Advanced Econometrics II

Nonlinear Methods and Applications

Jun.-Prof. Dr. Sven Otto

Last updated: July 7, 2025

Table of contents

Organization of the Course

Advanced Econometrics II – Nonlinear Methods and Applications is a graduate-level course in regression analysis focusing on specialized econometric tools. We cover topics such as linear regression, panel data methods, causal inference, high-dimensional regression, and time series methods. Emphasis is on both theoretical understanding of the methods and practical applications using the R programming language.

Timetable

See KLIPS Lecture and KLIPS Exercises for a detailed schedule.

Note: On Wednesday, April 16, we will have a lecture instead of exercises.

Please bring your own laptop to the Wednesday exercise sessions. If you do not have a laptop available, please let me know by email.

Lecture Material

- This online script and its pdf version
- eWhiteboard
- Rscripts and additional files: sciebo folder
- More info on exam: ILIAS course

Day	Time	Lecture Hall	Session Type
Monday	14:00 - 15:30	H80 (Philosophikum)	Lecture
Wednesday	17:45 - 19:15	S82 (Philosophikum)	Exercises

Literature

The script is self-contained. To prepare well for the exam, it's a good idea to read this script.

The course is based on James H. Stock and Mark W. Watson's **Introduction to Econometrics (Fourth Edition)**. The Stock and Watson textbook is available for download: PDF by chapter (Uni Köln VPN connection required).

Further recommended textbooks are:

- *Econometric Theory and Methods*, by Russell Davidson and James G. MacKinnon. PDF.
- *Econometric Analysis of Cross Section and Panel Data*, by Jeffrey M. Wooldridge. PDF by chapter.
- *An Introduction to Statistical Learning with Applications to R (Second Edition)*, by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. PDF.
- *Causal Inference: The Mixtape*, by Scott Cunningham. Online version.
- *Mostly Harmelss Econometrics*, by J. Angrist and J. Pischke PDF by chapter.

Printed versions of the books are available from the university library.

Assessment

The course will be graded by a 90-minute exam. For detailed information please visit the ILIAS course.

Communication

Feel free to use the ILIAS Metrics Forum to discuss lecture topics and ask questions. Please let me know if you find any typos in the lecture material. Of course, you can reach me via e-mail: sven.otto@uni-koeln.de

Important Dates

Registration deadline exam 1	July 28, 2025
Exam 1	August 04, 2025
Registration deadline exam 2	September 12, 2025
Exam 2 (alternate date)	September 19, 2025

Please register for the exam on time. If you miss the registration deadline, you will not be able to take the exam (the Examinations Office is very strict about this). You only need to

take one of the two exams to complete the course. The second exam will serve as a make-up exam for those who fail the first exam or do not take the first exam.

R-Packages

To run the R code of the lecture script, you will need to install some additional packages. Here are the most important ones for this lecture:

```
install.packages(
  c("fixest", "AER", "moments", "glmnet", "urca", "caret", "neuralnet",
    "dplyr", "knitr", "tinytex", "stargazer", "scatterplot3d", "readxl", "modelsummary")
  )
```

Some further datasets are contained in my package TeachData, which is available in a GitHub repository. It can be installed using the following command:

```
install.packages("remotes")
remotes::install_github("ottosven/TeachData")
```

Part I

Basic Principles

1 Data

1.1 Data Structures

Univariate Datasets

A **univariate dataset** consists of a sequence of observations:

$$Y_1, \dots, Y_n.$$

These n observations form a **data vector**:

$$\boldsymbol{Y} = (Y_1, \dots, Y_n)'.$$

Example: Survey of six individuals on their hourly earnings. Data vector:

$$\boldsymbol{Y} = \begin{pmatrix} 10.40 \\ 18.68 \\ 12.44 \\ 54.73 \\ 24.27 \\ 24.41 \end{pmatrix}.$$

Multivariate Datasets

Typically, we have data on more than one variable, such as years of education and gender. Categorical variables are often encoded as **dummy variables**, which are binary variables. The female dummy variable is defined as:

$$D_i = \begin{cases} 1 & \text{if person } i \text{ is female,} \\ 0 & \text{otherwise.} \end{cases}$$

person	wage	education	female
1	10.40	12	0

person	wage	education	female
2	18.68	16	0
3	12.44	14	1
4	54.73	18	0
5	24.27	14	0
6	24.41	12	1

A k**-variate dataset** (or multivariate dataset) is a collection of n observations on k variables:

$$\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n.$$

The i-th vector contains the data on all k variables for individual i:

$$\boldsymbol{X}_i = (X_{i1}, \ldots, X_{ik})'.$$

Thus, X_{ij} represents the value for the j-th variable of individual i. The full k-variate dataset is structured in the $n \times k$ **data matrix** $\boldsymbol{X}$:

$$\boldsymbol{X} = \begin{pmatrix} \boldsymbol{X}_1' \\ \vdots \\ \boldsymbol{X}_n' \end{pmatrix} = \begin{pmatrix} X_{11} & \ldots & X_{1k} \\ \vdots & \ddots & \vdots \\ X_{n1} & \ldots & X_{nk} \end{pmatrix}$$

The i-th row in $\boldsymbol{X}$ corresponds to the values from $\boldsymbol{X}_i$. Since $\boldsymbol{X}_i$ is a column vector, we use the transpose notation $\boldsymbol{X}_i'$, which is a row vector.

The data matrix for our example is:

$$\boldsymbol{X} = \begin{pmatrix} 10.40 & 12 & 0 \\ 18.68 & 16 & 0 \\ 12.44 & 14 & 1 \\ 54.73 & 18 & 0 \\ 24.27 & 14 & 0 \\ 24.41 & 12 & 1 \end{pmatrix}$$

with data vectors:

$$\boldsymbol{X}_1 = \begin{pmatrix} 10.40 \\ 12 \\ 0 \end{pmatrix}$$
$$\boldsymbol{X}_2 = \begin{pmatrix} 18.68 \\ 16 \\ 0 \end{pmatrix}$$
$$\boldsymbol{X}_3 = \begin{pmatrix} 12.44 \\ 14 \\ 1 \end{pmatrix}$$
$$\vdots$$

Matrix Algebra

Vector and matrix algebra provide a compact mathematical representation of multivariate data and an efficient framework for analyzing and implementing statistical methods. We will use matrix algebra frequently throughout this course.

To refresh or enhance your knowledge of matrix algebra, consult the following resources:

Crash Course on Matrix Algebra:

matrix.svenotto.com (in particular Sections 1-3)
Section 19.1 of the Stock and Watson textbook also provides a brief overview of matrix algebra concepts.

1.2 R Programming

The best way to learn statistical methods is to program and apply them yourself. We will use the `R` programming language for implementing econometric methods and analyzing datasets. If you are just starting with `R`, it is crucial to familiarize yourself with its basics. Here's an introductory tutorial, which contains a lot of valuable resources:

Getting Started with R:

rintro.svenotto.com

The interactive R package SWIRL offers an excellent way to learn directly within the R environment. A highly recommended online book to learn R programming is Hands-On Programming with R.

One of R's greatest strengths is its vast package ecosystem developed by the statistical community. The `AER` package ("Applied Econometrics with R") provides a comprehensive collection of tools for applied econometrics.

You can install the package with the command `install.packages("AER")` and you can load it with:

```
library(AER)
```

at the beginning of your code.

1.3 Datasets in R

CASchools Dataset

Let's load the `CASchools` dataset from the `AER` package:

```
data(CASchools, package = "AER")
```

The dataset is used throughout Sections 4-8 of Stock and Watson's textbook *Introduction to Econometrics*. It was collected in 1998 and captures California school characteristics including test scores, teacher salaries, student demographics, and district-level metrics.

Variable	Description	Variable	Description
district	District identifier	lunch	% receiving free meals
school	School name	computer	Number of computers
county	County name	expenditure	Spending per student ($)
grades	Through 6th or 8th	income	District avg income ($000s)
students	Total enrollment	english	Non-native English (%)
teachers	Teaching staff	read	Average reading score
calworks	% CalWorks aid	math	Average math score

The Environment pane in RStudio's top-right corner displays all objects currently in your workspace, including the `CASchools` dataset. You can click on `CASchools` to open a table viewer and explore its contents. To get a description of the dataset, use the `?CASchools` command.

Data Frames

The `CASchools` dataset is stored as a `data.frame`, R's most common data storage class for tabular data as in the data matrix $\boldsymbol{X}$. It organizes data in the form of a table, with variables as columns and observations as rows.

```
class(CASchools)
```

```
[1] "data.frame"
```

To inspect the structure of your dataset, you can use `str()`:

```
str(CASchools)
```

```
'data.frame':   420 obs. of  14 variables:
 $ district   : chr  "75119" "61499" "61549" "61457" ...
 $ school     : chr  "Sunol Glen Unified" "Manzanita Elementary" "Thermalito Union Elementary
 $ county     : Factor w/ 45 levels "Alameda","Butte",..: 1 2 2 2 2 6 29 11 6 25 ...
 $ grades     : Factor w/ 2 levels "KK-06","KK-08": 2 2 2 2 2 2 2 2 2 1 ...
 $ students   : num  195 240 1550 243 1335 ...
 $ teachers   : num  10.9 11.1 82.9 14 71.5 ...
 $ calworks   : num  0.51 15.42 55.03 36.48 33.11 ...
 $ lunch      : num  2.04 47.92 76.32 77.05 78.43 ...
 $ computer   : num  67 101 169 85 171 25 28 66 35 0 ...
 $ expenditure: num  6385 5099 5502 7102 5236 ...
 $ income     : num  22.69 9.82 8.98 8.98 9.08 ...
 $ english    : num  0 4.58 30 0 13.86 ...
 $ read       : num  692 660 636 652 642 ...
 $ math       : num  690 662 651 644 640 ...
```

The dataset contains variables of different types: `chr` for character/text data, `Factor` for categorical data, and `num` for numeric data.

The variable `students` contains the total number of students enrolled in a school. It is the fifth variable in the dataset. To access the variable as a vector, you can type `CASchools[,5]` (the fifth column in your data matrix), `CASchools[,"students"]`, or simply `CASchools$students`.

Subsetting and Manipulation

If you want to select the variables `students` and `teachers`, you can type `CASchools[,c("students", "teachers")]`. We can define our own dataframe `mydata` that contains a selection of variables:

```
mydata = CASchools[,c("students", "teachers", "english", "income", "math", "read")]
head(mydata)
```

```
  students teachers   english    income  math  read
1      195    10.90  0.000000 22.690001 690.0 691.6
2      240    11.15  4.583333  9.824000 661.9 660.5
3     1550    82.90 30.000002  8.978000 650.9 636.3
4      243    14.00  0.000000  8.978000 643.5 651.9
5     1335    71.50 13.857677  9.080333 639.9 641.8
6      137     6.40 12.408759 10.415000 605.4 605.7
```

The `head()` function displays the first few rows of a dataset, giving you a quick preview of its content.

The pipe operator `|>` efficiently chains commands. It passes the output of one function as the input to another. For example, `mydata |> head()` gives the same output as `head(mydata)`.

A convenient alternative to select a subset of variables of your dataframe is the `select()` function from the `dplyr` package. Let's chain the `select()` and `head()` functions:

```
library(dplyr)
CASchools |> select(students, teachers, english, income, math, read) |> head()
```

```
  students teachers   english    income  math  read
1      195    10.90  0.000000 22.690001 690.0 691.6
2      240    11.15  4.583333  9.824000 661.9 660.5
3     1550    82.90 30.000002  8.978000 650.9 636.3
4      243    14.00  0.000000  8.978000 643.5 651.9
5     1335    71.50 13.857677  9.080333 639.9 641.8
6      137     6.40 12.408759 10.415000 605.4 605.7
```

Piping in R makes code more readable by allowing you to read operations from left to right in a natural order, rather than nesting functions inside each other from the inside out.

We can easily add new variables to our dataframe, for instance, the student-teacher ratio (the total number of students per teacher) and the average test score (average of the math and reading scores):

```
# compute student-teacher ratio and append it to mydata
mydata$STR = mydata$students/mydata$teachers
# compute test score and append it to mydata
mydata$score = (mydata$read + mydata$math)/2
```

The variable `english` indicates the proportion of students whose first language is not English and who may need additional support. We might be interested in the dummy variable `HiEL`, which indicates whether the proportion of English learners is above 10 percent or not:

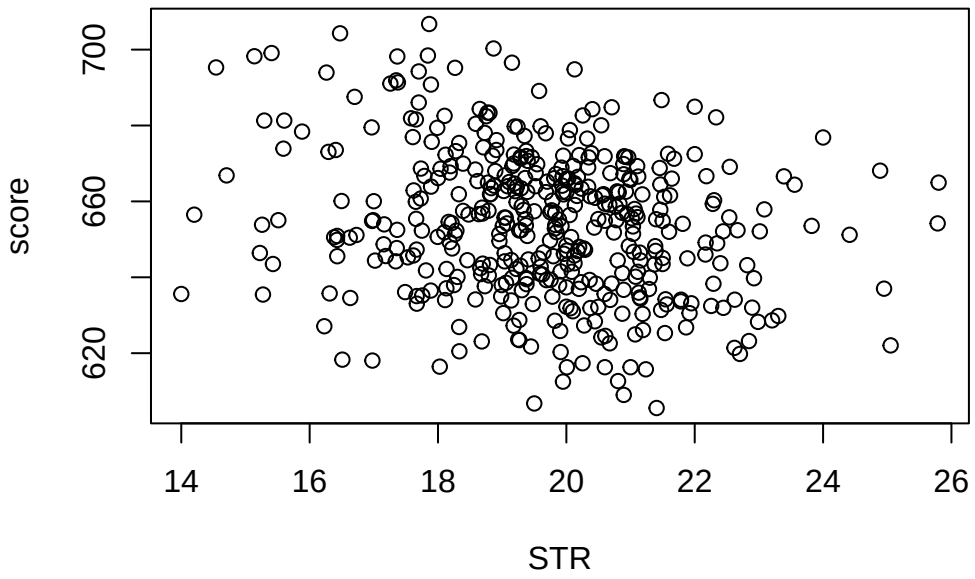
```
# append HiEL to mydata
mydata$HiEL = (mydata$english >= 10) |> as.numeric()
```

Note that `mydata$english >= 10` is a logical expression with either `TRUE` or `FALSE` values. The command `as.numeric()` creates a dummy variable by translating `TRUE` to `1` and `FALSE` to `0`.

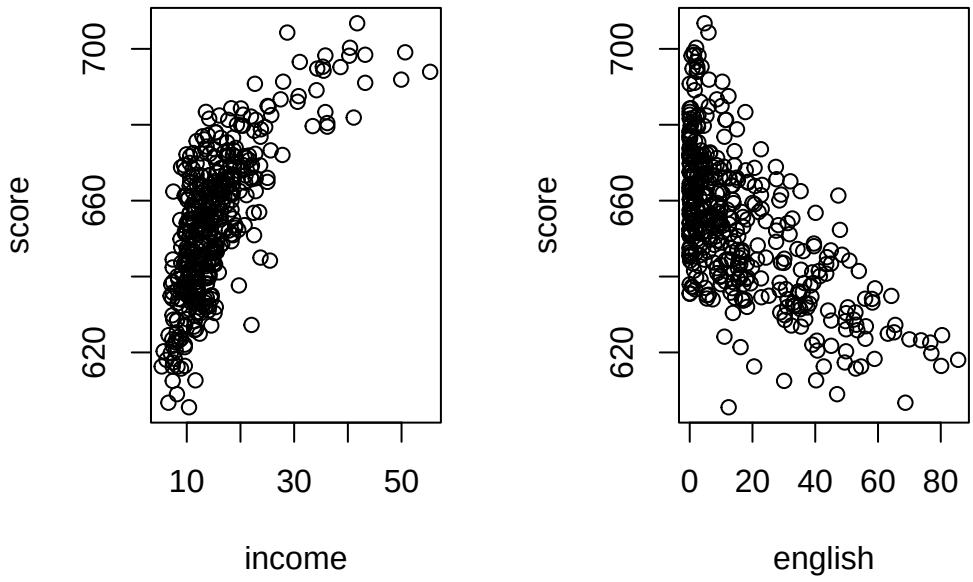
Plotting

Scatterplots provide further insights:

```
plot(score ~ STR, data = mydata)
```



```r
# Set up a plotting area with two plots side by side
par(mfrow = c(1,2))
# Scatterplots of score vs. income and score vs. english
plot(score ~ income, data = mydata)
plot(score ~ english, data = mydata)
```



The option `par(mfrow = c(1,2))` allows us to display multiple plots side by side. Try what happens if you replace `c(1,2)` with `c(2,1)`.

1.4 Importing Data

The internet serves as a vast repository for data in various formats, with `csv` (comma-separated values), `xlsx` (Microsoft Excel spreadsheets), and `txt` (text files) being the most commonly used.

R supports various functions for different data formats:

- `read.csv()` for reading comma-separated values
- `read.csv2()` for semicolon-separated values (adopting the German data convention of using the comma as the decimal mark)
- `read.table()` for whitespace-separated files
- `read_excel()` for Microsoft Excel files (requires the `readxl` package)
- `read_stata()` for STATA files (requires the `haven` package)

CPS Dataset

Let's import the CPS dataset from Bruce Hansen's textbook *Econometrics*.

The Current Population Survey (CPS) is a monthly survey conducted by the U.S. Census Bureau for the Bureau of Labor Statistics, primarily used to measure the labor force status of the U.S. population.

- Dataset: cps09mar.txt
- Description: cps09mar_description.pdf

```
url = "https://users.ssc.wisc.edu/~bhansen/econometrics/cps09mar.txt"
varnames = c("age", "female", "hisp", "education", "earnings", "hours",
             "week", "union", "uncov", "region", "race", "marital")
cps = read.table(url, col.names = varnames)
```

Let's create additional variables:

```
# wage per hour
cps$wage = cps$earnings/(cps$week * cps$hours)
# years since graduation
cps$experience = (cps$age - cps$education - 6)
# married dummy
cps$married = cps$marital %in% c(1, 2) |> as.numeric()
# Black dummy
cps$Black = (cps$race %in% c(2, 6, 10, 11, 12, 15, 16, 19)) |> as.numeric()
# Asian dummy
cps$Asian = (cps$race %in% c(4, 8, 11, 13, 14, 16, 17, 18, 19)) |> as.numeric()
```

We will need the CPS dataset later, so it is a good idea to save the dataset to your computer:

```
write.csv(cps, "cps.csv", row.names = FALSE)
```

This command saves the dataset to a file named `cps.csv` in your current working directory (you can check yours by running `getwd()`). It's best practice to use an R Project for your course work so that all files (data, scripts, outputs) are stored in a consistent and organized folder structure.

To read the data back into `R` later, just type `cps = read.csv("cps.csv")`.

1.5 Data Types

The most common types of economic data are:

- **Cross-sectional data**: Data collected on many entities at a single point in time without regard to temporal changes.
- **Time series data**: Data on a single entity collected over multiple time periods.
- **Panel data**: Data collected on multiple entities over multiple time points, combining features of both cross-sectional and time series data.

Cross-Sectional Data

The `cps` dataset is an example of a **cross-sectional** dataset, as it contains observations from various individuals at a single point in time.

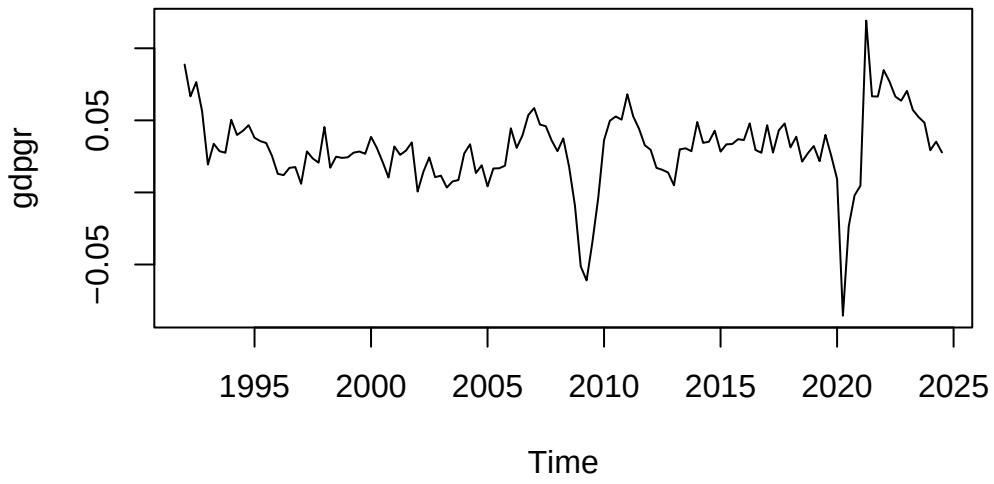
```
str(cps)
```

```
'data.frame':   50742 obs. of  20 variables:
 $ age       : int  52 38 38 41 42 66 51 49 33 52 ...
 $ female    : int  0 0 0 1 0 1 0 1 0 1 ...
 $ hisp      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ education : int  12 18 14 13 13 13 16 16 16 14 ...
 $ earnings  : int  146000 50000 32000 47000 161525 33000 37000 37000 80000 32000 ...
 $ hours     : int  45 45 40 40 50 40 44 44 40 40 ...
 $ week      : int  52 52 51 52 52 52 52 52 52 52 ...
 $ union     : int  0 0 0 0 1 0 0 0 0 0 ...
 $ uncov     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ region    : int  1 1 1 1 1 1 1 1 1 1 ...
 $ race      : int  1 1 1 1 1 1 1 1 1 1 ...
 $ marital   : int  1 1 1 1 1 5 1 1 1 1 ...
 $ experience: num  34 14 18 22 23 47 29 27 11 32 ...
 $ wage      : num  62.4 21.4 15.7 22.6 62.1 ...
 $ married   : num  1 1 1 1 1 0 1 1 1 1 ...
 $ college   : int  0 1 1 0 0 0 1 1 1 1 ...
 $ black     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ asian     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Black     : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Asian     : num  0 0 0 0 0 0 0 0 0 0 ...
```

Time Series

My repository TeachData contains several recent **time series** datasets. For instance, we can examine the annual growth rate of nominal quarterly GDP of Germany:

```
data("gdpgr", package="TeachData")
plot(gdpgr)
```



Panel Data

The dataset `Fatalities` is an example of a **panel** dataset. It contains variables related to traffic fatalities across different states (cross-sectional dimension) and years (time dimension) in the United States:

```
data(Fatalities, package = "AER")
str(Fatalities)
```

```
'data.frame':   336 obs. of  34 variables:
 $ state       : Factor w/ 48 levels "al","az","ar",..: 1 1 1 1 1 1 1 2 2 2 ...
 $ year        : Factor w/ 7 levels "1982","1983",..: 1 2 3 4 5 6 7 1 2 3 ...
 $ spirits     : num  1.37 1.36 1.32 1.28 1.23 ...
 $ unemp       : num  14.4 13.7 11.1 8.9 9.8 ...
 $ income      : num  10544 10733 11109 11333 11662 ...
 $ emppop      : num  50.7 52.1 54.2 55.3 56.5 ...
 $ beertax     : num  1.54 1.79 1.71 1.65 1.61 ...
 $ baptist     : num  30.4 30.3 30.3 30.3 30.3 ...
 $ mormon      : num  0.328 0.343 0.359 0.376 0.393 ...
```

```
$ drinkage    : num  19 19 19 19.7 21 ...
$ dry         : num  25 23 24 23.6 23.5 ...
$ youngdrivers: num  0.212 0.211 0.211 0.211 0.213 ...
$ miles       : num  7234 7836 8263 8727 8953 ...
$ breath      : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
$ jail        : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 2 2 2 ...
$ service     : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 2 2 2 ...
$ fatal       : int  839 930 932 882 1081 1110 1023 724 675 869 ...
$ nfatal      : int  146 154 165 146 172 181 139 131 112 149 ...
$ sfatal      : int  99 98 94 98 119 114 89 76 60 81 ...
$ fatal1517   : int  53 71 49 66 82 94 66 40 40 51 ...
$ nfatal1517  : int  9 8 7 9 10 11 8 7 7 8 ...
$ fatal1820   : int  99 108 103 100 120 127 105 81 83 118 ...
$ nfatal1820  : int  34 26 25 23 23 31 24 16 19 34 ...
$ fatal2124   : int  120 124 118 114 119 138 123 96 80 123 ...
$ nfatal2124  : int  32 35 34 45 29 30 25 36 17 33 ...
$ afatal      : num  309 342 305 277 361 ...
$ pop         : num  3942002 3960008 3988992 4021008 4049994 ...
$ pop1517     : num  209000 202000 197000 195000 204000 ...
$ pop1820     : num  221553 219125 216724 214349 212000 ...
$ pop2124     : num  290000 290000 288000 284000 263000 ...
$ milestot    : num  28516 31032 32961 35091 36259 ...
$ unempus     : num  9.7 9.6 7.5 7.2 7 ...
$ emppopus    : num  57.8 57.9 59.5 60.1 60.7 ...
$ gsp         : num  -0.0221 0.0466 0.0628 0.0275 0.0321 ...
```

1.6 Statistical Framework

Data is usually the result of a random experiment. The gender of the next person you meet, the daily fluctuation of a stock price, the monthly music streams of your favorite artist, the annual number of pizzas consumed - all of this information involves a certain amount of randomness.

Random Variables

In statistical sciences, we interpret a univariate dataset $Y_1, \dots, Y_n$ as a sequence of random variables. Similarly, a multivariate dataset $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ is viewed as a sequence of random vectors.

Cross-sectional data is typically characterized by an **identical distribution** across its individual observations, meaning each element in the sequence $Y_1, \dots, Y_n$ or $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ has the same distribution function.

For example, if $Y_1, \dots, Y_n$ represent the wage levels of different individuals in Germany, each Y_i is drawn from the same distribution F, which in this context is the wage distribution across the country.

Similarly, if $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ are bivariate random variables containing wages and years of education for individuals, each $\boldsymbol{X}_i$ follows the same bivariate distribution G, which is the joint distribution of wages and education levels.

Probability Theory

A primary goal of econometric methods and statistical inference is to gain insights about features of these true but unknown population distributions F or G using the available data.

Thus, a solid knowledge of probability theory is essential for econometric modeling. For a comprehensive recap on probability theory for econometricians, consider the following refresher:

Probability Theory for Econometricians:

probability.svenotto.com/
Section 2 of the Stock and Watson book also provides a review of the most important concepts.

Random Sampling

Econometric methods require specific assumptions about sampling processes. The ideal approach is simple random sampling, where each individual has an equal chance of being selected independently. This produces observations that are both identically distributed and independently drawn - what we call **independent and identically distributed (i.i.d.)** random variables or simply a **random sample**.

i.i.d. Sample

An independently and identically distributed (i.i.d.) sample, or random sample, consists of a sequence of k-variate random vectors $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ that:

1. Have the same probability distribution F (identically distributed), where $F(\boldsymbol{a}) = P(\boldsymbol{X}_i \leq \boldsymbol{a})$ for any i and $\boldsymbol{a} \in \mathbb{R}^k$
2. Are mutually independent, meaning their joint cumulative distribution function $F_{\boldsymbol{X}_1,\dots,\boldsymbol{X}_n}(\boldsymbol{a}_1, \dots, \boldsymbol{a}_n) = P(\boldsymbol{X}_1 \leq \boldsymbol{a}_1, \dots, \boldsymbol{X}_n \leq \boldsymbol{a}_n)$ factorizes completely:

$$F_{\boldsymbol{X}_1,\dots,\boldsymbol{X}_n}(\boldsymbol{a}_1, \dots, \boldsymbol{a}_n) = F(\boldsymbol{a}_1) \cdot F(\boldsymbol{a}_2) \cdot \dots \cdot F(\boldsymbol{a}_n)$$

for all $\boldsymbol{a}_1, \dots, \boldsymbol{a}_n \in \mathbb{R}^k$.

F is called the *population distribution* or *data-generating process (DGP)*.

An equivalent representation of the i.i.d. property can be obtained using the conditional distribution function $F_{\boldsymbol{X}_i|\boldsymbol{X}_j=\boldsymbol{a}_j}(\boldsymbol{a}_i) = P(\boldsymbol{X}_i \leq \boldsymbol{a}_i|\boldsymbol{X}_j = \boldsymbol{a}_j, j \neq i)$. Then, $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$ are i.i.d. if the conditional distributions equal the marginal distributions:

$$F_{\boldsymbol{X}_i|\boldsymbol{X}_j=\boldsymbol{a}_j}(\boldsymbol{a}_i) = F_{\boldsymbol{X}_i}(\boldsymbol{a}_i) = F(\boldsymbol{a}_i) \quad \text{for all } i \text{ and } \boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{R}^k.$$

For more details on independence see Probability Tutorial Part 1

The Current Population Survey (CPS) involves random interviews with individuals from the U.S. labor force and may be regarded as an i.i.d. sample. Methods that commonly yield i.i.d. sampling for economic cross-sectional datasets include:

- Survey sampling with appropriate randomization
- Administrative records with random selection
- Direct observation of randomly chosen subjects
- Web scraping with randomized targets
- Field or laboratory experiments with random assignment

In a random sample there is no inherent ordering that would introduce systematic dependencies between observations. If individuals i and j are truly randomly selected, then the observations $\boldsymbol{X}_i$ and $\boldsymbol{X}_j$ are independent random vectors. The order in which the observations appear in the dataset is arbitrary and carries no information.

Clustered Sampling

While simple random sampling provides a clean theoretical foundation, real-world data often exhibits clustering - where observations are naturally grouped or nested within larger units. This clustering leads to dependencies that violate the i.i.d. assumption in two important contexts:

In cross-sectional studies, clustering occurs when we collect data on individual units that belong to distinct groups. Consider a study on student achievement where researchers randomly select schools, then collect data from all students within those schools:

- Although schools might be selected independently, observations at the student level are dependent
- Students within the same school share common environments (facilities, resources, administration)

- They experience similar teaching quality and educational policies and they influence each other through peer effects and social interactions

For instance, if School A has an exceptional mathematics department, all students from that school may perform better in math tests compared to students with similar abilities in other schools.

Statistically, if Y_{ik} represents the test score of student k in school i:

- observations Y_{ik} and Y_{jl} are independent for $i \neq j$ (different students in different schools),
- observations Y_{ik} and $Y_{il}$are dependent (different students in the same school).

Panel Data Clustering

Panel data, by its very nature, introduces clustering across both cross-sectional units and time. Recall the `Fatalities` dataset which tracks traffic fatalities across different states and years.

For panel data with n states observed over T years, we can represent the structure as:

- The vectors $(Y_{i1}, \dots, Y_{iT})$ are i.i.d. across units $i = 1, \dots, n$ (different states' time series are independently sampled)
- But within each state i, the observations $Y_{i1}, \dots, Y_{iT}$ are generally not independent from each other

This structure reflects two important aspects of panel data:

- **Unit independence**: The complete time series for each state can be treated as an independent draw from the population distribution of all possible state time series
- **Temporal dependence**: Within each state, observations across different years are dependent due to persistent state-specific factors like road infrastructure, driving culture, and enforcement practices

For instance, if California implements effective traffic safety measures, the effects will likely persist across multiple years, creating a temporal correlation in that state's fatality rates. Similarly, economic downturns or changes in federal transportation policy may create dependencies across all states in particular years.

Time Dependence

Time series and panel data are intrinsically not independent due to the sequential nature of the observations. We usually expect observations close in time to be strongly dependent and observations at greater temporal distances to be less dependent.

Consider the quarterly GDP growth rates for Germany in the dataset `gdpgr`. Unlike cross-sectional data where the ordering of observations is arbitrary, the chronological ordering in time series carries crucial information about the dependency structure.

A simple way to formalize this temporal dependence is using an autoregression. If Y_t denotes the GDP growth at time t, a first-order autoregressive representation can be written as:

$$Y_t = \phi_0 + \phi_1 Y_{t-1} + \varepsilon_t$$

where ϕ_0 is a constant, ϕ_1 captures the persistence from one period to the next, and ε_t is a random disturbance.

If $\phi_1 \neq 0$, the current value Y_t directly depends on its previous value Y_{t-1}. For GDP growth, ϕ_1 is typically positive, indicating that strong growth in one quarter predicts stronger growth in the next quarter.

This time dependence means that the conditional distribution function differs from the marginal distribution:

$$F_{Y_t|Y_{t-1},Y_{t-2},\ldots}(y_t|y_{t-1}, y_{t-2}, \ldots) \neq F_{Y_t}(y_t)$$

In contrast to the i.i.d. case, where $F_{Y_i|Y_j}(y_i|y_j) = F_{Y_i}(y_i)$ for $i \neq j$, time series observations violate this independence property, making the i.i.d. assumption inappropriate for time series analysis.

1.7 R-codes

metrics-sec01.R

2 Summary Statistics

In statistics, a univariate dataset $Y_1, \dots, Y_n$ or a multivariate dataset $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ is often called a **sample**. It typically represents observations collected from a larger population. The sample distribution indicates how the sample values are distributed across possible outcomes.

Summary statistics, such as the sample mean and sample variance, provide a concise representation of key characteristics of the sample distribution. These summary statistics are related to the **sample moments** of a dataset.

2.1 Sample moments

The r**-th sample moment** about the origin (also called the r-th raw moment) is defined as

$$\overline{Y^r} = \frac{1}{n}\sum_{i=1}^{n} Y_i^r.$$

Mean

For example, the first sample moment ($r = 1$) is the **sample mean** (arithmetic mean):

$$\overline{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i.$$

The sample mean is the most common measure of central tendency. In i.i.d. samples, it converges in probability to the expected value as sample size grows (law of large numbers). This makes it a consistent estimator for the population mean:

$$\overline{Y} \xrightarrow{p} \mu = E[Y] \quad \text{as } n \to \infty.$$

To compute the sample mean of a vector `Y` in R, use `mean(Y)` or alternatively `sum(Y)/length(Y)`. The r-th sample moment can be calculated with `mean(Y^r)`.

2.2 Central sample moments

The r**-th central sample moment** is the average of the r-th powers of the deviations from the sample mean:

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y})^r$$

Variance

For example, the second central moment ($r = 2$) is the **sample variance**:

$$\hat{\sigma}_Y^2 = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y})^2 = \overline{Y^2} - \overline{Y}^2.$$

The sample variance measures the spread or dispersion of the data around the sample mean. It is a consistent estimator for the population variance

$$\sigma^2 = Var(Y) = E[(Y - E[Y])^2] = E[Y^2] - E[Y]^2$$

if the sample is i.i.d.

Standard Deviation

The **sample standard deviation** is the square root of the sample variance:

$$\hat{\sigma}_Y = \sqrt{\hat{\sigma}_Y^2} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y})^2} = \sqrt{\overline{Y^2} - \overline{Y}^2}$$

It quantifies the typical deviation of data points from the sample mean in the original units of measurement. It is a consistent estimator for the population standard deviation

$$sd(Y) = \sqrt{Var(Y)}.$$

2.3 Adjustments

Degrees of Freedom

When computing the sample mean $\overline{Y}$, we have n degrees of freedom because all data points $Y_1, \ldots, Y_n$ can vary freely.

When computing variances, we take the sample mean of the squared deviations

$$(Y_1 - \overline{Y})^2, \ldots, (Y_n - \overline{Y})^2.$$

These elements cannot vary freely because $\overline{Y}$ is computed from the same sample and implies the constraint

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y}) = 0.$$

This means that the deviations are connected by this equation and are not all free to vary. Knowing the first $n-1$ of the deviations determines the last one:

$$(Y_n - \overline{Y}) = -\sum_{i=1}^{n-1}(Y_i - \overline{Y}).$$

Therefore, only $n-1$ deviations can vary freely, which results in $n-1$ degrees of freedom for the sample variance.

Adjusted Sample Variance

Because $\sum_{i=1}^{n}(Y_i - \overline{Y})^2$ effectively contains only $n-1$ freely varying summands, it is common to account for this fact. The **adjusted sample variance** uses $n-1$ in the denominator:

$$s_Y^2 = \frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \overline{Y})^2.$$

The adjusted sample variance relates to the unadjusted sample variance as:

$$s_Y^2 = \frac{n}{n-1}\hat{\sigma}_Y^2.$$

The adjusted sample standard deviation is:

$$s_Y = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \overline{Y})^2} = \sqrt{\frac{n}{n-1}}\hat{\sigma}_Y.$$

To compute the sample variance and sample standard deviation of a vector `Y` in R, use `mean(Y^2)-mean(Y)^2` and `sqrt(mean(Y^2)-mean(Y)^2)`, respectively. The built-in functions `var(Y)` and `sd(Y)` compute their adjusted versions.

Let's compute the sample means, sample variances, and adjusted sample variances of some variables from the `cps` dataset.

```r
cps = read.csv("cps.csv")
exper = cps$experience
wage = cps$wage
edu = cps$education
fem = cps$female
```

```r
## Sample mean
c(mean(exper), mean(wage), mean(edu), mean(fem))
```

```
[1] 22.2071065 23.9026619 13.9246187  0.4257223
```

```r
## Sample variance
c(mean(exper^2)- mean(exper)^2, mean(wage^2) - mean(wage)^2,
  mean(edu^2) - mean(edu)^2, mean(fem^2) - mean(fem)^2)
```

```
[1] 136.1098206 428.9398785   7.5318408   0.2444828
```

```r
## Adjusted sample variance
c(var(exper), var(wage), var(edu), var(fem))
```

```
[1] 136.1125031 428.9483320   7.5319892   0.2444876
```

While the unadjusted version (using n in the denominator) yields a lower variance, it remains biased in finite samples. In contrast, the adjusted version (using $n-1$) eliminates this bias at the expense of slightly higher variance, illustrating a bias–variance tradeoff. In large samples, however, the difference becomes negligible and both estimators yield practically the same results.

2.4 Density estimation

A continuous random variable Y is characterized by a continuously differentiable CDF

$$F(a) = P(Y \leq a).$$

The derivative is known as the probability density function (PDF), defined as

$$f(a) = F'(a).$$

There are several methods to estimate this density function from sample data.

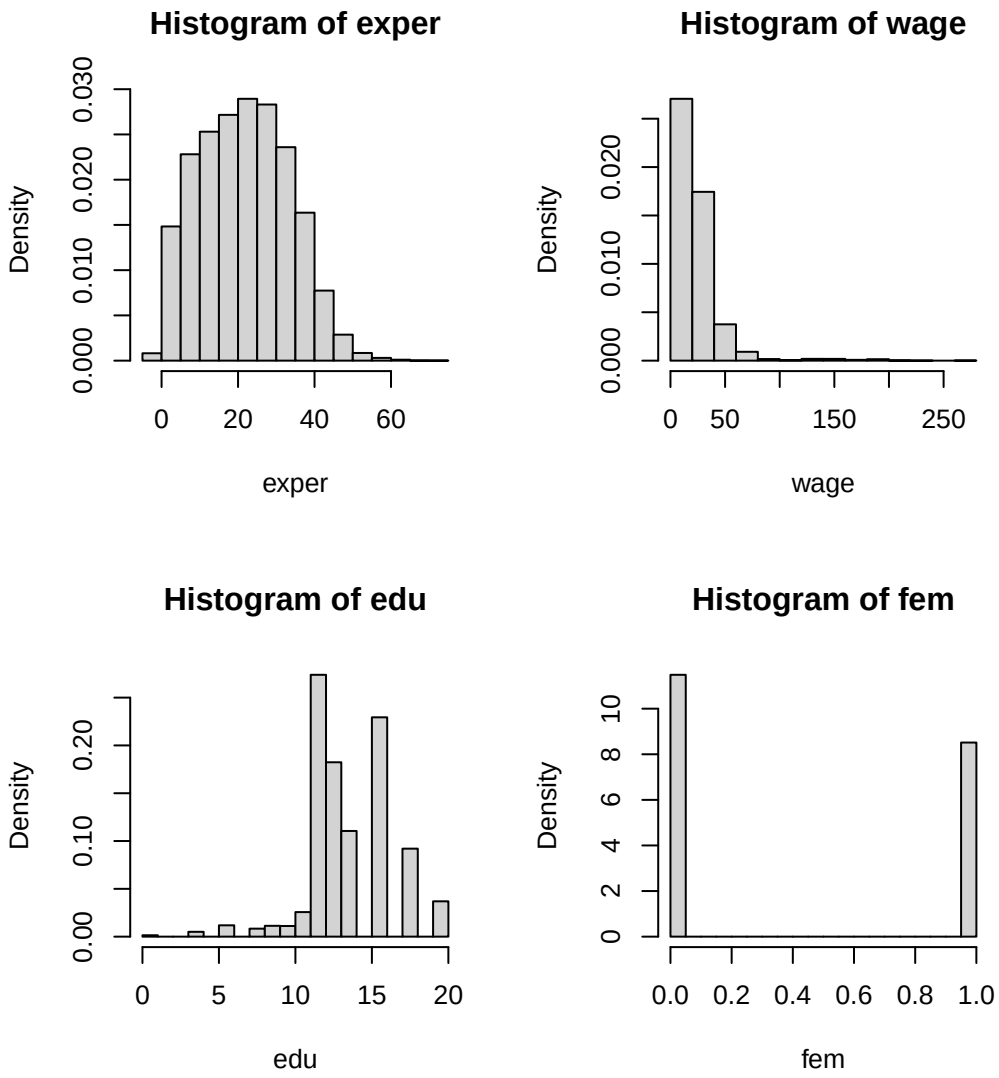
Histogram

Histograms offer an intuitive visual representation of the sample distribution of a variable. A histogram divides the data range into B bins, each of equal width h, and counts the number of observations n_j within each bin. The height of the histogram at a in the j-th bin is

$$\hat{f}(a) = \frac{n_j}{nh}.$$

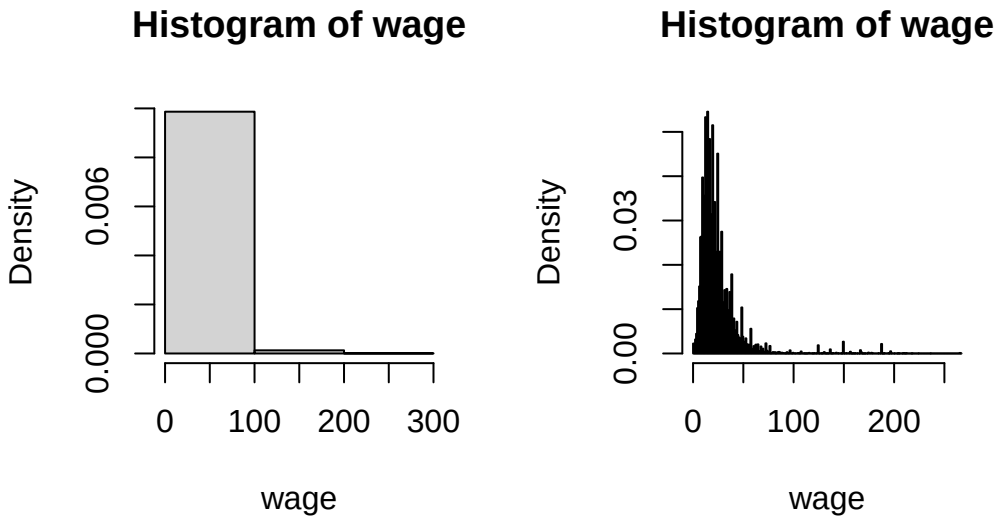
The histogram is the plot of these heights, displayed as rectangles, with their area normalized so that the total area equals 1.

```
par(mfrow = c(2,2))
hist(exper, probability = TRUE)
hist(wage, probability = TRUE)
hist(edu, probability = TRUE)
hist(fem, probability = TRUE)
```



Running `hist(wage, probability=TRUE)` automatically selects a suitable number of bins B. Note that `hist(wage)` will plot absolute frequencies instead of relative ones. The shape of a histogram depends on the choice of B. You can experiment with different values using the `breaks` option:

```
par(mfrow = c(1,2))
hist(wage, probability = TRUE, breaks = 3)
hist(wage, probability = TRUE, breaks = 300)
```



Kernel density estimator

Suppose we want to estimate the wage density at $a = 22$ and consider the histogram density estimate with $h = 10$. It is based on the frequency of observations in the interval $[20, 30)$ which is a skewed window about $a = 22$.

