CHAPTER 1

Why?

1.1 What is multilevel regression modeling?

Consider an educational study with data from students in many schools, predicting in each school the students' grades y on a standardized test given their scores on a pre-test x and other information. A separate regression model can be fit within each school, and the parameters from these schools can themselves be modeled as depending on school characteristics (such as the socioeconomic status of the school's neighborhood, whether the school is public or private, and so on). The student-level regression and the school-level regression here are the two levels of a *multilevel model*.

In this example, a multilevel model can be expressed in (at least) three equivalent ways as a student-level regression:

- A model in which the coefficients vary by school (thus, instead of a model such as $y = \alpha + \beta x + \text{error}$, we have $y = \alpha_j + \beta_j x + \text{error}$, where the subscripts j index schools),
- A model with more than one variance component (student-level and school-level variation),
- A regression with many predictors, including an indicator variable for each school in the data.

More generally, we consider a multilevel model to be a regression (a linear or generalized linear model) in which the parameters—the regression coefficients—are given a probability model. This second-level model has parameters of its own—the *hyperparameters* of the model—which are also estimated from data.

The two key parts of a multilevel model are varying coefficients, and a model for those varying coefficients (which can itself include group-level predictors). Classical regression can sometimes accommodate varying coefficients by using indicator variables. The feature that distinguishes multilevel models from classical regression is in the modeling of the variation between groups.

Models for regression coefficients

To give a preview of our notation, we write the regression equations for two multilevel models. To keep notation simple, we assume just one student-level predictor x (for example, a pre-test score) and one school-level predictor u (for example, average parents' incomes).

Varying-intercept model. First we write the model in which the regressions have the same slope in each of the schools, and only the intercepts vary. We use the

notation i for individual students and $j[i]$ for the school j containing student i:[1]

$$\begin{aligned} y_i &= \alpha_{j[i]} + \beta x_i + \epsilon_i, \quad \text{for students } i = 1, \ldots, n \\ \alpha_j &= a + b u_j + \eta_j, \quad \text{for schools } j = 1, \ldots, J. \end{aligned} \tag{1.1}$$

Here, x_i and u_j represent predictors at the student and school levels, respectively, and ϵ_i and η_j are independent error terms at each of the two levels. The model can be written in several other equivalent ways, as we discuss in Section 12.5.

The number of "data points" J (here, schools) in the higher-level regression is typically much less than n, the sample size of the lower-level model (for students in this example).

Varying-intercept, varying-slope model. More complicated is the model where intercepts and slopes both can vary by school:

$$\begin{aligned} y_i &= \alpha_{j[i]} + \beta_{j[i]} x_i + \epsilon_i, \quad \text{for students } i = 1, \ldots, n \\ \alpha_j &= a_0 + b_0 u_j + \eta_{j1}, \quad \text{for schools } j = 1, \ldots, J \\ \beta_j &= a_1 + b_1 u_j + \eta_{j2}, \quad \text{for schools } j = 1, \ldots, J. \end{aligned}$$

Compared to model (1.1), this has twice as many vectors of varying coefficients (α, β), twice as many vectors of second-level coefficients (a, b), and potentially correlated second-level errors η_1, η_2. We will be able to handle these complications.

Labels

"Multilevel" or "hierarchical." Multilevel models are also called *hierarchical*, for two different reasons: first, from the structure of the data (for example, students clustered within schools); and second, from the model itself, which has its own hierarchy, with the parameters of the within-school regressions at the bottom, controlled by the hyperparameters of the upper-level model.

Later we shall consider non-nested models—for example, individual observations that are nested within states and years. Neither "state" nor "year" is above the other in a hierarchical sense. In this sort of example, we can consider individuals, states, and years to be three different levels without the requirement of a full ordering or hierarchy. More complex structures, such as three-level nesting (for example, students within schools within school districts) are also easy to handle within the general multilevel framework.

Why we avoid the term "random effects." Multilevel models are often known as random-effects or mixed-effects models. The regression coefficients that are being modeled are called *random effects*, in the sense that they are considered random outcomes of a process identified with the model that is predicting them. In contrast, *fixed effects* correspond either to parameters that do not vary (for example, fitting the same regresslon line for each of the schools) or to parameters that vary but are not modeled themselves (for example, fitting a least squares regression model with various predictors, including indicators for the schools). A *mixed-effects* model includes both fixed and random effects; for example, in model (1.1), the varying intercepts α_j have a group-level model, but β is fixed and does not vary by group.

