Psephology and Technology, or: The Rise and Rise of the Script-Kiddie

Cross-references

- Cross-national data sources (48)
- Geo-location and Voter (34)
- Multi-Level Modelling (47)
- Social Media (44)

1 Introduction

From its very beginnings, psephology has been at the forefront of methodology and has sometimes pushed its boundaries (see e.g. King, 1997 on ecological regression). Methods such as factor analysis or logistic regression, that were considered advanced in the 1990s, are now part of many MA and even BA programs. Driven by the proliferation of fresh data and the availability of ever faster computers and more advanced, yet more user-friendly software, the pace of technical progress has once more accelerated over the last decade or so. Hence, and somewhat paradoxically, this chapter cannot hope to give a definitive account of the state of the art: The moment this book will ship, the chapter would already be outdated. Instead, it tries to identify important trends that have emerged in the last 15 years as well as likely trajectories for future research.

More specifically, the next section (2) discusses the general impact of the "open" movements on electoral research. Section 3 is devoted to new(ish) statistical methods – readily implemented in open source software – that are necessitated by the availability of data – often available under comparably open models for data distribution – that are structured in new ways. Section 4 is a primer on software tools that were developed in the open source community and that are currently underused in electoral research,

whereas the penultimate section discusses the double role of the internet as both an infrastructure for and an object of electoral research. Section 6 summarises the main points.

2 Open Source, Open Data, Open Science

Like many other subfields in the social sciences, psephology is heavily affected the rapid progress in computer and information technology. The two most significant developments in this respect are the twin open-source and open-data revolutions. *Open source* software has its roots in the free software movement of the 1980s (Lakhani and Hippel, 2003), a rebellion against increasingly more restrictive software licences that, amongst other things, aimed at patenting algorithms and banned the "reverse engineering" of software installed on private computers. Proponents of the free software movement, on the other hand, made their software available for free ("free as in free beer") and gave everyone and anyone the licence to modify their programs as they saw fit ("free as in free speech"), which required laying open the source code. The spread of the internet in the 1990s then facilitated large-scale collaboration on free software projects and gave rise to the current idea of open source software that is embodied in Raymond's (1999) manifesto "The Cathedral and the Bazaar", which emphasises the idea of distributed and only loosely co-ordinated teams as a strategy for quick and efficient development.

Whereas the free software movement had a certain anti-establishment bent, many of the largest and most successful open source projects, such as the Linux operating system, the Apache web server or the Firefox browser series, which collectively power much of the current internet, are happy to rely on the support of corporate backers who donate money, resources, and the time of some of their staff. In other instances, large companies have even created open "community editions" of their existing programs, or designed them as open source applications in the first place (Google's Android operating system). Companies may do this to raise their profile, or in order to attract the best software engineers for their commercial projects, but two other motives are more interesting: They may want to use the open source software instead of a closed-source alternative to generate and deliver their own products (e.g. tech companies relying on Linux for running their server farms), or they may offer a service that is based on the open-source software (professional support or hosted versions). Either way, corporate support for open source makes commercial sense – neatly illustrating Olson's (1965) argument about big players rationally investing in public goods – because open source is a, as Raymond suggests, a highly efficient model for organising large projects: It incorporates feedback from the user base almost instantaneously and turns the most capable and committed users into developers.

Open source is highly relevant for psepholog not only because it helped to build much of the internet infrastructure and some key tools – R (Ihaka and Gentleman, 1996; Crawley, 2013), Python (Lutz, 2013), and a plethora of others – but also because it has become the template for other "open" revolutions that impact on electoral research. In the broadest sense, *open data* refers to the idea that research data, or data that could be used for research, should be as accessible as possible. As such, it is old news. In the quantitative social sciences, data archives such as the Roper Center (`http://ropercenter.cornell.edu/`) or Michigan's Survey Research Center (`https://www.src.isr.umich.edu/`), which collect, archive, and disseminate existing data for secondary analyses, were established in the late 1940s. Patterns of co-operation and exchange between (European) archives were formalised with the formation of the Council of European Social Science Data Archives (CESSDA, `http://cessda.net/`) in the 1970s (Karvonen and Ryssevik, 2001, p. 45). In the public sector, one could argue that the practice of frequently publishing detailed information on key national statistics that was already well established in the late 19th century marks the beginning of open data. However, it was the key ingredients of the open source revolution – transparency, active involvement of the user base, and almost zero marginal transaction costs – that in the 2000s began to transform the production and use of data in unprecedented ways. Unless access is restricted for reasons of data protection, researchers no longer have to travel to a data archive to use a given data set, and for the distribution of data, physical media haven been all but abolished. Governments, large-scale research projects, and individual scholars are now opening up their *raw* data for download. Some agencies and some of the biggest internet companies (e.g. Google, Facebook, Twitter, and Yahoo) have even created application programming interfaces (APIs, see section 5.1) that give researchers the opportunity to access these data programmatically from a script.

The open data revolution has brought about some new problems of its own. While the body of data available for research is growing exponentially, researchers still have to know where and how to look, and the lack of a central repository and common interfaces seriously hampers progress. To be useful, data need to be stored, and, even more importantly, described and licenced in standardised ways that make them accessible and retrievable in the medium-to-long term. This, in turn, requires institutions that can be trusted, and that need funding. Moreover, the pressure on researchers to open up their own data is growing. Research councils now regularly make the deposition of research data and even the *open access* publication of the findings a precondition for funding. Similarly, more and more journals require that not only the data set itself but also the program code that generates the tables and graphs must be published along with the final article in some repository (see section 5.1).[1] While such rules reinforce traditional

[1]Pre-registration, a procedure that is becoming more prevalent in the Life Sciences and whose adoption

scientific standards of honesty, transparency, and reproducibility, many researchers are still anxious that they will be scooped if they are forced to reveal their data and methods at the beginning of a new project. Presumably due to the prevailing incentive structure, few social scientists currently adhere to the open source mantra of "release early, release often." Others, however, embrace the ideal of a (more) *open science* by posting work-in-progress on their personal homepages, opening draft chapters on social network sites for scientists, or even by moving their data and manuscripts to open source development sites such as Github, which could in theory provide an ideal environment for scientific collaboration.

3 Data, Statistical Models, and Software

3.1 Complex Data Structures and Statistical Models

Pure formal theory and simulation exercises aside, all electoral research rests on data: a body of systematic and usually quantified observations that can be used to test assumptions about the ways in which citizens, politicians, organised interests, and the media interact and thereby affect political decisions. While early studies emphasised the importance of macro factors (Siegfried, 1913) and of clustered sampling and mixed methods (Lazarsfeld, Berelson, and Gaudet, 1944), the lasting influence of the American Voter (Campbell et al., 1960) has led many researchers to focus on micro-level data coming from nationally representative samples of mass publics for much of the 1960s, 1970s, and 1980s.

