

# Logistic Regression

Professor: Hammou El Barmi  
Columbia University

- Simple linear regression: relationship between numerical response and a numerical or categorical predictor
- Multiple regression: relationship between numerical response and multiple numerical and or categorical predictors
- What we have not seen is what to when the response is categorical
- Odds: Odds are another way of quantifying the probability of an event (commonly used in gambling (and logistic regression))
- For some event  $E$ ,

$$\text{odds}(E) = P(E)/(1 - P(E)) = P(E)/P(E^c)$$

- Similarly, if we are told the odds of  $E$  are  $x$  to  $y$ , then

$$\text{odds}(E) = x/y = \frac{x/(x+y)}{y/(x+y)}$$

which implies that

$$P(E) = \frac{x}{x+y} \quad \text{and} \quad P(E^c) = \frac{y}{x+y}$$

- Logistic regression is a GLM used to model a binary categorical variable using numerical and categorical variables
- We assume a binomial distribution produced the outcome variable and we therefore want to model  $\pi$  the probability of success for a given set of predictors
- It turns out that there is a very general way of addressing this type of a problem and the resulting models are called generalized linear models. Logistic regression is just one example of this type of model
- All generalized linear models has the following three characteristics:
  - ① A probability distribution describing the outcome variable
  - ② A linear model

$$\eta = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

- ③ A link function that relates the linear model to the parameter of the outcome distribution

$$g(\pi) = \eta \quad \text{or} \quad \pi = g^{-1}(\eta)$$

- Logistic regression is a GLM used to model a binary categorical variable using numerical and categorical predictors
- We assume a binomial distribution produced the outcome variable therefore we want to model  $\pi$ , the probability of success, as a function of some predictors.
- There are a variety of reasonable link functions to use to connect  $\pi$  and  $\eta$ , One such function that is commonly used is the logit function

$$\text{logit}(\pi) = \log\left(\frac{\pi}{1 - \pi}\right), \quad 0 < \pi < 1.$$

- The logit function takes a value between 0 and 1 and maps it to a value between  $-\infty$  and  $+\infty$ .
- The inverse logit (logistic) function is

$$g^{-1}(x) = \frac{e^x}{1 + e^x}$$

- The inverse logit function takes a value between  $-\infty$  and  $\infty$  and maps it to a value between 0 and 1
- This formulation also some use when it comes to interpreting the model a logit can be interpreted as a the log odds of success

- The assumptions are

$$y|x_1, x_2, \dots, x_p = \begin{cases} 1 & \text{with probability } \pi(x_1, x_2, \dots, x_p) \\ 0 & \text{with probability } 1 - \pi(x_1, x_2, \dots, x_p) \end{cases}$$

$$\text{logit}(\pi(x_1, x_2, \dots, x_p)) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

- This implies that

$$\pi(x_1, x_2, \dots, x_p) = \frac{e^{\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p}}{1 + e^{\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p}}$$

- Also

$$\frac{\pi(x_1, x_2, \dots, x_p)}{1 - \pi(x_1, x_2, \dots, x_p)} = e^{\beta_0 + \beta_1 x_1 + \dots + \beta_p x_p}$$

- Interpretation of  $\beta_i$ : When we increase  $x_i$  by one while holding all the other  $x$ s fixed, the odds of getting of 1 change by a multiplicative factor equal to  $e^{\beta_i}$

In R we fit a GLM model in the same way as we did in linear regression except that we use `glm` instead of `lm` and we must specify the type of GLM to fit using the `family` argument.

The data include the number of students admitted, the total number of applicants broken down by gender (the variable `female`), and whether or not they had taken AP calculus (the variable `apcalc`). Since the dataset is so small, we will read it in directly.

# Example

Gender= 0 male 1 female, AP = 1 took AP calculus, 0 did not.

Admit =1 admitted 0 not admitted

Gender	AP	Admit
--------	----	-------

0	0	0
---	---	---

0	0	0
---	---	---

0	0	0
---	---	---

0	0	0
---	---	---

0	0	0
---	---	---

0	0	0
---	---	---

0	0	0
---	---	---

0	0	1
---	---	---

0	1	0
---	---	---

0	1	0
---	---	---

0	1	0
---	---	---

0	1	1
---	---	---

0	1	1
---	---	---

0	1	1
---	---	---

0	1	1
---	---	---

0	1	1
---	---	---

0	1	1
---	---	---

0	1	1
---	---	---

1	0	0
---	---	---

1	0	0
---	---	---

1	0	0
---	---	---

1	0	0
---	---	---

1	0	0
---	---	---



```
> glm(Admit~Gender+AP, family = binomial("logit"))
```

```
Call:  glm(formula = Admit ~ Gender + AP, family = binomial("logit"))
```

