Explainable Machine Learning

Neural Network Interpretation

Shim Jaewoong

jaewoong@seoultech.ac.kr

Detecting Concepts

Kim, Been, et al. "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)." *International conference on machine learning*. PMLR, 2018.

Concept-based approach

- **Limitations of methods using feature attribution**
  - Features are not necessarily user-friendly in terms of interpretability
    - For example, the importance of a single pixel in an image usually does not convey much meaningful interpretation.
  - The expressiveness of a feature-based explanation is constrained by the number of features.

→ Concept-based approach

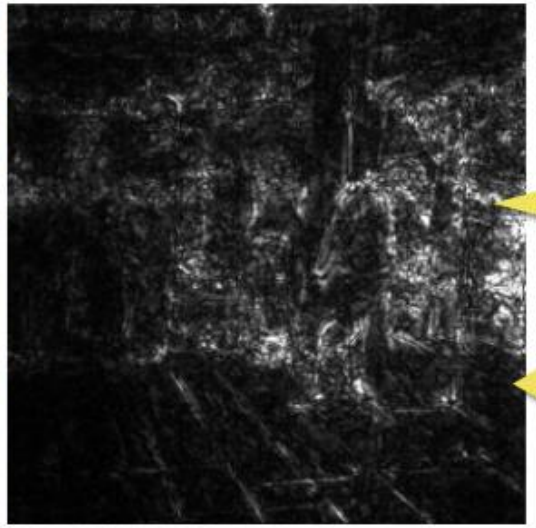
- A concept can be any abstraction, such as a color, an object, or even an idea.

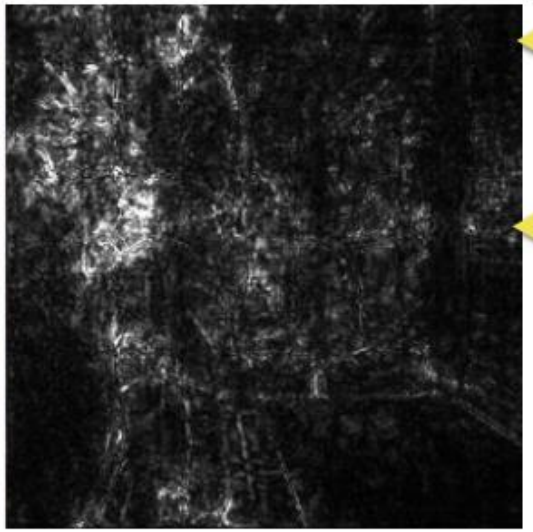
- Although a neural network **might not be explicitly trained with the given concept**, the concept-based approach detects that **concept embedded within the latent space** learned by the network.

TCAV: Testing with Concept Activation Vectors

https://beenkim.github.io/slides/TCAV_ICML_pdf.pdf

Saliency maps





Were there more pixels on the cash machine than on the person?

Did the 'human' concept matter?
Did the 'glasses' or 'paper' matter?

Which concept mattered more?

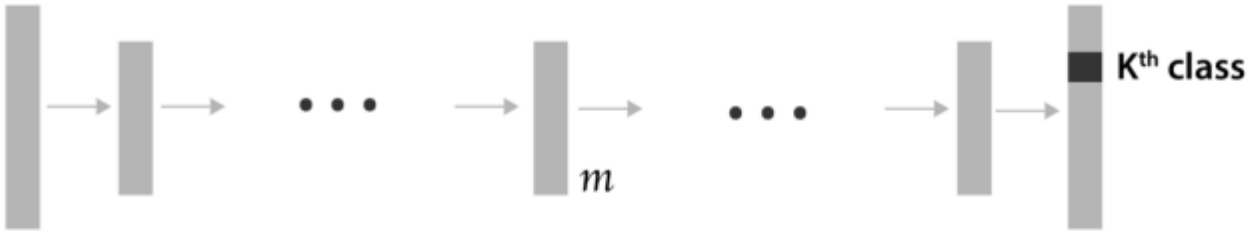
Is this true for all other cash machine predictions?

Oh no! I can't express these concepts as pixels!!
They weren't my input features either!

TCAV: Testing with Concept Activation Vectors

- Goal of **TCAV: Testing with Concept Activation Vectors**

  provides useful **global interpretation** for a model's overall behavior.



Quantitative explanation: how much a concept (e.g., gender, race) was important for a prediction in a trained model.

…even if the concept was not part of the training.

TCAV: Testing with Concept Activation Vectors

- Goal of **TCAV: Testing with Concept Activation Vectors**

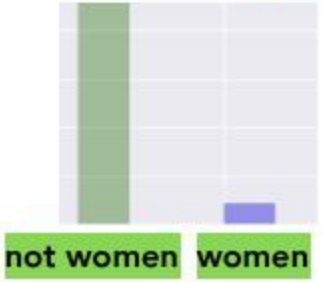
A trained machine learning model (e.g., neural network)

$p(z)$

Doctor-ness

Was gender concept important to this doctor image classifier?

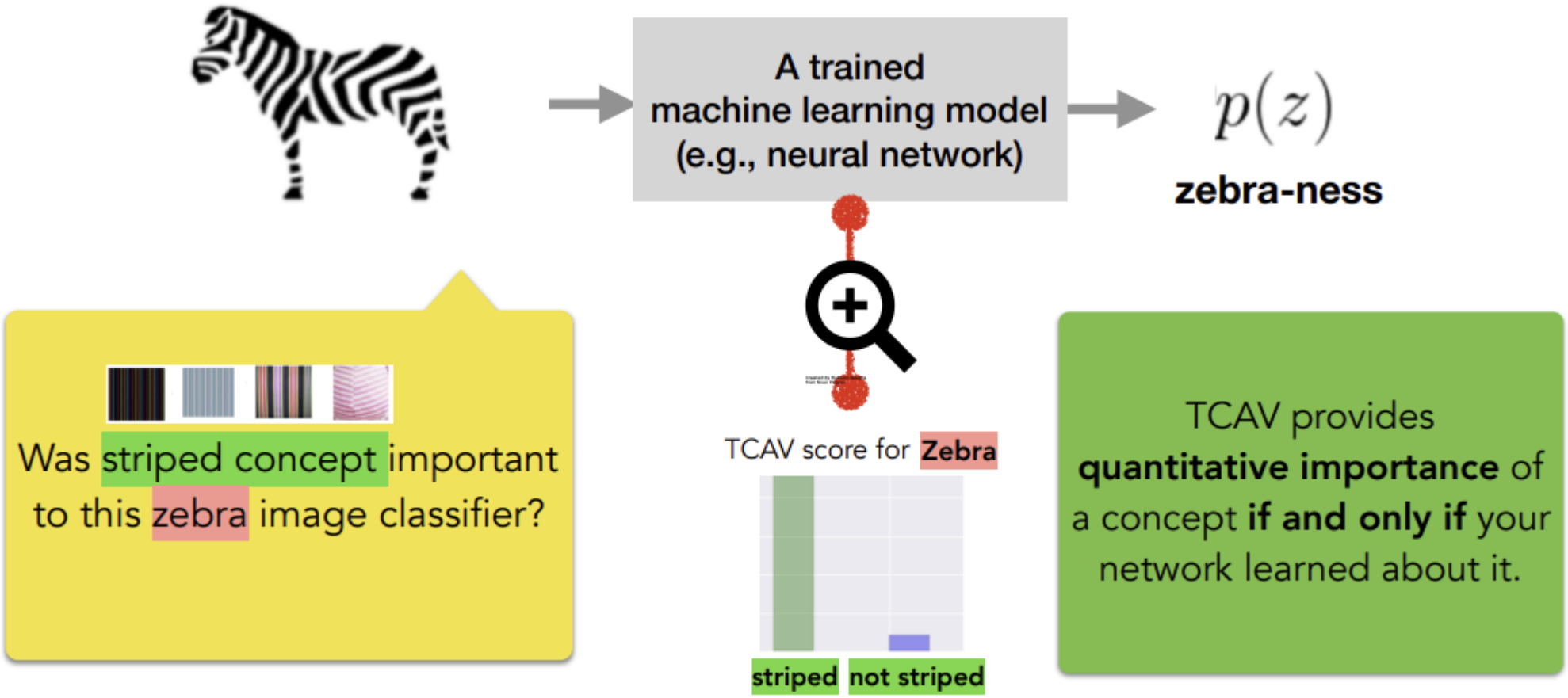
TCAV score for **Doctor**



TCAV provides **quantitative importance** of a concept **if and only if** your network learned about it.

TCAV: Testing with Concept Activation Vectors

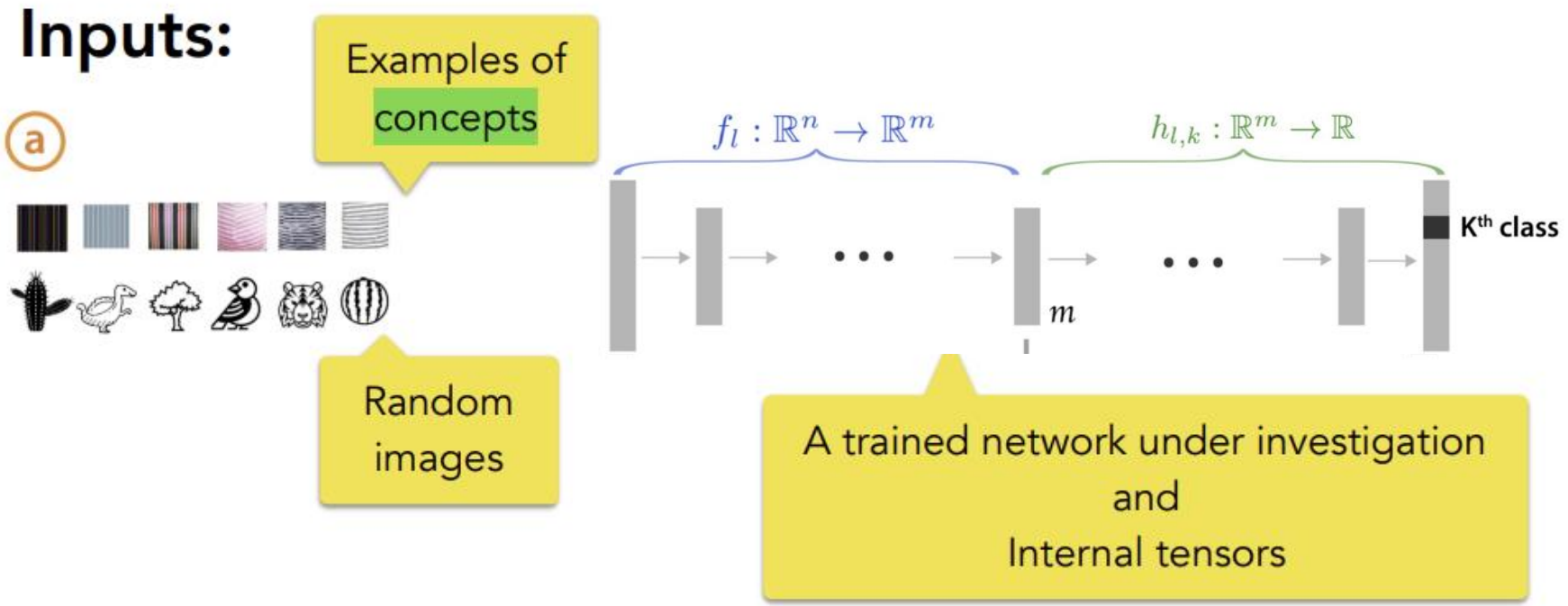
- Goal of **TCAV: Testing with Concept Activation Vectors**



