Introduction to Data Science

DSA1101

Semester 1, 2018/2019
Week 1, 17 August

Data manipulation with R (continued)

- We can access individual columns of the R dataframe sales in many different ways

```
> head(sales)
  cust_id sales_total num_of_orders gender
1  100001      800.64             3      F
2  100002      217.53             3      F
3  100003       74.58             2      M
4  100004      498.60             3      M
5  100005      723.11             4      F
6  100006       69.43             2      F
```

- A typical representation is dataframe[row indices, column indices]

```
> #extract gender data
> head(sales$gender)
[1] F F M M F F
Levels: F M
> #extract 4th dataframe column
> head(sales[,4])
[1] F F M M F F
Levels: F M
> #extract 1st two rows of dataframe
> head(sales[1:2,])
  cust_id sales_total num_of_orders gender
1  100001      800.64             3      F
2  100002      217.53             3      F
```

- A typical representation is dataframe[row indices, column indices]

```
> #extract 1st, 3rd and 4th columns of dataframe
> head(sales[,c(1,3,4)])
  cust_id num_of_orders gender
1  100001             3      F
2  100002             3      F
3  100003             2      M
4  100004             3      M
5  100005             4      F
6  100006             2      F
> #extract total sales and gender columns
> head(sales[,c("sales_total","gender")])
  sales_total gender
1      800.64      F
2      217.53      F
3       74.58      M
4      498.60      M
5      723.11      F
6       69.43      F
```

- Subsets of dataframes can be extracted according to defined rules.

```
> #extract all records whose gender is female
> head(sales[sales$gender=="F",])
   cust_id sales_total num_of_orders gender
1   100001      800.64             3      F
2   100002      217.53             3      F
5   100005      723.11             4      F
6   100006       69.43             2      F
9   100009      364.63             2      F
11  100011      216.41             1      F
>
> #extract all records with total sales above 500
> head(sales[sales$sales_total>500,])
   cust_id sales_total num_of_orders gender
1   100001      800.64             3      F
5   100005      723.11             4      F
14  100014     1044.40             7      M
22  100022      580.64             5      M
36  100036      710.93             4      M
57  100057      659.04             3      M
```

Basic data visualisation in R

- Good data visualisation technique is integral to data exploration and can facilitate facilitate model building and model validation immensely.
- For multi-dimensional data plots are useful in revealing their possible inter-relationships.
- As we discussed in the 1st lecture, presentation and visualization is vital to communicating and understanding data analytics results.

Basic data visualisation in R

- R provides many sophisticated data visualisation tools.
- The most basic function is `plot()` which produces a scatter plot.
- Other common plotting functions include `barplot()` for frequency plot with vertical or horizontal bars and `hist()` for producing histograms.
- Many R libraries (e.g. `ggplot`, `lattice`) with more sophisticated and visually pleasing plotters are available.
- We will learn how to install and load R libraries later in the course.

Basic data visualisation in R

- In the example below, a simple scatter plot is produced from our `sales` data.
- The plot `scatter_sales.png` is saved to the current working directory.

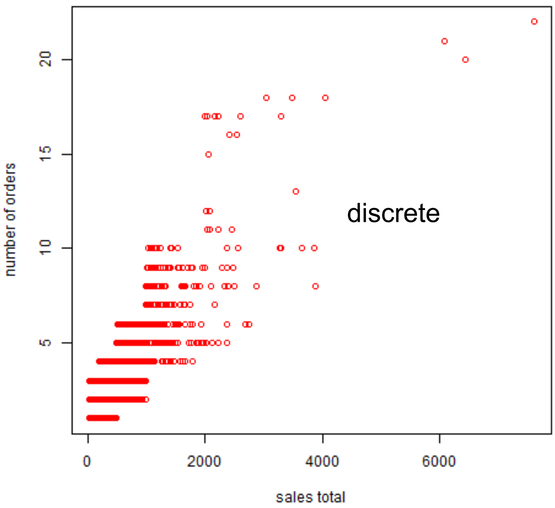
```r
#simple scatter plot of number of sales versus total sales

png("scatter_sales.png")

plot(x=sales$sales_total, y=sales$num_of_orders,
     xlab="sales total", ylab="number of orders", col="red")   (lab = label, col = color)

dev.off()
```

