Reusable Documents

Paul Murrell

The University of Auckland

NZSA September 1-2 2008

Motivation

Developing online exercises for a book
"Introduction to Data Technologies"
`http://www.stat.auckland.ac.nz/~paul/ItDT/`

- Multiple formats: HTML, PDF
- Multiple versions: with(out) solutions
- Multiple views: subset by topic or level of difficulty
- Multiple kinds of content: text, data sets, computer code

Motivation

- I want to write a **single** set of **source** files
- I want to **process** the source files in many different ways
- I want **others** to be able to access and process the source files
    - not just about **access**, but also involves using **standard tools**
- I want the **output** of code to be automatically **consistent** with the **code** itself.

The Indian Mothers data set

```
cord1 cord2 ... cord14 age edu alive middle poor work
2 1 1 NA NA NA NA NA NA NA NA NA NA NA 30 0 3 0 1 1
2 NA NA NA NA NA NA NA NA NA NA NA NA NA 32 0 1 0 1 0
2 1 1 1 2 NA NA NA NA NA NA NA NA NA 28 0 5 0 1 1
2 1 1 2 NA NA NA NA NA NA NA NA NA NA 39 0 4 1 0 0
1 1 NA NA NA NA NA NA NA NA NA NA NA NA 20 0 2 1 0 0
2 1 1 NA NA NA NA NA NA NA NA NA NA NA 25 0 3 1 0 1
1 1 1 NA NA NA NA NA NA NA NA NA NA NA 22 0 3 0 1 1
2 2 2 NA NA NA NA NA NA NA NA NA NA NA 35 6 3 1 0 0
1 2 NA NA NA NA NA NA NA NA NA NA NA NA 21 0 2 0 1 0
1 2 2 2 NA NA NA NA NA NA NA NA NA NA 35 0 4 0 1 1
...
```

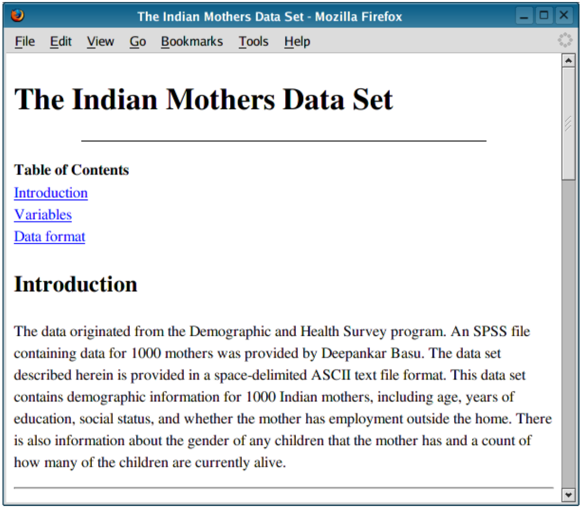
india.txt

The Indian Mothers data set

The aim is to develop several documents that describe the data set, provide the raw data, and provide exercises that make use of the data set.

- A formal (machine-readable) description of the data set.
- A general (human-readable) description of the data set.
- A set of exercises that explore how to work with the Indian Mothers data set in R.
- A set of exercises that explore the data storage options for the Indian Mothers data set.

indianMothers.html


The Indian Mothers Data Set - Mozilla Firefox
File Edit View Go Bookmarks Tools Help

