10 Dichotomous or binary responses

10.1 Introduction

Dichotomous or binary responses are widespread. Examples include being dead or alive, agreeing or disagreeing with a statement, and succeeding or failing to accomplish something. The responses are usually coded as 1 or 0, where 1 can be interpreted as the answer “yes” and 0 as the answer “no” to some question. For instance, in section 10.2, we will consider the employment status of women where the question is whether the women are employed.

We start by briefly reviewing single-level logistic and probit regression for dichotomous responses, formulating the models both as generalized linear models, as is common in statistics and biostatistics, and as latent-response models, which is common in econometrics and psychometrics. This lays the foundation for a discussion of various approaches for clustered dichotomous data. The focus will be on logistic regression with random effects, a special case of generalized linear mixed models. In this setting, the distinction between conditional or subject-specific effects and marginal or population-averaged effects is highlighted, and measures of dependence and heterogeneity are described.

We also discuss special features of statistical inference for generalized linear mixed models, including maximum likelihood (ML) estimation of model parameters, methods for assigning values to random effects, and how to obtain different kinds of predicted probabilities. This more technical material is provided here because the principles apply to all models discussed in this volume. However, you can skip it (sections 10.10 through 10.12) on first reading because it is not essential for understanding and interpreting the models.

Other approaches to clustered data with binary responses, such as fixed-effects (conditional ML) and generalized estimating equations (GEE) are briefly discussed in section 10.13.

10.2 Single-level logit and probit regression models for dichotomous responses

In this section, we will introduce logit and probit models without random effects that are appropriate for datasets without any kind of clustering. For simplicity, we will start by considering just one covariate x_i for unit (for example, subject) i. The models can be specified either as generalized linear models or as latent-response models. These two approaches and their relationship are described in sections 10.2.1 and 10.2.2.

10.2.1 Generalized linear model formulation

As in models for continuous responses, we are interested in the expectation (mean) of the response as a function of the covariate. The expectation of a binary (0 or 1) response is just the probability that the response is 1:

$$E(y_i|x_i) \;=\; \Pr(y_i = 1|x_i)$$

In linear regression, the conditional expectation of the response is modeled as a linear function $E(y_i|x_i) = \beta_1 + \beta_2 x_i$ of the covariate (see section 1.5). For dichotomous responses, this approach may be problematic because the probability must lie between 0 and 1, whereas regression lines increase (or decrease) indefinitely as the covariate increases (or decreases). Instead, a nonlinear function is specified in one of two ways:

$$\Pr(y_i = 1|x_i) \;=\; h(\beta_1 + \beta_2 x_i)$$

or

$$g\{\Pr(y_i = 1|x_i)\} \;=\; \beta_1 + \beta_2 x_i \equiv \nu_i$$

where ν_i (pronounced "nu") is referred to as the *linear predictor*. These two formulations are equivalent if the function $h(\cdot)$ is the inverse of the function $g(\cdot)$. Here $g(\cdot)$ is known as the *link function* and $h(\cdot)$ as the *inverse link function*, sometimes written as $g^{-1}(\cdot)$.

An appealing feature of generalized linear models is that they all involve a linear predictor resembling linear regression (without a residual error term). Therefore, we can handle categorical explanatory variables, interactions, and flexible curved relationships by using dummy variables, products of variables, and polynomials or splines, just as in linear regression.

Typical choices of link function for binary responses are the logit and probit links. In this section, we focus on the logit link, which is used for logistic regression, whereas both links are discussed in section 10.2.2. For the logit link, the model can be written as

$$\text{logit}\,\{\Pr(y_i = 1|x_i)\} \equiv \ln \underbrace{\left\{\frac{\Pr(y_i = 1|x_i)}{1 - \Pr(y_i = 1|x_i)}\right\}}_{\text{Odds}(y_i=1|x_i)} \;=\; \beta_1 + \beta_2 x_i \qquad (10.1)$$

where ln is the natural logarithm (base $e = 1.27$). The fraction in braces in (10.1) represents the odds that $y_i = 1$ given x_i, the expected number of 1 responses per 0 response or successes per failure. The odds *against*—or in other words, the expected number of failures per success—is the standard way of representing the chances *against* winning in gambling. It follows from (10.1) that the logit model can alternatively be expressed as an exponential function for the odds:

$$\text{Odds}(y_i = 1|x_i) \;=\; \exp(\beta_1 + \beta_2 x_i)$$

Because the relationship between odds and probabilities is

$$\text{Odds} = \frac{\text{Pr}}{1 - \text{Pr}} \qquad \text{and} \qquad \text{Pr} = \frac{\text{Odds}}{1 + \text{Odds}}$$

the probability that the response is 1 in the logit model is

$$\Pr(y_i = 1|x_i) \;=\; \text{logit}^{-1}(\beta_1 + \beta_2 x_i) \equiv \frac{\exp(\beta_1 + \beta_2 x_i)}{1 + \exp(\beta_1 + \beta_2 x_i)} \tag{10.2}$$

which is the inverse logit function (sometimes called logistic function) of the linear predictor.

We have introduced two components of a generalized linear model: the linear predictor and the link function. The third component is the distribution of the response given the covariates. Letting $\pi_i \equiv \Pr(y_i = 1|x_i)$, the distribution is specified as Bernoulli(π_i), or equivalently as binomial(1, π_i). There is no level-1 residual ϵ_i in (10.1), so the relationship between the probability and the covariate is deterministic. However, the responses are random because the covariate determines only the probability. Whether the response is 0 or 1 is the result of a Bernoulli trial. A Bernoulli trial can be thought of as tossing a biased coin with probability of heads equal to π_i. It follows from the Bernoulli distribution that the relationship between the conditional variance of the response and its conditional mean π_i, also known as the *variance function*, is $\text{Var}(y_i|x_i) = \pi_i(1 - \pi_i)$. (Including a residual ϵ_i in the linear predictor of binary regression models would lead to a model that is at best weakly identified[1] unless the residual is shared between units in a cluster, as in the multilevel models considered later in the chapter.)

The logit link is appealing because it produces a linear model for the log of the odds, implying a multiplicative model for the odds themselves. If we add one unit to x_i, we must add β_2 to the log odds or multiply the odds by $\exp(\beta_2)$. This can be seen by considering a one-unit change in x_i from some value a to $a + 1$. The corresponding change in the log odds is

$$\begin{aligned} \ln\{\text{Odds}(y_i = 1|x_i = a + 1)\} \;-\; & \ln\{\text{Odds}(y_i = 1|x_i = a)\} \\ & = \{\beta_1 + \beta_2(a + 1)\} - (\beta_1 + \beta_2 a) = \beta_2 \end{aligned}$$

Exponentiating both sides, we obtain the *odds ratio* (OR):

$$\begin{aligned} & \exp\big[\ln\{\text{Odds}(y_i = 1|x_i = a + 1)\} \;-\; \ln\{\text{Odds}(y_i = 1|x_i = a)\}\big] \\ &= \frac{\text{Odds}(y_i = 1|x_i = a + 1)}{\text{Odds}(y_i = 1|x_i = a)} = \frac{\Pr(y_i = 1|x_i = a + 1)}{\Pr(y_i = 0|x_i = a + 1)} \Big/ \frac{\Pr(y_i = 1|x_i = a)}{\Pr(y_i = 0|x_i = a)} \\ &= \exp(\beta_2) \end{aligned}$$

[1] Formally, the model is identified by functional form. For instance, if x_i is continuous, the level-1 variance has a subtle effect on the shape of the relationship between $\Pr(y_i = 1|x_i)$ and x_i. With a probit link, single-level models with residuals are not identified.

Consider the case where several covariates—for instance, x_{2i} and x_{3i}—are included in the model:

$$\text{logit}\,\{\Pr(y_i = 1|x_{2i}, x_{3i})\} \;=\; \beta_1 + \beta_2 x_{2i} + \beta_3 x_{3i}$$

In this case, $\exp(\beta_2)$ is interpreted as the OR comparing $x_{2i} = a+1$ with $x_{2i} = a$ for given x_{3i} (controlling or adjusting for x_{3i}), and $\exp(\beta_3)$ is the OR comparing $x_{3i} = a+1$ with $x_{3i} = a$ for given x_{2i}.

The predominant interpretation of the coefficients in logistic regression models is in terms of ORs, which is natural because the log odds is a *linear* function of the covariates. However, economists instead tend to interpret the coefficients in terms of marginal effects or partial effects on the response probability, which is a *nonlinear* function of the covariates. We relegate description of this approach to display 10.1, which may be skipped.

For a *continuous* covariate x_{2i}, economists often consider the partial derivative of the probability of success with respect to x_{2i}:

$$\Delta(x_{2i}|x_{3i}) \equiv \frac{\partial \Pr(y_i\!=\!1|x_{2i}, x_{3i})}{\partial x_{2i}} \;=\; \beta_2\,\frac{\exp(\beta_1 + \beta_2 x_{2i} + \beta_3 x_{3i})}{\{\exp(\beta_1 + \beta_2 x_i + \beta_3 x_{3i})\}^2}$$

A small change in x_{2i} hence produces a change of $\beta_2\,\frac{\exp(\beta_1+\beta_2 x_{2i}+\beta_3 x_{3i})}{\{\exp(\beta_1+\beta_2 x_{2i}+\beta_3 x_{3i})\}^2}$ in $\Pr(y_i = 1|x_{2i}, x_{3i})$. Unlike in linear models, where the partial effect simply becomes β_2, the derivative of the nonlinear logistic function is not constant but depends on x_{2i} and x_{3i}.

For a *binary* covariate x_{3i}, economists consider the difference in probabilities between groups,

$$\begin{aligned}\Delta(x_{3i}|x_{2i}) \;&\equiv\; \Pr(y_i\!=\!1|x_{2i}, x_{3i}\!=\!1) - \Pr(y_i\!=\!1|x_{2i}, x_{3i}\!=\!0)\\ &=\; \frac{\exp(\beta_1 + \beta_2 x_{2i} + \beta_3)}{1+\exp(\beta_1 + \beta_2 x_{2i} + \beta_3)} - \frac{\exp(\beta_1 + \beta_2 x_{2i})}{1+\exp(\beta_1 + \beta_2 x_{2i})}\end{aligned}$$

which, unlike linear models, depends on x_{2i}.

The partial effect at the average (PEA) is obtained by substituting the sample means $\overline{x}_{2\cdot} = \frac{1}{N}\sum_{i=1}^N x_{i2}$ and $\overline{x}_{3\cdot} = \frac{1}{N}\sum_{i=1}^N x_{i3}$ for x_{i2} and x_{i3}, respectively, in the above expressions. For binary covariates, the sample means are proportions, and subjects cannot be at the average (because the proportions are between 0 and 1).

The average partial effect (APE) overcomes this problem by taking the sample means of the individual partial effects, $\text{APE}(x_{2i}|x_{3i}) = \frac{1}{N}\sum_{i=1}^N \Delta(x_{2i}|x_{3i})$ and $\text{APE}(x_{3i}|x_{2i}) = \frac{1}{N}\sum_{i=1}^N \Delta(x_{3i}|x_{2i})$. Fortunately, the APE and PEA tend to be similar.

Display 10.1: Partial effects at the average (PEA) and average partial effects (APE) for the logistic regression model, $\text{logit}\,\{\Pr(y_i\!=\!1|x_{2i}, x_{3i})\} = \beta_1 + \beta_2 x_{2i} + \beta_3 x_{3i}$, where x_{2i} is continuous and x_{3i} is binary

Labor-participation data

To illustrate logistic regression, we will consider data on married women from the 1977 Canadian Women's Labor Force Participation Dataset used by Fox (1997):

```
. use https://www.stata-press.com/data/mlmus4/womenlf
```

The dataset `womenlf.dta` contains women's employment status and two explanatory variables:

- `workstat`: employment status
  (0: not working; 1: employed part time; 2: employed full time)
- `husbinc`: husband's income in $1,000
- `chilpres`: child present in household; dummy variable (0: absent; 1: present)

Fox (1997) considered a multiple logistic regression model for a woman being employed (full or part time) versus not working, with covariates `husbinc` and `chilpres`:

$$\text{logit}\{\Pr(y_i\!=\!1|\mathbf{x}_i)\} \;=\; \beta_1 + \beta_2 x_{2i} + \beta_3 x_{3i}$$

where $y_i\!=\!1$ denotes being employed, $y_i\!=\!0$ denotes not being employed, x_{2i} is `husbinc`, x_{3i} is `chilpres`, and $\mathbf{x}_i = (x_{2i}, x_{3i})'$ is a vector containing both covariates.

We first merge categories 1 and 2 (employed part time and full time) of `workstat` into a new category 1 for being employed:

```
. recode workstat 2=1
```

Estimation using logit

We then fit the model by ML using Stata's `logit` command:

```
. logit workstat husbinc i.chilpres
Logistic regression                                     Number of obs =    263
                                                        LR chi2(2)    =  36.42
                                                        Prob > chi2   = 0.0000
Log likelihood = -159.86627                             Pseudo R2     = 0.1023

------------------------------------------------------------------------------
    workstat | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
     husbinc |  -.0423084   .0197801    -2.14   0.032    -.0810768   -.0035401
             |
    chilpres |
     present |  -1.575648   .2922629    -5.39   0.000    -2.148473   -1.002824
       _cons |    1.33583   .3837634     3.48   0.000     .5836674    2.087992
------------------------------------------------------------------------------
```

Interpretation

The estimated coefficients are negative, so the estimated log odds of employment are lower if the husband earns more and if there is a child in the household. At the 5% significance level, we can reject the null hypotheses that the individual coefficients β_2 and β_3 are 0. The estimated coefficients and their estimated standard errors are also given in table 10.1.

Table 10.1: ML estimates for logistic regression model for women's labor force participation

	Est	(SE)	OR$=\exp(\beta)$	(95% CI)
β_1 [_cons]	1.34	(0.38)		
β_2 [husbinc]	−0.04	(0.02)	0.96	(0.92, 1.00)
β_3 [chilpres]	−1.58	(0.29)	0.21	(0.12, 0.37)

Instead of considering changes in log odds, it is more informative to obtain ORs, the exponentiated regression coefficients. This can be achieved using the `logit` command with the `or` option:

```
. logit workstat husbinc i.chilpres, or
Logistic regression                                     Number of obs =    263
                                                        LR chi2(2)    =  36.42
                                                        Prob > chi2   = 0.0000
Log likelihood = -159.86627                             Pseudo R2     = 0.1023

------------------------------------------------------------------------------
    workstat | Odds ratio   Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
     husbinc |   .9585741   .0189607    -2.14   0.032     .9221229    .9964662
             |
    chilpres |
     present |   .2068734   .0604614    -5.39   0.000     .1166621    .3668421
       _cons |    3.80315    1.45951     3.48   0.000     1.792601    8.068699
------------------------------------------------------------------------------
Note: _cons estimates baseline odds.
```

Comparing women with and without a child at home, whose husbands have the same income, the odds of working are estimated to be about 5 $(\approx 1/0.2068734)$ times as high for women who do not have a child at home as for women who do. Within these two groups of women, each \$1,000 increase in husband's income reduces the odds of working by an estimated 4% $\{-4\% = 100\%(0.9585741 - 1)\}$. Although this OR looks less important than the one for `chilpres`, remember that we cannot directly compare the magnitude of the two ORs. The OR for `chilpres` represents a comparison of two distinct groups of women, whereas the OR for `husbinc` merely expresses the effect of a \$1,000 increase in the husband's income. A \$10,000 increase would be associated with an OR of 0.66 $\{= 0.9585741^{10} = \exp(-0.0423084 \times 10)\}$.

The exponentiated intercept, estimated as 3.80, represents the odds of working for women who do not have a child at home and whose husbands' income is 0. This is not an OR as the column heading implies, but the odds when all covariates are 0. To avoid potential confusion, the exponentiated intercept was omitted from the output in earlier releases of Stata (until Stata 12.0) when the `or` option was used. As of Stata 15, a footnote is included to explain that this quantity represents the baseline odds, that is, the odds when all covariates are 0. The baseline odds is meaningful only if 0 is a possible value for all covariates.

In an attempt to make effects directly comparable and assess the relative importance of covariates, some researchers standardize all covariates to have standard deviation 1, thereby comparing the effects of a standard deviation change in each covariate. As discussed in section 1.5, there are many problems with such an approach, one of them being the meaningless notion of a standard deviation change in a dummy variable, such as `chilpres`.

The standard errors of exponentiated estimated regression coefficients should not be used for confidence intervals or hypothesis tests. Instead, the 95% confidence intervals in the above output were computed by taking the exponentials of the confidence limits for the regression coefficients β:

$$\exp\{\widehat{\beta} \pm 1.96 \times \mathrm{SE}(\widehat{\beta})\}$$

In table 10.1, we therefore report estimated ORs with 95% confidence intervals instead of standard errors.

To visualize the model, we can produce a plot of the predicted probabilities versus `husbinc`, with separate curves for women with and without children at home. Plugging in ML estimates for the parameters in (10.2), the predicted probability for woman i, often denoted $\widehat{\pi}_i$, is given by the inverse logit of the estimated linear predictor,

$$\widehat{\pi}_i \equiv \widehat{\Pr}(y_i = 1|x_i) \;=\; \frac{\exp(\widehat{\beta}_1 + \widehat{\beta}_2 x_{2i} + \widehat{\beta}_3 x_{3i})}{1 + \exp(\widehat{\beta}_1 + \widehat{\beta}_2 x_{2i} + \widehat{\beta}_3 x_{3i})} \;=\; \mathrm{logit}^{-1}(\widehat{\beta}_1 + \widehat{\beta}_2 x_{2i} + \widehat{\beta}_3 x_{3i}) \tag{10.3}$$

and can be obtained for the women in the dataset by using the `predict` command with the `pr` option:

```
. predict prob, pr
```

We can now produce the graph of predicted probabilities, shown in figure 10.1, by using

```
. twoway (line prob husbinc if chilpres==0, sort)
> (line prob husbinc if chilpres==1, sort lpatt(dash)),
> legend(order(1 "No child" 2 "Child"))
> xtitle("Husband´s income/$1000") ytitle("Probability that wife works")
```

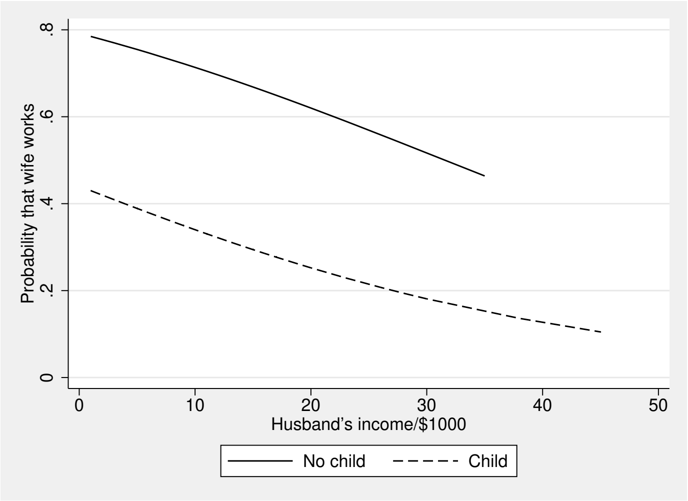


Figure 10.1: Predicted probability of working from logistic regression model (for range of husbinc in dataset)

The graph is similar to the graph of the predicted means from an analysis of covariance model (a linear regression model with a continuous and a dichotomous covariate; see section 1.7) except that the curves are not exactly straight. The curves have been plotted for the range of values of husbinc observed for the two groups of women, and for these ranges the predicted probabilities are nearly linear functions of husbinc.

Just for illustration, to see what the inverse logit function looks like, we will now plot the predicted probabilities for a widely extended range of values of husbinc (including negative values, although this does not make sense). This could be accomplished by inventing additional observations with more extreme values of husbinc and then using the predict command again. More conveniently, we can also use Stata's useful twoway plot type function:

```
. twoway (function y=invlogit(_b[husbinc]*x+_b[_cons]), range(-100 100))
> (function y=invlogit(_b[husbinc]*x+_b[1.chilpres]+_b[_cons]),
> range(-100 100) lpatt(dash)),
> xtitle("Husband´s income/$1000") ytitle("Probability that wife works")
> legend(order(1 "No child" 2 "Child")) xline(1) xline(45)
```

The estimated regression coefficients are referred to as _b[husbinc], _b[1.chilpres], and _b[_cons] (use the option coeflegend in the logit command to find out how to refer to estimated coefficients), and we used Stata's invlogit() function to obtain the predicted probabilities given in (10.3). The resulting graph is shown in figure 10.2.

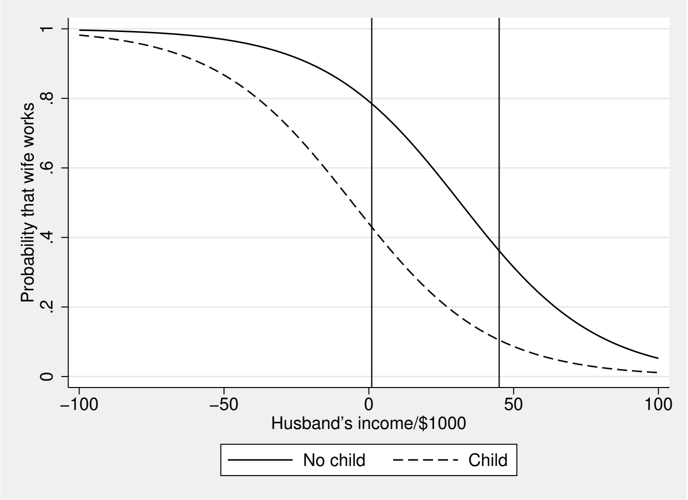


Figure 10.2: Illustration: Predicted probability of working from logistic regression model, extrapolated beyond the range of `husbinc` in the data

We could have alternatively used the `margins` and `marginsplot` commands to produce the same graph as above:

```
quietly margins chilpres, at(husbinc=(-100(10)100))
marginsplot, noci recast(line) plot2opts(lpatt(dash))
     legend(order(1 "No child" 2 "Child"))
     ytitle("Probability that wife works") xline(1) xline(45)
```

The range of `husbinc` actually observed in the data lies approximately between the two vertical lines. It would not be safe to rely on predicted probabilities extrapolated outside this range. The curves are approximately linear in the region where the linear predictor is close to 0 (and the predicted probability is close to 0.5) and then flatten as the linear predictor becomes extreme. This flattening ensures that the predicted probabilities remain in the permitted interval from 0 to 1.

