DS5110 Homework 5

Parth Shah

2023-04-02

Part A

Problem 1

Flash Paper - Akshi Saxena

Data source - https://www.kaggle.com/datasets/rtatman/chocolate-bar-ratings?select=flavors_of_cacao.csv

```
suppressPackageStartupMessages(library(tidyverse))
library(readr)
library(dplyr)
library(ggplot2)

path <- file.path(getwd(), "flavors_of_cacao.csv")
cocoa <- read_csv(file=path, show_col_types = FALSE) %>%
  setNames(c("company", "origin_bar", "ref", "review_year", "cocoa_perc",
             "company_location", "rating", "bean_type", "bean_orig")) %>%
  mutate(cocoa_perc = as.numeric(gsub("%", "", cocoa_perc)))
head(cocoa)
```

```
## # A tibble: 6 x 9
##   company  origin_bar    ref review_year cocoa_perc company_location rating
##   <chr>    <chr>       <dbl>       <dbl>      <dbl> <chr>              <dbl>
## 1 A. Morin Agua Grande  1876        2016         63 France              3.75
## 2 A. Morin Kpime        1676        2015         70 France              2.75
## 3 A. Morin Atsane       1676        2015         70 France              3
## 4 A. Morin Akata        1680        2015         70 France              3.5
## 5 A. Morin Quilla       1704        2015         70 France              3.5
## 6 A. Morin Carenero     1315        2014         70 France              2.75
## # i 2 more variables: bean_type <chr>, bean_orig <chr>
```

Preprocessing:

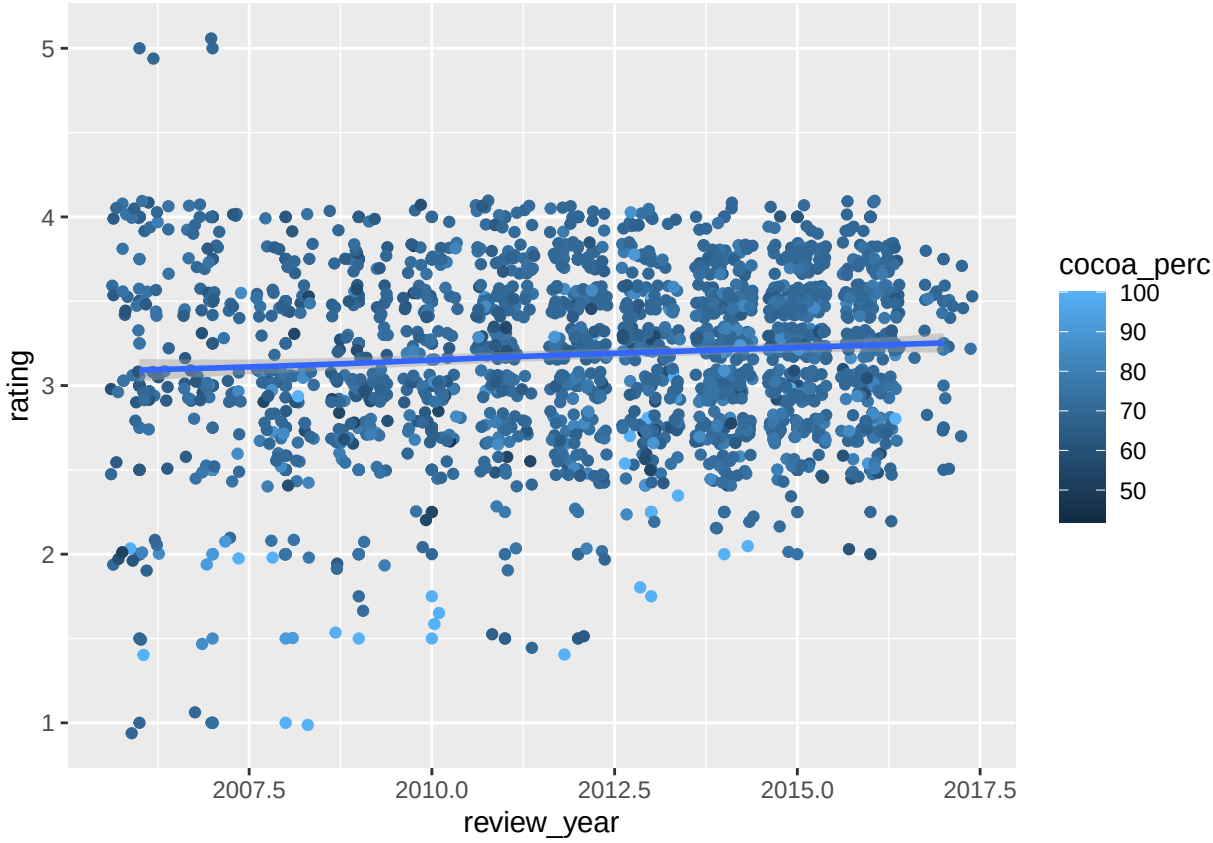
-The columns were renamed to eliminate white spaces and create shorter, more concise names.

