Chapter 8

Correlation and Regression

Learning objectives

After completing this topic, you should be able to:

select graphical displays that reveal the relationship between two continuous variables.

summarize model fit.

interpret model parameters, such as slope and R^2.

assess the model assumptions visually and numerically.

Achieving these goals contributes to mastery in these course learning outcomes:

1. organize knowledge.

5. define parameters of interest and hypotheses in words and notation.

6. summarize data visually, numerically, and descriptively.

8. use statistical software.

12. make evidence-based decisions.

8.1 Introduction

Suppose we select $n = 10$ people from the population of college seniors who plan to take the medical college admission test (MCAT) exam. Each takes the

test, is coached, and then retakes the exam. Let X_i be the pre-coaching score and let Y_i be the post-coaching score for the i^{th} individual, $i = 1, 2, \ldots, n$. There are several questions of potential interest here, for example: Are Y and X related (associated), and how? Does coaching improve your MCAT score? Can we use the data to develop a mathematical model (formula) for predicting post-coaching scores from the pre-coaching scores? These questions can be addressed using **correlation** and **regression** models.

The **correlation coefficient** is a standard measure of **association** or relationship between two features Y and X. Most scientists equate Y and X being correlated to mean that Y and X are associated, related, or **dependent** upon each other. However, correlation is only a measure of the strength of a **linear relationship.** For later reference, let ρ be the correlation between Y and X in the population and let r be the sample correlation. I define r below. The population correlation is defined analogously from population data.

Suppose each of n sampled individuals is measured on two quantitative characteristics called Y and X. The data are pairs of observations (X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n), where (X_i, Y_i) is the (X, Y) pair for the i^{th} individual in the sample. The sample correlation between Y and X, also called the **Pearson product moment correlation coefficient**, is

$$r = \frac{S_{XY}}{S_X S_Y} = \frac{\sum_i (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_i (X_i - \bar{X})^2 \sum_i (Y_i - \bar{Y})^2}},$$

where

$$S_{XY} = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{n - 1}$$

is the **sample covariance** between Y and X, and $S_Y = \sqrt{\sum_i (Y_i - \bar{Y})^2/(n-1)}$ and $S_X = \sqrt{\sum_i (X_i - \bar{X})^2/(n-1)}$ are the standard deviations for the Y and X samples.

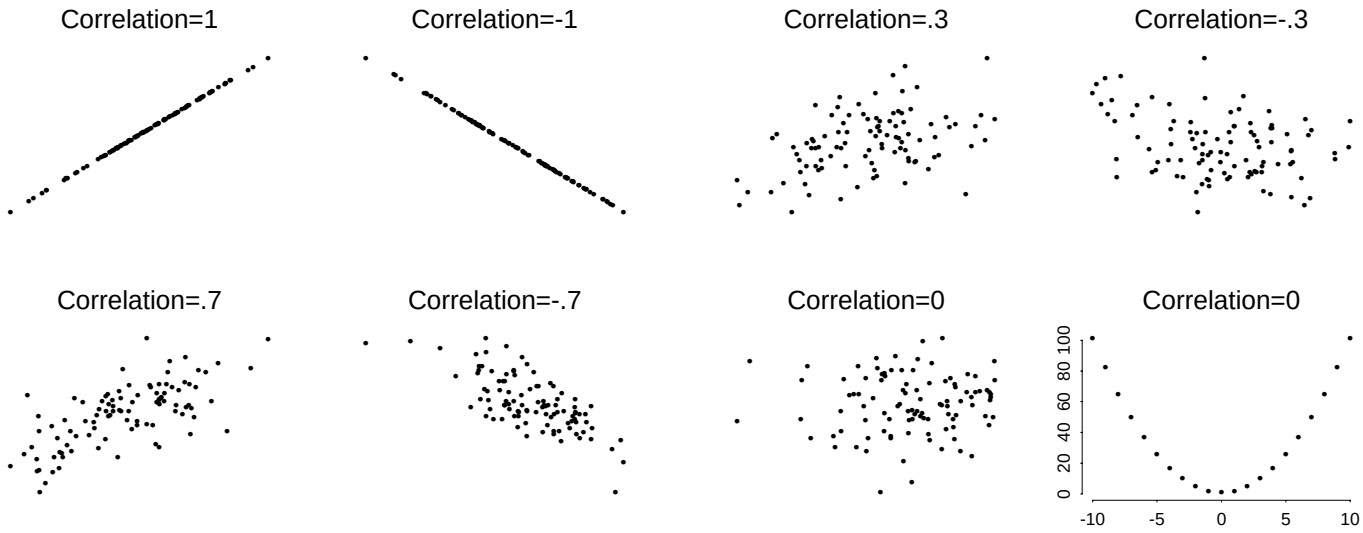
Important properties of r:

1. $-1 \leq r \leq 1$.

2. If Y_i tends to increase linearly with X_i then $r > 0$.
3. If Y_i tends to decrease linearly with X_i then $r < 0$.
4. If there is a perfect linear relationship between Y_i and X_i with a positive slope then $r = +1$.
5. If there is a perfect linear relationship between Y_i and X_i with a negative slope then $r = -1$.
6. The closer the points (X_i, Y_i) come to forming a straight line, the closer r is to ± 1.
7. The magnitude of r is unchanged if either the X or Y sample is transformed linearly (such as feet to inches, pounds to kilograms, Celsius to Fahrenheit).
8. The correlation does not depend on which variable is called Y and which is called X.

If r is near ± 1, then there is a strong linear relationship between Y and X in the sample. This suggests we might be able to accurately predict Y from X with a linear equation (i.e., linear regression). If r is near 0, there is a weak linear relationship between Y and X, which suggests that a linear equation provides little help for predicting Y from X. The pictures below should help you develop a sense about the size of r.

Note that $r = 0$ does not imply that Y and X are not related in the sample. It only implies they are not linearly related. For example, in the last plot $r = 0$ yet $Y_i = X_i^2$, exactly.



■ CLICKER Qs — Correlation coefficients STT.02.02.010 ■

■ CLICKER Qs — Strong correlation STT.02.02.020 ■

8.2 Logarithmic transformations

Logarithms[1] are useful for understanding data, partly because they allow numbers that vary by several orders of magnitude to be viewed on a common scale, and more importantly because they allow exponential and power-law relations to be transformed into linearity.

8.2.1 Log-linear and log-log relationships: amoebas, squares, and cubes

Suppose you have an amoeba that takes one hour to divide, and then the two amoebas each divide in one more hour, and so forth. What is the equation of the number of amoebas, y, as a function of time, x (in hours)? It can be written as $y = 2^x$ or, on the logarithmic scale, $\log(y) = (\log(2))x = 0.30x$.

[1]From: Gelman, Andrew and Deborah Nolan (2002). *Teaching statistics: A bag of tricks.* Oxford University Press.

Suppose you have the same example, but the amoeba takes three hours to divide at each step. Then the number of amoebas y after time x has the equation, $y = 2^{x/3} = (2^{1/3})^x = 1.26^x$ or, on the logarithmic scale, $\log(y) = (\log(1.26))x = 0.10x$. The slope of 0.10 is one-third the earlier slope of 0.30 because the population is growing at one-third the rate.

In the example of exponential growth of amoebas, y is logged while x remains the same. For power-law relations, it makes sense to log both x and y. How does the area of a square relate to its circumference (perimeter)? If the side of the cube has length L, then the area is L^2 and the circumference is $4L$; thus

$$\text{area} \;=\; (\text{circumference}/4)^2.$$

Taking the logarithm of both sides yields,

$$\begin{aligned} \log(\text{area}) &= 2(\log(\text{circumference}) - \log(4)) \\ \log(\text{area}) &= -1.20 + 2\log(\text{circumference}), \end{aligned}$$

a linear relation on the log-log scale.

What is the relation between the surface area and volume of a cube? In terms of the side length L, are $6L^2$ and L^3, respectively. On the original scale, this is

$$\text{surface area} \;=\; 6(\text{volume})^{2/3},$$

or, on the logarithmic scale,

$$\log(\text{surface area}) \;=\; \log(6) + (2/3)\log(\text{volume}).$$

Example: Log-linear transformation: world population Consider the world population from the year 0 to 2000. Compare the data in the table below with the plots on the original and logarithmic scales; again, the plots reveals the pattern much clearer than the table.

On the raw scale, all you can see is that the population has increased very fast recently. On the log scale, convex curvature is apparent — that is, the rate

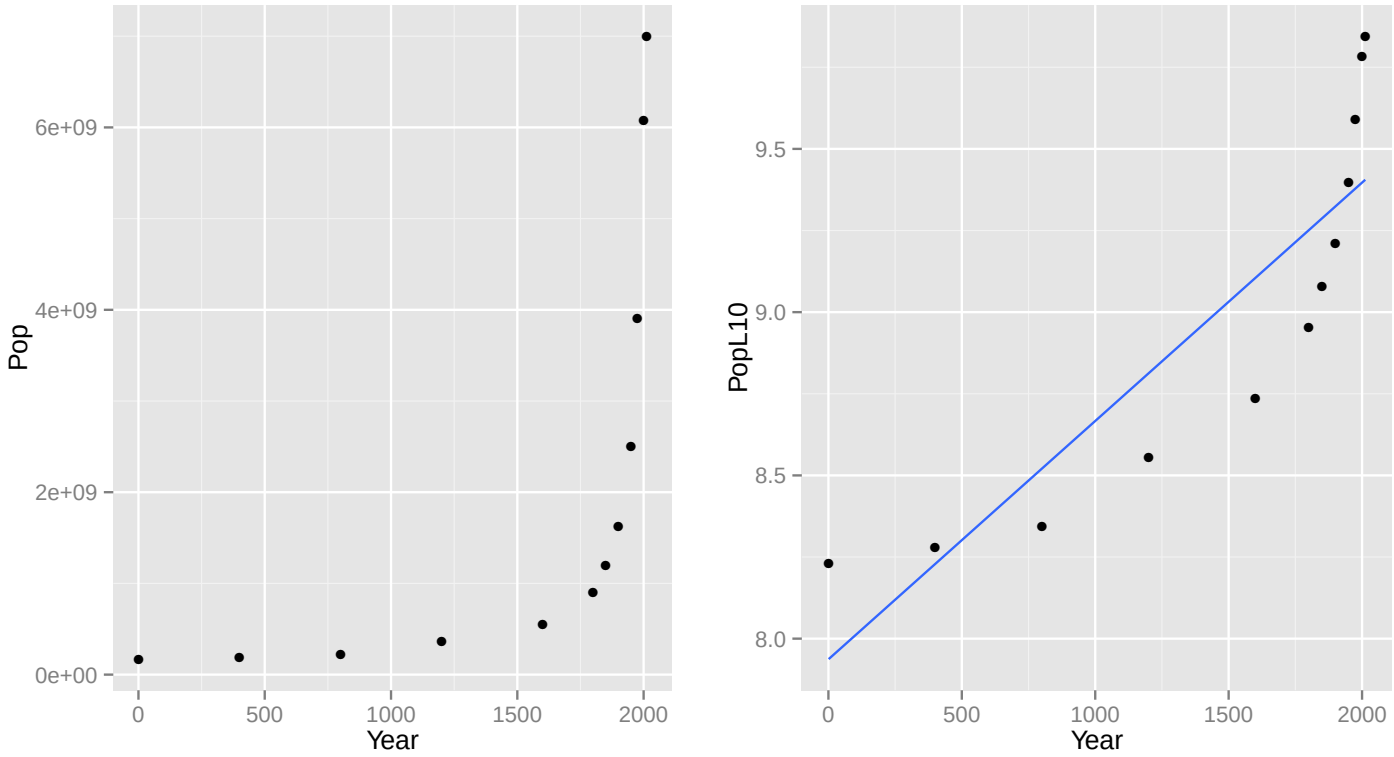
of increase has itself increased. On the logarithmic graph, the least-squares regression line is drawn. The estimated world population has been growing even faster than exponentially! What would you guess the population to be fore year 1400? Would you be surprised that it was 350 million? It is actually lower than the year 1200 population, because of plague and other factors. This is an illustration that even interpolation can sometimes go awry.

```
#### Example: Log-linear population growth
pop <- read.table(text="
Year Pop_M
   1   170
 400   190
 800   220
1200   360
1600   545
1800   900
1850  1200
1900  1625
1950  2500
1975  3900
2000  6080
2012  7000
", header=TRUE)
pop$Pop    <- 1e6 * pop$Pop_M  # convert to millions
pop$PopL10 <- log10(pop$Pop)

# calculate the residuals from a simple linear regression
lm.fit <- lm(PopL10 ~ Year, data = pop)
# include those residuals in the pop table
pop$Res <- residuals(lm.fit)
pop$R10 <- 10^residuals(lm.fit) # residuals on the original scale
```

	Year	Pop_M	Pop	PopL10	Res	R10
1	1	170	170000000	8.230	0.293	1.964
2	400	190	190000000	8.279	0.050	1.122
3	800	220	220000000	8.342	−0.178	0.663
4	1200	360	360000000	8.556	−0.256	0.554
5	1600	545	545000000	8.736	−0.368	0.428
6	1800	900	900000000	8.954	−0.296	0.505
7	1850	1200	1200000000	9.079	−0.208	0.619
8	1900	1625	1625000000	9.211	−0.113	0.771
9	1950	2500	2500000000	9.398	0.038	1.091
10	1975	3900	3900000000	9.591	0.213	1.631
11	2000	6080	6080000000	9.784	0.387	2.439
12	2012	7000	7000000000	9.845	0.440	2.752

```
library(ggplot2)

p1 <- ggplot(pop, aes(x = Year, y = Pop))
p1 <- p1 + geom_point()
#print(p1)

p2 <- ggplot(pop, aes(x = Year, y = PopL10))
p2 <- p2 + geom_point()
```

```
p2 <- p2 + geom_smooth(method = lm, se = FALSE)
#print(p2)

library(gridExtra)
grid.arrange(p1, p2, nrow=1)
```



When using data of this nature, consider the source of the population numbers. How would you estimate the population of the world in the year 1?

Example: Log-log transformation: metabolic rates A rich source of examples when covering log-log transformations are biological scaling relations[2]. The relationship[3] between body mass (M, g) and basal metabolic rate (BMR, ml of O_2 per h (similar to Watts?)) for mammalian orders for selected data are summarized in plots below, both on the original and log-log scales. The linear regression summarizes the dark points, the mean for each of the species groups, and the colored points are individual species. The curved regression is found by inverting the linear regression onto the original scale. The third plot displays

[2]One of the world experts in allometric scaling is Prof. Jim Brown, UNM Biology, `http://biology.unm.edu/jhbrown`

[3]White and Seymour (2003) PNAS, 10.1073/pnas.0436428100

the log axes with values on the original scale.

```
# http://www.ncbi.nlm.nih.gov/pmc/articles/PMC153045/
# Supp:
# http://www.ncbi.nlm.nih.gov/pmc/articles/PMC153045/bin/pnas_0436428100_index.html
# Supporting information for White and Seymour (2003)
#   Proc. Natl. Acad. Sci. USA, 10.1073/pnas.0436428100

library(gdata)
fn <- "data/ADA1_notes_08_data_log-logScaling_BodyMassMetabolicRate_2003_WhiteSeymour.xlsx"
bm.bmr <- read.xls(fn, skip = 4, stringsAsFactors = TRUE)
bm.bmr$Log10BodyMass <- log10(bm.bmr$BodyMass)
bm.bmr$Log10BaseMetRate <- log10(bm.bmr$BaseMetRate)

# remove a very strange group
bm.bmr <- subset(bm.bmr, !(Group == "Artiodactyla 7"))
str(bm.bmr)
```

