Extra help for Assignment 1 Exercise 1

credits to Tomas Morley and Jurre Veerman

12 November 2016

You will need a couple of tools for this assignment. The first is `regular expressions`, which are extremely useful and available in many programming languages including `R`. The two functions you will probably need are `strsplit` and `gsub` and below you can see a little example of how to use them. Try running this code for yourself to see how some of the regular expression functions work. And don't forget to look at the help page using `?grep`

```r
# first create some example data

Rteam <- c("pilot:maarten", "co-pilot:vincent", "steward:jurre", "steward:tom")
Rteam <- sample(Rteam, replace=TRUE, size=20)

# grep checks each element in a characer string for
# a pattern. If the pattern is found it returns
# the index of that element

grep(pattern="pilot", x=Rteam)
grep(pattern="steward", x=Rteam)

# gsub replaces a pattern in a character string
# with a different pattern

gsub(pattern="steward", replacement="TA", Rteam)

# strsplit splits a character string based
# on a pattern and returns a list where each component
# of the list contains the text which was on either
# side of the pattern

strsplit(Rteam, split = ":")
split <- strsplit(Rteam, split = ":")
sapply(split, function(x) x[1])
sapply(split, function(x) x[2])
```

The second tool is the ability to break down a complicated messy problem into smaller problems. We can see that the first variable in the data `unit.sectperf.geo.time` is very different from the rest. A closer look tells us that it always has three abbreviations, separated (split) by a comma. The other columns are all very similar to each other and have names like `X2003` and `X2008`. Sometimes there is a number and sometimes there is a number and a letter. Sometimes there is a `:`. When there is a number followed by a letter it is separated (split) by white space `" "`.

As the first column is different from the rest, we might want to separate it from the rest of the data and treat it differently. It's unlikely that we need a function for this because whatever we do to this column will not be repeated on other columns. But the `X20??` columns are very similar. So we should make a function because what we do to one of them is probably what we will do to the other ones. It's useful to think what this function would ideally do. For example, it might replace all the `:` with NA values, split the numbers and the letters into separate columns and then return two columns in a list. If we could do that then we would be able to combine all the pieces of this problem into one data frame!