THE AUSTRALIAN NATIONAL UNIVERSITY

RESEARCH SCHOOL OF FINANCE,
ACTUARIAL STUDIES AND STATISTICS

STAT3008/STAT7001
APPLIED STATISTICS

Assignment 2 Solution

Lecturer: Dr Tao Zou

Last Updated: Mon Oct 16 23:39:55 2017

This assignment is due at 12:00 pm, Oct 18, 2017.

This assignment is worth 10% of your final grade but is optional and redeemable. Students are expected to complete this assignment **individually**. Maximum points: 10.0. You cannot get partially correct for all the questions, since each question is only worth 0.5 points. **Assignments can only be submitted via the physical assignment box at the front of the reception on Level 4, CBE Building (26C). Hard copy submission is required.** Late submission will not be accepted and the weight will roll over to your final exam. Identical submissions are treated as cheating.

Please **exactly follow the instructions of questions** and write down the answers of the following questions in the **answer sheet** file on the Wattle. Note that you do not need to copy the questions in the answer sheet. Please only submit your finished answer sheet and do not paste any unrelated results. The data used in this assignment are on the Wattle or in the R package "Sleuth3", whose instruction manual is on the Wattle.

The significance level for all the questions is set to be 0.05.

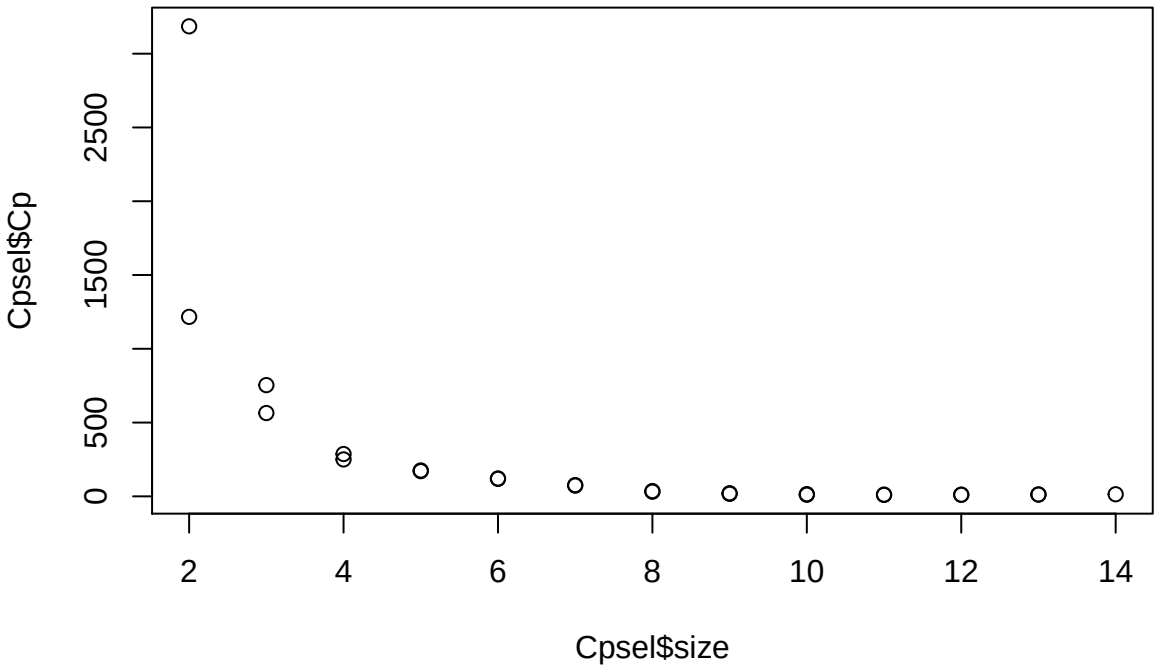
Question 1 (Variable Selection and Multicollinearity, 2.5 points)

Consider the data used in Quesitons 1 - 3 in Assignment 1. Please answer the following questions in the answer sheet.

(Some indicator variables selected can be different in this question. But the result should be the same.)

a) (0.5 points) If we only consider the response variable and explanatory variables used in Quesiton 3 e) in Assignment 1 (please use indicator variables instead of the original categorical variables), please paste the Cp plot among all subsets in the answer sheet (showing at most 2 subsets for each size).

Solution:



b) (0.5 points) Based on the above Cp statistics, which variables should we choose to predict the logarithm of "WeeklyEarnings" by using the variable selection among all subsets?

Solution:

```
   1 2 3 4 5 6 7 8 9 A B C D
1  0 0 0 1 0 0 0 0 0 0 0 0 0  2 1216.326632
1  0 0 0 0 0 0 0 0 0 0 0 0 1  2 3185.268384
2  0 0 0 1 0 0 0 0 0 0 0 0 1  3  564.570735
2  1 0 0 1 0 0 0 0 0 0 0 0 0  3  753.711695
3  1 0 0 1 0 0 0 0 0 0 0 0 1  4  250.153980
3  1 1 0 1 0 0 0 0 0 0 0 0 0  4  286.775212
4  1 1 1 1 0 0 0 0 0 0 0 0 0  5  170.622636
4  1 1 0 1 0 0 0 0 0 0 0 0 1  5  175.376397
5  1 1 1 1 0 0 0 0 1 0 0 0 0  6  118.252965
5  1 1 1 1 0 0 0 0 0 1 0 0 0  6  121.223179
6  1 1 1 1 0 0 0 0 1 1 0 0 0  7   72.634019
6  1 1 1 1 0 0 0 1 0 1 0 0 0  7   76.473310
7  1 1 1 1 0 1 0 0 1 1 0 0 0  8   31.262271
7  1 1 1 1 0 1 0 1 0 1 0 0 0  8   35.771192
8  1 1 1 1 0 1 0 0 1 1 0 0 1  9   16.821371
8  1 1 1 1 0 1 0 1 0 1 0 0 1  9   21.265080
9  1 1 1 1 0 1 1 0 1 1 0 0 1 10   10.672309
9  1 1 1 1 0 1 1 1 0 1 0 0 1 10   14.852475
10 1 1 1 1 0 1 1 0 1 1 0 1 1 11    9.662092
10 1 1 1 1 1 1 1 0 1 1 0 0 1 11   11.620092
11 1 1 1 1 1 1 1 0 1 1 0 1 1 12   10.553564
11 1 1 1 1 0 1 1 1 1 1 0 1 1 12   11.178670
12 1 1 1 1 1 1 1 1 1 1 0 1 1 13   12.200163
12 1 1 1 1 1 1 1 0 1 1 1 1 1 13   12.352262
13 1 1 1 1 1 1 1 1 1 1 1 1 1 14   14.000000
```

The selected variables are

```
 [1] "Age"              "IMale"            "IMarried"
 [4] "EdCode"           "INortheast"       "ISouth"
 [7] "INotMetropolitan" "IFedGov"          "IStateGov"
[10] "IMale...IMarried"
```

c) (0.5 points) If we still consider the response variable and explanatory variables used in Quesiton 3 e) in Assignment 1 (please use indicator variables instead of the original categorical variables), please use R to obtain the variance inflation factors (VIF) for each of the explanatory variables, and paste them in the answer sheet. Based on the "rule of thumb" cut-off for VIF, does the multicollinearity problem exist if we regress the response on the explanatory variables?

