对Bikeshare数据中的bikers建立线性回归模型

```
## 对bikers建立线性回归模型
lm0<- lm(bikers~workingday+temp+weathersit+mnth+hr,
         data = Bikeshare)
# 查看模型估计结果
print(summary(lm0$fitted.values))

# 查看每小时的bikers分布
print(plot(as.numeric(as.character(Bikeshare$hr)),
           Bikeshare$bikers))
```

```
> print(summary(lm0$fitted.values))
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
-143.74   61.89  142.38  143.79  221.90  441.36 
```



$Y = \beta_0 + \beta_1 * X_1$，出现的问题?

- 预测结果有负值
- 预测结果不为整数
- bikers方差显然随小时有明显分布差异

如果$\log(Y) = \beta_0 + \beta_1 * X_1$? 也有问题：结果解释，以及$Y = 0$的情况

1. 泊松分布(Poisson Distribution)

假设随机变量Y为非负整数，$Y \in \{0,1,2,\ldots\}$。如果Y服从泊松分布，则

$$P(Y=k)=\frac{e^{-\lambda}\lambda^k}{k!}$$

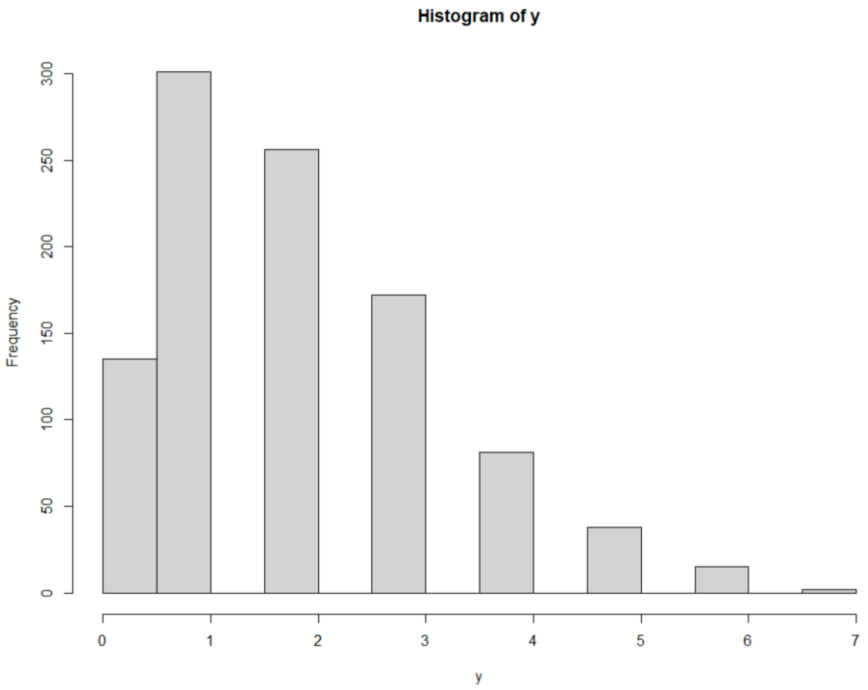
$$\text{for } k=0,1,2,\ldots$$

式中: $\lambda > 0$

对于泊松分布:

