Linear and Hierarchical Linear Models

Mixed Models, and Multilevel Models, GLMMs, etc…

Camila Pacheco Katrín Björnsdóttir

Today

- Linear Models
- What are hierarchical linear models
- Identify situations in which the use of mixed effects is appropriate
- Implement basic linear mixed models (LMM) with R
- Break ☻
- Practice

Required Material

You are required to have downloaded and installed

```
install.packages(c("lme4",
                   "ggeffects",
                   "stargazer"
                   ))
```

Required Material

Do not hesitate to ask questions!

Linear models

Linear models

Now we want to test if there is a relationship between petal length and petal width. For that we run a simple linear model.

```
iris.m1 <- lm(Petal.Length ~ Petal.Width, data = iris)
summary(iris.m1)
```

```
Call:
lm(formula = Petal.Length ~ Petal.Width, data = iris)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.33542 -0.30347 -0.02955  0.25776  1.39453 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.08356    0.07297   14.85   <2e-16 ***
Petal.Width  2.22994    0.05140   43.39   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4782 on 148 degrees of freedom
```

The summary function gives you the output from the model. We can see that petal width significantly effects petal length (as you might have expected). Our model did *very* good, explaining about 93% of the variation in petal length.

Linear models

But do we expect there to be a difference between species Lets add species as an interaction to make it a bit more complicated.

```r
iris.m2 <- lm(Petal.Length ~ Petal.Width * Species, data = iris)
summary(iris.m2)
```

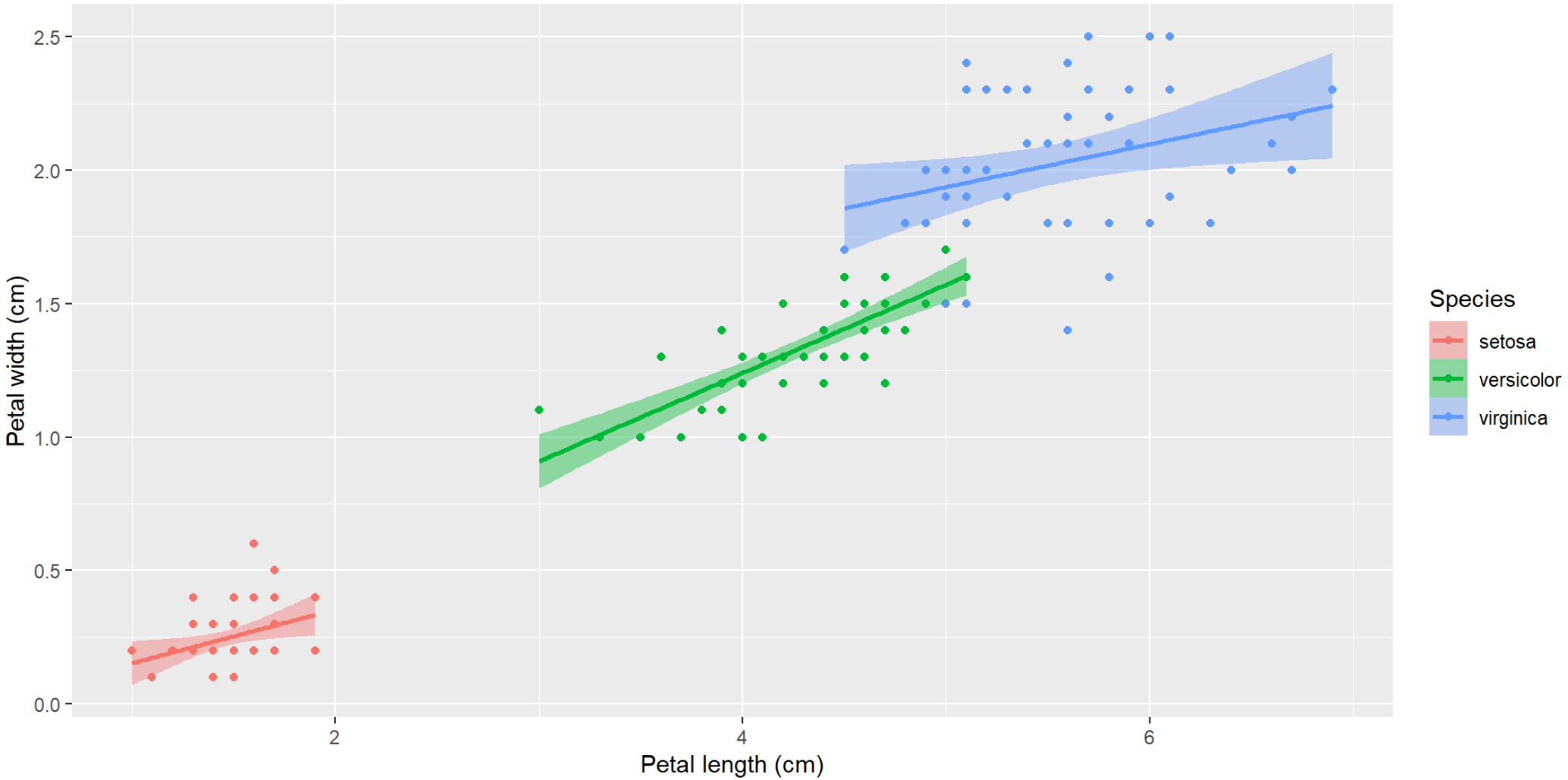
```
Call:
lm(formula = Petal.Length ~ Petal.Width * Species, data = iris)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.84099 -0.19343 -0.03686  0.16314  1.17065 

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)    
(Intercept)                   1.3276     0.1309  10.139  < 2e-16 ***
Petal.Width                   0.5465     0.4900   1.115   0.2666    
Speciesversicolor             0.4537     0.3737   1.214   0.2267    
Speciesvirginica              2.9131     0.4060   7.175 3.53e-11 ***
Petal.Width:Speciesversicolor 1.3228     0.5552   2.382   0.0185 *  
```

Linear models

Now lets visualize the relationship using ggplot (notice how we only included stat`smooth to our original scatter plot).

```
p<-iris %>%
  ggplot(aes(
    x = Petal.Length,
    y = Petal.Width,
    color = Species)) + # to assign a color to each group
  geom_point() +
  stat_smooth(method = "lm", aes(fill = Species, colour = Species))
  labs(
    x = "Petal length (cm)",
    y = "Petal width (cm)")
```

