# Business Analytics – Homework 4

## Due 9 am, Friday October 4th

This assignment consists of 4 exercises the last two **being completely optional**. It has been designed to make sure you master manipulation of dates, strings and data frames in R. As always, you will be asked to write short code snippets or code-chunks in R. You will be graded both on your code, and the written answers you provide. When evaluating the code, the grader will take on the role of a co-worker. Code will be evaluated both in terms of how correct and how clear it is. By correctness, I mean that the code fulfills the requirements of the question. By clarity, I mean that the grader should be able to understand what your code does within 30 seconds of reading it. As discussed in class, this is aided by clear comments, good variable names, proper indentation, and short lines.

# Problem 1: Date Manipulation

1. Program the function `Is_leap_Year` which takes a 4 digit year between 1950 and 2050 as input and returns TRUE only if it is a leap year and FALSE otherwise.

```
Is_leap_Year = function(Year){
            code here …

      }
```

2. Program the function `second_Saturday` that displays a list of the dates for the 2nd Saturday of every month for a given year. It does not need to return anything.

```
second_Saturday = function(Year){
            code here …

      }
```

3. Write a program that takes a date as input and one integer between -20000 and 20000. The function should return the day of the week of adding the integer number of days to the initial date. That is, new date = the input date +/- the number of days (the integer between -20000 and 2000) and then compute the day of the week of the new date.

```
day_Of_the_Week_in_N_days = function(date,N){
            code here …

}
```

```
solution = day_Of_the_Week_in_N_days(date = '2019-09-22',N=2);
print(solution);
```

Tuesday

# Problem 2: Analyzing Flight Delays

This exercise is intended to show you the advantages of working with data.tables instead of data frames. Please try to do each task using **both** using data.table and using standard data frames. Report the computation times. You can control the time using `proc.time()` or `Sys.time()`

1. Load the data frame flights.csv from LATE
    a. R data frames: use `read.csv()`
    b. data.tables use `fread()`

2. Order the rows of the data frame by Airline, Flight Number, Year, Monthly and day.

3. Conditional on both Airline and hour of the day (the hour of the day of 09:26 is 09), compute:

    a. Total number of flights
    b. Number of flights with a delay of more than 15 minutes
    c. Number of flights that depart with less 15 minutes delay

    d. Does the distribution of the % delayed flights conditional on time of the day and firm vary by firm? No statistical test is needed just visual analysis or simply report if they share similar same means, standard deviation for the statistic.

4. Compute the market share of number of flights that each firm has over:

    a. The entire year
    b. Month by month. Does the market share distribution across airlines vary by month? If so, can you give potential explanation?

    Report code, output table in exported in excel, one visualizations and 3-4 sentences explaining the results

## Problem 3: Movies (Painful but not Difficult)

Sometimes, as data scientists, we need to use scrape data from other websites. It is very common that the data is scraped with Python and analyzed in R. The problem is that scrapped data is never as tidy as when you pull it directly from a database.

If you can make this exercise you are good enough to do RA work for any empirical researcher here at Brandeis. Further explanation will be given in class.

In this exercise you are given the excel spreadsheet `movies_at_imdb.csv`.
It contains all the movies from 2000 to today. The data is not clean. We say that the data is dirty when, by columns the format varies by row.

Clean the data frame as follows.

- `Actors`: To be replaced with n_Actors (should be numeric) and the column should be transformed into a table that has as index both the actor and the imdb.. Name such table actors_movies.

- `Awards:` column should be removed and transformed into another table

- `BoxOffice:` Numeric, which means that summing it with NA.rm has to work

Do the same for:

- `Country: to be replaced with N_countries and transformed into a another table`
- `DVD:`
- `Director:`
- `Episode:`
- `Genre:`
- `Language:`
- `Metascore:`
- `Plot:`
- `Poster:`
- `Production:`
- `Rated:`
- `Ratings:`
- `Released:`
- `Response:`
- `Runtime: Column to be removed, no variation`
- `Season:`

- `Title:`
- `Type:`
- `Website:`
- `Writer:`
- `ImdbRating:`
- `Year:`
- `Imdb:`

Once the data is clean document 4 facts that you consider interesting

## Problem 4: Xavi and 宋扬威 in a plane. Very Difficult (don't bother too much)

This problem is difficult so I would not worry too much. It happened to me once and I wasted a lot of time solving it analytically instead of being smart and asking a computer to do it for me.

Yangwei and I were traveling from Shenyang to Germany once. As it is always the case in China, there was a lot of people, the plane was full.

I was the first in the line but Yangwei, as always, went to the washroom and stayed there for ever so she ended being the last one. Not only I was the first one and she was the last one but, on the top of it, when I was about to enter the plane I realized I had lost my boarding pass. I told the flight attendance and she told me to wait until the last passenger had to step in. I was surprised, I ask why? She answered that otherwise, if I were to sit in somebody else's sit then he or she may similarly chose another passenger's place and because there are a lot of people, nobody would end up sitting on his rightful place.

What made me super super happy and excited is that the last sentence she mentioned to me (in a kind of arrogant way) was… how come you don't see that the probability the last passenger gets his/her spot is zero… Cool right? A flight attendant lecturing me about probability….

*OK, so let's see if she is super super good in probability.*

Suppose there are 100 passengers in the line to enter a plane (as it was my case) and the first one loses his boarding card (as I did) but he is allowed to enter the airplane (not as they did with me). Passengers enter one by one. The first passenger (i.e. the one who has chosen the boarding car) chooses a sit at random, the second enters the plane and, if his spot is still free, she takes, but if it has been taken, let's assume he chooses another one completely randomly. Suppose this continues in this fashion until the Yangwei enters.

Can you tell me, what is the probability that when Yangwei reaches her sit, it has not been taken? Use a simulating algorithm as we did with the poker and do it for imaginary airplane sizes of 100, 150, 200,…, 500. Make a barplot.