模式识别基础 第一次作业

陈翰墨 自65

2016010302

2019.03.08

1. 名词解释

> 请对下列名词给出你的理解。

> 人工智能（Artificial Intelligence ）

人工智能概念较广泛，在计算机领域指，让计算机拥有像人一样或与人类相似的思考与行为的能力。

> 模式识别（Pattern recognition）

指观察外界事物（收集信息）后分类处理作出判断或决策的能力/行为。

> 机器学习（Machine Learning）

实现人工智能的途径之一：通过收集数据并用数据"训练"计算机从而拥有模式识别的能力。

> 深度学习（Deep Learning）

机器学习中的一类，有别于传统的机器学习，依靠多层神经网络完成机器学习。

> 统计学习（Statistical Learning）

机器学习的一类重要方法，基于统计学理论和大量数据进行的机器学习。

2. 证明

> 证明线性回归中的R^2 与相关系数 r 的关系
>
> $$R^2 = r^2 \tag{1}$$
>
> 其中$R^2 = \frac{\sum_{i=1}^{n}(\bar{y}-\hat{y}_i)^2}{\sum_{i=1}^{n}(\bar{y}-y_i)^2}, r = \frac{cov(x,y)}{\rho_x\rho_y}$，其中$x, y$ 均为一维向量

证明：

记$S_{xx} = \sum_{i=1}^{n}(x_i - \bar{x})^2, S_{yy} = \sum_{i=1}^{n}(y_i - \bar{y})^2, S_{xy} = \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$

则$R^2 = \frac{\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2}{S_{yy}}, r^2 = \frac{S_{xy}^2}{S_{xx}S_{yy}}$

$$R^2 = r^2 \iff S_{xx}\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2 = S_{xy}^2 \tag{2}$$

代入$\hat{y}_i = \bar{y} + \hat{\beta}_1(x_i - \bar{x})$,

得到$\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2 = \sum_{i=1}^{n}\hat{\beta}_1^2(x_i - \bar{x})^2$

而

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(\hat{y}_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} = \frac{S_{xy}}{S_{xx}} \tag{3}$$

代入即得 (2) 式从而得证。

3. 过拟合问题

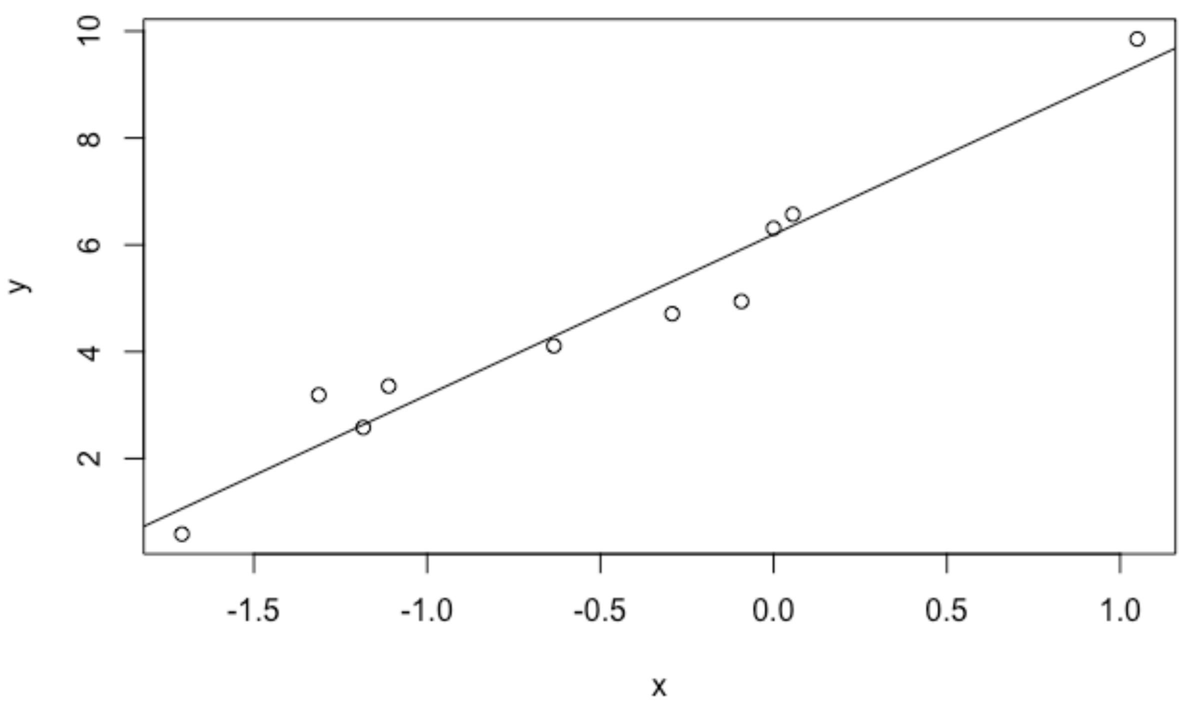
> 利用模型$y = \theta_1 x + \theta_0 + \epsilon$生成一组仿真数据$(x, y)$，其中$x$服从$N(0,1)$的正态分布。
> $\theta_1 = 3, \theta_0 = 6$。残差项$\epsilon$服从正态分布$N(0, \sigma^2)$，分别考虑$\sigma = 0.5$和2的情况。回答以下问题。

> 1. 随机生成10个训练样本数据，分别用线性模型，一元二次和一元三次模型对改组数据进行回归， 得到回归模型的参数，绘制散点图和回归曲线，计算RSS并比较大小。
> 2. 再随机生成100个测试样本，用（1）中的模型预测y值，并比较三种模型的预测效果。
> 3. 将（1）中的“随机生成10个训练样本数据”改为“随机生成100个训练样本数据”，重复步骤（1）（2）。
> 4. 请多次重复（1）-（3），对 σ 的取值、模型复杂程度、训练样本量和模型效果之间的关系进行总结。

(1)

