**Inför REWHARD:s Syntaxbibliotek**

Den nationella infrastrukturen *”RElations, Work and Health across the life-course—A Research Data infrastructure*” (REWHARD) syftar till att säkerställa den långsiktiga överlevnaden, utvecklingen och användningen av ett antal komplementära, högkvalitativa longitudinella databaser. Inom REWHARD kommer delning av syntax att ske, såväl till användare inom REWHARD som till externa användare. Inför detta har en sammanställning av publikationer gjorts i *syfte* att kartlägga problem som kan uppstå när man delar med sig av syntax och olika lösningar som använts för att hantera sådana problem, samt av hur effektiva dessa lösningar är.

**Metod**

Underlaget i rapporten har insamlats på följande sätt:

* Sökord har diskuterats i REWHARD samt med statistikforskare vid Stockholms Universitet. Följande sökord har använts: ”data sharing”, ”syntax sharing”, ”statistical analysis”, ”research code”, ”code sharing”, ”data management”, ”repository”, “methods for sharing code”, “sharing code”, “administrative data”.
* Litteratursökningar har gjorts i Google Scholar, Web of Science, Psychinfo, och i PubMed enligt tabell 1. Artiklar har valts ut efter en genomläsning av abstract.
* Publikationer i referenslistor i de publikationer som identifierades i litteratursökningarna valdes ut efter genomläsning av abstract.

Dessutom har docent Magnus Stenbeck kontaktats, han har bl.a. varit projektansvarig för *the* *Swedish Initiative for Research on Microdata in the Social and Medical Sciences* (SIMSAM), vars syfte var att främja användandet av registerdata i vetenskapliga studier.

Nedan används termen ”kod” i vid bemärkelse och inkluderar sådant man refererar till med närliggande termer som t.ex. ”syntax”, ”script” eller ”program”. Detta för att undvika förvirring men även för att den hittills identifierade litteraturen inom området uteslutande använder engelskans ”code”.

**Resultat**

Sammanlagt identifierades 155 publikationer, varav 16 bedömdes relevanta för projektets syfte. För att en publikation ska anses relevant ska delning av kod omnämnas i publikationens abstract.

**VCSeller hemsida**

Många av de identifierade studierna[[1]](#footnote-1) rekommenderar något som kallas *version control system* (VCS) när man ska dela med sig av kod. Ett VCS är en virtuell lagringsplats för filer. Efter att en kodfil lagts in i ett VCS kan de ändringar som görs i koden, och vem som gjort vilka ändringar, enkelt spåras. Varje version av koden sparas och det är enkelt att gå fram- och tillbaka mellan olika versioner av samma fil. Dessutom kan flera redaktörer arbeta på samma kod samtidigt. Ett ofta använt alternativ till att använda sig av ett VCS är att publicera sin kod på en hemsida, här är ett exempel taget från Playford et al. (2016):

<https://ccpr.ucla.edu/donald-treiman-quantitative-data-analysis-stata-files-and-data-sets/>

Det finns dock flera problem med att lägga upp sin kod på hemsidor av det här slaget[[2]](#footnote-2). Möjligheten att se kodfilernas historik finns inte och man kan därför inte på ett enkelt sätt återvända till en tidigare version. Detta är särskilt problematiskt om man har flera redaktörer och koden plötsligt slutar fungera t.ex. på grund av att en redaktör har ändrar namnet på en variabel. Vidare är den här typen av hemsidor ofta projektspecifika och koden uppdateras därför inte längre när projektet avslutats, i många fall försvinner till och med hemsidorna helt. Dessutom lyckas moderna sökmotorer sällan hitta koden.

