ADA HW5

Liangquan Zhou(lz2377)

10/09/2014

Consider the `Pima.te` dataset, in R library `MASS`, on Diabetes in Pima Indian Women.

a). Fit a multiple linear regression model of predict `glu`, plasma glucose concentration in an oral glucose tolerance test, using the following set of predictors:

`npreg` number of pregnancies

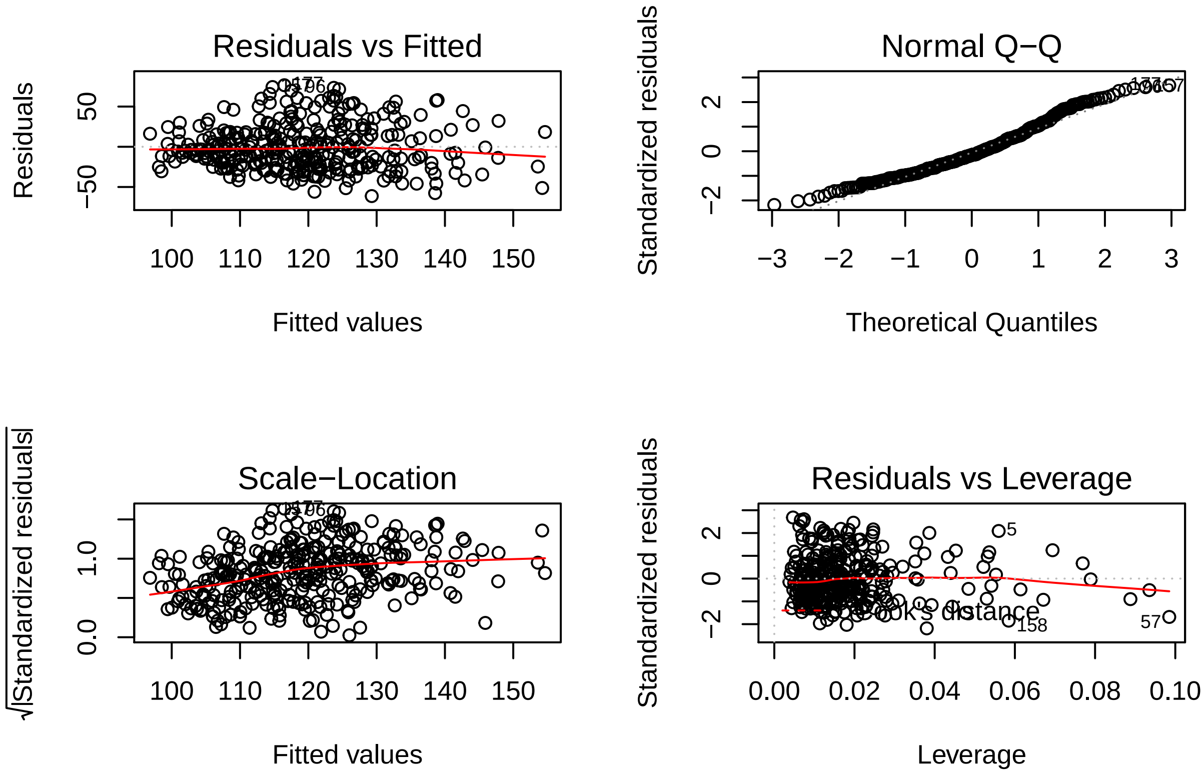
`bp` diastolic blood pressure (mm Hg)

`skin` triceps skin fold thickness (mm)

`bmi` body mass index (weight in kg/(height in m)^2)

`age` age in years

```
library(MASS)
library(lmtest)
fit = lm(glu ~ npreg + bp + skin + bmi + age, data = Pima.te)
par(mfrow=c(2,2))
plot(fit)
```



```r
summary(fit)
```

```
##
## Call:
## lm(formula = glu ~ npreg + bp + skin + bmi + age, data = Pima.te)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -61.29 -20.56  -4.36  17.37  76.51
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   56.831     10.309    5.51  7.2e-08 ***
## npreg         -0.875      0.647   -1.35  0.17735
## bp             0.104      0.138    0.75  0.45353
## skin           0.263      0.216    1.21  0.22575
## bmi            0.796      0.302    2.64  0.00880 **
## age            0.764      0.207    3.69  0.00026 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 28.6 on 326 degrees of freedom
## Multiple R-squared:  0.134,  Adjusted R-squared:  0.121
## F-statistic: 10.1 on 5 and 326 DF,  p-value: 5.58e-09
```

b). State and assess the validity of the underlying assumptions:

Linearity/functional form, including the need for any interaction terms

Compute R^2 to see check the functional form.

```r
summary(fit)$r.squared
```

```
## [1] 0.1338
```

Since R^2 is small, it suggest lack of fit. So we should consider high order and interaction terms.

Normality

Apply Shapiro-Wilk test on the residuals.

```r
shapiro.test(fit$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  fit$residuals
## W = 0.9703, p-value = 2.532e-06
```

Since $p-value < 0.05$, we reject H_0, and think the residuals is not normally distributed

Homoscedasticity

Apply Breusch-Pagan Test to test homoscedasticity.

```r
bptest(fit)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  fit
## BP = 19.58, df = 5, p-value = 0.0015
```

From the test we can see that $p-value$ is less than 0.05, then we reject H_0, and think the homoscedasticity is not valid.