It seems more sensible to center the window at 22, for example $[17, 27)$ instead of $[20, 30)$. It also seems sensible to give more weight to observations close to 22 and less to those at the edge of the window.

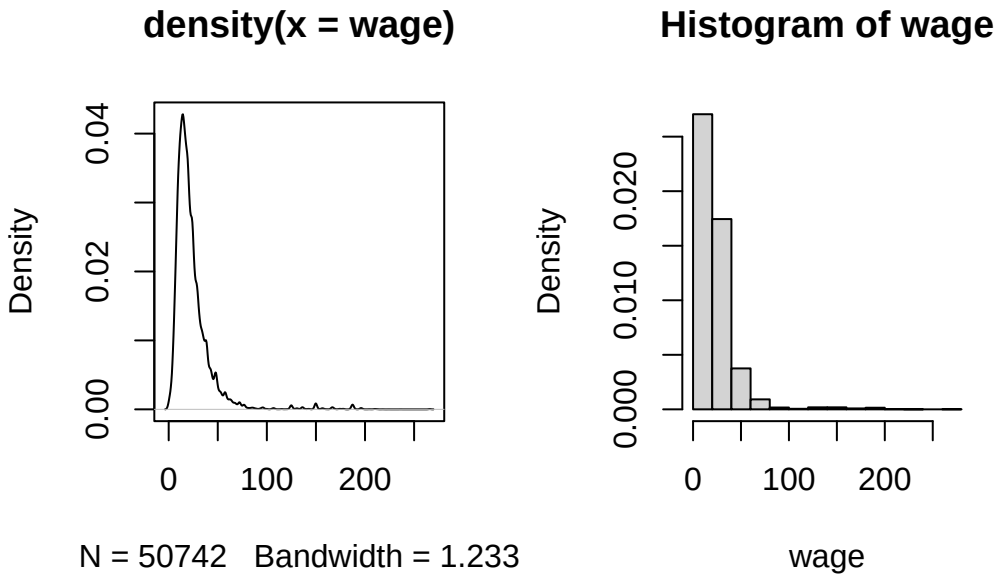
This idea leads to the **kernel density estimator** of $f(a)$, which is a smooth version of the histogram:

$$\hat{f}(a) = \frac{1}{nh} \sum_{i=1}^{n} K\Big(\frac{X_i - a}{h}\Big).$$

Here, $K(u)$ represents a weighting function known as a kernel function, and $h > 0$ is the **bandwidth**. A common choice for $K(u)$ is the Gaussian kernel:

$$K(u) = \phi(u) = \frac{1}{\sqrt{2\pi}} \exp(-u^2/2).$$

```
par(mfrow = c(1,2))
plot(density(wage))
hist(wage, probability=TRUE)
```



The `density()` function in R automatically selects an optimal bandwidth, but it also allows for manual bandwidth specification via `density(wage, bw = your_bandwidth)`.

2.5 Higher Moments

The **r-th standardized sample moment** is the central moment normalized by the sample standard deviation raised to the power of r. It is defined as:

$$\frac{1}{n}\sum_{i=1}^{n}\left(\frac{Y_i-\overline{Y}}{\hat{\sigma}_Y}\right)^r$$

Skewness

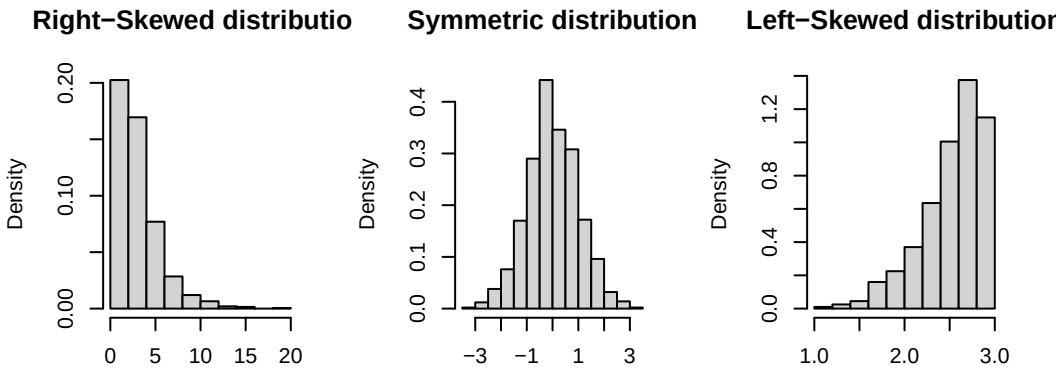
For example, the third standardized sample moment ($r=3$) is the **sample skewness**:

$$\widehat{\text{ske}}(Y)=\frac{1}{n\hat{\sigma}_Y^3}\sum_{i=1}^{n}(Y_i-\overline{Y})^3.$$

The skewness is a measure of asymmetry around the mean. A positive skewness indicates that the distribution has a longer or heavier tail on the right side (right-skewed), while a negative skewness indicates a longer or heavier tail on the left side (left-skewed). A perfectly symmetric distribution, such as the normal distribution, has a skewness of 0.

For i.i.d. samples, the sample skewness is a consistent estimator for the population skewness

$$ske(Y)=\frac{E[(Y-E[Y])^3]}{sd(Y)^3}.$$



To compute the sample skewness in R, use:

```
mean((Y-mean(Y))^3)/(mean(Y^2)-mean(Y)^2)^(3/2)
```

For convenience, you can use the `skewness(Y)` function from the `moments` package, which performs the same calculation.

```
library(moments)
c(skewness(exper), skewness(wage), skewness(edu), skewness(fem))
```

```
[1]  0.1862605  4.3201570 -0.2253251  0.3004446
```

Wages are right-skewed because a few very rich individuals earn much more than the many with low to medium incomes. The other variables do not indicate any pronounced skewness.

Kurtosis

The **sample kurtosis** is the fourth standardized sample moment ($r = 4$), commonly denoted as g_2:

$$\widehat{\text{kur}}(Y) = \frac{1}{n\hat{\sigma}_Y^4} \sum_{i=1}^{n} (Y_i - \overline{Y})^4.$$

Kurtosis measures the "tailedness" or heaviness of the tails of a distribution and can indicate the presence of extreme outliers. The reference value of kurtosis is 3, which corresponds to the kurtosis of a normal distribution. Values greater than 3 suggest heavier tails, while values less than 3 indicate lighter tails.

For i.i.d. samples, the sample kurtosis is a consistent estimator for the population kurtosis

$$kur(Y) = \frac{E[(Y - E[Y])^4]}{Var(Y)^2}.$$

To compute the sample kurtosis in R, use:

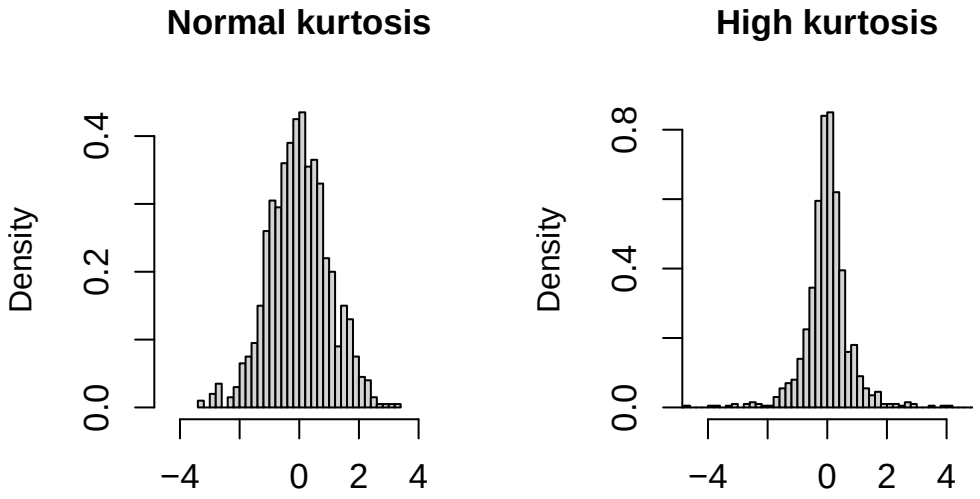
```
mean((Y-mean(Y))^4)/(mean((Y-mean(Y))^2))^2
```

For convenience, you can use the `kurtosis(Y)` function from the `moments` package, which performs the same calculation.

```
c(kurtosis(exper), kurtosis(wage), kurtosis(edu), kurtosis(fem))
```

```
[1]  2.374758 30.370331  4.498264  1.090267
```

The variable `wage` exhibits heavy tails due to a few super-rich outliers in the sample. In contrast, `fem` has light tails because there are approximately equal numbers of women and men.



The plots display histograms of two standardized datasets (both have a sample mean of 0 and a sample variance of 1). The left dataset has a normal sample kurtosis (around 3), while the right dataset has a high sample kurtosis with heavier tails.

Kurtosis not only measures the heaviness of a distribution's tails but also its peakedness. A high kurtosis indicates that data are more concentrated around the mean and in the extremes, meaning that extreme values occur more frequently than they would in a normal distribution.

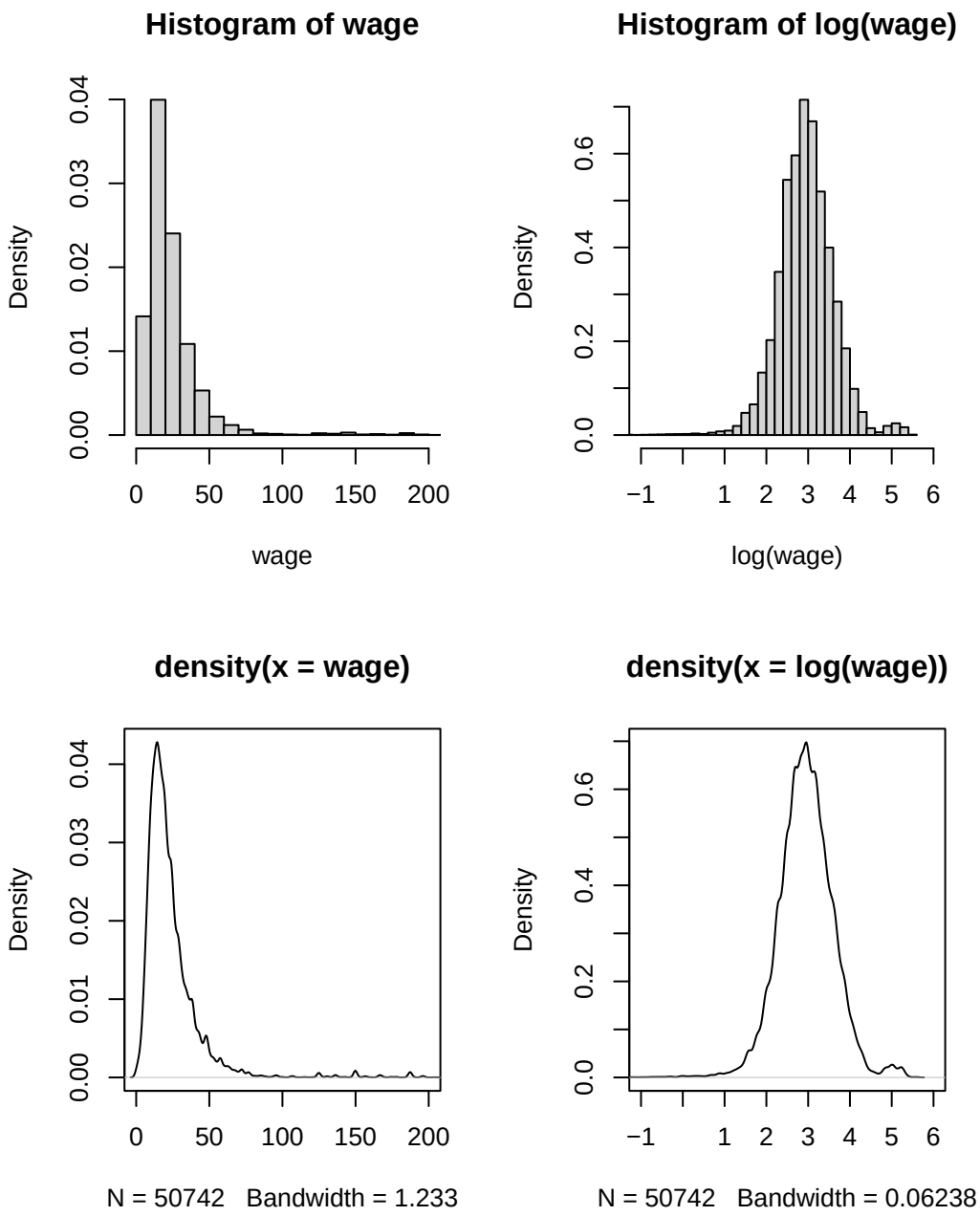
In contrast, a low kurtosis signifies a flatter peak with lighter tails, suggesting fewer extreme observations. In finance and risk management, these differences are crucial because they affect the probability of rare but impactful events.

Some statistical software reports the **excess kurtosis**, which is defined as $\widehat{kur} - 3$. This shifts the reference value to 0 (instead of 3), making it easier to interpret: positive values indicate heavier tails than the normal distribution, while negative values indicate lighter tails. For example, the normal distribution has an excess kurtosis of 0.

2.6 Logarithmic Transformations

Right-skewed, heavy-tailed variables are common in real-world datasets, such as income levels, wealth accumulation, property values, insurance claims, and social media follower counts. A common transformation to reduce skewness and kurtosis in data is to use the natural logarithm:

```
par(mfrow = c(2,2))
hist(wage, probability = TRUE, breaks = 20, xlim = c(0,200))
hist(log(wage), probability = TRUE, breaks = 50, xlim = c(-1, 6))
plot(density(wage), xlim = c(0,200))
plot(density(log(wage)), xlim = c(-1, 6))
```




```
c(skewness(wage), kurtosis(wage))
```

```
[1]  4.320157 30.370331
```

```r
c(skewness(log(wage)), kurtosis(log(wage)))
```

```
[1] -0.6990539 11.8566367
```

In econometrics, statistics, and many programming languages including R, $\log(\cdot)$ is commonly used to denote the natural logarithm (base e).

Note: On a pocket calculator, use **LN** to calculate the natural logarithm $\log(\cdot) = \log_e(\cdot)$. If you use **LOG**, you will calculate the logarithm with base 10, i.e., $\log_{10}(\cdot)$, which will give you a different result. The relationship between these logarithms is $\log_{10}(x) = \log_e(x)/\log_e(10)$.

2.7 Bivariate Statistics

For a bivariate sample $(Y_1, Z_1), \ldots, (Y_n, Z_n)$, we can compute cross moments that describe the relationship between the two variables. The (r, s)-th sample cross moment is defined as:

$$\overline{Y^r Z^s} = \frac{1}{n}\sum_{i=1}^{n} Y_i^r Z_i^s.$$

The most important cross moment is the $(1, 1)$-th sample cross moment, or simply the **first sample cross moment**:

$$\overline{YZ} = \frac{1}{n}\sum_{i=1}^{n} Y_i Z_i.$$

The central sample cross moments are defined as:

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y})^r (Z_i - \overline{Z})^s.$$

Covariance and Correlation

The $(1, 1)$-th central sample cross moment leads to the **sample covariance**:

$$\hat{\sigma}_{YZ} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y})(Z_i - \overline{Z}) = \overline{YZ} - \overline{Y} \cdot \overline{Z}.$$

Similar to the univariate case, we can define the **adjusted sample covariance**:

$$s_{YZ} = \frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \overline{Y})(Z_i - \overline{Z}) = \frac{n}{n-1}\hat{\sigma}_{YZ}.$$

The **sample correlation coefficient** is the standardized sample covariance:

$$r_{YZ} = \frac{s_{YZ}}{s_Y s_Z} = \frac{\sum_{i=1}^n (Y_i - \overline{Y})(Z_i - \overline{Z})}{\sqrt{\sum_{i=1}^n (Y_i - \overline{Y})^2}\sqrt{\sum_{i=1}^n (Z_i - \overline{Z})^2}} = \frac{\hat{\sigma}_{YZ}}{\hat{\sigma}_Y \hat{\sigma}_Z}.$$

If the sample is i.i.d., both $\hat{\sigma}_{YZ}$ and s_{YZ} are consistent estimators for the population covariance

$$\sigma_{YZ} = Cov(Y, Z) = E[(Y - E[Y])(Z - E[Z])].$$

The adjusted sample covariance s_{YZ} is unbiased, while $\hat{\sigma}_{YZ}$ is biased but has a lower sampling variance. Similarly, the sample correlation coefficient is a consistent estimator for the population correlation coefficient

$$\rho_{YZ} = Corr(Y, Z) = \frac{Cov(Y, Z)}{\sqrt{Var(Y)Var(Z)}}.$$

To compute these quantities for a bivariate sample collected in the vectors `Y` and `Z`, use `cov(Y,Z)` for the adjusted sample covariance and `cor(Y,Z)` for the sample correlation.

```
cov(wage, edu)
```

```
[1] 21.82614
```

```
cor(wage, edu)
```

```
[1] 0.3839897
```

2.8 Moment Matrices

Consider a multivariate dataset $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$, such as the following subset of the `cps` dataset:

```
dat = data.frame(wage, edu, fem)
```

Mean Vector

The **sample mean vector** $\overline{\boldsymbol{X}}$ contains the sample means of the k variables and is defined as

$$\overline{\boldsymbol{X}} = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{X}_i.$$

For i.i.d. samples, the sample mean vector is a consistent estimator for the population mean vector $E[\boldsymbol{X}]$.

```
colMeans(dat)
```

```
      wage        edu        fem 
23.9026619 13.9246187  0.4257223 
```

Covariance Matrix

The **sample covariance matrix** $\widehat{\Sigma}$ is the $k \times k$ matrix given by

$$\widehat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{X}_i - \overline{\boldsymbol{X}})(\boldsymbol{X}_i - \overline{\boldsymbol{X}})'.$$

Its elements $\hat{\sigma}_{h,l}$ represent the pairwise **sample covariance** between variables h and l:

$$\hat{\sigma}_{h,l} = \frac{1}{n}\sum_{i=1}^{n}(X_{ih} - \overline{X_h})(X_{il} - \overline{X_l}), \quad \overline{X_h} = \frac{1}{n}\sum_{i=1}^{n}X_{ih}.$$

The **adjusted sample covariance matrix** S is defined as

$$S = \frac{1}{n-1}\sum_{i=1}^{n}(\boldsymbol{X}_i - \overline{\boldsymbol{X}})(\boldsymbol{X}_i - \overline{\boldsymbol{X}})'$$

Its elements $s_{h,l}$ are the **adjusted sample covariances**, with main diagonal elements $s_h^2 = s_{h,h}$ being the adjusted sample variances:

$$s_{h,l} = \frac{1}{n-1}\sum_{i=1}^{n}(X_{ih} - \overline{X_h})(X_{il} - \overline{X_l}).$$

If the sample is i.i.d., both $\widehat{\Sigma}$ and S are consistent estimators for the population covariance matrix

$$\Sigma = Var(\boldsymbol{X}) = E[(\boldsymbol{X} - E[\boldsymbol{X}])(\boldsymbol{X} - E[\boldsymbol{X}])'].$$

The adjusted covariance matrix S is unbiased, while $\widehat{\Sigma}$ is biased but has lower sampling variance.

```
## Adjusted sample covariance matrix
cov(dat)
```

```
           wage         edu         fem
wage 428.948332 21.82614057 -1.66314777
edu   21.826141  7.53198925  0.06037303
fem   -1.663148  0.06037303  0.24448764
```

Correlation Matrix

The **sample correlation coefficient** between the variables h and l is the standardized sample covariance:

$$r_{h,l} = \frac{s_{h,l}}{s_h s_l} = \frac{\sum_{i=1}^{n}(X_{ih} - \overline{X_h})(X_{il} - \overline{X_l})}{\sqrt{\sum_{i=1}^{n}(X_{ih} - \overline{X_h})^2}\sqrt{\sum_{i=1}^{n}(X_{il} - \overline{X_l})^2}} = \frac{\hat{\sigma}_{h,l}}{\hat{\sigma}_h \hat{\sigma}_l}.$$

These coefficients form the **sample correlation matrix** R, expressed as:

$$R = D^{-1} S D^{-1},$$

where D is the diagonal matrix of adjusted sample standard deviations:

$$D = diag(s_1, \ldots, s_k) = \begin{pmatrix} s_1 & 0 & \ldots & 0 \\ 0 & s_2 & \ldots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \ldots & s_k \end{pmatrix}$$

The matrices $\widehat{\Sigma}$, S, and R are symmetric.

```
cor(dat)
```

```
           wage        edu         fem
wage  1.0000000 0.38398973 -0.16240519
edu   0.3839897 1.00000000  0.04448972
fem  -0.1624052 0.04448972  1.00000000
```

We find a strong positive correlation between `wage` and `edu`, a substantial negative correlation between `wage` and `fem`, and a negligible correlation between `edu` and `fem`.

2.9 R-codes

metrics-sec02.R

Part II

Linear Regression

3 Least Squares

This section introduces the least squares method, focusing exclusively on its geometric and computational aspects as an optimization problem that minimizes the sum of squared deviations between observed and fitted values. The statistical properties of least squares, including the formal linear model framework, hypothesis testing, and estimator properties, will be covered in the next sections.

3.1 Regression Fundamentals

Regression Problem

The idea of regression analysis is to approximate a univariate dependent variable Y_i (also known as the regressand or response variable) as a function of the k-variate vector of the independent variables $\boldsymbol{X}_i$ (also known as regressors or predictor variables). The relationship is formulated as

$$Y_i \approx f(\boldsymbol{X}_i), \quad i = 1, \dots, n,$$

where $Y_1, \dots, Y_n$ is a univariate dataset for the dependent variable and $\boldsymbol{X}_1, \dots, \boldsymbol{X}_n$ a k-variate dataset for the regressor variables.

The goal of the least squares method is to find the regression function that minimizes the squared difference between actual and fitted values of Y_i:

$$\min_{f(\cdot)} \sum_{i=1}^{n} (Y_i - f(\boldsymbol{X}_i))^2.$$

Linear Regression

If the regression function $f(\boldsymbol{X}_i)$ is linear in $\boldsymbol{X}_i$, i.e.,

$$f(\boldsymbol{X}_i) = b_1 + b_2 X_{i2} + \dots + b_k X_{ik} = \boldsymbol{X}_i' \boldsymbol{b}, \quad \boldsymbol{b} \in \mathbb{R}^k,$$

the minimization problem is known as the **ordinary least squares (OLS)** problem. The coefficient vector has k entries:

$$\boldsymbol{b} = (b_1, b_2, \dots, b_k)'.$$

To avoid the unrealistic constraint of the regression line passing through the origin, a constant term (intercept) is always included in $\boldsymbol{X}_i$, typically as the first regressor:

$$\boldsymbol{X}_i = (1, X_{i2}, \dots, X_{ik})'.$$

Despite its linear framework, linear regressions can be quite adaptable to nonlinear relationships by incorporating nonlinear transformations of the original regressors. Examples include polynomial terms (e.g., squared, cubic), interaction terms (combining different variables), and logarithmic transformations.

3.2 Ordinary least squares (OLS)

The **sum of squared errors** for a given coefficient vector $\boldsymbol{b} \in \mathbb{R}^k$ is defined as

$$S_n(\boldsymbol{b}) = \sum_{i=1}^{n} (Y_i - f(\boldsymbol{X}_i))^2 = \sum_{i=1}^{n} (Y_i - \boldsymbol{X}_i'\boldsymbol{b})^2.$$

It is minimized by the **least squares coefficient vector**

$$\hat{\boldsymbol{\beta}} = \operatorname{argmin}_{\boldsymbol{b} \in \mathbb{R}^k} \sum_{i=1}^{n} (Y_i - \boldsymbol{X}_i'\boldsymbol{b})^2.$$

Least squares coefficients

If the $k \times k$ matrix $(\sum_{i=1}^{n} \boldsymbol{X}_i \boldsymbol{X}_i')$ is invertible, the solution for the ordinary least squares problem is uniquely determined by

$$\hat{\boldsymbol{\beta}} = \Big(\sum_{i=1}^{n} \boldsymbol{X}_i \boldsymbol{X}_i' \Big)^{-1} \sum_{i=1}^{n} \boldsymbol{X}_i Y_i.$$

The **fitted values** or predicted values are

$$\widehat{Y}_i = \hat{\beta}_1 + \hat{\beta}_2 X_{i2} + \dots + \hat{\beta}_k X_{ik} = \boldsymbol{X}_i'\hat{\boldsymbol{\beta}}, \quad i = 1, \dots, n.$$

The **residuals** are the difference between observed and fitted values:

$$\hat{u}_i = Y_i - \widehat{Y}_i = Y_i - \boldsymbol{X}_i'\hat{\boldsymbol{\beta}}, \quad i = 1, \dots, n.$$

3.3 Regression Plots

Line Fitting

Let's examine the linear relationship between average test scores and the student-teacher ratio:

```
data(CASchools, package = "AER")
CASchools$STR = CASchools$students/CASchools$teachers
CASchools$score = (CASchools$read+CASchools$math)/2
fit1 = lm(score ~ STR, data = CASchools)
fit1$coefficients
```

```
(Intercept)         STR
 698.932949   -2.279808
```

We have

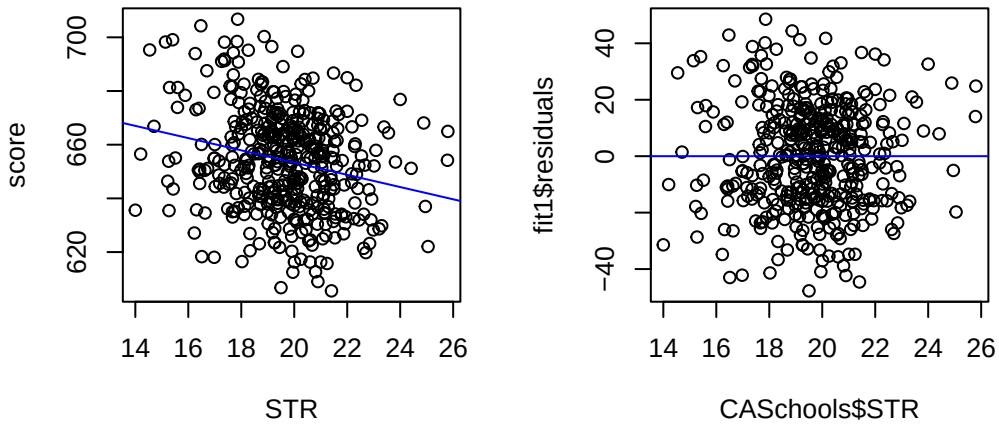
$$\hat{\boldsymbol{\beta}} = \begin{pmatrix} 698.9 \\ -2.28 \end{pmatrix}.$$

The fitted regression line is

$$698.9 - 2.28 \text{ STR}.$$

We can plot the regression line over a scatter plot of the data:

```
par(mfrow = c(1,2), cex=0.8)
plot(score ~ STR, data = CASchools)
abline(fit1, col="blue")
plot(CASchools$STR, fit1$residuals)
abline(0, 0, col="blue")
```



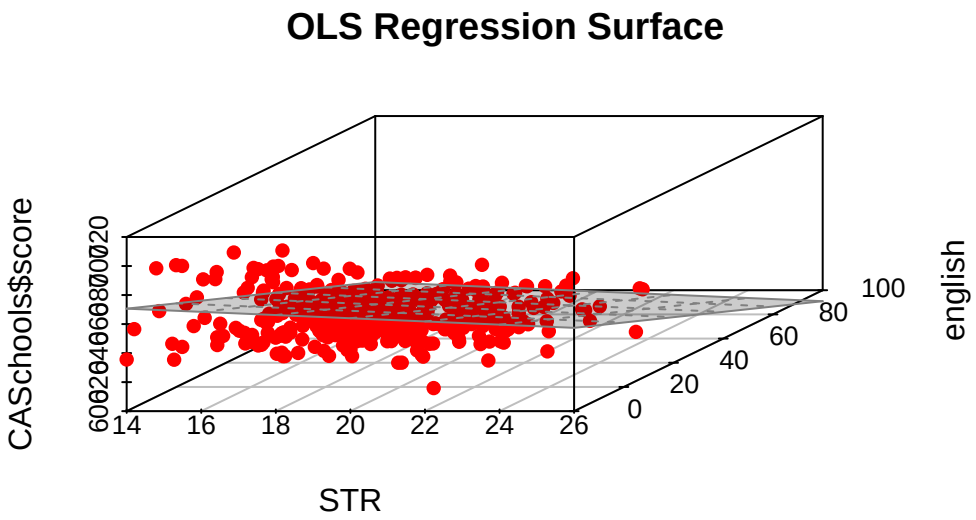
Multidimensional Visualizations

Let's include the percentage of english learners as an additional regressor:

```
fit2= lm(score ~ STR + english, data = CASchools)
fit2$coefficients
```

```
(Intercept)         STR     english
686.0322445  -1.1012956  -0.6497768
```

A 3D plot provides a visual representation of the resulting regression line (surface):



Adding the additional predictor `income` gives a regression specification with dimensions beyond visual representation:

```
fit3 = lm(score ~ STR + english + income, data = CASchools)
fit3$coefficients
```

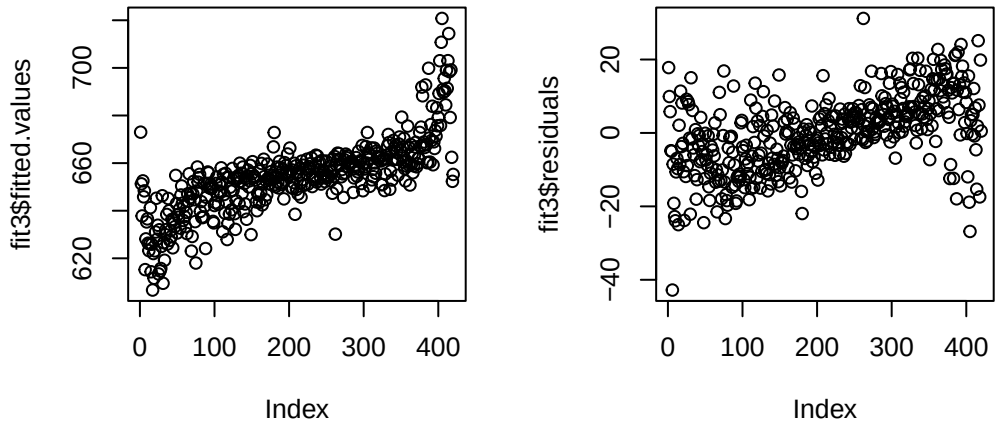
```
 (Intercept)          STR      english       income
640.31549821  -0.06877542  -0.48826683   1.49451661
```

The fitted regression line now includes three predictors and four coefficients:

$$640.3 - 0.07 \text{ STR} - 0.49 \text{ english} + 1.49 \text{ income}$$

For specifications with multiple regressors, fitted values and residuals can still be visualized:

```
par(mfrow = c(1,2), cex=0.8)
plot(fit3$fitted.values)
plot(fit3$residuals)
```



The pattern of fitted values arises because the observations in the `CASchools` dataset are sorted in ascending order by test score.

3.4 Matrix notation

OLS Formula

Matrix notation is convenient because it eliminates the need for summation symbols and indices. We define the response vector $\boldsymbol{Y}$ and the regressor matrix (design matrix) $\boldsymbol{X}$ as follows:

$$\boldsymbol{Y} = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{pmatrix}, \quad \boldsymbol{X} = \begin{pmatrix} \boldsymbol{X}_1' \\ \boldsymbol{X}_2' \\ \vdots \\ \boldsymbol{X}_n' \end{pmatrix} = \begin{pmatrix} 1 & X_{12} & \dots & X_{1k} \\ \vdots & & & \vdots \\ 1 & X_{n2} & \dots & X_{nk} \end{pmatrix}$$

Note that $\sum_{i=1}^n \boldsymbol{X}_i \boldsymbol{X}_i' = \boldsymbol{X}'\boldsymbol{X}$ and $\sum_{i=1}^n \boldsymbol{X}_i Y_i = \boldsymbol{X}'\boldsymbol{Y}$.

The least squares coefficient vector becomes

$$\hat{\boldsymbol{\beta}} = \Big(\sum_{i=1}^n \boldsymbol{X}_i \boldsymbol{X}_i' \Big)^{-1} \sum_{i=1}^n \boldsymbol{X}_i Y_i = (\boldsymbol{X}'\boldsymbol{X})^{-1} \boldsymbol{X}'\boldsymbol{Y}.$$

The vector of fitted values can be computed as follows:

$$\widehat{\boldsymbol{Y}} = \begin{pmatrix} \widehat{Y}_1 \\ \vdots \\ \widehat{Y}_n \end{pmatrix} = \boldsymbol{X}\hat{\boldsymbol{\beta}} = \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}.$$

Residuals

The vector of residuals is given by

$$\hat{\boldsymbol{u}} = \begin{pmatrix} \hat{u}_1 \\ \vdots \\ \hat{u}_n \end{pmatrix} = \boldsymbol{Y} - \widehat{\boldsymbol{Y}} = \boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}.$$

An important property of the residual vector is: $\boldsymbol{X}'\hat{\boldsymbol{u}} = \boldsymbol{0}$. To see that this property holds, let's rearrange the OLS formula:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y} \quad \Leftrightarrow \quad \boldsymbol{X}'\boldsymbol{X}\hat{\boldsymbol{\beta}} = \boldsymbol{X}'\boldsymbol{Y}.$$

The dependent dependent variable vector can be decomposed into the vector of fitted values and the residual vector:

$$\boldsymbol{Y} = \boldsymbol{X}\hat{\boldsymbol{\beta}} + \hat{\boldsymbol{u}}.$$

Substituting this into the OLS formula from above gives:

$$\boldsymbol{X}'\boldsymbol{X}\hat{\boldsymbol{\beta}} = \boldsymbol{X}'(\boldsymbol{X}\hat{\boldsymbol{\beta}} + \hat{\boldsymbol{u}}) \quad \Leftrightarrow \quad \boldsymbol{0} = \boldsymbol{X}'\hat{\boldsymbol{u}}.$$

This property has a geometric interpretation: it means the residuals are orthogonal to all regressors. This makes sense because if there were any linear relationship left between the residuals and the regressors, we could have captured it in our model to improve the fit.

3.5 Goodness of Fit

Analysis of Variance

The orthogonality property of the residual vector can be written in a more detailed way as follows:

$$\boldsymbol{X}'\hat{\boldsymbol{u}} = \begin{pmatrix} \sum_{i=1}^n \hat{u}_i \\ \sum_{i=1}^n X_{i2}\hat{u}_i \\ \vdots \\ \sum_{i=1}^n X_{ik}\hat{u}_i \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}. \tag{3.1}$$

In particular, the sample mean of the residuals is zero:

$$\frac{1}{n}\sum_{i=1}^n \hat{u}_i = 0.$$

Therefore, the sample variance of the residuals is simply the sample mean of squared residuals:

$$\hat{\sigma}^2_{\hat{u}} = \frac{1}{n}\sum_{i=1}^n \hat{u}_i^2.$$

The sample variance of the dependent variable is

$$\hat{\sigma}_Y^2 = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y})^2,$$

and the sample variance of the fitted values is

$$\hat{\sigma}_{\widehat{Y}}^2 = \frac{1}{n}\sum_{i=1}^{n}(\widehat{Y}_i - \overline{\widehat{Y}})^2.$$

The three sample variances are connected through the analysis of variance formula:

$$\hat{\sigma}_Y^2 = \hat{\sigma}_{\widehat{Y}}^2 + \hat{\sigma}_{\widehat{u}}^2.$$

Hence, the larger the proportion of the explained sample variance, the better the fit of the OLS regression.

R-squared

The analysis of variance formula motivates the definition of the **R-squared coefficient**:

$$R^2 = 1 - \frac{\hat{\sigma}_{\widehat{u}}^2}{\hat{\sigma}_Y^2} = 1 - \frac{\sum_{i=1}^{n}\hat{u}_i^2}{\sum_{i=1}^{n}(Y_i - \overline{Y})^2} = \frac{\sum_{i=1}^{n}(\widehat{Y}_i - \overline{\widehat{Y}})^2}{\sum_{i=1}^{n}(Y_i - \overline{Y})^2}.$$

The R-squared describes the proportion of sample variation in $\boldsymbol{Y}$ explained by $\widehat{\boldsymbol{Y}}$. We have $0 \leq R^2 \leq 1$.

In a regression of Y_i on a single regressor Z_i with intercept (simple linear regression), the R-squared is equal to the squared sample correlation coefficient of Y_i and Z_i.

An R-squared of 0 indicates no sample variation in $\widehat{\boldsymbol{Y}}$ (a flat regression line/surface), whereas a value of 1 indicates no variation in $\hat{\boldsymbol{u}}$, indicating a perfect fit. The higher the R-squared, the better the OLS regression fits the data.

However, a low R-squared does not necessarily mean the regression specification is bad. It just implies that there is a high share of unobserved heterogeneity in $\boldsymbol{Y}$ that is not captured by the regressors $\boldsymbol{X}$ linearly.

Conversely, a high R-squared does not necessarily mean a good regression specification. It just means that the regression fits the sample well. Too many unnecessary regressors lead to overfitting.

If $k = n$, we have $R^2 = 1$ even if none of the regressors has an actual influence on the dependent variable.

Adjusted R-squared

Recall that the deviations $(Y_i - \overline{Y})$ cannot vary freely because they are subject to the constraint $\sum_{i=1}^{n}(Y_i - \overline{Y})$, which is why we lose 1 degree of freedom in the sample variance of $\boldsymbol{Y}$.

For the sample variance of $\widehat{\boldsymbol{u}}$, we loose k degrees of freedom because the residuals are subject to the constraints from Equation 3.1. The adjusted sample variance of the residuals is therefore defined as:

$$s_{\widehat{u}}^2 = \frac{1}{n-k}\sum_{i=1}^{n}\widehat{u}_i^2.$$

By incorporating adjusted versions in the R-squared definition, we penalize regression specifications with large k. The **adjusted R-squared** is

$$\overline{R}^2 = 1 - \frac{\frac{1}{n-k}\sum_{i=1}^{n}\widehat{u}_i^2}{\frac{1}{n-1}\sum_{i=1}^{n}(Y_i - \overline{Y})^2} = 1 - \frac{s_{\widehat{u}}^2}{s_Y^2}.$$

The R-squared should be used for interpreting the share of variation explained by the fitted regression line. The adjusted R-squared should be used for comparing different OLS regression specifications.

3.6 Regression Table

The `modelsummary()` function can be used to produce comparison tables of regression outputs:

```
library(modelsummary)
mymodels = list(fit1, fit2, fit3)
modelsummary(mymodels,
             statistic = NULL,
             gof_map = c("nobs", "r.squared", "adj.r.squared", "rmse"))
```

Model (3) explains the most variation in test scores and provides the best fit to the data, as indicated by the highest R^2 and the lowest residual standard error.

In model (1), schools with one more student per class are predicted to have a 2.28-point lower test score. This effect decreases to 1.1 points in model (2), after accounting for the percentage of English learners, and drops further to just 0.07 points in model (3), once income is also included.

	(1)	(2)	(3)
(Intercept)	698.933	686.032	640.315
STR	−2.280	−1.101	−0.069
english		−0.650	−0.488
income			1.495
Num.Obs.	420	420	420
R2	0.051	0.426	0.707
R2 Adj.	0.049	0.424	0.705
RMSE	18.54	14.41	10.30

The **Root Mean Squared Error (RMSE)** is the squareroot of the mean squared error of the residuals:

$$RMSE(\hat{\boldsymbol{\beta}}) = \hat{\sigma}_{\hat{u}} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \hat{u}_i^2}.$$

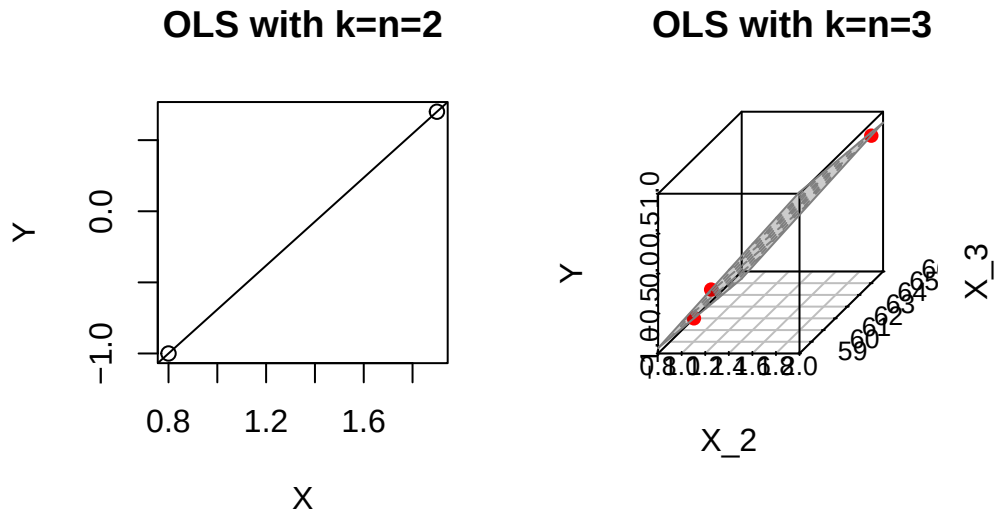
While the R-squared increases in the number of regressors, the RMSE decreases.

To give deeper meaning to these results and understand their interpretation within a broader context, we turn to a formal probabilistic model framework in the next section.

3.7 When OLS Fails

Too many regressors

OLS should be considered for regression problems with $k << n$ (small k and large n). When the number of predictors k approaches or equals the number of observations n, we run into the problem of overfitting. Specifically, at $k = n$, the regression line will perfectly fit the data.



If $k = n \geq 4$, we can no longer visualize the OLS regression line in the 3D space, but the problem of a perfect fit is still present. If $k > n$, there exists no unique OLS solution because $\boldsymbol{X}'\boldsymbol{X}$ is not invertible. Regression problems with $k \approx n$ or $k > n$ are called **high-dimensional regressions**.

Perfect multicollinearity

The only requirement for computing the OLS coefficients is the invertibility of the matrix $\boldsymbol{X}'\boldsymbol{X}$. As discussed above, a necessary condition is that $k \leq n$.

Another reason the matrix may not be invertible is if two or more regressors are perfectly collinear. Two variables are perfectly collinear if their sample correlation is 1 or -1. Multicollinearity arises if one variable is a linear combination of the other variables.

Common causes are duplicating a regressor or using the same variable in different units (e.g., GDP in both EUR and USD).

Perfect multicollinearity (or strict multicollinearity) arises if the regressor matrix does not have full column rank: $\text{rank}(\boldsymbol{X}) < k$. It implies $\text{rank}(\boldsymbol{X}'\boldsymbol{X}) < k$, so that the matrix is singular and $\hat{\boldsymbol{\beta}}$ cannot be computed.

Near multicollinearity occurs when two columns of $\boldsymbol{X}$ have a sample correlation very close to 1 or -1. Then, $(\boldsymbol{X}'\boldsymbol{X})$ is "near singular", its eigenvalues are very small, and $(\boldsymbol{X}'\boldsymbol{X})^{-1}$ becomes very large, causing numerical problems.

If $k \leq n$ and multicollinearity is present, it means that at least one regressor is redundant and can be dropped.

Dummy variable trap

A common cause of strict multicollinearity is the inclusion of too many dummy variables. Let's consider the `cps` data and add a dummy variable for non-married individuals:

```
cps = read.csv("cps.csv")
cps$nonmarried = 1-cps$married
fit4 = lm(wage ~ married + nonmarried, data = cps)
fit4$coefficients
```

```
(Intercept)     married  nonmarried
  19.338695    6.997155          NA
```

The coefficient for `nonmarried` is `NA`. We fell into the dummy variable trap!