$\sigma = 0.5$

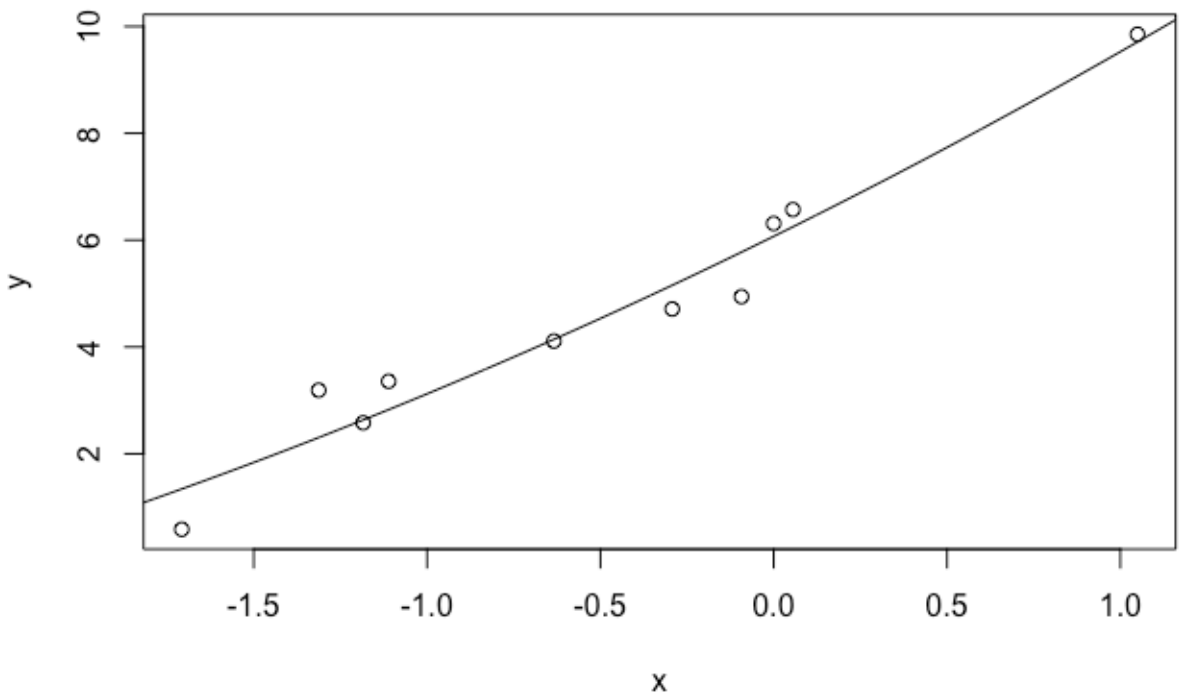
- 线性模型



$y = 5.998 + 2.891x$

Square Sum of Residuals (SSR):2.03

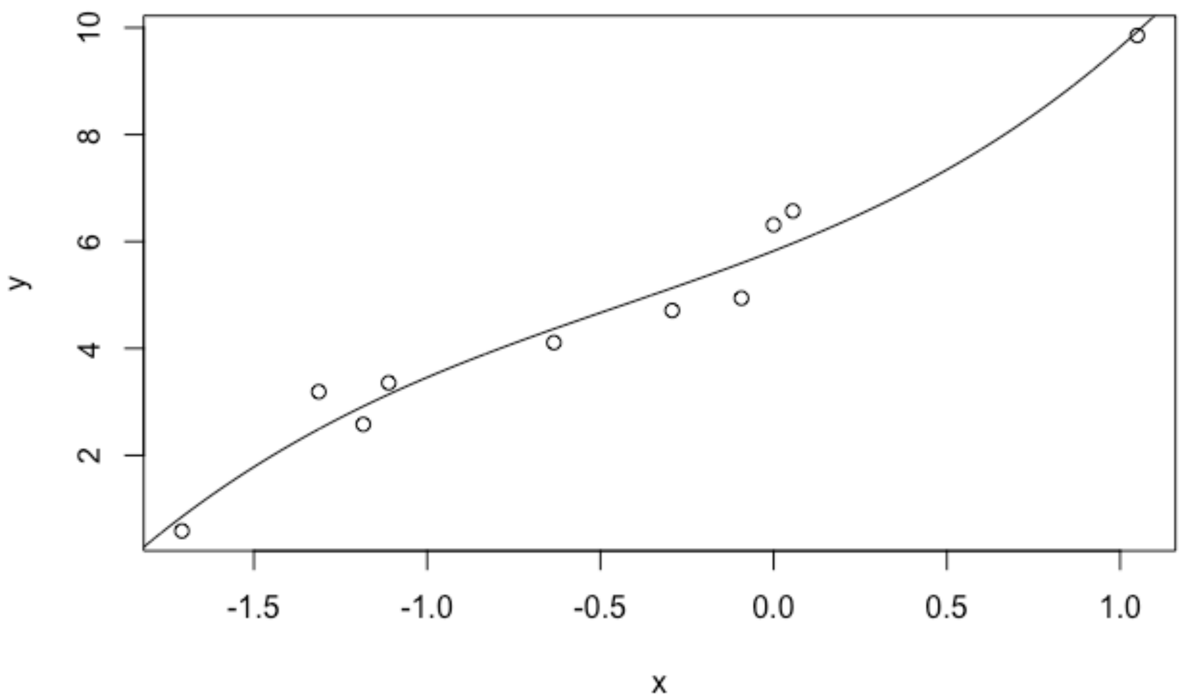
- 二次模型



$y = 5.7817 + 3.0262x + 0.3989x^2$

Square Sum of Residuals (SSR): 1.20

- 三次模型

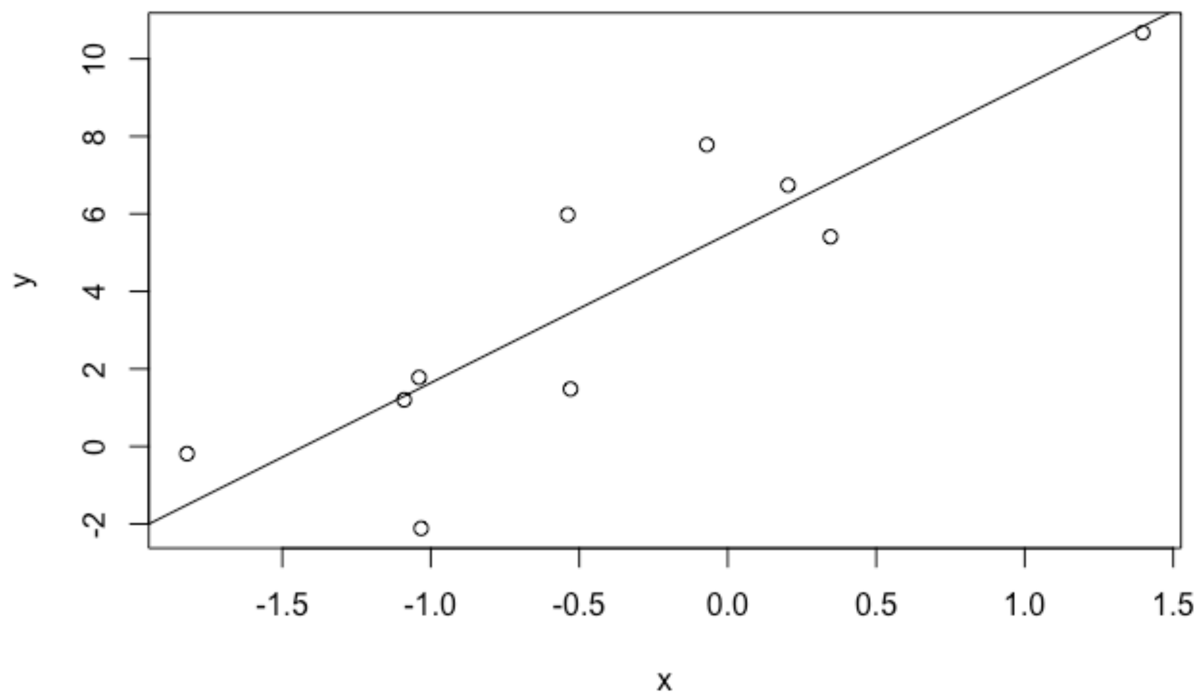


$y = 5.7754 + 2.3989x + 0.5934x^2 + 0.5970x^3$

Square Sum of Residuals (SSR): 0.78

$\sigma = 2$