TCAV: Testing with Concept Activation Vectors

1. How to define concepts?

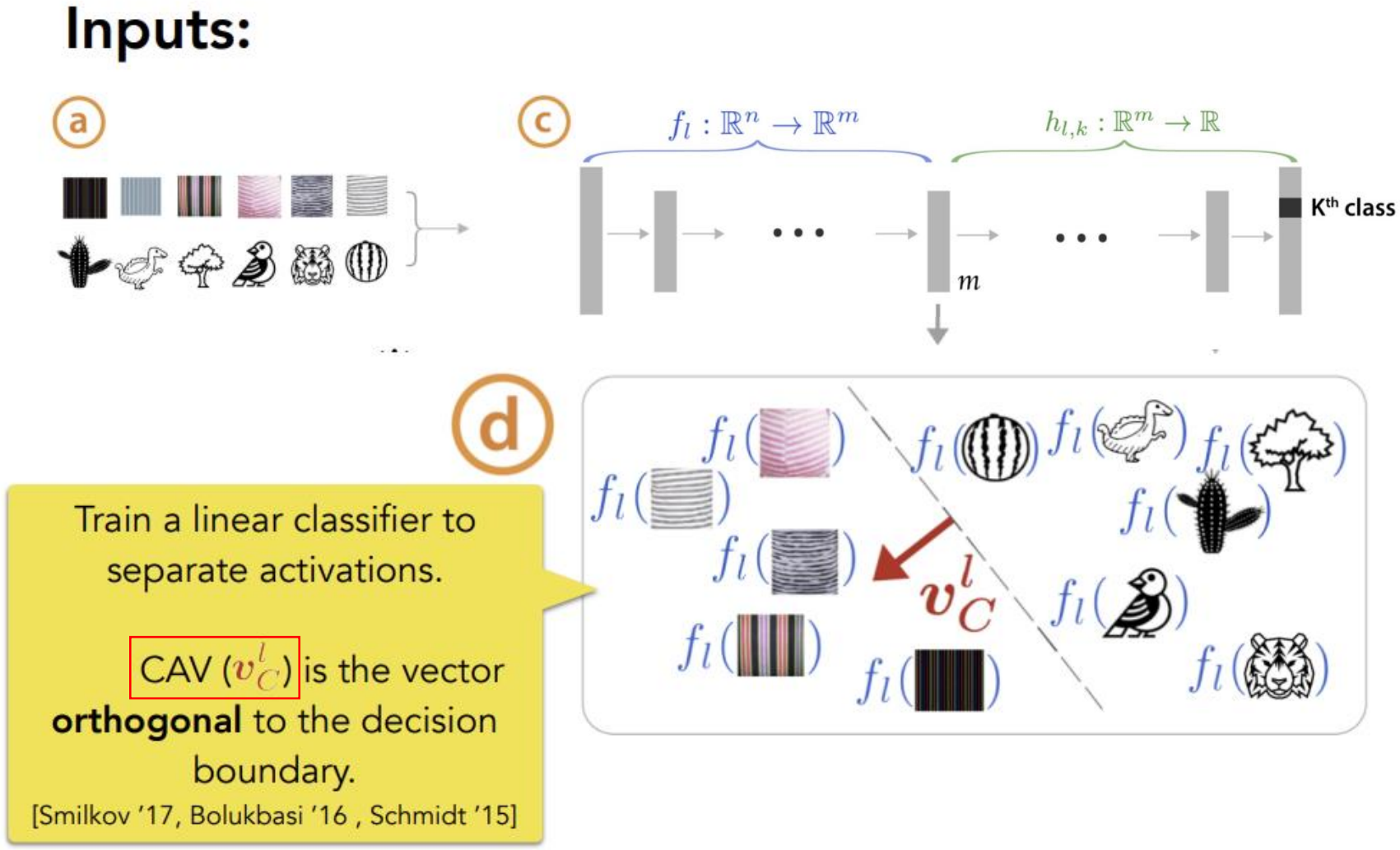
– Concept Activation Vector (CAV)



TCAV: Testing with Concept Activation Vectors

1. How to define concepts?

– Concept Activation Vector (CAV)

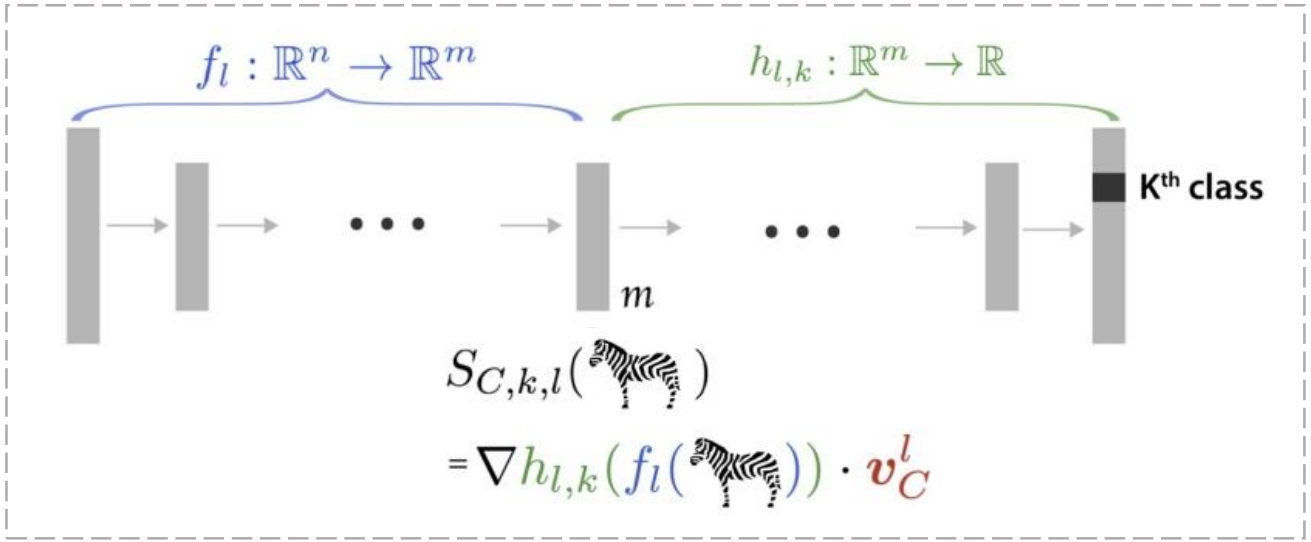


TCAV: Testing with Concept Activation Vectors

2. How are the CAVs useful to get explanations?

– Derivative with CAV to get prediction sensitivity

sensitivity of ML predictions to changes in inputs towards the direction of a concept



$$S_{C,k,l}(\boldsymbol{x}) = \lim_{\epsilon \to 0} \frac{h_{l,k}(f_l(\boldsymbol{x}) + \epsilon \boldsymbol{v}_C^l) - h_{l,k}(f_l(\boldsymbol{x}))}{\epsilon}$$
$$= \nabla h_{l,k}(f_l(\boldsymbol{x})) \cdot \boldsymbol{v}_C^l, \quad (1)$$

Same direction?

All inputs X with the given label

$S_{C,k,l}($ $)$

$S_{C,k,l}($ $)$

$S_{C,k,l}($ $)$

$S_{C,k,l}($ $)$

$$\text{TCAVQ}_{C,k,l} = \frac{|\{\boldsymbol{x} \in X_k : S_{C,k,l}(\boldsymbol{x}) > 0\}|}{|X_k|}$$

TCAV: Testing with Concept Activation Vectors

3. CAV validation

- Quantitative validation: Guarding against spurious CAV
  - Did my CAVs returned high sensitivity by chance? -> **significance test**

Learn many CAVs using different sets of random images

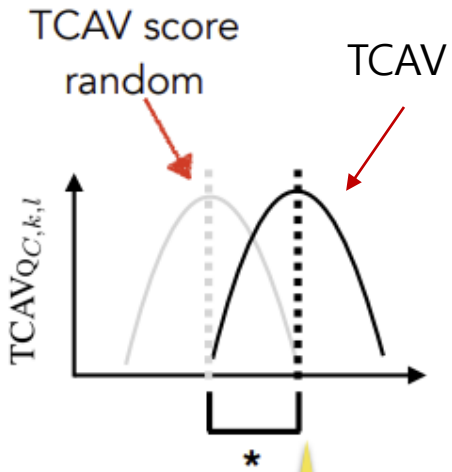
Zebra

$\rightarrow \text{TCAV}_{Q_{C,k,l}}$:

$\rightarrow \text{TCAV}_{Q_{C,k,l}}$:

$\rightarrow \text{TCAV}_{Q_{C,k,l}}$:

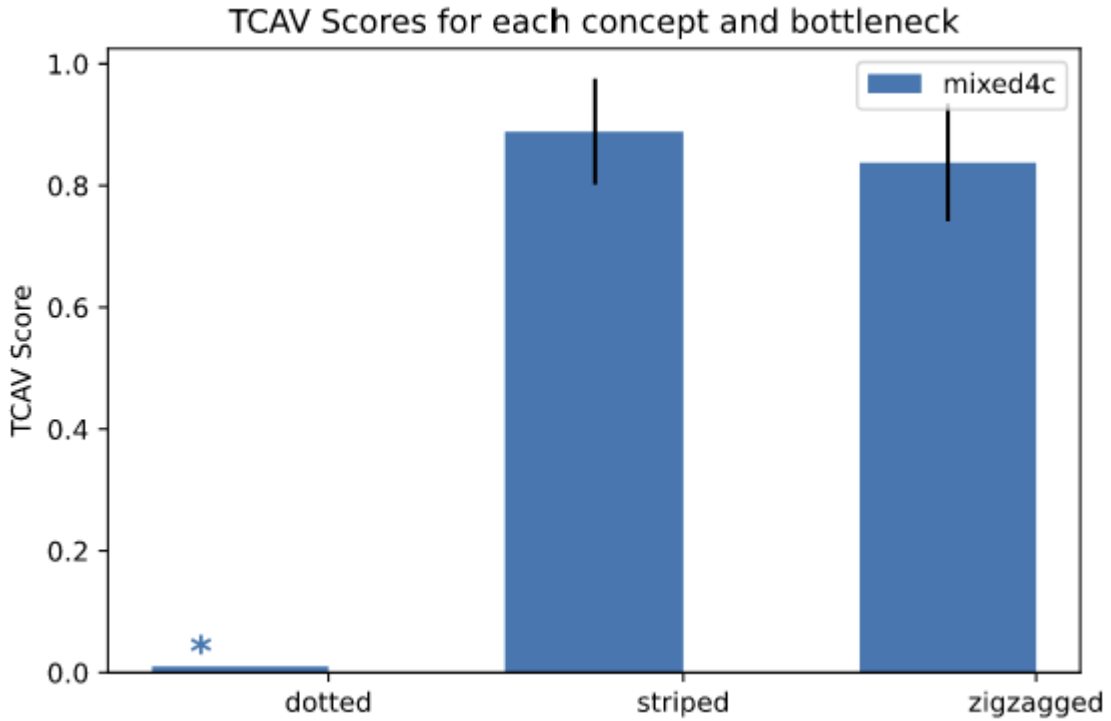
$\rightarrow \text{TCAV}_{Q_{C,k,l}}$:



Check the distribution of $\text{TCAV}_{Q_{C,k,l}}$ is statistically different from random using t-test

TCAV: Testing with Concept Activation Vectors

- Example
  - InceptionV3 model trained using ImageNet data
  - targeted bottleneck layer: “mixed4c”.
  - Zebra class
  - 3 concepts: striped, zigzagged, dotted
  - 50 images for each concept or random dataset
  - 10 random datasets for statistical test



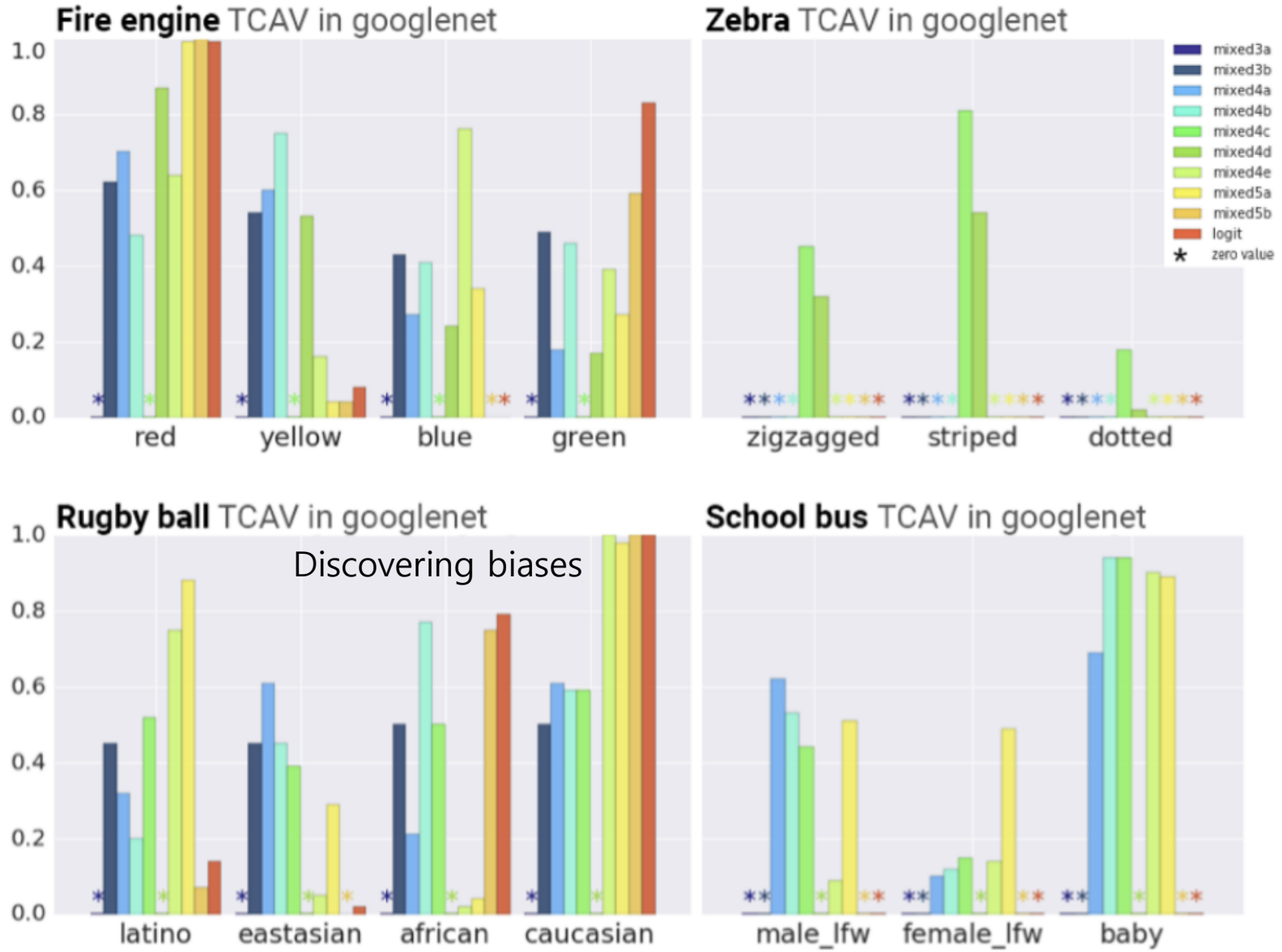
“dotted” has not passed the statistical significance test

