7. Regularized Regression
2019년 가을 학기

2019년 08월 25일
https://www.github.com/KU-BIG/KUBIG_2019_Autumn

이번 강의에서는 regression의 cost function에 penalty term을 추가한 분석 방법에 대해서 다룰 것이다. Regularized regression이 왜 필요하는지와 이에 속해있는 Ridge, Lasso, Elastic net을 살펴보고 회귀 분석 강의를 마치고자 한다.

1. Introduction

6강에서 선형 회귀(Linear regression)와 이것의 cost function인 MSE에 대해 배웠다. 하지만 MSE가 **많은 가정이 충족된다는 전제 하**에 최적의 방법이라고 이전 자료에서 언급했듯이, 현실에서 MSE만을 통해 모수를 추정하는 것에 한계가 있는 경우가 많다. 다음의 경우를 보자.

- 만약 feature가 수천, 수만개 이상이라면?

  변수들끼리 서로 연관되어 있을 가능성이 매우 높다. 즉 '변수들 간 서로 선형적으로 독립이어야 한다'는 선형회귀의 가정이 위반되고, 좋은 성능을 기대하기 어렵게 된다. 무엇보다, 단순 선형 회귀는 실제 y에 설명력이 없는 변수들을 없애주지 못하므로, 수많은 변수들이 포함된 모델을 보고 좋은 해석을 하기 어렵다.

- 만약 feature의 개수(p) 대비 데이터의 개수(n)가 적다면?

  선형회귀를 적합하기 위해서는 일단 최소 n=p이어야 최적의 모델을 구할 수 있다. 만약 n<p라면 training data에서 MSE를 최소화하는 모수 조합이 상당히 많아지고, 이들 중 대부분은 새로운 데이터에서는 제대로 예측을 하지 못하는 overfitting 현상을 야기할 가능성이 다분하다.

따라서 이 자료에서는 이러한 문제점들을 보완하는 선형 회귀 모델 Regularized Linear Regression에 대해 배우고자 한다. 먼저 Regularization에 대해 짚고 넘어가자.

2. Regularization

일단 regularization을 소개하기 전에, 'Bias와 Variance가 trade-off 관계이고, 모델이 복잡할수록 Bias는 감소하고 Variance는 증가하여 overfitting이 일어날 가능성이 높아진다'는 사실을 잘 알고 있을 것이다. Regularization의 핵심 목표는 바로 **overfitting을 방지**하는 것이다. 다시 말해서, **모델의 Bias는 살짝 증가시키더라도, Variance를 더욱 감소시키고자 하는 것**이다. 모델을 더욱 단순하게 만들자는 것이라고도 할 수 있다. 그렇다면 Regularization을 어떻게 한다는 것인가?

이는 모델의 cost function에 페널티항을 추가하는 것으로 이루어진다. 기존 선형회귀에서는 MSE만을 최소화했지만, Regularized 선형 회귀모델에서는 MSE+penalty를 최소화하는 것이다. 이런 의미에서 Regularization은 Penalization이라고 불리기도 한다. 더욱 구체적인 것은 Regularized regression의 종류인 Ridge, Lasso, Elastic Net와 함께 아래에서 설명하겠다.

3. Ridge Regression : L2-norm Regularization

먼저 다음과 같은 선형 회귀모델이 있다고 하자.

$$\mathrm{y_i} = \mathrm{w_{i1}} x_{i1} + w_{i2} x_{i2} + \cdots + w_{ip} x_{ip}, \qquad (i = 1, 2, \cdots, \mathrm{m})$$

앞서 Regularized regression의 비용함수는 MSE+penalty항이라고 했다. Ridge regression의 경우 그 penalty 항이 바로 가중치(회귀계수)의 제곱합이다. 이를 **L2 norm** 라고도 한다.

