HandyML v1.1.7

User & installation guide

This document serves as user and installation guide of HandyML solution

Table of contents

Table of figures 1
1. Installation guide 2
1.1. Windows compatibility 2
1.2. UiPath version 2
1.3. Python 2
1.3.1. Python packages 2
1.4. Environment variables 2
2. User guide 3
2.1. HandyML_Trainer 3
2.1.1. Specify data location information 3
2.1.2. Specify which columns should be considered as features 4
2.1.3. Specify which column should be considered as target 4
2.1.4. Specify the algorithm to use and its parameters 5
2.1.5. Review your configuration 6
2.1.6. Analyze the results of your training 7
2.2. HandyML_Predictor 8
2.2.1. Specify template location 8
3. Resources 9
4. Contact 9

Table of figures

Figure 1 - HandyML Trainer / 1. Where is your data? 3
Figure 2 - HandyML Trainer / 2. Select your features 4
Figure 3 - HandyML Trainer / 3. Select your target 5
Figure 4 - HandyML Trainer / 4. Select your algorithm 6
Figure 5 - HandyML Trainer / 5. Summary 7
Figure 6 - HandyML Trainer / Example of notification raised during training 7
Figure 7 - HandyML Trainer / 6. Report 8
Figure 8 - HandyML Predictor / Path of template file 9
Figure 9 - HandyML Predictor / Example of notification raised during prediction process 9

1. Installation guide

1.1. Windows compatibility

This solution has been developed and tested under Windows 10.

1.2. UiPath version

This project has been developed under UiPath Studio 2019.4.3. So, you will need a Studio/Robot with version 2019.4.3+.

1.3. Python

This project uses Python 3.6.8+. It has been tested and developed on a x64 architecture. You can easily have a working Python 3.6 environment by installing Anaconda 2019.03 version (https://www.anaconda.com/distribution).

Then you will need to create an Anaconda environment specifying Python 3.6.8+ version by executing the following in Anaconda Prompt (replace [environment-name] by the name you want, eg. env-3.6):

conda create -n [environment-name] python=3.6

1.3.1. Python packages

You will also need to install the following packages using either conda (if Anaconda installed) or pip to be able to use this tool. If using Anaconda, launch Anaconda Prompt and execute following commands:

activate [environment-name]

pip install numpy pandas scikit-learn matplotlib scikit-plot

pip install win10toast (add options --user --ignore-installed when installed on a Windows 7 machine)

You'll find below the packages' versions used during development of the solution:

- NumPy 1.16.4
- Pandas 0.24.2
- Scikit-Learn 0.21.2
- Matplotlib 3.1.1
- Win10toast 0.9
- Scikit-Plot 0.3.7

1.4. Environment variables

In order for the UiPath Robot to know where Python is installed and on which architecture, you'll have to create 2 environment variables:

- HANDYML_PYTHON_PATH, indicating the folder on which python.exe file can be found

With Anaconda installed you can know where the python.exe file is by following this: https://docs.anaconda.com/anaconda/user-guide/tasks/integration/python-path/

- HANDYML_PYTHON_TARGET, indicating the machine architecture (should be x64 or x86)

2. User guide

2.1. HandyML_Trainer

HandyML_Trainer is a Front Office Robot allowing to train a supervised machine learning model through 6 different steps described below.

Before launching the process, you'll need an Excel file holding your training data with a specific format (please see examples folder within the solution):

- All data used for training should be in one sheet
- First line may be dedicated to headers
- Each line is an observation and can contain only strings, numbers (including floating point numbers) and Booleans

2.1.1. Specify data location information

The first step of the training process is to specify the following parameters:

1. Where the training data is located, you can type directly the absolute path of the file or use the 3 dots button on the right to select desired file (should be less than 1MB)
2. On which sheet by giving its name
3. And if the first sheet should be considered as headers