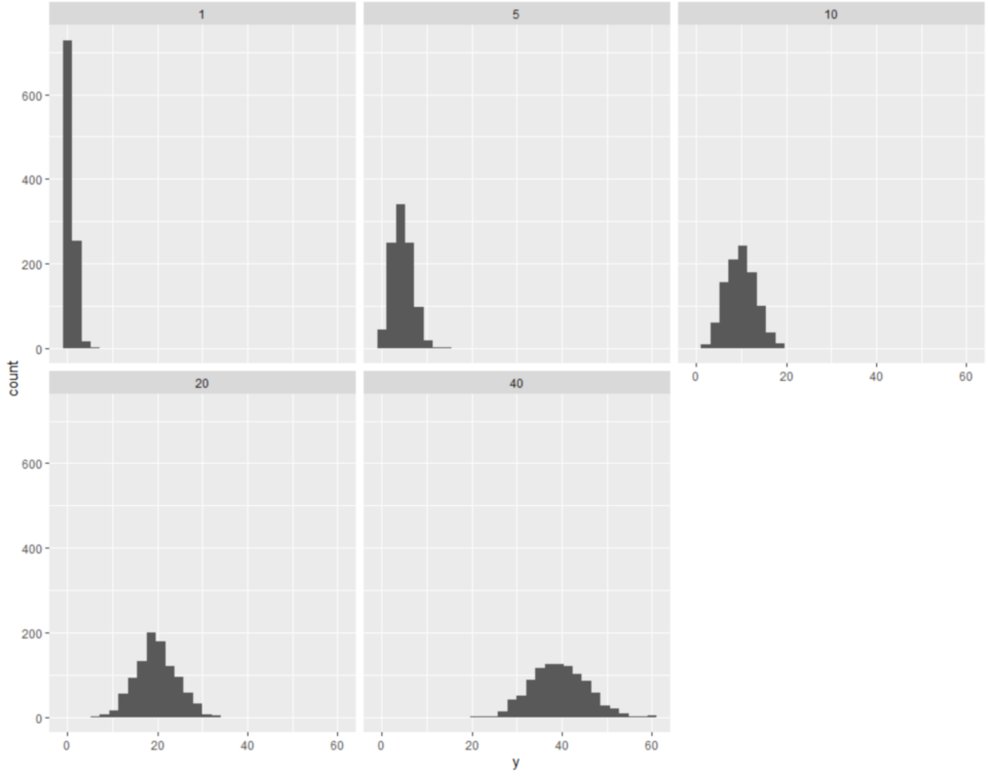
$$\lambda = E(Y) = Var(Y)$$

```r
### 1. 随机产生一个lambda为2的泊松数据集
y<- rpois(n=1000,lambda = 2)
print(hist(y))
```



1. 泊松分布(Poisson Distribution)

```
### 2. 创造不同lambda的数据集
lambdas<- c(1,5,10,20,40)
for(i in 1:length(lambdas)){
  y<- rpois(n=1000,lambda = lambdas[i])
  dat<- data.frame(y)
  dat$lambda<- lambdas[i]
  if(i==1){
    dats<- dat
  }else{
    dats<- rbind(dats,dat)
  }
  print(i)
}
print(head(dats))

## 画出不同lambda对应的直方图
g1<- ggplot(data = dats)+
  geom_histogram(aes(x=y))+
  facet_wrap(~lambda)
print(g1)
```



随着λ增大，泊松分布会趋向于正态分布。

1. 泊松分布(Poisson Distribution)

泊松分布：$P(y) = \frac{e^{-\lambda}\lambda^{y}}{k!}$

假设每次交通事故受伤符合一个$\lambda = 1$的泊松分布，那么每次交通事故受伤人数为0，1，2，不多于2人的概率分别为：

$$P(Y=0) = \frac{e^{-1} * 1^{0}}{0!} = 0.368$$

$$P(Y=1) = \frac{e^{-1} * 1^{1}}{1!} = 0.368$$

$$P(Y=2) = \frac{e^{-1} * 1^{2}}{2!} = 0.184$$

$$P(Y \le 2) = P(Y=0) + P(Y=1) + P(Y=2)$$

```
## 计算概率
#概率密度函数
p1<- dpois(x=0,lambda = 1)
print(p1)
p2<- dpois(x=1,lambda = 1)
print(p2)
p3<- dpois(x=2,lambda = 1)
print(p3)
# 分布函数F(x)
p4<- ppois(q=2,lambda = 1)
print(p4)
```

假设我们有n个观察值(y_i)来自于泊松回归模型，则其似然公式为

$$l(\lambda) = \prod_{i=1}^{n} \frac{e^{-\lambda}\lambda^{y_i}}{y_i!}, \lambda = \frac{\sum_{i=1}^{n} y_i}{n}$$

2. 泊松回归(Poisson Regression)

