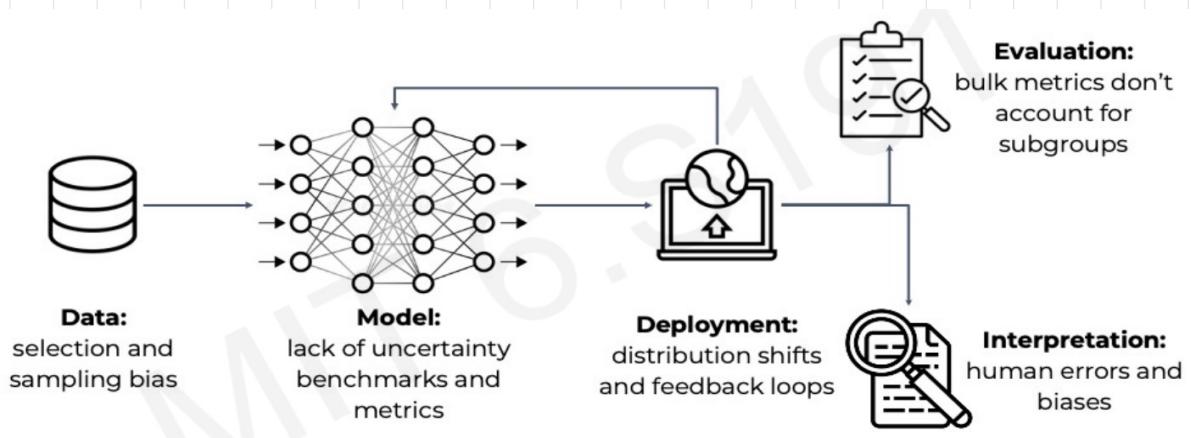


Problem in AI : 1. Bias : 편향

2. Uncertainty : 모델이 자신의 확률(증)을 얼마나 확신하나

Robust Deep Learning : 여러 가능한 상황에서도 성능이 크게 둘러치지 않는 딥러닝 모델
Noise, Miss data,
Adversarial attack...

1. Bias



- 특정 특징의 샘플을 선택하거나 여러 샘플을 특정 선택에서만 관리하기 쉬운 편향적 정부
- Model은 학습시간과 훈련 시간에 따라 훈련되는지에 따라 성능이 달라질 수 있다.
- Class Imbalance : Dataset 내에서 class 1과 class 2의 개수가 균형이 맞지 않는 상태
- 훈련?:

1. Sample Reweighting : Dataset 내에서 각 class의 데이터의 가중치를

→ Class 농도를 높은 것에 더 많은 가중치를 부여한다.

$$\rightarrow W_i = \frac{1}{\text{freq}(Y_i)}, L = \frac{1}{N} \sum_i W_i \cdot l(Y_i, Y_i)$$

2. Loss Reweighting : Loss function이 Underepresented class=1 dataset의 영향을 배제하도록 한다.

→ Loss의 가중치를 바꾼다.

3. Batch Selection : 각 Batch가 각 class마다 같은 개수의 데이터를 갖도록 한다.

→ 위 세 가지 방법은 수행하면 모든 Bias가 사라질까?

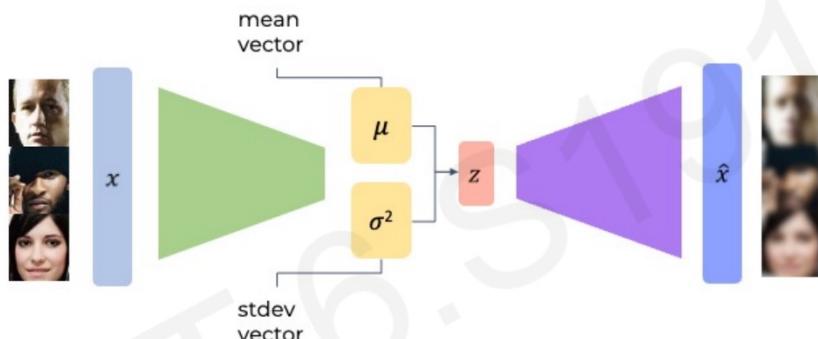
→ Latent space에서의 feature bias가 존재한다.

→ class bias를 제거하더라도 모든 feature를 관리해내는 것은 어렵다.

=> VAEs 등으로 Latent feature를 학습하고, 그에 따른 Biased feature sample을 Unbiased feature sample과 결합할 수 있다면 어떨까?

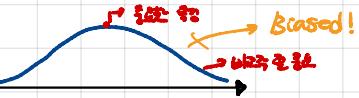
→ 모든 데이터에 적용 weight를 주는 방법입니다.

