第十章 多层模型

Wang Shujia

Department of Statistics, School of Economics
Shenzhen University

Outline

1. 多层模型的概念
2. 多层线性回归模型：运用 lm() 和 lmer()
3. 多层贝叶斯模型
4. 多层模型的预测

Outline

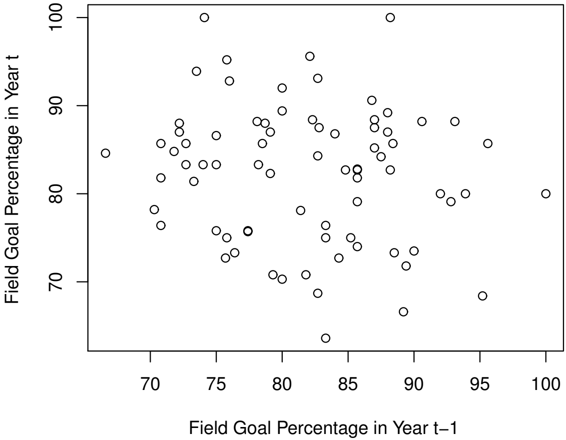
1. 多层模型的概念
   - 一个错误建模的实例
   - 多层模型
   - 面板数据的固定效应与随机效应
   - 多层贝叶斯模型

2006 年 11 月 12 日，星期天，纽约时报专栏作家 Aaron Schatz 发表文章 “N.F.L. Kickers Are Judged on the Wrong Criteria”，宣称 “有明显证据表明，球员一个赛季的射门命中率与下一季度的命中率没有相关性”
数据：全美橄榄球联盟（National Football League，NFL）19 名射门球员（至少射门 10 次以上）2002-2006 赛季。

	Name	Yeart	Teamt	FGAt	FGt	Team.t.1.	FGAtM1	FGtM1	FGAtM2	FGtM2
1	Adam Vinatieri	2003	NE	34	73.5	NE	30	90.0	NA	NA
2	Adam Vinatieri	2004	NE	33	93.9	NE	34	73.5	30	90.0
3	Adam Vinatieri	2005	NE	25	80.0	NE	33	93.9	34	73.5
4	Adam Vinatieri	2006	IND	19	89.4	NE	25	80.0	33	93.9
5	David Akers	2003	PHI	29	82.7	PHI	34	88.2	NA	NA
6	David Akers	2004	PHI	32	84.3	PHI	29	82.7	34	88.2

FGt：t 赛季射门得分率，FGtM1：$(t-1)$ 赛季射门得分率。
建立 FGt 对 FGtM1 的线性回归模型。

散点图显示两赛季得分率不相关



相关系数检验：不显著

```
setwd("F:/BaiduYun/Teaching/Rdata")
kicker<-read.csv("FieldGoals2003to2006.csv",header=T)
attach(kicker)
```

```
> cor.test(FGt,FGtM1)

        Pearson's product-moment correlation

data:  FGt and FGtM1
t = -1.2092, df = 74, p-value = 0.2305
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.3535538  0.0890568
sample estimates:
       cor
-0.1391935
```

线性回归模型：不显著

模型：$FGt = \beta_0 + \beta_1 FGtM1 + \varepsilon$

```
Call:
lm(formula = FGt ~ FGtM1)

Residuals:
     Min       1Q   Median       3Q      Max 
-18.4350  -7.0576   0.6933   5.3824  18.7047 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  94.6098    10.2525   9.228 6.18e-14 ***
FGtM1        -0.1510     0.1248  -1.209     0.23    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.723 on 74 degrees of freedom
Multiple R-squared:  0.01937,	Adjusted R-squared:  0.006123 
F-statistic: 1.462 on 1 and 74 DF,  p-value: 0.2305
```

但这个模型是错误的

因为 FGt 与 FGtM1 混杂了 19 位球员的能力水平的差异性，模型存在省略变量偏差。

新模型

对每位球员各赛季命中率进行回归（截距变，斜率不变），即增加 18 位球员（名字）作为虚拟变量。
即数学模型为：

$$\mathrm{FGt} = \alpha + \beta \mathrm{FGtM1} + \sum_{k=2}^{19} \gamma_k N_k + \varepsilon$$

其中 N_k 表示第 k 个球员的虚拟变量（1 或者 0），共 18 个，第一个球员是参考基准（其值为模型截距 α）。
R 代码：

```
fit.1 <- lm(FGt ~ FGtM1 + factor(Name), data = kicker)
summary(fit.1)
```

新模型结果

```
Call:
lm(formula = FGt ~ FGtM1 + factor(Name), data = kicker)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.1808  -4.0045  -0.5093   4.3053  13.3134 

Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)    
(Intercept)                      126.6872    10.0057  12.661  < 2e-16 ***
FGtM1                             -0.5037     0.1128  -4.467  3.9e-05 ***
factor(Name)David Akers           -4.6463     4.4007  -1.056  0.29559    
factor(Name)Jason Elam            -3.0167     4.4217  -0.682  0.49790    
factor(Name)Jason Hanson           2.1172     4.3949   0.482  0.63186    
factor(Name)Jay Feely            -10.3737     4.4514  -2.330  0.02341 *  
factor(Name)Jeff Reed             -8.2955     4.3994  -1.886  0.06454 .  
factor(Name)Jeff Wilkins           2.3102     4.3931   0.526  0.60106    
factor(Name)John Carney           -5.9774     4.4159  -1.354  0.18130    
factor(Name)John Hall             -8.4865     4.4528  -1.906  0.06180 .  
factor(Name)Kris Brown           -13.3598     4.5186  -2.957  0.00455 ** 
factor(Name)Matt Stover            8.7363     4.4060   1.983  0.05230 .  
factor(Name)Mike Vanderjagt        4.8955     4.3994   1.113  0.27055    
factor(Name)Neil Rackers          -6.6200     4.3985  -1.505  0.13793    
factor(Name)Olindo Mare          -13.0365     4.4528  -2.928  0.00493 ** 
factor(Name)Phil Dawson            3.5524     4.3931   0.809  0.42215    
factor(Name)Rian Lindell          -4.8674     4.4244  -1.100  0.27598    
factor(Name)Ryan Longwell         -2.2315     4.3970  -0.508  0.61379    
factor(Name)Sebastian Janikowski  -3.9763     4.4126  -0.901  0.37138    
factor(Name)Shayne Graham          2.1350     4.3932   0.486  0.62888    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

也可以用无常数项回归模型

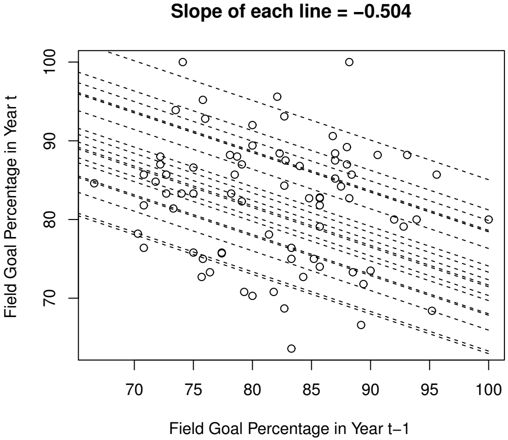
注意：无常数项回归的 R^2 是不准确的（计算方法问题）

```
fit.1 <- lm(FGt ~ FGtM1 + factor(Name)-1, data = kicker)
summary(fit.1)
```

```
Call:
lm(formula = FGt ~ FGtM1 + factor(Name) - 1, data = kicker)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.1808  -4.0045  -0.5093   4.3053  13.3134 

