1 Model Validation

Model validation is possibly the most important step in the model building sequence. It is also one of the most overlooked. Often the validation of a model seems to consist of nothing more than quoting the R^2 statistic from the fit (which measures the fraction of the total variability in the response that is accounted for by the model).

Unfortunately, a high R^2 value does not guarantee that the model fits the data well. Use of a model that does not fit the data well cannot provide good answers to the underlying engineering or scientific questions under investigation.

Model diagnostic techniques determine whether or not the distributional assumptions are satisfied, and to assess the influence of unusual observations.

1.1 Why Use Residuals?

If the model fit to the data were correct, the residuals would approximate the random errors that make the relationship between the explanatory variables and the response variable a statistical relationship. Therefore, if the residuals appear to behave randomly, it suggests that the model fits the data well. On the other hand, if non-random structure is evident in the residuals, it is a clear sign that the model fits the data poorly.

The subsections listed below detail the types of plots to use to test different aspects of a model and give guidance on the correct interpretations of different results that could be observed for each type of plot.

1.2 Checking Assumptions in ANOVA and Linear Regression Models

- The assumptions of normality and homogeneity of variance for linear models are not about Y, the dependent variable.
- The distributional assumptions for linear regression and ANOVA are for the distribution of Y—X (Y given X).
- You have to take out the effects of all the Xs before you look at the distribution of Y. As it turns out, the distribution of Y—X is, by definition, the same as the distribution of the residuals. So the easiest way to check the distribution of Y—X is to save your residuals and check their distribution.

What are those distributional assumptions of Y—X?

1. Independence
2. Normality
3. Constant Variance

These assumptions can be checked with a few residual plots a Q-Q plot of the residuals for normality and a scatterplot of Residuals on X or Predicted values of Y to check 1 and 3.

2 Simple Linear Regression

In simple linear regression, we predict values on one variable from the values of a second variable.

- The variable we are predicting is called the ***dependent variable*** (or response variable) and is referred to as Y.
- The variable we are basing our predictions on is called the **i*ndependent variable*** (or predictor variable) and is referred to as X.

Remark: When there is only one predictor variable, the prediction method is called simple regression. Linear regression can have more than one predictor variable, i.e. Multiple Linear Regression.

In simple linear regression, the predicted values of Y when plotted as a function of X form a straight line on the scatter plot. This line is known as the ***regression line***.

- Suppose we construct our model using n observed values of the response variable $y_1, y_2, \dots y_i \dots y_n$
- For the original data set, there is a predicted value of each case of Y that corresponds to an observed value of Y.
- The difference between an observed value of the dependent variable (y_i) and the corresponding predicted value $(\hat{y})$ is called the residual (e_i). Each data point from the data set has one residual.
- Simply put, the values of the residuals are derived as follows:

$$\text{Residual} = \text{Observed value - Predicted value}$$

$$e_i = y_i - \hat{y}_i$$

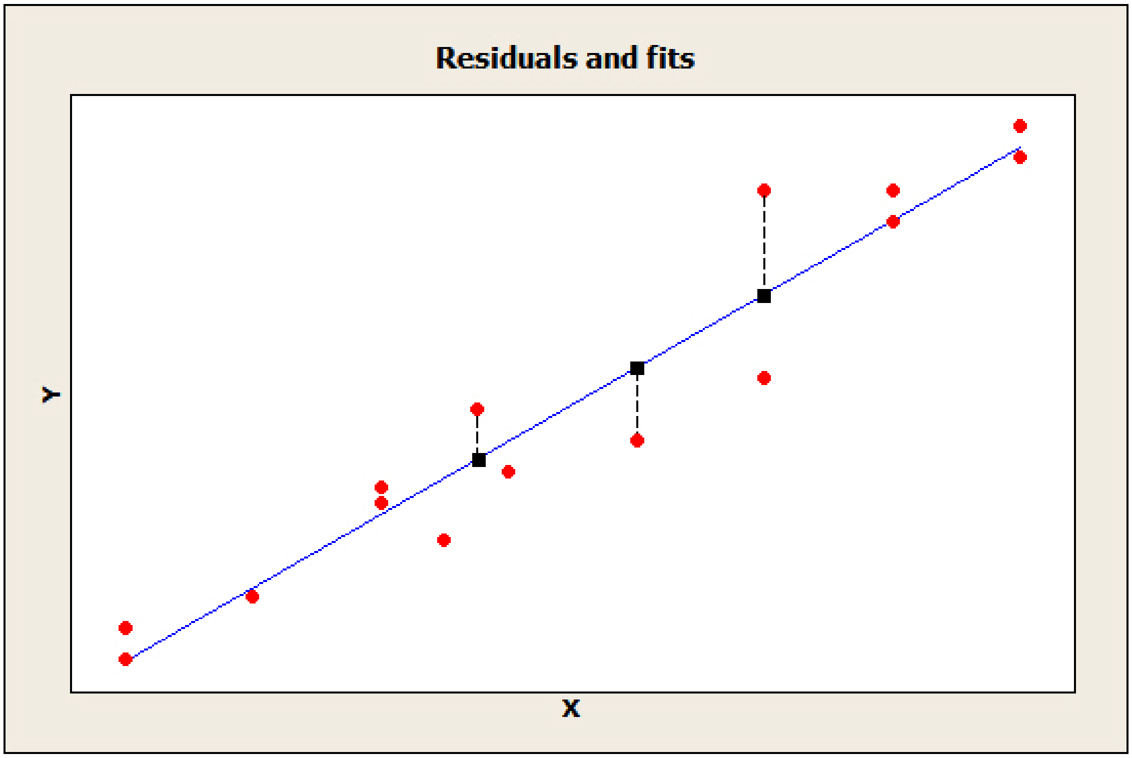


Figure 1:

- For three cases in the graphic above, the observed value (red dot) is linked to its corresponding predicted value (black dot) on the regression line (blue line). The difference (i.e. residual) is depicted using a dashed line. The magnitude of these residuals is of interest.
- The second of the three residuals will have a negative value.
- ***Ordinary Least Squares*** is a method of fitting a model, such that the total residual values are minimised.
- Important theoretical assumption underlying the OLS model: the sum of the residuals should equal to zero.

$$\sum e_i = 0$$

- An extension of this is that the expected value of the residuals is 0. $\mathrm{E}(e) = 0$
- Another Important Theoretical Assumption - The residuals are normally distributed. (more on that later)

2.1 Residual Plots

A residual plot is a graph that shows the residuals on the vertical axis and the independent variable on the horizontal axis. If the points in a residual plot are randomly dispersed around the horizontal axis, a linear regression model is appropriate for the data; otherwise, a non-linear model is more appropriate.

Summary of Important Terms

Some important terms in model diagnostics, essentially a plan for this talk.

Residual: The difference between the predicted value (based on the regression equation) and the actual, observed value.

Outlier: In linear regression, an outlier is an observation with large residual. In other words, it is an observation whose dependent-variable value is unusual given its value on the predictor variables. An outlier may indicate a sample peculiarity or may indicate a data entry error or other problem.

Leverage: An observation with an extreme value on a predictor variable is a point with high leverage. Leverage is a measure of how far an independent variable deviates from its mean. High leverage points can have a great amount of effect on the estimate of regression coefficients.

Influence: An observation is said to be influential if removing the observation substantially changes the estimate of the regression coefficients. Influence can be thought of as the product of leverage and outlierness.

Cook's distance (or Cook's D): A measure that combines the information of leverage and residual of the observation.

MultiCollinearity

An importan aspect in model diagnostics is checking for multicollinearity. We are not going to cover this in this talk - but rather include in n a talk about variable selection procedure.

3 mtcars example

*Several data sets , intended as learning tools, are automatically installed when **R** is installed. Many more are installed within packages to complement learning to use those packages. One of these is the famous **mtcars** data set, which is used in many data mining exercises.*

```
> data(mtcars)
> head(mtcars)
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

Suppose we fit a model with *mpg* (miles per gallon) as the response variable and *cyl* and *wt* (number of cylinders and weight of the car) as the predictor variables. We will call this fitted model `fit`.

```
fit <- lm(mpg ~ cyl + wt, data=mtcars)
```

```
> summary(fit)

Call:
lm(formula = mpg ˜ cyl + wt, data = mtcars)

Residuals:
Min      1Q  Median      3Q     Max
-4.2893 -1.5512 -0.4684  1.5743  6.1004

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept)  39.6863     1.7150  23.141  < 2e-16 ***
cyl          -1.5078     0.4147  -3.636 0.001064 **
wt           -3.1910     0.7569  -4.216 0.000222 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 2.568 on 29 degrees of freedom
Multiple R-squared:  0.8302,    Adjusted R-squared:  0.8185
F-statistic: 70.91 on 2 and 29 DF,  p-value: 6.809e-12
```

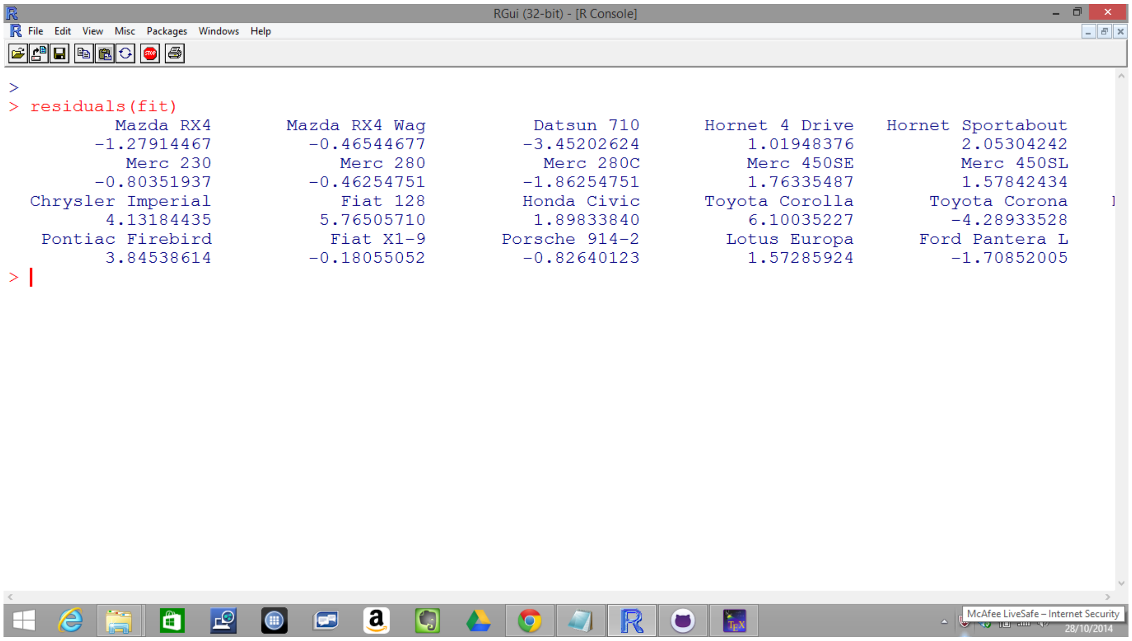
3.1 `resid` - Extracting Model Residuals

- `residuals` is a generic function which extracts model residuals from objects returned by modeling functions.
- The abbreviated form `resid` is an alias for residuals. It is intended to encourage users to access object components through an accessor function rather than by directly referencing an object slot.
- All object classes which are returned by model fitting functions should provide a residuals method. (Note that the method is for `residuals` and not `resid`.)
- Methods can make use of `naresid` methods to compensate for the omission of missing values. The default, nls and smooth.spline methods do.