Solution: The VIFs are

```
            Age           IMale        IMarried          EdCode
       1.085647        2.486618        2.196868        1.063858
        IMidwest      INortheast          ISouth    IMetropolitan
       1.583150        1.522163        1.617833       23.222433
INotMetropolitan         IFedGov       ILocalGov       IStateGov
      23.303701        1.021278        1.039201        1.047618
IMale...IMarried
       3.845590
```

The cut-off is 10. Hence, the multicollinearity problem exists.

d) (0.5 points) Please use the backward elimination idea to solve this multicollinearity problem and report the variables we should use in the regression model such that there is no multicollinearity problem. (Hint: similar to the codes on page 56 of Lecture Notes 8.)

Solution: The variables we should use and their final VIFs are

```
            Age           IMale        IMarried          EdCode
       1.085643        2.486562        2.196806        1.063564
        IMidwest      INortheast          ISouth INotMetropolitan
       1.567802        1.508072        1.604418        1.035306
         IFedGov       ILocalGov       IStateGov IMale...IMarried
       1.021165        1.039196        1.047613        3.845577
```

e) (0.5 points) Please paste the R codes for all the above analyses of Question 1 in the answer sheet.

Solution:

```
rm(list=ls())
setwd('~/Desktop/Research/AppliedStat2017/A2')
#Q1 a)
library('Sleuth3')
data=ex1225
attach(data)
```

```r
Y=log(WeeklyEarnings)
IMale=ifelse(Sex=="Male",1,0)
IMarried=ifelse(MaritalStatus=="Married",1,0)

IMidwest=ifelse(Region=="Midwest",1,0)
INortheast=ifelse(Region=="Northeast",1,0)
ISouth=ifelse(Region=="South",1,0)

IMetropolitan=ifelse(MetropolitanStatus=="Metropolitan",1,0)
INotMetropolitan=ifelse(MetropolitanStatus=="Not Metropolitan",1,0)

IFedGov=ifelse(JobClass=="FedGov",1,0)
ILocalGov=ifelse(JobClass=="LocalGov",1,0)
IStateGov=ifelse(JobClass=="StateGov",1,0)


X=data.frame(Age,IMale,IMarried,EdCode,IMidwest,INortheast,ISouth,
             IMetropolitan,INotMetropolitan,
             IFedGov,ILocalGov,IStateGov,IMale*IMarried)
library(leaps)
Cpsel=leaps(X,Y,method="Cp",nbest=2)
plot(Cpsel$size,Cpsel$Cp)
```

```r
#Q1 b)
cbind(Cpsel$which, Cpsel$size, Cpsel$Cp)

k=length(X[1,])
Variable=colnames(X)
result=cbind(Cpsel$which, Cpsel$size, Cpsel$Cp)
result=result[which(result[,k+2]==min(result[,k+2])),][1:k]

#Selected Variables
Variable[as.logical(result)]

#Q1 c)
fit=lm(Y~.,data=X)
library(car)
vif(fit)

#Q1 d)
```

```
#Try dropping IMetropolitan
X1=X[,-8]
fit1=lm(Y~.,data=X1)
d1=deviance(fit1)
#Try dropping INotMetropolitan
X2=X[,-9]
fit2=lm(Y~.,data=X2)
d2=deviance(fit2)
if (d1<d2){
  fit=fit1
  X=X1
}else{
  fit=fit2
  X=X2
}
vif(fit)
#Stop
detach(data)
```

Question 2 (Binary Logistic Regression, 2.0 points)

Bumpus Natural Selection Data (Revised based on ex 16 of Chapter 20 in "The Statistical Sleuth"). Hermon Bumpus analyzed various characteristics of some house sparrows that were found on the ground after a severe winter storm in 1898. Some of the sparrows survived and some perished. The data on male sparrows in Display 20.17 are survival status (1 = survived, 2 = perished), age (1 = adult, 2 = juvenile), the length from tip of beak to tip of tail (in mm), the alar extent (length from tip to tip of the extended wings, in mm), the weight in grams, the length of the head in mm, the length of the humerus (arm bone, in inches), the length of the femur (thigh bones, in inches), the length of the tibio-tarsus (leg bone, in inches), the breadth of the skull in inches, and the length of the sternum in inches. The dataset is stored in the object "ex2016" of the R library "Sleuth3".

Analyze the data to see whether the probability of survival is associated with physical characteristics of the birds. This would be consistent, according to Bumpus, with the theory of natural selection: those that survived did so because of some superior physical traits. Realize that the sampling is from a population of grounded sparrows.

DISPLAY 20.17 A subset of data on 51 male sparrows that survived (SV = 1) and 36 that perished (SV = 2) during a severe winter storm: Age (*AG*) is 1 for adults, 2 for juveniles; *TL* is total length; *AE* is alar extent; *WT* is weight; *BH* is length of beak and head; *HL* is length of humerus; *FL* is length of femur; *TT* is length of tibio-tarsus; *SK* is width of skull; and *KL* is length of keel of sternum

SV	*AG*	*TL*	*AE*	*WT*	*BH*	*HL*	*FL*	*TT*	*SK*	*KL*
1	1	154	241	24.5	31.2	0.687	0.668	1.022	0.587	0.830
1	1	160	252	26.9	30.8	0.736	0.709	1.180	0.602	0.841
1	1	155	243	26.9	30.6	0.733	0.704	1.151	0.602	0.846
1	1	154	245	24.3	31.7	0.741	0.688	1.146	0.584	0.839
1	1	156	247	24.1	31.5	0.715	0.706	1.129	0.575	0.821
2	1	162	247	27.6	31.8	0.731	0.719	1.113	0.597	0.869
2	1	163	246	25.8	31.4	0.689	0.662	1.073	0.604	0.836
2	1	161	246	24.9	30.5	0.739	0.726	1.138	0.580	0.803
2	1	160	242	26.0	31.0	0.745	0.713	1.105	0.600	0.803
2	1	162	246	26.5	31.5	0.720	0.696	1.092	0.606	0.809

Display taken from class text: "The Statistical Sleuth".

In order to investigate the above problem, please use R to answer the following questions in the answer sheet.

(The order of the levels of the specified factor(s) can be set differently in this question. But the result should be the same.)

The first one possible solution to Q2

a) (0.5 points) Please use the R function "factor()" to transform the vector "AG" in "ex2016" into a vector of factor values in the data frame. Then use R to regress survival status on all the other variables, including all the interactions between "AG" and the other continuous explanatory variables, in order to answer the question whether the probability of survival is associated with physical characteristics of the birds. Please do not use the indicator variable of "AG", but instead, use the factor values of "AG" directly in the fitting of the regression (Hint: Similar to page 12 of Lecture Notes 11). Based on the "summary" function output of this fitted model, can we use the "Null deviance" and "Residual deviance" in the output to construct a drop-in-deviance χ^2-test? If we can, what are the null hypothesis and the alternative hypothesis of this test?

