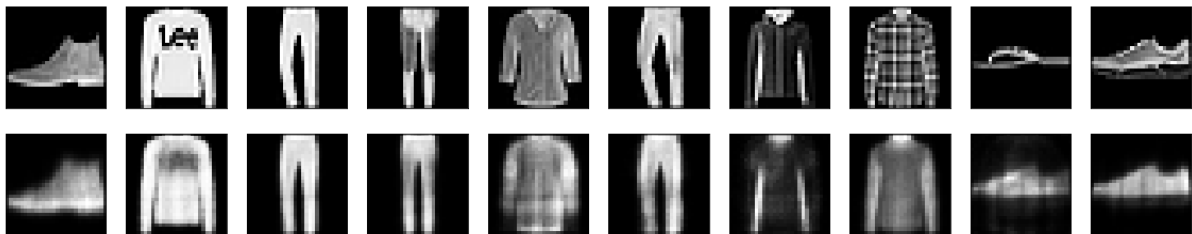


HW_8

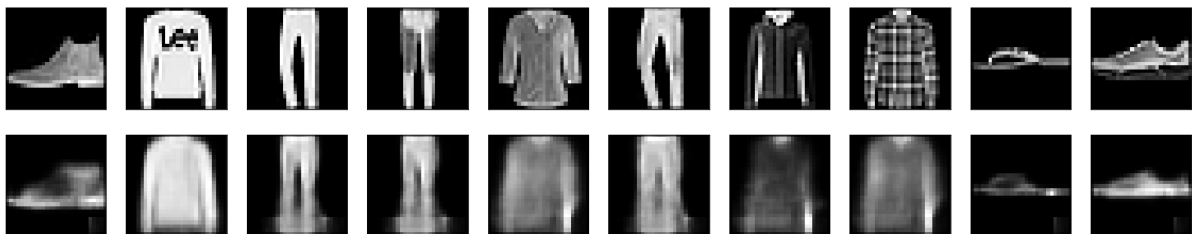
고찰

Fashion MNIST dataset을 사용하여 이미지를 복원해보았습니다.

Image pixel이 28x28으로 아주 간단하여, 기본 Dense로도 형태는 잘 복원됨을 알 수 있습니다.



Convolutional layer가 포함된 autoencoder_cnn를 같은 데이터로 학습시켜보았습니다

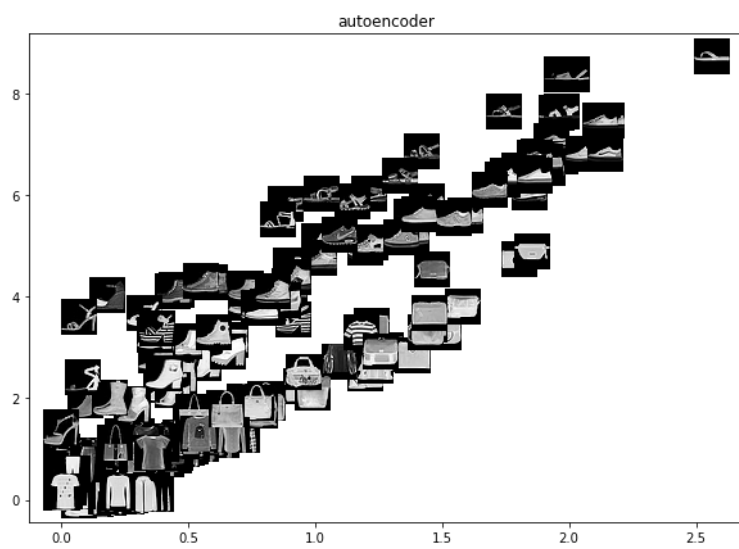
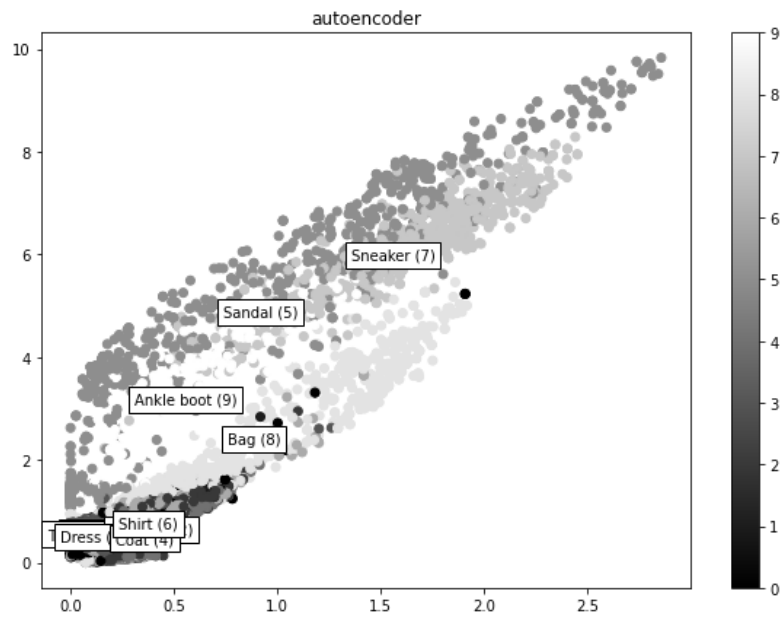