Estimation using glm

We can fit the same model by using the `glm` command for generalized linear models. The syntax is the same as that of the `logit` command except that we must specify the logit link function in the `link()` option and the binomial distribution in the `family()` option:

```
. glm workstat husbinc i.chilpres, link(logit) family(binomial)

Generalized linear models                         Number of obs   =        263
Optimization     : ML                             Residual df     =        260
                                                  Scale parameter =          1
Deviance         =  319.7325378                   (1/df) Deviance =   1.229741
Pearson          =  265.9615312                   (1/df) Pearson  =   1.022929

Variance function: V(u) = u*(1-u)                 [Bernoulli]
Link function    : g(u) = ln(u/(1-u))             [Logit]

                                                  AIC             =   1.238527
Log likelihood   = -159.8662689                   BIC             =  -1129.028

------------------------------------------------------------------------------
             |                 OIM
    workstat | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
     husbinc |  -.0423084   .0197801    -2.14   0.032    -.0810768   -.0035401
             |
    chilpres |
    present  |  -1.575648   .2922629    -5.39   0.000    -2.148473   -1.002824
       _cons |    1.33583   .3837634     3.48   0.000     .5836674    2.087992
------------------------------------------------------------------------------
```

To obtain estimated ORs, we use the `eform` option (for "exponentiated form"), and to fit a probit model, we simply change the `link(logit)` option to `link(probit)`.

10.2.2 Latent-response formulation

The logistic regression model and other models for dichotomous responses can also be viewed as latent-response models. Underlying the observed dichotomous response y_i (whether the woman works or not), we imagine that there is an unobserved or latent continuous response y_i^* representing the propensity to work or the excess utility of working as compared with not working. If this latent response is greater than 0, then the observed response is 1; otherwise, the observed response is 0:

$$y_i = \begin{cases} 1 & \text{if } y_i^* > 0 \\ 0 & \text{otherwise} \end{cases}$$

For simplicity, we will assume that there is one covariate x_i. A linear regression model is then specified for the latent response y_i^*,

$$y_i^* = \beta_1 + \beta_2 x_i + \epsilon_i$$

where ϵ_i is a residual error term, assumed to be independent of x_i and independent across women. As we will see, the model for the observed binary response y_i becomes a probit model if ϵ_i has a standard normal distribution and a logit model if ϵ_i has a standard logistic distribution.

The latent-response formulation has been used in various disciplines and applications. In genetics, where y_i is often a phenotype or qualitative trait, y_i^* is called a *liability*. For attitudes measured by agreement or disagreement with statements, the latent response can be thought of as a "sentiment" in favor of the statement. In economics, the latent response is often called an *index function*. In discrete-choice settings (see chapter 12), y_i^* is the *difference in utilities* between alternatives.

Figure 10.3 illustrates the relationship between the latent-response formulation, shown in the lower graph, and the generalized linear model formulation, shown in the upper graph in terms of a curve for the conditional probability that $y_i\!=\!1$. The regression line in the lower graph represents the conditional expectation of y_i^* given x_i as a function of x_i, and the density curves represent the conditional distributions of y_i^* given x_i. The dotted horizontal line at $y_i^*\!=\!0$ represents the threshold, so $y_i\!=\!1$ if y_i^* exceeds the threshold and $y_i\!=\!0$ otherwise. Therefore, the areas under the parts of the density curves that lie above the dotted line, here shaded gray, represent the probabilities that $y_i\!=\!1$ given x_i. For the value of x_i indicated by the vertical dotted line, the mean of y_i^* is 0; therefore, half the area under the density curve lies above the threshold, and the conditional probability that $y_i\!=\!1$ equals 0.5 at that point.

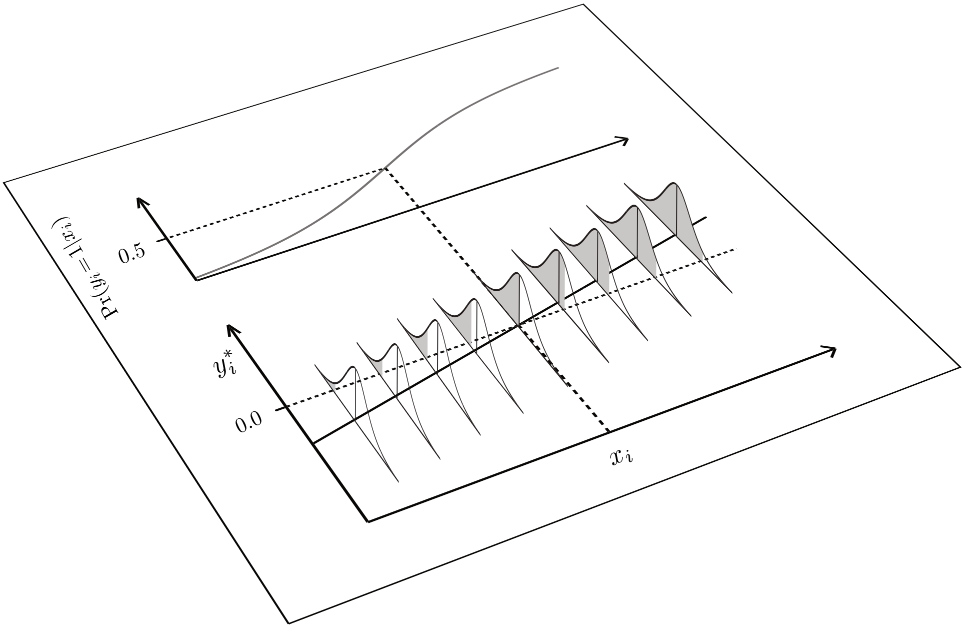


Figure 10.3: Illustration of equivalence of latent-response and generalized linear model formulations for logistic regression

We can derive the probability curve from the latent-response formulation as follows:

$$\begin{aligned}\Pr(y_i\!=\!1|x_i) &= \Pr(y_i^* > 0|x_i) \;=\; \Pr(\beta_1 + \beta_2 x_i + \epsilon_i > 0|x_i)\\ &= \Pr\{\epsilon_i > -(\beta_1 + \beta_2 x_i)|x_i\} \;=\; \Pr(-\epsilon_i \le \beta_1 + \beta_2 x_i|x_i)\\ &= F(\beta_1 + \beta_2 x_i)\end{aligned}$$

where $F(\cdot)$ is the cumulative density function of $-\epsilon_i$, or the area under the density curve for $-\epsilon_i$ from minus infinity to $\beta_1 + \beta_2 x_i$. If the density of ϵ_i is symmetric, the cumulative density function of $-\epsilon_i$ is the same as that of ϵ_i.

Logistic regression

In logistic regression, ϵ_i (and hence $-\epsilon_i$) is assumed to have a standard logistic cumulative density function given x_i,

$$\Pr(\epsilon_i < \tau | x_i) \; = \; \frac{\exp(\tau)}{1 + \exp(\tau)}$$

For this distribution, ϵ_i has mean 0 and variance $\pi^2/3 \approx 3.29$ (note that π here represents the famous mathematical constant pronounced "pi", the circumference of a circle divided by its diameter).

Probit regression

When a latent-response formulation is used, it seems natural to assume that ϵ_i (and hence $-\epsilon_i$) has a normal distribution given x_i, as is typically done in linear regression. If a standard (mean 0 and variance 1) normal distribution is assumed, the model becomes a probit model,

$$\Pr(y_i = 1 | x_i) \; = \; F(\beta_1 + \beta_2 x_i) = \; \Phi(\beta_1 + \beta_2 x_i) \tag{10.4}$$

Here $\Phi(\cdot)$ is the standard normal cumulative distribution function, the probability that a standard normally distributed random variable (here ϵ_i) is less than the argument. For example, when $\beta_1 + \beta_2 x_i$ equals 1.96, $\Phi(\beta_1 + \beta_2 x_i)$ equals 0.975. $\Phi(\cdot)$ is the inverse link function $h(\cdot)$, whereas the link function $g(\cdot)$ is $\Phi^{-1}(\cdot)$, the inverse standard normal cumulative distribution function, called the *probit link* function [the Stata function for $\Phi^{-1}(\cdot)$ is `invnormal()`].

To understand why a *standard* normal distribution is specified for ϵ_i, with the variance θ fixed at 1, consider the graph in figure 10.4. On the left, the standard deviation is 1, whereas the standard deviation on the right is 2. However, by doubling the slope of the regression line for y_i^* on the right (without changing the point where it intersects the threshold 0), we obtain the same curve for the probability that $y_i = 1$. Because we can obtain equivalent models by increasing both the standard deviation and the slope by the same multiplicative factor, the model with a freely estimated standard deviation is not identified.

This lack of identification is also evident from inspecting the expression for the probability if the variance θ were not fixed at 1. If $\epsilon_i \sim N(0, \theta)$, then $\epsilon_i/\sqrt{\theta} \sim N(0, 1)$, so (10.4) changes as follows:

$$\Pr(y_i = 1 | x_i) \; = \; \Pr(\epsilon_i \le \beta_1 + \beta_2 x_i) \; = \; \Pr\left(\frac{\epsilon_i}{\sqrt{\theta}} \le \frac{\beta_1 + \beta_2 x_i}{\sqrt{\theta}}\right) = \Phi\left(\frac{\beta_1}{\sqrt{\theta}} + \frac{\beta_2}{\sqrt{\theta}} x_i\right)$$

It is now easy to see that multiplication of the regression coefficients by a constant can be counteracted by multiplying $\sqrt{\theta}$ by the same constant. This is the reason for fixing the standard deviation in probit models to 1 (see also exercise 10.10). The variance of ϵ_i in logistic regression is also fixed but to a larger value, $\pi^2/3$.

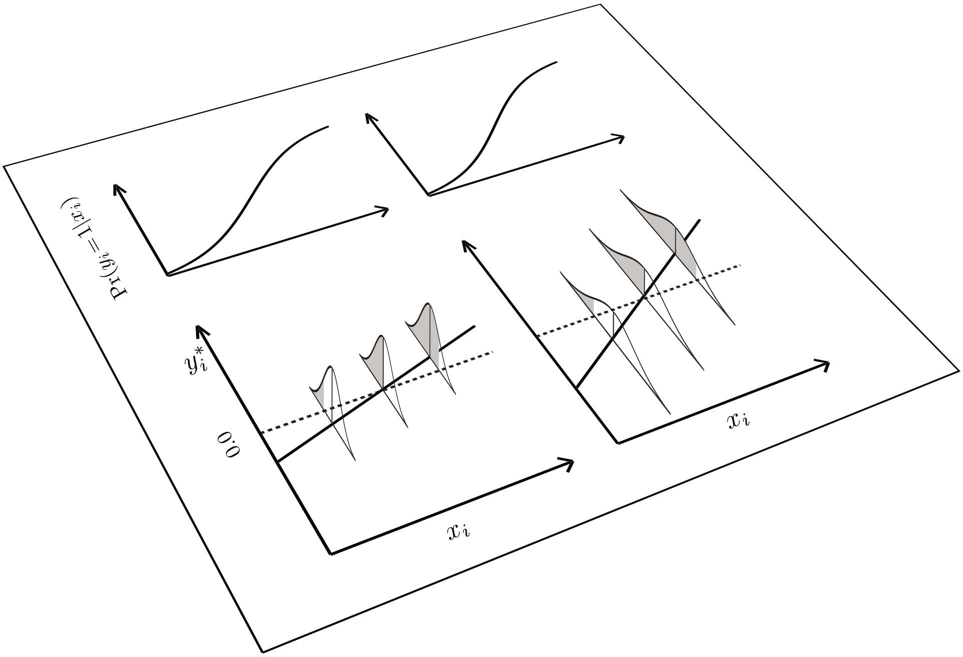


Figure 10.4: Illustration of equivalence between probit models with change in residual standard deviation counteracted by change in slope

Estimation using probit

A probit model can be fit to the women's employment data in Stata by using:

```
. probit workstat husbinc i.chilpres

Probit regression                                       Number of obs =    263
                                                        LR chi2(2)    =  36.19
                                                        Prob > chi2   = 0.0000
Log likelihood = -159.97986                             Pseudo R2     = 0.1016

------------------------------------------------------------------------------
    workstat | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
     husbinc |  -.0242081   .0114252    -2.12   0.034    -.0466011    -.001815
             |
    chilpres |
    present  |  -.9706164   .1769051    -5.49   0.000    -1.317344   -.6238887
       _cons |   .7981507   .2240082     3.56   0.000     .3591028    1.237199
------------------------------------------------------------------------------
```

Interpretation

These estimates are closer to 0 than those reported for the logit model in table 10.1 because the standard deviation of ϵ_i is 1 for the probit model and $\pi/\sqrt{3} \approx 1.81$ for the logit model. Therefore, as we have already seen in figure 10.4, the regression coefficients in logit models must be larger in absolute value to produce nearly the same curve for the conditional probability that $y_i = 1$. Here we say "nearly the same" because the shapes of the probit and logit curves are similar yet not identical. To visualize the subtle difference in shape, we can plot the predicted probabilities for women without children at home from both the logit and the probit models:

```
. twoway (function y=invlogit(1.3358-0.0423*x), range(-100 100))
> (function y=normal(0.7982-0.0242*x), range(-100 100) lpatt(dash)),
> xtitle("Husband´s income/$1000") ytitle("Probability that wife works")
> legend(order(1 "Logit link" 2 "Probit link")) xline(1) xline(45)
```

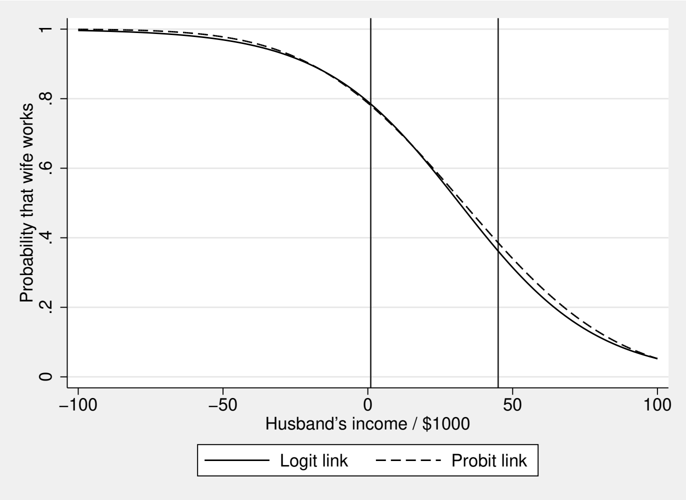


Figure 10.5: Predicted probabilities of working from logistic and probit regression models for women without children at home

Here the predictions from the models coincide nearly perfectly in the region where most of the data are concentrated and are very similar elsewhere. It is thus futile to attempt to empirically distinguish between the logit and probit links unless one has a huge sample.

Regression coefficients in probit models cannot be interpreted in terms of ORs as in logistic regression models. Instead, the coefficients can be interpreted as differences in the population means of the *latent response* y_i^*, controlling or adjusting for other covariates (the same kind of interpretation can also be made in logistic regression). Many people find interpretation based on latent responses less appealing than interpretation using ORs, because the latter refer to observed responses y_i. Alternatively, the coefficients can be interpreted in terms of average partial effects or partial effects at the average as shown for logit models[2] in display 10.1.

10.3 Which treatment is best for toenail infection?

Lesaffre and Spiessens (2001) analyzed data provided by De Backer et al. (1998) from a randomized, double-blind trial of treatments for toenail infection (dermatophyte onychomycosis). Toenail infection is not uncommon, with a prevalence of about 2% to 3% in the U.S. and a much higher prevalence among diabetics and the elderly. The infection is caused by a fungus that not only disfigures the nails but also can cause physical pain and impair the ability to work.

In this clinical trial, 378 patients were randomly allocated into two oral antifungal treatments (250 mg/day terbinafine and 200 mg/day itraconazole) and evaluated at seven visits, at weeks 0, 4, 8, 12, 24, 36, and 48. One outcome is onycholysis, the degree of separation of the nail plate from the nail bed, which was dichotomized ("moderate or severe" versus "none or mild") and is available for 294 patients.

The dataset `toenail.dta` contains the following variables:

- `patient`: patient identifier
- `outcome`: onycholysis (separation of nail plate from nail bed)
  (0: none or mild; 1: moderate or severe)
- `treatment`: treatment group (0: itraconazole; 1: terbinafine)
- `visit`: visit number (1, 2, ..., 7)
- `month`: exact timing of visit in months

We read in the toenail data by typing

```
. use https://www.stata-press.com/data/mlmus4/toenail, clear
```

The main research question is whether the treatments differ in their efficacy. In other words, do patients receiving one treatment experience a greater decrease in their probability of having onycholysis than those receiving the other treatment?

10.4 Longitudinal data structure

Before investigating the research question, we should look at the longitudinal structure of the toenail data by using, for instance, the `xtdescribe`, `xtsum`, and `xttab` com-

[2] For probit models with continuous x_{2i} and binary x_{3i}, $\Delta(x_{2i}|x_{3i}) = \beta_2\,\phi(\beta_1 + \beta_2 x_{2i} + \beta_3 x_{3i})$, where $\phi(\cdot)$ is the density function of the standard normal distribution, and $\Delta(x_{3i}|x_{2i}) = \Phi(\beta_1 + \beta_2 x_{2i} + \beta_3) - \Phi(\beta_1 + \beta_2 x_{2i})$.

mands, discussed in *Part III: Introduction to models for longitudinal and panel data* (in volume 1).

Here we illustrate the use of the `xtdescribe` command. Before using `xtdescribe`, we `xtset` the data with `patient` as the cluster identifier and `visit` as the time variable:

```
. xtset patient visit

Panel variable: patient (unbalanced)
 Time variable: visit, 1 to 7, but with gaps
         Delta: 1 unit
```

The output states that the data are unbalanced and that there are gaps. We would describe the time variable `visit` as *fixed occasions* because the values are identical across patients apart from the gaps caused by missing data; see *Part III: Introduction to models for longitudinal and panel data* (in volume 1). In contrast, `month` is a varying-occasion time variable that takes on a unique set of values for each individual. To use `xtdescribe`, it is important to `xtset` the data with a fixed-occasion variable like `visit` so that it is clear when data are missing for individuals:

```
. xtdescribe if !missing(outcome)

 patient:  1, 2, ..., 383                                   n =        294
   visit:  1, 2, ..., 7                                     T =          7
           Delta(visit) = 1 unit
           Span(visit)  = 7 periods
           (patient*visit uniquely identifies each observation)

Distribution of T_i:   min      5%     25%       50%       75%     95%     max
                         1       3       7         7         7       7       7

     Freq.  Percent    Cum. |  Pattern
 ---------------------------+---------
      224     76.19   76.19 |  1111111
       21      7.14   83.33 |  11111.1
       10      3.40   86.73 |  1111.11
        6      2.04   88.78 |  111....
        5      1.70   90.48 |  1......
        5      1.70   92.18 |  11111..
        4      1.36   93.54 |  1111...
        3      1.02   94.56 |  11.....
        3      1.02   95.58 |  111.111
       13      4.42  100.00 | (other patterns)
 ---------------------------+---------
      294    100.00         |  XXXXXXX
```

We see that 224 patients have complete data (the pattern “`1111111`”), 21 patients missed the sixth visit (“`11111.1`”), 10 patients missed the fifth visit (“`1111.11`”), and most other patients dropped out at some point, never returning after missing a visit. The latter pattern is sometimes referred to as *monotone missingness*, in contrast with *intermittent missingness*, which follows no particular pattern.

As discussed in section 5.9, a nice feature of ML estimation for incomplete data such as these is that all information is used. Thus, not only patients who attended all visits but also patients with missing visits contribute information. If the model is correctly specified, ML estimates are consistent when the responses are missing at random (MAR).

10.5 Proportions and fitted population-averaged or marginal probabilities

A useful graphical display of the data is a bar plot showing the proportion of patients with onycholysis at each visit by treatment group. The following Stata commands can be used to produce the graph shown in figure 10.6:

```
. label define tr 0 "Itraconazole" 1 "Terbinafine"
. label values treatment tr
. graph bar (mean) proportion = outcome, over(visit) by(treatment)
> ytitle(Proportion with onycholysis)
```

Here we defined value labels for `treatment` to make them appear on the graph.

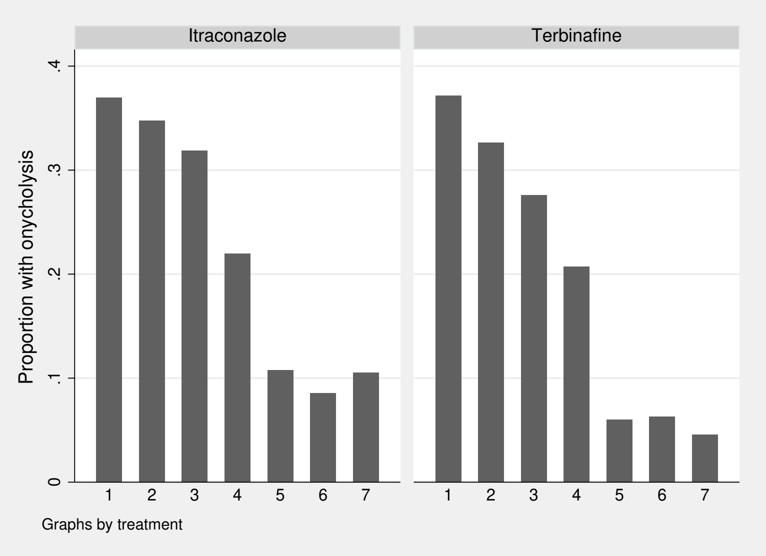


Figure 10.6: Bar plot of proportion of patients with toenail infection by visit and treatment group

An alternative display is a line graph, plotting the observed proportions at each visit against time. The most natural time scale would be `month`. However, the exact timing in months is unique for each patient, so we cannot estimate proportions for each timing. We therefore use the proportions associated with visit number `visit` and find the average time (`month`) associated with each visit number. Both variables can be obtained using the `egen` command with the `mean()` function:

```
. egen prop = mean(outcome), by(treatment visit)
. egen mn_month = mean(month), by(treatment visit)
. twoway line prop mn_month, by(treatment) sort
> xtitle(Time in months) ytitle(Proportion with onycholysis)
```

The resulting graph is shown in figure 10.7.

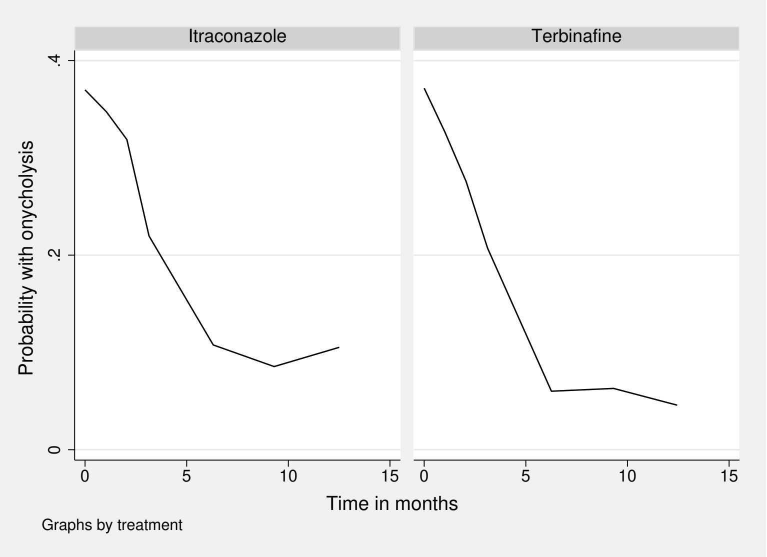


Figure 10.7: Line plot of proportion of patients with toenail infection by average time at visit and treatment group

The proportions shown in figure 10.7 represent the estimated average (or marginal) probabilities of onycholysis given the two covariates, time since randomization and treatment group. We are not attempting to estimate individual patients' personal probabilities, which may vary substantially, but are considering the population averages given the covariates.

