THE UNIVERSITY *of* EDINBURGH
Centre for Data, Culture & Society

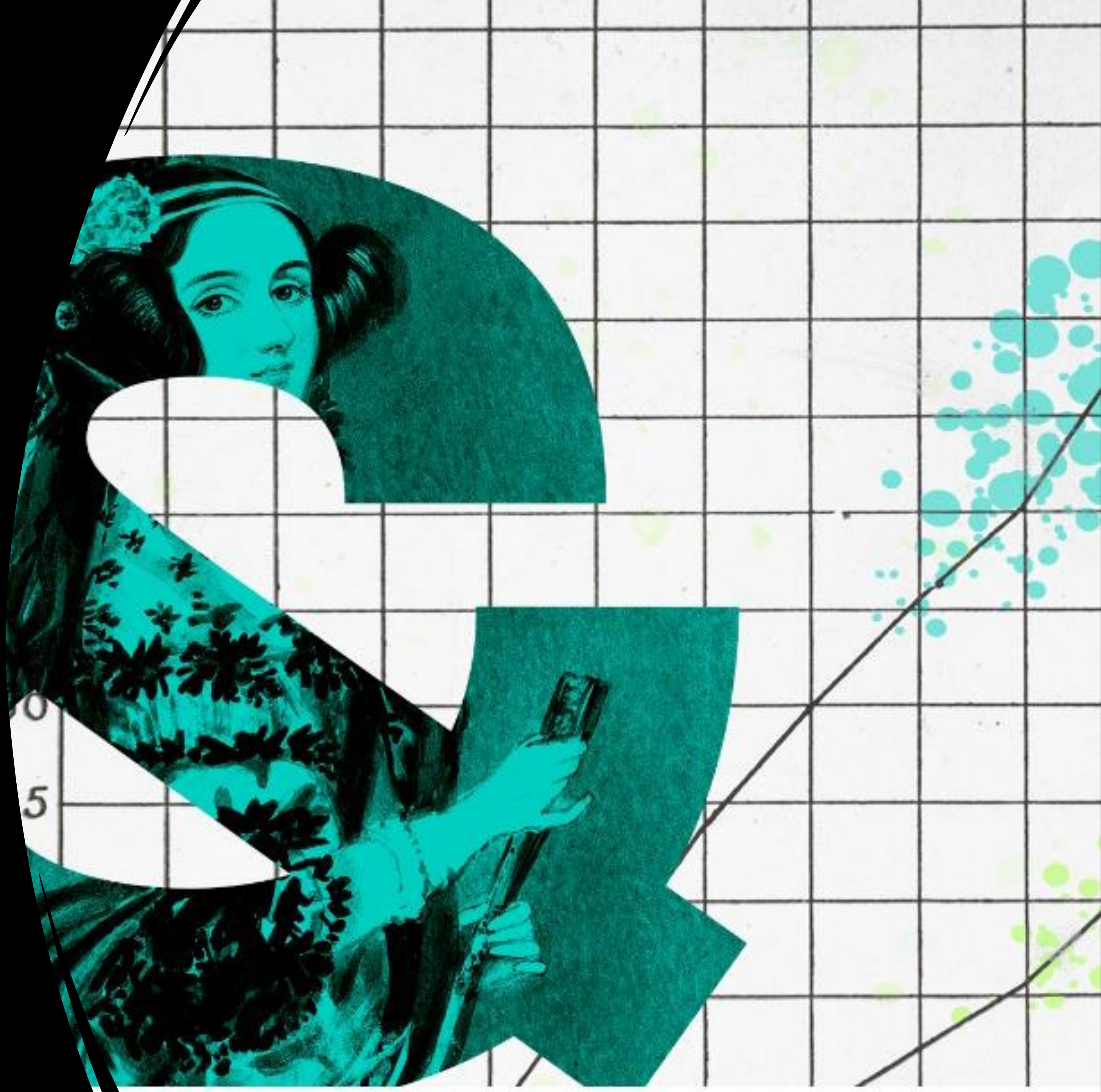
Regression & Mixed-Effects Modelling with R

Fang Jackson-Yang & Aislinn Keogh

29 April 2025

Course outline

- Session 1. Simple Regression; Individual Difference; Intro to Linear mixed-effect models (LMMs)
- Session 2. LMMs (lmer); Generalised LMMs (glmer)
- Session 3. Practical
- Session 4. Ggeneralised LMMs continued; Model assumptions and diagnostics
- Session 5. Practical



Session 1
Roadmap (today)

- 1. Get to know the data

- 2. Simple regression models with continuous outcome – lm()

