강원지역혁신플랫폼

기계학습

Machine Learning

선형 회귀 분석 실습(1)

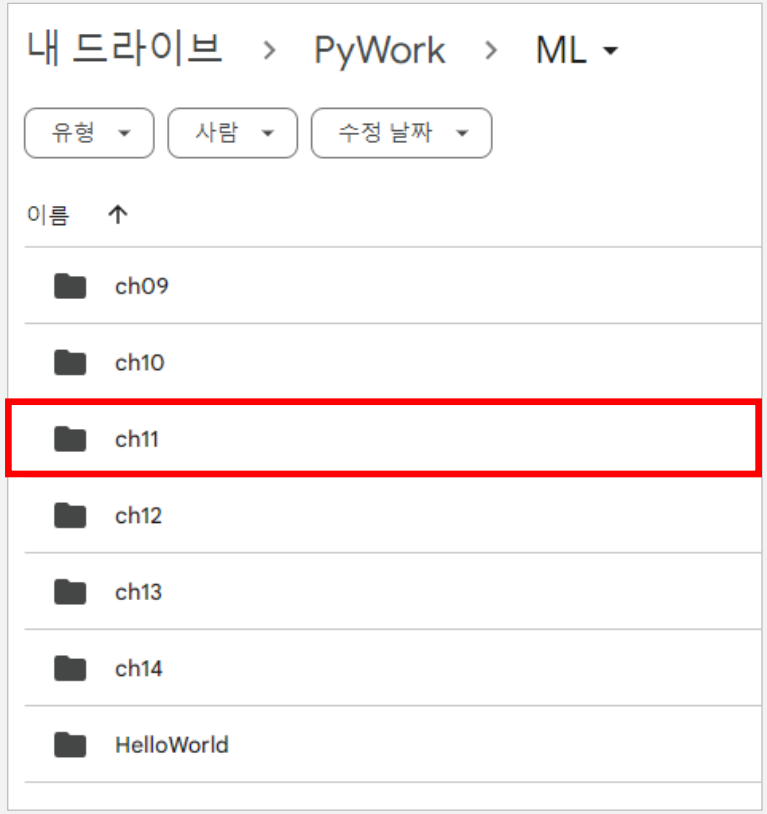
학습목표

백화점 고객 샘플 데이터 집합으로
군집 분석을 구현할 수 있습니다.