But theory suggests that credible (or at least plausible) accounts of human behaviour must encompass not just the individual (micro), but also the societal (macro) level, and ideally various "meso" layers and structures in between (see Coleman, 1994 for the general line of reasoning and Miller and Shanks, 1996 for an application to election studies). The thrust of this argument eventually led to a renewed interest in contextual variables and their effects (Jennings, 2007, pp. 35–38). From the late 1990s and early 2000s on, national election studies and comparative surveys alike began to include territorial identifier variables such as electoral district codes in their datasets. Using this information, it is possible to match data on individual respondents with government figures on the economy, on migration, and a whole host of other variables that can plausibly affect voting behaviour, while Multi-Level regression (see chapter 47) is a convenient tool for estimating the size of the alleged effects and their associated standard errors. Supple-

in Political Science is now being discussed (Monogan, 2015), goes on step further by demanding that researchers submit sampling plans, outlines of the intended analyses, and mock reports to a journal that are peer-reviewed before they even begin to collect new data.

menting micro-level information with contextual variables leads to "nested" data, where each level-1 unit (respondent) belongs to one (and only one) level-2 unit (electoral) district. Each level-2 unit may in turn be part of one (and only one) level-3 unit (say, a province), resulting in a tree-like structure.

Multi-Level regression modelling with contextual covariates derived from official sources has become almost the *de facto* standard for analysing both large-scale comparative data sets (see chapter 48) and case studies of nations for which sub-national data are available. While the technique provides asymptotically correct standard errors and opens up a number of flexible modelling options (see section 3.2.1), it is no panacea. When nations are the relevant contexts, their number is often too low for Multi-Level Modelling (Stegmueller, 2013), and one may well ask if it makes sense at all to treat countries as if they were a random sample from a larger population (Western and Jackman, 1994). Comparing political behaviour within subnational units across nations is more informative and often more appropriate, but suffers from specific limitations, too: Even within the European Union's complex and comprehensive system for the Nomenclature of Territorial Units for Statistics (NUTS, see Eurostat, 2015), subnational units that are supposed to be on the same level may differ vastly in terms of their size, population, and political, social and cultural relevance.[2]

Moreover, the integration of government statistics as regressors into a Multi-Level Model does not nearly exhaust the complexity of data that are now available for analysis. Building on earlier work by Lazarsfeld and Menzel (1961), Hox (2010) has developed a useful typology that clarifies the possibilities. On each level, there are *global* variables, which reflect inherent properties of the objects on the respective level. They are inherent in so far as they can neither be constructed by aggregating features of lower-level objects, nor by disaggregating features of higher-level contexts. Traditional (statistical) models of voting behaviour have focused on global variables at the indvidual level (level 1): an invidual vote for the Democrats is linked to the voter in question being female, unemployed, and identifying as a Democrat. A prototypical Multi-Level Model would add the unemployment rate and the ethnic composition of the electoral district as level-2

[2]NUTS-1 corresponds to the 16 powerful federal states in Germany, to clusters of provinces, states, or communities that have been grouped together for purely statistical purposes in Austria, Spain, and the Netherlands, and does not exist at all in many of the smaller countries (e.g. Croatia, Denmark, Luxembourg, or Slovenia). The lower-tier NUTS-2 level is equivalent to the federal states in Austria, the autonomous communities in Spain, the Regions in France, and the Provinces in the Netherlands, which all have their own elected assemblies. In other states such as Bulgaria, Finland, Germany, Romania, or Slovenia, NUTS-2 areas exist solely for the purpose of national planning and attracting EU funding, and citizens will be unaware of their existence. Similarly, NUTS-3 may be a district (Germany), a group of districts (Austria), a province (Denmark, Spain, Italy), a region (Finland), a statistical region (Slovenia), an island (Malta), or may not even exist (Cyprus, Luxembourg).

Level	1		2		3		...
Type of variable	global	→	analytical				
	relational	→	structural				
	contextual	←	global	→	analytical		
			relational	→	structural		
			contextual	←	global	→	
					relational	→	
					contextual	←	

→: Aggregation
←: Disaggregation

Source: Adapted from Hox (2010, p. 2)

Figure 1: A Typology of complex data structures

regressors. These are *analytical* variables, which are created by aggregating global features of the lower-level units to form upper-level averages, ratios, or percentages. As a corollary, these variables can enter the model simultaneously on multiple levels (see section 3.2.1).

Other properties of the district may also be meaningful additions to the model, but they cannot be understood as an aggregation of individual-level qualities or disaggregation of higher-level features and are hence global variables at the district level. Gender and political experience of the main candidates are cases in point. Because there is no variable at the lowest level that would correspond to them, they are strictly *contextual* for individual voters and can enter the model only once, at the upper level.

Finally, *relational* data convey information regarding the ties (e.g. presence and intensity of face-to-face contacts) between objects on the same level. Such network data are crucial for any micro-sociological explantation of voting behaviour: Obviously, a person that is the hub of a Democratic clique of friends is more likely to turn out to vote, and to vote in accordance with her peers than someone who is socially isolated. Like global/analytical variables, network data can enter a Multi-Level Model simultaneously on multiple levels: Information on relations between individual voters within a district may be aggregated to form *structural* variables at the upper level, e.g. to compare districts with dense/sparse or homogeneous/fragmented communication networks.

Network data are extremely attractive in theory. But they introduce an additional level of complexity and require specialised statistical methods, because a tie by definition involves two actors (see section 3.2.3). In addition, the collection of relational data necessitates specific (cluster) sampling plans, because a large number of the members of a

given network needs to be surveyed to assess the properties of the network itself. This, in turn, raises issues of representativeness, data confidentiality, and cost-effectiveness and goes against the dogma of the nationally representative sample.

Election surveys sometimes contain items refering to so-called *egocentric networks*, e.g. they might ask the respondent how many people she talks politics with, whether these are friends, family members, or just acquaintances, and how often she disagrees with them. But this information will be biased by the respondent's perceptions and provides only a fractional glimpse into the full network, as normally not even the ties amongst the respondent's immediate contacts can be reliably recovered.

As a readily available alternative, students of electoral behaviour are now turning to social media, where large and mostly complete political communication networks can be sampled and observed with ease. Just how well insights from these networks generalise to offline behaviour and the voting population as a whole is a different question. Either way, statistical procedures for analysing social networks are currently in the process of becoming part of the tool kit for electoral research.

