**Performance Assessment Tasks**

**PAT4: Assignment 2: Multi-centre study data harmonization and visualization – SAS & R (20%)**

**Directions**

This assignment will test your practical knowledge and analysis skills developed through weeks 5 through 9. The assignment will introduce you to working with multi-centre study data, harmonizing variables, and give you practice describing data.

The data source will be fictitious data from the Canadian Multicentre Osteoporosis Bone Quality Study. High resolution CT imaging data derived from different manufacturers will be provided.

Complete the following questions in SAS or R using the SAS dataset provided (xtrfxnd.sas7bdat)

### Q1. Read and create a subset of the xtrfxnd dataframe by selecting only these variables of interest:

['id', 't\_ttbmd', 't\_ctbmd', 'r\_ttbmd', 'r\_ctbmd', 't\_moart', 'r\_moart', 'hicnum', 'centre', 'age', 'osteomed', 'newosteofx', 'prevfx\_any']  
  
Remember to turn all variables to lower case if you're using R.

xtrfxnd = read.sas7bdat("xtrfxnd.sas7bdat")

colnames(xtrfxnd) = tolower(colnames(xtrfxnd))

xtrfxnd\_sub = xtrfxnd[c('id', 't\_ttbmd', 't\_ctbmd', 'r\_ttbmd', 'r\_ctbmd', 't\_moart', 'r\_moart',

'hicnum', 'centre', 'age', 'osteomed', 'newosteofx', 'prevfx\_any')]

*> xtrfxnd = read.sas7bdat("xtrfxnd.sas7bdat")*

*> colnames(xtrfxnd) = tolower(colnames(xtrfxnd))*

*>*

*> xtrfxnd\_sub = xtrfxnd[c('id', 't\_ttbmd', 't\_ctbmd', 'r\_ttbmd', 'r\_ctbmd', 't\_moart', 'r\_moart',*

*+ 'hicnum', 'centre', 'age', 'osteomed', 'newosteofx', 'prevfx\_any')]*

*> xtrfxnd\_sub*

### Q2. Do a frequency count for 'hicnum' and 'centre' - these are the study sites.

### table(xtrfxnd\_sub[c('centre', 'hicnum')])

### *> table(xtrfxnd\_sub[c('centre', 'hicnum')])*

### *hicnum*

### *centre CA HA SK TO VR*

### *1 0 352 0 0 0 0*

### *2 0 0 200 0 0 0*

### *3 0 0 0 161 0 0*

### *4 0 0 0 0 292 0*

### *5 0 0 0 0 0 225*

### Q3. a) Create a new variable called 'sitename', which serves as a proper label for the study site names. Base this off the 2 variables: CENTRE and HICNUM (according to the key below).

### Run another frequency procedure on the new 'sitename' variable to check if you did this correctly.

1 or CA = "Calgary"  
4 or TO = "Toronto - Referent"  
5 or VR = "Vancouver"  
2 or HA = "Hamilton"  
3 or SK = "Saskatoon"

(If it fits none of these sites, just let it be missing)

xtrfxnd\_sub['sitename'] = xtrfxnd\_sub

xtrfxnd\_sub$sitename = ifelse(xtrfxnd\_sub$centre == 1 & xtrfxnd\_sub$hicnum == 'CA',"Calgary",

ifelse(xtrfxnd\_sub$centre == 2 & xtrfxnd\_sub$hicnum == 'HA', "Hamilton",

ifelse(xtrfxnd\_sub$centre == 3 & xtrfxnd\_sub$hicnum == 'SK', "Saskatoon",

ifelse(xtrfxnd\_sub$centre == 4 & xtrfxnd\_sub$hicnum == 'TO', "Toronto - Referent",

ifelse(xtrfxnd\_sub$centre == 5 & xtrfxnd\_sub$hicnum == 'VR', "Vancouver",

NA)))))

table(xtrfxnd\_sub$sitename)

*> xtrfxnd\_sub$sitename = ifelse(xtrfxnd\_sub$centre == 1 & xtrfxnd\_sub$hicnum == 'CA',"Calgary",*

*+ ifelse(xtrfxnd\_sub$centre == 2 & xtrfxnd\_sub$hicnum == 'HA', "Hamilton",*

*+ ifelse(xtrfxnd\_sub$centre == 3 & xtrfxnd\_sub$hicnum == 'SK', "Saskatoon",*

*+ ifelse(xtrfxnd\_sub$centre == 4 & xtrfxnd\_sub$hicnum == 'TO', "Toronto - Referent",*

