Mixed effects model lab

AUTHOR
Melissa Pulley

This lab

If you're able to open the .qmd file on your computer, you can submit the lab that way. That will also make grading easier and faster!

Ideally, try to render it to create an html document. If that's not working, submit the .qmd file.

If that's not possible, please submit it as before.

Walleye and Hg data

Make sure to understand the model structure of a mixed effect model.

We are going to simulate the data. While usually I provide the data, I think it's important for you to understand how this data is simulated, as it relates to the way the underlying system works.

This is the same example as in the class. We are sampling walleye in Lake Michigan, and are worried about the concentration of Hg and how it relates to length.

We sample four sites and measure length and Hg of all individuals.

We are simulating the data, and in this case

Regions

In this dataset we have four regions, and we need to generate a random intercept around them:

```r
gamma <- rnorm(n=4,mean=0,sd=0.25)
```

Another random components is the number of individuals caught. The effort was the same in the four sites, and catchability is constant, meaning that they should catch about the same number of individuals.

```r
N<- round(rnorm(4,20,2))
N
```

```
[1] 22 19 20 22
```

Simulation

Once we obtain the number of individuals, we will use a random uniform distribution to simulate the number of individuals that were caught.

Finally, the amount of Hg in an individual will be given by:

$$Hg_{i,j} = 0.5 + 0.18(x_{i,j}) + \gamma_j + \epsilon$$

We will do this four times, and in order to save time, we will do it in a for loop and will save the results in a list

```
HgDat<-list()
```

Once we created the lists, we can run the simulation:

```
for(i in 1:4){
  x<-runif(N[i],20,60)
  y<-0.5 + 0.018*x + gamma[i] + rnorm(N[i],0,0.08)
  Region<-rep(LETTERS[i],N[i])
  HgDat[[i]]<-data.frame(size=x,Hg=y,region=Region)
}
```

Check whether the data makes sense. Run HgDat, and you should see four different dataframes, one for each region. There was an easy way to make this into a single dataframe, but lists can be very neat.

In order to have a single dataframe, we can do the following:

```
Warning: package 'dplyr' was built under R version 4.3.2
```

```
Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
HgDat_df<-bind_rows(HgDat)
HgDat_df$region<-as.factor(HgDat_df$region)
```

Now we have an object called HgDat_df with all of the data. Let’s look at it

```
library(knitr)
library(kableExtra)
```

```
Warning: package 'kableExtra' was built under R version 4.3.3
```

```
Attaching package: 'kableExtra'
```

```
The following object is masked from 'package:dplyr':

    group_rows
```

```
library(dplyr)
kbl(HgDat_df, col.names = gsub("[.]", " ", names(HgDat_df)))%>%
  kable_paper() %>%
  scroll_box(width = "500px", height = "200px")
```

size	Hg	region
40.97278	0.8726418	A
43.25552	0.8752156	A
56.72066	1.2067227	A
48.15031	1.0669432	A
47.30596	1.0424994	A

Model

In order to run a GLMM with a random intercept, we can use the following code:

```
library(glmmTMB)
```

```
Warning: package 'glmmTMB' was built under R version 4.3.3
```

```
#| warning: False
m1<- glmmTMB(Hg~size + (1|region), data=HgDat_df)

summary(m1)
```

```
 Family: gaussian  ( identity )
Formula:          Hg ~ size + (1 | region)
Data: HgDat_df

     AIC      BIC   logLik deviance df.resid
  -155.0   -145.3     81.5   -163.0       79

Random effects:

Conditional model:
 Groups   Name        Variance Std.Dev.
 region   (Intercept) 0.072972 0.27013
 Residual             0.006309 0.07943
Number of obs: 83, groups:  region, 4

Dispersion estimate for gaussian family (sigma^2): 0.00631
```

```
Conditional model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 0.367381   0.139945   2.625  0.00866 **
size        0.018812   0.000843  22.315  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

> **Note**
>
> Are you getting an error about NaN's found in the data? This is probably caused by the "Matrix" package that followign an upate started adding NAN's due to a bug. The best solution would be to reinstall the Matrix package independently

Let's remember our simulated data. Was the model good enough?

$$Hg_{i,j} = 0.5 + 0.18(x_{i,j}) + \gamma_j + \epsilon$$

Variance individuals and gamma

