

Conservative Objective Models for Effective Offline Model-Based Optimization

고려대학교 산업경영공학과

데이터 어널리틱스 및 헬스케어 시스템 연구실

권정을(Master Program 1)

Keywords: Model-based-Optimization, Gradient Ascent, Manifold Learning

DAHS

Data Analytics and
Healthcare Systems LAB

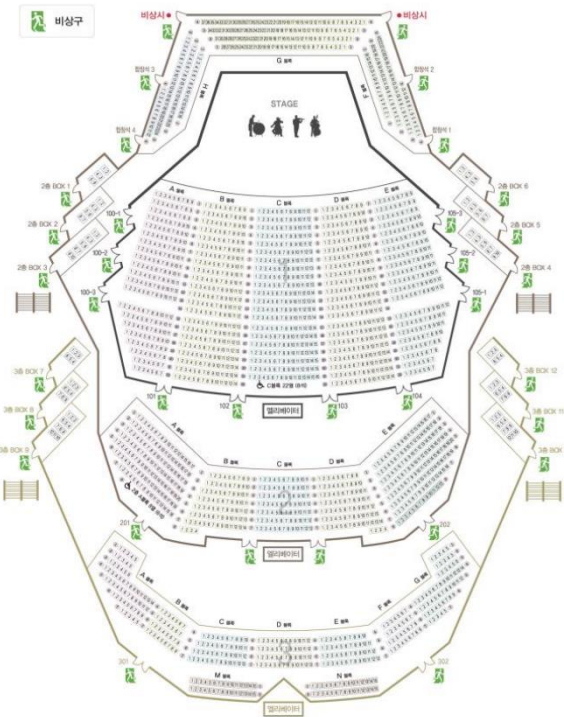
1. Introduction: Example: Real world, MNIST classification
2. Background
3. Conservative Objective Models for Effective Offline Model-Based Optimization
4. Experiment
5. Reference

❖ Example: Real world

클래식 공연 활성화를 위한 효과적 가격 모델 수립

예술의 전당-정형데이터 분석분야/어드밴스드 리그/2023 빅콘테스트

- 개인별 좌석에 대한 선호도, 공연특성 등을 고려하여 적정 가격을 도출할 수 있는 효과적인 가격 모델을 제시하는 것이 본 대회 문제임
- 예측된 가격은 현실성이 있어야 하며, 고객의 만족도를 최대화 할 수 있어야 함
- 객단가, 예매율, 기획사 공연 횟수, 국립 단체 여부, 매진 여부, 기획 여부 등의 과거 데이터를 기반으로 미래에 발생할 특정 공연에 대한 합리적인 가격을 도출할 수 있어야 함
- 또한데이터 내에는 2020년과 2021년에 수집된데이터도 포함되어있기 때문에 코로나 특수성을 고려해야함



❖ Example: Real world

클래식 공연 활성화를 위한 효과적 가격 모델 수립

예술의 전당-정형데이터 분석분야/어드밴스드 리그/2023 빅콘테스트

- 본 대회는 다음과 같은 사항을 중점으로 평가를 진행
 - 논리적으로 가설이 잘 설정이 되었는가(논리성)
 - 분석 결과값 및 예측 정확도(정합성)
 - 데이터 활용의 참신함 정도(참신성)
 - 분석 방법론의 논리성 및 효과적인 시각화(타당성)
 - 예술공연 특성 고려한 현실적인 결론 도출 여부(현실성)

테이블설명		예술의전당 공연예매정보			
No.	컬럼ID	컬럼명	타입	NULL	비고
1	age	나이	bigint	Y	10살 단위 / 80세 이상은 모두 80, 10세 이하는 모두 10
2	gender	성별	text	Y	
3	membership_type_1	멤버십종류1	text	Y	한 회원당 복수개의 멤버십 등록이 가능
4	membership_type_2	멤버십종류2	text	Y	
5	membership_type_3	멤버십종류3	text	Y	
6	membership_type_4	멤버십종류4	text	Y	
7	membership_type_5	멤버십종류5	text	Y	
8	membership_type_6	멤버십종류6	text	Y	
9	tran_date	예매 거래일자	text	N	YYYYMMDD
10	tran_time	예매 거래시간	text	N	HHMM
11	play_date	공연날짜	text	N	YYYYMMDD
12	play_st_time	공연시작시간	text	N	HHMM
13	seat	좌석번호	text	N	
14	price	최종예매가격	bigint	N	천원 단위
15	ticket_cancel	예매취소여부	text	N	2 : 예매취소, 0 : 예매취소하지 않음
16	discount_type	할인내역	text	Y	
17	performance_code	공연명 코드	text	N	
18	pre_open_date	선예매시작일	text	Y	YYYYMMDD
19	open_date	예매시작일	text	Y	YYYYMMDD
20	genre	장르	text	Y	
21	place	공연장소	text	N	
22	running_time	러닝타임	bigint	Y	단위 : 분
23	intermission	휴게시간	bigint	Y	단위 : 분 / 30분 이상 값 모두 30
24	member_yn	회원여부	text	N	Y : 예술의전당 회원, N : 비회원



이러한 문제를 Machine Learning이나 Deep Learning으로 풀 수 있는가?
만약에 있다면, 중점적으로 평가하고자 하는 사항을 만족할 수 있는가?