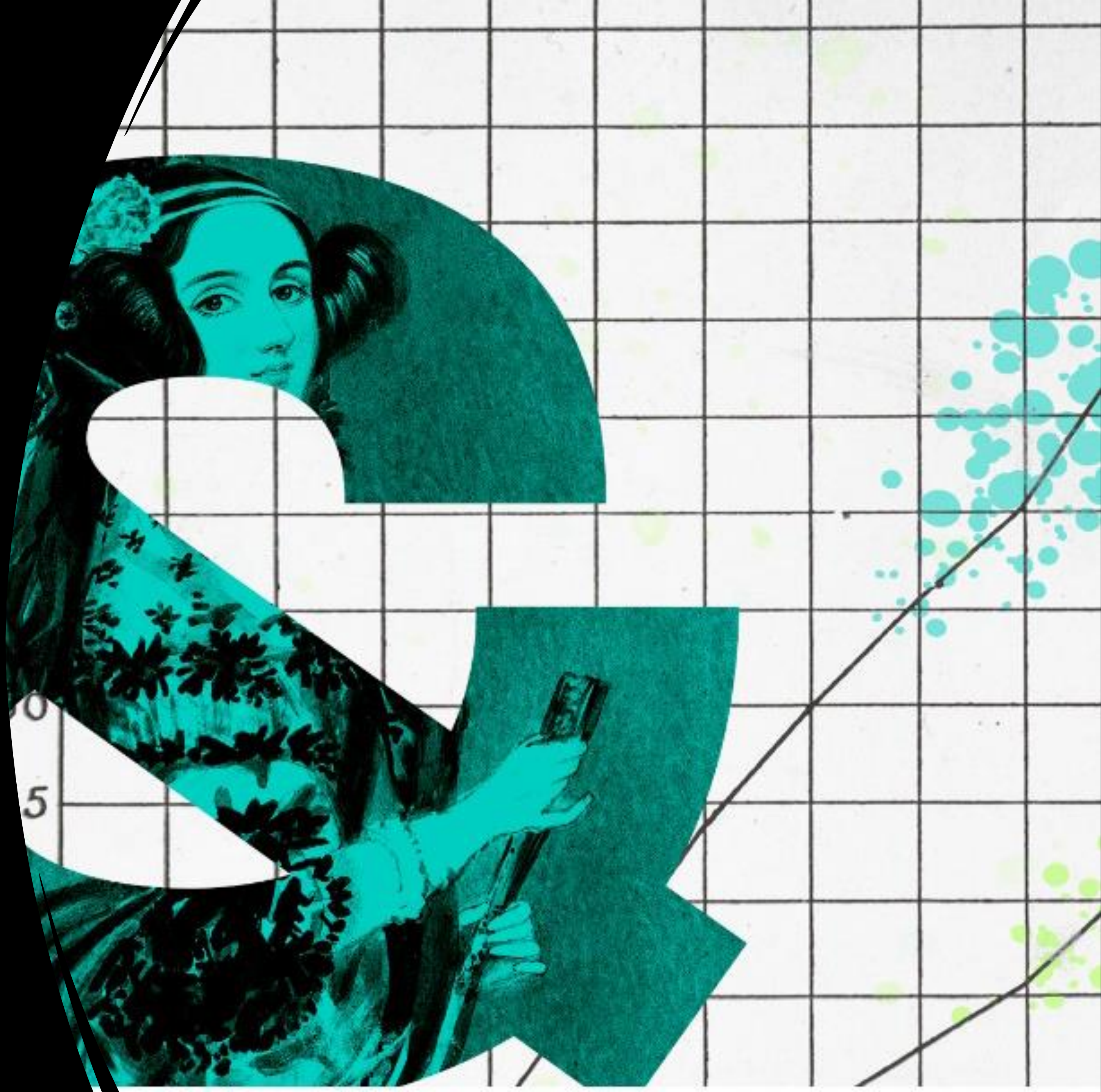
    2.1 One predictor

    2.2 Two predictors with interaction

- 3. Dealing with individual difference

    3.1 Alternative

    3.2 LMMs



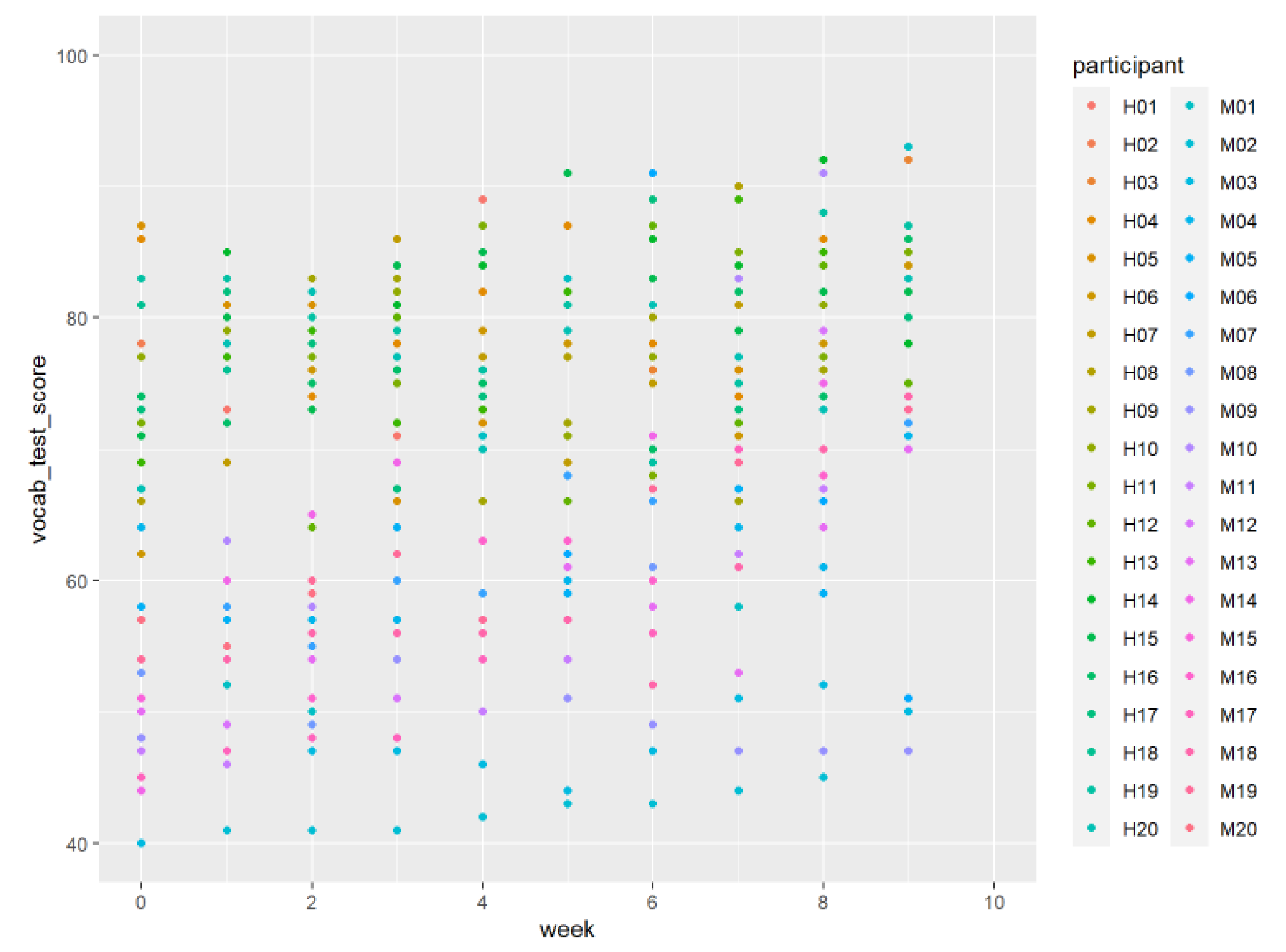
Simple Regression

Regression in Action

- Simulated data based on a real case
- A secondary school in Glasgow were considering purchasing a new online APP for teaching vocabulary in language classes.
- They tested the effectiveness of the course on 40 students over 10 weeks.
- How effective was the course? Worthy investing or not?

