Regression Diagnostics Notes

Lt Col Ken Horton   Lt Col Kris Pruitt   Professor Bradley Warner

10 November, 2020

Objectives

1) Obtain and interpret R-squared and the F-statistic.
2) Use `R` to evaluate the assumptions of a linear model.
3) Identify and explain outliers and leverage points.

Introduction

Over the last two lessons, we have detailed simple linear regression. First, we described the model and its underlying assumptions. Next, we obtained parameter estimates using the method of least squares. Finally, we obtained the distributions of parameter estimates and used that information to conduct inference on parameters and predictions. Implementation was relatively straightforward; once we obtained the expressions of interest, we used `R` to find parameters estimates, interval estimates, etc. In this lesson we will explore more tools to assess the quality of our simple linear regression model. Some these will generalize when we move to multiple predictors.

Assessing our model - understanding the output from `lm`

There is more that we can do with the output from the `lm()` function than just estimating parameters and predicting responses. There are metrics that allow us assess the fit of our model. To explore some of these ideas let's use the Starbucks example again.

First load the data:

```
starbucks <- read_csv("data/starbucks.csv")
```

Next build and summarize the model:

```
star_mod <- lm(calories~carb,data=starbucks)
```

```
summary(star_mod)
```

```
##
## Call:
## lm(formula = calories ~ carb, data = starbucks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -151.962  -70.556   -0.636   54.908  179.444
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 146.0204    25.9186   5.634 2.93e-07 ***
## carb          4.2971     0.5424   7.923 1.67e-11 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 78.26 on 75 degrees of freedom
## Multiple R-squared:  0.4556, Adjusted R-squared:  0.4484
## F-statistic: 62.77 on 1 and 75 DF,  p-value: 1.673e-11
```

You may have noticed some other information that appeared in the summary of our model. We discussed the output in a previous lesson but let’s go a little more in depth.

Residual Standard Error

The “residual standard error” is the estimate of σ, the unexplained variance in our response. In our example, this turned out to be 78.26. If the assumption of normality is valid, we would expect the majority, 68%, of the values to be within ±78.26 of the mean value.

If we want to extract just this value from the model object, first recognize that `summary(my.model)` is a list with several components:

```r
names(summary(star_mod))
```

```
##  [1] "call"          "terms"         "residuals"     "coefficients"
##  [5] "aliased"       "sigma"         "df"            "r.squared"
##  [9] "adj.r.squared" "fstatistic"    "cov.unscaled"
```

As expected, the `sigma` component shows the estimated value of σ.

```r
summary(star_mod)$sigma
```

```
## [1] 78.25956
```

Again, this value is smaller the closer the points are to the regression fit. It is a measure of unexplained variance in the response variable.

R-squared

Another quantity that appears is R-squared. R-squared is one measure of goodness of fit. Essentially, R-squared is a ratio of variance (in the response) explained by the model to overall variance of the response. It helps to describe the decomposition of variance:

$$\underbrace{\sum_{i=1}^{n}(y_i - \bar{y})^2}_{SS_{\text{Total}}} = \underbrace{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}_{SS_{\text{Regression}}} + \underbrace{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}_{SS_{\text{Error}}}$$

In other words, the overall variation in y can be separated into two parts: variation due to the linear relationship between y and the predictor variable(s), called $SS_{\text{Regression}}$, and residual variation (due to random scatter or perhaps a poorly chosen model), called SS_{Error}.

R-squared simply measures the ratio between $SS_{\text{Regression}}$ and SS_{Total}. A common definition of R-squared is the proportion of overall variation in the response that is explained by the linear model. R-squared can be between 0 and 1. Values of R-squared close to 1 indicate a tight fit (little scatter) around the estimated regression line. Value close to 0 indicate the opposite (large remaining scatter).

