STATS 782, 2017 **Assignment 2** **Due: 1pm Tue 29 Aug**

Note:

- Refer to the notes at the beginning of Assignment 1.
- For all questions in Assignment 2, you should avoid using explicit loops as much as you can. If you are unsure of your choices in regard to computational efficiency, use `system.time` to compare run times.

1. **[10 marks]** Write a recursive function that returns the Fibonacci sequence for a given number of terms (check Lab 3 for definition of the Fibonacci sequence). Demonstrate your code for the following cases:

```
> fib(1)
[1] 0
> fib(2)
[1] 0 1
> fib(3)
[1] 0 1 1
> fib(10)
 [1]  0  1  1  2  3  5  8 13 21 34
```

2. **[10 marks]** The k-means algorithm is widely used in cluster analysis for its simplicity. Since sample means can be severely affected by outliers, one may want to replace sample means with sample medians as cluster centers and thus obtain the following k-means algorithm based on medians for univariate data (perhaps we can call it the "k-medians algorithm"):

   > Given k initial cluster centers $c_1, \ldots, c_k$ for a sample $x_1, \ldots, x_n$, repeat the following two steps until cluster centers do not change:
   > (1) Calculate $d_{ij} = |x_i - c_j|$ for $i = 1, \ldots, n$ and $j = 1, \ldots, k$. Classify x_i into cluster m, if d_{im} is the smallest of $d_{i1}, \ldots, d_{ik}$.
   > (2) For $j = 1, \ldots, k$, set $c_j = \text{median}(x_i)$ for all x_i in cluster j.

   Implement the "k-medians algorithm" in an R function which, given a univariate sample and an initial set of cluster centers (both in vectors), computes and returns the final cluster centers. Apply it to the variable `eruptions` (for eruption times of the well-known Old Faithful Geyser) in the data set `faithful` in R, using the following initial cluster centers, respectively:

   (a) $(2, 4)^\top$;
   (b) $(2, 3, 4)^\top$;
   (c) $(2, 3, 4, 5)^\top$.

3. **[10 marks]** Let $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ be a random sample of two random variables (X, Y). Any pair of observations (x_i, y_i) and (x_j, y_j), where $i \neq j$, are said to be concordant if either both $x_i > x_j$ and $y_i > y_j$, or both $x_i < x_j$ and $y_i < y_j$. Denote by $C(x, y)$ the proportion of concordant cases out of all cases that are not ties, where a tie is either $x_i = x_j$ or $y_i = y_j$. Write a function named `conc` that, given vectors `x` and `y`, computes and returns the value of C. For example,

```
> conc(x=1:5, y=c(3,1,4,5,2))
[1] 0.6
```

Demonstrate that your function works for the above example, and apply it to the two vectors generated as follows:

```
> set.seed(782); x = round(rnorm(1000)); y = x + round(rnorm(1000))
```

4. **[20 marks]** The Bradley and Terry model is popularly used in sports to numerically rank teams playing in a league. A rank parameter $r_i > 0$ is used for team i, in the sense that team i beats team j with probability $r_i/(r_i + r_j)$, where ties are assumed impossible. Denote by y_{ij} the times that team i beats team j during a season. Assuming that all games are independent, the probability of the whole season is

$$L(r) = \prod_{i,j} \left(\frac{r_i}{r_i + r_j} \right)^{y_{ij}},$$

where $r = (r_1, \ldots, r_k)^\top$. To rank the teams, we need to find the vector $\hat{r}$ that maximizes $\log L(r)$. Since $\log L(\alpha r) = \log L(r)$ for any $\alpha \neq 0$, we have to constrain r, for example, by enforcing $\sum_i r_i = s$, where s is a fixed number and should better be positive.

Find the file `NBA2016-2017.csv` on Canvas that stores the game results between each pair of 30 NBA teams during the 2016–2017 regular season. The data in the file looks like

```
Team 1,Team 2,Wins
Atlanta Hawks,Boston Celtics,2
Atlanta Hawks,Brooklyn Nets,2
## <many more rows>
```

where `Wins` is the times `Team 1` beats `Team 2`. Download the data file and read in the data properly.

(a) [10 marks] Use the `optim()` function to find the maximum likelihood estimate $\hat{r}$, with $s = 1000$. You should use the BFGS (or L-BFGS-B) method, without providing the derivatives of the log-likelihood function. Scale the values of all $\hat{r}_i$ so that the top-ranking team has $\hat{r}_{[1]} = 100$. Show all team names, along with their $\hat{r}_i$ values sorted in descending order.

(b) [5 marks] Re-do part (a), with the derivatives of the log-likelihood function provided to `optim()`.

(c) [5 marks] Produce a contour plot that shows the profile log-likelihood of $r_{[1]}$ and $r_{[2]}$ for the two teams with the highest rankings, with the other r_i fixed at their maximum likelihood estimates.