Final report (May.20 2017)

## Part A

> setwd("C:/Users/H Dou/Desktop/STAT 420/test")  
> training=read.csv("training.csv",header=TRUE)

> training=training[c(-1)]

1.parameter estimates

We first calculating parameter estimates in R:

>fit=lm(health~.,data=training)

>fit

``` So the results of the parameter estimates could be easily found in the list above or the region below.

Coefficients:

(Intercept) PCDH12 DLG5 BC038559 SHISA5 AF161342

-3.403259 -0.112777 -0.743025 -0.298175 0.292498 0.568462

CARKD F2R PHKG1 CDCP1 PLEKHM1 SMC2

0.173380 1.587816 0.500789 0.124248 4.061004 0.086447

PSMB6 BX440400 A\_24\_P936373 PPAN BC007917 C14orf143

-0.872610 -0.005739 0.528840 0.324097 -0.282504 0.198250

LOC440104 THC2578957 ANKIB1

-0.868232 0.437893 0.083391

1. p-values, residuals variance estimates and goodness-of-fit

> summary(fit)

Call:

lm(formula = health ~ ., data = training)

Residuals:

Min 1Q Median 3Q Max

-2.3069 -0.7844 -0.1422 0.5988 12.8654

Coefficients:

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

(Intercept) -3.403259 0.883576 -3.852 0.000147 \*\*\*

PCDH12 -0.112777 0.076233 -1.479 0.140209

DLG5 -0.743025 0.227193 -3.270 0.001214 \*\*

BC038559 -0.298175 0.367246 -0.812 0.417555

SHISA5 0.292498 0.323874 0.903 0.367269

AF161342 0.568462 0.149707 3.797 0.000181 \*\*\*

CARKD 0.173380 0.097949 1.770 0.077842 .

F2R 1.587816 0.470229 3.377 0.000842 \*\*\*

PHKG1 0.500789 0.237720 2.107 0.036075 \*

CDCP1 0.124248 0.180410 0.689 0.491607

PLEKHM1 4.061004 0.309838 13.107 < 2e-16 \*\*\*

SMC2 0.086447 0.041384 2.089 0.037657 \*

PSMB6 -0.872610 0.354422 -2.462 0.014441 \*

BX440400 -0.005739 0.265745 -0.022 0.982787

A\_24\_P936373 0.528840 0.206534 2.561 0.010997 \*

PPAN 0.324097 0.265618 1.220 0.223472

BC007917 -0.282504 0.307060 -0.920 0.358383

C14orf143 0.198250 0.112007 1.770 0.077864 .

LOC440104 -0.868232 0.309428 -2.806 0.005384 \*\*

THC2578957 0.437893 0.321298 1.363 0.174059

ANKIB1 0.083391 0.342849 0.243 0.808013

---

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

Residual standard error: 1.39 on 269 degrees of freedom

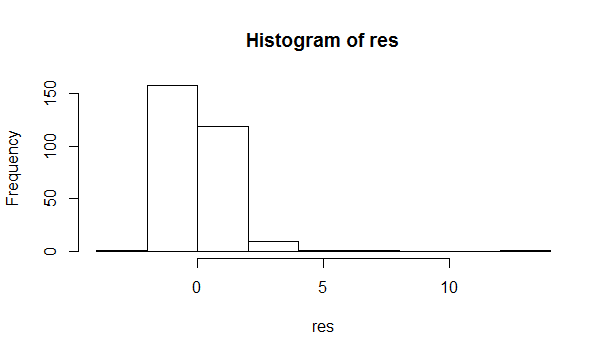
Multiple R-squared: 0.5611, Adjusted R-squared: 0.5284

F-statistic: 17.19 on 20 and 269 DF, p-value: < 2.2e-16

So the p-value should be 2.2e-16,the residual variance estimation is 1.39 on 269 degrees of freedom, goodness of fit is 0.5611.

Normal errors:

> res=fit$residuals

>hist(res)

From the picture above we could find that the residuals is not normally distributed.

