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

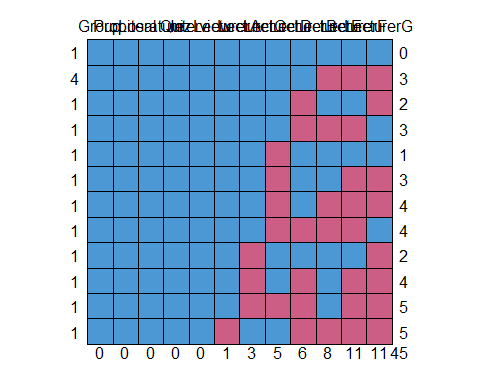
library(mice)

##   
## 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

## Warning: There were 2 divergent transitions after warmup. See  
## https://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup  
## to find out why this is a problem and how to eliminate them.

## Warning: Examine the pairs() plot to diagnose sampling problems

summary(model)

## Warning: There were 2 divergent transitions after warmup. Increasing  
## adapt\_delta above 0.8 may help. See  
## http://mc-stan.org/misc/warnings.html#divergent-transitions-after-warmup

## 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) 69.65 21.77 41.30 125.38 1.00 1334 2262  
##   
## Regression Coefficients:  
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## Intercept 0.44 5.36 -9.93 11.12 1.00 7714 6874  
## GroupGroup10 7.41 3.15 1.20 13.53 1.00 4837 6374  
## GroupGroup11 6.27 3.15 0.08 12.30 1.00 5064 7068  
## GroupGroup12 -2.63 3.96 -10.33 5.15 1.00 7158 6601  
## GroupGroup13 6.74 3.13 0.48 12.85 1.00 4621 6584  
## GroupGroup14 2.50 3.01 -3.39 8.40 1.00 4564 6231  
## GroupGroup15 -11.89 4.04 -19.76 -4.09 1.00 6157 6387  
## GroupGroup2 -0.35 3.46 -7.27 6.33 1.00 5514 6646  
## GroupGroup3 14.66 3.15 8.20 20.65 1.00 4713 5528  
## GroupGroup4 -3.70 2.88 -9.57 1.97 1.00 4515 5530  
## GroupGroup5 8.51 2.72 3.13 13.75 1.00 4301 6098  
## GroupGroup6 4.63 3.14 -1.61 10.65 1.00 4687 6281  
## GroupGroup7 4.18 3.43 -2.67 10.82 1.00 6214 6790  
## GroupGroup8 -10.59 2.99 -16.59 -4.81 1.00 4574 6261  
## GroupGroup9 4.23 3.47 -2.56 10.94 1.00 5661 6376  
##   
## Further Distributional Parameters:  
## Estimate Est.Error l-95% CI u-95% CI Rhat Bulk\_ESS Tail\_ESS  
## sigma 5.22 0.59 4.21 6.54 1.00 6023 7158  
##   
## 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")  
data.frame(data\_est)

