

## 과제 A

### 1. SGVB에서 Reparametrization trick을 쓰는 이유는?

Parameter  $\theta$ 를 Gaussian에서 샘플링하는데, 이때 미분가능한 파이프라인을 만들기 위하여 사용.

$\theta \sim N(\mu, \sigma^2)$ 에서 바로 샘플링하는 대신,  $\mu$ 와  $\sigma$ 를 모델링하고  $N(0, 1)$ 에서 샘플링한 값과 결합하여 샘플링.

### 2. Sparse VD에서 Local reparameterization을 사용하는 이유는?

Central limit theorem을 이용하여 minibatch size를 늘리면 ELBO 평균의 variance를 줄일 수 있음. 하지만 covariance를 줄이는 것은 불가능하기 때문에, 이를 해결하려면 minibatch안의 데이터간 covariance를 0으로 만들기 위해서 도입.

Minibatch 안의 데이터 수만큼  $\theta$ 를 reparameterization을 이용해서 sampling하면, minibatch각각에 적용되는  $\theta$ 는 서로 독립되어 ELBO간의 covariance를 0으로 만드는 것이 가능.

하지만 이경우 연산 코스트가 비싸지므로, 이전 레이어에서 넘어온 Activation map  $A$ 가 Gaussian이고 역시 Gaussian  $\theta$ 를 곱한 결과인  $B$ 도 Gaussian임을 이용해서  $B$ 에서  $\gamma$ 평균과  $\delta$  variation를  $\zeta \sim N(0,1)$ 로 reparameterization으로 sampling하였다.

### 3. Sparse VD에서 weight의 sparsity가 유도되는 이유는?

Weight 별로 dropout  $\alpha$  ( $1-p/p$ )를 학습하고,  $\alpha$ 가 제한없이 커질 수 있게 학습하여서  $p=1$ 이 되는 weight들을 학습하여 매우 sparse한 학습이 가능하게됨.

### 4. Variational Dropout의 prior가 log uniform인 이유는?

Sparsity induce prior로써 Zero 근처에서 높은 density를 가지고 ELBO에서 Expected loglikelihood term에 대해서만 학습 될 수 있도록 하는 조건을 만족함.

## 과제 B

Model	Test Acc(%)	Layer 1 Memory (%)	Layer 2 Memory (%)	Layer 3 Memory (%)
None	97.98	100%	100%	100%
BBD	98.70	47.70	39.60	53.00
SBPD	98.52	41.71	31.40	38.00

### 1. None Dropout

```
[19] train: epoch 40, (2.387 secs), cent 0.072233, acc 0.990033
test: epoch 40, (2.618 secs), cent 0.073178, acc 0.980200
train: epoch 41, (2.406 secs), cent 0.011078, acc 0.996567
test: epoch 41, (2.653 secs), cent 0.072442, acc 0.980100
train: epoch 42, (2.387 secs), cent 0.009565, acc 0.996950
test: epoch 42, (2.619 secs), cent 0.067614, acc 0.981600
train: epoch 43, (2.440 secs), cent 0.008964, acc 0.997183
test: epoch 43, (2.663 secs), cent 0.068562, acc 0.982100
train: epoch 44, (2.385 secs), cent 0.011073, acc 0.996500
test: epoch 44, (2.617 secs), cent 0.072409, acc 0.978900
train: epoch 45, (2.347 secs), cent 0.009003, acc 0.997317
test: epoch 45, (2.571 secs), cent 0.069944, acc 0.981700
train: epoch 46, (2.374 secs), cent 0.010369, acc 0.996467
test: epoch 46, (2.597 secs), cent 0.070261, acc 0.981700
train: epoch 47, (2.409 secs), cent 0.009643, acc 0.997117
```

## 2. Bbdropout

```
[21] n_active: [375, 199, 162]

train: epoch 55, (6.542 secs), cent 0.015616, acc 0.995317
test: epoch 55, (6.882 secs), cent 0.042695, acc 0.986900
Epoch 55 start, learning rate 0.000100
n_active: [374, 198, 162]

train: epoch 56, (6.612 secs), cent 0.015070, acc 0.995467
test: epoch 56, (6.941 secs), cent 0.042548, acc 0.987100
Epoch 56 start, learning rate 0.000100
n_active: [374, 198, 162]

train: epoch 57, (6.551 secs), cent 0.015562, acc 0.995450
test: epoch 57, (6.874 secs), cent 0.042587, acc 0.986900
Epoch 57 start, learning rate 0.000100
n_active: [374, 198, 160]

train: epoch 58, (6.533 secs), cent 0.014885, acc 0.995750
test: epoch 58, (6.881 secs), cent 0.042638, acc 0.987100
Epoch 58 start, learning rate 0.000100
n_active: [374, 198, 160]

train: epoch 59, (6.487 secs), cent 0.014846, acc 0.995600
test: epoch 59, (6.803 secs), cent 0.042617, acc 0.986800
Epoch 59 start, learning rate 0.000100
n_active: [374, 198, 159]

train: epoch 60, (6.464 secs), cent 0.014177, acc 0.996033
test: epoch 60, (6.801 secs), cent 0.042763, acc 0.987000
Epoch 60 start, learning rate 0.000100
n_active: [374, 198, 159]
```



### 3. Sbdropout

```
train: epoch 57, (12.090 secs), cent 0.004017, acc 0.999917
test: epoch 57, (14.211 secs), cent 0.046885, acc 0.985400
Epoch 57 start, learning rate 0.000100
n_active: [327, 157, 115]

train: epoch 58, (12.310 secs), cent 0.004032, acc 0.999867
test: epoch 58, (14.405 secs), cent 0.046757, acc 0.985300
Epoch 58 start, learning rate 0.000100
n_active: [327, 157, 115]

train: epoch 59, (12.111 secs), cent 0.004033, acc 0.999900
test: epoch 59, (14.163 secs), cent 0.046952, acc 0.985100
Epoch 59 start, learning rate 0.000100
n_active: [327, 157, 114]

train: epoch 60, (11.957 secs), cent 0.004055, acc 0.999900
test: epoch 60, (14.105 secs), cent 0.046863, acc 0.985200
Epoch 60 start, learning rate 0.000100
n_active: [327, 157, 114]
```

