**26 November 2018**

**Introduction**

**Names;**

Webke – making energy consumption in household surveys

Jeremy - blood transfusions screening, packages,

Wim – fitting stochastic, IBM to data; MVimputation through change equation

Laurette – HIV incidence estimation, simulate an age/time structured population

Epi – genomic mapping for ancestor; modelling natural selection – integrated tools for doing this.

Gloria – prediction of weather patterns

Benard – commercial lending through microfinance, SACOS (?) – evaluation of performance

Sam – cameras to detect objects (i.e. classification)

- abstractive text summarization, machine learning and neural networks

Mohua; develop of data pipeline for processing of spatial data; may be for people

Faikah – understanding impact of HIV prevention methods. Stochastic IBM; synthesizing a RCT to measure prevention impact. – need to improve run-time

Perseverance – communication network for autonomous driving vehicle, for vehicles from different manufacturers

Josephs ; natural language processing; google translate for

Burri – ABC computations

**Requirements (the prerequisites to design):**

Did you think about requirements ?

How do you use them to drive your work ?

Setting constraints about you project

Scientific process

* Observations, record your observations
* Formulate questions
* Generate hypothesis
* Test hypothesis with experiments

Name some flavors of requirements

* Hardware
* Operating system / platforms / applications
* Functional requirements
* Non-functional requirement.
  + How to do / not do
* Cost / Budget / time
* Access /Security
* Resources (i.e. computational - memory, speed, disk space)
* Inputs / Outputs (including formats)

Unique requirements for science

* Must be used to test some hypothesis / answer some question
* Must be able to communicate to others what’s going on
* Must be repeatable
  + Comprehensible source, open source

Applying requirements

* Pseudo-code, process maps, other design, documentation approaches
* Test driven development
  + We should try to develop tests that run a series of inputs vs. a series of outputs
* Documentation
* Project comprehensibility

Simple programming problem

“Hello World” test suite for different languages

* For platform x, y, z ……
* Framework tells computer to compile + run

**Initial Challenge**

**Step 1:**

* Must have list of staff all staff members (csv format, last name first name, initial)
* Must have location where staff photos are kept
* Must specify possible file type and sizes of photos
* Must specify desired file type and size of photos
* Must specify desired output location (i.e. website) where photos will be kept
* Must specify timelines for staff’s submission of photos

**Step 2:**

* Requirements
  + 95% of all 500 staff must be represented on website
  + File format, size and naming convention must be the same as the one you’ve already standardized in step 1
  + All staff should send “current” photos. Meaning; don’t rely on existing photos.
  + Must be “easy” way or staff to submit/you to receive photos.
  + Must be done before Christmas
* Identify project scope:
  + In 1-3 sentences, what does your project *do* now?

Simulates and age time structured population of infected and susceptible, based on the methods of lines.

Function 1: calculates a survival probability of given age and time (as matrix); user specified incidence and mortality function.

Function 2: converts the cumulative survival probability (as matrix).

Function 3: calculates susceptible population from cumulative survival probability and birth and age distribution

Function 4: Calculates number of infected per given age and time.

* + Imagine that you have completed this workshop, including the hack-a-thon on your project; write the new version of those 1-3 sentences. Keep in mind: that description might not change at all! If it won’t change, write 1-3 sentences about what *will* change (e.g., performance).

- improved run time.

- package in R

* + Are there similar projects (that is, performing mostly the same function, but perhaps in a different language or for different inputs) you can learn from or build on? Do an internet search as needed.
* Desolve
* Leslie.matrix
* ReacTran

Trying to recreate some aspects of what these packages do already.

* + Who will interact with your software? Think broadly about the definition of “interact” - for example, who might see the results? Who else might change the code? Who might run the software?
* Change the code ? - SACEMA incidence working group
* See the results ?
* Run the software ? Academics/public health professionals interested in simulating age/time-structured populations interested in a validation of various incidence estimators and observational studies.
  + What constraints are there on the user interface?
* R
* Birth counts, incidence function, mortality function specification of initial conditions.
* Output is a matrix
* Run-time
* I/O requirements:
  + What data does your project require (inputs)?

