Intro to GLM – Day 2: GLM and Maximum Likelihood

Federico Vegetti
Central European University

ECPR Summer School in Methods and Techniques

3 steps of GLM

1. Specify the distribution of the dependent variable
   - This is our assumption about how the data are generated
   - This is the stochastic component of the model
2. Specify the link function
   - We "linearize" the mean of Y by transforming it into the linear predictor
   - It has always an inverse function called "response function""
3. Specify how the linear predictor relates to the independent variables
   - This is done in the same way as with linear regression
   - This is the systematic component of the model

Some specifications

- In GLM, it is the **mean** of the dependent variable that is transformed, not the response itself
- In fact, when we model binary responses, applying the link function to the response itself will produce only values such as $-\infty$ and ∞
- How can we estimate the effect of individual predictors, going through all the steps that we saw, when all we have is a string of 0s and 1s?

Estimation of GLM

- GLMs are usually estimated via a method called "Maximum Likelihood" (ML)
- ML is one of the most common methods used to estimate parameters in statistical models
- ML is a technique to calculate the **most likely** values of our parameters β in the population, **given the data** that we observed
- If applied to linear regression, ML returns exactly the same estimates as OLS
- However, ML is a more general technique that can be applied to many different types of models

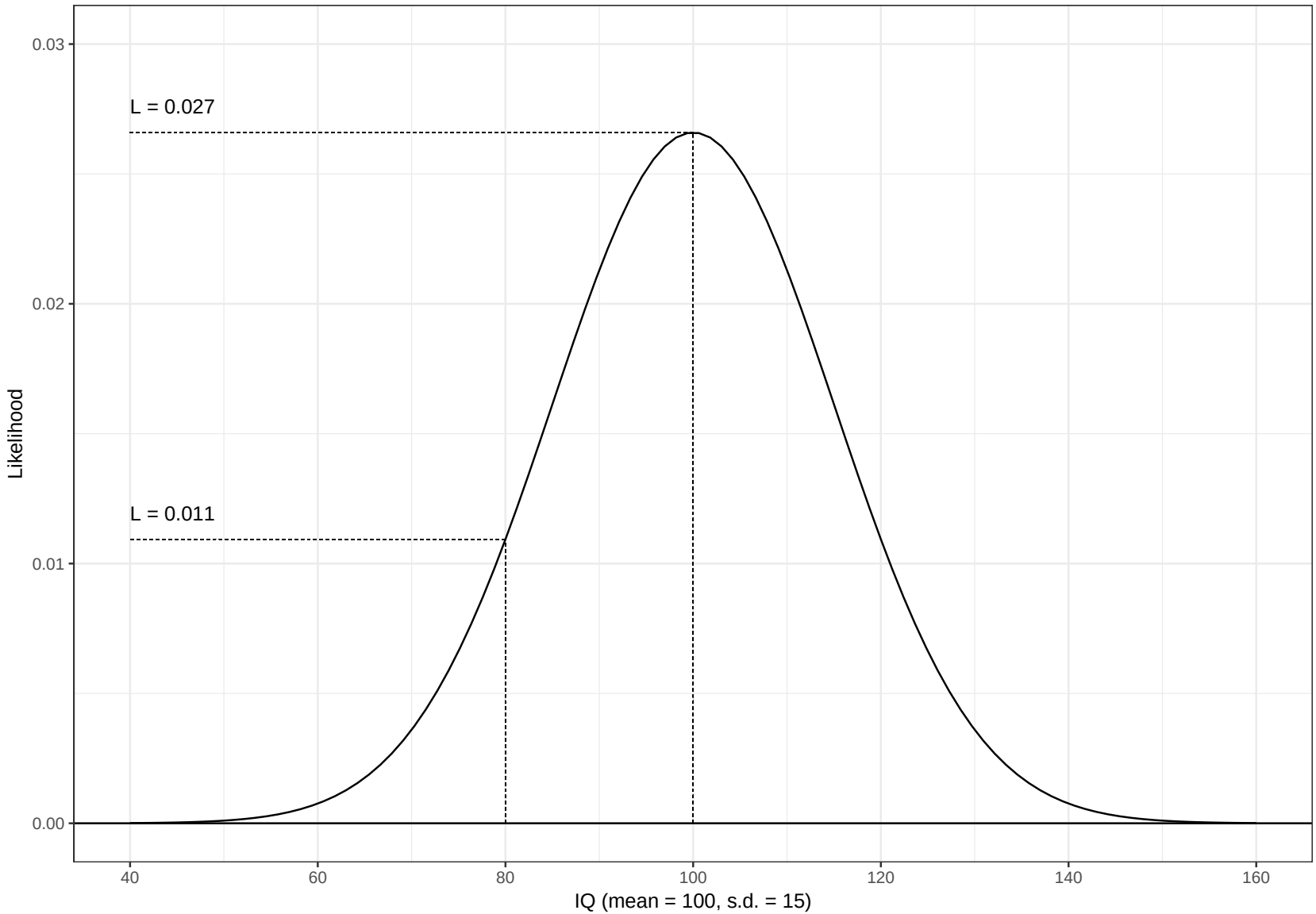
Maximum Likelihood Estimation

- The distribution of your data is described by a "probability density function" (PDF)
- PDF tells you the relative probability, or *likelihood*, to observe a certain value given the parameters of the distribution
- For instance, in a normal distribution, the closer a value is to the mean, the higher is the likelihood to observe it (compared to a value that is further away from the mean)
- The peak of the function (e.g. the mean of the distribution) is the point where it is most likely to observe a value

Likelihood: an example

- Within a given population, we know that IQ is distributed normally, with mean *100* and standard deviation *15*.
- We sample a random individual from the population
- What is more likely:
  - That the IQ of the individual is *100* or
  - That the IQ of the individual is *80*?
- To answer this question we can look at the relative probabilities to pick an individual with $IQ = 100$ and an individual with $IQ = 80$

Likelihood: an example (2)



Likelihood: an example (3)

- In this example we knew the parameters and we plugged in the data to see how likely it is that they will appear in our sample
- Clearly, an individual with $IQ = 100$ is more likely to be observed than an individual with $IQ = 80$, given that the mean of the population is 100
