Support Vector Regression for Coffee Quality Prediction

Luisa Kalkert

2025-12-18

```r
# Read the data
coffee_data <- read.csv("data/arabica_data_cleaned.csv")

# Remove unnecessary columns
coffee_data$Altitude <- NULL # redundant information with altitude_high_meters and altitude_low_meters
coffee_data$Species <- NULL # All coffees belong to the Arabica genus
coffee_data$Total.Cup.Points <- NULL # Linearly dependent from other columns

# Summary of data
summary(coffee_data)
```

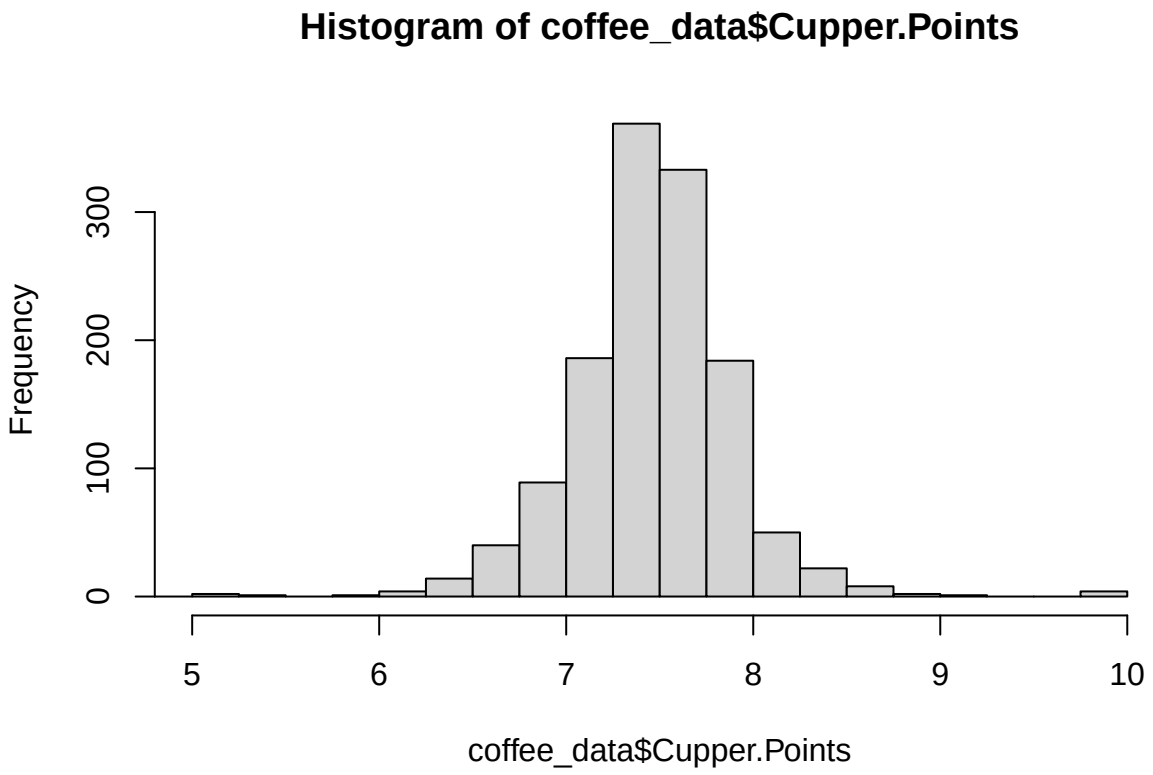
```
##        X             Owner           Country.of.Origin   Farm.Name
##  Min.   :   1.0   Length:1311        Length:1311        Length:1311
##  1st Qu.: 328.5   Class :character   Class :character   Class :character
##  Median : 656.0   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 656.0
##  3rd Qu.: 983.5
##  Max.   :1312.0
##
##   Lot.Number            Mill            ICO.Number          Company
##  Length:1311        Length:1311        Length:1311        Length:1311
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##     Region            Producer         Number.of.Bags    Bag.Weight
##  Length:1311        Length:1311        Min.   :   0.0   Length:1311
##  Class :character   Class :character   1st Qu.:  14.5   Class :character
##  Mode  :character   Mode  :character   Median : 175.0   Mode  :character
##                                        Mean   : 153.9
##                                        3rd Qu.: 275.0
##                                        Max.   :1062.0
##
##  In.Country.Partner Harvest.Year       Grading.Date         Owner.1
##  Length:1311        Length:1311        Length:1311        Length:1311
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
```

```
##
##    Variety          Processing.Method      Aroma            Flavor
##  Length:1311        Length:1311        Min.   :0.000   Min.   :0.000
##  Class :character   Class :character   1st Qu.:7.420   1st Qu.:7.330
##  Mode  :character   Mode  :character   Median :7.580   Median :7.580
##                                        Mean   :7.564   Mean   :7.518
##                                        3rd Qu.:7.750   3rd Qu.:7.750
##                                        Max.   :8.750   Max.   :8.830
##
##    Aftertaste       Acidity           Body          Balance
##  Min.   :0.000   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:7.250   1st Qu.:7.330   1st Qu.:7.330   1st Qu.:7.330
##  Median :7.420   Median :7.500   Median :7.500   Median :7.500
##  Mean   :7.398   Mean   :7.533   Mean   :7.518   Mean   :7.518
##  3rd Qu.:7.580   3rd Qu.:7.750   3rd Qu.:7.670   3rd Qu.:7.750
##  Max.   :8.670   Max.   :8.750   Max.   :8.580   Max.   :8.750
##
##    Uniformity       Clean.Cup        Sweetness      Cupper.Points
##  Min.   : 0.000   Min.   : 0.000   Min.   : 0.000   Min.   : 0.000
##  1st Qu.:10.000   1st Qu.:10.000   1st Qu.:10.000   1st Qu.: 7.250
##  Median :10.000   Median :10.000   Median :10.000   Median : 7.500
##  Mean   : 9.833   Mean   : 9.833   Mean   : 9.903   Mean   : 7.498
##  3rd Qu.:10.000   3rd Qu.:10.000   3rd Qu.:10.000   3rd Qu.: 7.750
##  Max.   :10.000   Max.   :10.000   Max.   :10.000   Max.   :10.000
##
##     Moisture       Category.One.Defects    Quakers           Color
##  Min.   :0.00000   Min.   : 0.0000      Min.   : 0.0000   Length:1311
##  1st Qu.:0.09000   1st Qu.: 0.0000      1st Qu.: 0.0000   Class :character
##  Median :0.11000   Median : 0.0000      Median : 0.0000   Mode  :character
##  Mean   :0.08886   Mean   : 0.4264      Mean   : 0.1771
##  3rd Qu.:0.12000   3rd Qu.: 0.0000      3rd Qu.: 0.0000
##  Max.   :0.28000   Max.   :31.0000      Max.   :11.0000
##                                         NA's   :1
##  Category.Two.Defects  Expiration        Certification.Body
##  Min.   : 0.000       Length:1311        Length:1311
##  1st Qu.: 0.000       Class :character   Class :character
##  Median : 2.000       Mode  :character   Mode  :character
##  Mean   : 3.592
##  3rd Qu.: 4.000
##  Max.   :55.000
##
##  Certification.Address Certification.Contact unit_of_measurement
##  Length:1311           Length:1311           Length:1311
##  Class :character      Class :character      Class :character
##  Mode  :character      Mode  :character      Mode  :character
##
##
##
##
##  altitude_low_meters altitude_high_meters altitude_mean_meters
##  Min.   :     1      Min.   :     1       Min.   :     1
##  1st Qu.:  1100      1st Qu.:  1100       1st Qu.:  1100
##  Median :  1311      Median :  1350       Median :  1311
##  Mean   :  1760      Mean   :  1809       Mean   :  1784
```

```
##  3rd Qu.:  1600      3rd Qu.:  1650       3rd Qu.:  1600
##  Max.   :190164      Max.   :190164       Max.   :190164
##  NA's   :227         NA's   :227          NA's   :227
```

```r
#head(coffee_data)