Specify initial conditions (Birth counts, age distribution), parameters (incidence function, mortality function)

* Where/how will you obtain it?
  + User specified (based on literature review, or available program/survey)
  + What are the outputs from your project?
  + The framework for generating age/time structured populations in the format of a matrix (count) according to user specified initial conditions and parameters.
  + Calculate prevalence per age/time
  + What data/file formats will you need to work with?
    - Matrices and arrays
* Divide and conquer:
  + Identify 3-6 subproblems that your project can be divided into
* Function 1: calculates a survival probability of given age and time (as matrix); user specified incidence and mortality function.
* Function 2: converts the cumulative survival probability (as matrix).
* Function 3: calculates susceptible population from cumulative survival probability and birth and age distribution
* Function 4: Calculates number of infected per given age and time.
  + What is the most natural order in which to tackle the subproblems?
    - Easist first? Hardest? From the interface in, or following the flow of information?
* From trivial to complex
  + Function 1 - 4
    - Which can be implemented independently? Do some absolutely have to be completed first?
      * Yes; they have to be done in sequence.
  + How will you verify that specific subproblems have been solved?
* The expected outputs come from the given inputs (i.e. the functions work).
* Draw a flowchart of your research problem and its parts.
  + Identify which steps each of your detailed requirements apply to.
  + Are the steps that do not have detailed requirements associated with them?
* Technical requirements:
  + What are the external technologies (for example, random number library, SQL database for recording data) your project uses? Or could use? Do an internet search as needed.
  + What do you need to know/learn in order to complete the project?
  + What platform does
* If the project is substantially completed, what things do you think should have been done differently?
* Testing:
  + How could you test specific requirements have been satisfied? As in, what comparisons would you make (not what framework would you use)?
  + To adequately & practically test your project, do you need to automate tests?
* Scientific Requirements:
  + What are the specific science-related concerns for your project?

**Presentations, 27 Nov 2018.**

Epi;

* FRANC, an adaptable/portable python resource; LAI (existing application challenge to those with limited programming skills).
* Currently; one line linux commands, user provides parameter file – which includes
* Can be done locally or on the server

Gloria

* Spatially interpolated of Real-Date Climate data
* Prediction of droughts important; but current methods are outdated.

Deborah

* Getting risk scores for smallholding farmers
  + Amount/duration of load
  + Python and PHP

Wim

* Improve code documentation
* Modify dependency on MICE package
* …..
* Package ready for CRAN submission

Xx

* Python package for textual summaries by abstraction; git hub; textual training data – learning algorithm

**Discussion**

* Scientific values at stake in code management
  + Reproducibility
  + Building/Accumulation of knowledge
* Data management obligation do you have for funders.
  + Non-disclosure
  + Publicly available datasets
  + IP depending on your employer
* How does data management effect you, partners, stakeholders?

**Organizing project resources**

* Folder structure, naming
* Documentation – comprehensibility, clarity, reproducibility
  + Have a “read\_me” text file at the root directory
  + Minimize separate locations for resources
  + Directory hierarchy for major dimensions
  + Maintain up to date project map
  + Use file list (s), i.e. manifest
* Version control
  + What is the value
  + How are you using it now?
    - Mostly saving files on my local machine, changing filename\_date when substantial updates are made.
    - I’d like to become more comfortable working on a shared application, such as GitHub. Will need to become more familiar with features / best practices.
* When to commit
  + At completion of each functional chunk
    - “dirty commit”
  + How much work would you tolerate losing ?
  + Consider breaking up work into – at most – daily commits
* Branching
* Tests
  + Decide that you want to do testing
    - Unit tests
    - Integrations tests
    - Continuous integration testing
  + What to test?
    - Normal conditions and bad arguments
  + Have you tested?
    - Not really; informal
    - Need to look at formal methods (e.g. unit test r)
* Issue Tracking software
  + Maintain an issue list; bugs, comments, requests, desirable features.
* Packages
  + Making your own pacages
    - Templates; devtools
    - Packaging protocols
      * Advice from H. Wickham
    - Dissemination - CRAN
* Documentation
  + Google “automatic documentation” of R
  + Discuss with Laurette what kind of documentation is needed.
* IDE
  + Integrated development environments
    - Different panels
    - Different colors (i.e. syntax highlights)
      * Files, functions, variables, comments
    - Staged/un-staged changes
  + Ask Carl about setting up R-studio with GitLab/Hub
    - You don’t load R studio on a super computer
    - Ask Laurette about R studio Git Integration
* Collaboration
  + Meetings
  + Asynchronous communication
  + File-sharing
  + Document sharing