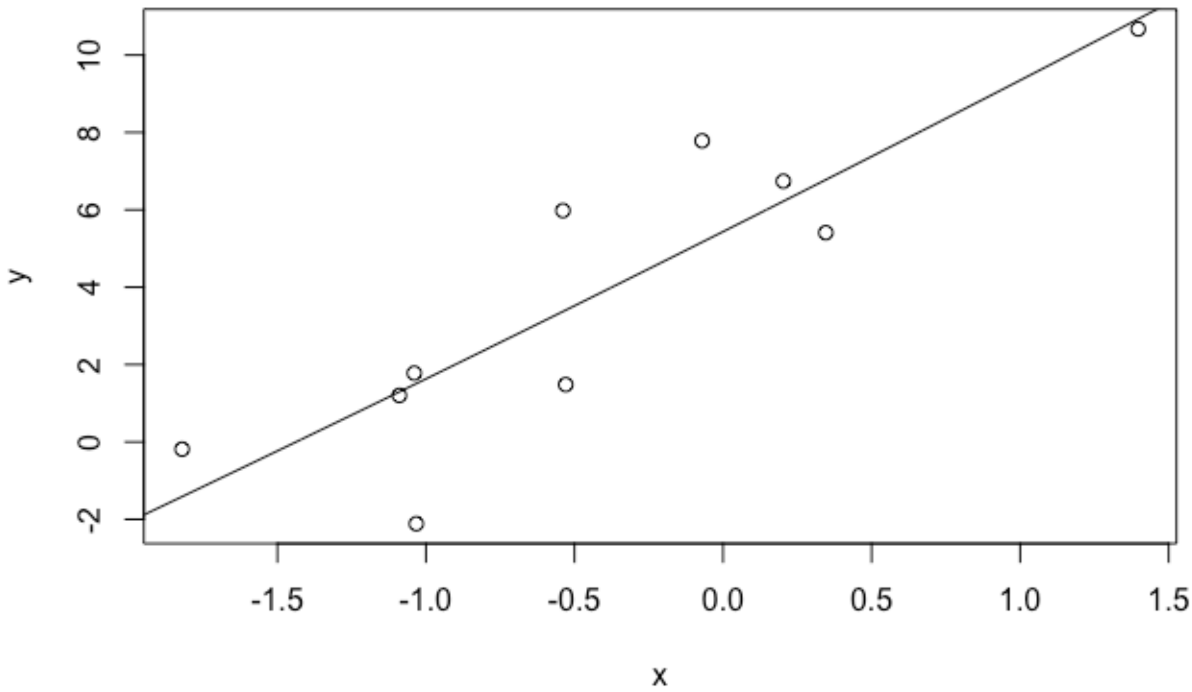
- 线性模型



$y = 5.476 + 3.837x$

Square Sum of Residuals (SSR):45.48

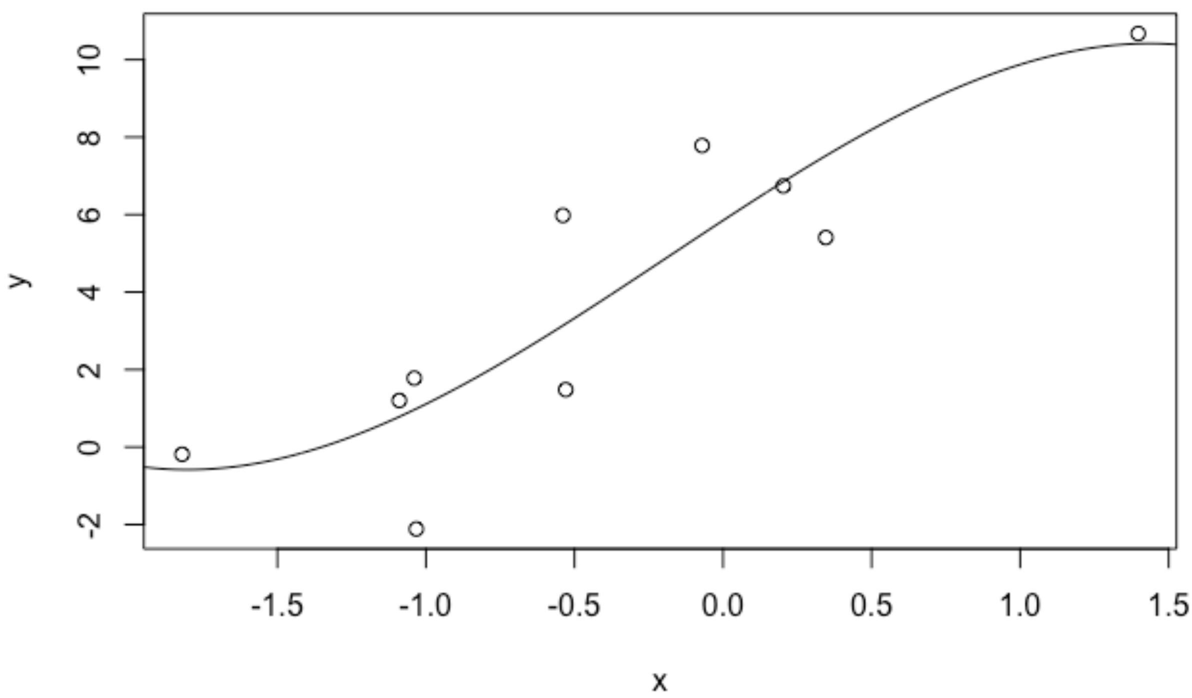
- 二次模型



$Y = 5.43592 + 3.85887x + 0.05326x^2$

Square Sum of Residuals (SSR):42.30

- 三次模型



$y = 5.8492 + 5.0285x + -0.3573x^2 + -0.6474x^3$

Square Sum of Residuals (SSR): 28.43

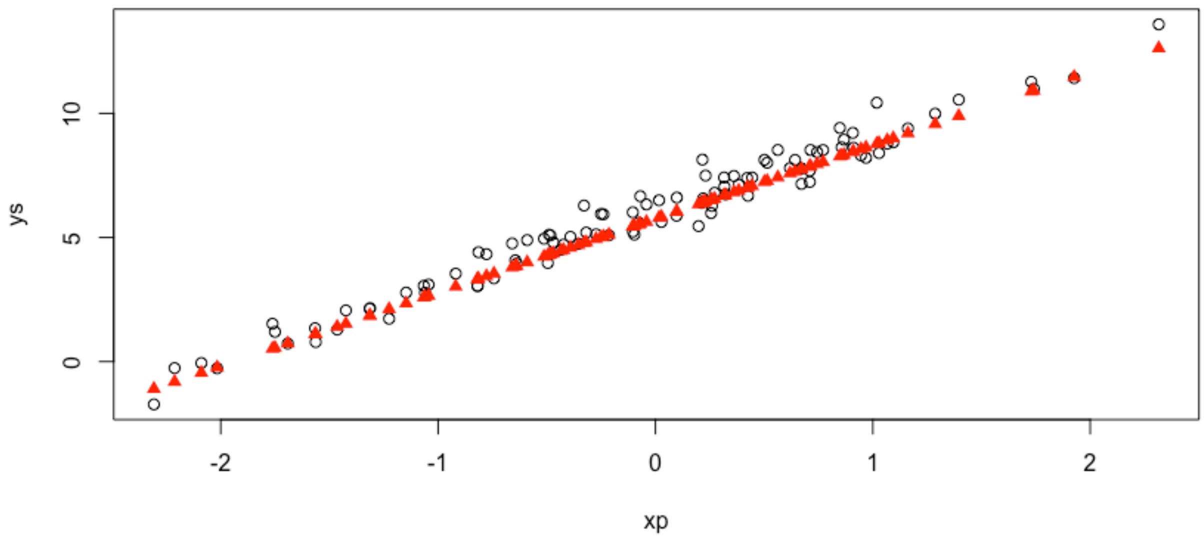
> 可以看出，随着模型参数的增加，拟合程度变好，残差平方和减小

(2)

