第8章　非線形回帰分析

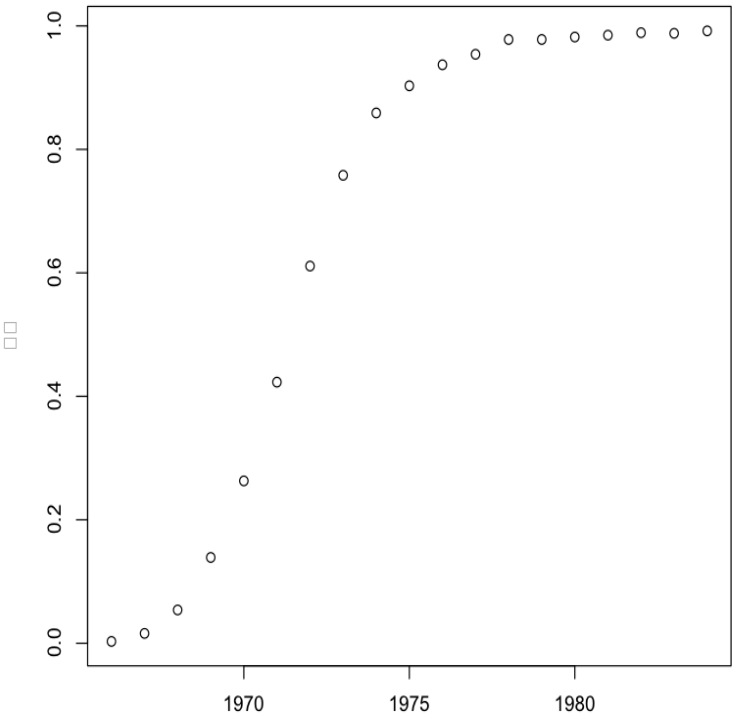
- ロジスティック回帰
- 多項式回帰
- 一般化線型モデル
- 平滑化回帰

1. ロジスティック回帰

普及率、成長率のデータは非線形曲線が多い。
通常このような曲線は、ロジスティック関数に当てはまる。

ロジスティック関数　　$y = a/1+be^{cx}$

```
> 年度<-c(1966:1984)
> 普及<-
c(0.003,0.016,0.054,0.139,0.263,0.423,0.611,0.758,0.859,0.903,0.937,0.954,0.9
78,0.978,0.982,0.985,0.989,0.988,0.992)
```



カラーテレビの普及率

2. 非線形モデルと関数nls

nls(formula,data,start,trace)

```
> fm<-nls(普及率~a/(1+b*exp(c*1:19)),start=c(a=1,b=1,c=-1),trace=TRUE)
```

