The Political Methodologist

Newsletter of the Political Methodology Section
American Political Science Association
Volume 16, Number 1, Fall 2008

Editors:
Paul M. Kellstedt, Texas A&M University
kellstedt@polisci.tamu.edu

David A.M. Peterson, Texas A&M University
dave@polisci.tamu.edu

Guy D. Whitten, Texas A&M University
whitten@polisci.tamu.edu

Editorial Assistant:
Mark D. Ramirez, Texas A&M University
mdramir@polisci.tamu.edu

Contents

Notes From the Editors

This edition of *The Political Methodologist* provides a range of advice from scholars on the cutting edge of the field.

In recent years there have been a number of prominent calls from political methodologists for the increased use of graphical tools to interpret the results of our models. Since the goal of such tools is to help readers make substantive inferences, it is important that we know what types of graphs readers do and do not interpret well. Christopher Zorn provides a valuable comparison of the accuracy of inferences drawn by readers from two commonly-used types of three dimensional plots.

Every day more data become available on the web. The obvious question from political methodologists is, "How can I get it quickly and in good working order?" Holger Döring provides a range of helpful answers to this question in a highly useful tour of scripting languages with an emphasis on Python.

As political methodology has evolved into a major component of the discipline, more departments have decided that they need to hire in this field. This is obviously a good thing for the readers of *The Political Methodologist*, but, as many of these same readers know, the junior faculty member hired as a methodologist faces a range of rather unique professional challenges. To help them meet these challenges we now have a new section of this newsletter titled "Professional Development." Corrine McConnaughy introduces this new section that is followed by the first set of questions posed by anonymous junior colleagues and answers provided by Gary King.

In our book review section, Robert Walker provides a thorough and insightful review of Kunio Takezawa's *Introduction to Nonparametric Regression.*

Earlier this summer the University of Michigan hosted the 25th Summer Meeting of the Society for Political Methodology. The host committee–Rob Franzese (chair), Nancy Burns, Bill Clark, Liz Gerber, John Jackson, Don Kinder, and Walter Mebane—did an excellent job of hosting the largest-ever meeting of the society. In our Announcements section Suzanna Linn, the chair of the committee charged with selecting the best graduate poster at the meeting, announces this year's winner. Last, but certainly not least, Janet Box-Steffensmeier provides the text from the citation of John Jackson as the second winner of the Political Methodology Career Achievement Award.

The Editors

Articles

Surface vs. Contour Plots for the Presentation of Three-Dimensional Data

Christopher Zorn
Pennsylvania State Univeristy
zorn@psu.edu

Political scientists are increasingly aware of the power of graphical displays in conveying empirical findings. In addition to their use as descriptive tools, figures are commonly used to illustrate the results of a wide variety of models and model specifications, including comparative statics, parameter estimates and confidence intervals, predicted values, and so forth (see, *inter alia*, Brambor et al. 2006; Gelman et al. 2002; Jacoby 2006; Kastellec and Leoni 2007). In addition, the availability of software for creating and managing such graphical displays has increased rapidly in the past two decades, allowing analysts easily to create plots that were formerly considered too complex or programming-intensive for regular use (e.g., Keele 2005).

One such class of plots is those for representing three-dimensional ("trivariate") data in two dimensions. While a number of approaches exist for displaying such data,[1] the two most commonly-used are what I term *surface plots* (which represent the three dimensions as a snapshot of a three-dimensional model) and *contour plots* (two-dimensional, "top-down" views of the data surface, with reference lines for varying levels of Y). Such plots are valuable for, among other things, presenting quantities of interest from regression-like models that include a multiplicative interaction of two continuous covariates.

Of these two types, a non-systematic review of published papers in political science suggests that surface plots tend to be more widely used than contour plots. Whether surface plots are the best available way of conveying the information contained in trivariate data is, however, open to question, and provides the motivation for this paper.

The experiment

In 2006, 28 students from the Interuniversity Consortium for Political and Social Research's summer program course on *Advanced Maximum Likelihood* were recruited for an experiment. The students question represented a range of institutions and substantive interests; most, however, were political scientists, and all were familiar with data analysis and statistical modeling up to the level of generalized linear models. In this respect, the students represented an ideal subject pool, in that all were—at a minimum—well-informed consumers of quantitative empirical social science, with solid training in statistical methods.

Each student in the class was randomly assigned into one of two groups. One group received a series of three surface plots, each corresponding to a particular function (what I term here *linear*, *interactive*, and *complex* functions), while the other received contour plots of the same three functions. The plots used are presented in Figure 1.[2] Students were not told the underlying function for each figure, only given either the surface plot or the contour plot for each. Each student was then asked to estimate the value of Y (the vertical axis in the plots) for four different pairs of (X, Z) coordinates for each function. In addition, each student was asked to rate, on a scale from zero (no accuracy) to ten (perfect accuracy) the accuracy of each of their estimates. The form

[1] It should be noted that a number of these—such as varying symbol sizes, shapes, and colors, as well as multiple, lattice-type plots, require the data be discrete or discretized. In contrast, I consider here the challenge of displaying data on three continuous variables simultaneously.

[2] Note that the plots presented to the subjects were full-sized (6" × 6") figures; they are reproduced smaller here. Careful readers will note that the contour and surface plots for the linear functions are different; the former shows the function $Y = 10 + 2X + 5Z$, while the latter plots $Y = 10 + 2X - 5Z$. This error (which the author attributes to the occurrence of a rare, backwards sunspot around the time of the experiment) was reflected in the plots received by the test subjects; as a result, the "actual" values for the linear plots differ for the two groups. Those differences were accounted for in what follows. My thanks to Gilles Spielvogel for noticing this error.

of the questionnaire is reproduced in the Appendix.

Results

Of the 28 students who completed the experiment, 12 were assigned contour plots, and 16 received surface plots. Table 1 presents the means and standard deviations of the subjects' estimates of Y for each of the three functions and four locations, by type of plot received, along with the actual (true) values of the functions at those points. For all three functions, subjects who received the contour plots provided estimates which were significantly more accurate than those who received the surface plots ($t = -2.71$, -5.70, and -6.85 for the linear, interactive, and complex functions, respectively, all $p < 0.01$). Across the three functions, the average estimate for subjects who received contour plots was closer to the true value than that for those receiving surface plots in ten of the twelve attempts. Additionally, the standard deviations for the estimates drawn from the surface plots were generally (in ten of the twelve cases) higher than those from the contour plots, indicating that subjects receiving the surface plots were less consistent in their responses than those receiving the contour plots.

Because the figures show widely varying ranges for Y, a more directly comparable measure of the respondents' accuracy is the *scaled deviation*, defined as

$$\text{Scaled Deviation} = \frac{|\text{Respondent's Guess} - \text{Actual Value}|}{\text{Empirical range of } Y \text{ for that function / plot}}$$

The scaled deviation variables for each of the three figures exhibit substantial skewness; both a ladder-of-powers plot and Shapiro-Francia tests confirm that a square root transformation brings the variable closest to normality. Boxplots of the square root of the scaled deviations for each plot and function type are presented in Figure 2. As Table 1 suggested, the normed deviations from the actual values were consistently highest for subjects receiving the surface plots; moreover, the extent to which this was the case is greater for the more complex plots than for the simple linear one. This general finding is also borne out by pooled regressions of $\sqrt{\text{Scaled Deviations}}$ on indicators for plot type and coordinate for each of the three functions (Table 2).