Instead of estimating the probabilities for each combination of `visit` and `treatment`, we can attempt to obtain smooth curves of the estimated probability as a function of time. We then no longer have to group observations for the same visit number together—we can use the exact timing of the visits directly. One way to accomplish this is by using a logistic regression model with `month`, `treatment`, and their interaction as covariates. This model for the dichotomous outcome y_{ij} at visit i for patient j can be written as

$$\text{logit}\{\Pr(y_{ij}\!=\!1|\mathbf{x}_{ij})\} \;=\; \beta_1 + \beta_2 x_{2j} + \beta_3 x_{3ij} + \beta_4 x_{2j} x_{3ij} \tag{10.5}$$

where x_{2j} represents `treatment`, x_{3ij} represents `month`, and $\mathbf{x}_{ij} = (x_{2j}, x_{3ij})'$ is a vector containing both covariates. This model allows for a difference between groups at baseline β_2, and linear changes in the log odds of onycholysis over time with slope β_3 in the itraconazole group and slope $\beta_3 + \beta_4$ in the terbinafine group. Therefore, β_4, the difference in the rate of improvement (on the log odds per month scale) between treatment groups, can be viewed as the treatment effect (terbinafine versus itraconazole).

This model makes the unrealistic assumption that the responses for a given patient are conditionally independent after controlling for the included covariates. We will relax this assumption in the next section. Here we can get satisfactory inferences for marginal effects by using robust standard errors for clustered data instead of model-based standard errors. This approach is analogous to pooled OLS in linear models and corresponds to the generalized estimating equations (GEE) approach discussed in section 6.6 with an independence working correlation structure (see section 10.13.2 for an example with a different working correlation structure).

Estimation using logit

We fit the model by ML with cluster–robust standard errors:

```
. logit outcome i.treatment##c.month, or vce(cluster patient)
Logistic regression                                     Number of obs =  1,908
                                                        Wald chi2(3)  =  64.30
                                                        Prob > chi2   = 0.0000
Log pseudolikelihood = -908.00747                       Pseudo R2     = 0.0830
                              (Std. err. adjusted for 294 clusters in patient)
------------------------------------------------------------------------------
             |               Robust
     outcome | Odds ratio   std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
   treatment |
Terbinafine  |   .9994184   .2511294    -0.00   0.998     .6107468    1.635436
       month |   .8434052   .0246377    -5.83   0.000     .7964725    .8931034
             |
  treatment# |
     c.month |
Terbinafine  |    .934988   .0488105    -1.29   0.198     .8440528     1.03572
             |
       _cons |   .5731389   .0982719    -3.25   0.001     .4095534    .8020642
------------------------------------------------------------------------------
Note: _cons estimates baseline odds.
```

Note that, as discussed in display 2.1 in volume 1, cluster–robust standard errors (based on a sandwich estimator) work only if there is a sufficiently large number of clusters. Inspired by Angrist and Pischke (2009, 319), we use the rule of thumb that the number of clusters minus the number of cluster-level covariates q should be at least 42.

We will leave interpretation of the estimates for later and first check how well the predicted probabilities from the logistic regression model correspond to the observed proportions in figure 10.7. The predicted probabilities are obtained and plotted together with the observed proportions by using the following commands, which result in figure 10.8:

```
. predict prob, pr
. twoway (line prop mn_month, sort) (line prob month, sort lpatt(dash)),
> by(treatment) legend(order(1 "Observed proportions" 2 "Fitted probabilities"))
> xtitle(Time in months) ytitle(Probability of onycholysis)
```

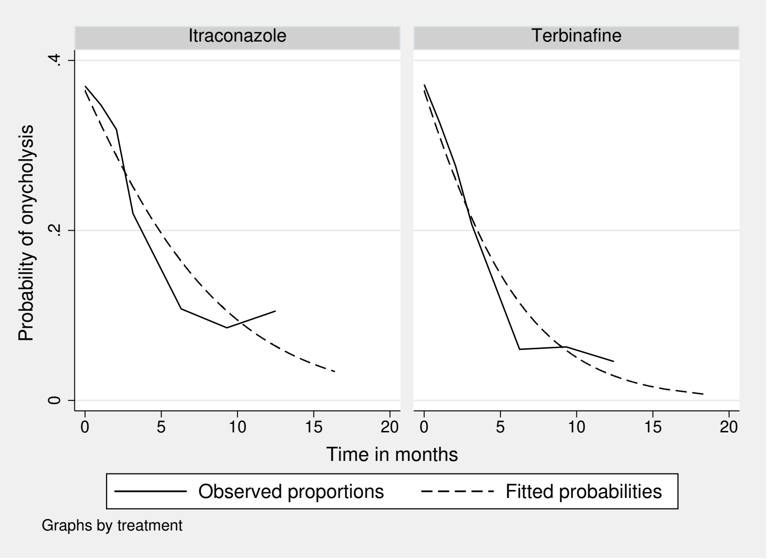


Figure 10.8: Proportions and fitted probabilities using ordinary logistic regression

The marginal probabilities predicted by the model fit the observed proportions reasonably well. However, we have treated the dependence among responses for the same patient as a nuisance by fitting an ordinary logistic regression model with robust standard errors for clustered data. We now add random effects to accommodate the dependence and estimate the degree of dependence instead of treating it as a nuisance.

10.6 Random-intercept logistic regression

10.6.1 Model specification

Reduced-form specification

To relax the assumption of conditional independence among the responses for the same patient given the covariates, we can include a patient-specific random intercept ζ_j in the linear predictor to obtain a random-intercept logistic regression model:

$$\text{logit}\{\Pr(y_{ij}=1|\mathbf{x}_{ij},\zeta_j)\} \;=\; \beta_1+\beta_2x_{2j}+\beta_3x_{3ij}+\beta_4x_{2j}x_{3ij}+\zeta_j \qquad (10.6)$$

The random intercepts are assumed to be independently normally distributed given the covariates, $\zeta_j|\mathbf{x}_{ij} \sim N(0,\psi)$. Given ζ_j and $\mathbf{x}_{ij}$, the responses y_{ij} for patient j at different occasions i are independently Bernoulli distributed. To write this down more formally, it is useful to define $\pi_{ij} \equiv \Pr(y_{ij}|\mathbf{x}_{ij},\zeta_j)$, giving

$$\begin{aligned}\text{logit}(\pi_{ij}) &= \beta_1+\beta_2x_{2j}+\beta_3x_{3ij}+\beta_4x_{2j}x_{3ij}+\zeta_j\\ y_{ij}|\pi_{ij} &\sim \text{Binomial}(1,\pi_{ij})\end{aligned}$$

This is a simple example of a *generalized linear mixed model* (GLMM) because it is a generalized linear model with both fixed effects β_1 to β_4 and a random effect ζ_j. The model is also sometimes referred to as a hierarchical generalized linear model (HGLM) in contrast to a hierarchical linear model (HLM). The random intercept can be thought of as the combined effect of omitted patient-specific (time-constant) covariates that cause some patients to be more prone to onycholysis than others (more precisely, the component of this combined effect that is independent of the covariates in the model—not an issue if the covariates are exogenous). It is appealing to model this unobserved heterogeneity in the same way as observed heterogeneity by simply adding the random intercept to the linear predictor. As we will explain later, be aware that ORs obtained by exponentiating regression coefficients in this model must be interpreted conditionally on the random intercept and are therefore often referred to as conditional or subject-specific ORs.

Using the latent-response formulation, the model can equivalently be written as

$$y^*_{ij} \;=\; \beta_1+\beta_2x_{2j}+\beta_3x_{3ij}+\beta_4x_{2j}x_{3ij}+\zeta_j+\epsilon_{ij} \qquad (10.7)$$

where $\zeta_j|\mathbf{x}_{ij} \sim N(0,\psi)$ and the $\epsilon_{ij}|\mathbf{x}_{ij},\zeta_j$ have independent standard logistic distributions. The binary responses y_{ij} are determined by the latent continuous responses via the threshold model

$$y_{ij} \;=\; \begin{cases} 1 & \text{if } y^*_{ij}>0\\ 0 & \text{otherwise}\end{cases}$$

Confusingly, logistic random-effects models are sometimes written as $y_{ij}=\pi_{ij}+e_{ij}$, where e_{ij} is a normally distributed level-1 residual with variance $\pi_{ij}(1-\pi_{ij})$. This formulation is clearly incorrect because such a model does not produce binary responses (see Skrondal and Rabe-Hesketh [2007]).

Two-stage formulation

Raudenbush and Bryk (2002) and others write two-level models in terms of a level-1 model and one or more level-2 models (see section 4.9 of volume I). In generalized linear mixed models, the need to specify a link function and distribution leads to two further stages of model specification.

Using the notation and terminology of Raudenbush and Bryk (2002), the level-1 sampling model, link function, and structural model are written as

$$\begin{aligned} y_{ij} &\sim \text{Bernoulli}(\varphi_{ij}) \\ \text{logit}(\varphi_{ij}) &= \eta_{ij} \\ \eta_{ij} &= \beta_{0j} + \beta_{1j}x_{2j} + \beta_{2j}x_{3ij} + \beta_{3j}x_{2j}x_{3ij} \end{aligned}$$

respectively.

The level-2 model for the intercept β_{0j} is written as

$$\beta_{0j} = \gamma_{00} + u_{0j}$$

where γ_{00} is a fixed intercept and u_{0j} is a residual or random intercept. The level-2 models for the coefficients β_{1j}, β_{2j}, and β_{3j} have no residuals for a random-intercept model,

$$\beta_{pj} = \gamma_{p0}, \quad p = 1, 2, 3$$

Plugging the level-2 models into the level-1 structural model, we obtain

$$\begin{aligned} \eta_{ij} &= \gamma_{00} + u_{0j} + \gamma_{01}x_{2j} + \gamma_{02}x_{3ij} + \gamma_{03}x_{2j}x_{3ij} \\ &\equiv \beta_1 + \zeta_{0j} + \beta_2 x_{2j} + \beta_3 x_{3ij} + \beta_4 x_{2j}x_{3ij} \end{aligned}$$

Equivalent models can be specified using either the reduced-form formulation (used for instance by Stata) or the two-stage formulation (used in the HLM software of Raudenbush et al. 2019). However, in practice, what models are being considered is to some extent influenced by the approach adopted as discussed in section 4.9.

10.6.2 Model assumptions

It is assumed that the ζ_j are independent across patients and independent of the covariates $\mathbf{x}_{ij}$ at occasion i. It is also assumed that the covariates at other occasions do not affect the response probabilities given the random intercept (called strict exogeneity conditional on the random intercept). For the latent response formulation, the ϵ_{ij} are assumed to be independent across both occasions and patients, and independent of both ζ_j and $\mathbf{x}_{ij}$. In the generalized linear model formulation, the analogous assumptions are implicit in assuming that the responses are independently Bernoulli distributed (with probabilities determined by ζ_j and $\mathbf{x}_{ij}$).

In contrast to linear random-effects models, consistent estimation in random-effects logistic regression requires that the random part of the model is correctly specified in addition to the fixed part. Specifically, consistency formally requires (1) a correct linear predictor (such as including relevant interactions), (2) a correct link function, (3) correct specification of covariates having random coefficients, (4) conditional independence of responses given the random effects and covariates, (5) independence of the random effects from covariates (for causal inference), and (6) normally distributed random effects. Hence, the assumptions are stronger than those discussed for linear models in section 3.3.2. However, for regression coefficients, the normality assumption for the random intercepts seems to be rather innocuous (McCulloch and Neuhaus 2011) in contrast to the assumption of independence between the random intercepts and covariates (Heagerty and Kurland 2001). As in standard logistic regression, the ML estimator is not necessarily unbiased in finite samples even if all the assumptions are true.

Although the sandwich estimator of the standard errors of regression coefficients is robust to misspecification (consistent as the number of clusters becomes large), it may be less useful than in linear models because in logistic regression misspecification also leads to inconsistent *point* estimates. For example, a reason for using the sandwich estimator in linear models is to protect against violation of the constant variance assumption (homoskedasticity) for the random intercepts. However, in logistic (or probit) models, heteroskedasticity of the random intercepts leads to inconsistent estimates of the regression coefficients (Heagerty and Kurland 2001). In this case, the sandwich estimator is consistent for the sampling variance of an inconsistent point estimator, which may be of limited use (see, for example, Freedman 2006).

10.6.3 Estimation

There are two commands for fitting random-intercept logistic models in Stata, `xtlogit` and `melogit`, and two commands for broader classes of models that include logistic mixed models, `meglm` and `gllamm`. (`melogit` was introduced as `xtmelogit` in Stata 10, and the `xt` prefix was removed in Stata 13.) All of these commands provide marginal ML estimation with model-based standard errors or robust standard errors (sandwich estimator) if the `vce(robust)` option is used (or just `robust` in `gllamm`). Note that the REML (restricted maximum likelihood) estimator, used extensively in volume I, is defined only for *linear* mixed models (although various approximations have been proposed for generalized linear mixed models). Adaptive quadrature is used to approximate the integrals involved in the likelihood (see section 10.10.1 for more information). The commands have essentially the same syntax as their counterparts for linear models discussed in volume I. Specifically, `xtlogit` corresponds to `xtreg`, `melogit` corresponds to `mixed`, and both `meglm` and `gllamm` use essentially the same syntax for linear, logistic, and other types of models.

All of these commands are relatively slow because they use numerical integration, but for random-intercept models, `xtlogit`, `melogit`, and `meglm` tend to be faster than `gllamm`. However, `gllapred`, the postestimation command of `gllamm`, is still the most useful command for predicting random effects and various types of probabilities, as we will see in sections 10.11 and 10.12. Each command uses a default for the number of terms (called "integration points") used to approximate the integral, and there is no guarantee that a sufficient number of terms has been used to achieve reliable estimates. It is therefore the user's responsibility to make sure that the approximation is adequate by increasing the number of integration points until the results stabilize. The more integration points are used, the more accurate the approximation at the cost of increased computation time.

We do not discuss random-coefficient logistic regression in this chapter, but such models can be fit with `melogit` and `gllamm` (but not with `xtlogit`), using essentially the same syntax as for linear random-coefficient models discussed in section 4.5. Random-coefficient logistic regression using `meologit` and `gllamm` is demonstrated in chapter 11 for ordinal responses. Chapter 16 uses `melogit` for a three-level random-coefficient logistic regression model. The probit versions of these models are available in `meprobit`, `meoprobit` and `gllamm` (see sections 11.10 through 11.12 for ordinal probit random-intercept models). Analogously to `xtlogit`, `xtprobit` is a fast command for two-level random-intercept models but cannot be used for random-coefficient or higher-level models.

Using xtlogit

The `xtlogit` command for fitting the random-intercept model is analogous to the `xtreg` command for fitting the corresponding linear model. We first use the `xtset` command to specify the clustering variable. In the `xtlogit` command, we use the `intpoints(30)` option (`intpoints()` stands for "integration points") to ensure accurate estimates (see section 10.10.1):

```
. quietly xtset patient

. xtlogit outcome i.treatment##c.month, intpoints(30)

Random-effects logistic regression                   Number of obs    =  1,908
Group variable: patient                              Number of groups =    294

Random effects u_i ~ Gaussian                        Obs per group:
                                                                  min =      1
                                                                  avg =    6.5
                                                                  max =      7

Integration method: mvaghermite                      Integration pts. =     30

                                                     Wald chi2(3)     = 150.65
Log likelihood = -625.38558                          Prob > chi2      = 0.0000

------------------------------------------------------------------------------
     outcome | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
   treatment |
 Terbinafine |   -.160608   .5796716    -0.28   0.782    -1.296744    .9755275
       month |   -.390956   .0443707    -8.81   0.000    -.4779209   -.3039911
             |
  treatment# |
     c.month |
 Terbinafine |  -.1367758   .0679947    -2.01   0.044     -.270043   -.0035085
             |
       _cons |  -1.618795   .4303891    -3.76   0.000    -2.462342   -.7752477
-------------+----------------------------------------------------------------
    /lnsig2u |   2.775749   .1890237                      2.405269    3.146228
-------------+----------------------------------------------------------------
     sigma_u |   4.006325   .3786451                      3.328876    4.821641
         rho |   .8298976    .026684                      .7710804    .8760322
------------------------------------------------------------------------------
LR test of rho=0: chibar2(01) = 565.24                 Prob >= chibar2 = 0.000
```

The estimated regression coefficients are given in the usual format. The value next to `sigma_u` represents the estimated residual standard deviation $\sqrt{\widehat{\psi}}$ of the random intercept and the value next to `rho` represents the estimated residual intraclass correlation of the latent responses (see section 10.8.1).

We can use the `or` option to obtain exponentiated regression coefficients, which are interpreted as conditional ORs here. Instead of refitting the model, we can simply change the way the results are displayed using the following short `xtlogit` command (known as “replaying the estimation results” in Stata parlance):

```
. xtlogit, or

Random-effects logistic regression                  Number of obs    =  1,908
Group variable: patient                             Number of groups =    294

Random effects u_i ~ Gaussian                       Obs per group:
                                                                 min =      1
                                                                 avg =    6.5
                                                                 max =      7

Integration method: mvaghermite                     Integration pts. =     30

                                                    Wald chi2(3)     = 150.65
Log likelihood = -625.38558                         Prob > chi2      = 0.0000

------------------------------------------------------------------------------
     outcome | Odds ratio   Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
   treatment |
 Terbinafine |   .8516258   .4936633    -0.28   0.782     .2734207    2.652566
       month |   .6764099   .0300128    -8.81   0.000     .6200712    .7378675
             |
  treatment# |
     c.month |
 Terbinafine |   .8721658   .0593027    -2.01   0.044     .7633467    .9964976
             |
       _cons |   .1981373   .0852762    -3.76   0.000     .0852351    .4605897
-------------+----------------------------------------------------------------
    /lnsig2u |   2.775749   .1890237                      2.405269    3.146228
-------------+----------------------------------------------------------------
     sigma_u |   4.006325   .3786451                      3.328876    4.821641
         rho |   .8298976    .026684                      .7710804    .8760322
------------------------------------------------------------------------------
Note: Estimates are transformed only in the first equation to odds ratios.
Note: _cons estimates baseline odds (conditional on zero random effects).
LR test of rho=0: chibar2(01) = 565.24                 Prob >= chibar2 = 0.000
```

Interpretation

The estimated ORs and their 95% confidence intervals are also given in table 10.2. We see that the estimated conditional odds (given ζ_j) for a subject in the itraconazole group are multiplied by 0.68 every month and the conditional odds for a subject in the terbinafine group are multiplied by 0.59 ($= 0.6764 \times 0.8722$) every month. In terms of percentage change in estimated odds, $100\%(\widehat{\mathrm{OR}} - 1)$, the conditional odds decrease 32% $[-32\% = 100\%(0.6764 - 1)]$ per month in the itraconazole group and 41% $[-41\% = 100\%(0.6764 \times 0.8722 - 1)]$ per month in the terbinafine group. (The difference in interpretation of ORs from random-intercept logistic regression and ordinary logistic regression is discussed in section 10.7).

Table 10.2: Estimates for toenail data

	Marginal effects				Conditional effects			
	Ordinary logistic		GEE[†] logistic		Random int. logistic		Conditional logistic	
Parameter	OR	(95% CI)[⋆]	OR	(95% CI)[⋆]	OR	(95% CI)	OR	(95% CI)
Fixed part								
$\exp(\beta_2)$ [`treatment`]	1.00	(0.61, 1.64)	1.01	(0.61, 1.68)	0.85	(0.27, 2.65)		
$\exp(\beta_3)$ [`month`]	0.84	(0.80, 0.89)	0.84	(0.79, 0.89)	0.68	(0.62, 0.74)	0.68	(0.62, 0.75)
$\exp(\beta_4)$ [`treatment#c.month`]	0.93	(0.84, 1.04)	0.93	(0.83, 1.03)	0.87	(0.76, 1.00)	0.91	(0.78, 1.05)
Random part								
ψ					16.08			
ρ					0.83			
Log likelihood	−908.01				−625.39		−188.94[•]	

[†] Using exchangeable working correlation
[⋆] Based on the sandwich estimator for clustered data
[•] Log conditional likelihood

Using melogit

The syntax for `melogit` is analogous to that for `mixed` except that we also specify the number of quadrature points, or integration points, by using the `intpoints()` option:

```
. melogit outcome i.treatment##c.month || patient:, intpoints(30)

Mixed-effects logistic regression               Number of obs     =      1,908
Group variable: patient                         Number of groups  =        294

                                                Obs per group:
                                                              min =          1
                                                              avg =        6.5
                                                              max =          7

Integration method: mvaghermite                 Integration pts.  =         30

                                                Wald chi2(3)      =     150.61
Log likelihood = -625.38557                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
     outcome | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
   treatment |
Terbinafine  |  -.1608934   .5802058    -0.28   0.782    -1.298076    .9762891
       month |  -.3911056   .0443906    -8.81   0.000    -.4781097   -.3041016
             |
  treatment# |
     c.month |
Terbinafine  |  -.1368286   .0680213    -2.01   0.044    -.2701479   -.0035093
             |
       _cons |  -1.620355   .4322382    -3.75   0.000    -2.467526   -.7731834
-------------+----------------------------------------------------------------
patient      |
   var(_cons)|    16.0841   3.062625                      11.07431    23.36021
------------------------------------------------------------------------------
LR test vs. logistic model: chibar2(01) = 565.24      Prob >= chibar2 = 0.0000
```

We store the estimates for later use within the same Stata session:

```
. estimates store melogit
```

(The command `estimates save` can be used to save the estimates in a file for use in a future Stata session.)

The results are almost but not exactly the same as those from `xtlogit` because the commands use slightly different algorithms. Estimated ORs can be obtained using the `or` option, possibly by just replaying the estimation output by using the command `melogit, or`. The estimated standard deviation of the random intercept (instead of the default variance) can be obtained by using the postestimation command `estat sd` (as of Stata 15).

`melogit` can be used with one integration point by specifying the `intpoints(1)` and `intmethod(mcaghermite)` options, which is equivalent to using the Laplace approximation, and can also be requested by specifying the `intmethod(laplace)` option. See section 10.10.2 for the results obtained using this less accurate but faster method for the toenail data.

Using gllamm

We now introduce the community-contributed command for multilevel and latent variable modeling, called gllamm (stands for generalized linear latent and mixed models) by Rabe-Hesketh, Skrondal, and Pickles (2002, 2005). See also http://www.gllamm.org where you can download the gllamm manual, the gllamm companion for this book, and find many other resources.