- However, let's suppose that:
  1. We don't know the mean
  2. We pick two random observations: $IQ = [80, 100]$
  3. We assume that IQ is normally distributed
- What is our best guess about the value of the mean?
- This is what ML estimation is all about:
  - We know the data, we assume a distribution, we need to estimate the parameters

Estimating parameters with ML

- Another example: government approval
- Y is our outcome variable, with two possible states:
  - Approve, $y = 1$, with probability p_1
  - Disapprove, $y = 0$, with probability p_0
  - $p_0 + p_1 = 1$
- We do not know p. In order to find it, we need to take a sample of observations and assume a probability distribution
- We want to estimate the probability that citizens support the government (p_1) on a sample of $N = 10$ citizens
- We observe $Y = [1, 1, 0, 1, 0, 1, 0, 1, 0, 1]$
- What is the most likely value of p_1?

Specify the distribution

- Y follows a binomial distribution with two parameters:
  - N, the number of observations
  - p, the probability to approve the government
- The probability to observe an outcome Y in a sample of size N is given by the formula:

$$P(Y|N,p) = \frac{N!}{(N-Y)!Y!}p^Y(1-p)^{N-Y}$$

- So in our case:

$$P(6|10,p) = \frac{10!}{(10-6)!6!}p^6(1-p)^{10-6}$$

- To find the most likely value of p given our data, we can let p vary from 0 to 1, and calculate the corresponding likelihood

Resulting likelihoods

```
b.fun <- function(p) {
  factorial(10)/(factorial(10-6)*factorial(6))*p^6*(1-p)^(10-6)
}
p <- seq(0, 1, by = 0.1)
xtable(data.frame(p = p, likelihood = b.fun(p)))
```

	p	likelihood
1	0.00	0.00
2	0.10	0.00
3	0.20	0.01
4	0.30	0.04
5	0.40	0.11
6	0.50	0.21
7	0.60	0.25
8	0.70	0.20
9	0.80	0.09
10	0.90	0.01
11	1.00	0.00

Maximum likelihood

- The values in the right column are **relative probabilities**
- They tell us, for an observed value Y, how likely it is that it was generated by a population characterized by a given value of p
- Their absolute value is essentially meaningless: they make sense with respect to one another
- Likelihood is a measure of *fit* between some observed data and the population parameters
- A higher likelihood implies a better fit between the observed data and the parameters
- The goal of ML estimation is to find the population parameters that are more likely to have generated our data

Individual data

- Our example was about grouped data: we modeled a proportion
- What do we do with individual data, where Y can take only values *0* or *1*?
- ML estimation can be applied to individual data too, we just need to specify the correct distribution of Y
- For binary data, this is the Bernoulli distribution: a special case of the binomial distribution with $N = 1$:

$$P(y|p) = p^y(1-p)^{1-y}$$

- Once we have a likelihood function for individual observations, the sample likelihood is simply their product:

$$L(p|y,n) = \prod_{i=1}^{n} p^{y_i}(1-p)^{1-y_i}$$

Individual data (2)