01 | 실습

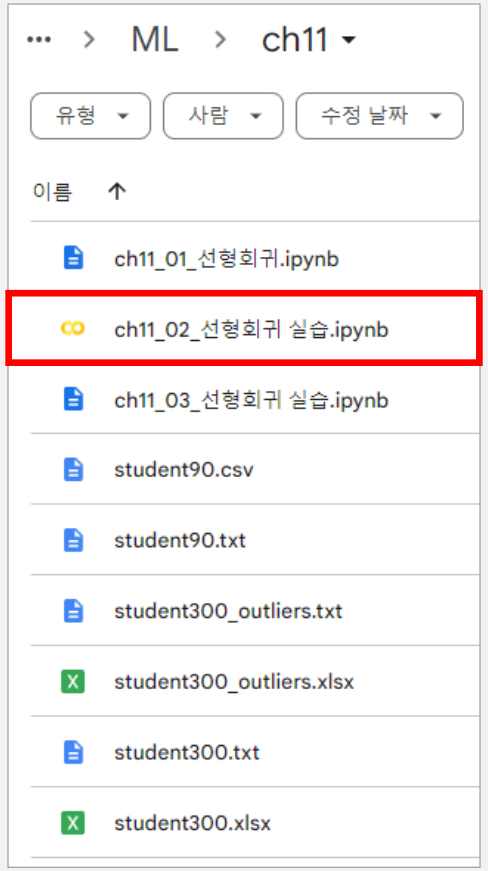
(권장) 아래와 같은 경로에 실행 소스가 존재하면 환경 구축 완료

- 구글 드라이브 "PyWork > ML" 폴더로 이동함
  - 아래의 [ch11] 폴더를 클릭하면 됨



01 | 실습

- “ML > ch11 >” 폴더를 클릭함
  - 아래의 [ch11_02_선형회귀 실습.ipynb] 스크립트를 클릭함



02 | 단순한 선형 회귀 실습

단순한 선형 회귀 실습

- 대학생 300명의 키와 몸무게 데이터 셋으로 선형 회귀 분석을 수행해보자.
  - 이 데이터로 키로 몸무게를 예측하는 단순 선형 회귀 모델을 만든다.
    - 여기서는 이상치 데이터를 전처리 하지 않음
    - CLRM(Classical Linear Regression Model)모델의 가정은 무시함
    - 간단하게 산포도, 회귀직선, 신뢰구간, 모델 학습 및 평가, 예측을 수행함
      - 예측은 나의 키로 몸무게를 예측함

성명	성별	학년	키(cm)	몸무게(kg)	취미
학생1	남	1	170.4	69.1	게임
학생2	여	3	169.3	62.0	음악
…	…	…	…	…	…

02 | 단순한 선형 회귀 실습

다음은 대학생 300명의 키와 몸무게 데이터셋을 읽어오는 코드이다.

- 실행결과 데이터 형상은 (300, 6)인 것을 알 수 있음

```
std = pd.read_excel(os.getcwd()+'/student300_outliers.xlsx')
print(std.shape)      # (300, 6)
print(std.info)
```

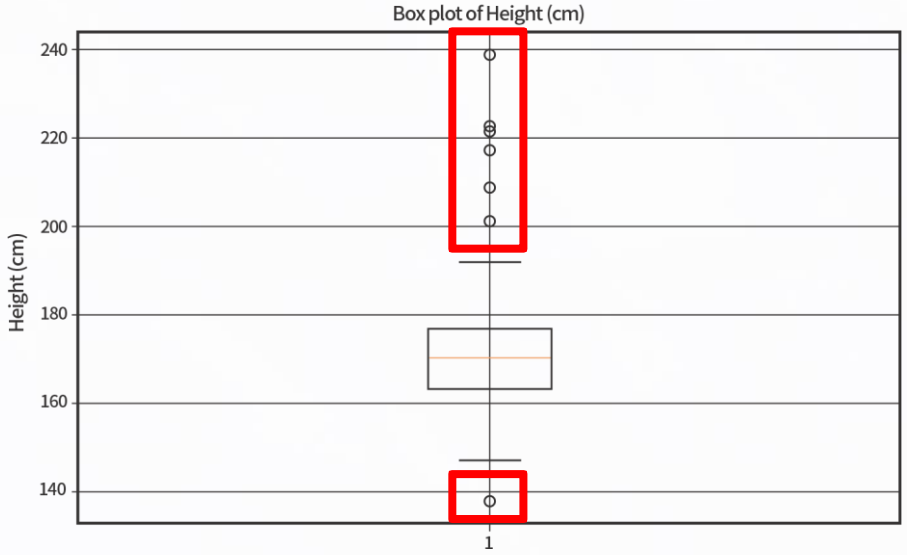
```
(300, 6)
<bound method DataFrame.info of         성명 성별  학년  키(cm)  몸무게(kg)  취미
0      학생1  남   1  170.5     69.0  게임
1      학생2  여   1  163.5     51.8  독서
2      학생3  남   3  191.4     60.2  음악
3      학생4  남   2  176.3     70.7  수영
4      학생5  남   2  149.7     57.1  수영
..     ...  ..  ..    ...      ...  ..
295  학생296  여   4  170.5     62.2  음악
296  학생297  여   1  172.6     63.7  등산
297  학생298  남   3  161.0     65.8  등산
298  학생299  남   4  176.4     49.8  수영
299  학생300  여   2  153.4     56.2  수영

[300 rows x 6 columns]>
```

02 | 단순한 선형 회귀 실습

다음은 키(cm)속성으로 상자그림을 그린 결과이다.

