MAST90104: A First Course in Statistical Learning

Assignment 4, 2024 Solution

1. (15 marks) The data *winequality-red.csv* includes data from the paper *Modeling wine preferences by data mining from physicochemical properties* by P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis (2009). The data consists of 1599 observations of red variants of the Portuguese "Vinho Verd" wine. The variables are:

   - fixed acidity
   - volatile acidity
   - citric acid
   - residual sugar
   - chlorides
   - free sulfur dioxide
   - total sulfur dioxide
   - density
   - pH
   - sulphates
   - alcohol : percent alcohol content of the wine
   - quality : Output variable (score between 0 and 10)

   The quality score in this data set ranges from 3 to 8, but we will recode the levels as *bad*, *average* and *good*:

```
wine$quality = factor(wine$quality)
levels(wine$quality ) <- c("bad","bad","average","average" , "good","good")
```

   (a) Fit a multinomial logit model to predict the wine quality by category, considering all available predictors. Refine the model using stepwise selection with AIC as selection criteria. (3pt)
   Solution

```
wine <- read.csv("winequality-red.csv",header=TRUE, sep = ";")
wine$quality = factor(wine$quality)
levels(wine$quality ) <- c("bad","bad","average","average" , "good","good")
library(nnet)
model1 = multinom(quality ~ ., data = wine)
model2 = step(model1,trace = 0)
summary(model2)

# Call:
# multinom(formula = quality ~ fixed.acidity + volatile.acidity +
#              citric.acid + residual.sugar + chlorides + total.sulfur.dioxide +
#              density + pH + sulphates + alcohol, data = wine)
#
# Coefficients:
#         (Intercept) fixed.acidity volatile.acidity citric.acid residual.sugar  chlorides
# average   -177.1713   -0.26252634        -4.565347  -1.4550010    -0.24141265  -5.434441
# good       125.7384    0.07207673        -6.952064  -0.8516209     0.02213188 -13.805282
#
#         total.sulfur.dioxide   density        pH sulphates   alcohol
# average          0.020732275  196.0641 -4.486351  1.255191 0.4420856
# good             0.006790962 -123.6146 -3.799881  5.050087 1.1345209
#
# Std. Errors:
#         (Intercept) fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
# average    2.704340     0.1545768        0.8275218    1.201010     0.07916868  3.111854
# good       3.125144     0.1708142        1.1010330    1.427641     0.09566638  4.468744
#
#         total.sulfur.dioxide  density       pH sulphates   alcohol
# average          0.006059025 2.649079 1.393743  1.273329 0.1756397
# good             0.006894894 3.048687 1.597277  1.360258 0.1946937
```

```
#
# Residual Deviance: 1296.663
# AIC: 1340.663
```

Stepwise selection on the multinomial chose 10 variables (all except free sulfur dioxide)

(b) Repeat the analysis in part (a) with an ordinal model, with "good" being the highest level of the output. Comment on any differences. (4pt)

Solution

```
library(MASS)
model_o1= polr(quality ~ ., data = wine)
model_o2 <- step(model_o1,trace = 0)
summary(model_o2)

# Call:
# polr(formula = quality ~ volatile.acidity + chlorides + pH +
#          sulphates + alcohol, data = wine)
#
# Coefficients:
#                    Value Std. Error t value
# volatile.acidity -4.0170     0.4688  -8.569
# chlorides        -6.5630     1.8940  -3.465
# pH               -2.1087     0.5221  -4.039
# sulphates         2.4779     0.4547   5.450
# alcohol           0.8712     0.0758  11.494
#
# Intercepts:
#                Value   Std. Error t value
# poor|average -3.0044  1.7780    -1.6898
# average|good  3.6146  1.7748     2.0366
#
# Residual Deviance: 1377.604
# AIC: 1391.604
```

With the same set of predictors, ordinal model has less parameters than multinomial model. Stepwise selection on the ordinal model only picked 5 predictors.

(c) We have a new observation with these attributes:

```
newobs
# fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
#           7.9              0.4         0.2             1.7       0.1
# free.sulfur.dioxide total.sulfur.dioxide  density   pH  sulphates alcohol
#                  10                   36    0.997  3.3        0.9      10
```

Compute the probabilities that this wine variant is a "bad", "average" and "good" wine, according to the refined multinomial and ordinal models. You should NOT use the function `predict()` for this question. (5pt)

Solution

Let $j = 1, 2, 3$ correspond to the outcome "bad", "average" and "good" respectively.