*+ ifelse(xtrfxnd\_sub$centre == 5 & xtrfxnd\_sub$hicnum == 'VR', "Vancouver",*

*+ NA)))))*

*> table(xtrfxnd\_sub$sitename)*

*Calgary Hamilton Saskatoon Toronto - Referent Vancouver*

*352 200 161 292 225*

### Q3. b) Compute the mean, standard deviation, min and max values for t\_ctbmd, r\_ctbmd and age across the 5 sites.

### # Creating t\_ctbmd Summary Data

### citynames = c('Calgary','Hamilton','Saskatoon','Toronto - Referent','Vancouver')

### sums\_tctbmd = matrix(ncol=6, nrow=length(citynames))

### for(i in 1:length(citynames)){

### sums\_tctbmd[i,] = (c(citynames[i],mean(xtrfxnd\_sub$t\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),

### sd(xtrfxnd\_sub$t\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),

### min(xtrfxnd\_sub$t\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),

### max(xtrfxnd\_sub$t\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE)," "))

### }

### colnames(sums\_tctbmd) = c('City', 'Mean', 'SD','Min','Max','Null')

### sums\_tctbmd = data.frame(sums\_tctbmd)

### sums\_tctbmd[c('Mean','SD','Min','Max')] = sapply(sums\_tctbmd[c('Mean','SD','Min','Max')],as.numeric)

### sums\_tctbmd

### # Creating r\_ctbmd Summary Data

### sums\_rctbmd = matrix(ncol=6, nrow=length(citynames))

### for(i in 1:length(citynames)){

### sums\_rctbmd[i,] = (c(citynames[i],mean(xtrfxnd\_sub$r\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),

### sd(xtrfxnd\_sub$r\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),

### min(xtrfxnd\_sub$r\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),

### max(xtrfxnd\_sub$r\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE)," "))

### }

### colnames(sums\_rctbmd) = c('City', 'Mean', 'SD','Min','Max','Null')

### sums\_rctbmd = data.frame(sums\_rctbmd)

### sums\_rctbmd[c('Mean','SD','Min','Max')] = sapply(sums\_rctbmd[c('Mean','SD','Min','Max')],as.numeric)

### sums\_rctbmd

### # Creating age Summary Data

### sums\_age = matrix(ncol=6, nrow=length(citynames))

### for(i in 1:length(citynames)){

### sums\_age[i,] = (c(citynames[i],mean(xtrfxnd\_sub$age[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),

### sd(xtrfxnd\_sub$age[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),

### min(xtrfxnd\_sub$age[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),

### max(xtrfxnd\_sub$age[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE)," "))

### }

### colnames(sums\_age) = c('City', 'Mean', 'SD','Min','Max','Null')

### sums\_age = data.frame(sums\_age)

### sums\_age[c('Mean','SD','Min','Max')] = sapply(sums\_age[c('Mean','SD','Min','Max')],as.numeric)

### sums\_age

### *> # Creating t\_ctbmd Summary Data*

### *> citynames = c('Calgary','Hamilton','Saskatoon','Toronto - Referent','Vancouver')*

### *> sums\_tctbmd = matrix(ncol=6, nrow=length(citynames))*

### *> for(i in 1:length(citynames)){*

### *+ sums\_tctbmd[i,] = (c(citynames[i],mean(xtrfxnd\_sub$t\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),*

### *+ sd(xtrfxnd\_sub$t\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),*

### *+ min(xtrfxnd\_sub$t\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),*

### *+ max(xtrfxnd\_sub$t\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE)," "))*

### *+ }*

### *> colnames(sums\_tctbmd) = c('City', 'Mean', 'SD','Min','Max','Null')*

### *> sums\_tctbmd = data.frame(sums\_tctbmd)*

### *> sums\_tctbmd[c('Mean','SD','Min','Max')] = sapply(sums\_tctbmd[c('Mean','SD','Min','Max')],as.numeric)*

### *> sums\_tctbmd*

