Confidence Estimation for Trustworthy and Efficient Speech Systems

Part 1: Theory (85 min)

Break (10 min)

Part 2: Applications (85 min)

MADHAV lab
Machine Analysis of Data
for Human Audition and Vision

Confidence Estimation for Trustworthy and Efficient Speech Systems

Part 1: Theory

Vipul Arora

MADHAV lab

Machine Analysis of Data for Human Audition and Vision

Outline

- Introduction
- Assessment
- Why mis-calibration happens?
- Confidence calibration: post hoc methods and Bayesian methods
- Disentangling sources of uncertainty: Epistemic and Aleatoric

Future of AI

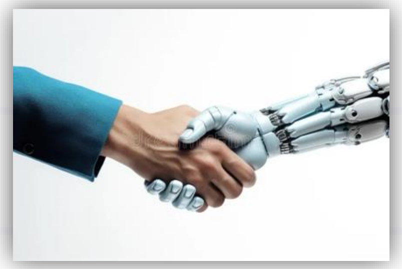
- Current AI

https://voicebot.ai/

Wikipedia.com

Wikipedia.com

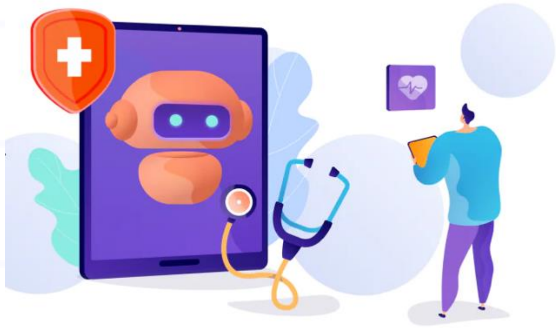
- Future AI
  - AGI
  - Human-Machine Collaboration



Dreamstime.com

Mistakes and Trust

Hello, I am giving the stock



Take paracetamol

Weather Tomorrow
8 AM

16°C ±1.2 °C

MADHAV

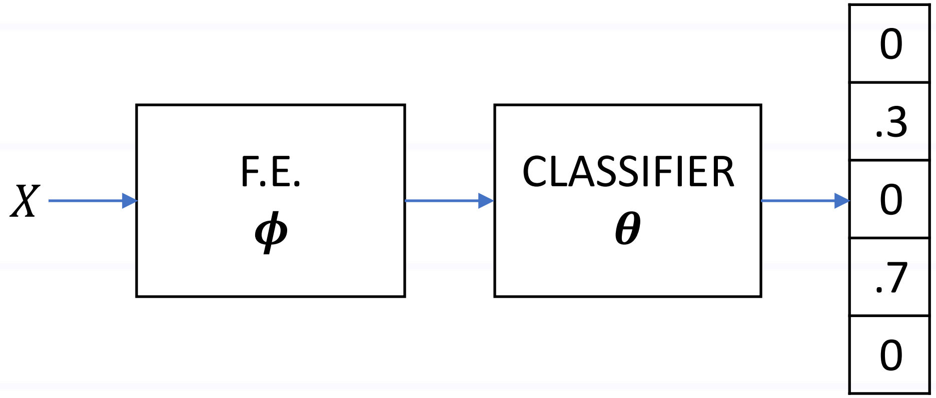
Need

- **Self-driving**: obstruction or not? Defer to human driver
- **Healthcare**: Operate or not? Defer to human doctor
- **Finance**: invest money or not? Defer to human expert
- **Screening**: accept or not? Defer to human examiner
- **Annotation**: annotate or not? Reduce manual effort

For Classification

Deep Networks

- Output class



$$y = \arg\max_j o_j$$

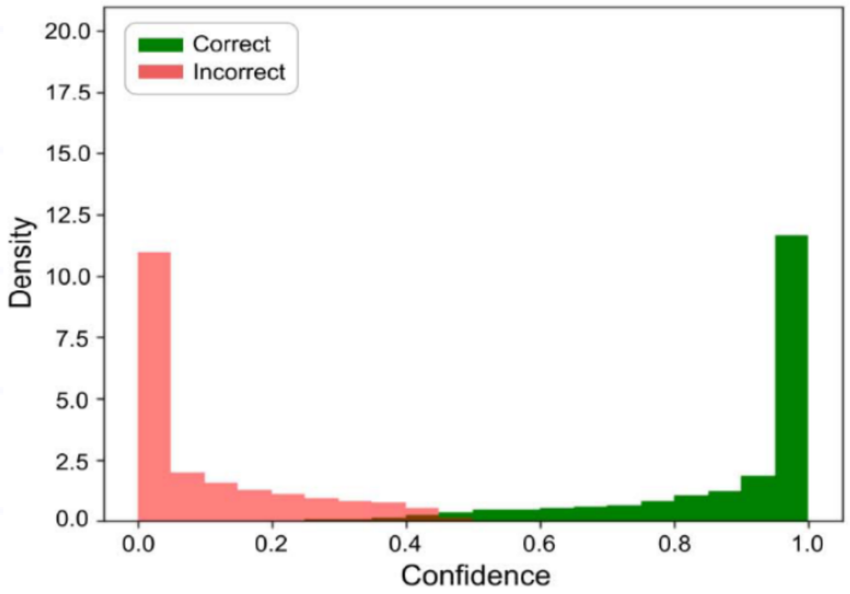
- Confidence, P(output=correct)

How to assess the calibration?

