```r
# Question 3

# Read in data
data <- read.csv("motivation.csv")

# Add column of treatment combos to data
data$Combo <- paste(data$Priming, data$Conscious_Goal, sep="_")
kable(head(data))

# Get group means
grouped_means <- data %>%
  group_by(Combo) %>%
  summarise(mean_funds_raised = mean(Funds_Raised))

print(grouped_means)

# Make Priming and Conscious_Goal columns into factors
data$Priming <- as.factor(data$Priming)
data$Conscious_Goal <- as.factor(data$Conscious_Goal)

# Plot data for EDA
# Parallel dot plot
ggplot(data, aes(x=Combo, y=Funds_Raised, col=Combo)) +
  geom_point(size=1) +
  ggtitle("Parellel Dot Plot") +
  theme(axis.text.x = element_blank())

# Main effects plot
source("~/Desktop/stat_158/midterm/multiStripchart.R")
multiStripchart(data$Funds_Raised, facnames=c("Priming","Conscious_Goal"),
                data=data,lines=FALSE, main="Main Effects Plot", jitter=0.4,
                pch = 20, cex = 0.8)

# Interaction plot
interaction.plot(x.factor=data$Priming, trace.factor=data$Conscious_Goal,
                 response=data$Funds_Raised)

# QUESTION 4
# ANOVA

aov.rslt <- aov(Funds_Raised ~ Priming * Conscious_Goal, data = data)

# Get F-values
obs_F <- summary(aov.rslt)[[1]]$`F value`
summary(aov.rslt)

# Randomization

set.seed(123)

# Run permutations
df_rand <- function(data) {
  newY <- sample(data$Funds_Raised)
```

```r
  xaov <- (anova(lm(newY ~ Priming * Conscious_Goal, data=data)))
  x <- xaov[,"F value"]
  names(x) <- rownames(xaov)
  x <- head(x,-1)
  return(x)
}

# Function to plot
makePlot<-function(perm, factorName, xlim){
     hist(permRand[, factorName], freq=FALSE, ylim=c(0,1), xlim=xlim, breaks=100,
          main=factorName, xlab = paste("Permuted F-values of", factorName))
     obsAnova <- anova(lm(Funds_Raised ~ Priming * Conscious_Goal, data=data))
     obsF<-obsAnova[factorName,"F value"]
        abline(v=obsF, col="red")
     cat("Pvalue permutation of:",factorName,"\n")
     print(sum(permRand[,factorName]>=obsF)/length(permRand[,factorName]))
}

# Permute 1000 times
permRand <- t(replicate(5000, df_rand(data)))

# Plot
makePlot(permRand, "Priming", xlim=range(0:20))
makePlot(permRand, "Conscious_Goal", xlim=range(0:60))
makePlot(permRand, "Priming:Conscious_Goal", xlim=range(0:10))

# Contrast 1

# Change contrasts to zero-sum constraints
options(contrasts = c("contr.sum", "contr.poly"))

# Generate ANOVA
anova <- aov(Funds_Raised ~ Priming * Conscious_Goal, data)

# Numerator of t-statistic
diff_mean1 <- mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "No_Image"])
  mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Photo_Backdrop"]) +
  mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Collage_Top"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "No_Image"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Photo_Backdrop"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Collage_Top"])


# Get number of observations in each level
n <- sum(data$Conscious_Goal == "Raise_1200" & data$Priming == "No_Image")

# Contrast vector
contrast1 <- c(1, -1)[as.numeric(data$Conscious_Goal)] / n

# Get SE of contrast
se_contrast1 <- as.numeric(se.contrast(anova, matrix(contrast1, ncol = 1)))

# Observed t-statistic
```

```r
t.obs1 <- diff_mean1 / se_contrast1

# Function for t-statistic
t.stat1 <- function(data){
num <- mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "No_Image"]) +
  mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Photo_Backdrop"]) +
  mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Collage_Top"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "No_Image"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Photo_Backdrop"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Collage_Top"])

return(num / se_contrast1)
}

# Randomization inference
rerandomize.t1 <- function(data){
  data.new <- data
  data.new$Funds_Raised <- sample(data$Funds_Raised)
  t.stat1(data.new)
}

# Replicate randomization
null.distro1 <- replicate(5000, rerandomize.t1(data))

# Plot null dist & observation
hist(null.distro1, xlim=range(-6:8), breaks=80)
abline(v=t.obs1, col="red")

# P-value
paste("P-value: ", mean(null.distro1 > t.obs1))

# Contrast 2

set.seed(123)

# Numerator of t-statistic
diff_mean2 <- 0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Photo
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Photo_Backdrop"]
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Collage_Top"]) +
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Collage_Top"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "No_Image"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "No_Image"])


# Get number of observations in each level
n <- sum(data$Conscious_Goal == "Raise_1200" & data$Priming == "No_Image")

# Contrast vector
contrast2 <- c(0.5, 0.5, -1)[as.numeric(data$Priming)] / n

# Get SE of contrast
se_contrast2 <- as.numeric(se.contrast(anova, matrix(contrast2, ncol = 1)))
```

```r
# Observed t-statistic
t.obs2 <- diff_mean2 / se_contrast2

# Function for t-statistic
t.stat2 <- function(data){
num <- 0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Photo_Backdr
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Photo_Backdrop"]
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Collage_Top"]) +
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Collage_Top"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "No_Image"]) -
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "No_Image"])

return(num / se_contrast2)
}

# Randomization inference
rerandomize.t2 <- function(data){
  data.new <- data
  data.new$Funds_Raised <- sample(data$Funds_Raised)
  t.stat2(data.new)
}

# Replicate randomization
null.distro2 <- replicate(5000, rerandomize.t2(data))

# Plot null dist & observation
hist(null.distro2, xlim=range(-7:7), breaks=80)
abline(v=t.obs2, col="red")

# P-value
mean(null.distro2 > t.obs2)

# Contrast 3

set.seed(123)

# Numerator of t-statistic
diff_mean3 <- mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Collage_Top
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Collage_Top"]) -
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "No_Image"]) -
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "No_Image"]) -
  0 * mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Photo_Backdrop"]) -
  0 * mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Photo_Backdrop"])

# Get number of observations in each level
n <- sum(data$Conscious_Goal == "Raise_1200" & data$Priming == "No_Image")

# Contrast vector
contrast3 <- c(0, 1, -1)[as.numeric(data$Priming)] / n

# Get SE of contrast
se_contrast3 <- as.numeric(se.contrast(anova, matrix(contrast3, ncol = 1)))
```

```r
# Observed t-statistic
t.obs3 <- diff_mean3 / se_contrast3

# Function for t-statistic
t.stat3 <- function(data){
num <- mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Collage_Top"]) +
  mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Collage_Top"]) -
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "No_Image"]) -
  0.5 * mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "No_Image"]) -
  0 * mean(data$Funds_Raised[data$Conscious_Goal == "Raise_1200" & data$Priming == "Photo_Backdrop"]) -
  0 * mean(data$Funds_Raised[data$Conscious_Goal == "Do_Your_Best" & data$Priming == "Photo_Backdrop"])

return(num / se_contrast3)
}

# Randomization inference
rerandomize.t3 <- function(data){
  data.new <- data
  data.new$Funds_Raised <- sample(data$Funds_Raised)
  t.stat3(data.new)
}

# Replicate randomization
null.distro3 <- replicate(5000, rerandomize.t3(data))

# Plot null dist & observation
hist(null.distro3, xlim=range(0:10), breaks=80)
abline(v=t.obs3, col="red")

# P-value
mean(null.distro3 > t.obs3)
```