PCA2

2022-08-16

WOULD NOT RECOMMEND USING THIS! Scale is different from other PCA graphs on PCA.Rmd

library(FactoMineR)  
library(factoextra)

## Loading required package: ggplot2

## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa

library(tidyverse)

## ── Attaching packages  
## ───────────────────────────────────────  
## tidyverse 1.3.2 ──

## ✔ tibble 3.1.8 ✔ dplyr 1.0.9  
## ✔ tidyr 1.2.0 ✔ stringr 1.4.0  
## ✔ readr 2.1.2 ✔ forcats 0.5.1  
## ✔ purrr 0.3.4   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(MASS)

##   
## Attaching package: 'MASS'  
##   
## The following object is masked from 'package:dplyr':  
##   
## select

library(reshape2)

##   
## Attaching package: 'reshape2'  
##   
## The following object is masked from 'package:tidyr':  
##   
## smiths

library(cowplot)  
library(flextable)

##   
## Attaching package: 'flextable'  
##   
## The following object is masked from 'package:purrr':  
##   
## compose

library(readxl)  
library(extrafont)

## Registering fonts with R

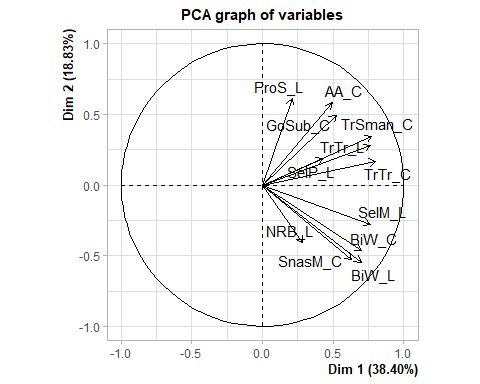
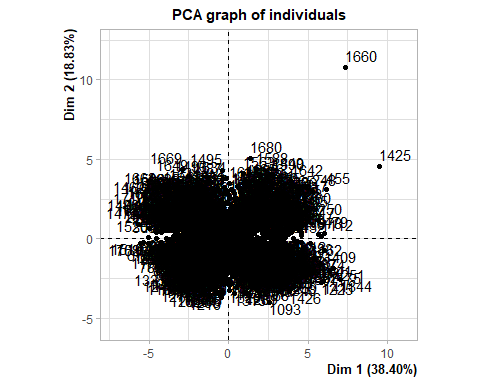
library(forcats)  
library(writexl)  
library(ggfortify)  
library(fauxnaif)

PCAdata\_full<-read\_excel("C:\\Users\\19177\\OneDrive - Colostate\\Desktop\\Dissertation\\headscan\_dissertation\\PCAdata\_full.xlsx")  
  
str(PCAdata\_full)

## tibble [1,711 × 15] (S3: tbl\_df/tbl/data.frame)  
## $ AA\_C : num [1:1711] 65 55 70 58 67 60 59 59 65 65 ...  
## $ BiW\_C : num [1:1711] 130 127 143 140 137 130 141 138 143 150 ...  
## $ BiW\_L : num [1:1711] 115 108 121 109 104 106 109 111 113 116 ...  
## $ GoSub\_C : num [1:1711] 93 93 115 93 103 100 79 106 85 102 ...  
## $ NRB\_L : num [1:1711] 17 18 19 21 19 14 17 18 16 17 ...  
## $ ProS\_L : num [1:1711] 17 18 14 13 20 20 18 12 24 22 ...  
## $ SelP\_L : num [1:1711] 42 41 51 44 47 48 46 41 46 44 ...  
## $ SelM\_L : num [1:1711] 122 99 130 115 119 126 117 112 117 117 ...  
## $ SnasM\_C : num [1:1711] 82 55 84 74 73 80 78 76 64 75 ...  
## $ TrSman\_C : num [1:1711] 177 145 178 147 157 164 149 159 151 160 ...  
## $ TrTr\_C : num [1:1711] 296 276 292 273 279 300 283 275 307 286 ...  
## $ TrTr\_L : num [1:1711] 155 141 156 149 146 146 147 151 157 144 ...  
## $ gender : chr [1:1711] "Male" "Female" "Male" "Male" ...  
## $ race\_eth : chr [1:1711] "Black" "white" "white" "white" ...  
## $ age\_group: chr [1:1711] "18-36" "37-54" "37-54" "18-36" ...