```
residuals(fit)

resid(fit)
```

```
residuals(fit1)
```

```
> sum(residuals(fit))
[1] 1.096345e-15

> #Shapiro-Wilk Test for Normality
> shapiro.test(resid(fit))

 Shapiro-Wilk normality test
```


RGui (32-bit) - [R Console]
File
Edit
View
Misc
Packages
Windows
Help
McAfee LiveSafe – Internet Security
28/10/2014

```
>
> residuals(fit)
          Mazda RX4       Mazda RX4 Wag          Datsun 710      Hornet 4 Drive   Hornet Sportabout
        -1.27914467         -0.46544677         -3.45202624          1.01948376          2.05304242
           Merc 230            Merc 280           Merc 280C          Merc 450SE          Merc 450SL
        -0.80351937         -0.46254751         -1.86254751          1.76335487          1.57842434
  Chrysler Imperial            Fiat 128         Honda Civic      Toyota Corolla       Toyota Corona
         4.13184435          5.76505710          1.89833840          6.10035227         -4.28933528
   Pontiac Firebird           Fiat X1-9       Porsche 914-2        Lotus Europa      Ford Pantera L
         3.84538614         -0.18055052         -0.82640123          1.57285924         -1.70852005
> 
```

Figure 2:

```
data:  resid(fit)
W = 0.9375, p-value = 0.06341
```

4 Standardization and Studentization

4.1 Standardization

A random variable is said to be standardized if the difference from its mean is scaled by its standard deviation. The residuals above have mean zero but their variance is unknown, it depends on the true values of θ. Standardization is thus not possible in practice.

4.2 Studentization

Instead, you can compute studentized residuals by dividing a residual by an estimate of its standard deviation.

4.3 Internal and External Studentization

If that estimate is independent of the $i-$th observation, the process is termed *external studentization*'external studentization'. This is usually accomplished by excluding the $i-$th observation when computing the estimate of its standard error. If the observation contributes to the standard error computation, the residual is said to be *internally studentization*internally studentized.

5 Standardized and Studentized Residuals

The standardized residual is the residual divided by its standard deviation.

Plot the standardized residual of the simple linear regression model of the data set faithful against the independent variable waiting.

We apply the lm function to a formula that describes the variable eruptions by the variable waiting, and save the linear regression model in a new variable eruption.lm. Then we compute the standardized residual with the rstandard function.

```
eruption.lm = lm(eruptions ~ waiting, data=faithful)
eruption.stdres = rstandard(eruption.lm)
```

We now plot the standardized residual against the observed values of the variable waiting.

