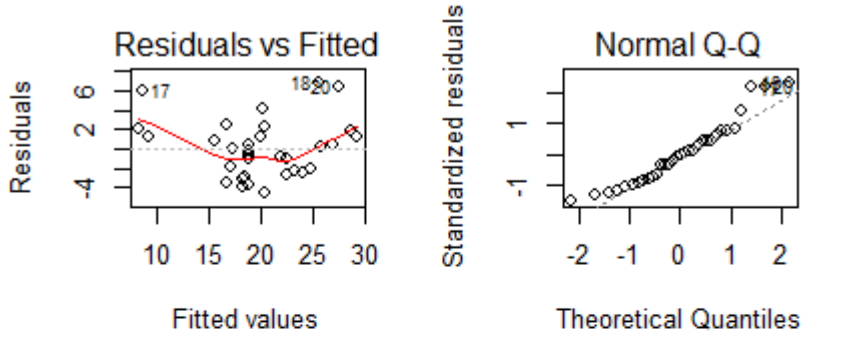
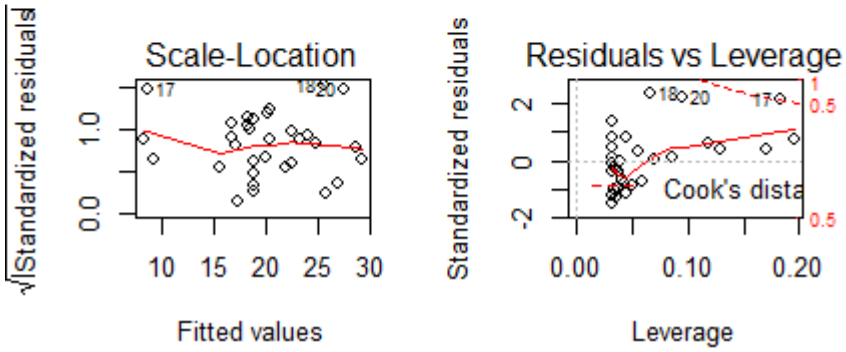
R Lab 5. REGRESSION DIAGNOSTICS

 # Run regression and look at residual plots. Split the plot window to get all the plots at once.

 # STUDENTIZED RESIDUALS AND OUTLIERS

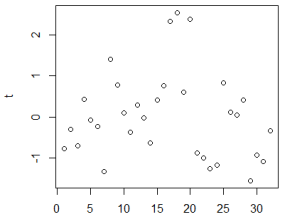
```
> attach(mtcars)
> reg = lm( mpg ~ wt )
> plot(wt,mpg)
> abline(reg)

> par(mfrow=c(2,2))
> plot(reg)     # This command gives residual plots, and without "par", it gives them one at a time
```





 # Studentized residuals and testing for outliers

```
> t = rstudent(reg)          # Studentized residuals
> par(mfrow=c(1,1))          # Return to the 1x1 plot window
> plot(t)                    # See if there are any nonlinear trends
```




```
> t[ abs(t)>2 ]
      17       18       20
2.328162 2.537801 2.383844
```

 # Which of these residuals can be considered as outliers?
 # Compare with the Bonferroni-adjusted quantile from t-distribution.

```
> n = length(wt)
> qt( 0.025/n, n-2 )
[1] -3.478736

> t[ abs(t) > abs(qt( 0.025/n, n-2 )) ]
named numeric(0)
```

 # The test revealed no outliers.

 # Testing NORMALITY

```
> shapiro.test(t)

        Shapiro-Wilk normality test

data:  t
W = 0.92916, p-value = 0.03711
```

 # Rather marginal. Also look at the Normal Q-Q plot among residual plots above. It is not straight, so the data may be non-Normal. Shapiro-Wilk statistic W measures how close the graph is to a straight line.

 # Testing HOMOSCEDASTICITY (constant variance).
Look at the residual plots including a plot of t^2 vs fitted values.
Below is the Breausch-Pagan test, and it is included in R package called "car".

```
> install.packages("car")
> library("car")
> ncvTest(reg)
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 0.03794177, Df = 1, p = 0.84556
```

 # This package also has a built-in outlier test

```
> outlierTest(reg)

No Studentized residuals with Bonferonni p < 0.05
Largest |rstudent|:
   rstudent unadjusted p-value Bonferonni p
18 2.537801           0.016788       0.5372
```

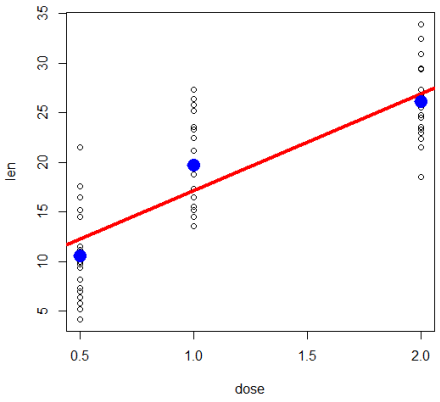
 # LACK OF FIT TEST

 # The ToothGrowth dataset has only 3 different values of X = dose

```
> attach(ToothGrowth)
> names(ToothGrowth)
[1] "len"  "supp" "dose"
> table(dose)
dose
0.5   1   2
 20  20  20
```

