Online Tutorial on Regression Modeling with Actuarial and Financial Applications

Edward W. (Jed) Frees, University of Wisconsin-Madison

Contents

Preface **5**
About Regression Modeling . . . 5
Resources . . . 5
Tutorial Description . . . 5
Welcome to the Tutorial Video . . . 6

1 Regression and the Normal Distribution **9**
1.1 Fitting a normal distribution . . . 9
1.1.1 Video . . . 9
1.1.2 Exercise. Fitting Galton's height data . . . 12
1.1.3 Exercise. Visualizing child's height distribution . . . 13
1.2 Visualizing distributions . . . 13
1.2.1 Video . . . 13
1.2.2 Exercise. Visualizing bodily injury claims with density plots . . . 17
1.3 Summarizing distributions . . . 17
1.3.1 Video . . . 18
1.3.2 Exercise. Summarizing bodily injury claims with box and qq plots . . . 21
1.3.3 Exercise. Effects on distributions of removing the largest claim . . . 22
1.4 Transformations . . . 22
1.4.1 Video . . . 22
1.4.2 Exercise. Distribution of transformed bodily injury claims . . . 24

2 Basic Linear Regression **25**
2.1 Correlation . . . 25
2.1.1 Video . . . 25
2.1.2 Exercise. Correlations and the Wisconsin lottery . . . 27
2.2 Method of least squares . . . 28
2.2.1 Video . . . 28
2.2.2 Exercise. Least squares fit using housing prices . . . 30
2.3 Understanding variability . . . 30
2.3.1 Video . . . 30
2.3.2 Exercise. Summarizing measures of uncertainty . . . 32
2.3.3 Exercise. Effects of linear transforms on measures of uncertainty . . . 32
2.4 Statistical inference . . . 32
2.4.1 Video . . . 33
2.4.2 Exercise. Statistical inference and Wisconsin lottery . . . 34
2.5 Diagnostics . . . 35
2.5.1 Video . . . 35
2.5.2 Exercise. Assessing outliers in lottery sales . . . 41

3 Multiple Linear Regression **43**
Term Life Data . . . 43

3.1 Method of least squares . . . 45
3.1.1 Video . . . 45
3.1.2 Exercise. Least squares and term life data . . . 49
3.1.3 Exercise. Interpreting coefficients as proportional changes . . . 49
3.1.4 Exercise. Interpreting coefficients as elasticities . . . 49
3.2 Statistical inference and multiple linear regresson . . . 50
3.2.1 Video . . . 50
3.2.2 Exercise. Statistical inference and term life . . . 52
3.3 Binary variables . . . 52
3.3.1 Video . . . 52
3.3.2 Exercise. Binary variables and term life . . . 56
3.4 Categorical variables . . . 56
3.4.1 Video . . . 57
3.4.2 Exercise. Categorical variables and Wisconsin hospital costs . . . 59
3.5 General linear hypothesis . . . 60
3.5.1 Video . . . 60
3.5.2 Exercise. Hypothesis testing and term life . . . 61
3.5.3 Exercise. Hypothesis testing and Wisconsin hospital costs . . . 62
3.5.4 Exercise. Hypothesis testing and auto claims . . . 62

4 Variable Selection 69
4.1 An iterative approach to data analysis and modeling . . . 69
4.1.1 Video . . . 69
4.1.2 MC Exercise. An iterative approach to data modeling . . . 72
4.2 Automatic variable selection procedures . . . 73
4.2.1 Video . . . 73
4.2.2 Exercise. Data-snooping in stepwise regression . . . 74
4.3 Residual analysis . . . 75
4.3.1 Video . . . 75
4.3.2 Exercise. Residual analysis and risk manager survey . . . 75
4.3.3 Exercise. Added variable plot and refrigerator prices . . . 76
4.4 Unusual observations . . . 78
4.4.1 Video . . . 79
4.4.2 Exercise. Outlier example . . . 81
4.4.3 Exercise. High leverage and risk manager survey . . . 81
4.5 Collinearity . . . 81
4.5.1 Video . . . 82
4.5.2 Exercise. Collinearity and term life . . . 82
4.6 Selection criteria . . . 83
4.6.1 Video . . . 83
4.6.2 Exercise. Cross-validation and term life . . . 84

5 Interpreting Regression Results 87
5.1 Case study: MEPS health expenditures . . . 87
5.1.1 Video . . . 87
5.1.2 Exercise. Summarizing data . . . 89
5.1.3 Exercise. Fit a benchmark multiple linear regression model . . . 89
5.1.4 Exercise. Variable selection . . . 90
5.1.5 Exercise. Model comparisons using cross-validation . . . 90
5.1.6 Exercise. Out of sample validation . . . 91
5.2 What the modeling procedure tells us . . . 91
5.2.1 Video . . . 92
5.3 The importance of variable selection . . . 93
5.3.1 Video . . . 93

Preface

Date: 05 November 2018

About Regression Modeling

Statistical techniques can be used to address new situations. This is important in a rapidly evolving risk management world. Analysts with a strong analytical background understand that a large data set can represent a treasure trove of information to be mined and can yield a strong competitive advantage. This book and online tutorial provides budding analysts with a foundation in multiple reression. Viewers will learn about these statistical techniques using data on the demand for insurance, healthcare expenditures, and other applications. Although no specific knowledge of actuarial or risk management is presumed, the approach introduces applications in which statistical techniques can be used to analyze real data of interest.

Resources

- This tutorial is based on the book Regression Modeling with Actuarial and Financial Applications.
  - For resources associated with the book, please visit the Regression Modeling book web site.
- For advanced regression applications in insurance, you may be interested in the series, Predictive Modeling Applications in Actuarial Science.
  - Sample code and data for the series are available at series website.
- An earlier version of this tutorial, a Short Course constructed for Indonesian actuaries, uses the Datacamp learning platform.

Tutorial Description

- This online tutorial is designed to guide you through the foundations of regession with applications in actuarial science.
- Anticipated completion time is approximately six hours.
- The tutorial assumes that you are familiar with the foundations in the statistical software `R`, such as Datacamp's Introduction to R.

General Layout. There are five chapters in this tutorial that summarize the foundations of multiple linear regression. Each chapter is subdivided into several sections. At the beginning of each section is a short video, typically 4-8 minutes, that summarizes the section key learning outcomes. Following the video, you can see more details about the underlying `R` code for the analysis presented in the video.

Role of Exercises. Following each video, there are one or two exercises that allow you to practice skills to make sure that you fully grasp the learning outcomes. The exercises are implented using an online learning platfor provided by Datacamp so that you need not install `R`. Feedback is programmed into the exercises so that you will learn a lot by making mistakes! You will be pacing yourself, so always feel free to reveal the

answers by hitting the `Solution` tab. Remember, going through quickly is not equivalent to learning deeply. Use this tool to enhance your understanding of one of the foundations of data science, regression analysis.

Welcome to the Tutorial Video

In this video, you learn how to:

- Describe regression briefly, i.e., in a nutshell
- Explain Galton's height example as a regression application

---

Video

Video Overhead

Show Overhead A. Galton's 1885 Regression Data

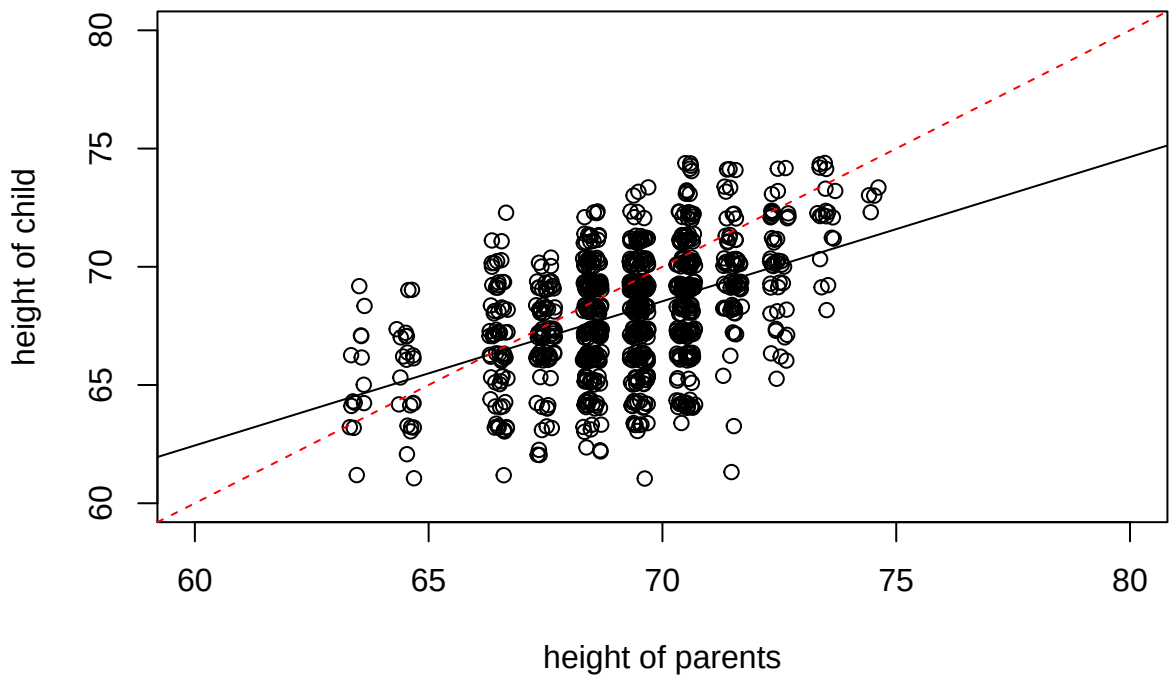
Height of adult child in inches	< 64.0	64.5	65.5	66.5	67.5	68.5	69.5	70.5	71.5	72.5	> 73.0	Totals
> 73.7	–	–	–	–	–	–	5	3	2	4	–	14
73.2	–	–	–	–	–	3	4	3	2	2	3	17
72.2	–	–	1	–	4	4	11	4	9	7	1	41
71.2	–	–	2	–	11	18	20	7	4	2	–	64
70.2	–	–	5	4	19	21	25	14	10	1	–	99
69.2	1	2	7	13	38	48	33	18	5	2	–	167
68.2	1	–	7	14	28	34	20	12	3	1	–	120
67.2	2	5	11	17	38	31	27	3	4	–	–	138
66.2	2	5	11	17	36	25	17	1	3	–	–	117
65.2	1	1	7	2	15	16	4	1	1	–	–	48
64.2	4	4	5	5	14	11	16	–	–	–	–	59
63.2	2	4	9	3	5	7	1	1	–	–	–	32
62.2	–	1	–	3	3	–	–	–	–	–	–	7
< 61.2	1	1	1	–	–	1	–	1	–	–	–	5
Totals	14	23	66	78	211	219	183	68	43	19	4	928

Show Overhead B. Supporting R Code

```
# Reformat Data Set
#heights <- read.csv("CSVData\\GaltonFamily.csv",header = TRUE)
heights <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/c85ede6c205d22049
str(heights)
head(heights)
heights$child_ht <- heights$CHILDC
heights$parent_ht <- heights$PARENTC
heights2 <- heights[c("child_ht","parent_ht")]
```

```
#heights <- read.csv("CSVData\\galton_height.csv",header = TRUE)
heights <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/c85ede6c205d22049
plot(jitter(heights$parent_ht),jitter(heights$child_ht), ylim = c(60,80), xlim = c(60,80),
     ylab = "height of child", xlab = "height of parents")
```

```r
abline(lm(heights$child_ht~heights$parent_ht))
abline(0,1,col = "red", lty=2)
```




```r
summary(lm(heights$child_ht~heights$parent_ht))
```

```
Call:
lm(formula = heights$child_ht ~ heights$parent_ht)

Residuals:
    Min      1Q  Median      3Q     Max
-8.2577 -1.4280  0.1323  1.5720  5.7918

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       25.84856    2.69009   9.609   <2e-16 ***
heights$parent_ht  0.60992    0.03882  15.710   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.26 on 926 degrees of freedom
Multiple R-squared:  0.2104,    Adjusted R-squared:  0.2096
F-statistic: 246.8 on 1 and 926 DF,  p-value: < 2.2e-16
```

Chapter 1

Regression and the Normal Distribution

Chapter description

Regression analysis is a statistical method that is widely used in many fields of study, with actuarial science being no exception. This chapter introduces the role of the normal distribution in regression and the use of logarithmic transformations in specifying regression relationships.

1.1 Fitting a normal distribution

In this section, you learn how to:

- Calculate and interpret two basic summary statistics
- Fit a data set to a normal curve
- Calculate probabilities under a standard normal curve

1.1.1 Video

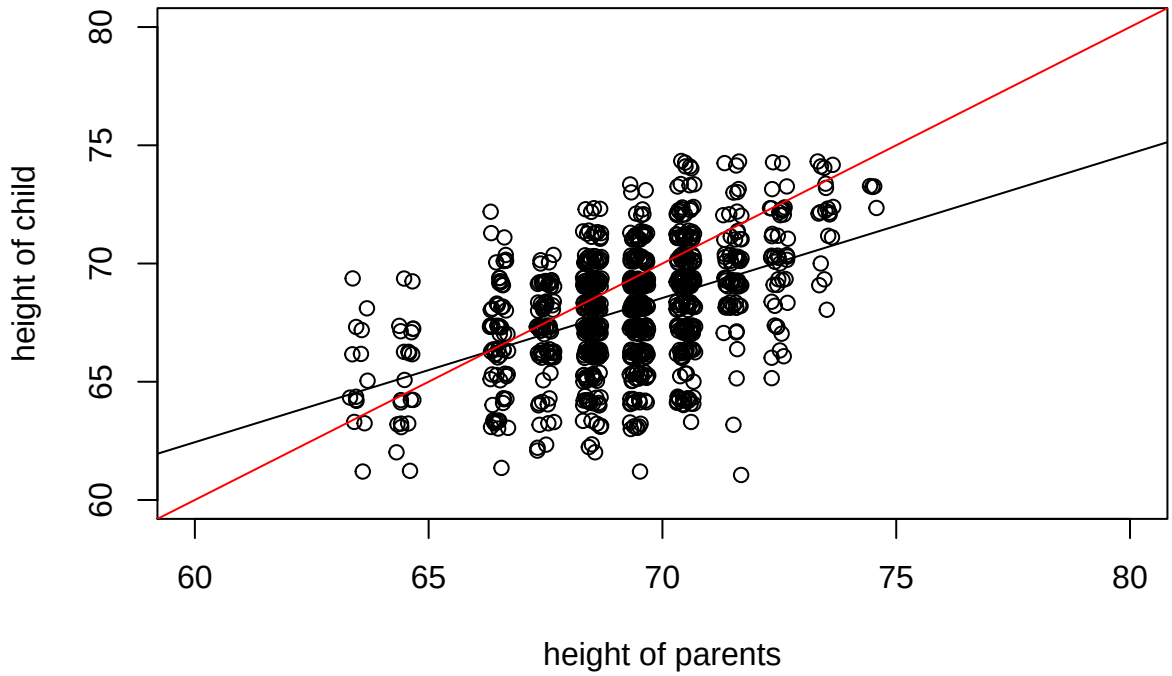
Video Overhead Details

Show Overhead A Details. Description of the data

To illustrate a data set that can be analyzed using regression methods, we consider some data included in Galton's 1885 paper. These data include the heights of 928 adult children (child_ht), together with an index of their parents' height (parent_ht). Here, all female heights were multiplied by 1.08, and the index was created by taking the average of the father's height and rescaled mother's height. Galton was aware that the parents' and the adult child's height could each be adequately approximated by a normal curve. In developing regression analysis, he provided a single model for the joint distribution of heights.

```
heights <- read.csv("CSVData\\galton_height.csv", header = TRUE)
#heights <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/c85ede6c205d220
plot(jitter(heights$parent_ht),jitter(heights$child_ht), ylim = c(60,80), xlim = c(60,80),
     ylab = "height of child", xlab = "height of parents")
```

```
abline(lm(heights$child_ht~heights$parent_ht))
abline(0,1,col = "red")
```



Show Overhead B Details. Read and examine data structure

The data has already been read into a dataset called `heights`. Examine the *structure* of the data with the function str() and use the head() command to looks at the first few records.

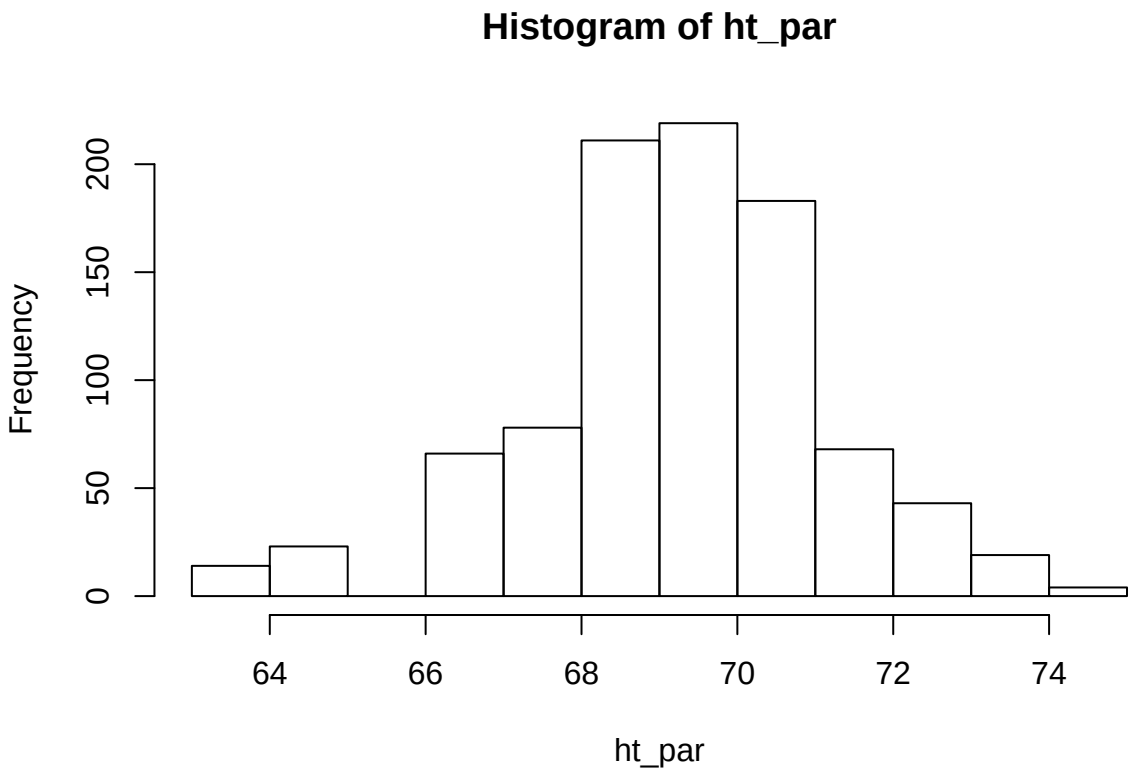
```
heights <- read.csv("CSVData\\galton_height.csv",header = TRUE)
#heights <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/c85ede6c205d220
str(heights)
head(heights)
```

```
'data.frame':   928 obs. of  2 variables:
 $ child_ht : num  72.2 73.2 73.2 73.2 68.2 ...
 $ parent_ht: num  74.5 74.5 74.5 74.5 73.5 73.5 73.5 73.5 73.5 73.5 ...
  child_ht parent_ht
1     72.2      74.5
2     73.2      74.5
3     73.2      74.5
4     73.2      74.5
5     68.2      73.5
6     69.2      73.5
```

Show Overhead C Details. Summary stats for parents' height

Next, examine the distribution of the child's height and then examine the distribution of the parents height.

```
ht_par <- heights$parent_ht
hist(ht_par)
```

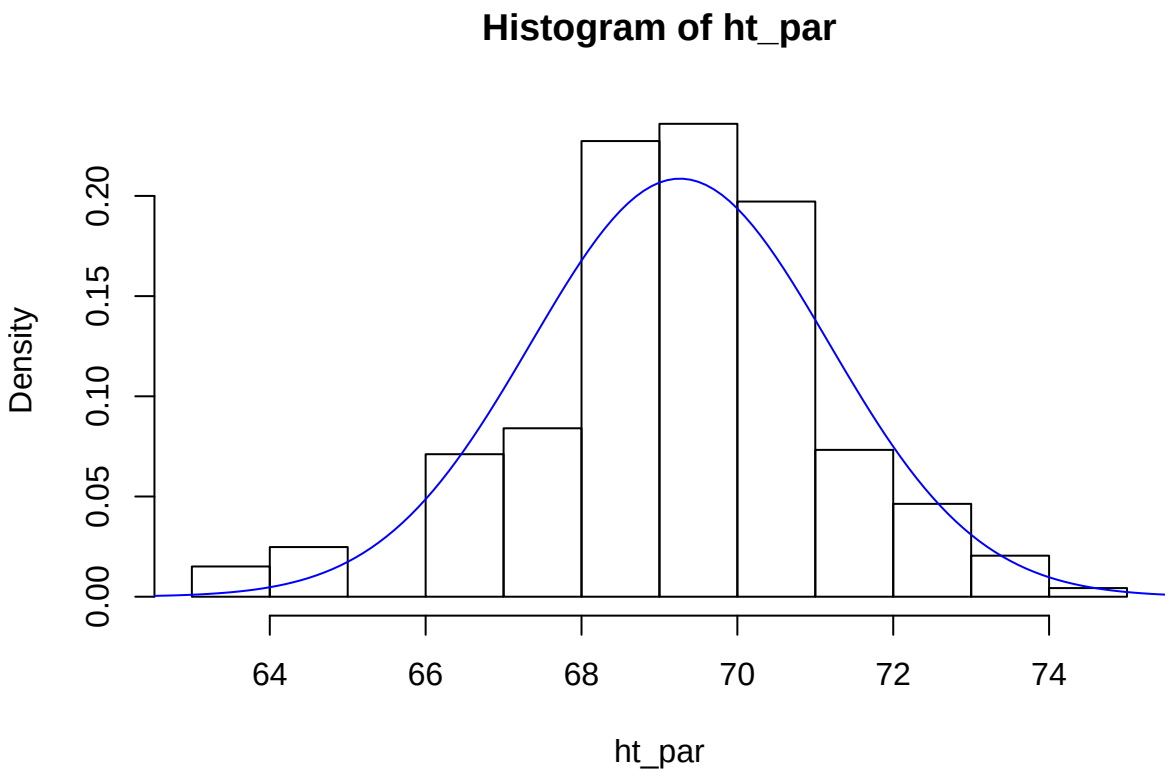



```
mean(ht_par)
sd(ht_par)
```

```
[1] 69.26293
[1] 1.912274
```

Show Overhead D. Fit a normal curve to parents' height details

```
(mparent <- mean(ht_par))
(sdparent <- sd(ht_par))
x <- seq(60, 80,by = 0.1)
hist(ht_par, freq = FALSE)
lines(x, dnorm(x, mean = mparent, sd = sdparent), col = "blue")
```




```
[1] 69.26293
[1] 1.912274
```

Show Overhead E Details. Use the normal approximation to determine the probability of the height of tall parents

```
TallHeight <- 72
pnorm(TallHeight, mean = mparent, sd = sdparent)
pnorm(72, mean = mean(ht_par), sd = sd(ht_par))
(StdUnitsTallHeight <- (TallHeight - mparent)/sdparent)
pnorm(StdUnitsTallHeight, mean = 0, sd = 1)
```

```
[1] 0.9238302
[1] 0.9238302
[1] 1.431317
[1] 0.9238302
```

1.1.2 Exercise. Fitting Galton's height data

Assignment Text

The Galton data has already been read into a dataframe called `heights`. These data include the heights of 928 adult children `child_ht`, together with an index of their parents' height `parent_ht`. The video explored the distribution of the parents' height; in this assignment, we investigate the distribution of the heights of the adult children.

Instructions

- Define the height of an adult child as a global variable
- Use the function mean() to calculate the mean and the function sd() to calculate the standard deviation
- Use the normal approximation and the function pnorm() determine the probability that an adult child's height is less than 72 inches

Hint. Remember that we can reference a variable, say `var`, from a data set such as `heights`, as `heights$var`.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNoZWlnaHRzIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxc

1.1.3 Exercise. Visualizing child's height distribution

Assignment Text

As in the prior exercise, from the Galton dataset `heights`, the heights of 928 adult children have been used to create a global variable called `ht_child`. We also have basic summary statistics, the mean height `mchild` and the standard deviation of heights in `sdchild`. In this exercise, we explore the fit of the normal curve to this distribution.