```
plot(faithful$waiting, eruption.stdres,
    ylab="Standardized Residuals",
     xlab="Waiting Time",
     main="Old Faithful Eruptions")
abline(0, 0)                  # the horizon
```

6 Assumption of Constant Variance

Homoscedasticity

- ***Homoscedascity*** is the technical term to describe the variance of the residuals being constant across the range of predicted values.

- ***Heteroscedascity*** is the converse scenario : the variance differs along the range of values.

Suppose you plot the individual residuals against the predicted value, the variance of the residuals predicted value should be constant. Consider the

red arrows in the picture below, intended to indicate the variance of the residuals at that part of the number line. For the OLS summption to be valid , the length of the red lines should be more or less the same.

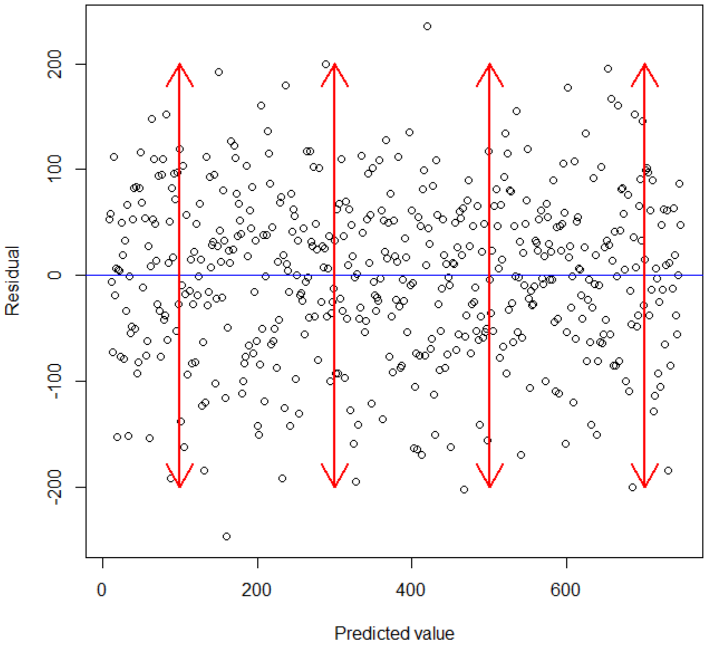


Figure 3:

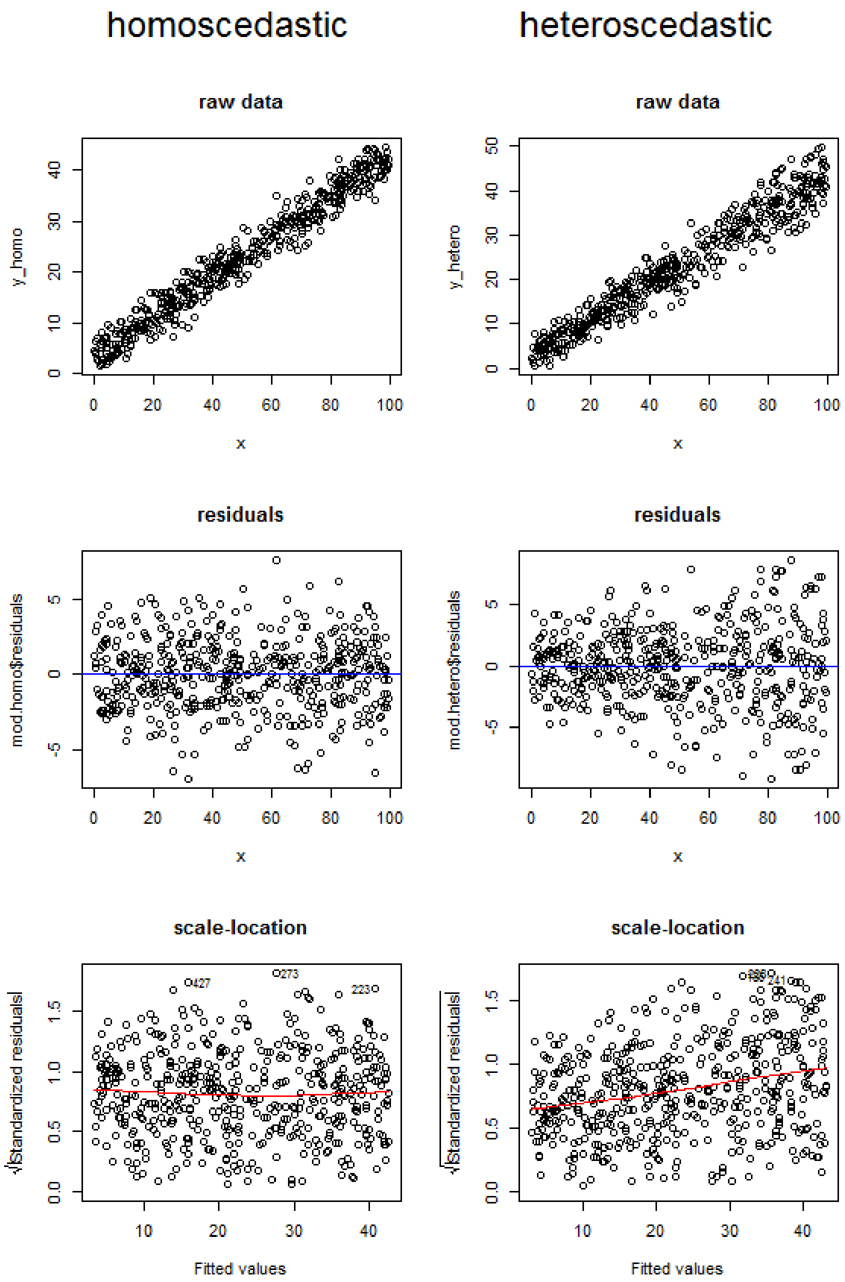

homoscedastic
heteroscedastic
raw data
y_homo
0
10
20
30
40
0
20
40
60
80
100
x
raw data
y_hetero
0
10
20
30
40
50
0
20
40
60
80
100
x
residuals
mod.homo$residuals
-5
0
5
0
20
40
60
80
100
x
residuals
mod.hetero$residuals
-5
0
5
0
20
40
60
80
100
x
scale-location
√|Standardized residuals|
0.0
0.5
1.0
1.5
427
273
223
10
20
30
40
Fitted values
scale-location
√|Standardized residuals|
0.0
0.5
1.0
1.5
241
10
20
30
40
Fitted values

Figure 4:

7 Diagnostic Plots for Linear Models with R

There are six plots (selectable by `which`) are currently available:

1. a plot of residuals against fitted values,
2. a Scale-Location plot of *sqrt(* | *residuals* |) against fitted values,
3. a Normal Q-Q plot,
4. a plot of Cook's distances versus row labels,
5. a plot of residuals against leverages,
6. a plot of Cook's distances against *leverage/(1-leverage).*

By default, the first three and 5 are provided, if you just type something like `plot(fit)`.

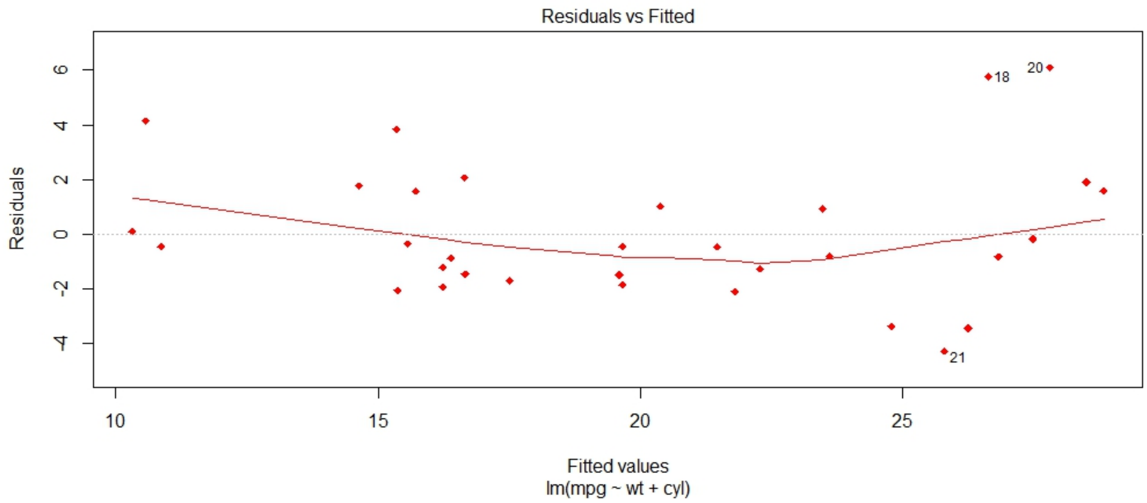
```
plot(lm(mpg~wt+cyl),which=c(1),pch=18,col="red")
plot(lm(mpg~wt+cyl),which=c(2),pch=18,col="red")
plot(lm(mpg~wt+cyl),which=c(3),pch=18,col="red")
plot(lm(mpg~wt+cyl),which=c(4),pch=18,col="red")
plot(lm(mpg~wt+cyl),which=c(5),pch=18,col="red")
plot(lm(mpg~wt+cyl),which=c(6),pch=18,col="red")
```

- **Plot 2** - The **Scale-Location** plot, also called Spread-Location (or S-L plot), takes the square root of the absolute residuals in order to diminish skewness (sqrt($|E|$)) is much less skewed than $|E|$ for Gaussian zero-mean E).
- **Plot 5** - The **Residual-Leverage** plot shows contours of equal Cook's distance, for values of `cook.levels` (by default 0.5 and 1) and omits cases with leverage one with a warning. If the leverages are constant (as is typically the case in a balanced aov situation) the plot uses factor level combinations instead of the leverages for the x-axis.
  (The factor levels are ordered by mean fitted value.)

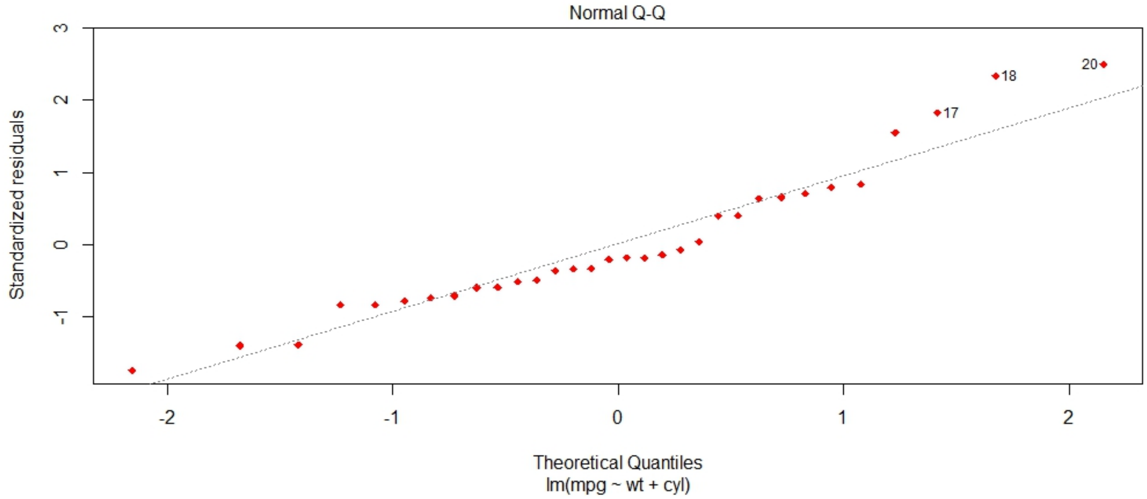
EDIT NOTE - FOLLOWING IN WRONG ORDER

Plot 1 : Residual Plot

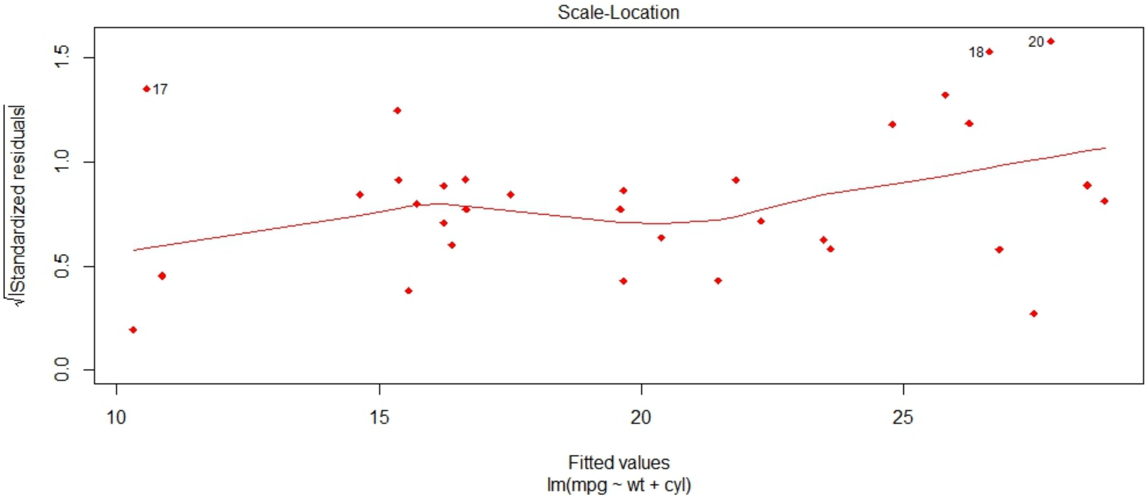
Test for Constant Variance



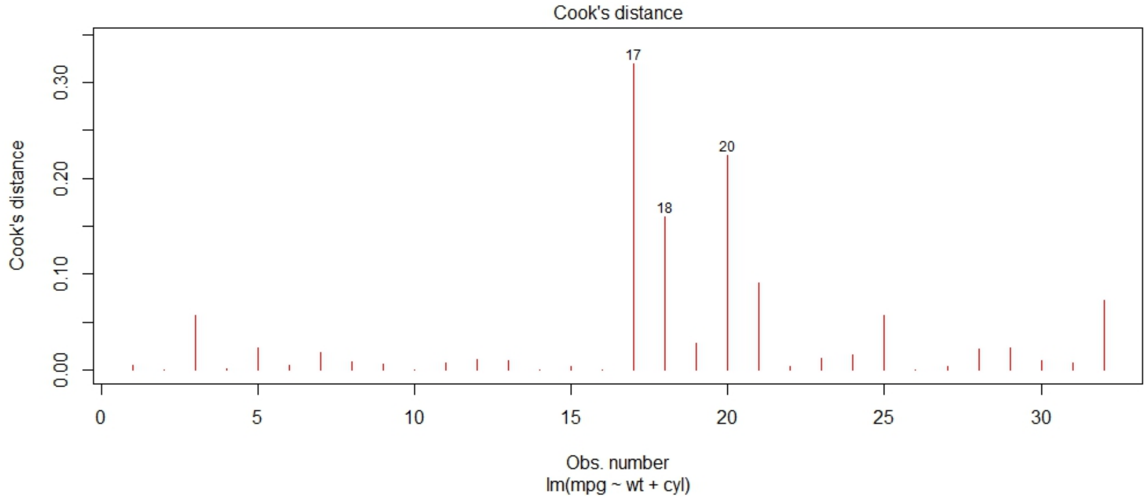
Plot 3 : Normal Probability Plot

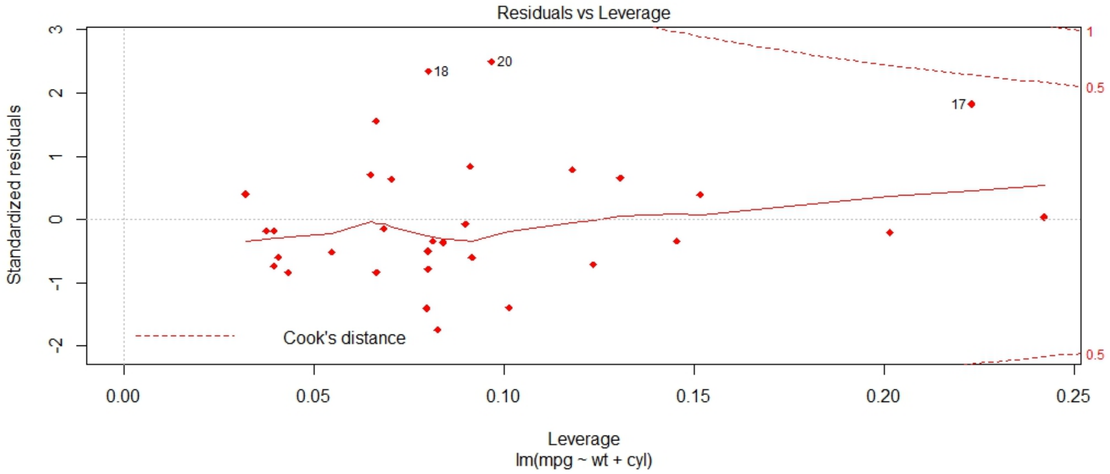


This plot is used to assess the validity of the normality of the residuals.

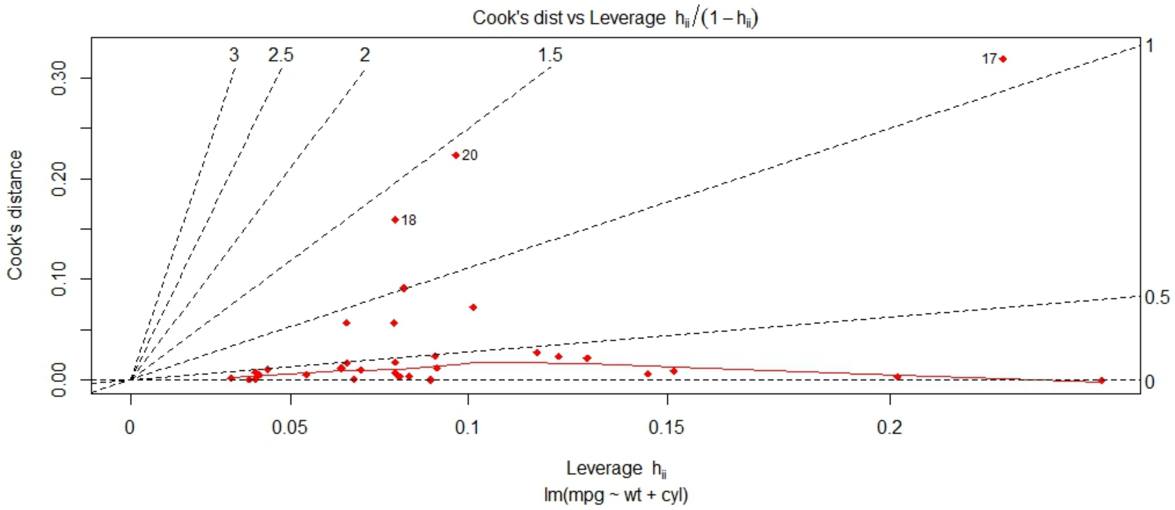


Plot 5 : Cook's Distance





Plot 6 : Cook's Distance vs Leverage



Plot the four default plots together:

```
par(mfrow=c(4,1))
plot(fittedmodel)
par(opar)
```

8 Outliers and Influential Observations

> ”Outliers are sample values that cause surprise in relation to the majority of the sample” (W.N. Venables and B.D. Ripley. 2002. Modern applied statistics with S. New York: Springer, p.119).

Crucially, surprise is in the mind of the beholder and is dependent on some explicit model of the data.

Importantly, Normality is only an assumption:There may be another model under which the outlier is not surprising at all, say if the data really are lognormal or gamma rather than normal.

8.1 Outliers

Data points that diverge in a big way from the overall pattern are referred to as “outliers”. In the case of Simple Linear Regression, there are four ways that a data point might be considered an outlier.

- It could have an extreme X value compared to other data points.
- It could have an extreme Y value compared to other data points.
- It could have extreme X and Y values.
- It might be distant from the rest of the data, even without extreme X or Y values.
- After a regression line has been computed for a group of data, a point which lies far from the line (and thus has a large residual value) is known as an outlier. Such points may represent erroneous data, or may indicate a poorly fitting regression line.
- If a point lies far from the other data in the horizontal direction, it is known as an ***influential observation***. The reason for this distinction is that these points have may have a significant impact on the slope of the regression line.

`outlierTest()`

Suppose we have a two fitted models and we would like to see if there are any outliers.
For this purpose, we can use `outlierTest()` from `library(car)` in R.

```
library(car)
outlierTest(fit1)

**Result:**
    rstudent unadjusted p-value  Bonferonni p
21    -4.12            4.39e-05        0.0209
15     -4.08            5.39e-05        0.0257

outlierTest(fit2)

**Result:**
No Studentized residuals with Bonferonni p < 0.05
Largest |rstudent|:
    rstudent unadjusted p-value      Bonferonni p
177    -2.52             0.0119                NA
```

The row numbers (here : 21, 15 and 177) indicate the outlier points in the data.

9 Influence

The influence of an observation can be thought of in terms of how much the predicted scores for other observations would differ if the observation in question were not included.

9.1 Influential Points

An influential point is an outlier that greatly affects the slope of the regression line. One way to test the influence of an outlier is to compute the regression equation with and without the outlier.

This type of analysis is illustrated below. The scatter plots are identical, except that the plot on the right includes an outlier. The slope is flatter when the outlier is present (-3.32 vs. -4.10), so this outlier would be considered an influential point.

9.2 Without Outlier

- Regression equation: $\hat{y} = 104.78 - 4.10x$
- Coefficient of determination: $R^2 = 0.94$
- Regression equation: $\hat{y} = 97.51 - 3.32x$
- Coefficient of determination: $R^2 = 0.55$

The charts below compare regression statistics for another data set with and without an outlier. Here, the chart on the right has a single outlier, located at the high end of the X axis (where x = 24). As a result of that single outlier, the slope of the regression line changes greatly, from -2.5 to -1.6; so the outlier would be considered an influential point.

Sometimes, an influential point will cause the coefficient of determination to be bigger; sometimes, smaller. In the first example above, the coefficient of determination is smaller when the influential point is present (0.94 vs. 0.55). In the second example, it is bigger (0.46 vs. 0.52).

If your data set includes an influential point, here are some things to consider.

- An influential point may represent bad data, possibly the result of measurement error. If possible, check the validity of the data point.
- Compare the decisions that would be made based on regression equations defined with and without the influential point. If the equations lead to contrary decisions, use caution.

10 Standardization and Studentization

10.1 Standardization

A random variable is said to be standardized if the difference from its mean is scaled by its standard deviation. The residuals above have mean zero but their variance is unknown, it depends on the true values of θ. Standardization is thus not possible in practice.

10.2 Studentization

Instead, you can compute studentized residuals by dividing a residual by an estimate of its standard deviation.

10.3 Internal and External Studentization

If that estimate is independent of the $i-$th observation, the process is termed *external studentization*'external studentization'. This is usually accomplished by excluding the $i-$th observation when computing the estimate of its standard error. If the observation contributes to the standard error computation, the residual is said to be *internally studentization*internally studentized.

11 Standardized and Studentized Residuals

The standardized residual is the residual divided by its standard deviation.

Plot the standardized residual of the simple linear regression model of the data set faithful against the independent variable waiting.

We apply the lm function to a formula that describes the variable eruptions by the variable waiting, and save the linear regression model in a new variable eruption.lm. Then we compute the standardized residual with the rstandard function.

```
eruption.lm = lm(eruptions ~ waiting, data=faithful)
eruption.stdres = rstandard(eruption.lm)
```

We now plot the standardized residual against the observed values of the variable waiting.

```
plot(faithful$waiting, eruption.stdres,
    ylab="Standardized Residuals",
     xlab="Waiting Time",
     main="Old Faithful Eruptions")
abline(0, 0)                  # the horizon
```

12 Regression Deletion Diagnostics

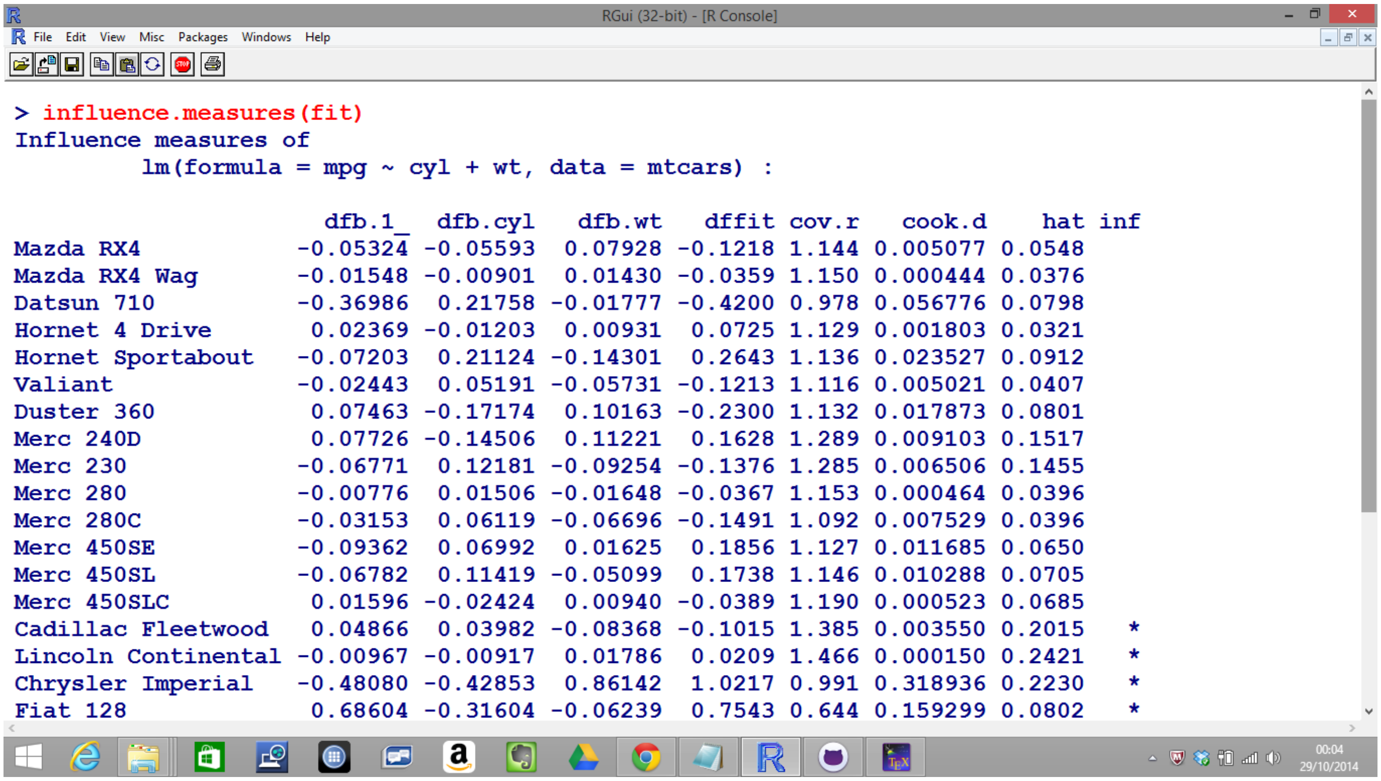
This suite of functions can be used to compute some of the regression (leave-one-out deletion) diagnostics for linear and generalized linear models discussed in Belsley, Kuh and Welsch (1980), Cook and Weisberg (1982), etc.

Details

- The primary high-level function is `influence.measures` which produces a class "infl" object tabular display showing the DFBETAS for each model variable, DFFITS, covariance ratios, Cook's distances and the diagonal elements of the hat matrix. Cases which are influential with respect to any of these measures are marked with an asterisk.
- The functions `dfbetas`, `dffits`, `covratio` and `cooks.distance` provide direct access to the corresponding diagnostic quantities.
- Functions `rstandard` and `rstudent` give the standardized and Studentized residuals respectively.
- (These functions re-normalize the residuals to have unit variance, using an overall and leave-one-out measure of the error variance respectively.)
- Values for generalized linear models are approximations, as described in Williams (1987) (except that Cook's distances are scaled as F rather than as chi-square values). The approximations can be poor when some cases have large influence.
- The optional `infl`, `res` and `sd` arguments are there to encourage the use of these direct access functions, in situations where, e.g., the underlying basic influence measures (from `lm.influence` or the generic influence) are already available.
- Note that cases with `weights == 0` are dropped from all these functions, but that if a linear model has been fitted with `na.action = na.exclude`, suitable values are filled in for the cases excluded during fitting.
- The function `hat()` exists mainly for S (version 2) compatibility; we recommend using `hatvalues()` instead.

```
Usage
influence.measures(model)

rstandard(model, ...)

## S3 method for class 'lm'
rstandard(model, infl = lm.influence(model, do.coef = FALSE),
sd = sqrt(deviance(model)/df.residual(model)), ...)