Besides Multi-Level and network data, the use of spatial or geo-referenced data is another emerging trend in electoral studies. A geo-reference is simply a set of coordinates that locate an object in space. Coordinates can either define a point or an area (polygon). In the most simple sense, the territorial identifiers mentioned above record that a voter is living in a given (usually large) area and hence are geo-references, too. More precise coordinates for voters (e.g. census blocks, ZIP code segments, electoral wards, street addresses, or even GPS readings), however, allow researchers to locate voters within much smaller contexts, for which census and market research data – in other words, global and analytical variables that can be integrated into a Multi-Level Model of electoral choice – may be available. Whilst many researchers are familiar with the idea of coarse geo-references, the availability of very fine-grained data as well as the growing awareness of spatial dependencies necessitates specialised software and models for the proper analysis of geo-referenced data (see section 3.2.4)

3.2 Statistical techniques and software implementations

3.2.1 Multi-Level Models and Structural Equation Models

As outlined above, students of electoral behaviour routinely collect data that, reflecting the Multi-Level nature of the underlying theoretical explanations, exhibit complex structures. Statistical Multi-Level Models, which are also known as "mixed models" or "random coefficient models" are the most adequate means to deal with such data.

They account for the correlation of unmeasured disturbances within a given context and hence provide correct standard errors for the effects of macro-level variables.

Moreover, they model context specific disturbances in the most efficient way possible by treating them as random. This is best illustrated with an example: In a study of N voters living in K electoral districts that aims at explaining individual turnout, one could try to capture the effects of unmeasured district-level variables (say local social capital) by introducing district specific intercepts (dummy variables). But this strategy has negative consequences for the identification of the model and becomes inefficient and impractical very quickly as the number of districts which are sampled grows (Steenbergen and Jones, 2002). A statistical Multi-Level Model will replace the *K-1* estimates for the local intercepts with a single estimate of their variation over districts (a *random intercept*) and thus dramatically reduces the number of parameters.

Moreover, Multi-Level Models also provide for a number of additional advanced modelling options. If there are good reasons to believe that the impact of an explanatory variable (say ideology as measured by left-right self-placement) on turnout will vary considerably across the K districts, the analyst can specify a *random effect* for this variable, which supplements the estimate for the average effect of ideology (the traditional point estimate) with an estimate of its variation. As the name implies, random effects are adequate if the variation in the effect of an independent variable can plausibly be treated as random.

If, on the other hand, the impact of a variable varies in a systematic fashion, this can be modelled by specifying a *cross-level* interaction, e. g. between ideology (a micro-level variable) and the number of candidates standing in the district. Cross-level interactions need not be confined to variables that are as conceptually different as the two in this example. On the contrary, theory often suggests that a variable such as unemployment could in essence interact with itself, albeit on different levels, hence entering the model thrice: as an individual feature (a global variable on the micro-level), as an analytical variable (the unemployment rate on the district-level), and as an cross-level interaction between the two. A high unemployment rate may reduce the propensity to participate in an election for *all* citizens, and individual unemployment status will normally depress turnout in an even stronger fashion. But this micro-level effect may well be confined to low-unemployment level environments, whereas individual unemployment may have no such negative impact or even increase the likelihood of voting in districts where the high unemployment rate attracts community organisers and other political entrepreneurs. Multi-Level Models are ideally suited for disentangling such complex causal relationships.

They can also deal with complexly structured political contexts that may have many tiers (voters within households within wards within municipalities within districts within provinces …), and that may cross-cut and overlap instead of forming a neat, tree-like hierarchy: A voter is not just affected by the characteristics of the electoral district she is

living in but has been politically socialised in completely different surroundings. Whilst Multi-Level Models can accommodate such complex structures, convergence will generally be slow, and estimates may be unstable. As with all other aspects of modelling, analysts should therefore strive for parsimony. If the there are not variables at the higher levels and if the objective is simply to reflect the multi-stage nature of the underlying sampling process, traditional survey estimators or even Huber-White standard errors that account for clustering may provide a fast and robust alternative to a fully specified Multi-Level Model.

Having said that, Multi-Level Models are a very flexible tool, as contexts need not be defined in spatial terms. For the analysis of panel data, it often makes sense to think of individual respondents as "contexts" for the interviews conducted in successive panel waves. Particularly when panel data are imbalanced or collected at irregular intervals, Multi-Level models can alleviate some of the problems that plague the traditional approaches to panel data.

Another statistical technique that has become indispensable for students of electoral behaviour is Structural Equation Modelling (SEM). SEM is an extension of traditional factor analysis that lets researchers specify multi-indicator measurement models for otherwise unobservable (=latent) theoretical constructs such as political attitudes. It is attractive, because it can simultaneously estimate coefficients for whole systems of equations, and because it can incorporate measurement models for attitudinal variables that account for relatively unreliable indicators. If the measurement models hold, SEM can also provide unbiased estimates of the equally unobservable, "structural" relationships amongst the latent variables. Given adequate data, it is possible to map a whole system of constructs and hypotheses about their relationships onto an equivalent system of equations.

In the past, its application in election studies was somewhat limited by the fact that they required measurements on a continuous scale that were distributed multivariate normal, whereas the key dependent variable in election studies as well as many relevant independent variables are categorical and usually distributed with considerable skew. In the 1990s, however, new estimators were developed that can accommodate non-normally distributed continuous data. In addition, generalisations of the original model allow for ordinal and nominal indicator variables and even for categorical latent variables (Jöreskog, 1990; Jöreskog, 1994; Muthén, 1979; Muthén, 2002). Moreover, Multi-Level Models and Structural Equation Models are closely related (Muthén, 2002; Skrondal and Rabe-Hesketh, 2004) and can be combined to form Multi-Level Structural Equation Models.

Until recently, the estimation of Multi-Level or Structural Equation Models required specialised (if relatively user-friendly) software: HLM or MLWin for Multi-Level Mod-

elling, LISREL, EQS, or AMOS for SEM, and MPlus for either. This is no longer true: Recent versions of Stata – currently the most popular general purpose statistical package in political science – can estimate all but the most complex Multi-Level and Structural Equation Models and so greatly extend the potential user base of these techniques. SPSS, another popular package, has some Multi-Level capabilities and works closely with AMOS, an SEM software that SPSS Inc. acquired in 2003 before it was in turn bought by IBM in 2009.

Perhaps more importantly, there are packages available for the R programming language that provide similar features: Lme4 and Rstan for Multi-Level Modelling, and Lavaan and Sem for SEM. While they may be slightly less capable, slower, and generally more clunky than the commercial software, they are, like any other R-package and the core of the language itself, open source and freely available for almost any combination of hardware and operating system. Moreover, while they may lack professional documentation and customer service, they are supported by a global community of enthusiasts, scriptable in a fully-fledged programming language with flexible data structures, and tie into the ever growing eco-system of more than 6000 user-written packages for R that aim at implementing the latest developments in statistics.

3.2.2 Bayesian methods

Most electoral researchers were trained within a "frequentist" framework of statistical reasoning that relies on the idea of a random sampling process that could be endlessly repeated under essentially identical conditions. So far, they have shown only modest interest in the (sometimes exaggerated) benefits of an alternative statistical framework: Bayesian statistics (Jackman, 2004). There are at least two causes for this inertia: The frequentist paradigm closely resembles the pattern of taking national large-scale random samples of the general population that has been the workhorse of election studies for most of the last seven decades, and in such large samples Bayesian and frequentist estimates will normally closely resemble each other.