Guo et al., On Calibration of Modern Neural Networks, ICML 2017

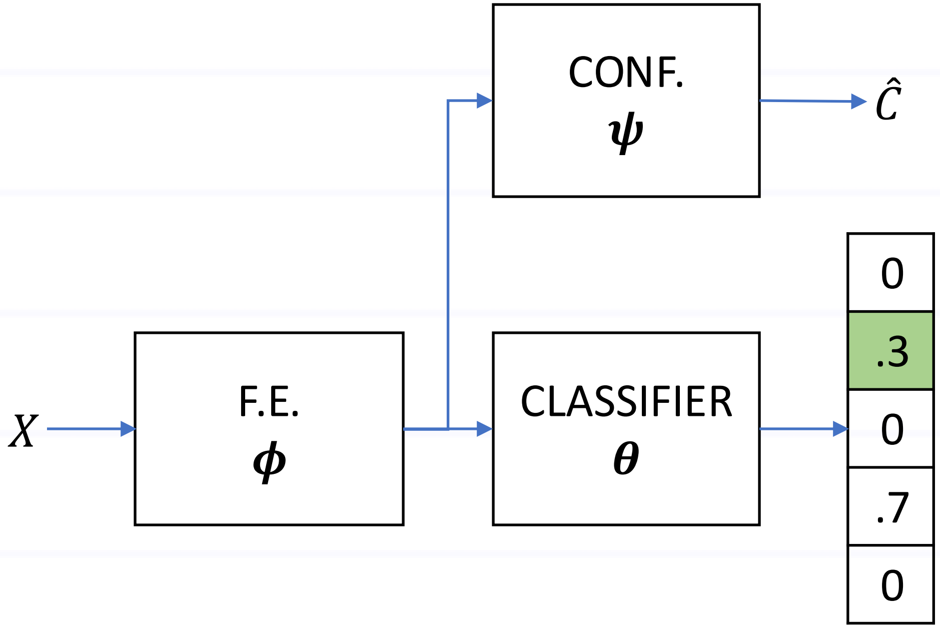
Intuitively

- Large confidence implies correct output
- Small confidence implies incorrect output



Notations

- input: X
- ground truth: $Y \in \mathcal{Y} = \{1, \dots, K\}$
- output: $\hat{Y} = f_\theta\left(f_\phi(X)\right)$
- output class: $\hat{Y}^* = \text{argmax}_k \hat{Y}[k]$
- confidence: $\hat{C} = f_\psi\left(f_\phi(X)\right)$



Calibration

- **Que**: When is the model perfectly calibrated, $P\left(\hat{Y}^* = Y \middle| \hat{C} = c\right) = ?$
- $P\left(\hat{Y}^* = Y \middle| \hat{C} = c\right) = c, \forall c \in [0,1]$
- **Que**: what terms above are functions of X?
- $Y, \hat{Y}^*, \hat{C}$
- **Que**: what terms above are independent variables?
- c
- **Que**: How do you compute $P\left(\hat{Y}^* = Y \middle| \hat{C} = c\right)$ from samples?
- = # of correct / total #, for all samples with $\hat{C} = c$

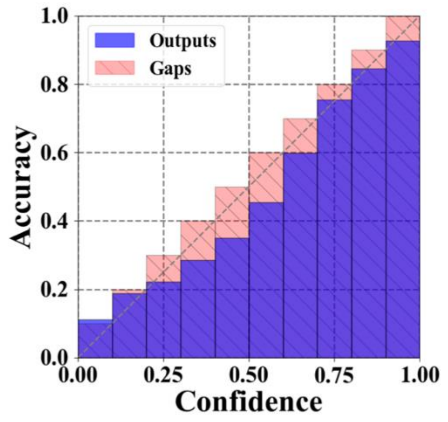
MADHAV

Problem

$$P(\hat{Y}^* = Y | \hat{C} = c) = c, \forall c \in [0,1]$$

- c is a continuous variable. How many X in a finite database can have $\hat{C} = c$
- So, we need approximation
- E.g., binning of c

Reliability Diagrams



- Accuracy vs confidence
- Let B_m be the set of samples with $\hat{C}$ falling in mth bin
- Accuracy, $\mathrm{acc}(B_m) = \frac{1}{|B_m|}\sum_{i \in B_m} 1(\hat{y}_i^* = y_i)$
- Confidence, $\mathrm{conf}(B_m) = \frac{1}{|B_m|}\sum_{i \in B_m} \hat{C}_i$
- For perfectly calibrated model, $\mathrm{acc}(B_m) = \mathrm{conf}(B_m) \forall m = \{1, \ldots M\}$

