Building Models:Documentation

- In R code documentation can be in any format as long as it starts with a #, it is helpful to use a constant format/structure for documentation
  - helpful when your model ends up having multiple functions
  - easier to interpret
- There is also a standard format for documentation that can be read by automatic programs (roxygen2) - an R package that generate “standard” R documentation
- These automated approaches for building documentation (like roxygen2) and meta data (descriptions of data sets) are increasingly common - so you should get into the practice of being structured in your approach to documentation
- We will use the conventions that work with roxygen2 - and later use this program to generate formal R documentation

Building Models: Documentation

- Two Parts
- Description - summary of what your model/function does
- Tagged (using special "key" words)
  - @param #inputs/parameter description
  - @return # what your function returns (outputs)
  - @example # how to use it
  - @references # citations or urls
  - @author # YOU
  - (you don't need all of these and there are more tags, but start with at least param and return, example is a good idea

Documentation

- In the header (before the function) part of your model, use #’ for comments (roxygen2 needs this)
- within the function, use #

```
#' Power Generation
#'
#' This function computes instantaneous power generation
#’ from a reservoir given its height and flow rate into turbines
#' @param rho Density of water (kg/m3) Default is 1000
#' @param g Acceleration due to gravity (m/sec2) Default is 9.8
#' @param Kefficiency Turbine Efficiency (0-1) Default is 0.8
#' @param height height of water in reservoir (m)
#' @param flow flow rate (m3/sec)
#' @author Naomi
#' @examples power_gen(20, 1)
#' @return Power generation (W/s)

power_gen = function(height, flow, rho=1000, g=9.8, K=0.8) {

# calculate power
result = rho * height * flow * g * K
return(result)
}
```

Automatic "Help Page" Documentation Creation in R

- You need two packages "devtools" and "roxygen2"
- We will do two things: both to help organize our model
  - create a package to contain our code
  - document objects in that package
    - objects can be models, or data
- Note that modern model development (the 4th paradigm) has a close link between models and data

Documentation Steps:

- Make your working directory your project directory
- Call the *document* routine - which will create documentation for ANY of the *.R files you have in the R directory (you can have many of them)
- use *help(test.R)* to see the results
- the next time your run R, load your new package

```
setwd("classexamples")
document()
help(power_gen)
```

Documentation

- Some other useful features doxygen2
- anything after the @param or @examples or @return until the next @ is included in the description

Documentation

- Under the “hood” - what doxygen2 does is create a file in your project under the man directory called *.Rd
- So we created classexamples/man/power_gen.Rd automatically
- It is a file that R uses to generate the documentation
- You could build it by hand if you want!

Documentation

- In the header (before the function) part of your model, use #’ for comments (roxygen2 needs this)
- within the function, use #

```
#' Power Generation
#'
#' This function computes instantaneous power generation
#’ from a reservoir given its height and flow rate into turbines
#' @param rho Density of water (kg/m3) Default is 1000
#' @param g Acceleration due to gravity (m/sec2) Default is 9.8
#' @param Kefficiency Turbine Efficiency (0-1) Default is 0.8
#' @param height height of water in reservoir (m)
#' @param flow flow rate (m3/sec)
#' @author Naomi
#' @examples power_gen(20, 1)
#' @return Power generation (W/s)

power_gen = function(height, flow, rho=1000, g=9.8, K=0.8) {

# calculate power
result = rho * height * flow * g * K
return(result)
}
```

```
% Generated by roxygen2 (4.1.0): do not edit by hand
% Please edit documentation in R/power_gen.R
\name{power_gen}
\alias{power_gen}
\title{Power Generation}
\usage{
power_gen(height, flow, rho = 1000, g = 9.8, Keff = 0.8)
}
\arguments{
\item{height}{height of water in reservoir (m)}

\item{flow}{flow rate (m3/sec)}

\item{rho}{Density of water (kg/m3) Default is 1000}

\item{g}{Acceleration due to gravity (m/sec2) Default is 9.8}

\item{Keff}{Turbine Efficiency (0-1) Default is 0.8}
}
\value{
Power generation (W/s)
}
\description{
This function computes instantaneous power generation from a reservoir given its height and flow rate into
turbines
}
\examples{
power_gen(20, 1)
}
\author{
Naomi
}
```

This is what it looks like….
the use of \tag {} is very common in this type of documentation language