CWRU DSCI351-451: Week12a-p Multiple Regression

Roger H. French, JiQi Liu

14 November, 2018

Contents

12.1.2.1 Reading, Homeworks, Projects, SemProjects

- Homework:
  - HW6 Due Thursday, November 14th
- Readings:
  - ISLR4 today
  - ISLR6 Thursday
- Projects: We will have four 2 week EDA projects
  - You have Proj 3
- 451 SemProjects:
  -
  -
- Final Exam
  - Tuesday December 18th, 12 noon to 3pm, Olin 303

12.1.2.2 Textbooks

- Peng: R Programming for Data Science
- Peng: Exploratory Data Analysis with R
- Open Intro Stats, v3
- Wickham: R for Data Science
- Hastie: Intro to Statistical Learning with R

12.1.2.3 Syllabus

12.1.2.4 Multiple Regression Practicum

Day:Date	Foundation	Practicum	Reading	Due
w1a:Tu:8/28/18	ODS Tool Chain	R, Rstudio, Git		
w1b:Th:8/30/18	Setup ODS Tool Chain	Bash, Git, Twitter	PRP4-33	HW1
w2a:Tu:9/4/18	What is Data Science	OIS:Intro2R	PRP35-64	**HW1 Due**
w2b:Th:9/6/18	Data Analytic Style, Git	451SempProj, Git	PRP65-93, OI1-1.9	HW2
w3a:Tu:9/11/18*	Struct. of Data Analysis	ISLR:Intro2R, Loops	PRP94-116, OIS3	**HW2 Due**
w3b:Th:9/13/18*	OIS3 Intro to Data	GapMinder, Dplyr, Magrittr		
w4a:Tu:9/18/18	OIS3, Intro2Data part 2, Data	EDA: PET Degr.	EDA1-31	Proj1
w4b:Th:9/20/18	Hypothesis Testing	GGPlot2 Tutorial	EDA32-58	HW3
w5a:Tu:9/25/18	Distributions	SemProj RepOut1	R4DS1-3	**HW3 Due**
w5b:Th:9/27/18	Wickham DSCI in Tidyverse	SemProj RepOut1	R4DS4-6	**SemProj1,**
w6a:Tu:10/2/18	OIS Found. of Inference	Inference	R4DS7-8	**Proj1 Due**
w6b:Th:10/4/18		Midterm Review	R4DS9-16 Wrangle	
w7a:Tu:10/9/18*	Summ. Stats & Vis.	Data Wrangling		
w7b:Th:10/11/18*	**MIDTERM EXAM**			HW4
w8a:Tu:10/16/18	Numerical Inference	Tidy Check Explore	OIS4	**HW4 Due**
w8b:Th:10/18/18	Algorithms, Models	Pairwise Corr. Plots	OIS5.1-4	Proj 2, HW5
Tu:10/23	**CWRU FALL BREAK**		R4DS17-21 Program	
w9b:Th:10/25/18	Categorical Infer	Predictive Analytics	OIS6.1,2	
w10a:Tu:10/30/18	SemProj	SemProj	OIS7	**SemProj2 HW5 Du**
w10b:Th:11/1/18	Lin. Regr.	Lin. Regr.	OIS8	**Proj.2 due**
w11a:Tu:11/6/18	Inf. for Regression	Curse of Dim.	OIS8	Proj 3
w11b:Th:11/8/18	Model Accuracy	Training Testing	ISLR3	HW6
w12a:Tu:11/13/18	Multiple Regr.	Mul. Regr. & Pred.	ISLR4	**HW6 due**
w12b:Th:11/15/18	Classification		ISLR6	
w13a:Tu:11/20/18	Classification	Clustering	ISLR5	**Proj 3 due**
Th:11/22/18	**THANKSGIVING**			Proj 4
w14a:Tu:11/27/18	Big Data	Hadoop		
w14b:Th:11/29/18	InfoSec	VerisDB		**SemProj3**
w15a:Tu:12/4/18	SemProj ReportOut3			
w15b:Th:12/6/18	SemProj ReportOut3			**Proj4**
	FINAL EXAM	**Monday12/17, 12:00-3:00pm**	Olin 313	**SemProj4 due**

