Logistic regression

Binary, ordinal and multinomial

David Barron

Hilary Term 2017

Generalised linear models

Generalised linear models

GLMs are a generalisation of the linear models we've looked at over the past two weeks. They allow us to investigate regression models where the outcome variable is one of several important special forms. The simplest of these is when the outcome variable has only two possible values, such as "success" and "failure". Some estimation software requires the variable to be coded 0/1, and we will use that coding in the explanation below.

All GLMs have three basic components:

- Probability distribution (sometimes called the "stochastic component");
- Linear predictor (the "systemmatic component");
- Link function

GLM for binary outcomes: probability distribution

The GLM for binary outcome variables is often called *logistic regression*. The probability distribution associated with it is the *binomial* distribution:

$$\Pr(Y = k|n, p) = \binom{n}{k} p^k (1-p)^{n-k},$$

for $k = 0, 1, 2, \ldots, n$ and where $\binom{n}{k} = \frac{n!}{k!(n-k)!}$. In the special case where $n = 1$, this reduces to

$$\Pr(Y = k|p) = p^k (1-p)^{1-k},$$

where $k = 0, 1$. The parameter p is what we are interested in estimating; it is the probability that the outcome variable, $Y = 1$.

Linear predictor

The linear predictor always has the same form in all GLMs. It consists of the explanatory variables that we think are associated with the probability that $Y = 1$. So, this looks very much like the linear regression model:

$$\eta(x) = \beta_o + \beta_1 x_1 + \cdots + \beta_j x_j.$$

Note, though, there is no "error term." The randomness is provided by the probability distribution we've just specified. (If you like, you could think of GLMs as having different "error terms" to the normal distribution we use in linear regression. Of you could think of linear regression as being a GLM with the normal distribution as its probability distribution.)

Link function

The general *link function* is defined as:

$$\eta(x) = f[\mu(x)],$$

where $\mu(x)$ is the parameter of the probability distribution we are interested in.

You might think that in this case we could just put these two together in a straightforward way:

$$p = \eta(x) = \beta_o + \beta_1 x_1 + \cdots + \beta_j + x_j,$$

but, while this is in fact technically possible, there would be significant problems with this model. The two main problems are:

- You could get predicted values of p that are either smaller than 0 or larger than 1, but as p is a probability, this is logically impossible.
- A linear model implies that the impact of a one-unit change in any x is the same regardless of the value of p, but this can't be true. It must be "harder" to increase the probability from, say, 0.90 to 0.95 than it would be to increase it from 0.50 to 0.55.

Therefore, a different link function is most commonly used.

Logit link function

The logit link function is

$$\eta(x) = \log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_i x_1 + \cdots + \beta_j x_j,$$

