QcDM Project User Guide

Chen Ying[1]

Ning Yilin[2,3]

Maudrene Tan[4]

Tan Chuen Seng[1]

1: Saw Swee Hock School of Public Health, National University of Singapore (NUS) and National University Health System (NUHS); 2: NUS Graduate School for Integrative Sciences and Engineering, NUS; 3: Department of Surgery, Yong Loo Lin School of Medicine, NUS and NUHS; 4: Department of Medicine, National University Hospital and NUHS.

Table of Contents

1. Overview and Installation

The QcDM (**Q**uality **c**are for **D**iabetes **M**ellitus) Project provides R-Shiny applications for generating glucometrics measures from point-of-care blood glucose data at three different units of analysis: patient-sample, patient-day, and patient-stay. The application facilitates convenient and flexible generation of glucometrics measures by providing user-friendly graphical user interfaces (GUIs) that runs in a browser on the user's desktop.

The QcDM Project consists of two R-Shiny GUIs and an R package. The first GUI, QcDMconverter, processes raw glucose data files to conform to the data format required by the application. The second GUI, QcDMui, is the main tool of the QcDM Project that uses functions in the QcDM package (https://github.com/nyilin/QcDM) to analyse processed glucose data to generate glucometrics measures.

The application operates on Windows and Mac operating systems, and is free under an academic non-commercial license. The current version of this application can be downloaded from a GitHub repository (https://github.com/nyilin/QcDM_Project). Installation of this application is described in details in the following sections.

1.1. Install R and RStudio

The QcDM Project runs on R version 3.6.0 or later, but users should avoid version 4.0.0 because it has been reported to have issues with Shiny applications. Users are recommended to install the latest version of R and RStudio by following the instructions below.

Install R from the installer downloadable from the official website of the R Project.

- For Windows: download the Setup Wizard from https://cran.r-project.org/bin/windows/base/. Follow through the installation steps and keep the default options.
- For macOS: download the installer (a pkg file) from https://cran.r-project.org/bin/macosx/. Follow through the installation steps and keep the default options.

Install RStudio Desktop from the installer downloadable from the official website, https://rstudio.com/products/rstudio/download/#download.

For macOS, an additional application, XQuartz, is required, which can be downloaded from the official website: https://www.xquartz.org.

1.2. Download QcDM Project

From the [GitHub repository for the QcDM Project](https://github.com/nyilin/QcDM_Project) (https://github.com/nyilin/QcDM_Project), download the project as a zip file, named 'QcDM_Project-main.zip'. **We recommend using Chrome to access the link and download the zip file:**
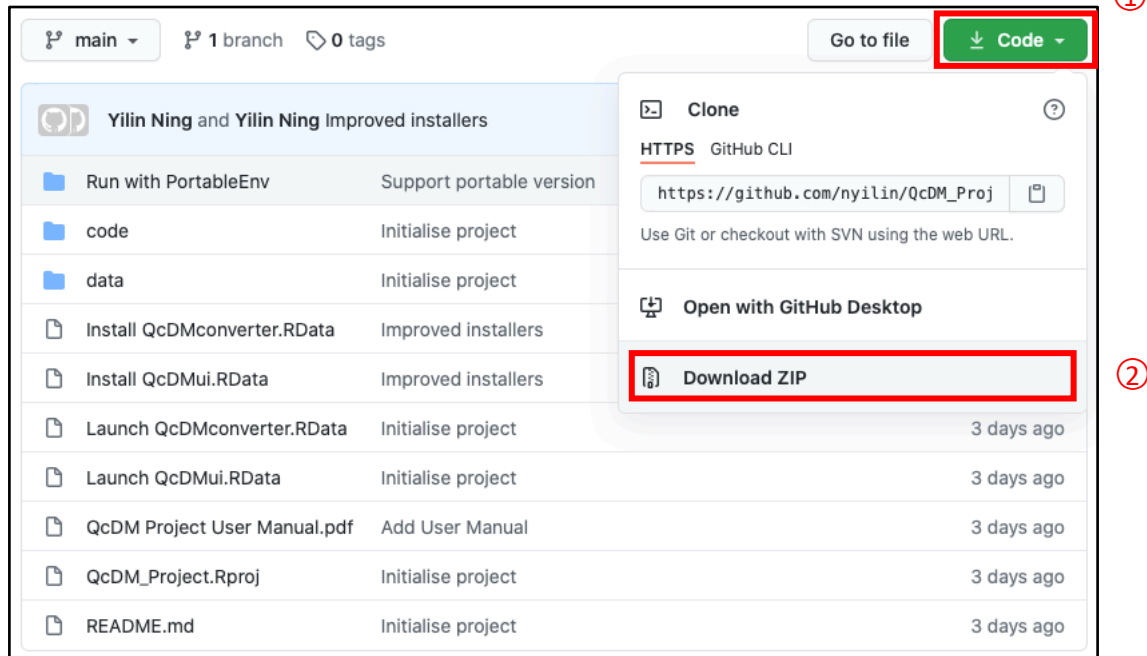
For Windows, unzip the file by right-clicking on 'QcDM_Project-main.zip' and selecting the following (and for illustration purposes we assume 7-Zip is used but any software that unzips file can be used):
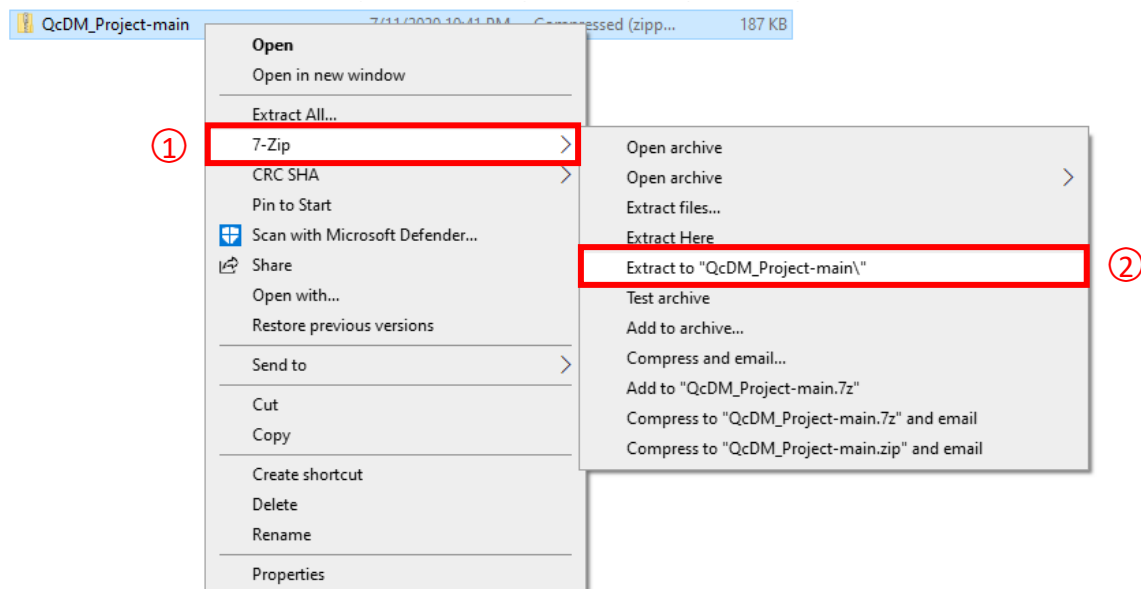
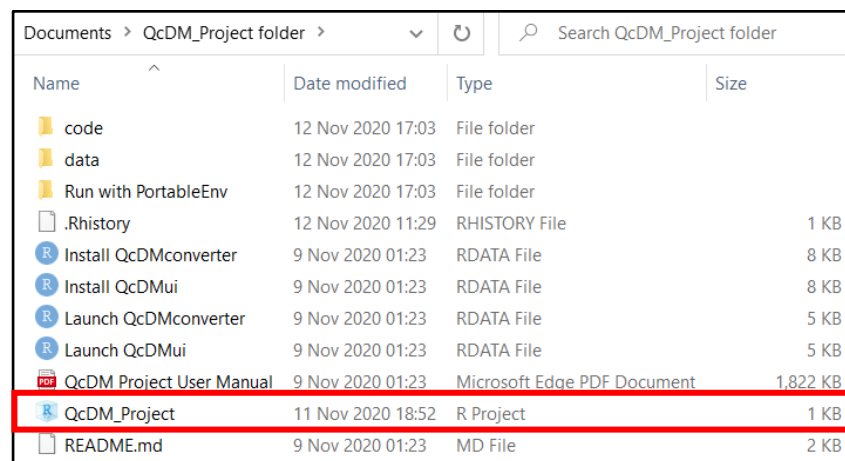For macOS, unzip the file by doubling clicking on it.