Figure 1: DSCI351-451 Syllabus

12.1.2.4.1 First steps in the data analysis

```r
library(psych)
if (!require("MASS")) install.packages("MASS")
```

```
## Loading required package: MASS
```

```r
library(MASS)
?MASS
```

```
## No documentation for 'MASS' in specified packages and libraries:
## you could try '??MASS'
```

```r
packageDescription('MASS')
```

```
## Package: MASS
## Priority: recommended
## Version: 7.3-51.1
## Date: 2018-10-31
## Revision: $Rev: 3492 $
## Depends: R (>= 3.1.0), grDevices, graphics, stats, utils
## Imports: methods
## Suggests: lattice, nlme, nnet, survival
## Authors@R: c(person("Brian", "Ripley", role = c("aut", "cre",
##        "cph"), email = "ripley@stats.ox.ac.uk"), person("Bill",
##        "Venables", role = "ctb"), person(c("Douglas", "M."),
##        "Bates", role = "ctb"), person("Kurt", "Hornik", role =
##        "trl", comment = "partial port ca 1998"),
##        person("Albrecht", "Gebhardt", role = "trl", comment =
##        "partial port ca 1998"), person("David", "Firth", role =
##        "ctb"))
## Description: Functions and datasets to support Venables and
##        Ripley, "Modern Applied Statistics with S" (4th edition,
##        2002).
## Title: Support Functions and Datasets for Venables and Ripley's
##        MASS
## LazyData: yes
## ByteCompile: yes
## License: GPL-2 | GPL-3
## URL: http://www.stats.ox.ac.uk/pub/MASS4/
## Contact: <MASS@stats.ox.ac.uk>
## NeedsCompilation: yes
## Packaged: 2018-10-31 08:34:59 UTC; ripley
## Author: Brian Ripley [aut, cre, cph], Bill Venables [ctb], Douglas
##        M. Bates [ctb], Kurt Hornik [trl] (partial port ca 1998),
##        Albrecht Gebhardt [trl] (partial port ca 1998), David Firth
##        [ctb]
## Maintainer: Brian Ripley <ripley@stats.ox.ac.uk>
## Repository: CRAN
## Date/Publication: 2018-11-01 11:22:24 UTC
## Built: R 3.5.1; x86_64-pc-linux-gnu; 2018-11-05 13:38:37 UTC; unix
##
## -- File: /home/frenchrh/R/x86_64-pc-linux-gnu-library/3.5/MASS/Meta/package.rds
```

In what follows, we will use a dataset of 40 cases

- generated from a covariance matrix

    - obtained from a subsample of real data we collected,
- which is about
    - burnout components,
    - work satisfaction,
    - work-family conflict, and
    - organizational commitment
- in hospitals.

There are six attributes in the dataset that we will analyze here;

- all are self-assessments made by nurses:
    - Commit: Commitment to their hospital (response here)
    - Exhaust: Emotional exhaustion (one of the three components of burnout)
    - Depers: Depersonalization (one of the three components of burnout)
    - Accompl: Accomplishment (one of the three components of burnout)
    - WorkSat: Work satisfaction
    - WFC: Work-family conflict

Our goal here is to understand

- how burnout dimensions and work satisfaction
- affect commitment of nurses to their hospital.

We start by

- generating the data and
- examining the correlation table
- and significance.

