Introduction

February 15, 2017

Contents

1 An Overview

1.1 Uses 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.

Wage Data

In this application (which we refer to as the Wage data set throughout this course), we examine a number of factors that relate to wages for a group of males 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.

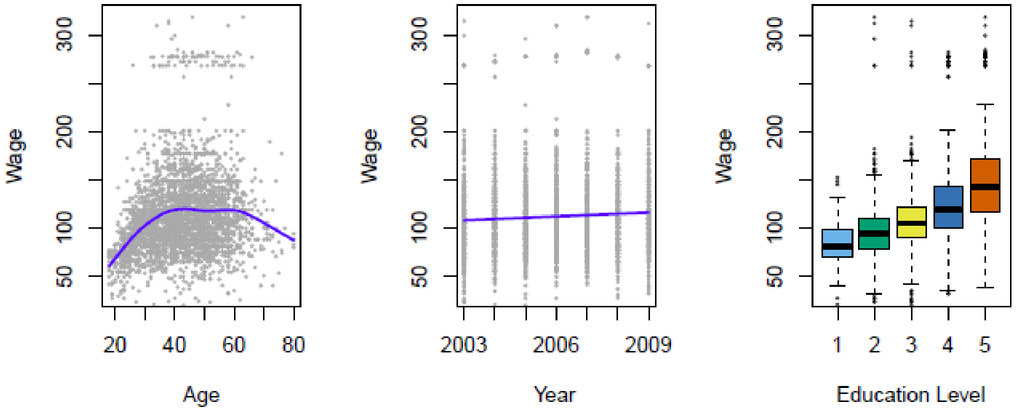

Figure 1: Wage data, which contains income survey information for males from the central Atlantic region of the United States. Left: wage as a function of age. On average, wage increases with age until about 60 years of age, at which point it begins to decline. Center: wage as a function of year. There 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.

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 cat- egorical 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 five-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 five 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 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!

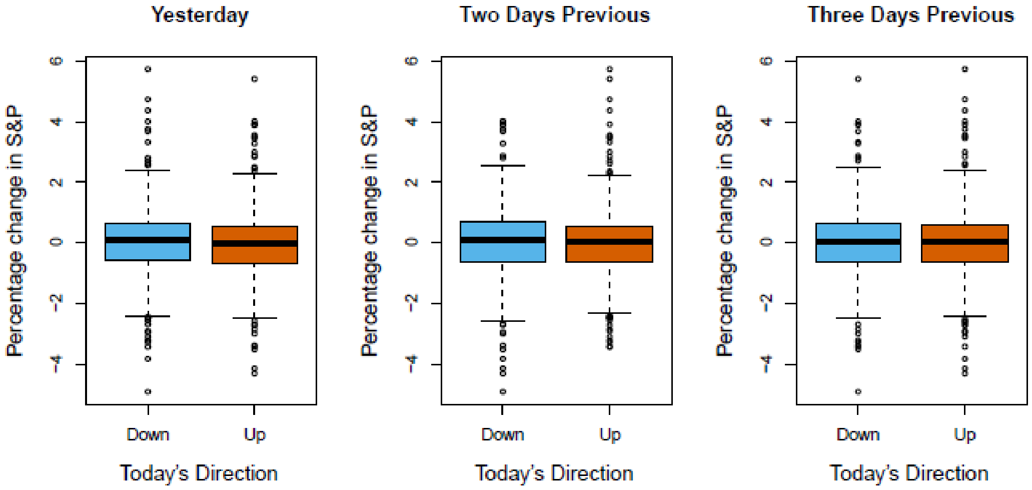

Figure 2: Left: Boxplots of the previous day's percentage change in the S&P index for the days for which the market increased or decreased, obtained from the Smarket data. Center and Right: Same as left panel, but the percentage changes for two and three days previous are shown.

The left-hand panel of Figure 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. 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.

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

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.

