

Logistic Regression

Sections 4.3

Cathy Poliak, Ph.D.
cpoliak@central.uh.edu

Department of Mathematics
University of Houston

Classification

- The response variable, Y , is **qualitative** or **categorical**.
- Predicting a qualitative response for an observations can be referred to as **classifying** that observation.
- These methods predict the probability of each of the categories of a qualitative variables, as the basis for making the classification.

Logistic Regression

- Logistic regression can be used to model and solve problems when the Y (response) variable is a categorical variable with 2 classes.
- Also called binary classification problems.
- This models the **probability** that Y belongs to one of the two categories.

The Logistic Model

- Given $Y = 0$ or 1 , let $p(X) = P(Y = 1|X)$. We want a model that shows the relationship between $p(X)$ and X .

The Logistic Model

- Given $Y = 0$ or 1 , let $p(X) = P(Y = 1|X)$. We want a model that shows the relationship between $p(X)$ and X .
- We use a model that gives outputs between 0 and 1 for all values of X . This is called the **logistic function**

$$p(X) = \frac{\exp^{\beta_0 + \beta_1 X}}{1 + \exp^{\beta_0 + \beta_1 X}}$$

The Logistic Model

- Given $Y = 0$ or 1 , let $p(X) = P(Y = 1|X)$. We want a model that shows the relationship between $p(X)$ and X .
- We use a model that gives outputs between 0 and 1 for all values of X . This is called the **logistic function**

$$p(X) = \frac{\exp^{\beta_0 + \beta_1 X}}{1 + \exp^{\beta_0 + \beta_1 X}}$$

- After some manipulation we get

$$\frac{p(X)}{1 - p(X)} = \exp^{\beta_0 + \beta_1 X}$$

The Logistic Model

- Given $Y = 0$ or 1 , let $p(X) = P(Y = 1|X)$. We want a model that shows the relationship between $p(X)$ and X .
- We use a model that gives outputs between 0 and 1 for all values of X . This is called the **logistic function**

$$p(X) = \frac{\exp^{\beta_0 + \beta_1 X}}{1 + \exp^{\beta_0 + \beta_1 X}}$$

- After some manipulation we get

$$\frac{p(X)}{1 - p(X)} = \exp^{\beta_0 + \beta_1 X}$$

- Take the logarithm of both sides:

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X$$

The left-hand side is called the *log-odds* or *logit*.

The Logistic Model

- Given $Y = 0$ or 1 , let $p(X) = P(Y = 1|X)$. We want a model that shows the relationship between $p(X)$ and X .
- We use a model that gives outputs between 0 and 1 for all values of X . This is called the **logistic function**

$$p(X) = \frac{\exp^{\beta_0 + \beta_1 X}}{1 + \exp^{\beta_0 + \beta_1 X}}$$

- After some manipulation we get

$$\frac{p(X)}{1 - p(X)} = \exp^{\beta_0 + \beta_1 X}$$

- Take the logarithm of both sides:

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X$$

The left-hand side is called the *log-odds* or *logit*.

- We use a method called **maximum likelihood** to determine the best coefficients and eventually a good fit.

Multiple Logistic Regression

We now look at predicting a binary response using multiple predictors.

$$\log \left(\frac{p(X)}{1 - p(X)} \right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

Where $X = (X_1, \dots, X_p)$ are p predictors. This can be rewritten as

$$p(X) = \frac{\exp(\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p)}{1 + \exp(\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p)}$$

We will use the maximum likelihood method to estimate $\beta_0, \beta_1, \dots, \beta_p$.