To check whether gllamm is installed on your computer, use the command

```
. which gllamm
```

If the message

```
command gllamm not found as either built-in or ado-file
```

appears, install gllamm (assuming that you have a net-aware Stata) by using the ssc command:

```
. ssc install gllamm
```

Occasionally, you should update gllamm by using ssc with the replace option:

```
. ssc install gllamm, replace
```

Before fitting the model, we construct a new variable, trt_month, for the interaction of treatment and month:

```
. generate trt_month = treatment*month
```

Using gllamm for the random-intercept logistic regression model requires that we specify a logit link and binomial distribution with the link() and family() options (exactly as for the glm command). We also use the nip() option (for the number of integration points) to request that 30 integration points be used. The cluster identifier is specified in the i() option:

```
. gllamm outcome treatment month trt_month, i(patient) link(logit)
> family(binomial) nip(30) adapt

number of level 1 units = 1908
number of level 2 units = 294

Condition Number = 23.076299

gllamm model

log likelihood = -625.38558

------------------------------------------------------------------------------
     outcome | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
   treatment |  -.1608751   .5802054    -0.28   0.782    -1.298057    .9763065
       month |  -.3911055   .0443906    -8.81   0.000    -.4781096   -.3041015
   trt_month |   -.136829   .0680213    -2.01   0.044    -.2701484   -.0035097
       _cons |  -1.620364   .4322408    -3.75   0.000     -2.46754   -.7731873
------------------------------------------------------------------------------


Variances and covariances of random effects
------------------------------------------------------------------------------


***level 2 (patient)

    var(1): 16.084107 (3.0626223)
------------------------------------------------------------------------------
```

The estimates are almost the same as those from `xtlogit` and `melogit`. The estimated random-intercept variance is given next to `var(1)`. We store the `gllamm` estimates for later use:

```
. estimates store gllamm
```

We can use the `eform` option to obtain estimated ORs, or we can alternatively use the command

```
gllamm, eform
```

to replay the estimation results after having already fit the model. We can also use the `robust` option to obtain robust standard errors based on the sandwich estimator. At the time of writing this book, `gllamm` does not accept factor variables (`i.`, `c.`, and `#`) but does accept `i.` if the `gllamm` command is preceded by the prefix command `xi:`.

10.7 Subject-specific or conditional versus population-averaged or marginal relationships

The estimated regression coefficients for the random-intercept logistic regression model are more extreme (more different from 0) than those for the ordinary logistic regression model (see table 10.2). Correspondingly, the estimated ORs are more extreme (more different from 1) than those for the ordinary logistic regression model. The reason for this discrepancy is that ordinary logistic regression fits overall *population-averaged* or *marginal* probabilities, whereas random-effects logistic regression fits *subject-specific* or *conditional* probabilities for the individual patients.

This important distinction can be seen in the way the two models are written in (10.5) and (10.6). Whereas the former is for the overall or population-averaged probability, conditioning only on covariates, the latter is for the subject-specific probability, conditioning on the covariates *and* the subject-specific random intercept ζ_j. ORs derived from these models can be referred to as population-averaged (although the averaging is applied to the probabilities) or subject-specific ORs, respectively.

For instance, in the random-intercept logistic regression model, we can interpret the estimated subject-specific or conditional OR of 0.68 for `month` (a covariate varying *within* patient) as the OR for each patient in the itraconazole group: the odds for *a given patient* hence decreases by 32% per month. In contrast, the estimated population-averaged OR of 0.84 for `month` means that the odds of having onycholysis *among the patients* in the itraconazole group decreases by 16% per month.

Considering instead the OR for `treatment` (a covariate only varying *between* patients) when `month` equals 1, the estimated subject-specific or conditional OR is estimated as 0.74 (=0.85×0.87) and the odds after one month of treatment are hence 26% lower for terbinafine than for itraconazole for each subject. However, because no patients are given both terbinafine and itraconazole, it might be best to interpret the OR in terms of a comparison between two patients j and j' with the same value of the random intercept $\zeta_j = \zeta_{j'}$, one of whom is given terbinafine and the other itraconazole. The estimated population-averaged or marginal OR of about 0.93 (=1.00×0.93) means that the odds after one month of treatment are 7% lower for the group of patients given terbinafine compared with the group of patients given itraconazole.

When interpreting subject-specific or conditional ORs, keep in mind that these are not purely based on within-subject information and are hence not free from subject-level confounding. In fact, for between-subject covariates like treatment group above, there is no within-subject information in the data. Although the ORs are interpreted as effects keeping the subject-specific random intercepts ζ_j constant, these random intercepts are assumed to be independent of the covariates included in the model and hence do not represent effects of unobserved *confounders*, which are by definition correlated with the covariates. Unlike fixed-effects approaches, we are therefore not controlling for unobserved confounders. Both conditional and marginal effect estimates suffer from omitted-variable bias if subject-level or other confounders are not included in the model. See section 3.7.4 for a discussion of this issue in linear random-intercept models. Section 10.13.1 is on conditional logistic regression, the fixed-effects approach in logistic regression that controls for observed and unobserved subject-level confounders.

The population-averaged probabilities implied by the random-intercept model can be obtained by averaging the subject-specific probabilities over the random-intercept distribution. Because the random intercepts are continuous, this averaging is accomplished by integration:

$$
\begin{aligned}
\Pr(y_{ij} = 1|x_{2j}, x_{3ij})& \\
&= \int \Pr(y_{ij} = 1|x_{2j}, x_{3ij}, \zeta_j)\phi(\zeta_j; 0, \psi)\, d\zeta_j \\
&= \int \frac{\exp(\beta_1 + \beta_2 x_{2j} + \beta_3 x_{3ij} + \beta_4 x_{2j} x_{3ij} + \zeta_j)}{1 + \exp(\beta_1 + \beta_2 x_{2j} + \beta_3 x_{3ij} + \beta_4 x_{2j} x_{3ij} + \zeta_j)}\, \phi(\zeta_j; 0, \psi)\, d\zeta_j \\
&\neq \frac{\exp(\beta_1 + \beta_2 x_{2j} + \beta_3 x_{3ij} + \beta_4 x_{2j} x_{3ij})}{1 + \exp(\beta_1 + \beta_2 x_{2j} + \beta_3 x_{3ij} + \beta_4 x_{2j} x_{3ij})} \qquad (10.8)
\end{aligned}
$$

where $\phi(\zeta_j; 0, \psi)$ is the normal density function with mean 0 and variance ψ.

The difference between population-averaged and subject-specific effects is due to the average of a nonlinear function not being the same as the nonlinear function of the average. In the present context, the average of the inverse logit of the linear predictor, $\beta_1 + \beta_2 x_{2j} + \beta_3 x_{3ij} + \beta_4 x_{2j} x_{3ij} + \zeta_j$, is not the same as the inverse logit of the average of the linear predictor, which is $\beta_1 + \beta_2 x_{2j} + \beta_3 x_{3ij} + \beta_4 x_{2j} x_{3ij}$. We can see this by comparing the simple average of the inverse logits of 1 and 2 with the inverse logit of the average of 1 and 2:

```
. display (invlogit(1) + invlogit(2))/2
.80592783

. display invlogit((1+2)/2)
.81757448
```

We can also see this in figure 10.9. Here the individual, thin, dashed curves represent subject-specific logistic curves for a made-up model, each with a subject-specific (randomly drawn) intercept. These are inverse logit functions of the subject-specific linear predictors (here the linear predictors are simply $\beta_1 + \beta_2 x_{ij} + \zeta_j$). The thick, dashed curve is the inverse logit function of the average of the linear predictor (that is, $\zeta_j = 0$) and this is not the same as the flatter average of the logistic functions shown as a thick, solid curve.

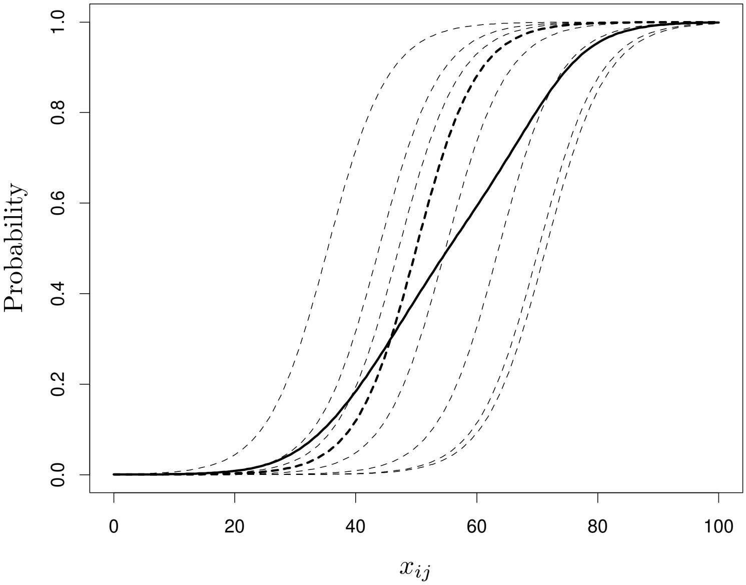


Figure 10.9: Subject-specific probabilities (thin, dashed curves), population-averaged probabilities (thick, solid curve), and population median probabilities (thick, dashed curve) for random-intercept logistic regression

The average curve has a different shape than the subject-specific curves. Specifically, the effect of x_{ij} on the average curve is smaller than the effect of x_{ij} on the subject-specific curves. However, the population median probability is the same as the subject-specific probability evaluated at the median of ζ_j ($\zeta_j = 0$), shown as the thick, dashed curve, because the inverse logit function is a strictly increasing function.

Another way of understanding why the subject-specific effects are more extreme than the population-averaged effects is by writing the random-intercept logistic regression model as a latent-response model:

$$y_{ij}^* = \beta_1 + \beta_2 x_{2j} + \beta_3 x_{3ij} + \beta_4 x_{2j} x_{3ij} + \underbrace{\zeta_j + \epsilon_{ij}}_{\xi_{ij}}$$

The total residual variance is

$$\text{Var}(\xi_{ij}) = \psi + \pi^2/3$$

estimated as $\widehat{\psi} + \pi^2/3 = 16.0841 + 3.2899 = 19.37$, which is much greater than the residual variance of about 3.29 for an ordinary logistic regression model. As we have already seen in figure 10.4 for probit models, the slope in the model for y_i^* has to increase when the residual standard deviation increases to produce an equivalent curve for the marginal probability that the observed response is 1. Therefore, the regression coefficients of the random-intercept model (representing subject-specific effects) must be larger in absolute value than those of the ordinary logistic regression model (representing population-averaged effects) to obtain a good fit of the model-implied marginal probabilities to the corresponding sample proportions (see exercise 10.10). In section 10.12, we will obtain predicted subject-specific and population-averaged probabilities for the toenail data.

In the special case of extremely rare outcomes, subject-specific and population averaged effects become practically identical in random-intercept logistic models (Lin, Psaty, and Kronmal 1998), in contrast to random-intercept probit models.

Having described subject-specific and population-averaged probabilities or expectations of y_{ij}, for given covariate values, we now consider the corresponding variances. The subject-specific or conditional variance is

$$\mathrm{Var}(y_{ij}|\mathbf{x}_{ij},\zeta_j) \;=\; \Pr(y_{ij}=1|\mathbf{x}_{ij},\zeta_j)\{1-\Pr(y_{ij}=1|\mathbf{x}_{ij},\zeta_j)\}$$

and the population-averaged or marginal variance (obtained by integrating over ζ_j) is

$$\mathrm{Var}(y_{ij}|\mathbf{x}_{ij}) \;=\; \Pr(y_{ij}=1|\mathbf{x}_{ij})\{1-\Pr(y_{ij}=1|\mathbf{x}_{ij})\}$$

We see that the random-intercept variance ψ does not affect the relationship between the marginal variance and the marginal mean. This is in contrast to models for counts described in chapter 13, where a random intercept (with $\psi > 0$) produces so-called overdispersion, with a larger marginal variance for a given marginal mean than the model without a random intercept ($\psi = 0$). It is sometimes not recognized that overdispersion is impossible for dichotomous responses (Skrondal and Rabe-Hesketh 2007).

10.8 Measures of dependence and heterogeneity

10.8.1 Conditional or residual intraclass correlation of the latent responses

Returning to the latent-response formulation, the dependence among the dichotomous responses for the same subject (or the between-subject heterogeneity) can be quantified by the *conditional intraclass correlation* or *residual intraclass correlation* ρ of the latent responses y^*_{ij} given the covariates:

$$\rho \equiv \mathrm{Cor}(y^*_{ij},y^*_{i'j}|\mathbf{x}_{ij},\mathbf{x}_{i'j}) \;=\; \mathrm{Cor}(\xi_{ij},\xi_{i'j}) \;=\; \frac{\psi}{\psi+\pi^2/3}$$

Substituting the estimated variance $\widehat{\psi} = 16.08$, we obtain an estimated conditional intraclass correlation of 0.83, which is large even for longitudinal data. The estimated intraclass correlation is also reported next to `rho` by `xtlogit` and can be obtained using `estat icc` after `melogit`:

```
. estimates restore melogit
(results melogit are active now)
. estat icc
Residual intraclass correlation
─────────────────────────────────────────────────────────────────────────────
                        Level │        ICC   Std. err.     [95% conf. interval]
──────────────────────────────┼──────────────────────────────────────────────
                      patient │   .8301913    .0268433      .7709672    .8765531
─────────────────────────────────────────────────────────────────────────────
```

For probit models, the expression for the residual intraclass correlation of the latent responses is as above with $\pi^2/3$ replaced by 1.

10.8.2 Median odds ratio

Larsen et al. (2000) and Larsen and Merlo (2005) suggest a measure of heterogeneity for random-intercept models with normally distributed random intercepts. They consider repeatedly sampling two subjects with the same covariate values and forming the OR comparing the subject who has the larger random intercept with the other subject. For a given pair of subjects j and j', this OR is given by $\exp(|\zeta_j - \zeta_{j'}|)$, and heterogeneity is expressed as the median of these ORs across repeated samples.

The median and other percentiles $a > 1$ can be obtained from the cumulative distribution function

$$\Pr\{\exp(|\zeta_j - \zeta_{j'}|) \leq a\} \;=\; \Pr\left\{\frac{|\zeta_j - \zeta_{j'}|}{\sqrt{2\psi}} \leq \frac{\ln(a)}{\sqrt{2\psi}}\right\} \;=\; 2\,\Phi\left\{\frac{\ln(a)}{\sqrt{2\psi}}\right\} - 1$$

If the cumulative probability is set to 1/2, a is the median OR, $\text{OR}_{\text{median}}$:

$$2\,\Phi\left\{\frac{\ln(\text{OR}_{\text{median}})}{\sqrt{2\psi}}\right\} - 1 \;=\; 1/2$$

Solving this equation gives

$$\text{OR}_{\text{median}} \;=\; \exp\{\sqrt{2\psi}\Phi^{-1}(3/4)\}$$

To be able to plug in the variance estimate, we first check how this is stored by `melogit` by using the `coeflegend` option:

```
. melogit, coeflegend
Mixed-effects logistic regression               Number of obs     =      1,908
Group variable: patient                         Number of groups  =        294

                                                Obs per group:
                                                              min =          1
                                                              avg =        6.5
                                                              max =          7

Integration method: mvaghermite                 Integration pts.  =         30

                                                Wald chi2(3)      =     150.61
Log likelihood = -625.38557                     Prob > chi2       =     0.0000

      outcome   Coefficient  Legend

    treatment
  Terbinafine    -.1608934  _b[1.treatment]
        month    -.3911056  _b[month]

   treatment#
      c.month
  Terbinafine    -.1368286  _b[1.treatment#c.month]

        _cons    -1.620355  _b[_cons]

patient
   var(_cons)      16.0841  _b[/var(_cons[patient])]

LR test vs. logistic model: chibar2(01) = 565.24      Prob >= chibar2 = 0.0000
```

Now we obtain $\widehat{\mathrm{OR}}_{\mathrm{median}}$ as follows:

```
. display exp(sqrt(2*_b[/var(_cons[patient])])*invnormal(3/4))
45.855915
```

When two subjects are chosen at random at a given time point from the same treatment group, the OR comparing the subject who has the larger odds with the subject who has the smaller odds will exceed 45.83 half the time, which is a very large OR. For comparison, the estimated OR comparing two subjects at 20 months who had the same value of the random intercept, but one of whom received itraconazole (`treatment`=0) and the other of whom received terbinafine (`treatment`=1), is 18.13 $\{= 1/\exp(-0.1609 - 20 \times 0.1368)\}$.

10.8.3 ❖ Measures of association for observed responses at median fixed part of the model

The reason why the degree of dependence is often expressed in terms of the residual intraclass correlation for the *latent* responses y_{ij}^* is that the intraclass correlation for the observed responses y_{ij} varies according to the values of the covariates.

One may nevertheless proceed by obtaining measures of association for specific values of the covariates. In particular, Rodríguez and Elo (2003) suggest obtaining the marginal association between the binary observed responses at the sample median value of the estimated fixed part of the model, $\widehat{\beta}_1 + \widehat{\beta}_2 x_{2j} + \widehat{\beta}_3 x_{3ij} + \widehat{\beta}_4 x_{2j} x_{3ij}$. Marginal association here refers to the fact that the associations are based on marginal probabilities (averaged over the random-intercept distribution with the ML estimate $\widehat{\psi}$ plugged in).

Rodríguez and Elo (2003) have written a program called `xtrho` that can be used after `xtlogit`, `xtprobit`, and `xtcloglog` to produce such marginal association measures and their confidence intervals. The program can be downloaded by issuing the command

```
. findit xtrho
```

clicking on `st0031`, and then clicking on `click here to install`. Having downloaded `xtrho`, we run it after refitting the random-intercept logistic model with `xtlogit`:

```
. quietly xtset patient
. quietly xtlogit outcome i.treatment##c.month, re intpoints(30)
. xtrho
Measures of intra-class manifest association in random-effects logit
Evaluated at median linear predictor
Measure              |    Estimate     [95% Conf.Interval]
---------------------+--------------------------------------
                     |
Marginal prob.       |     .250812     .217334     .283389
Joint prob.          |     .178265     .139538     .217568
Odds ratio           |     22.9189     16.2512     32.6823
Pearson´s r          |      .61392     .542645     .675887
Yule´s Q             |     .916384     .884066     .940622
```

We see that for a patient whose fixed part of the linear predictor is equal to the sample median, the marginal probability of having onycholysis (a measure of toenail infection) at an occasion is estimated as 0.25 and the joint probability of having onycholysis at two occasions is estimated as 0.18. From the estimated joint probabilities for the responses 00, 10, 01, and 11 in the 2×2 table for two occasions (with linear predictor equal to the sample median), `xtrho` estimates various measures of association for onycholysis for two occasions, given that the fixed part of the linear predictor equals the sample median.

The estimated OR of 22.92 means that the odds of onycholysis at one of the two occasions is almost 23 times as high for a patient who had onycholysis at the other occasion as for a patient with the same covariates (`treatment` and `month`) who did not have onycholysis at the other occasion. The estimated Pearson correlation of 0.61 for the observed responses is lower than the estimated residual correlation for the latent responses of 0.83, as would be expected from statistical theory. Squaring the Pearson correlation, we see that onycholysis at one occasion explains about 36% of the variation in onycholysis at the other occasion for fixed covariate values.

We can use the `detail` option to obtain the above measures of associations evaluated at sample percentiles other than the median. We can also use Rodríguez and Elo's (2003) `xtrhoi` command to obtain measures of associations for other values of the fixed part of the linear predictor and other values of the variance of the random-intercept distribution.

xtrho and xtrhoi assume that the fixed part of the linear predictor is the same across occasions. However, in the toenail example, month must change between any two occasions within a patient, and the linear predictor is a function of month. Considering two occasions with month equal to 3 and 6, the OR is estimated as 25.6 for patients in the control group and 29.4 for patients in the treatment group. Marginal 2×2 tables, taking into account that month changes, can be obtained using gllamm and gllapred with the ll option as demonstrated in a do-file that can copied into the working directory as follows:

```
copy https://www.stata-press.com/data/mlmus4/ch10table.do ch10table.do
```

10.9 Inference for random-intercept logistic models

10.9.1 Tests and confidence intervals for odds ratios

As discussed earlier, we can interpret the regression coefficient β as the difference in log odds associated with a unit change in the corresponding covariate, and we can interpret the exponentiated regression coefficient as an OR, OR $= \exp(\beta)$. The relevant null hypothesis for ORs usually is H_0: OR $= 1$, and this corresponds directly to the null hypothesis that the corresponding regression coefficient is 0, H_0: $\beta = 0$.

Wald tests can be used for regression coefficients just as described in section 3.6.1 for linear models. Ninety-five percent Wald confidence intervals for individual regression coefficients are obtained using

$$\widehat{\beta} \pm z_{0.975}\,\widehat{\mathrm{SE}}(\widehat{\beta})$$

where $z_{0.975} = 1.96$ is the 97.5th percentile of the standard normal distribution. The corresponding confidence interval for the OR is obtained by exponentiating both limits of the confidence interval:

$$\exp\{\widehat{\beta} - z_{0.975}\,\widehat{\mathrm{SE}}(\widehat{\beta})\} \quad \text{to} \quad \exp\{\widehat{\beta} + z_{0.975}\,\widehat{\mathrm{SE}}(\widehat{\beta})\}$$

Wald tests for linear combinations of regression coefficients can be used to test the corresponding multiplicative relationships among odds for different covariate values. For instance, for the toenail data, we may want to obtain the OR comparing the treatment groups after 20 months. The corresponding difference in log odds after 20 months is a linear combination of regression coefficients, namely, $\beta_2 + \beta_4 \times 20$ (see section 1.8 if this is not clear). We can test the null hypothesis that this difference in log odds is 0 and hence that the OR is 1 by using the lincom command:

```
. lincom 1.treatment + 1.treatment#c.month*20
 ( 1)  [outcome]1.treatment + 20*[outcome]1.treatment#c.month = 0

------------------------------------------------------------------------------
     outcome | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
         (1) |  -2.896123   1.309682    -2.21   0.027    -5.463053   -.3291935
------------------------------------------------------------------------------
```

If we require the corresponding OR with a 95% confidence interval, we can use the `lincom` command with the `or` option:

```
. lincom 1.treatment + 1.treatment#c.month*20, or
 ( 1)  [outcome]1.treatment + 20*[outcome]1.treatment#c.month = 0

------------------------------------------------------------------------------
     outcome | Odds ratio   Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
         (1) |   .0552369   .0723428    -2.21   0.027     .0042406    .7195038
------------------------------------------------------------------------------
```

After 20 months of treatment, the OR comparing terbinafine (`treatment`=1) with itraconazole is estimated as 0.055. Such small numbers are difficult to interpret, so we can switch the groups around by taking the reciprocal of the OR, 18 (= 1/0.055), which represents the OR comparing itraconazole with terbinafine. Alternatively, we can always switch the comparison around by simply changing the sign of the corresponding difference in log odds in the `lincom` command:

```
lincom -(1.treatment + 1.treatment#c.month*20), or
```

Multivariate Wald tests can be performed using `testparm`. Wald tests and confidence intervals can be based on robust standard errors from the sandwich estimator, obtained using the `vce(robust)` option in estimation commands (if the number of clusters J minus the number of cluster-level covariates q is at least 42).

