2019150445 신백록

원래 12.3절 모형을 training하면 loss와 prediction에 대한 plot이 시켜보면 다음과 같이 나온다.

Epoch 9/10

3750/3750 [==============================] - 19s 5ms/step - loss: 0.0470 - val\_loss: 0.0471

Epoch 10/10

텍스트, 전자기기이(가) 표시된 사진

자동 생성된 설명3750/3750 [==============================] - 18s 5ms/step - loss: 0.0469 - val\_loss: 0.0472

1.

Latent dim=3으로 두고 위와 똑같은 모델을 학습하면, loss는 다음과 같고, 원래 모형의 loss보다 약간 작은 값이 나온다. 이 모형에 dimension=3인 multivariate normal distribution이 어떤 식으로 이루어져 있는지를 알기 위해 decoder에 prediction해서 (420,420,420)의 figure output을 만들었지만 아무리 서치하고 노력해도 3D plot을 그리는 데에 실패했다.

Epoch 9/10

3750/3750 [==============================] - 16s 4ms/step - loss: 0.0427 - val\_loss: 0.0428

Epoch 10/10

3750/3750 [==============================] - 16s 4ms/step - loss: 0.0426 - val\_loss: 0.0432

2.

Channel 수를 encoder에서는 32, 64, 128로 늘리고, decoder에서는 128, 64, 32로 줄이는 과정을 시행하였다. 모수가 전보다 4배 이상 많아졌고, train 시키고 bivariate normal distribution의shape을 알아보기 위해 prediction을 시키니 다음과 같은 plot이 나왔다.

conv2d\_22 (Conv2D) (None, 7, 7, 64) 18496 conv2d\_21[0][0]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

conv2d\_23 (Conv2D) (None, 3, 3, 128) 73856 conv2d\_22[0][0]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

conv2d\_transpose\_8 (Conv2DTr (None, 7, 7, 128) 147584

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

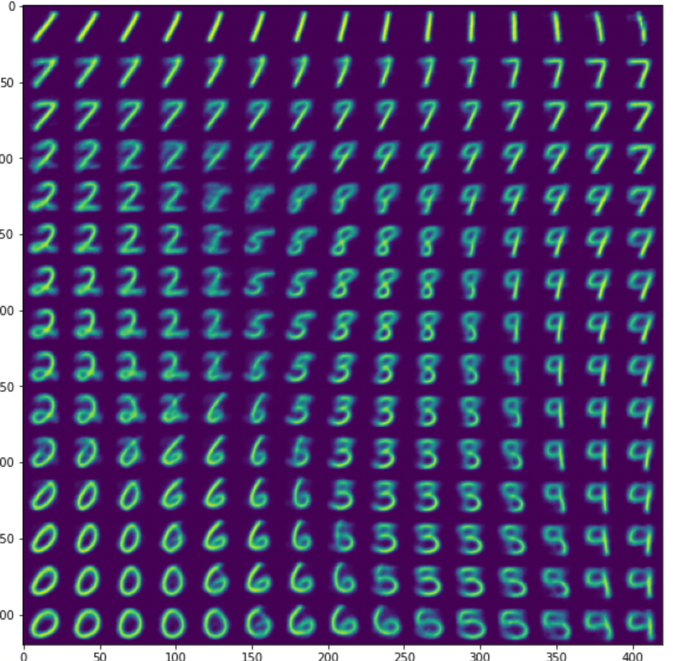
conv2d\_transpose\_9 (Conv2DTr (None, 14, 14, 64) 73792

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Epoch 9/10

3750/3750 [==============================] - 20s 5ms/step - loss: 0.0464 - val\_loss: 0.0472

Epoch 10/10

3750/3750 [==============================] - 20s 5ms/step - loss: 0.0464 - val\_loss: 0.0468텍스트, 패브릭이(가) 표시된 사진

자동 생성된 설명

전과 비슷한 모양이지만 숫자의 배열은 바뀌었다. 이는 train 과정에서 shuffle=True로 놔둬서 변하는 것 같아 같은 모형을 다시 train 해봤더니 plot의 모양이 바뀌었다. 5에서 2로 바뀌는 부분은 위와 동일하게 prediction을 잘 못하고 있고, blur 처리되어 있는 것처럼 보인다.

3.

Latent dim=3, channel은 32,64,128로 두고 모델을 만들었고, 학습을 하였다. Loss가 전의 모형들보다 많이 줄었다. 마찬가지로 3 dimension의 plot을 그리는 데에 실패하였다..

Epoch 9/10

3750/3750 [==============================] - 19s 5ms/step - loss: 0.0414 - val\_loss: 0.0413

Epoch 10/10

3750/3750 [==============================] - 19s 5ms/step - loss: 0.0413 - val\_loss: 0.0412

4.

Tile은 array를 복제하는 결과를 준다. np.tile(arg1,arg2)에서 arg1은 input array를 받고, arg2는 반복하는 횟수를 받는다.

z\_x와 z\_y는 norm.ppf로 0.05부터 0.95까지의 15개의 x quantile을 저장한다. 그 다음 z\_sample에 z\_x와 z\_y를 하나씩 넣고, decoder의 input으로 들어가기위한 shape으로 바꾸기 위해 np.tile으로 z\_sample을 16번 복제한다. 그러면 (2,16)의 array가 만들어지고, 이것을 reshape을 통해 (16,2)의 array로 바꿔 저장한다. 이것을 decoder에 predict하고, tile에 의해 16항이 다 같기 때문에 첫번째 행만 저장한다. 그리고 figure에 image size만큼 늘려가며 이미지를 저장한다. 위와 같은 과정을 for문에 의해 15\*15번 반복한다..