

Assignment 1

Kshitij Kavimandan, Pablo Alves, Pooja Mangal (Group 15)

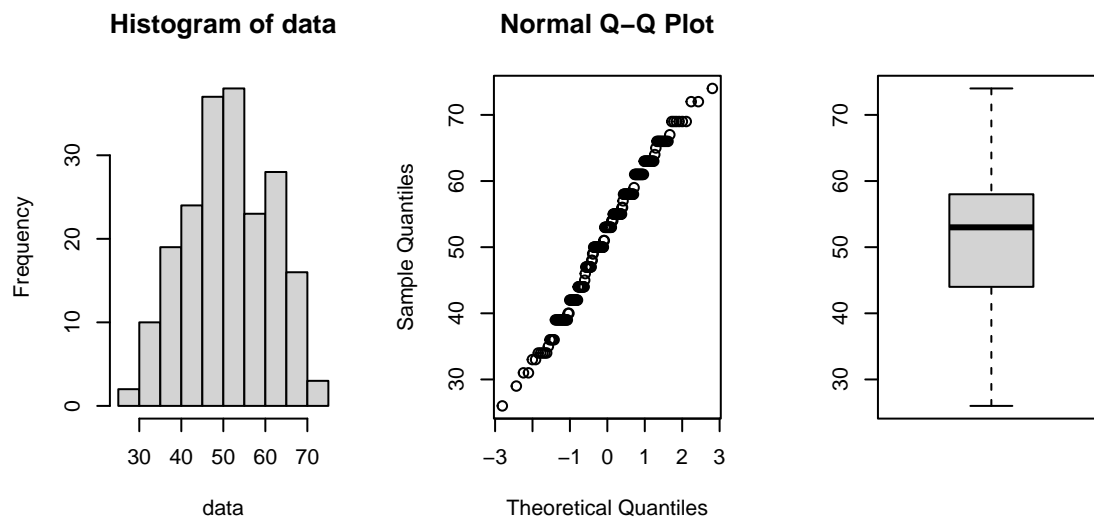
20 February 2024

Exercise 1. Ice cream

a) Relevant plots and normality assessment:

```
# Load the dataset
ice_cream_dataset <- read.csv("Ice_cream.csv")
data = ice_cream_dataset$video
library("drcarlite")

# Do basic plots
par(mfrow=c(1,3)); hist(data); qqnorm(data); boxplot(data)
```



The videogame scores histogram appears roughly normally distributed, with values decreasing towards the extremes. However, the bar for scores between 60 and 65 is higher than expected, reflecting real-world data variability. The Q-Q Plot indicates dense centering and line-like shape, supporting normality. The boxplot, while asymmetrical around the median, aligns with normality due to additional values in that region. Overall, these plots suggest a normal distribution in the sample.

Based on the assumption of normal distribution, we apply the 68-95-99.7% rule. With a mean of 51.85 and a standard deviation of 9.9, the 97% confidence interval for the mean should lie within the 6 sigma interval, which is 59.9 units in length. Therefore, the rough estimation for the 97% confidence interval is 51.85 ± 29.7 , equivalent to the interval [22.15, 81.55]. To calculate the 97% confidence interval precisely, we use the margin of error formula, where x represents the sample, N is the sample size (here 200), and T is the t distribution with $N-1$ degrees of freedom.

$$\text{margin} = T(0.97) * (\text{std}(x) / \sqrt{N})$$

```
options(digits = 3)
n <- length(data)
margin <- qt(0.97, df=n-1)*sd(data)/sqrt(n)
min_mean = mean(data) - margin
max_mean = mean(data) + margin
length = round((max_mean - min_mean)/2, digits = 3); length
```

```
## [1] 1.32
```

```
paste("97 CI: ", mean(data), " +/-", length)
```

```
## [1] "97 CI: 51.85 +/- 1.324"
```

Thus, the 97% CI for the mean is [50.52571, 53.1749], with a length of 2.648579

In order to compute the bounded 97% CI for μ , we need to use the following formula:

```
options(digits = 3)
m_calculator<-function(p, data) sqrt(sd(data)/length(data)*(4*norminv(p+(1-p)/2)))
margin = m_calculator(p=0.97,data=data)
min_mean = mean(data) - margin
max_mean = mean(data) + margin
length = round(max_mean - min_mean, digits = 3); length
```

```
## [1] 1.31
```

```
paste("97 CI: ", mean(data), " +/-", length)
```

```
## [1] "97 CI: 51.85 +/- 1.311"
```