In the multinomial model,

$$p_j^* = \frac{e^{\eta_j^*}}{\sum_{k=1}^{3} e^{\eta_k^*}}, \quad j = 1, 2, 3,$$

and $\eta_j^* = (\mathbf{x}^*)^T \boldsymbol{\beta}_j$; $\boldsymbol{\beta}_1 = 0$ so $\eta_1^* = 0$

```
xstar = c(1,7.9,0.4,0.2,1.7,0.1,36,0.997,3.3,0.9,10)
model2_coef_average = c(-177.1713, -0.26252634,-4.565347,-1.4550010, -0.24141265,
-5.434441 , 0.020732275, 196.0641, -4.486351,  1.255191, 0.4420856)
model2_coef_good = c( 125.7384, 0.07207673, -6.952064 , -0.8516209, 0.02213188,
-13.805282 ,0.006790962, -123.6146, -3.799881,  5.050087, 1.1345209)
eta = rep(0,3)
eta[2] = (xstar%*%model2_coef_average)
eta[3] = (xstar%*%model2_coef_good)
(probs_multinom = exp(eta)/sum(exp(eta)) )
# [1] 0.008591233 0.899949562 0.091459205
```

In the ordinal model, we have

$$\gamma_j^* = Pr(Y^* \leq j), \quad \log \frac{\gamma_j^*}{1 - \gamma_j^*} = \theta_j - (\mathbf{x}^*)^T \boldsymbol{\beta}.$$

So $p_1^* = \frac{1}{1 + e^{-\gamma_1^*}}$, $p_2^* = \frac{1}{1 + e^{-\gamma_2^*}} - \frac{1}{1 + e^{-\gamma_1^*}}$ and $p_3^* = 1 - p_1^* - p_2^* = 1 - \frac{1}{1 + e^{-\gamma_2^*}}$.

As can be seen below, there is some difference in predicted probabilities between the ordinal and multinomial models.

```
xstar_ord = c(0.4,0.1,3.3,0.9,10)
logitcumprob = rep(1,2)
logitcumprob[1] = (model_o2$zeta[1]  - xstar_ord%*%model_o2$coefficients)
logitcumprob[2] = (model_o2$zeta[2] - xstar_ord%*%model_o2$coefficients)
```

```
probs_ordinal = rep(NA,3)
probs_ordinal[1] = 1/(1+ exp(-logitcumprob[1]))
probs_ordinal[2] = 1/(1+ exp(-logitcumprob[2])) - 1/(1+ exp(-logitcumprob[1]))
probs_ordinal[3] = 1- 1/(1+ exp(-logitcumprob[2]))
probs_ordinal
# [1] 0.00879436 0.86042844 0.13077720
```

(d) Under the ordinal model, what is the odds ratio of being classified as "bad" or "average" of a wine variant with chlorides level 0.08 compared to a variant with chlorides level 0.2, given that the other attributes of the two variants are the same? (3pt)

Solution From week 10's workshop, we have the odds ratio is

$$\frac{Pr(Y \leq 2|\mathbf{x}_A)}{Pr(Y \leq 2|\mathbf{x}_B)} = \exp(-(\mathbf{x}_A - \mathbf{x}_B)^T \boldsymbol{\beta}),$$

where the elements of $\mathbf{x}_A$ and $\mathbf{x}_B$ are the same except for chlorides. So the odds ratio is $\exp(-(0.08 - 0.2)\beta_{\text{chlorides}})$

```
exp(-model_o2$coefficients[2]*(0.08-0.2))
# 0.4549514
```

2. (15 marks) *Use only content provided in this question to answer all parts:* The data `moons` contains the diameter, mass, distance from the sun, and number of moons for 13 planets, gas giants, and dwarf planets in our solar system. The variables are

   - `Name`: a character variable with the name of the planet, gas giant, or dwarf planet
   - `Distance`: distance from sun, relative to earth's
   - `Diameter`: diameter of the planet, relative to earth's
   - `Mass`: mass, relative to earth's
   - `Moons`: number of moons

   The first 3 rows of the data are shown below:

Name	Distance	Diameter	Mass	Moons
Mercury	0.39	0.382	0.0600	0
Venus	0.72	0.949	0.8200	0
Earth	1.00	1.000	1.0000	1

   We want to see if the number of moons of a planet is related to its size. We will use Poisson regression (with log link) to model the number of moons of a planet.

   (a) (6 marks) We first fit a model that assumes the number of moons does not depend on any other variable in the data (Model 0). Next, we fit a model for the number of moons with predictors `Diameter` and `Mass`. We call this Model 1. The following R output is provided.

```
model0 <- glm(Moons~1,family = poisson(link = 'log'),data = moons)
model1 <- glm(Moons~ Mass + Diameter,family = poisson(link = 'log'),data = moons)
summary(model1)

# Call:
# glm(formula = Moons ~ Mass + Diameter, family = poisson(link = "log"),
#       data = moons)
#
# Deviance Residuals:
#      Min        1Q    Median        3Q       Max
# -2.45936  -2.05425  -0.91278   0.27366   4.32665
#
# Coefficients:
#               Estimate  Std. Error z value  Pr(>|z|)
# (Intercept)  0.71322809  0.19912109  3.5819 0.0003411 ***
# Mass        -0.00401033  0.00094609 -4.2388 2.247e-05 ***
# Diameter     0.41803308  0.03270228 12.7830 < 2.2e-16 ***
```

```
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
#
# (Dispersion parameter for poisson family taken to be 1)
#
#     Null deviance: 388.2529  on 12  degrees of freedom
# Residual deviance:  44.8173  on 10  degrees of freedom
# AIC: 84.8331
> c(qchisq(0.9,10), qchisq(0.9,2), qnorm(0.95))
[1] 15.987179  4.605170  1.644854
```

Compared to Model 0, is Model 1 a better model for predicting the number of moons? Conduct an appropriate hypothesis test at 10% significance level and state clearly the test you use, the test statistic value, its null distribution, and the conclusion. *Hint: the Null deviance in the R output corresponds to the deviance of Model 0. The residual deviance corresponds to the deviance of Model 1.*

Solution: We can use the log-likelihood ratio test .

