1. From Serial\_Comm code, select TARGET (SBP, DBP, MAP) for estimation. This is essential in data preparation step to get the test data & labels accordingly (and to be shuffled w.r.t that target).
2. Run “Data Preparation ...” part of the code. This will load CSV attributes and labels, remove outliers, shuffle it, and divide it between train & test (but we are interested in the test part only at this stage)
3. Select Algorithm (Alg) .. this does not affect the Python code, but it is important for saving files with proper naming.
4. Edit text files that has the generated C codes (previously saved from Python with m2gen). Open them using Note++ and edit them according to the features as follows:

input[0], input[1], input[2], input[3] ----- replaced by -----> PTTh, PTTm, PTT, HR

1. Add those codes to the prediction function part in the Arduino IDE.
2. Turn off/on the MCUs so that they are ready for a new code upload. Upload the codes with proper PORT & BOARD selection, as explained in the other document: (Arduino: Arduino Uno, COM6, ESP: ESP32 Wrover Module, COM9, PyBadge: Adafruit pyBadge M4 Express (SAMD51, COM7)). Add the if condition in all files, it makes PyBadge work properly and does not affect other boards  
   #if defined(ARDUINO\_SAMD\_ZERO) && defined(SERIAL\_PORT\_USBVIRTUAL)

// Required for Serial on Zero based boards

#define Serial SERIAL\_PORT\_USBVIRTUAL

#endif

1. Arduino & PyBadge can work directly after uploading the code. For ESP, click Boot.

When loading each code, copy Memory information from the IDE.

1. Now, with the MCUs working, Run “PC-to\_MCU” part of the code so that PC sends data to the MCUs and get prediction. You can run the 3 Jupyter notebooks in parallel since MCUs are connected to different ports.   
   Note: for PyBadge, Comment the WaitforArduino() fn from the Python code.
2. Got predictions? now run “Error Calc” part to evaluate accuracy, and save variables, graphs, and other results.

Change “Alg” when moving to another algorithm within the same target. If target changes, change TARGET and re-run the data preparation step.