We recommend users to rename the folder as 'QcDM_Project folder' and move this folder to the 'Documents' folder. In the rest of this document, we assume users are using the

Windows operating system and have followed the recommendations to rename and move the QcDM Project folder.

1.3. Install required R packages

A few R packages are required to use the two GUIs in the QcDM Project, and they can be installed by using the RData files in the 'QcDM_Project folder' following the instructions below.
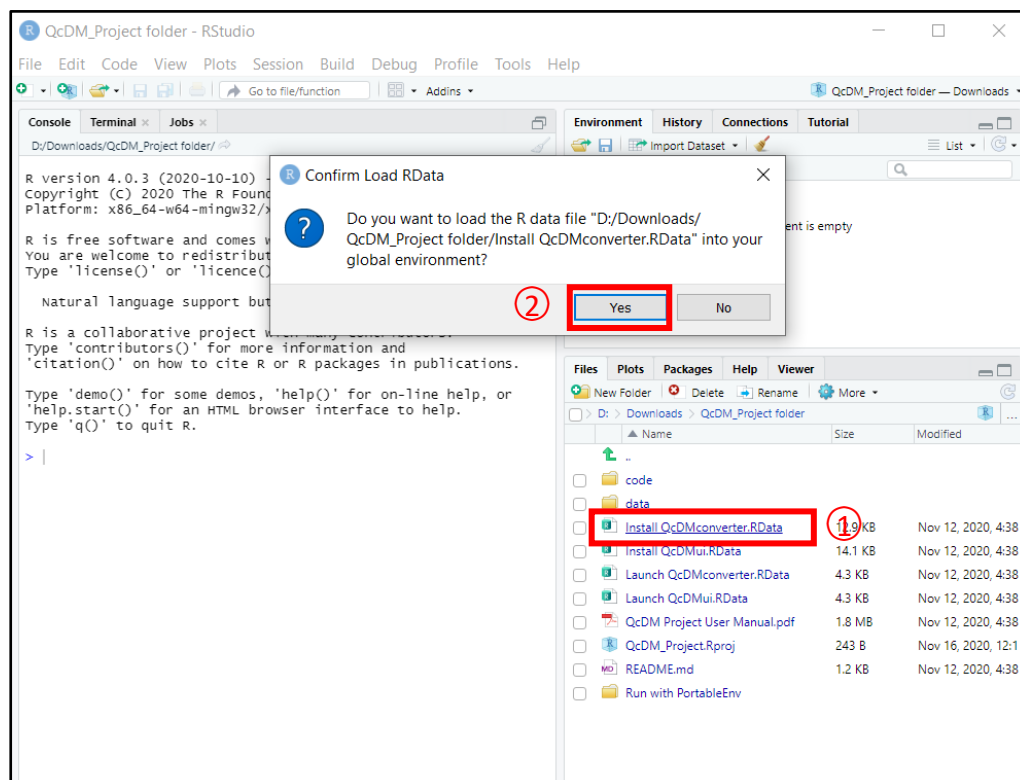
The screenshot below shows all the subfolders and files contained in the 'QcDM_Project folder'. Firstly, users need to open RStudio by doubling-clicking 'QcDM_Project.Rproj' in the 'QcDM_Project folder':
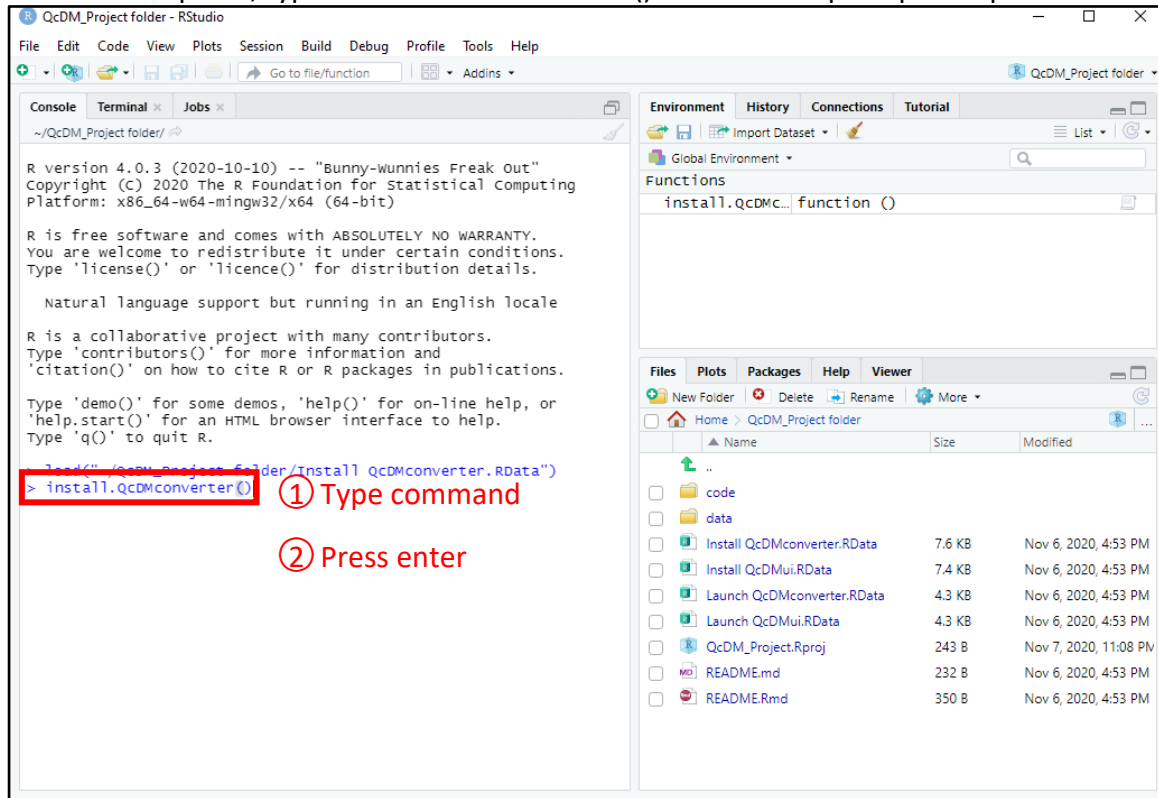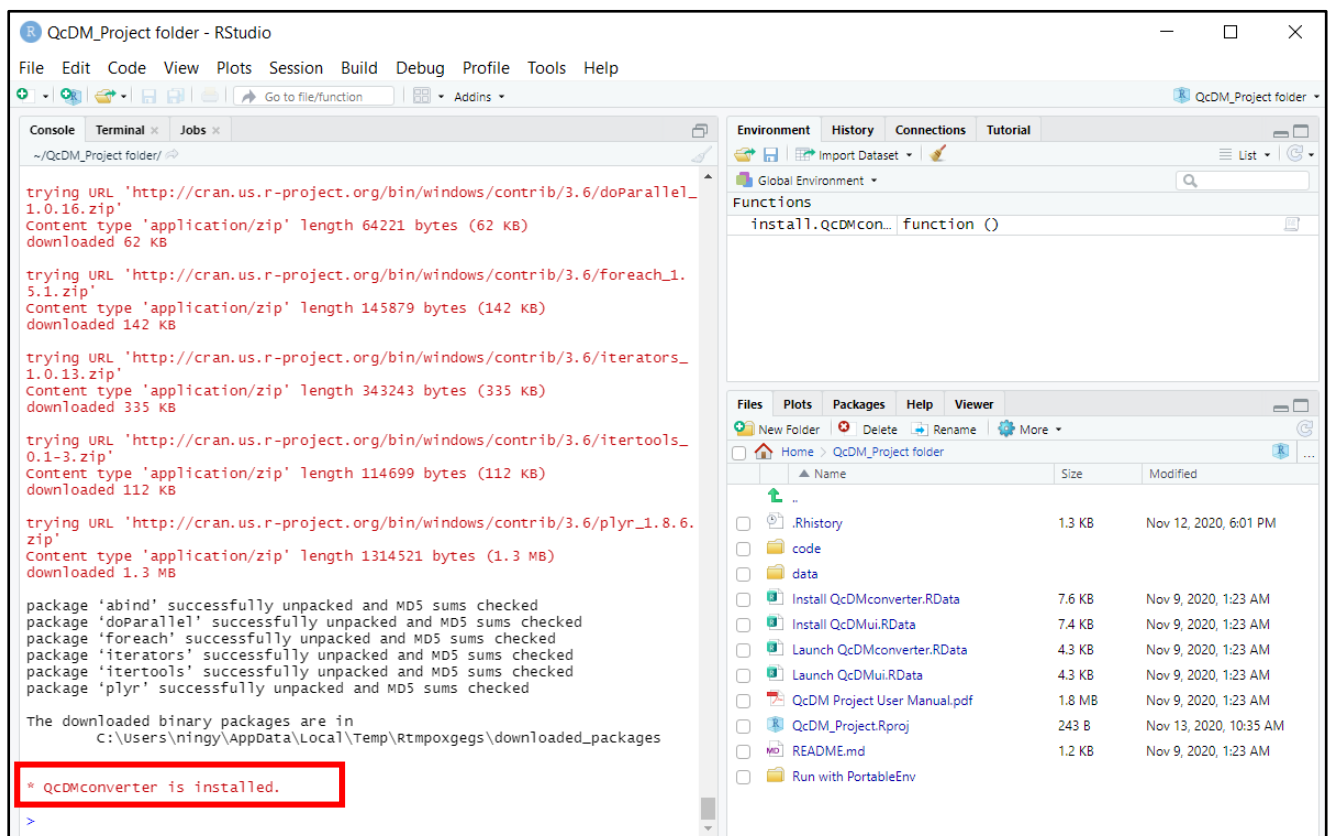

