# *Multilevel models*

Mark Andrews
Psychology Department, Nottingham Trent University

mark.andrews@ntu.ac.uk

# *Introduction*

- Multilevel models are a broad class of models that are applied to data that consist of sub-groups or clusters, including when these clusters are hierarchically arranged.
- A number of related terms are used to describe multilevel models: *hierarchical* models, *mixed effects* models, *random effects* models, and more.
- The defining feature of multilevel models is that they are *models of models*.
- In other words, for each cluster or sub-group in our data we create a statistical model, and then model how these statistical models vary across the clusters or sub-groups.

# *Plan*

- We will begin our coverage of multilevel models by exploring *random effects* models. These are some of the simplest types of multilevel models, but yet they can make clear the key defining characteristics of the multilevel models generally.
- We then proceed to cover multilevel linear models, which are often referred to a *linear mixed effects* models.
- After that, we will cover multilevel generalized linear models, which are the generalized linear model counterpart of linear mixed effects models.
- Finally, we will cover how to perform Bayesian multilevel models.

# *Random effects models*

- Let us consider the following data set, which is on rat tumours.

```
rats_df <- read_csv(here('data/rats.csv'),
                    col_types = cols(batch = col_character())
)
```

- Let us begin by focusing in on a single batch.

```
rats_df_42 <- filter(rats_df, batch == '42')
```

- In this batch, out of 13 rats, the recorded number of tumours was 2.
- With these numbers alone, we can provide a simple statistical model of the tumour rate in batch 42.

# *Random effects models*

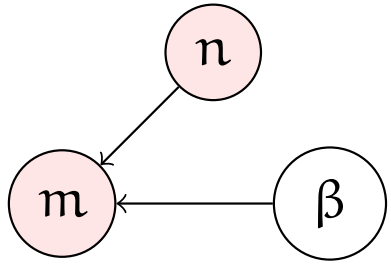
- In this model, we can say that there is a fixed but unknown probability of a tumour in this batch, which we will denote by $\theta$.
- In other words, our model is a binomial model:

$$m \sim \text{Binom}(\theta, n).$$

- This is identical to the following binomial logistic regression model.

$$m \sim \text{Binom}(\theta, n), \quad \log\left(\frac{\theta}{1-\theta}\right) = \beta.$$

This binomial model can be represented by the following diagram.



- We can implement this binomial logistic model in R using `glm`.

```
M <- glm(cbind(m, n-m) ~ 1,
         data = rats_df_42,
         family = binomial(link = 'logit')
)
```

- From this model `M`, we can see that our estimate of $\theta$ is as follows:

```
ilogit <- function(x) 1/(1 + exp(-x))
coef(M) %>% ilogit() %>% unname()
#> [1] 0.1538462
```

- This is expected given that out of 13 rats in this batch, the recorded number of tumours was 2, which is a proportion of 0.154.

- We can now easily extend this model to apply to all batches in our data set.

$$m_j \sim \text{Binom}(\theta_j, n_j), \quad \log\left(\frac{\theta_j}{1-\theta_j}\right) = \beta_j.$$

- This is implemented using `glm` as follows.

```
M <- glm(cbind(m, n-m) ~ 0 + batch,
         data = rats_df,
         family = binomial(link = 'logit')
)
```

- Although we have implemented a single `glm` model, this has effectively lead to J separate binomial models.

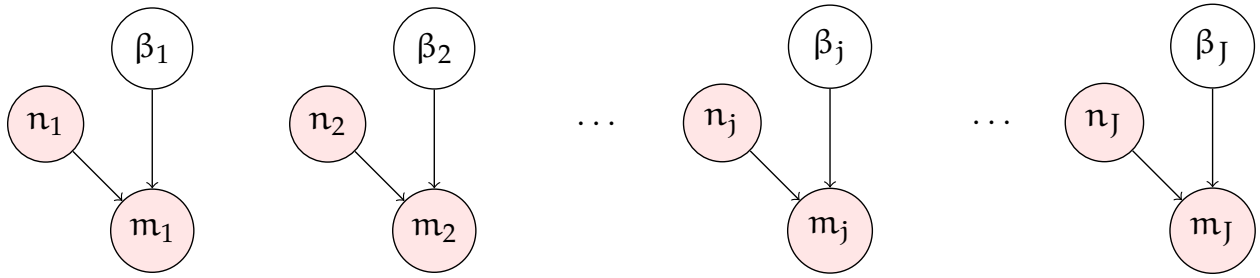



Figure 1: Inferring the log odds of a tumour $\beta_j$ in each of the J batches is identical to J independent binomial models.

- From this,we have a model of the tumour rate for batch 1, another for batch 2, and so on. We do not have a model of the distribution of tumour rates across all batches.
- We do not, for example, have a model that gives us the mean or standard deviation, or any other information, about the tumour rate across all possible batches in this experiment, of which our set of 71 batches are a sample.
- In order to obtain this model, we must perform a *multilevel model*.

- A multilevel model extension of the binomial logistic regression model above is as follows.

$$\text{for } j \in 1 \dots J, \quad m_j \sim \text{Binom}(\theta_j, n_j),$$
$$\log\left(\frac{\theta_j}{1-\theta_j}\right) = \beta_j,$$
$$\beta_j \sim N(b, \tau^2).$$

- The crucial added feature here is that the log odds of the tumour probabilities is being modelled as normally distributed with a mean of $b$ and a standard deviation of $\tau$.

- The random effects dependencies are shown in the following Bayesian network diagram.

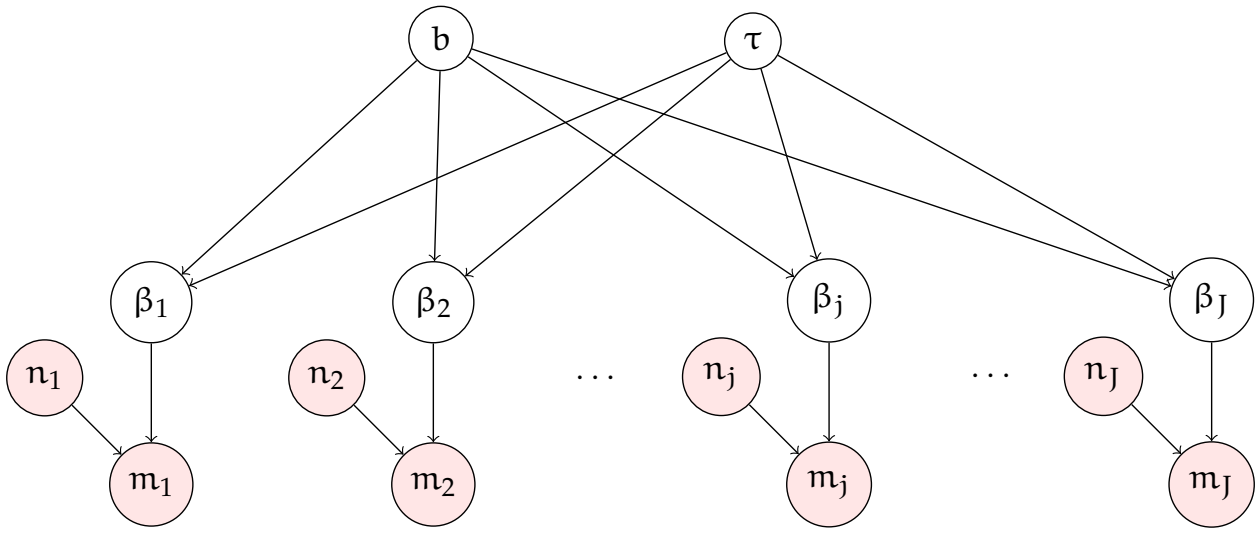



Figure 2: Inferring the log odds of a tumour $\beta_j$ in each of the J batches is identical to J independent binomial models.

