432 Class 20

Thomas E. Love, Ph.D.

2024-03-28

## Setup

knitr::opts\_chunk$set(comment=NA)  
options(width = 80)  
  
library(janitor)

Attaching package: 'janitor'

The following objects are masked from 'package:stats':  
  
 chisq.test, fisher.test

library(gt)  
library(lme4)

Loading required package: Matrix

library(arm)

Loading required package: MASS

arm (Version 1.13-1, built: 2022-8-25)

Working directory is D:/Teaching/432/2024/432-slides-2024

library(broom)  
library(broom.mixed)  
library(conflicted)  
library(tidyverse)

── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
✔ dplyr 1.1.4 ✔ readr 2.1.5  
✔ forcats 1.0.0 ✔ stringr 1.5.1  
✔ ggplot2 3.4.4 ✔ tibble 3.2.1  
✔ lubridate 1.9.3 ✔ tidyr 1.3.0  
✔ purrr 1.0.2

conflicts\_prefer(dplyr::select)

[conflicted] Will prefer dplyr::select over any other package.

theme\_set(theme\_bw())

## An Introduction to Working with Hierarchical Data

* In a moment, we’ll visit <http://mfviz.com/hierarchical-models/>.

There, we try to learn about nested (hierarchical) data on faculty salaries. For each subject (faculty member) in the data, we have information on their salary, department and years of experience.

## Faculty Salaries example

* outcome: faculty salary (in $)
* predictor: years of experience
* group: department (five levels: Informatics, English, Sociology, Biology, Statistics)

We expect that salary (and the relationship between salary and years of experience) may be different depending on department, and every subject is in exactly one department.

## Visual Explanation

We’ll visit <http://mfviz.com/hierarchical-models/> now to learn a bit about:

* Nested Data
* Linear Model on the Fixed Effects
* Adding Random Intercepts to the Fixed Effects Model
* Incorporating Random Slopes with a Constant Intercept
* Random Slope and Random Intercept

## Fitting Hierarchical Models in R

We’ll focus today on approaches using the lme4 package, which can be used both for linear mixed models and for generalized linear mixed models.

* There are many, many ways to do this.
* The Generalized Linear Mixed Models FAQ at <https://bbolker.github.io/mixedmodels-misc/glmmFAQ.html> describes lots of other options for fitting hierarchical models in R.

## How The Data Were Simulated

* From Github

# Parameters for generating faculty salary data  
departments <- c('sociology', 'biology', 'english',   
 'informatics', 'statistics')  
base.salaries <- c(40000, 50000, 60000, 70000, 80000)  
annual.raises <- c(2000, 500, 500, 1700, 500)  
faculty.per.dept <- 25  
total.faculty <- faculty.per.dept \* length(departments)

# Generate tibble of faculty and (random) years of experience  
set.seed(432)  
ids <- 1:total.faculty  
department <- rep(departments, faculty.per.dept)  
experience <- floor(runif(total.faculty, 0, 10))  
bases <- rep(base.salaries, faculty.per.dept) \*   
 runif(total.faculty, .9, 1.1) # noise  
raises <- rep(annual.raises, faculty.per.dept) \*   
 runif(total.faculty, .9, 1.1) # noise  
facsal <- tibble(ids, department, bases, experience, raises)  
# Generate salaries (base + experience \* raise)  
facsal <- facsal |>  
 mutate(salary = bases + experience \* raises,  
 department = factor(department))

## The facsal data

facsal

# A tibble: 125 × 6  
 ids department bases experience raises salary  
 <int> <fct> <dbl> <dbl> <dbl> <dbl>  
 1 1 sociology 39438. 2 1861. 43160.  
 2 2 biology 46046. 0 493. 46046.  
 3 3 english 63656. 9 458. 67776.  
 4 4 informatics 67330. 1 1573. 68903.  
 5 5 statistics 84116. 7 545. 87930.  
 6 6 sociology 41626. 9 1871. 58463.  
 7 7 biology 54687. 7 521. 58335.  
 8 8 english 64477. 2 468. 65413.  
 9 9 informatics 64456. 6 1722. 74787.  
10 10 statistics 87841. 9 548. 92774.  
# ℹ 115 more rows