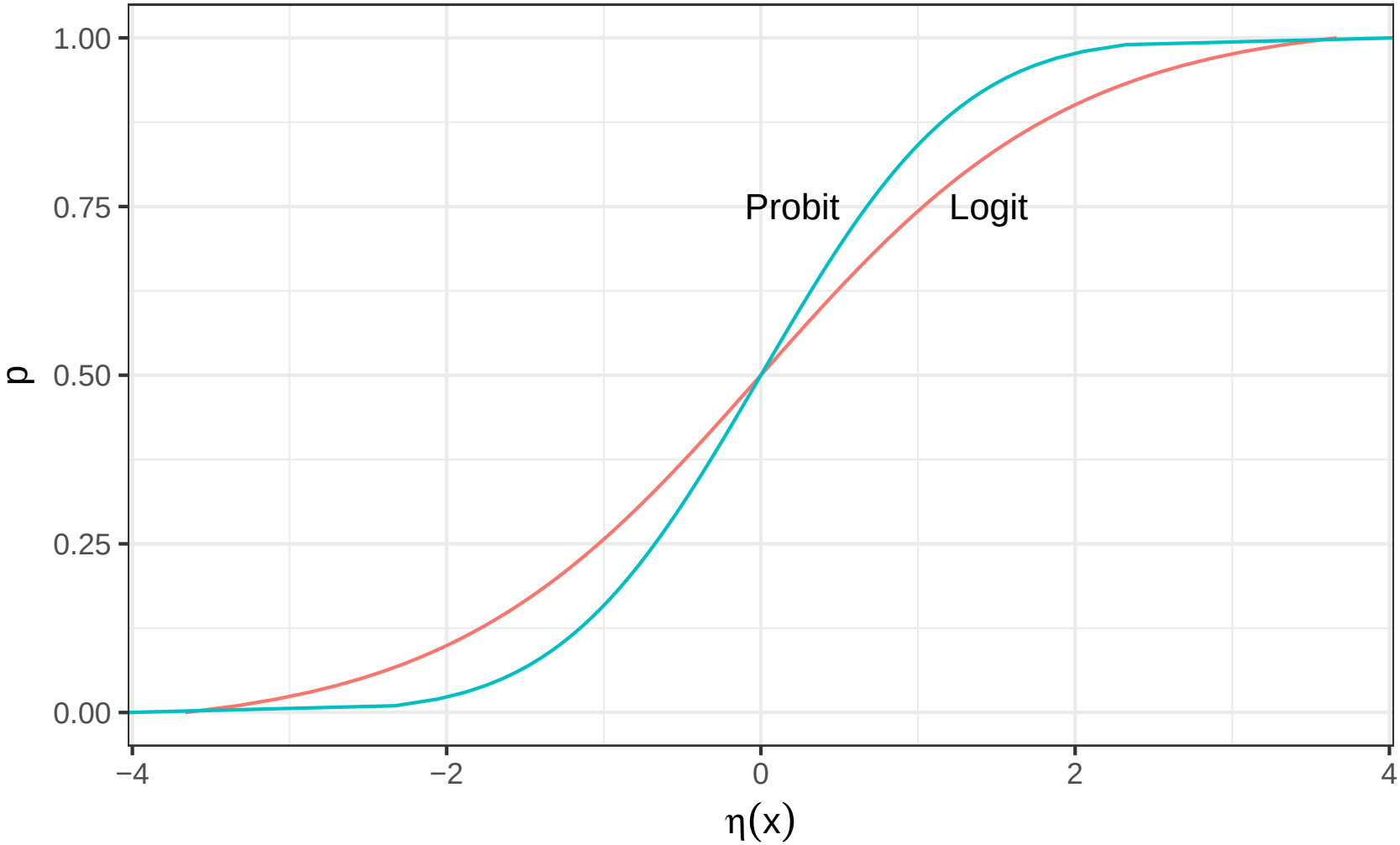
which can be rearranged to give

$$p = \frac{1}{1+e^{-\eta}}.$$

$p/(1-p)$ is often called the *odds* (or *odds ratio*), and so another name for the logit function is the *log odds*.

An alternative link function that is sometimes used is the normal cumulative probability distribution, often called the *probit* function. This is virtually identical to the logit function, and as interpretation of results using the logit function is generally easier, it is much more common.

Plot of the logit and probit functions



Maximum likelihood estimation

Estimation of GLMs is straightforward, but it's useful to have some intuition about what is going on "under the hood." Iterative (ie, trial and error) methods have to be used. The computer tries values of the βs in the model, uses them to calculate predicted values of p and then use that to calculate the likelihood of observing the actual outcomes given those values of p. The iterations continue until the values of p that result in the maximum likelihood is found. The corresponding values of the βs are the maximum likelihood estimate of those parameters.

Fortunately, while there are general purpose ML estimation functions in R (and if you want to make sure you really understand these principles, it is a good idea to see if you can figure out how to use them to implement logistic regression), there are special purpose functions that make it easy to implement any GLM.

Logistic regression

Logistic regression example

This example uses data from the Panel Study of Income Dynamics that relate to women's labour force participation. The respondents are all married women. The outcome is whether the woman is employed or not. Explanatory variables: *k5*: number of children 5 or under; *k618*: number of children 6–18; *age*; *wc*: attended college; *lwg*: log expected wage; *inc*: family income.

```
Call:
glm(formula = lfp ~ k5 + k618 + age + wc + lwg + inc, family = binomial,
    data = Mroz)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-2.099  -1.094   0.601   0.972   2.177

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.21664    0.64090    5.02  5.2e-07
k5          -1.45610    0.19614   -7.42  1.1e-13
k618        -0.06427    0.06797   -0.95     0.34
age         -0.06364    0.01270   -5.01  5.4e-07
wcyes        0.86225    0.20673    4.17  3.0e-05
lwg          0.60454    0.15062    4.01  6.0e-05
inc         -0.03318    0.00783   -4.24  2.3e-05

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1029.75  on 752  degrees of freedom
Residual deviance:  905.56  on 746  degrees of freedom
AIC: 919.6
```