Finally, it is interesting to examine how subjects' actual accuracy relates to their perceptions of that accuracy. Figure 3 plots subjects' own assessments of the accuracy of their guesses against the square root of their scaled deviations (that is, estimated versus actual accuracy) for each plot and function type. In general, one would expect these plots to be downward-sloping: the less accurate their actual guesses, the less confident subjects should be that their estimate was accurate. While the relationships are not strong, the general pattern for subjects evaluating contour plots is as expected; in addition, for those subjects, the extent to which perceived and actual accuracy are correlated increases as the functions grow more complex. In contrast, the findings for subjects evaluating surface plots show is no clear relationship between subjects' self-assessed accuracy and their actual performance.[3] As against contour plots, then, surface plots appear to lead readers both to assess the information contained in the plot less correctly and to overestimate the accuracy of those assessments.

Discussion

While it would be hasty to infer too much from a single experiment, the results here suggest that, even for the simplest of functions, contour plots generally provide a better approach to displaying three-dimensional continuous data than surface plots. In this context, "better" refers strictly to the accuracy with which readers perceive and extract information from the plotted function. To the extent that many (if not most) figures in political science are concerned with accurately conveying specific values (for quantities such as data points, predicted values, etc.), contour plots appear to be a superior choice to surface plots.

At the same time, there are a wide range of other means for visualizing multivariate continuous data, including ternary plots, spider charts, and parallel coordinates plots, as well as brushing and other interactive methods. Such alternative methods—along with the surface and contour plots discussed here—can be especially useful for exploratory data analysis and descriptive data display.

References

Brambor, Thomas, William R. Clark, and Matt Golder. 2006. "Understanding Interaction Models: Improving Empirical Analyses." *Political Analysis* 14(1):63-82.

Gelman, Andrew, Cristian Pasarica, and Rahul Dodhia. 2002. "Let's Practice What We Preach: Turning Tables Into Graphs." *American Statistician* 56:121-130.

Jacoby, William. 2006. "The Dot Plot: A Graphical Display for Labeled Quantitative Values." *The Political Methodologist* 14(1):6-14.

Kastellec, Jonathan P., and Eduardo L. Leoni. 2007. "Using Graphs Instead of Tables in Political Science." *Perspectives on Politics* 5(4):755-71.

Keele, Luke. 2005. "3-D Graphics in R." *The Political Methodologist* 13(2):2-6.

[3] Regression results bear this out as well; estimated linear slopes for the relationship between self-assesed accuracy and (the square root of) scaled deviations were small and imprecisely estimated for both subjects receiving surface plots ($\hat{\beta} = -1.15$, robust s.e. $= 0.77$) and contour plots ($\hat{\beta} = -1.11$, robust s.e. $= 2.11$).

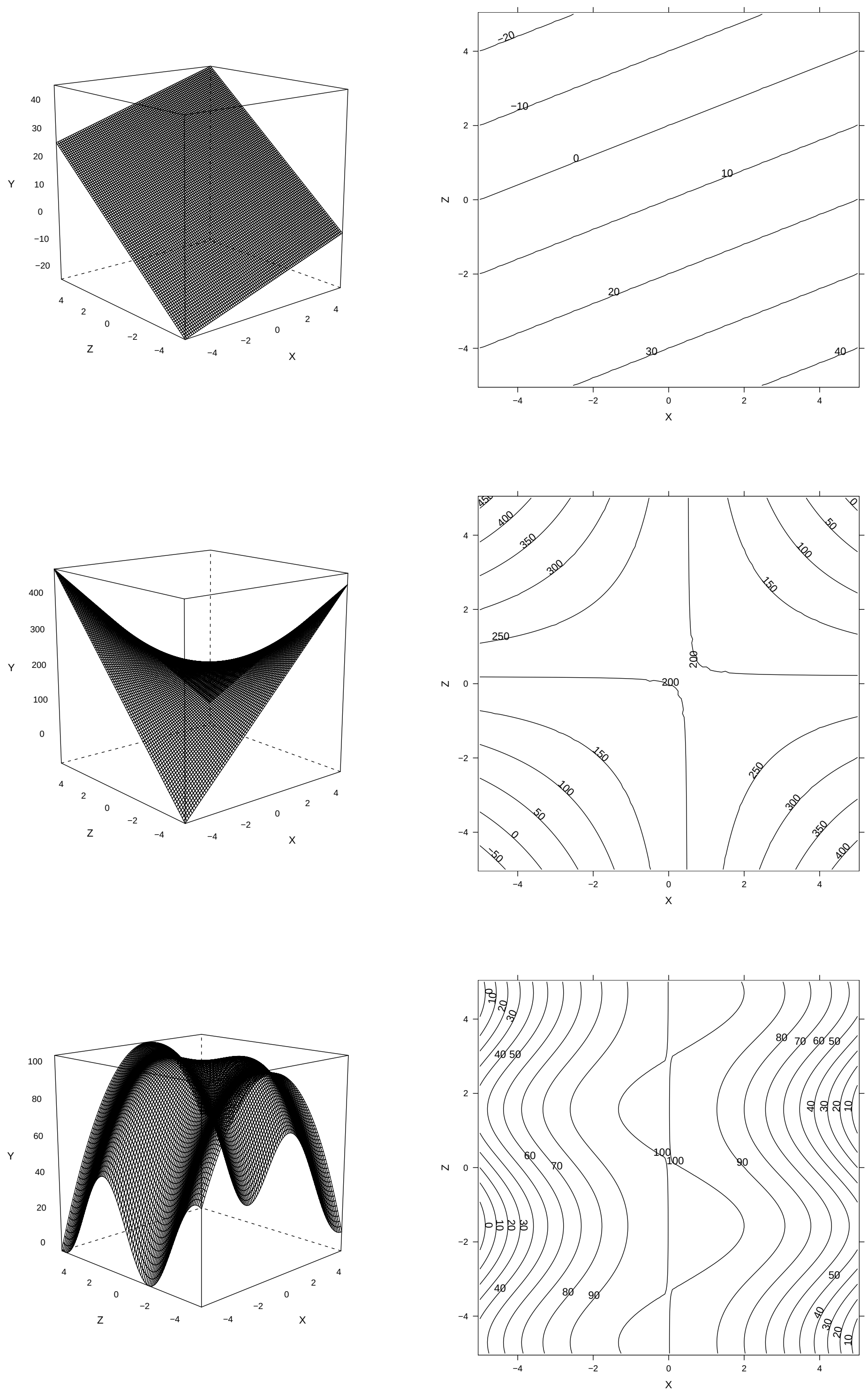


Figure 1: Surface and Contour Plots for Three Functions

Table 1: Means and Standard Deviations of Guesses, by Function and Plot Type

Function	Coordinates	Actual Value	μ_{Contour} (σ)	μ_{Surface} (σ)
$Y = 10 + 2X \pm 5Z$	(0,0)	10/10	9.08	2.50
			(2.87)	(7.30)
	(1,1)	7/17	6.17	9.39
			(6.09)	(10.56)
	(4,-4)	38/-2	32.92	-10.63
			(10.82)	(8.73)
	(-4,-3)	17/-13	20.42	-11.37
			(15.88)	(10.94)
$Y = 200 + 2X + 5Z - 10XZ$	(0,0)	200	196.25	133.75
			(6.08)	(78.90)
	(1,1)	197	193.92	129.06
			(24.96)	(55.32)
	(4,-4)	348	351.42	166.88
			(8.01)	(211.08)
	(-4,-3)	57	73.25	83.69
			(87.47)	(125.10)
$Y = 100 + X - 3X^2 - [5X\sin(Z)]$	(0,0)	100	100.5	82.81
			(1.45)	(14.83)
	(1,1)	93.79	92.17	81.06
			(4.39)	(13.19)
	(4,-4)	40.86	39.33	31.27
			(3.37)	(26.97)
	(-4,-3)	45.18	41.33	44.13
			(8.73)	(19.54)
N			12	16

Note: Rightmost cell entries are mean guesses; standard deviations are in parentheses.