Exercise

For a flower classification task,

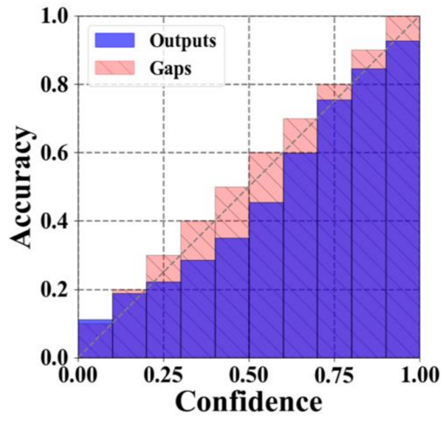
$$Y = \{RRRR\ JJJJ\ LLLL\}$$

$$\hat{Y}^* = \{RLRR\ JLJR\ LLRJ\}$$

$$\hat{C} = \{.8, .7, .4, .7, \qquad .7, .8, .8, .4, \qquad .8, .7, .4, .4\}$$

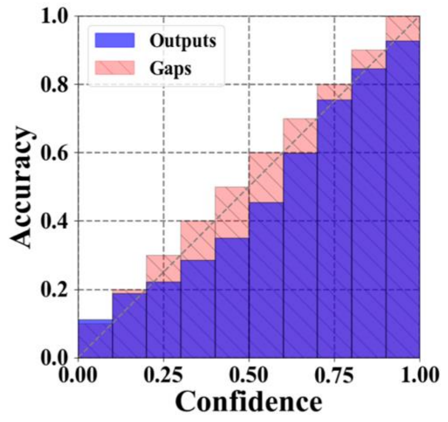
Draw the reliability diagram.

Expected Calibration Error



- Average over the reliability diagram
- ECE $= \mathbb{E}_{\hat{C}}\left[\left|P\left(\hat{Y}^* = Y \middle| \hat{C} = c\right) - c\right|\right]$
- **Que**: write the Monte Carlo approximate of ECE
- ECE $= \sum_{m=1}^{M} \frac{|B_m|}{n} |\mathrm{acc}(B_m) - \mathrm{conf}(B_m)|$