Interpretation

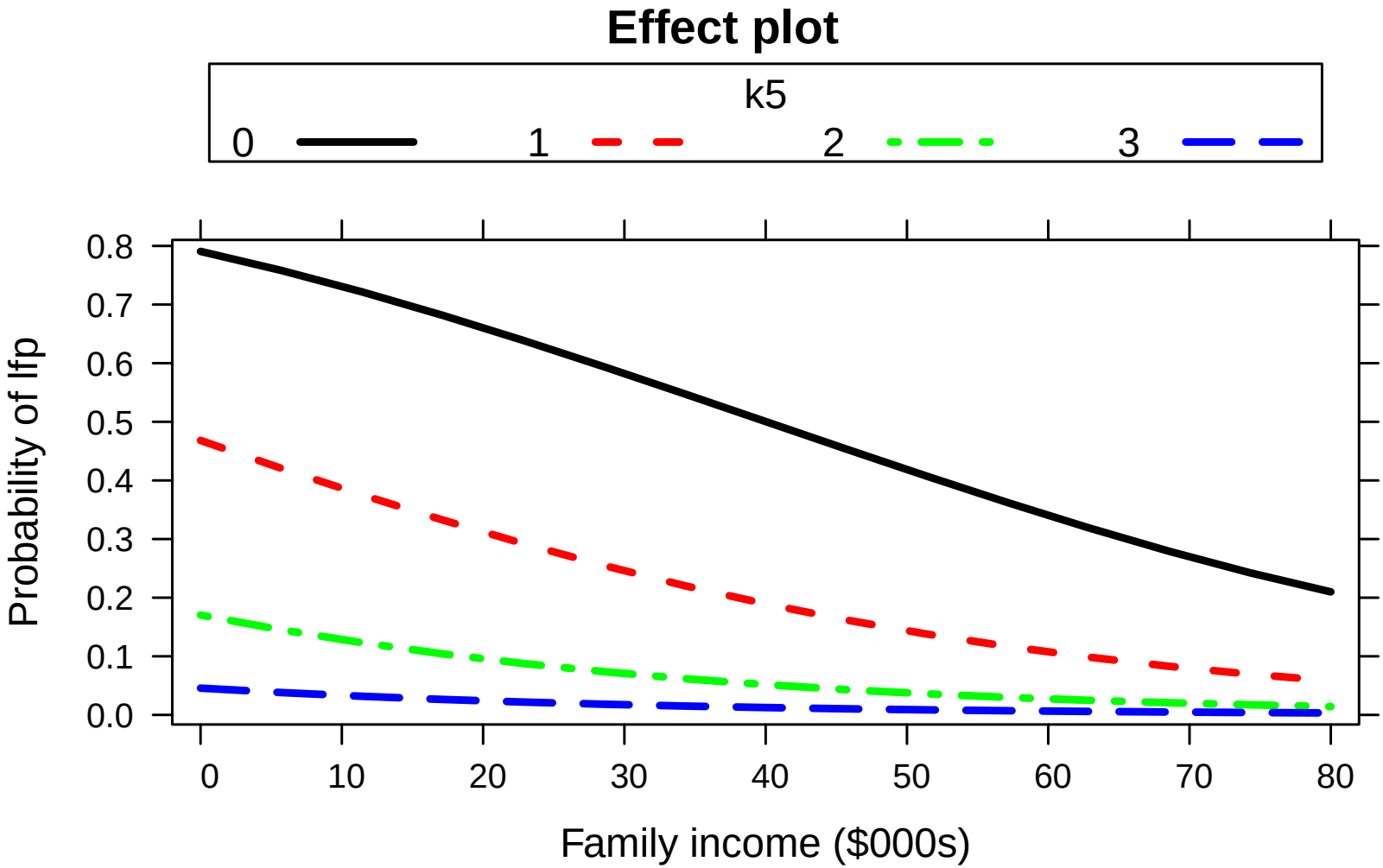
Parameter estimates are interpreted as effect on logit or log odds. For example, for each additional $1000 of family income, the log odds of being in the labour force declines by 0.033. This isn't intuitive, but it is easy to see the direction of the effect and to assess statistical significance. For example, you can see that the probability of a woman being in employment goes down as the number of pre-school children goes up, while having been to college increases the probability of employment. You might prefer to calculate confidence intervals:

```
Waiting for profiling to be done...
```

```
                2.5 %  97.5 %
(Intercept)  1.9787  4.4945
k5          -1.8522 -1.0821
k618        -0.1980  0.0689
age         -0.0889 -0.0391
wcyes        0.4619  1.2736
lwg          0.3143  0.9065
inc         -0.0490 -0.0182
```

Effect plot

In this example, we can see that the effect of family income varies depending on how many pre-school children are in the family.



Interpretation, continued

The effect of each explanatory variable on the probability varies both across values of that explanatory variable and across values of all the other explanatory variables. This is why effect plots are particularly useful for logistic regression (and all other GLMs). Even these involve some simplification. The example on the previous slide fixed the values of the number of school-age children, age, college educated, and log expected wage at their sample mean values. This is conventional, but you might ask yourself whether it makes sense for dummy variables.

Odds ratios

An alternative is to report parameter estimates as effects on the odds ratio, which you can obtain simply by using the anti-log:

```
round(exp(cbind(Estimate = coef(l1), confint(l1))), 2)
```

```
Waiting for profiling to be done...
```

	Estimate	2.5 %	97.5 %
(Intercept)	24.94	7.23	89.52
k5	0.23	0.16	0.34
k618	0.94	0.82	1.07
age	0.94	0.91	0.96
wcyes	2.37	1.59	3.57
lwg	1.83	1.37	2.48
inc	0.97	0.95	0.98

So, each additional $1000 of family income reduces the odds of working by 3 per cent.

Goodness of fit

We can compare goodness of fit of nested models using the deviance. The deviance is defined as twice the difference between the model log likelihood and the log likelihood of the saturated model (i.e., the best possible fit). The difference between the deviances of nested models has a χ^2 distribution with degrees of freedom equal to the number of extra parameters estimated in the more complex model. The `anova` function will calculate this for you:

Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
746	906	NA	NA	NA
745	904	1	1.06	0.302

Other goodness of fit statistics

A number of other GoF statistics have been suggested as analogues of the R^2 statistic often used in linear regression. In these formulae, L_0 is the likelihood of a regression with only an intercept, L_m is the likihood of the model actually estimated, and n is the sample size.

Cox and Snell Index

$$R^2_{CS} = 1 - (L_0/L_m)^{2/n}.$$

One drawback of this statistic is that the upper bound is not 1, but rather is $1 - L_0^{2/n}$.