对于泊松回归，我们对平均值λ进行建模，即

$$P(Y=k)=\frac{e^{-\lambda}\lambda^{k}}{k!}$$

$$log(\lambda)=\beta_0+\beta_1*X_1+\cdots+\beta_p*X_p$$

也就是$\lambda=e^{(\beta_0+\beta_1*X_1+\cdots+\beta_p*X_p)}$。

式中：β_0，β_1，…，β_p为需要估计的回归系数。

假设我们有n个观察值(x_i,y_i)来自于泊松回归模型，则其似然公式为

$$l(\beta_0,\ \beta_1,\ \ldots,\ \beta_p)=\prod_{i=1}^{n}\frac{e^{-\lambda_i}{\lambda_i}^{y_i}}{y_i!}$$

$$\lambda_i=e^{(\beta_0+\beta_1*x_{i1}+\cdots+\beta_p*x_{ip})}$$

$$x_i=(x_{i1},x_{i2},\ldots,x_{ip})$$

2. 泊松回归(Poisson Regression)

对Philly数据中的Injury分别建立线性回归模型与泊松回归模型

```
## 线性回归
lm1<- lm(Injury~Intersection+Daytime+Weather,
          data = philly)
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.914053   0.007343 124.472  < 2e-16 ***
Intersection  0.237813   0.007134  33.336  < 2e-16 ***
Daytime       0.088326   0.007392  11.949  < 2e-16 ***
WeatherRain  -0.076644   0.009981  -7.679 1.62e-14 ***
WeatherSnow  -0.194640   0.025402  -7.662 1.84e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.032 on 85705 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared:  0.01639,	Adjusted R-squared:  0.01635
F-statistic: 357.1 on 4 and 85705 DF,  p-value: < 2.2e-16
```

```
## 泊松回归
poisson1<- glm(Injury~Intersection+Daytime+Weather,
               data = philly,
               family = poisson)
```

```
Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.085487   0.007181 -11.905  < 2e-16 ***
Intersection  0.222348   0.006768  32.855  < 2e-16 ***
Daytime       0.082304   0.006956  11.832  < 2e-16 ***
WeatherRain  -0.072198   0.009503  -7.597 3.03e-14 ***
WeatherSnow  -0.198012   0.026045  -7.603 2.90e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 79072  on 85709  degrees of freedom
Residual deviance: 77658  on 85705  degrees of freedom
  (2 observations deleted due to missingness)
AIC: 222347
```

估计模型:

- 线性回归模型:

$$Y = 0.914053 + 0.237813 * Intersection + 0.088326 * Daytime - 0.076644 * WeatherRain - 0.194640 * WeatherSnow$$

- 泊松回归模型:

$$\lambda = e^{0.914053+0.237813*Intersection+0.088326*Daytime-0.076644*WeatherRain-0.194640*WeatherSnow}$$

2. 泊松回归(Poisson Regression)

对Philly数据中的Injury分别建立线性回归模型与泊松回归模型

```
## 线性回归
lm1<- lm(Injury~Intersection+Daytime+Weather,
         data = philly)
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)   0.914053   0.007343 124.472  < 2e-16 ***
Intersection  0.237813   0.007134  33.336  < 2e-16 ***
Daytime       0.088326   0.007392  11.949  < 2e-16 ***
WeatherRain  -0.076644   0.009981  -7.679 1.62e-14 ***
WeatherSnow  -0.194640   0.025402  -7.662 1.84e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.032 on 85705 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared:  0.01639,	Adjusted R-squared:  0.01635 
F-statistic: 357.1 on 4 and 85705 DF,  p-value: < 2.2e-16
```

```
## 泊松回归
poisson1<- glm(Injury~Intersection+Daytime+Weather,
               data = philly,
               family = poisson)
```

```
Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept)  -0.085487   0.007181 -11.905  < 2e-16 ***
Intersection  0.222348   0.006768  32.855  < 2e-16 ***
Daytime       0.082304   0.006956  11.832  < 2e-16 ***
WeatherRain  -0.072198   0.009503  -7.597 3.03e-14 ***
WeatherSnow  -0.198012   0.026045  -7.603 2.90e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 79072  on 85709  degrees of freedom
Residual deviance: 77658  on 85705  degrees of freedom
  (2 observations deleted due to missingness)
AIC: 222347
```

参数解释：

- Intersection：
  - 线性回归：发生在交叉口的交通事故平均受伤人数比非交叉口多0.237813人。
  - 泊松回归：发生在交叉口的交通事故平均受伤人数是非交叉口的 $\exp(0.222348) = 1.249006$ 倍
- Daytime：
  - 线性回归：发生在交叉口的交通事故平均受伤人数比非交叉口多_______人?
  - 泊松回归：发生在交叉口的交通事故平均受伤人数是非交叉口的_______倍?