1.3.1. Install QcDMconverter

In the File panel of RStudio, click 'Install QcDMconverter.RData', and click 'Yes' in the pop-up window.

In the Console panel, type 'install.QcDMconverter()' after the '>' prompt and press Enter.

Installation is successful when a message '* QcDMconverter is installed' is displayed in the Console panel (please ignore 'warning message' if any). Do not close the RStudio window as it will be used to install another R package (QcDMui).

1.3.2. Install QcDMui

Similar to the steps in the previous section, click on 'Install QcDMui.RData' in the File panel and click 'Yes' in the pop-up window.

In the Console panel, type 'install.QcDMui()' after the '>' prompt and press Enter.

Installation is successful when a message '* QcDMui is installed' is displayed in the Console panel.

Close the RStudio window after successfully installing QcDMconverter and QcDMui. If prompted to select whether to save workspace image or history when closing the window, there is no need to save either of them.

2. User guide for QcDMconverter

QcDMconverter is an R-Shiny application that processes raw input glucose data based on the requirements of the QcDMui to simplify the analytical steps when generating glucometrics measurements.
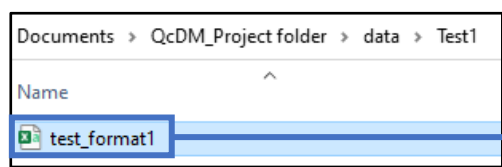
The following information is required from an input glucose data file for the generation of glucometrics:

- an indicator for hospital stays experienced by each patient,
- the date-time stamp associated with each glucose measurement,
- the value of each glucose measurement,
- the ward location associated with each glucose measurement.

New input glucose data may be provided as a single comma-separated version (CSV) file (e.g., see Test1 described in Section 2.1), or as multiple CSV files that follow the same format (e.g., see Test2 described in Section 2.2). QcDMconverter extracts essential information from input file(s) and converts them to a common format that conforms to the requirements by QcDMui.

2.1. Test1: Processing a single input file using QcDMconverter

To illustrate the use of QcDMconverter, we simulated glucose data with 4459 rows for 100 unique hospital stays in July 2020, randomly assigned into 4 ward locations (i.e., A, B, C and D). The unit of measurement of the glucose readings is mmol/L. For illustrative purposes, 6 of the 4459 rows had non-numerical glucose readings. This data, which conforms to the required data format by QcDMui (described in details in Section 3.1), is saved as a CSV file named 'test_formst1.csv' in the 'Test1' folder, which is in the 'data' subfolder of 'QcDM_Project folder'.


Documents › QcDM_Project folder › data › Test1

Name

test_format1

|	A	B	C	D
1	ADMISSION.ID	RESULT	RESULT.DATE	LOCATION
2	8226	5.7	07/01/2020 00:07	D
3	4194	6.3	07/01/2020 00:03	A
4	143	3.6	07/01/2020 00:08	C
5	6363	19	07/01/2020 00:06	C
6	3931	8.8	07/01/2020 00:04	C
7	4105	16.9	07/01/2020 00:08	C
8	270	7.1	07/01/2020 00:09	D
9	5598	7.5	07/01/2020 00:05	D
10	6709	7.7	07/01/2020 00:07	B

2.1.1. Launch QcDMconverter

To ensure QcDMconverter can be launched properly by using 'Launch QcDMconverter.RData' in the 'QcDM_Project folder', follow the instructions below.

- Goto 'QcDMui_project folder', double click 'QcDM_Project.Rproj' to open RStudio.
- Click 'Launch QcDMconverter.RData' in the File panel, click 'Yes' in the pop-up window.
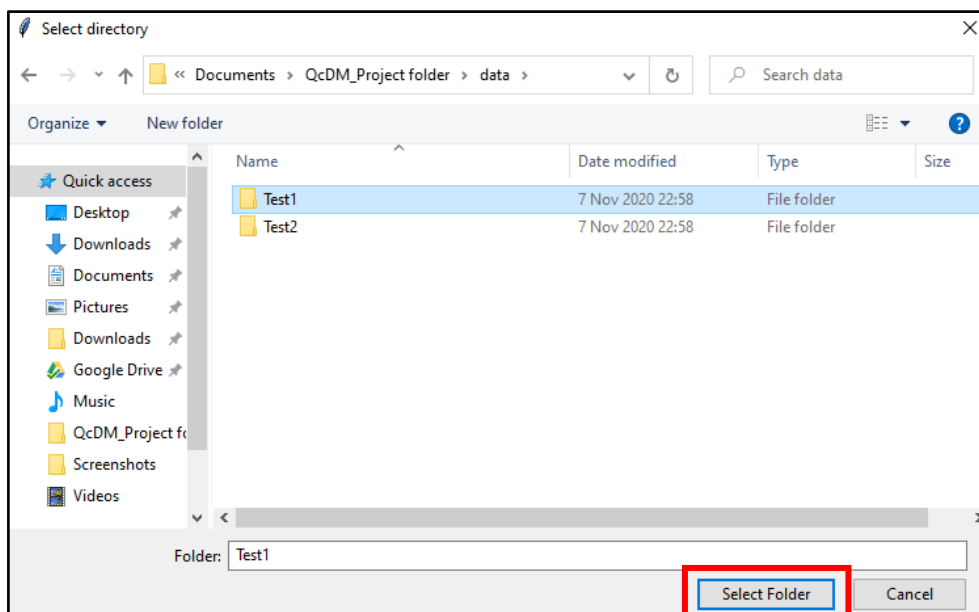- In the Console panel, execute command 'begin.analysis()' by typing it after the '>' prompt and pressing Enter.
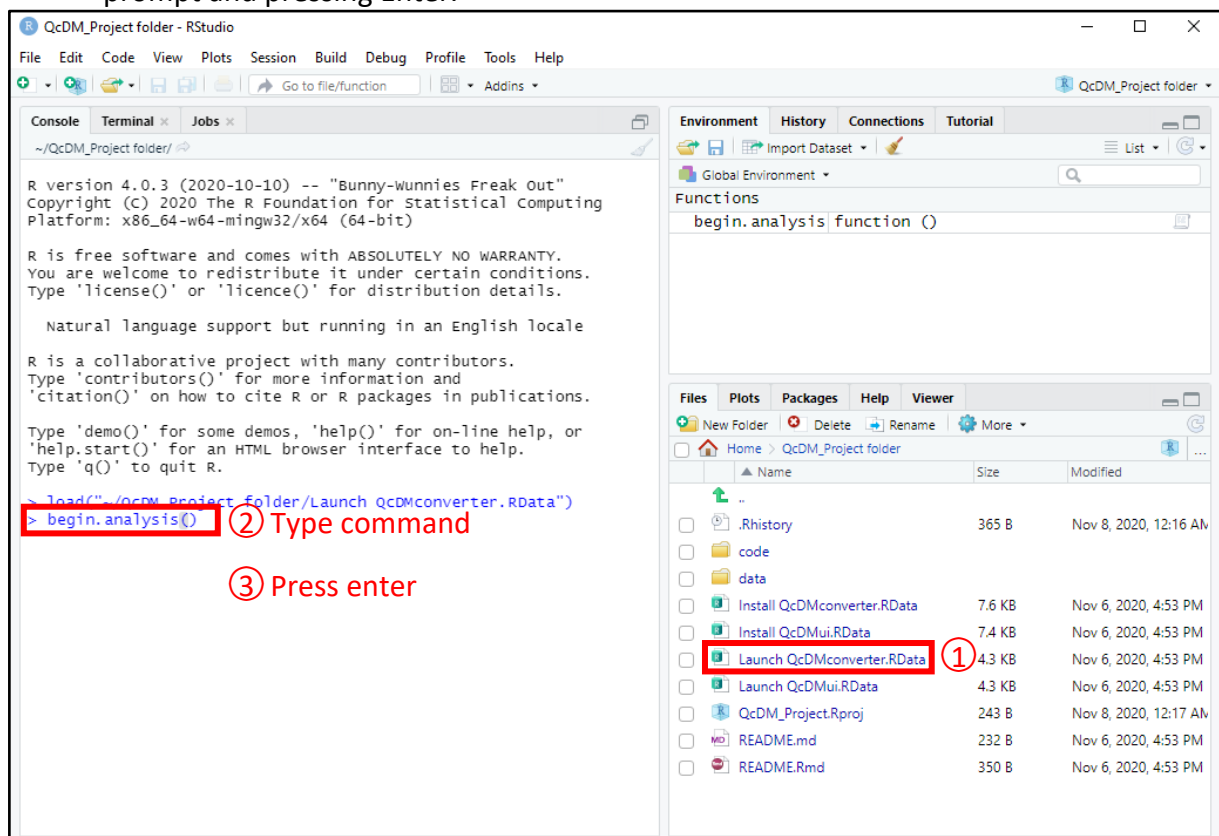
Upon launching QcDMconverter, the following window will be displayed:
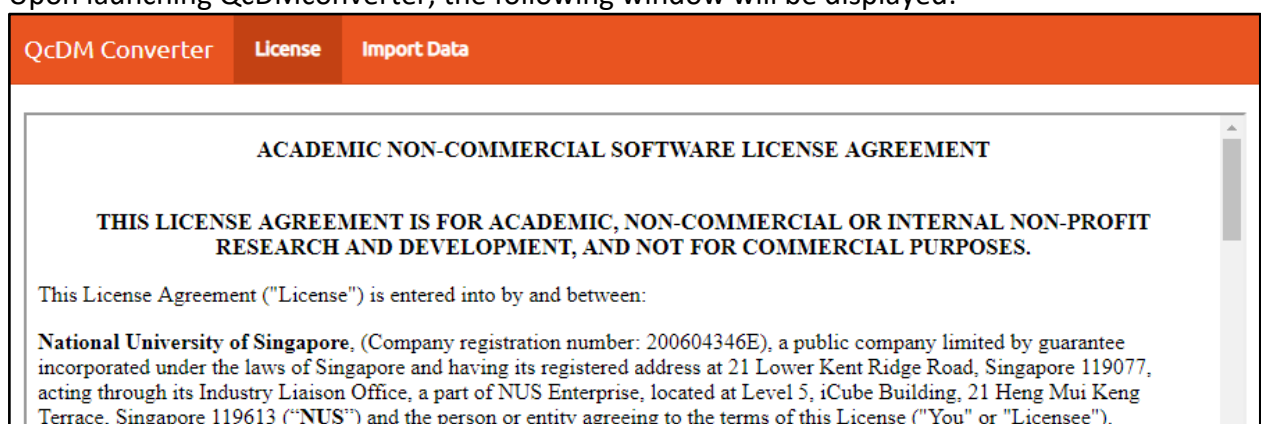
2.1.2. Process data

