CONTENT

1. EXECUTIVE SUMMARY

The purpose of this data manipulating challenge is to be familiar with the data structure of Amadeus (bookings and searches) and to get experience in term of processing large data sets. Through this challenge, there are some important drawing out learning experience as follow:

- The data project can be divide into 3 stages: Data Exploration, Data Cleansing and Data Analyzing. In which, Data Cleansing is the most time consuming stage that can take 80-90% the duration of the whole project;
- Always begin working with a small subset of data sets to get the idea about the structure of data and the possible errors that need to be cleaned;
- With large data sets, dividing the data into smaller parts (blocks) and processing data partially is a very efficiently approach which will reduce the memory shortage and help detect quickly processing errors before taking lots of time running through the whole data;
- Processing large data sets is huge time consuming, therefore, always do (1) back up and save the data sets before and after a big processing step and (2) only import the exact columns that are needed to the data analyzing;
- In term of programming languages, R is strong in statistics and model building, but Python is much more powerful in data manipulating and handling large data sets. The details will be explained in this report.

2. ANSWERS OF 5 QUESTIONS

Question 1: Count the number of lines in each files

bookings.csv

- 10,000,010 lines of data (NOT include the header line)
- 38 variables, 37 separators per line (i.e. "^")

searches.csv

- 20,390,198 lines of data (NOT include the header line)
- 45 variables, 44 separators per line (i.e. "^")

Question 2: Top 10 arrival airports in the world in 2013

IATA Code	Total Arrival	City	Airport Name
LHR	88,809	London	Heathrow
MCO	70,930	Orlando	International
LAX	70,530	Los Angeles	International/ Metropolitan Area
LAS	69,630	Las Vegas	McCarran International/Metropolitan Area
JFK	66,270	New York	John F Kennedy Intl
CDG	64,490	Paris	Charles de Gaulle
BKK	59,460	Bangkok	Suvarnabhumi Int'l/ Metropolitan Area
MIA	58,150	Miami	International/ Metropolitan Area
SFO	58,000	San Francisco	International
DXB	55,590	Dubai	International/ Metropolitan Areat

Source: IATA codes: http://www.iata.org/publications/Pages/code-search.aspx

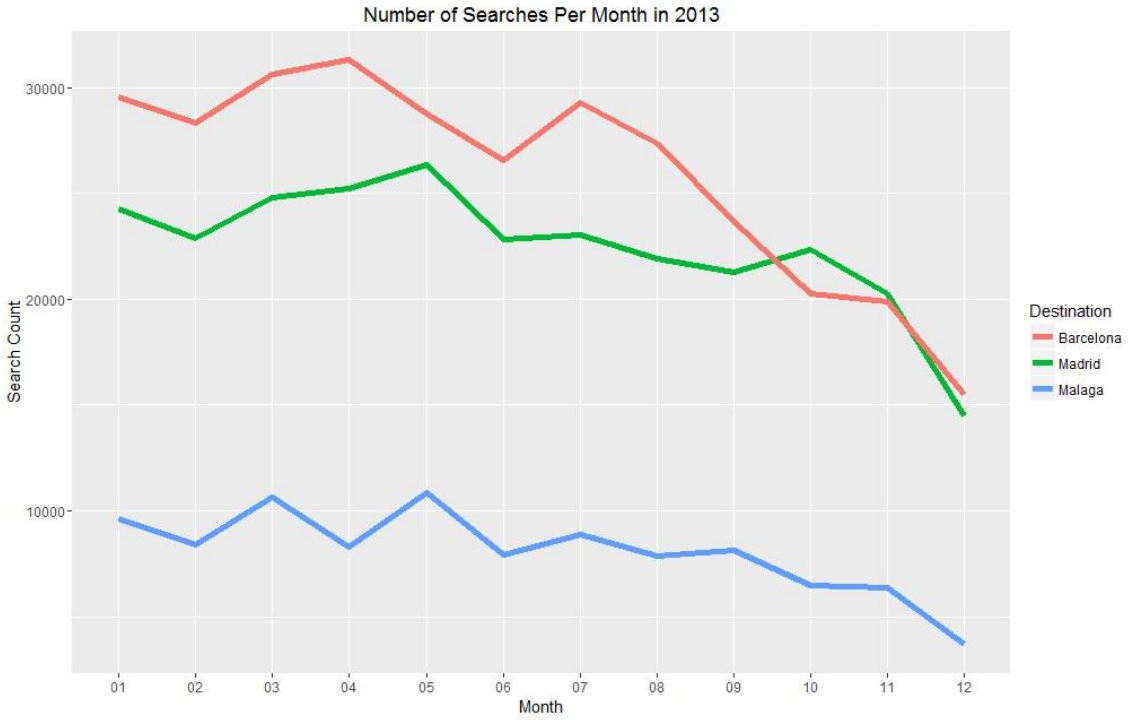
Question 3: Plot monthly number of searches for flights arriving at Malaga, Madrid or Barcelona

Location names and airport IATA codes:

- Malaga: AGP
- Madrid: MAD, CLQ, TOJ
- Barcelona: BCN, BLA

People tend to look for their trips in the first two quarters more than in the end of the year (2013).