The Indian Mothers Data Set

Table of Contents

Introduction

Variables

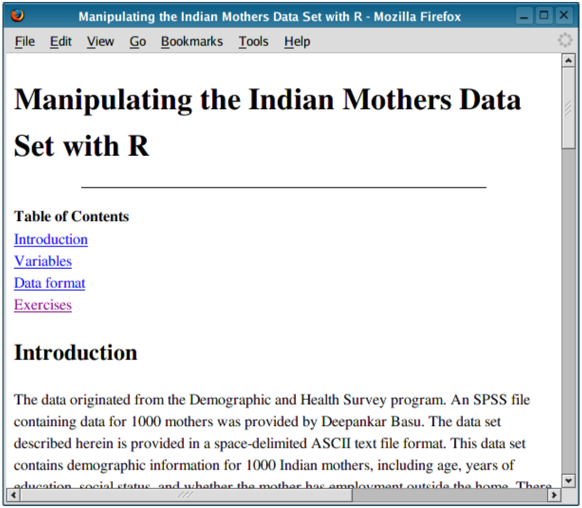
Data format

Introduction

The data originated from the Demographic and Health Survey program. An SPSS file containing data for 1000 mothers was provided by Deepankar Basu. The data set described herein is provided in a space-delimited ASCII text file format. This data set contains demographic information for 1000 Indian mothers, including age, years of education, social status, and whether the mother has employment outside the home. There is also information about the gender of any children that the mother has and a count of how many of the children are currently alive.

General information on data set.

indianMothers+R.html


Manipulating the Indian Mothers Data Set with R - Mozilla Firefox
File Edit View Go Bookmarks Tools Help

Manipulating the Indian Mothers Data Set with R

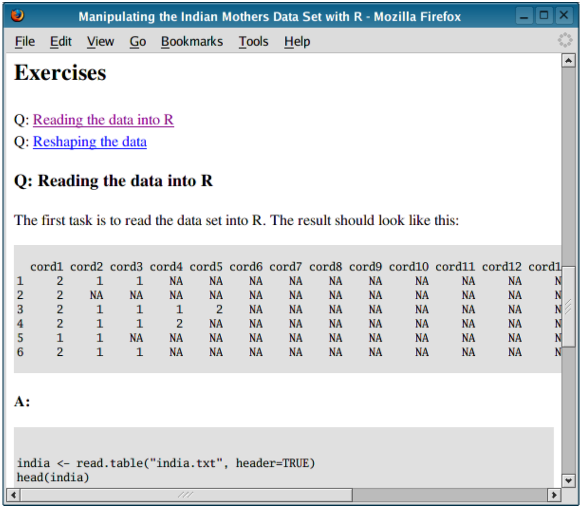
Table of Contents

Introduction

The data originated from the Demographic and Health Survey program. An SPSS file containing data for 1000 mothers was provided by Deepankar Basu. The data set described herein is provided in a space-delimited ASCII text file format. This data set contains demographic information for 1000 Indian mothers, including age, years of education, social status, and whether the mother has employment outside the home. There

Exercises repeat data set information.

indianMothers+R.html


Manipulating the Indian Mothers Data Set with R - Mozilla Firefox
File Edit View Go Bookmarks Tools Help

Exercises

Q: Reading the data into R
Q: Reshaping the data

Q: Reading the data into R

The first task is to read the data set into R. The result should look like this:

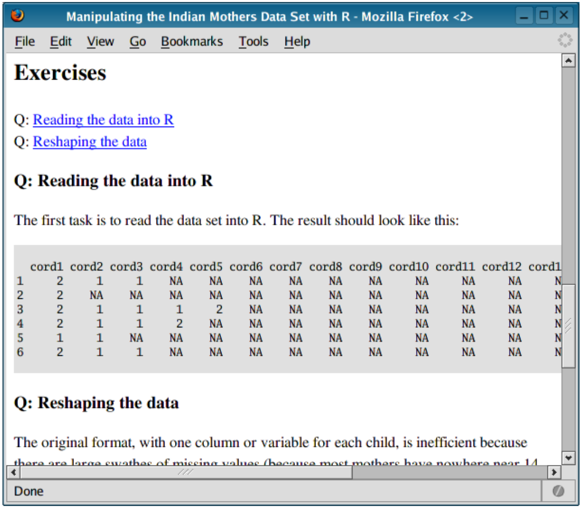
```
  cord1 cord2 cord3 cord4 cord5 cord6 cord7 cord8 cord9 cord10 cord11 cord12 cord1
1     2     1     1    NA    NA    NA    NA    NA    NA     NA     NA     NA     N
2     2    NA    NA    NA    NA    NA    NA    NA    NA     NA     NA     NA     N
3     2     1     1     1     2    NA    NA    NA    NA     NA     NA     NA     N
4     2     1     1     2    NA    NA    NA    NA    NA     NA     NA     NA     N
5     1     1    NA    NA    NA    NA    NA    NA    NA     NA     NA     NA     N
6     2     1     1    NA    NA    NA    NA    NA    NA     NA     NA     NA     N
```