The test statistics is 388.2529 - 44.8173 = 343.4356, follow a χ^2 distribution with degree of freedom 2.

We reject H_0 if test statistic is larger than 5.991465.

Since 343.4356 is within the rejection region, we reject the null hypothesis that Model 0 is correct. Model 1 is better/ correct.

(b) (6 marks) We suspect that distance from sun is also useful in predicting the number of moons of a planet. We then add `Distance` to our model, and call this new model Model 2. The model's summary is shown below (with a `MISSING` output):

```
model2 <- glm(Moons~ Mass + Diameter + Distance, family = poisson(link = 'log'),
data = moons)
summary(model2)

# Call:
# glm(formula = Moons ~ Mass + Diameter + Distance, family = poisson(link = "log"),
#     data = moons)
#
# Deviance Residuals:
#      Min        1Q    Median        3Q       Max
# -2.34540  -1.73936  -0.87962   0.27045   4.40580
#
# Coefficients:
#               Estimate  Std. Error z value  Pr(>|z|)
# (Intercept)  0.31230542  0.33261090  0.9390   0.34776
# Mass        -0.00392976  0.00094607 -4.1538 3.271e-05 ***
# Diameter     0.44500885  0.03824737 11.6350 < 2.2e-16 ***
# Distance     0.01410618  0.00826520  1.7067   0.08788 .
# ---
# Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
#
# (Dispersion parameter for poisson family taken to be 1)
#
#     Null deviance: 388.2529  on 12  degrees of freedom
# Residual deviance:  41.8974  on  9  degrees of freedom
# AIC: <MISSING>
```

Compute the missing output, i.e., model 2's AIC.

Solution:

Since model 1 is nested within model 2, the difference in AICs is given by the formula:

$$AIC^2 - AIC^1 = 2s + D^2/\phi - D^1/\phi$$

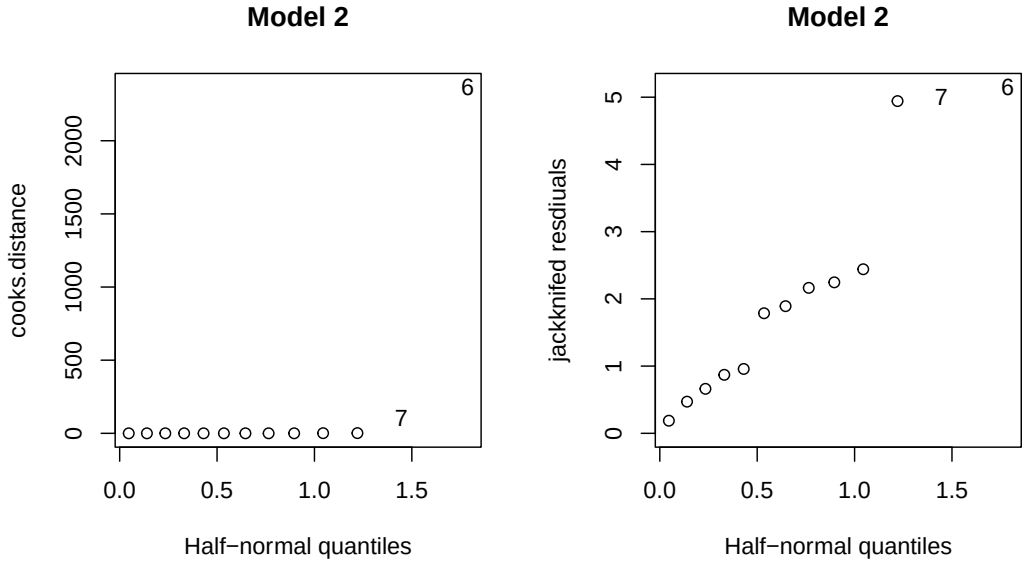
Since Model 2 has one extra parameter compared to Model 1, $s = 1$.

Since response is Poisson distributed, $\phi = 1$.

Also, $AIC^1 = 84.8331$, $D^2 = 41.8974$ and $D^1 = 44.8173$.

By plugging the numbers into the formula, we have $AIC^2 = 2 + 41.8974 - 44.8173 + 84.8331 = 83.9132$.

(c) (3 marks) The figure in the next page shows the half-normal plots of Cook's distance (left panel) and jackknife residuals (right panel) from Model 2. Can you identify any influential observations or outliers from the two plots?



Solution: The jackknife residuals of 6th and 7th observations are higher and deviate from the rest, suggesting that they might be outliers. This is confirmed by their large Cook's distance and hence they are indeed influential outliers