以下黑色圆点为实际值，红色三角点为预测值

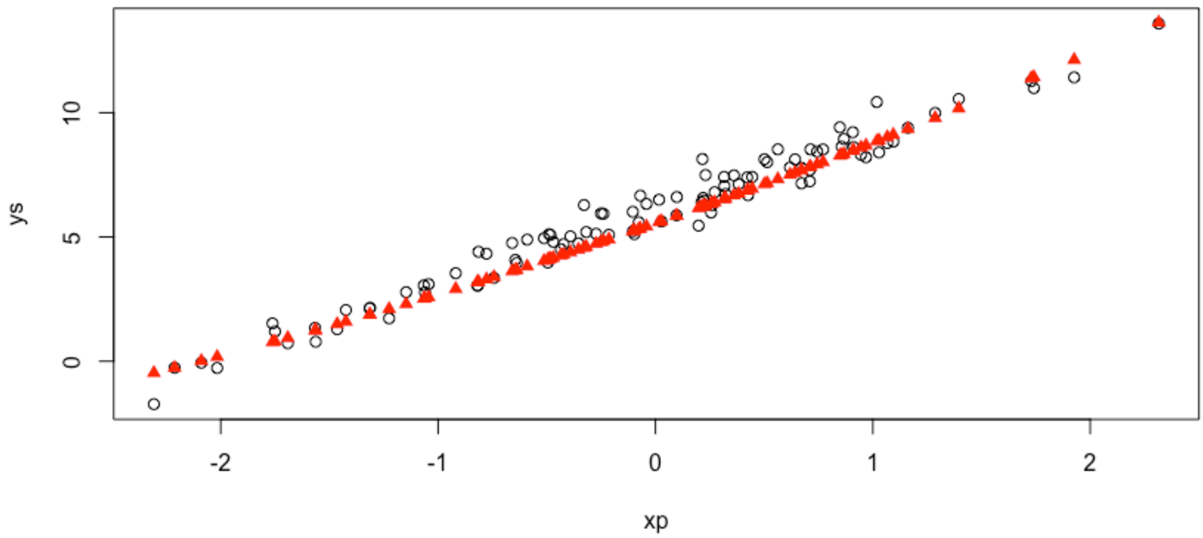
$\sigma = 0.5$

- 线性模型



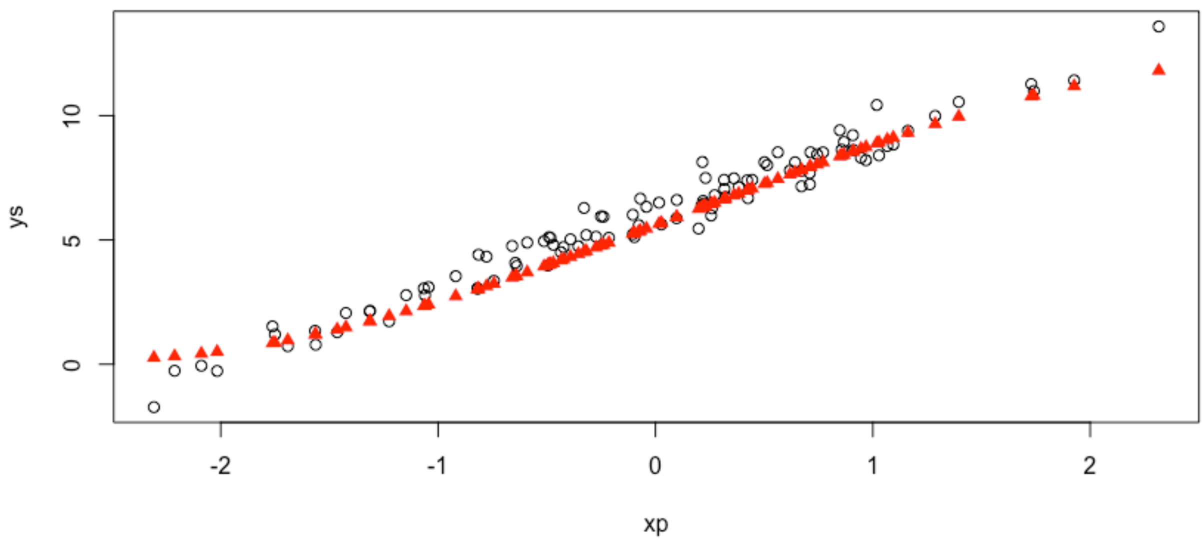
预测误差平方和：36.05

- 二次模型



预测误差平方和：46.52

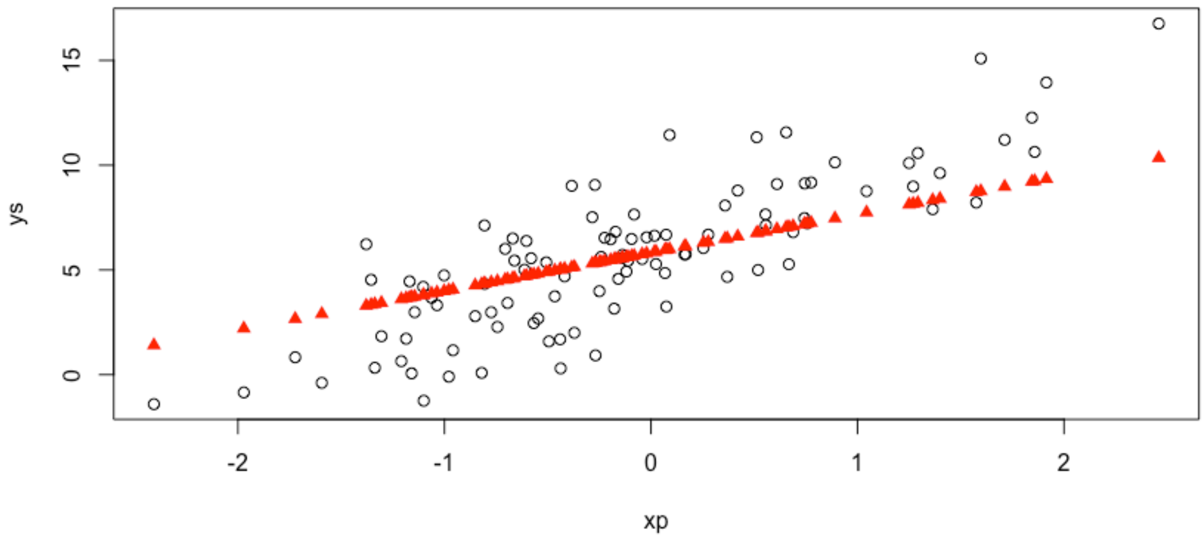
- 三次模型



预测误差平方和：56.85

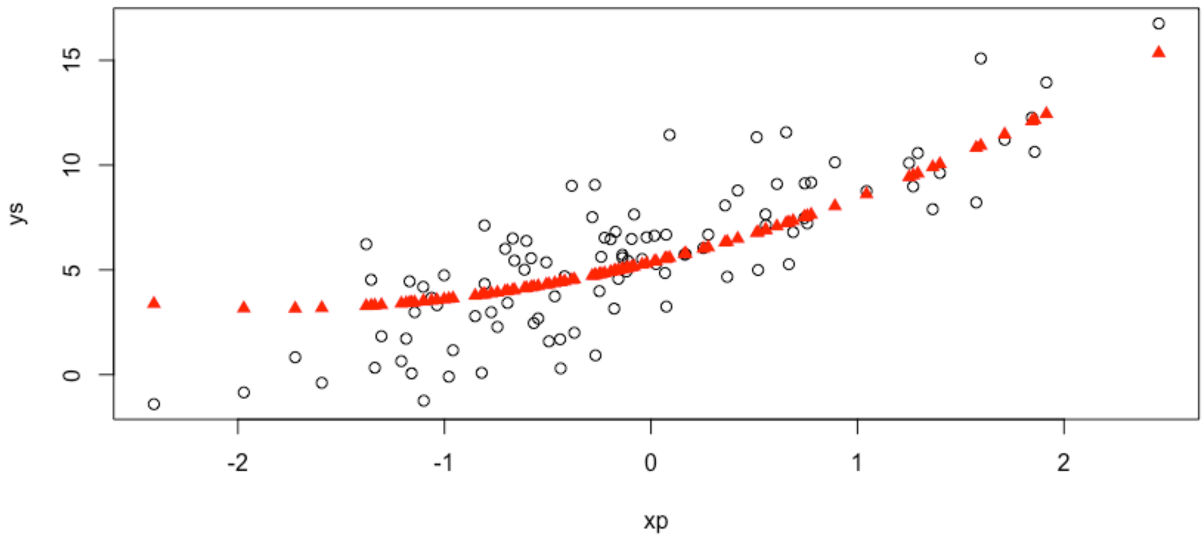
$\sigma = 2$

