LabAssignment2

Tian Fu

11/4/2018

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.0.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.6
## v tidyr   0.8.1     v stringr 1.3.1
## v readr   1.1.1     v forcats 0.3.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
redwine <- read.table('~/Desktop/MSiA400/redwine.txt',header=T)
```

Problem 1

```
redwine[!complete.cases(redwine),] # RS and SD have missing values
```

```
##     QA   FA    VA   CA  RS    CH   FS  SD      DE   PH   SU   AL
## 15   5  8.9 0.620 0.18  NA 0.176 52.0 145 0.99860 3.16 0.88  9.2
## 33   5  8.3 0.655 0.12 2.3 0.083 15.0  NA 0.99660 3.17 0.66  9.8
## 37   6  7.8 0.600 0.14  NA 0.086  3.0  15 0.99750 3.42 0.60 10.8
## 40   5  7.3 0.450 0.36 5.9 0.074 12.0  NA 0.99780 3.33 0.83 10.5
## 83   5  7.4 0.500 0.47  NA 0.086 21.0  73 0.99700 3.36 0.57  9.1
## 91   5  7.9 0.520 0.26  NA 0.079 42.0 140 0.99640 3.23 0.54  9.5
## 101  6  8.3 0.610 0.30 2.1 0.084 11.0  NA 0.99720 3.40 0.61 10.2
## 151  6  7.3 0.330 0.47  NA 0.077  5.0  11 0.99580 3.33 0.53 10.3
## 186  5  8.9 0.310 0.57  NA 0.111 26.0  85 0.99710 3.26 0.53  9.7
## 198  6 11.5 0.300 0.60 2.0 0.067 12.0  NA 0.99810 3.11 0.97 10.1
## 243  6  7.7 0.580 0.10  NA 0.102 28.0 109 0.99565 3.08 0.49  9.8
## 294  6  6.9 0.360 0.25  NA 0.098  5.0  16 0.99640 3.41 0.60 10.1
## 349  6  9.6 0.560 0.31  NA 0.089 15.0  46 0.99790 3.11 0.92 10.0
## 355  6  6.1 0.210 0.40 1.4 0.066 40.5  NA 0.99120 3.25 0.59 11.9
## 399  6 11.5 0.590 0.59  NA 0.087 13.0  49 0.99880 3.18 0.65 11.0
## 456  8 11.3 0.620 0.67  NA 0.086  6.0  19 0.99880 3.22 0.69 13.4
## 531  6  9.1 0.220 0.24 2.1 0.078  1.0  NA 0.99900 3.41 0.87 10.3
## 565  6 13.0 0.470 0.49  NA 0.085  6.0  47 1.00210 3.30 0.68 12.7
## 601  4  8.2 0.915 0.27 2.1 0.088  7.0  NA 0.99620 3.26 0.47 10.0
## 685  5  9.8 0.980 0.32 2.3 0.078 35.0  NA 0.99800 3.25 0.48  9.4
## 740  5  9.0 0.690 0.00  NA 0.088 19.0  38 0.99900 3.35 0.60  9.3
## 746  6  7.3 0.510 0.18  NA 0.070 12.0  28 0.99768 7.88 0.73  9.5
## 757  6  6.3 0.980 0.01 2.0 0.057 15.0  NA 0.99488 3.60 0.46 11.2
## 802  5  8.6 0.550 0.09  NA 0.068  8.0  17 0.99735 3.23 0.44 10.0
## 830  6  5.9 0.610 0.08  NA 0.071 16.0  24 0.99376 3.56 0.77 11.1
## 838  7  6.7 0.280 0.28 2.4 0.012 36.0  NA 0.99064 3.26 0.39 11.7
## 939  7  7.2 0.380 0.38 2.8 0.068 23.0  NA 0.99356 3.34 0.72 12.9
## 940   5  6.2 0.460 0.17  NA 0.073  7.0  11 0.99425 3.61 0.54 11.4
## 991   5  7.7 0.390 0.12  NA 0.097 19.0  27 0.99596 3.16 0.49  9.4
## 1017  7  8.9 0.380 0.40 2.2 0.068 12.0  NA 0.99486 3.27 0.75 12.6
## 1058  5  7.6 0.420 0.25  NA 0.104 28.0  90 0.99784 3.15 0.57  9.1
## 1092  6  7.9 0.340 0.42 2.0 0.086  8.0  NA 0.99546 3.35 0.60 11.4
## 1143  6  6.9 0.450 0.11  NA 0.043  6.0  12 0.99354 3.30 0.65 11.4
## 1167  5  9.9 0.540 0.26 2.0 0.111  7.0  NA 0.99709 2.94 0.98 10.2
## 1215  6 10.2 0.330 0.46  NA 0.081  6.0   9 0.99628 3.10 0.48 10.4
## 1248  5  7.4 0.550 0.19 1.8 0.082 15.0  NA 0.99655 3.49 0.68 10.5
## 1309  5  9.7 0.690 0.32  NA 0.088 22.0  91 0.99790 3.29 0.62 10.1
## 1320  6  9.1 0.760 0.68 1.7 0.414 18.0  NA 0.99652 2.90 1.33  9.1
## 1392  5  8.0 0.640 0.22 2.4 0.094  5.0  NA 0.99612 3.37 0.58 11.0
```

```
avg_rs <- mean(redwine$RS, na.rm=TRUE)
avg_rs
```

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

```
avg_sd <- mean(redwine$SD, na.rm=TRUE)
avg_sd
```

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

From above, after ignoring the missing values, average of RS is 2.537952 and average of SD is 46.29836

Problem 2