## Linear Model (no grouping by department)

m0 <- lm(salary ~ experience, data = facsal)  
  
tidy(m0, conf.int = TRUE) |>   
 select(term, estimate, std.error,   
 conf.low, conf.high) |>  
 gt() |> fmt\_number(decimals = 2) |>   
 tab\_options(table.font.size = 20)

| term | estimate | std.error | conf.low | conf.high |
| --- | --- | --- | --- | --- |
| (Intercept) | 56,100.96 | 2,285.57 | 51,576.81 | 60,625.10 |
| experience | 1,772.01 | 412.96 | 954.58 | 2,589.44 |

## Linear Model Summary

glance(m0) |>  
 select(r.squared, adj.r.squared, sigma, AIC, BIC) |>  
 gt() |> fmt\_number(decimals = 3) |>   
 tab\_options(table.font.size = 20)

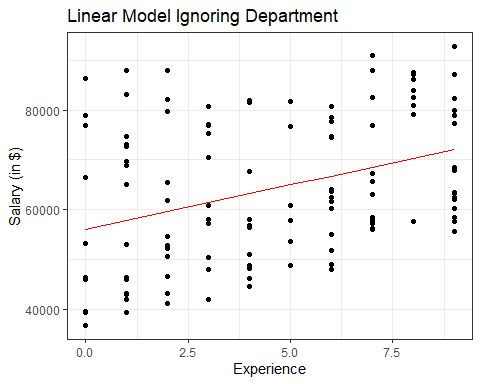
| r.squared | adj.r.squared | sigma | AIC | BIC |
| --- | --- | --- | --- | --- |
| 0.130 | 0.123 | 13,757.720 | 2,741.057 | 2,749.542 |

facsal$simple\_model\_preds <- predict(m0)  
  
head(predict(m0))

1 2 3 4 5 6   
59644.98 56100.96 72049.05 57872.97 68505.03 72049.05

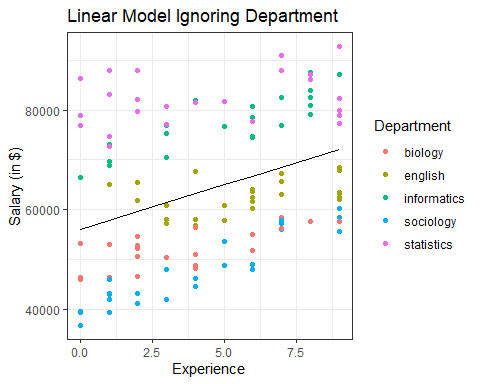
## Plotting the m0 predictions and the data

ggplot(data=facsal, aes(x=experience,   
 y=simple\_model\_preds)) +  
 geom\_line(col = "red") +   
 geom\_point(aes(x=experience, y=salary)) +  
 labs(x="Experience", y="Salary (in $)",  
 title = "Linear Model Ignoring Department")



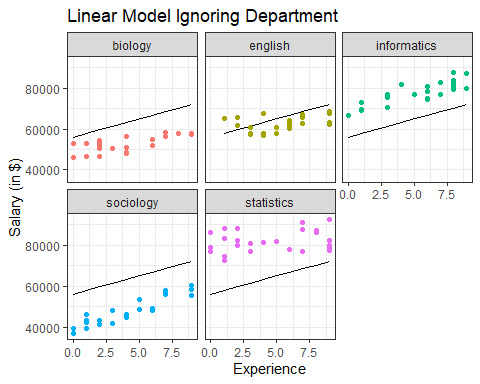
## m0 predictions with Department indicators

ggplot(data=facsal, aes(x=experience,   
 y=simple\_model\_preds)) +  
 geom\_line() +   
 geom\_point(aes(x=experience, y=salary,   
 group = department, colour = department)) +  
 labs(x="Experience",y="Salary (in $)",  
 title = "Linear Model Ignoring Department") +  
 scale\_color\_discrete('Department')



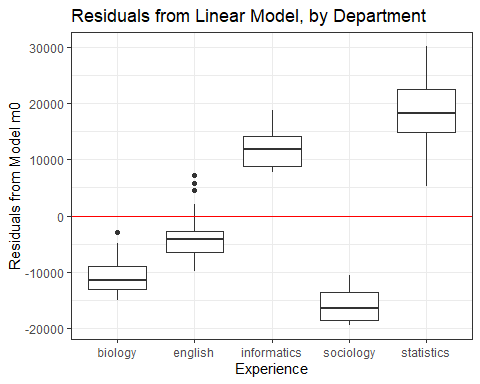
## m0 predictions and faceted results by Department

ggplot(data=facsal, aes(x=experience,   
 y=simple\_model\_preds)) +  
 geom\_line() +   
 geom\_point(aes(x=experience, y=salary,   
 group = department, colour = department)) +  
 labs(x="Experience",y="Salary (in $)",  
 title = "Linear Model Ignoring Department") +  
 guides(color = "none") +  
 scale\_color\_discrete('Department') +  
 facet\_wrap(~ department)