Instructions

- To visualize the distribution, use the function hist() to calculate the histogram. Use the `freq = FALSE` option to give a histogram with proportions instead of counts.
- Use the function seq() to determine a sequence that can be used for plotting. Then, with the function lines(), superimpose a normal curve on the histogram
- Determine the probability that a child's height is greater than 72 inches

Hint 1. Use the function dnorm() to calculate the normal density, similar to the cumulative probabilites that you calculated using pnorm()

Hint 2. To calculate probabilities greater that an amount, simply use 1 minus the cumulative probability

Pre-exercise code

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNoZWlnaHRzIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxc

1.2 Visualizing distributions

In this section, you learn how to:

- Calculate and interpret distributions using histograms
- Calculate and interpret distributions using density plots

1.2.1 Video

Video Overhead Details

Show Overhead Details. Data description

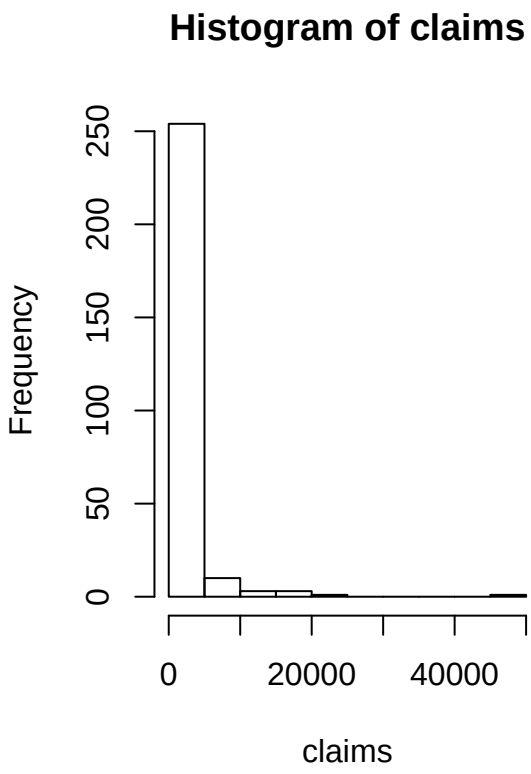
For our first look at an insurance data set, we consider data from Rempala and Derrig (2005). They considered claims arising from automobile bodily injury insurance coverages. These are amounts incurred for outpatient medical treatments that arise from automobile accidents, typically sprains, broken collarbones and the like. The data consists of a sample of 272 claims from Massachusetts that were closed in 2001 (by

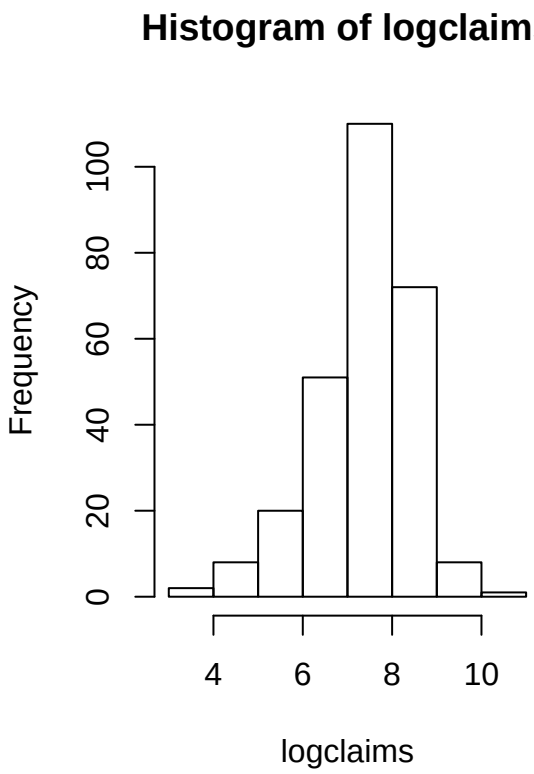
"closed," we mean that the claim is settled and no additional liabilities can arise from the same accident). Rempala and Derrig were interested in developing procedures for handling mixtures of "typical" claims and others from providers who reported claims fraudulently. For this sample, we consider only those typical claims, ignoring the potentially fraudulent ones.

```
# Reformat Data Set
injury <- read.csv("CSVData\\MassBodilyInjury.csv",header = TRUE)
str(injury)
head(injury)
#  PICK THE SUBSET OF THE DATA CORRESPONDING TO PROVIDER A
injury2 <- subset(injury, providerA ! = 0 )
injury2$claims <- 1000*injury2$claims
injury2$logclaims <- log(injury2$claims)
injury3 <- injury2[c("claims","logclaims")]
#write.csv(injury3,"CSVData\\MassBI.csv",row.names = FALSE)
```

Show Overhead A Details. Bring in Data, Introduce Logarithmic Claims

```
injury <- read.csv("CSVData\\MassBI.csv",header = TRUE)
#  CHECK THE NAMES, DIMENSION IN THE FILE AND LIST THE FIRST 8 OBSERVATIONS  ;
str(injury)
head(injury)
attach(injury)
claims <- injury$claims
par(mfrow = c(1, 2))
hist(claims)
hist(logclaims)
```

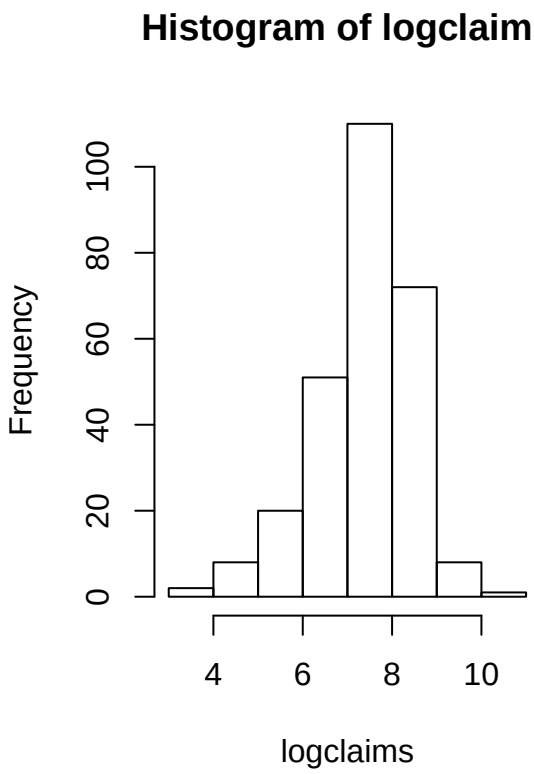


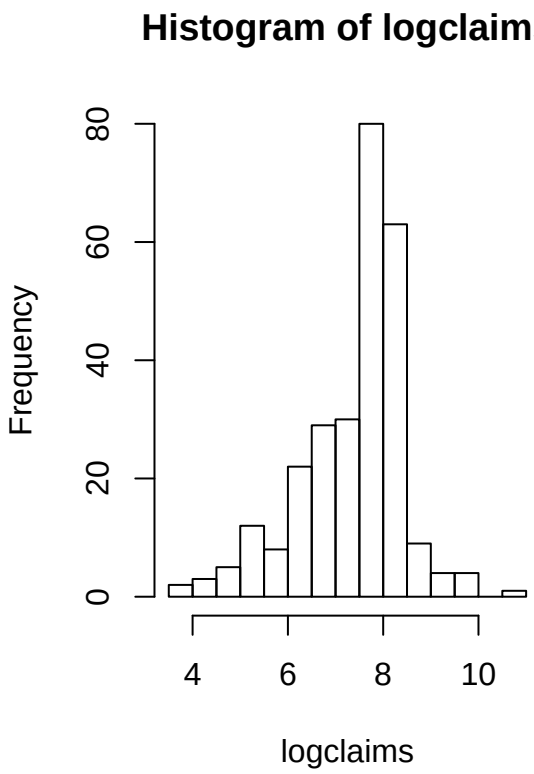



```
'data.frame':   272 obs. of  2 variables:
 $ claims   : int  45 47 70 75 77 92 117 117 140 145 ...
 $ logclaims: num  3.81 3.85 4.25 4.32 4.34 ...
  claims logclaims
1     45  3.806662
2     47  3.850148
3     70  4.248495
4     75  4.317488
5     77  4.343805
6     92  4.521789
```

Show Overhead B Details. Show how to get a finer grid for histograms

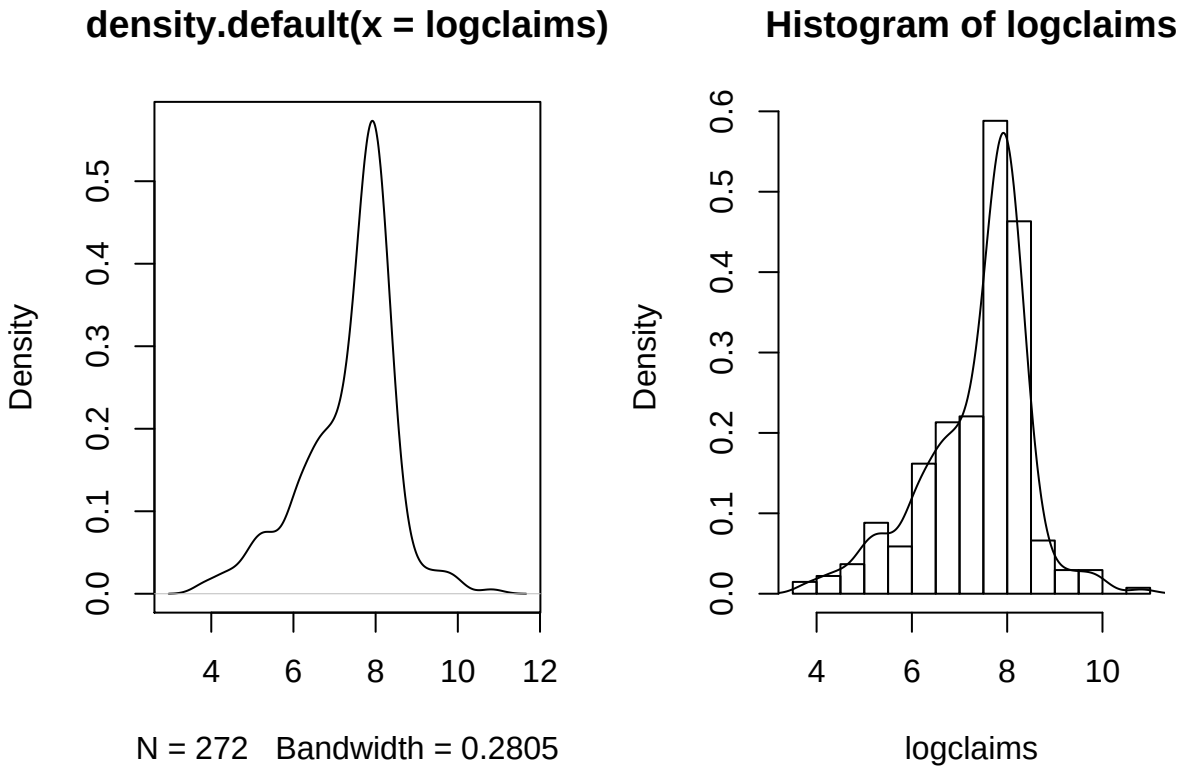
```
par(mfrow = c(1, 2))
hist(logclaims)
hist(logclaims,breaks = 15)
```





Show Overhead C Details. Introduce the density plot

```
par(mfrow = c(1, 2))
plot(density(logclaims))
hist(logclaims, breaks = 15,freq = FALSE)
lines(density(logclaims))
```



1.2.2 Exercise. Visualizing bodily injury claims with density plots

Assignment Text

In the prior video, you learned about the Massachusetts bodily injury dataset. This dataframe, `injury`, has been read in and the global variable `claims` has been created. This assignment reviews the hist() function for visualizing distributions and allows you to explore density plotting, a smoothed version of the histogram.

Instructions

- Use the function log() to create the logarithmic version of the claims variable
- Calculate a histogram of logarithmic with 40 bins using an option in the hist() function, `breaks =`.
- Create a density plot of logarithmic claims using the functions plot() and density().
- Repeat the density plot, this time using a more refined bandwidth equal to 0.03. Use an option in the density() function, `bw =`.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNpbmp1cnkgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXE

1.3 Summarizing distributions

In this section, you learn how to:

- Calculate and interpret basic summary statistics
- Calculate and interpret distributions using boxplots

- Calculate and interpret distributions using qq plots

1.3.1 Video

Video Overhead Details

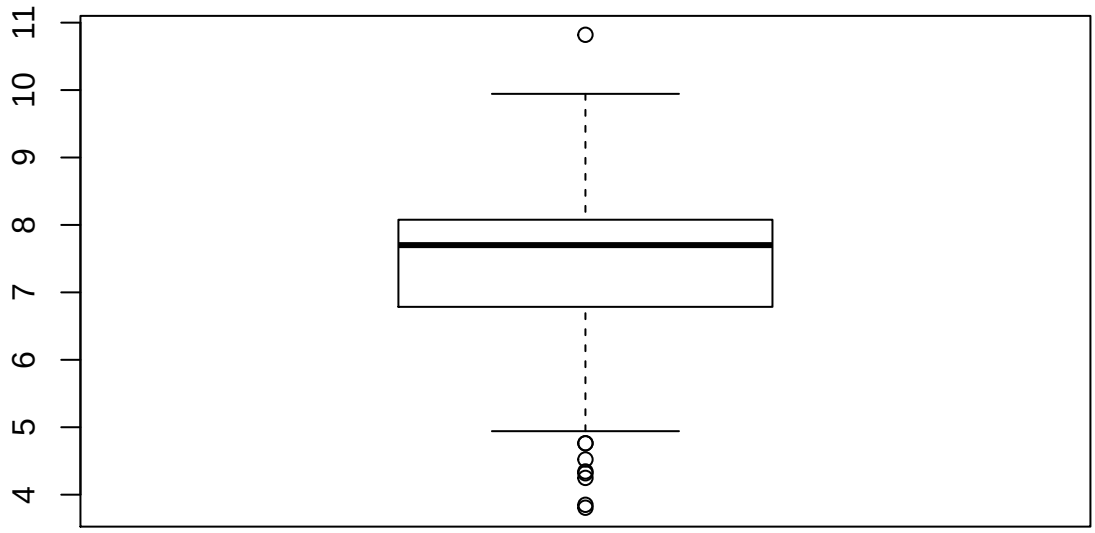
Show Overhead A Details. Summary statistics

```
injury <- read.csv("CSVData\\MassBI.csv",header = TRUE)
#injury <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/8cca19d0503fcf6e
attach(injury)

# SUMMARY STATISTICS
summary(injury)
sd(claims);sd(logclaims)
length(claims)
```

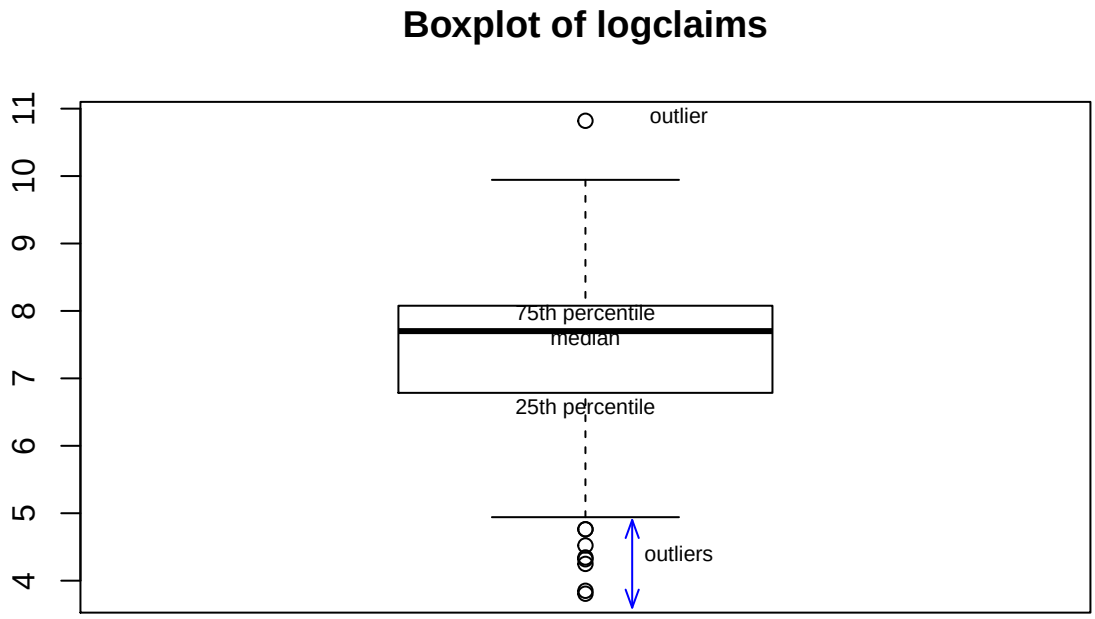
```
     claims          logclaims
 Min.   :   45.0   Min.   : 3.807
 1st Qu.:  892.5   1st Qu.: 6.794
 Median : 2210.0   Median : 7.701
 Mean   : 2697.7   Mean   : 7.388
 3rd Qu.: 3215.0   3rd Qu.: 8.076
 Max.   :50000.0   Max.   :10.820
[1] 3944.445
[1] 1.10093
[1] 272
```

Show Overhead B Details. Boxplot

```
# BASIC BOXPLOT
boxplot(logclaims)
```



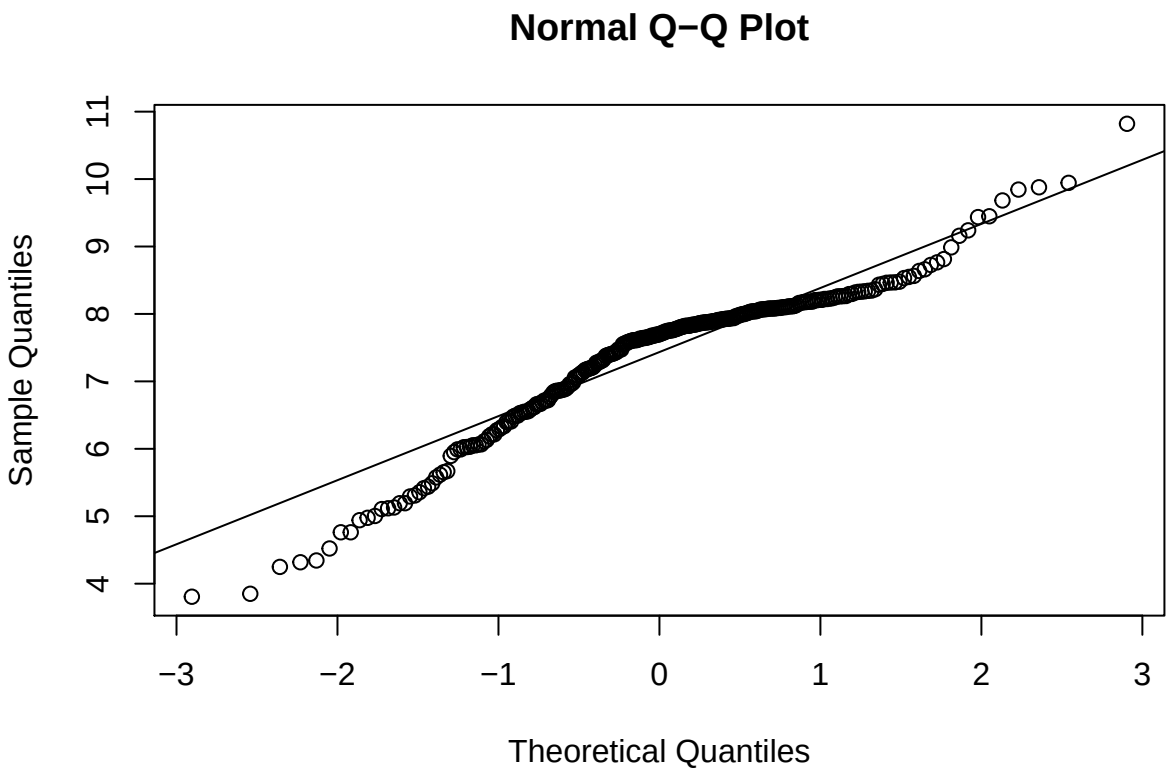

```
quantile(logclaims, probs = 0.75)

# BOXPLOT WITH ANNOTATION
boxplot(logclaims, main = "Boxplot of logclaims")
text(1,  7.6,  "median", cex = 0.7)
text(1,  6.55, "25th percentile", cex = 0.7)
text(1,  7.95, "75th percentile", cex = 0.7)
arrows(1.05, 4.9, 1.05, 3.6, col = "blue", code = 3, angle = 20, length = 0.1)
text(1.1,  4.4, "outliers", cex = 0.7)
text(1.1, 10.9, "outlier",  cex = 0.7)
```




```
     75%
8.075579
```

Show Overhead C Details. QQ Plot

```
summary(injury)
quantile(claims, probs = 0.75)
quantile(logclaims, probs = 0.75)
log(quantile(claims, probs = 0.75))
qnorm(p = 0.75, mean = mean(logclaims), sd = sd(logclaims))
(qnorm(p = 0.75, mean = mean(logclaims), sd = sd(logclaims)) -mean(logclaims)) /
       sd(logclaims)
qnorm(p = 0.75, mean = 0, sd = 1)

#  QUANTILE - QUANTILE PLOT
qqnorm(logclaims)
qqline(logclaims)
```




```
     claims          logclaims
 Min.   :   45.0   Min.   : 3.807
 1st Qu.:  892.5   1st Qu.: 6.794
 Median : 2210.0   Median : 7.701
 Mean   : 2697.7   Mean   : 7.388
 3rd Qu.: 3215.0   3rd Qu.: 8.076
 Max.   :50000.0   Max.   :10.820
 75%
3215
     75%
8.075579
     75%
8.075583
[1] 8.131056
[1] 0.6744898
[1] 0.6744898
```

1.3.2 Exercise. Summarizing bodily injury claims with box and qq plots

Assignment Text

The Massachusetts bodily injury data has already been read and used to create the global variable `claims` representing bodily injury claims. The previous video showed how to present the distribution of logarithmic claims which appeared to be approximately normally distributed. However, users are not really interested in log dollars but want to know about a unit of measurement that is more intuitive, such as dollars.

So this assignment is based on claims, not the logarithmic version. You will use the functions boxplot() and qqnorm() to visualize the distribution through boxplots and quantile-quantile, or qq-, plots. But, because we are working with such a skewed distribution, do not be surprised that it is difficult to interpret these results readily.

Instructions

- Produce a box plot for claims
- Determine the 25th empirical percentile for claims using the quantile() function.
- Determine the 25th percentile for claims based on a normal distribution using the qnorm() function.
- Produce a normal qq plot for claims using the function qqnorm(). The qqline() function is handy for producing a reference line.

Hint. Note that qnorm() (one `q`) is for a normal quantile and qqnorm(). (two `q`'s!) is for the normal qq plot

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNpbmp1cnkgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxE

1.3.3 Exercise. Effects on distributions of removing the largest claim

Assignment Text

The Massachusetts bodily injury dataframe `injury` has been read in; our focus is on the `claims` variable in that dataset.

In the previous exercise, we learned that the Massachusetts bodily injury `claims` distribution was not even close to approximately normal (as evidenced by the box and qq- plots). Non-normality may be induced by skewness (that we will handle via transformations in the next section). But, seeming non-normality can also be induced by one or two very large observations (called an *outlier* later in the course). So, this exercise examines the effects on the distribution of removing the largest claims.

Instructions

- Use the function tail() to examine the `injury` dataset and identify the largest claim
- Use the function subset() to create a subset omitting the largest claim
- Compare the summary statistics of the omitted claim distribution to the full distribution
- Compare the two distributions visually via histograms plotted next to another. `par(mfrow = c(1, 2))` is used to organize the plots you create. Do not alter this code.

Hint. For this data set, the [subset()] argument `claims < 25000` will keep all but the largest claim

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNpbmp1cnkgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxE

1.4 Transformations

In this exercise, you learn how to:

- Symmetrize a skewed distribution using a logarithmic transformation

---

1.4.1 Video

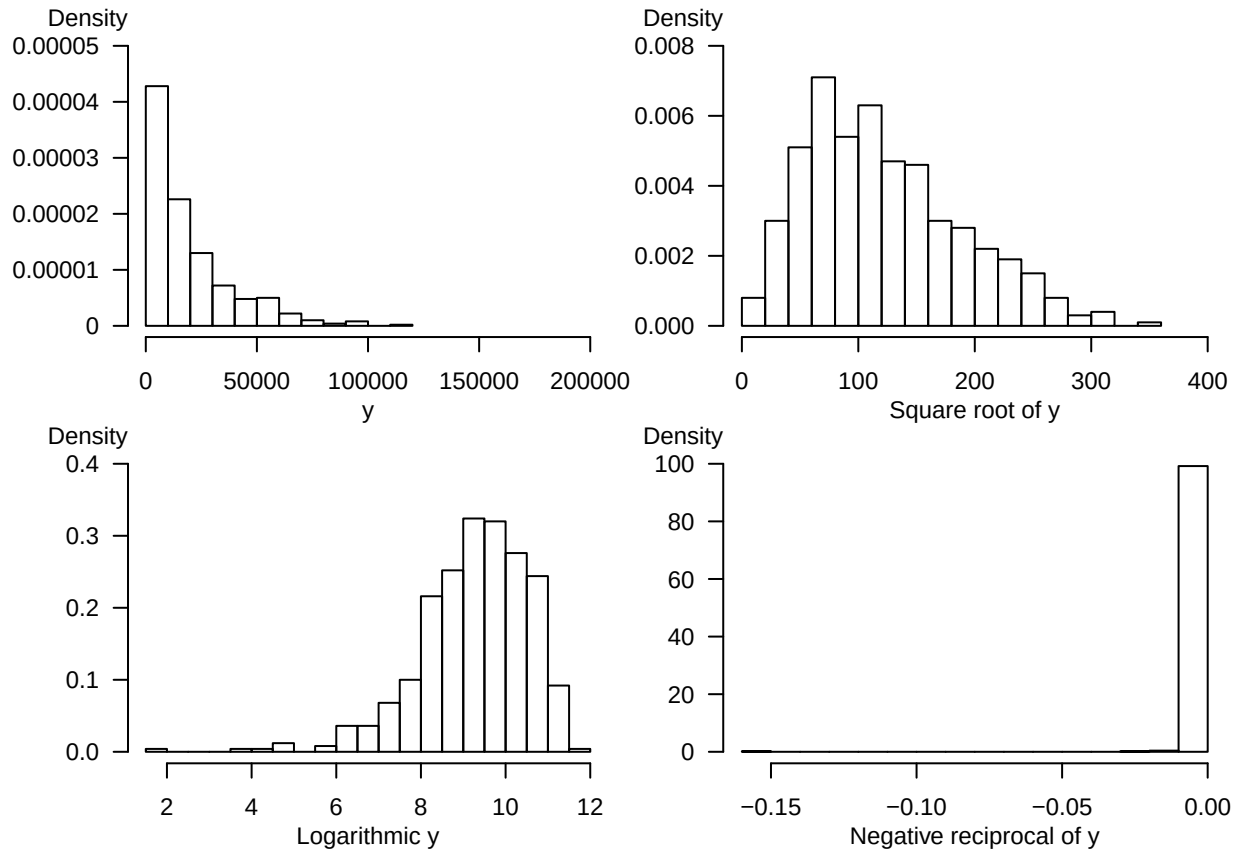
Video Overhead Details

Show Overhead A Details. Simulate a moderately skewed distribution, with transforms

```
#  FIGURE 1.7 - SIMULATE CHI-SQUARE, CREATE 3 TRANSFORMATIONS
set.seed(1237)                    # set the seed of the random number generator
                                  # allows us to replicate results
X1 <- 10000*rchisq(500, df = 2) # generate variables randomly from a skewed distribution
X2 <- X1^(0.5)                    # square root transform, could also use sqrt(X1)
X3 <- log(X1)                     # logarithmic transform
X4 <- -1/X1                       # negative reciprocal transform
```

Show Overhead B Details. Visualize the distributions

```
par(mfrow = c(2, 2), cex = .75, mar = c(3,5,1.5,0))
hist(X1, freq = FALSE,  nclass = 16, main = "", xlab = "", ylab = "",
      las = 1, yaxt = "n",xlim = c(0,200000),ylim = c(0,.00005))
axis(2, at = seq(0,.00005,.00001),las = 1, cex = .3,
  labels = c("0", "0.00001", "0.00002","0.00003", "0.00004", "0.00005"))
mtext("Density", side = 2, at = .000055, las = 1, cex = .75)
mtext("y", side = 1, cex = .75, line = 2)

par(mar = c(3,4,1.5,0.2))
hist(X2, freq = FALSE,  nclass = 16, main = "", xlab = "", ylab = "",
      las = 1,xlim = c(0,400), ylim = c(0,.008))
mtext("Density", side = 2, at = .0088, las = 1, cex = .75)
mtext("Square root of y", side = 1, cex = .75, line = 2)

par(mar = c(3.2,5,1,0))
hist(X3,  freq = FALSE,  nclass = 16, main = "", xlab = "", ylab = "", las = 1, ylim = c(0,.4))
mtext("Density", side = 2, at = .44, las = 1, cex = .75)
mtext("Logarithmic y", side = 1, cex = .75, line = 2)

par(mar = c(3.2,4,1,0.2))
hist(X4, freq = FALSE,  nclass = 16, main = "",xlab = "", ylab = "", las = 1, ylim = c(0,100))
mtext("Density", side = 2, at = 110, las = 1, cex = .75)
mtext("Negative reciprocal of y", side = 1, cex = .75, line = 2)
```



1.4.2 Exercise. Distribution of transformed bodily injury claims

Assignment Text

We have now examined the distributions of bodily injury claims and its logarithmic version. Grudgingly, we have concluded that to fit a normal curve the logarithmic version of claims is a better choice (again, we really do not like log dollars but you'll get used to it in this course). But, why logarithmic and not some other transformations?