Make sure the matcov.txt file is in your working directory before running this code:

```
matcov <- unlist(read.csv("./data/matcov.txt", header = F))
covs <- matrix(matcov, 6, 6)
means <- c(4.47,14.95,4.87,36.08,5,1.88)
set.seed(987)
nurses <- data.frame(mvrnorm(n = 40, means, covs))
colnames(nurses) <- c("Commit","Exhaus","Depers","Accompl",
                      "WorkSat","WFC")
corr.test(nurses)
```

```
## Call:corr.test(x = nurses)
## Correlation matrix
##         Commit Exhaus Depers Accompl WorkSat   WFC
## Commit    1.00  -0.64  -0.27    0.27    0.76 -0.52
## Exhaus   -0.64   1.00   0.20    0.12   -0.50  0.68
## Depers   -0.27   0.20   1.00    0.04   -0.51 -0.02
## Accompl   0.27   0.12   0.04    1.00    0.23  0.15
## WorkSat   0.76  -0.50  -0.51    0.23    1.00 -0.39
## WFC      -0.52   0.68  -0.02    0.15   -0.39  1.00
## Sample Size
## [1] 40
## Probability values (Entries above the diagonal are adjusted for multiple tests.)
##         Commit Exhaus Depers Accompl WorkSat  WFC
## Commit    0.00   0.00   0.73    0.73    0.00 0.01
## Exhaus    0.00   0.00   1.00    1.00    0.01 0.00
## Depers    0.10   0.23   0.00    1.00    0.01 1.00
## Accompl   0.09   0.45   0.79    0.00    0.96 1.00
## WorkSat   0.00   0.00   0.00    0.16    0.00 0.13
```

```
## WFC       0.00   0.00   0.92    0.35    0.01 0.00
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

The values with a probability value lower than 0.05

- are significant by common standards.

We can see, for instance, that, in this subsample,

- commitment is significantly correlated
  - with exhaustion, work satisfaction, and work-family conflict,
- but not with depersonalization and accomplishment.

We can also see that the predictors are intercorrelated—

- that is, they share part of their variance.

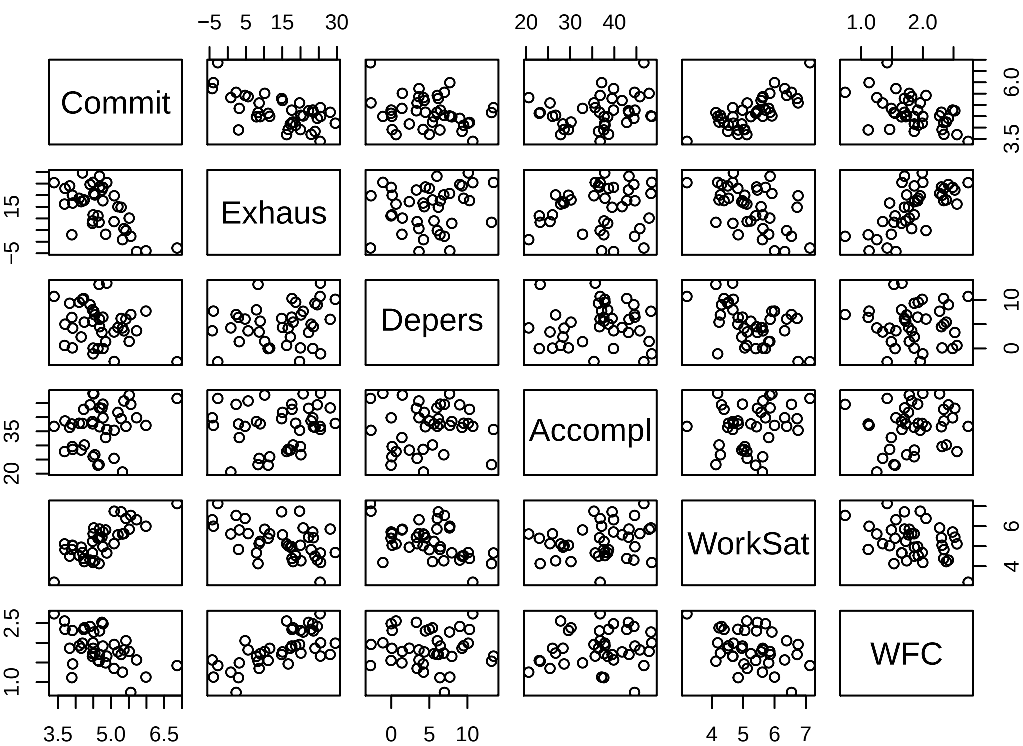
We will examine whether this constitutes a problem

- for a regression analysis later.

Let's plot the relationship

- to see if the relationships indeed seem linear:

```
plot(nurses)
```



Here, we will only comment on the scatterplots in which commitment is included.

We can see that there is visibly

- a negative linear association
  - between commitment and exhaustion and work-family conflict.
- There is visibly a positive linear relationship
  - between commitment and work satisfaction.
- Notice that there are also other relations visible on the plots,

    - such as the visible relation between work- family conflict and exhaustion.
- From these scatterplots,
    - nothing in the data seems problematic for the relationships we are exploring.

12.1.2.5 Performing the multiple linear regression

We want to know if there is a relationship

- between our predictors and the response.

We first want to know

- whether the three burnout dimensions
- predict commitment to the hospital.

We create the model by

- using the formula syntax
    - as an argument in the lm() function.
- What is on the left of the tilde (~) sign
    - is the response,
- on the right are the predictors,
    - separated by a plus (+) sign:

Let's examine

- the coefficients
- and their significance
- in the summary of the model:

```
model1 <- lm(Commit ~ Exhaus + Depers + Accompl, data = nurses)
```

The following output shows

- that exhaustion and accomplishment
    - are predictors of commitment to the hospital
    - (look at p-value under Pr(<|t|) or refer to *)
- exhaustion negatively
    - (more emotionally exhausted people are less committed)
- and accomplishment positively
    - (more accomplished people are more committed):