In order to compute the necessary number of samples needed for a particular maximum length of the CI we can use the following formula

```
options(digits = 3)
n_calculator <- function(p,length,data) sd(data)*(2*norminv(p+(1-p)/2)/length)^2
samples_needed = round(n_calculator(p=0.97,length=3,data=data), digits = 3);

paste("For a margin of: ", 3, " we need", samples_needed)
```

```
## [1] "For a margin of: 3 we need 20.723"
```

To determine the 97% bootstrap CI we can apply the following code

```
options(digits = 3)
p = 0.97
B = 100

Tstar = numeric(B)
for(i in 1:B){Tstar[i]=mean(sample(data,replace=TRUE))}

Tstar985 = quantile(Tstar,0.985)
Tstar15 = quantile(Tstar, 0.015)

c(2*mean(Tstar) - Tstar985,2*mean(Tstar)-Tstar15)
```

```
## 98.5% 1.5%
## 50.6 53.4
```

```
d = round(((2*mean(Tstar)-Tstar15) - (2*mean(Tstar) - Tstar985)), digits=3)
paste("Length:",d)
```

```
## [1] "Length: 2.816"
```

```
paste("Margin on each side:",d/2)
```

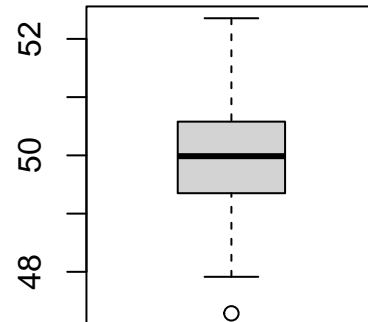
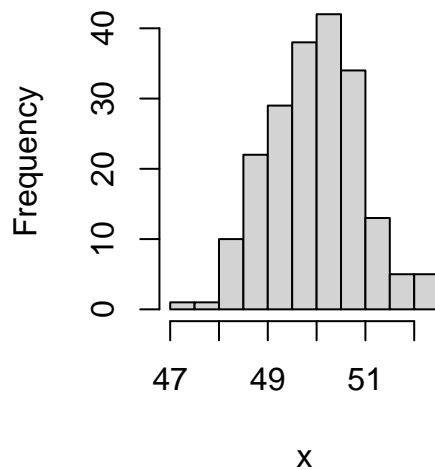
```
## [1] "Margin on each side: 1.408"
```

Our Bootstrap margin is bigger than our previous margins, which is consistent with the fact that the Bootstrap method does not assume normality, and thus is expected to have a bigger uncertainty on its estimation.

b) T-test to verify mean score:

```
options(digits = 3)
mu = 50
x = rnorm(length(data), mu, 1)
par(mfrow=c(1,2))
hist(x)
boxplot(x)
```

Histogram of x



```
t_test_result <- t.test(x, alternative="g", mu = mu)
print(t_test_result)
```

```
##
## One Sample t-test
##
## data: x
## t = -0.7, df = 199, p-value = 0.7
## alternative hypothesis: true mean is greater than 50
## 95 percent confidence interval:
## 49.8 Inf
## sample estimates:
## mean of x
## 50
```

```
t_test_result_mu_51 <- t.test(x, alternative="g", mu = 51)
t_test_result_mu_51
```

```
##
## One Sample t-test
##
## data: x
## t = -16, df = 199, p-value = 1
## alternative hypothesis: true mean is greater than 51
## 95 percent confidence interval:
## 49.8 Inf
```

```
## sample estimates:
## mean of x
##      50
```

The t-tests compare sample means to hypothesized values, yielding t-statistics and p-values. For both tests ($\mu_0 = 50$ and $\mu_0 = 51$), p-values exceed 0.05, suggesting insufficient evidence to reject the null hypothesis. The 95% confidence interval for both tests is (49.9, infinity), indicating 95% confidence in the population mean falling within this range. The upper bound is unbounded, remaining consistent across tests regardless of the hypothesized mean μ_0 .