Check Our Data

```
# A tibble: 6 x 4
  participant proficiency   week vocab_test_score
  <chr>       <chr>        <dbl>            <dbl>
1 M01         intermediate     0               50
2 M01         intermediate     1               52
3 M01         intermediate     2               50
4 M01         intermediate     3               64
5 M01         intermediate     4               71
6 M01         intermediate     5               83
```

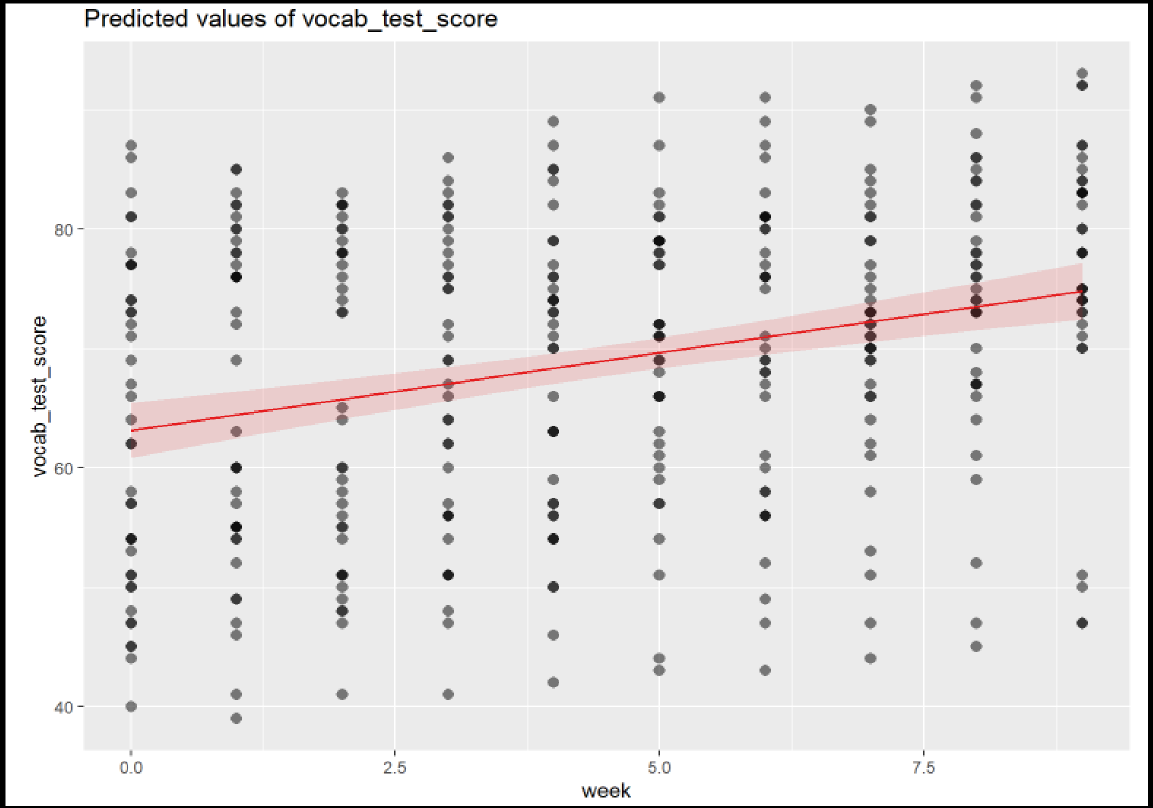


2.1 Simple Regression: One Predictor

- Simple regression with one predictor

```
m1 <- lm(vocab_test_score ~ week, data = vocabdata)
```

```
## Call:
## lm(formula = vocab_test_score ~ week, data = vocabdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -28.5202  -9.4438   0.7764  10.0731  23.8528
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  63.1472     1.1765  53.673  < 2e-16 ***
## week          1.2966     0.2229   5.818  1.3e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.3 on 367 degrees of freedom
##   (31 observations deleted due to missingness)
## Multiple R-squared:  0.08445,    Adjusted R-squared:  0.08195
## F-statistic: 33.85 on 1 and 367 DF,  p-value: 1.297e-08
```



2.1 Simple Regression: One Predictor

Intercept: the expected value of y when x = 0.

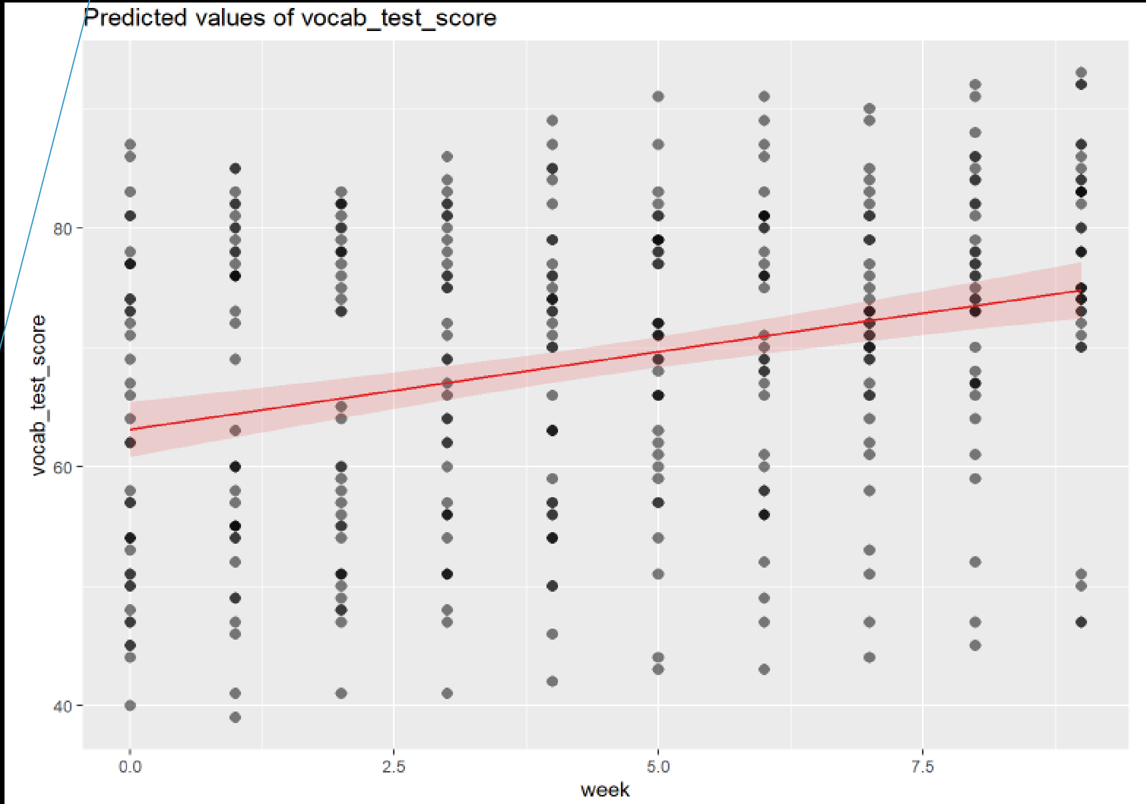
Slope: estimate for the number of units of increases in Y on average, as x increases by one unit