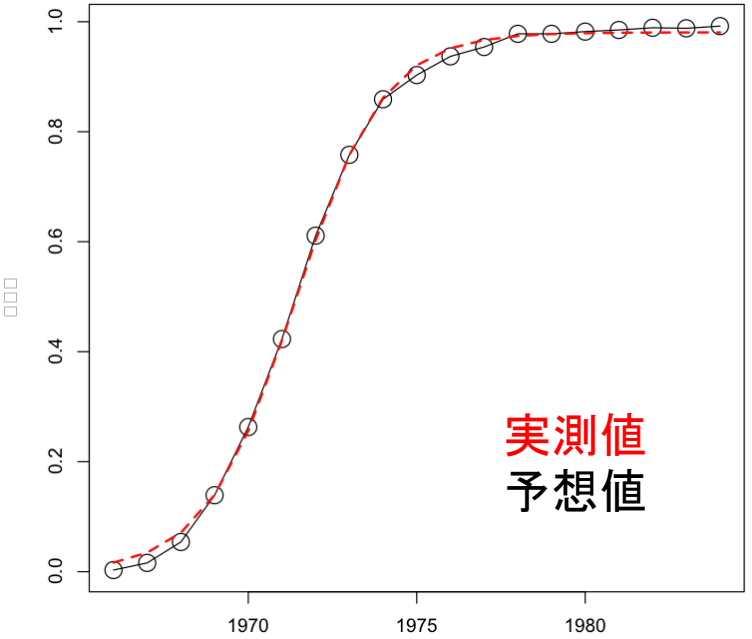
上記の失敗は、exp(年度)が大き過ぎるのが原因。
説明変数を1966～1984を１～19に置き換える。

```
> fm<-nls(普及率~a/(1+b*exp(c*1:19)),start=c(a=1,b=1,c=-1),trace=TRUE)
3.905671 :    1  1 -1
2.387674 :    0.9824052  0.4300442 -0.1029666
1.743185 :    0.8872618  0.8264732 -0.2623701
0.7740848 :    0.9841109  2.3123040 -0.2310466
0.5578214 :    0.9271411  7.5149745 -0.5270324
0.09229081 :    1.0001338 17.1134431 -0.4350728
0.06874582 :    0.9606817 40.3886048 -0.6606493
0.01653944 :    0.9826601 75.9510196 -0.7160221
0.003486704 :     0.9806949 110.6878618   -0.7509771
0.001959816 :     0.9804580 123.8500779   -0.7565368
0.00194977 :     0.9806034 123.8048223   -0.7553703
0.001949752 :     0.9806268 123.6621028   -0.7551686
0.001949752 :     0.9806279 123.6609455   -0.7551647

> summary(fm)

Formula: 普及率 ~ a/(1 + b * exp(c * 1:19))

Parameters:
   Estimate Std. Error t value Pr(>|t|)
a   0.98063    0.00384 255.401  < 2e-16 ***
b 123.66094   13.56739   9.115 9.82e-08 ***
c  -0.75516    0.01742 -43.347  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01104 on 16 degrees of freedom

Number of iterations to convergence: 12
Achieved convergence tolerance: 6.987e-06
```

2.非線形モデルと関数nls

```
> plot(年度,普及率,cex=2)
> lines(年度,普及率)
> lines(年度,fitted(fm),col=2,lty=2,lwd=2)
> legend(locator(1),c("実測値","予測値"),col=1:2,lty=1:2,lwd=2)
```

実測値と予測値は非常によく
近似している。



テレビの普及率の実測値と予測値

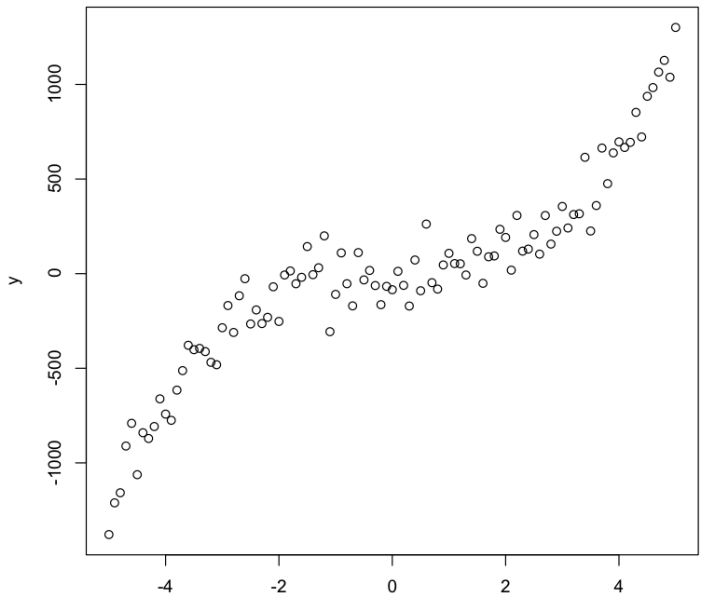
3. 多項式回帰

多項式の人口データを作る

```
> set.seed(30);x<-seq(-5,5,0.1)
> y<-10*x^3+100*rnorm(x,0,1)
> plot(x,y)
```