- In this multilevel model, just as in the previous non-multilevel model, $\beta_1, \beta_2, \ldots \beta_j \ldots \beta_J$ have fixed but unknown values.
- However, in addition, these values are modelled as all drawn from the same normal distribution.
- The two important consequences of this are as follows.
- First, it provides a model of the *population* from which $\beta_1, \beta_2, \ldots \beta_j \ldots \beta_J$ are a sample.
- Given that each $\beta_j$ effectively defines a model for a batch of rats, then the normal distribution from which $\beta_1, \beta_2 \ldots \beta_j \ldots \beta_J$ are drawn is a *model of models*.
- Amongst other things, this population model of the $\beta$'s allows to predict the log odds, or probability, of a tumour for any future batch of rats, i.e. batch $J + 1$.

- Second, because we are assuming that $\beta_1, \beta_2, \ldots \beta_j \ldots \beta_J$ are all drawn from the same normal distribution, this introduces constraints on the inference of the values of each $\beta_j$.
- In other words, to infer the value of $\beta_j$, the observed values of $m_j$ and $n_j$ are not the only relevant pieces of information.
- Now, the values of $b$ and $\tau$ are also relevant, and because $b$ and $\tau$ are also unknown, they themselves must be inferred from $\beta_1, \beta_2, \ldots \beta_j \ldots \beta_J$.
- This effectively means that the inferences concerning $\beta_1, \beta_2, \ldots \beta_j \ldots \beta_J$ are inter-dependent and mutually constrain one another.

- Given that we can rewrite $\beta_j \sim N(b, \tau^2)$ as $\beta_j = b + \xi_j$ where $\xi_j \sim N(0, \tau^2)$, we can rewrite the multilevel model as

$$\begin{aligned} \text{for } j \in 1 \dots J, \quad m_j &\sim \text{Binom}(\theta_j, n_j), \\ \log\left(\frac{\theta_j}{1-\theta_j}\right) &= b + \xi_j, \\ \xi_j &\sim N(0, \tau^2). \end{aligned}$$

- We can then implement this model using the `glmer` model that is part of the `lme4` package.

```
library(lme4)
M_ml <- glmer(cbind(m, n-m) ~ 1 + (1|batch),
              data = rats_df,
              family = binomial(link = 'logit')
)
```

Let us look at the summary of this model.

```
summary(M_ml)
#> Generalized linear mixed model fit by maximum likelihood (Laplace
#>   Approximation) [glmerMod]
#>  Family: binomial  ( logit )
#> Formula: cbind(m, n - m) ~ 1 + (1 | batch)
#>    Data: rats_df
#>
#>      AIC      BIC   logLik deviance df.resid
#>    319.9    324.4   -157.9    315.9       69
#>
#> Scaled residuals:
#>     Min      1Q  Median      3Q     Max
#> -1.2392 -0.6230 -0.1055  0.4795  1.0253
#>
#> Random effects:
#>  Groups Name        Variance Std.Dev.
#>  batch  (Intercept) 0.4417   0.6646
#> Number of obs: 71, groups:  batch, 71
#>
#> Fixed effects:
#>             Estimate Std. Error z value Pr(>|z|)
#> (Intercept)  -1.9369     0.1211     -16   <2e-16 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- From the summary, our model of the distribution of the log odds of the tumours is a normal distribution whose mean and standard deviation are estimated to be -1.937 and 0.665, respectively.

From this model, we can also obtain the estimates of $\xi_1, \xi_2 \ldots \xi_j \ldots \xi_J$ from the model by using the `ranef` command.

```
ranef(M_ml)$batch %>%
  head()
#>    (Intercept)
#> 1   -0.6298720
#> 10  -0.6096908
#> 11  -0.6096908
#> 12  -0.5888596
#> 13  -0.5888596
#> 14  -0.5673326
```

- We may obtain the estimates of $b$ using the `fixef` command.

```
b <- fixef(M_ml)
```

- We may then add on the estimates of $b$ to get the estimates of $\beta_1, \beta_2 \ldots \beta_j \ldots \beta_J$.

```
b + ranef(M_ml)$batch %>%
  head()
#>    (Intercept)
#> 1    -2.566785
#> 10   -2.546604
#> 11   -2.546604
#> 12   -2.525773
#> 13   -2.525773
#> 14   -2.504246
```

- We may obtain the estimates of $\beta_1, \beta_2 \ldots \beta_j \ldots \beta_J$ more directly by using the `coef` command.

