Introduction

Objectives

Data

Methodology

set Y=sales price X1=finished square feet X2=number of bedrooms X3=number of bathrooms X4=air conditioning X5=garage size X6=pool X7=Year built X8=style X9=lot size X10=adjacent to highway X11=high quality X12=low quality , and among them X4, X6, X11 X12 are qualitative predictors, since the "quality" has three classes, I split it into 2 qualitative predictors:

X11=X12= here, I set median quality as the reference category.

> sale=read.table("estatsale.txt",header=F)

> head(sale)

V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13

1 1 360000 3032 4 4 1 2 0 1972 2 1 22221 0

2 2 340000 2058 4 2 1 2 0 1976 2 1 22912 0

3 3 250000 1780 4 3 1 2 0 1980 2 1 21345 0

4 4 205500 1638 4 2 1 2 0 1963 2 1 17342 0

5 5 275500 2196 4 3 1 2 0 1968 2 7 21786 0

6 6 248000 1966 4 3 1 5 1 1972 2 1 18902 0

> dim(sale)

[1] 522 13

> attach(sale)

> V6=factor(V6)

> V8=factor(V8)

> V13=factor(V13)

> higq=as.numeric(V10>=3) #set median quality as category reference

> hif=factor(higq)

> lowq=as.numeric(V10<=1)

> lowf=factor(lowq)

> sale=cbind(sale[,-10],hif,lowf)

> sale=sale[,2:14] #get rid of the first column cuz it has nothing to do with price

> colnames(sale)=c("Y","X1","X2","X3","X4","X5","X6","X7","X8","X9","X10","X11","X12")

> dim(sale)

[1] 522 13

> n=dim(sale)[[1]] ## number of observations

> n

[1] 522

> p=13 ## number of regression coefficients parameters

> head(sale)

Y X1 X2 X3 X4 X5 X6 X7 X8 X9 X10 X11 X12

1 360000 3032 4 4 1 2 0 1972 1 22221 0 0 0

2 340000 2058 4 2 1 2 0 1976 1 22912 0 0 0

3 250000 1780 4 3 1 2 0 1980 1 21345 0 0 0

4 205500 1638 4 2 1 2 0 1963 1 17342 0 0 0

5 275500 2196 4 3 1 2 0 1968 7 21786 0 0 0

6 248000 1966 4 3 1 5 1 1972 1 18902 0 0 0

> fullreg=lm(Y~.,data=sale)

> summary(fullreg)

Call:

lm(formula = Y ~ ., data = sale)

Residuals:

Min 1Q Median 3Q Max

-214251 -27600 -3322 22639 266535

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -2.496e+06 3.883e+05 -6.427 2.99e-10 \*\*\*

X1 1.042e+02 7.380e+00 14.114 < 2e-16 \*\*\*

X2 -5.053e+03 3.208e+03 -1.575 0.1159

X3 9.605e+03 4.212e+03 2.280 0.0230 \*

X4 2.467e+03 7.936e+03 0.311 0.7560

X5 9.207e+03 4.969e+03 1.853 0.0645 .

X6 8.180e+03 1.020e+04 0.802 0.4229

X7 1.262e+03 1.982e+02 6.364 4.40e-10 \*\*\*

X8 -6.324e+03 1.332e+03 -4.750 2.65e-06 \*\*\*

X9 1.389e+00 2.341e-01 5.933 5.51e-09 \*\*\*

X10 -3.481e+04 1.780e+04 -1.956 0.0510 .

X111 -5.969e+03 7.805e+03 -0.765 0.4447

X121 1.312e+05 1.030e+04 12.728 < 2e-16 \*\*\*

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

Residual standard error: 57560 on 509 degrees of freedom

Multiple R-squared: 0.8298, Adjusted R-squared: 0.8258

F-statistic: 206.8 on 12 and 509 DF, p-value: < 2.2e-16

> anova(fullreg)

Analysis of Variance Table

Response: Y

Df Sum Sq Mean Sq F value Pr(>F)

X1 1 6.6555e+12 6.6555e+12 2008.5701 < 2.2e-16 \*\*\*

X2 1 2.7613e+10 2.7613e+10 8.3332 0.004058 \*\*

X3 1 1.4271e+11 1.4271e+11 43.0687 1.304e-10 \*\*\*

X4 1 3.3417e+10 3.3417e+10 10.0850 0.001585 \*\*

X5 1 2.0019e+11 2.0019e+11 60.4158 4.280e-14 \*\*\*

X6 1 1.2314e+08 1.2314e+08 0.0372 0.847211

X7 1 2.3521e+11 2.3521e+11 70.9843 3.705e-16 \*\*\*

X8 1 2.6761e+11 2.6761e+11 80.7611 < 2.2e-16 \*\*\*

X9 1 1.0241e+11 1.0241e+11 30.9061 4.377e-08 \*\*\*

X10 1 1.5782e+10 1.5782e+10 4.7628 0.029537 \*

X11 1 7.0059e+09 7.0059e+09 2.1143 0.146544

X12 1 5.3677e+11 5.3677e+11 161.9916 < 2.2e-16 \*\*\*

Residuals 509 1.6866e+12 3.3135e+09

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

We have full model:

Yi=β0+β1Xi1+β2Xi2+ … +β11Xi11+β12Xi12+εi

From the R we know that MSE(X1,X2, …,X12)=3.3135e+09

I think "air conditioning", "pool", "style" are factors that do not affect "sales price" much. I use the extra sum of squares principle to test whether they are significant or not.

