**프로젝트 #2**

**제 목: 실습 및 합성곱 신경망 구현**

1. **프로젝트 자체 평가(본인 평가)**

**① 완성도 및 성실성( 20점 ) : 상(20)( V ) / 중(18)( ) / 하(16)( )**

1. **프로젝트 외부 평가(교수 평가)**

**① 과제 #1의 완성도( 10점 ) : 상(10)( ) / 하(7)( )**

**② 과제 #2 완성도( 40점 ) : 상(40)( ) / 중(35)( ) / 하(30)( )**

**③ 보고서 작성 성실성( 30점 ) : 상(30)( ) / 중(26)( ) / 하(22)( )**

**④ 제출일 미준수로 인한 감점( -20점 ): 제출일 미준수(-20)( )**

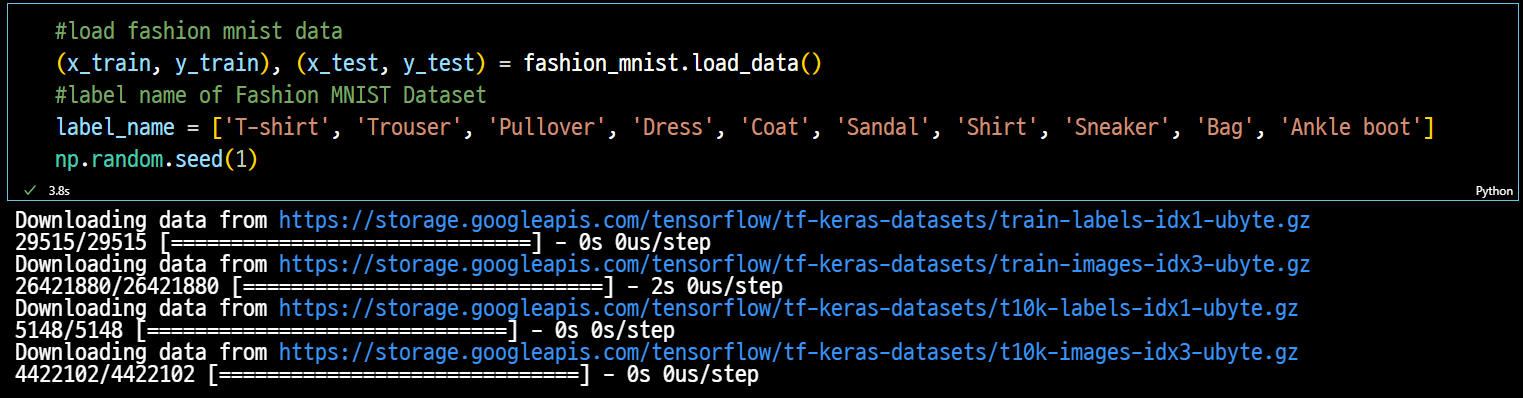