- Let's calculate it by hand with our data
- Remember: $Y = [1, 1, 0, 1, 0, 1, 0, 1, 0, 1]$
- What's the likelihood that $p = 0.5$?

$$L(p = 0.5|6, 10) = (0.5^1 * (1-0.5)^0)^6 * (0.5^0 * (1-0.5)^1)^4 = 0.0009765625$$

- What about $p = 0.6$?

$$L(p = 0.6|6, 10) = (0.6^1 * (1-0.6)^0)^6 * (0.6^0 * (1-0.6)^1)^4 = 0.001194394$$

- What about $p = 0.7$?

$$L(p = 0.7|6, 10) = (0.7^1 * (1-0.7)^0)^6 * (0.7^0 * (1-0.7)^1)^4 = 0.0009529569$$

Likelihood and Log-Likelihood

- The sample likelihood function produces extremely small numbers: this creates problems with rounding
- Moreover, working with multiplications can be computationally intensive
- These issues are solved by taking the logarithm of the likelihood function, the "*log-likelihood*"
- The formula becomes:

$$l(p|y,n) = \sum_{i=1}^{n} y_i log(p) + (1 - y_i) log(1 - p)$$

- It still qualifies relative probabilities, but on a different scale
- Since likelihoods are always between *0* and *1*, log-likelihoods are always negative

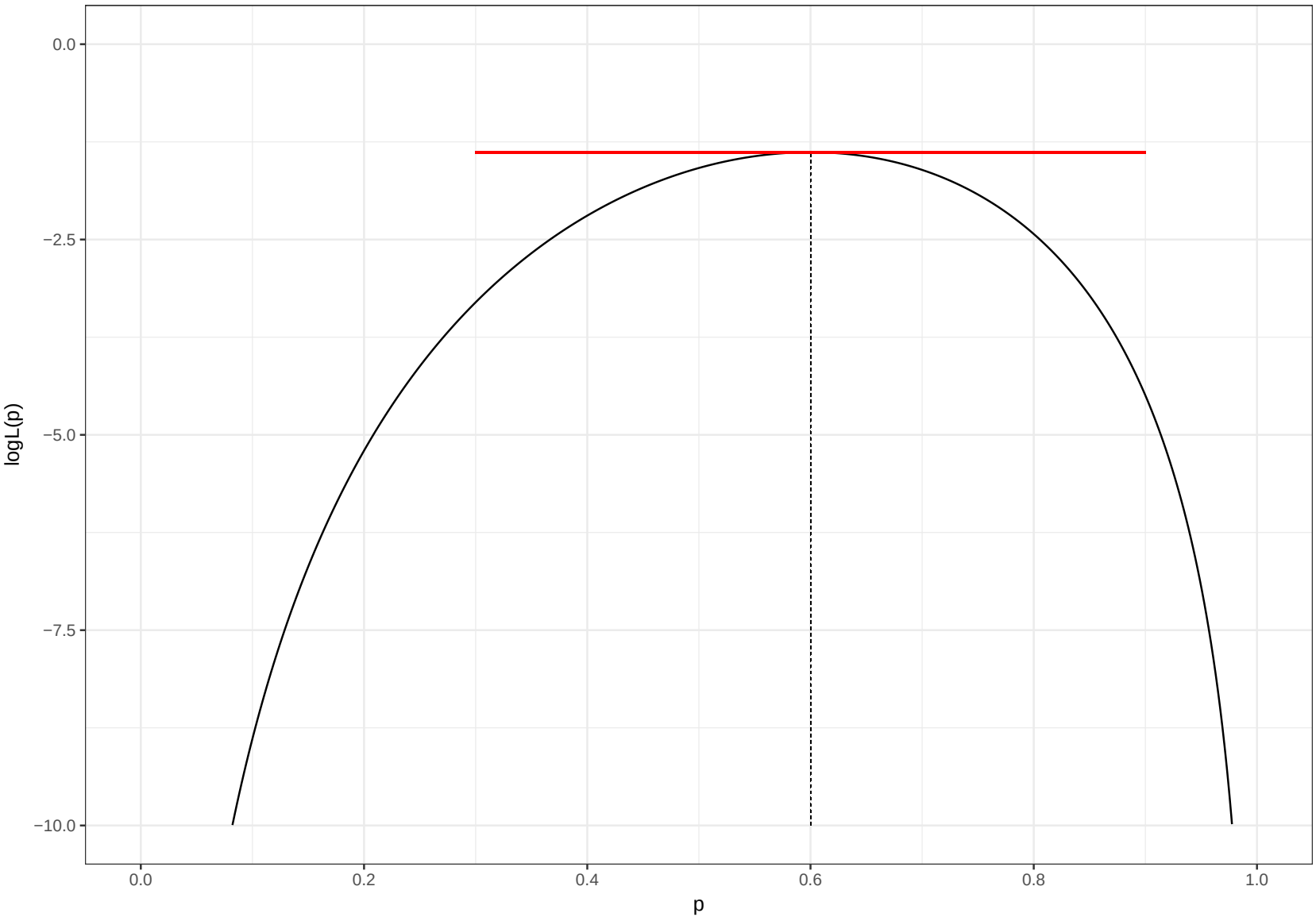
Individual Likelihood and Log-Likelihood