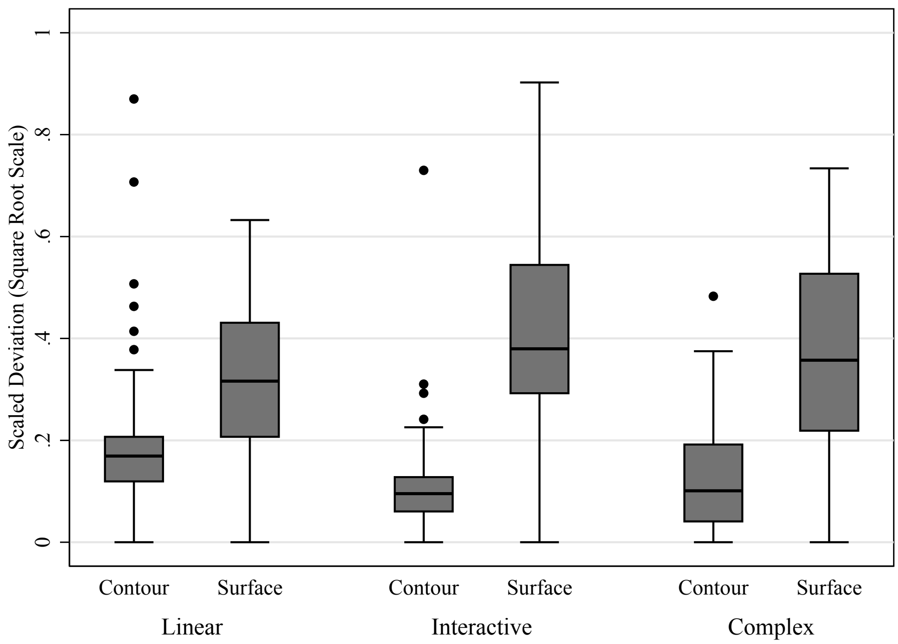
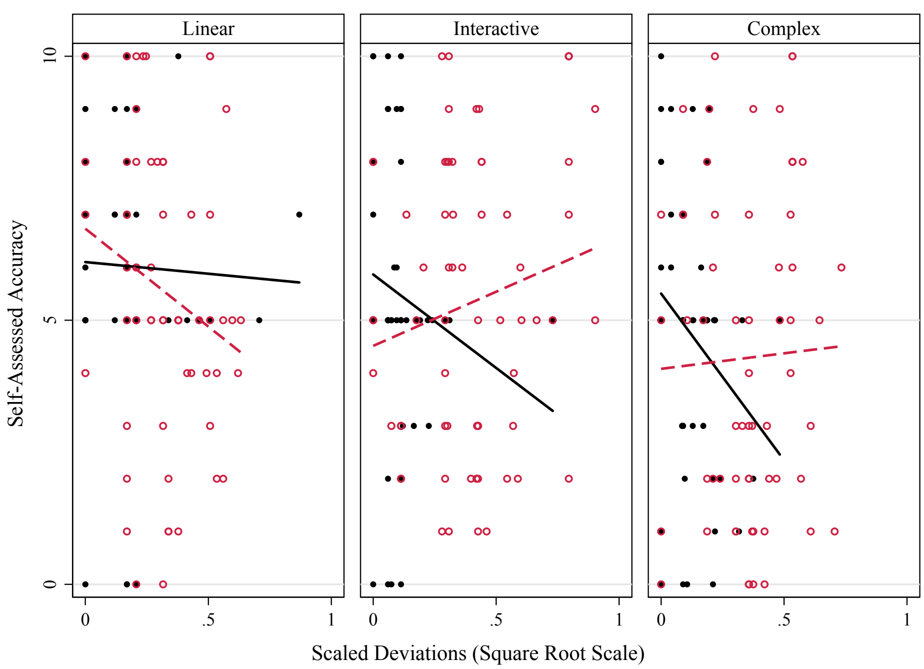


Figure 2: Boxplots of $\sqrt{\text{Scaled Deviations}}$, by Function and Plot Type



Note: Figure plots self-reported accuracy against $\sqrt{\text{Scaled Deviations}}$ for each of the three functions. Black closed circles are for respondents viewing contour plots, red hollow circles are for respondents viewing surface plots. Smooth black and dashed red lines are the respective linear fits. See text for details.

Figure 3: Scatterplots of Self-Assessed Accuracy Against $\sqrt{\text{Scaled Deviations}}$, by Function and Plot Type

Table 2: OLS Regressions of $\sqrt{\text{Scaled Deviation}}$ on Plot Type

Variable	Linear Function	Interactive Function	Complex Function
Constant	0.11	0.03	0.07
	(0.03)	(0.03)	(0.03)
Surface Plot	0.13	0.28	0.23
	(0.03)	(0.03)	(0.03)
Coordinate (1,1)	0.11	0.07	0.06
	(0.02)	(0.03)	(0.04)
Coordinate (4,-4)	0.22	0.19	0.09
	(0.04)	(0.05)	(0.05)
Coordinate (-4, -3)	0.09	0.09	0.09
	(0.05)	(0.05)	(0.05)
RMSE	0.16	0.18	0.15
N	112	112	110

Note: Cell entries are OLS estimates; numbers in parentheses are robust (Huber/White) standard errors, grouped by subject ID.

Appendix

This document contains {*surface*, *contour*} *plots* of three functions. The figures plot values of a third variable Y at various levels of X and Z. Based upon your examination of these plots, please indicate your best guess as to the value of Y at each of the following coordinates in (X, Z). In addition, rate—on a scale with **0** indicating **no accuracy** and **10** indicating **perfect accuracy**—how accurate you believe your guess of this value to be.

	Figure One		Figure Two		Figure Three	
Coordinates	Estimate of Y	Accuracy	Estimate of Y	Accuracy	Estimate of Y	Accuracy
$(0, 0)$						
$(1, 1)$						
$(4, -4)$						
$(-4, -3)$						

Computing and Software

Evaluating Scripting Languages: How Python Can Help Political Methodologists

Holger Döring
Universität Konstanz
Holger.Doering@uni-konstanz.de

Why Python?

Political methodologists tend to make passionate statements about their software tools. The PolMeth mailing list frequently gives strong advocacy for the use of Linux, LaTeX, Emacs and other specific programmes. For statistical analysis R has become the mainstream programming language. However, frequent encouragements to use PHP for web purposes or Perl for various scripting tasks highlight the need for a major scripting language beside R. Once political scientists need systematic parsing of markup languages or have to generate web presentations from their data, R quickly reaches its limits. For me, Python has become my favourite scripting language of choice. Having had some previous exposure to C, Java, PHP and Perl, Python turned out to meet all my needs for software development, that R can not fulfil. So let me introduce you to the beauty of Python.

Python helps with almost all of the data management tasks I need. Two applications of the language accompany my every day work: First, I use Python scripts to generate data sets from information provided at internet pages (web scraping). Second, I work with SQLite and Django to manage more complex data sets that require database operations, such as merging, virtual tables, and visualization in web pages. Both of these usages of a modern programming language have increased my productivity significantly and made data resources more easily available. In order to introduce you to Python, I first evaluate contemporary programming languages and their appropriateness for political methodology. Subsequently, I demonstrate how to use Python to generate a data set from an online source. In the last part, I discuss some more advanced issues of data analysis and evaluate how Python can help in a world of ever more easily available online data.