#https://tem11010.github.io/Plotting-PCAs/  
  
PCAdata\_full1 <- PCAdata\_full %>% filter (AA\_C > 0, BiW\_C > 0, BiW\_L > 0, GoSub\_C > 0,   
 NRB\_L > 0, ProS\_L > 0, SelP\_L > 0, SelM\_L > 0,   
 SnasM\_C > 0, TrSman\_C > 0, TrTr\_C > 0, TrTr\_L > 0)  
  
# log price  
  
# center and scale the data  
for (i in 1:length(colnames(PCAdata\_full1))){  
   
 if (is.numeric(PCAdata\_full1[, i])==TRUE)  
   
 PCAdata\_full1[, i] <- as.numeric(scale(PCAdata\_full1[, i]))  
   
 else  
   
 PCAdata\_full1[, i] <- PCAdata\_full1[, i]  
   
}

pca1 <- PCA(PCAdata\_full1[ ,c("AA\_C", "BiW\_C", "BiW\_L", "GoSub\_C", "NRB\_L", "ProS\_L", "SelP\_L",   
 "SelM\_L", "SnasM\_C", "TrSman\_C", "TrTr\_C", "TrTr\_L")])



summary(pca1)

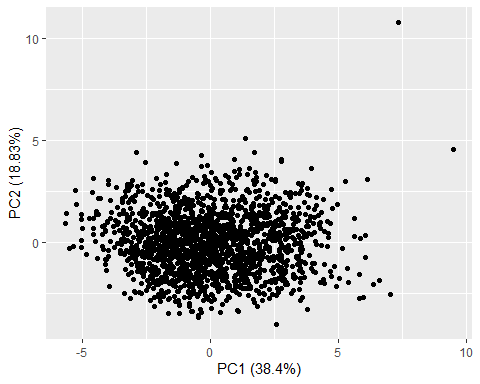
##   
## Call:  
## PCA(X = PCAdata\_full1[, c("AA\_C", "BiW\_C", "BiW\_L", "GoSub\_C",   
## "NRB\_L", "ProS\_L", "SelP\_L", "SelM\_L", "SnasM\_C", "TrSman\_C",   
## "TrTr\_C", "TrTr\_L")])   
##   
##   
## Eigenvalues  
## Dim.1 Dim.2 Dim.3 Dim.4 Dim.5 Dim.6 Dim.7  
## Variance 4.608 2.259 1.304 0.961 0.763 0.665 0.483  
## % of var. 38.398 18.827 10.867 8.010 6.359 5.541 4.021  
## Cumulative % of var. 38.398 57.224 68.092 76.102 82.462 88.003 92.024  
## Dim.8 Dim.9 Dim.10 Dim.11 Dim.12  
## Variance 0.374 0.264 0.155 0.092 0.072  
## % of var. 3.116 2.201 1.293 0.767 0.599  
## Cumulative % of var. 95.140 97.341 98.634 99.401 100.000  
##   
## Individuals (the 10 first)  
## Dist Dim.1 ctr cos2 Dim.2 ctr cos2 Dim.3 ctr  
## 1 | 2.835 | 1.816 0.042 0.410 | 0.112 0.000 0.002 | -0.928 0.039  
## 2 | 3.083 | -2.438 0.075 0.625 | 0.197 0.001 0.004 | -1.118 0.056  
## 3 | 4.401 | 3.530 0.158 0.644 | -0.217 0.001 0.002 | 0.012 0.000  
## 4 | 2.621 | -0.457 0.003 0.030 | -1.525 0.060 0.338 | -0.601 0.016  
## 5 | 1.556 | 0.499 0.003 0.103 | 0.827 0.018 0.283 | 0.653 0.019  
## 6 | 2.347 | 1.186 0.018 0.255 | 0.654 