Categorical Predictors

In this tutorial we will learn how to carry out multiple linear regression with categorical predictors in R. We will use the data set `wine.txt`. The data set contains ratings of various wines produced in California. For this tutorial, we will focus on the response variable y =`Quality` (average quality rating), x_1 =`Flavor` (average flavor rating), and `Region` indicating which of three regions in California the wine is produced in. The regions are coded 1 = North, 2 = Central, and 3 = Napa.

Read the data in and also load the `tidyverse` package

```
library(tidyverse)
```

```
Data<-read.table("wine.txt", header=TRUE, sep="")
head(Data)
```

```
##   Clarity Aroma Body Flavor Oakiness Quality Region
## 1       1   3.3  2.8    3.1      4.1     9.8      1
## 2       1   4.4  4.9    3.5      3.9    12.6      1
## 3       1   3.9  5.3    4.8      4.7    11.9      1
## 4       1   3.9  2.6    3.1      3.6    11.1      1
## 5       1   5.6  5.1    5.5      5.1    13.3      1
## 6       1   4.6  4.7    5.0      4.1    12.8      1
```

Notice that the variable `Region` is coded numerically, even though it is a categorical variable. We need to make sure that R is viewing its type correctly

```
class(Data$Region)
```

```
## [1] "integer"
```

We need to convert `Region` to be viewed as categorical by using `factor()`

```
Data$Region<-factor(Data$Region)
class(Data$Region)
```

```
## [1] "factor"
```

```
levels(Data$Region)
```

```
## [1] "1" "2" "3"
```

Notice the names of the levels are not descriptive. We should also give more descriptive names to the levels of `Region`

```
levels(Data$Region) <- c("North", "Central", "Napa")
levels(Data$Region)
```

```
## [1] "North"   "Central" "Napa"
```

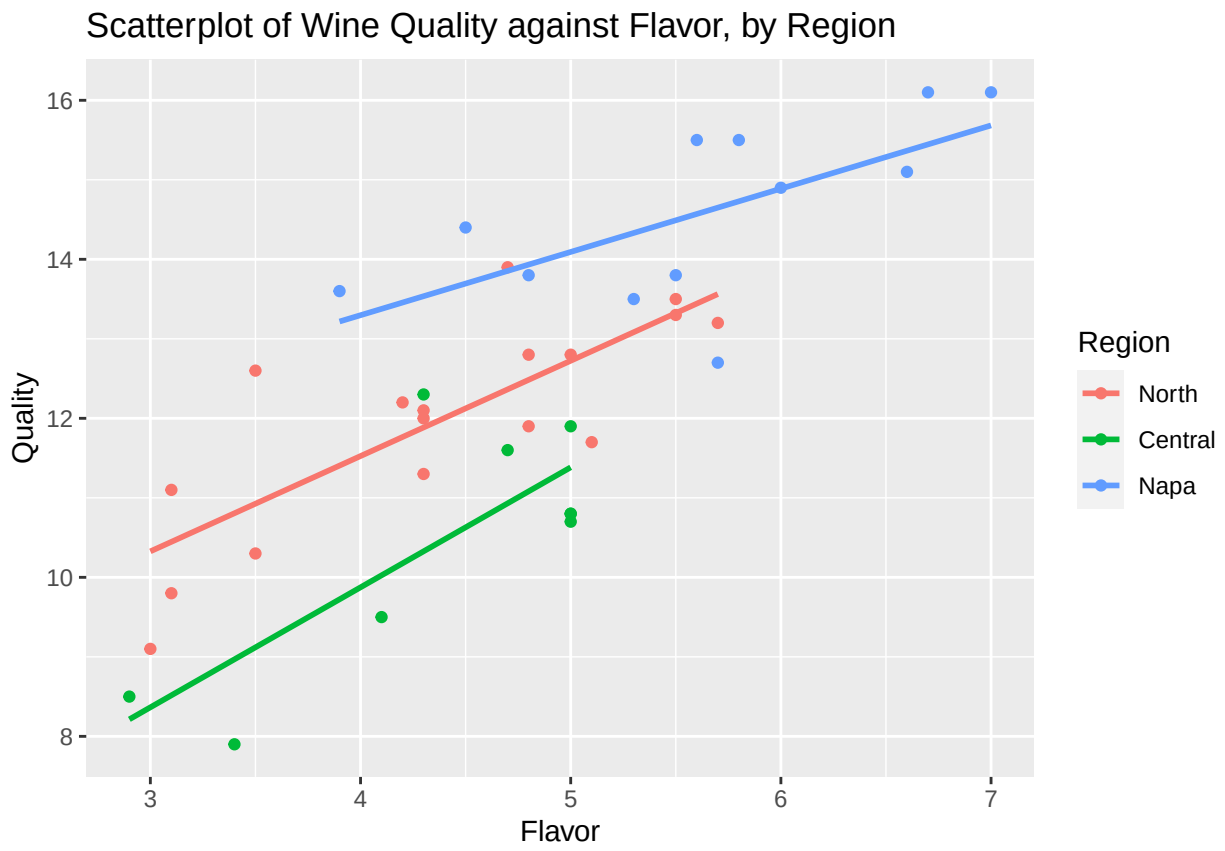
We have done the needed data wrangling for this tutorial.