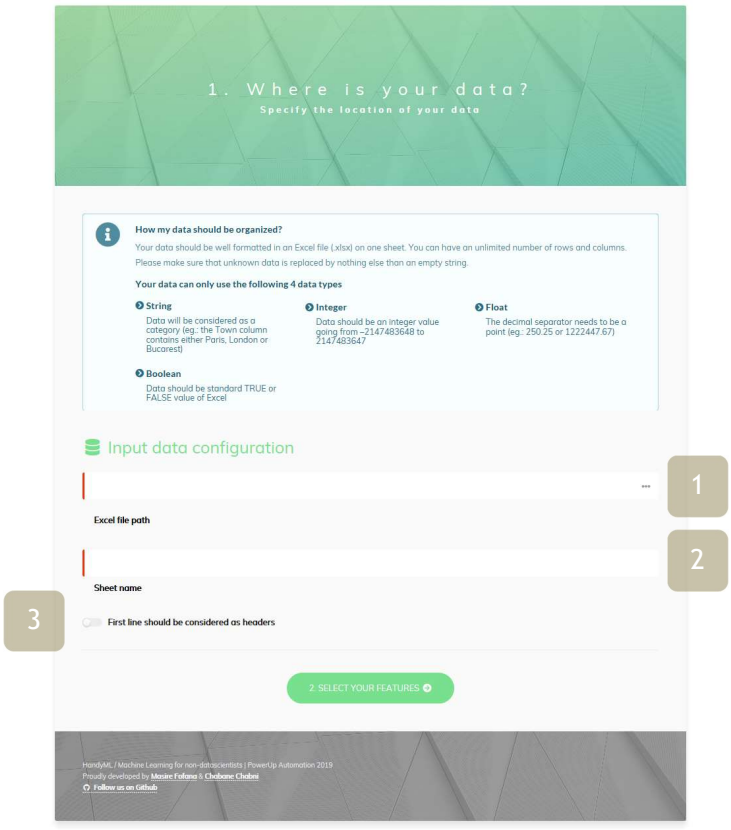

Figure 1 - HandyML Trainer / 1. Where is your data?

Then, you can go to next step by clicking on "2. Select your features" button.

2.1.2. Specify which columns should be considered as features

The second step of the training process is to select the columns in your training data that you want to flag as features. You must select at least one column and at most the number of columns minus one (because one column should be the target).

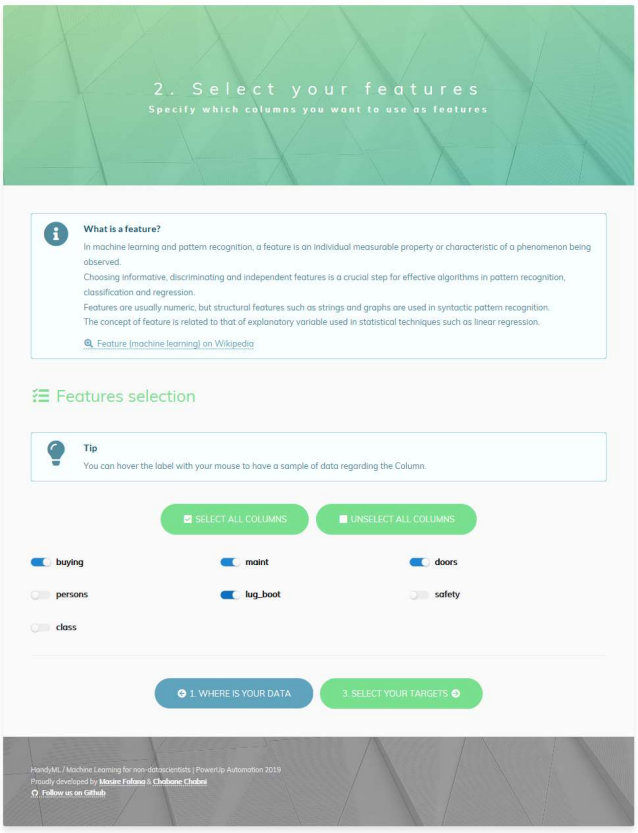

Figure 2 - HandyML Trainer / 2. Select your features

Then, you can go to next step by clicking on “3. Select your targets” button or go back to the previous step by clicking on “1. Where is your data” button.

