# Week 3 : 회귀분석의 변형

### < 목차 >



---

## 0. 복습

2주차에 회귀분석의 기본가정 4가지와 각각에 대한 진단법, 처방법에 대해 알아보았습니다.

→ 모델의 선형성, 오차의 정규성, 등분산성, 독립성

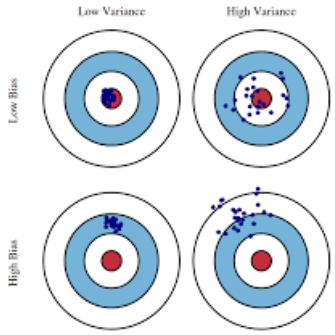
- 가정이 지켜졌는지 여부를 잔차 플랏으로 시각적으로 확인할 수도 있지만, 검정의 방법을 통해 더 객관적으로 확인할 수 있었다.
- 선형성 → 선형성은 선형회귀식을 위한 핵심으로, 변수 변환과 다항회귀, 국소회귀 등의 모델 적합을 통해 해결할 수 있다.
- 정규성 → 정규성이 위배된다면 정규분포를 가정한 분포를 사용할 수 없어 검정 및 예측 결과를 신뢰하기 어려운 문제가 발생한다. ECDF(경험적 누적 밀도 함수)와 정규분포의 분포적 특성을 통해 정규성의 여부를 파악한 후에 변수 변환의 과정을 적용해준다.
- 등분산성 → 등분산성이 위배된다면 추정된 LSE를  BLUE로 사용할 수 없고, 1종 오류가 발생할 가능성이 있다. 주로 B-P test로 검정을 진행하며 변수 변환과 가중 회귀 제곱의 방법을 통해 처방 가능하다.
- 독립성 → 독립성이 위배된다면 시계열 특성 혹은 공간 자기상관을 의심해볼 수 있다. 더빈-왓슨 검정을 통해 위배 여부를 확인하고 설명변수 추가나 분석 모델 변경 등의 방법을 통해 해결할 수 있다.
- 다중공선성 → 다중공선성은 plot이나 VIF, 고유값 등을 통해 진단할 수 있었다. 다중공선성은 모델의 추정을 불안정하게 하며 해석에 영향을 주기 때문에 회귀분석의 기본 가정 못지않게 중요한 문제이다.
- 다중공선성의 해결하는 방법에는 **변수선택법**(Variable Selection), **차원축소**(Dimension Reduction), **정규화**(Regularization) 등이 있다.

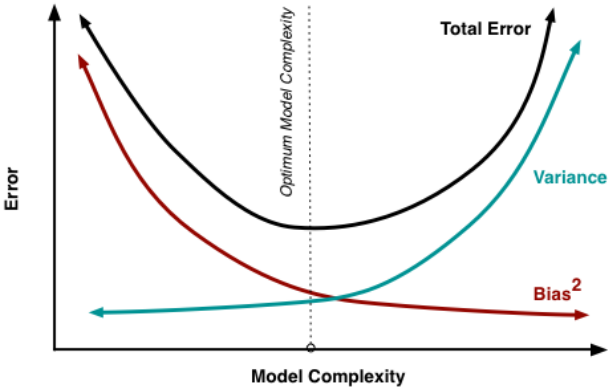
이번 주차에는 다중공선성을 해결하기 위해 언급된 3가지 방법들과, 독립성이 위배될 때 사용하게 되는 공간 회귀에 대해 알아봅니다. 벌써 3주차라니 시간이 참 빠르단 게 느껴지네요 ㅎㅎ 마지막까지 파이팅!!

---

좋은 모델이란 무엇일까? 미래 데이터에 대한 예측 성능이 좋은 모델이라고 할 수 있을 것이다. 이는 미래 데이터에 대한 기대 오차가 낮은 모델을 의미한다. Expected MSE는 $Irreducible\ Error + Bias^2 + Variance$ 로 계산된다. (데마팀 클린업 1주차 참고)

다중공선성이 일으키는 문제 중 하나가 OLS 추정량의 분산을 매우 크게 증가시킨다는 것이었다. 분산이 크게 증가하게 된다면 추정량의 변동폭이 매우 크게되어 예측이 불안정해지고, 이는 좋은 모델이라고 하기 어려울 것이다.



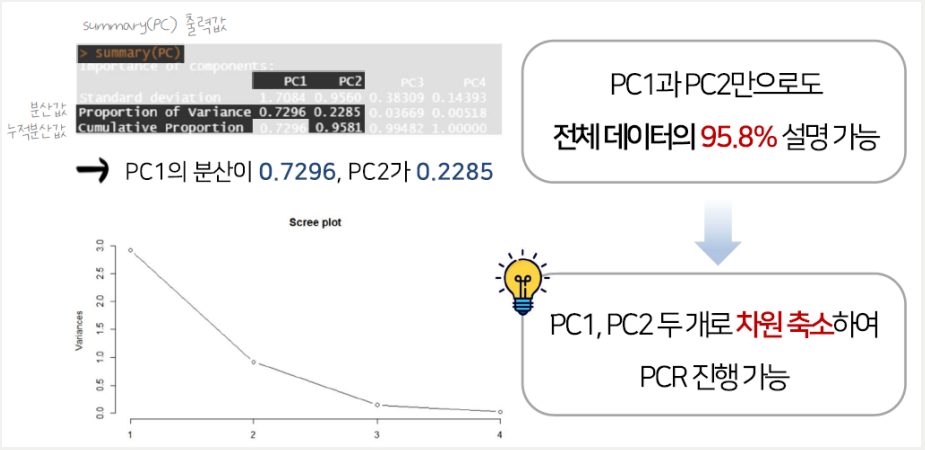

그러나 bias는 조금 포기하더라도 variance의 감소폭을 더 줄일 수 있다면, Expected MSE가 감소할 것이다. 변수 중 일부만을 사용하거나 $\beta$ 계수를 축소함으로써, bias를 조금 증가시키더라도 분산을 줄이는 것이 다음에 등장할 방법들의 기본적 아이디어라고 할 수 있다.

## 1. 차원 축소

- 차원축소 방법에는 PCA(Principle Component Analysis), PLS(Partial Least Square), 신경망 모형을 사용하는 AE(Autoencoder) 등이 있다.

📌 **차원 축소 (Dimension Reduction)** (선대팀 3주차 클린업 참고)

- **PCA (Principle Component Analysis, 주성분분석)**
  - 데이터들의 정보량(분산)을 최대한으로 보존하면서 직교하는 새로운 축을 찾는 방법.  기존의 X1와 X2는 상관관계가 강했지만, 직교하는 새로운 축을 찾아서 변수를 PC1, PC2로 바꿀 경우, 두 변수의 상관계수는 0이 된다.




즉 고차원의 데이터를 상관관계가 없는 저차원의 공간으로 축소시키는 방법이다!

하지만, 차원을 줄이는 과정에서 정보의 손실이 발생할 수 있고 개별 PC들의 해석이 어렵다는 단점이 있다.

- **PCR (Principle Component Regression, 주성분회귀)**
  - PCA를 통해 X변수를 Z로 변환해주고, 변환된 Z와 y에 대해 회귀분석을 적용해준다.
  - 다중공선성을 해결해주고, 적은 변수를 사용하기 때문에 과적합(Overfitting)을 방지한다.
  - 다만 다중공선성이 명확하지 않은 경우에는 성능이 떨어지고, 성능이 좋더라도 해석이 어려워지는 단점이 있다.

---

# 2. 변수선택법

## 1) 변수선택법이란?

- 변수선택법은 분석을 위해 고려한 수많은 변수들 중 적절한 변수의 조합을 찾아내는 방법이다. 우리에게 주어진 가능한 후보 변수(Candidate Regressor)들은 많고 그에 따라 후보

변수들의 조합도 많지만, 이 중 일부분만 중요하거나 예측에 유의미할 수 있다. 따라서 우리는 후보 변수들의 적절한 부분집합(subset)을 찾는 것을 목표로 한다.

- 변수선택법은 다중공선성이 존재할 때 많이 사용된다. 변수선택법을 통해 높은 상관관계를 가지는 변수를 제거할 수 있다. 물론 변수선택법을 했다고 다중공선성이 완벽히 제거되는 것은 아닐 수 있다.
- 변수선택법은 다중공선성이 발견되지 않더라도 사용될 수 있다. 변수선택을 통해 모델에 대한 해석력 증가, 혹은 최종 모델에 대한 확신을 얻을 목적 등으로 시행하기도 한다.
- 우리는 최대한 많은 변수들을 사용해서 y를 예측하기 위한 많은 정보를 포함하고 싶기도 하지만, 최대한 적은 변수들을 사용해서 모형의 분산을 줄이는 것을 원하기도 한다. 이 trade-off를 잘 고려해서 '최적의 회귀식(Best Regression Equation)'을 heuristic하게 찾는 방법이 변수선택법이다.
- 즉, 변수가 선택되고 제거되는 것에 논리성과 정당성을 부여해주는 방법이라고 할 수 있다.

## 2) 변수 선택 지표

변수 선택 알고리즘에 대해 배우기 전에, 먼저 변수를 어떤 기준으로 선택할지 알아야 한다.

- Partial F-test를 통한 변수선택

  1주차 클린업에서 Partial F-test를 통해 여러 변수들의 유의성을 검정했다. 이 검정을 통해 유의하지 않은 변수들을 없애는 방식으로 변수 선택을 진행할 수 있다. 하지만 Partial F-test는 Reduced Model(RM)에 있는 모든 변수가 Full Model(FM)에 있어야 한다는 단점이 있다.

  - 이런 모델들이 있다고 가정해보자.

    $model\ A : y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \quad \text{vs} \quad model\ B : y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3$

    $model\ A$ 의 설명변수가 $model\ B$의 설명 변수에 모두 포함되고 있다. 이 경우 Partial F-test를 통한 변수선택이 가능하다.

  - 하지만 변수가 모두 포함되지 않은 여러 모델을 비교해야 하는 경우도 많다.

    $model A : y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \quad \text{vs} \quad model B : y = \beta_0 + \beta_3 x_3 + \beta_4 x_4$

    $model\ A$과 $model\ B$는 사용된 변수가 전혀 다르므로 Partial F-test를 사용해 어느 모델이 더 설명력이 있는지 검정하지 못한다. 따라서 **포함 관계와 무관하게 Global하게 모델 간의 비교를 가능하게 해주는 기준이 필요하다.** 일반적인 상황에서 변수를 선택할 수 있도록 서로 비교가 가능한 지표를 알아보자.

※ 변수선택법의 핵심은 적은 변수로 데이터를 가장 잘 설명하는 모델을 찾는 것이다. 따라서 앞으로 소개할 변수선택의 지표들은 모델의 설명력과 변수의 개수를 모두 고려하는 형태로 구성되어 있다.

- **수정결정계수 ($R^2_{adj}$)**
  - 1주차 다중선형회귀의 적합성 검정 파트에서 등장했던 수정결정계수를 기준으로 변수선택을 할 수 있다. 수정결정계수의 식에는 설명력을 담당하는 결정계수와 변수 개수 패널티가 들어가기 때문에 이를 복합적으로 고려해줄 수 있다.
  - 당장의 SSE/RSS (Residual Sum of Square)를 충분히 작게 함과 동시에, 변수의 개수에 대한 penalty를 통해 줄이는 방식이 제일 직관적으로 간단해 보인다.