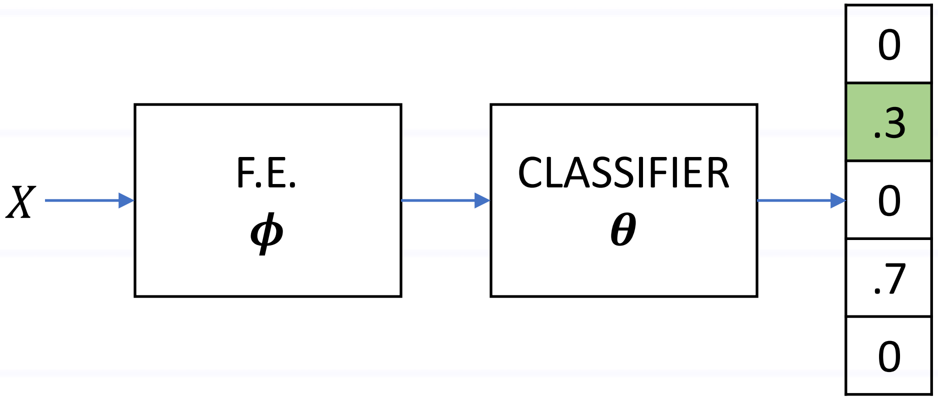
Maximum Calibration Error

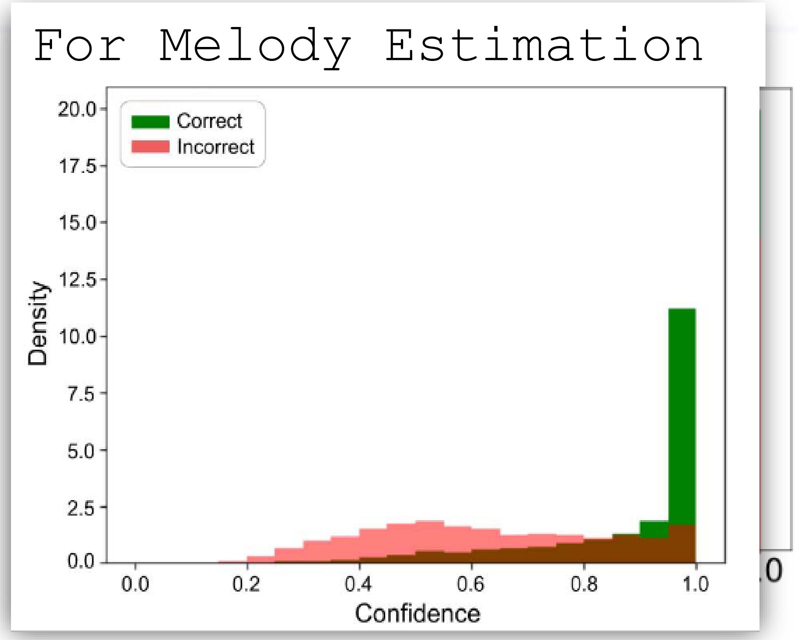


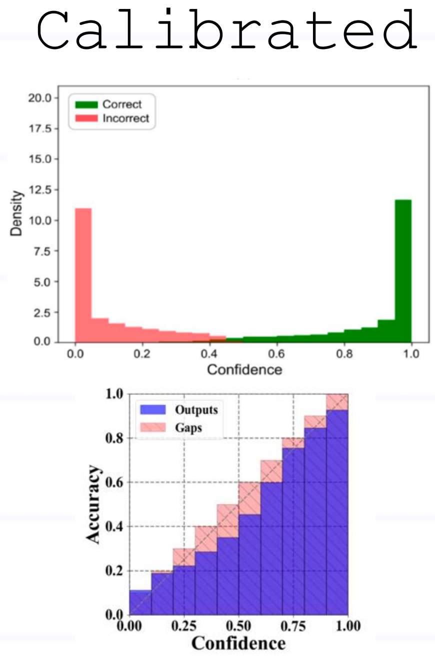
- Take max error over the reliability diagram
- MCE $= \max_{c \in [0,1]} \left| P\left(\hat{Y}^* = Y \middle| \hat{C} = c\right) - c \right|$