We can obtain R-squared "by hand" or by using the output of the `lm()` function:

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

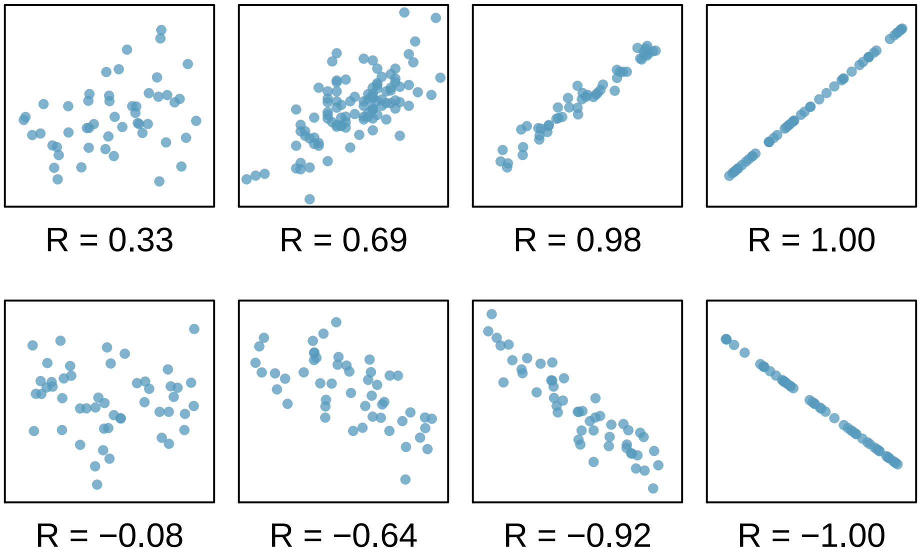
```
## [1] 0.4556237
```

For simple linear regression, R-squared is related to **correlation**. We can compute the correlation using a formula, just as we did with the sample mean and standard deviation. However, this formula is rather complex,[1] so we let `R` do the heavy lifting for us.

```
starbucks %>%
  summarize(correlation=cor(carb,calories),correlation_squared=correlation^2)
```

```
## # A tibble: 1 x 2
##   correlation correlation_squared
##         <dbl>               <dbl>
## 1       0.675               0.456
```

As a review, the figure below shows eight plots and their corresponding correlations. Only when the relationship is perfectly linear is the correlation either -1 or 1. If the relationship is strong and positive, the correlation will be near +1. If it is strong and negative, it will be near -1. If there is no apparent linear relationship between the variables, then the correlation will be near zero.



R = 0.33

R = 0.69

R = 0.98

R = 1.00

R = −0.08

R = −0.64

R = −0.92

R = −1.00

[1] Formally, we can compute the correlation for observations (x_1, y_1), (x_2, y_2), $\ldots$, (x_n, y_n) using the formula

$$R = \frac{1}{n-1} \sum_{i=1}^{n} \frac{x_i - \bar{x}}{s_x} \frac{y_i - \bar{y}}{s_y}$$

where $\bar{x}$, $\bar{y}$, s_x, and s_y are the sample means and standard deviations for each variable.

> **Exercise**
> If a linear model has a very strong negative relationship with a correlation of -0.97, how much of the variation in the response is explained by the explanatory variable?[2]

Note that one of the components of `summary(lm())` function is `adj.r.squared`. This is a value of R-squared adjusted for number of predictors. We'll cover this concept more closely in Math 378.

F-Statistic

Another quantity that appears in the summary of the model is the F-statistic. This value evaluates the null hypothesis that all of the non-intercept coefficients are equal to 0. Rejecting this hypothesis implies that the model is useful in the sense that at least one of the predictors shares a significant linear relationship with the response.

H_0: $\beta_1 = \beta_2 = \cdots = \beta_p = 0$ H_a: At least one coefficient not equal to 0.

where p is the number of predictors in the model. Just like in ANOVA, this is a simultaneous test of all coefficients and does not inform us which one(s) are different from 0.

The F-statistic is given by

$$\frac{n-p-1}{p}\frac{\sum(\hat{y}_i - \bar{y})^2}{\sum e_i^2}$$

Under the null hypothesis, the F-statistic follows the F distribution with parameters p and $n-p-1$.

In our example, the F-statistic is redundant since there is only one predictor. In fact, the p-value associated with the F-statistic is equal to the p-value associated with the estimate of β_1. However, when we move to cases with more predictor variables, we may be interested in the F-statistic.

```r
summary(star_mod)$fstatistic
```

```
##    value    numdf    dendf 
## 62.77234  1.00000 75.00000
```

Regression diagnostics

Finally, we can use the "lm" object to check the assumptions of the model. We have discussed the assumptions before but in this lesson we will use `R` to generate visual checks. There are also numeric diagnostic measures.