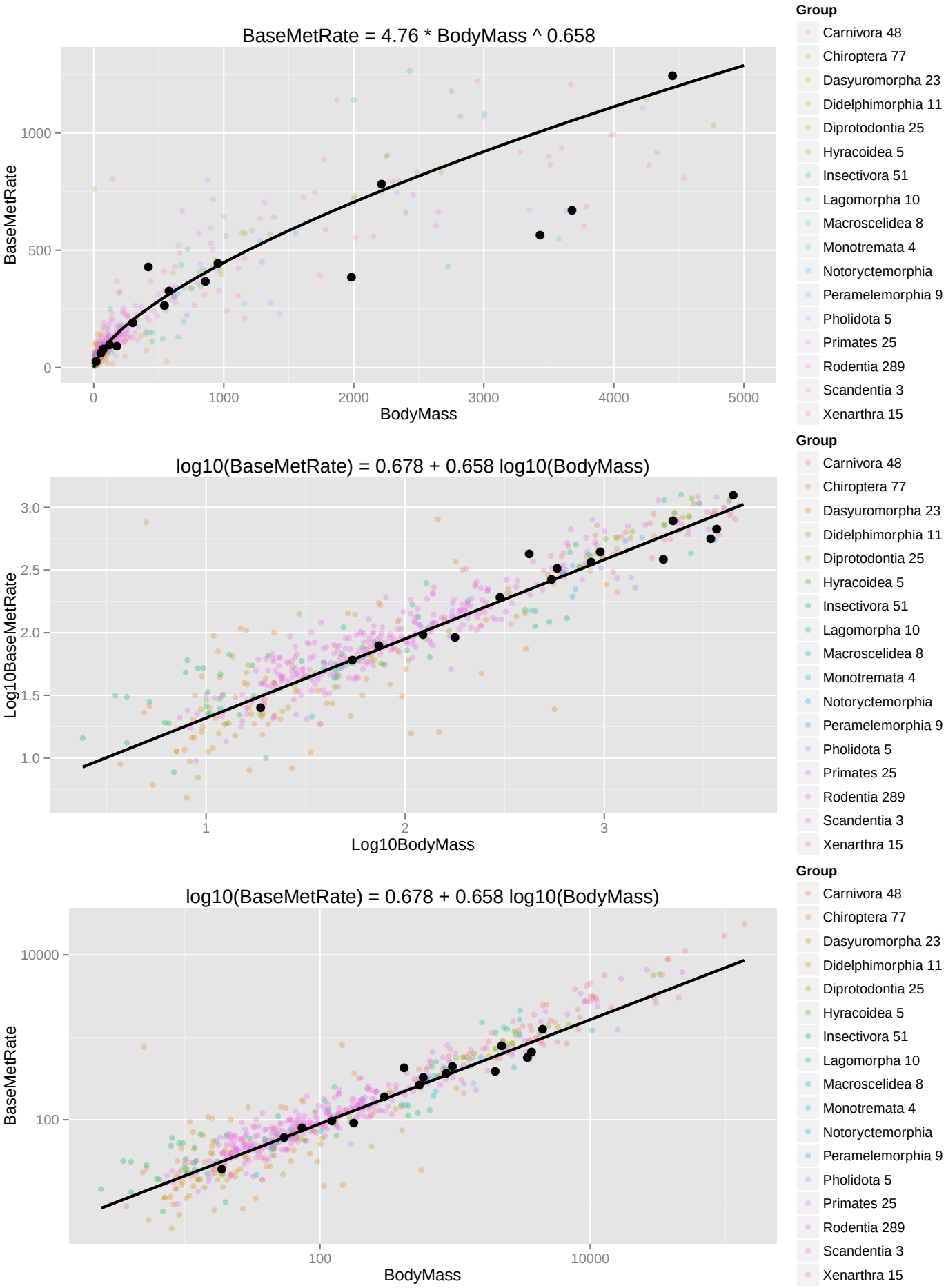
```
## 'data.frame': 634 obs. of  9 variables:
##  $ Group           : Factor w/ 18 levels "Artiodactyla 7",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ Genus           : Factor w/ 88 levels "","Acrobatidae",..: 1 12 12 12 12 12 12 12 12 31 ...
##  $ Species         : Factor w/ 621 levels "","2n = 52","2n = 54",..: 1 22 67 68 79 206 614 615 616 8 ...
##  $ BodyMass        : num  4452 3600 10000 7720 5444 ...
##  $ T               : num  37.5 38.6 37 38 38.2 38.8 38 38.7 NA 39 ...
##  $ BaseMetRate     : num  1244 1374 2687 3860 1524 ...
##  $ Ref             : Factor w/ 239 levels "","1","10","100",..: 1 230 3 15 24 37 3 50 61 72 ...
##  $ Log10BodyMass   : num  3.65 3.56 4 3.89 3.74 ...
##  $ Log10BaseMetRate: num  3.09 3.14 3.43 3.59 3.18 ...
```

```
# log-log scale linear regression
lm.fit <- lm(Log10BaseMetRate ~ Log10BodyMass, data = bm.bmr)
# coefficients for regression line
coef(lm.fit)
```

```
##   (Intercept) Log10BodyMass
##     0.6775600     0.6575572
```

```
library(ggplot2)

p1 <- ggplot(subset(bm.bmr, (Genus == "")), aes(x = BodyMass, y = BaseMetRate))
p1 <- p1 + geom_point(data = subset(bm.bmr, !(Genus == "")), aes(colour = Group), alpha = 0.2)
p1 <- p1 + geom_point(size = 3)
  # Using a custom function
  f.org.scale <- function(BodyMass) { 10^coef(lm.fit)[1] * BodyMass ^ coef(lm.fit)[2]}
p1 <- p1 + stat_function(fun = f.org.scale, size = 1)
p1 <- p1 + labs(title = paste("BaseMetRate = ", signif(10^coef(lm.fit)[1], 3), " * ", "BodyMass ^ ", signif(coef(lm.fit)[2], 3), sep = "
p1 <- p1 + scale_y_continuous(limits=c(0, 1300))
p1 <- p1 + scale_x_continuous(limits=c(0, 5000))
#print(p1)

p2 <- ggplot(subset(bm.bmr, (Genus == "")), aes(x = Log10BodyMass, y = Log10BaseMetRate))
p2 <- p2 + geom_point(data = subset(bm.bmr, !(Genus == "")), aes(colour = Group), alpha = 0.3)
p2 <- p2 + geom_point(size = 3)
p2 <- p2 + geom_smooth(method = lm, se = FALSE, fullrange = TRUE, size = 1, colour = "black")
p2 <- p2 + labs(title = paste("log10(BaseMetRate) = ", signif(coef(lm.fit)[1], 3), " + ", signif(coef(lm.fit)[2], 3), " log10(BodyMass)"
p2 <- p2 + scale_y_continuous(limits=c(NA, log10(1300)))
p2 <- p2 + scale_x_continuous(limits=c(NA, log10(5000)))
#print(p2)

p3 <- ggplot(subset(bm.bmr, (Genus == "")), aes(x = BodyMass, y = BaseMetRate))
p3 <- p3 + geom_point(data = subset(bm.bmr, !(Genus == "")), aes(colour = Group), alpha = 0.3)
p3 <- p3 + geom_point(size = 3)
p3 <- p3 + geom_smooth(method = lm, se = FALSE, fullrange = TRUE, size = 1, colour = "black")
p3 <- p3 + labs(title = paste("log10(BaseMetRate) = ", signif(coef(lm.fit)[1], 3), " + ", signif(coef(lm.fit)[2], 3), " log10(BodyMass)"
p3 <- p3 + scale_y_log10()#limits=c(NA, log10(1300)))
p3 <- p3 + scale_x_log10()#limits=c(NA, log10(5000)))
#print(p3)

library(gridExtra)
grid.arrange(p1, p2, p3, ncol=1)
```

```
## Warning: Removed 56 rows containing missing values (geom_point).
## Warning: Removed 5 rows containing missing values (geom_point).
## Warning: Removed 5 rows containing missing values (stat_smooth).
## Warning: Removed 56 rows containing missing values (geom_point).
## Warning: Removed 5 rows containing missing values (geom_point).
## Warning: Removed 5 rows containing missing values (stat_smooth).
## Warning: Removed 1 rows containing missing values (geom_point).
## Warning: Removed 5 rows containing missing values (geom_point).
```


BaseMetRate = 4.76 * BodyMass ^ 0.658
BaseMetRate
BodyMass
0
500
1000
0
1000
2000
3000
4000
5000
log10(BaseMetRate) = 0.678 + 0.658 log10(BodyMass)
Log10BaseMetRate
Log10BodyMass
1.0
1.5
2.0
2.5
3.0
1
2
3
log10(BaseMetRate) = 0.678 + 0.658 log10(BodyMass)
BaseMetRate
BodyMass
100
10000
100
10000
Group
Carnivora 48
Chiroptera 77
Dasyuromorpha 23
Didelphimorphia 11
Diprotodontia 25
Hyracoidea 5
Insectivora 51
Lagomorpha 10
Macroscelidea 8
Monotremata 4
Notoryctemorphia
Peramelemorphia 9
Pholidota 5
Primates 25
Rodentia 289
Scandentia 3
Xenarthra 15

The table below provides predictions over a rnage of scales. Note that the smallest mammel in this dataset is about 2.4 grams. A 5-gram mammal uses about 13.7 Watts, so 1000 5-gram mammals use about 13714 Watts. Whereas, one 5000-gram mammal uses 1287 Watts. Thus, larger mammals give off less heat than the equivalent weight of many smaller mammals.

```
pred.bm.bmr <- data.frame(BodyMass = 5 * c(1, 10, 100, 1000))
pred.bm.bmr$Log10BodyMass <- log10(pred.bm.bmr$BodyMass)
pred.bm.bmr$Log10BaseMetRate <- predict(lm.fit, pred.bm.bmr)
pred.bm.bmr$BaseMetRate <- 10^pred.bm.bmr$Log10BaseMetRate

tab.pred <- subset(pred.bm.bmr, select = c("BodyMass", "BaseMetRate"))
```

	BodyMass	BaseMetRate
1	5	13.71
2	50	62.33
3	500	283.33
4	5000	1287.79

We want to focus on the slope in the log-log plot, which is the exponent in original scale plot. On the log scale we have

$$\log(\text{BaseMetRate}) \;=\; 0.678 + 0.658\log(\text{BodyMass}).$$

To interpret the slope, for each unit increase in (predictor, x-variable) log(BodyMass), the expected increase in (response, y-variable) log(BaseMetRate) is 4.55. By exponentiating on both sides, the expression on the original scale is

$$\begin{aligned}\text{BaseMetRate} &= 10^{0.678} \times \text{BodyMass}^{0.658} \\ &= 4.76 \times \text{BodyMass}^{0.658}.\end{aligned}$$

For example, if you multiply body mass by 10, then you multiply metabolic rate by $10^{0.658} = 4.55$. If you multiply body mass by 100, then you multiply metabolic rate by $100^{0.658} = 20.7$, and so forth. The relation between metabolic rate and body mass is less than linear (that is, the exponent 0.658 is less than 1.0, and the line in the original-scal plot curves downward, not upward), which implies that the equivalent mass of small mammals gives off more heat, and the equivalent mass of large mammals gives off less heat.

This seems related to the general geometrical relation that surface area and volume are proportional to linear dimension to the second and third power, respectively, and thus surface area should be proportional to volume to the 2/3 power. Heat produced by a mammal is emitted from its surface, and it would thus be reasonable to suspect metabolic rate to be proportional to the 2/3 power of body mass. Biologists have considered whether the empirical slope is closer to 3/4 or 2/3; the important thing here is to think about log transformations and power laws (and have a chat with Jim Brown or someone from his lab at UNM for the contextual details). As an aside, something not seen from this plot is that males tend to be above the line and females below the line.

8.3 Testing that $\rho = 0$

Suppose you want to test $H_0 : \rho = 0$ against $H_A : \rho \neq 0$, where ρ is the population correlation between Y and X. This test is usually interpreted as a test of no association, or relationship, between Y and X in the population. Keep in mind, however, that ρ measures the strength of a **linear** relationship.

The standard test of $H_0 : \rho = 0$ is based on the magnitude of r. If we let

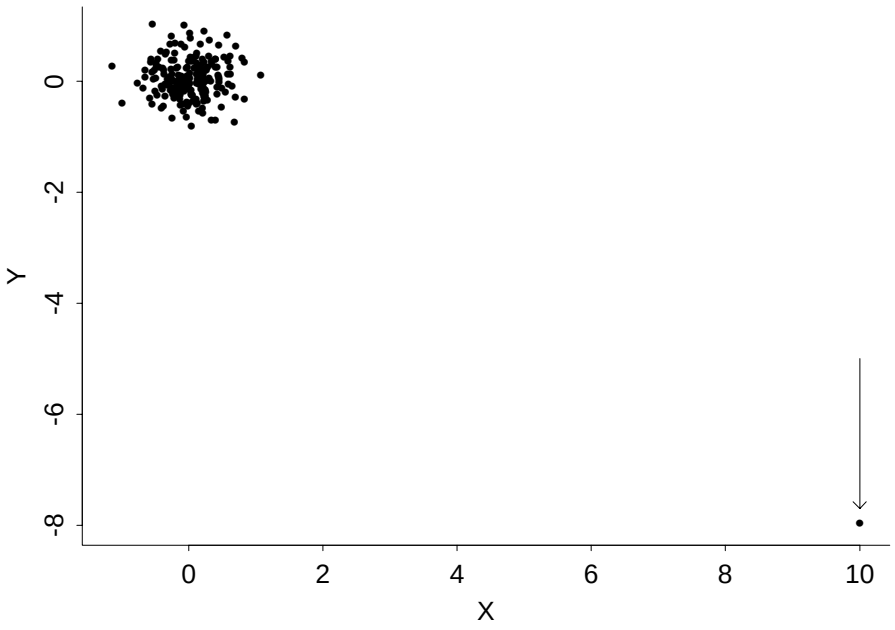
$$t_s = r\sqrt{\frac{n-2}{1-r^2}},$$

then the test rejects H_0 in favor of H_A if $|t_s| \geq t_{\text{crit}}$, where t_{crit} is the two-sided test critical value from a t-distribution with $df = n - 2$. The p-value for the test is the area under the t-curve outside $\pm t_s$ (i.e., two-tailed test p-value).

This test assumes that the data are a random sample from a **bivariate normal population** for $(X,\ Y)$. This assumption implies that all linear combinations of X and Y, say $aX + bY$, are normal. In particular, the (marginal) population frequency curves for X and Y are normal. At a minimum, you should make boxplots of the X and Y samples to check marginal normality. For large-sized samples, a plot of Y against X should be roughly an elliptical cloud, with the density of the points decreasing as the points move away from the center of the cloud.

8.3.1 The Spearman Correlation Coefficient

The Pearson correlation r can be highly influenced by outliers in one or both samples. For example, $r \approx -1$ in the plot below. If you delete the one extreme case with the largest X and smallest Y value then $r \approx 0$. The two analyses are contradictory. The first analysis (ignoring the plot) suggests a strong linear relationship, whereas the second suggests the lack of a linear relationship. I will not strongly argue that you should (must?) delete the extreme case, but I am concerned about any conclusion that depends heavily on the presence of a single observation in the data set.



Spearman's rank correlation coefficient r_S is a sensible alternative to r when normality is unreasonable or outliers are present. Most books give a computational formula for r_S. I will verbally describe how to compute r_S. First, order the X_is and assign them ranks. Then do the same for the Y_is and replace the original data pairs by the pairs of ranked values. The Spearman rank correlation is the Pearson correlation computed from the pairs of ranks.

The Spearman correlation r_S estimates the **population rank correlation coefficient**, which is a measure of the strength of linear relationship between population ranks. The Spearman correlation, as with other rank-based methods, is not sensitive to the presence of outliers in the data (or any information about the marginal distribution of X or Y). In the plot above, $r_S \approx 0$ whether the unusual point is included or excluded from the analysis. In samples without unusual observations and a linear trend, you often find that the Spearman and Pearson correlations are similar, $r_S \approx r$.

An important point to note is that the magnitude of the Spearman correlation does not change if either X or Y or both are transformed (monotonically). Thus, if r_S is noticeably greater than r, a transformation of the data might provide a stronger linear relationship.

Example: Blood loss Eight patients underwent a thyroid operation. Three variables were measured on each patient: weight in kg, time of operation in minutes, and blood loss in ml. The scientists were interested in the factors that influence blood loss.

```
#### Example: Blood loss
thyroid <- read.table(text="
weight time blood_loss
  44.3  105        503
  40.6   80        490
  69.0   86        471
  43.7  112        505
  50.3  109        482
  50.2  100        490
  35.4   96        513
  52.2  120        464
", header=TRUE)

# show the structure of the data.frame
str(thyroid)
```

```
## 'data.frame': 8 obs. of  3 variables:
##  $ weight    : num  44.3 40.6 69 43.7 50.3 50.2 35.4 52.2
##  $ time      : int  105 80 86 112 109 100 96 120
##  $ blood_loss: int  503 490 471 505 482 490 513 464
```

```
# display the data.frame
#thyroid
```

	weight	time	blood_loss
1	44.3	105	503
2	40.6	80	490
3	69.0	86	471
4	43.7	112	505
5	50.3	109	482
6	50.2	100	490
7	35.4	96	513
8	52.2	120	464

Below, we calculate the Pearson correlations between all pairs of variables (left), as well as the p-values (right) for testing whether the correlation is equal to zero.

```
p.corr <- cor(thyroid);
#p.corr

# initialize pvalue table with dim names
p.corr.pval <- p.corr;
for (i1 in 1:ncol(thyroid)) {
  for (i2 in 1:ncol(thyroid)) {
    p.corr.pval[i1,i2] <- cor.test(thyroid[, i1], thyroid[, i2])$p.value
  }
}
#p.corr.pval
```

	weight	time	blood_loss
weight	1.000	−0.066	−0.772
time	−0.066	1.000	−0.107
blood_loss	−0.772	−0.107	1.000

	weight	time	blood_loss
weight	0.0000	0.8761	0.0247
time	0.8761	0.0000	0.8003
blood_loss	0.0247	0.8003	0.0000

Similarly, we calculate the Spearman (rank) correlation table (left), as well as the p-values (right) for testing whether the correlation is equal to zero.

```
s.corr <- cor(thyroid, method = "spearman");
#s.corr

# initialize pvalue table with dim names
s.corr.pval <- p.corr;
for (i1 in 1:ncol(thyroid)) {
  for (i2 in 1:ncol(thyroid)) {
    s.corr.pval[i1,i2] <- cor.test(thyroid[, i1], thyroid[, i2],
                                   method = "spearman")$p.value
  }
}
## Warning in cor.test.default(thyroid[, i1], thyroid[, i2], method = "spearman"):  Cannot
compute exact p-value with ties
```

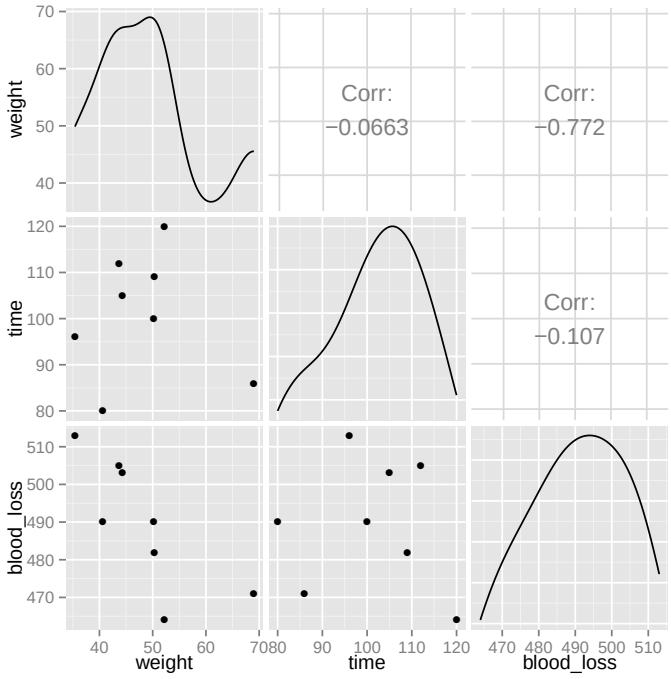
```
## Warning in cor.test.default(thyroid[, i1], thyroid[, i2], method = "spearman"): Cannot
compute exact p-value with ties
## Warning in cor.test.default(thyroid[, i1], thyroid[, i2], method = "spearman"): Cannot
compute exact p-value with ties
## Warning in cor.test.default(thyroid[, i1], thyroid[, i2], method = "spearman"): Cannot
compute exact p-value with ties
## Warning in cor.test.default(thyroid[, i1], thyroid[, i2], method = "spearman"): Cannot
compute exact p-value with ties
#s.corr.pval
```

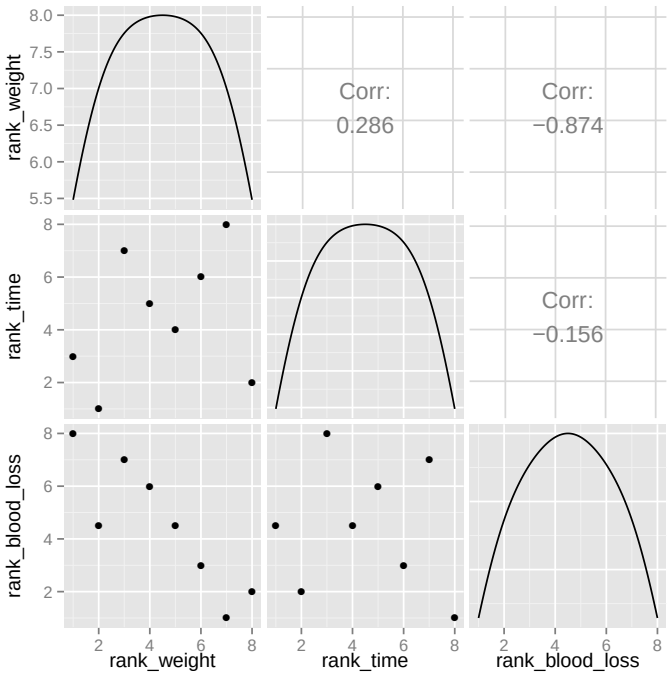
	weight	time	blood_loss
weight	1.000	0.286	−0.874
time	0.286	1.000	−0.156
blood_loss	−0.874	−0.156	1.000

	weight	time	blood_loss
weight	0.0000	0.5008	0.0045
time	0.5008	0.0000	0.7128
blood_loss	0.0045	0.7128	0.0000

Here are scatterplots for the original data and the ranks of the data using `ggpairs()` from the `GGally` package with `ggplot2`.

```
# Plot the data using ggplot
library(ggplot2)
library(GGally)
p1 <- ggpairs(thyroid[,1:3])
print(p1)


thyroid$rank_weight     <- rank(thyroid$weight    )
thyroid$rank_time       <- rank(thyroid$time      )
thyroid$rank_blood_loss <- rank(thyroid$blood_loss)

p2 <- ggpairs(thyroid[,4:6])
print(p2)
```





Comments:

1. (Pearson correlations). Blood loss tends to decrease linearly as weight increases, so r should be negative. The output gives $r = -0.77$. There is not much of a linear relationship between blood loss and time, so r should be close to 0. The output gives $r = -0.11$. Similarly, weight and time have a weak negative correlation, $r = -0.07$.
2. The Pearson and Spearman correlations are fairly consistent here. Only the correlation between blood loss and weight is significantly different from zero at the $\alpha = 0.05$ level (the p-values are given below the correlations).
3. (Spearman p-values) R gives the correct p-values. Calculating the p-value using the Pearson correlation on the ranks is not correct, strictly speaking.

