# Data management procedures for reproducible research pipelines

Ivan C. Hanigan

* *Corresponding author: ivan.hanigan@anu.edu.au*

## Contents



Unpublished working paper\ Draft: January 25, 2016

**Abstract:** This unpublished working paper was written to accompany the material included in the PhD thesis: 'Using Reproducible Research Pipelines to Help Disentangle Health Effects of Environmental Changes from Social Factors' by Ivan Hanigan (2016). It sets out the key data management and analysis principles that were found to be most effective for the reproducibile synthesis and integration of heterogeneous datasets for analysis and reporting.

The version submitted with the thesis is available as part of a Github repository at:

# 1 Introduction

There is a need for developing an evidence based set of best practice guidelines for data management in implementing an eco-social epidemiologic research programme Peng (2015). This can be conceptualised as implementing reproducible research pipelines in epidemiology.

## 1.1 Configuration versus convention: the case for standardised approaches

Reproducibility is the ability to recompute the results of a data analysis with the original data. It is possible to have analyses that are reproducible with varying degrees of difficulty. A data analysis might

be reproducible but require thousands of hours of work. A primary challenge to reproducible data analysis is to make analyses that are *easy* to reproduce.

To achieve this, a guiding principle is that analysts should effectivly implement 'pipelines' of method steps and tools. Data analysts should employ standardised and evidence-based methods based on conventions developed from many data analysts approaching the problems in a similar way, rather than each analyst configuring a pipeline to suit a particular individual or domain-specific preferences.

Noble (2009) points out that 'the principles behind organizing and documenting computational experiments are often learned on the fly, and this learning is strongly influenced by personal predilections'. Leek & Peng (2015) describe this as data analysis being 'taught through an apprenticeship model, and different disciplines develop their own analysis subcultures'. By codifying what an appropriate pipeline would contain, data analysis will be more robust. According to Peng (2015), there should not be a 'lonely data analyst' coming up with their own method. If a researcher conducted an analysis using a reproducible pipeline 'you could at least have a sense that something reasonable was done' and be confident that you could easily check what had been done if you needed to.

### 1.1.1 The core components of a pipeline

As mentioned in chapter 1, Peng et al. (2006) distilled a core set of components for reproducibility from earlier work including that of Schwab et al. (2000). These are:

- Hypothesis and design
- Data (measurement, pre-processing, analytic)
- Analysis Methods
- Documentation (of all steps)
- Distribution (of the paper, data and code).

In essence this requires attention to be turned to the issue of how the data and analytical steps amassed – toward a reality where this is archived and there is a good understanding all round as to how the study were set up and conducted. Different assumptions or different treatment of the data could conceivably lead to different inferences and conclusions being drawn, such as in the example where 29 research teams were given the same dataset but reached a wide variety of conclusions using different methods

on the same data set to answer the same question (about football players' skin colour and red cards) (Silberzahn & Uhlmann 2015).

This is partly because of an underlying complexity in the information drawn from complex systems involving multi-causality, and partly because of different assumptions and different backgrounds and viewpoints. A finding that a variable does or does not cause a disease, might be drawn honestly from the same set of data.

The techniques of pipelines described here are targeting the integrity of the process of data selection, the robustness and suitableness of the methods used, a commonsense and well-argued selection of health outcomes and environmental or social exposures, and the clarity and transparency of the assumptions made.

### 1.1.2 Hypothesis and design

The first stage of the pipeline is hypothesis generation and study design. In this stage documentation should explain the literature base supporting the study, the decisions made in selection of explanatory factors for inclusion, decisions made such as the experimental unit, observational unit, measurement method, as well as spatial or temporal extent. This information will also be needed for ethical review and approval.

### 1.1.3 Data

The data that were measured should be well managed, however the requirements for accessing the original raw data are less important than for the analytical dataset. Descriptions of how the measured data were transformed into the analytic data should be available. Public data repositories or institutional services such as university libraries should be used to ensure longevity of the data storage.

