PCA-imputed

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) #autoplot  
library(scales) #percent()

##   
## 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  
}

#PCAdata\_num <- PCAdata\_num %>% drop\_na()  
PCAdata\_full<-read\_excel("C:\\Users\\19177\\OneDrive - Colostate\\Desktop\\Dissertation\\headscan\_dissertation\\chosen\_imputed.xlsx")  
  
  
  
PCAdata\_num <- select\_if(PCAdata\_full, is.numeric)  
  
str(PCAdata\_num)

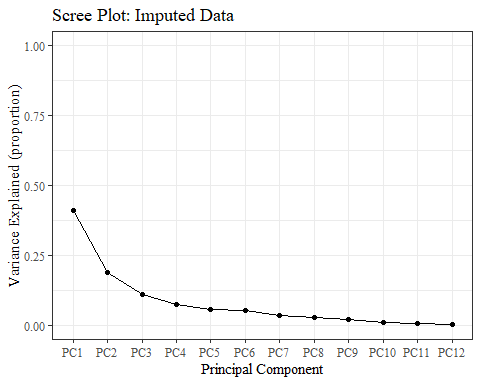
## tibble [2,016 × 12] (S3: tbl\_df/tbl/data.frame)  
## $ AA\_C : num [1:2016] 65 55 70 58 67 60 59 59 65 65 ...  
## $ BiW\_C : num [1:2016] 130 127 143 140 137 130 141 138 143 150 ...  
## $ BiW\_L : num [1:2016] 115 108 121 109 104 106 109 111 113 116 ...  
## $ GoSub\_C : num [1:2016] 93 93 115 93 103 100 79 106 85 102 ...  
## $ NRB\_L : num [1:2016] 17 18 19 21 19 14 17 18 16 17 ...  
## $ ProS\_L : num [1:2016] 17 18 14 13 20 20 18 12 24 22 ...  
## $ SelP\_L : num [1:2016] 42 41 51 44 47 48 46 41 46 44 ...  
## $ 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 ...  
## $ TrSman\_C: num [1:2016] 177 145 178 147 157 164 149 159 151 160 ...  
## $ 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 ...

#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 0.21983988 -0.38203375 -0.27650294 -0.11094529 0.3644535 -0.14390174  
## BiW\_C 0.32092078 0.30279971 -0.16193614 0.30392165 0.2957546 0.20889928  
## BiW\_L 0.32466094 0.34709978 0.05179616 0.28792541 0.2538292 0.18536527  
## GoSub\_C 0.26648474 -0.33014723 0.38628160 -0.06393095 -0.3704994 0.06050512  
## NRB\_L 0.12486946 0.25639409 0.24005629 -0.74437934 0.4208076 -0.26037704  
## ProS\_L 0.08667786 -0.44303872 -0.24128106 -0.25815749 0.2261863 0.68471801  
## SelP\_L 0.18996830 -0.09792091 -0.64940017 -0.01689198 -0.1891501 -0.46590898  
## SelM\_L 0.35448779 0.18558579 -0.27204999 -0.25589755 -0.3711357 0.12379561  
## SnasM\_C 0.30313919 0.36160429 -0.04850455 -0.22239913 -0.3254105 0.20584263  
## TrSman\_C 0.36598556 -0.22420410 0.28610088 -0.02058756 -0.1780400 -0.01761847  
## TrTr\_C 0.36930454 -0.11446455 0.15109100 0.15441161 0.1483059 -0.21623970  
## TrTr\_L 0.35381043 -0.18085353 0.16375668 0.21929801 0.1290741 -0.19788770  
## PC7 PC8 PC9 PC10 PC11  
## AA\_C 0.5414846 -0.50816202 0.007301270 -0.01839447 0.05632655  
## BiW\_C -0.2251254 -0.25863752 -0.009681616 0.03857301 -0.63537039  
## BiW\_L -0.1743878 -0.14270382 0.012767203 -0.15909578 0.67270881  
## GoSub\_C -0.2354904 -0.35994471 0.035461524 -0.57288334 -0.10490960  
## NRB\_L -0.2332141 0.01383911 -0.052017815 -0.05216160 -0.04274296  
## ProS\_L -0.2087531 0.29381400 0.007252144 -0.02305977 0.08679520  
## SelP\_L -0.4388781 0.02241733 -0.004001802 -0.07679620 0.17160076  
## SelM\_L 0.1205539 0.14348352 -0.013437369 0.03423066 -0.21102748  
## SnasM\_C 0.4324483 0.01328556 -0.039956219 -0.04300456 0.15831072  
## TrSman\_C -0.1968428 -0.18140018 0.006722809 0.78425874 0.11141547  
## TrTr\_C 0.1492648 0.45265471 0.702190459 -0.08371578 -0.07370665  
## TrTr\_L 0.1307425 0.42351666 -0.707647000 -0.10210532 -0.05779685  
## PC12  
## AA\_C -0.09178936  
## BiW\_C 0.18102165  
## BiW\_L -0.24689369  
## GoSub\_C 0.01681013  
## NRB\_L 0.01525775  
## ProS\_L 0.11541469  
## SelP\_L 0.22581342  
## SelM\_L -0.68288273  
## SnasM\_C 0.60131146  
## TrSman\_C 0.04824302  
## TrTr\_C 0.04896566  
## TrTr\_L 0.01980994