```
M_ml_estimates <- coef(M_ml)$batch

M_ml_estimates %>%
  head()
#>    (Intercept)
#> 1    -2.566785
#> 10   -2.546604
#> 11   -2.546604
#> 12   -2.525773
#> 13   -2.525773
#> 14   -2.504246
```

- Comparing these values to the corresponding values in the non-multilevel model, we can see how the estimates of $\beta_1, \beta_2 \ldots \beta_j \ldots \beta_J$ mutually constrain one another.
- This phenomenon is an example of *shrinkage*. In this model, it is easier to visualize this effect if we look at $\theta_1, \theta_2 \ldots \theta_j \ldots \theta_J$, which are simply the inverse logit transforms of $\beta_1, \beta_2 \ldots \beta_j \ldots \beta_J$.

In Figure 3, we compare the estimates of $\theta_1, \theta_2 \ldots \theta_j \ldots \theta_J$ from the flat or non-multilevel model `M` against those of the multilevel model `M_ml`.

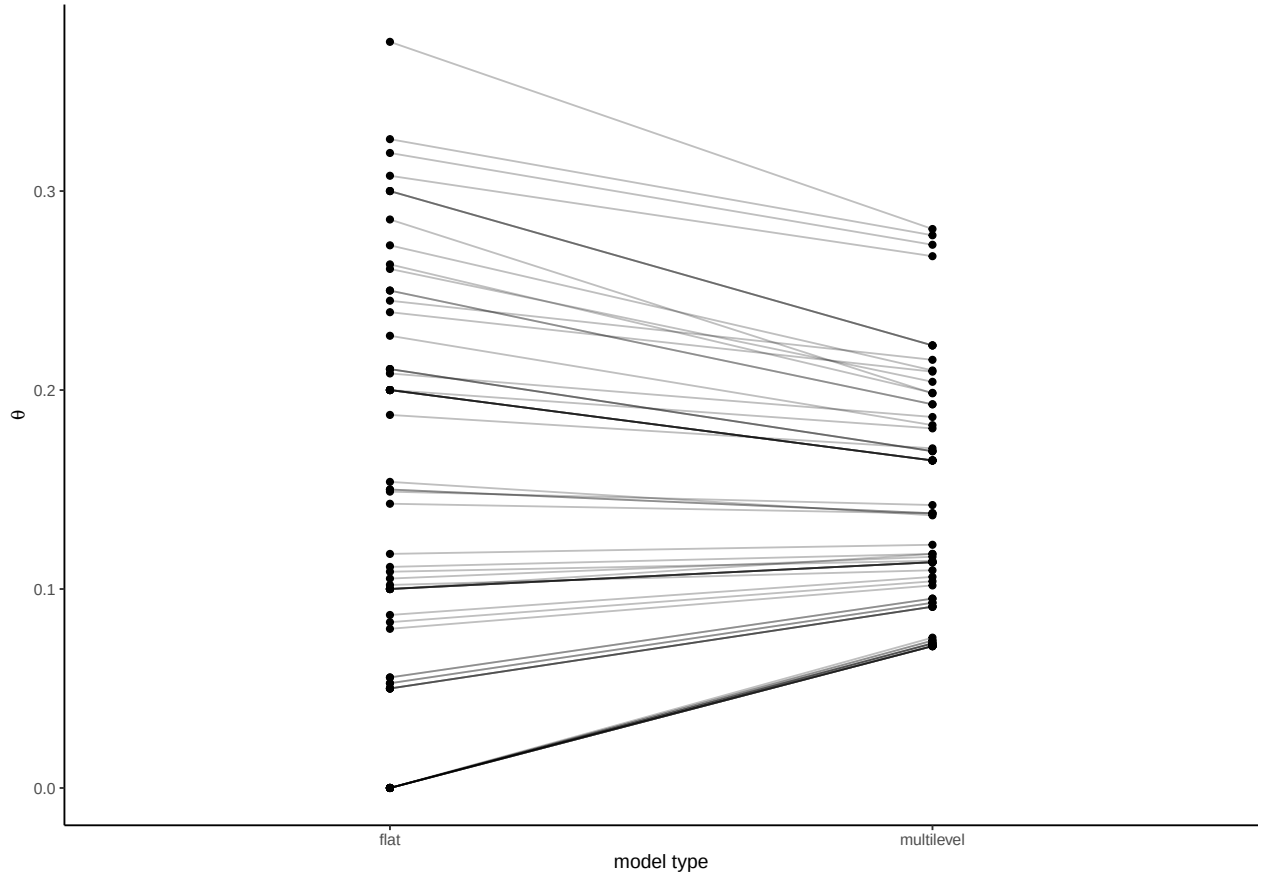



Figure 3: Estimates for $\theta_1, \theta_2 \ldots \theta_j \ldots \theta_J$ from the flat or non-multilevel model (left) and the multilevel model (right).

# *Normal random effects models*

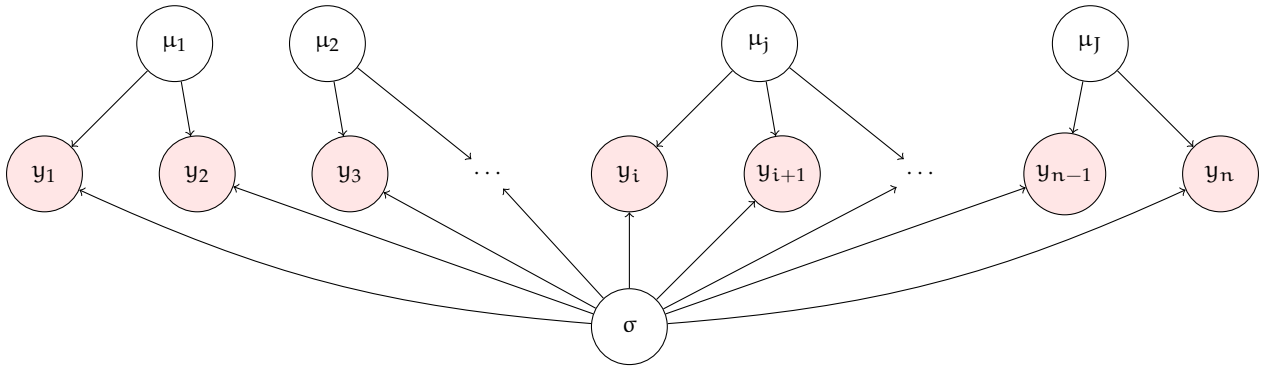
- Let us now consider a new data set.

```
alcohol_df <- read_csv(here('data/alcohol.csv'))
```

- In this, we have the per capita average alcohol consumption in $J = 189$ countries in $K = 22$ different years, though we do not necessarily have data from each country in each year. Let us denote the per capita alcohol values by $y_1, y_2 \dots y_i \dots y_n$. For each $y_i$, we have an indicator variable $x_i \in 1 \dots J$, which indicates the country that $y_i$ corresponds to.
- An initial model for $y_1, y_2 \dots y_i \dots y_n$ could then be

$$y_i \sim N(\mu_{[x_i]}, \sigma^2), \quad \text{for } i \in 1 \dots n,$$

where $\mu_1, \mu_2 \dots \mu_j \dots \mu_J$ are the country alcohol per capita consumption averages for the $J$ countries.




This is a non-multilevel model because the alcohol consumption averages in each country are being modelled independently of those of other countries.

A multilevel counterpart to the above model would be as follows.

$$y_i \sim N(\mu_{[x_i]}, \sigma^2), \quad \text{for } i \in 1 \dots n,$$
$$\mu_j \sim N(\phi, \tau^2), \quad \text{for } j \in 1 \dots J.$$

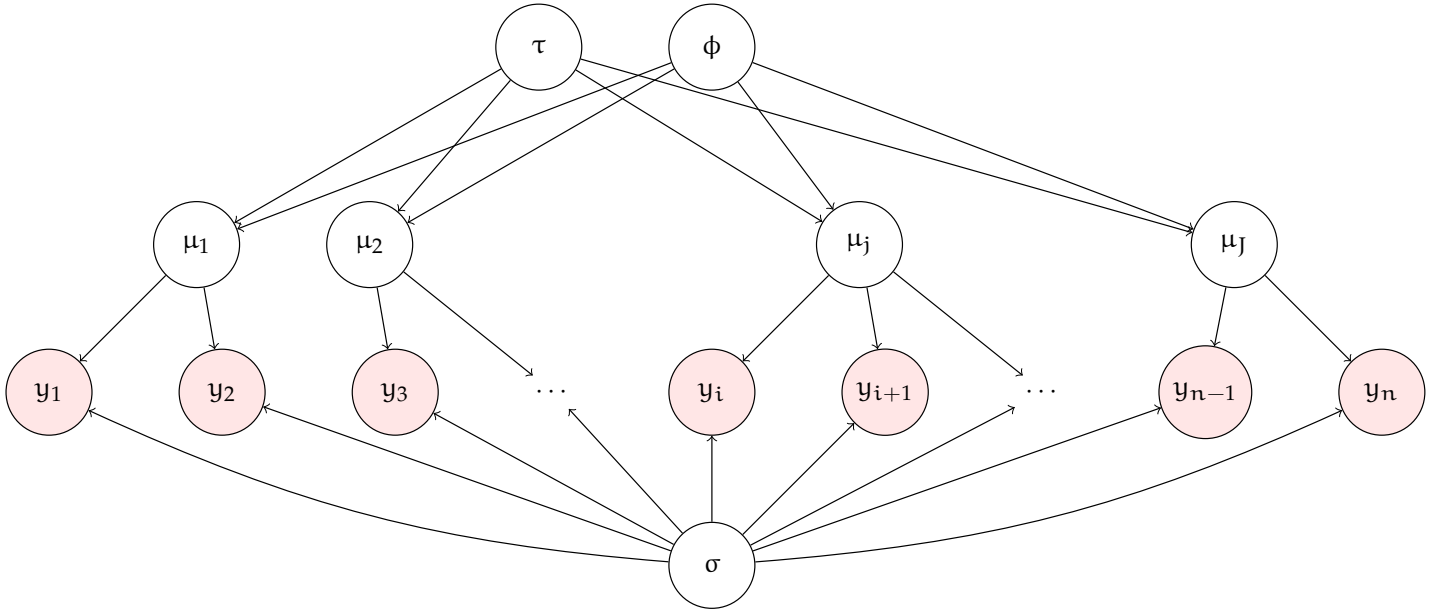
* This model extends the previous one by assuming that the $\mu_1, \mu_2 \dots \mu_j \dots \mu_J$ are drawn from a normal distribution with mean $\phi$ and standard deviation of $\tau$.

- Given that $y_i$ can be rewritten as $y_i = \mu_{[x_i]} + \epsilon_i$, where $\epsilon_i \sim N(0, \sigma^2)$, and that $\mu_j$ can be rewritten as $\mu_j = \phi + \xi_j$ where $\xi_j \sim N(0, \tau^2)$, we can rewrite the above model as

$$y_i = \phi + \xi_{[x_i]} + \epsilon_i, \quad \text{for } i \in 1 \dots n,$$

  where each $\xi_j \sim N(0, \tau^2)$ and each $\epsilon_i \sim N(0, \sigma^2)$.
- Here, $\phi$ signifies the global average per capita alcohol consumption rate.
- Each $\xi_j$ is the *random offset* of country $j$ from $\phi$, and each $\epsilon_i$ is the residual error for each observation.
- In this model, the residual error $\epsilon_i$ gives the random year by year deviation from the country $x_i$'s average consumption rate.

- The Bayesian model of this random effects normal linear model is as follows:



We can implement this model using `lme4::lmer` as follows.

