Chapter 12: Interactive Notebook for Students

Ram Gopal, Dan Philps, and Tillman Weyde

Summer 20202

Contents

Load packages

```
library(bbmle)
```

Introduction

We explore linear regression models, one of the most widely used modeling techniques in statistics. We will use a dataframe `MASchools` which has information about a number of school districts in the state of Massachusetts.

```
df = read.csv("../../data/MASchools.csv")
```

Our main variable of interest is *score*4, which is the score of fourth grade students. Along with this, we will work with a few other variables from this data frame.

```
df = df[,c(13,7,8,9,11,15)]
df[1:7,]
```

```
##   score4 exptot scratio special stratio salary
## 1    714   4646    16.6    14.6    19.0  34.36
## 2    731   4930     5.7    17.4    22.6  38.06
## 3    704   4281     7.5    12.1    19.3  32.49
## 4    704   4826     8.6    21.1    17.9  33.11
## 5    701   4824     6.1    16.8    17.5  34.44
## 6    714   6454     7.7    17.2    15.7     NA
## 7    725   5537     5.4    11.3    17.1  41.62
```

A brief explanation of the variables is provided below.

Variable	Description
score4	4th grade score (math + English + science)
exptot	Expenditures per pupil, total
scratio	Students per computer
special	Special education students (per cent)
stratio	Student-teacher ratio
salary	Average teacher salary

Dealing with Missing Data

You will notice that there are missing values in the data frame which are marked as NA. We want to remove these observations from our analysis. There is a function called `complete.cases()` which allows you to find rows that have no missing values. First, let us create a table to determine the number of complete rows.

```
round((colMeans(is.na(df)))*100,3)
```

```
##  score4  exptot scratio special stratio  salary
##   0.000   0.000   4.091   0.000   0.000  11.364
```

```
salarymean = mean(df$salary,na.rm = T)
salaryimputed = ifelse(is.na(df$salary),salarymean,df$salary)
salarymean
```

```
## [1] 35.99
```

```
mean(salaryimputed)
```

```
## [1] 35.99
```

```
sd(df$salary,na.rm = T)
```

```
## [1] 3.191
```

```
sd(salaryimputed)
```

```
## [1] 3.003
```

```
table(complete.cases(df))
```

```
##
## FALSE  TRUE
##    34   186
```

There are 34 rows that contain missing values. We will create and work with a new data frame that contains only complete observations.

```
df1 = df[complete.cases(df),]
table(complete.cases(df1))
```

```
##
## TRUE
##  186
```

Now the data is ready for analysis!

Baseline Model

Our outcome of interest is the variable score4, which is numeric, and its data generation process is shown in the figure below.

Data Generation Process
$$score4 \sim N(\mu, \sigma)$$

score4	exptot	scratio	special	stratio	salary
714	4646	16.6	14.6	19	34.36
731	4930	5.7	17.4	22.6	38.063
704	4281	7.5	12.1	19.3	32.491
704	4826	8.6	21.1	17.9	33.106
701	4824	6.1	16.8	17.5	34.4365

We will estimate the population parameters of score4.

```
library(bbmle)
LLnorm = function(mean1, standdev){
  p = dnorm(x = df1$score4, mean = mean1, sd = standdev)
  LL = sum(log(p))
  return(-1*LL)
}


res1 = mle2(minuslogl = LLnorm, start = list(mean1 = 700, standdev = 50))
summary(res1)
```

```
## Maximum likelihood estimation
##
## Call:
## mle2(minuslogl = LLnorm, start = list(mean1 = 700, standdev = 50))
##
## Coefficients:
##            Estimate Std. Error z value              Pr(z)
## mean1      709.038      1.129    628.1 <0.0000000000000002 ***
## standdev    15.396      0.798     19.3 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## -2 log L: 1545
```

Let us unpack and understand this a bit. Our estimation results show that

$$score4 \sim N(709, 15.4)$$

we can write this as

$$score4 = 709 + N(0, 15.4)$$

$$score4 = 709 + \epsilon, \ \epsilon \sim N(0, 15.4)$$

4

The way to interpret the above is that our estimation or prediction for score4 is 709 for any school district. Of course there is possibility for errors in prediction, and this error is normally distributed with a standard deviation of 15.4. Why is there an error in our prediction? This is because there are a number of variables which we do not have information about that impact the score of fourth graders. These are our **omitted variables**. For example, these variables could be family income, extracurricular activities available in the district, parental involvement, quality of teachers, and so on. These represent **what we do not know**. The higher the standard deviation of the error term, the more we do not know. From this perspective, it is good to develop models that lower the variance or standard deviation in the error term.