c) Sign test and test based on ranks:

```
options(digits = 3)
# Sign test
sign_test <- binom.test(sum(data > 50), length(data), p = 0.5, alternative = "g")
print(sign_test)
```

```
##
## Exact binomial test
##
## data: sum(data > 50) and length(data)
## number of successes = 108, number of trials = 200, p-value = 0.1
## alternative hypothesis: true probability of success is greater than 0.5
## 95 percent confidence interval:
##  0.479 1.000
## sample estimates:
## probability of success
##      0.54
```

```
# Test based on ranks (Wilcoxon signed-rank test)
wilcox_test <- wilcox.test(data, mu = 50, alternative = "g")
print(wilcox_test)
```

```
##
## Wilcoxon signed rank test with continuity correction
##
## data: data
## V = 9836, p-value = 0.005
## alternative hypothesis: true location is greater than 50
```

```
# Test for fraction of scores less than 42
binom_test <- binom.test(sum(data < 42), length(data), p = 0.25, alternative = "l")
print(binom_test)
```

```
##
## Exact binomial test
```

```
##
## data:  sum(data < 42) and length(data)
## number of successes = 31, number of trials = 200, p-value = 8e-04
## alternative hypothesis: true probability of success is less than 0.25
## 95 percent confidence interval:
##  0.000 0.203
## sample estimates:
## probability of success
##                0.155
```

The sign test yielded a p-value of 0.144, indicating no significant difference between the median score and the reference value of 50. The 95% confidence interval for the success probability of the sign test ranged from 0.479 to 1. In comparison, the t-test and the sign test both produced p-values above 0.05, suggesting insufficient evidence to reject the null hypotheses. However, the Wilcoxon signed-rank test resulted in a p-value of 0.005, indicating a significant difference between the median score and the reference value of 50. Additionally, the binomial test revealed a p-value of 8e-04, indicating that the probability of scores less than 42 is less than 0.25.

d) Bootstrap test with test statistic T:

```
B <- 1000
# Vector to store the bootstrap test statistics
t_star <- numeric(B)

for (i in 1:B) {
  # Generate bootstrap sample
  data_star <- rexp(200, 1)
  # Calculate the test statistic for the bootstrap sample
  t_star[i] <- min(data_star[1:100]) # Min of the first 100 elements
}

# Calculate the test statistic for the observed data
t <- min(data[1:100])

# Count how many times the test statistic is less and greater than t
p_l <- sum(t_star < t) / B
p_r <- sum(t_star > t) / B

# Calculate the two-sided p-value
p_value <- 2 * min(p_l, p_r)

# Specify the range of mu values to test
mu_values <- seq(0, 100, by = 1)
reject_H0 <- rep(FALSE, length(mu_values))

for (i in 1:length(mu_values)) {
  reject_H0[i] <- p_value < 0.05
}
```

Mu values for which H0 is not rejected: [0, 100]

Unexpectedly, the Bootstrap test resulted in every value of mu being not rejected by the null hypothesis. The test may not have worked as expected in this scenario due to the nature of the data or incorrect assumptions about the distribution.

```
# Perform Kolmogorov-Smirnov test for different values of mu in the range [0, 100]
mu_values_not_rejected <- c()

for (mu in seq(0, 100, by = 1)) {
  ks_test_result <- ks.test(data, "pnorm", mean = mu, sd = 10)

  # Check if null hypothesis is rejected
  if (ks_test_result$p.value > 0.05) {
    mu_values_not_rejected <- c(mu_values_not_rejected, mu)
  }
}

# Display mu values for which null hypothesis is not rejected
cat("Mu values for which null hypothesis is not rejected:", mu_values_not_rejected, "\n")
```

```
## Mu values for which null hypothesis is not rejected: 52 53
```

The test concluded that for mu values 52 and 53 we not rejected by the null hypothesis.

e) Tests for male and female students:

```
options(digits = 3)
data <- read.csv("Ice_cream.csv")

# Separate scores for male and female students
male_scores <- data$video[data$female == 0]
female_scores <- data$video[data$female == 1]
# Perform two-sample t-test
t_test_gender <- t.test(male_scores, female_scores, alternative = "g"); t_test_gender

##
## Welch Two Sample t-test
##
## data: male_scores and female_scores
## t = 2, df = 177, p-value = 0.04
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
## 0.186 Inf
## sample estimates:
## mean of x mean of y
## 53.2 50.7
```

