DEVOCEAN

다가오는 Automated AI 시대, 그 기반기술과 적용사례

- 부제 : Data Science 현업에서 바라보는 AutoML, 아군인가? 침략자인가?

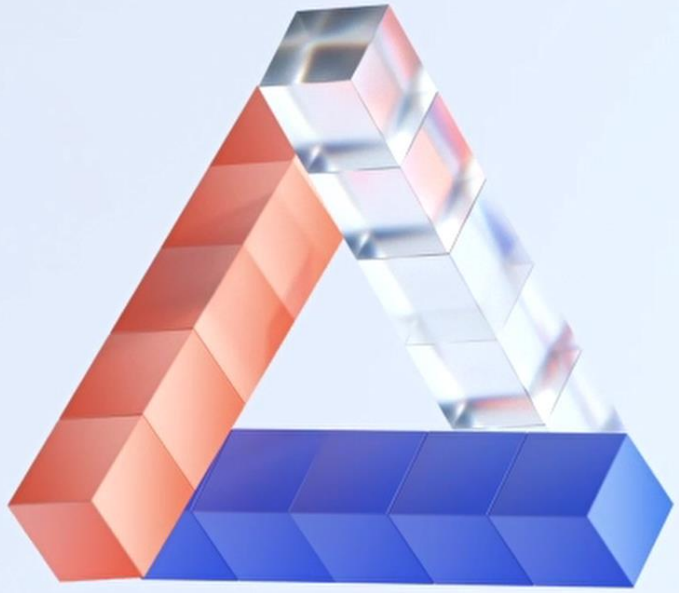
Byungsun, Park

SK주식회사 C&C, Senior Data Scientist

parkbs@sk.com

Byungsun.park@outlook.com

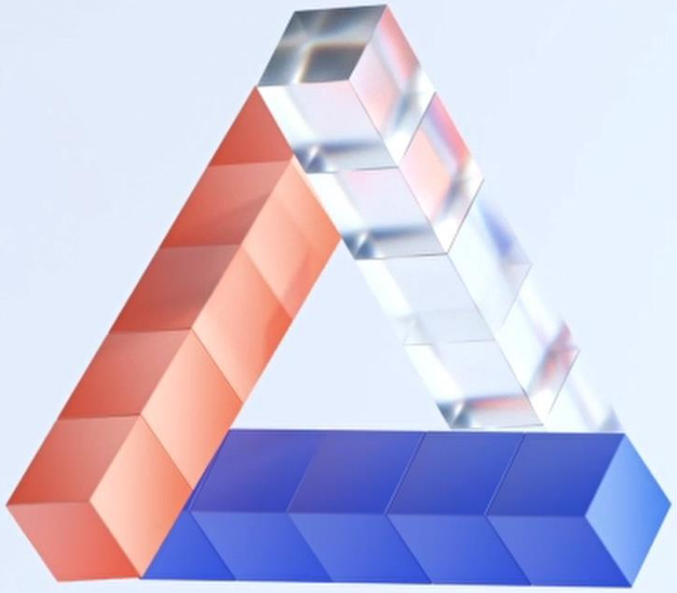
DEVOCEAN



Byung-sun, Park

- 現) SK주식회사 C&C, Automated AI팀, Senior Data Scientist
- 前) Fin-Tech Start-up Founder
- 前) 한국신용평가정보, 전략컨설팅 사업실
- 금융/유통 분야 Data Modeling에 다양한 경험과 전문성 보유
- 다양한 분야에 적용할 수 있는 추천 모델링에 관심이 많습니다.

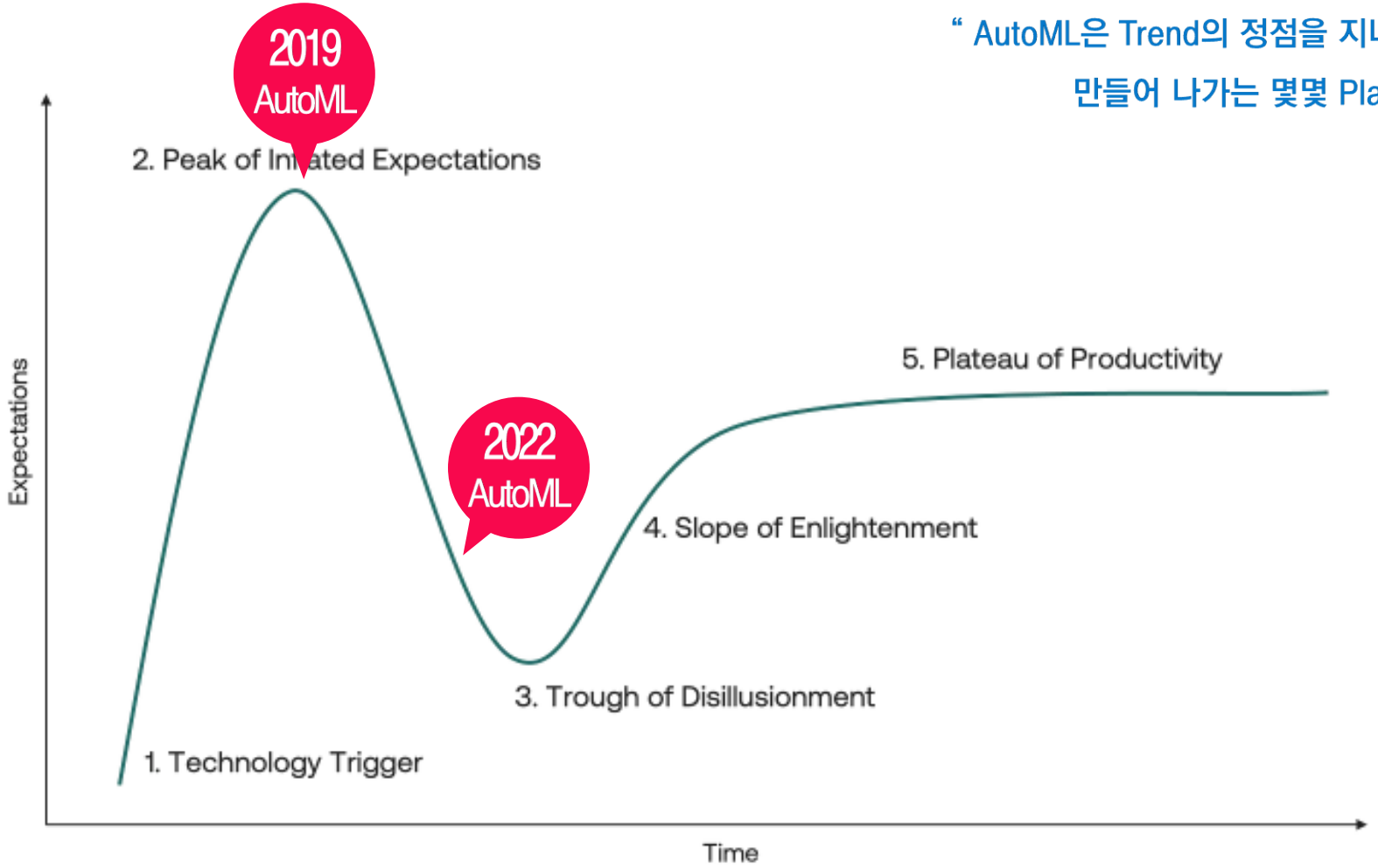
DEVOCEAN



1. Intro
2. Hyper-parameter Optimization
3. XAI (Permutation, Surrogate Model, LIME, SHAP, PDP)
4. Use-case, Etc.

Journey of AutoML Technique

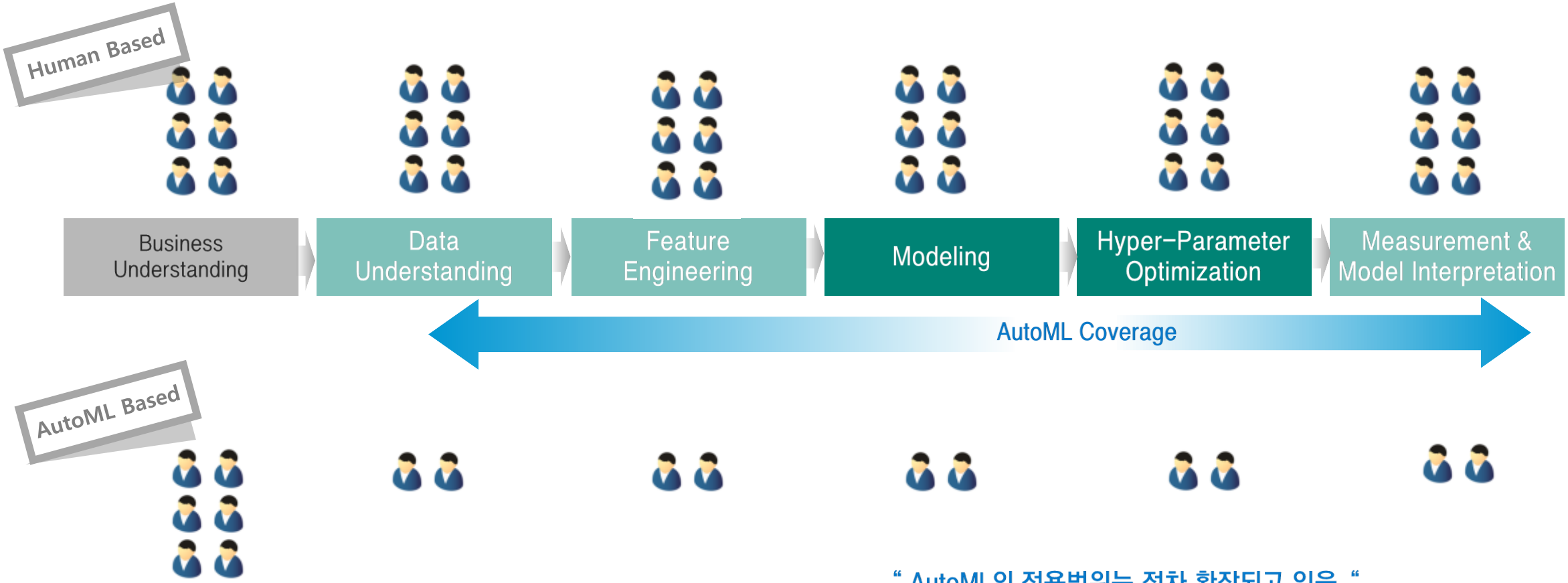
AutoML in Hype Cycle



" AutoML은 Trend의 정점을 지나 실제 성공적인 Use-case를
만들어 나가는 몇몇 Player 들을 중심으로 시장이 재편중 "

Coverage of AutoML Technique

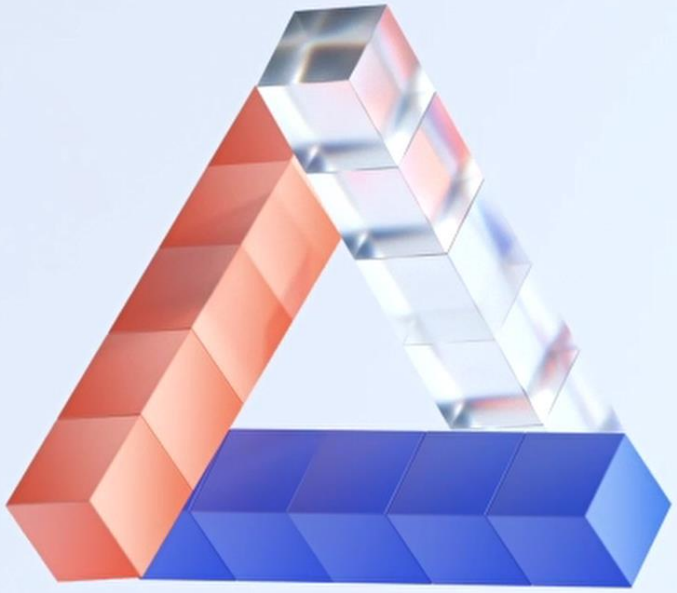
AutoML in Data Analysis Process



" AutoML의 적용범위는 점차 확장되고 있음 "

" Domain Expert, CDS도 스스로 분석이 가능한 방향으로 진화"