**28 November 2018**

**Activity - why does software not get used**

* Readability/documentation
* Availability - public, licensing
* Marketing, publications
* Useful – is it solving a problem that anyone is interested
* Compatibility/integration with other software
* Maintenance
* Elegance
* Vogue of language
  + When users need to care about the language
* Uniqueness
* Complexity/replaceability
* Efficiency

**Interface**

* Good interface
  + Age of empires 2
    - Well designed, good upgrade from AOE 1 (AOE 3 was too much)
    - Mouse, key board, monitory
  + HTML/CSS
    - Suboptimal in neatness
    - Prefer engaging HTML/CSS with Bootstrap, evolution of templates to code sniplets.
    - Works with jquery and javascript
  + WhatsApp vs. E-buddy

**28 December Practical**

On day 1 (project design) we talked about modularity, and you made a process map for your project. Think about that as you begin these exercises.

* Prepare to become a package author.
  + find out how and where packages are distributed in your preferred language, e.g., CRAN for R.

R, Git

* + create an account

Done -

* Imagine carving off part of your code as a re-usable package.
  + before thinking about other stakeholders, think about your future self. How could some parts of your code be used by yourself in other projects?

Coduct surveys; estimate prevalence, compare prevalence estimates from multiple surveys; benchmark /optimize other surveys

* + who are your target users or stakeholders, beyond yourself? What about your immediate co-workers? What about other people in the same field? In other fields? What is going to make your module attractive to them?
  + Write a few sentence about what your package does.
  + search the web for similar packages in the same or other languages. What did you learn?
* Research the mechanics of package generation in your preferred language
  + identify a source of guidelines or best practices and bookmark that

R packages Hadley Wickham

<http://web.mit.edu/insong/www/pdf/rpackage_instructions.pdf>

http://r-pkgs.had.co.nz

* + install any tools that you need to generate a package

devtools, Roxygen

* Recall your choices yesterday for workspace organization
  + now that you know more about package generation for your preferred language, are they any particulars about filesystem arrangement? How do they work with (or against) choices you made yesterday?

Each function will have own script/help files

* + In addition to directory structure, are there are system configuration elements (e.g., environment variables, program aliases) that are required for packaging?

For now, no

* Do a quick cost-benefit analysis. You have imagined a package and explored the tools and best practices for generating and distributing packages.
  + what extra work is required for you to make a package?

Going through the R package “checklist” Hadley

* + what are the costs and benefits?

Time, useful to others

Parts of your project that might be replaced with existing modules

This assumes that you know how to find packages for your preferred language and platform.

* Identify replaceable components in your project.
  + list 3 functionalities of your code that someone has probably implemented before. Maybe there is a better version out there that would improve your project.
* ODEs using desolve
* Car-box
  + poke around on the web. Try to find a module that does what you want
  + what extra functionality would you get from importing this module (e.g., handling errors or exceptions)? Is it going to make your life easier? how do you decide whether importing this is worth the effort?
* Map out the steps to replace part of your code with an imported module
  + does this add further dependencies? is that a problem? How do you decide?

Desolve dependencies; which is only base r

* + Will you wrap imported functions (methods) to match yours, or rewrite your code to call imported functions directly?
  + how will you declare this dependency to potential users?

List desolve among dependencies, svy

Ways to extend your project with existing modules

This assumes that you know how to find packages for your preferred language and platform.

* Write down 3 ways that you might want to extend your project with new functionality. Explore ways to implement that functionality using existing modules. For instance

Ggplot, svy, inctools

* + IO. Maybe you want to store outputs in a database instead of ad hoc files. Maybe you you want to input JSON instead of ad hoc files. Are there modules for that?