Basic data visualisation in R



Basic data visualisation in R

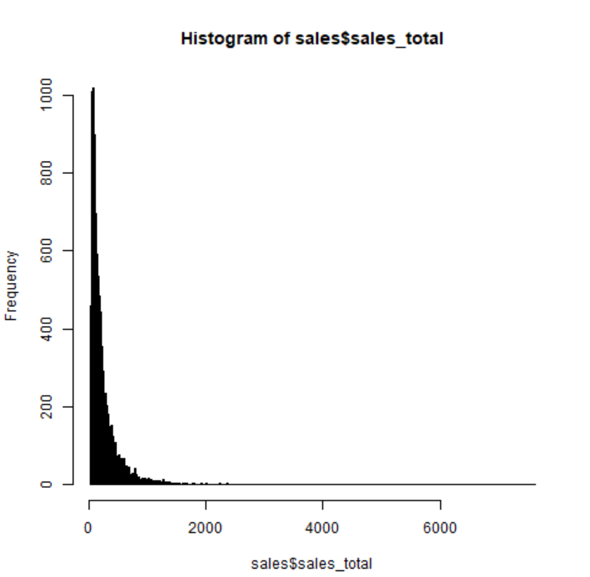
- Histograms are used to represent the density of a continuous data from the observations.
- In the example below, we generated a histogram for total sales.
- The plot `hist_sales.png` is saved to the current working directory.

```
#histogram of total sales

> png("hist_sales.png")
>
> hist(x=sales$sales_total, breaks=500,col="red")
> max(sales$sales_total)
[1] 7606.09
> min(sales$sales_total)
[1] 30.02
> dev.off()
```

more breaks more boxes

Basic data visualisation in R



Linear regression

Linear regression

- Suppose we are in a management consulting company and we have data on sales for a particular product, as well as advertising budgets for TV, radio and newspaper media, for $n = 200$ different markets.
- Sales number is expressed in thousands of units.
- Budgets are expressed in thousands of dollars

```
> advert=read.csv("Advertising.csv")
> head(advert)
  X    TV radio newspaper sales
1 1 230.1  37.8      69.2  22.1
2 2  44.5  39.3      45.1  10.4
3 3  17.2  45.9      69.3   9.3
4 4 151.5  41.3      58.5  18.5
5 5 180.8  10.8      58.4  12.9
6 6   8.7  48.9      75.0   7.2
```

Basic data visualisation in R

- Suppose the client who works in sales is engaging our consultation services and has provided the data.
- He or she is interested in data analytics which can potentially provide insights into how sales number can be increased.

Linear regression

- Is there a linear relationship between TV advertisement budget and sales? How strong is this relationship?
- Do all three media affect sales, or just one or two of them do?
- How accurate can we estimate the linear effects on sales?
- How accurate can we predict future sales , based on given level of advertising budget?

Simple linear regression

- Regression is the basic tool used in statistical modelling
- In simple linear regression we are given a response or a dependent variable y and a single explanatory variable x, and we fit the model,

  $$y = \beta_0 + \beta_1 x + \epsilon$$

  where β_0 (the intercept) and β_1 (the slope) are two unknown parameters and ϵ is an "error term". We will ignore ϵ for now, until later when we perform inference.
- We reframe our client's question as the linear model

  $$\text{sales} = \beta_0 + \beta_1 \text{TV} + \epsilon$$

- We need to estimate the unknown parameters β_0 and β_1 based on the $n = 200$ data points.

Simple linear regression

- First, we generate pair-wise scatter plots for a preliminary look at the data using the R package `lattice`.
- To install a package in R, we can use the following command:

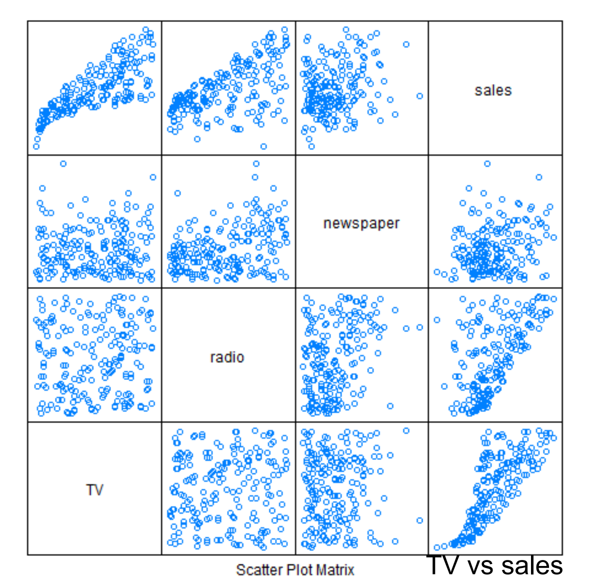
```
> install.packages("lattice")
```

  and then select a mirror site to download the package.

- After package has been installed, we have to load it first before using its `splom` function:

```
> library(lattice)
>
> png("scatter.png")
> #skip ID line, so just columns 2,3,4,5 in data
> splom(~advert[,c(2:5)], groups=NULL, data=advert,
+ axis.line.tck = 0,
+ axis.text.alpha = 0)
>
> dev.off()
```

Simple linear regression



Simple linear regression

- A strong positive trend is observed for sales as a function of TV advertising budget.
- We will help to quantify this effect for our client by estimating β_0 and β_1. Recall that our model is

$$\text{sales} = \beta_0 + \beta_1 \text{TV} + \epsilon$$

- We want the estimated values, denoted as $\hat{\beta}_0$ and $\hat{\beta}_1$ to be such that the resulting line is as close as possible to the $n = 200$ data points.
- This is achieved via the method of least squares.

Method of least squares

- The predicted sales for a given TV budget using our model is $\widehat{\text{sales}} = \hat{\beta}_0 + \hat{\beta}_1 \text{TV}$.
- For the i^{th} data point, the i^{th} residual is difference between the observed and predicted sales for the given TV budget in the data point:

$$e_i = \text{sales}_i - \widehat{\text{sales}}_i = \text{sales}_i - \left(\hat{\beta}_0 + \hat{\beta}_1 \text{TV}_i\right)$$

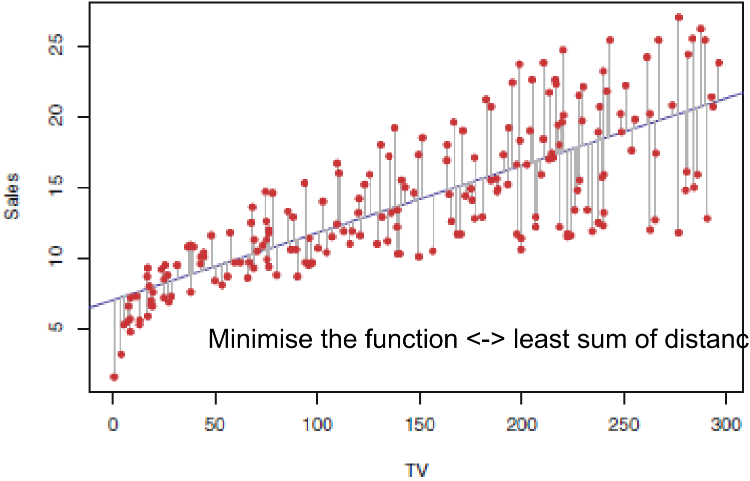
- The i^{th} squared residual is

$$e_i^2 = \left[\text{sales}_i - \left(\hat{\beta}_0 + \hat{\beta}_1 \text{TV}_i\right)\right]^2$$

- We want to minimize the residual sum of squares,

$$RSS = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left[\text{sales}_i - \left(\hat{\beta}_0 + \hat{\beta}_1 \text{TV}_i\right)\right]^2 .$$

Method of least squares



Minimise the function <-> least sum of distances

Figure 3.1 from James et al. [2013].