	p	L	logL
1	0.0	0.0000000	
2	0.1	0.0000007	-14.237
3	0.2	0.0000262	-10.549
4	0.3	0.0001750	-8.651
5	0.4	0.0005308	-7.541
6	0.5	0.0009766	-6.931
7	0.6	0.0011944	-6.730
8	0.7	0.0009530	-6.956
9	0.8	0.0004194	-7.777
10	0.9	0.0000531	-9.843
11	1.0	0.0000000	

How to estimate parameters with ML?

- For simple problems, we can just plug in all possible values of our parameter of interest, and see which one corresponds to the maximum likelihood (or log likelihood) from the table
- However, for more complex problems, we need to search directly for the maximum
- How? We look at the **first derivative** of the log-likelihood function with respect to our parameters
  - The first derivative tells you how steep is the slope of the log-likelihood function at a certain value of the parameters
  - When the first derivative is *0*, the slope is flat: the function has reached a peak
  - If we set the result of the derivative formula to *0* and solve for the unknown parameter values, the resulting values will be the maximum likelihood estimates

Log likelihood function and first derivative



How to estimate the parameters? (2)

- We also need the **second derivative** of the function. Why?
  - When it is *positive*, the function is *convex*, so we reached a "valley" rather than a peak
  - When it is *negative*, it confirms that the function is *concave*, and we reached a maximum
- Moreover, we use second derivatives to compute the standard errors:
  - The second derivative is a measure of the curvature of a function. The steeper the curve, the more certain we are about our estimates
  - The matrix of second derivatives is called "Hessian"
  - The inverse of the Hessian matrix is the variance-covariance matrix of the estimates
  - The standard errors of ML estimates are the square root of the diagonal entries of this matrix

In sum

- The likelihood function tells us the relative probability that a given set of population parameters has generated the data
- Maximum Likelihood is the common way to estimate parameters in GLM
- It's a very flexible technique, and can be applied to many different distributions

ML estimation of Logit models

- How do we apply this to a regression model?
- Say we want to estimate a vector of parameters β
- Our response variable follows a binomial distribution with $N = 1$:

$$P(y_i|\pi) = \pi^{y_i}(1-\pi)^{1-y_i}$$

- The likelihood function $L(\pi)$ is the product across individual contributions:

$$L(\pi|y) = \prod_{i=1}^{n} \pi^{y_i}(1-\pi)^{1-y_i}$$

Likelihood function of a Logit model

- Solving the likelihood equation from the previous slide leads to:

$$L(\pi|y) = \prod_{i=1}^{n} \left(\frac{\pi}{1-\pi}\right)^{y_i} (1-\pi)$$

- Remember the logit function:

$$\frac{\pi_i}{1-\pi_i} = exp(X_i\beta) \qquad \text{and} \qquad 1-\pi_i = \frac{1}{1+exp(X_i\beta)}$$

- Plugging this into the likelihood function for the binomial model leads to the probability function of the data in terms of the parameters of the logit model:

$$L(\beta|y, X) = \prod_{i=1}^{n} exp(X_i\beta)^{y_i} \frac{1}{1+exp(X_i\beta)}$$

Log-likelihood function of a Logit model

- However, we prefer to take the log-likelihood function:

$$l(\beta|y,X) = \sum_{i=1}^{n} y_i log(\pi) + (1-y_i) log(1-\pi)$$

- Which, after solving and plugging in the logit function, becomes:

$$l(\beta|y,X) = \sum_{i=1}^{n} y_i X_i \beta - \sum_{i=1}^{n} log[1 + exp(X_i \beta)]$$