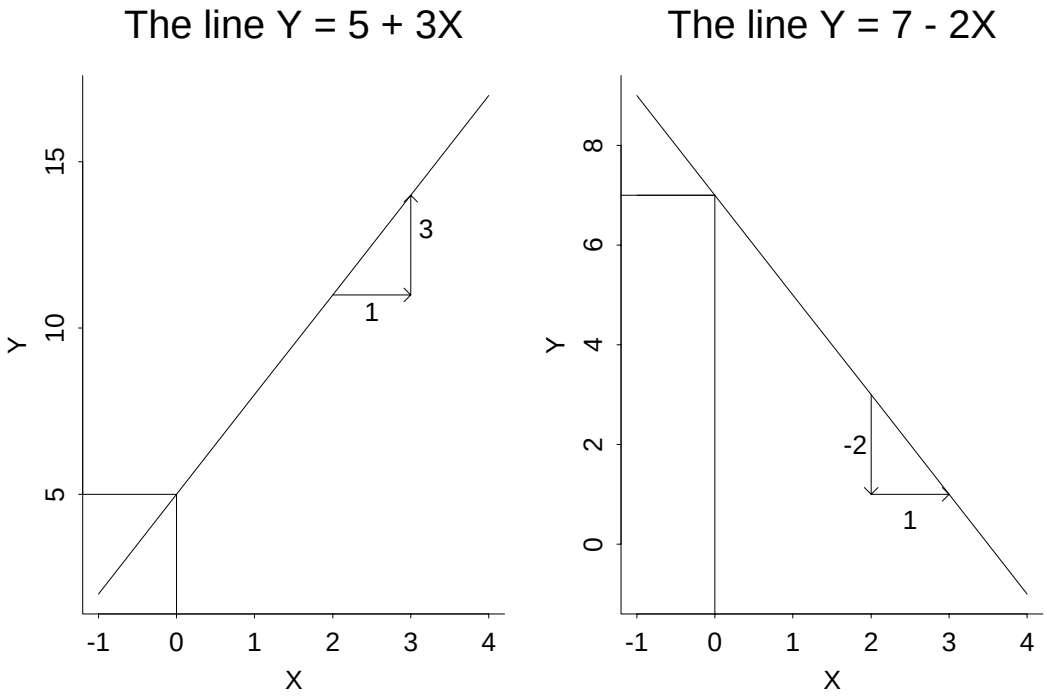
■ CLICKER Qs — Coney Island STT.02.02.050 ■

8.4 Simple Linear Regression

In linear regression, we are interested in developing a linear equation that best summarizes the relationship in a sample between the **response variable** Y and the **predictor variable** (or **independent variable**) X. The equation is also used to predict Y from X. The variables are not treated symmetrically in regression, but the appropriate choice for the response and predictor is usually apparent.

8.4.1 Linear Equation

If there is a perfect linear relationship between Y and X then $Y = \beta_0 + \beta_1 X$ for some β_0 and β_1, where β_0 is the Y-intercept and β_1 is the slope of the line. Two plots of linear relationships are given below. The left plot has $\beta_0 = 5$ and $\beta_1 = 3$. The slope is positive, which indicates that Y increases linearly when X increases. The right plot has $\beta_0 = 7$ and $\beta_1 = -2$. The slope is negative, which indicates that Y decreases linearly when X increases.



■ CLICKER Qs — Equation STT.02.03.010 ■

8.4.2 Least Squares

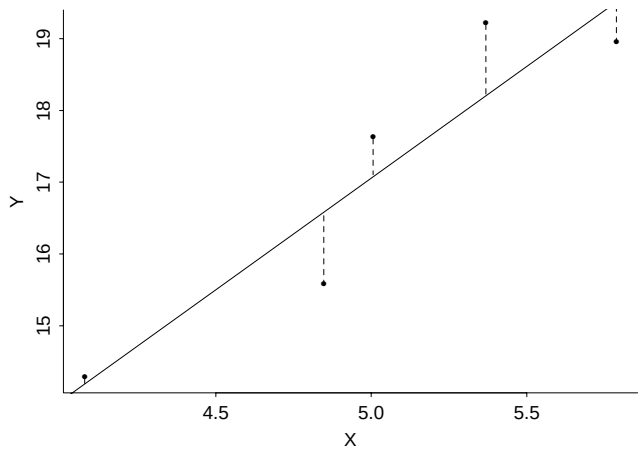
Data rarely, if ever, fall on a straight line. However, a straight line will often describe the **trend** for a set of data. Given a data set, (X_i, Y_i), $i = 1, \ldots, n$, with a **linear trend**, what linear equation "best" summarizes the observed relationship between Y and X? There is no universally accepted definition of "best", but many researchers accept the **Least Squares** line (LS line) as a reasonable summary.

Mathematically, the LS line chooses the values of β_0 and β_1 that minimize

$$\sum_{i=1}^{n} \{Y_i - (\beta_0 + \beta_1 X_i)\}^2$$

over all possible choices of β_0 and β_1. These values can be obtained using calculus. Rather than worry about this calculation, note that the LS line makes

the sum of squared (vertical) deviations between the responses Y_i and the line as small as possible, over all possible lines. The LS line goes through the mean point, $(\bar{X}, \bar{Y})$, which is typically in the "the heart" of the data, and is often closely approximated by an eye-ball fit to the data.



The equation of the LS line is

$$\hat{y} = b_0 + b_1 X$$

where the intercept b_0 satisfies

$$b_0 = \bar{Y} - b_1 \bar{X}$$

and the slope is

$$b_1 = \frac{\sum_i (Y_i - \bar{Y})(X_i - \bar{X})}{\sum_i (X_i - \bar{X})^2} = r \frac{S_Y}{S_X}.$$

As before, r is the Pearson correlation between Y and X, whereas S_Y and S_X are the sample standard deviations for the Y and X samples, respectively. The **sign of the slope** and the **sign of the correlation** are **identical** (i.e., + correlation implies + slope).

Special symbols b_0 and b_1 identify the LS intercept and slope to distinguish the LS line from the generic line $Y = \beta_0 + \beta_1 X$. You should think of $\hat{Y}$ as the **fitted value** at X, or the value of the LS line at X.

Fit a regression for the equation estimates from `summary()`. Note that we'll reuse the output of `lm()` over and over again.

```
lm.blood.wt <- lm(blood_loss ~ weight, data = thyroid)
lm.blood.wt
```

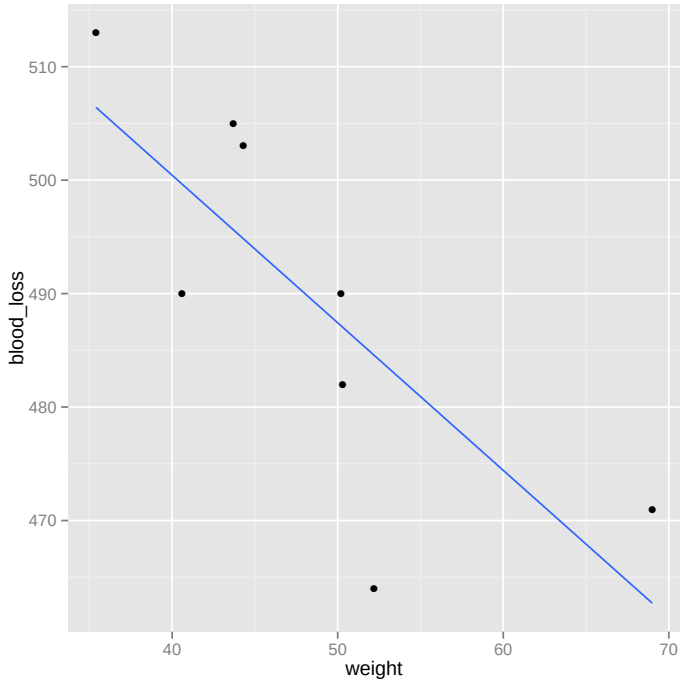
```
##
## Call:
## lm(formula = blood_loss ~ weight, data = thyroid)
##
## Coefficients:
## (Intercept)       weight
##       552.4         -1.3
```

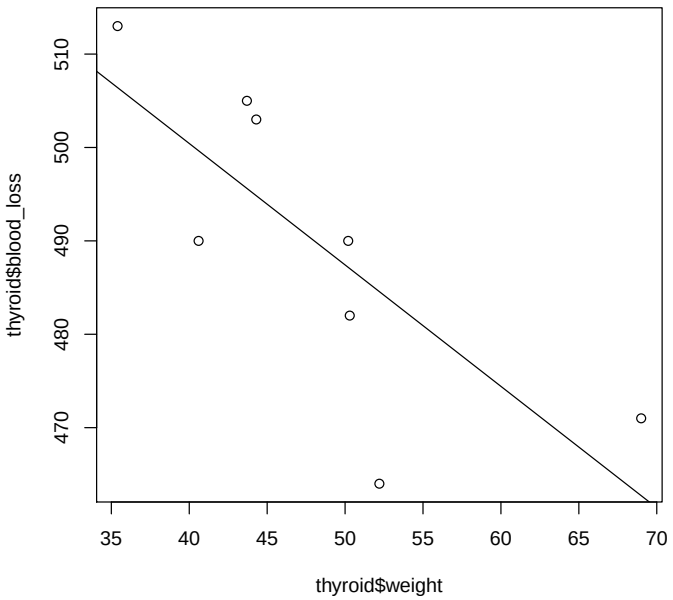
```
# use summary() to get t-tests of parameters (slope, intercept)
summary(lm.blood.wt)
```

```
##
## Call:
## lm(formula = blood_loss ~ weight, data = thyroid)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.565  -6.189   4.712   8.192   9.382
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 552.4420    21.4409   25.77 2.25e-07 ***
## weight       -1.3003     0.4364   -2.98   0.0247 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.66 on 6 degrees of freedom
## Multiple R-squared:  0.5967,Adjusted R-squared:  0.5295
## F-statistic: 8.878 on 1 and 6 DF,  p-value: 0.02465
```

Create a scatterplot with regression fit.

```
# ggplot: Plot the data with linear regression fit and confidence bands
library(ggplot2)
p <- ggplot(thyroid, aes(x = weight, y = blood_loss))
p <- p + geom_point()
p <- p + geom_smooth(method = lm, se = FALSE)
print(p)


# Base graphics: Plot the data with linear regression fit and confidence bands
# scatterplot
plot(thyroid$weight, thyroid$blood_loss)
# regression line from lm() fit
abline(lm.blood.wt)
```





For the **thyroid operation data** with Y = Blood loss in ml and X = Weight in kg, the LS line is $\hat{Y} = 552.44 - 1.30X$, or Predicted Blood Loss = $552.44 - 1.30$ Weight. For an $86kg$ individual, the Predicted Blood Loss = $552.44 - 1.30 \times 86 = 440.64ml$.

The LS regression coefficients for this model are interpreted as follows. The intercept b_0 is the predicted blood loss for a 0 kg individual. The intercept has no meaning here. The slope b_1 is the predicted increase in blood loss for each additional kg of weight. The slope is -1.30, so the predicted *decrease* in blood loss is 1.30 ml for each increase of 1 kg in weight.

Any fitted linear relationship holds only approximately and does not necessarily extend outside the range of the data. In particular, nonsensical predicted blood losses of less than zero are obtained at very large weights outside the range of data.

8.5 ANOVA Table for Regression

The LS line minimizes

$$\sum_{i=1}^{n} \{Y_i - (\beta_0 + \beta_1 X_i)\}^2$$

over all choices for β_0 and β_1. Inserting the LS estimates b_0 and b_1 into this expression gives

$$\text{Residual Sums of Squares} = \sum_{i=1}^{n} \{Y_i - (b_0 + b_1 X_i)\}^2.$$

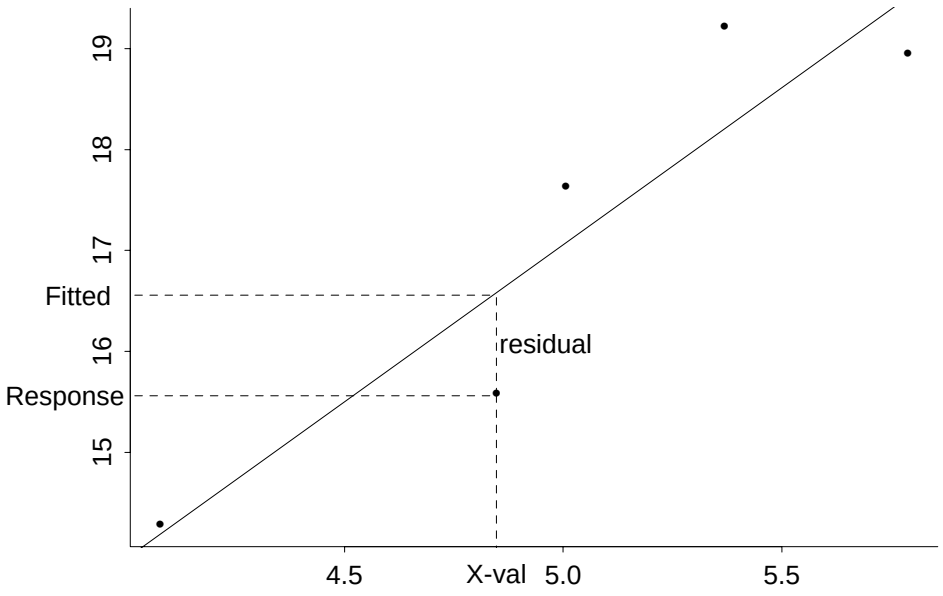
Several bits of notation are needed. Let

$$\hat{Y}_i = b_0 + b_1 X_i$$

be the **predicted** or fitted Y-value for an X-value of X_i and let $e_i = Y_i - \hat{Y}_i$. The fitted value $\hat{Y}_i$ is the value of the LS line at X_i whereas the **residual** e_i is the distance that the observed response Y_i is from the LS line. Given this notation,

$$\text{Residual Sums of Squares} = \text{Res SS} = \sum_{i=1}^{n} (Y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} e_i^2.$$

Here is a picture to clarify matters:



The Residual SS, or sum of squared residuals, is *small* if each $\hat{Y}_i$ is *close to* Y_i (i.e., the line closely fits the data). It can be shown that

$$\text{Total SS in Y} = \sum_{i=1}^{n}(Y_i - \bar{Y})^2 \geq \text{Res SS} \geq 0.$$

Also define

$$\text{Regression SS} = \text{Reg SS} = \text{Total SS} \ - \ \text{Res SS} = b_1 \sum_{i=1}^{n}(Y_i - \bar{Y})(X_i - \bar{X}).$$

The Total SS measures the variability in the Y-sample. Note that

$$0 \leq \text{Regression SS} \leq \text{Total SS}.$$

The percentage of the variability in the Y-sample that is **explained by the linear relationship** between Y and X is

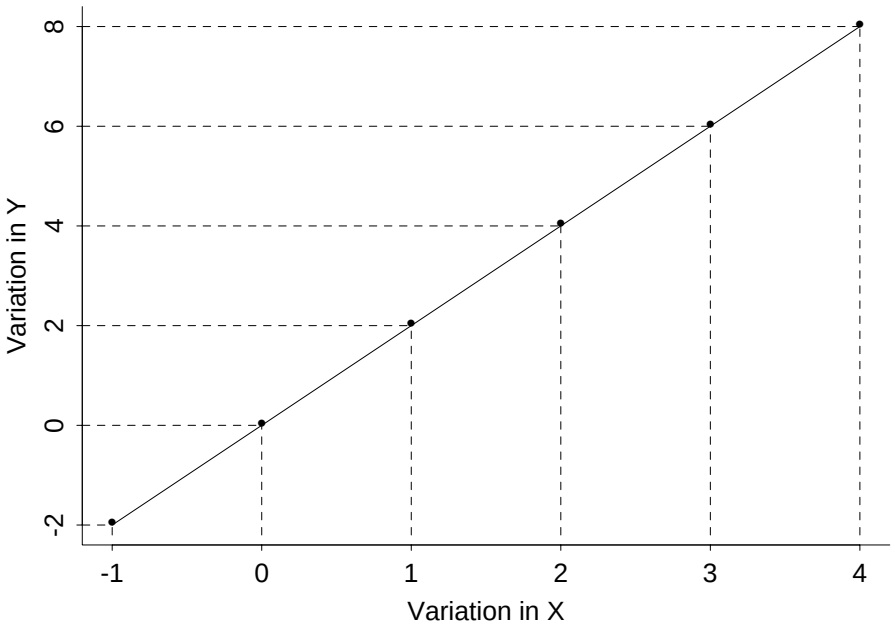
$$R^2 = \text{coefficient of determination} = \frac{\text{Reg SS}}{\text{Total SS}}.$$

Given the definitions of the Sums of Squares, we can show $0 \leq R^2 \leq 1$ and

$$R^2 = \text{square of Pearson correlation coefficient } = r^2.$$

To understand the interpretation of R^2, at least in two extreme cases, note that

$$\begin{aligned}
\text{Reg SS} = \text{Total SS} &\Leftrightarrow \text{Res SS} = 0 \\
&\Leftrightarrow \text{all the data points fall on a straight line} \\
&\Leftrightarrow \text{all the variability in } Y \text{ is explained by the linear relationshi} \\
&\quad \text{(which has variation)} \\
&\Leftrightarrow R^2 = 1. \quad \text{(see the picture below)}
\end{aligned}$$