Coefficients:
                                Estimate Std. Error t value Pr(>|t|)    
FGtM1                            -0.5037     0.1128  -4.467  3.9e-05 ***
factor(Name)Adam Vinatieri      126.6872    10.0057  12.661  < 2e-16 ***
factor(Name)David Akers         122.0409     9.7515  12.515  < 2e-16 ***
factor(Name)Jason Elam          123.6705     9.5247  12.984  < 2e-16 ***
factor(Name)Jason Hanson        128.8044    10.1425  12.700  < 2e-16 ***
factor(Name)Jay Feely           116.3135     9.3224  12.477  < 2e-16 ***
factor(Name)Jeff Reed           118.3916     9.7729  12.114  < 2e-16 ***
factor(Name)Jeff Wilkins        128.9973     9.9387  12.979  < 2e-16 ***
factor(Name)John Carney         120.7098     9.5753  12.606  < 2e-16 ***
factor(Name)John Hall           118.2007     9.3144  12.690  < 2e-16 ***
factor(Name)Kris Brown          113.3274     9.0041  12.586  < 2e-16 ***
factor(Name)Matt Stover         135.4234    10.3332  13.106  < 2e-16 ***
factor(Name)Mike Vanderjagt     131.5827    10.2391  12.851  < 2e-16 ***
factor(Name)Neil Rackers        120.0672     9.7889  12.266  < 2e-16 ***
factor(Name)Olindo Mare         113.6507     9.3144  12.202  < 2e-16 ***
factor(Name)Phil Dawson         130.2396    10.0754  12.926  < 2e-16 ***
factor(Name)Rian Lindell        121.8198     9.5033  12.819  < 2e-16 ***
factor(Name)Ryan Longwell       124.4557     9.8183  12.676  < 2e-16 ***
factor(Name)Sebastian Janikowski 122.7109     9.6073  12.773  < 2e-16 ***
factor(Name)Shayne Graham       128.8222     9.9334  12.969  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.212 on 56 degrees of freedom
Multiple R-squared:  0.9958,    Adjusted R-squared:  0.9943
```

新模型结果

- 明显下降，即上赛季高，本赛季就低
- 不能得出本赛季命中率与上赛季无关的结论



以上两个模型的缺点

以上分析采用两种方式：

1. 完全混合（complete-pooling ）：所有数据看作来自一个总体

以上两个模型的缺点

以上分析采用两种方式：

1. 完全混合（complete-pooling ）：所有数据看作来自一个总体
   - 结论不正确，忽略了各球员之间的能力差异

以上两个模型的缺点

以上分析采用两种方式：

1. 完全混合（complete-pooling ）：所有数据看作来自一个总体
    - 结论不正确，忽略了各球员之间的能力差异
2. 不混合（no-pooling）：把每个球员看作一个单独总体（各配一条回归直线）

以上两个模型的缺点

以上分析采用两种方式：

1. 完全混合（complete-pooling ）：所有数据看作来自一个总体
    - 结论不正确，忽略了各球员之间的能力差异
2. 不混合（no-pooling）：把每个球员看作一个单独总体（各配一条回归直线）
    - 每条直线数据少，标准误大，参数估计不可靠

以上两个模型的缺点

以上分析采用两种方式：

1. 完全混合（complete-pooling ）：所有数据看作来自一个总体
   - 结论不正确，忽略了各球员之间的能力差异
2. 不混合（no-pooling）：把每个球员看作一个单独总体（各配一条回归直线）
   - 每条直线数据少，标准误大，参数估计不可靠
   - 变系数的误差大小无法估计

以上两个模型的缺点

以上分析采用两种方式：

1. 完全混合（complete-pooling ）：所有数据看作来自一个总体
   - 结论不正确，忽略了各球员之间的能力差异
2. 不混合（no-pooling）：把每个球员看作一个单独总体（各配一条回归直线）
   - 每条直线数据少，标准误大，参数估计不可靠
   - 变系数的误差大小无法估计
   - 模型多，忽视了球员之间的共性规律

以上两个模型的缺点

以上分析采用两种方式：

1. 完全混合（complete-pooling ）：所有数据看作来自一个总体
    - 结论不正确，忽略了各球员之间的能力差异
2. 不混合（no-pooling）：把每个球员看作一个单独总体（各配一条回归直线）
    - 每条直线数据少，标准误大，参数估计不可靠
    - 变系数的误差大小无法估计
    - 模型多，忽视了球员之间的共性规律
3. 因此需要建立多层模型。

以上两个模型的缺点

以上分析采用两种方式：

1. 完全混合（complete-pooling ）：所有数据看作来自一个总体
   - 结论不正确，忽略了各球员之间的能力差异
2. 不混合（no-pooling）：把每个球员看作一个单独总体（各配一条回归直线）
   - 每条直线数据少，标准误大，参数估计不可靠
   - 变系数的误差大小无法估计
   - 模型多，忽视了球员之间的共性规律
3. 因此需要建立多层模型。

以上两个模型的缺点

以上分析采用两种方式：

1. 完全混合（complete-pooling ）：所有数据看作来自一个总体
   - 结论不正确，忽略了各球员之间的能力差异
2. 不混合（no-pooling）：把每个球员看作一个单独总体（各配一条回归直线）
   - 每条直线数据少，标准误大，参数估计不可靠
   - 变系数的误差大小无法估计
   - 模型多，忽视了球员之间的共性规律
3. 因此需要建立多层模型。

本章主要参考教材：

Gelman, A. and Hill, J. (2007). Data Analysis Using Regression and Multilevel/Hierarchical Models. Cambridge: Cambridge University Press.

Outline

1. 多层模型的概念
   - 一个错误建模的实例
   - 多层模型
   - 面板数据的固定效应与随机效应
   - 多层贝叶斯模型

多层模型的概念

多层模型 (Multilevel/Hierarchical) 包含两层含义：

1. 数据的多层结构
   先从 28 个学院中随机抽取若干学院（群组，group），然后从样本学院中随机抽取若干学生（个体，individual）。
   这样抽取的样本数据具有两层结构。

多层模型的概念

多层模型 (Multilevel/Hierarchical) 包含两层含义：

1. 数据的多层结构
   先从 28 个学院中随机抽取若干学院（群组，group），然后从样本学院中随机抽取若干学生（个体，individual）。
   这样抽取的样本数据具有两层结构。
   - 分类抽样数据

多层模型的概念

多层模型 (Multilevel/Hierarchical) 包含两层含义：

1. 数据的多层结构
   先从 28 个学院中随机抽取若干学院（群组，group），然后从样本学院中随机抽取若干学生（个体，individual）。
   这样抽取的样本数据具有两层结构。
   - 分类抽样数据
   - 重复测量数据（Repeated measurements）（若干个体重复测量得到的数据）

多层模型的概念

多层模型 (Multilevel/Hierarchical) 包含两层含义：

1. 数据的多层结构
   先从 28 个学院中随机抽取若干学院（群组，group），然后从样本学院中随机抽取若干学生（个体，individual）。
   这样抽取的样本数据具有两层结构。
   - 分类抽样数据
   - 重复测量数据（Repeated measurements）（若干个体重复测量得到的数据）
   - 面板数据（Panel / Time-series cross-sectional data/Longitudinal）[注意概念的不同：截面（个体）$\times$ 时间]

多层模型的概念

多层模型 (Multilevel/Hierarchical) 包含两层含义：

1. 数据的多层结构
   先从 28 个学院中随机抽取若干学院（群组，group），然后从样本学院中随机抽取若干学生（个体，individual）。
   这样抽取的样本数据具有两层结构。
   - 分类抽样数据
   - 重复测量数据（Repeated measurements）（若干个体重复测量得到的数据）
   - 面板数据（Panel / Time-series cross-sectional data/Longitudinal）[注意概念的不同：截面（个体）$\times$ 时间]
2. 模型的多层结构
   预测深大各个学院毕业生收入 (y)（基于 GPA(x) 及其它信息）

多层模型的概念

多层模型 (Multilevel/Hierarchical) 包含两层含义：

1. 数据的多层结构
   先从 28 个学院中随机抽取若干学院（群组，group），然后从样本学院中随机抽取若干学生（个体，individual）。
   这样抽取的样本数据具有两层结构。
   - 分类抽样数据
   - 重复测量数据（Repeated measurements）（若干个体重复测量得到的数据）
   - 面板数据（Panel / Time-series cross-sectional data/Longitudinal）[注意概念的不同：截面（个体）$\times$ 时间]
2. 模型的多层结构
   预测深大各个学院毕业生收入 (y)（基于 GPA(x) 及其它信息）