-The cocoa_percent column initially contained strings with a "%" symbol. The data type was converted to numeric, and the "%" symbol was removed for easier analysis.

Problem 2

```
ggplot(cocoa, aes(x= review_year, y = rating, color = cocoa_perc)) +
    geom_point() +
    geom_jitter() +
    geom_smooth()
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: The following aesthetics were dropped during statistical transformation: colour
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
```



Part B

Problem 3

```
suppressPackageStartupMessages(library(dplyr))
suppressPackageStartupMessages(library(tidyverse))
```

```
library(modelr)
library(ggplot2)

load("37938-0001-Data.rda")
my_data <- da37938.0001

my_data <- my_data %>% select(LIFESAT_I, SOCIALWB_I, NONAFFIRM_I,
                              NONDISCLOSURE_I, HCTHREAT_I, KESSLER6_I,
                              EVERYDAY_I)

my_data <- na.omit(my_data)

set.seed(2)
my_data_part <- resample_partition(my_data, p=c(train=0.5, test=0.5))
trainSet <- my_data[my_data_part$train$idx,]
testSet <- my_data[my_data_part$test$idx,]
```

```
predictors <- colnames(trainSet)[-1]
rmse_results <- c()
for (predictor in predictors) {
  formula <- as.formula(paste("LIFESAT_I ~", predictor))
  model <- lm(formula, data = trainSet)
  predictions <- predict(model, newdata = testSet)
  residuals <- testSet$LIFESAT_I - predictions
  rmse_value <- sqrt(mean(residuals^2))
  rmse_results <- append(rmse_results, rmse_value)
}
rmse_df <- data.frame(Predictor = predictors, RMSE = rmse_results)
rmse_df
```

```
##         Predictor     RMSE
## 1      SOCIALWB_I 1.557210
## 2     NONAFFIRM_I 1.609596
## 3 NONDISCLOSURE_I 1.667608
## 4      HCTHREAT_I 1.603713
## 5      KESSLER6_I 1.339974
## 6      EVERYDAY_I 1.518621
```

The model that uses the Mental Distress/Disorder (KESSLER6_I) predictor has the lowest RMSE value. Therefore, among these scales, it is the most effective single predictor for life satisfaction, as determined by the model's performance on the test set.

```
single_predictor <- rmse_df$RMSE[rmse_df$Predictor == "KESSLER6_I"]
```

