CLARITE GUI

Release 1.2.0

Mar 16, 2021

Contents

CLeaning to Analysis: Reproducibility-based Interface for Traits and Exposures

Motivation

CLARITE was created to provide an easy-to-use tool for analysis of traits and exposures.

This is a GUI developed to provide the same functionality, using the CLARITE python package.

Installation

1. Download or clone this repository and enter the folder

2. Ensure pipenv is installed

   ```
   pip install pipenv
   ```

3. Create/update the pipenv

   ```
   pipenv update
   ```

4. Run:

   ```
   pipenv run python main.py
   ```

Executables will be provided for future releases.

Citing CLARITE

If you use CLARITE in a scientific publication, please consider citing:

1. Lucas AM, et al (2019) CLARITE facilitates the quality control and analysis process for EWAS of metabolic-related traits. *Frontiers in Genetics*: 10, 1240

BibTeX entry:

```
@article{lucas2019clarite,
  title={CLARITE facilitates the quality control and analysis process for EWAS of␣
↪metabolic-related traits},
  author={Lucas, Anastasia M. and Palmiero, Nicole E. and McGuigan, John and Passero,␣
↪Kristin and Zhou, Jiayan and Orie, Deven and Ritchie, Marylyn D. and Hall, Molly A.}
↪,
  journal={Frontiers in Genetics},
  volume={10},
  pages={1240},
  year={2019},
  publisher={Frontiers},
  url={https://www.frontiersin.org/article/10.3389/fgene.2019.01240},
  doi={10.3389/fgene.2019.01240}
}
```

2. Passero K, et al (2020) Phenome-wide association studies on cardiovascular health and fatty acids considering phenotype quality control practices for epidemiological data. *Pacific Symposium on Biocomputing*: 25, 659

BibTeX entry:

```
@inproceedings{passero2020phenome,
  title={Phenome-wide association studies on cardiovascular health and fatty acids␣
↪considering phenotype quality control practices for epidemiological data.},
  author={Passero, Kristin and He, Xi and Zhou, Jiayan and Mueller-Myhsok, Bertram␣
↪and Kleber, Marcus E and Maerz, Winfried and Hall, Molly A},
  booktitle={Pacific Symposium on Biocomputing},
  volume={25},
  pages={659},
```

```
  year={2020},
  organization={World Scientific},
  URL={https://www.worldscientific.com/doi/abs/10.1142/9789811215636_0058},
  DOI={10.1142/9789811215636_0058}
}
```

Getting Started

4.1 Organization of Functions

CLARITE has many functions organized into several different modules:

Analyze Functions related to calculating EWAS results

Describe Functions used to gather information about data

Load Functions used to load data from different formats or sources

Modify Functions used to filter and/or modify data

Plot Functions that generate plots

Survey (not separated in the GUI) Functions and classes related to handling data with a complex survey design

4.2 Main parts of the UI

4.2.1 Data View

The data view includes a drop-down selector to move between different datasets. The currently selected dataset is shown. Variables are color-coded by their type (unknown, binary, categorical, or continuous) and are all continuous by default until the *categorize* function is used or they are converted via the right-click menu.

4.2.2 Command Bar

Each module (other than Survey) has a command tab, with buttons corresponding to each function in that module. The button for each function will open a unique dialog with all of the available parameters for that function. Further documentation on each function can be found in the CLARITE-Python Documentation.