There are several potential problems with a regression model:

1) Assumptions about the error structure. We assume:

- the errors are normally distributed
- the errors are independent
- the errors have constant variance, **homoskedastic**

2) Assumptions about the fit. We assume that fit of the model is correct. For a simple linear regression, this means that fit specified by the formula in `lm` is correct.

[2] About $R^2 = (-0.97)^2 = 0.94$ or 94% of the variation is explained by the linear model.

3) Problems with outliers and leverage points. In this case a small number of points in the data could have an unusually large impact on the parameter estimates. These points may give a mistaken sense that our model has a great fit or conversely that there is not relationship between the variables.

4) Missing predictors. We can potentially improve the fit and predictive performance of our model by including other predictors. We will spend one lesson on this topic, but Math 378 will be devoted extensively to building these more complex models. In the case of multivariate linear regression, many of the diagnostic tools discussed next will also be applicable.

Residual plots

The assumptions about the error structure can be checked with residual plots. We have already done this, but let's review again and provide a little more depth.

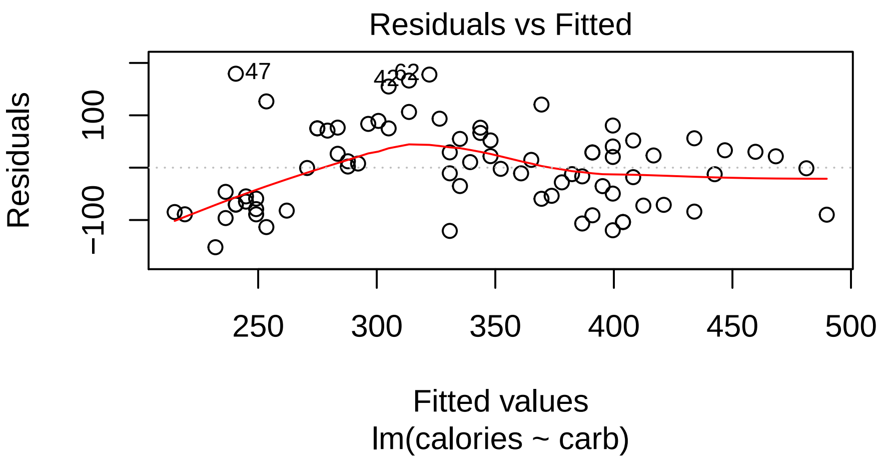
Applying the `plot()` function to an "lm" object provides several graphs that allow us visually evaluate a linear model's assumptions. There are actually six plots (selected by the `which` option) available:

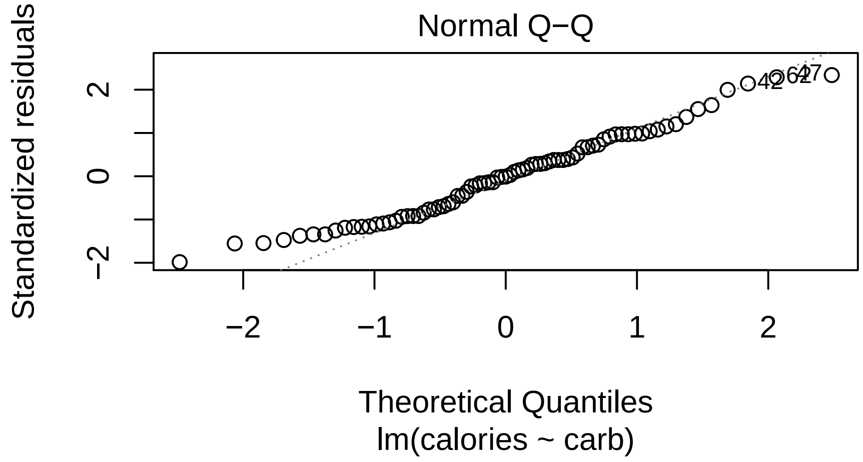
- a plot of residuals against fitted values,