Click on the 'Import Data' tab to begin data processing. The following window will pop out for users to locate the 'Test1' folder:

Click 'Select Folder' to proceed, and the following dashboard will be displayed. Use it to select appropriate options for reading in the input file, and click 'Next' to proceed.

Use the following tab to configuration additional data options based on the input file selected, then click 'Import' to begin data processing. **Note that this will overwrite existing output file that was previously generated from the same input file.**

The following text message will be displayed at the top of this tab to indicate the completion of data processing:

```
Single File Processing Mode:

Processing succeeded for the following files:
SUC: C:/Users/ningy/Documents/QcDM_Project folder/data/Test1/test_format1.csv

1 of 1 processed successfully
```

2.1.3.   Output from data processing

After performing the data processing steps in Section 0, a folder named 'Test1_processed' is created in the same folder as 'Test1', which contains four subfolders:
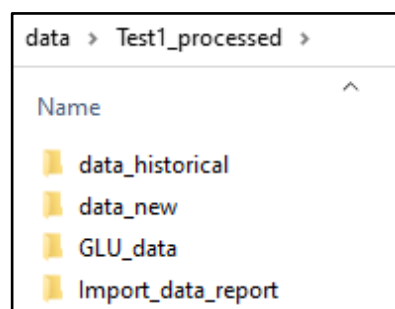
Subfolder 'data_new' contains a file named 'Jul2020_test_format1.csv', which was created from the input 'test_format1.csv', now with the month and year of the glucose data appended to the beginning of its file name, and with the date-time stamp converted to

format 'yyyy-mm-dd hh:mm:ss'. As seconds are missing in the date-time stamp in the input data, they are assigned to be '11' by QcDMconverter after data processing.

When proceeding to the generation of glucometrics measurements, QcDMui will look for input glucose data in the 'data_new' subfolder (see Section 3). Subfolders 'data_historical' and 'GLU_data', which are empty now, will be used later by QcDMui to archive input data and store processed data, respectively.

Subfolder 'Import_data_report' stores PDF reports that will be created by QcDMconverter. There will be two reports per input file, with the names of the corresponding input appended to the end of file names:

'Overall summary_test_format1.pdf' provides basic summary statistics for the input glucose values and date-time stamps to facilitate early detection of data entries errors. For example, the 6 entries in 'test_format1.csv' with non-numerical glucose values are reported:

'test_format1.csv':

	A	B	C	D
1	ADMISSION.ID	RESULT	RESULT.DATE	LOCATION
2	3588	d9.6	07/05/2020 11:45	B
3	4194	c7.7	07/05/2020 23:51	A
4	4558	a4.4	07/05/2020 06:06	C
5	3588	>33.5	07/09/2020 12:09	B
6	4194	> 33.3	07/10/2020 19:51	A
7	4194	> 33.3	07/10/2020 20:08	A
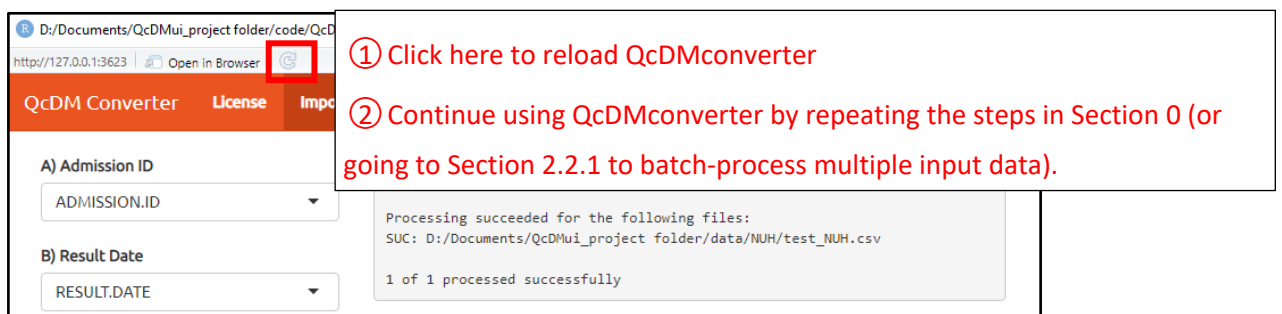
'Overall summary_test_format1.pdf':

2) Report of invalid readings

Invalid BG values

Label	N (%)
> 33.3	2 (0.0)
>33.5	1 (0.0)
a4.4	1 (0.0)
c7.7	1 (0.0)
d9.6	1 (0.0)

Invalid BG timestamps

There is no invalid BG timestamp in the input data.

'Detailed summary_test_format1.pdf' provides more details on the input glucose values and date-time stamps in each ward location.

Invalid glucose values and date-time stamps in the input glucose data will be recoded to 'NA' (which indicates missing data in R) by QcDMconverter in the processed data. If this is inline with users' intention (e.g., the non-numerical glucose values are codes used to indicate entries without glucose measurements and hence should be coded as missing), users may proceed to generate glucometrics from the processed data using QcDMui (see Section 3). If the invalid readings identified in the report reflect errors in data entry, users may want to make the necessary corrections to the raw input data and repeat the steps in Section 2.1.2.

