

Hyperparameter Tuning

TEAMLAB director

최성철

WARNING: 본 교육 콘텐츠의 지식재산권은 재단법인 네이버커넥트에 귀속됩니다. 본 콘텐츠를 어떠한 경로로도 외부로 유출 및 수정하는 행위를 엄격히 금합니다. 다만, 비영리적 교육 및 연구활동에 한정되어 사용할 수 있으나 재단의 허락을 받아야 합니다. 이를 위반하는 경우, 관련 법률에 따라 책임을 질 수 있습니다.

제대로 한번 학습을
해보고 싶다.

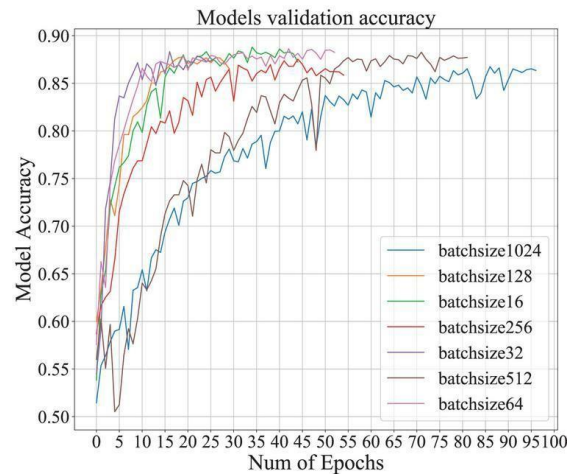
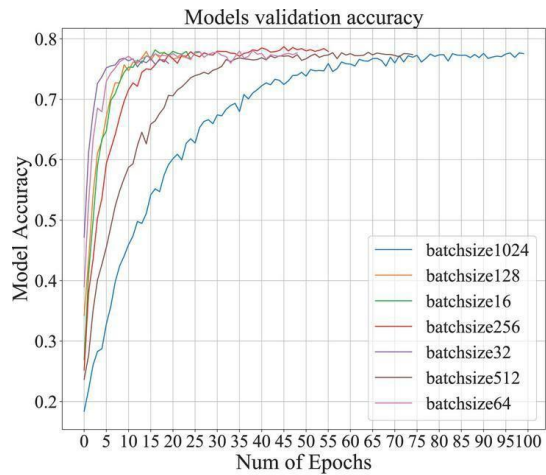


끝날때 까지 끝난게 아니다.



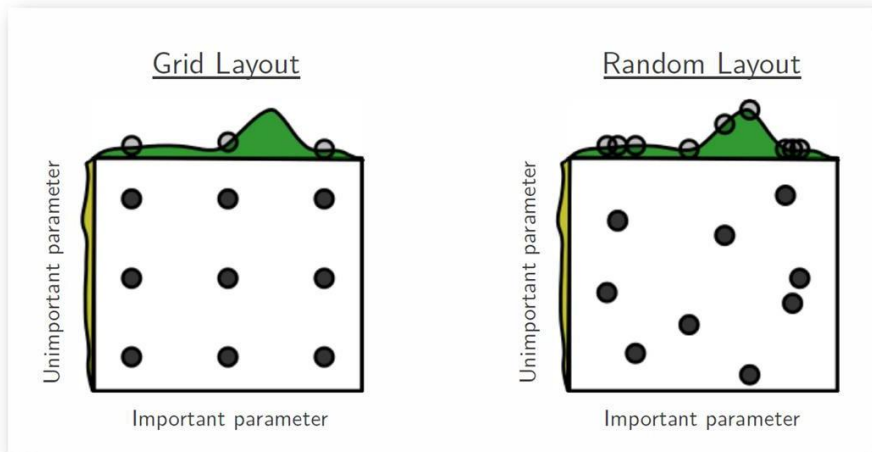
Hyperparameter Tuning

- 모델 스스로 학습하지 않는 값은 사람이 지정 (learning rate, 모델의 크기, optimizer 등)
- 하이퍼 파라미터에 의해서 값의 크게 좌우 될 때도 있음 (요즘은 그닥?)
- 마지막 0.01을 쥐어짜야 할 때 도전해볼만!



<https://content.iospress.com/articles/journal-of-intelligent-and-fuzzy-systems/ifs190033>

- 가장 기본적인 방법 – **grid vs random**
- 최근에는 베이지안 기반 기법들이 주도



<https://dl.acm.org/doi/pdf/10.5555/2188385.2188395>

- multi-node multi processing 지원 모듈
- ML/DL의 병렬 처리를 위해 개발된 모듈
- 기본적으로 현재의 분산병렬 ML/DL 모듈의 표준
- Hyperparameter Search를 위한 다양한 모듈 제공


```
data_dir = os.path.abspath("./data")
load_data(data_dir)
config = {
    "l1": tune.sample_from(lambda _: 2 ** np.random.randint(2, 9)),
    "l2": tune.sample_from(lambda _: 2 ** np.random.randint(2, 9)),
    "lr": tune.loguniform(1e-4, 1e-1),
    "batch_size": tune.choice([2, 4, 8, 16])
}
```

config 에 search space 지정

학습 스케줄링 알고리즘 지정

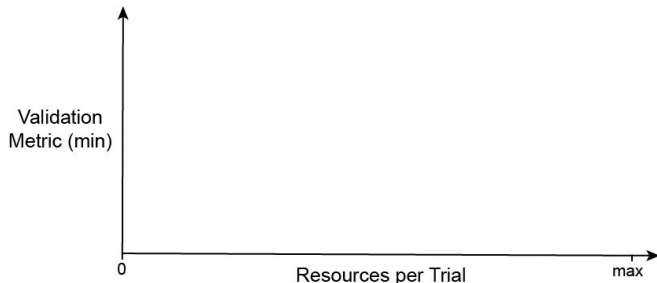
```
scheduler = ASHAScheduler(
    metric="loss", mode="min", max_t=max_num_epochs, grace_period=1, reduction_factor=2)

reporter = CLIReporter(
    metric_columns=["loss", "accuracy", "training_iteration"])
```

결과 출력 양식 지정

```
result = tune.run(
    partial(train_cifar, data_dir=data_dir),
    resources_per_trial={"cpu": 2, "gpu": gpus_per_trial},
    config=config, num_samples=num_samples,
    scheduler=scheduler,
    progress_reporter=reporter)
```

병렬 처리 양식으로 학습 시행



<https://towardsdatascience.com/a-novices-guide-to-hyperparameter-optimization-at-scale-bfb4e5047150>

End of Document
Thank You.