   - 学生层 (student level)：对每个学院的学生可建立 y 对 x 的一个模型。

多层模型的概念

多层模型 (Multilevel/Hierarchical) 包含两层含义：

1. 数据的多层结构
   先从 28 个学院中随机抽取若干学院（群组，group），然后从样本学院中随机抽取若干学生（个体，individual）。
   这样抽取的样本数据具有两层结构。
   - 分类抽样数据
   - 重复测量数据（Repeated measurements）（若干个体重复测量得到的数据）
   - 面板数据（Panel / Time-series cross-sectional data/Longitudinal）[注意概念的不同：截面（个体）× 时间]
2. 模型的多层结构
   预测深大各个学院毕业生收入 (y)（基于 GPA(x) 及其它信息）
   - 学生层 (student level)：对每个学院的学生可建立 y 对 x 的一个模型。
   - 学院层 (College level)：由于各学院特点不同（文理科、对成绩要求的宽严程度等），又可以对学生层模型的系数建立模型。

为什么要多层模型？

1. 可分析回归系数可变时自变量对因变量的影响

为什么要多层模型？

1. 可分析回归系数可变时自变量对因变量的影响
2. 对数据偏少的群组，借用所有数据进行推断

为什么要多层模型？

1. 可分析回归系数可变时自变量对因变量的影响
2. 对数据偏少的群组，借用所有数据进行推断
3. 预测更准确：可以预测新个体和新群组

为什么要多层模型？

1. 可分析回归系数可变时自变量对因变量的影响
2. 对数据偏少的群组，借用所有数据进行推断
3. 预测更准确：可以预测新个体和新群组
4. 能够对结构化数据建模

为什么要多层模型？

1. 可分析回归系数可变时自变量对因变量的影响
2. 对数据偏少的群组，借用所有数据进行推断
3. 预测更准确：可以预测新个体和新群组
4. 能够对结构化数据建模
5. 对多层结构的数据和模型，能得到正确的标准误估计

多层线性回归模型

考虑学生层只有一个预测变量 x（如 GPA），学院层有一个预测变量 u（比如校友捐赠数额）。
没有预测变量和有多预测变量情形类似。个
(1) 变截距模型（Varying-intercept model）:

$$
\begin{aligned}
y_i &= \alpha_{j[i]} + \beta x_i + \epsilon_i, \text{ for students } i = 1, 2, \ldots, n \\
\alpha_j &= a + b u_j + \delta_j, \delta_j \sim N(0, \sigma_\alpha^2) \text{ for schools j} = 1, 2, \ldots \text{J} \\
\epsilon_i &\sim N(0, \sigma_y^2)
\end{aligned}
$$

其中 i 表示第 i 个学生，$\alpha_{j[i]}$表示 第 i 个学生所属的学院 j；x_i 和 u_j 分别表示学生层和学院层的预测变量，ϵ_i 和 δ_j 分别表示学生层和学院层的独立随机误差。

(2) 变斜率模型（varying-slope model）：

$$
\begin{aligned}
y_i &= \alpha + \beta_{j[i]} x_i + \epsilon_i, \text{ for students } i = 1, 2, \ldots, n \\
\beta_j &= a_1 + b_1 u_j + \delta_{j2} \text{ for schools j} = 1, 2, \ldots \text{J} \\
\epsilon_i &\sim N(0, \sigma_y^2) \\
\delta_{j2} &\sim N(0, \sigma_\alpha^2)
\end{aligned}
$$

(3) 变截距和变斜率模型（Varying-intercept, varying-slope model）:

$$
\begin{aligned}
y_i &= \alpha_{j[i]} + \beta_{j[i]} x_i + \epsilon_i, \text{ for students } i = 1, 2, \ldots, n \\
\alpha_j &= a_0 + b_0 u_j + \delta_{j1} \text{ for schools j} = 1, 2, \ldots \text{J} \\
\beta_j &= a_1 + b_1 u_j + \delta_{j2} \text{ for schools j} = 1, 2, \ldots \text{J} \\
\epsilon_i &\sim N(0, \sigma_y^2) \\
\begin{pmatrix} \delta_1 \\ \delta_2 \end{pmatrix} &\sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_\alpha^2 & \rho\sigma_\alpha\sigma_\beta \\ \rho\sigma_\alpha\sigma_\beta & \sigma_\beta^2 \end{pmatrix} \right)
\end{aligned}
$$

固定效应和随机效应模型

多层模型常被称为随机效应模型（random-effects）或混合效应模型（mixed-effects）。

1. 随机效应模型（random-effects）：如果回归系数看作一个模型（或分布）的随机结果

固定效应和随机效应模型

多层模型常被称为随机效应模型（random-effects）或混合效应模型（mixed-effects）。

1. 随机效应模型（random-effects）：如果回归系数看作一个模型（或分布）的随机结果
   - 比如变截距模型的截距项就是随机效应

固定效应和随机效应模型

多层模型常被称为随机效应模型（random-effects）或混合效应模型（mixed-effects）。

1. 随机效应模型（random-effects）：如果回归系数看作一个模型（或分布）的随机结果
   - 比如变截距模型的截距项就是随机效应
2. 固定效应模型（fixed effects）：回归系数不变（如系数不随学院改变），或系数虽有变化但没有认为它们是随机模型的结果

固定效应和随机效应模型

多层模型常被称为随机效应模型（random-effects）或混合效应模型（mixed-effects）。

1. 随机效应模型（random-effects）：如果回归系数看作一个模型（或分布）的随机结果
   - 比如变截距模型的截距项就是随机效应
2. 固定效应模型（fixed effects）：回归系数不变（如系数不随学院改变），或系数虽有变化但没有认为它们是随机模型的结果
   - 橄榄球员作为虚拟变量的不混合回归模型

固定效应和随机效应模型

多层模型常被称为随机效应模型（random-effects）或混合效应模型（mixed-effects）。

1. 随机效应模型（random-effects）：如果回归系数看作一个模型（或分布）的随机结果
   - 比如变截距模型的截距项就是随机效应
2. 固定效应模型（fixed effects）：回归系数不变（如系数不随学院改变），或系数虽有变化但没有认为它们是随机模型的结果
   - 橄榄球员作为虚拟变量的不混合回归模型
   - 在变截距模型中，如果 α_j 没有概率分布假设，而是用 $(J-1)$ 个虚拟变量来表达群组层的差异，则属于固定效应

固定效应和随机效应模型

多层模型常被称为随机效应模型（random-effects）或混合效应模型（mixed-effects）。

1. 随机效应模型（random-effects）：如果回归系数看作一个模型（或分布）的随机结果
   - 比如变截距模型的截距项就是随机效应
2. 固定效应模型（fixed effects）：回归系数不变（如系数不随学院改变），或系数虽有变化但没有认为它们是随机模型的结果
   - 橄榄球员作为虚拟变量的不混合回归模型
   - 在变截距模型中，如果 α_j 没有概率分布假设，而是用 $(J-1)$ 个虚拟变量来表达群组层的差异，则属于固定效应
3. 混合效应模型（mixed-effects）：同时包括随机效应和固定效应（比如变截距模型的截距项是随机效应，而斜率是固定效应）

固定效应和随机效应模型：注意事项

- “随机效应”、“混合效应”和“固定效应”等名词在不同情境下有不同含义，不同作者也有不同定义，容易引起误解，应避免使用。如果要用，一定要明确定义其具体含义
- 用“固定效应”还是“随机效应”?
  - 固定效应”是“随机效应”特例：如变截距模型中设 σ_α^2 为 0 或 ∞
- 因此，**多层模型完全包含了随机效应、固定效应和混合效应模型**
- 贝叶斯框架下，不存在这些混淆。

方差比较

在变截距模型中，记

- σ_y^2 为群组内方差（within-group）（假设各群组方差相等）
- σ_α^2 为群组间方差（between-group）（各群组均值的差异性）

比较两个方差对多层模型极为重要：

- 从模型拟合角度看，σ_y^2 越小越好，σ_α^2 越大越好
- 如果组间方差相对很小，则可取消分组，用完全混合模型
- **组内相关系数** (intraclass correlation coefficient, ICC)

$$ICC = \frac{\sigma_\alpha^2}{\sigma_\alpha^2 + \sigma_y^2}$$

- Sagan (2013) 建议 ICC 小至 0.05~0.20 之间都应当采用分层模型。

多层模型的“借力作用”（borrow strength）

Gelman and Hill （2007）：参数 α_j 的多层模型估计

$$\hat{\alpha}_j \approx \left(\frac{n_j}{\sigma_y^2}\bar{y}_j + \frac{1}{\sigma_\alpha^2}\bar{y}_{\text{all}}\right) \Big/ \left(\frac{n_j}{\sigma_y^2} + \frac{1}{\sigma_\alpha^2}\right)$$

其中 $\bar{y}_j$ 表示第 j 组的平均值（不混合估计），$\bar{y}_{\text{all}}$ 表示所有数据的平均（完全混合估计）。

- 若 n_j 很小，则 $\bar{y}_j$ 代表性差。然而在多层模型估计中，该组所占比重就小，该组的估计值更多的依赖于 $\bar{y}_{\text{all}}$，而不是 $\bar{y}_j$，这就是多层模型的“借力作用”
- 若 $n_j = 0$，该组没有观察值（未知群组），则估计值等于 $\bar{y}_{\text{all}}$
- 若 n_j 很大，则 $\bar{y}_j$ 占很大比重
- 若 n_j 在中间大小，则多层模型的估计值介于两个极端之间
- **请类似分析方差 σ_y^2 和 σ_α^2 的作用**

Outline

1. 多层模型的概念
   - 一个错误建模的实例
   - 多层模型
   - 面板数据的固定效应与随机效应
   - 多层贝叶斯模型

面板模型

- 混合回归 (Pooled regression)：每个个体拥有完全相同的回归方程