**소계: ( ) 점**

**※ 총점: ( ) 점**

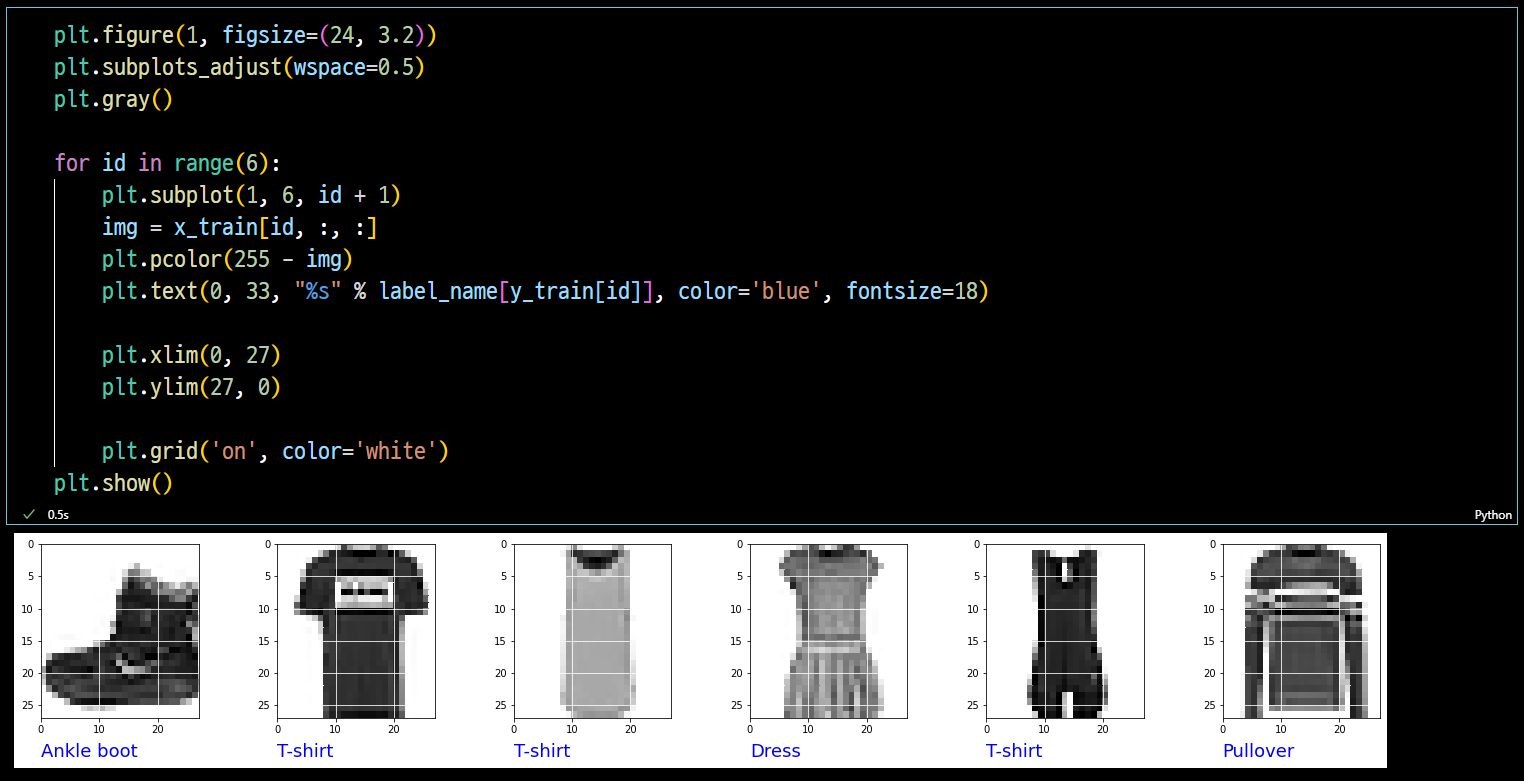
|  |  |
| --- | --- |
| **과 목 명:** | **Deep Learning**  **Network Design Application** |
| **학 과:** | 소프트웨어공학과 |
| **학 번:** | **4102210009** |
| **성 명:** | **이동훈** |
| **제 출 일:** | **2022년 6월 17일(금)** |

문제 해결과정

1. 필요 패키지를 import 하였습니다.
2. Fashion\_mnist 데이터 셋을 load 하고 라벨 이름을 지정하였습니다.