```
summary(model1)
```

```
##
## Call:
## lm(formula = Commit ~ Exhaus + Depers + Accompl, data = nurses)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.35915 -0.32590  0.02808  0.35635  0.97905
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.331261   0.398985  10.856 6.62e-13 ***
## Exhaus      -0.048725   0.008625  -5.649 2.05e-06 ***
## Depers      -0.027053   0.019795  -1.367  0.18021
## Accompl      0.032923   0.010392   3.168  0.00313 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4892 on 36 degrees of freedom
## Multiple R-squared:   0.55,  Adjusted R-squared:  0.5125
## F-statistic: 14.67 on 3 and 36 DF,  p-value: 2.116e-06
```

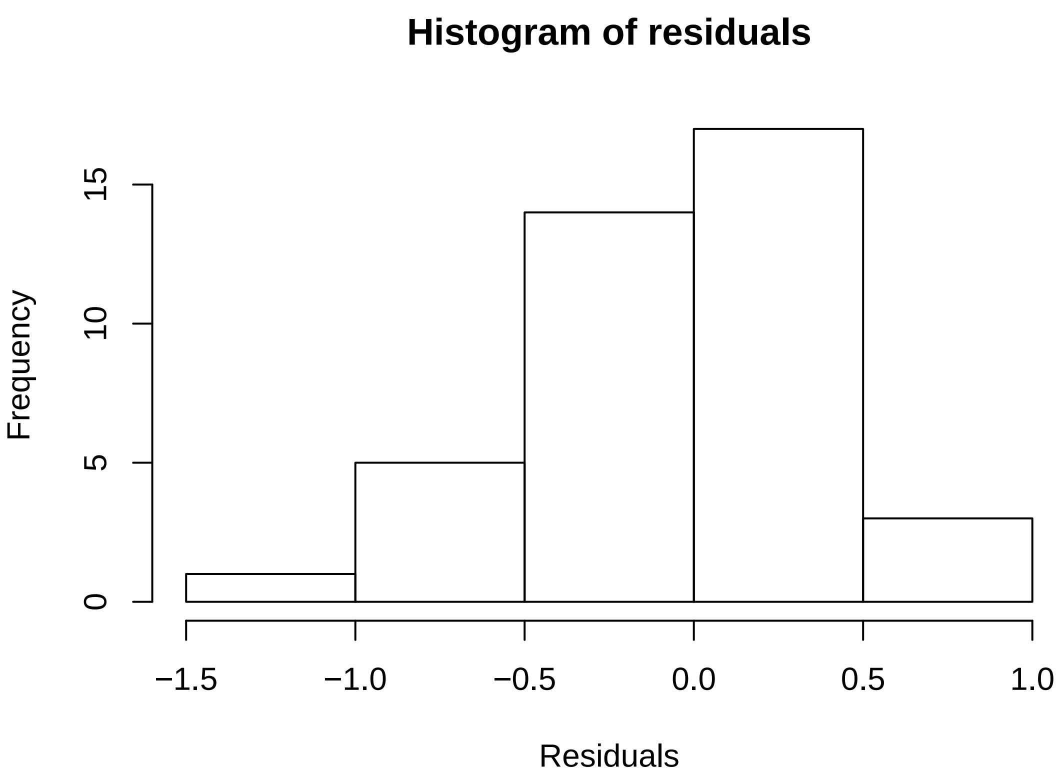
12.1.2.6 Checking for the normality of residuals

We have seen that it is important

- that residuals are normally distributed.

We can do this visually by plotting, as in the following line of code:

```
hist(resid(model1), main = "Histogram of residuals",
     xlab = "Residuals")
```



From the preceding output, we might suspect

- a slight deviation from normality.

The Shapiro-Wilk test

- is a test of normality in frequentist statistics

```
shapiro.test(resid(model1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(model1)
## W = 0.97757, p-value = 0.6001
```

We can also see that

- the p-value for F-statistic is significant (bottom of the output), and
- that 55 percent of variance (see Multiple R-squared) is predicted.

The adjusted R-squared

- considers the number of predictors
- in the calculation of its value.

It is recommended that you specify

- which value you use when reporting the results,
- or you can also report both values.

Here, we can see that

- Adjusted R-squared is just a bit lower than Multiple R-squared,
- meaning that the results are not much affected
  - by the number of predictors.

12.1.2.7 Checking for variance inflation

We also want to check whether

- there is a problem of variance inflation
- in our analysis
  - that is, whether the predictors are correlated a lot (multicollinear).
- For this purpose, we will rely on the vif() function of the HH package.
  - he function takes the lm formula as an argument:

```
# if (!require("HH")) install.packages("HH")
# install.packages("gmp")
# install.packages("Rmpfr")
# install.packages("HH")
# library(HH)
# vif(Commit ~ Exhaus + Depers + Accompl, data = nurses)
```

There are several rules-of-thumb to assess this.

- One is to consider vif values higher than 10 to be problematic,
- another is to consider a predictor as problematic
  - if the square root of the vif value is higher than 2.
- This is not the case here,
  - therefore, we consider our data to be non-multicollinear here.

12.1.2.8 Examining potential mediations and comparing models

Let's now examine whether

- including work-family conflict and work satisfaction
- permits to predict an additional part of variance.

We first will ask R to fit a second model, and

- then will compare model1 and model2 using the anova() function:

```
model2 <- lm(Commit ~ Exhaus + Depers + Accompl + WorkSat,
             data = nurses)
```

The following output shows that

- indeed the second model predicts additional variance

    - in comparison to model1
- (see the significance of the F statistic for the comparison (under Pr(>F)):

```
anova(model1,model2)
```

```
## Analysis of Variance Table
##
## Model 1: Commit ~ Exhaus + Depers + Accompl
## Model 2: Commit ~ Exhaus + Depers + Accompl + WorkSat
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     36 8.6161
## 2     35 5.7181  1     2.898 17.738 0.0001685 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We will now examine the second model,

- as the additional variance predicted is significantly different from 0:

```
summary(model2)
```

```
##
## Call:
## lm(formula = Commit ~ Exhaus + Depers + Accompl + WorkSat, data = nurses)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.98119 -0.22736 -0.01279  0.26613  0.73625
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.969672   0.650440   3.028 0.004598 **
## Exhaus      -0.029524   0.008460  -3.490 0.001326 **
## Depers       0.014686   0.019123   0.768 0.447656
## Accompl      0.017392   0.009345   1.861 0.071142 .
## WorkSat      0.463720   0.110103   4.212 0.000168 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4042 on 35 degrees of freedom
## Multiple R-squared:  0.7014, Adjusted R-squared:  0.6673
## F-statistic: 20.55 on 4 and 35 DF,  p-value: 8.662e-09
```

This model predicts 70 percent of variance in commitment,

- which is pretty good.

We can see that work satisfaction

- is a significant predictor of commitment to the hospital,
- that the unique contribution of accomplishment is no longer significant
    - (there is therefore a potential mediation),
- and that the contribution of exhaustion
    - has been reduced when including work satisfaction in the model
    - (there is therefore a potential partial mediation).
- This might be because of a mediation of the relationship
    - between the two burnout components
    - and commitment by job satisfaction.

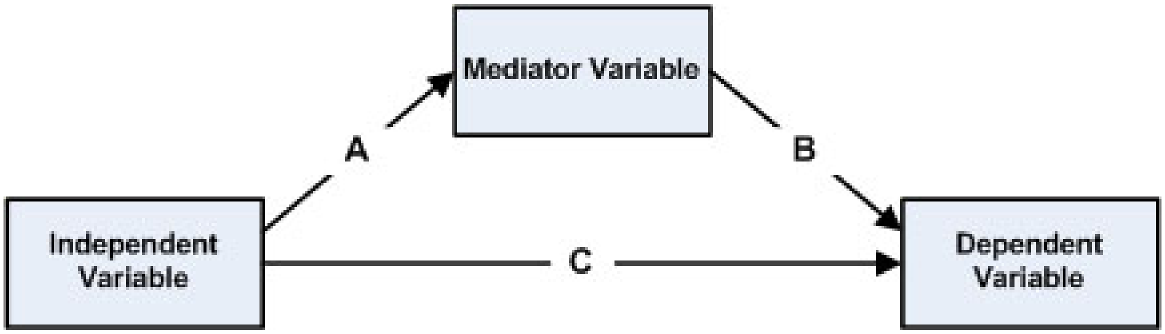


Figure 2: Mediation Model

What is Mediation?