Furthermore,

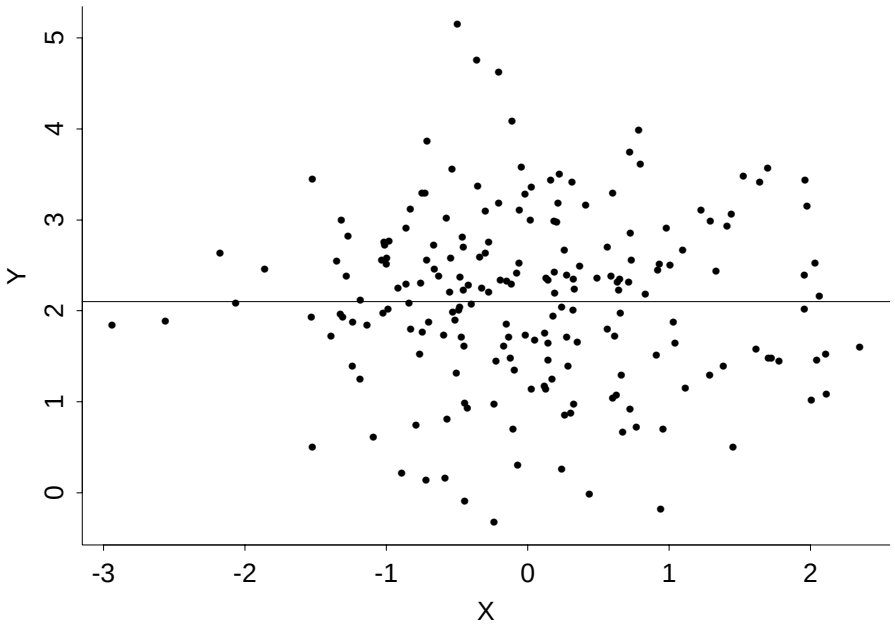
Reg SS $= 0 \Leftrightarrow$ Total SS $=$ Res SS

$\Leftrightarrow b_1 = 0$

$\Leftrightarrow$ LS line is $\hat{Y} = \bar{Y}$

$\Leftrightarrow$ none of the variability in Y is explained by a linear relationship

$\Leftrightarrow R^2 = 0.$



Each Sum of Squares has a corresponding *df* (degrees of freedom). The Sums of Squares and *df* are arranged in an analysis of variance (ANOVA) table:

Source	*df*	SS	MS
Regression	1	SSR	MSR
Residual (Error)	$n-2$	SSE	MSE
Total	$n-1$		

The Total *df* is $n-1$. The Residual *df* is n minus the number of parameters (2) estimated by the LS line. The Regression *df* is the number of predictor variables (1) in the model. A Mean Square is always equal to the Sum of Squares divided by the *df*. Sometime the following notation is used for the Residual MS: $s^2_{Y|X} = \text{Resid(SS)}/(n-2)$.

```
# ANOVA table of the simple linear regression fit
anova(lm.blood.wt)
```

```
## Analysis of Variance Table
##
## Response: blood_loss
##           Df  Sum Sq Mean Sq F value  Pr(>F)
```

```
## weight     1 1207.45 1207.45  8.8778 0.02465 *
## Residuals  6  816.05  136.01
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

8.5.1 Brief discussion of the output for blood loss problem

1. Identify fitted line: Blood Loss $= 552.44 - 1.30$ Weight (i.e., $b_0 = 552.44$ and $b_1 = -1.30$).
   What is the line of best fit? What is the direction of the relationship?
2. Locate Analysis of Variance Table.
   This tests the the hypothesis $H_0 : \beta_j = 0$, $j = 0, 1, \ldots, p$ (for all $p + 1$ beta coefficients), against H_A : not H_0, i.e., at least one $\beta_j \neq 0$. More on this later.
3. Locate Parameter Estimates Table.
   Given that not all the betas are zero from the ANOVA, which parameter betas are different from zero, and what are their estimated values and standard errors? More on this later.
4. Note that $R^2 = 0.5967 = (-0.77247)^2 = r^2$.
   R^2 indicates the proportion of the variance explained by the regression model. This indicates the predictive ability of the model, and is *not* a indication of model fit.

■ ClickerQs — Salaries STT.02.02.060 ■

8.6 The regression model

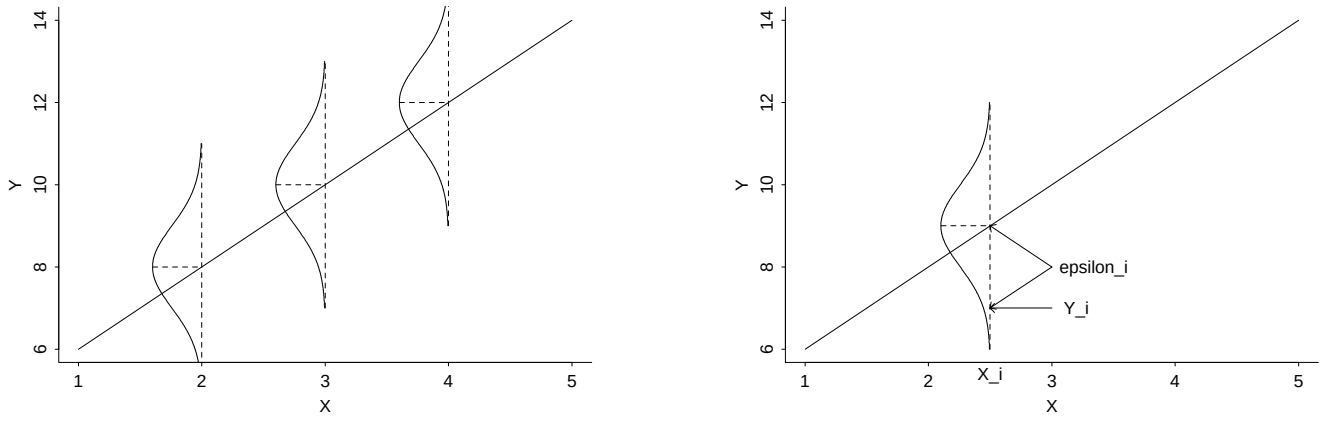
The following statistical model is assumed as a means to provide error estimates for the LS line, regression coefficients, and predictions. Assume that the data (X_i, Y_i), $i = 1, \ldots, n$, are a sample of (X, Y) values from the population of interest, and

1. The mean in the population of all responses Y at a given X value (sometimes called $\mu_{Y|X}$) falls on a straight line, $\beta_0+\beta_1 X$, called the population regression line.
2. The variation among responses Y at a given X value is the same for each X, and is denoted by $\sigma^2_{Y|X}$.
3. The population of responses Y at a given X is normally distributed.
4. The pairs (X_i, Y_i) are a random sample from the population. Alternatively, we can think that the X_is were fixed by the experimenter, and that the Y_i are random responses at the selected predictor values.

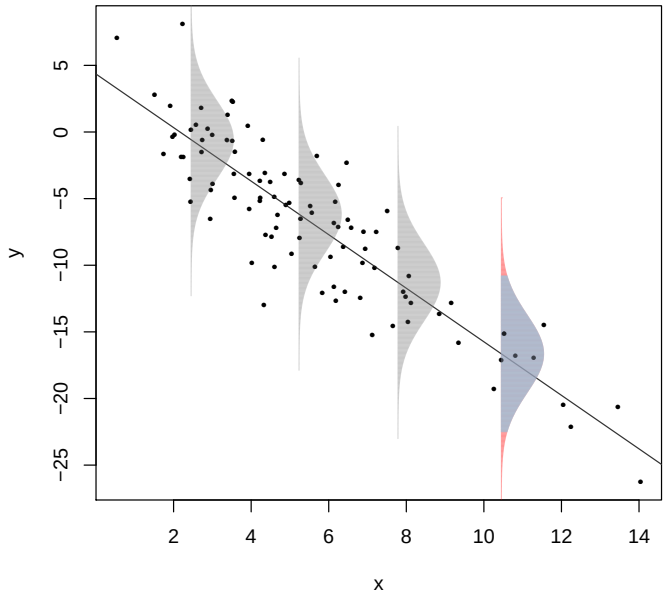
The model is usually written in the form

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

(i.e., Response = Mean Response + Residual), where the ε_is are, by virtue of assumptions 2, 3, and 4, independent normal random variables with mean 0 and variance $\sigma^2_{Y|X}$. The following picture might help see this. Note that the population regression line is unknown, and is estimated from the data using the LS line.



In the plot below, data are simulated where $y_i = 4 - 2x_i + e_i$, where $x_i \sim$ Gamma$(3, 0.5)$ and $e_i \sim$ Normal$(0, 3^2)$. The data are plotted and a linear regression is fit and the mean regression line is overlayed. Select normal distributions with variance estimated from the linear model fit are overlayed, one which indicates limits at two standard deviations. See the R code to create this image.



Model assumptions In decreasing order of importance, the model assumptions are

1. **Validity.** Most importantly, the data you are analyzing should map to the research question you are trying to answer. This sounds obvious but is often overlooked or ignored because it can be inconvenient.

2. **Additivity and linearity.** The most important mathematical assumption of the regression model is that its deterministic component is a linear function of the separate predictors.

3. **Independence of errors.** This assumption depends on how the data were collected.

4. **Equal variance of errors.**

5. **Normality of errors.**

Normality and equal variance are typically minor concerns, unless you're using the model to make predictions for individual data points.

8.6.1 Back to the Data

There are three unknown population parameters in the model: β_0, β_1 and $\sigma^2_{Y|X}$. Given the data, the LS line

$$\hat{Y} = b_0 + b_1 X$$

estimates the population regression line $Y = \beta_0 + \beta_1 X$. The LS line is our best guess about the unknown population regression line. Here b_0 estimates the intercept β_0 of the population regression line and b_1 estimates the slope β_1 of the population regression line.

The i^{th} **observed residual** $e_i = Y_i - \hat{Y}_i$, where $\hat{Y}_i = b_0 + b_1 X_i$ is the i^{th} **fitted value**, estimates the **unobservable residual** ε_i (ε_i is unobservable because β_0 and β_1 are unknown). The Residual MS from the ANOVA table is used to estimate $\sigma^2_{Y|X}$:

$$s^2_{Y|X} = \text{Res MS} = \frac{\text{Res SS}}{\text{Res df}} = \frac{\sum_i (Y_i - \hat{Y}_i)^2}{n-2}.$$

The denominator $df = n - 2$ is the number of observations minus the number of beta parameters in the model, i.e., β_0 and β_1.

8.7 CI and tests for β_1

A CI for β_1 is given by $b_1 \pm t_{\text{crit}} SE_{b_1}$, where the standard error of b_1 under the model is

$$SE_{b_1} = \frac{s_{Y|X}}{\sqrt{\sum_i (X_i - \bar{X})^2}},$$

and where t_{crit} is the appropriate critical value for the desired CI level from a t-distribution with df =Res df.

To test $H_0 : \beta_1 = \beta_{10}$ (a given value) against $H_A : \beta_1 \neq \beta_{10}$, reject H_0 if $|t_s| \geq t_{\text{crit}}$, where

$$t_s = \frac{b_1 - \beta_{10}}{SE_{b_1}},$$

and t_{crit} is the t-critical value for a two-sided test, with the desired size and df =Res df. Alternatively, you can evaluate a p-value in the usual manner to make a decision about H_0.

```
# CI for beta1
sum.lm.blood.wt <- summary(lm.blood.wt)
sum.lm.blood.wt$coefficients
```

```
##               Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 552.442023 21.4408832 25.765824 2.253105e-07
## weight       -1.300327  0.4364156 -2.979562 2.465060e-02
```

```
est.beta1 <- sum.lm.blood.wt$coefficients[2,1]
se.beta1  <- sum.lm.blood.wt$coefficients[2,2]
sum.lm.blood.wt$fstatistic
```

```
##    value    numdf    dendf
## 8.877788 1.000000 6.000000
```

```
df.beta1  <- sum.lm.blood.wt$fstatistic[3]
t.crit    <- qt(1-0.05/2, df.beta1)
t.crit
```

```
## [1] 2.446912
```

```
CI.lower  <- est.beta1 - t.crit * se.beta1
CI.upper  <- est.beta1 + t.crit * se.beta1
c(CI.lower, CI.upper)
```

```
## [1] -2.3681976 -0.2324567
```

The parameter estimates table gives the standard error, t-statistic, and p-value for testing $H_0 : \beta_1 = 0$. Analogous summaries are given for the intercept, β_0, but these are typically of less interest.

8.7.1 Testing $\beta_1 = 0$

Assuming the mean relationship is linear, consider testing $H_0 : \beta_1 = 0$ against $H_A : \beta_1 \neq 0$. This test can be conducted using a t-statistic, as outlined above, or with an ANOVA F-test, as outlined below.

For the analysis of variance (ANOVA) F-test, compute

$$F_s = \frac{\text{Reg MS}}{\text{Res MS}}$$

and reject H_0 when F_s exceeds the critical value (for the desired size test) from an F-table with numerator $df = 1$ and denominator $df = n - 2$ (see `qf()`).

The hypothesis of zero slope (or no relationship) is rejected when F_s is large, which happens when a significant portion of the variation in Y is explained by the linear relationship with X.

The p-values from the t-test and the F-test are always equal. Furthermore this p-value is equal to the p-value for testing no correlation between Y and X, using the t-test described earlier. Is this important, obvious, or disconcerting?

8.8 A CI for the population regression line

I can not overemphasize the **power** of the regression model. The model allows you to estimate the mean response at any X value in the range for which the model is reasonable, even if little or no data is observed at that location.

We estimate the mean population response among individuals with $X = X_p$

$$\mu_p = \beta_0 + \beta_1 X_p,$$

with the fitted value, or the value of the least squares line at X_p:

$$\hat{Y}_p = b_0 + b_1 X_p.$$

X_p is not necessarily one of the observed X_is in the data. To get a CI for μ_p, use $\hat{Y}_p \pm t_{\text{crit}} SE(\hat{Y}_p)$, where the standard error of $\hat{Y}_p$ is

$$SE(\hat{Y}_p) = s_{Y|X} \sqrt{\frac{1}{n} + \frac{(X_p - \bar{X})^2}{\sum_i (X_i - \bar{X})^2}}.$$

The t-critical value is identical to that used in the subsection on CI for β_1.

8.8.1 CI for predictions

Suppose a future individual (i.e., someone not used to compute the LS line) has $X = X_p$. The best prediction for the response Y of this individual is the value of the least squares line at X_p:

$$\hat{Y}_p = b_0 + b_1 X_p.$$

To get a CI (prediction interval) for an individual response, use $\hat{Y}_p \pm t_{\text{crit}} SE_{pred}(\hat{Y}_p)$, where

$$SE_{pred}(\hat{Y}_p) = s_{Y|X}\sqrt{1 + \frac{1}{n} + \frac{(X_p - \bar{X})^2}{\sum_i (X_i - \bar{X})^2}},$$

and t_{crit} is identical to the critical value used for a CI on β_1. The prediction variance has two parts: (1) the 1 indicates the variability associated with the data around the mean (regression line), and (2) the rest is the variability associated with estimating the mean.

For example, in the blood loss problem you may want to estimates the blood loss for an $50kg$ individual, and to get a CI for this prediction. This problem is different from computing a CI for the mean blood loss of all $50kg$ individuals!

```
# CI for the mean and PI for a new observation at weight=50
predict(lm.blood.wt, data.frame(weight=50), interval = "confidence", level = 0.95)
```

```
##        fit      lwr      upr
## 1 487.4257 477.1575 497.6938
```

```
predict(lm.blood.wt, data.frame(weight=50), interval = "prediction", level = 0.95)
```

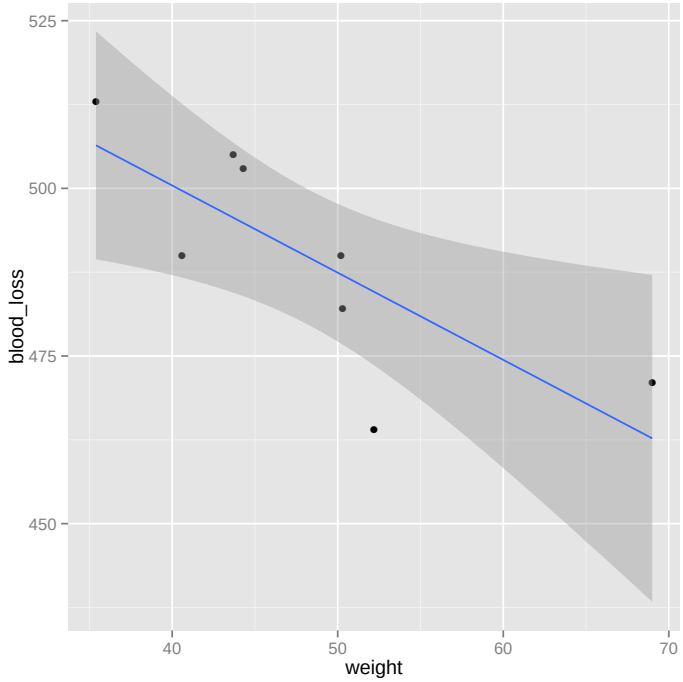
```
##        fit     lwr      upr
## 1 487.4257 457.098 517.7533
```

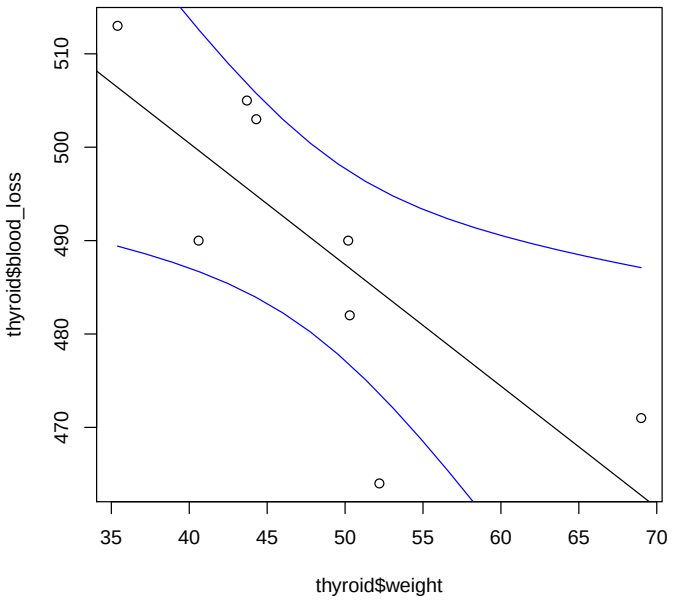
Comments

1. The prediction interval is wider than the CI for the mean response. This is reasonable because you are less confident in predicting an individual response than the mean response for all individuals.
2. The CI for the mean response and the prediction interval for an individual response become wider as X_p moves away from $\bar{X}$. That is, you get a more sensitive CI and prediction interval for X_ps near the center of the data.
3. In plots below include confidence and prediction bands along with the fitted LS line.