## S3 method for class 'glm'
rstandard(model, infl = influence(model, do.coef = FALSE),
type = c("deviance", "pearson"), ...)
```

```
rstudent(model, ...)

## S3 method for class 'lm'
rstudent(model, infl = lm.influence(model, do.coef = FALSE),
res = infl$wt.res, ...)

## S3 method for class 'glm'
rstudent(model, infl = influence(model, do.coef = FALSE), ...)

dffits(model, infl = , res = )
```



RGui (32-bit) - [R Console]
File Edit View Misc Packages Windows Help
00:04
29/10/2014

```
> influence.measures(fit)
Influence measures of
         lm(formula = mpg ~ cyl + wt, data = mtcars) :

                      dfb.1_  dfb.cyl   dfb.wt   dffit cov.r   cook.d    hat inf
Mazda RX4           -0.05324 -0.05593  0.07928 -0.1218 1.144 0.005077 0.0548
Mazda RX4 Wag       -0.01548 -0.00901  0.01430 -0.0359 1.150 0.000444 0.0376
Datsun 710          -0.36986  0.21758 -0.01777 -0.4200 0.978 0.056776 0.0798
Hornet 4 Drive       0.02369 -0.01203  0.00931  0.0725 1.129 0.001803 0.0321
Hornet Sportabout   -0.07203  0.21124 -0.14301  0.2643 1.136 0.023527 0.0912
Valiant             -0.02443  0.05191 -0.05731 -0.1213 1.116 0.005021 0.0407
Duster 360           0.07463 -0.17174  0.10163 -0.2300 1.132 0.017873 0.0801
Merc 240D            0.07726 -0.14506  0.11221  0.1628 1.289 0.009103 0.1517
Merc 230            -0.06771  0.12181 -0.09254 -0.1376 1.285 0.006506 0.1455
Merc 280            -0.00776  0.01506 -0.01648 -0.0367 1.153 0.000464 0.0396
Merc 280C           -0.03153  0.06119 -0.06696 -0.1491 1.092 0.007529 0.0396
Merc 450SE          -0.09362  0.06992  0.01625  0.1856 1.127 0.011685 0.0650
Merc 450SL          -0.06782  0.11419 -0.05099  0.1738 1.146 0.010288 0.0705
Merc 450SLC          0.01596 -0.02424  0.00940 -0.0389 1.190 0.000523 0.0685
Cadillac Fleetwood   0.04866  0.03982 -0.08368 -0.1015 1.385 0.003550 0.2015   *
Lincoln Continental -0.00967 -0.00917  0.01786  0.0209 1.466 0.000150 0.2421   *
Chrysler Imperial   -0.48080 -0.42853  0.86142  1.0217 0.991 0.318936 0.2230   *
Fiat 128             0.68604 -0.31604 -0.06239  0.7543 0.644 0.159299 0.0802   *
```

Figure 5:

13 Influential Points in Regression

```
inflm.fit <- influence.measures(fit)
which(apply(inflm.fit$is.inf, 1, any))
```

```
dfbeta(model, ...)
## S3 method for class 'lm'
dfbeta(model, infl = lm.influence(model, do.coef = TRUE), ...)

dfbetas(model, ...)
## S3 method for class 'lm'
dfbetas(model, infl = lm.influence(model, do.coef = TRUE), ...)

covratio(model, infl = lm.influence(model, do.coef = FALSE),
res = weighted.residuals(model))
```

```
cooks.distance(model, ...)
## S3 method for class 'lm'
cooks.distance(model, infl = lm.influence(model, do.coef = FALSE),
res = weighted.residuals(model),
sd = sqrt(deviance(model)/df.residual(model)),
hat = infl$hat, ...)
## S3 method for class 'glm'
cooks.distance(model, infl = influence(model, do.coef = FALSE),
res = infl$pear.res,
dispersion = summary(model)$dispersion,
hat = infl$hat, ...)
```

```
hatvalues(model, ...)
## S3 method for class 'lm'
```

```
hatvalues(model, infl = lm.influence(model, do.coef = FALSE), ...)

hat(x, intercept = TRUE)
```

Arguments

model an R object, typically returned by lm or glm.

infl influence structure as returned by lm.influence or influence (the la

res (possibly weighted) residuals, with proper default.

sd standard deviation to use, see default.

dispersion dispersion (for glm objects) to use, see default.

hat hat values H[i,i], see default.

type type of residuals for glm method for rstandard.

x the X or design matrix.

intercept should an intercept column be prepended to x?

... further arguments passed to or from other methods.

14 Cook's Distance

- Cook's distance or Cook's D is a commonly used estimate of the influence of a data point when performing least squares regression analysis.
- Cook's distance is useful for identifying outliers in the X values (observations for predictor variables). It also shows the influence of each observation on the fitted response values.
- **(Case Deletion Diagnostics)** If predictions are the same with or without the observation in question, then the observation has no influence on the regression model. If the predictions differ greatly when the observation is not included in the analysis, then the observation is influential.
- Cook's distance measures the effect of deleting a given observation. Data points with large residuals (outliers) and/or high leverage may distort the outcome and accuracy of a regression.
- Cook's distance is the scaled change in fitted values. Each resulting element in a diagnostic calculation is the normalized change in the vector of coefficients due to the deletion of an observation.
- In a practical ordinary least squares analysis, Cook's distance can be used in several ways: to indicate data points that are particularly worth checking for validity; to indicate regions of the design space where it would be good to be able to obtain more data points.
- It is named after the American statistician R. Dennis Cook, who introduced the concept in 1977.
- Points with a large Cook's distance are considered to merit closer examination in the analysis.
- Influential cases are not usually a problem when their removal from the dataset would leave the parameter estimates essentially unchanged: the

ones we worry about are those whose presence really does change the results.

Cook's Distance Formula

Cook's D is a good measure of the influence of an observation and is proportional to the sum of the squared differences between predictions made with all observations in the analysis and predictions made leaving out the observation in question.

It is calculated as:

$$D_i = \frac{\sum_{j=1}^{n}(\hat{Y}_j \ - \hat{Y}_{j(i)})^2}{p \text{ MSE}},$$

where:

- $\hat{Y}_j$ is the prediction from the full regression model for observation j;
- $\hat{Y}_{j(i)}$ is the prediction for observation j from a refitted regression model in which observation i has been omitted;
- p is the number of fitted parameters in the model;
- MSE is the mean square error of the regression model.

For the case of simple linear regression, the following are the algebraically equivalent expressions

$$D_i = \frac{e_i^2}{p \text{ MSE}} \left[\frac{h_{ii}}{(1 - h_{ii})^2} \right],$$

$$D_i = \frac{(\hat{\beta} - \hat{\beta}^{(-i)})^T (X^T X)(\hat{\beta} - \hat{\beta}^{(-i)})}{(1 + p)s^2},$$

where:

- h_{ii} is the leverage, i.e., the i-th diagonal element of the hat matrix

$$\mathbf{X} \left(\mathbf{X}^T \mathbf{X} \right)^{-1} \mathbf{X}^T$$

- e_i is the residual (i.e., the difference between the observed value and the value fitted by the proposed model).

R code for computing Cook's Distance

```
attach(mtcars)
fit = lm(mpg ~ cyl + wt )
cooks.distance(fit)
plot(cooks.distance(fit),type="b",pch=18,col="red")