Solution:

```
Call:
glm(formula = indStatus ~ AG + TL + AE + WT + BH + HL + FL +
    TT + SK + KL + AG * TL + AG * AE + AG * WT + AG * BH + AG *
    HL + AG * FL + AG * TT + AG * SK + AG * KL, family = binomial(link = logit))

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.86296  -0.06131   0.00000   0.23075   2.22454

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.291e+01  4.095e+01   1.048   0.2947
AG2         -1.686e+03  6.495e+05  -0.003   0.9979
TL          -9.242e-01  2.854e-01  -3.238   0.0012 **
AE           2.107e-01  1.870e-01   1.127   0.2599
WT          -9.999e-01  4.788e-01  -2.088   0.0368 *
BH          -4.987e-01  8.416e-01  -0.593   0.5535
HL           1.787e+01  4.428e+01   0.404   0.6865
FL           6.635e+01  5.225e+01   1.270   0.2042
TT           5.028e+00  2.298e+01   0.219   0.8269
SK           1.815e+01  3.546e+01   0.512   0.6088
KL           2.156e+01  1.731e+01   1.246   0.2129
AG2:TL      -1.331e+01  1.318e+04  -0.001   0.9992
```

```
AG2:AE      -1.588e+00  9.017e+03   0.000   0.9999
AG2:WT      -9.665e+00  1.070e+04  -0.001   0.9993
AG2:BH       8.377e+01  2.405e+04   0.003   0.9972
AG2:HL       1.361e+03  1.477e+06   0.001   0.9993
AG2:FL      -7.073e+02  1.605e+06   0.000   0.9996
AG2:TT      -1.006e+03  4.987e+05  -0.002   0.9984
AG2:SK       3.029e+03  1.217e+06   0.002   0.9980
AG2:KL       7.511e+02  4.767e+05   0.002   0.9987
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 118.008  on 86  degrees of freedom
Residual deviance:  38.207  on 67  degrees of freedom
AIC: 78.207

Number of Fisher Scoring iterations: 21
```

Yes. The null hypothesis is that: the regression coefficients of all the variables, except for the intercept, are zeros. The alternative hypothesis is that: at least one of those coefficients is not zero.

If we have warnings in R, it is OK and sometimes we can ignore them as in this case. However, if we have an error in R, we should deal with it.

b) (0.5 points) If we can construct a drop-in-deviance χ^2-test in the above question, please use R to accomplish this χ^2-test. What is the value of the test statistic? What conclusion can you obtain for this χ^2-test? If we cannot construct a drop-in-deviance χ^2-test in the above question, please state your reasons.

Solution:

```
Analysis of Deviance Table

Model 1: indStatus ~ 1
Model 2: indStatus ~ AG + TL + AE + WT + BH + HL + FL + TT + SK + KL +
    AG * TL + AG * AE + AG * WT + AG * BH + AG * HL + AG * FL +
    AG * TT + AG * SK + AG * KL
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1        86    118.008
```

```
2        67     38.207 19   79.802 2.012e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The value of the test statistic is 79.802. The p-value is smaller than 0.05. Hence, we reject the null hypothesis and conclude that at least one of the considered variables is useful in explaining the probability of survival.

c) (0.5 points) Consider all the variables involved in Question 2 a) and use R to perform the forward selection based on BIC. Which variables should we choose to predict the probability of survival by using this variable selection method?

Solution: TL, HL, WT and KL.

Note: Compare the following two approaches for variable selection:

1. Use "*" to represent interactions.

```
attach(data)
library(MASS)
n=length(data[,1])
a=stepAIC(fitr, direction = "forward",
          scope=list(lower=~1,upper=~
                       AG+TL+AE+WT+BH+HL+FL+TT+SK+KL
                     +AG*TL+AG*AE+AG*WT+AG*BH+AG*HL
                     +AG*FL+AG*TT+AG*SK+AG*KL),k=log(n))
```

```
Start:  AIC=122.47
indStatus ~ 1

       Df Deviance    AIC
+ TL    1   99.788 108.72
+ WT    1  111.249 120.18
<none>     118.008 122.47
+ HL    1  114.431 123.36
+ KL    1  116.321 125.25
+ FL    1  116.521 125.45
+ TT    1  116.942 125.87
+ BH    1  117.090 126.02
+ SK    1  117.854 126.79
```

```
+ AG    1  117.971 126.90
+ AE    1  117.986 126.92

Step:  AIC=108.72
indStatus ~ TL

       Df Deviance     AIC
+ HL    1   80.020  93.418
+ FL    1   85.223  98.621
+ AE    1   87.703 101.101
+ KL    1   89.162 102.559
+ TT    1   89.618 103.016
+ BH    1   92.820 106.218
+ SK    1   93.844 107.241
<none>      99.788 108.719
+ WT    1   99.546 112.944
+ AG    1   99.755 113.153

Step:  AIC=93.42
indStatus ~ TL + HL

       Df Deviance    AIC
+ WT    1   75.094 92.958
<none>      80.020 93.418
+ KL    1   76.708 94.572
+ SK    1   78.537 96.401
+ BH    1   78.846 96.709
+ AE    1   79.656 97.519
+ TT    1   79.775 97.639
+ FL    1   79.857 97.720
+ AG    1   80.020 97.884

Step:  AIC=92.96
indStatus ~ TL + HL + WT

       Df Deviance    AIC
+ KL    1   68.612 90.942
<none>      75.094 92.958
+ BH    1   72.512 94.842
+ SK    1   73.451 95.780
+ AE    1   74.450 96.779
```

```
+ TT    1   74.460 96.790
+ FL    1   74.789 97.118
+ AG    1   75.076 97.406

Step:  AIC=90.94
indStatus ~ TL + HL + WT + KL

       Df Deviance    AIC
<none>      68.612 90.942
+ BH    1   67.214 94.010
+ SK    1   67.496 94.291
+ TT    1   68.206 95.001
+ AE    1   68.334 95.130
+ FL    1   68.541 95.336
+ AG    1   68.612 95.407
```

```
summary(a)
```

```
Call:
glm(formula = indStatus ~ TL + HL + WT + KL, family = binomial(link = logit))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2234  -0.5648   0.1540   0.6094   2.2701

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  49.9861    18.4879   2.704 0.006857 **
TL           -0.6573     0.1683  -3.907 9.35e-05 ***
HL           72.3327    20.7640   3.484 0.000495 ***
WT           -0.7896     0.3097  -2.549 0.010800 *
KL           27.3775    11.7780   2.324 0.020101 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 118.008  on 86  degrees of freedom
Residual deviance:  68.612  on 82  degrees of freedom
AIC: 78.612
```

```
Number of Fisher Scoring iterations: 6
```

It is worth noting that by using this method, since the categorical variable "AG" has never been selected in the process of the forward selection, R never involves the interactions of "AG" and the other continuous variables in the middle of the forward selection procedure by default. The reason for it is that by using "*", R can recognize the interaction terms, and hence R follows the rule on page 19 of Lecture Notes 5:

"Except in special circumstances, a model including a product term for interaction between two explanatory variables should also include terms with each of the explanatory variables individually."

2. Construct an indicator variable and a lot of new variables to obtain interactions.