DEVOCEAN



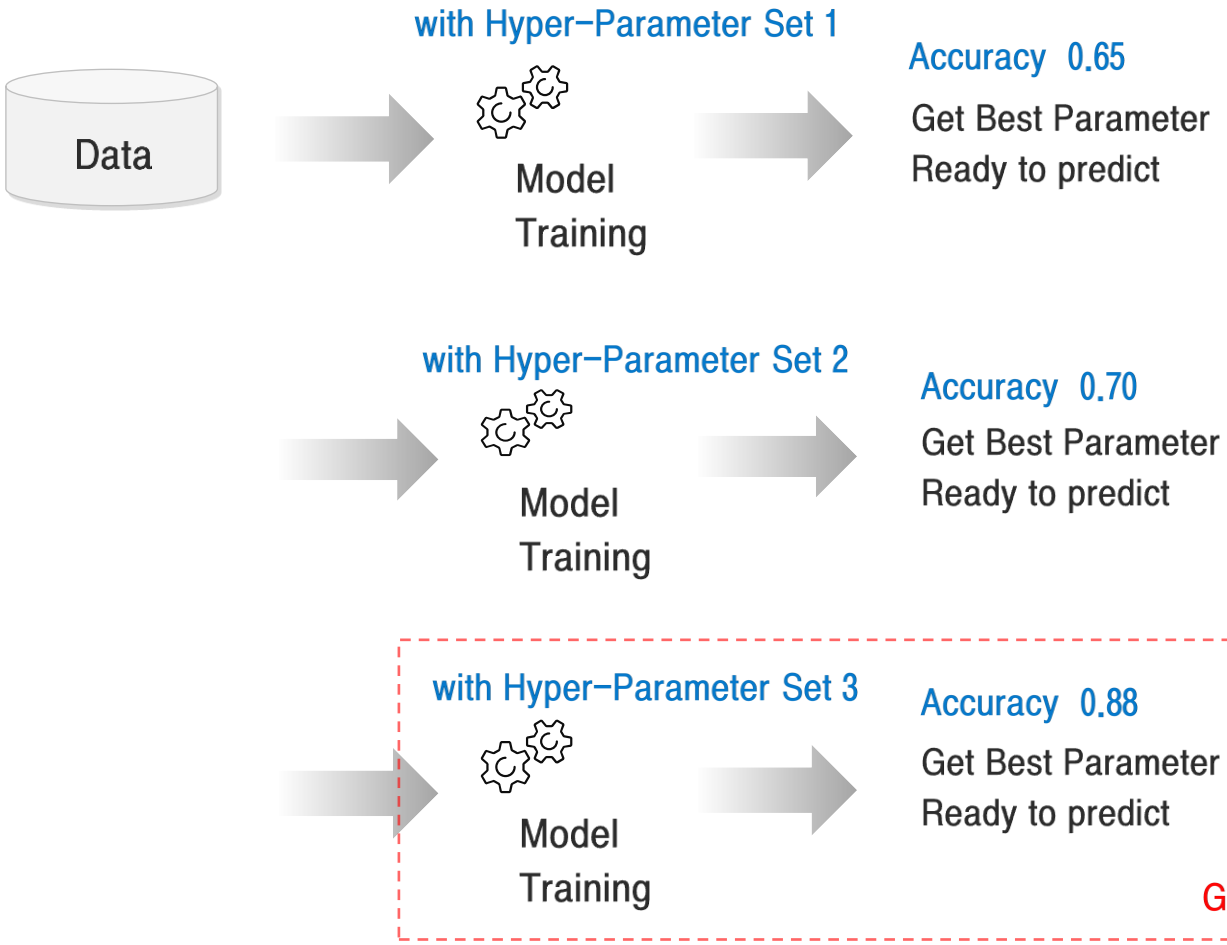
1. Intro

2. Hyper-parameter Optimization

3. XAI (Permutation, Surrogate Model, LIME, SHAP, PDP)

4. Use-case, Etc.

Parameter vs Hyper-parameter



Parameter

- 모델 내부에서 주어진 데이터로부터 결정

  Regression의 회귀계수(coefficient)

  D/L의 가중치(weight) ..
- 작업자에 의해 조정되지 않음

Hyper-parameter

- 모델 외부에서 작업자가 직접 세팅해 주는 값

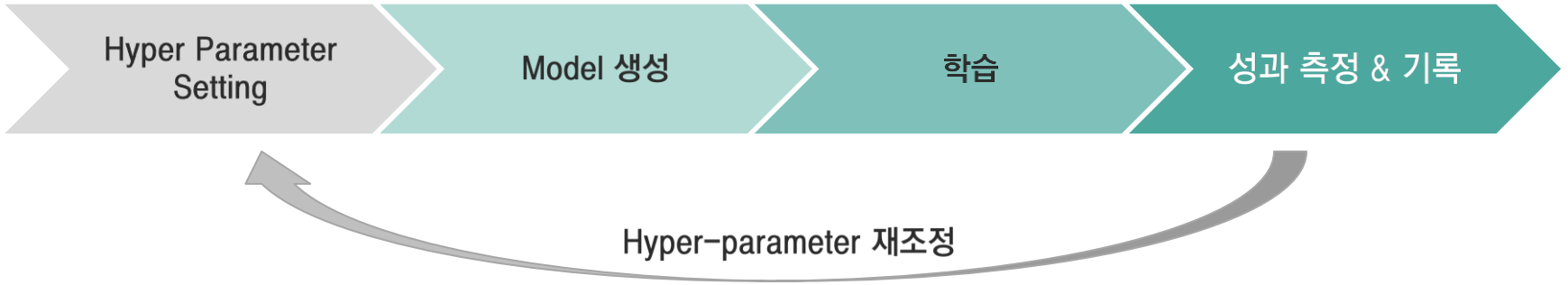
  K-NN의 k값

  D/L 의 Learning rate

  Tree Model의 max depth..
- 학습 데이터에 의해 변경되지 않음
- Heuristics 혹은 반복된 실험을 통해 결정

Hyper-parameter Optimization

HPO process by human



Problem

- HPO는 최적의 모델을 얻기 위해 Best Hyper-parameter를 역으로 찾는 과정
- Hyper-parameter 재조정에 따른 Model 재학습 등 **반복적인 탐색과정으로 많은 시간과 비용이 소요되는 절차**
- 데이터 양이 많아질수록, **모델의 복잡도가 증가할수록 연산비용은 급격히 증가**
- 대용량 학습 데이터를 사용하는 거대모델의 등장, 학습을 결정하는 Hyper-parameter수의 늘어나는 경향

Grid Search

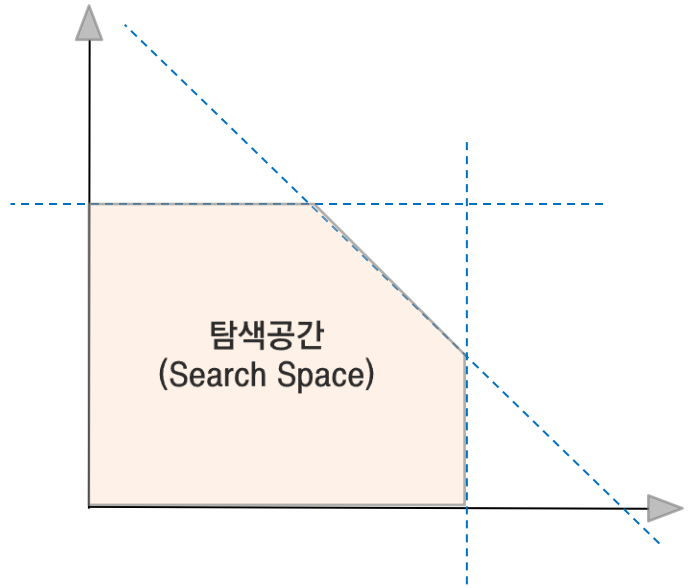
Concept

Hyper-parameter의 탐색 범위를 제한한 후 **모든 조합을 순차적으로 비교 탐색**

- Min/Max, 간격을 통해 탐색범위를 제한
- 탐색할 Hyper-parameter 조합들을 Grid에 보관
- Grid의 범위와 하이퍼 파마메타수가 늘어날수록 연산량이 기하급수적으로 증가

[Hyper-parameter 탐색범위 제한 예시]

- Parameter A : { 'Entropy' , 'Gini' }
- Parameter B : Range(6,20,2) → {6, 8, 10, 12, 14, 16, 18, 20}
- ..



Hyper parameter A 탐색값 10개 × Hyper parameter B 탐색값 10개 × Hyper parameter C 탐색값 10개 = 1,000 set

탐색할 Hyper parameter set

순차적으로 학습후 성과비교

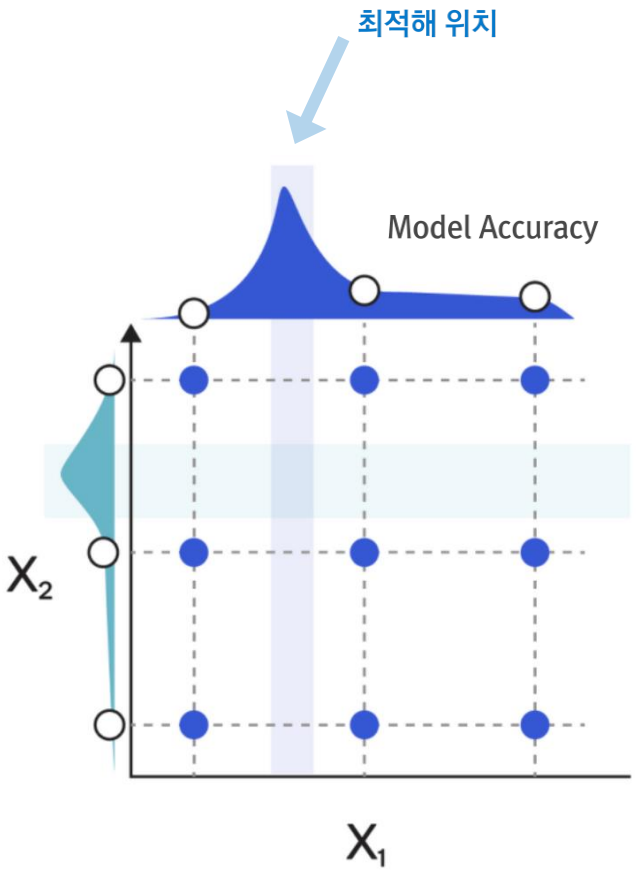
최적의 Set 탐색

Grid Search

Pros & Cons

- 구성이 쉽고 직관적인 방법
- 작업자가 원하는 탐색 범위를 정확하게 비교 분석 할 수 있음
  → 작업자가 숙련된 Data Scientist여서 적절한 탐색범위를 제한할 수 있다면 효율적

- **느리고 느리고 느리다!**
- 균일한 간격으로 탐색하기 때문에 최적의 해를 지나칠 수 있음
- 모든 구간을 전부 탐색하기에 비효율적



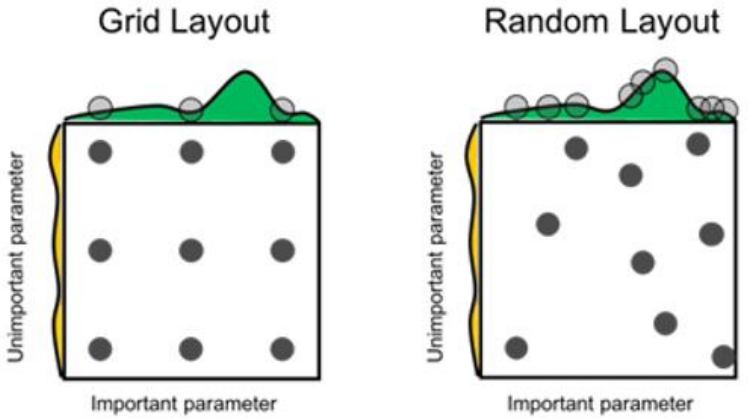
[Grid Search]

Random Search

Concept

Hyper-parameter의 탐색 범위를 제한한 후 **설정한 횟수만큼 임의의 조합을 탐색**

- 균일 간격 탐색이 아니라 Random 탐색을 통해 최적해를 발견할 가능성을 높임
- 탐색할 임의의(Random) 조합 수를 직접 설정하기에 불필요한 반복 탐색 감소



[Grid Search vs Random Search]

Pros & Cons

- 탐색범위가 넓을 경우 일반화된 결과가 나오지 않음 (할 때마다 결과 상이)
- 성과가 설정하는 탐색 횟수에 의존적일 수 있음

“ 성능이 가장 높은 최적해로 점차 찾아가는 기법이 아니라
단순히 탐색 시행을 통해 비교 분석해서 최적해를 찾아가는 기법 “

Bayesian Optimization

Bayesian Inference ?

내가 알고자 하는 값은 동일 실험을 수 만번 시행해서 가장 많이 나온 값일 것.. 하지만..

지금 알고 있는 정보(이전의 경험)와 현재의 증거(관측값)로 내가 알고자 하는 값을 점차 추론 해(업데이트) 나가는 것

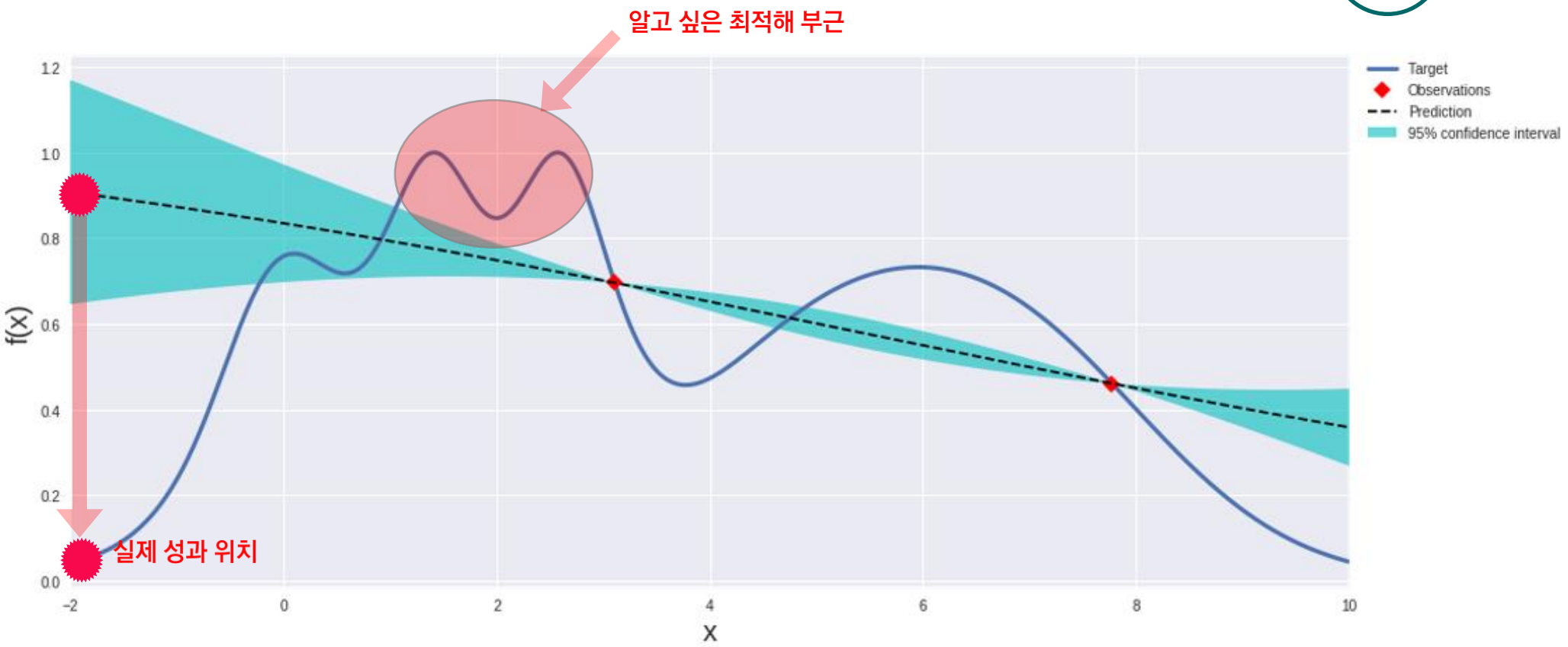
Bayesian Optimization

Concept

Hyper-parameter set와 성과 pair

이전 시행의 정보로부터 **최적의 성과를 낼 수 있을 것 같은** Hyper-parameter를 추론해 보자.

After 2 Steps



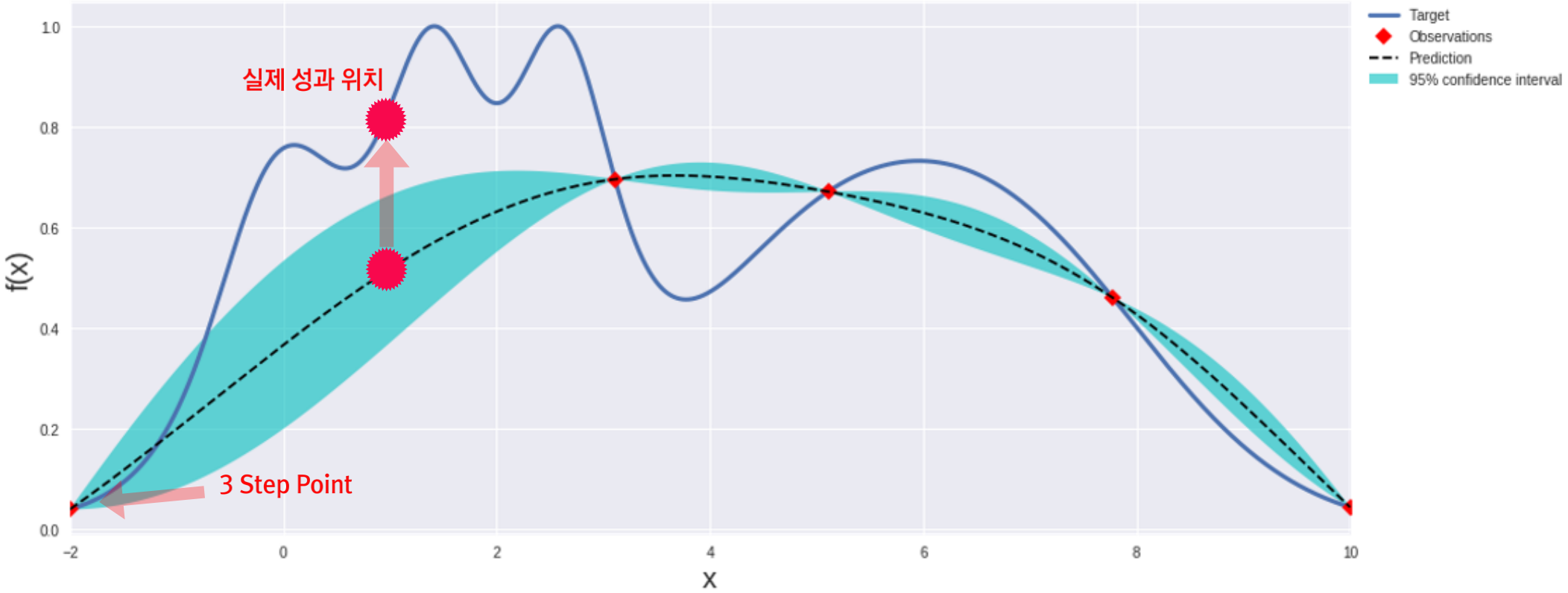
Bayesian Optimization

Concept

Hyper-parameter set와 성과 pair

이전 시행의 정보로부터 **최적의 성과를 낼 수 있을 것 같은** Hyper-parameter를 추론해 보자.

After 5 Steps



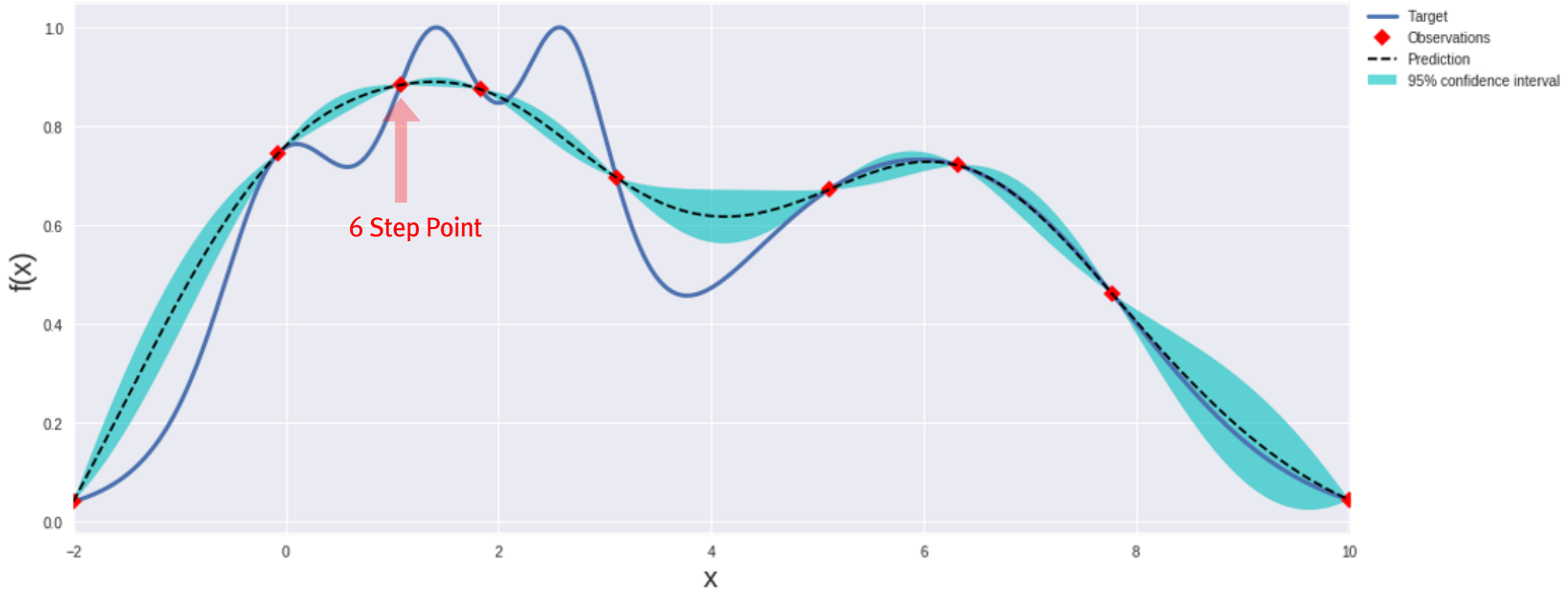
Bayesian Optimization

Concept

Hyper-parameter set와 성과 pair

이전 시행의 정보로부터 **최적의 성과를 낼 수 있을 것 같은** Hyper-parameter를 추론해 보자.

After 7 Steps



Bayesian Optimization

Pros & Cons

- 단순한 Search Method비해 보다 적은 시도로 최적의 결과를 낼 수 있는 이론적 근거를 가지고 있음
  → 높은 성능의 Hyper-parameter를 찾을 수 있는 가능성 향상

- 최적화 탐색의 **속도 문제**
- 기존의 탐색결과를 활용하는 **Sequential approach** 이기에 **속도향상의 한계 & 병렬처리 불가**
- 활용(Exploitation)와 탐색(Exploration)의 적절한 Trade-off

Successive Halving Algorithm(SHA)

Concept

Hyper-parameter조합의 모델 성능을 평가할 때, 꼭 끝까지 수행해야 할까?

적절한 중간 학습 과정에서 싹수가 없는 것은 Early stopping하자.

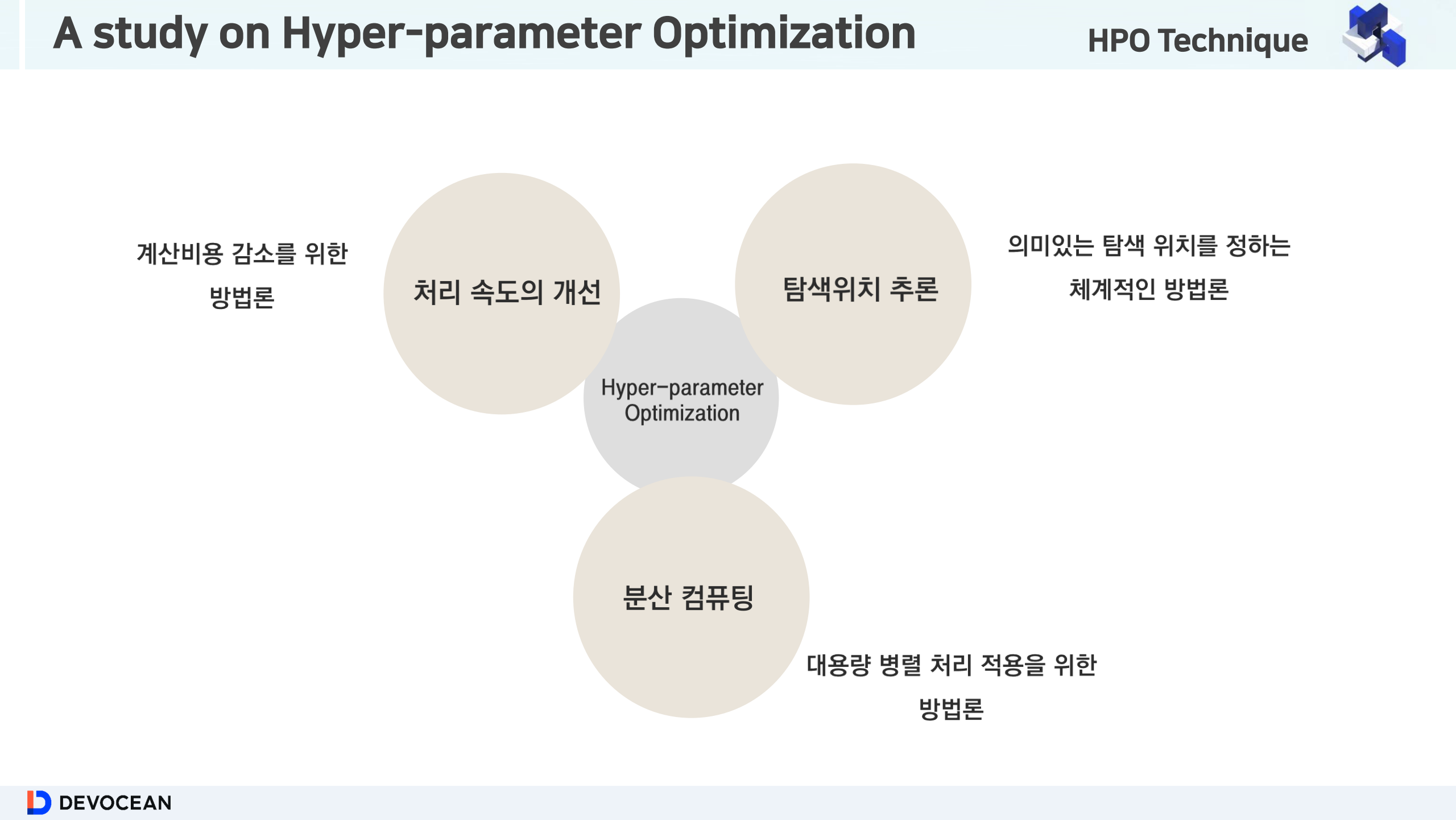
절반으로 줄이기(Halving),

"연속적으로 개별 모델에 대한 평가를 시행하여 절반은 남기고 나머지 절반은 버리자"

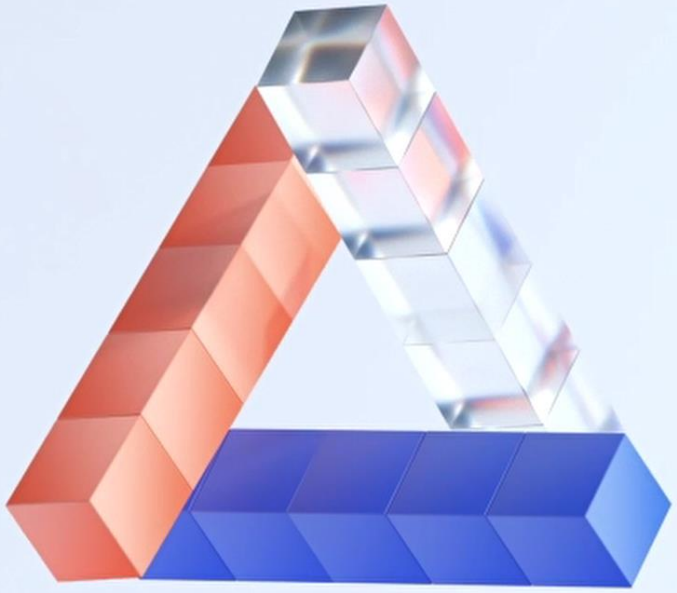
Ex) 1개 모델의 평가에 12시간이 소요되는데 어떤 중간 평가과정을 통해 2시간만에 해당 모델을 계속 가져갈지 버릴지 판단할 수 있다면?

→ 6배 속도 향상

A study on Hyper-parameter Optimization



DEVOCEAN



1. Intro
2. Hyper-parameter Optimization
3. **XAI** (Permutation, Surrogate Model, LIME, SHAP, PDP)
4. Use-case, Etc.

Why XAI?

모델을 해석한다는 것의 의미

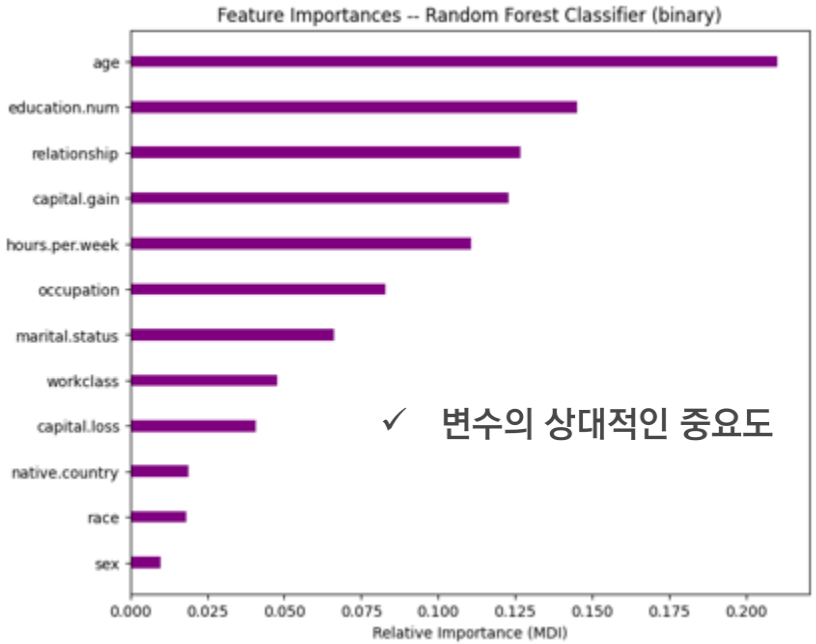
- Modeling을 통해서 해결해야하는 많은 문제, **높은 예측력만 보이면 되는 문제도 있지만..**
- **현실의 많은 문제는 Why 에 관심**

OLS Regression Results

Dep. Variable:	MEDV	R-squared:	0.727
Model:	OLS	Adj. R-squared:	0.722
Method:	Least Squares	F-statistic:	132.0
Date:	Wed, 26 Apr 2023	Prob (F-statistic):	1.01e-132
Time:	18:05:24	Log-Likelihood:	-1511.5
No. Observations:	506	AIC:	3045.
Df Residuals:	495	BIC:	3092.
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	40.0360	4.961	8.070	0.000	30.289	49.783
CRIM	-0.1185	0.033	-3.555	0.000	-0.184	-0.053
ZN	0.0368	0.014	2.692	0.007	0.010	0.064
CHAS	3.1323	0.872	3.593	0.000	1.419	4.845
NOX	-21.5397	3.657	-5.890	0.000	-28.724	-14.355
RM	3.8374	0.420	9.137	0.000	3.012	4.662
AGE	0.0021	0.013	0.157	0.875	-0.024	0.029
DIS	-1.4413	0.199	-7.257	0.000	-1.832	-1.051
RAD	0.1052	0.041	2.570	0.010	0.025	0.186
PTRATIO	-1.0035	0.131	-7.654	0.000	-1.261	-0.746
LSTAT	-0.5562	0.051	-10.899	0.000	-0.656	-0.456

- ✓ 영향력의 수준
- ✓ 영향력의 기여도
- ✓ 영향력의 방향
- ✓ 변수의 유의성



✓ 변수의 상대적인 중요도

Easy Way?

Problem

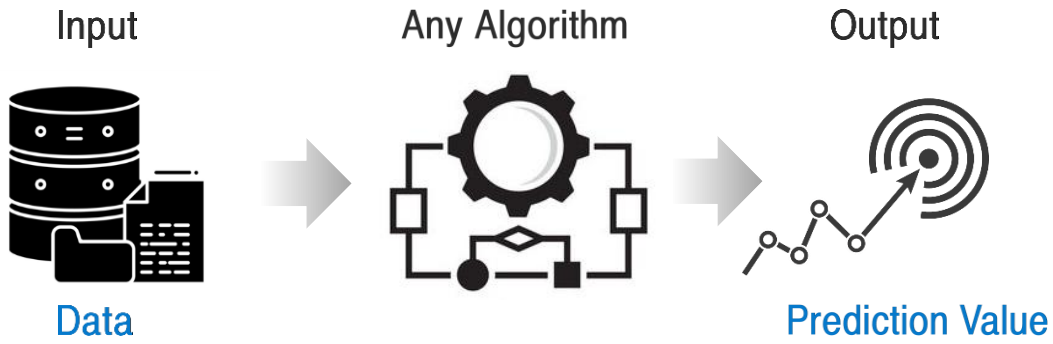
- 복잡도가 증가하는 모형들.. Black-Box Model은 어떻게 해석해야 할까?
- 다양한 알고리즘을 동일한 기준으로 해석해 볼 수는 없을까?

Easy Way

- 각각의 개별변수만 남기던가 빼고 학습해서 비교해보면 어떨까?
- 좋은 생각인가?

  (ex: Feature가 10개라면 10개의 개별모델을 각각 학습하여 성과 비교)

" Hint는 어떠한 모형이더라도 가시적으로 주어지는 Input과 Output "



"Model-agnostic"

Permutation Importance

Key Idea

특정 Feature 값들을 **임의로 섞으면** 모형성과는 **얼마나 나빠질까?**

- 임의의 값을 어떻게 부여할까?

  → 생소한 Random값보다 실제 있을 법한 기존 값들을 재배치 하자

- 모형 성과는 어떻게 측정하고 비교할까?

  → 재학습하지 말고 기존의 학습된 모형에 Original set vs Permutataion set을 예측하여 오차를 비교하자

Permutation Importance

Concept for Calculation

X2의 Permutation Importance를 구해보자

Original Data Set

curb_weight (X1)	cylinder_num (X2)	num_of_doors (X3)	…	city_mpg (x10)	car_price (Y)
2765	4	2	…	16	21105
3055	4	4	…	17	24565
…	…	…	…	…	…
3230	6	4	…	22	36880
1874	3	4	…	29	6575

Pre-trained Model

“X2 Permutation Set Error”와
“Original Set Error”의 차이값

X2의 Feature Importance

…

curb_weight (X1)	cylinder_num (X2_permuting)	num_of_doors (X3)	…	city_mpg (x10)	car_price (Y)
2765	3	2	…	16	21105
3055	6	4	…	17	24565
…	…	…	…	…	…
3230	4	4	…	22	36880
1874	4	4	…	29	6575

Permutation Data Set (X1..X10)

…

Permutation Importance

어떤 DataSet을 활용할까?

Train Data Set

- 모델은 학습과정을 통해서 어떤 Feature가 중요한지 배움
- 학습(Train) 데이터의 경향을 가져가는 것이 좋다

VS

Test Data Set

- Train 데이터를 활용할 경우 오차에 대한 측정이 낙관적
- 모형이 과적합 되었다면 FI(Feature Importance) 산출까지 따라옴
- Train 데이터로 계산된 FI는 신뢰도가 낮을 수 밖에 없음

Permutation FI가 (-) 의 값을 보이는 Feature는 어떤 의미일까?

Permutation Importance

Pros

- 모델을 재학습할 필요가 없기 때문에 계산 비용의 장점
- 비교가 용이한 직관적인 Feature 중요도를 얻음
- Feature간의 상호작용이 자동적으로 고려됨

Cons

- 영향의 방향은 알 수 없음
- Outlier가 있다면 편향될 수 있음
- 알고리즘 실행시마다 중요도는 다르게 계산될 수 있음 (why? Permutation 정도, 에러에 기반한 추정한계)
- 상관관계가 높은 Feature가 포함되어 있다면 해당 Feature의 중요도가 왜곡될 수 있음

Surrogate Model

Key Idea

현재 모델(Original Model)의 예측력을 활용하면서,

가볍고 **해석가능한 다른 모델(대리모델)의 설명력을 활용**할 순 없을까?

- 해석이 어려운 **블랙박스 모델의 예측에 근사하도록 훈련된** 해석가능한 **가볍고 빠른 대리 모델을 활용**함으로써 블랙박스 모델에 대한 설명력을 제공
- Approximation model, metamodel, response surface model로 불리기도 함

Good Surrogate Model ?

(1) 가능한 블랙박스 모델(Original Model)의 예측력에 근사

(2) 동시에 해석성을 제공하는 모델

(3) 가볍고 빠른 연산비용이 적은 모델

Surrogate Model

Concept for Calculation

Original Model (Black-box Model)

Original Model로부터 **예측값 활용**

데이터 준비 → 대리(Surrogate)모델 선정 → 학습한 대리모델의 설명력 활용

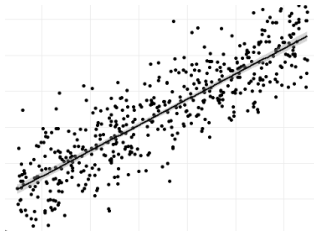
Original Data

(X1)	(X2)	(X3)	...	(x10)	(Y)
2765	4	2	...	16	21105
3055	4	4	...	17	24590
...	...	...	...	...	...
3230	6	4	...	22	36880
1874	3	4	...	29	6575

Or

Sub Data-set

(X1)	(X2)	(X3)	...	(x10)	(Y)
2765	4	2	...	16	21105
...	...	...	...	...	...
1874	3	4	...	29	6575

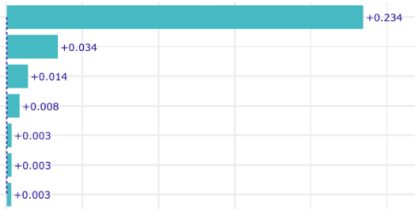


$$g(x) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

$$g(x) = \sum_{m=1}^{M} c_m I\{x \in R_m\}$$

Learning

Surrogate Model로 부터의 **설명력 활용**

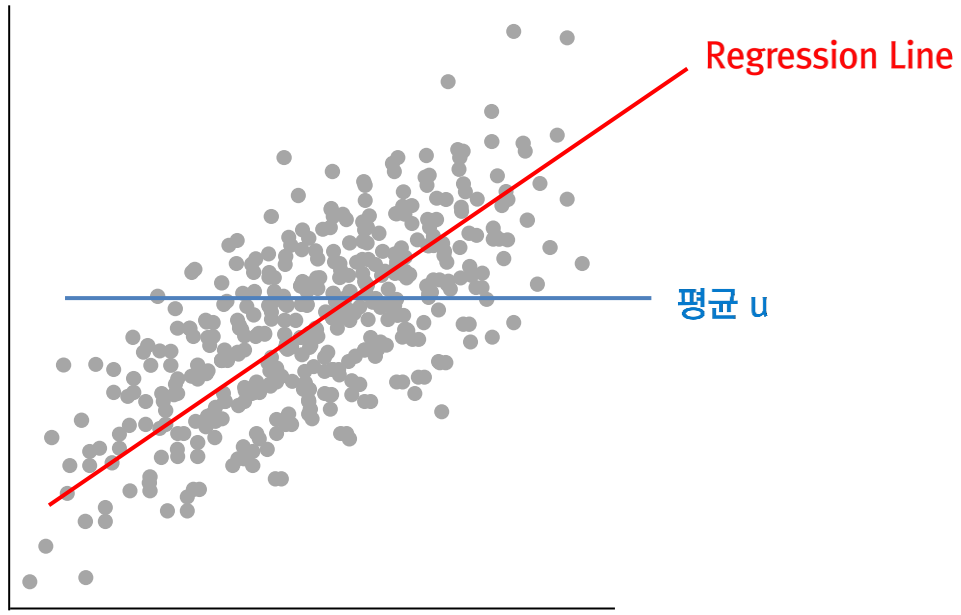


"대리모델은 Black-Box 모델에
근접한 성능(예측력)을 가질까? "

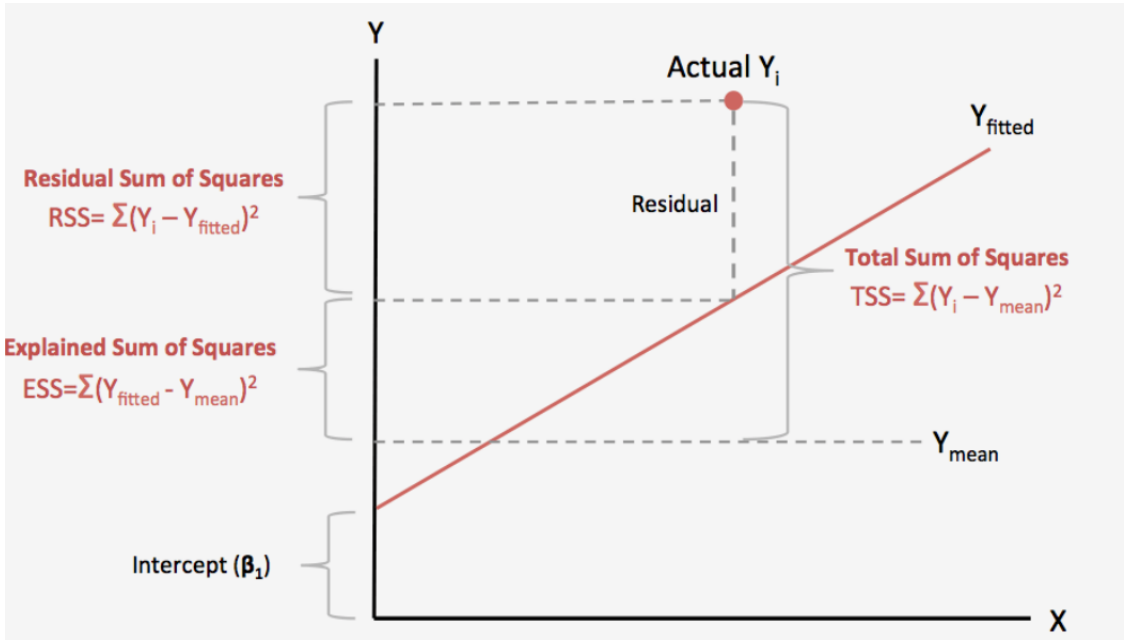
Surrogate Model

대리모델(Surrogate)이 Original Model의 성능에 근사한지는 어떻게 측정할까?

잠깐 선형 모형(Linear Model)을 설명력을 살펴보자 !



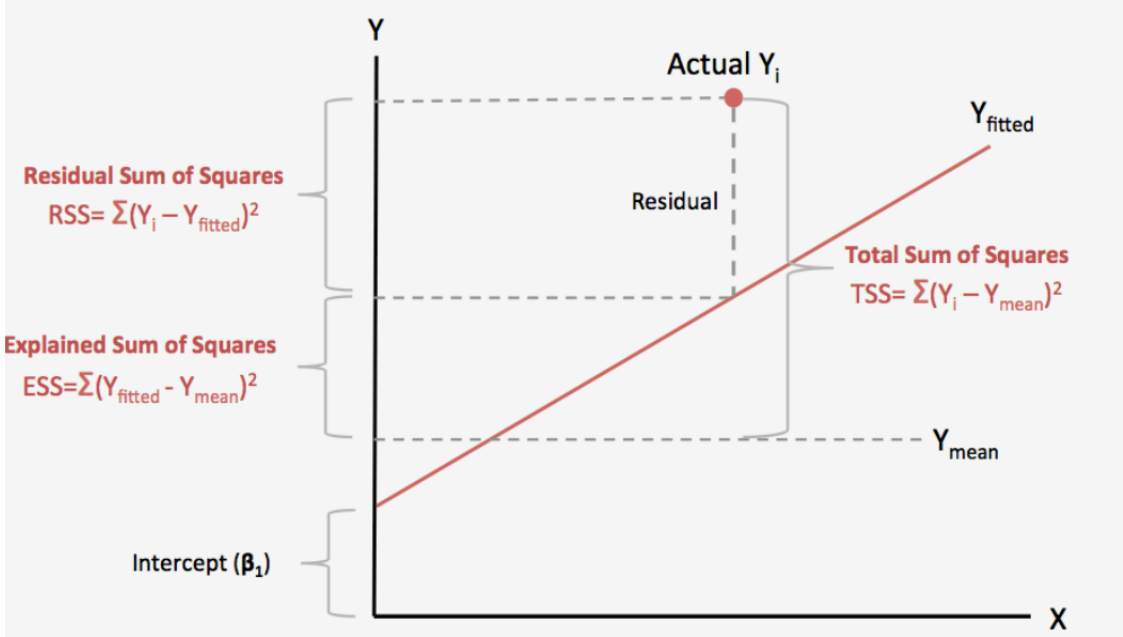
R Squared(Coefficient of Determination) = 1 – RSS/TSS



Surrogate Model

대리모델(Surrogate)이 Original Model의 성능에 근사한지는 어떻게 측정할까?

R Squared = 1 – RSS / TSS



Y hat from Surrogate Model

$$R^2 = 1 - \frac{\text{RSS}}{\text{TSS}} = 1 - \frac{\sum_{i=1}^{n}(\hat{y}_*^{(i)} - \hat{y}^{(i)})^2}{\sum_{i=1}^{n}(\hat{y}^{(i)} - \bar{\hat{y}})^2}$$

- R^2가 1에 가깝다는 의미는 대리모델의 성능이 Original Model의 성능에 매우 근사 한다는 것
- R^2이 0에 가깝다는 의미는 대리모델이 Original Model을 설명하지 못한다는 것

Surrogate Model

Pros

- 해석이 가능하면 **어떤 모델이든지 사용할 수 있는 유연함**
- 사용자와의 접근성에 따라 대리 모델을 선택할 수 있고, Multi-Surrogate 모델 구성 등 **설명력의 제공 방식이 열려 있음**

  Ex) D/L계열의 Original 모델을 통해 예측력 제공 + 선형/Tree 2가지 대리모델을 통해 다각도 설명력을 제시

  Ex) 평소 회귀분석 결과로 Comm.을 하던 업무방식 → 선형 대리모델을 통해 설명력을 제공

Cons

- 대리모형과 Original **모델의 근사함을 판단하는 기준이 명확치 않음**.. R2 0.5? 0.8? 0.99?
- 어떤 Sample에서는 대리모델과 Original 모델과 근사하지만, 어떤 Sample에서는 크게 다를 수 있음
- **해석력은 모든 데이터 포인트에 대해 동일하지 않음**

Think

Case 1)

- 블랙박스모델로 SVM 대리모델로 CART 의사결정트리를 활용하는데, 대리 모델의 R^2 는 0.75, 이는 블랙 박스 모델의 성능에 상당히 가깝지만 완벽하지는 않다는 것을 의미함. 만약 R^2 가 0.94라면?

Case 2)

- 블랙박스모델은 랜덤포레스트, 대리모델은 CART 의사결정트리를 활용하는데, 대리 모델의 R^2 는 0.14. 대리 모델의 해석력을 어떻게 봐야 할까?

Problem

Situation)

대출이 거부된 고객A씨,

은행의 대출 심사역은 고객A씨에게 대출이 거부된 이유에 대해 설명해 주어야 하는데..

고객A씨의 상황은 일반적인 고객의 경향과 다름.

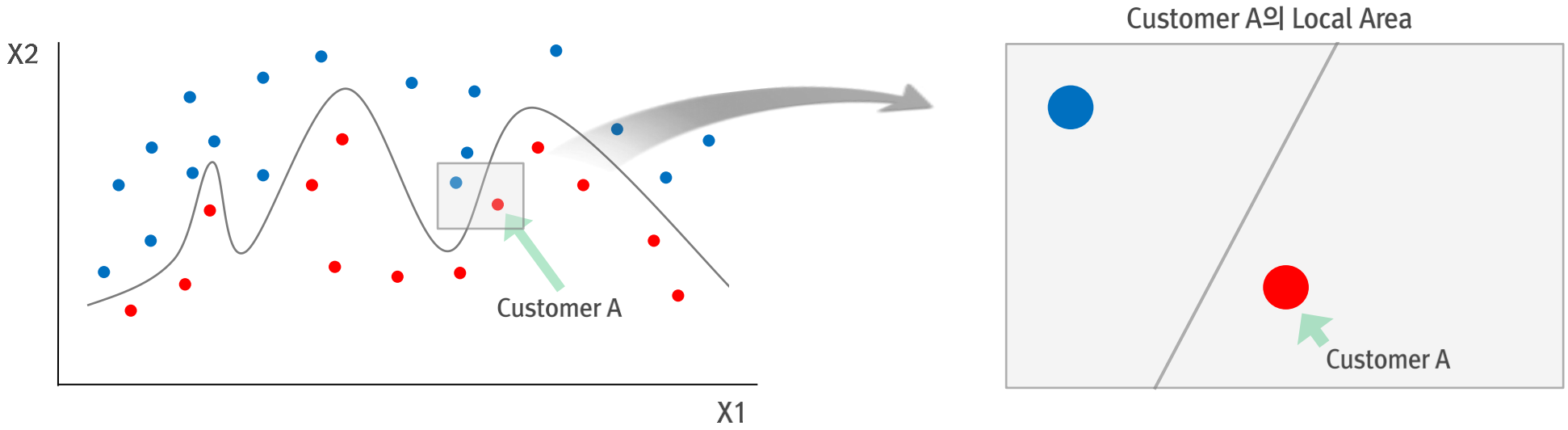
전체 데이터의 Feature Importance로 고객A의 대출 반려 요인을 잘 설명할 수 있을까?

LIME (Local Interpretable Model-agnostic Explanation)

Key Idea

내가 **설명하고 싶은 관측치에 초점을 맞추어서** 해석 가능한 **대리(Surrogate) 모델을 활용**하자

- 해석의 초점을 바꾸어서 **해당 관측치의 개별 예측을 보다 잘 설명할 수 있음**
- **Local Surrogate Model**
- **비선형적인 패턴을 학습한 모형이더라도 국소적으로 보면 선형 모형으로 설명할 수 있다**



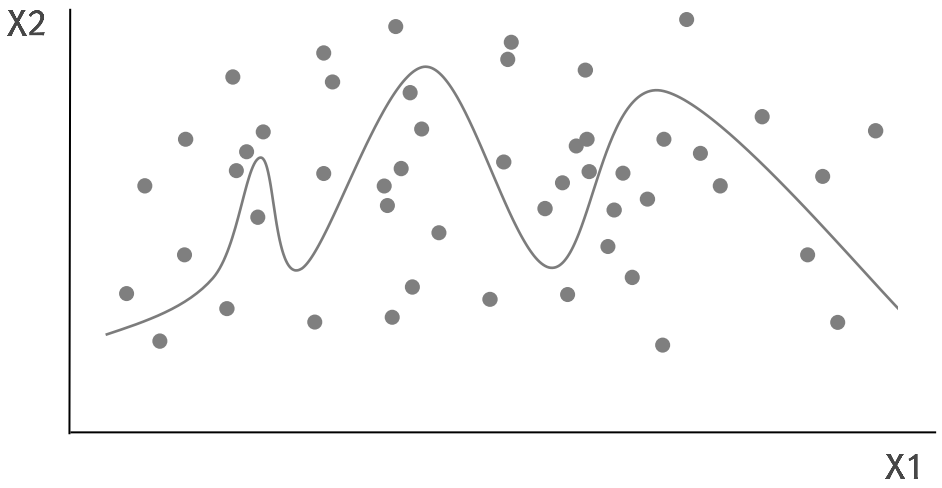
Local Surrogate Model은 왜 필요할까?

- 전체의 경향이 아닌 **개별 Instance에 대한 해석과 원인 분석**이 용이
- 개별 Instance(**개별 고객**..)에 대한 해석력을 제공함으로써 **높은 활용성**
- 기존(As-Is) 모형은 잘 맞추는데 신(To-Be) 모형에서는 **예측을 잘 못하는 특정 Case**가 있다면..
- 기존 데이터 분포와 다른 **새로운 Instance의 영향 파악**해야 한다면..

Concept for Calculation

Step1

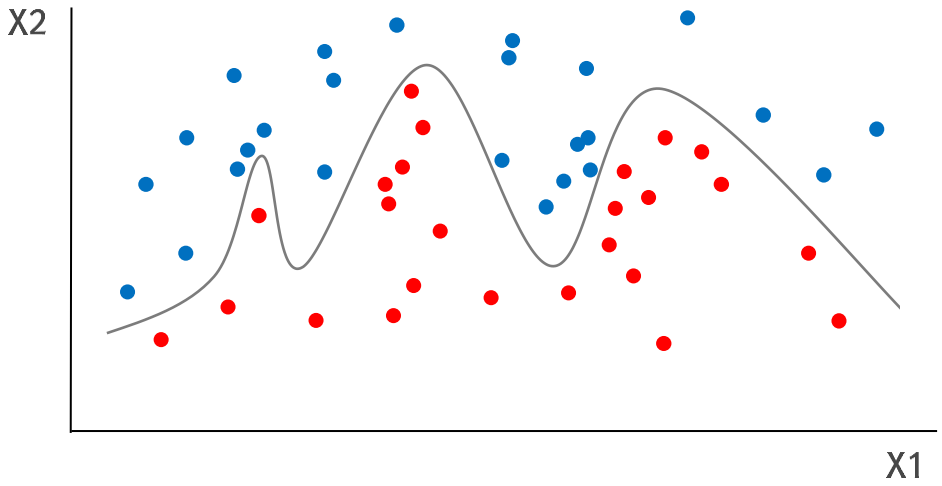
- Original Data의 변형(perturb)을 통해서 Sampling Data를 생성



Step2

- 만들어진 Sample을 Black-box Model(Original Model)을 통해서 예측값을 얻음(Labeling)

Getting Y value by Original Model



LIME (Local Interpretable Model-agnostic Explanation)

Concept for Calculation

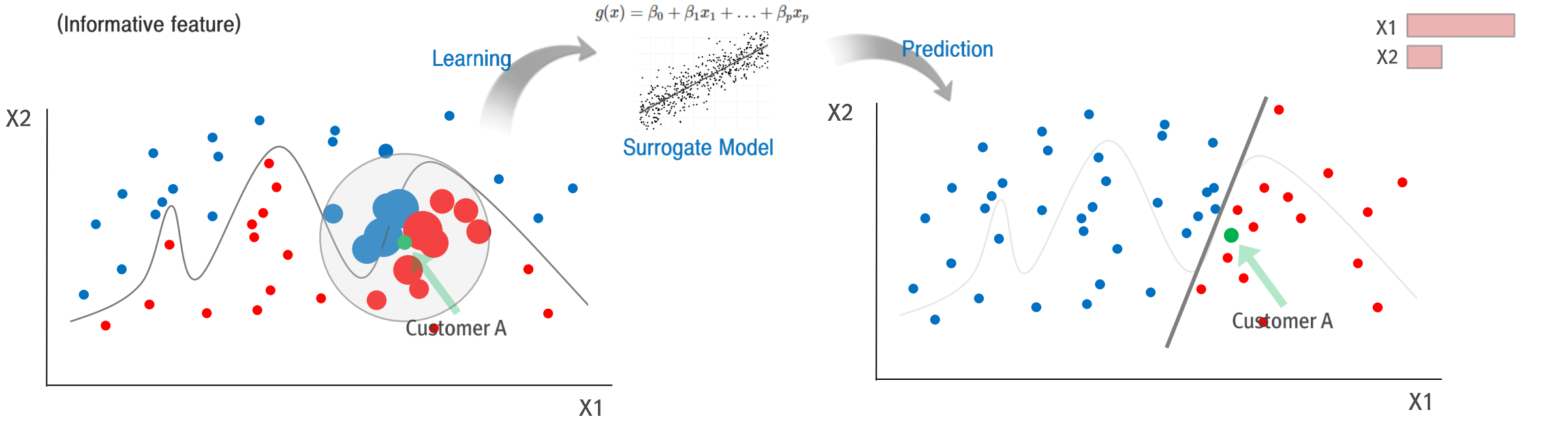
$$\text{explanation}(x) = \arg\min_{g \in G} L(f, g, \pi_x) + \Omega(g)$$

Step3

- 설명하고자 하는 관측치에 근접하는 정도에 따라 가중치를 부여
- 설명가능한 대리모델을 선정한 후 가중치가 부여된 Sample을 학습
- 계산 효율화를 위해 적정한 Feature 수를 도출할 수 있음 (Informative feature)

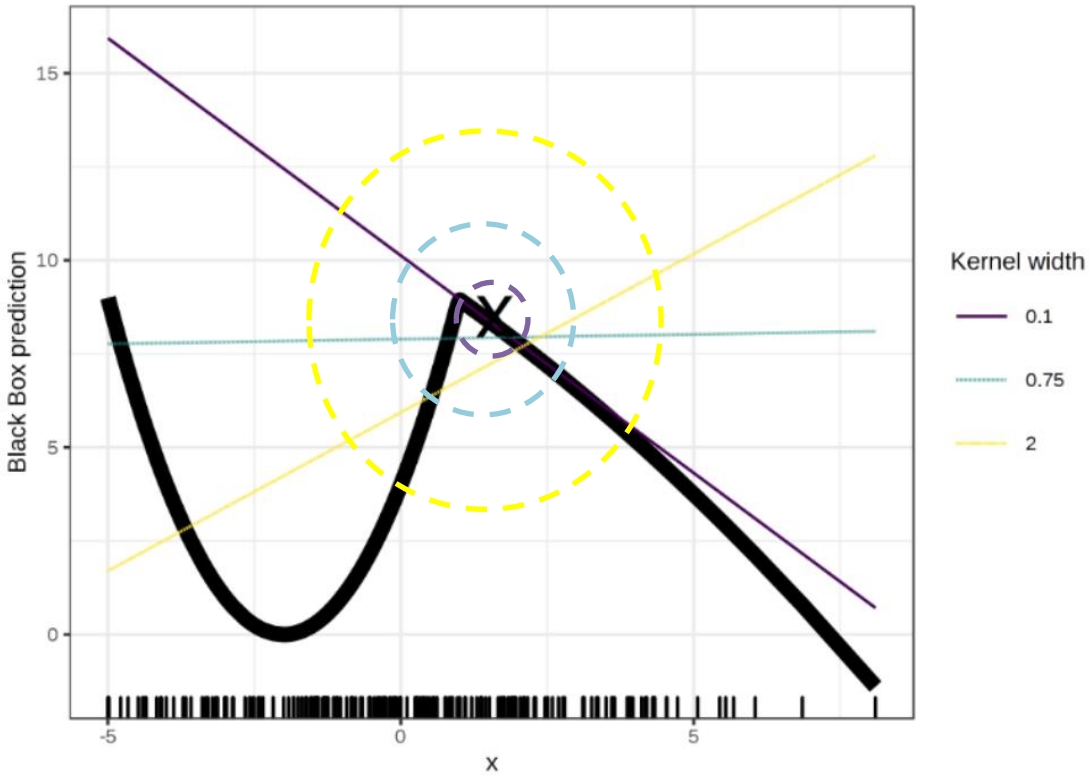
Step4

- 대리모델을 통해서 예측하고 설명하고자 하는 관측치에 대한 설명력을 제공



LIME (Local Interpretable Model-agnostic Explanation)

설명하고자 하는 관측치와의 근접도에 따라 가중치를 부여하는데, 근접하다는 것은 정의는?



“ Kernel width에 따라 전혀 다른 해석력 ! “

- Exponential smoothing을 이용해 가까운 점에는 높은 가중치, 먼 점에는 낮은 가중치를 부여함
- Smoothing kernel에서 kernel width는 얼마나 많은 이웃을 탐색할지 결정하기 때문에 매우 중요함
- 일반적으로 kernel width는 0.75 * sqrt(feature count)를 이용
- Feature의 차원이 많아질수록 결과가 좋지 않을 수 있음

LIME (Local Interpretable Model-agnostic Explanation)

Pros

- 기존 학습 모델을 교체하더라도 로컬 대리 모델을 사용하여 동일한 설명력을 제공할 수 있음

  Ex) “Decision Tree” 방식의 설명력 제공에 익숙한 사용자, 기존의 모델(SVM)보다 더 나은 예측력을 제공하는 신모델(DL)이 있을 때, 신모델로 교체하더라도 “Decision Tree”의 설명력을 계속 제공할 수 있음

- Local Surrogate Model을 통해서 해석이 용이한 Feature로 설명력을 제공할 수 있음

  Ex) Black-box Model은 PCA(추상화), 변수변환에 의한 Feature를 사용하여 예측값을 제공하지만, Local Surrogate Model은 Black-box Model이 학습한 Feature와 다르게 원본 Feature를 활용한 설명력을 제공할 수 있음

- Image, Text 등 비정형 데이터에 대해서도 유연하게 작동

Cons

- 해석에 큰 영향을 주는 이웃에 대한 적절한 정의는 해결하기 어려운 문제
- Gaussian 분포에서 제공되는 샘플링은 Feature간의 상관관계를 반영하기 어려움
- 같은 관측값임에도 대리모형이 만들어질 때마다/Sampling 될 때마다 결과가 상이할 수 있음 (Sample size를 늘려라..)

SHAP(SHapley Additive exPlanations)

Key Idea

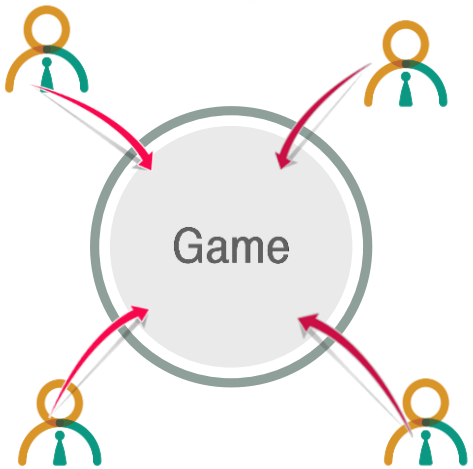
Feature간의 상호작용을 면밀히 고려해볼 수 없을까? Feature 조합의 모든 경우의 수를 따져보자

- Inspired by Geme Theory – 2012 노벨 경제학상 수상, Lloyd Shapley

Game Theory

- 상대방의 행위가 자신의 이익에 영향을 미칠 경우 이와 관련해 **이익을 극대화 하는 최선의 의사결정에 대한 연구**
- 상대방의 결정에 상관없이 취할 수 있는 최선의 전략 → Dominant Strategy
- 각 참가자의 전략이 주어져 있을 경우 → Nash Equilibrium
- 각 참가자가 협조적일때 (Win-Win) 최선의 전략 → Cooperative Game (Stable allocations)
- …

모두를 만족하게 하는 상금의 배분



SHAP(SHapley Additive exPlanations)

Concept for Calculation

X1, X2, X3 3개의 Feature를 활용해 Y를 예측 ,

X1의 Shapley Value?

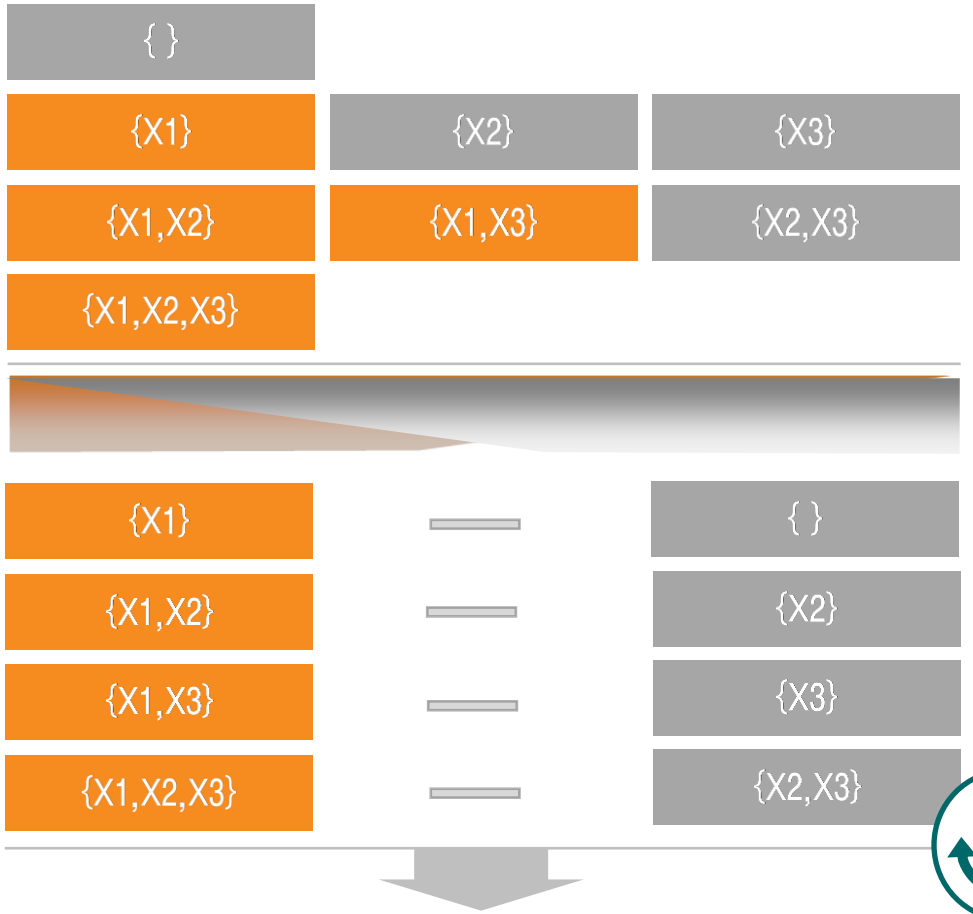
모든 경우의 수(조합)를 도출하자

$_3C_0 = 1$

$_3C_1 = 3$

$_3C_2 = 3$

$_3C_3 = 1$

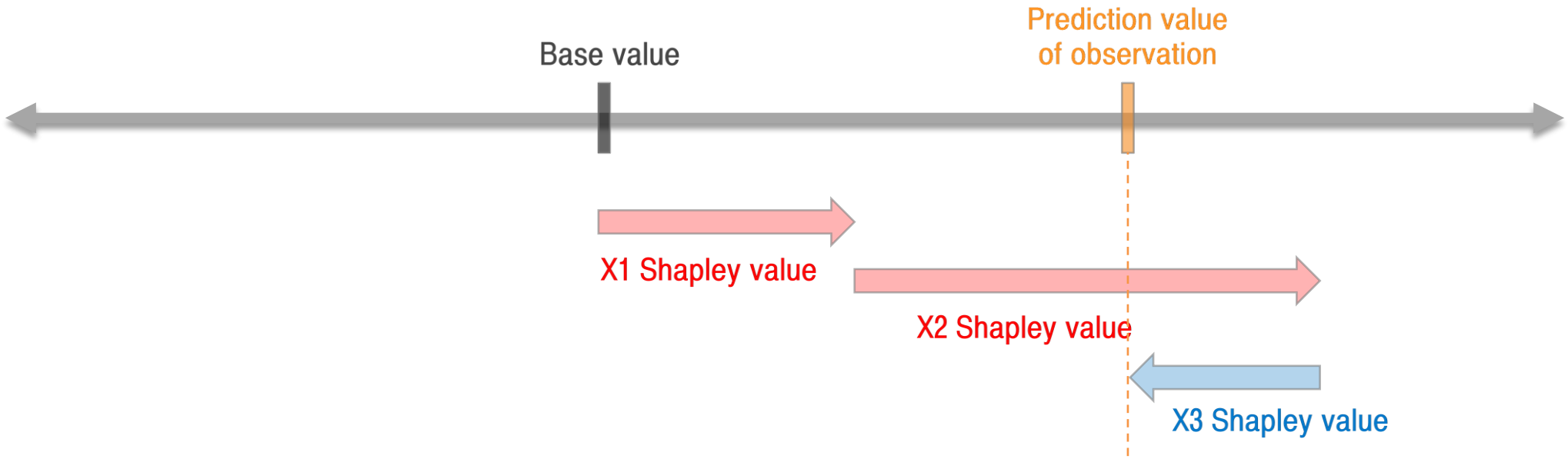


Shapley Value Estimation

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} \left[f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S) \right]$$

예측값 차이의 가중평균 = X1의 Shapley Value = Marginal Contribution Avg.

Concept for Calculation



" Base value와 개별 Instance의 예측값과의 차이 = Sum(Shapley Value) "

Shapley Value of Real World

- Shapley value은 견고한 이론적 기반과 흥미로운 속성을 가지고 있지만 계산하기 쉽지 않음
- Feature 3개 – 8조합
- Feature 10개 – 1,024조합
- Feature 20개 – 1,048,576조합
- ..

Kernel SHAP : Linear LIME + Shapley Value

Tree SHAP : Tree Based Model

Deep SHAP : Deeplearning based model

SHAP(SHapley Additive exPlanations)

Pros

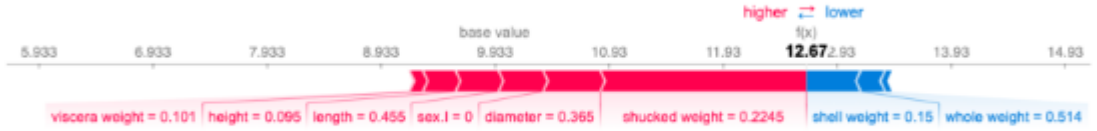
- 예측값(Y)를 결정하는 기준을 **균형있게 해석**할 수 있음
- Feature간 의존성을 고려해서 모델 영향력을 계산 (Multicollinearity)
- **음의 영향력을 고려할 수 있음** (Negative Feature Importance)
- Feature 값에 따라 Y가 어떻게 변화하는지 설명이 가능

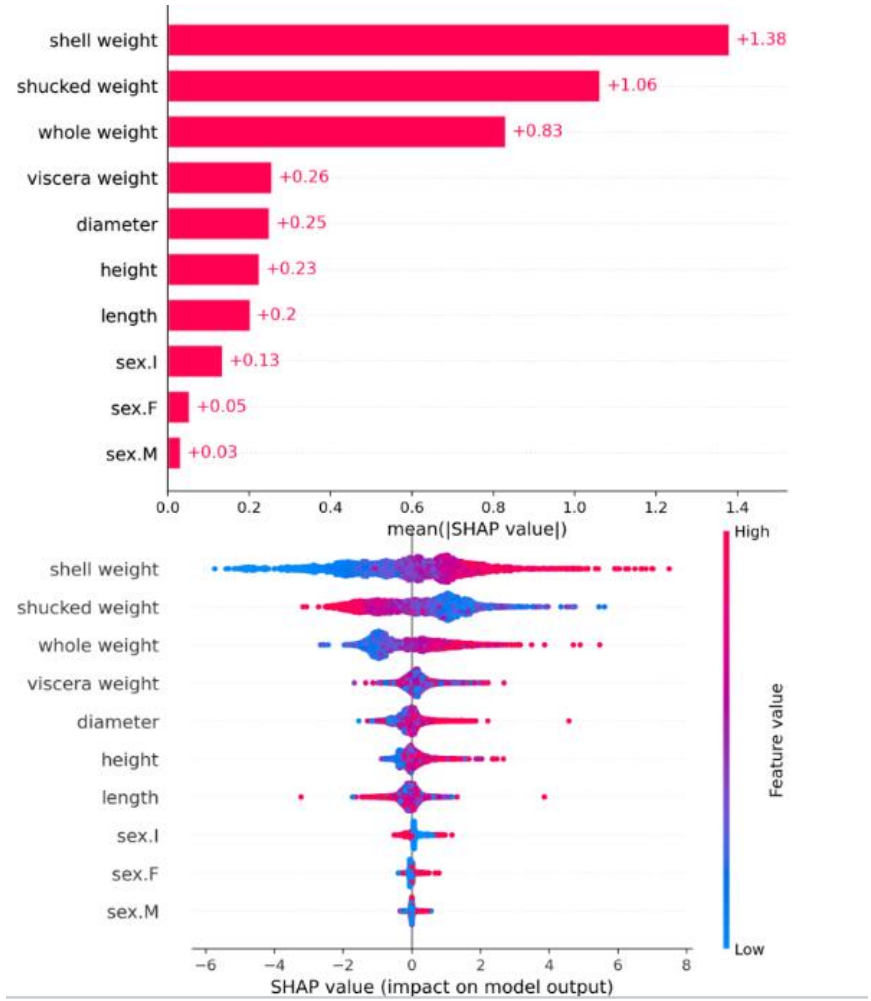
Cons

- Feature수 에 따라 **기하급수적으로 늘어나는 연산비용**
- 연산비용의 한계로 Sample **계산량을 줄이면 오차의 분산이 커짐**
- 이상치/새로운 데이터에 대해 허술한 해석을 내놓을 가능성

SHAP(SHapley Additive exPlanations)

Explanation chart





PDP (Partial Dependence Plot)

Key Idea

중요변수를 찾는 것을 넘어 특정 Feature와 Target(Y)의 관계를 설명하려면..

다른 Feature는 **고정하고** 관심 Feature를 **구간별로 변화시키면서** Target(Y) **예측의 변화를 관찰**해보자

- Partial Dependence – “단일 Feature의 입력에 따라 모델의 예측이 어떻게 의존하는가? (변화하는가?)“
- 결과를 직관적으로 확인할 수 있도록 그래프(plot)으로 나타내자

PDP (Partial Dependence Plot)

Concept for Calculation

Original Data Set

curb_weight (X1)	cylinder_num (X2)	num_of_doors (X3)	…	city_mpg (x10)	car_price (Y)
2765	4	2	…	16	21105
3055	4	4	…	17	24565
…	…	…	…	…	…
3230	6	4	…	8	36880
1874	3	4	…	29	6575

X10의 PDP를 그려보자

City_mpg(X10) 값 서열화

{ 8, 16, 17, … , 29 }

다른 Feature는 고정한채로 순차적으로 투입

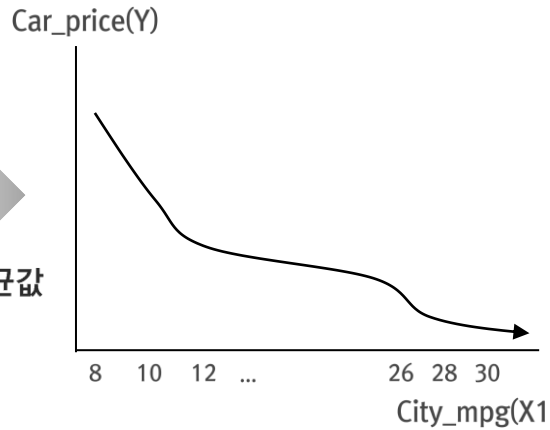
다른 Feature는 상수 취급

curb_weight (X1)	cylinder_num (X2)	num_of_doors (X3)	…	city_mpg (x10)	car_price (Y)
2765	4	2	…	8	21105
3055	4	4	…	8	24565
…	…	…	…	…	…
3230	6	4	…	8	36880
1874	3	4	…	8	6575

모든 Instance에 대해 예측

Pre-trained Model

예측 평균값 Plotting



PDP (Partial Dependence Plot)

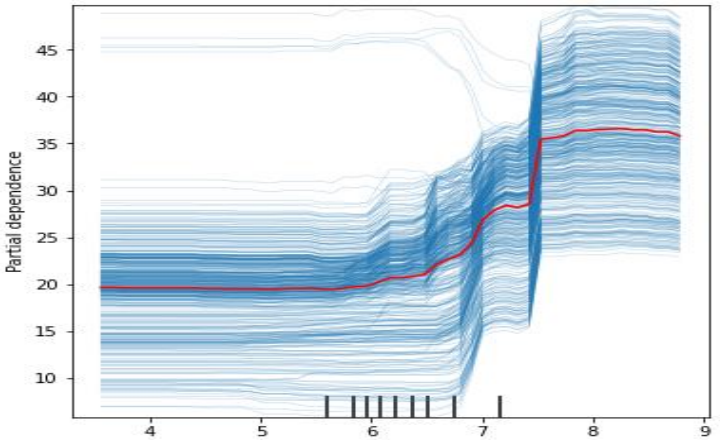
Pros

- Feature의 값이 변할 때 모델(예측값)에 미치는 영향을 **그래프를 통해 가시적으로 이해**할 수 있음
- Feature값에 개입하고 예측값의 변화를 측정함으로써 **특성과 예측사이의 인과관계를 추론**해 볼 수 있음
- Feature가 독립적일때 해석이 명확

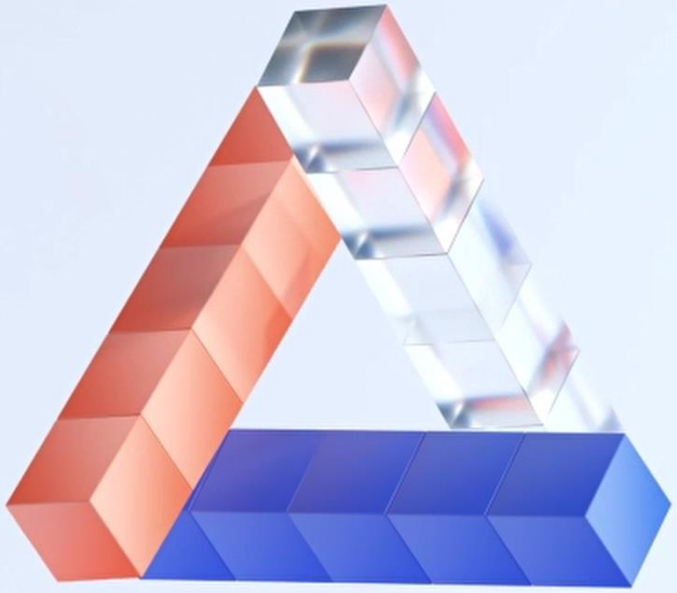
Cons

- 그래프(plot)을 통해 표현하므로, **표현의 한계**를 가짐 (최대 3차원, Feature 2개까지)
- 평균 효과로 표현하므로 이질적인 효과가 숨어있을 수 있음
  → 예측값의 분산이 클 경우 (크고 작은 예측값들..) **평균을 구하는 과정에서 서로 상쇄**
- Feature간 독립성을 가정하기 때문에 상관관계가 있을 경우 잘못된 설명력을 전달할 수 있음

[참고] 전체가 아닌 개별 Instance에 대해 Plot하면 ICE(Individual Condition Expectation)



DEVOCEAN



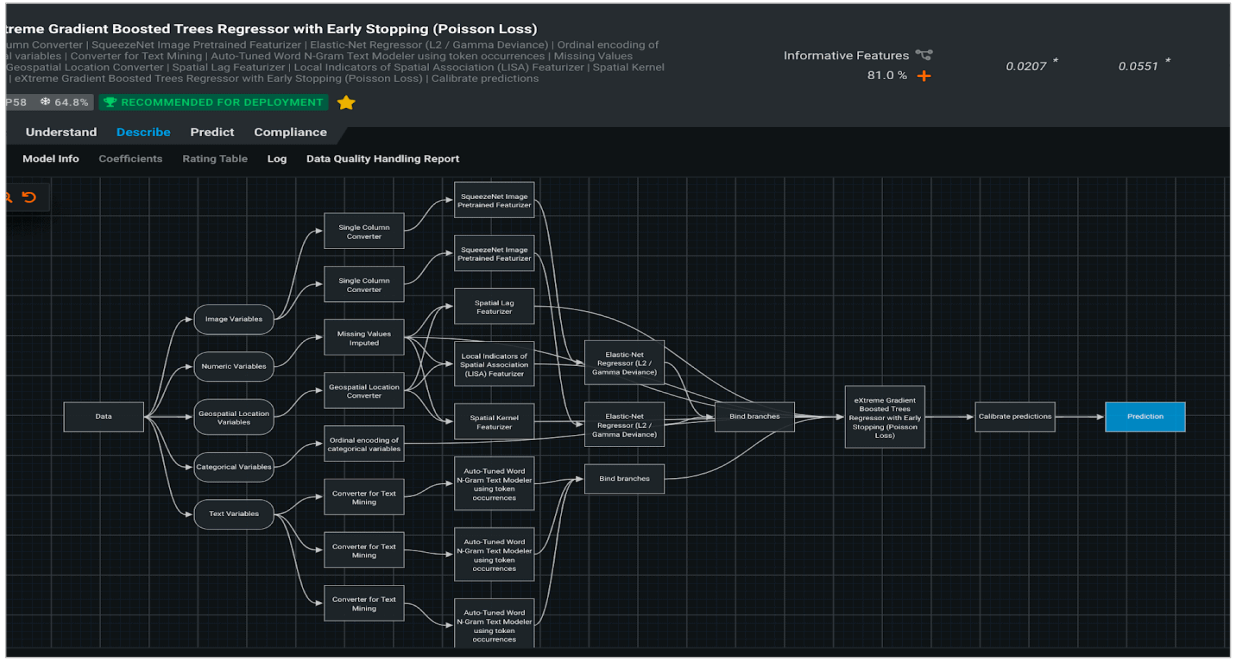
1. Intro
2. Hyper-parameter Optimization
3. XAI (Permutation, Surrogate Model, LIME, SHAP, PDP)
4. Use-case, Etc.

Think about..

경쟁할 모든 알고리즘에 대해 전체 데이터를 Train 해야 할까?

최적의 Metric은 무엇일까?

Feature engineering은 미지의 영역인가?



AutoML Use-case

Healthcare

응급환자 심정지 예측

목표 및 기대효과

- 응급실 환자의 24시간 이내 심정지 발생 위험을 AI로 예측하는 과제
- 예상하지 못한 악화 위험을 줄여주고, 의료진의 업무부담 낮추기 위함

주요 Input

- 호흡, 산소포화도, 체온, 혈압, 맥박 등 Vital 측정항목
- 산소처방여부 등 의사소견 데이터

Customer Voice

"머신러닝 입문자도 간편하게 여러가지 알고리즘으로 모델링 할 수 있어요"

"모델 리포트를 자동 생성해주는 것이 장점이에요"

알고리즘 최적화를 통한 예측력 향상

기존 예측모형 성과 **81%**

DataRobot

예측 정확도 **88%** 향상!

- 기존 예측모형과 동일한 Feature를 활용했음에도 DataRobot를 통한 최적의 알고리즘 선택으로 예측 정확도의 향상을 보임
- 다양한 최신의 후보 알고리즘군에서 선택한 최적의 알고리즘

예측 모델링 리스크 사전 제거

Target Data → 불균형 탐색

주요 변수 → 과적합 요인 발견

DataRobot에 의한 **예측 모델링 리스크 감지**

- DataRobot에서 제공하는 자동화 분석정보를 활용하여 위험요인 사전적 제거

AutoML Use-case

Epilogue

Commerce

온라인 주문 예측

목표 및 기대효과

- 품목별 특성을 반영한 주문수량 예측 모델 개발
- 예측 모델을 통해 최소화된 배송 리드타임 체계를 구축하고, 자동화된 주문 발주 시스템으로 연계

주요 Input

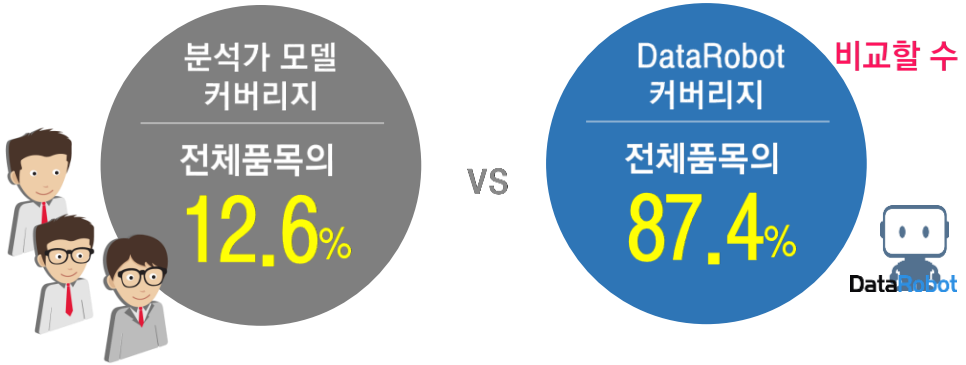
- 내부 거래 내역, 프로모션 내역
- 물가변동, 기상정보 등 외부 데이터
- 검색어 통계 등 마켓센싱 데이터

Customer Voice

"Auto TS를 통해 시계열 예측 모델링을 쉽게 적용할 수 있어요"

"DataRobot 덕분에 전 품목으로 주문예측을 확대할 수 있었어요"

예측 품목(커버리지)의 비약적 증가



비교할 수 없는 생산력 !

- 유통분야에선 다양한 품목별 특성을 반영한 예측이 중요
- 분석가 에 의해 일부 품목에만 적용하던 예측모형을 DataRobot 도입 후 전품목으로 예측모형 적용을 비약적으로 확대함

최소의 구축 비용으로 만든 신뢰할 수 있는 결과

구축Resource	개발 소요기간	모델 배포
90% 감소	1 week	5 min

- DataRobot의 자동화 머신러닝은 분석 전문가에 의해 개발되는 기존의 수요 예측모형 구축비용을 획기적으로 줄여줌

AutoML Benefit

01 최소의 Resource로 최대의 효과를!

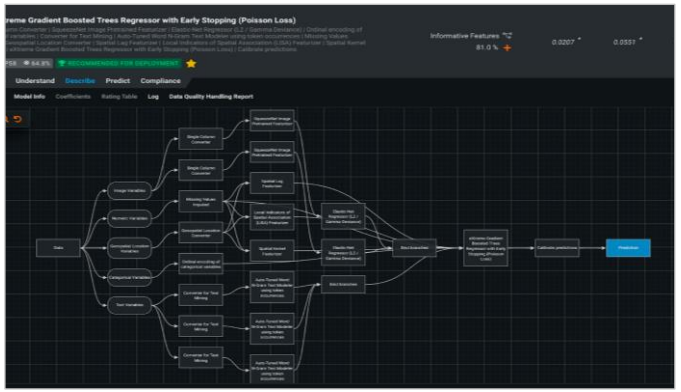
- Data Scientist를 포함한 인적구성과 일정 기간 소모가 불가피한 데이터 분석 과제 추진 시 DataRobot를 활용하면 **최소의 인력과 획기적인 시간 감축으로 데이터 분석 과제 추진**이 가능

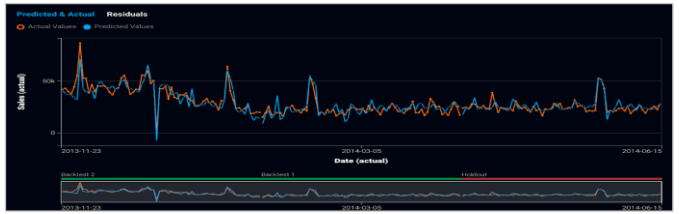
02 검증된 알고리즘, 신뢰할 수 있는 결과물

- 과제에 참여한 데이터 분석가의 숙련도에 따라 과제 결과물의 품질 차이를 수반하고, 결과물의 객관적인 성과를 판단하기 어려운 반면, DataRobot을 도입하면 **거의 모든 검증된 알고리즘을 활용하여 과제를 수행**할 수 있고, **알고리즘 간 경쟁을 통해 신뢰할 수 있는 과제 성과**를 담보

03 과제 아이디어, 데이터만 준비하면 ok

- 데이터 기반 비즈니스 혁신을 위한 아이디어와 관련 데이터만 준비하면 DataRobot을 통해 **쉽고 빠르게 아이디어가 동작하는지 실현 가능성 (feasibility)을 검증**
- 데이터 분석 과제 진행 시 비용/시간 등 **실패에 따른 내재 Risk를 최소화**





“Data Scientist 에게 AutoML은 어떤 느낌일까요?”

Task를 돕는 든든한 아군일까요? 나의 역할을 앗아가는 침략자일까요?

DEVOCEAN

감사합니다.

세미나 내용과 관련되어 궁금한점은 언제든지..

Byungsun, Park

SK주식회사 C&C, Senior Data Scientist

parkbs@sk.com

Byungsun.park@outlook.com