❖ Example: Real world

클래식 공연 활성화를 위한 효과적 가격 모델 수립

예술의 전당-정형데이터 분석분야/어드밴스드 리그/2023 빅콘테스트

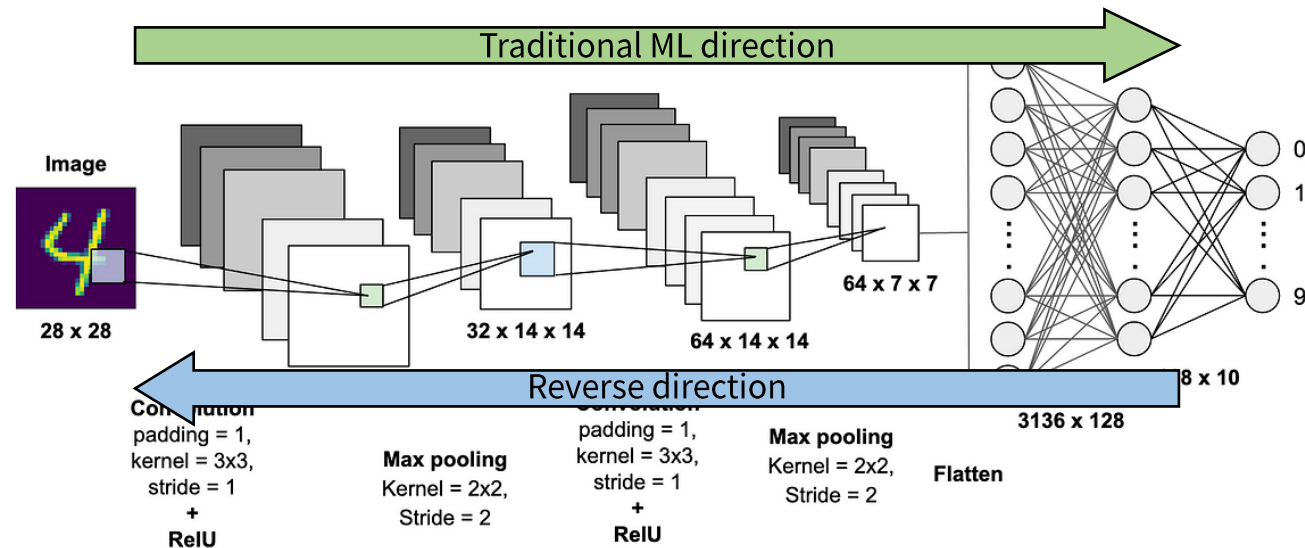
- 논리적으로 가설이 잘 설정이 되었는가(논리성):

“공연 A 관람에 어떤 예매 가격 M 을 제시했을 때, 관객수가 많을 수록 M 은 합리적인 가격이다.”

- 그렇다면 공연 별 관객 수를 추정하여 종속 변수(y)로 두고, 예매 가격을 포함한 여러 요인을 독립 변수(x)로 두어 y 를 **Maximize** 하는 최적의 x 들의 조합을 찾으면 되겠구나! → **Model Based Optimization(MBO)**