Null hypothesis: H0: β4=β6=β8=0

Alternative hypothesis: HA: Not allβ4, β6, β8 equal to zero

Under H0, we get the reduced model:

Yi=β0+β1Xi1+β2Xi2+β3Xi3+β5Xi5+β7Xi7+β9Xi9+β10Xi10+β11Xi11+β12Xi12+εi

> redureg=lm(Y~X1+X2+X3+X5+X7+X9+X10+X11+X12,data=sale)

> anova(redureg)

Analysis of Variance Table

Response: Y

Df Sum Sq Mean Sq F value Pr(>F)

X1 1 6.6555e+12 6.6555e+12 1929.4806 < 2.2e-16 \*\*\*

X2 1 2.7613e+10 2.7613e+10 8.0051 0.004848 \*\*

X3 1 1.4271e+11 1.4271e+11 41.3729 2.890e-10 \*\*\*

X5 1 2.2499e+11 2.2499e+11 65.2257 4.817e-15 \*\*\*

X7 1 2.3981e+11 2.3981e+11 69.5224 7.021e-16 \*\*\*

X9 1 1.5635e+11 1.5635e+11 45.3270 4.486e-11 \*\*\*

X10 1 7.5186e+09 7.5186e+09 2.1797 0.140456

X11 1 1.1992e+10 1.1992e+10 3.4767 0.062811 .

X12 1 6.7837e+11 6.7837e+11 196.6654 < 2.2e-16 \*\*\*

Residuals 512 1.7661e+12 3.4494e+09

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

> f=(((3.3417e+10)+(1.2314e+08)+(2.6761e+11))/3)/(3.3135e+09)

> f

[1] 30.29527

> qf(1-0.05,3,n-p-1)

[1] 2.622418

Test statistic F= = 30.29527 which is very large, greater than F(1-0.05,3;n-12-1)=2.622418. Therefore, we reject the null hypothesis at 0.05 level of significance and conclude that one or some predictors are significant.

Then I used both "forward selection" and "backward elimination" methods to select the right model.

> sale=as.data.frame(sale)

> null=lm(Y ~ 1, data = sale) ## Model with Intercept only ##

> full=lm(Y ~ ., data = sale) ## Model with all 3 predictors ##

> step(null, scope=list(lower=null, upper=full), direction="forward") ## Forward Selection ##

Start: AIC=12356.17

Y ~ 1

Df Sum of Sq RSS AIC

+ X1 1 6.6555e+12 3.2554e+12 11777

+ X12 1 5.5203e+12 4.3906e+12 11933

+ X3 1 4.6326e+12 5.2783e+12 12029

+ X5 1 3.3086e+12 6.6023e+12 12146

+ X7 1 3.0585e+12 6.8524e+12 12166

+ X11 1 2.5308e+12 7.3801e+12 12204

+ X2 1 1.6931e+12 8.2178e+12 12260

+ X8 1 1.2666e+12 8.6443e+12 12287

+ X4 1 8.2546e+11 9.0855e+12 12313

+ X9 1 4.9804e+11 9.4129e+12 12331

+ X6 1 2.1303e+11 9.6979e+12 12347

<none> 9.9109e+12 12356

+ X10 1 2.5746e+10 9.8852e+12 12357

…

Step: AIC=11451.31

Y ~ X1 + X12 + X7 + X9 + X8 + X3 + X10 + X5 + X2

Df Sum of Sq RSS AIC

<none> 1.6916e+12 11451

+ X11 1 2513859056 1.6891e+12 11452

+ X6 1 2226157397 1.6894e+12 11453

+ X4 1 1011510316 1.6906e+12 11453

Call:

lm(formula = Y ~ X1 + X12 + X7 + X9 + X8 + X3 + X10 + X5 + X2,

data = sale)

Coefficients:

(Intercept) X1 X12 X7 X9 X8 X3

-2.618e+06 1.052e+02 1.294e+05 1.321e+03 1.378e+00 -6.398e+03 1.095e+04

X10 X5 X2

-3.546e+04 9.885e+03 -4.983e+03

> step(full, data= sale, direction="backward") ## Backward Elimination ##

Start: AIC=11455.75

Y ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8 + X9 + X10 + X11 +

X12

Df Sum of Sq RSS AIC

- X4 1 3.2027e+08 1.6869e+12 11454

- X11 1 1.9382e+09 1.6885e+12 11454

- X6 1 2.1314e+09 1.6887e+12 11454

<none> 1.6866e+12 11456

- X2 1 8.2203e+09 1.6948e+12 11456

- X5 1 1.1376e+10 1.6980e+12 11457

- X10 1 1.2673e+10 1.6993e+12 11458

- X3 1 1.7231e+10 1.7038e+12 11459

- X8 1 7.4752e+10 1.7613e+12 11476

- X9 1 1.1663e+11 1.8032e+12 11489

- X7 1 1.3419e+11 1.8208e+12 11494

- X12 1 5.3677e+11 2.2234e+12 11598

- X1 1 6.6011e+11 2.3467e+12 11626

…

Step: AIC=11451.31

Y ~ X1 + X2 + X3 + X5 + X7 + X8 + X9 + X10 + X12

Df Sum of Sq RSS AIC

<none> 1.6916e+12 11451

- X2 1 8.0590e+09 1.6997e+12 11452

- X10 1 1.3182e+10 1.7048e+12 11453

- X5 1 1.3418e+10 1.7051e+12 11453

- X3 1 2.4496e+10 1.7161e+12 11457

- X8 1 7.6953e+10 1.7686e+12 11472

- X9 1 1.1895e+11 1.8106e+12 11485

- X7 1 1.7290e+11 1.8645e+12 11500

- X12 1 5.4185e+11 2.2335e+12 11594

- X1 1 6.8480e+11 2.3764e+12 11627

Call:

lm(formula = Y ~ X1 + X2 + X3 + X5 + X7 + X8 + X9 + X10 + X12,

data = sale)

Coefficients:

(Intercept) X1 X2 X3 X5 X7 X8

-2.618e+06 1.052e+02 -4.983e+03 1.095e+04 9.885e+03 1.321e+03 -6.398e+03

X9 X10 X12

1.378e+00 -3.546e+04 1.294e+05

> nreg=lm(Y~ X1+X2+X3+X5+X7+X8+X9+X10+X12,data=sale)

> anova(nreg) #MSE=3.3040e+09

Analysis of Variance Table

Response: Y

Df Sum Sq Mean Sq F value Pr(>F)

X1 1 6.6555e+12 6.6555e+12 2014.3810 < 2.2e-16 \*\*\*

X2 1 2.7613e+10 2.7613e+10 8.3573 0.004004 \*\*

X3 1 1.4271e+11 1.4271e+11 43.1933 1.223e-10 \*\*\*

X5 1 2.2499e+11 2.2499e+11 68.0957 1.328e-15 \*\*\*

X7 1 2.3981e+11 2.3981e+11 72.5815 < 2.2e-16 \*\*\*

X8 1 2.6733e+11 2.6733e+11 80.9116 < 2.2e-16 \*\*\*

X9 1 1.0366e+11 1.0366e+11 31.3729 3.480e-08 \*\*\*

X10 1 1.5830e+10 1.5830e+10 4.7912 0.029057 \*

X12 1 5.4185e+11 5.4185e+11 163.9989 < 2.2e-16 \*\*\*

Residuals 512 1.6916e+12 3.3040e+09

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

These two methods both dropped the X4, X6 and X11 and gave out the model:

Yi=β0+β1Xi1+β2Xi2+β3Xi3+β5Xi5+β7Xi7+β8Xi8+β9Xi9+β10Xi10+β12Xi12+ε

Then I split the original data set into three folds to validate this new regression model by calculating the MSPE's for both full regression model and this new regression model.

> CV.K.FOLD <- function(data, form, K = 2){

+

+ n <- dim(data)[[1]] ## number of observations ##

+ Y <- data$Y ## response variable ##

+ if(K > n) stop("K should be <= n, number of observations")

+ f <- ceiling(n/K)

+ s <- sample(rep(1:K, f), n) ## validation folds ##

+ ms <- max(s)

+ MSPE <- 0 ## initiation of the mean squared prediction error ##

+ coef <- NULL

+ MSE <- NULL

+ AY <- PY <- NULL

+

+ for(v in 1:ms){

+ j.in <- c(1:n)[(s != v)]

+ j.out <- c(1:n)[(s == v)]

+

+ data.train <- data[j.in,]

+ data.test <- data[j.out,]

+ data.train <- as.data.frame(data.train)

+ data.test <- as.data.frame(data.test)

+

+ fit.train <- lm(form, data = data.train) ## fitting model using training data ##

+ coef <- cbind(coef, coefficients(fit.train))

+ MSE <- c(MSE, summary(fit.train)$sigma^2)

+ pred.test <- predict(fit.train, newdata = data.test) ## prediction for the validated data ##

+ cat("Validation Set =", v, "\n\n\n") ## validation set number ##

+ cat("Fitted Model Using Training Data \n\n\n")

+ print(summary(fit.train)) ## fitted model to the training data ##

+ print(anova(fit.train)) ## anova table for the fitted model to the training data ##

+

+ cat("Observation Numbers =", c(1:n)[j.out], "\n\n\n") ## observation numbers in the validation set ##

