We use data.table on a daily basis. So most of the questions here are to be programed using data.table. For each of the questions, I expect

1. A short paragraph on what is being asked;
2. R code to carry out the task(s);
3. A segment of some output that can demonstrate the results;
4. Anything you care to share about what you have learned from the task;

**Question 1:**

1. Simulate two data sets:
   1. Data set called *Demo* with three variables: *id, birth\_dt, gender, county*. *id* variable uniquely identifies rows.
   2. The *birth\_dt* will range from ’01-01-1940’ to ’12-31-2021’.
   3. The county should have Los Angeles county and other counties in California.
   4. Data set called *Medical\_visit* of medical records with these variables: *id, service\_date, service\_code, service\_num*. The variables *id, service\_date* and *service\_num* uniquely identify rows.
   5. Make sure that 20% of unique *id*s in the Medical\_visit data set have multiple records.
   6. Make sure that these two data sets share at least 90% of the values of the id variable.
   7. The *service\_date* will range from ‘01-01-2018’ to ‘12-31-2019’.
2. Subsetting *Medical\_visit* to only female patients from Los Angeles county in the Demo data.
3. Subsetting *Medical\_visit* to those where the *service\_date* is prior to ’05-14-2018’.
4. Add a variable called *max\_svc\_date* to the *Medical\_visit* data set which is the *service\_date* when a patient had the maximal *service\_num* among all the medical records. For example, here are all medical records for a patient:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | service\_date | service\_code | service\_num | max\_svc\_date |
| 1 | 01-10-2018 | … | 9 | 03-20-2018 |
| 1 | 03-20-2018 | … | 20 | 03-20-2018 |
| 1 | 04-25-2018 | … | 3 | 03-20-2018 |
| 1 | 02-17-2019 | … | 15 | 03-20-2018 |
| 1 | 10-25-2019 | … | 19 | 03-20-2018 |

Randomly select a *service\_date* if there is a tie on the *service\_num*.

1. Assume that there is a sequence of data tuples (demo\_1, medical\_visit\_1), (demo\_2, medical\_visit\_2), …, (demo\_10, medical\_visit\_10), write a loop that can does part b) to each tuple, and at the end combine all the results into one data set.
2. Add an integer variable called *age\_by\_svc* to the *Medical\_visit* data set which represents a patient’s age by the *service\_date*. This variable can have missing values if no demographic information is available for a patient.
3. Please create a dataset called *Medical\_visit2018*:
   1. *Medical\_visit2018* subsets *Medical\_visit* data with added *age\_by\_svc* variable from part e) to those have county information in the Demo data and have *service\_date* in 2018.
   2. Create an *age\_by\_svc* by county distribution data called *Age\_distri* based on *Medical\_visit2018*. Please include Min, Max, Quartiles, Mean in the distribution and format the *Age\_distri* like the following:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| County | Min | 25% | Median | 75% | Max | Mean |
| Los Angeles |  |  |  |  |  |  |
| San Francisco |  |  |  |  |  |  |

**Question 2:**

Data set called *Claims* with several variables.

* 1. Finding all the variables ending with *“\_date”* and change the type of these variables to date. (they are string variables before like “02-18-2019”)
  2. There is a variable *NDC* in the data set which we stores as an integer to save memory. The *NDC* variable ranges from 2120030 to 8132362108. Please turns this variable into a character variable and add leading 0s so that this character variable has length 10 all the time.
  3. There are 3 variables: *id, birth\_date* and *service\_date* uniquely identify the row. For same id there should be only *one birth\_date,* but there are some ids having multiple *birth\_date.*

Please modify the *birth\_date* and keep the most frequent *birth\_date* as correct *birth date* for same id.