Uncorrelated error

Apply Durbin-Watson test for $1st$ order AR.

```r
dwtest(fit)
```

```
##
##  Durbin-Watson test
##
## data:  fit
## DW = 1.938, p-value = 0.2847
## alternative hypothesis: true autocorrelation is greater than 0
```

The $p - value$ is greater than 0.05, so we accept H_0, and think the uncorrelated error is valid.

Check for outliers and influential points.

Use Studentized deleted residuals to identiry Y-outliers, and cooks distance for influential points.

```
n = 332
p = 6
lmi = lm.influence(fit)
lms = summary(fit)
e <- resid(fit)
s <-lms$sigma
si <-lmi$sigma
xxi <-diag(lms$cov.unscaled)
h <-lmi$hat
bi <- coef(fit)-t(coef(lmi))
dfbetas <- bi/t(si%o%xxi^0.5)
stand.resid <- e/(s*(1-h)^0.5)
student.resid <- e/(si*(1-h)^0.5)
DFFITS <- h^0.5*e/(si*(1-h))
```

Then we use `studentized deleted residuals` to check Y-outliers. Using a Bonferroni test procedure to test whether the largest absolute studentized deleted residual is an outlier.

```
## check outliers
# p = 6
# n = 332
abs(max(student.resid)) > qt(0.05/(2*n), n-p-2)
```

```
## [1] TRUE
```

```
library(car)
outlierTest(fit)
```

```
##
## No Studentized residuals with Bonferonni p < 0.05
## Largest |rstudent|:
##     rstudent unadjusted p-value Bonferonni p
## 177    2.707            0.00715           NA
```

So the Y-outlier exists. And the 177th point is a Y-outlier.

Using diagonal of the hat matrix to check X-outliers

```r
## check outliers
h[h > 2* p / n]
```

```
##       5       8      12      18      21      41      43      57      72
## 0.05596 0.06709 0.03924 0.04439 0.05324 0.05299 0.05215 0.09850 0.06940
##      79      92     107     141     158     196     198     203     211
## 0.08882 0.04399 0.03743 0.04528 0.05845 0.03867 0.05361 0.09355 0.05414
##     217     232     249     262     287     291     292     320     330
## 0.07691 0.05529 0.03623 0.04843 0.07892 0.03803 0.06140 0.04329 0.04781
```

For influential points points, first use `cook's distance` to test influence on all fitted values.

```r
## use cook's distance(a aggregrate measure of influence)
which(cooks.distance(fit)>qf(0.05, p+1, n-p-1))
```

```
## named integer(0)
```

It shows there is no influential points.

But if we use `DFFITS`, since $n = 332$, we can think this is kind of a large dataset. `DFFITS` test influuence on Single Fitted Values, and we can find there are some influential points:

```r
DFFITS[which(abs(DFFITS) > 2*sqrt((p+1)/n))]
```

```
##       5      57      72      86     101     106     158     196     243
##  0.5114 -0.5582  0.3402  0.3240  0.2983  0.3453 -0.4617  0.4036  0.3034
##     281     291     305     330
##  0.2963 -0.4369  0.3509 -0.3369
```

And use `DFBETAS` to test Influence on the Regression Coefficients. Since large values of $DFBETAS_{k(i)}$ indicate the influence of the ith case on the kth regression coefficient estimate, and we have $p = 6$ coefficients, so for every coefficient we should check whether the the point is influential.

```r
which(dfbetas(fit)[,1] > 2 / sqrt(n)) # influtial points on 1st coefficient (Intercept)
```

```
##  45  79 153 157 158 166 172 281 291 305
##  45  79 153 157 158 166 172 281 291 305
```

```r
which(dfbetas(fit)[,2] > 2 / sqrt(n)) # influtial points on 2nd coefficient (npreg)
```

```
##  12  18  72  78 106 141 158 166 192 219 242 305 320
##  12  18  72  78 106 141 158 166 192 219 242 305 320
```

```r
which(dfbetas(fit)[,3] > 2 / sqrt(n)) # influtial points on 3rd coefficient (bp)
```

```
##   8  57 101 141 184 196 242
##   8  57 101 141 184 196 242
```

```
which(dfbetas(fit)[,4] > 2 / sqrt(n)) # influtial points on 4th coefficient (skin)
```

```
##   5  78  96 162 180 265 291 306
##   5  78  96 162 180 265 291 306
```

```
which(dfbetas(fit)[,5] > 2 / sqrt(n)) # influtial points on 5th coefficient (bmi)
```

```
##  21  86 101 107 198 233 243 284 293 323 329 330
##  21  86 101 107 198 233 243 284 293 323 329 330
```

```
which(dfbetas(fit)[,6] > 2 / sqrt(n)) # influtial points on 6th coefficient (age)
```

```
##   5   6  86 100 191 196 217 243 257 268 290
##   5   6  86 100 191 196 217 243 257 268 290
```

c). Propose remedial measures in case of violations of any of the underlying assumptions

Lack of fit

Apply Simple transformations, e.g., log

Use Non-linear model

Use Other predictors

Non-normality

Transformation

Use Robust regression methods

Non-constancy of the Error Variance

Use Transformation

Use Build variance structure into model: Weighted Least Square

Outliers and Influential Points

Use Robust regression methods, e.g., LAD regression, LMS regression.