“striped” and “zigzagged” have passed the test

→ useful for the model to identify “zebra” images

TCAV: Testing with Concept Activation Vectors

- Example



TCAV: Testing with Concept Activation Vectors

- Example

accuracy of more abstract concepts (e.g., objects) increases in higher layers of the network

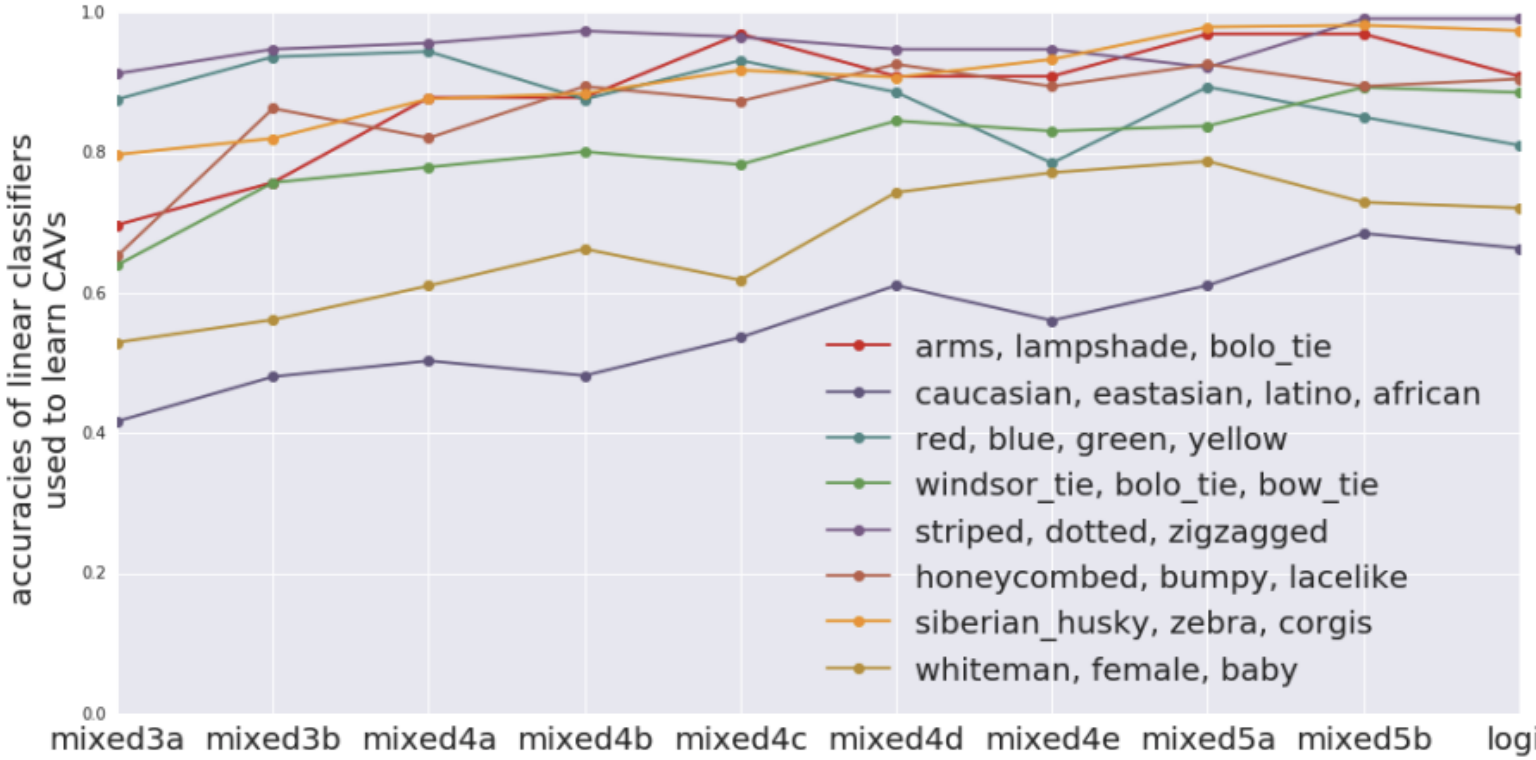


Figure 5. The accuracies of CAVs at each layer. Simple concepts (e.g., colors) achieve higher performance in lower-layers than more abstract or complex concepts (e.g. people, objects)

TCAV: Testing with Concept Activation Vectors

- **Advantages**
  - TCAV does not require users to have machine learning expertise
    - users are only required to collect data for training the concepts that they are interested in
  - Users can investigate any concept as long as the concept can be defined by its concept dataset.
    - If a domain expert understands the problem and concept very well, they can shape the concept dataset using more complicated data to generate a more fine-grained explanation.
  - TCAV generates **global explanations** that relate concepts to any class.
    - If a user can identify those ill-learned concepts, they can use the knowledge to **improve their model**.
- **Disadvantages**
  - **perform badly on shallower neural networks**.
  - requires **additional annotations** for concept datasets.
  - it is **difficult to apply to concepts that are too abstract or general**.
    - "happiness" ?
  - Only popular for image dataset.

Uncertainty Quantification
for neural network

https://www.cs.ox.ac.uk/people/yarin.gal/website/blog_2248.html

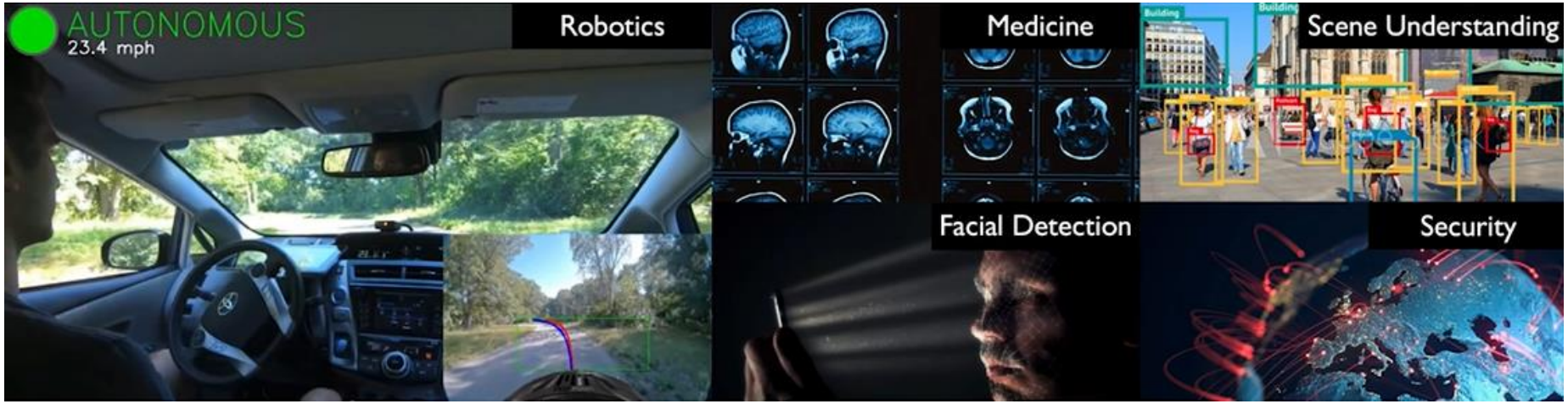
https://neurips.cc/media/neurips-2020/Slides/16649.pdf

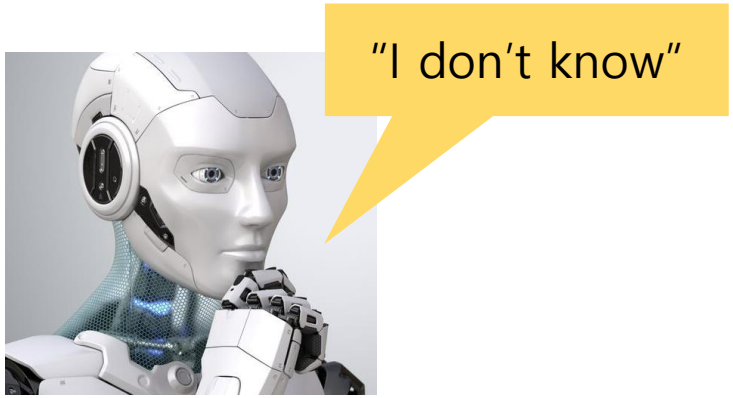
https://youtu.be/oCVEFv5P088

https://youtu.be/toTcf7tZK8c

The Importance of Knowing What We Don't Know

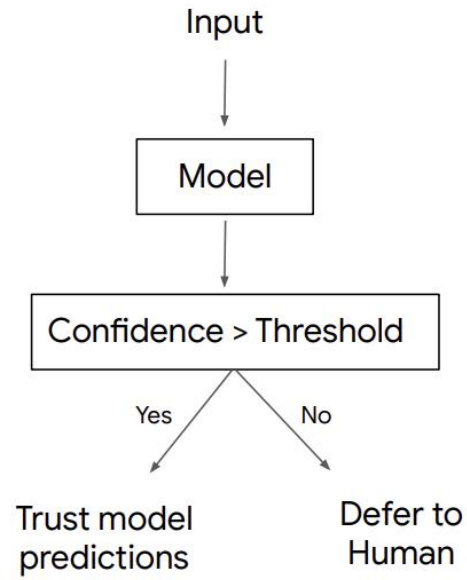
- For critical decisions…
  - Mistakes are not allowed.





AI safety
- Diagnosing a patient
- Autonomous vehicles
- Frequency trading
- …

*Use model **uncertainty** to decide when to trust the model or to defer to a human*

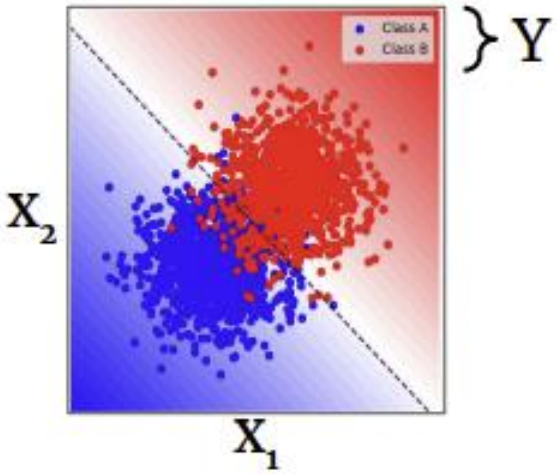


Uncertainty

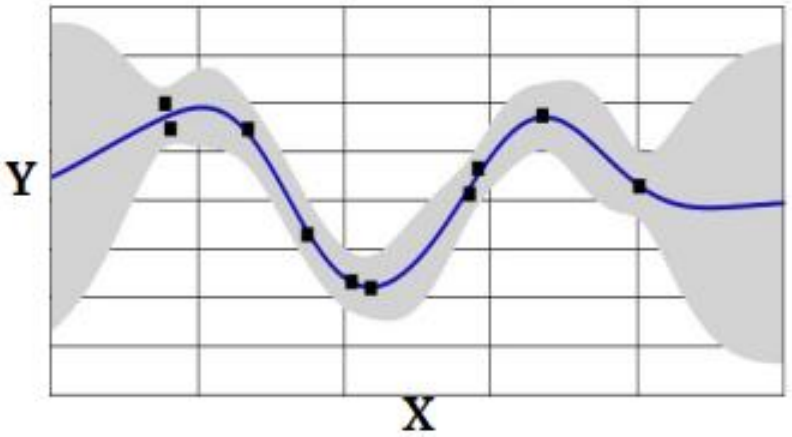
Return a distribution over predictions rather than a single prediction.

