PCA

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.3.6 ✔ purrr 0.3.4  
## ✔ tibble 3.1.8 ✔ dplyr 1.0.9  
## ✔ tidyr 1.2.0 ✔ stringr 1.4.0  
## ✔ readr 2.1.2 ✔ forcats 0.5.1  
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

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(scales)

##   
## Attaching package: 'scales'  
##   
## The following object is masked from 'package:purrr':  
##   
## discard  
##   
## The following object is masked from 'package:readr':  
##   
## col\_factor

#times new roman tables  
my\_ft\_theme <- function(ft, ...) {  
 # Remove vertical cell padding  
 ft <- padding(ft, padding.top = 0, padding.bottom = 0, part = "all")  
   
 # Change font to TNR 11  
 ft <- font(ft, fontname = "Times New Roman", part = "all")  
 ft <- fontsize(ft, part = "all", size = 12)  
 ft  
}

#remove identifiers and demographic data  
  
headscan\_full1<-read\_excel("C:\\Users\\19177\\OneDrive - Colostate\\Desktop\\Dissertation\\headscan\_dissertation\\headscan\_full1.xlsx")  
  
headscan\_full1$AA\_C <- headscan\_full1$AA\_C \*10  
headscan\_full1$BGl\_C <- headscan\_full1$BGl\_C \* 10  
headscan\_full1$BiW\_C <- headscan\_full1$BiW\_C \*10  
headscan\_full1$BiW\_L <- headscan\_full1$BiW\_L \*10  
headscan\_full1$ChCh\_C <- headscan\_full1$ChCh\_C \*10  
headscan\_full1$GoSub\_C <- headscan\_full1$GoSub\_C \*10  
headscan\_full1$NRB\_L <- headscan\_full1$NRB\_L \*10  
headscan\_full1$ProA\_L <- headscan\_full1$ProA\_L \*10  
headscan\_full1$ProA\_C <- headscan\_full1$ProA\_C \*10  
headscan\_full1$ProS\_C <- headscan\_full1$ProS\_C \*10  
headscan\_full1$ProS\_L <- headscan\_full1$ProS\_L \*10  
headscan\_full1$SelP\_C <- headscan\_full1$SelP\_C \*10  
headscan\_full1$SelP\_L <- headscan\_full1$SelP\_L \*10  
headscan\_full1$SelDH\_C <- headscan\_full1$SelDH\_C \*10  
headscan\_full1$SelM\_L <- headscan\_full1$SelM\_L \*10  
headscan\_full1$SnasM\_C <- headscan\_full1$SnasM\_C \*10  
headscan\_full1$SmanM\_C <- headscan\_full1$SmanM\_C \*10  
headscan\_full1$SmanM\_L <- headscan\_full1$SmanM\_L \*10  
headscan\_full1$SnasM\_L <- headscan\_full1$SnasM\_L \*10  
headscan\_full1$TrHO\_C <- headscan\_full1$TrHO\_C \*10  
headscan\_full1$TrEJ\_C <- headscan\_full1$TrEJ\_C \*10  
headscan\_full1$TrGo\_C <- headscan\_full1$TrGo\_C \*10  
headscan\_full1$TrSel\_C <- headscan\_full1$TrSel\_C \*10  
headscan\_full1$TrSman\_C <- headscan\_full1$TrSman\_C \*10  
headscan\_full1$TrSnas\_C <- headscan\_full1$TrSnas\_C \*10  
headscan\_full1$TrTr\_C <- headscan\_full1$TrTr\_C \*10  
headscan\_full1$TrTr\_L <- headscan\_full1$TrTr\_L \*10  
  
str(headscan\_full1)