After the completion of data processing for Test1, users may reload QcDMconverter to process other input data:
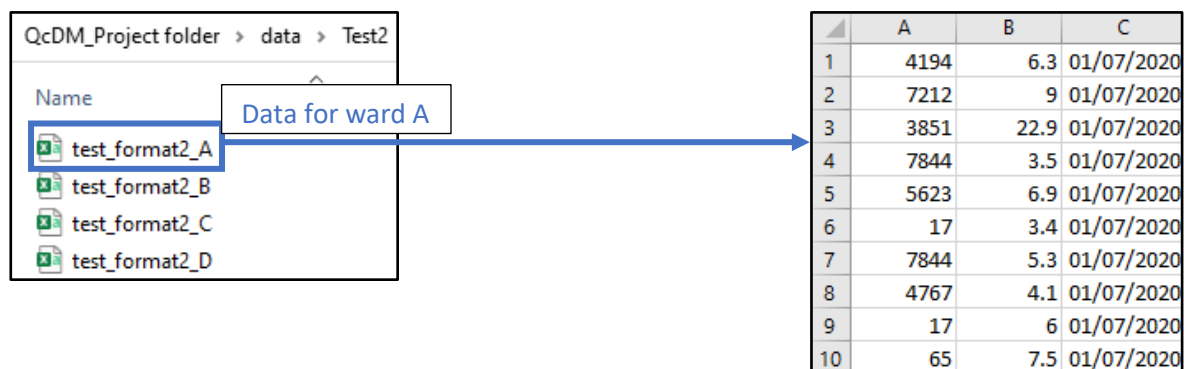
① Click here to reload QcDMconverter

② Continue using QcDMconverter by repeating the steps in Section 0 (or going to Section 2.2.1 to batch-process multiple input data).

```
Processing succeeded for the following files:
SUC: D:/Documents/QcDMui_project folder/data/NUH/test_NUH.csv

1 of 1 processed successfully
```

Users may also close the QcDMconverter window to return to the RStudio window, where they can proceed to the generation of glucometrics measurements using QcDMui (see Section 3), or close the RStudio window now and proceed later. If prompted to select whether to save workspace image when closing the RStudio window, there is no need to save either of them.

2.2. Test2: Processing multiple input files with the same format

QcDMconverter can batch-process multiple files in the folder if they follow the same format. To illustrate this mode of usage, 'test_format1.csv' is split into 4 CSV files and saved to the folder 'Test2' (which is in the 'data' subfolder of 'QcDM_Project folder'), where each file contains all the entries corresponding to the same ward.
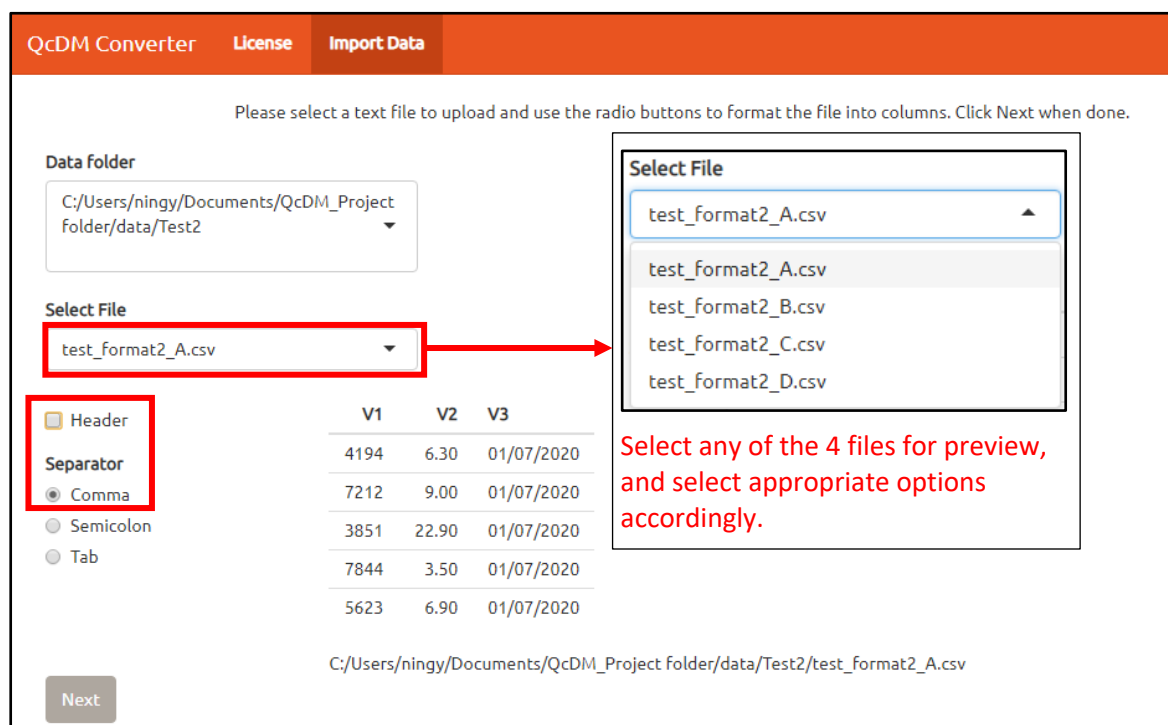
For illustrative purpose, each file now contains three columns, without column names, for the admission ID, glucose values and date stamp for the glucose measurements (without hours or minutes), respectively. Such file format is used in the Yale Glucometrics project when generating glucometrics (1). As the ward location is no longer included as a data column, it must be specified as the last few characters in the file names, preceded by '_'.

2.2.1. Process data

After launching the QcDMconverter GUI as described in Section 0 and selecting the 'Test2' folder using the GUI, users can select any of the 4 files in this folder for preview, in order to inform the specification of options when reading in these files. Make sure you uncheck 'Header'.

After clicking 'Next', the GUI displays an additional option E upon detecting multiple files in the folder selected. Users may choose to process one file each time, or process all files in the folder by selecting 'Yes', provided that all files follow the same format.

Ward location is now specified as part of the file name instead of as a column in the data.

When the GUI detects multiple files in the selected folder, users have this additional option of batch-processing all files using the same configuration.

After configuration, click 'Import' (under option E) to start processing.

2.2.2. Output from data processing

As described in Section 2.1.3, a folder named 'Test2_processed' is created by QcDMconverter, with the following four subfolders:
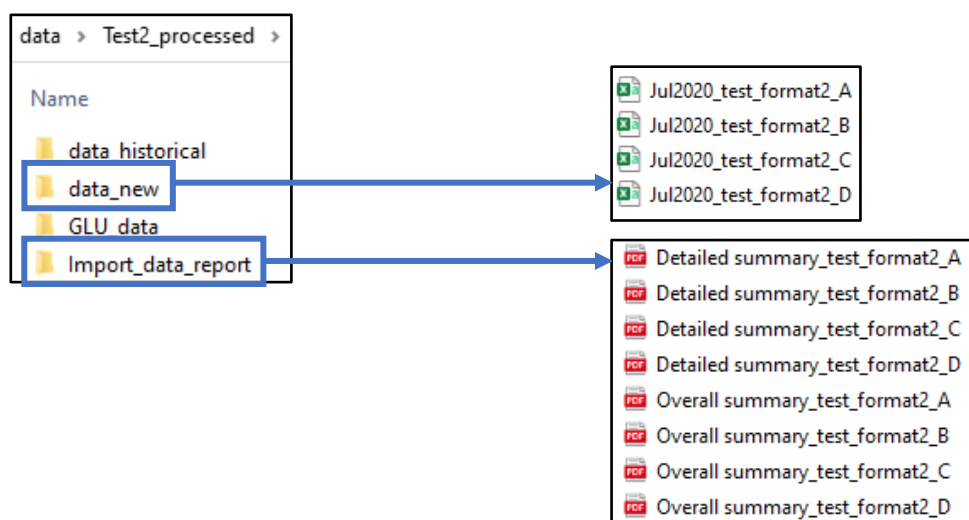
Despite the different data formats adopted in Test1 and Test2, the processed data files in the 'data_new' folder follow the same format:

	A	B	C	D
1	ADMISSION.ID	RESULT.DATE	RESULT	LOCATION
2	4194	2020-07-01 11:11:11	6.3	A
3	7212	2020-07-01 11:11:11	9	A
4	3851	2020-07-01 11:11:11	22.9	A
5	7844	2020-07-01 11:11:11	3.5	A
6	5623	2020-07-01 11:11:11	6.9	A
7	17	2020-07-01 11:11:11	3.4	A
8	7844	2020-07-01 11:11:11	5.3	A
9	4767	2020-07-01 11:11:11	4.1	A
10	17	2020-07-01 11:11:11	6	A
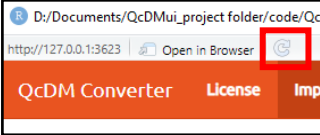
As time stamps are not available in the input data files in Test2, they are assigned as 11:11:11 AM after the data processing.