- ***Classification***: Output label along with its confidence.
- ***Regression***: Output mean along with its variance.

Good uncertainty estimates quantify ***when we can trust the model's predictions***.

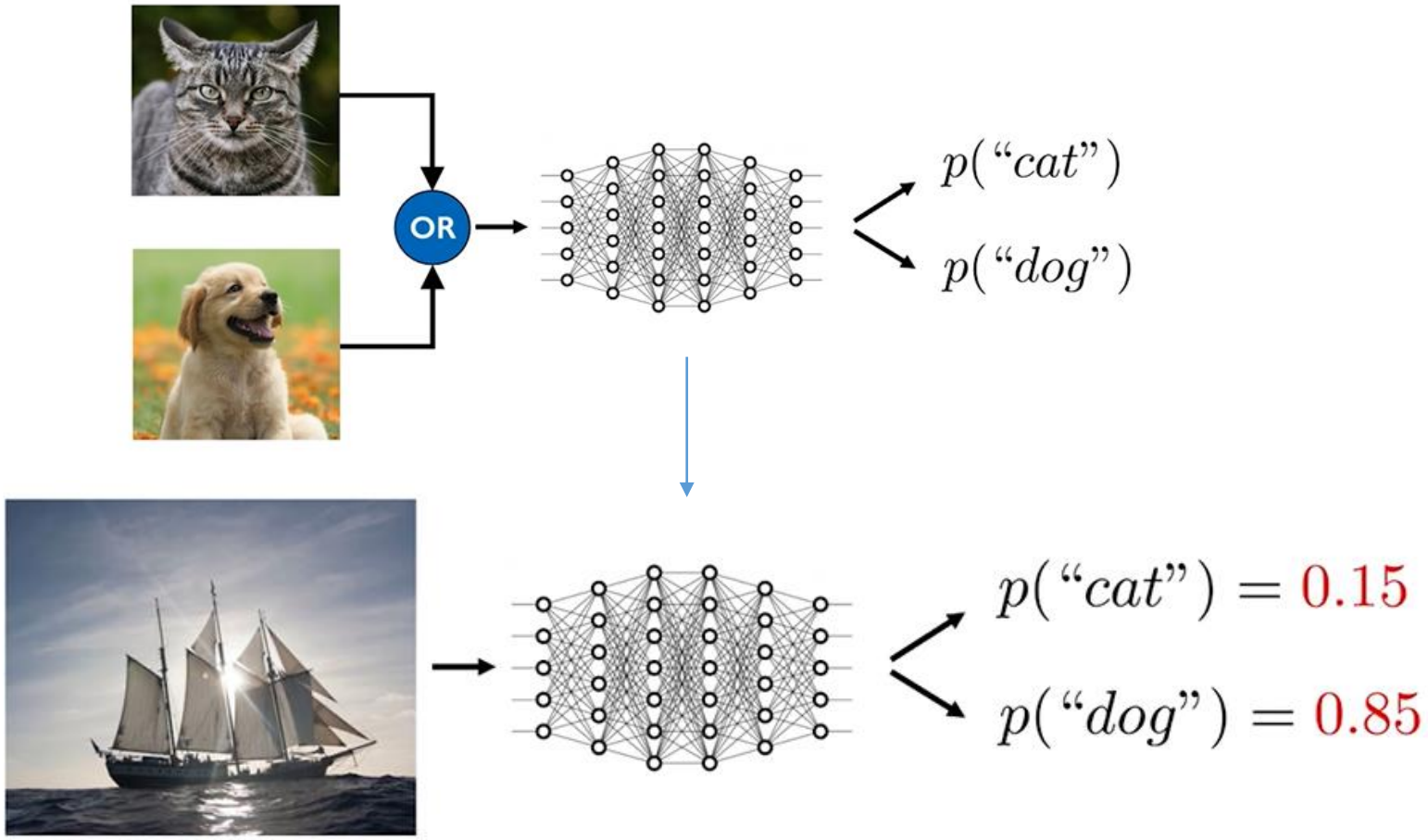


$$p(\mathbf{y}|\mathbf{x})$$



Neural networks do not know when they don't know

- Uncertainty using predicted probability ?
  - The output is unreliable if the input is unlike anything during training



⭐ $p(\text{“}cat\text{”}) + p(\text{“}dog\text{”}) = 1$ ⭐

Neural networks do not know when they don’t know

- Models assign high confidence predictions to OOD(Out-of-distribution) inputs

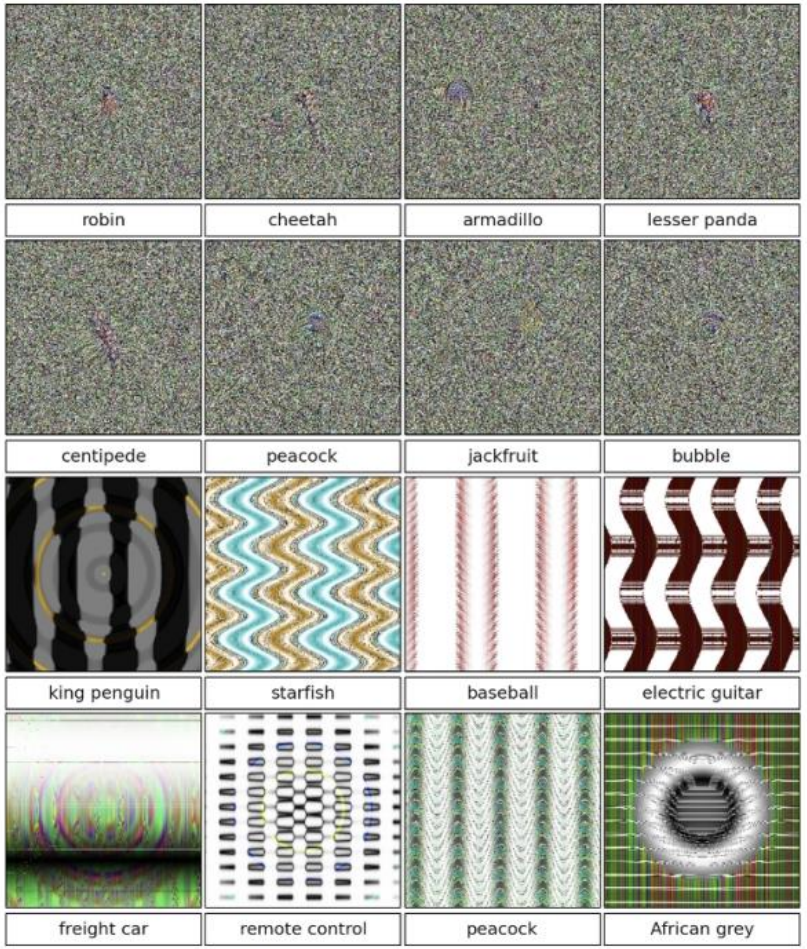
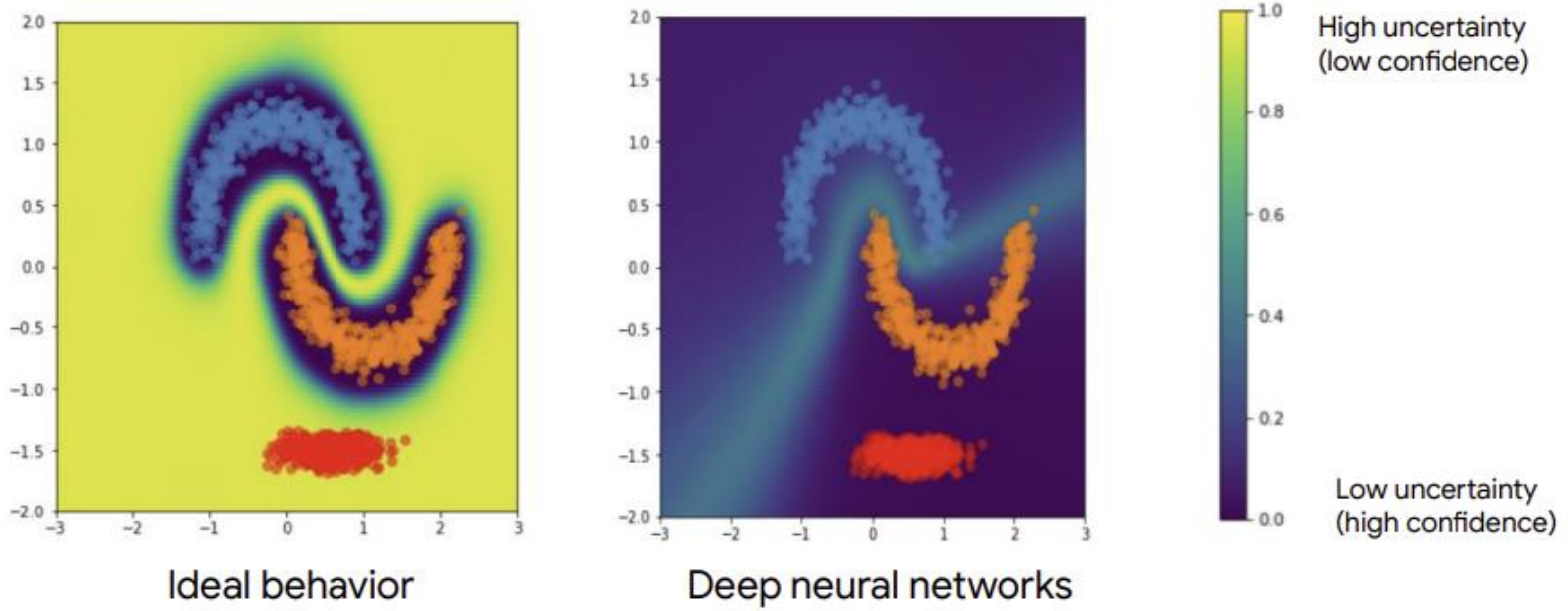


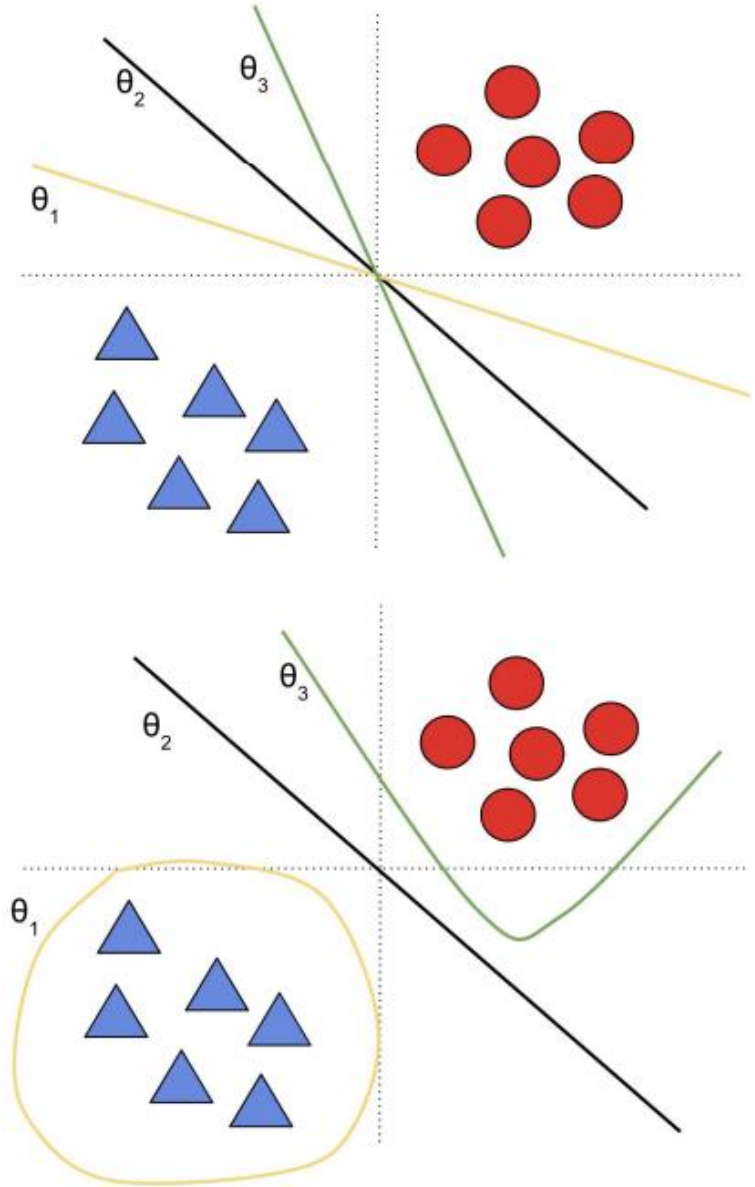
Figure 1. Evolved images that are unrecognizable to humans, but that state-of-the-art DNNs trained on ImageNet believe with $\geq$ 99.6% certainty to be a familiar object. This result highlights differences between how DNNs and humans recognize objects. Images are either directly (*top*) or indirectly (*bottom*) encoded.