- **AIC (Akaike Information Criterion)**

  $AIC = -2log(Likelihood) + 2p$

  $AIC = nlog(2\pi\hat{\sigma}^2) + \frac{SSE}{\hat{\sigma}^2} + 2p$
  - 위의 AIC는 일반적인 계산식, 아래의 AIC는 정규분포 가정 하의 AIC이다.
  - $p$는 모델의 모수 개수다. 변수 개수에 따른 패널티를 부과하는 것을 의미한다.
  - $\hat{\sigma}^2$은 $\sigma^2$의 MLE이다
  - $Likelihood$가 커질수록 모델이 데이터를 잘 설명한다는 의미다. 그런데 $Likelihood$가 커지면 $AIC$는 작아지기 때문에, $AIC$가 낮을수록 더 좋은 모형이라고 해석할 수 있다.
  - $AIC$는 KL-Divergence(Kullback-Leibler 쿨백-라이블러 발산)의 추정치로 사용될 수 있다. KL-Divergence는 실제 데이터의 분포와 통계 모형이 예측하는 분포 사이의 차이를 의미하는데, KL-Divergence 값이 작을수록 통계 모형이 데이터의 참된 분포를 잘 묘사한다는 뜻이다. 하지만 우리는 데이터의 true distribution(실제 생성되는 형태)을 모르기 때문에 AIC를 이에 대한 추정치로 간주하여 이용한다.

    → AIC는 두 확률분포 사이의 차이를 표현한 것으로 실제 데이터의 분포와 모형이 예측하는 분포 사이의 차이를 의미한다. AIC가 작다는 것은 모형이 자료의 실제 분포와 비슷하게 생겼다는 것을 의미한다. 그런데 AIC는 실제 데이터의 분포(true model) 자체에는 관심이 없고 예측을 잘하는지에 대해서만 관심이 있다.

    실제 데이터의 분포 자체에 관심이 있는 것은 **BIC**이다!

- **BIC (Bayesian Information Criterion)**

  $BIC = -2log(Likelihood) + p \times log(n)$

  $BIC = nlog(2\pi\hat{\sigma}^2) + \frac{SSE}{\hat{\sigma}^2} + p \times log(n)$

- 위의 BIC는 일반적인 계산식, 아래의 BIC는 정규분포 가정 하의 BIC이다.
- p는 모델의 모수 개수다. 마찬가지로 변수 개수에 따른 패널티를 부과하는 것을 의미한다.
- $\hat{\sigma}^2$은 $\sigma^2$의 MLE이다.
- AIC와 다르게 데이터의 개수를 모수의 개수에 곱함으로써 AIC보다 더 큰 패널티를 부과한다. n>8이라면 BIC가 AIC보다 더 많은 페널티를 부여하기 때문에 변수 개수가 더 적은 모델을 선호한다.

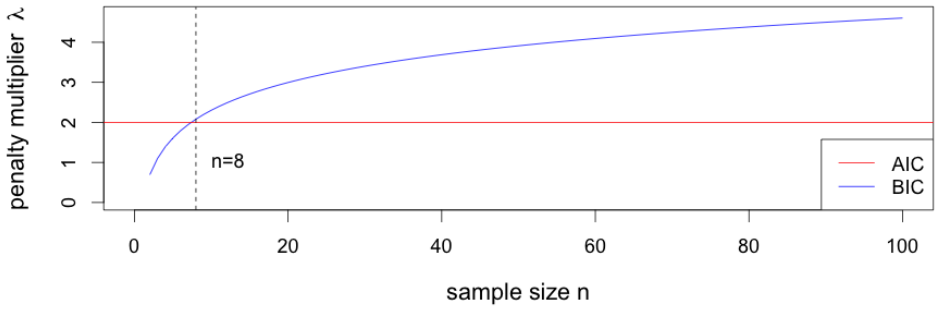
  → BIC가 AIC보다 변수 증가(복잡성)에 더 민감하게 반응하기 때문에 변수의 개수가 작은 것이 우선순위라면 AIC보다 BIC를 참고하는 것이 좋다.
- AIC와 같이, 작을수록 더 좋은 모형이다.
- BIC는 Bayes Factor(BF)의 로그 값에 대한 추정치로 이해할 수 있다. BF는 Consistency라는 성질이 있는데, 이는 비교 대상이 되는 모형들 중에 'True Model'이 있다는 가정 하에 관측치가 많아짐에 따라 참된 모형을 선택할 확률이 1에 가까워진다는 의미이다.

  → True Model을 고르는 목적일 경우 BIC를 사용해야 함!

더 깊은 이해를 위한 참고 링크

https://machinelearningmastery.com/probabilistic-model-selection-measures/

⇒ 정리 : AIC와 BIC를 모두 최소화한다는 것은 우도(likelihood)를 가장 크게 하는 동시에 변수의 개수는 가장 적은 최적의 모델(parsimonious & explainable)이라는 것이다!




AIC는 데이터 개수에 상관없이 패널티가 일정하지만, BIC는 데이터 개수가 커질수록 패널티도 함께 커진다. 따라서 BIC가 표본 크기가 커질수록 복잡한 모형을 더 강하게 구별하는 것이다. AIC는 실제 데이터의 분포와 모델이 예측하는 분포의 차이를 나타내기 때문에, AIC가 작으면 모델이 주어진 데이터의 진짜 분포와 유사하다는 뜻이지만 분포 자체보다는 예측에 중점을 둔다. BIC가 실제 데이터의 분포 자체를 판단하는 것이다.

따라서 예측을 위해서는 AIC를, 실제 데이터의 분포를 알고 싶거나 변수 증가에 민감한 데이터를 다룰 때에는 BIC를 사용한다.

물론, AIC와 BIC를 둘 다 보고 종합적으로 판단하는 것이 더 좋다. 고차원 데이터에서는 정확성이 떨어질 수 있고, AIC와 BIC 모두 각각 문제가 발생한다. 따라서 둘을 모두 고려해서 모형을 선택해야 한다.

*by Prof. Patrik Breheny(2016) 'High-Dimensional Data Analysis'*

그 외에도 $C_p = \frac{SSE_p}{\hat{\sigma}^2} + 2p - N$으로 계산되는 Mallow's Cp가 있다. 이는 정규성과 선형성의 가정 하에 AIC와 동일하다.

## 3) 변수 선택 방법

변수선택법은 모두 heuristic한(경험적인) 방법이다. 알고리즘에 따라 해당하는 모든 경우를 계산해서 제일 좋은 회귀식을 찾는 방법이다.

→ 그래서 계산량이 많다는 문제가 있다.

- **Best Subset Selection**

  가능한 모든 변수들의 조합을 다 고려하는 방법(All Possible Regression)이다. 변수의 개수가 $p$개라면, $2^p$개의 모형을 모두 적합하고 비교한다.

  ✅ **Best Subset Selection's Algorithm**

  1. $M_1, \cdots, M_p$개의 모형을 적합한다. 이때, $M_k(k = 1, 2, ..., p)$란 변수의 개수를 k개로 적합했을 때 적합한 회귀식 중 training error(주로 MSE)가 제일 작은 식이다.
  2. ($M_1$~$M_p$) $p$개의 모형 중 AIC 또는 BIC가 가장 작은 모형을 선택한다.
  3. 만약 AIC, BIC가 가장 작은 모형이 서로 다를 경우 다른 근거에 의해 두 개의 모형 중 하나를 선택한다. (주로 하나의 평가 기준을 두고 선택합니다.)

  - 장점

  가능한 모든 경우의 수를 고려하기 때문에 선택된 Best Model에 대한 더 신뢰할 수 있는 결과를 산출한다.

  - 단점

  변수의 개수가 p>40인 경우 계산 불가능하다.

  적당한 p에서도 많은 관측치를 지니고 있다면 모든 모델을 고려한 계산 비용이 많이 소모된다.

- **전진선택법 (Forward Selection)**

  Null Model $(y = \beta_0)$ 에서 시작해 변수를 하나씩 추가하는 방법이다.

  ✅ **Forward Selection's Algorithm**

  1. 상수항만을 포함하고 있는 모형인 Null Model$(y = \beta_0)$에서 시작해 $X_1$부터 $X_p$까지의 변수들 중에 어떤 것을 추가하는 것이 AIC와 BIC를 낮추는지 판단한다.
  2. 만약 1번의 과정에서 $X_1$이 선택되었다면, 이제 $y = \beta_0 + \beta_1 x_1$의 식에서 $X_2$부터 $X_p$까지의 변수들 중에 어떤 것을 추가하는 것이 AIC와 BIC를 낮추는지 판단한다.
  3. 이러한 과정을 반복하며 AIC와 BIC가 낮아지면 추가하고 더 이상 AIC와 BIC가 낮아지지 않는다면 프로세스를 중단한다.

  - 장점

  Best Subset Selection에 비해 계산이 매우 빠르다.

  변수의 개수가 관측치의 개수보다 많은 경우에도 사용 할 수 있다.

  - 단점

  Best Subset Selection처럼 가능한 모든 변수 조합을 고려하지는 않기 때문에 선택된 모형이 최적의 모형이라고 할 수 없다.

- **후진제거법 (Backward Elimination)**

  Full Model $(y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p)$에서 시작해 변수를 하나씩 제거하는 방법이다. Forward selection의 반대라고 생각하면 된다.

  ✅ **Backward Elimination's Algorithm**
  1. Full Model에서 시작해 $X_1$부터 $X_p$까지의 변수들 중에 가장 AIC와 BIC를 크게 낮추는 변수를 선택해 제거한다.
  2. 위의 과정을 반복하며 AIC와 BIC가 더 이상 낮아지지 않으면 프로세스를 중단한다.

  - 장점

  Best Subset Selection에 비해 계산이 매우 빠르다

  - 단점

Forward Selection보다 더 좋은 결과를 도출한다고 알려져 있지만, Best subset selection 방법과 마찬가지로 $p$>40 인 경우에는 사용할 수 없다.

그래도 Best Subset Selection처럼 가능한 모든 변수 조합을 고려하는 것은 아니기 때문에 선택된 모형이 최적의 모형이라고 할 수 없다.

- **단계적 선택법 (Stepwise Selection)**

  Forward Selection과 Backward Elimination 과정을 섞은 방법이다.

  Null model에서 시작할 수도 있고 Full Model에서 시작할 수도 있지만, 변수를 선택하거나 제거하는 경우를 모두 고려(조합)했을 때 AIC와 BIC가 감소하는 방향으로 움직인다.

  ✅ **Stepwise Selection's Algorithm 기본형**

  1. Null model 혹은 Full model에서 시작한다.
  2. 다른 변수 선택법들을 혼합하여 변수들을 제거 혹은 추가하며 모델을 평가한다.
  3. AIC/BIC가 가장 작은 모형을 선택한다.

  → **Stepwise Selection's Algorithm 예시**

  1. 먼저 forward selection 과정을 이용해 가장 유의한 변수들을 모델에 추가한다.
  2. 그 후 나머지 변수들에 대해 Backward Elimination을 적용해 새롭게 유의하지 않게 된 변수들을 제거한다.
  3. 제거된 변수는 다시 모형에 포함되지 않으며, 모형에 유의하지 않은 설명변수가 존재하지 않을 때까지 1번과 2번 과정을 반복한다.

  → 전진선택법을 사용할 때 한 변수가 선택되면, 이미 선택된 변수 중 중요하지 않은 변수가 있을 수 있다. 그래서 전진선택법의 각 단계에서 이미 선택된 변수들의 중요도를 다시 검사하여 중요하지 않은 변수를 제거하는 방법이다.

  - 장점

  Best Subset Selection에 비해 계산이 매우 빠르다.

  - 단점

  Stepwise selection은 변수를 제거 혹은 추가 모두를 할 수 있다는 점에서 유연하게 움직일 수 있지만, 모든 변수 조합을 고려하는 것이 아니기 때문에 Best Model이라고 할 수는 없다.

https://towardsdatascience.com/stopping-stepwise-why-stepwise-selection-is-bad-and-what-you-should-use-instead-90818b3f52df : Stepwise selection의 단점(참고)

1. $R^2$ 값이 크게 편향되어 추정된다.
2. F 통계치의 실제 분포가 가정된 F 분포로부터 벗어난다.
3. 모수 추정치들의 표준오차가 실제보다 작게 추정된다.
4. 이의 결과로 모수들의 신뢰구간이 실제보다 narrow하게 보고된다.
5. 다중비교로 인해 p-value가 매우 작게 추정되고 교정하기도 어렵다.
6. 모수 추정치들이 0이 아닌 것으로 편향되기 쉽다.
7. 다중공선성 문제가 심각해진다.

