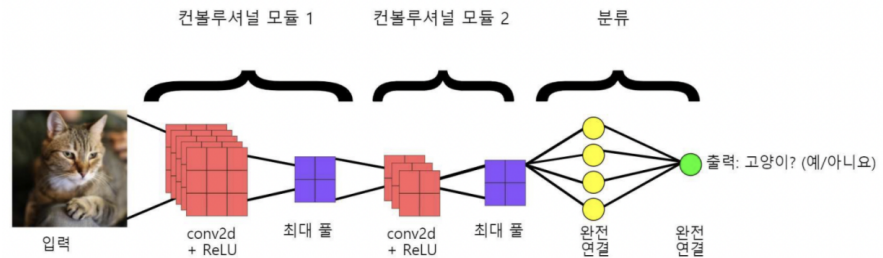


교육 제목	데이터 기반 인공지능 시스템 엔지니어 양성 과정																									
교육 일시	2021년 10월 29일																									
교육 장소	YGL C-6 학과장 & 자택(디스코드 이용한 온라인)																									
교육 내용																										
오전	<div>Review</div> <div>1. 모델별 주요 특징</div> <table><thead><tr><th>Model</th><th>Example</th><th>[Output] Activation Fun.</th><th>Loss Fun.</th><th>Metric</th></tr></thead><tbody><tr><td>Simple Linear Regression</td><td>이자율을 이용해서 집값을 예상하기</td><td>Linear Activation</td><td>Mean Squared Error</td><td>Mean Absolute Error</td></tr><tr><td>Multivariate Linear Regression</td><td>이자율 외 여러 변수를 이용해서 집값을 예상하기</td><td>Linear Activation</td><td>Mean Squared Error</td><td>Mean Absolute Error</td></tr><tr><td>Binary Classification</td><td>해당 이미지는 고양이일까요? 아닐까요?</td><td>Sigmoid / Logistic</td><td>Binary Cross Entropy</td><td>Binary Accuracy</td></tr><tr><td>Multi-Class Classification</td><td>해당 이미지는, 강아지일까요? 고양이일까요? 자동차일까요??</td><td>Softmax</td><td>Categorical Cross Entropy</td><td>Categorical Accuracy</td></tr></tbody></table> <div>2. 딥러닝 기본 구조1</div> <div>3. 딥러닝 기본구조2</div>	Model	Example	[Output] Activation Fun.	Loss Fun.	Metric	Simple Linear Regression	이자율을 이용해서 집값을 예상하기	Linear Activation	Mean Squared Error	Mean Absolute Error	Multivariate Linear Regression	이자율 외 여러 변수를 이용해서 집값을 예상하기	Linear Activation	Mean Squared Error	Mean Absolute Error	Binary Classification	해당 이미지는 고양이일까요? 아닐까요?	Sigmoid / Logistic	Binary Cross Entropy	Binary Accuracy	Multi-Class Classification	해당 이미지는, 강아지일까요? 고양이일까요? 자동차일까요??	Softmax	Categorical Cross Entropy	Categorical Accuracy
	Model	Example	[Output] Activation Fun.	Loss Fun.	Metric																					
	Simple Linear Regression	이자율을 이용해서 집값을 예상하기	Linear Activation	Mean Squared Error	Mean Absolute Error																					
	Multivariate Linear Regression	이자율 외 여러 변수를 이용해서 집값을 예상하기	Linear Activation	Mean Squared Error	Mean Absolute Error																					
Binary Classification	해당 이미지는 고양이일까요? 아닐까요?	Sigmoid / Logistic	Binary Cross Entropy	Binary Accuracy																						
Multi-Class Classification	해당 이미지는, 강아지일까요? 고양이일까요? 자동차일까요??	Softmax	Categorical Cross Entropy	Categorical Accuracy																						

CNN 모델 실습

1. CNN 이용한 동물 분류



이미지 읽기 → RGB 픽셀로 디코딩 → 텐서화



ImageDataGenerator API 사용

```
train_datagen = ImageDataGenerator
train_generator = train_datagen.flow_from_directory
model.fit_generator (구버전)
model.fit (신버전 tensorflow2.x)
```

오후

ImageDataGenerator API

라벨 읽어 올 때 **MODE**

- **class_mode**: one of "categorical", "binary", "sparse", "input", "other" or None. Default: "categorical". Mode for yielding the targets:
 - "binary" : 1D numpy array of binary labels,
 - "categorical" : 2D numpy array of one-hot encoded labels. Supports multi-label output.
 - "sparse" : 1D numpy array of integer labels,
 - "input" : images identical to input images (mainly used to work with autoencoders),
 - "other" : numpy array of `y_col` data,
 - None , no targets are returned (the generator will only yield batches of image data, which is useful to use in `model.predict_generator()`).

2. CNN 이용한 cifar-10 분류

```
argu_train_datagen = tf.keras.preprocessing.image.ImageDataGenerator(
    rotation_range=40,
    width_shift_range=0.2, ## 가로로 이동 비율
    height_shift_range=0.2, ## 세로로 이동 비율
    shear_range=0.2, # 전단의 각도
    zoom_range=0.2, ## 확대와 축소 범위 [1-0.2 ~ 1+0.2]
    horizontal_flip=True ## 수평기준 플립
)

valid_datagen = tf.keras.preprocessing.image.ImageDataGenerator()
test_datagen = tf.keras.preprocessing.image.ImageDataGenerator()
```

- **rotation_range** : 이미지 회전 범위 (degrees)
- **width_shift**, **height_shift** : 그림을 수평 또는 수직으로 랜덤하게 평행 이동시키는 범위 (원본 가로, 세로 길이에 대한 비율 값)
- **rescale** : 원본 영상은 0~255의 RGB 계수로 구성되어 있는데, 이 같은 입력값은 모델을 효과적으로 학습 시키기에 너무 높습니다 (통상적인 learning rate를 사용할 경우). 그래서 이를 1/255로 스케일링하여 0~1 범위로 변환시켜줍니다. 이는 다른 전처리 과정에 앞서 가장 먼저 적용됩니다.
- **shear_range** : 임의의 전단 변형 (shearing transformation) 범위
- **zoom_range** : 임의 확대/축소 범위
- **horizontal_flip** : True로 설정할 경우, 50% 확률로 이미지를 수평으로 뒤집습니다. 원본 이미지에 수평 비대칭성이 없을 때 효과적입니다. 즉, 뒤집어도 자연스러울 때 사용하면 좋습니다.
- **fill_mode** : 이미지를 회전, 이동하거나 축소할 때 생기는 공간을 채우는 방식

Exersice : 04_01_CNN_classification_augmentation.ipynb Practice : P_04_01_dog_cat_augmentation.ipynb

image augmentation 적용하여 오버피팅 방지 필요