Problem 4

Step 1:

```
model <- lm(LIFESAT_I ~ KESSLER6_I, data=trainSet)
```

```
steps <- function(response, predictors, candidates, train, test)
{
  rhs <- paste0(paste0(predictors, collapse="+"), "+", candidates)
  formulas <- lapply(paste0(response, "~", rhs), as.formula)
  rmses <- sapply(formulas,
                  function(fm) rmse(lm(fm, data=train),
                                    data=test))
  names(rmses) <- candidates
  attr(rmses, "best") <- rmses[which.min(rmses)]
  rmses
}
```

```
preds <- "KESSLER6_I"
cands <- c("SOCIALWB_I", "NONAFFIRM_I", "NONDISCLOSURE_I", "HCTHREAT_I",
           "EVERYDAY_I")
s1 <- steps("LIFESAT_I", preds, cands, trainSet, testSet)
model <- c(model, attr(s1, "best"))
s1
```

```
##      SOCIALWB_I     NONAFFIRM_I NONDISCLOSURE_I      HCTHREAT_I      EVERYDAY_I
##        1.298163        1.339775        1.340157        1.334188        1.320025
## attr(,"best")
## SOCIALWB_I
##   1.298163
```

Initially, I begin with the model utilizing the Mental Distress/Disorder (KESSLER6_I) predictor, as established in Problem 3. Following that, I discover that incorporating Social Well-being (SOCIALWB_I) as a second predictor yields the lowest RMSE value. As a result, I integrate Social Well-being into the model.

```
fit1 <- lm(LIFESAT_I ~ KESSLER6_I + SOCIALWB_I, data=trainSet)
double_predictor <- rmse(fit1, testSet)
```

Step 2:

```
preds <- c("KESSLER6_I", "SOCIALWB_I")
cands <- c("NONAFFIRM_I", "NONDISCLOSURE_I", "HCTHREAT_I", "EVERYDAY_I")
s1 <- steps("LIFESAT_I", preds, cands, trainSet, testSet)
model <- c(model, attr(s1, "best"))
s1
```

```
##     NONAFFIRM_I NONDISCLOSURE_I      HCTHREAT_I      EVERYDAY_I
##        1.306177        1.291490        1.294142        1.280575
## attr(,"best")
## EVERYDAY_I
##   1.280575
```

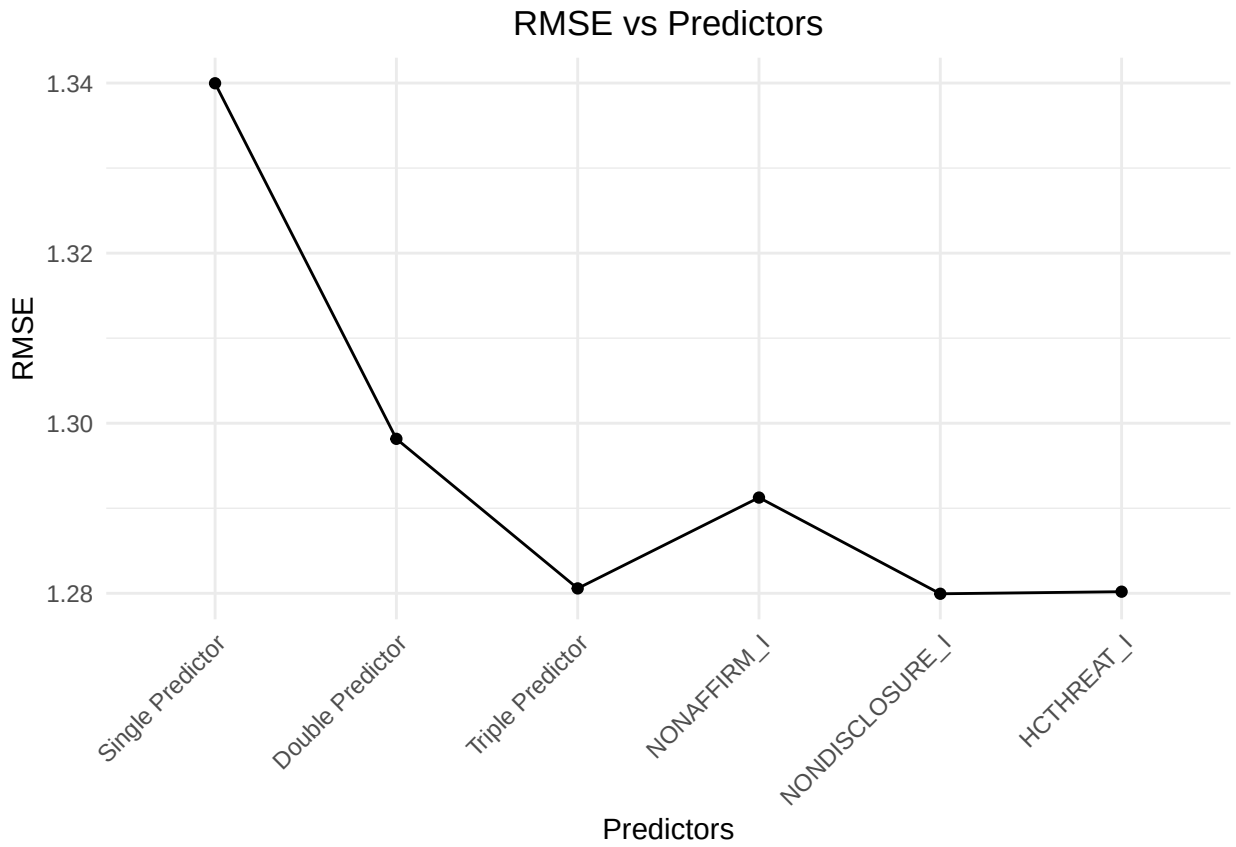
After completing step 1, I discovered that including Everyday Discrimination (EVERYDAY_I) as the third predictor led to the lowest RMSE value. As a result, I decided to incorporate Everyday Discrimination into the model as the third predictor.

```
fit2 <- lm(LIFESAT_I ~ KESSLER6_I + SOCIALWB_I + EVERYDAY_I, data=trainSet)
triple_predictor <- rmse(fit2, testSet)
```

