Data Science for Biomedical Informatics

Prediction in Multinomial Logistic Regression

Let us go back to the alligator data. The response variable is ‘food,’ which is ternary. There is only one covariate ‘size.’ We fit a multinomial logistic regression model to the data. Let us input the data.

> size <- c(1.24, 1.45, 1.63, 1.78, 1.98, 2.36, 2.79, 3.68, 1.30, 1.45, 1.65, 1.78, 2.03, 2.39, 2.84, 3.71, 1.30, 1.47, 1.65, 1.78, 2.03, 2.41, 3.25, 3.89, 1.32, 1.47, 1.65, 1.80, 2.16, 2.44, 3.28, 1.32, 1.50, 1.65, 1.80, 2.26, 2.46, 3.33, 1.40, 1.52, 1.68, 1.85, 2.31, 2.56, 3.56, 1.42, 1.55, 1.70, 1.88, 2.31, 2.67, 3.58, 1.42, 1.60, 1.73, 1.93, 2.36, 2.72, 3.66)

> length(size)

[1] 59

> food <- factor(c(rep("I", 5), rep("F", 2), "O", "I", rep("O", 2), "I",

rep("F", 4), rep("I", 3), "O", rep("F", 2), "O", rep("F", 4), "I", rep("F", 2),

"O", "F", "I", rep("F", 6), "I", rep("F", 3), "O", "F", rep("I", 4),

rep("F", 4), "I", "O", "I", "F", "I", "F"))

> head(food)

[1] I I I I I F

Levels: F I O

The levels are listed alphabetically. How many alligators in each food choice?

table(food)

food

F I O

31 20 8

Activate the ‘nnet’ package. (One could use the VGAM package too.) Fit the multinomial logistic regression model.

> MB <- multinom(food ~ size, data = Allig)

# weights: 9 (4 variable)

initial value 64.818125

iter 10 value 49.170710

final value 49.170622

converged

> summary(MB)

Call:

multinom(formula = food ~ size, data = Allig)

Coefficients:

(Intercept) size

I 4.079701 -2.3553303

O -1.617713 0.1101012

Std. Errors:

(Intercept) size

I 1.468640 0.8032870

O 1.307274 0.5170823

Comment: The response variable is ternary with levels F, I , O, listed alphabetically. One of them has to be the baseline. Alphabetically, F comes first. R (nnet) takes F to be the base line. Can we change the baseline? Yes.

Fitted model:

Change the baseline to “I.”

> food1 <- relevel(food, ref = "I")

> head(food1)

[1] I I I I I F

Levels: I F O

Let us fit the multinomial model with “I” as the baseline.

> MB1 <- multinom(food1 ~ size)

# weights: 9 (4 variable)

initial value 64.818125

iter 10 value 49.170622

iter 10 value 49.170622

final value 49.170622

converged

> summary(MB1)

Call:

multinom(formula = food1 ~ size)

Coefficients:

(Intercept) size

F -4.079750 2.355354

O -5.697419 2.465437

Std. Errors:

(Intercept) size

F 1.468647 0.8032912

O 1.793808 0.8996518

Residual Deviance: 98.34124

AIC: 106.3412

Residual deviance hasn’t changed. AIC hasn’t changed. The output does not give degrees of freedom. The prediction model is:

How good this model is?

There are two ways to judge it.

1. We have 59 alligators in the data. For each alligator in the data, feed its size into the estimated model to get three probabilities of F, I, and 0. Identify the dominant probability. Classify the alligator’s food type the corresponding one. We know the truth. We know its food type. Check whether the predicted food type matches with the observed one. Count the number of misclassifications. Calculate the misclassification rate. Use the ‘predict’ command.
2. Do a goodness-of-fit test.

Let us follow the first option. Using the dominant probability, the ‘predict’ command predicts the food choice of each alligator in the data.

We use the output folder MB in which “F” is the baseline.

MB2 <- predict(MB, newdata = food)

> MB2

[1] I I I F F F F F I I I F F F F F I I I F F F F F I I I F F F F I I I F F F F

[39] I I I F F F F I I I F F F F I I I F F F F

Levels: F I O

We could use MB1 in which “I” is the baseline.

> MB3 <- predict(MB1, newdata = food1)

> MB3

[1] I I I F F F F F I I I F F F F F I I I F F F F F I I I F F F F I I I F F F F

[39] I I I F F F F I I I F F F F I I I F F F F

Levels: I F O

Predictions are identical. Cross-tabulate MB2 and MB3.

> table(MB2, MB3)

MB3

MB2 I F O

F 0 35 0

I 24 0 0

O 0 0 0

> names(MB)

[1] "n" "nunits" "nconn" "conn"

[5] "nsunits" "decay" "entropy" "softmax"

[9] "censored" "value" "wts" "convergence"

[13] "fitted.values" "residuals" "lev" "call"

[17] "terms" "weights" "deviance" "rank"

[21] "lab" "coefnames" "vcoefnames" "xlevels"

[25] "edf" "AIC"

‘fitted.values’ actually gives the probabilities.