```
# ggplot: Plot the data with linear regression fit and confidence bands
library(ggplot2)
p <- ggplot(thyroid, aes(x = weight, y = blood_loss))
p <- p + geom_point()
p <- p + geom_smooth(method = lm, se = TRUE)
```

```
print(p)


# Base graphics: Plot the data with linear regression fit and confidence bands
# scatterplot
plot(thyroid$weight, thyroid$blood_loss)
# regression line from lm() fit
abline(lm.blood.wt)
# x values of weight for predictions of confidence bands
x.pred <- data.frame(weight = seq(min(thyroid$weight), max(thyroid$weight),
                         length = 20))
# draw upper and lower confidence bands
lines(x.pred$weight, predict(lm.blood.wt, x.pred,
                             interval = "confidence")[, "upr"], col = "blue")
lines(x.pred$weight, predict(lm.blood.wt, x.pred,
                             interval = "confidence")[, "lwr"], col = "blue")
```





8.8.2 A further look at the blood loss data

- The LS line is: Predicted Blood Loss = 552.442 - 1.30 Weight.
- The R^2 is 0.597 (i.e., 59.7%).
- The F-statistic for testing $H_0 : \beta_1 = 0$ is $F_{obs} = 8.88$ with a p-value=0.025. The Error MS is $s^2_{Y|X} = 136.0$; see ANOVA table.
- The Parameter Estimates table gives b_0 and b_1, their standard errors, and

t-statistics and p-values for testing $H_0 : \beta_0 = 0$ and $H_0 : \beta_1 = 0$. The t-test and F-test p-values for testing that the slope is zero are identical. We could calculate a 95% CI for β_0 and β_1. If we did so (using the t critical value) we find we are 95% confident that the slope of the population regression line is between -2.37 and -0.23.

- Suppose we are interested in estimating the average blood loss among all 50kg individuals. The estimated mean blood loss is $552.442 - 1.30033 \times 50 = 487.43$. Reading off the plot, we are 95% confident that the mean blood loss of all 50kg individuals is between (approximately) 477 and 498 ml. A 95% prediction interval for the blood loss of a single 50 kg person is less precise (about 457 to 518 ml).

As a summary we might say that weight is important for explaining the variation in blood loss. In particular, the estimated slope of the least squares line (Predicted Blood loss = 552.442 - 1.30 Weight) is significantly different from zero (p-value = 0.0247), with weight explaining approximately 60% (59.7%) of the variation in blood loss for this sample of 8 thyroid operation patients.

8.9 Model Checking and Regression Diagnostics

8.9.1 Introduction

The simple linear regression model is usually written as

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

where the ε_is are independent normal random variables with mean 0 and variance σ^2. The model implies (1) The average Y-value at a given X-value is linearly related to X. (2) The variation in responses Y at a given X value is constant. (3) The population of responses Y at a given X is normally distributed. (4) The observed data are a random sample.

A regression analysis is never complete until these assumptions have been checked. In addition, you need to evaluate whether individual observations, or groups of observations, are unduly influencing the analysis. A first step in any analysis is to plot the data. The plot provides information on the linearity and constant variance assumption.

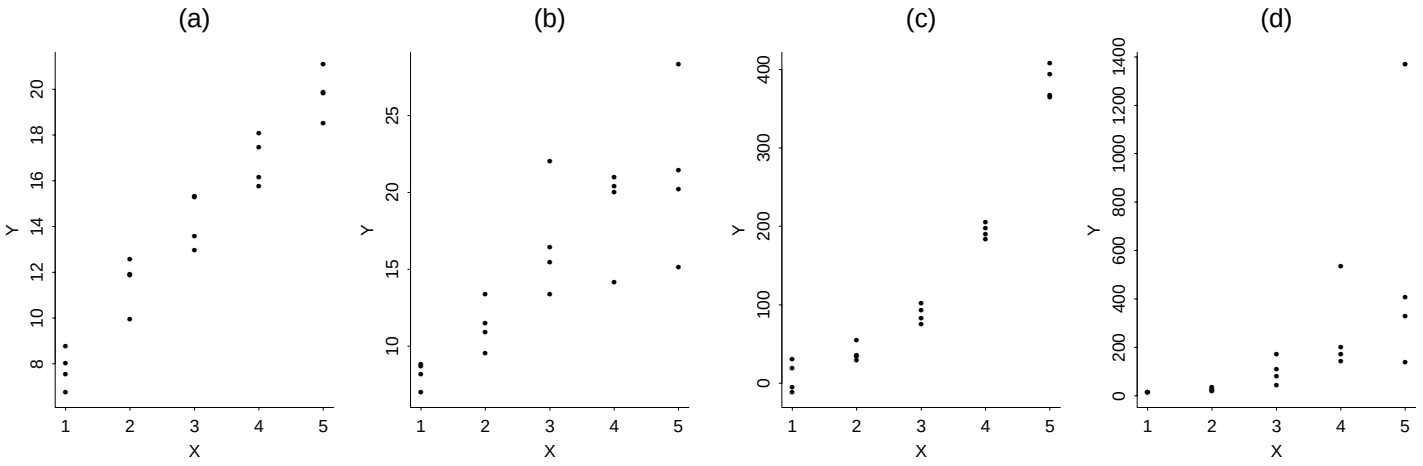


Figure (a) is the only plot that is consistent with the assumptions. The plot shows a linear relationship with constant variance. The other figures show one or more deviations. Figure (b) shows a linear relationship but the variability increases as the mean level increases. In Figure (c) we see a nonlinear relationship with constant variance, whereas (d) exhibits a nonlinear relationship with non-constant variance.

In many examples, nonlinearity or non-constant variability can be addressed by **transforming** Y or X (or both), or by fitting **polynomial models**. These issues will be addressed later.

8.9.2 Residual Analysis

A variety of methods for assessing model adequacy are based on the **observed residuals**,

$$e_i = Y_i - \hat{Y}_i \quad \text{i.e., Observed} - \text{Fitted values.}$$

The residual is the difference between the observed values and predicted or fitted values. The residual is the part of the observation that is not explained by the fitted model. You can analyze residuals to determine the adequacy of the model. A large residual identifies an observation poorly fit by the model.

The residuals are usually plotted in various ways to assess potential inadequacies. The observed residuals have different variances, depending on X_i. Recall that the standard error of $\hat{Y}_i$ (and therefore e_i) is

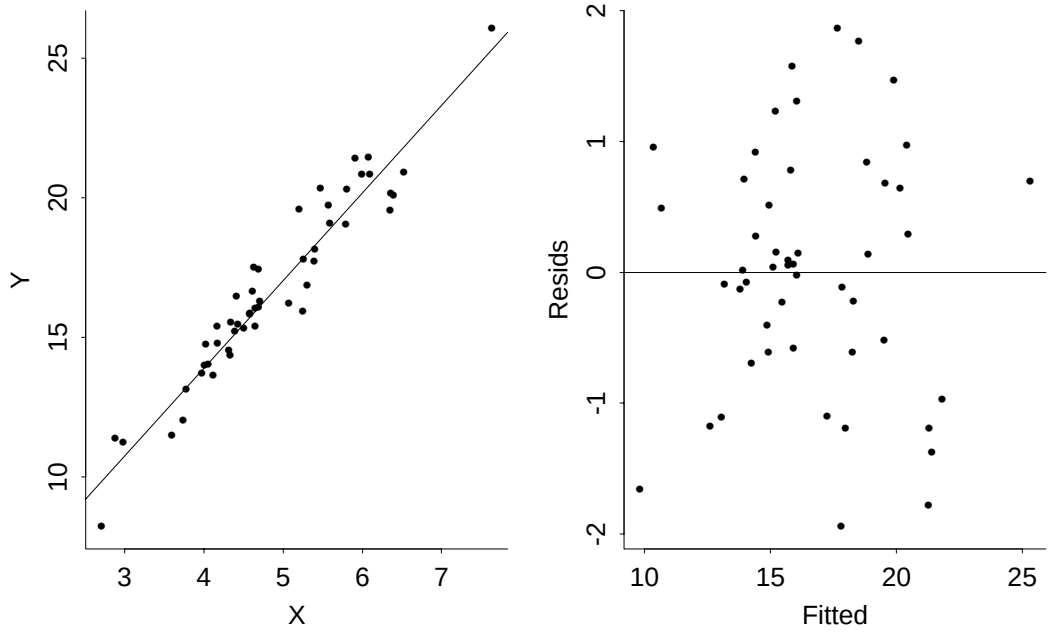
$$\text{SE}(\hat{Y}_i) \;=\; \text{SE}(e_i) \;=\; s_{Y|X}\sqrt{\frac{1}{n}+\frac{(X_i-\bar{X})^2}{\sum_j (X_j-\bar{X})^2}}.$$

So many statisticians prefer to plot the **studentized residuals** (sometimes called the standardized residuals or internally Studentized residual)

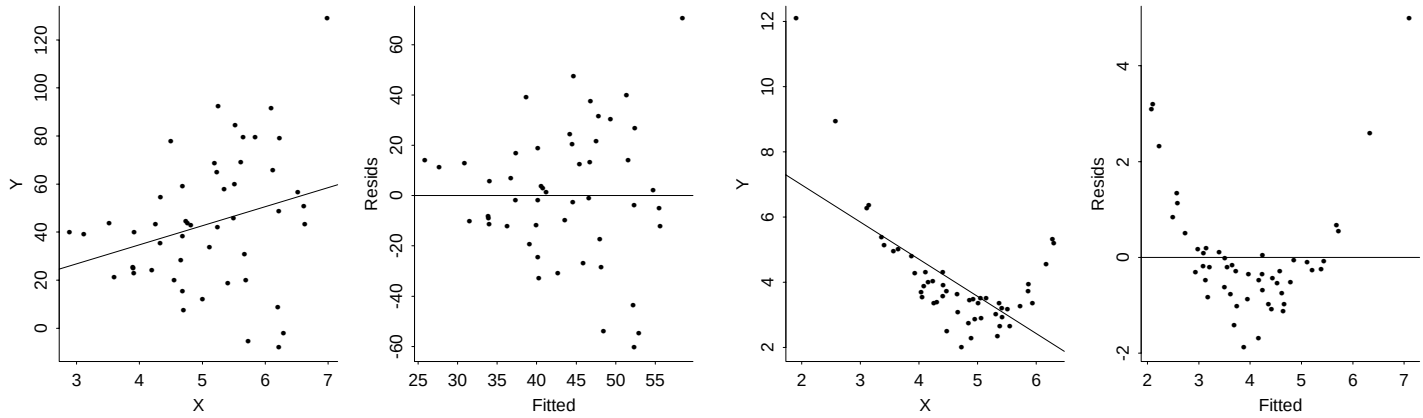
$$r_i = \frac{e_i}{SE(e_i)}.$$

The standardized residual is the residual, e_i, divided by an estimate of its standard deviation. This form of the residual takes into account that the residuals may have different variances, which can make it easier to detect outliers. The studentized residuals have a constant variance of 1 (approximately). Standardized residuals greater than 2 and less than -2 are usually considered large. I will focus on diagnostic methods using the studentized residuals.

A plot of the studentized residuals r_i against the fitted values $\hat{Y}_i$ often reveals inadequacies with the model. The real power of this plot is with multiple predictor problems (multiple regression). The information contained in this plot with simple linear regression is similar to the information contained in the original data plot, except it is scaled better and eliminates the effect of the trend on your perceptions of model adequacy. The residual plot should exhibit no systematic dependence of the sign or the magnitude of the residuals on the fitted values:

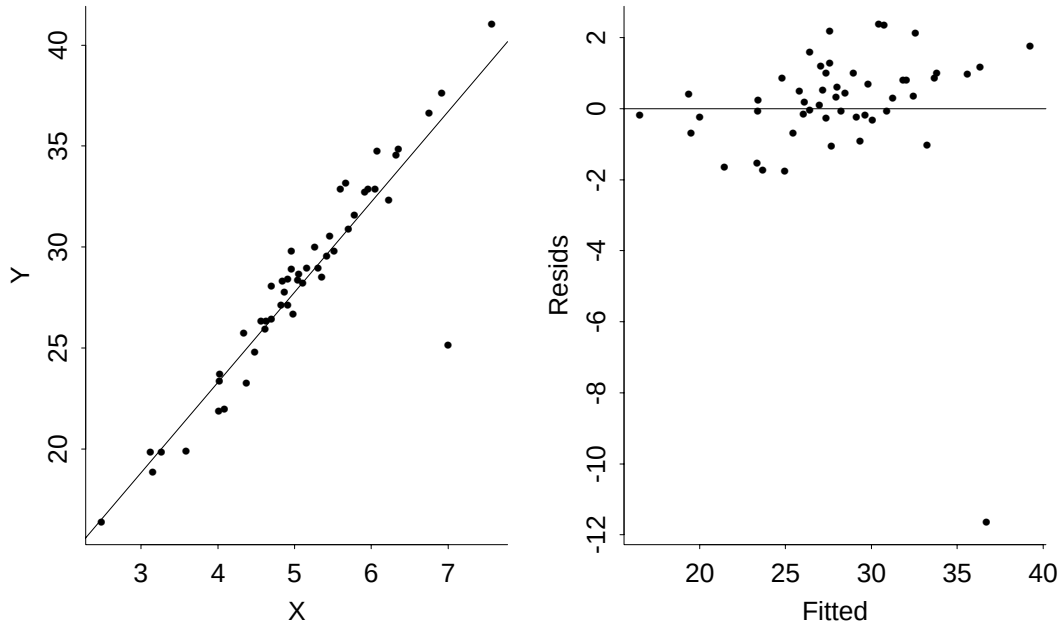


The following sequence of plots show how inadequacies in the data plot appear in a residual plot. The first plot shows a roughly linear relationship between Y and X with non-constant variance. The residual plot shows a megaphone shape rather than the ideal horizontal band. A possible remedy is a **weighted least squares** analysis to handle the non-constant variance (see end of chapter for an example), or to transform Y to stabilize the variance. Transforming the data may destroy the linearity.



The plot above shows a nonlinear relationship between Y and X. The residual plot shows a systematic dependence of the sign of the residual on the fitted value. Possible remedies were mentioned earlier.

The plot below shows an **outlier**. This case has a large residual and large studentized residual. A sensible approach here is to refit the model after holding out the case to see if any conclusions change.



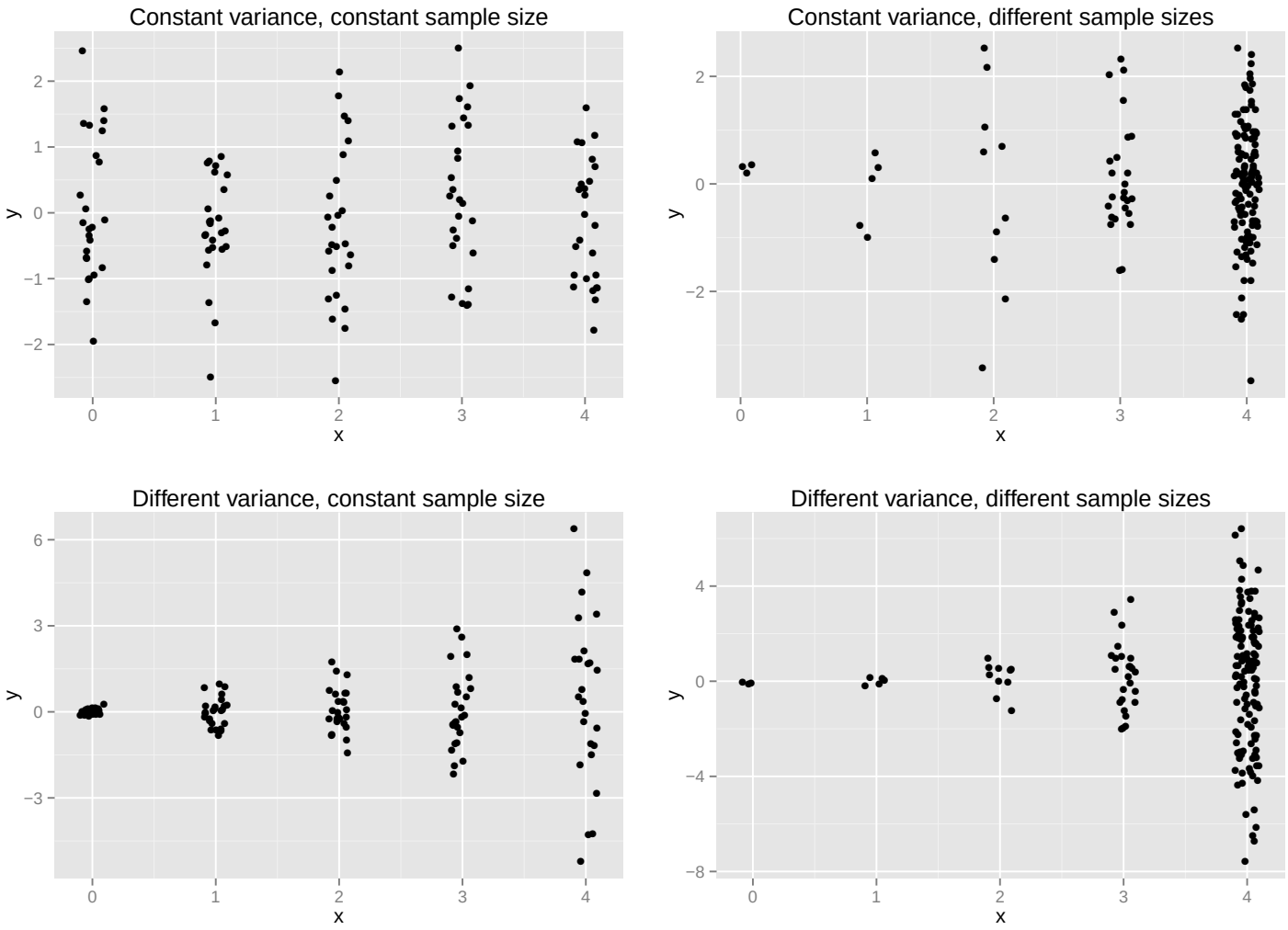
A third type of residual is called an **externally Studentized residual** (or Studentized deleted residual or deleted t-residual). The problem with residuals is that a highly influential value can force the residual to have a very small value. This measure tries to correct for that by looking at how well the model fits this observation without using this observation to construct the fit. It is quite possible for the deleted residual to be huge when the raw residual is tiny.

The studentized deleted residual for observation i^{th} is calculated by fitting the regression based on all of the cases except the i^{th} one. The residual is then divided by its estimated standard deviation. Since the Studentized deleted residual for the i^{th} observation estimates all quantities with this observation deleted from the data set, the i^{th} observation cannot influence these estimates. Therefore, unusual Y values clearly stand out. Studentized deleted residuals with large absolute values are considered large. If the regression model is appropriate, with no outlying observations, each Studentized deleted residual follows the t-distribution with $n - 1 - p$ degrees of freedom.

Nonconstant variance vs sample size Because more extreme observations are more likely to occur with larger sample sizes, sometimes when sample size depends on X it can appear as noncontant variance. Below sample sizes are either all 25 or (3, 5, 10, 25, 125), and standard deviations are either all 1 or (0.1, 0.5, 1, 1.5, 3). Note that constant variance and different sample sizes appears as though it has nonconstant variance.