Evaluating modern programming languages

Python has existed for almost two decades and became popular among programmers in the late nineties[1]. The language is open source software and available at `www.python.org`. It is preinstalled with Mac OS X and most Linux distributions as well as easily installable on Windows systems. Today, Python is a highly developed, well documented language widely used. Much of google is driven by Python. Its development is constantly evolving with a major new version released in October 2008 (Python 3.0). Most important, the language is easy to learn and still satisfies your needs even if you have reached high levels of proficiency. Consequently, it is a good choice for a first programming language to learn.

As with every modern programming language, Python comes with a big standard library that makes many tasks easier, such as reading and parsing files or using regular expressions. In addition, many projects have evolved around Python providing extensions to the language. These packages allow for example rapid development of webpages, advanced numerical analysis or processing of various data types. Most of these additional modules can be easily installed via the Python Package Index (`pypi.python.org/pypi`) a repository of software that contains many useful program packages, similar to CRAN. Finally, the language comes with an interpreter, as in R, that allows shell-like exploration of language features.

How does Python perform compared to other software languages? Most of the more traditional programming languages are too complex for our every day tasks in political methodology. Memory management as required in C is something political scientist should not worry about for almost any of their projects. Java may also be too complex and its syntax too verbose for most of our tasks, however it is widely used in agent-based modelling.

PHP and Perl were the most popular scripting languages in the late nineties and the first part of this decade. The syntax of both of these languages can not stand up to the simple beauty of Python code. Perl even has become famous for its idiosyncratic syntax. Its motto of “There's more than one way to do it” has made it known to be the “The Swiss Army Chainsaw” allowing one to hack scripts very quickly. However, Perl scripts are often hard to understand by others or even by oneself after some time has passed. Beside Perl, PHP has been very popular for developing dynamic webpages. However, PHP requires a web server to run and is often less powerful than Perl and Python in not web related tasks. Finally, there are issues of language

[1] To compare the spread of different computer languages see for example the TIOBE Programming Community Index `www.tiobe.com/tpci.htm`

design that make Python a superior choice but I am not the one who can talk about design issues very competently.

Currently, only Ruby has developed to be a major contender of Python's popularity. Especially, its web framework "Ruby on Rails" has generated a significant group of advocates for the language. Ruby and Python are very similar and debates over general benefits of either of these languages quickly turn into very detailed aspects of agile programming languages. Python is in wider use among scientists whereas Ruby is driven signifcantly by web development projects. NumPy and Python's superb Unicode support, even better in the upcoming Python 3.0, would be my arguments in favour of Python over Ruby. Nevertheless, if you worked with Ruby and feel comfortable with the language, it may not be worthwhile to switch to Python. Either of the two languages fulfils the demands political methodologists have in data processing and web related tasks.

Data management with Python

Let us now turn to applications of modern scripting languages and demonstrate how they can support our every day computing tasks. More and more data sources for political scientists are available online. Mostly, the information does not come in a neat spreadsheet like format that we can easily import into our statistical packages. Data may be available in a highly structured format such as XML, but most of current online information is provided in HTML. It is up to the data analyst to turn these digital resources into data sets quickly and the task at hand is significantly easier to handle with a modern scripting language.

Recently, Jackman (2006) demonstrated how R can be used to read reasonable structured online sources. However, in my experience, R is quickly stretched to its limits once the web pages get somewhat more complicated. Solely relying on string processing tools such as regular expressions quickly creates complicated programmes and ignores the structure of the document provided by the markup language. Modern scripting languages come with parsers that make reading structured data easier. These parsers allow very fine grained access to elements of a data source. As reading and analysing structured documents is one of the major tasks for modern scripting languages, these parsers are well developed and different approaches to access the data are implemented.

The structure of most online sources we want to import is very simple. First, we have one or various content pages with links to each page that contains the information we are interested in. Second, data has to be extracted from each of these pages and sometimes the documents contain links to further information we may want to include. As an example, take a website with information on MPs or legislation. While browsing these webpages, we start from navigating index pages to the specific information we are interested in. Hence, our computer programs should do the same to extract the information we need.

Let us start with an example on how to fetch data from nested HTML pages. On the following pages, I provide a Python script, that reads information about all postings on the PolMeth mailing list as provided at `http://polmeth.wustl.edu/polmeth.php` and its subpages (see Figure 1). From this information, we want to generate a data set with information on all postings (author, date, title, URL) at the PolMeth list. In addition, we want to calculate a top ten list of authors posting at PolMeth. I now demonstrate how these nested online data sources can be turned into a data set with a modern scripting language.

Figure 1: PolMeth mailing list archive

To fetch all postings from the PolMeth mailing list, I wrote a Python program that parses data from the online source. For this task, we rely on functions provided in the Python standard library. The library comes with an HTML parser of its own that is a little clumsy. An additional Python package that parses HTML documents, BeautifulSoup (`www.crummy.com/software/BeautifulSoup`), is easier to use and shows better performance with invalid HTML pages. In my work, I have had very good experiences using BeautifulSoup to parse very different online sources.

The following script fetches the data from the PolMeth list web page we are interested in. It proceeds by first reading the index page `http://polmeth.wustl.edu/polmeth.php` and extracts all links to the monthly summaries—links starting with /mailinglist/search.php. Subsequently, the program loops over all links, reads the related pages and extracts the information we are interested in. To read the information about a posting, the program processes every table row and extracts information from column entries and hyperlinks provided for posting. Finally, results are stored in a list structure.

Web scraping: An example

```
#/usr/bin/env python
import codecs, csv, re, urllib
from BeautifulSoup import BeautifulSoup

polmethurl = 'http://polmeth.wustl.edu'

## read online file and parse html
s = urllib.urlopen(polmethurl + '/polmeth.php')
index = BeautifulSoup(s.read())

regex = re.compile('^/mailinglist/search.php')
urlmonths = index.findAll(attrs={'href': regex})

## process monthly section of postings
data = []; m = {}   # initialize elements
for url in urlmonths:

## read page for current month
    url = polmethurl + url['href']
    print 'fetching ' + url
    s = urllib.urlopen(url)
    mails = BeautifulSoup(s.read())

## process table rows (skip headline)
    for mail in mails.table('tr')[1:]:
    entries = mail('td')   # row elements into list

## extract information from row elements
    m['date'], m['time'] = entries[0].string.split()
    m['title'] = entries[2].a.string.strip()

    m['url'] = entries[2].a['href']
    m['url'] = polmethurl + '/mailinglist/' + m['url']

## extract author name from first field (if existing)
    m['authorfull'] = entries[1].string.strip()
    if m['authorfull'].find('&lt;') != -1:
    m['author'] = m['authorfull'].split('&lt;')[0].strip(' "')
    else:
    m['author'] = m['authorfull']

## add elements to list of data
    data.append(m.copy())
```

Looking at the program listings you may have quickly realised that Python uses indentions to separate program blocks. This helps to write readable code. In the program, we use lists and dictionaries to assign our data. At the end of the program, information we are interested in is saved in a list where each entry contains information about one posting at the PolMeth list. Every entry has an element for the following information: author, date, time, authorfull, title, url.