A:

```
india <- read.table("india.txt", header=TRUE)
head(india)
```

Exercises.

indianMothers+R-Qonly.html


Manipulating the Indian Mothers Data Set with R - Mozilla Firefox <2>
File Edit View Go Bookmarks Tools Help
Done

Exercises

Q: Reading the data into R
Q: Reshaping the data

Q: Reading the data into R

The first task is to read the data set into R. The result should look like this:

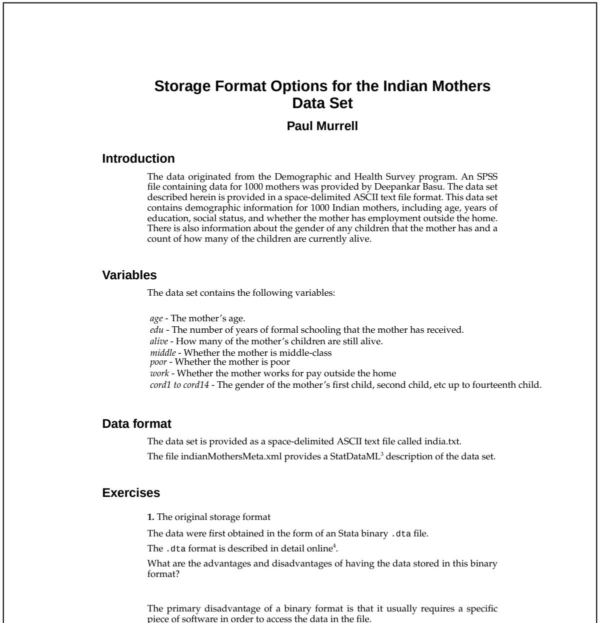
```
  cord1 cord2 cord3 cord4 cord5 cord6 cord7 cord8 cord9 cord10 cord11 cord12 cord1
1     2     1     1    NA    NA    NA    NA    NA    NA     NA     NA     NA     N
2     2    NA    NA    NA    NA    NA    NA    NA    NA     NA     NA     NA     N
3     2     1     1     1     2    NA    NA    NA    NA     NA     NA     NA     N
4     2     1     1     2    NA    NA    NA    NA    NA     NA     NA     NA     N
5     1     1    NA    NA    NA    NA    NA    NA    NA     NA     NA     NA     N
6     2     1     1    NA    NA    NA    NA    NA    NA     NA     NA     NA     N
```

Q: Reshaping the data

The original format, with one column or variable for each child, is inefficient because there are large swathes of missing values (because most mothers have nowhere near 14

Exercises with no solutions shown.

indianMothers+Format.pdf



Storage Format Options for the Indian Mothers Data Set

Paul Murrell

Introduction

The data originated from the Demographic and Health Survey program. An SPSS file containing data for 1000 mothers was provided by Deepankar Basu. The data set described herein is provided in a space-delimited ASCII text file format. This data set contains demographic information for 1000 Indian mothers, including age, years of education, social status, and whether the mother has employment outside the home. There is also information about the gender of any children that the mother has and a count of how many of the children are currently alive.

Variables

The data set contains the following variables:

age - The mother's age.
edu - The number of years of formal schooling that the mother has received.
alive - How many of the mother's children are still alive.
middle - Whether the mother is middle-class
poor - Whether the mother is poor
work - Whether the mother works for pay outside the home
cord1 to cord14 - The gender of the mother's first child, second child, etc up to fourteenth child.

Data format

The data set is provided as a space-delimited ASCII text file called india.txt.

The file indianMothersMeta.xml provides a StatDataML[3] description of the data set.