Let us plot the errors in a graph below.

```
x = seq(1,nrow(df1))
plot(x = x, y = df1$score4)
abline(h = 709, col = "blue")
segments(x,709,x,df1$score4,col = "red")
```
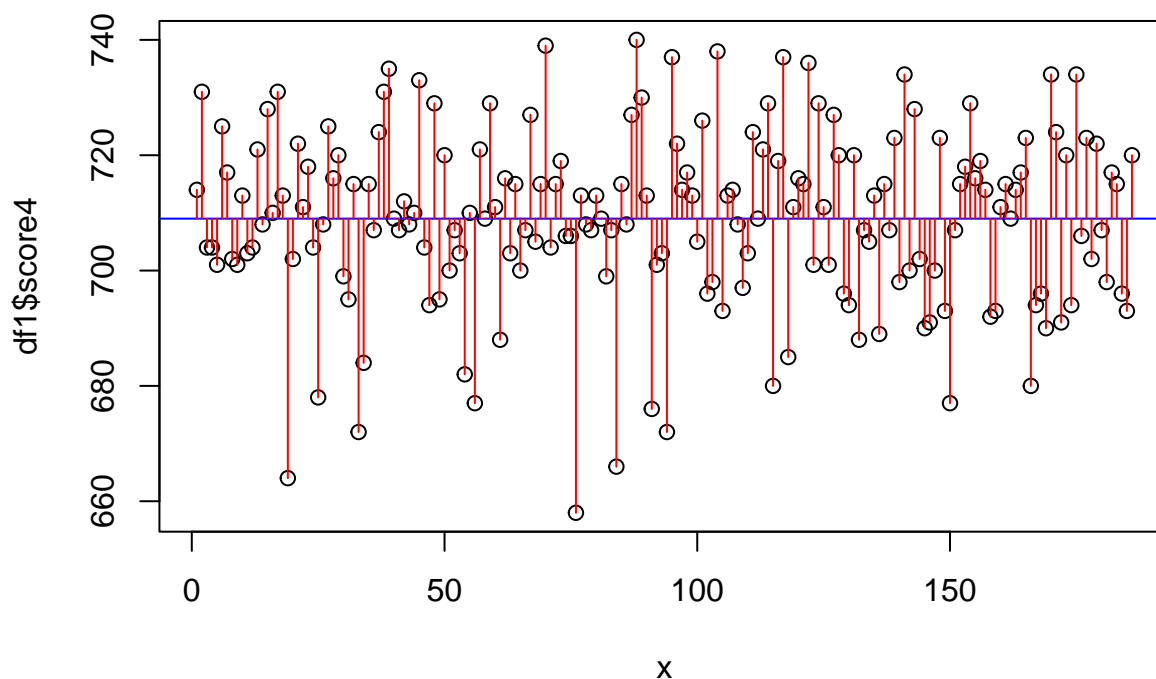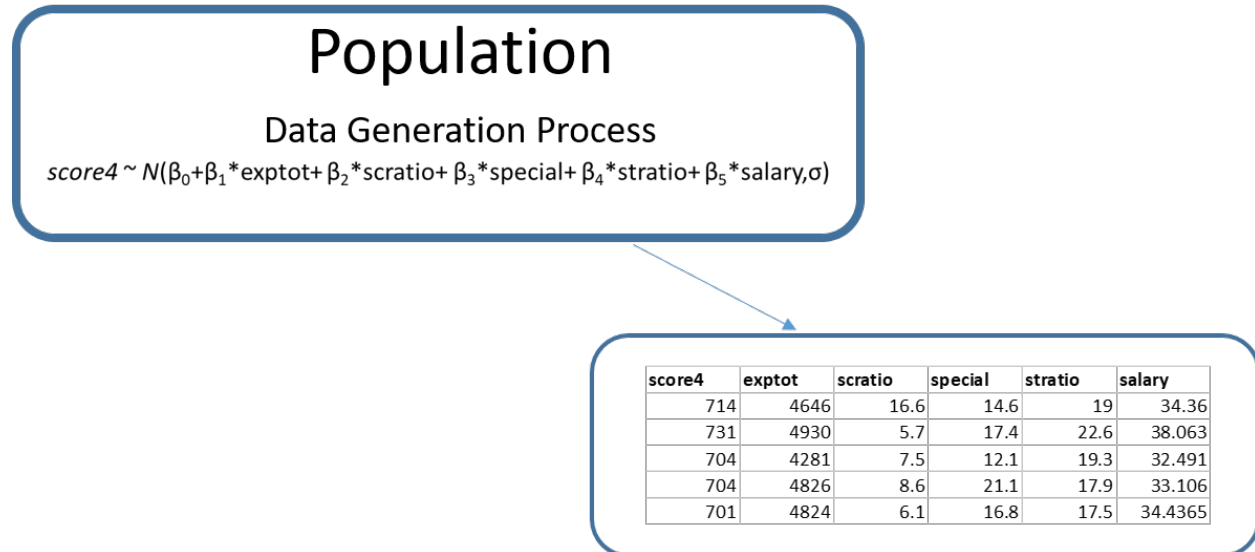

Let us compute the sum of squared errors for all the observations. This is called **SST (sum of squares total)**. We will come back to this later.

```
SST = sum((df1$score4 - 709)^2)
SST
```

```
## [1] 44087
```

Linear Regression Model

In the baseline model, our estimation or prediction of score4 for every school district is the same at 709. Now suppose we believe that we can explain why some school districts have higher scores and others have lower scores. The new, more intelligent, data generation process we propose is the following.