### 1.1.4 Methods

The software code underlying the principal results needs to be made available. In addition, the computer environment necessary to execute that code should be described adequately to 'deploy' a new computer

set-up that can reproduce the computations needed.

### 1.1.5 Documentation

Adequate documentation of the code and data should be available to enable others to repeat the analyses and to conduct other similar ones. This can take the form of metadata, reports, journal papers or even books (Peng & Dominici 2008). Indeed textbooks on statistical methods can benefit greatly from being accompanied by data and analytical code to enhance their pedagogic functions (Barnett & Dobson 2010; Barnett *et al.* 2014).

An important underpinning to reproducible research is the reproducible report. This is the ultimate form of documentation because the information that represents the outputs of the research is written alongside the code that performs the computations that are being described. There has been many recent advances made in terms of tools for reproducible reports such as R markdown and knitr (Xie 2014).

Metadata should be created and maintained as a priority task at all stages of the data analysis process. An international standard should be preferred over selectively choosing what information one collects and what fieldnames one uses to describe each item of documentation. Ecological Metadata Language (EML) and the Data Documentation Inititative (DDI) are two such standards that offer useful semantic constructs for describing epidemiological data.

### 1.1.6 Distribution

Distribution or dissemination of the material needs to use a standard method if they are to be used by others. It is not enough just to provide access to the software and data, but also adequate documentation is required to explain and potentially assist downstream users to piece these together.

## 1.2 Procedures when conducting a reproducible research analysis

Having defined above the principle components for a pipeline there are procedural questions about how to go about compiling those. The key steps include:

- Data Management Plans and Data Inventories
- Tracking method steps
- Developing code
- Maintaining data storage
- Writing reports
- Distributing the materials.

### 1.2.1 Data management plan and data inventory

TBA, there is better dev version at /home/ivan_hanigan/projects/swish-dmp/report_swish_data_management_p and datinv

### 1.2.2 Case study 2: Visualisation of methods steps using bespoke software

The method step is the key atomic unit of a scientific pipeline. It consists of inputs, outputs and a rationale for why the step is taken.

A simple way to keep track of the steps, inputs and outputs is shown in Table 1.

Table 1: A simple table to track method steps, data inputs and outputs

| STEP | INPUTS | OUTPUTS |
|---|---|---|
| Step1 | Input 1, Input 2 | Output 1 |
| Step2 | Input 3 | Output 2 |
| Step3 | Output 1, Output 2 | Output 3 |

The steps and data listed in Table 1 can be visualised. To achieve this an R function was written as part of this PhD project and is distributed in the author's own R package available on Github (https://github.com/ivanhanigan/disentangle). This is the `newnode` function. The function returns a string of text written in the `dot` language which can be rendered in R using the `DiagrammeR` package, or the standalone `graphviz` package. This creates the graph view shown in Figure 1. Note that a new field was added for Descriptions as these are highly recommended.

```
library(disentangle); library(stringr); library(readxl)

steps <- read_excel("steps_basic_workflow.xlsx")

nodes <- newnode(indat = steps, names_col = "STEP",

                 in_col = "INPUTS",out_col = "OUTPUTS")

DiagrammeR::grViz(nodes)
```

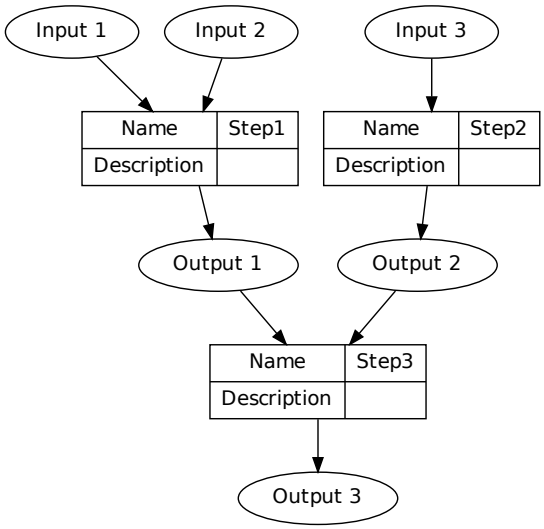


Figure 1: A graphical view of the steps that comprise a simple data analysis pipeline