N = 32
k = 2
cutoff = 4/ (N-k-1)
abline(h=cutoff,lty=2)
```

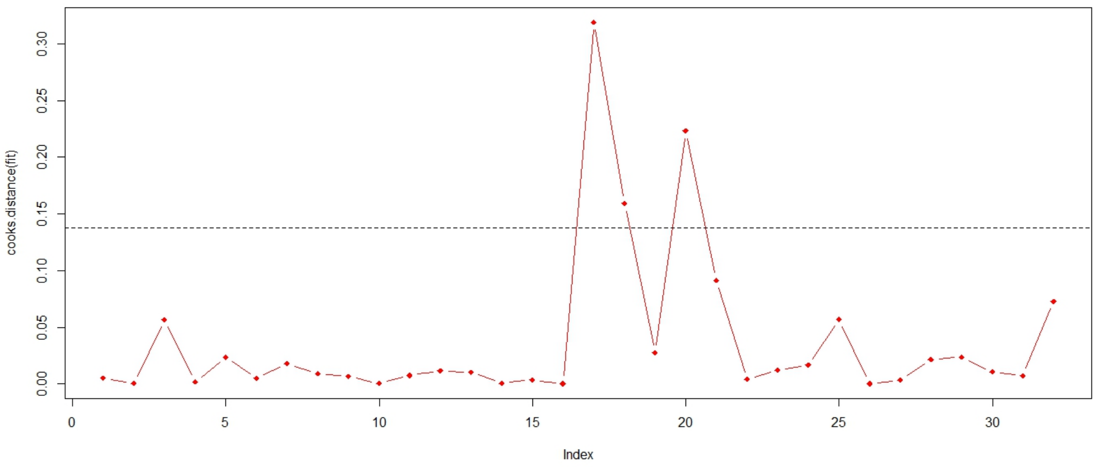


Figure 6:

14.1 Interpreting Cook's Distance

> A common rule of thumb is that an observation with a value of Cook's D over 1.0 has too much influence. As with all rules of thumb, this rule should be applied judiciously and not thoughtlessly. (John Fox) *(1991). Regression Diagnostics: An Introduction. Sage Publications.*

- Some texts tell you that points for which Cook's distance is higher than 1 are to be considered as influential.
- Other texts give you a threshold of $4/N$ or
$$\frac{4}{(N-k-1)},$$
where N is the number of observations and k the number of explanatory variables.
- The `R` help file advises that an observation with Cook's distance larger than three times the mean Cook's distance might be an outlier. .
- John Fox (mentioned above), in his booklet on regression diagnostics is rather cautious when it comes to giving numerical thresholds. He advises the use of graphics and to examine in closer details the points with "values of D that are substantially larger than the rest". According to Fox, thresholds should just be used to enhance graphical displays.

Cook's Distance in relation to other measures

- Cook's distance refers to how far, on average, predicted y-values will move if the observation in question is dropped from the data set.
- **DFBETA** : *DFBETA* refers to how much a parameter estimate changes if the observation in question is dropped from the data set.
- *Note that with k covariates, there will be k+1 DFBETAs (the intercept,* β_0*, and* β_1 *for each covariate).*
- Cook's distance is arguably more important if you are doing predictive modeling, whereas *DFBETA* is more important in explanatory modeling.
- **DFFITS**: Although the raw values resulting from the equations are different, Cook's distance and *DFFITS* are conceptually identical and there is a closed-form formula to convert one value to the other.

15 Some Important Definitions

To understand a diagnostic plot called the residual-leverage plot, we must understand three things:

- Leverage,
- Standardized residuals, and
- Cook's distance.

Consider the plots associated with four different situations:

1. a dataset where everything is fine
2. a dataset with a high-leverage, but low-standardized residual point
3. a dataset with a low-leverage, but high-standardized residual point
4. a dataset with a high-leverage, high-standardized residual point

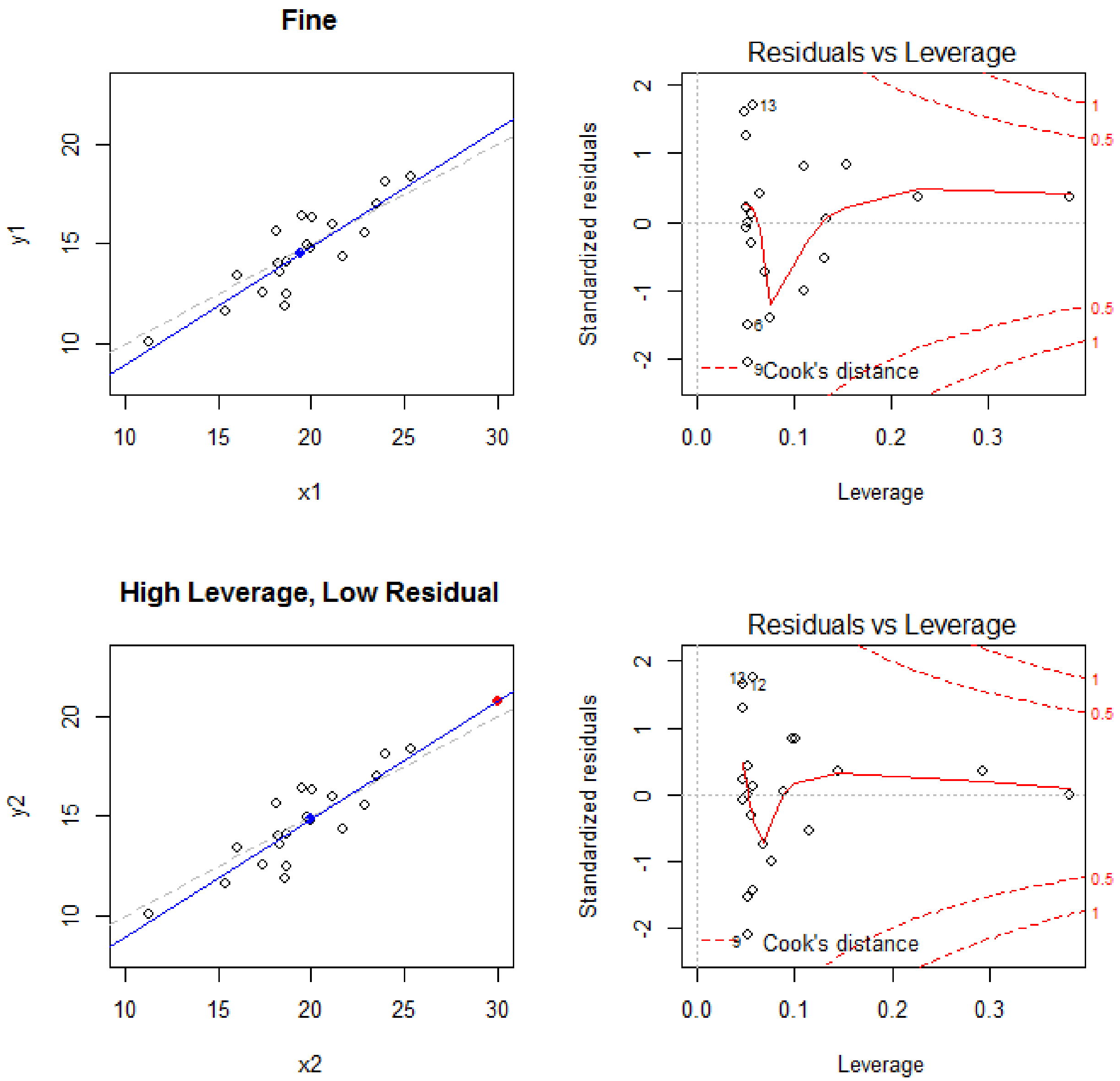

Fine
y1
10
15
20
x1
10
15
20
25
30
Residuals vs Leverage
Standardized residuals
-2
-1
0
1
2
Leverage
0.0
0.1
0.2
0.3
13
6
9
Cook's distance
1
0.5
0.5
1
High Leverage, Low Residual
y2
10
15
20
x2
10
15
20
25
30
Residuals vs Leverage
Standardized residuals
-2
-1
0
1
2
Leverage
0.0
0.1
0.2
0.3
13
12
9
Cook's distance
1
0.5
0.5
1

Figure 7:

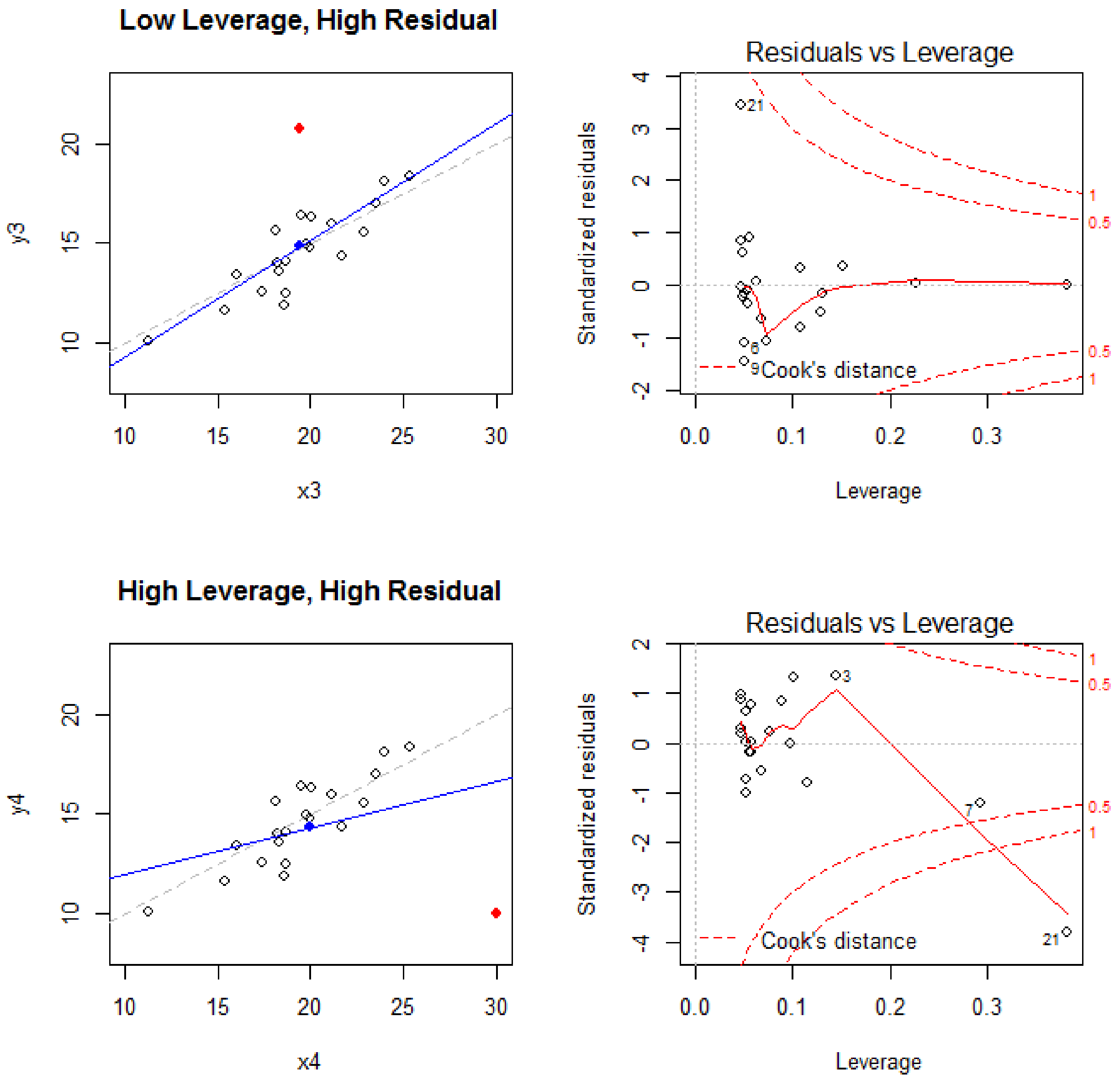

Low Leverage, High Residual
y3
x3
10
15
20
25
30
Residuals vs Leverage
Standardized residuals
Leverage
0.0
0.1
0.2
0.3
-2
-1
0
1
2
3
4
21
6
9
Cook's distance
1
0.5
High Leverage, High Residual
y4
x4
Residuals vs Leverage
-4
-3
3
7
21

Figure 8:

- The plots on the left show the data, the center of the data with a blue dot, the underlying data generating process with a dashed gray line, the model fit with a blue line, and the special point with a red dot.
- On the right are the corresponding residual-leverage plots; the special point is 21.
- The model is badly distorted primarily in the fourth case where there is a point with high leverage and a large (negative) standardized residual.

Regression Diagnostics

An excellent review of regression diagnostics is provided in John Fox's aptly named *Overview of Regression Diagnostics*.

Dr. Fox's ***car*** package provides advanced utilities for regression modeling.

15.1 Alcohol and Tobacco Data

This example is for exposition only. We will ignore the fact that this may not be a great way of modeling the this particular set of data!