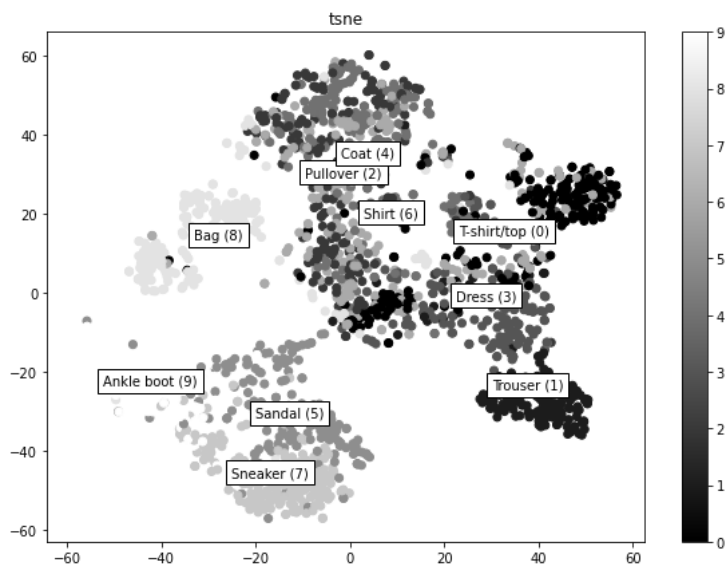
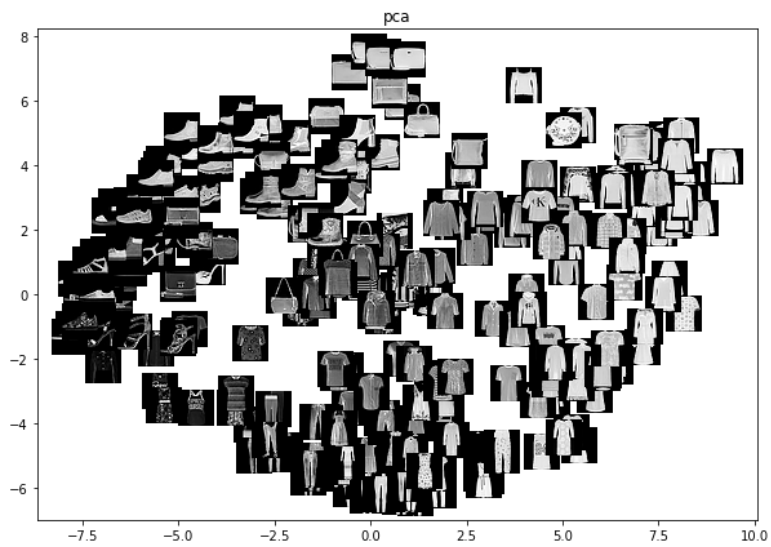
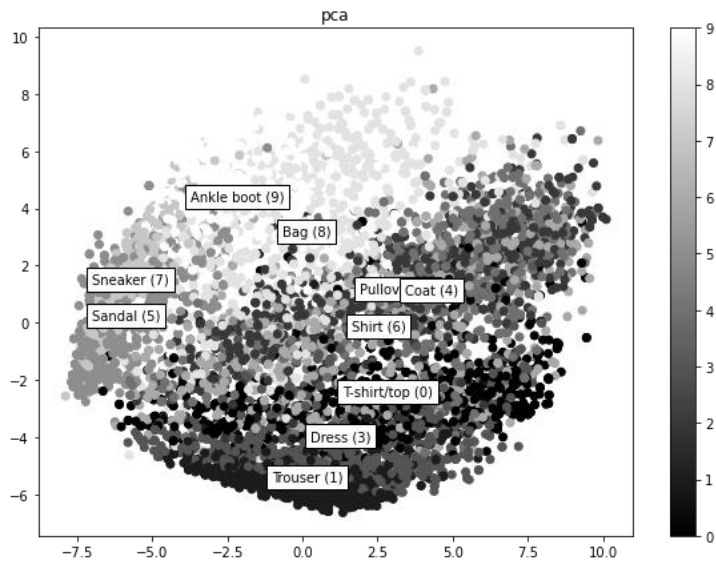
callback함수에 의해 조기종료(약 40epoch)되어 val_loss는 1.basic autoencoder보다 낮았습니다. 만약 earlystopping되지 않았다면 더 많은 epoch을 돌아 낮은 val_loss를 얻을 수 있었을것이라고 예상합니다. 하지만 학습시간이 굉장히 오래걸렸기 때문에 조기 종료한 후 결과를 비교하였습니다.

