Assignment6\_bayes

mkhulekeli Nkosi 2017159092

2025-05-24

# 1.

The residual error is caused by the different groups that will do presentation and by the different groups of assessors. Also not having enough variables to explain the the final mark.

# 2.

There is always a factor that influences an outcome but according to the above assumptions i would say in this case they are enough for the average assessor mark to be correct on average.

# 3.

data<- readxl::read\_xlsx("BayesAssignment6of2025.xlsx")  
summary(data)

## Group LecturerA LecturerB LecturerC   
## Length:15 Min. :60.00 Min. :49.00 Min. :60.00   
## Class :character 1st Qu.:72.00 1st Qu.:62.00 1st Qu.:63.50   
## Mode :character Median :74.00 Median :68.00 Median :67.50   
## Mean :74.14 Mean :65.33 Mean :69.75   
## 3rd Qu.:76.75 3rd Qu.:70.00 3rd Qu.:77.50   
## Max. :88.00 Max. :82.00 Max. :85.00   
## NA's :1 NA's :6 NA's :3   
## LecturerD LecturerE LecturerF LecturerG Proposal   
## Min. :60.00 Min. :52.00 Min. :53.00 Min. :60.0 Min. :57.00   
## 1st Qu.:68.00 1st Qu.:61.50 1st Qu.:71.75 1st Qu.:64.5 1st Qu.:63.50   
## Median :70.00 Median :68.00 Median :78.00 Median :69.5 Median :74.00   
## Mean :70.50 Mean :67.71 Mean :72.25 Mean :68.0 Mean :71.13   
## 3rd Qu.:76.25 3rd Qu.:76.00 3rd Qu.:78.50 3rd Qu.:73.0 3rd Qu.:78.00   
## Max. :78.00 Max. :79.00 Max. :80.00 Max. :73.0 Max. :84.00   
## NA's :5 NA's :8 NA's :11 NA's :11   
## Literature Quiz Interview   
## Min. :55.0 Min. :48.00 Min. :49.00   
## 1st Qu.:65.5 1st Qu.:66.50 1st Qu.:64.00   
## Median :69.0 Median :75.00 Median :71.00   
## Mean :69.4 Mean :72.47 Mean :68.13   
## 3rd Qu.:74.5 3rd Qu.:80.00 3rd Qu.:72.00   
## Max. :91.0 Max. :85.00 Max. :77.00   
##

sapply(data, class)

## Group LecturerA LecturerB LecturerC LecturerD LecturerE   
## "character" "numeric" "numeric" "numeric" "numeric" "numeric"   
## LecturerF LecturerG Proposal Literature Quiz Interview   
## "numeric" "numeric" "numeric" "numeric" "numeric" "numeric"

colSums(is.na(data))

## Group LecturerA LecturerB LecturerC LecturerD LecturerE LecturerF   
## 0 1 6 3 5 8 11   
## LecturerG Proposal Literature Quiz Interview   
## 11 0 0 0 0

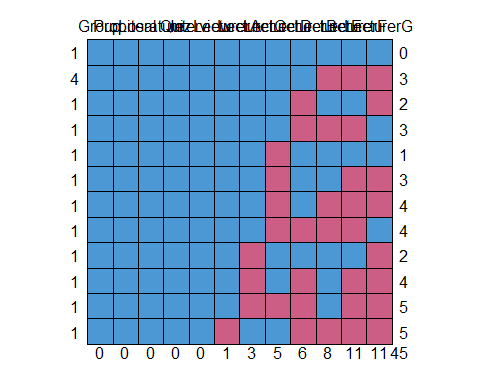
library(mice)

##   
## Attaching package: 'mice'

## The following object is masked from 'package:stats':  
##   
## filter

## The following objects are masked from 'package:base':  
##   
## cbind, rbind

md.pattern(data)