## Plot of m0 Residuals by Department

facsal <- facsal |>  
 mutate(simple\_model\_resids = salary - simple\_model\_preds)  
  
ggplot(data=facsal, aes(x=department,   
 y=simple\_model\_resids)) +  
 geom\_boxplot() +  
 geom\_hline(yintercept = 0, col = "red") +  
 labs(x="Experience", y="Residuals from Model m0",  
 title = "Residuals from Linear Model, by Department")



# Let the intercepts vary

## Model incorporating varying intercepts by department

m1 <- lmer(salary ~ experience + (1 | department),   
 data = facsal)  
  
m1

Linear mixed model fit by REML ['lmerMod']  
Formula: salary ~ experience + (1 | department)  
 Data: facsal  
REML criterion at convergence: 2428.544  
Random effects:  
 Groups Name Std.Dev.  
 department (Intercept) 14728   
 Residual 4069   
Number of obs: 125, groups: department, 5  
Fixed Effects:  
(Intercept) experience   
 59056 1138

## Tidied Coefficients

This is the *varying intercept* model.

tidy(m1, conf.int = TRUE) |>  
 select(-std.error, -statistic) |>  
 gt() |> fmt\_number(decimals = 1) |>  
 tab\_options(table.font.size = 20)

| effect | group | term | estimate | conf.low | conf.high |
| --- | --- | --- | --- | --- | --- |
| fixed | NA | (Intercept) | 59,056.3 | 46,075.5 | 72,037.2 |
| fixed | NA | experience | 1,138.4 | 890.1 | 1,386.6 |
| ran\_pars | department | sd\_\_(Intercept) | 14,728.0 | NA | NA |
| ran\_pars | Residual | sd\_\_Observation | 4,068.9 | NA | NA |

## Summarizing model m1

glance(m1) |>  
 select(sigma, AIC, BIC, logLik, df.residual) |>  
 gt() |> fmt\_number(decimals = 2) |>  
 tab\_options(table.font.size = 20)

| sigma | AIC | BIC | logLik | df.residual |
| --- | --- | --- | --- | --- |
| 4,068.93 | 2,436.54 | 2,447.86 | -1,214.27 | 121.00 |

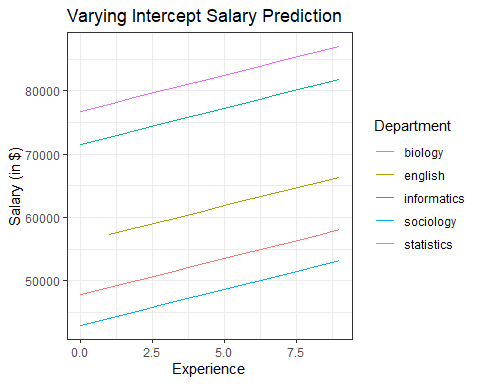
## Saving the Model m1 predictions

facsal$random\_intercept\_preds <- predict(m1)  
  
head(predict(m1))

1 2 3 4 5 6   
45226.95 47815.47 66405.51 72718.80 84743.65 53195.42

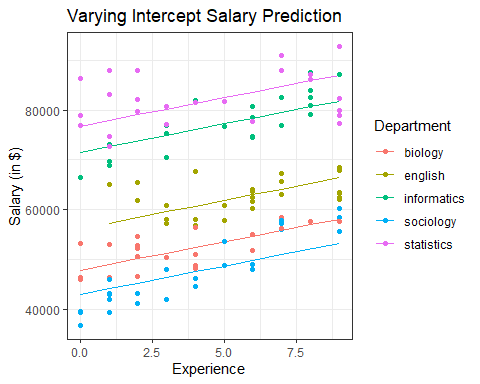
## Plotting the m1 predictions without the data

ggplot(data=facsal, aes(x=experience,   
 y=random\_intercept\_preds,   
 group = department,   
 col = department)) +  
 geom\_line() +   
 labs(x="Experience",y="Salary (in $)",  
 title = "Varying Intercept Salary Prediction") +  
 scale\_color\_discrete('Department')