Functions will (if applicable) use the currently selected dataset as input. Some functions may not be enabled if the currently selected dataset is not a compatible input, or if multiple datasets are required. For example, the manhattan plot function can only be used if the currently selected dataset is EWAS results.
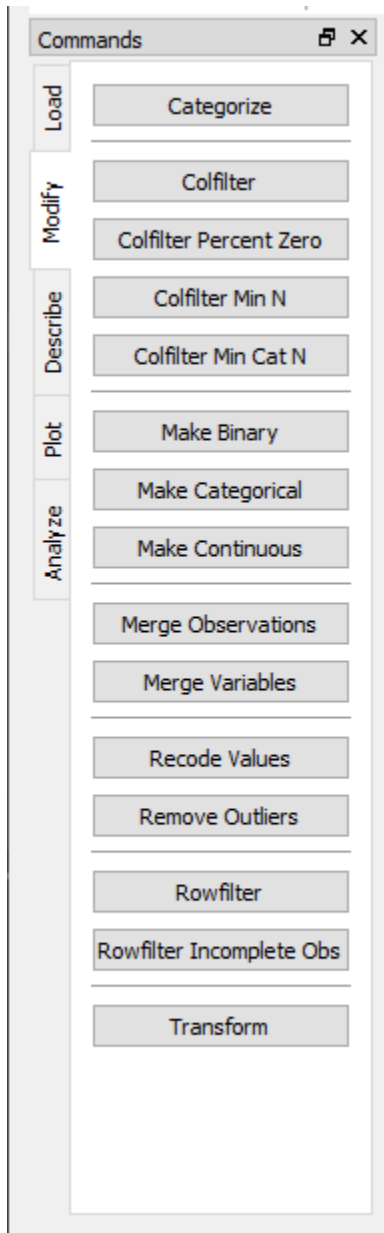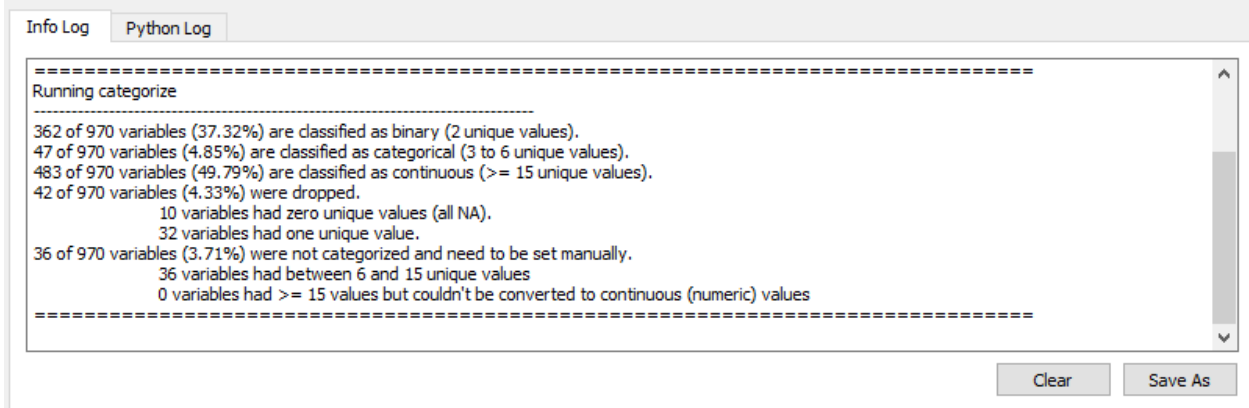
The command bar can be shown/hidden and docked/undocked via the "View" menu.
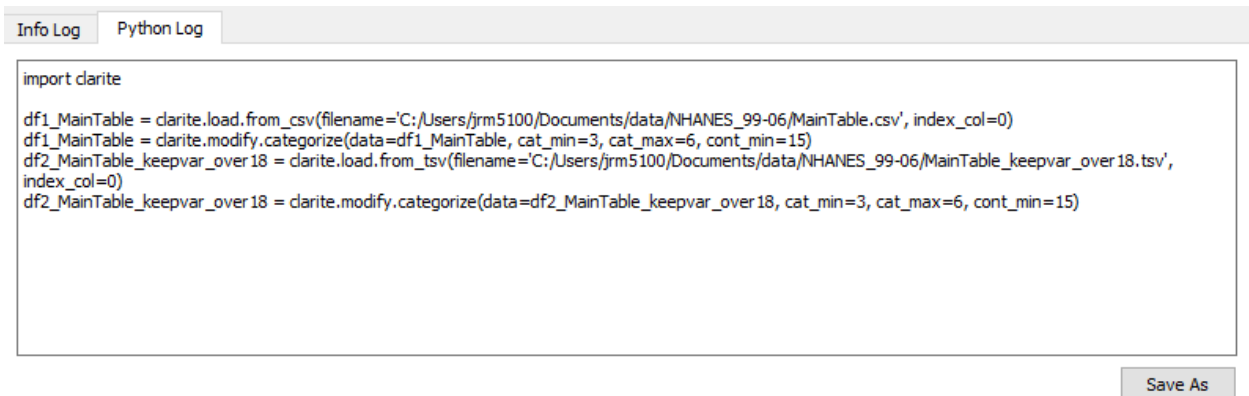
4.2.3 Logs

Info Log

The info log records the functions that are used, and any relevant logging information. It may be saved to a text file and/or cleared.

```
==================================================================================
Running categorize
----------------------------------------------------------------
362 of 970 variables (37.32%) are classified as binary (2 unique values).
47 of 970 variables (4.85%) are classified as categorical (3 to 6 unique values).
483 of 970 variables (49.79%) are classified as continuous (>= 15 unique values).
42 of 970 variables (4.33%) were dropped.
              10 variables had zero unique values (all NA).
              32 variables had one unique value.
36 of 970 variables (3.71%) were not categorized and need to be set manually.
              36 variables had between 6 and 15 unique values
              0 variables had >= 15 values but couldn't be converted to continuous (numeric) values
==================================================================================
```