Coefficients:

(Intercept)	Gender	AP
-2.0043	0.4537	2.8755

Degrees of Freedom: 28 Total (i.e. Null); 26 Residual

Null Deviance: 39.89

Residual Deviance: 28.67 AIC: 34.67

$$\text{logit}(P(\text{admit} = 1|\text{Gender}, \text{AP})) = -2.0043 + 0.4337\text{Gender} + 2.8755\text{AP}$$

This implies that

$$P(\text{admit} = 1 | \text{Gender}, \text{AP}) = \frac{e^{-2.0043 + 0.4337\text{Gender} + 2.8755\text{AP}}}{1 + e^{-2.0043 + 0.4337\text{Gender} + 2.8755\text{AP}}}$$

The estimated odds of a female being admitted is  $e^{0.4337} = 1.543$  times the estimated odds of a male being admitted controlling for AP (i.e. holding AP fixed).

The estimated odds of a person who has taken AP being admitted is  $e^{2.8755} = 17.73429$  times the estimated odds of a person being admitted controlling for gender (holding gender fixed).

```
> summary(glm(Admit~Gender+AP, family = binomial("logit")))
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.7667	-0.6203	-0.5028	0.8361	2.0643

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-2.0043	0.9170	-2.186	0.02884 *
Gender	0.4537	0.9908	0.458	0.64700
AP	2.8755	0.9898	2.905	0.00367 **

---

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

Null deviance: 39.892 on 28 degrees of freedom

Residual deviance: 28.666 on 26 degrees of freedom

- Note that the model output does not include any F-statistic, as a general rule there are not single model hypothesis tests for GLM models.
- To test  $H_0 : \beta_i = 0$  against  $H_a : \beta_i \neq 0$ , the test statistics is

$$Z = \frac{b_i - 0}{SE(b_i)}$$

and we reject  $H_0$  if  $|Z| > Z_{\alpha/2}$  or if  $p\text{-value} < \alpha$ .

- Example: Test  $H_0 : \beta_{Gender} = 0$  against  $H_a : \beta_{Gender} \neq 0$ . The test statistic is

$$Z = \frac{0.4537 - 0}{0.9908} = 0.458$$

- If  $\alpha = 0.05$  then  $Z_{0.025} = 1.96$ . Since  $|0.458| < 1.96$ , we fail to fail to reject  $H_0$ .

The Wald method: A  $100(1 - \alpha)\%$  confidence interval for  $\beta_i$  is

$$b_i \pm Z_{\alpha/2} SE(b_i)$$

A 95% confidence interval for  $\beta_{\text{Gender}}$  is

$$0.4537 \pm 1.96(0.9908) = [-1.488268, 2.395668]$$

A 95% confidence for  $e^{\beta_{\text{gender}}}$  is

$$[e^{-1.488268}, e^{2.395668}] = [0.2257633, 10.97553]$$

Interpretation: We are 95% confident that the odds of a female being admitted is a number between 0.226 and 10.975 times the odds of a male being admitted given they have the same status on AP.

To produce the confidence intervals for the betas using R, use the following

```
> library(MASS)
> confint.default(glm(Admit~Gender+AP, family = binomial("logit")))
```

if you use

```
> confint(glm(Admit~Gender+AP, family = binomial("logit")))
```

You will get

	2.5 %	97.5 %
(Intercept)	-4.206356	-0.450995
Gender	-1.456204	2.605742
AP	1.115573	5.130797

This result is slightly different because it is based on a different method than the one use above.

Suppose we have the following model

$$\text{logit}(\pi(x_1, x_2, \dots, x_p)) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

and wish to test

$$H_0 : \beta_1 = \beta_2 = \dots = \beta_p = 0$$

against  $H_a$  : at least one of them is not zero. We reject  $H_0$  if

$$-2 \log(L_R/L_F) = \text{Null Deviance} - \text{Residual Deviance} > \chi_p^2(1 - \alpha)$$

$$L_R = \text{maximized likelihood for the reduced model } \text{logit}(\pi(x_1, x_2, \dots, x_p)) = \beta_0$$

and

$$L_F = \text{maximized likelihood for the full model } \text{logit}(\pi(x_1, x_2, \dots, x_p)) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p$$

In our example, Null deviance = 39.89, Residual Deviance = 28.67 and  $p = 2$

$$\text{Null deviance} - \text{Residual Deviance} = 11.22$$

If  $\alpha = 0.05$ ,  $\chi^2_2(.95) = 5.99$



Probit Model