But whether applied researchers like it or not, the ever more popular Multi-Level Models are Bayesian models at their core (Gelman and Hill, 2007). While many political scientists still have some reservations regarding the underlying paradigm (or might be blissfully unaware of it), Bayesian statistics keeps making inroads into electoral research. There are a number of reasons for this. First, Bayesian models can sometimes be tailored to a problem for which no off-the-shelf frequentist solution has been implemented in any statistical package. Models that aim at predicting the distribution of seats in parliament from a rolling series of published opinion surveys are case in point. Second, Bayesian statistics may be able to provide an estimator that is better in terms of bias and efficiency than any frequentist alternative, as it is the case with Multi-Level Models and some

SEMs. Third, Bayesian statistics, which for most of its existence was a rather arcane pasttime because of the computational demands implied, only gained practical relevance for applied researchers with the twin advent of simulation-based methods and affordable fast processors in the late-1990s-to-early 2000s. Even a decade ago, getting Bayesian estimates in MLWin for a reasonably complex Multi-Level Model could easily take an hour or more on a then-modern desktop computer, much as it was the case with SEM in the 1990s.

At the moment, most Bayesian estimation still requires access to specialised software (Winbugs, Openbugs, Jags, Stan ...), preferably via R. However, the implementation of Bayesian analysis in recent editions of Stata (from version 14 on) could be a game-changer in this respect.

3.2.3 Networks

So far, election studies have mostly missed out on the renaissance of Social Network Analysis (SNA) in political science (for some notable exceptions see e. g. Huckfeldt and Sprague, 1987; McClurg, 2006). Although interest in *relational* or network data has grown exponentially in Political Science, psephology has been somewhat late to the party because relevant data are generally not available. While large societies may display the properties of a "small-world" network in which everyone is related to everyone else through a relatively small number of contacts (say six), such network structures are very sparse and will rarely have an effect on political behaviour. Social embededness certainly plays a role for opinion formation and political behaviour, but mainstream election studies cannot hope to uncover the relevant networks. Traditional community studies as well as explorations of online communities, on the other hand, can do just that.

Although it is far from clear if and how findings gained here generalise to the electorate as a whole, statistical procedures for analysing social networks are currently in the process of becoming part of the tool kit for electoral research. Understanding these methods can present a formidable challenge.

By definition, network data break the mould of traditional data analysis, where cases correspond to the rows of the data matrix, and variables to its columns. In network applications, cases form both the rows and the columns of an (adjacency) data matrix, whose cells represent the existence, direction and possibly strength of ties between them. Recording traditional variables requires a second data matrix, specialised software, and, more importantly, and adjustment of the analyst's mind set.

Once collected, data on ties between actors can be employed to calculate three broad classes of statistical measures (Knoke and Yang, 2008): indices that reflect the position of an individual within the local or global network (e.g. one's centrality), measures that

refer to features of an actual or potential tie between two actors (e.g. the importance of this tie for the coherence of the network as a whole), and statistics that describe some features of the network as a whole (e.g. the degree to which it resembles the "small-world" scenario outlined above). Often, the aim of SNA is chiefly descriptive and the analysis would end with their calculation and interpretation, but in principle, all network measures can subsequently be employed as dependent or independent variables in a regression framework.

Relational data do not fit into the single-matrix paradigm of general statistical packages such as Stata or SPSS. Moreover, before the rise of social networking sites, there was little commercial interest in SNA. Therefore, most software that is potentially useful for students of electoral behaviour is developed by academics (often as an open source project) and available for free or at a very modest price. Historically, UCINET, which was created in the early 1980s and has been under constant development ever since has been a very popular choice. UCINET is grounded in the tradition of (mathematical) sociology and incorporates a multitude of procedures for manipulating and analysing relational data. However, according to its authors, many of these procedures become tediously slow in networks with more than 5000 nodes. Pajek and Pajek XXL, on the other hand, are slightly newer programs specifically geared towards large and very large networks of several million nodes. Their user interface is idiosyncratic, and the terminology used in the documentation as well as many of the procedures may be unfamiliar to social scientists, as the authors have their roots in mathematical graph theory and computer science. However, Pajek is unrivalled in terms of speed and sheer processing capacity.

UCINET, Pajek, and other SNA software make it possible to perform analyses that are unfeasible with standard statistical software. However, moving one's data from a standard software suite to an external program for network analysis, then back to the general purpose package for further analysis is a disruptive, tedious, and error-prone process. The various SNA packages that exist for the R system are therefore an attractive alternative to the stand-alone SNA programs. The most generally useful are Statnet (a "meta" package that includes many procedures from more specialised packages), and Igraph, which seems to be slightly more accessible (and is also available as a package for the Python language). In all likelihood, either package will fulfil all but the most exotic needs of psephologists.

3.2.4 Geo-spatial analysis

Geo-spatial analysis is a broad term that encompasses at least two distinct (if related) approaches: the use of geographical variables in "normal" regression models of electoral behaviour on the one hand, and the estimation of specific statistical models that account for spatial dependencies on the other.

The first approach may simply use geo-references to merge micro data with contextual information (see section 3.1). Under more advanced scenarios, psephologists will calculate geographical variables (most often distances) from sets of geo-references.

This is best illustrated by an example: For various theoretical reasons, voters should *ceteris paribus* prefer local candidates, i. e. candidates that live closer to a given voter's residence than other candidates. If candidates are obliged to have their home addresses on the ballot paper and the addresses of voters are known,[3] the spatial distance between candidates and their prospective voters can be calculated (Arzheimer and Evans, 2012; Arzheimer and Evans, 2014). This variable varies across voter-candidate pairs within districts (unless voters live at the same address) and is therefore a global variable at the level of the individual voters.

Geo-spatial methods are required for (1) the translation of addresses into physical co-ordinates (a step known as *geocoding*) and (2) the calculation of various distance measures (e. g. travel time by car or public transport). Apart from the calculation of straight-line distance, which is a purely geometrical problem, the second step requires access to digital road maps, timetables, data on congestion, and routing algorithms. However, once the distance has been calculated, the analysis can proceed with the usual linear and non-linear regression models, which may account for nesting or clustering of the observations by imposing a structure on the variance-covariance matrix.

Various types of *spatial regression* models take this idea one step further. They correct for dependencies amongst the observations by taking the spatial co-ordinates of cases into account and adjusting the structure of the variance-covariance matrix accordingly.

The relevance of spatial regression models for psephology is most obvious in the case of district-level aggregate analyses: Whereas standard regression models assume that disturbances are identically and independently distributed, it stands to reason that neighbouring[4] districts will be affected by similar disturbances and hence will display a pattern auto-correlation that renders standard errors dubious at best. In spatial regression, the matrix of distances between the centroids of the electoral districts can be employed to estimate this auto-correlation, which in turn can be used to derive corrected standard errors within a spatial regression model (Ward and Skrede Gleditsch, 2008). Spatial regression could be applied to individual-level data too, but it is generally easier and often also more appropriate in terms of the underlying theoretical assumptions about causal mechanisms to use a Multi-Level Model (possibly involving more than

[3]For reasons of data protection, usually only an approximate geo-reference of the respondent is recorded.

