Web Appendix 1:
Automatically Creating a Spreadsheet from APSA eJobs Listings with muRL

Ryan T. Moore∗ Andrew Reeves†
March 7, 2011

The muRL package can scrape job listings posted on the American Political Science Association (APSA)’s website (called eJobs) and import them directly into a spreadsheet. This allows for easy integration into customized letters created by muRL or any other mail-merge protocol you choose.

This document describes how to import eJobs from the APSA website. The process involves downloading the html files for the jobs to which you plan to apply, opening R, and running apsahtml2csv, a function in the muRL package.

∗Assistant Professor, Department of Political Science, Washington University in St. Louis, 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.
1 Downloading eJobs

1. Go to http://www.apsanet.org and login. You must be a member of APSA in order to access eJobs.
2. Go to the eJobs link and click on it.
3. At the eJobs interface, enter your search criteria. In the search below, we limit our search to assistant professor positions in Comparative Politics posted for all dates.
4. The search returns the following results. We click on the search results for Iona College.

<table>
<thead>
<tr>
<th>ID</th>
<th>Position Title and Institution</th>
<th>Date Posted</th>
<th>Salary Range</th>
<th>Subfields</th>
<th>Date Posted and Valid Through</th>
<th>Deadline</th>
</tr>
</thead>
<tbody>
<tr>
<td>17376</td>
<td>Assistant Professor, Iona College</td>
<td>02/03/2011</td>
<td>Negotiable</td>
<td>Public Policy, Comparative Politics</td>
<td>02/03/2011 through 03/15/2011</td>
<td>03/15/2011</td>
</tr>
<tr>
<td>17376</td>
<td>Assistant Professor, Iona College</td>
<td>02/03/2011</td>
<td>Negotiable</td>
<td>Public Policy, Comparative Politics</td>
<td>02/03/2011 through 03/15/2011</td>
<td>03/15/2011</td>
</tr>
</tbody>
</table>
5. We decide that we would like to apply for the job at Iona College.
6. Now save the Iona College job listing as an html file. From your web browser’s menu bar, select “Save Page As” or “Save As”, and save only the page’s source html code into a directory. The file should be saved in a folder dedicated to all eJobs listings you wish to import to the jobs spreadsheet. You can name the file any unique name of their choosing. We recommend that you maintain the .html file extension for each file.1

7. Repeat the previous step until you have downloaded all of the eJobs that you would like to import into the spreadsheet. Below is the folder containing all of the eJobs for which we plan to apply. Again, note that you can specify any name, although the files should use a consistent file extension.

2 Importing eJobs into muRL

Once you have downloaded eJobs onto your local machine in a specified folder, you can then use muRL to scrape these html files and create a spreadsheet. Once you have loaded the muRL package in R, you can use the apsahtml2csv function to generate the spreadsheet. The function takes arguments for the location of the directory containing the saved eJobs html files, the extensions of those files (e.g. .html or .htm) and the

1In apsahtml2csv, the function searches for .html and .htm files by default in the (user specified) jobs folder. See the help file for apsahtml2csv for further information.
The jobs.csv file is a spreadsheet containing the details of each job position downloaded into the eJobs directory by the user. The fields include the APSA listing ID, date the job was posted, type of institution, title of the position, start date, salary, region, name of institution, contact person, address, phone, and description of the job. Note that eJobs has no standard method of entering the contact person, so the user must make the appropriate adjustments. For example, in the listing, the name of the contact person may appear as Dr. Mary Smith, Mary Smith, Mary J. Smith, PhD, or Professor M. Smith. We recommend creating additional fields of title, fname, lname and manually filling these fields based on the information in the contact field. If the user prefers other field names, muRL’s read.murl function should be used to process the spreadsheet.

3 Creating a Mail-merge

Once the eJobs spreadsheet has been created using apsahtml2csv, we use muRL to create a T\TeX file. Again, we manually add the title (title), first name (fname), and last name (lname) fields based on the contact field scraped from eJobs. The following lines of code generate the T\TeX file which can be compiled to create the typeset letters. They also report the name of the T\TeX file and the current working directory:

> murljobs <- read.murl("myjobs.csv")
> write.murl(murljobs)
Data stored as file ‘mailmerge.tex’.
The current working directory is /Users/JohnnyP/Desktop

Last, compile the T\TeX file into a PDF or PS file of letters and mailing labels.

---

2See the full documentation’s help file for apsahtml2csv for more information.