- **Que**: write the Monte Carlo approximate of MCE
- MCE $= \max_{m \in \{1,\ldots,M\}} |\mathrm{acc}(B_m) - \mathrm{conf}(B_m)|$

Real world systems

- Confidence = probability of being correct







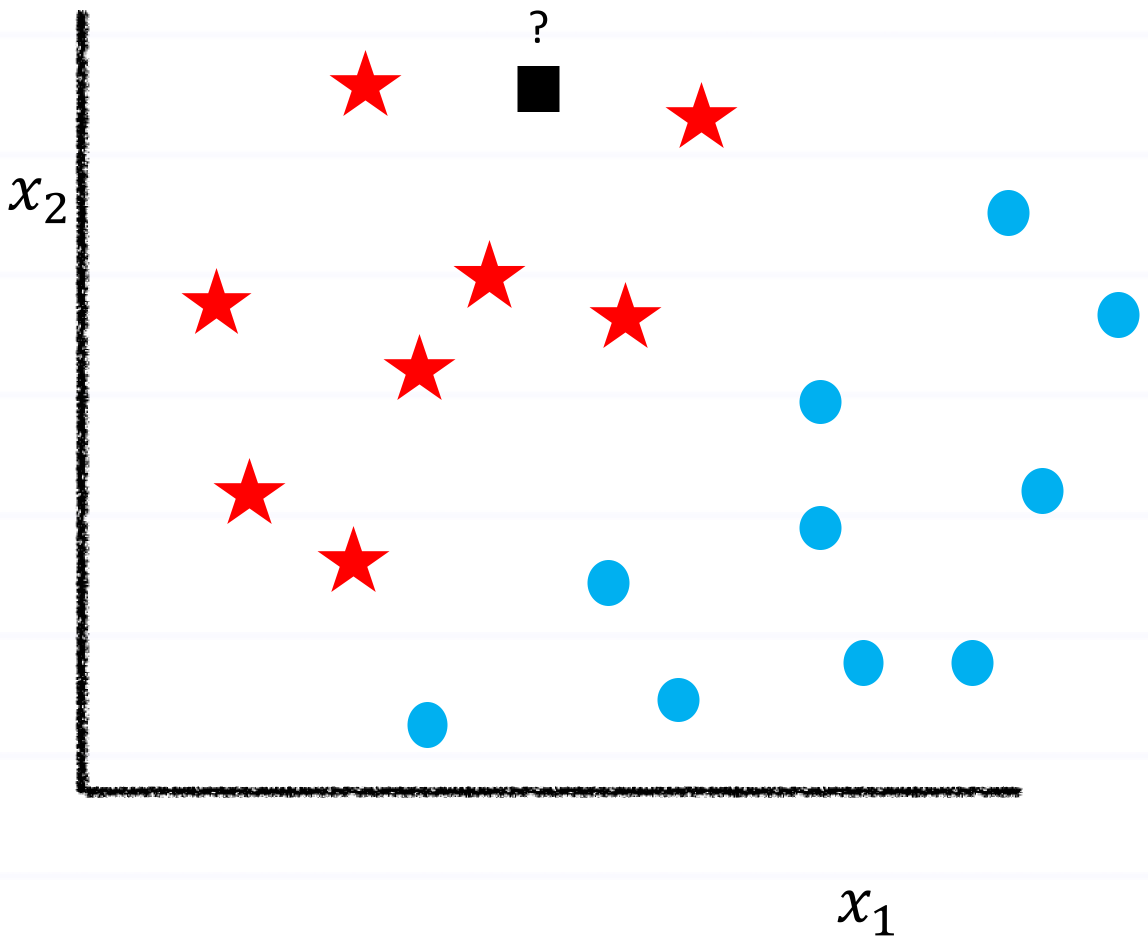
MADHAV

Why are Deep Networks Mis-calibrated?

