식품영양학과 논문 데이터셋 준비

2021-07-20

컴퓨터과학과 황승현

저번주 한 일

- 2021-07-09
  - KOGES 데이터 원본 받음 (약 300MB)
- 2021-07-13
  - 데이터 정규화 사전 조사
- 2021-07-14
  - 수정한 데이터 받음

이번주 한 일

- 2021-07-16
  - 김혜림 박사님과 미팅
  - 질병관리청 교육 수강
- 2021-07-17
  - 데이터 CSV 파일을 일부 수정함
    - 변수 HYPERTENSION 추가
    - 일부 변수 수정
  - Pandas를 이용하여 CSV 파일 다룸
- 2021-07-19
  - Keras 당뇨병 발병 데이터 실습

변수 HYPERTENSION 추가

- =IF(OR(RC[-3]=2,RC[-2]>=140,RC[-1]>=90),2,1)
  - RC[-3]: AS1_DRUGHTCU
  - RC[-2]: AS1_BPLIE2S_A
  - RC[-1]: AS1_BPLIE2D_A

변수 수정

- 기반데이터 변수의 특징
  - 66666 = 조사 안 함
  - 77777 = 해당 없음
  - 99999 = 미상/무응답/미측정
  - 그 외, 비어 있는 값.
- 변수의 특성에 맞게 0, NA 등으로 수정

Pandas 라이브러리

- Pandas 라이브러리
  - opensource data analysis and manipulation
  - 지난 학기 Pandas를 다룬 경험이 있어 사용함
- Pandas 데이터프레임
  - 열과 행으로 볼 수 있는 NumPy Array
  - TensorFlow와 호환
- Pandas로 한 것.
  - 결측 값 제거
  - 데이터 수정, 분할
  - 정규화

Keras
당뇨병 예측

Attribute no.	Attribute
1	Number or times pregnant (NTP)
2	Plasma glucose concentration (PGC)
3	Diastolic blood pressure (mmHg) (DBP)
4	Triceps skin-fold thickness (mm) (TSFT)
5	2-h serum insulin (mu U/mL) (H2SI)
6	Body mass index (kg/m2) (BMI)
7	Diabetes pedigree function (DPF)
8	Age
9	Class 0 and 1 (Diagnosis of type 2 diabetes

- 피마족 인디언의 당뇨병 발병 데이터셋
  - 여러 요소로 당뇨병의 발병확률을 예측한다.
- 고혈압 모델 작성에 참고할 예정
- 소스코드 참고

데이터셋

```
dataset = pd.read_csv('year0_NA.csv', index_col=0, na_values=['NA'])
# pandas multi index 를 이용하여 기반, 2, 5, 8 한번에 다룰 예정
# 범주형: 1, 3, 4 ~ 7, 9, 12 ~ 19, 21, 22
# 연속형: 2, 8, 10, 11, 20, 23 ~ 52 ; 정규화 함
# 고혈압 변수: 53

df = dataset.dropna()  # 10000명 중 결측값 없는 사람만 추출 약 1000명
normalization_df = (df - df.mean()) / df.std()

ctn = df.reindex(columns=['AS1_AGE', 'AS1_TOTALC', 'AS1_HVSMAM', 'AS1_HVSMDU',
                          'AS1_SLPAMTM', 'AS1_RGMEALFQA', 'AS1_HEIGHT', 'AS1_WEIGHT',
                          'AS1_B01', 'AS1_B02', 'AS1_B03', 'AS1_B04', 'AS1_B05',
                          'AS1_B06', 'AS1_B07', 'AS1_B08', 'AS1_B09', 'AS1_B10',
                          'AS1_B11', 'AS1_B12', 'AS1_B13', 'AS1_B14', 'AS1_B15',
                          'AS1_B16', 'AS1_B17', 'AS1_B18', 'AS1_B19', 'AS1_B20',
                          'AS1_B21', 'AS1_B23', 'AS1_B24', 'P1', 'P2', 'P3', 'P4'
                          ])
normalization_ctn = (ctn - ctn.mean()) / ctn.std()
print(normalization_ctn.head())

X = normalization_df.drop(['HYPERTENSION'], axis=1)  # 고혈압을 제외한 나머지 변수
print(X.head())

y = normalization_df['HYPERTENSION']  # 고혈압 변수
print(y.head())
# 데이터 샘플링으로 훈련 데이터 나눔

# TensorFlow를 이용한 모델 예측
# model = tf.keras.models.Sequential([
```

Keras 당뇨병 소스코드

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.3,random_state=9)

X_val,X_test,y_val,y_test = train_test_split(X_test, y_test, test_size=0.5, random_state=123)

print(X_train.shape)
print(y_train.shape)

print(X_val.shape)
print(y_val.shape)

print(X_test.shape)
print(y_test.shape)

model = Sequential()

model.add(Dense(20, input_dim=8, activation='relu'))
model.add(Dropout(0.3))
model.add(Dense(8, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(1, activation='sigmoid'))

print(model.summary())

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

batch_size = 16
epochs = 100

history = model.fit(X, y, epochs=epochs,
                    batch_size=batch_size,
                    validation_data=(X_val, y_val), shuffle=True, verbose=1)

train_accuracy = model.evaluate(X_train, y_train)
test_accuracy = model.evaluate(X_test, y_test)

print(train_accuracy)
print(test_accuracy)
```

다음주 할 일

- 고혈압 예측 모델 구현
  - Keras 당뇨병 모델 참고
- 식품영양학과 김혜림 박사님과 미팅
  - 전처리된 데이터 설명
  - 모델 시연