THE UCLA ELECTRONIC STATISTICS TEXTBOOK

JAN DE LEEUW

CONTENTS

1. INTRODUCTION

The *UCLA Electronic Statistics Textbook*, from now on UCLA-EST, is an attempt to write a statistics textbook which is

1. freely available to everyone on the Internet;
2. independent of the level of the student, i.e. useful at the undergraduate, graduate, and postdoc level;
3. interactive, using graphics and demos;
4. complete, i.e. it covers most of statistical theory as traditionally taught.

We use the model of a classical statistics textbook to bring together hypertext describing the traditional concepts and techniques of statistics and graphical components illustrating these concepts and techniques.

This particular mix, except for the hypertext, would just be a classical textbook that can be read online. But we also add interactive graphics, demos, and calculators to the mix, and thus we create something with combines a classical textbook with computer software for demonstrations. Ultimately, we also plan to add exercises, examples, glossaries, and instructors manuals.

UCLA-EST is very much *under construction*, and it will probably be under construction forever. It is not a product, but a project. One of the major advantages of the electronic form is that there is no limit on the size of the book, and that material can be added at any point in time. Moreover the number of printings and the number of editions is equally unlimited.

In this paper we will give an overview of UCLA-EST, its relationships with some other projects, its structure, and some of the technical problems in its implementation. We will *not* include graphical representations of the pages. It is much better if the reader browses the textbook while reading this paper. Thus, we suggest you now go to

`http://www.stat.ucla.edu/textbook/`

2. Structure of UCLA-EST

2.1. **Structure of a Page.** Each page of the textbook has a header consisting of the textbook logo (a colorful histogram), plus the title of the book, which is

Statistics. The Study of Stability in Variation

Each page has a footer, which consists of text-based buttons linking to the glossary, the table of contents, the toolbox, the textbook background, the formulas section, the textbook homepage, and the UCLA Statistics homepage. Readers can go to these places from any textbook page.

The footer also has a contact address, a counter showing the number of visitors, the full URL of the page, and the date it was last modified. The counter is text-based, and done with a CGI script, the modification date comes from a server-side include.

The implementation of headers and footers is explained in more detail in subsection **??**.

2.2. **Table of Contents.** The table of contents of the textbook is fairly traditional. We only give it up to two levels deep.

1. Introduction
   (a) Variables
   (b) Variation
   (c) Sampling
   (d) Models
   (e) Inference
2. Analysis of a Single Variable
   (a) A Single Variable, Descriptive.
   (b) A Single Variable, Several Conditions.
   (c) A Single Variable, in Time.
   (d) A Single Variable, with Structured Conditions
3. Analysis of a Pair of Variables
   (a) Cross Tables
   (b) Correlation
   (c) Probability Models for Bivariation
4. Analysis of Multivariables
   (a) Describing Multivariables
   (b) Decomposing Multivariables
   (c) Path Analysis, Factor Analysis, and friends
   (d) Probability Models for Multivariables

(e) Eigenvalues and Friends

If your browser can do Javascript, then you get a better idea by browsing

`http://www.stat.ucla.edu/textbook/textbookol.html`

Again, many of the section are empty, or almost empty, and more sections will be added.

3. Philosophy

There are undoubtedly those, who think that an author's or editor's philosophy should not influence the contents of the textbook they write or edit. I do not agree. Ultimately, one of the factors that will hold the textbook together is a strong influence of a certain interpretation of statistics. One should not forget that one of the major disadvantages over hypertext and on-line books is that they have, by definition, much less unity and cohesiveness as a bound book. They are collections of pages, and it is imperative to remind the reader at many points that they are still reading or browsing the same object.

3.1. **Models and Techniques.** Statistics is about the properties of *statistical techniques*. Statistical techniques are tools to transform data (input) into representation (output) – presumably with the aim of a gain in clarity, communicability, stability, and so on.