Nagelkerke's Index

$$R^2_N = \frac{R^2_{CS}}{1 - L_0^{2/n}}.$$

As you can see, this is the Cox and Snell index divided by the upper bound of this index, which therefore now has an upper bound of 1.

McFadden's R^2

$$R^2_{McF} = 1 - \log(L_m)/\log(L_0)$$

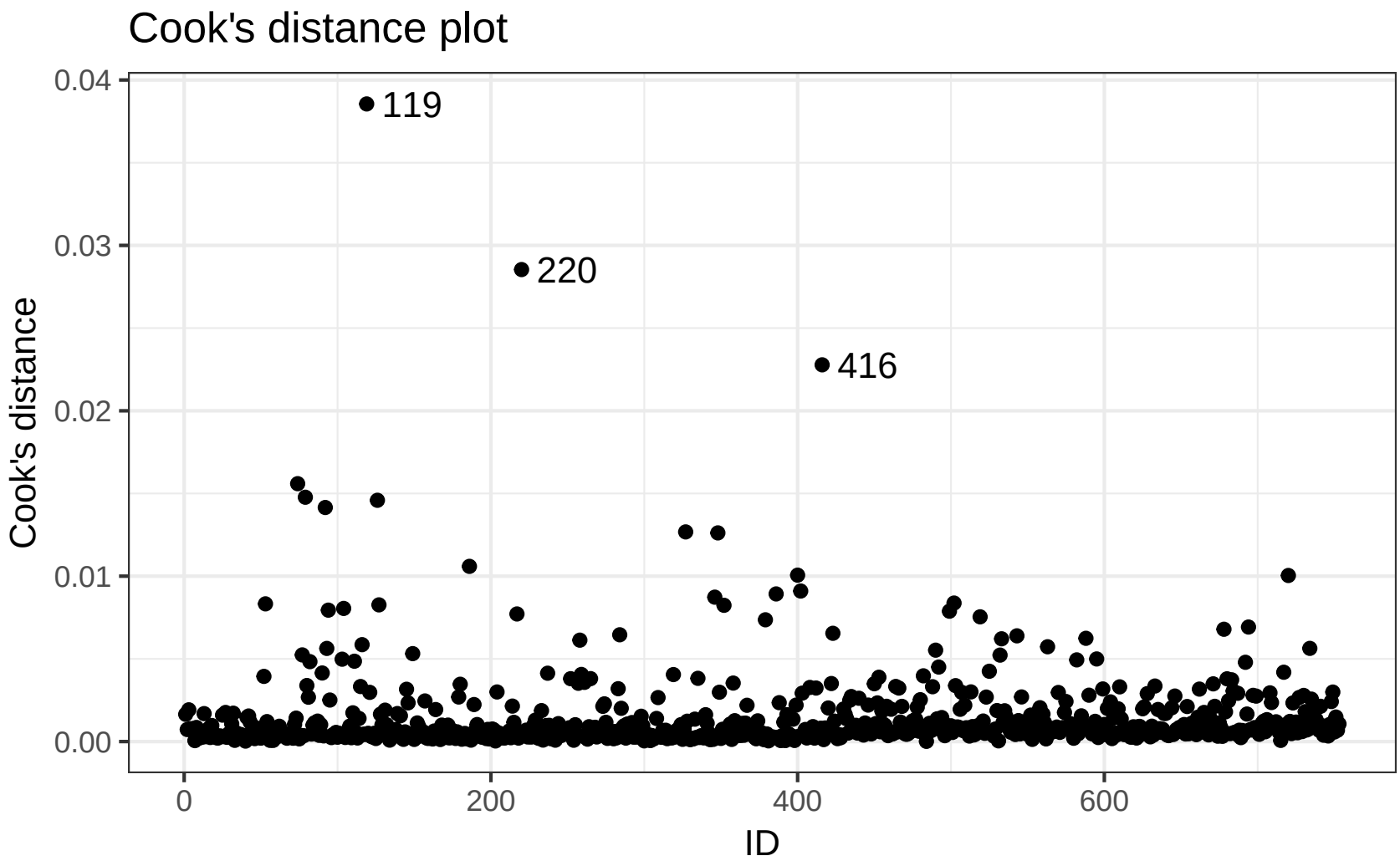
Example

```
descr::LogRegR2(l1)
```

```
Chi2                  124
Df                    6
Sig.                  0
Cox and Snell Index   0.152
Nagelkerke Index      0.204
McFadden's R2         0.121
```

Outlier detection

Similar methods to those used in linear regression can be used to check for outliers.



Ordinal logistic regression

Ordinal outcomes

Sometimes we have outcome variables that take a small number of discrete, ordered categories. (If there are many categories, you would probably be best advised to treat it as a numeric variable.) For example, I have been doing research into the quality of adult residential care facilities, and this has categories "Poor", "Fair", "Good", and "Excellent." We want to use a method that uses the information about ordering in the data. There are several possible alternatives, but I am going to explain only the most straightforward. It is often just called **ordinal logistic regression**, although strictly speaking it is just one version of ordinal logit. Sometimes it is called the *proportional odds* model, which would be a less ambiguous name for it.

