**MSiA 400**

**Lab Assignment 2**

**Problem 1**

Means were calculated by ignoring mean values.

Residual Sugar Mean (RS Mean) = 2.538

Total Sulfur Dioxide Mean (SD Mean) = 46.298

**Problem 2**

Simple linear regression fit of SD with FS:

SD = 13.186 + 2.086\*FS

Coefficients

Intercept: 13.185505

FS: 2.086077

**Problem 3**

Average of SD after imputation: 46.30182

Average of SD before imputation: 46.29836

There is not a very big change after imputation.

**Problem 4**

Average of RS after imputation: 2.537952

Average of RS before imputation: 2.537952

The average remains exactly the same after imputation.

**Problem 5**

Coefficients for regression model after all imputations:

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

**Problem 6**

Output of winemodel summary:

Call:

lm(formula = redwine$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

Judging by the p-values PH has the highest of 0.414, and so it is least likely to be related to QA.

**Problem 7**

In just one run of 5 fold CV the SSE for each fold was: 122,132,121,151,155

MSE for each fold was 0.38,0.41,0.38,0.47,0.48

(each fold had about about 1599/5~320 observations in test).

The total SSE for the CV run was : 681, the average for each fold was 681/5 = 136

The average MSE per fold was :0.424

The average SSE after 20 replicates of 5 fold CV was 684.245.

**Problem 8**

The dimension of redwine 2 is 1580 rows and 12 columns. The dimension of redwine is 1599 and 12 columns. There were 19 observations removed in order to remove outlying PH values.

**Problem 9**

Summary output of winemodel2:

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

PH has a much lower p-value, and therefore much more significant after the outliers were removed.

This model is better than the model in Problem 6 as it has a slightly higher R-squared and Adjusted R-squared values. This model fits the data better.

The attributes VA (volatile acidity), CH (chlorides), SD (total sulfur dioxide), SU (sulphates), AL (alcohol) are 5 very significant attributes as they have the lowest p-value. Therefore, they are most likely to be related to QA (quality).

**R Code Appendix**

SabCMSiA400LabAsgnmnt2

*Sabarish Chockalingam*

*November 13, 2018*

MSiA 400 - Lab Assignment 2

Problem 1

**library**(expm)

## Loading required package: Matrix

##

## Attaching package: 'expm'

## The following object is masked from 'package:Matrix':

##

## expm

redwine <- read.table('redwine.txt',header=TRUE)

RSmean <- mean(redwine$RS,na.rm=T)

SDmean <- mean(redwine$SD,na.rm=T)

RSmean

## [1] 2.537952

SDmean

## [1] 46.29836

Problem 2

sdfit <- lm(SD~FS,data=redwine, na.action=na.omit)

coefficients(sdfit)

## (Intercept) FS

## 13.185505 2.086077

Problem 3

naFS <- redwine['FS'][is.na(redwine$SD),]

*# getting predicted values*

predSD <- predict(sdfit,data.frame(FS=naFS))

*# mean before imputation*

print(paste("Average of SD before imputation: ",mean(redwine$SD[!is.na(redwine$SD)])))

## [1] "Average of SD before imputation: 46.2983565107459"

newSD <- redwine$SD

*# replacing na's with predicted values*

newSD[is.na(newSD)]<-predSD

*# getting mean with predicted values*

print(paste("Average of SD after imputation: ",mean(newSD)))

## [1] "Average of SD after imputation: 46.3018196746507"

Problem 4

*# function for average imputation*

avg.imp <- **function** (a, avg){

missing <- is.na(a)

imputed <- a

imputed[missing] <- avg

**return** (imputed)

}

newRS <- redwine$RS

*# average before imputation*

print(paste('Average of RS before imputation: ',mean(na.omit(newRS))))

## [1] "Average of RS before imputation: 2.53795180722892"

newRS <- avg.imp(newRS,mean(na.omit(newRS)))

*# average after imputation*

print(paste('Average RS after imputation: ',mean(newRS)))

## [1] "Average RS after imputation: 2.53795180722892"

Problem 5

*# replacing imputed columns in original data*

redwine$SD <- newSD

redwine$RS <- newRS

winemodel <- lm(redwine$QA~.,data=redwine)

coefficients(winemodel)

## (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

winecoeffs <- data.frame(coefficients(winemodel))

summary(winemodel)

##

## Call:

## lm(formula = redwine$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

Problem 7

*# install.packages('DAAG')*

**library**(DAAG)

## Loading required package: lattice

winemodel = lm(QA~.,data=redwine)

winemodelcv <- CVlm(form=winemodel, data = redwine, m=5,plotit=c('Observed','Residuals'),printit=F)

