<前半部為中文版本，後半部為英文版本>

[前言]

本研究致力於使研究學者具備一種多功能、簡便的兩組樣本間甲基化程度分析工具包，基於這個目的本研究亦設計一套對使用者友善的圖形使用者介面程式，以提升研究人員之使用效率。

[設計架構]

本研究主要程式採用python撰寫而成，基於此本介面程式亦採用python撰寫而成的pyqt5模組開發，藉以提升程式的維護性。

按照本研究流程的需求，介面程式內分別設計檔案選擇區、參數設定區、工具選用及雜項設定區及狀態監控欄及一款簡易型報告閱讀器。

[介面程式]

本介面程式採用pyqt5模組開發，得益於該模組的高度封裝及完整性，簡化及加速設計維護的難度及時間。

本介面程式以一個main window物件容納各設計區，以及一個widget物件容納簡易型報告閱讀器，以及兩個widget物件用於參數設定區之設定（於後面詳細介紹）。main window物件內建widget、選單列以及狀態列，程式開發者可在上述三個容器中加入各式物件。widget則為單一視窗容器，亦可容納各式物件。

本介面程式之各區域排版均使用pyqt5中的GridLayout函數完成，採用矩陣式排列各物件與容器，排版簡潔以符合使用者友善之目標。

[檔案選擇區]

本區域功能為令使用者選取分析所需之輸入檔案，為符合此需求，開發者設計一組檔案選擇器，內部包含觸發檔案選取機制之按鈕及可讓使用者確認檔案路徑之文字標籤，按鈕連接觸發pyqt5中File Dialogue函數，使使用者選取檔案後返還其路徑，並列於文字標籤上供檢視。本區域再以一個Group box物件包裝，排版於整個介面程式之最上方處。

[參數設定區]

本區域功能為令使用者設定分析所需之必要參數，為符合此需求，開發者設計一個參數設定模塊，其包含一系列文字標籤及文字方塊，文字方塊使用pyqt5中Validator函數功能按參數性質予以限制輸入格式，避免非法參數進入分析中。文字標籤顯示參數名稱，並加入pyqt5中文字標籤的ToolTip功能，基於此功能，文字標籤上得以顯示參數之說明對話窗，貫徹使用者友善的目標。本區域再以一Group box物件包裝之，排版於介面程式中間處，佔有較大面積比例，具有使用直覺性。

[工具選用區]

本區域功能為令使用者選取欲使用的工具及設定其餘雜項，工具選取部分開發者設計一組工具勾選區塊，包含一系列checkbox物件及文字標籤，文字標籤顯示工具名稱，checkbox為pyqt5內建的勾選方塊，透過checkState函數可獲取其勾選狀態，於分析中檢查之，達成選用（啟用及禁用）部分工具的功能需求。

雜項設定部分，其功能包含令使用者指定檔案輸出路徑及自動關機功能。基於上述需求，指定輸出路徑部分亦採用File Dialogue函數令使用者取取資料夾後返還其路徑，並以一文字標籤顯示之。自動關機部分，設計一勾選方塊提供啟用禁用功能，此方塊若啟用將於分析完成後觸發command ‘shutdown -s -t 60’，當有足夠權限下，電腦將於60秒後關機。

以上兩部份各自以一Group box物件包裝之，排版於本介面程式下方。

[狀態監控爛]

本區域功能為令使用者得以實時監控分析進度及狀態，為符合此需求，開發者使用pyqt5中progress bar物件及文字標籤設計一進度檢視區塊，於分析中分析程式將不斷返還分析狀態與進度百分比（透過pyqt5中的emit及connection函數），進度檢視區塊將承接之並顯示。此外，設計一按鈕用於觸發分析程式開始分析，一按鈕用於觸發開啟簡易型報告閱讀器。

[介面程式與分析程式]

開始分析按鈕被按下觸發時，若所有參數即設定均完成無誤將進入分析程序，技術上為保持介面程式之穩定性，分析程序另開執行緒執行，透過pyqt5中Threading物件函數功能達成。分析程序一開啟後，前述所有參數即設定於本分析程序起頭處進行初始化動作，其後分析程序與介面程式之溝通透過pyqt5中emit及connect函數進行，以確保兩執行緒的穩定性，本分析程序初始化完成後即按本研究預定流程進行資料分析，過程中分析程序於數個檢查點都將傳送進度與狀態予介面程式。本分析程序完成後即關閉執行緒，並返還完成狀態。

[簡易型報告閱讀器]

本閱讀器功能為令使用者快速且便利得預覽分析結果，為符合此需求，設計有pyqt5物件tab view容器，此容器含三個分頁，分別用於容納三種甲基化類型之報告，各分頁包含數個文字方塊，利用文字方塊之Pix map屬性顯示分析輸出之圖表，並利用resize函數適當縮放。本閱讀器以直瀑式排版顯示之，並以pyqt5中ScrollArea物件實現卷軸功能。本閱讀器於分析未完成前為未啟用狀態，此狀態下本閱讀器上僅以一文字標籤顯示說明訊息，待分析完成後本閱讀器即啟用提供功能。

<下頁開始為英文版本>

**[Introduction]**

This research is dedicated to enabling researchers to have a versatile and simple two-sample inter-sample methylation analysis toolkit. For this purpose, this research also designed a user-friendly graphical user interface program to improve efficiency.