[1] The model can also be written as $y_{ij} = \alpha_j + \beta x_{ij} + \epsilon_{ij}$, where y_{ij} is the measurement from student i in school j. We prefer using the single sequence i to index all students (and $j[i]$ to label schools) because this fits in better with our multilevel modeling framework with data and models at the individual and group levels. The data are y_i because they can exist without reference to the groupings, and we prefer to include information about the groupings as numerical data—that is, the index variable $j[i]$—rather than through reordering the data through subscripting. We discuss the structure of the data and models further in Chapter 11.

Fixed effects can be viewed as special cases of random effects, in which the higher-level variance (in model (1.1), this would be σ_α^2) is set to 0 or ∞. Hence, in our framework, all regression parameters are "random," and the term "multilevel" is all-encompassing. As we discuss on page 245, we find the terms "fixed," "random," and "mixed" effects to be confusing and often misleading, and so we avoid their use.

1.2 Some examples from our own research

Multilevel modeling can be applied to just about any problem. Just to give a feel of the ways it can be used, we give here a few examples from our applied work.

Combining information for local decisions: home radon measurement and remediation

Radon is a carcinogen—a naturally occurring radioactive gas whose decay products are also radioactive—known to cause lung cancer in high concentrations and estimated to cause several thousand lung cancer deaths per year in the United States. The distribution of radon levels in U.S. homes varies greatly, with some houses having dangerously high concentrations. In order to identify the areas with high radon exposures, the Environmental Protection Agency coordinated radon measurements in a random sample of more than 80,000 houses throughout the country.

To simplify the problem somewhat, our goal in analyzing these data was to estimate the distribution of radon levels in each of the approximately 3000 counties in the United States, so that homeowners could make decisions about measuring or remediating the radon in their houses based on the best available knowledge of local conditions. For the purpose of this analysis, the data were structured hierarchically: houses within counties. If we were to analyze multiple measurements within houses, there would be a three-level hierarchy of measurements, houses, and counties.

In performing the analysis, we had an important predictor—the floor on which the measurement was taken, either basement or first floor; radon comes from underground and can enter more easily when a house is built into the ground. We also had an important county-level predictor—a measurement of soil uranium that was available at the county level. We fit a model of the form (1.1), where y_i is the logarithm of the radon measurement in house i, x is the floor of the measurement (that is, 0 for basement and 1 for first floor), and u is the uranium measurement at the county level. The errors ϵ_i in the first line of (1.1) represent "within-county variation," which in this case includes measurement error, natural variation in radon levels within a house over time, and variation between houses (beyond what is explained by the floor of measurement). The errors η_j in the second line represent variation between counties, beyond what is explained by the county-level uranium predictor.

The hierarchical model allows us to fit a regression model to the individual measurements while accounting for systematic unexplained variation among the 3000 counties. We return to this example in Chapter 12.

Modeling correlations: forecasting presidential elections

It is of practical interest to politicians and theoretical interest to political scientists that the outcomes of elections can be forecast with reasonable accuracy given information available months ahead of time. To understand this better, we set up a

model to forecast presidential elections. Our predicted outcomes were the Democratic Party's share of the two-party vote in each state in each of the 11 elections from 1948 through 1988, yielding 511 data points (the analysis excluded states that were won by third parties), and we had various predictors, including the performance of the Democrats in the previous election, measures of state-level and national economic trends, and national opinion polls up to two months before the election.

We set up our forecasting model two months before the 1992 presidential election and used it to make predictions for the 50 states. Predictions obtained using classical regression are reasonable, but when the model is evaluated historically (fitting to all but one election and then using the model to predict that election, then repeating this for the different past elections), the associated predictive intervals turn out to be too narrow: that is, the predictions are not as accurate as claimed by the model. Fewer than 50% of the predictions fall in the 50% predictive intervals, and fewer than 95% are inside the 95% intervals. The problem is that the 511 original data points are *structured*, and the state-level errors are *correlated*. It is overly optimistic to say that we have 511 independent data points.