### 3.Performing model diagnostics:

Shapiro-Wilk test:

> shapiro.test(fit$residuals)

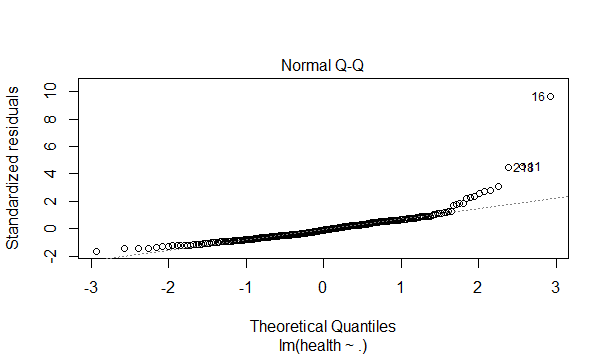
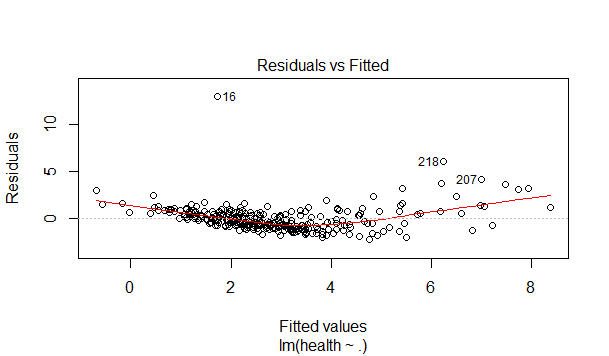
Shapiro-Wilk normality test

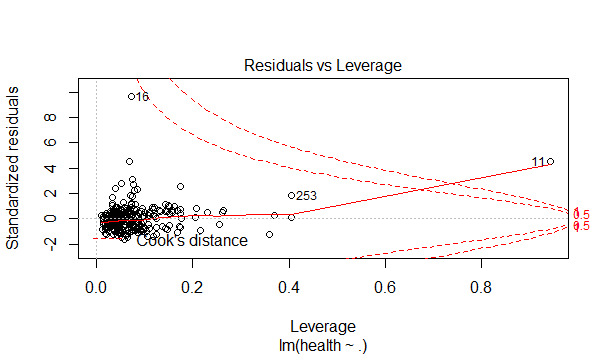
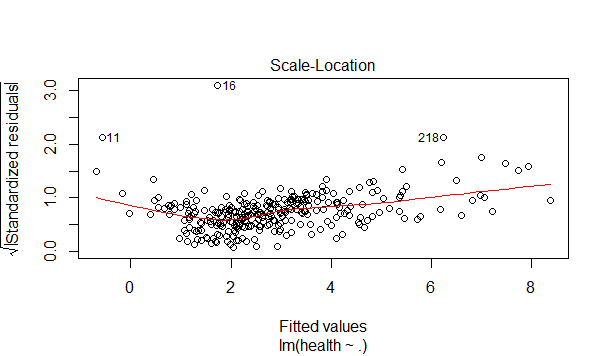
data: res

W = 0.7696, p-value < 2.2e-16

The hypothesis is that the residuals follow a normal distribution. As the p-value is very small, so we will reject the hypothesis. And the res doesn't follow a normal distribution.

We will use data visualization to analyze about it:

>plot(fit) 



From the four pictures above, we could get the conclusion that the variance is not a constant. So we will use BP test to analyze about the question:

> bptest(fit)

studentized Breusch-Pagan test

data: fit BP = 14.104, df = 21, p-value = 0.8651

As the p-value is very big, we need to accept the hypothesis. And it is possible that the variance is a constant. However with the data visualization, we found that the variance seems not a constant. BP-test doesn't work here. We will use the conclusion of data visualization and reject the conclusion of BP test and consider that the variance is not a constant.

Absense of influential cases: From the data visualization, case 16 is very influeial.

> which.max(cooks.distance(fit))

11 11

So I will consider both case 11 and 16 are influential cases. We are going to estimate our own variance.