*by Frank Harrell(2001) 'Regression Modeling Strategies'*

- 정리

비록 Best Subset Selection을 제외한 나머지 방법들의 장점이 계산이 매우 빠른 것이라고 했지만, 이는 상대적인 것이다. 위 4가지 방법 모두 계산 비용이 굉장히 많이 소모된다. 또 Forward Selection과 Backward Elimination의 결과를 고려했을 때, 둘의 결과가 상이할 수 있다. 즉, 이렇게 기계적으로 변수를 추가 혹은 제거하는 행위는 매우 위험하다.

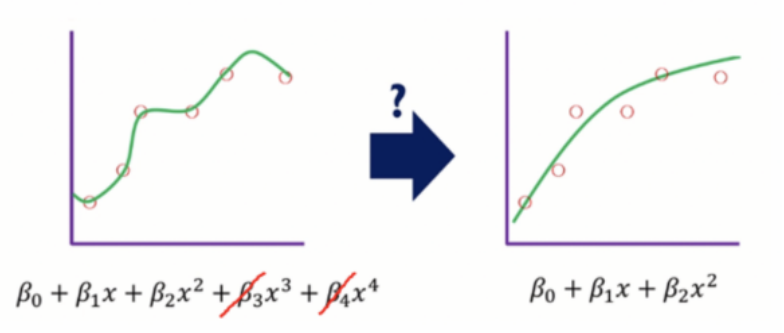
⇒ 그래서 다음에 배울 **정규화 방법**을 추천한다!

---

# 3. 정규화(Regularization)

## 0) 정규화란?

정규화란 모든 변수로 모델을 적합하되, 회귀 계수가 가질 수 있는 값에 제약조건을 부여함으로써 계수들을 작게 만들거나 0으로 만드는 방법이다.



$\beta$를 추정하기 위한 목적함수를 다음과 같이 계산한다고 해보자. 우리가 알던 것과 다르게 10000~이라는 (이상한) 항이 추가되었다. 이 때문에, $\beta_3$나 $\beta_4$가 조금만 증가하여도 식이 큰 폭으로 증가할 것이다.

$$\min_{\beta} \sum_{i=1} (y_i - \hat{y}_i)^2 + 10000\beta_3^2 + 10000\beta_4^2$$

만약 이렇게 $\beta_3$와 $\beta_4$에 큰 페널티를 부여한다면 이를 최소화하기 위해서 $\beta_3 \approx 0,\ \beta_4 \approx 0$가 되어야 할 것이고, 오른쪽의 그림으로 식이 바뀔 것이다.

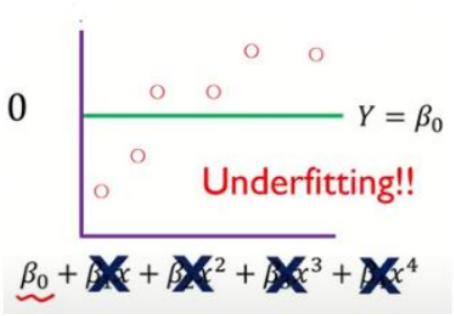
이런 식으로 회귀계수에 제약을 주는 것이 정규화의 기본 컨셉이라고 할 수 있다.

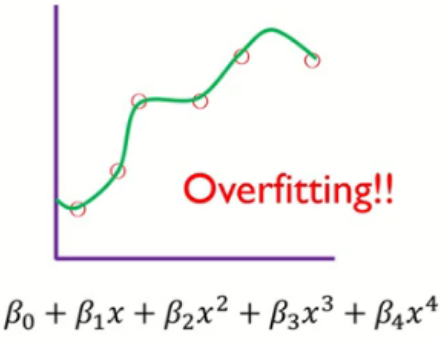
더 일반화된 수식으로 확장해보자.

$L(\beta) = L(\beta) = \min_{\beta} \sum_{i=1} (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$

앞의 $\min_{\beta} \sum_{i=1} (y_i - \hat{y}_i)^2$는 training accuracy에 해당하는 LSE를 의미한다. training data 관점에서 오차가 최소화되도록 한다고도 이해할 수 있으며, 이는 OLS추정량과 같다. 하지만 우리는 현재 데이터에 대한 해석 뿐 아니라 미래 데이터에 대한 예측도 원한다. 과적합을 방지하기 위해 의미가 없는 특정 계수가 주는 영향을 줄이고 싶다. 그래서 뒤의 $\lambda \sum_{j=1}^{p} \beta_j^2$ 항을 추가해주는 것이다. 이는 generalization accuracy라고도 하며, $\beta$에 제약을 주는 항이다.

여기에서 $\lambda$는 우리가 조절할 수 있는 hyper-parameter로, (1)과 (2) 사이의 trade-off를 조절하는 역할을 수행한다.




$\lambda$가 매우 크다면, $\beta_1 \approx 0, \beta_2 \approx 0, \beta_3 \approx 0, \beta_4 \approx 0 \rightarrow y = \beta_0$ (직선),

$\lambda$가 매우 작다면, $\beta$에 대한 제약이 거의 없는 것이다. (최소제곱법과 다를게 없음)

정규화 방법은 정말 다양한데, 대표적인 것들을 알아보자.

## 1) Ridge (L2 Regularization)

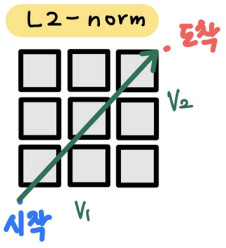
Ridge regression은 SSE를 최소화하면서 회귀계수 $\beta$에 제약 조건을 거는 방법이다. 이때 제약 조건식이 L2-norm 형태이기 때문에 L2 Regularization이라고도 불린다. (norm은 선대팀 2주차 클린업 참고)

✅ L2-norm

$$L_2 = \sqrt{|v_1|^2 + |v_2|^2 + \cdots + |v_n|^2}$$

유클리드 노름이라고 하며,

원점에서 벡터에 연결된 직선거리를 의미




Ridge regression에서 최소화하고자 하는 목적함수는 다음과 같다.

1. 목적함수

$$\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 \quad \text{subject to} \quad \sum_{j=1}^{p}\beta_j^2 \le s$$

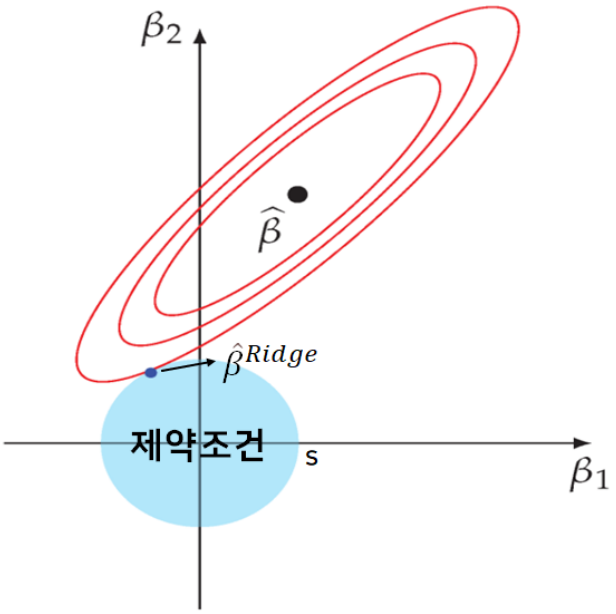
$$\Leftrightarrow \hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 + \lambda \sum_{j=1}^{p}\beta_j^2 \right\}$$

위의 식을 Ridge regression의 목적함수라고 하며, 우리는 위 식을 최소화 함으로써 회귀계수의 Ridge estimator를 얻을 수 있다. (두 식은 같은 form이다) 위 식은 미분이 가능하므로 미분을 통해서 추정량을 구할 수 있다. 단, 설명 변수들은 표준화 된 상태여야 한다.

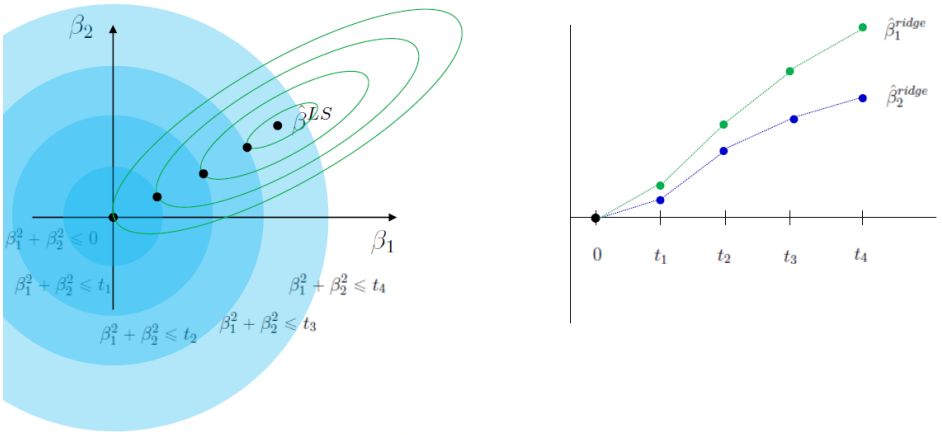
- 목적함수에 대한 이해 (1)

  $\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 \quad \text{subject to} \quad \sum_{j=1}^{p}\beta_j^2 \le s$

  왜 위 식이 회귀계수를 작게 만드는지 기하학적으로 이해해보자. 목적함수 중 위의 식을 아래와 같이 시각화 함으로써 직관적으로 이해할 수 있다. (타원모양으로 범위가 넓어짐)




빨간색 타원은 목적함수 SSE가 만드는 도형이며, 파란색 원은 제약 조건 $\sum_{j=1}^{p} \beta_j^2 \leq s$가 만드는 도형이다. 제약 조건을 반드시 만족시켜야 하므로 우리의 회귀계수는 반드시 원 내부에 존재해야 한다. 동시에 SSE를 최소화 해야 하므로 그래프 상에서 **타원과 원의 접점이 회귀계수의 Ridge estimator**가 된다.




tuning parameter에 따른 $\hat{\beta}^{ridge}$ 값의 변화 → t가 커지면 회귀계수에 대한 제약이 없는 것이나 마찬가지

$s$가 커질수록 원의 넓이는 커지고, 제약 조건이 완화된다. 이때의 추정량은 0에서 멀어진다. 반대로 $s$가 작아질수록 원의 넓이는 작아지고 제약 조건은 강화되어 추정량은 0에 가까워진다. 따라서 제약 조건이 추가됨으로써 회귀계수를 작게 만들 수 있는 것이다. (shrinkage)

- 목적함수에 대한 이해 (2)

  (1)의 식을 라그랑주 승수법을 이용해 아래와 같이 제약이 없는 최적화 문제로 변형 가능하다.

  → $\lambda$로 제약을 가하는 꼴

  $\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 + \lambda \sum_{j=1}^{p}\beta_j^2$

  이번엔 아래 식을 보자. 앞 term은 계속 봐왔던 오차제곱합(SSE)이며 뒤에 새로 붙은 것은 regularization term이다. 즉, 우리는 여전히 오차제곱합을 최소화 하고, 덧붙인 regularization term을 통해 개별 회귀계수가 너무 많이 커지는 것까지 조정해주고자 한다. 이 regularization term에 붙은 $\lambda$는 위에서 설명했듯이 음수가 아닌 tuning parameter로 최적의 모델을 찾는 과정에서 우리가 직접 CV(Cross Validation)를 통해 조정해 주어야하는 모수이다. $\lambda$는 앞서 말했던 (1)의 원의 크기를 결정하는 $s$처럼 제약 정도를 결정한다. 하지만 $s$와는 반대 관계이다.

  → $\lambda$가 커진다면 $\lambda$의 영향력이 커지게 되고, 위 식을 최소화하기 위해 $\sum_{j=1}^{p}\beta_j^2$은 작아져야 한다(trade-off 관계). 따라서 회귀계수들은 작아진다.