Exercises

1. The original storage format

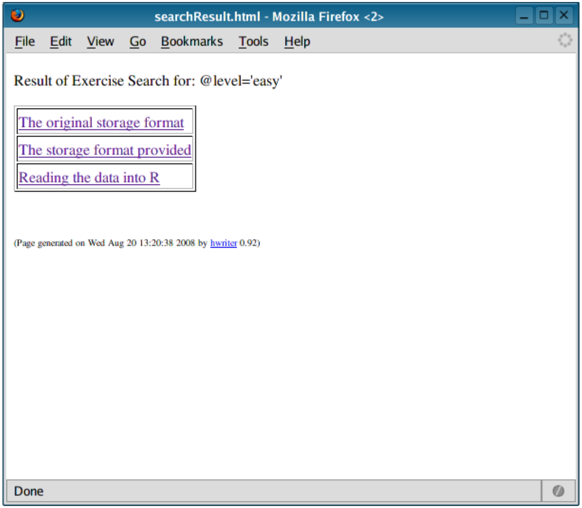
The data were first obtained in the form of an Stata binary `.dta` file.

The `.dta` format is described in detail online[4].

What are the advantages and disadvantages of having the data stored in this binary format?

The primary disadvantage of a binary format is that it usually requires a specific piece of software in order to access the data in the file.

searchResult.html


searchResult.html - Mozilla Firefox <2>
File Edit View Go Bookmarks Tools Help

Result of Exercise Search for: @level='easy'

The original storage format
The storage format provided
Reading the data into R

(Page generated on Wed Aug 20 13:20:38 2008 by hwriter 0.92)

Find all "easy" exercises.

Solutions

- Proprietary format (GUI)

Solutions

- ~~Proprietary format (GUI)~~
    - Uncollegial
    - **Source** is secondary (at best)
    - A dead-end

Solutions

- ~~Proprietary format (GUI)~~
    - Uncollegial
    - **Source** is secondary (at best)
    - A dead-end
- LaTeX

Solutions

- ~~Proprietary format (GUI)~~
  - Uncollegial
  - **Source** is secondary (at best)
  - A dead-end
- ~~LaTeX~~
  - Code and output disconnected
  - Conversion to HTML flaky

Solutions

- ~~Proprietary format (GUI)~~
  - Uncollegial
  - **Source** is secondary (at best)
  - A dead-end
- ~~LaTeX~~
  - Code and output disconnected
  - Conversion to HTML flaky
- Sweave et al.

Solutions

- ~~Proprietary format (GUI)~~
    - Uncollegial
    - **Source** is secondary (at best)
    - A dead-end
- ~~LaTeX~~
    - Code and output disconnected
    - Conversion to HTML flaky
- ~~Sweave et al.~~
    - “Only” distinguishes between code and text

Solutions

- ~~Proprietary format (GUI)~~
  - Uncollegial
  - **Source** is secondary (at best)
  - A dead-end
- ~~LaTeX~~
  - Code and output disconnected
  - Conversion to HTML flaky
- ~~Sweave et al.~~
  - "Only" distinguishes between code and text
- XML + XPath + XInclude + XSLT + Sxslt

Solutions

- ~~Proprietary format (GUI)~~
  - Uncollegial
  - **Source** is secondary (at best)
  - A dead-end
- ~~LaTeX~~
  - Code and output disconnected
  - Conversion to HTML flaky
- ~~Sweave et al.~~
  - “Only” distinguishes between code and text
- **XML + XPath + XInclude + XSLT + Sxslt**
  - **Semantic** markup
  - Flexible, powerful and **extensible** transformations
  - Modern open standard

indianMothersMeta.xml

```
<?xml version="1.0" encoding="utf-8"?>
<StatDataML>

  <description>
    <title>Demographics for 1000 Indian Mothers</title>
    <source>Deepankar Basu</source>
    <date>2007-10-30</date>
    <comment>
      The data originated from the Demographic and Health Survey
      program.  An SPSS file containing data for 1000 mothers
      was provided by Deepankar Basu.  The data set described herein
      is provided in a space-delimited ASCII text file format.

      This data set contains demographic information for
      1000 Indian mothers, including
      age, years of education, social status, and whether the
...
```

indianMothersMeta.xml

indianMothers.xml

```
<?xml version="1.0" encoding="utf-8"?>
<article xmlns:xi="http://www.w3.org/2001/XInclude">
  <articleinfo>
    <title>The Indian Mothers Data Set</title>
    <author><firstname>Paul</firstname><surname>Murrell</surname></author>
  </articleinfo>

  <section>
    <title>Introduction</title>
    <para>
      <xi:include href="indianMothersMeta.xml"
                  parse="xml"
                  xpointer="xpointer(/StatDataML/description/comment/text())"
    </para>
  </section>

...
```

indianMothers.xml

XInclude

For the inclusion to take place, the XML file must be processed. Most XSLT processors will do this job.

```
xsltproc -o indianMothers.docbook --xinclude
         indianMothers.xsl indianMothers.xml
```