Co-linearity:

> summary(fit)

Call:

lm(formula = health ~ ., data = training)

Residuals:

Min 1Q Median 3Q Max

-2.3069 -0.7844 -0.1422 0.5988 12.8654

Coefficients:

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

(Intercept) -3.403259 0.883576 -3.852 0.000147 \*\*\*

PCDH12 -0.112777 0.076233 -1.479 0.140209

DLG5 -0.743025 0.227193 -3.270 0.001214 \*\*

BC038559 -0.298175 0.367246 -0.812 0.417555

SHISA5 0.292498 0.323874 0.903 0.367269

AF161342 0.568462 0.149707 3.797 0.000181 \*\*\*

CARKD 0.173380 0.097949 1.770 0.077842 .

F2R 1.587816 0.470229 3.377 0.000842 \*\*\*

PHKG1 0.500789 0.237720 2.107 0.036075 \*

CDCP1 0.124248 0.180410 0.689 0.491607

PLEKHM1 4.061004 0.309838 13.107 < 2e-16 \*\*\*

SMC2 0.086447 0.041384 2.089 0.037657 \*

PSMB6 -0.872610 0.354422 -2.462 0.014441 \*

BX440400 -0.005739 0.265745 -0.022 0.982787

A\_24\_P936373 0.528840 0.206534 2.561 0.010997 \*

PPAN 0.324097 0.265618 1.220 0.223472

BC007917 -0.282504 0.307060 -0.920 0.358383

C14orf143 0.198250 0.112007 1.770 0.077864 .

LOC440104 -0.868232 0.309428 -2.806 0.005384 \*\*

THC2578957 0.437893 0.321298 1.363 0.174059

ANKIB1 0.083391 0.342849 0.243 0.808013

---

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

Residual standard error: 1.39 on 269 degrees of freedom

Multiple R-squared: 0.5611, Adjusted R-squared: 0.5284

F-statistic: 17.19 on 20 and 269 DF, p-value: < 2.2e-16

As we can see, R2 is 0.5611(Not very high), but there are still several parameters that are very significant.Thus we need to consider about XTX matrix and then using ridge regression to select a prefect lamda.

install.packages("PerformanceAnalytics")

library(PerformanceAnalytics)

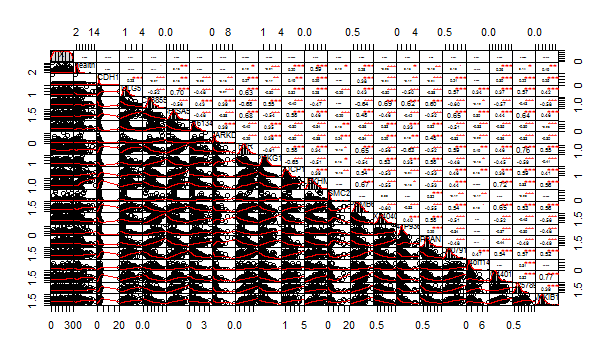
chart.Correlation(training)

ridge.fit=lm.ridge(health~.,data=training,lambda=seq(1,100,1)) plot(ridge.fit)

which.min(ridge.fit$GCV)

4

4



## Part B

Let us first print the hist for every predictors and see which one needs to be transformed.