2.1.3. Specify which column should be considered as target

The third step of the training process consists in selecting the column in your training data that you want to flag as target. It works the same way as for the features unless you must select only one column.

Note that the application will automatically hide columns already selected as features.

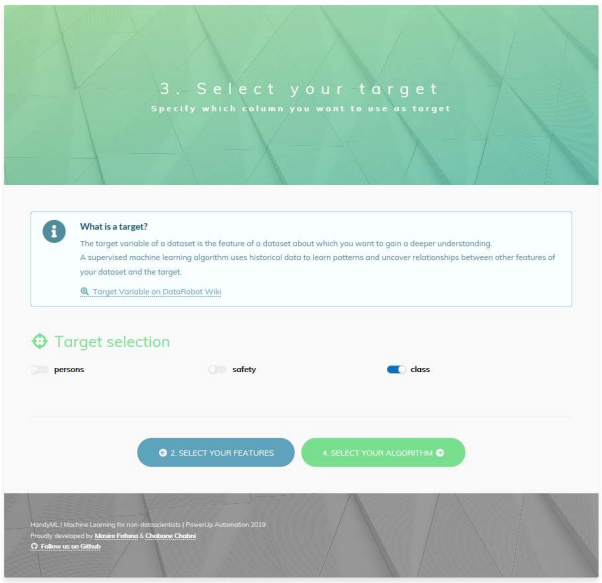

Figure 3 - HandyML Trainer / 3. Select your target

Then, you can go to next step by clicking on “4. Select your algorithm” button or go back to the previous step by clicking on “2. Select your features” button.

2.1.4. Specify the algorithm to use and its parameters

The fourth step of the training process is to select the algorithm you want to use for the machine learning model training according to the problem you want to solve:

- Regression (continuous value)
- Classification (categorical value)

Depending on the algorithm you have chosen, you may have additional parameters to specify. For example, if you choose “Random Forest Classification”, you’ll have to specify “Criterion” & “N_Estimators” parameters.

Note that all parameters come with a default value.

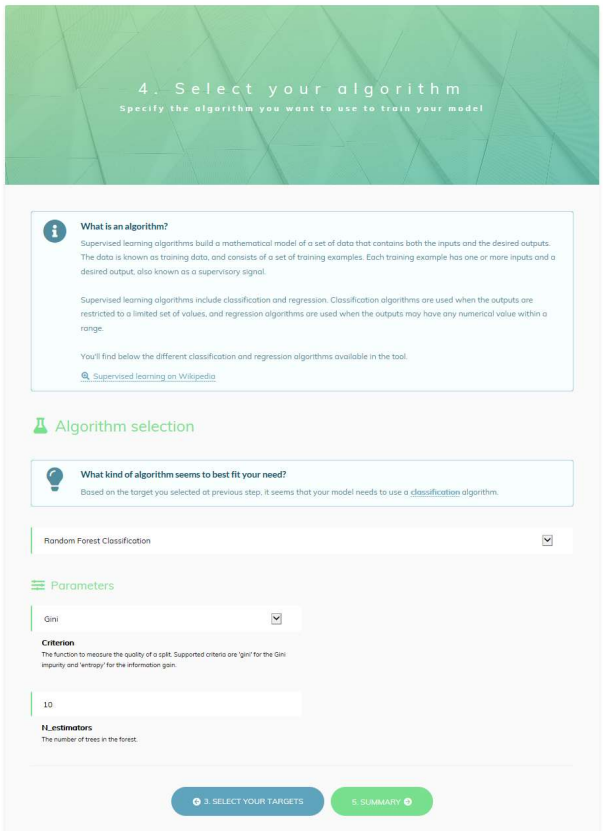

Figure 4 - HandyML Trainer / 4. Select your algorithm

Then, you can go to next step by clicking on “5. Summary” button or go back to the previous step by clicking on “3. Select your targets” button.

