Lecture 5 Handout

Elizabeth Colantuoni

4/6/2021

I. Objectives

Upon completion of this session, you will be able to do the following:

- Use the asymptotic normality of the mle's in the GLM family to conduct estimation and inference on linear combinations of regression coefficients
- Understand and explain the likelihood ratio test and Wald tests for the GLM family
- Describe how to use the fit of a logistic regression model to make predictions / classify individual outcomes
- Understand and explain the receiver-operator characteristic (ROC) curve
- Use cross-validation to obtain valid assessments of prediction error for logistic regression-based classification

II. One last bit on MLE for logistic regression

In Lecture 3, we derived the solution for the maximum likelihood estimates of β in a logistic regression model. We will review that briefly here and link the Newton-raphson algorithm to weighted least squares for logistic regression models.

First,

- Define the $(p+1) \times 1$ vector of covariates for subject i as $x_i = (1, x_{1i}, x_{2i}, ..., x_{pi})$.
- Define the $(p+1) \times 1$ vector of association parameters $\beta = (\beta_0, \beta_1, ..., \beta_p)$.

Then we assume the following model:

- $Y_i \sim Bernoulli(\mu_i)$ for $i = 1, ..., n$ independent observations.
- Define the vector of covariates for subject i as $x_i = (1, x_{1i}, x_{2i}, ..., x_{pi})$.
- Define the vector of association parameters $\beta = (\beta_0, \beta_1, ..., \beta_p)$.
- Assume the logit link such that:

$$log\left(\frac{\mu_i}{1-\mu_i}\right) = x_i^{'}\beta \rightarrow \mu_i = \frac{e^{x_i^{'}\beta}}{1+e^{x_i^{'}\beta}}$$

A. Likelihood, log likelihood, score equations

We can express the likelihood function as:

$$\begin{aligned} L(\beta|y) &= Pr(Y_1 = y_1, Y_2 = y_2, ..., Y_n = y_n|\beta) \\ &= \prod_{i=1}^{n} Pr(Y_i = y_i|\beta) \\ &= \prod_{i=1}^{n} \mu_i(\beta)^{y_i}[1-\mu_i(\beta)]^{1-y_i} \end{aligned}$$

The log-likelihood function is:

$$log[L(\beta|y)] = \sum_{i=1}^{n} y_i log[\mu_i(\beta)] + (1-y_i) log[1-\mu_i(\beta)]$$

The score equation is:

$$\begin{aligned} U(\beta) &= \frac{\partial log[L(\beta|y)]}{\partial \beta} \\ &= \sum_{i=1}^{n} x_i(y_i - \mu_i(\beta)) \\ &= X^{'}(Y - \mu(\beta)) \end{aligned}$$

B. Iteratively Reweighted Least Squares

We solve for β using Iteratively Reweighted Least Squares (IRLS), where

$$\hat{\beta}^{(k+1)} = (X^{'}V^{(k)}X)^{-1}(X^{'}V^{(k)}Z^{(k)})$$

where

$$V^{(k)} = diag(\mu_i(\beta^{(k)})[1-\mu_i(\beta^{(k)})])$$

$$Z^{(k)} = X\hat{\beta}^{(k)} + V^{-1(k)}(Y - \mu(\hat{\beta}^{(k)}))$$

C. Comparison to weighted least squares

Compare the IRLS to the weighted least squares solution we derived last term:

$$\hat{\beta}_{WLS} = \left(X^{'}\hat{V}^{-1}X\right)^{-1}\left(X^{'}\hat{V}^{-1}Y\right)$$

These are different! $\hat{V}$ vs. $\hat{V}^{-1}$.

Why??

Recall that we derived: $\frac{\partial \mu(\beta))}{\partial \beta} = VX = diag\left[\mu(\beta)(1-\mu(\beta))\right]X$

So that,

$$\begin{aligned}\hat{\beta}^{(k+1)} &= (X^{'}V^{(k)}X)^{-1}(X^{'}V^{(k)}Z^{(k)}) \\ &= \left(\frac{\partial\hat{\mu}(\beta^{(k)})}{\partial\beta}^{'}\hat{V}^{(k)-1}\frac{\partial\hat{\mu}(\beta^{(k)})}{\partial\beta}\right)^{-1}\left(\frac{\partial\hat{\mu}(\beta^{(k)})}{\partial\beta}^{'}\hat{V}^{(k)-1}Z^{*(k)}\right)\end{aligned}$$

where $Z^{*(k)} = \frac{\partial\hat{\mu}(\beta^{(k)})}{\partial\beta}\hat{\beta}^{(k)} + (Y - \mu(\hat{\beta}^{(k)}))$.