```
> glm(Admit~Gender+AP, family = binomial("probit"))
```

```
Call: glm(formula = Admit ~ Gender + AP, family = binomial("probit"))
```

Coefficients:

(Intercept)	Gender	AP
-1.1848	0.2561	1.7276

Degrees of Freedom: 28 Total (i.e. Null); 26 Residual

Null Deviance: 39.89

Residual Deviance: 28.67 AIC: 34.67

## Example 2

The following data (described in New York Times, Feb. 15, 1191) is used to study the effect of AZT in slowing the development of AIDS symptoms. In the study 338 veterans whose immune systems were beginning to falter after infection with AIDS virus were randomly assigned wither to receive AZT immediately or to wait until their T cells showed severe immune weakness. The data is a 2x2x2 cross classification of the veterans' race, whether they received AZT immediately and whether they developed AIDS symptoms during the three year study.

```
> aids<-read.csv("C:\\Users\\helbarmi\\Desktop\\deathpenalty.csv",
header=TRUE, sep=',')
> attach(aids)
> aids
  race AZTuse yes no
1    w    yes  14 93
2    w    no   32 81
3    b    yes  11 52
4    b    no   12 43
```

The model we want to use here is

$$\text{logit}(P(\text{yes}|\text{race}, \text{AZTuse})) = \beta_0 + \beta_1 \text{race} + \beta_2 \text{AZTuse}$$

To fit this model in R, we use

```
> logit1<-glm(cbind(yes, no)~factor(race)+factor(AZTuse), family=binomial)
> logit1
```

```
Call:  glm(formula = cbind(yes, no) ~ factor(race) + factor(AZTuse),
          family = binomial)
```

Coefficients:

(Intercept)	factor(race)w	factor(AZTuse)yes
-1.07357	0.05548	-0.71946

Degrees of Freedom: 3 Total (i.e. Null); 1 Residual

Null Deviance: 8.35

Residual Deviance: 1.384 AIC: 24.86

Interpretation of the result:

- 1 Interpretation of  $b_1$  the estimate of  $\beta_1$ .: If we hold AZTuse fixed (i.e controlling for AZT use), we estimate the odds that a white person develops AIDS symptoms to be  $e^{0.05548} = 1.057$  times the odds that a black person does (a 5.7% increase roughly)
- 2 Interpretation of  $b_2$  the estimate of  $\beta_2$ .: If we hold race fixed (i.e controlling for race), we estimate the odds that a person who takes AZT develops AIDS symptoms to be  $e^{-0.71946} = 0.49$  times the odds that a person does who does not (a 50% decrease roughly)

## Example 2

You can compute these numbers using

```
> OR=exp(coef(logit1))
```

```
> OR
```

(Intercept)	factor(race)w	factor(AZTuse)yes
0.3417849	1.0570527	0.4870152

- Let  $L_M$  denote the maximized log-likelihood value for a model  $M$  of interest.
- Let  $L_S$  denote the maximized log-likelihood value for the most complex model  $M$ . This model has a separate parameter for each observation and it provides a perfect fit to the data. The model is said to be saturated.
- Because the saturated model has additional parameters, its maximized log-likelihood  $L_S$  is at least as large as the maximized log-likelihood  $L_M$  for a simpler model
- The Deviance of a model  $M$  is defined as

$$\text{Deviance} = -2[L_M - L_S]$$

- In R it is called residual deviance. When the  $M$  is the model that we fit.
- If the model  $M$  is correct, the residual deviance has a chi-square distribution with df equal df of Residual
- If the null deviance is too large, or if the p-value is too small, the model does not capture all the feature in the data and we reject it.
- The Null Deviance is the deviance corresponding to the model

$$\text{logit}(\pi(x_1, x_2, \dots, x_p)) = \beta_0.$$

In the AIDS example, we have

Null deviance: 8.3499 on 3 degrees of freedom

Residual deviance: 1.3835 on 1 degrees of freedom

AIC: 24.86

The residual deviance is 1.3835 with one degree of freedom (WHY?). The p-value in this case