Proportional odds logistic regression

The simplest model for ordinal logistic regression. Our linear predictor is:

$$\eta(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k.$$

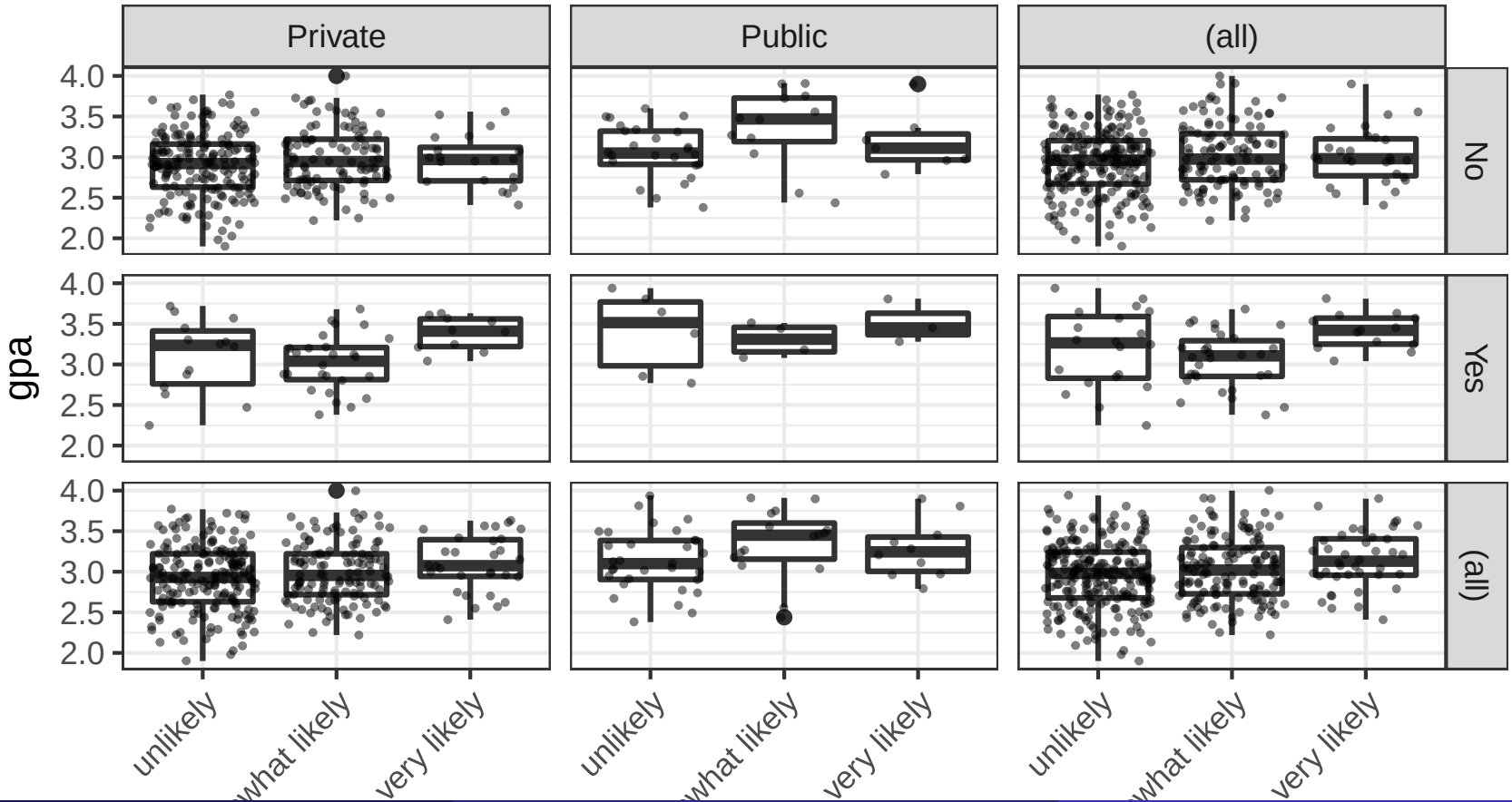
Then we have

$$\begin{aligned}
\text{logit}(p_m) &= \eta(x) \\
\text{logit}(p_m + p_{m-1}) &= \eta(x) + \alpha_1 \\
\text{logit}(p_m + p_{m-1} + p_{m-2}) &= \eta(x) + \alpha_1 + \alpha_2 \\
&\cdots \\
\text{logit}(p_1) &= 1 - (\eta(x) + \alpha_1 + \alpha_2 + \cdots + \alpha_{m-2})
\end{aligned}$$

So, if we have an outcome variable with three categories, we first consider the log odds of being in the highest category against being in either of the other two categories, then the log odds of being in the middle category agains being in the lowest category. The linear predictor is constrained to be the same in each case, with a threshold parameter (the αs) being estimated for each one.