## tibble [2,016 × 33] (S3: tbl\_df/tbl/data.frame)  
## $ ID : chr [1:2016] "400-20201012-002" "400-20201012-003" "400-20201012-004" "400-20201012-005" ...  
## $ AA\_C : num [1:2016] 65 55 70 58 67 60 59 59 65 65 ...  
## $ BGl\_C : num [1:2016] 315 289 293 313 288 306 320 NA 300 277 ...  
## $ BiW\_C : num [1:2016] 130 127 143 140 137 130 141 138 143 150 ...  
## $ BiW\_L : num [1:2016] 115 108 121 109 104 106 109 111 113 116 ...  
## $ ChCh\_C : num [1:2016] 62 64 68 70 70 70 67 69 67 63 ...  
## $ GoSub\_C : num [1:2016] 93 93 115 93 103 100 79 106 85 102 ...  
## $ NRB\_L : num [1:2016] 17 18 19 21 19 14 17 18 16 17 ...  
## $ ProA\_L : num [1:2016] 28 25 31 23 28 28 26 27 32 28 ...  
## $ ProA\_C : num [1:2016] 31 27 33 27 31 29 27 29 34 31 ...  
## $ ProS\_C : num [1:2016] 18 20 14 13 22 22 19 14 26 24 ...  
## $ ProS\_L : num [1:2016] 17 18 14 13 20 20 18 12 24 22 ...  
## $ SelP\_C : num [1:2016] 42 41 51 45 47 48 46 42 47 44 ...  
## $ SelP\_L : num [1:2016] 42 41 51 44 47 48 46 41 46 44 ...  
## $ SelDH\_C : num [1:2016] 15 9 9 11 13 15 9 9 12 14 ...  
## $ SelM\_L : num [1:2016] 122 99 130 115 119 126 117 112 117 117 ...  
## $ SnasM\_C : num [1:2016] 82 55 84 74 73 80 78 76 64 75 ...  
## $ SmanM\_C : num [1:2016] 59 51 45 43 33 34 55 37 61 41 ...  
## $ SmanM\_L : num [1:2016] 55 50 45 42 33 34 50 36 59 40 ...  
## $ SnasM\_L : num [1:2016] 75 53 78 69 67 76 69 71 62 69 ...  
## $ TrHO\_C : num [1:2016] 179 163 169 166 159 162 169 NA 167 166 ...  
## $ TrEJ\_C : num [1:2016] 40 32 39 29 46 42 29 32 29 33 ...  
## $ TrGo\_C : num [1:2016] 84 57 70 61 68 70 75 61 67 64 ...  
## $ TrSel\_C : num [1:2016] 149 138 150 133 140 151 140 138 156 143 ...  
## $ TrSman\_C : num [1:2016] 177 145 178 147 157 164 149 159 151 160 ...  
## $ TrSnas\_C : num [1:2016] 163 142 167 145 152 157 148 149 157 NA ...  
## $ TrTr\_C : num [1:2016] 296 276 292 273 279 300 283 275 307 286 ...  
## $ TrTr\_L : num [1:2016] 155 141 156 149 146 146 147 151 157 144 ...  
## $ coder : chr [1:2016] "Kayna" "Kayna" "Kayna" "Kayna" ...  
## $ age : num [1:2016] 31 49 49 34 49 55 26 18 25 27 ...  
## $ gender : chr [1:2016] "Male" "Female" "Male" "Male" ...  
## $ race\_eth : chr [1:2016] "Black" "white" "white" "white" ...  
## $ age\_group: chr [1:2016] "18-36" "37-54" "37-54" "18-36" ...

PCAdata\_num <- headscan\_full1[c(1,2,4,5,7,8,12,14,16,17,25,27,28)]  
  
PCAdata\_full <- headscan\_full1[c(1,2,4,5,7,8,12,14,16,17,25,27,28,31,32,33)]

measureNAs <- read\_excel("C:\\Users\\19177\\OneDrive - Colostate\\Desktop\\Dissertation\\headscan\_dissertation\\measureNAs.xlsx")  
  
PCA\_NAs <- slice(measureNAs, c(1,3,4,6,7,11,13,15,16,24,26,27))  
  
#Size 12 Table TNR  
flextable(PCA\_NAs) %>%  
 my\_ft\_theme()%>%   
 bold(part = "header") %>%   
 set\_caption("NA values for PCA Measurement Locations, total 891 NA values") %>%   
 autofit() %>%   
 set\_header\_labels(values = list(measure\_name = "Measurement Location",  
 measureNAprops = "Proportion of NA values",  
 measureNAsums = "Count of NA values"))