The dummy variables `married` and `nonmarried` are collinear with the intercept variable because $married + nonmarried = 1$, which leads to a singular matrix $\boldsymbol{X}'\boldsymbol{X}$ and therefore to perfect multicollinearity.

The solution is to use one dummy variable less than factor levels, as R automatically does by omitting the last dummy variable. Another solution would be to remove the intercept from the model, which can be done by adding `-1` to the model formula:

```
fit5 = lm(wage ~ married + nonmarried - 1, data = cps)
fit5$coefficients
```

```
   married nonmarried
  26.33585   19.33869
```

3.8 R-codes

metrics-sec03.R

4 Linear Model

4.1 Conditional Expectation

In econometrics, we often analyze how a variable of interest (like wages) varies systematically with other variables (like education or experience). The **conditional expectation function (CEF)** provides a powerful framework for describing these relationships.

The conditional expectation of Y given X is the expected value of Y for each possible value of X. For a continuous random variable Y we have

$$E[Y|X = x] = \int_{-\infty}^{\infty} y \, f_{Y|X}(y|x) \, dy$$

where $f_{Y|X}(y|x)$ is the conditional density of Y given $X = x$.

The CEF maps values of X to corresponding conditional means of Y. As a function of the random variable X, the CEF itself is a random variable:

$$E[Y|X] = m(X), \quad \text{where } m(x) = E[Y|X = x]$$

For a comprehensive treatment of conditional expectations see Probability Tutorial Part 2

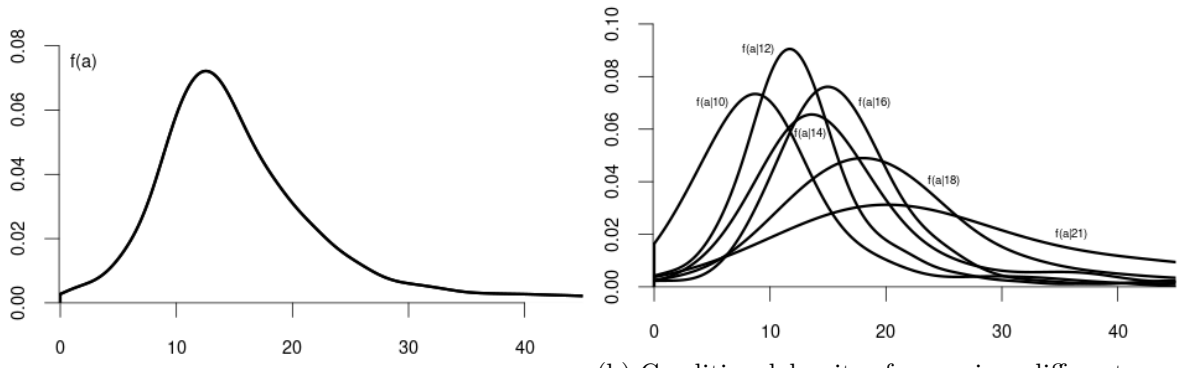
Examples

Let's examine this concept using wage and education as examples. When X is discrete (such as years of education), we can analyze how wage distributions change across education levels by comparing their **conditional distributions**:

Notice how the conditional distributions shift rightward as education increases, indicating higher average wages with higher education.

From these conditional densities, we can compute the expected wage for each education level. Plotting these conditional expectations gives the CEF:

$$m(x) = E[\text{wage} \mid \text{edu} = x]$$

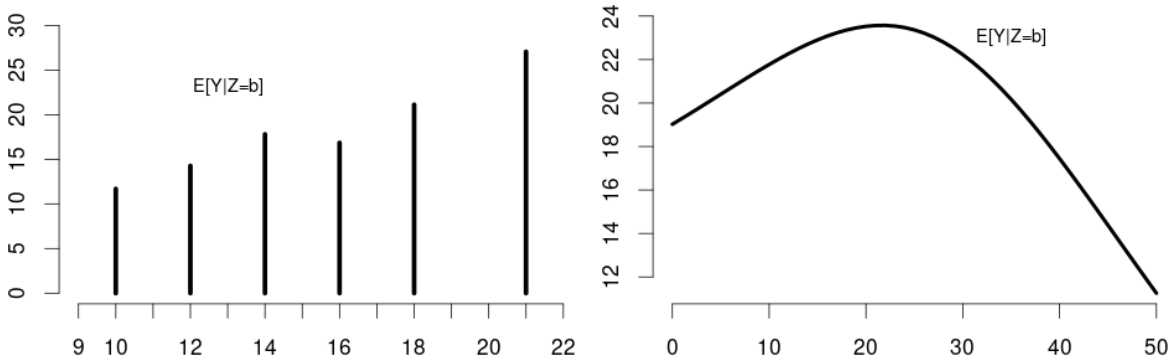


(a) Unconditional density of wage

(b) Conditional density of wage given different years of education

Figure 4.1: Unconditional density $f(y)$ and conditional densities $f_{Y|X}(y|x)$ of wage given x years of education

Since education is discrete, the CEF is defined only at specific values, as shown in the left plot below:



(a) CEF of wage given education

(b) CEF of wage given experience

Figure 4.2: Conditional expectations of wage given education (left) and experience (right)

When X is continuous (like years of experience), the CEF becomes a smooth function (right plot). The shape of E[wage|experience] reflects real-world patterns: wages rise quickly early in careers, then plateau, and may eventually decline near retirement.

The CEF as a Random Variable

It's important to distinguish between:

- $E[Y|X = x]$: a **number** (the conditional mean at a specific value)

- $E[Y|X]$: a **function of** X, which is itself a **random variable**

For instance, if $X =$ education has the probability mass function:

$$P(X = x) = \begin{cases} 0.06 & \text{if } x = 10 \\ 0.43 & \text{if } x = 12 \\ 0.16 & \text{if } x = 14 \\ 0.08 & \text{if } x = 16 \\ 0.24 & \text{if } x = 18 \\ 0.03 & \text{if } x = 21 \\ 0 & \text{otherwise} \end{cases}$$

Then $E[Y|X]$ as a random variable has the probability mass function:

$$P(E[Y|X] = y) = \begin{cases} 0.06 & \text{if } y = 11.68 \text{ (when } X = 10) \\ 0.43 & \text{if } y = 14.26 \text{ (when } X = 12) \\ 0.16 & \text{if } y = 17.80 \text{ (when } X = 14) \\ 0.08 & \text{if } y = 16.84 \text{ (when } X = 16) \\ 0.24 & \text{if } y = 21.12 \text{ (when } X = 18) \\ 0.03 & \text{if } y = 27.05 \text{ (when } X = 21) \\ 0 & \text{otherwise} \end{cases}$$

The CEF assigns to each value of X the expected value of Y given that information.

4.2 CEF Properties

The conditional expectation function has several important properties that make it a fundamental tool in econometric analysis.

Law of Iterated Expectations (LIE)

The **law of iterated expectations** connects conditional and unconditional expectations:

$$E[Y] = E[E[Y|X]]$$

This means that to compute the overall average of Y, we can first compute the average of Y within each group defined by X, then average those conditional means using the distribution of X.

This is analogous to the law of total probability, where we compute marginal probabilities or densities as weighted averages of conditional ones:

When X is discrete:

$$P(Y = y) = \sum_x P(Y = y \mid X = x) \cdot P(X = x)$$

When X is continuous:

$$f_Y(y) = \int_{-\infty}^{\infty} f_{Y|X}(y \mid x) \cdot f_X(x)\, dx$$

Similarly, the LIE states:

When X is discrete:

$$E[Y] = \sum_x E[Y \mid X = x] \cdot P(X = x)$$

When X is continuous:

$$E[Y] = \int_{-\infty}^{\infty} E[Y \mid X = x] \cdot f_X(x)\, dx$$

Let's apply this to our wage and education example. With $X =$ education and $Y =$ wage, we have:

$$\begin{aligned}
E[Y|X = 10] &= 11.68, \qquad P(X = 10) = 0.06 \\
E[Y|X = 12] &= 14.26, \qquad P(X = 12) = 0.43 \\
E[Y|X = 14] &= 17.80, \qquad P(X = 14) = 0.16 \\
E[Y|X = 16] &= 16.84, \qquad P(X = 16) = 0.08 \\
E[Y|X = 18] &= 21.12, \qquad P(X = 18) = 0.24 \\
E[Y|X = 21] &= 27.05, \qquad P(X = 21) = 0.03
\end{aligned}$$

The law of iterated expectations gives us:

$$\begin{aligned}
E[Y] &= \sum_x E[Y|X = x] \cdot P(X = x) \\
&= 11.68 \cdot 0.06 + 14.26 \cdot 0.43 + 17.80 \cdot 0.16 \\
&\quad + 16.84 \cdot 0.08 + 21.12 \cdot 0.24 + 27.05 \cdot 0.03 \\
&= 0.7008 + 6.1318 + 2.848 + 1.3472 + 5.0688 + 0.8115 \\
&= 16.91
\end{aligned}$$

This unconditional expected wage of 16.91 aligns with what we would calculate from the unconditional density. The LIE provides us with a powerful way to bridge conditional expectations (within education groups) and the overall unconditional expectation (averaging across all education levels).

Conditioning Theorem (CT)

The **conditioning theorem** (also called the factorization rule) states:

$$E[g(X)Y \mid X] = g(X) \cdot E[Y \mid X]$$

This means that when taking the conditional expectation of a product where one factor is a function of the conditioning variable, that factor can be treated as a constant and factored out. Once we condition on X, the value of $g(X)$ is fixed.

If $Y = \text{wage}$ and $X = \text{education}$, then for someone with 16 years of education:

$$E[16 \cdot \text{wage} \mid \text{edu} = 16] = 16 \cdot E[\text{wage} \mid \text{edu} = 16]$$

More generally, if we want to find the expected product of education and wage, conditional on education:

$$E[\text{edu} \cdot \text{wage} \mid \text{edu}] = \text{edu} \cdot E[\text{wage} \mid \text{edu}]$$

Best Predictor Property

The conditional expectation $E[Y|X]$ is the **best predictor** of Y given X in terms of mean squared error:

$$E[Y|X] = \arg\min_{g(\cdot)} E[(Y - g(X))^2]$$

This means that among all possible functions of X, the CEF minimizes the expected squared prediction error. In practical terms, if you want to predict wages based only on education, the optimal prediction is exactly the conditional mean wage for each education level.

For example, if someone has 18 years of education, our best prediction of their wage (minimizing expected squared error) is $E[\text{wage}|\text{education} = 18] = 21.12$.

No other function of education, whether linear, quadratic, or more complex, can yield a better prediction in terms of expected squared error than the CEF itself.

Independence Implications

If Y and X are independent, then:

$$E[Y|X] = E[Y]$$

When variables are independent, knowing X provides no information about Y, so the conditional expectation equals the unconditional expectation. The CEF becomes a constant function that doesn't vary with X.

In our wage example, if education and wage were completely independent, the CEF would be a horizontal line at the overall average wage of 16.91. Each conditional density $f_{Y|X}(y|x)$ would be identical to the unconditional density $f(y)$, and the conditional means would all equal the unconditional mean.

The fact that our CEF for wage given education has a positive slope indicates that these variables are not independent—higher education is associated with higher expected wages.

4.3 Linear Model Specification

Prediction Error

Consider a sample $\{(Y_i, \boldsymbol{X}_i)\}_{i=1}^n$. We have established that the **conditional expectation function (CEF)** $E[Y_i|\boldsymbol{X}_i]$ is the best predictor of Y_i given $\boldsymbol{X}_i$, minimizing the mean squared prediction error.

This leads to the following prediction error:

$$u_i = Y_i - E[Y_i|\boldsymbol{X}_i]$$

By construction, this error has a conditional mean of zero:

$$E[u_i|\boldsymbol{X}_i] = 0$$

This zero conditional mean property follows directly from the law of iterated expectations:

$$\begin{aligned} E[u_i|\boldsymbol{X}_i] &= E[Y_i - E[Y_i|\boldsymbol{X}_i] \mid \boldsymbol{X}_i] \\ &= E[Y_i \mid \boldsymbol{X}_i] - E[E[Y_i|\boldsymbol{X}_i] \mid \boldsymbol{X}_i] \\ &= E[Y_i \mid \boldsymbol{X}_i] - E[Y_i \mid \boldsymbol{X}_i] = 0 \end{aligned}$$

We can thus always decompose the outcome as:

$$Y_i = E[Y_i|\boldsymbol{X}_i] + u_i$$

where $E[u_i|\boldsymbol{X}_i] = 0$. This equation is not yet a regression model. It's simply the decomposition of Y_i into its conditional expectation and an unpredictable component.

Linear Regression Model

To move to a regression framework, we impose a structural assumption about the form of the CEF. The key assumption of the **linear regression model** is that the conditional expectation is a **linear function** of the regressors:

$$E[Y_i \mid \boldsymbol{X}_i] = \boldsymbol{X}_i'\boldsymbol{\beta}$$

Substituting this into our decomposition yields the **linear regression equation**:

$$Y_i = \boldsymbol{X}_i'\boldsymbol{\beta} + u_i \tag{4.1}$$

with the crucial assumption:

$$E[u_i \mid \boldsymbol{X}_i] = 0 \tag{4.2}$$

Exogeneity

This assumption (Equation 9.3) is called **exogeneity** or **mean independence**. It ensures that the linear function $\boldsymbol{X}_i'\boldsymbol{\beta}$ correctly captures the conditional mean of Y_i.

Under the linear regression equation (Equation 4.1) we have the following equivalence:

$$E[Y_i \mid \boldsymbol{X}_i] = \boldsymbol{X}_i'\boldsymbol{\beta} \quad \Leftrightarrow \quad E[u_i \mid \boldsymbol{X}_i] = 0$$

Therefore, the linear regression model in its most general form is characterized by the two conditions: linear regression equation (Equation 4.1) and exogenous regressors (Equation 9.3).

For example, in a wage regression, exogeneity means that the expected wage conditional on education and experience is exactly captured by the linear combination of these variables. No systematic pattern remains in the error term.

Model Misspecification

If the true conditional expectation function is nonlinear (e.g., if wages increase with education at a diminishing rate), then $E[Y_i \mid \boldsymbol{X}_i] \neq \boldsymbol{X}_i'\boldsymbol{\beta}$, and the model is **misspecified**. In such cases, the linear model provides the best linear approximation to the true CEF, but systematic patterns remain in the error term.

It's important to note that u_i may still be statistically dependent on $\boldsymbol{X}_i$ in ways other than its mean. For example, the **variance** of u_i may depend on $\boldsymbol{X}_i$ in the case of **heteroskedasticity**. For instance, wage dispersion might increase with education level. The assumption $E[u_i \mid \boldsymbol{X}_i] = 0$ requires only that the conditional **mean** of the error is zero, not that the error is completely independent of the regressors.

4.4 Population Regression Coefficient

Under the linear model

$$Y_i = \boldsymbol{X}_i'\boldsymbol{\beta} + u_i, \quad E[u_i \mid \boldsymbol{X}_i] = 0,$$

we are interested in the **population regression coefficient** $\boldsymbol{\beta}$, which indicates how the conditional mean of Y_i varies **linearly** with the regressors in $\boldsymbol{X}_i$.

Moment Condition

A key implication of the exogeneity condition $E[u_i \mid \boldsymbol{X}_i] = 0$ is that the regressors are **mean uncorrelated** with the error term:

$$E[\boldsymbol{X}_i u_i] = \mathbf{0}$$

This can be derived from the exogeneity condition using the law of iterated expectations:

$$E[\boldsymbol{X}_i u_i] = E[E[\boldsymbol{X}_i u_i \mid \boldsymbol{X}_i]] = E[\boldsymbol{X}_i \cdot E[u_i \mid \boldsymbol{X}_i]] = E[\boldsymbol{X}_i \cdot 0] = \mathbf{0}$$

Substituting the linear model into the mean uncorrelatedness condition gives a moment condition that identifies $\boldsymbol{\beta}$:

$$\mathbf{0} = E[\boldsymbol{X}_i u_i] = E[\boldsymbol{X}_i(Y_i - \boldsymbol{X}_i'\boldsymbol{\beta})] = E[\boldsymbol{X}_i Y_i] - E[\boldsymbol{X}_i \boldsymbol{X}_i']\boldsymbol{\beta}$$

Rearranging to solve for $\boldsymbol{\beta}$:

$$E[\boldsymbol{X}_i Y_i] = E[\boldsymbol{X}_i \boldsymbol{X}_i']\boldsymbol{\beta}$$

Assuming that the matrix $E[\boldsymbol{X}_i \boldsymbol{X}_i']$ is invertible, we can express the population regression coefficient as:

$$\boldsymbol{\beta} = \left(E[\boldsymbol{X}_i \boldsymbol{X}_i']\right)^{-1} E[\boldsymbol{X}_i Y_i]$$

This expression shows that $\boldsymbol{\beta}$ is entirely determined by the joint distribution of $(Y_i, \boldsymbol{X}_i')$ in the population.

The invertibility of $E[\boldsymbol{X}_i \boldsymbol{X}_i']$ is guaranteed if there is no perfect linear relationship among the regressors. In particular, no pair of regressors should be perfectly correlated, and no regressor should be a perfect linear combination of the other regressors.

OLS Estimation

To estimate $\boldsymbol{\beta}$ from data, we replace population moments with sample moments. Given a sample $\{(Y_i, \boldsymbol{X}_i)\}_{i=1}^n$, the **ordinary least squares (OLS)** estimator is:

$$\hat{\boldsymbol{\beta}} = \left(\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}_i \boldsymbol{X}_i'\right)^{-1} \left(\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}_i Y_i\right)$$

This can be simplified to the familiar form:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y}$$

The OLS estimator solves the **sample moment condition**:

$$\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}_i (Y_i - \boldsymbol{X}_i'\hat{\boldsymbol{\beta}}) = \mathbf{0}$$

or equivalently:

$$\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}_i \hat{u}_i = \mathbf{0}$$

where $\hat{u}_i = Y_i - \boldsymbol{X}_i'\hat{\boldsymbol{\beta}}$ are the sample residuals.

In this framework, OLS can be viewed as a **method of moments estimator**, solving the sample analogue of the population moment condition $E[\boldsymbol{X}_i u_i] = \mathbf{0}$. The method of moments principle replaces theoretical moments with their empirical counterparts to obtain estimates of unknown parameters.

4.5 Marginal Effects

Consider the regression model of hourly wage on education (years of schooling),

$$\text{wage}_i = \beta_1 + \beta_2 \text{edu}_i + u_i, \quad i = 1, \dots, n,$$

where the exogeneity assumption holds:

$$E[u_i | \text{edu}_i] = 0.$$

The population regression function, which gives the conditional expectation of wage given education, can be derived as:

$$\begin{aligned} m(\text{edu}_i) &= E[\text{wage}_i|\text{edu}_i] \\ &= \beta_1 + \beta_2 \cdot \text{edu}_i + E[u_i|\text{edu}_i] \\ &= \beta_1 + \beta_2 \cdot \text{edu}_i \end{aligned}$$

Thus, the average wage level of all individuals with z years of schooling is:

$$m(z) = \beta_1 + \beta_2 \cdot z.$$

Interpretation of Coefficients

In the linear regression model

$$Y_i = \boldsymbol{X}_i'\boldsymbol{\beta} + u_i,$$

the coefficient vector $\boldsymbol{\beta}$ captures the way the **conditional mean of** Y_i changes with the regressors $\boldsymbol{X}_i$. Under the exogeneity assumption,

$$E[Y_i \mid \boldsymbol{X}_i] = \boldsymbol{X}_i'\boldsymbol{\beta} = \beta_1 + \beta_2 X_{i2} + ... + \beta_k X_{ik}.$$

This linearity allows for a simple interpretation. The coefficient β_j represents the **partial derivative** of the conditional mean with respect to X_{ij}:

$$\frac{\partial E[Y_i \mid \boldsymbol{X}_i]}{\partial X_{ij}} = \beta_j.$$

This means that β_j measures the **marginal effect** of a one-unit increase in X_{ij} on the expected value of Y_i, holding all other variables constant.

If X_{ij} is a dummy variable (i.e., binary), then β_j measures the discrete change in $E[Y_i \mid \boldsymbol{X}_i]$ when X_{ij} changes from 0 to 1.

For our wage-education example, the marginal effect of education is:

$$\frac{\partial E[\text{wage}_i|\text{edu}_i]}{\partial \text{edu}_i} = \beta_2.$$

This theoretical population parameter can be estimated using OLS:

```
cps = read.csv("cps.csv")
lm(wage ~ education, data = cps)
```

```
Call:
lm(formula = wage ~ education, data = cps)

Coefficients:
(Intercept)    education
    -16.448        2.898
```

Interpretation: People with one more year of education are paid on average \$2.90 USD more per hour than people with one year less of education, assuming the exogeneity condition holds.

Correlation vs. Causation

The coefficient β_2 describes the **correlative relationship** between education and wages, not necessarily a causal one. To see this connection to correlation, consider the covariance of the two variables:

$$\begin{aligned} Cov(\text{wage}_i, \text{edu}_i) &= Cov(\beta_1 + \beta_2 \cdot \text{edu}_i + u_i, \text{edu}_i) \\ &= Cov(\beta_1 + \beta_2 \cdot \text{edu}_i, \text{edu}_i) + Cov(u_i, \text{edu}_i) \end{aligned}$$

The term $Cov(u_i, \text{edu}_i)$ equals zero due to the exogeneity assumption. To see this, recall that $E[u_i] = E[E[u_i|\text{edu}_i]] = 0$ by the LIE and $E[u_i\text{edu}_i] = 0$ by mean uncorrelatedness, which implies

$$Cov(u_i, \text{edu}_i) = E[u_i\text{edu}_i] - E[u_i] \cdot E[\text{edu}_i] = 0$$

The coefficient β_2 is thus proportional to the population correlation coefficient:

$$\beta_2 = \frac{Cov(\text{wage}_i, \text{edu}_i)}{Var(\text{edu}_i)} = Corr(\text{wage}_i, \text{edu}_i) \cdot \frac{sd(\text{wage}_i)}{sd(\text{edu}_i)}.$$

The marginal effect is a correlative effect and does not necessarily reveal the source of the higher wage levels for people with more education.

Regression relationships do not necessarily imply causal relationships.

People with more education may earn more for various reasons:

- They might be naturally more talented or capable
- They might come from wealthier families with better connections
- They might have access to better resources and opportunities
- Education itself might actually increase productivity and earnings

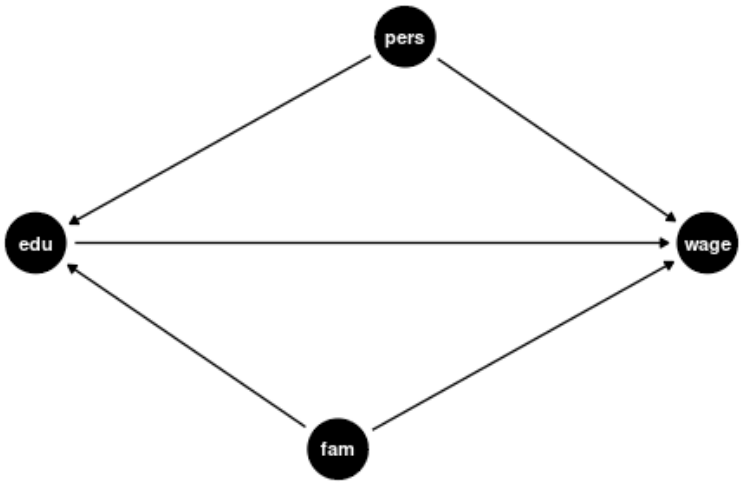


Figure 4.3: A DAG (directed acyclic graph) showing potential confounding factors in the education-wage relationship

The coefficient β_2 measures how strongly education and earnings are correlated, but this association could be due to other factors that correlate with both wages and education, such as:

- Family background (parental education, family income, ethnicity)
- Personal background (gender, intelligence, motivation)

Remember: **Correlation does not imply causation!**

Omitted Variable Bias

To understand the causal effect of an additional year of education on wages, it is crucial to consider the influence of family and personal background. These factors, if not included in our analysis, are known as **omitted variables**. An omitted variable is one that:

(i) is correlated with the dependent variable (wage_i, in this scenario)

(ii) is correlated with the regressor of interest (edu_i)

(iii) is omitted in the regression

The presence of omitted variables means that we cannot be sure that the regression relationship between education and wages is purely causal. We say that we have **omitted variable bias** for the causal effect of the regressor of interest.

The coefficient β_2 in the simple regression model measures the correlative or marginal effect, not the causal effect. This must always be kept in mind when interpreting regression coefficients.

Control Variables

We can include **control variables** in the linear regression model to reduce omitted variable bias so that we can interpret β_2 as a **ceteris paribus marginal effect** (ceteris paribus means holding other variables constant).

For example, let's include years of experience as well as racial background and gender dummy variables for Black and female:

$$\text{wage}_i = \beta_1 + \beta_2 \text{edu}_i + \beta_3 \text{exper}_i + \beta_4 \text{Black}_i + \beta_5 \text{fem}_i + u_i.$$

In this case,

$$\beta_2 = \frac{\partial E[\text{wage}_i | \text{edu}_i, \text{exper}_i, \text{Black}_i, \text{fem}_i]}{\partial \text{edu}_i}$$

is the marginal effect of education on expected wages, holding experience, race, and gender fixed.

```
lm(wage ~ education + experience + Black + female, data = cps)
```

```
Call:
lm(formula = wage ~ education + experience + Black + female,
    data = cps)

Coefficients:
(Intercept)    education   experience        Black       female
   -21.7095       3.1350       0.2443      -2.8554      -7.4363
```

Interpretation of coefficients:

- **Education**: Given the same experience, racial background, and gender, people with one more year of education are paid on average $3.14 USD more than people with one year less of education.
- **Experience**: Each additional year of experience is associated with an average wage increase of $0.24 USD per hour, holding other factors constant.
- **Black**: Black workers earn on average $2.86 USD less per hour than non-Black workers with the same education, experience, and gender.
- **Female**: Women earn on average $7.43 USD less per hour than men with the same education, experience, and racial background.

Note: This regression does not control for other unobservable characteristics (such as ability) or variables not included in the regression (such as quality of education), so omitted variable bias may still be present.

Good vs. Bad Controls

It's important to recognize that control variables are always selected with respect to a particular regressor of interest. A researcher typically focuses on estimating the effect of one specific variable (like education), and control variables must be designed specifically for this relationship.

In causal inference terminology, we can distinguish between different types of variables:

- **Confounders**: Variables that affect both the regressor of interest and the outcome. These are good controls because they help isolate the causal effect of interest.
- **Mediators**: Variables through which the regressor of interest affects the outcome. Controlling for mediators can block part of the causal effect we're trying to estimate.
- **Colliders**: Variables that are affected by both the regressor of interest and the outcome (or by factors that determine the outcome). Controlling for colliders can create spurious associations.

Confounders

Examples of **good controls** (confounders) for education are:

- Parental education level (affects both a person's education and their wage potential)
- Region of residence (geographic factors can influence education access and job markets)
- Family socioeconomic background (affects educational opportunities and wage potential)

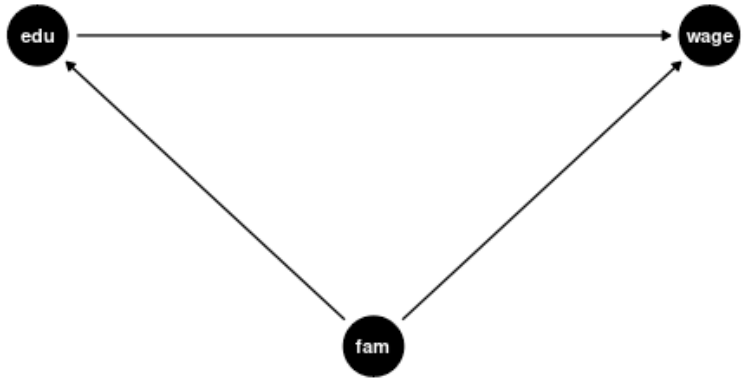


Figure 4.4: A DAG of the education-wage relationship with family confounder

Mediators and Colliders

Examples of **bad controls** include:

- **Mediators**: Variables that are part of the causal pathway from education to wages
  - Current job position (education → job position → wage)
  - Professional sector (education may determine which sector someone works in)
  - Number of professional certifications (likely a result of education level)

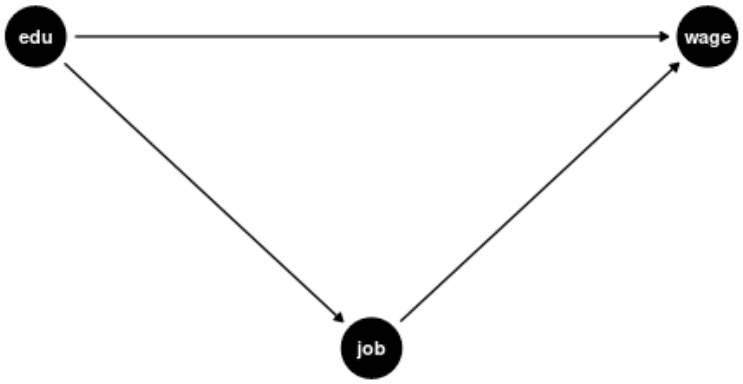


Figure 4.5: A DAG of the education-wage relationship with job position mediator

- **Colliders**: Variables affected by both education and wages (or their determinants)
  - Happiness/life satisfaction (might be affected independently by both education and wages)
  - Work-life balance (both education and wages might affect this independently)

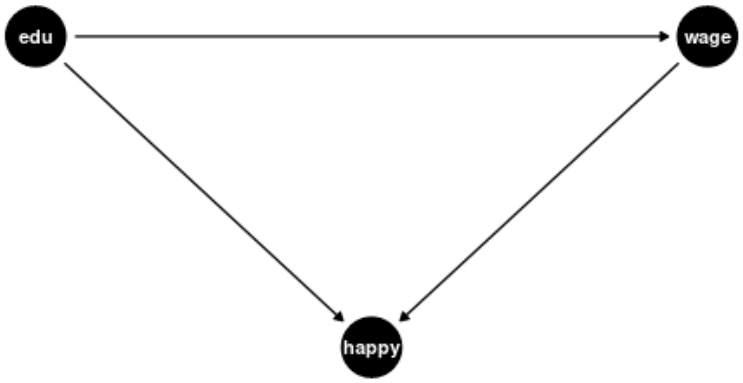


Figure 4.6: A DAG of the education-wage relationship with happiness collider

Bad controls create two problems:

1. **Statistical issue**: High correlation with the variable of interest (like education) causes high variance in the coefficient estimate (imperfect multicollinearity).
2. **Causal inference issue**: They distort the relationship we're trying to estimate by either blocking part of the causal effect (mediators) or creating artificial associations (colliders).

Good control variables are typically determined before the level of education is determined, while bad controls are often outcomes of the education process itself or are jointly determined with wages.

The appropriate choice of control variables requires not just statistical knowledge but also subject-matter expertise about the causal structure of the relationships being studied.

4.6 Application: Class Size Effect

Let's apply these concepts to a real-world research question: How does class size affect student performance?

Recall the `CASchools` dataset used in the Stock and Watson textbook, which contains information on California school characteristics:

```
data(CASchools, package = "AER")
CASchools$STR = CASchools$students/CASchools$teachers
CASchools$score = (CASchools$read+CASchools$math)/2
```

We are interested in the effect of the student-teacher ratio `STR` (class size) on the average test score `score`. Following our previous discussion on causal inference, we need to consider potential confounding factors that might affect both class sizes and test scores.

Control Strategy

Let's examine several control variables:

- `english`: proportion of students whose primary language is not English.
- `lunch`: proportion of students eligible for free/reduced-price meals.
- `expenditure`: total expenditure per pupil.

First, we should check whether these variables are correlated with both our regressor of interest (`STR`) and the outcome (`score`):

	(1)	(2)	(3)	(4)
(Intercept)	698.933	686.032	700.150	665.988
STR	−2.280	−1.101	−0.998	−0.235
english		−0.650	−0.122	−0.128
lunch			−0.547	−0.546
expenditure				0.004
Num.Obs.	420	420	420	420
R2	0.051	0.426	0.775	0.783
R2 Adj.	0.049	0.424	0.773	0.781
RMSE	18.54	14.41	9.04	8.86

```
library(dplyr)
CASchools |> select(STR, score, english, lunch, expenditure) |> cor()
```

```
                  STR      score     english       lunch expenditure
STR         1.0000000 -0.2263627  0.18764237  0.13520340 -0.61998216
score      -0.2263627  1.0000000 -0.64412381 -0.86877199  0.19127276
english     0.1876424 -0.6441238  1.00000000  0.65306072 -0.07139604
lunch       0.1352034 -0.8687720  0.65306072  1.00000000 -0.06103871
expenditure -0.6199822  0.1912728 -0.07139604 -0.06103871  1.00000000
```

The correlation matrix reveals that `english`, `lunch`, and `expenditure` are indeed correlated with both `STR` and `score`. This suggests they could be confounders that, if omitted, might bias our estimate of the class size effect.

Let's implement a control strategy, adding potential confounders one by one to see how the estimated marginal effect of class size changes:

```
fit1 = lm(score ~ STR, data = CASchools)
fit2 = lm(score ~ STR + english, data = CASchools)
fit3 = lm(score ~ STR + english + lunch, data = CASchools)
fit4 = lm(score ~ STR + english + lunch + expenditure, data = CASchools)
library(modelsummary)
mymodels = list(fit1, fit2, fit3, fit4)
modelsummary(mymodels,
             statistic = NULL,
             gof_map = c("nobs", "r.squared", "adj.r.squared", "rmse"))
```

Interpretation of Marginal Effects

Let's interpret the coefficients on STR from each model more precisely:

- **Model (1)**: Between two classes that differ by one student, the class with more students scores on average 2.280 points lower. This represents the unadjusted association without controlling for any confounding factors.
- **Model (2)**: Between two classes that differ by one student but have the same share of English learners, the larger class scores on average 1.101 points lower. Controlling for English learner status cuts the estimated effect by more than half.
- **Model (3)**: Between two classes that differ by one student but have the same share of English learners and students with reduced meals, the larger class scores on average 0.998 points lower. Adding this socioeconomic control further reduces the estimated effect slightly.
- **Model (4)**: Between two classes that differ by one student but have the same share of English learners, students with reduced meals, and per-pupil expenditure, the larger class scores on average 0.235 points lower. This represents a dramatic reduction from the previous model.

The sequential addition of controls demonstrates how sensitive the estimated marginal effect is to model specification. Each coefficient represents the partial derivative of the expected test score with respect to the student-teacher ratio, holding constant the variables included in that particular model.

Identifying Good and Bad Controls

Based on our causal framework from the previous section, we can evaluate our control variables:

- **Confounders (good controls)**: `english` and `lunch` are likely **good controls** because they represent pre-existing student characteristics that influence both class size assignments (schools might create smaller classes for disadvantaged students) and test performance.
- **Mediator (bad control)**: `expenditure` appears to be a **bad control** because it's likely a mediator in the causal pathway from class size to test scores. Smaller classes mechanically increase per-pupil expenditure through higher teacher salary costs per student.

The causal relationship can be visualized as:

Class Size → Expenditure → Test Scores

When we control for `expenditure`, we block this causal pathway and "control away" part of the effect we actually want to measure. This explains the dramatic drop in the coefficient in Model (4) and suggests this model likely underestimates the true effect of class size.

This application demonstrates the crucial importance of thoughtful control variable selection in regression analysis. The estimated marginal effect of class size on test scores varies substantially depending on which variables we control for. Based on causal reasoning, we should prefer Model (3) with the appropriate confounders but without the mediator.

4.7 Nonlinear Modeling

Polynomials

A linear dependence on wages and experience is a strong assumption. We can reasonably expect a nonlinear marginal effect of another year of experience on wages. For example, the effect may be higher for workers with 5 years of experience than for those with 40 years of experience.

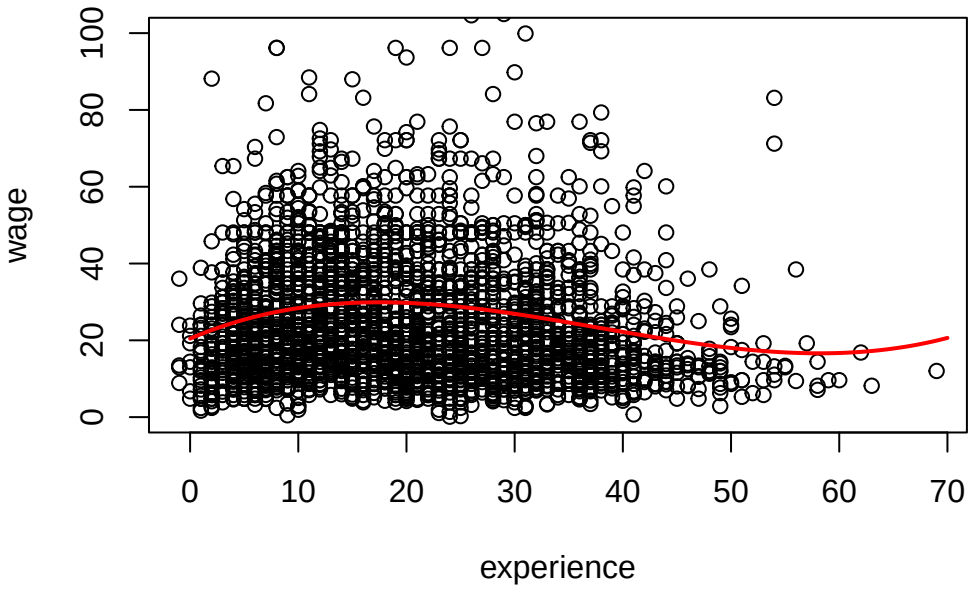
Polynomials can be used to specify a nonlinear regression function:

$$\text{wage}_i = \beta_1 + \beta_2 \text{exper}_i + \beta_3 \text{exper}_i^2 + \beta_4 \text{exper}_i^3 + u_i.$$

```
## we focus on people with Asian background only for illustration
cps.as = cps |> subset(Asian == 1)
fit = lm(wage ~ experience + I(experience^2) + I(experience^3),
         data = cps.as)
beta = fit$coefficients
beta |> round(4)
```

```
    (Intercept)      experience I(experience^2) I(experience^3)
        20.4547          1.2013         -0.0447          0.0004
```

```
## Scatterplot
plot(wage ~ experience, data = cps.as, ylim = c(0,100))
## plot the cubic function for fitted wages
curve(
  beta[1] + beta[2]*x + beta[3]*x^2 + beta[4]*x^3,
  from = 0, to = 70, add=TRUE, col='red', lwd=2
  )
```



The marginal effect depends on the years of experience:

$$\frac{\partial E[\text{wage}_i|\text{exper}_i]}{\partial \text{exper}_i} = \beta_2 + 2\beta_3 \text{exper}_i + 3\beta_4 \text{exper}_i^2.$$

For instance, the additional wage for a worker with 11 years of experience compared to a worker with 10 years of experience is on average

$$1.2013 + 2 \cdot (-0.0447) \cdot 10 + 3 \cdot 0.0004 \cdot 10^2 = 0.4273.$$

Interactions

A linear regression with interaction terms:

$$\text{wage}_i = \beta_1 + \beta_2 \text{edu}_i + \beta_3 \text{fem}_i + \beta_4 \text{marr}_i + \beta_5 (\text{marr}_i \cdot \text{fem}_i) + u_i$$

```
lm(wage ~ education + female + married + married:female, data = cps)
```

```
Call:
lm(formula = wage ~ education + female + married + married:female,
    data = cps)

Coefficients:
   (Intercept)       education          female         married  female:married
       -17.886           2.867          -3.266           7.167          -5.767
```

The marginal effect of gender depends on the person's marital status:

$$\frac{\partial E[\text{wage}_i|\text{edu}_i, \text{fem}_i, \text{marr}_i]}{\partial \text{fem}_i} = \beta_3 + \beta_5 \text{marr}_i$$

Interpretation: Given the same education, unmarried women are paid on average 3.27 USD less than unmarried men, and married women are paid on average 3.27+5.77=9.04 USD less than married men.

The marginal effect of the marital status depends on the person's gender:

$$\frac{\partial E[\text{wage}_i|\text{edu}_i, \text{fem}_i, \text{marr}_i]}{\partial \text{marr}_i} = \beta_4 + \beta_5 \text{fem}_i$$

Interpretation: Given the same education, married men are paid on average 7.17 USD more than unmarried men, and married women are paid on average 7.17-5.77=1.40 USD more than unmarried women.

Logarithms

When analyzing wage data, we often use logarithmic transformations because they help model proportional relationships and reduce the skewness of the typically right-skewed distribution of wages. A common specification is the log-linear model, where we take the logarithm of wages while keeping education in its original scale:

In the logarithmic specification

$$\log(\text{wage}_i) = \beta_1 + \beta_2 \text{edu}_i + u_i$$

we have

$$\frac{\partial E[\log(\text{wage}_i)|edu_i]}{\partial \text{edu}_i} = \beta_2.$$

This implies

$$\underbrace{\frac{\partial E[\log(\text{wage}_i)|\text{edu}_i]}{}}_{\substack{\text{absolute} \\ \text{change}}} = \beta_2 \cdot \underbrace{\partial \text{edu}_i}_{\substack{\text{absolute} \\ \text{change}}}.$$

That is, β_2 gives the average absolute change in log-wages when education changes by 1.

Another interpretation can be given in terms of relative changes. Consider the following approximation:

$$E[\text{wage}_i|\text{edu}_i] \approx \exp(E[\log(\text{wage}_i)|\text{edu}_i]).$$

The left-hand expression is the conventional conditional mean, and the right-hand expression is the geometric mean. The geometric mean is slightly smaller because $E[\log(Y)] < \log(E[Y])$, but this difference is small unless the data is highly skewed.

The marginal effect of a change in *edu* on the geometric mean of *wage* is

$$\frac{\partial exp(E[\log(\text{wage}_i)|\text{edu}_i])}{\partial \text{edu}_i} = \underbrace{exp(E[\log(\text{wage}_i)|\text{edu}_i])}_{\text{outer derivative}} \cdot \beta_2.$$

Using the geometric mean approximation from above, we get

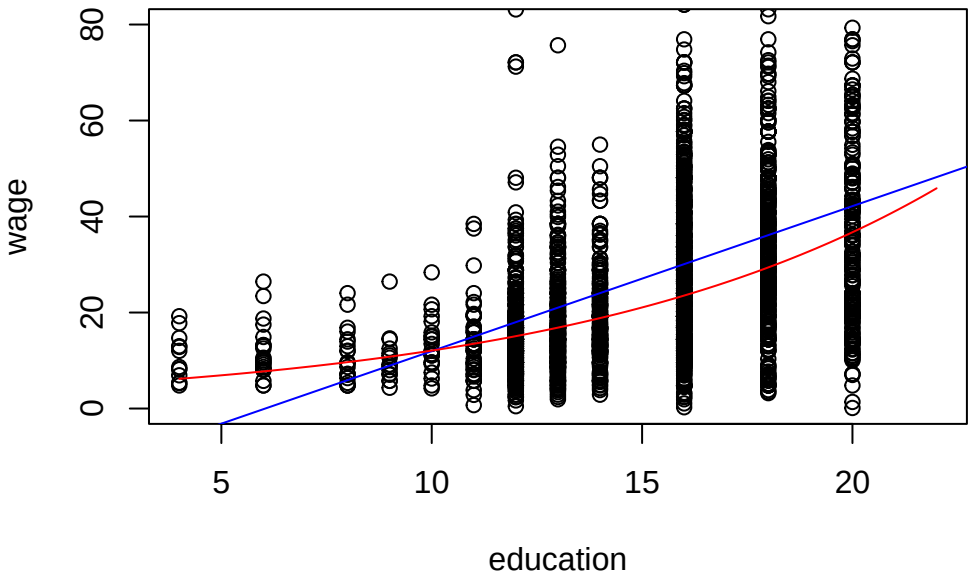
$$\underbrace{\frac{\partial E[\text{wage}_i|\text{edu}_i]}{E[\text{wage}_i|\text{edu}_i]}}_{\substack{\text{percentage} \\ \text{change}}} \approx \frac{\partial exp(E[\log(\text{wage}_i)|\text{edu}_i])}{exp(E[\log(\text{wage}_i)|\text{edu}_i])} = \beta_2 \cdot \underbrace{\partial \text{edu}_i}_{\substack{\text{absolute} \\ \text{change}}}.$$

```
linear_model = lm(wage ~ education, data = cps.as)
log_model = lm(log(wage) ~ education, data = cps.as)
log_model
```

```
Call:
lm(formula = log(wage) ~ education, data = cps.as)

Coefficients:
(Intercept)    education
     1.3783       0.1113
```

```
plot(wage ~ education, data = cps.as, ylim = c(0,80), xlim = c(4,22))
abline(linear_model, col="blue")
coef = coefficients(log_model)
curve(exp(coef[1]+coef[2]*x), add=TRUE, col="red")
```



Interpretation: A person with one more year of education has a wage that is 11.13% higher on average.

