05.01 확률론적 선형 회귀모형

6) 단일계수 t-검정 (single coefficient t-test)

정규화된 모수오차를 검정통계량으로 사용하면

$$\frac{\hat{w}}{se_i}$$

1. wi가 0인지 아닌지 검정 가능

$$H_0 : w_i = 0 (i = 0, 1, \ldots, K-1, K\text{개의가중치})$$

2. wi가 40인지로 놓고도 검증 가능

*코드 : print(result.t_test("X1 = 40)) *X1=40이란 건, X1가중치가 40 이라고 적는 것과 같은 의미

$$H_0 : w_i = 40 (i = 0, 1, \ldots, K-1, K\text{개의가중치})$$

3. w1 = w2 인지도 검증 가능

*코드 : print(result.nottem.t_test("C(month)[01] = C(month)]02"))

$$H_0 : w_1 = w_2 (i = 0, 1, \ldots, K-1, K\text{개의가중치})$$

7) 회귀분석 F-검정

- 단일계수 t-검정 : single coefficient t-test. 단일 계수(w_i) 에 대한 검정

- F-검정 : 모든 가중치가 다 쓸모 없다(종속변수와 feature들은 모두 상관성이 없다.)
"이 모델은 쓸모 없다" 를 반론하고 싶을 때, 사용 (이 가설이 아주 낮은 p값으로 기각되어야 좋음)
보통 "어느 모델이 성능이 더 좋다"를 증명할 때, 사용하는 검정 (성능이 좋을 수록 p값이 낮게 나올 것)

$$H_0 : w_0 = w_1 = w_2 = \ldots = w_{k-1} = 0$$

- 현실적으로 이런 H_0 가설은 받아들여질 가능성은 없다.
- 모두 쓸모 없다는 가설이 reject가 되더라도 0.01 로 기각, 0.000000001로 기각 되느냐의 차이
- 결국, 0.0000001로 기각되어야, p-value가 더 작게 기각되어야 역설적으로 '모델'이 쓸모 있다는 확률적 증명이 된다

8) statsmodel 패키지 회귀분석 결과표 해석 (05.01 14p)

05.02 회귀분석의 기하학

- 투영행렬, 햇행렬, 영향도행렬
- 잔차행렬

중요하기 때문에, 유도하는 증명 해봐야함

05.03 레버리지와 아웃라이어

지금까지는 데이터 행렬 X 의 열단위 접근 (개별 feature에 대한 이야기(가중치))

이제부터는 데이터 행렬 X 의 행단위 접근 (개별 데이터에 대한 이야기)

- 개별 데이터 표본 하나하나가 회귀분석 결과에 미치는 영향력 분석
  : 레버리지 분석 / 아웃라이어 분석

1) 레버리지

레버리지 : 실제 종속변수 값 y 가 $\hat{y}$ 에 미치는 영향

레버리지 : 영향도 행렬(H)의 대각성분 h_{ii}

$$\hat{y} = Hy$$

레버리지의 성질

1. $$0 \leq h_{ii} \leq 1$$
2. $$\mathrm{tr}(H) = \sum_{i}^{N} h_{ii} = \mathrm{K}$$

[시사점]

1. 현실적으론 각각의 레버리지값(H의 대각성분)은 대부분 매우 작게 나오기 마련

why? 현실에선

데이터의 갯수(대각성분의 갯수) N >> 모수의 갯수(가중치, 열의 갯수) K

작은 수 K를 N 으로 쪼개서 가져가면, 각 대각성분은 그만큼 작아질 수 밖에!

2. 레버리지의 평균값

$$h_{ii} \approx \frac{K}{N}$$

보통, 이 평균값의 2~4배 보다 레버리지 값이 크면, 레버리지가 크다고 이야기 함

2) statsmodels를 이요한 레버리지 계산

코드 (4page, 5page)

[시사점]

1. 무리지어 있지 않은 애들이 레버리지가 큼
2. 큰 레버리지 특징 : 그 지역에서 대표성 큰 애들 (혼자 그 구간을 담당하는 데이터)

3) 레버리지 영향

레버리지의 영향 크기 : 해당 데이터의 잔차 크기에 달려있음 (6,7 page)

[시사점]

1. 데이터 제거 시 주의사항

   1) **'레버리지', '잔차' 모두 큰 데이터를 빼면**, 모델(회귀선) 자체가 흔들릴 수 있는 영향력을 갖기 때문에, 주의해야함

   그런데 '잔차'는 우리가 아는 그 잔차가 아닌, '표준화된 잔차'를 봐야 한다!

4) 아웃라이어

아웃라이어 : '표준화된 잔차'가 큰 데이터

$y - \hat{y} = e$(잔차)

표준화된 잔차 : 잔차를 표준화한 것

4-1) 표준화 잔차

- 데이터 각각의 개별적인 영향들을 제거해, 모든 데이터의 잔차를 표준화된 상태에서 비교할 수 있게 함
  (개별적인 영향 : 개별 데이터의 레버리지 값)

개별 데이터의 잔차 ==>> 표준편차가 레버리지에 따라 달라짐 (9page)

- 원래 목적대로, 실 데이터 - 모델 간의 차이를 보려면, 이 개별적인 요인들을 다 제거해준 값으로 비교해줘야 공정한 비교!
  - 레버리지가 큰 데이터는 잔차크기가 상대적으로 작게 나옴. 모델과 차이가 큼에도 불구하고

4-2) So, 어떤 데이터를 제거해야 하는 가?

- 표준화된 잔차로 본 아웃라이어를 제거
  *대개는, 표준화 잔차가 2~4보다 크면 아웃라이어로 봄
  *엄밀하게는, Cook's Distance -> Fox' outlier recommendation을 기준으로 판단

- 대신, 레버리지가 큰 데이터는 일단 다시 한 번 살펴봐야 함(모델, 회귀선에 주는 영향이 크기 때문)

5) statsmodels 를 이용한 '표준화 잔차' 계산

- regressionresult 객체의 regid 속성

(10page)

5-1) Cook's Distance

- (10p 하단)

아웃라이어 판단 기준

(11p)

5-2) 레버리지가 큰 아웃라이어 시각화

- plot_leverage_resid2

11p

- influence_plot

12p

5-3) Cook's distance - Fox에 의한 아웃라이어 판단

- 제거 대상 (잔차 or 레버리지가 기준 이상으로 큰 데이터)

13p