After we have processed the data from our online sources, we have to decide if we want to perform our data analysis in Python as well. In the Python community, there is the NumPy package, that allows mathematical array processing and is widely used by scientists. For our current task, determining the top ten posting authors, python standard tools are sufficient.

```
## determine the number of postings per author and sort results
authorlist = [x['author'] for x in data]
authors = [(authorlist.count(x), x) for x in set(authorlist)]
authors.sort()
authors.reverse()

print '\nTop ten postings on PolMeth list'
print '--------------------------------'
for i, j in authors[0:9]:
    print str(i) + '\t' + j
```

The listings show that processing spreadsheet like data in python is rather awkward. Here, we rely on list comprehension to generate a variable with the information we are interested in, the number of postings per author.

For me, once I have downloaded and parsed my data from online sources I quickly change horses and turn to R for data analysis. Python is very powerful to process online sources and text files. However, once I have converted these information into something that is more spreadsheet like, R feels more natural.

In order to export our data from the Python program we generate a csv file. Python's syntax for exporting data into a csv file is somewhat verbose. For completeness, I give the listing here:

```
## write all data into csv file
outfile = codecs.open("polmeth.csv", "wb", "utf-8")

cols = ['author', 'date', 'time', 'authorfull', 'title', 'url']
writer = csv.DictWriter(outfile, cols, quoting=csv.QUOTE_NONNUMERIC)
writer.writerow(dict(zip(cols, cols)))
for i in data:
    try:
        writer.writerow(i)
```

```
    except:
        "Print can't write row" + i['date'] + i['time']

outfile.close()
```

You can also write the information directly into a database, such as MySQL or SQLite. Once you have exported the data from Python you can proceed by analysing the information in your favourite statistical package. For our example, generating the top ten list of authors posting at PolMeth is straight forward in R. It just requires three lines of code:

```
pm <- read.csv("polmeth.csv", as.is=TRUE)
authors <- sort(tapply(pm$author, pm$author, length),
                decreasing = TRUE)
authors[1:10]
```

Maybe, you have gained some ideas how Python, or any other modern scripting language, may help to derive data from structured information sources. In the previous examples, we used Python to extract data from the PolMeth list archive. Relying on a HTML parser that comes with any high level language allows fine grained access to page elements. Running the scripts, you may have figured out what the top ten list looks like.

Future online presentations will provide information that is even better structured (eg. XML, JSON, etc.), hence easier to access. Hopefully, I have convinced you that modern scripting languages are extremely helpful to turn various data sources into data sets. My discussion of recent trends in data provision at the last section will explore more applications for modern scripting languages. Before, let me shortly present some more tasks I manage with Python.

Organising and presenting data

I regularly use Python to convert online data into data sets. In addition, I apply the language to process textual data and for record linkage of different data sets. The latter requires use of a simple fuzzy string matching algorithms. The language also helps with a couple of minor tasks that show up every now and then, such as renaming multiple files, data conversion etc. For most of these requirements I can rely solely on Pyhton's standard library and sometimes I add external open source packages. I rarely have to code extensivly to fulfil my data management tasks.

Some of my data sets have become rather complex and require a database design to be managed coherently. I have started to use the Python web framework Django (`www.djangoproject.com`) in order to develop a web interface for this data. Web application frameworks allow one to create dynamic websites easily. Once you want to provide an interface to more complex data structures, frameworks allow you to develop a web interface quickly. Web frameworks help with accessing your data from a database with user management and templating of HTML pages. Django, a popular Python framework, has allowed me to manage my data very easily. In my view, it is easier and faster to develop a web presentation in Django than in PHP, which is often used for these tasks. You will get familiar with Django quickly if you have some knowledge of Python.

Building web applications for data sets is not a major task for political methodologists. However, once you work with different people on the creation of a data set, web based data coding makes the data generation more reliable. Web frameworks also allow you to include modern features of webpages, such as wikis and comment sections. In addition, see the shift from paper based to Internet based uses of surveys. Nowadays, everyone can setup a small online survey with very limited resources. As this article promotes the usage of Python for data related tasks, web frameworks that are provided for modern scripting languages make the creation of data presentations online significantly easier.

Trends in data provision and analysis

There are current trends in the online world that will make knowledge of a modern scripting language even more important in the coming years. There are web pages that provide systematic data on questions of political science and these data sources are regularly updated. Combining these data sources allows one to generate data sets for studies of political science automatically.

Take for example the web page at `projects.washingtonpost.com/congress`, a so called mashup. The page collects and presents information on voting in the Congress from various online sources. It is important to note that the providers do not code any of the data themselves. They just combine existing online sources. The page has been created with Python and Django, tools I have previously presented. Political scientists that want to make systematic use of data in a similiar way need programmed scripts that include these data sources automatically or download them at regular intervals. Modern data generation does not require one to code extensive amounts of data manually but to code a computer script that extracts information. Hence, political methodologists have to gain knowledge of a scripting language in order to write these scripts.

There is a strong trend in the online community towards a more systematic distinction between data and layout. This trend is referred to as the semantic web or web 3.0. Out of this work data will be presented in a way that allows a combination of different data sources. One interesting project in this respect is `www.freebase.com`. Whereas Wikipedia provides a huge data source of information online, its different information is difficult to extract automatically. Freebase provides an open database that can be extended by its users and the project provides an open interface to systematically extract relevant information. In the same line, Wikipedia has also started to include some semantic information in its articles.

Other online sources are also provided in a more systematic way. I do not want to go into too much detail. Creating data sets by drawing on systematically organised online sources will be a substantial part of our future work on data generation. Contrary, to the approach that I provided in my example, these data sources can be read by systematically specifying the content of the element to be extracted instead of relying on format parameters such as the table entry as used in my example.

Combining and analysing the ever growing amount of information available has led to new methods of data mining. Segaran (2007) gives a nice and accessible demonstration of how Python can be used to analyse different online information. He provides examples of modern data mining techniques applied to various online information that provide systematic interfaces to their data (APIs). Segran's book shows how you access the online resources via Python and discusses different data mining algorithms to analyse these data. The scripts are short and easy to read, most of the statistical techniques are similar to the ones we apply in political methodology. The book gives many inspirations on how to make use of new opportunities provided through structured online data.

To make systematic use of modern methods of data provision and analysis you need to have some knowledge of a powerful scripting language. All these languages come with package repositories that provide many scripts to work with online data and to access various web resources. To turn the information you are interested into a data set requires you to include these packages in your own script and to modify them for your needs.

Conclusion

Hopefully, this note was sufficient to convince you of the benefits that modern scripting languages provide for political methodologists. In my opinion, Python and Ruby have the right balance between power and complexity for all programming tasks that statistical programming languages can not fulfil. Both languages are easy to learn and are still powerful programming languages once you have gained more proficiency. It may well be that every decade has its programming language and I believe that Python and Ruby are today's languages for computing tasks in political methodology. Knowing one of these modern scripting languages becomes even more important as online data sources become more numerous and better structured. A powerful scripting language at hand will help you to draw on this information quickly and adds to many more of your scripting needs. If I have convinced you on the power of Python, pick up Chun (2007) to get a thorough introduction into the language and its applications.

References

Chun, Wesley J. 2007. *Core python programming.* Upper Saddle River, NJ: Prentice Hall.

Jackman, Simon. 2006. "Data from the web into R." *The Political Methodologist* 14(2): 11-15.

Segaran, Toby. 2007. *Programming collective intelligence: Building smart Web 2.0 applications.* Sebastopol, CA: O'Reilly Media.

