**Skyn Data Manager (SDM)**

**Instructional Manual**

Status… *Experimental testing*

SDM Version… v1.6.1

OS compatibility… Windows 10

Skyn firmware compatibility… 2.0.8

Manual Last updated… *6.17.23*

For help or to report bugs… *email* [*ndidier@bsd.uchicago.edu*](mailto:ndidier@bsd.uchicago.edu)

**Table of Contents**

Page 2… SDM Functions

Page 3… Downloading SDM

Page 3-4… Running SDM

Page 5… Inputs (Metadata and Skyn files)

Page 6… Replicating Development Environment

**SDM Functions**

*File Configuration*

* Configures Skyn dataset filenames with metadata
* File splitting tool for converting multi-day datasets into separate Excel files

*Signal processing*

* Low quality datasets can be filtered out using user-determined criteria
* Datasets can be cropped according to user-designated durations/times
* Imputes gaps in data (due to device removal or device off), artifacts, and noise
* Provides temperature visualization and detection of potential non-wear

*Feature engineering*

* For each Skyn dataset, the SDM will calculate
  + TAC descriptors: Peak, TAC-AUC, Rise Duration, Fall Duration, Rise Rate, Fall Rate, and several others features.
  + Quality descriptors: Percent/duration of device removal, artifacts, etc.

*Make Predictions on New Data using Built-In Models*

* A random forest model and logistic regression model can be used to make predictions of whether or not a single Skyn dataset corresponds to alcohol consumption
* Model was trained & tested using group k-fold cross validation with 30 alcohol episodes and 30 non-alcohol episodes.
* Random forest and logistic regression achieved accuracy of 97% (58/60)

*Train and Make Predictions with New Models using New Data*

* 2 models are trained: random forest and logistic regression
* Models are trained in two ways: using features from cleaned (processed) data and using features from raw (unprocessed) data
* Altogether, 4 models will be trained and tested

*Automated Excel Workbooks and Reports*

* Each individual dataset will saved as an Excel file that includes 3 tabs:
  + Processed Skyn datasets will be saved with new columns for cleaned & smoothed data
  + Variable key that defines the column names
  + Graphs to visualize the raw versus cleaned data, temperature, etc.
* Whole-cohort analysis will produce an Excel file that includes several tabs for:
  + Feature data
  + Model prediction results
  + Compiled visual summary of each Skyn dataset.
* Several graphs summarizing the models

**Download Software**

Go to: [GitHub - ndidier3/skyn\_data\_manager](https://github.com/ndidier3/skyn_data_manager/tree/main)

Click: Code -> Download ZIP

Open ZIP file (likely in Downloads folder).

Move skyn\_data\_manager folder to a safe location.

**Running the Software**

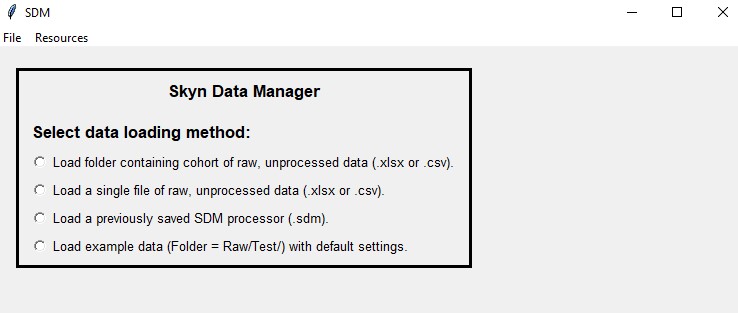
**Step 1**: Double click the SDM file within the skyn\_data\_manager folder. The user interface may take 20-30 seconds to appear.

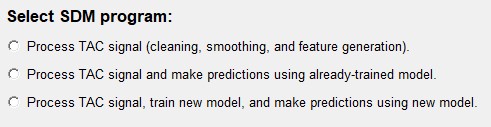
***First time running?*** *Check out the File and Resource dropdowns…*

* **File** includes a File Splitting Tool (slicing a long file into several files) and a Create Metadata Tool.
* **Resources** includes information about citing, documentation, and variable descriptions.