Statistical techniques can be studied in terms of their *stability* under small variations of the input, and in terms of their performance in idealized situations, which are often called *models*.

In this interpretation of statistics the emphasis is strongly on the techniques. Models can be used to generate techniques, using some principle such as least squares or maximum likelihood, but this is just auxiliary. After this initial step is carried out, the technique still has to be evaluated. And not just on the model that was used to generate the technique, that would be cheating.

Thus in this interpretation, we do not care if models are *true*. It is not even clear what this means – the question is if models are useful, and they are if they generate good (stable, illuminating) techniques.

3.2. **Variables.** It is useful to develop statistics using the general concept of a variable. This is discussed in the (very minimal) first chapter of UCLA-EST, and also in some interesting publications by Donald Macnaughton [**?**].

3.3. **Inference.** The whole notion of statistical inference is murky. Of course, we want to make statements about the population from the sample, that's only natural. We want to interpolate and extrapolate, as in all of science, and as in many other of our activities as well. The problem starts if we want to justify this activity in rational or even logical terms. This has absolutely nothing to do with statistics.

We construct techniques, and we study their behaviour in idealized situations. We then apply the same techniques in real situations and we hope they work – which is a leap of faith. In some cases we can persuade our peers, or the public at large, or the jury, that our technique has worked and is better than the competition. In other cases we can only convince a very small group of our colleagues who are working on the same narrow topic.

4. Related Projects

4.1. Other UCLA Projects.

4.1.1. *The Journal of Statistical Software.* The *Journal of Statistical Software* [?] is an electronic journal published by UCLA Statistics. It has two main functions. In the first place it provides a service to the statistical community, because it provides useful statistical software in a central location. Here "useful" means both relevant, well-tested and well-documented. All three aspects are guaranteed by the review process.

The central location aspect is particularly important, because the RSS has stopped publishing algorithms in `Applied Statistics`. Of course there is a lot of statistical software in `statlib` [?], but it is poorly organized and in many cases very sparsely documented.

The second function of the journal is to provide a "respectable" outlet for authors of software and manuals, so that they can use these publications in their academic record. Again, the peer review process is critical here. The relation of JSS with UCLA-EST is clear. In the textbook we provide, among other things, a repository of instructional software for statistics. The code is usually available, or will be made available, but the emphasis is on *using* the programs over the net. In JSS we have a journal-type organization, i.e. a linear list of contributions ordered in time. Each unit is completely self contained, except perhaps by the classical mechanism of references. In UCLA-EST the organization is by topic, and hypertext takes care of the many links between different units.

4.1.2. *The UCLA Xlisp-Stat Archive.* Xlisp-Stat [?] is a statistical and graphics environment based on the Lisp language. It can be compared

with S-plus, but in contrast with S-plus it is both completely open and completely free. Xlisp-Stat also has better memory management, better dynamic graphics, and a much better extension language. At least that is what I think. It is a less complete and a less smooth product than S-plus, with less support, but potentially much more valuable.

UCLA Statistics maintains an archive of contributed software in Xlisp-Stat at `http://www.stat.ucla.edu/archive/xlisp-stat`. UCLA-EST uses many interactive graphical demos written in Xlisp-Stat. Otherwise, the differences are clear. Although the Xlisp-Stat Archive is organized by topic, the units are basically not connected, and often documented poorly. There is a subdirectory of the archive which deals specifically with statistics teaching (`statistics/introstat` in the archive), and this contains some well-developed environments, but basically without the (hyper)text.

4.1.3. *The UCLA Course Pages.* At `http://www.stat.ucla.edu/courses` there is a menu of statistics courses at UCLA. In many of them the instructors maintain a set of homepages with lecture notes, outlines, homework, chat rooms, case studies, and so on. Quite a few of the components are linked into portions of UCLA-EST, which shows another way in which the textbook can be used. For a particular course, the instructor uses only particular pieces by providing the links to the students.

There are also many courses outside UCLA which link parts of the textbook.