**[Design Architecture]**

The main program of this research was written in python. Based on this, the interface program is also developed using pyqt5 module written in python to improve the maintainability of the program.

According to the requirements of this research process, the file selection area, parameter setting area, tool selection and miscellaneous setting area and status monitoring bar and a simple report browser are respectively designed in the interface program.

**[Interface Program]**

This interface program is developed using pyqt5 module, which benefits from the high degree of packaging and integrity of the module, simplifying and accelerating the difficulty and time of design and maintenance.  
The interface program accommodates each design area with a main window object, and a widget object accommodates a simple report reader, and two widget objects for parameter setting area settings (described in more detail later). The main window object has built-in widgets, menu columns, and status columns. Program developers can add various objects to the above three containers. The widget is a single window container that can hold a variety of objects.  
The layout of each interface of the interface program is completed by using the GridLayout function in pyqt5. The objects and containers are arranged in a matrix, and the layout is simple to meet the user-friendly goal.

**[Files Selection Area]**

The function of this area is to let the user select the input file needed for analysis. To meet this need, the developer designs a set of file selectors that contain buttons that trigger the file selection mechanism and text labels that allow the user to confirm the file path. The button connection triggers the File Dialogue function providing by pyqt5, which allows the user to select the file and return its path. Also list it on the text label for viewing. The area is packaged in a GroupBox object, formatted at the top of the entire interface.

**[Parameters Setting Area]**

This area is designed to allow the user to set the necessary parameters for the analysis. To meet this need, developers have designed a parameter setting module that contains a series of text labels and text boxes. The text box uses the Validator function in pyqt5 to limit the input format by the nature of the parameters, avoiding errors caused by illegal parameters. The text label displays the parameter name and adds the ToolTip function of the text label in pyqt5. Based on this function, the parameter label dialog box is displayed on the text label to implement the user-friendly target. This area is packaged in a GroupBox object and is typeset in the middle of the interfac. It occupies a large area ratio and is intuitive.

**[Tool Selection & Miscellaneous Setting Area]**

The function of this area is to let the user select the tools to be used and set the remaining miscellaneous items. The tool selection section contains a series of checkbox objects and text labels. The text label shows the tool name. The checkbox is a built-in checkbox in pyqt5 and its enable/disable state can be checked by the checkState function. Check its enable/disable state in analysis to achieve the functional requirements of selecting (enabling and disabling) some tools.

The Miscellaneous Settings section, its function includes the user to specify the file output path and automatic shutdown function. Based on the above requirements, the specified output path portion also uses the File Dialogue function to allow the user to retrieve the folder and return its path, and display it on label. In the Auto Shutdown section, a checkbox is provided to enable and disable. If this block is enabled, the command ‘shutdown -s -t 60’ will be triggered after the analysis is completed. When there is sufficient authority, the computer will shut down after 60 seconds.

The above two parts are each packaged in a GroupBox object and typeset near the bottom of the interface.

**[Status monitoring area]**

This area is designed to allow users to monitor the progress and status of the analysis in real time. To meet this need, the region uses the progress bar object and text labels in pyqt5 to design a progress view block. In the analysis, the analysis program will continuously return the analysis status and the progress percentage (Through emit and connection function in pyqt5), and the progress view block will receive and display it. In addition, a button is designed to trigger the analysis program to start the analysis, and a button is used to trigger the simple report browser.

**[Interface Program & Analysis Program]**

When the Start Analysis button is pressed, if all parameters and settings are set, the program will enter the analysis phase. Technically, in order to maintain the stability of the interface program, the analysis program will open a new thread to execute, which is achieved through the Threading object function in pyqt5. After the analysis program is started, all the above parameters are set at the beginning of the analysis program for initialization, and then the communication between the analysis program and the interface program is performed through the pyqt5 emit and connect functions to ensure the stability of the two threads. After the initialization of the program is completed, the data analysis is carried out according to the predetermined process of the research pipeline. During the process, the analysis program transmits the progress and status to the interface program at several checkpoints. After the analysis program is completed, the thread is closed and the completion status will be returned.

**[Simple Analysis Report Browser]**

This function of this browser is that allowing users to preview the analysis results quickly and easily. To meet this need, the browser uses the tab widget container which is in pyqt5. This container is set up with three tabs, which are used to hold three types of methylation type reports. Each page contains several text labels. The Pix map attribute of the text label is used to display the graph of the analysis output, and is appropriately scaled by the resize function. The browser is displayed in a straight waterfall type and uses the ScrollArea object in pyqt5 to implement the scroll function. The browser is not enabled before the analysis is completed. In this state, only the text message is displayed on the browser, and the browser is enabled to provide the function after the analysis is completed.