### *City Mean SD Min Max Null*

### *1 Calgary 805.5005 64.75705 639.6612 975.6482*

### *2 Hamilton 751.1466 102.19021 20.3000 885.5000*

### *3 Saskatoon 781.1675 85.05054 501.8000 976.8000*

### *4 Toronto - Referent 793.9955 68.05069 567.1000 936.3000*

### *5 Vancouver 712.8100 89.65446 374.6000 878.4000*

### *>*

### *> # Creating r\_ctbmd Summary Data*

### *> sums\_rctbmd = matrix(ncol=6, nrow=length(citynames))*

### *> for(i in 1:length(citynames)){*

### *+ sums\_rctbmd[i,] = (c(citynames[i],mean(xtrfxnd\_sub$r\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),*

### *+ sd(xtrfxnd\_sub$r\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),*

### *+ min(xtrfxnd\_sub$r\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),*

### *+ max(xtrfxnd\_sub$r\_ctbmd[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE)," "))*

### *+ }*

### *> colnames(sums\_rctbmd) = c('City', 'Mean', 'SD','Min','Max','Null')*

### *> sums\_rctbmd = data.frame(sums\_rctbmd)*

### *> sums\_rctbmd[c('Mean','SD','Min','Max')] = sapply(sums\_rctbmd[c('Mean','SD','Min','Max')],as.numeric)*

### *> sums\_rctbmd*

### *City Mean SD Min Max Null*

### *1 Calgary 885.6607 59.97328 700.4126 1013.097*

### *2 Hamilton 782.8075 64.44884 637.3000 911.900*

### *3 Saskatoon 802.0395 86.52246 491.8000 955.800*

### *4 Toronto - Referent 805.7571 71.27643 556.3000 967.800*

### *5 Vancouver 721.0914 87.56635 467.7000 900.200*

### *>*

### *> # Creating age Summary Data*

### *> sums\_age = matrix(ncol=6, nrow=length(citynames))*

### *> for(i in 1:length(citynames)){*

### *+ sums\_age[i,] = (c(citynames[i],mean(xtrfxnd\_sub$age[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),*

### *+ sd(xtrfxnd\_sub$age[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),*

### *+ min(xtrfxnd\_sub$age[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE),*

### *+ max(xtrfxnd\_sub$age[xtrfxnd\_sub$sitename == citynames[i]],na.rm=TRUE)," "))*

### *+ }*

### *> colnames(sums\_age) = c('City', 'Mean', 'SD','Min','Max','Null')*

### *> sums\_age = data.frame(sums\_age)*

### *> sums\_age[c('Mean','SD','Min','Max')] = sapply(sums\_age[c('Mean','SD','Min','Max')],as.numeric)*

### *> sums\_age*

### *City Mean SD Min Max Null*

### *1 Calgary 73.70170 6.459975 60 86*

### *2 Hamilton 74.27500 7.742682 58 89*

### *3 Saskatoon 75.06211 6.514109 60 87*

### *4 Toronto - Referent 69.60616 9.853865 40 91*

### *5 Vancouver 73.02667 7.133873 61 85*

### Which site had the youngest average age?

### which.min(sums\_age$Mean)

### sums\_age[4,]

### *> which.min(sums\_age$Mean)*

### *[1] 4*

### *> sums\_age[4,]*

### *City Mean SD Min Max Null*

### *4 Toronto - Referent 69.60616 9.853865 40 91*

### Which site had the highest cortical bone density at the tibia (t\_ctbmd) and radius(r\_ctbmd)?

### which.max(sums\_tctbmd$Max)

### sums\_tctbmd[3,]

### which.max(sums\_rctbmd$Max)

### sums\_rctbmd[1,]

### *> which.max(sums\_tctbmd$Max)*

### *[1] 3*

### *> sums\_tctbmd[3,]*

### *City Mean SD Min Max Null*

### *3 Saskatoon 781.1675 85.05054 501.8 976.8*

### *> which.max(sums\_rctbmd$Max)*

### *[1] 1*

### *> sums\_rctbmd[1,]*

### *City Mean SD Min Max Null*

### *1 Calgary 885.6607 59.97328 700.4126 1013.097*

### Q4. a) Any motion grade (t\_moart and r\_moart) of 3 or higher is considered a failed, unusable scan.

### Create a variable called "t\_QCPass and r\_QCPass" and assign a value of 1 if motion grade (t\_moart and r\_moart) is under 3, otherwise assign a value of 0.