A partial response to this question will appear in later chapters when we describe interpretation of regression coefficients. Another partial response is that the log transform seems to work well with skewed insurance data sets, as we demonstrate visually in this exercise.

Instructions

Use the code `par(mfrow = c(2, 2))` so that four graphs appear in a 2 by 2 matrix format for easy comparisons. Plot the density() of

- claims
- square root of claims
- logarithmic claims
- negative reciprocal of claims

Hint. For negative reciprocal claims, use `plot(density(-claims^(-1)))`

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNpbmp1cnkgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXE

Chapter 2

Basic Linear Regression

Chapter description

This chapter considers regression in the case of only one explanatory variable. Despite this seeming simplicity, many deep ideas of regression can be developed in this framework. By limiting ourselves to the one variable case, we can illustrate the relationships between two variables graphically. Graphical tools prove to be important for developing a link between the data and a predictive model.

2.1 Correlation

In this section, you learn how to:

- Calculate and interpret a correlation coefficient
- Interpret correlation coefficients by visualizing scatter plots

---

2.1.1 Video

Video Overhead Details

Show Overhead A Details. Wisconsin lottery data description

```
Lot <- read.csv("CSVData\\Wisc_lottery.csv")
#Lot <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/a792b30fb32b0896dd6
str(Lot)
```

```
'data.frame':   50 obs. of  3 variables:
 $ pop    : int  435 4823 2469 2051 13337 17004 38283 9859 4464 20958 ...
 $ sales  : num  1285 3571 2407 1224 15046 ...
 $ medhome: num  71.3 98 58.7 65.7 96.7 66.4 91 61 91.5 68.8 ...
```

Show Overhead B Details. Summary statistics

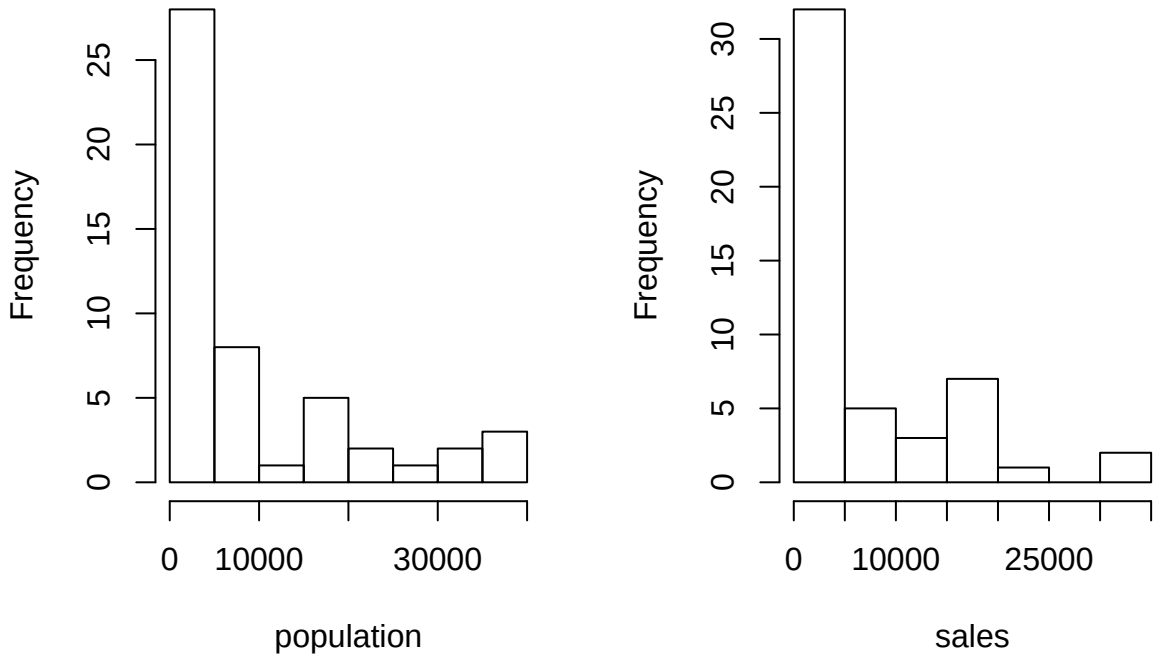
```
#options(scipen = 100, digits = 4)
#numSummary(Lot[,c("pop", "sales")], statistics = c("mean", "sd", "quantiles"), quantiles = c(0,.5,1))
```

```
(as.data.frame(psych::describe(Lot)))[,c(2,3,4,5,8,9)]
#Rcmdr::numSummary(Lot[,c("pop", "sales")], statistics = c("mean", "sd", "quantiles"), quantiles = c(0,
```

```
         n     mean          sd   median   min     max
pop     50 9311.040 11098.15695 4405.500 280.0 39098.0
sales   50 6494.829  8103.01250 2426.406 189.0 33181.4
medhome 50   57.092    18.37312   53.900  34.5   120.0
```

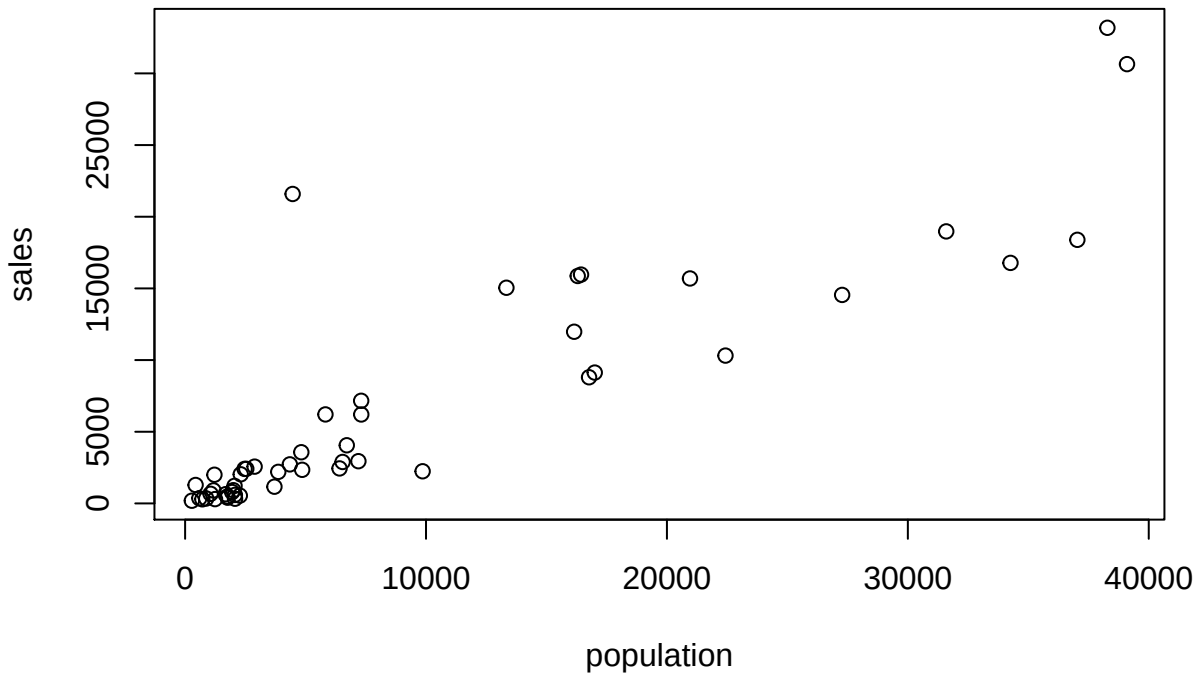
Show Overhead C Details. Visualizing skewed distributions

```
par(mfrow = c(1, 2))
hist(Lot$pop, main = "", xlab = "population")
hist(Lot$sales, main = "", xlab = "sales")
```



Show Overhead D Details. Visualizing relationships with a scatter plot

```
plot(Lot$pop, Lot$sales, xlab = "population", ylab = "sales")
```



Show Overhead E Details. Correlation coefficient

```
cor(Lot$pop, Lot$sales)
```

```
[1] 0.8862827
```

2.1.2 Exercise. Correlations and the Wisconsin lottery

Assignment Text

The Wisconsin lottery dataset, `Wisc_lottery`,has already been read into a dataframe `Lot`.

Like insurance, lotteries are uncertain events and so the skills to work with and interpret lottery data are readily applicable to insurance. It is common to report sales and population in thousands of units, so this exercise gives you practice in rescaling data via linear transformations.

Instructions

- From the available population and sales variables, create new variables in the dataframe `Lot`, `pop_1000` and `sales_1000` that are in thousands (of people and of dollars, respectively).
- Create summary statistics for the dataframe that includes these new variables.
- Plot `pop_1000` versus `sales_1000`.
- Calculate the correlation between `pop_1000` versus `sales_1000` using the function cor(). How does this differ between the correlation between population and sales in the original units?

Hint. Use the dataframe to refer to pop and sales as `Lot$pop` and `Lot$sales`, respectively

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNMb3QgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXFdpc

2.2 Method of least squares

In this section, you learn how to:

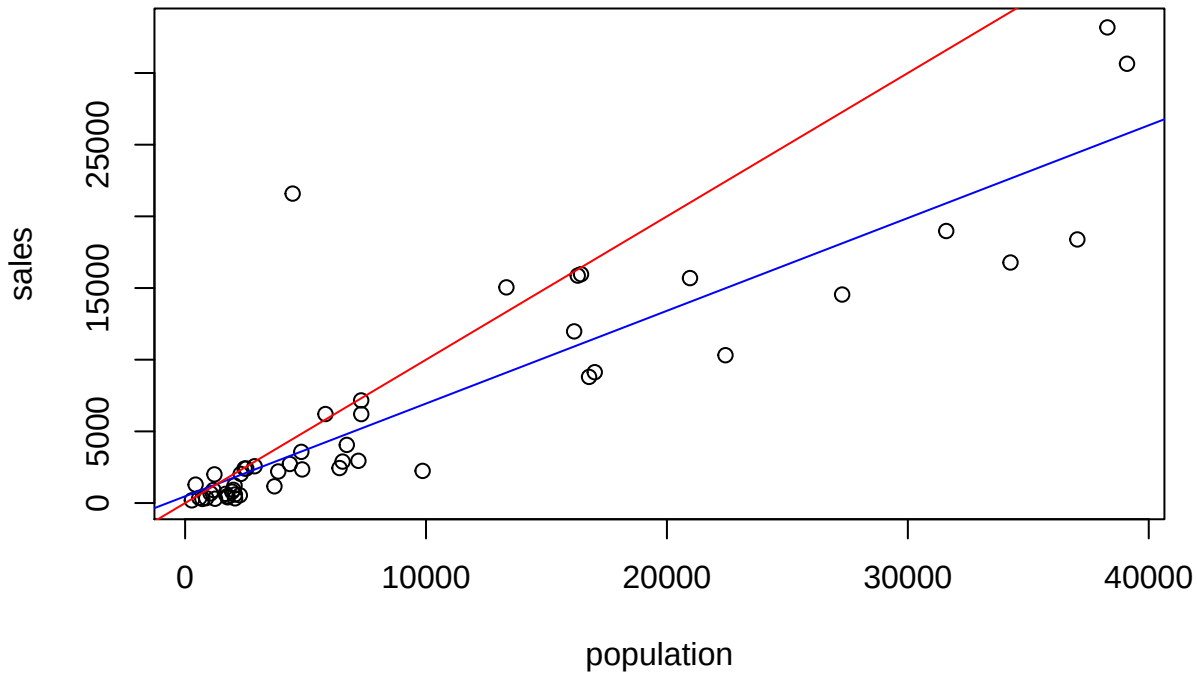
- Fit a line to data using the method of least squares
- Predict an observation using a least squares fitted line

2.2.1 Video

Video Overheads

Show Overhead A Details. Where to fit the line?

```
model_blr <- lm(sales ~ pop, data = Lot)
plot(Lot$pop, Lot$sales,xlab = "population", ylab = "sales")
abline(model_blr, col="blue")
abline(0,1, col="red")
```



Show Overhead B Details. Method of least squares

- For observation $\{(y, x)\}$, the height of the regression line is

$$b_0 + b_1 x.$$

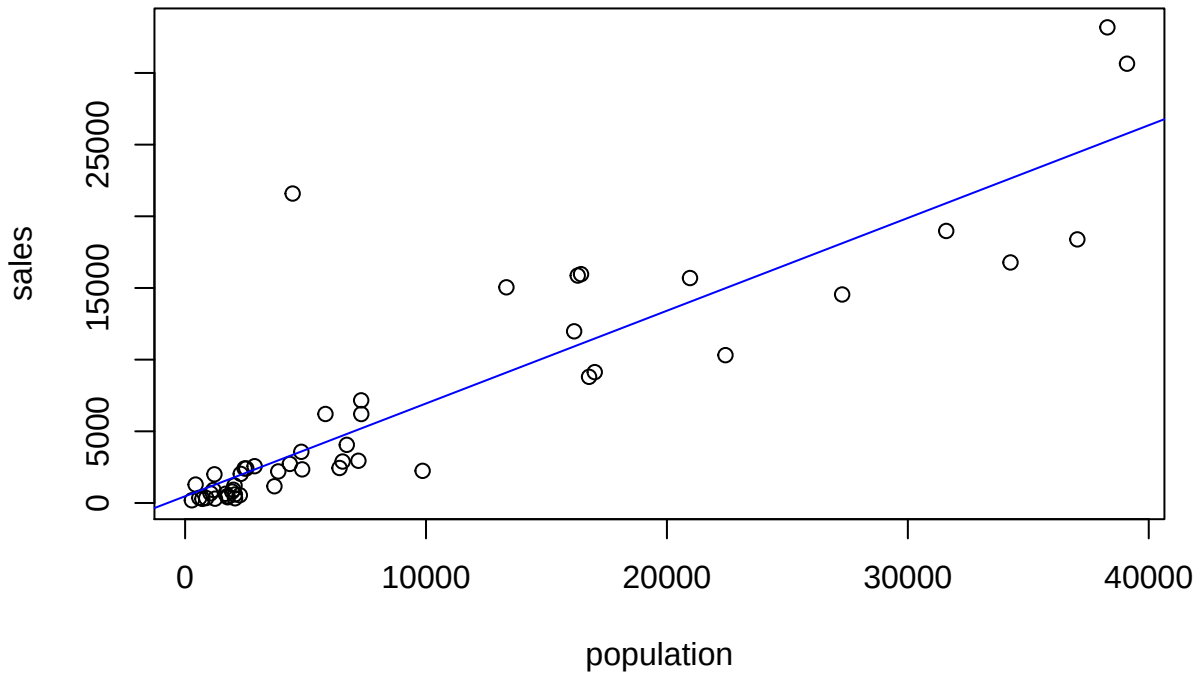
- Thus, $y - (b_0 + b_1 x)$ represents the deviation.
- The sum of squared deviations is

$$SS(b_0, b_1) = \sum (y - (b_0 + b_1 x))^2.$$

- The *method of least squares* – determine values of b_0, b_1 that minimize SS.

Show Overhead C Details. Regression coefficients

```
model_blr <- lm(sales ~ pop, data = Lot)
round(coefficients(model_blr), digits=4)
plot(Lot$pop, Lot$sales,xlab = "population", ylab = "sales")
abline(model_blr, col="blue")
```




```
(Intercept)         pop
   469.7036      0.6471
```

Show Overhead D Details. Prediction

```
round(coefficients(model_blr), digits=6)
coefficients(model_blr)[1] + coefficients(model_blr)[2]*30000

newdata <- data.frame(pop = 30000)
predict(model_blr, newdata)
```

```
(Intercept)         pop
 469.703598    0.647095
```

```
(Intercept)
   19882.55
          1
19882.55
```

2.2.2 Exercise. Least squares fit using housing prices

Assignment Text

The prior video analyzed the effect that a zip code's population has on lottery sales. Instead of population, suppose that you wish to understand the effect that housing prices have on the sale of lottery tickets. The dataframe `Lot`, read in from the Wisconsin lottery dataset `Wisc_lottery`, contains the variable `medhome` which is the median house price for each zip code, in thousands of dollars. In this exercise, you will get a feel for the distribution of this variable by examining summary statistics, examine its relationship with sales graphically and via correlations, fit a basic linear regression model and use this model to predict sales.

Instructions

- Summarize the dataframe `Lot` that contains `medhome` and `sales`.
- Plot `medhome` versus `sales`. Summarize this relationship by calculating the corresponding correlation coefficient using the function cor().
- Using the function lm(), regress `medhome`, the explanatory variable, on `sales`, the outcome variable. Display the regression coefficients to four significant digits.
- Use the function predict() and the fitted regression model to predict sales assuming that the median house price for a zip code is 50 (in thousands of dollars).

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNMb3QgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXFdpc

2.3 Understanding variability

In this section, you learn how to:

- Visualize the ANOVA decomposition of variability
- Calculate and interpret R^2, the coefficient of determination
- Calculate and interpret s^2 the mean square error
- Explain the components of the ANOVA table

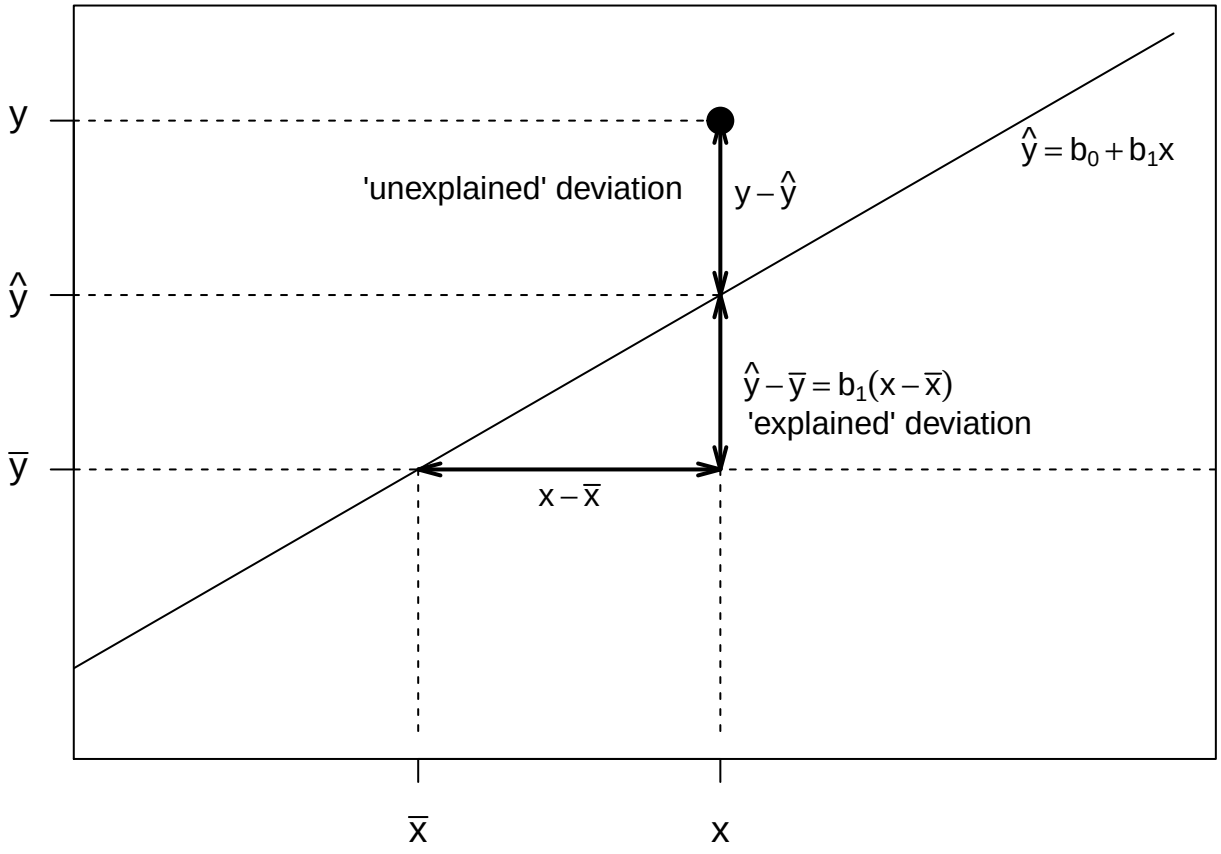
2.3.1 Video

Video Overhead Details

Show OverheadS A and B Details. Visualizing the uncertainty about a line

```
par(mar=c(2.2,2.1,.2,.2),cex=1.2)
x <- seq(-4, 4, len=101)
y <- x
plot(x, y, type = "l", xlim=c(-3, 4), xaxt="n", yaxt="n", xlab="", ylab="")
axis(1, at = c(-1, 1),lab = expression(bar(x), x))
axis(2, at = c(-1, 1, 3),lab = expression(bar(y), hat(y), y), las=1)
abline(-1, 0, lty = 2)
```

```
segments(-4, 1, 1, 1, lty=2)
segments(-4, 3, 1, 3, lty = 2)
segments(1, -4, 1, 3, lty = 2)
segments(-1, -4, -1, -1, lty = 2)

points(1, 3, cex=1.5, pch=19)

arrows(1.0, 1, 1.0, 3, code = 3, lty = 1, angle=15, length=0.12, lwd=2)
text(1.3, 2.2,   expression( y-hat(y)),cex=0.8)
text(-.3,2.2,"'unexplained' deviation", cex=.8)
arrows(1.0, -1, 1.0, 1, code = 3, lty = 1, angle=15, length=0.12, lwd=2)
text(1.85, 0, expression(hat(y)-bar(y) == b[1](x-bar(x)) ), cex=0.8 )
text(2.1, -0.5, " 'explained' deviation", cex=0.8)
arrows(-1, -1.0, 1, -1.0, code = 3, lty = 1, angle=15, length=0.12, lwd = 2)
text(0, -1.3, expression( x-bar(x)), cex=0.8  )
text(3.5, 2.7, expression( hat(y)== b[0]+ b[1]*x), cex=0.8  )
```



Show Overheads C, D and E Details. ANOVA Table

```
model_blr <- lm(sales ~ pop, data = Lot)
anova(model_blr)
sqrt(anova(model_blr)$Mean[2])
summary(model_blr)$r.squared
```

```
Analysis of Variance Table
```

```
Response: sales
          Df     Sum Sq    Mean Sq F value    Pr(>F)
pop        1 2527165015 2527165015  175.77 < 2.2e-16 ***
Residuals 48  690116755   14377432
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
[1] 3791.758
[1] 0.7854969
```

2.3.2 Exercise. Summarizing measures of uncertainty

Assignment Text

In a previous exercise, you developed a regression line to fit the variable `medhome`, the median house price for each zip code, as a predictor of lottery sales. The regression of `medhome` on `sales` has been summarized in the `R` object `model_blr`.

How reliable is the regression line? In this excercise, you will compute some of the standard measures that are used to summarize the goodness of this fit.

Instructions

- Summarize the fitted regression model in an ANOVA table.
- Determine the size of the typical residual, s.
- Determine the coefficient of determination, R^2.

Hint. Learn more about possibilities through the `Rdocumentation` site. If you have not done so already, check out the function anova()

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNMb3QgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXFdpc2

2.3.3 Exercise. Effects of linear transforms on measures of uncertainty

Assignment Text

Let us see how rescaling, a linear transformation, affects our measures of uncertainty. As before, the Wisconsin lottery dataset `Wisc_lottery` has been read into a dataframe `Lot` that also contains `sales_1000`, sales in thousands of dollars, and `pop_1000`, zip code population in thousands. How do measures of uncertainty change when going from the original units to thousands of those units?

Instructions

- Run a regression of `pop` on `sales_1000` and summarize this in an ANOVA table.
- For this regression, determine the s and the coefficient of determination, R^2.

- Run a regression of `pop_1000` on `sales_1000` and summarize this in an ANOVA table.
- For this regression, determine the s and the coefficient of determination, R^2.

Hint. The residual standard error is also available as `summary(model_blr1)$sigma`. The coefficient of determination is also available as `summary(model_blr1)$r.squared`.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNMb3QgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXFdpc2

2.4 Statistical inference

In this section, you learn how to:

- Conduct a hypothesis test for a regression coefficient using either a rejection/acceptance procedure or a p-value
- Calculate and interpret a confidence interval for a regression coefficient
- Calculate and interpret a prediction interval at a specific value of a predictor variable

---

2.4.1 Video

Video Overhead Details

Show Overhead A Details. Summary of basic linear regression model

Introduce the output in the *summary* of the basic linear regression model.

```
Lot <- read.csv("CSVData\\Wisc_lottery.csv", header = TRUE)
#Lot <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/a792b30fb32b0896dd6
#options(scipen = 8, digits = 4)
model_blr <- lm(sales ~ pop, data = Lot)
summary(model_blr)
```

```
Call:
lm(formula = sales ~ pop, data = Lot)

Residuals:
    Min      1Q  Median      3Q     Max
-6046.7 -1460.9  -670.5   485.6 18229.5

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 469.70360  702.90619   0.668    0.507
pop           0.64709    0.04881  13.258   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3792 on 48 degrees of freedom
Multiple R-squared:  0.7855,    Adjusted R-squared:  0.781
F-statistic: 175.8 on 1 and 48 DF,  p-value: < 2.2e-16
```

Show Overhead B Details. Hypothesis testing

```
> summary(model_blr)$coefficients
            Estimate Std. Error t value  Pr(>|t|)
(Intercept) 469.7036  702.90619  0.6682 5.072e-01
pop           0.6471    0.04881 13.2579 1.158e-17
```

Show Overhead C Details. Confidence intervals

```
confint(model_blr, level = .90)
confint(model_blr, level = .95)
```

```
                    5 %         95 %
(Intercept) -709.2276710 1648.6348666
pop            0.5652327    0.7289569
                  2.5 %     97.5 %
(Intercept) -943.5840183 1882.99121
pop            0.5489596    0.74523
```

Show Overhead D Details. Confidence intervals check

```
# Just for checking
summary(model_blr)$coefficients[2,1]
summary(model_blr)$coefficients[2,2]
qt(.975, 48)

summary(model_blr)$coefficients[2,1] -
  summary(model_blr)$coefficients[2,2]*qt(.975, 48)

confint(model_blr, level = .95)
confint(model_blr, level = .95)
```

```
[1] 0.6470948
[1] 0.04880808
[1] 2.010635
[1] 0.5489596
                  2.5 %     97.5 %
(Intercept) -943.5840183 1882.99121
pop            0.5489596    0.74523
                  2.5 %     97.5 %
(Intercept) -943.5840183 1882.99121
pop            0.5489596    0.74523
```

Show Overhead E Details. Prediction intervals

```
NewData <- data.frame(pop = 10000)
predict(model_blr, NewData, interval = "prediction", level = .90)
predict(model_blr, NewData, interval = "prediction", level = .99)
```

```
       fit      lwr      upr
1 6940.651 517.4933 13363.81
       fit       lwr      upr
1 6940.651 -3331.214 17212.52
```

2.4.2 Exercise. Statistical inference and Wisconsin lottery

Assignment Text

In a previous exercise, you developed a regression line with the variable `medhome`, the median house price for each zip code, as a predictor of lottery sales. The regression of `medhome` on `sales` has been summarized in the `R` object `model_blr`.