1. Scatterplot with categorical predictor

Since we have a quantitative response variable, `Quality`, a quantitative predictor `Flavor` and a categorical predictor `Region`, we can create a scatterplot, with `Quality` on the y-axis, `Flavor` on the x-axis, and use different colored plots to denote the different regions

```
ggplot(Data, aes(x=Flavor, y=Quality, color=Region))+
  geom_point()+
  geom_smooth(method=lm, se=FALSE)+
  labs(title="Scatterplot of Wine Quality against Flavor, by Region")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



We notice a positive linear association between `Quality` and `Flavor` across all three regions, the better the flavor of the wine, the higher the quality rating of the wine.

The slopes are not exactly parallel, indicating that perhaps there is an interaction between the region of the wine and its flavor; the impact of flavor on quality rating differs among the regions.

2. Fitting MLR with interaction

Since the categorical variable `Region` has three levels, we know that there will be two indicator variables created to represent the various regions. To check the dummy coding

```
contrasts(Data$Region)
```

```
##         Central Napa
## North         0    0
## Central       1    0
## Napa          0    1
```

This output informs use the `North` region is the reference class, as it is coded 0 for both indicator variables. To change the reference class to the `Napa` region

```
Data$Region<-relevel(Data$Region, ref = "Napa")
contrasts(Data$Region)
```

```
##         North Central
## Napa        0       0
## North       1       0
## Central     0       1
```

Based on the possibility of non-parallel slopes in the scatterplot, we consider fitting a regression model with an interaction term between the predictors

```
result<-lm(Quality~Flavor*Region, data=Data)
summary(result)
```

```
##
## Call:
## lm(formula = Quality ~ Flavor * Region, data = Data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.94964 -0.58463  0.04393  0.49607  1.97295
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           10.1144     1.6692   6.060 9.14e-07 ***
## Flavor                 0.7957     0.2936   2.710   0.0107 *
## RegionNorth           -3.3833     2.0153  -1.679   0.1029
## RegionCentral         -6.2775     2.4491  -2.563   0.0153 *
## Flavor:RegionNorth     0.4029     0.3878   1.039   0.3066
## Flavor:RegionCentral   0.7137     0.4992   1.430   0.1625
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8914 on 32 degrees of freedom
## Multiple R-squared:  0.8357, Adjusted R-squared:  0.8101
## F-statistic: 32.56 on 5 and 32 DF,  p-value: 1.179e-11
```

Given that the t tests for the interaction terms are insignificant, we conduct a partial F test to see if the interaction terms can all be dropped