[4]Being neighbours is a somewhat fluid concept, as these shared influences will be stronger where districts are physically closer and less pronounced, yet still present, where a pair of districts is further apart. This scenario is very different from nesting, where there are clearly delineated, fixed groups of lower-level units.

two levels) that accounts for nesting within political-administrative contexts.

Mapping and processing geo-referenced data traditionally required access to and training in the use of a Geographical Information System (GIS). A GIS is essentially a relational database with special capabilities for dealing with 2D- and 3D-coordinates. GIS software tends to be expensive, proprietary, and complex. In recent years, however, government agencies and other organisations aiming at opening up their data have set up websites that hide at least some of the complexity of the underlying system. In the most simple case, users may create chloropleth maps, or look up data for a single or a limited number of localities. More useful systems allow one to download pre-built or customised tables in some machine-readable format that can be merged with existing individual-level data. In very few ideal cases, there is an API that researchers can access programatically (see section 5.1).

Moreover, algorithms for collecting, storing, and processing geo-referenced data are now freely available and have been implemented in a host of stand-alone programs and/or packages for the R system. GRASS (Geographic Resources Analysis Support System) is a fully featured GIS that has a wide range of applications in engineering, the natural science, and the social sciences. GRASS runs on all major operating systems. It can be used both interactively through its graphical user interface (GUI) and programmatically via scripts. Its real power, however, lies in its interfaces with two popular programming languages: Python and R. Through these interfaces (Pygrass for Python and Rgrass6/7), users can at the one hand program the GRASS system and extend its capabilities. On the other hand, researchers who regularly run their analyses in Python or R can selectively make use of data stored in GRASS and of the nearly 2,700 industry-grade functions available in the system. QGIS is a more light-weight alternative to GRASS. While it interfaces with R and Python, too, it is mostly geared towards interactive use.

In many cases, however, analysts working in R or Python will want to altogether avoid the overhead of plugging into a GIS. Much of the functionality of traditional GIS software is now available in the form of addons for these two languages. R in particular currently has more than a hundred packages for loading, manipulating, analysing, and mapping geo-referenced data (`https://cran.r-project.org/web/views/Spatial.html`).

4 Tools for successful, reproducible research

The previous two sections have dwelled on the rapid pace of technical progress in psephology. Somewhat paradoxically, this section suggests that in the face of ever more complex data and software, psephologists should turn to very basic tools, concepts and techniques that computer scientists developed decades ago: plain-text files and editors, directories (folders), and some utilities commonly used in medium-sized programming projects. As

the old saying goes: In election studies, regression is progress.

4.1 Establishing a reproducible workflow

Data analysis involves a number of distinct phases (see Long 2009, chapter 1 for a similar outline):

1. Data must be collected, either by the researchers themselves or by some third party, and stored electronically
2. These machine readable data need to be transferred to the researchers, usually via the internet
3. The data must be recoded or otherwise normalised, possibly after being converted to some other format first.
4. A number of exploratory analyses and preliminary models are run on the data, perhaps using more than one computer program
5. The researchers settle on a small set of final analyses and models whose results are stored
6. For presentation and publication, graphs and tables are produced from these results, possibly using additional software

To be reproducible by the original researchers and their peers, every step as well as the rationale behind the decisions involved must be documented. Realistically, that means that as much as possible of the whole process should be automated by means of *scripts*: short sets of instructions for a computer program. Graphical user interfaces are useful for getting to know a program, and possibly for putting the finishing touches on graphs for publication, but scripts are infinitely more powerful, efficient, and reliable. When properly commented, scripts are also self-documenting, although researchers should strive to keep a separate research journal. For smaller projects, presentations, and teaching, researchers may even want to pursue a "literate programming" (Knuth, 1984) approach that combines code for several programs, text for publication, and documentation in a single document, from which intermediate tables and graphs as well as slides and PDF documents may be produced using either the Knitr package for R or the even more general Orgmode package for Emacs (see below). However, while literate programming is attractive in principle, it may not scale well to larger projects.

Most statistics packages have simple script-editing capacities built in, but in the long term, it is more efficient to use stand-alone text editors, which offer much more powerful

editing features as well as syntax highlighting, proper indentation, and basic project managing capabilities. One of the most quirky and powerful of these editors is Emacs (`https://www.gnu.org/software/emacs/`), which was first released in the mid-1970s and has been under active development ever since. Despite its age, interest in Emacs has surged in recent years, and many quantitative social scientists swear by it. Emacs can be endlessly customised and extended, which can be baffling for new users. Cameron et al. (2005) provide a useful introduction, but documentation for the many more features and extensions is best searched on the internet. Psephologists may also want to install one of the configurations aimed specifically at social scientists that can be found online.

With the right set of extensions, Emacs supports almost every scripting language known to humankind, including the command languages of statistical packages such as Julia, OpenBUGS/JAGS, R, S-Plus, Stan, Stata, and SAS. At a minimum, "support" means syntax highlighting, indentation, and checking for balanced parentheses. Moreover, Emacs normally gives access to the respective help systems for these languages and can find documentation for related functions. It can insert boiler plate code (e.g. loops) and can execute snippets of code or whole scripts. Emacs was designed as an editor for computer programmers and so has the ability to keep track of variables and look up the definition of functions across an arbitrary number of files, making use of text tools such as Diff, Grep, or Find, version control systems such as Git (more on them below). The more complicated the toolchain becomes, the more it shines, as R, Stata, Python, and many other applications can be conveniently managed from a single keyboard- and script-centric interface.

4.2 Buildtools, revision control, and other open source goodies

Ideally, there should be a separate script for each of the six steps outlined in section 4.1. Shorter scripts are easier to maintain, and it would be inefficient to re-run the whole process only to add a horizontal line to a table. It is also vitally important that data are only ever edited in a non-destructive way: Each script must save its results as a new file, keeping the data collected and transferred in steps one and two in pristine condition. It is also good research practice to keep all files belonging to a given project in a directory of their own, and to create separate sub-directories for scripts, graphs, tables, and datasets (Long, 2009).

Once a project grows beyond a handful of individual scripts, further automation of the process or meta-scripting becomes a necessity, because the individual jobs need to be executed in a certain order. In principle, a degree of automation can be achieved within the statistical package of choice itself: Both Stata and R are capable of processing scripts that it turn "include" or "source" other scripts. Moreover, both programs have

rudimentary infrastructure for starting external programs and can so, at least in theory, manage a tool chain. In practice, however, it is easier and less error-prone to rely on an external scripting language, e.g. Python or the scripting language of the operating system's native command line interpreter (shell), to manage complex workflows.