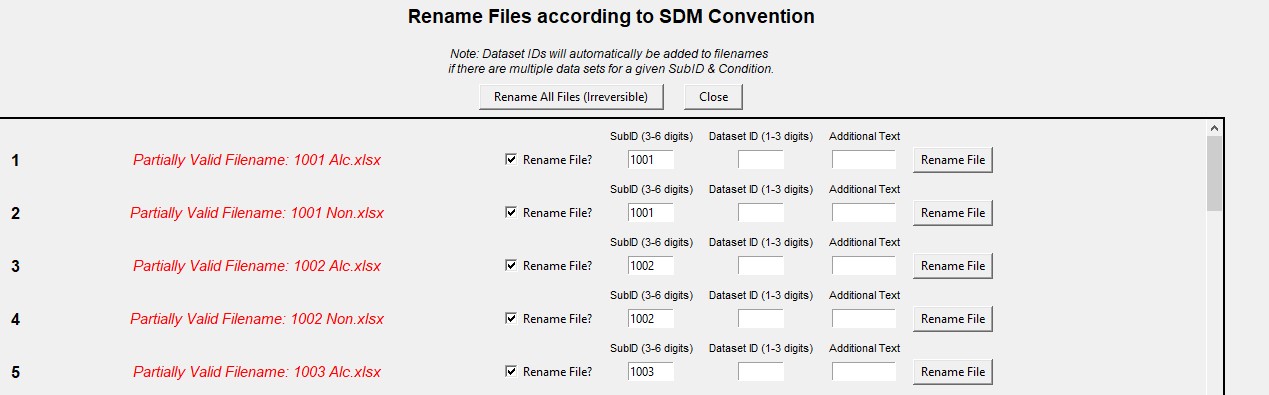
**Step 2**: Select data loading method.

* To conduct a quick test to make sure the software is setup and can be used on your computer, select the last option.
* To load a whole cohort of data, select the top option. After SDM processes a whole cohort, it will export the associated data and/or models within a .pickle file.
  + If you split a file using the File Splitting Tool, select the Split\_Files folder where all the separated data sets exist.
* To load only a single file, select the second option.
* The third option is for loading a .sdm file, i.e., an already-processed cohort that was generated previously by the SDM.

****

**Step 3**: Select SDM program. Depending on the data loading method, SDM may display different settings or options.

**Step 4.** Fill in Cohort Name and Select Data.



If filename(s) of the selected data are not compatible with SDM, you will be prompted to correct the filenames using SDM’s built-in file renaming tool.

**Step 5.** Select metadata. If no metadata file has been created, click “Create Metadata File” under File.

**Step 6**. Click “Submit”

**Step 7**. Customize settings as needed. If unsure, use defaults. When ready, click “Launch SDM”

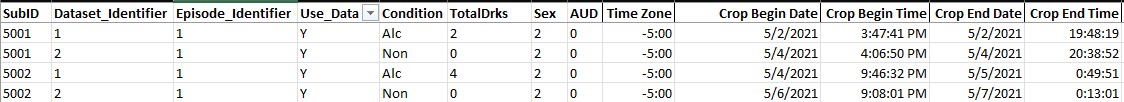
*Note: The SDM window will freeze when the program is running. Training models (the third program option) can take more than an hour on a standard computer.*

**A screenshot of a computer

Description automatically generatedInputs**

*The Inputs folder includes Excel files of Metadata, Skyn Data, and other Excel documents that may be helpful to have on hand.*