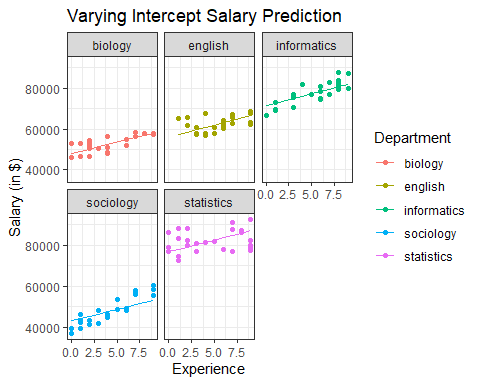
## Plotting the m1 predictions and the data

ggplot(data=facsal, aes(x=experience,   
 y=random\_intercept\_preds,   
 group = department,   
 col = department)) +  
 geom\_line() +   
 geom\_point(aes(x=experience, y=salary,   
 group = department, colour = department)) +  
 labs(x="Experience",y="Salary (in $)",  
 title = "Varying Intercept Salary Prediction") +  
 scale\_color\_discrete('Department')



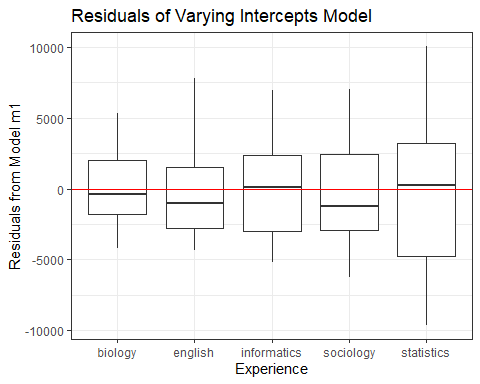
## m1 predictions and the data, faceted by Department

ggplot(data=facsal, aes(x=experience,   
 y=random\_intercept\_preds,   
 group = department,   
 col = department)) +  
 geom\_line() +   
 geom\_point(aes(x=experience, y=salary,   
 group = department, colour = department)) +  
 labs(x="Experience",y="Salary (in $)",  
 title = "Varying Intercept Salary Prediction") +  
 scale\_color\_discrete('Department') +  
 facet\_wrap(~ department)



## Plot of m1 Residuals by Department

facsal <- facsal |>  
 mutate(random\_intercept\_resids =   
 salary - random\_intercept\_preds)  
  
ggplot(data=facsal, aes(x=department,   
 y=random\_intercept\_resids)) +  
 geom\_boxplot() +  
 geom\_hline(yintercept = 0, col = "red") +  
 labs(x="Experience", y="Residuals from Model m1",  
 title = "Residuals of Varying Intercepts Model")



# Let the slopes vary

## Model incorporating varying slopes by department

m2 <- lmer(salary ~ experience +   
 (0 + experience | department),   
 data = facsal)

## Varying Slopes Model

m2

Linear mixed model fit by REML ['lmerMod']  
Formula: salary ~ experience + (0 + experience | department)  
 Data: facsal  
REML criterion at convergence: 2626.859  
Random effects:  
 Groups Name Std.Dev.  
 department experience 2082   
 Residual 9438   
Number of obs: 125, groups: department, 5  
Fixed Effects:  
(Intercept) experience   
 57705 1249

## Tidied m2 Coefficients

tidy(m2, conf.int = TRUE) |>  
 select(-std.error, -statistic) |>  
 gt() |> fmt\_number(decimals = 1) |>   
 tab\_options(table.font.size = 20)

| effect | group | term | estimate | conf.low | conf.high |
| --- | --- | --- | --- | --- | --- |
| fixed | NA | (Intercept) | 57,704.7 | 54,616.9 | 60,792.5 |
| fixed | NA | experience | 1,249.3 | -661.1 | 3,159.6 |
| ran\_pars | department | sd\_\_experience | 2,081.6 | NA | NA |
| ran\_pars | Residual | sd\_\_Observation | 9,437.9 | NA | NA |

## Summarizing model m2

glance(m2) |>  
 select(sigma, AIC, BIC, logLik, df.residual) |>  
 gt() |> fmt\_number(decimals = 2) |>   
 tab\_options(table.font.size = 20)

| sigma | AIC | BIC | logLik | df.residual |
| --- | --- | --- | --- | --- |
| 9,437.90 | 2,634.86 | 2,646.17 | -1,313.43 | 121.00 |

