Using MATLAB functions in Docker using Python

Compiling the MATLAB functions

In order to compile the functions into a Python package make sure “MATLAB Compiler” and “MATLAB Compiler SDK” add-ons are installed in your MATLAB environment.

Run the ‘libraryCompiler’ command in MATLAB command windows

Under “Type” choose “Python package” and under “Exported functions” choose the function files you wish to add.

Enter the library name (which will be the Python package name eventually)

Click “Package” and choose where to save the package files

Copy the folder “for_redistribution_files_only” to your source code folder, this folder contains the Python package we will install.

Note: Only works in specific Python versions, for me 3.6 worked great.

Importing the package

Inside the “for_redistribution_files_only” folder you will find “setup.py” file, we will use it to install the compiled package in the following method:

Using the terminal, navigate to the “for_redistribution_files_only” folder, and run:

```
python setup.py install --user
```

Make sure your Python version is less than 3.7 (run `python --version`).

Now the compiled package is installed in your Python environment, you can import the package using the same name you specified in MATLAB libraryCompiler.

Using the functions while initializing MATLAB runtime

In order to use the compiled function from MATLAB we need “MATLAB Runtime” to be installed on the host machine (where you are running the code, e.g, Docker container).

Follow this link in order to install the MATLAB Runtime.

After installing the MATLAB Runtime (already installed in the docker image i provided) and installing the Python package we need to initialize the MATLAB Runtime before using the functions.

For example, if the package name is “aimlab” we will do the following:

```
Import aimlab
aimlab_runtime = aimlab.initialize()
aimlab_runtime.some_matlab_written_function()
```

Useful links

Install a MATLAB Compiler SDK Python Package
Install MATLAB Engine API for Python
Import Compiled Python Packages
Initialize the MATLAB Runtime

Dockerfile

```
FROM python:3.6-buster

ENV DEBIAN_FRONTEND noninteractive
RUN apt-get -q update && \
    apt-get install -q -y --no-install-recommends \
        xorg \
      unzip \
      wget \
      curl && \
    apt-get clean && \
    rm -rf /var/lib/apt/lists/*

# Install the MCR dependencies and some things we'll need and download the MCR
# from Mathworks -silently install it
RUN mkdir /mcr-install && \
    mkdir /opt/mcr && \
    cd /mcr-install && \
    wget -q
https://ssd.mathworks.com/supportfiles/downloads/R2020a/Release/4/deployment_fi
les/installer/complete/glnxa64/MATLAB_Runtime_R2020a_Update_4_glnxa64.zip && \
    unzip -q MATLAB_Runtime_R2020a_Update_4_glnxa64.zip && \
    rm -f MATLAB_Runtime_R2020a_Update_4_glnxa64.zip && \
    ./install -destinationFolder /opt/mcr -agreeToLicense yes -mode silent && \
    rm -rf /mcr-install

# Configure environment variables for MCR
ENV LD_LIBRARY_PATH
/opt/mcr/v98/runtime/glnxa64:/opt/mcr/v98/bin/glnxa64:/opt/mcr/v98/sys/os/glnxa
64:/opt/mcr/v98/sys/opengl/lib/glnxa64
ENV XAPPLRESDIR /opt/mcr/v98/X11/app-defaults

ENV PATH /opt/mcr/v98/bin:$PATH

COPY . /data
WORKDIR /data/for_redistribution_files_only
RUN python setup.py install
```