+ cat("Actual Observations =", Y[j.out], "\n\n\n") ## observed response in the validation set ##

+ cat("Predicted Observations =", pred.test, "\n\n\n") ## predicted response in the validation set ##

+ MSPE <- MSPE + sum((Y[j.out] - pred.test)^2)

+ AY <- c(AY, Y[j.out])

+ PY <- c(PY, pred.test)

+ }

+

+ MSPE <- MSPE/n ## overall mean squared prediction error ##

+ cat("Overall Mean Squared Prediction Error", MSPE, "\n\n")

+ coef <- rbind(coef, MSE)

+ colnames(coef) <- paste("Fold", 1:K, sep = ":")

+ cat("Estimated Coefficients \n\n")

+ print(coef)

+ plot(AY, PY, xlab = "Observed Response", ylab = "Predicted Response", xlim = range(c(AY, PY)), ylim = range(c(AY, PY)), main = "")

+ }

>

> set.seed(1)

> form=as.formula(Y ~ .)

> CV.K.FOLD(data = sale, form, K = 3) #3416140279

Validation Set = 1

Fitted Model Using Training Data

Call:

lm(formula = form, data = data.train)

Residuals:

Min 1Q Median 3Q Max

-211373 -27229 -4195 21228 201744

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -2.679e+06 4.509e+05 -5.941 7.10e-09 \*\*\*

X1 9.871e+01 8.434e+00 11.704 < 2e-16 \*\*\*

X2 -3.238e+03 3.744e+03 -0.865 0.38778

X3 6.626e+03 4.903e+03 1.351 0.17747

X4 2.620e+03 8.972e+03 0.292 0.77045

X5 6.279e+03 6.123e+03 1.025 0.30587

X6 2.207e+04 1.116e+04 1.978 0.04875 \*

X7 1.359e+03 2.306e+02 5.893 9.23e-09 \*\*\*

X8 -4.515e+03 1.600e+03 -2.821 0.00507 \*\*

X9 1.541e+00 2.970e-01 5.189 3.67e-07 \*\*\*

X10 -2.567e+04 2.125e+04 -1.208 0.22799

X11 -7.688e+03 8.764e+03 -0.877 0.38100

X12 1.249e+05 1.181e+04 10.576 < 2e-16 \*\*\*

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

Residual standard error: 54580 on 335 degrees of freedom

Multiple R-squared: 0.8254, Adjusted R-squared: 0.8191

F-statistic: 131.9 on 12 and 335 DF, p-value: < 2.2e-16

…

Overall Mean Squared Prediction Error 3416140279

Estimated Coefficients

Fold:1 Fold:2 Fold:3

(Intercept) -2.678816e+06 -2.650901e+06 -2.234646e+06

X1 9.871423e+01 1.105980e+02 1.027207e+02

X2 -3.237885e+03 -6.554572e+03 -5.187250e+03

X3 6.626268e+03 8.816868e+03 1.328814e+04

X4 2.619922e+03 -8.856393e+03 1.035319e+04

X5 6.279111e+03 1.195729e+04 8.778897e+03

X6 2.207124e+04 -1.231129e+03 2.051197e+03

X7 1.359154e+03 1.345978e+03 1.120248e+03

X8 -4.514985e+03 -7.559016e+03 -6.395533e+03

X9 1.541413e+00 1.119841e+00 1.493775e+00

X10 -2.566717e+04 -5.323634e+04 -2.576610e+04

X11 -7.687536e+03 -7.695377e+03 -1.943602e+03

X12 1.248571e+05 1.253519e+05 1.395192e+05

MSE 2.979293e+09 3.391169e+09 3.589891e+09

> set.seed(1)

> form=as.formula(Y ~ X1+X2+X3+X5+X7+X8+X9+X10+X12)

> CV.K.FOLD(data = sale, form, K = 3)#3352320342

Validation Set = 1

Fitted Model Using Training Data

Call:

lm(formula = form, data = data.train)

Residuals:

Min 1Q Median 3Q Max

-199987 -25353 -3820 20919 195173

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -2.770e+06 4.246e+05 -6.525 2.48e-10 \*\*\*

X1 1.011e+02 8.360e+00 12.091 < 2e-16 \*\*\*

X2 -3.222e+03 3.735e+03 -0.863 0.38899

X3 8.856e+03 4.712e+03 1.879 0.06106 .

