

環境構築手順

勉強会に必要な環境構築手順と必要なライブラリについてドキュメントを残す。

今回の勉強会で使用されるGithubのレポジトリ： <https://github.com/shibuiwilliam/ml-system-in-actions>

前提条件

- ・ pyenvのインストールが完了している。(その1が完了していることを前提とする)

目的

Chapter2を動かすための以下のライブラリのインストール

- ・ Anaconda
- ・ Docker

手順

Githubで指定されている手順通り、Anacondaとライブラリのインストールは以下の手順でインストール。

```
# 2021/02時点最新のanaconda環境を選択
$ pyenv install anaconda3-5.3.1
Downloading Anaconda3-5.3.1-MacOSX-x86_64.sh.sh...
-> https://repo.continuum.io/archive/Anaconda3-5.3.1-MacOSX-x86_64.sh
Installing Anaconda3-5.3.1-MacOSX-x86_64.sh...
Installed Anaconda3-5.3.1-MacOSX-x86_64.sh to ~/.pyenv/versions/anaconda3-5.3.1
```

これでpyenvによるAnaocndaのインストールができています。以降の各フォルダでの作業については"local ~"で環境を指定する。

例) Chapter2のcifar10で環境をセットするときには、以下のようにする。

```
# 環境をセットする。
# まずはcdでディレクトリの変更
cd ./chapter2_training/cifar10

# pyenvで仮想環境にanaconda3-2020.02を選択
$ pyenv local anaconda3-5.3.1

# 仮想環境がanaconda3-2020.02になっていることを確認
$ pyenv versions
```

```
system
* anaconda3-5.3.1
```

この操作についてまとめると、以下のように実行される。

```
ubuntu@ip-172-31-35-241:~/tmp/ml-system-in-actions$ cd /home/ubuntu/tmp/ml-system-in-actions/chapter2_training/cifar10
ubuntu@ip-172-31-35-241:~/tmp/ml-system-in-actions/chapter2_training/cifar10$ pyenv local anaconda3-5.3.1
ubuntu@ip-172-31-35-241:~/tmp/ml-system-in-actions/chapter2_training/cifar10$ pyenv versions
system
* anaconda3-5.3.1 (set by /home/ubuntu/tmp/ml-system-in-actions/chapter2_training/cifar10/.python-version)
ubuntu@ip-172-31-35-241:~/tmp/ml-system-in-actions/chapter2_training/cifar10$
```

```
# 依存ライブラリをインストール
# requirements.txtの中に指定されている環境をすべてインストール
$ pip install --upgrade pip

# 今回はpip install -r ~/.txtで上手くいかない場合は、--no-cache-dirを試してみる。pipにはキャッシュディレクトリを消すコマンドが含まれていないので、オプションを付けてcacheを作らないようにインストールを行う。

#$ pip install -r requirements.txt
$ pip install --no-cache-dir -r requirements.txt

# これについては、make devコマンドに対応する、makefileの編集が必要。

# mlflowをインストール
$ pip install mlflow
```

Dockerのインストールについて

まずはインストール前に、Dockerが既にインストールされているケースを想定して、古いバージョンのものを破棄する。

```
sudo apt-get remove docker docker-engine docker.io containerd runc
```

次に、Docker Engineに必要なパッケージのインストールを行う。

```
sudo apt-get update
sudo apt-get install \
    ca-certificates \
    curl \
    gnupg \
    lsb-release
```

Dockerの公式サイトからパッケージをインストールする。それに先立って、オフィシャルサイトのGPGキーをダウンロードし、登録。

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

Dockerダウンロードサイトからaptレポジトリに追加。

```
echo \  
"deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-  
archive-keyring.gpg] https://download.docker.com/linux/ubuntu \  
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list >  
/dev/null
```

Dockerのインストール

```
# パッケージのアップデート  
sudo apt-get update  
  
# Docker Engine他の一式のインストール  
sudo apt-get install docker-ce docker-ce-cli containerd.io
```

Dockerはバージョンを指定してインストール可能。詳細については公式HP参照のこと。

グループの設定

このままだと、Dockerのユーザーがルートユーザーのみになっているので、"ubuntu"ユーザーをdockerのグループに追加する。

```
# ログインしているユーザーをdockerグループへ追加。  
sudo gpasswd -a $(whoami) docker  
  
# 次にdocker.sock にグループでの書き込み権限を付与。  
  
$ sudo chgrp docker /var/run/docker.sock  
  
# 最後にdocker daemonを再起動します。  
$ sudo service docker restart
```

その他注意事項：

mlflowをインストール時、"PyYAML"をアンインストールできませんというエラーに遭遇するケースがある。これについては、pipのバージョンを8.1.1まで落として手動でアンインストールさせたのち、pipのバージョンを戻す対処が考えられる。 詳細についてはこちらのサイト参考のこと

https://note.com/junk_matsumoto/n/nc4f472ed0d80

参考文献：

Dockerのインストール時のマニュアル : <https://docs.docker.com/engine/install/ubuntu/>

Docker使用時のpermission denied errorに対応 :

<https://tech.librastudio.co.jp/entry/index.php/2018/07/14/post-1924/>