  - 忽视了个体间的异质性，该异质性可能与解释变量有关
- 个体独立回归：每个个体拥有单独的回归方程
  - 忽视了个体间的共性，个别个体观察样本数可能过少

个体效应模型 (Individual-specified effects model)

个体效应模型：个体回归方程拥有相同斜率，但可以有不同截距

$$y_{it} = X_{it}'\beta + Z_i'\delta + u_i + \varepsilon_{it} \tag{1}$$

其中

- Z_i 为不随时间而变得个体特征（如性别）
- X_{it} 可以随个体和时间而变 (Time varying)
- u_i 为代表个体异质性的截距项（个体效应，是不可观测的随机变量）
- $(u_i + \varepsilon_{it})$ 称为“复合扰动项”
- ε_{it} 假定独立同分布且与 u_i 不相关
- 以上模型也叫“不可观测效应模型”(Unobserved effects model)。

面板数据的固定效应

如果 u_i 与一个或多个自变量存在相关性，则称为**固定效应模型** (Fixed Effects, FE)

- 即使在固定效应模型中，个体效应 u_i 也是随机的，而不是常数

面板数据的固定效应

如果 u_i 与一个或多个自变量存在相关性，则称为**固定效应模型** (Fixed Effects, FE)

- 即使在固定效应模型中，个体效应 u_i 也是随机的，而不是常数
- 在模型 (1) 中对时间平均，

$$(y_{it} - \bar{y}_i) = (X'_{it} - \bar{X}'_i)\beta + (\varepsilon_{it} - \bar{\varepsilon}_i) \tag{2}$$

只要 $\bar{\varepsilon}_{it} = \varepsilon_{it} - \bar{\varepsilon}_i$ 与 $\bar{X}_{it} = X_{it} - \bar{X}_i$ 不相关，就可以用 OLS 得到一致的**固定效应估计量** (Fixed Effects Estimator) $\hat{\beta}_{FE}$

面板数据的固定效应

如果 u_i 与一个或多个自变量存在相关性，则称为**固定效应模型** (Fixed Effects, FE)

- 即使在固定效应模型中，个体效应 u_i 也是随机的，而不是常数
- 在模型 (1) 中对时间平均，

$$(y_{it} - \bar{y}_i) = (X'_{it} - \bar{X}'_i)\beta + (\varepsilon_{it} - \bar{\varepsilon}_i) \quad (2)$$

  只要 $\bar{\varepsilon}_{it} = \varepsilon_{it} - \bar{\varepsilon}_i$ 与 $\bar{X}_{it} = X_{it} - \bar{X}_i$ 不相关，就可以用 OLS 得到一致的**固定效应估计量** (Fixed Effects Estimator) $\hat{\beta}_{FE}$
- 由于 $\hat{\beta}_{FE}$ 只用了组内离差信息，所以也叫**组内估计量** (Within Estimator)

面板数据的固定效应

如果 u_i 与一个或多个自变量存在相关性，则称为**固定效应模型** (Fixed Effects, FE)

- 即使在固定效应模型中，个体效应 u_i 也是随机的，而不是常数
- 在模型 (1) 中对时间平均，

$$(y_{it} - \bar{y}_i) = (X'_{it} - \bar{X}'_i)\beta + (\varepsilon_{it} - \bar{\varepsilon}_i) \tag{2}$$

  只要 $\bar{\varepsilon}_{it} = \varepsilon_{it} - \bar{\varepsilon}_i$ 与 $\bar{X}_{it} = X_{it} - \bar{X}_i$ 不相关，就可以用 OLS 得到一致的**固定效应估计量** (Fixed Effects Estimator) $\hat{\beta}_{FE}$
- 由于 $\hat{\beta}_{FE}$ 只用了组内离差信息，所以也叫**组内估计量** (Within Estimator)
- 个体固定效应模型可以解决**不随时间变化但随个体而异**的遗漏变量问题

面板数据的固定效应

如果 u_i 与一个或多个自变量存在相关性，则称为**固定效应模型** (Fixed Effects, FE)

- 即使在固定效应模型中，个体效应 u_i 也是随机的，而不是常数
- 在模型 (1) 中对时间平均，

$$(y_{it} - \bar{y}_i) = (X'_{it} - \bar{X}'_i)\beta + (\varepsilon_{it} - \bar{\varepsilon}_i) \tag{2}$$

  只要 $\bar{\varepsilon}_{it} = \varepsilon_{it} - \bar{\varepsilon}_i$ 与 $\bar{X}_{it} = X_{it} - \bar{X}_i$ 不相关，就可以用 OLS 得到一致的**固定效应估计量** (Fixed Effects Estimator) $\hat{\beta}_{FE}$
- 由于 $\hat{\beta}_{FE}$ 只用了组内离差信息，所以也叫**组内估计量** (Within Estimator)
- 个体固定效应模型可以解决**不随时间变化但随个体而异**的遗漏变量问题
- 李子奈：固定效应指模型截距对于不同个体存在实质上的差异

面板数据的固定效应

- 如果在模型（1）中引入 $(n-1)$ 个虚拟变量来代表不同个体（如果没有截距项，则引入 n 个），可以得到与离差模型（2）相同的结果

$$y_{it} = X_{it}'\beta + Z_i'\delta + \sum_{k=1}^{n} \lambda_k \mathrm{IE}_k + \varepsilon_{it} \tag{3}$$

面板数据的固定效应

- 如果在模型（1）中引入 $(n-1)$ 个虚拟变量来代表不同个体（如果没有截距项，则引入 n 个），可以得到与离差模型（2）相同的结果

$$y_{it} = X_{it}'\beta + Z_i'\delta + \sum_{k=1}^{n} \lambda_k \mathrm{IE}_k + \varepsilon_{it} \tag{3}$$

- 所以 FE 也叫最小二乘虚拟变量模型 (Least Square Dummy Variable Model, LSDVM)

面板数据的固定效应

- 如果在模型（1）中引入 $(n-1)$ 个虚拟变量来代表不同个体（如果没有截距项，则引入 n 个），可以得到与离差模型（2）相同的结果

$$y_{it} = X_{it}'\beta + Z_i'\delta + \sum_{k=1}^{n} \lambda_k \mathrm{IE}_k + \varepsilon_{it} \tag{3}$$

- 所以 FE 也叫最小二乘虚拟变量模型 (Least Square Dummy Variable Model, LSDVM)
- 还可以加入时间固定效应：

$$y_{it} = X_{it}'\beta + Z_i'\delta + \sum_{k=1}^{n} \lambda_k \mathrm{IE}_k + \sum_{s=1}^{T} \gamma_s \mathrm{TE}_s + \varepsilon_{it} \tag{4}$$

面板数据的固定效应

- 如果在模型（1）中引入 $(n-1)$ 个虚拟变量来代表不同个体（如果没有截距项，则引入 n 个），可以得到与离差模型（2）相同的结果

$$y_{it} = X_{it}'\beta + Z_i'\delta + \sum_{k=1}^{n} \lambda_k \mathrm{IE}_k + \varepsilon_{it} \tag{3}$$

- 所以 FE 也叫最小二乘虚拟变量模型 (Least Square Dummy Variable Model, LSDVM)
- 还可以加入时间固定效应：

$$y_{it} = X_{it}'\beta + Z_i'\delta + \sum_{k=1}^{n} \lambda_k \mathrm{IE}_k + \sum_{s=1}^{T} \gamma_s \mathrm{TE}_s + \varepsilon_{it} \tag{4}$$

- 缺点 1：条件严。要求 $\bar{\varepsilon}_{it}$ 与 $\bar{X}_{it}$ 不相关，扰动项必须与各期的解释变量都不相关（而不仅仅是当期解释变量）（严格外生性）

面板数据的固定效应

- 如果在模型（1）中引入 $(n-1)$ 个虚拟变量来代表不同个体（如果没有截距项，则引入 n 个），可以得到与离差模型（2）相同的结果

$$y_{it} = X_{it}'\beta + Z_i'\delta + \sum_{k=1}^{n} \lambda_k \mathrm{IE}_k + \varepsilon_{it} \tag{3}$$

- 所以 FE 也叫最小二乘虚拟变量模型 (Least Square Dummy Variable Model, LSDVM)
- 还可以加入时间固定效应：

$$y_{it} = X_{it}'\beta + Z_i'\delta + \sum_{k=1}^{n} \lambda_k \mathrm{IE}_k + \sum_{s=1}^{T} \gamma_s \mathrm{TE}_s + \varepsilon_{it} \tag{4}$$

- 缺点 1：条件严。要求 $\bar{\varepsilon}_{it}$ 与 $\bar{X}_{it}$ 不相关，扰动项必须与各期的解释变量都不相关（而不仅仅是当期解释变量）（严格外生性）
- 缺点 2：$\hat{\beta}_{FE}$ 无法估计不随时间变化的变量（性别、距离等）对因变量的影响，因为 $Z_i'\delta$ 被消去了

面板数据的随机效应

如果 u_i 与所有自变量 $\{X_{it}, Z_i\}$ 均不相关，则称为**随机效应模型** (Randon Effects, RE)。

- OLS 是一致的，但 $(u_i + \varepsilon_{it})$ 不是球形扰动项，因此 OLS 不是最有效率的

面板数据的随机效应

如果 u_i 与所有自变量 $\{X_{it}, Z_i\}$ 均不相关，则称为**随机效应模型** (Randon Effects, RE)。

- OLS 是一致的，但 $(u_i + \varepsilon_{it})$ 不是球形扰动项，因此 OLS 不是最有效率的
- 同一个体不同时期之间的扰动项存在自相关，

$$\rho = \mathrm{corr}(u_i + \varepsilon_{it}, u_i + \varepsilon_{is}) = \frac{\sigma_u^2}{\sigma_u^2 + \sigma_\varepsilon^2}(t \neq s)$$

面板数据的随机效应