X5 7.525e+03 6.084e+03 1.237 0.21702

X7 1.401e+03 2.180e+02 6.424 4.50e-10 \*\*\*

X8 -4.825e+03 1.597e+03 -3.021 0.00271 \*\*

X9 1.488e+00 2.956e-01 5.033 7.85e-07 \*\*\*

X10 -2.671e+04 2.127e+04 -1.256 0.21003

X12 1.233e+05 1.173e+04 10.515 < 2e-16 \*\*\*

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

Residual standard error: 54750 on 338 degrees of freedom

Multiple R-squared: 0.8227, Adjusted R-squared: 0.818

F-statistic: 174.2 on 9 and 338 DF, p-value: < 2.2e-16

Analysis of Variance Table

…

Overall Mean Squared Prediction Error 3352320342

Estimated Coefficients

Fold:1 Fold:2 Fold:3

(Intercept) -2.770314e+06 -2.694709e+06 -2.385941e+06

X1 1.010810e+02 1.110401e+02 1.026581e+02

X2 -3.221991e+03 -6.826785e+03 -4.715903e+03

X3 8.855568e+03 9.442975e+03 1.408773e+04

X5 7.524553e+03 1.159563e+04 9.810685e+03

X7 1.400703e+03 1.361651e+03 1.199023e+03

X8 -4.825422e+03 -7.391173e+03 -6.430802e+03

X9 1.488133e+00 1.180582e+00 1.448560e+00

X10 -2.671412e+04 -5.354858e+04 -2.776491e+04

X12 1.232896e+05 1.248620e+05 1.386483e+05

MSE 2.998017e+09 3.371929e+09 3.572965e+09

The results have shown that the MSPE of new regression model is smaller than that of the initial full regression model, which implies that the new regression model is better than the initial one.

I noticed that some of the predictor variables may have interacting effects with some other predictor variables, for example, "number of bedrooms" versus "finished square feet", "number of bathrooms" versus "finished square feet", "garage size" versus "lot size", therefore I added some interaction terms into the regression model: xi1xi2, xi1xi3, xi5xi9 where xik = Xik - K so that we can reduce the multicollinearities and improve computational accuracy. My new polynomial regression model becomes:

Yi=β0+β1xi1+β2xi2+β3xi3+β5xi5+β7xi7+β8xi8+β9xi9+β10xi10+β12xi12+β13xi1xi2+β14xi1xi3+β15 xi5xi9+εi

> myfun=function(x) x-mean(x)

> salec=apply(salen[2:10],2,myfun)#xi=Xi-mean(Xi)

> sale2=cbind(sale[,1],salec)

> detach(sale)

> sale2=as.data.frame(sale2)

> colnames(sale2)=c("Y","x1","x2","x3","x5","x7","x8","x9","x10","x12")

> attach(sale2)

> x1x2=x1\*x2

> x1x3=x1\*x3

> x5x9=x5\*x9

> polreg=lm(Y~x1+x2+x3+x5+x7+x8+x9+x10+x12+x1x2+x1x3+x5x9)

> anova(polreg)

Analysis of Variance Table

Response: Y

Df Sum Sq Mean Sq F value Pr(>F)

x1 1 6.6555e+12 6.6555e+12 2086.5112 < 2.2e-16 \*\*\*

x2 1 2.7613e+10 2.7613e+10 8.6566 0.003407 \*\*

x3 1 1.4271e+11 1.4271e+11 44.7400 5.940e-11 \*\*\*

x5 1 2.2499e+11 2.2499e+11 70.5341 4.527e-16 \*\*\*

x7 1 2.3981e+11 2.3981e+11 75.1805 < 2.2e-16 \*\*\*

x8 1 2.6733e+11 2.6733e+11 83.8088 < 2.2e-16 \*\*\*

x9 1 1.0366e+11 1.0366e+11 32.4963 2.025e-08 \*\*\*

x10 1 1.5830e+10 1.5830e+10 4.9627 0.026336 \*

x12 1 5.4185e+11 5.4185e+11 169.8713 < 2.2e-16 \*\*\*

x1x2 1 6.2899e+10 6.2899e+10 19.7190 1.101e-05 \*\*\*

x1x3 1 1.7648e+09 1.7648e+09 0.5533 0.457334

x5x9 1 3.3851e+09 3.3851e+09 1.0612 0.303422

Residuals 509 1.6236e+12 3.1898e+09

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

> mse=3.1898e+09

> F=((6.2899e+10)+(1.7648e+09)+(3.3851e+09)/3)/mse

> F

[1] 20.6258

> qf(1-0.05,3,n-13)

[1] 2.622418

Test statistic F= = 20.6258 which is larger than F(1-0.05,3;n-13)=2.622418, therefore we reject the null hypothesis at 0.05 level of significance and conclude that one or more of the interaction terms should be included.

I then tested whether each of x1x2, x1x3,x5x9 should be included in the regression model :

i) Model: Yi=β0+β1xi1+β2xi2+β3xi3+β5xi5+β7xi7+β8xi8+β9xi9+β10xi10+β12xi12+β13xi1xi2+εi

Hypothesis: H0: β13 = 0; versus HA: β13 ≠ 0

> polreg1=lm(Y~x1+x2+x3+x5+x7+x8+x9+x10+x12+x1x2)

> anova(polreg1)