**Table** : NA values for PCA Measurement Locations, total 891 NA values

| **Measurement Location** | **Proportion of NA values** | **Count of NA values** |
| --- | --- | --- |
| AA\_C | 0.0084 | 17 |
| BiW\_C | 0.0084 | 17 |
| BiW\_L | 0.0084 | 17 |
| GoSub\_C | 0.0630 | 127 |
| NRB\_L | 0.0074 | 15 |
| ProS\_L | 0.0094 | 19 |
| SelP\_L | 0.0074 | 15 |
| SelM\_L | 0.1111 | 224 |
| SnasM\_C | 0.1171 | 236 |
| TrSman\_C | 0.0655 | 132 |
| TrTr\_C | 0.0188 | 38 |
| TrTr\_L | 0.0169 | 34 |

PCAdata\_num <- PCAdata\_num %>% drop\_na()  
  
str(PCAdata\_num)

## tibble [1,711 × 13] (S3: tbl\_df/tbl/data.frame)  
## $ ID : chr [1:1711] "400-20201012-002" "400-20201012-003" "400-20201012-004" "400-20201012-005" ...  
## $ 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 ...

PCAdata\_full <- left\_join(PCAdata\_num, PCAdata\_full, by= "ID")  
  
PCAdata\_full <- PCAdata\_full %>%   
 rename(AA\_C = "AA\_C.x",  
 BiW\_C = "BiW\_C.x",  
 BiW\_L = "BiW\_L.x",  
 GoSub\_C = "GoSub\_C.x",  
 NRB\_L = "NRB\_L.x",  
 ProS\_L = "ProS\_L.x",  
 SelP\_L = "SelP\_L.x",  
 SelM\_L = "SelM\_L.x",  
 SnasM\_C = "SnasM\_C.x",  
 TrSman\_C = "TrSman\_C.x",  
 TrTr\_C = "TrTr\_C.x",  
 TrTr\_L = "TrTr\_L.x")  
  
PCAdata\_full <- PCAdata\_full[-c(14:25)]

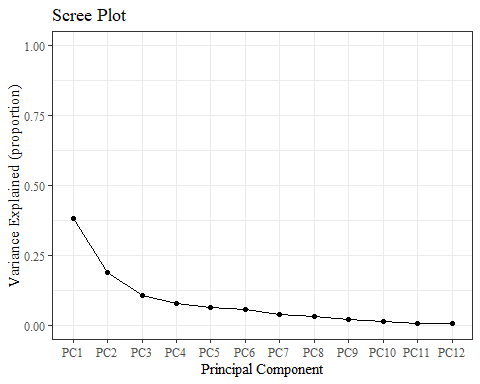
PCAdata\_num <- PCAdata\_num[-c(1)]  
PCAdata\_full <- PCAdata\_full[-c(1)]

#https://www.statology.org/principal-components-analysis-in-r/  
  
#calculate principal components  
pca\_res <- prcomp(PCAdata\_num, scale=TRUE)  
  
#reverse the signs (R calculates eigenvectors in negative direction)  
pca\_res$rotation <- -1\*pca\_res$rotation  
  