Example

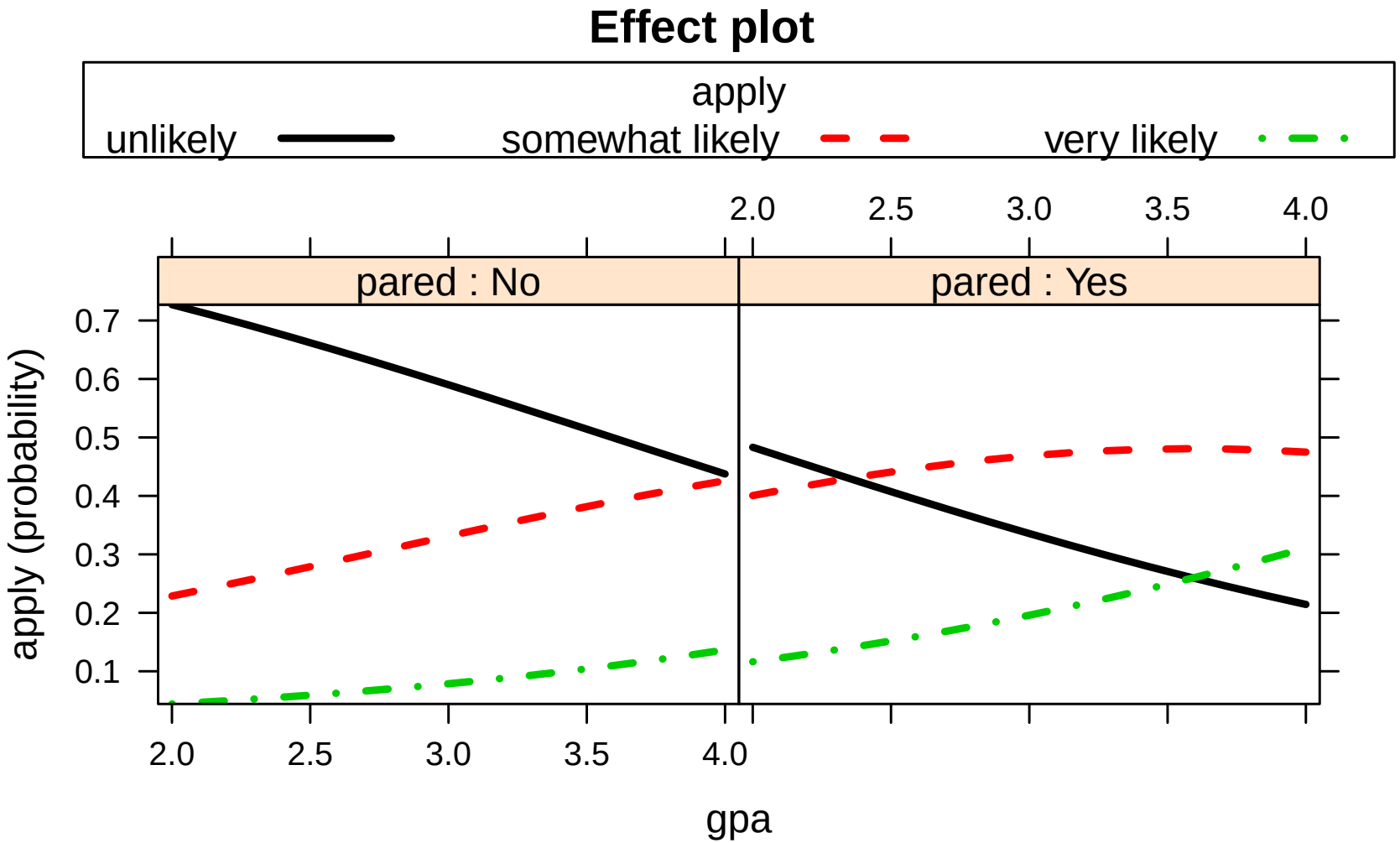
Three level variable called *apply*, with levels "unlikely", "somewhat likely", and "very likely", coded 1, 2, and 3, respectively, that we will use as our outcome variable. Three explanatory variables: *pared*, dummy variable indicating whether at least one parent has a graduate degree; *public*, dummy variable indicating whether undergrad college is public or private, and *gpa*, student's grade point average.



Regression example

```
formula: apply ~ pared + public + gpa
data:    dat

 link  threshold nobs logLik  AIC    niter max.grad cond.H
 logit flexible  400  -358.51 727.02 5(0)  1.63e-10 1.3e+03

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
paredYes       1.0477     0.2658    3.94  8.1e-05
publicPublic  -0.0587     0.2979   -0.20    0.844
gpa            0.6157     0.2606    2.36    0.018

Threshold coefficients:
                           Estimate Std. Error z value
unlikely|somewhat likely      2.203      0.780    2.83
somewhat likely|very likely   4.299      0.804    5.34
```

Effect plot


Effect plot
apply
unlikely
somewhat likely
very likely
pared : No
pared : Yes
apply (probability)
0.7
0.6
0.5
0.4
0.3
0.2
0.1
2.0
2.5
3.0
3.5
4.0
gpa

Test assumption

The "proportional odds" assumption is quite a strong one, so it's important to test it. The easiest way to do this is with the `nominal_test` function in the `ordinal` package.