ML Estimation

- At this point we can look for the maximum likelihood by taking the first derivative of the equation in respect to β, and setting it to 0
- This is usually done iteratively:
  - We choose some arbitrary starting values of β
  - We evaluate the vector of partial derivatives of the log-likelihood function
  - We update the values of β using the information given by the partial derivatives
  - We stop when we reach values sufficiently close to 0
- There are several optimizing algorithms, more or less precise (and fast)
- The good news is, the software will take care of this

Wald test

- When we estimate one or more coefficients in a logit model, we typically want to test for the null hypothesis $H_0 : \beta = 0$
- This is done with the Wald test, which is equivalent to the t-test done for linear regression

$$Wald = \frac{\beta}{se_\beta}$$

- The Wald statistic is approximately normally distributed, so it is interpreted in the same way as z-scores

Likelihood and model fit

- We want to test whether the fit of our model improves as we add predictors
- *Fit* refers to the likelihood that a model with a given set of predictors has generated the observed data
- The **likelihood ratio** test compares the log-likelihood of our "unrestricted" model (UM) with the one of a "restricted" model where all βs are set to *0* (RM)

$$LR = -2(l(\beta_{RM}) - l(\beta_{UM}))$$

- It requires that the two models are *nested*:
  - All the terms in the restricted model occur also in the unrestricted model
  - In other words, the restricted model must not include parameters (or observations) that are not included in the unrestricted model

Likelihood and model fit (2)

- The same logic applies to the **deviance test**
- Here we compare the fit of the proposed model (PM) with the fit of a "saturated" model (SM)
- The saturated model has one parameter per observation, so it describes the data perfectly

$$D = -2(l(\beta_{PM}) - l(\beta_{SM}))$$

- R reports two measures of deviance:
  - The *residual deviance*, which is equal to $-2l(\beta_{PM})$
  - The *null deviance*, which is equal to $-2l(\beta_{NM})$, where NM is a model that includes only the intercept
- These measures are different from D
- However, if compared to each other, they describe the increase in model fit from the null model

Likelihood and model fit (3)

- Other statistics that *R* reports are the **Akaike Information Criterion** and the **Bayesian Information Criterion**
- Both measures are based on the log-likelihood, but penalize for the number of parameters included in the model
- Moreover, the BIC takes into account the number of observations as well

$$AIC = -2l(\beta) + 2p$$

$$BIC = -2l(\beta) + log(n)p$$

- Where p is the number of parameters in the model, and n is the number of observations
- In both cases, the smaller the value, the better the model fit
- They can be used to compare models that are not nested, i.e. with different parameters

Classification table

- An alternative strategy to assess the fit of a model is to look at the quality of the *predictions* made by the model
- Logit and probit models can produce predicted values of π by multiplying the observed Xs with the estimated βs
- We can predict individual responses by setting a threshold above which we expect $y = 1$ (e.g. $\pi = 0.5$)
- By comparing the **predicted** with the **observed** values of y, we can determine how well the model describes the data

Classification table (2)

	Predicted	
Observed	0	1
0	n_{00}	n_{01}
1	n_{10}	n_{11}

- A useful measure is the percentage of correctly classified cases:

$$\frac{n_{00} + n_{11}}{n_{00} + n_{01} + n_{10} + n_{11}}$$

- Since we could achieve *0.5* with a series of coin flips, a good-fitting model should produce a much larger value

Hosmer–Lemeshow test

- The Hosmer–Lemeshow test consists of 3 steps:

1. We divide the distribution of our predicted π in quantiles
2. For each quantile, we calculate the **expected** number of $y = 1$ and $y = 0$
3. And we compare it with the **observed** number of $y = 1$ and $y = 0$

- We perform a Pearson's chi-squared test to see whether the difference between the expected and observed distribution of y is significant
- If it's not, then the model fits the data well

The pseudo R-square

- Sometimes, articles report measures of "pseudo R-square"
- *SPSS* and *Stata* report them too
- There are different kinds of pseudo R-square. The most famous are probably the Cox & Snell and the Nagelkerke R^2 reported by *SPSS*
- Generally speaking, these measures involve comparing the likelihood of the null model (the model with the intercept only) to the likelihood of our model.
- Pseudo R-squares are **not** a measure of explained variance
- They can be interpreted as a measure of proportional improvement of fit
- Some of them are not even bounded between 0 and 1