Population

Data Generation Process

$score4 \sim N(\beta_0 + \beta_1 * exptot + \beta_2 * scratio + \beta_3 * special + \beta_4 * stratio + \beta_5 * salary, \sigma)$

score4	exptot	scratio	special	stratio	salary
714	4646	16.6	14.6	19	34.36
731	4930	5.7	17.4	22.6	38.063
704	4281	7.5	12.1	19.3	32.491
704	4826	8.6	21.1	17.9	33.106
701	4824	6.1	16.8	17.5	34.4365

Let us write the code to estimate this data generation process. First, we will create the function to compute the loglikehood.

```
library(bbmle)
LLnorm = function(b0,b1,b2,b3,b4,b5, standdev){
  mean1 = b0 + b1*df1$exptot + b2*df1$scratio + b3*df1$special + b4*df1$stratio + b5*df1$salary
  p = dnorm(x = df1$score4, mean = mean1, sd = standdev)
  LL = sum(log(p))
  return(-1*LL)
}
```

Before we optimize, we need to set some starting values for all the parameters. A reasonable set of starting values can be obtained from the baseline model we did previously. With this, we write the code as:

```
res2 = mle2(minuslogl = LLnorm,
            start = list(b0 = 709,b1 = 0,b2 = 0,b3=0,b4=0,b5=0,standdev = 15.4))
summary(res2)
```

```
## Maximum likelihood estimation
##
## Call:
## mle2(minuslogl = LLnorm, start = list(b0 = 709, b1 = 0, b2 = 0,
##     b3 = 0, b4 = 0, b5 = 0, standdev = 15.4))
##
## Coefficients:
##          Estimate Std. Error   z value               Pr(z)
## b0      714.76110    0.00417 171455.36 < 0.0000000000000002 ***
## b1       -0.00503    0.00143     -3.53              0.00042 ***
## b2       -0.10839    0.35168     -0.31              0.75793
```

6

```
## b3           -0.71674    0.25416    -2.82                  0.00480 **
## b4           -2.64722    0.39698    -6.67     0.0000000000259 ***
## b5            2.20965    0.31554     7.00     0.0000000000025 ***
## standdev     13.10422    0.67941    19.29 < 0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## -2 log L: 1485
```

We just estimated our first regression!

Understanding Regression Results

Regression Equation

Our regression result is the following.

$$score4 = 714.8 - 0.005 \times exptot - 0.108 \times scratio - 0.717 \times special - 2.65 \times stratio + 2.21 \times salary$$

Let us write out the estimation or prediction for each observation, both from our baseline model and from the regression model. We will also write out the error from the regression equation.

```
coef(res2)
```

```
##        b0        b1        b2        b3        b4        b5  standdev
## 714.76110  -0.00503  -0.10839  -0.71674  -2.64722   2.20965  13.10422
```

```
df1$baseline = 709
df1$regression = as.matrix(df1[,2:6]) %*% coef(res2)[2:6] + coef(res2)[1]
df1$regerror = df1$score4 - df1$regression
round(df1[1:5,c(1,7,8,9)],3)
```

```
##   score4 baseline regression regerror
## 1    714      709      704.8    9.244
## 2    731      709      701.2   29.845
## 3    704      709      704.4   -0.446
## 4    704      709      700.2    3.800
## 5    701      709      707.6   -6.562
```

Let us plot the errors from the regression equation.

```
x = seq(1,nrow(df1))
plot(x = x, y = df1$regerror, col = "purple", pch = 19)
abline(h=0)
segments(x,df1$regerror,x,0,col = "red")
```

7
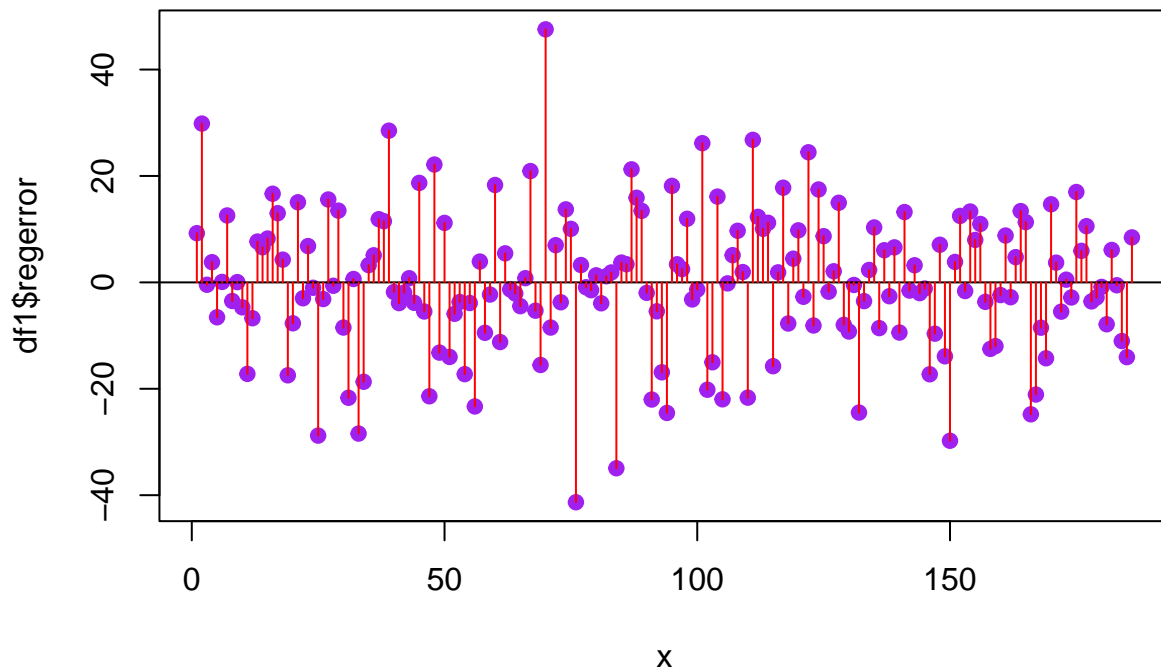
Model Fit

If the regression model is better than the baseline model, then we expect the errors to be smaller. Similar to before, let us compute the sum of the squared errors. This is called **SSE (sum of squares errors)**.

```
SSE = sum((df1$regression - df1$score4)^2)
SSE
```

```
## [1] 31941
```