For now, not really ? Maybe, user provided matrix of cum surv prob.

* + estimation methods. Are there some other ways to do parameter estimation, perhaps more powerful or flexible than what you are doing?
  + graphical user interfaces. Would your code benefit from a web interface? What’s the easiest way to do that?
* Thinking about other packages your hypothetical package might be used with - what are their interfaces? Is your interface written with the same approach? What new way of thinking would potential users have to learn to combine your library with others? Are there ways to make your interface more similar to that of other libraries a scientist would commonly use alongside your work?

**PRACTICALS, 29 November 2018**

Code vs Input

* Does your project have “code” currently that might alternatively be thought of as input? E.g., parameter or configuration values that are included directly in your scripts, instead of in input files.

Yes – there are input values. For example; input for user defined incidence function which may be supplied as a dataframe/array/matrix of values

However, perhaps no efficiency would be gained.

* For cases where you do, do you find yourself regularly changing those values, e.g. to run a new scenario? If other people were to use your code, might they want to change those values?

No. The functions are generic enough where a user can have flexibility

Kinds of Input & Output

* List your inputs and outputs, and label those data with their best fit as tabular, relational, hierarchical, key-value, or human-oriented (e.g., a plot or sound file).?

Inputs; arguments to functions

Outputs;

“Final”

* **Matrix of susceptible**
* **Matrix of infected**

“intermediate”

* Probability of survival in among susceptible
* Cumulative probability of survival among susceptible
* Probability of infection among susceptible
* Probability of survival among infected
* For your inputs and outputs, which are general purpose formats (e.g., csv, json, sqlite, protobuf) with libraries for many languages, specialized (e.g., rds, docx) with application-specific access, or custom (e.g., human readable, but with specialized parsing rules)? What are your reasons for using general vs specialized vs custom formats?

.csv, txt, maybe sql

* For your inputs and outputs, which are human-readable (plaintext) vs machine-only readable (binary)? What are your reasons for using plaintext vs binary formats for different parts of your project?

Everything is human readable

Intermediate Results

* Compared to your project overall outputs, are there intermediate files that are only used internally to your project? Where do they fall relative to the overall process steps you outlined on Monday?

Matrix of cumulative survival probability; this is an intermediate output to the matrices of infected and susceptible. But also possibly a final output result in and of itself

* Does your project have a “resume” capability? If so, how do you implement it? If not, how might it benefit from one?

No; we should consider it as computing requirements increase.

* Do you cache any inputs? If so, why?

No; we don’t

Testing & Validation

* Which of your project requirements specifically concern an input or output file feature? E.g., must read / write particular format, size, …? Or more generally the creation or consumption of specific data in specific files?

None at present

* Does your project include in testing of inputs or outputs? E.g., validation of inputs, checking for missing data, confirmation of output consistency with some independent standard, …?
* We have to develop tests for functions and output tables.

Amounts & Locations

* How much input does your project rely on? Estimate in kilobytes, Mbs, Gbs, (Tbs?) as appropriate.

bytes

* …how much output does it create?

Could be a little could be a lot, depending on time/age steps, runs, params, etc

* Is the input used or output created by your project all local? Or does some travel over a network?

Local and network for example shared DB, OD, repository folders

* How much time does your project spend loading data–e.g., reading in a csv vs how much time the program spends analyzing the information in the file?

No time in, variable time out (not entirely clear at this point);

* How much time does your project spend writing data–e.g., generating and saving a plot vs generating the time series being plotted.

not entirely clear at this point, but probably not much

Sharing

* Are your raw inputs shared? Cleaned up inputs? Why or why not?

We should consider a way to store/output the inputs (initial states, parameters) as they relate to outputting tables.

* How do your collaborators manage data? Which services or processes did your group consider before settling on the current one?

We should consider a way to store/output the inputs (initial states, parameters) as they relate to outputting tables.

* Git repo, DB, OneDrive ? TBD

**Meeting with Faikah**

* Population model
  + Determine important factors for base population
    - Prior Distributions 🡪run simulation 🡪 posterior