- 실행결과 키 속성에는 이상치 데이터가 포함된 것을 알 수 있음

```
plt.figure(figsize=(10, 6))
plt.boxplot(std['키(cm)'])
plt.title('Box Plot of Height (cm)')
plt.ylabel('Height (cm)')
plt.grid(True)
plt.show()
```

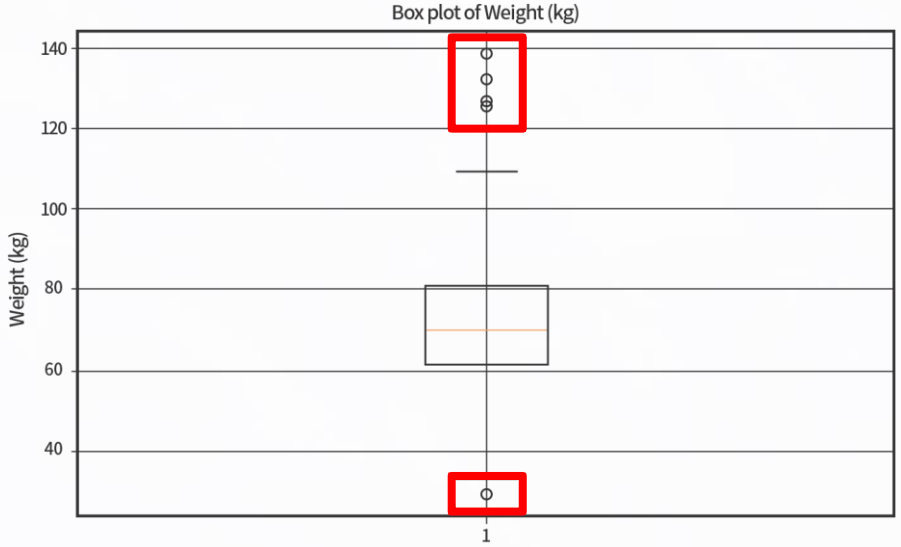


02 | 단순한 선형 회귀 실습

다음은 몸무게(kg)속성으로 상자그림을 그린 결과이다.

- 실행결과 몸무게 속성에는 이상치 데이터가 포함된 것을 알 수 있음

```
plt.figure(figsize=(10, 6))
plt.boxplot(std['몸무게(kg)'])
plt.title('Box Plot of Weight (kg)')
plt.ylabel('Weight (kg)')
plt.grid(True)
plt.show()
```



