HW2

Grace Cui

10/22/2017

```
redwine <- read.table('/Users/yuecui/Desktop/Everything Starts with Data/Week6/redwine.txt',
                      stringsAsFactors = F,header = T)
attach(redwine)
```

Problem 1

Calculate the averages of RS and SD by ignoring the missing values.

```
mean(RS, na.rm = T)
```

```
## [1] 2.537952
```

```
mean(SD, na.rm = T)
```

```
## [1] 46.29836
```

Problem 2

Create vectors of SD.obs and FS.obs by omitting observations with missing values in SD.

```
SD.obs <- SD[is.na(SD) == F]
FS.obs <- FS[is.na(SD) == F]
```

Build (simple) linear regression model to estimate SD.obs using FS.obs.

```
m1<-lm(SD.obs~FS.obs)
coef(m1)
```

```
## (Intercept)      FS.obs
##   13.185505    2.086077
```

Problem 3

Create a vector (of length 17) of estimated SD values using the regression model in Problem 2 and FS values of the observations with missing SD values.

```
SD.na <- predict(m1, data.frame(FS.obs = FS[is.na(SD) == T] ))
```

Impute missing values of SD using the created vector. Print out the average of SD after the imputation.

```
redwine$SD[is.na(SD) == T] <- SD.na
mean(redwine$SD)
```

```
## [1] 46.30182
```

Problem 4

Impute missing values of RS using the average value imputation method from the lab. Print out the average of RS after the imputation.

```
avg.imp <- function(a, avg){
  missing <- is.na(a)
  n.missing <- sum(missing)
  a.obs <- a[!missing]
  imputed <- a
  imputed[missing] <- avg
  return(imputed)
}
RSavg = mean(na.omit(RS))
RSavgimp = avg.imp(RS, RSavg)
redwine$RS<-RSavgimp
mean(redwine$RS)
```

```
## [1] 2.537952
```

Problem 5

Build multiple linear regression model for the new data set and save it as winemodel. Print out the coefficients of the regression model.

```
m2<-lm(QA~., data=redwine)
coef(m2)
```

```
##   (Intercept)            FA            VA            CA            RS
##  47.202815335   0.068406796  -1.097686420  -0.178949797   0.025926958
##            CH            FS            SD            DE            PH
##  -1.631290466   0.003530106  -0.002854970 -44.816652166   0.035996993
##            SU            AL
##   0.944871182   0.247046550
```

Problem 6

Print out the summary of the model. Pick one attribute that is least likely to be related to QA based on p-values.

```
summary(m2)
```

```
##
## Call:
## lm(formula = QA ~ ., data = redwine)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.78010 -0.36249 -0.06331  0.44595  1.98828
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.720e+01  1.782e+01   2.649 0.008151 **
## FA           6.841e-02  1.872e-02   3.654 0.000267 ***
## VA          -1.098e+00  1.213e-01  -9.053  < 2e-16 ***
```

```
## CA          -1.789e-01  1.474e-01  -1.214 0.224954
## RS           2.593e-02  1.419e-02   1.827 0.067944 .
## CH          -1.631e+00  4.097e-01  -3.982 7.14e-05 ***
## FS           3.530e-03  2.159e-03   1.635 0.102262
## SD          -2.855e-03  7.248e-04  -3.939 8.54e-05 ***
## DE          -4.482e+01  1.789e+01  -2.505 0.012329 *
## PH           3.600e-02  4.409e-02   0.816 0.414413
## SU           9.449e-01  1.136e-01   8.321  < 2e-16 ***
## AL           2.470e-01  2.265e-02  10.906  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6491 on 1587 degrees of freedom
## Multiple R-squared:  0.3584, Adjusted R-squared:  0.354
## F-statistic:  80.6 on 11 and 1587 DF,  p-value: < 2.2e-16
```

PH is least likely to be related to QA because it has the largest p-value.

Problem 7

Perform 5-fold cross validation for the model you just built. Print out the average error rate.

```r
library(boot)
m2<-glm(QA~., data=redwine)
cv<-cv.glm(data=redwine, glmfit = m2, K = 5)
cv$delta
```

```
## [1] 0.4249947 0.4242240
```

Problem 8

Calculate the average μ and standard deviation σ of the selected attribute.

```r
mu_ph<-mean(PH)
mu_sigma<-sd(PH)
```

Create a new data set after removing observations that is outside of the range and name the data set as redwine2.

```r
lb = mu_ph - 3*mu_sigma
ub = mu_ph + 3*mu_sigma
redwine2<-subset(redwine,PH<ub & PH>lb)
```

```r
dim(redwine2)
```

```
## [1] 1580   12
```

Problem 9

Build regression model winemodel2 using the new data set from Problem 8 and print out the summary.

```r
m3<-lm(QA~., data=redwine2)
summary(m3)
```

```
##
## Call:
## lm(formula = QA ~ ., data = redwine2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.68933 -0.36336 -0.04368  0.45221  2.01272
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  19.036170  21.211609   0.897   0.3696
## FA            0.024613   0.026019   0.946   0.3443
## VA           -1.072147   0.122031  -8.786  < 2e-16 ***
## CA           -0.178017   0.148120  -1.202   0.2296
## RS            0.012955   0.014968   0.866   0.3869
## CH           -1.902552   0.420766  -4.522 6.60e-06 ***
## FS            0.004421   0.002182   2.026   0.0429 *
## SD           -0.003145   0.000738  -4.261 2.16e-05 ***
## DE          -14.973653  21.652465  -0.692   0.4893
## PH           -0.424704   0.192653  -2.205   0.0276 *
## SU            0.913456   0.114860   7.953 3.46e-15 ***
## AL            0.282744   0.026553  10.648  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6475 on 1568 degrees of freedom
## Multiple R-squared:  0.3629, Adjusted R-squared:  0.3585
## F-statistic: 81.21 on 11 and 1568 DF,  p-value: < 2.2e-16
```

Compare this model with the model obtained in Problem 6 and decide which one is better.

This model is better than the last one since the R^2_{adj} increased from 0.3516 to 0.358.

Pick 5 attributes that is most likely to be related to QA based on p-values.

VA, SD, PH, SU, AL are the 5 attributes that are most likely to be related to QA because they all have p-values less than 0.05, which means their coefficients are all statistically significant.