如果 u_i 与所有自变量 $\{X_{it}, Z_i\}$ 均不相关，则称为**随机效应模型** (Randon Effects, RE)。

- OLS 是一致的，但 $(u_i + \varepsilon_{it})$ 不是球形扰动项，因此 OLS 不是最有效率的
- 同一个体不同时期之间的扰动项存在自相关，

$$\rho = \mathrm{corr}(u_i + \varepsilon_{it}, u_i + \varepsilon_{is}) = \frac{\sigma_u^2}{\sigma_u^2 + \sigma_\varepsilon^2}(t \neq s)$$

- 有时称

$$\theta = 1 - \frac{\sigma_\varepsilon}{\sqrt{T\sigma_u^2 + \sigma_\varepsilon^2}}$$

  的估计量 $\hat{\theta}$ 为随机效应估计 (random effects estimator)，$\hat{\theta} = 0$ 为混合估计，$\hat{\theta} = 1$ 则为组内估计

面板数据的随机效应

如果 u_i 与所有自变量 $\{X_{it}, Z_i\}$ 均不相关，则称为**随机效应模型** (Randon Effects, RE)。

- OLS 是一致的，但 $(u_i + \varepsilon_{it})$ 不是球形扰动项，因此 OLS 不是最有效率的
- 同一个体不同时期之间的扰动项存在自相关，

$$\rho = \mathrm{corr}(u_i + \varepsilon_{it}, u_i + \varepsilon_{is}) = \frac{\sigma_u^2}{\sigma_u^2 + \sigma_\varepsilon^2}(t \neq s)$$

- 有时称

$$\theta = 1 - \frac{\sigma_\varepsilon}{\sqrt{T\sigma_u^2 + \sigma_\varepsilon^2}}$$

  的估计量 $\hat{\theta}$ 为随机效应估计 (random effects estimator)，$\hat{\theta} = 0$ 为混合估计，$\hat{\theta} = 1$ 则为组内估计
- 所有扰动项的协方差矩阵记为 Ω，则基于广义最小二乘法可以得到**随机效应估计量** $\hat{\beta}_{RE}$

面板数据的随机效应

如果 u_i 与所有自变量 $\{X_{it}, Z_i\}$ 均不相关，则称为**随机效应模型** (Randon Effects, RE)。

- OLS 是一致的，但 $(u_i + \varepsilon_{it})$ 不是球形扰动项，因此 OLS 不是最有效率的
- 同一个体不同时期之间的扰动项存在自相关，

$$\rho = \mathrm{corr}(u_i + \varepsilon_{it}, u_i + \varepsilon_{is}) = \frac{\sigma_u^2}{\sigma_u^2 + \sigma_\varepsilon^2}(t \neq s)$$

- 有时称

$$\theta = 1 - \frac{\sigma_\varepsilon}{\sqrt{T\sigma_u^2 + \sigma_\varepsilon^2}}$$

  的估计量 $\hat{\theta}$ 为随机效应估计 (random effects estimator)，$\hat{\theta} = 0$ 为混合估计，$\hat{\theta} = 1$ 则为组内估计
- 所有扰动项的协方差矩阵记为 Ω，则基于广义最小二乘法可以得到**随机效应估计量** $\hat{\beta}_{RE}$
- 李子奈：随机效应指模型截距对于不同个体只存在随机扰动的差异

Outline

1. 多层模型的概念
   - 一个错误建模的实例
   - 多层模型
   - 面板数据的固定效应与随机效应
   - 多层贝叶斯模型

设 y_i 表示第 i 个观察值，θ_j 表示第 j 组的参数。

$$
\begin{aligned}
Y_i &\sim f(y_i|\theta_j)(i = 1, 2, \ldots, n,\ j = 1, 2, \ldots, J) \\
\theta_j &\sim p(\theta_j|\phi) \text{ prior for parameter } \theta_j \\
\phi &\sim p(\phi) \text{ prior for hyperparemeter } \phi
\end{aligned}
$$

以上 $p(\phi)$ 可认为是 θ_j 的共同先验分布，θ_j 可视为来自这一分布的不同随机样本（不一定独立）。
后验分布

$$
p(\theta_1, \theta_2, \ldots, \theta_J, \phi|y) \propto p(\phi)p(\theta_1, \theta_2, \ldots, \theta_J|\phi)f(y|\theta_1, \theta_2, \ldots, \theta_J, \phi)
$$

模型中 $(\theta_1, \theta_2, \ldots, \theta_J)$ 满足**可交换性** (exchangeability)：即分布 $p(\theta_1, \theta_2, \ldots, \theta_J)$ 与变量顺序无关。

变截距贝叶斯模型

- 变截距模型（群组层没有预测变量）:

$$
\begin{aligned}
&y_i = \alpha_{j[i]} + \beta x_i + \epsilon_i,\ \ \epsilon_i \sim N(0, \sigma_y^2) \\
&\alpha_j \sim N(\mu_\alpha, \sigma_\alpha^2), \beta \sim N(\mu_{\beta 0}, \sigma_{\beta 0}^2), \sigma_y \sim \text{uniform}(0, 100) \\
&\mu_\alpha \sim N(\mu_{\alpha 0}, \sigma_{\alpha 0}^2), \sigma_\alpha \sim \text{uniform}(0, 100)
\end{aligned}
$$

变截距贝叶斯模型

- 变截距模型（群组层没有预测变量）:

$$
\begin{aligned}
y_i = \alpha_{j[i]} + \beta x_i + \epsilon_i,\ \epsilon_i \sim N(0, \sigma_y^2) \\
\alpha_j \sim N(\mu_\alpha, \sigma_\alpha^2), \beta \sim N(\mu_{\beta 0}, \sigma_{\beta 0}^2), \sigma_y \sim \text{uniform}(0, 100) \\
\mu_\alpha \sim N(\mu_{\alpha 0}, \sigma_{\alpha 0}^2), \sigma_\alpha \sim \text{uniform}(0, 100)
\end{aligned}
$$

- 变截距模型（群组层有一个预测变量）:

$$
\begin{aligned}
y_i &= \alpha_{j[i]} + \beta x_i + \epsilon_i,\ \epsilon_i \sim N(0, \sigma_y^2) \\
\alpha_j &= a + b u_j + \delta_j, \delta_j \sim N(0, \sigma_\alpha^2) \\
a &\sim N(\mu_{a0}, \sigma_{a0}^2), b \sim N(\mu_{b0}, \sigma_{b0}^2) \\
\beta &\sim N(\mu_\beta, \sigma_\beta^2) \\
\sigma_y &\sim \text{uniform}(0, 100), \sigma_\alpha \sim \text{uniform}(0, 100)
\end{aligned}
$$

变截距和变斜率：独立先验

变截距和变系数模型:

$$
\begin{aligned}
y_i &= \alpha_{j[i]} + \beta_{j[i]} x_i + \epsilon_i \\
\epsilon_i &\sim N(0, \sigma^2)
\end{aligned}
$$

分层独立先验：

$$
\begin{aligned}
\alpha_j &\sim N(\alpha_1, \sigma_\alpha^2) \\
\beta_j &\sim N(\beta_1, \sigma_\beta^2)
\end{aligned}
$$

超参数 $\alpha_1, \beta_1, \sigma_\alpha, \sigma_\beta, \sigma$ 相互独立，且均为无信息先验分布，如 $\text{uniform}(0, 100), N(0, 1000), \text{gamma}(0.001, 0.001)$ 等。

变截距和变斜率：多元正态先验

变截距和变变斜率模型:

$$
\begin{aligned}
y_i &= \alpha_{j[i]} + \beta_{j[i]} x_i + \epsilon_i \\
\epsilon_i &\sim N(0, \sigma^2)
\end{aligned}
$$

多元正态分层先验：

$$
\begin{aligned}
\boldsymbol{\theta}_j = (\alpha_j, \beta_j)^T &\sim N_2(\boldsymbol{\mu}_\theta, \boldsymbol{\Sigma}) \\
\boldsymbol{\Sigma} &\sim \mathrm{Inv-Wishart}(2, \boldsymbol{\Omega})
\end{aligned}
$$

超参数 $\boldsymbol{\mu}_\theta$ 为无信息先验，$\boldsymbol{\Omega}$ 为已知。

Outline

处理多层线性回归模型的一般策略

1. 拟合传统线性模型：完全混合或不混合（用 lm() 和 glm()）

处理多层线性回归模型的一般策略

1. 拟合传统线性模型：完全混合或不混合（用 lm() 和 glm()）
2. 建立多层模型：变截距、变斜率（用 lme4 软件包中的 lmer()）

处理多层线性回归模型的一般策略

1. 拟合传统线性模型：完全混合或不混合（用 lm() 和 glm()）
2. 建立多层模型：变截距、变斜率（用 lme4 软件包中的 lmer()）
3. 建立贝叶斯多层模型。可方便地得到参数估计、参数不确定性、预测以及其他感兴趣的量（用 Bugs 或 rstan）

处理多层线性回归模型的一般策略

1. 拟合传统线性模型：完全混合或不混合（用 lm() 和 glm()）
2. 建立多层模型：变截距、变斜率（用 lme4 软件包中的 lmer()）
3. 建立贝叶斯多层模型。可方便地得到参数估计、参数不确定性、预测以及其他感兴趣的量（用 Bugs 或 rstan）
4. 对于很大或很复杂的模型，可能还要应用 R 编程

不混合模型:lm()