Linear models



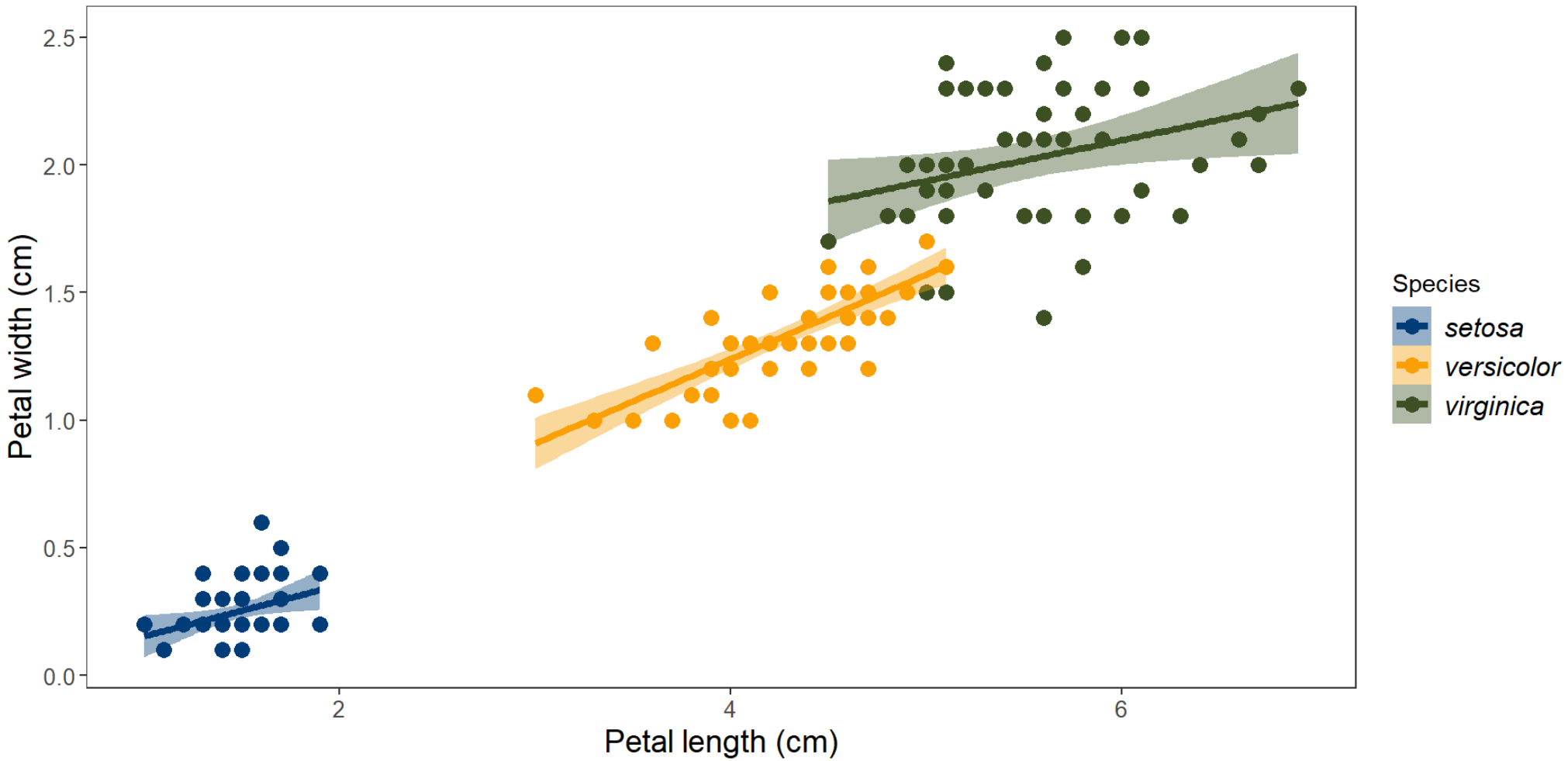
Linear models

In the previous plot we just made we used the default ggplot settings. Although this is a great looking plot, we can make it even better since ggplot has a lot of power in customization, here is one example but feel free to play around with this, it can actually be quite satisfying when you get the hang of it.

```
p<-iris %>%
  ggplot(aes(
    x = Petal.Length,
    y = Petal.Width,
    color = Species # to assign a color to each group
  )) +
  geom_point(size = 3) + # to plot a scatter plot
  stat_smooth(method = "lm", aes(fill = Species, colour = Species),
  scale_color_manual(name = "Species", values = c("#023d79", "#faa3
  scale_fill_manual(name = "Species", values = c("#023d79", "#faa30
  labs(
    x = "Petal length (cm)", #x axis label
    y = "Petal width (cm)" #y axis label
```

```
14    ) +
15    theme_bw()+ #theme within the ggplot2 package (different themes a
16    theme(axis.text.x = element_text(size = 10), #use the theme funct
17          axis.text.y = element_text(size = 10),
18          axis.title.x = element_text(size = 14, face = "plain"),
```

Linear models



Generalize linear models

Get familiar with different data distributions Here is a brief summary of the data distributions you might encounter most often.

- Gaussian - Continuous data (normal distribution and homoscedasticity assumed)
- Poisson - Count abundance data (integer values, zero-inflated data, left-skewed data)
- Binomial - Binary variables (TRUE/FALSE, 0/1, presence/absence data)

Generalize linear models

(a non-exhaustive list of…)

COMMON DATA TYPES IN ENVIRONMENTAL SCIENCES AND THEIR ASSOCIATED DISTRIBUTIONS & TESTS

DATA TYPE	DISTRIBUTION: TESTS	EXAMPLES
Continous data	Gaussian (normal): lm, mixed-effects models	Traits like height, weight, nutrient content, and anything else you can measure across a range centered on a mean value
Count data (whole numbers)	Poisson: glm, glmm	Population counts, number of species in an area, of youngs in a litter, and other discrete measures
Proportion data	If 2 possible outcomes, Binomial: glm, glmm	Survival, seed germination trials, other events that can be described as “successes” and “failures”
	If more outcomes: chi-squared test	Habitat selection (does a species utilise a type of habitat in greater proportion than its availability?), differences in vegetation types between sites or over time

Generalize linear models

Syntax for a model with poisson (count) distribution

```
glm(y ~ x, family = poisson, data = df)
```

Syntax for a model with binomial distribution

```
glm(y ~ x, family = binomial, data = df)
```

Hierarchical Linear Models

Ecological and biological data can be complex!

- Hierarchical structure in the data
- Many covariates and grouping factors
- Unbalanced study/experimental design

What is Independence Assumption?

A linear regression model assumes that :

Any other data point does not influence each data point in a dataset

- We are **NOT** referring to your independent and responses variable
- *Ideally* we want them to be correlated it

We are referring withing the variable

Diagnosis of dependence?

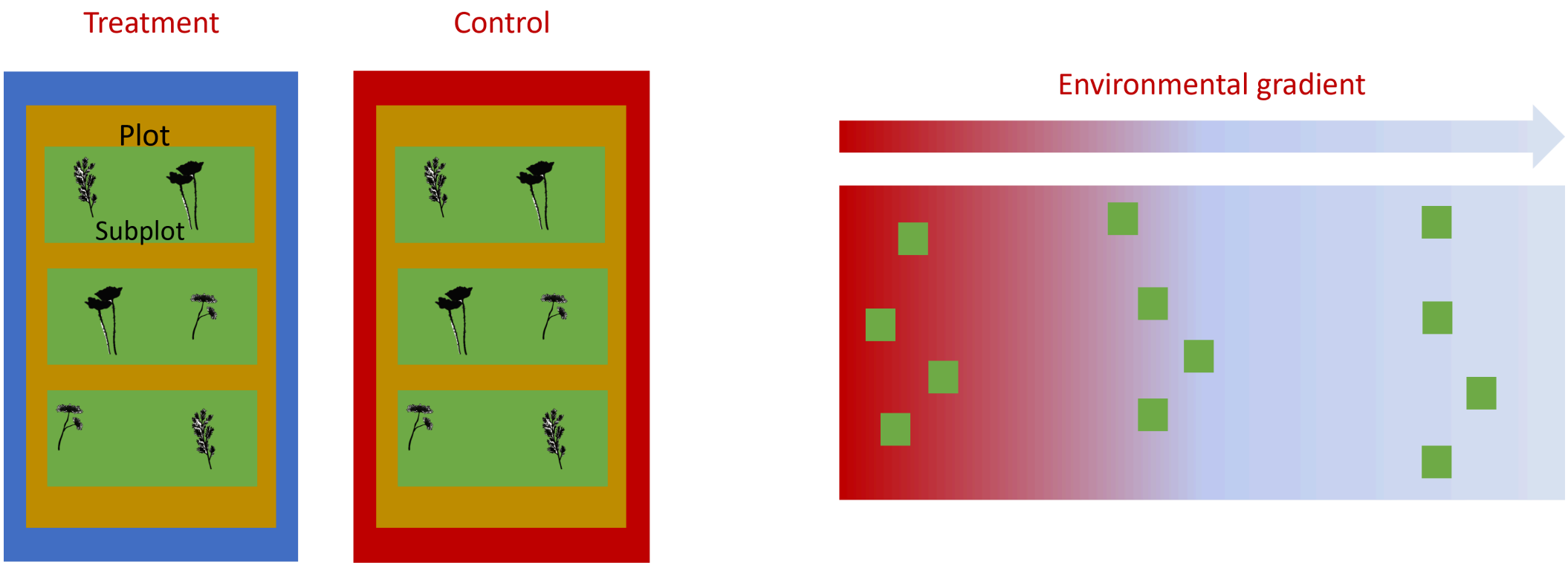
Think about how the data were collected.

Are there clusters in my data?

For example:

Are there data from individual species clustered within sampling area?

Hierarchical structres example



Hierarchical structres in ecology

- Grouping factors: populations, species, sites
- Sample sizes: Species area relationship
- Time: Might take repeated measurements of the same plant in time
- Space: the closer the similar

How could we analyze this data?

We need to explicitly account for the correlated nature of the data

The *random effects* structure will aid correct inference about *fixed effects*, depending on which level of the system's hierarchy is being manipulated.

- What is random effects ?
- What is fixed effects?
- When to used them ?

When to used them ?

You will need to used random and fixed effects every time your data has a **Hierarchical** structure.

- It’s important to not that this difference has little to do with the variables themselves, and a lot to do with your research question!
- What is just variation (“**noise**”) that you need to control for?