### xtrfxnd\_sub$t\_QCPass = ifelse(xtrfxnd\_sub$t\_moart < 3,1,0)

### xtrfxnd\_sub$r\_QCPass = ifelse(xtrfxnd\_sub$t\_moart < 3,1,0)

### *> xtrfxnd\_sub$t\_QCPass = ifelse(xtrfxnd\_sub$t\_moart < 3,1,0)*

### *> xtrfxnd\_sub$r\_QCPass = ifelse(xtrfxnd\_sub$t\_moart < 3,1,0)*

### Q4. b) Compute the median, q1 (first quartile) and q3 (third quartile) for t\_ttbmd, t\_ctbmd, r\_ttbmd and r\_ctbmd across each of the 5 sites only for cases when QC passed (QCPass=1).

### #Creating t\_ttbmd summary table

### sums2\_tttbmd = matrix(ncol=5, nrow=length(citynames))

### for(i in 1:length(citynames)){

### sums2\_tttbmd[i,] = (c(citynames[i],median(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ttbmd'],na.rm=TRUE),

### quantile(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ttbmd'],0.25,na.rm=TRUE),

### quantile(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ttbmd'],0.75,na.rm=TRUE),

### " "))

### }

### colnames(sums2\_tttbmd) = c('City', 'Median', 'Q1','Q3','Null')

### sums2\_tttbmd = data.frame(sums2\_tttbmd)

### sums2\_tttbmd[c('Median','Q1','Q3')] = sapply(sums2\_tttbmd[c('Median','Q1','Q3')],as.numeric)

### sums2\_tttbmd

### #Creating t\_ctbmd summary table

### sums2\_tctbmd = matrix(ncol=5, nrow=length(citynames))

### for(i in 1:length(citynames)){

### sums2\_tctbmd[i,] = (c(citynames[i],median(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ctbmd'],na.rm=TRUE),

### quantile(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ctbmd'],0.25,na.rm=TRUE),

### quantile(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ctbmd'],0.75,na.rm=TRUE),

### " "))

### }

### colnames(sums2\_tctbmd) = c('City', 'Median', 'Q1','Q3','Null')

### sums2\_tctbmd = data.frame(sums2\_tctbmd)

### sums2\_tctbmd[c('Median','Q1','Q3')] = sapply(sums2\_tctbmd[c('Median','Q1','Q3')],as.numeric)

### sums2\_tctbmd

### #Creating r\_ttbmd summary table

### sums2\_rttbmd = matrix(ncol=5, nrow=length(citynames))

### for(i in 1:length(citynames)){

### sums2\_rttbmd[i,] = (c(citynames[i],median(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ttbmd'],na.rm=TRUE),

### quantile(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ttbmd'],0.25,na.rm=TRUE),

### quantile(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ttbmd'],0.75,na.rm=TRUE),

### " "))

### }

### colnames(sums2\_rttbmd) = c('City', 'Median', 'Q1','Q3','Null')

### sums2\_rttbmd = data.frame(sums2\_rttbmd)

### sums2\_rttbmd[c('Median','Q1','Q3')] = sapply(sums2\_rttbmd[c('Median','Q1','Q3')],as.numeric)

### sums2\_rttbmd

### #Creating r\_ctbmd summary table

### sums2\_rctbmd = matrix(ncol=5, nrow=length(citynames))

### for(i in 1:length(citynames)){

### sums2\_rctbmd[i,] = (c(citynames[i],median(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ctbmd'],na.rm=TRUE),

### quantile(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ctbmd'],0.25,na.rm=TRUE),

### quantile(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ctbmd'],0.75,na.rm=TRUE),

### " "))

### }

### colnames(sums2\_rctbmd) = c('City', 'Median', 'Q1','Q3','Null')

### sums2\_rctbmd = data.frame(sums2\_rctbmd)

### sums2\_rctbmd[c('Median','Q1','Q3')] = sapply(sums2\_rctbmd[c('Median','Q1','Q3')],as.numeric)

### sums2\_rctbmd

### *> #Creating t\_ttbmd summary table*

### *> sums2\_tttbmd = matrix(ncol=5, nrow=length(citynames))*

### *> for(i in 1:length(citynames)){*

### *+ sums2\_tttbmd[i,] = (c(citynames[i],median(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ttbmd'],na.rm=TRUE),*