III. Inference using $\hat{\beta}_{mle}$

Using similar arguments as we did for determining the distribution of $\hat{\beta}_{mle}$ in linear models, you can show that (we won't prove this in class):

$$\hat{\beta}_{mle} \approx N(\beta, \left[X^{'}VX\right]^{-1})$$

So from this we can derive the following set of tests or estimation relating to β.

A. Inference for β_j

Test $H_0 : \beta_j = b$ via $Z = \frac{\hat{\beta}_j - b}{\sqrt{[X^{'}VX]^{-1}_{jj}}}$

Confidence intervals can be derived as: $\hat{\beta}_j \pm 1.96\sqrt{[X^{'}VX]^{-1}_{jj}}$

B. Estimating linear combinations of β

Define $d = w^{'}\beta$ where w is a $(p+1)\times 1$ vector of scalars to create the relevant linear combination of β.

Estimate d via $w^{'}\hat{\beta}$ and $se(\hat{d}) = \sqrt{w^{'}\left[X^{'}VX\right]^{-1}w}$

Confidence interval for d: $\hat{d} \pm 1.96se_{\hat{d}}$.

Test $H_0 : d = \delta$ via $Z = \frac{\hat{d}-\delta}{se_{\hat{d}}}$.

C. Nested models

Here we assume we have a model with $\beta = (\beta_0, \beta_1, ..., \beta_p, \beta_{p+1}, ..., \beta_{p+s})$ and define $\beta^{+} = (\beta_{p+1}, .., \beta_{p+s})$.

1. Wald Test

To conduct a Wald test of $H_0: all $\beta_{p+j} = 0, for j = 1, ..., s$,

$$W = \hat{\beta}^{+'}\left[(X^{'}VX)^{-1}_{(+,+)}\right]^{-1}\hat{\beta}^{+} \approx \sum_{j=1}^{s} Z_j^2 \sim \chi^2_s$$

reject H_0 if $W > \chi^2_{s,1-0.05/2}$.

2. Likelihood ratio test

Alternatively, use a likelihood ratio test!

When the null hypothesis is true and sample size is large enough:

$$\Delta = -2\left[logLike_N(y,\hat{\beta}_N) - logLike_E(y,\hat{\beta}_E)\right] \sim \chi^2_s$$

Δ represents the "change in deviance" where

$$deviance = -2\left[logLike_N(y,\hat{\beta}_N) - logLike_E(y,y)\right] \sim \chi^2_s$$

where $logLike_E(y,y)$ is the biggest possible value.

The deviance is a measure of fidelity of the model to the data, like the residual sum of squares for linear regression.

D. Example: NMES big expenditure - MSCD relationship

```
load('./nmes.rdata')
data = nmes
data[data=='.'] = NA

## Create the necessary variables:
data$posexp=ifelse(data$totalexp>0,1,0)
data$mscd=ifelse(data$lc5+data$chd5>0,1,0)
data1=data[!is.na(data$eversmk),]
data1$bigexp=ifelse(data1$totalexp>1000,1,0)
data1$agec = data1$lastage - 60
data1$agesp1 = ifelse(data1$lastage>65,data1$lastage-65,0)
data1$agesp2 = ifelse(data1$lastage>80,data1$lastage-80,0)

fit0 = glm(bigexp~mscd+agec+agesp1+agesp2,data=data1,family="binomial")
fit1 = glm(bigexp~mscd*(agec+agesp1+agesp2),data=data1,family="binomial")
```

Write out the two models we are fitting:

1. Testing a single coefficient

In Model0, test the null hypothesis that after adjusting for age, there is no relationship between a big expenditure and a MSCD.