```
> pchisq(1.3835,df=1, lower.tail=FALSE)
[1] 0.2395059
```

- Log likelihoods can also be used to test the hypotheses of nested models (similar to F and partial F in regression)
- Suppose we have two models (a reduced model R and a full model F) and we want to test

$$H_0 : \text{Reduced model} \quad \text{against} \quad H_a : \text{Full model}$$

- The test statistics

$$-2[L_R - L_F]$$

where  $L_R$  and  $L_F$  are the maximized log-likelihoods corresponding to the reduced and the full model, respectively

- Then the likelihood ratio is the ratio likelihood of imposing  $H_0$  over the unrestricted likelihood
- If  $H_0$  is true, the ratio should be near 1



- Under general  $H_0$

$$-2 (\log \text{ of the likelihood ratio}) = -2[L_R - L_F] \sim \chi_k^2$$

where  $k$  is difference in the number of parameters between the full and the reduced. (number of parameters set equal to zero to get the reduced model from the full model).

- Reject  $H_0$  if

$$-2 (\log \text{ of the likelihood ratio}) > \chi_k^2(1 - \alpha)$$

- Under general  $H_0$ ,  $-2 \log(\log \text{ of the likelihood ratio})$  can be computed using the deviances in the output
- Notice that

$$\begin{aligned} -2 (\log \text{ of the likelihood ratio}) &= -2[L_R - L_F] \\ &= -2[(L_R - L_S) - (L_F - L_S)] \\ &= \text{deviance of reduced model} - \text{deviance of full model} \end{aligned}$$

Suppose we want to test

$$H_0 : \beta_1 = \beta_2 = 0$$

against  $H_a$  : Note  $H_0$ . In this case the reduced model is

$$\text{logit}(P(\text{yes}|\text{race}, \text{AZTuse})) = \beta_0$$

and the full model is

$$\text{logit}(P(\text{yes}|\text{race}, \text{AZTuse})) = \beta_0 + \beta_1\text{race} + \beta_2\text{AZTuse}$$

## Example (AIDS example)

The deviances are

Null deviance: 8.3499 on 3 degrees of freedom

Residual deviance: 1.3835 on 1 degrees of freedom

The test statistics  $= 8.3499 - 1.3835 = 6.9674$ . Since  $p = 2$  we reject  $H_0$  since  $6.9674 > \chi^2_2(0.95) = 5.99$ .

Next we look at each individual  $\beta$

```
> summary(logit1)
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-1.07357	0.26294	-4.083	4.45e-05	***
factor(race)w	0.05548	0.28861	0.192	0.84755	
factor(AZTuse)yes	-0.71946	0.27898	-2.579	0.00991	**

---

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

Null deviance: 8.3499 on 3 degrees of freedom

Residual deviance: 1.3835 on 1 degrees of freedom

AIC: 24.86

- From this output we see that the p-value for testing that  $H_0 : \beta_1 = 0$  against  $H_a : \beta_1 \neq 0$  is 0.84755. Therefore, we fail to reject  $H_0$
- Another way to carry out this test (this is just for illustration of the test before for the reduced and full models) is to test

$$H_0 \text{logit}(P(\text{yes}|\text{race}, \text{AZTuse})) = \beta_0 + \beta_2 \text{AZTuse}$$

against

$$H_a : \text{logit}(P(\text{yes}|\text{race}, \text{AZTuse})) = \beta_0 + \beta_1 \text{race} + \beta_2 \text{AZTuse}$$

We fit two models

## ① Full model

```
> fitF<-glm(cbind(yes,no)~factor(race)+factor(AZTuse), family=binomial)
> summary(fitF)
```

```
glm(formula = cbind(yes, no) ~ factor(race) + factor(AZTuse),
     family=binomial)
```

Residual deviance: 1.3835 on 1 degrees of freedom

## ② Reduced model

```
> fitR<-glm(cbind(yes,no)~factor(AZTuse), family=binomial)
> summary(fitR)
```

```
glm(formula = cbind(yes, no) ~ factor(AZTuse), family = binomial)
```

Residual deviance: 1.4206 on 2 degrees of freedom

To get the value of the test statistics we use in R

```
> anova(fitR,fitF, test="Chisq") Analysis of Deviance Table
Model 1: cbind(yes, no)    factor(AZTuse) Model 2: cbind(yes, no)    factor(race)
+ factor(AZTuse) Resid. Df Resid. Dev Df Deviance Pr(>Chi) 1 2 1.4206 2 1
1.3835 1 0.037084 0.8473
> pchisq(fitRdeviance - fitFdeviance, 1, lower=FALSE) [1] 0.847295
```

The p-value = 0.8473

Now we fit the model with AZTuse only

```
> logit2
```

```
Call: glm(formula = cbind(yes, no) ~ factor(AZTuse), family = binomial)
```

Coefficients:

(Intercept)	factor(AZTuse)yes
-1.0361	-0.7218

Degrees of Freedom: 3 Total (i.e. Null); 2 Residual

Null Deviance: 8.35

Residual Deviance: 1.421 AIC: 22.9



```
> summary(logit2)
```

Call:

```
glm(formula = cbind(yes, no) ~ factor(AZTuse), family = binomial)
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-1.0361	0.1755	-5.904	3.54e-09	***
factor(AZTuse)yes	-0.7218	0.2787	-2.590	0.00961	**

---

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

The predicted probabilities in this case are obtained using

```
> predict(fitR, type="response")  
      1      2      3      4  
0.1470588 0.2619048 0.1470588 0.2619048
```

For example

$$0.1470588 = \frac{e^{-1.0361 - 0.7218(1)}}{1 + e^{-1.0361 - 0.7218(1)}}$$

- The  $i$ th residual is given by  $e_i = y_i/n_i - \hat{\pi}_i$  where  $\hat{\pi}_i$  is the predicted probability for the  $i$ th case and  $y_i/n_i$  is the observed proportion of 'yes'. They are obtained in R by typing

```
> residuals(fit, type="response")
```

- The Pearson residuals compare  $y_i$  to its predicted value by the model and they are given by

$$r_i = \frac{y_i - n\hat{\pi}_i}{\sqrt{n\hat{\pi}_i(1 - \hat{\pi}_i)}}$$

- 

- The standardized residuals compare  $y_i$  to its predicted value by the model and they are given by

$$e_i = \frac{y_i - n\hat{\pi}_i}{\sqrt{n\hat{\pi}_i(1 - \hat{\pi}_i)(1 - h_{ii})}}$$

where  $h_{ii}$  is the leverage of the  $i$ th case

To get them in R use

```
stdres<-residuals(fit, type="pearson")/sqrt(1-lm.influence(fit)$hat)
```

- We can also look at deviance residuals defined as

$$r_i^D = \text{sign}(y_i - \hat{y}_i) \sqrt{d_i}$$

where  $d_i$  is the contribution of the observation  $i$  to the residual deviance:

$$d_i = 2 \left( y_i \log \frac{y_i}{\hat{y}_i} + (n_i - y_i) \log \frac{n_i - y_i}{n_i - \hat{y}_i} \right)$$

- They are obtained in R using