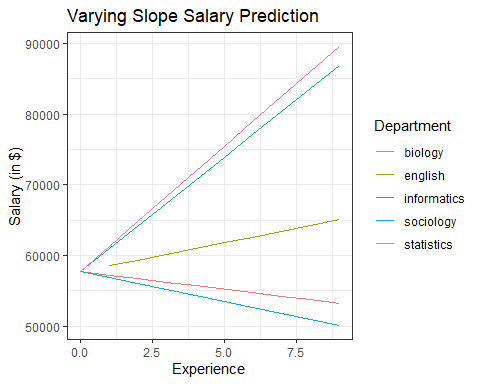
## Saving the Model m2 predictions

facsal$random\_slope\_preds <- predict(m2)  
  
head(predict(m2))

1 2 3 4 5 6   
56009.39 57704.71 65027.81 60941.97 82501.68 50075.79

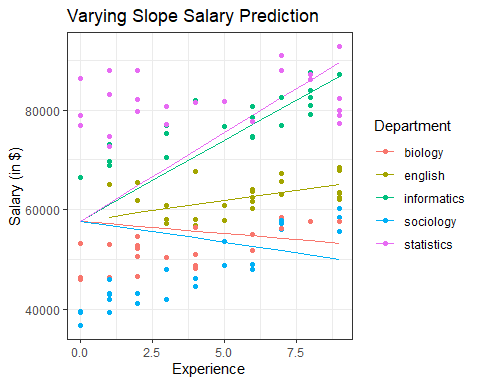
## Plotting the m2 predictions without the data

ggplot(data=facsal, aes(x=experience,   
 y=random\_slope\_preds,   
 group = department,   
 col = department)) +  
 geom\_line() +   
 labs(x="Experience",y="Salary (in $)",  
 title = "Varying Slope Salary Prediction") +  
 scale\_color\_discrete('Department')



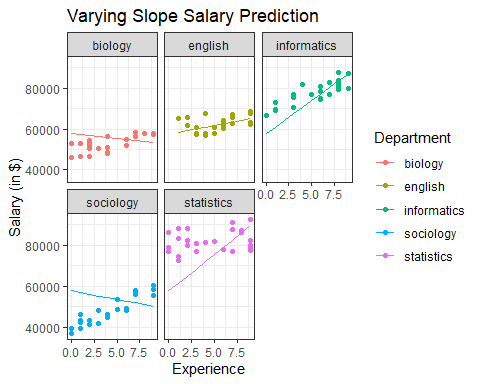
## Plotting the m2 predictions and the data

ggplot(data=facsal, aes(x=experience,   
 y=random\_slope\_preds,   
 group = department,   
 col = department)) +  
 geom\_line() +   
 geom\_point(aes(x=experience, y=salary,   
 group = department, colour = department)) +  
 labs(x="Experience",y="Salary (in $)",  
 title = "Varying Slope Salary Prediction") +  
 scale\_color\_discrete('Department')



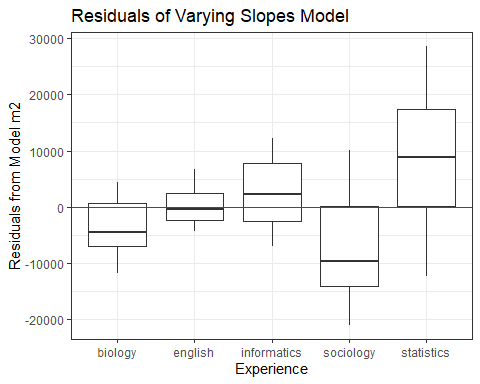
## m2 predictions and the data, faceted by Department

ggplot(data=facsal, aes(x=experience,   
 y=random\_slope\_preds,   
 group = department,   
 col = department)) +  
 geom\_line() +   
 geom\_point(aes(x=experience, y=salary,   
 group = department, colour = department)) +  
 labs(x="Experience",y="Salary (in $)",  
 title = "Varying Slope Salary Prediction") +  
 scale\_color\_discrete('Department') +  
 facet\_wrap(~ department)



## Plot of m2 Residuals by Department

facsal <- facsal |>  
 mutate(random\_slope\_resids =   
 salary - random\_slope\_preds)  
  
ggplot(data=facsal, aes(x=department,   
 y=random\_slope\_resids)) +  
 geom\_boxplot() +  
 geom\_hline(yintercept = 0, col = "red") +  
 labs(x="Experience", y="Residuals from Model m2",  
 title = "Residuals of Varying Slopes Model")