In addition to the linear-linear and log-linear specifications, we also have the linear-log specification

$$Y = \beta_1 + \beta_2 \log(X) + u$$

and the log-log specification

$$\log(Y) = \beta_1 + \beta_2 \log(X) + u.$$

Linear-log interpretation: When X is 1% higher, we observe, on average, a $0.01\beta_2$ higher Y.

Log-log interpretation: When X is 1% higher, we observe, on average, a β_2% higher Y.

4.8 R-codes

metrics-sec04.R

5 Regression Inference

Recall the linear regression framework. We observe a sample $\{(\boldsymbol{X}_i, Y_i)\}_{i=1}^n$ and assume

$$Y_i = \boldsymbol{X}_i'\boldsymbol{\beta} + u_i, \quad E[u_i \mid \boldsymbol{X}_i] = 0,$$

where $\boldsymbol{X}_i$ is a k-dimensional regressor vector (including an intercept), $\boldsymbol{\beta}$ is the unknown parameter vector, and u_i is the error term. In matrix form we have

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{u},$$

where $\boldsymbol{X}$ is the $n \times k$ design matrix (its rows are: $\boldsymbol{X}_i'$), $\boldsymbol{Y}$ is the n-vector of dependent variables, and $\boldsymbol{u}$ is the n-vector of errors.

The OLS estimator $\widehat{\boldsymbol{\beta}}$ is obtained by minimizing the sum of squared residuals:

$$\begin{aligned}
\widehat{\boldsymbol{\beta}} &= \arg\min_{\boldsymbol{b}} \sum_{i=1}^n (Y_i - \boldsymbol{X}_i'\boldsymbol{b})^2 \\
&= \Big(\sum_{i=1}^n \boldsymbol{X}_i \boldsymbol{X}_i' \Big)^{-1} \sum_{i=1}^n \boldsymbol{X}_i Y_i \\
&= (\boldsymbol{X}'\boldsymbol{X})^{-1} (\boldsymbol{X}'\boldsymbol{Y}).
\end{aligned}$$

5.1 Strict Exogeneity

The **weak exogeneity condition**

$$E[u_i \mid \boldsymbol{X}_i] = 0$$

ensures that the regressors are uncorrelated with the error at the individual observation level. However, this condition is **not sufficient** to guarantee that the OLS estimator is unbiased. It still allows for u_i to be correlated with regressors from other observations ($\boldsymbol{X}_j$ for $j \neq i$), which can lead to a biased estimation.

To ensure unbiasedness, we require the stronger condition of **strict exogeneity**:

$$E[u_i \mid \boldsymbol{X}_j] = 0 \quad \text{for each } j = 1, \dots, n,$$

or, equivalently in matrix form:

$$E[\boldsymbol{u} \mid \boldsymbol{X}] = \boldsymbol{0}.$$

Strict exogeneity requires the entire vector of errors $\boldsymbol{u}$ to be mean independent of the full regressor matrix $\boldsymbol{X}$. That is, no systematic relationship exists between any regressors and any error term across observations.

> **Note**
>
> Under i.i.d. sampling, strict exogeneity typically holds automatically: independence across observations ensures u_i is uncorrelated with $\boldsymbol{X}_j$ for $j \neq i$.

However, strict exogeneity may fail in dynamic time series settings, e.g.:

$$Y_t = \beta_1 + \beta_2 Y_{t-1} + u_t, \quad E[u_t | Y_{t-1}] = 0. \tag{5.1}$$

Here, u_t is uncorrelated with Y_{t-1}, but it is correlated through Equation 5.1 with Y_t, which is the regressor for the dependent variable Y_{t+1}:

$$Y_{t+1} = \beta_1 + \beta_2 Y_t + u_{t+1}, \quad E[u_{t+1} | Y_t] = 0. \tag{5.2}$$

Therefore the error of Equation 5.1 is correlated with the regressor of Equation 5.2, violating strict exogeneity.

5.2 Unbiasedness

To derive the **unbiasedness** of the OLS estimator, recall the model:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{u}.$$

Plugging this into the OLS formula:

$$\begin{aligned} \hat{\boldsymbol{\beta}} &= (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y} \\ &= (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{u}) \\ &= \boldsymbol{\beta} + (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{u}. \end{aligned}$$

Taking the conditional expectation:

$$E[\hat{\boldsymbol{\beta}} \mid \boldsymbol{X}] = \boldsymbol{\beta} + (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}' E[\boldsymbol{u} \mid \boldsymbol{X}].$$

Under **strict exogeneity**, $E[\boldsymbol{u} \mid \boldsymbol{X}] = \boldsymbol{0}$, so:

$$E[\hat{\boldsymbol{\beta}} \mid \boldsymbol{X}] = \boldsymbol{\beta}.$$

Taking the expectation over the sampling distribution of $\boldsymbol{X}$:

$$E[\hat{\boldsymbol{\beta}}] = E[E[\hat{\boldsymbol{\beta}} \mid \boldsymbol{X}]] = \boldsymbol{\beta}.$$

Thus, **each element** of the OLS estimator is **unbiased**:

$$E[\hat{\beta}_j] = \beta_j \quad \text{for } j = 1, \dots, k.$$

Under strict exogeneity, the OLS estimator $\hat{\boldsymbol{\beta}}$ is **unbiased**:

$$E[\hat{\boldsymbol{\beta}}] = \boldsymbol{\beta}.$$

Even when strict exogeneity fails (as in time-dependent settings) **asymptotic unbiasedness** may still hold:

$$\lim_{n \to \infty} E[\hat{\boldsymbol{\beta}}] = \boldsymbol{\beta}.$$

For time series regressions, OLS remains asymptotically unbiased if far distant future regressors are independent of current errors, and the underlying relationship remains stable over time, i.e., there are no structural changes in the conditional mean function over time.

5.3 Sampling Variance of OLS

The OLS estimator $\hat{\boldsymbol{\beta}}$ provides a **point estimate** of the unknown population parameter $\boldsymbol{\beta}$. For example, in the regression

$$\text{wage}_i = \beta_1 + \beta_2 \text{edu}_i + \beta_3 \text{fem}_i + u_i,$$

we obtain specific coefficient estimates:

```
cps = read.csv("cps.csv")
fit = lm(wage ~ education + female, data = cps)
fit |> coef()
```

```
(Intercept)   education      female
 -14.081788    2.958174   -7.533067
```

The estimate for *education* is $\hat{\beta}_2 = 2.958$. However, this point estimate tells us nothing about how far it might be from the true value β_2. That is, it does not reflect **estimation uncertainty**, which arises because $\hat{\boldsymbol{\beta}}$ depends on a finite sample that could have turned out differently.

Larger samples tend to reduce estimation uncertainty, but in practice we only observe one finite sample. To quantify this uncertainty, we study the **sampling variance** of the OLS estimator:

$$Var(\hat{\boldsymbol{\beta}} \mid \boldsymbol{X}),$$

the conditional variance of $\hat{\boldsymbol{\beta}}$ given the regressor matrix $\boldsymbol{X}$.

General formula for sampling variance of OLS:

Let $\boldsymbol{D} = Var(\boldsymbol{u} \mid \boldsymbol{X})$ be the conditional covariance matrix of the error terms. Then,

$$Var(\hat{\boldsymbol{\beta}} \mid \boldsymbol{X}) = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{D}\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}$$

This follows from

$$\hat{\boldsymbol{\beta}} = \boldsymbol{\beta} + (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{u}$$

together with the general rule that for any matrix $\boldsymbol{A}$,

$$Var(\boldsymbol{A}\boldsymbol{u}) = \boldsymbol{A}\, Var(\boldsymbol{u})\, \boldsymbol{A}'.$$

Depending on the structure of the data and the behavior of the error term, this expression takes different forms:

Homoskedasticity

Let $\{(\boldsymbol{X}_i, Y_i)\}_{i=1}^n$ be an i.i.d. sample and let the error term be **homoskedastic**, meaning

$$Var(u_i \mid \boldsymbol{X}_i) = \sigma^2 \quad \text{for all } i.$$

Homoskedasticity means that the variance of the error does not depend on the value of the regressor. For instance, in a regression of `wage` on `female`, homoskedasticity means that men and women have the same error variance. Homoskedasticity holds if the error u_i is independent of the regressor $\boldsymbol{X}_i$.

The homoskedastic error covariance matrix has the following simple form:

$$\boldsymbol{D} = \sigma^2 \boldsymbol{I}_n = \begin{pmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2 \end{pmatrix}.$$

In this case, the sampling variance simplifies to:

$$Var(\hat{\boldsymbol{\beta}} \mid \boldsymbol{X}) = \sigma^2(\boldsymbol{X}'\boldsymbol{X})^{-1}.$$

This is the Gauss-Markov setting, in which OLS is the Best Linear Unbiased Estimator (BLUE).

Heteroskedasticity

If the sample is i.i.d., but $Var(u_i \mid \boldsymbol{X}_i)$ depends on $\boldsymbol{X}_i$, the errors are **heteroskedastic**:

$$Var(u_i \mid \boldsymbol{X}_i) = \sigma^2(\boldsymbol{X}_i) = \sigma_i^2.$$

For instance, in a regression of `wage` on `gender`, the wage variability might differ between men and women.

In this case, $\boldsymbol{D}$ remains diagonal but no longer proportional to the identity matrix:

$$\boldsymbol{D} = \begin{pmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n^2 \end{pmatrix}.$$

The sampling variance becomes:

$$Var(\hat{\boldsymbol{\beta}} \mid \boldsymbol{X}) = (\boldsymbol{X}'\boldsymbol{X})^{-1} \left[\sum_{i=1}^{n} \sigma_i^2 \boldsymbol{X}_i \boldsymbol{X}_i' \right] (\boldsymbol{X}'\boldsymbol{X})^{-1}.$$

Clustered Sampling

For clustered observations we can use the notation $(\boldsymbol{X}_{ig}, Y_{ig})$ for $i = 1, \dots, n_g$ observations in cluster $g = 1, \dots, G$:

$$Y_{ig} = \boldsymbol{X}_{ig}'\boldsymbol{\beta} + u_{ig}, \quad i = 1, \dots, n_g, \quad g = 1, \dots, G.$$

We assume:

(i) Weak exogeneity within clusters: $E[u_{ig} \mid \boldsymbol{X}_g] = 0$ for all $g = 1, \dots, G$.

(ii) Independence across clusters: $(\boldsymbol{Y}_{1g}, \dots, Y_{n_g g}, \boldsymbol{X}'_{1g}, \dots, \boldsymbol{X}'_{n_g g})$ are i.i.d. for $g = 1, \dots, G$.

This together ensures strict exogenity and unbiasedness of OLS, but allow for arbitrary correlation of errors within each cluster. The covariance matrix $\boldsymbol{D}$ has a block-diagonal form:

$$\boldsymbol{D} = \begin{pmatrix} \boldsymbol{D}_1 & 0 & \cdots & 0 \\ 0 & \boldsymbol{D}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \boldsymbol{D}_G \end{pmatrix},$$

where each block $\boldsymbol{D}_g$ is an $n_g \times n_g$ matrix capturing the error covariances within cluster g:

$$\boldsymbol{D}_g = \begin{pmatrix} E[u_{1g}^2|\boldsymbol{X}] & E[u_{1g}u_{2g}|\boldsymbol{X}] & \cdots & E[u_{1g}u_{n_g g}|\boldsymbol{X}] \\ E[u_{2g}u_{1g}|\boldsymbol{X}] & E[u_{2g}^2|\boldsymbol{X}] & \cdots & E[u_{2g}u_{n_g g}|\boldsymbol{X}] \\ \vdots & \vdots & \ddots & \vdots \\ E[u_{n_g g}u_{1g}|\boldsymbol{X}] & E[u_{n_g g}u_{2g}|\boldsymbol{X}] & \cdots & E[u_{n_g g}^2|\boldsymbol{X}] \end{pmatrix}.$$

The middle part of the sandwich form of the covariance matrix $Var(\hat{\boldsymbol{\beta}} \mid \boldsymbol{X})$ becomes:

$$\boldsymbol{X}'\boldsymbol{D}\boldsymbol{X} = \sum_{g=1}^{G} E\Big[\Big(\sum_{i=1}^{n_g} \boldsymbol{X}_{ig} u_{ig}\Big)\Big(\sum_{i=1}^{n_g} \boldsymbol{X}_{ig} u_{ig}\Big)' \Big| \boldsymbol{X}\Big].$$

Time Series Data

In time series regressions, errors u_t are often **serially correlated**. A typical example is an AR(1) process:

$$u_t = \phi u_{t-1} + \varepsilon_t,$$

where $|\phi| < 1$ and ε_t is i.i.d. with mean 0 and variance σ_ε^2.

Then the autocovariance structure is:

$$Cov(u_t, u_{t-h}) = \sigma^2 \phi^h, \quad \text{for } h \geq 0,$$

where

$$\sigma^2 = \frac{\sigma_\varepsilon^2}{1 - \phi^2}.$$

The resulting covariance matrix $\boldsymbol{D}$ has a Toeplitz structure:

$$\boldsymbol{D} = \sigma^2 \begin{pmatrix} 1 & \phi & \phi^2 & \cdots & \phi^{n-1} \\ \phi & 1 & \phi & \cdots & \phi^{n-2} \\ \phi^2 & \phi & 1 & \cdots & \phi^{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi^{n-1} & \phi^{n-2} & \phi^{n-3} & \cdots & 1 \end{pmatrix}.$$

5.4 Gaussian Regression

The **Gaussian regression model** builds on the linear regression framework by adding a distributional assumption. It assumes an i.i.d. sample and that the error terms are conditionally normally distributed:

$$u_i \mid \boldsymbol{X}_i \sim \mathcal{N}(0, \sigma^2) \tag{5.3}$$

That is, conditional on the regressors, the error has mean zero (exogeneity), constant variance (homoskedasticity), and a normal distribution. This assumption implies that the OLS estimator itself is normally distributed, since it is a linear combination of normally distributed errors:

$$\hat{\boldsymbol{\beta}} \mid \boldsymbol{X} \sim \mathcal{N}(\boldsymbol{\beta}, \sigma^2(\boldsymbol{X}'\boldsymbol{X})^{-1}).$$

In particular, each standardized coefficient follows a standard normal distribution:

$$\frac{\hat{\beta}_j - \beta_j}{sd(\hat{\beta}_j \mid \boldsymbol{X})} \sim \mathcal{N}(0, 1),$$

with conditional standard deviation

$$sd(\hat{\beta}_j \mid \boldsymbol{X}) = \sigma\sqrt{(\boldsymbol{X}'\boldsymbol{X})^{-1}_{jj}}.$$

Classical Standard Errors

The conditional standard deviation of $\hat{\beta}_j$ is unknown because the population error variance σ^2 is unknown.

A **standard error** of $\hat{\beta}_j$ is an estimator of the conditional standard deviation. To construct a valid standard error under this setup, we can use the adjusted residual variance to estimate σ^2:

$$s^2_{\widehat{u}} = \frac{1}{n-k}\sum_{i=1}^{n}\hat{u}_i^2.$$

The classical standard error (valid under homoskedasticity) is defined as:

$$se_{hom}(\hat{\beta}_j) = s_{\widehat{u}}\sqrt{(\boldsymbol{X}'\boldsymbol{X})^{-1}_{jj}}.$$

Under the Gaussian assumption Equation 5.3, $\hat{\boldsymbol{\beta}}$ and $s^2_{\widehat{u}}$ are independent and $s^2_{\widehat{u}}$ has the following property:

$$\frac{(n-k)s^2_{\widehat{u}}}{\sigma^2} \sim \chi^2_{n-k}.$$

This allows us to derive the exact distribution of the standardized OLS coefficient when we replace the population standard deviation with its sample estimate (the standard error):

$$\frac{\hat{\beta}_j - \beta_j}{se_{hom}(\hat{\beta}_j \mid \boldsymbol{X})} = \frac{\hat{\beta}_j - \beta_j}{sd(\hat{\beta}_j \mid \boldsymbol{X})} \cdot \frac{\sigma}{s_{\widehat{u}}} \sim \frac{\mathcal{N}(0,1)}{\sqrt{\chi^2_{n-k}/(n-k)}} = t_{n-k}$$

This means that the OLS coefficient standardized with the homoskedastic standard error instead of the standard deviation follows a t-distribution with $n-k$ degrees of freedom.

> For a refresher on the normal and t-distribution, see
> Probability Tutorial Part 4

To estimate the full sampling covariance matrix $Var(\hat{\boldsymbol{\beta}} \mid \boldsymbol{X})$, the classical covariance matrix estimator is:

$$\widehat{\boldsymbol{V}}_{hom} = s^2_{\widehat{u}}(\boldsymbol{X}'\boldsymbol{X})^{-1}.$$

```
## classical homoskedastic covariance matrix estimator:
vcov(fit)
```

```
            (Intercept)     education        female
(Intercept)  0.18825476 -0.0127486354 -0.0089269796
education   -0.01274864  0.0009225111 -0.0002278021
female      -0.00892698 -0.0002278021  0.0284200217
```

Classical standard errors $se_{hom}(\hat{\beta}_j)$ are the square roots of the diagonal entries:

```
## classical standard errors:
sqrt(diag(vcov(fit)))
```

```
(Intercept)   education      female
 0.43388334  0.03037287  0.16858239
```

They are also displayed in parentheses in a typical regression summary table:

```
library(modelsummary)
modelsummary(fit, gof_map = "none")
```

The argument `gof_map = "none"` omits all goodness of fit statistics like R-squared and RMSE.

---

	(1)
(Intercept)	−14.082
	(0.434)
education	2.958
	(0.030)
female	−7.533
	(0.169)

Confidence Intervals

A confidence interval is a range of values that is likely to contain the true population parameter with a specified **confidence level** or **coverage probability**, often expressed as a percentage (e.g., 95%).

A $(1-\alpha)$ confidence interval for β_j is an interval $I_{1-\alpha}$ such that

$$P(\beta_j \in I_{1-\alpha}) = 1-\alpha.$$

Under the Gaussian assumption Equation 5.3, this property is satisfied for the classical homoskedastic confidence interval:

$$I_{1-\alpha} = \left[\widehat{\beta}_j - t_{n-k,1-\alpha/2} \cdot se_{hom}(\widehat{\beta}_j); \widehat{\beta}_j + t_{n-k,1-\alpha/2} \cdot se_{hom}(\widehat{\beta}_j)\right],$$

where $t_{n-k,1-\alpha/2}$ is the $1-\alpha/2$-quantile from the t-distribution with $n-k$ degrees of freedom. Common coverage probabilities are 0.90, 0.95, 0.99, and 0.999.

Table 5.1: Student's t-distribution quantiles

df	0.95	0.975	0.995	0.9995
1	6.31	12.71	63.66	636.6
2	2.92	4.30	9.92	31.6
3	2.35	3.18	5.84	12.9
5	2.02	2.57	4.03	6.87
10	1.81	2.23	3.17	4.95
20	1.72	2.09	2.85	3.85
50	1.68	2.01	2.68	3.50
100	1.66	1.98	2.63	3.39
$\to \infty$	1.64	1.96	2.58	3.29

	(1)
(Intercept)	−14.082
	[−14.932, −13.231]
education	2.958
	[2.899, 3.018]
female	−7.533
	[−7.863, −7.203]

The last row (indicated by $\to \infty$) shows the quantiles of the standard normal distribution $\mathcal{N}(0, 1)$.

You can display 95% confidence intervals in the `modelsummary` output using the `conf.int` argument:

```
modelsummary(fit, gof_map = "none", statistic = "conf.int")
```

Note: the confidence interval is **random**, while the parameter β_j is **fixed but unknown**.

A correct interpretation of a 95% confidence interval is:

- If we were to repeatedly draw samples and construct a 95% confidence interval from each sample, about 95% of these intervals would contain the true parameter.

Common misinterpretations to avoid:

- "There is a 95% probability that the true value lies in this interval."
- "We are 95% confident this interval contains the true parameter."

These mistakes incorrectly treat the parameter as random and the interval as fixed. In reality, it's the other way around.

A 95% confidence interval should be understood as a coverage probability: Before observing the data, there is a 95% probability that the random interval will cover the true parameter.

A helpful visualization:

https://rpsychologist.com/d3/ci/

Limitations of the Gaussian Approach

The Gaussian regression framework assumes:

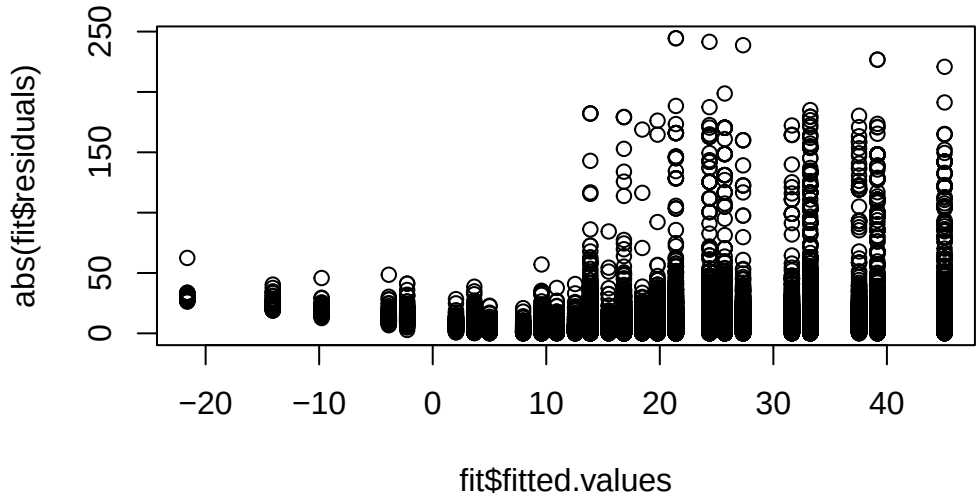
- Weak exogeneity: $E[u_i \mid \boldsymbol{X}_i] = 0$
- I.i.d. sample: $\{(Y_i, \boldsymbol{X}_i)\}_{i=1}^n$
- Homoskedastic, normally distributed errors: $u_i|\boldsymbol{X}_i \sim \mathcal{N}(0, \sigma^2)$
- $\boldsymbol{X}'\boldsymbol{X}$ is invertible (i.e. $\boldsymbol{X}$ has full rank)

While mathematically convenient, these assumptions are often violated in practice. In particular, the normality assumption implies homoskedasticity and that the conditional distribution of Y_i given $\boldsymbol{X}_i$ is normal, which is an unrealistic scenario in many economic applications.

Historically, homoskedasticity has been treated as the "default" assumption and heteroskedasticity as a special case. But in empirical work, **heteroskedasticity is the norm**.

A plot of the absolute value of the residuals against the fitted values shows that individuals with predicted wages around 10 USD exhibit residuals with lower variance compared to those with higher predicted wage levels. Hence, the homoskedasticity assumption is implausible:

```
# Plot of absolute residuals against fitted values
plot(abs(fit$residuals) ~ fit$fitted.values)
```

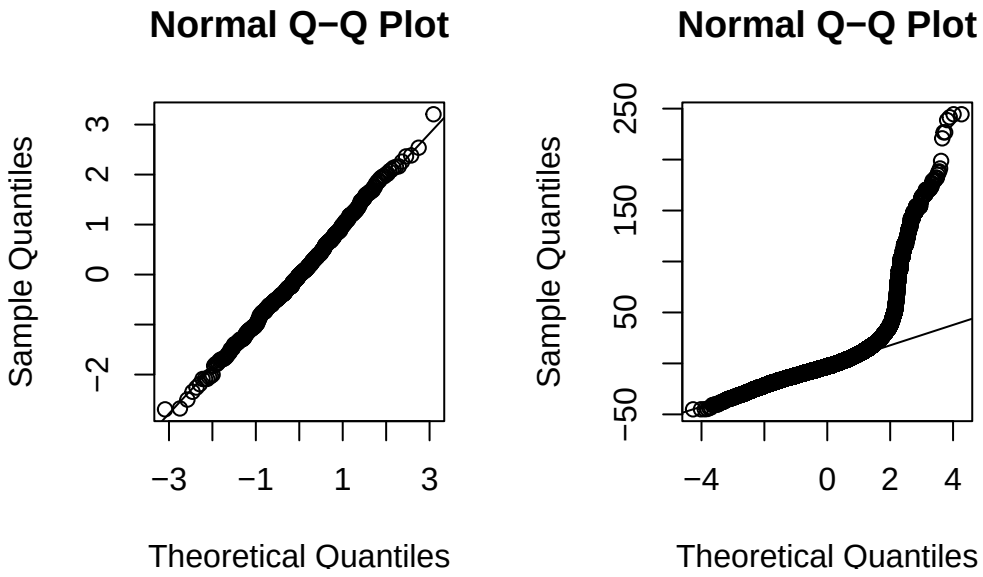


The Q-Q-plot is a graphical tool to help us assess if the errors are conditionally normally distributed.

Let $\hat{u}_{(i)}$ be the sorted residuals (i.e. $\hat{u}_{(1)} \leq ... \leq \hat{u}_{(n)}$). The Q-Q-plot plots the sorted residuals $\hat{u}_{(i)}$ against the $((i-0.5)/n)$-quantiles of the standard normal distribution.

If the residuals are lined well on the straight dashed line, there is indication that the distribution of the residuals is close to a normal distribution.

```
set.seed(123)
par(mfrow = c(1,2))
## auxiliary regression with simulated normal errors:
fit.aux = lm(rnorm(500) ~ 1)
## Q-Q-plot of the residuals of the auxiliary regression:
qqnorm(residuals(fit.aux))
qqline(residuals(fit.aux))
## Q-Q-plot of the residuals of the wage regression:
qqnorm(residuals(fit))
qqline(residuals(fit))
```



In the left plot you see the Q-Q-plot for an example with simulated normally distributed errors, where the Gaussian regression assumption is satisfied.

The right plot indicates that, in our regression of `wage` on `education` and `female`, the normality assumption is implausible.

5.5 Heteroskedastic Linear Model

The classical approach to regression relies on strong distributional assumptions: normality and homoskedasticity of the errors. While this enables exact inference in small samples, it is rarely justified in empirical applications.

The modern econometric approach avoids such assumptions and instead relies on asymptotic approximations under weaker conditions (i.e., finite kurtosis instead of normality and homoskedasticity).

Heteroskedastic Linear Model

We assume that the sample $\{(Y_i, \boldsymbol{X}_i')\}_{i=1}^n$ satisfies the linear regression equation

$$Y_i = \boldsymbol{X}_i'\boldsymbol{\beta} + u_i, \quad i = 1, \dots, n,$$

under the following conditions:

- **(A1)** $E[u_i|\boldsymbol{X}_i] = 0$ (weak exogeneity)
- **(A2)** $\{(Y_i, \boldsymbol{X}_i')\}_{i=1}^n$ is an i.i.d. sample (random sampling)

- **(A3)** $kur(Y_i) < \infty$ and $kur(X_{ij}) < \infty$ for all $j = 1, \dots, k$
  (bounded kurtosis: large outliers are unlikely)

- **(A4)** $\sum_{i=1}^{n} \boldsymbol{X}_i \boldsymbol{X}_i'$ is invertible (OLS is well defined)

Under heteroskedasticity, the error variance may depend on the regressor:

$$\sigma_i^2 = \mathrm{Var}(u_i \mid \boldsymbol{X}_i),$$

and the conditional standard deviation of $\hat{\beta}_j$ is

$$sd(\hat{\beta}_j \mid \boldsymbol{X}) = \sqrt{\left[(\boldsymbol{X}'\boldsymbol{X})^{-1}\Big(\sum_{i=1}^{n} \sigma_i^2 \boldsymbol{X}_i \boldsymbol{X}_i'\Big)(\boldsymbol{X}'\boldsymbol{X})^{-1}\right]_{jj}}.$$

Unlike in the Gaussian case, the standardized OLS coefficient does **not** follow a standard normal distribution in finite samples:

$$\frac{\hat{\beta}_j - \beta_j}{sd(\hat{\beta}_j \mid \boldsymbol{X})} \nsim \mathcal{N}(0,1).$$

However, for large samples, the **central limit theorem** guarantees that the OLS estimator is **asymptotically normal**:

$$\frac{\hat{\beta}_j - \beta_j}{sd(\hat{\beta}_j \mid \boldsymbol{X})} \xrightarrow{d} \mathcal{N}(0,1) \quad \text{as } n \to \infty.$$

This result holds because the OLS estimator can be expressed as:

$$\begin{aligned}
\sqrt{n}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) &= \sqrt{n}\left(\sum_{i=1}^{n} \boldsymbol{X}_i \boldsymbol{X}_i'\right)^{-1} \sum_{i=1}^{n} \boldsymbol{X}_i u_i \\
&= \left(\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}_i \boldsymbol{X}_i'\right)^{-1} \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \boldsymbol{X}_i u_i,
\end{aligned}$$

where:

- By the **law of large numbers**:

$$\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}_i \boldsymbol{X}_i' \xrightarrow{p} E[\boldsymbol{X}_i \boldsymbol{X}_i'] = \boldsymbol{Q},$$

- And by the **central limit theorem**:

$$\frac{1}{\sqrt{n}} \sum_{i=1}^{n} \boldsymbol{X}_i u_i \xrightarrow{d} \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Omega}), \quad \text{where } \boldsymbol{\Omega} = E[u_i^2 \boldsymbol{X}_i \boldsymbol{X}_i'].$$

For more details on stochastic convergence and the central limit theorem, see Probability Tutorial Part 4

Asymptotic Distribution of OLS Estimator

Under the heteroskedastic linear model:

$$\sqrt{n}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \xrightarrow{d} \mathcal{N}(\mathbf{0},\, \boldsymbol{Q}^{-1}\boldsymbol{\Omega}\boldsymbol{Q}^{-1}),$$

where $\boldsymbol{Q} = E[\boldsymbol{X}_i\boldsymbol{X}_i']$ and $\boldsymbol{\Omega} = E[u_i^2\boldsymbol{X}_i\boldsymbol{X}_i']$.

This asymptotic distribution forms the basis for **heteroskedasticity-robust inference**.

5.6 Heteroskedasticity-Robust Standard Errors

The asymptotic distribution of the OLS estimator under heteroskedasticity depends on two population matrices:

- $\boldsymbol{Q} = E[\boldsymbol{X}_i\boldsymbol{X}_i']$, and
- $\boldsymbol{\Omega} = E[u_i^2\boldsymbol{X}_i\boldsymbol{X}_i']$

While $\boldsymbol{Q}$ can be consistently estimated by its sample counterpart,

$$\widehat{\boldsymbol{Q}} = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{X}_i\boldsymbol{X}_i',$$

estimating $\boldsymbol{\Omega}$ is more challenging because the error terms u_i are unobserved.

To overcome this, we replace the unobserved u_i with the OLS residuals:

$$\hat{u}_i = Y_i - \boldsymbol{X}_i'\hat{\boldsymbol{\beta}}.$$

This yields a consistent estimator of $\boldsymbol{\Omega}$:

$$\widehat{\boldsymbol{\Omega}} = \frac{1}{n}\sum_{i=1}^{n}\hat{u}_i^2\boldsymbol{X}_i\boldsymbol{X}_i'.$$

Substituting into the asymptotic variance formula, we obtain the **heteroskedasticity-consistent covariance matrix estimator**, also known as the **White estimator** (White, 1980):

White (HC0) Estimator

$$\widehat{\boldsymbol{V}}_{hc0} = (\boldsymbol{X}'\boldsymbol{X})^{-1} \left(\sum_{i=1}^{n} \hat{u}_i^2 \boldsymbol{X}_i \boldsymbol{X}_i' \right) (\boldsymbol{X}'\boldsymbol{X})^{-1}$$

This estimator remains consistent for $Var(\hat{\boldsymbol{\beta}} \mid \boldsymbol{X})$ even if the errors are heteroskedastic. However, it can be biased downward in small samples.

HC1 Correction

To reduce small-sample bias, MacKinnon and White (1985) proposed the **HC1 correction**, which rescales the estimator using a degrees-of-freedom adjustment:

$$\widehat{\boldsymbol{V}}_{hc1} = \frac{n}{n-k} \cdot (\boldsymbol{X}'\boldsymbol{X})^{-1} \left(\sum_{i=1}^{n} \hat{u}_i^2 \boldsymbol{X}_i \boldsymbol{X}_i' \right) (\boldsymbol{X}'\boldsymbol{X})^{-1}.$$

The **HC1 standard error** for the j-th coefficient is then:

$$se_{hc1}(\hat{\beta}_j) = \sqrt{[\widehat{\boldsymbol{V}}_{hc1}]_{jj}}.$$

These standard errors are widely used in applied work because they are valid under general forms of heteroskedasticity and easy to compute. Most statistical software (including R and Stata) uses HC1 by default when robust inference is requested.

Robust Confidence Intervals

Using heteroskedasticity-robust standard errors, we can construct confidence intervals that remain valid under heteroskedasticity.

For large samples, a $(1-\alpha)$ confidence interval for β_j is:

$$I_{1-\alpha} = \left[\hat{\beta}_j \pm z_{1-\alpha/2} \cdot se_{hc1}(\hat{\beta}_j) \right],$$

where $z_{1-\alpha/2}$ is the standard normal critical value (e.g., $z_{0.975} = 1.96$ for a 95% interval).

For moderate sample sizes, using a t-distribution with $n-k$ degrees of freedom gives better finite-sample performance:

$$I_{1-\alpha} = \left[\hat{\beta}_j \pm t_{n-k,1-\alpha/2} \cdot se_{hc1}(\hat{\beta}_j)\right].$$

These robust intervals satisfy the asymptotic coverage property:

$$\lim_{n\to\infty} P(\beta_j \in I_{1-\alpha}) = 1-\alpha.$$

> **i** Why software uses t-quantiles:
>
> Under heteroskedasticity, there's no theoretical justification for using t-quantiles instead of normal ones. However, most software use t_{n-k} by default to match the homoskedastic case and improve finite-sample performance. For large samples, this makes little difference, as t-quantiles converge to standard normal quantiles as degrees of freedom grow large.

The `fixest` package provides the `feols` function to estimate regression models with heteroskedasticity-robust standard errors. The `vcov` argument allows you to specify the type of covariance matrix estimator to use.

```
library(fixest)
fit.hom = feols(wage ~ education + female, data = cps, vcov = "iid")
fit.het = feols(wage ~ education + female, data = cps, vcov = "hc1")
```

```
mymodels = list(
  "Homoskedastic" = fit.hom,
  "Heteroskedastic" = fit.het
)
## Standard error comparison:
modelsummary(mymodels)
```

```
## Confidence interval comparison:
modelsummary(mymodels, statistic = "conf.int")
```

AIC (Akaike Information Criterion) and BIC (Bayesian Information Criterion) are statistical measures that evaluate model quality by balancing goodness-of-fit against complexity. A smaller value indicates a better model. In this example we see the same values for both models because the regression equations are the same and only the standard errors differ.

	Homoskedastic	Heteroskedastic
(Intercept)	−14.082	−14.082
	(0.434)	(0.500)
education	2.958	2.958
	(0.030)	(0.040)
female	−7.533	−7.533
	(0.169)	(0.162)
Num.Obs.	50 742	50 742
R2	0.180	0.180
R2 Adj.	0.180	0.180
AIC	441 515.9	441 515.9
BIC	441 542.4	441 542.4
RMSE	18.76	18.76
Std.Errors	IID	Heteroskedasticity-robust

	Homoskedastic	Heteroskedastic
(Intercept)	−14.082	−14.082
	[−14.932, −13.231]	[−15.062, −13.102]
education	2.958	2.958
	[2.899, 3.018]	[2.880, 3.037]
female	−7.533	−7.533
	[−7.863, −7.203]	[−7.850, −7.216]
Num.Obs.	50 742	50 742
R2	0.180	0.180
R2 Adj.	0.180	0.180
AIC	441 515.9	441 515.9
BIC	441 542.4	441 542.4
RMSE	18.76	18.76
Std.Errors	IID	Heteroskedasticity-robust

5.7 R-codes

metrics-sec05.R

6 Robust Testing

In applied regression analysis, we often want to assess whether a regressor has a statistically significant relationship with the outcome variable (conditional on other regressors).

6.1 t-Test

The most common hypothesis test evaluates whether a regression coefficient equals zero:

$$H_0 : \beta_j = 0 \quad \text{vs.} \quad H_1 : \beta_j \neq 0.$$

This corresponds to testing whether the marginal effect of the regressor X_{ij} on the outcome Y_i is zero, holding other regressors constant.

We use the t-statistic:

$$T_j = \frac{\hat{\beta}_j}{se(\hat{\beta}_j)},$$

where $se(\hat{\beta}_j)$ is a standard error.

You may use the classical standard error if you have strong evidence that the errors are homoskedastic. However, in most economic applications, heteroskedasticity-robust standard errors are more reliable.

Under the null, T_j follows approximately a t_{n-k} distribution. We reject H_0 at the significance level α if:

$$|T_j| > t_{n-k,1-\alpha/2}.$$

This decision rule is equivalent to checking whether the confidence interval for β_j includes 0:

- Reject H_0 if 0 lies **outside** the $1-\alpha$ confidence interval
- Fail to reject (accept) H_0 if 0 lies **inside** the $1-\alpha$ confidence interval

---

6.2 p-Value

The **p-value** is a criterion to reach a hypothesis test decision conveniently:

$$\begin{aligned} \text{reject } H_0 \quad & \text{if p-value} < \alpha \\ \text{do not reject } H_0 \quad & \text{if p-value} \geq \alpha \end{aligned}$$

Formally, the p-value represents the probability of observing a test statistic as extreme or more extreme than the one we computed, assuming H_0 is true. For the t-test, the p-value is:

$$p\text{-value} = P(|T| > |T_j| \mid H_0 \text{ is true})$$

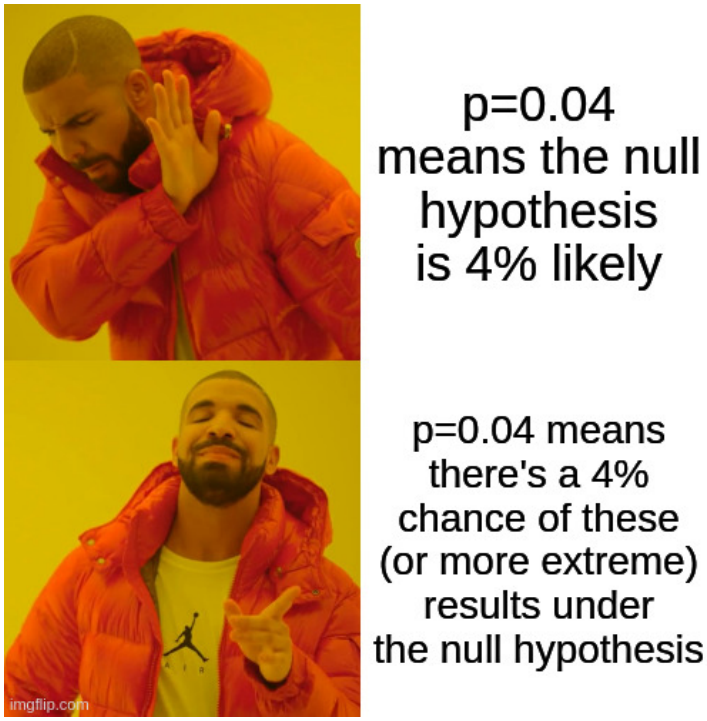
Here, T is a random variable following the null distribution $Z \sim t_{n-k}$, and T_j is the observed value of the test statistic.

Another way of representing the p-values of a t-test is:

$$p\text{-value} = 2(1 - F_{t_{n-k}}(|T_j|)),$$

where $F_{t_{n-k}}$ is the cumulative distribution function (CDF) of the t_{n-k}-distribution.

A common misinterpretation of p-values is treating them as the probability that the null hypothesis is being true. This is incorrect. The p-value is not a statement about the probability of the null hypothesis itself.



The correct interpretation is that the p-value represents the probability of observing a test statistic at least as extreme as the one calculated from our sample, assuming that the null hypothesis is true.

In other words, a p-value of 0.04 means:

- NOT "There's a 4% chance that the null hypothesis is true"
- INSTEAD "If the null hypothesis were true, there would be a 4% chance of observing a test statistic this extreme or more extreme"

Small p-values indicate that the observed data would be unlikely under the null hypothesis, which leads us to reject the null in favor of the alternative. However, they do not tell us the probability that our alternative hypothesis is correct, nor do they directly measure the magnitude or significance of the marginal effect.

> **i** Relation to Confidence Intervals:
>
> Zero lies outside the $(1-\alpha)$ confidence interval for β_j if and only if the p-value for testing $H_0 : \beta_j = 0$ is less than α.

6.3 Significance Stars

Regression tables often use asterisks to indicate levels of statistical significance. Stars summarize statistical significance by comparing the t-statistic to critical values (or equivalently, the p-value or whether 0 is covered by the confidence interval)

The convention within `R` is:

Stars	p-value	t-statistic	Confidence interval
***	$p < 0.001$	$\lvert T_j \rvert > t_{n-k,0.995}$	0 outside $I_{0.999}$
**	$0.001 \leq p < 0.01$	$t_{n-k,0.995} \geq \lvert T_j \rvert > t_{n-k,0.975}$	0 outside $I_{0.99}$, but inside $I_{0.999}$
*	$0.01 \leq p < 0.05$	$t_{n-k,0.975} \geq \lvert T_j \rvert > t_{n-k,0.95}$	0 outside $I_{0.95}$, but inside $I_{0.99}$

	(1)	(2)
(Intercept)	−14.082***	−14.082***
	(0.434)	(0.500)
education	2.958***	2.958***
	(0.030)	(0.040)
female	−7.533***	−7.533***
	(0.169)	(0.162)
Num.Obs.	50 742	50 742
R2	0.180	0.180
R2 Adj.	0.180	0.180
AIC	441 515.9	441 515.9
BIC	441 542.4	441 542.4
RMSE	18.76	18.76
Std.Errors	IID	Heteroskedasticity-robust

+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001

Significance Stars Convention

Note that most economists use the following significance levels: `***` for 1%, `**` for 5%, and `*` for 10%. In this lecture, we follow the convention of `R`, which uses the significance levels `***` for 0.1%, `**` for 1%, and `*` for 5%.

