Categorical Predictors Tutorial

In this tutorial, we will use the data set `wine.txt`. The data set contains ratings of various wines produced in California. 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
```

1 Data Wrangling

Notice from the description that the variable `Region` is recorded numerically, even though it is a categorical variable. We need to make sure that R is viewing its type correctly by applying the function `class()` to this variable:

```
##is Region a factor?
class(Data$Region)
```

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

We need to convert `Region` to be viewed as categorical by using `factor()`, otherwise R will treat it as a quantitative predictor and not use dummy coding:

```
##convert Region to factor
Data$Region<-factor(Data$Region)
##check Region is now the correct type
class(Data$Region)
```

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

We can check how the levels are being described:

```
##how are levels described
levels(Data$Region)
```

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

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

```r
##Give names to the levels
levels(Data$Region) <- c("North", "Central", "Napa")
levels(Data$Region)
```

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

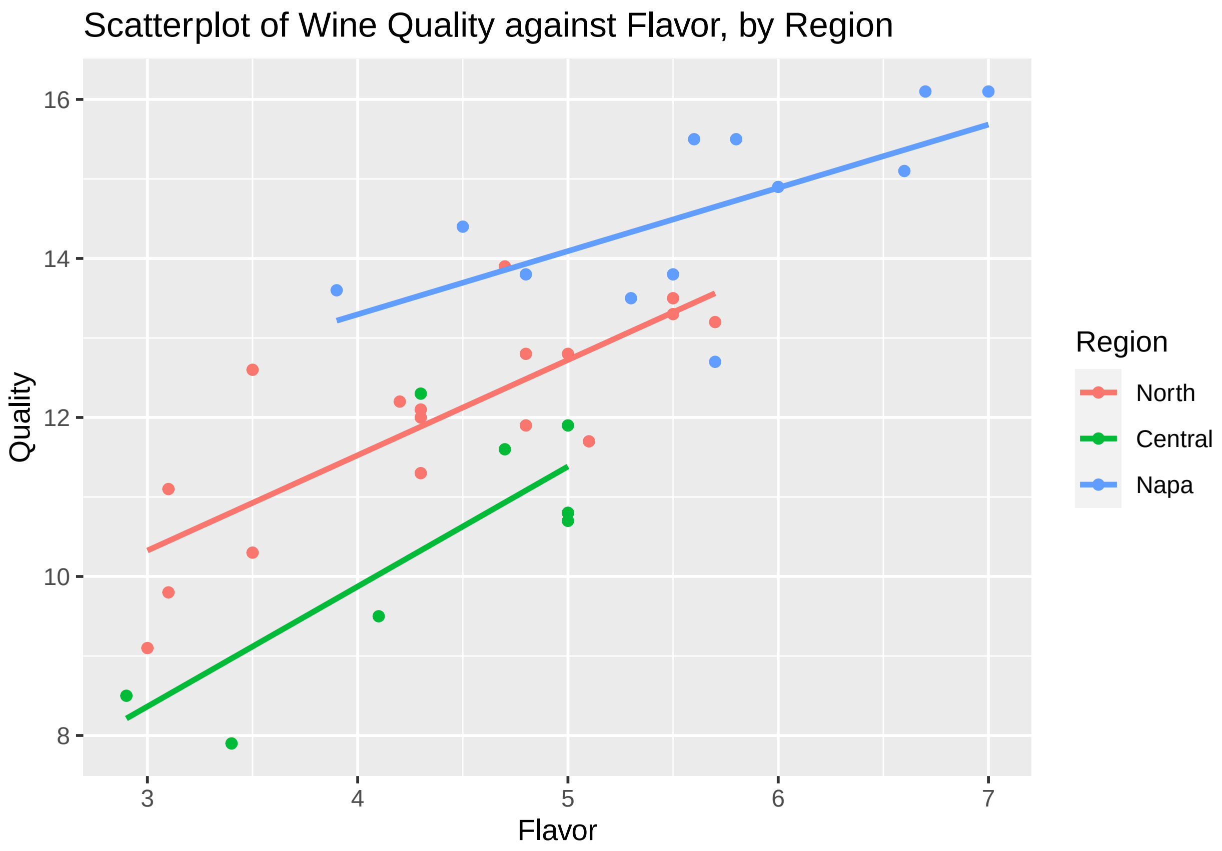
We have done the needed data wrangling: making sure categorical variables are viewed as factors, and giving descriptive names to the levels of the categorical variable.

Note: if categorical variables are already dummy coded, we do not need to convert them to factor when fitting MLR using `lm()`. The `lm()` function converts factors to dummy codes.

2 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:

```r
ggplot2::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")
```



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 there may exist an interaction between the region of the wine and its flavor; the impact of flavor on quality rating differs among the regions. So a regression model with interaction between region and flavor may be appropriate.

3 Fitting MLR

Since the categorical variable `Region` has three levels, we know that there will be two indicator variables created to represent the various regions. We check the dummy coding using the `contrasts()` function:

```
##check dummy coding
contrasts(Data$Region)
```

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

This output informs us the `North` region is the reference class, as it is coded 0 for both indicator variables. We can change the reference class to the `Napa` region via the `relevel()` function:

```
##Set a different reference class
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
```

```r
##notice hierarchical principle is used
```

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

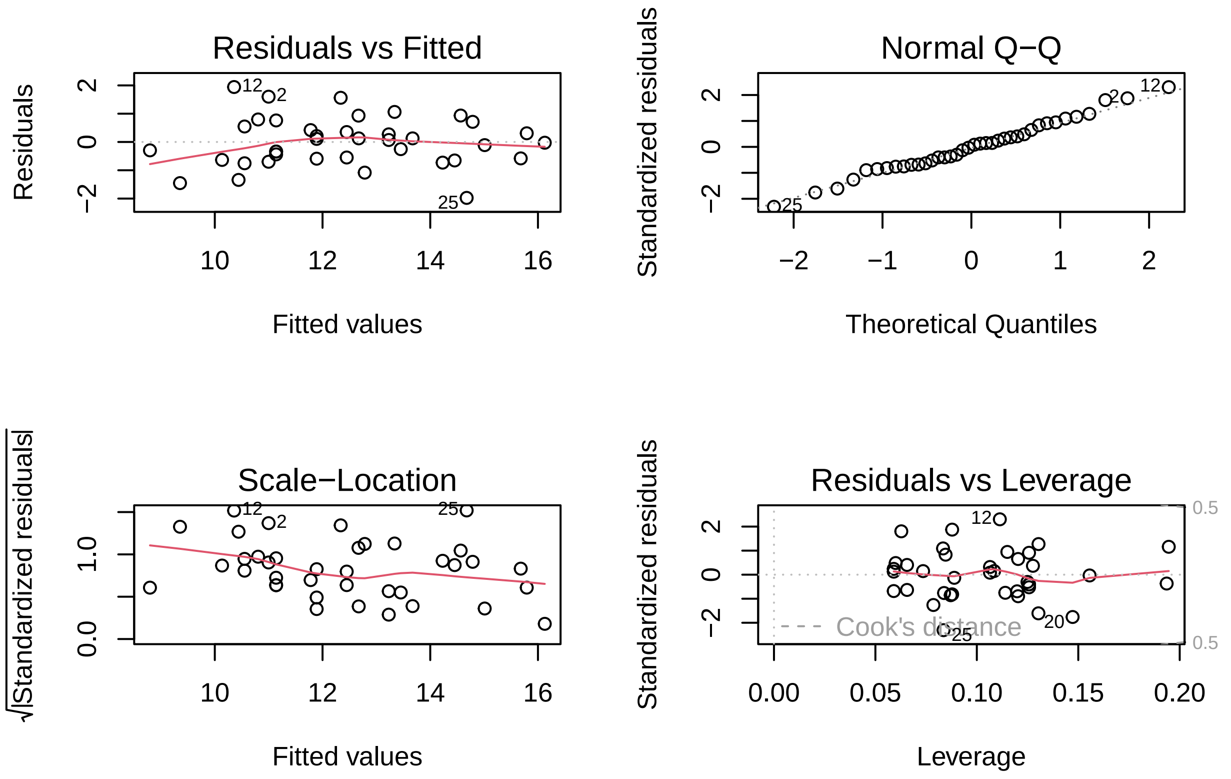
```r
##fit regression with no interaction
reduced<-lm(Quality~Flavor+Region, data=Data)
##general linear F test for interaction terms
anova(reduced,result)
```

```
## 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.

The regression assumptions when a categorical predictor is involved are pretty much the same, assessed similarly as before.

```r
par(mfrow=c(2,2))
plot(reduced)
```



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. Our regression equation is

$$E(y|x) = \beta_0 + \beta_1 x_1 + \beta_2 I_1 + \beta_3 I_2$$

where $I_1 = 1$ if `North` and 0 otherwise, $I_2 = 1$ if `Central` and 0 otherwise.

Plugging in the values of the indicator variables, we can write regression equations for each region

- `North`: $E(y|x) = \beta_0 + \beta_2 + \beta_1 x_1$.
- `Central`: $E(y|x) = \beta_0 + \beta_3 + \beta_1 x_1$.
- `Napa`: $E(y|x) = \beta_0 + \beta_1 x_1$.