### *+ quantile(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ttbmd'],0.25,na.rm=TRUE),*

### *+ quantile(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ttbmd'],0.75,na.rm=TRUE),*

### *+ " "))*

### *+ }*

### *> colnames(sums2\_tttbmd) = c('City', 'Median', 'Q1','Q3','Null')*

### *> sums2\_tttbmd = data.frame(sums2\_tttbmd)*

### *> sums2\_tttbmd[c('Median','Q1','Q3')] = sapply(sums2\_tttbmd[c('Median','Q1','Q3')],as.numeric)*

### *> sums2\_tttbmd*

### *City Median Q1 Q3 Null*

### *1 Calgary 253.3723 221.4605 282.6416*

### *2 Hamilton 243.9000 207.4500 281.2000*

### *3 Saskatoon 260.7000 223.1000 290.7000*

### *4 Toronto - Referent 265.3000 233.7000 293.0000*

### *5 Vancouver 244.5000 207.0000 279.9000*

### *>*

### *> #Creating t\_ctbmd summary table*

### *> sums2\_tctbmd = matrix(ncol=5, nrow=length(citynames))*

### *> for(i in 1:length(citynames)){*

### *+ sums2\_tctbmd[i,] = (c(citynames[i],median(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ctbmd'],na.rm=TRUE),*

### *+ quantile(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ctbmd'],0.25,na.rm=TRUE),*

### *+ quantile(xtrfxnd\_sub[xtrfxnd\_sub$t\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'t\_ctbmd'],0.75,na.rm=TRUE),*

### *+ " "))*

### *+ }*

### *> colnames(sums2\_tctbmd) = c('City', 'Median', 'Q1','Q3','Null')*

### *> sums2\_tctbmd = data.frame(sums2\_tctbmd)*

### *> sums2\_tctbmd[c('Median','Q1','Q3')] = sapply(sums2\_tctbmd[c('Median','Q1','Q3')],as.numeric)*

### *> sums2\_tctbmd*

### *City Median Q1 Q3 Null*

### *1 Calgary 801.3451 762.0302 853.9825*

### *2 Hamilton 759.2000 735.0000 795.3500*

### *3 Saskatoon 768.3000 718.2000 860.8000*

### *4 Toronto - Referent 793.2000 763.2500 840.1000*

### *5 Vancouver 725.1000 680.9000 776.5000*

### *>*

### *> #Creating r\_ttbmd summary table*

### *> sums2\_rttbmd = matrix(ncol=5, nrow=length(citynames))*

### *> for(i in 1:length(citynames)){*

### *+ sums2\_rttbmd[i,] = (c(citynames[i],median(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ttbmd'],na.rm=TRUE),*

### *+ quantile(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ttbmd'],0.25,na.rm=TRUE),*

### *+ quantile(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ttbmd'],0.75,na.rm=TRUE),*

### *+ " "))*

### *+ }*

### *> colnames(sums2\_rttbmd) = c('City', 'Median', 'Q1','Q3','Null')*

### *> sums2\_rttbmd = data.frame(sums2\_rttbmd)*

### *> sums2\_rttbmd[c('Median','Q1','Q3')] = sapply(sums2\_rttbmd[c('Median','Q1','Q3')],as.numeric)*

### *> sums2\_rttbmd*

### *City Median Q1 Q3 Null*

### *1 Calgary 282.5978 242.6655 319.4977*

### *2 Hamilton 255.5000 227.7000 303.4000*

### *3 Saskatoon 269.9500 232.1500 325.9000*

### *4 Toronto - Referent 282.0000 233.7000 319.9000*

### *5 Vancouver 251.9000 207.4000 296.7000*

### *>*

### *> #Creating r\_ctbmd summary table*

### *> sums2\_rctbmd = matrix(ncol=5, nrow=length(citynames))*

### *> for(i in 1:length(citynames)){*

### *+ sums2\_rctbmd[i,] = (c(citynames[i],median(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ctbmd'],na.rm=TRUE),*

### *+ quantile(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ctbmd'],0.25,na.rm=TRUE),*

### *+ quantile(xtrfxnd\_sub[xtrfxnd\_sub$r\_QCPass == 1 & xtrfxnd\_sub$sitename == citynames[i],'r\_ctbmd'],0.75,na.rm=TRUE),*

