Gareth James • Daniela Witten • Trevor Hastie
Robert Tibshirani • Jonathan Taylor

An Introduction to Statistical Learning

with Applications in Python

First Printing: July 5, 2023

To our parents:

Alison and Michael James

Chiara Nappi and Edward Witten

Valerie and Patrick Hastie

Vera and Sami Tibshirani

John and Brenda Taylor

and to our families:

Michael, Daniel, and Catherine

Tessa, Theo, Otto, and Ari

Samantha, Timothy, and Lynda

Charlie, Ryan, Julie, and Cheryl

Lee-Ann and Isobel

Preface

Statistical learning refers to a set of tools for *making sense of complex datasets*. In recent years, we have seen a staggering increase in the scale and scope of data collection across virtually all areas of science and industry. As a result, statistical learning has become a critical toolkit for anyone who wishes to understand data — and as more and more of today's jobs involve data, this means that statistical learning is fast becoming a critical toolkit for *everyone*.

One of the first books on statistical learning — *The Elements of Statistical Learning* (ESL, by Hastie, Tibshirani, and Friedman) — was published in 2001, with a second edition in 2009. ESL has become a popular text not only in statistics but also in related fields. One of the reasons for ESL's popularity is its relatively accessible style. But ESL is best-suited for individuals with advanced training in the mathematical sciences.

An Introduction to Statistical Learning, With Applications in R (ISLR) — first published in 2013, with a second edition in 2021 — arose from the clear need for a broader and less technical treatment of the key topics in statistical learning. In addition to a review of linear regression, ISLR covers many of today's most important statistical and machine learning approaches, including resampling, sparse methods for classification and regression, generalized additive models, tree-based methods, support vector machines, deep learning, survival analysis, clustering, and multiple testing.

Since it was published in 2013, ISLR has become a mainstay of undergraduate and graduate classrooms worldwide, as well as an important reference book for data scientists. One of the keys to its success has been that, beginning with Chapter 2, each chapter contains an `R` lab illustrating how to implement the statistical learning methods seen in that chapter, providing the reader with valuable hands-on experience.

However, in recent years `Python` has become an increasingly popular language for data science, and there has been increasing demand for a `Python`-

based alternative to ISLR. Hence, this book, *An Introduction to Statistical Learning, With Applications in Python* (ISLP), covers the same materials as ISLR but with labs implemented in `Python` — a feat accomplished by the addition of a new co-author, Jonathan Taylor. Several of the labs make use of the `ISLP` `Python` package, which we have written to facilitate carrying out the statistical learning methods covered in each chapter in `Python`. These labs will be useful both for `Python` novices, as well as experienced users.

The intention behind ISLP (and ISLR) is to concentrate more on the applications of the methods and less on the mathematical details, so it is appropriate for advanced undergraduates or master's students in statistics or related quantitative fields, or for individuals in other disciplines who wish to use statistical learning tools to analyze their data. It can be used as a textbook for a course spanning two semesters.

We are grateful to these readers for providing valuable comments on the first edition of ISLR: Pallavi Basu, Alexandra Chouldechova, Patrick Danaher, Will Fithian, Luella Fu, Sam Gross, Max Grazier G'Sell, Courtney Paulson, Xinghao Qiao, Elisa Sheng, Noah Simon, Kean Ming Tan, Xin Lu Tan. We thank these readers for helpful input on the second edition of ISLR: Alan Agresti, Iain Carmichael, Yiqun Chen, Erin Craig, Daisy Ding, Lucy Gao, Ismael Lemhadri, Bryan Martin, Anna Neufeld, Geoff Tims, Carsten Voelkmann, Steve Yadlowsky, and James Zou. We are immensely grateful to Balasubramanian "Naras" Narasimhan for his assistance on both ISLR and ISLP.

It has been an honor and a privilege for us to see the considerable impact that ISLR has had on the way in which statistical learning is practiced, both in and out of the academic setting. We hope that this new `Python` edition will continue to give today's and tomorrow's applied statisticians and data scientists the tools they need for success in a data-driven world.

It's tough to make predictions, especially about the future.