This exercise allows you to practice the standard inferential tasks: hypothesis testing, confidence intervals, and prediction.

Instructions

- Summarize the regression model and identify the t-statistic for testing the importance of the regression coefficient associated with `medhome`.
- Use the function confint() to provide a 95% confidence interval for the regression coefficient associated with `medhome`.
- Consider a zip code with a median housing price equal to 50 (in thousands of dollars). Use the function predict() to provide a point prediction and a 95% prediction interval for sales.

Hint. Taking a [summary()] of a regression object produces a new objeect. You can use the [str()] structure command to learn more about the new object. Try out a command such as `str(summary(model_blr))`

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNMb3QgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXFdpc

2.5 Diagnostics

In this section, you learn how to:

- Describe how diagnostic checking and residual analysis are used in a statistical analysis
- Describe several model misspecifications commonly encountered in a regression analysis

---

2.5.1 Video

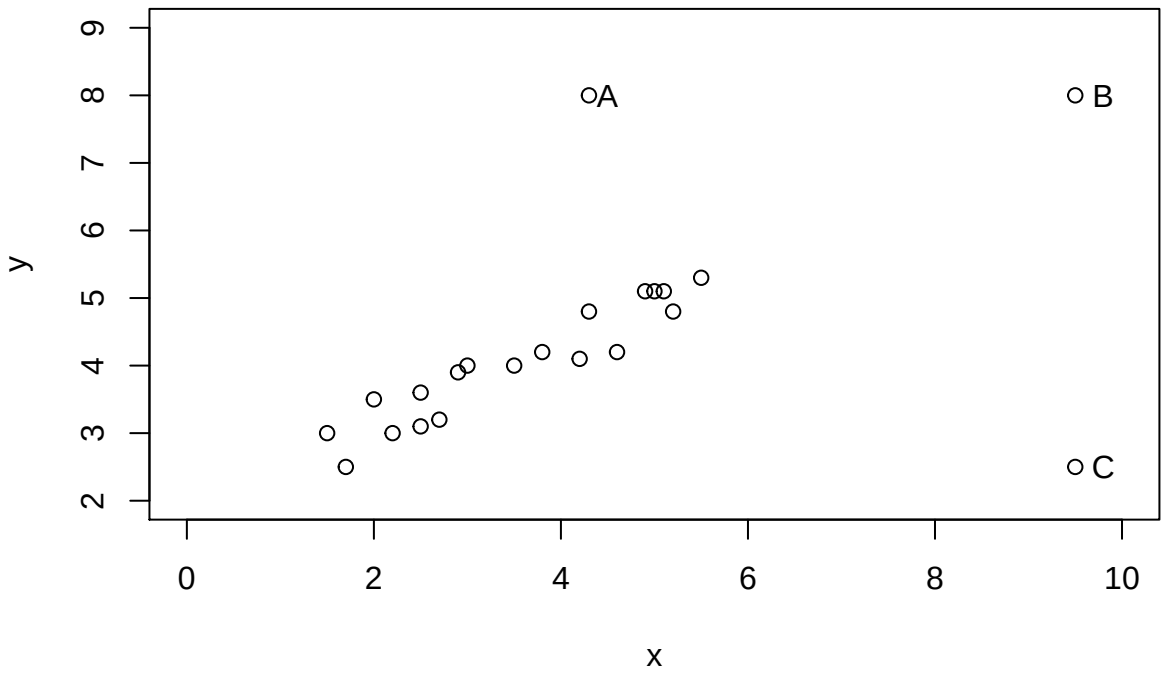
Video Overhead Details

Show Overhead A Details. Unusual observations in regression

- We have defined regression estimates as minimizers of a least squares objective function.
- An appealing intuitive feature of linear regressions is that regression estimates can be expressed as weighted averages of outcomes.
- The weights vary by observation, some observations are more important than others.
- "Unusual" observations are far from the majority of the data set:
- Unusual in the vertical direction is called an *outlier*.
- Unusual in the horizontal directional is called a *high leverage point*.

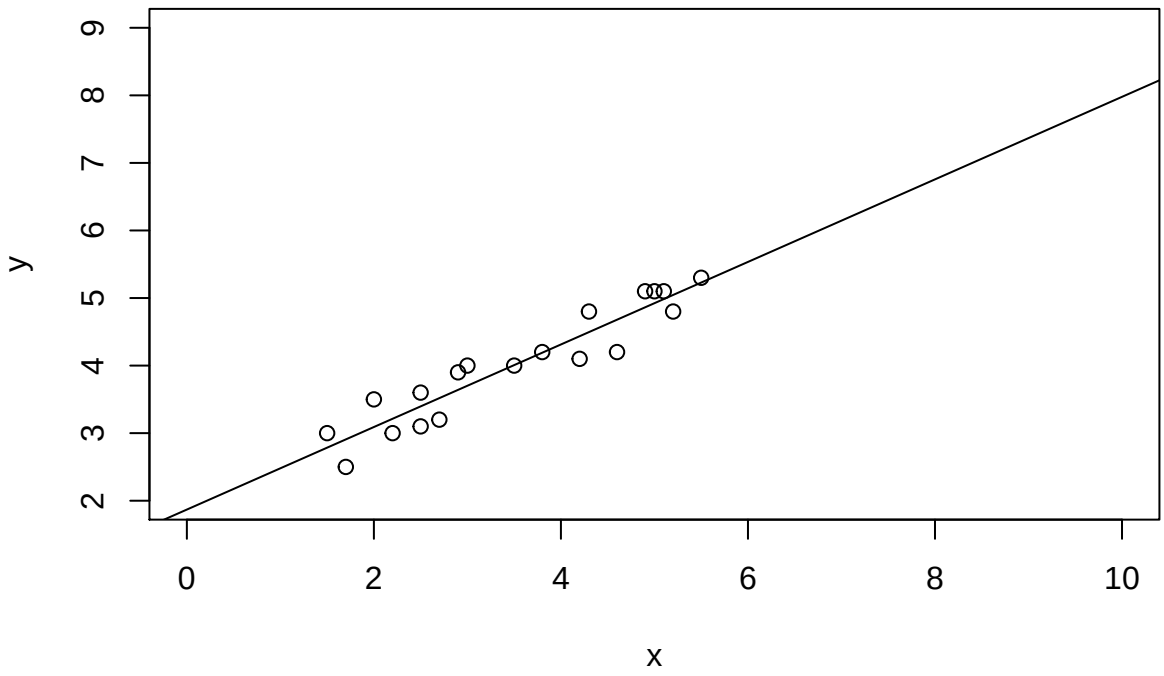
Show Overhead B Details. Example. Outliers and High Leverage Points

```
outlr <- read.csv("CSVData\\Outlier.csv", header = TRUE)

#  FIGURE 2.7
plot(outlr$x, outlr$y, xlim = c(0, 10), ylim = c(2, 9), xlab = "x", ylab = "y")
text(4.5, 8.0, "A")
text(9.8, 8.0, "B")
text(9.8, 2.5, "C")
```



Show Overhead C Details. Regression fit with 19 base observations

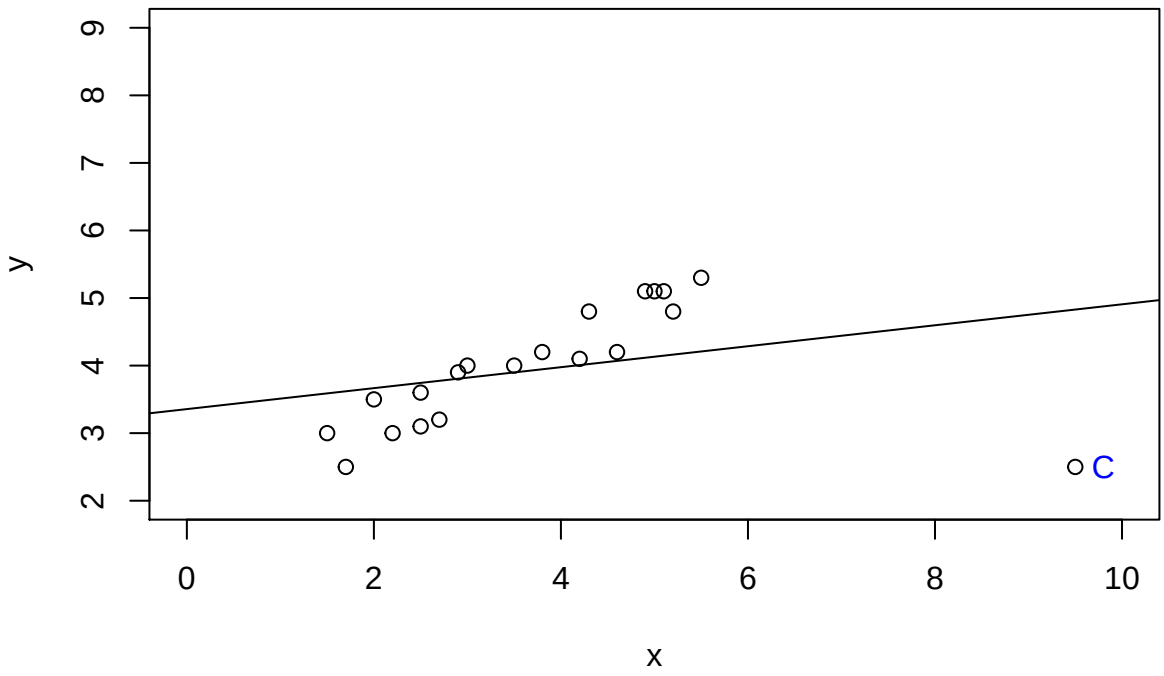
```
model_outlr0 <- lm(y ~ x, data = outlr, subset = -c(20,21,22))
summary(model_outlr0)
plot(outlr$x[1:19], outlr$y[1:19], xlab = "x", ylab = "y", xlim = c(0, 10), ylim = c(2, 9))
abline(model_outlr0)
```




```
Call:
lm(formula = y ~ x, data = outlr, subset = -c(20, 21, 22))

Residuals:
    Min      1Q  Median      3Q     Max
-0.4790 -0.2708  0.0711  0.2263  0.4094

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.86874    0.19583   9.543 3.06e-08 ***
x            0.61094    0.05219  11.705 1.47e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2883 on 17 degrees of freedom
Multiple R-squared:  0.8896,    Adjusted R-squared:  0.8831
F-statistic:   137 on 1 and 17 DF,  p-value: 1.471e-09
```

Show Overhead D Details. Regression fit with 19 base observations plus C

```
model_outlrC <- lm(y ~ x, data = outlr, subset = -c(20,21))
summary(model_outlrC)
plot(outlr$x[c(1:19,22)], outlr$y[c(1:19,22)], xlab = "x", ylab = "y", xlim = c(0, 10), ylim = c(2, 9))
text(9.8, 2.5, "C", col = "blue")
abline(model_outlrC)
```




```
Call:
lm(formula = y ~ x, data = outlr, subset = -c(20, 21))

Residuals:
     Min       1Q   Median       3Q      Max
-2.32947 -0.57819  0.09772  0.67240  1.09097

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.3559     0.4560   7.360 7.87e-07 ***
x             0.1551     0.1078   1.439    0.167
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8648 on 18 degrees of freedom
Multiple R-squared:  0.1031,    Adjusted R-squared:  0.0533
F-statistic:  2.07 on 1 and 18 DF,  p-value: 0.1674
```

Show Overhead E Details. R code

```
model_outlr0 <- lm(y ~ x, data = outlr, subset = -c(20,21,22))
summary(model_outlr0)
model_outlrA <- lm(y ~ x, data = outlr, subset = -c(21,22))
summary(model_outlrA)
model_outlrB <- lm(y ~ x, data = outlr, subset = -c(20,22))
```

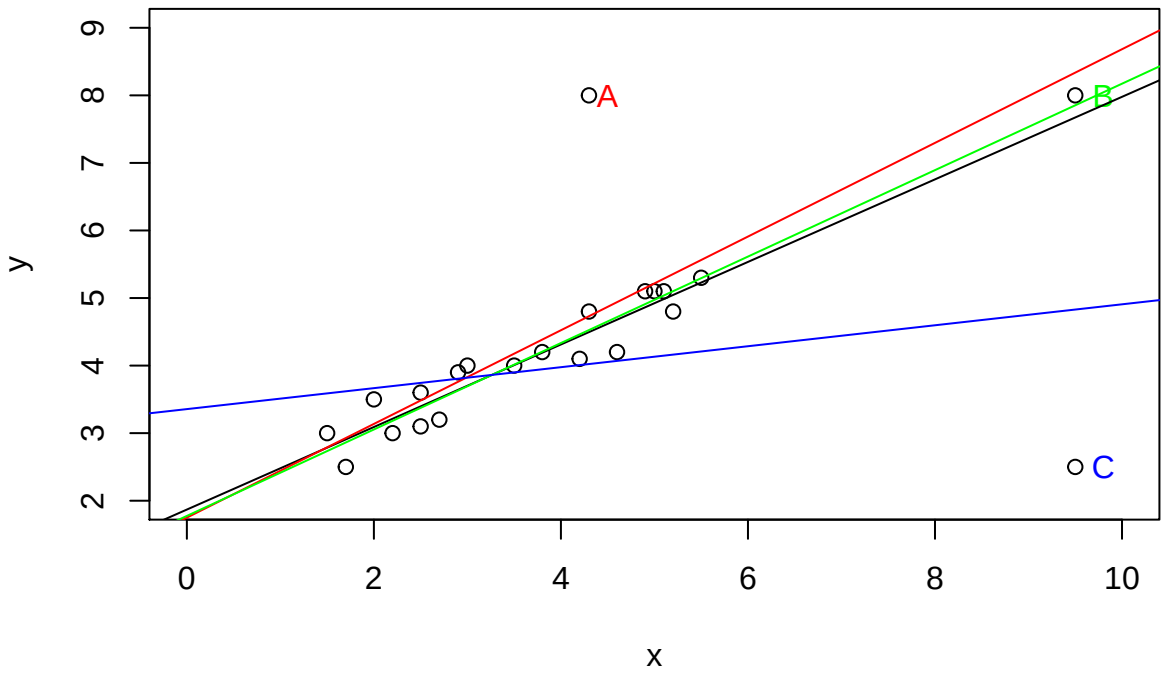
```r
summary(model_outlrB)
model_outlrC <- lm(y ~ x, data = outlr, subset = -c(20,21))
summary(model_outlrC)
```

```
Call:
lm(formula = y ~ x, data = outlr, subset = -c(20, 21, 22))

Residuals:
    Min      1Q  Median      3Q     Max
-0.4790 -0.2708  0.0711  0.2263  0.4094

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.86874    0.19583   9.543 3.06e-08 ***
x            0.61094    0.05219  11.705 1.47e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2883 on 17 degrees of freedom
Multiple R-squared:  0.8896,    Adjusted R-squared:  0.8831
F-statistic:   137 on 1 and 17 DF,  p-value: 1.471e-09


Call:
lm(formula = y ~ x, data = outlr, subset = -c(21, 22))

Residuals:
    Min      1Q  Median      3Q     Max
-0.7391 -0.3928 -0.1805  0.1225  3.2689

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   1.7500     0.5736   3.051 0.006883 **
x             0.6933     0.1517   4.570 0.000237 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8455 on 18 degrees of freedom
Multiple R-squared:  0.5371,    Adjusted R-squared:  0.5114
F-statistic: 20.89 on 1 and 18 DF,  p-value: 0.0002374


Call:
lm(formula = y ~ x, data = outlr, subset = -c(20, 22))

Residuals:
     Min       1Q   Median       3Q      Max
-0.51763 -0.28094  0.03452  0.23586  0.44581

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.77463    0.15020   11.81 6.48e-10 ***
x            0.63978    0.03551   18.02 5.81e-13 ***
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2849 on 18 degrees of freedom
Multiple R-squared:  0.9474,    Adjusted R-squared:  0.9445
F-statistic: 324.5 on 1 and 18 DF,  p-value: 5.808e-13


Call:
lm(formula = y ~ x, data = outlr, subset = -c(20, 21))

Residuals:
     Min       1Q   Median       3Q      Max
-2.32947 -0.57819  0.09772  0.67240  1.09097

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3.3559     0.4560   7.360 7.87e-07 ***
x             0.1551     0.1078   1.439    0.167
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8648 on 18 degrees of freedom
Multiple R-squared:  0.1031,    Adjusted R-squared:  0.0533
F-statistic:  2.07 on 1 and 18 DF,  p-value: 0.1674
```

Show Overhead F Details. Visualizing four regression fits

```
plot(outlr$x, outlr$y, xlim = c(0, 10), ylim = c(2, 9), xlab = "x", ylab = "y")
text(4.5, 8.0, "A", col = "red")
text(9.8, 8.0, "B", col = "green")
text(9.8, 2.5, "C", col = "blue")
abline(model_outlr0)
abline(model_outlrA, col = "red")
abline(model_outlrB, col = "green")
abline(model_outlrC, col = "blue")
```



Show Overhead G Details. Results from four regression models

Results from Four Regressions

Data	b_0	b_1	s	$R^2(\%)$	$t(b_1)$
19 Base Points	1.869	0.611	0.288	89.0	11.71
19 Base Points + A	1.750	0.693	0.846	53.7	4.57
19 Base Points + B	1.775	0.640	0.285	94.7	18.01
19 Base Points + C	3.356	0.155	0.865	10.3	1.44

2.5.2 Exercise. Assessing outliers in lottery sales

Assignment Text

In an earlier video, we made a scatter plot of population versus sales. This plot exhibits an outlier; the point in the upper left-hand side of the plot represents a zip code that includes Kenosha, Wisconsin. Sales for this zip code are unusually high given its population.

This exercise summarizes the regression fit both with and without this zip code in order to see how robust our results are to the inclusion of this unusual observation.

Instructions

- A basic linear regression fit of population on sales has already been fit in the object `model_blr`. Re-fit this same model to the data, this time omitting Kenosha (observation number 9).
- Plot these two least squares fitted lines superimposed on the full data set.
- What is the effect on the distribution of residuals by removing this point? Calculate a normal qq plot with and without Kenosha.

Hint. You can extract the residuals from a regression object with the function [residuals()].

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNMb3QgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXFdpc

Chapter 3

Multiple Linear Regression

Chapter description

This chapter introduces linear regression in the case of several explanatory variables, known as multiple linear regression (**MLR**). Many basic linear regression concepts extend directly, including goodness of fit measures such as the coefficient of determination and inference using t-statistics. Multiple linear regression models provide a framework for summarizing highly complex, multivariate data. Because this framework requires only linearity in the parameters, we are able to fit models that are nonlinear functions of the explanatory variables, thus providing a wide scope of potential applications.

Term Life Data

Video Overhead Details

Show Overhead A Details. Demand for term life insurance

"Who buys insurance and how much do they buy?"

- Companies have data on current customers
- How do get info on potential (new) customers?

To understand demand, consider the Survey of Consumer Finances (*SCF*)

- This is a nationally representative sample that contains extensive information on potential U.S. customers.
- We study a random sample of 500 of the 4,519 households with positive income that were interviewed in the 2004 survey.
- We now focus on $n = 275$ households that purchased term life insurance

Show Overhead B Details. Term life insurance summary statistics

We study $y = face$, the amount that the company will pay in the event of the death of the named insured.

We focus on $k = 3$ explanatory variables - annual *income*, - the number of years of *education* of the survey respondent and - the number of household members, *numhh*.

The data suggest that *income* and *face* are skewed so we also introduce logarithmic versions.

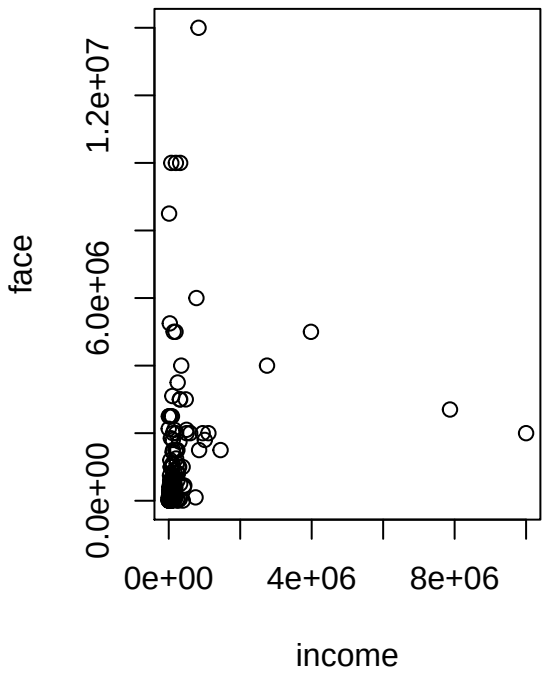
Show Overhead C Details. Summary statistics

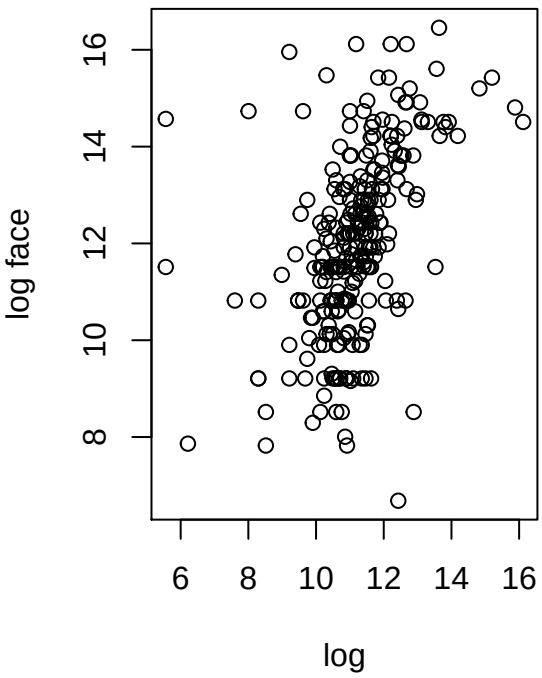
```
#Term <- read.csv("CSVData\\term_life.csv", header = TRUE)
Term <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/efc64bc2d78cf6b48ad1
#  PICK THE SUBSET OF THE DATA CORRESPONDING TO TERM PURCHASE
Term1 <- subset(Term, subset = face > 0)
str(Term1)
head(Term1)

library(psych)
Term2 <- Term1[, c("education", "face", "income", "logface", "logincome", "numhh")]
#options(scipen = 100, digits = 4)
head(Term2)
describe(Term2)[,c(3,4,8,5,9,2)]
```

```
'data.frame':   275 obs. of  7 variables:
 $ education: int  16 9 16 17 11 16 17 16 14 12 ...
 $ face     : int  20000 130000 1500000 50000 220000 600000 100000 2500000 250000 50000 ...
 $ income   : int  43000 12000 120000 40000 28000 100000 112000 15000 32000 25000 ...
 $ logface  : num  9.9 11.8 14.2 10.8 12.3 ...
 $ logincome: num  10.67 9.39 11.7 10.6 10.24 ...
 $ numhh    : int  3 3 5 4 4 3 2 4 1 2 ...
 $ marstat  : int  1 1 1 1 2 1 1 1 0 1 ...
  education    face income   logface logincome numhh marstat
1        16   20000  43000  9.903488 10.668955     3       1
2         9  130000  12000 11.775290  9.392662     3       1
3        16 1500000 120000 14.220976 11.695247     5       1
4        17   50000  40000 10.819778 10.596635     4       1
6        11  220000  28000 12.301383 10.239960     4       2
8        16  600000 100000 13.304685 11.512925     3       1
  education    face income   logface logincome numhh
1        16   20000  43000  9.903488 10.668955     3
2         9  130000  12000 11.775290  9.392662     3
3        16 1500000 120000 14.220976 11.695247     5
4        17   50000  40000 10.819778 10.596635     4
6        11  220000  28000 12.301383 10.239960     4
8        16  600000 100000 13.304685 11.512925     3
               mean         sd    min    median       max   n
education     14.52       2.55   2.00     16.00 1.700e+01 275
face      747581.45 1674362.43 800.00 150000.00 1.400e+07 275
income    208974.62  824009.77 260.00  65000.00 1.000e+07 275
logface       11.99       1.87   6.68     11.92 1.645e+01 275
logincome     11.15       1.30   5.56     11.08 1.612e+01 275
numhh          2.96       1.49   1.00      3.00 9.000e+00 275
```

Show Overhead D Details. Scatter plots of income versus face in original and logarithmic units

```
par(mfrow = c(1, 2))
plot(Term2$income, Term2$face, xlab = "income", ylab = "face")
plot(Term2$logincome, Term2$logface, xlab = "log", ylab = "log face")
```





3.1 Method of least squares

In this section, you learn how to:

- Interpret correlation coefficients by visualizing a scatterplot matrix
- Fit a plane to data using the method of least squares
- Predict an observation using a least squares fitted plane