Since there are 3 levels, there will be 3 possible pairs of regions to compare (while controlling for flavor rating):

- `North` and `Napa`: denoted by β_2
- `Central` and `Napa`: denoted by β_3
- `North` and `Central`: $\beta_2 - \beta_3$

Let's take a look at the estimated coefficients

```
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
```

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

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 to use multiple comparison methods to ensure that the probability of making at least one Type I error is at most the significance level of 0.05.

4.1 Bonferroni procedure

As noted earlier, there are 3 pairs of regions to compare (while controlling for flavor rating):

- `North` and `Napa`: denoted by β_2
- `Central` and `Napa`: denoted by β_3
- `North` and ‘Central: $\beta_2 - \beta_3$

The t multiplier and critical value is now $t_{1-\frac{\alpha}{2g},n-p}$, which we can find with:

```
n<-dim(Data)[1]
p<-4
g<-3
t.bon<-qt(1-0.05/(2*g), n-p)
t.bon
```

```
## [1] 2.518259
```

4.1.1 Pairwise comparison with reference class

For the difference in mean quality rating between wines in the North and Napa regions, when controlling for flavor, we use the confidence interval for β_2, i.e.,

$$\hat{\beta}_2 \pm t_{1-\frac{\alpha}{2g},n-p} se(\hat{\beta}_2) = -1.2234 \pm 2.518259 \times 0.4003 = (-2.2315, -0.2153)$$

which excludes 0, so we have a significant difference in the mean quality rating between wines in the North and Napa regions, when controlling for flavor. Since the interval is negative, we can say that the mean quality rating for wines in the Napa region is higher than the North region, when controlling for flavor.

To conduct the hypothesis test $H_0 : \beta_2 = 0, H_a : \beta_2 \neq 0$, we can use the critical value (which is the same as the t multiplier) of 2.518259. The test statistic is

$$\frac{\hat{\beta}_2}{se(\hat{\beta}_2)} = \frac{-1.2234}{0.4003} = -3.056$$

which is larger in magnitude than the critical value, so we reject the null hypothesis.

Notice the result from the confidence interval and hypothesis test are consistent at the same level of significance.

4.1.2 Pairwise comparison excluding reference class

Comparisons excluding the reference require a bit more work. For the difference in mean quality rating between wines in the North and Central regions, when controlling for flavor, we use the CI for $\beta_2 - \beta_3$:

$$(\hat{\beta}_2 - \hat{\beta}_3) \pm t_{1-\frac{\alpha}{2g},n-p} se(\hat{\beta}_2 - \hat{\beta}_3)$$

The output from `summary()` does not provide $se(\hat{\beta}_2 - \hat{\beta}_3)$. To obtain the values to calculate this, we need to produce the variance-covariance matrix for the estimated coefficients

```
vcov(reduced)
```

```
##                (Intercept)      Flavor RegionNorth RegionCentral
## (Intercept)      1.0201363 -0.16975148 -0.27722389   -0.27700199
## Flavor          -0.1697515  0.03022282  0.03748222    0.03744271
## RegionNorth     -0.2772239  0.03748222  0.16026554    0.11313506
## RegionCentral   -0.2770020  0.03744271  0.11313506    0.20201780
```

So we have

$$Var(\hat{\beta}_2 - \hat{\beta}_3) = Var(\hat{\beta}_2) + Var(\hat{\beta}_3) - 2Cov(\hat{\beta}_2, \hat{\beta}_3) = 0.1603 + 0.2020 - 2 \times 0.1131 = 0.1361$$

Therefore, the CI for $\beta_2 - \beta_3$ is

$$(-1.2234 - -2.7569) \pm 2.518259 \times \sqrt{0.1361} = (0.6045, 2.4625)$$

which excludes 0, so we have a significant difference in the mean quality rating between wines in the North and Central regions, when controlling for flavor. Since the interval is positive, we can say that the mean quality rating for wines in the North region is higher than the Central region, when controlling for flavor.

To perform the hypothesis test $H_0 : \beta_2 - \beta_3 = 0, H_a : \beta_2 - \beta_3 \neq 0$, we need to calculate the t-statistic

$$t - stat = \frac{\hat{\beta}_2 - \hat{\beta}_3}{se(\hat{\beta}_2 - \hat{\beta}_3)} = \frac{-1.2234 - -2.7569}{\sqrt{0.1361}} = 4.1568$$

which is larger in magnitude than the critical value of 2.518259, so we reject the null hypothesis. Again, the conclusions from the hypothesis test and CI are consistent.

4.2 Tukey procedure

Another procedure for multiple pairwise comparisons is the Tukey procedure. We use the `glht()` function from the `multcomp` package

```r
library(multcomp)
pairwise<-multcomp::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   0.0117 *
## Central - Napa == 0    -2.7569     0.4495  -6.134   <0.001 ***
## Central - North == 0   -1.5335     0.3688  -4.158   <0.001 ***
## ---
## 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.