Python Log

The equivalent python command for each function is recorded in the Python Log. This file can be saved as a python script which should repeat the analysis as it was performed in the GUI, as long as the location if input files doesn't change.

```
import clarite

df1_MainTable = clarite.load.from_csv(filename='C:/Users/jrm5100/Documents/data/NHANES_99-06/MainTable.csv', index_col=0)
df1_MainTable = clarite.modify.categorize(data=df1_MainTable, cat_min=3, cat_max=6, cont_min=15)
df2_MainTable_keepvar_over18 = clarite.load.from_tsv(filename='C:/Users/jrm5100/Documents/data/NHANES_99-06/MainTable_keepvar_over18.tsv',
index_col=0)
df2_MainTable_keepvar_over18 = clarite.modify.categorize(data=df2_MainTable_keepvar_over18, cat_min=3, cat_max=6, cont_min=15)
```

4.2.4 Preferences

Various display settings can be adjusted in the Preferences dialog (found in the *Edit* section of the menu).

Additional Notes
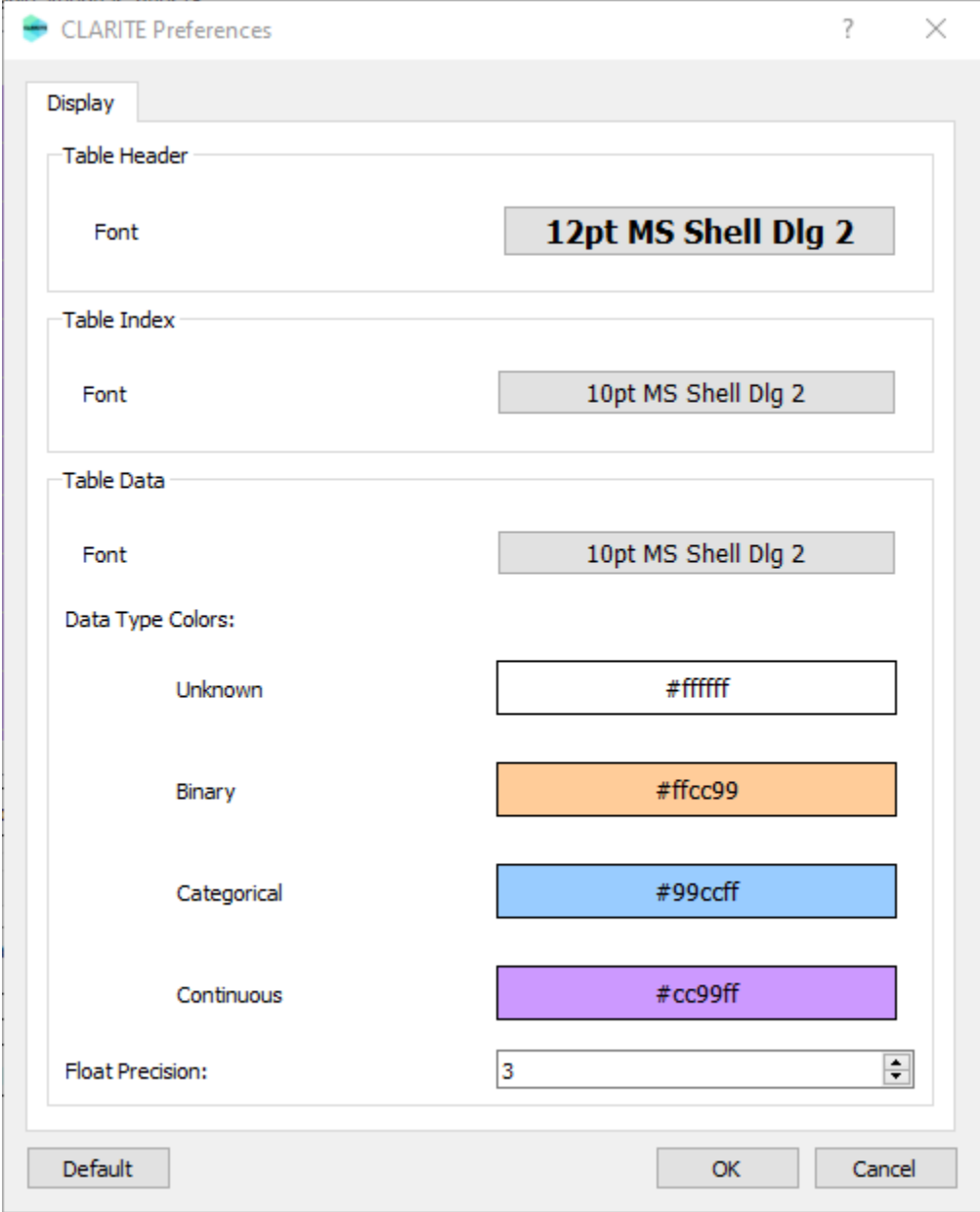
Release History, etc

5.1 Release History

5.1.1 v1.0.0 (2019-11-25)

Initial Release using CLARITE v0.9.1