```
0.08^2
```

```
[1] 0.0064
```

```
0.25^2
```

```
[1] 0.0625
```

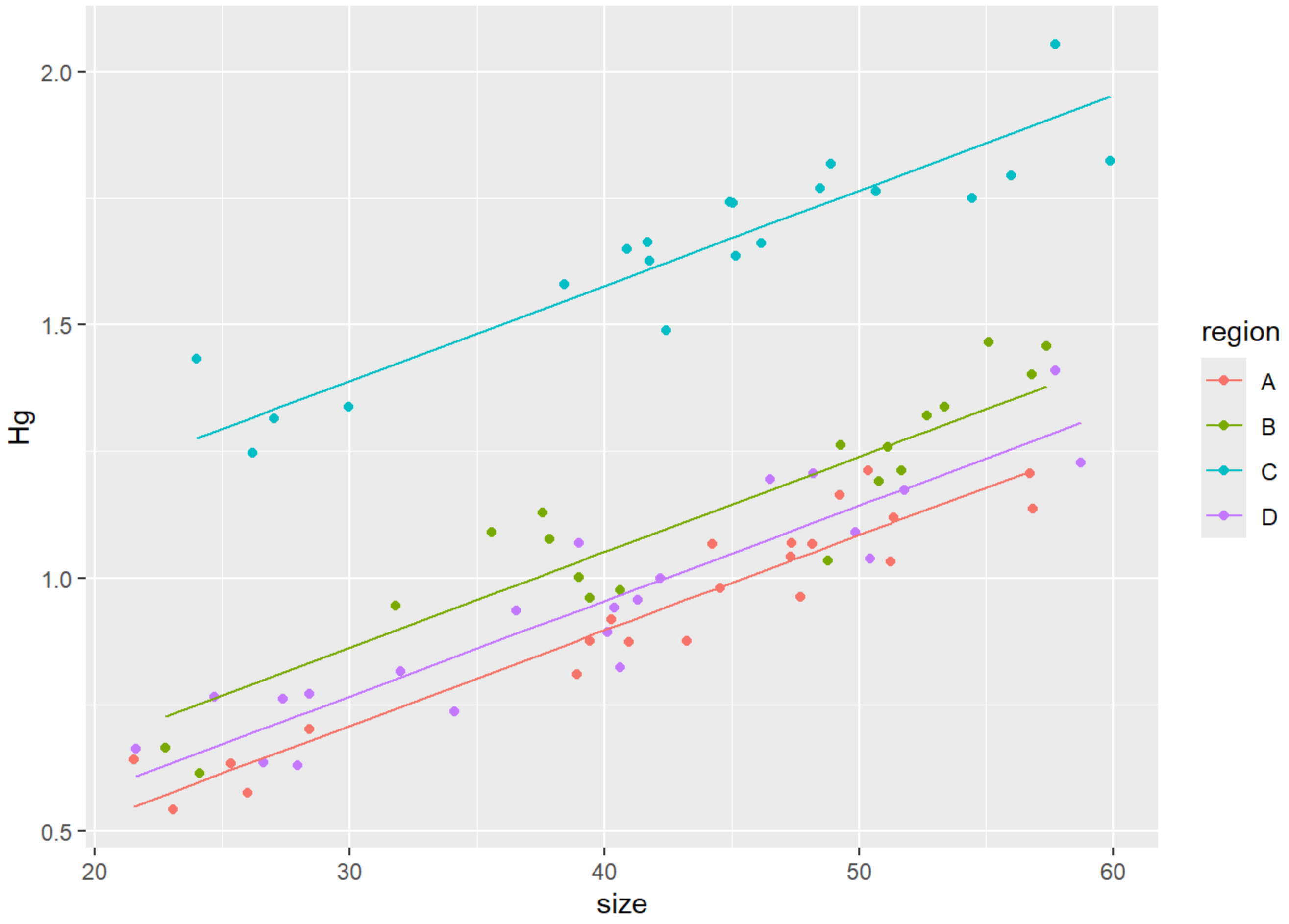
Predictions and plot

Make a prediction data frame

