Introduction to the `muRL` Package
Mailmerge using `R`, L^AT_EX, and the Web
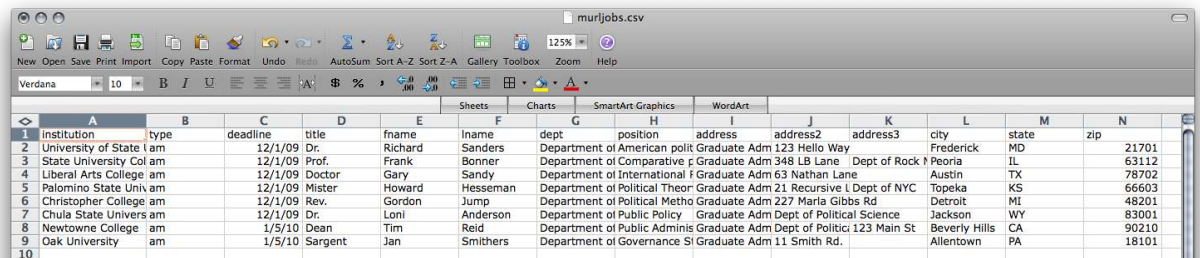
Ryan T. Moore[*]   Andrew Reeves[†]

July 17, 2009

This document describes how to use `muRL` to create beautiful mailmerged letters typeset in `R`. In it we:

1. Prepare the jobs spreadsheet;

2. Create the T_EX file using `muRL` and `R`;

3. Customize the T_EX file; and

4. Compile the T_EX file.

Be sure to include all of the files in the same directory or else point the input commands appropriately. In this example, the user's directory is /Users/JohnnyF/jobletter/.

1   Preparing the jobs spreadsheet

In the spreadsheet application of your choosing, enter the mailmerge information. The spreadsheet pictured below is included with `muRL` and is used in the example here.

---

[*]Assistant Professor, Department of Political Science, Washington University in St. Lous, 241 Seigle Hall, Campus 1063, One Brookings Drive, St. Louis MO 63130; `http://rtm.wustl.edu`; `rtm@wustl.edu`; 314/935-3461.

[†]Assistant Professor, Department of Political Science; 232 Bay State Road, Boston University, Boston MA 02215; `http://people.bu.edu/areeves`; `areeves@bu.edu`; 617/353-5284.

2   Creating the TEX file using `muRL` and `R`

In `R`:

```
> library(muRL)
Loading required package: maps
> murljobs <- read.murl("/Users/JohnnyF/jobletter/jobs.csv")
```

The jobs spreadsheet is now loaded into `R` and ready for the mailmerge. First, we use `zip.plot` to plot the zip codes of our addresses on a map of the U.S. Figure 1 presents that map.

```
> zip.plot(murljobs)
```
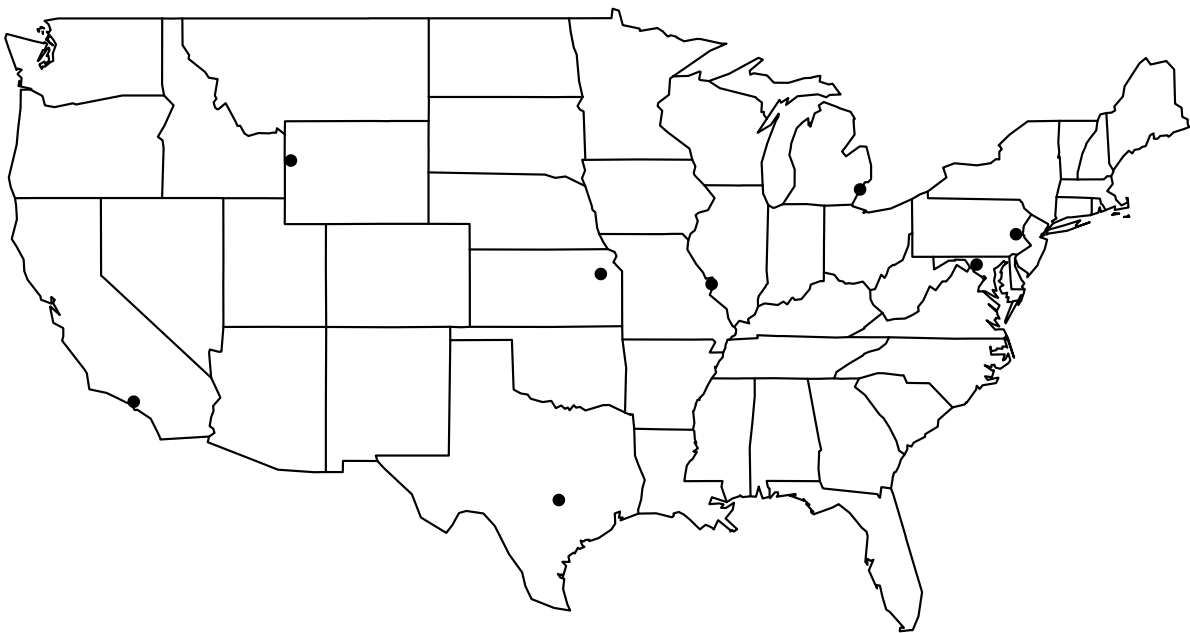

Figure 1: Location of mailmerge addresses (by ZIP code)

Since our jobs spreadsheet is already a `muRL` object in `R`, we can use the `write.murl` command to implement a mailmerge. The result will be a TEX file, which will need to be compiled. The file is generated in the working directory (see `getwd()`) of `R`.

```
> getwd()
[1] "/Users/JohnnyF/jobletter"
> write.murl(murljobs)
Data stored as file 'mailmerge.tex'.
The current working directory is /Users/JohnnyF/jobletter
```

3   Customizing the TₑX file

Using `write.murl`, we create a TₑX file (mailmerge.tex is the default filename) in the working directory of `R`. Open this file using a text editor like emacs. Like any TₑX file, mailmerge.tex is highly customizable, and the TₑX file contains many annotations to aid the user. The user can specify formatting options like the margin or the font size as well as content like date, the signature, or the text of the letter. For both sets of options, defaults are used if they are not specified by the user.

In the body of mailmerge.tex we provide descriptions of various fields that the user can change if they did not already do so using `write.murl`. In the example below, we provide a default date of March 1, 1972. It is unlikely that a user will use this date and so they may replace it with the date of their choosing or use the `\today` command to use today's date.

```
\date{March 1, 1972} %alternatively, can use \date{\today} to specify today's date.
\signature{Johnny Fever} %your name, which will follow the valediction.
```

In addition to the date and signature, the user will almost certain need to specify the return address and the body of the letter (see the help file for `write.murl` for further details).

Of special interest for the mailmerge is the use ability to create labels using the `envlab` package. By default, the package is used and it generates labels based on the Avery® 5164 label format. These labels include barcodes for ZIP codes ensuring prompt and accurate delivery.

4   Compiling the TₑX file

The final step is to compile the TₑX file. Once `mailmerge.tex` is to your specification, you can compile the file. This will result in mailmerged letters typeset by LaTeX. For example, if you use pdfLaTeX, the results with be a pdf document of the letters and mailing labels.