```
> residuals(fit, type="deviance")
```

and they approximately normally distributed when the model fits the data (we can use a normal plot)

Binomial logistic regression has two outcomes but how do you deal with 2+ outcomes?

- Re-categorize into 2 outcomes
  - E.g. Pain scale as an outcome; options are: No pain, Mild pain, Moderate Pain, Severe Pain
  - Could re-categorize as: No Pain versus Mild pain, Moderate Pain, Severe Pain OR No Pain/Mild pain versus Moderate Pain, Severe Pain
- Challenges with this approach:
  - Results in an inevitable loss of information
  - Results can change depending on how you collapse your categories
  - Could lead to seriously misleading conclusions

- Another model which considers the full form of the outcome is called the multinomial or polytomous logistic model because the outcome is no longer assumed to be BINOMIAL but rather MULTINOMIAL
- Powerful
- Slightly more complicated interpretation because you are no longer comparing two outcomes (but this isn't a reason not to use it)

Types of multinomial outcomes

- Nominal
- Ordinal

When the response is ordinal, this information can result in simpler and more parsimonious model One can look at

$$\text{logit}(P(Y \leq j)) = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k.$$

- Suppose the response has  $J$  categories
- The response of the  $i$ th individual is  $(Y_{i1}, Y_{i2}, \dots, Y_{iJ})$  where

$$Y_{ij} = \begin{cases} 1 & \text{if the response is } j \\ 0 & \text{otherwise} \end{cases}$$

so that

$$\sum_{j=1}^J Y_{ij} = 1.$$

- Let

$$\pi_{ij} \equiv \pi_{ij}(x_{1i}, x_{2i}, \dots, x_{pi}) = P[Y_{ij} = 1 | x_{1i}, x_{2i}, \dots, x_{pi}].$$

- Choose a baseline or reference response category, the  $J$ th say and let

$$\log \left( \frac{\pi_{ij}(x_{1i}, x_{2i}, \dots, x_{pi})}{\pi_{iJ}(x_{1i}, x_{2i}, \dots, x_{pi})} \right) = \beta_{0j} + \beta_{1j}x_{1i} + \beta_{2j}x_{2i} + \dots + \beta_{pj}x_{pi}$$



- This is equivalent to

$$\pi_{ij}(x_{1i}, x_{2i}, \dots, x_{pi}) = \pi_{iJ}(x_{1i}, x_{2i}, \dots, x_{pi}) e^{\beta_{0j} + \beta_{1j}x_{1i} + \beta_{2j}x_{2i} + \dots + \beta_{pj}x_{pi}}$$

or

$$\pi_{ij} = \pi_{iJ} e^{\beta_{0j} + \beta_{1j}x_{1i} + \beta_{2j}x_{2i} + \dots + \beta_{pj}x_{pi}}$$

- But

$$1 = \sum_{j=1}^J \pi_{ij} = \pi_{iJ} \left( 1 + \sum_{j=1}^{J-1} e^{\beta_{0j} + \beta_{1j}x_{1i} + \beta_{2j}x_{2i} + \dots + \beta_{pj}x_{pi}} \right)$$

therefore

$$\pi_{iJ}(x_{1i}, x_{2i}, \dots, x_{pi}) = \frac{1}{1 + \sum_{j=1}^{J-1} e^{\beta_{0j} + \beta_{1j}x_{1i} + \beta_{2j}x_{2i} + \dots + \beta_{pj}x_{pi}}}$$

and

$$\pi_{ij}(x_{1i}, x_{2i}, \dots, x_{pi}) = \frac{e^{\beta_{0j} + \beta_{1j}x_{1i} + \beta_{2j}x_{2i} + \dots + \beta_{pj}x_{pi}}}{1 + \sum_{j=1}^{J-1} e^{\beta_{0j} + \beta_{1j}x_{1i} + \beta_{2j}x_{2i} + \dots + \beta_{pj}x_{pi}}}$$

- We estimate  $\beta_{0j}, \beta_{1j}, \dots, \beta_{pj}, j = 1, 2, \dots, J - 1$ , by  $b_{0j}, b_{1j}, \dots, b_{pj}, j = 1, 2, \dots, J - 1$ . The estimation technique is the maximum likelihood approach.
- Interpretation of the  $\beta_{\ell j}$ : Given that the response is either  $j$  or  $J$ , if we increase  $x_{i\ell}$  by 1 while holding everything else fixed, the odds of the response is  $j$  change by a multiplicative factor equal to  $e^{\beta_{\ell j}}$

## Example

To illustrate, we analyze the data below. The response  $Y$  = belief in afterlife and it has three categories (yes, undecided and no). The predictor variables are  $X_1$  = race and  $X_2$  = gender. We use no as the baseline. The model is (Yes =1, Undecided=2 and No=3)

$$\log \left( \frac{\pi_j}{\pi_3} \right) = \beta_{0j} + \beta_{1j}X_1 + \beta_{2j}X_2.$$

The SAS program to fit this model is

```
data polylogistic;
input Race Gender afterlife count ;
datalines;
w f yes 371
w f und 49
w f no 74
w m yes 250
w m und 45
w m no 71
b f yes 64
b f und 9
b f no 15
b m yes 25
b m und 5
b m no 13
```

```
proc logistic descending;  
class Race(ref='b') Gender(ref='m');  
freq count;  
model afterlife= Race Gender/link=glogit;  
run;
```

The estimations resulted in

$$\log \left( \frac{\hat{\pi}_1}{\hat{\pi}_3} \right) = 0.883 + 0.342X_1 + 0.419X_2$$

and

$$\log \left( \frac{\hat{\pi}_2}{\hat{\pi}_3} \right) = -0.758 + 0.271X_1 + 0.105X_2$$

This gives

$$\hat{\pi}_3(x_1, x_2) = \frac{1}{1 + e^{0.883+0.342x_1+0.419x_2} + e^{-0.758+0.271x_1+0.105x_2}}$$

$$\hat{\pi}_1(x_1, x_2) = \frac{e^{0.883+0.342x_1+0.419x_2}}{1 + e^{0.883+0.342x_1+0.419x_2} + e^{-0.758+0.271x_1+0.105x_2}}$$

$$\hat{\pi}_2(x_1, x_2) = \frac{e^{-0.758+0.271x_1+0.105x_2}}{1 + e^{0.883+0.342x_1+0.419x_2} + e^{-0.758+0.271x_1+0.105x_2}}$$

Suppose the the response in an ordinal categorical variable with categories  $1, 2, \dots, J$  and suppose we have one predictor variable  $X$ . On way to model this data is

$$\log \left( \frac{P(Y \leq j)}{1 - P(Y \leq j)} \right) = \beta_{j0} + \beta_{j1}X, j = 1, 2, \dots, J - 1$$

Example: You need to install the VGAM library