-Yogi Berra

Contents

1
Introduction

An Overview of Statistical Learning

Statistical learning refers to a vast set of tools for *understanding data*. These tools can be classified as *supervised* or *unsupervised*. Broadly speaking, supervised statistical learning involves building a statistical model for predicting, or estimating, an *output* based on one or more *inputs*. Problems of this nature occur in fields as diverse as business, medicine, astrophysics, and public policy. With unsupervised statistical learning, there are inputs but no supervising output; nevertheless we can learn relationships and structure from such data. To provide an illustration of some applications of statistical learning, we briefly discuss three real-world data sets that are considered in this book.

Wage Data

In this application (which we refer to as the `Wage` data set throughout this book), we examine a number of factors that relate to wages for a group of men from the Atlantic region of the United States. In particular, we wish to understand the association between an employee's `age` and `education`, as well as the calendar `year`, on his `wage`. Consider, for example, the left-hand panel of Figure 1.1, which displays `wage` versus `age` for each of the individuals in the data set. There is evidence that `wage` increases with `age` but then decreases again after approximately age 60. The blue line, which provides an estimate of the average `wage` for a given `age`, makes this trend clearer. Given an employee's `age`, we can use this curve to *predict* his `wage`. However, it is also clear from Figure 1.1 that there is a significant amount of variability associated with this average value, and so `age` alone is unlikely to provide an accurate prediction of a particular man's `wage`.

G. James et al., *An Introduction to Statistical Learning*, Springer Texts in Statistics,
https://doi.org/10.1007/978-3-031-38747-0_1

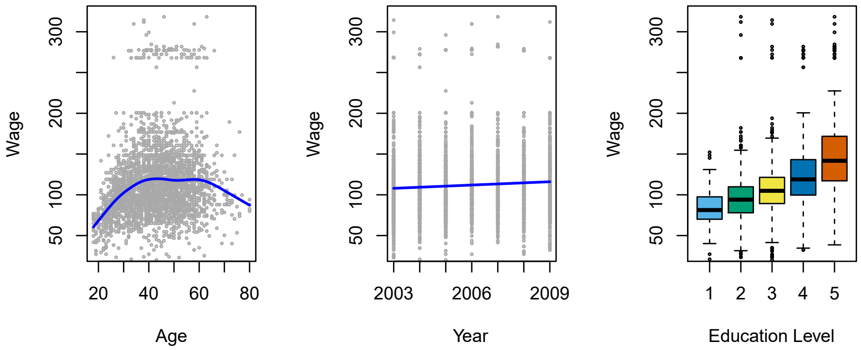

FIGURE 1.1. `Wage` *data, which contains income survey information for men from the central Atlantic region of the United States.* Left: `wage` *as a function of* `age`*. On average,* `wage` *increases with* `age` *until about* 60 *years of age, at which point it begins to decline.* Center: `wage` *as a function of* `year`*. There is a slow but steady increase of approximately* $10,000 *in the average* `wage` *between* 2003 *and* 2009*.* Right: *Boxplots displaying* `wage` *as a function of* `education`*, with* 1 *indicating the lowest level (no high school diploma) and* 5 *the highest level (an advanced graduate degree). On average,* `wage` *increases with the level of education.*

We also have information regarding each employee's education level and the `year` in which the `wage` was earned. The center and right-hand panels of Figure 1.1, which display `wage` as a function of both `year` and `education`, indicate that both of these factors are associated with `wage`. Wages increase by approximately $10,000, in a roughly linear (or straight-line) fashion, between 2003 and 2009, though this rise is very slight relative to the variability in the data. Wages are also typically greater for individuals with higher education levels: men with the lowest education level (1) tend to have substantially lower wages than those with the highest education level (5). Clearly, the most accurate prediction of a given man's `wage` will be obtained by combining his `age`, his `education`, and the `year`. In Chapter 3, we discuss linear regression, which can be used to predict `wage` from this data set. Ideally, we should predict `wage` in a way that accounts for the non-linear relationship between `wage` and `age`. In Chapter 7, we discuss a class of approaches for addressing this problem.