> hist(trainingDLG5) > hist(trainingSHISA5) > hist(trainingCARKD) > hist(trainingPHKG1) > hist(trainingPLEKHM1) > hist(trainingPSMB6) > hist(trainingA\_24\_P936373) > hist(trainingBC007917) > hist(trainingLOC440104) > hist(trainingANKIB1)

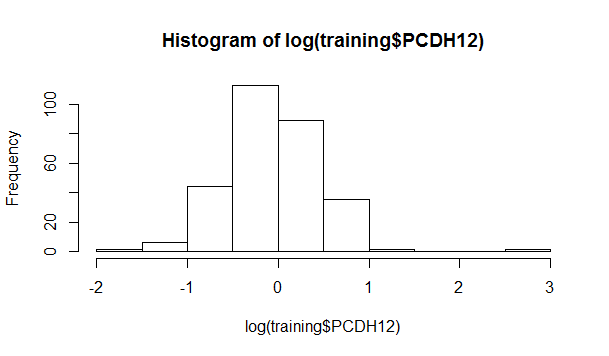
From the hist we find the following predictors need to be transformed: PCDH12, CARKD, SMC2, C14orf143

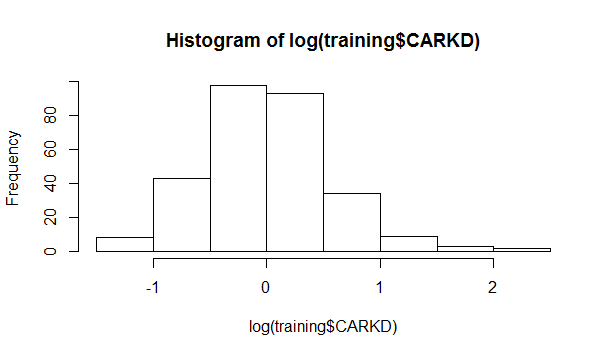
> hist(log(training$PCDH12))

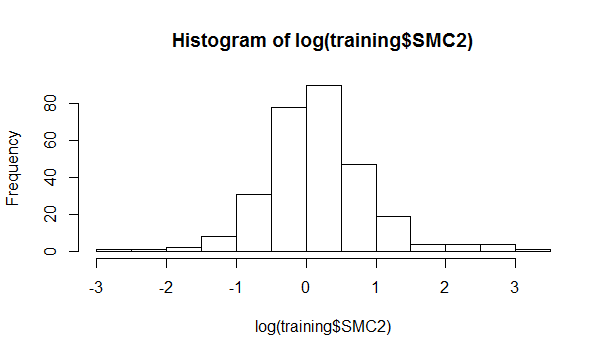
> hist(log(training$CARKD))

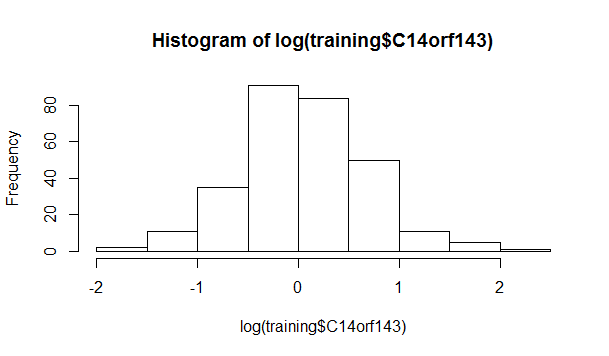
> hist(log(training$SMC2))

> hist(log(training$C14orf143))



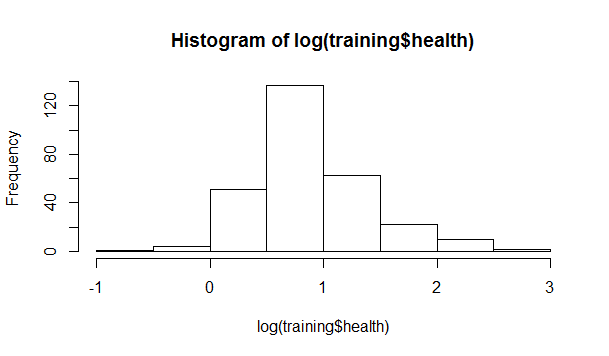






And we discovered that after the log transformation, the linearity of the terms improved a lot.

> hist(log(training$health))



And after we use log transform to deal with health, we found that the residuals of health become more normally distributed.