The file `indianMothers.docbook` now contains copies of the information from `indianMothersMeta.xml`.

What's XSLT? Glad you asked ...

XSLT

The translation from `indianMothersTemplate.xml` to `indianMothers.docbook` involved more than just XIncludes.

The variable type information was also transformed using XSLT templates from `indianMothers.xsl`.

```
<xsl:template match="seglistitem/type">
  <xsl:apply-templates />
</xsl:template>

...

<xsl:template match="seglistitem/type/numeric/integer">
  <seg>
    integer (min: <xsl:value-of select="min" />)
  </seg>
</xsl:template>
```

indianMothers.xsl

XSLT

Original elements from `indianMothersMeta.xml`:

```
<type>
  <numeric>
    <integer>
      <min>0</min>
    </integer>
  </numeric>
</type>
```

Transformed elements in `indianMothers.docbook`:

```
<seg>
  integer (min: 0)
</seg>
```

DocBook

The file `indianMothers.docbook` is a DocBook file.

The advantage of transforming the document into DocBook is that predefined stylesheets are provided for transforming DocBook into a variety of formats.

This command produces `indianMothers.html`:

```
docbook2html -u indianMothers.docbook
```

This command produces `indianMothers.pdf`:

```
docbook2pdf -d mystyle.dsl indianMothers.docbook
```

These tools actually use DSSSL stylesheets and the jade processor, but XSLT and xsltproc equivalents also exist.

The Sxslt package

With **Sxslt**, we can create literate documents with embedded code chunks.

```
<xsl:template match="chunk[@lang='R']">
  ...
  <xsl:if test="not(@eval) or @eval = 'true'">
    <xsl:choose>
      <xsl:when test="@results and @results = 'hide'">
          <xsl:value-of select="r:eval(.)" />
      </xsl:when>
      <xsl:otherwise>
        <programlisting>
          <xsl:value-of select="r:evalWithOutput(.)" />
        </programlisting>
      </xsl:otherwise>
    </xsl:choose>
  </xsl:if>
  ...
</xsl:template>
```

exercises2docbook.xsl

The **Sxslt** package

```
    <chunk name="numChildren" lang="R"
           echo="false"><![CDATA[
    maxChild <- aggregate(as.numeric(indiaLong$born),
                          list(max=indiaLong$id), max)
    table(maxChild$x)
    ]]></chunk>
```

indianMothers+R.xml

```
    <programlisting>
  1   2   3   4   5   6
 61 447 296 125  52  19
    </programlisting>
```

indianMothers+R.docbook

The **XML** package

With the documents marked up as XML, it is possible to process **any** subset of the document.

The following code uses the **XML** package in R to find all exercises of a specific level of difficulty, then obtains the `id` attributes from those exercises.

```
indiaDoc <- xmlTreeParse("indianMothers+R.xml",
                         useInternalNodes=TRUE)
exercises <- getNodeSet(indiaDoc,
                        "//exercise[@level='easy']")
sapply(exercises, xmlGetAttr, "id")
```

This is the basis for generating `searchResult.html` (using the **hwriter** package to generate HTML output).