Clearly, errors are lower than in the baseline model. Thus, the regression model is giving us a better approach to estimate score4. There is a measure called R^2 which is defined as

$$R^2 = 1 - \frac{SSE}{SST}$$

The value for us is

```
Rsquare = 1 - (SSE/SST)
round(Rsquare,3)
```

```
## [1] 0.276
```

R^2 is a measure of overall goodness of fit. When R^2 is 0, we know that SSE=SST, which means the regression model is the same as the unintelligent baseline model. When SSE = 0, which indicates that there are no errors in the regression, then we get an R^2 of 1.

Another way to understand R^2 is by looking at the standard deviation (or equivalently, variance) of the error terms in the baseline and regression models.

$$R^2 = 1 - \frac{\text{variance of error term in the regression model}}{\text{variance of error term in the baseline model}}$$

Let us check this with the code.

```
varbaseline = 15.39563^2
varregression = 13.1042240^2
round(1 - (varregression/varbaseline),3)
```

```
## [1] 0.276
```

R^2 is an indication of what proportion of the variation observed in the outcome is being explained by the model. Our regression model is able to explain about 27.5% of the variation in score4. The purpose of building a good regression model is to lower the standard deviation of the error term from the baseline model as much as possible.

Marginal Effects

Assuming all the assumptions of the regression model hold, the coefficients for each of the independent variables (these are the variables on the right-hand side of the regression equation) are called the **marginal effects**. For example, a unit increase in salary is expected to increase score4 by 2.2096528. You have to be careful with this statement. We will have more to say about this soon.

Statistical Significance

Let us take a look at the regression result for the variable scratio. The estimated value is -0.1083852 and the standard error is 0.3516757. As we discussed in the previous lesson, the point estimate based on the data is -0.1083852. The value of the coefficient in the population follows a normal distribution with a standard deviation of 0.3516757. Let us plot this parameter distribution and compute the 95% confidence interval.

```
m1 = -0.1083852
se1 = 0.3516757
curve(dnorm(x,mean = m1, sd = se1), from = -1, to = 1)
abline(v = m1)
```
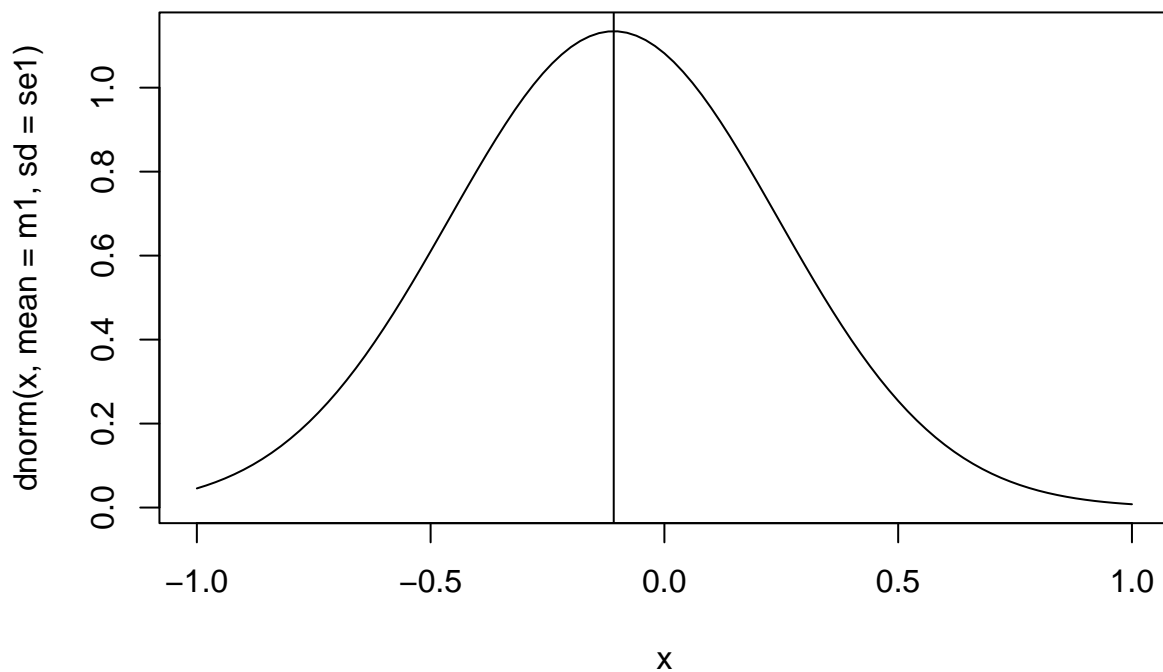
```
qvalues = qnorm(p = c((.05/2), 1-(.05/2)))
qvalues
```

```
## [1] -1.96  1.96
```

Since the 95% confidence interval [-1.959964,1.959964] contains 0, we cannot statistically rule out the possibility that the coefficient value is 0. This means that there is a possibility that scratio has no effect on the outcome. We term this **not statistically significant**.