Liu, Jeremiah, et al. "Simple and principled uncertainty estimation with deterministic deep learning via distance awareness." *Advances in Neural Information Processing Systems* 33 (2020): 7498-7512.

Sources of uncertainty

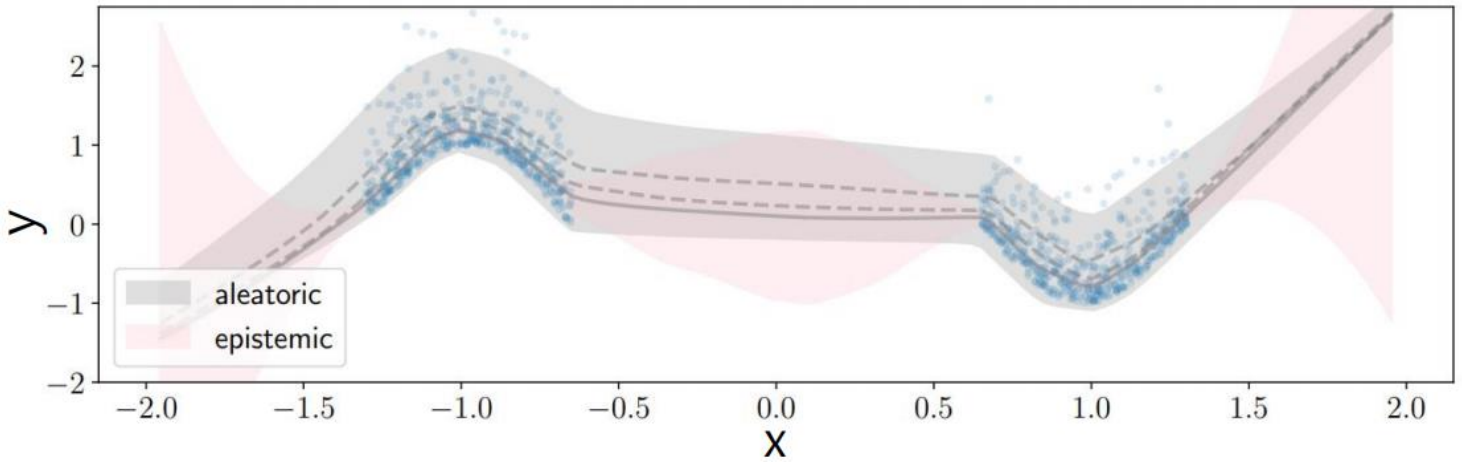
- Model uncertainty (epistemic uncertainty)
  - **Many models can fit the training data well**
  - Model uncertainty is "reducible"
    - Vanishes in the limit of infinite data
  - Either model parameters or model structure



Sources of uncertainty

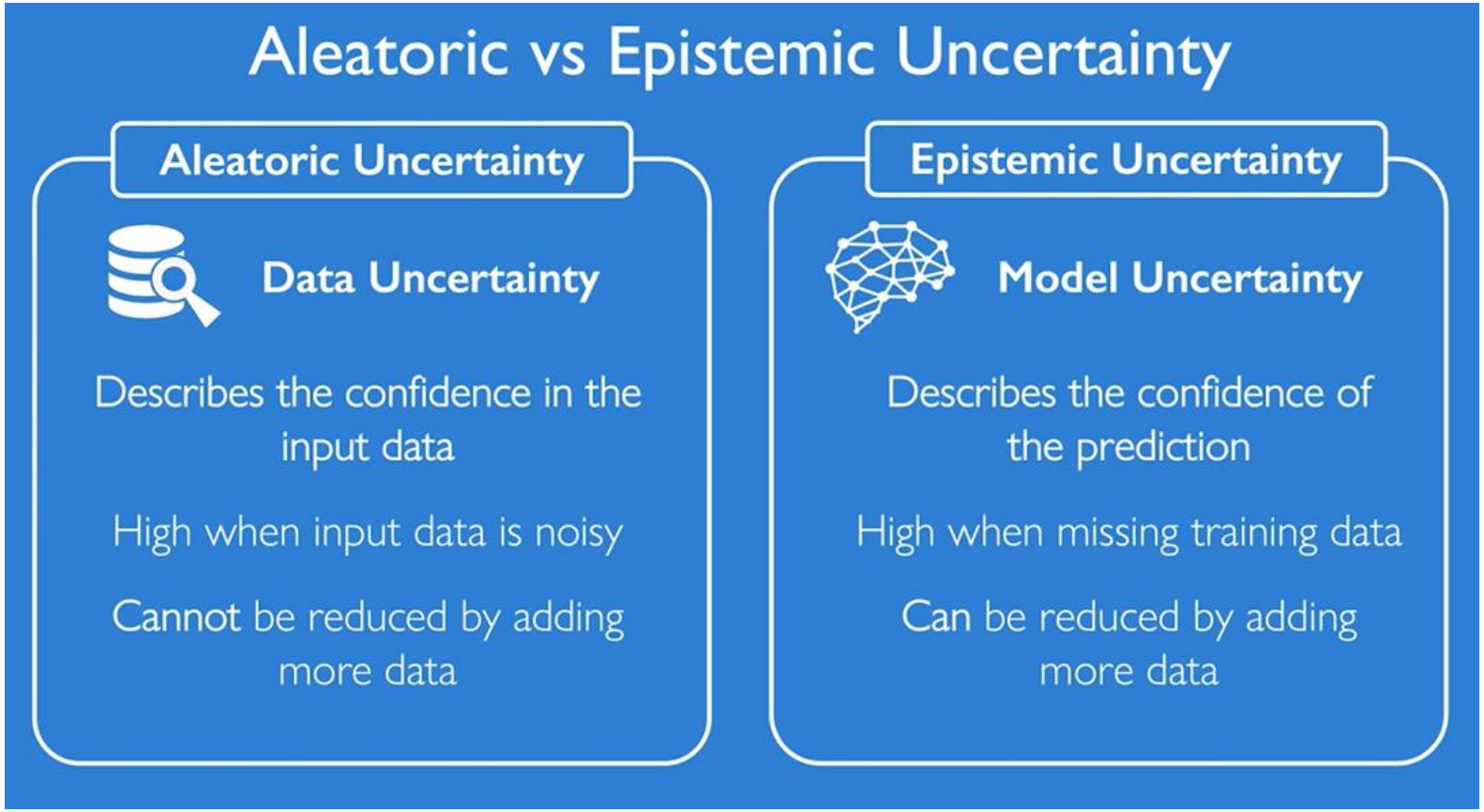
- Data uncertainty (aleatoric uncertainty)
  - Labeling noise (ex: human disagreement)
  - Measurement noise (ex: imprecise tools)
  - Missing data (ex: partially observed features, unobserved confounders)

  - Data uncertainty is "irreducible*"
    - Persists even in the limit of infinite data
    - *Could be reduced with additional features/views



<Example for regression problem>

- Model uncertainty (epistemic uncertainty) vs Data uncertainty (aleatoric uncertainty)



Bayesian Neural Networks

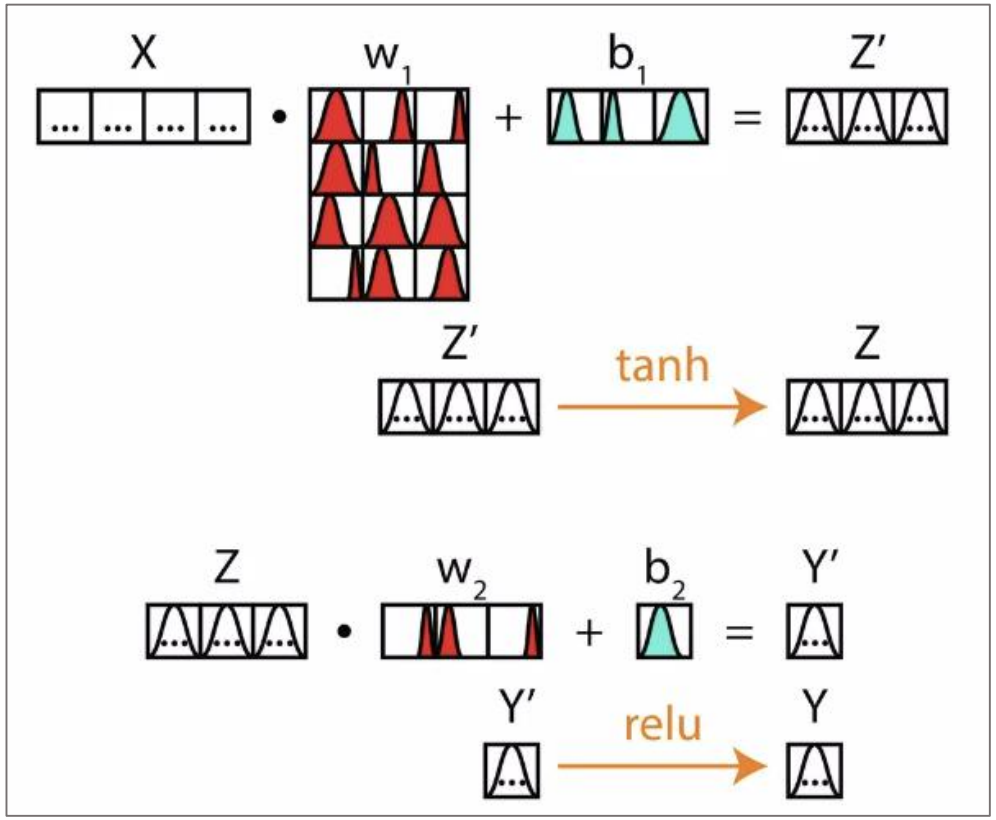
- Bayes theorem

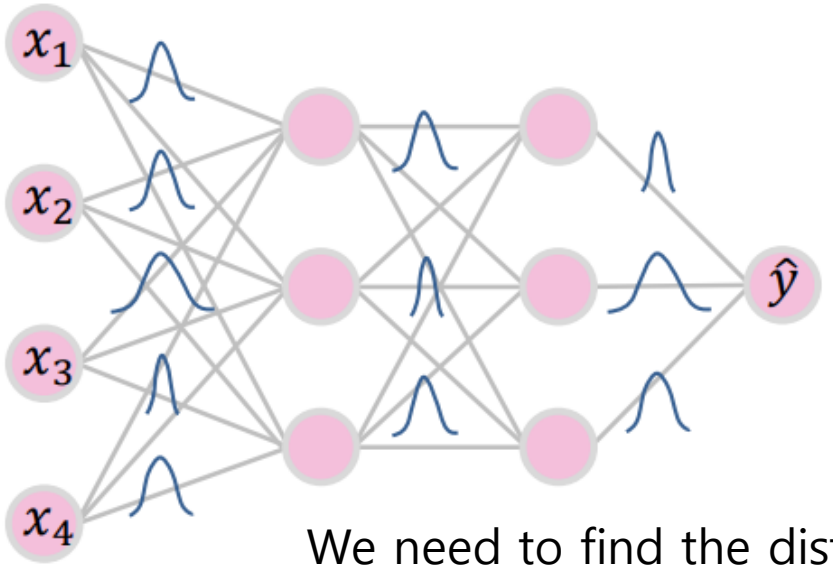
$$P(\theta|y) = \frac{P(y|\theta).P(\theta)}{P(y)}$$

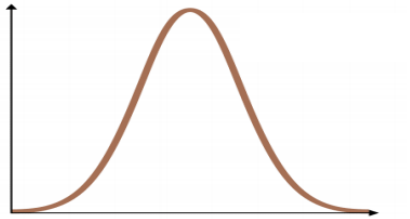
Frequentist	Bayesian
• **Data** is considered **random** • **Model** parameters are **fixed** • **Probabilities** are fundamentally related to frequencies of events	• **Data** is considered **fixed** • **Model** parameters are "*random*" (conditioned to observations - sampled from distribution) • **Probabilities** are fundamentally related to their own knowledge about an event

Bayesian Neural Networks

- Bayesian model (vs Standard model)
  - Each weight w is a distribution (instead of single point estimate)







We need to find the distribution of w

Deterministic neural networks (NNs) learn a fixed set of weights, $\boldsymbol{W}$

Bayesian neural networks aim to learn a posterior over weights, $P(\boldsymbol{W}|\boldsymbol{X},\boldsymbol{Y})$

Bayesian Neural Networks

- Bayesian model
  - In practice, posterior is **intractable** to compute analytically
  - Approximating the posterior
    - Variational inference
    - MCMC (Markov Chain Monte Carlo)
    - …
  - Computation is too expensive

Monte Carlo Dropout

Gal, Yarin, and Zoubin Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." *international conference on machine learning*. PMLR, 2016.

Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning

Yarin Gal YG279@CAM.AC.UK
Zoubin Ghahramani ZG201@CAM.AC.UK
University of Cambridge

Abstract

Deep learning tools have gained tremendous attention in applied machine learning. However such tools for regression and classification do

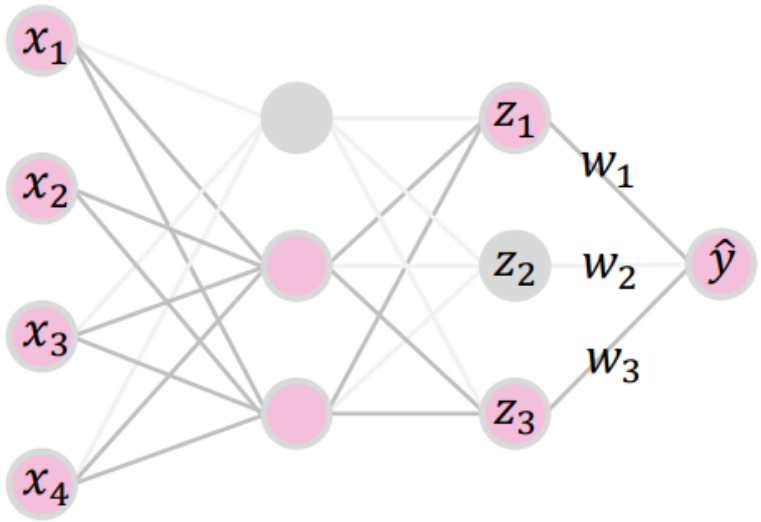
mow & Marks, 2015; Nuzzo, 2014), new needs arise from deep learning tools.

Standard deep learning tools for regression and classification do not capture model uncertainty. In classification,

Monte Carlo Dropout

- Dropout
  - Regularization method
  - To prevent overfitting

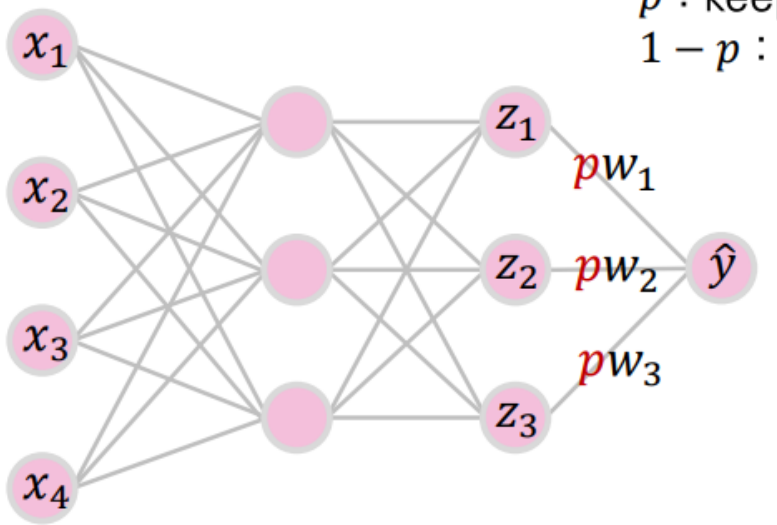
Training Phase



$$\hat{y} = z_1 w_1 + z_2 w_2 + z_3 w_3$$

Testing Phase

p : keep probability
$1 - p$: dropout probability

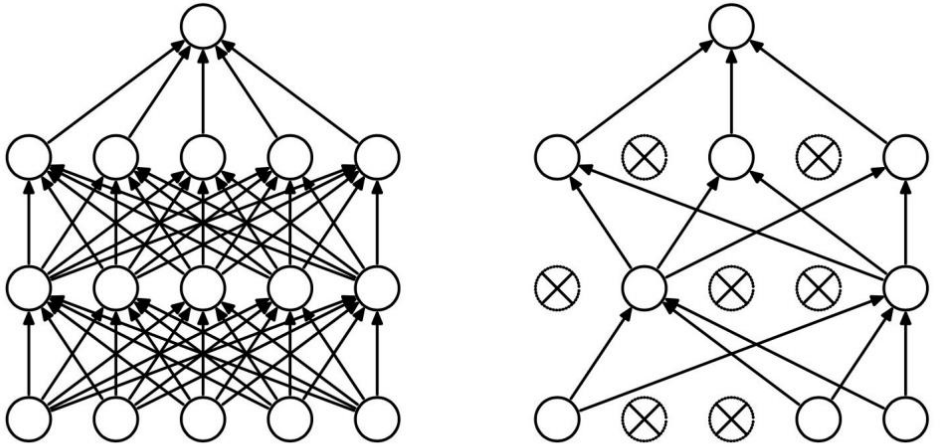


$$\hat{y} = pz_1 w_1 + pz_2 w_2 + pz_3 w_3$$

Monte Carlo Dropout

- Dropout
  - Model trained via dropout with L2 regularization is approximation of BNN

$$\mathcal{L}_{\text{dropout}} := \frac{1}{N}\sum_{i=1}^{N} E(\mathbf{y}_i, \widehat{\mathbf{y}}_i) + \lambda \sum_{i=1}^{L} \left(||\mathbf{W}_i||_2^2 + ||\mathbf{b}_i||_2^2 \right).$$



(a) Standard Neural Net

(b) After applying dropout.

Monte Carlo Dropout

- MC Dropout
  - Keep dropout even in test phase
  - 동일한 input x에 대해서도 매 추론마다 다른 결과 산출
    - Predicted Y의 분포에서 샘플링하는 것과 동일

T stochastic forward passes through the network

$$\mathbb{E}_{q(\mathbf{y}^*|\mathbf{x}^*)}(\mathbf{y}^*) \approx \frac{1}{T}\sum_{t=1}^{T}\hat{\mathbf{y}}^*(\mathbf{x}^*, \mathbf{W}_1^t, ..., \mathbf{W}_L^t)$$

prediction

$$\mathrm{Var}_{q(\mathbf{y}^*|\mathbf{x}^*)}(\mathbf{y}^*) \approx \tau^{-1}\mathbf{I}_D + \frac{1}{T}\sum_{t=1}^{T}\hat{\mathbf{y}}^*(\mathbf{x}^*, \mathbf{W}_1^t, ..., \mathbf{W}_L^t)^T\hat{\mathbf{y}}^*(\mathbf{x}^*, \mathbf{W}_1^t, ..., \mathbf{W}_L^t) - \mathbb{E}_{q(\mathbf{y}^*|\mathbf{x}^*)}(\mathbf{y}^*)^T\mathbb{E}_{q(\mathbf{y}^*|\mathbf{x}^*)}(\mathbf{y}^*)$$

uncertainty
(Epistemic uncertainty)

Aleatoric / Epistemic uncertainty

- Accurate understanding of aleatoric and epistemic uncertainties
  - Especially for computer vision task

Kendall, Alex, and Yarin Gal. "What uncertainties do we need in bayesian deep learning for computer vision?." *Advances in neural information processing systems* 30 (2017).

What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?

Alex Kendall
University of Cambridge
agk34@cam.ac.uk

Yarin Gal
University of Cambridge
yg279@cam.ac.uk

Abstract

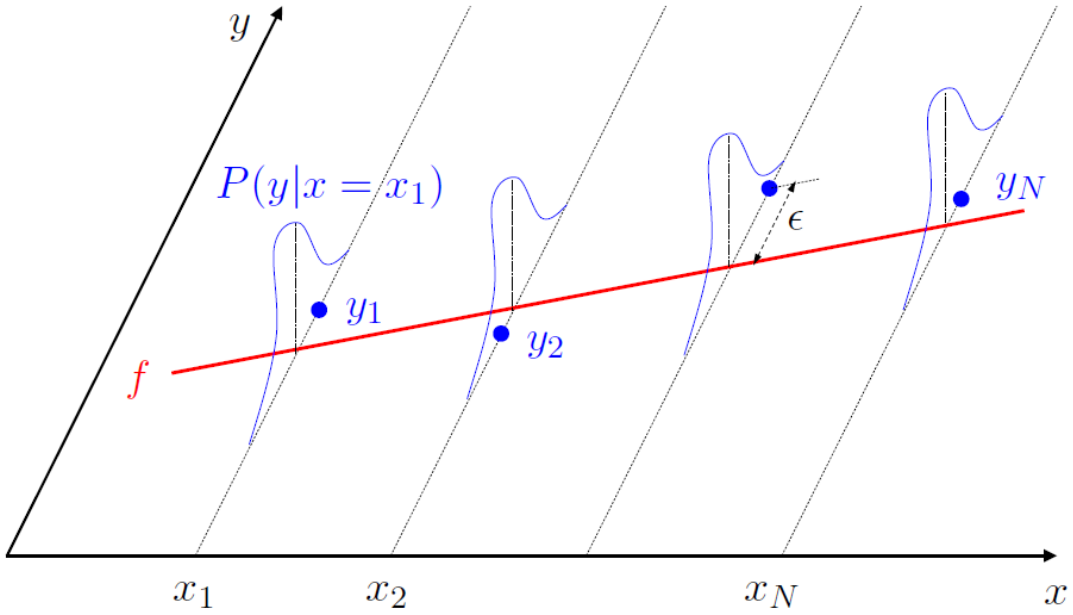
There are two major types of uncertainty one can model. *Aleatoric* uncertainty captures noise inherent in the observations. On the other hand, *epistemic* uncertainty accounts for uncertainty in the model – uncertainty which can be explained away given enough data. Traditionally it has been difficult to model epistemic uncertainty in computer vision, but with new Bayesian deep learning tools this

Recap: Linear Regression

- **Probabilistic Interpretation of Linear Regression**
  - Assume $y \sim \mathcal{N}(\hat{y}, \sigma^2), \hat{y} = \mathbf{w}^T \boldsymbol{x}$

$$p(y|\boldsymbol{x}; \mathbf{w}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y - \mathbf{w}^T \boldsymbol{x})^2}{2\sigma^2}\right)$$

p.d.f. of $N(\hat{y}, \sigma^2)$



Recap: Linear Regression

- **Maximum Likelihood Estimation** (with respect to $\hat{y}$)

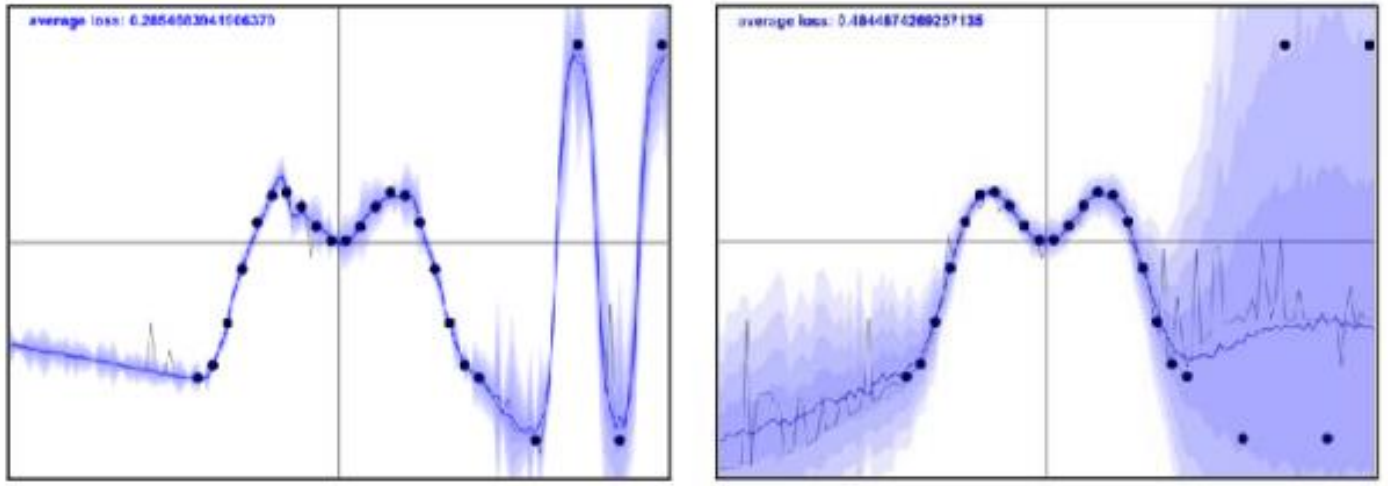
log-likelihood

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmax}} \prod_{(\boldsymbol{x}_i, y_i) \in D} p(y_i | \boldsymbol{x}_i; \mathbf{w}) = \underset{\mathbf{w}}{\operatorname{argmax}} \sum_{(\boldsymbol{x}_i, y_i) \in D} \log p(y_i | \boldsymbol{x}_i; \mathbf{w})$$

$$= \underset{\mathbf{w}}{\operatorname{argmax}} \left[-\frac{n}{2} \log 2\pi\sigma^2 - \frac{1}{2\sigma^2} \sum_{(\boldsymbol{x}_i, y_i) \in D} (y_i - \mathbf{w}^T \boldsymbol{x}_i)^2 \right]$$