```
M_ml <- lmer(alcohol ~ 1 + (1|country),
             data = alcohol_df)
```

- The `(Intercept)` estimate in the `Fixed effects` and the `Std.Dev.` for `country` in the `Random effects`, the normal distribution of the $\mu$ values has a mean of $\phi = 6.661$ and standard deviation of $\tau = 4.713$.
- The residual standard deviation $\sigma$ is given by the `Std.Dev.` for `Residual` in the `Random effects`, and has the value of $\sigma = 1.053$.

# *Intraclass correlation*

- Given the nature of the random effects model, i.e. each $y_i$ is modelled as $y_i = \phi + \xi_{[x_i]} + \epsilon_i$, the variance of $y$ is equal to $\tau^2 + \sigma^2$.
- The value

$$\frac{\tau^2}{\tau^2 + \sigma^2}$$

  is known as the *intraclass correlation* (ICC), which takes on values between 0 and 1.
- Obviously, ICC tells us how much of the total variance in the data is due to variation between the countries.
- If the ICC is relatively high, and so $\tau^2/\sigma^2$ is relatively high, the observed values *within* countries will be close together relative to the *between* country averages, and thus there will be relatively high clustering of the data.
- In this data, the ICC is 0.95.

# *Linear mixed effects models*

- We will now consider multilevel linear regression models.
- These are often referred to as linear mixed effects models, for reasons that will be clear after we describe them in more detail.
- As with random effects models, these models are best introduced by way of example.
- For this, we will use he `sleepstudy` data set from `lme4`, which provides the average reaction time for each person on each day of a sleep deprivation experiment that lasted 10 days.

```
sleepstudy <- lme4::sleepstudy %>%
  as_tibble()
```

# *Sleep deprivation study*

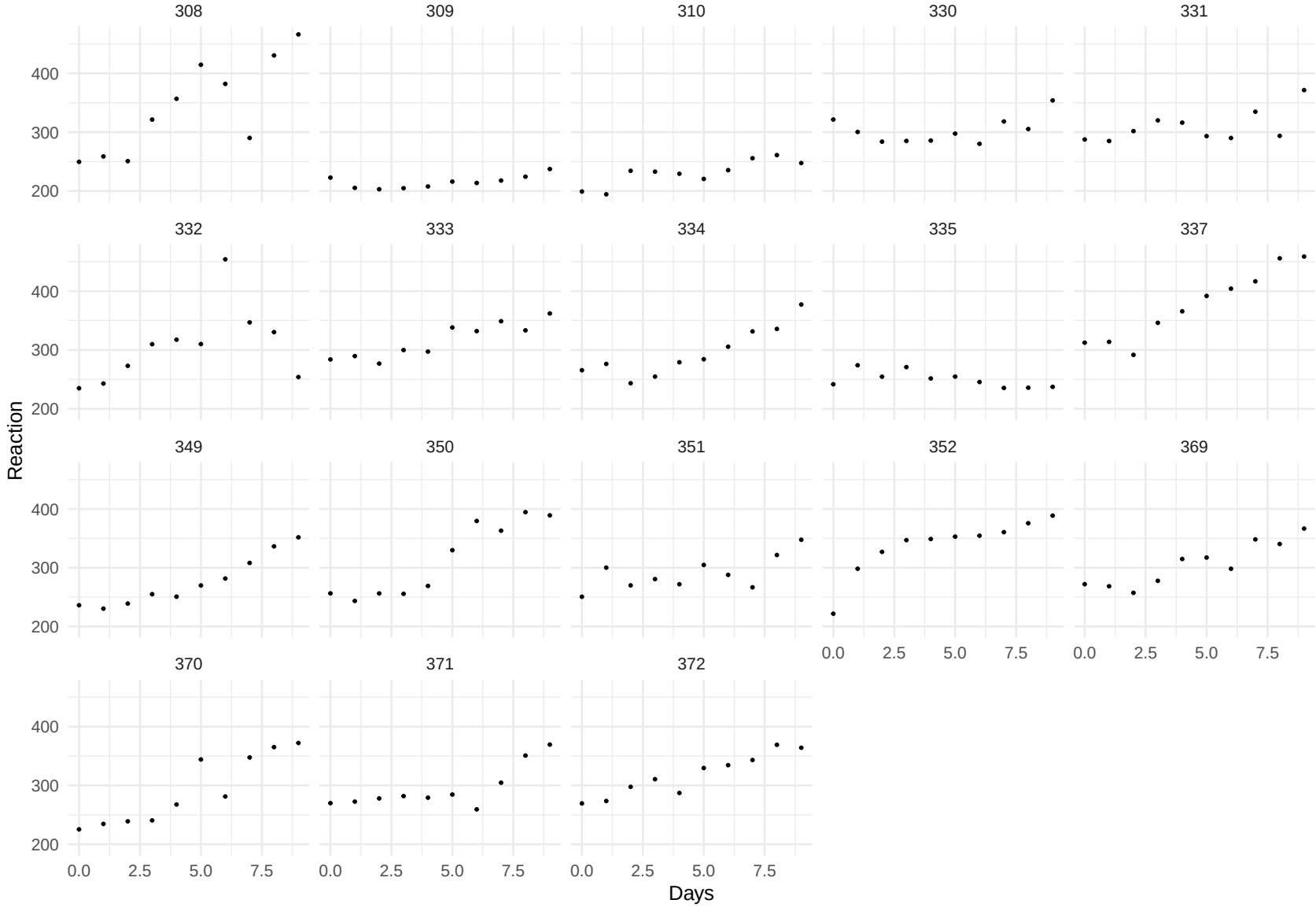



Figure 4: Each figure shows the average reaction time data from a subject in sleep deprivation on each day of the 10 day experiment.

- To begin our analysis, let us first focus on one arbitrarily chosen experimental subject, namely subject `350`.
- The trend over time in this subject's average reaction time can be modelled using the following normal linear model:

$$y_d \sim N(\mu_d, \sigma^2), \quad \mu_d = \beta_0 + \beta_1 x_d, \quad \text{for } d \in 1 \dots n,$$

  where $y_d$ represents the subject's reaction time on their $d$th observation, and $x_d \in \{0, 2, \dots n = 9\}$ indicates the day when this observation happened.

Using $\vec{\beta} = [\beta_0, \beta_1]^\intercal$, we can represent this model using a Bayesian network diagram as we do in Figure 5. In that figure, we provide two equivalent diagrams, with Figure 5b using a plate notation that denotes a repetition of nodes within a bounding plate according to an index, which in this case is $d \in 1 \dots n$.

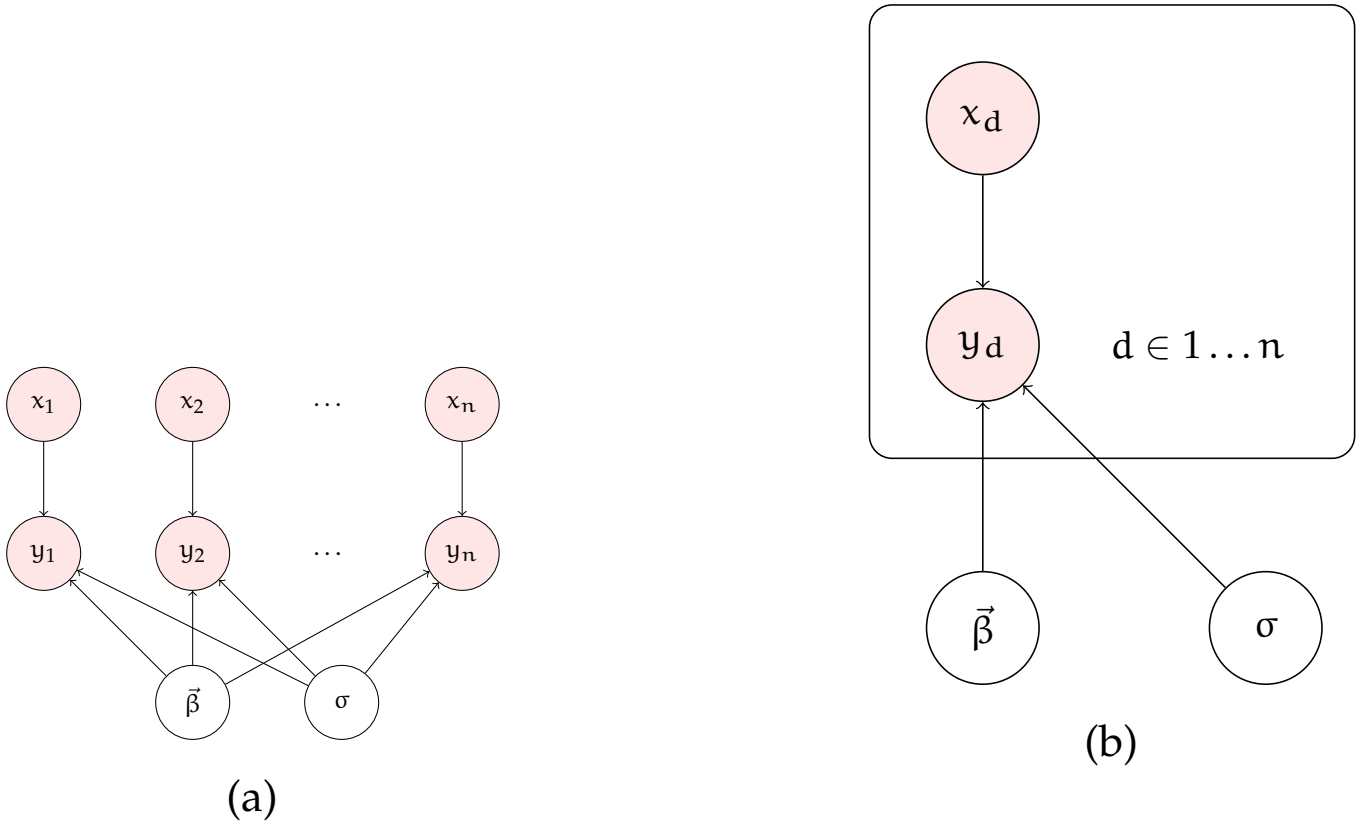



Figure 5: Two equivalent Bayesian network diagrams representing a normal linear model with one predictor variable. Diagram b) uses a compact plate notation whereby all variables within the plate are repeated for all values of the index $i$, which takes values from 1 to $n$.

- This model is implemented in R as follows.