3.1.1 Video

Video Overhead Details

Show Overhead A Details. Correlation table

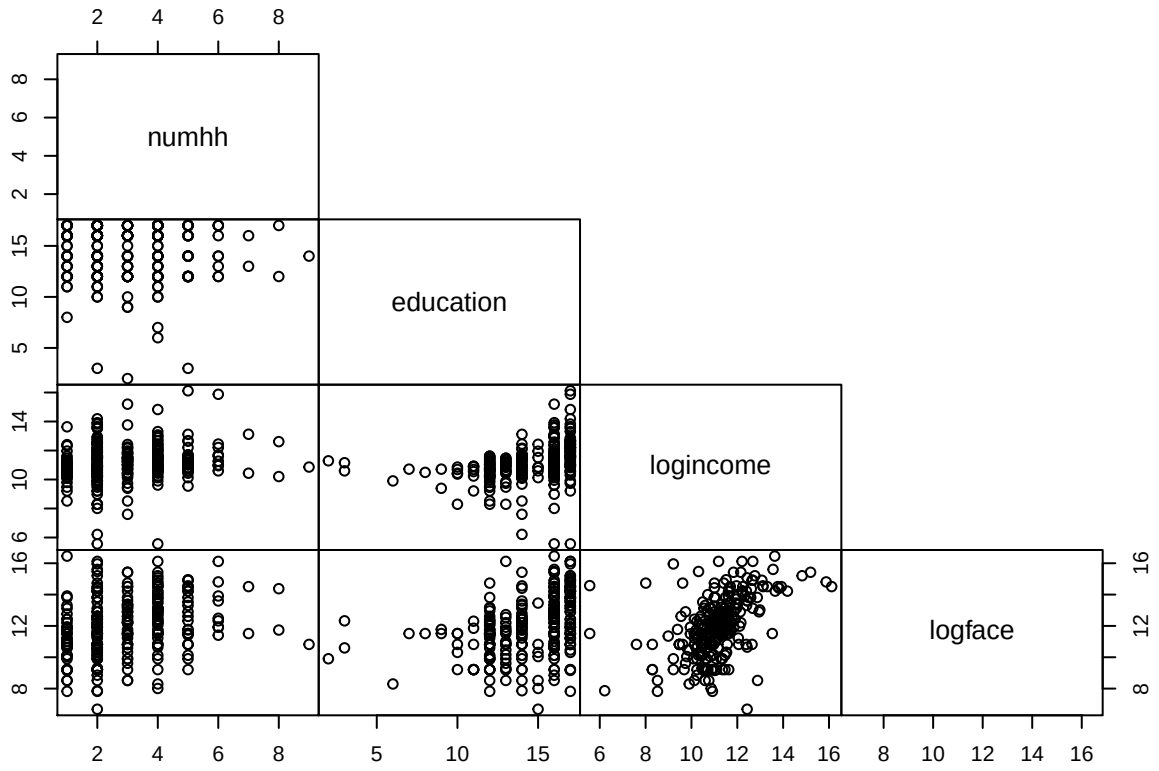
```
round(cor(Term2), digits=3)
```

```
           education  face income logface logincome  numhh
education      1.000 0.244  0.163   0.383     0.343 -0.064
face           0.244 1.000  0.217   0.656     0.323  0.107
income         0.163 0.217  1.000   0.251     0.518  0.142
logface        0.383 0.656  0.251   1.000     0.482  0.288
```

```
logincome      0.343 0.323  0.518   0.482      1.000  0.179
numhh         -0.064 0.107  0.142   0.288      0.179  1.000
```

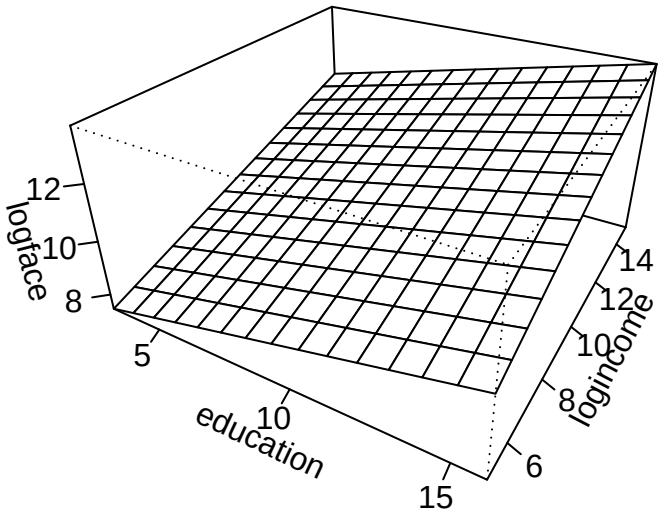
Show Overhead B Details. Scatterplot matrix

```
Term3 <- Term1[,c("numhh", "education", "logincome", "logface")]
pairs(Term3, upper.panel = NULL, gap = 0, cex.labels = 1.25)
```



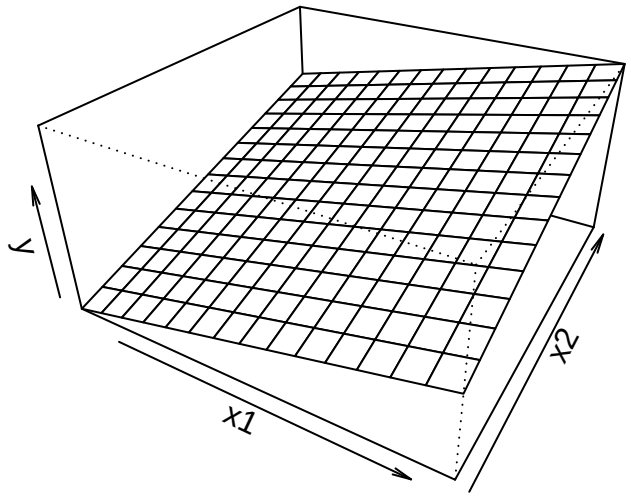
Show Overhead C Details. Visualizing a regression plane

```
education <- seq(3, 16, length = 15)
logincome <- seq(5, 15, length = 15)
f <- function(education,logincome){
  r <- 5 + 0.221*education + 0.354*logincome
}
logface <- outer(education, logincome, f)
persp(education, logincome, logface, theta = 30,
      phi = 30, expand = 0.5, ticktype = "detailed")
```




```
rm(education,logincome,logface)
```

```
education <- seq(3, 16, length = 15)
logincome <- seq(5, 15, length = 15)
f <- function(education,logincome){
  r <- 5 + 0.221*education + 0.354*logincome
}
logface <- outer(education, logincome, f)
persp(education, logincome, logface, theta = 30,
      phi = 30, expand = 0.5, ticktype = "simple", #ticktype = "detailed", #
      xlab = "x1", ylab="x2",zlab="y", nticks = 1)
```




```
rm(education,logincome,logface)
```

Show Overhead D Details. Method of least squares

- For observation $\{(y, x_1, \ldots, x_k)\}$, the height of the regression plane is

$$b_0 + b_1 x_1 + \cdots + b_k x_k.$$

- Thus, $y - (b_0 + b_1 x_1 + \cdots + b_k x_k)$ represents the deviation.
- The sum of squared deviations is

$$SS(b_0, \ldots, b_k) = \sum (y - (b_0 + b_1 x_1 + \cdots + b_k x_k))^2.$$

- The *method of least squares* – determine values of $b_0, \ldots, b_k$ that minimize SS.

Show Overhead E Details. Fit a multiple linear regression model

```
Term_mlr <- lm(logface ~ education + numhh + logincome, data = Term2)
round(coefficients(Term_mlr), digits=4)
newdata <- data.frame(logincome = log(60000), education = 12, numhh = 3)
exp(predict(Term_mlr, newdata))
```

```
(Intercept)   education       numhh   logincome
     2.5841      0.2064      0.3060      0.4935
       1
90135.86
```

3.1.2 Exercise. Least squares and term life data

Assignment Text

The prior video introduced the *Survey of Consumer Finances* (SCF) term life data. A subset consisting of only those who purchased term life insurance, has already been read into a dataframe `Term2`.

Suppose that you wish to predict the amount of term life insurance that someone will purchase but are uneasy about the `education` variable. The SCF `education` variable is the number of completed years of schooling and so 12 corresponds to completing high school in the US. Your sense is that, for purposes of purchasing life insurance, high school graduates and those that attend college should be treated the same. So, in this exercise, your will create a new variable, `education1`, that is equal to years of education for those with education less than or equal to 12 and is equal to 12 otherwise.

Instructions

- Use the pmin() function to create the `education1` variable as part of the `Term2` dataframe.
- Check your work by examining summary statistics for the revised `Term2` dataframe.
- Examine correlations for the revised dataframe.
- Using the method of least squares and the function lm(), fit a MLR model using `logface` as the dependent variables and using `education`, `numhh`, and `logincome` as explanatory variables.
- With this fitted model and the function predict(), predict the face amount of insurance that someone with income of 40,000, 11 years of education, and 4 people in the household would purchase.

Hint. Remember that your prediction is in log dollars so you need to exponentiate it to get the results in the original dollar units

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNUZXJtIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFx0Z

3.1.3 Exercise. Interpreting coefficients as proportional changes

Assignment Text

In a previous exercise, you fit a MLR model using `logface` as the outcome variable and using `education`, `numhh`, and `logincome` as explanatory variables; the resulting fit is in the object `Term_mlr`. For this fit, the coefficient associated with `education` is 0.2064. We now wish to interpret this regression coefficient.

The typical interpretation of coefficients in a regression model is as a partial slope. That is, for the coefficient b_1 associated with x_1, we interpret b_1 to be amount that the expected outcome changes per unit change in x_1, holding the other explanatory variables fixed.

For the term life example, the units of the outcome are in logarithmic dollars. So, for small values of b_1, we can interpret this to be a *proportional* change in dollars.

Instructions

- Determine least square fitted values for several selected values of `education`, holding other explantory variables fixed. For this part of the demonstration, we used their mean values.
- Determine the proportional changes. Note the relation between these values from a discrete change approximation to the regression coefficient for `education` equal to 0.2064.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNUZXJtIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFx0Z

3.1.4 Exercise. Interpreting coefficients as elasticities

Assignment Text

In a previous exercise, you fit a MLR model using `logface` as the outcome variable and using `education`, `numhh`, and `logincome` as explanatory variables; the resulting fit is in the object `Term_mlr`. From this fit, the coefficient associated with `logincome` is 0.4935. We now wish to interpret this regression coefficient.

The typical interpretation of coefficients in a regression model is as a partial slope. When both x_1 and y are in logarithmic units, then we can interpret b_1 to be ratio of two percentage changes, known as an *elasticity* in economics. Mathematically, we summarize this as

$$\frac{\partial \ln y}{\partial \ln x} = \left(\frac{\partial y}{y}\right) \; / \; \left(\frac{\partial x}{x}\right).$$

Instructions

- For several selected values of `logincome`, determine the corresponding proportional changes.
- Determine least square fitted values for several selected values of `logincome`, holding other explantory variables fixed.
- Determine the corresponding proportional changes for the fitted values.
- Calculate the ratio of proportional changes of fitted values to those for income. Note the relation between these values (from a discrete change approximation) to the regression coefficient for `logincome` equal to 0.4935.

Hint. When you calculate the ratio of proportional changes of fitted values to those for income, note the relation between these values (from a discrete change approximation) to the regression coefficient for `logincome` equal to 0.4935.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNUZXJtIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFx0Z

3.2 Statistical inference and multiple linear regresson

In this section, you learn how to:

- Explain mean square error and residual standard error in terms of degrees of freedom
- Develop an ANOVA table and use it to derive the coefficient of determination
- Calculate and interpret the coefficient of determination adjusted for degrees of freedom
- Conduct a test of a regression coefficient
- Summarize regression coefficients using point and interval estimators

---

3.2.1 Video

Video Overhead Details

Show Overhead A Details. Goodness of fit

Summarize

- deviations
- s^2
- R^2
- R_a^2
- ANOVA table

Show Overhead B Details. Goodness of fit and term life

```
Term_mlr <- lm(logface ~ education + numhh + logincome, data = Term2)
summary(Term_mlr)
anova(Term_mlr)
```

```

Call:
lm(formula = logface ~ education + numhh + logincome, data = Term2)

Residuals:
    Min      1Q  Median      3Q     Max
-5.7420 -0.8681  0.0549  0.9093  4.7187

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.58408    0.84643   3.053  0.00249 **
education    0.20641    0.03883   5.316 2.22e-07 ***
numhh        0.30605    0.06333   4.833 2.26e-06 ***
logincome    0.49353    0.07754   6.365 8.32e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.525 on 271 degrees of freedom
Multiple R-squared:  0.3425,    Adjusted R-squared:  0.3353
F-statistic: 47.07 on 3 and 271 DF,  p-value: < 2.2e-16

Analysis of Variance Table

Response: logface
           Df Sum Sq Mean Sq F value    Pr(>F)
education   1 140.55 140.549  60.417 1.601e-13 ***
numhh       1  93.68  93.681  40.270 9.251e-10 ***
logincome   1  94.24  94.238  40.510 8.316e-10 ***
Residuals 271 630.43   2.326
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Show Overhead C Details. Statistical inference

- hypothesis testing of a regression coefficient
- confidence intervals

Show Overhead D Details. Statistical inference and term life

```
Term_mlr <- lm(logface ~ education + numhh + logincome, data = Term2)
model_sum <- summary(Term_mlr)
model_sum$coefficients

round(confint(Term_mlr, level = .95), digits = 3)

round(confint(Term_mlr, level = .95), digits = 3)
```

```
             Estimate Std. Error  t value     Pr(>|t|)
(Intercept) 2.5840786 0.84642972 3.052916 2.491588e-03
```

```
education   0.2064139 0.03883186 5.315581 2.223619e-07
numhh       0.3060455 0.06332511 4.832926 2.255708e-06
logincome   0.4935323 0.07754198 6.364711 8.316097e-10
            2.5 % 97.5 %
(Intercept) 0.918  4.250
education   0.130  0.283
numhh       0.181  0.431
logincome   0.341  0.646
            2.5 % 97.5 %
(Intercept) 0.918  4.250
education   0.130  0.283
numhh       0.181  0.431
logincome   0.341  0.646
```

3.2.2 Exercise. Statistical inference and term life

Assignment Text

In later chapters, we will learn how to specify a model using diagnostics techniques; these techniques were used to specify face in log dollars for the outcome and similarly income in log dollars as an explanatory variable. Just to see how things work, in this exercise we will create new variables `face` and `income` that are in the original units and run a regression with these. We have already seen that rescaling by constants do not affect relationships but can be helpful with interpretations, so we define both `face` and `income` to be in thousands of dollars. A prior video introduced the term life dataframe `Term2`.

Instructions

- Create `Term2$face` by exponentiating `logface` and dividing by 1000. For convenience, we are storing this variable in the data set `Term2`. Use the same process to create `Term2$income`.
- Run a regression using `face` as the outcome variable and `education`, `numhh`, and `income` as explanatory variables.
- Summarize this model and identify the residual standard error (s) as well as the coefficient of determination (R^2) and the version adjusted for degrees of freedom (R^2_a).

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNUZXJtIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFx0Z

3.3 Binary variables

In this section, you learn how to:

- Interpret regression coefficients associated with binary variables
- Use binary variables and interaction terms to create regression models that are nonlinear in the covariates

---

3.3.1 Video

Video Overhead Details

Show Overhead A Details. Binary variables

- We can define a new variable

$$single = \begin{cases} 0 & \text{for other respondents} \\ 1 & \text{for single respondents} \end{cases}$$

- The variable *single* is said to be an *indicator*, or *dummy*, variable.

- To interpret coefficients, we now consider the regression function

$$\text{E } logface = \beta_0 + \beta_1 logincome + \beta_2 single$$

- This can be expressed as two lines

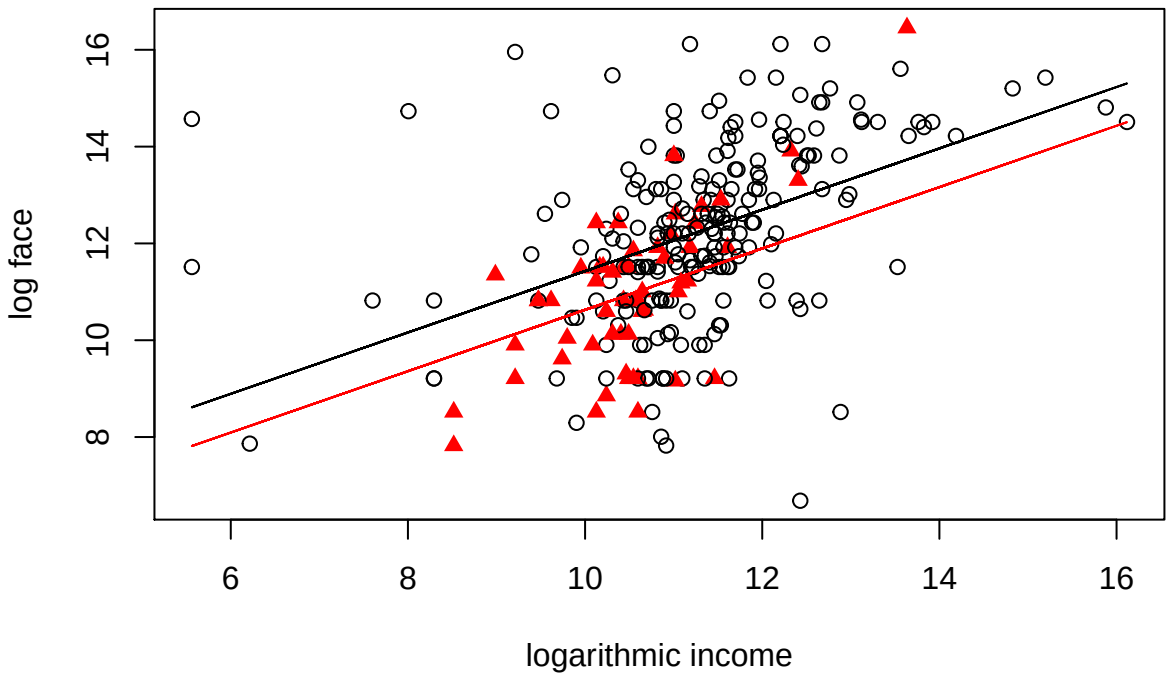
$$\text{E } logface = \begin{cases} \beta_0 + \beta_1 logincome & \text{for other respondents} \\ \beta_0 + \beta_2 + \beta_1 logincome & \text{for single respondents} \end{cases}.$$

- The least squares method of calculating the estimators, and the resulting theoretical properties, are the still valid when using binary variables.

Show Overhead B Details. Visualize effect of binary variables

Show Overhead C Details. R script for visualization

```
Term4 <- Term1[,c("numhh", "education", "logincome", "logface", "marstat")]
Term4$marstat <- as.factor(Term4$marstat)
table(Term4$marstat)
Term4$single <- 1*(Term4$marstat == 0)
model_single <- lm(logface ~ logincome + single, data = Term4)
summary(model_single)

plot(Term4$logincome,Term4$logface,xlab="logarithmic income", ylab="log face",
    pch= 1+16*Term4$single, col = c("red", "black", "black")[Term4$marstat])
Ey1 <- model_single$coefficients[1]+model_single$coefficients[2]*Term4$logincome
Ey2 <- Ey1 + model_single$coefficients[3]
lines(Term4$logincome, Ey1)
lines(Term4$logincome, Ey2, col="red")
```




```
  0   1   2
 57 208  10

Call:
lm(formula = logface ~ logincome + single, data = Term4)

Residuals:
    Min      1Q  Median      3Q     Max
-6.2828 -0.8785  0.0364  0.9227  5.9573

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.09007    0.88643   5.742 2.49e-08 ***
logincome    0.63378    0.07776   8.151 1.33e-14 ***
single      -0.80006    0.24796  -3.227  0.00141 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.615 on 272 degrees of freedom
Multiple R-squared:  0.2605,    Adjusted R-squared:  0.255
F-statistic:  47.9 on 2 and 272 DF,  p-value: < 2.2e-16
```

Show Overhead D Details. Interaction Terms

- Linear regression models are defined in terms of linear combinations of explanatory varibles but we

can expand their scope through nonlinear transformations

- One type of nonlinear transform is the product of two varibles that is used to create what is known as an *interaction* variable
- To interpret coefficients, we now consider the regression function

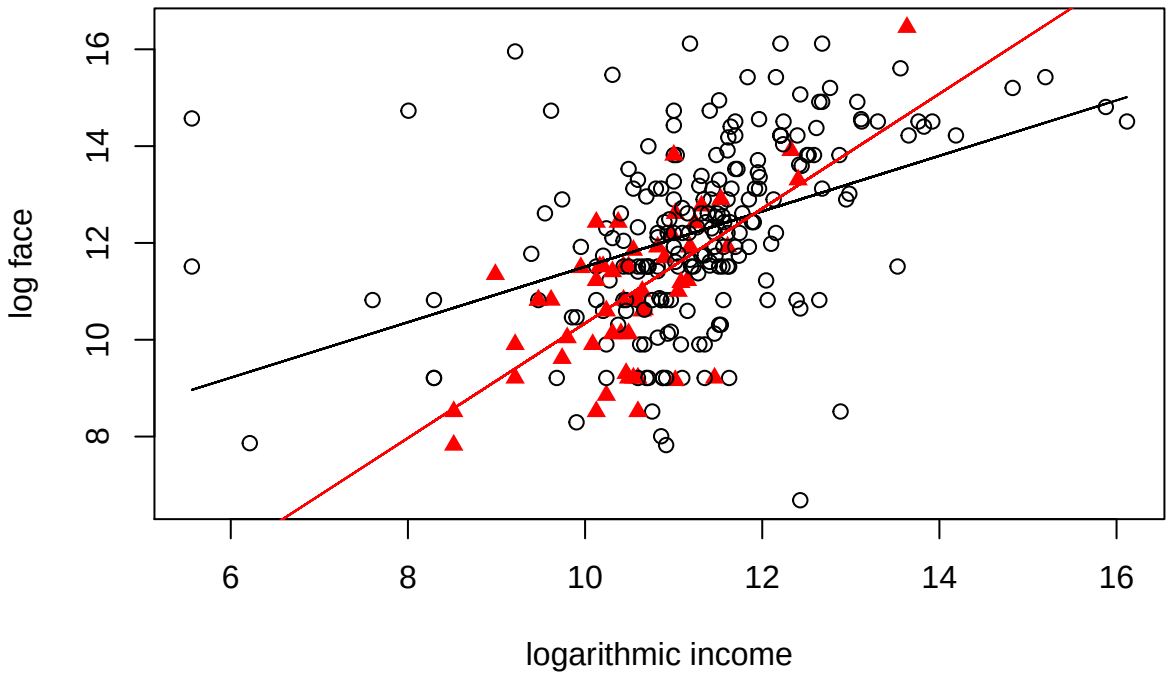
$$\text{E } logface = \beta_0 + \beta_1 logincome + \beta_2 single + \beta_3 single * logincome$$

- This can be expressed as two lines with different slopes

$$\text{E } logface = \begin{cases} \beta_0 + \beta_1 logincome & \text{for other respondents} \\ \beta_0 + \beta_2 + (\beta_1 + \beta_3) logincome & \text{for single respondents} \end{cases}.$$

Show Overhead E Details. Visualizing binary variables with interactions terms

```
Term4 <- Term1[,c("numhh", "education", "logincome", "logface", "marstat")]
Term4$marstat <- as.factor(Term4$marstat)
table(Term4$marstat)
Term4$single <- 1*(Term4$marstat == 0)
model_single_inter <- lm(logface ~ logincome + single + single*logincome, data = Term4)
summary(model_single_inter)

plot(Term4$logincome,Term4$logface,xlab="logarithmic income", ylab="log face",
    pch= 1+16*Term4$single, col = c("red", "black", "black")[Term4$marstat])
Ey1 <- model_single_inter$coefficients[1]+model_single_inter$coefficients[2]*Term4$logincome
Ey2 <- Ey1 + model_single_inter$coefficients[3]+model_single_inter$coefficients[4]*Term4$logincome
lines(Term4$logincome, Ey1)
lines(Term4$logincome, Ey2, col="red")
```



```
  0   1   2
 57 208  10

Call:
lm(formula = logface ~ logincome + single + single * logincome,
    data = Term4)

Residuals:
    Min      1Q  Median      3Q     Max
-6.2149 -0.8287  0.0696  0.9308  5.6070

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      5.77902    0.92550   6.244 1.64e-09 ***
logincome        0.57288    0.08124   7.051 1.47e-11 ***
single          -7.29211    2.74216  -2.659   0.0083 **
logincome:single  0.61244    0.25764   2.377   0.0181 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.601 on 271 degrees of freedom
Multiple R-squared:  0.2756,    Adjusted R-squared:  0.2676
F-statistic: 34.36 on 3 and 271 DF,  p-value: < 2.2e-16
```

3.3.2 Exercise. Binary variables and term life

Assignment Text

In the prior video, we saw how the variable `single` can be used with logarithmic income to explain logarithmic face amounts of term life insurance that people purchase. The coefficient associated with this variable turns out to be negative which is intuitively appealing; if an individual is single, then that person may not have the strong need to purchase financial security for others in the event of unexpected death.

In this exercise, we will extend this by incorporating `single` into our larger regression model that contains other explanatory varibles, `logincome`, `education` and `numhh`. The data have been pre-loaded into the dataframe `Term4`.

Instructions

- Calculate a table of correlation coefficients to examine pairwise linear relationships among the variables `numhh`, `education`, `logincome`, `single`, and `logface`.
- Fit a MLR model of `logface` using explanatory variables `numhh`, `education`, `logincome`, and `single`. Examine the residual standard deviation s, the coefficient of determination R^2, and the adjusted version R_a^2. Also note the statistical significance of the coefficient associated with `single`.
- Repeat the MLR model fit while adding the interaction term `single*logincome`.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNUZXJtIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFx0Z

3.4 Categorical variables

In this section, you learn how to:

- Represent categorical variables using a set of binary variables
- Interpret the regression coefficients associated with categorical variables
- Describe the effect of the reference level choice on the model fit

---

3.4.1 Video

Video Overhead Details

Show Overhead A Details. Categorical variables

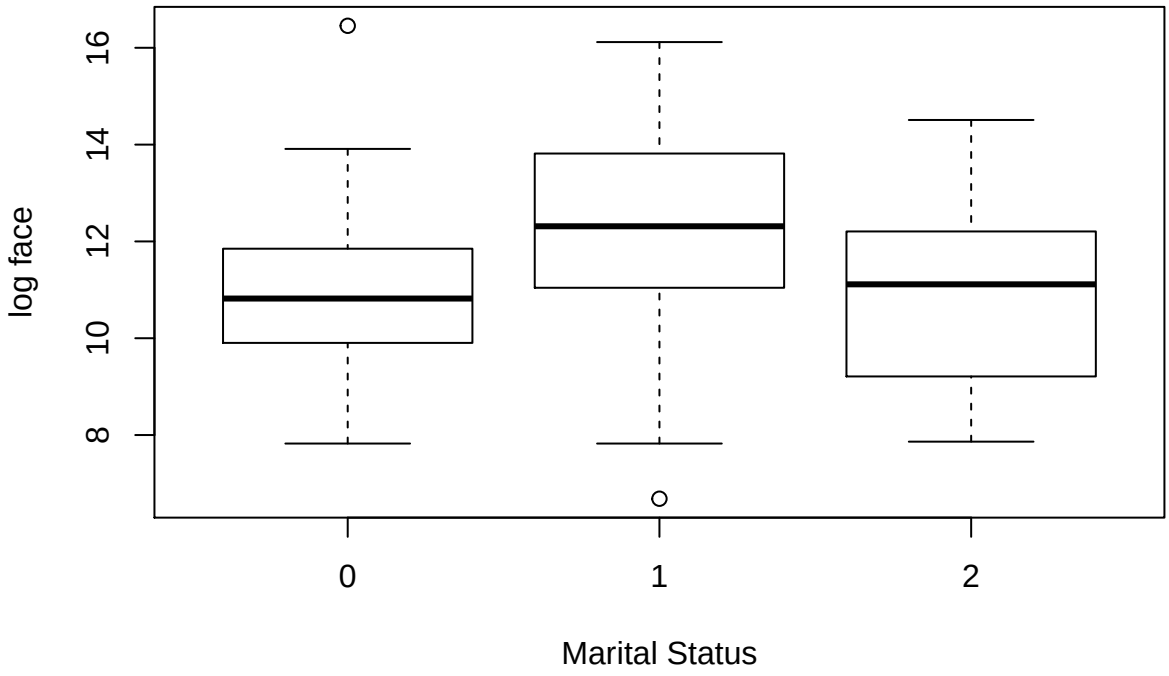
- *Categorical variables* provide labels for observations to denote membership in distinct groups, or categories.
- A *binary variable* is a special case of a categorical variable.
  - To illustrate, a binary variable may tell us whether or not someone has health insurance.
  - A categorical variable could tell us whether someone has (i) private individual health insurance, (ii) private group insurance, (iii) public insurance or (iv) no health insurance.
- For categorical variables, there may or may not be an ordering of the groups.
  - For health insurance, it is difficult to say which is 'larger', private individual versus public health insurance (such as Medicare).
  - However, for education, we may group individuals from a dataset into 'low', 'intermediate' and 'high' years of education.
- *Factor* is another term used for a (unordered) categorical explanatory variable.

Show Overhead B Details. Term life example

- We studied $y = logface$, the amount that the company will pay in the event of the death of the named insured (in logarithmic dollars), focusing on the explanatory variables *logincome*, *education*, and *numhh*.
- We now supplement this by including the categorical variable, *marstat*, that is the marital status of the survey respondent. This may be:
  - 1, for married
  - 2, for living with partner
  - 0, for other (SCF actually breaks this category into separated, divorced, widowed, never married and inapplicable, for persons age 17 or less or no further persons)

Show Overhead C Details. Term life boxplots