```
> cumdata
```

dose	death	VegState	MajDis	MiDis	GoodR
1	59	25	46	48	32
2	48	21	44	47	30
3	44	14	54	63	31
4	43	4	49	58	41

Here has the response variable in ordinal and has 5 categories (Death, Vegetative State, Major Disability, Minor Disability, Full Recovery) Suppose we to have the same slope (i.e parallel lines)

```
> fit1<-vglm(cbind(death, VegState, MajDis,MiDis,GoodR)~dose,  
family=cumulative(parallel=TRUE))
```

```
>fit1
```

```
Call:
```

```
vglm(formula = cbind(death, VegState, MajDis, MiDis, GoodR) ~  
      dose, family = cumulative(parallel = TRUE))
```

```
Coefficients:
```

(Intercept):1	(Intercept):2	(Intercept):3	(Intercept):4	dose
-0.7192732	-0.3185389	0.6926380	2.0535418	-0.1747914

```
Degrees of Freedom: 16 Total; 11 Residual
```

```
Residual deviance: 17.99791
```

```
Log-likelihood: -48.77511
```

## Different slopes

```
> fit<-vglm(cbind(death, VegState, MajDis,MiDis,GoodR)~dose, family=cumulative)
>fit
Call:
vglm(formula = cbind(death, VegState, MajDis, MiDis, GoodR) ~
      dose, family = cumulative)
```

## Coefficients:

(Intercept):1	(Intercept):2	(Intercept):3	(Intercept):4	dose:1
-0.86493194	-0.09393276	0.70604063	1.90848851	-0.11211296
dose:2	dose:3	dose:4		
-0.26810555	-0.18115444	-0.11979198		

Degrees of Freedom: 16 Total; 8 Residual

Residual deviance: 3.698068

Log-likelihood: -41.62519



To compare the two fits, we can also use

```
> pchisq(deviance(fit1)-deviance(fit), df=df.residual(fit1)-df.residual(fit),  
lower.tail=FALSE)  
[1] 0.002524161
```

For small frequency tables, it is often convenient to enter data in frequency form using `expand.grid` for the factors and `c()` to list the counts in a vector.

```
example<-data.frame(expand.grid(sex=c("female", "male"),  
  party=c("dem", "ind", "rep")), count=c(279,165,73,47,225,191))
```

```
> example
```

	sex	party	count
1	female	ind	279
2	male	ind	165
3	female	dem	73
4	male	dem	47
5	female	rep	225
6	male	rep	191

```
> names(gss)
```

```
[1] "sex" "party" "count"
```

```
> str(example)
```

```
'data.frame': 6 obs. of 3 variables:
```

```
$ sex : Factor w/ 2 levels "female","male": 1 2 1 2 1 2
```

```
$ party: Factor w/ 3 levels "ind","dem","rep": 1 1 2 2 3 3
```

```
$ count: num 279 165 73 47 225 191
```

```
> sum(example$count)
```

```
[1] 980
```

You can also enter the frequencies in a matrix and assign dimnames, giving the variables names and category labels. Note that, by default, `matrix()` uses the elements supplied by columns in the result, unless you specify `byrow=TRUE`.

```
> ## A 4 x 4 table Agresti (2002, Table 2.8, p. 57) Job Satisfaction
> JobSat <- matrix(c(1,2,1,0, 3,3,6,1, 10,10,14,9, 6,7,12,11), 4, 4)
> dimnames(JobSat) = list(income=c("< 15k", "15-25k", "25-40k", "> 40k"),
  satisfaction=c("VeryD", "LittleD", "ModerateS", "VeryS"))
> JobSat
satisfaction
income VeryD LittleD ModerateS VeryS
< 15k 1 3 10 6
15-25k 2 3 10 7
25-40k 1 6 14 12
> 40k 0 1 9 11
```

JobSat is a matrix, not an object of class("table"), and some functions are happier with tables than matrices. You can coerce it to a table with `as.table()`,

```
> JobSat <- as.table(JobSat)
> str(JobSat)
table [1:4, 1:4] 1 2 1 0 3 3 6 1 10 10 ...
- attr(*, "dimnames")=List of 2
..$ income : chr [1:4] "< 15k" "15-25k" "25-40k" "> 40k"
..$ satisfaction: chr [1:4] "VeryD" "LittleD" "ModerateS" "VeryS"
2.1
```

The `xtabs()` function allows you to create crosstabulations of data using formula style input. This typically works with case-form data supplied in a data frame or a matrix. The result is a contingency table in array format, whose dimensions are determined by the terms on the right side of the formula.

The `xtabs()` function allows you to create crosstabulations of data using formula style input. This typically works with case-form data supplied in a data frame or a matrix. The result is a contingency table in array format, whose dimensions are determined by the terms on the right side of the formula.