```
#### Nonconstant variance vs sample size
dat.var.sam <- function(n, s) {
  dat <- data.frame(
           x = c(rep(0, n[1]),
                 rep(1, n[2]),
                 rep(2, n[3]),
                 rep(3, n[4]),
                 rep(4, n[5])),
           y = c(rnorm(n[1], mean = 0, sd = s[1]),
                 rnorm(n[2], mean = 0, sd = s[2]),
                 rnorm(n[3], mean = 0, sd = s[3]),
                 rnorm(n[4], mean = 0, sd = s[4]),
                 rnorm(n[5], mean = 0, sd = s[5]))
         )
```

```
  return(dat)
}

library(ggplot2)

n <- c(25, 25, 25, 25, 25)
s <- c(1, 1, 1, 1, 1)
dat <- dat.var.sam(n, s)
p1 <- ggplot(dat, aes(x = x, y = y))
p1 <- p1 + geom_point(position = position_jitter(width = 0.1))
p1 <- p1 + labs(title = "Constant variance, constant sample size")
#print(p1)

n <- c(3, 5, 10, 25, 125)
s <- c(1, 1, 1, 1, 1)
dat <- dat.var.sam(n, s)
p2 <- ggplot(dat, aes(x = x, y = y))
p2 <- p2 + geom_point(position = position_jitter(width = 0.1))
p2 <- p2 + labs(title = "Constant variance, different sample sizes")
#print(p2)

n <- c(25, 25, 25, 25, 25)
s <- c(0.1, 0.5, 1, 1.5, 3)
dat <- dat.var.sam(n, s)
p3 <- ggplot(dat, aes(x = x, y = y))
p3 <- p3 + geom_point(position = position_jitter(width = 0.1))
p3 <- p3 + labs(title = "Different variance, constant sample size")
#print(p3)

n <- c(3, 5, 10, 25, 125)
s <- c(0.1, 0.5, 1, 1.5, 3)
dat <- dat.var.sam(n, s)
p4 <- ggplot(dat, aes(x = x, y = y))
p4 <- p4 + geom_point(position = position_jitter(width = 0.1))
p4 <- p4 + labs(title = "Different variance, different sample sizes")
#print(p4)

library(gridExtra)
grid.arrange(p1, p2, p3, p4, nrow=2, ncol=2)
```



8.9.3 Checking Normality

The normality assumption for the ε_is can be evaluated visually with a boxplot or a normal probability plot (rankit plot) of the r_i, or formally with a Shapiro-Wilk test. The normal probability plot often highlights **outliers**, or poorly fitted cases. If an outlier is held out of the data and a new analysis is performed, the resulting normal scores plot may be roughly linear, but often will show a short-tailed distribution. (Why?).

You must interpret regression tests and CI with caution with non-normal data. Statisticians developed robust regression methods for non-normal data which are available in R packages.

8.9.4 Checking Independence

Diagnosing dependence among observations requires an understanding of the data collection process. There are a variety of graphical and inferential tools for checking independence for data collected over time (called a time series). The easiest check is to plot the r_i against time index and look for any suggestive patterns.

8.9.5 Outliers

Outliers are observations that are poorly fitted by the regression model. The response for an outlier is far from the fitted line, so outliers have large positive or negative values of the studentized residual r_i. Usually, $|r_i| > 2$ is considered large. Outliers are often highlighted in residual plots.

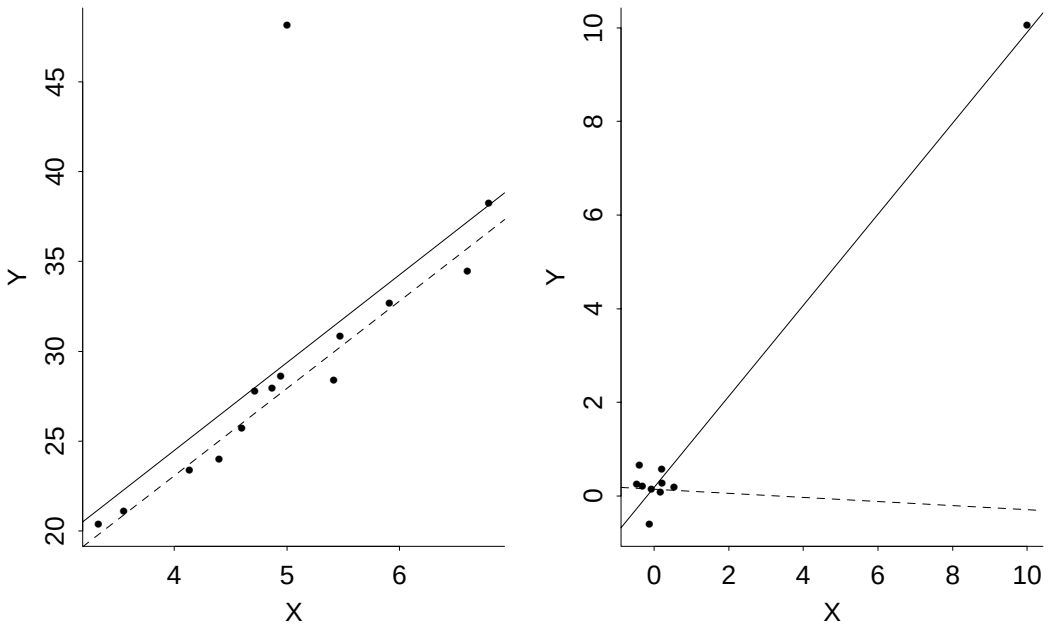
What do you do with outliers? Outliers may be due to incorrect recordings of the data or failure of the measuring device, or indications or a change in the mean or variance structure for one or more cases. Incorrect recordings should be fixed if possible, but otherwise deleted from the analysis.

Routine deletion of outliers from the analysis is not recommended. This practice can have a dramatic effect on the fit of the model and the perceived precision of parameter estimates and predictions. Analysts who routinely omit outliers without cause tend to overstate the significance of their findings and get a false sense of precision in their estimates and predictions. To assess effects of outliers, a data analyst should repeat the analysis holding out the outliers to see whether any substantive conclusions are changed. Very often the only real effect of an outlier is to inflate MSE and hence make p-values a little larger and CIs a little wider than necessary, but without substantively changing conclusions. They can completely mask underlying patterns, however.

■ CLICKER Qs — Outliers STT.02.04.040 ■

8.9.6 Influential Observations

Certain data points can play an important role in determining the position of the LS line. These data points may or may not be outliers. In the plots below, the solid line is the LS line from the full data set, whereas the dashed line is the LS line after omitting the unusual point. For example, the observation with $Y > 45$ in the first plot is an outlier relative to the LS fit. The extreme observation in the second plot has a very small r_i. Both points are highly **influential observations** — the LS line changes dramatically when these observations are held out.



In the second plot, the extreme value is a **high leverage** value, which is basically an outlier among the X values; Y does not enter in this calculation. This influential observation is not an outlier because its presence in the analysis determines that the LS line will essentially pass through it! These are values

with the *potential* of greatly distorting the fitted model. They may or may not actually have distorted it.

The `hat` variable from the `influence()` function on the object returned from `lm()` fit will give the leverages: `influence(lm.output)$hat`. Leverage values fall between 0 and 1. Experts consider a leverage value greater than $2p/n$ or $3p/n$, where p is the number of predictors or factors plus the constant and n is the number of observations, large and suggest you examine the corresponding observation. A rule-of-thumb is to identify observations with leverage over $3p/n$ or 0.99, whichever is smaller.

Dennis Cook developed a measure of the impact that individual cases have on the placement of the LS line. His measure, called **Cook's distance** or Cook's D, provides a summary of how far the LS line changes when each individual point is held out (one at a time) from the analysis. While high leverage values indicate observations that have the *potential* of causing trouble, those with high Cook's D values actually *do* disproportionately affect the overall fit. The case with the largest D has the greatest impact on the placement of the LS line. However, the actual influence of this case may be small. In the plots above, the observations I focussed on have the largest Cook's Ds.

A simple, but not unique, expression for Cook's distance for the j^{th} case is

$$D_j \propto \sum_i (\hat{Y}_i - \hat{Y}_{i[-j]})^2,$$

where $\hat{Y}_{i[-j]}$ is the fitted value for the i^{th} case when the LS line is computed from all the data except case j. Here $\propto$ means that D_j is a multiple of $\sum_i (\hat{Y}_i - \hat{Y}_{i[-j]})^2$ where the multiplier does not depend on the case. This expression implies that D_j is also an overall measure of how much the fitted values change when case j is deleted.

Observations with large D values may be outliers. Because D is calculated using leverage values and standardized residuals, it considers whether an observation is unusual with respect to both x- and y-values. To interpret D, compare it to the F-distribution with $(p, n-p)$ degrees-of-freedom to determine the corresponding percentile. If the percentile value is less than 10% or

20%, the observation has little influence on the fitted values. If the percentile value is greater than 50%, the observation has a major influence on the fitted values and should be examined.

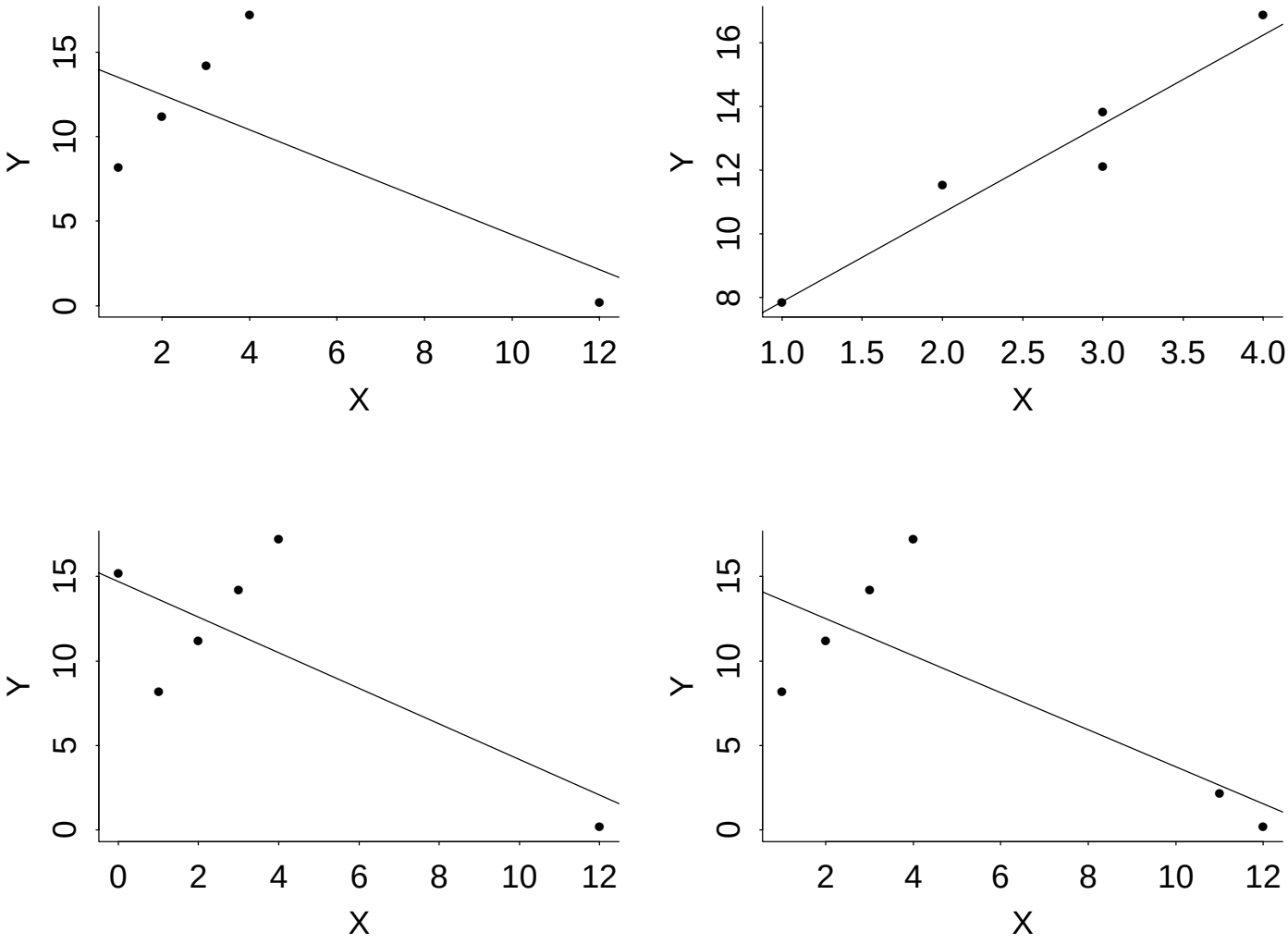
Many statisticians make it a lot simpler than this sounds and use 1 as a cutoff value for large Cook's D (when D is on the appropriate scale). Using the cutoff of 1 can simplify an analysis, since frequently one or two values will have noticeably larger D values than other observations without actually having much effect, *but it can be important to explore any observations that stand out*. Cook's distance values for each observation from a linear regression fit are given with `cooks.distance(lm.output)`.

Given a regression problem, you should locate the points with the largest D_js and see whether holding these cases out has a decisive influence on the fit of the model or the conclusions of the analysis. You can examine the relative magnitudes of the D_js across cases without paying much attention to the actual value of D_j, but there are guidelines (see below) on how large D_j needs to be before you worry about it.

It is difficult to define a good strategy for dealing with outliers and influential observations. Experience is the best guide. I will show you a few examples that highlight some standard phenomena. One difficulty you will find is that certain observations may be outliers because other observations are influential, or vice-versa. If an influential observation is held out, an outlier may remain an outlier, may become influential, or both, or neither. Observations of moderate influence may become more, or less influential, when the most influential observation is held out.

Thus, any sequential refitting of models holding out of observations should not be based on the original (full-data) summaries, but rather on the summaries that result as individual observations are omitted. I tend to focus more on influential observations than outliers.

In the plots below, which cases do you think are most influential, and which are outliers. What happens in each analysis if I delete the most influential case? Are any of the remaining cases influential or poorly fitted?



Many researchers are hesitant to delete points from an analysis. I think this view is myopic, and in certain instances, such as the Gesell example to be discussed, can not be empirically supported. Being rigid about this can lead to some silly analyses of data, but one needs a very good reason and full disclosure if any points are deleted.

8.9.7 Summary of diagnostic measures

The various measures discussed above often flag the same observations as unusual, but they certainly can flag different observations. At the very least I examine standardized residuals and Cooks's D values. They are invaluable diagnostic measures, but nothing is perfect. Observations can be unusual in

groups — a pair of unusual high leverage values close to each other will not necessarily be flagged by Cook's D since removing just one may not affect the fit very much. Any analysis takes some careful thought.

These measures and techniques really are designed for multiple regression problems where several predictor variables are used. We are using them in simple linear regression to learn what they do and see what they have to say about data, but in truth it is fairly simple with one variable to see what may be an outlier in the x-direction, to see if the data are poorly fit, etc. With more variables all that becomes quite difficult and these diagnostics are essential parts of those analyses.

8.10 Regression analysis suggestion

There are a lot options allowed in R. I will make a few suggestions here on how to start an analysis. What you find once you get started determines what more you might wish to explore.

1. **Plot the data.** With lots of variables the matrix plot is valuable as a quick screen. If you want to see better resolution on an individual scatter plot, do the individual scatter plot.
2. Do any obvious **transformations** of the data. We will discuss this in a lot more detail later. Re-plot with transformations.
3. **Fit the least squares equation**.
4. **Examine the residual plots and results.** Check for the patterns discussed earlier.
   (a) Plotting several diagnostic plots together seems convenient to me. This gives you the essential plot (residuals vs. fitted values) plus a quick check on normality and possible violations of independence and influence. If you see something that needs closer investigation, you may need to recreate one of the plots larger by itself.
   (b) Do you see curvature in the standardized residual plot? If the sign of the residuals has a distinct pattern vs. the fitted values, the linear fit

is not adequate and you need some remedy, such as transformations. Note that standardized residuals are more conventional and show you what actually happened, while deleted residuals are probably the better diagnostic tool for identifying problem cases.

(c) Does it appear $\sigma_{Y|X}$ depends upon X (we are assuming it does not)? A megaphone pattern in residuals vs. fits is the classic (not the only) pattern to look for. Weighted least squares or transformations may be called for.

(d) Do you see obvious outliers? Make sure you do not have a mis-recorded data value. It might be worth refitting the equation without the outlier to see if it affects conclusions substantially.

(e) Is the normality assumption reasonable? This can be very closely related to the preceding points.

(f) Is there a striking pattern in residuals vs. order of the data? This can be an indication that the independence assumption is not valid.

5. **Check the Cook's D values.** The Cook's distance plot and Residuals vs. Leverage (with Cook's D) plot are both helpful.
6. If you found problem observations, omit them from the analysis and see if any conclusions change substantially. There are two good ways to do this.
   (a) Subset the `data.frame` using `subset()`.
   (b) Use `lm()` with the `weights=` option with weights of 0 for the excluded observations, weights of 1 for those included.

You may need to repeat all these steps many times for a complete analysis.

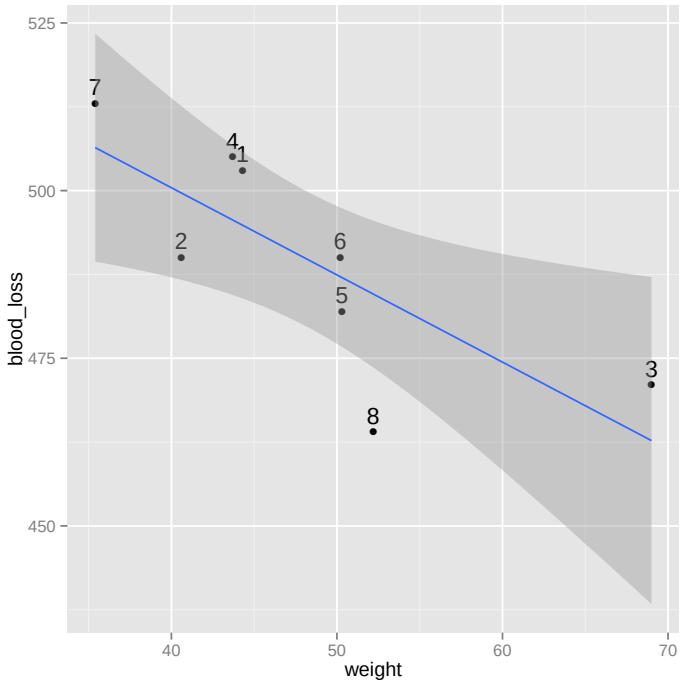
8.10.1 Residual and Diagnostic Analysis of the Blood Loss Data

We looked at much of this before, but let us go through the above steps systematically. Recall the data set (we want to predict blood loss from weight):

	weight	time	blood_loss
1	44.3	105	503
2	40.6	80	490
3	69.0	86	471
4	43.7	112	505
5	50.3	109	482
6	50.2	100	490
7	35.4	96	513
8	52.2	120	464

1. *Plot the data.* Plot blood loss vs. weight.