Instead, we model

$$y_i = \beta_0 + X_{i1}\beta_1 + X_{i2}\beta_2 + \cdots + X_{ik}\beta_k + \eta_{t[i]} + \delta_{r[i],t[i]} + \epsilon_i, \text{ for } i = 1, \ldots, n, \quad (1.2)$$

where $t[i]$ is a indicator for time (election year), and $r[i]$ is an indicator for the region of the country (Northeast, Midwest, South, or West), and $n = 511$ is the number of state-years used to fit the model. For each election year, η_t is a nationwide error and the $\delta_{r,t}$'s are four independent regional errors.

The error terms must then be given distributions. As usual, the default is the normal distribution, which for this model we express as

$$\begin{aligned} \eta_t &\sim \text{N}(0, \sigma_\eta^2), \text{ for } t = 1, \ldots, 11 \\ \delta_{r,t} &\sim \text{N}(0, \sigma_\delta^2), \text{ for } r = 1, \ldots, 4;\ t = 1, \ldots, 11 \\ \epsilon_i &\sim \text{N}(0, \sigma_\epsilon^2), \text{ for } i = 1, \ldots, 511. \end{aligned} \quad (1.3)$$

In the multilevel model, all the parameters $\beta, \sigma_\eta, \sigma_\delta, \sigma_\epsilon$ are estimated from the data.

We can then make a prediction by simulating the election outcome in the 50 states in the next election year, $t = 12$:

$$y_i = \beta_0 + X_{i1}\beta_1 + X_{i2}\beta_2 + \cdots + X_{ik}\beta_k + \eta_{12} + \delta_{r[i],12} + \epsilon_i, \text{ for } i = n+1, \ldots, n+50.$$

To define the predictive distribution of these 50 outcomes, we need the point predictors $X_i\beta = \beta_0 + X_{i1}\beta_1 + X_{i2}\beta_2 + \cdots + X_{ik}\beta_k$ and the state-level errors ϵ as before, but we also need a new national error η_{12} and four new regional errors $\delta_{r,12}$, which we simulate from the distributions (1.3). The variation from these gives a more realistic statement of prediction uncertainties.

Small-area estimation: state-level opinions from national polls

In a micro-level version of election forecasting, it is possible to predict the political opinions of individual voters given demographic information and where they live. Here the data sources are opinion polls rather than elections.

For example, we analyzed the data from seven CBS News polls from the 10 days immediately preceding the 1988 U.S. presidential election. For each survey respondent i, we label $y_i = 1$ if he or she preferred George Bush (the Republican candidate), 0 if he or she preferred Michael Dukakis (the Democrat). We excluded respondents who preferred others or had no opinion, leaving a sample size n of

about 6000. We then fit the model,

$$\Pr(y_i = 1) = \text{logit}^{-1}(X_i\beta),$$

where X included 85 predictors:

- A constant term
- An indicator for "female"
- An indicator for "black"
- An indicator for "female and black"
- 4 indicators for age categories (18–29, 30–44, 45–64, and 65+)
- 4 indicators for education categories (less than high school, high school, some college, college graduate)
- 16 indicators for age × education
- 51 indicators for states (including the District of Columbia)
- 5 indicators for regions (Northeast, Midwest, South, West, and D.C.)
- The Republican share of the vote for president in the state in the previous election.

In classical regression, it would be unwise to fit this many predictors because the estimates will be unreliable, especially for small states. In addition, it would be necessary to leave predictors out of each batch of indicators (the 4 age categories, the 4 education categories, the 16 age × education interactions, the 51 states, and the 5 regions) to avoid collinearity.

With a multilevel model, the coefficients for each batch of indicators are fit to a probability distribution, and it is possible to include all the predictors in the model. We return to this example in Section 14.1.

Social science modeling: police stops by ethnic group with variation across precincts

There have been complaints in New York City and elsewhere that the police harass members of ethnic minority groups. In 1999 the New York State Attorney General's Office instigated a study of the New York City police department's "stop and frisk" policy: the lawful practice of "temporarily detaining, questioning, and, at times, searching civilians on the street." The police have a policy of keeping records on every stop and frisk, and this information was collated for all stops (about 175,000 in total) over a 15-month period in 1998–1999. We analyzed these data to see to what extent different ethnic groups were stopped by the police. We focused on blacks (African Americans), hispanics (Latinos), and whites (European Americans). We excluded others (about 4% of the stops) because of sensitivity to ambiguities in classifications. The ethnic categories were as recorded by the police making the stops.

