



9주차 중간고사 시험

```
df.shape -> 열, 행 정보
df.columns -> Index(['~~'])
df.info() -> # / Column / Non-Null Count / Dtype
df.describe() -> 해당 column에 따른 count, mean, std, min , 25%, 50%, 75%, max
```

데이터 전처리

```
df.isnul().sum()
X=df.iloc[:, :8]
y=df.iloc[:, 8]
```

```
from sklearn.preprocessing import StandardScaler
```

```
sc_X=StandardScaler() #표준화
X=sc_X.fit_transform(X) #only_train data set
X
```

transform() - traindata로 학습된 mean값과 variance값을 test data에 적용하기 위해 method사용함
test_Data의 값도 학습될 수 있기때문에

```
X_train,X_test,y_train,y_test=train_test_split(X,y,train_size=0.7,random_state=156)
```

은닉층 3개 , 각각의 노드 8, 15, 15 , 은닉층 활성화 함수 relu, 출력층 활성화 함수 sigmoid (sigmoid 이
진분류 출력함수) Dense 1

```
model=Sequential()
model.add(Dense(8,input_shape=(8,),activation='relu'))
#input에 들어가는 것 8 -> output 8개
model.add(Dense(15,activation='relu'))
model.add(tf.keras.layers.Dropout(0.5)) #과적합 방지
model.add(Dense(15,activation='relu'))
model.add(Dense(1,activation='sigmoid'))
```

Model: "sequential"

Layer (type) Output Shape Param

dense (Dense)	(None, 8)	72 (8 * 8 + 8)
dense_1 (Dense)	(None, 15)	135 (8*15 +8)
dropout (Dropout)	(None, 15)	0

dense_2 (Dense)	(None, 15)	240
dense_3 (Dense)	(None, 1)	16

=====

Total params: 463

Trainable params: 463

Non-trainable params: 0

Dropout

서로 연결된 연결망에서 0~1 사이의 확률로 뉴런을 제거하는 기법

Drop-out rate = 0.5 → 각 0.5의 확률로 제거될지 말지 랜덤하게 결정됨

하이퍼 파라미터 → 사람이 직접 넣어줘야하는 값임

Validation → Early Stop 기법

val_loss 손실이 가장 낮을때 학습을 중단시키고 싶어 !

```
import keras.callbacks
from tensorflow.keras import callbacks
earlystopping = callbacks.EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=20)
```

⇒ fit()시 사용됨

```
import keras.callbacks
from tensorflow.keras import callbacks

+ compile
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
earlystopping=callbacks.EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=20)
model.fit(X_train, y_train, batch_size=32, epochs=150, validation_data=(X_test, y_test), callbacks=[earlystopping])

model.evaluate(X_test, y_test)
```