02 | 단순한 선형 회귀 실습

- 다음은 대학생 300명의 **키와 몸무게 데이터셋의 산점도** 그래프이다.
  - 여기에서 **키와 몸무게 평균**도 함께 표시함

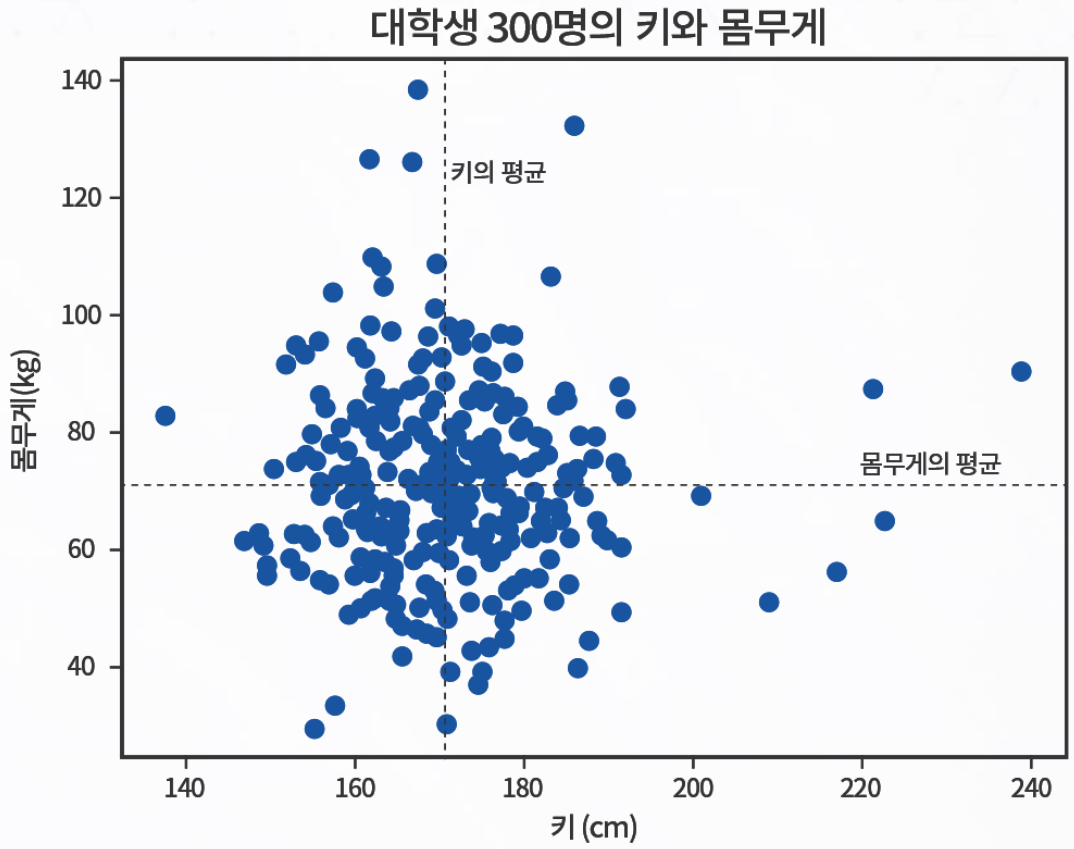
```
# 몸무게 평균
w_avg = np.mean(std['몸무게(kg)'])
print('몸무게 평균:', w_avg)

# 키 평균
h_avg = np.mean(std['키(cm)'])
print('키 평균:', h_avg)

# 키와 몸무게로 산점도 그리기
plt.scatter(std['키(cm)'], std['몸무게(kg)'])
plt.title('대학생 300명 키와 몸무게', fontsize=16)
plt.xlabel('키(cm)', fontsize=12)
plt.ylabel('몸무게(kg)', fontsize=12)
plt.axhline(w_avg, color='gray', linestyle='--', linewidth=1)
plt.axvline(h_avg, color='gray', linestyle='--', linewidth=1)
plt.text(171,123,"키의 평균")
plt.text(220,73,"몸무게의 평균")
plt.show()
```

02 | 단순한 선형 회귀 실습

✦ 아래 그림과 같이 몸무게 평균은 약 71kg, 키 평균은 약 170cm인 것을 알 수 있음

› 산점도 그래프에서 키와 몸무게 데이터의 분산이 큰 것을 알 수 있음

몸무게 평균: 71.1021829948021
키 평균: 170.65868600237798



02 | 단순한 선형 회귀 실습

다음은 대학생 300명의 키와 몸무게 데이터 셋으로 산점도에 회귀직선을 추가한다.