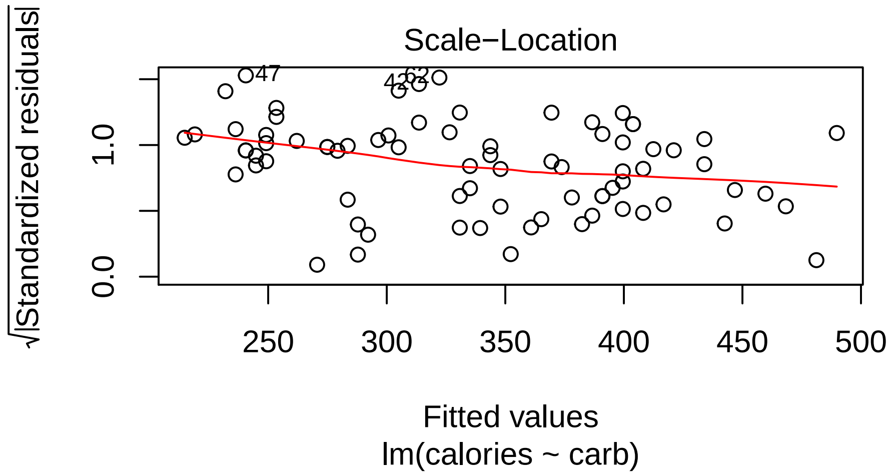
- a Scale-Location plot of $\sqrt{(|\text{residuals}|)}$ against fitted values,
- a Normal Q-Q plot, a plot of Cook's distances versus row labels,
- a plot of residuals against leverages,
- and a plot of Cook's distances against leverage/(1-leverage).

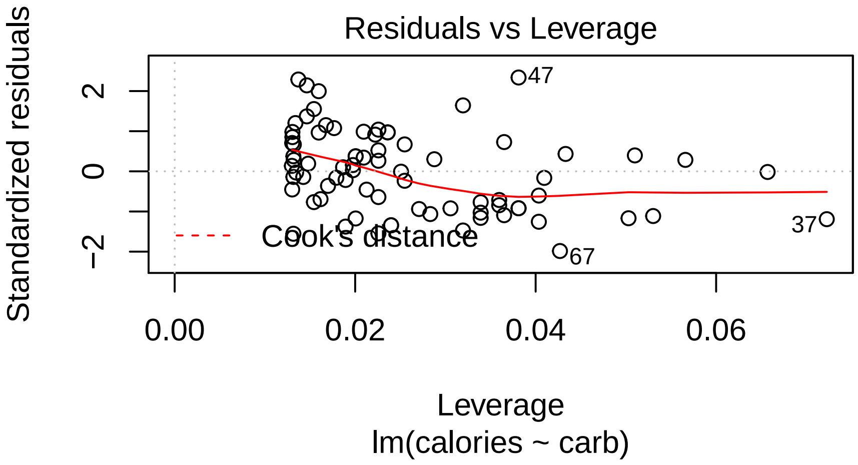
By default, the first three and 5 are provided by applying `plot()` to an "lm" object. To obtain all four at once, simply use `plot(my.model)`. Here are the four plots.

```r
plot(star_mod)
```




Normal Q−Q
Standardized residuals
−2
0
2
42
62
47
−2
−1
0
1
2
Theoretical Quantiles
lm(calories ~ carb)


Scale−Location
√|Standardized residuals|
0.0
1.0
47
42
62
250
300
350
400
450
500
Fitted values
lm(calories ~ carb)

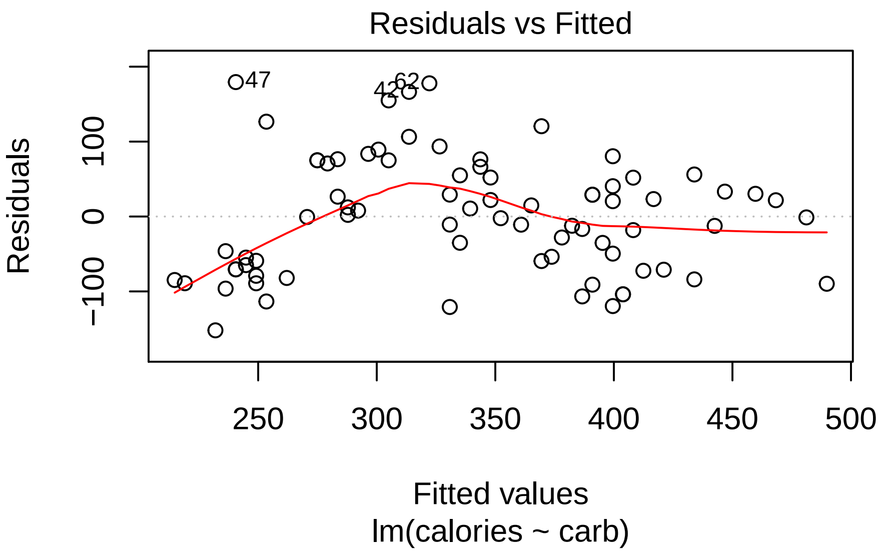


However, it's best to walk through each of these four plots in our Starbucks example.