## Warning in CVlm(form = winemodel, data = redwine, m = 5, plotit = c("Observed", :

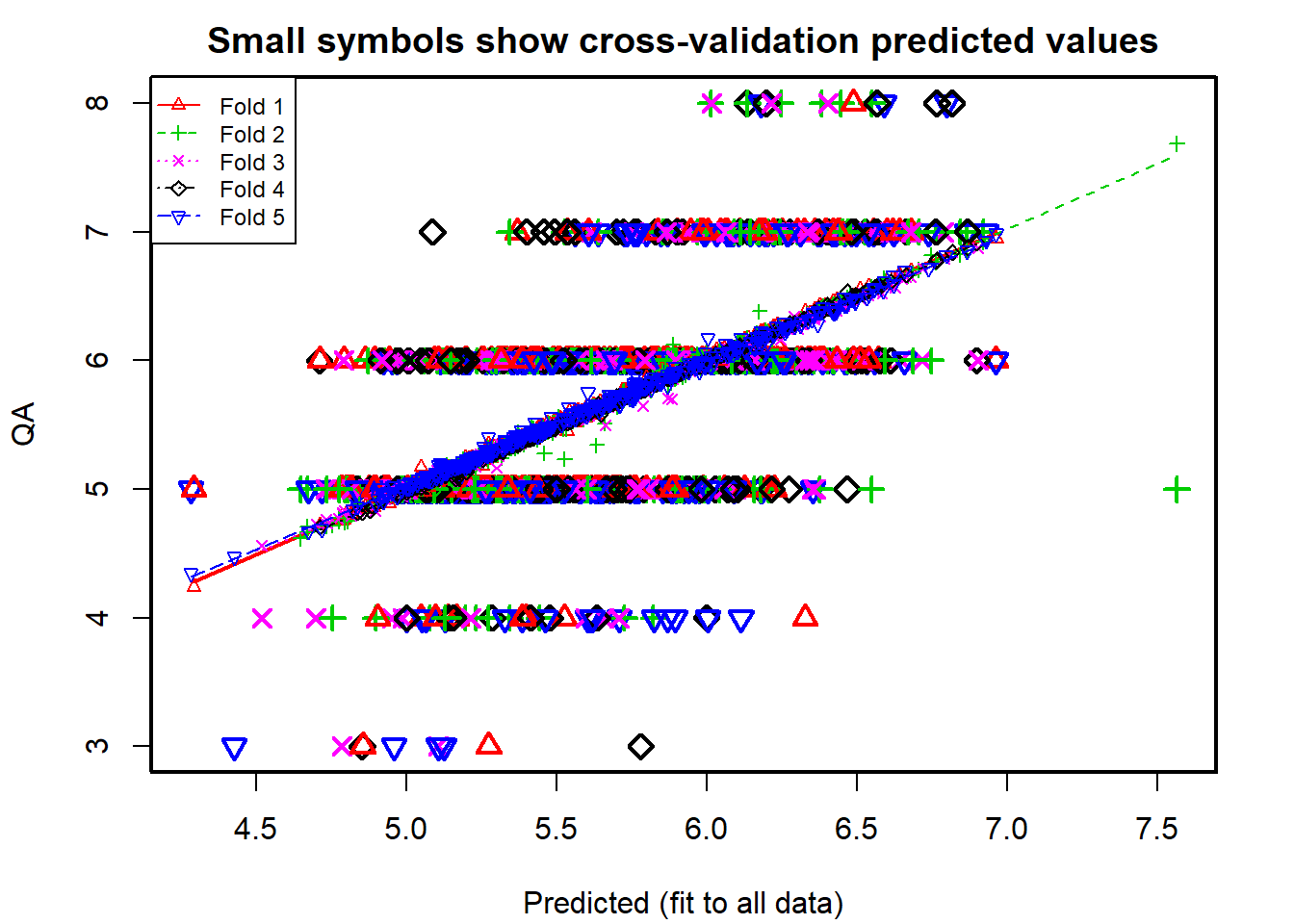
##

## As there is >1 explanatory variable, cross-validation

## predicted values for a fold are not a linear function

## of corresponding overall predicted values. Lines that

## are shown for the different folds are approximate

****

attr(winemodelcv,"ms")

## [1] 0.425802

Problem 7 - Repeating 5 fold CV 20 times

*#######Function for creating list of K index sets for K-fold CV##################*

CVInd <- **function**(n,K) { *#n is sample size; K is number of parts; returns K-length list of indices for each part*

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) 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

}

*####################################################################*

Nrep<-20 *#number of replicates of CV*

K<-5 *#K-fold CV on each replicate*

n=nrow(redwine)

y<-redwine$QA

p<-ncol(redwine)-1 *# number of predictors*

SSE<-matrix(0,Nrep,1)

MSE<-matrix(0,Nrep,1)

**for** (j **in** 1:Nrep) {

Ind<-CVInd(n,K)

yhat<-y;

**for** (k **in** 1:K) {

out1<-lm(QA~.,redwine[-Ind[[k]],])

yhat[Ind[[k]]]<-as.numeric(predict(out1,redwine[Ind[[k]],]))

} *#end of k loop*

SSE[j,]=sum((y-yhat)^2)

*#MSE[j,]=sum((y-yhat)^2)/(length(y)-p-1)*

} *#end of j loop*

*#SSE*

avgSSE<-apply(SSE,2,mean)

*#avgMSE<-apply(MSE,2,mean)*

print(paste("Average SSE from 20 runs of CV: ",avgSSE))

## [1] "Average SSE from 20 runs of CV: 682.547154156354"

*#print(paste("Average MSE from 20 runs of CV: ",avgMSE))*

Problem 8

*# calculating mean and average of seemingly insiginificant variable*

phmean <- mean(redwine$PH)

phsd <- sd(redwine$PH)

*# calculating lower and upper limit of outlier threshold*

phlow <- phmean - 3\*phsd

phhi <- phmean + 3\*phsd

redwine2 <- redwine[redwine$PH>phlow & redwine$PH<phhi,]

dim(redwine2)

## [1] 1580 12

Problem 9

*# calculating new winemodel after remove outlying PH values*

winemodel2 <- lm(QA~., data=redwine2)

summary(winemodel2)

##

## 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