Stock Market Data

The `Wage` data involves predicting a *continuous* or *quantitative* output value. This is often referred to as a *regression* problem. However, in certain cases we may instead wish to predict a non-numerical value—that is, a *categorical* or *qualitative* output. For example, in Chapter 4 we examine a stock market data set that contains the daily movements in the Standard & Poor's 500 (S&P) stock index over a 5-year period between 2001 and 2005. We refer to this as the `Smarket` data. The goal is to predict whether the index will *increase* or *decrease* on a given day, using the past 5 days' percentage changes in the index. Here the statistical learning problem does not involve predicting a numerical value. Instead it involves predicting whether a given

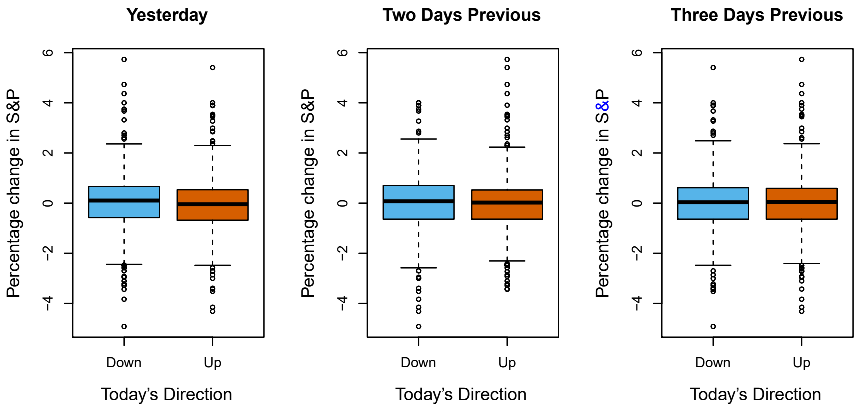

FIGURE 1.2. Left: *Boxplots of the previous day's percentage change in the S&P index for the days for which the market increased or decreased, obtained from the* `Smarket` *data.* Center and Right: *Same as left panel, but the percentage changes for 2 and 3 days previous are shown.*

day's stock market performance will fall into the `Up` bucket or the `Down` bucket. This is known as a *classification* problem. A model that could accurately predict the direction in which the market will move would be very useful!

The left-hand panel of Figure 1.2 displays two boxplots of the previous day's percentage changes in the stock index: one for the 648 days for which the market increased on the subsequent day, and one for the 602 days for which the market decreased. The two plots look almost identical, suggesting that there is no simple strategy for using yesterday's movement in the S&P to predict today's returns. The remaining panels, which display boxplots for the percentage changes 2 and 3 days previous to today, similarly indicate little association between past and present returns. Of course, this lack of pattern is to be expected: in the presence of strong correlations between successive days' returns, one could adopt a simple trading strategy to generate profits from the market. Nevertheless, in Chapter 4, we explore these data using several different statistical learning methods. Interestingly, there are hints of some weak trends in the data that suggest that, at least for this 5-year period, it is possible to correctly predict the direction of movement in the market approximately 60% of the time (Figure 1.3).

Gene Expression Data

The previous two applications illustrate data sets with both input and output variables. However, another important class of problems involves situations in which we only observe input variables, with no corresponding output. For example, in a marketing setting, we might have demographic information for a number of current or potential customers. We may wish to understand which types of customers are similar to each other by grouping individuals according to their observed characteristics. This is known as a

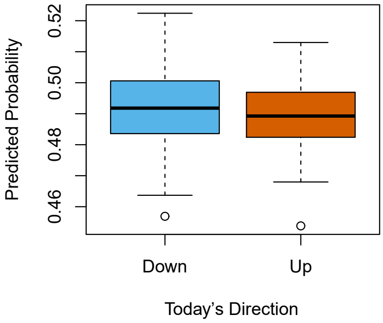

FIGURE 1.3. *We fit a quadratic discriminant analysis model to the subset of the* `Smarket` *data corresponding to the 2001–2004 time period, and predicted the probability of a stock market decrease using the 2005 data. On average, the predicted probability of decrease is higher for the days in which the market does decrease. Based on these results, we are able to correctly predict the direction of movement in the market 60% of the time.*

clustering problem. Unlike in the previous examples, here we are not trying to predict an output variable.