```
preddata <- HgDat_df
preddata$predHg <- predict(m1, preddata)
```

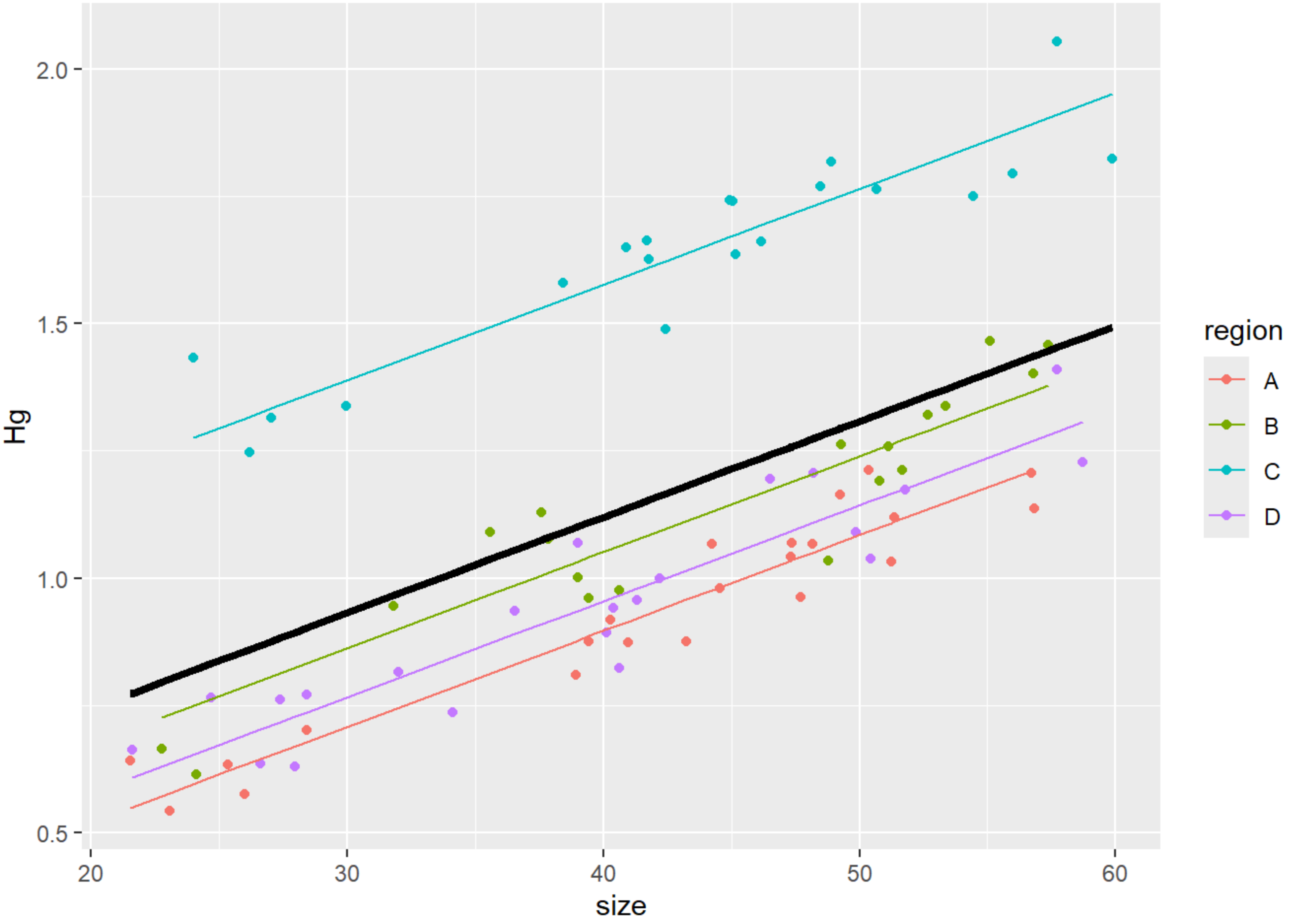
And plot the model

```
library(ggplot2)
ggplot(data=HgDat_df, aes(x=size, y=Hg, col=region)) +
    geom_point() +
    geom_line(data=preddata, aes(x=size, y=predHg, col=region))
```



How to obtain a population “line”