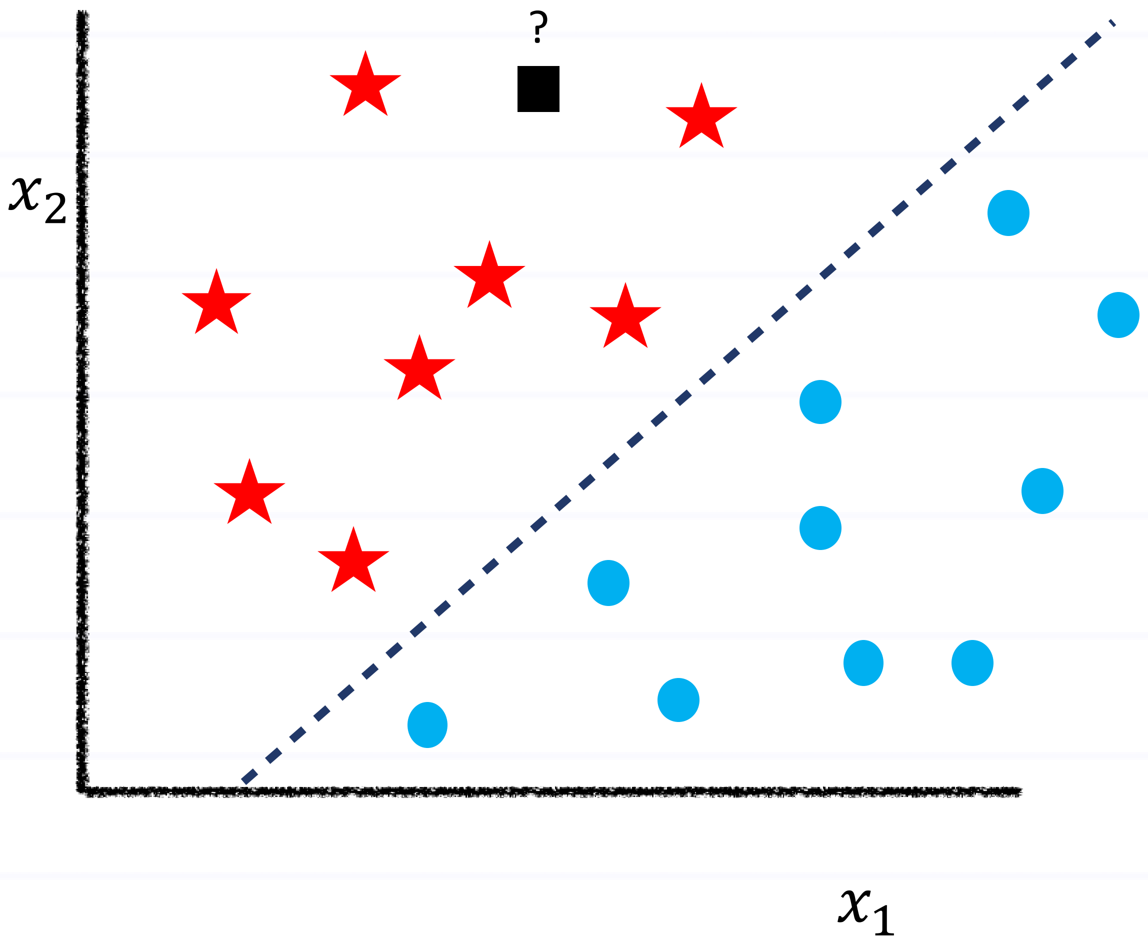
Classification

Feature Extraction

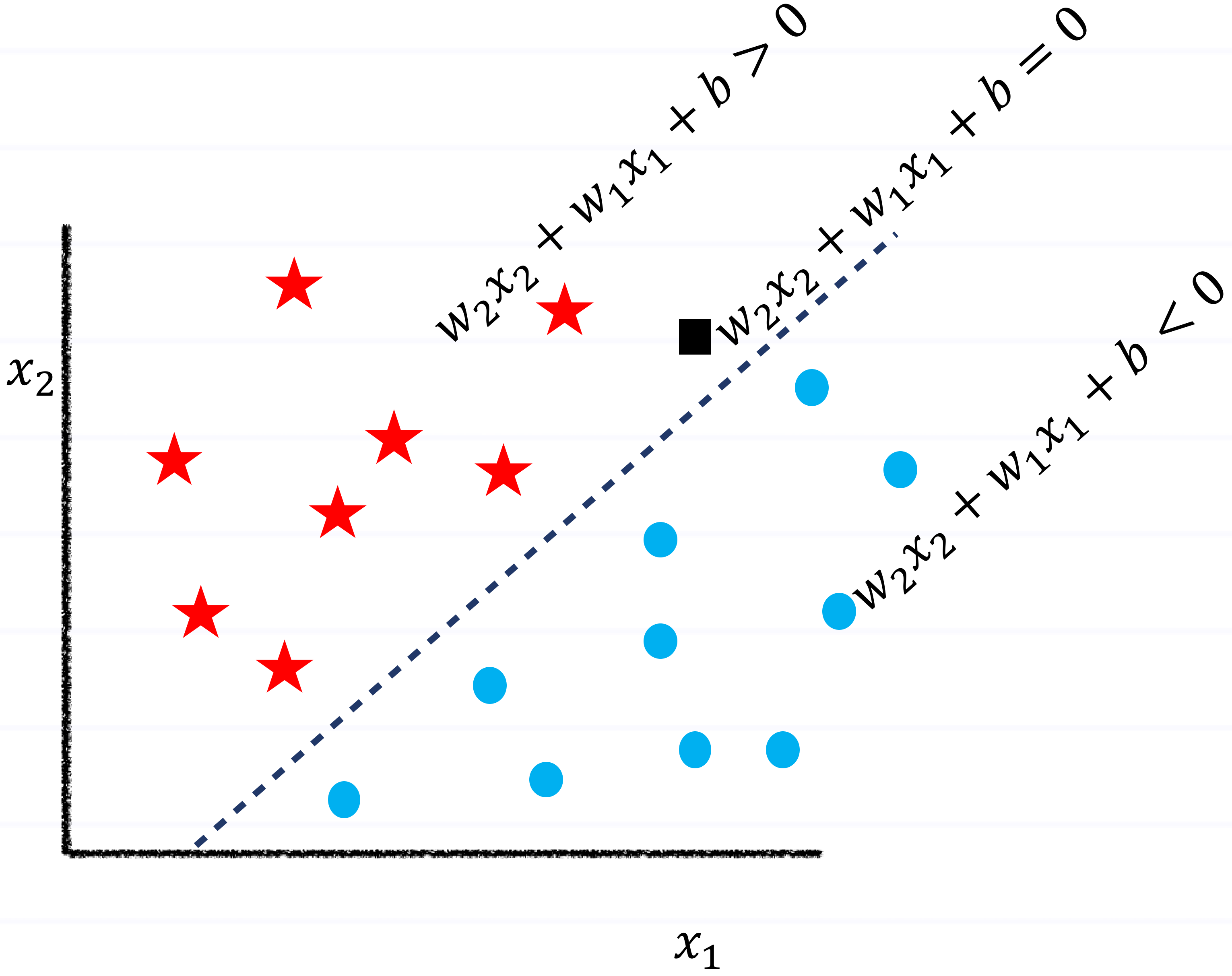
- height = x_1
- diameter = x_2


?
x_2
x_1

Que: confidence?



Que: What are roughly w_1, w_2, b?