It was found that blacks and hispanics represented 50% and 33% of the stops, respectively, despite constituting only 26% and 24%, respectively, of the population of the city. An arguably more relevant baseline comparison, however, is to the number of crimes committed by members of each ethnic group. Data on actual crimes are not available, of course, so as a proxy we used the number of arrests within New York City in 1997 as recorded by the Division of Criminal Justice Services (DCJS) of New York State. We used these numbers to represent the frequency of crimes that the police might suspect were committed by members of each group. When compared in that way, the ratio of stops to previous DCJS arrests was 1.24 for

whites, 1.53 for blacks, and 1.72 for hispanics—the minority groups still appeared to be stopped disproportionately often.

These ratios are suspect too, however, because they average over the whole city. Suppose the police make more stops in high-crime areas but treat the different ethnic groups equally within any locality. Then the citywide ratios could show strong differences between ethnic groups even if stops are entirely determined by location rather than ethnicity. In order to separate these two kinds of predictors, we performed a multilevel analysis using the city's 75 precincts. For each ethnic group $e = 1, 2, 3$ and precinct $p = 1, \ldots, 75$, we model the number of stops y_{ep} using an overdispersed Poisson regression. The exponentiated coefficients from this model represent relative rates of stops compared to arrests for the different ethnic groups, after controlling for precinct. We return to this example in Section 15.1.

1.3 Motivations for multilevel modeling

Multilevel models can be used for a variety of inferential goals including causal inference, prediction, and descriptive modeling.

Learning about treatment effects that vary

One of the basic goals of regression analysis is estimating treatment effects—how does y change when some x is varied, with all other inputs held constant? In many applications, it is not an overall effect of x that is of interest, but how this effect varies in the population. In classical statistics we can study this variation using *interactions*: for example, a particular educational innovation may be more effective for girls than for boys, or more effective for students who expressed more interest in school in a pre-test measurement.

Multilevel models also allow us to study effects that vary by group, for example an intervention that is more effective in some schools than others (perhaps because of unmeasured school-level factors such as teacher morale). In classical regression, estimates of varying effects can be noisy, especially when there are few observations per group; multilevel modeling allows us to estimate these interactions to the extent supported by the data.

Using all the data to perform inferences for groups with small sample size

A related problem arises when we are trying to estimate some group-level quantity, perhaps a local treatment effect or maybe simply a group-level average (as in the small-area estimation example on page 4). Classical estimation just using the local information can be essentially useless if the sample size is small in the group. At the other extreme, a classical regression ignoring group indicators can be misleading in ignoring group-level variation. Multilevel modeling allows the estimation of group averages and group-level effects, compromising between the overly noisy within-group estimate and the oversimplified regression estimate that ignores group indicators.

Prediction

Regression models are commonly used for predicting outcomes for new cases. But what if the data vary by group? Then we can make predictions for new units in existing groups or in new groups. The latter is difficult to do in classical regression:

if a model ignores group effects, it will tend to understate the error in predictions for new groups. But a classical regression that includes group effects does not have any automatic way of getting predictions for a new group.

A natural attack on the problem is a two-stage regression, first including group indicators and then fitting a regression of estimated group effects on group-level predictors. One can then forecast for a new group, with the group effect predicted from the group-level model, and then the observations predicted from the unit-level model. However, if sample sizes are small in some groups, it can be difficult or even impossible to fit such a two-stage model classically, and fully accounting for the uncertainty at both levels leads directly to a multilevel model.

Analysis of structured data

Some datasets are collected with an inherent multilevel structure, for example, students within schools, patients within hospitals, or data from cluster sampling. Statistical theory—whether sampling-theory or Bayesian—says that inference should include the factors used in the design of data collection. As we shall see, multilevel modeling is a direct way to include indicators for clusters at all levels of a design, without being overwhelmed with the problems of overfitting that arise from applying least squares or maximum likelihood to problems with large numbers of parameters.

