PROV in StatJR

Danius T. Michaelides and Luc Moreau

Electronics and Computer Science, University of Southampton, UK
{dtm,L.Moreau}@ecs.soton.ac.uk

1 StatJR

StatJR [1] is a suite of tools designed to aid in the use and teaching of statistical analysis techniques with an emphasis on their use in social science. Our tools are designed facilitate reproducible research both in terms of providing provenance of outputs for use in publications and also in terms of recording the activities of users. Clearly for publishing results, PROV is key but we also use PROV to support interoperability between tools.

The tools are: 1) A web frontend to run statistical operations. Here we use provenance as a record of the operations that the user has performed. 2) An e-book interface that allows embedding of statistical processes and outputs into a document. Here we record the provenance of executions to facilitate the operation of the e-book player software. PROV-O and an RDF store are used here with SPARQL queries used to extract information from the RDF graphs. 3) A workflow execution engine that allows users to build more complex analyses. The workflow execution engine records provenance in a form that allows generating of workflows for reproducibility. We use PROV-Template to generate PROV.

2 Issues with PROV

The execution model in StatJR is that of a process using named inputs and generating a number of named outputs. We chose to model this by using an attribute on `used` and `wasGeneratedBy` relations. An alternative representation would be to model an input set and an output set as Collections. However we donnt take this approach because PROV does not allow attributes on `hadMember` and, even if it did, we'd still have to use our own attribute to for the name value. PROV-Dictionary does have the right relation with a common understanding of the name/key value, but it is only a W3C note and support is patchy in libraries. Note that we'd only require the `hadDictionaryMember` relation from PROV-Dictionary and not the insertion and removal operations.

[1] StatJR was funded by the ESRC under the projects RES-149-25-1084 and ES/K007246/1.