Breast Cancer Data

```
summary(glm(Class~Cl.thickness+Cell.shape+Cell.size,  
            family="binomial",  
            data=bc))
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-7.7210	0.6969	-11.079	< 2e-16	***
Cl.thickness	0.5918	0.1030	5.746	9.14e-09	***
Cell.shape	0.7240	0.1661	4.358	1.31e-05	***
Cell.size	0.6390	0.1704	3.751	0.000176	***

$H_0: \beta_i = 0$,
given $\beta_j \neq 0$
 $j \neq i$
 $H_1: \beta_i \neq 0$

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

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 884.35 on 682 degrees of freedom

Residual deviance: 176.50 on 679 degrees of freedom

AIC: 184.5

Number of Fisher Scoring iterations: 7

$$P(Y = \text{malignant} | X) = \frac{\exp\{-7.721 + 0.5918 \times \text{Cl.th} + 0.7240 \times \text{cel.sh} + 0.6390 \times \text{cel.s}\}}{1 + \exp\{-7.721 + 0.5918 \times \text{Cl.th} + 0.7240 \times \text{cel.sh} + 0.6390 \times \text{cel.s}\}}$$

Comments

Our model:

$$\hat{p}(X) = \frac{\exp^{-7.7210 + 0.5918 \times \text{Cl.thickness} + 0.7240 \times \text{Cell.shape} + 0.6394 \times \text{Cell.size}}}{1 + \exp^{-7.7210 + 0.5918 \times \text{Cl.thickness} + 0.7240 \times \text{Cell.shape} + 0.6394 \times \text{Cell.size}}}$$

Cl.thickness = 5, cell.shape = 5 and cell.size = 5

$$\hat{p}(x) = \frac{\exp\{-7.7210 + 0.5918(5) + 0.724(5) + 0.6394(5)\}}{1 + \exp\{-7.721 + 0.5918(5) + 0.724(5) + 0.6394(5)\}}$$

$$\hat{p}(Y = \text{malignant}) = 0.8863$$

If $\hat{p}(x) < 0.5$ $Y = 0$ "benign"
 $\hat{p}(x) > 0.5$ $Y = 1$ "malignant"

Confusion Matrix

- The set up is as follows

		Predicted	
		$\hat{Y} = 0$	$\hat{Y} = 1$
Actual	$Y = 0$	Correct true negatives	Incorrect false positives
	$Y = 1$	Incorrect false negative	Correct true positives

- Accuracy: Overall, how often is the classifier correct?

$$\frac{\text{true positive} + \text{true negative}}{\text{total}}$$

- Miss-classification Rate: Overall, how often is it wrong?

$$\frac{\text{false positive} + \text{false negative}}{\text{total}}$$

- Sensitivity: When its actually positive, how often does it predict positive? Also called the true positive rate.

$$\frac{\text{true positives}}{\text{total positives}}$$

$Y = 1$

- Specificity: When it is actually negative, how often does it predict negative? Also called true negative rate.

$$\frac{\text{true negative}}{\text{total negatives}}$$

$Y = 0$

Example

- Confusion matrix for the model: $\hat{p}(X) = \frac{\exp(-5.1645 + 1.4272 \times \text{Cell.shape})}{1 + \exp(-5.1645 + 1.4272 \times \text{Cell.shape})}$

Accuracy rate:

$$\frac{425 + 207}{683} = 0.9253$$

	Predicted: benign	Predicted: malignant
Actual: benign	425	19
Actual: malignant	32	207

$$n = 683$$

Sensitivity

$$\frac{207}{239} = 0.866$$

Specificity

$$\frac{425}{444} = 0.9572$$

- Model:

$$\hat{p}(X) = \frac{\exp(-7.7210 + 0.5918 \times \text{Cl.thickness} + 0.7240 \times \text{Cell.shape} + 0.6394 \times \text{Cell.size})}{1 + \exp(-7.7210 + 0.5918 \times \text{Cl.thickness} + 0.7240 \times \text{Cell.shape} + 0.6394 \times \text{Cell.size})}$$

Accuracy rate:

$$\frac{430 + 219}{683} = 0.95$$

	Predicted: benign	Predicted: malignant
Actual: benign	430	14
Actual: malignant	20	219

Sensitivity

$$\frac{219}{239} = 0.92$$

Specificity

$$\frac{430}{444} = 0.968$$

Lab Questions

1. What is the accuracy rate for the model with three predictors?

a) 0.95

b) 0.05

c) 0.92

d) 0.96

2. What is the specificity rate?

a) 0.95

b) 0.05

c) 0.92

d) 0.96

Why Use A Test and Training Set

- It is important to recall that the confusion matrix will be always biased towards unrealistic good classification rates if it is computed in the same sample used for fitting the logistic model.
- A familiar analogy is asking to your mother (data) whether you (model) are a good-looking human being (good predictive accuracy) – the answer will be highly positively biased.
- To get a fair confusion matrix, the right approach is to split randomly the sample into two: a training dataset, used for fitting the model, and a test dataset, used for evaluating the predictive accuracy.
- From [Statistics for Sciences II](#)

Using the Test and Training Set