```
AG2=ifelse(AG=='2',1,0)
AG2TL=AG2*TL
AG2AE=AG2*AE
AG2WT=AG2*WT
AG2BH=AG2*BH
AG2HL=AG2*HL
AG2FL=AG2*FL
AG2TT=AG2*TT
AG2SK=AG2*SK
AG2KL=AG2*KL

a=stepAIC(fitr, direction = "forward",
          scope=list(lower=~1,upper=~
                       AG2+TL+AE+WT+BH+HL+FL+TT+SK+KL
                     +AG2TL+AG2AE+AG2WT+AG2BH+AG2HL
                     +AG2FL+AG2TT+AG2SK+AG2KL),k=log(n))
```

```
Start:  AIC=122.47
indStatus ~ 1

        Df Deviance    AIC
+ TL     1   99.788 108.72
+ WT     1  111.249 120.18
<none>      118.008 122.47
+ HL     1  114.431 123.36
+ KL     1  116.321 125.25
```

```
+ FL     1  116.521 125.45
+ TT     1  116.942 125.87
+ BH     1  117.090 126.02
+ SK     1  117.854 126.79
+ AG2WT  1  117.927 126.86
+ AG2TL  1  117.949 126.88
+ AG2AE  1  117.966 126.90
+ AG2TT  1  117.968 126.90
+ AG2FL  1  117.970 126.90
+ AG2    1  117.971 126.90
+ AG2KL  1  117.983 126.92
+ AG2HL  1  117.984 126.92
+ AE     1  117.986 126.92
+ AG2SK  1  117.986 126.92
+ AG2BH  1  117.989 126.92

Step:  AIC=108.72
indStatus ~ TL

        Df Deviance     AIC
+ HL     1   80.020  93.418
+ FL     1   85.223  98.621
+ AE     1   87.703 101.101
+ KL     1   89.162 102.559
+ TT     1   89.618 103.016
+ BH     1   92.820 106.218
+ SK     1   93.844 107.241
<none>       99.788 108.719
+ WT     1   99.546 112.944
+ AG2KL  1   99.696 113.094
+ AG2HL  1   99.708 113.106
+ AG2BH  1   99.718 113.115
+ AG2SK  1   99.718 113.116
+ AG2FL  1   99.728 113.125
+ AG2TT  1   99.733 113.131
+ AG2AE  1   99.737 113.135
+ AG2WT  1   99.747 113.144
+ AG2TL  1   99.747 113.145
+ AG2    1   99.755 113.153

Step:  AIC=93.42
```

```
indStatus ~ TL + HL

        Df Deviance    AIC
+ WT     1   75.094 92.958
<none>       80.020 93.418
+ KL     1   76.708 94.572
+ SK     1   78.537 96.401
+ BH     1   78.846 96.709
+ AE     1   79.656 97.519
+ TT     1   79.775 97.639
+ FL     1   79.857 97.720
+ AG2KL  1   80.015 97.879
+ AG2BH  1   80.016 97.879
+ AG2SK  1   80.016 97.880
+ AG2WT  1   80.019 97.883
+ AG2TL  1   80.020 97.884
+ AG2HL  1   80.020 97.884
+ AG2AE  1   80.020 97.884
+ AG2    1   80.020 97.884
+ AG2FL  1   80.020 97.884
+ AG2TT  1   80.020 97.884

Step:  AIC=92.96
indStatus ~ TL + HL + WT

        Df Deviance    AIC
+ KL     1   68.612 90.942
<none>       75.094 92.958
+ BH     1   72.512 94.842
+ SK     1   73.451 95.780
+ AE     1   74.450 96.779
+ TT     1   74.460 96.790
+ FL     1   74.789 97.118
+ AG2FL  1   75.072 97.402
+ AG2TT  1   75.075 97.405
+ AG2    1   75.076 97.406
+ AG2AE  1   75.077 97.406
+ AG2HL  1   75.080 97.409
+ AG2TL  1   75.080 97.410
+ AG2WT  1   75.082 97.412
+ AG2SK  1   75.087 97.417
```

```
+ AG2BH  1   75.089 97.419
+ AG2KL  1   75.090 97.419

Step:  AIC=90.94
indStatus ~ TL + HL + WT + KL

        Df Deviance    AIC
<none>       68.612 90.942
+ BH     1   67.214 94.010
+ SK     1   67.496 94.291
+ TT     1   68.206 95.001
+ AE     1   68.334 95.130
+ FL     1   68.541 95.336
+ AG2BH  1   68.606 95.401
+ AG2SK  1   68.609 95.404
+ AG2WT  1   68.610 95.406
+ AG2KL  1   68.611 95.406
+ AG2TL  1   68.611 95.407
+ AG2HL  1   68.612 95.407
+ AG2    1   68.612 95.407
+ AG2AE  1   68.612 95.407
+ AG2FL  1   68.612 95.407
+ AG2TT  1   68.612 95.407
```

```
summary(a)
```

```
Call:
glm(formula = indStatus ~ TL + HL + WT + KL, family = binomial(link = logit))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2234  -0.5648   0.1540   0.6094   2.2701

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  49.9861    18.4879   2.704 0.006857 **
TL           -0.6573     0.1683  -3.907 9.35e-05 ***
HL           72.3327    20.7640   3.484 0.000495 ***
WT           -0.7896     0.3097  -2.549 0.010800 *
KL           27.3775    11.7780   2.324 0.020101 *
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 118.008  on 86  degrees of freedom
Residual deviance:  68.612  on 82  degrees of freedom
AIC: 78.612

Number of Fisher Scoring iterations: 6
```

```
detach(data)
```

In this case, still the indicator variable of "AG" has never been selected in the process of the forward selection. However, R involves the interactions, for instance "AG2TL" in the middle of the forward selection procedure. The reason for it is that by not using "*", R cannot recognize the interaction terms, and R treats for instance "AG2TL" as a new variable. Hence, R cannot follow the rule on page 19 of Lecture Notes 5.

However, it is lucky that even though the processes of the forward selection are differnt for these two methods, but the results are exactly the same.

d) (0.5 points) Please paste the R codes for all the above analyses of Question 2 in the answer sheet.

Solution:

```
#Q2 a)
data=ex2016
data$AG=factor(data$AG)
attach(data)
indStatus=ifelse(Status=='Survived', 1, 0)
fit=glm(indStatus~AG+TL+AE+WT+BH+HL+FL+TT+SK+KL
        +AG*TL+AG*AE+AG*WT+AG*BH+AG*HL+AG*FL+AG*TT+AG*SK+AG*KL,
        family=binomial(link=logit))
```

```
Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
summary(fit)

#Q2 b)
fitr=glm(indStatus~1,family=binomial(link=logit))
anova(fitr,fit,test='Chisq')

#Q2 c)
n=length(data[,1])
library(MASS)
a=stepAIC(fitr, direction = "forward",
          scope=list(lower=~1,upper=~
                       AG+TL+AE+WT+BH+HL+FL+TT+SK+KL
                     +AG*TL+AG*AE+AG*WT+AG*BH+AG*HL
                     +AG*FL+AG*TT+AG*SK+AG*KL),k=log(n))
summary(a)
detach(data)
```

An alternative solution to Q2