Fixed effects : deterministic processes

- They are like drivers, categories or groups that you think might directly influence the outcome you're studying.
- They're called "*fixed*" because you're interested in the specific levels of these categories.
- `"I want to know how these different things (fixed effects) affect the outcome."`
- levels of a factor (qualitative variable)
- a predictor (quantitative variable)

Random effects : stochastic processes

Data for this part of the session

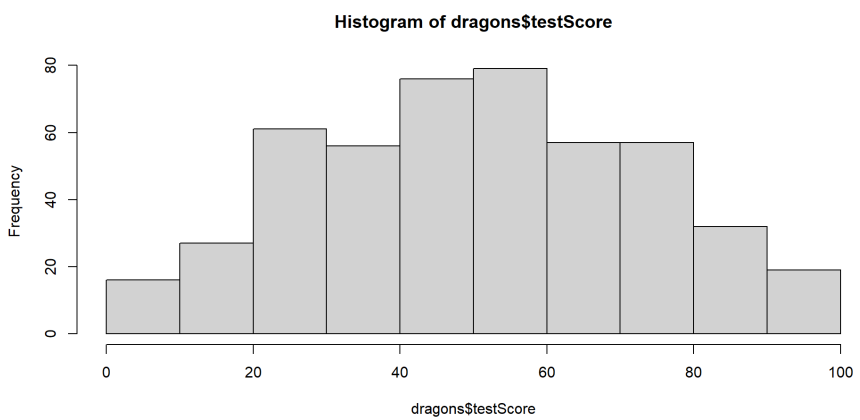
Coding Club Mixed models

Explore the data

```
1 library(tidyverse)
2 load("CC-Linear-mixed-models/dragons.RData")
3 head(dragons)
```

```
  testScore bodyLength mountainRange site
1 16.147309   165.5485      Bavarian    a
2 33.886183   167.5593      Bavarian    a
3  6.038333   165.8830      Bavarian    a
4 18.838821   167.6855      Bavarian    a
5 33.862328   169.9597      Bavarian    a
6 47.043246   168.6887      Bavarian    a
```

```
1 hist(dragons$testScore)
```



Scaling the data

It is good practice to standardize your explanatory variables before proceeding so that they have a mean of zero ("centering") and standard deviation of one ("scaling").

If two variables in the same model have very different scales, the mixed model will likely return a `convergence error` when trying to compute the parameters.

```
dragons<- dragons |>
  mutate(bodyLength2=scale(bodyLength,center = TRUE, scale = TRUE))
```

Research question

Is the test score affected by body length?

Linear model

```r
basic.lm <- lm(testScore ~ bodyLength2, data = dragons)
summary(basic.lm)
```

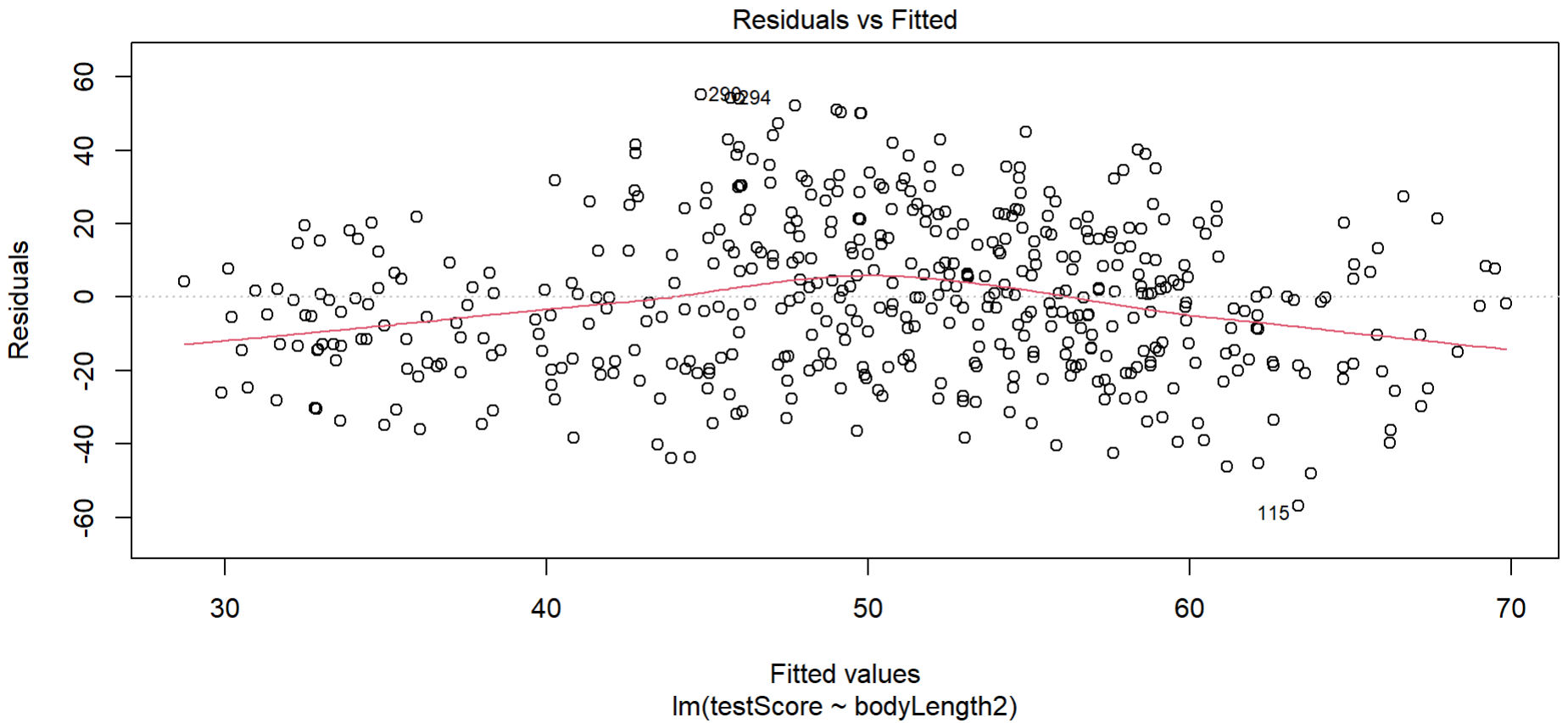
```
Call:
lm(formula = testScore ~ bodyLength2, data = dragons)

Residuals:
    Min      1Q  Median      3Q     Max
-56.962 -16.411  -0.783  15.193  55.200

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  50.3860     0.9676  52.072   <2e-16 ***
bodyLength2   8.9956     0.9686   9.287   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 21.2 on 478 degrees of freedom
```