2.1.5. Review your configuration

The fifth step allows you to review your configuration and change it if necessary by using the “previous step” buttons. If everything is OK, you can start the machine learning model training process by clicking on “Train your model” button.

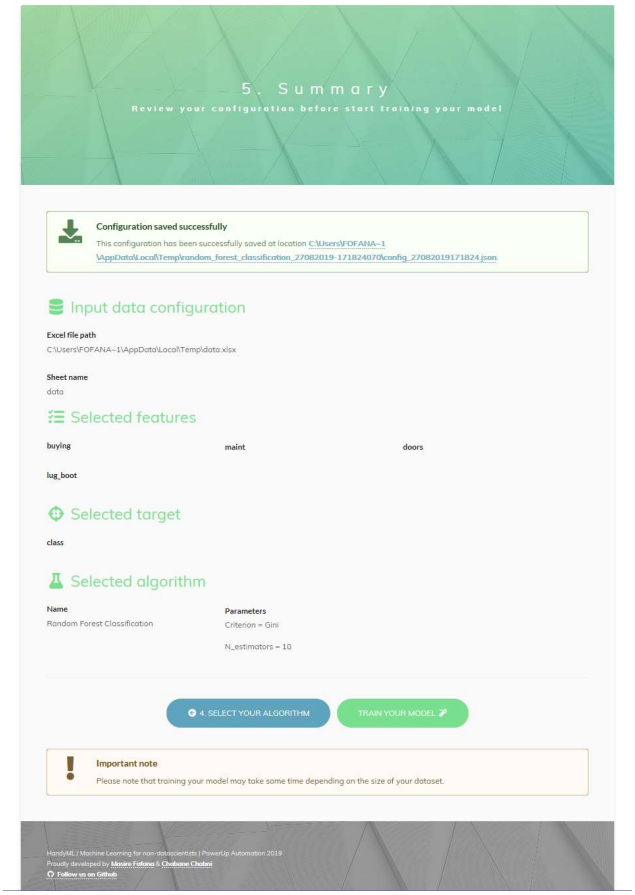

Figure 5 - HandyML Trainer / 5. Summary

During the training, you'll observe small notifications indicating the progress, it may take some time if you have a lot of observations in your training data.

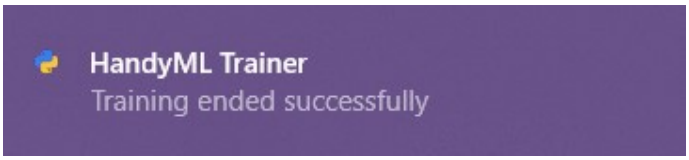

Figure 6 - HandyML Trainer / Example of notification raised during training

2.1.6. Analyze the results of your training

The sixth and last step of the training process consists in analyzing provided results (score, duration, plots, etc.) and generating a template for HandyML_Predictor to be able to use the newly created model to make predictions on new data. You can achieve that by clicking on "Generate template" button.

In case of any error during training, you'll have a box at the top (replacing the "Model saved successfully" box) displaying all encountered errors to help you fix them.