We devote Chapter 12 to a discussion of statistical learning methods for problems in which no natural output variable is available. We consider the `NCI60` data set, which consists of 6,830 gene expression measurements for each of 64 cancer cell lines. Instead of predicting a particular output variable, we are interested in determining whether there are groups, or clusters, among the cell lines based on their gene expression measurements. This is a difficult question to address, in part because there are thousands of gene expression measurements per cell line, making it hard to visualize the data.

The left-hand panel of Figure 1.4 addresses this problem by representing each of the 64 cell lines using just two numbers, Z_1 and Z_2. These are the first two *principal components* of the data, which summarize the 6,830 expression measurements for each cell line down to two numbers or *dimensions*. While it is likely that this dimension reduction has resulted in some loss of information, it is now possible to visually examine the data for evidence of clustering. Deciding on the number of clusters is often a difficult problem. But the left-hand panel of Figure 1.4 suggests at least four groups of cell lines, which we have represented using separate colors.

In this particular data set, it turns out that the cell lines correspond to 14 different types of cancer. (However, this information was not used to create the left-hand panel of Figure 1.4.) The right-hand panel of Figure 1.4 is identical to the left-hand panel, except that the 14 cancer types are shown using distinct colored symbols. There is clear evidence that cell lines with the same cancer type tend to be located near each other in this two-dimensional representation. In addition, even though the cancer information was not used to produce the left-hand panel, the clustering obtained does bear some resemblance to some of the actual cancer types observed in the right-hand panel. This provides some independent verification of the accuracy of our clustering analysis.

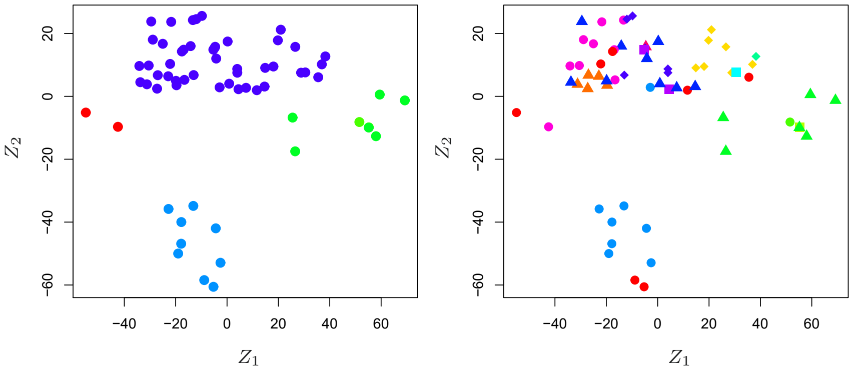

FIGURE 1.4. Left: *Representation of the* `NCI60` *gene expression data set in a two-dimensional space,* Z_1 *and* Z_2*. Each point corresponds to one of the* 64 *cell lines. There appear to be four groups of cell lines, which we have represented using different colors.* Right: *Same as left panel except that we have represented each of the* 14 *different types of cancer using a different colored symbol. Cell lines corresponding to the same cancer type tend to be nearby in the two-dimensional space.*

A Brief History of Statistical Learning

Though the term *statistical learning* is fairly new, many of the concepts that underlie the field were developed long ago. At the beginning of the nineteenth century, the method of *least squares* was developed, implementing the earliest form of what is now known as *linear regression*. The approach was first successfully applied to problems in astronomy. Linear regression is used for predicting quantitative values, such as an individual's salary. In order to predict qualitative values, such as whether a patient survives or dies, or whether the stock market increases or decreases, *linear discriminant analysis* was proposed in 1936. In the 1940s, various authors put forth an alternative approach, *logistic regression*. In the early 1970s, the term *generalized linear model* was developed to describe an entire class of statistical learning methods that include both linear and logistic regression as special cases.

By the end of the 1970s, many more techniques for learning from data were available. However, they were almost exclusively *linear* methods because fitting *non-linear* relationships was computationally difficult at the time. By the 1980s, computing technology had finally improved sufficiently that non-linear methods were no longer computationally prohibitive. In the mid 1980s, *classification and regression trees* were developed, followed shortly by *generalized additive models*. *Neural networks* gained popularity in the 1980s, and *support vector machines* arose in the 1990s.