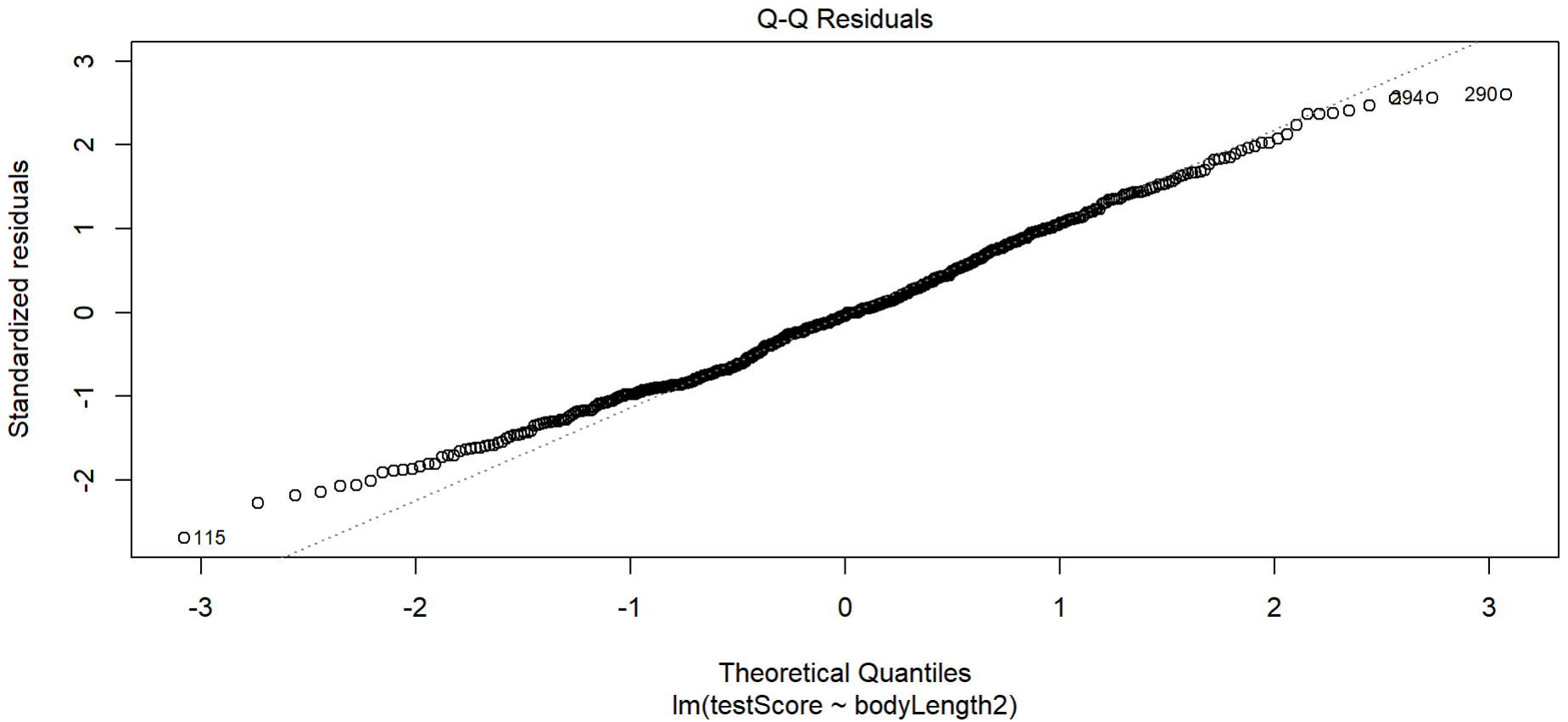
Linear model

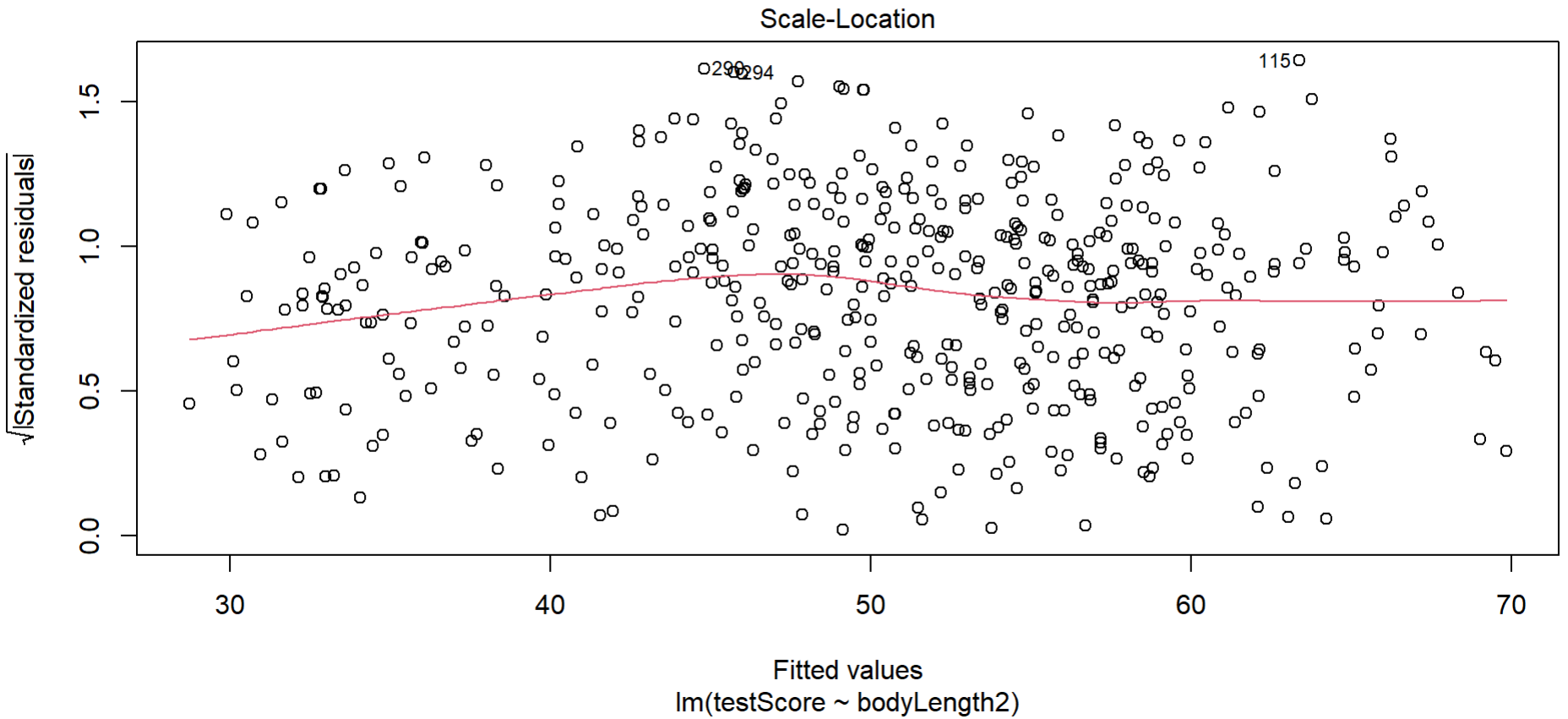
```r
(prelim_plot <- ggplot(dragons, aes(x = bodyLength, y = testScore))
  geom_point() +
  geom_smooth(method = "lm"))+
  theme_classic()
```

Assumptions check

```
1 plot(basic.lm)
```




Q-Q Residuals
Standardized residuals
3
2
1
0
-1
-2
115
294
290
-3
-2
-1
0
1
2
3
Theoretical Quantiles
lm(testScore ~ bodyLength2)


Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
30
40
50
60
70
Fitted values
lm(testScore ~ bodyLength2)

Assumptions check

Are our data independent?

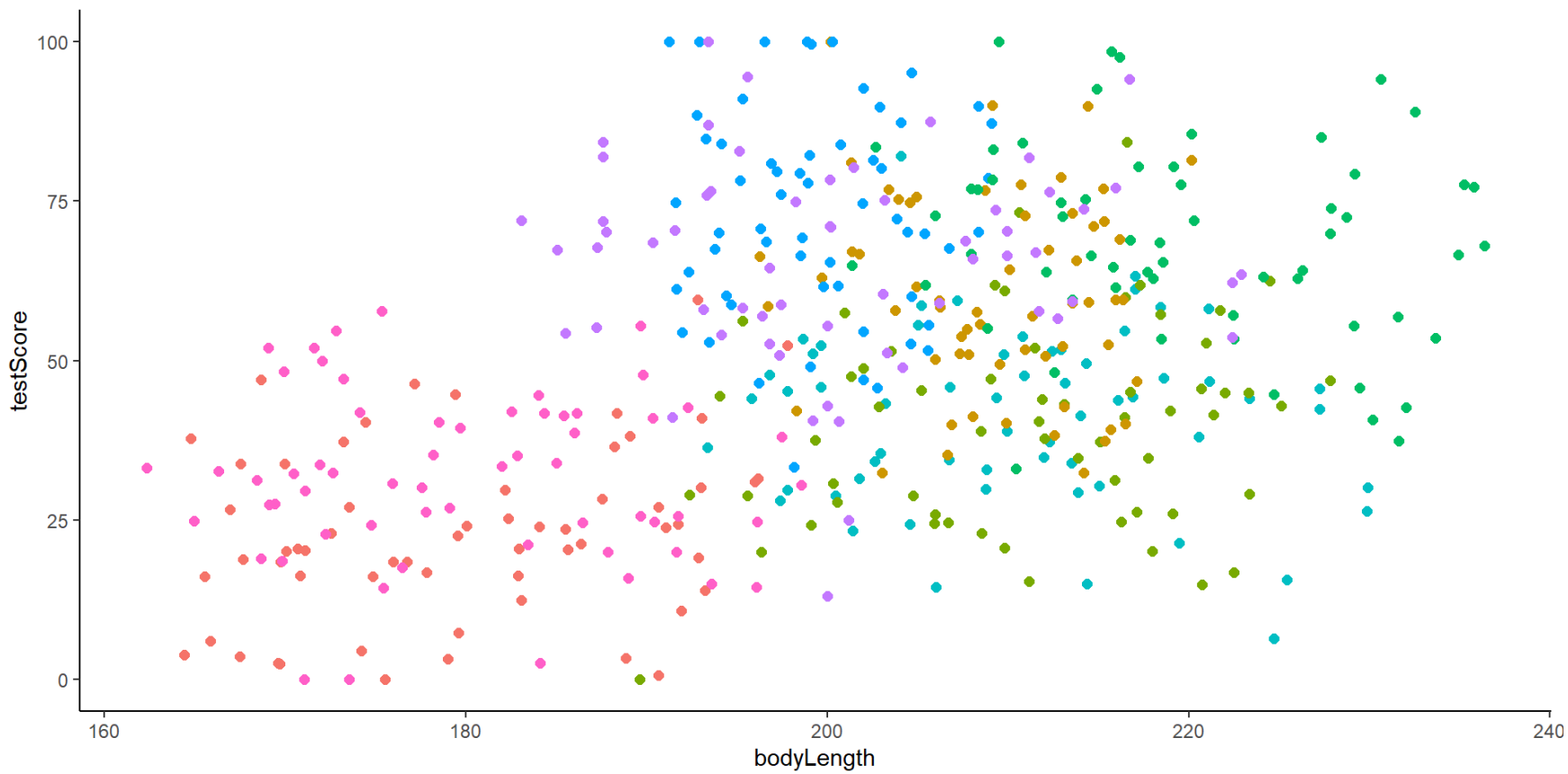
Assumptions check

`Data description:` The data were collected from multiple samples from eight mountain ranges.