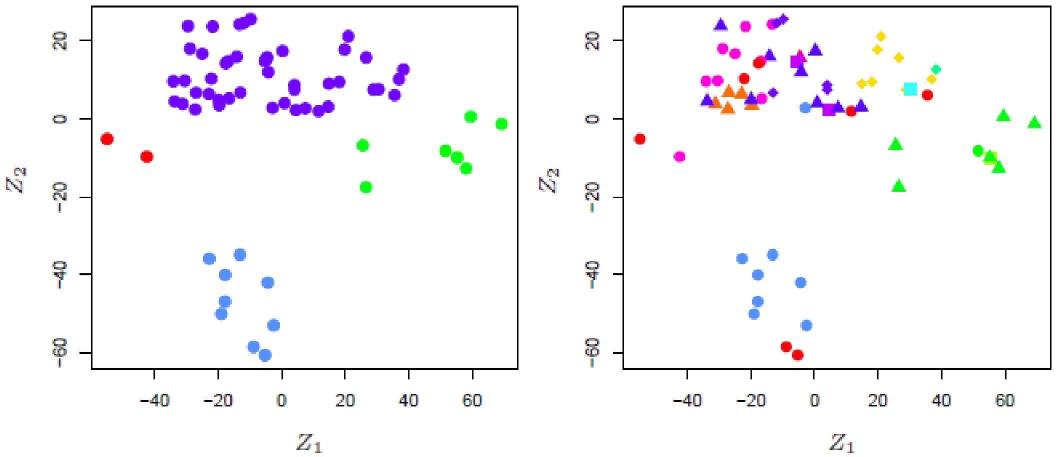

Figure 3: Left: Representation of the NCI60 gene expression data set in a two-dimensional space, Z1 and Z2. Each point corresponds to one of the 64 cell lines. There appear to be four groups of cell lines, which we have represented using different colors. Right: Same as left panel except that we have 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.

The left-hand panel of Figure addresses this problem by representing each of the 64 cell lines using just two numbers, Z1 and Z2. 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 suggests at least four groups of cell lines, which we have represented using separate colors. We can now examine the cell lines within each cluster for similarities in their types of cancer, in order to better understand the relationship between gene expression levels and cancer.

Statistical learning is primarily interested in making sense of complex data. As you might expect, statistical learning is used widely. For instance, it has been used to:

- Predict the outcomes of elections
- Identify and filter spam messages from e-mail
- Foresee criminal activity
- Automate traffic signals according to road conditions
- Produce financial estimates of storms and natural disasters
- Examine customer churn
- Create auto-piloting planes and auto-driving cars
- Identify individuals with the capacity to donate
- Target advertising to specific types of consumers

1.2 Steps to apply Statistical Learning to your data

Any Statistical learning task can be broken down into a series of more manageable steps.

1. Collecting data: Whether the data is written on paper, recorded in text files and spreadsheets, or stored in an SQL database, you will need to gather it in an electronic format suitable for analysis. This data will serve as the learning material an algorithm uses to generate actionable knowledge.

2. Exploring and preparing the data: The quality of any statistical learning project is based largely on the quality of data it uses. This step in the statistical learning process tends to require a great deal of human intervention. An often cited statistic suggests that 80 percent of the effort in statistical learning is devoted to data. Much of this time is spent learning more about the data and its nuances during a practice called data exploration.

3. Training a model on the data: By the time the data has been prepared for analysis, you are likely to have a sense of what you are hoping to learn from the data. The specific statistical learning task will inform the selection of an appropriate algorithm, and the algorithm will represent the data in the form of a model.

4. Evaluating model performance: Because each statistical learning model results in a biased solution to the learning problem, it is important to evaluate how well the algorithm learned from its experience. Depending on the type of model used, you might be able to evaluate the accuracy of the model using a test dataset, or you may need to develop measures of performance specific to the intended application.

5. Improving model performance: If better performance is needed, it becomes necessary to utilize more advanced strategies to augment the performance of the model. Sometimes, it may be necessary to switch to a different type of model altogether. You may need to supplement your data with additional data, or perform additional preparatory work as in step two of this process.

There is no free lunch in statistics: no one method dominates all others over all possible data sets. On a particular data set, one specific method may work best, but some other method may work better on a similar but different data set. Hence it is an important task to decide for any given set of data which method produces the best results. Selecting the best approach can be one of the most challenging parts of performing statistical learning in practice.