4.2. **Other EST projects.**

4.2.1. *The Chance Project.* The Chance Project [?], by Laurie Snell at Dartmouth, maintains the Chance Database. The Chance Database contains material useful for Chance Courses, which are *quantitative literacy courses based on current chance events in the news.* Clearly, the emphasis is on case-studies, and the Chance Database is a repository of case studies.

It is clear that UCLA-EST can use materials from the Chance Database, because ultimately case-studies and examples will have to be integrated in the textbook. Unfortunately, the news is not a very stable entity. A component explicitly defined in term of the news must be upgraded continuously, which is not very desirable, even in an on-line textbook.

4.2.2. *HyperStat.* Some time ago, David Lane pioneered electronic statistics hypertext by publishing HyperStat, written in Apple's HyperCard. This is now being converted to HTML [**?**].

HyperStat is quite close to UCLA-EST, in the sense that it is clearly a textbook. It is at an elementary level, and it is written, basically, to fit on a floppy. The emphasis is on the hypertext and the static graphics, there are no interactive calculators and demos.

4.2.3. *GASP.* The *Globally Accessible Statistical Procedures* [**?**] project originates with Webster West of the University of South Carolina. Its purpose is to make statistical procedures widely available over the net. There are basically two classes of procedures, which we call *calculators* and *demos* in UCLA-EST. In a calculator you enter input, often in a form, or by giving the program the URL of a dataset. You maybe also set some options. The program then returns the results to you, in the browser window, or maybe over email. A demo is usually meant for instruction – it illustrates statistical concepts graphically and often dynamically. GASP has CGI-based calculators and Java demos. They are just listed, and not organized in any way (except for the distinction we just discussed).

4.2.4. *Journal of Statistics Education.* The *Journal of Statistics Education* is an electronic journal, published by the statistics department at NCSU. It publishes papers on statistics education, in may cases with downloadable software, but as far as I know not interactive. Papers are written in HTML. JSE is closer to classical journals than JSS, because it works with volumes, and it is paper-oriented and not software oriented. JSS is a software archive in disguise, JSE is a journal in disguise. Obviously, material in JSE has been used as background for UCLA-EST, but nothing is used directly.

4.2.5. *Statlib.* `statlib` is an archive of many things that have to do with statistics. There is some attempt at organization, by bringing stuff under a large number of headings, but this makes it really into a number of archives. The amount of useful material in `statlib` is immense, but searching and browsing is a pain if you don't know precisely what you are looking for. A lot of the software is poorly documented, and there is no subclassification in terms of statistical topics or statistical activities (teaching, consulting, research).

5. Technical Considerations

5.1. **HTML versions.** HTML has been changing rapidly over the last two years. Fortunately, there is now some order in the HTML world.

One reasons is the rivalry between Netscape and Microsoft. If Netscape adds a feature to its browser, such as an HTML extension, then Microsoft will add the same feature pretty soon. Since their browsers control 90% of the market, this guarantees a *de facto* standard.

The actual standard, as usual, is much slower to develop. HTML 2.0 [?] is generally accepted, but it did not incorporate many of the Netscape extensions. HTML 3.0 was a promising development, but it tried to do too much at the same time. Thus it was abandoned. It seems that HTML 3.2 [?] is a reasonable standard, and we try to conform to this as much as possible.

5.2. **Graphics.** The common denominator for graphics on the WWW is still the `gif` format. Although sometimes `jpeg` graphics is used in UCLA-EST, if we construct graphics files we use `gif`. The basic tools on UNIX are `xfig` [?], a drawing program that can save its output as `gif` files, and `xv`, the graphics viewer and manipulator [?]. Many other useful tools are available, of course, on the Macintosh.

For the CGI programs that have to produce `gif` files we rely on Thomas Boutell's `gd` library [?], which can actually produce its `gifs` "on-the-fly". Thus they do not have to be stored anywhere on the system, we merely use tags such as

```
<IMG SRC="/cgi-bin/foo.cgi">
```

where `foo.cgi` is the program generating the `gif`.

