# 17. Logistic Regression

Until now, we have been looking at *linear* regression. We now move onto *Logistic* Regression, which is a form of regression for modelling binary outcome measures. One major difference between linear and logistic regression is in the interpretation of the coefficients, which in logistic regression take the form of *log odds* or *odds ratios*.

### 17.1 Tasks for this week

Conceptual material is covered in the lecture. In addition to the live lecture, you can find the lecture recording and additional materials on Canvas.

Please work through the guided exercises in this section (everything except the page labelled “Tutorial Exercises”) in advance of the computer-based tutorial session.

### 17.2. Learning Objectives

#### CONCEPTUAL

After this week, you should be able to:

* Understand the principles underpinning logistic regression, including the *logit transformation*.
* Recognise and interpret coefficients as log odds and odds ratios.
* Be able to convert the output of logistic regression into predicted probabilities.
* Understand the fit statistics for logistic regression.

These points will be covered in the lecture.

#### PYTHON

We will be working with the “statsmodels” in Python again. This week, you will learn to:

* Fit logistic regression models in Python.
* Request model output in odds ratios.
* Convert the model output into predicted probabilities.
* Interpret basic model fit statistics.
* Understand, and work with, the ‘effective sample’ in statistical models.

### 17.3 Logistic Regression basics

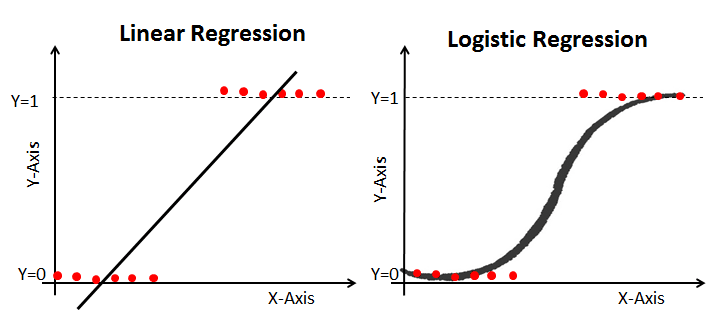
Q: When should we use logistic regression rather than linear regression?

A: We use logistic regression (also called binary logistic regression) when we have a binary outcome variable, such as: is the person vaccinated? Does the patient smoke? Is the pregnancy test positive or negative? Did it rain yesterday? All of these questions have two possible outcomes (vaccinated or not vaccinated, smoker or non-smoker, etc.) that will be coded as 0,1 in the data.

------------------------------------------

Q: Why can’t we use linear regression for binary outcomes?

A: One of the assumptions of linear regression is that the observed data should contain a linear relationship, and when we have a binary outcome measure this assumption is violated. The problem is illustrated in the graph below. As the binary outcome variable can only take a value of 0 or 1 (illustrated by the red dots), the linear regression is also a problem because it is ‘unbounded’. It can take values below 0 or above 1 which would be hard to interpret in terms of predicting the probability of a yes/no outcome. In logistic regression we ‘transform’ the outcome variable so that it is bounded and can be interpreted in terms of the probability (or odds, we’ll come to this) of observing the outcome.



(Graphic from DataCamp)

-------------------------------------------

Q: What is a *logit* transformation?

A: In logistic regression, we use **the log of the odds** of observing the outcome, which is called the *logistic transformation*, or *logit* for short.

But let’s start at the beginning: What are odds? What’s a log?

-------------------------------------------

Q: What are ‘odds’?

In everyday language, the term *odds* is used interchangeably with probability. But in statistics, there is a specific definition. Let’s imagine a sample of 100 adults in the UK. We find out that 92 of those 100 have had at least one dose of a covid-19 vaccine. The probability is straightforwardly:

But, the **odds** of observing the outcome are:

So, here:

= = 11.5

We can interpret the odds to mean that being vaccinated is 11.5 times more likely than not being vaccinated. **Note that probability and odds of two ways of expressing the same idea.**

Say we also have a sample of children aged 5-11. Among these, only 11 of 100 have been vaccinated. In this case, the odds are = 0.124.

