70쪽 소스 제일 위 (추가)

#2. 모델 구성

55쪽 결과 중간부분 (수정)

acc : 0.908497….

mse : 0.908497…

65쪽 첫번째 소스 하단 (추가)

#RMSE 구하기

from sklearn.metrics import mean\_squared\_error

def RMSE(y\_test, y\_predict):

return np.sqrt(mean\_squared\_error(y\_test, y\_predict))

print("RMSE : ", RMSE(y\_test, y\_predict))

# R2 구하기

from sklearn.metrics import r2\_score

r2\_y\_predict = r2\_score(y\_test, y\_predict)

print("R2 : ", r2\_y\_predict)

72쪽 중간부분 (수정)

mse = model.evaluate(x\_test, y\_test, batch\_size=3)

mse = model.evaluate(x\_test, y\_test, batch\_size=1)

(batch의 크기를 evaluate시 변경하는 것은 상관은 없으나, 책의 본문의 문맥상 batch를 1로 잡는 것이 나을 것 같습니다.)

78쪽 밑에서 3번째 줄 (수정)

마지막을

마지막으로

99쪽 첫번째 소스 하단 (수정)

x\_predict = np.array(11,12,13)

x\_predict = np.array([11, 12, 13])

113쪽 첫번째 소스 하단 (수정)

x\_predict = np.array(11,12,13)

x\_predict = np.array([11, 12, 13])

116쪽 첫번째 소스 (수정)

\_train



x\_train

121쪽 2번째 소스 하단에 (추가)

model = Model(inputs=input1, outputs=output1)

129쪽 2번째 소스 밑에서 2번째 줄(수정)

dense1 = Dense(7)(dense1\_2)

dense1 = Dense(7)(dense1)

130쪽 마지막 소스 2군데 (수정)

output1 = Dense(3)(output1)

output1 = Dense(2)(output1)

output2 = Dense(3)(output2)

output2 = Dense(2)(output2)

171쪽 소스 상단 (추가)

#3. 훈련

model.compile(loss='mse', optimizer='adam', metrics=['mse'])

model.fit(x\_train, y\_train, epochs=100, batch\_size=1)

174쪽 소스 중간 (수정)

#2. 두 번

#2. 모델 구성

186쪽 첫번째 소스 (수정)

verbose=0

verbose=1

234쪽 마지막소스 마지막 줄 (수정)

print(x\_test.shape)



print(x\_pred.shape)

252쪽 중간 부분 (수정)

그렇다면 위 모델은

그렇다면 위 모델을

263쪽 3번째 소스 상단 (추가)

#2. 모델 구성

from keras.models import Sequential

from keras.layers import Dense, LSTM