### 1.2.3 Data storage and access

Some datasets such as sensitive personal information about suicide or climate change scenarios with restrictions due to privacy and confidentiality rules, or because of protected intellectual property, need to be accessed in a restricted way. This complicates the implementation of the method of pipelines which dictates that all the steps, models and assumptions need to be made transparent and available for scientific debate even though the datasets may require authorisation to access. Restrictions around access to data have increased recently in Australia. As an example the custodians of the national mortality database made it virtually impossible to access these data for several years after the discovery of an incident in which Australian population health researcher Dr Stephen Begg was reported to have hacked into the database in an illegal act (O'Keefe 2007). The subsequent investigation by the data custodians led to a wide ranging modification to the procedures for approval and provision of these data that make the access much more restricted. Appropriate access to data is therefore required to address this issue. In the work reported in the conference presentation in this thesis, a range of available workflow tools for data management and analysis were investigated and developed.

### 1.2.4 Reports

Reproducible research reports are written using a scripting language for statistical computing and graphics. The report is made up of ordinary text written in a suitable format that enables the computational process to recognise it as text. An example is the Rmarkdown format which is very similar to text used when authoring word processor documents (http://rmarkdown.rstudio.com). There are also chunks of pure statistical programming code (such as R codes) that perform data manipulations and analyses when the document is 'evaluated'. When the processing stage is run a report document is generated that includes both content as well as the output of any embedded computer code 'chunks' within the document. An example of this is provided in the Supporting Information document of Paper 1 of this thesis.

## 1.3 Planning and implementing a pipeline

It can be much easier to conceptualise a complicated data analysis method than to implement this as a reproducible research pipeline. The most effective way to implement a pipeline is by methodically tracking each of the steps taken, the data inputs needed and all the outputs of the step. If done in a disciplined way then the analyst or some other person could 'audit' the procedure easily and access the details of the pipeline they need to scrutinise.

### 1.3.1 A standardised data analysis pipeline framework

One method that was selected for use in the papers of this thesis was the concept of the Load-Clean-Functions-Do (LCFD) framework. This was first proposed by Josh Reich on the open-source software discussion forum called 'stack overflow' (http://stackoverflow.com/a/1434424), and then encoded into the 'makeProject' R package (http://cran.r-project.org/web/packages/makeProject/makeProject.pdf). The approach is demonstrated in case study 3 below.

### 1.3.2 Case study 3: Simple pipeline using the makeProject package

```
# in an interactive R session at the command line choose your project directory
setwd("~/projects")
# load the required functions from the makeProject package
library(makeProject)
# use the makeProject function to
makeProject("my_first_pipelines_project")

### gives
/my_first_pipelines_project/
    /code/*.R
    /data/
    /DESCRIPTION
    /main.R

# in main.R you put these lines into the script and run them as the steps of the pipeline evo
source("code/load.R")
source("code/clean.R")
source("code/func.R")
source("code/do.R")

# Reporting is then a matter of choice
## If using the rmarkdown approach there would be an Rmd file that contained the prose
## and turned into a PDF, HTML or Word document with a line such as
rmarkdown::render("My-Pipeline-Report.Rmd", "pdf_document")
```

### 1.3.3 File organization and naming

In many stages of a pipeline, an analyst will want to include details of the settings or what dataset they started out with. Rather than saving a folder or file name that is long and uninformative there are many different ways to organizing folders and files.