Regression Tables

Let's revisit the regression of `wage` on `education` and `female`.

```
library(fixest)
library(modelsummary)
cps = read.csv("cps.csv")
fit.hom = feols(wage ~ education + female, data = cps, vcov = "iid")
fit.het = feols(wage ~ education + female, data = cps, vcov = "hc1")
mymodels = list(fit.hom, fit.het)
modelsummary(mymodels, stars = TRUE)
```

To see the exact t-statistics and p-values, you can use the `summary()` function:

```
summary(fit.hom)
```

```
OLS estimation, Dep. Var.: wage
Observations: 50,742
Standard-errors: IID
             Estimate Std. Error  t value  Pr(>|t|)
(Intercept) -14.08179   0.433883 -32.4552 < 2.2e-16 ***
education     2.95817   0.030373  97.3953 < 2.2e-16 ***
female       -7.53307   0.168582 -44.6848 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 18.8   Adj. R2: 0.179696
```

```
summary(fit.het)
```

```
OLS estimation, Dep. Var.: wage
Observations: 50,742
Standard-errors: Heteroskedasticity-robust
             Estimate Std. Error  t value  Pr(>|t|)
(Intercept) -14.08179   0.500078 -28.1592 < 2.2e-16 ***
education     2.95817   0.040110  73.7512 < 2.2e-16 ***
female       -7.53307   0.161644 -46.6027 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 18.8   Adj. R2: 0.179696
```

All p-values are super small: `2.2e-16` means $2.2 \cdot 10^{-16}$ (15 zeros after the decimal point, followed by 22).

Let's also revisit the `CASchools` dataset and examine four regression models on test scores.

```
library(AER)
data(CASchools, package = "AER")
CASchools$STR = CASchools$students/CASchools$teachers
CASchools$score = (CASchools$read + CASchools$math)/2

fitA = feols(score ~ STR, data = CASchools)
fitB = feols(score ~ STR + english, data = CASchools)
fitC = feols(score ~ STR + english + lunch, data = CASchools)
fitD = feols(score ~ STR + english + lunch + expenditure, data = CASchools)
```

	(1)	(2)	(3)	(4)
(Intercept)	698.933***	686.032***	700.150***	665.988***
	(9.467)	(7.411)	(4.686)	(9.460)
STR	−2.280***	−1.101**	−0.998***	−0.235
	(0.480)	(0.380)	(0.239)	(0.298)
english		−0.650***	−0.122***	−0.128***
		(0.039)	(0.032)	(0.032)
lunch			−0.547***	−0.546***
			(0.022)	(0.021)
expenditure				0.004***
				(0.001)
Num.Obs.	420	420	420	420
R2	0.051	0.426	0.775	0.783
R2 Adj.	0.049	0.424	0.773	0.781
AIC	3648.5	3439.1	3049.0	3034.1
BIC	3656.6	3451.2	3065.2	3054.3
RMSE	18.54	14.41	9.04	8.86
Std.Errors	IID	IID	IID	IID

+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001

Classical (Homoskedastic) Standard Errors

```
mymodels = list(fitA, fitB, fitC, fitD)
modelsummary(mymodels, stars = TRUE, vcov = "iid")
```

Robust (HC1) Standard Errors

```
mymodels = list(fitA, fitB, fitC, fitD)
modelsummary(mymodels, stars = TRUE, vcov = "HC1")
```

	(1)	(2)	(3)	(4)
(Intercept)	698.933***	686.032***	700.150***	665.988***
	(10.364)	(8.728)	(5.568)	(10.377)
STR	−2.280***	−1.101*	−0.998***	−0.235
	(0.519)	(0.433)	(0.270)	(0.325)
english		−0.650***	−0.122***	−0.128***
		(0.031)	(0.033)	(0.032)
lunch			−0.547***	−0.546***
			(0.024)	(0.023)
expenditure				0.004***
				(0.001)
Num.Obs.	420	420	420	420
R2	0.051	0.426	0.775	0.783
R2 Adj.	0.049	0.424	0.773	0.781
AIC	3648.5	3439.1	3049.0	3034.1
BIC	3656.6	3451.2	3065.2	3054.3
RMSE	18.54	14.41	9.04	8.86
Std.Errors	HC1	HC1	HC1	HC1

+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001

Interpretation of `STR` coefficient:

- Models A–C: The coefficient is negative and statistically significant. However, when using robust standard errors, the coefficient in model B becomes only weakly significant.
- Model D: The coefficient remains negative but becomes insignificant when controlling for `expenditure`.

As discussed earlier, `expenditure` is a **bad control** in this context and should not be used to estimate a ceteris paribus effect of class size on test scores.

6.4 Testing for Heteroskedasticity: Breusch-Pagan Test

Classical standard errors should only be used if you have statistical evidence that the errors are homoskedastic. A statistical test for this is the **Breusch-Pagan Test**.

Under homoskedasticity, the variance of the error term is constant and does not depend on the values of the regressors:

$$Var(u_i \mid \boldsymbol{X}_i) = \sigma^2 \quad \text{(constant)}.$$

To test this assumption, we perform an auxiliary regression of the squared residuals on the original regressors:

$$\hat{u}_i^2 = \boldsymbol{X}_i'\boldsymbol{\gamma} + v_i, \quad i = 1, \dots, n,$$

where:

- $\hat{u}_i$ are the OLS residuals from the original model,
- $\boldsymbol{\gamma}$ are auxiliary coefficients,
- v_i is the error term in the auxiliary regression.

If homoskedasticity holds, the regressors should not explain any variation in $\hat{u}_i^2$, which means the auxiliary regression should have low explanatory power.

Let R^2_{aux} be the R-squared from this auxiliary regression. Then, the **Breusch–Pagan (BP) test statistic** is:

$$BP = n \cdot R^2_{\text{aux}}$$

Under the null hypothesis of homoskedasticity,

$$H_0 : Var(u_i \mid \boldsymbol{X}_i) = \sigma^2,$$

the test statistic follows an asymptotic chi-squared distribution with $k-1$ degrees of freedom:

$$BP \xrightarrow{d} \chi^2_{k-1}$$

We **reject** H_0 at significance level α if:

$$BP > \chi^2_{1-\alpha,\, k-1}.$$

This basic variant of the BP test is Koenker's version of the test. Other variants include further nonlinear transformations of the regressors.

In R, the test is implemented via the `bptest()` function from the **AER** package. Unfortunately, the `bptest()` function does not work directly with `feols` objects, so we need to estimate the model first with `lm()`:

```
fit = lm(wage ~ education + female, data = cps)
bptest(fit)
```

```
    studentized Breusch-Pagan test

data:  fit
BP = 1070.3, df = 2, p-value < 2.2e-16
```

In the wage regression the BP test clearly rejects H_0, which is strong statistical evidence that the errors are heteroskedastic.

Let's apply the test to the `CASchools` model:

```
lm(score ~ STR + english, data = CASchools) |> bptest()
```

```
    studentized Breusch-Pagan test

data:  lm(score ~ STR + english, data = CASchools)
BP = 29.501, df = 2, p-value = 3.926e-07
```

```
lm(score ~ STR + english + lunch, data = CASchools) |> bptest()
```

```
    studentized Breusch-Pagan test

data:  lm(score ~ STR + english + lunch, data = CASchools)
BP = 9.9375, df = 3, p-value = 0.0191
```

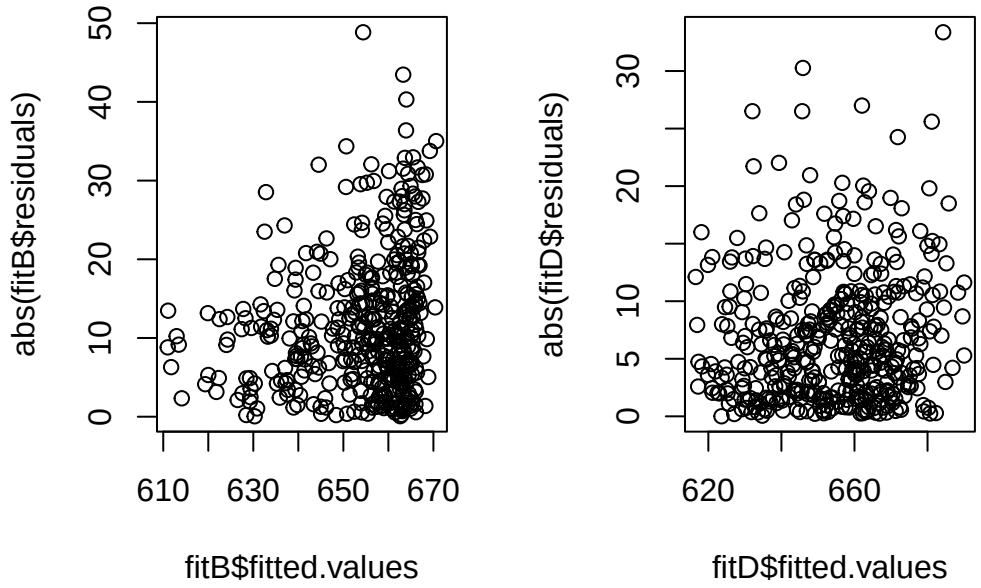
```
lm(score ~ STR + english + lunch + expenditure, data = CASchools) |> bptest()
```

```
    studentized Breusch-Pagan test

data:  lm(score ~ STR + english + lunch + expenditure, data = CASchools)
BP = 5.9649, df = 4, p-value = 0.2018
```

In the regression of `score` on `STR` and `english` there is strong statistical evidence that errors are heteroskedastic, whereas when adding `lunch` and `expenditure` there is no evidence of heteroskedasticity. See the difference in the absolute residuals against fitted values plot:

```
par(mfrow = c(1,2))
plot(abs(fitB$residuals) ~ fitB$fitted.values)
plot(abs(fitD$residuals) ~ fitD$fitted.values)
```



The heteroskedasticity pattern in model (2) likely occurred because of a nonlinear dependence of the omitted variables `lunch` and `expenditure` with the included regressors `STR` and `english`. The inclusion of these variables in model (4) eliminated the heteroskedasticity (apparent heteroskedasticity). Therefore, heteroskedasticity is sometimes a sign of model misspecification.

6.5 Testing for Normality: Jarque–Bera Test

A general property of a normally distributed variable is that it has zero skewness and kurtosis of three. In the Gaussian regression model, this implies:

$$u_i|\boldsymbol{X}_i \sim \mathcal{N}(0, \sigma^2) \quad \Rightarrow \quad E[u_i^3] = 0, \quad E[u_i^4] = 3\sigma^4.$$

The **sample skewness** and **sample kurtosis** of the OLS residuals are:

$$\widehat{\text{ske}}(\hat{\boldsymbol{u}}) = \frac{1}{n\hat{\sigma}_{\hat{u}}^3} \sum_{i=1}^{n} \hat{u}_i^3, \quad \widehat{\text{kur}}(\hat{\boldsymbol{u}}) = \frac{1}{n\hat{\sigma}_{\hat{u}}^4} \sum_{i=1}^{n} \hat{u}_i^4$$

A joint test for normality — assessing both skewness and kurtosis — is the **Jarque–Bera (JB) test**, with statistic:

$$JB = n \left(\frac{1}{6} \widehat{\text{ske}}(\hat{\boldsymbol{u}})^2 + \frac{1}{24} (\widehat{\text{kur}}(\hat{\boldsymbol{u}}) - 3)^2 \right)$$

Under the null hypothesis of normal errors, this test statistic is asymptotically chi-squared distributed:

$$JB \xrightarrow{d} \chi_2^2$$

We reject H_0 at level α if:

$$JB > \chi_{1-\alpha,\,2}^2.$$

In R, we can apply the test using the `moments` package:

```
library(moments)
jarque.test(fitD$residuals)
```

```
    Jarque-Bera Normality Test

data:  fitD$residuals
JB = 8.9614, p-value = 0.01133
alternative hypothesis: greater
```

Although the Breusch–Pagan test does not reject homoskedasticity for `fitD` (so classical standard errors are valid asymptotically), the JB rejects the null hypothesis of normal errors at the 5% level and provides statistical evidence that the errors are not normally distributed.

This means that exact inference based on t-distributions is not valid in finite samples, and confidence intervals or t-test results give only large sample approximations.

In econometrics, asymptotic large sample approximations have become the convention because exact finite sample inference is rarely feasible.

6.6 Joint Hypothesis Testing

So far, we've tested whether a single coefficient is zero. But often we want to test **multiple restrictions simultaneously**, such as whether a group of variables has a joint effect.

The **joint exclusion** hypothesis formulates the null hypothesis that a set of coefficients or linear combinations of coefficients are equal to zero:

$$H_0 : \boldsymbol{R}\boldsymbol{\beta} = \mathbf{0}$$

where:

- $\boldsymbol{R}$ is a $q \times k$ restriction matrix,
- $\mathbf{0}$ is the $q \times 1$ vector of zeros,
- q is the number of restrictions.

Consider for example the `score` on `STR` regression with interaction effects:

$$\text{score}_i = \beta_1 + \beta_2 \text{STR}_i + \beta_3 \text{HiEL}_i + \beta_4 \text{STR}_i \cdot \text{HiEL}_i + u_i.$$

```
## Create dummy variable for high proportion of English learners
CASchools$HiEL = (CASchools$english >= 10) |> as.numeric()
fitE = feols(score ~ STR + HiEL + STR:HiEL, data = CASchools, vcov = "hc1")
fitE |> summary()
```

```
OLS estimation, Dep. Var.: score
Observations: 420
Standard-errors: Heteroskedasticity-robust
             Estimate Std. Error   t value  Pr(>|t|)
(Intercept) 682.245837  11.867815 57.487065 < 2.2e-16 ***
STR          -0.968460   0.589102 -1.643961   0.10094
HiEL          5.639135  19.514560  0.288971   0.77275
STR:HiEL     -1.276613   0.966920 -1.320289   0.18746
```

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 15.8   Adj. R2: 0.305368
```

The model output reveals that none of the individual t-tests reject the null hypothesis that the individual coefficients are zero.

However, these results are misleading because the true marginal effects are a mixture of these coefficients:

$$\frac{\partial E[\text{score}_i \mid \boldsymbol{X}_i]}{\partial \text{STR}_i} = \beta_2 + \beta_4 \cdot \text{HiEL}_i.$$

Therefore, to test if `STR` has an effect on `score`, we need to test the joint hypothesis:

$$H_0 : \beta_2 = 0 \quad \text{and} \quad \beta_4 = 0.$$

In terms of the multiple restriction notation $H_0 : \boldsymbol{R\beta} = \boldsymbol{0}$, we have

$$\boldsymbol{R} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Similarly, the marginal effects of `HiEL` is:

$$\frac{\partial E[\text{score}_i \mid \boldsymbol{X}_i]}{\partial \text{HiEL}_i} = \beta_3 + \beta_4 \cdot \text{STR}_i.$$

We test the joint hypothesis that $\beta_3 = 0$ and $\beta_4 = 0$:

$$\boldsymbol{R} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Wald Test

The **Wald test** is based on the **Wald distance**:

$$\boldsymbol{d} = \boldsymbol{R}\hat{\boldsymbol{\beta}},$$

which measures how far the estimated coefficients deviate from the hypothesized restrictions.

The covariance matrix of the Wald distance is: $Var(\boldsymbol{d}|\boldsymbol{X}) = \boldsymbol{R}Var(\hat{\boldsymbol{\beta}}|\boldsymbol{X})\boldsymbol{R}'$, which can be estimated as:

$$\widehat{Var}(\boldsymbol{d} \mid \boldsymbol{X}) = \boldsymbol{R}\widehat{\boldsymbol{V}}\boldsymbol{R}'.$$

The **Wald statistic** is the squared, variance-standardized distance:

$$W = \boldsymbol{d}'(\boldsymbol{R}\widehat{\boldsymbol{V}}\boldsymbol{R}')^{-1}\boldsymbol{d},$$

where $\widehat{\boldsymbol{V}}$ is a consistent estimator of the covariance matrix of $\hat{\boldsymbol{\beta}}$ (e.g., HC1 robust: $\widehat{\boldsymbol{V}} = \widehat{\boldsymbol{V}}_{hc1}$).

Under the null hypothesis, and assuming (A1)–(A4), the Wald statistic has an asymptotic chi-squared distribution:

$$W \xrightarrow{d} \chi^2_q,$$

where q is the number of restrictions.

The null is rejected if $W > \chi^2_{1-\alpha,q}$.

F-test

The Wald test is an asymptotic size-α-test under (A1)–(A4). Even if normality and homoskedasticity hold true as well, the Wald test is still only asymptotically valid, i.e.:

$$\lim_{n\to\infty} P(\text{Wald test rejects } H_0 | H_0 \text{ true}) = \alpha.$$

The F-test is the small sample correction of the Wald test. It is based on the same distance as the Wald test, but it is scaled by the number of restrictions q:

$$F = \frac{W}{q} = \frac{1}{q}(\boldsymbol{R}\hat{\boldsymbol{\beta}} - \boldsymbol{r})'(\boldsymbol{R}\widehat{\boldsymbol{V}}\boldsymbol{R}')^{-1}(\boldsymbol{R}\hat{\boldsymbol{\beta}} - \boldsymbol{r}).$$

Under the restrictive assumption that the Gaussian regression model holds, and if $\widehat{\boldsymbol{V}} = \widehat{\boldsymbol{V}}_{hom}$ is used, it can be shown that

$$F \sim F_{q;n-k}$$

for any finite sample size n. Here, $F_{q;n-k}$ is the F-distribution with q degrees of freedom in the numerator and $n-k$ degrees of freedom in the denominator.

The test decision for the **F-test**:

$$\begin{aligned} \text{do not reject } H_0 \quad & \text{if } F \leq F_{(1-\alpha,q,n-k)}, \\ \text{reject } H_0 \quad & \text{if } F > F_{(1-\alpha,q,n-k)}, \end{aligned}$$

where $F_{(p,m_1,m_2)}$ is the p-quantile of the F distribution with m_1 degrees of freedom in the numerator and m_2 degrees of freedom in the denominator.

F- and Chi-squared distribution

Similar to how the t-distribution t_{n-k} approaches the standard normal as sample size increases, we have $q \cdot F_{q;n-k} \to \chi^2_q$ as $n \to \infty$. Therefore, the F-test and Wald test become asymptotically equivalent and lead to identical statistical conclusions in large samples. For single constraint (q=1) hypotheses of the form $H_0 : \beta_j = 0$, the F-test is equivalent to a two-sided t-test.
The F-test can be viewed as a finite-sample correction of the Wald test. It tends to be more conservative than the Wald test in small samples, meaning that rejection by the F-test generally implies rejection by the Wald test, but not necessarily vice versa. Due to this more conservative nature, which helps control false rejections (Type I errors) in small samples, the F-test is often preferred in practice.

F-tests in R

The function `wald()` from the `fixest` package performs an F-test:

```
wald(fitE, keep = "STR")
```

```
Wald test, H0: joint nullity of STR and STR:HiEL
 stat = 5.6381, p-value = 0.003837, on 2 and 416 DoF, VCOV: Heteroskedasticity-robust.
```

```
wald(fitE, keep = "HiEL")
```

```
Wald test, H0: joint nullity of HiEL and STR:HiEL
 stat = 89.9, p-value < 2.2e-16, on 2 and 416 DoF, VCOV: Heteroskedasticity-robust.
```

The hypotheses that `STR` and `HiEL` have no effect on `score` can be clearly rejected.

Another research question is whether the effect of `STR` on `score` is zero only for the subgroup of schools with a high proportion of English learners (`HiEL` $= 1$). In this case, the marginal effect is:

$$\frac{\partial E[\text{score}_i \mid \boldsymbol{X}_i, \text{HiEL}_i = 1]}{\partial \text{STR}_i} = \beta_2 + \beta_4 \cdot 1,$$

and the null hypothesis is:

$$H_0 : \beta_2 + \beta_4 = 0.$$

The corresponding restriction matrix is:

$$\boldsymbol{R} = \begin{pmatrix} 0 & 1 & 0 & 1 \end{pmatrix},$$

where the number of restrictions is $q = 1$.

The function `linearHypothesis()` from the `AER` package is more flexible for these cases:

```
## Define hypothesis matrix:
R = matrix(c(0,1,0,1), ncol = 4)
linearHypothesis(fitE, hypothesis.matrix = R, test = "F", vcov. = vcovHC(fitE, type = "HC1"))
```

```
Linear hypothesis test:
STR  + STR:HiEL = 0

Model 1: restricted model
Model 2: score ~ STR + HiEL + STR:HiEL

Note: Coefficient covariance matrix supplied.

  Res.Df Df      F   Pr(>F)
1    417
2    416  1 8.5736 0.003598 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Similarly, this hypothesis can be rejected at the 0.01 level.

6.7 Jackknife Methods

Projection Matrix

Recall the vector of fitted values $\widehat{\boldsymbol{Y}} = \boldsymbol{X}\hat{\boldsymbol{\beta}}$. Inserting the model equation gives:

$$\widehat{\boldsymbol{Y}} = \boldsymbol{X}\hat{\boldsymbol{\beta}} = \underbrace{\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'}_{=\boldsymbol{P}}\boldsymbol{Y} = \boldsymbol{P}\boldsymbol{Y}.$$

The **projection matrix** $\boldsymbol{P}$ is also known as the *influence matrix* or *hat matrix* and maps observed values to fitted values.

Leverage Values

The diagonal entries of $\boldsymbol{P}$, given by

$$h_{ii} = \boldsymbol{X}_i'(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}_i,$$

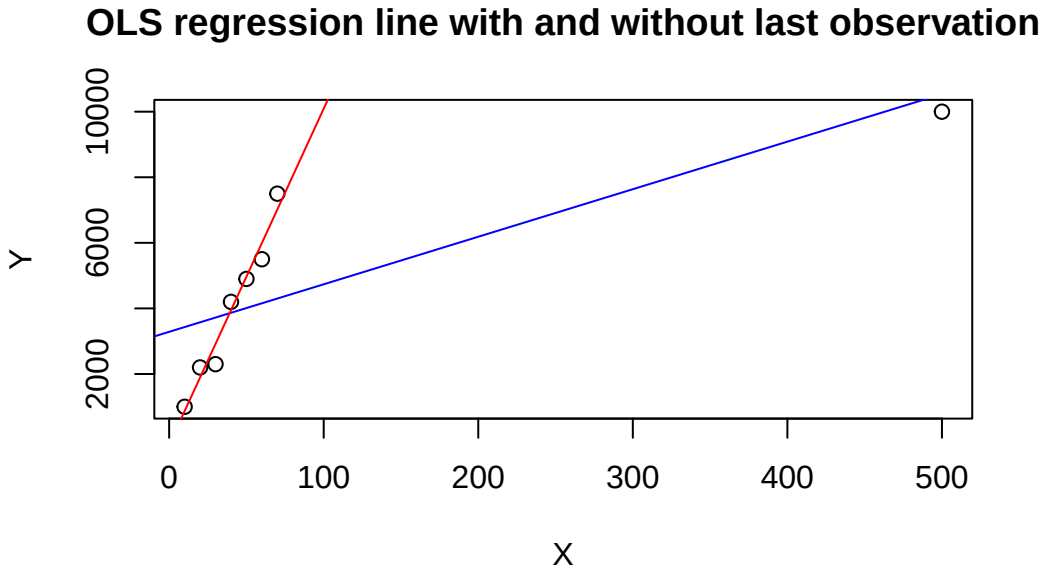
are called **leverage values** or *hat values* and measure how far away the regressor values of the i-th observation X_i are from those of the other observations.

Properties of leverage values:

$$0 \leq h_{ii} \leq 1, \quad \sum_{i=1}^{n} h_{ii} = k.$$

Leverage values h_{ii} indicate how much influence an observation $\boldsymbol{X}_i$ has on the regression fit, e.g., the last observation in the following artificial dataset:

```
X=c(10,20,30,40,50,60,70,500)
Y=c(1000,2200,2300,4200,4900,5500,7500,10000)
plot(X,Y, main="OLS regression line with and without last observation")
abline(lm(Y~X), col="blue")
abline(lm(Y[1:7]~X[1:7]), col="red")
```




```
hatvalues(lm(Y~X))
```

```
        1         2         3         4         5         6         7         8
0.1657356 0.1569566 0.1492418 0.1425911 0.1370045 0.1324820 0.1290237 0.9869646
```

A low leverage implies the presence of many regressor observations similar to $\boldsymbol{X}_i$ in the sample, while a high leverage indicates a lack of similar observations near $\boldsymbol{X}_i$.

An observation with a high leverage h_{ii} but a response value Y_i that is close to the true regression line $\boldsymbol{X}_i'\boldsymbol{\beta}$ (indicating a small error u_i) is considered a **good leverage point**. Despite being unusual in the regressor space, this point improves estimation precision because it provides valuable information about the regression relationship in regions where data is sparse.

Conversely, a **bad leverage point** occurs when both h_{ii} and the error u_i are large, indicating both unusual regressor and response values. This can misleadingly impact the regression fit.

The actual error term is unknown, but standardized residuals can be used to differentiate between good and bad leverage points.

Standardized Residuals

Many regression diagnostic tools rely on the residuals of the OLS estimation $\hat{u}_i$ because they provide insight into the properties of the unknown error terms u_i.

Under the homoskedastic linear regression model (A1)–(A5), the errors are independent and have the property

$$Var(u_i \mid \boldsymbol{X}) = \sigma^2.$$

Since $\boldsymbol{PX} = \boldsymbol{X}$ and, therefore,

$$\hat{\boldsymbol{u}} = (\boldsymbol{I}_n - \boldsymbol{P})\boldsymbol{Y} = (\boldsymbol{I}_n - \boldsymbol{P})(\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{u}) = (\boldsymbol{I}_n - \boldsymbol{P})\boldsymbol{u},$$

the residuals have a different property:

$$Var(\hat{\boldsymbol{u}} \mid \boldsymbol{X}) = \sigma^2(\boldsymbol{I}_n - \boldsymbol{P}).$$

The i-th residual satisfies

$$Var(\hat{u}_i \mid \boldsymbol{X}) = \sigma^2(1 - h_{ii}),$$

where h_{ii} is the i-th leverage value.

Under the assumption of homoskedasticity, the variance of $\hat{u}_i$ depends on $\boldsymbol{X}$, while the variance of u_i does not. Dividing by $\sqrt{1 - h_{ii}}$ removes the dependency:

$$Var\left(\frac{\hat{u}_i}{\sqrt{1 - h_{ii}}} \,\middle|\, \boldsymbol{X} \right) = \sigma^2$$

The **standardized residuals** are defined as follows:

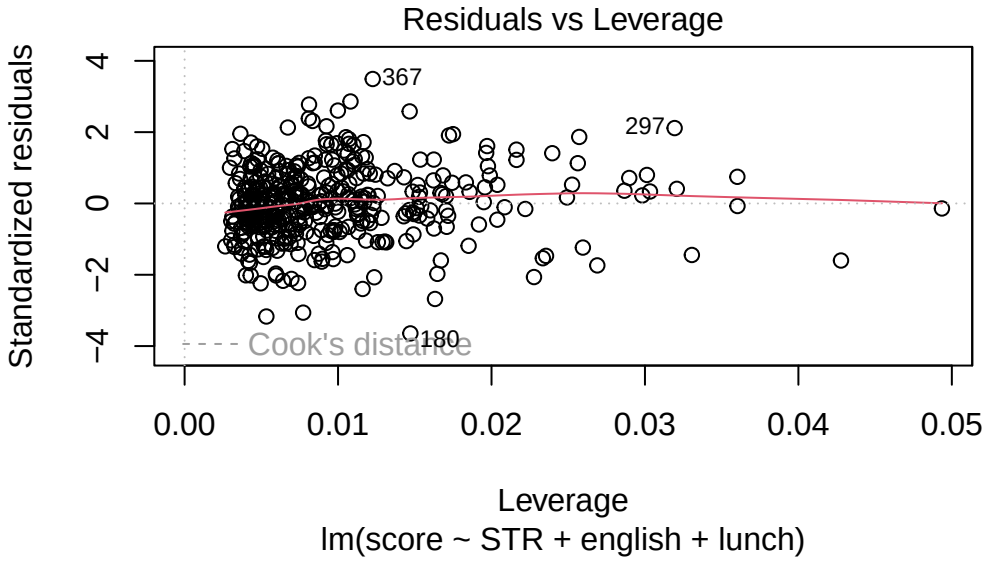
$$r_i := \frac{\hat{u}_i}{\sqrt{s_{\hat{u}}^2(1 - h_{ii})}}.$$

Standardized residuals are available using the R command `rstandard()`.

Residuals vs. Leverage Plot

Plotting standardized residuals against leverage values provides a graphical tool for detecting outliers. High leverage points have a strong influence on the regression fit. High leverage values with standardized residuals close to 0 are good leverage points, and high leverage values with large standardized residuals are bad leverage points.

```
fit = lm(score ~ STR + english + lunch, data = CASchools)
plot(fit, which = 5)
```

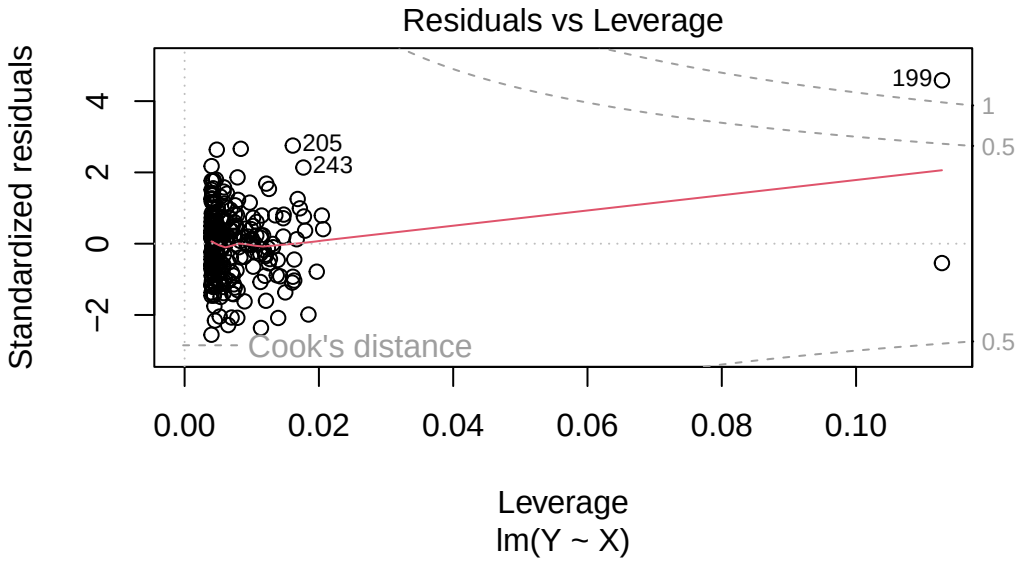


The plot indicates that some observations have a higher leverage value than others, but none of these have a large standardized residual, so they are not bad leverage points.

Here is an example with two high leverage points. Observation $i = 200$ is a good leverage point and $i = 199$ is a bad leverage point:

```
## simulate regressors and errors
X = rnorm(250)
u = rnorm(250)
## set some unusual observations manually
X[199] = 6
X[200] = 6
u[199] = 5
u[200] = 0
## define dependent variable
Y = X + u
```

```r
## residuals vs leverage plot
plot(lm(Y ~ X), which = 5)
```



The plot also shows Cook's distance thresholds. Cook's distance for observation i is defined as

$$D_i = \frac{(\hat{\boldsymbol{\beta}}_{(-i)} - \hat{\boldsymbol{\beta}})' \boldsymbol{X}' \boldsymbol{X} (\hat{\boldsymbol{\beta}}_{(-i)} - \hat{\boldsymbol{\beta}})}{k s^2_{\widehat{u}}},$$

where

$$\hat{\boldsymbol{\beta}}_{(-i)} - \hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1} \boldsymbol{X}_i \frac{\hat{u}_i}{1 - h_{ii}}.$$

Here, $\hat{\boldsymbol{\beta}}_{(-i)}$ is the i-th leave-one-out estimator (the OLS estimator when the i-th observation is left out).

This principle is called **Jackknife** because it is similar to the way a jackknife is used to cut something. The idea is to "cut" the data by removing one observation at a time and then re-estimating the model. The impact of cutting the i-th observation is proportional to $\hat{u}_i/(1 - h_{ii})$.

We should pay special attention to points outside Cook's distance thresholds of 0.5 and 1 and check for measurement errors or other anomalies.

Jackknife Standard Errors

Recall the heteroskedasticity-robust White estimator for the meat matrix $\boldsymbol{\Omega} = E[u_i^2 \boldsymbol{X}_i \boldsymbol{X}_i']$ in the sandwich formula tor the OLS variance:

$$\widehat{\boldsymbol{\Omega}} = \frac{1}{n} \sum_{i=1}^{n} \hat{u}_i^2 \boldsymbol{X}_i \boldsymbol{X}_i'.$$

If there are leverage points in the data, their presence might have a large influence on the estimation of $\boldsymbol{\Omega}$.

An alternative way of estimating the covariance matrix is to weight the observations by the leverage values:

$$\widehat{\boldsymbol{\Omega}}_{\text{jack}} = \frac{1}{n} \sum_{i=1}^{n} \frac{\hat{u}_i^2}{(1 - h_{ii})^2} \boldsymbol{X}_i \boldsymbol{X}_i'.$$

Observations with high leverage values have a small denominator $(1 - h_{ii})^2$ and are therefore downweighted, which makes this estimator more robust to the influence of leverage points.

The full jackknife covariance matrix estimator is conventionally labeled as the **HC3** estimator:

$$\widehat{\boldsymbol{V}}_{\text{jack}} = \widehat{\boldsymbol{V}}_{\text{hc3}} = (\boldsymbol{X}'\boldsymbol{X})^{-1} \widehat{\boldsymbol{\Omega}}_{\text{jack}} (\boldsymbol{X}'\boldsymbol{X})^{-1}.$$

There is also the HC2 estimator, which uses $\hat{u}_i^2(1 - h_{ii})$ instead of $\hat{u}_i^2/(1 - h_{ii})^2$, but this is less common.

The HC3 standard errors are:

$$se_{hc3}(\hat{\beta}_j) = \sqrt{[\widehat{\boldsymbol{V}}_{hc3}]_{jj}}.$$

If you have a small sample size and you are worried about influential observations, you should use the HC3 standard errors instead of the HC1 standard errors.

To display the HC3 standard errors in the regression table, you can use `modelsummary(fit, vcov = "HC3")`.

6.8 Cluster-robust Inference

Recall that in many economic applications, observations are naturally clustered. For instance, students within the same school, workers in the same firm, or households in the same village may share common unobserved factors that induce correlation in their outcomes.

As discussed in Section 5, for clustered observations we can use the notation $(\boldsymbol{X}_{ig}, Y_{ig})$, where the linear regression equation is:

$$Y_{ig} = \boldsymbol{X}_{ig}'\boldsymbol{\beta} + u_{ig}, \quad i = 1, \dots, n_g, \quad g = 1, \dots, G.$$

Under independence across clusters but allowing for arbitrary correlation within clusters, the OLS estimator remains unbiased, but its standard variance formula is no longer valid. As we saw in Section 5, the conditional variance

$$Var(\hat{\boldsymbol{\beta}} \mid \boldsymbol{X}) = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{D}\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}$$

satisfies

$$\boldsymbol{X}'\boldsymbol{D}\boldsymbol{X} = \sum_{g=1}^{G} E\Big[\Big(\sum_{i=1}^{n_g} \boldsymbol{X}_{ig} u_{ig}\Big)\Big(\sum_{i=1}^{n_g} \boldsymbol{X}_{ig} u_{ig}\Big)' \Big| \boldsymbol{X}\Big].$$

Cluster-robust Standard Errors

When observations within clusters are correlated, using ordinary standard errors (even heteroskedasticity-robust ones) will typically underestimate the true sampling variability of the OLS estimator.

To account for within-cluster correlation, we use **cluster-robust standard errors**. The key insight is to estimate the middle part of the sandwich formula above by allowing for arbitrary within-cluster correlation, while maintaining the independence assumption across clusters.

The **cluster-robust variance estimator** is:

$$\widehat{\boldsymbol{V}}_{CR0} = (\boldsymbol{X}'\boldsymbol{X})^{-1} \sum_{g=1}^{G} \Big(\sum_{i=1}^{n_g} \boldsymbol{X}_{ig} \hat{u}_{ig}\Big)\Big(\sum_{i=1}^{n_g} \boldsymbol{X}_{ig} \hat{u}_{ig}\Big)' (\boldsymbol{X}'\boldsymbol{X})^{-1}.$$

This estimator, also known as the **clustered sandwich estimator**, allows for arbitrary correlation of errors within clusters, including both heteroskedasticity and serial correlation. Like the HC estimators, it is consistent under large-sample asymptotics.

Finite Sample Correction

Similar to the HC1 correction for heteroskedasticity, a small-sample correction for the cluster-robust estimator is commonly applied:

$$\widehat{\boldsymbol{V}}_{CR1} = \frac{G}{G-1} \cdot \frac{n-1}{n-k} \cdot \widehat{\boldsymbol{V}}_{CR0},$$

where G is the number of clusters, n is the total sample size, and k is the number of regressors.

The corresponding cluster-robust standard errors are:

$$se_{CR1}(\hat{\beta}_j) = \sqrt{[\widehat{\boldsymbol{V}}_{CR1}]_{jj}}.$$

When to Cluster

You should use cluster-robust standard errors when:

1. There's a clear grouping structure in your data (schools, villages, firms, etc.)
2. You expect errors to be correlated within these groups
3. You have a sufficient number of clusters (generally at least 30-50)

Common examples include: - Student-level data clustered by school or classroom - Firm-level data clustered by industry - Individual-level data clustered by geographic region - Panel data clustered by individual or time period

Implementation in R

The `CASchools` dataset contains information on 420 California Schools from 45 different counties, which can be viewed as clusters.

The `fixest` package makes it easy to implement cluster-robust standard errors:

```
feols(score ~ STR + english, data = CASchools, cluster = "county") |> summary()
```

```
OLS estimation, Dep. Var.: score
Observations: 420
Standard-errors: Clustered (county)
             Estimate Std. Error   t value  Pr(>|t|)
(Intercept) 686.032245  15.802838  43.41196 < 2.2e-16 ***
STR          -1.101296   0.754387  -1.45986   0.15143
english      -0.649777   0.030230 -21.49427 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 14.4   Adj. R2: 0.423681
```

After accounting for clustering, the coefficient on `STR` is no longer statistically significant.

You can also use the `modelsummary()` function to compare the same regression with different standard errors:

```
fit1 = feols(score ~ STR + english, data = CASchools)
## List of standard errors:
myvcov = list("IID", "HC1", "HC3", ~county)
modelsummary(fit1, stars = TRUE, statistic = "conf.int", vcov = myvcov)
```

	(1)	(2)	(3)	(4)
(Intercept)	686.032***	686.032***	686.032***	686.032***
	[671.464, 700.600]	[668.875, 703.189]	[668.710, 703.354]	[654.969, 717.095]
STR	−1.101**	−1.101*	−1.101*	−1.101
	[−1.849, −0.354]	[−1.952, −0.250]	[−1.960, −0.242]	[−2.584, 0.382]
english	−0.650***	−0.650***	−0.650***	−0.650***
	[−0.727, −0.572]	[−0.711, −0.589]	[−0.711, −0.588]	[−0.709, −0.590]
Num.Obs.	420	420	420	420
R2	0.426	0.426	0.426	0.426
R2 Adj.	0.424	0.424	0.424	0.424
AIC	3439.1	3439.1	3439.1	3439.1
BIC	3451.2	3451.2	3451.2	3451.2
RMSE	14.41	14.41	14.41	14.41
Std.Errors	IID	HC1	HC3	by: county

+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001

Challenges with Cluster-robust Inference

The cluster-robust variance estimator relies on having a large number of clusters. With few clusters (generally $G < 30$), the estimator may be biased downward, leading to confidence intervals that are too narrow and overly frequent rejection of null hypotheses.

To account for high leverage points, the CR3 correction is similar to HC3 and applies a leverage adjustment at the cluster level:

$$\widehat{\boldsymbol{V}}_{CR3} = (\boldsymbol{X}'\boldsymbol{X})^{-1} \sum_{g=1}^{G} \Big(\sum_{i=1}^{n_g} \boldsymbol{X}_{ig} \frac{\hat{u}_{ig}}{1-h_{ig}} \Big) \Big(\sum_{i=1}^{n_g} \boldsymbol{X}_{ig} \frac{\hat{u}_{ig}}{1-h_{ig}} \Big)' (\boldsymbol{X}'\boldsymbol{X})^{-1}.$$

6.9 R-codes

metrics-sec06.R

Part III

Panel Data Methods

7 Fixed Effects

```
library(fixest)
library(modelsummary)
library(AER)
```

7.1 Panel Data

In panel data, we observe multiple individuals or entities over multiple time periods. Each observation is indexed by both individual $i = 1, \dots, n$ and time period $t = 1, \dots, T$. We denote a variable Y for individual i at time period t as Y_{it}.

Unlike cross-sectional data (which observes multiple individuals at a single point) or time series data (which tracks a single individual over time), panel data combines both dimensions.

Economic applications include:

- Growth: GDP and productivity across countries over time
- Corporate finance: Firm investment and capital structure dynamics
- Labor economics: Individual wage trajectories and employment patterns
- International trade: Bilateral trade flows between country pairs over years

In the case of multiple regressor variables, we denote the j-th regressor for individual i at time period t as $X_{j,it}$, where $j = 1, \dots, k$.

If each individual has observations for all time periods, we call this a **balanced panel**. The total number of observations is nT.

In typical economic panel datasets, we often have $n > T$ (more individuals than time points) or $n \approx T$ (roughly the same number of individuals as time points).

When some observations are missing for at least one individual or time period, we have an **unbalanced panel**.

7.2 Pooled Regression

Model Setup

The simplest approach to panel data is the **pooled regression**, which treats all observations as if they came from a single cross-section.

Consider a panel dataset with dependent variable Y_{it} and k independent variables $X_{1,it}, \dots, X_{k,it}$ for $i = 1, \dots, n$ and $t = 1, \dots, T$.

The first regressor variable represents an intercept (i.e., $X_{1,it} = 1$). We stack the regressor variables into the $k \times 1$ vector:

$$\boldsymbol{X}_{it} = \begin{pmatrix} 1 \\ X_{2,it} \\ \vdots \\ X_{k,it} \end{pmatrix}.$$

Pooled Panel Regression Model

The pooled linear panel regression model equation for individual $i = 1, \dots, n$ and time $t = 1, \dots, T$ is:

$$Y_{it} = \boldsymbol{X}_{it}'\boldsymbol{\beta} + u_{it},$$

where $\boldsymbol{\beta} = (\beta_1, \dots, \beta_k)'$ is the $k \times 1$ vector of **regression coefficients** and u_{it} is the **error term** for individual i at time t.

It is not reasonable to assume that Y_{it} and Y_{jt} are independent. Therefore, the random sampling assumption (A2) needs to be adapted to the cluster level. Instead of (A2), we assume that

$$(Y_{i1}, \dots, Y_{iT}, \boldsymbol{X}_{i1}', \dots, \boldsymbol{X}_{iT}')$$

are i.i.d. draws from their joint population distribution for $i = 1, \dots, n$.

This implies that observations across different individuals are independent. However, observations within an individual across time points may be dependent.

Therefore, to conduct inference about the population, we require n to be large, while T can be small or large.