Residuals vs Fitted

By providing the `which` option, we can select the plot we want.

```r
plot(star_mod,which = 1)
```



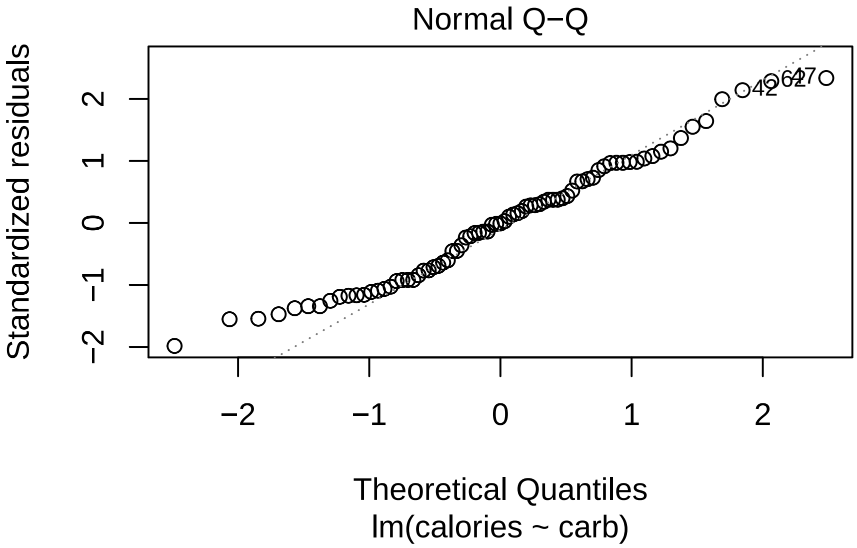
This plot assesses linearity of the model and homoscedasticity (constant variance). The red line in a smoothed estimate of the fitted values versus the residuals. Ideally, the red line should be centered around the dashed horizontal line. This would indicate that a linear fit is appropriate. Furthermore, the scatter around the dashed line should be relatively constant across the plot, homoskedasticity. In this case, it looks like there is some minor concern over linearity and non-constant error variance. We noted this earlier with the cluster of points in the lower left hand corner of the scatterplot.

Note: the points that are labeled are points with a high residual value. They are extreme. We will discuss outliers and leverage points shortly.

Normal Q-Q Plot

As it's name suggests, this plot evaluates the normality of the residuals. We have seen and used this plot several times in our course of study this semester. Remember is the number of data points is small, this plot has a greatly reduced effectiveness.

```
plot(star_mod,which = 2)
```



Along the y-axis are the actual standardized residuals. Along the x-axis is where those points should be if the residuals were actually normally distributed. Ideally, the dots should fall along the diagonal dashed line. In this case, it appears there is some skewness to the right. We can tell this because for the smaller residuals, they don't increase as they should to match a normal distribution, the points are above the line. This is concerning.

