**In this tutorial, students will learn:**

* How to setup MLFlow: a support tool for managing the machine learning pipeline
* How to monitor the machine learning experiments with MLFlow
* How to deploy the obtained models

**I Setup MLFlow**

* Install Anaconda
  + Go to the Anaconda page:
    - <https://www.anaconda.com/distribution/>
  + Download Anaconda bash script
    - curl -O <https://repo.anaconda.com/archive/Anaconda3-2019.03-Linux-x86_64.sh>
  + Execute the script:
    - $ bash Anaconda3-2019.03-Linux-x86\_64.sh
* Install mlflow
  + $ pip install mlflow
* Install scikit-learn
  + $ pip install scikit-learn

**II Monitoring the machine learning experiments**

1. Example code:

https://www.mlflow.org/docs/latest/tutorials-and-examples/tutorial.html

1. Compare the results
   1. $ ./script\_of\_experiments
      1. Change the arguments of the training code
      2. Create a script
   2. $ mlflow ui
      1. Check and compare the results.
2. Packing the code in a virtual environment such as conda
   1. Create MLProject file

*# sklearn\_elasticnet\_wine/MLproject*

name: tutorial

conda\_env: conda.yaml

entry\_points:

main:

parameters:

alpha: float

l1\_ratio: {type: float, default: 0.1}

command: "python train.py {alpha} {l1\_ratio}"

* 1. Create conda.yaml

*# sklearn\_elasticnet\_wine/conda.yaml*

name: tutorial

channels:

- defaults

dependencies:

- numpy=1.14.3

- pandas=0.22.0

- scikit-learn=0.19.1

- pip:

- mlflow

**III Deploying the model and put it into product**

1. Go to the UI to check the saving model
2. Deploy the server using the saving model:
   * mlflow models serve -m /home/phuong/PycharmProjects/monitoring/tutorial2/examples/mlruns/0/79936866205949f0843a941829e59f0a/artifacts/model -p 1234
3. Using the deployed model to do prediction
   * curl -X POST -H "Content-Type:application/json; format=pandas-split" --data '{"columns":["alcohol", "chlorides", "citric acid", "ddity", "free sulfur dioxide", "pH", "residual sugar", "sulphates", "total sulfur dioxide", "volatile acidity"],"data":[[12.8, 0.029, 0.48, 0.98, 6.2, 29, 3.33, 1.2, 0.39, 75, 0.66]]}' <http://127.0.0.1:1234/invocations>