```
##fit regression with no interaction
reduced<-lm(Quality~Flavor+Region, data=Data)
```

```r
##Partial F test for interaction terms
anova(reduced,result)
```

```
## Analysis of Variance Table
##
## Model 1: Quality ~ Flavor + Region
## Model 2: Quality ~ Flavor * Region
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     34 27.213
## 2     32 25.429  2    1.7845 1.1229 0.3378
```

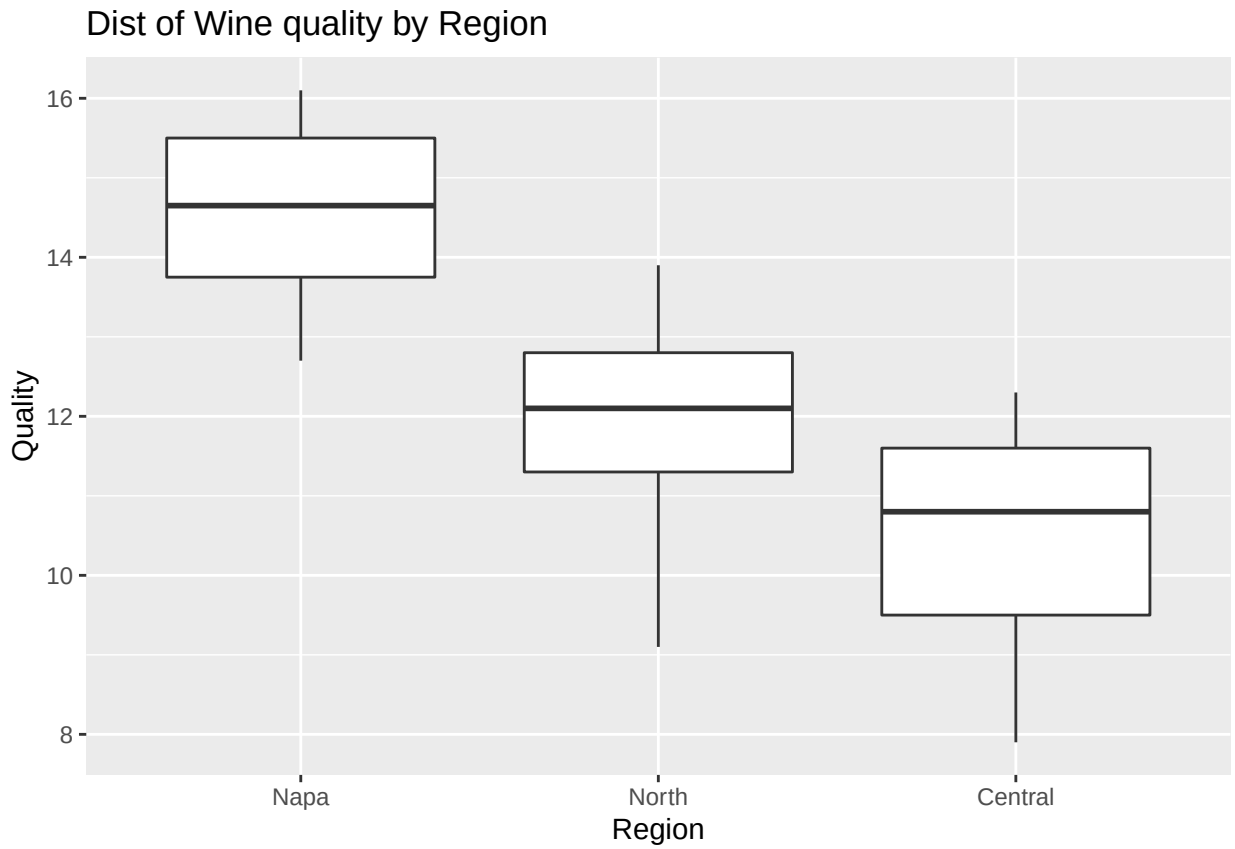
The insignificant result of the partial F test means we can drop the interaction terms. There is little evidence that the slopes are truly different.

3. Levene's test for equality of variances across levels

The regression assumptions when a categorical predictor is involved are pretty much the same, assessed similarly as before using a residual plot, ACF plot of the residuals, and QQ plot of the residuals.

There is an additional assumption to check: that the variance of the response variable is constant across all levels of the categorical predictor. We can use a box plot to visualize this