- It's perfectly plausible that the data from within each mountain range are more similar to each other than the data from different mountain ranges
- they are Hierarchical!

Assumptions check

```
(colour_plot <- ggplot(dragons, aes(x = bodyLength, y = testScore,
  geom_point(size = 2) +
  theme_classic() +
  theme(legend.position = "none"))
```



How to implement mixed models in R?

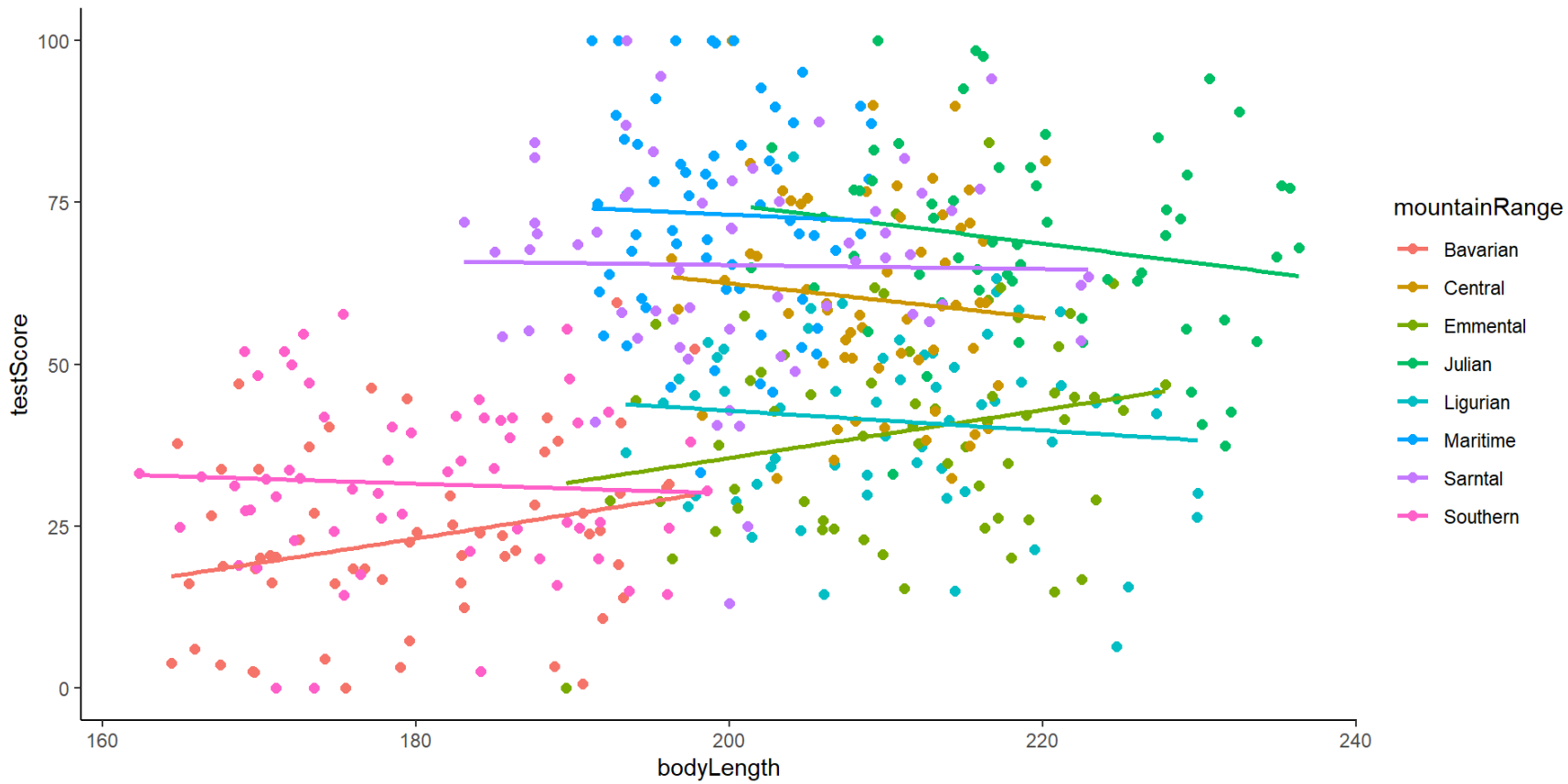
- Step 1: Model building
- Step 2: Model validation
- Step 3: Model interpretation
- Step 4: Model visualization

Step 1: Model building

Hierarchical linear models do is they essentially fit a separate regression line for each and every cluster.

Step 1: Model building Dragons

```r
(colour_plot <- ggplot(dragons, aes(x = bodyLength, y = testScore,
  geom_point(size = 2) +
  geom_smooth(method = "lm",se=FALSE)+
  theme_classic() )
```



Step 1: Model building

Hierarchical linear models do is they essentially fit a separate regression line for each and every cluster. And then estimates what we call the *Fixed slope*. Average slope between x and y across my clusters.

Mathematically speaking it is more complicated than that.

Step 1: Model building Dragons

```r
library(lme4) #"linear mixed model" function from lme4 package
mixed.lmer <- lmer(testScore ~ bodyLength2 +
                     (1|mountainRange), #random effect
                   data = dragons,
                   REML = TRUE #estimation method other method ML b
                   )
summary(mixed.lmer)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: testScore ~ bodyLength2 + (1 | mountainRange)
   Data: dragons

REML criterion at convergence: 3985.6

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.4815 -0.6513  0.0066  0.6685  2.9583

Random effects:
 Groups        Name        Variance Std.Dev.
 mountainRange (Intercept) 339.7    18.43
 Residual                  223.8    14.96
Number of obs: 480, groups:  mountainRange, 8
```

Step 1: Model building Dragons

Mountain ranges are clearly important: they explain a lot of variation:

Factor	Variance
Mountain range	339.7
Residuals	223.8
Body length	?

```
1  (339.7/(339.7 + 223.8) )* 100
```

```
[1] 60.28394
```