Furthermore, while $\boldsymbol{X}_{is}$ and $\boldsymbol{X}_{it}$ can now be correlated, we require that the regressors are strictly exogenous, meaning $E[u_{it}|\boldsymbol{X}] = 0$. Therefore, assumption (A1) must be replaced by:

$$E[u_{it}|\boldsymbol{X}_{i1}, \dots, \boldsymbol{X}_{iT}] = 0.$$

Pooled OLS

The pooled OLS estimator is:

$$\hat{\boldsymbol{\beta}}_{\text{pool}} = \left(\sum_{i=1}^{n} \sum_{t=1}^{T} \boldsymbol{X}_{it} \boldsymbol{X}_{it}' \right)^{-1} \left(\sum_{i=1}^{n} \sum_{t=1}^{T} \boldsymbol{X}_{it} Y_{it} \right).$$

This can be written in matrix notation, where we define the pooled regressor matrix $\boldsymbol{X}$ of order $nT \times k$ and the dependent variable vector $\boldsymbol{Y}$ of order $nT \times 1$:

$$\hat{\boldsymbol{\beta}}_{\text{pool}} = (\boldsymbol{X}'\boldsymbol{X})^{-1} \boldsymbol{X}'\boldsymbol{Y}.$$

Pooled OLS is unbiased and consistent under the following assumptions:

Pooled OLS Assumptions

- **(A1-pool)** $E[u_{it} | \boldsymbol{X}_{i1}, \dots, \boldsymbol{X}_{iT}] = 0$
- **(A2-pool)** $\{(Y_{i1}, \dots, Y_{iT}, \boldsymbol{X}_{i1}', \dots, \boldsymbol{X}_{iT}')\}_{i=1}^{n}$ is an i.i.d. sample
- **(A3-pool)** $kur(Y_{it}) < \infty$ and $kur(X_{j,it}) < \infty$
- **(A4-pool)** $\sum_{i=1}^{n} \sum_{t=1}^{T} \boldsymbol{X}_{it} \boldsymbol{X}_{it}'$ is invertible

Under these assumptions, the asymptotic distribution of the pooled OLS estimator is:

$$\sqrt{n}(\hat{\boldsymbol{\beta}}_{\text{pool}} - \boldsymbol{\beta}) \xrightarrow{d} N(0, \boldsymbol{Q}^{-1} \boldsymbol{\Omega} \boldsymbol{Q}^{-1}), \qquad \text{as } n \to \infty,$$

where $\boldsymbol{Q} = E(\frac{1}{T} \sum_{t=1}^{T} \boldsymbol{X}_{it} \boldsymbol{X}_{it}')$ and $\boldsymbol{\Omega} = E((\frac{1}{T} \sum_{t=1}^{T} \boldsymbol{X}_{it} u_{it})(\frac{1}{T} \sum_{t=1}^{T} \boldsymbol{X}_{it} u_{it})')$.

To illustrate, consider the `Grunfeld` dataset, which provides investment, capital stock, and firm value data for 10 firms over 20 years:

```
data(Grunfeld, package = "AER")
head(Grunfeld)
```

```
  invest  value capital           firm year
1  317.6 3078.5     2.8 General Motors 1935
2  391.8 4661.7    52.6 General Motors 1936
3  410.6 5387.1   156.9 General Motors 1937
4  257.7 2792.2   209.2 General Motors 1938
5  330.8 4313.2   203.4 General Motors 1939
6  461.2 4643.9   207.2 General Motors 1940
```

```r
fit_pool = lm(invest ~ capital, data = Grunfeld)
fit_pool
```

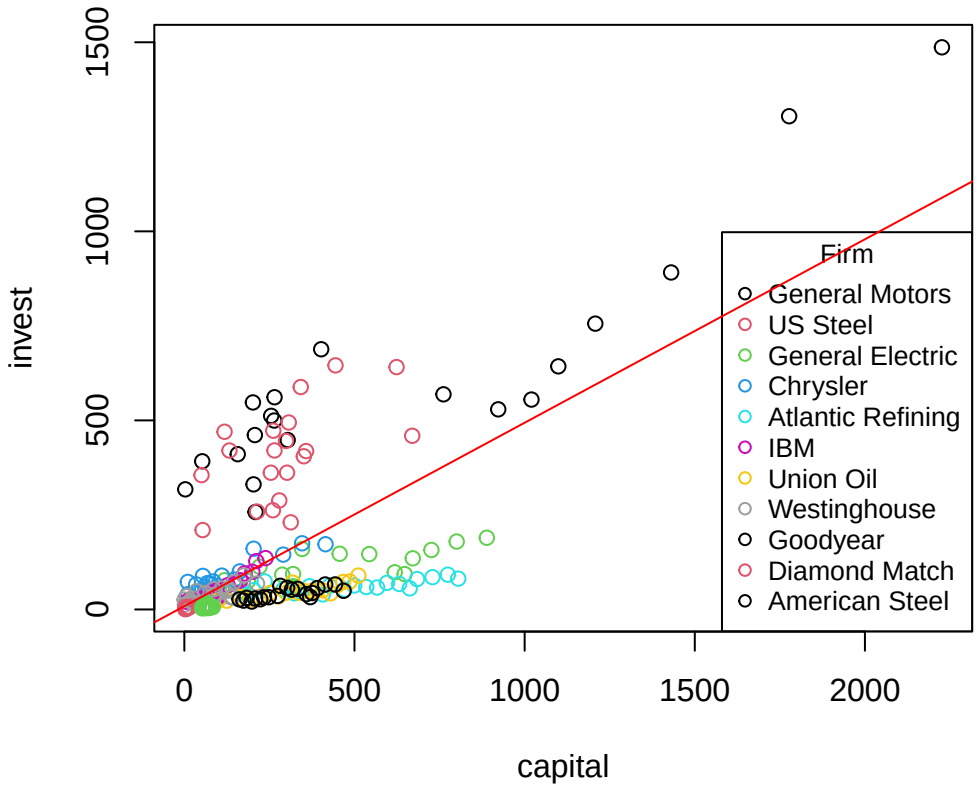
```
Call:
lm(formula = invest ~ capital, data = Grunfeld)

Coefficients:
(Intercept)      capital
     8.5651       0.4852
```

Cluster-Robust Inference

Let's visualize the data:

```r
plot(invest ~ capital, col = as.factor(firm), data = Grunfeld)
legend("bottomright", legend = unique(Grunfeld$firm), col = 1:10, pch = 1,
       title = "Firm", cex = 0.8)
abline(fit_pool, col = "red")
```



The observations appear in clusters, with each firm forming a cluster. This suggests potential problems with the pooled approach if we use classical standard errors.

The error covariance matrix for panel data has a block-diagonal structure:

$$\boldsymbol{D} = \text{Var}[\boldsymbol{u}|\boldsymbol{X}] = \begin{pmatrix} \boldsymbol{D}_1 & \boldsymbol{0} & \dots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{D}_2 & \dots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{0} & \dots & \boldsymbol{D}_n \end{pmatrix}$$

where $\boldsymbol{D}_i$ is the $T \times T$ covariance matrix for individual i:

$$\boldsymbol{D}_i = \begin{pmatrix} E[u_{i,1}^2|\boldsymbol{X}] & E[u_{i,1}u_{i,2}|\boldsymbol{X}] & \dots & E[u_{i,1}u_{i,T}|\boldsymbol{X}] \\ E[u_{i,2}u_{i,1}|\boldsymbol{X}] & E[u_{i,2}^2|\boldsymbol{X}] & \dots & E[u_{i,2}u_{i,T}|\boldsymbol{X}] \\ \vdots & \vdots & \ddots & \vdots \\ E[u_{i,T}u_{i,1}|\boldsymbol{X}] & E[u_{i,T}u_{i,2}|\boldsymbol{X}] & \dots & E[u_{i,T}^2|\boldsymbol{X}] \end{pmatrix}$$

The variance of the pooled OLS estimator is:

$$\text{Var}[\hat{\boldsymbol{\beta}}_{\text{pool}}|\boldsymbol{X}] = (\boldsymbol{X}'\boldsymbol{X})^{-1}(\boldsymbol{X}'\boldsymbol{D}\boldsymbol{X})(\boldsymbol{X}'\boldsymbol{X})^{-1}$$

The cluster-robust covariance matrix estimator is:

$$\widehat{\boldsymbol{V}}_{\text{pool}} = (\boldsymbol{X}'\boldsymbol{X})^{-1} \sum_{i=1}^{n} \left(\sum_{t=1}^{T} \boldsymbol{X}_{it}\hat{u}_{it} \right) \left(\sum_{t=1}^{T} \boldsymbol{X}_{it}\hat{u}_{it} \right)' (\boldsymbol{X}'\boldsymbol{X})^{-1}$$

We can implement this using the `fixest` package:

```
# Pooled regression with fixest
fit_pool_fe = feols(invest ~ capital, data = Grunfeld)

# Incorrect Classical Standard Errors
summary(fit_pool_fe)
```

```
OLS estimation, Dep. Var.: invest
Observations: 220
Standard-errors: IID
             Estimate Std. Error   t value  Pr(>|t|)
(Intercept) 8.565056  13.967368  0.613219   0.54037
capital     0.485191   0.035861 13.529645 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 154.9   Adj. R2: 0.453935
```

```r
# Cluster-robust standard errors (clustered by firm)
summary(fit_pool_fe, cluster = "firm")
```

```
OLS estimation, Dep. Var.: invest
Observations: 220
Standard-errors: Clustered (firm)
            Estimate Std. Error  t value  Pr(>|t|)
(Intercept) 8.565056  25.729726 0.332886 0.7460942
capital     0.485191   0.132374 3.665310 0.0043507 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 154.9   Adj. R2: 0.453935
```

7.3 Time-invariant Regressors

Consider a simple panel regression model:

$$Y_{it} = \beta_1 + \beta_2 X_{it} + \beta_3 Z_i + u_{it} \tag{7.1}$$

Here, Z_i represents a time-invariant variable specific to individual i (e.g., gender, ethnicity, birthplace).

With the usual exogeneity condition $E[u_{it}|X_{it}, Z_i]$, the coefficient β_2 can be interpreted as the marginal effect of X_{it} on Y_{it}, holding Z_i constant.

The key advantage of panel data is that we can control for a time-invariant variable Z_i even if it is unobserved.

To see this, consider data from just two time periods, $t = 1$ and $t = 2$. Taking the difference between time periods:

$$\begin{aligned} Y_{i2} - Y_{i1} &= (\beta_1 + \beta_2 X_{i2} + \beta_3 Z_i + u_{i2}) - (\beta_1 + \beta_2 X_{i1} + \beta_3 Z_i + u_{i1}) \\ &= \beta_2 (X_{i2} - X_{i1}) + (u_{i2} - u_{i1}) \end{aligned}$$

This **first-differencing** transformation eliminates both the intercept β_1 and the effect of the time-invariant variable $\beta_3 Z_i$.

The coefficient β_2 is simply the regression coefficient from the first-differenced model:

$$\Delta Y_i = \beta_2 \Delta X_i + \Delta u_i,$$

where $\Delta Y_i = Y_{i2} - Y_{i1}$, $\Delta X_i = X_{i2} - X_{i1}$, and $\Delta u_i = u_{i2} - u_{i1}$.

Therefore, β_2 can be estimated from a regression of ΔY_i on ΔX_i without intercept. We do not need to observe Z_i to estimate β_2 from model Equation 7.1.

We can combine the terms β_1 and $\beta_3 Z_i$ into a single **individual fixed effect** $\alpha_i = \beta_1 + \beta_3 Z_i$. This term represents all unobserved, time-constant factors that affect the dependent variable.

7.4 The Fixed Effects Model

Let's formalize the fixed effects model. Consider a panel dataset with dependent variable Y_{it}, a vector of k independent variables $\boldsymbol{X}_{it}$, and an unobserved individual fixed effect α_i for $i = 1, \dots, n$ and $t = 1, \dots, T$.

Fixed Effects Regression Model

The fixed effects regression model for individual $i = 1, \dots, n$ and time $t = 1, \dots, T$ is:

$$Y_{it} = \alpha_i + \boldsymbol{X}_{it}'\boldsymbol{\beta} + u_{it} \tag{7.2}$$

where $\boldsymbol{\beta} = (\beta_1, \dots, \beta_k)'$ is the $k \times 1$ vector of regression coefficients, α_i is the individual fixed effect, and u_{it} is the error term.

Identification Assumptions

To identify β_j as the ceteris paribus marginal effect of $X_{j,it}$ on Y_{it}, holding constant the fixed effect α_i and the other regressors, we need to make some assumptions.

1. **Strict exogeneity conditional on fixed effects**: $E[u_{it}|\boldsymbol{X}_{i1}, \dots, \boldsymbol{X}_{iT}, \alpha_i] = 0$ for all t. This means that the error u_{it} is uncorrelated with the regressors in all time periods, conditional on the fixed effect.
2. **Time-varying regressors**: There must be variation in $\boldsymbol{X}_{j,it}$ over time within each individual. Time-invariant regressors are absorbed by the fixed effect α_i and cannot be separately identified.

If strict exogeneity is violated (e.g., due to feedback effects where Y_{it} affects future values of $\boldsymbol{X}_{is}$ for $s > t$), then the fixed effects estimator will be inconsistent. In this case, dynamic panel data models may be appropriate.

First-Differencing Estimator

As shown earlier, we can eliminate the fixed effects by taking first differences. Using $\Delta Y_{it} = Y_{it} - Y_{i,t-1}$ as the dependent variable and inserting model Equation 7.2, we get:

$$\Delta Y_{it} = (\Delta \boldsymbol{X}_{it})'\boldsymbol{\beta} + \Delta u_{it} \tag{7.3}$$

where $\Delta \boldsymbol{X}_{it} = \boldsymbol{X}_{it} - \boldsymbol{X}_{i,t-1}$ and $\Delta u_{it} = u_{it} - u_{i,t-1}$.

We can then apply OLS to this transformed model:

```
# Create first differences manually for demonstration
diffcapital = c(aggregate(Grunfeld$capital, by = list(Grunfeld$firm), FUN = diff)$x)
diffinvest = c(aggregate(Grunfeld$inv, by = list(Grunfeld$firm), FUN = diff)$x)

# First-difference regression
lm(diffinvest ~ diffcapital - 1)
```

```
Call:
lm(formula = diffinvest ~ diffcapital - 1)

Coefficients:
diffcapital
     0.2307
```

A problem with this differenced estimator is that the transformed error term Δu_{it} defines an artificial correlation structure, which makes the estimator non-optimal. $\Delta u_{i,t+1} = u_{i,t+1} - u_{i,t}$ is correlated with $\Delta u_{i,t} = u_{i,t} - u_{i,t-1}$ through $u_{i,t}$.

Within Estimator

An efficient estimator can be obtained by a different transformation. The idea is to consider the individual specific means

$$\overline{Y}_{i\cdot} = \frac{1}{T}\sum_{t=1}^{T} Y_{it}, \quad \overline{\boldsymbol{X}}_{i\cdot} = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{X}_{it}, \quad \overline{u}_{i\cdot} = \frac{1}{T}\sum_{t=1}^{T} u_{it}$$

Taking the means over t of both sides of Equation 7.2 implies

$$\overline{Y}_{i\cdot} = \alpha_i + \overline{\boldsymbol{X}}_{i\cdot}'\boldsymbol{\beta} + \overline{u}_{i\cdot}. \tag{7.4}$$

Then, we subtract these means from the original equation:

$$Y_{it} - \overline{Y}_{i\cdot} = (\boldsymbol{X}_{it} - \overline{\boldsymbol{X}}_{i\cdot})'\boldsymbol{\beta} + (u_{it} - \overline{u}_{i\cdot})$$

The fixed effect α_i drops out.

The deviations from the individual specific means are called **within transformations**:

$$\dot{Y}_{it} = Y_{it} - \overline{Y}_{i\cdot}, \quad \dot{\boldsymbol{X}}_{it} = \boldsymbol{X}_{it} - \overline{\boldsymbol{X}}_{i\cdot}, \quad \dot{u}_{it} = u_{it} - \overline{u}_{i\cdot}$$

The within-transfromed model equation is

$$\dot{Y}_{it} = \dot{\boldsymbol{X}}_{it}'\boldsymbol{\beta} + \dot{u}_{it} \tag{7.5}$$

The **within estimator** (also called the **fixed effects estimator**) is:

$$\hat{\boldsymbol{\beta}}_{\text{fe}} = \left(\sum_{i=1}^{n} \sum_{t=1}^{T} \dot{\boldsymbol{X}}_{it} \dot{\boldsymbol{X}}_{it}' \right)^{-1} \left(\sum_{i=1}^{n} \sum_{t=1}^{T} \dot{\boldsymbol{X}}_{it} \dot{Y}_{it} \right)$$

```
# Fixed effects estimation using fixest
fit_fe = feols(invest ~ capital, fixef = "firm", data = Grunfeld)
fit_fe$coefficients
```

```
  capital
0.3707023
```

Fixed Effects Regression Assumptions

- (A1-fe) $E[u_{it}|\boldsymbol{X}_{i1}, \dots, \boldsymbol{X}_{iT}, \alpha_i] = 0$.
- (A2-fe) $(\alpha_i, Y_{i1}, \dots, Y_{iT}, \boldsymbol{X}_{i1}', \dots, \boldsymbol{X}_{iT}')_{i=1}^{n}$ is an i.i.d. sample.
- (A3-fe) $kur(Y_{it}) < \infty$, $kur(u_{it}) < \infty$.
- (A4-fe) $\sum_{i=1}^{n} \sum_{t=1}^{T} \dot{\boldsymbol{X}}_{it} \dot{\boldsymbol{X}}_{it}'$ is invertible.

(A1-fe) is the same as (A1-pool), but now we condition on the unobserved fixed effect α_i.

(A2-fe) is a standard random sampling assumption indicating that individuals $i = 1, \dots, n$ are randomly sampled.

(A3-fe) ensures finite fourth moments, which is a requirement for asymptotic normality of the estimator.

(A4-fe) is satisfied if there is no perfect multicollinearity and if no regressor is constant over time for any individual.

Under (A2-fe), the collection of the within-transformed variables of individual i,

$$(\dot{Y}_{i1}, \dots, \dot{Y}_{iT}, \dot{\boldsymbol{X}}_{i1}, \dots, \dot{\boldsymbol{X}}_{iT}, \dot{u}_{i1}, \dots, \dot{u}_{iT}),$$

forms an i.i.d. sequence for $i = 1, \dots, n$.

The within-transformed variables satisfy (A1-pool)–(A4-pool), which mean that its asymptotic distribution is:

$$\sqrt{n}(\hat{\boldsymbol{\beta}}_{\text{fe}} - \boldsymbol{\beta}) \xrightarrow{d} N(0, \boldsymbol{W}^{-1}\boldsymbol{\Psi}\boldsymbol{W}^{-1}), \qquad \text{as } n \to \infty,$$

where $\boldsymbol{W} = E(\frac{1}{T}\sum_{t=1}^{T} \dot{\boldsymbol{X}}_{it}\dot{\boldsymbol{X}}_{it}^{'})$ and $\boldsymbol{\Psi} = E((\frac{1}{T}\sum_{t=1}^{T} \dot{\boldsymbol{X}}_{it}\dot{u}_{it})(\frac{1}{T}\sum_{t=1}^{T} \dot{\boldsymbol{X}}_{it}\dot{u}_{it})')$.

Hence, we can apply the cluster-robust covariance matrix estimator of the pooled regression to the within-transformed variables:

```
# Inference with cluster-robust standard errors
summary(fit_fe, cluster = "firm")
```

```
OLS estimation, Dep. Var.: invest
Observations: 220
Fixed-effects: firm: 11
Standard-errors: Clustered (firm)
        Estimate Std. Error t value  Pr(>|t|)
capital 0.370702   0.064785 5.72203 0.0001924 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 58.9     Adj. R2: 0.91717
             Within R2: 0.659603
```

Dummy Variable Approach

An equivalent way to estimate the fixed effects model is to include a dummy variable for each individual. This approach is known as the **least squares dummy variable (LSDV)** estimator:

```
# Equivalent to fit_fe
fit_fe_lsdv = lm(invest ~ capital + factor(firm) - 1, data = Grunfeld)
fit_fe_lsdv$coefficients
```

```
                  capital    factor(firm)General Motors
                0.3707023                   367.6436372
      factor(firm)US Steel  factor(firm)General Electric
```

```
                301.1715657                    -46.0502428
factor(firm)Chrysler 41.1776965 factor(firm)Atlantic Refining -118.6424177
        factor(firm)IBM 16.7523079         factor(firm)Union Oil -69.1553441
factor(firm)Westinghouse 11.1445528          factor(firm)Goodyear -68.5432229
factor(firm)Diamond Match 0.8819721    factor(firm)American Steel -18.3676804
```

The coefficient on the regressor `capital` is the same as in the within estimator. However, the LSDV approach becomes computationally intensive with many individuals, and the standard errors need to be adjusted for clustering.

7.5 Time Fixed Effects

While individual fixed effects control for unobserved heterogeneity across individuals, we might also want to control for factors that vary over time but are constant across individuals (e.g., macroeconomic conditions, policy changes).

The **time fixed effects** model is:

$$Y_{it} = \lambda_t + \boldsymbol{X}_{it}'\boldsymbol{\beta} + u_{it} \tag{7.6}$$

where λ_t captures time-specific effects. Similar to individual fixed effects, we can rewrite this model by demeaning across time:

$$Y_{it} - \overline{Y}_{\cdot t} = (\boldsymbol{X}_{it} - \overline{\boldsymbol{X}}_{\cdot t})'\boldsymbol{\beta} + (u_{it} - \overline{u}_{\cdot t})$$

where the time-specific means are:

$$\overline{Y}_{\cdot t} = \frac{1}{n}\sum_{i=1}^{n} Y_{it}, \quad \overline{\boldsymbol{X}}_{\cdot t} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{X}_{it}, \quad \overline{u}_{\cdot t} = \frac{1}{n}\sum_{i=1}^{n} u_{it}.$$

Hence, we regress $Y_{it} - \overline{Y}_{\cdot t}$ on $\boldsymbol{X}_{it} - \overline{\boldsymbol{X}}_{\cdot t}$ to estimate $\boldsymbol{\beta}$ in Equation 7.6.

```
# Time fixed effects
fit_timefe = feols(invest ~ capital, fixef = "year", data = Grunfeld)
summary(fit_timefe, cluster = "firm")
```

```
OLS estimation, Dep. Var.: invest
Observations: 220
Fixed-effects: year: 20
Standard-errors: Clustered (firm)
        Estimate Std. Error t value  Pr(>|t|)
capital 0.539676   0.163321 3.30438 0.0079544 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 151.1     Adj. R2: 0.430515
              Within R2: 0.450115
```

7.6 Two-way Fixed Effects

We can combine both individual and time fixed effects in a **two-way fixed effects** model:

$$Y_{it} = \alpha_i + \lambda_t + \boldsymbol{X}'_{it}\boldsymbol{\beta} + u_{it} \tag{7.7}$$

This model controls for both individual-specific and time-specific unobserved factors. To estimate it, we apply a two-way transformation that subtracts individual means, time means, and adds back the overall mean:

$$\ddot{Y}_{it} = Y_{it} - \overline{Y}_{i\cdot} - \overline{Y}_{\cdot t} + \overline{Y}$$
$$\ddot{\boldsymbol{X}}_{it} = \boldsymbol{X}_{it} - \overline{\boldsymbol{X}}_{i\cdot} - \overline{\boldsymbol{X}}_{\cdot t} + \overline{\boldsymbol{X}}$$

To see why this is useful, consider the following transformations applied to the left-hand side of Equation 7.7:

- Individual specific mean:
$$\overline{Y}_{i\cdot} = \alpha_i + \overline{\lambda} + \overline{\boldsymbol{X}}'_{i\cdot}\boldsymbol{\beta} + \overline{u}_{i\cdot},$$
where $\overline{\lambda} = \frac{1}{T}\sum_{t=1}^{T}\lambda_t$.
- Time specific mean:
$$\overline{Y}_{\cdot t} = \overline{\alpha} + \lambda_t + \overline{\boldsymbol{X}}'_{\cdot t}\boldsymbol{\beta} + \overline{u}_{\cdot t},$$
where $\overline{\alpha} = \frac{1}{n}\sum_{i=1}^{n}\alpha_i$.
- Total mean:
$$\overline{Y} = \frac{1}{nT}\sum_{i=1}^{n}\sum_{t=1}^{T}Y_{it} = \overline{\alpha} + \overline{\lambda} + \overline{\boldsymbol{X}}'\boldsymbol{\beta} + \overline{u},$$
where $\overline{\boldsymbol{X}} = \frac{1}{nT}\sum_{i=1}^{n}\sum_{t=1}^{T}\boldsymbol{X}_{it}$ and $\overline{u} = \frac{1}{nT}\sum_{i=1}^{n}\sum_{t=1}^{T}u_{it}$.

The transformed model is:

$$\ddot{Y}_{it} = \ddot{\boldsymbol{X}}_{it}^{'}\boldsymbol{\beta} + \ddot{u}_{it} \tag{7.8}$$

where $\ddot{u}_{it} = u_{it} - \overline{u}_{i\cdot} - \overline{u}_{\cdot t} + \overline{u}$.

Hence, we estimate $\boldsymbol{\beta}$ by regressing $\ddot{Y}_{it}$ on $\ddot{\boldsymbol{X}}_{it}$.

```
# Two-way fixed effects
fit_2wayfe = feols(invest ~ capital, fixef = c("firm", "year"), data = Grunfeld)
summary(fit_2wayfe, cluster = "firm")
```

```
OLS estimation, Dep. Var.: invest
Observations: 220
Fixed-effects: firm: 11,  year: 20
Standard-errors: Clustered (firm)
        Estimate Std. Error t value   Pr(>|t|)
capital  0.40875   0.062522 6.53767 6.5744e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 54.7     Adj. R2: 0.921459
             Within R2: 0.60632
```

For inference, we use cluster-robust standard errors:

```
# Cluster-robust standard errors
summary(fit_2wayfe, cluster = "firm")
```

```
OLS estimation, Dep. Var.: invest
Observations: 220
Fixed-effects: firm: 11,  year: 20
Standard-errors: Clustered (firm)
        Estimate Std. Error t value   Pr(>|t|)
capital  0.40875   0.062522 6.53767 6.5744e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 54.7     Adj. R2: 0.921459
             Within R2: 0.60632
```

	OLS-IID	OLS-CL	FE	Time FE	Two-way FE
(Intercept)	8.565	8.565			
	(13.967)	(25.730)			
capital	0.485***	0.485**	0.371***	0.540**	0.409***
	(0.036)	(0.132)	(0.065)	(0.163)	(0.063)
Num.Obs.	220	220	220	220	220
R2	0.456	0.456	0.921	0.483	0.932
R2 Adj.	0.454	0.454	0.917	0.431	0.921
R2 Within			0.660	0.450	0.606
R2 Within Adj.			0.658	0.447	0.604
AIC	2847.2	2847.2	2441.9	2874.4	2447.2
BIC	2854.0	2854.0	2482.7	2945.6	2552.4
RMSE	154.91	154.91	58.93	151.14	54.70
Std.Errors	IID	by: firm	by: firm	by: firm	by: firm
FE: firm			X		X
FE: year				X	X

+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001

7.7 Comparison of Panel Models

Let's compare the different panel regression approaches:

```
# Create a list of models
models = list(
  "OLS-IID" = feols(invest ~ capital, data = Grunfeld),
  "OLS-CL" = feols(invest ~ capital, data = Grunfeld, cluster = "firm"),
  "FE" = feols(invest ~ capital, fixef = "firm", data = Grunfeld, cluster = "firm"),
  "Time FE" = feols(invest ~ capital, fixef = "year", data = Grunfeld, cluster = "firm"),
  "Two-way FE" = feols(invest ~ capital, fixef = c("firm", "year"), data = Grunfeld, cluster
)

# Generate the comparison table with clustered standard errors
modelsummary(models, stars = TRUE)
```

7.8 Panel R-squared

In panel data models with fixed effects, two different R-squared measures provide distinct information about model fit:

Within R-squared

The within R-squared measures the proportion of within-individual variation explained by the model. For the three different fixed effects specifications, the within R-squared is defined as follows:

- For individual fixed effects:

$$R^2_{wit} = 1 - \frac{\sum_{i=1}^{n}\sum_{t=1}^{T}(\dot{Y}_{it} - \dot{\boldsymbol{X}}_{it}^{'}\hat{\boldsymbol{\beta}})^2}{\sum_{i=1}^{n}\sum_{t=1}^{T}\dot{Y}_{it}^2}$$

- For time fixed effects:

$$R^2_{wit} = 1 - \frac{\sum_{i=1}^{n}\sum_{t=1}^{T}(Y_{it} - \overline{Y}_{\cdot t} - (\boldsymbol{X}_{it} - \overline{\boldsymbol{X}}_{\cdot t})'\hat{\boldsymbol{\beta}})^2}{\sum_{i=1}^{n}\sum_{t=1}^{T}(Y_{it} - \overline{Y}_{\cdot t})^2}$$

- For two-way fixed effects:

$$R^2_{wit} = 1 - \frac{\sum_{i=1}^{n}\sum_{t=1}^{T}(\ddot{Y}_{it} - \ddot{\boldsymbol{X}}_{it}^{'}\hat{\boldsymbol{\beta}})^2}{\sum_{i=1}^{n}\sum_{t=1}^{T}\ddot{Y}_{it}^2}$$

In the panel models for the Grunfeld data, the individual fixed effects model has the highest within R-squared (0.660), suggesting that within-firm variations in capital explain 66% of within-firm variations in investment.

This drops to 0.450 in the time fixed effects model, indicating that year-specific factors share substantial variation with capital stock within each year.

The higher within R-squared for individual fixed effects (0.660) compared to time fixed effects (0.450) suggests that firm-specific characteristics play a greater role in explaining variation in investment than year-specific factors.

The two-way fixed effects model shows an intermediate within R-squared (0.606). This model controls for more confounding factors from both dimensions, resulting in an estimate that is likely closer to the true causal effect of capital on investment, though with somewhat reduced statistical power.

Overall R-squared

The overall R-squared measures how well the complete model (including fixed effects) explains the total variation:

$$R^2_{ov} = 1 - \frac{\sum_{i=1}^{n}\sum_{t=1}^{T}(Y_{it} - \hat{Y}_{it})^2}{\sum_{i=1}^{n}\sum_{t=1}^{T}(Y_{it} - \overline{Y})^2}$$

Here, $\hat{Y}_{it}$ is the fitted value of the corresponding model.

The overall R-squared values reveal how different specifications explain investment variation: pooled OLS (45.6%), firm fixed effects (92.1%), time fixed effects (48.3%), and two-way fixed effects (93.2%). The large jump when adding firm fixed effects, compared to the minimal improvement from time fixed effects, confirms that firm-specific characteristics are far more important determinants of investment behavior than year-specific factors.

The within R-squared is typically more relevant because it isolates the relationship of interest after controlling for unobserved heterogeneity. However, if you're interested in overall predictive power, the overall R-squared provides that information.

Fitted Values

The overall R-squared requires the computation of the fitted values $\hat{Y}_{it}$. To compute them, we require some estimates or averages of the fixed effects themselves.

- For individual fixed effects:

$$\begin{aligned}\hat{Y}_{it} &= \hat{\alpha}_i + \boldsymbol{X}'_{it}\hat{\boldsymbol{\beta}} \\ \hat{\alpha}_i &= \overline{Y}_{i\cdot} - \overline{\boldsymbol{X}}'_{i\cdot}\hat{\boldsymbol{\beta}}\end{aligned}$$

- For time fixed effects:

$$\begin{aligned}\hat{Y}_{it} &= \hat{\lambda}_t + \boldsymbol{X}'_{it}\hat{\boldsymbol{\beta}} \\ \hat{\lambda}_t &= \overline{Y}_{\cdot t} - \overline{\boldsymbol{X}}'_{\cdot t}\hat{\boldsymbol{\beta}}\end{aligned}$$

- For two-way fixed effects:

$$\hat{Y}_{it} = \hat{\alpha}_i + \hat{\lambda}_t - \hat{\mu} + \boldsymbol{X}'_{it}\hat{\boldsymbol{\beta}},$$

  where

$$\begin{aligned}\hat{\alpha}_i &= \overline{Y}_{i\cdot} - \overline{\boldsymbol{X}}'_{i\cdot}\hat{\boldsymbol{\beta}} - \hat{\mu} \\ \hat{\lambda}_t &= \overline{Y}_{\cdot t} - \overline{\boldsymbol{X}}'_{\cdot t}\hat{\boldsymbol{\beta}} - \hat{\mu} \\ \hat{\mu} &= \overline{Y} - \overline{\boldsymbol{X}}'\hat{\boldsymbol{\beta}}\end{aligned}$$

While these fixed effects estimates are useful for calculating fitted values, they are not recommended for direct interpretation. Fixed effects capture all time-invariant (or unit-invariant) factors, observed and unobserved, making them a "black box" rather than specific causal parameters.

7.9 Application: Traffic Fatalities

To illustrate the importance of fixed effects in empirical work, let's examine how government policies affect traffic fatalities. We'll use the `Fatalities` dataset from the AER package, which contains panel data on traffic fatalities, drunk driving laws, and beer taxes for U.S. states from 1982 to 1988.

```
data(Fatalities, package = "AER")
# Create the fatality rate per 10,000 population
Fatalities$fatal_rate = Fatalities$fatal / Fatalities$pop * 10000
```

Cross-sectional Analysis

First, let's examine the relationship between beer taxes and traffic fatality rates using pooled OLS:

```
fatal_cs = feols(fatal_rate ~ beertax, data = Fatalities, cluster = "state")
summary(fatal_cs)
```

```
OLS estimation, Dep. Var.: fatal_rate
Observations: 336
Standard-errors: Clustered (state)
            Estimate Std. Error  t value  Pr(>|t|)
(Intercept) 1.853308   0.118519 15.63719 < 2.2e-16 ***
beertax     0.364605   0.119686  3.04636 0.0037916 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.542116   Adj. R2: 0.090648
```

Surprisingly, we find a positive relationship between beer taxes and fatality rates. This counterintuitive result likely stems from omitted variable bias.

Fixed Effects Approach

Now, let's use the panel structure to control for unobserved state-specific factors:

```
# State fixed effects model
fatal_fe = feols(fatal_rate ~ beertax, fixef = "state", data = Fatalities, cluster = "state")
summary(fatal_fe)
```

```
OLS estimation, Dep. Var.: fatal_rate
Observations: 336
Fixed-effects: state: 48
Standard-errors: Clustered (state)
         Estimate Std. Error  t value Pr(>|t|)
beertax -0.655874   0.291856 -2.24725 0.029358 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.17547     Adj. R2: 0.889129
                Within R2: 0.040745
```

With state fixed effects, the coefficient becomes negative, aligning with our theoretical expectation that higher beer taxes should reduce drunk driving and fatalities.

Let's add time fixed effects

```
# State fixed effects model
fatal_twoway = feols(fatal_rate ~ beertax, fixef = c("state", "year"), data = Fatalities, cl
summary(fatal_twoway)
```

```
OLS estimation, Dep. Var.: fatal_rate
Observations: 336
Fixed-effects: state: 48,  year: 7
Standard-errors: Clustered (state)
        Estimate Std. Error  t value Pr(>|t|)
beertax -0.63998   0.357078 -1.79227 0.079528 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.171819     Adj. R2: 0.891425
                 Within R2: 0.036065
```

Finally, let's add control variables that are neither constant over time nor across states:

```
Fatalities$punish = ifelse(Fatalities$jail == "yes" | Fatalities$service == "yes",
                           "yes", "no")
fatal_full = feols(fatal_rate ~ beertax + drinkage + punish + miles + unemp + log(income), fi
                   data = Fatalities, cluster = "state")
```

```
NOTE: 1 observation removed because of NA values (RHS: 1).
```

```
summary(fatal_full)
```

```
OLS estimation, Dep. Var.: fatal_rate
Observations: 335
Fixed-effects: state: 48,  year: 7
Standard-errors: Clustered (state)
               Estimate Std. Error   t value   Pr(>|t|)
beertax     -0.45646674 0.30680756 -1.487795 0.14348400
drinkage    -0.00215674 0.02151945 -0.100223 0.92059358
punishyes    0.03898148 0.10316089  0.377871 0.70722783
miles        0.00000898 0.00000710  1.265052 0.21208923
unemp       -0.06269441 0.01322938 -4.739031 0.00002021 ***
log(income)  1.78643540 0.64339251  2.776587 0.00786399 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.140556     Adj. R2: 0.926185
                 Within R2: 0.356781
```

This comprehensive model still produces a negative coefficient, though effect becomes insignificant with the addition of control variables.

```
# Create model list
fatal_models = list(
  fatal_cs,
  fatal_fe,
  fatal_twoway,
  fatal_full
)
# Generate comparison table
modelsummary(fatal_models, stars = TRUE)
```

The changing sign of the `beertax` coefficient across specifications illustrates the importance of controlling for unobserved heterogeneity in panel data:

	(1)	(2)	(3)	(4)
(Intercept)	1.853***			
	(0.119)			
beertax	0.365**	−0.656*	−0.640+	−0.456
	(0.120)	(0.292)	(0.357)	(0.307)
drinkage				−0.002
				(0.022)
punishyes				0.039
				(0.103)
miles				0.000
				(0.000)
unemp				−0.063***
				(0.013)
log(income)				1.786**
				(0.643)
Num.Obs.	336	336	336	335
R2	0.093	0.905	0.909	0.939
R2 Adj.	0.091	0.889	0.891	0.926
R2 Within		0.041	0.036	0.357
R2 Within Adj.		0.037	0.033	0.343
AIC	546.1	−117.9	−120.1	−243.9
BIC	553.7	69.1	89.9	−15.1
RMSE	0.54	0.18	0.17	0.14
Std.Errors	by: state	by: state	by: state	by: state
FE: state		X	X	X
FE: year			X	X

+ p <0.1, * p <0.05, ** p <0.01, *** p <0.001

1. In the pooled model, the positive coefficient might reflect that states with higher fatality rates tend to implement higher beer taxes as a policy response.
2. Once we control for state fixed effects, we isolate the within-state variation and find the expected negative relationship: when a state raises its beer tax, fatality rates decrease.
3. Adding year fixed effects accounts for national trends in fatality rates, such as changes in vehicle safety technology or nationwide campaigns against drunk driving.
4. In the full model with additional controls, the beer tax coefficient remains negative but loses statistical significance. This suggests that its effect may be partially captured by other policy variables or that we lack statistical power to precisely estimate the effect when including multiple controls.

7.10 R-codes

metrics-sec06.R

Part IV

Causal Inference

8 Endogeneity

8.1 The Linear Model and Exogeneity

So far we have written the conditional mean of an outcome Y_i as a linear function of observed covariates $\boldsymbol{X}_i$:

$$\begin{aligned} Y_i &= \boldsymbol{X}_i'\boldsymbol{\beta} + u_i, \\ E[u_i \mid \boldsymbol{X}_i] &= 0 \end{aligned} \tag{A1}$$

If (A1) holds, then $E[Y_i \mid \boldsymbol{X}_i] = \boldsymbol{X}_i'\boldsymbol{\beta}$, which makes $\boldsymbol{X}_i'\boldsymbol{\beta}$ the best predictor of Y_i given $\boldsymbol{X}_i$. Each coefficient β_j is a **conditional marginal effect**:

> **Interpretation:** *"Among individuals who share the same values of all included control variables, those whose X_{ij} is higher by one unit have, on average, a Y_i that is higher by β_j."*

So far the course has provided three empirical tactics to narrow the gap between correlation and causation:

- Add observed confounders. Whenever economic theory identifies a variable that influences both X_{ij} and Y_i, we try to measure it and augment $\boldsymbol{X}_i$.
- Exploit panel structure. With panel data data we include individual and time fixed effects to control for unobserved factors that are constant across individuals or time periods.
- Use flexible functional forms. Polynomials, interactions, or other transformations can absorb nonlinearities that would otherwise leak into u_i.

Even after taking these steps, important issues remain. For example, there may be reverse causality, which occurs when Y_i feeds back into X_i. Additionally, there may be control variables with a dual role that act as both confounders and mediators/colliders simultaneously.

Nothing in (A1) – nor in the additional assumptions (A2)–(A4) about i.i.d. sampling, finite moments, and full rank – guarantees that β_j is **causal**. It represents only a conditional **correlative** relationship unless X_{ij} is uncorrelated with all unobserved determinants of Y_i.

8.2 Conditional vs Causal Effects: Price Elasticities

Economists often want *causal* price effects, not merely conditional associations. Consider the following structural system in a competitive market written in logs so that slopes are elasticities:

$$\begin{aligned} \text{Demand:} \quad & \log(Q_i) = \beta_1 + \beta_2 \log(P_i) + u_i, \\ \text{Supply (pricing rule):} \quad & \log(P_i) = \gamma_1 + \gamma_2 \log(C_i) + \gamma_3 u_i + \eta_i. \end{aligned}$$

We have $\beta_2 < 0$ by theory.

- Index i denotes a market (e.g., city or store) observed at a single point in time; the data are cross-sectional and i.i.d.
- Q_i is the total quantity demanded in market i.
- P_i is price.
- C_i is the exogenous wholesale cost of the product.
- u_i captures consumers' taste shocks unobserved by the econometrician (though retailers may infer them and respond when setting prices); η_i captures supply-side shocks.

Because higher demand (large u_i) in a particular store leads retailers to charge higher prices ($\gamma_3 > 0$), we have $Cov(\log(P_i), u_i) > 0$. Hence, (A1) is violated in the demand equation.

Suppose a researcher estimates

$$\log(Q_i) = \alpha_1 + \alpha_2 \log(P_i) + \varepsilon_i$$

or

$$\log(Q_i) = \theta_1 + \theta_2 \log(P_i) + \theta_3 \log(C_i) + v_i$$

Both regressions (one simple and one with wholesale-cost controls) deliver conditional marginal effects α_2 or θ_2. They answer

> *"Among markets with the same wholesale cost (and any other included controls), how does observed quantity co-move with observed price?"*

But the policy-relevant question is different:

> *"By how much would quantity fall if we exogenously raised price – say, via a* 1% *tax – holding everything else constant?"*

That causal elasticity is β_2. Because P_i responds to u_i, OLS estimates suffer simultaneity bias and α_2 or θ_2 generally differ from β_2.

Endogeneity arises because we want the parameter to be causal, not because the regression is mechanically misspecified. Even if the conditional mean is correctly linear, interpreting β_2 causally implies $Cov(\log(P_i), u_i) \neq 0$.

8.3 Measurement Error

Another important source of endogeneity arises from measurement error. Suppose we consider the structural model:

$$Y_i^0 = \beta_1 + \beta_2 X_i^0 + u_i^0, \quad i = 1, \dots, n, \quad u_i^0 \sim \text{i.i.d.}(0, \sigma^2),$$

but we do not observe the latent variables Y_i^0 and X_i^0 directly. Instead, we observe:

$$Y_i = Y_i^0 + \eta_i, \quad X_i = X_i^0 + \zeta_i,$$

where $\eta_i \sim \text{i.i.d.}(0, \sigma_\eta^2)$ and $\zeta_i \sim \text{i.i.d.}(0, \sigma_\zeta^2)$ denote classical measurement errors that are assumed independent of each other and of X_i^0, Y_i^0, and u_i^0.

Plugging the observed variables into the structural equation yields:

$$Y_i - \eta_i = \beta_1 + \beta_2 (X_i - \zeta_i) + u_i^0,$$

which can be rearranged as:

$$Y_i = \beta_1 + \beta_2 X_i + \underbrace{(u_i^0 + \eta_i - \beta_2 \zeta_i)}_{\text{composite error term}}.$$

The composite error term is problematic:

$$E[u_i^0 + \eta_i - \beta_2 \zeta_i \mid X_i] \neq 0,$$

because X_i contains ζ_i, which also appears in the error term. This violates the exogeneity condition, resulting in a biased and inconsistent OLS estimator. Specifically, the bias tends to attenuate the coefficient estimate $\widehat{\beta}_2$ toward zero (known as attenuation bias). For positive true coefficients, this leads to underestimation; for negative coefficients, overestimation.

By contrast, if only the dependent variable Y_i is measured with error, OLS remains unbiased, although the variance of the error term increases.

8.4 Endogeneity as a Violation of (A1)

Formally, a regressor X_{ij} is **endogenous** if it correlates with the structural error term:

$$Cov(X_{ij}, u_i) \neq 0 \quad \Rightarrow \quad E[u_i \mid X_i] \neq 0$$

When this happens, OLS estimates remain descriptive but lose their causal interpretation. Whether you care depends on your goal:

Purpose	Is (A1) needed?	Parameter meaning
Prediction / description	*No.* Bias relative to causal truth is irrelevant if forecasting is the aim.	Conditional marginal effect
Causal policy evaluation	**Yes!** You need $E[u\|X] = 0$ in the causal sense, or an alternative identification strategy.	Structural (causal) effect

---

8.5 Sources of Endogeneity

Besides the functional-form misspecification that we have already discussed in previous sections, there are four other common sources of endogeneity in practice:

Mechanism	Typical manifestation
Omitted-variable bias	Unobserved ability affects both schooling (X) and wages (Y)
Simultaneity / reverse causality	Price and quantity determined jointly in markets
Measurement error in X	Measurement error inflates the variance of the regressor, so OLS slopes are biased toward zero (attenuation bias)
Dual role controls	A variable (e.g., health) acts as both confounder and mediator/collider

All four cases yield $E[\boldsymbol{u}|\boldsymbol{X}] \neq 0$ and threaten causal inference.

