JOHNS HOPKINS
BLOOMBERG SCHOOL
of PUBLIC HEALTH

Lecture 5

Review MLE and inference in logistic regression models
Prediction/classification using logistic regression models

The material in this video is subject to the copyright of the owners of the material and is being provided for educational purposes under rules of fair use for registered students in this course only. No additional copies of the copyrighted work may be made or distributed.

Lecture 4 Review

- We spent all of the time in the 10:30am session working on the confounding analysis.

- We will start with review of MLE and then delve into inference.

MLE in logistic models

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:

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

NOTE: We should really write $\mu_i(x_i, \beta)$ i.e. μ_i is a function of x_i and β. In this handout, I will simplify this to $\mu_i(\beta)$.

MLE in logistic models

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:

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

MLE in logistic models

The score equation, $U(\beta)$ is the derivative of the log-likelihood function with respect to β.

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

MLE in logistic models

NOTE: We will also need to know $U'(\beta) = \frac{\partial U(\beta)}{\partial \beta}$

$$
\begin{aligned}
U'(\beta) &= \frac{\partial U(\beta)}{\partial \beta} \\
&= \frac{\partial}{\partial \beta} X'(Y - \mu(\beta)) \\
&= -X' \frac{\partial \mu_i(\beta)}{\partial \beta} \\
&= -X'VX
\end{aligned}
$$

where we already showed that:

$$\frac{\partial \mu_i(\beta)}{\partial \beta} = \mu_i(\beta) \frac{\partial log[\mu_i(\beta)]}{\partial \beta} = \mu_i(\beta)(1 - \mu_i(\beta))x_i$$

and $V_{n \times n} = diag(\mu_i(\beta)[1 - \mu_i(\beta)])$.

Newton-Raphson Method to find "beta"

- Step 0: Pick an initial starting value for β, call this $\hat{\beta}^{(k)}$.
- Step 1: Compute the slope of $U(\beta)$ at $\hat{\beta}^{(k)}$, i.e. compute $U'(\hat{\beta}^{(k)})$.
- Step 2: Construct the tangent line, which is a line that passes through the points $(\hat{\beta}^{(k)}, U(\hat{\beta}^{(k)}))$ and $(\hat{\beta}^{(k+1)}, 0)$ and has slope $U'(\hat{\beta}^{(k)})$.
- Step 3: Solve the following for $\hat{\beta}^{(k+1)}$:

$$
\begin{aligned}
U'(\hat{\beta}^{(k)}) &= \frac{U(\hat{\beta}^{(k)})-0}{\hat{\beta}^{(k)}-\hat{\beta}^{(k+1)}} \\
[\hat{\beta}^{(k)} - \hat{\beta}^{(k+1)}]U'(\hat{\beta}^{(k)}) &= U(\hat{\beta}^{(k)}) \\
\hat{\beta}^{(k)} - \hat{\beta}^{(k+1)} &= U'(\hat{\beta}^{(k)})^{-1}U(\hat{\beta}^{(k)}) \\
\hat{\beta}^{(k+1)} &= \hat{\beta}^{(k)} - U'(\hat{\beta}^{(k)})^{-1}U(\hat{\beta}^{(k)}) \\
&= U'(\hat{\beta}^{(k)})^{-1}\left(U'(\hat{\beta}^{(k)})\hat{\beta}^{(k)} - U(\hat{\beta}^{(k)})\right)
\end{aligned}
$$

- Step 4: Stop if $|\hat{\beta}^{(k+1)} - \hat{\beta}^{(k)}|$ is small. If not, let $k = k + 1$ and repeat Steps 2 through 4.

Newton-Raphson Method to find "beta"

- In general, when "beta" is a vector:

$$
\begin{aligned}
\hat{\beta}^{(k+1)} &= U'(\hat{\beta}^{(k)})^{-1}\left(U'(\hat{\beta}^{(k)})\hat{\beta}^{(k)} - U(\hat{\beta}^{(k)})\right) \\
&= -(X'V^{(k)}X)^{-1}\left[-(X'V^{(k)}X)\hat{\beta}^{(k)} - X'(Y-\mu(\hat{\beta}^{(k)}))\right] \\
&= (X'V^{(k)}X)^{-1}\left[X'V^{(k)}\left(X\hat{\beta}^{(k)} + V^{-1(k)}(Y-\mu(\hat{\beta}^{(k)}))\right)\right] \\
&= (X'V^{(k)}X)^{-1}(X'V^{(k)}Z^{(k)})
\end{aligned}
$$

where

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

$$Z^{(k)} = X\hat{\beta}^{(k)} + V^{-1(k)}\left(Y - \mu(\hat{\beta}^{(k)})\right) = \text{a surrogate response.}$$