- 线性模型



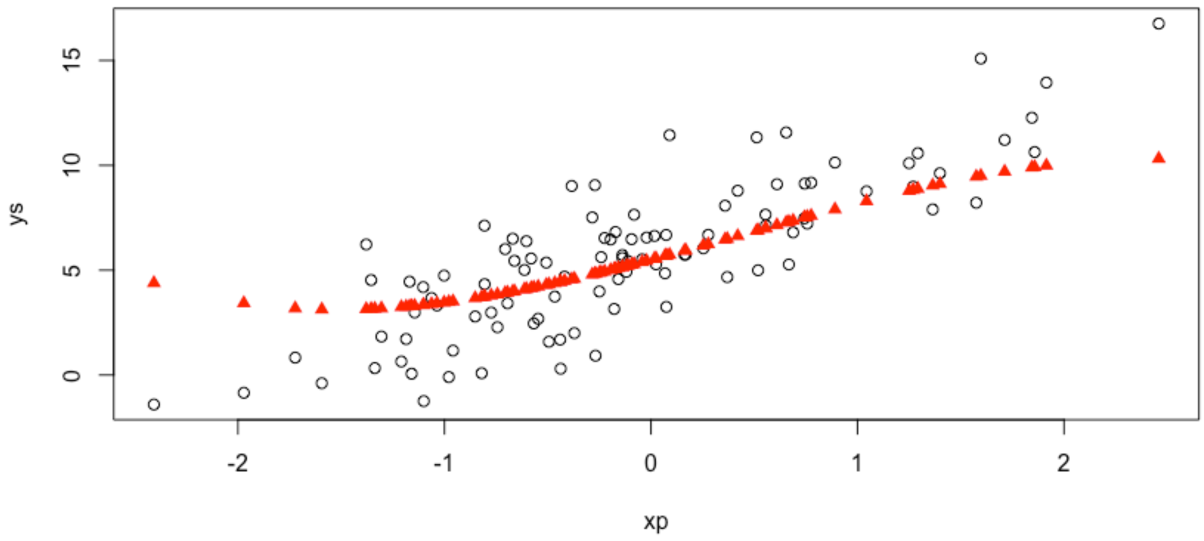
预测误差平方和：566

- 二次模型



预测误差平方和：478

- 三次模型



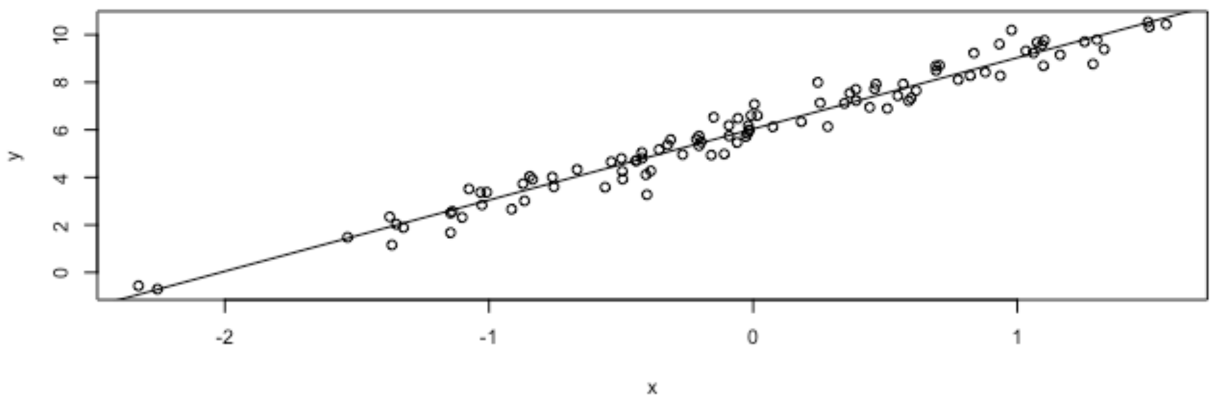
预测误差平方和：549

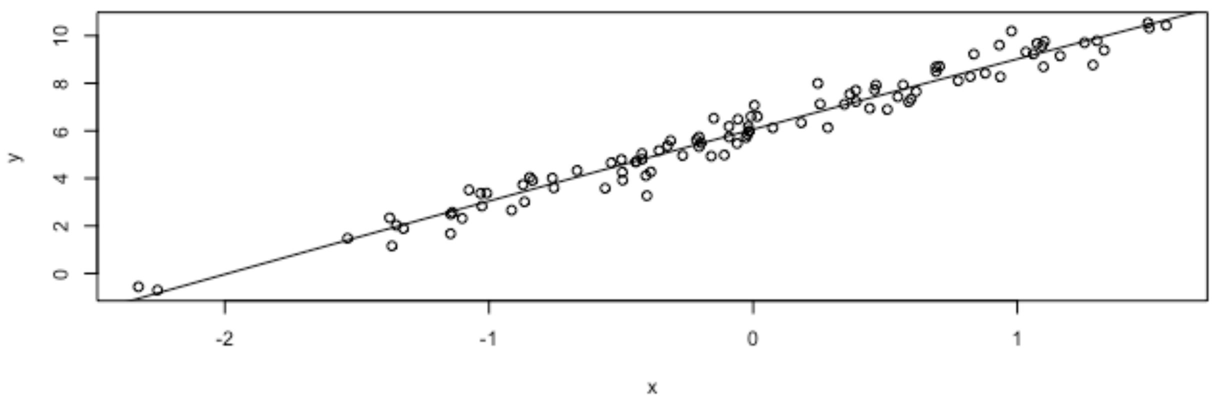
> 可以看出，尽管在训练样本中表现良好，但是在测试样本中，随着模型参数的增加，预测误差不仅没有减小，反而可能增加。

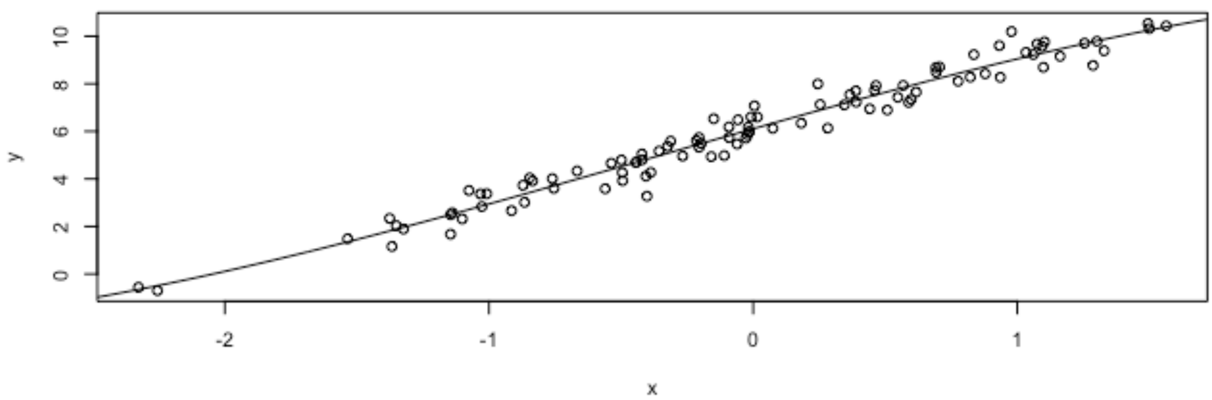
(3)