  $\lambda$가 무한대가 되면, 개별 회귀계수의 영향력은 무시될 만큼 작아져 회귀계수는 0에 근사한다. 반대로, $\lambda$가 작아지면 $\lambda$의 영향력이 작아지게 되고, 상대적으로 $\sum_{j=1}^{p}\beta_j^2$의 영향력이 커지게 되므로 회귀계수는 커진다. $\lambda$가 0이 된다면 regularization term이 사라지므로 기존 OLS 추정량을 도출한다.

2. 특징

- Scaling

  Ridge regression을 사용하기 위해서 개별 변수들을 scaling 해줘야 한다. 정규화 방법을 통해 우리는 각각의 회귀계수의 크기를 조정할 수 있는데, 회귀계수의 크기는 변수의 단위에 가장 크게 영향을 받는다. 무게를 나타내는 동일한 변수여도 단위가 Kg인지 g인지에 따라 회귀 계수는 1000배 차이가 날 것이다. 따라서 이러한 단위의 영향력을 제거하고 순수한 영향력만 나타내기 위해 scaling을 해준다. 주로 standard scaling을 사용한다.

- 계산 비용 절약

  L2 norm의 형태를 갖는 regularization term 덕분에 미분이 가능하다. $\lambda$값만 바꿔주면서 미분과 함께 행렬 연산을 하면 되기 때문에 계산 비용이 많이 절약된다.

- 예측 성능

  상관관계가 높은 변수들이 모델에 존재한다면, Ridge regression은 좋은 예측 성능을 보여준다.

- 변수 선택 불가

  Ridge regression이 다중공선성을 해결하는 정규화 방법인 건 맞지만, 다중공선성을 일으키는 변수를 제거하지는 못한다. $\lambda$값이 커짐에 따라 개별 회귀계수가 0에 가까워지기는 하지만, 0이 되지는 않기 때문에 영향력만 줄어들 뿐 모델에서 변수가 없어지는 것은 아니다.

그렇기 때문에 변수 제거는 할 수 없다. 따라서 Ridge regression을 통해 해석력을 증가시키는 것은 어렵다.

※ Ridge Regression 행렬로 이해하기

$$Q(\beta) = (y - X\beta)^T(y - X\beta) + \lambda\beta^T\beta$$
$$= y^Ty - 2\beta^TX^Ty + \beta^T(X^TX + \lambda I_p)\beta$$

→ $\frac{\partial}{\partial y}Q(\beta) = 2X^Ty + 2(X^TX + \lambda I_p)\beta = 0$

$\Rightarrow \hat{\beta}^{ridge} = (X^TX + \lambda I_p)^{-1}X^Ty$ vs $\hat{\beta}^{OLS} = (X^TX)^{-1}X^Ty$

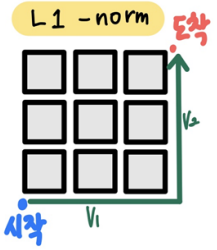
여기서 $I_p$는 $p \times p$크기의 Identity matrix이므로 각 $\lambda$만큼 더해주었다고 이해하면 된다! 다음과 같이 closed form이 존재하기 때문에 계산 비용이 줄어드는 것.

또 다중공선성이 갖는 문제점을 remind 해보자. 다중공선성의 문제점은 $X^TX$가 full rank가 되지 못해 역행렬이 존재하지 않거나, $det(X^TX)$가 매우 작아져 추정량의 분산이 매우 커진다는 것이었다.

그러나 ridge regression의 추정량은 $\lambda I_p$가 더해진 꼴이므로, matrix를 full rank로 만들거나, $det(X^TX)$를 크게 만들 수 있을 것이다. 그러므로 다중공선성 문제를 해결할 수 있는 것

## 2) Lasso (L1 Regularization)

Lasso regression도 마찬가지로 SSE를 최소화하면서 회귀계수 $\beta$에 제약 조건을 거는 방법이다.(Ridge와 아이디어는 동일함) 그러나 이때 제약 조건식이 L1-norm 형태이기 때문에 L1 Regularization이라고도 불린다. (선대팀 2주차 클린업 참고)




Lasso regression에서 최소화하고자 하는 목적함수는 다음과 같다.

1. 목적함수

$$\hat{\beta}^{Lasso} = \underset{\beta}{\text{argmin}} \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 \quad \text{subject to} \quad \sum_{j=1}^{p}|\beta_j| \leq s$$

$$\Leftrightarrow \hat{\beta}^{Lasso} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 + \lambda \sum_{j=1}^{p}|\beta_j|$$

위의 식이 Lasso regression의 목적함수이며, 우리는 위 식을 최소화 함으로써 회귀계수의 Lasso estimator를 얻을 수 있다. 하지만 미분이 불가능하기 때문에 수치적인 방법을 이용해 최적화 문제를 해결한다. 단, 설명 변수들은 마찬가지로 표준화된 상태여야 한다.

s와 $\lambda$ 역할은 같다. 하지만 미치는 영향을 반대 방향이다.

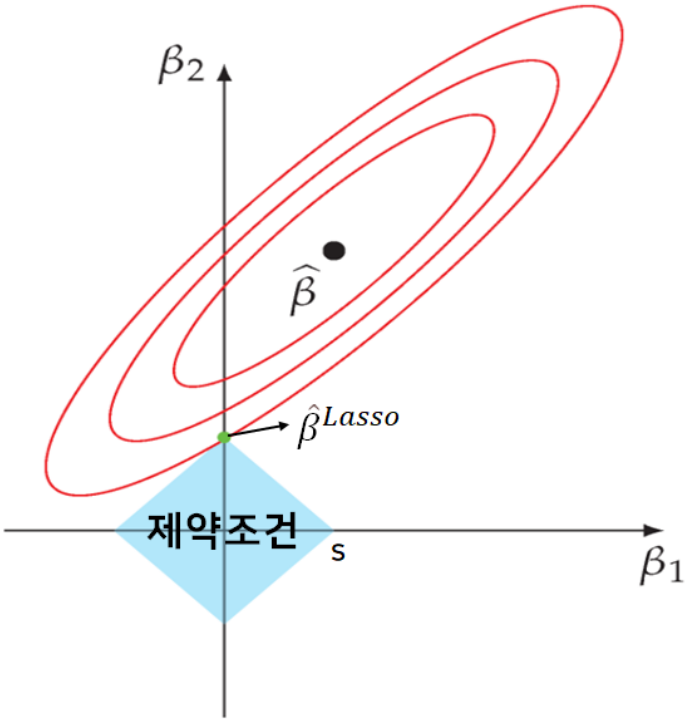
s가 작음 = $\lambda$가 큼 = 제약을 많이 가함

s가 큼 = $\lambda$가 작음= 제약을 조금 가함

- 목적함수에 대한 이해(1)

$$\hat{\beta}^{Lasso} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 \quad \text{subject to} \quad \sum_{j=1}^{p}|\beta_j| \le s$$

왜 위 식이 회귀계수를 작게 만드는지 이해해보자. 목적함수 중 위의 식을 아래와 같이 시각화 함으로써 직관적으로 이해할 수 있다.

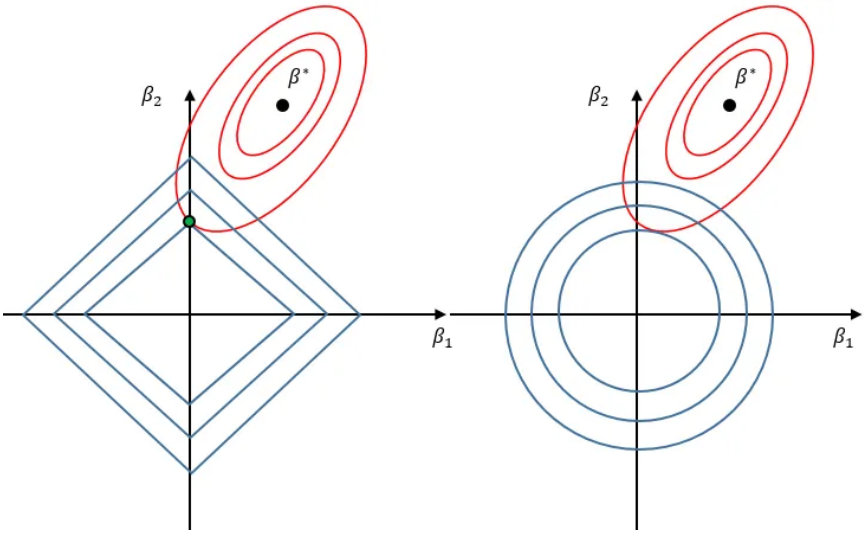



꼭짓점 부분에서 미분이 불가능하여 불연속점을 가진다는 특징

빨간색 타원은 목적함수 SSE가 만드는 도형이며, 파란색 마름모는 제약 조건 $\sum_{j=1}^{p}|\beta_j| \le s$ 이 만드는 도형이다. Ridge regression과 달리 제약조건의 형태가 마름모

꼴이다. 제약조건을 만족시키는 동시에 SSE를 최소화 해야 하므로 그래프 상에서 타원과 마름모의 접점이 회귀계수의 Lasso estimator가 된다. $s$가 커질수록 마름모의 넓이는 커지고, 제약 조건이 완화된다.

이때의 회귀계수 추정량은 0에서 멀어진다. 반대로 $s$가 작아질수록 마름모의 넓이는 작아지고 제약 조건은 강화되어 회귀계수 추정량은 0에 가까워지고, 결국 0이 된다.




Ridge regression과 그 원리가 매우 비슷하다. 하지만 다른 점이 있는데, 제약 조건의 형태 때문에 위 그림에서 볼 수 있듯이 일부 회귀계수(저기서는 $\hat{\beta}_1$)가 0이 되는 추정량이 도출될 수 있다. (항상 그런 것은 아님) 이처럼 Ridge regression과 달리 회귀계수를 정확히 0으로 만들 수 있는 것이 특징이다.

→ 만약 $\beta_2 = 0$이라고 나온다면 $y$를 예측하는데 있어 $x_2$변수가 유의하지 않다는 것이기 때문에, 이를 variable selection 과정에서 활용할 수도 있음!

- 목적함수에 대한 이해 (2)

(1)의 식을 라그랑주 승수법을 이용해 아래와 같이 제약이 없는 최적화 문제로 변형 가능하다.

$$\hat{\beta}^{Lasso} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 + \lambda \sum_{j=1}^{p}|\beta_j|$$

Ridge regression과 동일하다. 앞의 term은 계속 봐왔던 오차제곱합(SSE)이며 뒤에 새로 붙은 것은 regularization term이다. 즉, 우리는 여전히 오차제곱합을 최소화 하고, 덧붙인 regularization term을 통해 개별 회귀계수가 너무 많이 커지는 것을 조정해준다. $\lambda$는 CV를 통해 최적값을 찾는다.

→ $\lambda$가 커진다면 $\lambda$의 영향력이 커지게 되고, 위 식을 최소화하기 위해 $\sum_{j=1}^{p}|\beta_j|$은 작아져야 한다(trade-off 관계). 따라서 회귀계수들은 작아진다. $\lambda$가 무한대가 되면, 개별 회귀계수의 영향력은 무시될 만큼 작아져 회귀계수는 0에 근사한다. $\lambda$가 작아지면 $\lambda$의

영향력이 작아지게 되고, 상대적으로 $\sum_{j=1}^{p}|\beta_j|$의 영향력이 커지게 되므로 회귀계수는 커진다. $\lambda$가 0이 된다면 regularization term이 사라지므로 기존 OLS 추정량을 도출한다.

| 큰 $\lambda$값 | 작은 $\lambda$값 |
|---|---|
| 적은 변수(계수가 0이 되므로) | 많은 변수 |
| 간단한 모델 | 복잡한 모델 |
| 해석 쉬움 | 해석 어려움 |
| 높은 학습 오차 (underfitting 위험 증가) | 낮은 학습오차 (overfitting 위험 증가) |

2. 특징

- Scaling

  Lasso regression을 사용하기 위해서 Ridge regression과 동일하게 개별 변수들을 scaling 해줘야 한다. 변수의 단위에 의한 영향력을 제거하고 순수한 영향력만 나타내기 위해 scaling을 해준다. 주로 standard scaling을 사용한다.