Since that time, statistical learning has emerged as a new subfield in statistics, focused on supervised and unsupervised modeling and prediction. In recent years, progress in statistical learning has been marked by the increasing availability of powerful and relatively user-friendly software, such as the popular and freely available `Python` system. This has the potential to continue the transformation of the field from a set of techniques used and developed by statisticians and computer scientists to an essential toolkit for a much broader community.

This Book

The Elements of Statistical Learning (ESL) by Hastie, Tibshirani, and Friedman was first published in 2001. Since that time, it has become an important reference on the fundamentals of statistical machine learning. Its success derives from its comprehensive and detailed treatment of many important topics in statistical learning, as well as the fact that (relative to many upper-level statistics textbooks) it is accessible to a wide audience. However, the greatest factor behind the success of ESL has been its topical nature. At the time of its publication, interest in the field of statistical learning was starting to explode. ESL provided one of the first accessible and comprehensive introductions to the topic.

Since ESL was first published, the field of statistical learning has continued to flourish. The field's expansion has taken two forms. The most obvious growth has involved the development of new and improved statistical learning approaches aimed at answering a range of scientific questions across a number of fields. However, the field of statistical learning has also expanded its audience. In the 1990s, increases in computational power generated a surge of interest in the field from non-statisticians who were eager to use cutting-edge statistical tools to analyze their data. Unfortunately, the highly technical nature of these approaches meant that the user community remained primarily restricted to experts in statistics, computer science, and related fields with the training (and time) to understand and implement them.

In recent years, new and improved software packages have significantly eased the implementation burden for many statistical learning methods. At the same time, there has been growing recognition across a number of fields, from business to health care to genetics to the social sciences and beyond, that statistical learning is a powerful tool with important practical applications. As a result, the field has moved from one of primarily academic interest to a mainstream discipline, with an enormous potential audience. This trend will surely continue with the increasing availability of enormous quantities of data and the software to analyze it.

The purpose of *An Introduction to Statistical Learning* (ISL) is to facilitate the transition of statistical learning from an academic to a mainstream field. ISL is not intended to replace ESL, which is a far more comprehensive text both in terms of the number of approaches considered and the depth to which they are explored. We consider ESL to be an important companion for professionals (with graduate degrees in statistics, machine learning, or related fields) who need to understand the technical details behind statistical learning approaches. However, the community of users of statistical learning techniques has expanded to include individuals with a wider range of interests and backgrounds. Therefore, there is a place for a less technical and more accessible version of ESL.

In teaching these topics over the years, we have discovered that they are of interest to master's and PhD students in fields as disparate as business administration, biology, and computer science, as well as to quantitatively-oriented upper-division undergraduates. It is important for this diverse group to be able to understand the models, intuitions, and strengths and weaknesses of the various approaches. But for this audience, many of the technical details behind statistical learning methods, such as optimization algorithms and theoretical properties, are not of primary interest. We believe that these students do not need a deep understanding of these aspects in order to become informed users of the various methodologies, and in order to contribute to their chosen fields through the use of statistical learning tools.

ISL is based on the following four premises.

1. *Many statistical learning methods are relevant and useful in a wide range of academic and non-academic disciplines, beyond just the statistical sciences.* We believe that many contemporary statistical learning procedures should, and will, become as widely available and used as is currently the case for classical methods such as linear regression. As a result, rather than attempting to consider every possible approach (an impossible task), we have concentrated on presenting the methods that we believe are most widely applicable.

2. *Statistical learning should not be viewed as a series of black boxes.* No single approach will perform well in all possible applications. Without understanding all of the cogs inside the box, or the interaction between those cogs, it is impossible to select the best box. Hence, we have attempted to carefully describe the model, intuition, assumptions, and trade-offs behind each of the methods that we consider.

3. *While it is important to know what job is performed by each cog, it is not necessary to have the skills to construct the machine inside the box!* Thus, we have minimized discussion of technical details related to fitting procedures and theoretical properties. We assume that the reader is comfortable with basic mathematical concepts, but we do not assume a graduate degree in the mathematical sciences. For instance, we have almost completely avoided the use of matrix algebra, and it is possible to understand the entire book without a detailed knowledge of matrices and vectors.