Null hypotheses about individual regression coefficients or several regression coefficients can also be tested using likelihood-ratio tests. Although likelihood-ratio and Wald tests are asymptotically equivalent, the test statistics are not identical in finite samples. (See display 2.2 for the relationships between likelihood-ratio, Wald, and score tests.) If the statistics are very different, there may be a sparseness problem, for instance with mostly "1" responses or mostly "0" responses in one of the groups.

10.9.2 Tests of variance components

The last line of the output of both `xtlogit` and `melogit` provides a likelihood-ratio test for the null hypothesis that the residual between-cluster variance ψ is 0. The p-value is based on the correct asymptotic sampling distribution $[0.5\chi^2(0) + 0.5\chi^2(0)$, as described for linear models in section 2.6.2, not the naïve $\chi^2(1)]$. For the toenail data, the likelihood-ratio statistic is 565.2, giving $p < 0.001$, which suggests that a multilevel model is required.

10.10 Maximum likelihood estimation

10.10.1 ❖ Adaptive quadrature

The marginal likelihood is the joint probability of all observed responses given the observed covariates. For linear mixed models, this marginal likelihood can be evaluated and maximized relatively easily (see section 2.10). However, in generalized linear mixed models, the marginal likelihood does not have a closed form and must be evaluated by approximate methods.

To see this, we will now construct this marginal likelihood step by step for a random-intercept logistic regression model with one covariate x_j. The responses are conditionally independent given the random intercept ζ_j and the covariate x_j. Therefore, the joint probability of all the responses y_{ij} $(i=1,\ldots,n_j)$ for cluster j given the random intercept and covariate is simply the product of the conditional probabilities of the individual responses:

$$\Pr(y_{1j},\ldots,y_{n_jj}|x_j,\zeta_j) \;=\; \prod_{i=1}^{n_j}\Pr(y_{ij}|x_j,\zeta_j) \;=\; \prod_{i=1}^{n_j}\frac{\exp(\beta_1+\beta_2x_j+\zeta_j)^{y_{ij}}}{1+\exp(\beta_1+\beta_2x_j+\zeta_j)}$$

In the last term,

$$\frac{\exp(\beta_1+\beta_2x_j+\zeta_j)^{y_{ij}}}{1+\exp(\beta_1+\beta_2x_j+\zeta_j)} \;=\; \begin{cases} \frac{\exp(\beta_1+\beta_2x_j+\zeta_j)}{1+\exp(\beta_1+\beta_2x_j+\zeta_j)} & \text{if } y_{ij}=1 \\ \frac{1}{1+\exp(\beta_1+\beta_2x_j+\zeta_j)} & \text{if } y_{ij}=0 \end{cases}$$

as specified by the logistic regression model.

To obtain the marginal joint probability of the responses, not conditioning on the random intercept ζ_j (but still on the covariate x_j), we integrate out the random intercept:

$$\Pr(y_{1j},\ldots,y_{n_jj}|x_j) \;=\; \int \Pr(y_{1j},\ldots,y_{n_jj}|x_j,\zeta_j)\,\phi(\zeta_j;0,\psi)\,d\zeta_j \tag{10.9}$$

where $\phi(\zeta_j,0,\psi)$ is the normal density of ζ_j with mean 0 and variance ψ. Unfortunately, this integral does not have a closed-form expression.

The marginal likelihood is just the joint probability of all responses for all clusters. Because the clusters are mutually independent, this is given by the product of the marginal joint probabilities of the responses for the individual clusters:

$$L(\beta_1,\beta_2,\psi) \;=\; \prod_{j=1}^{N}\Pr(y_{1j},\ldots,y_{n_jj}|x_j)$$

This marginal likelihood is viewed as a function of the parameters β_1, β_2, and ψ (with the observed responses treated as given). The parameters are estimated by finding the values of β_1, β_2, and ψ that yield the largest likelihood. The search for the maximum is iterative, beginning with some initial guesses or starting values for the parameters and updating these step by step until the maximum is reached, typically by using a Newton–Raphson or expectation-maximization (EM) algorithm.

The integral over ζ_j in (10.9) can be approximated by a sum of R terms with e_r substituted for ζ_j and the normal density replaced by a weight w_r for the rth term, $r = 1, \ldots, R$,

$$\Pr(y_{1j}, \ldots, y_{n_j j}|x_j) \approx \sum_{r=1}^{R} \Pr(y_{1j}, \ldots, y_{n_j j}|x_j, \zeta_j = e_r)\, w_r$$

where e_r and w_r are called Gauss–Hermite quadrature locations and weights, respectively. This approximation can be viewed as replacing the continuous density of ζ_j with a discrete distribution with R possible values of ζ_j having probabilities $w_r = \Pr(\zeta_j = e_r)$. The Gauss–Hermite approximation is illustrated for $R = 5$ in figure 10.10. Obviously, the approximation improves when the number of points R increases.

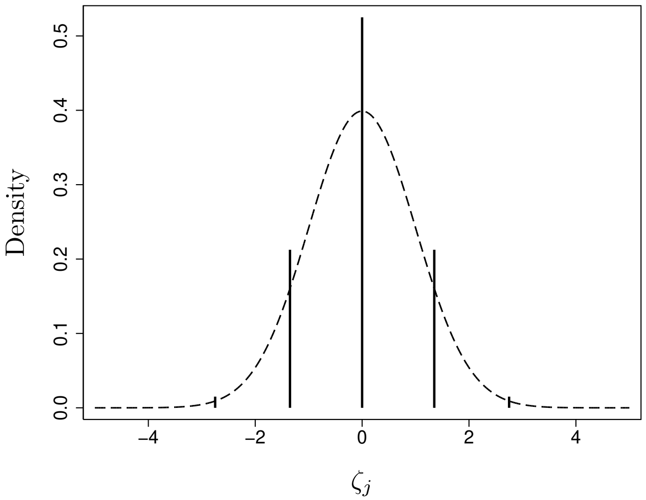


Figure 10.10: Gauss–Hermite quadrature: Approximating continuous density (dashed curve) by discrete distribution (bars)

The ordinary quadrature approximation described above can perform poorly if the function being integrated, called the *integrand*, has a sharp peak, as discussed in Rabe-Hesketh, Skrondal, and Pickles (2002, 2005). Sharp peaks can occur when the clusters are very large so that many functions (the individual response probabilities as functions of ζ_j) are multiplied to yield $\Pr(y_{1j}, \ldots, y_{n_j j}|x_j, \zeta_j)$. Similarly, if the responses are counts or continuous responses, even a few terms can result in a highly peaked function. Another potential problem is a high intraclass correlation. Here the functions being multiplied coincide with each other more closely because of the greater similarity of responses within clusters, yielding a sharper peak. In fact, the toenail data we have been analyzing, which has an estimated conditional intraclass correlation for the latent responses of 0.83, poses real problems for estimation using ordinary quadrature, as pointed out by Lesaffre and Spiessens (2001).

The top panel in figure 10.11 shows the same five-point quadrature approximation and density of ζ_j as in figure 10.10. The solid curve is proportional to the integrand for a hypothetical cluster. Here the quadrature approximation works poorly because the peak of the integrand falls between adjacent quadrature points.

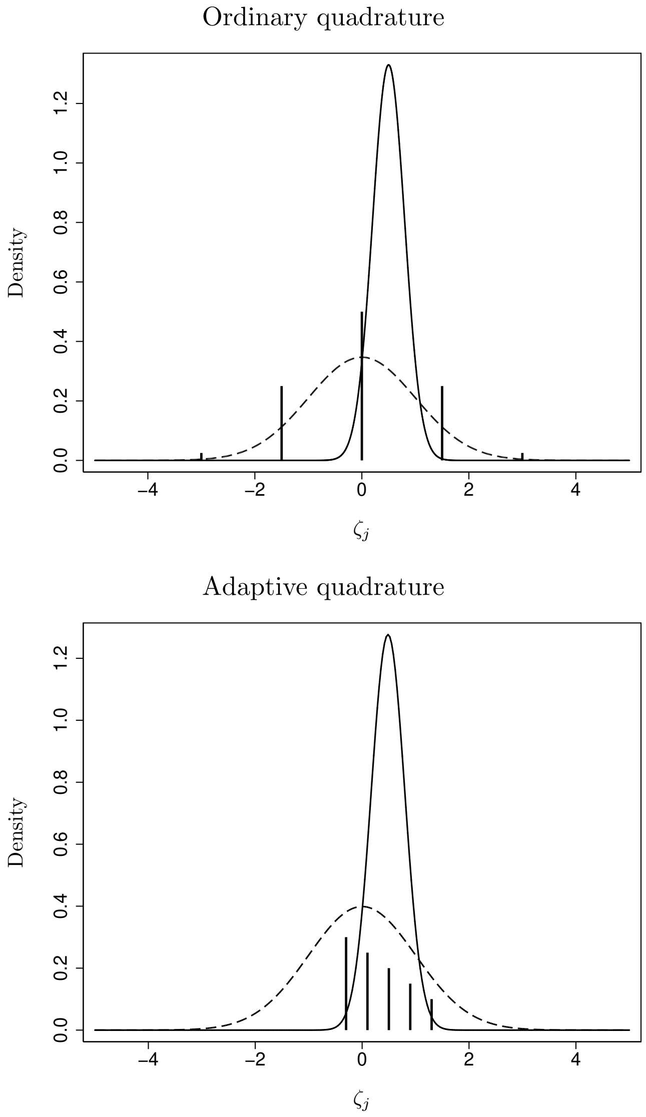


Figure 10.11: Density of ζ_j (dashed curve), normalized integrand (solid curve), and quadrature weights (bars) for ordinary quadrature and adaptive quadrature (*source:* Rabe-Hesketh, Skrondal, and Pickles 2002)

The bottom panel of figure 10.11 shows an improved approximation, known as *adaptive quadrature* with rescaled and translated locations,

$$e_{rj} = a_j + b_j e_r \tag{10.10}$$

that are concentrated under the peak of the integrand, where a_j and b_j are cluster-specific constants. This transformation of the locations is accompanied by a transformation of the weights w_r that also depends on a_j and b_j. The method is called *adaptive* because the quadrature locations and weights are adapted to the data for the individual clusters.

To maximize the likelihood, we start with a set of initial or starting values of the parameters and then keep updating the parameters until the likelihood is maximized. The quantities a_j and b_j needed to evaluate the likelihood are functions of the parameters (as well as the data) and must therefore be updated or "readapted" when the parameters are updated.

There are two different implementations of adaptive quadrature in Stata that differ in the values used for a_j and b_j in (10.10). The method called `mvaghermite` for "mean–variance adaptive Gauss–Hermite quadrature" uses the posterior mean of ζ_j for a_j and the posterior standard deviation for b_j. This is the default in `melogit`, `xtlogit`, and `gllamm`. Obtaining the posterior mean and standard deviation requires numerical integration (which requires values of a_j and b_j, making the process iterative), so the `mvaghermite` version of adaptive quadrature sometimes does not work when there are too few quadrature points (for example, fewer than five). Details of the algorithm are given in Rabe-Hesketh, Skrondal, and Pickles (2002, 2005) and Skrondal and Rabe-Hesketh (2004).

The method called `mcaghermite` for "mode-curvature adaptive quadrature Gauss–Hermite quadrature" uses the posterior mode of ζ_j for a_j and for b_j it uses the standard deviation of the normal density whose logarithm has the same curvature (second derivative) as the log posterior of ζ_j at the mode. An advantage of this approach is that it does not rely on numerical integration and can therefore be implemented even with one quadrature point. With one quadrature point, this version of adaptive quadrature becomes a Laplace approximation. The `laplace` method is the default in `melogit` for models with crossed random effects (see chapter 16) and can be requested for other models by using the `intmethod(laplace)` option.

10.10.2 Some speed and accuracy considerations

Integration methods and number of quadrature points

As discussed in section 10.10.1, the likelihood involves integrals that are evaluated by numerical integration. The marginal likelihood itself, as well as the marginal ML estimates, are therefore only approximate. The accuracy increases as the number of quadrature points increases, at the cost of increased computation time. We can assess whether the approximation is adequate in a given situation by repeating the analysis with a larger number of quadrature points. If we get essentially the same result, the lower number of quadrature points is likely to be adequate. Such checking should always be done before estimates are taken at face value. See section 16.3.4 for an example in `gllamm`. For a given number of quadrature points, adaptive quadrature is more accurate than ordinary quadrature. Stata's commands therefore use adaptive quadrature by default, and we recommend using the `adapt` option in `gllamm`.

Because of numerical integration, estimation can be slow, especially if there are many random effects. The time it takes to fit a model is approximately proportional to the product of the number of quadrature points for all random effects. For example, if there are two random effects at level 2 (a random intercept and slope) and eight quadrature points are used for each random effect, the time will be approximately proportional to 64. Therefore, using four quadrature points for each random effect will take only about one-fourth (16/64) as long as using eight. The time is also approximately proportional to the number of observations. `melogit` and the other "`me`" commands use analytical differentiation as of Stata 15, making them much faster than before when they used numerical differentiation. `gllamm` still uses numerical differentiation and its computation time is therefore approximately quadratic in the number of parameters.

For large problems, it may be advisable to estimate how long estimation will take before starting work on a project. In this case, we recommend fitting a similar model with fewer random effects, fewer integration points, fewer observations, or some combination of those, and using the above approximate proportionality factors to estimate the time that will be required for the larger problem.

For random-intercept models `melogit` and `xtlogit` are relatively fast because they uses analytical derivatives. For random-coefficient models or higher-level models introduced in chapter 16, the quickest way of obtaining results is with `melogit` by specifying `intmethod(laplace)` or `intmethod(mcaghermite)` together with `intpoints(1)`. Although this Laplace approximation sometimes works well, it can produce severely biased estimates, especially if the clusters are small and the (true) random-intercept variance is large, as for the toenail data. For these data, we obtain the following:

```
. melogit outcome i.treatment##c.month || patient:, intmethod(laplace)

Mixed-effects logistic regression               Number of obs     =      1,908
Group variable: patient                         Number of groups  =        294

                                                Obs per group:
                                                              min =          1
                                                              avg =        6.5
                                                              max =          7

Integration method: laplace

                                                Wald chi2(3)      =     131.96
Log likelihood = -627.80894                     Prob > chi2       =     0.0000

------------------------------------------------------------------------------
     outcome | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
   treatment |
Terbinafine  |  -.3070105   .6899463    -0.44   0.656     -1.65928    1.045259
       month |  -.4000896   .0470581    -8.50   0.000    -.4923217   -.3078574
             |
   treatment#|
     c.month |
Terbinafine  |  -.1372588    .069586    -1.97   0.049    -.2736448   -.0008728
             |
       _cons |  -2.523249   .7882044    -3.20   0.001    -4.068101   -.9783969
-------------+----------------------------------------------------------------
patient      |
   var(_cons)|   20.89221   6.580464                      11.26885    38.73371
------------------------------------------------------------------------------
LR test vs. logistic model: chibar2(01) = 560.40      Prob >= chibar2 = 0.0000
```

We see that the estimated intercept and coefficient of `treatment` are very different from the estimates in section 10.6.3 using adaptive quadrature with 30 quadrature points. As mentioned in the previous section, the integration method `mvaghermite` typically requires at least five quadrature points, but the method `mcaghermite` used above (with one integration point which makes this Laplace) could also be used with just two or three integration points to check the accuracy of the Laplace approximation while keeping estimation relatively fast (see section 16.8).

Starting values

To speed up estimation in `melogit` or `gllamm`, use good starting values if they are available. For instance, when increasing the number of quadrature points or adding or dropping covariates, use the previous estimates as starting values. Another possibility, when estimation is slow because the dataset is large, is fitting the model on a subset of the data to obtain starting values for the full dataset. Starting values can be specified using the `from()` option. This option should be combined with the `skip` option if the new model contains fewer parameters than supplied. You can also use the `copy` option if your parameters are supplied in the correct order yet are not necessarily labeled correctly. Use of these options is demonstrated in sections 16.3 and 16.8. The `startvalues()` option in `melogit` can be used to specify how starting values should be computed.

Using melogit and gllamm for collapsible data

For some datasets and models, you can represent the data by using fewer rows than there are observations, thus speeding up estimation. For example, if the response is dichotomous and we are using one dichotomous covariate in a two-level dataset, then we can use one row of data for each combination of covariate and response (00, 01, 10, 11) for each cluster, leading to at most four rows per cluster. We can then specify a variable containing level-1 frequency weights equal to the number of observations, or level-1 units, in each cluster having each combination of the covariate and response values. Level-2 weights can be used if several clusters have the same response and covariate pattern across units. For example, if clusters are of size $n_j = 5$ and the response variable is binary, then there are only 32 different response patterns; if there is one cluster-level binary covariate, the dataset can be reduced to 64 cluster types, which can be treated as "clusters" in the collapsed data, giving a total of 320 ($= 64 \times 5$) units. Level-2 weights then specify how many original clusters each of the 64 "clusters" in the collapsed dataset represents. For collapsed data with frequency weights, `melogit` can be used with the `fweight()` and `[fweight=]` options for weights at the highest and lowest level, respectively. Weights at all levels can be specified in `gllamm` by using the `weight()` option. See exercise 10.7 for an example with level-1 weights, and see exercises 10.3 and 2.3 for examples with level-2 weights. In exercise 16.11, collapsing the data reduces computation time by about 99%.

Spherical quadrature in gllamm

For models involving several random effects at the same level, such as two-level random-coefficient models with a random intercept and slope, the multivariate integral can be evaluated more efficiently using *spherical quadrature* instead of the default Cartesian-product quadrature. For the random intercept and slope example, Cartesian-product quadrature consists of evaluating the function being integrated on the rectangular grid of quadrature points consisting of all combinations of $\zeta_{1j} = e_1, \ldots, e_R$ and $\zeta_{2j} = e_1, \ldots, e_R$, giving R^2 terms. In contrast, spherical quadrature consists of evaluating ζ_{1j} and ζ_{2j} at values falling on concentric circles (spheres in three dimensions). The important point is that the same accuracy can now be achieved with fewer than R^2 points. For example, when $R = 8$, Cartesian-product quadrature requires 64 evaluations, while spherical quadrature requires only 44 evaluations, taking nearly 30% less time to achieve the same accuracy. Here accuracy is expressed in terms of the degree of the approximation given by $d = 2R-1$ for Cartesian-product quadrature. For example, $R = 8$ gives $d = 15$. To use spherical quadrature, specify the `ip(m)` option in `gllamm` and give the degree d of the approximation by using the `nip(#)` option. Unfortunately, spherical integration is available only for certain combinations of numbers of dimensions (or numbers of random effects) and degrees of accuracy, d: For two dimensions, d can be 5, 7, 9, 11, or 15, and for more than two dimensions, d can be 5 or 7. See Rabe-Hesketh, Skrondal, and Pickles (2005) for more information.

10.11 Assigning values to random effects

Having estimated the model parameters (the β's and ψ), we may want to assign values to the random intercepts ζ_j for individual clusters j. The ζ_j are not model parameters, but as for linear models, we can treat the estimated parameters as known and then either estimate or predict ζ_j.

Such predictions are useful for making inferences for the clusters in the data, important examples being assessment of institutional performance (see section 4.8.5) or of abilities in item response theory (see exercise 10.4). The estimated or predicted values of ζ_j should generally not be used for model diagnostics in random-intercept logistic regression because their distribution if the model is true is not known. In general, the values should also not be used to obtain cluster-specific predicted probabilities (see section 10.12.2).

10.11.1 Maximum "likelihood" estimation

As discussed for linear models in section 2.11.1, we can estimate the intercepts ζ_j by treating them as the only unknown parameters, after estimates have been plugged in for the model parameters:

$$\text{logit}\{\widehat{\Pr}(y_{ij} = 1|\mathbf{x}_{ij}, \zeta_j)\} \quad = \quad \underbrace{\text{offset}_{ij}}_{\widehat{\beta}_1 + \widehat{\beta}_2 x_{2ij} + \cdots} \quad + \quad \zeta_j$$

This is a logistic regression model for cluster j with offset (a term with regression coefficient set to 1) given by the estimated fixed part of the linear predictor and with a cluster-specific intercept ζ_j.

We then maximize the corresponding likelihood for cluster j,

$$\text{Likelihood}(y_{1j}, y_{2j}, \ldots, y_{n_j j}|\mathbf{X}_j, \zeta_j)$$

with respect to ζ_j, where $\mathbf{X}_j$ is a matrix containing all covariates for cluster j. As explained in section 2.11.1, we put "likelihood" in quotes in the section heading because it differs from the marginal likelihood that is used to estimate the model parameters.

Maximization can be accomplished by fitting logistic regression models to the individual clusters. First, obtain the offset from the melogit estimates:

```
. estimates restore melogit
(results melogit are active now)
. predict offset, xb
```

Then use the `statsby` command to fit individual logistic regression models for each patient, specifying an offset:

```
. statsby mlest=_b[_cons], by(patient) saving(ml, replace): logit outcome,
> offset(offset)
(running logit on estimation sample)

      Command: logit outcome, offset(offset)
        mlest: _b[_cons]
           By: patient
(file ml.dta not found)

Statsby groups
────┼─── 1 ───┼─── 2 ───┼─── 3 ───┼─── 4 ───┼─── 5
......xx.......xx..xxx...x.x...xxxxx.xx...xxx.xxxx    50
xx.xxxxxxxxx.xxxx..xxxxxxxxx.x..xx..x.xxx.xxx.x...   100
xx.xxxxxxxxxxx.xxx.x.x...x.xx.xxxxx.xx....xxx.x.xx   150
.x..x.xxxx..xxxxx.xx..xxxx..xxx.x.xxxxx.x.x.xxx...   200
.xxxxx.xx.xx..x.xxx...xx.x..xxxxx.x..x.x..x..xxxxx   250
x.xx.x..xxxxxx..x..x..xxx.x..xxxxxxxx.x.x...
```

Here we have saved the estimates under the variable name `mlest` in a file called `ml.dta` in the working directory. The `x`'s in the output indicate that the `logit` command did not converge for many clusters. For these clusters, the variable `mlest` is missing. This happens for clusters where all responses are 0 or all responses are 1 because the maximum "likelihood" estimate then is $-\infty$ and $+\infty$, respectively.

