Sensory Study of Cold Brew Arabica Coffee Based on Brewing Time and Water Temperature

Adrian Cuyugan

2024-06-12

Introduction

A summary of R codes used to do statistical analysis on an experimental design case study. This is a supplemental document to be included in the research paper.

```
#### Library management ####
## Install necessary libraries
install.packages(c('car','psych','nortest','ggplot2','dplyr', 'corrplot',
                   'data.table', 'gridExtra', 'agricolae', 'car'),
                 repos = "http://cran.us.r-project.org")

## Load libraries
library(car)          # Regression analysis
library(psych)        # Descriptive stats
library(nortest)      # Anderson-Darling test for normality
library(ggplot2)      # Graphics
library(dplyr)        # Data preparation
library(data.table)   # Data preparation
library(gridExtra)    # Side by side plots
library(agricolae)    # Post-hoc tests
library(corrplot)     # Correlation plot
```

Dataset preparation

Randomize experimental units Randomization of 12 experimental units using standard normal distribution to form a 2 x 2 factorial experiment.

Treatment levels

1. Water temperature
    - Room
    - Fridge
2. Brewing time
    - 4 hours
    - 12 hours

```r
# Set random seed for reproducibility
set.seed(100)

## Create dataset of treatment levels
# Create standard normal random numbers
# Create 3 replicates of water temperature x brewing time
df <- data.frame(random = rnorm(n = 12, mean = 0, sd = 1),
                 temp = as.factor(c(rep('ROOM', each = 3),
                                    rep('FRIDGE', each = 3),
                                    rep('ROOM', each = 3),
                                    rep('FRIDGE', each = 3))),
                 time = as.factor(c(rep('4HR', each = 6),
                                    rep('8HR', each = 6))))

# Arrange random numbers in ascending order for randomization of respondents
df <- df %>% arrange(random)

# Remove randomization column
df <- df[,-1]

# Create respondent index after randomization
df <- data.frame(df, index = as.double(1:12))
```

Sensory scores Five sensory attributes were collected during coffee tasting experience scored on an unlabeled 5-point scale. These are:

1. Aroma
2. Flavor
3. Acidity
4. Body
5. Aftertaste

The sensory score is the sum of all sensory attributes.

```r
## Sensory score dataset based on respondents' answers ##
sensory <- data.frame(index = as.double(1:12),
                      aroma = as.numeric(c(5,3,5,4,3,3,4,4,2,4,2,3)),
                      flavor = as.numeric(c(5,4,4,4,4,4,5,5,4,5,4,4)),
                      acidity = as.numeric(c(5,5,5,4,4,3,5,5,3,5,1,5)),
                      body = as.numeric(c(4,4,5,3,4,5,5,4,3,5,3,5)),
                      aftertaste = as.numeric(c(4,5,5,3,4,3,5,5,3,5,1,5)))

# Row sums of Sensory Score
sensory$score <- rowSums(sensory[,2:6])

# Merge datasets
df <- merge(x = df, y = sensory, by.x = 'index')
```