## Group Proposal Literature Quiz Interview LecturerA LecturerC LecturerD  
## 1 1 1 1 1 1 1 1 1  
## 4 1 1 1 1 1 1 1 1  
## 1 1 1 1 1 1 1 1 1  
## 1 1 1 1 1 1 1 1 1  
## 1 1 1 1 1 1 1 1 0  
## 1 1 1 1 1 1 1 1 0  
## 1 1 1 1 1 1 1 1 0  
## 1 1 1 1 1 1 1 1 0  
## 1 1 1 1 1 1 1 0 1  
## 1 1 1 1 1 1 1 0 1  
## 1 1 1 1 1 1 1 0 0  
## 1 1 1 1 1 1 0 1 1  
## 0 0 0 0 0 1 3 5  
## LecturerB LecturerE LecturerF LecturerG   
## 1 1 1 1 1 0  
## 4 1 0 0 0 3  
## 1 0 1 1 0 2  
## 1 0 0 0 1 3  
## 1 1 1 1 1 1  
## 1 1 1 0 0 3  
## 1 1 0 0 0 4  
## 1 0 0 0 1 4  
## 1 1 1 1 0 2  
## 1 0 1 0 0 4  
## 1 0 1 0 0 5  
## 1 0 0 0 0 5  
## 6 8 11 11 45

All the columns are numeric excerpt for the group column. And 45 missing values are observed in total. The missingness patterns given the visualisations on can conclude that the middingness is MAR because the missin values in lecturer E are missing when lecture F And G are missing missingness is dependent on the two variables also lecture D only one is not dependent on G and F.

# 4.

library(tidyr)  
  
long\_data <- pivot\_longer(data, cols= c(LecturerA,LecturerB,LecturerC,LecturerD,LecturerE,LecturerF,LecturerG) ,names\_to = c("Lecturer"), values\_to = "Score")  
  
new\_data<- na.omit(long\_data)  
unique(new\_data$Lecturer)

## [1] "LecturerA" "LecturerC" "LecturerD" "LecturerG" "LecturerB" "LecturerE"  
## [7] "LecturerF"

# 5.

In our case the group of students is our Fixed effect because we not interested in how the next possible group will affect the final mark, with the lecturer as the random effect each group will experience the lecture effect and one would like to know how a different lecturer not included in this fit will grade each group.(read slides)

# 6.

The prior for the group intercepts, and intercept is a normal prior and also the sigma as a cauchy.

library(brms)

## Loading required package: Rcpp

## Loading 'brms' package (version 2.22.0). Useful instructions  
## can be found by typing help('brms'). A more detailed introduction  
## to the package is available through vignette('brms\_overview').

##   
## Attaching package: 'brms'

## The following object is masked from 'package:stats':  
##   
## ar

model <- brm(  
 formula = Score ~ Group + (1 | Lecturer),  
 data = new\_data,  
 prior = c(  
 set\_prior("normal(0, 10)", class = "b"),   
 set\_prior("normal(0, 5)", class = "Intercept"),  
 set\_prior("cauchy(0, 5)", class = "sd")   
 ),  
 iter = 5000  
)

## Compiling Stan program...

## Start sampling

summary(model)