```
preddata <- HgDat_df
preddata$predHg <- predict(m1, preddata)
preddata$predHg_population <- predict(m1, preddata, re.form=~0)

ggplot(data=HgDat_df, aes(x=size, y=Hg, col=region)) +
    geom_point() +
    geom_line(data=preddata, aes(x=size, y=predHg, col=region))+
  geom_line(data=preddata, aes(x=size, y=predHg_population),                col='bla
```



Now, let’s repeat this analysis, but in this case, let’s assume the intercept is fixed, and the slope is random

Question 1: points:10

Simulate the same study, but in this case, there is a fixed intercept and a random slope. 1) Present the equation you used for the simulation (similar to the equation I showed in this document). 2) Run the mixed-effects model, 3) Report whether the model output was a good representation of reality, and 4) plot your model

We have the model with fixed intercept and random slope

$$Hg_{i,j} = 0.5 + 0.18(x_{i,j} + \psi_j) + \epsilon$$

```
psi <- rnorm(n=4,mean=0,sd=0.25)
gamma <- 0
N<- round(rnorm(4,20,2))

HgDat<-list()

for(i in 1:4){
  x<-runif(N[i],20,60)
  y<-0.5 + (0.018+psi[i])*x + gamma + rnorm(N[i],0,0.08)
  Region<-rep(LETTERS[i],N[i])
```

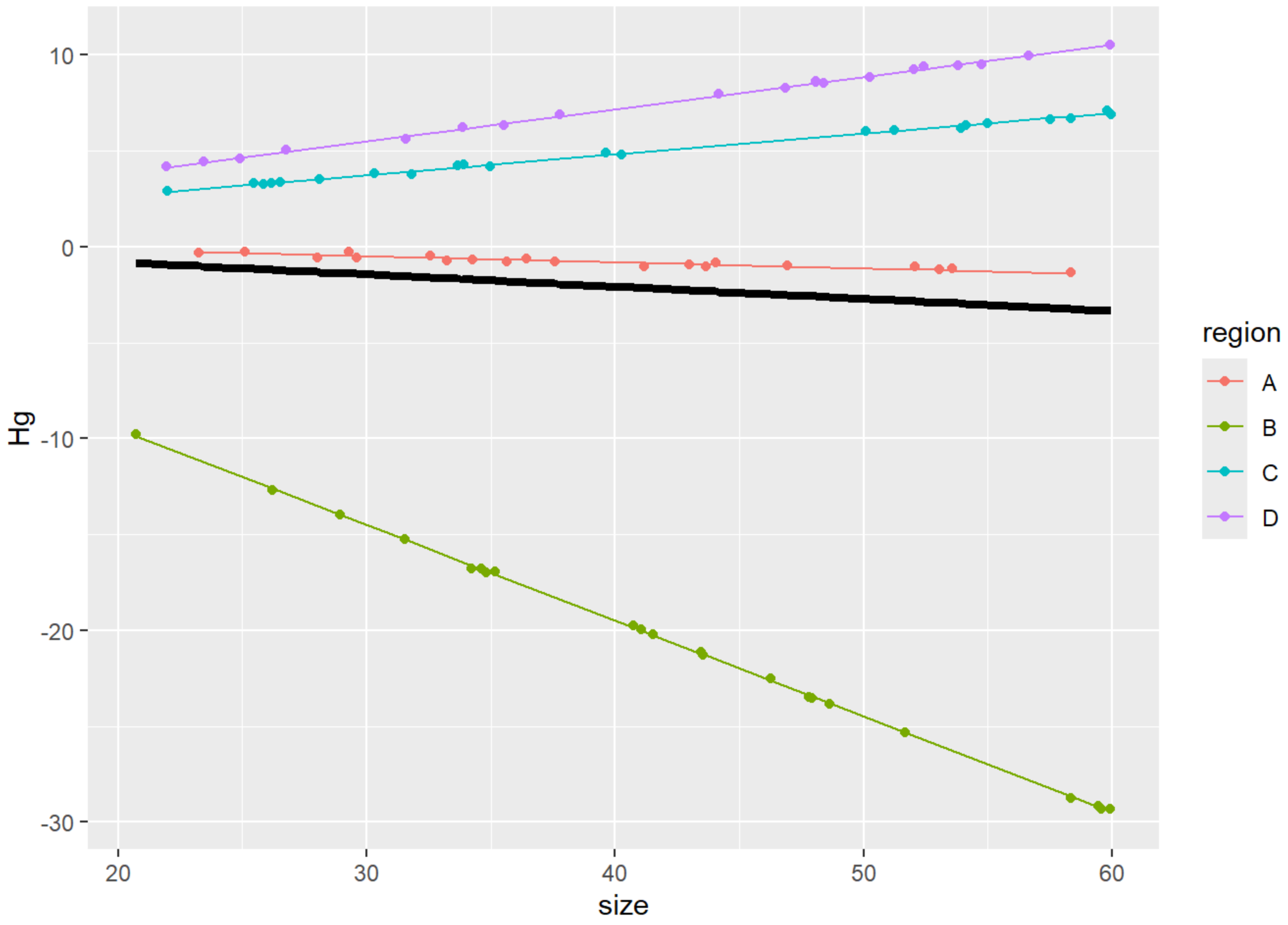
```
            HgDat[[i]]<-data.frame(size=x,Hg=y,region=Region)
          }