4. *We presume that the reader is interested in applying statistical learning methods to real-world problems.* In order to facilitate this, as well as to motivate the techniques discussed, we have devoted a section within each chapter to computer labs. In each lab, we walk the reader through a realistic application of the methods considered in that chapter. When we have taught this material in our courses, we have allocated roughly one-third of classroom time to working through the labs, and we have found them to be extremely useful. Many of the less computationally-oriented students who were initially intimidated by the labs got the hang of things over the course of the quarter or semester. This book originally appeared (2013, second edition 2021)

with computer labs written in the R language. Since then, there has been increasing demand for Python implementations of the important techniques in statistical learning. Consequently, this version has labs in Python. There are a rapidly growing number of Python packages available, and by examination of the imports at the beginning of each lab, readers will see that we have carefully selected and used the most appropriate. We have also supplied some additional code and functionality in our package ISLP. However, the labs in ISL are self-contained, and can be skipped if the reader wishes to use a different software package or does not wish to apply the methods discussed to real-world problems.

Who Should Read This Book?

This book is intended for anyone who is interested in using modern statistical methods for modeling and prediction from data. This group includes scientists, engineers, data analysts, data scientists, and quants, but also less technical individuals with degrees in non-quantitative fields such as the social sciences or business. We expect that the reader will have had at least one elementary course in statistics. Background in linear regression is also useful, though not required, since we review the key concepts behind linear regression in Chapter 3. The mathematical level of this book is modest, and a detailed knowledge of matrix operations is not required. This book provides an introduction to Python. Previous exposure to a programming language, such as MATLAB or R, is useful but not required.

The first edition of this textbook has been used to teach master's and PhD students in business, economics, computer science, biology, earth sciences, psychology, and many other areas of the physical and social sciences. It has also been used to teach advanced undergraduates who have already taken a course on linear regression. In the context of a more mathematically rigorous course in which ESL serves as the primary textbook, ISL could be used as a supplementary text for teaching computational aspects of the various approaches.

Notation and Simple Matrix Algebra

Choosing notation for a textbook is always a difficult task. For the most part we adopt the same notational conventions as ESL.

We will use n to represent the number of distinct data points, or observations, in our sample. We will let p denote the number of variables that are available for use in making predictions. For example, the Wage data set consists of 11 variables for 3,000 people, so we have $n = 3{,}000$ observations and $p = 11$ variables (such as year, age, race, and more). Note that throughout this book, we indicate variable names using colored font: Variable Name.

In some examples, p might be quite large, such as on the order of thousands or even millions; this situation arises quite often, for example, in the analysis of modern biological data or web-based advertising data.

In general, we will let x_{ij} represent the value of the jth variable for the ith observation, where $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, p$. Throughout this book, i will be used to index the samples or observations (from 1 to n) and j will be used to index the variables (from 1 to p). We let $\mathbf{X}$ denote an $n \times p$ matrix whose (i, j)th element is x_{ij}. That is,

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{pmatrix}.$$

For readers who are unfamiliar with matrices, it is useful to visualize $\mathbf{X}$ as a spreadsheet of numbers with n rows and p columns.

At times we will be interested in the rows of $\mathbf{X}$, which we write as $x_1, x_2, \ldots, x_n$. Here x_i is a vector of length p, containing the p variable measurements for the ith observation. That is,

$$x_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{ip} \end{pmatrix}. \tag{1.1}$$

(Vectors are by default represented as columns.) For example, for the `Wage` data, x_i is a vector of length 11, consisting of `year`, `age`, `race`, and other values for the ith individual. At other times we will instead be interested in the columns of $\mathbf{X}$, which we write as $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p$. Each is a vector of length n. That is,

$$\mathbf{x}_j = \begin{pmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{nj} \end{pmatrix}.$$

For example, for the `Wage` data, $\mathbf{x}_1$ contains the $n = 3{,}000$ values for `year`.