```
#Split the bc data set into training and test
sample <- sample.int(n = nrow(bc),
                    size = floor(.75*nrow(bc)),
                    replace = F)

train <- bc[sample, ]
test  <- bc[-sample, ]

train.bc <- glm(Class ~ Cl.thickness + Cell.shape + Cell.size,
                data = train,
                family = "binomial")

#Using the test data to determine the confusion matrix
glm.pred <- predict.glm(train.bc, newdata = test, type = "response")
yHat <- glm.pred > 0.5
table(test$Class, yHat)
```

```
yHat
  FALSE TRUE
0    108   4
1     3  56
```

Goodness-Of-Fit for These Models

- **Deviance** is a measure of goodness-of-fit for the model. Higher numbers indicates bad fit.
- The **null** deviance shows how well the response variable is predicted by a model that includes only the intercept.
- The **residual** deviance show how well the response variable is predicted by a model that includes the independent variables.
- We can use these values as a **generalization** of the R^2 statistic.

R^2 in Logistic Regression

$$R^2 = 1 - \frac{\text{residual deviance}}{\text{null deviance}}$$

- It is a quantity between 0 and 1.
- Similar to the linear regression the closer R^2 is to 1, the better fit.
- Not like the linear regression, this is a ratio indicating how close is the fit to being perfect or the worst.

Lab Question

Predictor(s)	Null Deviance	Residual Deviance	R^2	AIC
Cell.shape	884.35	267.59	0.6974	271.59
Cl.thickness + Cell.shape + Cell.size	884.35	176.5	0.80	184.5

3. What is the R^2 for the model with only Cell.shape as the predictor?

$$1 - \frac{176.5}{884.35} = 0.8$$

a) 0.6974

b) 0.3026

c) 0.6596

d) 0.3404

$$1 - \frac{267.59}{884.35}$$

Example 2

- We will use the `mtcars` data set to predict type of engine base on three variables, `disp`, `hp`, and `wt`.
- First, create a test and training data set based on 80/20 split.

```
set.seed(110)
sample <- sample.int(n = nrow(mtcars),
                    size = floor(.8*nrow(mtcars)),
                    replace = F)
train <- mtcars[sample, ]
test  <- mtcars[-sample, ]
```

- Second, use the `glm()` function to determine the model with all three predictors.

```
cars.glm = glm(vs ~ disp + hp + wt,
              data = train,
              family = "binomial")
summary(cars.glm)
```

Lab Questions

Use the `summary()` function to answer the following questions.

4. In this model is `disp` significant?

a) Yes

b) No

5. Use the `step()` function. Which predictor can be associated with the engine style?

a) `hp`

b) `disp`

c) `wt`

d) none

```
hp.glm = glm(vs ~ hp, data = train, family = "binomial")
summary(hp.glm)
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	8.16993	3.40451	2.400	0.0164 *
hp	-0.06340	0.02822	-2.247	0.0246 *

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

Null deviance: 34.617 on 24 degrees of freedom

Residual deviance: 13.637 on 23 degrees of freedom

AIC: 17.637

Number of Fisher Scoring iterations: 7

```
glm.pred <- predict.glm(hp.glm,newdata = test,type = "response")
```

```
yHat <- glm.pred > 0.5
```

```
table(test$vs,yHat)
```

	yHat	
	FALSE	TRUE
0	4	1
1	0	2