```r
ggplot(Data, aes(x=Region, y=Quality))+
  geom_boxplot()+
  labs(title="Dist of Wine quality by Region")
```



The spread of `Quality` appears to be similar across the regions.

We can use Levene's test to test for the equality of variances across all levels. We use the function `levene.test()` from the `lawstat` package

```
library(lawstat)
```

```
levene.test(Data$Quality,Data$Region)
```

```
##
##  Modified robust Brown-Forsythe Levene-type test based on the absolute
##  deviations from the median
##
## data:  Data$Quality
## Test Statistic = 0.1052, p-value = 0.9004
```

The null hypothesis for Levene's test is that the variances are equal across all classes of the categorical predictor. Since the p-value is high, we do not have evidence that this assumption is not met.

4. Multiple comparisons

Since we have a model with no interactions, we can interpret the coefficients of the indicator variables as the difference in the mean quality rating, given the flavor rating, between the class in question and the reference class.

```
summary(reduced)
```

```
##
## Call:
## lm(formula = Quality ~ Flavor + Region, data = Data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.97630 -0.58844  0.02184  0.51572  1.94232
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     8.3177     1.0100   8.235 1.31e-09 ***
## Flavor          1.1155     0.1738   6.417 2.49e-07 ***
## RegionNorth    -1.2234     0.4003  -3.056  0.00435 **
## RegionCentral  -2.7569     0.4495  -6.134 5.78e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8946 on 34 degrees of freedom
## Multiple R-squared:  0.8242, Adjusted R-squared:  0.8087
## F-statistic: 53.13 on 3 and 34 DF,  p-value: 6.358e-13
```

For example, the difference in the mean quality rating between wines in the `North` and `Napa` regions is -1.22, for given flavor ratings.

Since there are 3 classes, there will be 3 possible pairs of regions to compare:

- `North` and `Napa`
- `Central` and `Napa`
- `North` and `Central`

The first two comparisons are easy as we can just refer to the coefficients for the indicator variables. A little bit of work needs to be done to compare the `North` and `Central` regions. Also, note we are performing three hypothesis tests. So we need multiple comparison methods to control the family-wide error rate.

One such method is Tukey's method. We use the `glht()` function from the `multcomp` package

```
library(multcomp)
```

```
pairwise<-glht(reduced, linfct = mcp(Region= "Tukey"))
summary(pairwise)
```

```
##
##   Simultaneous Tests for General Linear Hypotheses
##
## Multiple Comparisons of Means: Tukey Contrasts
##
##
## Fit: lm(formula = Quality ~ Flavor + Region, data = Data)
##
## Linear Hypotheses:
##                       Estimate Std. Error t value Pr(>|t|)
## North - Napa == 0      -1.2234     0.4003  -3.056  0.01162 *
## Central - Napa == 0    -2.7569     0.4495  -6.134  < 1e-04 ***
## Central - North == 0   -1.5335     0.3688  -4.158  0.00054 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## (Adjusted p values reported -- single-step method)
```

We can see that there is a significant difference in the mean Quality rating for all pairs of regions, for given flavor rating, since these tests are all significant.

Given the negative values for the difference in the estimated coefficients, wines from the Napa valley have the highest ratings, followed by wines from the North region, and then wines from the Central region, when flavor rating is controlled.

These conclusions are not surprising given the scatterplot that we created earlier.