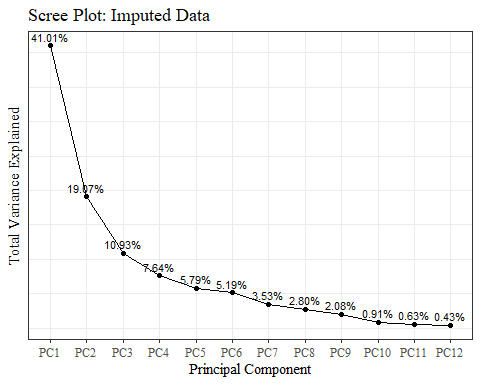
var\_explained = pca\_res$sdev^2 / sum(pca\_res$sdev^2)  
  
var\_explained\_total <- var\_explained

qplot(c(1:12), var\_explained) +   
 geom\_line() +   
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 xlab("Principal Component") +   
 ylab("Variance Explained (proportion)") +  
 ggtitle("Scree Plot: Imputed Data") +  
 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")

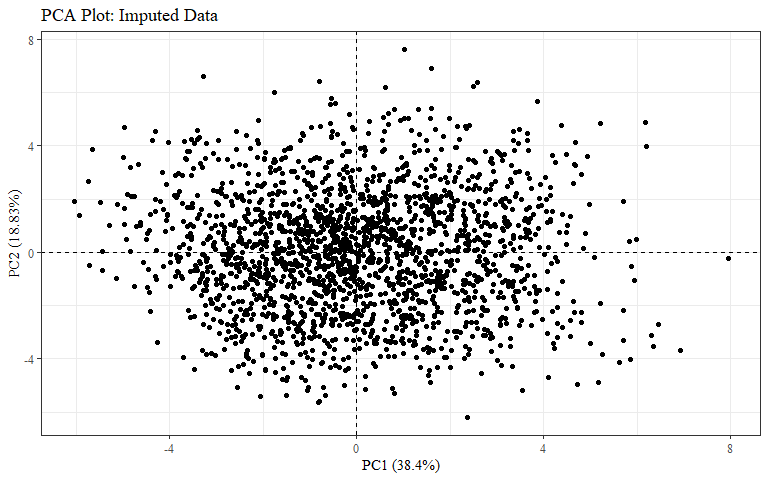
ggplot(data=var\_explained\_data, aes(x=PC\_num, y=v\_e, group=1)) +   
 geom\_line() +  
 geom\_point() +   
 geom\_text(aes(label=vep),  
 position= position\_dodge(0.9),  
 vjust = -0.5,   
 size = 3)+  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 theme(axis.text.y=element\_blank(),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