Ett VCS undviker ovan nämnda problem. Det går tydligt att se filhistorik, det är enkelt att återvända till en tidigare version av filen och i de fall flera redaktörer är involverade går det att se vem som har gjort vad. Koden kan också överleva det projekt som den skapats i. Ett bra exempel på detta är den kod som publicerats inom ramen för SIMSAM-INFRA. Projektet avslutades 2016 men koden är fortfarande tillgänglig, dessutom hittar man den direkt ifall man söker på ”registerresearch code”. Dessa skillnader mellan hemsidor och VCS är sammanfattade i tabell 2.

Det finns flera olika VCSsom fungerar på ungeför samma sätt, däribland GitHub, Bitbucket, Mercurial och Subversion. En viktig fråga är hur redaktörskapet ska se ut. Muşlu et al (2014) är en empirisk studie av övergången mellan två olika sorters redaktörskap hos ett stort företag. Företaget gick från ett centraliserat VCS till ett decentraliserat VCS. Ett centraliserat VCS innebär att kodens historik sparas på en central server. De anställda ville ha möjligheten att arbeta offline, de ville även kunna arbeta i mindre steg och mer utforskande. Detta var inte möjligt med deras centraliserade VCS utan de anställda var tvungna att vara inloggade för att arbeta med koden. När företaget övergick till ett decentraliserat VCS, som innebar att kodens historik sparades lokalt på varje anställds dator blev det enklare att arbeta mer utforskande och att arbeta offline. Sammantaget var de anställda nöjda med övergången men några anställda uppgav att decentraliserade VCS är svårare att lära sig. Mer tekniska resonemang kring den här typen av decentraliserat redaktörskap går att finna i Bird et al. (2009). För en kort men tydlig introduktion till kärnan i VCS rekommenderas Hardie (2014).

**VCS och SIMSAM**

Magnus Stenbeck har inom ramen för SIMSAM gjort ett försök att stimulera forskare att dela kod med hjälp av ett VCS som kallas GitHub[[3]](#footnote-3). Tyvärr har denna satsning inte gått så bra. Stenbeck menar att förklaringen kan vara bristande resurser i form av arbetstid och tekniskt kunnande, men också på att många forskare är tveksamt inställda till att dela med sig av kod.

Det finns empiriska studier av forskares attityder till att dela med sig av kod. En studie av Van den Eynden et al. (2016) bygger på en enkät som gått ut till 583 forskare i syfte att ta reda på mer om detta. Forskarna som deltog (134, alltså en svarsfrekvens på 23 %) uppgav en rad olika skäl till varför de inte vill dela med sig av sin kod. Nästan samtliga (82%) uppger att de avstår från att dela med sig eftersom det är för tidskrävande att få koden i presentabelt skick. Enligt Stodden (2010) är många dessutom bekymrade över att det kommer att kräva mycket tid att svara på frågor från dem som vill använda koden (54%), och många är oroliga för att inte bli citerade på ett korrekt sätt (47%). Andra svar som förekom var t.ex. att konkurrenter kan dra fördel av koden, eller att det endast är möjligt att ladda upp ett begränsat antal filer till vissa hemsidor.

**Andra lösningar**

Det finns några ytterligare alternativ om man inte vill använda ett VCS. Gemensamt för dessa är att de tydligt kopplar kod till en specifik text, och att de ställer höga krav på färdigheter i programmering. Kitchin (2015) beskriver noggrant två metoder för att koppla ihop en artikel med dess data och kod. Den första metoden går ut på att bifoga en s.k. "supporting information"-fil. Denna fil kan sedan öppnas i t.ex. Python. Den andra metoden går ut på att data och kod integreras i tillhörande artikel (PDF-fil). Textfilen måste då vara i s.k. "org-mode" och öppnas av en mjukvara som kallas EMAC. Johnson et al. (2018) föreslår något som kallas "executable documents" som är en del av ett särskilt VCS som kallas ”MIMIC code repository”. Till sist, Stodden (2015) beskriver ett verktyg, <http://researchcompendia.science/>, som kopplar samman artiklar med filer som innehåller all kod som använts för att genomföra studien ifråga.