### *+ " "))*

### *+ }*

### *> colnames(sums2\_rctbmd) = c('City', 'Median', 'Q1','Q3','Null')*

### *> sums2\_rctbmd = data.frame(sums2\_rctbmd)*

### *> sums2\_rctbmd[c('Median','Q1','Q3')] = sapply(sums2\_rctbmd[c('Median','Q1','Q3')],as.numeric)*

### *> sums2\_rctbmd*

### *City Median Q1 Q3 Null*

### *1 Calgary 891.0968 843.0291 928.7572*

### *2 Hamilton 794.7500 732.0500 826.2000*

### *3 Saskatoon 795.1000 736.5000 880.6750*

### *4 Toronto - Referent 799.1500 755.1000 853.7000*

### *5 Vancouver 737.8000 659.6000 780.8000*

### Which site has the lowest q1 (first quartile) for total bone density at the tibia (t\_ttbmd)?

### which.min(sums2\_tttbmd$Q1)

### sums2\_tttbmd[5,]

### *> which.min(sums2\_tttbmd$Q1)*

### *[1] 5*

### *> sums2\_tttbmd[5,]*

### *City Median Q1 Q3 Null*

### *5 Vancouver 244.5 207 279.9*

### How large of a difference in median total bone density at the radius (r\_ctbmd) is the site with the highest versus lowest value? (round to 2 decimal places)

### r\_ctbmd\_median\_diff = round((max(sums2\_rctbmd$Median) - min(sums2\_rctbmd$Median)),2)

### r\_ctbmd\_median\_diff

### *> r\_ctbmd\_median\_diff = round((max(sums2\_rctbmd$Median) - min(sums2\_rctbmd$Median)),2)*

### *> r\_ctbmd\_median\_diff*

### *[1] 153.30*

### Q5. Data harmonization

### When the study was completed, the CT scanners were cross-calibrated

* meaning an object was scanned at each of the CT scanners from every site. This was done to check for comparability and to allow any future adjustments.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Vancouver** |  | **Calgary** |  | **Saskatoon** |  |
| **Bone variable** | Slope (95%CI) | Intercept(95%CI) | Slope (95%CI) | Intercept(95%CI) | Slope (95%CI) | Intercept (95%CI) |
| **vBMDi(mg/cm3)**  **(t\_ttbmd or r\_ttbmd)** | 0.99 | -3.93 | 0.97 | 19.06 | 1.03 | -23.26 |
| **vBMDc (mg/cm3)**  **(t\_ctbmd or r\_ctbmd)** | 1.04 | -0.05 | 1.05 | -6.82 | 0.99 | -0.03 |

Toronto was the reference site. Hamilton used the same scanner as in Toronto.

All other sites' bone values had to be adjusted according to Toronto's values. To do this, slopes and intercepts represent **multipliers** and **additions** of values needed, respectively. (note, some are subtraction not addition: those with negative signs).

For example, all bone density values in Vancouver need to first be multiplied by 0.99 then - 3.93.

#### **Q5. Create new cross-calibrated values for all sites' 't\_ttbmd', 't\_ctbmd', 'r\_ttbmd', 'r\_ctbmd' values according to this table.**

#### **Assign the new values to new variables following the same original name but ending in an extension of "\_new".**

#### NOTE: Use the vBMDi row values for ttbmd and vBMDc row values for ctbmd

#### Toronto and Hamilton should just take on its original value, unchanged.

As an example:

if site is Vancouver: t\_ttbmd\_new = 0.99\*t\_ttbmd - 3.93

xtrfxnd\_sub$t\_ttbmd\_new =

ifelse(xtrfxnd\_sub$sitename == 'Vancouver', xtrfxnd\_sub$t\_ttbmd \* 0.99 - 3.93,

ifelse(xtrfxnd\_sub$sitename == 'Calgary', xtrfxnd\_sub$t\_ttbmd \* 0.97 + 19.06,

ifelse(xtrfxnd\_sub$sitename == 'Saskatoon', xtrfxnd\_sub$t\_ttbmd \* 1.03 - 23.26,

xtrfxnd\_sub$t\_ttbmd)))

xtrfxnd\_sub$t\_ctbmd\_new =

ifelse(xtrfxnd\_sub$sitename == 'Vancouver', xtrfxnd\_sub$t\_ctbmd \* 1.04 - 0.05,