```
# Pre-exercise code
#Term <- read.csv("CSVData\\term_life.csv", header = TRUE)
Term <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/efc64bc2d78cf6b48ad1
Term1 <- subset(Term, subset = face > 0)
Term4 <- Term1[,c("numhh", "education", "logincome", "marstat", "logface")]
Term4$single <- 1*(Term4$marstat == 0)
Term4$marstat<- as.factor(Term4$marstat)
boxplot(logface ~ marstat, ylab = "log face", xlab = "Marital Status", data = Term4)
```




```
table(Term4$marstat)
#  SUMMARY BY LEVEL OF MARSTAT
#library(Rcmdr)
#numSummary(Term4[, "logface"], groups = Term4$marstat, statistics = c("mean", "sd"))
#numSummary(Term4[, "logface"], statistics = c("mean", "sd"))
```

```
  0   1   2
 57 208  10
```

Show Overhead D Details. Regression with a categorical variable

```
Term4$marstat <- as.factor(Term4$marstat)
Term4$marstat <- relevel(Term4$marstat, ref = "2")
summary(lm(logface ~ logincome+education+numhh+marstat, data = Term4))
```

```
Call:
lm(formula = logface ~ logincome + education + numhh + marstat,
    data = Term4)

Residuals:
    Min      1Q  Median      3Q     Max
-5.8875 -0.8505  0.1124  0.8468  4.5173

Coefficients:
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.60536    0.95218   2.736 0.006629 **
logincome    0.45151    0.07872   5.736 2.61e-08 ***
education    0.20467    0.03862   5.299 2.42e-07 ***
numhh        0.24770    0.06940   3.569 0.000424 ***
marstat0     0.23234    0.53283   0.436 0.663155
marstat1     0.78941    0.49532   1.594 0.112169
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.513 on 269 degrees of freedom
Multiple R-squared:  0.358, Adjusted R-squared:  0.3461
F-statistic:    30 on 5 and 269 DF,  p-value: < 2.2e-16
```

Show Overhead E Details. t-ratios depend on the reference level

	Model 1		Model 2		Model 3	
Var	Coef	t-stat	Coef	t-stat	Coef	t-stat
logincome	0.452	5.74	0.452	5.74	0.452	5.74
education	0.205	5.30	0.205	5.30	0.205	5.30
numhh	0.248	3.57	0.248	3.57	0.248	3.57
Intercept	3.395	3.77	2.605	2.74	2.838	3.34
mar=0	−0.557	−2.15	0.232	0.44		
mar=1			0.789	1.59	0.557	2.15
mar=2	−0.789	−1.59			−0.232	−0.44

3.4.2 Exercise. Categorical variables and Wisconsin hospital costs

Assignment Text

This exercise examines the impact of various predictors on hospital charges. Identifying predictors of hospital charges can provide direction for hospitals, government, insurers and consumers in controlling these variables that in turn leads to better control of hospital costs. The data, from 1989, are aggregated by:

- `drg`, diagnostic related groups of costs,
- `payer`, type of health care provider (Fee for service, HMO, and other), and
- `hsa`, nine major geographic areas in Wisconsin.

Some preliminary analysis of the data has already been done. In this exercise, we will analyze `logcharge`, the logarithm of total hospital charges per number of discharges, in terms of `log_numdschg`, the logarithm of the number of discharges. In the dataframe `Hcost` which has been loaded in advance, we restrict consideration to three types of drgs, numbers 209, 391, and 431.

Instructions

- Fit a basic linear regression model using logarithmic number of discharges to predict logarithmic hospital costs and superimposed the fitted regression line on the scatter plot.
- Produce a scatter plot of logarithmic number of discharges to predict logarithmic hospital costs. Allow plotting symbols and colors to vary by diagnostic related group.
- Fit a MLR model using logarithmic number of discharges to predict logarithmic hospital costs, allowing intercepts and slopes to vary by diagnostic related groups.
- Superimpose the fits from the MLR model on the scatter plot of logarithmic number of discharges to predict logarithmic hospital costs.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNIY29zdCA8LSByZWFkLmNzdihcIkNTVkRhdGFcXFxcV2l

3.5 General linear hypothesis

In this section, you learn how to:

- Jointly test the significance of a set of regression coefficients using the general linear hypothesis
- Conduct a test of a regression coefficient versus one- or two-side alternatives

---

3.5.1 Video

Video Overhead Details

Show Overhead A Details. Testing the significance of a categorical variable

```
#Term <- read.csv("CSVData\\term_life.csv", header = TRUE)
Term <- read.csv("https://assets.datacamp.com/production/repositories/2610/datasets/efc64bc2d78cf6b48ad1
Term1 <- subset(Term, subset = face > 0)
Term4 <- Term1[,c("numhh", "education", "logincome", "marstat", "logface")]
Term_mlr1 <- lm(logface ~ logincome + education + numhh + as.factor(Term4$marstat), data = Term4)
anova(Term_mlr1)
Term_mlr2 <- lm(logface ~ logincome + education + numhh, data = Term4)

Fstat <- (anova(Term_mlr2)$`Sum Sq`[4] - anova(Term_mlr1)$`Sum Sq`[5])/(2*anova(Term_mlr1)$`Mean Sq`[5]
Fstat
cat("p-value is", 1 - pf(Fstat, df1 = 2 , df2 = anova(Term_mlr1)$Df[5]))
```

```
Analysis of Variance Table

Response: logface
                         Df Sum Sq Mean Sq F value    Pr(>F)
logincome                 1 222.63 222.629  97.280 < 2.2e-16 ***
education                 1  51.50  51.502  22.504 3.407e-06 ***
numhh                     1  54.34  54.336  23.743 1.883e-06 ***
as.factor(Term4$marstat)  2  14.81   7.406   3.236   0.04085 *
Residuals               269 615.62   2.289
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
[1] 3.236048
p-value is 0.04085475
```

Show Overhead B Details. Overview of the general linear hypothesis

- The likelihood ratio is a general statistical test procedure that compares a model to a subset
- The general linear hypothesis test procedure is similar.
  - Start with a (large) linear regression model, examine the fit to a set of data
  - Compare this to smaller model that is a subset of the large model.
  - "Subset" is the sense that regression coefficients from the small model are linear combinations of regression coefficients of the large model (e.g., set them to zero)
- Although the likelihood ratio test is more generally available, the general linear hypothesis test is more accurate for smaller data sets (for normally distributed data)

Show Overhead C Details. Procedure for conducting the general linear hypothesis

- Run the full regression and get the error sum of squares and mean square error, which we label as $(ErrorSS)_{full}$ and s^2_{full}, respectively.
- Run a reduced regression and get the error sum of squares, labelled $(ErrorSS)_{reduced}$.
- Using p for the number of linear restrictions, calculate

$$F - ratio = \frac{(ErrorSS)_{reduced} - (ErrorSS)_{full}}{ps^2_{full}}.$$

- The probability value is $p - value = \Pr(F_{p,df} > F - ratio)$ where $F_{p,df}$ has an F distribution with degrees of freedom p and df, respectively. (Here, df is the degrees of freedom for the full model.)

Show Overhead D Details. The general linear hypothesis for a single variable

- Suppose that you wish to test the hypothesisthat a regression coefficient equals 0.
  - One could use the general linear hypothsis procedure with $p = 1$.
  - One could also examine the corresponding $t - ratio$.
  - Which is correct?
- Both. One can show that $(t - ratio)^2 = F - ratio$, so they are equivalent statistics.
  - The general linear hypothesis is useful because it can be extended to multiple coefficients.
  - The *t-ratio* is useful because it can be used to examine one-sided alternative hypotheses.

3.5.2 Exercise. Hypothesis testing and term life

Assignment Text

With our `Term life` data, let us compare a model based on the binary variable that indicates whether a survey respondent is single versus the more complex marital status, `marstat`. In principle, more detailed information is better. But, it may be that the additional information in `marstat`, compared to `single`, does not help fit the data in a significantly better way.

As part of the preparatory work, the dataframe `Term4` is available that includes the binary variable `single` and the factor `marstat`. Moreover, the regression object `Term_mlr` contains information in a multiple linear regression fit of `logface` on the base explanatory variables 'logincome,education, andnumhh'.

Instructions

- Fit a MLR model using the base explanatory variables plus `single` and another model using the base variables plus `marstat`.
- Use the F test to decide whether the additional complexity `marstat` is warranted by calculating the p-value associated with this test.
- Fit a MLR model using the base explanatory variables plus `single` interacted with `logincome` and another model using the base variables plus `marstat` interacted with `logincome`.
- Use the F test to decide whether the additional complexity `marstat` is warranted by calculating the p-value associated with this test.

Hint

```
Here is the code to calculate it by hand
Fstat12 <- (anova(Term_mlr1)$`Sum Sq`[5] -
              anova(Term_mlr2)$`Sum Sq`[5])/(1*anova(Term_mlr2)$`Mean Sq`[5])
Fstat12
cat("p-value is", 1 - pf(Fstat12, df1 = 1 , df2 = anova(Term_mlr2)$Df[5]))
```

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNUZXJtIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFx0Z

3.5.3 Exercise. Hypothesis testing and Wisconsin hospital costs

Assignment Text

In a previous exercise, you were introduced to a dataset with hospital charges aggregated by:

- `drg`, diagnostic related groups of costs,
- `payer`, type of health care provider (Fee for service, HMO, and other), and
- `hsa`, nine major geographic areas.

We continue our analysis of the outcome variable `logcharge`, the logarithm of total hospital charges per number of discharges, in terms of `log_numdschg`, the logarithm of the number of discharges, as well as the three categorical variables used in the aggregation. As before, we restrict consideration to three types of drgs, numbers 209, 391, and 431 that has been preloaded in the dataframe `Hcost1`.

Instructions

- Fit a basic linear regression model using logarithmic hospital costs as the outcome variable and explanatory variable logarithmic number of discharges.
- Fit a MLR model using logarithmic hospital costs as the outcome variable and explanatory variables logarithmic number of discharges and the categorical variable diagnostic related group. Identify the F statistic and p value that test the importance of diagnostic related group.
- Fit a MLR model using logarithmic hospital costs as the outcome variable and explanatory variable logarithmic number of discharges interacted with diagnostic related group. Identify the F statistic and p value that test the importance of diagnostic related group interaction with logarithmic number of discharges.
- Calculate a coefficient of determination, R^2, for each of these models as well as for a model using logarithmic number of discharges and categorical variable `hsa` as predictors.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNIY29zdCA8LSByZWFkLmNzdihcIkNTVkRhdGFcXFxcV2l

3.5.4 Exercise. Hypothesis testing and auto claims

Assignment Text

As an actuarial analyst, you are working with a large insurance company to help them understand their claims distribution for their private passenger automobile policies. You have available claims data for a recent year, consisting of:

- `state`: codes 01 through 17 used, with each code randomly assigned to an actual individual state
- `class`: rating class of operator, based on age, gender, marital status, and use of vehicle
- `gender`: operator gender
- `age`: operator age
- `paid`: amount paid to settle and close a claim.

You are focusing on older drivers, 50 and higher, for which there are $n = 6{,}773$ claims available.

Instructions

a. Run a regression of `logpaid` on `age`. Is `age` a statistically significant variable? To respond to this question, use a formal test of hypothesis. State your null and alternative hypotheses, decision-making criterion, and your decision-making rule. Also comment on the goodness of fit of this variable.

b. Consider using class as a single explanatory variable. Use the one factor to estimate the model and respond to the following questions.

b (i). What is the point estimate of claims in class C7, drivers 50-69, driving to work or school, less than 30 miles per week with annual mileage under 7500, in natural logarithmic units?

b (ii). Determine the corresponding 95% confidence interval of expected claims, in natural logarithmic units.

b (iii). Convert the 95% confidence interval of expected claims that you determined in part b(ii) to dollars.

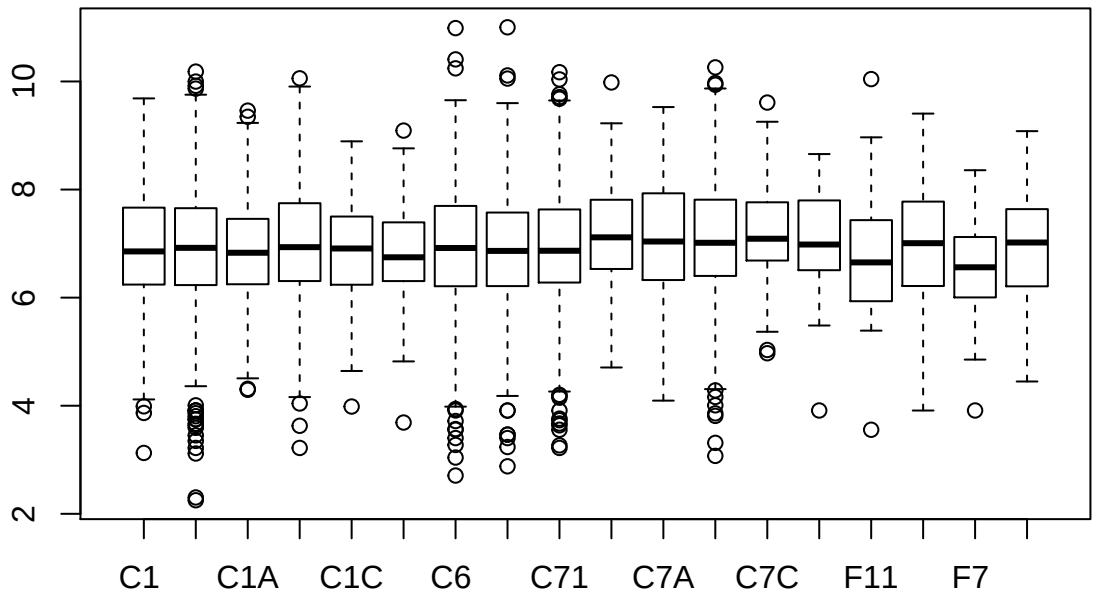
c. Run a regression of `logpaid` on `age`, `gender` and the categorical variables `state` and `class`.

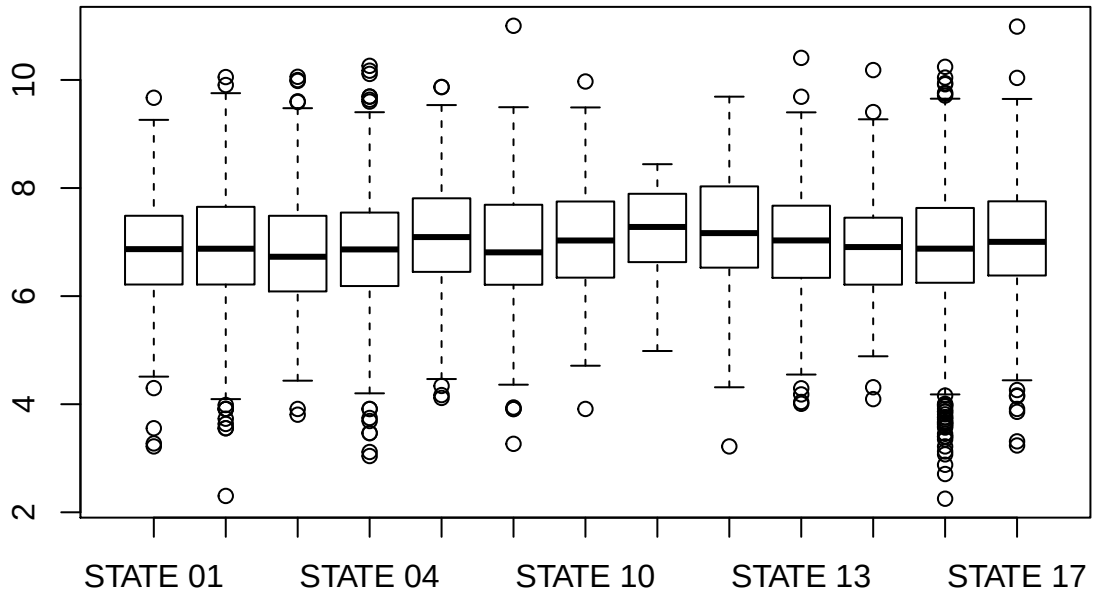
c (i). Is `gender` a statistically significant variable? To respond to this question, use a formal test of hypothesis. State your null and alternative hypotheses, decision-making criterion, and your decision-making rule.

c (ii). Is `class` a statistically significant variable? To respond to this question, use a formal test of hypothesis. State your null and alternative hypotheses, decision-making criterion, and your decision-making rule.

c (iii). Use the model to provide a point estimate of claims in dollars (not log dollars) for a male age 60 in STATE 2 in `class` C7.

c (iv). Write down the coefficient associated with `class` C7 and interpret this coefficient.






```
Call:
lm(formula = logpaid ~ class, data = AutoC)

Residuals:
    Min      1Q  Median      3Q     Max
-4.7002 -0.6912 -0.0437  0.7128  4.1009

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.940938   0.039699 174.838   <2e-16 ***
classC11     0.010564   0.050696   0.208   0.8349
classC1A    -0.075432   0.128202  -0.588   0.5563
classC1B     0.057420   0.065380   0.878   0.3798
classC1C    -0.154707   0.178007  -0.869   0.3848
classC2     -0.139442   0.142595  -0.978   0.3282
classC6     -0.015057   0.053217  -0.283   0.7772
classC7     -0.039775   0.053191  -0.748   0.4546
classC71     0.012730   0.050887   0.250   0.8025
classC72     0.241716   0.122626   1.971   0.0487 *
classC7A     0.122755   0.108174   1.135   0.2565
classC7B     0.131512   0.056956   2.309   0.0210 *
classC7C     0.302596   0.125307   2.415   0.0158 *
classF1      0.062962   0.202561   0.311   0.7559
classF11    -0.136891   0.173726  -0.788   0.4307
classF6     -0.030546   0.094148  -0.324   0.7456
```

```
classF7     -0.363874   0.144807  -2.513   0.0120 *
classF71    -0.005476   0.117810  -0.046   0.9629
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.07 on 6755 degrees of freedom
Multiple R-squared:  0.005048,  Adjusted R-squared:  0.002544
F-statistic: 2.016 on 17 and 6755 DF,  p-value: 0.00786


Call:
lm(formula = logpaid ~ state, data = AutoC)

Residuals:
    Min      1Q  Median      3Q     Max
-4.6555 -0.6847 -0.0415  0.7005  4.0788

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    6.82202    0.08286  82.328  < 2e-16 ***
stateSTATE 02  0.11739    0.08878   1.322 0.186133
stateSTATE 03  0.02224    0.10071   0.221 0.825246
stateSTATE 04  0.05195    0.09262   0.561 0.574893
stateSTATE 06  0.30735    0.09327   3.295 0.000988 ***
stateSTATE 07  0.10127    0.10537   0.961 0.336562
stateSTATE 10  0.21247    0.10486   2.026 0.042787 *
stateSTATE 11  0.17462    0.36540   0.478 0.632751
stateSTATE 12  0.40876    0.10715   3.815 0.000138 ***
stateSTATE 13  0.21792    0.11111   1.961 0.049890 *
stateSTATE 14  0.06485    0.11667   0.556 0.578297
stateSTATE 15  0.08479    0.08596   0.986 0.323956
stateSTATE 17  0.22406    0.09585   2.338 0.019442 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.068 on 6760 degrees of freedom
Multiple R-squared:  0.008109,  Adjusted R-squared:  0.006348
F-statistic: 4.606 on 12 and 6760 DF,  p-value: 1.749e-07

Analysis of Variance Table

Response: logpaid
            Df Sum Sq Mean Sq F value    Pr(>F)
state       12   63.0  5.2495  4.6055 1.749e-07 ***
Residuals 6760 7705.2  1.1398
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Call:
lm(formula = logpaid ~ class + state + age + gender, data = AutoC)

Residuals:
    Min      1Q  Median      3Q     Max
-4.7266 -0.6802 -0.0433  0.7072  4.1809
```

```
Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    6.974923   0.140144  49.770  < 2e-16 ***
classC11       0.055573   0.051695   1.075 0.282410
classC1A      -0.110591   0.128238  -0.862 0.388506
classC1B       0.022671   0.066388   0.341 0.732740
classC1C      -0.160355   0.178056  -0.901 0.367842
classC2       -0.183935   0.142784  -1.288 0.197719
classC6        0.057560   0.058767   0.979 0.327395
classC7       -0.021854   0.054531  -0.401 0.688605
classC71       0.024622   0.053049   0.464 0.642569
classC72       0.260057   0.123451   2.107 0.035192 *
classC7A       0.119315   0.108879   1.096 0.273183
classC7B       0.124196   0.059098   2.102 0.035634 *
classC7C       0.299023   0.126300   2.368 0.017934 *
classF1        0.130054   0.202380   0.643 0.520491
classF11      -0.058684   0.174394  -0.337 0.736503
classF6        0.068612   0.098270   0.698 0.485079
classF7       -0.309974   0.145271  -2.134 0.032898 *
classF71       0.029459   0.118848   0.248 0.804239
stateSTATE 02  0.098538   0.089280   1.104 0.269765
stateSTATE 03  0.006302   0.101050   0.062 0.950272
stateSTATE 04  0.019266   0.093712   0.206 0.837118
stateSTATE 06  0.283853   0.094105   3.016 0.002568 **
stateSTATE 07  0.076593   0.106361   0.720 0.471475
stateSTATE 10  0.181802   0.105448   1.724 0.084739 .
stateSTATE 11  0.172272   0.365574   0.471 0.637487
stateSTATE 12  0.388006   0.108333   3.582 0.000344 ***
stateSTATE 13  0.188491   0.111980   1.683 0.092371 .
stateSTATE 14  0.068141   0.116720   0.584 0.559373
stateSTATE 15  0.065570   0.086624   0.757 0.449110
stateSTATE 17  0.199398   0.096914   2.057 0.039680 *
age           -0.003021   0.001690  -1.787 0.073914 .
genderM        0.038953   0.026907   1.448 0.147747
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.066 on 6741 degrees of freedom
Multiple R-squared:  0.01365,   Adjusted R-squared:  0.009113
F-statistic: 3.009 on 31 and 6741 DF,  p-value: 4.354e-08

Analysis of Variance Table

Response: logpaid
            Df Sum Sq Mean Sq F value    Pr(>F)
class       17   39.2  2.3066  2.0293  0.007346 **
state       12   61.0  5.0841  4.4728 3.354e-07 ***
age          1    3.4  3.4284  3.0162  0.082480 .
gender       1    2.4  2.3823  2.0958  0.147747
Residuals 6741 7662.2  1.1367
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Submission Correctness Tests (SCT)

success_msg("Congratulations!")

Chapter 4

Variable Selection

Chapter description

This chapter describes tools and techniques to help you select variables to enter into a linear regression model, beginning with an iterative model selection process. In applications with many potential explanatory variables, automatic variable selection procedures are available that will help you quickly evaluate many models. Nonetheless, automatic procedures have serious limitations including the inability to account properly for nonlinearities such as the impact of unusual points; this chapter expands upon the Chapter 2 discussion of unusual points. It also describes collinearity, a common feature of regression data where explanatory variables are linearly related to one another. Other topics that impact variable selection, including out-of-sample validation, are also introduced.

4.1 An iterative approach to data analysis and modeling

In this section, you learn how to:

- Describe the iterative approach to data analysis and modeling.