```
M_350 <- lm(Reaction ~ Days, data = sleepstudy_350)
```

- The estimated values of the coefficients are as follows.

```
#> (Intercept)        Days
#>   225.83460    19.50402
```

- Thus, we estimate that the average reaction time of subject 350 increases by 19.5 on each day of study.
- In addition, because the first day of the study was indicated by $x_i = 0$, this subject's average reaction prior to any sleep deprivation was 225.83.

- Were we to provide a similar model for each subject in the experiment, whom we will index by $j \in 1 \dots J$, this would lead to J independent normal linear models.
- If we denote the average reaction time on observation $d$ for subject $j$ by $y_{jd}$, this set of models is as follows.

$$
\begin{aligned}
y_{jd} &\sim N(\mu_{jd}, \sigma_j^2), \\
\mu_{jd} &= \beta_{j0} + \beta_{j1} x_{jd}, \quad \text{for } j \in 1 \dots J, \text{ for } d \in 1 \dots n_j.
\end{aligned}
$$

- This model is represented in a Bayesian network diagram in Figure 6a.

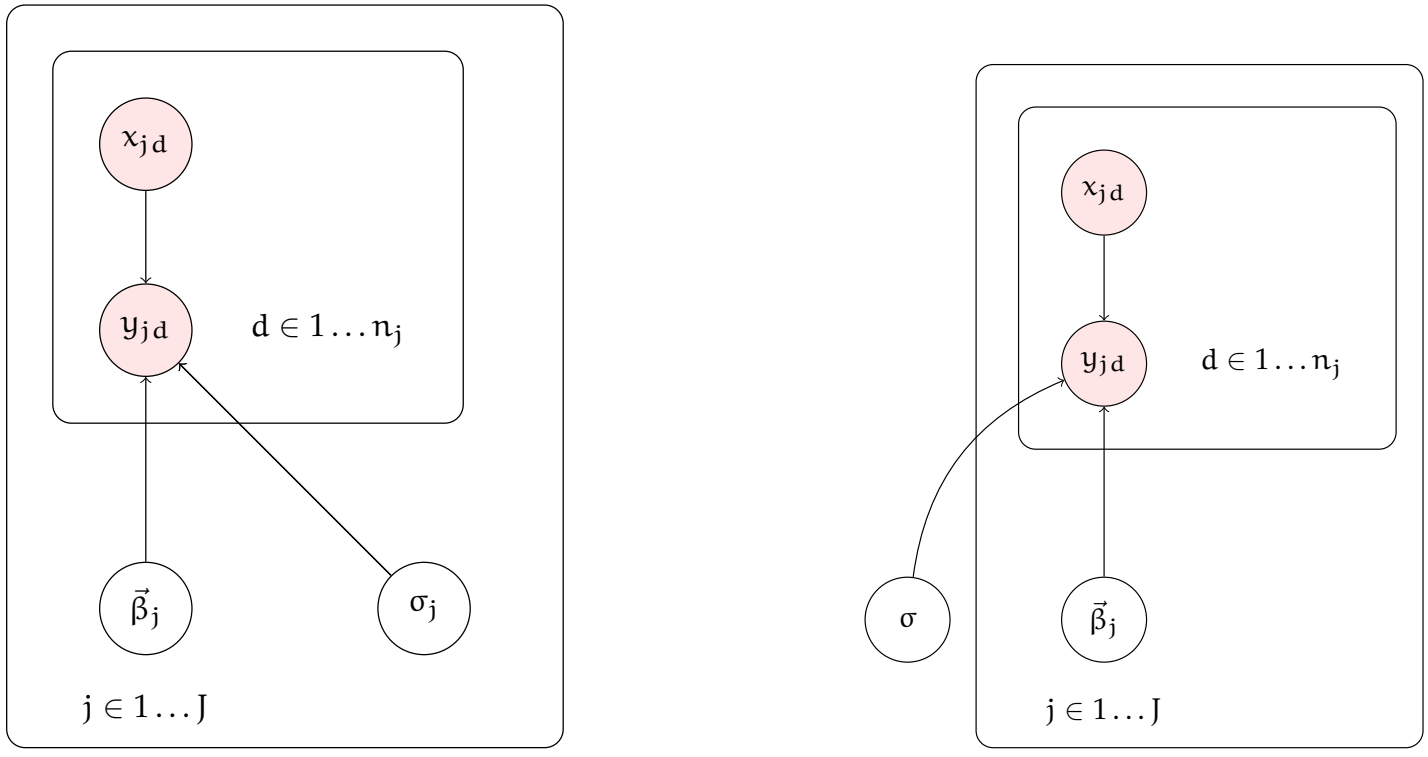



Figure 6: Bayesian network diagrams for a) a set of J independent normal linear models, and b) a varying slope and varying intercept linear model whereby the slope and intercept vary by a categorical variable with J levels.

- If we assume that there is a common residual standard deviation term $\sigma$, rather than one per each of the $J$ subjects, this model is identical to a varying intercept and varying slope linear model.
- Using R, we can implement this model as follows.