Water temperature while brewing The water temperature of each treatment were monitored hourly to assess variation.

```
## Temperature Monitoring per Treatment
# Create dataset
monitor_df <- setNames(data.frame(t(data.frame(
  c('2024-05-31 16:15:00', 'Fridge', '4Hr', 13.1),
  c('2024-05-31 16:17:00', 'Fridge', '8Hr', 13),
  c('2024-05-31 16:20:00', 'Room', '4Hr', 29.7),
  c('2024-05-31 16:24:00', 'Room', '8Hr', 29.6),
  c('2024-05-31 17:25:00', 'Fridge', '4Hr', 14.3),
  c('2024-05-31 17:25:00', 'Fridge', '8Hr', 14.8),
  c('2024-05-31 17:27:00', 'Room', '4Hr', 31.1),
  c('2024-05-31 17:28:00', 'Room', '8Hr', 31.1),
  c('2024-05-31 18:34:00', 'Fridge', '4Hr', 11.3),
  c('2024-05-31 18:35:00', 'Fridge', '8Hr', 11.3),
  c('2024-05-31 18:31:00', 'Room', '4Hr', 31.6),
  c('2024-05-31 18:32:00', 'Room', '8Hr', 31.5),
  c('2024-05-31 19:37:00', 'Fridge', '4Hr', 8.1),
  c('2024-05-31 19:38:00', 'Fridge', '8Hr', 8.3),
  c('2024-05-31 19:39:00', 'Room', '4Hr', 31.9),
  c('2024-05-31 19:40:00', 'Room', '8Hr', 32),
  c('2024-05-31 20:34:00', 'Fridge', '4Hr', 6),
  c('2024-05-31 20:34:00', 'Fridge', '8Hr', 5.8),
  c('2024-05-31 20:30:00', 'Room', '4Hr', 32.1),
  c('2024-05-31 20:31:00', 'Room', '8Hr', 32.1),
  c('2024-05-31 21:25:00', 'Fridge', '8Hr', 5.3),
  c('2024-05-31 21:26:00', 'Room', '8Hr', 32.1),
  c('2024-05-31 22:32:00', 'Fridge', '8Hr', 5.7),
  c('2024-05-31 22:31:00', 'Room', '8Hr', 32.1),
  c('2024-05-31 23:28:00', 'Fridge', '8Hr', 4.9),
  c('2024-05-31 23:27:00', 'Room', '8Hr', 32),
  c('2024-06-01 00:18:00', 'Fridge', '8Hr', 6.2),
  c('2024-06-01 00:28:00', 'Room', '8Hr', 31.9)
  )), row.names = NULL),
  c('datetimestamp', 'temp', 'time', 'celcius'))

# Adjust datatype per column in monitor_df
monitor_df$datetimestamp <- as.POSIXct(monitor_df$datetimestamp, tz = 'Asia/Manila')
monitor_df$temp <- as.factor(monitor_df$temp)
monitor_df$time <- as.factor(monitor_df$time)
monitor_df$celcius <- as.double(monitor_df$celcius)
```

Exploratory data analysis

Summary statistics and descriptive plots.

Summary statistics

```
# Score by water temperature and brewing time
describeBy(x = df, group = df$temp, digits = 2)
describeBy(x = df, group = df$time, digits = 2)
```

Descriptive plots

Water temperature and brewing time Histogram and boxplot

```
## Descriptive plots
temp_hist <- ggplot(data = df, aes(x = score, col = temp, fill = temp)) +
  geom_histogram(binwidth = 1, alpha = 0.5) +
  ggtitle('Histogram of Sensory Score per Water Temperature') +
  facet_wrap(~ temp) +
  theme_bw() +
  theme(legend.position = "bottom",
        legend.title = element_text(size = 5),
        legend.text = element_text(size = 5))

time_hist <- ggplot(data = df, aes(x = score, col = time, fill = time)) +
  geom_histogram(binwidth = 1, alpha = 0.5) +
  ggtitle('Histogram of Sensory Score per Brewing Time') +
  facet_wrap(~ time) +
  theme_bw() +
  theme(legend.position = "bottom",
        legend.title = element_text(size = 5),
        legend.text = element_text(size = 5))

temp_box <- ggplot(data =  df, aes(x = temp, y = score)) +
  geom_boxplot(aes(col = temp), alpha = 0.5) +
  stat_summary(fun = mean, colour = 'black', geom = "point",
               shape = 18, size = 3, show.legend = F) +
  ggtitle('Boxplot of Sensory Score per Water Temperature') +
  theme_bw() +
  theme(legend.position = "bottom",
        legend.title = element_text(size = 6),
        legend.text = element_text(size = 6))

time_box <- ggplot(data =  df, aes(x = time, y = score)) +
  geom_boxplot(aes(col = time), alpha = 0.5) +
  stat_summary(fun = mean, colour = 'black', geom = "point",
               shape = 18, size = 3, show.legend = F) +
  ggtitle('Boxplot of Sensory Score per Brewing Time') +
  theme_bw() +
  theme(legend.position = "bottom",
        legend.title = element_text(size = 6),
        legend.text = element_text(size = 6))

# Arrange the histogram and boxplots into one plot
grid.arrange(temp_hist, time_hist,
             temp_box, time_box, nrow = 2, ncol = 2)
```