```
alctob <- data.frame( cbind(
Alcohol = c(6.47, 6.13, 6.19, 4.89, 5.63, 4.52,
            5.89, 4.79, 5.27, 6.08, 4.02),
Tobacco = c(4.03, 3.76, 3.77, 3.34, 3.47, 2.92,
            3.20, 2.71, 3.53, 4.51, 4.56)),
row.names = c("North", "Yorkshire", "Northeast",
"East Midlands", "West Midlands", "East Anglia",
"Southeast", "Southwest", "Wales",
"Scotland", "N. Ireland"))

# Assume that we are fitting a multiple linear regression
# on the MTCARS data
library(car)
fit <- lm(mpg˜disp+hp+wt+drat, data=mtcars)
```

```
alctobwo <- subset(alctob,rownames(alctob)!="N. Ireland")
#without North Ireland

plot(alctob$Tobacco, alctob$Alcohol,
main="Weekly Household Spending on Alcohol vs. Tobacco",
xlab="Tobacco Spending (GBP)",
ylab="Alcohol Spending (GBP)",
pch=16,col="red",cex=1.5,font.lab=2)
#note N. Ireland in the bottom-right
```

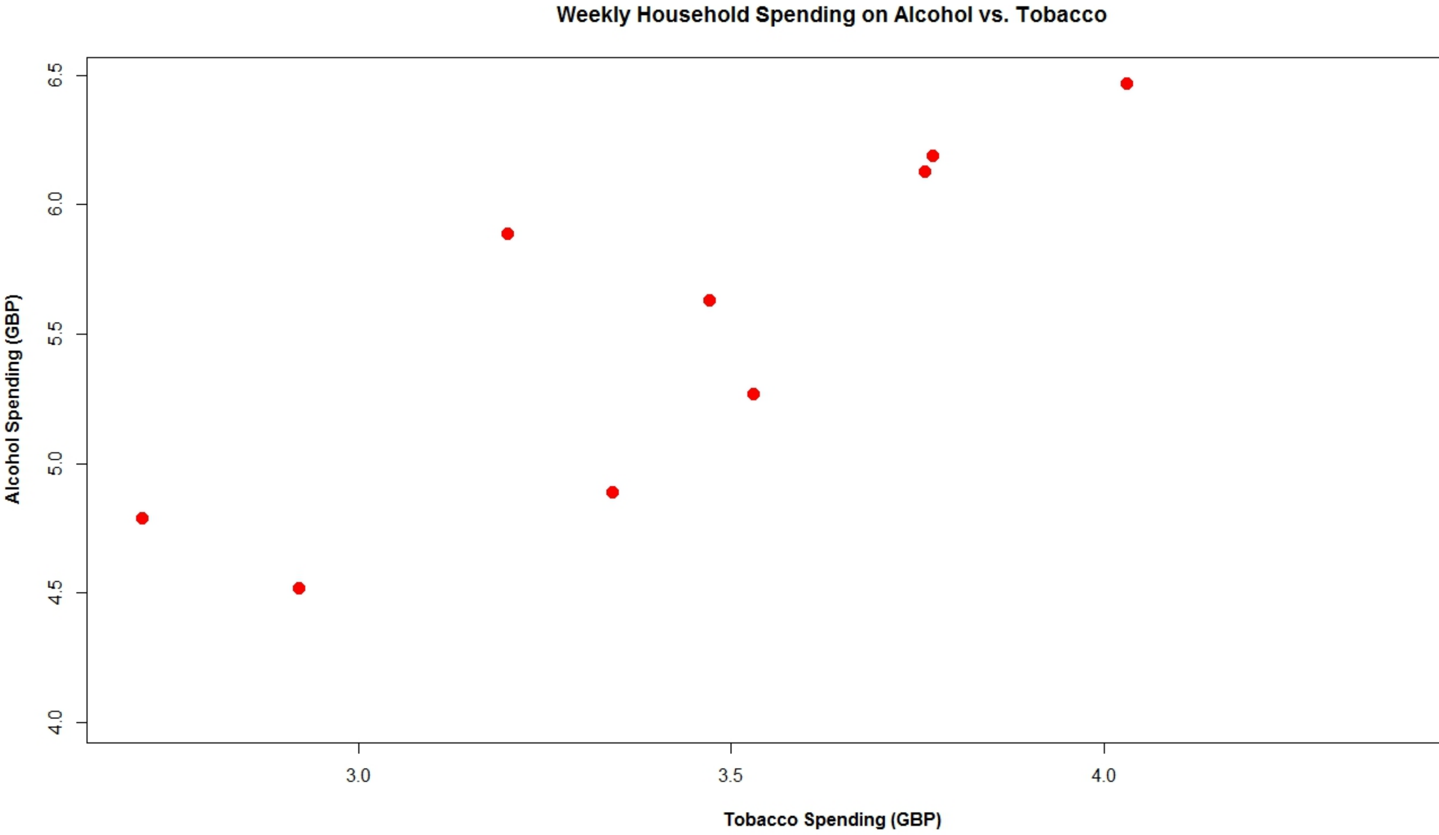

Weekly Household Spending on Alcohol vs. Tobacco
Alcohol Spending (GBP)
6.5
6.0
5.5
5.0
4.5
4.0
3.0
3.5
4.0
Tobacco Spending (GBP)

Figure 9:

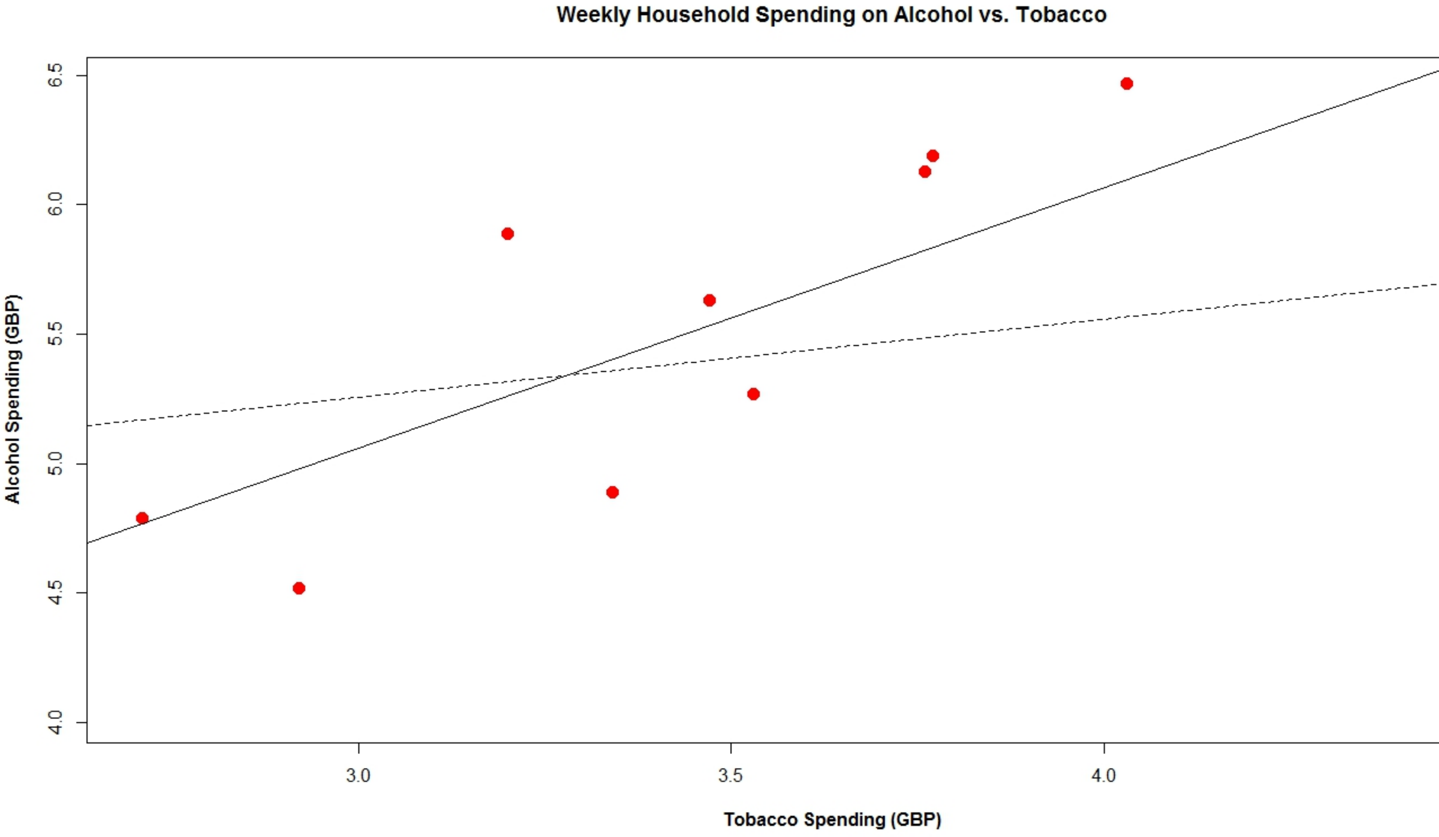

Weekly Household Spending on Alcohol vs. Tobacco
Alcohol Spending (GBP)
Tobacco Spending (GBP)
4.0
4.5
5.0
5.5
6.0
6.5
3.0
3.5
4.0

Figure 10:

All Observations

```
> summary(fit1)

Call:
lm(formula = Alcohol ˜ Tobacco, data = alctob)

Residuals:
Min      1Q  Median      3Q     Max
-1.7080 -0.4245  0.2311  0.6081  0.9020

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept)   4.3512     1.6067   2.708   0.0241 *
Tobacco       0.3019     0.4388   0.688   0.5087
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.8196 on 9 degrees of freedom
Multiple R-squared:  0.04998,   Adjusted R-squared:  -0.05557
F-statistic: 0.4735 on 1 and 9 DF,  p-value: 0.5087
```

Outlier Removed

```
> summary(fit2)

Call:
lm(formula = Alcohol ˜ Tobacco, data = alctobwo)

Residuals:
Min       1Q   Median       3Q      Max
-0.51092 -0.42434  0.06056  0.34406  0.62991

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.0412     1.0014   2.038  0.07586 .
```