4.1.1 Video

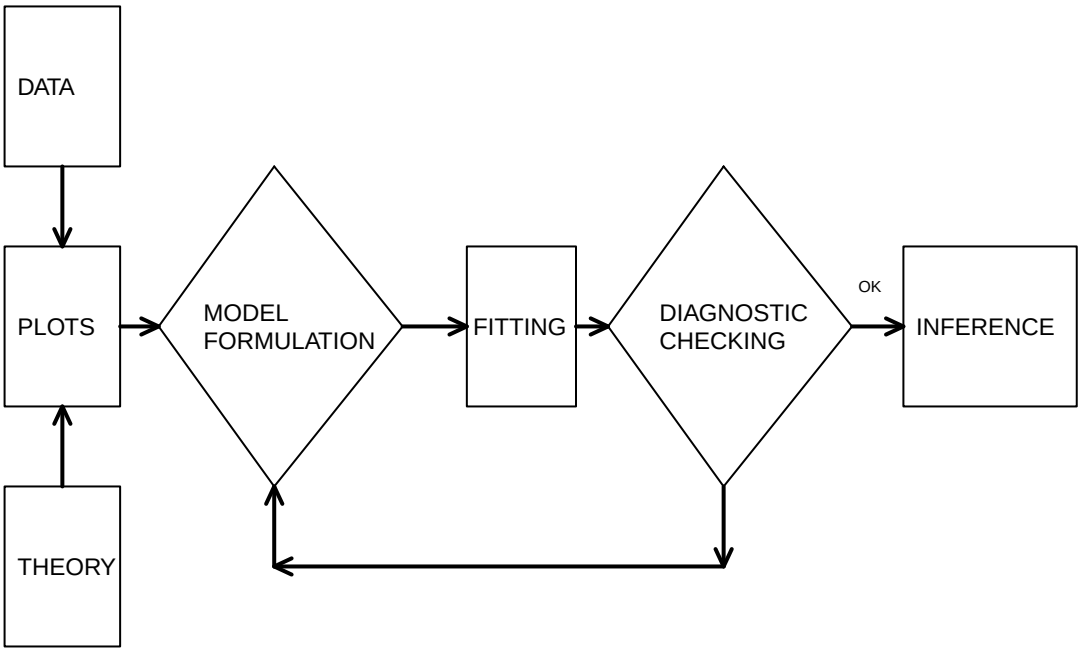
Video Overhead Details

Show Overhead A Details. Iterative approach

- Model formulation stage
- Fitting
- Diagnostic checking - the data and model must be consistent with one another before additional inferences can be made.

```
plot.new()
par(mar=c(0,0,0,0), cex=0.9)
plot.window(xlim=c(0,18),ylim=c(-5,5))

text(1,3,labels="DATA",adj=0, cex=0.8)
text(1,0,labels="PLOTS",adj=0, cex=0.8)
```

```
text(1,-3,labels="THEORY",adj=0, cex=0.8)
text(3.9,0,labels="MODEL\nFORMULATION",adj=0, cex=0.8)
text(8.1,0,labels="FITTING",adj=0, cex=0.8)
text(11,0,labels="DIAGNOSTIC\nCHECKING",adj=0, cex=0.8)
text(15,0,labels="INFERENCE",adj=0, cex=0.8)
text(14.1,0.5,labels="OK",adj=0, cex=0.6)

rect(0.8,2.0,2.6,4.0)
arrows(1.7,2.0,1.7,1.0,code=2,lwd=2,angle=25,length=0.10)
rect(0.8,-1.0,2.6,1.0)
arrows(1.7,-2.0,1.7,-1.0,code=2,lwd=2,angle=25,length=0.10)
rect(0.8,-4.0,2.6,-2.0)

arrows(2.6,0,3.2,0,code=2,lwd=2,angle=25,length=0.10)

x<-c(5,7.0,5,3.2)
y<-c(2,0,-2,0)
polygon(x,y)
arrows(7.0,0,8.0,0,code=2,lwd=2,angle=25,length=0.10)

rect(8.0,-1.0,9.7,1.0)
arrows(9.7,0,10.2,0,code=2,lwd=2,angle=25,length=0.10)

x1<-c(12,14.0,12,10.2)
y1<-c(2,0,-2,0)
polygon(x1,y1)
arrows(14.0,0,14.8,0,code=2,lwd=2,angle=25,length=0.10)

rect(14.8,-1.0,17.5,1.0)
arrows(12,-2.0,12,-3,code=2,lwd=2,angle=25,length=0.10)
arrows(12,-3.0,5,-3,code=2,lwd=2,angle=25,length=0.10)
arrows(5,-3.0,5,-2,code=2,lwd=2,angle=25,length=0.10)
```



Show Overhead B Details. Many possible models

E $y = \beta_0$	1 model with no variables
E $y = \beta_0 + \beta_1 x_i$,	4 models with one variable
E $y = \beta_0 + \beta_1 x_i + \beta_2 x_j$,	6 models with two variables
E $y = \beta_0 + \beta_1 x_1 + \beta_2 x_j + \beta_3 x_k$,	4 models with three variables
E $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4$	1 model with all variables

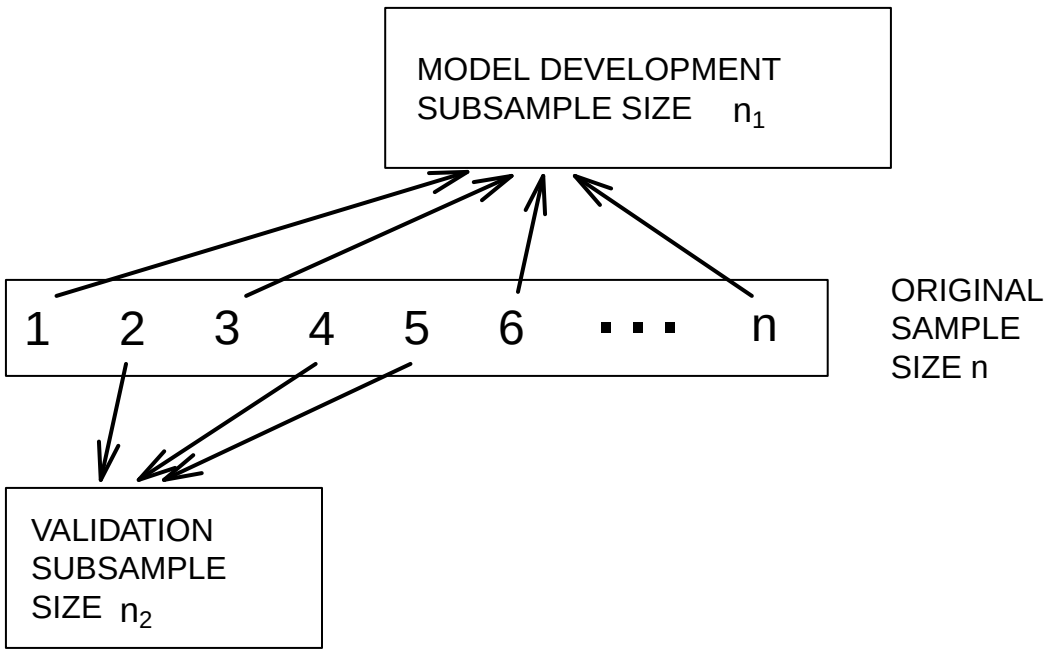
- With k explanatory variables, there are 2^k possible linear models
- There are infinitely many nonlinear ones!!

Show Overhead C Details. Model validation

- Model validation is the process of confirming our proposed model.
- Concern: *data-snooping* - fitting many models to a single set of data.
  - Response to concern: *out-of-sample validation*.
  - Divide the data into *model development*, or *training* and *validation*, or *test*, subsamples.

```
par(mai=c(0,0.1,0,0))
plot.new()
plot.window(xlim=c(0,18),ylim=c(-10,10))
rect(1,-1.2,14,1.2)
rect(7,4,15,8)
rect(1,-8,6,-4)
x<-seq(1.5,9,length=6)
```

```
y<-rep(0,6)
text(x,y,labels=c(1:6),cex=1.5)
x1<-seq(10.5,11.5,length=3)
y1<-rep(0,3)
text(x1,y1,labels=rep(".",3),cex=3)
text(13,0,labels="n",cex=1.5)

text(15,0,labels="ORIGINAL\nSAMPLE\nSIZE n",adj=0)
text(7.5,6,labels="MODEL DEVELOPMENT\nSUBSAMPLE SIZE",adj=0)
text(12.5,5.3, expression(n[1]), adj=0, cex=1.1)
text(1.4,-6,labels="VALIDATION\nSUBSAMPLE\nSIZE",adj=0)
text(2.8,-7.2,expression(n[2]),adj=0, cex=1.1)

arrows(1.8,0.8,8.3,3.9,code=2,lwd=2,angle=15,length=0.2)
arrows(4.8,0.8,9,3.8,code=2,lwd=2,angle=15,length=0.2)
arrows(9.1,0.9,9.5,3.8,code=2,lwd=2,angle=15,length=0.2)
arrows(12.8,0.8,10,3.8,code=2,lwd=2,angle=15,length=0.2)
arrows(2.9,-0.9,2.5,-3.8,code=2,lwd=2,angle=15,length=0.2)
arrows(5.9,-0.9,3.1,-3.8,code=2,lwd=2,angle=15,length=0.2)
arrows(7.4,-0.9,3.5,-3.8,code=2,lwd=2,angle=15,length=0.2)
```



4.1.2 MC Exercise. An iterative approach to data modeling

Which of the following is not true?

- A. Diagnostic checking reveals symptoms of mistakes made in previous specifications.
- B. Diagnostic checking provides ways to correct mistakes made in previous specifications.
- C. Model formulation is accomplished by using prior knowledge of relationships.
- D. Understanding theoretical model properties is not really helpful when matching a model to data or inferring general relationships based on the data.

4.2 Automatic variable selection procedures

In this section, you learn how to:

- Identify some examples of automatic variable selection procedures
- Describe the purpose of automatic variable selection procedures and their limitations
- Describe "data-snooping"

4.2.1 Video

Video Overhead Details

Show Overhead A Details. Classic stepwise regression algorithm

Suppose that the analyst has identified one variable as the outcome, y, and k potential explanatory variables, $x_1, x_2, \ldots, x_k$.

- (i). Consider all possible regressions using one explanatory variable. Choose the one with the highest t-statistic.
- (ii). Add a variable to the model from the previous step. The variable to enter is with the highest t-statistic.
- (iii). Delete a variable to the model from the previous step. Delete the variable with the small t-statistic if the statistic is less than, e.g., 2 in absolute value.
- (iv). Repeat steps (ii) and (iii) until all possible additions and deletions are performed.

Show Overhead B Details. Drawbacks of stepwise regression

- The procedure "snoops" through a large number of models and may fit the data "too well."
- There is no guarantee that the selected model is the best.
  - The algorithm does not consider models that are based on nonlinear combinations of explanatory variables.
  - It ignores the presence of outliers and high leverage points.

Show Overhead C Details. Data-snooping in stepwise regression

- Generate y and $x_1 - x_{50}$ using a random number generator
- By design, there is no relation between y and $x_1 - x_{50}$.
- **But**, through stepwise regression, we **"discover"** a relationship that explains 14% of the variation!!!

```
Call: lm(formula = y ~ xvar27 + xvar29 + xvar32, data = X)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.04885    0.09531  -0.513   0.6094
```

```
xvar27        0.21063    0.09724   2.166   0.0328 *
xvar29        0.24887    0.10185   2.443   0.0164 *
xvar32        0.25390    0.09823   2.585   0.0112 *

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9171 on 96 degrees of freedom
Multiple R-squared:  0.1401,    Adjusted R-squared:  0.1132
F-statistic: 5.212 on 3 and 96 DF,  p-value: 0.002233
```

Show Overhead D Details. Variants of stepwise regression

This uses the `R` function step()

- The option `direction` can be used to change how variables enter
  - Forward selection. Add one variable at a time without trying to delete variables.
  - Backwards selection. Start with the full model and delete one variable at a time without trying to add variables.
- The option `scope` can be used to specify which variables must be included

Show Overhead E Details. Automatic variable selection procedures

- Stepwise regression is a type of automatic variable selection procedure.
- These procedures are useful because they can quickly search through several candidate models. They mechanize certain routine tasks and are excellent at discovering patterns in data.
- They are so good at detecting patterns that they analyst must be wary of overfitting (data-snooping)

- They can miss certain patterns (nonlinearities, unusual points)
- A model suggested by automatic variable selection procedures should be subject to the same careful diagnostic checking procedures as a model arrived at by any other means

4.2.2 Exercise. Data-snooping in stepwise regression

Assignment Text

Automatic variable selection procedures, such as the classic stepwise regression algorithm, are very good at detecting patterns. Sometimes they are too good in the sense that they detect patterns in the sample that are not evident in the population from which the data are drawn. The detect "spurious" patterns.

This exercise illustrates this phenomenom by using a simulation, designed so that the outcome variable (y) and the explanatory variables are mutually independent. So, by design, there is no relationship between the outcome and the explanatory variables.

As part of the code set-up, we have $n = 100$ observations generated of the outcome y and 50 explanatory variables, `xvar1` through `xvar50`. As anticipated, collections of explanatory variables are not statistically significant. However, with the step() function, you will find some statistically significant relationships!

Instructions

- Fit a basic linear regression model and MLR model with the first ten explanatory variables. Compare the models via an F test.
- Fit a multiple linear regression model with all fifty explanatory variables. Compare this model to the one with ten variables via an F test.
- Use the `step` function to find the best model starting with the fitted model containing all fifty explanatory variables and summarize the fit.

Hint. The code shows stepwise regression using BIC, a criterion that results in simpler models than AIC. For AIC, use the option `k=2` in the [step()] function (the default)

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6InNldC5zZWVkKDEyMzcpXG5YIDwtIGFzLmRhdGEuZnJhb

4.3 Residual analysis

In this section, you learn how to:

- Explain how residual analysis can be used to improve a model specification
- Use relationships between residuals and potential explanatory variables to improve model specification

---

4.3.1 Video

Video Overhead Details

Show Overhead A Details. Residual analysis

- Use $e_i = y_i - \hat{y}_i$ as the ith residual.
- Later, I will discuss rescaling by, for example, s, to get a standardized residual.
- *Role of residuals*: If the model formulation is correct, then residuals should be approximately equal to random errors or "white noise."
- *Method of attack*: Look for patterns in the residuals. Use this information to improve the model specification.

Show Overhead B Details. Using residuals to select explanatory variables

- Residual analysis can help identify additional explanatory variables that may be used to improve the formulation of the model.
- If the model is correct, then residuals should resemble random errors and contain no discernible patterns.
- Thus, when comparing residuals to explanatory variables, we do not expect any relationships.
- If we do detect a relationship, then this suggests the need to control for this additional variable.

Show Overhead C Details. Detecting relationships between residuals and explanatory variables

- Calculate summary statistics and display the distribution of residuals to identify outliers.
- Calculate the correlation between the residuals and additional explanatory variables to search for linear relationships.
- Create scatter plots between the residuals and additional explanatory variables to search for nonlinear relationships.

4.3.2 Exercise. Residual analysis and risk manager survey

Assignment Text

This exercise examines data, pre-loaded in the dataframe `survey`, from a survey on the cost effectiveness of risk management practices. Risk management practices are activities undertaken by a firm to minimize the

potential cost of future losses, such as the event of a fire in a warehouse or an accident that injures employees. This exercise develops a model that can be used to make statements about cost of managing risks.

A measure of risk management cost effectiveness, `logcost`, is the outcome variable. This variable is defined as total property and casualty premiums and uninsured losses as a proportion of total assets, in logarithmic units. It is a proxy for annual expenditures associated with insurable events, standardized by company size. Explanatory variables include `logsize`, the logarithm of total firm assets, and `indcost`, a measure of the firm's industry risk.

Instructions

- Fit and summarize a MLR model using `logcost` as the outcome variable and `logsize` and `indcost` as explanatory variables.
- Plot residuals of the fitted model versus `indcost` and superimpose a locally fitted line using the `R` function lowess().
- Fit and summarize a MLR model of `logcost` on `logsize`, `indcost` and a squared version of `indcost`.
- Plot residuals of the fitted model versus 'indcost' and superimpose a locally fitted line using lowess().

Hint. You can access model residuals using `mlr.survey1$residuals` or `mlr.survey1($residuals)`

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNzdXJ2ZXkgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXF

4.3.3 Exercise. Added variable plot and refrigerator prices

Assignment Text

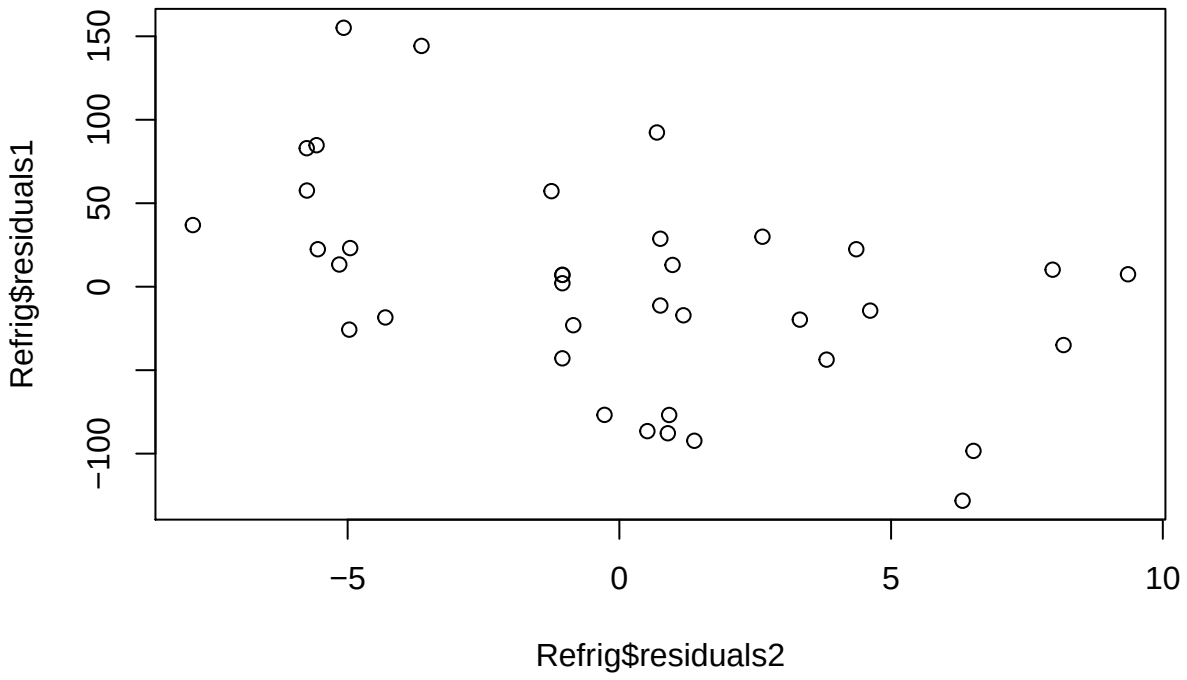
What characteristics of a refrigerator are important in determining its price (`price`)? We consider here several characteristics of a refrigerator, including the size of the refrigerator in cubic feet (`rsize`), the size of the freezer compartment in cubic feet (`fsize`), the average amount of money spent per year to operate the refrigerator (`ecost`, for energy cost), the number of shelves in the refrigerator and freezer doors (`shelves`), and the number of features (`features`). The features variable includes shelves for cans, see-through crispers, ice makers, egg racks and so on.

Both consumers and manufacturers are interested in models of refrigerator prices. Other things equal, consumers generally prefer larger refrigerators with lower energy costs that have more features. Due to forces of supply and demand, we would expect consumers to pay more for these refrigerators. A larger refrigerator with lower energy costs that has more features at the similar price is considered a bargain to the consumer. How much extra would the consumer be willing to pay for this additional space? A model of prices for refrigerators on the market provides some insight to this question.

To this end, we analyze data from $n = 37$ refrigerators.

Instructions

```
# Pre-exercise code
Refrig <- read.table("CSVData\\Refrig.csv", header = TRUE, sep = ",")
summary(Refrig)
Refrig1 <- Refrig[c("price", "ecost", "rsize", "fsize", "shelves", "s_sq_ft", "features")]
round(cor(Refrig1), digits = 3)
refrig_mlr1 <- lm(price ~ rsize + fsize + shelves + features, data = Refrig)
summary(refrig_mlr1)
Refrig$residuals1 <- residuals(refrig_mlr1)
refrig_mlr2 <- lm(ecost ~ rsize + fsize + shelves + features, data = Refrig)
summary(refrig_mlr2)
Refrig$residuals2 <- residuals(refrig_mlr2)
plot(Refrig$residuals2, Refrig$residuals1)
```




```
#library(Rcmdr)
#refrig_mlr3 <- lm(price ~ rsize + fsize + shelves + features + ecost, data = Refrig)
#avPlots(refrig_mlr3, terms = "ecost")
```

```
     price            ecost           rsize          fsize      
 Min.   : 460.0   Min.   :60.00   Min.   :12.6   Min.   :4.100  
 1st Qu.: 545.0   1st Qu.:66.00   1st Qu.:12.9   1st Qu.:4.400  
 Median : 590.0   Median :68.00   Median :13.2   Median :5.100  
 Mean   : 626.4   Mean   :70.51   Mean   :13.4   Mean   :5.184  
 3rd Qu.: 685.0   3rd Qu.:75.00   3rd Qu.:13.9   3rd Qu.:5.700  
 Max.   :1200.0   Max.   :94.00   Max.   :14.7   Max.   :7.400  
    shelves         s_sq_ft         features    
 Min.   :1.000   Min.   :20.60   Min.   : 1.000  
 1st Qu.:2.000   1st Qu.:23.40   1st Qu.: 2.000  
 Median :2.000   Median :24.00   Median : 3.000  
 Mean   :2.514   Mean   :24.53   Mean   : 3.459  
 3rd Qu.:3.000   3rd Qu.:25.50   3rd Qu.: 5.000  
 Max.   :5.000   Max.   :30.20   Max.   :12.000  
          price  ecost  rsize  fsize shelves s_sq_ft features
price     1.000  0.522 -0.024  0.720   0.400   0.155    0.697
ecost     0.522  1.000 -0.033  0.855   0.188   0.058    0.334
rsize    -0.024 -0.033  1.000 -0.235  -0.363   0.401   -0.096
fsize     0.720  0.855 -0.235  1.000   0.251   0.110    0.439
shelves   0.400  0.188 -0.363  0.251   1.000  -0.527    0.160
s_sq_ft   0.155  0.058  0.401  0.110  -0.527   1.000    0.083
features  0.697  0.334 -0.096  0.439   0.160   0.083    1.000
```

```
Call:
lm(formula = price ~ rsize + fsize + shelves + features, data = Refrig)

Residuals:
     Min       1Q   Median       3Q      Max
-128.200  -34.963    7.081   28.716  155.096

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -698.89     302.60  -2.310  0.02752 *
rsize          56.50      20.56   2.748  0.00977 **
fsize          75.40      13.93   5.414 5.96e-06 ***
shelves        35.92      11.08   3.243  0.00277 **
features       25.16       5.04   4.992 2.04e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 68.11 on 32 degrees of freedom
Multiple R-squared:  0.789, Adjusted R-squared:  0.7626
F-statistic: 29.92 on 4 and 32 DF,  p-value: 2.102e-10


Call:
lm(formula = ecost ~ rsize + fsize + shelves + features, data = Refrig)

Residuals:
    Min      1Q  Median      3Q     Max
-7.8483 -4.3064  0.5154  2.6324  9.3596

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -14.2165    20.9365  -0.679   0.5020
rsize         2.8744     1.4225   2.021   0.0517 .
fsize         8.9085     0.9636   9.245 1.49e-10 ***
shelves       0.2895     0.7664   0.378   0.7081
features     -0.2006     0.3487  -0.575   0.5692
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.712 on 32 degrees of freedom
Multiple R-squared:  0.7637,    Adjusted R-squared:  0.7342
F-statistic: 25.86 on 4 and 32 DF,  p-value: 1.247e-09
```

4.4 Unusual observations

In this section, you learn how to:

- Compare and contrast three alternative definitions of a standardized residual
- Evaluate three alternative options for dealing with outliers
- Assess the impact of a high leverage observation

- Evaluate options for dealing with high leverage observations
- Describe the notion of influence and Cook's Distance for quantifying influence

---

4.4.1 Video

Video Overhead Details

Show Overhead A Details. Unusual observationsUnusual observations

- Regression coefficients can be expressed as (matrix) weighted averages of outcomes
  - Averages, even weighted averages can be strongly influenced by unusual observations
- Observations may be unusual in the y direction or in the X space
- For unusual in the y direction, we use a residual $e = y - \hat{y}$
  - By subtracting the fitted value $\hat{y}$, we look to the y distance from the regression plane
  - In this way, we "control" for values of explanatory variables

Show Overhead B Details. Standardized residuals

We standardize residuals so that we can focus on relationships of interest and achieve carry-over of experience from one data set to another.

Three commonly used definitions of standardize residuals are:

$$\text{(a) } \frac{e_i}{s}, \quad \text{(b) } \frac{e_i}{s\sqrt{1-h_{ii}}}, \quad \text{(c)} \frac{e_i}{s_{(i)}\sqrt{1-h_{ii}}}.$$

- First choice is simple
- Second choice, from theory, $\text{Var}(e_i) = \sigma^2(1 - h_{ii})$. Here, h_{ii} is the ith *leverage* (defined later).
- Third choice is termed "studentized residuals". Idea: numerator is independent of the denominator.

Show Overhead C Details. Outlier - an unusal standardized residual

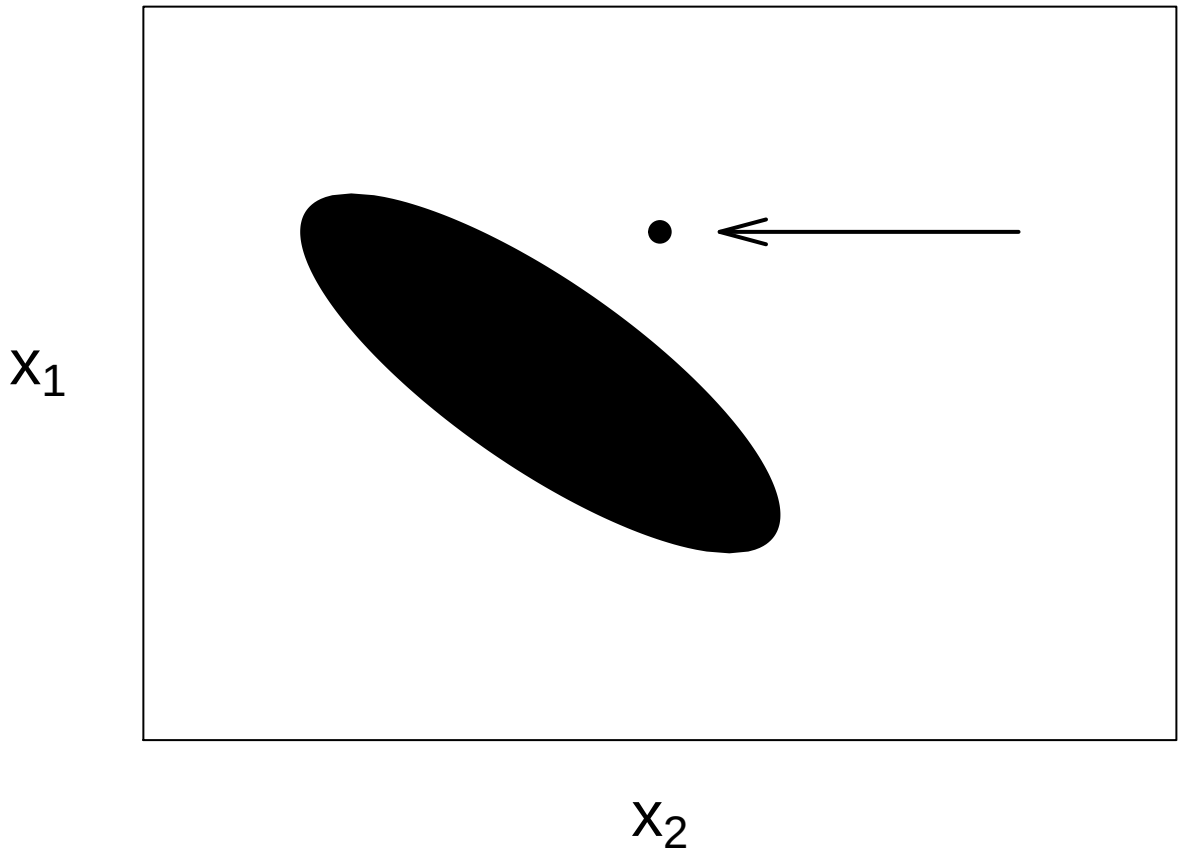
- An *outlier* is an observation that is not well fit by the model; these are observations where the residual is unusually large.
- Unusual means what? Many packages mark a point if the |standardized residual| > 2.
- Options for handling outliers
  - Ignore them in the analysis but be sure to discuss their effects.
  - Delete them from the data set (but be sure to discuss their effects).
  - Create a binary variable to indicator their presence. (This will increase your R^2!)