Method of least squares

****exam

- We would like to minimize the function $RSS = \sum_{i=1}^{n} \left[y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 x_i \right) \right]^2$ in terms of $\hat{\beta}_0$ and $\hat{\beta}_1$
- Recall that to minimize RSS, we can set the derivatives to zero and solve
- $\frac{\partial RSS}{\partial \hat{\beta}_0} = \sum_{i=1}^{n} 2 \times \left[y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 x_i \right) \right] \times (-1) = 0.$
  $\rightarrow n\hat{\beta}_0 + \hat{\beta}_1 \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} y_i = 0$
  $\rightarrow \hat{\beta}_0 + \hat{\beta}_1 \bar{x} - \bar{y} = 0 \qquad (1)$
- $\frac{\partial RSS}{\partial \hat{\beta}_1} = \sum_{i=1}^{n} 2 \times \left[y_i - \left(\hat{\beta}_0 + \hat{\beta}_1 x_i \right) \right] \times (-x_i) = 0.$
  $\rightarrow \hat{\beta}_0 \sum_{i=1}^{n} x_i + \hat{\beta}_1 \sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n} x_i y_i = 0 \qquad (2)$

Method of least squares

- Substitute equation (1), $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$, into (2) gives $\hat{\beta}_1 = \{\sum_{i=1}^{n} x_i y_i - \bar{y} \sum_{i=1}^{n} x_i\} / \{\sum_{i=1}^{n} x_i^2 - \bar{x} \sum_{i=1}^{n} x_i\}$
- Let's calculate $\hat{\beta}_1$ in R:

```
> beta1=
+ (sum(x*y)-mean(y)*sum(x))/
+ (sum(x^2)-mean(x)*sum(x))
> beta1
[1] 0.04753664
```

- $\hat{\beta}_0$ follows from (1):

```
> beta0=
+ mean(y)-beta1*mean(x)
> beta0
[1] 7.032594
```

Method of least squares

- We can use the lm() function in *R* to fit a linear model:

```
> lm(sales~TV, data=advert)

Call:
lm(formula = sales ~ TV, data = advert)

Coefficients:
(Intercept)           TV
    7.03259      0.04754 <- slope
```

- In words, sales is expected to increase by approximately 47.5 units per every $1,000 increase in TV advertising budget.

Inference

- So far we have ignored the error term ϵ in the linear model

$$y = \beta_0 + \beta_1 x + \epsilon$$

- If we think of ϵ as a random variable, then there is inherent uncertainty in our least squares estimates $\hat{\beta}_0$ and $\hat{\beta}_1$.
- One way to quantify this uncertainty is by computing the standard errors $SE(\hat{\beta}_0)$ and $SE(\hat{\beta}_1)$.
- The `lm()` function in R computes these standard errors assuming (by default) that the error terms are independent, normally distributed with mean equal to zero and constant variance.

Method of least squares

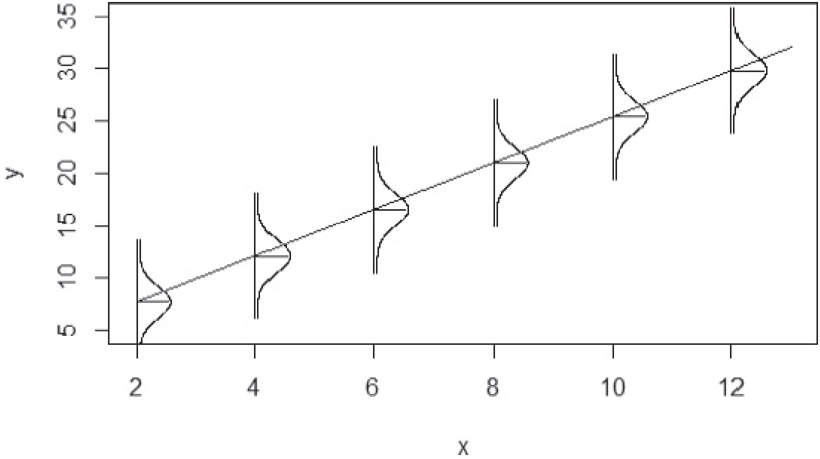


Figure 6-3 from Dietrich et al. [2015]. Normally distributed errors.