```
#Q2 a)
data=ex2016
data$AG=factor(data$AG)
attach(data)
library(nnet)
fit=multinom(formula=Status~AG+TL+AE+WT+BH+HL+FL+TT+SK+KL
        +AG*TL+AG*AE+AG*WT+AG*BH+AG*HL+AG*FL+AG*TT+AG*SK+AG*KL,
        data=data)
```

```
# weights:  21 (20 variable)
initial  value 60.303805
iter  10 value 36.065349
iter  20 value 28.855480
iter  30 value 24.749606
iter  40 value 23.908054
iter  50 value 23.249142
iter  60 value 22.610742
iter  70 value 22.092953
iter  80 value 21.211306
iter  90 value 20.807424
iter 100 value 20.573644
final  value 20.573644
stopped after 100 iterations
```

```
summary(fit)
```

```
Call:
multinom(formula = Status ~ AG + TL + AE + WT + BH + HL + FL +
    TT + SK + KL + AG * TL + AG * AE + AG * WT + AG * BH + AG *
    HL + AG * FL + AG * TT + AG * SK + AG * KL, data = data)

Coefficients:
                  Values   Std. Err.
(Intercept)   33.6029776 0.065975515
AG2          -56.9292941 0.008284374
TL            -0.9211652 0.223476693
AE             0.2488094 0.135123299
WT            -0.9924911 0.400451295
BH            -0.5029843 0.730334551
```

```
HL            -2.1673717 0.105636153
FL            65.6564040 0.320894313
TT            10.8590585 0.643486615
SK            29.7744077 0.020546571
KL            22.6787092 0.398815843
AG2:TL        -2.4267263 0.865364664
AG2:AE        -0.4334210 0.606356138
AG2:WT         1.8196057 1.136393926
AG2:BH        16.1223571 5.124445737
AG2:HL        -1.5971545 0.065732335
AG2:FL       -58.9177337 0.286564849
AG2:TT      -117.5198721 0.531013048
AG2:SK       124.7415362 0.065087021
AG2:KL       119.2969852 0.303514090

Residual Deviance: 41.14729
AIC: 81.14729
```

So the answer is no for Q2 a).

```
#Q2 b)
```

The reason is that in the output we only have "Residual Deviance" but we do not have "Null deviance".

```
#Q2 c)
fitr=multinom(formula=Status~1,data=data)
```

```
# weights:  2 (1 variable)
initial  value 60.303805
final  value 59.004217
converged
```

```
n=length(data[,1])
library(MASS)
a=stepAIC(fitr, direction = "forward",
          scope=list(lower=~1,upper=~
                       AG+TL+AE+WT+BH+HL+FL+TT+SK+KL
                     +AG*TL+AG*AE+AG*WT+AG*BH+AG*HL
                     +AG*FL+AG*TT+AG*SK+AG*KL),k=log(n))
```

```
Start:  AIC=122.47
Status ~ 1

# weights:  3 (2 variable)
initial  value 60.303805
final  value 58.985662
converged
# weights:  3 (2 variable)
initial  value 60.303805
iter  10 value 54.188870
iter  20 value 51.215198
iter  30 value 50.380856
iter  40 value 50.086472
iter  50 value 49.970234
iter  60 value 49.923619
iter  70 value 49.905194
iter  80 value 49.898066
iter  90 value 49.895362
iter 100 value 49.894351
final  value 49.894351
stopped after 100 iterations
# weights:  3 (2 variable)
initial  value 60.303805
iter  10 value 59.002872
iter  20 value 58.995733
iter  30 value 58.993639
iter  40 value 58.993116
iter  50 value 58.992988
final  value 58.992971
converged
# weights:  3 (2 variable)
initial  value 60.303805
final  value 55.624467
converged
# weights:  3 (2 variable)
initial  value 60.303805
iter  10 value 58.654815
iter  20 value 58.561781
iter  30 value 58.547790
iter  40 value 58.545542
iter  50 value 58.545177
```

```
final  value 58.545119
converged
# weights:  3 (2 variable)
initial  value 60.303805
iter  10 value 57.880685
iter  20 value 57.402901
iter  30 value 57.271213
iter  40 value 57.232358
iter  50 value 57.220595
iter  60 value 57.216983
iter  70 value 57.215866
iter  80 value 57.215519
iter  90 value 57.215411
final  value 57.215380
converged
# weights:  3 (2 variable)
initial  value 60.303805
iter  10 value 58.473117
iter  20 value 58.312120
iter  30 value 58.273726
iter  40 value 58.263938
iter  50 value 58.261429
iter  60 value 58.260784
iter  70 value 58.260618
final  value 58.260577
converged
# weights:  3 (2 variable)
initial  value 60.303805
iter  10 value 58.547492
iter  20 value 58.470757
iter  20 value 58.470757
iter  20 value 58.470757
final  value 58.470757
converged
# weights:  3 (2 variable)
initial  value 60.303805
iter  10 value 58.975101
iter  20 value 58.952612
iter  30 value 58.940684
iter  40 value 58.934344
iter  50 value 58.930977
```

```
iter  60 value 58.929191
iter  70 value 58.928244
iter  80 value 58.927743
iter  90 value 58.927478
iter 100 value 58.927338
final  value 58.927338
stopped after 100 iterations
# weights:  3 (2 variable)
initial  value 60.303805
iter  10 value 58.239058
iter  20 value 58.161478
final  value 58.160545
converged
       Df    AIC
+ TL    1 108.72
+ WT    1 120.18
<none>    122.47
+ HL    1 123.36
+ KL    1 125.25
+ FL    1 125.45
+ TT    1 125.87
+ BH    1 126.02
+ SK    1 126.79
+ AG    1 126.90
+ AE    1 126.92
# weights:  3 (2 variable)
initial  value 60.303805
iter  10 value 54.188870
iter  20 value 51.215198
iter  30 value 50.380856
iter  40 value 50.086472
iter  50 value 49.970234
iter  60 value 49.923619
iter  70 value 49.905194
iter  80 value 49.898066
iter  90 value 49.895362
iter 100 value 49.894351
final  value 49.894351
stopped after 100 iterations

Step:  AIC=108.72
```

```
Status ~ TL

# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 52.503412
iter  20 value 50.116190
iter  30 value 49.878379
final  value 49.877721
converged
# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 45.830493
iter  20 value 43.900746
iter  30 value 43.862523
iter  40 value 43.851975
final  value 43.851415
converged
# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 51.877512
iter  20 value 49.809181
final  value 49.773203
converged
# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 48.800591
final  value 46.409975
converged
# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 40.659332
iter  20 value 40.011655
iter  30 value 40.010240
iter  30 value 40.010240
iter  30 value 40.010240
final  value 40.010240
converged
# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 43.284388
iter  20 value 42.648728
```