Variance Body length

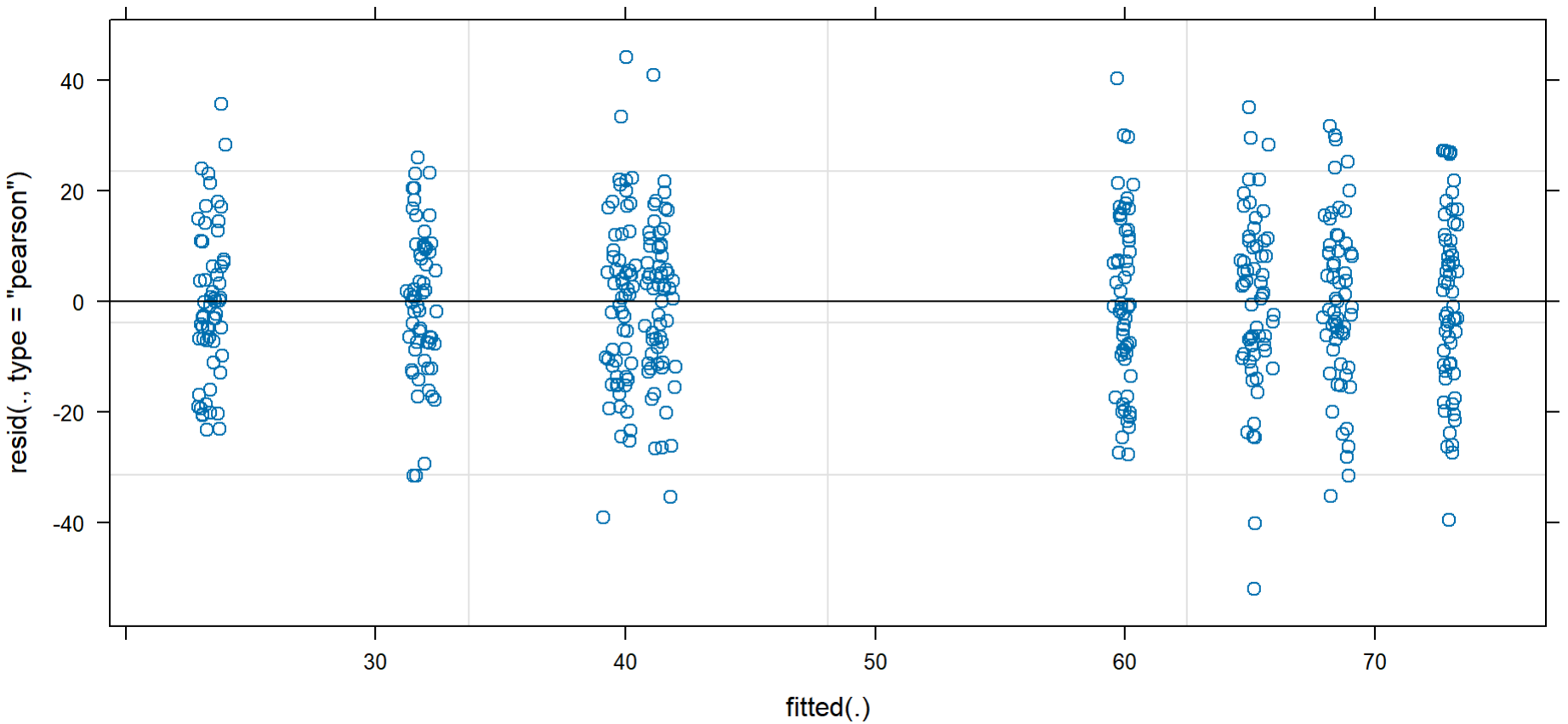
```
1  100-60.28
```

```
[1] 39.72
```

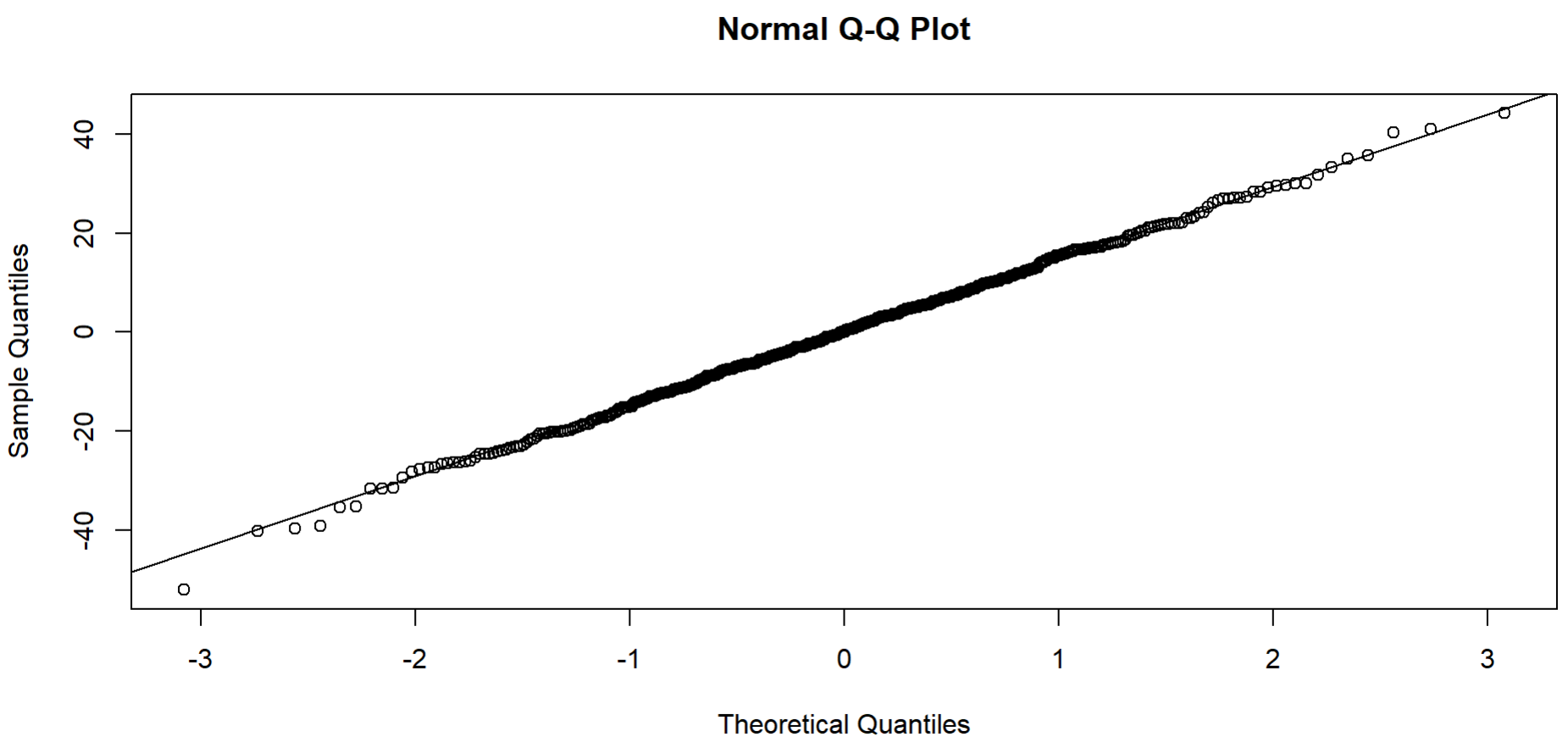
Step 2: Model validation Dragons

Assumptions check

```
1 plot(mixed.lmer)
```



```
1 qqnorm(resid(mixed.lmer))
2 qqline(resid(mixed.lmer))
```



Step 1: Model building Nesting

Example:

10 control | 10 experimental

- 3 years
- Each season
- 20 beds
- 50 seedlings
- 5 leaves
- 5 leaves x 50 seedlings x 20 beds x 4 seasons x 3 years = 60 000 measurements per treatment

Step 1: Model building Nesting

Effect of treatment in leaf length

```
leafLength ~ treatment
```

- *Pseudoreplication*
- Massively increasing sampling size

Better model

```
leafLength ~ treatment + (1|Bed/Plant/Leaf)
```

What about the crossed effects ?

- Crossed (or partially crossed) random factors that do not represent levels in a hierarchy.
- This account for the fact that all plants in the experiment, regardless of the fixed (treatment) effect, may have experienced a very hot summer in the second year.

```
leafLength ~ treatment + (1|Bed/Plant/Leaf) + (1|Season)
```

Step 1: Model building Dragons Nesting

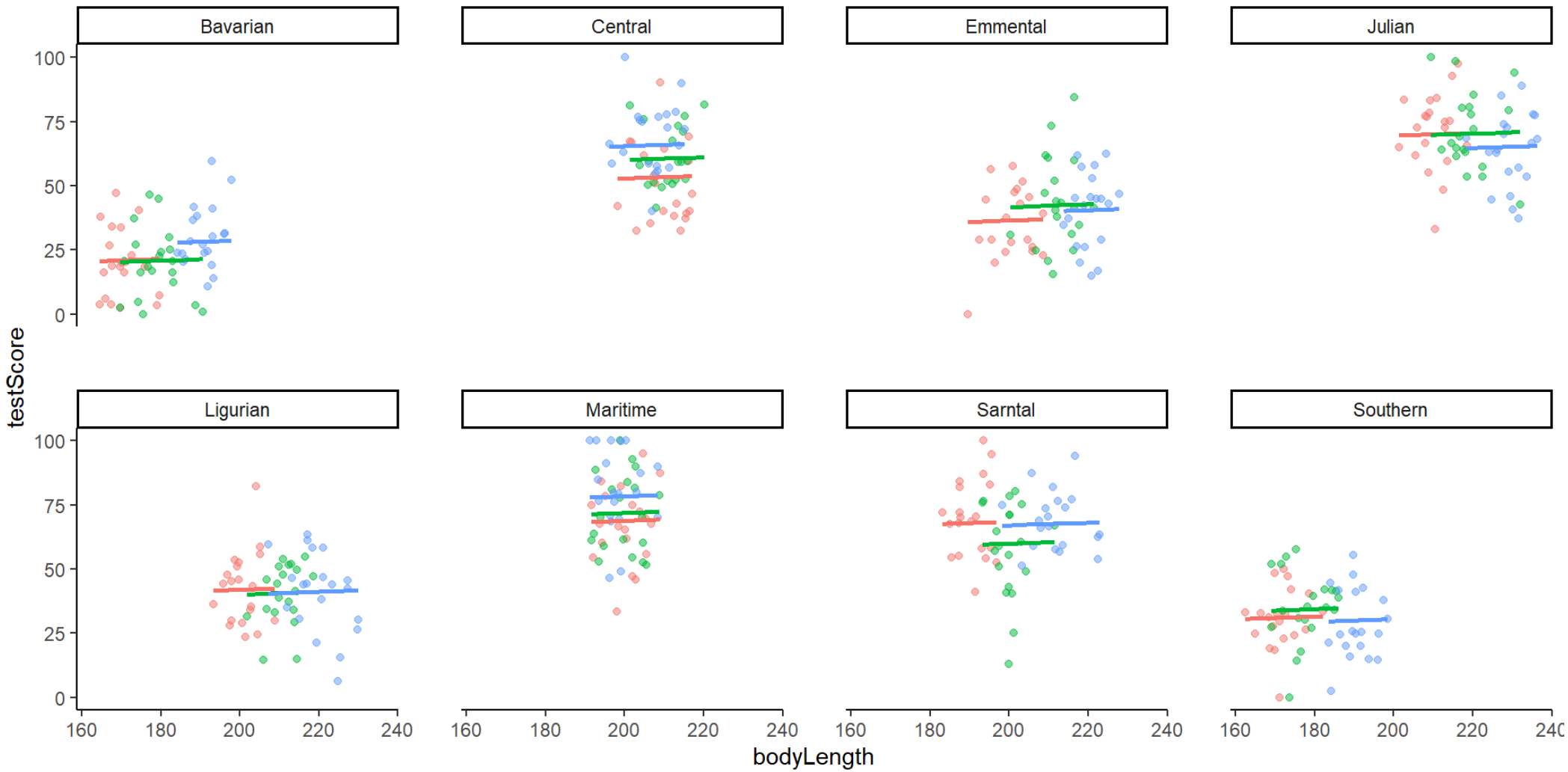
```
mixed.lmer2 <- lmer(testScore ~ bodyLength2+
                      (1|mountainRange/site),
                    data = dragons)
summary(mixed.lmer2)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: testScore ~ bodyLength2 + (1 | mountainRange/site)
   Data: dragons

REML criterion at convergence: 3970.4

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.2425 -0.6752 -0.0117  0.6974  2.8812

Random effects:
 Groups              Name        Variance Std.Dev.
 site:mountainRange (Intercept)  23.09    4.805
 mountainRange      (Intercept) 327.56   18.099
 Residual                       208.58   14.442
Number of obs: 480, groups:  site:mountainRange, 24; mountainRange, 8
```

```r
mixed.lmer3 <- lmer(testScore ~ bodyLength2
                    +(1|mountainRange) + (1|mountainRange:site),
                    data = dragons)
summary(mixed.lmer3)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula:
testScore ~ bodyLength2 + (1 | mountainRange) + (1 | mountainRange:site)
   Data: dragons

REML criterion at convergence: 3970.4

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.2425 -0.6752 -0.0117  0.6974  2.8812

Random effects:
 Groups              Name        Variance Std.Dev.
 mountainRange:site (Intercept)  23.09    4.805
 mountainRange      (Intercept) 327.56   18.099
 Residual                       208.58   14.442
```