# Let the slopes and intercepts vary

## Model with varying slopes and intercept by department

m3 <- lmer(salary ~ experience +   
 (1 + experience | department),   
 data = facsal)

## Varying Slopes and Intercepts Model

m3

Linear mixed model fit by REML ['lmerMod']  
Formula: salary ~ experience + (1 + experience | department)  
 Data: facsal  
REML criterion at convergence: 2405.105  
Random effects:  
 Groups Name Std.Dev. Corr   
 department (Intercept) 16320.1   
 experience 722.4 -0.64  
 Residual 3569.5   
Number of obs: 125, groups: department, 5  
Fixed Effects:  
(Intercept) experience   
 59083 1165

## Tidied m3 Coefficients

tidy(m3) |>  
 gt() |> fmt\_number(decimals = 1) |>   
 tab\_options(table.font.size = 20)

| effect | group | term | estimate | std.error | statistic |
| --- | --- | --- | --- | --- | --- |
| fixed | NA | (Intercept) | 59,083.0 | 7,325.8 | 8.1 |
| fixed | NA | experience | 1,164.9 | 342.3 | 3.4 |
| ran\_pars | department | sd\_\_(Intercept) | 16,320.1 | NA | NA |
| ran\_pars | department | cor\_\_(Intercept).experience | -0.6 | NA | NA |
| ran\_pars | department | sd\_\_experience | 722.4 | NA | NA |
| ran\_pars | Residual | sd\_\_Observation | 3,569.5 | NA | NA |

## Summarizing model m3

glance(m3) |>  
 select(sigma, AIC, BIC, logLik, df.residual) |>  
 gt() |> fmt\_number(decimals = 2) |>   
 tab\_options(table.font.size = 20)

| sigma | AIC | BIC | logLik | df.residual |
| --- | --- | --- | --- | --- |
| 3,569.50 | 2,417.11 | 2,434.08 | -1,202.55 | 119.00 |

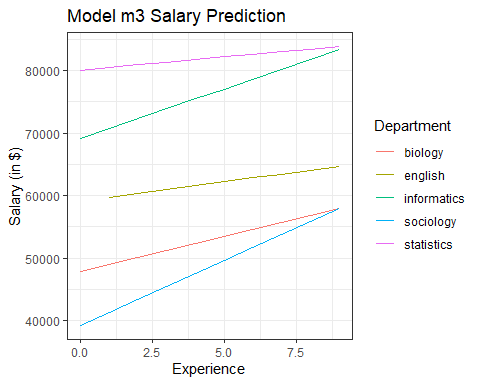
## Saving the Model m3 predictions

facsal$random\_slope\_int\_preds <- predict(m3)  
  
head(predict(m3))

1 2 3 4 5 6   
43426.37 47863.80 64685.23 70714.44 82974.75 57995.15

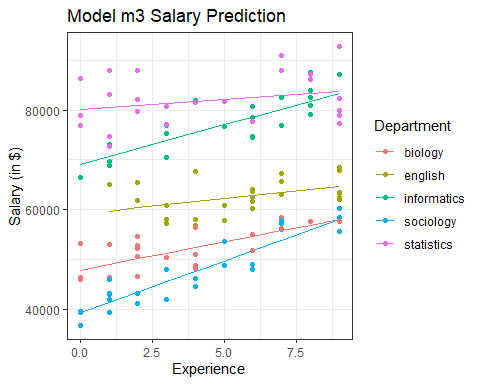
## Plotting the m3 predictions without the data

ggplot(data=facsal, aes(x=experience,   
 y=random\_slope\_int\_preds,   
 group = department,   
 col = department)) +  
 geom\_line() +   
 labs(x="Experience",y="Salary (in $)",  
 title = "Model m3 Salary Prediction") +  
 scale\_color\_discrete('Department')