```
Tobacco       1.0059     0.2813   3.576  0.00723 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.446 on 8 degrees of freedom
Multiple R-squared:  0.6151,    Adjusted R-squared:  0.567
F-statistic: 12.78 on 1 and 8 DF,  p-value: 0.007234
```

Outliers

The conservative outlier test that we talked about in class uses the Bonferonni inequality to calculate the p-values we associate with the Student's-t test.

In R, we can use the outlierTest command to perform this test on our model. Remember that when we test for influence, we are testing the effect of an observation on model coefficients.

Therefore we need to give the outlierTest command a linear model as its input.

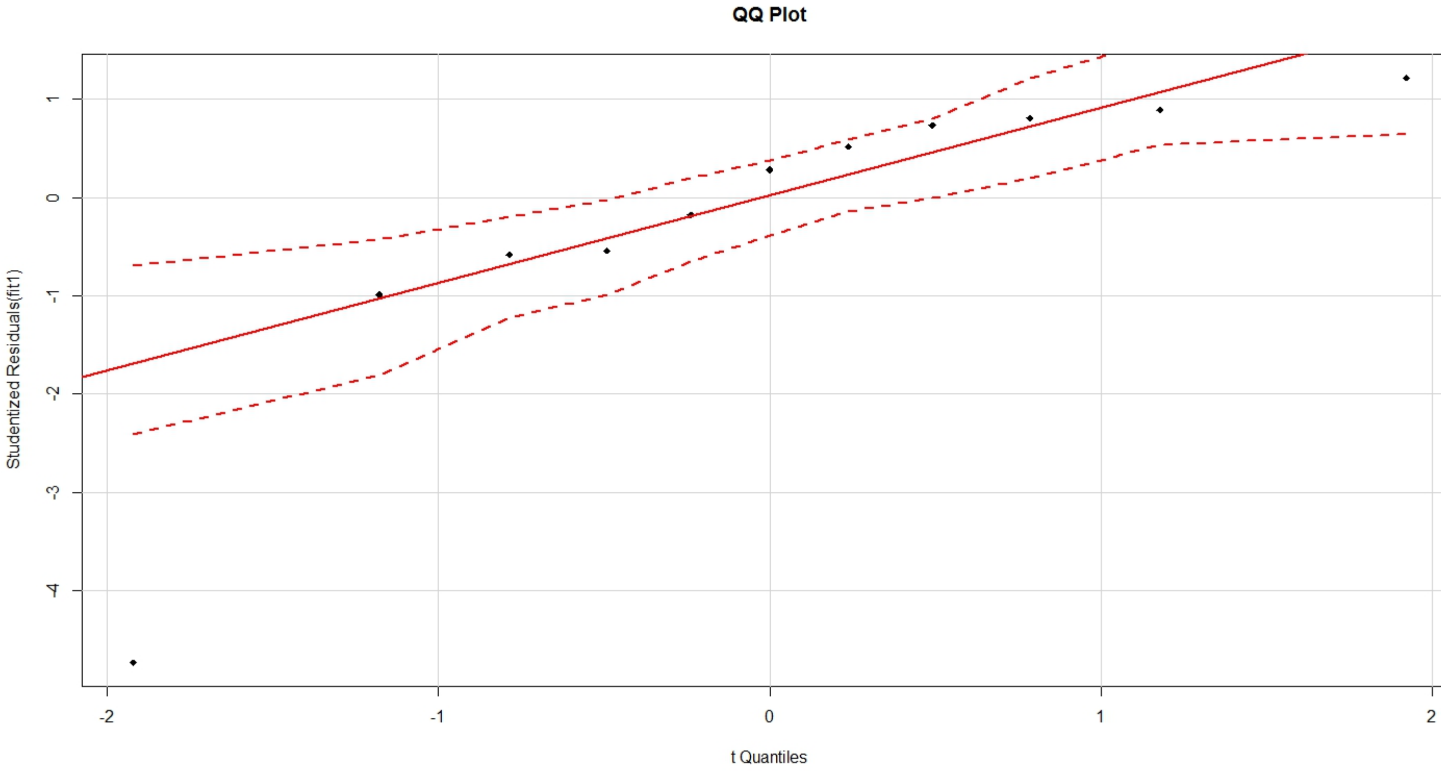
```
> # Assessing Outliers
> # syntax: outlierTest(fit)
> # Bonferonni p-value for most extreme obs
>
> outlierTest(fit1)
            rstudent  unadjusted p-value Bonferonni p
N. Ireland -4.732091           0.0014789     0.016268
>
```

We can also use R to calculate Cook's distance. Remember that generally we label any observation with Cook's distance greater than 1 as influential.

```
> cooks.distance(fit1)
North           Yorkshire       Northeast     East Midlands
0.114101051     0.036517838     0.043728951   0.023600304
West Midlands   East Anglia     Southeast     Southwest
0.004740759     0.147326647     0.046646563   0.077488350
Wales           Scotland        N. Ireland
0.001821694     0.068921892     1.747233521
```

Finally, one of the easier ways to evaluate our residuals and look for for influential points is through plots.

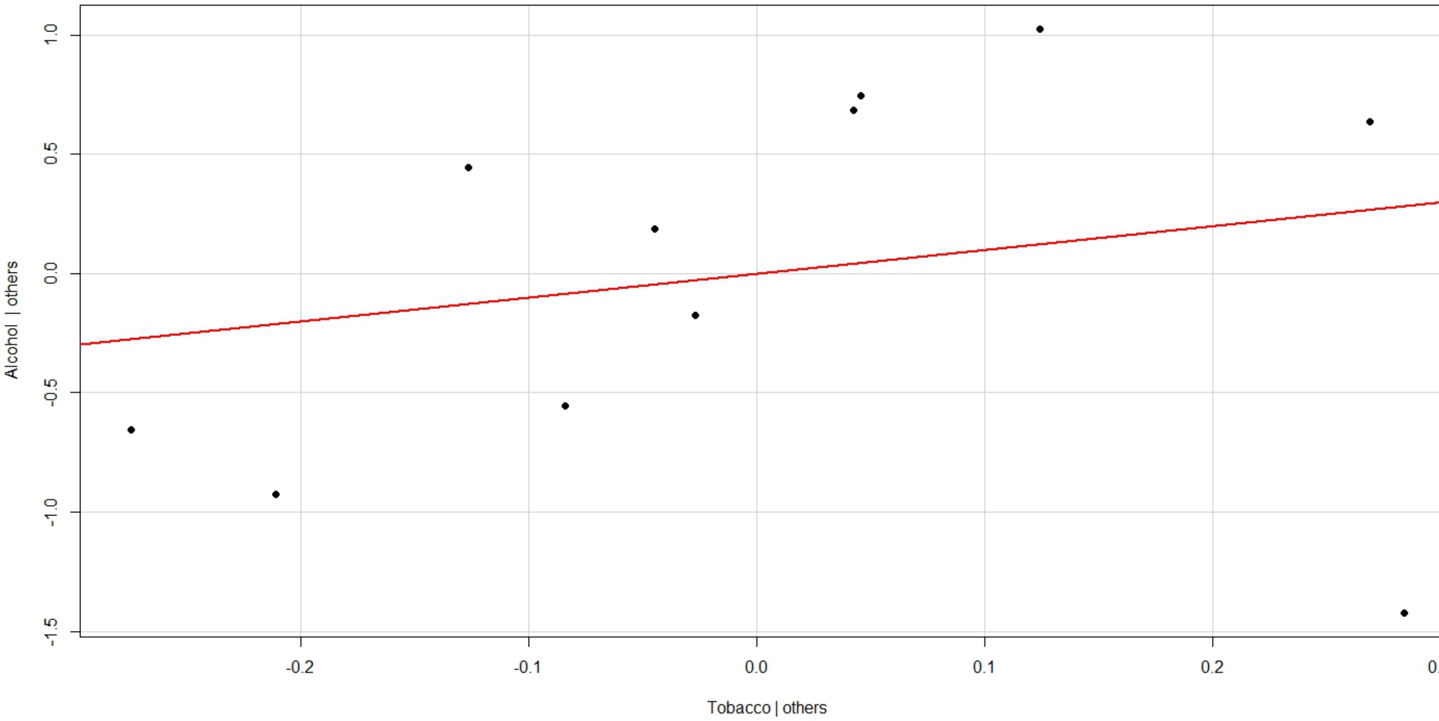
```
qqPlot(fit1, main="QQ Plot",pch=18, lim=c(-3,2)) #qq plot for studentized resid
leveragePlots(fit1) # leverage plots
```



In R, the `plot` command takes a special form when you pass it an `lm` object (see ?plot.lm for all of the details).

Here we want to focus on three of the plots available through the command.

- The first one displays the residuals vs. the fitted values we use this to evlauate the mean, variance and correlation of residuals. If our assumptions of constant variance and uncorrelated residuals are violated we may be able to correct this with a variance-stabilizing transformation.
- The second plot helps us check the normality of the residuals. If the residuals are indeed normal, they should fall along the dashed line. Remember that the normality assumption for our errors allows us to determine the standard errors of our coefficients and predictions.
- The final plot will display our residuals vs. their leverage. The dashed red lines are level curves that denote a particular value of Cook's distance. We will pay attention to points lying beyond the distance of 1. Notice that when we have data with row labels, the points will be labeled with their names. Otherwise, the row number will be shown.


1.0
0.5
0.0
-0.5
-1.0
-1.5
Alcohol | others
-0.2
-0.1
0.0
0.1
0.2
0.
Tobacco | others

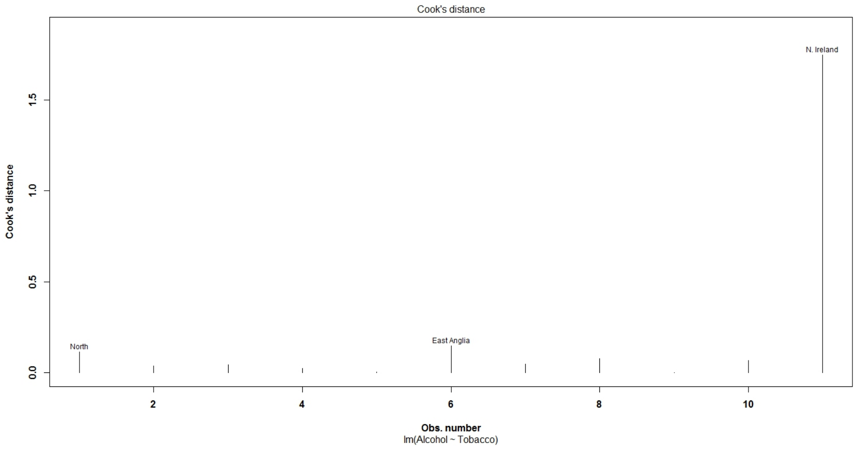


Figure 11:

```
plot(lmwith, which=c(1,2,5))
```

```
Influential Observations
# Influential Observations
# added variable plots
avPlots(fit1)


# Cook’s D plot
# identify D values > 4/(n-k-1)
cutoff <- 4/((nrow(mtcars)-length(fit1$coefficients)-2))
plot(fit, which=4, cook.levels=cutoff)
```

```
# Influence Plot
influencePlot(fit1,id.method="identify",
main="Influence Plot",
col="red"
sub="Circle size is proportial to Cook's Distance" )
```

Non-normality

```
# Normality of Residuals
# qq plot for studentized resid
qqPlot(fit1, main="QQ Plot")


# distribution of studentized residuals
library(MASS)
sresid <- studres(fit)
hist(sresid, freq=FALSE,
main="Distribution of Studentized Residuals")
xfit<-seq(min(sresid),max(sresid),length=40)
yfit<-dnorm(xfit)
lines(xfit, yfit)
```

Non-constant Error Variance

```
# Evaluate homoscedasticity
# non-constant error variance test
ncvTest(fit)
# plot studentized residuals vs. fitted values
spreadLevelPlot(fit)
spread vs. levels click to view
```

```
Multi-collinearity
# Evaluate Collinearity
vif(fit) # variance inflation factors
sqrt(vif(fit)) > 2 # problem?
```

15.2 Nonlinearity

```
# Evaluate Nonlinearity
# component + residual plot
crPlots(fit)
# Ceres plots
ceresPlots(fit)
component plus residual plot Ceres plots click to view
Non-independence of Errors
# Test for Autocorrelated Errors
durbinWatsonTest(fit)
```

15.3 Additional Diagnostic Help

The gvlma( ) function in the gvlma package, performs a global validation of linear model assumptions as well separate evaluations of skewness, kurtosis, and heteroscedasticity.

```
# Global test of model assumptions
library(gvlma)
gvmodel <- gvlma(fit)
summary(gvmodel)
```

15.4 Going Further

If you would like to delve deeper into regression diagnostics, two books written by John Fox can help: Applied regression analysis and generalized linear models (2nd ed) and An R and S-Plus companion to applied regression.