2. 泊松回归(Poisson Regression)

对Philly数据中的Injury分别建立线性回归模型与泊松回归模型

```
## 线性回归
lm1<- lm(Injury~Intersection+Daytime+Weather,
         data = philly)
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)   0.914053   0.007343 124.472  < 2e-16 ***
Intersection  0.237813   0.007134  33.336  < 2e-16 ***
Daytime       0.088326   0.007392  11.949  < 2e-16 ***
WeatherRain  -0.076644   0.009981  -7.679 1.62e-14 ***
WeatherSnow  -0.194640   0.025402  -7.662 1.84e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.032 on 85705 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared:  0.01639,	Adjusted R-squared:  0.01635 
F-statistic: 357.1 on 4 and 85705 DF,  p-value: < 2.2e-16
```

```
## 泊松回归
poisson1<- glm(Injury~Intersection+Daytime+Weather,
               data = philly,
               family = poisson)
```

```
Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept)  -0.085487   0.007181 -11.905  < 2e-16 ***
Intersection  0.222348   0.006768  32.855  < 2e-16 ***
Daytime       0.082304   0.006956  11.832  < 2e-16 ***
WeatherRain  -0.072198   0.009503  -7.597 3.03e-14 ***
WeatherSnow  -0.198012   0.026045  -7.603 2.90e-14 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 79072  on 85709  degrees of freedom
Residual deviance: 77658  on 85705  degrees of freedom
  (2 observations deleted due to missingness)
AIC: 222347
```

平均值-方差关系：

- 线性回归：方差恒定，σ^2。
- 泊松回归：方差等于平均值，也就是说方差是会变化的，$Var(Y) = \lambda = e^{(\beta_0+\beta_1 * X_1+\cdots+\beta_p * X_p)}$。

平均值-方差关系：

- 线性回归：可能为负值。
- 泊松回归：总是非负值。

3. 广义线性模型(Generalized Linear Model)

- 线性回归模型(Linear Regression):

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

$$E(Y|X_1, \ldots, X_p) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

- 逻辑回归模型(Logistic Regression):

$$Y \sim Ber(p)$$

$$\log\left(\frac{E(Y|X_1, \ldots, X_p)}{1 - E(Y|X_1, \ldots, X_p)}\right) = \log(\frac{p}{1-p}) = \beta_0 + \beta_1 * X_1 + \cdots + \beta_p * X_p$$

- 泊松回归模型(Poisson Regression):

$$Y \sim Poi(\lambda)$$

$$\log(E(Y|X_1, \ldots, X_p)) = \log(\lambda(X_1, \ldots, X_p)) = \beta_0 + \beta_1 * X_1 + \cdots + \beta_p * X_p$$

3. 广义线性模型(Generalized Linear Model)

上面均对$E(Y|X_1,\dots,X_p)$进行一些变换，使得其成为预测变量的线性函数

- 线性回归：$E(Y|X_1,\dots,X_p)=\beta_0+\beta_1X_1+\cdots+\beta_pX_p$
- 逻辑回归： $\log\left(\frac{E(Y|X_1,\dots,X_p)}{1-E(Y|X_1,\dots,X_p)}\right)=\beta_0+\beta_1*X_1+\cdots+\beta_p*X_p$
- 泊松回归：$\log(E(Y|X_1,\dots,X_p))=\log(\lambda(X_1,\dots,X_p))=\beta_0+\beta_1*X_1+\cdots+\beta_p*X_p$

这个变换函数我们称之为连接函数(Link Function)，η。

$$\eta\left(E(Y|X_1,\dots,X_p)=\mu\right)=\beta_0+\beta_1X_1+\cdots+\beta_pX_p$$

线性回归：$\eta(\mu)=\mu$

逻辑回归：$\eta(\mu)=\log(\frac{\mu}{1-\mu})$

线性回归：$\eta(\mu)=\log(\mu)$

3. 广义线性模型(Generalized Linear Model)

