

# coding challenge 7

lihui xiang

2025-04-03

#####11. Read in the data called “PlantEmergence.csv” using a relative file path and load the following libraries. tidyverse, lme4, emmeans, multcomp, and multcompView. Turn the Treatment , DaysAfterPlanting and Rep into factors using the function as.factor

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats    1.0.0      v stringr   1.5.1
## v ggplot2    3.5.1      v tibble    3.2.1
## v lubridate  1.9.3      v tidyr     1.3.1
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(lme4)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
```

```
library(emmeans)
```

```
## Welcome to emmeans.
## Caution: You lose important information if you filter this package's results.
## See '? untidy'
```

```
library(multcomp)
```

```
## Loading required package: mvtnorm
## Loading required package: survival
## Loading required package: TH.data
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
##
##
## Attaching package: 'TH.data'
##
## The following object is masked from 'package:MASS':
##
##     geyser
```

```
library(multcompView)
STAND <- read.csv("PlantEmergence.csv")
##Turn the Treatment , DaysAfterPlanting and Rep into factors using the function as.factor
STAND$Treatment <- as.factor(STAND$Treatment)
STAND$DaysAfterPlanting <- as.factor(STAND$DaysAfterPlanting)
STAND$Rep <- as.factor(STAND$Rep)
```

###2. Fit a linear model to predict Emergence using Treatment and DaysAfterPlanting along with the interaction. Provide the summary of the linear model and ANOVA results

```
lm_model <- lm(Emergence ~ Treatment * DaysAfterPlanting, data = STAND)
summary(lm_model)
```

```
##
## Call:
## lm(formula = Emergence ~ Treatment * DaysAfterPlanting, data = STAND)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.250  -6.062  -0.875   6.750  21.875
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.823e+02  5.324e+00  34.229  <2e-16 ***
## Treatment2    -1.365e+02  7.530e+00 -18.128  <2e-16 ***
## Treatment3     1.112e+01  7.530e+00  1.477    0.142
## Treatment4     2.500e+00  7.530e+00  0.332    0.741
## Treatment5     8.750e+00  7.530e+00  1.162    0.248
## Treatment6     7.000e+00  7.530e+00  0.930    0.355
## Treatment7    -1.250e-01  7.530e+00 -0.017    0.987
## Treatment8     9.125e+00  7.530e+00  1.212    0.228
## Treatment9     2.375e+00  7.530e+00  0.315    0.753
## DaysAfterPlanting14  1.000e+01  7.530e+00  1.328    0.187
## DaysAfterPlanting21  1.062e+01  7.530e+00  1.411    0.161
## DaysAfterPlanting28  1.100e+01  7.530e+00  1.461    0.147
## Treatment2:DaysAfterPlanting14  1.625e+00  1.065e+01  0.153    0.879
## Treatment3:DaysAfterPlanting14 -2.625e+00  1.065e+01 -0.247    0.806
## Treatment4:DaysAfterPlanting14 -6.250e-01  1.065e+01 -0.059    0.953
## Treatment5:DaysAfterPlanting14  2.500e+00  1.065e+01  0.235    0.815
```

```
## Treatment6:DaysAfterPlanting14 1.000e+00 1.065e+01 0.094 0.925
## Treatment7:DaysAfterPlanting14 -2.500e+00 1.065e+01 -0.235 0.815
## Treatment8:DaysAfterPlanting14 -2.500e+00 1.065e+01 -0.235 0.815
## Treatment9:DaysAfterPlanting14 6.250e-01 1.065e+01 0.059 0.953
## Treatment2:DaysAfterPlanting21 3.500e+00 1.065e+01 0.329 0.743
## Treatment3:DaysAfterPlanting21 -1.000e+00 1.065e+01 -0.094 0.925
## Treatment4:DaysAfterPlanting21 1.500e+00 1.065e+01 0.141 0.888
## Treatment5:DaysAfterPlanting21 2.875e+00 1.065e+01 0.270 0.788
## Treatment6:DaysAfterPlanting21 4.125e+00 1.065e+01 0.387 0.699
## Treatment7:DaysAfterPlanting21 -2.125e+00 1.065e+01 -0.200 0.842
## Treatment8:DaysAfterPlanting21 -1.500e+00 1.065e+01 -0.141 0.888
## Treatment9:DaysAfterPlanting21 -1.250e+00 1.065e+01 -0.117 0.907
## Treatment2:DaysAfterPlanting28 2.750e+00 1.065e+01 0.258 0.797
## Treatment3:DaysAfterPlanting28 -1.875e+00 1.065e+01 -0.176 0.861
## Treatment4:DaysAfterPlanting28 3.123e-13 1.065e+01 0.000 1.000
## Treatment5:DaysAfterPlanting28 2.500e+00 1.065e+01 0.235 0.815
## Treatment6:DaysAfterPlanting28 2.125e+00 1.065e+01 0.200 0.842
## Treatment7:DaysAfterPlanting28 -3.625e+00 1.065e+01 -0.340 0.734
## Treatment8:DaysAfterPlanting28 -1.500e+00 1.065e+01 -0.141 0.888
## Treatment9:DaysAfterPlanting28 -8.750e-01 1.065e+01 -0.082 0.935
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.65 on 108 degrees of freedom
## Multiple R-squared: 0.9585, Adjusted R-squared: 0.945
## F-statistic: 71.21 on 35 and 108 DF, p-value: < 2.2e-16
```