Key techniques for this are available and known in the data analysis community as 'Tidy Data' guidelines. In the words of Wickham (2014) the order that data should be arranged in follows some generic principles:

> 'A good ordering makes it easier to scan the raw values. One way of organizing variables is by their role in the analysis: are values fixed by the design of the data collection, or are they measured during the course of the experiment? Fixed variables describe the experimental design and are known in advance. Computer scientists often call fixed variables dimensions, and statisticians usually denote them with subscripts on random variables. Measured variables are what we actually measure in the study. Fixed variables should come first, followed by measured variables, each ordered so that related variables are contiguous. Rows can then be ordered by the first variable, breaking ties with the second and subsequent (fixed) variables.'

### 1.3.4 An exemplar

The following protocol was developed for an ecology and biodiversity database that the author of this PhD thesis was involved with. The naming convention relied heavily on a sequence of information being used to order the names of folders, subfolders and files. This is:

1. The project name (and optional sub-project name)
2. Data type (such as experimental unit, observational unit, and/or measurement methods)
3. Geographic location (State, Country)
4. Temporal frequency and coverage (such as annual or seasonal tranches).

### 1.3.5 The concepts of slow moving dimensions and fast moving variables

The concept of dimensions and variables can be useful here, and especially for deciding on filenames. Dimensions are fixed or change slowly while variables change more quickly. By 'change', this means that there are more of them. For example the project name is 'fixed', that is it does not change across the files, but the sub-project name does change, just more slowly (say there may be 2-3 different sub-projects within a project). Then there may be a set of data types, and these 'change' more quickly than the sub-project name. Then the geographic and temporal variables might change quickest of all.

So a general rule for the order of things can be stated. The fixed and slowly changing variables should come first (those things that don't change, or don't change much), followed by the more fluid variables (or things that change more across the project). List elements can then be ordered so that the groups of things that are similar will always be contiguous, and vary sequentially within clusters.

An example is shown in Table 2 to describe this and make it easier to understand. Here is a set of file names that were constructed for an ecological field sites project that sampled plot-based measurement locations. At the begining of the procedure a controlled vocabulary of data types and their acronyms was created.

Table 2: An example of standardised filename conventions to simplify tracking complicated datasets

| Filename | Title |
|---|---|
| asn_fnqr_soil_charact_robson_2011.csv | Soil Data, Far North Queensland Rainforest SuperSite, Robson Creek, 2011 |
| asn_fnqr_soil_pit_robson_2012.csv | Soil Pit Data, Water Content and Temperature, Far North Queensland Rainforest SuperSite, Robson Creek, 2012 |
| asn_fnqr_veg_seedling_robson_2010-2012.csv | Seedling Survey, Far North Queensland Rainforest SuperSite, Robson Creek, 2010-2012 |
| asn_fnqr_veg_seedling_transect_coord_robson_2010-2012.csv | Seedling Survey, Far North Queensland Rainforest SuperSite, Robson Creek, 2010-2012 |
| asn_fnqr_core_1ha_robson_2014.csv | Soil Pit Data, Soil Characterisation, Far North Queensland Rainforest SuperSite, Robson Creek, Core 1 ha plot, 2014 |
| asn_fnqr_fauna_biodiversity_ctbcc_2012.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, 2012 |
| asn_fnqr_fauna_biodiversity_ctbcc_2013.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, 2013 |
| asn_fnqr_fauna_biodiversity_ctbcc_capetrib_2014.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, Cape Tribulation, 2014 |
| asn_fnqr_fauna_biodiversity_ctbcc_lu11a_2014.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, LU11A, 2014 |
| asn_fnqr_fauna_biodiversity_ctbcc_lu7a_2014.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, LU7A, 2014 |
| asn_fnqr_fauna_biodiversity_ctbcc_lu7b_2014.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, LU7B, 2014 |
| asn_fnqr_fauna_biodiversity_ctbcc_lu9a_2014.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, LU9A, 2014 |
| asn_fnqr_fauna_biodiversity_ctbcc-lu11a_2009-2011.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, LU11A, 2009-2011 |
| asn_fnqr_fauna_biodiversity_ctbcc-lu7a_2009-2011.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, LU7A, 2009-2011 |
| asn_fnqr_fauna_biodiversity_ctbcc-lu9a_2009-2011.csv | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, LU9A, 2009-2011 |
| asn_fnqr_fauna_biodiversity_habitat codes_ctbcc-lu11a_2009-2011.pdf | Vertebrate Fauna Biodiversity Monitoring, Far North Queensland Rainforest SuperSite, CTBCC, LU11A, 2009-2011 |