Hard Classification

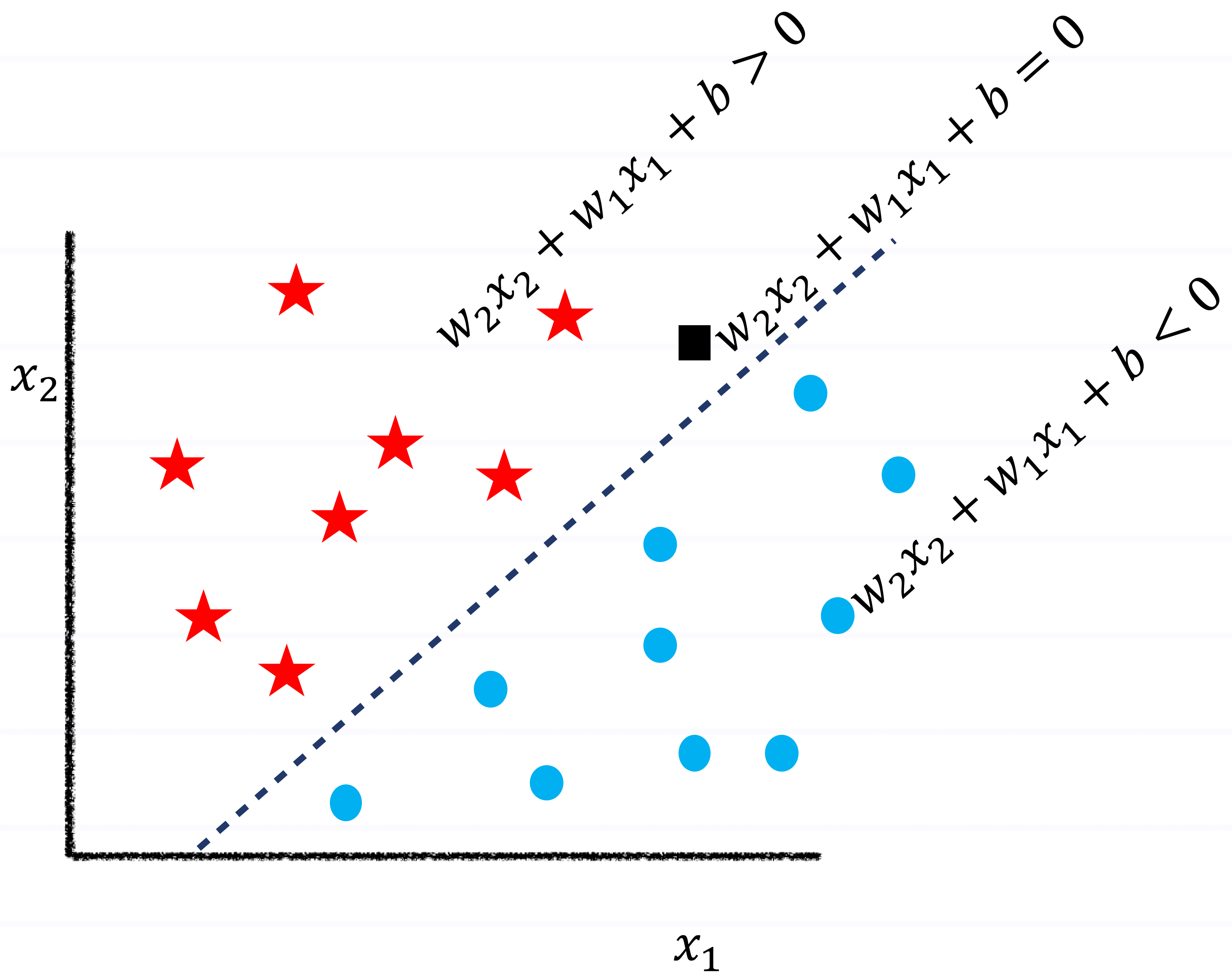
■ = ★

Soft Classification

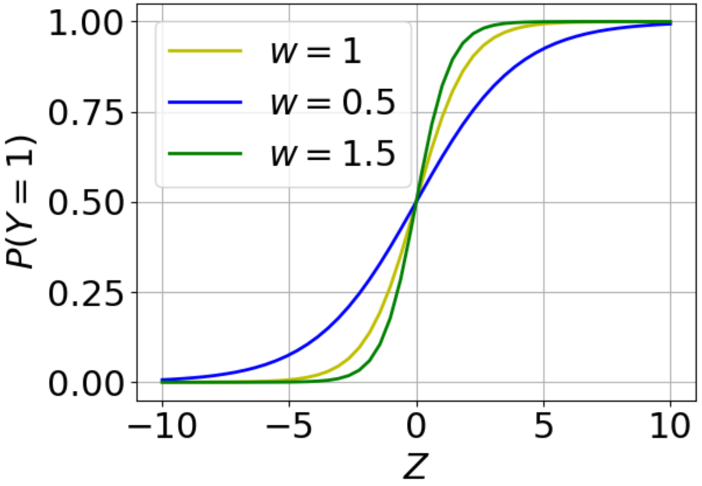
$P($■ = ★$) = 0.6$

$P($■ = ●$) = 0.4$

$\hat{c} = 0.6$

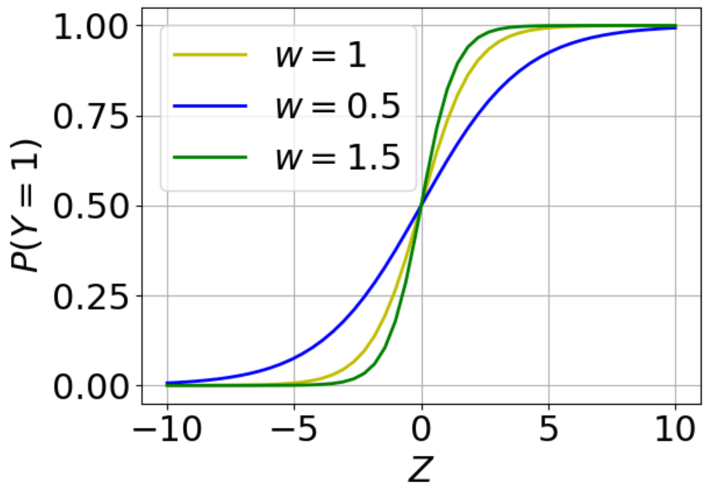


Maths



- $z = w_2 x_2 + w_1 x_1 + b$
- **Que**: What is the output of a binary classifier?
- $P(y = 1|x) = \frac{1}{1+\exp(-z)} = \sigma(z)$
- **Que**: Keeping the class boundary same, what decides this P?
- $P(y = 1|x) = \sigma\left(w \times (w_2 x_2 + w_1 x_1 + b)\right)$

Why mis-calibration



- **Que**: What is the loss function?
- Loss $= -\mathbb{E}[y \ln \hat{y} + (1-y)\ln(1-\hat{y})]$
- It is min if $\hat{y} = y \in \{0,1\}$
- The loss does not favor fractional $\hat{y}$
- w is untuned