$\sigma = 0.5$

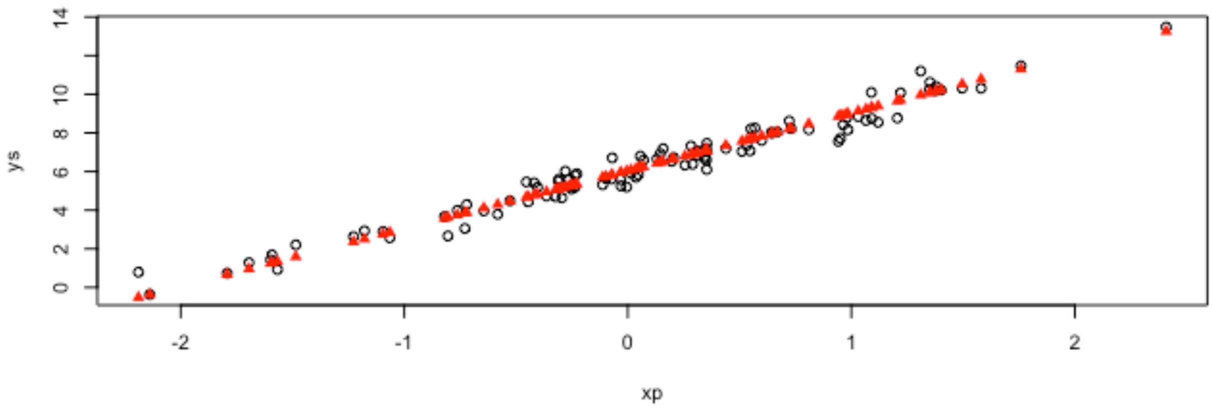
散点图和拟合曲线

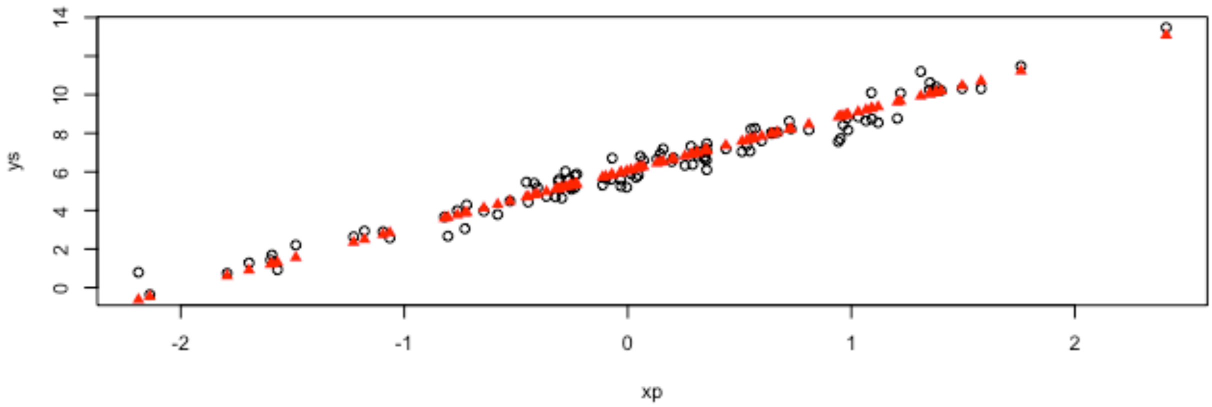


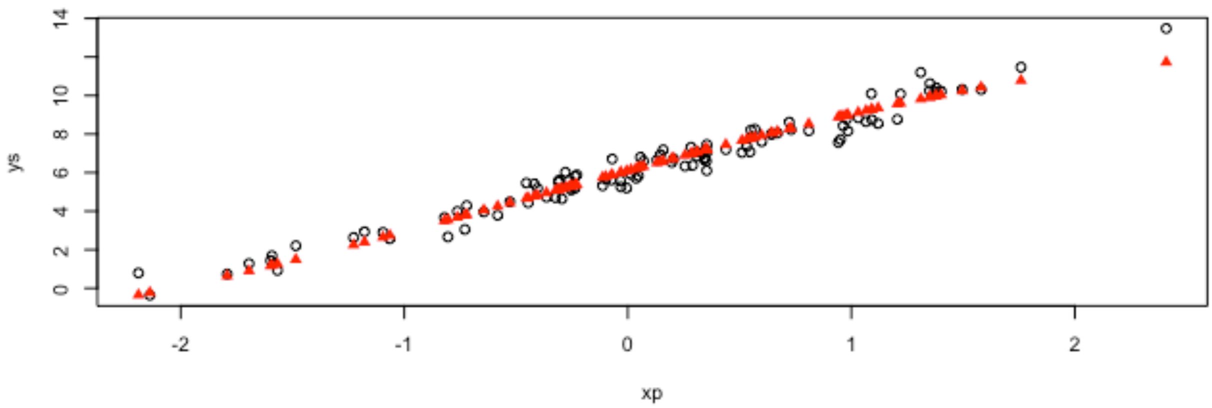




预测值与实际值

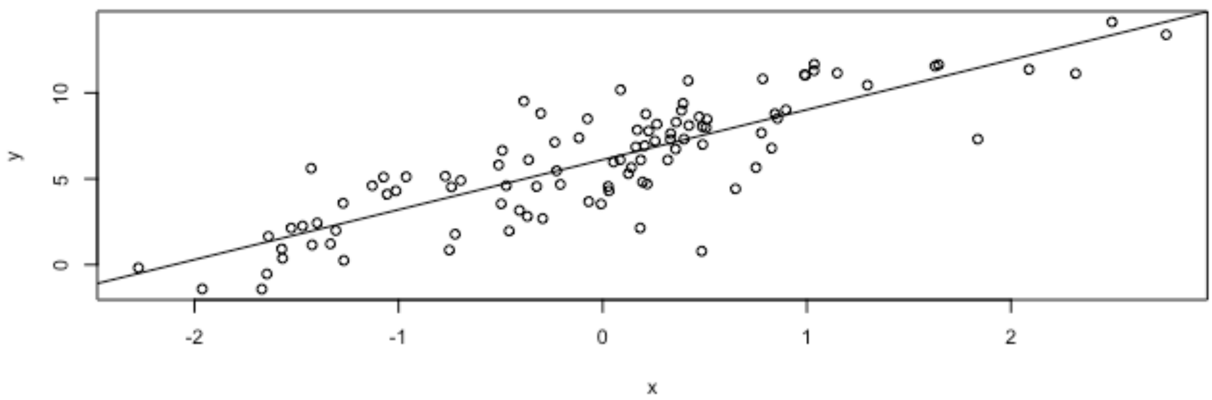


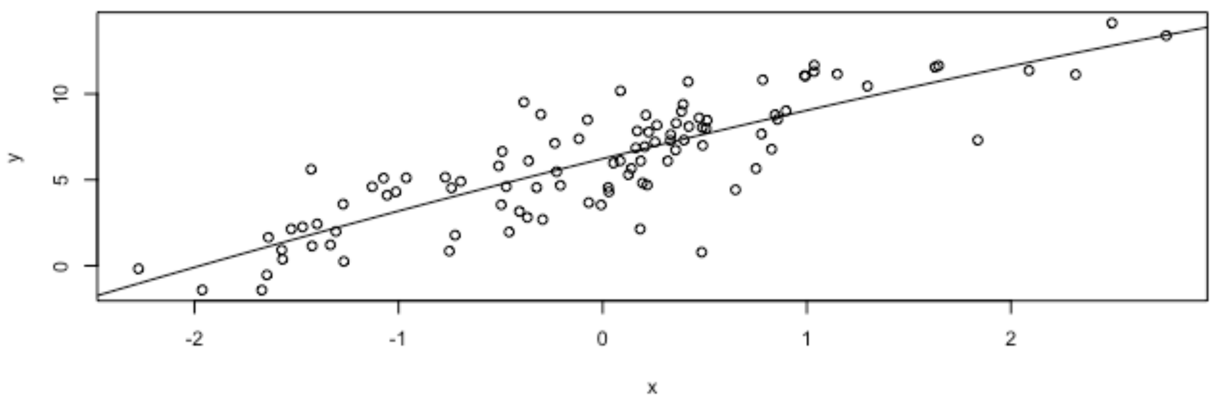


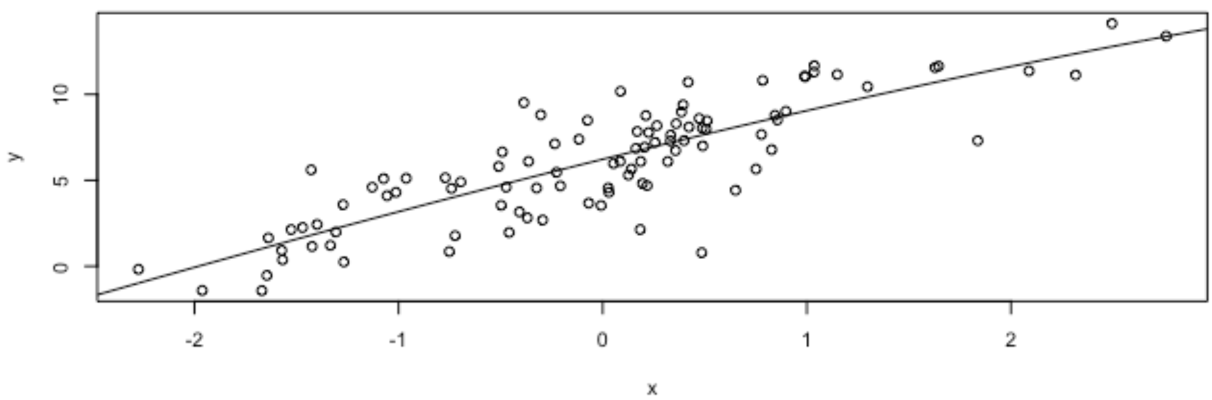


$\sigma = 2$

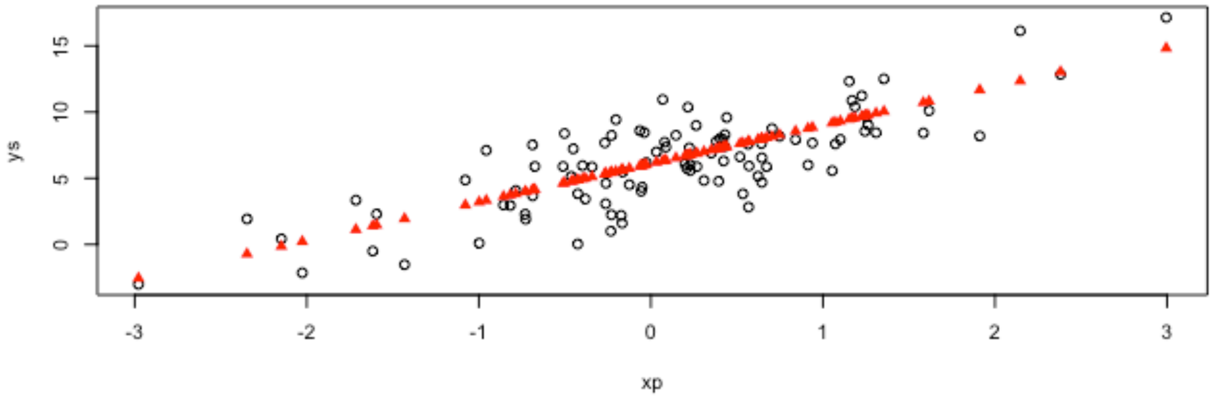
散点图和拟合曲线

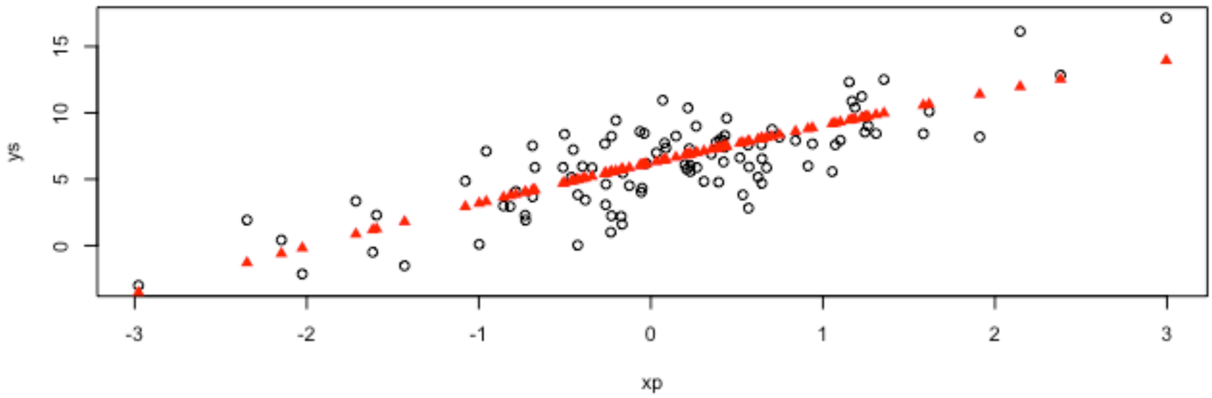


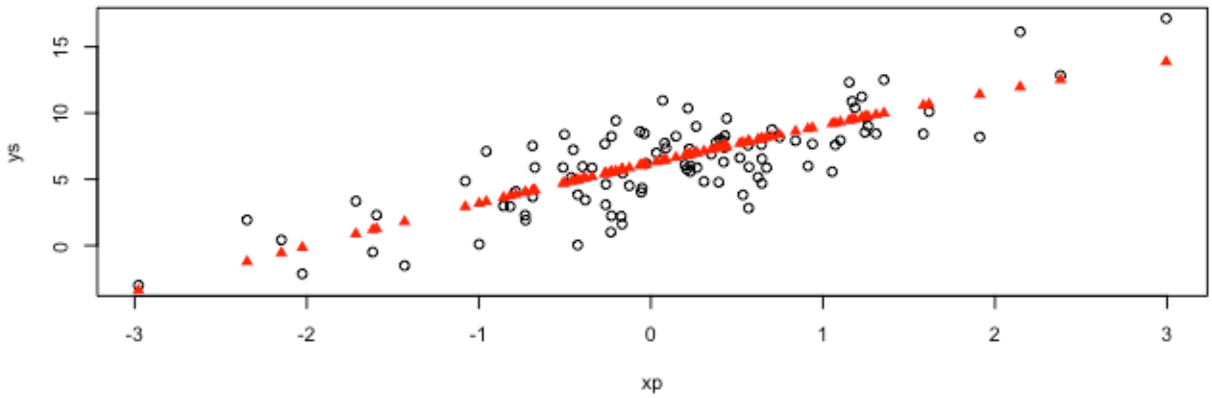




预测值与实际值







(4)

经过多次实验可以看出

- 在**样本量较小**时，**增加参数个数（模型复杂程度）**可以**显著提高在训练集上拟合效果**（减小残差），但是对于**减少预测误差**并无作用。