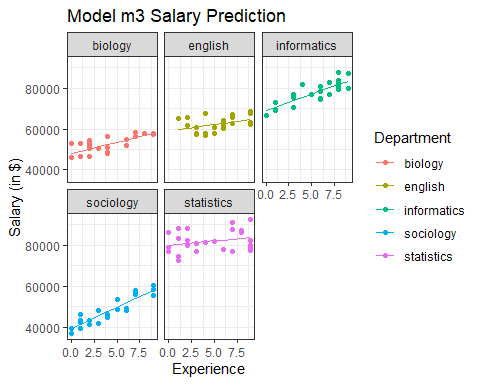
## Plotting the m3 predictions and the data

ggplot(data=facsal, aes(x=experience,   
 y=random\_slope\_int\_preds,   
 group = department,   
 col = department)) +  
 geom\_line() +   
 geom\_point(aes(x=experience, y=salary,   
 group = department, colour = department)) +  
 labs(x="Experience",y="Salary (in $)",  
 title = "Model m3 Salary Prediction") +  
 scale\_color\_discrete('Department')



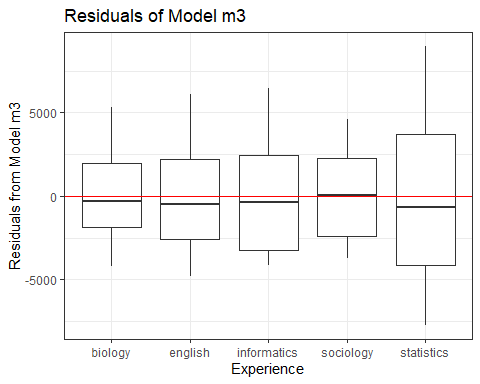
## m3 predictions and the data, faceted by Department

ggplot(data=facsal, aes(x=experience,   
 y=random\_slope\_int\_preds,   
 group = department,   
 col = department)) +  
 geom\_line() +   
 geom\_point(aes(x=experience, y=salary,   
 group = department, colour = department)) +  
 labs(x="Experience",y="Salary (in $)",  
 title = "Model m3 Salary Prediction") +  
 scale\_color\_discrete('Department') +  
 facet\_wrap(~ department)



## Plot of m3 Residuals by Department

facsal <- facsal |>  
 mutate(random\_slope\_int\_resids =   
 salary - random\_slope\_int\_preds)  
  
ggplot(data=facsal, aes(x=department,   
 y=random\_slope\_int\_resids)) +  
 geom\_boxplot() +  
 geom\_hline(yintercept = 0, col = "red") +  
 labs(x="Experience", y="Residuals from Model m3",  
 title = "Residuals of Model m3")



## Comparing the Models

AIC(m0, m1, m2, m3)

df AIC  
m0 3 2741.057  
m1 4 2436.544  
m2 4 2634.859  
m3 6 2417.105

BIC(m0, m1, m2, m3)

df BIC  
m0 3 2749.542  
m1 4 2447.857  
m2 4 2646.172  
m3 6 2434.075

## Can we test for an effect of experience?

Let’s refit model m3 and compare it to an appropriate null model (without the experience information), using an anova driven likelihood ratio test.

m3 <- lmer(salary ~ experience +   
 (1 + experience | department),   
 data = facsal, REML = FALSE)  
  
m\_null <- lmer(salary ~ (1 | department),  
 data = facsal, REML = FALSE)

The REML = FALSE lets us get the likelihood ratio test we want.

## Likelihood Ratio Test comparing m3 to m\_null

anova(m\_null, m3)

Data: facsal  
Models:  
m\_null: salary ~ (1 | department)  
m3: salary ~ experience + (1 + experience | department)  
 npar AIC BIC logLik deviance Chisq Df Pr(>Chisq)   
m\_null 3 2527.7 2536.2 -1260.9 2521.7   
m3 6 2449.5 2466.5 -1218.8 2437.5 84.196 3 < 2.2e-16 \*\*\*  
---  
Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

## Tidied coefficients from m3

tidy(m3, conf.int = TRUE) |>  
 select(-std.error, -statistic) |>  
 gt() |> fmt\_number(decimals = 1) |>   
 tab\_options(table.font.size = 20)

| effect | group | term | estimate | conf.low | conf.high |
| --- | --- | --- | --- | --- | --- |
| fixed | NA | (Intercept) | 59,078.1 | 46,212.3 | 71,943.9 |
| fixed | NA | experience | 1,165.7 | 565.6 | 1,765.9 |
| ran\_pars | department | sd\_\_(Intercept) | 14,610.5 | NA | NA |
| ran\_pars | department | cor\_\_(Intercept).experience | -0.6 | NA | NA |
| ran\_pars | department | sd\_\_experience | 636.3 | NA | NA |
| ran\_pars | Residual | sd\_\_Observation | 3,569.5 | NA | NA |