We have

$$E[\hat{\boldsymbol{\beta}}|\boldsymbol{X}] = \boldsymbol{\beta} + (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'E[\boldsymbol{u}|\boldsymbol{X}] \neq \boldsymbol{\beta}.$$

9 Instrumental Variables

```
library(fixest)
```

In Section 8, we discussed endogeneity problems that lead to the inconsistency of the ordinary least squares (OLS) estimator. One important solution is the **instrumental variables (IV)** method, which allows for consistent estimation under certain conditions when regressors are endogenous.

9.1 Endogenous Regressors Model

In most applications only a subset of the regressors are treated as endogenous.

Let's assume that we have k endogenous regressors $\boldsymbol{X}_i = (X_{i1}, \dots, X_{ik})'$ and r exogenous regressors $\boldsymbol{W}_i = (1, W_{i2}, \dots, W_{ir})'$.

In many practical applications the number of endogenous regressors k is small (like 1 or 2). The exogenous regressors $\boldsymbol{W}_i$ include the intercept, which is constant and therefore exogenous, and all control variables for which we do not wish to interpret their coefficients in a causal sense.

Consider the linear model equation:

$$Y_i = \boldsymbol{X}_i'\boldsymbol{\beta} + \boldsymbol{W}_i'\boldsymbol{\gamma} + u_i, \quad i = 1, \dots, n. \tag{9.1}$$

We have

- the dependent variable Y_i;
- the error term u_i;
- the endogenous regressors $\boldsymbol{X}_i = (X_{i1}, \dots, X_{ik})'$;
- the regression coefficients of interest $\boldsymbol{\beta}$;
- the remaining r regressors $\boldsymbol{W}_i = (1, W_{i2}, \dots, W_{ir})'$, which are assumed to be exogenous or simply control variables;
- the regression coefficients of the exogenous variables $\boldsymbol{\gamma}$.

Recall (A1), which is in this case given by $E[u_i|\boldsymbol{X}_i, \boldsymbol{W}_i] = 0$ but fails under endogeneity.

Since $\boldsymbol{X}_i$ is endogenous, we have $E[\boldsymbol{X}_i u_i] \neq \boldsymbol{0}$, which is a violation of (A1). Thus, the OLS estimator $\hat{\boldsymbol{\beta}}$ for $\boldsymbol{\beta}$ is inconsistent.

9.2 Instrumental Variables Model

To consistently estimate $\boldsymbol{\beta}$ in the endogenous regressors model we require additional information. One type of information which is commonly used in economic applications are what we call **instruments**.

A vector of **instrumental variables (IV)** $\boldsymbol{Z}_i = (Z_{i1}, \dots, Z_{im})$ for the endogenous variable X_{ij} is a variable that is

1) **relevant**, meaning that it has a non-zero conditional marginal effect on X_{ij} after controlling for $\boldsymbol{W}_i$. That is, when regressing X_{ij} on $\boldsymbol{Z}_i$ and $\boldsymbol{W}_i$ we have:

$$X_{ij} = \boldsymbol{Z}_i'\boldsymbol{\pi}_{1j} + \boldsymbol{W}_i'\boldsymbol{\pi}_{2j} + v_{ij}, \quad \boldsymbol{\pi}_{1j} \neq \mathbf{0}. \tag{9.2}$$

2) **exogenous** with respect to the error term u_i, i.e.:

$$E[\boldsymbol{Z}_i u_i] = \mathbf{0}. \tag{9.3}$$

   This means $\boldsymbol{Z}_i$ doesn't have a direct causal effect on Y_i after controlling for $\boldsymbol{W}_i$, only an indirect effect through the endogenous variable X_{ij}.

If there are k endogenous regressors, we need at least k instruments. If $m = k$, we say that the coefficients are exactly identified and if $m > k$ we say that they are overidentified. Then the relevance condition can be expressed jointly as:

$$\operatorname{rank}\Big(E[\tilde{\boldsymbol{Z}}_i \boldsymbol{X}_i']\Big) = k \tag{9.4}$$

where $\tilde{\boldsymbol{Z}}_i := (\boldsymbol{Z}_i', \boldsymbol{W}_i')'$.

Because $\boldsymbol{\pi}_{1j} \neq \mathbf{0}$, some part of the variation in X_{ij} can be explained by $\boldsymbol{Z}_i$. Because $\boldsymbol{Z}_i$ is exogenous, that part of the variation in X_{ij} explained by $\boldsymbol{Z}_i$ is exogenous as well and can be used to estimate β_j consistently.

Example 1: Years of schooling -> wage (returns to education). Ability bias: unobserved ability affects both education choices and wages. Possible instruments for years of schooling: distance to nearest colleges, school construction programs, quarter-of-birth, birth order.

Example 2: Market price -> quantity demanded (price elasticity of demand). Simultaneity: quantity demanded feeds back into equilibrium price. Possible instruments for market price: input-costs (e.g., raw materials, energy costs), weather conditions, tax changes.

Example 3: Police presence -> crime (deterrence effect). Reverse causality: more police are deployed to high-crime areas. Possible instruments for police presence: election cycles, sports/large public events, fire-fighters employment.

The idea of instrumental variable regression is to decompose the endogenous regressor X_{ij} into two parts: the "good" exogenous variation explained by the exogenous instruments $\boldsymbol{Z}_i$

and further exogenous control variables, and the "bad" endogenous variation that is correlated with the error term u_i.

This is exactly what is done in Equation 9.2: $\boldsymbol{Z}_i'\boldsymbol{\pi}_{1j}+\boldsymbol{W}_i'\boldsymbol{\pi}_{2j}$ is the part of X_{ij} that is exogenous and v_{ij} is the part of X_{ij} that is endogenous.

9.3 Two Stage Least Squares

The two stage least squares (TSLS) estimator exploits exactly the idea discussed above: first extracting the exogenous part of the endogenous regressors explained by the instruments as described in Equation 9.2 and then use only this exogenous part to estimate the causal relationship of interest.

The *first stage regression* is:

$$X_{ij} = \boldsymbol{Z}_i'\boldsymbol{\pi}_{1j} + \boldsymbol{W}_i'\boldsymbol{\pi}_{2j} + v_{ij}.$$

This equation is sometimes called the *reduced form* equation for X_{ij}. We estimate this regression for $j = 1, \dots, k$ and collect the fitted values

$$\widehat{X}_{ij} = \boldsymbol{Z}_i'\hat{\boldsymbol{\pi}}_{1j} + \boldsymbol{W}_i'\hat{\boldsymbol{\pi}}_{2j}, \quad j = 1, \dots, k, \quad i = 1, \dots, n.$$

Let

$$\widehat{\boldsymbol{X}}_i = (\widehat{X}_{i1}, \dots, \widehat{X}_{ik})', \quad i = 1, \dots, n.$$

be the vector of the fitted values for the k endogenous variables from the first stage.

Note that $\widehat{\boldsymbol{X}}_i$ is a function of $\boldsymbol{Z}_i$ and $\boldsymbol{W}_i$ and is therefore exogenous, i.e., uncorrelated with u_i.

Then, the *second stage regression* is

$$Y_i = \widehat{\boldsymbol{X}}_i'\boldsymbol{\beta} + \boldsymbol{W}_i'\boldsymbol{\gamma} + w_i, \quad i = 1, \dots, n. \tag{9.5}$$

Note that w_i absorbs v_{ij}, the part of X_{ij} that is endogenous. Therefore, the second stage regression does not suffer any more from an endogeneity problem and can be used to consistently estimate $\boldsymbol{\beta}$.

The OLS estimator of the second stage (Equation 9.5), denoted as $\hat{\boldsymbol{\beta}}_{TSLS}$ is called the **two-stage least squares estimator** for $\boldsymbol{\beta}$.

9.4 TSLS Assumptions

- (A1-iv) $E[u_i|\boldsymbol{W}_i] = 0$.
- (A2-iv) $(Y_i, \boldsymbol{X}_i', \boldsymbol{W}_i', \boldsymbol{Z}_i')_{i=1}^n$ is an i.i.d. sample.
- (A3-iv) All variables have finite kurtosis.
- (A4-iv) The instrument exogeneity and relevance conditions from Equation 9.3 and Equation 9.4 hold, and $E[\tilde{\boldsymbol{Z}}_i\tilde{\boldsymbol{Z}}_i^{'}]$ is invertible

(A1-iv) is the exogeneity condition for the control variables $\boldsymbol{W}_i$.

(A2-iv) is the standard random sampling assumption for the data.

(A3-iv) is the standard light-tails assumption, meaning large outliers are unlikely

(A4-iv) is the exogeneity and relevance condition for the instruments together with a condition that excludes perfect multicollinearity

9.5 Large-Sample Properties of TSLS

Under assumptions (A1-iv)–(A4-iv), the TSLS estimator is consistent:

$$\hat{\boldsymbol{\beta}}_{TSLS} \xrightarrow{p} \boldsymbol{\beta} \quad (\text{as } n \to \infty).$$

Furthermore, the estimator is asymptotically normal:

$$\sqrt{n}(\hat{\boldsymbol{\beta}}_{TSLS} - \boldsymbol{\beta}) \xrightarrow{d} \mathcal{N}(\boldsymbol{0}, \boldsymbol{V}_{TSLS}),$$

where

$$\boldsymbol{V}_{TSLS} = (\boldsymbol{Q}_{XZ}\boldsymbol{Q}_{ZZ}^{-1}\boldsymbol{Q}_{ZX})^{-1}\boldsymbol{Q}_{XZ}\boldsymbol{Q}_{ZZ}^{-1}\boldsymbol{\Omega}\boldsymbol{Q}_{ZZ}^{-1}\boldsymbol{Q}_{ZX}(\boldsymbol{Q}_{XZ}\boldsymbol{Q}_{ZZ}^{-1}\boldsymbol{Q}_{ZX})^{-1},$$

with

$$\boldsymbol{Q}_{XZ} = E[\boldsymbol{X}_i\tilde{\boldsymbol{Z}}_i^{'}], \quad \boldsymbol{Q}_{ZX} = E[\tilde{\boldsymbol{Z}}_i\boldsymbol{X}_i'], \quad \boldsymbol{Q}_{ZZ} = E[\tilde{\boldsymbol{Z}}_i\tilde{\boldsymbol{Z}}_i^{'}], \quad \boldsymbol{\Omega} = E[u_i^2\tilde{\boldsymbol{Z}}_i\tilde{\boldsymbol{Z}}_i^{'}].$$

The asymptotic variance can be estimated as:

$$\widehat{\boldsymbol{V}}_{TSLS} = \frac{n}{n-k-r}\left(\frac{1}{n}\sum_{i=1}^n \widehat{\boldsymbol{X}}_i\widehat{\boldsymbol{X}}_i^{'}\right)^{-1}\left(\frac{1}{n}\sum_{i=1}^n \hat{u}_i^2\widehat{\boldsymbol{X}}_i\widehat{\boldsymbol{X}}_i^{'}\right)\left(\frac{1}{n}\sum_{i=1}^n \widehat{\boldsymbol{X}}_i\widehat{\boldsymbol{X}}_i^{'}\right)^{-1}$$

This is the HC1 covariance matrix estimator for the TSLS estimator. It can be used to construct confidence intervals, t-tests, and F-tests in the usual way as discussed in previous sections.

9.6 Example: Return of Education

Consider a wage equation for a cross-section of 429 married women:

$$\log(\text{wage}) = \beta_1 + \beta_2 \text{educ}_i + \beta_3 \text{exper}_i + \beta_4 \text{exper}_i^2 + u_i,$$

where

- *wage* is the wife's 1975 average hourly earnings
- *educ* is her educational attainment in years
- *exper* are the actual years of her labor market experience

We use the dataset `mroz` available in this repository: link.

OLS yields:

```
feols(log(wage) ~ educ + exper + exper^2, data = mroz, vcov = "HC1")
```

```
OLS estimation, Dep. Var.: log(wage)
Observations: 428
Standard-errors: Heteroskedasticity-robust
             Estimate Std. Error  t value   Pr(>|t|)
(Intercept) -0.522041   0.201650 -2.58884 9.9611e-03 **
educ         0.107490   0.013219  8.13147 4.7203e-15 ***
exper        0.041567   0.015273  2.72156 6.7651e-03 **
I(exper^2)  -0.000811   0.000420 -1.93108 5.4139e-02 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.663299   Adj. R2: 0.150854
```

If *educ* is correlated with omitted variables like ability or motivation, the estimated coefficient for *educ* does not represent the causal effect of *educ* on *wage*.

Ability is an unobserved confounder that affects both *educ* and *wage*.

In the following, we assume that mother's education (*mothereduc*) is a valid instrument for *educ* in the wage equation because:

1) *mothereduc* should not appear in a wife's wage equation
2) Instrument relevance: *mothereduc* should be correlated with *educ*: high educated mothers typically have high educated daughters
3) Instrument exogeneity: assume that a woman's ability and motivation are uncorrelated with *mothereduc*

The first stage regression is:

```
firststage = lm(educ ~ mothereduc + exper + I(exper^2), data = mroz)
firststage
```

```
Call:
lm(formula = educ ~ mothereduc + exper + I(exper^2), data = mroz)

Coefficients:
(Intercept)   mothereduc        exper   I(exper^2)
   9.775103     0.267691     0.048862    -0.001281
```

The second stage regression is:

```
Xhat = firststage$fitted.values
secondstage = lm(log(wage) ~ Xhat + exper + I(exper^2), data = mroz)
secondstage
```

```
Call:
lm(formula = log(wage) ~ Xhat + exper + I(exper^2), data = mroz)

Coefficients:
(Intercept)         Xhat        exper   I(exper^2)
  0.1981861    0.0492630    0.0448558   -0.0009221
```

Note that standard errors from these two separate steps should not be used. Instead, the `feols` function gives you the correct standard errors by using the following notation:

```
feols(log(wage) ~ exper + exper^2 | educ ~ mothereduc, data = mroz, vcov = "HC1")
```

```
TSLS estimation - Dep. Var.: log(wage)
                  Endo.    : educ
                  Instr.   : mothereduc
Second stage: Dep. Var.: log(wage)
Observations: 428
Standard-errors: Heteroskedasticity-robust
             Estimate Std. Error   t value  Pr(>|t|)
(Intercept)  0.198186   0.489146  0.405167 0.6855588
```

```
fit_educ     0.049263   0.038040  1.295045 0.1960095
exper        0.044856   0.015604  2.874667 0.0042481 **
I(exper^2)  -0.000922   0.000432 -2.135025 0.0333316 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.67642   Adj. R2: 0.116926
F-test (1st stage), educ: stat = 73.9   , p < 2.2e-16 , on 1 and 424 DoF.
              Wu-Hausman: stat =  2.9683, p = 0.085642, on 1 and 423 DoF.
```

- The coefficient for *educ* drops from 0.107 to 0.059
- OLS overestimates the impact of education on wages
- The t-statistic has a p-value of 0.19
- Using *mothereduc* as an instrument, *educ* is no longer significant

To improve the precision of the IV estimator, we can include further instruments like *fathereduc*

```
feols(log(wage) ~ exper + exper^2 | educ ~ mothereduc + fathereduc, data = mroz, vcov = "HC1
```

```
TSLS estimation - Dep. Var.: log(wage)
                  Endo.    : educ
                  Instr.   : mothereduc, fathereduc
Second stage: Dep. Var.: log(wage)
Observations: 428
Standard-errors: Heteroskedasticity-robust
             Estimate Std. Error   t value  Pr(>|t|)
(Intercept)  0.048100   0.429798  0.111914 0.9109447
fit_educ     0.061397   0.033339  1.841609 0.0662307 .
exper        0.044170   0.015546  2.841202 0.0047111 **
I(exper^2)  -0.000899   0.000430 -2.090220 0.0371931 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.671551   Adj. R2: 0.129593
F-test (1st stage), educ: stat = 55.4     , p < 2.2e-16 , on 2 and 423 DoF.
              Wu-Hausman: stat =  2.79259 , p = 0.095441, on 1 and 423 DoF.
                  Sargan: stat =  0.378071, p = 0.538637, on 1 DoF.
```

- Estimated return to education increases from 0.049 to 0.061
- The t-statistic has a p-value of 0.066
- Stronger instruments leads to more efficient IV estimation: *educ* is now significantly different from zero at least at the 10% level.

9.7 IV Diagnostics

The TSLS estimator relies on the exogeneity and relevance of the instruments. In empirical applications, these assumptions should be critically assessed. This section introduces three diagnostic tools used to evaluate different aspects of the IV strategy:

- The **F-test** for instrument relevance
- The **Sargan test** for instrument exogeneity
- The **Wu-Hausman test** for regressor endogeneity

F-test for instrument relevance

The **first-stage F-test** indicates whether the instruments $\boldsymbol{Z}_i \in \mathbb{R}^m$ contain enough information about the endogenous regressors $\boldsymbol{X}_i \in \mathbb{R}^k$, conditional on the exogenous controls $\boldsymbol{W}_i$.

Consider the one endogenous regressor $k = 1$ case with the first-stage regression,

$$X_i = \boldsymbol{Z}_i'\boldsymbol{\pi}_1 + \boldsymbol{W}_i'\boldsymbol{\pi}_2 + v_i,$$

and test the joint null hypothesis

$$H_0 : \boldsymbol{\pi}_1 = \mathbf{0}.$$

To compute the F-statistic for this hypothesis, we follow the usual procedure and use a suitable robust covariance matrix (e.g., HC1 or cluster-robust), with an F-statistic whose null distribution is asymptotically $F_{m,\infty}$.

If the statistic exceeds its critical value you reject H_0 and conclude the instruments are *relevant*.

Large-n 5% critical values for $F_{m,\infty}$ are 3.84 for $m = 1$, 3.00 for $m = 2$, 2.60 for $m = 3$, etc. (compute with `qf(.95, m, Inf)`).

Weak instruments

Relevance alone is not enough: the instruments may be **weak** if their correlation with X_i is small. Weakness matters because two-stage least squares (2SLS) can then suffer a large finite-sample bias toward OLS. Define the *relative bias*

$$\text{relBias} = \frac{E[\hat{\beta}_{TSLS}] - \beta}{E[\hat{\beta}_{OLS}] - \beta}.$$

Staiger and Stock (1997) and Stock and Yogo (2005) derive critical values for the homoskedastic first-stage statistic that control the null hypothesis "relative bias $>$ 10% of the OLS bias" at

the 5% significance level. With one instrument the 5% cut-off is approximately **10**. Hence, the following rule of thumb is established in applied work:

$$\begin{aligned} \text{First-stage } F > 10 \quad &\Rightarrow \quad \text{instruments strong} \\ \text{First-stage } F \leq 10 \quad &\Rightarrow \quad \text{instruments weak} \end{aligned}$$

This is a quick approximation that relies on the homoskedasticity assumption and only works well when m is small.

For heteroskedastic (or cluster-robust) settings, Montiel Olea and Pflueger (2013) replace the standard rule of thumb: To reject the null hypothesis of a relative bias larger than 10% at the 5% level you need a robust F-statistic that exceeds its critical value, which varies between about **11 and 23.1** depending on m and the estimated error-covariance matrix (HC1, cluster-robust, HAC, etc.). The conservative rule

$$\begin{aligned} \text{First-stage robust } F > 23.1 \quad &\Rightarrow \quad \text{instruments strong} \\ \text{First-stage robust } F \leq 23.1 \quad &\Rightarrow \quad \text{instruments weak} \end{aligned}$$

is therefore *sufficient* (but not always necessary) for any number of instruments when $k = 1$.

If several regressors are endogenous ($k \geq 2$), each has its own first-stage equation, and the scalar F no longer summarizes the joint instrument strength. An alternative is the matrix-based Kleibergen–Paap tests of Kleibergen and Paap (2006), which extend the Staiger-Stock-Yogo logic to the multivariate case.

Anderson-Rubin Test

To conduct inference when the first-stage is weak, the usual TSLS t-, F- or Wald tests are unreliable – they tend to over-reject and their confidence intervals undercover.

A simple, robust alternative is the **Anderson–Rubin (AR) test**. The logic is that, under the structural model, the instruments $\boldsymbol{Z}_i$ should contain no information about the structural error

$$u_i = Y_i - \boldsymbol{X}_i'\boldsymbol{\beta} - \boldsymbol{W}_i'\boldsymbol{\gamma}.$$

Hence, if the null hypothesis $H_0 : \boldsymbol{\beta} = \boldsymbol{\beta}_0$ holds, the adjusted outcome $Y_i - \boldsymbol{X}_i'\boldsymbol{\beta}_0$ must be uncorrelated with the instruments conditional on the controls. In practice one runs the auxiliary regression

$$Y_i - \boldsymbol{X}_i'\boldsymbol{\beta}_0 = \boldsymbol{Z}_i'\boldsymbol{\pi} + \boldsymbol{W}_i'\boldsymbol{\theta} + e_i$$

and computes the heteroskedastic- or cluster-robust F-statistic, F_{rob}, for the joint null $\boldsymbol{\pi} = \mathbf{0}$ (numerator d.f. $= m$). Reject H_0 when

$$F_{\text{rob}} > F_{m,\infty;1-\alpha},$$

where m is the number of instruments. This decision rule delivers correct size regardless of instrument strength, but it has lower power than the TSLS-based tests when instruments are strong.

Repeating the test over a grid of candidate $\boldsymbol{\beta}_0$ values and retaining those not rejected yields a $(1-\alpha)$ Anderson–Rubin confidence region that remains valid even when the first-stage F is very small.

Sargan Test for Instrument Exogeneity

When the set of instruments is overidentified ($m > k$), we can statistically assess whether all instruments satisfy the exogeneity condition $E[\boldsymbol{Z}_i u_i] = 0$.
The classical procedure is the **Sargan test** (also called the test of over-identifying restrictions or the J-test).

Null and alternative hypotheses

- H_0 (all instruments are valid): every instrument is uncorrelated with the structural error term u_i.
- H_1 (at least one instrument is invalid): some instrument is correlated with u_i.

Computation of the Sargan J-statistic

1. **Estimate the structural equation by TSLS** (using *all* m instruments) and obtain the residuals
$$\hat{u}_i^{\text{TSLS}} = Y_i - (\boldsymbol{X}_i'\hat{\boldsymbol{\beta}}_{TSLS} + \boldsymbol{W}_i'\hat{\boldsymbol{\gamma}}_{TSLS}).$$
2. **Regress $\hat{u}_i^{\text{TSLS}}$ on the full set of instruments and exogenous controls**
$$\hat{u}_i^{\text{TSLS}} = \delta_0 + \delta_1 Z_{i1} + \cdots + \delta_m Z_{im} + \boldsymbol{W}_i'\boldsymbol{\theta} + e_i.$$
3. Let F be the (homoskedastic-only) F-statistic for the joint null $\delta_1 = \cdots = \delta_m = 0$. The Sargan statistic is
$$J = m \cdot F.$$

Under H_0 and homoskedastic errors, $J \sim \chi^2_{m-k}$ in large samples .

If heteroskedasticity is suspected, the Hansen robust J-statistic should be used.

Decision rule and interpretation

- **Reject** H_0 if J exceeds the critical value of the χ^2_{m-k} distribution (or if the p-value is below the chosen significance level). This implies that the data are inconsistent with the joint exogeneity of the instruments; at least one instrument is likely invalid.
- **Fail to reject** H_0 when J is small. This provides no evidence *against* instrument validity, but does not prove exogeneity.

Practical remarks

- The test cannot be performed when the model is exactly identified ($m = k$); then $J = 0$ by construction and instrument validity must be argued on theoretical grounds.
- A significant J-statistic tells us that something is wrong with the instrument set, but not which instrument(s) are problematic. Empirical judgment and auxiliary tests (e.g. re-estimating with different subsets of instruments) are required.

9.7.1 Wu-Hausman Test for Endogeneity

The **Wu-Hausman test** evaluates whether the regressors $\boldsymbol{X}_i$ are in fact endogenous. That is, it tests the null hypothesis of exogeneity, i.e.: $H_0 : E[\boldsymbol{X}_i u_i] = \boldsymbol{0}$.

Recall the first stage regressions

$$X_{ij} = \boldsymbol{Z}_i'\boldsymbol{\pi}_{1j} + \boldsymbol{W}_i'\boldsymbol{\pi}_{2j} + v_{ij}, \quad j = 1, \dots, k,$$

and let $\boldsymbol{v}_i = (v_{i1}, \dots, v_{ik})'$ be the stacked error terms of the first-stage regressions.

As discussed previously, $\boldsymbol{Z}_i'\boldsymbol{\pi}_{1j} + \boldsymbol{W}_i'\boldsymbol{\pi}_{2j}$ represents the exogenous part of X_{ij} and v_{ij} the endogenous part. Thus, $\boldsymbol{v}_i$ is the endogenous part of the full vector of endogenous regressors $\boldsymbol{X}_i$. Therefore,

$$E[\boldsymbol{X}_i u_i] = \boldsymbol{0} \quad \Leftrightarrow \quad E[\boldsymbol{v}_i u_i] = \boldsymbol{0}.$$

Consider $\boldsymbol{\delta} = E[\boldsymbol{v}_i \boldsymbol{v}_i']^{-1} E[\boldsymbol{v}_i u_i]$, which is the population regression coefficient of the auxiliary regression

$$u_i = \boldsymbol{v}_i'\boldsymbol{\delta} + \epsilon_i, \quad E[\boldsymbol{v}_i \epsilon_i] = 0. \tag{9.6}$$

From the definition of $\boldsymbol{\delta}$ we see that

$$\boldsymbol{\delta} = \boldsymbol{0} \quad \Leftrightarrow \quad E[\boldsymbol{v}_i u_i] = \boldsymbol{0}.$$

Therefore, testing $H_0 : E[\boldsymbol{X}_i u_i] = \boldsymbol{0}$ is equivalent to testing $\boldsymbol{\delta} = \boldsymbol{0}$.

Note that Equation 9.6 is an infeasible regression because u_i and $\boldsymbol{v}_i$ are unknown. While $\boldsymbol{v}_i$ can be estimated using the residuals $\hat{\boldsymbol{v}}_i$ from the first-stage regressions, there are no suitable sample counterparts for u_i available under endogeneity.

We may insert Equation 9.6 into the structural equation given by Equation 9.1:

$$Y_i = \boldsymbol{X}_i'\boldsymbol{\beta} + \boldsymbol{W}_i'\boldsymbol{\gamma} + \boldsymbol{v}_i'\boldsymbol{\delta} + \epsilon_i. \tag{9.7}$$

Equation 9.7 is a well defined regression model with regressors $\boldsymbol{X}_i, \boldsymbol{W}_i, \boldsymbol{v}_i$ and regression error ϵ_i. To see this note that

(i) $E[\boldsymbol{v}_i \epsilon_i] = \boldsymbol{0}$ by Equation 9.6;
(ii) $E[\boldsymbol{W}_i \epsilon_i] = \boldsymbol{0}$ because $\boldsymbol{W}_i$ are exogenous;
(iii) $E[\boldsymbol{X}_i \epsilon_i] = \boldsymbol{0}$ because $E[\boldsymbol{X}_i \epsilon_i] = E[\boldsymbol{v}_i \epsilon_i]$.

Therefore, we may apply an F-test on the restriction $\boldsymbol{\delta} = \boldsymbol{0}$ in Equation 9.7 when $\boldsymbol{v}_i$ is replaced by $\hat{\boldsymbol{v}}_i$, which is known as the Wu-Hausman test.

Wu-Hausman Procedure:

1. Run the first-stage regression for each endogenous regressor X_{ij} and obtain residuals $\hat{v}_{ij}$, $j = 1, \dots, k$.
2. Stack the residuals as $\hat{\boldsymbol{v}}_i = (\hat{v}_{i1}, \dots, \hat{v}_{ik})'$.
3. Run the **augmented regression**:

$$Y_i = \boldsymbol{X}_i'\boldsymbol{\beta} + \boldsymbol{W}_i'\boldsymbol{\gamma} + \hat{\boldsymbol{v}}_i'\boldsymbol{\delta} + \varepsilon_i.$$

4. Test $H_0 : \boldsymbol{\delta} = \boldsymbol{0}$ using an F-test or Wald test, which has k restrictions.

If the test does not reject H_0, then there is evidence for exogenous regressors with $E[\boldsymbol{X}_i u_i] = 0$, and the conventional OLS without instruments should be used because it is more efficient than TSLS.

9.8 Example: Return of Education Revisited

Recall the previous TSLS regression with instrument `mothereduc`

```
feols(log(wage) ~ exper + exper^2 | educ ~ mothereduc, data = mroz, vcov = "HC1")
```

```
TSLS estimation - Dep. Var.: log(wage)
                  Endo.    : educ
                  Instr.   : mothereduc
Second stage: Dep. Var.: log(wage)
Observations: 428
Standard-errors: Heteroskedasticity-robust
             Estimate Std. Error   t value  Pr(>|t|)
(Intercept)  0.198186   0.489146  0.405167 0.6855588
fit_educ     0.049263   0.038040  1.295045 0.1960095
exper        0.044856   0.015604  2.874667 0.0042481 **
I(exper^2)  -0.000922   0.000432 -2.135025 0.0333316 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.67642   Adj. R2: 0.116926
F-test (1st stage), educ: stat = 73.9   , p < 2.2e-16 , on 1 and 424 DoF.
              Wu-Hausman: stat =  2.9683, p = 0.085642, on 1 and 423 DoF.
```

The first stage F-statistic is 73.9 indicating that the instrument is strong. The Wu-Hausman statistic has a p-value of 0.08, which indicates that `educ` is significantly endogenous at the 10% level. The Sargan test is not displayed because of exact identification.

We also discussed the TSLS results with two instruments:

```
feols(log(wage) ~ exper + exper^2 | educ ~ mothereduc + fathereduc, data = mroz, vcov = "HC1"
```

```
TSLS estimation - Dep. Var.: log(wage)
                  Endo.    : educ
                  Instr.   : mothereduc, fathereduc
Second stage: Dep. Var.: log(wage)
Observations: 428
Standard-errors: Heteroskedasticity-robust
             Estimate Std. Error   t value  Pr(>|t|)
(Intercept)  0.048100   0.429798  0.111914 0.9109447
fit_educ     0.061397   0.033339  1.841609 0.0662307 .
exper        0.044170   0.015546  2.841202 0.0047111 **
I(exper^2)  -0.000899   0.000430 -2.090220 0.0371931 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
RMSE: 0.671551   Adj. R2: 0.129593
F-test (1st stage), educ: stat = 55.4     , p < 2.2e-16 , on 2 and 423 DoF.
              Wu-Hausman: stat =  2.79259 , p = 0.095441, on 1 and 423 DoF.
                  Sargan: stat =  0.378071, p = 0.538637, on 1 DoF.
```

Similarly, the F-statistic of 55.4 indicates that the instruments are strong and the Wu-Hausman test gives some statistical evidence of an endogeneity problem. The Sargan test does not reject, which indicates no evidence against instrument validity (but does not prove exogeneity of the instruments).

9.9 R-codes

metrics-sec09.R

References

Kleibergen, Frank, and Richard Paap. 2006. "Generalized Reduced Rank Tests Using the Singular Value Decomposition." *Journal of Econometrics* 133 (1): 97–126.

Montiel Olea, José Luis, and Carolin Pflueger. 2013. "A Robust Test for Weak Instruments." *Journal of Business & Economic Statistics* 31 (3): 358–69.

Staiger, Douglas, and James H Stock. 1997. "Instrumental Variables Regression with Weak Instruments." *Econometrica* 65 (3): 557–86.

Stock, James H., and Motohiro Yogo. 2005. "Testing for Weak Instruments in Linear IV Regression." In *Identification and Inference for Econometric Models: Essays in Honor of Thomas Rothenberg*, edited by Donald W. K. Andrews and James H.Editors Stock, 80–108. Cambridge University Press.

Part V

Big Data Econometrics

10 Shrinkage Estimation

```
library(glmnet)
```

Shrinkage estimation is a highly valuable technique in the context of high-dimensional regression analysis. It allows for the estimation of regression models with more regressors than observations.

10.1 Mean squared error

The key measure of estimation accuracy is the **mean squared error (MSE)**. The MSE of an estimator $\hat{\theta}$ for a parameter θ is

$$MSE(\hat{\theta}) = E[(\hat{\theta} - \theta)^2].$$

The MSE can be decomposed into the variance plus squared bias:

$$MSE(\hat{\theta}) = \underbrace{E[(\hat{\theta} - E[\hat{\theta}])^2]}_{=Var[\hat{\theta}]} + \underbrace{(E[\hat{\theta}] - \theta)^2}_{=Bias(\hat{\theta})^2}$$

Proof. Subtracting and adding $E[\hat{\theta}]$ gives

$$\begin{aligned}(\hat{\theta} - \theta)^2 &= (\hat{\theta} - E[\hat{\theta}] + E[\hat{\theta}] - \theta)^2 \\ &= (\hat{\theta} - E[\hat{\theta}])^2 + 2(\hat{\theta} - E[\hat{\theta}])(\underbrace{E[\hat{\theta}] - \theta}_{Bias(\hat{\theta})}) + \underbrace{(E[\hat{\theta}] - \theta)^2}_{=Bias(\hat{\theta})^2}.\end{aligned}$$

The middle term is zero after taking the expectation:

$$E[(\hat{\theta} - \theta)^2] = \underbrace{E[(\hat{\theta} - E[\hat{\theta}])^2]}_{=Var[\hat{\theta}]} + 2\underbrace{E[\hat{\theta} - E[\hat{\theta}]]}_{=0} Bias(\hat{\theta}) + Bias(\hat{\theta})^2.$$

□ □

For instance, consider an i.i.d. sample $X_1, \dots, X_n$ with population mean $E[X_i] = \mu$ and variance $Var[X_i] = \sigma^2$. Let's study the sample mean

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} X_i$$

as an estimator of μ. You will find that

$$E[\hat{\mu}] = \mu, \quad Var[\hat{\mu}] = \frac{\sigma^2}{n}.$$

Proof. By the linearity of the expectation, we have

$$E[\hat{\mu}] = \frac{1}{n} \sum_{i=1}^{n} \underbrace{E[X_i]}_{\mu} = \mu.$$

The independence of $X_1, \dots, X_n$ implies

$$Var[\hat{\mu}] = \frac{1}{n^2} Var\left[\sum_{i=1}^{n} X_i\right] = \frac{1}{n^2} \sum_{i=1}^{n} Var[X_i] = \frac{\sigma^2}{n}$$

□ □

The sample mean is unbiased for μ, i.e., $Bias(\hat{\mu}) = E[\hat{\mu}] - \mu = 0$. The MSE equals its variance:

$$MSE(\hat{\mu}) = \frac{\sigma^2}{n}.$$

The sample mean is the best unbiased estimator for the population mean, but there exists estimators with a lower MSE if we allow for a small bias.

10.2 A simple shrinkage estimator

Let us shrink our sample mean a bit towards 0 and define the alternative estimator

$$\tilde{\mu} = (1 - w)\hat{\mu}, \quad w \in [0, 1].$$

Setting the shrinkage weight to $w = 0$ gives $\tilde{\mu} = \hat{\mu}$ (no shrinkage) and $w = 1$ gives $\tilde{\mu} = 0$ (full shrinkage). Our shrinkage estimator has the bias

$$Bias(\tilde{\mu}) = E[(1 - w)\hat{\mu}] - \mu = (1 - w) \underbrace{E[\hat{\mu}]}_{=\mu} - \mu = -w\mu.$$

The variance is

$$Var[\tilde{\mu}] = Var[(1-w)\hat{\mu}] = (1-w)^2 Var[\hat{\mu}] = (1-w)^2 \frac{\sigma^2}{n},$$

and the MSE is

$$MSE(\tilde{\mu}) = Var[\tilde{\mu}] + Bias(\tilde{\mu})^2 = (1-w)^2 \frac{\sigma^2}{n} + w^2\mu^2.$$

The optimal weight in terms of the MSE is

$$w^* = \frac{1}{1 + n\mu^2/\sigma^2}$$

Proof. We take the derivative of $mse(\tilde{\mu})$ across w to obtain the first order condition:

$$-2(1-w)\sigma^2/n + 2w\mu^2 = 0.$$

Solving for w gives $w(1 + n\mu^2/\sigma^2) = 1$. Then, w^* is the global minimum because the second derivative is $2\sigma^2/n + 2\mu^2 > 0$. □ □

For instance, if $\mu = 1$, $\sigma^2 = 1$, and $n = 99$, we have $w^* = 0.01$.

The shrunk sample mean

$$\tilde{\mu}^* = (1-w^*)\hat{\mu} = \frac{n\mu^2/\sigma^2}{1 + n\mu^2/\sigma^2} \frac{1}{n} \sum_{i=1}^{n} X_i$$

has a lower MSE than the usual sample mean:

$$MSE(\tilde{\mu}^*) = (1-w^*)^2 \frac{\sigma^2}{n} + (w^*)^2\mu^2 < \frac{\sigma^2}{n} = mse(\hat{\mu})$$

This is a remarkable result because it tells us that the sample mean is not the best we can do in the MSE sense to estimate a population mean. The shrinked estimator is more efficient. It is biased, but the bias vanishes asymptotically since $\lim_{n\to\infty} w^* = 0$.

The optimal shrinkage parameter w^* is infeasible because μ^2/σ^2 is unknown. While insightful theoretically, this result is not directly applicable in empirical work, and taking sample means is still recommended.

However, the shrinkage principle can be very useful in the context of high-dimensional regression.

10.3 High-dimensional regression

Least squares regression works well when the number of regressors k is small relative to the number of observations n. In a previous section on "too many regressors", we discussed how ordinary least squares (OLS) can overfit when k is too large compared to n. Specifically, if $k = n$, the OLS regression line perfectly fits the data.

Many economic applications involve categorical variables that are transformed into a large number of dummy variables. If we include pairwise interaction terms among J variables, we get another $\sum_{i=1}^{J-1} i = J(J-1)/2$ regressors (for example, 190 for J=20 and 4950 for J=100).

Accounting for further nonlinearities by adding squared and cubic terms or higher-order interactions can result in thousands or even millions of regressors. Many of these regressors may provide low informational value, but it is difficult to determine a priori which are relevant and which are irrelevant.

If $k > n$, the OLS estimator is not uniquely defined because $\boldsymbol{X'X}$ does not have full rank. If $k \approx n$ the matrix $\boldsymbol{X'X}$ can be near singular, resulting in numerically unstable OLS coefficients or high variance.

For the vector-valued (k-variate) estimator $\hat{\boldsymbol{\beta}}_{ols}$ the (conditional) MSE is

$$\begin{aligned} MSE(\hat{\boldsymbol{\beta}}_{ols}|\boldsymbol{X}) &= E[(\hat{\boldsymbol{\beta}}_{ols} - \boldsymbol{\beta})'(\hat{\boldsymbol{\beta}}_{ols} - \boldsymbol{\beta})|\boldsymbol{X}] \\ &= Var[\hat{\boldsymbol{\beta}}_{ols}|\boldsymbol{X}] + Bias(\hat{\boldsymbol{\beta}}_{ols}|\boldsymbol{X})(Bias(\hat{\boldsymbol{\beta}}_{ols}|\boldsymbol{X}))', \end{aligned}$$

where, under random sampling, OLS is unbiased:

$$Bias(\hat{\boldsymbol{\beta}}_{ols}|\boldsymbol{X}) = E[\hat{\boldsymbol{\beta}}_{ols}|\boldsymbol{X}] - \boldsymbol{\beta} = \boldsymbol{0}.$$

Consequently, the MSE of OLS equals its variance:

$$MSE(\hat{\boldsymbol{\beta}}_{ols}|\boldsymbol{X}) = Var[\hat{\boldsymbol{\beta}}_{ols}|\boldsymbol{X}] = (\boldsymbol{X'X})^{-1}\boldsymbol{X'DX}(\boldsymbol{X'X})^{-1}.$$

10.4 Ridge Regression

To avoid that $(\boldsymbol{X'X})^{-1}$ becomes very large or undefined for large k, we can introduce a shrinkage parameter λ and define the **ridge regression estimator**

$$\hat{\boldsymbol{\beta}}_{ridge} = (\boldsymbol{X'X} + \lambda \boldsymbol{I}_k)^{-1}\boldsymbol{X'Y}. \tag{10.1}$$

This estimator is well defined and does not suffer from multicollinearity problems, even if $k > n$. The inverse $(\boldsymbol{X'X} + \lambda \boldsymbol{I}_k)^{-1}$ exists as long as $\lambda > 0$. For $\lambda = 0$, the ridge estimator coincides with the OLS estimator.

While the OLS estimator is motivated from the minimization problem

$$\min_{\boldsymbol{\beta}}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}),$$

the ridge estimator is the minimizer of

$$\min_{\boldsymbol{\beta}}(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta})+\lambda\boldsymbol{\beta}'\boldsymbol{\beta}. \tag{10.2}$$

The minimization problem introduces a penalty for large values of $\boldsymbol{\beta}$. The solution is then shrunk towards zero by $\lambda > 0$.

10.5 Standardization

It is common practice to standardize the regressors in ridge regression:

$$\widetilde{X}_{ij}=\frac{X_{ij}-\overline{\boldsymbol{X}}_j}{\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(X_{ij}-\overline{\boldsymbol{X}}_j)^2}},\quad \overline{\boldsymbol{X}}_j=\frac{1}{n}\sum_{i=1}^{n}X_{ij}$$

Without standardization, variables with larger scales (i.e., larger variances) will disproportionately influence the penalty term through $\lambda\boldsymbol{\beta}'\boldsymbol{\beta}=\lambda\sum_{j=1}^{k}\beta_j^2$. Variables with smaller variance may be under-penalized, while those with larger variance may be over-penalized.

Standardization ensures that each variable contributes equally to the penalty term, making the penalty independent of the scale of the variables.

Standardizing makes the coefficient estimates more interpretable, as they will all be on the same scale, which helps in understanding the relative importance of each variable.

10.6 Ridge Properties

The bias of the ridge estimator is

$$Bias(\hat{\boldsymbol{\beta}}_{ridge}|\boldsymbol{X})=-\lambda(\boldsymbol{X}'\boldsymbol{X}+\lambda\boldsymbol{I}_k)^{-1}\boldsymbol{\beta},$$

and the covariance matrix is

$$Var[\hat{\boldsymbol{\beta}}_{ridge}|\boldsymbol{X}]=(\boldsymbol{X}'\boldsymbol{X}+\lambda\boldsymbol{I}_k)^{-1}\boldsymbol{X}'\boldsymbol{D}\boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X}+\lambda\boldsymbol{I}_k)^{-1}.$$

In the homoskedastic linear regression model, we have

$$MSE(\hat{\boldsymbol{\beta}}_{ridge}|\boldsymbol{X})<MSE(\hat{\boldsymbol{\beta}}_{ols}|\boldsymbol{X})$$

if $0<\lambda<2\sigma^2/\boldsymbol{\beta}'\boldsymbol{\beta}$.

Similarly to the sample mean case, the upper bound $2\sigma^2/\boldsymbol{\beta}'\boldsymbol{\beta}$ does not give practical guidance for selecting λ because $\boldsymbol{\beta}$ and σ^2 are unknown.

10.7 Mean squared prediction error

The optimal value for λ minimizes the MSE, but estimating the MSE of the ridge estimator is not straightforward because it depends on the parameter $\boldsymbol{\beta}$ being estimated. Instead, it is better to focus on the out-of-sample mean squared prediction error (MSPE).

Let $(Y_1, \boldsymbol{X}_1), \ldots, (Y_n, \boldsymbol{X}_n)$ be our data set (in-sample observations) with ridge estimator Equation 10.1, and let $(Y^{oos}, \boldsymbol{X}^{oos})$ be another observation pair (out-of-sample observation) that is independently drawn from the same population as $(Y_1, \boldsymbol{X}_1), \ldots, (Y_n, \boldsymbol{X}_n)$.