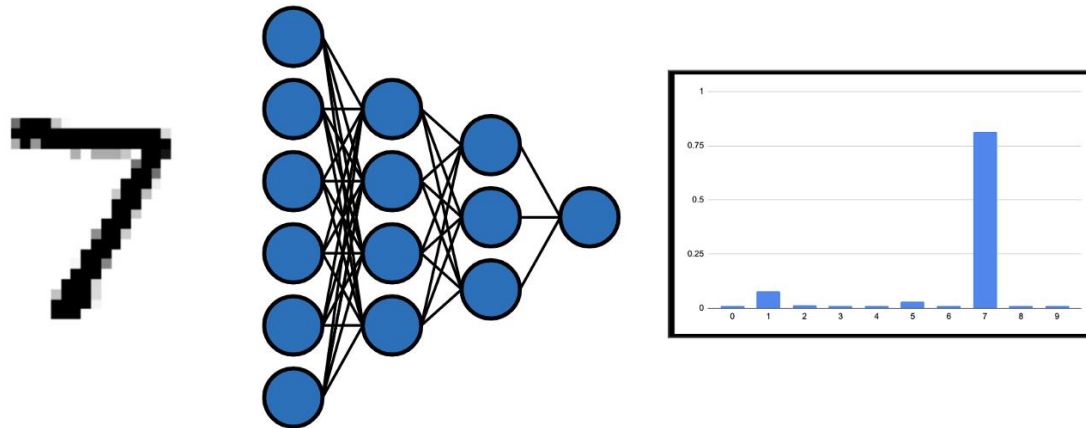
❖ Example: MNIST classification

- MBO는 전통적 ML의 **direction**을 **reverse** 하는 것
- 전통적 ML의 **direction**: Data point에 대해 특정한 property를 갖게 하는 함수(model)을 만드는 것
(MNIST classification → 모델에 입력된 숫자 이미지 “3”을 “3”이라 정확히 예측하는 것(data point, 여기서는 픽셀 값이 property를 갖게 됨))
- 전통적 ML의 **reverse direction**: Data point의 desired property가 숫자 “7”이라면, 숫자 “7”을 묘사하는 이미지를 만들 때까지 data point, 즉 픽셀 값을 수정하는 것



❖ Example: MNIST classification

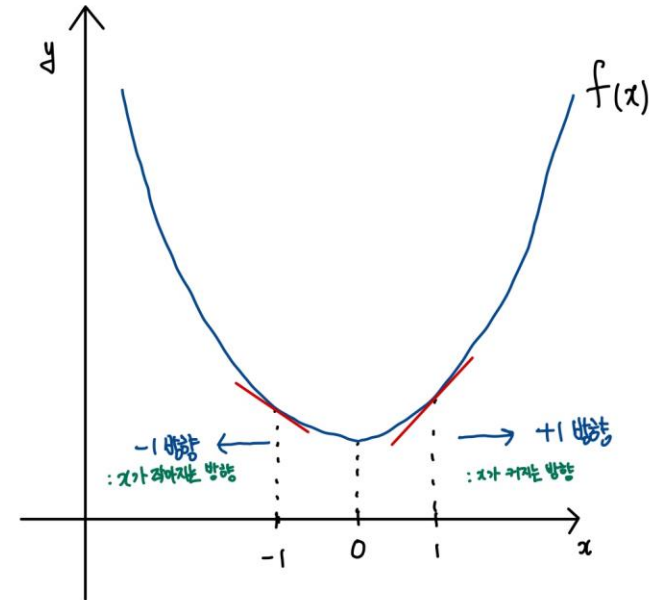
- Usual Goal(or direction): To build a classifier that takes as input an image of a hand written digit and predicts the digit that is being depicted.
- 예측 프로세스는 Neural Net의 likelihood vector가 0에서 9까지의 숫자에 특정 확률을 할당하는 것으로 이해할 수 있음
(아래 사진은 likelihood vector가 숫자 7에 가장 높은 확률을 할당하고 있음)
- 여기서 direction을 reverse하여 다른 likelihood vector의 확률이 큰 값을 가지도록 한다음 (예: 숫자 2) 입력 이미지를 실제로 숫자 2를 묘사하도록 수정할 수 있을까?
- 이렇게 할 수 있는 접근법이 있다면 원하는 property를 가진 새로운 샘플을 만들어 낼 수 있을 것임 → **Model Based Optimization(MBO)**



❖ Meaning of the Gradient of $f(x)$

- Gradient 는 공간에 대한 기울기를 말하며, 기울기는 벡터라고 말할 수 있고, 벡터는 방향을 가지고 있음
- 따라서 함수 $f(x)$ 의 gradient는 벡터이며, 어떤 방향을 가리키고 있음
 - $x = 1$ 일 때 $f'(1) \rightarrow x$ 가 커지는 방향을 가리킴
 - $x = -1$ 일 때 $f'(-1) \rightarrow x$ 가 작아지는 방향을 가리킴

“각각의 방향은 함수 값 $f(x)$ 가 커지는 방향을 의미”