解析と結果

3次多項式の一般式
$y = a+bx+cx^2+dx^3$
この式を用いた関数nlsの例。



3次多項式の人工データ

3. 多項式回帰

```
> fm3<-nls(y~a+b*x+c*x^2+d*x^3,start=c(a=1,b=1,c=1,d=1),trace=T)
21031980 :  1 1 1 1
1073246 :   1.3077234 13.8639457 -0.9720568  9.4123383
>
> summary(fm3)

Formula: y ~ a + b * x + c * x^2 + d * x^3

Parameters:
  Estimate Std. Error t value Pr(>|t|)
a   1.3077    15.7011   0.083    0.934
b  13.8639     8.9781   1.544    0.126
c  -0.9721     1.3769  -0.706    0.482
d   9.4123     0.5379  17.498   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 105.2 on 97 degrees of freedom

Number of iterations to convergence: 1
Achieved convergence tolerance: 9.23e-08
```

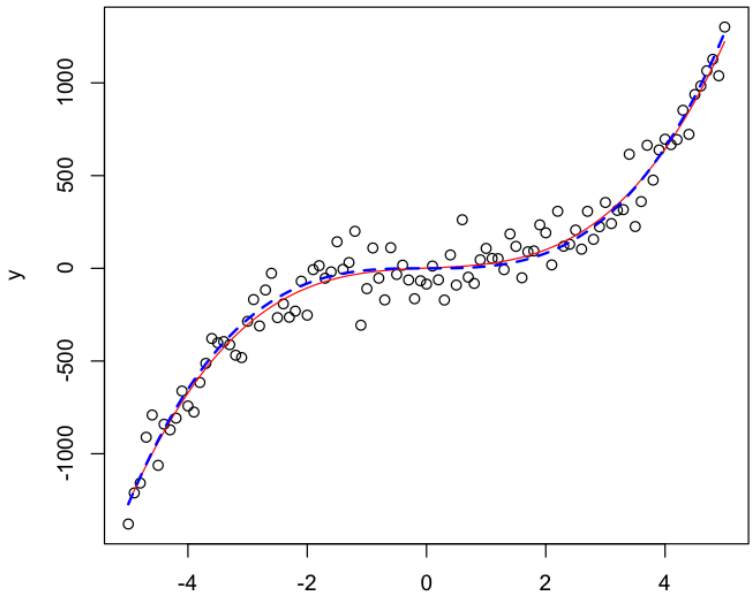
係数a、cのPr(>|t|)は若干大きい。そこで、簡潔な3次式$y=bx^3$を用いて同じデータについて非線形回帰分析を行う。

```
> fm4<-nls(y~d*x^3,start=c(d=1),trace=T)
21236247 :  1
1110029 :  10.1737
> summary(fm4)

Formula: y ~ d * x^3

Parameters:
  Estimate Std. Error t value Pr(>|t|)
d  10.1737     0.2154   47.22   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 105.4 on 100 degrees of freedom

Number of iterations to convergence: 1
Achieved convergence tolerance: 4.411e-10
```

```
> AIC(fm3)
[1] 1233.004
> AIC(fm4)
[1] 1230.408
```

AIC値に大きな差はないが、後者が若干小さい。

```
> lines(x,fitted(fm3),lty=1,col=2)
> lines(x,fitted(fm4),lty=2,col=4,lwd=2)
> legend(locator(1),c("一般式","簡潔式"),lty=1:2,col=c(2,4),lwd=2,cex=1.5)
```



人工データの実値と両モデルの予測値

4. 一般化線形モデル

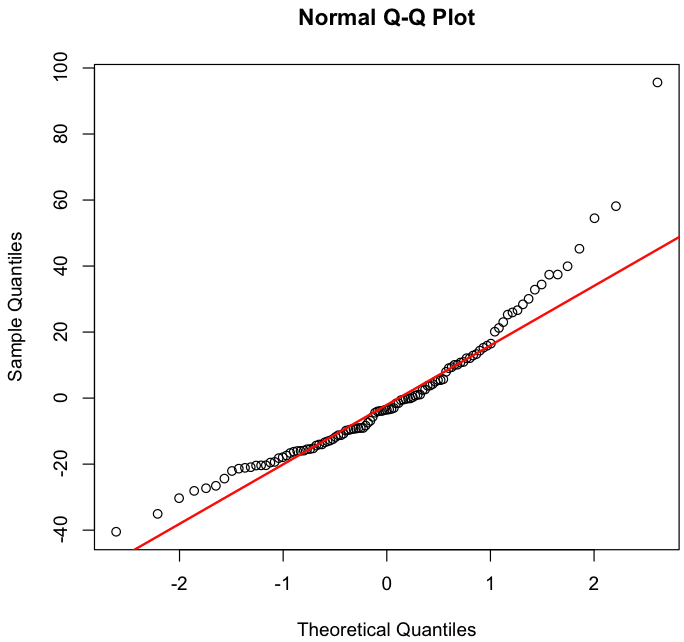
glm(formula, family, data)