* The Skyn\_Data subfolder is designated to house folders of raw Skyn data. Each folder stored within Skyn\_Data should contain only .csv or .xlsx files.
* The Metadata subfolder is used to store metadata files that provide configuration information about a respective cohort of Skyn data.
  + A metadata file is required to run SDM.
  + Column headers must include SubID, Condition, Dataset\_Identifier, Episode\_Identifier, Use\_Data, and TotalDrks
  + This file can be created manually in Excel or you can use the SDM “**Create Metadata**” which is found in the Tools header
  + To exclude a dataset from analyses, label the row with “N” in the Use\_Data column.
  + Optional: Fill in TotalDrks column to indicate the # of drinks consumed for that episode. AUD, sex, are also optional. Any additional columns can be added to the right, as needed.
  + Optional: If you have timestamps to indicate the beginning/end of drinking episodes, fill in the columns Time Zone, Crop Begin Date, Crop Begin Time, Crop End Date, and Crop End Time. Use the following formatting:
    - Dates: month/day/year (5/25/2023)
    - Use military time (HH:MM:SS)

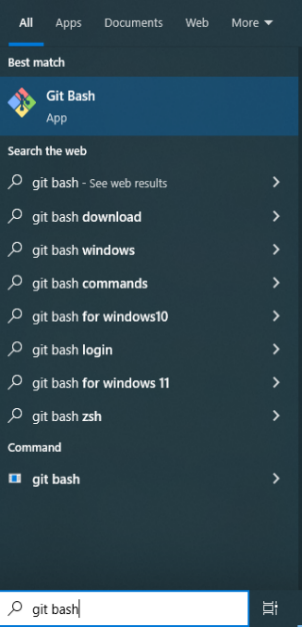
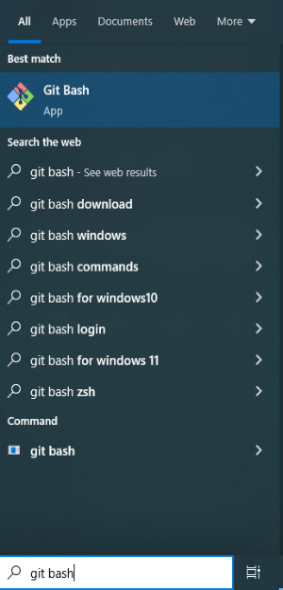


**Replicating Development Environment**

1. Download git:

* If on Windows, download from here [Git - Downloads (git-scm.com)](https://git-scm.com/downloads)
* If on Mac, enter these 3 commands into the terminal:
  + /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
  + brew update
  + brew install git

1. Download Anaconda: [Anaconda | The World’s Most Popular Data Science Platform](https://www.anaconda.com/)
   * This is a software package that includes Python so SDM can run. It also includes several IDEs such as VS Code – these are helpful if you would like to view or edit the code.



1. Open git bash (if on Windows) or terminal (if on Mac)
2. Navigate to the folder you would like to store the SDM:

Text

Description automatically generatedcd <path to folder>

1. Download SDM:

git clone <https://github.com/ndidier3/skyn_data_manager.git>

A screenshot of a computer screen

Description automatically generated

1. Check that folder looks like this:

A screenshot of a computer

Description automatically generated

* + App/ is where the software code exists
  + Raw/ is where raw Skyn data are stored
  + Resources/ include metadata and other files to help configure data
  + Results/ is where output files will be saved

Launch the app

* A blue background with white text

  Description automatically generatedOpen Anaconda Prompt (Windows) or terminal (Mac)
* Go to SDM folder with command:
  + *cd <path-to-folder>*

A black screen with white text

Description automatically generated

* **If first time using SDM,** run these below commands. Enter ‘y’ if commands have follow up questions. Some of these installs may take a few minutes.

*conda deactivate*

*conda create -n sdm-env python=3.8.8*

*conda activate sdm-env*

*conda install --file requirements.txt*

*conda install -c conda-forge kneed=0.7.0*

*conda install -c conda-forge scikit-learn=1.3.0*

* + - *this may take 10+ minutes, not because it takes up a lot of storage, but because Anaconda is creating compatibility between this package and all the other packages*



* **Launch SDM** using commands:

*conda activate sdm-env*

*python App/sdm\_user\_interface.py*