 # Fit two regression models:
 # reduced = simple linear regression predicting Y = length in terms of X = dose
 # full = using group means to predict Y for each value of X, thus treating X as a categorical variable

```
> reduced = lm(len ~ dose)
> full = lm(len ~ as.factor(dose))

> plot(dose,len)
> abline(reduced,col="red",lwd=4)
> points( dose, predict(full), col="blue", lwd=10 )
```



 # Here is the rigorous F-test for the lack of fit

```
> anova(reduced,full)
Analysis of Variance Table

Model 1: len ~ dose
Model 2: len ~ as.factor(dose)
  Res.Df    RSS Df Sum of Sq      F    Pr(>F)
```

```
1     58 1227.9
2     57 1025.8  1    202.13 11.232 0.001432 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

 # Conclusion: the difference in SSreg is significant; the linear regression model does have a lack of fit.

Box-Cox Transformation

 # Find the best power transformation of responses that fixes non-normality.

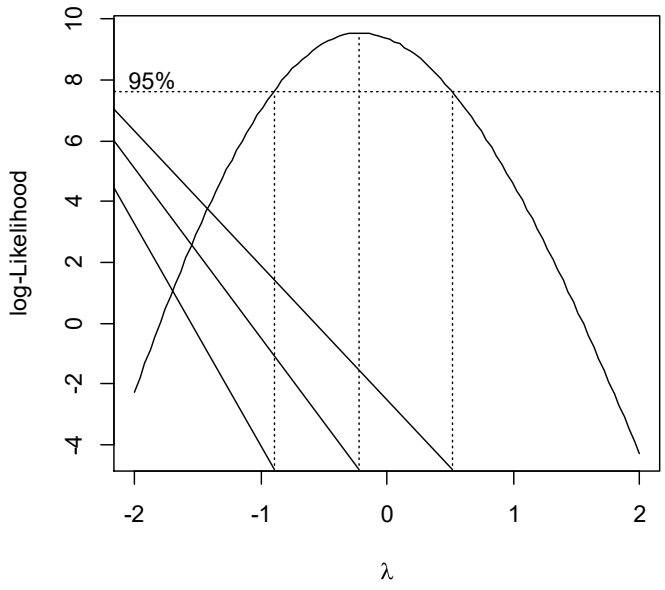
```
> attach(mtcars)
```

 # Fit a linear regression model, save studentized residuals, and test their Normal distribution

```
> reg = lm( mpg ~ wt )
> t = rstudent(reg)
> shapiro.test(t)

        Shapiro-Wilk normality test

data:  t
W = 0.92916, p-value = 0.03711
```

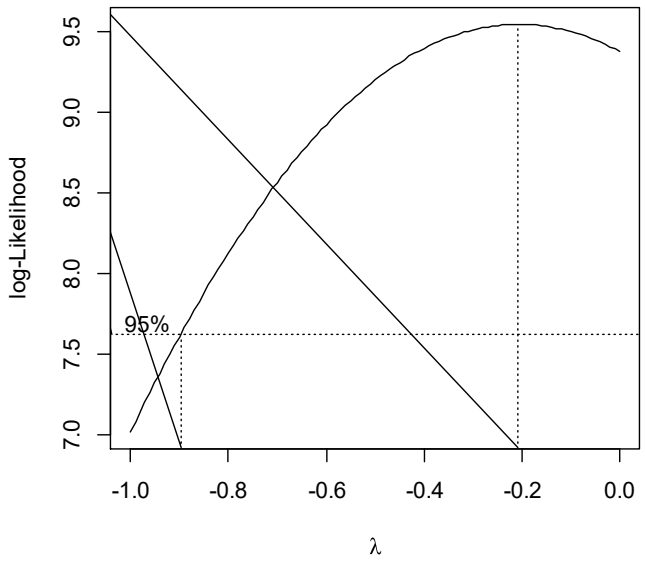
 # Results are marginal, so let’s look for the best transformation. Box-Cox is available in package “MASS”

```
> install.packages("MASS")
> library(MASS)

> boxcox(reg)
```



 # Following the maximum likelihood principle, we are looking for the value of lambda that maximizes the likelihood function, or equivalently, the logarithm of this likelihood, which is on the graph. We see that the optimal lambda is somewhere between -1 and 0. We can limit the search to this range. The line `seq(-1,0,0.01)` **means all values of lambda from -1 to 0 with the step of 0.01.**

```
> boxcox(reg,lambda=seq(-1,0,0.01))
```



 # Now we can see that the best lambda is very close to -0.2. Let's introduce a variable that is the corresponding power transform of our response Y, fir this new regression, and check residuals for Normality.

```
> Z = mpg^(-2)
> newreg = lm(Z ~ wt)
> t = rstudent(newreg)
> shapiro.test(t)

        Shapiro-Wilk normality test

data:  t
W = 0.97259, p-value = 0.5736

> Z = mpg^(-0.2)
> newreg = lm(Z ~ wt)
> t = rstudent(newreg)
> shapiro.test(t)

        Shapiro-Wilk normality test

data:  t
W = 0.95892, p-value = 0.2566
```

 # This regression model passes the Shapiro-Wilk test for Normality; the p-value is high. No evidence that this assumption is violated.