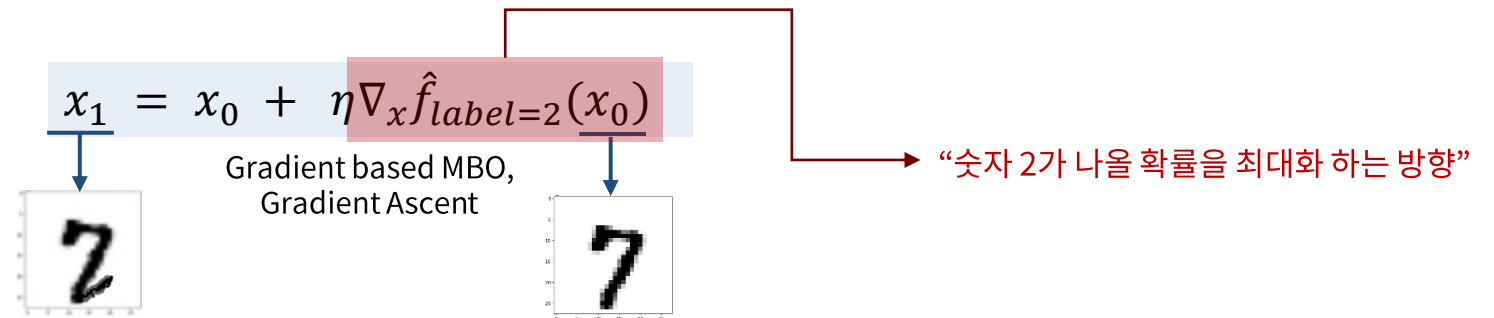
❖ Gradient Based MBO

- MBO의 궁극적인 목표는 Data point(입력 데이터)가 특정한 property를 갖도록 이를 변경하는 것
- 하지만 Data point의 property는 closed form으로 접근하기 어려움
→ 이러한 특성 때문에 (good)예측 모델을 먼저 만들고, 모델의 예측 값을 Optimize 하는 것이 일반적

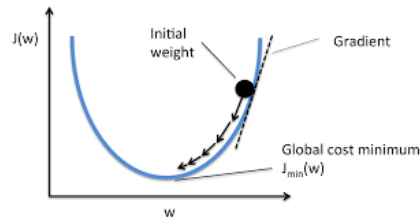
● Implementation

Remind: MNIST classification

- Classifier F 는 숫자 0 ~ 9 까지를 잘 분류하도록 훈련되어 있음
- 숫자 7 이미지를 입력 받은 모델이 MBO를 통해 입력을 수정하여 **입력 이미지가 숫자 2를 묘사하도록 만드는 상황**을 가정
- 해볼 수 있는 시도는 숫자 2에 확률을 할당하는 likelihood vector를 미분하고, 이 gradient를 숫자 7이미지에 더해 변형 시키는 것

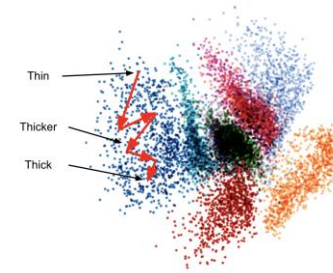


❖ Gradient Based MBO



$$w_{new} = w_{old} - \eta \nabla_w L$$

Gradient Descent



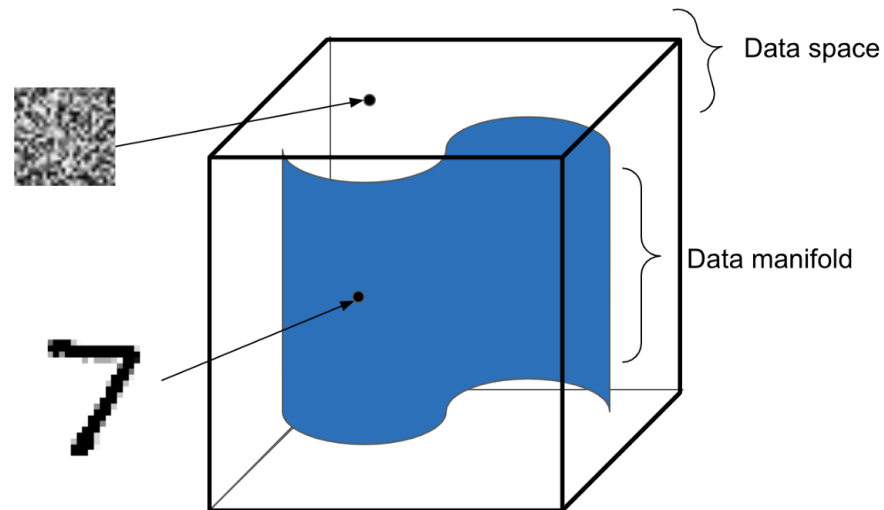
$$x_1 = x_0 + \eta \nabla_x \hat{f}(x_0)$$

Gradient Ascent

- Gradient Descent와 매우 비슷한 형태를 가지고 있음
- Gradient Descent는 Neural Net의 파라미터를 최적화하는 반면 Gradient Ascent는 input을 최적화
- 그러나 이러한 방식은 관점에 따라 **성공과 실패를 모두 갖고 있음**

