|  |  |
| --- | --- |
| 교육 제목 | **NN Rotation & CNN** |
| 교육 일시 | 21.10.28 |
| 교육 장소 | 온라인 (집) |
| **교육 내용** | |
| 오전 | **▣ NN Rotation**  -------------------------------------------------------------------------------------------  **∇ NN Rotation**  \* 과적합 방지를 위해 로테이션, 이동 등을 해줌.  # 이미지 회전시키기. # Matrix 생성  M = cv2.getRotationMatrix2D((28/2, 28/2), 0, 1)  # 이미지 이동 변환 매트릭스 구하기  M[0, 2] = M[0, 2] + 3  M[1, 2] = M[1, 2] + 3  -------------------------------------------------------------------------------------------  **∇ 모델 불러오기**  \* 테스트할 때 작성 된 모델 불러오기  model = tf.keras.models.load\_model  ('/content/drive/MyDrive/Colab Notebooks  /00.자료/my\_NN\_Test.h5')  -------------------------------------------------------------------------------------------  **▣ NN Rotation**  -------------------------------------------------------------------------------------------  **∇ 데이터 나누기**  \* Train, Valid, Test 로 나누고 인코딩 까지 해줌  # 데이터를 (배치사이즈(몇 묶음짜리인지) x 28 x 28 x 1)로   이미지를 변환해줌. -> 그레이 스케일이므로 채널은 1  x\_train = x\_train.reshape(x\_train.shape[0], 28, 28, 1)  .astype('float64')  x\_test = x\_test.reshape(x\_test.shape[0], 28, 28, 1)  .astype('float64')  # 정답을 바이너리화 함.  y\_train = tf.keras.utils.to\_categorical(y\_train)  # 타겟들 원핫인코딩해줌  y\_test = tf.keras.utils.to\_categorical(y\_test)  ------------------------------------------------------------------------------------------- |
| 오후 | **▣ CNN – 효율적인 분석**  -------------------------------------------------------------------------------------------  **∇ Convolution**  \*모델 형성 때 kernel을 이용해서 부분의 특징들을 뽑아줌  \* 뽑아진 특징들로 activation map을 새로 형성  \* channel 사이즈는 padding을 사용하지 않으면 변한다.  \* tf.keras.layers.Conv2D()  -------------------------------------------------------------------------------------------  **∇ MaxPooling**  \* convolution을 통해 만들어진 activation map을 따로 분석을 거치지 않고,  평균값(Average) 또는 최대값(Max)만 뽑아서 사이즈를 줄여줌.  \* channel size의 변화는 없음.  \* tf.keras.layers.MaxPooling2D()  -------------------------------------------------------------------------------------------  **▣ CNN – 데이터 줄이기**  -------------------------------------------------------------------------------------------  **∇ Dropout**  \* 아예 삭제해서 크기를 줄여줌  \* tf.keras.layers.Dropout()  -------------------------------------------------------------------------------------------  **∇ Batch Normalization**  \*  \* tf.keras.layers.BatchNormalization()  ------------------------------------------------------------------------------------------- |