## 1.4 Visualisation techniques

### 1.4.1 Make a list of steps, inputs and outputs

A very simple example of a pipeline is shown in Table 3. The steps and data listed in Table 3 can be visualised using the `newnode` function described above in case study 2. This creates the graph of this pipeline shown in Figure 2. As the analysis progresses through the phases of testing, refinement and final versions. The linked table and graphical depiction can be very helpful for reference by the analyst. The optional setting to define a status of each step (TODO, DONE, WONTDO) can be used to add colour, and show steps that remain to be done. The addition of short summary descriptions are also very useful for orienting oneself to the required tasks and their priorities. Such flow chart diagrams can be printed up on large sheets of paper and stuck on the wall beside a computer workstation for use in day-to-day work.

Table 3: A table with the steps of a simple data analysis pipeline

| STEP | INPUTS | OUTPUTS | DESCRIPTION | STATUS |
|---|---|---|---|---|
| Step1 | Source 1, Source 2 | Derived 1, QC check | This might be latitude and longitude of sites | DONE |
| Step2 | Source 3 | Derived 2 | This might be weather data | DONE |
| Step3 | Derived 1, Derived 2 | Derived 3 | Merging these data means they can be analysed | TODO |
| Step4 | Derived 3 | Model selection | | TODO |
| Step5 | Model selection | Sensitivity analysis | | TODO |