Measuring the Quality of Fit

In order to evaluate the performance of a statistical learning method on a given data set, we need some way to measure how well its predictions actually match the observed data. That is, we need to quantify the extent to which the predicted response value for a given observation is close to the true response value for that observation. In the regression setting, the most commonly-used measure is the mean squared error (MSE), given by

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2,$$

where $\hat{f}(x_i)$ is the prediction that $\hat{f}$ gives for the ith observation. The MSE will be small if the predicted responses are very close to the true responses, and will be large if for some of the observations, the predicted and true responses differ substantially.

1.3 Supervised versus Unsupervised Learning

Most statistical learning problems fall into one of two categories: supervised or unsupervised. The examples that we have discussed so far in this chapter all fall into the supervised learning domain. For each observation of the predictor measurement(s) $x_i, i = 1, ..., n$ there is an associated response measurement y_i. We wish to fit a model that relates the response to the predictors, with the aim of accurately predicting the response for future observations (prediction) or better understanding the relationship between the response and the predictors (inference). Many classical statistical learning methods such as linear regression and logistic regression, as well as more modern approaches such as GAM, boosting, and support vector machines, operate in the supervised learning domain.

In contrast, unsupervised learning describes the somewhat more challenging situation in which for every observation $i = 1, ..., n$, we observe a vector of measurements x_i but no associated response y_i. It is not possible to fit a linear regression model, since there is no response variable to predict. In this setting, we are in some sense working blind; the situation is referred to as unsupervised because we lack a response variable that can supervise our analysis. What sort of statistical analysis is possible? We can seek to understand the relationships between the variables or between the observations. One statistical learning tool that we may use in this setting is cluster analysis, or clustering. The goal of cluster analysis is to ascertain, on the basis of $x_1, ..., x_n$, whether the observations fall into relatively distinct groups. For example, in a market segmentation study we might observe multiple characteristics (variables) for potential customers, such as zip code, family income, and shopping habits. We might believe that the customers fall into different groups, such as big spenders versus low spenders. If the information about each customer's spending patterns were available, then a supervised analysis would be possible. However, this information is not available — that is, we do not know whether each potential customer is a big spender or not. In this setting, we can try to cluster the customers on the basis of the variables measured, in order to identify distinct groups of potential customers. Identifying such groups can be of interest because it might be that the groups differ with respect to some property of interest, such as spending habits.

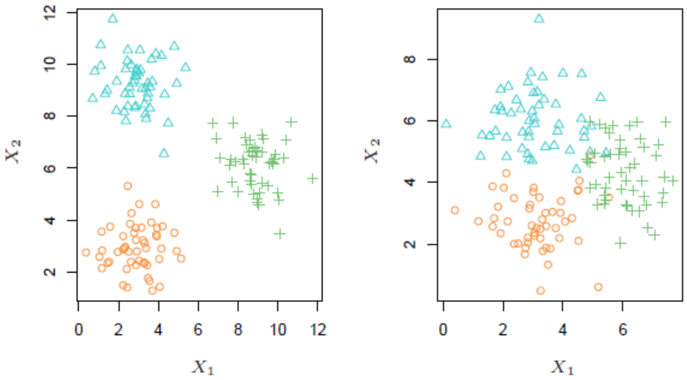

Figure 4: A clustering data set involving three groups. Each group is shown using a different colored symbol. Left: The three groups are well-separated. In this setting, a clustering approach should successfully identify the three groups. Right: There is some overlap among the groups. Now the clustering task is more challenging.

2 About R

2.1 History of R

R started in the early 1990's as a project by Ross Ihaka and Robert Gentleman at the University of Auckland, New Zealand, intended to provide a statistical environment in their teaching lab.

Why use R?