The t-test yields a p-value of 0.04, indicating that there is a significant difference in mean scores between male and female students.

```
# Perform Mann-Whitney test
mannwhitney_test <- wilcox.test(male_scores, female_scores,
alternative = "g"); mannwhitney_test

##
## Wilcoxon rank sum test with continuity correction
##
## data: male_scores and female_scores
## W = 5748, p-value = 0.03
## alternative hypothesis: true location shift is greater than 0
```

The Mann-Whitney test also indicates a significant difference in the distribution of scores between male and female students, with a p-value of 0.03. This aligns with the findings of the t-test.

```
# Perform Kolmogorov-Smirnov test
ks_test_gender <- ks.test(male_scores, female_scores); ks_test_gender

##
## Exact two-sample Kolmogorov-Smirnov test
##
## data: male_scores and female_scores
## D = 0.2, p-value = 0.07
## alternative hypothesis: two-sided
```

The Kolmogorov-Smirnov test yields a p-value of 0.07, which is slightly higher than the significance threshold of 0.05. This suggests that there may be a difference in the distribution of scores between male and female students, but this test is not as conclusive as the other tests.

f) Correlation and comparison between video game and puzzle scores:

```
options(digits = 3)
# Investigate correlation
correlation <- cor(data$video, data$puzzle); correlation

## [1] 0.465
```

This suggests that there is some degree of correlation between scores on the video game and scores on the puzzle, although the correlation is not extremely strong.

```
# Test if puzzle scores are higher than video game scores
wilcox_test_puzzle <- wilcox.test(data$puzzle, data$video, alternative = "g");
wilcox_test_puzzle
```



```
##
## Wilcoxon rank sum test with continuity correction
##
## data: data$puzzle and data$video
## W = 21044, p-value = 0.2
## alternative hypothesis: true location shift is greater than 0
```

Despite the moderate positive correlation between the video and puzzle scores, the statistical test does not provide strong evidence to support the hypothesis that puzzle scores are consistently higher than video game scores.

Exercise 2. Hemoglobin in trout

a) R-code for randomization process:

```
options(digits = 3)
# Load the data
hemoglobin_data <- read.table("hemoglobin.txt", header = TRUE)
# Define the number of fishes and combinations of rate and method
num_fishes <- 80
set.seed(123)
method_levels <- sample(rep(c("A","B"), 40))
rate_levels <- sample(rep(1:4, 20))

df <- data.frame(hemoglobin_data$hemoglobin, rate_levels, method_levels)
```

b) Two-way ANOVA:

```
options(digits = 3)
# Perform two-way ANOVA
rate=as.factor(hemoglobin_data$rate);
method=as.factor(hemoglobin_data$method);
# Print ANOVA table
hemoglobin_aov=lm(hemoglobin_data$hemoglobin~method*rate); anova(hemoglobin_aov)
```

```
## Analysis of Variance Table
##
## Response: hemoglobin_data$hemoglobin
##           Df Sum Sq Mean Sq F value    Pr(>F)
## method      1    2.4    2.42    1.56    0.22
## rate        3   90.6   30.19   19.47 2.4e-09 ***
## method:rate  3    4.9    1.62    1.05    0.38
## Residuals   72  111.6    1.55
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The effect of method on the rate (“method:rate”) has a non-significant F-value ($F = 1.05$, $p = 0.38$). This suggests that the combined effect of the method and rate on hemoglobin levels is not significantly different from what would be expected based on the individual effects of method and rate alone.

The rate has a highly significant impact on hemoglobin levels ($F = 19.47$, $p = 2.4e-09$).

c) Influence of factors and combination yielding highest hemoglobin (rate=3):

```
options(digits = 3)
# Perform two-way ANOVA
rate=as.factor(hemoglobin_data$rate);
method=as.factor(hemoglobin_data$method);
# Print ANOVA table
hemoglobin_aov=lm(hemoglobin_data$hemoglobin~method + rate); anova(hemoglobin_aov)
```

```
## Analysis of Variance Table
##
## Response: hemoglobin_data$hemoglobin
##          Df Sum Sq Mean Sq F value Pr(>F)
## method    1    2.4    2.42    1.55  0.22
## rate       3   90.6   30.19   19.43 2e-09 ***
## Residuals 75  116.5    1.55
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