- 在不改变模型的前提下，**增加样本量**有助于**对模型参数更加精确的估计**（大数定律），具体体现为拟合时高次项系数趋近于0。
- 随着**噪声**的增加，在训练集上的拟合效果变差，同时预测的误差也增加。

4. 前列腺特异抗原水平预测

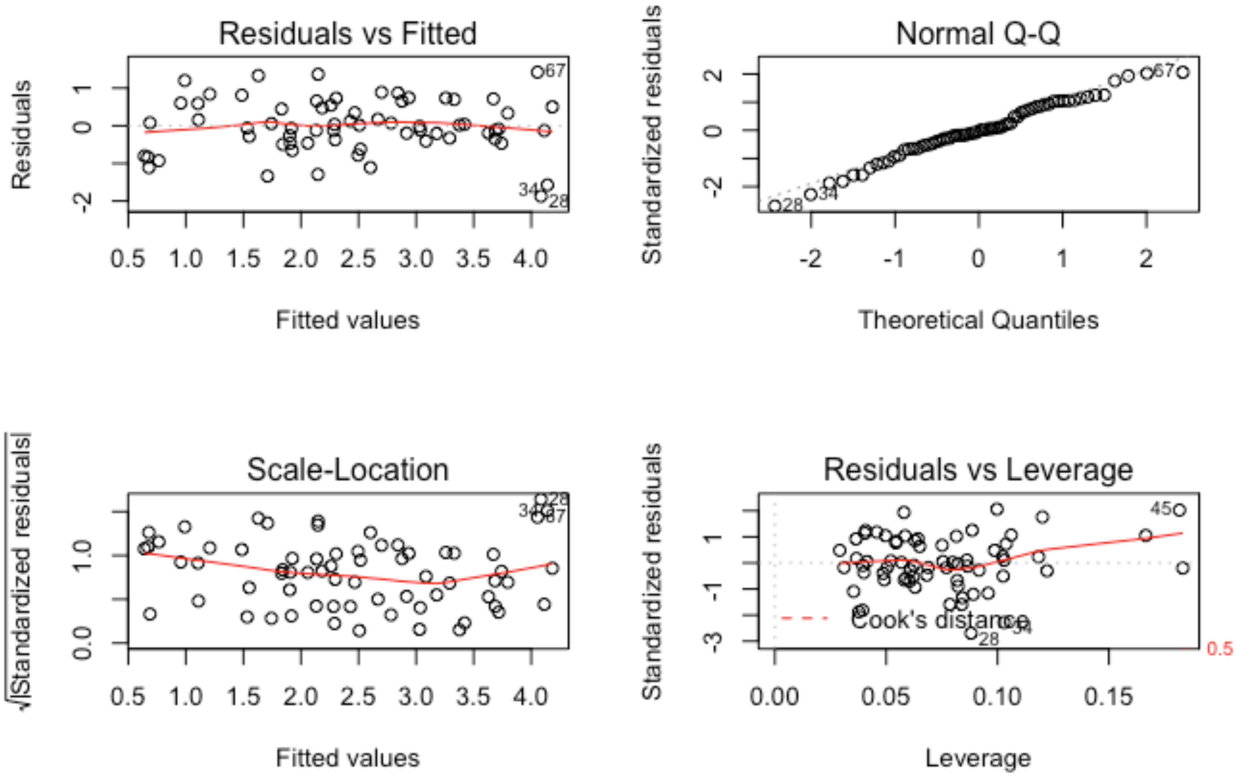
> 附件提供了一些前列腺癌患者临床指标的数据。请使用前四个临床数据（即lcavol，lweight， lbph，svi）对前列腺特异抗原水平（lpsa）进行预测。在给出的prostate_train.txt文件和 prostate_test.txt文件中，前4列每一列代表一个临床数据（即特征），最后一列是测量的前列腺特异抗 原水平（即预测目标的真实值）；每一行代表一个样本。
>
> 1. 在不考虑交叉项的情况下，利用Linear Regression对prostate_train.txt的数据进行回归，给出回归结果，并对prostate_test.txt文件中的患者进行预测，给出结果评价。
> 2. 如果考虑交叉项，是否会有更好的预测结果？请给出你的理由。