Professional Development

Introducing the Advice to Junior Faculty Column: Advice from Gary King

Corrine McConnaughy
The Ohio State University
mcconnaughy.3@polisci.osu.edu

Junior faculty members of the Society for Political Methodology can find themselves in real need of mentorship and advice from senior colleagues perhaps even more so than graduate students. Some junior faculty members are the only political methodologists in their home departments. Others are not so alone, but do not always feel safe to ask questions about their career considerations of their own senior colleagues; doing so in their first year or two on the job, when they have yet to forge relationships with those colleagues, may be especially daunting. In response to these concerns, SPM Long Range Planning Committee proposed an advice column in *The Political Methodologist* as one venue for increased mentorship of junior faculty by senior SPM members.

To ensure the relevance of the column to our junior faculty members, I contacted a diverse set of junior faculty SPM members to solicit suggested questions or topics. I tried to get feedback from members at different points on the tenure clock, at different types of institutions, and of differing methodological interests. The responses indicated that our junior members have a variety of concerns they would like to hear the more senior members discuss from publishing concerns such as what makes one's work a real contribution to the field and where to publish it, to questions about how to justify what we do to our colleagues and students, to questions of navigating our workloads. Using this feedback, I developed a list of questions to be addressed by senior faculty members over the next several issues of *TPM*. We hope you find these mentoring conversations in print informative and useful. Suggestions for future topics and reactions are always welcome.

Publishing considerations

> I have both substantive and methodological research interests. For tenure review purposes, however, I wonder whether I should produce more research on substantive topics or spend more time on purely methodological topics? How are junior scholars who do both substantive and methodological work evaluated in tenure reviews, including in external tenure letters from methodologists?

Optimally, you should choose based on where you are likely to have the biggest impact. In most cases, this is the area you like the most, or are most interested in, since making a big impact will normally take a great deal of devotion in terms of time and effort. It matters much less exactly which combination of interests you choose than doing what you choose to do well. Of course, very few people know what their big research hits will be before working on them, and so its best to push forward wherever you think you may make a difference, including on several projects at the same time. You are contributing to a collective enterprise, rather than working entirely on your own, which means you can contribute via methods, substance, or some of both. In the end, political science is a substantive discipline, and so there tends to be great suspicion of anyone (theoretical or empirical) who does not take the substance very seriously. So if you contribute primarily to methods, then be sure your methods are tuned to the specifics of the substantive political problem at hand. But similarly, if you only pursue substance, it is in your interest—both in terms of the likelihood of learning the most and, for that reason, in being attractive to prospective employers—to have the most cutting edge tools available.

> I have a paper that was just rejected from *Political Analysis*, but the decision seemed like a close call, and the reviews suggest the paper could be revised for another outlet. What journals, perhaps in other disciplines, other than *Political Analysis* would be a good fit for work that tackles econometric issues? How receptive are these journals to our work? And do we need to frame or otherwise write the paper any differently than we would for *Political Analysis*?

Some other journals you might consider include the Workshop in the *American Journal of Political Science*, *Sociological Methods and Research*, *Public Opinion Quarterly*, *Historical Methods*, *Psychometrika*, *Journal of Statistical Software*, and many others. I would also look to the substantive journals; these can help you reach potential users of the methods you describe or produce.

But FYI, one of the great things about *Political Analysis* is that the editors tend to work closely *with* authors—rather than only judging their work from a distance. This means that if it really is a close call, the editor of *PA* will often help you get your paper in shape for publication. But even if every paper you write could get accepted at *PA* on the first round, you shouldn't publish all your work there. Publishing in different venues shows tenure review committees that your work can pass muster with editorial boards with different standards, and it helps you reach new audiences. And if you're having a hard time with reviewers, remember that there are numerous scholarly journals. If you studiously try to improve your paper after a negative review, unless something in your paper is wrong, you are likely to get it published at some point. Sometimes it just takes perseverance. Don't let a paper sit in your drawer; send it out.

> I'm not sure I know what it takes to make a contribution to the field of political methodology. I've noticed that there seem to be a lot of conversations among political methodologists these days about creating new methodological techniques, rather than simply borrowing and adapting techniques from other disciplines. Do these conversations imply that I will get little or no credit for smartly importing techniques? If I will get credit for importing, what are the criteria for making that importation a real contribution to our field?

You'll get credit if your work makes a difference for applied political scientists, no matter what it was you did. Your question, of course, implies a large dose of math envy. If you import methods, you'll be outclassed, it seems, by those inventing new methods, but those people feel outclassed by those who come up with new classes of methods, and they, in turn, worry about being outclassed by "real" statisticians who may develop new ways of deriving new classes of methods. If you pay close enough attention to professors in statistics departments, you will learn that they also have inferiority complexes because their math skills are dwarfed by mathematical statisticians, and the mathematical statisticians worry about the "true" mathematicians. And the mathematicians have their own hierarchy.

No matter what you do, you will run into impressions like these. My view, and I think that of most political methodologists, is that this hierarchy is not appropriate for us (even though some political scientists use it too). Technical skills are very important in our technical subfield, and so go out and scoop up all the skills you can. But in the end, what matters in our substantive discipline is making a difference to the practice of political science. So I like technical work, but I care the most about work that improves how much real political scientists learn and can learn from political data. Sometimes, you can make a huge difference by importing a method from another field, and explaining it clearly in our language so political scientists can benefit. Sometimes, you may need to explain the method and write some software so that its easy for researchers in our field to use. Other times, you may need to adapt that method in some ways to our problems. And still other times, you may need to develop a new method from scratch. You can also make a huge difference by collecting a new set of data and making it available, promulgating informatics techniques that preserve and distribute data, and being among the first to apply a new method. It's great when statisticians and methodologists cite the work of political methodologists, build off of it, and even contribute methods that help us solve problems for political scientists. But I would do whatever you are capable of that makes the biggest difference for the field. We all benefit by having a subfield that includes all of these types of contributions: you need not do everything yourself.

Teaching issues/workload considerations

> What is your advice on adding new techniques that you have not yet worked with, but would be willing to learn along with the students, to an advanced political analysis class? Are there benefits? Pitfalls? Could it undermine my authority in the classroom? Can I really learn along with the students, or do I need to learn first and teach later?

I wouldn't agree to teach something I didn't know well ex ante. Methods, and math in general, is just plain harder than learning about some area of government and politics. Think about it this way. Suppose you had to give a lecture on the American presidency in 10 days, and you were to learn the exact topic only 5 days from now. How worried would you be? I doubt any political scientist would be terribly worried. Even if you don't know much about whatever the topic turns out to be, I'm sure you're confident that a good evening of reading will be enough to figure it out. But suppose we did the same for a statistical or mathematical topic; its just not the same. Another way to think about this is that, as a student, statistics is the only area within political science where you can't really understand what the class is about until after you've completed it. If you want to learn something new and teach it to your students, either do it before you write the syllabus or change the syllabus during the year after you've figured it out.

> Although my department seems generally to allow junior faculty to teach the same three or four classes every year, I have been asked to prepare more new classes because of our department's methodological training needs. While I am sensitive to the fact that I was hired because of these needs, I also feel the requests are unfair, as they make my teaching workload unequal to my junior colleagues and, frankly, I worry about being able to meet the research expectations for tenure with the additional effort I need to put into teaching. How would you suggest I deal with this situation?

Teaching a methods course requires considerably more preparation time than most substantive courses, even though one is not more important than the other of course. A methods course requires hundreds of hours of work carefully choreographing lectures, preparing slides, writing handouts, and giving assignments. In contrast, a discussion course on your substantive area of interest probably may require very little preparation. This difference in workload could hardly be more stark, and it's important that your chair understand this. So don't be a pest, but try to educate your chair.