If some of the tasks involved are time-consuming or otherwise expensive (i.e. model estimation by numerical means or metered data acquisition from the internet), psephologists should rely on "build tools": software that is normally used by computer programmers to compile ("build") complex software from hundreds of text files via a potentially large number of intermediary files. If a single text file is edited, it is normally sufficient to recompile a small fraction of the whole project that is directly affected by this change. Build tools can identify, manage, visualise and most importantly exploit such dependencies, thereby making substantial efficiency gains possible.

On average, workflows for software project are more complex than workflows for the analysis of electoral data by several orders of magnitude, but psephologists can still benefit from learning to use build tools. This is best illustrated by an example. Consider the following simple workflow:

1. Download (with R, or with a specialised program such as wget) a data set (say the European Social Survey) from the internet *if the file on the web has changed*
2. Save a relevant subset of the data after recoding some variables
3. Load the subset, estimate some complex model, and save the parameters to a file
4. Illustrate the findings by
    - Producing a number of graphs from the parameters and save them as separate files
    - Producing a number of tables from the parameters and save them as separate files
5. Generate a PDF report with the LaTeX document preparation system by processing a text file that includes the graphs and tables

For an efficient and managable workflow, each task should be performed by a single program acting on a single set of instructions (either a script or simply a number of options and arguments submitted when the program starts). Moreover, each task takes one ore more inputs, and leaves behind one or more outputs.[5] The way in which these

[5]Ideally, the number of inputs and outputs should be as low as possible (i.e. by writing one individual script for every graph that goes into the final document), but that can become very tedious and is not always feasible.

individual tasks are listed makes it very easy to recognise the dependencies amongst them: If a new version of the European Social Survey is published, all steps must be repeated in that exact order. If, on the other hand, the researcher decides to change the coding of the variables (step 2), the estimator employed by the model (step 3), or the look of the graphs (step 4), only the subsequent steps must be repeated. Incidentally, the latter modification would not require rebuilding the tables: If the dependencies were visualised as a tree, both tasks would appear on the same level, as they are completely independent of each other. In a computing environment with sufficient ressources, they could be executed in parallel, thereby further speeding up the process.

Build tools such as the venerable Make program (Mecklenburg, 2005, generally available on Unix-like systems) and its many modern successors require that the dependencies are specified in yet another textfile. While this may sound like a chore, it is usually just a matter of writing down which script generates what files ("targets") from which inputs. Moreover, this helps clarifying and streamlining the workflow. Once this set of rules is in place, the build tool will analyse the dependencies and execute the tasks in the required order. After this initial run, targets will only be regenerated if the scripts or inputs from which they originate change.

A final tool that psephologists should borrow from the world of software development are revision control systems. Most researchers will be (painfully) aware of the value of automated backup systems, which retain a number of old copies to avoid the situation where a good backup is replaced by a corrupted copy. Modern systems usuallly provide a number of hourly or daily snapshots alongside increasingly older (weekly, monthly, yearly) copies. Revision control systems take this idea of snapshots one step further by retaining a complete history of changes to each (text) file in a project directory.[6] Modern revision control systems such as the somewhat unfortunately named Git (Loeliger and McCullough, 2012) can track the whole state of a directory and quickly reset all files in a directory to the state in which they were yesterday evening, or show which changes were made to a specific file since Monday night. They provide tools for finding the exact point at which some changes in the way the variables were recoded stopped the model from converging or brought about a dramatic change in the estimates further down the line.

But most importantly, using a revision control system introduces another layer of reliability and reproducibility. Modern revision control sytems cannot just easily revert unwanted changes to one's project files, they can effortlessly maintain an arbitrary large number of timelines ("branches") for a project directory. This is great tool for testing

[6]Unless they are kept in sync with a remote repository, revision control systems operate on local files only and hence do *not* protect against data loss through hardware failure. Researchers still need to make sure that they backup their working files as well as their revision control repository regularly.

code and ideas: One can easily try out a variety of operationalisations, model specifications, or graphical styles in various branches, once more recording all the changes made to the files, then switch back to a more stable line of development that represents the current state of the analysis and selectively copy over anything that worked. Revision control sytems are based on the assumption that each of these changes should be documented in a comment and so strongly encourage the analysts to keep a log of the rationale behind the myriad of tiny decisions they take while analysing their data and presenting their findings.

Like many other tools discussed in this chapter, revision control systems have been used by computer programmers for decades. Their modern incarnations are designed to deal with millions of lines of code spread across hundreds of files, on which large teams of developers may work concurrently. Psephologists may well think that a system like Git (which is relatively difficult to learn but can be tamed through a number of GUIs) is ridiculously overpowered for their needs. However, experimenting with one's code and data in a safe environment where each change, each experiment is documented and can be reverted, modified and even re-applied at any later point is ultimately much more rational, rewarding, and productive than the common practice of endlessly commenting in and out lines of code, or creating lots of increasingly cryptically named scripts whose exact purpose we cannot remember after a couple of weeks.

5 The Internet as an Infrastructure for and as an Object of Electoral Studies

5.1 Infrastructure

Psephology has been transformed by the availability of large-scale comparative opinion surveys such as the ISSP, the EES, or the Eurobarometer series (see chapter 48). The websites of CESSDA members and other large archives are now the default option for the distribution of these datasets, allowing speedy and cost-efficient proliferation of data, whereas physical media (e.g. DVDs or CD ROMs) have been phased out, unless partiuclarly restrictive usage rules apply.

While the archives are unrivalled when it comes to providing documentation and long-term safe storage for large numbers of datasets, preparing one's data for a release through the archive system is seen as a chore by many researchers. Thus, there has always been a tradition of more informal data-sharing in psephology with friends and close colleagues. With the advent of the web, individual researchers and small teams began to put their datasets on personal or departmental websites. However, data on such

sites is often difficult to find, because there is no central catalogue, and may disappear at any moment, because it is not backed up by a professional infrastructure.

Moreover, data may be stored in any number of formats and without even minimal documentation. The open source Dataverse project (`http://dataverse.org/`) and some related initiatives aim at solving these problems by providing means for the (semi-)automatic conversion, documentation, versioning, and retrieval of data. They also provide globally unique identifiers and checksums to solve the problem of data integrity. Journals, research groups, and individual scholars can easily create their own repositories to facilitate re-analysis and replication. Dataverse and similar software go a long way towards making data from smaller, often self-funded projects that would otherwise be lost to the scientific community available for secondary analyses. But they still rely on a professional and sustainable IT infrastructure. At the moment, this infrastructure is provided for free by Harvard University and some other global players. Wheter they will continue to provide this service to the community if usage of the system picks up remains to be seen.