We can also compute a p value here similar to what we did earlier with statistical testing.

```
gap = abs(m1 - 0)
curve(dnorm(x,mean = m1, sd = se1), from = -1, to = 1)
abline(v = m1)
abline(v = m1 - gap, col = "blue")
abline(v = m1 + gap, col = "blue")
```
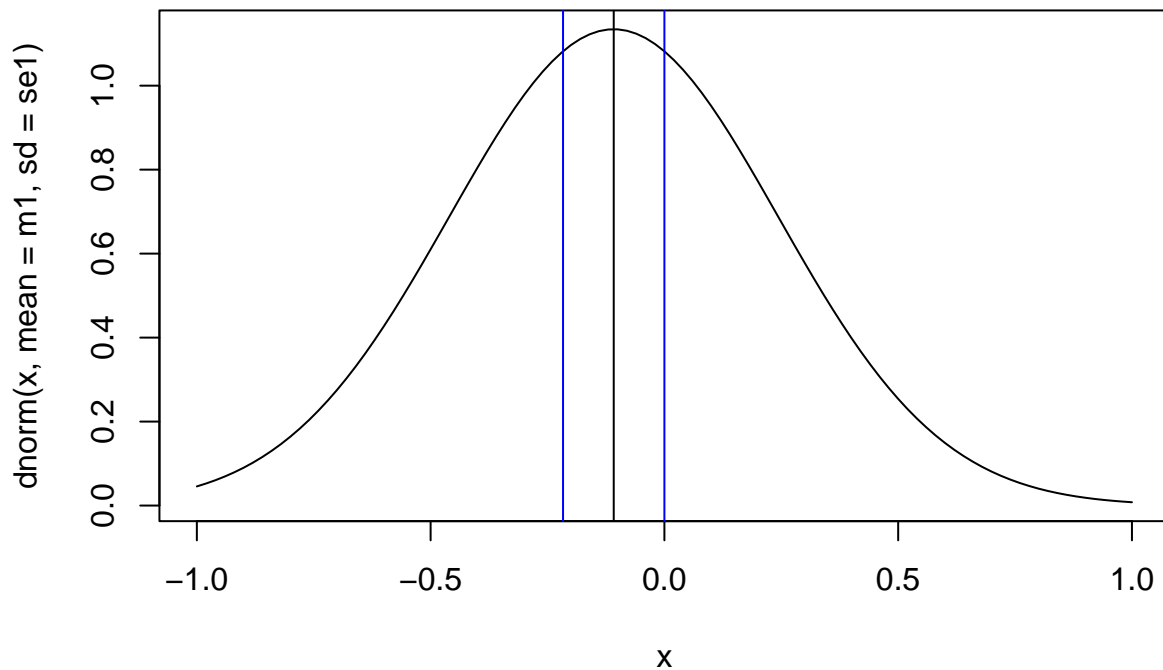
```
a = pnorm(q = m1 + gap, mean = m1, sd = se1)
b = pnorm(q = m1 - gap, mean = m1, sd = se1)
pval1 = 1 - (a - b)
pval1
```

```
## [1] 0.7579
```

The p value we just computed is exactly what the `mle2()` function produces. When the p value is small (typically less than 0.05), we say that the variable has a statistically significant impact on the outcome, and otherwise, not.

Based on the results we obtained, we can conclude that all the variables except scratio has an effect on score4. Furthermore, all statistically significant variables except salary have a negative effect on score4. Interestingly, higher expenditure per pupil is associated with lower grade four scores.

Assumptions of the Regression Model

As you can imagine, regression models are very widely used. Many important decisions and policies are implemented based on regression studies. These actions and policies entail expending significant time, energy, and resources based on the results of the regression analysis. Given the importance of this, before implementing or taking critical actions, it is important that you understand that the model you built satisfies all the key assumptions on which the model is built. Violations of these assumptions may provide misleading insights and put you down the wrong path in terms of actions taken. Let us take a look at some of the important assumptions.

11

Normality

A key assumption made is that error terms are normally distributed. Is this true in our case? We can check this easily.

```
plot(density(df1$regerror))
```