> training$health=log(training$health)

> training$PCDH12=log(training$PCDH12)

> training$CARKD=log(training$CARKD)

> training$SMC2=log(training$SMC2)

> training$C14orf143=log(training$C14orf143)

> fit2=lm(log(training$health)~log(training$PCDH12)+training$DLG5+training$BC038559+training$SHISA5+training$AF161342+log(training$CARKD)+training$F2R+training$PHKG1+training$CDCP1+training$PLEKHM1+log(training$SMC2)+training$PSMB6+training$BX440400+training$A\_24\_P936373+training$PPAN+training$BC007917+log(training$C14orf143)+training$LOC440104+training$THC2578957+training$ANKIB1)

> summary(fit2)

Call:

lm(formula = log(training$health) ~ log(training$PCDH12) + training$DLG5 +

training$BC038559 + training$SHISA5 + training$AF161342 +

log(training$CARKD) + training$F2R + training$PHKG1 + training$CDCP1 +

training$PLEKHM1 + log(training$SMC2) + training$PSMB6 +

training$BX440400 + training$A\_24\_P936373 + training$PPAN +

training$BC007917 + log(training$C14orf143) + training$LOC440104 +

training$THC2578957 + training$ANKIB1)

Residuals:

Min 1Q Median 3Q Max

-0.61574 -0.18653 -0.00818 0.17811 2.05751

Coefficients:

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

(Intercept) -1.388452 0.195675 -7.096 1.14e-11 \*\*\*

log(training$PCDH12) -0.496896 0.061049 -8.139 1.49e-14 \*\*\*

training$DLG5 -0.336299 0.049129 -6.845 5.16e-11 \*\*\*

training$BC038559 -0.097811 0.079715 -1.227 0.2209

training$SHISA5 0.107494 0.074632 1.440 0.1509

training$AF161342 0.186654 0.033235 5.616 4.86e-08 \*\*\*

log(training$CARKD) 0.107542 0.047010 2.288 0.0229 \*

training$F2R 0.642134 0.100895 6.364 8.41e-10 \*\*\*

training$PHKG1 0.251808 0.051914 4.850 2.09e-06 \*\*\*

training$CDCP1 0.060937 0.038811 1.570 0.1176

training$PLEKHM1 1.189143 0.069653 17.072 < 2e-16 \*\*\*

log(training$SMC2) 0.018753 0.039080 0.480 0.6317

training$PSMB6 -0.049543 0.078378 -0.632 0.5279

training$BX440400 -0.004515 0.056706 -0.080 0.9366

training$A\_24\_P936373 0.103161 0.044350 2.326 0.0208 \*

training$PPAN 0.113536 0.059194 1.918 0.0562 .

training$BC007917 -0.080980 0.066449 -1.219 0.2240

log(training$C14orf143) 0.067868 0.039660 1.711 0.0882 .

training$LOC440104 -0.043688 0.066894 -0.653 0.5143

training$THC2578957 0.105453 0.070072 1.505 0.1335

training$ANKIB1 -0.034680 0.072527 -0.478 0.6329

---

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

Residual standard error: 0.2965 on 269 degrees of freedom

Multiple R-squared: 0.6997, Adjusted R-squared: 0.6774

F-statistic: 31.34 on 20 and 269 DF, p-value: < 2.2e-16

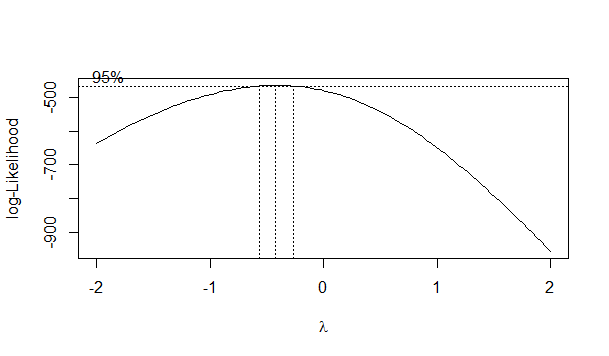
So after the log transformation the linearity increased.

So we need to implement box-cox here.

> fit=lm(health~.,data=training)

> library(MASS)

> bc=boxcox(fit,plotit=TRUE)



As lambda is not maximized at 0 we need further implementation onto it.

###############Consider about lambda################

>lambda=bcy)]

So we could use lambda to better the result.

From the former observation, we found that case 11 and case 16 got severe effect on the final result. So we will establish a new model without case 11 and case 16.