- 변수 선택

  ridge와 달리 $\lambda$값에 따라 0이 되는 회귀 계수가 존재하기 때문에 변수 선택이 가능해진다. 변수 선택이 가능해짐으로써 변수들의 해석 가능성도 증가한다. 하지만 변수 간 상관관계가 높다면 변수 선택 성능이 떨어진다. 0이 되는 계수의 존재로 인해 sparsity(희박성)를 가진다고 말한다.

  LASSO는 Least Absolute Shrinkage and "Selection" Operator의 약자기도 하다.

- 예측 성능

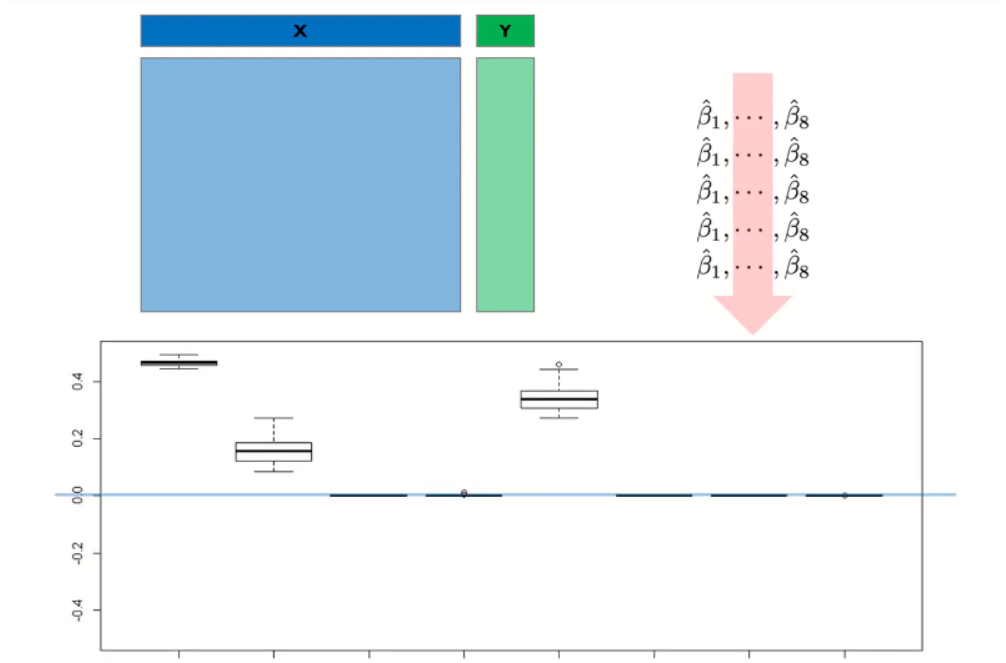
  변수들 간 상관관계가 큰 경우, lasso regression은 예측에 유의미한 변수들을 0으로 만들 수 있기 때문에 ridge에 비해 상대적으로 예측 성능이 떨어진다.

- 미분 불가능한 점이 있기 때문에 closed form solution을 구할 수 없다.

  → 그래서 수치최적화방법(numerical optimization methods)를 사용한다.

- 꽤 robust한 모델이다.

  데이터가 달라질 때마다 변수 선택의 결과도 달라진다면, 그것은 좋은 모델이라고 하기는 어려울 것이다. 간단한 실험 결과를 살펴보자. 데이터의 위부터 일부분만을 사용하며, 베타 값을 추정한 결과를 박스 플롯으로 그린 것이다.



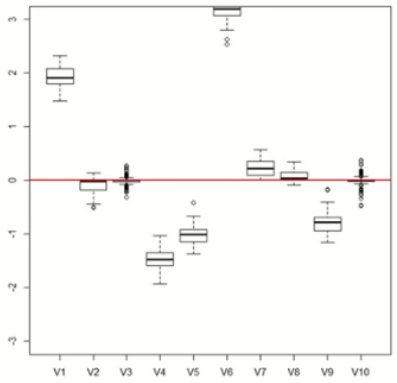

box plot을 통해 변화 폭이 크지 않음을 확인할 수 있고, 0이 된 변수들은 계속 0이 되는 것을 확인할 수 있다. ⇒ 데이터가 바뀌더라도 결과가 강건하다

⇒ Ridge와 Lasso 정리

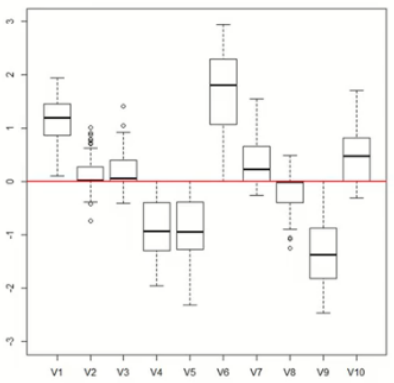
| Ridge regression (L2-norm) | Lasso regression (L1-norm) |
|---|---|
| 변수 선택 불가능 | 변수 선택 가능 |
| Closed form solution 존재 (미분을 통해) | Closed form solution 존재 X (numerical optimization 이용) |
| 상관관계가 높은 상황에서 좋은 예측 성능 (상관성이 있는 변수들에 대해서 적절한 가중치를 배분한다) | 변수 간 상관관계가 높은 상황에서 Ridge에 비해 상대적으로 예측 성능이 떨어짐 |
| 제약 범위가 원이다 | 제약 범위가 마름모이다 (최적값이 모서리 부분에서 나타날 확률이 높아 몇몇 유의하지 않은 변수들에 대해 계수를 0으로 추정해 주어 변수 선택 효과가 있다 → 보다 엄격함) |
| 크기가 큰 변수를 우선적으로 줄이는 경향이 있음 | |

참고

변수 간 상관관계가 낮을 경우




변수 간 상관관계가 높을 경우




하지만 아까 robust하다고 했던 Lasso도 변수 간 상관관계가 높다면 robust하지 않음

### 3) Elastic-Net

Elastic Net = Ridge+Lasso의 regularization term을 혼합한 형태이다. 변수 간 상관관계가 존재할 때 Lasso의 성능이 떨어지는 한계를 보완하기 위해 고안된 방법이다. Lasso의 경우 상관관계가 있는 다수의 변수들 중 하나를 선택해서 계수를 줄인다. 하지만 Elastic Net은 상관성이 있는 변수들을 모두 선택하거나 모두 제거(동시에 선택하거나 제거)함으로써 성능을 보완한다. 이를 grouping effect라고 한다.

- Grouping effect 설명

  증명에 대해 살펴보고 싶다면 : https://hastie.su.domains/Papers/B67.2 (2005) 301-320 Zou & Hastie.pdf *by Zou and Hastie(2005) 'Regularization and variable selection via the elastic net'*

  $|\hat{\beta}_i^{enet} - \hat{\beta}_j^{enet}| \leq \frac{\sum_{i=1}^{n} |y_i|}{\lambda_2} \sqrt{2(1 - p_{ij})}$

  (여기서 $p_{ij}$는 $x_i$와 $x_j$의 상관계수를 의미)

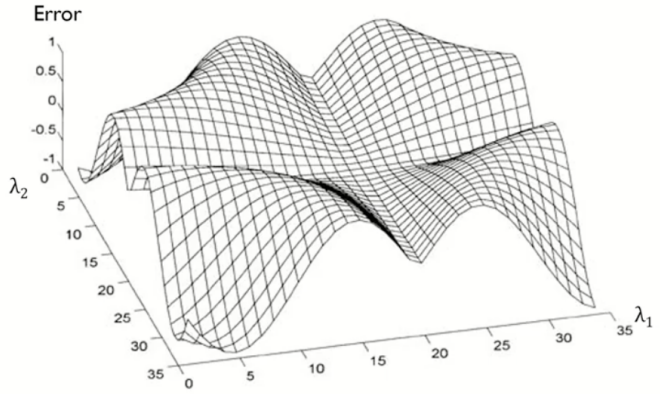
  $p_{ij} = 1 \rightarrow |\hat{\beta}_i^{enet}| = |\hat{\beta}_j^{enet}|$ : 만약 하나라도 중요하다면 둘 다 똑같이 중요하다는 의미

  (상관계수가 높은 것은 같이 선택하거나 제거함 → 같은 pattern을 보이는 것끼리 grouping)

  $\Rightarrow$ $p_{ij}$가 증가하거나 $\lambda_2$가 증가한다면, $|\hat{\beta}_i^{enet} - \hat{\beta}_j^{enet}|$는 감소한다!

- Elastic Net의 Parameter

Grid Search 방법을 사용: $\lambda_1$과 $\lambda_2$의 범위를 정해서 error가 최소화되는 조합을 찾아가는 방법이다.

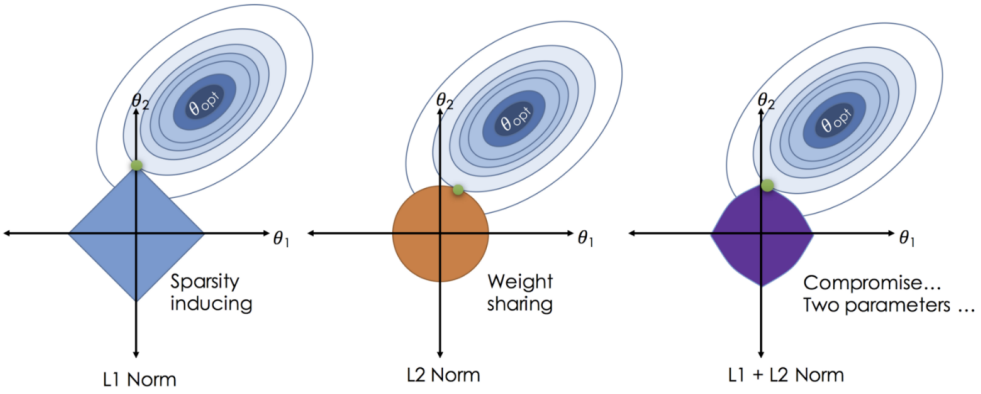



$\lambda_1, \lambda_2$ 값 구하는 방법(모델 성능 향상을 위한 hyper parameter 찾기)

- 목적함수

$$\hat{\beta}^{elastic}$$
$$= \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 subject\ to\ t_1 \sum_{j=1}^{p}|\beta_j| +$$
$$t_1 \sum_{j=1}^{p}\beta_j^2 \le s$$
$$\Leftrightarrow \hat{\beta}^{elastic} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 + \lambda_1 \sum_{j=1}^{p}|\beta_j| +$$
$$\lambda_2 \sum_{j=1}^{p}\beta_j^2$$

위 식을 보면 Ridge의 L2 term과 Lasso의 L1 term이 같이 목적함수에 들어있는 것을 알 수 있다.

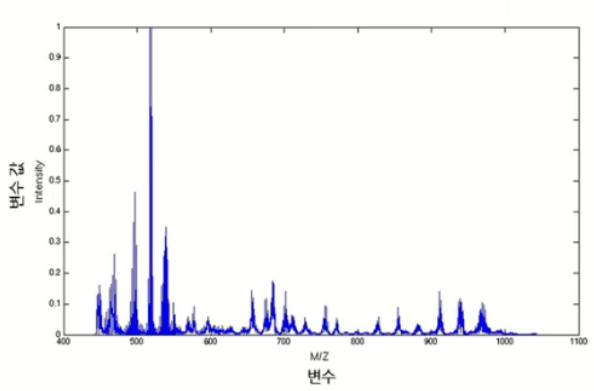


왼쪽부터 Lasso, Ridge, Elastic Net이다. 제약 조건이 변함에 따라 추정량이 만들어지는 공간도 변화하고 있음을 알 수 있다. (맨 오른쪽) 원도 아니고, 마름모도 아닌 그 중간의 형태이다.

## 4) Fused Lasso

Elastic Net은 상관관계가 높은 변수들이 존재할 때 Lasso 방법의 단점을 보완하고자 만들어진 모델이다. 이는 달리 생각했을 때, 변수들 간에 상관관계가 존재한다는 사전 지식을 활용하는 회귀 모델이기도 하다.