Mitigating Bias through Learned Latent Structure (VAEs)



1. Learn Latent Structure

2. Estimate distribution : $P(z|x)$



3. Adaptively guide learning



Using Latent Variables for Automated Debiasing

$$\hat{Q}(z|X) \propto \prod_i \hat{Q}_i(z_i|X)$$

Estimated joint distribution
Independence every latent variable z_i

$$W(z(x)|X) \propto \prod_i \frac{1}{\hat{Q}_i(z_i(x)|X) + \alpha}$$

Probability of selecting datapoint
Histogram for every latent variable z_i

$\hat{Q}_i(z_i)$ frequencies 선택하면 $W = \frac{1}{\text{Hist}}$

$\alpha \downarrow$
→ weight ↑ : debiasing

$\alpha \uparrow$
→ Uniform distribution 허용된다

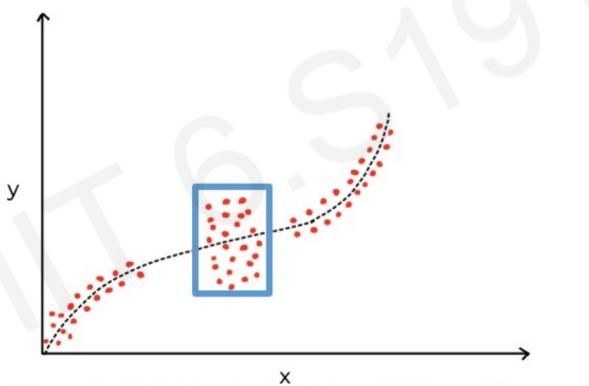
↳ feature간의 차이가 많아지면
flatten 된다.
→ 특징 간의 차이를 줄여야 하는 경우 debiasing
차이가 있다.

2. Uncertainty

Example of Uncertainty : 개별 고양이를 구분하는 이런 분류기에 말 데이터를 넣는 경우에,
Model은 개일 확률, 고양일 확률을 출력할 수 없어 같다. 이런 경우
개인의 예측 / 불확실하다는 표현을 해줘야 한다.

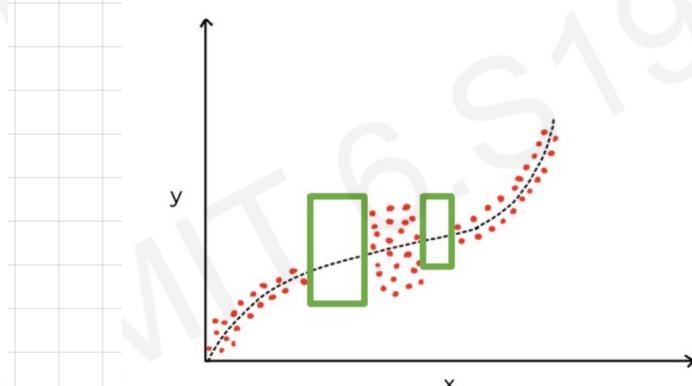
→ Model이 확률의 높은 시로운 예측이나 나타나는 경우

Uncertainty type 1



- Data를 설명할 distribution과 동일하지 않음,
비슷한 Input에 따른 Output의 차이는 큼

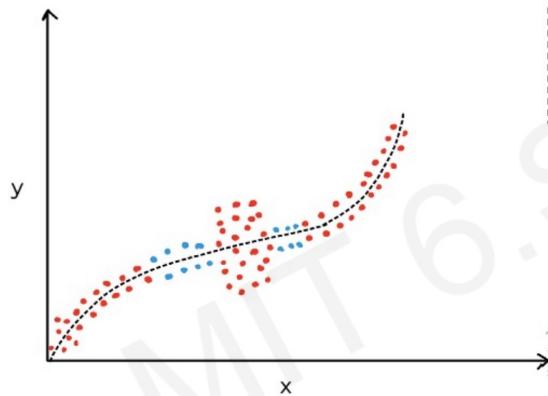
Uncertainty type 2



- Input data distribution에서 빈 공간은 모델이
정확히 추정할 수 없어 불확실성이 높다.

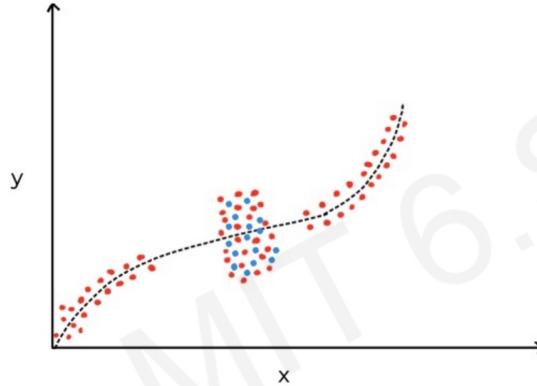
Mitigating Uncertainty : Add data?

1.



→ Add data? std type 2 error

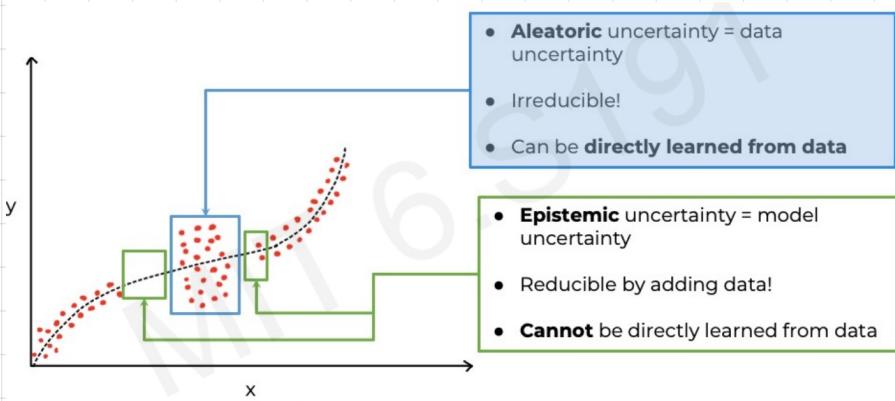
2.