We now merge the estimates with the data for later use:

```
. sort patient

. merge m:1 patient using ml
    Result                           Number of obs
    ─────────────────────────────────────────────
    Not matched                                  0
    Matched                                  1,908  (_merge==3)
    ─────────────────────────────────────────────

. drop _merge
```

10.11.2 Empirical Bayes prediction

The ideas behind empirical Bayes prediction discussed in section 2.11.2 for linear variance-components models also apply to other generalized linear mixed models. Instead of basing inference completely on the "likelihood" of the responses for a cluster given the random intercept, we combine this information with the prior of the random intercept, which is just the estimated density of the random intercept (a normal density with mean 0 and estimated variance $\widehat{\psi}$), to obtain the posterior distribution:

$$\text{Posterior}(\zeta_j|y_{1j},\ldots,y_{n_jj},\mathbf{X}_j) \;\propto\; \text{Prior}(\zeta_j)\times\text{Likelihood}(y_{1j},\ldots,y_{n_jj}|\mathbf{X}_j,\zeta_j)$$

The product on the right is proportional to, but not equal to, the posterior distribution. Obtaining the posterior distribution requires dividing this product by a normalizing constant that can only be obtained by numerical integration. Note that the model parameters are treated as known, and estimates are plugged into the expression for the posterior, giving what is sometimes called an estimated posterior distribution.

The estimated posterior distribution is no longer normal as for linear models, and hence its mode does not equal its mean. There are therefore two different types of predictions we could consider: the mean of the posterior and its mode. The first is undoubtedly the most common and is referred to as empirical Bayes prediction [sometimes called expected a posterior (EAP) prediction], whereas the second is referred to as empirical Bayes modal prediction [sometimes called modal a posterior (MAP) prediction].

The empirical Bayes prediction of the random intercept for a cluster j is the mean of the estimated posterior distribution and can be obtained as

$$\widetilde{\zeta}_j \;=\; \int \zeta_j \, \text{Posterior}(\zeta_j | y_{1j}, \ldots, y_{n_j j}, \mathbf{X}_j) \, d\zeta_j$$

by using numerical integration. This is <u>not</u> a best linear unbiased prediction (BLUP) as in linear models.

We can obtain empirical Bayes predictions by using the postestimation command `predict` for `melogit` with the `reffects` and `ebmeans` (for empirical Bayes) options:

```
. estimates restore melogit
(results melogit are active now)
. predict eb, reffects ebmeans reses(semean)
(calculating posterior means of random effects)
(using 30 quadrature points)
```

The variable `eb` contains the empirical Bayes predictions. In the next section, we will produce a graph of these predictions, together with maximum “likelihood” estimates and empirical Bayes modal predictions.

The posterior standard deviations produced by the `reses()` option of `predict` above and placed in the variable `semean` represent the conditional standard deviations of the prediction errors, given the observed responses and treating the parameter estimates as known. The square of `semean` is also the conditional mean squared error of the prediction, conditional on the observed responses. As in section 2.11.3, we refer to this standard error as the *comparative standard error* because it can be used to make inferences regarding the random effects of individual clusters and to compare clusters.

We mentioned in section 2.11.3 that, for linear models, the posterior variance is the same as the unconditional mean squared error of prediction (MSEP) or diagnostic standard error. However, this is not true for generalized linear mixed models not having an identity link, such as the random-intercept logistic model discussed here.

There is also no longer an easy way to obtain the MSEP or diagnostic standard error, but an approximate version can be obtained as the square root of the difference between the random-intercept variance estimate and the posterior variance estimate, here $\sqrt{\widehat{\psi} - \texttt{semean}^2}$. (See Skrondal and Rabe-Hesketh [2004, 231–232] or Skrondal and Rabe-Hesketh [2009] for details). This approximation is used by `gllamm` with the ustd option to obtain empirical Bayes predictions divided by their approximate diagnostic standard errors.

10.11.3 Empirical Bayes modal prediction

Instead of basing prediction of random effects on the mean of the posterior distribution, we can use the mode. Such empirical Bayes modal predictions are easy to obtain using the predict command with the reffects and ebmodes (for empirical Bayes modal) options:

```
. predict ebmodal, reffects ebmodes reses(semode)
(calculating posterior modes of random effects)
```

To see how the various methods compare, we now produce a graph of the empirical Bayes modal predictions (circles) and nonmissing maximum “likelihood” estimates (triangles) versus the empirical Bayes predictions, connecting empirical Bayes modal predictions and maximum “likelihood” estimates with vertical lines.

```
. twoway (rspike mlest ebmodal eb if visit==1)
> (scatter mlest  eb if visit==1, msize(small) msym(th) mcol(black))
> (scatter ebmodal eb if visit==1, msize(small) msym(oh) mcol(black))
> (function y=x, range(eb) lpatt(solid)),
> xtitle(Empirical Bayes prediction)
> legend(order(2 "Maximum likelihood" 3 "Empirical Bayes modal"))
```

The graph is given in figure 10.12.

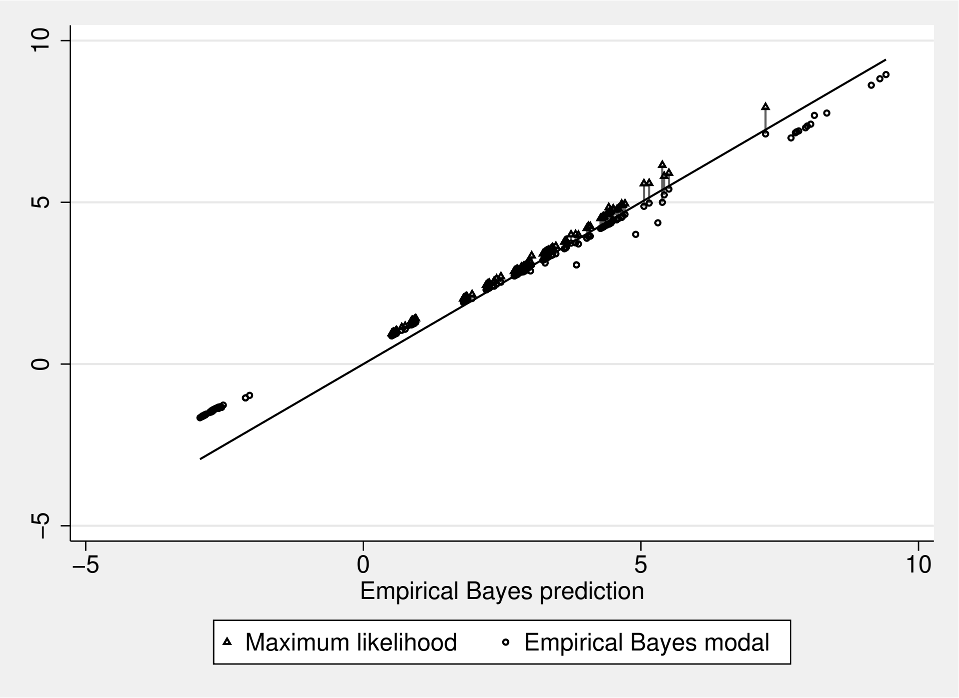


Figure 10.12: Empirical Bayes modal predictions (circles) and nonmissing maximum "likelihood" estimates (triangles) versus empirical Bayes predictions

We see that the maximum "likelihood" estimates are missing when the empirical Bayes predictions are extreme (where the responses are all 0 or all 1) and that the empirical Bayes modal predictions tend to be quite close to the empirical Bayes predictions (close to the line).

When using the posterior modes as a predictions of the random intercepts instead of the posterior means, it makes sense to base standard errors on the curvature of the log posterior distribution at the mode (just as is done in the `mcaghermite` method of adaptive quadrature). The `reses()` option of `predict` combined with the `ebmodes` option therefore produces standard errors that are standard deviations of normal densities that approximate the posterior at the mode. These approximate standard errors have been placed in the variable `semode`. Below we list the predictions and standard errors produced by the `ebmeans` option (`eb` and `semean`) with those produced by the `ebmodes` option (`ebmodal` and `semode`), together with the number of 0 responses, `num0`, and the number of 1 responses, `num1`, for the first 16 patients:

```
. egen num0 = total(outcome==0), by(patient)
. egen num1 = total(outcome==1), by(patient)
. list patient num0 num1 eb ebmodal semean semode if visit==1&patient<=12, noobs
```

```
  ┌──────────────────────────────────────────────────────────────────────────┐
  │ patient   num0   num1          eb     ebmodal     semean     semode │
  ├──────────────────────────────────────────────────────────────────────────┤
  │       1      4      3    3.742003    3.738238   1.053459   1.025657 │
  │       2      4      2    1.834451    1.935999   1.019206   .9423842 │
  │       3      6      1    .5889862    .9492028    1.30982    1.13152 │
  │       4      6      1    .6017116    .9566681   1.314893   1.136407 │
  │       6      4      3    3.283585    3.255311    1.01189   .9710647 │
  ├──────────────────────────────────────────────────────────────────────────┤
  │       7      4      3    3.403232    3.369022   1.030795   .9956948 │
  │       9      7      0   -2.680705   -1.399398   2.707369   2.610388 │
  │      10      7      0   -2.888323   -1.604582   2.645096   2.505447 │
  │      11      3      4    4.464952    4.363659   1.088513    1.07268 │
  │      12      4      3    2.727964     2.73049   .9417346    .899028 │
  └──────────────────────────────────────────────────────────────────────────┘
```

We see that the predictions and standard errors agree reasonably well for some patients with several 1s and 0s, but there are large discrepancies when the responses are all 0 (patients 9 and 10), which also leads to large standard errors. Such large discrepancies suggest that the (empirical) posteriors are asymmetric. Although the posterior mean is generally preferred by Bayesians because it minimizes the posterior mean squared error loss, the mean becomes a less useful summary of the posterior when the posterior is very asymmetric.

10.12 Different kinds of predicted probabilities

10.12.1 Predicted population-averaged or marginal probabilities

Population-averaged or marginal probabilities $\overline{\pi}(\mathbf{x}_{ij})$ can be predicted for random-intercept logistic regression models by evaluating the integral in (10.8) numerically for the estimated parameters and values of covariates in the data, that is, evaluating

$$\overline{\pi}(\mathbf{x}_{ij}) \equiv \int \widehat{\Pr}(y_{ij}=1|x_{2j},x_{3ij},\zeta_j)\phi(\zeta_j;0,\widehat{\psi})\,d\zeta_j$$

To obtain these predicted marginal probabilities after estimation using `melogit`, use the `predict` command with the options `pr` or `mu` (for the probability or mean response) and `marginal` (for integrating over the random-intercept distribution):

```
. predict margprob, pr marginal
(using 30 quadrature points)
```

(After estimation using `gllamm`, the command `predict margprob, mu marginal` gives identical results.)

We now compare predictions of population-averaged or marginal probabilities from the ordinary logit model (previously obtained under the variable name `prob`) and the random-intercept logit model, giving figure 10.13.

```
. twoway (line prob month, sort) (line margprob month, sort lpatt(dash) ),
>  by(treatment) legend(order(1 "Ordinary logit" 2 "Random intercept logit"))
>  xtitle(Time in months) ytitle(Fitted marginal probabilities of onycholysis)
```

The predictions are nearly identical. This is not surprising because estimators of the marginal relationships based on generalized linear mixed models are consistent for the true marginal relationships even if the random-intercept distribution is misspecified (Heagerty and Kurland 2001), implying that both single-level and random-intercept logistic models are consistent. (However, unlike probit models, logit models have subtly different functional forms for the marginal relationship depending on the estimated random-intercept variance).

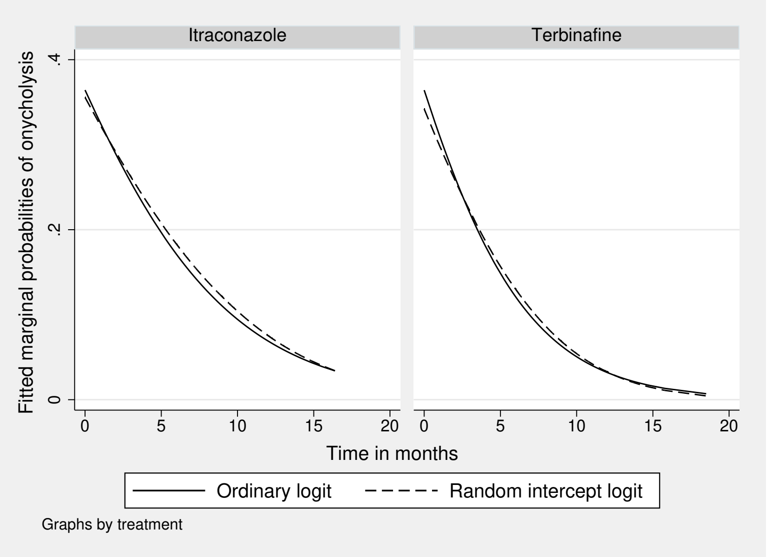


Figure 10.13: Fitted marginal probabilities using ordinary and random-intercept logistic regression

10.12.2 Predicted subject-specific probabilities

Predictions for hypothetical subjects: Conditional probabilities

Subject-specific or conditional predictions of $\widehat{\Pr}(y_{ij} = 1|x_{2j}, x_{3ij}, \zeta_j)$ for different values of ζ_j can be produced using the inverse logit function of the linear predictor. We first use `predict` with the `xb` option to obtain the predicted fixed part of the model,

```
. predict fixedpart, xb
```

and then produce predicted probabilities for ζ_j equal to -4, -2, 0, 2, and 4:

```
. generate condprobm4 = invlogit(fixedpart-4)
. generate condprobm2 = invlogit(fixedpart-2)
. generate condprob0  = invlogit(fixedpart)
. generate condprob2  = invlogit(fixedpart+2)
. generate condprob4  = invlogit(fixedpart+4)
```

Plotting all of these conditional probabilities together with the observed proportions and marginal probabilities produces figure 10.14.

```
. twoway (line prop mn_month, sort)
> (line margprob month, sort lpatt(dash))
> (line condprob0 month, sort lpatt(shortdash_dot))
> (line condprob4 month, sort lpatt(shortdash))
> (line condprobm4 month, sort lpatt(shortdash))
> (line condprob2 month, sort lpatt(shortdash))
> (line condprobm2 month, sort lpatt(shortdash)),
> by(treatment)
> legend(order(1 "Observed proportion" 2 "Marginal probability"
>              3 "Median probability" 4 "Conditional probabilities"))
> xtitle(Time in months) ytitle(Probabilities of onycholysis)
```

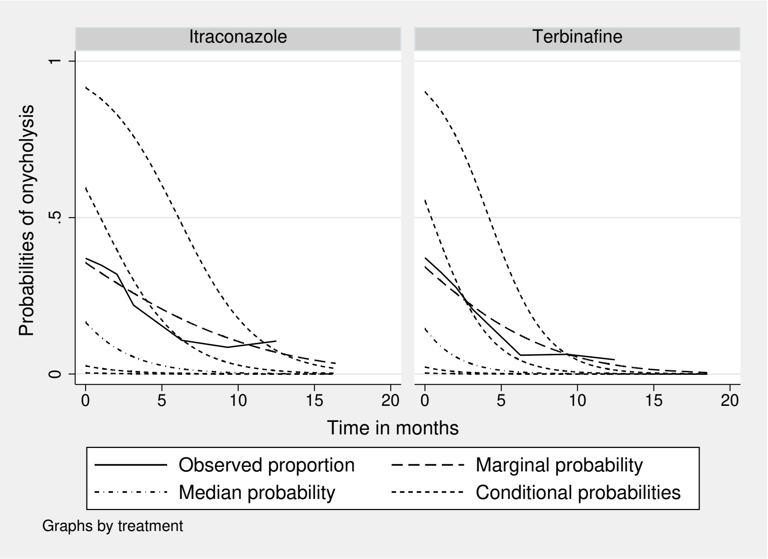


Figure 10.14: Conditional and marginal predicted probabilities for random-intercept logistic regression model

Clearly, the conditional curves have steeper downward slopes than does the marginal curve. The conditional curve represented by a dash-dot line is for $\zeta_j = 0$ and hence represents the population *median* curve.

Predictions for the subjects in the sample: Posterior mean probabilities

We may also want to predict the probability that $y_{ij} = 1$ for a given subject j. The predicted conditional probability, given the unknown random intercept ζ_j, is

$$\widehat{\Pr}(y_{ij} = 1|\mathbf{x}_{ij}, \zeta_j) \;=\; \frac{\exp(\widehat{\beta}_1 + \widehat{\beta}_2 x_{2j} + \widehat{\beta}_3 x_{3ij} + \widehat{\beta}_4 x_{2j} x_{3ij} + \zeta_j)}{1 + \exp(\widehat{\beta}_1 + \widehat{\beta}_2 x_{2j} + \widehat{\beta}_3 x_{3ij} + \widehat{\beta}_4 x_{2j} x_{3ij} + \zeta_j)}$$

Because our knowledge about ζ_j for subject j is represented by the posterior distribution, a good prediction $\widetilde{\pi}_j(\mathbf{x}_{ij})$ of the unconditional probability is obtained by integrating over the posterior distribution:

$$\begin{aligned} \widetilde{\pi}_j(\mathbf{x}_{ij}) &\equiv \int \widehat{\Pr}(y_{ij} = 1|\mathbf{x}_{ij}, \zeta_j) \times \text{Posterior}(\zeta_j | y_{1j}, \ldots, y_{n_j j}, \mathbf{X}_j)\, d\zeta_j \quad (10.11) \\ &\neq \widehat{\Pr}(y_{ij} = 1|\mathbf{x}_{ij}, \widetilde{\zeta}_j) \end{aligned}$$

This minimizes the mean squared error of prediction for known parameters and can be thought of as an empirical Bayes version of *posterior predictive probabilities* where the random intercepts are the only unknown parameters.

We cannot simply plug in the posterior mean of the random intercept $\widetilde{\zeta}_j$ for ζ_j in generalized linear mixed models (for example, by using `melogit` followed by `predict` with the `mu` and `conditional(ebmeans)` options). The reason is that the mean of a given nonlinear function of ζ_j does not in general equal the same function evaluated at the mean of ζ_j.

At the time of writing, `melogit` and `xtlogit` did not have postestimation commands for computing the posterior mean predicted probabilities as defined in (10.12). We therefore use `gllapred` with the `mu` option (and *not* the `marginal` option) after retrieving the `gllamm` estimates:

```
. estimates restore gllamm
(results gllamm are active now)

. gllapred cmu, mu
(mu will be stored in cmu)
Non-adaptive log-likelihood: -625.52573
 -625.3853  -625.3856  -625.3856
log-likelihood:-625.38558
```

`gllapred` can produce predicted posterior mean probabilities also for occasions where the response variable is missing. This is useful for making forecasts for a patient or for making predictions for visits where the patient did not attend the assessment. As we saw in section 10.4, such missing data occur frequently in the toenail data.

Listing `patient` and `visit` for patients 2 and 15,

```
. sort patient visit
. list patient visit if patient==2|patient==15, sepby(patient) noobs

  +-----------------+
  | patient   visit |
  |-----------------|
  |       2       1 |
  |       2       2 |
  |       2       3 |
  |       2       4 |
  |       2       5 |
  |       2       6 |
  |-----------------|
  |      15       1 |
  |      15       2 |
  |      15       3 |
  |      15       4 |
  |      15       5 |
  |      15       7 |
  +-----------------+
```

we see that these patients each have one missing visit: visit 7 is missing for patient 2 and visit 6 is missing for patient 15. To make predictions for these visits, we must first create rows of data (or records) for these visits. A very convenient command to accomplish this is `fillin`:

```
. fillin patient visit
. list patient visit _fillin if patient==2|patient==15, sepby(patient) noobs

  +---------------------------+
  | patient   visit   _fillin |
  |---------------------------|
  |       2       1         0 |
  |       2       2         0 |
  |       2       3         0 |
  |       2       4         0 |
  |       2       5         0 |
  |       2       6         0 |
  |       2       7         1 |
  |---------------------------|
  |      15       1         0 |
  |      15       2         0 |
  |      15       3         0 |
  |      15       4         0 |
  |      15       5         0 |
  |      15       6         1 |
  |      15       7         0 |
  +---------------------------+
```

`fillin` finds all values of `patient` that occur in the data and all values of `visit` and fills in all combinations of these values that do not already occur in the data, for example, patient 2 and visit 7. The command creates a new variable, `_fillin`, taking the value 1 for filled-in records and 0 for records that existed before. All variables have missing values for these new records except `patient`, `visit`, and `_fillin`.

Before we can make predictions, we must fill in values for the covariates: treatment, month, and the interaction trt_month. Note that, by filling in values for covariates, we are not imputing missing data but just specifying for which covariate values we would like to make predictions.

We start by filling in the appropriate values for treatment, taking into account that treatment is a time-constant variable,

```
. egen trt = mean(treatment), by(patient)
. replace treatment = trt if _fillin==1
```

and proceed by filling in the average time (month) associated with the visit number for the time-varying variable month:

```
. drop mn_month
. egen mn_month = mean(month), by(treatment visit)
. replace month = mn_month if _fillin==1
```

Finally, we obtain the filled-in version of the interaction variable, trt_month, by multiplying the variables treatment and month that we have constructed:

```
. replace trt_month = treatment*month
```

It is important that the response variable, outcome, remains missing; the posterior distribution should only be based on the responses that were observed. We also cannot change the covariate values corresponding to these observed responses because that would change the posterior distribution.

We can now make predictions for the entire dataset by repeating the gllapred command (after deleting cmu) with the fsample (for "full sample") option:

```
. drop cmu
. gllapred cmu, mu fsample
(mu will be stored in cmu)
Non-adaptive log-likelihood: -625.52573
 -625.3853  -625.3856  -625.3856
log-likelihood:-625.38558
```

```
. list patient visit _fillin cmu if patient==2|patient==15, sepby(patient) noobs

  +---------------------------------------+
  | patient   visit   _fillin         cmu |
  |---------------------------------------|
  |       2       1         0   .54654227 |
  |       2       2         0   .46888925 |
  |       2       3         0    .3867953 |
  |       2       4         0   .30986966 |
  |       2       5         0   .12102271 |
  |       2       6         0   .05282663 |
  |       2       7         1   .01463992 |
  |---------------------------------------|
  |      15       1         0   .59144346 |
  |      15       2         0   .47716226 |
  |      15       3         0   .39755635 |
  |      15       4         0   .30542907 |
  |      15       5         0   .08992082 |
  |      15       6         1   .01855957 |
  |      15       7         0   .00015355 |
  +---------------------------------------+
```

The predicted forecast probability for visit 7 for patient 2 hence is 0.015.

To look at some patient-specific posterior mean probability curves, we will produce trellis graphs of 16 randomly chosen patients from each treatment group. We will first randomly assign consecutive integer identifiers (1, 2, 3, etc.) to the patients in each group, in a new variable, `randomid`. We will then plot the data for patients with `randomid` 1 through 16 in each group.