Overall quality of the model

- Simple regression with one predictor

```
m1 <- lm(vocab_test_score ~ week, data = vocabdata)
```

```
## Call:
## lm(formula = vocab_test_score ~ week, data = vocabdata)
##
## Residuals:
##       Min       1Q   Median       3Q      Max
## -28.5202  -9.4438   0.7764  10.0731  23.8528
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  63.1472     1.1765  53.673  < 2e-16 ***
## week          1.2966     0.2229   5.818  1.3e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.3 on 367 degrees of freedom
##   (31 observations deleted due to missingness)
## Multiple R-squared:  0.08445,	Adjusted R-squared:  0.08195
## F-statistic: 33.85 on 1 and 367 DF,  p-value: 1.297e-08
```

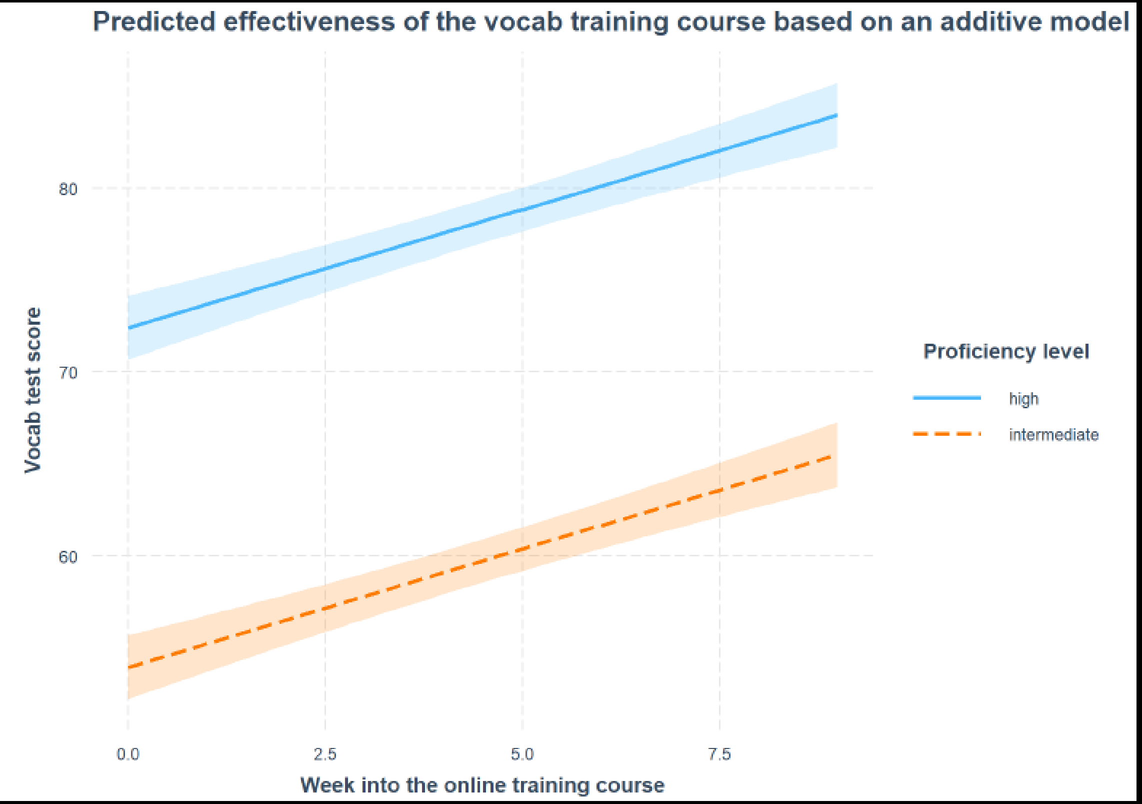


2.2(a). Simple Regression: Two Predictors

- **Additive** model

```
m2a <- lm(vocab_test_score ~ week + proficiency, data = vocabdata)
```

```
## Call:
## lm(formula = vocab_test_score ~ week + proficiency, data = vocabdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.2192  -5.4079  -0.1221   5.0211  29.3524
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)              72.4073     0.8835  81.951   <2e-16 ***
## week                      1.2858     0.1469   8.752   <2e-16 ***
## proficiencyintermediate -18.4745     0.8445 -21.876   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.111 on 366 degrees of freedom
##   (31 observations deleted due to missingness)
## Multiple R-squared:  0.6032, Adjusted R-squared:  0.6011
## F-statistic: 278.2 on 2 and 366 DF,  p-value: < 2.2e-16
```



Intercept: the expected value of y when X1 = 0, and X2 is at its reference category.