#display PCs  
pca\_res$rotation

## PC1 PC2 PC3 PC4 PC5 PC6  
## AA\_C 0.22968624 -0.3868735 -0.26540780 -0.01798014 0.3597350 -0.117452456  
## BiW\_C 0.32702777 0.3102392 -0.12117472 0.34097274 0.1653129 0.229856969  
## BiW\_L 0.32525795 0.3649563 0.07936387 0.30803903 0.1559941 0.194634486  
## GoSub\_C 0.24287218 -0.3295897 0.40269831 -0.15877720 -0.4054882 0.070815668  
## NRB\_L 0.13265539 0.2714656 0.21072639 -0.61614538 0.5031755 -0.388497864  
## ProS\_L 0.09743472 -0.4085516 -0.26521844 -0.28449031 0.3228209 0.634712672  
## SelP\_L 0.20019321 -0.1232991 -0.62381777 0.05046997 -0.2392005 -0.461990724  
## SelM\_L 0.35585108 0.1840093 -0.29161257 -0.30163508 -0.3191582 0.028258053  
## SnasM\_C 0.29389129 0.3496853 -0.08513569 -0.32694951 -0.2246200 0.246988968  
## TrSman\_C 0.36022402 -0.2306176 0.31811893 -0.07163661 -0.2000426 -0.005140471  
## TrTr\_C 0.37289546 -0.1122401 0.15154213 0.19906265 0.1569710 -0.192425589  
## TrTr\_L 0.35571715 -0.1858114 0.16144704 0.23991985 0.1513895 -0.156176046  
## PC7 PC8 PC9 PC10 PC11  
## AA\_C 0.2224279 -0.71916998 0.024874523 -0.001444158 -0.05755985  
## BiW\_C -0.3525145 -0.16659137 -0.017611124 -0.004791259 -0.25437831  
## BiW\_L -0.2342117 -0.06440236 0.006374994 0.087257047 0.22988518  
## GoSub\_C -0.3267249 -0.20998425 0.014994823 0.567858514 -0.02402646  
## NRB\_L -0.2636098 0.05795526 -0.047458486 0.054504503 -0.00022585  
## ProS\_L -0.1607298 0.36148483 0.018685342 0.026925148 0.08444620  
## SelP\_L -0.3612046 0.17110479 -0.023354513 0.055156683 0.34421415  
## SelM\_L 0.1575824 0.10963966 0.003271860 -0.034295764 -0.67482082  
## SnasM\_C 0.4424577 -0.15224660 -0.043620461 0.061194178 0.53447373  
## TrSman\_C -0.1811630 -0.03928898 0.007543588 -0.790664523 0.10522237  
## TrTr\_C 0.3020792 0.32959419 0.705297880 0.130657227 0.01257406  
## TrTr\_L 0.3069028 0.31323796 -0.704445372 0.126559413 -0.02577442  
## PC12  
## AA\_C 0.113976570  
## BiW\_C -0.607153697  
## BiW\_L 0.691452321  
## GoSub\_C -0.038264398  
## NRB\_L -0.049800016  
## ProS\_L 0.023073259  
## SelP\_L -0.023563501  
## SelM\_L 0.259625442  
## SnasM\_C -0.235560763  
## TrSman\_C -0.006047486  
## TrTr\_C -0.097642160  
## TrTr\_L -0.052638309

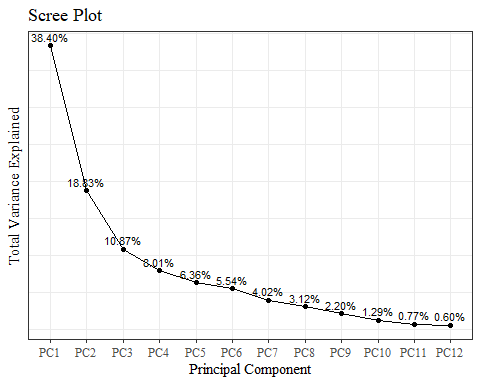
var\_explained = pca\_res$sdev^2 / sum(pca\_res$sdev^2)  
  
var\_explained\_total <- var\_explained

qplot(c(1:12), var\_explained) +   
 geom\_line() +   
 theme\_bw()+  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 xlab("Principal Component") +   
 ylab("Variance Explained (proportion)") +  
 ggtitle("Scree Plot") +  
 ylim(0, 1) +  
 scale\_x\_discrete(limits=c("PC1","PC2","PC3","PC4","PC5","PC6","PC7","PC8","PC9","PC10","PC11","PC12"))



var\_explained\_data <- data.frame(var\_explained)  
  
var\_explained\_data <- var\_explained\_data %>%   
 rename(v\_e = "var\_explained")  
  
  
var\_explained\_data$vep <- var\_explained\_data$v\_e  
  
var\_explained\_data <- rownames\_to\_column(var\_explained\_data, "PC\_num")  
  
var\_explained\_data$vep <- percent(var\_explained\_data$vep, accuracy=0.01)  
  
var\_explained\_data$PC\_num <- as.factor(var\_explained\_data$PC\_num)  
  
var\_explained\_data$PC\_num <-   
 recode\_factor(var\_explained\_data$PC\_num,   
 '1'= "PC1",  
 '2'= "PC2",  
 '3'= "PC3",  
 '4'= "PC4",  
 '5'= "PC5",  
 '6'= "PC6",  
 '7'= "PC7",  
 '8'= "PC8",  
 '9'= "PC9",  
 '10'= "PC10",  
 '11'= "PC11",  
 '12'= "PC12")

