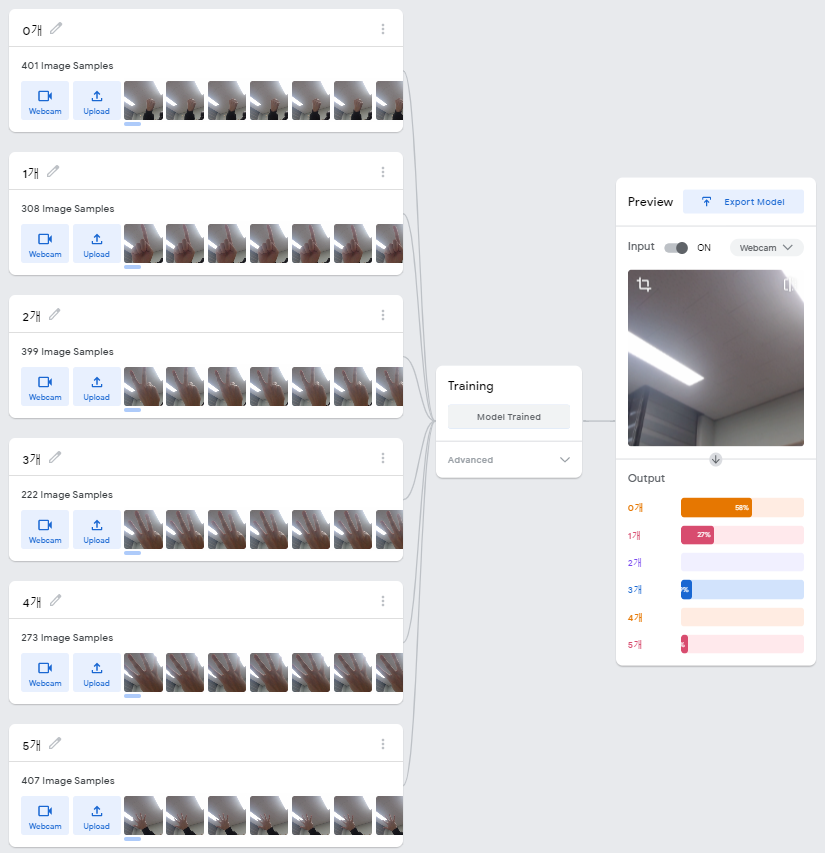
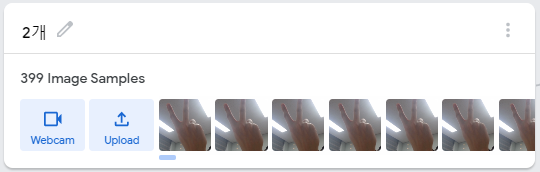
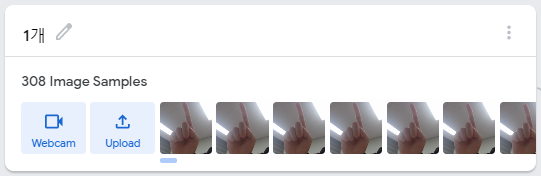
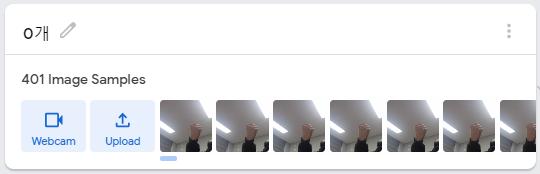
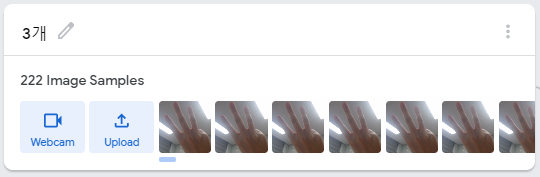
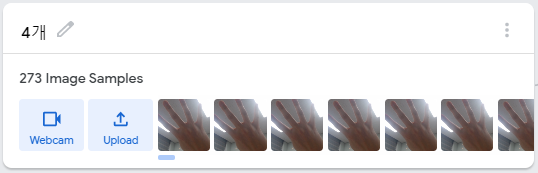
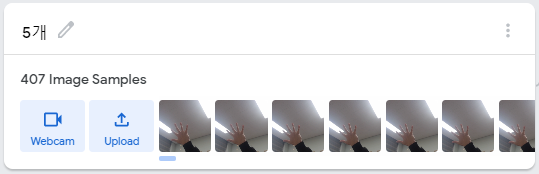
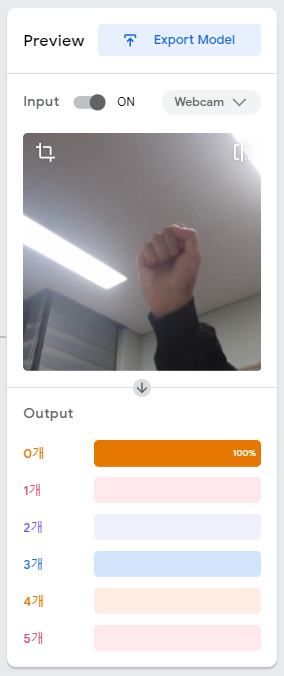
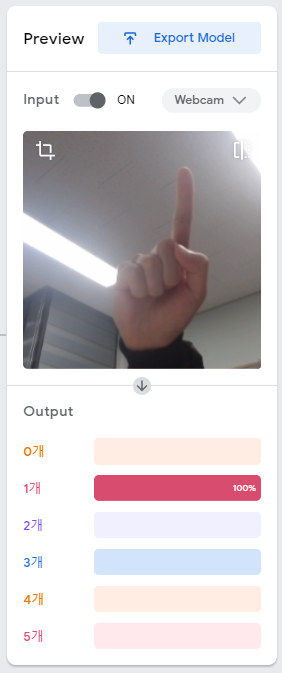
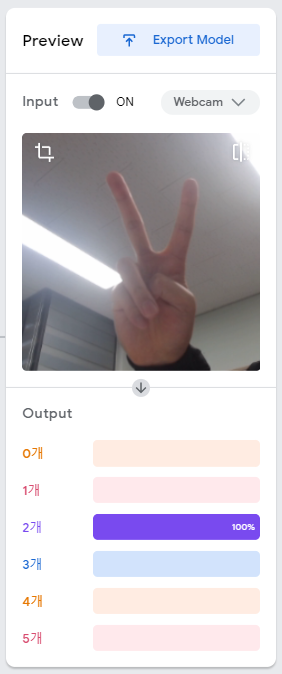
|  |  |  |
| --- | --- | --- |
|  | 9주차 과제 수행. | **9주차** |
| 학번: \_\_\_\_\_201413711\_, 이름:\_\_\_\_\_\_정 보 경\_\_ | | |