```
hair<-c(rep("Black",8), rep("Brown", 8), rep("Blond",8))
gender<-c(rep(rep(Male, 4), rep(Female,4)), 4)
eye<-c(rep(Brown, 8), rep(Hazel, 8), rep(Green, 8), rep(Blue, 8))
count<-c(32,10,3,11,36,5,2,9,53,25,15,50,66,29,14,34,10,7,7,10,16,7,
7,7,3,5,8,30,4,5,8,64)
```

```
> xtabs(count~gender+eye+hair)
, , hair = Black
```

	eye			
gender	Blue	Brown	Green	Hazel
Female	9	36	2	5
Male	11	32	3	10

```
, , hair = Blond
```

	eye			
gender	Blue	Brown	Green	Hazel
Female	64	4	8	5
Male	30	3	8	5

```
, , hair = Brown
```

	eye			
gender	Blue	Brown	Green	Hazel
Female	34	66	14	29
Male	50	53	15	25

```
, , hair = Red
```

	eye			
gender	Blue	Brown	Green	Hazel
Female	7	16	7	7
Male	10	10	7	7

```
structable()
```

For 3-way and larger tables the `structable()` function in `vcd` provides a convenient and flexible tabular display. The variables assigned to the rows and columns of a two-way display can be specified by a model formula.

```
> table<-xtabs(count~gender+eye+hair)
> structable(table)
```

		eye Blue	Brown	Green	Hazel
sex	hair				
Female	Black	9	36	2	5
	Blond	64	4	8	5
	Brown	34	66	14	29
	Red	7	16	7	7
Male	Black	11	32	3	10
	Blond	30	3	8	5
	Brown	50	53	15	25
	Red	10	10	7	7

```
> race<-c("w", "w", "b", "b")
> AZTuse<-c("yes", "no", "yes", "no")
> yes<-c(14,32,11,12)
> now<-c(93,81,52,43)
> fit<-glm(cbind(yes,no)~factor(race)+factor(AZTuse),family=binomial)
> fit$deviance
[1] 1.38353
> fit$null.deviance
[1] 8.349946
> fit$coefficients
      (Intercept)    factor(race)w  factor(AZTuse)yes
      -1.07357363       0.05548453      -0.71945991
> fit$df.null
[1] 3
```



```
> fit$df.residual
[1] 1
> residuals(fit, type="pearson")
      1      2      3      4
-0.5447025  0.4281964  0.7239488 -0.6219896
>
> residuals(fit, type="deviance")
      1      2      3      4
-0.5546645  0.4252503  0.7034699 -0.6325896
> stdres<-residuals(fit, type="pearson")/sqrt(1-lm.influence(fit)$hat)
> stdres
      1      2      3      4
-1.179418  1.179418  1.179418 -1.179418
```

```
> fitF<-glm(cbind(yes,no)~factor(race)+factor(AZTuse),family=binomial)
> fitF
```

```
Call: glm(formula = cbind(yes, no) ~ factor(race) + factor(AZTuse),
  family = binomial)
```

Coefficients:

(Intercept)	factor(race)w	factor(AZTuse)yes
-1.07357	0.05548	-0.71946

Degrees of Freedom: 3 Total (i.e. Null); 1 Residual

Null Deviance: 8.35

Residual Deviance: 1.384 AIC: 24.86

```
> fitR<-glm(cbind(yes,no)~factor(AZTuse),family=binomial)
> fitR
```

```
Call:  glm(formula = cbind(yes, no) ~ factor(AZTuse), family = binomial)
```

```
Coefficients:
```

(Intercept)	factor(AZTuse)yes
-1.0361	-0.7218

```
Degrees of Freedom: 3 Total (i.e. Null); 2 Residual
```

```
Null Deviance:      8.35
```

```
Residual Deviance: 1.421  AIC: 22.9
```

```
> anova(fitR,fitF)
Analysis of Deviance Table

Model 1: cbind(yes, no) ~ factor(AZTuse)
Model 2: cbind(yes, no) ~ factor(race) + factor(AZTuse)
  Resid. Df Resid. Dev Df Deviance
1          2      1.4206
2          1      1.3835  1 0.037084
>
> pchisq(fitR$deviance-fitF$deviance, 1, lower=FALSE)
[1] 0.847295
>
```