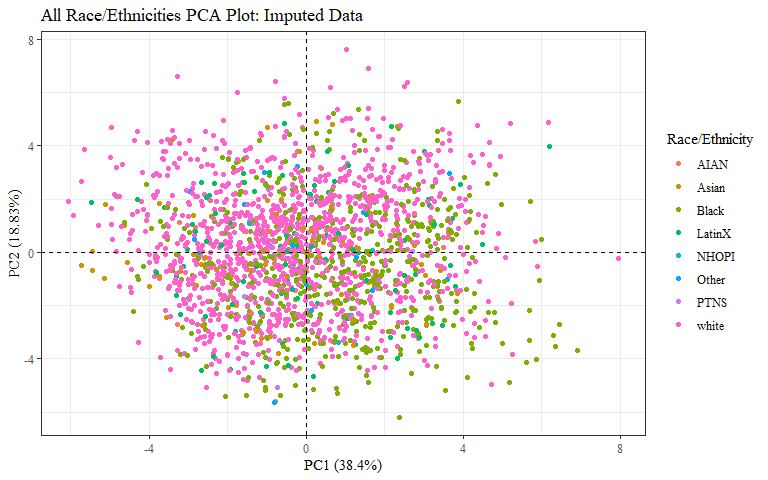
#making plot axis match the ggplot versions in PCA2.Rmd!!!   
#compare to PCAdata\_full1 pc1 and pc2 columns (should be same sign, here times 100)  
pca\_res$x <- pca\_res$x \* 100  
pca\_res$x[,1] <- pca\_res$x[,1] \* -1  
#pca\_res$x <- pca\_res$x[,2] \* 100  
  
#pca\_res$x

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

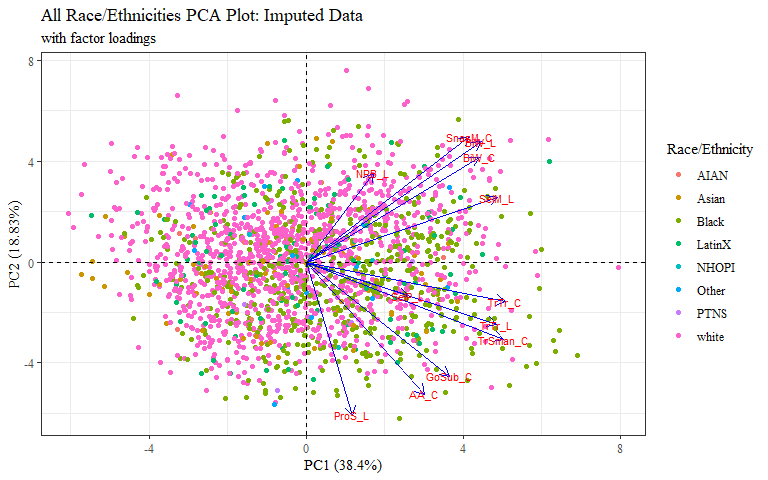
autoplot(pca\_res, data=PCAdata\_full)+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+   
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title="PCA Plot: Imputed Data")



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: Imputed Data",  
 color = "Race/Ethnicity")