0.011 0.078 | 0.589 0.016  
## 7 | 1.538 | -0.024 0.000 0.000 | -0.830 0.018 0.291 | 0.710 0.023  
## 8 | 3.004 | -0.087 0.000 0.001 | -1.055 0.029 0.123 | -1.701 0.130  
## 9 | 3.463 | 1.440 0.026 0.173 | 1.336 0.046 0.149 | 0.812 0.030  
## 10 | 2.079 | 1.176 0.018 0.320 | 0.270 0.002 0.017 | 0.323 0.005  
## cos2   
## 1 0.107 |  
## 2 0.132 |  
## 3 0.000 |  
## 4 0.053 |  
## 5 0.176 |  
## 6 0.063 |  
## 7 0.213 |  
## 8 0.321 |  
## 9 0.055 |  
## 10 0.024 |  
##   
## Variables (the 10 first)  
## Dim.1 ctr cos2 Dim.2 ctr cos2 Dim.3 ctr cos2   
## AA\_C | 0.493 5.276 0.243 | 0.581 14.967 0.338 | 0.303 7.044 0.092 |  
## BiW\_C | 0.702 10.695 0.493 | -0.466 9.625 0.217 | 0.138 1.468 0.019 |  
## BiW\_L | 0.698 10.579 0.487 | -0.549 13.319 0.301 | -0.091 0.630 0.008 |  
## GoSub\_C | 0.521 5.899 0.272 | 0.495 10.863 0.245 | -0.460 16.217 0.211 |  
## NRB\_L | 0.285 1.760 0.081 | -0.408 7.369 0.166 | -0.241 4.441 0.058 |  
## ProS\_L | 0.209 0.949 0.044 | 0.614 16.691 0.377 | 0.303 7.034 0.092 |  
## SelP\_L | 0.430 4.008 0.185 | 0.185 1.520 0.034 | 0.712 38.915 0.507 |  
## SelM\_L | 0.764 12.663 0.583 | -0.277 3.386 0.076 | 0.333 8.504 0.111 |  
## SnasM\_C | 0.631 8.637 0.398 | -0.526 12.228 0.276 | 0.097 0.725 0.009 |  
## TrSman\_C | 0.773 12.976 0.598 | 0.347 5.318 0.120 | -0.363 10.120 0.132 |

PCAdata\_full1$pc1 <- pca1$ind$coord[, 1] # indexing the first column  
  
PCAdata\_full1$pc2 <- pca1$ind$coord[, 2] # indexing the second column

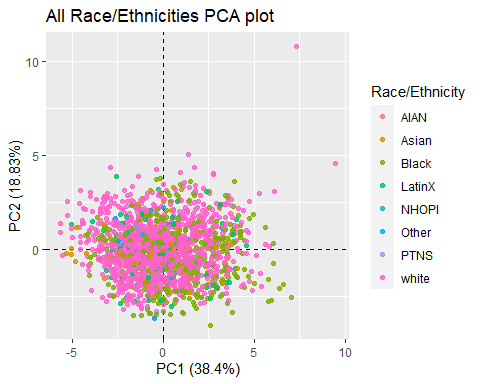
pca.vars <- pca1$var$coord %>% data.frame  
  
pca.vars$vars <- rownames(pca.vars)  
  
pca.vars.m <- melt(pca.vars, id.vars = "vars")

#circleFun <- function(center = c(0,0),diameter = 1, npoints = 100){  
 #r = diameter / 2  
 #tt <- seq(0,2\*pi,length.out = npoints)  
 #xx <- center[1] + r \* cos(tt)  
 #yy <- center[2] + r \* sin(tt)  
 #return(data.frame(x = xx, y = yy))  
#}  
  