          HgDat_df<-bind_rows(HgDat)
          HgDat_df$region<-as.factor(HgDat_df$region)

          kbl(HgDat_df, col.names = gsub("[.]", " ", names(HgDat_df)))%>%
            kable_paper() %>%
            scroll_box(width = "500px", height = "200px")
```

size	Hg	region
43.64647	-1.0158644	A
53.57941	-1.1305466	A
52.06707	-1.0043637	A
32.56139	-0.4387404	A
33.23455	-0.7273662	A

```
          m2<- glmmTMB(Hg~size + (0+size|region), data=HgDat_df)


          summary(m2)
```

```
 Family: gaussian  ( identity )
Formula:          Hg ~ size + (0 + size | region)
Data: HgDat_df

     AIC      BIC   logLik deviance df.resid
  -103.5    -93.8     55.8   -111.5       80

Random effects:

Conditional model:
 Groups   Name Variance Std.Dev.
 region   size 0.06846  0.2616
 Residual      0.00850  0.0922
Number of obs: 84, groups:  region, 4

Dispersion estimate for gaussian family (sigma^2): 0.0085

Conditional model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.47663    0.03757  12.688   <2e-16 ***
size        -0.06358    0.13082  -0.486    0.627
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
preddata <- HgDat_df
preddata$predHg <- predict(m2, preddata)

preddata <- HgDat_df
preddata$predHg <- predict(m2, preddata)
preddata$predHg_population <- predict(m2, preddata, re.form=~0)

ggplot(data=HgDat_df, aes(x=size, y=Hg, col=region)) +
    geom_point() +
    geom_line(data=preddata, aes(x=size, y=predHg, col=region))+
  geom_line(data=preddata, aes(x=size, y=predHg_population),col='black',linewidth=1.5)
```



Answer 1.

$$Hg_{i,j} = 0.5 + 0.18(x_{i,j} + \psi_j) + \epsilon$$

The p-value for the fixed intercept is significant, but the p-value for the random slope is $0.459 > 0.05$, which is not significant. Thus, this model may not be the best fit.

Question 2. Points: 10

> Simulate the same study, but in this case, there is a random intercept and a random slope. 1) Present the equation you used for the simulation (similar to the equation I showed in this document). 2) Run the mixed-effects model, 3) Report whether the model output was a good representation of reality, and 4) plot your model

```
psi <- rnorm(n=4,mean=0,sd=0.25)
gamma <- rnorm(n=4,mean=0,sd=0.25)
N<- round(rnorm(4,20,2))

HgDat<-list()


for(i in 1:4){
  x<-runif(N[i],20,60)
  y<-0.5 + (0.018+psi[i])*x + gamma[i] + rnorm(N[i],0,0.08)
  Region<-rep(LETTERS[i],N[i])
  HgDat[[i]]<-data.frame(size=x,Hg=y,region=Region)
}