* 1. Data called *Enrollment* with these variables: *id, gender, enroll\_date*. An *id* can have multiple *enroll\_date.*

Please add the *Enrollment* data to the *Claims* data and meet the following requirements:

* + 1. Keeping only the *id* in the *Claims* data.
    2. Keeping only the earliest *enroll\_date* when adding to the *Claims* data.
  1. Data called *Monthly\_Enrollment* in wide format with these variables: id, *`07/01/2018`, `08/01/2018`, `09/01/2018`, `10/01/2018`, `11/01/2018`, `12/01/2018`, `01/01/2019`, `02/01/2019`, `03/01/2019`, `04/01/2019`*. The variables *`07/01/2018`~`04/01/2019`* are 1/0 variables indicating whether the id was enrolled during the month. Each *id* has one unique row of record.

Please turn this wide data into long format with variables: *id, year, month, enr\_length* and meet the following requirements:

* + 1. For each *id*, only year and month records during which the id was enrolled should be kept in long format.
    2. *enr\_length* is the number of months the id was enrolled. Please use at least two different methods to create *enr\_length*.

For example,

Wide format:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | `07/01/2018` | `08/01/2018` | `09/01/2018` | `10/01/2018` | `11/01/2018` | `12/01/2018` | `01/01/2019` | `02/01/2019` | `03/01/2019` | `04/01/2019` |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |

Long format:

|  |  |  |  |
| --- | --- | --- | --- |
| id | year | month | enr\_length |
| 1 | 2018 | 7 | 7 |
| 1 | 2018 | 8 | 7 |
| 1 | 2018 | 9 | 7 |
| 1 | 2018 | 12 | 7 |
| 1 | 2019 | 1 | 7 |
| 1 | 2019 | 2 | 7 |
| 1 | 2019 | 3 | 7 |

* 1. In the *Claims* data we havefour variables *code1, code2, code3, code4.* These four variables are character variables containing only digits or/and alphabets (case sensitive) with different lengths.
     1. Please turn all the lowercase alphabets appeared in these four variables into uppercase.
     2. Please extract out all the unique codes appeared in *code1* that begins with “A0”.
     3. Please create a 1/0 variable *Flag* indicating whether “1234” or “4321” appears in any of the *code1-4.*
  2. Aggregating the *Flag* above by *id, year, month* level. (creating *year, month based on service\_date variable)*

The data will look like:

|  |  |  |  |
| --- | --- | --- | --- |
| id | year | month | Flag |
| 1 | 2019 | 1 | 1 |
| 1 | 2019 | 2 | 0 |
| 2 | 2019 | 1 | 0 |

* 1. After getting monthly data above:
     1. Create a variable *Flag2* indicating whether the id have *Flag* equal to 1 in the current month or prior months.
     2. Create an integer variable *Group* to index the situation where *Flag*=1 for each id. The *Group* variable begins with 0 for each id and starts to increment every time *Flag*=1.

The data will look like:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id | year | month | Flag | Flag2 | Group |
| 1 | 2019 | 1 | 1 | 1 | 1 |
| 1 | 2019 | 2 | 0 | 1 | 1 |
| 1 | 2019 | 4 | 0 | 1 | 1 |
| 1 | 2019 | 5 | 1 | 1 | 2 |
| 1 | 2019 | 6 | 0 | 1 | 2 |
| 1 | 2019 | 8 | 1 | 1 | 3 |
| 2 | 2019 | 1 | 0 | 0 | 0 |
| 2 | 2019 | 3 | 1 | 1 | 1 |
| 2 | 2019 | 7 | 0 | 1 | 1 |
| 3 | 2019 | 10 | 0 | 0 | 0 |
| 3 | 2019 | 11 | 0 | 0 | 0 |

* 1. There is a variable called *modifier1* which is a single uppercase letter or empty or missing. Please create a 1/0 variable *Flag3* indicating whether code1 has the format “A0(…)9” and modifier1 has the value “U” at the same time. The (…) part of the format “A0(…)9” has be any lengths of digits or alphabets, i.e., A0B9, A0879, A0DDD9, etc.

The data will look like:

|  |  |  |
| --- | --- | --- |
| code1 | modifier1 | Flag3 |
| A0B9 | T | 0 |
| A0B89 | U | 1 |
| 78950 | U | 0 |
| BB080 |  | 0 |