## Parametric Bootstrap test for department effect (Part 1)

nBoot=100 # should probably be 1000 at a minimum  
lrStat=rep(NA,nBoot)  
# first fit appropriate null and alternate models  
ft.null <- lm(salary ~ experience, data = facsal) #null model  
ft.alt <- lmer(salary ~ experience + (1 | department),  
 data=facsal, REML=F) # alternate model  
# calculate observed test statistic (deviance = -2 \* loglik)  
lrObs <- 2\*logLik(ft.alt) - 2\*logLik(ft.null) # test stat

## Parametric Bootstrap test for department effect (Part 2)

set.seed(432)  
for(iBoot in 1:nBoot)  
{  
 facsal$SalSim=unlist(simulate(ft.null)) #resampled data  
 # calculate results for our two models in resampled data  
 bNull <- lm(SalSim ~ experience,   
 data=facsal) #null model  
 bAlt <- lmer(SalSim ~ experience + (1|department),  
 data=facsal, REML=F) # alternate model  
 # calculate and store resampled test stat  
 lrStat[iBoot] <- 2\*logLik(bAlt) - 2\*logLik(bNull)   
}

boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')  
boundary (singular) fit: see help('isSingular')

## Parametric Bootstrap Test for Department effect (Part 3)

mean(lrStat>lrObs) # P-value for test of department effect

[1] 0

### Even this “simple” model might not be simple.

Our parametric bootstrap can hit up on the edge of a problem with the random effects.

boundary (singular) fit: see ?isSingular

is a common warning we might see, for instance.

## What is a Mixed Model?

A model for an outcome that incorporates both fixed and random effects.

Or, alternatively,…

Mixed models are those with a mixture of fixed and random effects. Random effects are categorical factors where the levels have been selected from many possible levels and the investigator would like to make inferences beyond just the levels chosen.

* From <http://environmentalcomputing.net/mixed-models/>

## A Random Effect?

A random factor:

* is categorical
* has a large number of levels
* only a subsample (often a random subsample) of levels is included in your design
* you want to make inference in general, and not only for the levels you observed

## A Random Factor?

Think of a random factor as a group where:

* you want to quantify variation between group levels
* you want to make predictions about unobserved groups
* but you don’t want to compare outcome differences between particular group levels

Sources: <https://bbolker.github.io/morelia_2018/notes/glmm.html> and <http://environmentalcomputing.net/mixed-models-1/>

## Why Use a Random Effect?

* You want to combine information across groups
* You have variation in information per group level (number of samples or amount of noisiness)
* You have a categorical predictor that is a nuisance variable (something not of direct interest but that we want to control for)
* You have more than 5-6 groups

Source: Crawley (2002) and Gelman (2005) quoted at <https://bbolker.github.io/morelia_2018/notes/glmm.html>

## What is a Fixed Effect vs. a Random Effect?

The one I most often use is something like:

* Fixed effects are constant across individuals, while random effects vary.

The various definitions in the literature are incompatible with each other[[1]](#footnote-126).

## Problems with our definitions

From Scahabenberger and Pierce (2001), we have this gem:

One modeler’s random effect is another modeler’s fixed effect.

A more practical definition might be to ask the question posed by Crawley (2002):

Are there enough levels of the factor in the data on which to base an estimate of the variance of the population of effects? No, means [you should probably treat the variable as] fixed effects.

## Models We Might Consider

Suppose we have an outcome y, predictor x and group group

* y ~ x = linear regression on x: not a mixed model
* y ~ 1 + (1 | group) = random intercept on group: null model
* y ~ x + (1 | group) = fixed slope and random intercept
* y ~ (0 + x | group) = random slope of x within group, no variation in intercept
* y ~ x + (x | group) = random intercept and random slope

## A “More” Realistic Example

The most common example in modern medicine has measurements nested within people. Repeated measures and longitudinal data provide typical settings for this sort of approach.

Another setting where a hierarchical approach is of interest occurs when you have variables measured at multiple levels, for instance you have information on patients, who are nested within providers, who are nested within hospitals.

## Today was just one example

Nothing of what I’ve talked about today should be taken as the final word on how to extend these ideas beyond the very simple example I’ve provided this afternoon.

### What’s Next?

Survival (time to event) regression models, specifically Cox proportional hazards models.

1. See, for instance, the GLMM FAQ referenced earlier [↑](#footnote-ref-126)