As described in Section 2.1.3, the overall summary reports generated for each input file is useful for detecting invalid entries in glucose values and date-time stamp. For example, 3 glucose values in the input file for ward A are reported to be non-numerical in 'Overall summary_test_format2_A.pdf':

2) Report of invalid readings

Invalid BG values

Label	N (%)
> 33.3	2 (0.3)
c7.7	1 (0.2)

Invalid BG timestamps

There is no invalid BG timestamp in the input data.

More detailed summaries of glucose values and date-time stamps for each ward can be found in their respective detailed summary reports. Users may proceed to use QcDMui (see Section 3) provided information you are fine with the information provided in the pdf files.

After the completion of data processing for Test2, users may reload QcDMconverter to process other input data:

D:/Documents/QcDMui_project folder/code/Qc
http://127.0.0.1:3623 Open in Browser
QcDM Converter   License   Imp

① Click here to reload QcDMconverter.

② Continue using QcDMconverter by repeating the steps in Section 0 or 2.2.1.
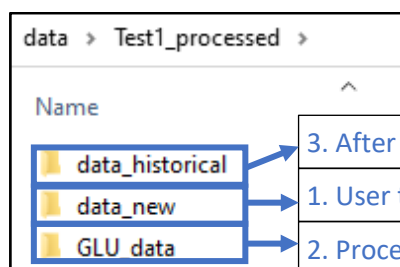
Users may also close the QcDMconverter window to return to the RStudio window, where they can proceed to the generation of glucometrics measurements using QcDMui (see Section 3), or close the RStudio window now and proceed later. If prompted to select whether to save workspace image when closing the RStudio window, there is no need to save either of them.

3. User guide for QcDMui

QcDMui is an R-Shiny graphical user interface developed for convenient and flexible generation of glucometrics measures from point-of-care glucose data.

3.1. Required data format of QcDMui

QcDMui requires a specific data format and folder structure for the input glucose data files. Specifically, QcDMui expects a data folder to have the following subfolders:

For easy management, users are recommended to provide glucose data by month, and indicate the calendar time of each file using the first seven characters of the file name, where the first three characters are the abbreviated month, and the next four characters correspond to the four-digit calendar year (e.g., 'Jul2020' for glucose data in July 2020).

As illustrated in the snapshot below, each data file in 'data_new' should contain at least four variables:

- ADMISSION.ID: the identifier for a specific hospital stay experienced by a patient.
- RESULT.DATE: the date-time stamp associated with each glucose measurement, where:
  - only '/' or '-' is used to separate date, month and year, and no mix of the two symbols is allowed in a single time variable (see below for an example)
  - ':' is used to separate hour, minute and second, and the order is always 'hh:mm:ss' (see below for an example). Second can be missing, but not minute or hour. This is to allow us to perform distributional analytics for assessing timeliness of glucose measurements (2).
- RESULT: the value of glucose measurement. It can be in mg/dL or mmol/L, but not a mix of the two units. If 95% of glucose values are greater than 33.3, then the unit of these values is taken as mg/dL, otherwise, the unit will be taken as mmol/L.
- LOCATION: the ward location where the glucose measurements were taken.

Users are strongly recommended to process raw input data files using QcDMconverter, as described in Section 2, to create the required folder structure and conform to the required data format to facilitate their usage of the QcDMui.

The next two sections illustrate the usage of QcDMui and the corresponding output using example data provided in folder 'Test1', which is available in the 'data' subfolder of 'QcDM_Project folder'. We assume users have followed the instructions in Section 2.1 to create a folder 'Test1_processed' that contains the required subfolders and processed data.

3.2. Launch QcDMui

QcDMui can be launched by using 'Launch QcDMui.RData' in the 'QcDM_Project folder' following the instructions below.

- Goto 'QcDM_Project folder', double click 'QcDM_Project.Rproj' to open RStudio.
- Click 'Launch QcDMui.RData' in the File panel, click 'Yes' in the pop-up window.
- In the Console panel, execute command 'begin.analysis()' by typing it after the '>' prompt and pressing Enter.
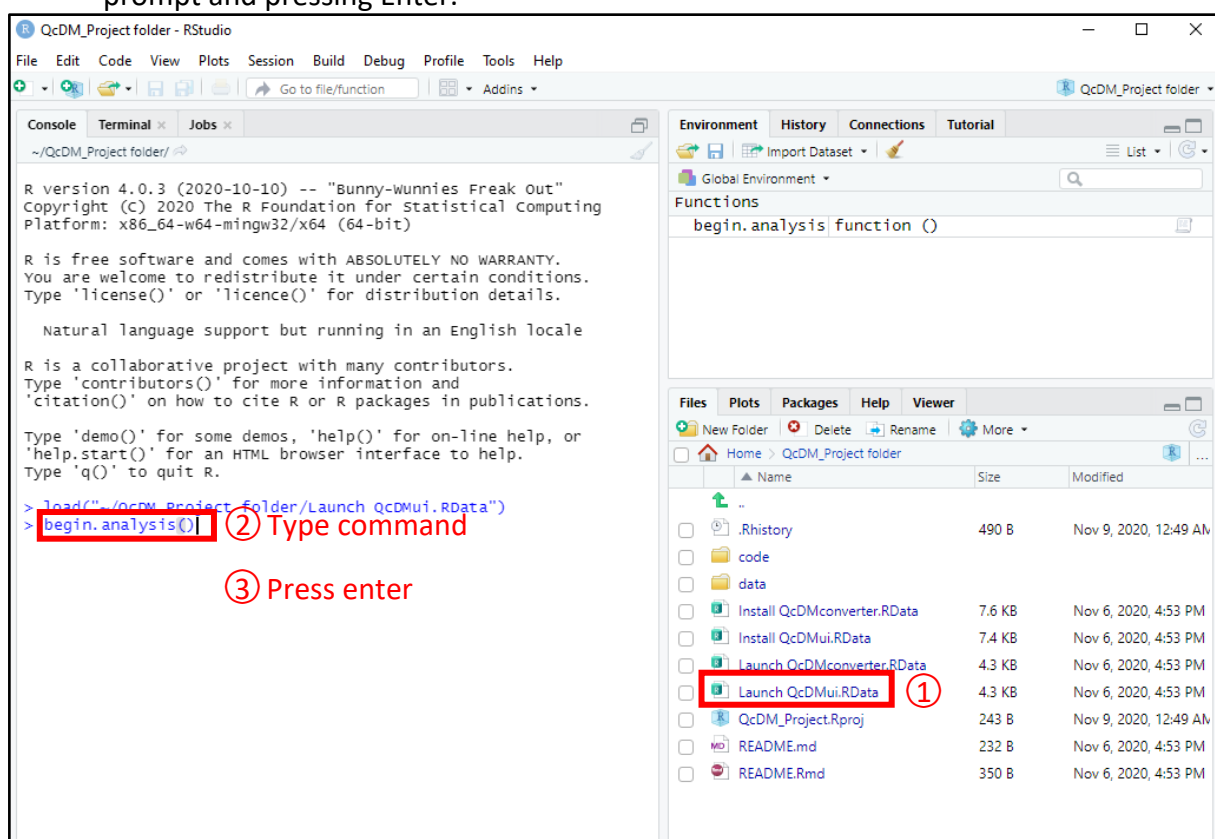
Upon launching QcDMui, the following window will be displayed:

18
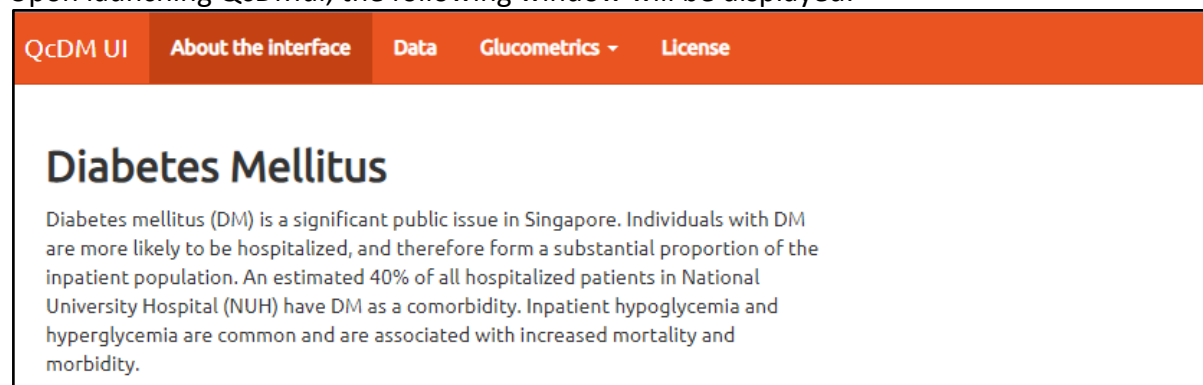
An introduction to the tool and a brief instruction on how to use it is available under the tab 'About the interface'.