Besides traditional data archives and more individual repositories, countless government agencies and other public institutions around the globe have set up websites where they share parts of their records and hence have become data providers. Especially at the subnational level, the main problem with these sites is fragmentation. Even if they would adhere to common standards for their website design and the presentation of their data, finding and navigating hundreds or thousands of individual sites to collect, say, data on candidates in local elections, is obviously inefficient and often infeasible. Thankfully, governments around the world have woken up to the potential value of free access to their data and are implementing open data legislation. As a result, government-sponsored regional, national, or even supra-national "portal" or "data store" sites, which gather and distribute fine-grained data from lower levels, are becoming more prevalent. While these initiatives are often primarily aimed at policy makers and business communities, the social sciences also benefit from the emerging consensus that government data should be open in principle. For psephologists, the growing availability of geo-referenced electoral results and other statistical data (e.g. census or landuse data) is of particular importance.

In an ideal world, websites would offer the exact data set required by a researcher in a format that can be read directly into one's favourite statistical package. In reality, datasets are often offered in the guise of Excel sheets or textfiles that need to be imported. While this is not too problematic, such files are often created "on the fly" from an underlying data base according to some specifications that need to be entered manually. If the same dataset (or different variants and iterations of the same dataset) needs to be downloaded more than a couple of times, it may be worthwhile to do this programmati-

cally by means of a script. Moreover, there still exist (government) websites that present the required data not as a file for download, but rather as a series of formatted tables on the screen, possibly in a paginated format. In these cases, researchers should consider writing a "scraper", i.e. a small program that simulates the activities of a website user and stores the results as a dataset. While Python has a whole suite of libraries that make it an ideal tool for scraping tasks, some modern packages for the R system offer very similar capabilities from within the statistical package. Munzert et al. (2015) provide an excellent introduction to "scraping" and "mining" the internet. While they focus on R, the techniques and standards they discuss translate easily to workflows that are based on other tools.

Finally, many service providers – amongst them a handful of government agencies – provide "application programming interfaces" (APIs) to their data. APIs bypass the traditional website altogether. They represent complex and very specific mechanisms for interacting with the underlying database of a service as a series of simple commands for high-level programming languages such as R or Python. Using these commands, scripts can directly access these services without even simulating the activities of a human user of a website. From the point of view of the person writing a script, accessing a service on the internet is not different from calling a function that is hardwired into the respective programming language.

For instance, psephologists may have a variable that contains the addresses of candidates as stated on the ballot papers (a messy combination of names and numbers, possibly with typos). To convert these to proper geographical co-ordinates, they would want to make use of a "geocoding" service. There are APIs for various such services (e.g. Google Maps, Bing Maps, and the OpenStreetMap project), which wrap the necessary low-level instructions into a simple function call. Usage limits and possibly payment options aside, switching from one service to another is usually just a matter of applying slightly different functions to a variable. Using yet another API, the resulting coordinates could then be mapped to census tracts, for which a host of socio-economic and demographic data are available that could provide a rough-and-ready approximation of the respective environment the candidates live in.

5.2 The internet as an object

Since its inception as a research infrastructure, the internet has been thoroughly transformed. While the usual caveats about selective access and use apply, the internet's role as a political medium is becoming more and more relevant for psephologists. Current research is very much focussed on political communication as it happens on social networking platforms, with Facebook, Twitter, and Instagram being the most prominent

ones. Scraping these sites with simple scripts would not just violate their terms of use, but is virtually impossible due to their heavy use of interactive web technology, the networked nature of communication on these sites, and the sheer volume of posts. However, once more there are APIs available through which these services can be mined programmatically. While limits apply, analysts will often find the free tiers perfectly suitable for their needs. Moreover, both Twitter and Facebook have strong research departments that are open to forming partnerships with social scientists.

Research into social-networked communication on the internet is currently dominated by computer scientists and linguists, who often operate without any underlying theory of social behaviour. Psephologists interested in this field will have to learn a whole host of techniques and concepts, and will have to link these with their own substantive interests. Grimmer and Stewart (2013) provide a useful introduction to automatic content analysis, whereas Ward, Stovel, and Sacks (2011) give a *tour d'horizon* of concepts in social network theory that matter for political scientists.

Using the internet for analysing conventional media sources is less problematic in many ways. Although many publishers aim at implementing paywalls to secure their revenue streams, many mainstream outlets still put all or at least most of their content online. Moreover, Google, Microsoft and other companies have created aggregator sites that can be accessed programmatically. Using these sources, psephologists can retrospectively track the development of a given issue during a campaign, or assess the tonality of media reports on a set of candidates. In short, using the internet and a scripting language, researchers can achieve most of what would have required a host of research assistants and an extensive newspaper archive (or an expensive database subscription) only a few years ago.

The Google-supported Global Data on Events, Location and Tone (GDELT, `http://www.gdeltproject.org/`) database takes this idea one step further. GDELT, which is based on older event databases (Gerner et al., 1994), aims at automatically extracting information on actors and events from newswire reports and making them available on a global scale. The GDELT project is somewhat controversial, because its original founders fell out, and because of worries over the quality of the inferences that are drawn from the raw inputs. However, the project, which has generated enormous interest in the IR community, has great potential for psephology, too.

6 Conclusion

From its very beginnings, electoral research has been a beneficiary, and often a driver of technological and methodological progress in the larger field of Political Science. In recent years, this progress has accelerated: User-friendly software, ever faster comput-

ers, and last not least the proliferation of data mean that yesterday's advanced methods quickly turn into today's new normal. By and large, this chapter has argued that psephologists should continue to embrace technology in general and the open source and open data revolutions in particular. As the examples in the natural sciences (e.g. biology) show, psephologists can do more, and more reliably, if they think a little bit more like software developers and make use of freely available tool chains that haven been tried and tested for decades in much harsher environments.

There is, however, a flip side. Technology is a valuable tool, but it can be a distraction, too, and psephologists should never lose sight of their core competency: the ability to put singular findings into a larger context, making use of nearly a century of theory-building. The world is full of "data scientists", who will happily and rapidly analyse electoral data just as they would analyse any other kind of data. Trying to compete with them on a purely technical level would be a hopeless endeavour. As a profession, we can only stand our ground if we can base our own analyses on profound theoretical insights.

References

Arzheimer, Kai and Jocelyn Evans (2012). "Geolocation and Voting: Candidate-Voter Distance Effects on Party Choice in the 2010 General Election in England". In: *Political Geography* 31.5, pp. 301–310. DOI: 10.1016/j.polgeo.2012.04.006.

— (2014). "Candidate Geolocation and Voter Choice in the 2013 English County Council Elections". In: *Research & Politics*. DOI: 10.1177/2053168014538769.

Cameron, Debra et al. (2005). *Learning GNU Emacs. A Guide to the World's Most Extensible Customizable Editor*. 3rd ed. Sebastopol: O.

Campbell, Angus et al. (1960). *The American Voter*. New York: John Wiley.

Coleman, James S. (1994). *Foundations of Social Theory*. Cambridge, London: The Belknap Press of Harvard University Press.

Crawley, Michael J. (2013). *The R Book*. Chichester: Wiley.