HgDat_df<-bind_rows(HgDat)
HgDat_df$region<-as.factor(HgDat_df$region)

kbl(HgDat_df, col.names = gsub("[.]", " ", names(HgDat_df)))%>%
  kable_paper() %>%
  scroll_box(width = "500px", height = "200px")
```

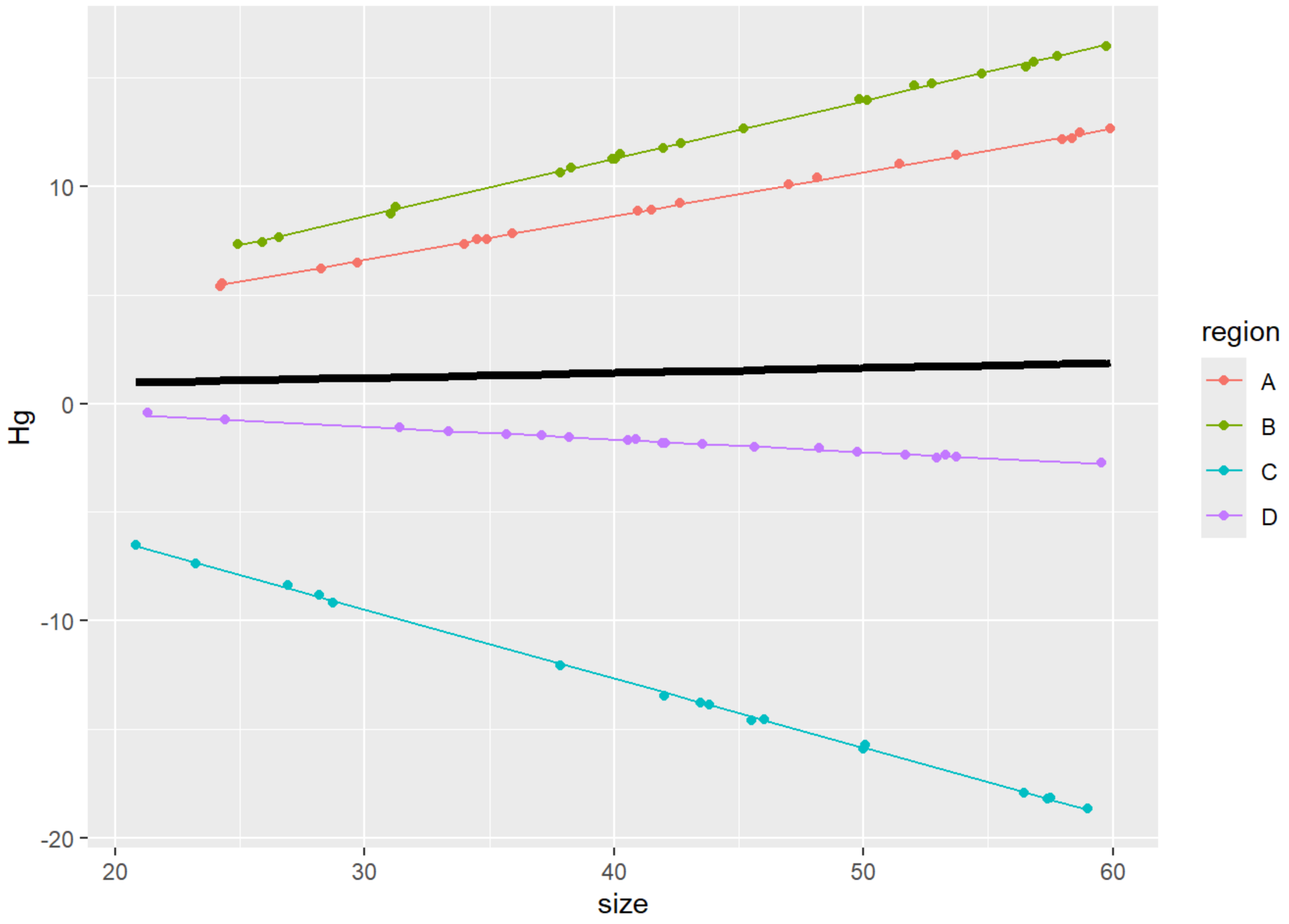
size	Hg	region
59.88629	12.6480849	A
28.23818	6.2206194	A
41.48334	8.9395170	A
33.97968	7.3555729	A
34.49305	7.5639979	A