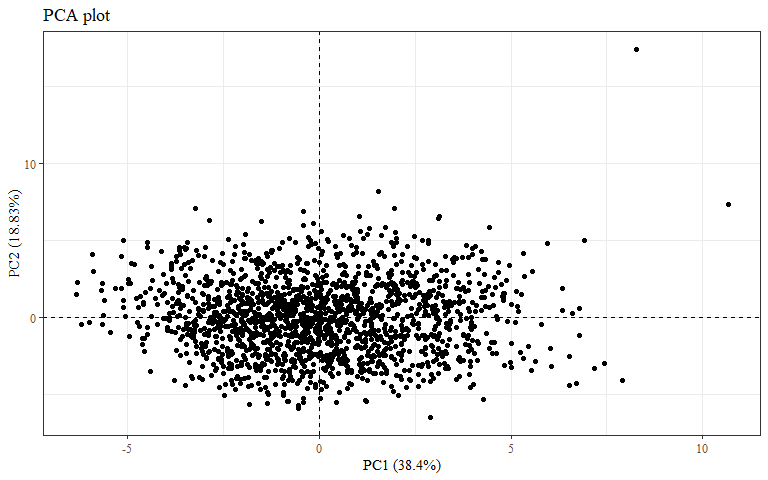
ggplot(data=var\_explained\_data, aes(x=PC\_num, y=v\_e, group=1)) +   
 geom\_line() +  
 geom\_point() +   
 geom\_text(aes(label=vep),  
 position= position\_dodge(0.9),  
 vjust = -0.5,   
 size = 3)+  
 theme\_bw()+  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 theme(axis.text.y=element\_blank(),  
 axis.ticks.y=element\_blank()   
 )+  
 labs(title="Scree Plot",  
 x="Principal Component",  
 y="Total Variance Explained")



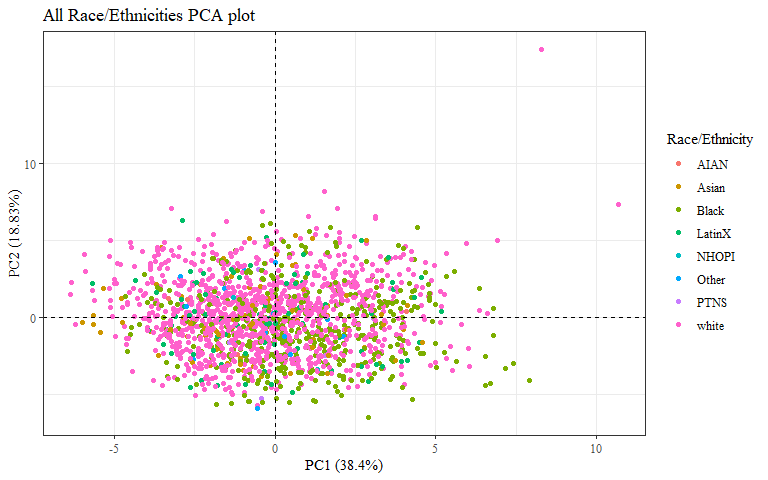
#making plot axis match the ggplot versions in PCA2.Rmd!!!   
#compare to PCAdata\_full1 pc1 and pc2 columns (should be same sign, here times 100)  
pca\_res$x <- pca\_res$x \* 100  
pca\_res$x[,1] <- pca\_res$x[,1] \* -1  
#pca\_res$x <- pca\_res$x[,2] \* 100  
  