Q: If the probability of something is 50/50, what are the odds?

A: 1. So, we have seen that a probability >50% produces odds with a value > 1 and a probability <50% produces odds with a value < 1. And when a probability is exactly 50/50 the odds will be exactly 1. We’ll come back to this later when we interpret the output of our logistic regression models.

-------------------------------------------

Q: What’s a “log”?

A log (or logarithm) expresses the inverse of an exponential. If you did A-level maths, this should sound familiar, but if not, here’s quick overview. Exponentials refer to ‘power’ functions, such as 23 (two to the power of three, also known as two cubed), which means 2 x 2 x 2 (and equals 8).

23 can be expressed as:

Which says that the log base 2 of 8 is equal to 3.

More generally, when = then

Graphical user interface, application

Description automatically generated

Q: Can you express 103 as a log?

A:

------------------------------------------

Q: Let’s try this one, . What is ?

A: The question we are trying to solve is: two to the power of *what* equals 16? The answer is 4 (because 24 = 16).

On Canvas, you’ll find additional materials on logs, if you need them.

-------------------------------------------

In our examples above, we have seen log base 2 and log base 10 (these are the easiest to work with for working out in your head). Instead of using 2 or 10, the logit function in logistic regression uses the “natural logarithm” or where the base is the constant which is much used in maths and statistics. has a value of 2.718 and is known as Euler’s number or Euler’s constant. You don’t need to worry too much about *why* this is the way it is done, or *where* Euler’s constant comes from, but this is what is going on in our logarithmic transformation. (For the curious, there is an easy video on Canvas about Euler’s constant). The natural logarithm is often written as *ln*.

Q: Can you figure out what the answer to this one would be?

A: = 0 (because anything to the power of zero = 1).

----------------------------------------------------------------------

### 17.4 The logistic regression equation

The equation uses the log of the odds = as the outcome.

We model the logit (the log of the odds) as follows:

In a regression model with independent variables.

The natural log of the odds would take a value of zero when the probability is 0.5. With a probability of 1 logit(p) would be infinity, and with a probability of 0, logit(p) would be minus infinity.

The logit transformation gets around the problem that the assumption of linearity has been violated. The transformation is a way of expressing a non-linear relationship in a linear way.

Apart from the outcome variable, the form of the regression equation is very familiar! Like in linear regression the slope estimate describes the change in the outcome variable for each unit of , and like in linear regression, the intercept is the value of the outcome variable when all variables take the value zero. However, in interpreting the coefficients, we need to keep in mind the transformation function of , which we’ll practice in the next section.

### 17.5 Predicted probabilities (worked example).

**Example:** A logistic regression model describes whether the probability of voting for Candidate X in an election depends on = the voter’s total family income (in thousands of dollars) the previous year. The prediction equation is:

Before we get to python, this exercise is a chance to practice converting logistic regression output into predicted probabilities by hand (i.e., by calculator or in Excel!) to help you see what is going on.

Q: Identify and interpret its sign.

A: 0.03. The estimated probability of voting for Candidate X increases as income increases.

--------------------------------

Q: Find the estimated probability of voting for the candidate when income = 10,000.

A: When we plug in a value of = 10, we find logit(p) = -1.7.

The alternative equation for logistic regression (derived from the equation above, see lecture slides) that expresses the probability directly is:

=

Thus, =



The probability of voting for Candidate X when income = 10,000 (i.e., = 10) is 0.15, or 15%.

---------------------------------

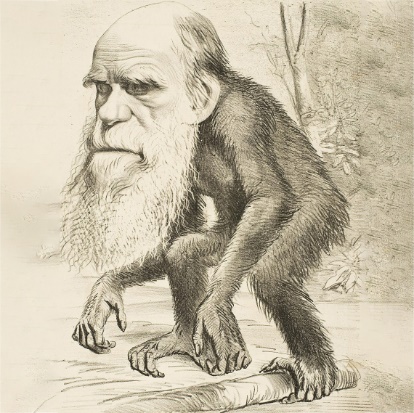
Q: At which income level is the estimated probability for the candidate equal to 0.50?