```
M_flat <- lm(Reaction ~ 0 + Subject + Subject:Days, data = sleep
```

- Formally, this model is equivalent to

$$
\begin{aligned}
y_{jd} &\sim N(\mu_{jd}, \sigma^2), \\
\mu_{jd} &= \beta_{j0} + \beta_{j1} x_{jd}, \quad \text{for } j \in 1 \dots J, \text{ for } d \in 1 \dots n_j,
\end{aligned}
$$

and we have provided a Bayesian network diagram of it in Figure 6b.

- Let us now consider a multilevel variant of this non-multilevel varying intercept and slope model.
- In this, we assume that the vector of coefficients $\vec{\beta}_j = [\beta_{j0}, \beta_{j1}]^\intercal$ is drawn from a multivariate Normal distribution with mean vector $\vec{b}$ and covariance matrix $\Sigma$.
- This model can be written as follows.

$$
\begin{aligned}
y_{jd} &\sim N(\mu_{jd}, \sigma), \\
\mu_{jd} &= \beta_{j0} + \beta_{j1} x_{jd}, \quad \text{for } j \in 1 \dots J, \text{ for } d \in 1 \dots n_j,, \\
\vec{\beta}_j &\sim N(\vec{b}, \Sigma) \quad \text{for } j \in 1 \dots J,
\end{aligned}
$$

- The Bayesian network diagram for this model is shown in Figure 7.
- As we can see, this is an extension of the Bayesian network diagram in Figure 6b, with the extension being that each $\vec{\beta}_j$ are modelled as functions of $\vec{b}$ and $\Sigma$.

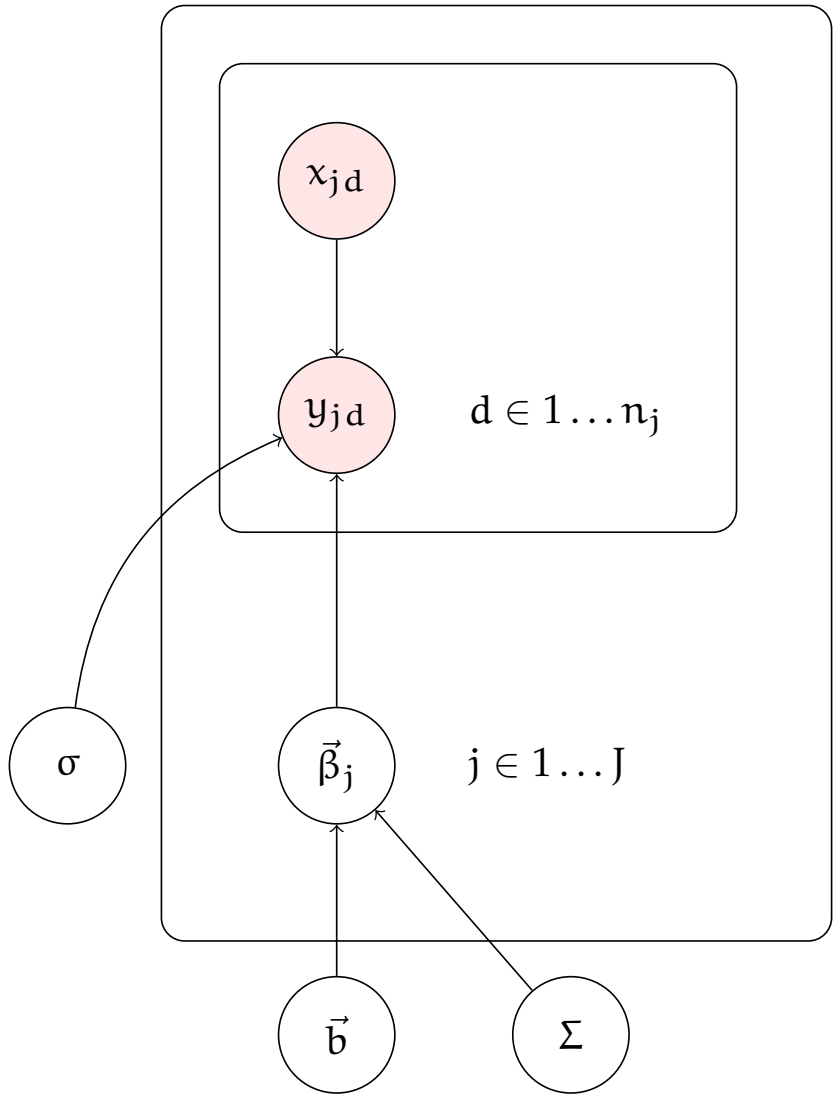



Figure 7: Bayesian network diagrams for a multilevel varying slopes and intercepts linear model.

- We can rewrite this multilevel model in the following manner.

$$\begin{aligned} \text{for } i \in 1 \dots n, \quad & y_i \sim N(\mu_i, \sigma^2), \\ & \mu_i = \beta_{[s_i]0} + \beta_{[s_i]1} x_i, \\ \text{for } j \in 1 \dots J, \quad & \vec{\beta}_j \sim N(\vec{b}, \Sigma). \end{aligned}$$

- Note that here the $i$ index ranges over all values in the entire data-set, i.e. $i \in 1, 2 \dots n$, and each $s_i \in 1, 2 \dots J$ is an indicator variable that indicates the identity of the subject on observation $i$.
- This notation with a single subscript per observation and indicator variables is more extensible, especially for complex models.
- Using this new notation, given that $\vec{\beta}_j \sim N(\vec{b}, \Sigma)$, we can rewrite $\vec{\beta}_j$ as $\vec{\beta}_j = \vec{b} + \vec{\zeta}_j$ where $\vec{\zeta}_j \sim N(0, \Sigma)$.

- Substituting $\vec{b} + \zeta_j$ for $\vec{\beta}$, and thus substituting $b_0 + \zeta_{j0}$ and $b_1 + \zeta_{j1}$ for $\beta_{j0}$ and $\beta_{j1}$, respectively, we have the following model.

$$\begin{aligned} \text{for } i \in 1 \dots n, \quad & y_i \sim N(\mu_i, \sigma^2), \\ & \mu_i = \underbrace{b_0 + b_1 x_i}_{\text{fixed effects}} + \underbrace{\zeta_{[s_i]0} + \zeta_{[s_i]1} x_i}_{\text{random effects}}, \\ \text{for } j \in 1 \dots J, \quad & \vec{\zeta}_j \sim N(0, \Sigma). \end{aligned}$$

- As we can see from this, a multilevel normal linear model is equivalent to a non-multilevel model (the *fixed effects* models) plus a normally distributed random variation to the intercept and slope for each subject (the *random effects*).

- The fixed effects are sometimes known as *population level* effects: they apply to all observations.
- The random effects, on the other hand, vary across each different value of the grouping variable, which in this example is an individual participant in the experiment.
- Put another way, the fixed effects give the average effects in the population.
- The extent to which each individual varies around this average is given by the random effects.
- That the multilevel linear model can be described in terms of fixed and random effects is why these models are known as a *linear mixed effects model*.

- We can implement this model in R using `lme4::lmer`.