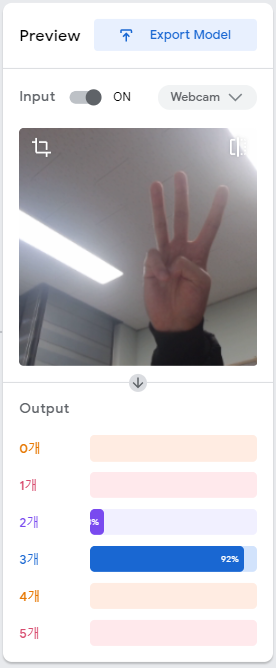
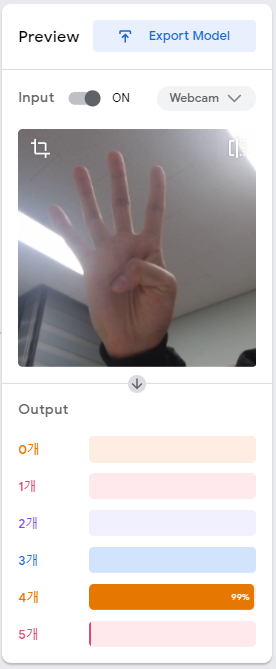
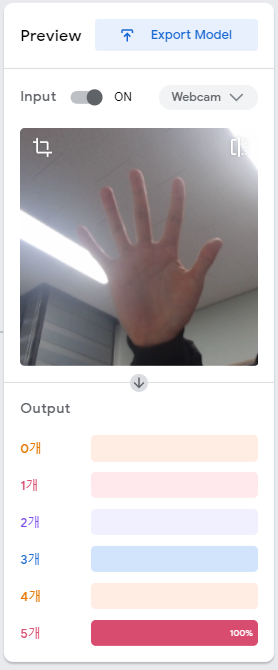
1. 티쳐블 머신을 사용하여 손가락 개수를 맞추는 모델을 학습시키고 테스트하라.  
     