Month	Malaga	Madrid	Barcelona
1	9,633	24,258	29,526
2	8,379	22,857	28,329
3	10,659	24,795	30,609
4	8,265	25,251	31,293
5	10,830	26,334	28,728
6	7,923	22,800	26,562
7	8,892	23,028	29,298
8	7,866	21,888	27,360
9	8,151	21,261	23,655
10	6,499	22,351	20,276
11	6,384	20,272	19,880
12	3,696	14,504	15,512



Bonus question 1: Match searches with bookings

To match 20,390,198 searches with 10,000,010 bookings, we temporary assume these following matching criteria:

- The departure of the search matches with the booking departure
  [searches] Origin = [bookings] dep_port
- The arrival of the search matches with the booking arrival
  [searches] Destination = [bookings] arr_port
- The date doing search is the same date making booking
  [searches] Date = [bookings] act_date

To improve the matching results, we can add more information about the searching place and the booking location, e.g. [bookings] pos_ctry = [searches] Country.

Number of searches match with bookings	996,008	4.9%
Number of searches NO match with bookings	19,394,190	95.1%
Total searches	**20,390,198**	100%

Bonus question 2: Write a Web Service (Extract to JSON)

Check the enclosed JSON file.

3. PROCESSES OF THE PROJECT

Stage 1: Data Exploration

- Purpose: working with a small subset of data (n = 100 rows) to be acquainted with the structure of the data, type and meanings of each columns (fields), potential errors that need to be cleaned, etc.
- Package using: ff (reading large data set)
- Time consuming: 1-2 hour(s), 5% total time
- Main challenge: unfamiliar with the data sets

Stage 2: Data Cleansing

- Purpose: mandatory step, detect and clean all the data errors, create a clean data set before doing data analyzing.
- Package using: ff (reading large data set), stringr (text cleansing)
- Time consuming: 16-20 hours, 85% total time
- Main challenge:
    - Selecting the right package to use: ff, bigmemory, sqldf.
    - RAM and hard disk shortage.
    - Huge time consuming and computer crash.

Stage 3: Data Analyzing

- Purpose: analyze data to answer business question
- Package using: sqldf and ff (reading large data set), ggplot2 (visualize results), ffbase (analyzing data)
- Time consuming: 2-4 hours, 10% total time
- Main challenge: R doesn't have to complete and stable package for handling large data set, have to using multiple packages for different purpose.

R packages for large data sets comparison: there are 4 most famous packages

Package	Advantage	Disadvantage	Usage
ff	• Read data very fast • Support many types of data, including string • Calculate function run very fast	• Convert string to factor, and store all factor levels in RAM → out of memory for this kind of data sets • No option to select specific columns when import large data sets → have to import all data set	• Data exploration • Analyzing a subset of data with less columns • Using to answer bonus question 1
bigmemory	• "Big Family" contains many powerful	• No support string (text)	• N/A

	packages for many purposes • Recent update on 28/03/2016		
sqldf	• Support almost all data type, including character • Store every data in hard drive, no RAM shortage • Can use SQL syntax to select specific columns to import → reduce data to read	• Calculating functions run very slow, SQL queries can take forever	• Support many types of data, including string • Using to answer simple questions, i.e. question 1, 2, 3
data.table	• Very powerful package, read data very fast • Support select specific columns to import • Support multiple separators (e.g. "^" and ",") • Very handy and powerful calculation	• No support data set larger than 1.2GB	• Data exploration

4. APPENDIXES

- The ff package: Handling Large Data Sets in R with Memory Mapped Pages of Binary Flat Files, 2007, D. Adler, O. Nenadić, W. Zucchini, C. Gläser, Source: https://www.r-project.org/conferences/useR-2007/program/presentations/adler.pdf
- Managing large datasets in R – ff examples and concepts, 2010, Jens Oehlschlägel, Source: http://ff.r-forge.r-project.org/bit&ff2.1-2_WU_Vienna2010.pdf
- Taking R to the Limit, Part II: Working with Large Data sets, 2010, Ryan R. Rosario, Source: http://www.bytemining.com/wp-content/uploads/2010/07/r_hpc.pdf
- An introduction to data cleaning with R, 2013, Edwin de Jonge, Mark van der Loo, Source: https://cran.r-project.org/doc/contrib/de_Jonge+van_der_Loo-Introduction_to_data_cleaning_with_R.pdf
- A data.table R tutorial by DataCamp, 2015, DataCamp, Source: https://www.datacamp.com/community/tutorials/data-table-r-tutorial
- CRAN Task View: High-Performance and Parallel Computing with R, 2016, Dirk Eddelbuettel, Source: https://cran.r-project.org/web/views/HighPerformanceComputing.html
- R Data Imp ort/Export, 2016, R Core Team, Source: https://cran.r-project.org/doc/manuals/r-release/R-data.html
- The R Package bigmemory: Supporting Efficient Computation and Concurrent Programming with Large Data Sets, John W. Emerson, Michael J. Kane, Source: http://www.stat.yale.edu/~mjk56/temp/bigmemory-vignette.pdf