



# #18 - Handling Visualization of 'Many' Experiments

주제 : Handling Visualization of 'Many' Experiments


링크 :

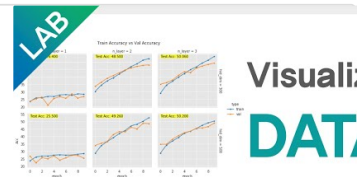
[#18.Lab] Handling Visualization of 'Many' Experiments - 딥러닝 홀로서기

실습자료 링크 : [https://github.com/heartcored98/Standalone-](https://github.com/heartcored98/Standalone-DeepLearning/blob/master/Lec4/Lab6_result_report.ipynb)

DeepLearning/blob/master/Lec4/Lab6\_result\_report.ipynb자료 저장소 링크 :

<https://github.com/heartcored98/Standa...>

 <https://youtu.be/7CMxgqvrDgg>



## 1. Assignment #2 Review

- How to save experiment results
  - 무엇을 저장해야 하는가?
    - Args (파라미터 세팅값)
    - Epoch에 따른 Train Loss, Val Loss, Train Acc, Val Acc
    - 최종 Train Acc, Val Acc, Test Acc도 저장해야 함
  - 어떻게 저장해야 하는가?
    - 각 실험 결과들을 Dictionary에 넣고 리스트에 하나씩 Append
    - 문제점
      - 만약 중간에 인터넷이 끊어져서 프로세스를 종료해야 하는 상황이 온다면?
      - 램에 올라가 있다면 지금까지의 실험 결과가 사라짐
    - 해결법 : 하드디스크에 저장하면 안전함 → JSON 포맷을 활용

```
import json
a = {'value1':5, 'value2':10, 'seq': [1,2,3,45]}
filename = 'test.json'

# with문으로 Write 권한으로 파일 오픈
# with를 사용하는 이유 : 기존에 열려있던 세션을 종료시키고 새로운 세션으로 오픈
with open(filename, 'w') as f:
    json.dump(a, f)

#Fold안의 파일 리스트와 test.json파일 내용 살펴보기
```

```
!ls
!cat test.json

# Read 권한으로 파일 오픈
with open(filename, 'r') as f:
    result = json.load(f)
    print(result)
```

- 파일이름은 뭘로 해야하는가?
  - 시간\_랜덤 숫자.Json? (e.g. 190128\_3242.json)
  - 근데 같은 실험 세팅으로 다시 돌리면 시간이 달라서 다른 Json파일이 또 생김
  - 그럼 변수 값들을 파일 제목에 같이 넣어볼까? (e.g. N\_Layer\_4\_hid\_dim\_100.json)
  - 문제점
    - 하이퍼파라미터가 너무 많아지면 최대 파일 이름 길이 초과
  - 해결법
    - 실험이름 + 실험 세팅 값의 해쉬.json으로 파일 이름을 정하자
    - 해쉬 함수 : 임의 길이의 문자열이 입력되었을 경우, 고정된 길이의 데이터로 매핑해주는 함수
  - 결과 파일 이름 : exp1-0a7972.json

```
import hashlib
setting = {'value1':6, 'value2':10, 'seq': [1,2,3,45], 'exp_name':'exp1'}

hash_key = hashlib.sha1(str(setting).encode()).hexdigest()[:6]
exp_name = setting['exp_name']
filename = '{}-{}.json'.format(exp_name, hash_key)
print(hash_key)
```