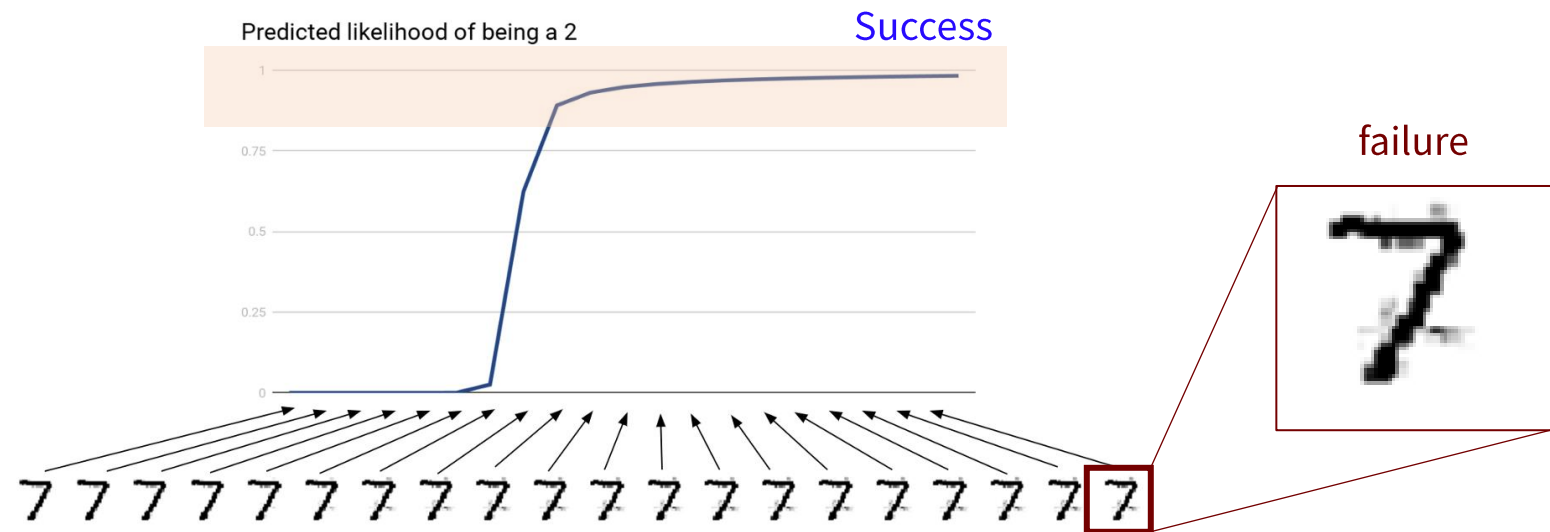
❖ About Manifolds

- Data Space: 모든 Data point를 포함한 공간
 - MNIST의 경우에는 1개의 이미지에 128x128 개의 픽셀 값이 있고, 각 픽셀은 가능한 모든 Real value를 가질 수 있음
 - 이 공간에 있는 하나의 이미지가 무조건 숫자를 묘사할지는 보장할 수 없음
- Data Manifold: Data Space의 sub space로써, 실제로 “유효한”(Valid) data point (more sparse!)
 - MNIST image의 경우 이 공간에 있는 data point는 실제 숫자를 묘사할 수 있는 픽셀 값



❖ Gradient Based MBO

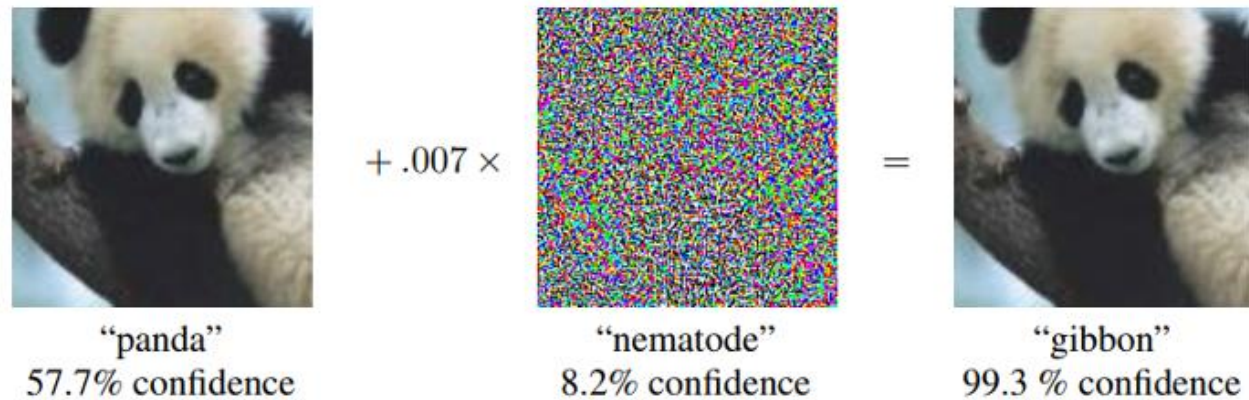
- Manifold는 Data space에 비해 너무 **sparse** 하기 때문에 Data point를 변경하는 작업이 **manifold 밖의 Data point**를 생성할 확률이 높음
- 이렇게 되면 변경된 입력을 다시 예측 모델에 넣었을 때, 원하는 모델의 출력(즉, 2라 예측)을 얻을 수는 있지만 실제 변경된 입력 값은 “**유효**”하지 않을 수 있음