# Target variable: Cupper.points -> How much did the taster like this coffee?
p <- hist(coffee_data$Cupper.Points, xlim = c(5,10), breaks = seq(0,10,0.25))
```




```r
#print(p)


# For categorical columns, get the number of unique values
for (col in names(coffee_data)) {
    print(paste(col, "has", length(unique(coffee_data[[col]])), "unique values"))
}
```

```
## [1] "X has 1311 unique values"
## [1] "Owner has 306 unique values"
## [1] "Country.of.Origin has 37 unique values"
## [1] "Farm.Name has 558 unique values"
## [1] "Lot.Number has 222 unique values"
## [1] "Mill has 449 unique values"
## [1] "ICO.Number has 843 unique values"
## [1] "Company has 271 unique values"
## [1] "Region has 344 unique values"
## [1] "Producer has 677 unique values"
## [1] "Number.of.Bags has 130 unique values"
```

```
## [1] "Bag.Weight has 56 unique values"
## [1] "In.Country.Partner has 27 unique values"
## [1] "Harvest.Year has 47 unique values"
## [1] "Grading.Date has 558 unique values"
## [1] "Owner.1 has 310 unique values"
## [1] "Variety has 30 unique values"
## [1] "Processing.Method has 6 unique values"
## [1] "Aroma has 33 unique values"
## [1] "Flavor has 35 unique values"
## [1] "Aftertaste has 35 unique values"
## [1] "Acidity has 31 unique values"
## [1] "Body has 31 unique values"
## [1] "Balance has 32 unique values"
## [1] "Uniformity has 10 unique values"
## [1] "Clean.Cup has 11 unique values"
## [1] "Sweetness has 8 unique values"
## [1] "Cupper.Points has 42 unique values"
## [1] "Moisture has 23 unique values"
## [1] "Category.One.Defects has 16 unique values"
## [1] "Quakers has 12 unique values"
## [1] "Color has 5 unique values"
## [1] "Category.Two.Defects has 38 unique values"
## [1] "Expiration has 557 unique values"
## [1] "Certification.Body has 26 unique values"
## [1] "Certification.Address has 30 unique values"
## [1] "Certification.Contact has 27 unique values"
## [1] "unit_of_measurement has 2 unique values"
## [1] "altitude_low_meters has 189 unique values"
## [1] "altitude_high_meters has 189 unique values"
## [1] "altitude_mean_meters has 202 unique values"
```

```r
# Data Preprocessing
cloumns_to_use <- c(
    "Number.of.Bags",
    #"Year", # Badly formatted, remove
    "Bag.Weight", # KG and pound mixed, so we need to convert to KG
    "Variety", # Many rare varieties, Remove or  consider the top 4 and name everything else "Other"
    "Processing.Method", # 6 different varieties
    "Aroma",
    "Flavor",
    "Aftertaste",
    "Acidity",
    "Body",
    "Balance",
    "Uniformity",
    "Clean.Cup",
    "Sweetness",
    "Cupper.Points",
    "Moisture",
    "Category.One.Defects",
    "Quakers",
    #"Color", # 216 missinng values, remove?
    "Category.Two.Defects",
    #"Expiration", # Only relevant with year, but year is badly formatted - remove
```

```
    "altitude_mean_meters" # Impute missing values <- maybe from region?
)



# Pool varieties into top 4 and everything else as "Other"
top_4_varieties <- coffee_data$Variety[order(coffee_data$Variety, decreasing = TRUE)][1:4]

coffee_data$Variety <- ifelse(coffee_data$Variety %in% top_4_varieties, coffee_data$Variety, "Other")
coffee_data$Variety <- factor(coffee_data$Variety)

# Impute missing values for processing method
coffee_data$Processing.Method <- ifelse(coffee_data$Processing.Method == "", "Unknown", coffee_data$Pro
coffee_data$Processing.Method <- factor(coffee_data$Processing.Method)

# Get all missing values
missing_values <- colSums(is.na(coffee_data))
print(missing_values)
```

```
##                      X                  Owner     Country.of.Origin
##                      0                      0                      0
##             Farm.Name            Lot.Number                   Mill
##                      0                      0                      0
##            ICO.Number                Company                 Region
##                      0                      0                      0
##              Producer         Number.of.Bags            Bag.Weight
##                      0                      0                      0
##    In.Country.Partner           Harvest.Year          Grading.Date
##                      0                      0                      0
##               Owner.1                Variety     Processing.Method
##                      0                      0                      0
##                 Aroma                 Flavor            Aftertaste
##                      0                      0                      0
##               Acidity                   Body               Balance
##                      0                      0                      0
##            Uniformity              Clean.Cup              Sweetness
##                      0                      0                      0
##         Cupper.Points              Moisture  Category.One.Defects
##                      0                      0                      0
##               Quakers                  Color  Category.Two.Defects
##                      1                      0                      0
##            Expiration    Certification.Body Certification.Address
##                      0                      0                      0
## Certification.Contact   unit_of_measurement   altitude_low_meters
##                      0                      0                    227
##  altitude_high_meters  altitude_mean_meters
##                   227                    227
```

```
# Many missing values for altitude_mean_meters -> impute from region? Continue here

# If altitude_mean_meters is missing, use the mean altitude of all coffees from the same region to impu
missing_altitude_mean_meters_indices <- which(is.na(coffee_data$altitude_mean_meters))
```

```r
for (i in missing_altitude_mean_meters_indices) {
    region_i <- coffee_data$Region[i]
    region_mean <- mean(
        coffee_data$altitude_mean_meters[coffee_data$Region == region_i],
        na.rm = TRUE
    )
    coffee_data$altitude_mean_meters[i] <- region_mean
}




 # Use this to impute the missing values


# If still missing, mean of country?
```