Using this notation, the matrix $\mathbf{X}$ can be written as

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_p \end{pmatrix},$$

or

$$\mathbf{X} = \begin{pmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{pmatrix}.$$

The ^{T} notation denotes the *transpose* of a matrix or vector. So, for example,

$$\mathbf{X}^T = \begin{pmatrix} x_{11} & x_{21} & \ldots & x_{n1} \\ x_{12} & x_{22} & \ldots & x_{n2} \\ \vdots & \vdots & & \vdots \\ x_{1p} & x_{2p} & \ldots & x_{np} \end{pmatrix},$$

while

$$x_i^T = \begin{pmatrix} x_{i1} & x_{i2} & \cdots & x_{ip} \end{pmatrix}.$$

We use y_i to denote the ith observation of the variable on which we wish to make predictions, such as wage. Hence, we write the set of all n observations in vector form as

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}.$$

Then our observed data consists of $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$, where each x_i is a vector of length p. (If $p = 1$, then x_i is simply a scalar.)

In this text, a vector of length n will always be denoted in *lower case bold*; e.g.

$$\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}.$$

However, vectors that are not of length n (such as feature vectors of length p, as in (1.1)) will be denoted in *lower case normal font*, e.g. a. Scalars will also be denoted in *lower case normal font*, e.g. a. In the rare cases in which these two uses for lower case normal font lead to ambiguity, we will clarify which use is intended. Matrices will be denoted using *bold capitals*, such as $\mathbf{A}$. Random variables will be denoted using *capital normal font*, e.g. A, regardless of their dimensions.

Occasionally we will want to indicate the dimension of a particular object. To indicate that an object is a scalar, we will use the notation $a \in \mathbb{R}$. To indicate that it is a vector of length k, we will use $a \in \mathbb{R}^k$ (or $\mathbf{a} \in \mathbb{R}^n$ if it is of length n). We will indicate that an object is an $r \times s$ matrix using $\mathbf{A} \in \mathbb{R}^{r \times s}$.

We have avoided using matrix algebra whenever possible. However, in a few instances it becomes too cumbersome to avoid it entirely. In these rare instances it is important to understand the concept of multiplying two matrices. Suppose that $\mathbf{A} \in \mathbb{R}^{r \times d}$ and $\mathbf{B} \in \mathbb{R}^{d \times s}$. Then the product of $\mathbf{A}$ and $\mathbf{B}$ is denoted $\mathbf{AB}$. The (i, j)th element of $\mathbf{AB}$ is computed by multiplying each element of the ith row of $\mathbf{A}$ by the corresponding element of the jth column of $\mathbf{B}$. That is, $(\mathbf{AB})_{ij} = \sum_{k=1}^{d} a_{ik} b_{kj}$. As an example, consider

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \quad \text{and} \quad \mathbf{B} = \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}.$$

Then

$$\mathbf{AB} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix} = \begin{pmatrix} 1 \times 5 + 2 \times 7 & 1 \times 6 + 2 \times 8 \\ 3 \times 5 + 4 \times 7 & 3 \times 6 + 4 \times 8 \end{pmatrix} = \begin{pmatrix} 19 & 22 \\ 43 & 50 \end{pmatrix}.$$

Note that this operation produces an $r \times s$ matrix. It is only possible to compute $\mathbf{AB}$ if the number of columns of $\mathbf{A}$ is the same as the number of rows of $\mathbf{B}$.

Organization of This Book

Chapter 2 introduces the basic terminology and concepts behind statistical learning. This chapter also presents the K*-nearest neighbor* classifier, a very simple method that works surprisingly well on many problems. Chapters 3 and 4 cover classical linear methods for regression and classification. In particular, Chapter 3 reviews *linear regression*, the fundamental starting point for all regression methods. In Chapter 4 we discuss two of the most important classical classification methods, *logistic regression* and *linear discriminant analysis*.

A central problem in all statistical learning situations involves choosing the best method for a given application. Hence, in Chapter 5 we introduce *cross-validation* and the *bootstrap*, which can be used to estimate the accuracy of a number of different methods in order to choose the best one.

Much of the recent research in statistical learning has concentrated on non-linear methods. However, linear methods often have advantages over their non-linear competitors in terms of interpretability and sometimes also accuracy. Hence, in Chapter 6 we consider a host of linear methods, both classical and more modern, which offer potential improvements over standard linear regression. These include *stepwise selection*, *ridge regression*, *principal components regression*, and the *lasso*.

