Review for Exam 1

Andriana Manousidaki

2023-10-16

Linear Regression

The body weights of the chickens were measured at birth and every second day thereafter until day 20. They were also measured on day 21. There were four groups on chicks on different protein diets,we will focus only on 2 groups.

```
path = 'https://vincentarelbundock.github.io/Rdatasets/csv/datasets/ChickWeight.csv'
chicken_weights <- read.csv(path, row.names = 1)
chicks=chicken_weights[chicken_weights$Diet%in%c(1,2),]
head(chicks)
```

```
##   weight Time Chick Diet
## 1     42    0     1    1
## 2     51    2     1    1
## 3     59    4     1    1
## 4     64    6     1    1
## 5     76    8     1    1
## 6     93   10     1    1
```

Fit a linear model for weight as a function of time and diet. Consider the possiblity that diet may also affect time.

```
fm = lm(weight ~ Time + Diet + Time*Diet, data = chicks)
summary(fm)
```

```
##
## Call:
## lm(formula = weight ~ Time + Diet + Time * Diet, data = chicks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -135.425  -16.780   -0.853   11.284  130.391
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  33.2284    10.9471   3.035  0.00259 **
## Time          5.0745     0.8700   5.833 1.28e-08 ***
## Diet         -2.2974     7.6938  -0.299  0.76543
## Time:Diet     1.7673     0.6052   2.920  0.00373 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 36.07 on 336 degrees of freedom
## Multiple R-squared:  0.6748, Adjusted R-squared:  0.6719
## F-statistic: 232.4 on 3 and 336 DF,  p-value: < 2.2e-16
```

Does diet affect weight?

```
HO = lm(weight ~ Time, data = chicks)
HA = fm

anova(HO, HA)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ Time
## Model 2: weight ~ Time + Diet + Time * Diet
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    338 469862
## 2    336 437076  2     32786 12.602 5.278e-06 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

Conclusion the variable diet is significant for the estimation of weight, since from the Ftest we have extremely strong evidence against Ho.

Provide CI for the coefficient of time

```
summary(fm)
```

```
##
## Call:
## lm(formula = weight ~ Time + Diet + Time * Diet, data = chicks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -135.425  -16.780   -0.853   11.284  130.391
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  33.2284    10.9471   3.035  0.00259 **
## Time          5.0745     0.8700   5.833 1.28e-08 ***
## Diet         -2.2974     7.6938  -0.299  0.76543
## Time:Diet     1.7673     0.6052   2.920  0.00373 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 36.07 on 336 degrees of freedom
## Multiple R-squared:  0.6748, Adjusted R-squared:  0.6719
## F-statistic: 232.4 on 3 and 336 DF,  p-value: < 2.2e-16
```

```r
estimate =  5.0745
se_time = 0.87
c(estimate -1.96*se_time, estimate +1.96*se_time)
```

```
## [1] 3.3693 6.7797
```

Splines

Continuing from the previous problem: A researcher wants to learn whether we have evidence of non-linear growth under Diet 1. Using the data for Diet 1 only, test for non-linearity using a cubic B-spline with 4 DF.

```r
library(splines)
DATA=chicks[chicks$Diet==1,]
H0=lm(weight~Time,data=DATA)
HA=lm(weight~bs(Time,intercept=FALSE,df=4,degree=3),data=DATA)
anova(H0,HA)
```

```
## Analysis of Variance Table
##
## Model 1: weight ~ Time
## Model 2: weight ~ bs(Time, intercept = FALSE, df = 4, degree = 3)
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    218 235212
## 2    215 229543  3    5668.4 1.7698 0.1539
```

Maximum Likelihood Estimation

The following data come from a Beta distribution. Find the mle of the two shape parameters of beta distribution, using optim().Information about the beta distribution can be find in https://en.wikipedia.org/wiki/Beta_distribution

The given data:

```r
y=rbeta(n=500, shape1=10, shape2=4)
# Since we know how those simulated data were created,
# 10 and 4 are true values for the shape1 parameter
```

We create a function that calculates the negative loglikelihood of a sample of Beta distributed values. Notice that we use dbeta to avoid complex calculations.

```r
negloglik_beta = function(y,shape){
  loglik_beta = sum(dbeta(y, shape1 = shape[1], shape2= shape[2], log = TRUE))
  return(-loglik_beta)
}

negloglik_beta(y, shape=c(10,4))
```

```
## [1] -363.2155
```

```
mle_opti= optim(par=c(5,5),y=y, fn = negloglik_beta, hessian =TRUE)
mle_opti
```

```
## $par
## [1] 9.423296 3.876451
##
## $value
## [1] -364.0352
##
## $counts
## function gradient
##       83       NA
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##           [,1]      [,2]
## [1,]  16.93128 -39.04345
## [2,] -39.04345 107.98975
```

The mle of shape1 is 9.905468 and for shape2 is 3.970956. Those estimatros are actually pretty close to the true ones(10, 4).

Construct a 95% CI for the two shape parameters

Extract the variance-covariance matrix from the hessian of the negative loglikelihood calculated on the mle values.

```
VAR = solve(mle_opti$hessian)
SE = sqrt(diag(VAR))
SE
```

```
## [1] 0.5959977 0.2359925
```

```
CI =data.frame(lower= mle_opti$par-1.96*SE, upper = mle_opti$par+1.96*SE)
CI
```

```
##      lower     upper
## 1 8.255140 10.591451
## 2 3.413906  4.338997
```

Test if shape1 isn't equal with the mle estimator but it is actually eaqually with 5

HO: shape1 = 5, HA: shape1 != 5.

```
lo = -negloglik_beta(y=y, shape = c(5, mle_opti$par[2]))
la = -negloglik_beta(y=y, shape = c(mle_opti$par[1], mle_opti$par[2]))

lrt = -2*lo +2*la

lrt
```

```
## [1] 469.3105
```

```
p_val = pchisq(lrt, df=1, lower.tail = FALSE)
p_val
```

```
## [1] 4.527084e-104
```

Other topics: Logistic regression, dimension reduction with svd.