Aleatoric / Epistemic uncertainty

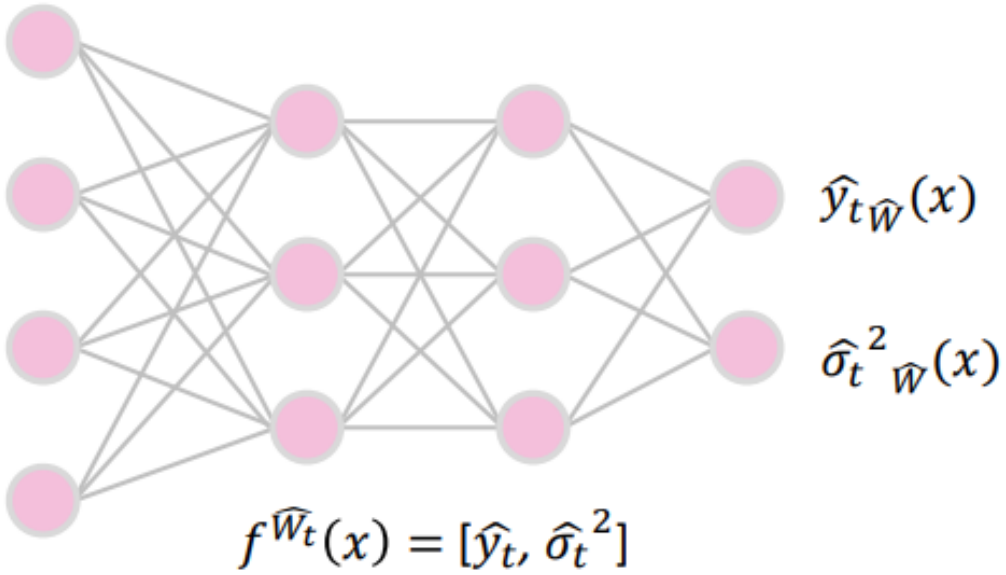
- Data uncertainty (aleatoric uncertainty)
  - Homoscedastic aleatoric
  - Heteroscedastic aleatoric



Aleatoric / Epistemic uncertainty

- To learn a **Heteroscedastic uncertainty** model, we simply can replace the loss function with the following:

$$Loss = ||y - \hat{y}||_2 \quad \longrightarrow \quad Loss = \frac{||y - \hat{y}||_2}{2\sigma^2} + \frac{1}{2}\log\sigma^2$$



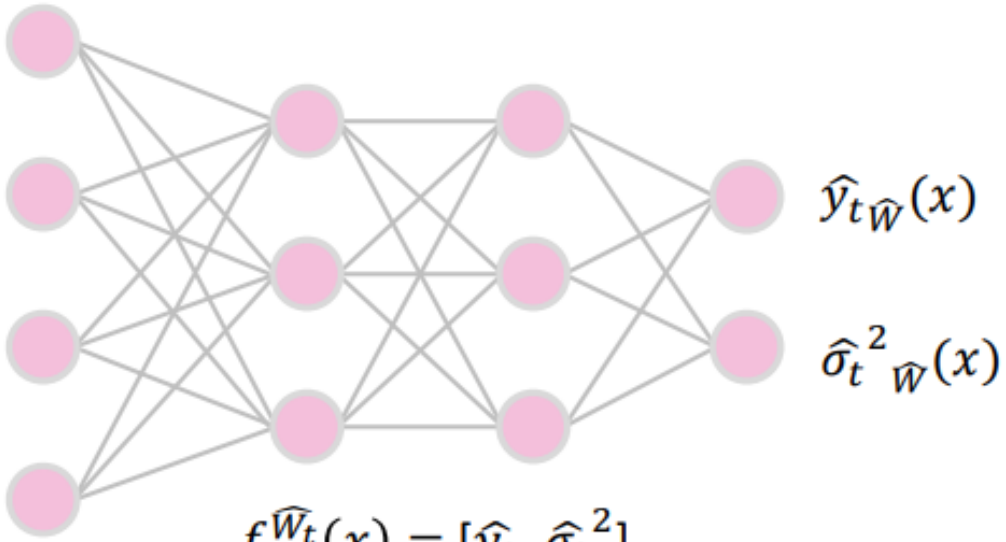
Loss function

$$\mathcal{L}_{BNN}(\theta) = \frac{1}{D}\sum_i \frac{1}{2}\hat{\sigma}_i^{-2}||\mathbf{y}_i - \hat{\mathbf{y}}_i||^2 + \frac{1}{2}\log\hat{\sigma}_i^2$$

- if the model predicts something very wrong, then it will be encouraged to attenuate the residual term, by increasing uncertainty σ2
- However, the logσ2 prevents the uncertainty term growing infinitely large.

Aleatoric / Epistemic uncertainty

- Uncertainty



Uncertainty

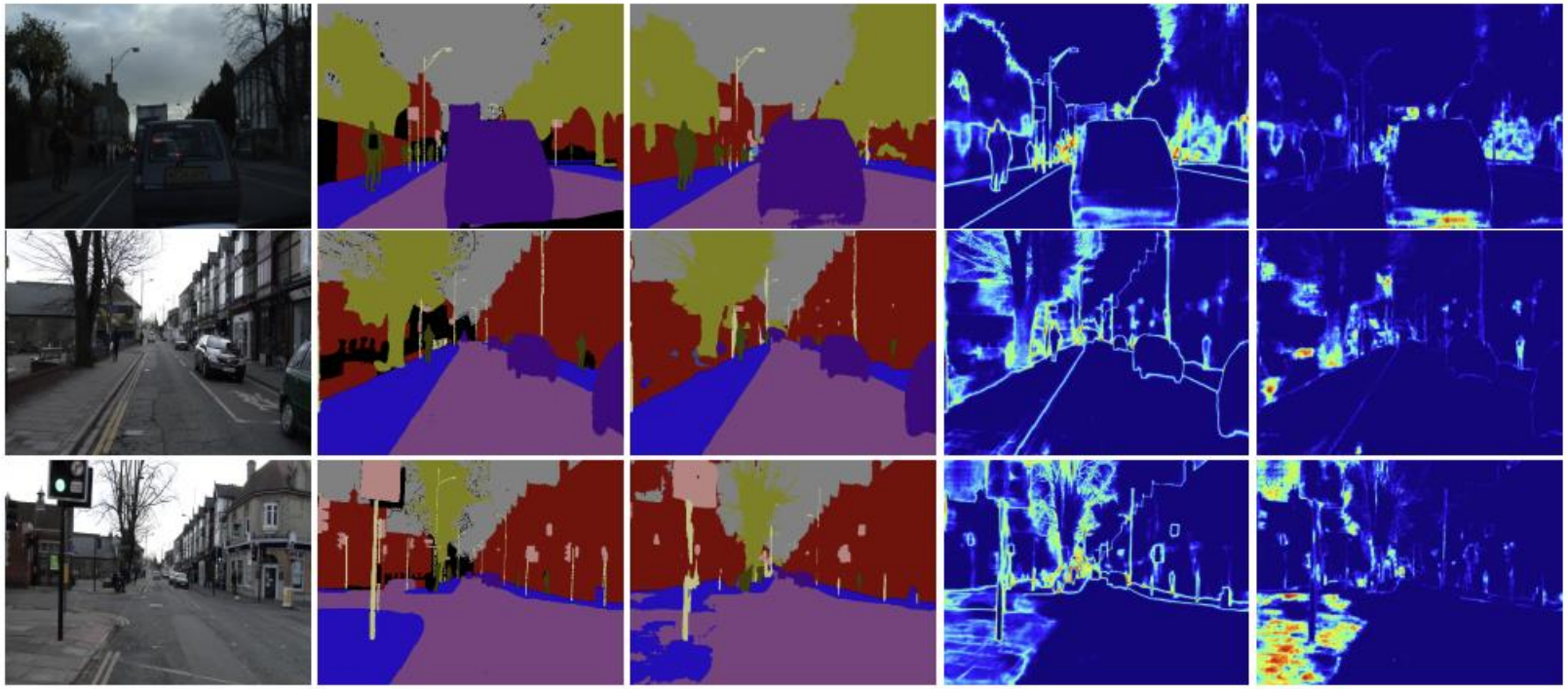
$$\mathrm{Var}(\mathbf{y}) \approx \frac{1}{T}\sum_{t=1}^{T}\hat{\mathbf{y}}_t^2 - \left(\frac{1}{T}\sum_{t=1}^{T}\hat{\mathbf{y}}_t\right)^2 + \frac{1}{T}\sum_{t=1}^{T}\hat{\sigma}_t^2$$

Epistemic uncertainty (MC dropout)

Aleatoric uncertainty (heteroscedastic)

Aleatoric / Epistemic uncertainty

- Semantic segmentation



(a) Input Image (b) Ground Truth (c) Semantic Segmentation (d) Aleatoric Uncertainty (e) Epistemic Uncertainty

Figure 1: **Illustrating the difference between aleatoric and epistemic uncertainty** for semantic segmentation on the CamVid dataset [8]. *Aleatoric* uncertainty captures noise inherent in the observations. In (d) our model exhibits increased aleatoric uncertainty on object boundaries and for objects far from the camera. *Epistemic* uncertainty accounts for our ignorance about which model generated our collected data. This is a notably different measure of uncertainty and in (e) our model exhibits increased epistemic uncertainty for semantically and visually challenging pixels. The bottom row shows a failure case of the segmentation model when the model fails to segment the footpath due to increased epistemic uncertainty, but not aleatoric uncertainty.

Aleatoric / Epistemic uncertainty

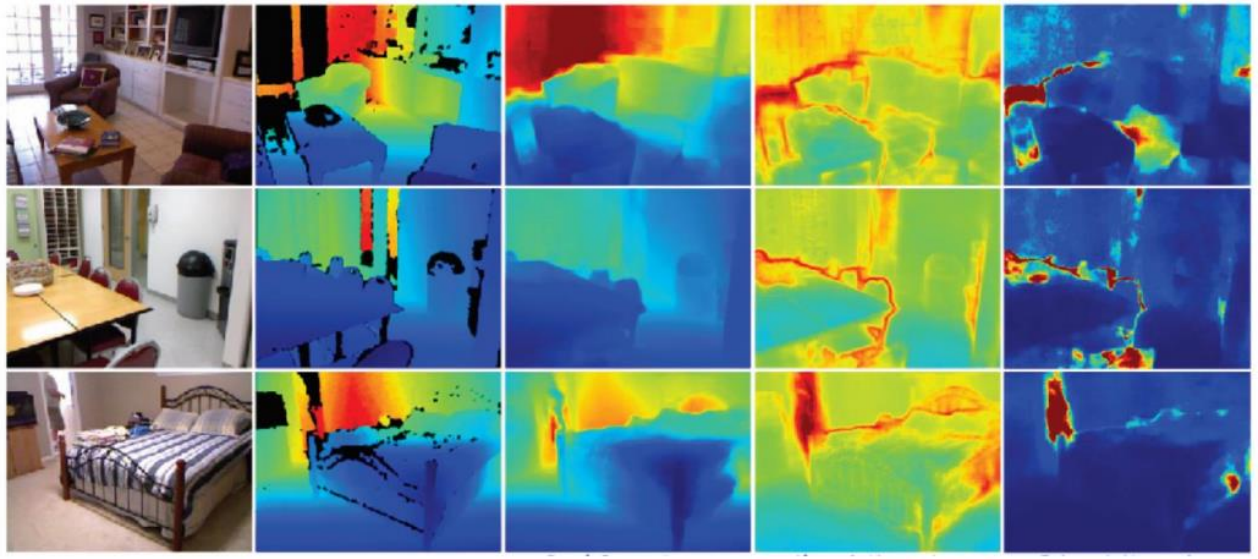
- Depth regression

Training Data	Testing Data	Aleatoric Variance	Epistemic Variance
Trained on dataset #1	Tested on dataset #1	0.485	2.78
Trained on 25% dataset #1	Tested on dataset #1	0.506	7.73
Trained on dataset #1	Tested on dataset #2	0.461	4.87
Trained on 25% dataset #1	Tested on dataset #2	0.388	15.0

https://alexgkendall.com/computer_vision/bayesian_deep_learning_for_safe_ai/

when we train on less data, or test on data which is significantly different from the training set, then our epistemic uncertainty increases drastically.