```
fit1<-lm(FGt~FGtM1+factor(Name)-1,data=kicker)
summary(fit1)
```

```
Call:
lm(formula = FGt ~ FGtM1 + factor(Name) - 1, data = kicker)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.1808  -4.0045  -0.5093   4.3053  13.3134 

Coefficients:
                                Estimate Std. Error t value Pr(>|t|)    
FGtM1                            -0.5037     0.1128  -4.467  3.9e-05 ***
factor(Name)Adam Vinatieri      126.6872    10.0057  12.661  < 2e-16 ***
factor(Name)David Akers         122.0409     9.7515  12.515  < 2e-16 ***
factor(Name)Jason Elam          123.6705     9.5247  12.984  < 2e-16 ***
factor(Name)Jason Hanson        128.8044    10.1425  12.700  < 2e-16 ***
factor(Name)Jay Feely           116.3135     9.3224  12.477  < 2e-16 ***
factor(Name)Jeff Reed           118.3916     9.7729  12.114  < 2e-16 ***
factor(Name)Jeff Wilkins        128.9973     9.9387  12.979  < 2e-16 ***
factor(Name)John Carney         120.7098     9.5753  12.606  < 2e-16 ***
factor(Name)John Hall           118.2007     9.3144  12.690  < 2e-16 ***
factor(Name)Kris Brown          113.3274     9.0041  12.586  < 2e-16 ***
factor(Name)Matt Stover         135.4234    10.3332  13.106  < 2e-16 ***
factor(Name)Mike Vanderjagt     131.5827    10.2391  12.851  < 2e-16 ***
factor(Name)Neil Rackers        120.0672     9.7889  12.266  < 2e-16 ***
factor(Name)Olindo Mare         113.6507     9.3144  12.202  < 2e-16 ***
factor(Name)Phil Dawson         130.2396    10.0754  12.926  < 2e-16 ***
factor(Name)Rian Lindell        121.8198     9.5033  12.819  < 2e-16 ***
factor(Name)Ryan Longwell       124.4557     9.8183  12.676  < 2e-16 ***
factor(Name)Sebastian Janikowski 122.7109    9.6073  12.773  < 2e-16 ***
factor(Name)Shayne Graham       128.8222     9.9334  12.969  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.212 on 56 degrees of freedom
Multiple R-squared:  0.9958,    Adjusted R-squared:  0.9943
```

多层模型的 R 函数：lmer()

- lmer：Linear Mixed-Effects Models
- Usage：
  lmer(formula, data, control = lmerControl(), offset, ...)
- formula 与 lm 类似，包含：

(1|N)	Random intercept with fixed mean
0+offset(o)+(1|N)	Random intercept with *a priori* means
x+(x|N)	Correlated random intercept and slope
x+(x||g)	Uncorrelated random intercept and slope

变截距模型

```
> M1 <- lmer(formula = FGt ~ FGtM1 + (1|Name))
> summary(M1)
Linear mixed model fit by REML ['lmerMod']
Formula: FGt ~ FGtM1 + (1 | Name)

REML criterion at convergence: 516.3

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.97654 -0.63878  0.01243  0.70155  2.08720

Random effects:
 Groups   Name        Variance Std.Dev.
 Name     (Intercept) 23.83    4.882
 Residual             39.33    6.271
Number of obs: 76, groups:  Name, 19

Fixed effects:
            Estimate Std. Error t value
(Intercept) 113.5942     9.0760   12.52
FGtM1        -0.3830     0.1097   -3.49

Correlation of Fixed Effects:
      (Intr)
FGtM1 -0.989
```

共有 $n = 76$ 个观察值，$J = 19$ 个群组。

- 组间方差 $\sigma_\alpha^2 = 23.83$
- 组内方差 $\sigma_y^2 = 39.33$
- 组内相关系数 $ICC = -9.989$
- 固定效应与随机效应：固定效应计算群组平均值，随机效应计算分组层的误差
- 系数估计 = 固定效应 + 随机效应

```
> fixef (M1)
(Intercept)       FGtM1
113.5941631 -0.3830001
```

- 回归方程为：$FGt = 113.594 - 0.383FgtM1$

变截距模型：随机效应

```
> ranef(M1)
                    (Intercept)
Adam Vinatieri        2.0614356
David Akers          -1.0248991
Jason Elam            0.3103561
Jason Hanson          3.4513437
Jay Feely            -4.7355979
Jeff Reed            -3.6254304
Jeff Wilkins          3.7503049
John Carney          -1.8262472
John Hall            -3.3931570
Kris Brown           -6.5932217
Matt Stover           7.9855487
Mike Vanderjagt       5.3413200
Neil Rackers         -2.4520622
Olindo Mare          -6.6142648
Phil Dawson           4.5207692
Rian Lindell         -0.9827594
Ryan Longwell         0.6312282
Sebastian Janikowski -0.4352241
Shayne Graham         3.6305575
```

变截距模型：系数估计