   1) 티쳐블 머신 학습  
     
     
   - 손가락이 0개 또는 아무것도 없을 때부터, 손가락이 5개인 경우까지 총 6가지 케이스 학습





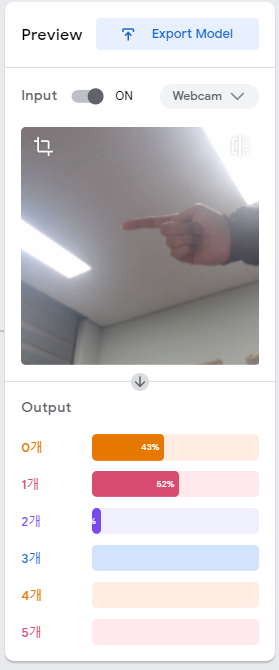
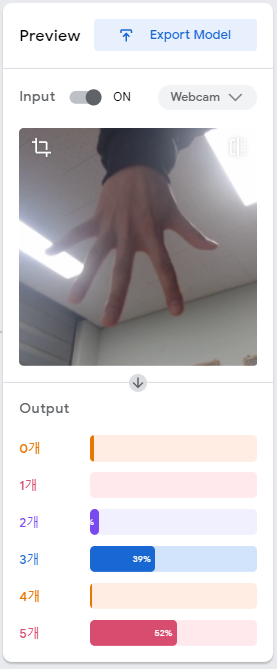
2) 학습 결과

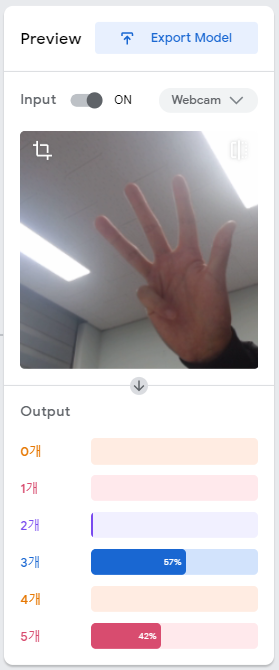
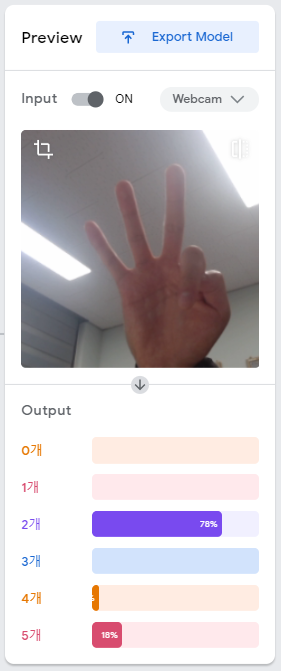
  

* 높은 인식율을 보이는 모델 구축 성공

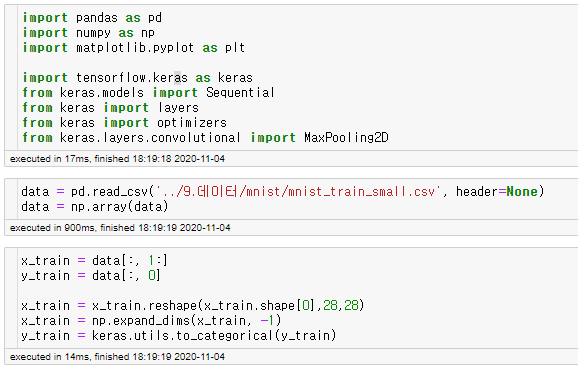
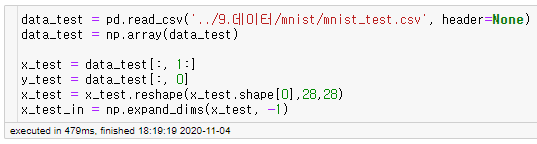
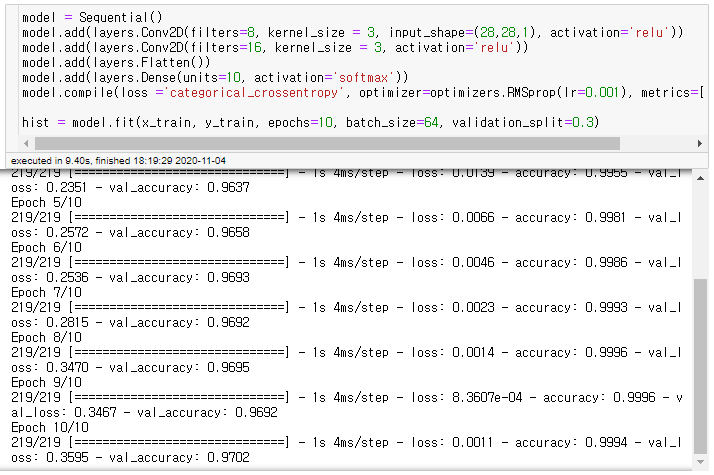
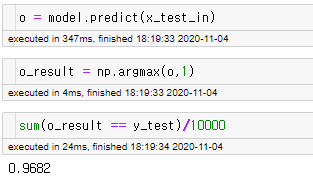
3) 모델의 한계