density.default(x = df1$regerror)
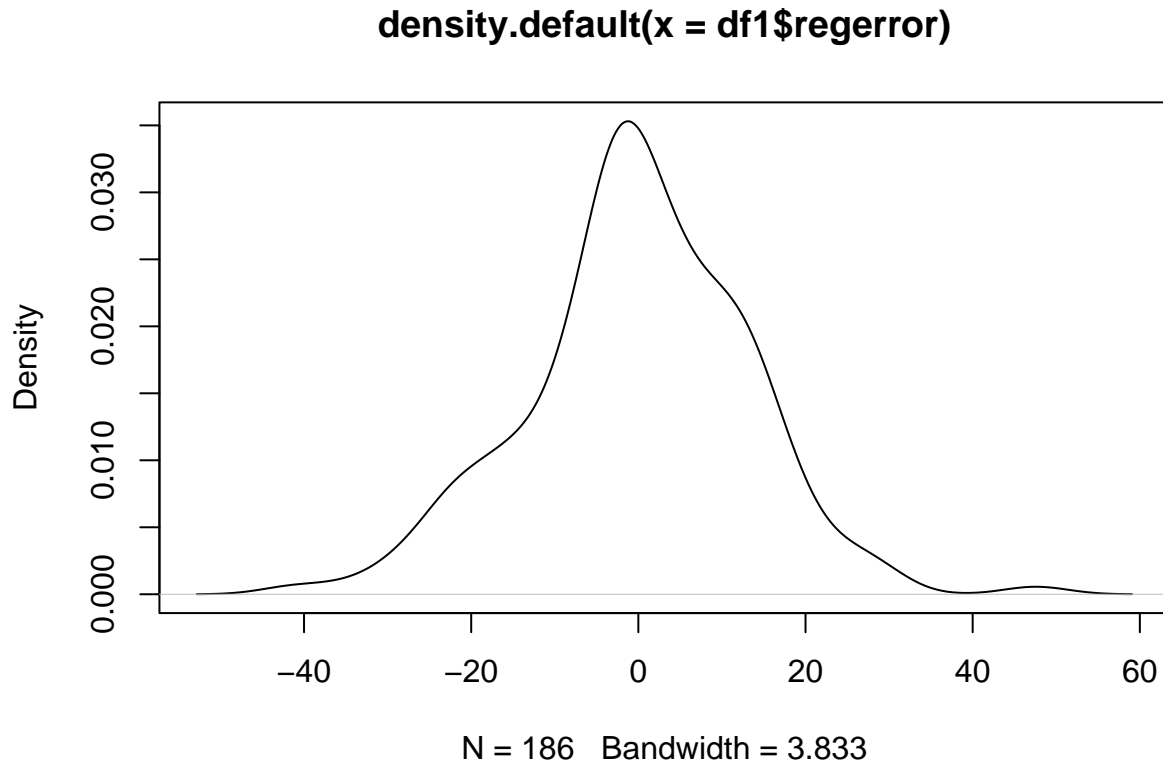


```
shapiro.test(x = df1$regerror)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df1$regerror
## W = 0.99, p-value = 0.1
```

The plot and the Shapiro-Wilk normality test confirm that the errors are reasonably normally distributed. If you find significant deviations from normality, a good approach to take is to reexamine the regression structure. This, for example, could be to transform the independent or dependent variables.

Homoscedasticity

The regression model assumes a constant standard deviation. This assumption means that the variance or standard deviation of the error term is not correlated to the dependent or outcome variable. In other words, we should not see the magnitude of errors varying with the values of the outcome. Let us take a look at this through a plot of predicted outcomes and the errors.

```
plot(df1$regression, df1$regerror)
abline(h = 0)
segments(df1$regression, 0, df1$regression, df1$regerror, col = "red")
```
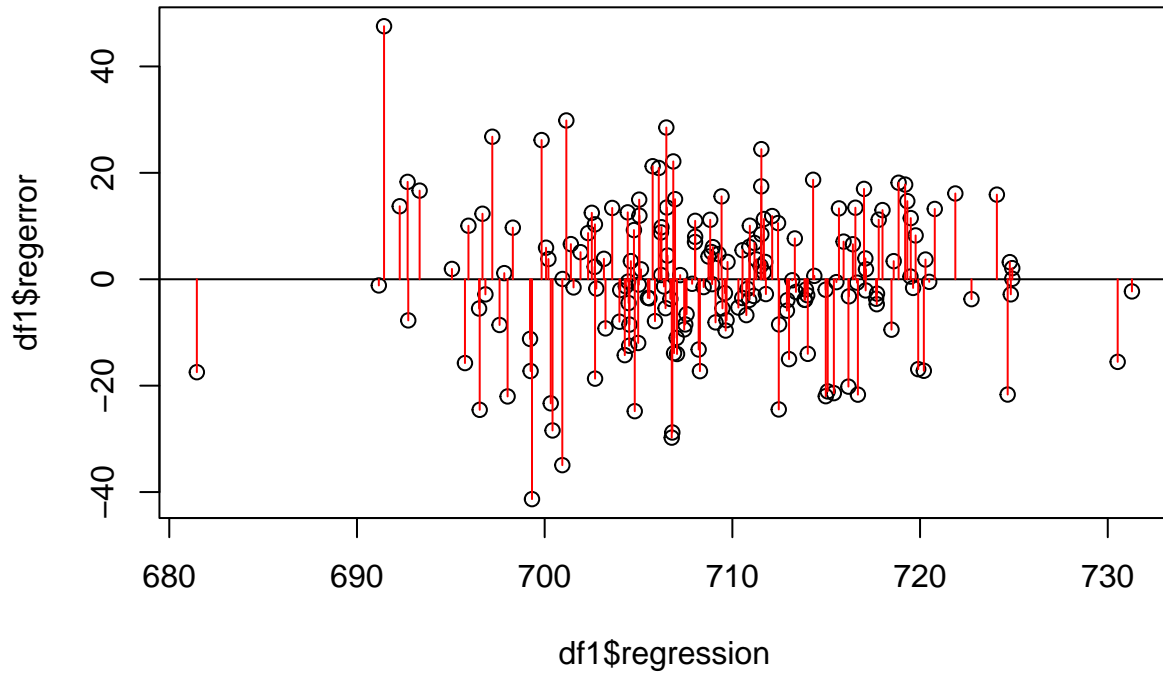

Visually, there does not appear to be a violation of this assumption in our model. There are formal tests that you can conduct for check for **heteroscedasticity**.

Multicollinearity

Suppose we are interested in the effect of salary on score4. To place our focus on the effect of salary, let's express the regression equation as follows.