```
iter  30 value 42.611516
iter  30 value 42.611516
iter  30 value 42.611516
final  value 42.611516
converged
# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 48.566190
iter  20 value 45.136108
iter  30 value 44.814666
final  value 44.809022
converged
# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 47.466759
iter  20 value 46.986402
iter  30 value 46.923888
iter  40 value 46.922168
final  value 46.921907
converged
# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 47.563737
iter  20 value 44.720626
iter  30 value 44.581348
final  value 44.580896
converged
       Df     AIC
+ HL    1  93.418
+ FL    1  98.621
+ AE    1 101.101
+ KL    1 102.560
+ TT    1 103.016
+ BH    1 106.218
+ SK    1 107.242
<none>    108.721
+ WT    1 112.944
+ AG    1 113.153
# weights:  4 (3 variable)
initial  value 60.303805
iter  10 value 40.659332
```

```
iter  20 value 40.011655
iter  30 value 40.010240
iter  30 value 40.010240
iter  30 value 40.010240
final  value 40.010240
converged

Step:  AIC=93.42
Status ~ TL + HL

# weights:  5 (4 variable)
initial  value 60.303805
iter  10 value 40.232066
final  value 40.013057
converged
# weights:  5 (4 variable)
initial  value 60.303805
iter  10 value 43.806573
iter  20 value 41.039848
iter  30 value 39.830218
final  value 39.827870
converged
# weights:  5 (4 variable)
initial  value 60.303805
iter  10 value 37.934739
iter  20 value 37.549108
final  value 37.547180
converged
# weights:  5 (4 variable)
initial  value 60.303805
iter  10 value 40.109513
iter  20 value 39.451745
final  value 39.424485
converged
# weights:  5 (4 variable)
initial  value 60.303805
iter  10 value 41.786257
iter  20 value 40.066569
iter  30 value 40.002743
iter  40 value 39.966019
iter  50 value 39.937290
```

```
iter  60 value 39.932722
iter  70 value 39.929912
iter  80 value 39.928960
final  value 39.928881
converged
# weights:  5 (4 variable)
initial  value 60.303805
iter  10 value 45.981705
iter  20 value 40.031167
iter  30 value 39.893712
iter  40 value 39.888799
iter  50 value 39.887689
final  value 39.887586
converged
# weights:  5 (4 variable)
initial  value 60.303805
iter  10 value 39.303214
iter  20 value 39.270837
final  value 39.268774
converged
# weights:  5 (4 variable)
initial  value 60.303805
iter  10 value 42.623391
iter  20 value 38.354807
final  value 38.354198
converged
       Df    AIC
+ WT    1 92.958
<none>    93.418
+ KL    1 94.572
+ SK    1 96.401
+ BH    1 96.713
+ AE    1 97.519
+ TT    1 97.639
+ FL    1 97.721
+ AG    1 97.890
# weights:  5 (4 variable)
initial  value 60.303805
iter  10 value 37.934739
iter  20 value 37.549108
final  value 37.547180
```

```
converged

Step:  AIC=92.96
Status ~ TL + HL + WT

# weights:  6 (5 variable)
initial  value 60.303805
iter  10 value 39.093701
iter  20 value 37.563201
iter  30 value 37.538097
final  value 37.538018
converged
# weights:  6 (5 variable)
initial  value 60.303805
iter  10 value 42.935023
iter  20 value 38.259102
iter  30 value 37.263738
iter  40 value 37.230232
iter  50 value 37.225019
iter  50 value 37.225018
iter  50 value 37.225018
final  value 37.225018
converged
# weights:  6 (5 variable)
initial  value 60.303805
iter  10 value 45.733262
iter  20 value 38.834679
iter  30 value 36.273884
final  value 36.256152
converged
# weights:  6 (5 variable)
initial  value 60.303805
iter  10 value 38.580427
iter  20 value 37.636307
iter  30 value 37.423686
iter  40 value 37.418657
final  value 37.404377
converged
# weights:  6 (5 variable)
initial  value 60.303805
iter  10 value 42.718842
```

```
iter  20 value 39.038223
iter  30 value 38.016800
iter  40 value 37.565837
iter  50 value 37.449663
final  value 37.333332
converged
# weights:  6 (5 variable)
initial  value 60.303805
iter  10 value 37.166492
iter  20 value 36.732281
iter  30 value 36.725485
iter  40 value 36.725424
final  value 36.725377
converged
# weights:  6 (5 variable)
initial  value 60.303805
iter  10 value 38.023532
iter  20 value 34.794106
iter  30 value 34.541190
iter  40 value 34.312349
iter  50 value 34.310188
final  value 34.305993
converged
       Df    AIC
+ KL    1 90.942
<none>    92.958
+ BH    1 94.842
+ SK    1 95.780
+ AE    1 96.780
+ TT    1 96.996
+ FL    1 97.138
+ AG    1 97.406
# weights:  6 (5 variable)
initial  value 60.303805
iter  10 value 38.023532
iter  20 value 34.794106
iter  30 value 34.541190
iter  40 value 34.312349
iter  50 value 34.310188
final  value 34.305993
converged
```

```
Step:  AIC=90.94
Status ~ TL + HL + WT + KL

# weights:  7 (6 variable)
initial  value 60.303805
iter  10 value 38.148262
iter  20 value 36.713850
iter  30 value 34.514180
iter  40 value 34.314386
iter  40 value 34.314386
final  value 34.314386
converged
# weights:  7 (6 variable)
initial  value 60.303805
iter  10 value 40.695617
iter  20 value 35.502930
iter  30 value 34.468330
iter  40 value 34.220309
iter  40 value 34.220309
final  value 34.220309
converged
# weights:  7 (6 variable)
initial  value 60.303805
iter  10 value 43.767801
iter  20 value 34.438463
iter  30 value 33.763126
iter  40 value 33.620038
iter  50 value 33.609221
final  value 33.609125
converged
# weights:  7 (6 variable)
initial  value 60.303805
iter  10 value 37.052140
iter  20 value 34.784843
final  value 34.460509
converged
# weights:  7 (6 variable)
initial  value 60.303805
iter  10 value 39.305451
iter  20 value 35.764453
```

```
iter  30 value 34.619319
iter  40 value 34.125050
iter  50 value 34.111414
iter  60 value 34.104538
iter  70 value 34.103646
iter  80 value 34.103272
iter  90 value 34.103105
final  value 34.103089
converged
# weights:  7 (6 variable)
initial  value 60.303805
iter  10 value 36.665535
iter  20 value 34.529791
iter  30 value 33.856005
iter  40 value 33.759676
iter  50 value 33.751082
iter  60 value 33.750091
final  value 33.748225
converged
       Df    AIC
<none>    90.942
+ BH    1 94.014
+ SK    1 94.292
+ TT    1 95.002
+ AE    1 95.236
+ AG    1 95.424
+ FL    1 95.716
```

```
summary(a)
```

```
Call:
multinom(formula = Status ~ TL + HL + WT + KL, data = data)

Coefficients:
                Values   Std. Err.
(Intercept) 49.9654093  2.16545745
TL          -0.6574626  0.04703282
HL          72.3905856  0.76928867
WT          -0.7895921  0.27146103
KL          27.3735525 11.08331971
```