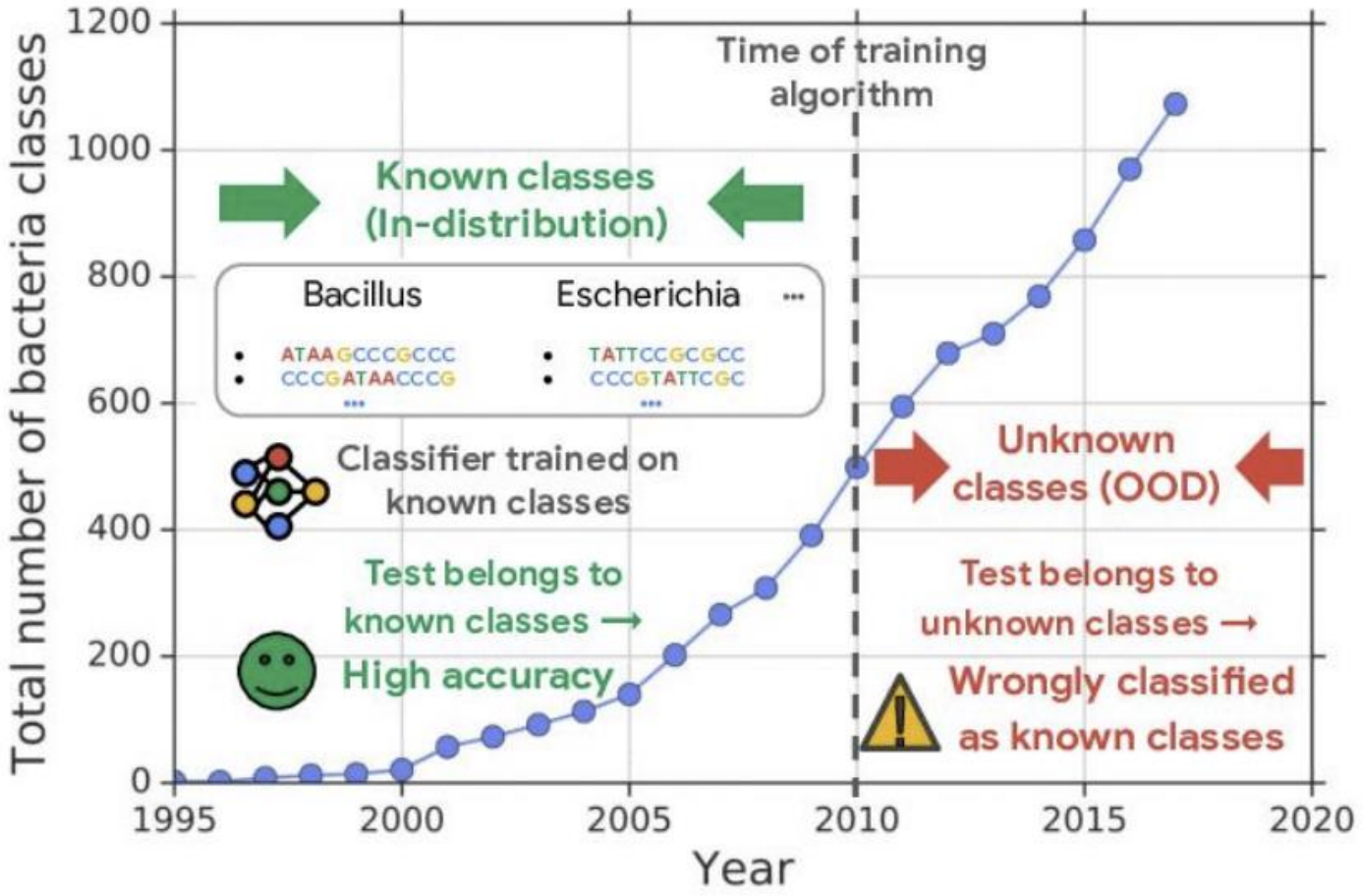
aleatoric uncertainty remains relatively constant – which it should – because it is tested on the same problem with the same sensor.



Applications

- OOD (Out-of-distribution) detection (~ Open set recognition)
  - detect inputs that do not belong to one of the known classes

- Example: Classification of genomic sequences
- High accuracy on known classes is not sufficient
- Need to be able to detect inputs that do not belong to one of the known classes



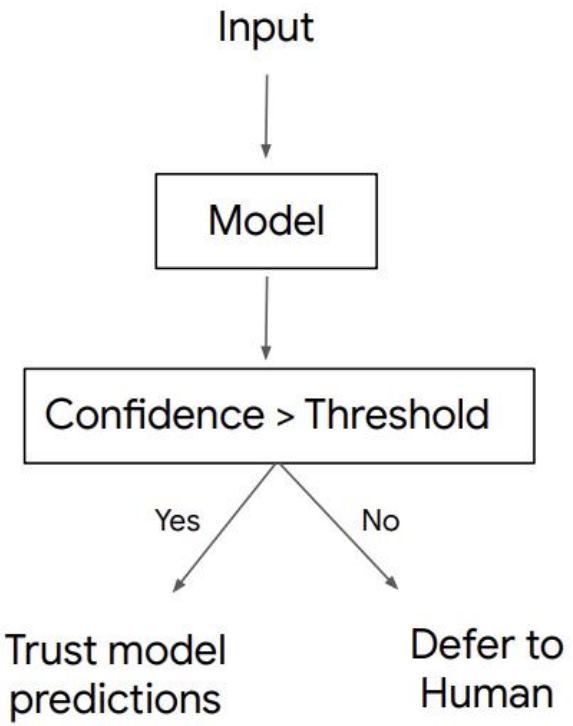
Applications

- Detecting out-of-scope utterances

Figure 1: Example exchanges between a user (blue, right side) and a task-driven dialog system for personal finance (grey, left side). The system correctly identifies the user's query in (1), but in (2) the user's query is mis-identified as in-scope, and the system gives an unrelated response. In (3) the user's query is correctly identified as out-of-scope and the system gives a fallback response.

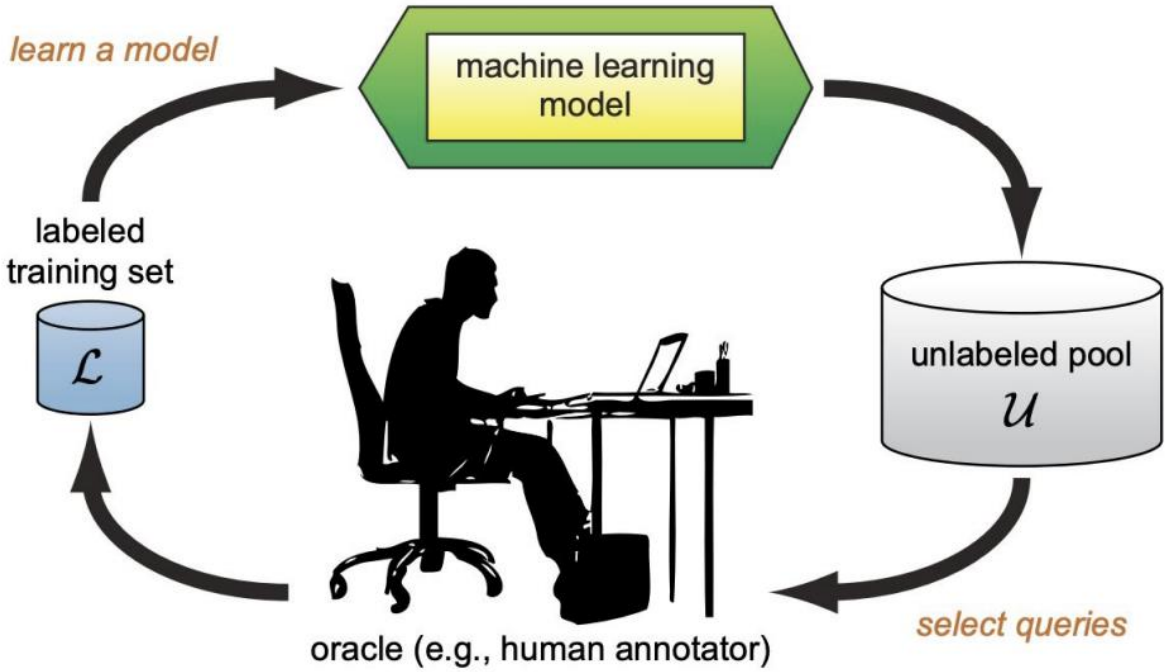
Applications

- Semi-automation
  - 불확실성 정량화의 적용 – 예측 불확실성 여부에 따른 예측모델의 선택적 활용
  - 완전 자동화 대비 부정확한 예측 발생의 위험성을 줄여줌
  - 예측모델의 불확실성이 높은 경우 → 사람 또는 장비를 이용하여 분석/검증
  - 예측모델의 불확실성이 낮은 경우 → 예측모델에 의한 예측 활용



Applications

- Active learning
  - Large amounts of training data, but few are labeled.
  - 목표: Unlabeled data에 적은 labeling 작업만으로 데이터의 정보 극대화
    - Unlabeled Data에 대한 추가 labeling이 가능한 경우?
      - Labeling 작업의 우선순위를 결정 – 현재 확보된 label이 설명하지 못하는 영역 우선
    - 주로, labeling 비용이 높은 경우 활용



Applications

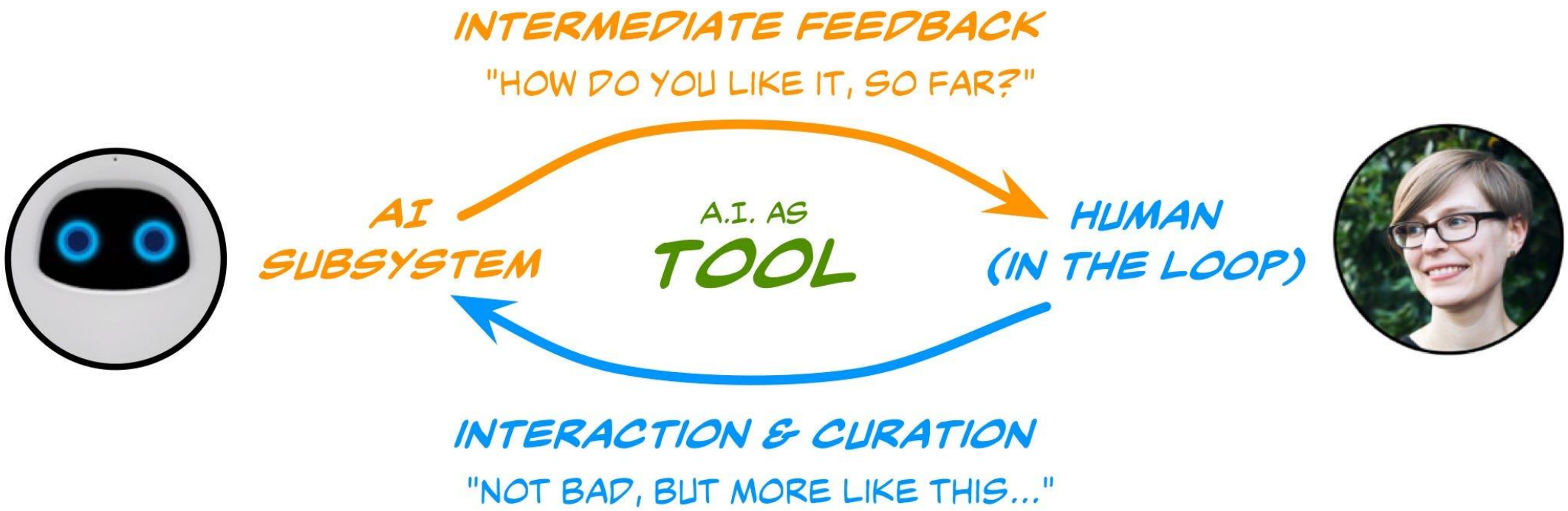
- Active learning
  - Active Learning을 위한 데이터 형태
  - Labeled Data의 정보를 활용하여 Unlabeled Data 중 Label을 확보할 데이터포인트 선택
  - 순차적 Labeling 수행을 통해 데이터의 정보 극대화

Example data for active learning

id	X_1	X_2	X_3	...	X_d	Y	Y	Y
1	x_{11}	x_{12}	x_{13}	...	x_{1d}	y_1	y_1	y_1
2	x_{21}	x_{22}	x_{23}	...	x_{2d}	-	-	-
3	x_{31}	x_{32}	x_{33}	...	x_{3d}	-	-	-
4	x_{41}	x_{42}	x_{43}	...	x_{4d}	-	-	-
5	x_{51}	x_{52}	x_{53}	...	x_{5d}	-	y_5	y_5
6	x_{61}	x_{62}	x_{63}	...	x_{6d}	-	-	y_6
7	x_{71}	x_{72}	x_{73}	...	x_{7d}	-	-	-
...	...	...	...	...	...	...	...	...

Human in the loop

- Cooperation of human and AI



https://hai.stanford.edu/news/humans-loop-design-interactive-ai-systems