## Family: gaussian   
## Links: mu = identity; sigma = identity   
## Formula: Score ~ Group + (1 | Lecturer)   
## Data: new\_data (Number of observations: 60)   
## Draws: 4 chains, each with iter = 5000; warmup = 2500; thin = 1;  
## total post-warmup draws = 10000  
##   
## Multilevel Hyperparameters:  
## ~Lecturer (Number of levels: 7)   
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## sd(Intercept) 69.33 21.09 40.84 120.82 1.00 1031 2155  
##   
## Regression Coefficients:  
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## Intercept 0.55 5.40 -9.88 11.29 1.00 7095 7001  
## GroupGroup10 7.30 3.18 1.05 13.60 1.00 5527 6547  
## GroupGroup11 6.19 3.13 -0.02 12.23 1.00 4615 6502  
## GroupGroup12 -2.59 3.99 -10.34 5.07 1.00 7141 6194  
## GroupGroup13 6.65 3.16 0.38 12.68 1.00 5480 6823  
## GroupGroup14 2.41 2.98 -3.36 8.23 1.00 5303 7006  
## GroupGroup15 -11.89 4.02 -19.73 -4.01 1.00 7601 7109  
## GroupGroup2 -0.41 3.51 -7.30 6.36 1.00 6098 6427  
## GroupGroup3 14.59 3.19 8.29 20.78 1.00 5312 5941  
## GroupGroup4 -3.74 2.82 -9.39 1.78 1.00 4767 5913  
## GroupGroup5 8.46 2.72 3.04 13.74 1.00 4616 5778  
## GroupGroup6 4.55 3.21 -1.79 10.71 1.00 5620 6590  
## GroupGroup7 4.17 3.51 -2.64 11.11 1.00 6602 6853  
## GroupGroup8 -10.65 3.01 -16.60 -4.85 1.00 5001 6363  
## GroupGroup9 4.18 3.58 -3.09 11.15 1.00 6175 6509  
##   
## Further Distributional Parameters:  
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## sigma 5.23 0.60 4.22 6.54 1.00 5611 6455  
##   
## Draws were sampled using sampling(NUTS). For each parameter, Bulk\_ESS  
## and Tail\_ESS are effective sample size measures, and Rhat is the potential  
## scale reduction factor on split chains (at convergence, Rhat = 1).

# 7.

fixed\_effects <- fixef(model, summary = TRUE)   
fit<-fitted(model)  
pred\_vals <- predict(model, summary = TRUE)  
data\_est<- cbind(new\_data$Group,fit,pred\_vals)  
data\_est<- data\_est[,c(-3,-6,-7)]  
colnames(data\_est) <- c("Groups","estimates","CI2.5", "CI97.5", "PI2.5","PI97.5")  
print(data\_est)