Analysis of Variance Table

Response: Y

Df Sum Sq Mean Sq F value Pr(>F)

x1 1 6.6555e+12 6.6555e+12 2088.0865 < 2.2e-16 \*\*\*

x2 1 2.7613e+10 2.7613e+10 8.6631 0.003395 \*\*

x3 1 1.4271e+11 1.4271e+11 44.7738 5.826e-11 \*\*\*

x5 1 2.2499e+11 2.2499e+11 70.5873 4.385e-16 \*\*\*

x7 1 2.3981e+11 2.3981e+11 75.2372 < 2.2e-16 \*\*\*

x8 1 2.6733e+11 2.6733e+11 83.8721 < 2.2e-16 \*\*\*

x9 1 1.0366e+11 1.0366e+11 32.5209 1.997e-08 \*\*\*

x10 1 1.5830e+10 1.5830e+10 4.9665 0.026278 \*

x12 1 5.4185e+11 5.4185e+11 169.9996 < 2.2e-16 \*\*\*

x1x2 1 6.2899e+10 6.2899e+10 19.7339 1.092e-05 \*\*\*

Residuals 511 1.6287e+12 3.1874e+09

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

> MSE1=3.1874e+09

> f21=((6.2899e+10)/1)/MSE1

> f21 #19.73364 included

[1] 19.73364

> qf(1-0.05,1,n-11) #3.859721

[1] 3.859721

Test statistic F= = 19.73364 which is larger than F(1-0.05,3;n-11)=3.859721, therefore we reject the null hypothesis at 0.05 level of significance and conclude that interaction term x1x2 should be included.

ii) Model: Yi=β0+β1xi1+β2xi2+β3xi3+β5xi5+β7xi7+β8xi8+β9xi9+β10xi10+β12xi12+β13xi1xi2+β14xi1xi3+εi

Hypothesis: H0: β14 = 0; versus HA: β14 ≠ 0

> polreg2=lm(Y~x1+x2+x3+x5+x7+x8+x9+x10+x12+x1x2+x1x3)

> anova(polreg2)

Analysis of Variance Table

Response: Y

Df Sum Sq Mean Sq F value Pr(>F)

x1 1 6.6555e+12 6.6555e+12 2086.2607 < 2.2e-16 \*\*\*

x2 1 2.7613e+10 2.7613e+10 8.6556 0.003409 \*\*

x3 1 1.4271e+11 1.4271e+11 44.7346 5.945e-11 \*\*\*

x5 1 2.2499e+11 2.2499e+11 70.5256 4.526e-16 \*\*\*

x7 1 2.3981e+11 2.3981e+11 75.1714 < 2.2e-16 \*\*\*

x8 1 2.6733e+11 2.6733e+11 83.7988 < 2.2e-16 \*\*\*

x9 1 1.0366e+11 1.0366e+11 32.4924 2.027e-08 \*\*\*

x10 1 1.5830e+10 1.5830e+10 4.9621 0.026344 \*

x12 1 5.4185e+11 5.4185e+11 169.8509 < 2.2e-16 \*\*\*

x1x2 1 6.2899e+10 6.2899e+10 19.7167 1.102e-05 \*\*\*

x1x3 1 1.7648e+09 1.7648e+09 0.5532 0.457360

Residuals 510 1.6270e+12 3.1902e+09

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

> MSE2=3.1902e+09

> f22=((1.7648e+09)/1)/MSE2

> f22 #0.5531942 excluded

[1] 0.5531942

> qf(1-0.05,1,n-12)#3.859757

[1] 3.859757

Test statistic F= = 0.5531942 which is very small and smaller than F(1-0.05,3;n-12)=3.859757, therefore we accept the null hypothesis at 0.05 level of significance and conclude that interaction term x1x3 should not be included.

iii) Model: Yi=β0+β1xi1+β2xi2+β3xi3+β5xi5+β7xi7+β8xi8+β9xi9+β10xi10+β12xi12+β13xi1xi2+β15xi5xi9+εi

Hypothesis: H0: β15 = 0; versus HA: β15 ≠ 0

> polreg3=lm(Y~x1+x2+x3+x5+x7+x8+x9+x10+x12+x1x2+x5x9)

> anova(polreg3)

Analysis of Variance Table

Response: Y

Df Sum Sq Mean Sq F value Pr(>F)

x1 1 6.6555e+12 6.6555e+12 2089.6631 < 2.2e-16 \*\*\*

x2 1 2.7613e+10 2.7613e+10 8.6697 0.003383 \*\*

x3 1 1.4271e+11 1.4271e+11 44.8076 5.744e-11 \*\*\*

x5 1 2.2499e+11 2.2499e+11 70.6406 4.300e-16 \*\*\*

x7 1 2.3981e+11 2.3981e+11 75.2940 < 2.2e-16 \*\*\*

x8 1 2.6733e+11 2.6733e+11 83.9354 < 2.2e-16 \*\*\*

x9 1 1.0366e+11 1.0366e+11 32.5454 1.976e-08 \*\*\*

x10 1 1.5830e+10 1.5830e+10 4.9702 0.026222 \*

x12 1 5.4185e+11 5.4185e+11 170.1279 < 2.2e-16 \*\*\*

x1x2 1 6.2899e+10 6.2899e+10 19.7488 1.084e-05 \*\*\*

x5x9 1 4.4138e+09 4.4138e+09 1.3858 0.239660

Residuals 510 1.6243e+12 3.1850e+09

---

Signif. codes: 0 ?\*\*?0.001 ?\*?0.01 ??0.05 ??0.1 ??1

> MSE3=3.1850e+09

> f23=((4.4138e+09)/1)/MSE3

> f23 # 1.385808 excluded

[1] 1.385808

> qf(1-0.05,1,n-12)#3.859757

[1] 3.859757

Test statistic F = = 1.385808 which is smaller than F(1-0.05,3;n-12)=3.859757, therefore we accept the null hypothesis at 0.05 level of significance and conclude that interaction term x5x9 should not be included.