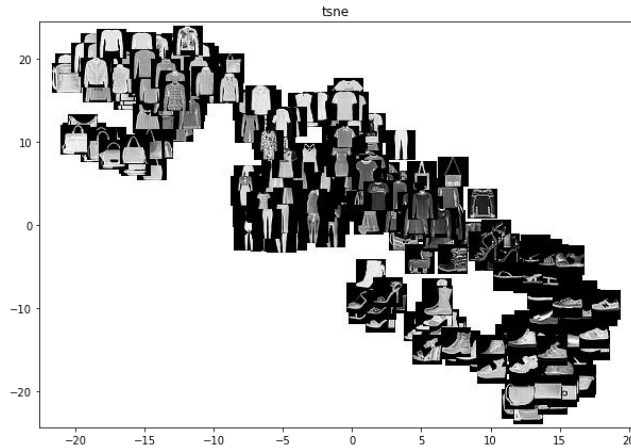
형태만 잡아냈던 basic autoencoder와는 다르게, 이미지의 feature를 뽑아내는 Convolutional layer가 포함된 autoencoder_cnn에서 더 세부적인 feature를 뽑아냄을 볼 수 있었습니다(신발종류들). Convolutional layer가 깊어질수록 형태뿐만아니라 세부적인 feature를 잘 잡아내기 때문에 Convolutional layer를 더 깊이 쌓을수록 더 복원이 잘 될것입니다.

Convolutional layer로 구성된 autoencoder를 차원축소를 하여 군집을 판단하는 PCA, T-

sne와 비교해보았습니다.



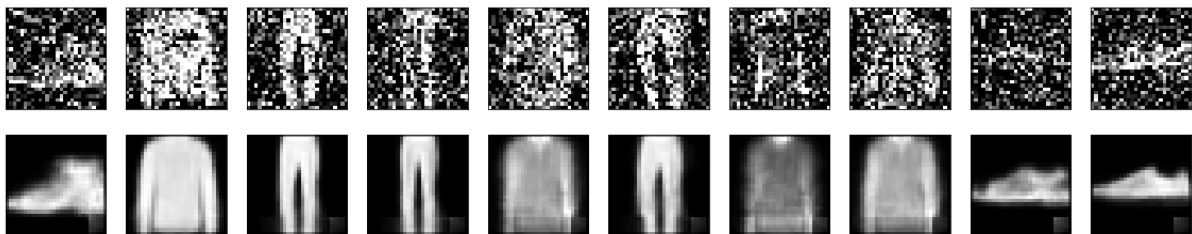




Basic autoencoder세부적인 feature를 잘 잡아내기 위해 dense를 깊이 하기 하였지만, PCA와 T-SNE와 비교하였을 때는 많이 부족한 결과를 보입니다. 이 dataset에는 T-SNE가 가장 좋은 성능으로 label을 분류하였습니다.

성능개선을 위해서는 더 깊은 Convolution layer와 더 많은 epoch으로 학습시킬 필요성을 느꼈습니다.

마지막으로 denosing autoencoder를 구현해보았습니다.



이미 2.convolutional autoencoder에서 적당히 깊게 encoder와 decoder를 구현하였기 때문에, 같은 모델에 noisy data로 학습시켰습니다. Convolutional layer의 깊이가 깊었기 때문에, 데이터에 추가한 noise가 기대보다 아주 잘 제거되었습니다.

구성한 autoencoder_cnn은 denosing역할은 잘 수행하였지만, 더 정확한 classification을 위해서는 앞서 언급한바와 같이 깊은 Conv와 많은 학습이 필요합니다.