2.2 Features of R

R is a language and environment for statistical computing and graphics, similar to the S language originally developed at Bell Labs. It's an open source solution to data analysis that's supported by a large and active worldwide research community. But there are many popular statistical and graphing packages available (such as Microsoft Excel, SAS, IBM SPSS, Stata, and Minitab). Why turn to R? R has many features to recommend it:

- Most commercial statistical software platforms cost thousands, if not tens of thousands of dollars. R is free! If you're a teacher or a student, the benefits are obvious.
- R runs on a wide array of platforms, including Windows, Unix, and Mac OS X. It's likely to run on any computer you might have.
- Quite lean, as far as software goes; functionality is divided into modular packages.
- R provides an unparalleled platform for programming new statistical methods in an easy and straightforward manner. It's easily extensible and provides a natural language for quickly programming recently published methods.
- R contains advanced statistical routines not yet available in other packages. In fact, new methods become available for download on a weekly basis.
- Graphics capabilities very sophisticated and better than most stat packages.
- Very active and vibrant user community; R-help and R-devel mailing lists and Stack Overflow. http://www.r-project.org/mail.html
- and so on.

R and Statistics

- Most packages deal with statistics and data analysis.
- Powerful statistical graphics.
- Well crosstalking with other statistical softwares.
- Most R user are statistical experts. You can learn more modern analysis method from they by email.
- You can do it when you come across a thing no body do it before.

2.3 Drawbacks of R

Unfortunately, R can have a steep learning curve. Because it can do so much, the documentation and help files available are voluminous. Additionally, because much of the functionality comes from optional modules created by independent contributors, this documentation can be scattered and difficult to locate. In fact, getting a handle on all that R can do is a challenge. R is not the right tool for every problem. Clearly, it would be ridiculous to write a video game in R, but it's not even the best tool for all data problems. R is very good at plotting graphics, analyzing data, and fitting statistical models using data that fits in the computer's memory. It's not as good at storing data in complicated structures, efficiently querying data, or working with data that doesn't fit in the computer's memory.

- Essentially based on 40 year old technology.
- Little built in support for dynamic or 3-D graphics (but things have improved greatly since the "old days").
- Functionality is based on consumer demand and user contributions. If no one feels like implementing your favorite method, then it's your job!
  - (Or you need to pay someone to do it)
- Objects must generally be stored in physical memory.
- Not ideal for all possible situations (but this is a drawback of all software packages)

2.4 Design of the R System

R is freely available from the Comprehensive R Archive Network (CRAN) at http://cran.r-project.org. Precompiled binaries are available for Linux, Mac OS X, and Windows. Follow the directions for installing the base product on the platform of your choice. Later we'll talk about adding functionality through optional modules called packages (also available from CRAN). About installing and setting R, see Appendix for details.

The R system is divided into 2 conceptual parts:

1. The "base" R system that you download from CRAN
2. Everything else

R functionality is divided into a number of packages

1. The "base" R system contains, among other things, the base package which is required to run R and contains the most fundamental functions.
2. The other packages contained in the "base" system include utils, stats, datasets, graphics, grDevices, grid, methods, tools, parallel, compiler, splines, tcltk, stats4.
3. There are also "Recommend" packages: boot, class, cluster, codetools, foreign, KernSmooth, lattice, mgcv, nlme, rpart, survival, MASS, spatial, nnet, Matrix.

And there are many other packages available:

1. There are about 10000+ packages on CRAN that have been developed by users and programmers around the world.
2. There are also many packages associated with the Bioconductor project (http://bioconductor.org).
3. People often make packages available on their personal websites; there is no reliable way to keep track of how many packages are available in this fashion.

Several projects are aiming to build an easier-to-use GUI for R:

- RStudio is a free and open source integrated development environment (IDE) for R, a programming language for statistical computing and graphics. It can be download from http://www.rstudio.com/.
- The Rcmdr project is an R package that provides an alternative GUI for R. You can install it as an R package. It provides some buttons for loading data and menu items for many common R functions.

You can find a list of additional projects at http://www.sciviews.org/_rgui/.

3 Getting helps

3.1 Helps

R provides extensive help facilities, and learning to navigate them will help you significantly in your programming efforts. The built-in help system provides details, references, and examples of any function contained in a currently installed package.