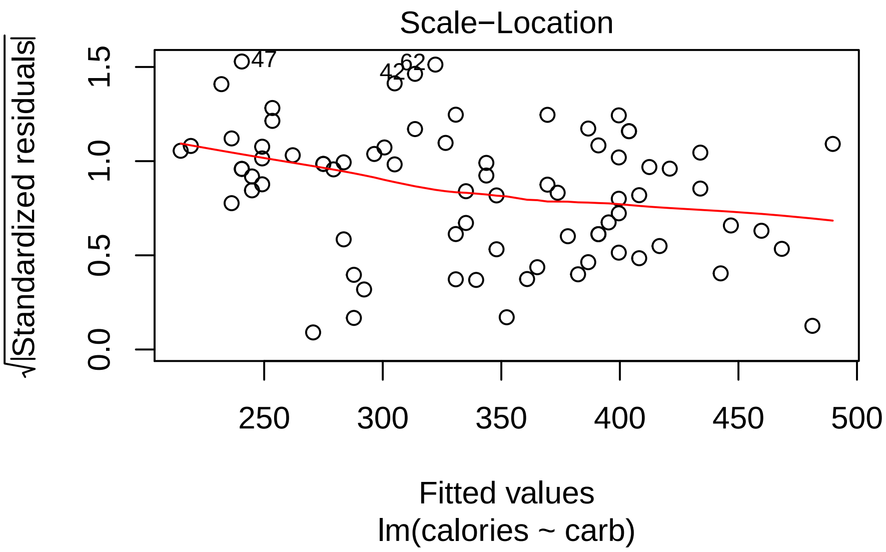
Scale-Location Plot

The scale-location plot is a better indicator of non-constant error variance. It is a plot of fitted values versus absolute value of the square root of the standardized residuals. A standardized residual is the residual divided by its standard deviation

$$e_i^{'} = \frac{e_i}{s}$$

This plot illustrated the the spread of the residuals over the entire range of the predictor. We are using fitted values because this will generalize well if we have more than one predictor.

```r
plot(star_mod,which=3)
```



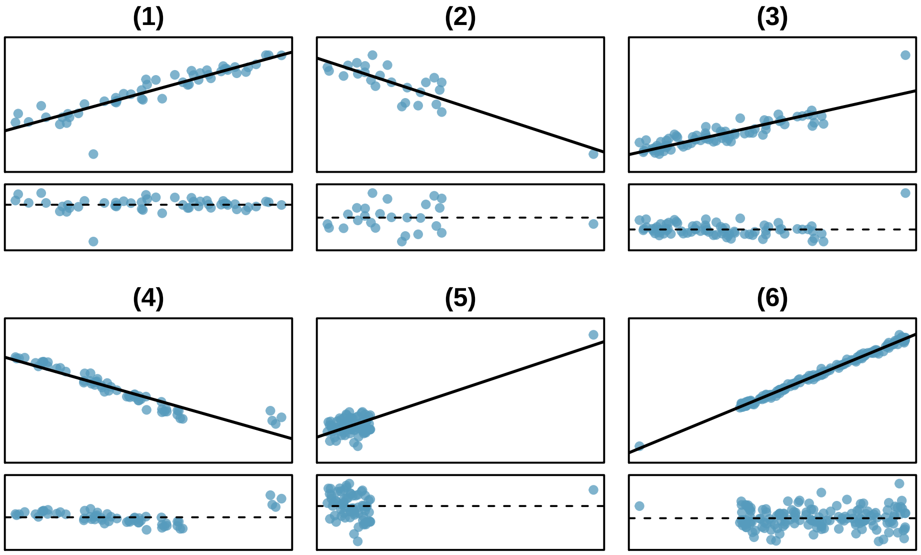
A straight horizontal red line indicates constant error variance. In this case, there is some indication error variance is higher for lower carb counts.

Outliers and leverage

Before discussing the last plot, we need to spend some time discussing outliers. Outliers in regression are observations that fall far from the "cloud" of points. These points are especially important because they can have a strong influence on the least squares line.

In regression, there are two types of outliers:
- An outlier in the response variable is one that is not predicted well by the model. This could either be a problem with the data or the model. The residuals for this outlier will be large in absolute value.
- An outlier in the explanatory variable. These are typically called **leverage points** because they can have a undue impact of the parameter estimates.

> **Exercise**:
> There are six plots shown in figure below along with the least squares line and residual plots. For each scatterplot and residual plot pair, identify any obvious outliers and note how they influence the least squares line. Recall that an outlier is any point that doesn't appear to belong with the vast majority of the other points.



(1) There is one outlier far from the other points, though it only appears to slightly influence the line. This is an outlier in the response and will have a large residual in magnitude.

(2) There is one outlier on the right, though it is quite close to the least squares line, which suggests it wasn't very influential although it is a leverage point.

(3) There is one point far away from the cloud, and this leverage point appears to pull the least squares line up on the right; examine how the line around the primary cloud doesn't appear to fit very well. This point has a higher influence on the estimated slope.

(4) There is a primary cloud and then a small secondary cloud of four outliers. The secondary cloud appears to be influencing the line somewhat strongly, making the least square line fit poorly almost everywhere. There might be an interesting explanation for the dual clouds, which is something that could be investigated.

(5) There is no obvious trend in the main cloud of points and the outlier on the right appears to largely control the slope of the least squares line. This point is an outlier in both the response and predictor. It is a highly influential point.