A: When , the odds , and . So, we take 0 = and solve for . We then find that when

Thus, for this example, thousand dollars.

---------------------------------

### 17.6 Python code and interpreting logistic regression coefficients.



**Example:** The General Social Survey is a regular survey of the population of the United States that gathers information on people’s opinions since 1972. Here we use data from 2018, when the following question was put the respondents: “Human beings, as we know them today, developed from earlier species of animals. True or false?”

In this analysis, = opinion about evolution (1 = true, 0 = false). We have three variables for the analysis: age (measured in years), political ideology (measured from 1 = extremely conservative to 7 = extremely liberal, which we treat as continuous here), and whether the participant studied science at college (yes, no). Before we start, have a think about these variables: which direction do you expect the relationship to go? E.g., will older or younger people be more likely to answer “true”?

For analysis in python, we’ll import the relevant packages as usual and import the data as a csv file.

# Set-up Python libraries - you need to run this but you don't need to change it

import numpy as np

import matplotlib.pyplot as plt

import scipy.stats as stats

import pandas

import seaborn as sns

import statsmodels.api as sm

import statsmodels.formula.api as smf

Code for importing evolution.csv data.

We ask python for a logistic regression model with the following code:

Code for logistic regression, y = evolution, x = age + polviews + colsci

The standard model output looks rather like the output table from linear regression. Add output table. The coefficients in the column “coef” are in log odds. We can interpret these coefficients in the following way: as people get older they are less likely to agree that humans evolved from earlier species and as people’s political views get more liberal they are more likely to agree. The log odds for studying science at college is also positive; however, it is not statistically significant.

Because logs odds are rather hard to interpret intuitively (for most of us anyway), we can convert them into odds ratios by exponentiating them, to help with interpretation.

Code for giving output as odds ratios.

Add output table. Remember the practice session with logs above? The same principles apply here. Odds of 1 relate to a 50/50 probability, which we can interpret in a logistic regression model as no effect (i.e., the variable is not associated with a higher or lower probability of the outcome occurring). Positive log odds become odds ratios with a value greater than 1, and negative log odds become odds ratios with a value below 1. Here, we can interpret the model out as follows: The odds ratio of 1.47 for political views suggests that for each additional point along the scale from conservative to liberal is associated with a 1.47 times greater likelihood of agreeing that humans evolved from earlier species. The output from python provides confidence intervals too. The 95% confidence intervals for the odds ratio are 1.29 – 1.68. For age, the odds ratio tells us that for each additional year the odds of answering ‘true’ are 0.02 (or 2%) lower (using 1-0.98 for the effect size). (Here I will not interpret ‘colsci’ as it is not statistically significant. We know this from the p-value but also because the confidence intervals include 1. Note the contrast with the original model with log odds in this regard).

Python will compute a predicted probability for us. Let’s find out the predicted probability of believing in evolution for a 50-year-old with a score of 5 on the conservative scale, who did not study science at college.

Code for calculating predicted probability.

### 17.7 Assessing the model.

Just as we did in linear regression, we can obtain predicted values for the model. The predicted values can help us to understand how well our model did. They take a value between 0 and 1 and can be treated as a predicted probability of each individual answering ‘true’ to the survey question, given the variables that we have modelled.

Code for predicted values and inspect data.

We can compare how well the model prediction matches the observed data in a classification table which classifies (using Pr(y=1)>0.5 as cut-off) which cases would be predicted as true or false, in a table by whether the observed value was true or false. The classification table shows that for the =1 cases (n=282), the model predicted 212 correctly, and 70 incorrectly. For the cases reporting ‘false’ (y=0, n = 221), 115 were correctly classified and 106 incorrectly. Overall, 65% of the cases were correctly classified by this model. Show output.