- In statistics, a mediation model is
  - one that seeks to identify and explain the mechanism or process
  - that underlies an observed relationship between
  - an independent variable and a dependent variable
  - via the inclusion of a third hypothetical variable,
  - known as a mediator variable (also a mediating variable, intermediary

Let's test this relationship:

```
model3 <- lm(WorkSat ~ Exhaus + Depers + Accompl, data = nurses)
summary(model3)
```

```
##
## Call:
## lm(formula = WorkSat ~ Exhaus + Depers + Accompl, data = nurses)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.57359 -0.26967 -0.06299  0.24855  1.47504
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.09270    0.49899  10.206 3.59e-12 ***
## Exhaus      -0.04141    0.01079  -3.839 0.000482 ***
## Depers      -0.09001    0.02476  -3.636 0.000860 ***
## Accompl      0.03349    0.01300   2.577 0.014217 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6118 on 36 degrees of freedom
## Multiple R-squared:  0.5162, Adjusted R-squared:  0.4758
## F-statistic:  12.8 on 3 and 36 DF,  p-value: 7.59e-06
```

We can notice that

- 51 percent of the variance of job satisfaction
- is predicted by the burnout components.

All three burnout components

- are significantly related to work satisfaction ($p < .05$),
- negatively for emotional exhaustion and depersonalization

- and positively for personal accomplishment.

In order to ascertain mediation,

- we need to proceed to Sobel tests.
- The bda package contains the necessary function,
  - called mediation.test().

The Sobel Test

- is basically a specialized t test
  - that provides a method to determine whether the reduction
  - in the effect of the independent variable,
  - after including the mediator in the model,
  - is a significant reduction and
- therefore whether the mediation effect is statistically significant.

Let's try to see whether the effect of exhaustion on commitment

- is mediated by work satisfaction:

```
if (!require("bda")) install.packages("bda")
```

```
## Loading required package: bda
```

```
library(bda)
mediation.test(nurses$WorkSat,nurses$Exhaus,nurses$Commit)
```

```
##                 Sobel       Aroian      Goodman
## z.value -2.972270400 -2.936471185 -3.009411683
## p.value  0.002956062  0.003319697  0.002617542
```

In the following output, under Sobel,

- we can see that p.value is significant,
- as the presence of work satisfaction in the model
  - decreases the effect of exhaustion,
- that work satisfaction is significant
  - even though exhaustion is present in the model,
- and that, because the Sobel test is significant,
  - we can confirm that there is indeed
  - a partial mediation of the effect of exhaustion
  - on commitment by work satisfaction.

In other words,

- exhaustion decreases work satisfaction,
- and in turn, work satisfaction increases commitment.

The value resulting from the Sobel test follows a z distribution.

In order to obtain this value,

- the slope coefficients of the predictor regressed on the mediator (a)
  - are multiplied by the slope coefficient of the mediator
  - regressed on the response (b).

This value is then divided by the square root of b squared

- multiplied by the squared standard error of a
- plus a squared multiplied by the squared standard error of b. The formula is as follows:

Showing this is important, as very often,

$$z = \frac{a*b}{\sqrt{(b^2 * s_a^2 + a^2 * s_b^2)}}$$

Figure 3: Sobel

- analysts include dozens or hundreds of predictors in their models
- without taking into consideration that the included predictors
  - could themselves be related to each other.

Readers are therefore advised to check

- for meaningful relationships between the attributes
- they intend to include as predictors in regression analyses
- before drawing conclusions on the final model!

12.1.2.9 Predicting new data

A particularly interesting use of regression

- is to examine how well a model predicts new data.

This is easily achieved in R.

We will first build the dataset named nurses2

- in the same way we did for the first dataset:

```
matcov2 <- unlist(read.csv("./data/matcov2.txt", header = F))
covs2 <- matrix(matcov2, 6, 6)
means2 <- c(4.279, 13.152, 5.156, 39.28, 5.153, 1.875)
set.seed(987)
nurses2 <- data.frame(mvrnorm(n = 40, means2, covs2))
colnames(nurses2) <- c("Commit","Exhaus","Depers","Accompl",
                       "WorkSat","WFC")
predicted <- predict.lm(model1, nurses2)
```

The following output shows that the correlation

- between the predicted values
- and the real values is 0.5766194.

