BUS212 6b Automated Data Collection

## Introduction

This code is adapted from the book *AUTOMATED DATA COLLECTION WITH R*, by Simon Munzert, Christian Rubba, Peter Meissner, Dominic Nyhuis. The example comes from Chater 1, and demonstrates how to read in data from a website, prepare it for analysis, and then produce some visualization.

The example uses the Wikipedia page about the *Endangered World Heritage Sites*, which are places in the world that are (a) included in the list of antiquities and natural wonders known as World Heritage Sites and (b) in danger due to natural or political reasons.

The script requirs four packages:

# load packages  
library(stringr)  
library(XML)  
library(maps)  
library(RCurl)

## Loading required package: bitops

The relevant website is <https://en.wikipedia.org/wiki/List_of_World_Heritage_in_Danger>. The first code chunk reads the relevant data into an object called heritage\_parsed using the command getURL.

# parsing from Wikipedia web site  
fileURL <- "https://en.wikipedia.org/wiki/List\_of\_World\_Heritage\_in\_Danger"  
xData <- getURL(fileURL)  
heritage\_parsed <- htmlParse(xData)

The heritage\_parsed object contains the full list of sites as well as those in danger. The next few lines separate the full table and the table of endangered sites.

tables <- readHTMLTable(heritage\_parsed, stringsAsFactors = FALSE)  
danger\_table <- readHTMLTable(heritage\_parsed, stringsAsFactors = FALSE, which = 2) # alternatively: directly select second table  
  
# extract desired table  
danger\_table <- tables[[2]]  
names(danger\_table)

## [1] "Name" "Image" "Location" "Criteria"   
## [5] "Area\nha (acre)" "Year (WHS)" "Endangered" "Reason"   
## [9] "Refs"

# select and rename columns  
danger\_table <- danger\_table[,c(1,3,4,6,7)]  
colnames(danger\_table) <- c("name","locn","crit","yins","yend")  
danger\_table$name[1:3]

## [1] "Abu Mena" "Air and Ténéré Natural Reserves"  
## [3] "Ancient City of Aleppo"

In its raw form, the data are not ready for analysis. This chunk performs several steps to clear and reorganize the data. Towards the end of the chunk, notice that it is pulling out longitude and latitudes in order to map the sites.

# cleanse criteria  
danger\_table$crit <- ifelse(str\_detect(danger\_table$crit, "Natural")==T, "nat", "cult")  
  
# cleanse years  
danger\_table$yins <- as.numeric(danger\_table$yins)  
danger\_table$yins

## [1] 1979 1991 1986 1980 1979 2011 1982 1982 1982 2003 1994 1996 1986 2012  
## [15] 1983 1993 2006 2003 1998 1979 1980 1980 1985 2000 1993 2005 1980 2004  
## [29] 1988 2002 2004 1981 2016 1981 1996 1981 1986 1986 1988 1982 2007 1982  
## [43] 1985 1981 1984 2007 1978 1980 1988 2004 2001 2004 1979 2014

danger\_table$yend

## [1] "2001-" "1992-" "2013-"   
## [4] "2013-" "2013-" "2013-"   
## [7] "2016-" "2016-" "2016-"   
## [10] "2003-" "2010-" "2009-"   
## [13] "1986-" "2012-" "2003-"   
## [16] "2005-" "2013-" "2003-"   
## [19] "2013-" "1993<U+0096>2007, 2010-" "2012-"   
## [22] "1984<U+0096>1992, 1996-" "2015-" "2016-"   
## [25] "2000-" "2005-" "1997-"   
## [28] "2012-" "1997-" "2002-"   
## [31] "2006-" "1992-" "2016-"   
## [34] "2007-" "1997-" "1982-"   
## [37] "2015-" "2016-" "2016-"   
## [40] "2015-" "2010-" "1996<U+0096>2007, 2011-"  
## [43] "2016-" "2004-" "1999-"   
## [46] "2007-" "1996-" "2013-"   
## [49] "2012-" "2012-" "2010-"   
## [52] "2011-" "1994-" "2014-"

yend\_clean <- unlist(str\_extract\_all(danger\_table$yend, "^[[:digit:]]{4}"))  
danger\_table$yend <- as.numeric(yend\_clean)  
danger\_table$locn[c(1,3,5)]

## [1] "EgyAbusir, Egypt30°50'30<U+2033>N 29°39'50<U+2033>E<U+FEFF> / <U+FEFF>30.84167°N 29.66389°E<U+FEFF> / 30.84167; 29.66389<U+FEFF> (Abu Mena)"  
## [2] "Syria !Aleppo Governorate,  Syria36°14'0<U+2033>N 37°10'0<U+2033>E<U+FEFF> / <U+FEFF>36.23333°N 37.16667°E<U+FEFF> / 36.23333; 37.16667<U+FEFF> (Ancient City of Aleppo)"  
## [3] "Syria !Damascus Governorate,  Syria33°30'41<U+2033>N 36°18'23<U+2033>E<U+FEFF> / <U+FEFF>33.51139°N 36.30639°E<U+FEFF> / 33.51139; 36.30639<U+FEFF> (Ancient City of Damascus)"

# get countries  
reg <- "[[:alpha:] ]+(?=[[:digit:]])"  
country <- str\_extract(danger\_table$locn, perl(reg)) # use forward assertion in Perl regular expression

## perl is deprecated. Please use regex() instead

head(country)

## [1] "Egypt" "Niger" "Syria" "Syria" "Syria" "Syria"

country[29] <- "Côte d'Ivoire / Guinea"  
country[32] <- ""  
danger\_table$country <- country  
  