## Groups estimates CI2.5 CI97.5 PI2.5  
## 1 Group1 71.7899155475059 67.4779908906494 76.3451200629799 60.7695010927813  
## 2 Group1 65.6816786038162 61.0567167908002 70.330136351063 54.7700927351708  
## 3 Group1 67.0492560343489 62.2393147758662 71.978592243546 55.8727229083264  
## 4 Group1 65.6225182316326 59.5027028963555 71.9549949871398 53.7200969917084  
## 5 Group2 71.4394482053501 65.1393074578692 77.7911558567287 59.269551092472  
## 6 Group2 61.2198614263584 54.8442001140935 67.5689710065727 48.7600625589985  
## 7 Group2 65.3312112616603 58.9760363523655 71.6050928995998 53.399875311066  
## 8 Group3 86.4548996596001 80.857661516971 92.0309274633249 74.71408734517  
## 9 Group3 76.2353128806084 70.3286992357793 82.0138271768776 64.460218702359  
## 10 Group3 80.3466627159103 74.6029449027804 85.9750025298642 68.3267055897392  
## 11 Group3 81.7142401464431 75.6869956330724 87.3684804479602 69.6782781479306  
## 12 Group4 68.0937846608894 62.9114749837875 73.1792027567903 56.5330182563317  
## 13 Group4 57.8741978818977 52.4586717180583 63.3975555896503 46.5638055367646  
## 14 Group4 61.9855477171997 56.7788980277451 67.3106884039731 50.4308998693356  
## 15 Group4 64.4394161199823 58.7893239096149 70.0570667676474 52.7120380578995  
## 16 Group4 69.2066349668499 62.8750031490604 75.7741115089413 57.048927402683  
## 17 Group4 61.9263873450161 55.4640827789718 68.2871278132527 49.5366500621053  
## 18 Group5 80.3021034159551 75.5685325566551 85.079314709731 69.0149265820846  
## 19 Group5 70.0825166369634 64.956542115899 75.1944592609737 58.8612154969422  
## 20 Group5 74.1938664722654 69.3347522857943 79.0333752196693 63.0642681189573  
## 21 Group5 75.5614439027981 70.3505817187148 80.5784823260801 64.0738598229077  
## 22 Group5 76.647734875048 71.1044346929184 82.0775398062907 65.2086010204417  
## 23 Group5 81.4149537219157 75.1448842288773 87.5903510167942 69.2748887412149  
## 24 Group5 74.1347061000818 67.6866651450167 80.4165251804217 61.6537584908799  
## 25 Group6 76.4166762197072 70.7232159383923 81.9670206392983 64.5181381700937  
## 26 Group6 66.1970894407155 60.2797342827513 72.0442293813966 54.0035874865905  
## 27 Group6 70.3084392760175 64.5898691026183 75.9396773689022 58.2492598815536  
## 28 Group6 71.6760167065503 65.8468852030795 77.413325462581 59.7083339540442  
## 29 Group7 75.9697885980578 69.7150821727541 82.3278689761543 63.7836602600499  
## 30 Group7 69.861551654368 63.4649230509411 76.2811459347642 57.7498360455188  
## 31 Group7 69.8023912821844 62.9449277823334 76.9122720701028 57.3870574186749  
## 32 Group8 61.2018454690337 55.9148622754879 66.4508872230592 49.6822264111834  
## 33 Group8 50.982258690042 45.3827690147428 56.5469740122298 39.2836764799773  
## 34 Group8 56.4611859558767 50.9558080128104 62.0038066652595 44.6265030692089  
## 35 Group8 57.5474769281265 51.7842155110966 63.2866532155984 45.8166794247607  
## 36 Group8 62.3146957749942 56.0167205049377 68.7661349436676 50.3860163172364  
## 37 Group9 76.0167729389962 69.6483975092107 82.2283633483863 63.9120085507815  
## 38 Group9 71.2761134258392 64.9536297495846 77.5840238471493 59.0531357834948  
## 39 Group9 72.362404398089 65.7792815147968 78.7813405676855 60.0841399495385  
## 40 Group10 79.1952009382374 73.6027256861254 84.7246126034484 67.2438435719138  
## 41 Group10 68.9756141592457 63.1493626306498 74.7729935434917 57.2710795988931  
## 42 Group10 73.0869639945476 67.3870947083379 78.7439047675325 61.5034574018876  
## 43 Group10 75.5408323973302 69.5626136083182 81.562958088102 63.4942329751099  
## 44 Group11 78.0624231276781 72.4233851439817 83.6756006725816 66.1645562489251  
## 45 Group11 67.8428363486864 61.9374819653024 73.7951230739405 55.8877050578752  
## 46 Group11 71.9541861839883 66.256860448181 77.4979444237167 59.9265841001587  
## 47 Group11 73.3217636145211 67.6335035140224 79.1425781718259 61.4689036504036  
## 48 Group12 63.0529055729513 55.6215219811885 70.4412620241776 50.3907848456382  
## 49 Group12 64.4204830034841 57.0958141170578 71.6971233198104 51.7481549943555  
## 50 Group13 78.5289907507831 72.910109645371 84.1740791578429 66.5877317504385  
## 51 Group13 68.3094039717914 62.2981163052152 74.1627474481382 56.5480672594636  
## 52 Group13 72.4207538070933 66.7338534710142 78.1192180553264 60.4743362936679  
## 53 Group13 73.7883312376261 67.9141668933809 79.6044220644119 61.8879748874276  
## 54 Group14 74.2939268309485 69.1226066805153 79.4755841827298 62.8284701716493  
## 55 Group14 68.1856898872587 62.6453660158848 73.679422993245 56.5290524097694  
## 56 Group14 69.5532673177915 64.0727032364998 75.079526564011 57.8310416334478  
## 57 Group14 70.6395582900413 64.7469338570383 76.3991834836732 58.7011202047973  
## 58 Group14 75.406777136909 69.072479684672 81.7214398990217 63.1947703013249  
## 59 Group15 59.8968273981553 52.7213176008374 67.3500362512858 47.4795617897037  
## 60 Group15 56.2424588572481 49.0229225572643 63.7085385388559 43.605059916185  
## PI97.5  
## 1 83.1409429837435  
## 2 77.1356924413352  
## 3 78.3936186200394  
## 4 77.9571407231848  
## 5 83.6822344596676  
## 6 73.3976261892243  
## 7 77.524014386507  
## 8 98.0891413437127  
## 9 87.9174172198909  
## 10 91.8303988305198  
## 11 93.6676190222721  
## 12 79.5035178010331  
## 13 69.572186583464  
## 14 73.6538225258908  
## 15 76.1826349498424  
## 16 81.4962144075696  
## 17 74.0172681376406  
## 18 91.7093547367791  
## 19 81.8467133654907  
## 20 85.5943226224974  
## 21 87.2761411154152  
## 22 88.1344387556984  
## 23 93.3137157960364  
## 24 86.2172635453047  
## 25 88.0454913939307  
## 26 77.8122484061476  
## 27 81.9102114349655  
## 28 83.4043316521798  
## 29 88.1038975629658  
## 30 81.9458907082428  
## 31 82.3559126387988  
## 32 72.833167953758  
## 33 62.3340906083967  
## 34 68.0838051154676  
## 35 69.5434342516263  
## 36 74.8694404598859  
## 37 87.9342935805961  
## 38 83.6449355224177  
## 39 84.6331702993192  
## 40 90.7397112411706  
## 41 80.8511137731415  
## 42 85.0110980903549  
## 43 87.5011961586331  
## 44 89.8604800570373  
## 45 79.7649932417485  
## 46 83.6719562008678  
## 47 84.9998445603138  
## 48 75.6096157656007  
## 49 77.1540963725415  
## 50 90.3609679569527  
## 51 80.2016552172838  
## 52 84.0720609116387  
## 53 85.5792753714262  
## 54 86.1502303631281  
## 55 79.8585465509079  
## 56 81.0250318542175  
## 57 82.3640385489075  
## 58 87.5087894613598  
## 59 72.5959483525098  
## 60 68.8887783850468

# 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 61.13432 5.405386 50.32331 71.69689

# 9.