- 여기에서 키와 몸무게 평균도 함께 표시함

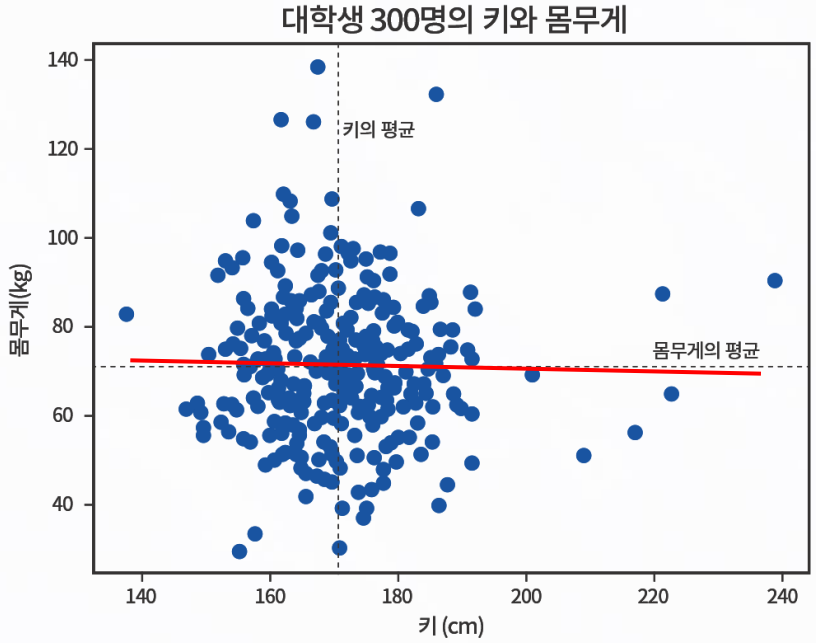
```
# 몸무게 평균
w_avg = np.mean(std['몸무게(kg)'])

# 키 평균
h_avg = np.mean(std['키(cm)'])

# x를 설명변수, y를 반응변수로 하는 1차 회귀 곡선(즉 직선을 적합)
b1, b0 = np.polyfit(std['키(cm)'], std['몸무게(kg)'], 1)  # 지울기(=b1), 절편(=b0)을 반환
print('b0=', b0, 'b1=', b1)
fit = b0 + b1 * std['키(cm)']

# 키와 몸무게로 산점도, 회귀직선 그리기
plt.scatter(std['키(cm)'], std['몸무게(kg)']) # 산점도
plt.plot(std['키(cm)'], fit, color='red')    # polyfit() 함수 : 절편, 기울기 계산
plt.title('대학생 300명 키와 몸무게', fontsize=20)
plt.xlabel('키(cm)', fontsize=14)
plt.ylabel('몸무게(kg)', fontsize=14)
plt.axhline(w_avg, color='gray', linestyle='--', linewidth=1)
plt.axvline(h_avg, color='gray', linestyle='--', linewidth=1)
plt.text(171,123,"키의 평균")
plt.text(220,73,"몸무게의 평균")
plt.show()
```

02 | 단순한 선형 회귀 실습

› 실행결과 산점도에 회귀직선이 추가된 것을 볼 수 있음

학생 몸무게 = 74.433 – 0.0195 * 학생의 키
(74.433: 절편, 0.0195: 계수)

몸무게 평균: 71.1021829948021
키 평균: 170.65868600237798
b0= 74.43302974455192 b1= -0.019517592850230 93



02 | 단순한 선형 회귀 실습

다음은 대학생 300명의 키와 몸무게 데이터 셋으로 산점도에 회귀직선과 신뢰구간을 그려보자.

- 여기에서 키와 몸무게 평균, 신뢰구간은 유의수준 95%로 한다.