The **XML** package

```
<exerciseset>
  <exercise id="indianMothers-R-import"
            level="easy" topic="import">
    <title>Reading the data into R</title>

    ...

  <exercise id="indianMothers-R-reshape"
            level="medium" topic="R+reshape">
    <title>Reshaping the data</title>
```

indianMothers+R.xml

```
<table border="1">

   ...

<tr><td>
  <a href="indianMothers+R.html#indianMothers-R-import">
    Reading the data into R
  </a>
</td></tr>
</table>
```

researchResult.html

Final Thoughts

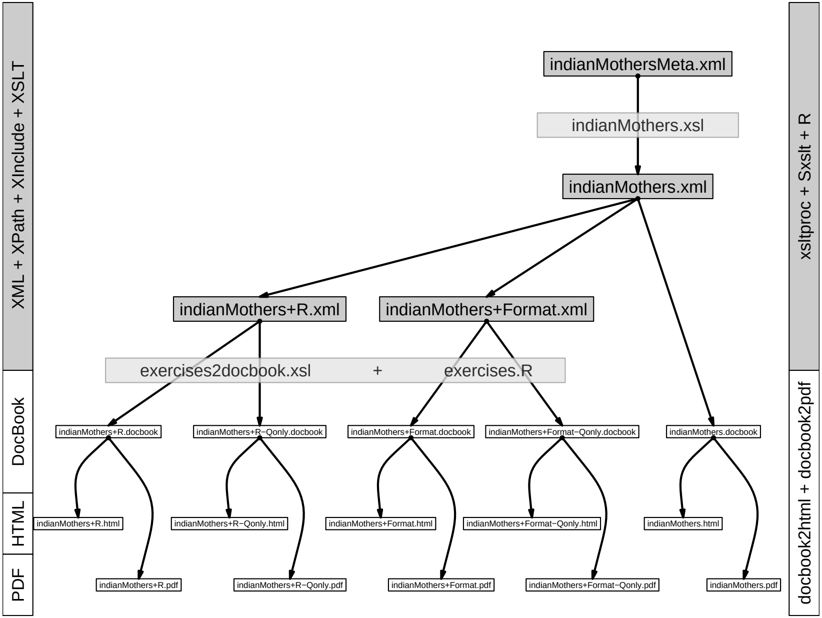
- Open standards
- Source is primary
- Literate documents
- **Everything** is marked up (not just code and text)
- Markup is for **structure** not presentation

Acknowledgements

- Duncan Temple-Lang (the **XML** and **Sxslt** packages) `http://www.omegahat.org/Sxslt/`
- Robert Gentleman (Bioconductor Compendiums) `http://www.bioconductor.org/docs/papers/2003/Compendium/`
- Friedrich Leisch (Sweave) `http://www.statistik.lmu.de/~leisch/Sweave/`
- Max Kuhn (odfWeave) `http://cran.r-project.org/web/packages/odfWeave/index.html`
- Russell Lenth (SASweave and StatWeave) `http://www.stat.uiowa.edu/~rlenth/StatWeave/`
- Tony Rossini (Literate Data Analysis) `http://www.bepress.com/cgi/viewcontent.cgi?article=1017&context=uwbiostat`
- The World Wide Web Consortium (W3C) Recommendations (XML, XPath, XInclude, and XSLT) `http://www.w3.org/`
- Daneil Veillard (libxml2 and libxslt) `http://xmlsoft.org/`
- Gergoire Pau (the **hwriter** package) `http://cran.r-project.org/web/packages/hwriter/index.html`

Technology Summary

- XML: markup **everything**
- XPath: specify **any** XML subset
- XInclude: reuse **any** XML subset
- XSLT: transform **any** XML subset to **anything**
- Sxslt: include **R code** in XSLT


XML + XPath + XInclude + XSLT
DocBook
HTML
PDF
xsltproc + Sxslt + R
docbook2html + docbook2pdf
indianMothersMeta.xml
indianMothers.xsl
indianMothers.xml
indianMothers+R.xml
indianMothers+Format.xml
exercises2docbook.xsl + exercises.R
indianMothers+R.docbook
indianMothers+R−Qonly.docbook
indianMothers+Format.docbook
indianMothers+Format−Qonly.docbook
indianMothers.docbook
indianMothers+R.html
indianMothers+R−Qonly.html
indianMothers+Format.html
indianMothers+Format−Qonly.html
indianMothers.html
indianMothers+R.pdf
indianMothers+R−Qonly.pdf
indianMothers+Format.pdf
indianMothers+Format−Qonly.pdf
indianMothers.pdf