```
m3<- glmmTMB(Hg~size + (1 + size|region), data=HgDat_df)

summary(m3)
```

```
 Family: gaussian  ( identity )
Formula:          Hg ~ size + (1 + size | region)
Data: HgDat_df

     AIC      BIC   logLik deviance df.resid
  -103.9    -89.7     57.9   -115.9       72

Random effects:
```

```
Conditional model:
 Groups   Name        Variance Std.Dev. Corr
 region   (Intercept) 0.059692 0.24432
          size        0.053696 0.23172  0.75
 Residual             0.006305 0.07941
Number of obs: 78, groups:  region, 4

Dispersion estimate for gaussian family (sigma^2): 0.00631

Conditional model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.49550    0.12736   3.890   0.0001 ***
size         0.02268    0.11587   0.196   0.8448
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
          preddata <- HgDat_df
          preddata$predHg <- predict(m3, preddata)
          preddata$predHg_population <- predict(m3, preddata, re.form=~0)

          ggplot(data=HgDat_df, aes(x=size, y=Hg, col=region)) +
              geom_point() +
              geom_line(data=preddata, aes(x=size, y=predHg, col=region))+
            geom_line(data=preddata, aes(x=size, y=predHg_population),                    col='bla
```



Answer 2

$$Hg_{i,j} = 0.5 + 0.18(x_{i,j} + \psi_j) + \gamma_j + \epsilon$$

The p-value for the fixed intercept is significant, but the p-value for the random slope is $0.872 > 0.05$, which is not significant. Thus, this model may not be the best fit.

Finally, let’s look at what happens with a low N.

You will repeat question 2, but using the following to obtain your sample:

Question 3

```
psi <- rnorm(n=4,mean=0,sd=0.25)
gamma <- rnorm(n=4,mean=0,sd=0.25)
N<- round(rnorm(4,150,20))

HgDat<-list()


for(i in 1:4){
  x<-runif(N[i],20,60)
  y<-0.5 + (0.018+psi[i])*x + gamma[i] + rnorm(N[i],0,0.08)
  Region<-rep(LETTERS[i],N[i])
  HgDat[[i]]<-data.frame(size=x,Hg=y,region=Region)
}

HgDat_df<-bind_rows(HgDat)
HgDat_df$region<-as.factor(HgDat_df$region)