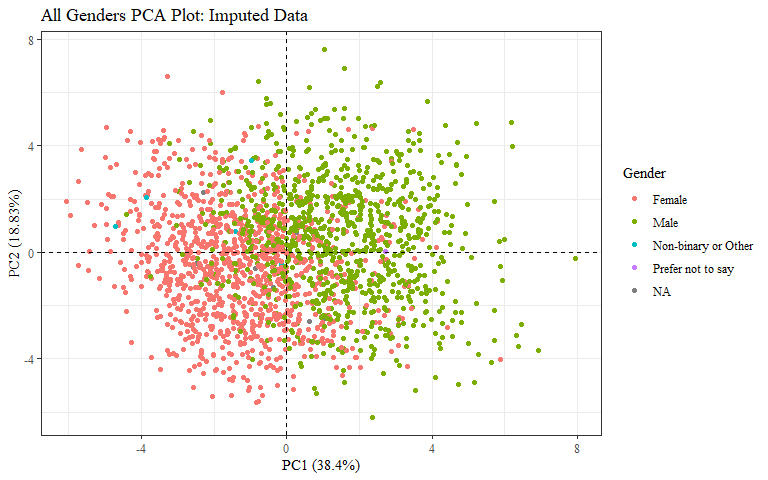


autoplot(pca\_res, data=PCAdata\_full, colour="race\_eth",  
 loadings = TRUE, loadings.colour = 'blue',  
 loadings.label = TRUE, loadings.label.size = 3)+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New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

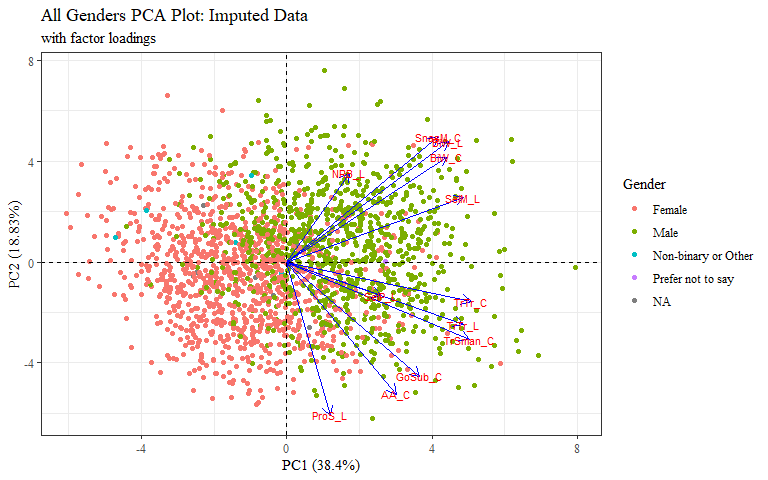
 arrows close together=higher correlation

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

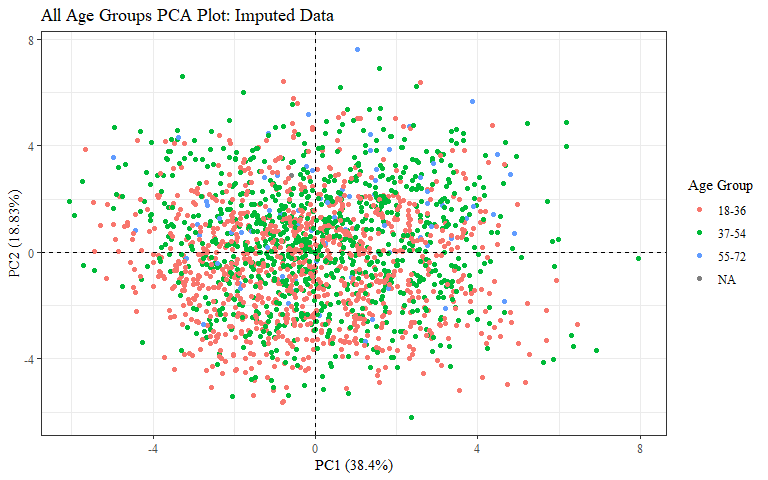
autoplot(pca\_res, data=PCAdata\_full, colour="gender")+  
 geom\_hline(yintercept = 0, lty = 2) +  
 geom\_vline(xintercept = 0, lty = 2) +  
 theme\_bw() + theme(text=element\_text(family= "Times New Roman"))+  
 labs(y="PC2 (18.83%)",  
 x="PC1 (38.4%)",  
 title = "All Genders PCA Plot: Imputed Data",  
 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: Imputed Data",  
 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: Imputed Data",  
 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: Imputed Data",  
 subtitle = "with factor loadings",  
 color = "Age Group")