```
M_ml <- lmer(Reaction ~ Days + (Days|Subject),
             data = sleepstudy)
```

- The syntax here matches the fixed and random effects description of the model.
- The `Reaction ~ Days` tells us that the fixed effects model is a simple linear regression model with one predictor, and so with one intercept and one slope term.
- The `(Days|Subjects)` tells us that there is random variation to the slope for `Days` and implicitly there's also random variation to the intercept term.
- We could make the variation to the intercept term explicit by writing `(1 + Days|Subject)`, which is identical to `(Days|Subject)` because the `1 +` is included always by default just as it is included by default in fixed effects part, as it is in any R regression formula syntax.

- The results of this model is obtained as follows.

```
summary(M_ml)
```

- The value of $\vec{b}$ is available under `Estimate` in the `Fixed effects`, and we can get these directly as follows.

```
b <- fixef(M_ml)
b
#> (Intercept)        Days
#>   251.40510    10.46729
```

- Thus, the average effect of sleep deprivation on reaction time across all individuals is that their reaction time increases by 10.47 each day.
- Also, the average individual has an average reaction time of 251.41 on day 0 of the experiment, which means that this is the average reaction time of the average person generally.

- The values in the covariance matrix $\Sigma$ and of the residual standard deviation $\sigma$ can be obtained from the values provided under `Random effects`.
- These are available more directly as follows.

```
VarCorr(M_ml)
#>  Groups   Name        Std.Dev. Corr
#>  Subject  (Intercept) 24.7407
#>           Days         5.9221  0.066
#>  Residual              25.5918
```

- Note that the covariance matrix is defined as follows.

$$\Sigma = \begin{bmatrix} \tau_0^2 & \tau_0 \rho \tau_1 \\ \tau_0 \rho \tau_1 & \tau_1^2 \end{bmatrix}.$$

- The estimates of each $\vec{\beta}_j$ for $j \in 1 \ldots J$ can be obtained using the `coef` function.

```
coef(M_ml)$Subject %>%
  head()
#>     (Intercept)      Days
#> 308    253.6637 19.666262
#> 309    211.0064  1.847605
#> 310    212.4447  5.018429
#> 330    275.0957  5.652936
#> 331    273.6654  7.397374
#> 332    260.4447 10.195109
```

# *Shrikage*

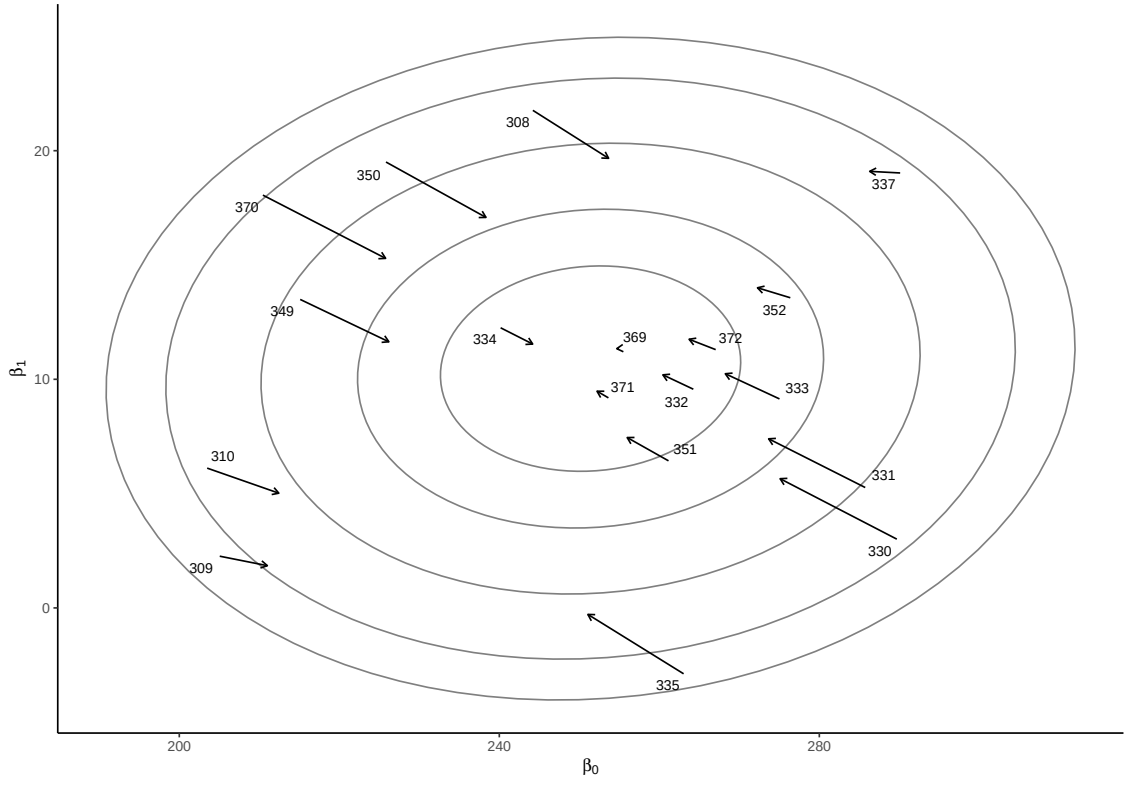



Figure 8: The contour plot shows the contours of the 2d normal distribution centered at $\vec{b}$ and whose covariance matrix is $\Sigma$.

# *Varying intercepts or varying slopes only models*

- The above model allowed for random variation in both the intercepts and slopes but we can choose to have random variation in only one or the other.
- A varying intercept only multilevel model is defined as follows.

$$\begin{aligned} \text{for } i \in 1 \dots n, \quad y_i &\sim N(\mu_i, \sigma^2), \\ \mu_i &= \beta_{[s_i]0} + b_1 x_i, \\ \text{for } j \in 1 \dots J, \quad \beta_{j0} &\sim N(b_0, \tau_0^2), \end{aligned}$$

which can be rewritten, using the same reasoning as above,

$$\begin{aligned} \text{for } i \in 1 \dots n, \quad y_i &\sim N(\mu_i, \sigma^2), \\ \mu_i &= b_0 + b_1 x_i + \zeta_{[s_i]0}, \\ \text{for } j \in 1 \dots J, \quad \zeta_{j0} &\sim N(0, \tau_0^2). \end{aligned}$$

- Using `lmer`, we would implement this as follows.

```
M_ml_vi <- lmer(Reaction ~ Days + (1|Subject),
                data = sleepstudy)
```

- The fixed effects give us an estimate of the slope and intercept as before.

```
fixef(M_ml_vi)
#> (Intercept)        Days
#>   251.40510    10.46729
```

- The random effects just provide a measure of standard deviation $\tau_0$ for the random intercepts as well as residual standard deviation $\sigma$.