2.2(a). Simple Regression: Two Predictors

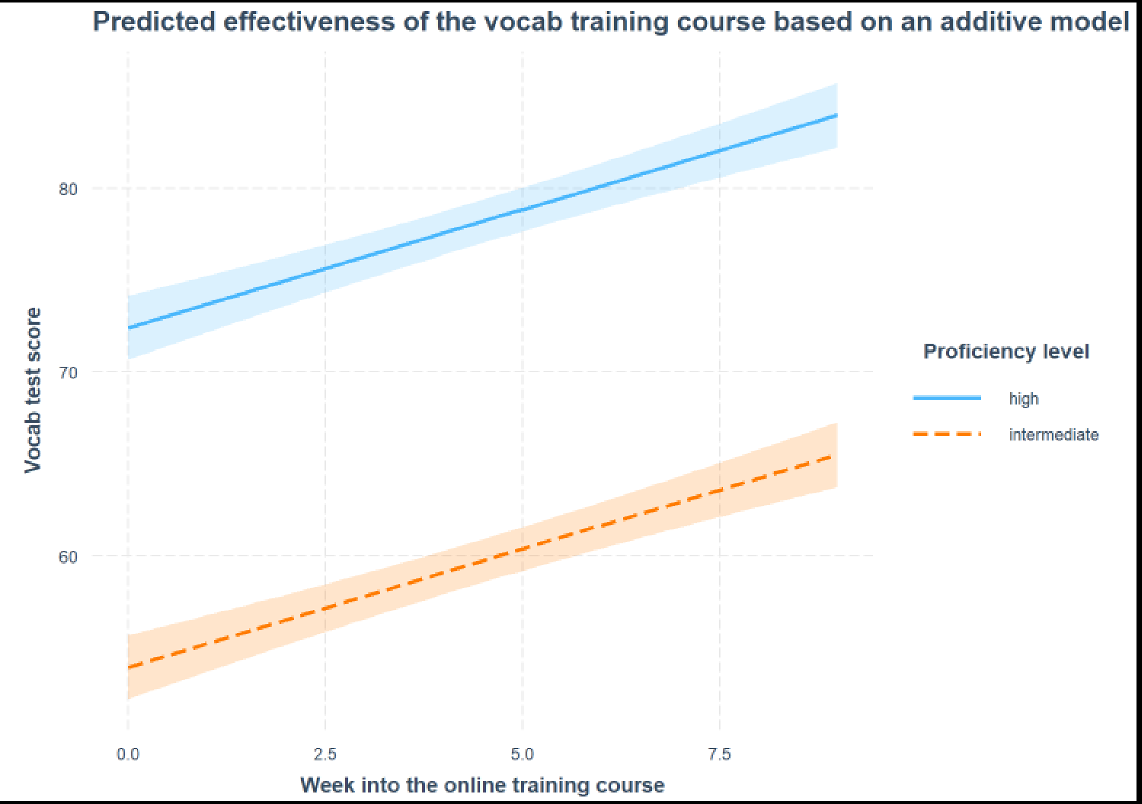
Slope: expected change of y on average as X1 increases by one unit.

Slope: expected change of y when X1=0 and X2 changes from its reference category to another category.

- **Additive** model

```
m2a <- lm(vocab_test_score ~ week + proficiency, data = vocabdata)
```

```
## Call:
## lm(formula = vocab_test_score ~ week + proficiency, data = vocabdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.2192  -5.4079  -0.1221   5.0211  29.3524
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)             72.4073     0.8835  81.951   <2e-16 ***
## week                     1.2858     0.1469   8.752   <2e-16 ***
## proficiencyintermediate -18.4745    0.8445 -21.876   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.111 on 366 degrees of freedom
##   (31 observations deleted due to missingness)
## Multiple R-squared:  0.6032, Adjusted R-squared:  0.6011
## F-statistic: 278.2 on 2 and 366 DF,  p-value: < 2.2e-16
```



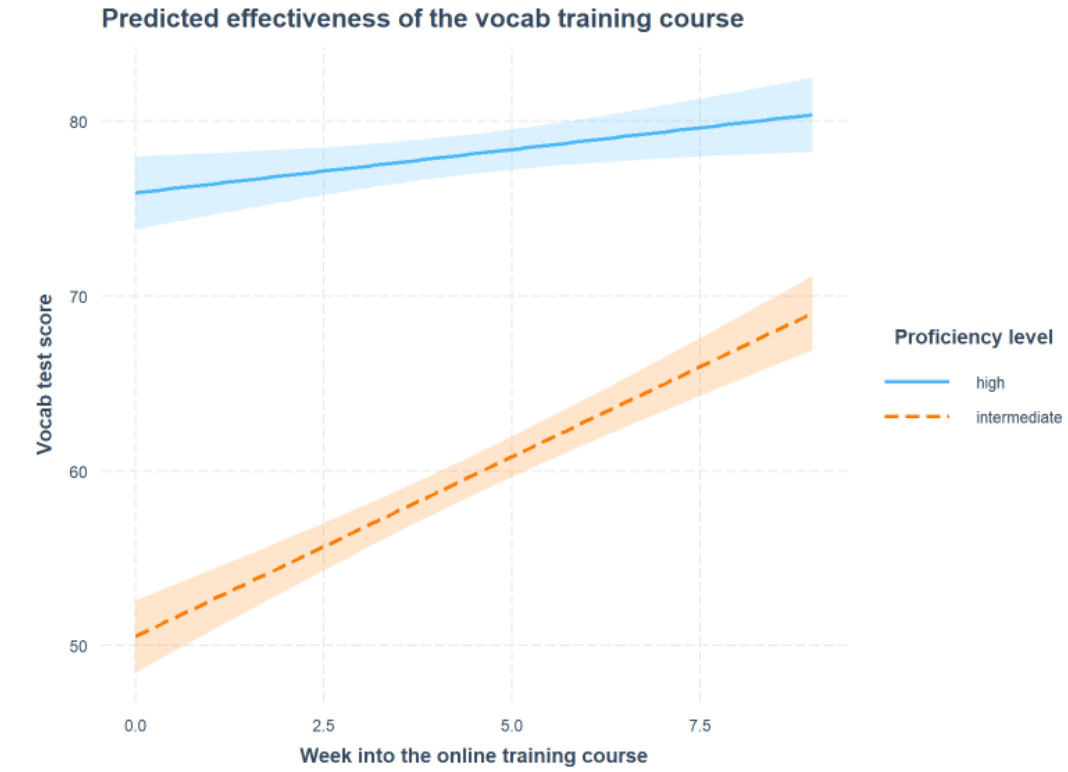
2.2(b). Simple Regression – Two Predictors

- **Interactive** model

```
# version 1
m2bv1 <- lm(vocab_test_score ~ week + proficiency + week:proficiency, data = vocabdata)
# version 2
m2bv2 <- lm(vocab_test_score ~ week * proficiency, data = vocabdata)
```

→ identical

```
## Call:
## lm(formula = vocab_test_score ~ week * proficiency, data = vocabdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.0330  -4.6421   0.5941   4.6204  28.1345
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   75.9092     1.0610  71.545  < 2e-16 ***
## week                           0.4967     0.2011   2.470    0.014 *
## proficiencyintermediate      -25.3788     1.4924 -17.005  < 2e-16 ***
## week:proficiencyintermediate   1.5591     0.2827   5.515 6.61e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.803 on 365 degrees of freedom
##   (31 observations deleted due to missingness)
## Multiple R-squared:  0.6338, Adjusted R-squared:  0.6308
## F-statistic: 210.5 on 3 and 365 DF,  p-value: < 2.2e-16
```



Intercept: the expected value of y when X1 = 0, and X2 is at its reference category.

