Titles and citations — Letchford et al. 2015

> Letchford et al. (2015) found an interesting pattern: papers that have shorter titles tend to fare better in terms of citations. They took top-cited papers from a variety of journals, and ranked them by title length (in number of characters), and number citations received (as of November 2014). Then they performed a Kendall's tau test to see whether these rankings are correlated. A negative correlation would mean that the articles with longer titles tend to be ranked low for citations.
>
> The file `Letchford2015_data.csv` contains the data needed to replicate their results.
>
> 1. Write a program that performs the test described above using all the papers published in 2010. The program should do the following: 1) read the data; 2) extract all the papers published in 2010; 3) rank the articles by citations, and by title length; 4) compute the Kendall's tau expressing the correlation between the two rankings. For this dataset, the Authors got a tau of about -0.07 with a significant p-value.

First, we need to read the data:

```
# Read the data
l2015 <- read.csv("../data/Letchford2015_data.csv", stringsAsFactors = FALSE)
# Check dimensions
dim(l2015)
```

```
## [1] 140000      4
```

```
# Print first few lines
head(l2015)
```

```
##   year                              journal title_length cites
## 1 2007    Molecular Biology and Evolution           75 17958
## 2 2007                     Nature Materials           20 11633
## 3 2007                       Bioinformatics           35  8437
## 4 2007                                 Cell           83  5846
## 5 2007 Ca-A Cancer Journal for Clinicians           23  5744
## 6 2007 American Journal of Human Genetics           84  5651
```

Now extract only the papers published in 2010:

```
p2010 <- l2015[l2015$year == 2010,]
```

To rank the manuscripts according to title length, we can use the function `rank` (check `?rank`), which ranks the entries, resolving possible ties:

```
# Example of use of rank
rank(c(1,2,3,2,1,5,3,4))
```

```
## [1] 1.5 3.5 5.5 3.5 1.5 8.0 5.5 7.0
```

Let's store the ranking of title lengths and citations separately:

```
rank_titlelength <- rank(p2010$title_length)
rank_citations <- rank(p2010$cites)
```

Finally, let's calculate Kendall's tau. A simple way is to invoke `cor`:

```
cor(rank_citations, rank_titlelength, method = "kendall", use = "pairwise")
```

```
## [1] -0.06551962
```

Which is similar to that reported by the Authors (they peformed some filtering of the articles before analyzing them, so we don't expect a perfect match). To get also a p-value, you can use `cor.test`:

```
cor.test(rank_citations, rank_titlelength, method = "kendall", use = "pairwise")
```

```
##
##  Kendall's rank correlation tau
##
## data:  rank_citations and rank_titlelength
## z = -13.753, p-value < 2.2e-16
## alternative hypothesis: true tau is not equal to 0
## sample estimates:
##         tau
## -0.06551962
```

2. Write a function that repeats the analysis for a particular journal-year combination. Try to run the function for the top scientific publications `Nature` and `Science`, and for the top medical journals `The Lancet` and `New Eng J Med`, for all years in the data (2007-2013). Do you always find a negative, significant correlation (i.e., negative tau with low p-value)?

For this point, we need to write a function, that takes as input the data, a `journal` and a `year`. The function will then extract the relevant data, and run the test.

We can start building the function from this skeleton:

```
compute_tau_journal_year <- function(my_data, my_journal, my_year){
  # First, filter the data
  my_subset <- my_data[my_data$journal == my_journal & my_data$year == my_year,]
  print(c(my_journal, my_year, "Articles:", dim(my_subset)[1]))
}
```

and try running it to make sure everything is good:

```
compute_tau_journal_year(12015, "Nature", 2010)
```

```
## [1] "Nature"    "2010"      "Articles:" "575"
```