```
VarCorr(M_ml_vi)
#>  Groups   Name        Std.Dev.
#>  Subject  (Intercept) 37.124
#>  Residual             30.991
```

- Absent here, compared to the varying intercepts and varying slopes model is the estimate for $\tau_1$ and $\rho$.

# *Varying slope only*

- The varying slope only multilevel model allows only the slopes to vary across subjects and it leaves the intercepts fixed.
- It is defined as follows.

$$\begin{aligned} \text{for } i \in 1 \dots n, \quad & y_i \sim N(\mu_i, \sigma^2), \\ & \mu_i = b_0 + \beta_{[s_i]1} + x_i, \\ \text{for } j \in 1 \dots J, \quad & \beta_{j1} \sim N(b_0, \tau_1^2), \end{aligned}$$

which can be rewritten

$$\begin{aligned} \text{for } i \in 1 \dots n, \quad & y_i \sim N(\mu_i, \sigma^2), \\ & \mu_i = b_0 + b_1 x_i + \zeta_{[s_i]1} x_i, \\ \text{for } j \in 1 \dots J, \quad & \zeta_{j1} \sim N(0, \tau_1^2). \end{aligned}$$

- Using `lmer`, we would implement this as follows.

```
M_ml_vs <- lmer(Reaction ~ Days + (0+Days|Subject),
                data = sleepstudy)
```

- The fixed effects give us an estimate of both the slope and intercept as with the previous models.

```
fixef(M_ml_vs)
#> (Intercept)        Days
#>   251.40510    10.46729
```

- The random effect provide a measure of standard deviation $\tau_1$ and $\sigma$.

```
VarCorr(M_ml_vs)
#>  Groups   Name Std.Dev.
#>  Subject  Days  7.260
#>  Residual       29.018
```

Absent here compared to the full model is the estimate for $\tau_0$ and $\rho$.

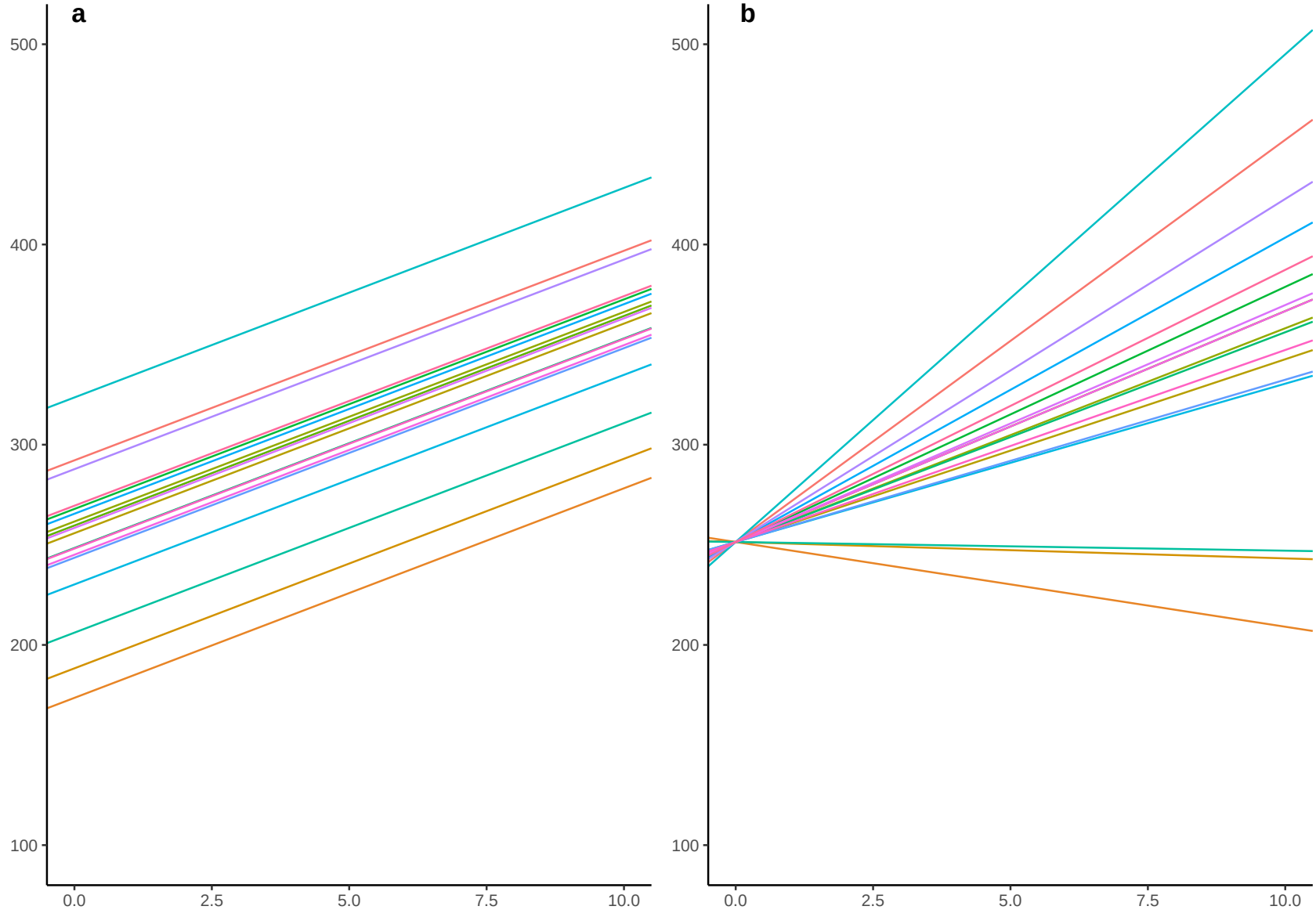



Figure 9: Lines of best fit for each data group in a varying intercepts only a) or varying slopes only b) multilevel linear model.

- One final variant of the full model is where we allow for both varying slopes and intercepts but assume no correlation between each $\beta_{j0}$ and $\beta_{j1}$.
- In other words, we assume that these are drawn from independent normal distributions.

$$\begin{aligned}
\text{for } i \in 1 \dots n, \quad & y_i \sim N(\mu_i, \sigma^2), \\
& \mu_i = \beta_{[s_i]0} + \beta_{[s_i]1} x_i, \\
\text{for } j \in 1 \dots J, \quad & \beta_{j0} \sim N(b_0, \tau_0^2), \\
& \beta_{j1} \sim N(b_1, \tau_1^2),
\end{aligned}$$

which is identical to each $\beta_j$ vector being drawn from a diagonal covariance matrix, i.e. where $\rho = 0$.

- Using `lmer`, we would implement this as follows.

```
M_ml_diag <- lmer(Reaction ~ Days + (1|Subject) + (0+Days|Subjec
                  data = sleepstudy)
```

- We can obtain the same model using the following formula syntax.

```
M_ml_diag2 <- lmer(Reaction ~ Days + (Days||Subject),
                   data = sleepstudy)
```

- The fixed effects give us an estimate of both the slope and intercept as with each of the previous models.

```
fixef(M_ml_diag2)
#> (Intercept)        Days
#>   251.40510    10.46729
```

- The random effect provide a measure of the $\tau_0$, $\tau_1$ and $\sigma$ standard deviations.

```
VarCorr(M_ml_diag2)
#>  Groups    Name        Std.Dev.
#>  Subject   (Intercept) 25.0513
#>  Subject.1 Days         5.9882
#>  Residual               25.5653
```

- The only quantity that is absent here compared to the full model is the estimate for $\rho$.