This value is significant and might seem pretty good at first sight:

```
cor.test(predicted, nurses2$Commit)
```

```
##
##  Pearson's product-moment correlation
##
## data:  predicted and nurses2$Commit
## t = 4.3506, df = 38, p-value = 9.848e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```

```
##  0.3231561 0.7528925
## sample estimates:
##       cor
## 0.5766194
```

Let's square this value

- to know how much of the variance in the commitment of the individuals
- of the second sample is predicted by the model:

The output is 33.24899.

This means only 33 percent of the variance in commitment

- is predicted by the model,
- compared to 55 percent in the training data!

Now, we can also compute the residuals:

```
residuals_test <- nurses2$Commit - predicted
```

We are now able to compute the F value for our model.

Statistical Tests

t-statistic

F-test

We have seen that the F value is used

- to assess the overall significance of the model.

In our case, the F value is obtained as follows:

1) First, we need to know the number of degrees of freedom for the model;

- this is equal to the number of predictors we have, which is 3.
- We also need the degrees of freedom for the error;
  - this is the number of observations
  - minus the degrees of freedom of the model, minus 1.

2) We then compute the sum of squares for the model

- as the sum of squared differences
  - between the predicted values
  - and the mean of the response.

The sum of squares for the error is obtained as

- the sum of the squared differences
- between the observed and the predicted values.

3) We then compute the mean squares for the model

- as the sum of squares for the model
- divided by the degrees of freedom for the model.

We compute the mean squares for the error

- as the sum of squares for the error
- divided by the degrees of freedom for the error.

4) Finally, we obtain the F-statistic

- by dividing the means squares for the model
- by the mean squares for the error.

The following function does just that:

```
ComputeF <- function(predicted, observed, npred) {
  DFModel <- npred  # the number of predictors
  DFError <- length(observed) - DFModel - 1
  SSModel <- sum((predicted - mean(observed))^2)
  SSError <- sum((observed - predicted)^2)
  MSModel <- SSModel / DFModel
  MSError <- SSError / DFError
  F <- MSModel / MSError
  F
}

ComputeF(unlist(model1[5]), nurses$Commit, 3)
```

```
## [1] 14.66868
```

```
ComputeF(predicted, nurses2$Commit, 3)
```

```
## [1] 10.4842
```

The outputted F value is 10.4842.

We can test this value using the following line of code.

The output shows that the threshold F value

- at a ceiling of 0.05 on the F distribution for our model is 2.866266:

```
qf(.95, df1 = 3, df2 = 36)
```

```
## [1] 2.866266
```

We can therefore, trust that our model significantly predicts new data.

12.1.2.10 Robust regression

In the example datasets that we used in this section,

- we have seen that some observations might threaten
- the reliability of our results,
  - because of the deviations of their residuals from a normal distribution.

The Shapiro-Wilk test performed on the residuals of model1 (nurses dataset)

- has shown that the distribution of the residuals
- was not significantly different from a normal distribution.

However, let's be particularly cautious

- and analyze the same data using robust regression.

As we mentioned earlier, robust regression

- does not require the residuals to be normally distributed,
- and therefore, fits our purpose.

We will not explore the algorithm.

- For details about this, the reader can consult
  - Robust Regression in R by Fox and Weisberg, in readings.
- Here, we simply perform robust regression using the rlm() function
  - of the MASS package.

Let's first install and load it:

```r
model1.rr <- rlm(Commit ~ Exhaus + Depers + Accompl, data = nurses)
summary(model1.rr)
```

```
##
## Call: rlm(formula = Commit ~ Exhaus + Depers + Accompl, data = nurses)
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.4052046 -0.3233886 -0.0003426  0.3734567  1.0108386
##
## Coefficients:
##             Value   Std. Error t value
## (Intercept)  4.3602  0.3849    11.3271
## Exhaus      -0.0518  0.0083    -6.2306
## Depers      -0.0279  0.0191    -1.4602
## Accompl      0.0338  0.0100     3.3676
##
## Residual standard error: 0.5536 on 36 degrees of freedom
```

You might notice that the output of rlm() is laconical

- in comparison to the output of lm().
- There are no p-values provided, no R-squared values, no F test.

This makes the use of rlm() quite unpractical,

- as the user will have to compute them by hand.

There is so much controversy on how to do it

- that the computations in other software packages are currently questioned!

The reader interested in computing the robust R-squared

- can read the paper
  - A robust coefficient of determination for regression
  - by Renaud and Victoria-Feser (2010), which is in readings.

For our example,

- it seems that the results using lm() and rlm() are pretty similar
- (see the output of the preceding summary of model1).

Therefore, relying on lm() is advised here.

However, if you want to be really sure,

- why not try bootstrapping.

12.1.2.11 Bootstrapping (Advanced topic)

The bootstrap is covered in ISLR Chapter 5 Resampling Methods, in Section 5.2.

The principle of (nonparametric) bootstrapping

- is to create a number of sample K of size N
  - drawn with replacement from the original sample,
- where N is the original sample size.

The parameters are estimated for each sample separately.

This allows computing their confidence intervals,

- a measure of the variability of the parameters.

Apart from making deviations from normal distributions less problematic,

- using bootstrapping is useful for samples
- that have a small number of observations
  - (less than 100), as with ours.

Bootstrapping is easily performed using several functions in R

- for instance, the boot() function in the boot package.

But let's have a little fun and perform bootstrapping ourselves, 2,000 times.

We will first generate the samples and obtain the estimates.

We then display the estimates for the first six samples

- (rounded to the third decimal place):