Inference

-
```
> linear.reg<- lm(sales~TV, data=advert)
> summary(linear.reg)

Call:
lm(formula = sales ~ TV, data = advert)

Residuals:
    Min      1Q  Median      3Q     Max
-8.3860 -1.9545 -0.1913  2.0671  7.2124

                                 *
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 7.032594   0.457843   15.36   <2e-16
TV          0.047537   0.002691   17.67   <2e-16
```

Confidence intervals

- The standard errors $SE(\hat{\beta}_0)$ and $SE(\hat{\beta}_1)$ can be used to compute *confidence intervals*.
- A 95% confidence interval is defined as a range of values such that, were this procedure to compute the range be repeated on numerous samples, the fraction of calculated confidence intervals (which would differ for each sample) that encompass the true parameter value would tend toward 95%.
- For linear regression, the 95% confidence interval for β_0 is approximately

$$\left[\hat{\beta}_0 - 2 \times SE(\hat{\beta}_0), \hat{\beta}_0 + 2 \times SE(\hat{\beta}_0)\right]$$

- Similarly, the 95% confidence interval for β_1 is approximately

$$\left[\hat{\beta}_1 - 2 \times SE(\hat{\beta}_1), \hat{\beta}_1 + 2 \times SE(\hat{\beta}_1)\right]$$

Hypothesis testing

- The standard errors $SE(\hat{\beta}_0)$ and $SE(\hat{\beta}_1)$ can also be used to perform hypothesis testing.
- The most common hypothesis test involves testing the null hypothesis of

$$H_0 : \text{There is no relationship between } x \text{ and } y$$

- In terms of our model parameters, this corresponds to testing

$$H_0 : \beta_1 = 0$$

versus

$$H_0 \quad \beta_1 \neq 0$$

- In practice, we compute a *t-statistic* given by

$$t = \frac{\hat{\beta}_1 - 0}{SE(\hat{\beta}_1)}$$

Hypothesis testing

- Under H_0, the statistic t has a *t*-distribution with $n-2$ degrees of freedom for simple linear regression
- The *p-value* is the probability of observing any number equal or greater than $|t|$ in absolute value under H_0.
- We can compute *p-values* directly from the *t-statistic*,

```
> summary(linear.reg)$coef
              Estimate  Std. Error  t value     Pr(>|t|)
(Intercept) 7.03259355 0.457842940 15.36028 1.40630e-35
TV          0.04753664 0.002690607 17.66763 1.46739e-42
```

- Or we can compute *p-values* directly from the *t-statistic* using the `pt()` function in R: no need to calculate p-value

```
> 2*pt(-abs(summary(linear.reg)$coef[2,3]),df=(200-2))
[1] 1.46739e-42
```

Hypothesis testing

- If *p-value*$<\alpha$, then we reject H_0 and conclude that there is a linear relationship between x and y at the α significance level.
- Typically α is selected to be equal to 0.05 or 0.01.

```
> summary(linear.reg)$coef
              Estimate  Std. Error  t value     Pr(>|t|)
(Intercept) 7.03259355 0.457842940 15.36028 1.40630e-35
TV          0.04753664 0.002690607 17.66763 1.46739e-42
```

Confidence interval on expected outcome

- We are often interested to obtain a *confidence interval* for the expected outcome.
- In our example, we are interested to find out among all the different markets of interest, what is the expected total sales for a given level of TV advertising budget.
- Using the `predict()` function in R, a confidence interval on the expected outcome can be obtained for a given set of predictor variable values.
- Suppose we are interested in a confidence interval for the expected total sales when we spend $200,000 on TV advertising.

Confidence interval on expected outcome

```
> TV=200
> new_pt=data.frame(TV)
> predict(linear.reg, new_pt, level=0.95, interval
  ="confidence")
       fit      lwr      upr
1 16.53992 16.00567 17.07418
```

- Therefore, when we spend \$200,000 on TV advertising, the expected total sales is approximately 16,540 units with a 95% confidence interval of $(16006, 17074)$.

Prediction interval for a particular outcome

- Suppose we are also interested to predict the actual total sales in a particular new market for a given level of TV advertising budget.
- There is additional variability associated with sampling a particular new market from among all the markets, and therefore we cannot use the previous *confidence interval*.
- The `predict()` function in R also provides the ability to calculate upper and lower bounds on a particular outcome; the range defined by these bounds is called a *prediction interval*.
- Suppose we are interested in a prediction interval for total sales in a new market when we spend $200,000 on TV advertising.