# get coordinates  
reg\_y <- "[/][ -]\*[[:digit:]]\*[.]\*[[:digit:]]\*[;]"  
reg\_x <- "[;][ -]\*[[:digit:]]\*[.]\*[[:digit:]]\*"  
y\_coords <- str\_extract(danger\_table$locn, reg\_y)  
(y\_coords <- as.numeric(str\_sub(y\_coords, 3, -2)))

## [1] 30.84167 18.28300 36.23333 32.51806 33.51139 36.33417 32.82500  
## [8] 32.63833 32.80528 35.45667 42.26222 17.31700 -8.11111 31.70444  
## [15] 9.16700 11.41700 34.78167 34.83194 -11.68306 25.31700 9.55389  
## [22] 4.00000 35.58806 39.05000 14.20000 -20.20833 -2.50000 53.40667  
## [29] 9.00000 34.39667 42.66111 7.60000 6.83972 13.00000 2.00000  
## [36] 31.77667 15.35556 30.13333 13.90639 15.92694 -14.46700 15.74444  
## [43] 24.83333 -8.95778 -2.00000 34.20000 13.18300 34.55417 16.77333  
## [50] 16.28933 0.32917 -2.50000 0.91700 31.71972

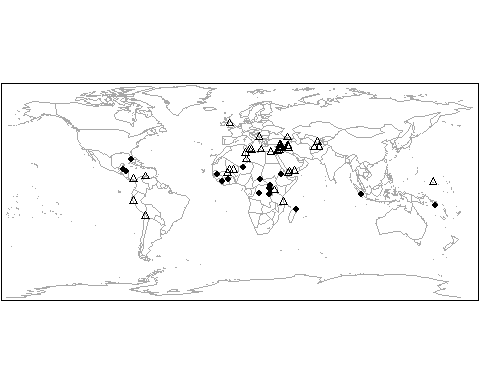
danger\_table$y\_coords <- y\_coords  
x\_coords <- str\_extract(danger\_table$locn, reg\_x)  
(x\_coords <- as.numeric(str\_sub(x\_coords, 3, -1)))

## [1] 29.6638900 8.0000000 37.1666700 36.4816700 36.3063900  
## [6] 36.8441700 21.8583300 14.2930600 12.4850000 43.2625000  
## [11] 42.7163900 -87.5330000 -79.0750000 35.2075000 -3.6670000  
## [16] -69.6670000 36.2630600 67.8266700 160.1830600 -80.9330000  
## [21] -79.6558300 29.2500000 42.7183300 66.8333300 43.3170000  
## [26] -69.7944400 28.7500000 -2.8444400 21.5000000 64.5161100  
## [31] 20.2655600 -8.3830000 158.3308300 -12.6670000 28.5000000  
## [36] 35.2341700 44.2080600 9.5000000 -4.5550000 48.6266700  
## [41] 49.7000000 -84.6750000 10.3333300 39.5227800 21.0000000  
## [46] 43.8670000 38.0670000 38.2666700 -2.9994400 0.0449111  
## [51] 32.5533300 101.5000000 29.1670000 35.1305600

danger\_table$x\_coords <- x\_coords  
danger\_table$locn <- NULL

With data preparation completed, here is a map of the locations of the endangered sites.

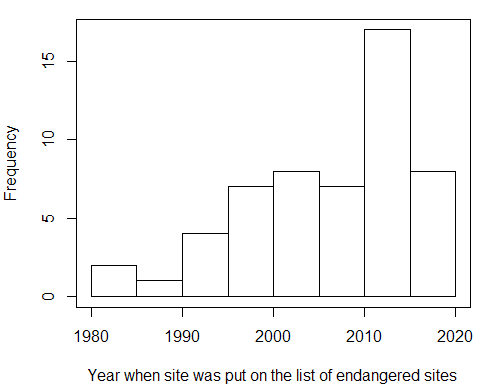
par(oma=c(0,0,0,0))  
par(mar=c(0,0,0,0))  
pch <- ifelse(danger\_table$crit == "nat", 19, 2)  
map("world", col = "darkgrey", lwd = .5, mar = c(0.1,0.1,0.1,0.1))  
points(danger\_table$x\_coords, danger\_table$y\_coords, pch = pch, col = "black", cex = .8)  
box()

 Next, the code creates a histogram showing the years in which sites became endangered.

# table heritage criteria  
table(danger\_table$crit)

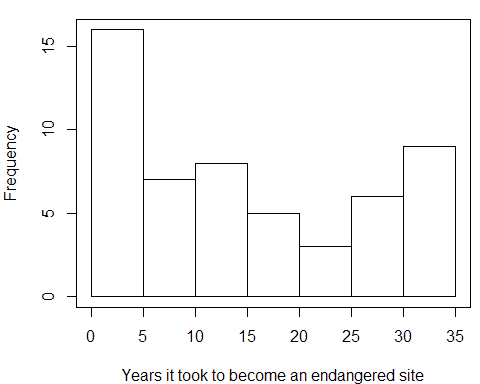
##   
## cult nat   
## 37 17

# plot year of endangerment  
  
par(oma=c(0,0,0,0))  
par(mar=c(4,4,1,.5))  
hist(danger\_table$yend, freq=TRUE, xlab="Year when site was put on the list of endangered sites", main="")  
box()



And finally, a graph showing time between inscription and endangerment.

duration <- danger\_table$yend - danger\_table$yins  
par(oma=c(0,0,0,0))  
par(mar=c(4,4,1,.5))  
hist(duration, freq=TRUE, xlab="Years it took to become an endangered site", main="")  
box()



```