> head(round(MB$fitted.values, 3))

F I O

1 0.227 0.722 0.052

2 0.315 0.612 0.073

3 0.399 0.507 0.095

4 0.468 0.418 0.113

5 0.554 0.309 0.137

6 0.673 0.153 0.173

We could use the output folder MB1 too. The probabilities are identical. Interpretation of the probabilities …

> MB4 <- round(MB$fitted.values, 3)

> head(MB4)

F I O

1 0.227 0.722 0.052

2 0.315 0.612 0.073

3 0.399 0.507 0.095

4 0.468 0.418 0.113

5 0.554 0.309 0.137

6 0.673 0.153 0.173

The command ‘predict’ gives predicted food choice based on the dominant probability. We know what actually the food choice is. We can compare the observed and predicted.

> MB5 <- data.frame(ObservedFood = food, PredictedFood = MB2)

> head(MB5)

ObservedFood PredictedFood

1 I I

2 I I

3 I I

4 I F

5 I F

6 F F

Let us cross-tabulate these two columns to find out how good or bad the model is.

> ConfusionMatrix <- table(MB5$ObservedFood, MB5$PredictedFood)

> ConfusionMatrix

F I O

F 23 8 0

I 7 13 0

O 5 3 0

Interpretation:

1. There are 31 alligators whose food choice is Fish. Twenty-three of them are correctly classified by the model.

2. There 20 alligators whose food choice is Invertebrates. Thirteen of them are correctly classified.

3. There are 8 alligators whose food choice is other. They all are misclassified.

Misclassification rate: 23/59 = 39%

Discussion …

Individual predictions as per the model can also be carried out. I have an alligator with size = 3.5 meters. Predict its food choice.

> predict(MB, newdata = data.frame(size = 3.5))

[1] F

Levels: 0 F I

Get the predicted probabilities as per the model of this alligator.

> predict(MB, newdata = data.frame(size = 3.5), type = "prob")

0 F I

0.22309217 0.76501441 0.01189342

There is another package, VGAM, which can fit a multinomial logistic regression model. This model is more versatile than the nnet package. Let us download and activate the package.

MB6 <- vglm(food ~ size, family = multinomial)

> summary(MB6)

Call:

vglm(formula = food ~ size, family = multinomial)

Pearson residuals:

Min 1Q Median 3Q Max

log(mu[,1]/mu[,3]) -2.330 -0.5075 0.5538 0.6836 1.452

log(mu[,2]/mu[,3]) -2.687 -0.4821 -0.1653 0.7093 3.439

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept):1 1.6177 1.3073 1.237 0.21591

(Intercept):2 5.6974 1.7937 3.176 0.00149 \*\*

size:1 -0.1101 0.5171 -0.213 0.83137

size:2 -2.4654 0.8996 -2.741 0.00613 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Number of linear predictors: 2

Names of linear predictors: log(mu[,1]/mu[,3]), log(mu[,2]/mu[,3])

Dispersion Parameter for multinomial family: 1

Residual deviance: 98.3412 on 114 degrees of freedom

Log-likelihood: -49.1706 on 114 degrees of freedom

Number of iterations: 5

Residual Deviance: 98.34124 on 114 degrees of freedom

Log-likelihood: -49.17062 on 114 degrees of freedom

Number of Iterations: 5

What is the baseline category of the model fitted?

D = 1 + exp(5.70 - 2.47\*size) + exp(1.62 - 0.11\*size)

VGAM chose ‘O’ to be the baseline. See the difference between the nnet and VGAM packages. VGAM gives degrees of freedom for the residual deviance.

Goodness-of-fit test

Look at the residual deviance and degrees of freedom.

Degrees of freedom = Total number of observations – Number of parameters = 118 – 4 = 114.

For each alligator in the data, we want to estimate its distribution of food choices: F, I, and O. It is enough to estimate two of the three probabilities. The third probability can be obtained by subtraction. Suppose you don’t believe in modeling. You are left to your own devices to estimate the probability distribution of food choices for each alligator. You have 59\*2 = 118 degrees of freedom at your disposal. Suppose you believe in model building and multinomial logistic regression model. All you need to do is estimate the four parameters of the model. The model then estimates food choice distribution for each alligator. The number of degrees of freedom is 118 – 4 = 114.

Null hypotheses: H0: The response probabilities have the multinomial logistic regression model pattern for some choice of the parameters βs.

Calculate the p-value under the null hypothesis. Theoretically, the residual deviance has a chi-squared distribution with the stipulated degrees of freedom when the null hypothesis is true, i.e., when the model is multinomial logistic regression type.

p-value = What are the chances of observing the residual deviance at least as large as the one we have observed when null hypothesis is true = Pr(≥ 98.34124 | H0). Use R.

> pchisq(98.34124, 114, lower.tail = F)

[1] 0.8517622

The chances are very high. The threshold is 5%. Do not reject the null hypothesis. The fit is good.

Discussion …

1. Model fit as vouchsafed by goodness-of-fit is very good. Prediction is so-so.
2. One reason could be that there are not many alligators with food choice ‘O.’ The mis-classification rate was jacked up by these alligators.
3. Focus on alligators with food choice either ‘F’ or ‘I.’ We will have 51 observations. Fit a logistic regression model. Check on goodness-of-fit and prediction. Homework?