For example:

```r
help.start()
help()
help("options")
?options
```

If the exact name of a command is not known, as will often be the case for beginners, the functions to use are help.search() and apropos(). help.search() returns help files with aliases or concepts or titles matching a “pattern”using fuzzy matching. Thus, if help on options settings is desired but the exact command name, here options(), is unknown, a search for objects containing the pattern“option”might be useful. help.search("option") will return a (long) list of commands, data frames, etc., containing this pattern. Alternatively, the function apropos() lists all functions whose names include the pattern entered.

```
apropos("help")
```

Furthermore, there are several collections of frequently asked questions (FAQs) at http://CRAN.R-project.org/faqs.html that provide answers to general questions about R and also about platform-specific issues on Microsoft Windows and Mac OS X.

Moreover, there is an online newsletter named R News, launched in 2001. It is currently published about three times per year and features, among other things, recent developments in R (such as changes in the language or new addon packages), a “programmer’s niche”, and examples analyzing data with R. See http://CRAN.R-project.org/doc/Rnews/ for further information.

For a growing number of R packages, there exist corresponding publications in the Journal of Statistical Software; see http://www.jstatsoft.org/. This is an open-access journal that publishes articles and code snippets (as well as book and software reviews) on the subject of statistical software and algorithms.

3.2 Several commands

If you know the name of the function you want help with, you just type a question mark ? at the command line prompt followed by the name of the function. So to get help on read.table, just type

```
?read.table
```

Sometimes you cannot remember the precise name of the function, but you know the subject on which you want help (e.g. data input in this case). Use the help.search function (without a question mark) with your query in double quotes like this:

```
help.search("data input")
```

The find function tells you what package something is in:

```
find("lowess")

## [1] "package:stats"
```

Apropos returns a character vector giving the names of all objects in the search list that match your (potentially partial) enquiry:

```
apropos("lm")
```

To see a worked example just type the function name (e.g. linear models, lm)

```
example(lm)
```

Demonstrations of R functions are

```
demo(persp)
demo(graphics)
demo(Hershey)
demo(plotmath)
```

Update R

```
install.packages("installr")
require(installr)
updateR()
```

3.3 Some R Resources

Available from CRAN (http://cran.r-project.org)

- An Introduction to R
- Writing R Extensions
- R Data Import/Export
- R Installation and Administration (mostly for building R from sources)
- R Internals (not for the faint of heart)

Other resources

- Springer has a series of books called Use R!.
- A longer list of books is at http://www.r-project.org/doc/bib/R-books.html

3.4 Finding Answers

- Try to find an answer by searching the archives of the forum you plan to post to.
- Try to find an answer by searching the Web.
- Try to find an answer by reading the manual.
- Try to find an answer by reading a FAQ. (http://cran.r-project.org/)
- Try to find an answer by inspection or experimentation.
- Try to find an answer by asking a skilled friend.

3.5 Ask Questions

First, you may read the posting guide: http://www.r-project.org/posting-guide.html

- Asking questions via email is different from asking questions in person
- People on the other side do not have the background information you have
  - they also don't know you personally (usually)
- Other people are busy; their time is limited

- The instructor (me) is here to help in all circumstances but may not be able to answer all questions!

It's important to let other people know that you've done all of the previous things already. If the answer is in the documentation, the answer will be "Read the documentation". One email round wasted!

- What steps will reproduce the problem?
- What is the expected output?
- What do you see instead?
- What version of the product (e.g. R, packages, etc.) are you using?
- What operating system?
- Additional information

Ask questions as follows:

- Describe the goal, not the step
- Be explicit about your question
- Do provide the minimum amount of information necessary
- Follow up with the solution (if found)

Don't

- Email multiple mailing lists at once
- Ask others to debug your broken code without giving a hint as to what sort of problem they should be searching for

4 Collecting data

Data collection is the process of gathering and measuring information on variables of interest, in an established systematic fashion that enables one to answer stated research questions, test hypotheses, and evaluate outcomes.

Where will we collect the data?

1. Penn World Table (https://pwt.sas.upenn.edu/)
2. World bank (http://databank.worldbank.org/data/databases.aspx)
3. OECD database (http://stats.oecd.org/Index.aspx)
4. International Monetary Fund (http://www.imf.org/external/data.htm#data)
5. National Bureau of Statistics of China (http://www.stats.gov.cn/tjsj/ndsj/)
6. Yahoo! Finance (http://finance.yahoo.com/)
7. Datastream (http://library.xmu.edu.cn/portal/database_detail.asp?id=313)