Hence, I get the potential final model(only xi1xi2 included):

Yi=β0+β1xi1+β2xi2+β3xi3+β5xi5+β7xi7+β8xi8+β9xi9+β10xi10+β12xi12+β13xi1xi2+εi

where xik = Xik - K

> polregn=polreg1

> salepol=cbind(sale2,x1x2)

> head(salepol)

Y x1 x2 x3 x5 x7 x8 x9 x10

1 360000 771.37356 0.5287356 1.3582375 -0.09961686 5.095785 -2.344828 -2148.705 -0.0210728

2 340000 -202.62644 0.5287356 -0.6417625 -0.09961686 9.095785 -2.344828 -1457.705 -0.0210728

3 250000 -480.62644 0.5287356 0.3582375 -0.09961686 13.095785 -2.344828 -3024.705 -0.0210728

4 205500 -622.62644 0.5287356 -0.6417625 -0.09961686 -3.904215 -2.344828 -7027.705 -0.0210728

5 275500 -64.62644 0.5287356 0.3582375 -0.09961686 1.095785 3.655172 -2583.705 -0.0210728

6 248000 -294.62644 0.5287356 0.3582375 2.90038314 5.095785 -2.344828 -5467.705 -0.0210728

x12 x1x2

1 -0.1302682 407.8527

2 -0.1302682 -107.1358

3 -0.1302682 -254.1243

4 -0.1302682 -329.2048

5 -0.1302682 -34.1703

6 -0.1302682 -155.7795

Again, validate this potential final model by checking its MSPE, we get a MSPE even smaller than that of the regression model: Yi=β0+β1Xi1+β2Xi2+β3Xi3+β5Xi5+β7Xi7+β8Xi8+β9Xi9+β10Xi10+β12Xi12+ε. Therefore we can say that it is the best model so far.

> set.seed(1)

> form=as.formula(Y ~ x1+x2+x3+x5+x7+x8+x9+x10+x12+x1x2)

> CV.K.FOLD(data = salepol, form, K = 3)#3245551663

Validation Set = 1

Fitted Model Using Training Data

Call:

lm(formula = form, data = data.train)

Residuals:

Min 1Q Median 3Q Max

-201284 -25741 -4611 22476 189120

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 2.829e+05 3.163e+03 89.430 < 2e-16 \*\*\*

x1 1.168e+02 8.850e+00 13.200 < 2e-16 \*\*\*

x2 -1.369e+03 3.656e+03 -0.374 0.708421

x3 4.463e+03 4.686e+03 0.952 0.341589

x5 3.664e+03 5.979e+03 0.613 0.540419

x7 1.410e+03 2.121e+02 6.650 1.18e-10 \*\*\*

x8 -5.594e+03 1.563e+03 -3.579 0.000396 \*\*\*

x9 1.452e+00 2.877e-01 5.047 7.36e-07 \*\*\*

x10 -3.133e+04 2.071e+04 -1.513 0.131319

x12 1.266e+05 1.143e+04 11.075 < 2e-16 \*\*\*

x1x2 -1.583e+01 3.514e+00 -4.505 9.18e-06 \*\*\*

…

Overall Mean Squared Prediction Error 3245551663

Estimated Coefficients

Fold:1 Fold:2 Fold:3

(Intercept) 2.829113e+05 2.846248e+05 2.818050e+05

x1 1.168202e+02 1.193048e+02 1.092764e+02

x2 -1.368560e+03 -5.030971e+03 -3.780249e+03

x3 4.462626e+03 7.691481e+03 1.257006e+04

x5 3.664014e+03 9.089633e+03 6.351240e+03

x7 1.410341e+03 1.231298e+03 1.126495e+03

x8 -5.594241e+03 -7.548730e+03 -6.341446e+03

x9 1.451839e+00 1.091535e+00 1.412697e+00

x10 -3.133248e+04 -5.435767e+04 -2.891604e+04

x12 1.265669e+05 1.299231e+05 1.457356e+05

x1x2 -1.582781e+01 -1.285795e+01 -1.119259e+01

MSE 2.836141e+09 3.255483e+09 3.499173e+09

Based on this potential final model, I did the "DFFITS", "Cook's Distance" and "DFBETAS" three tests respectively to find the influential outliers. From the "DFFITS" test, the 103rd observation has been found as an influential outlier whose |(DFFITS)103| =1.160169, which is greater than 1; And no influential outliers was founded in "Cook's Distance" or "DFBETAS". After deleting the 103rd observation, I used the "Variance Inflation Factor" to check if there is multicollinearity in the data set.

