Logistic regression

Steven Kerr

Supported by
THE UNIVERSITY of EDINBURGH | Data-Driven Innovation

Logistic regression

- For ordinary least squares, we looked at models of the form
$$y_i = \beta \cdot x_i + \epsilon_i.$$
- Suppose, however, that I wished to predict the probability of a binary outcome.
- For example, p_i is the probability that individual i tests positive for COVID-19.
- We could try something similar,
$$p_i = \beta \cdot x_i.$$
- The problem is that $p_i \in [0, 1]$. However, in our model there is nothing stopping p_i being greater than 1, or less than 0.
- We need a new modelling strategy that takes account of this.

Logistic regression

- Introducing the logistic model:

$$p_i = \frac{1}{1+e^{\beta \cdot x_i}}.$$

- $e = 2.71828\ldots$ is *Euler's number*. It is a number that has special significance in mathematics, alongside e.g. π.

- The logistic model has the nice property that no matter what value β and x_i take, p_i is *always* between 0 and 1.

Logistic regression

- It is possible to show that (exercise for the mathematically inclined)

$$\ln\left(\frac{p_i}{1-p_i}\right) = \beta \cdot x_i.$$

- $\ln$ is the *natural logarithm*.

- $\frac{p_i}{1-p_i}$ is called the *odds* of the event. It is the probability of the event occurring, divided by the probability of the event not occurring.

Maximum likelihood

- Logistic models are trained using *maximum likelihood* methods.
- This involves choosing $\hat{\beta}$ to maximise the probability that we saw the data that we did, assuming our model is correct.
- Let's say individual i tests positive for COVID-19. The probability of this happening is $p_i = \frac{1}{1+e^{\beta \cdot x_i}}$.
- Individual j does not test positive for COVID-19. The probability of this happening is $(1 - p_i) = 1 - \frac{1}{1+e^{\beta \cdot x_i}} = \frac{1}{1+e^{-\beta \cdot x_i}}$.

Maximum likelihood

- We assume that observations are statistically independent of each other.

- The probability of seeing the data that we saw is

$$L(\beta) = \Pi_{i|y_i=1}\, p_i\, \Pi_{i|y_i=0}(1 - p_i).$$

- We choose $\hat{\beta}$ to maximise this *likelihood function*.

Odds ratios

- If β_k is the coefficient on a binary variable x_{ik} in a logistic model, it is possible to show that e^{β_k} is the *odds ratio* between individuals who have $x_{ik} = 1$, and individuals who have $x_{ik} = 0$ (another exercise for the mathematically inclined).

- That is, if we divide the event odds for those with $x_{ik} = 1$ by the event odds for those with $x_{ik} = 0$, we will get e^{β_k}.

- Estimated odds ratios are typically the primary result that is reported in papers that fit logistic models.

Confidence intervals

- Recall that when the sample is large, our estimates $\hat{\beta}_k$ are normally distributed.
- We can also estimate the standard deviation of that normal distribution, $\hat{\sigma}$.
- We can then use $\hat{\sigma}$ to calculate a $(1 - \alpha)$ confidence interval.
- For example, take $\alpha = 0.05$. If we repeated our analysis a large number of times with a fresh sample, 95% of the time the true value would be in the 95% confidence interval.

Confidence intervals

- The 95% confidence interval for $\hat{\beta}_k$ is

$$(\hat{\beta}_k - 1.96\hat{\sigma},\ \hat{\beta}_k + 1.96\hat{\sigma})$$

- The 95% confidence interval for the estimated odds ratio $\mathrm{e}^{\widehat{\beta}_k}$ is

$$(\mathrm{e}^{\widehat{\beta}_k - 1.96\widehat{\sigma}},\ \mathrm{e}^{\widehat{\beta}_k + 1.96\widehat{\sigma}}).$$

- If you use OLS in your project, you should report parameter estimates and their 95% confidence intervals.
- If you use a logistic model in your project, you should report odds ratios and their 95% confidence intervals.

sparklyr

- Logistic regression can also be used in the case where the dependent variable is categorical with many levels.

- In this case, it is referred to as a *multinomial model*.

- In sparklyr, you can use the following command to fit a logistic/multinomial model:

  ml_logistic_regression(data, formula)

- However, this does not report standard deviations of parameter estimates.

sparklyr

- In sparklyr, a logistic model with a binary dependent variable can also be fitted using the following command:

  ml_generalized_linear_regression(data, formula, family = 'binomial')

- This cannot be used for multinomial models, but it does report standard deviations of parameter estimates.