ifelse(xtrfxnd\_sub$sitename == 'Calgary', xtrfxnd\_sub$t\_ctbmd \* 1.05 - 6.82,

ifelse(xtrfxnd\_sub$sitename == 'Saskatoon', xtrfxnd\_sub$t\_ctbmd \* 0.99 - 0.03,

xtrfxnd\_sub$t\_ctbmd)))

xtrfxnd\_sub$r\_ttbmd\_new =

ifelse(xtrfxnd\_sub$sitename == 'Vancouver', xtrfxnd\_sub$r\_ttbmd \* 0.99 - 3.93,

ifelse(xtrfxnd\_sub$sitename == 'Calgary', xtrfxnd\_sub$r\_ttbmd \* 0.97 + 19.06,

ifelse(xtrfxnd\_sub$sitename == 'Saskatoon', xtrfxnd\_sub$r\_ttbmd \* 1.03 - 23.26,

xtrfxnd\_sub$r\_ttbmd)))

xtrfxnd\_sub$r\_ctbmd\_new =

ifelse(xtrfxnd\_sub$sitename == 'Vancouver', xtrfxnd\_sub$r\_ctbmd \* 1.04 - 0.05,

ifelse(xtrfxnd\_sub$sitename == 'Calgary', xtrfxnd\_sub$r\_ctbmd \* 1.05 - 6.82,

ifelse(xtrfxnd\_sub$sitename == 'Saskatoon', xtrfxnd\_sub$r\_ctbmd \* 0.99 - 0.03,

xtrfxnd\_sub$r\_ctbmd)))

*> xtrfxnd\_sub$t\_ttbmd\_new =*

*+ ifelse(xtrfxnd\_sub$sitename == 'Vancouver', xtrfxnd\_sub$t\_ttbmd \* 0.99 - 3.93,*

*+ ifelse(xtrfxnd\_sub$sitename == 'Calgary', xtrfxnd\_sub$t\_ttbmd \* 0.97 + 19.06,*

*+ ifelse(xtrfxnd\_sub$sitename == 'Saskatoon', xtrfxnd\_sub$t\_ttbmd \* 1.03 - 23.26,*

*+ xtrfxnd\_sub$t\_ttbmd)))*

*>*

*> xtrfxnd\_sub$t\_ctbmd\_new =*

*+ ifelse(xtrfxnd\_sub$sitename == 'Vancouver', xtrfxnd\_sub$t\_ctbmd \* 1.04 - 0.05,*

*+ ifelse(xtrfxnd\_sub$sitename == 'Calgary', xtrfxnd\_sub$t\_ctbmd \* 1.05 - 6.82,*

*+ ifelse(xtrfxnd\_sub$sitename == 'Saskatoon', xtrfxnd\_sub$t\_ctbmd \* 0.99 - 0.03,*

*+ xtrfxnd\_sub$t\_ctbmd)))*

*>*

*> xtrfxnd\_sub$r\_ttbmd\_new =*

*+ ifelse(xtrfxnd\_sub$sitename == 'Vancouver', xtrfxnd\_sub$r\_ttbmd \* 0.99 - 3.93,*

*+ ifelse(xtrfxnd\_sub$sitename == 'Calgary', xtrfxnd\_sub$r\_ttbmd \* 0.97 + 19.06,*

*+ ifelse(xtrfxnd\_sub$sitename == 'Saskatoon', xtrfxnd\_sub$r\_ttbmd \* 1.03 - 23.26,*

*+ xtrfxnd\_sub$r\_ttbmd)))*

*>*

*> xtrfxnd\_sub$r\_ctbmd\_new =*

*+ ifelse(xtrfxnd\_sub$sitename == 'Vancouver', xtrfxnd\_sub$r\_ctbmd \* 1.04 - 0.05,*

*+ ifelse(xtrfxnd\_sub$sitename == 'Calgary', xtrfxnd\_sub$r\_ctbmd \* 1.05 - 6.82,*

*+ ifelse(xtrfxnd\_sub$sitename == 'Saskatoon', xtrfxnd\_sub$r\_ctbmd \* 0.99 - 0.03,*