```
> coef(M1)$Name
                     (Intercept)      FGtM1
Adam Vinatieri          115.6556 -0.3830001
David Akers             112.5693 -0.3830001
Jason Elam              113.9045 -0.3830001
Jason Hanson            117.0455 -0.3830001
Jay Feely               108.8586 -0.3830001
Jeff Reed               109.9687 -0.3830001
Jeff Wilkins            117.3445 -0.3830001
John Carney             111.7679 -0.3830001
John Hall               110.2010 -0.3830001
Kris Brown              107.0009 -0.3830001
Matt Stover             121.5797 -0.3830001
Mike Vanderjagt         118.9355 -0.3830001
Neil Rackers            111.1421 -0.3830001
Olindo Mare             106.9799 -0.3830001
Phil Dawson             118.1149 -0.3830001
Rian Lindell            112.6114 -0.3830001
Ryan Longwell           114.2254 -0.3830001
Sebastian Janikowski    113.1589 -0.3830001
Shayne Graham           117.2247 -0.3830001
```

变截距及变斜率 (且相关) 模型：lmer()

```
M2<-lmer(FGt~FGtM1+(FGtM1|Name),data=kicker)
summary(M2)
```

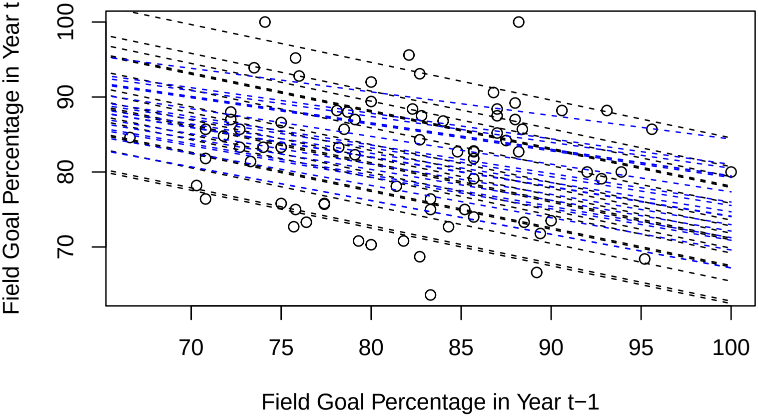
其中 (FGtM1|Name) 等价于 (1+FGtM1|Name), 变截距和变斜率

```
> # varying intercept and slope with a predictor
> M2 <- lmer( FGt ~ FGtM1 + (FGtM1 | Name), data = kicker)
> summary(M2)
Linear mixed model fit by REML ['lmerMod']
Formula: FGt ~ FGtM1 + (FGtM1 | Name)
   Data: kicker

REML criterion at convergence: 516.2

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.96553 -0.64381  0.01633  0.73555  2.09899

Random effects:
 Groups   Name        Variance Std.Dev. Corr
 Name     (Intercept)  1.533   1.23824
          FGtM1        0.002   0.04472  1.00
 Residual             39.190   6.26016
Number of obs: 76, groups:  Name, 19

Fixed effects:
            Estimate Std. Error t value
(Intercept) 113.8294     9.0246  12.613
FGtM1        -0.3872     0.1105  -3.505

Correlation of Fixed Effects:
      (Intr)
FGtM1 -0.989
```

变截距及变斜率 (且相关) 模型：lmer()

```
> coef(M2)$Name
                     (Intercept)      FGtM1
Adam Vinatieri          114.3153 -0.3696282
David Akers             113.5988 -0.3955098
Jason Elam              113.9210 -0.3838704
Jason Hanson            114.7296 -0.3546660
Jay Feely               112.6338 -0.4303615
Jeff Reed               112.9154 -0.4201931
Jeff Wilkins            114.7736 -0.3530747
John Carney             113.3670 -0.4038817
John Hall               112.9904 -0.4174832
Kris Brown              112.1164 -0.4490518
Matt Stover             115.8408 -0.3145270
Mike Vanderjagt         115.1090 -0.3409607
Neil Rackers            113.1816 -0.4105756
Olindo Mare             112.1146 -0.4491151
Phil Dawson             114.9987 -0.3449431
Rian Lindell            113.6138 -0.3949663
Ryan Longwell           114.0201 -0.3802922
Sebastian Janikowski    113.7374 -0.3905027
Shayne Graham           114.7804 -0.3528287
```

unpooled and multilevel



Outline

准备数据及编号指标

```
setwd("F:/BaiduYun/Teaching/Rdata")
kicker<-read.csv("FieldGoals2003to2006.csv",header=T)
y<-kicker$FGt
x<-kicker$FGtM1
n<-length(y)
player.name <- as.vector(kicker$Name)
uniq<- unique(player.name)
J<-length(uniq)
player <- rep (NA, J)
for (i in 1:J){
   player[player.name==uniq[i]]<-i
}
```

模型代码（中心化）：保存为 ch8kicker_c.txt

```
# varying-intercept model(centering)
model {
  for (i in 1:n){
     y[i] ~ dnorm (y.hat[i], tau.y)
     y.hat[i]<-intercept[player[i]]+b*(x[i]-mean(x[]))
     }
   b ~ dnorm (0, .0001)
   tau.y <- pow(sigma.y, -2)
   sigma.y~dunif(0,100)
  for (j in 1:J){
      intercept[j] ~ dnorm (mu.a, tau.a)
      a[j] <- intercept[j]-b*mean(x[])
      }
  mu.a ~ dnorm (0, .0001)
  tau.a <- pow(sigma.a, -2)
  sigma.a ~ dunif(0,100)
  }
```

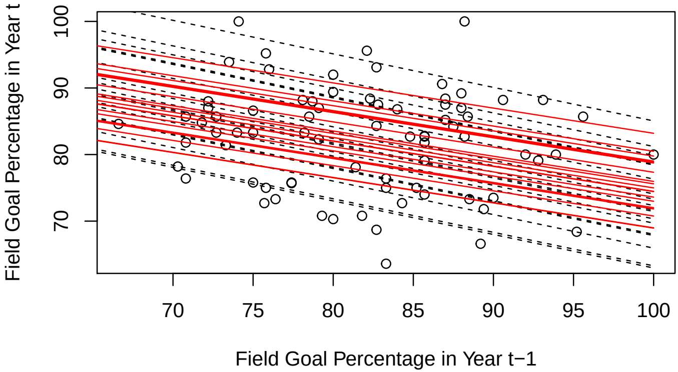
调用 R2WinBUGS

```
data <- list ("n", "J", "x", "y", "player")
inits <- function (){
  list(intercept=rnorm(J,80,10), b=rnorm(1),
  mu.a=rnorm(1),sigma.y=runif(1),sigma.a=runif(1))}
parameters <- c ("a", "b", "mu.a", "sigma.y", "sigma.a")
output<-bugs(data,
             inits,
             parameters,
             n.chains=3,
             n.iter=2000,
             n.burnin=1000,
             n.thin=5,
             debug=FALSE,
             codaPkg=FALSE,
             model.file="ch8kicker_c.txt",
             bugs.directory="C:/WinBUGS14/")
```

运行结果

```
Inference for Bugs model at "F:\BaiduYun\Teaching\Rdata\Ch8kmodel.txt", fit using WinBUGS
 2 chains, each with 2000 iterations (first 1000 discarded), n.thin = 5
 n.sims = 400 iterations saved
                 mean    sd   2.5%    25%    50%    75%  97.5% Rhat n.eff
intercept[1]   115.31 10.92  94.32 107.27 115.30 122.72 136.51 1.00   400
intercept[2]   112.23 10.85  92.83 105.00 112.05 120.35 132.71 1.00   400
intercept[3]   113.57 10.33  94.63 106.27 113.60 120.72 132.80 1.01   400
intercept[4]   116.58 11.00  95.37 108.87 117.15 124.42 137.40 1.01   400
intercept[5]   108.70  9.72  90.16 101.67 108.30 116.00 126.12 1.01   400
intercept[6]   110.37 10.28  90.82 102.77 110.40 117.72 129.80 1.01   400
intercept[7]   116.81 10.95  95.57 109.40 116.90 124.05 137.31 1.01   400
intercept[8]   111.31 10.17  92.21 104.40 111.60 118.10 130.32 1.01   400
intercept[9]   110.20 10.12  90.80 102.80 109.60 117.05 129.50 1.00   400
intercept[10]  107.09  9.48  90.11 100.40 107.10 113.70 124.92 1.01   400
intercept[11]  121.01 11.86  97.33 112.60 120.80 129.70 143.00 1.01   400
intercept[12]  118.47 11.21  96.10 110.95 118.05 126.32 140.30 1.01   400
intercept[13]  111.10 10.43  91.30 103.30 111.55 118.90 130.70 1.00   400
intercept[14]  107.14  9.79  88.92  99.42 106.90 114.12 125.81 1.00   400
intercept[15]  117.77 11.05  97.07 109.05 118.05 125.42 137.45 1.01   400
intercept[16]  112.36  9.94  93.40 105.60 111.90 119.92 130.50 1.00   400
intercept[17]  113.80 10.63  94.24 105.87 114.10 120.52 134.40 1.00   400
intercept[18]  112.95 10.43  92.58 105.72 112.70 119.85 132.31 1.00   400
intercept[19]  116.77 10.96  95.34 109.20 117.10 124.90 137.01 1.01   400
b               -0.38  0.12  -0.60  -0.47  -0.38  -0.29  -0.14 1.01   400
mu.a            82.26  1.37  79.77  81.36  82.16  83.19  84.90 1.00   400
sigma.y          6.51  0.63   5.53   6.06   6.43   6.91   7.84 1.00   400
sigma.a          5.14  1.65   1.98   4.00   5.08   6.17   8.38 1.04   190
deviance       498.55  9.39 483.89 492.10 497.10 503.90 520.63 1.01   150
```

结果比较：贝叶斯方法借力、收缩



贝叶斯方法：借力，收缩（红色：贝叶斯，虚线：线性回归）

Outline

Outline

4. 多层模型的预测
   - 传统模型的预测
   - 贝叶斯模型的预测

传统回归模型的预测：回顾

假定有新的观察值向量 $\tilde{X}$，计算预测因子 $\tilde{X}\beta$，然后模拟预测值：

- 线性回归模型
  - 运用 R 的函数 prediction():