Show Overhead D Details. High leverage points

- A high leverage point is an observation that is "far away" in the x-space from others.
- One can get a feel for high leverage observations by looking a summary statistics (mins, maxs) for each explanatory variable.
- Options for dealing with high leverage points are comparable to outliers, we can ignore their effects, delete them, or mark them with a binary indicator variable.

Show Overhead E Details. High leverage point graph

```
library(cluster)
#library(MASS)
par(mar=c(3.2,5.4,.2,.2))
plot(1,5,type="p",pch=19,cex=1.5,xlab="",ylab="",cex.lab=1.5,xaxt="n",yaxt="n",xlim=c(-3,5),ylim=c(-12,
mtext(expression(x[2]), side=1,line=2, cex=2.0)
mtext(expression(x[1]), side=2, line=2, las=2, cex=2.0)
arrows(1.5,5,4,5,code=1,lwd=2,angle=15,length=0.25)
xycov<-matrix(c(2, -5,-5, 20),nrow=2,ncol=2)
xyloc<-matrix(c(0, 0),nrow=1,ncol=2)
polygon(ellipsoidPoints(xycov, d2 = 2, loc=xyloc),col="black")
```



Show Overhead F Details. Leverage

- Using matrix algebra, one can express the ith fitted value as a linear combination of observations

$$\hat{y}_i = h_{i1}y_1 + \cdots + h_{ii}y_i + \cdots + h_{in}y_n.$$

- The term h_{ii} is known as the ith leverage
  - The larger the value of h_{ii}, the greater the effect of the ith observation y_i on the ith fitted value $\hat{y}_i$.
  - Statistical routines have values of the leverage coded, so computing this quantity. The key thing to know is that h_{ii} is based solely on the explanatory variables. If you change the y values, the leverage does not change.
  - As a commonly used rule of thumb, a leverage is deemed to be "unusual" if its value exceeds three times the average (= number of regression coefficients divided by the number of observations.)

4.4.2 Exercise. Outlier example

In chapter 2, we consider a fictitious data set of 19 "base" points plus three different types of unusual points. In this exercise, we consider the effect of one unusal point, "C", this both an outlier (unusual in the "y" direction) and a high leverage point (usual in the x-space). The data have been pre-loaded in the dataframe `outlrC`.

Instructions

- Fit a basic linear regression model of `y` on `x` and store the result in an object.
- Use the function rstandard() to extract the standardized residuals from the fitted regression model object and summarize them.
- Use the function hatvalues() to extract the leverages from the model fitted and summarize them.
- Plot the standardized residuals versus the leverages to see the relationship between these two measures that calibrate how unusual an observation is.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNvdXRsciA8LSByZWFkLmNzdihcIkNTVkRhdGFcXFxcT3V

4.4.3 Exercise. High leverage and risk manager survey

Assignment Text

In a prior exercise, we fit a regression model of `logcost` on `logsize`, `indcost` and a squared version of `indcost`. This model is summarized in the object `mlr_survey2`. In this exercise, we examine the robustness of the model to unusual observations.

Instructions

- Use the `R` functions rstandard() and hatvalues() to extract the standardized residuals and leverages from the model fitted. Summarize the distributions graphically.
- You will see that there are two observations where the leverages are high, numbers 10 and 16. On looking at the dataset, these turn out to be observations in a high risk industry. Create a histogram of the variable `indcost` to corroborate this.
- Re-run the regression omitting observations 10 and 16. Summarize this regression and the regression in the object `mlr_survey2`, noting differences in the coefficients.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNzdXJ2ZXkgPC0gcmVhZC5jc3YoXCJDU1ZEYXRhXFxcXF

4.5 Collinearity

In this section, you learn how to:

- Define collinearity and describe its potential impact on regression inference
- Define a variance inflation factor and describe its effect on a regression coefficients standard error
- Describe rules of thumb for assessing collinearity and options for model reformulation in the presence of severe collinearity
- Compare and contrast effects of leverage and collinearity

4.5.1 Video

Video Overhead Details

Show Overhead A Details. Collinearity

- *Collinearity*, or *multicollinearity*, occurs when one explanatory variable is, or nearly is, a linear combination of the other explanatory variables.
  - Useful to think of the explanatory variables as being highly correlated with one another.
- Collinearity neither precludes us from getting good fits nor from making predictions of new observations.
  - Estimates of error variances and, therefore, tests of model adequacy, are still reliable.
- In cases of serious collinearity, standard errors of individual regression coefficients can be large.
  - With large standard errors, individual regression coefficients may not be meaningful.
  - Because a large standard error means that the corresponding t-ratio is small, it is difficult to detect the importance of a variable.

Show Overhead B Details. Quantifying collinearity

A common way to quantify collinearity is through the *variance inflation factor (VIF).*

- Suppose that the set of explanatory variables is labeled $x_1, x_2, \ldots, x_k$.
- Run the regression using x_j as the "outcome" and the other x's as the explanatory variables.
- Denote the coefficient of determination from this regression by R_j^2.
- Define the variance inflation factor

$$VIF_j = \frac{1}{1 - R_j^2}, \quad \text{for } j = 1, 2, \ldots, k.$$

Show Overhead C Details. Options for handling collinearity

- Rule of thumb: When VIF_j exceeds 10 (which is equivalent to $R_j^2 > 90\%$), we say that severe collinearity exists. This may signal is a need for action.
- Recode the variables by "centering" - that is, subtract the mean and divide by the standard deviation.
- Ignore the collinearity in the analysis but comment on it in the interpretation. Probably the most common approach.
- Replace one or more variables by auxiliary variables or transformed versions.
- Remove one or more variables. Easy. Which One? is hard.
  - Use interpretation. Which variable(s) do you feel most comfortable with?
  - Use automatic variable selection procedures to suggest a model.

4.5.2 Exercise. Collinearity and term life

Assignment Text We have seen that adding an explanatory variable x^2 to a model is sometimes helpful even though it is perfectly related to x (such as through the function $f(x) = x^2$). But, for some data sets, higher order polynomials and interactions can be approximately linearly related (depending on the range of the data).

This exercise returns to our term life data set `Term1` (preloaded) and demonstrates that collinearity can be severe when introducing interaction terms.

Instructions

- Fit a MLR model of `logface` on explantory variables `education`, `numhh` and `logincome`
- Use the function vif() from the `car` package (preloaded) to calculate variance inflation factors.

- Fit and summarize a MLR model of `logface` on explantory variables `education` , `numhh` and `logincome` with an interaction between `numhh` and `logincome`, then extract variance inflation factors.

Hint. If the `car` package is not available to you, then you could calculate vifs using the [lm()] function, treating each variable separately. For example

```
1/(1-summary(lm(education ~ numhh + logincome, data = Term1))$r.squared)
```

gives the `education` vif.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNUZXJtIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFx0Z

4.6 Selection criteria

In this section, you learn how to:

- Summarize a regression fit using alternative goodness of fit measures
- Validate a model using in-sample and out-of-sample data to mitigate issues of data-snooping
- Compare and contrast *SSPE* and *PRESS* statistics for model validation

---

4.6.1 Video

Video Overhead Details

Show Overhead A Details. Goodness of fit

- Criteria that measure the proximity of the fitted model and realized data are known as *goodness of fit* statistics.
- Basic examples include:
  - the coefficient of determination (R^2),
  - an adjusted version (R^2_a),
  - the size of the typical error (s), and
  - t-ratios for each regression coefficient.

Show Overhead B Details. Goodness of fit and information criteria

A general measure is *Akaike's Information Criterion*, defined as

$$AIC = -2 \times (fitted\ log\ likelihood) + 2 \times (number\ of\ parameters)$$

- For model comparison, the smaller the AIC, the better is the fit.
- This measures balances the fit (in the first part) with a penalty for complexity (in the second part)
- It is a general measure - for linear regression, it reduces to

$$AIC = n \ln(s^2) + n \ln(2\pi) + n + k + 3.$$

- So, selecting a model to minimize s or s^2 is equivalent to model selection based on minimizing AIC (same k).

Show Overhead C Details. Out of sample validation

- When you choose a model to minimize s or AIC, it is based on how well the model fits the data at hand, or the *model development*, or *training*, data
- As we have seen, this approach is susceptible to overfitting.
- A better approach is to validate the model on a *model validation*, or *test* data set, held out for this purpose.

Show Overhead D Details. Out of sample validation procedure

- (i) Using the model development subsample, fit a candidate model.
- (ii) Using the Step (ii) model and the explanatory variables from the validation subsample, "predict" the dependent variables in the validation subsample, $\hat{y}_i$, where $i = n_1 + 1, ..., n_1 + n_2$.
- (iii) Calc the *sum of absolute prediction errors**

$$SAPE = \sum_{i=n_1+1}^{n_1+n_2} |y_i - \hat{y}_i|.$$

Repeat Steps (i) through (iii) for each candidate model. Choose the model with the smallest $SAPE$.

Show Overhead E Details. Cross - validation

- With out-of-sample validation, the statistic depends on a random split between in-sample and out-of-sample data (a problem for data sets that are not large)
- Alternatively, one may use *cross-validation*
  - Use a random mechanism to split the data into k subsets, (e.g., 5-10)
  - Use the first *k-1* subsamples to estimate model parameters. Then, "predict" the outcomes for the kth subsample and use SAE to summarize the fit
  - Repeat this by holding out each of the k sub-samples, summarizing with a cumulative SAE.
- Repeat these steps for several candidate models.
  - Choose the model with the lowest cumulative SAE statistic.

4.6.2 Exercise. Cross-validation and term life

Assignment Text

Here is some sample code to give you a better feel for cross-validation.

The first part of the randomly re-orders ("shuffles") the data. It also identifies explanatory variables `explvars`.

The function starts by pulling out only the needed data into `cvdata`. Then, for each subsample, a model is fit based on all the data except for the subsample, in `train_mlr` with the subsample in `test`. This is repeated for each subsample, then results are summarized.

Show Code

```
# Randomly re-order data - "shuffle it"
n <- nrow(Term1)
set.seed(12347)
shuffled_Term1 <- Term1[sample(n), ]
explvars <- c("education", "numhh", "logincome")

## Cross - Validation
```

```
crossvalfct <- function(explvars){
  cvdata   <- shuffled_Term1[, c("logface", explvars)]
  crossval <- 0
  k <- 5
  for (i in 1:k) {
    indices <- (((i-1) * round((1/k)*nrow(cvdata))) + 1):((i*round((1/k) * nrow(cvdata))))
    # Exclude them from the train set
    train_mlr <- lm(logface ~ ., data = cvdata[-indices,])
    # Include them in the test set
    test  <- data.frame(cvdata[indices, explvars])
    names(test)  <- explvars
    predict_test <- exp(predict(train_mlr, test))
    # Compare predicted to held-out and summarize
    predict_err  <- exp(cvdata[indices, "logface"]) - predict_test
    crossval <- crossval + sum(abs(predict_err))
  }
  crossval/1000
}

crossvalfct(explvars)
```

Instructions

- Calculate the cross-validation statistic using only logarithmic income, `logincome`.
- Calculate the cross-validation statistic using `logincome`, `education` and `numhh`.
- Calculate the cross-validation statistic using `logincome`, `education`, `numhh` and `marstat`.

The best model has the lowest cross-validation statistic.

Hint. The function [sample()] is for taking random samples. We use it without replacement so it results in a re-ordering of data.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNUZXJtIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFx0Z

Chapter 5

Interpreting Regression Results

Chapter description

A case study, on determining an individual's characteristics that influence its health expenditures, illustrates the regression modeling process from start to finish. Subsequently, the chapter summarizes what we learn from the modeling process, underscoring the importance of variable selection.

5.1 Case study: MEPS health expenditures

5.1.1 Video

Video Overhead Details

Show Overhead A Details. MEPS health expenditures

This exercise considers data from the *Medical Expenditure Panel Survey* (MEPS), conducted by the U.S. Agency of Health Research and Quality. MEPS is a probability survey that provides nationally representative estimates of health care use, expenditures, sources of payment, and insurance coverage for the U.S. civilian population. This survey collects detailed information on individuals of each medical care episode by type of services including physician office visits, hospital emergency room visits, hospital outpatient visits, hospital inpatient stays, all other medical provider visits, and use of prescribed medicines. This detailed information allows one to develop models of health care utilization to predict future expenditures. You can learn more about MEPS at http://www.meps.ahrq.gov/mepsweb/.

We consider MEPS data from the panels 7 and 8 of 2003 that consists of 18,735 individuals between ages 18 and 65. From this sample, we took a random sample of 2,000 individuals that appear in the file `HealthExpend`. From this sample, there are 1,352 that had positive outpatient expenditures.

Our dependent variable is the amount of expenditures for outpatient visits, `expendop`. For MEPS, outpatient events include hospital outpatient department visits, office-based provider visits and emergency room visits excluding dental services. (Dental services, compared to other types of health care services, are more predictable and occur in a more regular basis.) Hospital stays with the same date of admission and discharge, known as "zero-night stays," were included in outpatient counts and expenditures. (Payments associated with emergency room visits that immediately preceded an inpatient stay were included in the inpatient expenditures. Prescribed medicines that can be linked to hospital admissions were included in inpatient expenditures, not in outpatient utilization.)

Show Overhead B Details. Overhead MEPS health expenditures

Data from the Medical Expenditure Panel Survey (MEPS), conducted by the U.S. Agency of Health Research and Quality (AHRQ).

- A probability survey that provides nationally representative estimates of health care use, expenditures, sources of payment, and insurance coverage for the U.S. civilian population.
- Collects detailed information on individuals of each medical care episode by type of services including
  - physician office visits,
  - hospital emergency room visits,
  - hospital outpatient visits,
  - hospital inpatient stays,
  - all other medical provider visits, and
  - use of prescribed medicines.
- This detailed information allows one to develop models of health care utilization to predict future expenditures.
- We consider MEPS data from the first panel of 2003 and take a random sample of $n = 2,000$ individuals between ages 18 and 65.

Show Overhead C Details. Outcome variable

Our dependent variable is expenditures for outpatient admissions.

- For MEPS, inpatient admissions include persons who were admitted to a hospital and stayed overnight.
- In contrast, outpatient events include hospital outpatient department visits, office-based provider visits and emergency room visits excluding dental services.
  - Hospital stays with the same date of admission and discharge, known as "zero-night stays," were included in outpatient counts and expenditures.
  - Payments associated with emergency room visits that immediately preceded an inpatient stay were included in the inpatient expenditures.
  - Prescribed medicines that can be linked to hospital admissions were included in inpatient expenditures, not in outpatient utilization.

Show Overhead D Details. Explanatory variables

9 variables in the database. Here 13 most relevant.

expendop	Amounts of expenditures for outpatient visits
gender	Indicate gender of patient (=1 if female, =0 if male)
age	Age in years between 18 and 65
race	Race of patient described as Asian, Black, Native, White and other
region	Region of patient described as WEST, NORTHEAST, MIDWEST and SOUTH
educ	Level of education received described by words (LHIGHSC, HIGHSCH and COLLEGE)
phstat	Self-rated physical health status described as EXCE, VGOO, GOOD, FAIR and POOR
mpoor	Self-rated mental health (=1 if poor or fair, =0 if good to excellent mental health)
anylimit	Any activity limitation (=1 if any functional/activity limitation, =0 if otherwise)
income	Income compared to poverty line described as POOR, NPOOR, LINCOME, MINCOME and HINCOME
insure	Insurance coverage (=1 if covered by public/private health insurance in any month of 1996, =0 otherwise)
usc	1 if dissatisfied with one's usual source of care
unemploy	Employment status of patients
managedcare	1 if enrolled in an HMO or gatekeeper plan

Show Overhead E Details. Case study outline

The next series of exercises leads you through an analysis of the steps for understanding a complex data set. Because of the complexity of the data, in each step only a sample of procedures will be executed.

The outline consists of:

- Summary statistics
- Splitting the data into training and testing portions with initial model fits
- Selecting variables to be included in the model

5.1.2 Exercise. Summarizing data

Assignment Text

With a complex dataset, you will probably want to take a look at the structure of the data. You are already familiar with taking a [summary()] of a dataframe which provides summary statistics for many variables. You will see that several variables in this dataframe are categorical, or factor, variables. We can use the table() function to summarize them.

After getting a sense of the distributions of explanatory variables, we want to take a deeper dive into the distribution of the outcome variable, `expendop`. We will do this by comparing the histograms of the variable to that of its logarithmic version.

To examine relationships of the outcome variable visually, we look to scatterplots for continuous variables (such as `age`) and boxplots for categorical variables (such as `phstat`).

Instructions

- Examine the structure of the `meps` dataframe using the str() function. Also, get a [summary()] of the dataframe.
- Examine the distribution of the `race` variable using the table() function.
- Compare the expenditures distribution to its logarithmic version visually via histograms plotted next to another. `par(mfrow = c(1, 2))` is used to organize the plots you create.
- Examine the distribution of logarithmic expenditures in terms of levels of `phstat` visually using the boxplot() function.
- Examine the relationship of age versus logarithmic expenditures using a scatter plot. Superimpose a local fitting line using the lines() and lowess() functions.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNtZXBzIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFxIZ

5.1.3 Exercise. Fit a benchmark multiple linear regression model

Assignment Text

As part of the pre-processing for the model fitting, we will split the data into training and test subsamples. For this exercise, we use a 75/25 split although other choices are certainly suitable. Some analysts prefer to do this splitting before looking at the data. Another approach, adopted here, is that the final report typically contains summary statistcs of the entire data set and so it makes sense to do so when examining summary statistics.

We start by fitting a benchmark model. It is common to use all available explanatory variables with the outcome on the original scale and so we use this as our benchmark model. This exercise shows that when you plot() a fitted linear regression model in `R`, the result provides four graphs that you have seen before. These can be useful for identifying an appropriate model.

Instructions

- Randomly split the data into a training and a testing data sets. Use 75% for the training, 25% for the testing.

- Fit a full model using `expendop` as the outcome and all explanatory variables. Summarize the results of this model fitting.
- You can plot() the fitted model to view several diagnostic plots. These plots provide evidence that expenditures may not be the best scale for linear regression.
- Fit a full model using `logexpend` as the outcome and all explanatory variables and summarize the fit. Use the plot() function for evidence that this variable is more suited for linear regression methods than expenditures on the original scale.

Hint. A `plot` of a regression object such as plot(mlr) provides four diagnostic plots. These can be organized as a 2 by 2 array using `par(mfrow = c(2, 2))`.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNtZXBzIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFxIZ

5.1.4 Exercise. Variable selection

Assignment Text

Modeling building can be approached using a "ground-up" strategy, where the analyst introduces a variable, examines residuls from a regression fit, and then seeks to understand the relationship between these residuals and other available variables so that these variables might be added to the model.

Another approach is a "top-down" strategy where all available variables are entered into a model and unnecessary variables are pruned from the model. Both approaches are helpful when using data to specify models. This exercise illustrates the latter approach, using the [step()] function to help narrow our search for the best fitting model.

Instructions

From our prior work, the training dataframe `train_meps` has already been loaded in. A multiple linear regression model fit object `meps_mlr2` is available that summarizes a fit of `logexpend` as the outcome variable using all 13 explanatory variables.

- Use the step() function function to drop unnecessary variables from the full fitted model summarized in the object `meps_mlr2` and summarize this recommended model.
- As an alternative, use the explanatory variables in the recommended model and add the varibles `phstat`. Summarize the fit and note that statistical significance of the new variable.
- You have been reminded by your boss that use of the variable `gender` is unsuitable for actuarial pricing purposes. As an another alternative, drop `gender` from the recommended model (still keeping `phstat`). Note the statistical significance of the variable `uscwith` this fitted model.

Hint

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNtZXBzIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFxIZ

5.1.5 Exercise. Model comparisons using cross-validation

Assignment Text

To compare alternative models, you decide to utilize cross-validation. For this exercise, you split the training sample into six subsamples of approximately equal size.

In the sample code, the cross-validation procedure has been summarized into a function that you can call. The input to the function is a list of variables that you select as your model explanatory variables. With this function, you can readily test several candidate models.

Instructions

- Run the cross validation (`crossvalfct`) function using the explanatory variables suggested by the stepwise function.

- Run the function again but adding the `mpoor` variable
- Run the function again but omitting the `gender` variable

Note which model is suggested by the cross validation function.

Hint. The cross validation function of this is very similar to the one we did earlier. Different number of subsamples, different test/training data and a different outcome variable. Except for these minor changes, it is the same function that we worked with earlier.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNtZXBzIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFxIZ

5.1.6 Exercise. Out of sample validation

Assignment Text

From our prior work, the training `train_meps` and test `test_meps` dataframes have already been loaded in. We think our best model is based on logarithmic expenditures as the outcome and the following explanatory variables:

```
explvars3 <- c("gender", "age", "phstat", "anylimit", "insure", "mpoor")
```

We will compare this to a benchmark model that is based on expenditures as the outcome and all 13 explanatory variables

```
explvars4 <- c(explvars3, "race", "income", "region", "educ", "unemploy", "managedcare", "usc")
```

The comparisons will be based on expenditures in dollars using the held-out validation sample.

Instructions

- Use the training sample to fit a linear model with `logexpend` and explanatory variables listed in `explvars3`
- Predict expenditures (not logged) for the test data and summarize the fit using the sum of absolute prediction errors.
- Use the training sample to fit a benchmark linear model with `expendop` and explanatory variables listed in `explvars4`
- Predict expenditures for the test data and summarize the fit for the benchmark model using the sum of absolute prediction errors.
- Compare the predictions of the models graphically.

eyJsYW5ndWFnZSI6InIiLCJwcmVfZXhlcmNpc2VfY29kZSI6IiNtZXBzIDwtIHJlYWQuY3N2KFwiQ1NWRGF0YVxcXFxIZ

5.2 What the modeling procedure tells us

In this section, you learn how to:

- Interpret individual effects, based on their substantive and statistical significance
- Describe other purposes of regression modeling, including regression function for pricing, benchmarking studies, and predicting future observations.

5.2.1 Video

Video Overhead Details

Show Overhead A Details. Interpreting individual effects

- Substantive Effect
  - Does a 1 unit change in x imply an economically meaningful change in y?
  - Example: Looking at urban and rural claims experience, is there a big enough difference to warrant differentiating prices by location?
- Statistical Significance
  - We have standards for deciding whether or not a variable is statistically significant.
  - A "statistically significant effect" is the result of a regression coefficient that is large relative to its standard error.
- Statistical significance is driven by
  - precision of s,
  - collinearity (VIF) and
  - sample size
- **Causal Effects**
  - If we change x, would y change?

Show Overhead B Details. Other Interpretations

- Regression function and pricing
  - The regression function is E $y = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$.
  - Think about expected claims as our baseline price for short-term insurance coverages.
- Benchmarking studies
  - In studies of CEO's salaries, who is making a lot (or a little), controlled for industry, years of experience and so forth?
  - In studies of medical claims, who are the high-cost patients?
- Prediction
  - A new patient comes in with a given set of characteristics, what can I say about his or her future medical claims?

MC Exercise. Which of the following are not important when interpreting the effects of individual variables?

Substantive significance Statistical significance The amount of effort that it took to gather the data and do the analysis Role of causality

Submit Answer

MC Exercise. Which of the following is not a potential explanation for the lack of statistical significance of an explanatory variable?

Large variation of the disturbance term High collinearity, so that the variable may be confounded with other variables The coefficient of determination, R^2, is not sufficiently large

Submit Answer

MC Exercise. Which of the following is not an important purpose of regression modeling?

Pricing of risks such as insurance contracts Benchmarking studies, to compare an observation to others Prediction Keeping a computer occupied with work

Submit Answer

5.3 The importance of variable selection

In this section, you learn how to:

- Describe the bias that can occur when omitting important variables
- Describe the principle of parsimony and reasons for adopting this approach

---

5.3.1 Video

Video Overhead Details

Show Overhead A Details. The importance of variable selection

- With too many or too few variables, s is too large an estimate of σ.
  - Prediction intervals are too large
  - Standard errors for the partial slopes are too large
- With too few or incorrect variables, we produce biased estimates of the slopes β. Thus, our predictions are biased and hence inaccurate.

Show Overhead B Details. Example. Regression using one explanatory variable

- **Too Many Variables**
  - The "true" model is $y_i = \beta_0 + \varepsilon_i$
  - We mistakenly use $y_i = \beta_0 + \beta_1 x_i^* + \varepsilon_i$
  - The prediction at a generic level x is $b_0^* + b_1^* x$.
  - It is not to hard to confirm that $Bias = \mathrm{E}(b_0^* + b_1^* x) - \mathrm{E}y = 0$.
- **Too Few Variables**
  - The "true" model is $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$.
  - We mistakenly use $y_i = \beta_0^* \varepsilon_i$.
  - Under the true model, $\overline{y} = \beta_0 + \beta_1 \overline{x} + \overline{\varepsilon}$
  - Thus, the bias is

$$Bias = \mathrm{E}\ \bar{y} - \mathrm{E}\ (\beta_0 + \beta_1 x + \varepsilon) = \mathrm{E}\ (\beta_0 + \beta_1 \bar{x} + \bar{\varepsilon}) - (\beta_0 + \beta_1 x) = \beta_1 (\bar{x} - x).$$

There is a persistent, long-term error in omitting the explanatory variable x.

Show Overhead C Details. Principle of parsimony

- The principle of parsimony, also known as *Occam's Razor*, states that when there are several possible explanations for a phenomenon, use the simplest.
  - A simpler explanation is easier to interpret.
  - Simpler models, also known as "more parsimonious" models, often do well on fitting out-of-sample data
  - Extraneous variables can cause problems of collinearity, leading to difficulty in interpreting individual coefficients.
- In contrast, in a quote often attributed to Albert Einstein, we should use "the simplest model possible, but no simpler."
  - Omitting important variables can lead to biased results, a potentially serious error.
  - Including extraneous variables decreases the degrees of freedom and increases the estimate of variability, typically of less concern in actuarial applications.

MC Exercise

MC Exercise. Which of the following is true about under- and over-fitting a model?

When we over-fit a model, estimates of regression coefficients are over-biased as is s^2, the estimate of model variance σ^2. When we over-fit a model, estimates of regression coefficients remain unbiased whereas s^2, the estimate of model variance σ^2, is over-biased. When we over-fit a model, estimates of regression coefficients remain under-biased as is s^2, the estimate of model variance σ^2. When we under-fit a model, estimates of regression coefficients remain unbiased whereas s^2, the estimate of model variance σ^2, is over-biased.

Submit Answer

MC Exercise. Which of the following is not true of Occam's Razor?

When there are several possible explanations for a phenomenon, use the simplest one. Simpler models are easier to interpret. Variables can be statistically significant but practically unimportant. Simpler models often do better for predicting out-of-sample data

Submit Answer