#This code is meant to read in the data, clean it, and run ANOVA and GLM modeling tests on various variables to come up with the best multivariable model to explain concentration levels

library(readr)

library(dplyr)

college\_food\_data <- read\_csv("college\_food\_data.csv")

View(college\_food\_data)

#Clean data

college\_food\_data$GPA <- as.integer(college\_food\_data$GPA)

college\_food\_data$healthy\_feeling <- as.integer(college\_food\_data$healthy\_feeling)

college\_food\_data$life\_rewarding <- as.integer(college\_food\_data$life\_rewarding)

#Create Concentration Index

college\_food\_data$concentration\_index = (college\_food\_data$GPA\*2.5)\*0.33 + (college\_food\_data$healthy\_feeling)\*0.33 + (college\_food\_data$life\_rewarding)\*0.33

#Modifying Comfort Values to fit Linear Model

college\_food\_data$comfort\_food\_reasons\_coded[college\_food\_data$comfort\_food\_reasons\_coded == "1"] <- 0

college\_food\_data$comfort\_food\_reasons\_coded[college\_food\_data$comfort\_food\_reasons\_coded == "2"] <- 0

college\_food\_data$comfort\_food\_reasons\_coded[college\_food\_data$comfort\_food\_reasons\_coded == "3"] <- 0

college\_food\_data$comfort\_food\_reasons\_coded[college\_food\_data$comfort\_food\_reasons\_coded == "5"] <- 0

college\_food\_data$comfort\_food\_reasons\_coded[college\_food\_data$comfort\_food\_reasons\_coded == "4"] <- 1

college\_food\_data$comfort\_food\_reasons\_coded[college\_food\_data$comfort\_food\_reasons\_coded == "6"] <- 1

college\_food\_data$comfort\_food\_reasons\_coded[college\_food\_data$comfort\_food\_reasons\_coded == "8"] <- 1

college\_food\_data$comfort\_food\_reasons\_coded[college\_food\_data$comfort\_food\_reasons\_coded == "9"] <- 1

college\_food\_data$comfort\_food\_reasons\_coded[college\_food\_data$comfort\_food\_reasons\_coded == "7"] <- 2

#Modifying Diet Values to fit Linear Model

college\_food\_data$diet\_current\_coded[college\_food\_data$diet\_current\_coded == "2"] <- 0

college\_food\_data$diet\_current\_coded[college\_food\_data$diet\_current\_coded == "1"] <- 2

college\_food\_data$diet\_current\_coded[college\_food\_data$diet\_current\_coded == "3"] <- 1

college\_food\_data$diet\_current\_coded[college\_food\_data$diet\_current\_coded == "4"] <- 1

#Modifying Eating Changes Values to fit Linear Model

college\_food\_data$eating\_changes\_coded[college\_food\_data$eating\_changes\_coded == "1"] <- 0

college\_food\_data$eating\_changes\_coded[college\_food\_data$eating\_changes\_coded == "3"] <- 1

college\_food\_data$eating\_changes\_coded[college\_food\_data$eating\_changes\_coded == "4"] <- 1

#Anova tests for 4 best predictor variables

aov\_1 <- aov(concentration\_index ~ diet\_current\_coded, college\_food\_data)

summary(aov\_1)

aov\_2 <- aov(concentration\_index ~ comfort\_food\_reasons\_coded, college\_food\_data)

summary(aov\_2)

aov\_3 <- aov(concentration\_index ~ eating\_changes\_coded, college\_food\_data)

summary(aov\_3)

aov\_4 <- aov(concentration\_index ~ vitamins, college\_food\_data)

summary(aov\_4)

#GLM and AIC tests for 4 best predictor variables

glm\_1 <- glm(concentration\_index ~ diet\_current\_coded, family = gaussian, college\_food\_data)

summary(glm\_1)

AIC(glm\_1)

glm\_2 <- glm(concentration\_index ~ comfort\_food\_reasons\_coded, family = gaussian, college\_food\_data)

summary(glm\_2)

AIC(glm\_2)

glm\_3 <- glm(concentration\_index ~ eating\_changes\_coded, family = gaussian, college\_food\_data)

summary(glm\_4)

AIC(glm\_4)

glm\_4 <- glm(concentration\_index ~ vitamins, family = gaussian, college\_food\_data)

summary(glm\_4)

AIC(glm\_4)

#Final Model GLM and AIC

final\_model <- glm(concentration\_index ~ diet\_current\_coded + comfort\_food\_reasons\_coded + eating\_changes\_coded + vitamins + diet\_current\_coded:comfort\_food\_reasons\_coded + diet\_current\_coded:eating\_changes\_coded + diet\_current\_coded:vitamins + comfort\_food\_reasons\_coded:eating\_changes\_coded + comfort\_food\_reasons\_coded:vitamins + eating\_changes\_coded:vitamins, family = gaussian, college\_food\_data)

summary(final\_model)

AIC(final\_model)