```
x.new<-data.frame()
predict(model,x.new,interval="prediction",level=0.95)
```

  - 随机模拟：模拟误差项 $\tilde{\varepsilon} \sim N(0, \sigma^2)$，然后计算 $\tilde{y} = \tilde{X}\beta + \tilde{\varepsilon}$
- Logistic 回归：对新数据点 i，计算

$$\Pr(\tilde{y}_i) = \text{logit}^{-1}(\tilde{X}\beta)$$

- Poisson 回归：模型 $y_1, y_2, \ldots, y_n \sim \text{Poisson}(u\lambda)$，对新观察值及指定 exposures $\tilde{u}_i$，模拟

$$\tilde{y}_i = \text{Poisson}\left(\tilde{u}_i e^{\tilde{X}_i\beta}\right)$$

多层模型：预测已有群组的一个新观察值

已知第一位球员 Adam 在 2006 赛季的得分为 89.4 分，预测他下赛季的得分。
用变截距模型 (M1)：

```
NewFGtM1 <- 89.4
pred.new<-coef(M1)$Name[1,1]+coef(M1)$Name[1,2]*NewFGtM1
pred.new
[1] 81.41539
```

- 多层模型的预测标准差计算困难，难以计算预测区间
- lmer 中的 predict 函数没有预测标准差选项。
- 办法：随机模拟

预测区间：已有群组的一个新观察值

对第一位球员 $j = 1$，$\tilde{x} = 89.4$，给定所有参数条件下，

$$\tilde{y}|\theta \quad \sim \quad N(\alpha_1 + \beta\tilde{x}, \sigma_y^2)$$

```
> mu.hat <- pred.new
> sigma.y.hat <- summary(M1)$sigma
> n.sims <- 1000
> y.tilde <- rnorm (n.sims,mu.hat,sigma.y.hat)
> quantile (y.tilde, c(.25,.5,.75))
     25%      50%      75%
77.28298 81.59877 85.37019
```

预测一个新群组的一个新观察值

假设新来一位球员，他本赛季成绩是 $\tilde{x} = 89.4$，预测他下赛季的成绩。必须先模拟一个球员层：$\tilde{\alpha} \sim N(\alpha, \sigma_\alpha^2)$，然后 $\tilde{y}|\theta \sim N(\tilde{\alpha} + \beta\tilde{x}, \sigma_y^2)$。

```
> sigma.y.hat <- summary(M1)$sigma
> sigma.a.hat <- sigma.hat(M1)$sigma$Name
> mu.a <- fixef(M1)["(Intercept)"]
> n.sims <- 1000
> a.tilde <- rnorm (n.sims, mu.a, sigma.a.hat)
> NewFGtM1 <- 89.4
> mu.y <- a.tilde +  coef(M1)$Name[1,2]*NewFGtM1
> y.tilde <- rnorm (n.sims, mu.y, sigma.y.hat)
> quantile (y.tilde, c(.25,.5,.75))
     25%      50%      75%
73.99056 79.19958 83.94172
```

Outline

用 Bugs 预测

计算贝叶斯预测有两种方法：

1. Bugs：模型中增加新数据或群组

用 Bugs 预测

计算贝叶斯预测有两种方法：

1. Bugs：模型中增加新数据或群组
2. R：根据现有数据得出的模型模拟

方法 1：直接在 Bugs 中增加新数据或群组

预测已有群组的一个新观察值：第一位球员 Adam 在 2006 赛季的得分为 89.4 分，预测他下赛季的得分。

- 需预测的数据以 NA 给出，Bugs 会自动给出预测值。
- 只需扩展数据：第一位球员 $(J = 1)$，新数据 NewFGtM1=89.4

```
kicker.data <- list ("n", "J", "x", "y", "player")
data.save<-save("n","y","player","x","J",
                file="kicker.data")
n <- n + 1
y <- c (y, NA)
player <- c (player, 1)
NewFGtM1 <- 89.4
x <- c (x, NewFGtM1)
```

- 然后在模型代码文件中的加上：y.tilde <- y[n]
- 文件另存为：ch8kicker_c.pred.txt

模型代码及运行结果

```
inits <- function (){
  list(intercept=rnorm(J,80,10),b=rnorm(1),
  sigma.y=runif(1),sigma.a=runif(1))}
parameters <- c ("a", "b", "sigma.y", "sigma.a")
parameters <- c (parameters, "y.tilde")
kicker.pred1 <- bugs (kicker.data,
                      inits,parameters,
                      "ch8kicker_c.pred.txt",
                      n.iter=2000,
                      n.burnin=1000,
                      n.thin=5,
                      bugs.directory="C:/WinBUGS14/")
attach.bugs (kicker.pred1)
>quantile (y.tilde, c(.25, .75))
  25%   75%
76.82 86.14
```

预测新群组的一个新观察值

只需扩展数据：第 $(J+1)$ 位球员，新数据 NewFGtM1=89.4

```
n <- n + 1
y <- c (y, NA)
player <- c (player, J+1)
NewFGtM1 <- 89.4
x <- c (x, NewFGtM1)
J <- J + 1
```

模型代码与前面一样，结果如下：

```
> attach.bugs (kicker.pred2)
> quantile (y.tilde, c(.25, .75))
    25%     75%
73.7475 85.2800
```

方法 2：在 R 中利用 Bugs 模型的结果进行模拟

前面已经介绍用 lmer() 得到的模型进行模拟，Bugs 模型类似。

预测已有群组的一个新观察值：第一位球员 Adam 在 2006 赛季的得分为 89.4 分，预测他下赛季的得分。

$$\tilde{y}|\theta = N(\alpha_1 + \beta\tilde{x}, \sigma_y^2)$$

运行 Bugs 得到模型的输出 output 后，

```
attach.bugs(output)
NewFGtM1 <- 89.4
y.tilde <- rnorm (n.sims, a[,1] + b*NewFGtM1, sigma.y)
quantile (y.tilde, c(.25, .75))
> quantile (y.tilde, c(.25, .75))
     25%      75%
76.46642 85.75404
```

预测新群组的一个新观察值

为改善 MCMC 收敛性，Bugs 模型进行中心化，

$$\tilde{y}|\theta \sim N(\text{intercept} + \beta(\tilde{x} - \bar{x}), \sigma_y^2)$$

其中 $\tilde{\alpha} = \text{intercept} - \beta\bar{x}$, $\text{intercept} \sim N(\alpha, \sigma_\alpha^2)$，

```
intercept.tilde<-rnorm (n.sims, mu.a, sigma.a)
a.tilde <- intercept.tilde - b*mean(x[])
y.tilde <- rnorm (n.sims, a.tilde + b*NewFGtM1, sigma.y)
> quantile (y.tilde, c(.25, .75))
     25%      75%
73.97203 85.08360
```

Recap

1. 多层模型的概念
   - 一个错误建模的实例
   - 多层模型
   - 面板数据的固定效应与随机效应
   - 多层贝叶斯模型
2. 多层线性回归模型：运用 lm() 和 lmer()
3. 多层贝叶斯模型
4. 多层模型的预测
   - 传统模型的预测
   - 贝叶斯模型的预测