More efficient inference for regression parameters

Data often arrive with multilevel structure (students within schools and grades, laboratory assays on plates, elections in districts within states, and so forth). Even simple cross-sectional data (for example, a random sample survey of 1000 Americans) can typically be placed within a larger multilevel context (for example, an annual series of such surveys). The traditional alternatives to multilevel modeling are *complete pooling*, in which differences between groups are ignored, and *no pooling*, in which data from different sources are analyzed separately. As we shall discuss in detail throughout the book, both these approaches have problems: no pooling ignores information and can give unacceptably variable inferences, and complete pooling suppresses variation that can be important or even the main goal of a study. The extreme alternatives can in fact be useful as preliminary estimates, but ultimately we prefer the *partial pooling* that comes out of a multilevel analysis.

Including predictors at two different levels

In the radon example described in Section 1.2, we have outcome measurements at the individual level and predictors at the individual and county levels. How can this information be put together? One possibility is simply to run a classical regression with predictors at both levels. But this does not correct for differences between counties *beyond* what is included in the predictors. Another approach would be to augment this model with indicators (dummy variables) for the counties. But in a classical regression it is not possible to include county-level indicators as well along with county-level predictors—the predictors would become collinear (see the end of Section 4.5 for a discussion of collinearity and nonidentifiability in this context).

Another approach is to fit the model with county indicators but without the county-level predictors, and then to fit a second model. This is possible but limited because it relies on the classical regression estimates of the coefficients for those

county-level indicators—and if the data are sparse within counties, these estimates won't be very good. Another possibility in the classical framework would be to fit separate models in each group, but this is not possible unless the sample size is large in each group. The multilevel model provides a coherent model that simultaneously incorporates both individual- and group-level models.

Getting the right standard error: accurately accounting for uncertainty in prediction and estimation

Another motivation for multilevel modeling is for predictions, for example, when forecasting state-by-state outcomes of U.S. presidential elections, as described in Section 1.2. To get an accurate measure of predictive uncertainty, one must account for correlation of the outcome between states in a given election year. Multilevel modeling is a convenient way to do this.

For certain kinds of predictions, multilevel models are essential. For example, consider a model of test scores for students within schools. In classical regression, school-level variability might be modeled by including an indicator variable for each school. In this framework though, it is impossible to make a prediction for a new student in a new school, because there would not be an indicator for this new school in the model. This prediction problem is handled seamlessly using multilevel models.

1.4 Distinctive features of this book

The topics and methods covered in this book overlap with many other textbooks on regression, multilevel modeling, and applied statistics. We differ from most other books in these areas in the following ways:

- We present methods and software that allow the reader to fit complicated, linear or nonlinear, nested or non-nested models. We emphasize the use of the statistical software packages R and Bugs and provide code for many examples as well as methods such as redundant parameterization that speed computation and lead to new modeling ideas.
- We include a wide range of examples, almost all from our own applied research. The statistical methods are thus motivated in the best way, as successful practical tools.
- Most books define regression in terms of matrix operations. We avoid much of this matrix algebra for the simple reason that it is now done automatically by computers. We are more interested in understanding the "forward," or predictive, matrix multiplication $X\beta$ than the more complicated inferential formula $(X^tX)^{-1}X^ty$. The latter computation and its generalizations are important but can be done out of sight of the user. For details of the underlying matrix algebra, we refer readers to the regression textbooks listed in Section 3.8.
- We try as much as possible to display regression results graphically rather than through tables. Here we apply ideas such as those presented in the books by Ramsey and Schafer (2001) for classical regression and Kreft and De Leeuw (1998) for multilevel models. We consider graphical display of model estimates to be not just a useful teaching method but also a necessary tool in applied research.

  Statistical texts commonly recommend graphical displays for model diagnostics. These can be very useful, and we refer readers to texts such as Cook and Weisberg

(1999) for more on this topic—but here we are emphasizing graphical displays of the fitted models themselves. It is our experience that, even when a model fits data well, we have difficulty understanding it if all we do is look at tables of regression coefficients.

- We consider multilevel modeling as generally applicable to structured data, not limited to clustered data, panel data, or nested designs. For example, in a random-digit-dialed survey of the United States, one can, and should, use multilevel models if one is interested in estimating differences among states or demographic subgroups—even if no multilevel structure is in the survey design.

Ultimately, you have to learn these methods by doing it yourself, and this chapter is intended to make things easier by recounting stories about how we learned this by doing it ourselves. But we warn you ahead of time that we include more of our successes than our failures.