*+ xtrfxnd\_sub$r\_ctbmd)))*

**Q6. Convert t\_ttbmd\_new, r\_ttbmd\_new, t\_ctbmd\_new, r\_ctbmd\_new from wide to long (just ttbmd and ctbmd as variable names) with (a) specifier variable(s) to indicate whether the scan was done at the radius (r\_) or tibia (t\_).**

xtrfxnd\_sub\_long = reshape(xtrfxnd\_sub, idvar = "id", varying=c(colnames(xtrfxnd\_sub)[c(17,19)]),timevar = "time", times=c('(t\_)','(r\_)'), v.names = c('ttbmd'), direction = "long")

xtrfxnd\_sub\_long2 = reshape(xtrfxnd\_sub, idvar = "id", varying=c(colnames(xtrfxnd\_sub)[c(18,20)]),timevar = "time", times=c('(t\_)','(r\_)'), v.names = c('ctbmd'), direction = "long")

*> xtrfxnd\_sub\_long = reshape(xtrfxnd\_sub, idvar = "id", varying=c(colnames(xtrfxnd\_sub)[c(17,19)]),timevar = "time", times=c('(t\_)','(r\_)'), v.names = c('ttbmd'), direction = "long")*

*> xtrfxnd\_sub\_long2 = reshape(xtrfxnd\_sub, idvar = "id", varying=c(colnames(xtrfxnd\_sub)[c(18,20)]),timevar = "time", times=c('(t\_)','(r\_)'), v.names = c('ctbmd'), direction = "long")*

**Q7. Create a new ID variable called SID that includes the 2 letter site code from the variable "HICNUM" followed by 5 numbers representing the ID, with leading 0's where applicable.**

max(nchar(xtrfxnd\_sub$id))

xtrfxnd\_sub$SID = paste(xtrfxnd\_sub$hicnum,ifelse(nchar(xtrfxnd\_sub$id) == 3,"00","0"),xtrfxnd\_sub$id, sep='')

xtrfxnd\_sub$SID[100:120]

*> max(nchar(xtrfxnd\_sub$id))*

*[1] 4*

*> xtrfxnd\_sub$SID = paste(xtrfxnd\_sub$hicnum,ifelse(nchar(xtrfxnd\_sub$id) == 3,"00","0"),xtrfxnd\_sub$id, sep='')*

*> xtrfxnd\_sub$SID[100:120]*

*[1] "CA00990" "CA00991" "CA00992" "CA00993" "CA00994" "CA00995" "CA00996" "CA00997" "CA00998" "CA00999"*

*[11] "CA01000" "CA01001" "CA01002" "CA01003" "CA01004" "CA01005" "CA01006" "CA01007" "CA01008" "CA01009"*

*[21] "CA01010"*

**Target Course Competencies**

This exercise targets course competencies 1 and 2.

**Rating Scale**

|  |  |
| --- | --- |
| **Value** | **Description** |
| **3** | **Fully meets or exceeds the outcome** |
| **2** | **Meets most of the outcome** |
| **1** | **Meets some of the outcome** |
| **0** | **Does not meet the outcome** |

**Scoring Standard**

This assignment is designed to give the student experience working with clinical data, harmonizing multi-centre study variables, and generating basic statistical descriptives. The questions in this assignment help prepare them to assist investigators with data transformations, while considering inconsistencies across sites. They are designed to detail specific aspects of variables, data collection, and procedures to enable comparability.

A score of 60% on this assignment is required.

**Scoring Guide**

|  |  |  |
| --- | --- | --- |
| **Competency** | **Criteria** | **Ratings** |
| Q1. | Ability to read and subset a dataframe | 3 2 1 0 |
| Q2. | Ability to complete a frequency check | 3 2 1 0 |
| Q3.a | Ability to recode a variable using conditions | 3 2 1 0 |
| Q3.b | Ability to generate basic statistical descriptives of central tendency and distribution | 3 2 1 0 |
| Q4.a | Ability to recode a variable using conditions | 3 2 1 0 |
| Q4.b | Ability to generate basic statistical descriptives of for specific variable classes | 3 2 1 0 |
| Q5. | Ability to transform variables | 3 2 1 0 |
| Q6 | Ability to convert wide to long datasets | 3 2 1 0 |
| Q7. | Ability to generate and manipulate string characters | 3 2 1 0 |
|  |  | / 27 |

**Deadline**

November 13th 11:59 pm.