> trainingi=training[c(-11,-16),]

> fit3=lm(log(trainingi$health)~log(trainingi$PCDH12)+trainingi$DLG5+trainingi$BC038559+trainingi$SHISA5+trainingi$AF161342+log(trainingi$CARKD)+trainingi$F2R+trainingi$PHKG1+trainingi$CDCP1+trainingi$PLEKHM1+log(trainingi$SMC2)+trainingi$PSMB6+trainingi$BX440400+trainingi$A\_24\_P936373+trainingi$PPAN+trainingi$BC007917+log(trainingi$C14orf143)+trainingi$LOC440104+trainingi$THC2578957+trainingi$ANKIB1)

> summary(fit3)

Call:

lm(formula = log(trainingi$health) ~ log(trainingi$PCDH12) +

trainingi$DLG5 + trainingi$BC038559 + trainingi$SHISA5 +

trainingi$AF161342 + log(trainingi$CARKD) + trainingi$F2R +

trainingi$PHKG1 + trainingi$CDCP1 + trainingi$PLEKHM1 + log(trainingi$SMC2) +

trainingi$PSMB6 + trainingi$BX440400 + trainingi$A\_24\_P936373 +

trainingi$PPAN + trainingi$BC007917 + log(trainingi$C14orf143) +

trainingi$LOC440104 + trainingi$THC2578957 + trainingi$ANKIB1)

Residuals:

Min 1Q Median 3Q Max

-0.6737 -0.1613 -0.0144 0.1575 0.9611

Coefficients:

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

(Intercept) -1.525531 0.174316 -8.752 2.44e-16 \*\*\*

log(trainingi$PCDH12) -0.618506 0.065028 -9.511 < 2e-16 \*\*\*

trainingi$DLG5 -0.384016 0.043427 -8.843 < 2e-16 \*\*\*

trainingi$BC038559 -0.041342 0.070867 -0.583 0.56013

trainingi$SHISA5 0.202741 0.066468 3.050 0.00252 \*\*

trainingi$AF161342 0.167018 0.029447 5.672 3.67e-08 \*\*\*

log(trainingi$CARKD) 0.136277 0.042076 3.239 0.00135 \*\*

trainingi$F2R 0.598738 0.088686 6.751 9.11e-11 \*\*\*

trainingi$PHKG1 0.254664 0.045944 5.543 7.13e-08 \*\*\*

trainingi$CDCP1 0.028096 0.034680 0.810 0.41857

trainingi$PLEKHM1 1.227947 0.062898 19.523 < 2e-16 \*\*\*

log(trainingi$SMC2) 0.022245 0.034365 0.647 0.51798

trainingi$PSMB6 0.030969 0.069743 0.444 0.65737

trainingi$BX440400 -0.026219 0.049820 -0.526 0.59913

trainingi$A\_24\_P936373 0.091352 0.038994 2.343 0.01988 \*

trainingi$PPAN 0.095739 0.051998 1.841 0.06670 .

trainingi$BC007917 -0.086709 0.058605 -1.480 0.14017

log(trainingi$C14orf143) 0.063062 0.034807 1.812 0.07115 .

trainingi$LOC440104 -0.035224 0.059396 -0.593 0.55366

trainingi$THC2578957 0.118970 0.061609 1.931 0.05453 .

trainingi$ANKIB1 -0.003422 0.063734 -0.054 0.95722

---

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

Residual standard error: 0.2601 on 267 degrees of freedom

Multiple R-squared: 0.7577, Adjusted R-squared: 0.7395

F-statistic: 41.74 on 20 and 267 DF, p-value: < 2.2e-16

From the R-square we found that some of the model without the effect of case 11 and case 16 is really better. And some predictors are much severe than the others. So we will use AIC algorithm to better improve the model.

> step(fit3,direction = "backward")

From the AIC algorithm we could obtain the final fitting model:

> fit\_final=lm(log(trainingi$health) ~ log(trainingi$PCDH12) + + trainingi$DLG5 + trainingi$SHISA5 + trainingi$AF161342 + + log(trainingi$CARKD) + trainingi$F2R + trainingi$PHKG1 + + trainingi$PLEKHM1 + trainingi$A\_24\_P936373 + trainingi$PPAN + + + trainingi$THC2578957)> summary(fit5)

> fitted\_values=(((fit\_final$fitted.values)\*lambda)+1)^(1/lambda)

> write.csv(as.data.frame(fitted\_values),"SUBMITTED VALUE420.csv")

Now it is finished! Please check my submitted to see my final result.