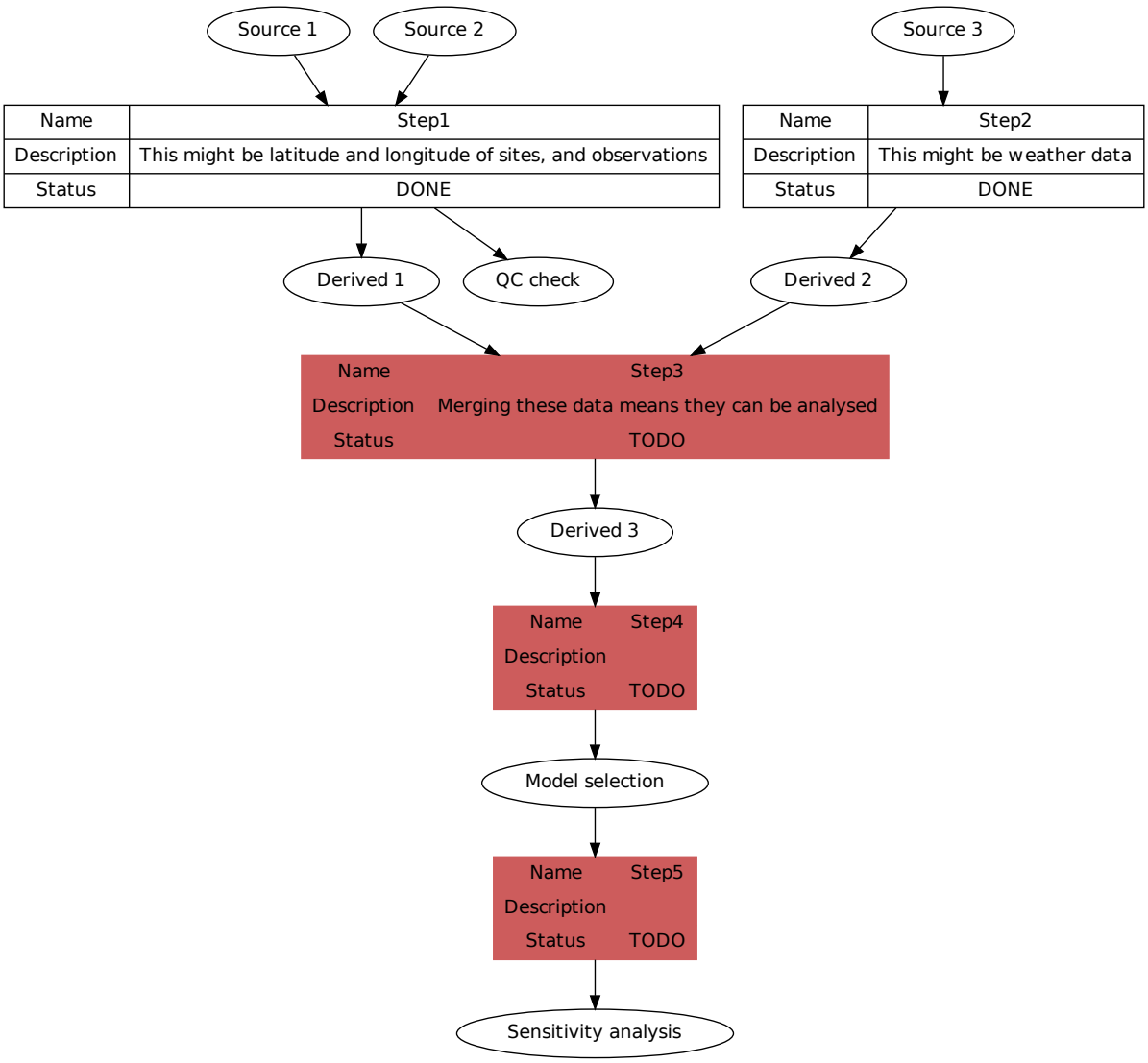


Figure 2: A visualisation of a data analysis pipeline showing the use of colour

As an example of the kinds of tangible steps such a workflow might entail a schematic diagram has been created and is shown in Figure 3.