Example) MNIST classification

❖ Gradient Based MBO

- 이는 Data space가 Manifold에 비해 너무 크며 Data point가 **Manifold** 공간을 벗어나지 않게 하기 위한 어떠한 제약 조건도 없기 때문임
- Neural Network는 “adversarial perturbation”에 굉장히 취약한 모델임
(이미지에 사람이 감지하기 힘든 아주 작은 노이즈를 추가했을 때 예측 모델의 성능이 현저히 낮아지는 현상)
- [1]에서는 이를 **Out-of-Distribution**이 발생했다고 표현하였음



[1] Trabucco, Brandon, et al. "Conservative objective models for effective offline model-based optimization." *International Conference on Machine Learning*. PMLR, 2021.

❖ Preliminaries

The Goal of MBO

- The goal of MBO is to find best possible solution x^* , to optimization of the form where $f(x)$ is an unknown objective function

$$x^* \leftarrow \operatorname{argmax}_x f(x) \quad (1)$$

Offline MBO

- An offline MBO algorithm is provided access to a static dataset \mathcal{D} of inputs and their objective values,

$$\mathcal{D} = \{(x_1, y_1), \dots, (x_N, y_N)\}$$

❖ Preliminaries

MBO

- Most method for tackling MBO problems fit a parametric model to the samples of the true objective function in \mathcal{D} , $\hat{f}(x)$, via supervised training:

$$\hat{f}_{\theta}(x) \leftarrow \operatorname{argmin}_{\theta} \sum_i (\hat{f}_{\theta}(x_i) - y_i)^2$$

- Find x^* in Equation 1 by optimizing x against this learned model $\hat{f}_{\theta}(x)$
- One choice for optimizing x in Equation 1 is Gradient Ascent on the learned function as given by:

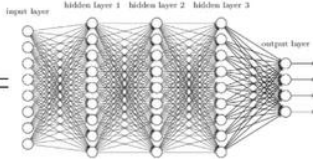
$$\underline{x_{k+1} \leftarrow x_k + \eta \nabla_x \hat{f}_{\theta}(x) \Big|_{x=x_k}, \text{ for } k \in [1, T], x^* = x_T.} \quad (2)$$

→ Risk of Out-of-Distribution !

❖ Proposed Method: COMs

1. Learning Conservative Objective Models(COMs)

- Standard supervised regression Using \mathcal{D}

$$\hat{f}_\theta(x) \leftarrow \operatorname{argmin}_\theta \sum_i (\hat{f}_\theta(x_i) - y_i)^2 \quad \hat{f}_\theta(\mathbf{x}) =$$


- Run multiple steps of gradient ascent on the current snapshot of the learned function $\hat{f}_\theta(x)$ starting from various inputs in the training dataset to obtain such adversarial inputs.
- all adversarial inputs found via this gradient ascent procedure as $\mu(x)$.

$$\mu(x) = \sum_{x_0 \in \mathcal{D}} \delta_{x=x_{t+1}} : x_{t+1} = x_t + \eta \nabla_x \hat{f}_\theta(x) \Big|_{x=x_k} \quad (3)$$

COMs regularization term

$$\begin{aligned} \hat{f}^*(x) \leftarrow \operatorname{argmin}_{\theta \in \Theta} & \alpha (\mathbb{E}_{x \sim \mu(x)} [\hat{f}_\theta(x_T)] - \mathbb{E}_{(x,y) \sim \mathcal{D}} [\hat{f}_\theta(x)]) \\ & + \frac{1}{2} \mathbb{E}_{(x,y) \sim \mathcal{D}} [(\hat{f}_\theta(x) - y)^2] \end{aligned} \quad (4)$$

❖ Proposed Method: COMs

1. Learning Conservative Models (COM)

Algorithm 1 COM: Training Conservative Models

1: Initialize \hat{f}_θ . Pick η, α and initialize dataset \mathcal{D} .

2: **for** $i = 1$ to training_steps **do**

3: Sample $(\mathbf{x}_0, y) \sim \mathcal{D}$

4: Find $\mathbf{x}_T(\mathbf{x}_0)$ via gradient ascent from \mathbf{x}_0 :
 $\mathbf{x}_{t+1} = \mathbf{x}_t + \eta \nabla_{\mathbf{x}} \hat{f}_\theta(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_t}; \mu(\mathbf{x}) = \sum_{\mathbf{x}_0 \in \mathcal{D}} \delta_{\mathbf{x}=\mathbf{x}_T(\mathbf{x}_0)}.$

5: Minimize $\mathcal{L}(\theta; \alpha)$ with respect to θ .
 $\mathcal{L}(\theta; \alpha) = \mathbb{E}_{\mathbf{x}_0 \sim \mathcal{D}} (\hat{f}_\theta(\mathbf{x}_0) - y)^2 - \alpha \mathbb{E}_{\mathbf{x}_0} [\hat{f}_\theta(\mathbf{x}_0)] + \alpha \mathbb{E}_{\mu(\mathbf{x})} [\hat{f}_\theta(\mathbf{x})]$
 $\theta \leftarrow \theta - \lambda \nabla_\theta \mathcal{L}(\theta; \alpha)$

6: **end for**

- α 의 값이 너무 작으면 $\hat{f}^*(x)$ 는 GT $f(x)$ 를 근사하지 못할 것임
- 반대로 매우 크면 입력 값 x 를 거의 변화시키지 못할 것임
→ 실험적으로, Continuous task에서는 0.5, Discrete task에서는 2.0이 적절한 값

❖ Proposed Method: COMs

2. Optimizing a Conservative Objective Model

- *eq 4*를 통해 conservative model $\hat{f}_{\theta}^*(x)$ 을 train 한 후에는 원래 목적인 x^* 를 찾아야 함
- COMs이 regularization term을 통해 Out-of-Distribution의 문제를 완화시켰기 때문에, $\hat{f}_{\theta}^*(x)$ 를 가지고 간단한 Gradient Ascent 만으로 입력 값을 Manifold 상에서 변경할 수 있다는 것이 저자의 주장

Optimizing Conservative Objective Model

- we can use a simple gradient ascent style procedure in the input space to find the best possible solution.

$$\forall t \in [T], x_0 \in \mathcal{D}; x_{t+1} = x_t + \eta \nabla_x \mathcal{L}_{opt}(x) \Big|_{x=x_t} \quad (5)$$

where $\mathcal{L}_{opt}(x) := \hat{f}_{\theta}^*(x)$

❖ Implementation Detail

- For all of experiments, the conservative objective model $\hat{f}_\theta(x)$ is modeled as a neural network (two hidden layers of size 2048, Leaky ReLU activation)
- We use the Adam optimizer with a learning rate of 10^{-3}
- We select the largest η such that successive \mathbf{x}_t follow the gradient vector field of \hat{f}^* .
(0.05 for continuous tasks and 2.0 for discrete tasks.)
- We use 50 gradient ascent steps($T=50$) starting from a given design in the dataset, $\mathbf{x}_0 \in \mathcal{D}$

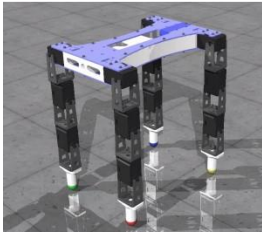
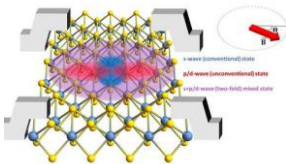
❖ 7 tasks derived from real-world design problems

1. Continuous Design

	Superconductor	Ant Morphology	D'Kitty Morphology	Hopper Controller
\mathcal{D} (best)	0.399	0.565	0.884	1.0
Auto. CbAS	0.421 ± 0.045	0.882 ± 0.045	0.906 ± 0.006	0.137 ± 0.005
CbAS	0.503 ± 0.069	0.876 ± 0.031	0.892 ± 0.008	0.141 ± 0.012
BO-qEI	0.402 ± 0.034	0.819 ± 0.000	0.896 ± 0.000	0.550 ± 0.118
CMA-ES	0.465 ± 0.024	1.214 ± 0.732	0.724 ± 0.001	0.604 ± 0.215
Grad.	0.518 ± 0.024	0.293 ± 0.023	0.874 ± 0.022	1.035 ± 0.482
Grad. Min	0.506 ± 0.009	0.479 ± 0.064	0.889 ± 0.011	1.391 ± 0.589
Grad. Mean	0.499 ± 0.017	0.445 ± 0.080	0.892 ± 0.011	1.586 ± 0.454
MINs	0.469 ± 0.023	0.913 ± 0.036	0.945 ± 0.012	0.424 ± 0.166
REINFORCE	0.481 ± 0.013	0.266 ± 0.032	0.562 ± 0.196	-0.020 ± 0.067
COMs (Ours)	0.439 ± 0.033	0.944 ± 0.016	0.949 ± 0.015	2.056 ± 0.314

Generative Models

Standard MBO

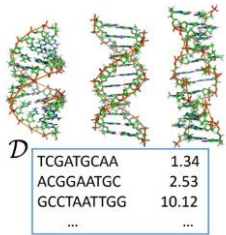


Superconductor task: high critical temperature를 가진 초전도 물질의 화학식을 설계하는 것 (유효성 검사 집합과 상관계수를 maximize 하도록) superconductor의 design space는 각 초전도체의 화학식의 원자 수와 이에 대한 혼합으로 86개의 실수이다.
Ant & D'kitty Morphology task: D'kitty라는 로봇의 네 개의 다리의 길이, 위치 및 방향을 나타내는 것을 포함한 로봇의 형태 구조를 정해진 위치로 움직이도록 design 하는 것이다.
Hopper Controller task: OpenAI Gym의 Hopper-v2이동 작업에서 reward를 최대화하기 위해 신경망 policy의 가중치를 최적화하는 것

❖ 7 tasks derived from real-world design problems

2. Discrete Design

	GFP	TF Bind 8	UTR	# Optimal	Norm. avg. perf.
\mathcal{D} (best)	0.789	0.439	0.593		
Auto. CbAS	0.865 \pm 0.000	0.910 \pm 0.044	0.691 \pm 0.012	1 / 7	0.687
CbAS	0.865 \pm 0.000	0.927 \pm 0.051	0.694 \pm 0.010	3 / 7	0.699
BO-qEI	0.254 \pm 0.352	0.798 \pm 0.083	0.684 \pm 0.000	0 / 7	0.629
CMA-ES	0.054 \pm 0.002	0.953 \pm 0.022	0.707 \pm 0.014	2 / 7	0.674
Grad.	0.864 \pm 0.001	0.977 \pm 0.025	0.695 \pm 0.013	3 / 7	0.750
Grad. Min	0.864 \pm 0.000	0.984 \pm 0.012	0.696 \pm 0.009	3 / 7	0.829
Grad. Mean	0.864 \pm 0.000	0.986 \pm 0.012	0.693 \pm 0.010	2 / 7	0.852
MINs	0.865 \pm 0.001	0.905 \pm 0.052	0.697 \pm 0.010	4 / 7	0.745
REINFORCE	0.865 \pm 0.000	0.948 \pm 0.028	0.688 \pm 0.010	1 / 7	0.541
COMs (Ours)	0.864 \pm 0.000	0.945 \pm 0.033	0.699 \pm 0.011	4 / 7	0.985



GFP(green fluorescent protein): GFP에서 파생된 단백질 구조 최적화 문제
TF Bind 8, UTR: DNA sequence 최적화 문제

- ❖ 본 논문을 한마디로 정의하면?
 - ✓ Data point가 Data Manifold 상에서 변경 될 수 있도록 Conservative한 estimate를 학습할 수 있는 MBO method 제안
- ❖ 본 논문과 관련된 본인의 아이디어는?
 - ✓ Early Prediction of Circulatory Failure:
 - ✓ 순환 부전 발생 여부를 예측하는 문제에서 순환 부전이 발생하는 관측치와 그렇지 않은 관측치의 비율이 극도로 왜곡된 경우가 많음
 - ✓ 이를 해결하기 위해 MBO를 활용하여 데이터 증강을 이룰 수 있을 것 같음

$$\mu(x) = \sum_{x_0 \in \mathcal{D}} \delta_{x=x_{t+1}}: x_{t+1} = x_t + \eta \nabla_x \hat{f}_\theta(x) \Big|_{x=x_k}$$

$$\hat{f}^*(x) \leftarrow \operatorname{argmin}_{\theta \in \Theta} \alpha(\mathbb{E}_{x \sim \mu(x)}[\hat{f}_\theta(x_T)] - \mathbb{E}_{(x,y) \sim \mathcal{D}}[\hat{f}_\theta(x)]) + \frac{1}{2} \mathbb{E}_{(x,y) \sim \mathcal{D}}[(\hat{f}_\theta(x) - y)^2]$$

$$\forall t \in [T], x_0 \in \mathcal{D}; x_{t+1} = x_t + \eta \nabla_x \mathcal{L}_{opt}(x) \Big|_{x=x_t}$$

where $\mathcal{L}_{opt}(x) := \hat{f}_\theta^*(x)$

T시간 이내 순환 부전 여부를 잘 맞추도록 훈련

T시간 이내 순환 부전이 일어날 Likelihood vector를 Maximize 하는 방향으로 입력 값을 변환
→ 증강된 데이터로서 활용

05 Reference

- [1] Trabucco, Brandon, et al. "Conservative objective models for effective offline model-based optimization." International Conference on Machine Learning. PMLR, 2021.
- [2] Trabucco, Brandon, et al. "Design-bench: Benchmarks for data-driven offline model-based optimization." International Conference on Machine Learning. PMLR, 2022.
- [3] <https://unit8.com/resources/model-based-optimization/>

Thank you

Question?