	Df	logLik	AIC	LRT	Pr(>Chi)
	NA	-359	727	NA	NA
pared	1	-358	729	0.025	0.875
public	1	-357	725	3.883	0.049
gpa	1	-358	728	0.802	0.371

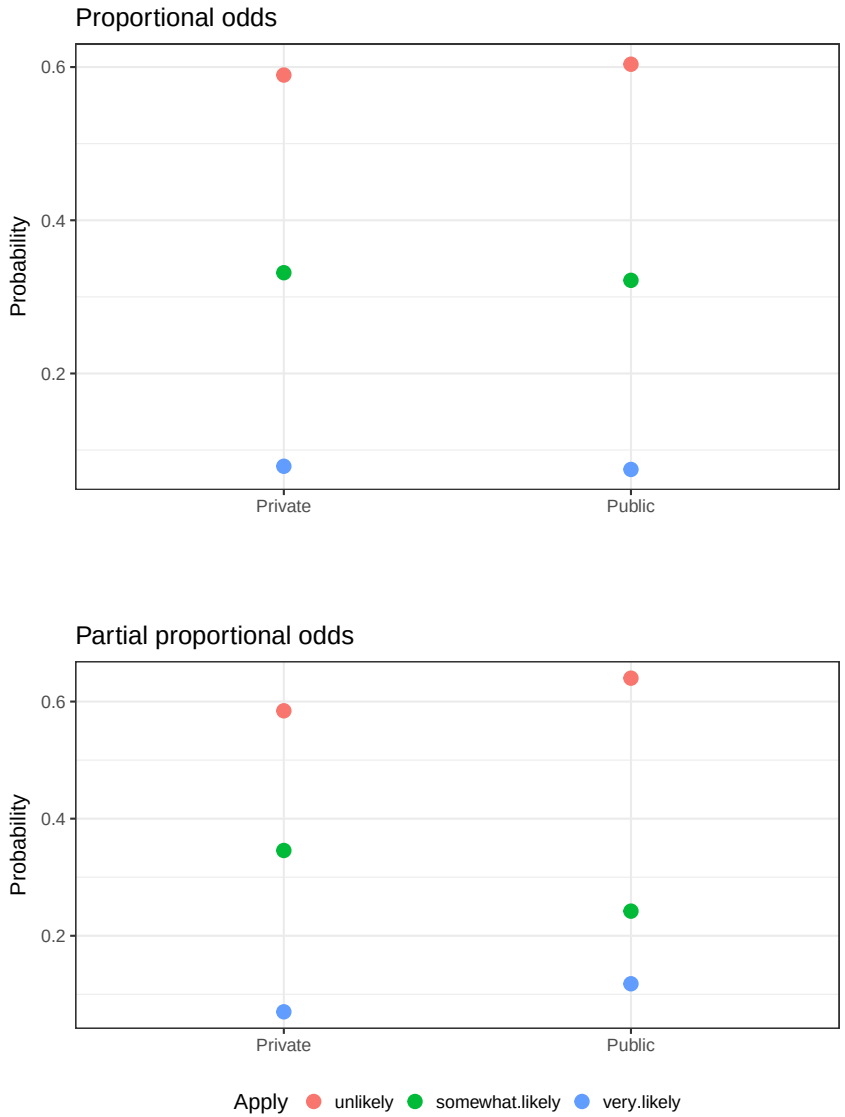
The likelihood ratio test can be thought of as a test of the hypothesis that relaxing the proportional odds assumption does not improve model fit. In this case, we can see evidence against the PO assumption for the `public` variable, so we can re-estimate the model as follows.

Partial proportional odds

```
o2 <- clm(apply ~ pared + gpa, nominal = ~ public, data = dat, Hess = TRUE)
summary(o2)
```

```
formula: apply ~ pared + gpa
nominal: ~public
data:    dat

 link  threshold nobs logLik  AIC    niter max.grad cond.H
 logit flexible  400  -356.57 725.14 5(0)  2.44e-09 1.3e+03

Coefficients:
         Estimate Std. Error z value Pr(>|z|)
paredYes    1.058      0.267    3.97  7.2e-05
gpa         0.611      0.261    2.34    0.019

Threshold coefficients:
                                        Estimate Std. Error z value
unlikely|somewhat likely.(Intercept)       2.166      0.780    2.78
somewhat likely|very likely.(Intercept)    4.411      0.809    5.45
unlikely|somewhat likely.publicPublic      0.235      0.305    0.77
somewhat likely|very likely.publicPublic  -0.573      0.411   -1.40
```

Effect plot


Proportional odds
Probability
0.6
0.4
0.2
Private
Public
Partial proportional odds
Probability
0.6
0.4
0.2
Private
Public
Apply
unlikely
somewhat.likely
very.likely

Multinomial logistic regression

Multinomial logistic regression

This method is used when an outcome variable consists of discrete but unordered categories. Common examples involve individuals making choices among a set of alternatives, such as the form of transport to commute to work, brand of toothpaste purchased, political party voted for, etc. The basic intuition is that we perform logistic regressions on each pair of alternatives as follows:

$$\log\left(\frac{p_a}{p_b}\right) = \beta_{1ab}(x_{1a} - x_{1b}) + \beta_{2ab}(x_{2a} - x_{2b}) + \cdots + \beta_{kab}(x_{ka} - x_{kb})$$