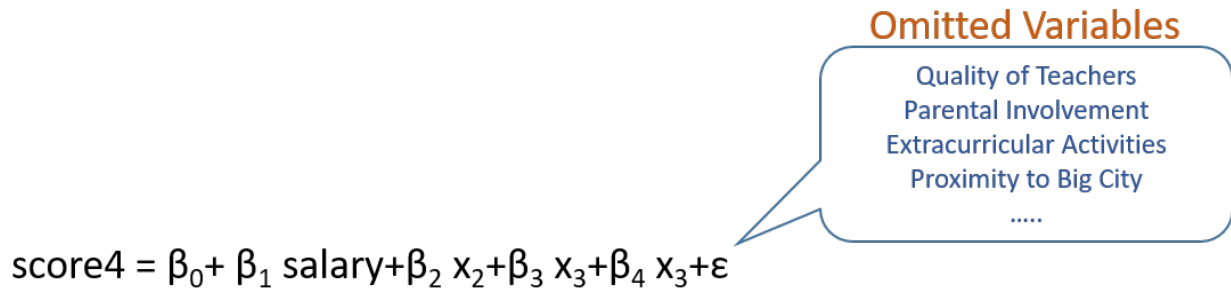$$score4 = \beta_0 + \beta_1 salary + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \epsilon$$

Take the first derivative with respect to salary

$$\frac{d(score4)}{d(salary)} = \beta_1 + \beta_2 \frac{dx_2}{d(salary)} + \beta_3 \frac{dx_3}{d(salary)} + \beta_4 \frac{dx_4}{d(salary)} + \frac{d\epsilon}{d(salary)}$$

For now let us assume that $\frac{d\epsilon}{d(salary)} = 0$. The problem of **multicollinearity** exists when the independent variable salary is correlated to the other independent variables x_2, x_3 and x_4. The marginal effect of salary on the outcome is β_1 only if the salary is uncorrelated to other independent variables. There are formal tests available to check for multicollinearity. The approaches to deal with multicollinearity if it exists include dropping, combining and transforming variables. Be cautious of multicollinearity before intrepreting β_1 as the marginal effect of salary on score4.
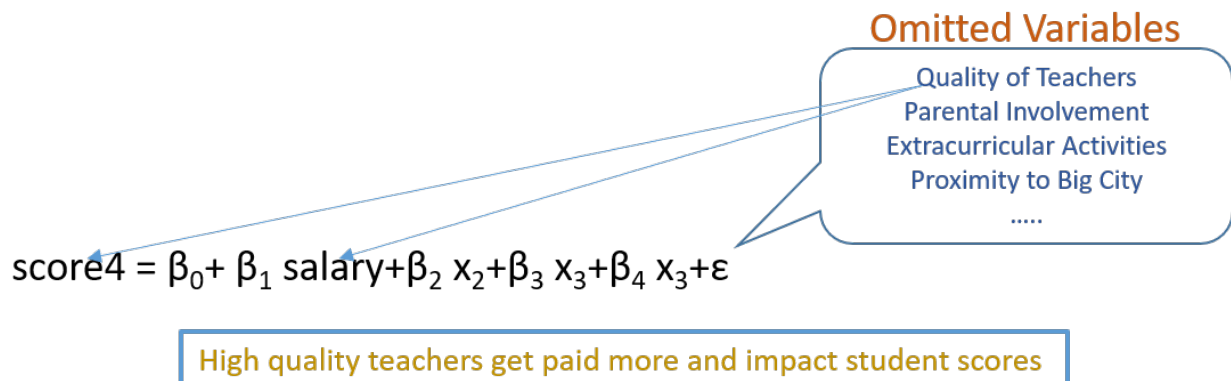
13

Endogeneity

A much bigger problem is with the assumption $\frac{d\epsilon}{d(salary)} = 0$. What does this really mean? As we discussed before, the error term encompasses all the ommitted variables that impact our outcome. Some possible ommitted variables are illustrated in the figure below.

Omitted Variables

Quality of Teachers
Parental Involvement
Extracurricular Activities
Proximity to Big City
.....

$$score4 = \beta_0 + \beta_1 \, salary + \beta_2 \, x_2 + \beta_3 \, x_3 + \beta_4 \, x_3 + \epsilon$$

The assumption $\frac{d\epsilon}{d(salary)} = 0$ implies that none of the ommitted variables are correlated with salary. If this assumption is violated, then our coefficient β_1 cannot be correctly interpreted as the marginal effect of salary on score4. This estimate then become unreliable and biased. This is termed endogeneity.
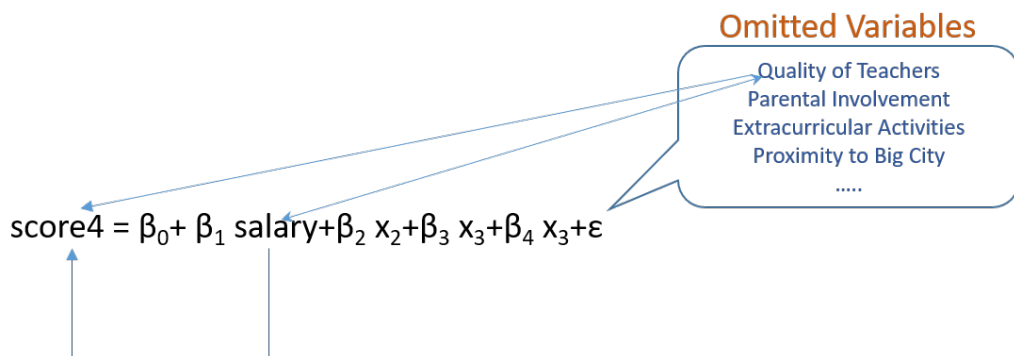
In our example, it is possible that one of the ommitted variables, let us say "quality of teachers", is related to salary. It is possible that high quality teachers command a higher salary and high quality teachers are the main reason for high values of score4. If this is the case, the effect of salary we observe from the regression result is spurious and misleading. One can argue that salary has no **causal effect** on score4. It is just that high quality teachers have high salaries and high quality teachers are the main cause of high student scores. In other words, there is correlation between salary and score4 but there is no causation. This is an example of the famous phrase in statistics: **correlation does not imply causation**. If this arguement is true, then one should not invest in paying higher salaries to teachers with the hope that this will lead to higher student scores. This is illustrated in the figure below.