```
Residual Deviance: 68.61199
AIC: 78.61199
```

```
detach(data)
```

The selected variables are TL, HL, WT and KL.

Question 3 (Multicategory Response Regression, 4.0 points)

(Revised based on ex 16 of Chapter 3 in "Analysis of Categorical Data with R".)

Researchers at Penn State University performed a study to determine the optimal fat content for ice cream. Details of the study and corresponding data analysis are available at https://onlinecourses.science.psu.edu/stat504/node/187. In summary, 496 individuals were asked to taste and then rate a particular type of ice cream on a 9-point scale (1 to 9 with 1 equating to not liking and 9 equating to really liking). The ice cream given to the individuals had a fat proportion level of 0, 0.04, ..., or 0.28 (fat). We treat "fat" as continuous variable in this question.

The data for 496 individuals is randomly split into two parts. The training dataset (including 471 individuals) and the test dataset (including 25 individulas) are available in "ice_cream1.csv" and "ice_cream2.csv" on the Wattle, respectively. Using these data, please use R to answer the following questions in the answer sheet.

a) (0.5 points) Obviously, the ratings 1-9 are ordinal data. Please use the ordinal response regression model to regress the ordinal response "rating" on "fat" by utilizing the training dataset only. How many unknown regression parameters in the above ordinal response regression model? (Hint: similar to Question 2, please use the R function "factor()" to transform the vector "rating" in the data into a vector of factor values in the data frame first.)

Solution:

```
Call:
polr(formula = factor(rating) ~ fat, data = a, method = "logistic")

Coefficients:
    Value Std. Error t value
fat 1.157     0.9072   1.275

Intercepts:
    Value    Std. Error t value
1|2  -3.5796   0.3272   -10.9414
2|3  -2.0914   0.1978   -10.5740
3|4  -1.4739   0.1749    -8.4280
4|5  -0.6902   0.1601    -4.3115
5|6  -0.3831   0.1575    -2.4329
```

```
6|7   0.2934   0.1565     1.8748
7|8   1.2573   0.1655     7.5960
8|9   3.2277   0.2576    12.5300

Residual Deviance: 1889.248
AIC: 1907.248
```

In the summary output, we can see that 9 regression coefficients are estimated. Hence, we have 9 unknown regression parameters in the above ordinal response regression model.

b) (0.5 points) What is the 95% confidence interval for the coefficient of "fat" (rounded to four decimal places) based on the above fitted ordinal response regression model?

Solution: The CI is

```
[1] -0.6211
```

```
[1] 2.9352
```

c) (0.5 points) If we are interested in testing whether or not "fat" is needed based on the above fitted ordinal response regression model, please construct an appropriate test. What is the p-value for your test (rounded to four decimal places)? What conclusion can you obtain based on the p-value?

Solution:

```
Analysis of Deviance Table (Type II tests)

Response: factor(rating)
    LR Chisq Df Pr(>Chisq)
fat   1.6264  1     0.2022
```

The p-value is 0.2022, which is larger than 0.05. Hence we cannot reject the null hypothesis and conclude that we **do not have enough evidencen to reject** that "fat" is not needed in the model.

d) (0.5 points) Suppose we ignore the order of the ratings 1-9 in this part and treat "rating" as a nominal response. Please use the nominal response regression model to regress the nominal response "rating" on "fat"" by utilizing the training dataset only. How many unknown regression parameters in the above nominal response regression model?

Solution:

```
# weights:  27 (16 variable)
initial  value 1034.892776
iter  10 value 942.039992
iter  20 value 940.091015
final  value 940.090049
converged

Call:
multinom(formula = factor(rating) ~ fat, data = a)

Coefficients:
  (Intercept)         fat
2  1.64810030 -4.31010722
3  1.12006584 -0.38265708
4  1.58967982  1.19634382
5  1.29658445 -1.74724366
6  1.78607313  1.20603291
7  2.22945709 -0.01750565
8  2.21143647 -0.25274541
9 -0.03093644  4.34274863

Std. Errors:
  (Intercept)      fat
2   0.6092965 3.855028
3   0.6319170 3.832246
4   0.5971711 3.579825
5   0.6240487 3.841323
6   0.5885707 3.532210
7   0.5759994 3.478975
8   0.5768812 3.487836
9   0.7317725 4.147323

Residual Deviance: 1880.18
AIC: 1912.18
```

In the summary output, we can see that 16 regression coefficients are estimated. Hence, we have 16 unknown regression parameters in the above nominal response regression model.

e) (0.5 points) If we are interested in testing whether or not "fat" is needed based on the above fitted nominal response regression model, please construct an appropriate test. What is the p-value for your test (rounded to four decimal places)? What conclusion can you obtain based on the p-value?

Solution:

```
# weights:  18 (8 variable)
initial  value 1034.892776
iter  10 value 946.456808
final  value 945.436991
converged

Analysis of Deviance Table (Type II tests)

Response: factor(rating)
    LR Chisq Df Pr(>Chisq)
fat   10.694  8     0.2197
```

The p-value is 0.2197, which is larger than 0.05. Hence we cannot reject the null hypothesis and conclude that we **do not have enough evidencen to reject** that "fat" is not needed in the model.

f) (0.5 points) Now we consider the test dataset. Let Y_ℓ be the "rating" for observation $\ell = 1, \cdots, n_{\text{test}}$ in the test dataset. Suppose $\hat{Y}_\ell$ to be the corresponding prediction of response based on either of the above two fitted models from the training dataset. Define an indicator variable

$$I\{Y_\ell = \hat{Y}_\ell\} = \begin{cases} 1, & \text{if } Y_\ell = \hat{Y}_\ell; \\ 0, & \text{otherwise.} \end{cases}$$

Then the percentage of correct forecast (PCF) can be defined by

$$\text{PCF} = \frac{1}{n_{\text{test}}} \sum_{\ell=1}^{n_{\text{test}}} I\{Y_\ell = \hat{Y}_\ell\}.$$

Based on the definition, what is the PCF for the ordinal response regression model?

Solution: The PCF is

```
[1] 0.24
```

g) (0.5 points) What is the PCF for the nominal response regression model? Based on these two PCFs, which model is better?

Solution: The PCF is

```
[1] 0.2
```

Since the PCF for the ordinal response regression model is larger, the ordinal model is better.

h) (0.5 points) Please paste the R codes for all the above analyses of Question 3 in the answer sheet.

Solution:

```r
#Q3 a)
a=read.csv(file='ice_cream1.csv')
library(MASS)
mod.fit.ord=polr(factor(rating)~fat,data=a,method='logistic')
summary(mod.fit.ord)
```

```
Re-fitting to get Hessian
```

```r
#Q3 b)
round(summary(mod.fit.ord)$coefficient[1,1]
      -qnorm(0.975)*summary(mod.fit.ord)$coefficient[1,2],4)
```

```
Re-fitting to get Hessian


Re-fitting to get Hessian
```

```
round(summary(mod.fit.ord)$coefficient[1,1]
      +qnorm(0.975)*summary(mod.fit.ord)$coefficient[1,2],4)
```

```
Re-fitting to get Hessian
```

```
Re-fitting to get Hessian
```

```
#Q3 c)
Anova(mod.fit.ord)

#Q3 d)
library(nnet)
mod.fit<-multinom(formula=factor(rating)~fat,data=a)
summary(mod.fit)

