

Introduction

In recent months the editors of this report have been more and more exposed to a stats program called R. Its an open source program which is freely available online along with a user interface called R-Studio which makes working with this program much easier. R has got a number of packages written by scientists and authors around the world which make it easier to get an output for specific problems. The problem with a program like R is that it's difficult to get used to and use if you are not doing it regularly. A recent workshop we attended has again piqued our interest in R and in an effort to learn the program for ourselves we'll be doing a monthly lab primarily focussed on R used to answer a specific question. We'll be using data sources that we make available to you so that you can run the code as you see it below in R Studio and you can see the output. Even if you just copy and paste this code once a month it means that you get a chance to learn a great program. We also want to encourage all of you that will start following our Back Page Lab to consider sending us a dataset and the code to answer one of your epi problems. We're just starting out but we'll try keep the format the same, giving you the best chance to get into this software. We are hoping that this leads to future labs outside of learning R-but the rule will be that the data is freely available and the software is open source

We want to make this work for you - please email john@elsenburg.com if you need any help installing the basic R programs/packages which you'll need - we'll take this step by step and its all quite fun. We're also learning as we go.

Epi Lab color code

Software/Packages/Add-ins required

Software/Packages/Add-ins recommended

Description text

R code to copy/paste into console

R code to copy/paste into console that needs adjustment to your personal workspace

Website where you can download requirements

System requirements

- R - <http://cran.r-project.org/bin/windows/base/>
- R Studio - www.rstudio.com/ide/download/desktop
- **epitools R package** (this download code will be included in the code below so no need to pre-install this)
- **Internet connection**

Lab description

This month we create a basic epidemic curve using the **epitools** package in R. Epidemic curves are used in outbreak investigations. They can assist in determining an unknown outbreak aetiology or in determining if control measures have assisted in controlling an outbreak. Epidemic curves give an indication of the type of outbreak, e.g. sporadic vs endemic vs propagating vs point.

The code

#You can paste this entire code into your R console - the # makes a line of text comment out

#I recommend that you copy and paste the individual commands (in BLUE) to show how you built your epidemic curve. Hit #ENTER after each command in your console

#To read in data from an online source use the code below. This data is our Mamre 2011 AHS outbreak case data set

```
ahscases<-read.table('http://www.jdata.co.za/backpagelabs/backpagelabs_jdg_ahsoutbreak.txt',header=TRUE,as.is=TRUE)
```

#This file is a text file with a space between columns which is the format that read.table takes as a default - you can copy and paste that website address into your web browser and have a look at the data

#I have specified that the headers must be imported as my text file had headers as column names

#The as.is function is NB! and in this case is used so that dates are imported as character classes and NOT as factors

#which is what would have happened if the as.is function had been left out

#A data frame is made called ahscases (note: the symbol "<-" in R code indicates that you are allocating whatever data

#is made from your function to a data set or variable - look to see if you see ahscases in your ENVIRONMENT WINDOW #in R Studio Top Right Window)

```
head(ahscases)
```

#head shows the top 6 rows of data of a data source (tail would have shown the bottom 6)

```
class(ahscases)
```

#this shows a data frame has been imported

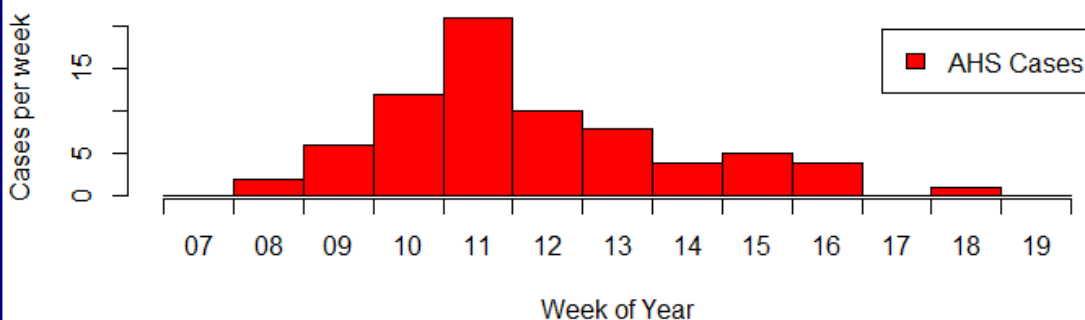
```

class(ahscases$casedate)
class(ahscases$deathdate)
#death date and case date are character classes which we need to convert to a date format
#to convert a character to a date use the as.Date function
ahscases$casedate<-as.Date(ahscases$casedate)
ahscases$deathdate<-as.Date(ahscases$deathdate)
#now we re check the class of these columns
class(ahscases$casedate)
class(ahscases$deathdate)
#OK now for the output of this back page lab - an epidemic curve of the cases
#On the x axis we want the week of outbreak, on the y axis we want the number of cases within that week
#if you haven't installed the epitools package do so
install.packages("epitools")
#load the package - tick it in the packages section (bottom right window in R Studio) is the easiest
#You can also load it like this (but your directory will be different)
#library("epitools", lib.loc="C:/Users/johng/Documents/R/win-library/3.0")#
#run the code to make your epicurve
graphlabels<-epicurve.weeks(ahscases$casedate, axisnames=FALSE, legend= "AHS Cases",xlab = "Week of Year", ylab =
"Cases per week", col="red")
#here we have made a curve but we have taken out the axis labels (the default is not helpful), we have added a legend,
#we have added a X and a Y Axis label as well as changing the column colour to red
#note above: along with plotting the graph we have also allocated the graph data to a dataset called graphlabels Within
#this dataset (you can view it by typing in graphlabels into your console) are weeks of the year within a column called
#cweek
axis(1, at = graphlabels$xvals, labels = graphlabels$cweek, tick = FALSE, line = 0)
#here we have put in an axis - 1 indicates the bottom axis (X), at indicates where we want the labels, labels indicates
#what labels we want to add, we have removed tick marks and we said not to add a line to the axis we have created
#Now to add a title and you're done!
title("Epidemic curve of the 2011 AHS Outbreak")

```

The output

Epidemic curve of the 2011 AHS Outbreak



Citations

R Core Team (2014). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>

Tomas J. Aragon Developer (2012). epitools: Epidemiology Tools. R package version 0.5-7. <http://CRAN.R-project.org/package=epitools>