#circ <- circleFun(c(0,0),2,npoints = 500)

ggplot(data = PCAdata\_full1, aes(x = pc1, y = pc2)) +  
 geom\_point()+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)")

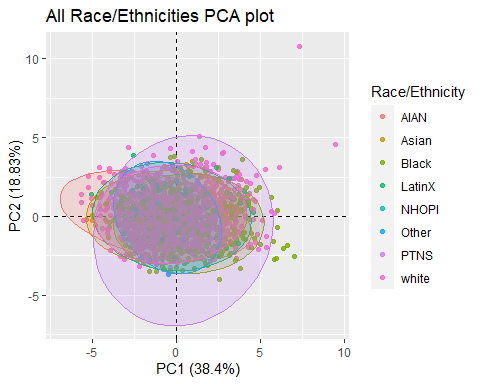


#checking to make sure this matches plot in PCA.Rmd  
ggplot(data = PCAdata\_full1, aes(x = pc1, y = pc2, color = race\_eth)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Race/Ethnicities PCA plot",  
 color = "Race/Ethnicity")



ggplot(data = PCAdata\_full1, aes(x = pc1, y = pc2, color = race\_eth)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +   
 stat\_ellipse(geom="polygon", aes(fill = race\_eth),   
 alpha = 0.2,   
 show.legend = FALSE,  
 level = 0.95) +  
 #theme\_minimal() +  
 #theme(panel.grid = element\_blank(), panel.border = element\_rect(fill= "transparent"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Race/Ethnicities PCA plot",  
 color = "Race/Ethnicity")

## Too few points to calculate an ellipse

 Need a plot with just Asian, Black, LatinX, and white

PCAdata\_full2 <- PCAdata\_full1  
  
PCAdata\_full2$race\_eth <- as.factor(PCAdata\_full2$race\_eth)  
PCAdata\_full2$gender <- as.factor(PCAdata\_full2$gender)  
PCAdata\_full2$age\_group <- as.factor(PCAdata\_full2$age\_group)

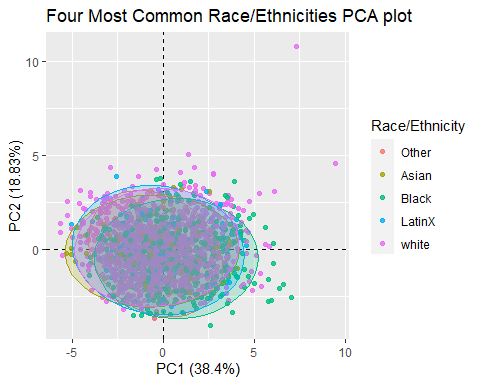
PCAdata\_race\_eth <- PCAdata\_full2  
  
PCAdata\_race\_eth$race\_eth <-   
 recode\_factor(PCAdata\_race\_eth$race\_eth, 'AIAN'= "Other",  
 'NHOPI' = "Other",  
 'PTNS' = "Other")  
  
PCAdata\_race\_eth\_nas <- PCAdata\_full1 %>%   
 mutate(race\_eth = na\_if\_in(race\_eth, c("AIAN", "Other", "NHOPI", "PTNS")))  
  
str(PCAdata\_race\_eth$race\_eth)

## Factor w/ 5 levels "Other","Asian",..: 3 5 5 5 5 5 3 3 5 5 ...

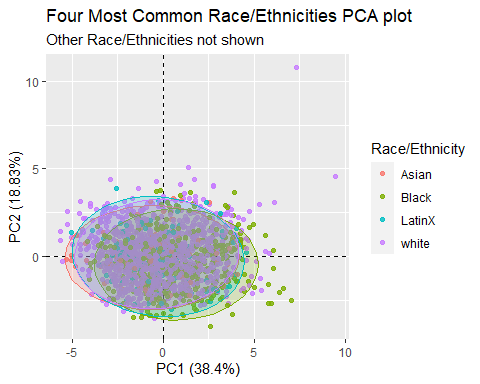
str(PCAdata\_race\_eth\_nas$race\_eth)