```
# create data ids
thyroid$id <- 1:nrow(thyroid)
# ggplot: Plot the data with linear regression fit and confidence bands
library(ggplot2)
p <- ggplot(thyroid, aes(x = weight, y = blood_loss, label = id))
p <- p + geom_point()
# plot labels next to points
p <- p + geom_text(hjust = 0.5, vjust = -0.5)
# plot regression line and confidence band
p <- p + geom_smooth(method = lm)
print(p)
```



Clearly the heaviest individual is an unusual value that warrants a closer look (maybe data recording error). I might be inclined to try a transformation here (such as `log(weight)`) to make that point a little less influential.

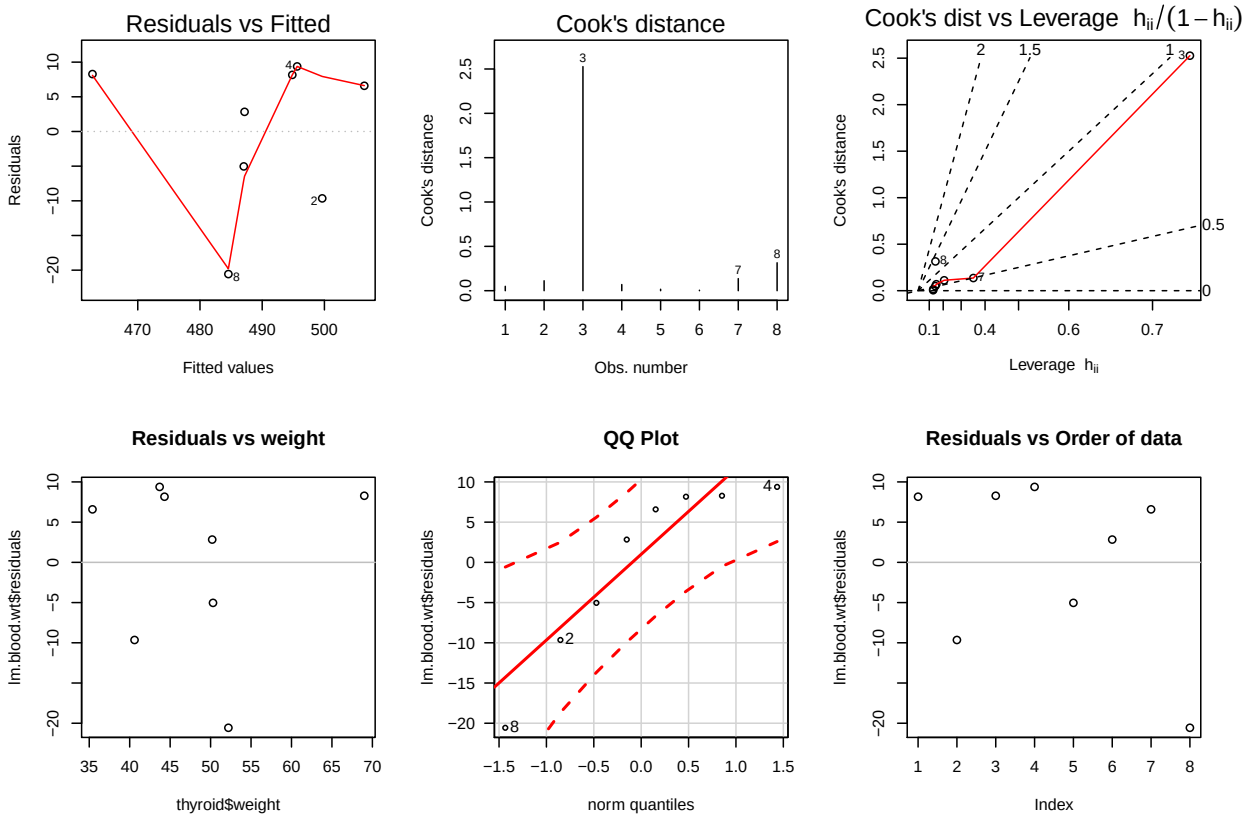
2. *Do any obvious transformations of the data.* We will look at transformations later.
3. *Fit the least squares equation.* Blood Loss appears significantly negatively associated with weight.

```
lm.blood.wt <- lm(blood_loss ~ weight, data = thyroid)
# use summary() to get t-tests of parameters (slope, intercept)
summary(lm.blood.wt)

##
## Call:
## lm(formula = blood_loss ~ weight, data = thyroid)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.565  -6.189   4.712   8.192   9.382
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 552.4420    21.4409   25.77 2.25e-07 ***
## weight       -1.3003     0.4364   -2.98   0.0247 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.66 on 6 degrees of freedom
## Multiple R-squared:  0.5967,Adjusted R-squared:  0.5295
## F-statistic: 8.878 on 1 and 6 DF,  p-value: 0.02465
```

   (a) *Graphs: Check Standardized Residuals (or the Deleted Residuals).* The residual plots:

```
# plot diagnistics
par(mfrow=c(2,3))
plot(lm.blood.wt, which = c(1,4,6))

# residuals vs weight
plot(thyroid$weight, lm.blood.wt$residuals, main="Residuals vs weight")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# Normality of Residuals
library(car)
# qq plot for studentized resid
#   las    =  1 : turns labels on y-axis to read horizontally
#   id.n   =  n : labels n most extreme observations, and outputs to console
#   id.cex =  1 : is the size of those labels
#   lwd    =  1 : line width
qqPlot(lm.blood.wt$residuals, las = 1, id.n = 3, main="QQ Plot")

## 8 2 4
```

```
## 1 2 8

# residuals vs order of data
plot(lm.blood.wt$residuals, main="Residuals vs Order of data")
  # horizontal line at zero
  abline(h = 0, col = "gray75")
```



4. *Examine the residual plots and results.*
   (a) *Do you see curvature?* There does not appear to be curvature (and it could be hard to detect with so few points).
   (b) *Does it appear $\sigma_{Y|X}$ depends upon X?* Not much evidence for this.
   (c) *Do you see obvious outliers?* Observation 3 is an outlier in the x direction, and therefore possibly a high leverage point and influential on the model fit.
   (d) *Is the normality assumption reasonable?* There appears to be some skewness, but with so few points normality may be reasonable.
   (e) *Is there a striking pattern in residuals vs. order of the data?* No striking pattern.
5. *Check the Cook's D values.* We anticipated that the 3rd observation is

affecting the fit by a lot more than any other values. The D-value is much larger than 1. Note that the residual is not large for this value.

6. *Omit problem observations from the analysis and see if any conclusions change substantially.* Let us refit the equation without observation 3 to see if anything changes drastically. I will use the weighted least squares approach discussed earlier on this example. Define a variable **wt** that is 1 for all observations except obs. 3, and make it 0 for that one.

```
# wt = 1 for all except obs 3 where wt = 0
thyroid$wt <- as.numeric(!(thyroid$id == 3))
thyroid$wt
```

```
## [1] 1 1 0 1 1 1 1 1
```

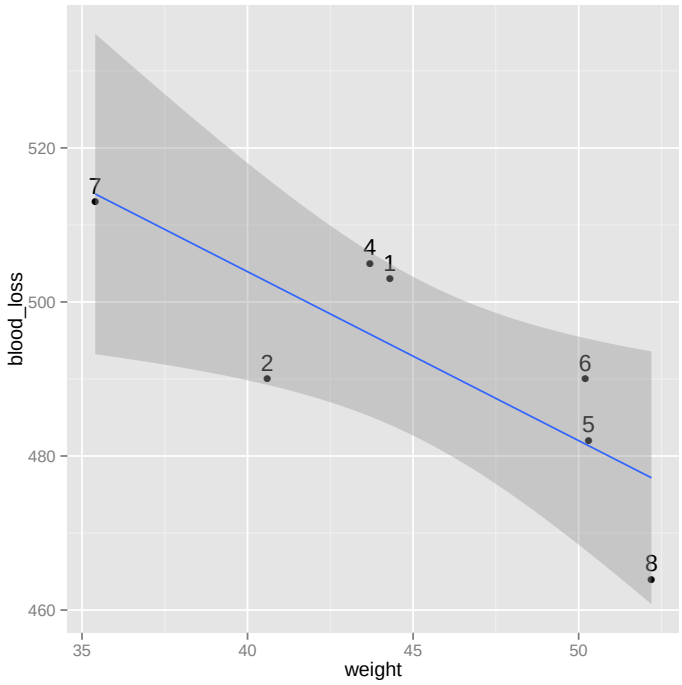
What changes by deleting case 3? The fitted line gets steeper (slope changes from -1.30 to -2.19), adjusted R^2 gets larger (up to 58% from 53%), and S changes from 11.7 to 10.6. Because the Weight values are much less spread out, $SE(\hat{\beta}_1)$ becomes quite a bit larger (to 0.714, up from 0.436) and we lose a degree of freedom for MS Error (which will penalize us on tests and CIs). Just about any quantitative statement we would want to make using CIs would be about the same either way since CIs will overlap a great deal, and our qualitative interpretations are unchanged (Blood Loss drops with Weight). Unless something shows up in the plots, I don't see any very important changes here.

```
lm.blood.wt.no3 <- lm(blood_loss ~ weight, data = thyroid, weights = wt)
# use summary() to get t-tests of parameters (slope, intercept)
summary(lm.blood.wt.no3)
```

```
##
## Call:
## lm(formula = blood_loss ~ weight, data = thyroid, weights = wt)
##
## Weighted Residuals:
##       1       2       3       4       5       6       7
##  8.5033 -12.6126   0.0000   9.1872   0.6641   8.4448  -1.0186
##       8
## -13.1683
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 591.6677    32.5668  18.168 9.29e-06 ***
## weight       -2.1935     0.7144  -3.071   0.0278 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.6 on 5 degrees of freedom
## Multiple R-squared:  0.6535,Adjusted R-squared:  0.5842
```

```
## F-statistic: 9.428 on 1 and 5 DF,  p-value: 0.02777
```

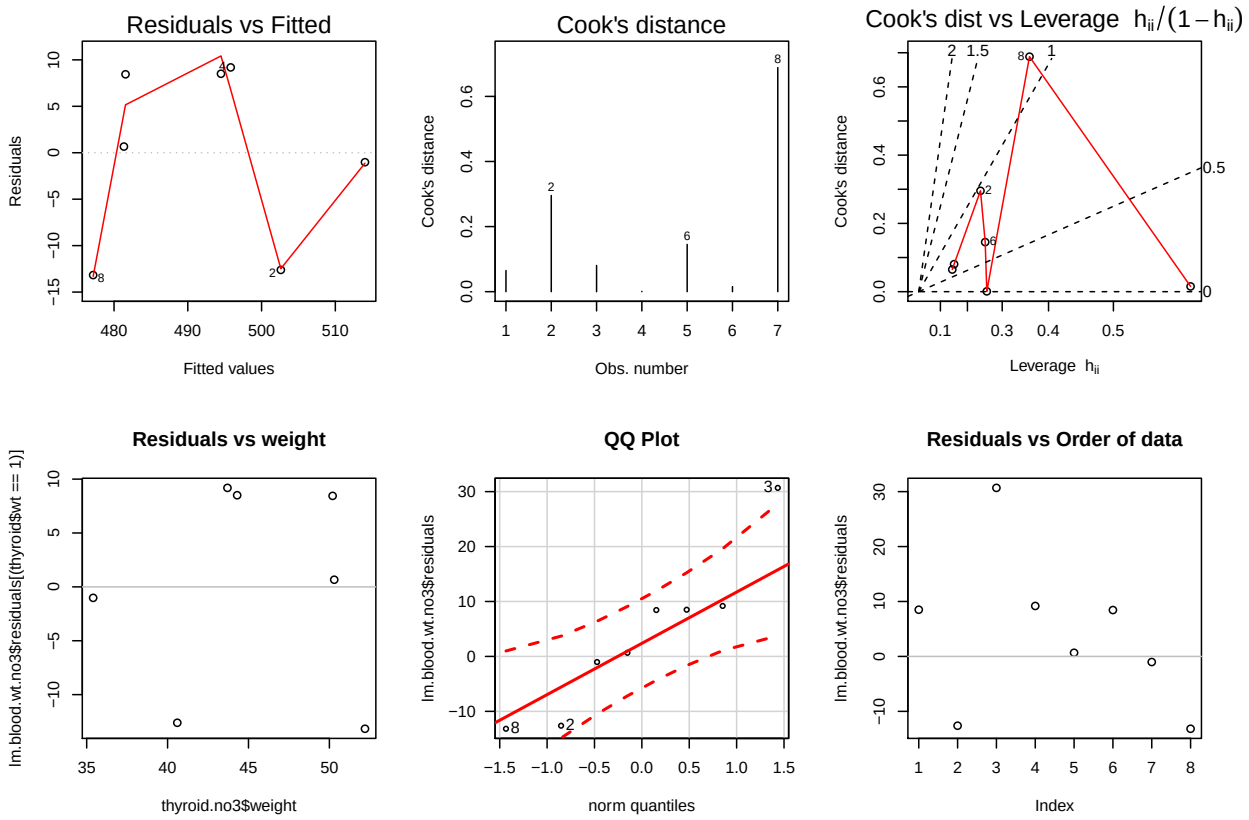
```
# exclude obs 3
thyroid.no3 <- subset(thyroid, wt == 1)
# ggplot: Plot the data with linear regression fit and confidence bands
library(ggplot2)
p <- ggplot(thyroid.no3, aes(x = weight, y = blood_loss, label = id))
p <- p + geom_point()
# plot labels next to points
p <- p + geom_text(hjust = 0.5, vjust = -0.5)
# plot regression line and confidence band
p <- p + geom_smooth(method = lm)
print(p)
```



Nothing very striking shows up in the residual plots, and no Cook's D values are very large among the remaining observations.

```
# plot diagnostics
par(mfrow=c(2,3))
plot(lm.blood.wt.no3, which = c(1,4,6))

# residuals vs weight
plot(thyroid.no3$weight, lm.blood.wt.no3$residuals[(thyroid$wt == 1)]
   , main="Residuals vs weight")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# Normality of Residuals
library(car)
# qq plot for studentized resid
#   las    =  1 : turns labels on y-axis to read horizontally
```

```
#   id.n   =  n : labels n most extreme observations, and outputs to console
#   id.cex =  1 : is the size of those labels
#   lwd    =  1 : line width
qqPlot(lm.blood.wt.no3$residuals, las = 1, id.n = 3, main="QQ Plot")

## 3 8 2
## 8 1 2

# residuals vs order of data
plot(lm.blood.wt.no3$residuals, main="Residuals vs Order of data")
  # horizontal line at zero
  abline(h = 0, col = "gray75")
```



How much difference is there in a practical sense? Examine the 95% prediction interval for a new observation at Weight = 50kg. Previously we saw that interval based on all 8 observations was from 457.1 to 517.8 ml of Blood Loss. Based on just the 7 observations the prediction interval is 451.6 to 512.4 ml. There really is no practical difference here.

```
# CI for the mean and PI for a new observation at weight=50
predict(lm.blood.wt    , data.frame(weight=50), interval = "prediction")
##        fit     lwr      upr
## 1 487.4257 457.098 517.7533
predict(lm.blood.wt.no3, data.frame(weight=50), interval = "prediction")
```

```
## Warning in predict.lm(lm.blood.wt.no3, data.frame(weight = 50), interval = "prediction"):
Assuming constant prediction variance even though model fit is weighted
##        fit      lwr      upr
## 1 481.9939 451.5782 512.4096
```

Therefore, while obs. 3 was potentially influential, whether the value is included or not makes very little difference in the model fit or relationship between Weight and BloodLoss.

8.10.2 Gesell data

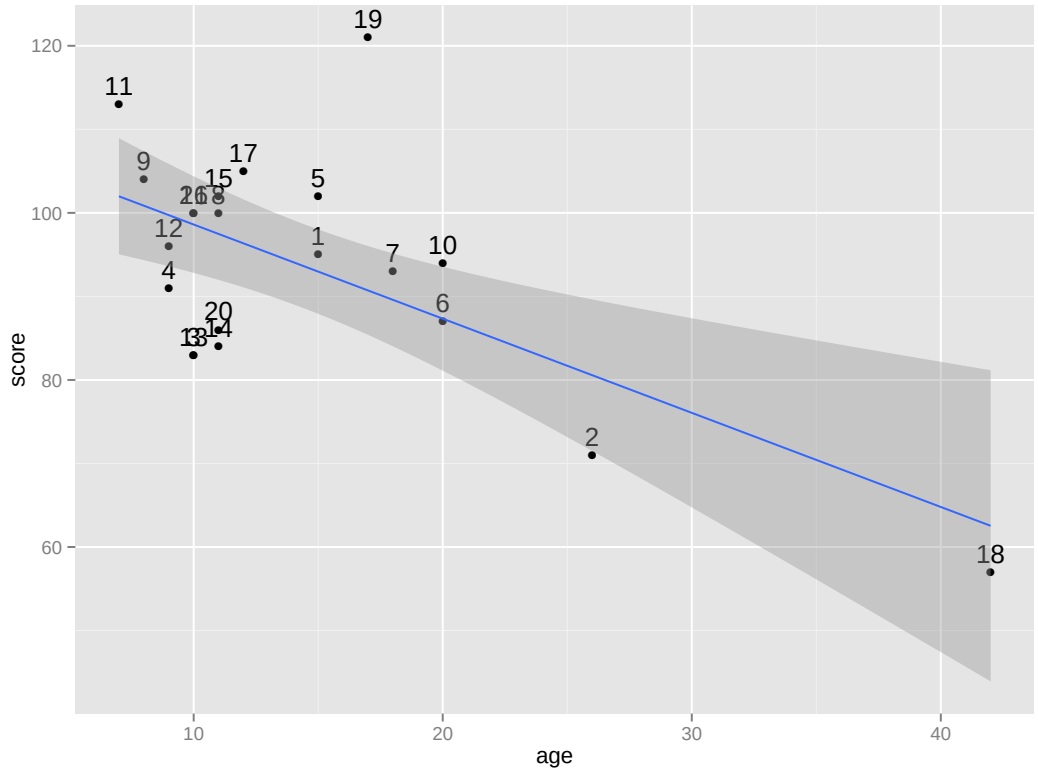
These data are from a UCLA study of cyanotic heart disease in children. The predictor is the age of the child in months at first word and the response variable is the Gesell adaptive score, for each of 21 children.

	id	age	score
1	1	15	95
2	2	26	71
3	3	10	83
4	4	9	91
5	5	15	102
6	6	20	87
7	7	18	93
8	8	11	100
9	9	8	104
10	10	20	94
11	11	7	113
12	12	9	96
13	13	10	83
14	14	11	84
15	15	11	102
16	16	10	100
17	17	12	105
18	18	42	57
19	19	17	121
20	20	11	86
21	21	10	100

Let us go through the same steps as before.

1. Plot Score versus Age. Comment on the relationship between Score and Age.