**Referenser**

Bird, C., Rigby, P. C., Barr, E. T., Hamilton, D. J., German, D. M. and Devanbu, P. (2009) ‘The promises and perils of mining git’, in *Mining Software Repositories, 2009. MSR’09. 6th IEEE International Working Conference on*. IEEE, pp. 1–10.

Crook, S. M., Davison, A. P. and Plesser, H. E. (2013) ‘Learning from the Past: Approaches for Reproducibility in Computational Neuroscience’, in Bower, J. M. (ed.) *20 Years of Computational Neuroscience*. New York, NY: Springer New York (Springer Series in Computational Neuroscience), pp. 73–102. doi: [10.1007/978-1-4614-1424-7\_4](https://doi.org/10.1007/978-1-4614-1424-7_4).

Gentzkow, M. and Shapiro, J. M. (2014) ‘Code and data for the social sciences: A practitioner’s guide’, *Chicago, IL: University of Chicago*.

Hardie, A. (2014) ‘version control software | ESRC Centre for Corpus Approaches to Social Science (CASS)’. Available at: <http://cass.lancs.ac.uk/tag/version-control-software/> (Accessed: 31 January 2019).

Johnson, A. E., Stone, D. J., Celi, L. A. and Pollard, T. J. (2018) ‘The MIMIC Code Repository: enabling reproducibility in critical care research’, *Journal of the American Medical Informatics Association*, 25(1), pp. 32–39. doi: [10.1093/jamia/ocx084](https://doi.org/10.1093/jamia/ocx084).

Kiddle, S. J., Voyle, N. and Dobson, R. J. (2018) ‘A Blood Test for Alzheimer’s Disease: Progress, Challenges, and Recommendations’, *Journal of Alzheimer’s Disease*, (Preprint), pp. 1–9.

Kitchin, J. R. (2015) *Examples of effective data sharing in scientific publishing*. ACS Publications.

Muşlu, K., Bird, C., Nagappan, N. and Czerwonka, J. (2014) ‘Transition from centralized to decentralized version control systems: A case study on reasons, barriers, and outcomes’, in *Proceedings of the 36th international conference on software engineering*. ACM, pp. 334–344.

Playford, C. J., Gayle, V., Connelly, R. and Gray, A. J. G. (2016) ‘Administrative social science data: The challenge of reproducible research’, *Big Data & Society*, 3(2), p. 2053951716684143.

Shamir, L., Wallin, J. F., Allen, A., Berriman, B., Teuben, P., Nemiroff, R. J., Mink, J., Hanisch, R. J. and DuPrie, K. (2013) ‘Practices in source code sharing in astrophysics’, *Astronomy and Computing*, 1, pp. 54–58. doi: [10.1016/j.ascom.2013.04.001](https://doi.org/10.1016/j.ascom.2013.04.001).

Stodden, V. (2010) ‘The Scientific Method in Practice: Reproducibility in the Computational Sciences’, *SSRN Electronic Journal*. doi: [10.2139/ssrn.1550193](https://doi.org/10.2139/ssrn.1550193).

Stodden, V. (2015) ‘Reproducing statistical results’, *Annual Review of Statistics and Its Application*, 2, pp. 1–19.

Stodden, V. and Miguez, S. (2014) ‘Best Practices for Computational Science: Software Infrastructure and Environments for Reproducible and Extensible Research’, *Journal of Open Research Software*, 2(1), p. e21. doi: [10.5334/jors.ay](https://doi.org/10.5334/jors.ay).

Van den Eynden, V., Knight, G., Vlad, A., Radler, B., Tenopir, C., Leon, D., Manista, F., Whitworth, J. and Corti, L. (2016) ‘Survey of Wellcome researchers and their attitudes to open research’. doi: [10.6084/m9.figshare.4055448.v1](https://doi.org/10.6084/m9.figshare.4055448.v1).