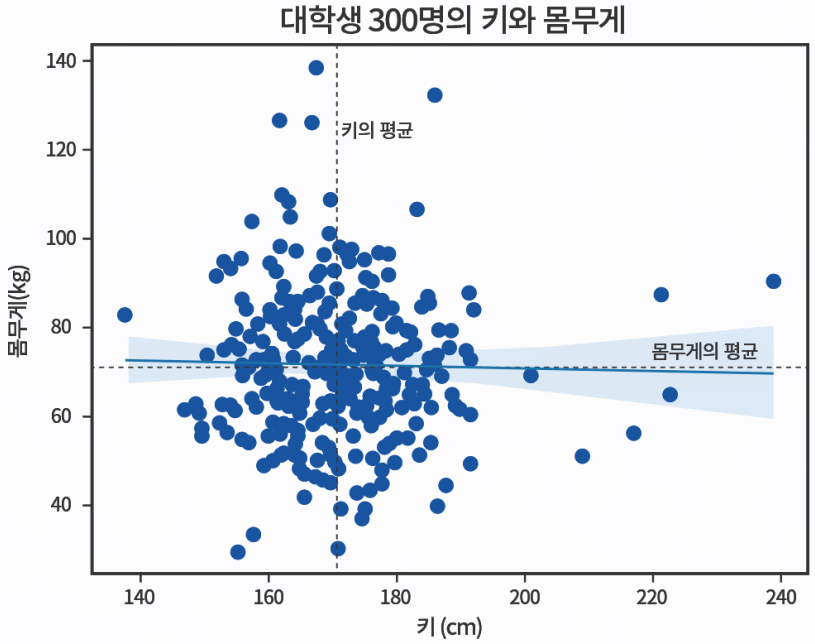
```
# 몸무게 평균
w_avg = np.mean(std['몸무게(kg)'])

# 키 평균
h_avg = np.mean(std['키(cm)'])

# x를 설명변수, y를 반응변수로 하는 1차 회귀 곡선(즉 직선을 적합)
b1, b0 = np.polyfit(std['키(cm)'], std['몸무게(kg)'], 1)  # 지울기(=b1), 절편(=b0)을 반환
print('b0=', b0, 'b1=', b1)
fit = b0 + b1 * std['키(cm)']

# 키와 몸무게로 산점도, 선형회귀선, 95% 신뢰구간 그리기
plt.scatter(std['키(cm)'], std['몸무게(kg)'])    # 산점도
sns.regplot(x='키(cm)', y='몸무게(kg)', data=std)          # 회귀직선
plt.title('대학생 300명 키와 몸무게', fontsize=20)
plt.xlabel('키(cm)', fontsize=14)
plt.ylabel('몸무게(kg)', fontsize=14)
plt.axhline(w_avg, color='gray', linestyle='--', linewidth=1)
plt.axvline(h_avg, color='gray', linestyle='--', linewidth=1)
plt.text(171,123,"키의 평균")
plt.text(220,73,"몸무게의 평균")
plt.show()
```

02 | 단순한 선형 회귀 실습

› 실행결과 산점도에 회귀직선, 신뢰구간(95%)이 추가된 것을 볼 수 있음

학생 몸무게 = 74.433 (절편) − 0.0195 (계수) * 학생의 키

몸무게 평균: 71.1021829948021
키 평균: 170.65868600237798
b0= 74.43302974455192 b1= -0.019517592850230 93



02 | 단순한 선형 회귀 실습

다음은 대학생 300명의 키와 몸무게 데이터 셋으로 **모델 생성 및 학습**을 수행하는 코드이다.

- 모델 생성 및 학습결과 다음과 같은 **회귀식**이 계산된 것을 알 수 있음

학생 몸무게 = 74.433 (절편) − 0.0195 (계수) * 학생의 키

```
# 모형 생성 및 학습하기
model_lr = LinearRegression().fit(np.c_[std['키(cm)']], np.c_[std['몸무게(kg)']])

# 회귀 계수 : 절편, 기울기
print("intercept=", model_lr.intercept_)   # 절편 intercept= [74.43302974]
print("coef=", model_lr.coef_)             # 기울기(계수) coef= [[-0.01951759]]

# 학생 몸무게(kg) = 74.43302974 - 0.01951759 * 학생의 키(cm)
```

02 | 단순한 선형 회귀 실습

다음은 학습된 모델로 **모델 성능평가**를 수행하는 코드이다.

- 여기서는 모델 성능평가 지표로 MSE를 이용함
  - 실행결과 MSE = 267.791인 것을 볼 수 있음

```
# 모델의 성능 확인
mse = mean_squared_error(y_true = std['몸무게(kg)'], y_pred = model_lr.predict(np.c_[std['키(cm)']]))
mse          # 267.79104387996335
```

02 | 단순한 선형 회귀 실습

- 다음은 학습된 모델로 예측을 수행하는 코드이다.
  - 여기서는 새로운 학생의 키가 175cm 임
    - 실행결과 몸무게가 약 71.02(kg)인 것을 볼 수 있음

```
X_new = [[175]]                                  # 새로운 학생의 키(cm) = 175
print("Predict=", model_lr.predict(X_new))       # 새로운 학생 키에 대한 예측 결과 = [[71.017451]]
```