(6) There is one outlier in both the response and predictor far from the cloud, however, it falls quite close to the least squares line and does not appear to be very influential.

Examining the residual plots in the figure, you will probably find that there is some trend in the main clouds of (3) and (4). In these cases, the outliers influenced the slope of the least squares lines. In (5), data with no clear trend were assigned a line with a large trend simply due to one outlier!

Leverage
Points that fall horizontally away from the center of the cloud tend to pull harder on the line, so we call them points with **high leverage**.

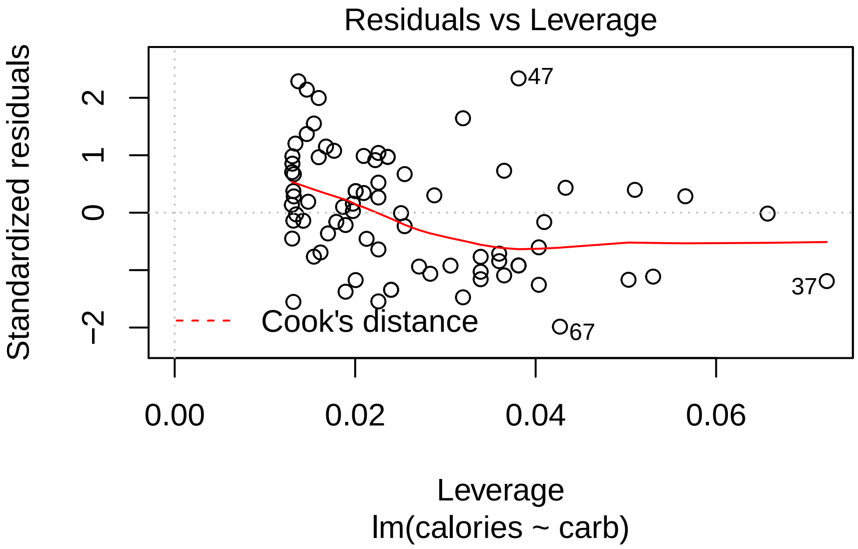
Points that fall horizontally far from the line are points of high leverage; these points can strongly influence the slope of the least squares line. If one of these high leverage points does appear to actually invoke its influence on the slope of the line – as in cases (3), (4), and (5) – then we call it an **influential point**. Usually we can say a point is influential if, had we fitted the line without it, the influential point would have been unusually far from the least squares line.

A point can be an outlier but not a leverage point as we have already dicussed. It is tempting to remove outliers from your data set. Don't do this without a very good reason. Models that ignore exceptional (and interesting) cases often perform poorly. For instance, if a financial firm ignored the largest market swings – the "outliers" – they would soon go bankrupt by making poorly thought-out investments.

Residuals vs Leverage Plot

The residuals vs leverage plot is a good way to identify influential observations. Sometimes, influential observations are representative of the population, but they could also indicate an error in recording data, or an otherwise unrepresentative outlier. It could be worth looking into these cases.

```
plot(star_mod,5)
```



This plot helps us to find influential cases, those leverage points that impact the estimated slope. Unlike the other plots, this time patterns are not relevant. We watch out for outlying values at the upper right corner or at the lower right corner. Those spots are the places where cases can be influential against a regression line. Look for cases outside of a dashed line, Cook's distance. When cases are outside of the Cook's distance (meaning they have high Cook's distance scores), the cases are influential to the regression results. The regression results will be altered if we exclude those cases. In this example, there are no points that tend to have undue influence.

What If Our Assumptions Are Violated

If the assumptions of the model are violated and/or we have influential points, a linear regression model with normality assumptions is not appropriate. Sometimes it is appropriate to transform the data (either response or predictor), so that the assumptions are met on the transformed data. Other times, it is appropriate to explore other models. There are entire courses on regression where blocks of material are devoted to diagnostics and transformations to reduce the impact of violations of assumptions. We will not go into these methods in this class. Instead, when confronted with clear violated assumptions, we will use resampling as a possible solution. We will learn about this in the next lesson because it does not assume normality in the residuals. This is a limited solution, but as this is an introductory class, this is an excellent first step.

File Creation Information

- File creation date: 2020-11-10
- Windows version: Windows 10 x64 (build 18362)
- R version 3.6.3 (2020-02-29)
- `mosaic` package version: 1.7.0
- `tidyverse` package version: 1.3.0