Prediction interval for a particular outcome

```
> TV=200
> new_pt=data.frame(TV)
> predict(linear.reg, new_pt, level=0.95, interval
    ="prediction")
       fit      lwr      upr
1 16.53992 10.09162 22.98822
```

- Therefore, when we spend $200,000 on TV advertising, the predicted total sales in a particular new market is approximately 16,540 units with a 95% prediction interval of $(10091, 22988)$.
- Notice that this 95% *prediction interval* is wider than the 95% *confidence interval* computed previously, due to additional variability of sampling a particular new market from among all the markets of interest.

Model diagnostics

Evaluating the linearity assumption

- A major assumption in linear regression modeling is that the relationship between the input variables and the outcome variable is linear.
- The most fundamental way to evaluate such a relationship is to plot the outcome variable against the input variable(s).
- If the linear relationship does not seem to apply, it is often useful to do any of the following:

(a) Transform the outcome variable, e.g. by taking the logarithm

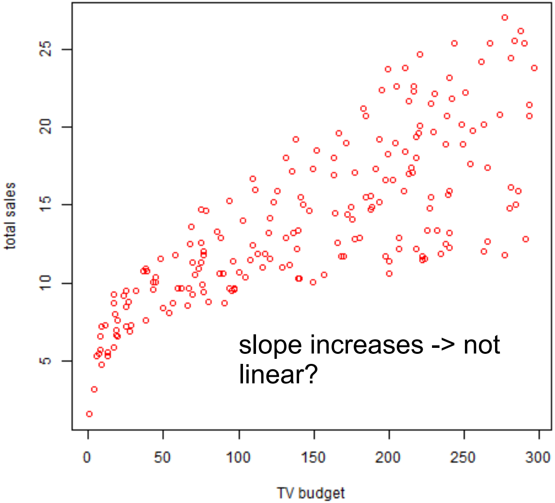
(b) Transform the input variable(s)

(c) Add extra input variables or terms to the regression model

Evaluating the linearity assumption

- Let's plot total sales versus TV advertising budget to evaluate the linearity assumption:

```
> png("TV_sales.png")
>
> plot(x=advert$TV, y=advert$sales,
+       xlab="TV budget", ylab="total sales", col="red")
>
> dev.off()
```



slope increases -> not linear?

- There appears to be a non-linear relationship between total sales and TV advertising budget
- We may want to, for example, add the term $\texttt{TV}^2$ as an additional input variable in the regression model and perform further model diagnostics higher order-> more concise

$$\text{sales} = \beta_0 + \beta_1 \text{TV} + \beta_2 \text{TV}^2 + \epsilon$$

- This leads to *multiple linear regression*

Assessing the accuracy of the model

- Suppose we are interested to quantify the extent to which the model fits the data
- The quality of a linear regression fit can be assessed using the residual standard error (RSE).
- Larger RSE indicates poorer model fit
- Recall that for our example,

$$RSS = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} \left[\text{sales}_i - \left(\hat{\beta}_0 + \hat{\beta}_1 \text{TV}_i\right)\right]^2$$

$$= \sum_{i=1}^{n} \left[\text{sales}_i - \hat{\text{sales}}_i\right]^2$$

- Then RSE in simple linear regression is defined as

$$RSE = \sqrt{\frac{1}{n-2} RSS}$$

Assessing the accuracy of the model

- We can calculate RSS directly: *RSE

```
> sqrt(sum((advert$sales-linear.reg$fitted)^2)/(
    length(advert$sales)-2))
[1] 3.258656
```

- We can also read off the RSS after fitting the linear model in R:

```
> summary(linear.reg)
Call:
lm(formula = sales ~ TV, data = advert)

Residuals:
    Min       1Q   Median       3Q      Max
-8.3860 -1.9545 -0.1913  2.0671  7.2124
# I have skipped some parts of the output here
Residual standard error: 3.259 on 198 degrees of
    freedom
```