지금부터 소개할 Fused Lasso 역시 변수들 사이의 인접성이 있을 때, 이런 사전 지식을 활용하는 모델이다.



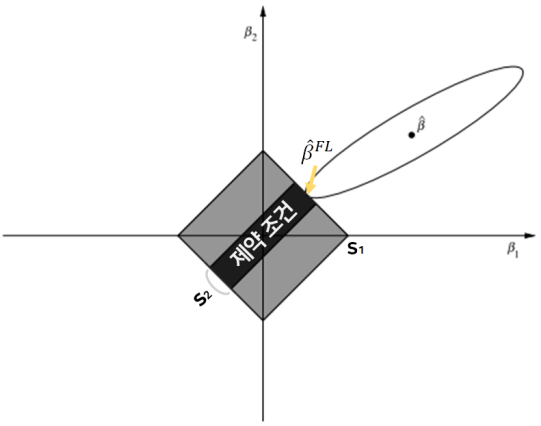

그래프에서 볼 수 있듯이 몇몇 데이터(Signal, Spectra 등)들은 중요한 변수들이 peak를 중심으로 연속적으로 나타난다. 이렇게 인접한 변수들끼리 비슷한 값을 가짐을 동시에 고려하기 위해 Fused Lasso가 마련되었다.

- 목적함수

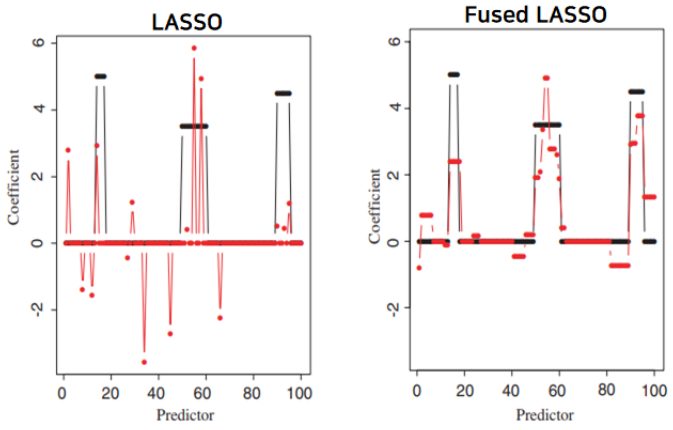
제약식 부분을 보면 첫 번째 부분은 Lasso와 같은 term이고, 두 번째 term이 새로 추가되었다. 이는 상관관계와 관계 없이 물리적으로 인접한 변수들의 회귀계수를 비슷한 값으로 추정하게 만드는 역할을 한다. (양 옆에 있는 변수들의 회귀계수 값을 최소화하는 것! → smoothness)

$$\hat{\beta}^{FL} = \underset{\beta}{\operatorname{argmin}}(\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_i)^2 + \lambda_1 \sum_{j=1}^{p}|\beta_j| + \lambda_2 \sum_{j=1}^{p}|\beta_j - \beta_{j-1}|)$$

제약식을 시각화하면 아래와 같다. 기존 제약 공간 내에 더 strict한 제약 공간이 추가되었음을 알 수 있다.




변수 선택의 측면에서도 Lasso와 Fused Lasso를 비교해볼 수 있는데, 왼쪽의 기존 Lasso 같은 경우는 인접한 실제 계수들(검정색)을 제대로 추정해내지 못하고 있다(빨간색). 반면, 오른쪽의 Fused Lasso는 서로 인접한 변수들의 계수가 비슷하게 추정되는 것을 알 수 있다.




# 4. 공간회귀분석

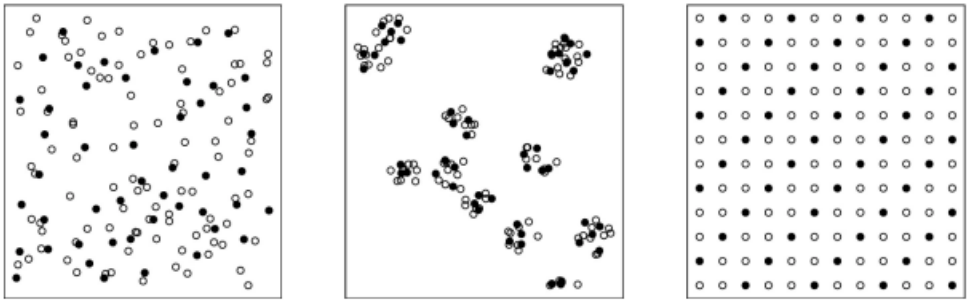
이번 주에 배운 변수 선택 및 정규화와는 다른 결이지만, 공간회귀에 대해 알아보도록 하자. 2주차 회귀분석의 가정에서, 오차의 독립성 위배는 시간적/공간적 자기상관의 존재를 의미한다고 볼 수 있었다. 공간회귀는 이러한 공간적 자기상관을 해결하기 위한 모델이라고 할 수 있다. (기억하시죠?ㅎㅎ)

## 1) 공간 데이터

공간상의 특정 위치 또는 특정 영역의(즉, 좌표와 관련된) 속성의 집합

- **Spatial Randomness**

  공간 통계는 '공간상에서 어떠한 사건의 발생할 확률이 전부 같은 확률분포를 가지고 있기 때문에, 아무런 패턴이 존재하지 않는다' 는 spatial randomness를 기초로 한다.



1. Complete Spatial Random (CSR) pattern : 점의 위치가 랜덤하게 분포
2. Cluster pattern : 점이 클러스터를 형성하며 분포
3. Regular grid pattern : 점이 규칙적으로 분포

→ 위 사진의 경우 두 번째와 세 번째 분포에서 특정한 패턴이 존재한다고 볼 수 있다.

- **공간 패턴(Spatial Pattern) 분석**

  특정한 현상이 공간 상에 분산 또는 집중되었는가를 파악하고, 이러한 공간 패턴을 형성하는데 영향을 미친 공간 과정을 밝히는 것 (분포에 영향을 미친 요인을 파악!)

## 2) 공간자기상관(Spatial Autocorrelation)

- **Tobler 지리학 제 1 법칙**

> **Everything is related to everything else, but near things are more related than distant things**

이를 요약하면 가까이 있을 수록 유사성을 띄는 것을 말하는데, 이를 **공간자기상관**이라고 한다.

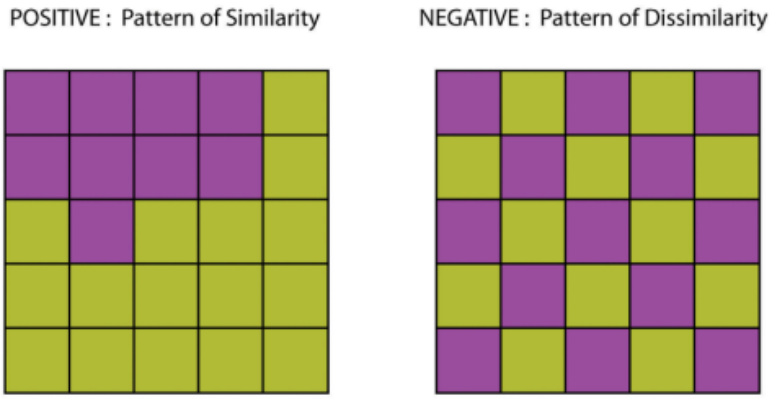
ex) 아파트 가격 : 종로 - 성북 vs 종로 - 강동

- 공간적 의존성과 공간적 이질성

1. **공간적 의존성(Spatial Dependence)** : 앞서 언급한 자기상관의 개념으로, **인접지역의 Y가 다른 Y에게 영향을 주는 것**

   - 특정 지역의 사건 강도가 인접지역의 사건에 영향을 주는가?

ex) 종로구의 지가 상승이 성북구의 지가를 상승시키는가?




(1) 강한 양의 Spatial Autocorrelation : 근처의 관측치들과 매우 유사한 형태

(2) 강한 음의 Spatial Autocorrelation : 근처의 관측치들과 매우 상반된 형태

이 자기상관을 공간의 크기에 따라 다음과 같이 구분해 볼 수 있다.

- **전역적(Global) 공간 자기상관** : 전체 구역이 가지는 하나의 공간자기상관
- **국지적(Local) 공간 자기상관** : 특정 지점이 공간자기상관

2. **공간적 이질성(Spatial Heterogeneity)** : 넓은 지역에서 나타나는 불규칙한 분포를 의미하며, 한 지역내에 서로 성격이 다른 하위 집단이 존재함

   **즉, 변수의 값이 공간 단위 사이에 불균등하게 분포되어 있는 것**

   ex) 지하철 개통이 지가에 미치는 영향력의 크기가 도시와 농촌에서 같은가?

## 3) 공간자기상관 진단

먼저 어떠한 공간 패턴이 우연적인 것인지, 또는 어떤 체계를 따라 비슷한 것끼리 몰려 있는 것인지 파악해야 한다. 이를 위해 공간 패턴에 대해 알고자 하는 지역들이 공간적으로 인접한지 부터 확인해야 한다. 이를 위해 인접 여부를 수치화 할 필요가 있다.

- 공간가중행렬(Spatial Weights Matrix)

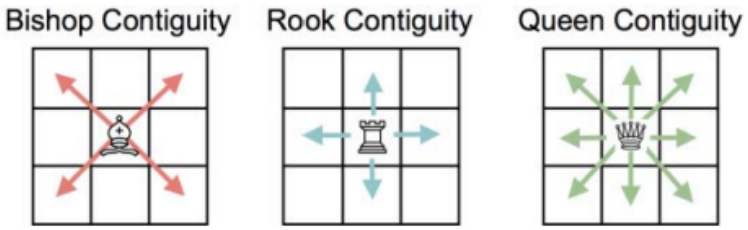
  연구대상지역 내 **다수의 지점들이 서로 공간적으로 인접하고 있는가**의 여부를 파악할 수 있도록 행렬로 나타낸 것으로**, 지역 간 잠재적 상호작용의 강도**를 말해준다. i관측치

와 j관측치의 거리의 역수 등으로도 구성할 수 있지만, 우선 간단하게 이웃하면 1이고 아니면 0의 값을 가지는 행렬만을 생각해보자.

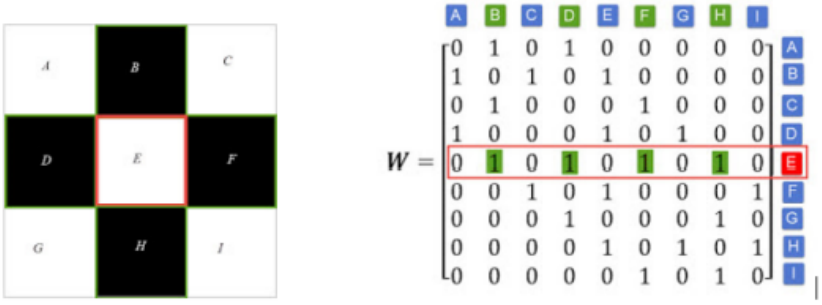
$$w_{ij} = \begin{cases} 1 & if\ i,j\ is\ neighbor \\ 0 & otherwise \end{cases}$$

그렇다면 이웃을 결정하는 기준은 어떻게 될까? 대표적인 3가지 경우를 살펴보자.