Now that plug-ins become generally available (and portable over the main browsers) we can also at least think about using QuickTime movies. UCLA-EST has some QuickTime demos, written by Berrie Zielman (Netherlands).

5.3. **Client-side vs Server-Side.** If we start a program from the HTML page, then in some cases it is executed on the server (i.e. our machine), and in some cases on the client (i.e. your machine). Both options have advantages and disadvantages. If the program is executed on the server, then we do not have to assume anything about the client, and we can set things up so that they work correctly. But lots of users will cause a lot of load on our machine. And since the results have to be send over the network to the client, the response may be a bit slow. On the other hand, if the program is executed on your machine, it has to be downloaded first (in the case of a Java applet), and we are powerless to help you if your setup is not correct. Similarly, if your browser cannot do Java, JavaScript, client-pull, server-push, and so on, there is no way we can help.

Some aspects of this same problem will be discussed again in subsequent sections.

5.4. **Java and Javascript.** The three most popular browsers at the moment are Netscape's Navigator, Microsoft's Internet Explorer, and Apple's Cyberdog. All three understand Java, the first two understand JavaScript (Cyberdog will understand JavaScript soon, because it will use a version of Navigator as its WWW browser). Also, all three browsers can be extended by using plug-ins (which make more file formats available for display in the main browser window).

Thus it seems more and more the case that we can simply assume Java and Javascript are available to our clients. UCLA-EST does not really assume this yet, and warns the reader in the text that some demos will only work on some browsers. On the other hand, many very appealing applets are now available, written by Balasubraminian Narasimhan (Stanford), Webster West (SC), David Lane (Rice), Tony Rossini (SC), and others. If they are available on the net, they are incorporated in the textbook in the appropriate section.

5.5. **Helpers.** UCLA-EST, in its current form, relies rather heavily on Xlisp-Stat demos. This means that the person reading the textbook needs to have Xlisp-Stat installed on their local machines (and have it installed correctly). At the moment, we could run Xlisp-Stat on the server side, and have it display on the client machine, but this is a tricky process limited to clients running X11. We can also run Xlisp-Stat on the server side and have it display its text output in the browser window. This is actually done in

`http://www.stat.ucla.edu/cgi-bin/Xlisp-Stat.cgi`

Unfortunately we cannot do the same thing with graphical output, and only the client solution is open to us at the moment. One way around the dilemma is to rewrite the demos in Java, but it would be so much nicer if Xlisp-Stat could write is graphics output to a browser window.

5.6. **CGI.** UCLA-EST uses many CGI programs, mostly calculators. These are all written in C, using the `cgic` [?] library of Thomas Boutell. If the programs have to produce graphics, they do this using Boutell's `gd` [?] library. Most people think CGI programs should be written in perl, and only masochists use C. So be it. The most interesting CGI programs in the textbook are perhaps the probability distribution calculators in

`http://www.stat.ucla.edu/textbook/singles/describe_single/probmodels/calc.html`

All CGI processing takes place on the server-side, of course, which is one reason to use C instead of perl. Of course everything that can be done using CGI can be done with Java and JavaScript, but we do give up some portability. Observe that combining CGI and HTML forms even gives reasonable results in text-only browsers such as `lynx`.

5.7. **Preprocessing.** UCLA-EST has hundreds of HTML pages, even in its current incomplete form. It is important, both from the aesthetical point of view, and from the point of view that we want to emphasize the unity of the project, to make these pages look about the same. This is done by giving them identical headers and footers.

