University of Manitoba
Department of Statistics

STAT 4600: Computational Statistics

Assignment 2

Due: Thursday, January 25

Question 1:

Let f be a function defined on some interval of the real line. It is often of interest to identify the zeroes of f, also referred to as the *roots* of the equation

$$f(x) = 0.$$

Newton's method of root finding is quite popular for this and works quite well in many problems.

Specifically, starting with an initial guess x_0 to a root of f, one sequentially calculates the improved guesses

$$x_i = x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})}, \qquad \text{for } i = 1, 2, \ldots \tag{1}$$

until convergence, i.e. until $f(x_i)$ is considered to be close enough to zero. Usually, this implies iterating (1) until $|f(x_i)| < \varepsilon$ for some preselected tolerance level $\varepsilon > 0$.

(A) Using R, plot the function

$$f(x) = \cos(\pi x) + e^{x/2},$$

from $x = -10$ to $x = 2$. To that graph, superimpose the curve of $\cos(\pi x)$. Using a different colour and/or type of line would be a good idea. From this, what can you conclude about the approximate zeroes of f?

Hint: The function `curve()` could be used here. The option `add=TRUE` allows to add a curve to an existing graph.

(B) Write an R function that performs the Newton iteration from the following arguments:

- the function `f` whose approximate root is desired,
- an initial guess `x.0`,
- the prescribed tolerance level `tol.level`,
- a `max.iter` argument (with default value of 100) that is used to stop the updating of the approximate root even if convergence is not reached and that many iterations have been performed, returning a message to that effect.

Your function should return a list containing the following three elements:

- the approximate root,
- the value of f at the approximate root,
- the number of iterations that were performed.

(C) Use your function to approximate the two largest roots of f with a tolerance level of $\varepsilon = 10^{-7}$.

Hint: Compare the results of your function with those given by the function `uniroot()`.

Question 2:

From the UMLearn page of the course, download the file `EPL_1617.csv`, which contains details on each game played in the 2016/17 season of the English Premier League of soccer, including the date, home team, away team and number of goals scored by each team.

Construct an R function named `season.summary` that produces, given the outcomes to all the games of a soccer season, a table as the one shown in Figure 1 below. The returned table should be an integer matrix with appropriate column names and row names corresponding to the team names for improved readability.

Note that the number of points accumulated by each team are obtained using the following very simple rule:

- 3 points for each win,
- 1 point for each tie,
- no points for losses.

Note also that teams are ordered

- first according to total points,
- then, in case of a tie, in terms of goal difference,
- finally, if two teams are still tied, in terms of the number of wins.

This is important as the ordering of some of the teams would be wrong otherwise in Figure 1.

It is desired that the function be fully adaptive to the provided results, i.e., that no information outside of the file that is read be necessary to produce the wanted table. Essentially, the function should work with similar data coming from any sports league with different teams, number of games between teams, etc. More details will be given in class.

Note: Some R functions that will be useful here are

- `read.csv()`, to load CSV files in R,
- `unique()`, to identify the teams that are involved,
- `order()`, to help re-order the table according to decreasing point totals.

Figure 1: Season summary that should be produced for Question 2

	W	T	L	Pts	GF	GA	GD
Chelsea	30	3	5	93	85	33	52
Tottenham Hotspur	26	8	4	86	86	26	60
Manchester City	23	9	6	78	80	39	41
Liverpool	22	10	6	76	78	42	36
Arsenal	23	6	9	75	77	44	33
Manchester United	18	15	5	69	54	29	25
Everton	17	10	11	61	62	44	18
Southampton	12	10	16	46	41	48	-7
AFC Bournemouth	12	10	16	46	55	67	-12
West Bromwich Albion	12	9	17	45	43	51	-8
West Ham United	12	9	17	45	47	64	-17
Leicester City	12	8	18	44	48	63	-15
Stoke City	11	11	16	44	41	56	-15
Crystal Palace	12	5	21	41	50	63	-13
Swansea City	12	5	21	41	45	70	-25
Burnley	11	7	20	40	39	55	-16
Watford	11	7	20	40	40	68	-28
Hull City	9	7	22	34	37	80	-43
Middlesbrough	5	13	20	28	27	53	-26
Sunderland	6	6	26	24	29	69	-40