Iteratively Re-weighted Least Squares

The general procedure is:

- Step 0: Set an initial value for $\hat{\beta}^{(k)}$, $k = 0$.
- Step 1: Calculate: $V^{(k)}$, $\hat{\mu}(\hat{\beta}^{(k)})$, $Z^{(k)}$.
- Step 2: Update $\hat{\beta}^{(k+1)} = (X'V^{(k)}X)^{-1}(X'V^{(k)}Z^{(k)})$
- Step 3: Stop if $\sum_{j=1}^{p+1}\left(\hat{\beta}_j^{(k+1)} - \hat{\beta}_j^{(k)}\right)^2 < \epsilon$; if not, let $k = k + 1$ and repeat Steps 2 and 3.

IRLS vs 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. $\hat{V}^{-1}$.

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

Inference in logistic regression models

- Using similar arguments as we did for linear models: $\hat{\beta}_{mle} \approx N(\beta, [X'VX]^{-1})$

- Inference for a single coefficient:

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

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

- Inference for a linear combination of coefficients:

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 $se(\hat{d}) = \sqrt{w'[X'VX]^{-1}w}$

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

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

Inference in logistic regression models: 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 $\beta^+ = (\beta_{p+1}, .., \beta_{p+s})$.

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

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

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

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

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

Δ represents the "change in deviance" where

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

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.

Examples

```
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 model you are fitting in “fit0” and “fit1”.

Example: Testing a single coefficient

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

```r
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
```

Example: Linear combination of coefficients

- 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?

```
## 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
```

Example: Linear combination of coefficients

```r
## 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
```

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

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

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

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

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

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

Example: Nested models

- Model0 is nested within Model1.

- What null and alternative hypothesis are you testing if you compare Model1 and Model 0?

- Wald test

```
## 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
```

Example: Nested models

- Likelihood ratio test