For the headers and footers we use the `pphtml` program of [?]. This preprocesses a file `foo.phtml` of the form

```
<!--%/bin/sh TB_HEAD ``A Title''-->
--- insert HTML here ---
<!--%/bin/sh TB_TAIL-->
```

After processing there will be a file `foo.html` in which the HTML comments have been replaced by the UCLA-EST header and footer. For completeness we give `TB_HEAD` and `TB_TAIL` here. They are

```
echo "<HTML>"
echo "<HEAD>"
echo "<TITLE>$1</TITLE>"
echo "</HEAD>"
echo "<BODY BGCOLOR="#FFFFFF">"
echo "<TABLE>"
echo "<TR>"
echo "<TD>"
echo "<IMG SRC=\"/graphics/stat.gif\" ALIGN=LEFT>"
echo "<TD VALIGN=BOTTOM><FONT SIZE=6>Statistics</FONT><BR>"
echo "<FONT SIZE=5>The Study of Stability in Variation</FONT>"
echo "</TABLE>"
echo "<HR>"
```

and

```
echo "<HR>"
echo "<CENTER>"
echo "[<A HREF=\"http://www.cas.lancs.ac.uk/glossary_v1.1/main.html\">"
echo "Statistics Glossary</A>]"
echo "[<A HREF=\"/textbook/textbookol.html\">Table of Contents</A>]"
echo "[<A HREF=\"/calculators/\">Statistics Toolbox</A>]<BR>"
echo "[<A HREF=\"/textbook/background.html\">Textbook Background</A>]"
echo "[<A HREF=\"/textbook/formulas/\">Formulas</A>]<BR>"
echo "[<A HREF=\"/textbook/\">Textbook Homepage</A>]"
echo "[<A HREF=\"/\">UCLA Statistics Homepage</A>]"
```

```
echo "</CENTER>"
echo "<HR>"
echo "<ADDRESS>"
echo "Textbook Editor: Jan de Leeuw<BR>"
echo "Email us at: "
echo "<A HREF=\"mailto:deleeuw@stat.ucla.edu\">deleeuw@stat.ucla.edu</A>"
echo "Document: http://www.stat.ucla.edu<!--#echo var=\"DOCUMENT_URI\"--><BR>"
echo "Visitors since <!--#exec cgi=\"/cgi-bin/counter-date\"--><BR>"
echo -n "Last revision: "
date
echo "</ADDRESS>"
echo "</BODY>"
echo "</HTML>"
```

5.8. **Equations.** Incorporating equations into HTML pages is a bit of a problem. There is a slew of possibilities, but there does not seem to be a perfect solution to the problem yet. One possibility, and one that many people have been waiting for a long time, is the `<MATH></MATH>` tags in HTML 3.0, which allowed for LaTeX-like commands directly in the HTML. But these tags were implemented only in experimental browsers such as `Arena` [?], and the HTML 3.0 standard was abandoned recently.

A second possibilty is to render your pages to either Postscript or `pdf`, and to display the page with a plug-in (for `pdf`) or helper (for Postscript). This assumes that the client is set up correctly to deal with these formats, and in the case of Postscript, it does away with all hypertext features. We can use hypertext links in `pdf`, but this requires use of commercial software, and too much heavy machinery for my taste.

There are basically three ways to get equations into your HTML documents in the common browsers. The first one is to use Java applets. This is done by `EqnViewer` [?] and by `WebEQ` [?]. Secondly, we can use a CGI program to preprocess the HTML page and render the equations. The prime example here is `MINSE` [?]. A finally we can preprocess the equations to `gif` files, and incorporate them into the HTML. This is the most portable procedure, and it is the one used in UCLA-EST (although in the future we might switch to Java).

One brute force way to deal with equations is to write a document in LaTeX, and then use the `latextohtml` translator [?]. The translator translates the document into a number of linked HTML pages, and the equations are translated to `gifs`. Although this works well for some manuscripts, in other cases it produces chaos. We prefer to have more control. For this control, we once again use `pphtml`.