Costs and benefits of our approach

Doing statistics as described in this book is not easy. The difficulties are not mathematical but rather conceptual and computational. For classical regressions and generalized linear models, the actual fitting is easy (as illustrated in Part 1), but programming effort is still required to graph the results relevantly and to simulate predictions and replicated data. When we move to multilevel modeling, the fitting itself gets much more complicated (see Part 2B), and displaying and checking the models require correspondingly more work. Our emphasis on R and Bugs means that an initial effort is required simply to learn and use the software. Also, compared to usual treatments of multilevel models, we describe a wider variety of modeling options for the researcher so that more decisions will need to be made.

A simpler alternative is to use classical regression and generalized linear modeling where possible—this can be done in R or, essentially equivalently, in Stata, SAS, SPSS, and various other software—and then, when multilevel modeling is really needed, to use functions that adapt classical regression to handle simple multilevel models. Such functions, which can be run with only a little more effort than simple regression fitting, exist in many standard statistical packages.

Compared to these easier-to-use programs, our approach has several advantages:

- We can fit a greater variety of models. The modular structure of Bugs allows us to add complexity where needed to fit data and study patterns of interest.
- By working with simulations (rather than simply point estimates of parameters), we can directly capture inferential uncertainty and propagate it into predictions (as discussed in Chapter 7 and applied throughout the book). We can directly obtain inference for quantities other than regression coefficients and variance parameters.
- R gives us flexibility to display inferences and data flexibly.

We recognize, however, that other software and approaches may be useful too, either as starting points or to check results. Section C.4 describes briefly how to fit multilevel models in several other popular statistical software packages.

1.5 Computing

We perform computer analyses using the freely available software R and Bugs. Appendix C gives instructions on obtaining and using these programs. Here we outline how these programs fit into our overall strategy for data analysis.

Our general approach to statistical computing

In any statistical analysis, we like to be able to directly manipulate the data, model, and inferences. We just about never know the right thing to do ahead of time, so we have to spend much of our effort examining and cleaning the data, fitting many different models, summarizing the inferences from the models in different ways, and then going back and figuring how to expand the model to allow new data to be included in the analysis.

It is important, then, to be able to select subsets of the data, to graph whatever aspect of the data might be of interest, and to be able to compute numerical summaries and fit simple models easily. All this can be done within R—you will have to put some initial effort into learning the language, but it will pay off later.

You will almost always need to try many different models for any problem: not just different subsets of predictor variables as in linear regression, and not just minor changes such as fitting a logit or probit model, but entirely different formulations of the model—different ways of relating observed inputs to outcomes. This is especially true when using new and unfamiliar tools such as multilevel models. In Bugs, we can easily alter the internal structure of the models we are fitting, in a way that cannot easily be done with other statistical software.

Finally, our analyses are almost never simply summarized by a set of parameter estimates and standard errors. As we illustrate throughout, we need to look carefully at our inferences to see if they make sense and to understand the operation of the model, and we usually need to postprocess the parameter estimates to get predictions or generalizations to new settings. These inference manipulations are similar to data manipulations, and we do them in R to have maximum flexibility.

Model fitting in Part 1

Part 1 of this book uses the R software for three general tasks: (1) fitting classical linear and generalized linear models, (2) graphing data and estimated models, and (3) using simulation to propagate uncertainty in inferences and predictions (see Sections 7.1–7.2 for more on this).

Model fitting in Parts 2 and 3

When we move to multilevel modeling, we begin by fitting directly in R; however, for more complicated models we move to Bugs, which has a general language for writing statistical models. We call Bugs from R and continue to use R for preprocessing of data, graphical display of data and inferences, and simulation-based prediction and model checking.

R and S

Our favorite all-around statistics software is R, which is a free open-source version of S, a program developed in the 1970s and 1980s at Bell Laboratories. S is also available commercially as S-Plus. We shall refer to R throughout, but other versions of S generally do the same things.

R is excellent for graphics, classical statistical modeling (most relevant here are the `lm()` and `glm()` functions for linear and generalized linear models), and various nonparametric methods. As we discuss in Part 2, the `lmer()` function provides quick fits in R for many multilevel models. Other packages such as `MCMCpack` exist to fit specific classes of models in R, and other such programs are in development.