To create the random identifier, we first generate a random number from the uniform distribution whenever `visit` is 1 (which happens once for each patient):

```
. set seed 1234421
. sort patient
. generate rand = runiform() if visit==1
```

Here use of the `set seed` and `sort` commands ensures that you get the same values of `randomid` as we do, because the same "seed" is used for the random-number generator. We now define a variable, `randid`, that represents the rank order of `rand` within treatment groups and is missing when `rand` is missing:

```
. by treatment (rand), sort: generate randid = _n if rand<.
```

`randid` is the required random identifier, but it is only available when `visit` is 1 and missing otherwise. We can fill in the missing values by using

```
. egen randomid = mean(randid), by(patient)
```

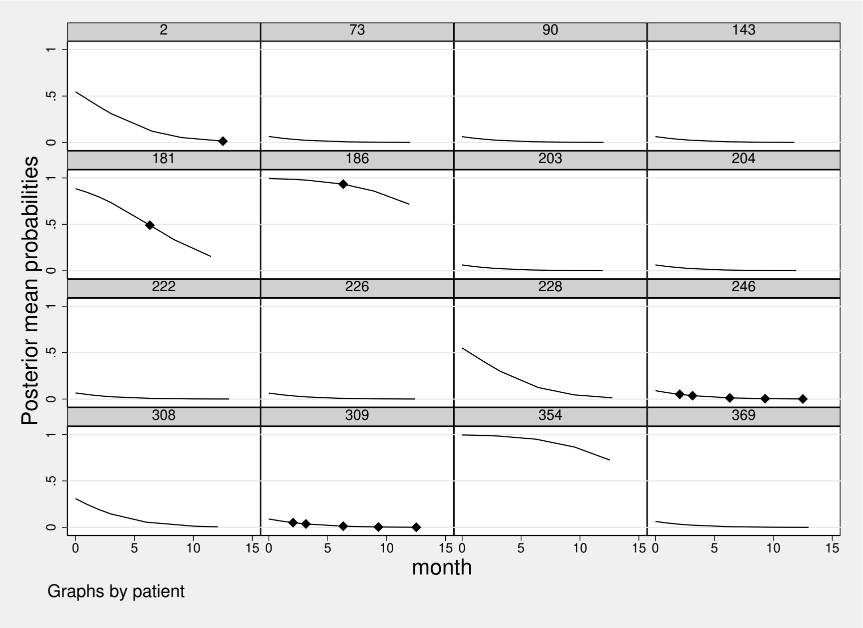
We are now ready to produce the trellis graphs:

```
. twoway (line cmu month, sort) (scatter cmu month if _fillin==1, mcol(black))
> if randomid<=16&treatment==0,  by(patient, compact legend(off)
> l1title("Posterior mean probabilities"))
```

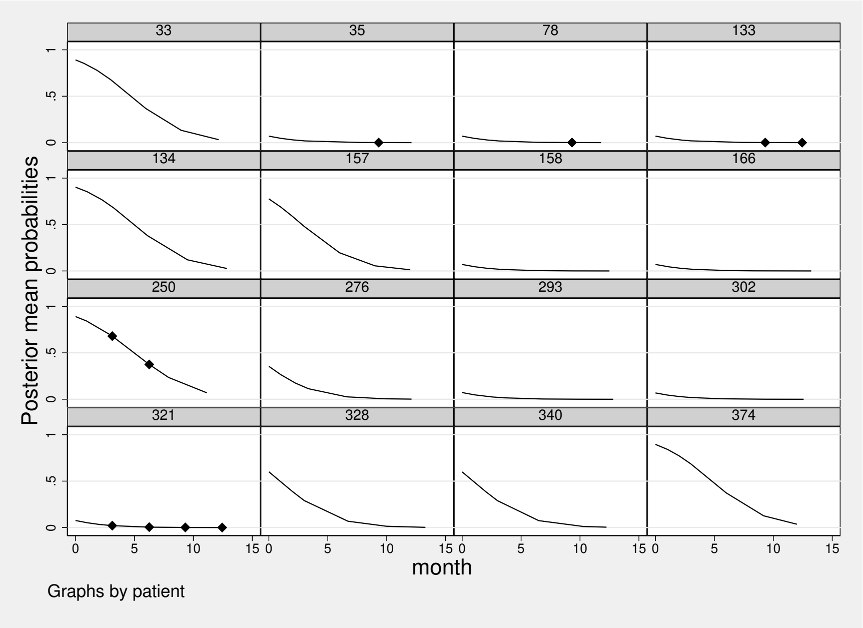
and

```
. twoway (line cmu month, sort) (scatter cmu month if _fillin==1, mcol(black))
> if randomid<=16&treatment==1,  by(patient, compact legend(off)
> l1title("Posterior mean probabilities"))
```

The graphs are shown in figure 10.15. We see that there is considerable variability in the probability trajectories of different patients within the same treatment group.



(a)



(b)

Figure 10.15: Posterior mean probabilities against time for 16 patients in the control group (a) and treatment group (b) with predictions for missing responses shown as diamonds

After estimation with `melogit`, the `predict` command with the options `mu` and `conditional(ebmodes)` gives the posterior mode of the predicted conditional probability $\widehat{\Pr}(y_{ij}|\mathbf{x}_{ij},\zeta_j)$ instead of the posterior mean. This is achieved by substituting the posterior mode of ζ_j into the expression for the conditional probability. [The mode of a strictly increasing function of ζ_j (here an inverse logit), is the same function evaluated at the mode of ζ_j.]

10.13 Other approaches to clustered dichotomous data

10.13.1 Conditional logistic regression

Instead of using random intercepts for clusters (patients in the toenail application), it would be tempting to use fixed intercepts by including a dummy variable for each patient (and omitting the overall intercept). This would be analogous to the fixed-effects estimator of within-patient effects discussed for linear models in section 3.7.2. However, in logistic regression, this approach would lead to inconsistent estimates of the within-patient effects unless the cluster size n is large, due to what is known as the *incidental parameter problem*. Roughly speaking, this problem occurs because the number of cluster-specific intercepts (the incidental parameters) increases in tandem with the sample size (number of clusters), so that the usual asymptotic or large-sample results break down. Obviously, we also cannot eliminate the random intercepts in nonlinear models by simply cluster-mean-centering the responses and covariates, as in (3.12).

Instead, we can eliminate the patient-specific intercepts by constructing a likelihood that is conditional on the number of responses that take the value 1, a sufficient statistic for the patient-specific intercept. This approach is called *conditional* ML estimation, in contrast to *marginal* ML estimation where the patient-specific intercepts are integrated out, the approach used thus far in this chapter and implemented in `melogit`. In linear random-intercept models, conditional ML estimation is equivalent to ordinary least squares estimation of the cluster-mean centered model, that is, fixed-effects estimation as implemented in `xtreg` with the `fe` option. In logistic regression, conditional ML estimation is more involved and is known as *conditional logistic regression*—see display 12.2 for a derivation of the likelihood contribution of a cluster with one response equal to 1. By eliminating the random intercepts, conditional logistic regression relaxes all assumptions regarding the random intercepts, including any patient-level exogeneity assumptions, which is why the method is sometimes referred to as *fixed-effects logistic regression*. Importantly, this method estimates conditional or subject-specific effects, just like random-effects logistic regression, but without assuming that there is no patient-level unobserved confounding. When using conditional logistic regression, we can only estimate the effects of within-patient or time-varying covariates. Patient-specific covariates, such as `treatment`, cannot be included. However, interactions between patient-specific and time-varying variables, such as `treatment` by `month`, can be estimated.

Estimation using clogit

Conditional logistic regression can be performed using Stata's `xtlogit` command with the `fe` option or using the `clogit` command (with the `or` option to obtain ORs):

```
. clogit outcome month i.treatment#c.month, group(patient) or
note: multiple positive outcomes within groups encountered.
note: 179 groups (1,141 obs) omitted because of all positive or
      all negative outcomes.
Conditional (fixed-effects) logistic regression         Number of obs =    767
                                                        LR chi2(2)    = 290.97
                                                        Prob > chi2   = 0.0000
Log likelihood = -188.94377                             Pseudo R2     = 0.4350

------------------------------------------------------------------------------
     outcome | Odds ratio   Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
       month |   .6827717   .0321547    -8.10   0.000     .6225707     .748794
             |
  treatment# |
     c.month |
 Terbinafine |   .9065404   .0667426    -1.33   0.183     .7847274    1.047262
------------------------------------------------------------------------------
```

Interpretation

The subject-specific or conditional OR for the treatment effect (treatment by time interaction) is now estimated as 0.91 and is no longer significant at the 5% level. However, both this estimate and the estimate for `month`, also given in the last column of table 10.2, are quite similar to the marginal ML estimates for the random-intercept model. This is likely to be due to the randomization of treatments in the toenail study, ensuring that they are exogenous.

The subject-specific or conditional ORs from conditional logistic regression represent within-effects, where patients serve as their own controls. As discussed in chapter 5, within-patient estimates cannot be confounded with omitted between-patient covariates and are hence less sensitive to model misspecification than marginal ML estimates for the random-intercept model (which makes the strong assumption that the patient-specific intercepts are independent of the covariates). A further advantage of conditional ML estimation is that it does not make any assumptions regarding the distribution of the patient-specific effect. Therefore, it is reassuring that the conditional ML estimates are fairly similar to the marginal ML estimates.

Skrondal and Rabe-Hesketh (2014a) show that conditional logistic regression may be beneficial when there are missing data. Specifically, conditional ML can be protective against *not* missing at random (NMAR) mechanisms, such as missingness depending on the random intercept and on the (observed or missing) responses.

If the random-intercept model is correct, the marginal ML estimator is more efficient and tends to yield smaller standard errors leading to smaller p-values, as we can see for the treatment by time interaction. Here the conditional logistic regression method is inefficient because, as noted in the output, 179 subjects whose responses were all 0 or all 1 cannot contribute to the analysis. This is because the conditional probabilities of these response patterns, conditioning on the sum of responses across time, are 1 regardless of the covariates (for example, if the sum is 0, all responses must be 0) and the conditional probabilities therefore do not provide any information on covariate effects.

The conditional logistic regression model is sometimes referred to as the Chamberlain fixed-effects logit model in econometrics and is used for matched case–control studies in epidemiology. The same trick of conditioning is also used for the Rasch model in psychometrics and the conditional logit model for discrete choice and nominal responses (see sections 12.2.2 and 12.2.3). Unfortunately, there is no counterpart to conditional logistic regression for probit models.

Note that dynamic models with subject-specific effects cannot be estimated consistently by simply including lagged responses in conditional logistic regression. Also, subject-specific predictions are not possible in conditional logistic regression because no inferences are made regarding the subject-specific intercepts.

10.13.2 Generalized estimating equations (GEE)

Generalized estimating equations (GEE), first introduced in section 6.6, can be used to estimate marginal or population-averaged effects. Dependence among the responses of units in a given cluster is taken into account but treated as a nuisance, whereas this dependence is of central interest in multilevel modeling.

The basic idea of GEE is that an algorithm, known as reweighted iterated least squares, for ML estimation of single-level generalized linear models requires only the mean structure (expectation of the response variable as a function of the covariates) and the variance function. The algorithm iterates between linearizing the model given current parameter estimates and then updating the parameters by using weighted least squares, with weights determined by the variance function. In GEE, this iterative algorithm is extended to two-level data by assuming a within-cluster correlation structure, in addition to the mean structure and variance function, so that the weighted least-squares step becomes a generalized least-squares step (see section 3.10.1), and another step is required for updating the correlation matrix. GEE can be viewed as a special case of generalized methods of moments (GMM) estimation (implemented in Stata's `gmm` command).

In addition to specifying a model for the marginal relationship between the response variable and covariates, it is necessary to choose a structure for the correlations among the observed responses (conditional on covariates). The variance function follows from the Bernoulli distribution. The most common correlation structures are (see section 6.6 for some other correlation structures):

- ***Independence***
  Same as ordinary logistic regression.
- ***Exchangeable***
  Same correlation for all pairs of units.
- ***Autoregressive lag-1*** *[*AR(1)*]*
  Correlation declines exponentially with the time lag—only makes sense for longitudinal data and assumes equally spaced occasions, that is, constant time intervals between occasions, except for gaps due to missing data.
- ***Unstructured***
  A different correlation for each pair of responses—only makes sense if units are not exchangeable within clusters, in the sense that the labels i attached to the units mean the same thing across clusters. For instance, it is meaningful for fixed-occasion longitudinal where the time associated with a given occasion i is identical across individuals, but not for data on students nested in schools where the labels i assigned to students are arbitrary. In addition, each pair of unit labels i and i' must occur sufficiently often across clusters to estimate the pairwise correlations. Finally, the number of different unique unit labels, say, m, should not be too large because the number of parameters is $m(m-1)/2$.

The reason for specifying a correlation structure is that more efficient estimates (with smaller standard errors) are obtained if the specified correlation structure resembles the true dependence structure. Using ordinary logistic regression is equivalent to assuming an independence structure. GEE is therefore generally more efficient than ordinary logistic regression for clustered data although the gain in precision can be meager for balanced data (Lipsitz and Fitzmaurice 2009).

An important feature of GEE (and ordinary logistic regression) is that *marginal effects* can be consistently estimated, even if the dependence among units in clusters is not properly modeled. For this reason, correct specification of the correlation structure is downplayed by using the term "working correlations".

In GEE, the standard errors for the marginal effects are usually based on the robust sandwich estimator, which takes the dependence into account. Use of the sandwich estimator implicitly relies on there being many replications of the responses associated with each distinct combination of covariate values. Otherwise, the estimated standard errors can be biased downward. Furthermore, estimated standard errors based on the sandwich estimator can be very unreliable unless the number of clusters is large (42 or larger as a rough rule of thumb), so in this case model-based (nonrobust) standard errors may be preferable. See Lipsitz and Fitzmaurice (2009) for further discussion.

Estimation using xtgee

We now use GEE to estimate marginal ORs for the toenail data. We request an exchangeable correlation structure (the default) and robust standard errors by using `xtgee` with the `corr(exchangeable)`, `vce(robust)`, and `eform` options:

```
. quietly xtset patient

. xtgee outcome i.treatment##c.month, link(logit)
> family(binomial) corr(exchangeable) vce(robust) eform

GEE population-averaged model                       Number of obs    =  1,908
Group variable: patient                             Number of groups =    294
Family: Binomial                                    Obs per group:
Link:   Logit                                                    min =      1
Correlation: exchangeable                                        avg =    6.5
                                                                 max =      7
                                                    Wald chi2(3)     =  63.44
Scale parameter = 1                                 Prob > chi2      = 0.0000

                              (Std. err. adjusted for clustering on patient)
------------------------------------------------------------------------------
             |               Robust
     outcome | Odds ratio   std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
   treatment |
 Terbinafine |   1.007207   .2618022     0.03   0.978     .6051549    1.676373
       month |   .8425856   .0253208    -5.70   0.000     .7943911     .893704
             |
  treatment#
     c.month |
 Terbinafine |   .9252113   .0501514    -1.43   0.152     .8319576    1.028918
             |
       _cons |   .5588229   .0963122    -3.38   0.001     .3986309    .7833889
------------------------------------------------------------------------------
Note: _cons estimates baseline odds (conditional on zero random effects).
```

Interpretation

These estimates are given under “GEE” in table 10.2 and can alternatively be obtained using `xtlogit` with the `pa` option. Comparing the standard error estimates for GEE with an exchangeable correlation structure to those for ordinary logistic regression (that is, GEE with an independence correlation structure) suggests that specification of a more realistic working correlation structure does not result in much efficiency gain here.

We can display the fitted working correlation matrix by using `estat wcorrelation`:

```
. estat wcorrelation, format(%4.3f)

Estimated within-patient correlation matrix R:

            c1      c2      c3      c4      c5      c6      c7
--------------------------------------------------------------
  r1     1.000
  r2     0.422   1.000
  r3     0.422   0.422   1.000
  r4     0.422   0.422   0.422   1.000
  r5     0.422   0.422   0.422   0.422   1.000
  r6     0.422   0.422   0.422   0.422   0.422   1.000
  r7     0.422   0.422   0.422   0.422   0.422   0.422   1.000
```

A problem with the exchangeable correlation structure is that the true marginal (over the random effects) correlation of the responses is in general not constant but varies according to values of the observed covariates. Using Pearson correlations for dichotomous responses is also somewhat peculiar because the OR is the measure of association in logistic regression.

GEE is an *estimation method* that does not require the specification of a full statistical model. While the mean structure, variance function, and correlation structure are specified, it is often not be possible to find a statistical model with such a structure. As we already pointed out, it may not be possible to specify a model for binary responses where the residual Pearson correlation matrix is exchangeable. For this reason, the approach is called an estimating equation approach rather than a modeling approach. This is in stark contrast to multilevel modeling, where statistical models are explicitly specified.

The fact that no full statistical model is specified has three important implications. First, there is no likelihood and therefore likelihood-ratio tests cannot be used. Instead, comparison of nested models typically proceeds by using Wald tests. Unless the sample size is large, this approach may be problematic because it is known that these tests do not work as well as likelihood-ratio tests in ordinary logistic regression. Second, it is not possible to simulate or predict individual responses based on the estimates from GEE (see section 10.12.2 for prediction and forecasting based on multilevel models). Third, GEE does not share the useful property of ML that estimators are consistent when data are MAR. Although GEE produces consistent estimates of marginal effects if the probability of responses being missing is covariate dependent [and for the special case of responses missing completely at random (MCAR)], it produces inconsistent estimates if the probability of a response being missing for a unit depends on observed responses for other units in the same cluster. Such missingness is likely to occur in longitudinal data where dropout could depend on a subjects' previous responses (see sections 5.9.1 and 13.12).

10.14 Summary and further reading

We described various approaches to modeling clustered dichotomous data, focusing on random-intercept logistic models for longitudinal data. Alternatives to multilevel modeling, such as conditional ML estimation and GEE, were also briefly discussed. The important distinction between conditional or subject-specific effects and marginal or population-averaged effects was emphasized.

We described adaptive quadrature for marginal ML estimation. We recommended that you compare a sequence of estimates with an increasing number of quadrature (or integration) points and establish that the estimates stabilize before concluding that you have used enough quadrature points for a given model and application. We demonstrated the use of a variety of predictions, either cluster-specific predictions, based on empirical Bayes, or population-averaged predictions. Keep in mind that consistent estimation in logistic regression models with random effects, in principle, requires a completely correct model specification. Diagnostics for generalized linear mixed models are still being developed (see, for example, Breinegaard, Rabe-Hesketh, and Skrondal (2017, 2018) and the references therein).

We did not cover random-coefficient models for binary responses in this chapter but have included two exercises (10.3 and 10.8), with solutions provided, involving these models. The issues discussed in chapter 4 regarding linear models with random coefficients are also relevant for other generalized linear mixed models. The syntax for random-coefficient logistic models is analogous to the syntax for linear random-coefficient models except that `mixed` is replaced with `melogit` and `gllamm` is used with a different link function and distribution (the syntax for linear random-coefficient models in `gllamm` can be found in the `gllamm` companion). Three-level random-coefficient logistic models for binary responses are discussed in chapter 16. In chapter 11, `meologit` is used to fit random-coefficient ordinal logistic regression models; see section 11.7.1.

Dynamic or lagged-response models for binary responses were not discussed in this chapter. Such models, sometimes called transition models, can suffer from similar kinds of endogeneity problems as those discussed for dynamic models with random intercepts in section 5.8 of volume 1. Skrondal and Rabe-Hesketh (2014b) review several approaches for handling these endogeneity problems. The paper provides a link to a do-file for the Wooldridge (2005) solution using `melogit` or `meprobit` and for the Heckman (1981) approach using `gllamm`. Rabe-Hesketh and Skrondal (2013) point out that Wooldridge's solution has often been implemented incorrectly, and their proposed correct version is implemented in the community-contributed Stata program `xtpdyn` by Grotti and Cutuli (2018).

We discussed the most common link functions for dichotomous responses, namely, logit and probit links. A third link sometimes used is the complementary log–log link, which is introduced in section 14.6. Dichotomous responses are sometimes aggregated into counts, giving the number of successes y_i in m_i trials for unit i. In this situation, it is usually assumed that y_i has a binomial(m_i,π_i) distribution. `melogit` can then be used as for dichotomous responses but with the `binomial()` option to specify the variable containing the values m_i. Similarly, `gllamm` can be used with the binomial distribution and any of the link functions together with the `denom()` option to specify the variable containing m_i.

Good introductions to single-level logistic regression include Collett (2003), Long (1997), and Hosmer, Lemeshow, and Sturdivant (2013). Logistic and other types of regression using Stata are discussed by Long and Freese (2014), primarily with examples from social science, and by Vittinghoff et al. (2012), with examples from medicine.

Generalized linear mixed models are described in the books by McCulloch, Searle, and Neuhaus (2008), Skrondal and Rabe-Hesketh (2004), Molenberghs and Verbeke (2005), and Hedeker and Gibbons (2006). See also Goldstein (2011), Raudenbush and Bryk (2002), and volume 3 of the anthology by Skrondal and Rabe-Hesketh (2010). Several examples with dichotomous responses are discussed in Skrondal and Rabe-Hesketh (2004, chap. 9). Guo and Zhao (2000) is a good introductory paper on multilevel modeling of binary data with applications in social science. We also recommend the book chapter by Rabe-Hesketh and Skrondal (2009), the article by Agresti et al. (2000), and the encyclopedia entry by Hedeker (2005) for overviews of generalized linear mixed models. Prediction of random effects and outcomes in generalized linear mixed models is treated in Skrondal and Rabe-Hesketh (2009).

A classic paper on conditional logistic regression for longitudinal data is Chamberlain (1980). Skrondal and Rabe-Hesketh (2014a) discuss advantages of conditional logistic regression when there are missing data. Detailed accounts of GEE are given in Hardin and Hilbe (2013), Diggle et al. (2002), and Lipsitz and Fitzmaurice (2009).

Exercises 10.1, 10.2, 10.3, and 10.6 are on longitudinal or panel data. There are also exercises on cross-sectional datasets on students nested in schools (10.7 and 10.8), cows nested in herds (10.5), questions nested in respondents (10.4) and wine bottles nested in judges (10.9). Exercise 10.2 involves GEE, whereas exercises 10.4 and 10.6 involve conditional logistic regression. The latter exercise also asks you to perform a Hausman test. Exercises 10.3 and 10.8 consider random-coefficient models for dichotomous responses (solutions are provided for both exercises). Exercise 10.4 introduces the idea of item-response theory, and exercise 10.8 shows how `melogit` and `gllamm` can be used to fit multilevel models with survey weights.

10.15 Exercises

10.1 Toenail data

1. Fit the probit version of the random-intercept model in (10.6) with `meprobit`. How many quadrature points appear to be needed?
2. Estimate the residual intraclass correlation for the latent responses, both by plugging the estimates into the appropriate expression and by using the appropriate postestimation command.
3. Obtain empirical Bayes predictions of the random intercepts for both the logit and probit models and estimate the approximate constant of proportionality between these.
4. ❖ By considering the residual standard deviations of the latent response for the logit and probit models, work out what you think the constant of proportionality should be for the logit- and probit-based empirical Bayes predictions. How does this compare with the constant estimated in step 3?