## chr [1:1711] "Black" "white" "white" "white" "white" "white" "Black" ...

ggplot(data = PCAdata\_race\_eth, aes(x = pc1, y = pc2, color = race\_eth)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +   
 stat\_ellipse(geom="polygon", aes(fill = race\_eth),   
 alpha = 0.2,   
 show.legend = FALSE,  
 level = 0.95) +  
 #theme\_minimal() +  
 #theme(panel.grid = element\_blank(), panel.border = element\_rect(fill= "transparent"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title="Four Most Common Race/Ethnicities PCA plot",  
 color = "Race/Ethnicity")



PCAdata\_race\_eth\_nas %>%   
 drop\_na(race\_eth) %>%   
 ggplot(aes(x = pc1, y = pc2, color = race\_eth)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +   
 stat\_ellipse(geom="polygon", aes(fill = race\_eth),   
 alpha = 0.2,   
 show.legend = FALSE,  
 level = 0.95) +  
 #theme\_minimal() +  
 #theme(panel.grid = element\_blank(), panel.border = element\_rect(fill= "transparent"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title="Four Most Common Race/Ethnicities PCA plot",  
 subtitle="Other Race/Ethnicities not shown",  
 color = "Race/Ethnicity")



Just Asian and Black?

PCAdata\_race\_eth1 <- PCAdata\_race\_eth  
  
PCAdata\_race\_eth1$race\_eth <-  
 recode\_factor(PCAdata\_race\_eth1$race\_eth, 'white'= "Other",  
 'LatinX' = "Other")  
  
PCAdata\_race\_eth\_nas1 <- PCAdata\_race\_eth\_nas %>%   
 mutate(race\_eth = na\_if\_in(race\_eth, c("white", "LatinX")))  
  
str(PCAdata\_race\_eth1$race\_eth)

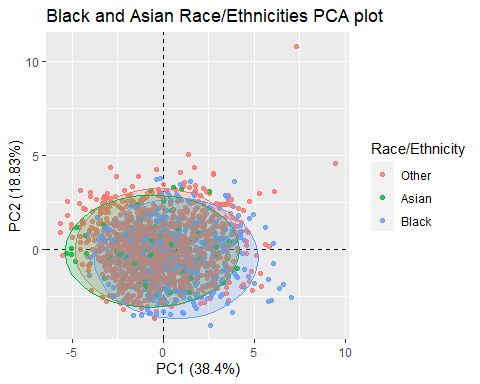
## Factor w/ 3 levels "Other","Asian",..: 3 1 1 1 1 1 3 3 1 1 ...

str(PCAdata\_race\_eth\_nas1$race\_eth)

## chr [1:1711] "Black" NA NA NA NA NA "Black" "Black" NA NA NA NA NA NA ...

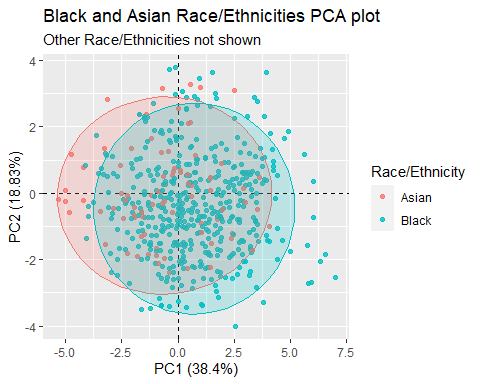
ggplot(data = PCAdata\_race\_eth1, aes(x = pc1, y = pc2, color = race\_eth)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +   
 stat\_ellipse(geom="polygon", aes(fill = race\_eth, na.rm=TRUE),   
 alpha = 0.2,   
 show.legend = FALSE,  
 level = 0.95) +  
 #theme\_minimal() +  
 #theme(panel.grid = element\_blank(), panel.border = element\_rect(fill= "transparent"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title="Black and Asian Race/Ethnicities PCA plot",  
 color = "Race/Ethnicity")