Slope: expected change of y on average as X1 increases by one unit, when X2 is in its reference category.

2.2(b). Simple Regression – Two Predictors

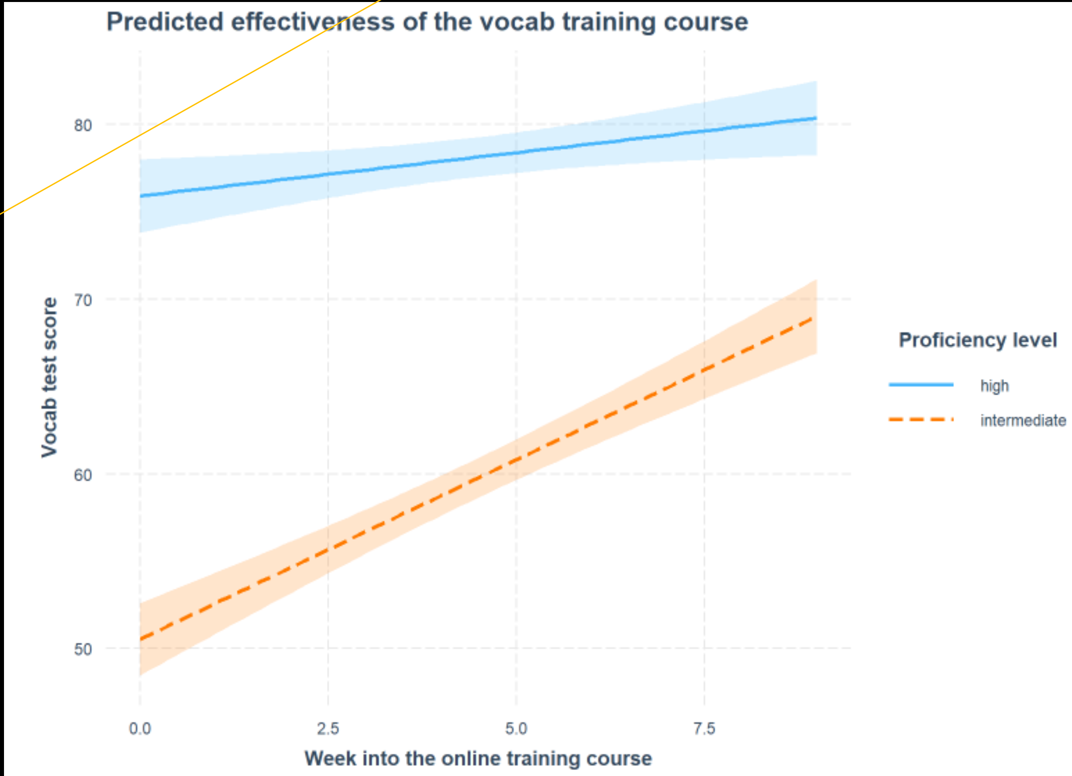
Slope: expected change of y when X1=0 and X2 changes from its reference category to another category.

- **Interactive** model

Slope: difference between the slope (rate of change) of the two regression lines

```
# version 1
m2bv1 <- lm(vocab_test_score ~ week + proficiency + week:proficiency, data = vocabdata)
# version 2
m2bv2 <- lm(vocab_test_score ~ week * proficiency, data = vocabdata)
```

```
## Call:
## lm(formula = vocab_test_score ~ week * proficiency, data = vocabdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.0330  -4.6421   0.5941   4.6204  28.1345
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   75.9092     1.0610  71.545  < 2e-16 ***
## week                           0.4967     0.2011   2.470    0.014 *
## proficiencyintermediate      -25.3788     1.4924 -17.005  < 2e-16 ***
## week:proficiencyintermediate   1.5591     0.2827   5.515 6.61e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.803 on 365 degrees of freedom
##   (31 observations deleted due to missingness)
## Multiple R-squared:  0.6338, Adjusted R-squared:  0.6308
## F-statistic: 210.5 on 3 and 365 DF,  p-value: < 2.2e-16
```