(1)

简单回归结果：$lpsa = -0.3259 + 0.5055 lcavol + 0.5388 lweight + 0.1400 lbph + 0.6718 svi$

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.32592    0.77998  -0.418   0.6775
lcavol       0.50552    0.09256   5.461 8.85e-07 ***
lweight      0.53883    0.22071   2.441   0.0175 *
lbph         0.14001    0.07041   1.988   0.0512 .
svi          0.67185    0.27323   2.459   0.0167 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7275 on 62 degrees of freedom
Multiple R-squared:  0.6592,    Adjusted R-squared:  0.6372
F-statistic: 29.98 on 4 and 62 DF,  p-value: 6.911e-14
```



在测试样本上的$R^2=0.54$ 结果仍需要改进

(2)

原模型：Multiple R-squared: 0.6592,	Adjusted R-squared: 0.6372

先只考虑二次交叉项

- 引入$lcavol*lweight$ 对应的P值为0.11514 不显著 Adjusted R-squared: 0.6461 略有增加
- 引入$lcavol*lbph$ 对应的P值为0.5005 不显著 Adjusted R-squared: 0.634 减少
- 引入$lcavol*svi$ 对应的P值为0.4687 不显著 Adjusted R-squared: 0.6344 减少
- 引入$lweight*lbph$ 对应的P值为0.0540 不显著 Adjusted R-squared: 0.6532 增加
- 引入$lweight*svi$ 对应的P值为0.16409 不显著 Adjusted R-squared: 0.6429 略有增加
- 引入$lbph*svi$ 对应的P值为0.2046 不显著 Adjusted R-squared: 0.6409 略有增加

三次及以上交叉项类似

综上，可以考虑引入$lweight*lbph$，但对模型无明显提升。

源代码

1.3.1

```r
x<-rnorm(10,0,1);# X

x2=x^2;
x3=x^3;
sigma=2;#σ=0.5,2
e=rnorm(10,0,sigma);#ϵ

y=3*x+6+e;


# 用于绘图
xs<- seq(min(x)-1,max(x)+1,length.out = 1000)
xs2=xs^2;
xs3=xs^3;


# 线性
model<- lm (y~x);
plot(x,y);
abline(model);

res<-model$residuals;
rss1=sum(res^2);

# 二次
model2<- lm(y~x+x2)
ys<-predict(model2,data.frame(x=xs,x2=xs2))
plot(x,y);
lines(xs,ys);

res<-model2$residuals;
rss2=sum(res^2);

# 三次
model3<- lm(y~x+x2+x3)
ys<-predict(model3,data.frame(x=xs,x2=xs2,x3=xs3))
```

```
plot(x,y);
lines(xs,ys);

res<-model3$residuals;
rss3=sum(res^2);
```

1.3.2

```r
x<-rnorm(10,0,1);# X

x2=x^2;
x3=x^3;
sigma=2;#σ=0.5,2
e=rnorm(10,0,sigma);#ϵ

y=3*x+6+e;

# 用于预测
xp<- rnorm(100,0,1);
e=rnorm(100,0,sigma);#ϵ
xp2=xp^2;
xp3=xp^3;
ys=3*xp+6+e;

# 线性
model<- lm (y~x);
yp<-predict(model,data.frame(x=xp));

plot(xp,ys);
points(xp,yp,pch=17,col="red")
sum1=sum((ys-yp)^2)

# 二次
model2<- lm(y~x+x2)
yp<-predict(model2,data.frame(x=xp,x2=xp2))
plot(xp,ys)
points(xp,yp,pch=17,col="red")
sum2=sum((ys-yp)^2)

# 三次
model3<- lm(y~x+x2+x3)
yp<-predict(model3,data.frame(x=xp,x2=xp2,x3=xp3))
plot(xp,ys)
points(xp,yp,pch=17,col="red")
sum3=sum((ys-yp)^2)
```

1.3.3

```r
x<-rnorm(100,0,1);# X
x2=x^2;
x3=x^3;
sigma=2;#σ=0.5,2
e=rnorm(100,0,sigma);#ϵ
y=3*x+6+e;
par(mfrow=c(3,1))

#用于绘图
xs<- seq(min(x)-1,max(x)+1,length.out = 1000)
xs2=xs^2;
xs3=xs^3;

# 线性
model<- lm (y~x);
plot(x,y);
abline(model);

res<-model$residuals;
rss1=sum(res^2);

# 二次
model2<- lm(y~x+x2)
ys<-predict(model2,data.frame(x=xs,x2=xs2))
plot(x,y);
lines(xs,ys);

res<-model2$residuals;
rss2=sum(res^2);

# 三次
model3<- lm(y~x+x2+x3)
ys<-predict(model3,data.frame(x=xs,x2=xs2,x3=xs3))
plot(x,y);
lines(xs,ys);

res<-model3$residuals;
rss3=sum(res^2);
```

```r
rss3=sum(res^2);


# 用于预测
xp<- rnorm(100,0,1);
e=rnorm(100,0,sigma);#ϵ
xp2=xp^2;
xp3=xp^3;
ys=3*xp+6+e;

# 线性
model<- lm (y~x);
yp<-predict(model,data.frame(x=xp));

plot(xp,ys);
points(xp,yp,pch=17,col="red")
sum1=sum((ys-yp)^2)

# 二次
model2<- lm(y~x+x2)
yp<-predict(model2,data.frame(x=xp,x2=xp2))
plot(xp,ys)
points(xp,yp,pch=17,col="red")
sum2=sum((ys-yp)^2)

# 三次
model3<- lm(y~x+x2+x3)
yp<-predict(model3,data.frame(x=xp,x2=xp2,x3=xp3))
plot(xp,ys)
points(xp,yp,pch=17,col="red")
sum3=sum((ys-yp)^2)
```

1.4

```r
# 读入数据
testSet<-read.table("prostate_test.txt",head=TRUE);
trainSet<-read.table("prostate_train.txt",head=TRUE);

# 回归
model<-lm(lpsa~lcavol+lweight+lbph+svi,data = trainSet);
predicted<-predict.lm(model,newdata=testSet);

# 模型评价


par(mfrow=c(2,2))
plot(model)

summary(model)

# 交叉项
model1<-lm(lpsa~lcavol+lweight+lbph+lbph:lweight+svi,data = trainSet
summary(model1)
```