Vandewalle, P. (2012) ‘Code Sharing Is Associated with Research Impact in Image Processing’, *Computing in Science & Engineering*, 14(4), pp. 42–47. doi: [10.1109/MCSE.2012.63](https://doi.org/10.1109/MCSE.2012.63).

Wilson, G., Aruliah, D. A., Brown, C. T., Chue Hong, N. P., Davis, M., Guy, R. T., Haddock, S. H. D., Huff, K. D., Mitchell, I. M., Plumbley, M. D., Waugh, B., White, E. P. and Wilson, P. (2014) ‘Best Practices for Scientific Computing’, *PLOS Biology*, 12(1), p. e1001745. doi: [10.1371/journal.pbio.1001745](https://doi.org/10.1371/journal.pbio.1001745).

Tabell : Sammanställning av litteratursökningar i databaser. Datum för sökning, sökord och antal träffar redovisas tillsammans med utvalda publikationer för varje sökning. I tabellen redovisas ej publikationer som valts ut efter genomläsning av referenslistor till redan utvalda publikationer.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Datum** | **Databas** | **Sökning** | **Antal träffar** | **Utvalda publikationer** |
| 2019-08-22 | *Google Scholar* | *"data sharing" AND "syntax sharing" AND "statistical analysis"* | *4* | *Playford et al. (2016)* |
| *"research code" AND "code sharing"* | *38* | *Kiddle et al. (2018) Van den Eynden et al. (2016) Stodden (2015)* |
| *"code sharing" AND "data management" AND “code repository”* | *46* | *Stodden and Miguez (2014) Kitchin (2015)* |
| *"methods for sharing code"* | *6* | *Crook et al. (2013)* |
| *"sharing code" AND "administrative data"* | *61* | *Shamir et al. (2013)* |
| 2019-09-05 | *PsychInfo* | *“syntax sharing” OR “code sharing” OR “sharing code” OR “research code”* | *22* |  |
| *“share syntax” OR “share code”* | *2* |  |
| *PubMed* | *“syntax sharing” OR “code sharing” OR “sharing code” OR “research code”* | *818* |  |
| *“share syntax” OR “share code”* | *1165* |  |
| *Web of Science* | *“syntax sharing” OR “code sharing” OR “sharing code” OR “research code”* | *375* |  |
| *“share syntax” OR “share code”* | *18* |  |

|  |  |  |
| --- | --- | --- |
|  | ***Hemsida*** | ***VCS*** |
| ***Sökmotorer*** | *Svårt att hitta kod genom sökmotorer som t.ex. google.* | *Enkelt att hitta koden, se t.ex.*  ”registerresearch code”. |
| ***Livslängd*** | *Begränsad livslängd, oftast inte längre än forskningsprojekt.* | *Livslängd oberoende av forskningsprojekt.* |
| ***Redaktörsskap*** | *Svårt att hantera flera redaktörer.* | *Möjligheten att enkelt se olika versioner gör det enkelt att redigera från flera olika håll samtidigt.* |
| ***Filhistorik*** | *Filversioner måste sparas och laddas upp manuellt.* | *Alla versioner av en fil sparas automatiskt och kan plockas fram vid behov.* |

Tabell 2: Jämförelse av metoder för att dela kod. De metoder som jämförs är hemsidor och VCS och de jämförs med avseende på sökmotorer, livslängd, redaktörsskap och filhistorik.

1. Playford et al. (2016), Shamir et al. (2013), Crook et al. (2013), Wilson et al. (2014), Kiddle et al. (2018), Stodden och Miguez (2014), Vandewalle (2012), Gentzkow och Shapiro (2014). [↑](#footnote-ref-1)
2. Se Playford et al. (2016) för en mer ingående diskussion. Även Gentzkow och Shapiro (2014) har många tydliga exempel. [↑](#footnote-ref-2)
3. Privat korrespondens. [↑](#footnote-ref-3)