```
anova(lm_model)
```

```
## Analysis of Variance Table
##
## Response: Emergence
##
##           Df Sum Sq Mean Sq F value    Pr(>F)
## Treatment      8 279366   34921 307.9516 < 2.2e-16 ***
## DaysAfterPlanting 3   3116    1039  9.1603 1.877e-05 ***
## Treatment:DaysAfterPlanting 24    142      6  0.0522      1
## Residuals    108  12247    113
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

###3. Based on the results of the linear model in question 2, do you need to fit the interaction term? Provide a simplified linear model without the interaction term but still testing both main effects. Provide the summary and ANOVA results. Then, interpret the intercept and the coefficient for Treatment 2.

```
###we do not need to fit the interaction term as the interaction term is not significant.
# Fit a simplified linear model without the interaction term
lm_simple <- lm(Emergence ~ Treatment + DaysAfterPlanting, data = STAND)

# Display the summary of the simplified linear model
summary(lm_simple)
```

```
##
## Call:
```

```
## lm(formula = Emergence ~ Treatment + DaysAfterPlanting, data = STAND)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.1632  -6.1536  -0.8542   6.1823  21.3958
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      182.163      2.797   65.136 < 2e-16 ***
## Treatment2      -134.531      3.425  -39.277 < 2e-16 ***
## Treatment3        9.750      3.425   2.847  0.00513 **
## Treatment4        2.719      3.425   0.794  0.42876
## Treatment5       10.719      3.425   3.129  0.00216 **
## Treatment6        8.812      3.425   2.573  0.01119 *
## Treatment7       -2.188      3.425  -0.639  0.52416
## Treatment8        7.750      3.425   2.263  0.02529 *
## Treatment9        2.000      3.425   0.584  0.56028
## DaysAfterPlanting14  9.722      2.283   4.258 3.89e-05 ***
## DaysAfterPlanting21 11.306      2.283   4.951 2.21e-06 ***
## DaysAfterPlanting28 10.944      2.283   4.793 4.36e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.688 on 132 degrees of freedom
## Multiple R-squared:  0.958, Adjusted R-squared:  0.9545
## F-statistic: 273.6 on 11 and 132 DF, p-value: < 2.2e-16
```

```
# Perform ANOVA to test the significance of main effects
anova(lm_simple)
```

```
## Analysis of Variance Table
##
## Response: Emergence
##              Df Sum Sq Mean Sq F value    Pr(>F)
## Treatment      8 279366   34921 372.070 < 2.2e-16 ***
## DaysAfterPlanting 3   3116    1039  11.068 1.575e-06 ***
## Residuals     132  12389     94
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#The intercept (182.25) represents the expected emergence when the treatment is at level 1 (reference level)
```

####4. Calculate the least square means for Treatment using the emmeans package and perform a Tukey separation with the compact letter display using the cld function. Interpret the results.

```
lsmeans_treat <- emmeans(lm_simple, ~ Treatment)
results_cld <- cld(lsmeans_treat, alpha = 0.05, Letters = letters)
print(results_cld)
```

```
## Treatment emmean SE df lower.CL upper.CL .group
## 2          55.6 2.42 132    50.8    60.4 a
## 7          188.0 2.42 132   183.2   192.8 b
```

```
## 1      190.2 2.42 132      185.4      194.9      bc
## 9      192.2 2.42 132      187.4      196.9      bc
## 4      192.9 2.42 132      188.1      197.7      bc
## 8      197.9 2.42 132      193.1      202.7      bc
## 6      199.0 2.42 132      194.2      203.8      c
## 3      199.9 2.42 132      195.1      204.7      c
## 5      200.9 2.42 132      196.1      205.7      c
##
## Results are averaged over the levels of: DaysAfterPlanting
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 9 estimates
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping symbol,
##       then we cannot show them to be different.
##       But we also did not show them to be the same.
```