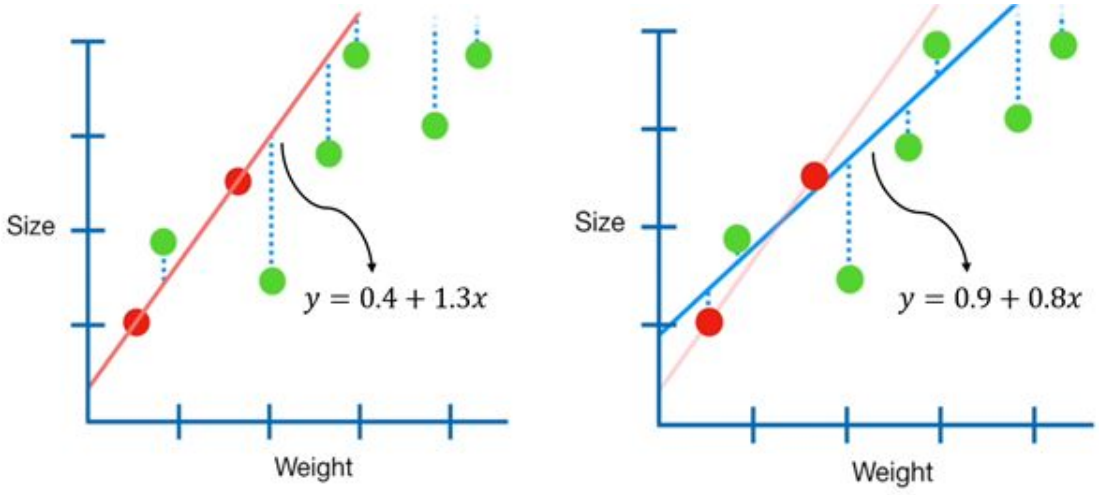
$$\mathrm{RSS_{Ridge}}(\mathbf{w}, \mathrm{b}) = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 + \boxed{\alpha \sum_{j=1}^{p} w_j^2}$$

Ridge Regression은 아래 수식과 같이 회귀계수에 대한 일정 제약 안에서 MSE를 최소화하는 회귀계수를 추정하는 것으로 볼 수 있다. (이는 나중에 보게 될 기하학적 해석에 도움이 된다.)

$$\hat{w}_{ridge} = \min_{w} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 \; subject\ to \sum_{j=1}^{p} w_j^2 \leq s$$

이렇게 페널티를 부여하게 되면, 큰 가중치 값을 가진 모델들에게 더 큰 페널티로 작용하게 된다. 즉, training data에 대해 성능이 같은 두 모델이 존재할 때, Ridge regression은 비용함수의 전반적인 합이 더 작은 모델을 선호하게 되는데, 대체로 그 경우는 변수들의 회귀계수가 작은 값을 가질 때이다. 다음의 예시를 보자. (Youtube 채널 StatQuest참고.)

아래 그림에서 빨간색 점이 training data, 초록색 점이 test data이다. 먼저 왼쪽 그림부터 보자.



빨간 직선은 MSE를 최소화시키는 일반 선형 회귀모델이다. 하지만, 막상 실제 초록색 점들과는 상당한 거리가 있는 것으로 나타난다. 이 때, 알파를 1이라 하고, Ridge Regression의 제곱합 공식을 도입해 보자.

$$\sum_{i=1}^{2}(y_i - \hat{y})^2 + w^2 = 0 + 0 + 1.3^2 = 1.69$$

이제 오른쪽 그림의 경우를 보자. 파란 직선은 training data와는 조금 멀어졌지만 test data와는 가까워졌다. 빨간 점들과 직선의 차이 $|y - \hat{y}|$를 0.3, 0.1 정도라고 해보자. 마찬가지로 계산하면 다음과 같다.

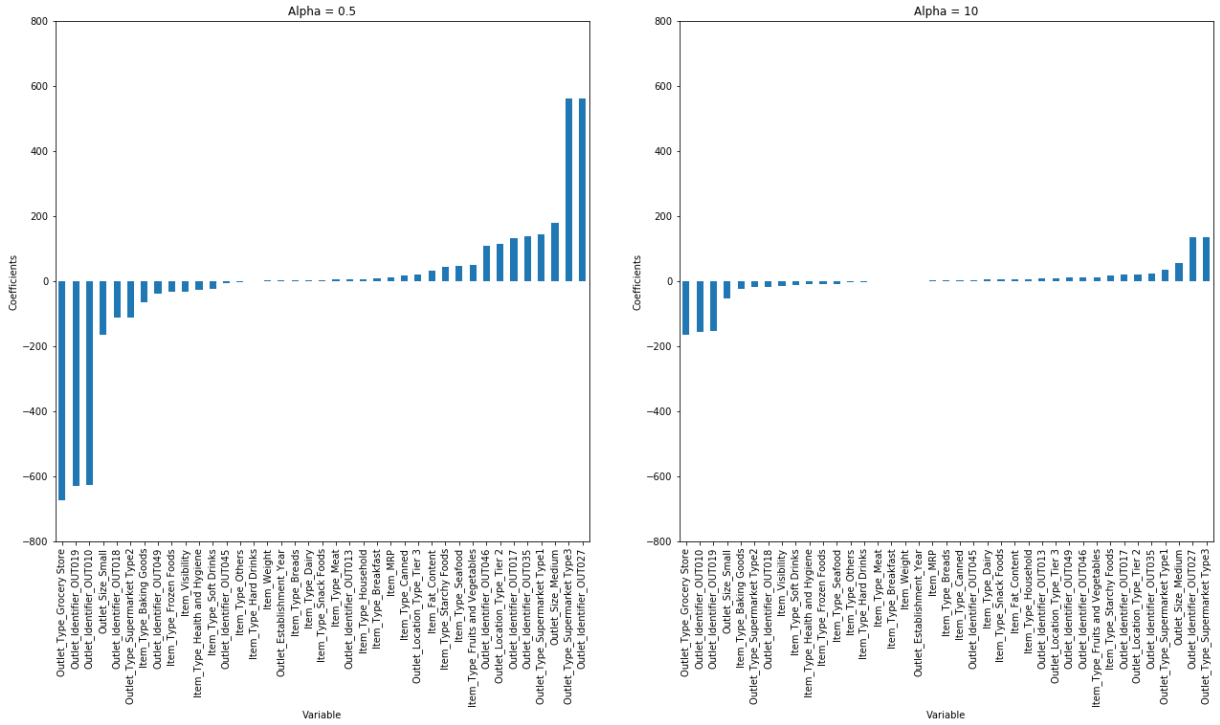
$$\sum_{i=1}^{2}(y_i - \hat{y})^2 + w^2 = 0.3^2 + 0.1^2 + 0.8^2 = 0.74$$

계산 결과, 오른쪽 그림의 제곱합의 값이 더 작다. 따라서 Ridge Regression에서는 파란 직선을 더 선호한다. Training data에서 약간의 bias를 용인한 결과, test data에서 variance가 크게 감소할 것을 기대할 수 있기 때문이다. 이를 통해 회귀 분석 결과로 얻어진 회귀계수들의 평균이 실제 회귀계수와 조금 다를지언정 예측 결과의 폭을 줄임으로써 안정적인 결과를 얻을 수 있다.

이런 점에서 Ridge regression은 추정 모수 개수에 비해 데이터의 수가 적을 때 유용하다는 것도 유추할 수 있다. 앞서 설명했듯이, MSE만을 이용할 때에는 추정 모수 만큼의 데이터 수가 있어야만 모델 비교가 된다. 반면, Ridge에서는 MSE도 감소시켜야 하지만, 각 회귀계수의 크기에도 일정 제약이 존재하기 때문에 가능한 모수 조합이 감소하게 된다.

한편, α는 정규화의 정도를 조절하는 hyperparameter이다. α값이 클수록 정규화의 정도가 커지게 되어, 대부분의 회귀계수의 값이 거의 0에 가깝게 되고, 더욱 단순한 모델이 된다. 반면에 α값이 0일 때는 일반적인 linear regression과 다를바 없게 된다.

아래 그림에서 왼쪽 그림은 α의 크기가 0.5일 때이고, 오른쪽 그림은 α의 크기가 10일 때이다. α값이 커질수록 회귀계수의 크기가 감소하는 것을 확인할 수 있다. 다만 Ridge는 다음에 설명할 Lasso와 같이 설명력이 약한 변수의 회귀계수를 완전히 0으로 줄이지 못한다.



https://bit.ly/2rSSCJu

4. Lasso Regression : L1-norm Regularization

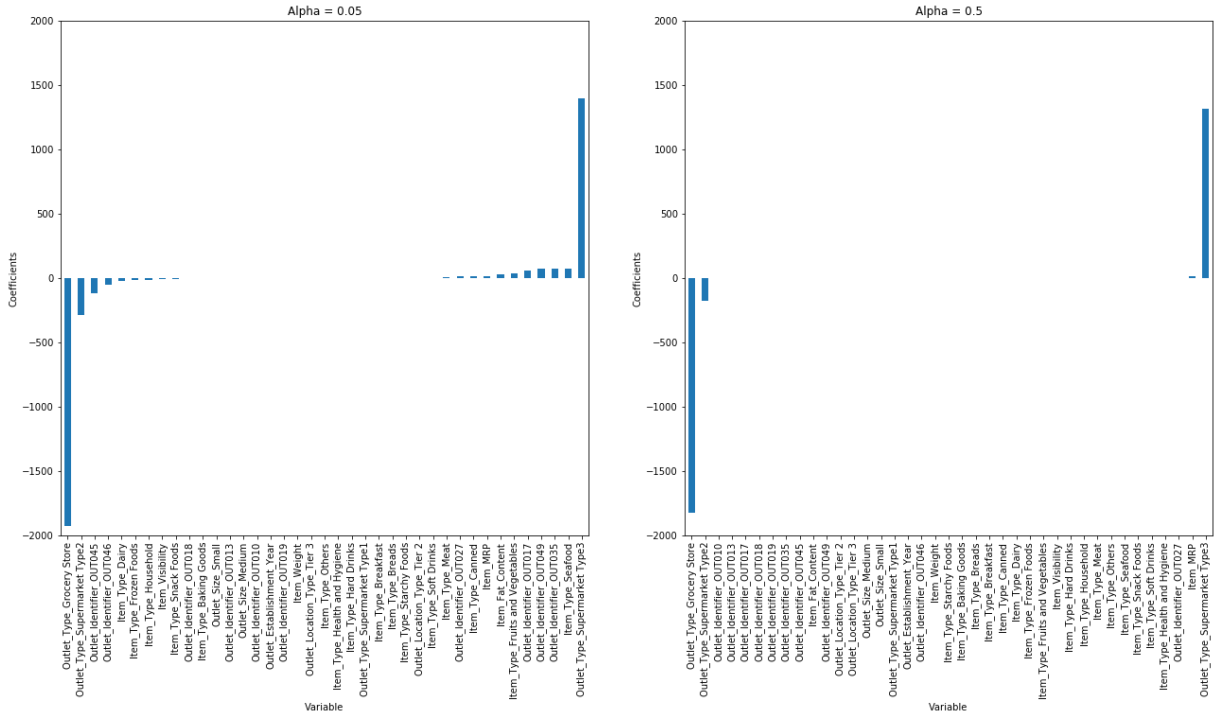
이제 Lasso Regression에 대해 알아보자. Ridge Regression을 이해했다면 Lasso Regression도 이해한 셈이다. 이들의 공식 면에서의 차이는 penalty항에서 제곱이 절대값으로 바뀌었다는 것밖에 없다. 이 페널티를 **L1 norm**이라고 한다.

$$\mathrm{RSS}_{\mathrm{Lasso}}(\mathbf{w}, \mathrm{b}) = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 + \boxed{\alpha \sum_{j=1}^{p} |w_j|}$$

마찬가지로, 일정 제약 안에서 MSE를 최소화하는 회귀계수를 추정한다.

$$\hat{w}_{Lasso} = \min_{w} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 \; subject\ to \sum_{j=1}^{p} |w_j| \leq s$$

Penalty가 L1 norm으로 바뀌게 되면서 Lasso regression은 중요한 특성을 가지게 되었다. 바로 Lasso regression에서는 일부 회귀계수를 0으로 보낼 수 있다는 것이다. (Ridge에서는 거의 0에 가깝게 보내는 것이지, 0이 되게 하지는 않는다!) 아래 그림을 보자.

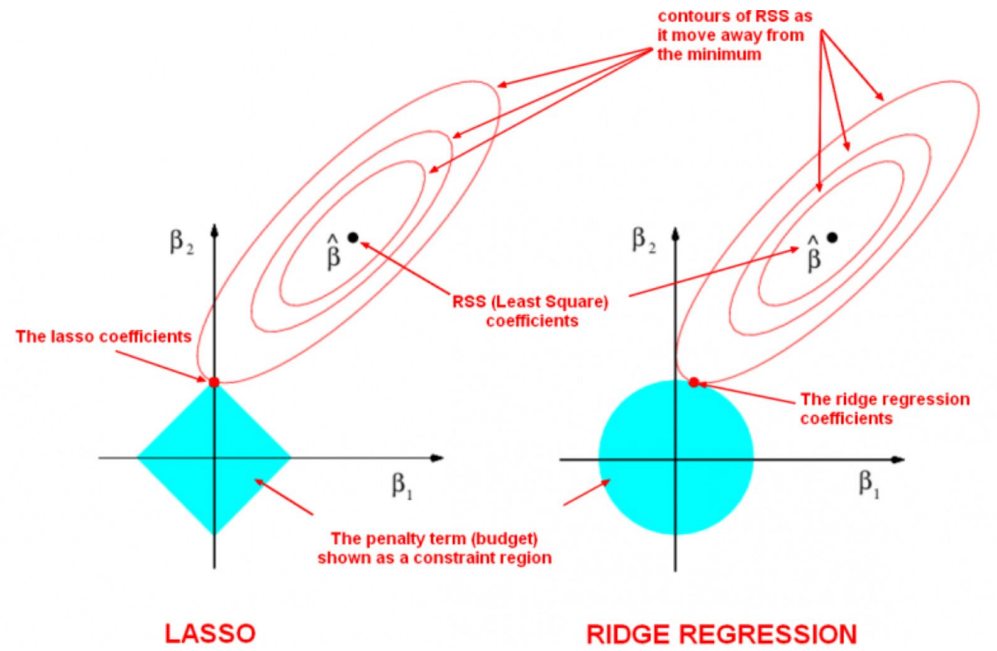


https://bit.ly/2rSSCJu

왼쪽 그림은 α가 0.05, 오른쪽은 α가 0.5일 때이다. α의 값이 커질수록 제일 중요한 변수들만 남고 나머지 변수들은 회귀계수가 0이 되는 것을 확인할 수 있다. 즉, 가장 중요한 변수들만 0이 아니게 되어, 자동으로 feature selection(변수 선택) 기능을 한다고 할 수 있다.

그렇다면 왜 Ridge regression과는 다르게 Lasso regression에서는 회귀계수가 0이 될 수 있는 것일까? 이를 기하학적으로 이해해보자. 앞서 설명했던 Ridge regression과 Lasso regression의 제약 범위를 2차원 평면으로 나타내어 볼 것이다. 이해하기 쉽게 변수가 2개라고 하자.

$$y = \beta_1 x_1 + \beta_2 x_2$$



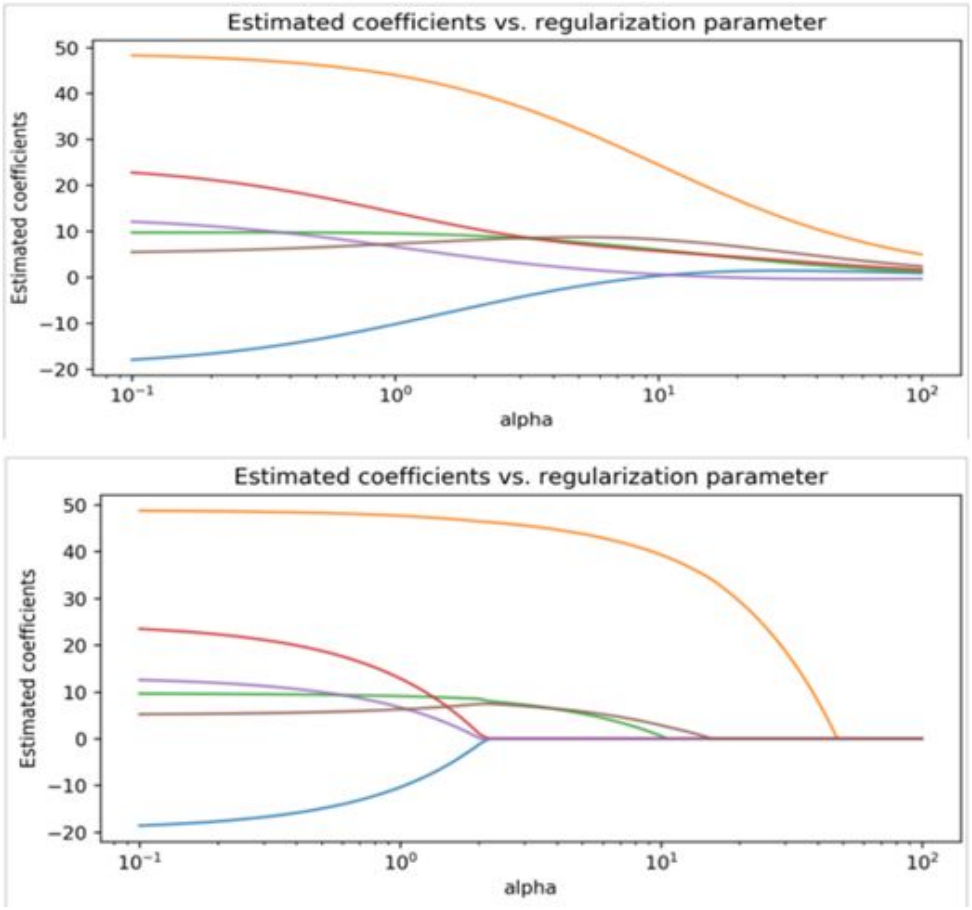
https://bit.ly/2Ugf7Hu

먼저 왼쪽 Lasso의 경우를 보자. 빨간색 등고선은 $\mathrm{MSE}(\beta_1, \beta_2) = c, \text{ (for arbitrary c)}$를 그림으로 표현한 것이다. 청색 다이아몬드 도형은 페널티항 ($|\beta_1| + |\beta_2| \leq s$)의 시각적 표현이다. 오른쪽 Ridge의 경우를 보면, 빨간색 등고선은 동일하고, 청색 도형이 원으로 바뀌었다. Ridge regression의 penalty항 ($\beta_1^2 + \beta_2^2 \leq s^2$)을 시각화한 것이다.

이때 빨간색 등고선 안에 있는 $\hat{\beta}$는 기존 비용함수 MSE를 통해 구해진 회귀계수 벡터라고 할 수 있다. Ridge와 Lasso regression에서 새롭게 추정되는 회귀계수 벡터는 각 제약범위 안에서 bias를 최소화하기 위해 빨간색 등고선과 청색 도형이 만나는 지점이 된다. 이때, Lasso regression의 제약 범위는 다이아몬드 형태이기 때문에 꼭짓점이 존재하므로 이들의 교점이 정확히 한 축 위에 있을 가능성이 충분히 생긴다.

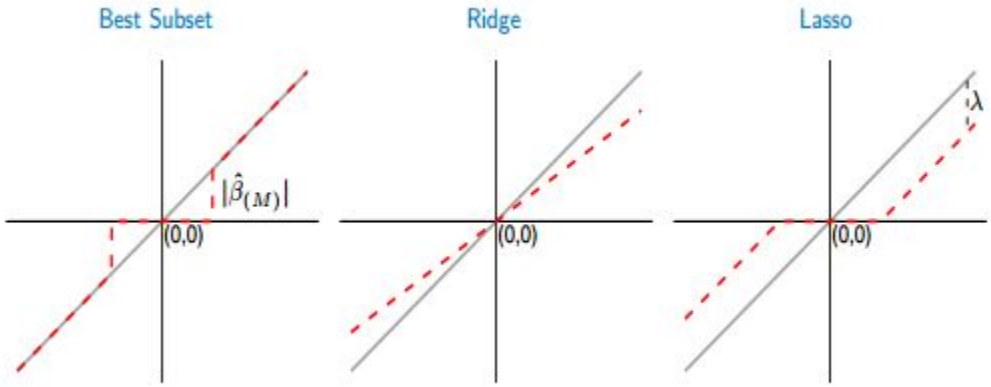
위 그림과 같이 교점이 정확히 β_2 축 위에 있게 되면, β_1 은 0이 될 것이다. 반면, Ridge regression의 경우 제약 범위가 원의 형태이기 때문에, 이들의 교점이 정확히 한 축 위에 있기는 힘들다. 따라서 Lasso는 변수들을 0으로 보내버릴 수 있지만, Ridge는 변수들을 0으로 가깝게는 줄이되 0이 되게 하지는 않는다.

이제 아래 그림을 보면 어떤 것이 Lasso와 Ridge인지 구별할 수 있어야 한다. α 값이 커질수록 위 그림은 계수들이 0에 가까워지고 아래 그림은 아예 0에 수렴한다. 물론 위 그림이 Ridge regression, 오른쪽 그림이 Lasso regression이다.



https://www.coursera.org/learn/python-machine-learning/

아래의 그림을 통해서 Ridge와 Lasso의 biasedness를 한눈에 볼 수 있다. 각 plot의 x축은 OLS(ordinary least squares)를 통해서 도출된 기본적인 회귀계수이다. y축은 Regularized regression으로부터 나온 회귀계수이다.



https://bit.ly/2UaRiRb

Ridge 그래프를 보면 회귀 계수의 선이 OLS로 구해진 직선보다 y=0 쪽으로 기울어져 있음을 확인할 수 있다. 회귀계수가 0에 가까울수록 L2 penalty가 계수를 0으로 잡아당기는 힘이 약하고 회귀계수가 커질수록 계수를 0으로 잡아당기는 힘이 큰 것을 볼 수 있다. 이는 L2 penalty가 제곱으로 구성되어있어 값이 커질수록 더더욱 penalty가 커지기 때문이다.

Lasso 그래프를 보면 OLS로 구해진 직선과 평행하게 그 절대값이 λ만큼 작다는 것을 알 수 있다. 이는 L2 penalty와 달리 L1 penalty는 절대값으로 구성되어있기 때문이다. 이 때문에 Lasso의 계산은 Ridge보다 다소 복잡하다. 절대값은 미분 불가능하기 때문에 sub-gradient 개념을 가지고 Lasso의 비용함수를 최소화시킨다.

가장 왼쪽의 Best subset은 이상적인, 우리가 궁극적으로 원하는 작은 값 만을 0으로 보내는 경우이다. 이와 최대한 비슷한 penalty term들이 개발되고 있다. 그 중에서 가장 대표적인 것은 Ridge와 Lasso의 치명적인 단점인 oracle property를 만족하지 못한다는 것을 극복한 SCAD가 있다. Oracle property와 SCAD에 대해 자세히 알고싶은 사람은 구글링 하기 바란다.

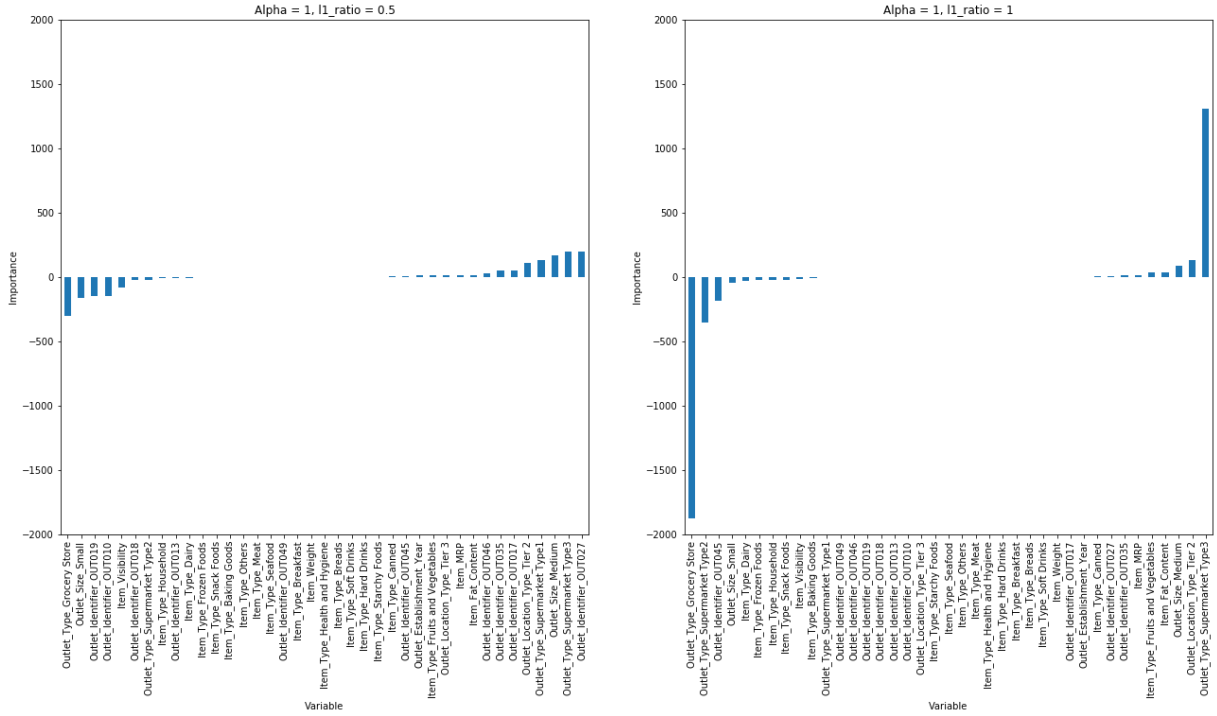
정리하면, 만약 오로지 일부 변수들만이 중간 이상의 영향력을 행사할 때에는 Lasso regression이 적절하다고 할 수 있다. 반면, 많은 변수들이 중간 이하의 영향력을 모두 행사할 때에는 Ridge regression이 더 적절하게 된다. 하지만 실제로 이렇게 데이터에 대한 정보를 미리 잘 알고 있는 경우는 드물다. 더군다나 feature의 개수가 수천, 수만개 이상일 때 상황의 심각성은 더 커진다. 이 때에는 이 두가지 특성을 모두 지닌 Elastic Net을 사용하면 된다.

5. Elastic Net

Elastic Net은 Lasso와 Ridge를 결합한 모델이다. 따라서 이들의 성격을 모두 가지고 있다.

$$\mathrm{RSS}_{\mathrm{Elastic}}(\mathbf{w}, b) = \sum_{i=1}^{m}(y_i - \hat{y}_i)^2 + \lambda[\frac{1}{2}(1-\alpha)\sum_{j=1}^{p} w_j^2 + \alpha\sum_{j=1}^{p}|w_j|]$$

Elastic Net은 hyperparameter가 두 개(α, λ) 존재한다. λ는 penalty의 강도를 나타내는 hyper parameter이다. α는 두 penalty 중에서 Lasso penalty의 비율이다. 따라서 α가 1일 때는 Lasso regression이 되고, α가 0일 때는 Ridge regression이 된다. 이들을 선택하는 과정은, α과 λ의 범위를 정해서 이들의 조합들 중 error가 최소가 되는 점을 찾으면 된다. 아래의 그림을 보자.



https://bit.ly/2rSSCJu

잘 보이지는 않겠지만 위의 plot에서 Alpha는 Ridge와 Lasso penalty앞에 곱해진 상수들의 합이며 l1_ratio는 Alpha중에서 L1 penalty에 할당된 비율을 의미한다. 즉, 위 식의 형식으로 바꾸자면, Alpha는 λ에 해당하고, l1_ratio는 α에 해당한다. 즉 오른쪽 그림은 $\lambda = 1, \alpha = 0.5$ 인 Elastic net이며, 왼쪽은 $\lambda = 1, \alpha = 1$ 인 Lasso regression 결과에 해당한다.

Elastic Net은 특히 변수들 간 상관관계가 존재할 때 유용하다. 변수들 간 상관관계가 존재할 때, Lasso는 상관관계가 존재하는 변수들 중 하나만 선택하고 나머지를 배제하는 경향이 있고, Ridge는 모든 회귀계수들을 전반적으로 줄인 채 남겨놓는다는 특징이 있다. 변수들을 모두 남기는 것도 문제이지만, 상관관계가 존재하는 변수들을 무조건 대부분 없애버리는 것은 정보가 손실되면서 큰 문제를 야기할 수 있다.

실제로 변수들 간 상관관계가 높을 때 Lasso의 성능이 Ridge보다 떨어지는 경우가 급증한다고 한다. Elastic Net은 변수들 간 상관관계가 존재할 때, 변수들끼리 그룹을 지어 계수들을 감소시키고 그 그룹 전체를 모델에 남기거나 제거한다. 즉, 상관관계가 존재하는 변수들에 대하여 Lasso와 다르게 동시 선택이 가능해졌다고 보면 된다. 정리하면, Elastic Net은 Lasso의 feature elimination 기능과 Ridge의 coefficients reduction 기능을 모두 하는 것이다.