We can preprocess `phtml` pages which contain HTML comments of the form

```
<!--%TEX
\begin{multline*}
\huge
\begin{bmatrix}
A & B\\
B' & C
\end{bmatrix}^{-1}=
\\
\end{multline*}
-->
```

Here `TEX` is a shell script which contains

```
#!/bin/sh
DOC=$PPIDOC
NUM=$PPINUM
cat > $DOC-$NUM.tex << HEAD
\documentclass[12pt]{amsart}
\usepackage{amssymb}
\usepackage{uclastat,verbatim,float}
\unitlength 0.2in
\thispagestyle{empty}
\begin{document}
\noindent
HEAD
cat >> $DOC-$NUM.tex
cat >> $DOC-$NUM.tex << FOOT
\end{document}
FOOT
latex $DOC-$NUM.tex > .latex-errors
dvips $DOC-$NUM.dvi > $DOC-$NUM.ps
pstogif -out $DOC-$NUM.GIF $DOC-$NUM.ps > .pstogif-errors
giftrans -t 1 -b 0 $DOC-$NUM.GIF > $DOC-$NUM.gif
rm $DOC-$NUM.tex $DOC-$NUM.dvi $DOC-$NUM.ps
rm $DOC-$NUM.aux $DOC-$NUM.log $DOC-$NUM.GIF
echo "<IMG SRC=\"$DOC-$NUM.gif\" ALT=\"$DOC-$NUM\">"
```

After preprocessing the file `foo.phtml`, the comments will have been replaced by

```
<IMG SRC=''foo-1.gif'' ALT=''foo-1''>
```

and the file `foo-1.gif` will be in the same directory as `foo.html`.

References

[1] Ka-Ping Yee, *Mathematics has arrived on the Web at last...*
http://www.lfw.org/ping/

[2] Philip Thrift, *Preprocessing instructions: Embedding external notations in HTML.*
http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/DDay/thrift/PPI.html

[3] Nikos Drakos, *All About LaTeX2HTML.*
http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html/latex2html

[4] *Journal of Statistical Software,*
http://www.stat.ucla.edu/journals/jss/

[5] Luke Tierney, *Lisp-Stat Information*,
http://stat.umn.edu/~ luke/xls/xlsinfo/xlsinfo.html

[6] statlib,
http://lib.stat.cmu.edu/

[7] *WebEQ Equation Rendering*,
http://www.geom.umn.edu/software/WebEQ/

[8] *EqnViewer on the Web*,
http://www.hookup.net/~ cbazza/EqnViewer.html

[9] *Arena*,
http://www.w3.org/pub/WWW/Arena/

[10] *HTML 2.0 Proposed Standard Materials*,
http://www.w3.org/pub/WWW/MarkUp/html-spec/

[11] Dave Raggett, *HTML 3.2 Reference Specification*,
http://www.w3.org/pub/WWW/TR/WD-html32.html

[12] *Globally Accessible Statistical Procedures*,
http://www.stat.sc.edu/rsrch/gasp/

[13] *Journal of Statistics Education* http://www2.ncsu.edu/ncsu/pams/stat/info/jse/homepage.html

[14] David Lane, *Overview of HyperStat*,
http://www.ruf.rice.edu/~ lane/hyperstat/overview.html

[15] *The Chance Database Welcome Page*,
http://www.geom.umn.edu/docs/snell/chance/welcome.html

[16] *Xfig 3.1.4 Release Notes*,
http://www.madness.net/~ vince/software/SGI_freeware_CD/relnotes/xfig.html

[17] John Bradley, *Note on XV* http://www.sun.com/sunsoft/catlink/xv/note.html

[18] Thomas Boutell, *cgic: an ANSI C library for CGI Programming*,
http://www.boutell.com/cgic/

[19] Thomas Boutell, *gd 1.2. A graphics library for fast GIF creation*,
http://www.boutell.com/gd/

[20] Donald Macnaughton, *The Entity-Property-Relationship Approach to Statistics: An Introduction for Students*,
http://www.hookup.net/ donmac/

UCLA Statistics Program, 405 Hilgard Avenue, Los Angeles, CA 90095-1554

E-mail address: deleeuw@stat.ucla.edu