> yhati=fitted(polregn) # predicted values

> ei=residuals(polregn) # residuals

> X=as.matrix(cbind(1, salepol[, c(2:11)])) ## X matrix

> hat\_mat=X %\*% solve(t(X) %\*% X) %\*% t(X) ## hat matrix

> hii=diag(hat\_mat) ## diagonal elements of that hat matrix

> MSE=(summary(polregn)$sigma)^2 ## mean squared error

> s2ei=MSE \* (1 - hii) ## estimated variance of the ith residual

> I=matrix(0, nrow = dim(hat\_mat)[[1]], ncol = dim(hat\_mat)[[1]])

> diag(I)=1 ## I: identity matrix

> s2e=MSE \* (I - hat\_mat) ## estimated variance-covariance matrix of the residuals

> ri=ei/sqrt(s2ei) ## studentized residuals

> SSE=sum(ei^2) ## error sum of squares

> ti=ei \* sqrt((n - p - 1)/(SSE \* (1 - hii) - ei^2)) ## studentized deleted residuals

> alpha=0.05

## Computing DFFITS\_i ##

> dffits=ti \* sqrt(hii/(1 - hii))

> adffits=abs(dffits)

> cutoff= 1

> index\_of\_infl=c(1:n)[adffits > cutoff]

> index\_of\_infl # the 103rd observations are influential outliers

[1] 103

> adffits[103]

103

1.160169

#Cook's Distance

> Di=((ei^2)/ (p \* MSE)) \* (hii/(1 - hii)^2)

> ## percentile of F(p, n - p) distribution

> PERF=pf(Di, df1 = p, df2 = (n - p))

> head(cbind(Di, PERF))

Di PERF

1 1.924852e-04 7.765499e-23

2 1.507270e-03 4.971984e-17

3 1.196960e-05 1.120874e-30

4 3.551501e-05 1.317220e-27

5 4.273406e-04 1.383662e-20

6 9.328867e-04 2.206060e-18

> index\_of\_infl=c(1:n)[PERF >= 0.50]

> index\_of\_infl

integer(0)

> index\_of\_infl=c(1:n)[PERF >= 0.10]

> index\_of\_infl

integer(0)

#DFBETAS#

> DFBETAS <- function(data, form){

+ ## we assume that the response variable is in the first column of data ##

+ n <- dim(data)[[1]] ## number of observations ##

+ X <- as.matrix(cbind(1, data[, - 1])) ## X matrix

+ invXX <- solve(t(X) %\*% X) ## inverse of t(X)X ##

+ dfbetas <- NULL

+ for(i in 1:n){

+ j.in <- c(1:n)[(c(1:n) != i)]

+ data.train <- data[j.in,]

+ data.train <- as.data.frame(data.train)

+

+ fit.all <- lm(form, data = data) ## fitting model using all of the observations

+ coef.all <- coefficients(fit.all) # coefficients of the model that uses all of the observations

+

+ fit.train <- lm(form, data = data.train) ## fitting model using training data ##

+ coef.train <- coefficients(fit.train) # coefficients of the model that uses all of the observations

+

+ MSE\_i <- summary(fit.train)$sigma^2

+ c\_kk <- diag(invXX)

+

+ dfbetas\_i <- (coef.all - coef.train)/sqrt(MSE\_i \* c\_kk)

+ dfbetas <- rbind(dfbetas, dfbetas\_i)

+ }

+ rownames(dfbetas) <- c(1:n)

+ print(dfbetas)

+ }

>

> form <- as.formula(Y~x1+x2+x3+x5+x7+x8+x9+x10+x12+x1x2)

> DFBETAS(data = salepol, form)

>

> salepol2=salepol[-103,]

> dim(salepol2)

[1] 521 11

> VIF(salepol2)

Variance Inflation Factors = 4.528294 1.65068 3.01961 1.655881 1.651842 1.818359 1.138563 1.027423 1.833077 1.216957

Maximum VIF = 4.528294

Mean VIF = 1.954069

The mean VIF = 1.954069 which is not far away from 1, so we conclude that there is no indication of serious multicollinearity problems in this data set.

> salefinal=as.data.frame(salepol2)

> coefficients(polregn) # model coefficients

(Intercept) x1 x2 x3 x5 x7 x8

283180.544414 114.818420 -3443.085444 8520.105109 6654.938416 1247.857388 -6612.037858

x9 x10 x12 x1x2

1.319106 -37139.902271 134810.944492 -13.163860

The final fitted polynomial regression model is:

=283180.54+114.82x1-3443.09x2+8520.11x3+6654.94x5+1247.86x7-6612.04x8+1.32x9-37139.90x10+134810.94x12-13.163860x1x2

With 9 predictor variables and 521 observations.

Interpretations of estimated parameters:

