글로벌투자전략팀

김동영, CFA
Quant Analyst
dy76.kim@samsung.com

Macro팀

안미성, Ph.D.
Economist
misung11.ahn@samsung.com

퀀트 모델링 A to Z

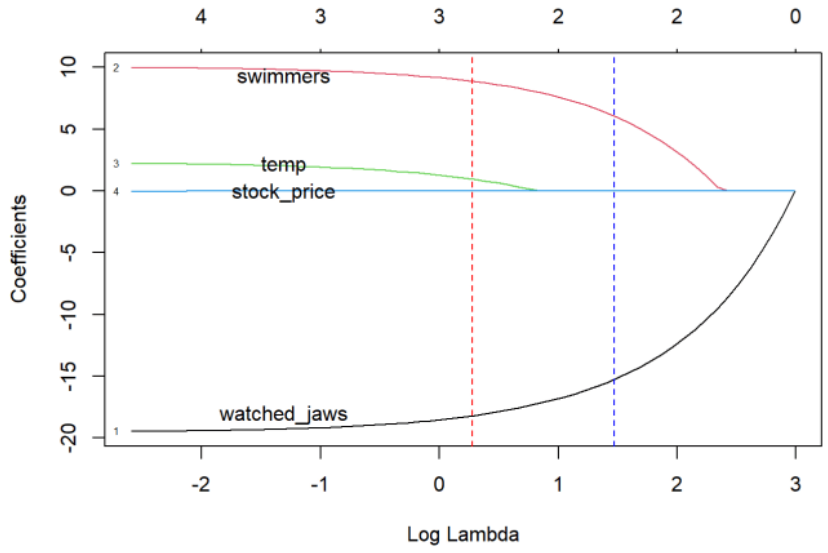
(3) Penalized Regression (벌점 회귀 분석)

- Ridge, LASSO, Elastic Net regression 대한 설명 및 사용 방법 수록

Penalized Regression(벌점 회귀) 혹은 Regularized Regression(규제화 회귀)라는 것은, 회귀분석 방법의 일종으로 회귀계수 축소를 통해 모형의 과적합을 피하려는 기법을 말한다.

"벌점화"란 회귀계수 값의 과잉에 대해서 벌점을 준다는 의미로, 회귀분석식의 회귀계수 해를 찾을 때, 벌점화 조건이 추가된다는 뜻이다. 구체적인 방법에 따라 Penalized Regression에는 Ridge, LASSO, Elastic Net 등의 방식이 존재한다.

파이썬의 머신러닝 패키지인 sklearn에서 Ridge, Lasso, ElasticNet 클래스를 통해 해당 모델들을 손쉽게 사용할 수 있다.

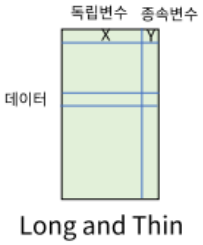


I. Penalized Regression

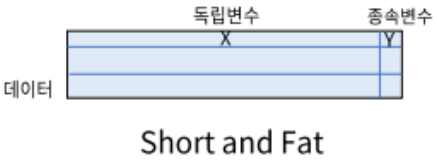
Penalized Regression(벌점 회귀) 혹은 Regularized Regression(규제화 회귀)라는 것은, 회귀분석 방법의 일종으로 회귀계수 축소를 통해 모형의 과적합을 피하려는 기법을 말한다.

데이터 분석을 위한 이상적인 데이터 형태는 1) 독립변수 개수는 작고 2) 관측 샘플은 많은 모양이다. 그러나, 현실에서는 관측 샘플은 적고, 동원 가능한 독립변수 개수는 많은 데이터 형태가 일반적이다. 이 경우, 독립변수를 많이 동원하면 할수록 모델의 정확도가 올라가 보이지만, 실제로는 과적합(overfitting)의 문제가 발생하게 된다.

이상적인 데이터 형태

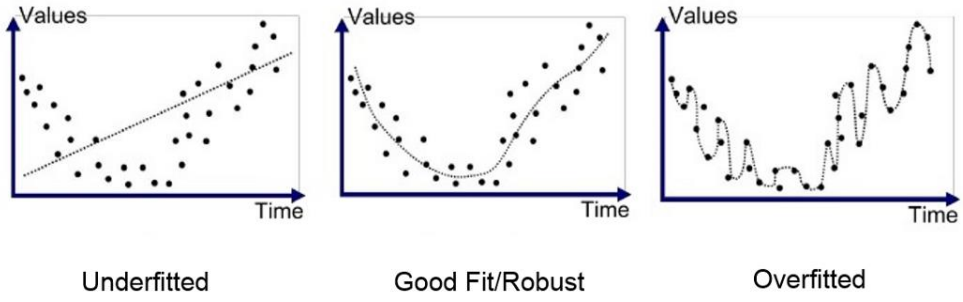


일반적인 데이터 형태



참고: 현실은 항상 다중회귀분석의 세계임

과적합(overfitting) 문제



위 오른쪽 그림은 overfitting의 문제점을 보여준다. 오른쪽 그림은 모든 데이터 샘플을 100% 설명하도록 만들어진 모델이다. 인위적인 측면이 너무 강하다고 볼 수 있다. '너무 많은 변수들을 고려하다 보면' 과거 데이터에만 완벽히 들어맞는 모델을 만들게 된다. 이런 과적합된 모델은, 실제 미래 예측력은 떨어질 수밖에 없다.

Penalized Regression는 회귀 계수 축소를 통해서 단순한 모델을 만들어 이런 과적합을 피하는 변형된 형태의 회귀분석 기법이다. "벌점화"란 회귀계수 값의 과잉에 대해서 벌점을 준다는 의미로, 회귀분석식의 회귀계수 해를 찾을 때, 벌점화 조건이 추가된다는 뜻이다. 구체적인 방법에 따라 Penalized Regression에는 Ridge, LASSO, Elastic Net 등의 방식이 존재한다.

지난번 퀀트 모델링 A to Z 1편에서 회귀분석 모델의 OLS 방법은, 오차항의 제곱의 합을 최소화 하도록 회귀계수를 찾는 방법이라고 이야기한 바 있다.

기존 OLS (일반 다중회귀분석) 방식:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + u \text{ 의 다중회귀분석 식에서}$$

$$\sum_{i=1}^{n} (y_i - \hat{y_i})^2 \text{ 을 최소화하는 } \boldsymbol{\beta}(\beta_0, \cdots, \beta_k) \text{찾기}$$

여기서, $\hat{y_i}$은 회귀분석 모델 상 y_i의 추정치임

Ridge 회귀:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + u \text{ 의 다중회귀분석 식에서}$$

$$\sum_{i=1}^{n} (y_i - \hat{y_i})^2 + \lambda \sum_{j=1}^{n} {\beta_j}^2 \text{ 을 최소화하는 } \boldsymbol{\beta} \text{ 찾기}$$

Ridge(리지) 회귀는, 처음에 본 OLS의 $\boldsymbol{\beta}$ (회귀계수) 해를 찾는 목적함수에 패널티 항목을 추가한 방식이다.

Ridge(리지) 회귀에서의 패널티는 $\lambda \sum_{j=1}^{n} {\beta_j}^2$항이다. 즉, 각 독립변수의 회귀계수인 β_j의 크기가 작은 쪽으로 선택하게끔 목적 함수가 바뀐 방식이다.

위 식은 회귀계수($\boldsymbol{\beta}$)의 해를 선택할 때, 1) 샘플의 제곱 예측 오차와 2) 회귀계수의 제곱 값을 동시에 최소화하라는 서로 다른 두 가지 목표가 균형을 이루게 만드는 식이다. 이에 따라 회귀분석의 회귀계수 해를 구할 때, 오차 최소화 뿐 아니라, 회귀계수의 크기가 작아지는 형태로, 해가 결정된다.

(Ridge 회귀에서는 벌점화 항목에 제곱합의 산식을 사용하는데, 이를 L2 norm 방식이라고 한다)

λ(람다)가 0으로 갈수록 목적 함수의 오른쪽 항이 무시되므로, Ridge 회귀의 결과는 기본 OLS 결과로 수렴하게 된다. 만역, λ가 매우 크다고 하면, 왼편의 제곱 예측 오차 항은 무시되고 오른편의 β_j크기 최소화만 중요해진다. 이 때는, 모든 회귀계수 β_j가 0에 수렴하는 결과가 나온다.

이 λ는 목적 함수의 두 타켓 간의 조화를 의미하는 하이퍼 파라미터(모델 외부 세팅 값)에 해당한다. 보통 대부분의 모델링은 하이퍼 파라미터를 필요로 한다. 예를 들어, 이전 HP 필터에서도 스무딩 정도를 나타내는 평활화 계수 λ가 있었다. Ridge 회귀에서 쓰는 하이퍼 파라미터 λ의 경우에는, 보통 교차 검증 등을 통해서 학습으로 결정하는 편이다. 머신 러닝 방법론에서는 여러 번의 교차 검증 등을 통해서 하이퍼 파라미터 값을 결정하는 방식(튜닝 과정)을 주로 사용한다.

한편, 리지 회귀의 장점은 회귀계수의 해를 닫힌 형식의 공식으로 구할 수 있다는 점이다. 먼저, 퀀트 모델링 A to Z 1편 리포트의 Appendix를 보면, 다중 회귀분석의 회귀계수 해에 대한 행렬 형태 공식이 나온다.

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y}$$

이와 유사하게 리지 회귀의 회귀계수 해 공식은 다음과 같다.

$$\hat{\boldsymbol{\beta}}^{Ridge} = (\boldsymbol{X}'\boldsymbol{X} + \lambda \boldsymbol{I})^{-1}\boldsymbol{X}'\boldsymbol{y}$$

LASSO 회귀:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + u \text{ 의 다중회귀분석 식에서}$$

$$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^{n}|\beta_j| \text{ 을 최소화하는 } \boldsymbol{\beta} \text{ 찾기}$$

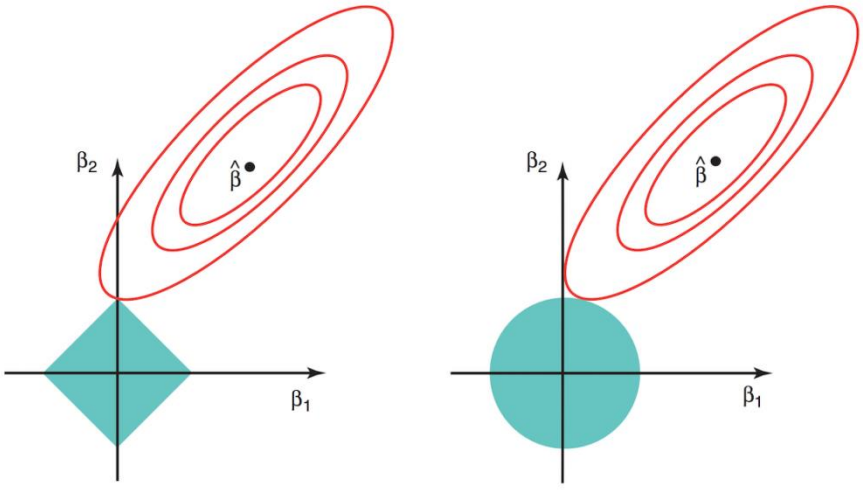
LASSO(라쏘, Least Absolute Shrinkage and Selection Operator) 회귀 또한, OLS의 $\boldsymbol{\beta}$ (회귀계수) 해를 찾는 목적함수에 별도의 패널티 항목을 추가하는 방식이다.

LASSO(라쏘) 회귀에서의 패널티는 $\lambda \sum_{j=1}^{n}|\beta_j|$항이다. 즉, 여기서도 각 독립변수의 회귀계수인 β_j의 크기가 작은 쪽으로 선택하게끔 목적 함수가 바뀐 방식이다. 다만, LASSO 회귀에서의 회귀계수 최소화의 방식은 절대값을 사용하는 방식으로, 리지 회귀에서의 제곱합을 최소화하는 방식과 약간 다르다.

(LASSO 회귀에서는 벌점화 항목에 절대값의 산식을 사용하는데, 이를 L1 norm 방식이라고 한다)

LASSO 회귀의 특징은, 독립변수 개수 축소를 더 잘 한다는 점이다. LASSO 회귀에서는 절대값 산식을 사용하므로, 여러 독립 변수들의 회귀계수를 아예 0으로 만들 가능성을 높였다. 회귀계수가 0이라는 것은, 해당 독립변수를 다중회귀분석 식에서 제외한다는 뜻이며, 이 것이 바로 변수 축소를 의미한다. LASSO 회귀의 변수 축소 측징은 다음 그림 예제를 통해서 쉽게 이해 가능하다.

LASSO 회귀의 해 찾기 방식 **Ridge 회귀의 해 찾기 방식**



참고: 독립변수가 2개인(따라서 회귀계수가 β_1, β_2 2개인) 다중회귀분석 가정

그림 상에서 등고선 안의 $\hat{\beta}$는 단순 OLS 상의 기본 해 위치를 말한다. 벌점 회귀 목적 함수의 왼쪽 항만 있고 오른쪽 항이 없다면, $\hat{\beta}$ 위치가 회귀계수의 해 위치가 된다. 빨간 색 등고선은 목적 함수 왼쪽 항 즉 잔차 제곱합이 동일한 지점들의 집합을 뜻한다. LASSO 그림에서 녹색 사각형은, 목적 함수의 오른쪽 항 즉 회귀계수 크기 패널티가 동일한 등고선을 겹쳐 그린 것이다. LASSO 회귀에서는 '빨간 색 등고선 상의 해 후보' 중에서 '녹색의 사각형 회귀계수 크기 패널티'와 만나는 지점이 최종적인 회귀계수 해가 된다. 이 때, 절대값 방식을 쓰는 LASSO 회귀에서는 녹색 패널티 등고선이 각지게 되어, β_1이나 β_2가 0이 되는 축 상의 위치가 최종 해

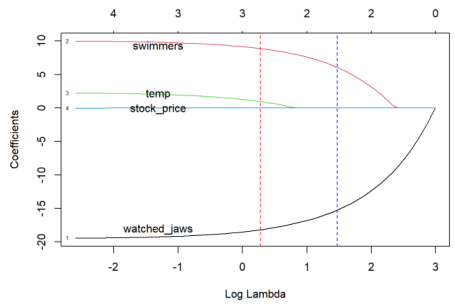
가 된다. 그림 상에서는 $\beta_1 = 0$인 y축 상의 점이 회귀계수의 최종 해가 된다. 절대값 형태의 패널티를 쓰게 되면 '각진 패널티 등고선'을 가지기 때문에, 많은 회귀계수의 추정치를 0으로 만들 수 있게 된다.

최초에 독립변수를 많이 집어넣은 회귀 분석에서는, LASSO 회귀를 쓰면 효과적으로 독립변수의 축소를 만들어 낼 수 있다.

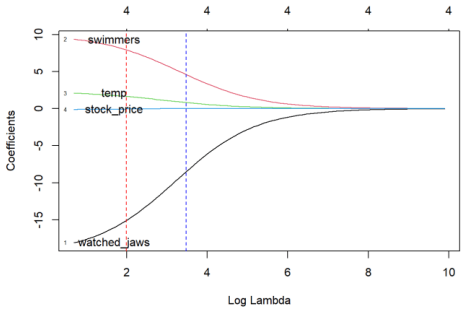
LASSO 회귀에서도 샘플의 잔차 제곱합(왼쪽 항)과 회귀계수 크기 패널티(오른쪽 항) 간의 중요도를 배분하는 하이퍼 파라미터(모델 외부 세팅값)인 λ가 존재한다. LASSO 회귀에서의 λ 또한, 교차 검증 등을 통해서 튜닝으로 결정하게 된다.

LASSO는, Ridge 회귀와는 다르게 '닫힌 해' 즉 '해의 공식'이 없다는 것 또한 특징이다. 보통 컴퓨터를 통해서 LASSO 회귀의 해를 찾게 된다.

λ 에 따른 LASSO 회귀계수 변화 그래프



λ 에 따른 Ridge 회귀계수 변화 그래프



위의 샘플 그래프를 보면, LASSO와 Ridge의 따른 회귀계수 변화 패턴의 차이를 볼 수 있다. λ를 올릴수록(x축) 회귀계수 축소가 진행되는데, LASSO 회귀에서는 각 독립변수의 회귀계수가 0으로 순차적으로 도달하는 모습을 보이고 있다. 반면, Ridge 회귀에서는 회귀계수들이 0에 빨리 도달하는 방식이 아니고, 회귀계수 값의 축소가 전반적으로 같이 진행되는 모습을 보여준다.

Elastic Net 회귀:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + u \text{ 의 다중회귀분석 식에서}$$

$$\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \lambda_1 \sum_{j=1}^{n}|\beta_j| + \lambda_2 \sum_{j=1}^{n}{\beta_j}^2 \text{ 을 최소화하는 } \boldsymbol{\beta} \text{ 찾기}$$

짜장을 시킬지 짬뽕을 시킬지 고민될 때 해법은, 짬짜면을 시키는 것이다. 벌점 회귀에서의 짬짜면이 바로 Elastic Net(일레스틱넷) 방식이다.

Elastic Net은 LASSO의 패널티 수식과 Ridge의 패널티 수식을 모두 한번에 가지고 있다. 이를 통해 회귀계수를 축소하는 회귀분석을 진행하게 된다. 위 식에서 하이퍼 파라미터인 λ_1, λ_2의 조절을 통해서 전체적인 패널티 수준, 즉 LASSO의 강조 정도와 Ridge의 강조 정도를 조절하게 된다. λ_1, λ_2가 모두 0이 아닌 한 Elastic Net은 원래 회귀분석에도 잘 맞고, 회귀계수의 축소가 잘 이뤄지는 방식으로 회귀계수 해를 찾게 된다.

Elastic Net의 하이퍼 파리미터 λ_1, λ_2도 교차 검증 등을 사용하여 모델 튜닝을 통해서 보통 결정하게 된다.

II. 예제

예를 들어, 다양한 매크로 환경 하에서 POSCO 기업의 주가가 결정되는 구조를 알고 싶다고 하자.

POSCO 주가의 변동에는, 기본적인 한국 주식시장의 흐름(코스피), 여러 환율 변수, 원자재 변동의 대표 지표인 WTI 유가 등이 영향을 준다고 생각할 수 있다(독립변수로 철강가격을 넣는 것은, 너무 뻔한 결과이자 예측이 어려운 변수를 동원하는 사례로 볼 수 있음). 이를 기초로 입수 가능한 데이터들로 POSCO 주가에 대한 다중회귀분석식을 러프하게 세우면 다음과 같다.

$$\left[POSCO\ 월수익률\right] = \beta_0 + \beta_1\left[KOSPI\ 월수익률\right] + \beta_2\left[원/달러\ 월수익률\right] + \beta_3\left[원/엔\ 월수익률\right] + \beta_4\left[WTI\ 월수익률\right] + u$$

위식은 종속변수로 POSCO 월수익률을 넣고 독립변수 개수는 4개인 다중회귀분석의 형태다. 여기에 들어가는 표본 데이터는 아래 표와 같이 조사했다. 일단 데이터를 모으다보니 원/달러, 원/엔 변수가 둘 다 들어간 모습이 중복 투자라는 느낌을 다소 준다.

X, y 표본 데이터

월	y: POSCO 월수익률	x_1: KOSPI 월수익률	x_2: 원달러 월수익률	x_3: 원엔 월수익률	x_4: WTI 월수익률
2011-01-31	-6.8%	0.9%	-1.2%	-2.9%	0.9%
2011-02-28	1.3%	-6.3%	0.6%	1.8%	5.2%
2011-03-31	9.8%	8.6%	-2.8%	-3.6%	10.1%
2021-03-31	13.7%	1.6%	0.7%	-1.5%	-6.9%
2021-04-30	13.8%	2.8%	-1.7%	-1.0%	7.5%
2021-05-31	-2.5%	1.8%	-0.1%	-0.1%	4.3%

자료: Bloomberg

우선 기본적인 OLS 방식의 다중회귀분석을 실행하면, 회귀계수 해는 다음과 같이 나온다.

β_0	β_1	β_2	β_3	β_4
0.00	0.89	-0.37	0.01	0.03

즉, OLS 방식의 최종 회귀분석식은 다음과 같다.

$$posco\_m = 0.00 + 0.89kospi\_m - 0.37krwusd\_m + 0.01krwjpy\_m + 0.03wti\_m + u$$

분석식 결과를 보면, 원달러 월수익률의 회귀계수는 -0.37, 원엔 월수익률의 회귀계수는 0.01이다. 즉, 원/달러가 1% 상승하면 POSCO 주가는 0.37% 하락하고, 원/엔이 1% 상승하면 POSCO 주가는 원달러 때와 다르게 0.01% 상승한다는 해석이다. 여기서, 실제로 POSCO 주가가 원/엔의 영향을 받는지도 의문이고, 원/엔 영향이랑 원/달러 영향이랑 반대 방향인 것도 수긍하기 어렵다.

이런 결론이 나오는 것은, OLS 방식의 기본 회귀분석은 항상 주어진 샘플 데이터를 제일 잘 설명하는 형태로만 회귀계수를 추정하기 때문이다. 단순 OLS에서는 독립변수로 데이터를 넣기만 하면 항상 해당 회귀계수(베타)가 지정될 수밖에 없다.

여러 벌점 회귀들을 이 문제점을 해결할 수 있다. 동일한 데이터에 대해서, 벌점 회귀의 하나인 LASSO 회귀분석을 실시해 보았다.

먼저, LASSO의 람다 기준을 0.0001로 설정한 LASSO 회귀의 분석 결과는 다음과 같다.

LASSO 회귀 결과(람다=0.0001): 회귀계수 해

β_0	β_1	β_2	β_3	β_4
0.00	0.87	-0.20	0.00	0.04

LASSO 회귀는, 회귀분석 식을 찾을 때, 회귀계수의 절대값 합이 작도록 유도하게 된다. 목적함수에 약한 수준의 절대값 합 최소화 조건을 넣음으로써 원/엔 월수익률 변수의 회귀계수(β_3)가 0으로 바뀌었다. 즉, LASSO를 동원해서 독립변수 수를 하나 줄인 간결한(parsimonimous) 모델을 만든 것이다.

이 결과에 따른 새로 쓴 회귀분석식은 다음과 같다.

LASSO 첫 번째:

$$posco\_m = 0.00 + 0.87kospi\_m - 0.20krwusd\_m + 0.04wti\_m + u$$

이 식은 POSCO 주가변동을 코스피 변동, 원/달러 변동, 유가 변동으로만 설명하고 있다.

이 관계식에 만족할 수도 있고, 조금 더 간결한 매크로 영향 구조를 보고 싶을 수도 있다. 3개의 매크로 변수가 아닌 2개 정도의 매크로 변수로 설명하기 위해, 람다 기준을 0.0005로 강화한 LASSO 회귀 분석을 다시 실시해 보자.

LASSO 회귀 결과(람다=0.0005): 회귀계수 해

β_0	β_1	β_2	β_3	β_4
0.00	0.69	0.00	0.00	0.05

회귀계수의 절대값 합이 작아지도록 더 강하게 유도한 결과, 원/달러 월수익률과 원/엔 월수익률 변수의 회귀계수가(β_2와 β_3) 모두 0으로 변경되었다. 이 결과에 따른 더 간결해진 회귀분석식은 다음과 같다.

LASSO 두 번째:

$$posco\_m = 0.00 + 0.69kospi\_m + 0.05wti\_m + u$$

이 식을 보면, 더 간결한 두 번째 LASSO 모델에서는 코스피 지수가 1% 상승하면 POSCO 주가는 0.69% 상승하고, WTI가 1% 상승하면 POSCO 주가는 0.05% 상승하는 관계임을 볼 수 있다.

앞서 본 최초의 OLS 모델은, POSCO 수익률에 코스피 변화율이 0.89배, 원/달러 변화율이 -0.37배의 영향을 준다고 해석했다. 두 변수 중에서는 코스피 변화율 변수의 영향력이 더 큰 편이었고 두 변수의 상호 작용이 있는 걸 감안하면, 이 부분을 코스피 변화율의 영향으로만 정리하는 것도 의미가 있어 보인다. 더 간결한 모델을 만든 결과를 보면(LASSO 두 번째), 환율 변수가 빠지고, 코스피 변화율에의 민감도가 0.69로 바뀌게 되었다. 환율 변수를 빼면서 이에 해당하는 효과 중 일부는 코스피 변화율에, 일부는 WTI 변화율에 반영되어 각각 회귀계수 수치가 바뀌었다고 볼 수 있다.

이 상으로 본 LASSO 뿐 아니라 Ridge, Elastic Net 기법 모두가 실용적인 모델 수립에 도움을 줄 수 있다. 이런 기법들은 전통적인 회귀분석 튜닝 방법인 forward selection, backward elimination 방식보다 훨씬 편리하고 효율적으로 회귀 계수 축소 기능을 할 수 있다.

III. 사용법

1. 파이썬(Python) 사용법

엑셀 기본 프로그램에서는 Ridge나 LASSO 같은 고급 모델을 제공하지 않는다. 대신, 파이썬에서는 이들 모델을 손쉽게 사용할 수 있다.

파이썬의 대표적인 머신러닝 패키지인 sklearn에서 다양한 벌점 회귀분석 모델들을 제공한다. Ridge, LASSO, Elastic Net 모델은 각각 sklearn 하의 각각 Ridge, Lasso, ElasticNet 클래스와 대응된다.

Sklearn의 대부분의 계량모델은, *fit* 메서드로 모델 분석(혹은 학습)을 하며 *coef_* 속성에 모델 계수들을 저장한다.

```
# -*- coding: utf-8 -*-

#모듈 가져오기 # 파이썬에서는 샵(#)기호로 주석을 작성
import pandas as pd          # pandas를 가져와 pd로 사용
from sklearn.datasets import load_boston    #load_boston 가져오기
from sklearn.linear_model import LinearRegression #기본 회귀분석 가져오기
from sklearn.linear_model import Ridge, Lasso, ElasticNet #3가지 벌점 회귀


#보스턴 주택가격(파이썬 내재 데이터) 샘플 가져오기
# 범죄율, 방개수 등 13개 독립변수들의 샘플데이터가 X에, 주택가격이 종목변수이며 y에 들어감
X, y = load_boston(return_X_y=True)


# 다중회귀분석(OLS)
lr = LinearRegression() #다중회귀분석 모델을 하나 생성
lr.fit(X, y) #X(506x13 사이즈)와 y(506x1 사이즈)로 회귀분석 실시
print('regression coef:\n ', lr.coef_) #13개 독립변수의 각 회귀계수 출력
print('\nregression intercept:\n', lr.intercept_) #절편값 출력
```

```
regression coef:
  [-1.08011358e-01  4.64204584e-02  2.05586264e-02  2.68673382e+00
 -1.77666112e+01  3.80986521e+00  6.92224640e-04 -1.47556685e+00
  3.06049479e-01 -1.23345939e-02 -9.52747232e-01  9.31168327e-03
 -5.24758378e-01]

regression intercept:
 36.459488385089855
```

```
# Elastic Net 회귀
elastic_net = ElasticNet() #Elastic Net 모델을 하나 생성(옵션은 디폴트 값으로)
elastic_net.fit(X, y) #X와 y로 분석 실시
print('Elastic_Net coef:\n ', elastic_net.coef_) #13개의 회귀계수 출력
print('\nElastic_Net intercept:\n', elastic_net.intercept_) #절편값 출력
```

```
Elastic_Net coef:
  [-0.08037077  0.05323951 -0.0126571   0.
  -0.          0.93393555  0.0205792  -0.76204391
   0.30156906 -0.01643916 -0.7480458   0.00833878
 -0.75842612]

Elastic_Net intercept:
 42.22956397215435
```

```
# Elastic Net 회귀 예측 (임의의 X_test 변수 준비)
elastic_net.predict(X_test) #변수 X_test에 대한 Elastic Net 모델의 예측값 계산
```

모델 사용법을 위한 예제로, 보스턴 지역의 주택가격과 이를 설명해 줄 수 있는 다양한 관련 변수들 데이터를 샘플로 삼았다.

먼저 다중회귀분석(Linear Regression)을 실행한 결과를 보면, 13개 독립변수 들에 대해서 모두 회귀계수 값이 결정되는 형태로 나왔다.

(위의 처음 출력화면 부분에서 "-1.08011358e-01 4.64204584e-02 …" 정보)

그 다음으로는 Ridge와 LASSO를 결합한 Elastic Net 모델을 수립했다. 모델의 하이퍼 파라미터는 클래스 선언 시에 입력하게 되는데, 여기서는 디폴트 수치가 들어가도록 선언했다(elastic_net = ElasticNet()).

fit 메서드를 통해서 모델의 학습이 진행된다. 즉, *fit*을 실행하면 회귀모델 상의 회귀계수 해가 결정된다.

예제에서 *coef_* 속성을 통해 모델의 학습 결과를 보면, 4번째와 5번째 변수의 회귀계수가 0으로 바뀐 것을 볼 수 있다. (Elastic Net 출력화면 부분에서 "-0.08037077 0.05323951 -0.0126571 0. -0. …" 정보)

즉, Elastic Net 모델의 회귀식에서는 2개 변수가 줄어든 11개 변수 기준으로 회귀계수 축소가 이뤄졌음을 알 수 있다. (실제로 제외된 4번째와 5번째 변수는 '강 인접 여부 더미'와 '질소산화물 농도'였음)

sklearn에서는 학습된 모델을 가지고 예측할 때, 보통 *predict* 메서드를 사용하게 된다. 여기서도 *predict* 메서드를 통해, 새로운 X_test 데이터를 기반으로 각각의 y 값에 대해 예측을 실행할 수 있다.

IV. Appendix

그리스 문자 표

대문자	소문자	명칭	
Α	α	Alpha	알파
Β	β	Beta	베타
Γ	γ	Gamma	감마
Δ	δ	Delta	델타
Ε	ϵ	Epsilon	엡실론
Ζ	ζ	Zeta	제타
Η	η	Eta	에타
Θ	θ	Theta	쎄타
Ι	ι	Iota	이오타
Κ	κ	Kappa	카파
Λ	λ	Lambda	람다
Μ	μ	Mu	뮤
Ν	ν	Nu	뉴
Ξ	ξ	Xi	크사이
Ο	ο	Omicron	오미크론
Π	π	Pi	파이
Ρ	ρ	Rho	로
Σ	σ	Sigma	시그마
Τ	τ	Tau	타우
Υ	υ	Upsilon	입실론
Φ	ϕ	Phi	파이
Χ	χ	Chi	카이
Ψ	ψ	Psi	프사이
Ω	ω	Omega	오메가

신뢰에 가치로 답하다

삼성증권 SAMSUNG

삼성증권주식회사
서울특별시 서초구 서초대로74길 11(삼성전자빌딩)
Tel: 02 2020 8000 / www.samsungpop.com
삼성증권 지점 대표번호: 1588 2323 / 1544 1544
고객 불편사항 접수: 080 911 0900