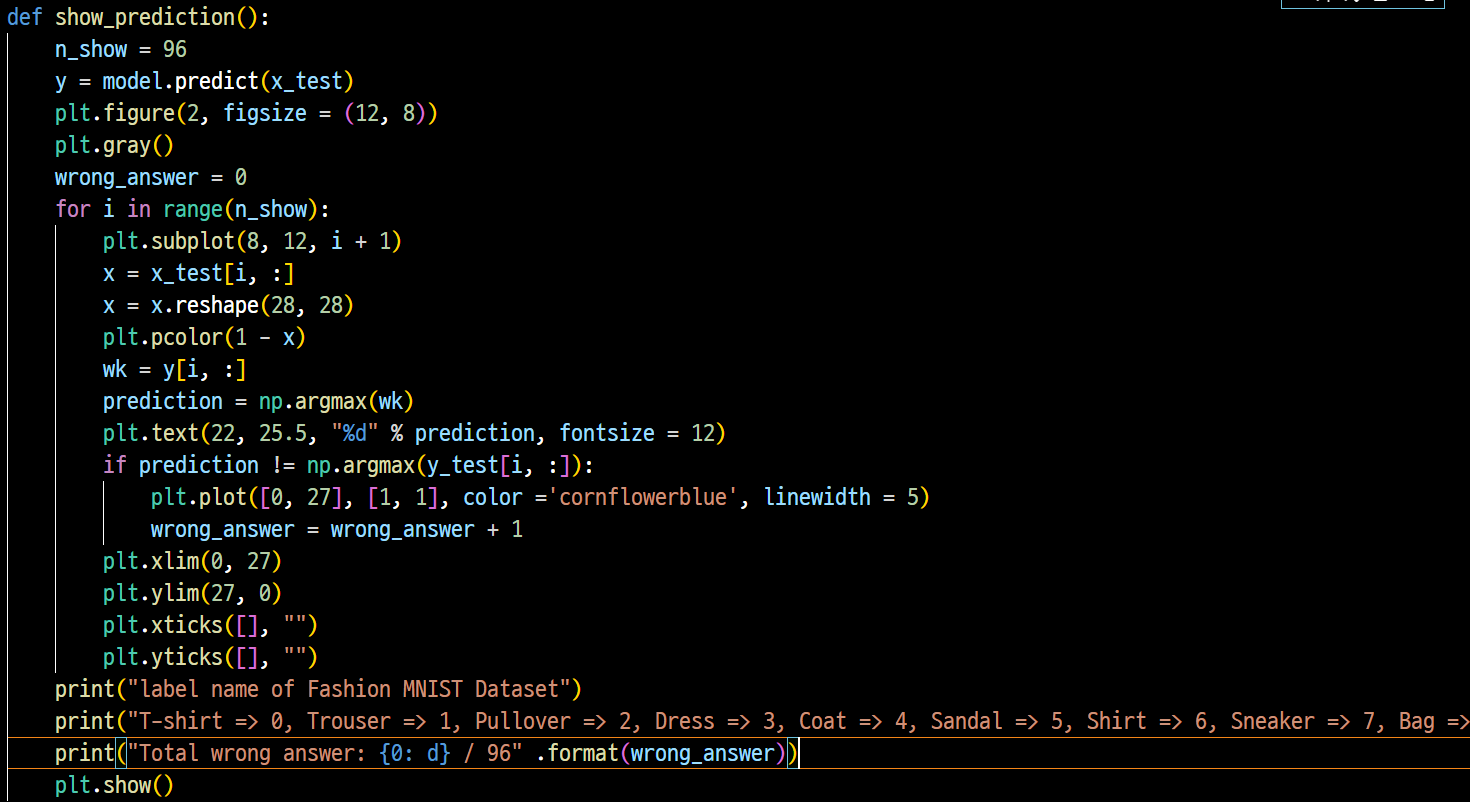
1. 훈련데이터를 추출하여 상위 벡터 이미지 6개를 샘플로 시각화합니다.



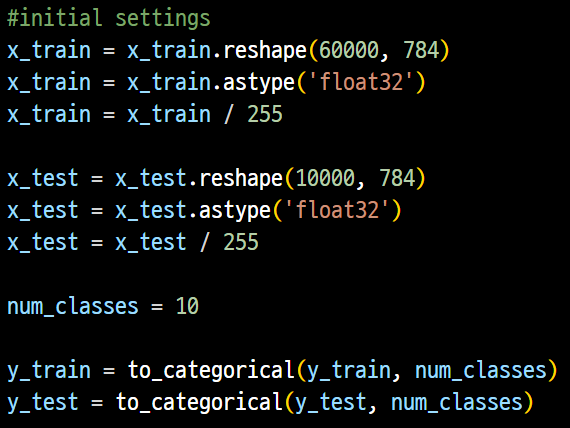
1. 아래 세가지의 주요 기능을 하는 함수입니다. 인자는 모델과 학습 및 훈련데이터, epoch과 batch size입니다.
   1. 모델을 주어진 훈련데이터로 학습시킵니다.
   2. x = epoch와 y = loss, accuracy에 대한 인과관계를 2차원 좌표평면 상에 시각화 합니다.
   3. 소요시간과 테스트 데이터에 대한 모델의 평가 값을 출력합니다.