## Warning: Ignoring unknown aesthetics: na.rm

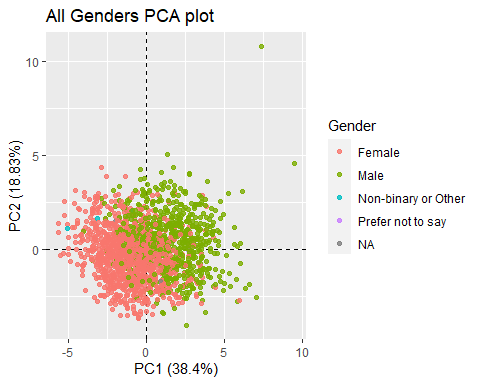


PCAdata\_race\_eth\_nas1 %>%   
 drop\_na(race\_eth) %>%   
 ggplot(aes(x = pc1, y = pc2, color = race\_eth)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +   
 stat\_ellipse(geom="polygon", aes(fill = race\_eth, na.rm=TRUE),   
 alpha = 0.2,   
 show.legend = FALSE,  
 level = 0.95) +  
 #theme\_minimal() +  
 #theme(panel.grid = element\_blank(), panel.border = element\_rect(fill= "transparent"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title="Black and Asian Race/Ethnicities PCA plot",  
 subtitle = "Other Race/Ethnicities not shown",  
 color = "Race/Ethnicity")

## Warning: Ignoring unknown aesthetics: na.rm

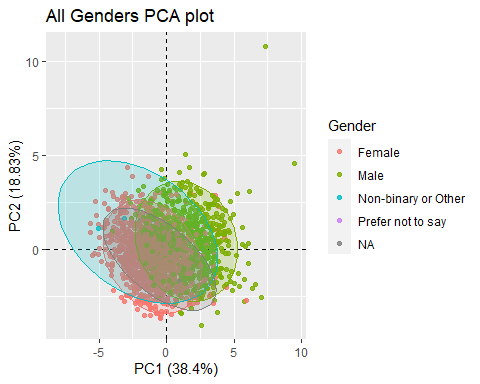


#checking to make sure this matches plot in PCA.Rmd  
ggplot(data = PCAdata\_full1, aes(x = pc1, y = pc2, color = gender)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Genders PCA plot",  
 color= "Gender")



ggplot(data = PCAdata\_full1, aes(x = pc1, y = pc2, color = gender)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +   
 stat\_ellipse(geom="polygon", aes(fill = gender),   
 alpha = 0.2,   
 show.legend = FALSE,  
 level = 0.95) +  
 #theme\_minimal() +  
 #theme(panel.grid = element\_blank(), panel.border = element\_rect(fill= "transparent"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title="All Genders PCA plot",  
 color= "Gender")

## Too few points to calculate an ellipse

 Need plot with just Male and Female

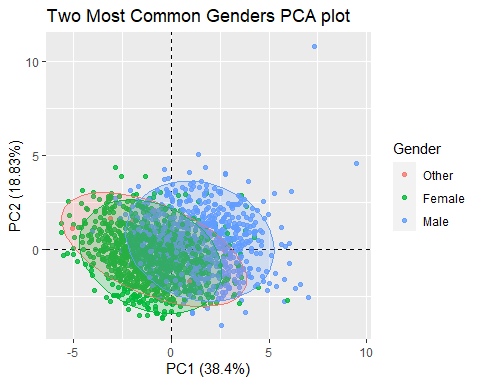
PCAdata\_gender <- PCAdata\_full2  
  
PCAdata\_gender$gender <-   
 recode\_factor(PCAdata\_gender$gender, 'Non-binary or Other'= "Other",  
 'Prefer not to say' = "Other")  
  