##5. The provided function lets you dynamically add a linear model plus one factor from that model and plots a bar chart with letters denoting treatment differences. Use this model to generate the plot shown below. Explain the significance of the letters.

```
plot_cldbars_onefactor <- function(lm_model, factor) {
  data <- lm_model$model
  variables <- colnames(lm_model$model)
  dependent_var <- variables[1]
  independent_var <- variables[2:length(variables)]

  lsmeans <- emmeans(lm_model, as.formula(paste("~", factor))) # estimate lsmeans
  Results_lsmeans <- cld(lsmeans, alpha = 0.05, reversed = TRUE, details = TRUE, Letters = letters) # c

  # Extracting the letters for the bars
  sig.diff.letters <- data.frame(Results_lsmeans$emmeans[,1],
                                str_trim(Results_lsmeans$emmeans[,7]))
  colnames(sig.diff.letters) <- c(factor, "Letters")

  # for plotting with letters from significance test
  ave_stand2 <- lm_model$model %>%
    group_by(!sym(factor)) %>%
    dplyr::summarize(
      ave.emerge = mean(.data[[dependent_var]], na.rm = TRUE),
      se = sd(.data[[dependent_var]]) / sqrt(n())
    ) %>%
    left_join(sig.diff.letters, by = factor) %>%
    mutate(letter_position = ave.emerge + 10 * se)

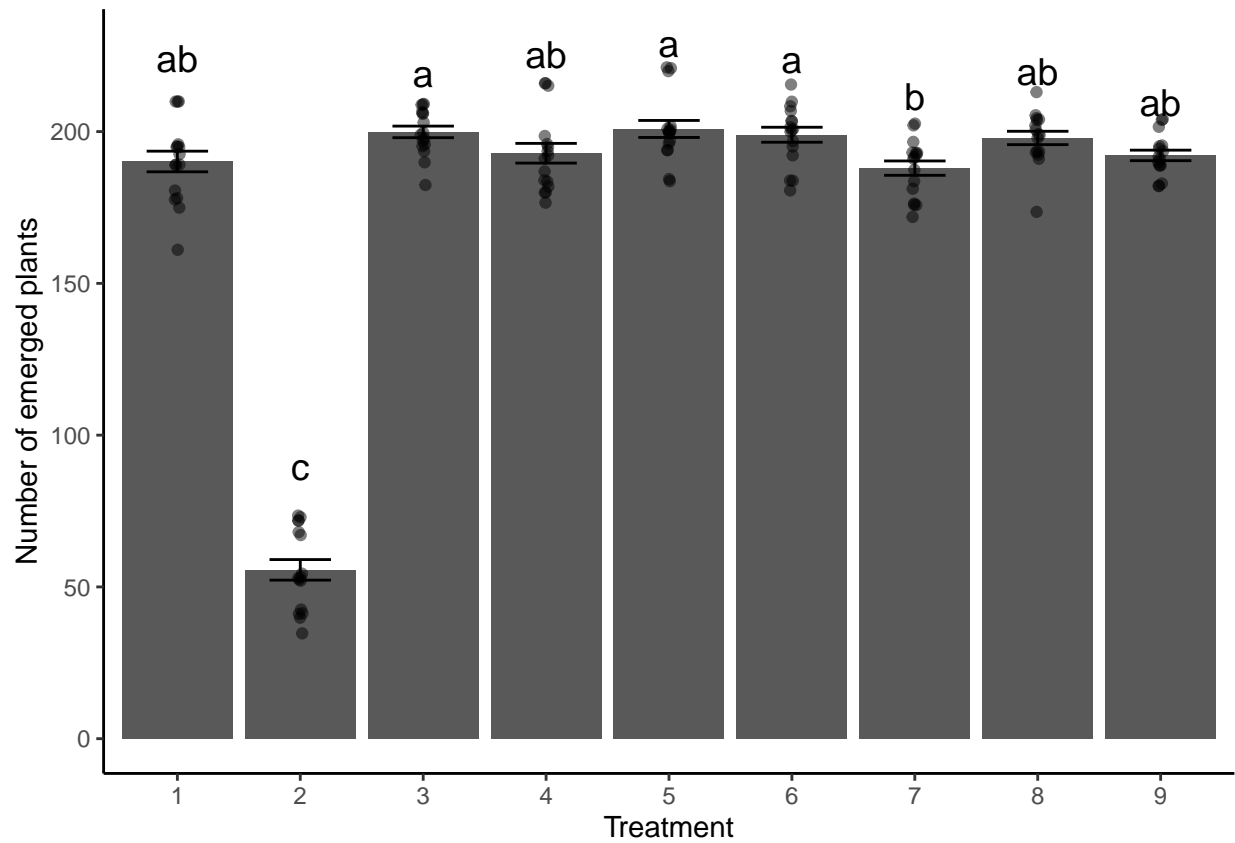
  plot <- ggplot(data, aes(x = !! sym(factor), y = !! sym(dependent_var))) +
    stat_summary(fun = mean, geom = "bar") +
    stat_summary(fun.data = mean_se, geom = "errorbar", width = 0.5) +
    ylab("Number of emerged plants") +
    geom_jitter(width = 0.02, alpha = 0.5) +
    geom_text(data = ave_stand2, aes(label = Letters, y = letter_position), size = 5) +
    xlab(as.character(factor)) +
    theme_classic()

  return(plot)
```

```

}
plot1 <- plot_cldbars_onefactor(lm_simple, "Treatment")
print(plot1)

```



*##This plot shows the average number of emerged plants for each treatment, with error bars showing how*

###6. You can view the GitHub folder for this challenge here:  
 GitHub - coding\_challenge7