State the null and alternative hypotheses:

```
## In Model 0: Test \beta_{mscd} = 0
summary(fit0)$coefficients
```

```
##                 Estimate  Std. Error     z value      Pr(>|z|)
## (Intercept) -0.716235408 0.030036992 -23.8451109 1.138097e-125
## mscd         1.603178804 0.068286173  23.4773561 6.949175e-122
## agec         0.028079056 0.002891139   9.7121075  2.677428e-22
## agesp1      -0.005830743 0.007465457  -0.7810296  4.347851e-01
## agesp2      -0.002128496 0.019276490  -0.1104193  9.120769e-01
```

2. Computing a linear combination of β

Using Model1, estimate the log odds ratio of a big expenditure comparing persons with and without a MSCD whom are 70 years old.

What is the appropriate linear combination of β?

```
## In Model 1: Compute the OR for big expenditure vs. mscd for 70 year olds
w = c(0,1,0,0,0,10,5,0)
var.cov = summary(fit1)$cov.scaled
beta = fit1$coefficients
# estimate
t(w) %*% beta
```

```
##          [,1]
## [1,] 1.513507
```

```
# standard error
t(w) %*% var.cov %*% w
```

```
##              [,1]
## [1,] 0.006824451
```

```
# test statistic
t(w) %*% beta / sqrt(t(w) %*% var.cov %*% w)
```

```
##          [,1]
## [1,] 18.32106
```

```
# Square test statistic ~ chi-square 1
(t(w) %*% beta / sqrt(t(w) %*% var.cov %*% w))^2
```

```
##          [,1]
## [1,] 335.6613
```

```
## Confirm using lincom command
lincom(fit1,c("mscd+10*mscd:agec+5*mscd:agesp1"))
```

```
##                                Estimate    2.5 %  97.5 %    Chisq   Pr(>Chisq)
## mscd+10*mscd:agec+5*mscd:agesp1 1.513507 1.351594 1.67542 335.6613 5.620428e-75
```

3. Comparing nested models

Compare Model0 and Model1 using both a Wald test and a likelihood ratio test.