指数模型家族(Exponential Family)：

- 高斯分布(Gaussian Distribution)
- 伯努利分布(Bernoulli Distribution)
- 泊松分布(Poisson Distribution)
- 负二项分布(Negative Distribution)
- 指数分布(Exponential Distribution)
- 伽马分布(Gamma Distribution)
- …

- An exponential family distribution has the following form,

$$p(x \mid \eta) = h(x) \exp\{\eta^\top t(x) - a(\eta)\} \tag{1}$$

- The different parts of this equation are
  - The natural parameter η
  - The sufficient statistic $t(x)$
  - The underlying measure $h(x)$, e.g., counting measure or Lebesgue measure
  - The log normalizer $a(\eta)$,

$$a(\eta) = \log \int h(x) \exp\{\eta^\top t(x)\}. \tag{2}$$

Here we integrate the unnormalized density over the sample space. This ensures that the density integrates to one.

Source: Prof. David M. Blei

广义线性模型(Generalized Linear Model, GLM)：如果因变量Y符合指数模型家族的某种分布，且可以将其平均值表示为自变量的线性组合，那么这个回归模型就被称为广义线性模型。

4. 多项式(Polynomial)

用Auto数据集，回顾线性回归模型：lm vs. glm。

```r
## 线性回归：lm vs. glm
lm1<- lm(mpg~horsepower,data = Auto)
print(lm1)

glm1<- glm(mpg~horsepower,data = Auto)
print(glm1)
```

注意：glm中的family没有做任何设置！

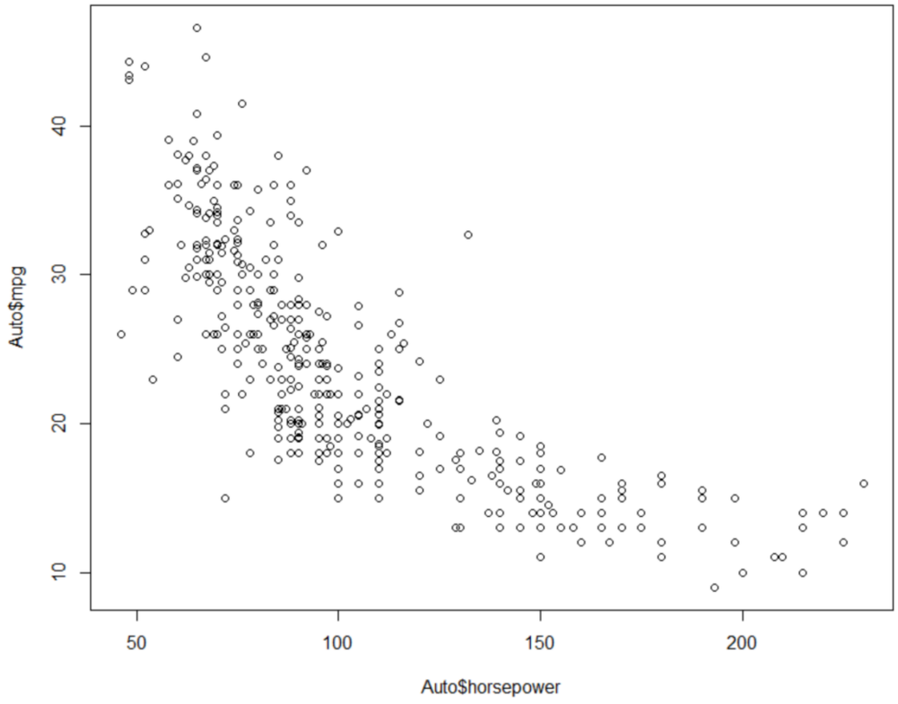
```
> print(lm1)

Call:
lm(formula = mpg ~ horsepower, data = Auto)

Coefficients:
(Intercept)   horsepower
    39.9359      -0.1578
```

```
> print(glm1)

Call:  glm(formula = mpg ~ horsepower, data = Auto)

Coefficients:
(Intercept)   horsepower
    39.9359      -0.1578
```

小知识1：lm与glm结果完全一致！

4. 多项式(Polynomial)

用Auto数据集，查看多项式(Polynomial)数据：



根据散点图，mpg与horsepower是否呈现线性关系，或者是其他关系？

4. 多项式(Polynomial)