## Groups estimates CI2.5 CI97.5   
## [1,] "Group1" "71.8272780567707" "67.3816959338289" "76.4112142948512"  
## [2,] "Group1" "65.7530073826222" "61.1340245567454" "70.3879099284146"  
## [3,] "Group1" "67.0585095799337" "62.2179243556079" "71.9085978604208"  
## [4,] "Group1" "65.6705623144752" "59.4064345126611" "71.8446284371244"  
## [5,] "Group2" "71.4208303568521" "65.0027090835831" "77.5059755235686"  
## [6,] "Group2" "61.2094442346895" "54.8867991241413" "67.5396776182671"  
## [7,] "Group2" "65.3465596827036" "59.1047599499906" "71.5743184187067"  
## [8,] "Group3" "86.4172705610843" "80.7260450650427" "92.0439757142154"  
## [9,] "Group3" "76.2058844389217" "70.2311466134201" "82.0572419537057"  
## [10,] "Group3" "80.3429998869358" "74.5937957207956" "85.9971111088454"  
## [11,] "Group3" "81.6485020842473" "75.9203752195219" "87.559335199664"   
## [12,] "Group4" "68.0883283980066" "63.0886187448433" "73.0234567657312"  
## [13,] "Group4" "57.876942275844" "52.4857880135655" "63.273016555994"   
## [14,] "Group4" "62.0140577238581" "56.9641910388435" "67.1495746696826"  
## [15,] "Group4" "64.4781733370902" "58.8507950932111" "70.0650308301038"  
## [16,] "Group4" "69.2742624782087" "62.9145760395773" "75.721021775319"   
## [17,] "Group4" "61.9316126557111" "55.5175945351252" "68.3083534677343"  
## [18,] "Group5" "80.2874769805864" "75.6229885358906" "84.9487996910524"  
## [19,] "Group5" "70.0760908584238" "64.8656681201077" "75.2194785458619"  
## [20,] "Group5" "74.2132063064379" "69.2741979720579" "79.0941189983148"  
## [21,] "Group5" "75.5187085037494" "70.4088081946458" "80.5985805501959"  
## [22,] "Group5" "76.6773219196699" "71.2892428751487" "82.0558730428591"  
## [23,] "Group5" "81.4734110607885" "75.1224599664591" "87.6593396661382"  
## [24,] "Group5" "74.1307612382909" "67.7582583398329" "80.527614810006"   
## [25,] "Group6" "76.375284456637" "70.6003907133281" "82.0217298729476"  
## [26,] "Group6" "66.1638983344744" "60.3862381416535" "72.042573552306"   
## [27,] "Group6" "70.3010137824885" "64.5548896359787" "75.952396185621"   
## [28,] "Group6" "71.6065159798" "65.6952579626673" "77.3350560709897"  
## [29,] "Group7" "75.9984937441552" "69.4482304143084" "82.49735871699"   
## [30,] "Group7" "69.9242230700067" "63.3371327254529" "76.3059887769748"  
## [31,] "Group7" "69.8417780018598" "62.6067410468639" "76.9774789921705"  
## [32,] "Group8" "61.1752079519541" "56.040594034897" "66.4472817565804"  
## [33,] "Group8" "50.9638218297915" "45.4864541674483" "56.6310717439124"  
## [34,] "Group8" "56.4064394751171" "50.8662107537131" "61.9628315050973"  
## [35,] "Group8" "57.5650528910377" "51.7004528161433" "63.4417788379237"  
## [36,] "Group8" "62.3611420321563" "55.9321631886962" "68.9170444464736"  
## [37,] "Group9" "76.0062641887419" "69.5958324069556" "82.4551120500761"  
## [38,] "Group9" "71.2374957119049" "64.4626513717561" "77.6906945742784"  
## [39,] "Group9" "72.3961091278255" "65.599554954103" "79.1489782937374"  
## [40,] "Group10" "79.1274661003822" "73.4945554755566" "84.6533574848276"  
## [41,] "Group10" "68.9160799782196" "63.0762009066233" "74.8385060432002"  
## [42,] "Group10" "73.0531954262337" "67.2815550552564" "78.6629353654037"  
## [43,] "Group10" "75.5173110394658" "69.3821620171653" "81.4785890891563"  
## [44,] "Group11" "78.0201597755601" "72.538452987675" "83.501180185899"   
## [45,] "Group11" "67.8087736533975" "62.0237200258789" "73.581278748563"   
## [46,] "Group11" "71.9458891014116" "66.3293294975236" "77.4095346074365"  
## [47,] "Group11" "73.2513912987231" "67.596040126721" "79.0825536571339"  
## [48,] "Group12" "63.1627854784364" "56.0022364125137" "70.5742067622025"  
## [49,] "Group12" "64.4682876757479" "57.2297135751689" "71.9031227757326"  
## [50,] "Group13" "78.480260502871" "72.766595133667" "84.1536689462767"  
## [51,] "Group13" "68.2688743807084" "62.5146319104754" "74.0707345315984"  
## [52,] "Group13" "72.4059898287225" "66.704336463304" "78.0571982865767"  