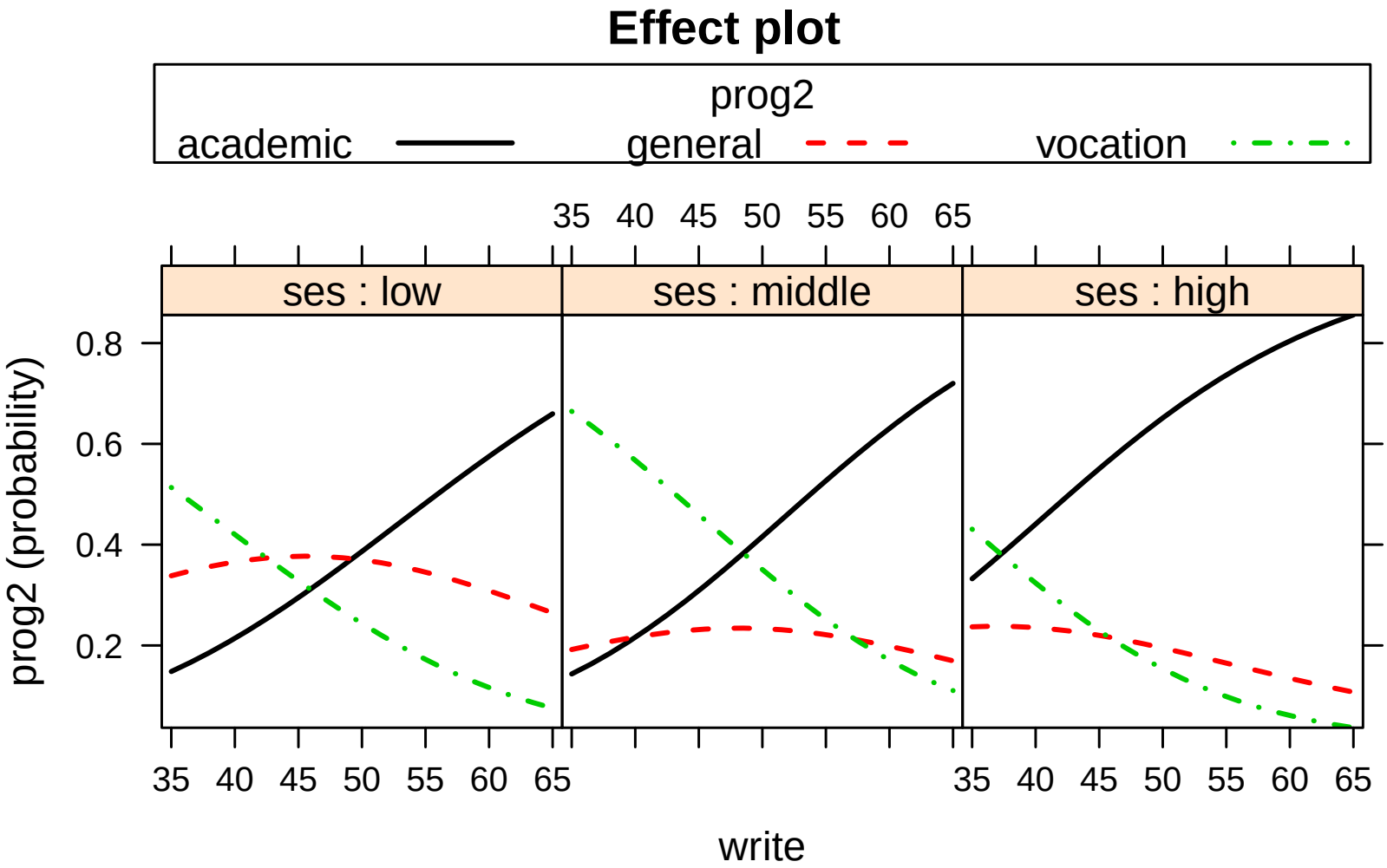
For example, the impact of variable x_1 (say, price) on choice of toothpaste brand depends on how different the price of brand a is compared with brand b. We get different parameter estimates for each pair of choices. Characteristics of individuals can also be included.

Example

The data are 200 high school students. Outcome variable is programme choice (general, academic or vocational). Explanatory variables are socio-economic status and writing test score.

```
# weights:  15 (8 variable)
initial  value 219.722458
iter  10 value 179.982880
final  value 179.981726
converged


Call:
multinom(formula = prog2 ~ ses + write, data = ml)

Coefficients:
         (Intercept) sesmiddle seshigh   write
general         2.85    -0.533  -1.163 -0.0579
vocation        5.22     0.291  -0.983 -0.1136

Std. Errors:
         (Intercept) sesmiddle seshigh  write
general         1.17     0.444   0.514 0.0214
vocation        1.16     0.476   0.596 0.0222

Residual Deviance: 360
AIC: 376
```

Effect plot


Effect plot
prog2
academic
general
vocation
ses : low
ses : middle
ses : high
prog2 (probability)
0.2
0.4
0.6
0.8
35 40 45 50 55 60 65
write