* 학습한 데이터와 다른 방향의 데이터를 주면 신뢰성이 급격하게 떨어짐

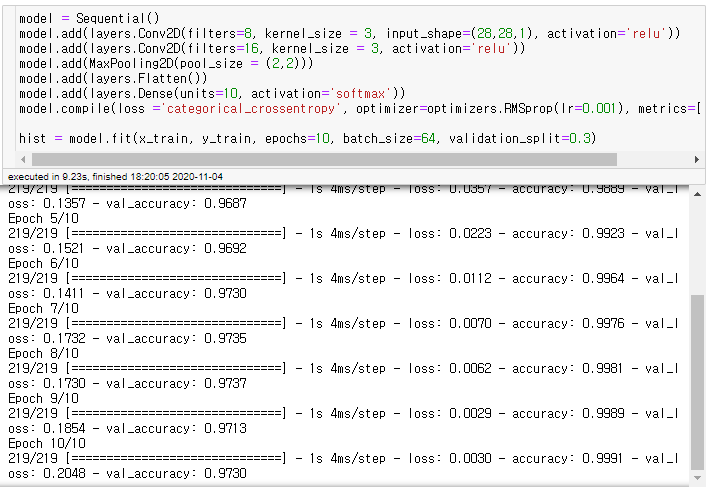
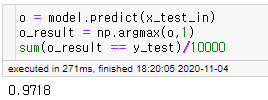
 

* 학습한 데이터와 다른 손가락으로 테스트를 할 시 인식률이 급격하게 떨어짐
* 결국 각 클래스에 해당하는 다양한 종류의 양질의 데이터를 학습시켜줘야 인식률이 높아짐을 확인
* 또한 이미지 자체를 학습하는 것 보다, 손가락의 각 관절을 따로 디텍트하여 관절에 상태에 따른 클래스분류를 수행한다면, 배경이 어디든, 어떤 손가락을 사용하든 인식률이 높을것으로 기대

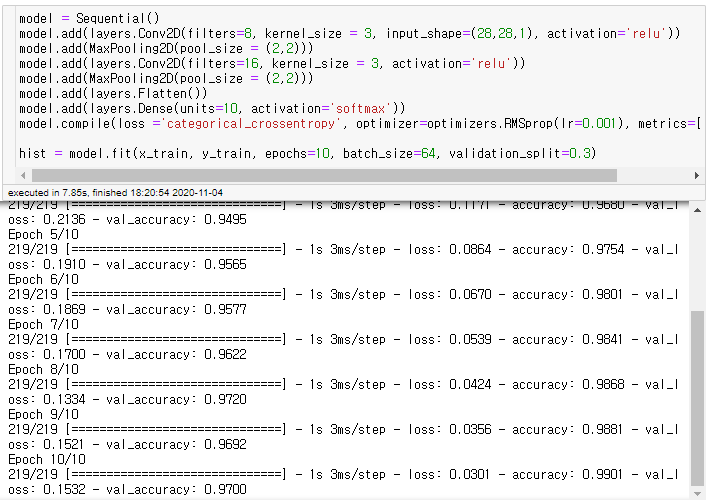
1. 파이썬 코드를 사용하여 mnist 학습을 수행하라. 네트워크 구조를 바꾸어 수업 동영상의 결과보다 높은 성능을 획득하라.
2. Mnist 글씨체 데이터 전처리(학습 및 테스트 데이터)  
     
