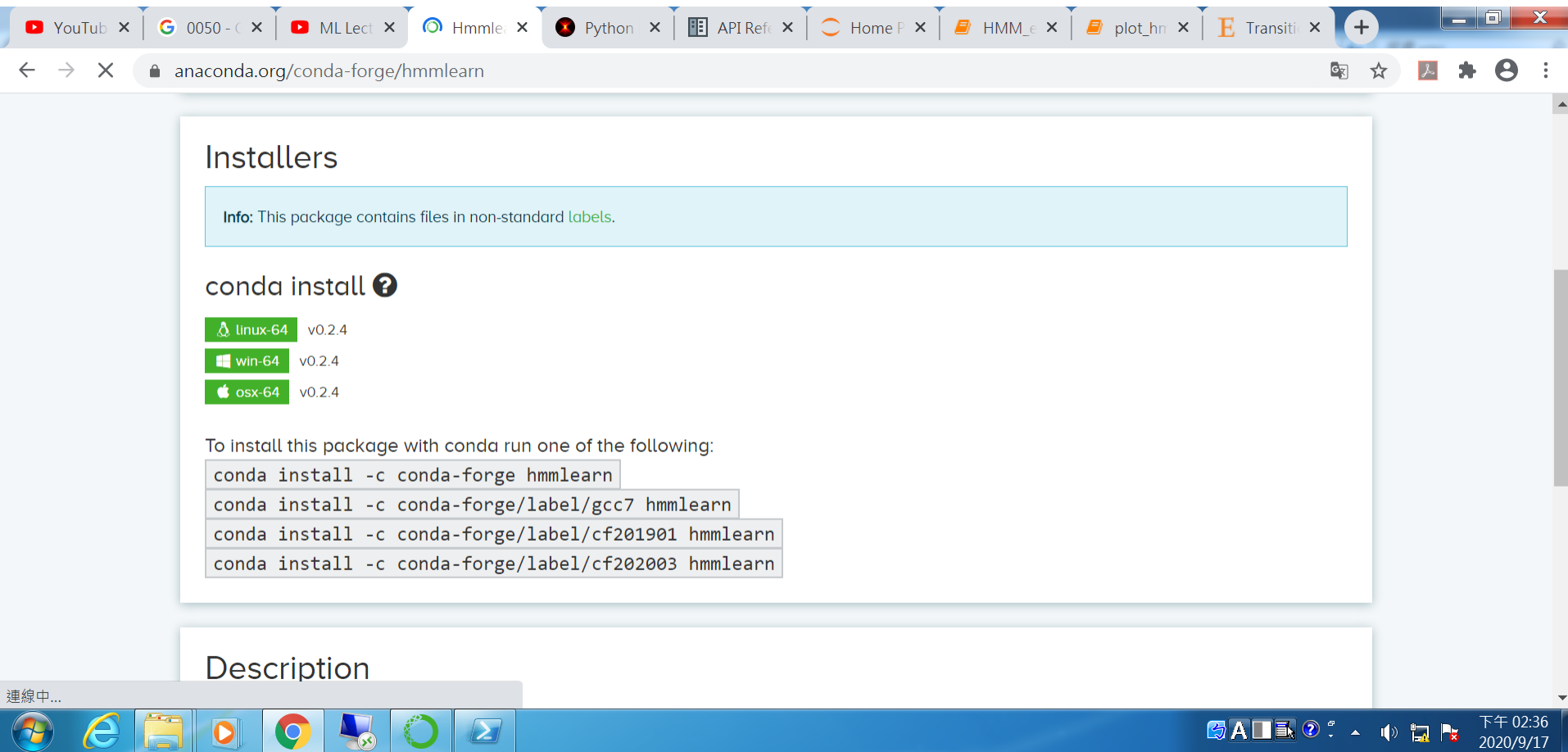


RUNNING HMM & DECISION TREE

Install HMM package

□ Try the following to see if it works



YouTube x 0050 - C x ML Lect x Hmmler x Python x API Refe x Home P x HMM_c x plot_hn x Transiti x

anaconda.org/conda-forge/hmmlearn

Installers

Info: This package contains files in non-standard labels.

conda install ?

- linux-64 v0.2.4
- win-64 v0.2.4
- osx-64 v0.2.4

To install this package with conda run one of the following:

```
conda install -c conda-forge hmmlearn
conda install -c conda-forge/label/gcc7 hmmlearn
conda install -c conda-forge/label/cf201901 hmmlearn
conda install -c conda-forge/label/cf202003 hmmlearn
```

Description

連線中...

下午 02:36
2020/9/17

Install HMM package

- If it does not work, need alternatives
- Figure out your OS (win32 or win64)
 - ▣ From setting -> system -> about (win 10)
- Figure out your Python version
 - ▣ Jupyter Notebook -> about
- Goto <https://www.lfd.uci.edu/~gohlke/pythonlibs/>
 - ▣ Download hmmlearn-0.24-cpxx-cpxxm-(win32 or win64) to **Downloads folder**
 - ▣ CP xx means python version, such as 3.8 in my computer

Install HMM package

- Use powershell prompt
 - ▣ Change working directory to Downloads
 - ▣ Type in the following after the CMD prompt
 - `pip install hmmlearn-0.2.4-cp38-cp38-win32.whl`
 - Use your actual file name here
 - Don't forget file name extension (.whl)
- If everything goes OK, you will see
 - ▣ Installing collected packages: hmmlearn
 - ▣ Successfully installed hmmlearn-0.2.4

Use HMM

- Use this line
 - ▣ `from hmmlearn import hmm`
- Check the document for usage
 - ▣ <https://hmmlearn.readthedocs.io/en/latest/>

Decision tree

- Scikit_learn has a library for decision tree
- However, there is no option to choose between ID3, C4.5, or CART
 - ▣ Use criterion='entropy' for ID3 (maybe, not fully confirmed)
 - ▣ Default is to use gini criteria, which leads to CART
 - ▣ Don't know how to invoke C4.5 (entropy ratio)

Decision tree

- Sklearn decision tree does NOT support **categorical** input data
 - ▣ Such as sunny, overcast, rainy
- One walk around is to use **one-hot encoding**
 - ▣ Sunny, overcast, rainy
 - ▣ 1,0,0 (for sunny)
 - ▣ 0,1,0 (for overcast)
 - ▣ 0,0,1 (for rainy)

Decision tree

- Is one-hot a good encoding approach
 - ▣ Hard to say, not very good
 - ▣ Attributes are **sparse**
 - ▣ Generated tree may not match what we get with categorical data
- How about numerical encoding
 - ▣ Sunny = 2, overcast = 1, rainy = 0
 - ▣ It means **ordering** (sunny > overcast > rainy)
 - ▣ Not truly correct