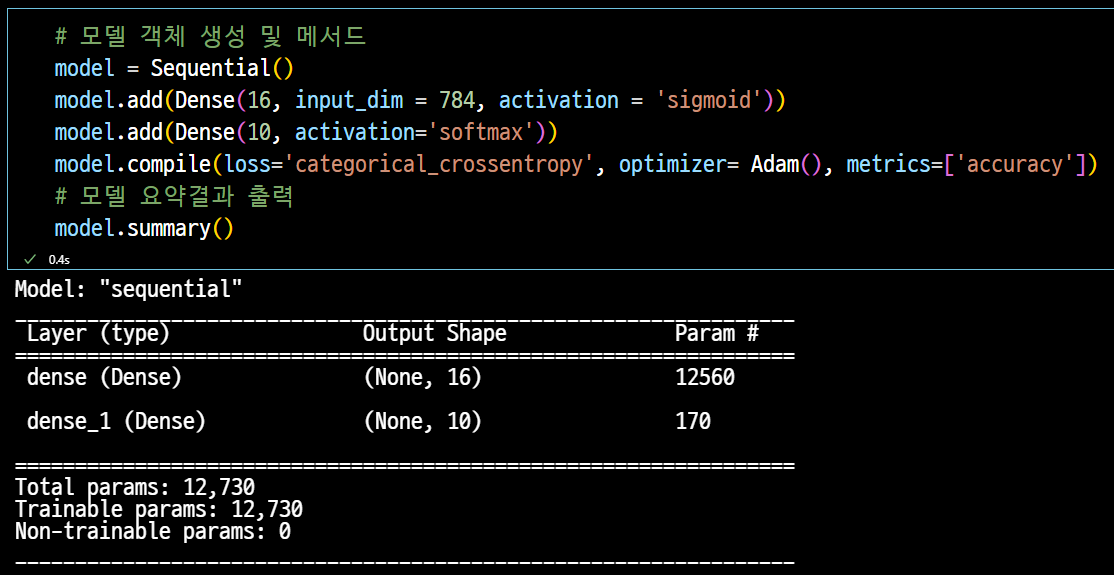
1. 시각화 함수로 인자는 없습니다.  
   상위 96개의 데이터를 12 \* 8의 그리드로 시각화 합니다.  
   예측에 실패한 레이블은 상단에 파란 테두리입니다.



1. 60000 \* 28 \* 28의 데이터를 60000\* 784의 데이터로 바꾸고 실수형으로 바꿔 0과 1사이의 값으로 정규화 합니다. 레이블의 개수를 Keras 함수를 이용하여 1-of-K의 부호화법으로 변형합니다.

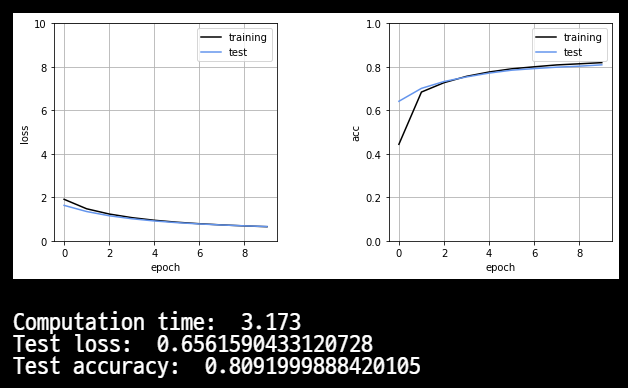


1. 중간층은 784개의 입력을 갖는 16개의 뉴런 출력층은 10개의 뉴런으로 구성된 Sequential 모델을 생성합니다. 손실함수는 categorical cross-entropy, optimizer는 adam을 사용합니다.



1. Epoch 10, batch size 1000으로 수행한 결과는 3.173초의 수행시간, loss는 약 0.66 정확도는 0.81의 학습 수준을 보여줍니다.

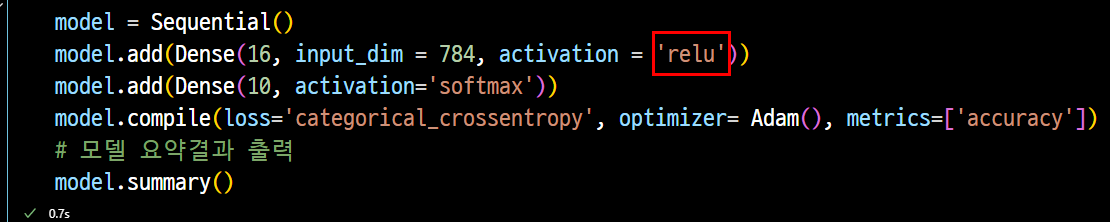




1. 오답은 96개중 24개로 낮은 정확도를 보여줍니다.