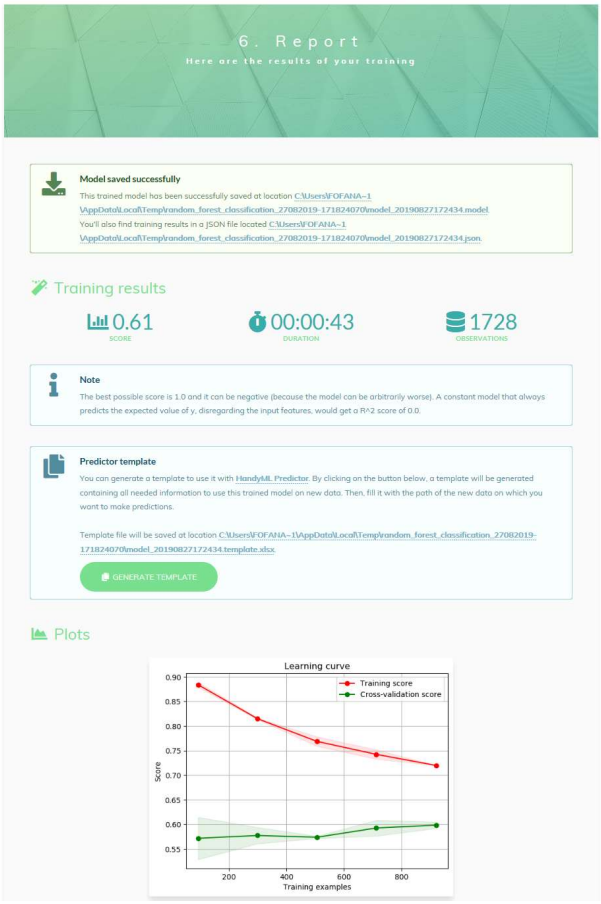

Figure 7 - HandyML Trainer / 6. Report

2.2. HandyML_Predictor

HandyML_Predictor is a Front Office Robot allowing to make predictions on new data using the template (Excel file) produced using HandyML_Trainer.

The template file must contain new data on “data” sheet respecting strictly the same format as the one used for training.

The first sheet of this template file is used to hold information about the generated model, do not edit this sheet unless you know what you’re doing.

2.2.1. Specify template location

The first thing you need to do is to provide the absolute path of the template file containing all information necessary to make predictions.

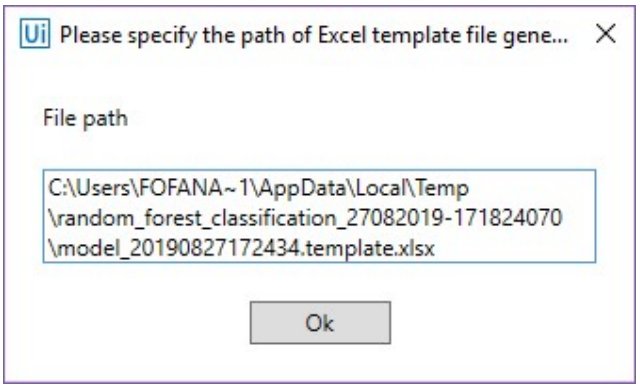

Figure 8 - HandyML Predictor / Path of template file

Then, click on “OK” button to start the prediction process.

During the process, you’ll observe small notifications indicating the progress, it may take some time if you have a lot of predictions to make.

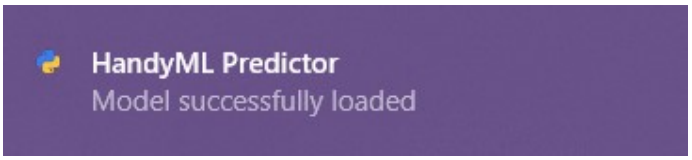

Figure 9 - HandyML Predictor / Example of notification raised during prediction process

In case of any error, the root cause is reported in the template file on the first sheet to help you fix it.

3. Resources

STEP-BY-STEP SCREENCAST OF THE SOLUTION: https://youtu.be/hiwNWctMc9Y

GITHUB REPOSITORY: https://github.com/masiref/HandyML

UIPATH CONNECT PROJECT PAGE: https://connect.uipath.com/community/project/pua-virtual-handyml

DEVPOST PROJECT PAGE: https://devpost.com/software/machine-learning-for-non-datascientists

4. Contact

Let us know if you have any issues. Please send an email to following addresses, we’ll give you an answer as soon as possible.

- masire.fofana@natixis.com
- chabnichab@eisti.eu