→ Input data contains irreducible noise!

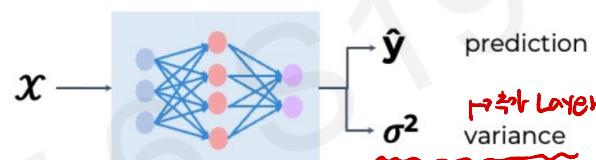
→ Irreducible

Aleatoric VS Epistemic Uncertainty



Estimating Aleatoric Uncertainty : Regression

→ Model o.1 Input $\in \mathbb{R}^n$ σ^2 (Variance) $\in \mathbb{R}$ \rightarrow $f_\theta(x)$



$$f_\theta(x) \rightarrow \hat{y}, \sigma^2$$

This variance is **not constant** and depends on the value of x!

→ $\sigma^2 \uparrow$: Uncertainty \uparrow

→ Loss function?

→ Loss Constant Variance \rightarrow MSE

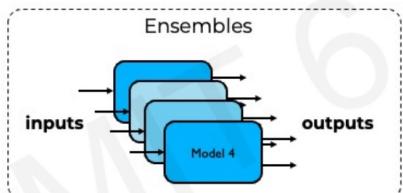
Negative Log Likelihood (NLL) is a **generalization** of MSE to **non-constant variances**:

$$\mathcal{L} = \frac{1}{N} \times \sum_{i=1}^N \frac{(\hat{y}_i - y_i)^2}{2\sigma_i^2} + \ln \sigma_i^2$$

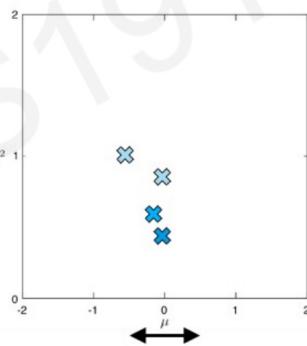
↑↑에 폐널티

Estimating Epistemic Uncertainty : Ensembling

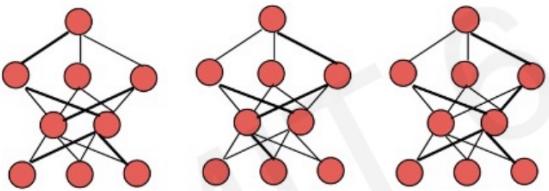
What if we train the same network multiple times (an **ensemble** of networks) and compare outputs?



→ Initialized weights 한骤로 모드 깊이 초기화
모든 모델은 같은 초기 가중치를 갖습니다.
Output 값



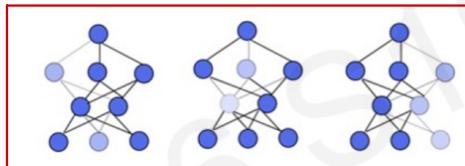
→ Output distribution
Mean, Variance 정의



단점: Cost ↑

Estimating Epistemic Uncertainty : Dropout

To introduce **stochasticity**, we can also add dropout layers and compute forward passes multiple times while saving memory and compute



Model 1 에서 확률적 에서 대비 확률적 Uncertainty ↑

- Inference 단계에서 Dropout 사용

→ 같은 데이터에 대해 구조는 다른
결과가 나오게 된다.

→ 미리 ensemble 계산 풍부

→ 해당 Output을 통한 = Uncertainty

→ Cost efficiency!

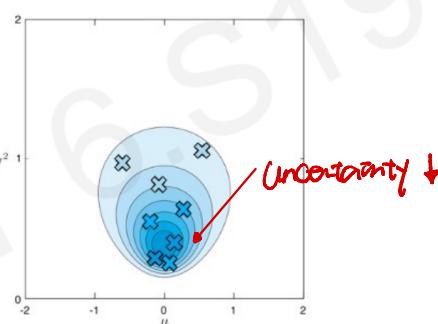
Estimating Epistemic Uncertainty : Reconstruction error

→ 학습한 주어진 데이터가 Generative model Input으로 되었을 때의 Reconstruction error ↑
→ Uncertainty

Estimating Epistemic Uncertainty : Evidential Deep Learning

Estimating Epistemic Uncertainty: Evidential Deep Learning

Learn the variance **directly**, without sampling by placing priors on the distribution that the evidence comes from.



- 학습 데이터가 어떤 Distribution 으로부터 온다고 가정한다

- Model이 해당 데이터의 방법
Method parameters를 충족하도록 한다.

Using Risk-Awareness to Transform AI Workflows