Next, we write the analysis proper:

```
compute_tau_journal_year <- function(my_data, my_journal, my_year){
  # First, filter the data
  my_subset <- my_data[my_data$journal == my_journal & my_data$year == my_year,]
  # Rank by title length and citations
  rank_titlelength <- rank(my_subset$title_length)
  rank_citations <- rank(my_subset$cites)
  # Return the value of tau
  return(data.frame(Journal = my_journal,
                    Year = my_year,
                    tau = cor(rank_citations, rank_titlelength,
                              method = "kendall", use = "pairwise")))
}

# Try running it
compute_tau_journal_year(l2015, "Nature", 2010)
```

```
##   Journal Year         tau
## 1  Nature 2010 -0.03833785
```

We can write a fancier version that stores also the p-values, and checks that there are enough articles:

```
compute_tau_journal_year <- function(my_data, my_journal, my_year){
  # First, filter the data
  my_subset <- my_data[my_data$journal == my_journal & my_data$year == my_year,]
  if (dim(my_subset)[1] < 2) {
    tau <- NA
    p.value <- NA
  } else {
    # Rank by title length and citations
    rank_titlelength <- rank(my_subset$title_length)
    rank_citations <- rank(my_subset$cites)
    # Run the test
    my_test <- cor.test(rank_citations, rank_titlelength,
                        method = "kendall", use = "pairwise")
    tau <- as.numeric(my_test$estimate)
    p.value <- as.numeric(my_test$p.value)
  }
  return(data.frame(Journal = my_journal,
                    Year = my_year,
                    tau = tau,
                    p.value = p.value))
}
compute_tau_journal_year(l2015, "Nature", 2010)
```

```
##   Journal Year         tau   p.value
## 1  Nature 2010 -0.03833785 0.1746622
```

Now let's run it for all years and a few journals:

```
results <- data.frame()

for (year in 2007:2013){
```

```
  for (jr in c("Nature", "Science", "The Lancet", "New Eng J Med")){
   results <- rbind(results,
                    compute_tau_journal_year(l2015, jr, year))
  }
}
```

We can see that we can get both positive or negative tau(s):

```
results
```

```
##          Journal Year           tau     p.value
## 1         Nature 2007  0.0115327643 0.695512738
## 2        Science 2007 -0.0232164706 0.443764950
## 3    The Lancet 2007            NA          NA
## 4  New Eng J Med 2007  0.0865486187 0.066693254
## 5         Nature 2008 -0.0804201182 0.004800352
## 6        Science 2008  0.0067644073 0.824789544
## 7    The Lancet 2008  0.1499256823 0.007689070
## 8  New Eng J Med 2008  0.1170901670 0.012675009
## 9         Nature 2009 -0.0104426422 0.723168401
## 10       Science 2009 -0.0218930503 0.462036960
## 11   The Lancet 2009  0.0592239256 0.305138499
## 12 New Eng J Med 2009  0.1276904509 0.006284098
## 13        Nature 2010 -0.0383378489 0.174662196
## 14       Science 2010 -0.0311213024 0.314116809
## 15   The Lancet 2010  0.1210024231 0.029256634
## 16 New Eng J Med 2010  0.0849086854 0.069678724
## 17        Nature 2011 -0.0279099578 0.323149648
## 18       Science 2011  0.0171414424 0.579261893
## 19   The Lancet 2011  0.0648182972 0.209265286
## 20 New Eng J Med 2011 -0.0217783333 0.646159372
## 21        Nature 2012  0.0159125585 0.571748517
## 22       Science 2012  0.0211241243 0.508981567
## 23   The Lancet 2012  0.1024408564 0.066156323
## 24 New Eng J Med 2012  0.1454023269 0.002005165
## 25        Nature 2013 -0.0344594481 0.242130179
## 26       Science 2013  0.0257910388 0.419193129
## 27   The Lancet 2013  0.1396061773 0.016167487
## 28 New Eng J Med 2013  0.0002704937 0.995615193
```

However, to be sure we have a meaningful result, we should correct for multiple testing (when trying very many tests, we can obtain a number of significant p-values just by chance), either applying Bonferroni's correction, or using more sophisticated false-discovery-rate approaches.