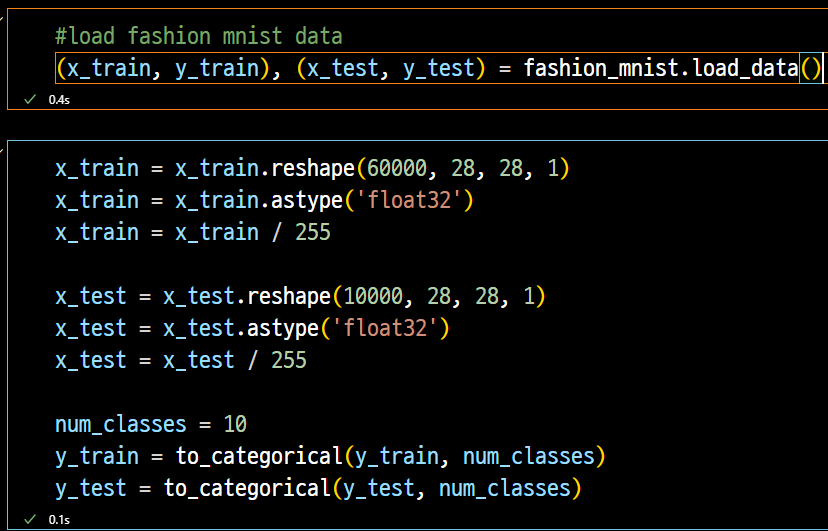
1. 활성화 함수를 sigmoid 함수에서 relu 함수로 변경하여 다시 모델을 만들고 학습하였습니다.



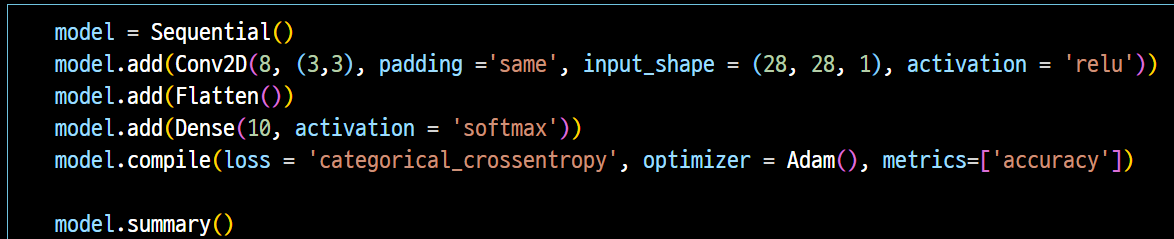
1. Epoch 10, batch size 1000으로 수행한 결과는 3.528초의 수행시간, loss는 약 0.46 정확도는 0.84의 학습 수준을 보여줍니다. 관련하여 해당 데이터 셋의 활성화 함수로는 relu를 사용해야 좀더 정밀한 예측이 가능함을 확인하였습니다. Sigmoid 함수보다 실패한 개수가 8개 감소하였습니다.



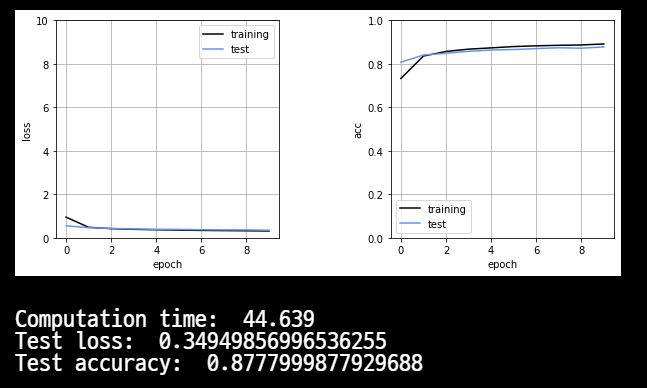
1. 학습 데이터(60000 \* 28 \* 28 \* 1)를 다시 로드합니다.