While a linear regression uses the method of Ordinary Least Squares (OLS) to fit the model, logistic regression uses the *Maximum Likelihood Estimation* method. This method uses the values of the model parameters that are most consistent with the observed data, so that with the intercept and slope values estimated in the model, the observed data have a greater chance of occurring than with any other estimated values (See Agresti, Chapter 5). The log-likelihood is based on summing the probabilities of the observed and actual outcomes and tells us how much unexplained information there is after the model has been fitted (thus analogous to residual sum of squares in linear regression). Like in linear regression R2, we use a ‘baseline’ model with no variables (predicting the outcome that occurs most often), and compare the log-likelihood after adding variables. The likelihood-ratio test, (in python the *LLR p-value*) produces a p-value so that we can tell if our model is statistically significant. Low p-values (below 0.05) tell us that the model with the s is significantly better at predicting the outcome than the baseline model.

### 17.8 Tutorial Exercises

In the tutorial you will be analysing data (real but edited) from the 2021 General Social Survey from the USA. The General Social Survey is a regular survey of the population of the United States that gathers information on people’s opinions. The sample is intended to be representative of the adult population. The outcome variable of interest (‘afraid’) is whether people feel afraid to walk in their neighbourhoods at night. The question wording was as follows: "Is there any area around your home--that is, within a mile--where you would be afraid to walk alone at night?" The answer options are yes (1) or no (0). (Some people also answered “don’t know” but these are set to missing).

The variables are as follows:

* Sex (male, female)
* Age (in years)
* Race of respondent (white, black, other)
* Born (whether respondent was born in the USA, yes or no)
* Raclive (a binary variable indicating whether the neighbourhood is racially mixed).
* Educ (education measured in years of education, where more years assumes higher educational qualifications)
* Income (an ordinal variable of household income, where higher values mean higher incomes. Treat as continuous for this analysis)

First, install the relevant packages, import the data ‘fear.csv’, and inspect your data.

Students own code for installing packages.

Students own code for importing fear.csv and inspecting data.

Before running your first logistic regression model, think about your expectations here. Would you expect men or women to be more afraid of walking in their neighbourhoods at night? And would you expect older or younger people to feel more afraid?

Let’s test these associations between age and sex with fear of walking in neighbourhoods. Run a logistic regression model with = afraid, = sex + age.

Students own code for logistic regression model with = afraid, = sex + age (model 1).

Interpret the output. Which coefficients are statistically significant? Is the direction of effect as expected? Find the p-value for the log-likelihood ratio. Is this model statistically significant compared to the baseline model?

Compute the predicted probability that a 65-year-old woman reports feeling afraid.

Students own code for predicted probability.

Run a second logistic regression model. Keep age and sex in the model, and add race, born, raclive. and educ. (We are treating educ as a continuous variable). Print the coefficients as odds ratios.

Students own code for logistic regression model with = afraid, = sex + age + race + born + raclive + educ (model 2).

Students own code for odds ratios.

Make a few notes on the findings using the odds ratios. How would you report the size of the effect using the odds ratios?

How good is this second model? Check the classification table based on predicted probabilities.

Students own code for classification table.

Run a third logistic regression model adding a control variable for income. Request the output in odds ratios again.

Students own code for model 3.

Interpret the results carefully. Do any of your conclusions change now that we are controlling for income? (Compare the coefficients in model 2 and model 3). Why do you think this might be?

Extra exercise: look back at the sample size in each of the three logistic regression models. Why do you think the sample size is changing each time?

If you want to keep the same sample throughout all of the analysis, we can save the “effective sample” from model 3 with the following code.

Code for saving effective sample from model 3.

Now re-run model 1 and check the sample size. Are there any changes in coefficient estimates once you have restricted the analysis to the effective sample?

Run model 1 on effective sample.