10.2 Ohio-wheeze data

In this exercise, we use data from the Six Cities Study (Ware et al. 1984), previously analyzed by Fitzmaurice (1998), among others. The dataset includes 537 children from Steubenville, Ohio, who were examined annually four times from age 7 to age 10 to ascertain their wheezing status. The smoking status of the mother was also determined at the beginning of the study to investigate whether maternal smoking increases the risk of wheezing in children. The mother's smoking status is treated as time constant, although it may have changed for some mothers over time.

The dataset `wheeze.dta` has the following variables:

- `id`: child identifier (j)
- `age`: number of years since ninth birthday (x_{2ij})
- `smoking`: mother smokes regularly (1: yes; 0: no) (x_{3j})
- `y`: wheeze status (1: yes; 0: no) (y_{ij})

1. Fit the following transition model considered by Fitzmaurice (1998):

   $$\text{logit}\{\Pr(y_{ij}\!=\!1|\mathbf{x}_{ij}, y_{i-1,j})\} \;=\; \beta_1 + \beta_2 x_{2ij} + \beta_3 x_{3j} + \gamma y_{i-1,j}, \quad i = 2,3,4$$

   where x_{2ij} is `age` and x_{3j} is `smoking`. (The lagged responses can be included in the model by first typing `xtset id age` and then including `L.y` as a covariate in the model.)
2. Fit the following random-intercept model considered by Fitzmaurice (1998):

   $$\text{logit}\{\Pr(y_{ij}\!=\!1|\mathbf{x}_{ij}, \zeta_j)\} \;=\; \beta_1 + \beta_2 x_{2ij} + \beta_3 x_{3j} + \zeta_j, \quad i = 1,2,3,4$$

   It is assumed that $\zeta_j|\mathbf{x}_{ij} \sim \text{N}(0,\psi)$ and that ζ_j is independent across children.
3. Use GEE to fit the marginal model

   $$\text{logit}\{\Pr(y_{ij}\!=\!1|\mathbf{x}_{ij})\} \;=\; \beta_1 + \beta_2 x_{2ij} + \beta_3 x_{3j}, \quad i = 1,2,3,4$$

   specifying an unstructured correlation matrix (`xtset` the data using `xtset id age`). Try some other correlation structures and compare the fit (using `estat wcorrelation`) to the unstructured version.
4. Interpret the estimated effects of mother's smoking status for the models in steps 1, 2, and 3.

10.3 Vaginal-bleeding data Solutions

Fitzmaurice, Laird, and Ware (2011) analyzed data from a trial reported by Machin et al. (1988). Women were randomized to receive an injection of either 100 mg or 150 mg of the long-lasting injectable contraception depot medroxyprogesterone acetate (DMPA) at the start of the trial and at three successive 90-day intervals. In addition, the women were followed up 90 days after the final injection. Throughout the study, each woman completed a menstrual diary that recorded any vaginal bleeding-pattern disturbances. The diary data were used to determine whether a woman experienced amenorrhea, defined as the absence of menstrual bleeding for at least 80 consecutive days.

The response variable for each of the four 90-day intervals is whether the woman experienced amenorrhea during the interval. Data are available on 1,151 women for the first interval, but there was considerable dropout after that.

The dataset `amenorrhea.dta` has the following variables:

- `dose`: high dose (1: yes; 0: no)
- `y1`–`y4`: responses for time intervals 1–4 (1: amenorrhea; 0: no amenorrhea)
- `wt2`: number of women with the same dose level and response pattern

1. Produce an identifier variable for women, and reshape the data to long form, stacking the responses `y1`–`y4` into one variable and creating a new variable, `occasion`, taking the values 1–4 for each woman.
2. Fit the following model considered by Fitzmaurice, Laird, and Ware (2011):

   $$\text{logit}\{\Pr(y_{ij}=1|x_j,t_{ij},\zeta_j)\} \;=\; \beta_1+\beta_2 t_{ij}+\beta_3 t_{ij}^2+\beta_4 x_j t_{ij}+\beta_5 x_j t_{ij}^2+\zeta_j$$

   where $t_{ij}=1,2,3,4$ is the time interval and x_j is `dose`. It is assumed that $\zeta_j \sim N(0,\psi)$ and that ζ_j is independent across women and independent of x_j and t_{ij}. Use `melogit` with the `fweight(wt2)` option to specify that `wt2` are level-2 frequency weights.
3. Write down the above model, adding a random slope of t_{ij}, and fit the extended model. (See section 11.7.1 for an example of a random-coefficient model for ordinal responses fit in `meologit`.)
4. Interpret the estimated coefficients.
5. Plot marginal predicted probabilities as a function of time, separately for women in the two treatment groups.

10.4 Verbal-aggression data

De Boeck and Wilson (2004) discuss a dataset from Vansteelandt (2000) where 316 participants were asked to imagine the following four frustrating situations where either another or oneself is to blame:

1. Bus: A bus fails to stop for me (another to blame)
2. Train: I miss a train because a clerk gave me faulty information (another to blame)
3. Store: The grocery store closes just as I am about to enter (self to blame)
4. Operator: The operator disconnects me when I have used up my last 10 cents for a call (self to blame)

For each situation, the participant was asked if it was true (yes, perhaps, or no) that

1. I would (want to) curse
2. I would (want to) scold
3. I would (want to) shout

For each of the three behaviors above, the words "want to" were both included and omitted, yielding six statements with a 3 × 2 factorial design (3 behaviors in 2 modes) combined with the four situations. Thus, there were 24 items in total.

The dataset `aggression.dta` contains the following variables:

- `person`: subject identifier
- `item`: item (or question) identifier
- `description`: item description
  (situation: bus/train/store/operator; behavior: curse/scold/shout; mode: do/want)
- `i1`–`i24`: dummy variables for the items, for example, `i5` equals 1 when `item` equals 5 and 0 otherwise
- `y`: ordinal response (0: no; 1: perhaps; 2: yes)
- Person characteristics:
  - `anger`: trait anger score (STAXI, Spielberger [1988]) (w_{1j})
  - `gender`: dummy variable for being male (1: male; 0: female) (w_{2j})
- Item characteristics:
  - `do_want`: dummy variable for mode being "do" (that is, omitting words "want to") versus "want" (x_{2ij})
  - `other_self`: dummy variable for others to blame versus self to blame (x_{3ij})
  - `blame`: variable equal to 0.5 for blaming behaviors curse and scold and −1 for shout (x_{4ij})
  - `express`: variable equal to 0.5 for expressive behaviors curse and shout and −1 for scold (x_{5ij})

1. Recode the ordinal response variable `y` so that either a "2" or a "1" for the original variable becomes a "1" for the recoded variable.
2. De Boeck and Wilson (2004, sec. 2.5) consider the following "explanatory item-response model" for the dichotomous response:

$$\text{logit}\{\Pr(y_{ij}=1|\mathbf{x}_{ij},\zeta_j)\} \;=\; \beta_1+\beta_2x_{2ij}+\beta_3x_{3ij}+\beta_4x_{4ij}+\beta_5x_{5ij}+\zeta_j$$

   where $\zeta_j \sim N(0,\psi)$ can be interpreted as the latent trait "verbal aggressiveness". Fit this model using `melogit`, and interpret the estimated coefficients. In De Boeck and Wilson (2004), the first five terms have minus signs, so their estimated coefficients have the opposite sign.
3. De Boeck and Wilson (2004, sec. 2.6) extend the above model by including a latent regression, allowing verbal aggressiveness (now denoted η_j instead of ζ_j) to depend on the person characteristics w_{1j} and w_{2j}:

$$\begin{aligned}\text{logit}\{\Pr(y_{ij}=1|\mathbf{x}_{ij},\eta_j)\} &= \beta_1+\beta_2x_{2ij}+\beta_3x_{3ij}+\beta_4x_{4ij}+\beta_5x_{5ij}+\eta_j\\ \eta_j &= \gamma_1w_{1j}+\gamma_2w_{2j}+\zeta_j\end{aligned}$$

   Substitute the level-2 model for η_j into the level-1 model for the item responses, and fit the model using `melogit`.

4. Use `melogit` to fit the "descriptive item-response model", usually called a one-parameter logistic item response theory (IRT) model or *Rasch model*, considered by De Boeck and Wilson (2004, sec. 2.3):

$$\text{logit}\{\Pr(y_{ij}=1|d_{1i},\ldots,d_{24,i},\zeta_j)\} \;=\; \sum_{m=1}^{24}\beta_m d_{mi} + \zeta_j$$

   where d_{mi} is a dummy variable for item i, with $d_{mi} = 1$ if $m = i$ and 0 otherwise. In De Boeck and Wilson (2004), the first term has a minus sign, so their β_m coefficients have the opposite sign; see also their page 53.
5. The model above is known as a one-parameter item-response model because there is one parameter β_m for each item. The negative of these item-specific parameters, $-\beta_m$, can be interpreted as "difficulties"; the larger $-\beta_m$, the larger the latent trait (here verbal aggressiveness, but often ability) has to be to yield a given probability (for example, 0.5) of a 1 response.

   Sort the items in increasing order of the estimated difficulties. For the least and most difficult items, look up the variable `description`, and discuss whether it makes sense that these items are particularly easy and hard to endorse (requiring little and a lot of verbal aggressiveness), respectively.
6. Replace the random intercepts ζ_j with fixed parameters α_j. Set the difficulty of the first item to 0 for identification and fit the model by conditional ML. Verify that differences between estimated difficulties for the items are similar to those in step 4.
7. ❖ Obtain empirical Bayes (also called EAP) predictions and empirical Bayes modal (also called MAP) predictions, and ML estimates of the latent trait. Also obtain standard errors (for ML, this means saving `_se[_cons]` in addition to `_b[_cons]` by adding `mlse = _se[_cons]` in the `statsby` command). Does there appear to be much shrinkage? Calculate the total score (sum of item responses) for each person and plot the different kinds of standard errors against the total score, connecting adjacent points. Comment on what you find.

See also exercise 11.2 for further analyses of these data.

10.5 Dairy-cow data

Dohoo et al. (2001) and Dohoo, Martin, and Stryhn (2010) analyzed data on dairy cows from Reunion Island. One outcome considered was the "risk" of conception at the first insemination attempt (first service) since the previous calving. This outcome was available for several lactations (calvings) per cow.

The variables in the dataset `dairy.dta` used here are

- `cow`: cow identifier
- `herd`: herd identifier
- `region`: geographic region

- fscr: first-service conception risk (dummy variable for cow becoming pregnant)
- lncfs: log of time interval (in log days) between calving and first service (insemination attempt)
- ai: dummy variable for artificial insemination being used (versus natural) at first service
- heifer: dummy variable for being a young cow that has calved only once

1. Fit a two-level random-intercept logistic regression model for the response variable fscr, an indicator for conception at the first insemination attempt (first service). Include a random intercept for cow and the covariates lncfs, ai, and heifer. (Use either melogit, xtlogit, or gllamm.)
2. Obtain estimated ORs with 95% confidence intervals for the covariates and interpret them.
3. Obtain the estimated residual intraclass correlation between the latent responses for two observations on the same cow.
4. Obtain the estimated median OR for two randomly chosen cows with the same covariates, comparing the cow that has the larger random intercept with the cow that has the smaller random intercept. Is there much variability in the cows' fertility?

See also exercises 8.8 and 16.1.

10.6 Union-membership data

Vella and Verbeek (1998) analyzed panel data on 545 young males taken from the U.S. National Longitudinal Survey (Youth Sample) for the period 1980–1987. In this exercise, we will focus on modeling whether the men were members of unions or not.

The dataset wagepan.dta was provided by Wooldridge (2010) and was previously used in exercise 3.6 and *Part III: Introduction to models for longitudinal and panel data* (in volume 1). The subset of variables considered here is

- nr: person identifier (j)
- year: 1980–1987 (i)
- union: dummy variable for being a member of a union (that is, wage being set in collective bargaining agreement) (y_{ij})
- educ: years of schooling (x_{2j})
- black: dummy variable for being Black (x_{3j})
- hisp: dummy variable for being Hispanic (x_{4j})
- exper: labor market experience, defined as age−6−educ (x_{5ij})
- married: dummy variable for being married (x_{6ij})
- rur: dummy variable for living in a rural area (x_{7ij})
- nrtheast: dummy variable for living in Northeast (x_{8ij})

- `nrthcen`: dummy variable for living in Northern Central (x_{9ij})
- `south`: dummy variable for living in South ($x_{10,ij}$)

You can use the `describe` command to get a description of the other variables in the file.

1. Use ML estimation to fit the random-intercept logistic regression model

   $$\mathrm{logit}\{\Pr(y_{ij}=1|\mathbf{x}_{ij},\zeta_j)\} \;=\; \beta_1+\beta_2x_{2j}+\cdots+\beta_{11}x_{10,ij}+\zeta_j$$

   where $\zeta_j \sim N(0,\psi)$, and ζ_j is assumed to be independent across persons and independent of $\mathbf{x}_{ij}$. Use `xtlogit` here.
2. Interpret the estimated effects of the covariates from step 1 in terms of ORs, and report the estimated residual intraclass correlation of the latent responses.
3. Fit the marginal model

   $$\mathrm{logit}\{\Pr(y_{ij}=1|\mathbf{x}_{ij})\} \;=\; \beta_1+\beta_2x_{2j}+\cdots+\beta_{11}x_{10,ij}$$

   using GEE with an exchangeable working correlation structure.
4. Interpret the estimated effects of the covariates from step 3 in terms of ORs, and compare these estimates with those from step 1. Why are the estimates different?
5. Explore the within and between variability of the response variable and covariates listed above. For which of the covariates is it impossible to estimate an effect using a fixed-effects approach? Are there any covariates whose effects you would expect to be imprecisely estimated when using a fixed-effects approach?
6. Use conditional ML estimation to fit the fixed-intercept logistic regression model

   $$\mathrm{logit}\{\Pr(y_{ij}=1|\mathbf{x}_{ij})\} \;=\; \beta_1+\beta_2x_{2j}+\cdots+\beta_{11}x_{10,ij}+\alpha_j$$

   where the α_j are unknown person-specific parameters.
7. Interpret the estimated effects of the covariates from step 6 in terms of ORs, and compare these estimates with those from step 1. Why are the estimates different?
8. Perform a Hausman test to assess the validity of the random-intercept model. What do you conclude?
9. Fit the probit versions of the random-intercept model from step 1 using `xtprobit`. Which type of model do you find easiest to interpret?

10.7 School-retention-in-Thailand data

A national survey of primary education was conducted in Thailand in 1988. The data were previously analyzed by Raudenbush and Bhumirat (1992) and are distributed with the HLM software (Raudenbush et al. 2019). Here we will model the probability that a child repeats a grade any time during primary school.

The dataset `thailand.dta` has the following variables:

- `rep`: dummy variable for child having repeated a grade during primary school (y_{ij})
- `schoolid`: school identifier (j)
- `pped`: dummy variable for child having preprimary experience (x_{2ij})
- `male`: dummy variable for child being male (x_{3ij})
- `mses`: school mean socioeconomic status (SES) (x_{4j})
- `wt1`: number of children in the school having a given set of values of `rep`, `pped`, and `male` (level-1 frequency weights)

1. Fit the model

$$\text{logit}\{\Pr(y_{ij} = 1|\mathbf{x}_{ij}, \zeta_j)\} \;=\; \beta_1 + \beta_2 x_{2ij} + \beta_3 x_{3ij} + \beta_4 x_{4j} + \zeta_j$$

where $\zeta_j \sim N(0, \psi)$, and ζ_j is independent across schools and independent of the covariates $\mathbf{x}_{ij}$. Use `gllamm` with the `weight(wt)` option to specify that each row in the data represents `wt1` children (level-1 units).

2. Obtain and interpret the estimated ORs and the estimated residual intraschool correlation of the latent responses.
3. Use `gllapred` to obtain empirical Bayes predictions of the probability of repeating a grade. These probabilities will be specific to the schools, as well as dependent on the student-level predictors.

    a. List the values of `male`, `pped`, `rep`, `wt1`, and the predicted probabilities for the school with `schoolid` equal to 10104. Explain why the predicted probabilities are greater than 0 although none of the children in the sample from that school have been retained. For comparison, list the same variables for the school with `schoolid` equal to 10105.

    b. Produce box plots of the predicted probabilities for each school by `male` and `pped` (for instance, using `by(male)` and `over(pped)`). To ensure that each school contributes no more than four probabilities to the graph (one for each combination of the student-level covariates), use only responses where `rep` is 0 (that is, `if rep==0`). Do the schools appear to be variable in their retention probabilities?

10.8 PISA data Solutions

Here we consider data from the Program for International Student Assessment (PISA) 2000 conducted by the Organization for Economic Cooperation and Development (OECD 2000) that are made available with permission from Mariann Lemke. The survey assessed educational attainment of 15-year-olds in 43 countries in various areas, with an emphasis on reading. Following Rabe-Hesketh and Skrondal (2006), we will analyze reading proficiency, treated as dichotomous (1: proficient; 0: not proficient), for the U.S. sample.

The variables in the dataset `pisaUSA2000.dta` are

- `id_school`: school identifier
- `pass_read`: dummy variable for being proficient in reading
- `female`: dummy variable for student being female
- `isei`: international socioeconomic index
- `high_school`: dummy variable for highest education level by either parent being high school
- `college`: dummy variable for highest education level by either parent being college
- `test_lang`: dummy variable for test language (English) being spoken at home
- `one_for`: dummy variable for one parent being foreign born
- `both_for`: dummy variable for both parents being foreign born
- `w_fstuwt`: student-level or level-1 survey weights
- `wnrschbq`: school-level or level-2 survey weights

1. Fit a logistic regression model with `pass_read` as the response variable and the variables `female` to `both_for` above as covariates and with a random intercept for schools using `melogit`. (Use the default seven quadrature points.)
2. Fit the model from step 1 with the school mean of `isei` as an additional covariate.
3. Interpret the estimated coefficients of `isei` and school mean `isei` and comment on the change in the other parameter estimates due to adding school mean `isei`.
4. From the estimates in step 2, obtain an estimate of the between-school effect of socioeconomic status.
5. Rerun the command but this time with robust standard errors.
6. ❖ In this survey, schools were sampled with unequal probabilities, π_j, and given that a school was sampled, students were sampled from the school with unequal probabilities $\pi_{i|j}$. The reciprocals of these probabilities are given as school- and student-level survey weights, `wnrschbg` ($w_j = 1/\pi_j$) and `w_fstuwt` ($w_{i|j} = 1/\pi_{i|j}$), respectively. As discussed in Rabe-Hesketh and Skrondal (2006), incorporating survey weights in multilevel models using a so-called *pseudolikelihood* approach can lead to biased estimates, particularly

if the level-1 weights $w_{i|j}$ are substantially different from 1 and if the cluster sizes are small. Neither of these issues arises here, so implement pseudo ML estimation as follows:

a. Rescale the student-level weights by dividing them by their cluster means [this is scaling method 2 in Rabe-Hesketh and Skrondal (2006)].
b. Rename the level-2 weights and rescaled level-1 weights to `wt2` and `wt1`, respectively.
c. Run the `melogit` command from step 2 above, adding `[pw=wt1]` before `||` to specify level-1 weights and the additional option `pweight(wt2)` to specify level-2 weights.
d. Compare the estimates with those from step 2. Robust standard errors are computed by `melogit` because model-based standard errors are not appropriate with survey weights.

For useful discussions of the use of survey weights in multilevel modeling, we refer to Rabe-Hesketh and Skrondal (2006) and Snijders and Bosker (2012, chap. 14).

10.9 Wine-tasting data

Tutz and Hennevogl (1996) and Fahrmeir and Tutz (2001) analyzed data on the bitterness of white wines from Randall (1989).

The dataset `wine.dta` has the following variables:

- `bitter`: dummy variable for bottle being classified as bitter (y_{ij})
- `judge`: judge identifier (j)
- `temp`: temperature (low=1; high=0) x_{2ij}
- `contact`: skin contact when pressing the grapes (yes=1; no=0) x_{3ij}
- `repl`: replication

Interest concerns whether conditions that can be controlled while pressing the grapes, such as temperature and skin contact, influence the bitterness. For each combination of temperature and skin contact, two bottles of white wine were randomly chosen. The bitterness of each bottle was rated by the same nine judges, who were selected and trained for the ability to detect bitterness. Here we consider the binary response "bitter" or "nonbitter".

To allow the judgment of bitterness to vary between judges, a random-intercept logistic model is specified

$$\ln\left\{\frac{\Pr(y_{ij}=1|x_{2ij},x_{3ij},\zeta_j)}{\Pr(y_{ij}=0|x_{2ij},x_{3ij},\zeta_j)}\right\} = \beta_1 + \beta_2 x_{2ij} + \beta_3 x_{3ij} + \zeta_j$$

where $\zeta_j \sim N(0,\psi)$. The random intercepts are assumed to be independent across judges and independent of the covariates x_{2ij} and x_{3ij}. ML estimates and estimated standard errors for the model are given in table 10.3 below.

Table 10.3: ML estimates for bitterness model

	Est	(SE)
Fixed part		
β_1	−1.50	(0.90)
β_2	4.26	(1.30)
β_3	2.63	(1.00)
Random part		
ψ	2.80	
Log likelihood	−25.86	

1. Interpret the estimated effects of the covariates as ORs.
2. State the expression for the residual intraclass correlation of the latent responses for the above model and estimate this intraclass correlation.
3. Consider two bottles characterized by the same covariates and judged by two randomly chosen judges. Estimate the median OR comparing the judge who has the larger random intercept with the judge who has the smaller random intercept.
4. ❖ Based on the estimates given in table 10.3, provide an approximate estimate of ψ if a probit model is used instead of a logit model. Assume that the estimated residual intraclass correlation of the latent responses is the same as for the logit model.
5. ❖ Based on the estimates given in the table, provide approximate estimates for the marginal effects of x_{2ij} and x_{3ij} in an ordinary logistic regression model (without any random effects).

See also exercise 11.8 for further analysis of these data.

10.10 ❖ Random-intercept probit model

In a hypothetical study, an ordinary probit model was fit for students clustered in schools. The response was whether students gave the right answer to a question, and the single covariate was socioeconomic status (SES). The intercept and regression coefficient of SES were estimated as $\widehat{\beta}_1 = 0.2$ and $\widehat{\beta}_2 = 1.6$, respectively. The analysis was then repeated, this time including a normally distributed random intercept for school with variance estimated as $\widehat{\psi} = 0.15$.

1. Using a latent-response formulation for the random-intercept probit model, derive the marginal probability that $y_{ij} = 1$ given SES. See section 10.2.2 and remember to replace ϵ_{ij} with $\zeta_j + \epsilon_{ij}$.
2. Obtain the values of the estimated school-specific regression coefficients for the random-intercept probit model.
3. Obtain the estimated residual intraclass correlation for the latent responses.