What null and alternative hypotheses are you testing?

```
## Nested model: Wald test for interaction
index = 6:8
# Compute the wald test
w = t(fit1$coeff[index]) %*% solve(var.cov[index,index]) %*% fit1$coeff[index]
w
```

```
##          [,1]
## [1,] 14.53997
```

```
pchisq(w,lower.tail=FALSE,df=3)
```

```
##             [,1]
## [1,] 0.002255128
```

```
## Nested model: likelihood ratio test
lrtest(fit1,fit0)
```

```
## Likelihood ratio test
##
## Model 1: bigexp ~ mscd * (agec + agesp1 + agesp2)
## Model 2: bigexp ~ mscd + agec + agesp1 + agesp2
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   8 -7126.9
## 2   5 -7134.5 -3 15.185   0.001665 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

IV. Classifying persons using logistic regression models

Predictions from logistic regression models or other methods for analysis of binary responses (classification trees, random forests, etc. More to come on these next) can be used to classify individuals. We may be interested in identifying

- persons at high risk vs low risk of large expenditures
- persons with HIV infection within a community clinic
- persons at high risk of needing post acute care after a hospitalization

In fact, all medical diagnoses are applications of classification.

A. Basic idea

Based on data from a training dataset, fit a logistic regression model using observed binary responses and potential predictors/exposures X. Use the resulting predicted values to classify subsequent individuals as being positive or negative for the outcome.

B. Notation and definitions

- Data: $(Y_1, X_1), ..., (Y_n, X_n)$ where X_i is a $(p+1) \times 1$ vector of exposures/predictors.
- Model: $logit[Pr(Y_i = 1|X_i)] = X_i^{'}\beta$
- Fit the Model: $\hat{\beta} \rightarrow \hat{\mu}_i = \frac{exp(X_i^{'}\hat{\beta})}{1+exp(X_i^{'}\hat{\beta})}$
- Define a classification rule: $d_i(\hat{\mu}_i, c) = 1$ if $\hat{\mu} > c, 0$ if $\hat{\mu} \leq c$
- For a given classification rule c, you can compute:

$$\hat{sens}(c) = Pr(d_i(\hat{\mu}_i, c) = 1|Y_i = 1) = Pr(\hat{\mu}_i > c|Y_i = 1)$$

$$\hat{spec}(c) = Pr(d_i(\hat{\mu}_i, c) = 0|Y_i = 0) = Pr(\hat{\mu}_i \leq c|Y_i = 0)$$

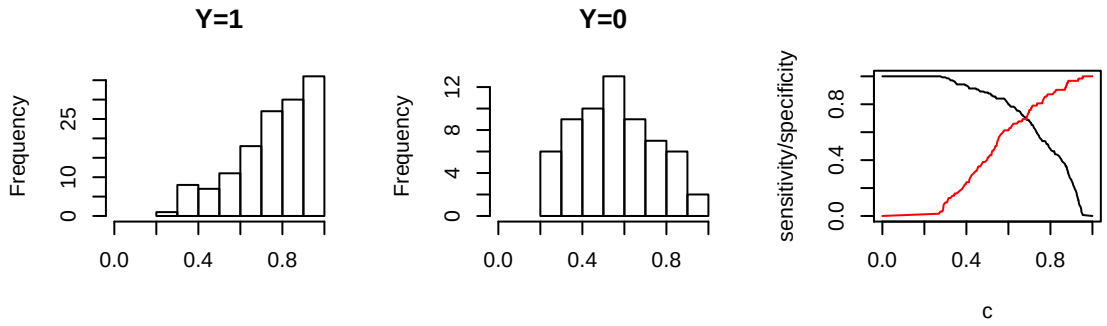
C. Goals and evaluation of a classifier

The goal of the classification problem is to build a logistic regression model / classifier that has sensitivity and specificity as close to 1, as nature will allow!

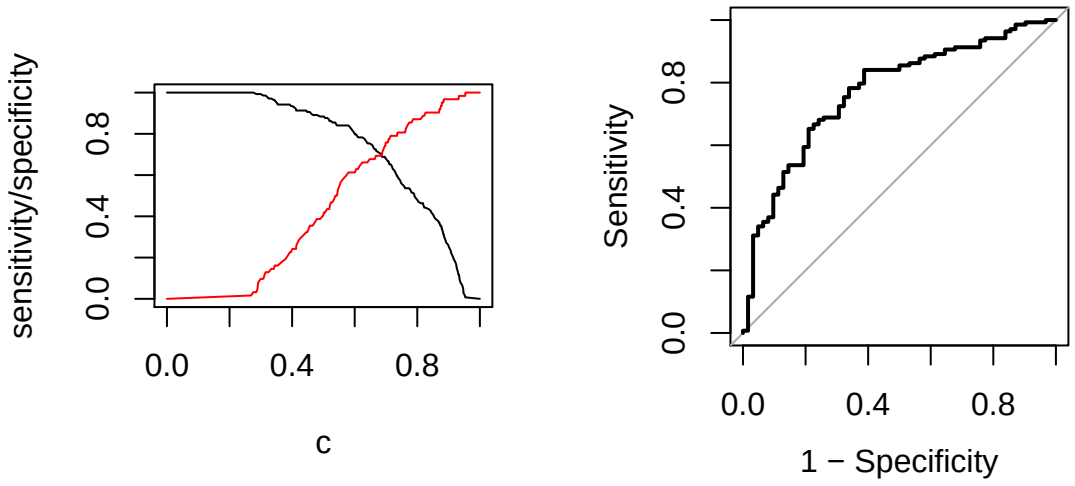
To evaluate the classifier, you can plot:

- $\hat{sens}(c)$ and $\hat{spec}(c)$ as a function of c, $0 \leq c \leq 1$.