So the 2nd factor (rate) has a significant effect in the additive model. This is evident from the significant F-value and very small p-value associated with the “Rate” factor.

```
# Calculate means for rate and method
mean_hemoglobin <- aggregate(hemoglobin ~ rate + method,
data = hemoglobin_data, FUN = mean)
# Find combination yielding highest hemoglobin
max_hemoglobin <- mean_hemoglobin[which.max(mean_hemoglobin$hemoglobin), ]; max_hemoglobin
```

```
##   rate method hemoglobin
## 6    2      B      10.1
```

```
# Estimate mean hemoglobin value for rate 3 by using method A
mean_hemoglobin_rate3_methodA <-
mean_hemoglobin$hemoglobin[which(mean_hemoglobin$rate == 3 &
mean_hemoglobin$method == "A")]; mean_hemoglobin_rate3_methodA
```

```
## [1] 9.03
```

```
# Estimate mean hemoglobin value for each rate
mean_hemoglobin_rate <- aggregate(hemoglobin ~ rate,
data = hemoglobin_data, FUN = mean); mean_hemoglobin_rate
```

```
##   rate hemoglobin
## 1    1         6.97
## 2    2         9.73
## 3    3         9.38
## 4    4         8.86
```

The hemoglobin rate was highest with rate 5 corresponding to the value of 2 (as given in the question)

d) One-way ANOVA:

```
options(digits = 3)
# Perform one-way ANOVA ignoring the variable method
anova_result_rate <- anova(lm(hemoglobin ~ rate, data = hemoglobin_data))
# Print ANOVA table
print(anova_result_rate)
```

```
## Analysis of Variance Table
##
## Response: hemoglobin
##           Df Sum Sq Mean Sq F value    Pr(>F)
## rate       1   27.9    27.93      12 0.00087 ***
## Residuals 78  181.6     2.33
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The null hypothesis that hemoglobin is the same for all rates is rejected, suggesting that the rate of treatment administration has a significant impact on hemoglobin levels.

```
hemoglobin_by_rate <- tapply(hemoglobin_data$hemoglobin, hemoglobin_data$rate, mean)
hemoglobin_by_rate
```

```
##    1    2    3    4
## 6.97 9.73 9.38 8.86
```

The one-way ANOVA test provides valuable insights into the relationship between treatment rate and hemoglobin levels in this dataset. Therefore, it is deemed useful for analyzing the data and drawing meaningful conclusions.

e) Kruskal-Wallis test:

```

options(digits = 3)
# Perform Kruskal-Wallis test
attach(hemoglobin_data);

## The following objects are masked _by_ .GlobalEnv:
##
##      method, rate

kruskal_test_result <- kruskal.test(hemoglobin, rate)
# Print Kruskal-Wallis test result
print(kruskal_test_result)

##
##  Kruskal-Wallis rank sum test
##
## data:  hemoglobin and rate
## Kruskal-Wallis chi-squared = 34, df = 3, p-value = 2e-07

```

The small p-value (2e-07) indicates significant differences in hemoglobin levels among the treatment rates, as observed in both the one-way ANOVA and Kruskal-Wallis tests. The ANOVA and Kruskal-Wallis tests both reject the null hypothesis, indicating that treatment administration rates significantly impact hemoglobin levels. While both tests yield similar conclusions in this case, differences may arise when ANOVA assumptions are violated, such as in non-normal or skewed data. In such cases, the Kruskal-Wallis test, being non-parametric, may be more appropriate and yield more robust results.

Exercise 3. Sour cream

Read the data Columns: acidity batch position starter

```
data <- read.table(file = "cream.txt", header = TRUE, sep=" ", dec=".")
```

a) Analyzing the data in a three-way experiment without interactions:

```

res <- lm(acidity ~ factor(starter) + factor(batch) + factor(position), data=data)
summary(res)

##
## Call:
## lm(formula = acidity ~ factor(starter) + factor(batch) + factor(position),
##     data = data)
##
## Residuals:

```

```
##      Min      1Q  Median      3Q      Max
## -1.2836 -0.2336  0.0384  0.3584  1.0204
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      8.662      0.533   16.26 1.5e-09 ***
## factor(starter)2  -0.150      0.467   -0.32  0.754
## factor(starter)3  -0.980      0.467   -2.10  0.058 .
## factor(starter)4   2.810      0.467    6.01 6.1e-05 ***
## factor(starter)5  -0.484      0.467   -1.04  0.321
## factor(batch)2    -1.348      0.467   -2.88  0.014 *
## factor(batch)3     0.276      0.467    0.59  0.566
## factor(batch)4     1.368      0.467    2.93  0.013 *
## factor(batch)5     0.200      0.467    0.43  0.676
## factor(position)2 -0.618      0.467   -1.32  0.211
## factor(position)3 -0.038      0.467   -0.08  0.937
## factor(position)4 -0.764      0.467   -1.63  0.128
## factor(position)5 -0.264      0.467   -0.56  0.583
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.739 on 12 degrees of freedom
## Multiple R-squared:  0.909, Adjusted R-squared:  0.818
## F-statistic: 9.96 on 12 and 12 DF, p-value: 0.000178
```

Starter 1 has intercept has p 1.55e-09 and starter 2 p of 0.7358 > 0.05 So no significant difference between starter 1 and 2