Step 1: Model building Dragons Nesting



Random slopes and Random intercept

A *random-intercept* model recognizes that each cluster might have its own starting point (intercept), but keeps the slope constant among them. So in our example we acknowledge that some populations may be smarter or dumber to begin with.

Lets say we expect that dragons in all mountain ranges do not exhibit the same relationship between body length and intelligence (rando, slope)

Random slopes and Random intercept

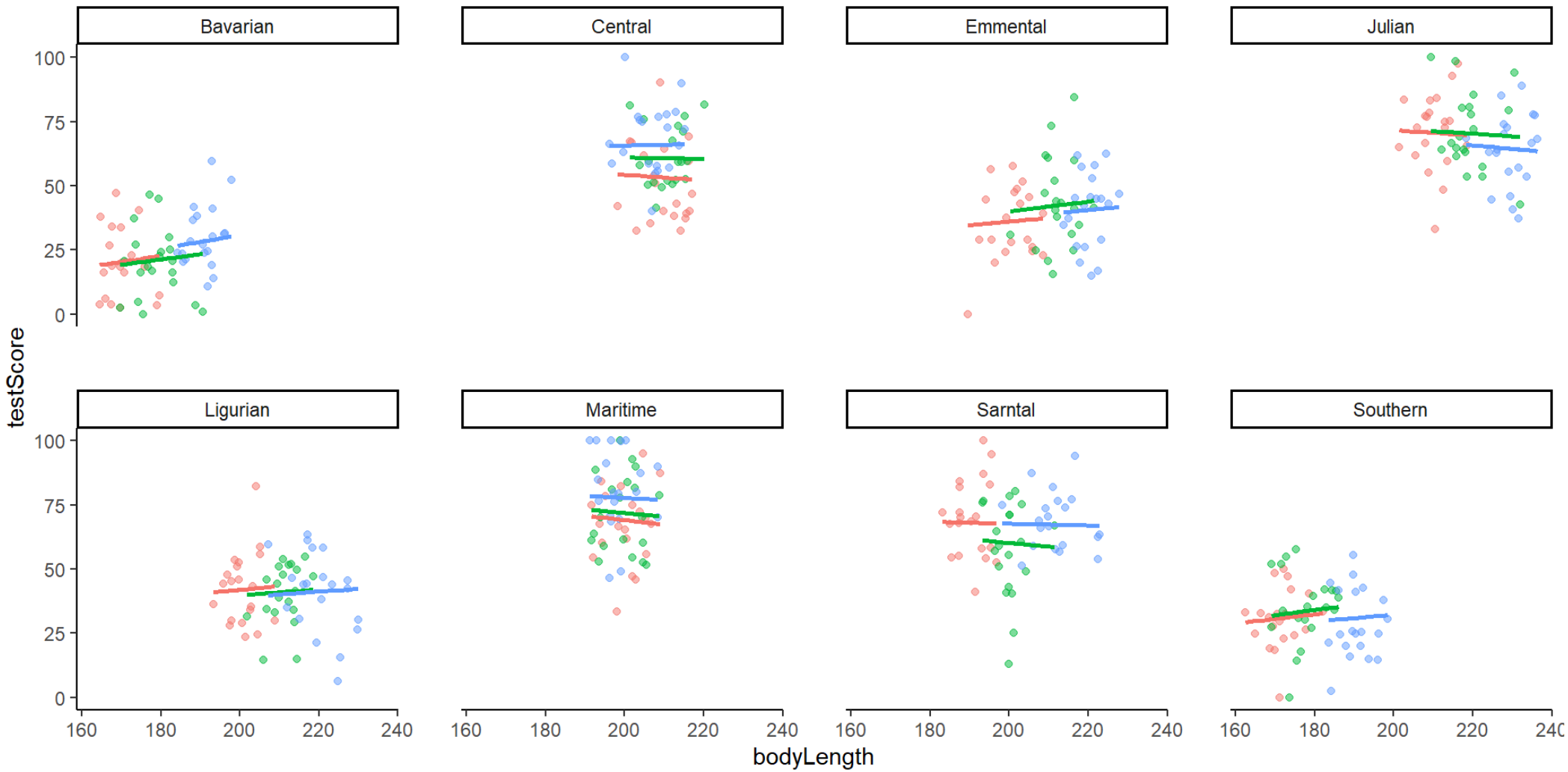
We only need to make one change to our model to allow for random slopes as well as intercept, and that's adding the fixed variable into the random effect brackets:

```
mixed.ranslope <- lmer(testScore ~ bodyLength2 +
                         (1 + bodyLength2|mountainRange/site),
                       data = dragons)

summary(mixed.ranslope)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: testScore ~ bodyLength2 + (1 + bodyLength2 | mountainRange/site)
   Data: dragons

REML criterion at convergence: 3968.4

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.2654 -0.6737 -0.0200  0.6931  2.8432

Random effects:
 Groups              Name        Variance Std.Dev. Corr
```

```
 site:mountainRange (Intercept)  19.8156  4.4515
                    bodyLength2   0.7178  0.8472  1.00
 mountainRange      (Intercept) 310.9691 17.6343
```

Random slopes and Random intercept



Step 3: Model interpretation

```r
library(stargazer)
stargazer(mixed.lmer2,
          digits = 3,
          type="text",
          star.cutoffs = c(0.05, 0.01, 0.001),
          digit.separator = "")
```

```
================================================
                        Dependent variable:
                    ---------------------------
                              testScore
------------------------------------------------
bodyLength2                     0.831
                               (1.681)

Constant                       50.386***
                               (6.507)

------------------------------------------------
Observations                     480
Log Likelihood                -1985.195
Akaike Inf. Crit.             3980.389
```