```r
## 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
```

Logistic regression models as classifiers!

- Models for binary responses can be used to classify individuals
  - Logistic regression models
  - Classification and regression trees
  - Random forests

- May be interested in identifying
  - Persons at high risk for a big expenditure
  - Persons from a community clinic who are infected with HIV
  - Patients at high risk for requiring post acute care placement

- Diagnosis of disease or screening for procedures is classification!

Notation and definitions

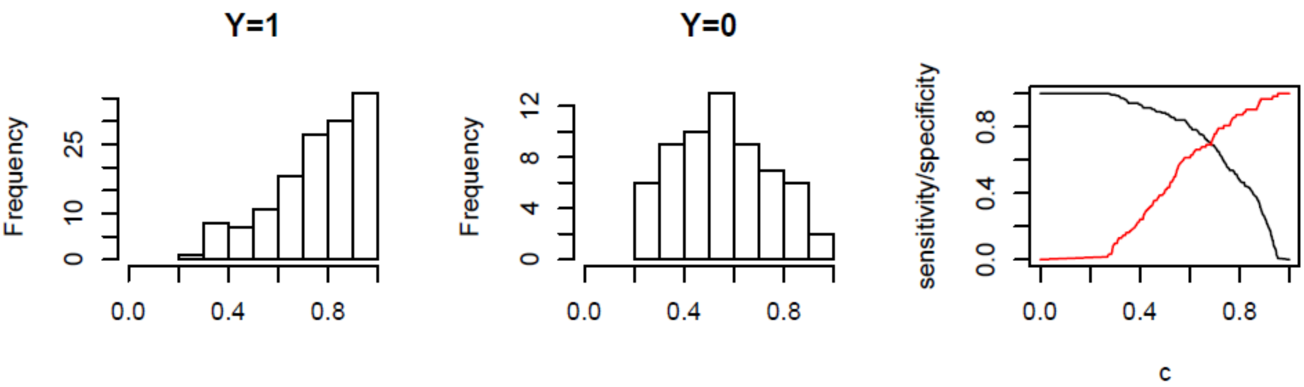
- Data: $(Y_1, X_1), ..., (Y_n, X_n)$ where X_i is a $(p+1) \times 1$ vector of exposures/predictors.
- Model: $logit[Pr(Y_i = 1|X_i)] = X_i^{'}\beta$
- Fit the Model: $\hat{\beta} \rightarrow \hat{\mu}_i = \frac{exp(X_i^{'}\hat{\beta})}{1+exp(X_i^{'}\hat{\beta})}$

- Define a classification rule:

- Define sensitivity and specificity based on the classification rule:

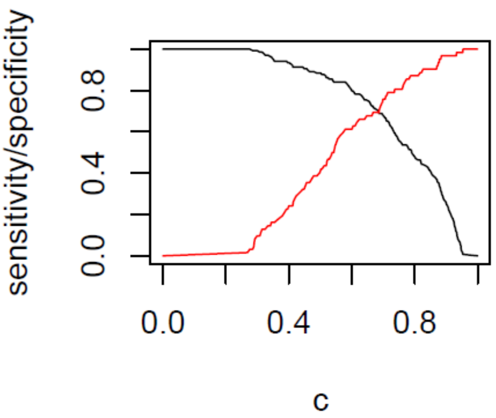
Defining and evaluating the classifier

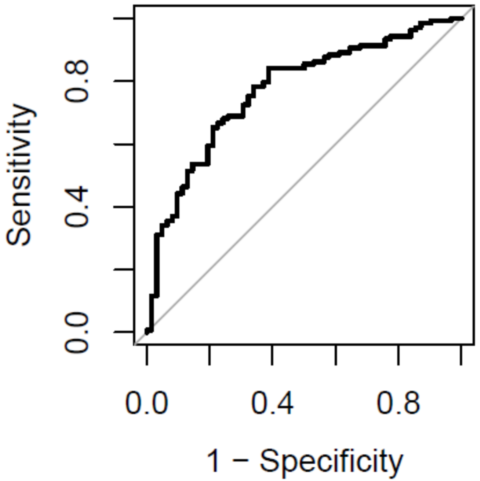
- Set c so we can maximize both sensitivity and specificity
  - Plot sens and spec as a function of c



Defining and evaluating the classifier

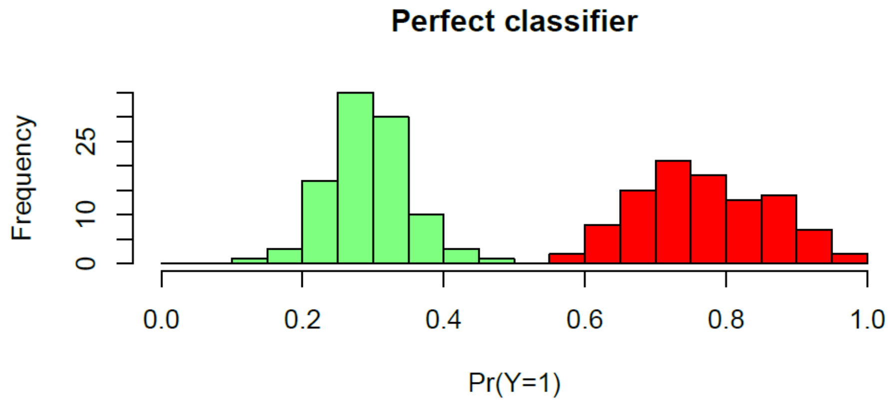
- Evaluate the entire model upon which the classifier is based
  - Receiver Operating Characteristic (ROC) curve
  - Plot sens vs. 1-spec for each c





Example: Perfect classifier

- Plot sens vs. 1-spec for each c



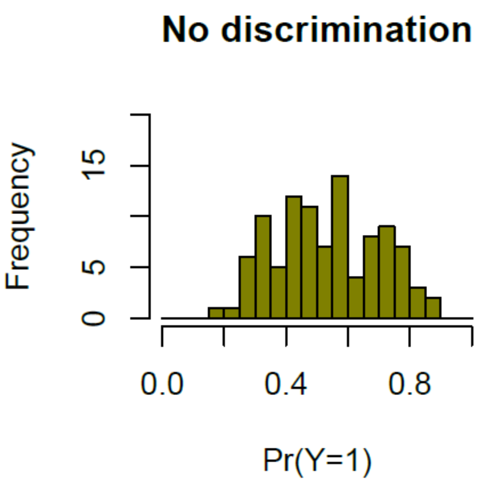
- When c = 1:

- When 0.55 < c < 1:

- When 0 < c < 0.55:

Example: No discrimination

- When c = 1:
- When 0 < c < 1:
- When c = 0:



Area under the ROC curve (AUC)

- The area under the ROC curve represents a measure of discrimination between cases and controls
- Probability that a randomly selected case (Y = 1) has a higher predicted probability than a randomly selected control (Y = 0).
  - You can prove this for fun!
- Perfect discrimination: AUC = 1
- No discrimination: AUC = 0.5

Minimize optimism

- To minimize optimism for your classifier/prediction, you should generate the ROC curve and compute the AUC based on a cross-validation procedure.

Where to next?

- So far, we have considered using a logistic regression model to define a classifier.
- This approach requires that we build the regression model, i.e. we know the key predictors, including functional form for continuous variables and important interactions, etc.
- Instead of building a logistic regression model for developing a classifier, we will consider a classification and regression tree.
  - Removes the need for us to specify the model.