The remaining chapters move into the world of non-linear statistical learning. We first introduce in Chapter 7 a number of non-linear methods that work well for problems with a single input variable. We then show how these methods can be used to fit non-linear *additive* models for which there is more than one input. In Chapter 8, we investigate *tree*-based methods, including *bagging*, *boosting*, and *random forests*. *Support vector machines*, a set of approaches for performing both linear and non-linear classification, are discussed in Chapter 9. We cover *deep learning*, an approach for non-linear regression and classification that has received a lot of attention in recent years, in Chapter 10. Chapter 11 explores *survival analysis*, a regression approach that is specialized to the setting in which the output variable is *censored*, i.e. not fully observed.

In Chapter 12, we consider the *unsupervised* setting in which we have input variables but no output variable. In particular, we present *principal components analysis*, K*-means clustering*, and *hierarchical clustering*. Finally, in Chapter 13 we cover the very important topic of multiple hypothesis testing.

At the end of each chapter, we present one or more `Python` lab sections in which we systematically work through applications of the various methods discussed in that chapter. These labs demonstrate the strengths and weaknesses of the various approaches, and also provide a useful reference for the syntax required to implement the various methods. The reader may choose to work through the labs at their own pace, or the labs may be the focus of group sessions as part of a classroom environment. Within each `Python` lab, we present the results that we obtained when we performed the lab at the time of writing this book. However, new versions of `Python` are continuously released, and over time, the packages called in the labs will be updated. Therefore, in the future, it is possible that the results shown in

Name	Description
Auto	Gas mileage, horsepower, and other information for cars.
Bikeshare	Hourly usage of a bike sharing program in Washington, DC.
Boston	Housing values and other information about Boston census tracts.
BrainCancer	Survival times for patients diagnosed with brain cancer.
Caravan	Information about individuals offered caravan insurance.
Carseats	Information about car seat sales in 400 stores.
College	Demographic characteristics, tuition, and more for USA colleges.
Credit	Information about credit card debt for 400 customers.
Default	Customer default records for a credit card company.
Fund	Returns of 2,000 hedge fund managers over 50 months.
Hitters	Records and salaries for baseball players.
Khan	Gene expression measurements for four cancer types.
NCI60	Gene expression measurements for 64 cancer cell lines.
NYSE	Returns, volatility, and volume for the New York Stock Exchange.
OJ	Sales information for Citrus Hill and Minute Maid orange juice.
Portfolio	Past values of financial assets, for use in portfolio allocation.
Publication	Time to publication for 244 clinical trials.
Smarket	Daily percentage returns for S&P 500 over a 5-year period.
USArrests	Crime statistics per 100,000 residents in 50 states of USA.
Wage	Income survey data for men in central Atlantic region of USA.
Weekly	1,089 weekly stock market returns for 21 years.

TABLE 1.1. *A list of data sets needed to perform the labs and exercises in this textbook. All data sets are available in the* ISLP *package, with the exception of* USArrests*, which is part of the base* R *distribution, but accessible from* Python*.*

the lab sections may no longer correspond precisely to the results obtained by the reader who performs the labs. As necessary, we will post updates to the labs on the book website.

We use the ◆ symbol to denote sections or exercises that contain more challenging concepts. These can be easily skipped by readers who do not wish to delve as deeply into the material, or who lack the mathematical background.

Data Sets Used in Labs and Exercises

In this textbook, we illustrate statistical learning methods using applications from marketing, finance, biology, and other areas. The ISLP package contains a number of data sets that are required in order to perform the labs and exercises associated with this book. One other data set is part of the base R distribution (the USArrests data), and we show how to access it from Python in Section 12.5.1. Table 1.1 contains a summary of the data sets required to perform the labs and exercises. A couple of these data sets are also available as text files on the book website, for use in Chapter 2.

Book Website

The website for this book is located at

`www.statlearning.com`

It contains a number of resources, including the `Python` package associated with this book, and some additional data sets.

Acknowledgements

A few of the plots in this book were taken from ESL: Figures 6.7, 8.3, and 12.14. All other plots were produced for the `R` version of ISL, except for Figure 13.10 which differs because of the `Python` software supporting the plot.