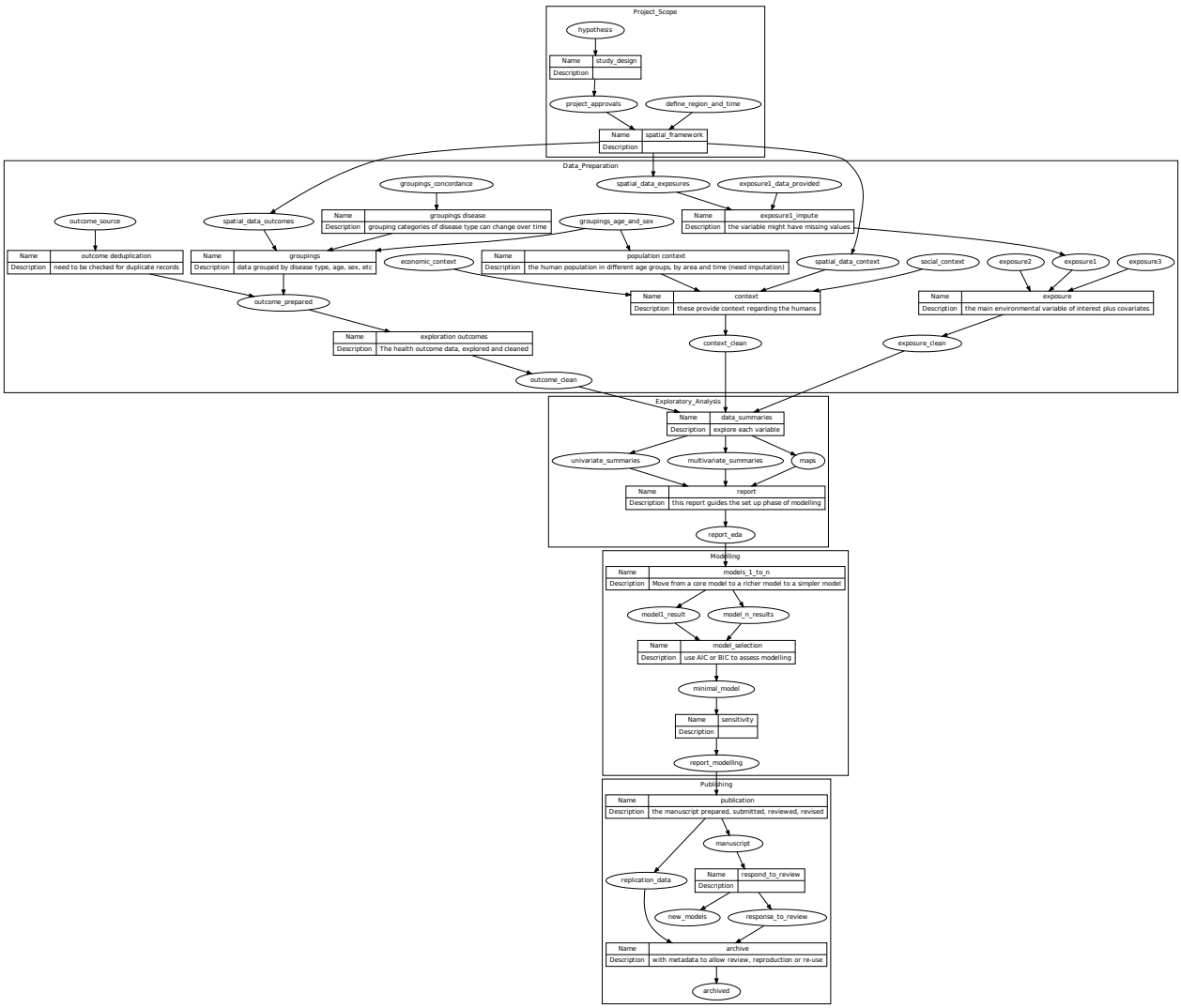


Figure 3: A schematic flow chart showing the steps required to prepare and conduct an analysis of health, environmental and social data.

# 2 Discussion and Conclusion

- TODO

# 3 References


Barnett, A.G. & Dobson, A.J. (2010). *Analysing Seasonal Health Data*. Springer, Berlin, Heidelberg, Germany.

Barnett, A.G., Baker, P. & Dobson, A.J. (2014). Package 'season': Analysing seasonal data R functions. R package version 0.3-5.

Leek, J.T. & Peng, R.D. (2015). Statistics: P values are just the tip of the iceberg. *Nature*, 520(7549), 612–612.

Noble, W.S. (2009). A quick guide to organizing computational biology projects. *PLoS Computational Biology*, 5(7), 1–5.

O'Keefe, B. (2007). Hackers pick up UQ cash prize. *The Australian*. http://www.theaustralian.com.au/higher-education/hackers-pick-up-uq-cash-prize/story-e6frgcjx-1111113191659 [Accessed 14 Oct. 2015]

Peng, R.D. (2015). *Report Writing for Data Science in R*. Leanpub. Unpublished Draft (Accessed 22 Dec. 2015). https://leanpub.com/reportwriting

Peng, R.D. & Dominici, F. (2008). *Statistical Methods for Environmental Epidemiology with R. A Case Study in Air Pollution and Health*. Springer Science & Business Media, New York, USA.

Peng, R.D., Dominici, F. & Zeger, S.L. (2006). Reproducible epidemiologic research. *American Journal of Epidemiology*, 163(9), 783–789.

Schwab, M., Karrenbach, M. & Claerbout, J. (2000). Making scientific computations reproducible. *Computing in Science and Engineering*, 2(6), 61–67.

Silberzahn, R. & Uhlmann, E.L. (2015). Crowdsourced research: Many hands make tight work. *Nature*, 526(7572), 189–191.

Wickham, H. (2014). Tidy data. *Journal of Statistical Software*, 59(10), 1–23.

Xie, Y. (2014). Chapter 1. Knitr: A comprehensive tool for reproducible research in R. *Implementing Reproducible Research* (eds V. Stodden, F. Leisch & R. Peng). CRC Press, Boca Raton, USA.