The mean squared prediction error (MSPE) is

$$MSPE(\hat{\boldsymbol{\beta}}_{ridge}) = E[(Y^{oos} - (\boldsymbol{X}^{oos})'\hat{\boldsymbol{\beta}}_{ridge})^2].$$

Note that $(Y^{oos}, \boldsymbol{X}^{oos})$ is independent of $\hat{\boldsymbol{\beta}}_{ridge}$ because it has not been used for estimation. $\widehat{Y}(\boldsymbol{X}^{oos}) = (\boldsymbol{X}^{oos})'\hat{\boldsymbol{\beta}}_{ridge}$ is the predicted value of Y^{oos}.

To estimate the MSPE, we can use a **split sample**.

1) We divide our observations randomly into a training sample (in-sample) of size n_{train} and a testing sample (out-of-sample) of size n_{test} with $n = n_{train} + n_{test}$:

$$(Y_1^{ins}, \boldsymbol{X}_1^{ins}), \ldots (Y_{n_{train}}^{ins}, \boldsymbol{X}_{n_{train}}^{ins}), \quad (Y_1^{oos}, \boldsymbol{X}_1^{oos}), \ldots (Y_{n_{test}}^{oos}, \boldsymbol{X}_{n_{test}}^{oos})$$

2) We estimate $\boldsymbol{\beta}$ using the training sample:

$$\hat{\boldsymbol{\beta}}_{ridge}^{ins} = \left(\sum_{i=1}^{n_{train}} \boldsymbol{X}_i^{ins} (\boldsymbol{X}_i^{ins})' + \lambda \boldsymbol{I}_k \right)^{-1} \sum_{i=1}^{n_{train}} \boldsymbol{X}_i^{ins} Y_i^{ins}.$$

3) We evaluate the empirical MSPE using the testing sample,

$$\widehat{MSPE}_{split} = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \left(Y_i^{oos} - (\boldsymbol{X}_i^{oos})' \hat{\boldsymbol{\beta}}_{ridge}^{ins} \right)^2 \tag{10.3}$$

Steps 2 and 3 are repeated for different values for λ. We select the value for λ that gives the smallest estimated MSPE.

10.8 Cross validation

A problem with the split sample estimator is that it highly depends on the choice of the two subsamples. An alternative is to select m subsamples (folds) and evaluate the MSPE using each fold separately:

m-fold cross validation

1) Divide the sample into $j = 1, \dots, m$ randomly chosen folds/subsamples of approximately equal size:

$$\begin{aligned}
&(Y_1^{(1)}, \boldsymbol{X}_1^{(1)}), \dots, (Y_{n_1}^{(1)}, \boldsymbol{X}_{n_1}^{(1)}) \\
&(Y_1^{(2)}, \boldsymbol{X}_1^{(2)}), \dots, (Y_{n_2}^{(2)}, \boldsymbol{X}_{n_2}^{(2)}) \\
&\quad \vdots \\
&(Y_1^{(m)}, \boldsymbol{X}_1^{(m)}), \dots, (Y_{n_m}^{(m)}, \boldsymbol{X}_{n_m}^{(m)})
\end{aligned}$$

2) Select $j \in \{1, \dots, m\}$ as left-out test sample and use the other subsamples to compute the ridge estimator $\hat{\boldsymbol{\beta}}_{ridge}^{(-j)}$, where the j-th fold is not used.

3) Compute Equation 10.3 using the j-th folds as a test sample, i.e.,

$$\widehat{MSPE}_j = \frac{1}{n_j} \sum_{i=1}^{n_j} \left(Y_i^{(j)} - (\boldsymbol{X}_i^{(j)})' \hat{\boldsymbol{\beta}}_{ridge}^{(-j)} \right)^2$$

4) The m-fold cross validation estimator is the weighted average over the m subsample estimates of the MSPE:

$$\widehat{MSPE}_{mfold} = \sum_{j=1}^{m} \frac{n_j}{n} \widehat{MSPE}_j,$$

where $n = \sum_{j=1}^{m} n_j$ is the total number of observations.

5) Repeat these steps over a grid of tuning parameters for λ, and select the value for λ that minimizes $\widehat{MSPE}_{mfold}$.

Common values for m are $m = 5$ and $m = 10$. The larger m, the less biased the estimation of the MSPE is, but also the more computationally expensive the cross validation becomes.

The largest possible value for m is $m = n$, where each observation represents a fold. This is also known as leave-one-out cross validation (LOOCV). LOOCV might be useful for small datasets but is often infeasible for large dataset because of the large computation time.

10.9 L2 Regularization: Ridge

The ℓ_p-norm of a vector $\boldsymbol{a} = (a_1, \dots, a_k)'$ is defined as

$$\|\boldsymbol{a}\|_p = \left(\sum_{j=1}^{k} |a_j|^p \right)^{1/p}.$$

Important special cases are the ℓ_1-norm and ℓ_2-norm:

$$\|\boldsymbol{a}\|_1 = \sum_{j=1}^{k} |a_j|, \quad \|\boldsymbol{a}\|_2 = \left(\sum_{j=1}^{k} a_j^2 \right)^{1/2} = \sqrt{\boldsymbol{a}'\boldsymbol{a}}.$$

The ℓ_1-norm is the sum of absolute values, and the ℓ_2-norm, also known as the Euclidean norm, represents the length of the vector in the Euclidean space.

Ridge regression is also called **L2 regularization** because it penalizes the sum of squared errors by the square of the ℓ_2-norm of the coefficient vector, $\|\boldsymbol{\beta}\|_2^2 = \boldsymbol{\beta}'\boldsymbol{\beta}$. Ridge is the solution to the minimization problem Equation 10.2, which can be written as

$$\hat{\boldsymbol{\beta}}_{ridge} = \text{argmin}_{\boldsymbol{\beta}} (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_2^2.$$

10.10 L1 Regularization: Lasso

An alternative approach is **L1 regularization**, also known as **lasso**. The lasso estimator is defined as

$$\hat{\boldsymbol{\beta}}_{lasso} = \text{argmin}_{\boldsymbol{\beta}} (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_1,$$

where $\|\boldsymbol{\beta}\|_1 = \sum_{j=1}^{k} |\beta_j|$.

The **elastic net** estimator is a hybrid method. It combines L1 and L2 regularization using a weight $0 \leq \alpha \leq 1$:

$$\hat{\boldsymbol{\beta}}_{net,\alpha} = \text{argmin}_{\boldsymbol{\beta}} (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda(\alpha \|\boldsymbol{\beta}\|_1 + (1-\alpha)\|\boldsymbol{\beta}\|_2^2).$$

This includes ridge ($\alpha = 0$) and lasso ($\alpha = 1$) as special cases.

Ridge has a closed form solution given by Equation 10.1. Lasso and elastic net with $\alpha > 0$ require numerical solutions by means of quadratic programming. The solution typically involves some zero coefficients.

10.11 Implementation in R

Let's consider the `mtcars` dataset, which is available in base R. Have a look at `?mtcars` to see the data description.

We estimate a ridge regression model to predict the variable `mpg` (miles per gallon) using the other variables. We consider the values $\lambda = 0.5$ and $\lambda = 2.5$.

Ridge, lasso, and elastic net are implemented in the `glmnet` package. The `glmnet()` function requires matrix-valued data as input. The `model.matrix()` command is useful because it produces the regressor matrix $\boldsymbol{X}$ and converts categorical variables into dummy variables.

```
Y = mtcars$mpg
X = model.matrix(mpg ~., data = mtcars)[,-1]
## Number of observations n and regressors k:
dim(X)
```

```
[1] 32 10
```

```
fit.ridge1 = glmnet(x=X, y=Y, alpha=0, lambda = 0.5)
fit.ridge2 = glmnet(x=X, y=Y, alpha=0, lambda = 2.5)
fits = cbind(coef(fit.ridge1), coef(fit.ridge2))
colnames(fits) = c("lambda=0.5", "lambda=2.5")
fits
```

```
11 x 2 sparse Matrix of class "dgCMatrix"
               lambda=0.5   lambda=2.5
(Intercept) 19.420400249 21.179818696
cyl         -0.250698757 -0.368541841
disp        -0.001893223 -0.005184086
hp          -0.013079878 -0.011710951
drat         0.978514241  1.052837310
wt          -1.902328296 -1.264016952
qsec         0.316107066  0.164790158
vs           0.472551434  0.755205256
am           2.113922488  1.655241565
gear         0.631836101  0.546732963
carb        -0.661215998 -0.560023425
```

By default the regressors are standardized. Therefore the coefficients represent the marginal effects as the change in the response variable for a one standard deviation change in the

regressor. For instance, with $\lambda = 0.5$, the coefficient of `wt` (weight) is -1.9, which means that a one standard deviation increase in weight leads to a decrease of 1.9 miles per gallon.

When we exclude the intercept, the average coefficient size (with respect to the ℓ_2 norm) becomes small for larger values of λ:

```
c(
  sqrt(sum(coef(fit.ridge1)[-1]^2)),
  sqrt(sum(coef(fit.ridge2)[-1]^2))
)
```

```
[1] 3.204323 2.606156
```

The lasso estimator ($\alpha = 1$) sets many coefficients equal to zero:

```
fit.lasso1 = glmnet(x=X, y=Y, alpha=1, lambda = 0.5)
fit.lasso2 = glmnet(x=X, y=Y, alpha=1, lambda = 2.5)
fits = cbind(coef(fit.lasso1), coef(fit.lasso2))
colnames(fits) = c("lambda=0.5", "lambda=2.5")
fits
```

```
11 x 2 sparse Matrix of class "dgCMatrix"
             lambda=0.5 lambda=2.5
(Intercept) 35.88689755 30.0625817
cyl         -0.85565434 -0.7090799
disp         .           .
hp          -0.01411517  .
drat         0.07603453  .
wt          -2.67338139 -1.7358069
qsec         .           .
vs           .           .
am           0.48651385  .
gear         .           .
carb        -0.10722338  .
```

The `cv.glmnet()` command estimates the optimal shrinkage parameter using 10-fold cross validation:

```
set.seed(123) ## for reproducibility
cv.glmnet(x=X, y=Y, alpha = 0)$lambda.min
```

```
[1] 2.746789
```

```
cv.glmnet(x=X, y=Y, alpha = 1)$lambda.min
```

```
[1] 0.8007036
```

We can use ridge and lasso to estimate linear models with more variables than observations. The command `^2` includes all pairwise interaction terms, which produces 55 variables in total. The dataset has $n = 32$ observations.

```
X.large = model.matrix(mpg ~. ^2, data = mtcars)[,-1]
dim(X.large) # more regressors than observations
```

```
[1] 32 55
```

```
fit.ridgelarge = glmnet(x=X.large, y=Y, alpha=0, lambda = 0.5)
fit.lassolarge = glmnet(x=X.large, y=Y, alpha=1, lambda = 0.5)
fits = cbind(
  coef(fit.ridgelarge), coef(fit.lassolarge)
)
colnames(fits) = c("ridge", "lasso")
fits
```

```
56 x 2 sparse Matrix of class "dgCMatrix"
                    ridge        lasso
(Intercept)  1.315259e+01 23.655330629
cyl         -4.061218e-02 -0.036308043
disp        -8.137358e-04  .
hp          -5.588290e-03  .
drat         4.386174e-01  .
wt          -5.547986e-01 -1.301739306
qsec         2.308772e-01  .
vs           6.705889e-01  .
am           4.379822e-01  .
gear         8.788479e-01  .
carb        -1.537294e-01  .
cyl:disp     6.830897e-05  .
cyl:hp       1.351742e-04  .
cyl:drat     2.455464e-02  .
cyl:wt      -2.621868e-03  .
```

```
cyl:qsec     3.358094e-03  .
cyl:vs       1.591177e-01  .
cyl:am       6.102385e-02  .
cyl:gear     3.481957e-02  .
cyl:carb     7.499023e-04  .
disp:hp      8.592521e-06  .
disp:drat   -9.421536e-05  .
disp:wt      2.191122e-04  .
disp:qsec   -1.789464e-05  .
disp:vs     -1.280463e-03  .
disp:am     -9.043597e-03  .
disp:gear   -3.601317e-04  .
disp:carb   -1.255358e-04  .
hp:drat     -2.086003e-03  .
hp:wt        4.404097e-04  .
hp:qsec     -4.347470e-04 -0.001328046
hp:vs       -1.858343e-02  .
hp:am       -2.604620e-03  .
hp:gear     -3.464491e-04  .
hp:carb      9.107116e-04  .
drat:wt     -1.766081e-01 -0.337667877
drat:qsec    3.828881e-02  0.073725291
drat:vs      1.123963e-01  .
drat:am      5.047132e-02  .
drat:gear    8.294201e-02  .
drat:carb   -4.770358e-02  .
wt:qsec     -3.289204e-02  .
wt:vs       -3.239643e-01  .
wt:am       -4.197733e-01  .
wt:gear     -1.890703e-01  .
wt:carb     -1.497574e-02  .
qsec:vs      3.114409e-02  .
qsec:am      5.199239e-02  .
qsec:gear    7.035311e-02  0.041623415
qsec:carb   -1.859676e-02  .
vs:am        8.688134e-01  2.429571498
vs:gear      3.311330e-01  .
vs:carb     -2.768199e-01  .
am:gear      1.462749e-01  .
am:carb      1.588431e-01  .
gear:carb    8.165764e-03  .
```

To get the fitted values you may use the `predict()` command:

```
Yhatridge = predict(fit.ridgelarge, newx = X.large)
Yhatlasso = predict(fit.lassolarge, newx = X.large)
Yhats = cbind(Y, Yhatridge, Yhatlasso)
colnames(Yhats) = c("Y", "Yhat-ridge", "Yhat-lasso")
Yhats
```

```
                     Y Yhat-ridge Yhat-lasso
Mazda RX4         21.0   20.94312   21.64528
Mazda RX4 Wag     21.0   20.47797   21.14997
Datsun 710        22.8   26.12112   25.98585
Hornet 4 Drive    21.4   19.57785   19.91064
Hornet Sportabout 18.7   17.25059   17.35026
Valiant           18.1   19.25815   19.52858
Duster 360        14.3   14.80168   15.42082
Merc 240D         24.4   23.06386   22.50685
Merc 230          22.8   23.69586   22.78181
Merc 280          19.2   18.47341   19.75241
Merc 280C         17.8   18.75521   19.92770
Merc 450SE        16.4   15.39830   15.79922
Merc 450SL        17.3   16.19856   16.61670
Merc 450SLC       15.2   16.21931   16.54465
Cadillac Fleetwood 10.4  12.25717   12.57063
Lincoln Continental 10.4 11.74625   11.88810
Chrysler Imperial 14.7   11.64161   11.58002
Fiat 128          32.4   28.79845   27.43656
Honda Civic       30.4   31.07410   29.68475
Toyota Corolla    33.9   30.63399   28.72288
Toyota Corona     21.5   22.35048   22.60097
Dodge Challenger  15.5   17.17402   17.68091
AMC Javelin       15.2   17.70056   17.97138
Camaro Z28        13.3   14.14050   14.67766
Pontiac Firebird  19.2   16.37763   16.39890
Fiat X1-9         27.3   29.32240   27.93021
Porsche 914-2     26.0   26.15812   24.43481
Lotus Europa      30.4   28.93150   27.72235
Ford Pantera L    15.8   16.69717   17.16642
Ferrari Dino      19.7   20.27929   20.20595
Maserati Bora     15.0   14.07394   14.80373
Volvo 142E        21.4   23.30782   24.50302
```

10.12 R-codes

metrics-sec10.R

11 Principal Components

If two regressors are highly correlated, we can typically drop one of the regressors because it mostly contains the same information.

The idea of principal component regression is to exploit the correlations among the regressors to reduce their number while retaining as much of the original information as possible.

11.1 Principal Components

The principal components (PC) are linear combinations of the regressor variables that capture as much of the variation in the original variables as possible.

Principal Components

Let $\boldsymbol{X}_i$ be a k-variate vector of regressor variables.

The **first principal component** is $P_{i1} = \boldsymbol{w}_1' \boldsymbol{X}_i$, where $\boldsymbol{w}_1$ satisfies

$$\boldsymbol{w}_1 = \text{argmax}_{\boldsymbol{w}'\boldsymbol{w}=1} \; Var[\boldsymbol{w}'\boldsymbol{X}_i]$$

The **second principal component** is $P_{i2} = \boldsymbol{w}_2' \boldsymbol{X}_i$, where $\boldsymbol{w}_2$ satisfies

$$\boldsymbol{w}_2 = \text{argmax}_{\substack{\boldsymbol{w}'\boldsymbol{w}=1 \\ \boldsymbol{w}'\boldsymbol{w}_1=0}} \; Var[\boldsymbol{w}'\boldsymbol{X}_i]$$

The l**-th principal component** is $P_{il} = \boldsymbol{w}_l' \boldsymbol{X}_i$, where $\boldsymbol{w}_l$ satisfies

$$\boldsymbol{w}_l = \text{argmax}_{\substack{\boldsymbol{w}'\boldsymbol{w}=1 \\ \boldsymbol{w}'\boldsymbol{w}_1=\ldots=\boldsymbol{w}'\boldsymbol{w}_{l-1}=0}} Var[\boldsymbol{w}'\boldsymbol{X}_i]$$

A k-variate regressor vector $\boldsymbol{X}_i$ has k principal components $P_{i1}, \ldots, P_{ik}$ and k corresponding **weights** or **principal component loadings** $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_k$.

By definition, the principal components are descendingly ordered by their variance:

$$Var[P_{i1}] \geq Var[P_{i2}] \geq \ldots \geq Var[P_{ik}] \geq 0$$

The principal component weights are orthonormal:

$$\boldsymbol{w}_i' \boldsymbol{w}_j = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

Moreover, $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_k$ form an orthonormal basis for the k-dimensional vector space $\mathbb{R}^k$. The regressor vector admits the following decomposition into its principal components:

$$\boldsymbol{X}_i = \sum_{l=1}^{k} P_{il} \boldsymbol{w}_l \tag{11.1}$$

The decomposition of a dataset into its principal components is called **principal component analysis (PCA)**.

11.2 Analytical PCA Solution

In this subsection, we will use some matrix calculus and eigenvalue theory. To recap the relevant matrix algebra, the following resources will be useful:

- Eigenvalues and Eigenvectors: https://matrix.svenotto.com/04_furtherconcepts.html
- Derivative rules for vectors: https://matrix.svenotto.com/05_calculus.html

The maximization problem for the first principal component is

$$\max_{\boldsymbol{w}} \; Var[\boldsymbol{w}'\boldsymbol{X}_i] \quad \text{subject to } \boldsymbol{w}'\boldsymbol{w} = 1. \tag{11.2}$$

The variance of interest can be rewritten as

$$\begin{aligned} Var[\boldsymbol{w}'\boldsymbol{X}_i] &= E[(\boldsymbol{w}'(\boldsymbol{X}_i - E[\boldsymbol{X}_i]))^2] \\ &= E[(\boldsymbol{w}'(\boldsymbol{X}_i - E[\boldsymbol{X}_i]))((\boldsymbol{X}_i - E[\boldsymbol{X}_i])'\boldsymbol{w})] \\ &= \boldsymbol{w}' E[(\boldsymbol{X}_i - E[\boldsymbol{X}_i])(\boldsymbol{X}_i - E[\boldsymbol{X}_i])']\boldsymbol{w} \\ &= \boldsymbol{w}'\Sigma\boldsymbol{w} \end{aligned}$$

where $\Sigma = Var[\boldsymbol{X}_i]$ is the population covariance matrix of $\boldsymbol{X}_i$. Thus, the constrained maximization problem Equation 11.2 has the Lagrangian

$$\mathcal{L}(\boldsymbol{w}, \lambda) = \boldsymbol{w}'\Sigma\boldsymbol{w} - \lambda(\boldsymbol{w}'\boldsymbol{w} - 1),$$

where λ is a Lagrange multiplier.

Recall the derivative rules for vectors: If $\boldsymbol{A}$ is a symmetric matrix, then the derivative of $\boldsymbol{a}'\boldsymbol{A}\boldsymbol{a}$ with respect to $\boldsymbol{a}$ is $2\boldsymbol{A}\boldsymbol{a}$. Therefore, the first order condition with respect to $\boldsymbol{w}$ is

$$\Sigma\boldsymbol{w} = \lambda\boldsymbol{w}. \tag{11.3}$$

The pair $(\lambda, \boldsymbol{w})$ must satisfy the eigenequation Equation 11.3, which is precisely the eigenequation which defines an eigenvalue-eigenvector pair. The Lagrange multiplier λ must be an eigenvalue of Σ and the weight vector $\boldsymbol{w}$ must be a corresponding eigenvector.

By the first order condition with respect to λ,

$$\boldsymbol{w}'\boldsymbol{w} = 1,$$

the eigenvector is normalized to length 1.

Therefore, the variance of interest is

$$Var[\boldsymbol{w}'\boldsymbol{X}_i] = \boldsymbol{w}'\Sigma\boldsymbol{w} = \boldsymbol{w}'(\lambda\boldsymbol{w}) = \lambda. \tag{11.4}$$

Consequently, $Var[\boldsymbol{w}'\boldsymbol{X}_i]$ must be an eigenvalue of Σ and $\boldsymbol{w}$ is a corresponding normalized eigenvector.

The expression $Var[\boldsymbol{w}'\boldsymbol{X}_i] = \lambda$ is maximized if we use the largest eigenvalue $\lambda = \lambda_1$. Consequently, the variance of the first principal component P_{i1} is equal to the largest eigenvalue λ_1 of Σ, and the first principal component weight $\boldsymbol{w}_1$ is a normalized eigenvector corresponding to the eigenvalue λ_1.

Analogously, the second principal component weight $\boldsymbol{w}_2$ must also be a normalized eigenvector of Σ with the additional restriction that it is orthogonal to $\boldsymbol{w}_1$. Therefore, it cannot be an eigenvector corresponding to the first eigenvalue, and we use the second largest eigenvalue $\lambda = \lambda_2$ to maximize Equation 11.4.

The variance of the second principal component P_{i2} is equal to the second largest eigenvalue λ_2 of Σ, and the second principal component weight $\boldsymbol{w}_2$ is a corresponding normalized eigenvector.

To continue this pattern, the variance of the l-th principal component P_{il} is equal to the l-th largest eigenvalue λ_l of Σ, and the l-th principal component weight $\boldsymbol{w}_l$ is a corresponding normalized eigenvector.

Principal Components Solution

Let Σ be the covariance matrix of the k-variate vector of regressor variables $\boldsymbol{X}_i$, let $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_k \geq 0$ be the eigenvalues ordered in descending order of Σ, and let $\boldsymbol{v}_1, ..., \boldsymbol{v}_k$ be corresponding orthonormal eigenvectors.

- The principal component weights are $\boldsymbol{w}_l = \boldsymbol{v}_l$ for $l = 1, ..., k$
- The principal components are $P_{il} = \boldsymbol{v}_l'\boldsymbol{X}_i$, and they have the properties

$$Var[P_{il}] = \lambda_l, \quad Cov(P_{il}, P_{im}) = 0, \quad l \neq m.$$

Principal components are uncorrelated because

$$\begin{aligned} Cov(P_{im}, P_{il}) &= E[\boldsymbol{w}_m'(\boldsymbol{X}_i - E[\boldsymbol{X}_i])(\boldsymbol{X}_i - E[\boldsymbol{X}_i])'\boldsymbol{w}_l] \\ &= \boldsymbol{w}_m'\Sigma\boldsymbol{w}_l = \lambda_m\boldsymbol{w}_m'\boldsymbol{w}_l, \end{aligned}$$

where $\boldsymbol{w}_m'\boldsymbol{w}_l = 1$ if $m = l$ and $\boldsymbol{w}_m'\boldsymbol{w}_l = 0$ if $m \neq l$

11.3 Sample principal components

The covariance matrix $\Sigma = Var[\boldsymbol{X}_i]$ is unknown in practice. Instead, we estimate it from the sample $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$:

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{n-1}\sum_{i=1}^{n}(\boldsymbol{X}_i - \overline{\boldsymbol{X}})(\boldsymbol{X}_i - \overline{\boldsymbol{X}})'.$$

Let $\hat{\lambda}_1 \geq \hat{\lambda}_2 \geq \ldots, \hat{\lambda}_k \geq 0$ be the eigenvalues of $\widehat{\boldsymbol{\Sigma}}$ and let $\hat{\boldsymbol{v}}_1, \ldots, \hat{\boldsymbol{v}}_k$ be corresponding orthonormal eigenvectors. Then,

- The l-th sample principal component for observation i is

$$\widehat{P}_{il} = \widehat{\boldsymbol{w}}_l' \boldsymbol{X}_i$$

- The l-th sample principal component weight vector is

$$\widehat{\boldsymbol{w}}_l = \hat{\boldsymbol{v}}_l$$

- The (adjusted) sample variance of the l-th sample principal components series $\widehat{P}_{1l}, \ldots, \widehat{P}_{nl}$ is $\hat{\lambda}_l$, and the sample covariances of different principal components series are zero.

11.4 PCA in R

Let's compute the sample principal components of the `mtcars` dataset:

```
pca = prcomp(mtcars)
## the principal components are arranged by columns
## first few rows of principal components:
pca$x |> head()
```

```
                        PC1       PC2       PC3        PC4        PC5
Mazda RX4         -79.596425  2.132241 -2.153336 -2.7073437 -0.7023522
Mazda RX4 Wag     -79.598570  2.147487 -2.215124 -2.1782888 -0.8843859
Datsun 710       -133.894096 -5.057570 -2.137950  0.3460330  1.1061111
Hornet 4 Drive      8.516559 44.985630  1.233763  0.8273631  0.4240145
Hornet Sportabout 128.686342 30.817402  3.343421 -0.5211000  0.7365801
Valiant           -23.220146 35.106518 -3.259562  1.4005360  0.8029768
                         PC6          PC7         PC8        PC9         PC10
Mazda RX4        -0.31486106 -0.098695018  0.07789812 -0.2000092 -0.29008191
Mazda RX4 Wag    -0.45343873 -0.003554594  0.09566630 -0.3533243 -0.19283553
Datsun 710        1.17298584  0.005755581 -0.13624782 -0.1976423  0.07634353
```

```
Hornet 4 Drive    -0.05789705 -0.024307168 -0.22120800  0.3559844 -0.09057039
Hornet Sportabout -0.33290957  0.106304777  0.05301719  0.1532714 -0.18862217
Valiant           -0.08837864  0.238946304 -0.42390551  0.1012944 -0.03769010
                        PC11
Mazda RX4         -0.1057706
Mazda RX4 Wag     -0.1069047
Datsun 710        -0.2668713
Hornet 4 Drive    -0.2088354
Hornet Sportabout  0.1092563
Valiant           -0.2757693
```

```
## the principal components weights
pca$rotation |> head()
```

```
              PC1          PC2         PC3          PC4         PC5
mpg  -0.038118199  0.009184847  0.98207085  0.047634784 -0.08832843
cyl   0.012035150 -0.003372487 -0.06348394 -0.227991962  0.23872590
disp  0.899568146  0.435372320  0.03144266 -0.005086826 -0.01073597
hp    0.434784387 -0.899307303  0.02509305  0.035715638  0.01655194
drat -0.002660077 -0.003900205  0.03972493 -0.057129357 -0.13332765
wt    0.006239405  0.004861023 -0.08491026  0.127962867 -0.24354296
              PC6          PC7           PC8          PC9         PC10
mpg  -0.143790084 -0.039239174 -2.271040e-02 -0.002790139  0.030630361
cyl  -0.793818050  0.425011021  1.890403e-01  0.042677206  0.131718534
disp  0.007424138  0.000582398  5.841464e-04  0.003532713 -0.005399132
hp    0.001653685 -0.002212538 -4.748087e-06 -0.003734085  0.001862554
drat  0.227229260  0.034847411  9.385817e-01 -0.014131110  0.184102094
wt   -0.127142296 -0.186558915 -1.561907e-01 -0.390600261  0.829886844
              PC11
mpg   0.0158569365
cyl  -0.1454453628
disp -0.0009420262
hp    0.0021526102
drat  0.0973818815
wt    0.0198581635
```

```
## the standard deviations of the principal components
## are the square roots of the sample eigenvalues
pca$sdev
```

```
 [1] 136.5330479  38.1480776   3.0710166   1.3066508   0.9064862   0.6635411
 [7]   0.3085791   0.2859604   0.2506973   0.2106519   0.1984238
```

Principal components are sensitive to the scaling of the data. Consequently, it is recommended to first scale each variable in the dataset to have mean zero and unit variance: `scale(mtcars)`. In this case, Σ is the correlation matrix.

```
pca = mtcars |> scale() |> prcomp()
pca$x |> head()
```

```
                          PC1        PC2        PC3         PC4        PC5
Mazda RX4         -0.64686274  1.7081142 -0.5917309  0.11370221  0.9455234
Mazda RX4 Wag     -0.61948315  1.5256219 -0.3763013  0.19912121  1.0166807
Datsun 710        -2.73562427 -0.1441501 -0.2374391 -0.24521545 -0.3987623
Hornet 4 Drive    -0.30686063 -2.3258038 -0.1336213 -0.50380035 -0.5492089
Hornet Sportabout  1.94339268 -0.7425211 -1.1165366  0.07446196 -0.2075157
Valiant           -0.05525342 -2.7421229  0.1612456 -0.97516743 -0.2116654
                          PC6         PC7          PC8         PC9         PC10
Mazda RX4         -0.01698737 -0.42648652  0.009631217 -0.14642303  0.06670350
Mazda RX4 Wag     -0.24172464 -0.41620046  0.084520213 -0.07452829  0.12692766
Datsun 710        -0.34876781 -0.60884146 -0.585255765  0.13122859 -0.04573787
Hornet 4 Drive     0.01929700 -0.04036075  0.049583029 -0.22021812  0.06039981
Hornet Sportabout  0.14919276  0.38350816  0.160297757  0.02117623  0.05983003
Valiant           -0.24383585 -0.29464160 -0.256612420  0.03222907  0.20165466
                         PC11
Mazda RX4         0.17969357
Mazda RX4 Wag     0.08864426
Datsun 710        -0.09463291
Hornet 4 Drive    0.14761127
Hornet Sportabout 0.14640690
Valiant           0.01954506
```

```
pca$rotation |> head()
```

```
            PC1         PC2         PC3          PC4         PC5         PC6
mpg  -0.3625305  0.01612440 -0.22574419 -0.022540255 -0.10284468 -0.10879743
cyl   0.3739160  0.04374371 -0.17531118 -0.002591838 -0.05848381  0.16855369
disp  0.3681852 -0.04932413 -0.06148414  0.256607885 -0.39399530 -0.33616451
hp    0.3300569  0.24878402  0.14001476 -0.067676157 -0.54004744  0.07143563
drat -0.2941514  0.27469408  0.16118879  0.854828743 -0.07732727  0.24449705
wt    0.3461033 -0.14303825  0.34181851  0.245899314  0.07502912 -0.46493964
              PC7          PC8         PC9        PC10        PC11
mpg   0.367723810  0.754091423 -0.23570162 -0.13928524 -0.12489563
cyl   0.057277736  0.230824925 -0.05403527  0.84641949 -0.14069544
```

```
disp  0.214303077 -0.001142134 -0.19842785 -0.04937979  0.66060648
hp   -0.001495989  0.222358441  0.57583007 -0.24782351 -0.25649206
drat  0.021119857 -0.032193501  0.04690123  0.10149369 -0.03953025
wt   -0.020668302  0.008571929 -0.35949825 -0.09439426 -0.56744870
```

```
pca$sdev
```

```
 [1] 2.5706809 1.6280258 0.7919579 0.5192277 0.4727061 0.4599958 0.3677798
 [8] 0.3505730 0.2775728 0.2281128 0.1484736
```

11.5 Variance of principal components

Since the sample principal components are uncorrelated, the total variation in the data is

$$Var\left[\sum_{m=1}^{k}\widehat{P}_{im}\right] = \sum_{m=1}^{k} Var[\widehat{P}_{im}] = \sum_{m=1}^{k}\hat{\lambda}_l.$$

The proportion of variance explained by the l-th principal component is

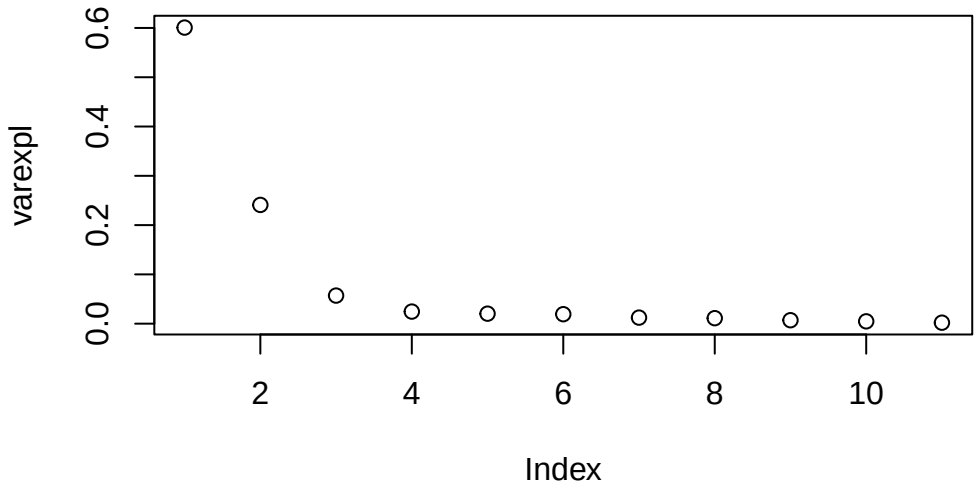
$$\frac{Var[\widehat{P}_{il}]}{Var[\sum_{m=1}^{k}\widehat{P}_{im}]} = \frac{\hat{\lambda}_l}{\sum_{m=1}^{k}\hat{\lambda}_m}$$

A scree plot is useful to see how much each principal component contributes to the total variation:

```
pcvar = pca$sdev^2
varexpl = pcvar/sum(pcvar)
varexpl
```

```
 [1] 0.600763659 0.240951627 0.057017934 0.024508858 0.020313737 0.019236011
 [7] 0.012296544 0.011172858 0.007004241 0.004730495 0.002004037
```

```
plot(varexpl)
```




```
cumsum(varexpl)
```

```
 [1] 0.6007637 0.8417153 0.8987332 0.9232421 0.9435558 0.9627918 0.9750884
 [8] 0.9862612 0.9932655 0.9979960 1.0000000
```

The first principal component explains more than 60% of the variation, the first four explain more than 90% of the variation, the first 6 more than 95%, and the first 9 principal components more than 99% of the variation.

11.6 Linear regression with principal components

Principal components can be used to estimate the high-dimensional (large k) linear regression model

$$Y_i = \boldsymbol{X}_i'\boldsymbol{\beta} + u_i, \quad i = 1, \dots, n.$$

While ridge and lasso shrink coefficients to prevent overfitting, PCA reduces dimensionality by transforming variables into orthogonal components before estimation.

Since the principal component weights $\boldsymbol{w}_1, \dots, \boldsymbol{w}_k$ form a basis of $\mathbb{R}^k$, the regressors have the basis representation given by Equation 11.1. Similarly, we can represent the coefficient vector in terms of the principal component basis:

$$\boldsymbol{\beta} = \sum_{l=1}^{k} \theta_l \boldsymbol{w}_l, \quad \theta_l = \boldsymbol{w}_l'\boldsymbol{\beta}. \tag{11.5}$$

Inserting in the regression function gives

$$\boldsymbol{X}_i'\boldsymbol{\beta} = \sum_{l=1}^{k} \underbrace{\boldsymbol{X}_i'\boldsymbol{w}_l}_{=P_{il}} \theta_l,$$

and the regression equation becomes

$$Y_i = \sum_{l=1}^{k} P_{il}\theta_l + u_i. \tag{11.6}$$

This regression equation is convenient because the regressors P_{il} are uncorrelated, and OLS estimates for θ_l can be inserted back into Equation 11.5 to get an estimate for $\boldsymbol{\beta}$.

When k is large, this approach is still prone to overfitting. The k principal components of $\boldsymbol{X}_i$ explain 100% of its variance, but it may be reasonable to select a smaller number of principal components $p < k$ that explain 95% or 99% of the variance.

The remaining $k - p$ principal components explain only 5% or 1% of the variance. The idea is that we truncate the model by assuming that the remaining principal components contain only noise that is uncorrelated with Y_i.

Assumption (PC): $E[P_{im}Y_i] = 0$ for all $m = p + 1, \dots, k$.

This assumption implies that the components with indices larger than p contribute no systematic predictive power for Y_i, and hence only introduce noise.

Because the principal components are uncorrelated, we have $\theta_l = E[Y_iP_{il}]/E[P_{il}^2]$, and, therefore $\theta_m = 0$ for $m = p + 1, \dots, k$. Consequently,

$$\boldsymbol{\beta} = \sum_{l=1}^{p} \theta_l \boldsymbol{w}_l, \tag{11.7}$$

and Equation 11.6 becomes a factor model with p factors:

$$Y_i = \sum_{l=1}^{p} \theta_l P_{il} + u_i = \boldsymbol{P}_i'\boldsymbol{\theta} + u_i,$$

where $\boldsymbol{P}_i = (P_{i1}, \dots, P_{ip})'$ and $\boldsymbol{\theta} = (\theta_1, \dots, \theta_p)'$. The least squares estimator of $\boldsymbol{\theta}$ using the regressors $\boldsymbol{P}_i$, $i = 1, \dots n$ can then be inserted to Equation 11.7 to obtain an estimate for $\boldsymbol{\beta}$.

In practice, the principal components are unknown and must be replaced by the first p sample principal components

$$\widehat{\boldsymbol{P}}_i = (\widehat{P}_{i1}, \dots, \widehat{P}_{ip})', \quad \widehat{P}_{il} = \widehat{\boldsymbol{w}}_l'\boldsymbol{X}_i.$$

The feasible least squares estimator for θ is

$$\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \dots, \hat{\theta}_p)' = \left(\sum_{i=1}^{n} \widehat{\boldsymbol{P}}_i \widehat{\boldsymbol{P}}_i' \right)^{-1} \sum_{i=1}^{n} \widehat{\boldsymbol{P}}_i Y_i,$$

and the principal components estimator for $\boldsymbol{\beta}$ is

$$\hat{\boldsymbol{\beta}}_{pc} = \sum_{l=1}^{p} \hat{\theta}_l \widehat{\boldsymbol{w}}_l.$$

11.7 Selecting the number of factors

To select the number of principal components, one practical approach is to choose those that explain a pre-specified percentage (90-99%) of the total variance.

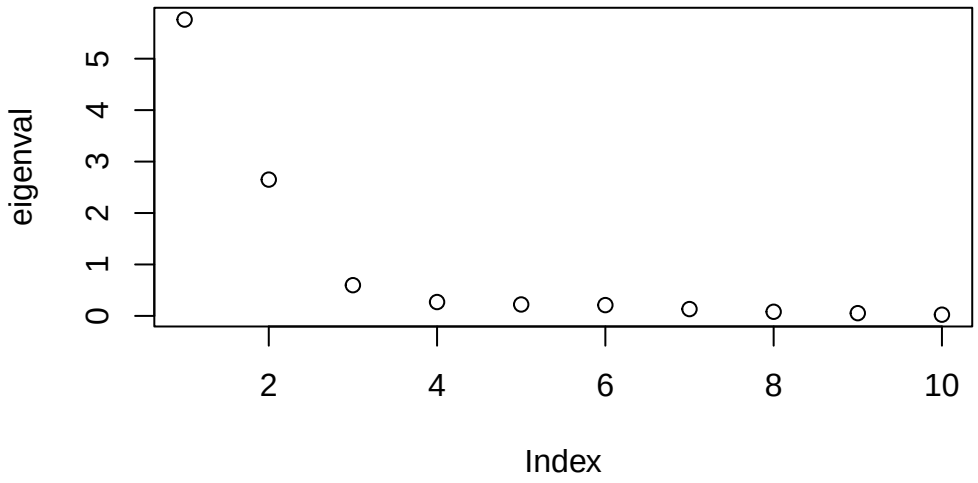
```
Y = mtcars$mpg
X = model.matrix(mpg ~., data = mtcars)[,-1] |> scale()
## principal component analysis
pca = prcomp(X)
P = pca$x #full matrix of all principal components
## variance explained
eigenval = pca$sdev^2
varexpl = eigenval/sum(eigenval)
cumsum(varexpl)
```

```
 [1] 0.5760217 0.8409861 0.9007075 0.9276582 0.9498832 0.9708950 0.9841870
 [8] 0.9922551 0.9976204 1.0000000
```

The first four principal components explain more than 92% of the variance, and the first seven more than 98%.

Another method involves creating a scree plot to display the eigenvalues (variances) for each principal component and identifying the point where the eigenvalues sharply drop (elbow point).

```
plot(eigenval)
```



We find an elbow at four principal components.

Selecting the number of principal components, similar to shrinkage estimation, involves balancing variance and bias. If the Assumption (PC) holds, the PC estimator is unbiased; if it doesn't, a small bias is introduced. Increasing the number of components p reduces bias but increases variance, while decreasing p reduces variance but increases bias.

Similarly to the shrinkage parameter in ridge and lasso estimation, the number of factors p can be treated as a tuning parameter. We can use m-fold cross validation to select p such that the MSE is minimized.

The `caret` package in R provides a convenient way to perform cross-validation and select the optimal number of principal components.

```
set.seed(111)
## PCR 10-fold cross-validation
library(caret)
```

```
Lade nötiges Paket: ggplot2
```

```
Lade nötiges Paket: lattice
```

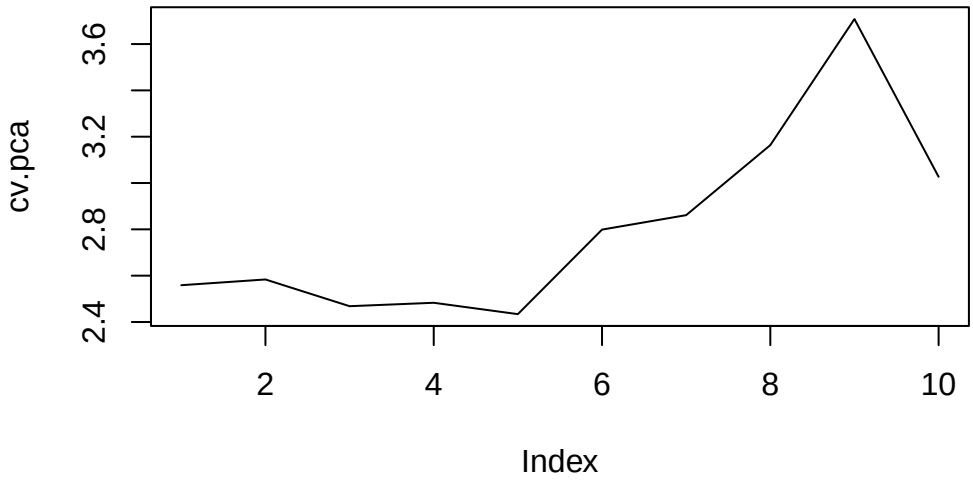
```
myfunc.cvpca = function(p){
  data_pca = data.frame(Y, P[,1:p])
  cv = train(
    Y ~ ., data = data_pca,
    method = "lm",
    metric = "RMSE",
    trControl = trainControl(method = "cv", number = 10)
```

```
  )
  return(cv$results$RMSE)
}
# Iterate function crossval over ncomp = 1, ..., maxcomp
maxcomp = 10 # select not more than number of variables (for data_small select <=4)
cv.pca = sapply(1:maxcomp, myfunc.cvpca) # sapply is useful for iterating over function argu

# Find the number of components with the lowest RMSPE
which.min(cv.pca)
```

```
[1] 5
```

```
plot(cv.pca, type="l")
```



The 10-fold cross validation method suggests to use 5 principal components.

11.8 R-codes

metrics-sec11.R