But that said, I would (and I do!) try to teach methods courses. There's a lot more startup costs, but if you prepare carefully the first year, preparation in subsequent years can be somewhat less onerous. (And a note to your chair: this is not like teaching introductory calculus, since our subfield is among the most dynamic in the discipline; those lecture notes and slides will need to be updated every year!) However, teaching students how to do research and how to think and learn about the world can make an enormous difference for your students, and so tends to be more gratifying for you too. And as important, what you will learn from the experience of learning to explain difficult statistical concepts to novices will likely be very important to your work and your career.

Justifying our work to others

> I keep getting asked by intelligent and well-meaning colleagues whether someone could

> teach a one semester statistics class that would teach students everything they really need to know about statistics. You must get this question all the time. How do you respond? How do you think I should respond?

If there is such a course, please do sign me up! Its unlikely because in our subfield, knowledge is cumulative. As a consequence, scientific progress has been spectacular in our field and those that depend on us for methodological advice and innovations. But also as a consequence, what we teach requires more and more sophistication, and our courses are therefore taught in a sequence, where it really is true that you can't understand the second course without the first. Although we do have prerequisites in the rest of political science, its not difficult to skip the introductory American government course and jump to the course on Congress for example; that's much more rare in methods and less likely to work.

> In introductory courses, how would you suggest dealing with the following sort of student protest: I just want to run regressions. And I can just say 'regress' in Stata. Why should I care about matrix algebra or calculus? How does learning about Var(. . .) help me type 'regress' more effectively or better? I mean, I don't know matrix algebra and I'd need a whole course just to learn it, and I'm not going to be a methodologist, and my advisers all got jobs and they don't know any matrix algebra and they just type 'regress'? (Note that I'm concerned both with providing an effective and sensible response, and with navigating around saying something impolitic about my senior colleagues.)

Many academics use the same methods their whole career that they learned in graduate school; if you want to do that, fine, you already know what you need to know. But the really successful empirical social scientists tend to update much more frequently. When we teach *our* students, we try to teach not only the latest and greatest methods. We also try to give them the tools to learn a new method when it becomes available. Given the fast-paced and accelerating progress in political methodology, we know that a great new method will be invented right after our students graduate this year (and next, and next...). We don't want their knowledge to be obsolete immediately upon graduation. And so we do teach them the latest and greatest, but we also try to teach them the fundamentals of how methods are created. When that new method is created and is relevant to our students' work, we want them to have the tools in hand to be able to read, understand, evaluate, implement, and use the new method in their work. To do that, they may even need to know some matrix algebra!

The advantage of this kind of response is that it is not only completely accurate, but it also helps explain to your colleague that political methodology is not like learning French or fulfilling some other support role. It is an important, dynamic field making great progress, and making things possible that were never before contemplated. You ought to be able to convey this so that they will value you, your contribution to your students, and the importance of your subfield.

Book Review

Review of Kunio Takezawa Introduction to Nonparametric Regression

Robert W. Walker
Washington University in St. Louis
rww@wustl.edu

Introduction to Nonparametric Regression. Kunio Takezawa. John Wiley & Sons, 2005, 568 pages. $130.50, ISBN 978-0-471-74583-9 (hardcover).

Overview

> Note: The book is an English edition of the first volume of Kunio Takezawa's (2003) second-edition Japanese language text on nonparametric regression in two volumes with omitted chapters on wavelets, neural networks, and tree-based models. (Preface)

Kunio Takezawa takes the reader on an extensive and well organized tour of the broad field of nonparametric regression that fulfills the need for an instructional text on nonparametric regression. Weighing in at over 500 pages, inclusive, the book is hefty and achieves much. The meat of the book—the study of single outcomes—is a comprehensive textbook for the study of nonparametric regression; in the remainder, the "Introduction to ..." is much more fitting.

As a textbook on general nonparametric regression, the book is generally thorough and clear. At the same time, where Takezawa goes beyond nonparametric regression for single outcomes, the book leaves room for improvement.[1] Functionally, the book is organized around an introductory chapter, bivariate techniques (Chapters 2 and 3), multivariate techniques (Chapter 4), nonparametric regression with explanatory densities (Chapter 5), density estimation (Chapter 6), and pattern recognition and classification (Chapter 7) with inline code from the $S+$ language and an appendix covering conversion to R objects for the code in all chapters. Transparency of this sort in the presentation of difficult or unfamiliar topics is a considerable virtue and the author's frequent reliance on illustrations and graphics provides both intuition for the mathematics and practical ideas on the presentation of applications of nonparametric regression. Moreover, the book is unique in that each chapter concludes with problems for reviewing and cementing the concepts and this feature makes the book ideal for self-study or classroom use. In short, there is much to like about Takezawa's text. With this in mind, I will first describe the layout of the book with a few general comments on the individual chapters and themes, before turning to a few broader points about the book.

Chapter 1 introduces the text and provides the argument for nonparametric approaches. Unfortunately, Takezawa's view of nonparametric statistics may be somewhat off-putting to the social scientist who willingly wields Ockham's razor, e.g. "... what distinguishes nonparametric regression from parametric regression is the nonexistence of an orientation toward the reduction of the number of parameters (regression coefficients) ... Parametric regression favors expressions with as small a number of parameters as possible and selects a regression equation with a larger number of parameters only when the use of such a regression equation is indispensable for representing the data well. On the other hand, nonparametric regression is focused chiefly on the goal of deriving beneficial trends from the data and does not give special consideration to the reduction of parameters." (p. 19). In the presence of numerous other justifications for nonparametric regression, this choice may severely limit the degree to which Takezawa's text becomes the choice of social scientists studying nonparametric regression.

The real strength of the book is regression analysis of single outcomes. Chapter 2 continues the assault on parametric methods with an examination of smoothing on equispaced predictors involving the binomial filter, moving averages, splines, local linear regression. Where possible, there is a welcome emphasis on detailed relationships among many of the estimators and, in grouping estimators by basic properties, a broader understanding of the range of estimation strategies and their relative strengths and weaknesses emerges that will be of considerable use as a practitioners guide for a given application. Chapter 3 is more constructive in a comprehensive examination of nonparametric regression for the general one-dimensional predictor (Nadaraya-Watson estimator, local polynomials, natural and smoothing splines, LOESS, LOWESS, Supersmoother). Though relatively new developments like bent-cable regression do not receive coverage, the sweep is as extensive as any I know. To round out the analysis of single outcomes, Chapter 4 dedicates 100 pages to multidimensional smoothing including additive models, LOESS, LOWESS, local polynomial, projection pursuit, and ACE (Alternative Conditional Expectations) regression along with kriging and thin plate smoothing splines. These three chapters together form a nice and thorough introduction to nonparametric regression.

The final three chapters represent somewhat more advanced topics. For example, Chapter 5 (30 pages) on nonparametric regression with explanatory densities and distributions was unfamiliar to me before Takezawa's presentation and it is always nice when introductory texts in crowded fields can provide insights into a novel and relevant topic. Chapter 6 (50 pages) is on histogram smoothing and nonparametric densities. The hardest of these to evaluate is perhaps Chapter 7. Chapter 7 (60 pages) focuses primarily on classification methods and pattern recognition and this chapter is hard to evaluate. The chapter covers a host of interesting topics—decision, discrimination, and classification rules, logistic regression and neural networks, tree-based models, nearest-neighbor classification, and others—but all are discussed in a rather terse presentation that is quite unlike the rest of the book. Though the author recognizes this and cites additional work on pattern recognition to bolster the brief treatment, I would prefer that the discussion either be eliminated or strengthened to the level of the preceding chapters. Of course, it should be said that there are a host of books specifically dedicated to this subject, e.g. (Ripley 1996).[2]

Four general shortcomings merit consideration. First, it does not really live up to an expressed desire to be the "most beginner friendly" text on nonparametric regression. Moreover, the publisher's characterization of the text as requiring "Only a basic knowledge of linear algebra and statistics," is unlikely to be true. For example, how such knowledge renders the multiple integrals in thin plate

[1] These deficiencies are apparently remedied in the Japanese language version, though I am regrettably unable to verify this.