Eurostat (2015). *Nomenclature of Territorial Units for Statistics NUTS 2013/EU-28. Regions in the European Union*. Luxembourg: Publications Office of the European Union, URL: http://ec.europa.eu/eurostat/documents/3859598/6948381/KS-GQ-14-006-EN-N.pdf/b9ba3339-b121-4775-9991-d88e807628e3.

Gelman, Andrew and Jennifer Hill (2007). *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge: Cambridge University Press.

Gerner, Deborah J et al. (1994). "Machine Coding of Event Data Using Regional and International Sources". In: *International Studies Quarterly* 38.1, pp. 91–119.

Grimmer, Justin and Brandon M. Stewart (2013). "Text as Data. The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts". In: *Political Analysis* 21.3, pp. 267–297. DOI: 10.1093/pan/mps028.

Hox, Joop J. (2010). *Multilevel Analysis. Techniques and Applications*. 2nd ed. New York: Routledge.

Huckfeldt, Robert and John Sprague (1987). "Networks in Context: The Social Flow of Political Information". In: *The American Political Science Review* 81.4, pp. 1197–1216. URL: http://www.jstor.org/stable/1962585.

Ihaka, Ross and Robert Gentleman (1996). "R: A Language for Data Analysis and Graphics". In: *Journal of Computational and Graphical Statistics* 5.3, pp. 299–314. DOI: 10.2307/1390807.

Jackman, Simon (2004). "Bayesian Analysis for Political Research". In: *Annual Review of Political Science* 7, pp. 483–505.

Jennings, M. Kent (2007). "Political Socialization". In: *The Oxford Handbook of Political Behaviour*. Ed. by Russell J. Dalton and Hans-Dieter Klingemann. Oxford: Oxford University Press, pp. 29–45. DOI: 10.1093/oxfordhb/9780199270125.003.0002.

Jöreskog, Karl G. (1990). "New Developments in LISREL: Analysis of Ordinal Variables Using Polychoric Correlations and Weighted Least Squares". In: *Quality and Quantity* 24.4, pp. 387–404. DOI: 10.1007/BF00152012.

— (1994). "On the Estimation of Polychoric Correlations and Their Asymptotic Covariance Matrix". In: *Psychometrika* 59.3, pp. 381–389.

Karvonen, Lauri and Jostein Ryssevik (2001). "How Bright was the future? The Study of Parties, Cleavages and Voters in the Age of the Technological Revolution". In: *Party Systems and Voter Alignments Revisited*. Ed. by Lauri Karvonen and Stein Kuhnle. London: Routledge, pp. 45–61.

King, Gary (1997). *A Solution to the Ecological Inference Problem. Reconstructing Individual Behavior from Aggregate Data*. Princeton: Princeton University Press.

Knoke, David and Song Yang (2008). *Social Network Analysis*. 2nd ed. Thousand Oaks: Sage.

Knuth, Donald Ervin (1984). "Literate Programming". In: *The Computer Journal* 27.2, pp. 97–111.

Lakhani, Karim R. and Eric von Hippel (2003). "How Open Source Software Works: "'"Free"' User-to-User Assistance". In: *Research Policy* 32.6, pp. 923–943. DOI: 10.1016/S0048-7333(02)00095-1.

Lazarsfeld, Paul F., Bernard Berelson, and Hazel Gaudet (1944). *The People's Choice. How the Voter Makes up His Mind in a Presidential Campaign*. Chicago: Columbia University Press.

Lazarsfeld, Paul F. and Herbert Menzel (1961). “On the Relation Between Individual and Collective Properties”. In: *Complex Organizations. A Sociological Reader*. Ed. by Amitai Etzioni. New York: Holt, Rinehart & Winston, pp. 422–440.

Loeliger, Jon and Matthew McCullough (2012). *Version Control with Git*. 2nd ed. Sebastopol: O’Reilly.

Long, J. Scott (2009). *The Workflow of Data Analysis. Principles and Practice*. College Station: Stata Press.

Lutz, Mark (2013). *Learning Python*. 5th ed. Sebastopol: O’Reilly.

McClurg, Scott D. (2006). “The Electoral Relevance of Political Talk: Examining Disagreement and Expertise Effects in Social Networks on Political Participation”. In: *American Journal of Political Science* 50.3, pp. 737–754. URL: http://www.jstor.org/stable/3694246.

Mecklenburg, Robert (2005). *Managing Projects with GNU Make*. Sebastopol: O’Reilly.

Miller, Warren E. and J. Merrill Shanks (1996). *The New American Voter*. Cambridge, London: Harvard University Press.

Monogan, James E. III (2015). “Research Preregistration in Political Science. The Case, Counterarguments, and a Response to Critiques”. In: *PS: Political Science & Politics* 48 (3), pp. 425–429. DOI: 10.1017/S1049096515000189.

Munzert, Simon et al. (2015). *Automated Data Collection with R. A Practical Guide to Web Scraping and Text Mining*. Chichester: Wiley.

Muthén, Bengt O. (1979). “A Structural Probit Model with Latent Variables”. In: *Journal of the American Statistical Association* 74, pp. 807–811.

— (2002). “Beyond SEM. General Latent Variable Modeling”. In: *Behaviormetrika* 29, pp. 81–117. URL: /home/kai/Work/Texte/Muthen2002.pdf.

Olson, Mancur (1965). *The Logic of Collective Action. Public Goods and the Theory of Groups*. Cambridge: Harvard University Press.

Raymond, Eric S. (1999). *The Cathedral and the Bazaar. Musings on Linux and Open Source by an Accidental Revolutionary*. Beijing et al.: O’Reilly.

Siegfried, André (1913). *Tableau politique de la France de l’Ouest sous la Troisieme Republique*. Paris: A. Colin.

Skrondal, Anders and Sophia Rabe-Hesketh (2004). *Generalized Latent Variable Modeling*. Boca Raton u.a.: Chapman & Hall.

Steenbergen, Marco R. and Bradford S. Jones (2002). “Modelling Multilevel Data Structures”. In: *American Journal of Political Science* 46, pp. 218–237.

Stegmueller, Daniel (2013). “How Many Countries for Multilevel Modeling? A Comparison of Frequentist and Bayesian Approaches”. In: *American Journal of Political Science*. DOI: 10.1111/ajps.12001.

Ward, Michael D. and Kristian Skrede Gleditsch (2008). *Spatial Regression Models*. Quantitative Applications in the Social Sciences 155. Thousand Oaks: Sage.

Ward, Michael D., Katherine Stovel, and Audrey Sacks (2011). "Network Analysis and Political Science". In: *Annual Review of Political Science* 14.1, pp. 245–264. DOI: 10.1146/annurev.polisci.12.040907.115949.

Western, Bruce and Simon Jackman (1994). "Bayesian Inference for Comparative Research". In: *American Political Science Review* 88, pp. 412–423.