Omitted Variables

Quality of Teachers
Parental Involvement
Extracurricular Activities
Proximity to Big City
.....

$$score4 = \beta_0 + \beta_1 \, salary + \beta_2 \, x_2 + \beta_3 \, x_3 + \beta_4 \, x_3 + \epsilon$$

High quality teachers get paid more and impact student scores

$$\text{score4} = \beta_0 + \beta_1 \text{ salary} + \beta_2\, x_2 + \beta_3\, x_3 + \beta$$

High quality teachers get paid more and impact s
more time and energy on students which improv

There are, of course, other possible causal links as shown in the figures below.

Omitted Variables

Quality of Teachers
Parental Involvement
Extracurricular Activities
Proximity to Big City
.....

$$\text{score4} = \beta_0 + \beta_1 \text{ salary} + \beta_2\, x_2 + \beta_3\, x_3 + \beta_4\, x_3 + \varepsilon$$

High quality teachers get paid more and impact student scores. Higher pay also induces teachers to spend more time and energy on students which improves student scores. Higher pay also attracts high quality teachers.

Intrepreting the coefficient of salary, 2.2096528, from the regression result as the marginal causal effect on score4 means the following: if the salary is increased by one unit then the score4 will increase by 2.21 units. This is reasonable only if you make the strong assumption that there are no ommitted variables that are correlated with salary. There are many stories where important policy decisions (such as increasing the salaries of teachers with the expectation that scores will go up) have been made assuming causality which subsequently backfired (link).

One major practical problem is that there is no conclusive way to check for this assumption. We have ommitted variables because we do not have information on them. Often, we do not know what and how many ommitted variables there are. In this case, how can we really check for the validity of the assumption? After all, we don't know what we don't know!

At the core, this is a **data** problem. This problem arises because of how we collect data and what data we collect. If the data is collected through a **randomized, controlled experiment**, then we can easily rule out problem with ommitted variables. In our case, it could work as follows. We randomly identify some school districts as the control group and the rest as the treatment group. We then randomly assign teachers to school districts. Teachers assigned to the treatment group are paid a higher salary than the teachers in the control group. We then observe and collect data over a period of time. If we were to do this, then we will not face the problem of bias induced by the ommitted variables. Why? Because of random assignment of teachers, salary is not longer correlated to any ommitted variables. Quite obviously, this is simply not feasible in our context. But this is a standard approach in clinical trials to study the effect of medical treatments and drugs.

When we are restricted to **observational data**, it becomes impossible to prove or guarantee causality(link).

However, we can take steps to alleviate the concerns with causality. For example, with panel data (i.e. data collected on individuals or entities at multiple points in time) we can address some of the concerns that ommitted variables might cause. Another approach is through **instrumental variables** which are variables related to salary but uncorrelated with the error term. These are hard to find in many situations. Please read one of my earlier research papers on the use of instrumental variables to examine the link between music piracy and sales (link). The ommitted variable that we had to deal with was 'popularity' which impacts both piracy and sales. You are strongly recommended to google these terms to learn more.

iid

The observations should be independent and identically distributed. This is called the **iid** assumption. We need this assumption because in computing the likelihood, we multiply the probabilities of the observations. We can do so only if the observations are independent. This assumption is true for all maximum likelihood estimations. Examples where this is not true are time series data and network data.

lm() Function

The function to estimate a regression model in R is `lm()`. Let us estimate our regression using this function.

```
res3 = lm(score4~exptot+scratio+special+stratio+salary,data=df1)
summary(res3)
```

```
##
## Call:
## lm(formula = score4 ~ exptot + scratio + special + stratio +
##     salary, data = df1)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -41.32  -7.46  -0.52   8.78  47.57
##
## Coefficients:
##             Estimate Std. Error t value             Pr(>|t|)
## (Intercept) 714.70352   16.34343   43.73 < 0.0000000000000002 ***
## exptot       -0.00503    0.00150   -3.35              0.00097 ***
## scratio      -0.10788    0.36733   -0.29              0.76935
## special      -0.71628    0.28046   -2.55              0.01148 *
## stratio      -2.64688    0.49160   -5.38           0.000000225 ***
## salary        2.21067    0.37381    5.91           0.000000016 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.3 on 180 degrees of freedom
## Multiple R-squared:  0.276,  Adjusted R-squared:  0.255
## F-statistic: 13.7 on 5 and 180 DF,  p-value: 0.000000000025
```