#pca\_res$x

#<https://cran.r-project.org/web/packages/ggfortify/vignettes/plot_pca.html>

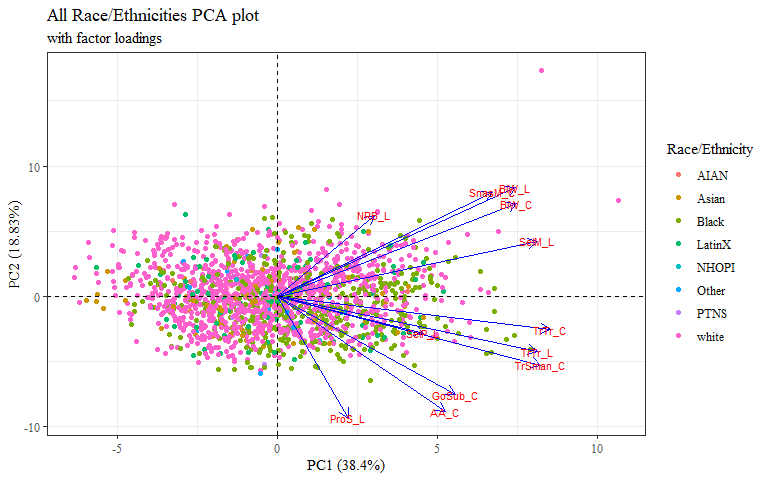
autoplot(pca\_res, data=PCAdata\_full)+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+   
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title="PCA plot")



autoplot(pca\_res, data=PCAdata\_full, colour="race\_eth")+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman")) +  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Race/Ethnicities PCA plot",  
 color = "Race/Ethnicity")

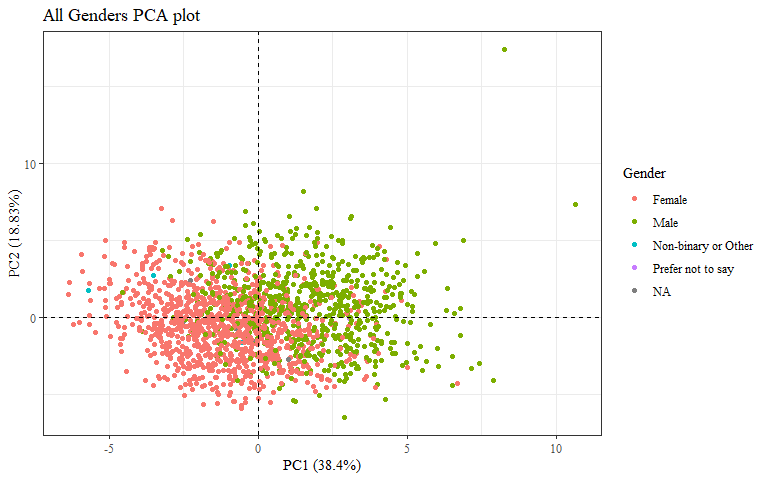


autoplot(pca\_res, data=PCAdata\_full, colour="race\_eth",  
 loadings = TRUE, loadings.colour = 'blue',  
 loadings.label = TRUE, loadings.label.size = 3)+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Race/Ethnicities PCA plot",  
 subtitle = "with factor loadings",  
 color = "Race/Ethnicity")

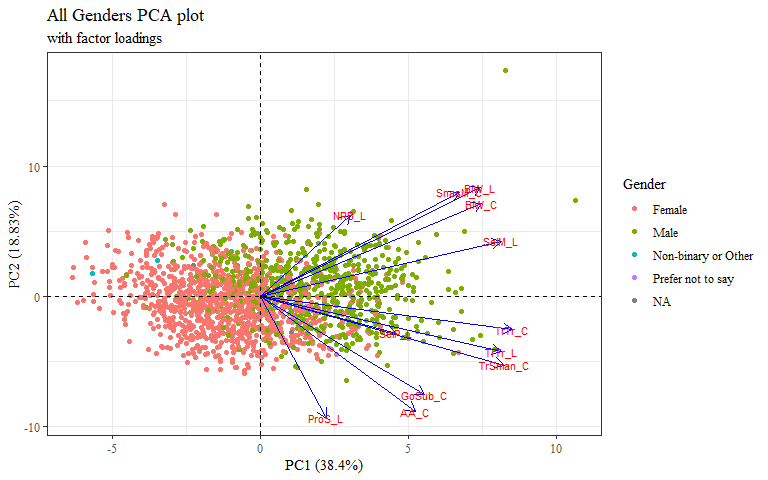
 arrows close together=higher correlation