2.2(c). Model comparison

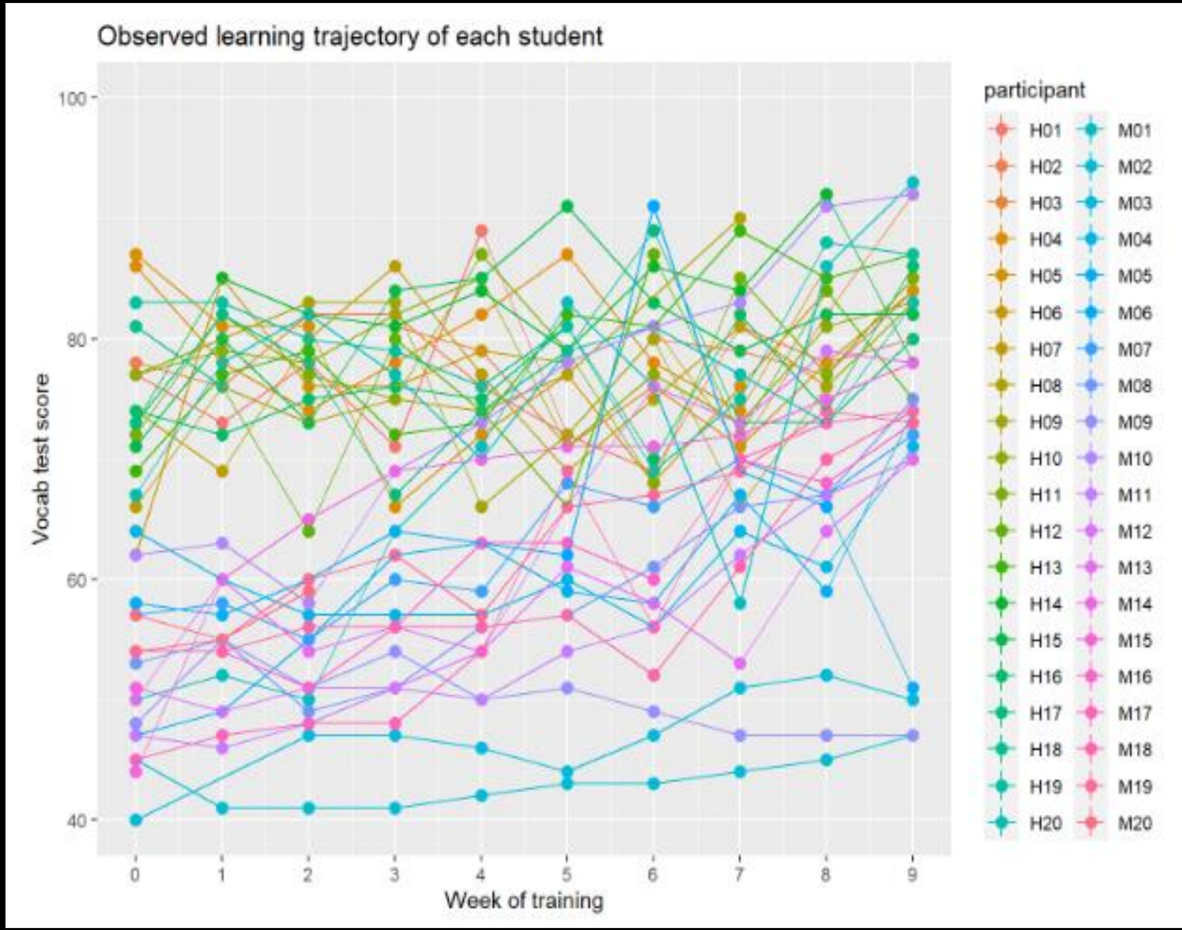
- Which model is better? How can we tell?