1. 아래와 같이 합성곱 신경망으로 모델을 정의하였습니다.



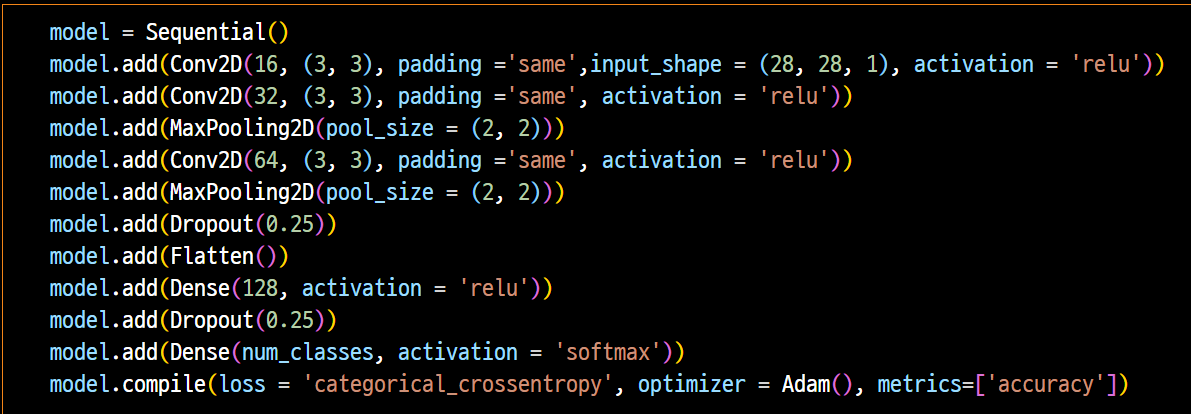
1. Epoch 10, batch size 1000으로 수행한 결과는 44.639초의 수행시간, loss는 약 0.35 정확도는 0.88의 학습 수준을 보여줍니다. 합성곱 신경망은 레이어가 많아지고 시간이 오래걸리지만 그만큼 정확한 예측률을 보여줍니다.



1. 예측 실패는 10개 정도로 감소하였습니다.



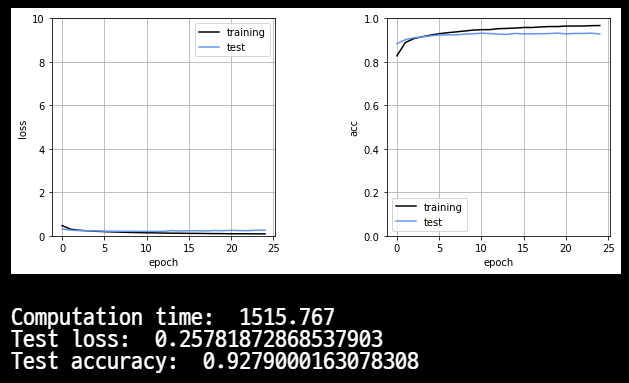
1. 합성곱 신경망에 풀링과 드롭아웃을 도입하여 계층의 수를 늘린 모델을 정의하였습니다.



1. Epoch 10, batch size 1000으로 수행한 결과는 440.249초의 수행시간, loss는 약 0.25 정확도는 0.90의 학습 수준을 보여줍니다. 합성곱 신경망에 풀링과 드롭아웃을 적용하여 레이어를 증가시키면 더 좋은 예측모델이 됨을 확인하였습니다.
2. 상위 96개의 오답률은 똑같지만 전체적인 예측능력은 더 높을 것으로 예상됩니다.



1. 방금과 모델은 동일하지만 Epoch를 25 batch size를 50으로 조정하여 학습하였습니다. 결과는 1515.767초의 학습시간, loss는 약 0.26 정확도는 0.93의 학습 수준을 보여줍니다. 정확도가 개선되긴 하였으나 훈련 데이터는 개선되는 것이 보이지만 테스트 데이터는 미미하게 개선되는 것으로 보아 과도하게 학습하면 오히려 모델의 예측능력이 떨어지는 것으로 확인하였습니다.



1. 해당 모델은 7개의 예측실패를 보여주고 있습니다.