```
sample1 = data[data$starter == 1, ]
sample2 = data[data$starter == 2, ]
t.test(sample1$acidity, sample2$acidity, var.equal=TRUE)
```

```
##
## Two Sample t-test
##
## data: sample1$acidity and sample2$acidity
## t = 0.2, df = 8, p-value = 0.8
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -1.6 1.9
## sample estimates:
## mean of x mean of y
##      8.42      8.27
```

b) Removing insignificant block variables and performing ANOVA:

Previous test showed position had no impact, but starter 4, batches 2 and 4 did. Thus we remove position and do ANOVA

```
model <- aov(acidity ~ factor(starter) + factor(batch), data=data)
summary(model)
```

```
##                Df Sum Sq Mean Sq F value    Pr(>F)
## factor(starter)  4   44.1    11.03    19.84 4.8e-06 ***
## factor(batch)    4    18.8     4.69     8.44 0.00073 ***
## Residuals       16     8.9     0.56
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Indeed both starter and batch are important factors, but starter is more significant $p = 4.82e-06$
 $< p = 0.000735$

```
t.test(data[data$starter==1,]$acidity, data[data$starter==4,]$acidity, var.equal=TRUE)
```

```
##
## Two Sample t-test
##
## data:  data[data$starter == 1,]$acidity and data[data$starter == 4,]$acidity
## t = -4, df = 8, p-value = 0.004
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -4.43 -1.19
## sample estimates:
## mean of x mean of y
##      8.42      11.23
```

All T-Test of the samples of starter 4 with the other samples showed significant difference p value = 0.003928, 0.002609, 0.001324, 0.0007191 Thus, sample 4 makes significant difference.

c) Applying the Friedman test:

Applying Friedman to the model from b)

```
# The values, groups and blocks refers to acidity, starter and batch in the Friedman test
result_c = friedman.test(acidity ~ starter | batch, data=data)
print(result_c)
```

```
##
## Friedman rank sum test
##
## data:  acidity and starter and batch
## Friedman chi-squared = 13, df = 4, p-value = 0.01
```

Because $p = 0.01028 < 0.05$, that means that there is some effect of starter on acidity

d) Performing a mixed effects analysis:

Load required library

```
library("lme4")
```

```
## Loading required package: Matrix
```

```
lm(acidity ~ factor(starter) + factor(batch), data = data)
```

```
##
## Call:
## lm(formula = acidity ~ factor(starter) + factor(batch), data = data)
##
## Coefficients:
##      (Intercept)  factor(starter)2  factor(starter)3  factor(starter)4
##           8.325          -0.150          -0.980           2.810
## factor(starter)5  factor(batch)2    factor(batch)3    factor(batch)4
##          -0.484          -1.348           0.276           1.368
##   factor(batch)5
##           0.200
```

```
lmer(acidity ~ factor(starter) + factor(batch) + (1|position), data = data)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: acidity ~ factor(starter) + factor(batch) + (1 | position)
##   Data: data
## REML criterion at convergence: 48.9
## Random effects:
##   Groups   Name      Std.Dev.
##   position (Intercept) 0.0905
##   Residual              0.7389
## Number of obs: 25, groups:  position, 5
## Fixed Effects:
##      (Intercept)  factor(starter)2  factor(starter)3  factor(starter)4
##           8.325          -0.150          -0.980           2.810
## factor(starter)5  factor(batch)2    factor(batch)3    factor(batch)4
##          -0.484          -1.348           0.276           1.368
##   factor(batch)5
##           0.200
```

The code provides the same result for both scenarios with “starter” 4 and “batches” 2 and 4 being significant (away from 0). The code designates “starter 1” as the intercept, serving as the reference point for comparison with other values.