kbl(HgDat_df, col.names = gsub("[.]", " ", names(HgDat_df)))%>%
  kable_paper() %>%
  scroll_box(width = "500px", height = "200px")
```

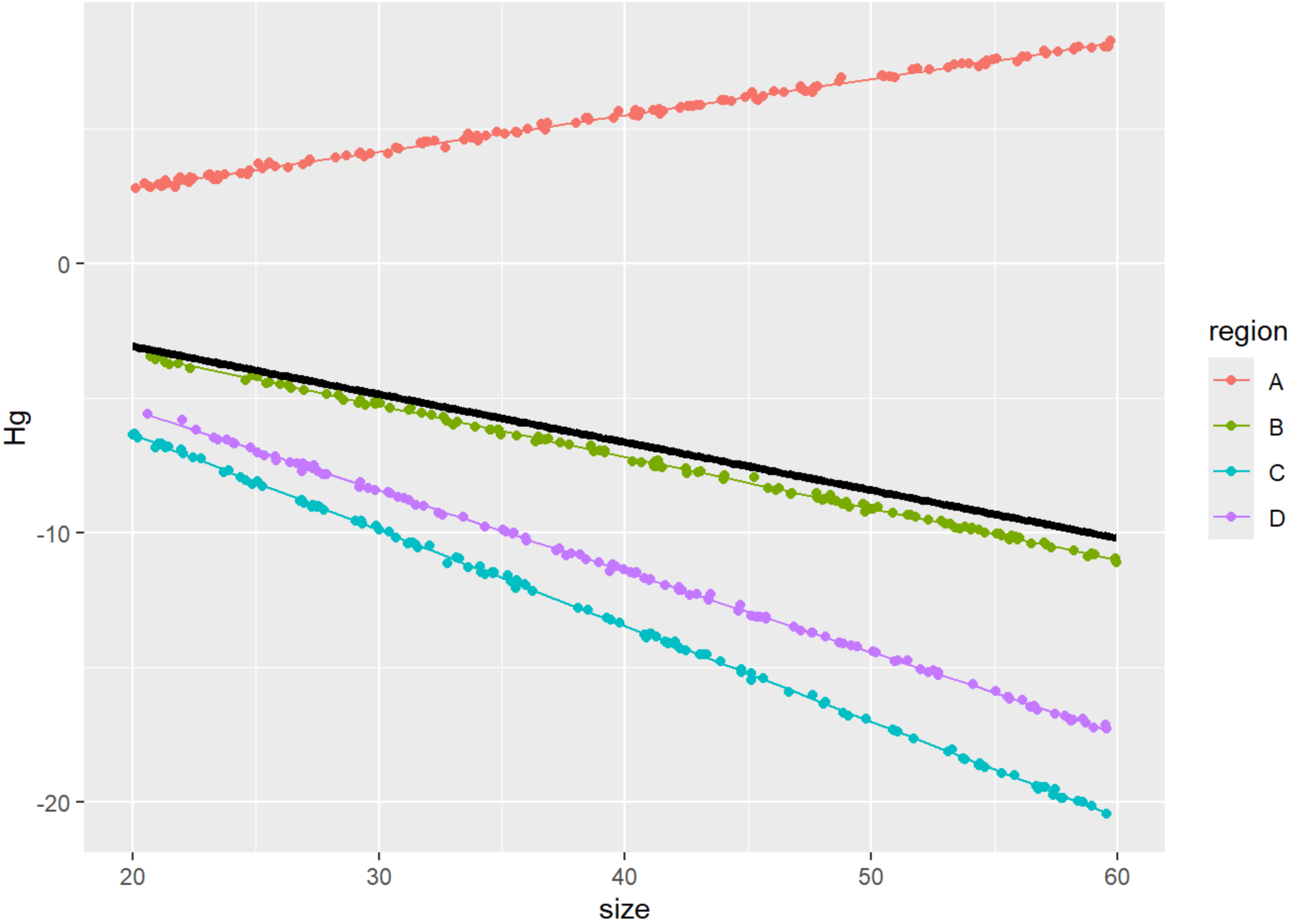
size	Hg	region
42.74688	5.827256	A
55.92876	7.497917	A
45.61966	6.208056	A
57.60299	7.854233	A
20.63423	2.867237	A

```
  m4<- glmmTMB(Hg~size + (1+size|region), data=HgDat_df)

  summary(m4)
```

```
 Family: gaussian  ( identity )
Formula:          Hg ~ size + (1 + size | region)
Data: HgDat_df

     AIC      BIC   logLik deviance df.resid
 -1087.5  -1062.1    549.8  -1099.5      511

Random effects:

Conditional model:
 Groups   Name        Variance Std.Dev. Corr
 region   (Intercept) 0.056925 0.2386
          size        0.036167 0.1902   -0.98
 Residual             0.006194 0.0787
Number of obs: 517, groups:  region, 4

Dispersion estimate for gaussian family (sigma^2): 0.00619

Conditional model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.48059    0.11995   4.007 6.16e-05 ***
```

```
size         -0.17806    0.09509  -1.873   0.0611 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
preddata <- HgDat_df
preddata$predHg <- predict(m4, preddata)
preddata$predHg_population <- predict(m4, preddata, re.form=~0)

ggplot(data=HgDat_df, aes(x=size, y=Hg, col=region)) +
    geom_point() +
    geom_line(data=preddata, aes(x=size, y=predHg, col=region))+
  geom_line(data=preddata, aes(x=size, y=predHg_population),                    col='bla
```



Answer 3

The intercept and slope significant p-values. This appears to be the best model.

Question 3. Points:4

Re-run the code you used for question 2, but using the new N. Report any differences that you see after running the model with a low N

Total points: 24

Resources:

ggplot cheat sheet: https://github.com/rstudio/cheatsheets/blob/main/data-visualization.pdf