```
# ggplot: Plot the data with linear regression fit and confidence bands
library(ggplot2)
p <- ggplot(gesell, aes(x = age, y = score, label = id))
p <- p + geom_point()
# plot labels next to points
p <- p + geom_text(hjust = 0.5, vjust = -0.5)
# plot regression line and confidence band
p <- p + geom_smooth(method = lm)
print(p)
```



2. There are no obvious transformations to try here.
3. Fit a simple linear regression model. Provide an equation for the LS line. Does age at first word appear to be an "important predictor" of Gesell adaptive score? (i.e., is the estimated slope significantly different from zero?)

```
lm.score.age <- lm(score ~ age, data = gesell)
# use summary() to get t-tests of parameters (slope, intercept)
summary(lm.score.age)
```

```
##
## Call:
## lm(formula = score ~ age, data = gesell)
##
```

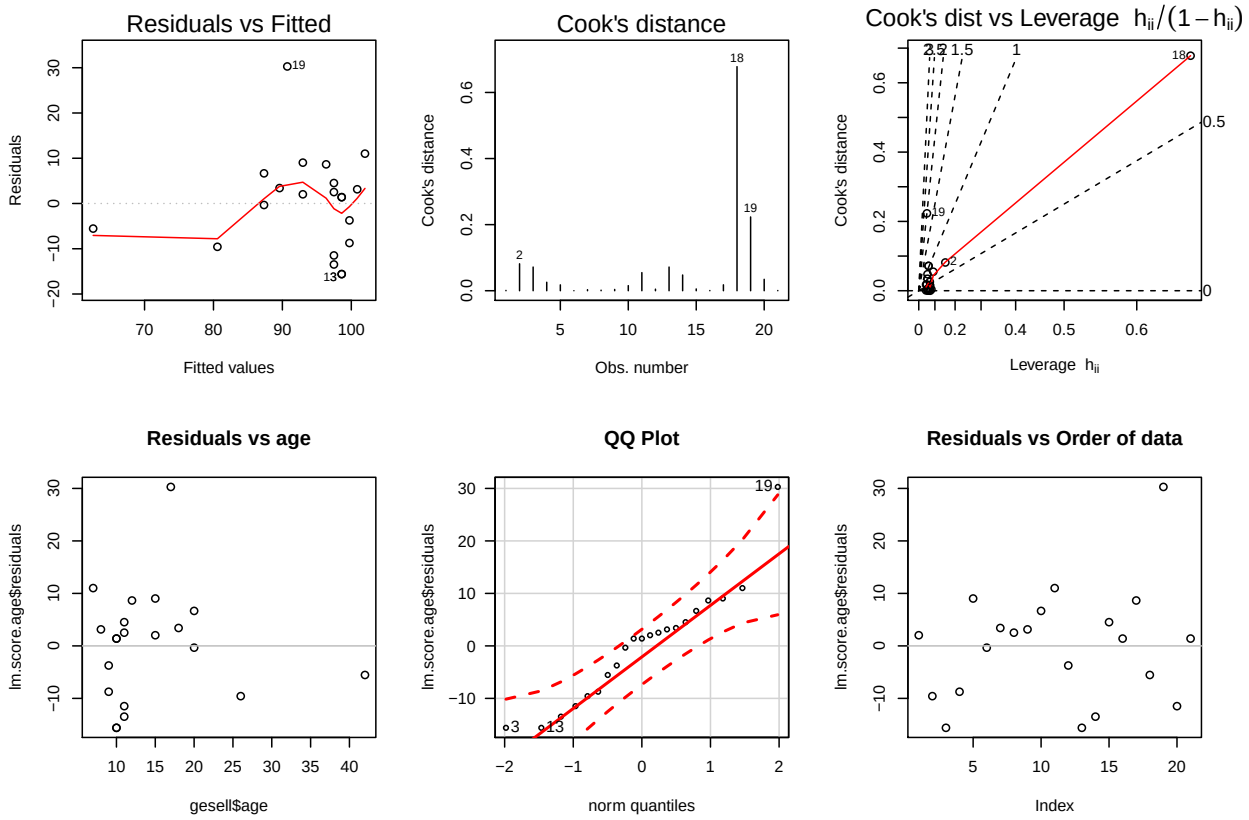
```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.604  -8.731   1.396   4.523  30.285
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 109.8738     5.0678  21.681 7.31e-15 ***
## age          -1.1270     0.3102  -3.633  0.00177 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.02 on 19 degrees of freedom
## Multiple R-squared:   0.41,Adjusted R-squared:  0.3789
## F-statistic:  13.2 on 1 and 19 DF,  p-value: 0.001769
```

4. Do these plots suggest any inadequacies with the model?

```
# plot diagnistics
par(mfrow=c(2,3))
plot(lm.score.age, which = c(1,4,6))

# residuals vs weight
plot(gesell$age, lm.score.age$residuals, main="Residuals vs age")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# Normality of Residuals
library(car)
qqPlot(lm.score.age$residuals, las = 1, id.n = 3, main="QQ Plot")
```

```
## 19  3 13
## 21  1  2
```

```
# residuals vs order of data
plot(lm.score.age$residuals, main="Residuals vs Order of data")
  # horizontal line at zero
  abline(h = 0, col = "gray75")
```



5. Observations 18 and 19 stand out with relatively high Cook's D. The cutoff line is only a rough guideline. Those two were flagged with high influence and standardized residual, respectively, also. Be sure to examine the scatter plot carefully to see why 18 and 19 stand out.
6. Consider doing two additional analyses: Analyze the data after omitting case 18 only and analyze the data after omitting case 19 only. Refit the regression model for each of these two scenarios. Provide a summary table such as the following, giving the relevant summary statistics for the three analyses. Discuss the impact that observations 18 and 19 have individually on the fit of the model.

   When observation 18 is omitted, the estimated slope is not significantly different from zero (p-value = 0.1489), indicating that age is not an important predictor of Gesell score. This suggests that the significance of age as a predictor in the original analysis was due solely to the presence of observation 18. Note the dramatic decrease in R^2 after deleting observation 18.

The fit of the model appears to improve when observation 19 is omitted. For example, R^2 increases noticeably and the p-value for testing the significance of the slope decreases dramatically (in a relative sense). These tendencies would be expected based on the original plot. However, this improvement is misleading. Once observation 19 is omitted, observation 18 is much more influential. Again the significance of the slope is due to the presence of observation 18.

Feature	Full data	Omit 18	Omit 19
b_0	109.87	105.63	109.30
b_1	-1.13	-0.78	-1.19
$SE(b_0)$	5.07	7.16	3.97
$SE(b_1)$	0.31	0.52	0.24
R^2	0.41	0.11	0.57
p-val for $H_0 : \beta_1 = 0$	0.002	0.149	0.000

Can you think of any reasons to justify doing the analysis without observation 18?

If you include observation 18 in the analysis, you are assuming that the mean Gesell score is linearly related to age over the entire range of observed ages. Observation 18 is far from the other observations on age (age for observation 18 is 42; the second highest age is 26; the lowest age is 7). There are no children with ages between 27 and 41, so we have no information on whether the relationship is roughly linear over a significant portion of the range of ages. I am comfortable deleting observation 18 from the analysis because it's inclusion forces me to make an assumption that I can not check using these data. I am only willing to make predictions of Gesell score for children with ages roughly between 7 and 26. However, once this point is omitted, age does not appear to be an important predictor.

A more complete analysis would delete observation 18 and 19 together. What would you expect to see if you did this?

8.11 Weighted Least Squares

Earlier I indicated that nonconstant error variance can be addressed (sometimes) with weighted least squares. The *scedastic function* is the conditional variance of Y given X. Y is said to be **heteroscedastic** if it's variance depends on the value of X (variance changes), and **homoscedastic** if it's variance does not depend on X (constant variance).

Recall the LS (OLS or ordinary LS) line chooses the values of β_0 and β_1 that minimize

$$\sum_{i=1}^{n}\{Y_i - (\beta_0 + \beta_1 X_i)\}^2$$

over all possible choices of β_0 and β_1. The weighted LS (WLS) line chooses the values of β_0 and β_1 that minimize

$$\sum_{i=1}^{n} w_i\{Y_i - (\beta_0 + \beta_1 X_i)\}^2$$

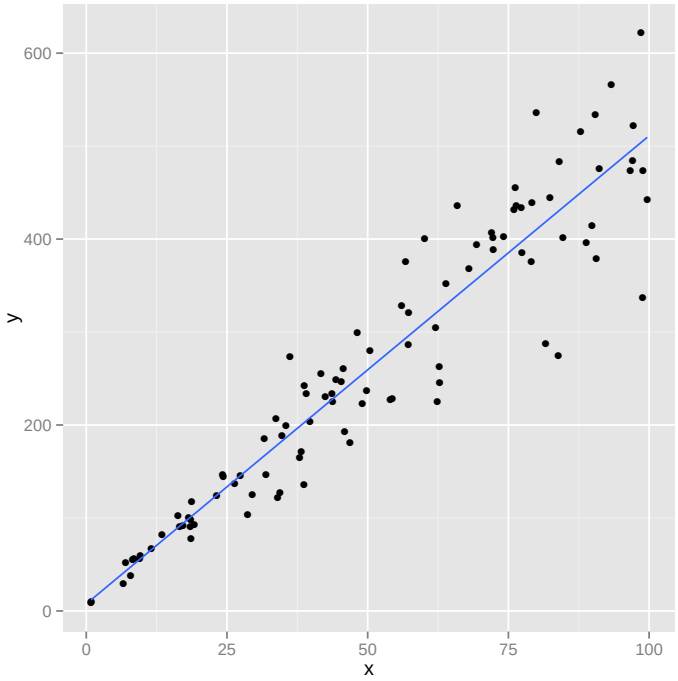
over all possible choices of β_0 and β_1. If $\sigma_{Y|X}$ depends up X, then the correct choice of weights is inversely proportional to *variance*, $w_i \propto \sigma^2_{Y|X}$.

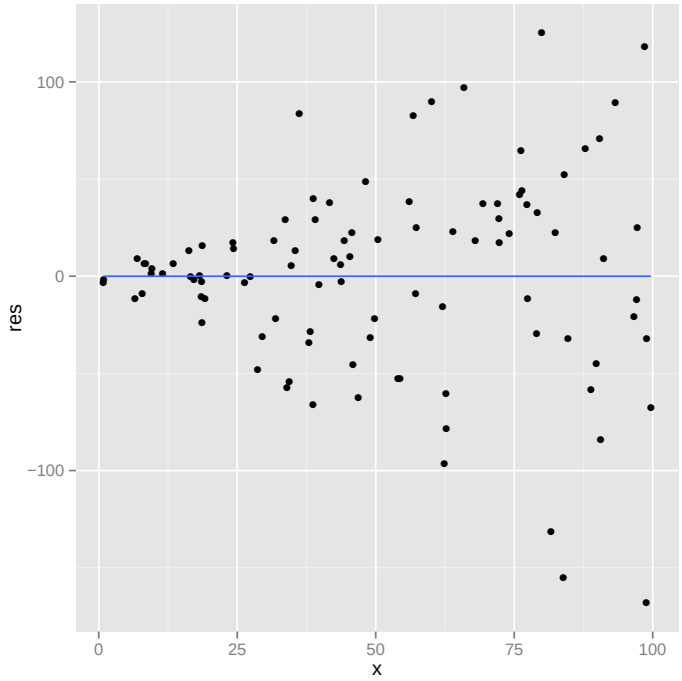
Consider the following data and plot of y vs. x and standardized OLS residuals vs x. It is very clear that variability increases with x.

```
#### Weighted Least Squares
# R code to generate data
set.seed(7)
n <- 100
# 1s, Xs uniform 0 to 100
X <- matrix(c(rep(1,n),runif(n,0,100)), ncol=2)
# intercept and slope (5, 5)
beta <- matrix(c(5,5),ncol=1)
# errors are X*norm(0,1), so variance increases with X
e <- X[,2]*rnorm(n,0,1)
# response variables
y <- X %*% beta + e

# put data into data.frame
wlsdat <- data.frame(y, x = X[,2])

# fit regression
lm.y.x <- lm(y ~ x, data = wlsdat)

# put residuals in data.frame
wlsdat$res <- lm.y.x$residuals
```

```
# ggplot: Plot the data with linear regression fit
library(ggplot2)
p <- ggplot(wlsdat, aes(x = x, y = y))
```

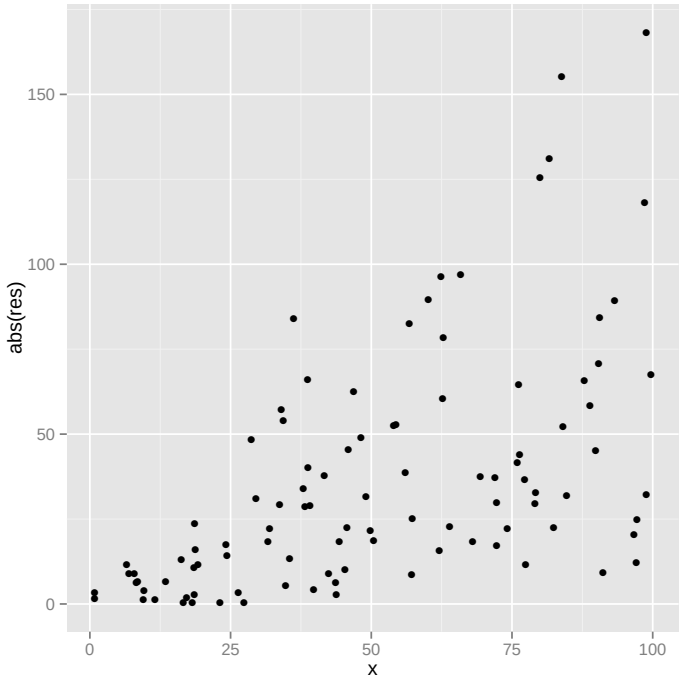
```
p <- p + geom_point()
p <- p + geom_smooth(method = lm, se = FALSE)
print(p)


# ggplot: Plot the residuals
library(ggplot2)
p <- ggplot(wlsdat, aes(x = x, y = res))
p <- p + geom_point()
p <- p + geom_smooth(method = lm, se = FALSE)
print(p)
```





In order to use WLS to solve this problem, we need some form for $\sigma^2_{Y|X}$. Finding that form is a real problem with WLS. It can be useful to plot the absolute value of the standardized residual vs. x to see if the top boundary seems to follow a general pattern.

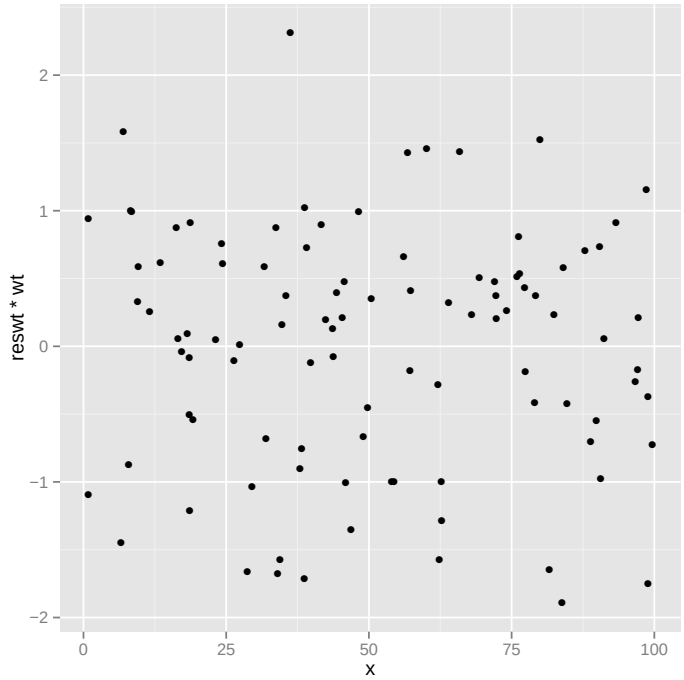
```
# ggplot: Plot the absolute value of the residuals
library(ggplot2)
p <- ggplot(wlsdat, aes(x = x, y = abs(res)))
p <- p + geom_point()
print(p)
```



It is plausible the upper boundary is linear, so let us try $w_i = \frac{1}{x^2}$. Standardized residuals from this WLS fit look very good. Note that raw (non-standardized) residuals will still have the same pattern — it is essential to use standardized residuals here.

```
# fit regression
lm.y.x.wt <- lm(y ~ x, data = wlsdat, weights = x^(-2))

# put residuals in data.frame
wlsdat$reswt <- lm.y.x.wt$residuals
wlsdat$wt <- lm.y.x.wt$weights^(1/2)
```

```
# ggplot: Plot the absolute value of the residuals
library(ggplot2)
p <- ggplot(wlsdat, aes(x = x, y = reswt*wt))
p <- p + geom_point()
print(p)
```



Compare also the OLS fitted equation:

```
summary(lm.y.x)
```

```
##
## Call:
## lm(formula = y ~ x, data = wlsdat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -168.175  -24.939    2.542   24.973  125.366
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.6308    10.4256   0.732    0.466
## x             5.0348     0.1791  28.116   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 50.53 on 98 degrees of freedom
## Multiple R-squared:  0.8897,Adjusted R-squared:  0.8886
## F-statistic: 790.5 on 1 and 98 DF,  p-value: < 2.2e-16
```

to the WLS fitted equation:

```
summary(lm.y.x.wt)
```

```
##
## Call:
## lm(formula = y ~ x, data = wlsdat, weights = x^(-2))
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -1.8939 -0.6707  0.1777  0.5963  2.3132
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.09309    0.54931   9.272 4.61e-15 ***
## x            5.10690    0.09388  54.399  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8902 on 98 degrees of freedom
## Multiple R-squared:  0.9679,Adjusted R-squared:  0.9676
## F-statistic:  2959 on 1 and 98 DF,  p-value: < 2.2e-16
```

Clearly the weighted fit looks better, although note that everything is based on the weighted SS. In practice it can be pretty difficult to determine the correct set of weights, but WLS works much better than OLS if appropriate. I actually simulated this data set using $\beta_0 = \beta_1 = 5$. Which fit actually did better?