COULD add ellipses (like in PCA2) to this graph with factor loadings.. but cannot add loadings to plot in PCA2 with ellipses

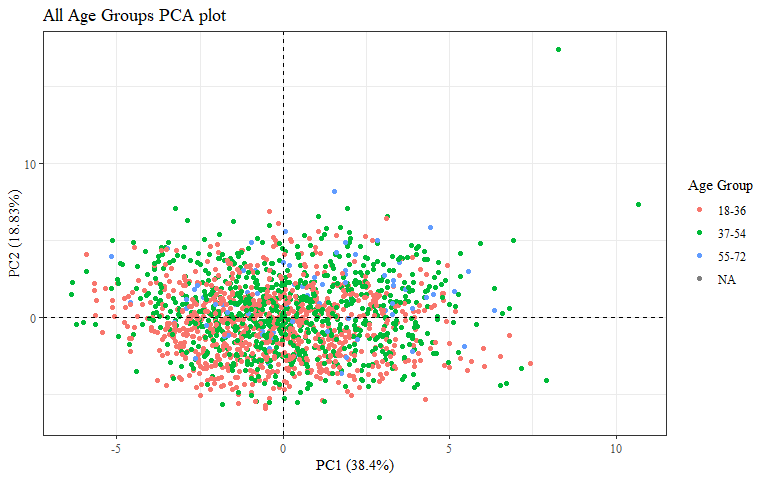
autoplot(pca\_res, data=PCAdata\_full, colour="gender")+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Genders PCA plot",  
 color = "Gender")



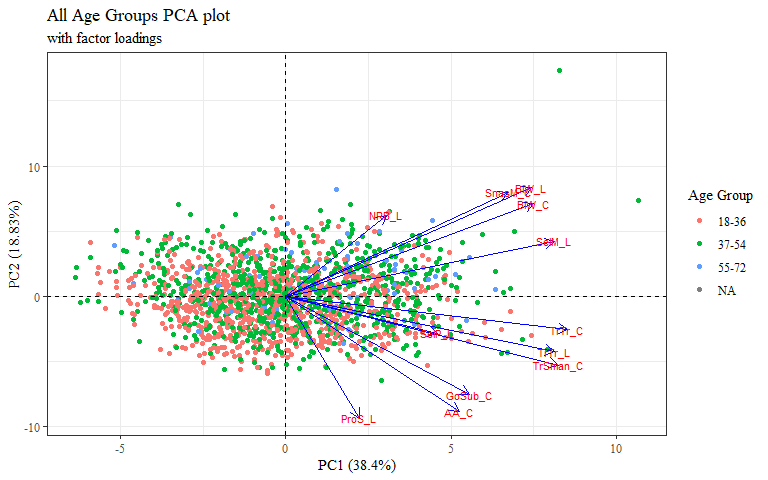
autoplot(pca\_res, data=PCAdata\_full, colour="gender",  
 loadings = TRUE, loadings.colour = 'blue',  
 loadings.label = TRUE, loadings.label.size = 3)+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Genders PCA plot",  
 subtitle = "with factor loadings",  
 color = "Gender")



autoplot(pca\_res, data=PCAdata\_full, colour="age\_group")+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Age Groups PCA plot",  
 color = "Age Group")



autoplot(pca\_res, data=PCAdata\_full, colour="age\_group",  
 loadings = TRUE, loadings.colour = 'blue',  
 loadings.label = TRUE, loadings.label.size = 3)+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Age Groups PCA plot",  
 subtitle = "with factor loadings",  
 color = "Age Group")



write\_xlsx(PCAdata\_full, "C:\\Users\\19177\\OneDrive - Colostate\\Desktop\\Dissertation\\headscan\_dissertation\\PCAdata\_full.xlsx")