ケーススタディ

一般化線形モデル関数glmの使用法について、本項では
データセットairqualityを用いる。

```
> air.lm<-lm(Ozone~Solar.R+Wind+Temp,data=airquality)
```

```
> qqnorm(resid(air.lm))
> qqline(resid(air.lm),lwd=2,col="red")
```



残差のQ-Qプロットを図示する。
点は直線より乖離しているので、残差
は正規分布とは言い難い。

4. 一般化線形モデル

そこで、正規分布とガンマ分布を仮定した両モデルの
当てはめの良さを比較してみる。

```
> air.glm1<- glm(Ozone~Solar.R+Wind+Temp,data= airquality,family=gaussian)
> air.glm2<- glm(Ozone~Solar.R+Wind+Temp,data= airquality,family= Gamma)
```

正規分布
ガンマ分布

関数AICを用いて、モデルの当てはめの良さを評価する。

```
> AIC(air.lm)
[1] 998.7171
> AIC(air.glm1)
[1] 998.7171
> AIC(air.glm2)
[1] 939.8778
```

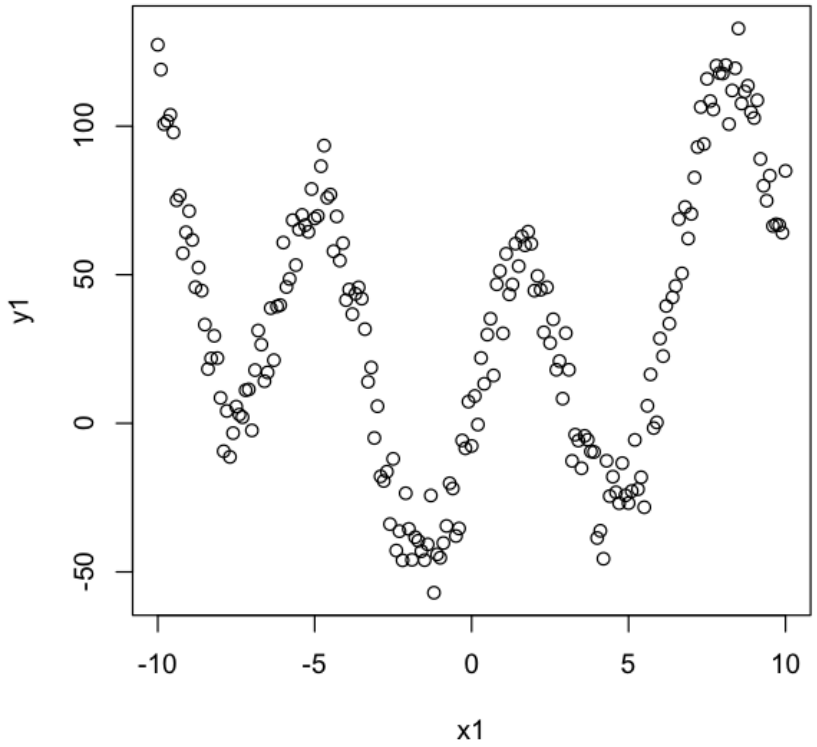
AIC値が小さいので、ガンマ分布を用いた方が当てはめが良いと
判断できる。

5. 平滑化回帰モデル

関数lm,glm,nlsが対応できない複雑なデータの回帰モデルの説明のため、
次の人工データを用いる。

```
> x1=seq(-10,10,0.1);set.seed(10)
> y1=50*sin(x1)+x1^2+10*rnorm(length(x1
> plot(x1,y1)
```