Sensory scores Boxplot of each sensory score per treatment.

```
## Distribution of Sensory Scores
# Create dataset from final to remove index and overall score
drop_cols <- c('index', 'score')
sensory_df <- df[, !(names(df) %in% drop_cols)]
```

```r
# Combine treatment levels into one column
sensory_df$treatment <- as.factor(paste(sensory_df$temp, sensory_df$time, sep = '_'))

# Remove Water Temperature and Brewing Time columns
sensory_df <- sensory_df[, -1:-2]

# Transform sensory_df from wide to long format
sensory_df_tall <- melt(setDT(sensory_df), id.vars = 'treatment')

# Boxplot of sensory scores faceted by brewing time
ggplot(data = sensory_df_tall, aes(x = value, y = variable)) +
  geom_boxplot(aes(col = treatment), alpha = 0.5) +
  stat_summary(fun = mean, col = 'black', geom = "point",
               shape = 18, size = 3, show.legend = F) +
  facet_wrap(~ treatment) +
  xlab('Score') + ylab('Sensory Attribute') +
  ggtitle('Boxplot of Each Sensory Score per Treatment') +
  theme_bw() +
  theme(legend.position = 'none')
```

Correlation Spearman correlation matrix plot of sensory attributes.

```r
# Compute correlation matrix
sensory_corr <- cor(sensory_df[,1:5], method = 'spearman')
corrplot(sensory_corr, method="number", mar=c(0,0,2,0), diag = F, type = 'upper',
         title = 'Spearman Correlation Matrix of Sensory Attributes')
```

Lineplot of temperature monitoring Lineplot of temperature monitoring per treatment.

```r
## Line plot of Temperature Monitoring per Treatment
ggplot(monitor_df, aes(x = datetimestamp, y = celcius)) +
  geom_line(aes(col = time:temp)) +
  facet_wrap(~time) +
  ylim(0, 35) +
  ggtitle('Temperature Monitoring per Treatment') +
  xlab('Timestamp') + ylab('Celcius') +
  theme_bw() +
  theme(legend.position = 'bottom')
```

Model

ANOVA

```r
## ANOVA
model <- aov(score ~ temp*time, data = df)
Anova(model, type = "III")
```

Residual analysis

1. Residuals vs fitted values / homogeneity of variance
2. Normal probability plot of residuals / normality of residuals
3. Standardized residuals vs Fitted values
4. Spread level plot

```
par(mfrow = c(1,2))
par(mar = c(5,4,4,2))
plot(model, which=c(1,2))
mtext("Residual Analysis", side = 3, line = -1, outer = T)
par(mfrow = c(1,1))

# Normality tests on residuals
shapiro.test(model$residuals)
ks.test(x = model$residuals, y = 'rnorm')
ad.test(model$residuals)

# Test of constancy of variance on residuals
leveneTest(score ~ temp*time, data = df)
```

Interaction effect

```
# Calculate means for each treatment combination
score_means <- df %>%
  group_by(temp, time) %>%
  summarise(Means = mean(score),
            SD = sd(score))
print(score_means)

# Interaction plot
ggplot(data = score_means, aes(x = temp, y = Means, col = time, group = time)) +
  geom_point(size = 4) + geom_line() +
  ggtitle('Interaction Plot of Sensory Score by Water Temperature and Brewing Temperature') +
  theme_bw()
```

Post-hoc tests

1. Scheffe's Test
2. Tukey HSD Test
3. Duncan's New Multiple Range Test
4. Fisher's LSD Test

```
# Scheffe's test
scheffe <- scheffe.test(model_aov, trt = c('temp', 'time'), console = T)
print(scheffe)

# Tukey HSD
tukey <- HSD.test(model_aov, trt = c('temp', 'time'), console = T)
print(tukey)
```

```r
# Duncan's new multiple range test
duncan <- duncan.test(model_aov, trt = c('temp', 'time'), console = T)
print(duncan)

# Least Significant difference
lsd <- LSD.test(model_aov, trt = c('temp', 'time'), console = T)
print(lsd)

# Plot post-hoc tests groups
par(mfrow = c(2,2))
par(mar = c(3,5,6,1))

plot(scheffe, main = "Scheffe's Test")
plot(tukey, main = 'Tukey HSD Test')
plot(duncan, main = "Duncan's New Multiple Range Test")
plot(lsd, main = "Fisher's LSD Test")
mtext("Sensory Score Mean Differences Between Water Temperature and Brewing Time",
      side = 3, line = -1.2, outer = T)

par(mfrow = c(1,1))
```