[2] As mentioned before, it seems that this shortcoming results from editorial decisions about what to purge in translation.

smoothing or even the discussion of selection criteria for regression equations which revolves around a *Mean Integrated Squared Error* criterion clear is open to question.[3] I also somehow doubt that such preparation would make passing references to the delta function and Fourier analysis clearly understood. Though the book is a comprehensive introduction, the level of preparation required to digest the full text is considerable.

Second, the general presentation strategy may not appeal to all audiences. One of the great virtues of Simonoff's (1996) text is a discussion, intuition, and illustration with appendices that fill in the fine detail for more advanced readers. Takezawa has opted out of this approach in favor of presenting all the material in the chapters with appendices reserved for computer code. For more advanced readers, this is quite nice, perhaps spectacular, but those seeking an introductory text on nonparametric regression may find the inline rigor distracts from the broader point in many places.

Third, though there is a frequent reliance on graphics and illustrations, the graphics themselves can be dramatically improved. The graphics are typeset sparsely without legends. Moreover, the graphics generally lack informative labels in multipanel graphics or informative titles to make it easy to determine the exact point of a given graphic. Though the graphics are helpful, once understood, these shortcomings make additional (and unnecessary) work for the reader.

Fourth, the book is woefully short on examples with real data. Frequently, points are illustrated with the help of (targeted) simulated data and real applications are quite few and far between. Thus, those who wish Takezawa's text to provide a guide for application will find the task made harder by a lack of easy to follow illustrations of the techniques applied to real world problems. At the same time, the inline code does make application easier, even if there are no direct examples to follow in most cases.

In summary, though the English edition of Takezawa's *Introduction to Nonparametric Regression* is not without shortcomings, the text fills an important niche with an extensive introduction to nonparametric regression with a single outcome and further advanced topics. For those looking for the theory behind a given nonparametric estimator and some R or $S+$ code to implement said estimator, Takezawa's text is ideal even if it often lacks worked real world examples. Though there are numerous other books on smoothing and nonparametric regression in general, Takezawa's is among the most comprehensive journeys through nonparametric statistics. If researchers are willing to tackle Takezawa's *Introduction to Nonparametric Regression* in its entirety, the journey is not without considerable reward.

References

Ripley, Brian D. 1996. *Pattern Recognition and Neural Networks.* Cambridge, UK: Cambridge University Press.

Simonoff, Jeffrey S. 1996. *Smoothing Methods in Statistics.* New York, NY: Springer-Verlag.

Announcements

John Jackson winner of the Section's Career Achievement Award

Janet Box-Steffensmeier
The Ohio State University
jboxstef+@osu.edu

The citation for John's award that went up to APSA reads as follows:

> The Society for Political Methodology will award its 2nd Annual Political Methodology Career Achievement Award at the 2008 American Political Science Association's Annual Meeting. This year we recognize JOHN JACKSON's outstanding career of intellectual accomplishment and service to the profession.
>
> John was at the forefront in the establishment of the field. Long before most were aware of what political methodology was about, John was extremely active, bringing scholars together and laying the foundation for work to come. John was publishing high quality statistical analyses in the APSR in the early 1970s, and his influential text "Statistical Methods for Social Scientists," coauthored with Eric Hanushek in 1977, is still considered one of the best. Likewise, John's pioneering empirical work showed that party identification need not be seen as an essentially permanent identity learned in childhood. Instead, John demonstrated that partisanship also reflects an accumulation of citizens'

[3] This is a standard mean squared error criterion that is integrated over the density of the predictor.

adult experiences with the parties—a perspective that has been built on by many empirical and theoretical scholars, and that has become the most widely accepted view of how partisan identity is formed.

John's record of service to the subfield is equally impressive. He served as the 2nd President of the Society for Political Methodology from 1985-1987, and was instrumental in securing funding for the early meetings from the National Election Studies and later the National Science Foundation. Moreover, John has always been (and continues to be) known for reaching out to graduate students, spending time with them at them at the Political Methodology Meetings and assisting with their integration into the discipline. And, John has been instrumental in the maturation of the subfield in another way, as he has led the charge when it comes to the forging of ties between political methodologists and methodologists in other fields (both in his own collaborations and in institutions)—this has been of fundamental importance to the bettering of the subfield.

John's work has always brought methodological insight to important substantive questions, and he continues to publish state of the art work, having recently co-authored a book on Polish elections ("The Political Economy of Poland's Transition, with Jacek Klich and Krystyna Poznanska). In addition, John is still extremely active in the Society for Political Methodology, being both a regular at the summer meetings and a mentor to many. This year, John's career achievements were recognized in his election to the American Academy of Arts and Sciences, confirming what one eloquent nominator noted: "John is an icon for political methodology."

Award committee: Janet M. Box-Steffensmeier (The Ohio State University), Christopher Achen (Princeton), William Berry (Florida State University), and Simon Jackman (Stanford University).

Best Graduate Student Poster in Political Methodology

Suzanna Linn
Pennsylvania State University
suzanna.linn@GMAIL.COM

After viewing a large number of outstanding posters and much deliberation, the committee to select the best graduate student poster in political methodology chose Xun Pang's (Washington University in St Louis) poster, "Binary and Ordinal Time Series with AR(p) Errors: Bayesian Model Determination for Latent High-Order Markovian Processes." In her paper, Pang provides a solution to a problem that has vexed time series specialists in all fields for some time—how to estimate serial correlation in models with limited dependent variables and, if discovered, how to model it in a way that does not create problems for the structural portion of the empirical model. This is an excellent paper that makes an original contribution to the statistics and empirical methods literatures that will be important for scholars across a range of substantive fields.

The committe was composed of Tom Carsey, Curtis Signorino, Jana von Stein, Bill Clark, Dean Lacy, Drew Linzer, Robert Erikson, Karen Jusko, and myself (chair).

Department of Political Science
2010 Allen Building
Texas A&M University
Mail Stop 4348
College Station, TX 77843-4348

The Political Methodologist is the newsletter of the Political Methodology Section of the American Political Science Association. The support of the Department of Political Science at Texas A&M in helping to defray the editorial and production costs of the newsletter is gratefully acknowledged.

Subscriptions to *TPM* are free to members of the APSA's Methodology Section. Please contact APSA `http://www.apsanet.org/section_71.cfm`) to join the section. Dues are $25.00 per year and include a free subscription to *Political Analysis*, the quarterly journal of the section.

Submissions to *TPM* are always welcome. Articles should be sent to the editors by e-mail (`tpm@polisci.tamu.edu`) if possible. Alternatively, submissions can be made on diskette as plain ascii files sent to Paul Kellstedt, Department of Political Science, 4348 TAMU, College Station, TX 77843-4348. See the *TPM* web-site, `http://polmeth.wustl.edu/tpm.html`, for the latest information and for downloadable versions of previous issues of *The Political Methodologist*.

TPM was produced using LaTeX on a Mac OS X running iTeXMac.

The Society for Political Methodology

ad indicia spectate