

A Brief Introduction to R

Cameron Bracken

21 January 2010

1 What is R?

<http://www.r-project.org/about.html>

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly ATT, now Lucent Technologies) by John Chambers and colleagues.

R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, ...) and graphical techniques, and is highly extensible. R is very popular in Hydrology.

One of R's strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed.

More concisely R is:

- A powerful interpreted programming language,
- A data analysis and visualization tool,
- Very addicting once you get the hang of it.

2 Getting Started

2.1 Installation

R can easily be installed for most any operating systems, visit:

<http://www.r-project.org/>

2.2 The Interface

To start R in windows open R.exe, in Mac OS X open R.app, in Linux or Mac OS X type R on the command line.

2.3 Data Types

The basic data types in R are:

- **Integer**, eg. `x <- 5L` (Basic numbers in R will all be stored as double precision unless you add the L)
- **Numeric**, eg. `x <- pi/2`

- **Logical**, eg. `x <- TRUE`
- **Character**, eg. `x <- "foo"`

You may have noticed the `<-` assignment operator. This works in both ways, eg. `5 -> x`. The `=` operator is equivalent to the `<-` operator.

All of the above examples result in `x` as a scalar. There are a few way of collecting scalars into a single variable:

- **Vector** - A one dimensional collection of scalars with no orientation (there are no such things as column or row vectors). Vectors have a single data type depending on their contents. Eg. `x <- c(1,2,3,4)` is a numeric vector and `y = c("foo","bar")` is a character vector. The `c()` function sticks things together into vectors.

Individual elements can be extracted or assigned brackets. Eg. If `x = 1:5` then `x[1]` would give back 1. Also subsets of vectors can also be vectors, eg., if `y = c("a","b","c","d")`, `y[c(1,2,3)]` would give back the vector `c("a","b","c")`.

- **Matrix** - A two dimensional collection of scalars. eg. `x <- matrix(0,nrow=2,ncol=2)`. A matrix is subset with two indices like `x[1,1]`.
- **List** - Is a vector with arbitrary data types in each element. For example, If

```
L <- list(x = 1:5, y = c("foo","bar"))
```

then we can extract the first entry using `L[[1]]` or by `L$x`.

- **Data frame** - A special data format for statistical data, it can be treated as a matrix with column names or a list. The example below uses a data frame.

3 Example - Data manipulation

This example uses the data file available at: <http://animas.colorado.edu/~bracken/CVEN4333/plant-inflow.txt>.

To load a local file without using absolute paths we need to set the working directory using the `setwd('/path/to/data')`, just like the `cd` command in unix.

```
> # First read in the data file
> dat <- read.table('plant-inflow.txt',header=T)
> dat[1:10,]
```

	year	month	day	inflow
1	1988	1	1	1.99
2	1988	1	2	2.43
3	1988	1	3	2.72
4	1988	1	4	2.49
5	1988	1	5	2.08
6	1988	1	6	2.03
7	1988	1	7	2.88
8	1988	1	8	4.02

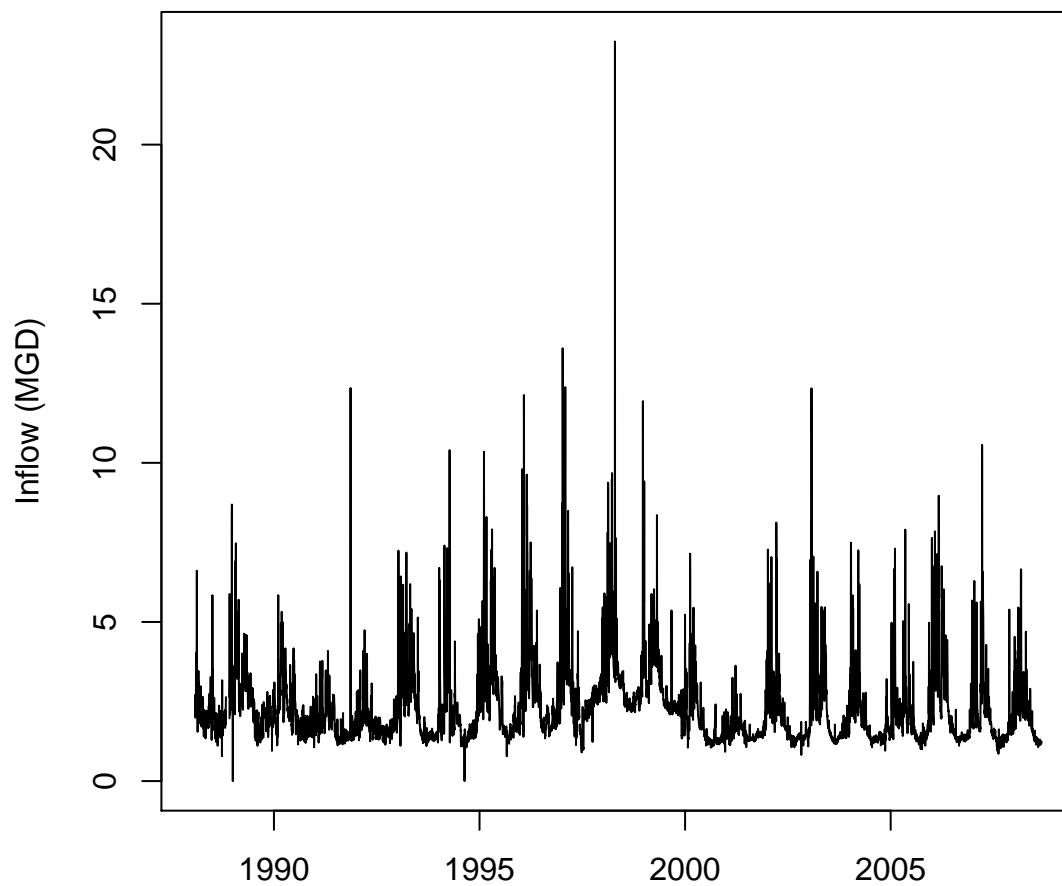
```

9 1988      1   9   3.77
10 1988     1  10   3.89

>      # Remove all of the missing values
> dat$inflow[dat$inflow==--9999] <- NA
>      # Create a vector of decimal years
> years <- dat$year + dat$month/12 + dat$day/365
>      # And plot it
> plot(years, dat$inflow, type='l', xlab='',
+       ylab='Inflow (MGD)', main='AWTP Inflow time series')

```

AWTP Inflow time series



4 Links

R Home page - <http://www.r-project.org/>

R Reference Card - <http://cran.r-project.org/doc/contrib/Short-refcard.pdf>

A useful tutorial - <http://www.cyclismo.org/tutorial/R>