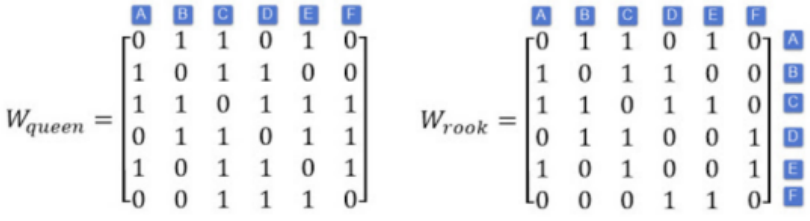
1. **Binary Contiguity Weights** : **근접하고 있는 경우**를 이웃으로 보는 방식




- **Bishop Contiguity** : 각 모서리에 있는 영역을 이웃으로 간주
- **Rook Contiguity** : 각 면에 접하는 영역을 이웃으로 간주 → **가장 보편적으로 사용**
- **Bishop Contiguity** : 모든 면과 모서리가 접하는 접하는 영역을 이웃으로 간주




Rook Contiguity를 사용하여 만든 공간가중행렬




Queen Contiguity와 Bishop Contiguity 를 사용하여 만든 공간가중행렬

2. **Distance-based Weights** : 특정 거리보다 가까우면 이웃, 멀면 이웃이 아니라고 정의

$$w_{ij} = \begin{cases} 1 & if\ d_{ij} < d \\ 0 & otherwise \end{cases}, where\ d\ =\ minimum\ distance$$

최소거리가 너무 작게 설정되면 특정점에서 다른 점 까지의 모든 거리보다도 작아지고, 이웃이 없는 고립된 점이 생길 수 있다. 따라서 최소거리를 정할 때 각 관측치별 최단거리보다는 크게 정해야 한다.

→ 분석의 결과가 판별기준(최소거리)에 민감하기 때문에 분석 시 유의해야 한다.

3. **K-nearest Neighbors Weights** : 머신러닝에서 KNN 알고리즘과 비슷한 메커니즘으로, 가장 근접한 k개의 점을 이웃으로 정의하는 법

**- 공간가중행렬의 정규화**

공간 가중행렬은 그대로 쓰이지 않고, 정규화하여 사용한다.

1. Row Standardized Weights : 행 단위로 정규화하는 방법

$$w_{ij}^{*} = \frac{w_{ij}}{\sum_{all\ j} w_{ij}}$$

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \Rightarrow \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix}$$

2. Stochastic Weights : 전체 행렬을 정규화하는 방법

$$w_{ij}^{*} = \frac{w_{ij}}{\sum_{all\ i,j} w_{ij}}$$

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \Rightarrow \begin{pmatrix} 1/7 & 1/7 & 1/7 \\ 1/7 & 0 & 1/7 \\ 1/7 & 1/7 & 0 \end{pmatrix}$$

이렇게 공간 가중행렬을 구축한 후, 공간상에서 나타나고 있는 특정한 현상이 공간적 자기상관성을 갖고 있는가에 대해 가설을 수립하고 통계적 검정을 실행한다.

- 통계적 검정방법

1. **Moran's I 지수**

   Moran's I는 값 분포가 공간 무작위 상태(CSR)와 얼마나 다른지 파악하는 수치로 사용된다. 지역 간의 인접성을 나타내는 **공간가중행렬과 인접하는 지역들 간의 속성 데이터의 유사성**을 측정하는 방법.

   식이 복잡하니 설명으로 이해해보자!

   한 지역에 여러 구역이 있고, 구역마다 통계수치(발생 건수, 평균 농도 등)가 있다고 해보자. Moran's I값은 **평균을 기준**으로 생각한다. 모든 구역의 값을 평균내고, 인접한 두 구역$(i, j)$의 값에서 평균을 뺀다. 두 구역 값이 평균보다 크다면 둘 다 양수일 것이고 평균보다 작다면 둘 다 음수일 것이다. 둘 모두 양수거나 음수라면 곱했을 때 양수가 될 것이고, 두 관측값이 전체 평균에서 같이 멀수록 두 값에서 평균을 뺀 값을 곱한 수는 커질 것이다.

$$I = \frac{N\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}(Y_i - \bar{Y})(Y_j - \bar{Y})}{(\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}\sum_{i=1}^{N}(Y_i - \bar{Y})^2}$$

$$Z_I = \frac{I - E(I)}{\sqrt{Var(I)}} \quad where\ E(I) = -\frac{1}{N-1}$$

- $N$ : 지역 단위 수 / $Y_i$ : $i$ 지역의 속성 / $Y_j$ : $j$지역의 속성 / $\bar{Y}$ : 평균값 / $w_{ij}$ : 가중치(공간가중행렬의 원소)
- $I$값의 범위는 $-1 \sim 1$, 1은 완전한 양의 자기상관, -1은 완전한 음의 자기상관을 의미함
- $Z_I$ 값이 통계량이 되며, Z 검정을 통해서 **전역 공간패턴**의 통계적 유의성 판단
- 한계

  **전역 자기상관성만을 검정**하기 때문에, 전체공간에서 패턴의 존재유무만 알 수 있을 뿐, 핫스팟이나 콜드스팟의 위치는 알 수 없다.

  ※ 핫스팟과 콜드스팟

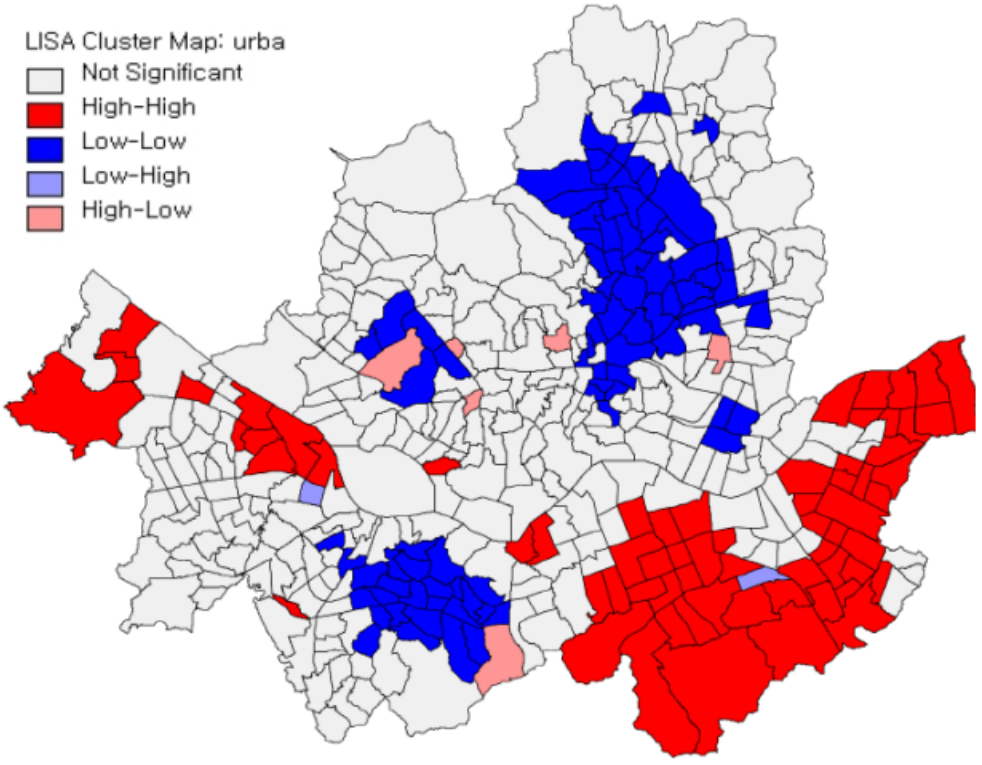
  특정 값이 높거나 낮은 지역이 군집되어 있는 것

2. **LISA(Local Indicator of Spatial Association) 지표**

특정 지역들이 전체 지역의 공간자기상관성에 얼마나 영향을 미치는지 파악하는 방법. 또한 핫스팟과 콜드스팟의 위치를 알 수 없다는 Moran's I 지표의 한계를 극복할 수 있다.

전역적 자기상관성에서 공간적 자기상관이 있다고 나타나면, 그것이 **세부적으로 어느 지역에서 나타나는 것인지 알기 위해 사용**한다.

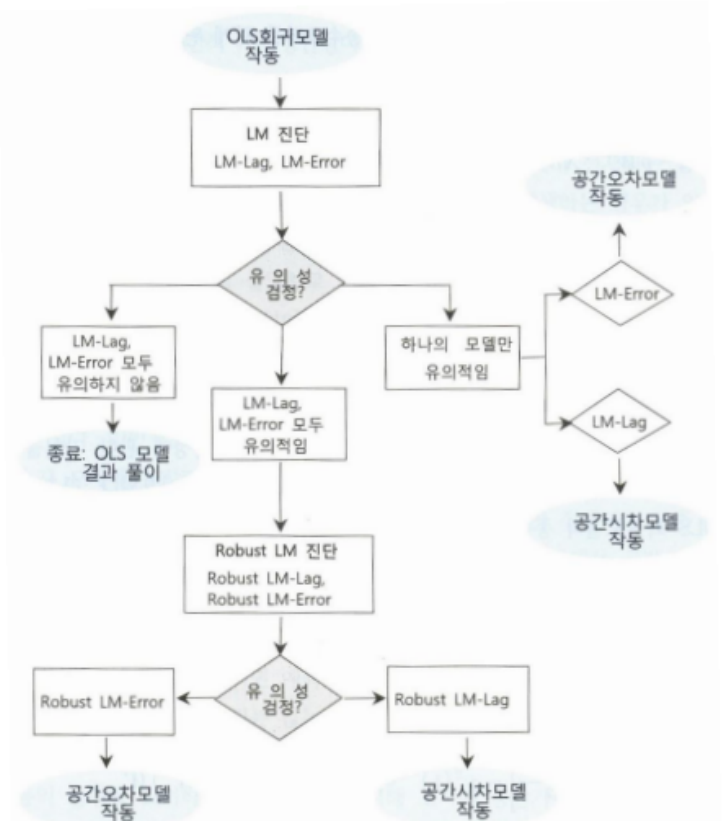
4가지 유형의 공간적 연관성 분포를 모란 산포도를 통해 나타내는데, 이를 매핑하여 공간적 클러스터 패턴이 어떻게 나타나는가를 분석할 수 있다.




{HH(High-High) 유형, LL(Low-Low) 유형 : 공간적 군집지역
HL유형,LH유형 : 공간적 이례지역(Spatial Outlier)

### 3. 라그랑지 승수검정(LM : Lagrange Multiplier)

OLS 회귀모델의 종속변수 또는 오차에서 **공간자기상관이 실재하지 않는다**는 귀무가설에 대해 검정하는 것
공간회귀모델에서 공간자기상관은 종속변수 또는 오차에서 나타날 수 있다. 다만, 어디에 공간자기상관이 존재하는지에 따라 사용해야 하는 공간회귀모델이 달라지기 때문에 그 둘을 구별할 필요가 있다.




i) LM이 유의하지 않을 경우, 귀무가설 기각하지 않음 → **OLS 회귀모델** 사용

ii) LM-Lag 값이 유의한 경우 → **공간시차모델** 사용

iii) LM-Error 값이 유의한 경우 → **공간오차모델** 사용

iv) LM-Lag, LM-Error 값 모두 유의한 경우 → Robust LM 진단

iv-i) Robust-LM Lag 유의한 경우 → **공간시차모델** 사용

iv-i) Robust-LM Error 유의한 경우 → **공간오차모델** 사용

## 4) 공간자기상관 처방

앞서 말했듯이 공간데이터의 문제 유형은 두 가지로 나눌 수 있는데, 이 유형에 따라 해결 방법도 달라진다.

