

# Deep Learning

## Practice#3 Report

2016024766

김서현

### 1. 실습 기본 정보

- Train set은 10000개, Test set은 500개의 데이터를 갖고 있습니다.
- Activation function은 모두 sigmoid를 사용했습니다.
- 모든 조건에서 Epochs=1000으로 고정하였습니다.
- Google colab의 GPU 환경에서 실습하였습니다.

### 2. Loss function별 비교

- Mini-batch=128로 고정했습니다.
- Optimizer는 SGD를 사용했습니다.

	BinaryCrossentropy	MeanSquaredError
Accuracy (with train set)	99.9%	99.9%
Accuracy (with test set)	99.8%	99.8%
Train time [sec]	234s	217s

### 3. Optimizer별 비교

- Mini-batch=128로 고정했습니다.
- Loss function은 binary cross entropy를 사용했습니다.

	SGD result	RMSProp	Adam
Accuracy (with train set)	99.9%	99.9%	100%
Accuracy (with test set)	99.8%	99.8%	99.8%
Train time [sec]	233s	251s	240s

#### 4. Mini-batch별 비교

- Optimizer는 SGD를 사용했습니다.
- Loss function은 binary cross entropy를 사용했습니다.

	Mini-batch=4	Mini-batch=32	Mini-batch=128
Accuracy (with train set)	99.9%	99.9%	99.9%
Accuracy (with test set)	99.8%	99.8%	99.8%
Train time [sec]	5778s	882s	231s

#### 5. Discussion

- Binary cross entropy와 Mean squared error를 loss function으로 사용했을 때 정확도는 둘 다 매우 높지만 mean squared error의 training 속도가 더 빠른 것을 알 수 있었습니다.
- Optimizer로 사용한 SGD, RMSProp, Adam 모두 정확도가 매우 높은 것을 알 수 있었습니다.
- SGD, Adam, RMSProp의 순으로, optimizer를 사용했을 때 training 속도가 더 빠르다는 것을 알 수 있었습니다.
- Mini-batch 사이즈가 클수록 training 속도가 빠르다는 것을 확인했습니다.