散布図から考えて、多項式回帰
モデルが良さそう。

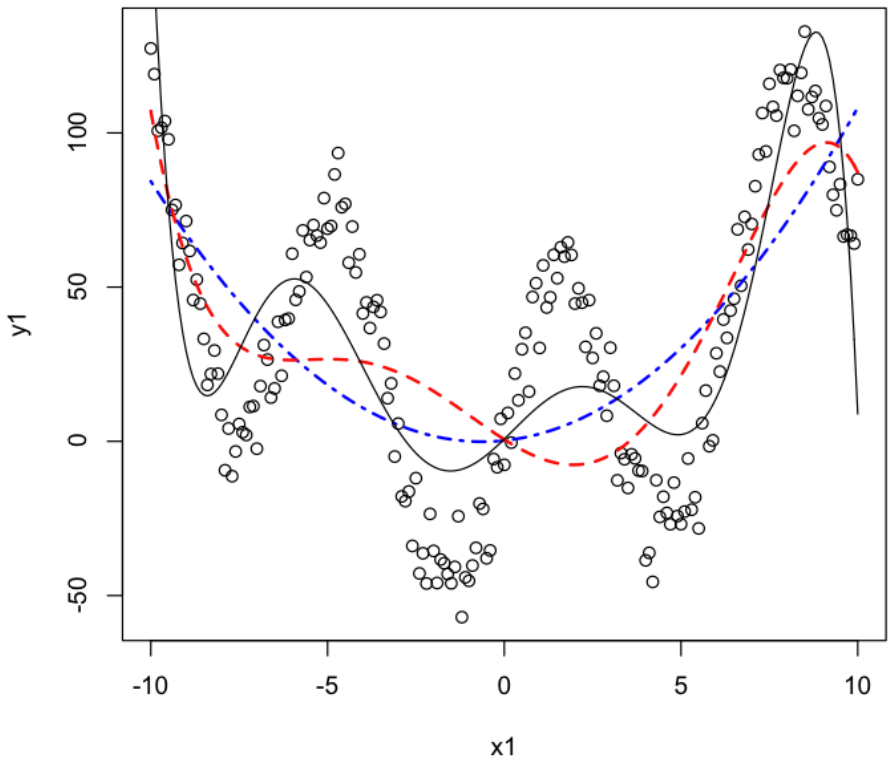


5. 平滑化回帰モデル

次の3種類の多項式回帰モデルと、その回帰モデルによる予測値を散布図に加える。

```
> lmp2<-fitted(lm(y1~poly(x1,2)))
> lines(x1,lmp2,lty=4,col=4,lwd=2)
> lmp5<-fitted(lm(y1~poly(x1,5)))
> lines(x1,lmp5,lty=2,col=2,lwd=2)
> lmp8<-fitted(lm(y1~poly(x1,8)))
> lines(x1,lmp8,)
```

多項式によるモデルの当てはめが良いとは言い難い。



5. 平滑化回帰モデル

このように複雑に変化するデータの回帰方法として、平滑化回帰がある。
Rに用意されている関数smooth,spline,ksmooth,supsmu,lowessを用いた平滑コマンドの例を次に示す。

```
> par(mfrow=c(2,2),oma=c(2,2,2,2),mar=c(2,2,2,
> plot(x1,y1,main="関数smooth.splineによる平滑結果
> lines(smooth.spline(x1,y1),col=2,lwd=2)
> plot(x1,y1,main="関数ksmoothによる平滑結果")
> lines(ksmooth(x1,y1),col=4,lwd=2)
> plot(x1,y1,main="関数supsmuによる平滑結果")
> lines(supsmu(x1,y1),col=1,lwd=2)
> plot(x1,y1,main="関数lowessによる平滑結果")
> lines(lowess(x1,y1),col=5,lwd=2)
> lines(lowess(x1,y1,f=0.1),col=5,lwd=2)
```