Assessing the accuracy of the model

- The R^2 statistic is another measure of the fit of the model
- Since R^2 takes on a value between 0 and 1, it is independent of the scale of the outcome, unlike RSS.
- To calculate R^2, we use the formula

$$R^2 = \frac{TSS - RSS}{TSS} = 1 - \frac{RSS}{TSS},$$

where $TSS = \sum_{i=1}^{n}(y_i - \bar{y})^2$ is the *total sum of squares*

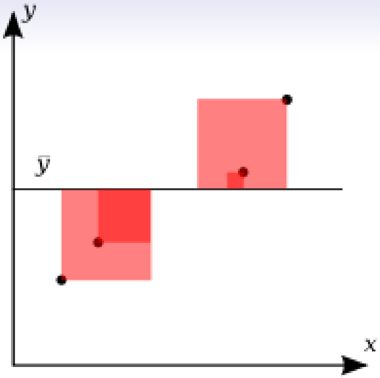
- In our example,

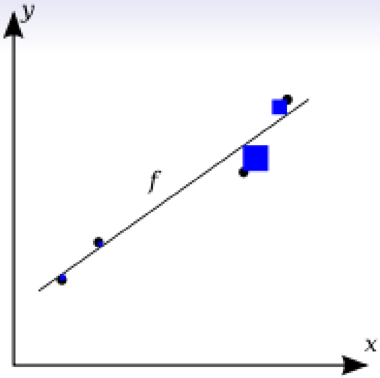
$$TSS = \sum_{i=1}^{n}\left[\text{sales}_i - \overline{\text{sales}}\right]^2,$$

- TSS measures the total variance in the response Y, and can be thought of as the amount of variability inherent in the response before the regression is performed.

Assessing the accuracy of the model

- In contrast,*RSS* measures the amount of variability that is left unexplained after performing the regression.
- $TSS - RSS$ measures the amount of variability in the response that is explained (or removed) by performing the regression
- Hence, R^2 measures the proportion of variability in outcome Y that can be explained using the predictor variable X with the linear model.
- Larger R^2 indicates better model fit.





$R^2 = 1 - \frac{RSS}{TSS}$. The areas of the blue squares represent the squared residuals with respect to the fitted regression line. The areas of the red squares represent the squared residuals with respect to the average value $\bar{y}$. (source: Wikipedia)

Assessing the accuracy of the model

- We can calculate R^2 value directly:

```
> RSS=sum((advert$sales-linear.reg$fitted)^2)
> TSS=sum((advert$sales-mean(advert$sales))^2)
> R2=1-RSS/TSS
> R2
[1] 0.6118751
```

- We can also read off the R^2 value after fitting the linear model in R:

```
> summary(linear.reg)
Call:
lm(formula = sales ~ TV, data = advert)

Residuals:
    Min      1Q  Median      3Q     Max
-8.3860 -1.9545 -0.1913  2.0671  7.2124
# I have skipped some parts of the output here
Multiple R-squared:  0.6119,     Adjusted R-squared
     :  0.6099
```

Checking the normality assumption

- We have assumed that the error terms ϵ in the simple linear model

$$\text{sales} = \beta_0 + \beta_1 \text{TV} + \epsilon$$

  are normally distributed with a mean of zero and constant variance
- To check for constant variance across all outcome (sales in our example) values along the regression line, use a simple plot of the residuals against the fitted outcome values.
- Recall that the i^{th} fitted value is $\widehat{\text{sales}}_i = \hat{\beta}_0 + \hat{\beta}_1 \text{TV}_i$, and the i^{th} residual is

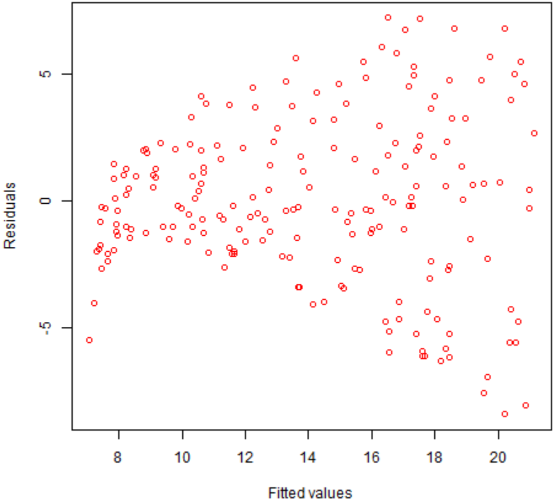
$$e_i = \text{sales}_i - \widehat{\text{sales}}_i = \text{sales}_i - \left(\hat{\beta}_0 + \hat{\beta}_1 \text{TV}_i\right)$$