```
ret <- data.frame(matrix(nrow = 0, ncol = 6))
set.seed(567)
for (i in 1:2000) {
  data <- nurses[sample(nrow(nurses), 40, replace = T),]
  model_i <- lm(Commit ~ Exhaus + Depers + Accompl,
                data = data)
  ret <- rbind(ret,c(coef(model_i),summary(model_i)$r.square,
                     summary(model_i)$fstatistic[1]))
}
names(ret) <- c("Intercept","Exhaus","Depers",
               "Accomp","R2","F")
round(head(ret), 3)
```

```
##   Intercept Exhaus Depers Accomp    R2      F
## 1     4.080 -0.037 -0.055  0.041 0.585 16.928
## 2     4.196 -0.052 -0.048  0.040 0.694 27.273
## 3     5.054 -0.052 -0.047  0.022 0.736 33.416
## 4     4.103 -0.041 -0.042  0.037 0.545 14.373
## 5     4.663 -0.041 -0.022  0.022 0.454  9.980
## 6     4.525 -0.049 -0.035  0.029 0.497 11.874
```

```
set.seed(567)
sample(nrow(nurses), 40, replace = T)
```

```
##  [1] 30 36 26 20 11 10  3 21 24 22 14 11 15 24  1  3 21 30  2 12  6 21  9
## [24] 16  1 34 37 34 16 39 22 29  2 38 29 38 37 28 12  1
```

```
qnorm(0.975)
```

```
## [1] 1.959964
```

```
CIs <- data.frame(matrix(nrow = ncol(ret), ncol = 2))
for (j in 1:ncol(ret)) {
  M <- mean(ret[,j])
  SD <- sd(ret[,j])
  lowerb <- M - (1.96 * (SD / sqrt(2000)))
  upperb <- M + (1.96 * (SD / sqrt(2000)))
  CIs[j,1] <- round(lowerb,3)
  CIs[j,2] <- round(upperb,3)
}
names(CIs) <- c("95% C.I.lower bound", "95% C.I.upper bound")
```

```r
rownames(CIs) <- colnames(ret)
CIs
```

```
##           95% C.I.lower bound 95% C.I.upper bound
## Intercept               4.297               4.325
## Exhaus                 -0.048              -0.048
## Depers                 -0.029              -0.027
## Accomp                  0.033               0.033
## R2                      0.558               0.570
## F                      18.179              19.139
```

The confidence intervals

- encompass all the values
- between the lower and upper bounds.

We can see that no confidence interval contains 0,

- meaning that, with a 95 percent threshold,
- values reported are statistically different from 0
  - (more correctly put, there is only a 5 percent chance
  - of observing values inside these bounds
  - if the true value of the parameters in the population is 0).

So we conclude that

- bootstrapped coefficients are different from 0,
- as is the multiple R-squared value.

As you might have noticed,

- the value to which to compare the confidence intervals for F is not 0,
- but a value that depends upon the degrees of freedom.

We computed this value earlier and it was 2.866266.

As the confidence interval for F does not include this value,

- we can be assured that the bootstrapped model
- predicts a significant part of variance.

12.1.2.12 Summary

We examined how to develop functions

- that perform simple regression analyses,
- and how to multiply regression in R
- using a real life example.

We have examined the importance of significance tests for regression,

- and have briefly discussed
  - robust regression
  - and bootstrapping.

Note that, when data about the predictors and the response

- are collected simultaneously,
- causation cannot be established.

In order to ascertain causation,

- data must be collected longitudinally

- that is, the predictors before the response.

12.1.2.13 Links

12.1.2.13.1 Learning Predictive Analytics with R, Eric Mayor, Packtpub 2015