In conclusion, I determined that Mental Distress/Disorder (KESSLER6_I), Social Well-being (SOCIALWB_I), and Everyday Discrimination (EVERYDAY_I) are the top three predictors (among these scales) for life satisfaction among trans people.

Problem 5

```
library(ggplot2)
library(lattice)

s1_df <- data.frame(Variable = character(), RMSE = numeric())
s1_df <- rbind(s1_df, data.frame(Variable = "Single Predictor", RMSE = single_predictor))
s1_df <- rbind(s1_df, data.frame(Variable = "Double Predictor", RMSE = double_predictor))
s1_df <- rbind(s1_df, data.frame(Variable = "Triple Predictor", RMSE = triple_predictor))

preds <- c("KESSLER6_I", "SOCIALWB_I", "EVERYDAY_I")

cands <- c("NONAFFIRM_I", "NONDISCLOSURE_I", "HCTHREAT_I")

steps <- function(response, predictors, candidates, train, test)
{
  rhs <- paste0(paste0(predictors, collapse="+"), "+", candidates)
  formulas <- lapply(paste0(response, "~", rhs), as.formula)
  rmses <- sapply(formulas,
                  function(fm) rmse(lm(fm, data=train),
                                    data=test))
  names(rmses) <- candidates
  attr(rmses, "best") <- rmses[which.min(rmses)]
  rmses
}

s1 <- steps("LIFESAT_I", preds, cands, trainSet, testSet)
s1_df <- rbind(s1_df, data.frame(Variable = names(s1), RMSE = as.numeric(s1)))

s1_df$Variable <- factor(s1_df$Variable, levels = s1_df$Variable)

ggplot(s1_df, aes(x = Variable, y = RMSE)) +
  geom_point() +
  geom_line(group = 1) +
  ggtitle("RMSE vs Predictors") +
  xlab("Predictors") +
  ylab("RMSE") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        axis.text.x = element_text(angle = 45, hjust = 1))
```



The plot shows a minimal decrease in RMSE after adding HCTHREAT_I or NONDISCLOSURE_I, suggesting that a model with more than three predictors may not be reasonable for predicting life satisfaction using these scales. Adding more predictors can increase model complexity without significantly improving performance, making it harder to interpret and increasing the risk of overfitting. Therefore, it's generally preferable to opt for a simpler model with fewer predictors, for better interpretability and to avoid overfitting.