3. 케라스를 사용하여 합성곱 네트워크 구성 및 학습 및 정확도 확인  
     
    
4. 네트워크 구조 변경 후, 정확도 변화 확인

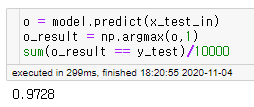
(1) MaxPooling 레이어 추가

· 1개 추가

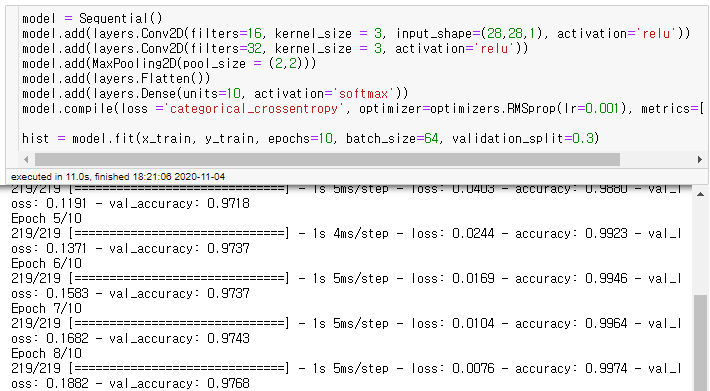
· 2개 추가

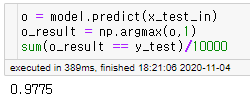




* MaxPooling 레이어 추가시 정확도가 소폭 상승하는 것을 확인

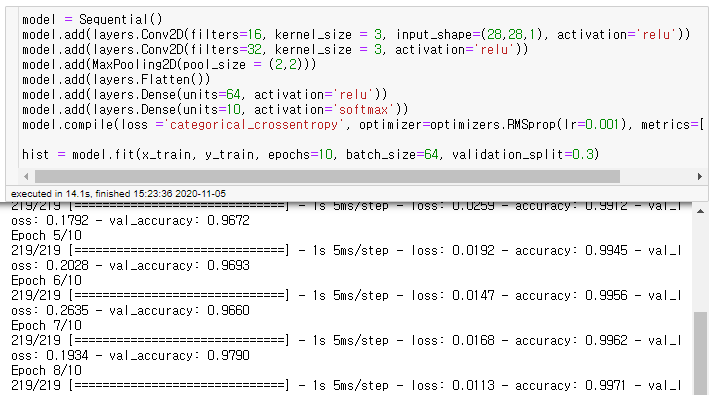
(2) 뉴런 개수의 추가

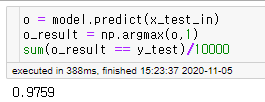




* 뉴런개수의 추가에 따라 정확도가 소폭 상승하는 것을 확인

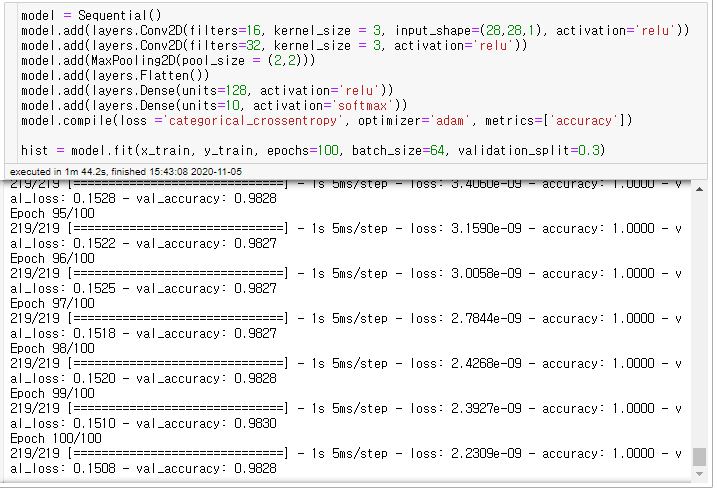
(3) Dense 레이어 추가

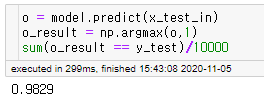




* Dense 레이어 추가는 정확도 상승에 큰 영향을 끼치지 못하는 것을 확인

(4) Optimizer 및 epochs 횟수 변경





* Optimizer에 Adam을 사용하고 epochs를 100회로 늘이니 정확도가 0.9829까지 올라가는 것을 확인