**1) 공간적 의존성 : 인접지역의 영향력을 변수에 포함시켜 통제**

- 공간시차모형 (SLM, Spatial Lag Model)
- 공간오차모형 (SEM, Spatial Error Model)

**2) 공간적 이질성 : 각 지역별 영향력(추정계수)를 추정**

- 지리가중회귀모형 (GWR, Geographically Weighted Regression)

- **공간시차모형(SLM, Spatial Lag Model)**

  한 지역의 관측치가 인접지역들의 관측치와 상관성이 있는 경우, 통계모델에 공간적 의존성을 변수로 투입시켜야 한다.
  이 때, **공간시차변수**를 하나의 설명변수로 회귀모델에 삽입한다.

$$\begin{aligned} Y &= \rho WY + X\beta + \varepsilon \\ &= (I-\rho W)^{-1}(X\beta+\varepsilon), \ \ \varepsilon \sim MVN(0,\sigma^2 I_n) \end{aligned}$$

  - **예시**

    주택가격 = 주택면적 + 건축년도 + 가구주의 소득 + CBD로부터의 거리 + 오차

    → 공간시차변수 추가 (주택가격에 공간가중행렬 $W$를 곱한 것)

    **주택가격 = $W$*주택가격 + 주택면적 + 건축년도 + 가구주의 소득 + CBD로부터의 거리 + 오차**

- **공간오차모형(SEM, Spatial Error Model)**

  오차가 공간자기상관성을 갖고 있다면, 이는 주로 숨겨진 설명변수를 고려하지 못하여 오차가 공간자기상관성을 갖고 있는 변수로 나타나는 것이다. 이 때, 오차를 공간오차변수로 변형시켜준다.

$$\begin{aligned} Y &= X\beta + \mu \\ &= X\beta + (I-\lambda W)^{-1}\varepsilon, \ \ where\ \mu = \lambda W\mu + \varepsilon\ and\ \varepsilon \sim MVN(0,\sigma^2 I_n) \end{aligned}$$

  - **예시**

    주택가격 = 주택면적 + 건축년도 + 가구주의 소득 + CBD로부터의 거리 + 오차

    → 오차를 공간오차변수로 변형

    **주택가격 = 주택면적 + 건축년도 + 가구주의 소득 + CBD로부터의 거리 + (공간오차)**

✅ 공간더미변수와의 차이

공간 데이터를 다룰 때 지역별로 더미변수를 인코딩하여 사용할 수도 있다. 이를 통해 특정 공간이 종속변수에 미치는 영향만을 분석에 포함시킬 수 있지만, 공간자기상관모형은 **인접 지역의 종속변수, 설명변수, 혹은 오차가 그 지역의 종속변수에 미치는 영향**을 다루는 모형으로 성격이 다르다.

- **지리가중회귀모델 (GWR, Geographically Weighted Regression)**

  국지적 차원에서, 변수들 간의 관계를 추정하는 회귀계수가 지역 간에 서로 다르다는 것을 전제하여 **지역별 회귀모델을 추정**하는 방법으로, 공간적 이질성으로 발생하는 이분산성을 해결할 수 있다.

$$W_i^{1/2} = W_i X \beta_i + W_i^{1/2} \varepsilon_i$$
$$\beta(u_i, v_i) = [X' W(u_i, v_i) X]^{-1} X' W(u_i, v_i) Y$$

  GWR의 경우 회귀분석이 분석 단위 별로 이루어져 격자별로 회귀계수 값이 모두 달라지므로, **추정된 계수 값은 해당 격자에서만 의미가 있다.**

  각 위치좌표 $(u_i, v_i)$ 별로 하나의 $W$값을 가지며, $W$값은 인근 관측치들로부터 도출한다.

  지리가중회귀모델을 사용했을 때, **같은 변수에 대한 지역별 회귀계수의 차이가 크다면 공간적 이질성이 존재한다**고 볼 수 있다. 또한, 지리가중회귀모델 사용 후에는 반드시 F검정을 이용해 대안모형(GWR)이 기준모형(OLS)을 개선하였는지 검정해야 한다!

---

# Appendix(진짜 마지막 한 숟갈..)

### 1. MSE의 Contour plot이 타원의 형태인 이유

MSE의 식을 정리해보자.

$$
\begin{aligned}
MSE(\beta_1, \beta_2) &= \sum_{i=1}^{n} (y_i - \beta_1 x_{i1} - \beta_2 x_{i2})^2 \\
&= \sum_{i=1}^{n} {y_i}^2 - 2\sum_{i=1}^{n} y_i(\beta_1 x_{i1} + \beta_2 x_{i2}) + \sum_{i=1}^{n} (\beta_1 x_{i1} + \beta_2 x_{i2})^2 \\
&= \sum_{i=1}^{n} {y_i}^2 - 2\left(\sum_{i=1}^{n} y_i x_{i1}\right)\beta_1 - 2\left(\sum_{i=1}^{n} y_i x_{i2}\right)\beta_2 + \sum_{i=1}^{n} (\beta_1^2 x_{i1}^2 + \beta_2^2 x_{i2}^2 + 2\beta_1\beta_2 x_{i1} x_{i2}) \\
&= \left(\sum_{i=1}^{n} x_{i1}^2\right)\beta_1^2 + \left(\sum_{i=1}^{n} x_{i2}^2\right)\beta_2^2 + \left(2\sum_{i=1}^{n} x_{i1} x_{i2}\right)\beta_1\beta_2 \\
&\qquad -2\left(\sum_{i=1}^{n} y_i x_{i1}\right)\beta_1 - 2\left(\sum_{i=1}^{n} y_i x_{i2}\right)\beta_2 + \sum_{i=1}^{n} {y_i}^2
\end{aligned}
$$

$= A\beta_1^2 + B\beta_1\beta_2 + C\beta_2^2 + D\beta_1 + E\beta_2 + F$

✅ 판별식을 통해 contour를 파악할 수 있다.

$B^2 - 4AC = 0$ (포물선)

$> 0$ (쌍곡선)

$< 0$ (타원)

$if\ B = 0\ and\ A = C$ (원)

그래서 MSE값을 판별식에 적용해보면 코시-슈바르츠 부등식에 의해 다음과 같은 결과를 얻을 수 있다.

$$
MSE(\beta_1, \beta_2) = \left(\sum_{i=1}^{n} x_{i1}^2\right)\beta_1^2 + \left(\sum_{i=1}^{n} x_{i2}^2\right)\beta_2^2 + \left(2\sum_{i=1}^{n} x_{i1} x_{i2}\right)\beta_1\beta_2 - 2\left(\sum_{i=1}^{n} y_i x_{i1}\right)\beta_1 - 2\left(\sum_{i=1}^{n} y_i x_{i2}\right)\beta_2 + \sum_{i=1}^{n} {y_i}^2
$$

$$
\begin{aligned}
B^2 - 4AC &= \left(2\sum_{i=1}^{n} x_{i1} x_{i2}\right)^2 - 4\sum_{i=1}^{n} x_{i1}^2 \sum_{i=1}^{n} x_{i2}^2 \\
&= 4\left\{\left(\sum_{i=1}^{n} x_{i1} x_{i2}\right)^2 - \sum_{i=1}^{n} x_{i1}^2 \sum_{i=1}^{n} x_{i2}^2\right\} < 0
\end{aligned}
\qquad \text{By Cauchy-Schwartz inequality}
$$

→ 즉, MSE의 contour가 **타원의 형태**가 됨을 수식적으로 이해할 수 있다.

## 2. Adaptive Lasso

Lasso는 $\lambda$가 커지며 회귀계수의 크기에 대해 강한 페널티를 주는데, 변수별로 0으로 shrink 하는 속도가 달라 변수선택이 가능한 모형이었다. 이를 위해 설명변수를 표준화하여 척도의 영향을 줄인다. 그러나 표준화 후에도 특정 변수의 회귀계수가 크다면, Lasso의 목적함수는 그 변수의 크기를 줄이는데에 집중하게 된다. 이런 경우에 활용될 수 있는 모형이 Adaptive Lasso 이다.

Adaptive Lasso는 Lasso가 발전한 형태로 모든 변수의 절대값에 대해 동일하게 제약을 가하는 $\lambda$와 함께, 각 회귀계수마다 다른 값으로 제약조건을 주는 $\hat{\omega_j}$도 포함된 회귀모형이다.

기존의 Lasso에서는 모든 $\beta_j$에 동일한 제약을 주기 때문에, $\beta_j$자체가 크다면 그 회귀계수를 줄이는 데에 최적화 과정이 overfitting 될 위험이 있다. Adaptive Lasso에서는 큰 $\beta_j$ 에게 작은 $\hat{\omega_j}$를 줌으로써 큰 회귀계수를 최소화하는데 overfitting하던 문제를 해결한다.

Adaptive Lasso의 목적함수를 살펴보면 다음과 같다.

$$\sum_{i=1}^{n}(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_i)^2+\lambda\sum_{j=1}^{p}\hat{\omega_j}|\beta_j|$$

Adaptive Lasso의 정의에서 살펴봤듯이 RSS뒤에 붙은 제약조건에 $\hat{\omega_j}$라는 새로운 제약이 추가되어 있는 것을 볼 수 있다. 이 $\hat{\omega_j}$를 구하는 법은 다음과 같다.

$$\hat{\omega_j}=\frac{1}{(|\hat{\beta_j}^{ini}|)^\gamma}$$

이때 $\beta_j^{ini}$는 주로 Ridge regression에 의한 $X_j$의 회귀계수이다. $\beta_j^{ini}$가 크다면, $\hat{\omega_j}$는 그에 반비례하여 작아지게 되고, 그 $\beta_j$에는 더 작은 제약이 가해진다. 반대로 $\beta_j^{ini}$가 작다면, $\hat{\omega_j}$는 그에 반비례하여 커지게 되고, 그 $\beta_j$에는 더 큰 제약이 가해진다. 이를 통해 앞서 개념에서 언급했듯이 더 큰 회귀계수의 최소화에 목적함수 전체가 overfitting되는 것을 막고, Oracle property 또한 가질 수 있게 된다.

$$\hat{\beta}^{AL}=\underset{\beta}{\operatorname{argmin}}\sum_{i=1}^{n}(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_i)^2+\lambda\sum_{j=1}^{p}\hat{\omega_j}|\beta_j|$$

💡 Oracle property

1. Model has Consistency in variable selection
2. Model has optimal estimation rate

Oracle property를 간단히 요약하면, 그 모형에서 최적으로 선택된 변수가 항상 일관적으로 동일해야 하면서, 그 변수들로 fit된 모형의 예측도는 항상 최고여야 한다는 특성이라고 할 수 있다. 이 특성은 예측 정확도라는 예측 모형의 제 1목표와 최적의 변수 선택이라는 또다른 중요한 성질을 동시에 만족시킨다는 점에서, 훌륭한 모형이 가져야 할 일종의 충분조건처럼 여겨진다.

하지만 실제로 특정 $\lambda$가 주어진 상태에서 $\hat{\beta_j^{Lasso}}$는 이 특성을 만족하지 못하는 것으로 나타났다.  즉, Lasso에서 어떤 $\lambda$ 값이 주어졌을 때, 예측 정확도가 가장 높은 모형에는 일종의 noise variables를 포함해 최적의 변수선택에 실패하거나, 혹은 최적의 변수 선택을 시행 했을 때 그 모형의 예측 정확도가 가장 높지 않은 경우가 존재한다. Adaptive Lasso는 Lasso와 달리 Oracle property를 가지는 모형이다.

Adaptive Lasso가 Oracle property를 가진다는 사실에 대한 증명은 다음의 논문을 통해 참고하면 좋을 것 같다.

https://pages.cs.wisc.edu/~shao/stat992/zou2006.pdf

---

## 3주 간의 클린업을 마치며…

드디어 3주차 클린업이~~~~~ 끝났습니다!!!!

제가 욕심이 많아서, 팀원분들을 고생시킨 것 같아 미안한 마음 뿐입니다 ㅠㅠ 피피티라도 좀 덜 만들도록 뺄 부분은 최대한 빼고자 했는데, 그래도 양이 많았죠..? 회귀분석이 아무래도 전공필수에 기초적인 과목이다 보니 다시 들으면 지루할 것 같아서, 그냥 넘어갈 수도 있었던 부분을 많이 담으려고 했던 것 같아요. 왜 이런 수식이 나온 거지? 이걸 시각적으로는 어떻게 이해할 수 있는 거지? 하는 부분들이요..ㅎㅎ 그 덕분에 저도 전공새내기 시절 얼렁뚱땅 넘어갔던 회귀분석을 제대로 공부해볼 수 있었던 좋은 기회였는데, 여러분도 그런 경험이 되었다면 좋겠네요 ㅎㅎ 시간 내어 청강 와주신 분들도 감사합니다. 덕분에 저도 더 책임감을 갖고 열심히 준비할 수 있었던 것 같아요.

다들 중간고사 공부 열심히 하시고, 주제분석 때 또 동고동락(?)하면서 좋은 추억 남겨봐요~!