用Auto数据集，查看多项式(Polynomial)数据：

```r
## 多项式(polynomial)数据创造：poly()和I()
# 一次方
a1<- poly(x=3, degree = 1,raw = TRUE)
print(a1)
# 二次方(Quadratic)
a2<- poly(x=3, degree = 2,raw = TRUE)
print(a2)
# 立方(Cubic)
a3<- poly(x=3, degree = 3,raw = TRUE)
print(a3)
# 四次方(Quartic)
a4<- poly(x=Auto$horsepower, degree = 4,raw = TRUE)
print(head(a4))

a5<- I(Auto$horsepower^4)
print(head(a5))
```

```
> print(a2)
     1 2
[1,] 3 9
attr(,"degree")
[1] 1 2
attr(,"class")
[1] "poly"   "matrix"
```

```
> print(a3)
     1 2  3
[1,] 3 9 27
attr(,"degree")
[1] 1 2 3
attr(,"class")
[1] "poly"   "matrix"
```

```
> head(a4)
       1     2       3          4
[1,] 130 16900 2197000  285610000
[2,] 165 27225 4492125  741200625
[3,] 150 22500 3375000  506250000
[4,] 150 22500 3375000  506250000
[5,] 140 19600 2744000  384160000
[6,] 198 39204 7762392 1536953616
> head(a5)
[1]  285610000  741200625  506250000  506250000  384160000 1536953616
```

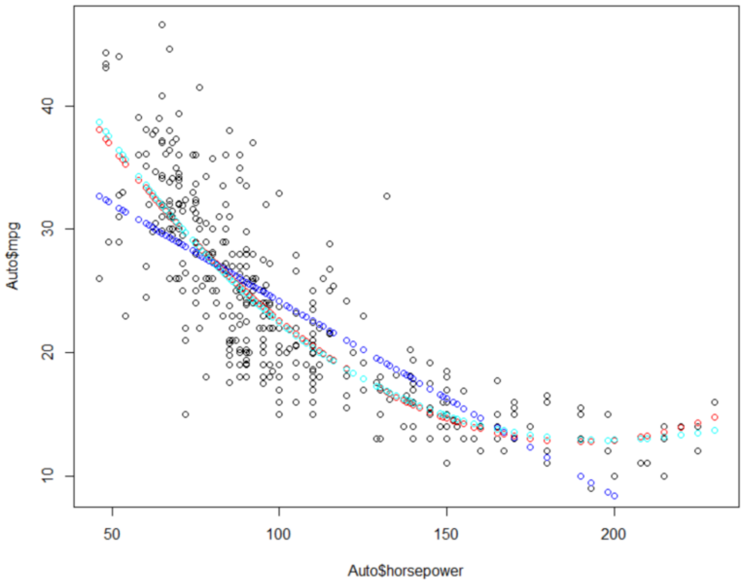
4. 多项式(Polynomial)

用Auto数据集，查看多项式(Polynomial)数据：

```
## 1.0 多项式公式分析mpg影响因素
# 模型1：一次方
glm1<- glm(mpg~horsepower,data = Auto)
# 模型2：二次方：I()
glm2<- glm(mpg~horsepower+I(horsepower^2),data = Auto)
# 模型3：三次方：poly()
glm3<- glm(mpg~poly(horsepower,3,raw = TRUE),data = Auto)
# 模型评价：AIC和MSE(Mean squared error)
aics<- c(glm1$aic,glm2$aic,glm3$aic)
mses<- c(mean((glm1$y - glm1$fitted.values)^2),
         mean((glm2$y - glm2$fitted.values)^2),
         mean((glm3$y - glm3$fitted.values)^2))
print(aics)
print(mses)

# 三个模型fit结果图示
plot1<- plot(x=Auto$horsepower,y=Auto$mpg)+
  points(x=Auto$horsepower,y=glm1$fitted.values,col="blue")+
  points(x=Auto$horsepower,y=glm2$fitted.values,col="red")+
  points(x=Auto$horsepower,y=glm3$fitted.values,col="cyan")
print(plot1)
```

```
> print(aics)
[1] 2363.324 2274.354 2275.531
> print(mses)
[1] 23.94366 18.98477 18.94499
```

根据散点图，mpg与horsepower的二次项拟合良好!



谢 谢！