```
#cor(na.omit(redwine))
#cor(redwine, use = "pairwise.complete.obs")
# omitting observations with missing values in SD
SD_vec <- redwine[complete.cases(redwine$SD),]$SD
FS_vec <- redwine[complete.cases(redwine$SD),]$FS
fit <- lm(SD_vec~FS_vec)
coefficients(fit)
```

```
## (Intercept)      FS_vec
##   13.185505    2.086077
```

From above, in the model SD_vec $= \beta_0 + \beta_1$FS_vec, the coefficients are:

$\hat{\beta}_0 = 13.185505$ and $\hat{\beta}_1 = 2.086077$

Problem 3

```
# FS values of the observations with missing SD values
FS_input <- redwine[!complete.cases(redwine$SD),]$FS
# estimated SD based on linear regression results above
SD_estimate <- predict(fit, data.frame(FS_vec=FS_input))
SD_estimate
```

```
##        1        2        3        4        5        6        7        8
## 44.47667 38.21843 36.13236 38.21843 97.67164 15.27158 27.78805 86.19821
##        9       10       11       12       13       14       15       16
## 44.47667 88.28429 61.16528 38.21843 29.87412 27.78805 44.47667 50.73490
##       17
## 23.61589
```

```
ind <- which(is.na(redwine$SD)) # get indices of observations with missing SD
redwine[ind,'SD']<-SD_estimate # replace NA with estimated values
```

```
mean(redwine$SD)
```

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

The average of SD after the imputation is 46.30182

Problem 4

```
# impute missing values of RS using the its mean
redwine$RS[is.na(redwine$RS)] <- mean(redwine$RS, na.rm=TRUE)
```

```
mean(redwine$RS)
```

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

The average of RS after the imputation is 2.537952

Problem 5

```
sum(is.na(redwine)) # all missing values are imputed
```

```
## [1] 0
```

```
# build multiple linear regression model for the new data set
winemodel <- lm(QA~FA+VA+CA+RS+CH+FS+SD+DE+PH+SU+AL, data=redwine)
coefficients(winemodel) # coefficients of the model
```

```
##   (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
```

The coefficients of this regression model are shown above.

Problem 6

```
summary(winemodel) # summary of the model
```

```
##
## Call:
## lm(formula = QA ~ FA + VA + CA + RS + CH + FS + SD + DE + PH +
##     SU + AL, 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
```

From above, as the p-value of PH is 0.414413, which is nonsignificant and the largest among other p-values, pH(PH) is least likely to be related to quality(QA).

Problem 7

```
# Function for creating list of K index sets for K-fold CV
# n is sample size; K is number of parts;
# returns K-length list of indices for each part
CVInd <- function(n,K) {
   m<-floor(n/K)  #approximate size of each part
   r<-n-m*K
   I<-sample(n,n)  #random reordering of the indices
   Ind<-list()  #will be list of indices for all K parts
   length(Ind)<-K
   for (k in 1:K) {
      if (k <= r){
        # in the example of 5-fold CV for 1599 observations
        # first 4 sets will have 320 observations and 5th will have 319
        kpart <- ((m+1)*(k-1)+1):((m+1)*k)
      }
      else{
        kpart<-((m+1)*r+m*(k-r-1)+1):((m+1)*r+m*(k-r))
      }
      Ind[[k]] <- I[kpart]  #indices for kth part of data
   }
   Ind # return a list of indices
}
```

```
Nrep<-20 #number of replicates of CV
K<-5  #5-fold CV on each replicate
n=nrow(redwine) # total number of observations
y<-redwine$QA # QA is the response variable
SSE<-matrix(0,Nrep,1) # to store SSE for each test
for (j in 1:Nrep) {
  Ind<-CVInd(n,K) # randomly grouped list of indices
  yhat11<-y; # 11 predictor variables in the model
  for (k in 1:K) {
     # fit a model after excluding a set of indices as a test set
     model_cv <- lm(QA~FA+VA+CA+RS+CH+FS+SD+DE+PH+SU+AL,redwine[-Ind[[k]],])
     # use the fitted model to predict y values of the test set
     yhat11[Ind[[k]]]<-as.numeric(predict(model_cv,redwine[Ind[[k]],]))
  } #end of k loop
#sum((y-yhat11)/y)/n
  SSE[j,1]=sum((y-yhat11)^2)
} #end of j loop
#SSE
apply(SSE,2,mean)
```

```
## [1] 683.1171
```

Problem 8

```
mean_ph <- mean(redwine$PH)
mean_ph
```

```
## [1] 3.306202
```

```
sd_ph <- sd(redwine$PH)
sd_ph
```

```
## [1] 0.3924948
```

PH is the selected attribute, its average $\mu = 3.306202$ and its standard deviation $\sigma = 0.3924948$

```
# create a new dataset after removing observations that is outside of
# three standard deviations of the mean of PH
redwine2 <- redwine[(abs(redwine$PH-mean_ph) <= 3*sd_ph),]
```

```
dim(redwine)
```

```
## [1] 1599   12
```

```
dim(redwine2)
```

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

The dimension of redwine2 is shown above. As 1599-1580=19, there are 19 observations removed.

Problem 9

```r
# build a new model of the new data set
winemodel2 <- lm(QA~FA+VA+CA+RS+CH+FS+SD+DE+PH+SU+AL, data=redwine2)
summary(winemodel2)
```

```
## 
## Call:
## lm(formula = QA ~ FA + VA + CA + RS + CH + FS + SD + DE + PH + 
##     SU + AL, 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
```

Comparing with winemodel obtained from Problem 6, this model has a higher R^2, a higher adjusted R^2 and a higher overall F-statistic. Out of 11 predictor variables, both models have 7 predictors that are significant. So I think winemodel2 is better.

As shown above, five attributes with smallest p-values are VA, CH, SD, SU, AL. All of them are highly significant.