## [53,] "Group13" "73.711492026034" "67.7903880933131" "79.4895172344469"  
## [54,] "Group14" "74.2346100147019" "68.9564877987208" "79.5728542140803"  
## [55,] "Group14" "68.1603393405534" "62.8708645702476" "73.6684991153242"  
## [56,] "Group14" "69.4658415378649" "63.9512725324973" "74.9456333043901"  
## [57,] "Group14" "70.6244549537854" "64.6156820600251" "76.6406167213093"  
## [58,] "Group14" "75.420544094904" "68.8842012956683" "81.8760918931131"  
## [59,] "Group15" "59.936884350606" "52.6339027467024" "67.4675222261554"  
## [60,] "Group15" "56.3267292896895" "48.9360511809967" "64.0786497938627"  
## PI2.5 PI97.5   
## [1,] "60.4065279581106" "83.137296788773"   
## [2,] "54.4572791992896" "77.0418335949967"  
## [3,] "55.9494672407719" "78.6774846638292"  
## [4,] "53.8071728666516" "77.7275393392395"  
## [5,] "59.4430049099204" "83.4082858665242"  
## [6,] "49.1111869951962" "73.2213278028708"  
## [7,] "53.2530567686497" "77.6379947229517"  
## [8,] "74.4587092459479" "97.986456228298"   
## [9,] "64.2076266693005" "87.9725112422199"  
## [10,] "68.5004462436281" "92.2172603225143"  
## [11,] "69.729547965189" "93.5626984818983"  
## [12,] "56.6233671539857" "79.5763398678036"  
## [13,] "46.3861872896246" "69.5648314228824"  
## [14,] "50.376639470849" "73.7638782385496"  
## [15,] "52.7626926687926" "76.2194996834338"  
## [16,] "57.062781067509" "81.5169053053449"  
## [17,] "49.5878652967018" "74.1298691349211"  
## [18,] "68.7697157820018" "91.8654357110091"  
## [19,] "58.1319337852532" "81.5557568844457"  
## [20,] "62.4552780366887" "85.780737347761"   
## [21,] "64.0939421589276" "86.9749013064858"  
## [22,] "65.0161851971176" "88.6637371448674"  
## [23,] "69.1241564648905" "93.5198845634234"  
## [24,] "61.733117173898" "86.2642988234008"  
## [25,] "64.7107109003848" "88.3114496328109"  
## [26,] "54.4401973262443" "78.0324202206171"  
## [27,] "58.5113426879855" "82.1063370617349"  
## [28,] "59.4562014138564" "83.2728023546783"  
## [29,] "63.6953883734381" "88.0606586308747"  
## [30,] "57.482245054093" "81.9912793021463"  
## [31,] "57.5152134845802" "82.2702768205298"  
## [32,] "49.5561795055253" "72.7662309189912"  
## [33,] "39.2229626666422" "62.8916005466645"  
## [34,] "44.8632095666826" "68.1796236024637"  
## [35,] "45.7092020293289" "69.4316540037957"  
## [36,] "50.0736091958088" "74.589076719306"   
## [37,] "63.464082599961" "88.2539888337848"  
## [38,] "59.0306585853351" "83.1020208256652"  
## [39,] "60.0739637370645" "84.5418967438473"  
## [40,] "67.1500063166837" "90.7513358295395"  
## [41,] "57.0659357871602" "80.7338002362295"  
## [42,] "61.0701395132688" "85.0032309208408"  
## [43,] "63.4055372855378" "87.796045925465"   
## [44,] "66.3445468862034" "89.6938408414282"  
## [45,] "56.0582577793975" "79.6389205063761"  
## [46,] "60.0368097510303" "83.7127280212161"  
## [47,] "61.272601008131" "84.7055432644049"  
## [48,] "50.5214375759365" "75.9759590711846"  
## [49,] "52.1048382587593" "77.0912589839781"  
## [50,] "66.699840594653" "90.0425792479024"  
## [51,] "56.2188021837574" "80.0533513159761"  
## [52,] "60.7301688286684" "84.179398732093"   
## [53,] "61.8768315674836" "85.5291039935551"  
## [54,] "62.7272960385888" "85.6162629450797"  
## [55,] "56.3081315472428" "79.9272093144586"  
## [56,] "58.0078219212207" "81.214600545324"   
## [57,] "59.1274963239031" "82.6644522935205"  
## [58,] "63.2714377286066" "87.6313678129049"  
## [59,] "47.1742942448166" "72.7441379354684"  
## [60,] "43.7744764183804" "68.8416411606077"

# 8.

lecturer B is least biased

library(dplyr)

##   
## 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

assessor\_biases <- ranef(model)$Lecturer %>%  
 as.data.frame() %>%  
 arrange(Estimate.Intercept)  
  
least\_biased <- assessor\_biases[which.min(abs(assessor\_biases$Estimate.Intercept)), ]  
least\_biased

## Estimate.Intercept Est.Error.Intercept Q2.5.Intercept Q97.5.Intercept  
## LecturerB 61.06767 5.37193 50.56455 71.55279

# 9.