```
par(mfrow=c(1,3))
hist(fit$fitted.values[y==1],xlim=c(0,1),main="Y=1",xlab=" ")
hist(fit$fitted.values[y==0],xlim=c(0,1),main="Y=0",xlab=" ")
x = roc_fit$thresholds
x[1] = 0
x[length(x)] = 1
plot(x,roc_fit$sensitivities,
     type="l",xlim=c(0,1),ylim=c(0,1),xlab="c",ylab="sensitivity/specificity")
points(x,roc_fit$specificities,
       type="l",col="red")
```

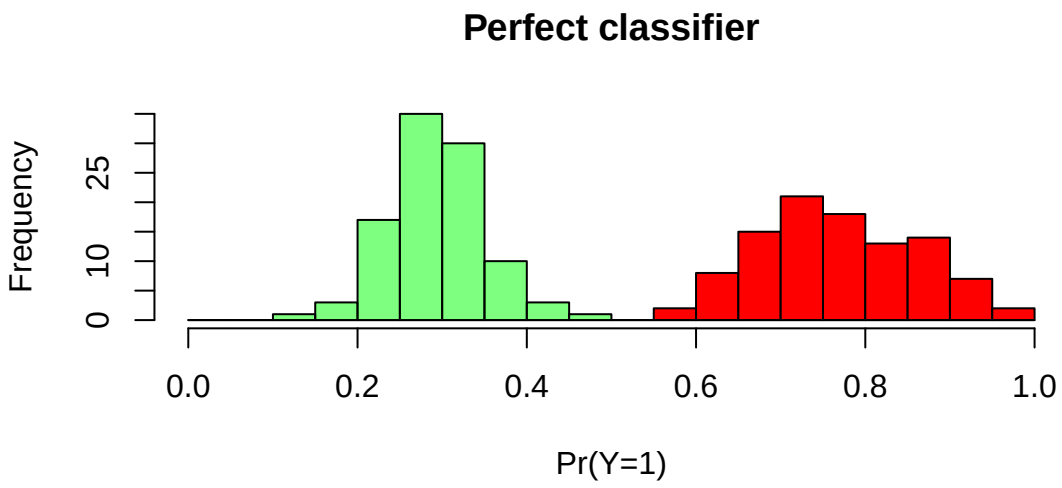


- $\hat{sens}(c)$ vs. $1 - \hat{spec}(c) \rightarrow$ receiver operating characteristic (ROC) curve



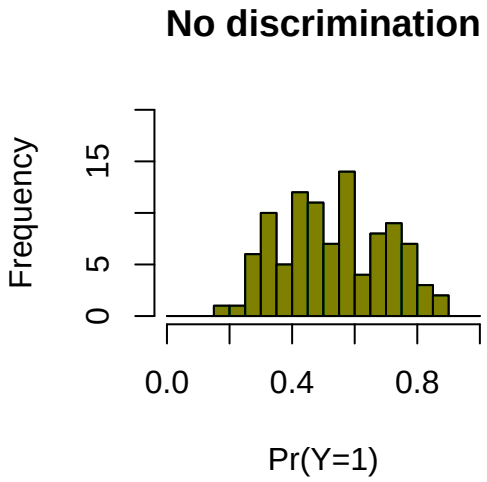
In addition, the area under the ROC curve, referred to as the "AUC" is a measure of how well the logistic regression model discriminates between the "cases" ($Y = 1$) and "controls" ($Y = 0$). In the simple example, the auc is 0.77.

1. Perfect classifier: AUC is 1!



- At $c = 1$, what is $\hat{sens}(c)$ and $\hat{spec}(c)$ and $1 - \hat{spec}(c)$?
- Move from $c = 1$ down to $c = 0.55$, specificity stays the same, but sensitivity increases from 0 to 1.
- Move from $c = 0.55$ to $c = 0$, sensitivity stays the same at 1, specificity moves from 1 to 0 (1-specificity from 0 to 1).
- You draw the picture of the ROC! What is the AUC?

2. Useless classifier: The model does not distinguish between cases and controls. AUC = 0.5



- Regardless of the value of c, sensitivity = 1 - specificity, AUC = 0.5

D. What does the AUC represent?

The area under the ROC curve is $Pr(\hat{\mu}_{(y=1)} > \hat{\mu}_{(y=0)})$, that is, the probability that a randomly selected "case" (y = 1) has a predicted probability that is greater than a randomly selected "control" (y = 0).

You prove this for extra enjoyment!

E. Cross-validation

To avoid being overly optimistic, the ROC and AUC should be constructed based on a cross-validation procedure.