PCAdata\_gender$gender[is.na(PCAdata\_gender$gender)]="Other"  
  
PCAdata\_gender\_nas <- PCAdata\_full1 %>%   
 mutate(gender = na\_if\_in(gender, c("Non-binary or Other", "Prefer not to say")))  
  
str(PCAdata\_gender$gender)

## Factor w/ 3 levels "Other","Female",..: 3 2 3 3 3 3 3 3 3 3 ...

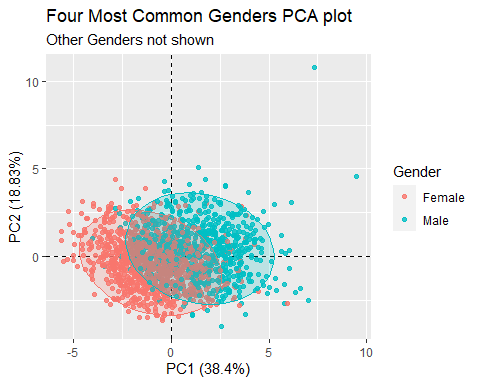
str(PCAdata\_gender\_nas$gender)

## chr [1:1711] "Male" "Female" "Male" "Male" "Male" "Male" "Male" "Male" ...

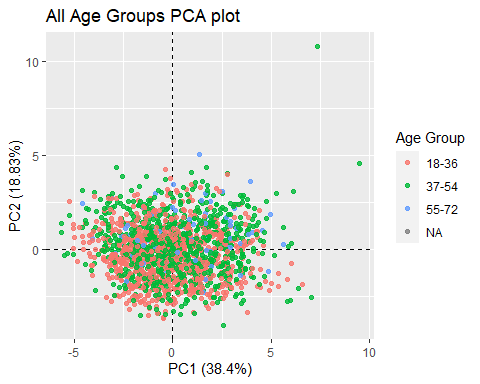
ggplot(data = PCAdata\_gender, aes(x = pc1, y = pc2, color = gender)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +   
 stat\_ellipse(geom="polygon", aes(fill = gender),   
 alpha = 0.2,   
 show.legend = FALSE,  
 level = 0.95) +  
 #theme\_minimal() +  
 #theme(panel.grid = element\_blank(), panel.border = element\_rect(fill= "transparent"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title="Two Most Common Genders PCA plot",  
 color = "Gender")



PCAdata\_gender\_nas %>%   
 drop\_na(gender) %>%   
 ggplot(aes(x = pc1, y = pc2, color = gender)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +   
 stat\_ellipse(geom="polygon", aes(fill = gender),   
 alpha = 0.2,   
 show.legend = FALSE,  
 level = 0.95) +  
 #theme\_minimal() +  
 #theme(panel.grid = element\_blank(), panel.border = element\_rect(fill= "transparent"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title="Four Most Common Genders PCA plot",  
 subtitle="Other Genders not shown",  
 color = "Gender")



#checking to make sure this matches plot in PCA.Rmd  
ggplot(data = PCAdata\_full1, aes(x = pc1, y = pc2, color = age\_group)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Age Groups PCA plot",  
 color= "Age Group")



ggplot(data = PCAdata\_full1, aes(x = pc1, y = pc2, color = age\_group)) +  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 geom\_point(alpha = 0.8) +   
 stat\_ellipse(geom="polygon", aes(fill = age\_group),   
 alpha = 0.2,   
 show.legend = FALSE,  
 level = 0.95) +  
 #theme\_minimal() +  
 #theme(panel.grid = element\_blank(), panel.border = element\_rect(fill= "transparent"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Age Groups PCA plot",  
 color= "Age Group")

## Too few points to calculate an ellipse