Checking the normality assumption

- Because of the importance of examining the residuals, the `lm()` function in R automatically calculates and stores the fitted values and the residuals, in the components fitted.values and residuals in the output of the `lm()` function.
-

```
> png("residuals.png")
>
> plot(x=linear.reg$fitted.values, y=linear.reg$residuals,
+       xlab="Fitted values", ylab="Residuals", col="red")
>
> dev.off()
```

- Check to see if the residuals are observed somewhat evenly on both sides of the reference zero line, and the spread of the residuals is fairly constant from one fitted value to the next.


Residuals
5
0
-5
Fitted values
8
10
12
14
16
18
20

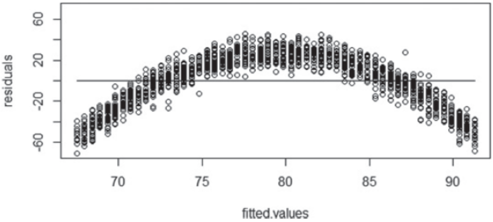


FIGURE 6-7 *Residuals with a nonlinear trend*

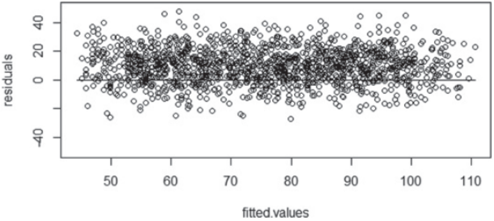


FIGURE 6-8 *Residuals not centered on the zero line*

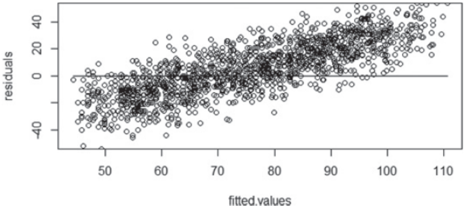


FIGURE 6-9 *Residuals with a linear trend*

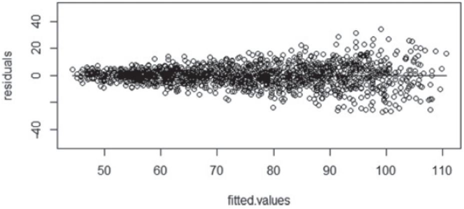


FIGURE 6-10 *Residuals with nonconstant variance*

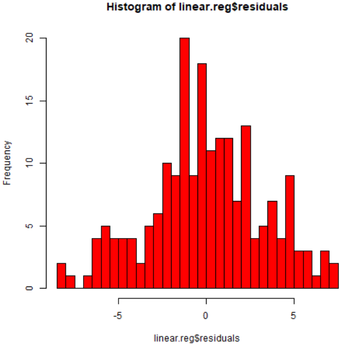
Checking the normality assumption

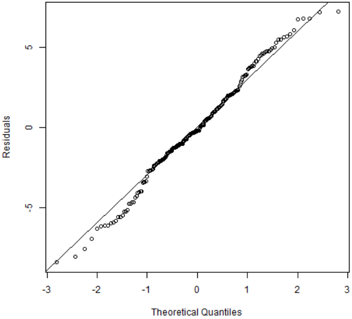
- We can produce a histogram of the residuals to further check the normality assumption for the error terms.

-
  ```
  > png("hist_residuals.png")
  >
  > hist(x=linear.reg$residuals, breaks=50,col="red")
  > dev.off()
  ```
- In addition, we can also produce a Quantile-Quantile (QQ) plot that compares the observed residuals against the quantiles of the theoretical normal distribution.

-
  ```
  > png("qq_residuals.png")
  >
  > qqnorm(linear.reg$residuals, ylab="Residuals", main="")
  > qqline(linear.reg$residuals)
  > dev.off()
  ```

Checking the normality assumption





References I

David Dietrich, Barry Heller, and Biebie Yang. Data science & big data analytics: discovering, analyzing, visualizing and presenting data. *EMC Education Services*, 2015.

Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.