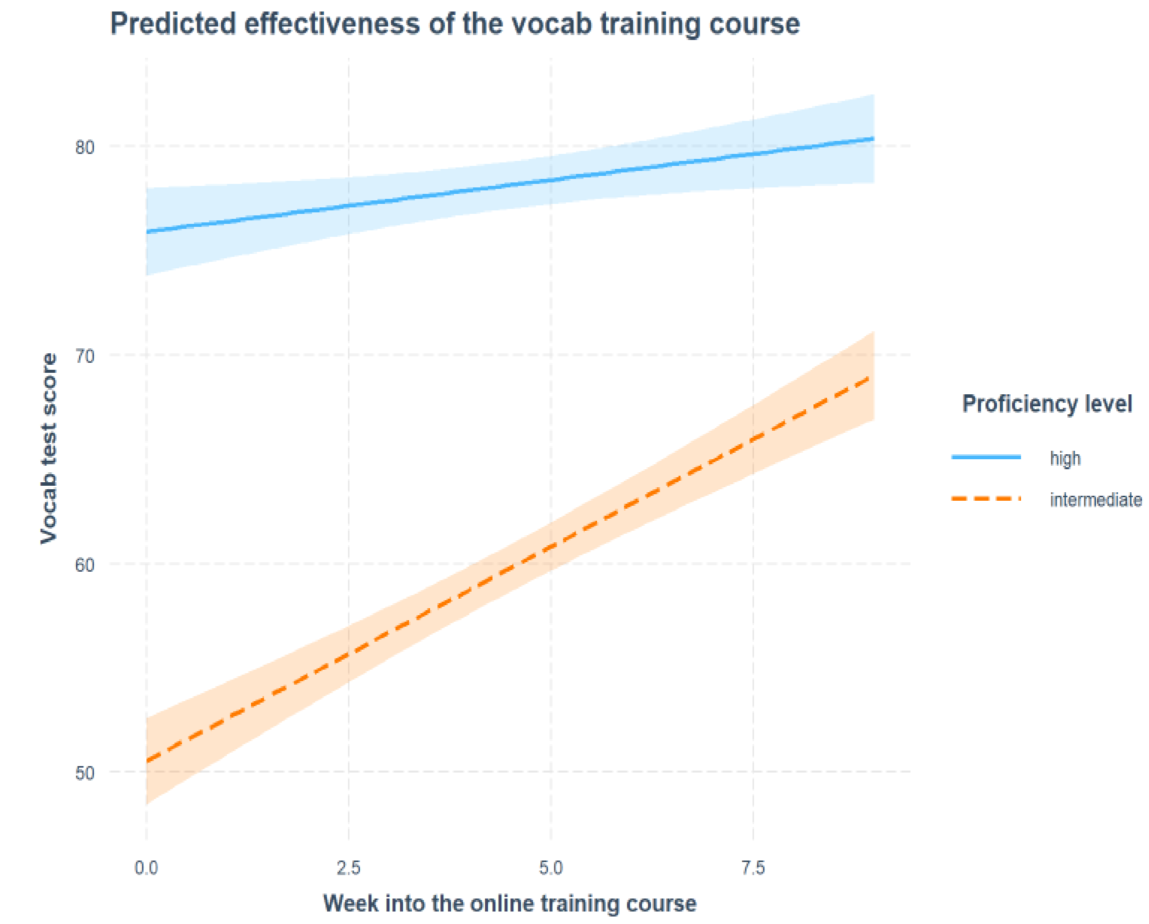
```
anova(m2a, m2bv1)
```

```
## Analysis of Variance Table
##
## Model 1: vocab_test_score ~ week + proficiency
## Model 2: vocab_test_score ~ week + proficiency + week:proficiency
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    366 24079
## 2    365 22226  1    1852.3 30.418 6.605e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Individual Difference

Recall the Data and the Regression(model) Plots





Any problems?

3.1 Deal with Individual Difference

- How can we deal with variances stemming from individual difference?

- Some thoughts:

A good idea? Why?

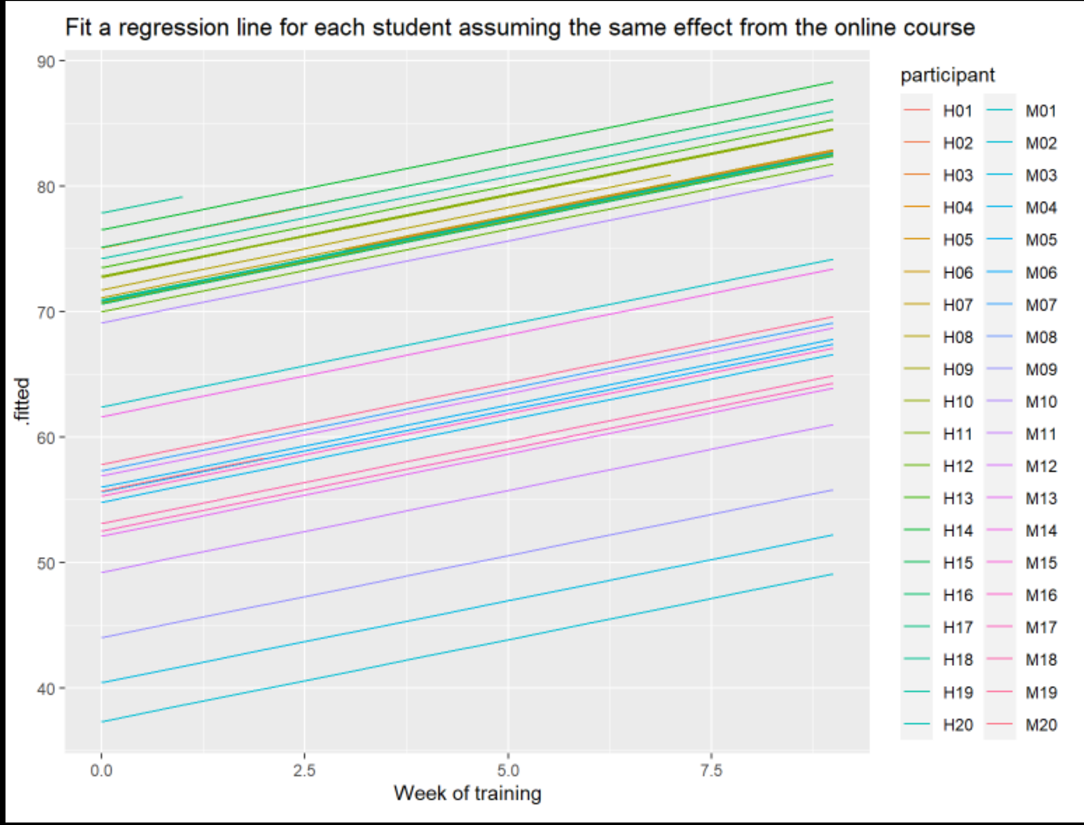
- Include "participant" as a predictor?

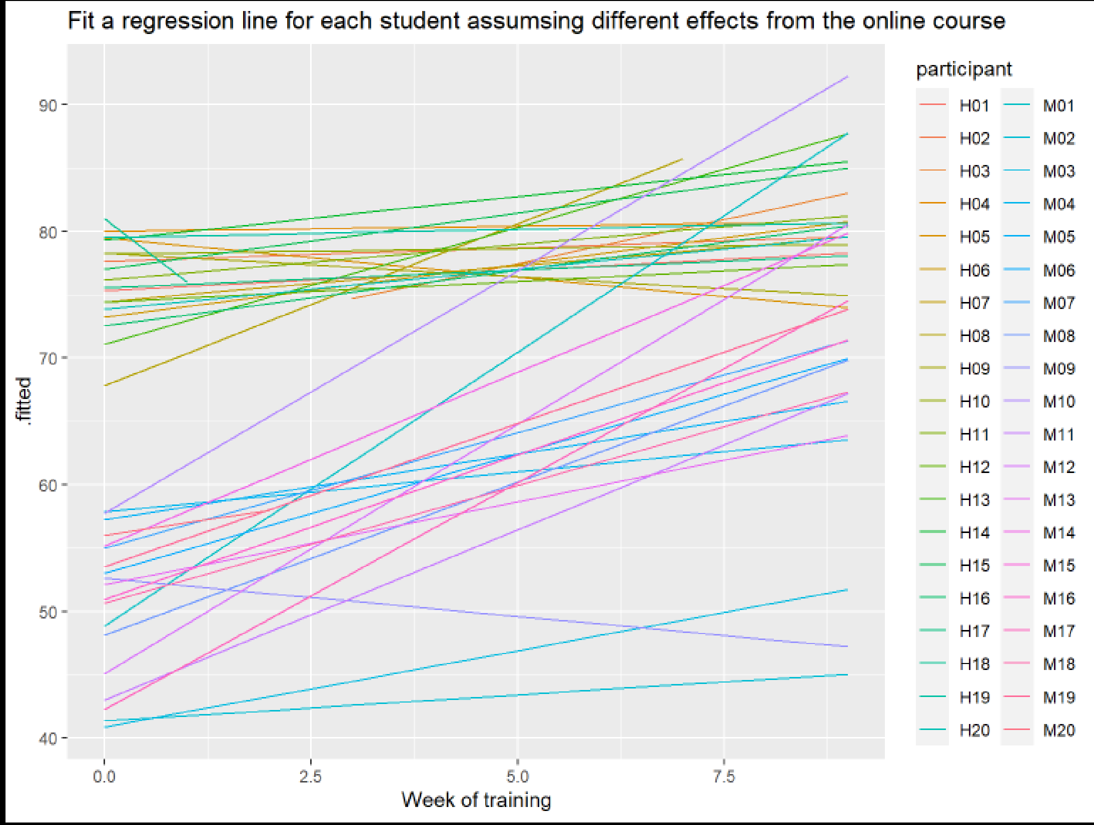
- Control for it as a covariate?

3.1 Deal with Individual Difference

```
m3a <- lm(vocab_test_score ~ week + participant, data = vocabdata)
summary(m3a)
```

```
m3b <- lm(vocab_test_score ~ week * participant, data = vocabdata)
summary(m3b)
```





3.2 Individual Difference in LMMs

- Recall the structure of a simple regression

```
m2bv2 <- lm(vocab_test_score ~ week * proficiency,
            data = vocabdata)
```

- Check out the structure of linear mixed-effect models

```
mixedm1 <- lmer(vocab_test_score ~ week * proficiency + (1 | participant),
                data = vocabdata)
```

```
mixedm2 <- lmer(vocab_test_score ~ week*proficiency + (1 + week | participant),
                data = vocabdata)
```

What differences do you notice?

THANK YOU

THE UNIVERSITY *of* EDINBURGH
Centre for Data, Culture & Society