3.3. Process data

Users can use the 'Data' tab to specify the glucose data for which glucometrics measurements need to be generated.

The following window will pop out upon clicking on the 'Data' tab for users to locate the 'Test1_processed' folder. Click 'Select Folder' to proceed. Confirm the selection by clicking 'Confirm data folder' in the GUI.

The following two subsections describe the 'Data' tab in detail.

3.3.1. Specify cutoffs and exclusion criteria

*: If 95% of glucose values are greater than 33.3, then the unit of these values is taken as mg/dL, otherwise, the unit will be taken as mmol/L.

After specifying appropriate cutoff values and exclusion criteria, click on 'Confirm' button. QcDMui will import data in the 'data_new' subfolder, split them by month and ward, and

organize the processed data into the respective subfolders within the 'GLU_data' subfolder. A text message 'Process complete.' will be displayed under the 'Confirm' button upon completion of data processing. Users can then proceed to the panel on the right to select time period and wards for the generation of glucometrics.

3.3.2. Select time period and wards

Users are required to use these two parts to specify the period and ward for glucometrics generation, based on the availability of data in the 'GLU_data' subfolder.

After selecting the ward and time range, a summary of the selected data by ward and month will be presented here, so that users can modify their selection if the selected data is not available.

3.4. View glucometrics

After processing glucose data in 'Data' tab, click on 'Glucometrics' tab and then go to 'Summary Statistics' tab to view the following tables that summarize the glucometrics information:

Data Summary

Summary of BG measurements	Summary
Total (N)	603
Minimum (mmol/L)	3.3
5-th percentile (mmol/L)	4.4
25-th percentile (mmol/L)	5.4
75-th percentile (mmol/L)	9.9
95-th percentile (mmol/L)	16.4
Maximum (mmol/L)	31
Non-numeric values (N)	3

Exclusion Summary

Total number of patient-stays	15
No exclusion criterion was selected.	

Glucometrics

Summary		Patient-sample	Patient-day$^{\#1}$	Patient-stay$^{\#2}$
Location				A
Period				01 Jul 2020 to 31 Jul 2020
Count		600	141	15
GLYCEMIC CONTROL				
Hyperglycemia				
Percent with glucose >= 14 mmol/L		56 (9.3%)	25 (17.7%)	6 (40%)
Percent with glucose >= 20 mmol/L		15 (2.5%)	7 (5%)	2 (13.3%)
Percent with glucose >= 24 mmol/L		8 (1.3%)	3 (2.1%)	1 (6.7%)
Hyperglycemia index (HGI): AUC (> 10 mmol/L)/LOS (in hours)*				
	Median (IQR)			0 (1.8)
	Mean (SD)			1.6 (2.7)
Other metrics				
Percent with glucose >= 4 and < 10 mmol/L		439 (73.2%)	110 (78%)	11 (73.3%)
Glucose (mmol/L)*				
	Median (IQR)	6.7 (4.9)	6.9 (4)	7 (4.9)
	Mean (SD)	8.2 (4.1)	8.4 (4.2)	8.7 (4)
Average patient-day mean glucose for a patient-stay (mmol/L)*				
	Median (IQR)			7 (4.9)
	Mean (SD)			8.7 (4)
HYPOGLYCEMIA				
Percent with glucose < 4 mmol/L		13 (2.2%)	10 (7.1%)	7 (46.7%)
Percent with glucose < 3 mmol/L		0 (0%)	0 (0%)	0 (0%)
Percent with glucose < 2.5 mmol/L		0 (0%)	0 (0%)	0 (0%)
Percent of patient-stays with a recurrent hypoglycemia (< 4 mmol/L) day (10-240 mins)				2 (13.3%)
GLYCEMIC VARIABILITY				
Standard deviation: SD (mmol/L)^{2*}				
	Median (IQR)		1.2 (2)	2 (1.9)
	Mean (SD)		1.6 (1.3)	2.4 (1.9)
J-index (mmol/L)^{2*}				
	Median (IQR)		22.5 (37.9)	31.2 (57.6)
	Mean (SD)		41.5 (48.4)	52 (50.2)

*:The summary statistics of the glucometrics for patient-day means and patient-stay means.
$^{\#1}$: 17 (12.06%) patient-days were removed for calculation of patient-day SD and J-index.
$^{\#2}$: 1 (6.67%) patient-stays were removed for calculation of patient-stay SD and J-index.

Summary Statistics report is saved in the folder: 'C:/Users/ningy/Documents/QcDM_Project folder/data/Test1_processed/Glucometric_report' as a CSV file named 'Test1_processed_2020_11_16_12_38_17.csv' and as a HTML report named 'Test1_processed_2020_11_16_12_38_17.html'.

Tables above are also saved as a CSV file and a HTML report in the 'Glucometric_report' folder for later use:

data > Test1_processed > Glucometric_report

Name
Test1_processed_2020_11_16_12_38_17
Test1_processed_2020_11_16_12_38_17

The HTML report can also be saved as a PDF document by opening it using the Chrome browser:

After viewing glucometrics tables, users may reload QcDMui to process other glucose data:

Users may also close the QcDMui window to return to the RStudio window. When closing the RStudio window, if users are prompted to select whether to save workspace image, there is no need to save either of them.

4. Running QcDM Project from USB Drive

In the previous three sections, we have described how to install and use the QcDM Project on a desktop with a Windows or Mac operating system. Alternatively, the QcDM Project can be installed into a USB Drive and used in a plug-and-play approach on any computer with the **Windows** operating system. This portable approach is not available for the Mac operating system.

4.1. Set up the portable environment

The portable environment for running the QcDM Project, which includes R-portable (version 3.6.3) and two software (Pandoc and TinyTeX) for generating PDF reports.

Follow the instructions in Section 1.2 to download the QcDM Project as zip file from the GitHub repository, unzip it, rename the resulting folder as 'QcDM_Project folder', and move this folder to the root directory of the USB Drive.

Download the portable environment as a zip file named 'PortableEnv.zip' from the following link:
https://drive.google.com/file/d/1bPk8pwZ2HaCDbXhP8e67wcVzWNXs4R76/view?usp=sharing.
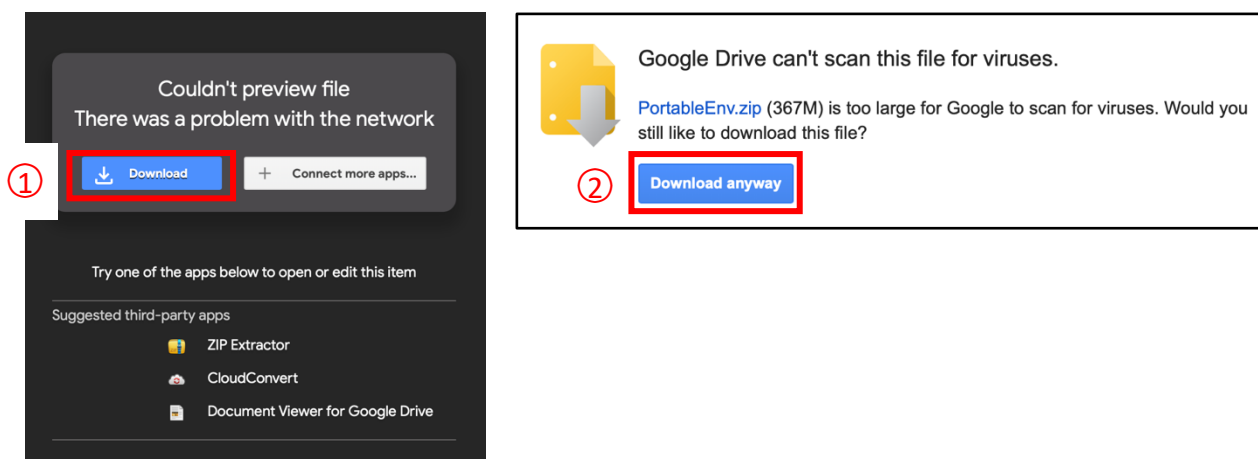
Unzip the file (see Section 1.2 for instructions) to a folder named 'PortableEnv', and move this folder to the root directory of the USB Drive.
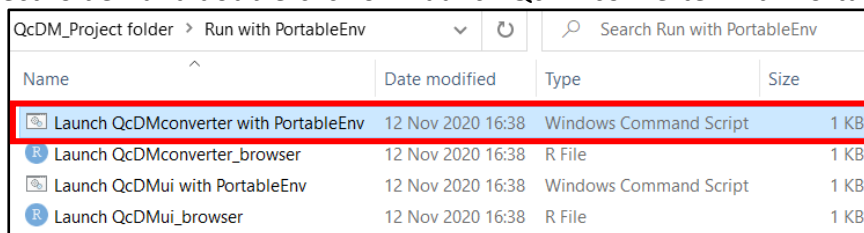
The root directory of the USB Drive should contain the following two folders, which take up approximately 1.8 GB on the disk:

The portable environment has been configured by installing the R packages and software required by the QcDM Project, therefore users do not need to go through the installation steps in Section 1.1 and 1.3.
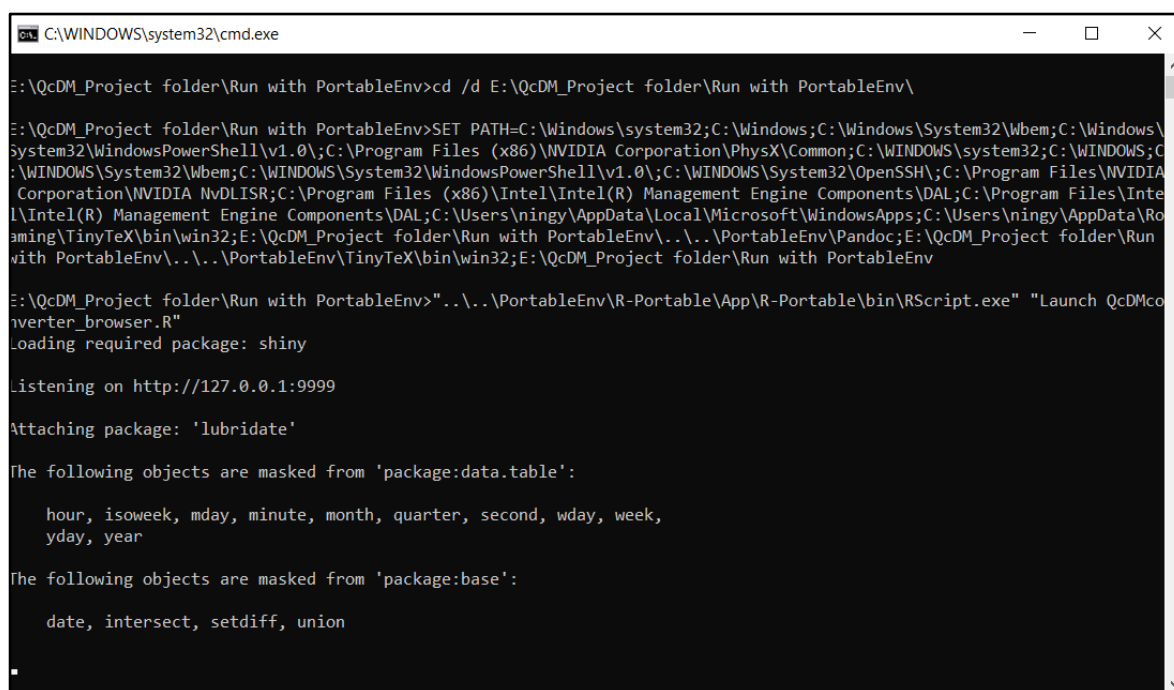
4.2. Run QcDMconverter with PortableEnv

To launch QcDMconverter from the USB Drive, go to the 'Run with PortableEnv' subfolder of 'QcDM_Project folder' and double-click on 'Launch QcDMconverter with PortableEnv.cmd':

The following command line window will pop-up:

and then QcDMconverter will open in the default web browser (in this example Chrome):
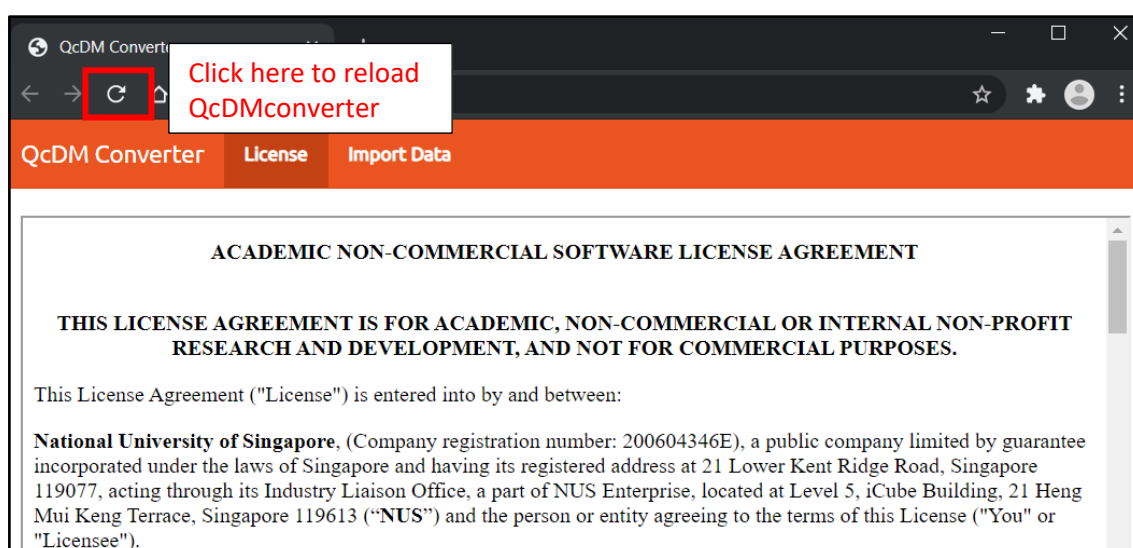
Users can follow the instructions in Section 2.1.2 or 2.2.1 to process glucose data.

After successful completion of data processing, users may reload QcDMconverter and proceed to additional data by clicking the reload button of the web browser (as shown in the screenshot above). To close QcDMconverter, first close the QcDMconverter window in the web browser, then close the command line window.

4.3. Run QcDMui with PortableEnv

To launch QcDMui from the USB Drive, go to the 'Run with PortableEnv' subfolder of 'QcDM_Project folder' and double-click on 'Launch QcDMui with PortableEnv.cmd':

The following command line window will pop-up:

and then QcDMui will open in the default web browser (in this example Chrome):
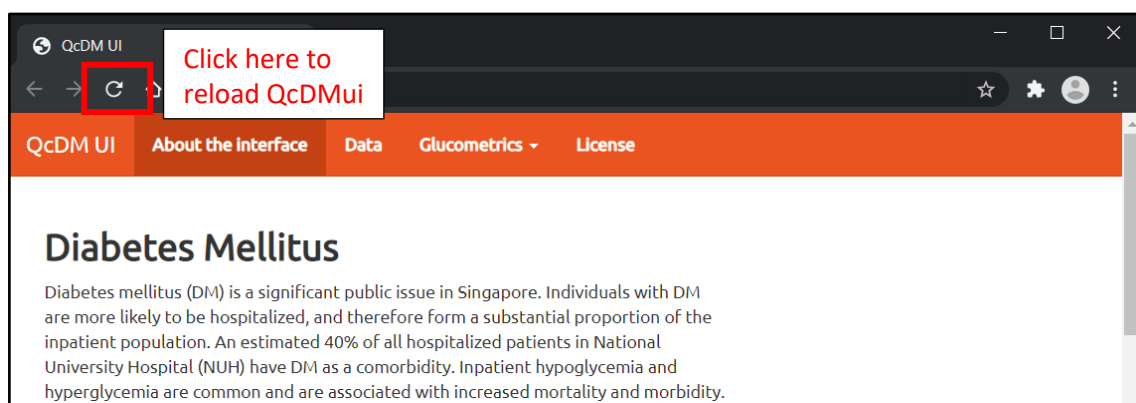
Users can follow the instructions in Section 3.3 and 3.4 to process data and view glucometrics.

After successful completion of glucometrics generation, users may reload QcDMui and analyse additional glucose data by clicking the reload button of the web browser (as shown in the screenshot above). To close QcDMui, first close the QcDMui window in the web browser, then close the command line window.

Bibliography

1. Thomas P, Inzucchi SE. An internet service supporting quality assessment of inpatient glycemic control. In: Journal of Diabetes Science and Technology. SAGE Publications Inc.; 2008. p. 402–8.
2. Chen Y, Kao SL, Tai E-S, Wee HL, Khoo EYH, Ning Y, et al. Utilizing distributional analytics and electronic records to assess timeliness of inpatient blood glucose monitoring in non-critical care wards. BMC Med Res Methodol. 2016;16(1).