Step 4. Visualization

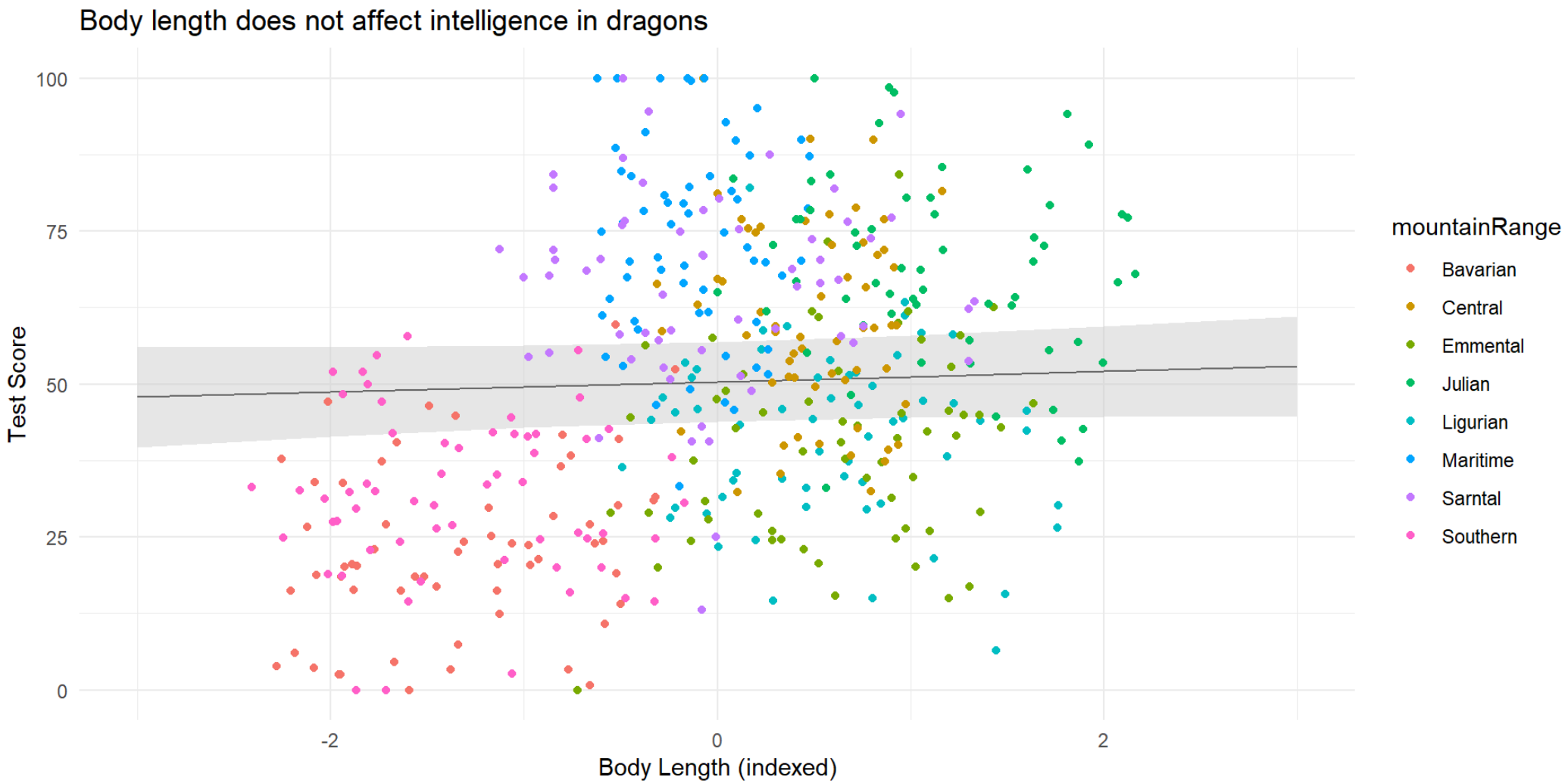
```r
library(ggeffects)

# Extract the prediction data frame
pred.mm <- ggpredict(mixed.lmer2, terms = c("bodyLength2"))   # this
head(pred.mm)
```

```
# Predicted values of testScore

bodyLength2 | Predicted |       95% CI
-------------------------------------
          -3 |     47.89 | 31.72, 64.07
          -2 |     48.72 | 34.33, 63.12
          -1 |     49.56 | 36.35, 62.76
           0 |     50.39 | 37.60, 63.17
           1 |     51.22 | 38.01, 64.42
           2 |     52.05 | 37.66, 66.44

Adjusted for:
*           site = 0 (population-level)
* mountainRange = 0 (population-level)
```

Step 4. Visualization

```r
# Plot the predictions
p<-ggplot(pred.mm) +
    # slope
    geom_line(aes(x = x, y = predicted)) +
    # error band
    geom_ribbon(
      aes(
        x = x,
        ymin = predicted - std.error,
        ymax = predicted + std.error
      ),
      fill = "lightgrey",
      alpha = 0.5
    ) +
    # adding the raw data (scaled values)
    geom_point(data = dragons,
               aes(x = bodyLength2, y = testScore, colour = mountai
    labs(x = "Body Length (indexed)",
```

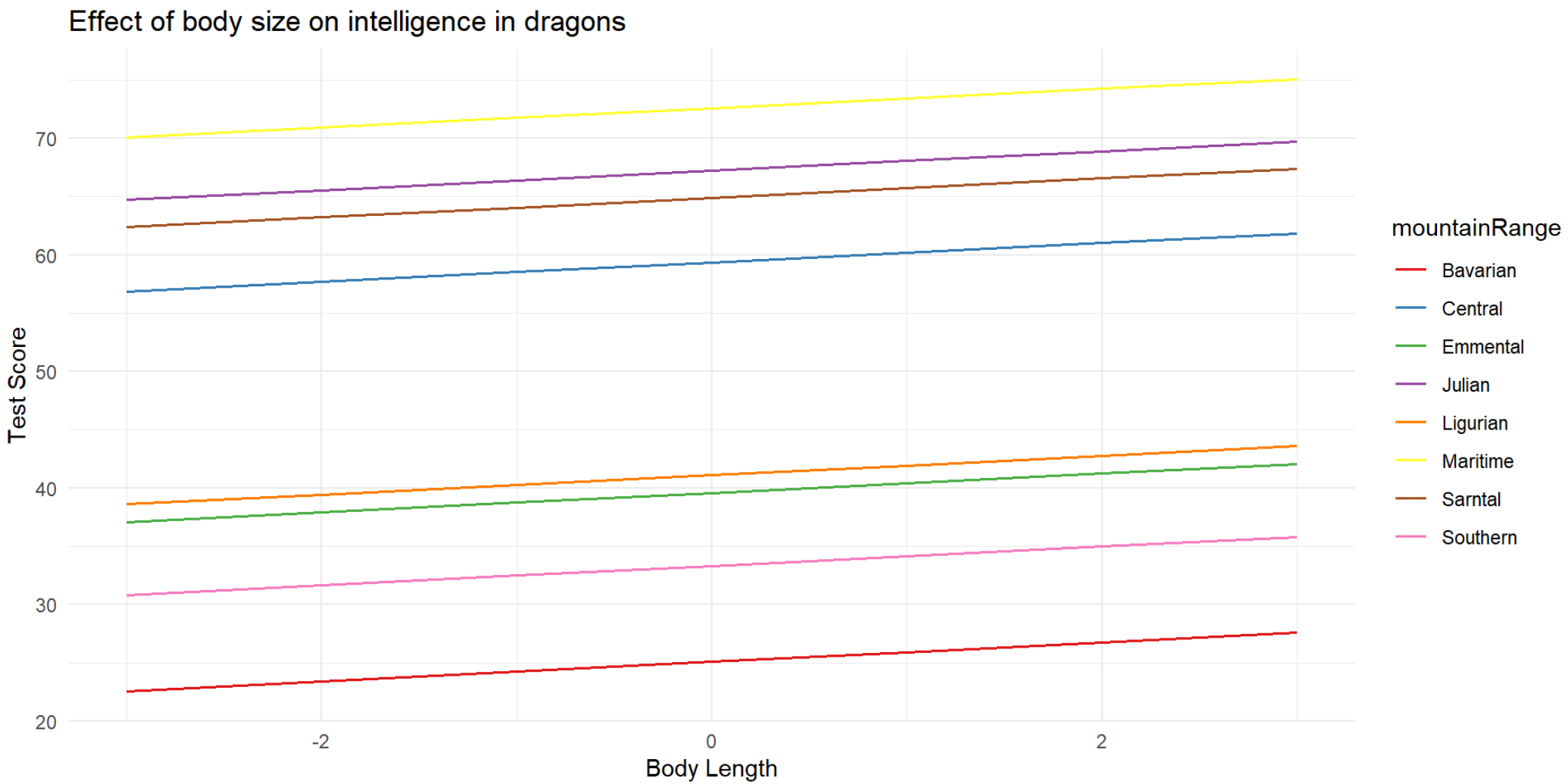
Step 4. Visualization



Step 4. Visualization

```
p<-
  ggpredict(mixed.lmer2, terms = c("bodyLength2", "mountainRange"),
            type = "random") %>%
   plot(show_ci = FALSE) +
   labs(x = "Body Length", y = "Test Score",
        title = "Effect of body size on intelligence in dragons") +
   theme_minimal()
```

Step 4. Visualization



Additional ressources

Popular libraries for (G)LMMs:

- Frequentist : `nlme`, `lme4`, `glmmTMB`
- Bayesian : `brms`, `rstan`, `rstanarm`, `MCMCglmm`
- Nice visualization : link
- Visit: Coding Club

BREAK