#Q3 e)
Anova(mod.fit)

#Q3 f)
b=read.csv(file='ice_cream2.csv')
round(mean(as.numeric(as.numeric(predict(mod.fit.ord,b,
                                         type='class'))==b$rating)),4)

#Q3 g)
round(mean(as.numeric(as.numeric(predict(mod.fit,b,
                                         type='class'))==b$rating)),4)
```

Question 4 (Simulation for Binary Logistic Regression, 1.5 points)

Consider the binary logistic regression model $\text{logit}(\mu\{Y|X\}) = \beta_0 + \beta_1 X$ for the observations $\{Y_i, X_i\}_{i=1}^n$, and the maximum likelihood estimation (MLE) $\hat{\beta}_0$ and $\hat{\beta}_1$ for the coefficients β_0 and β_1 can be obtained.

Lily wants to use R to generate random samples based on the binary logistic regression model assumptions, in order to understand the "roughly" unbiased property for MLE, as well as "approximate" normality for the sampling distribution of MLE. She follows the steps below.

STEP 1: Specify $\beta_0 = 2$ and $\beta_1 = 1$.

STEP 2: Suppose the observations $X_1, \cdots, X_n$ are $0.001, 0.002, 0.003, \cdots, 1.000$, so the number of observations $n = 1000$.

STEP 3: Compute

$$\pi_i = \frac{e^{\beta_0+\beta_1 X_i}}{1 + e^{\beta_0+\beta_1 X_i}} \text{ for } i = 1, \cdots, n.$$

STEP 4: Generate Y_i independently from the Bernoulli distribution with $\text{P}(Y_i = 1|X_i) = \pi_i$ for $i = 1, \cdots, n$. (Hint: R function "rbinom(1,1,π_i)" returns one random number of Y_i from the Bernoulli distribution with $\text{P}(Y_i = 1|X_i) = \pi_i$.)

STEP 5: Repeat Step 4 1,000 times and obtain 1,000 different datasets of $\{Y_i, X_i\}_{i=1}^n$.

Lei Li is a friend of Lily. Lily hands over the above 1,000 datasets to him but she does not tell him the true values of β_0 and β_1. Based on each dataset, Lei Li computes the MLE $\hat{\beta}_0$ and $\hat{\beta}_1$. Ultimately, he obtains 1,000 different MLEs.

Then Lily tells Lei Li the true value of β_1 and both Lily and Lei Li compare this true value to the sample average of these 1,000 different MLEs $\hat{\beta}_1$, as well as the histogram of these 1,000 estimates $\hat{\beta}_1$.

Please answer the following questions in the answer sheet.

a) (0.5 points) Suppose you play both roles of Lily and Lei Li and realise the above steps in R. Please paste the complete R codes for all the above procedures in the answer sheet. (Hint: similar to the codes on page 7 of Lecture Notes 2.)

Solution:

```r
#Q4
rm(list=ls())
beta0=2
beta1=1
X=(1:1000)/1000
n=length(X)
Y=rep(0,n)
numsamp=1000
b0=rep(0,numsamp)
b1=rep(0,numsamp)
pi=exp(beta0+beta1*X)/(1+exp(beta0+beta1*X))
set.seed(1)
for(j in 1:numsamp) {
  for (i in 1:n){
    Y[i]=rbinom(1,1,pi[i])
  }
  fit=glm(Y~X,family=binomial(link=logit))
  b0[j]=fit$coef[1]
  b1[j]=fit$coef[2]
}
round(mean(b1),4)
hist(b1)
```

b) (0.5 points) What is the sample average of 1,000 estimates $\hat{\beta}_1$ (rounded to four decimal places). Is it close to the true value of β_1? Please answer this question in the answer sheet.

Solution: The sample average is

```r
round(mean(b1),4)
```

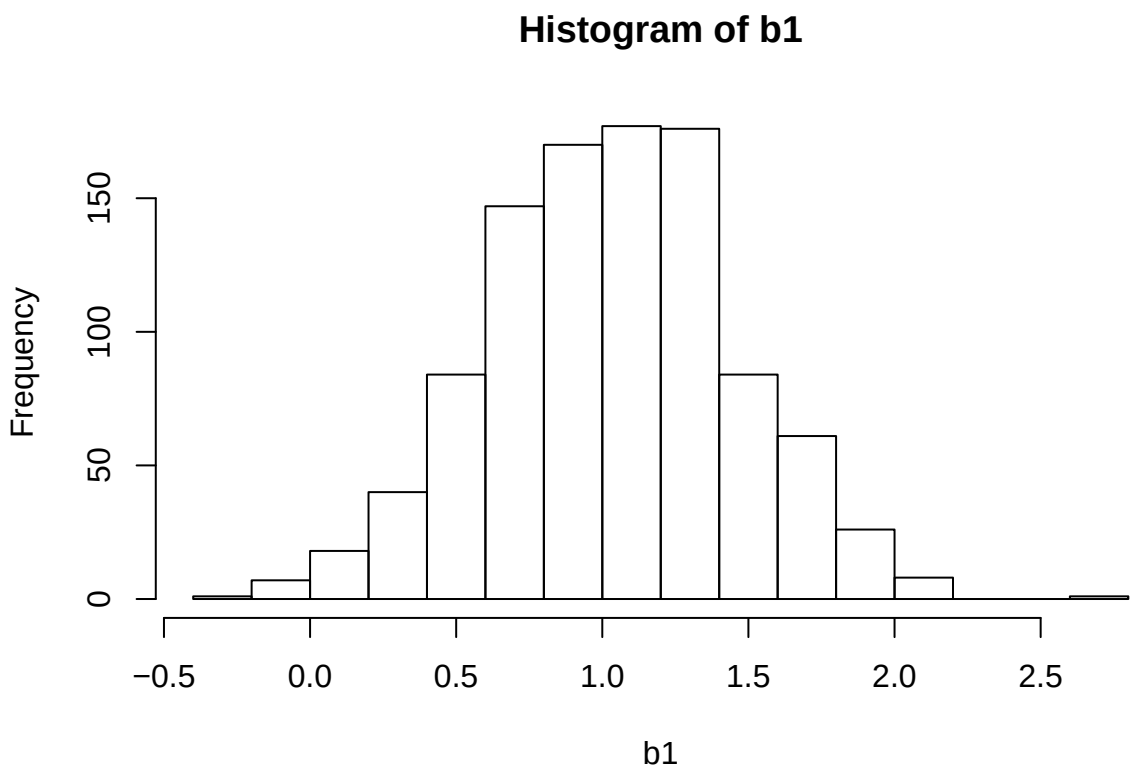
1.0330

This figure can be different by using different "set.seed()" but should be close to the true value $\beta_1 = 1$.

c) (0.5 points) Please paste the histogram plot of 1,000 estimates $\hat{\beta}_1$ in the answer sheet. Is it close to the normal distribution?

Solution:

```
hist(b1)
```



Yes, it is close to the normal distribution. (It is also OK to say no, since the random samples sometimes are generated not very well. The answer should be based on the histogram. If persuasive reasons are given to say no, it is totally fine.)