YouML

INTRODUCTION

I'm Chongya Song, a Ph.D. interested in data science (esp., machine learning).

The macOS application resides in this repository (i.e., YouML) intends to provide the community a free and no-code toolkit for preprocessing data and building machine learning models. Several key features will be released after no major bug can be found in the current version. The ultimate goal is to deliver a platform where users can obtain solutions to address tough problems in their machine learning tasks. Accordingly, YouML stands for "YOUr (free) Machine Learning (platform)"

The number of algorithms for each task:
Data Cleaning: 3
Feature Selection: 3
Data Preprocessing: 35 + unlimited & customized SQL queries
Splitting: 4 + unlimited & customized SQL queries
Machine Learning: 50 (23 for classification and 27 for regression)

PERFORMANCE

Mainstream data science libraries are employed by YouML to ensure high-efficient manipulations, they are:

- Data Import: Pandas, SQLite
- Data Visualization: Matplotlib
- Feature Selection: Scikit-learn
- Feature Engineering: Scikit-learn, SQLite, Numpy, Scipy, blist
- Machine Learning Models: Scikit-learn, Intel Extension for Scikit-learn

TARGET GROUP

YouML is designed for entry-to-mid level users (prior knowledge of statistics is desirable), sophisticated users could purchase cloud-based machine learning products from tech giants (e.g., Microsoft, Google, Amazon) if they need such an automation toolkit.

Entry-level

- data science learners (e.g., students)
- programmers who intend to build and incorporate machine learning models into their programs

Mid-level

- users who intend to utilize YouML as a data preprocessing toolkit and feed the preprocessed data into their model scripts
- users who expect to pinpoint top-N machine learning models in terms of accuracy and/or efficiency (so that they can focus on N models when composing their model scripts)

- small businesses that eager to gain insights from their data

PLAN

A preliminary technical solution to accomplish the aforementioned ultimate goal has been determined, which also includes Windows and Linux versions. I will make every effort and try to look for others to ensure on-schedule completion. What I pursue is to create an application that is/with (a):

- Toolkit & Solution hub
- Script-level flexibility
- No-code & Low learning curve
- Free

CONTACT

As a personal project, YouML has never been tested publicly, so I'm eager to hear about your experiences. Bug reports, feature requests, questions and suggestions are very welcomed.

Author: Chongya Song
Email: schongy523@gmail.com
Profile: https://www.linkedin.com/in/chongya-song/

INSTALLATION

YouML adopts the same installation approach as common Python packages (i.e., run a setup.py script), so experienced users may skip this section and directly install YouML in their preferred manners.

Downloading and uncompressing the file named "YouML" in the repository. YouML runs on top of nearly 50 dependencies which can be installed by two package managers:

1. Conda (recommended)
2. Pip3

Although it is easier to install dependencies using pip3, I still recommend beginners to install everything through conda due to the following reasons:

1. PyPI (i.e., Python official package repository) is currently inaccessible via pip3 due to security threats. Consequently, you have to find alternative tools or search & install dependencies manually if there is something wrong with dependencies (e.g., version conflict).
2. Pip3 installs dependencies in a serial and recursive way, so no effort is made to ensure the compatibilities among dependencies are fulfilled simultaneously.

3. The prerequisite of using pip3 is to install a Python3 interpreter on your local machine. If the version of the installed Python3 interpreter is lower than 3.7.11, YouML is unable to drop features/columns via SQL queries and you have to search & install a new one to fulfill this requirement. Different interpreters reside in the same environment may result in conflicts and/or confuse you when using.

Instruction (conda)

1. Installing Anaconda or Miniconda (recommended) by following the instruction below: (beginners could download and use any installer to simplify the installation)
   https://docs.conda.io/projects/conda/en/latest/user-guide/install/macos.html

2. Opening a command prompt/terminal and navigating to the uncompressed folder YouML.

3. Executing command: "conda config --append channels conda-forge"
   This command enables your conda to download dependencies from conda-forge repository because a few dependencies and/or specific versions are not available in the default repository.

4. Executing command: "conda create --name YouML --file Dependency.txt --yes"
   This command creates a new conda virtual environment called "YouML" (case-insensitive) in which all dependencies are installed.

5. Executing command: "conda activate YouML"
   This command brings you into the conda virtual environment YouML.

6. Executing command: "python3 setup_conda.py install"
   This command installs YouML into the conda virtual environment, which can be launched by command: "YouML" (case-insensitive). Reminder: the prerequisite of launching YouML in this manner is to enter the conda virtual environment YouML (i.e., step no.5).

Instruction (pip)

1. Installing a Python3 interpreter with a version of 3.7.11 or later (pip is included by default).
   https://www.python.org/downloads/

2. Opening a command prompt/terminal and navigating to the uncompressed folder YouML.

3. Executing command: "python3 setup_pip.py install"
   This command installs YouML and the associated dependencies on your machine.

Optional Instruction (highly recommended)

Executing command: "which YouML | xargs -I {} cp {} /Applications"
This command creates a copy of YouML executable in folder /Applications. Now, you can drag & drop it anywhere (e.g., dock) and open it by clicking.

USAGE

1. Auto-save
   YouML is able to track and save your progress automatically, so there is no save button and data will not loss unless YouML is quit forcibly.

2. Terminating redundant plotting and useless data loading
   The employed plotting library Matplotlib is not designed for real-time display, but for generating publication-quality figures. By default, a small figure associated with each feature is produced and filled into a table at the bottom after each preprocessing manipulation. As a result, YouML may take more than 10 seconds to draw all figures if the numbers of samples and features exceed 50,000 and 50, simultaneously. It is the fact that the small figures are not that informative due to the limitation of the size. Therefore, YouML also visualizes each feature in separate 6X-larger widgets and doesn't generate these redundant (i.e., small) figures when the product of #samples and #features is greater than 2.5e6 (i.e., 50,000 times 50). The 6X-larger figures are publication-quality (i.e., ppi = 300) and are generated within a few seconds even if the number of samples exceeds 1e5.

   Furthermore, samples are also loaded into the table after each preprocessing manipulation. However, users would refer to summary and statistical information instead of specific samples when preprocessing big data in practice. Consequently, YouML allows users to manually turn off the data loading and the figure plotting features.

3. Adaptive plotting algorithm
   You may find some of the small figures are different from the corresponding large ones. This is not caused by a bug, but result from a built-in adaptive plotting algorithm. In short, the algorithm is able to mine data pattern from various perspectives by adjusting 6 decoupled parameters (will be available in YouML). As a result, users may discover more valuable patterns, generate more informative data and build more accurate models.

   In addition, if the number of unique target values doesn't exceed 20 (configurable in future versions), the large figures are colorful and each color represents one value, otherwise, the large figures are plain. This is due to a fact that users may not gain useful information from complicated figures.

   On the other hand, the labels on each axis are separately replaced by unique letters if the number of labels exceeds 26 or the conjunction of all labels is longer than the corresponding axis. The mapping relation between labels and letter can be found in a list adjacent to figures.