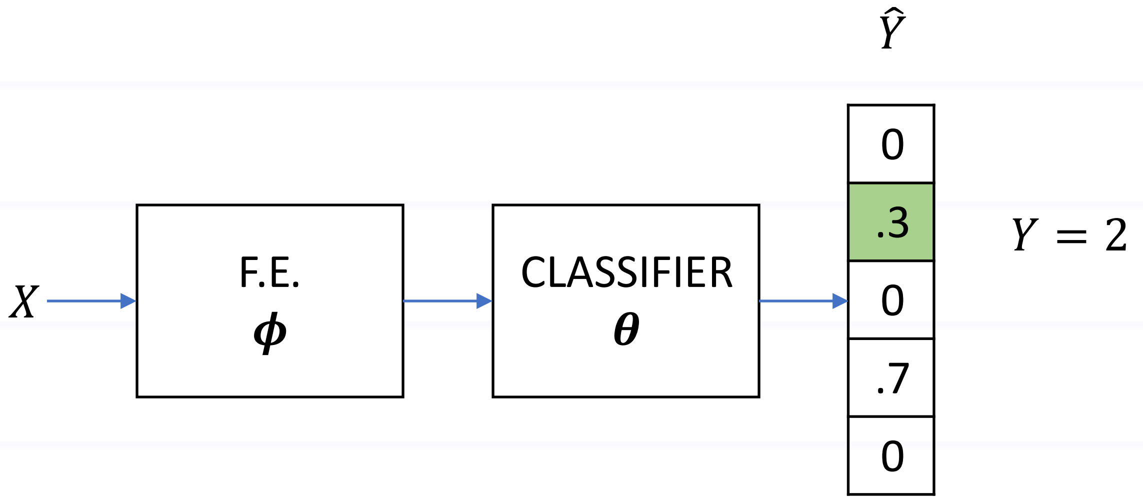
Confidence Calibration

Guo et al., On Calibration of Modern Neural Networks, ICML 2017

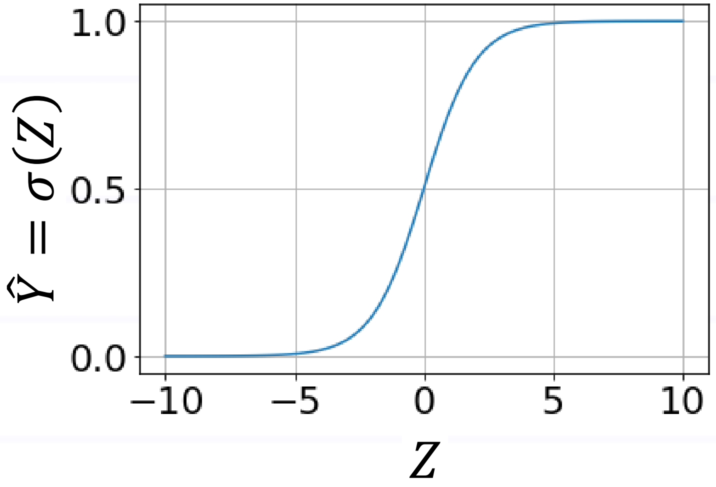
Goal

- Derive $\hat{C}$ using $\hat{Y}, \hat{Y}^*, Z, X$

$$\hat{C}(\hat{Y}, \hat{Y}^*, Z, X)$$

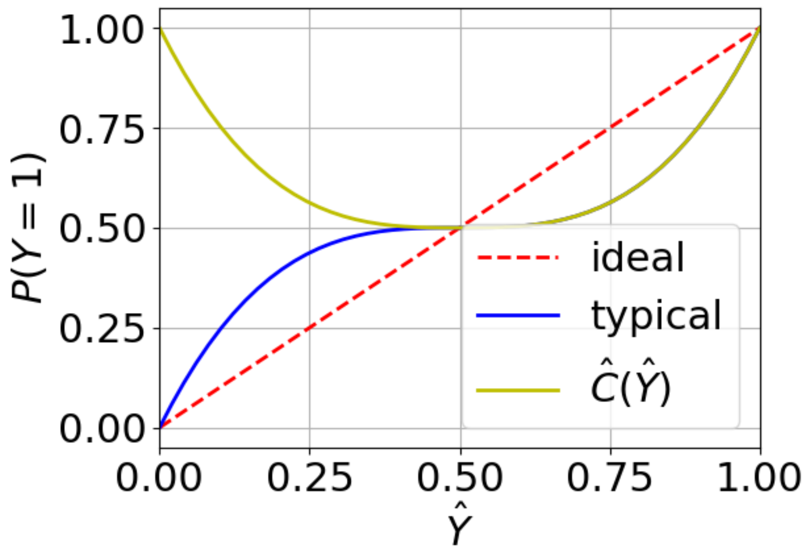


- Can't use Y during testing
- Let's focus on binary classification
- **Que**: Draw $\hat{Y}$ vs Z



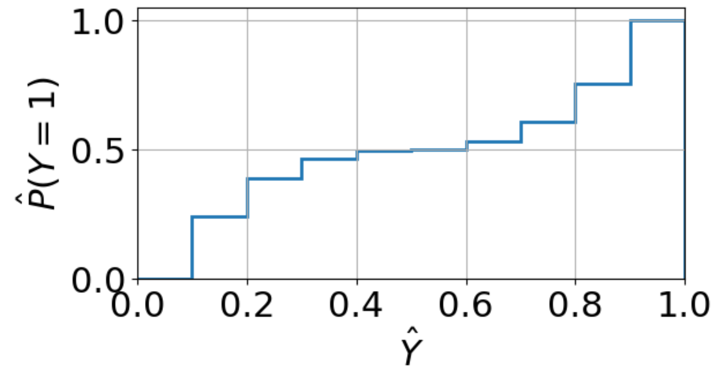
Estimating $\hat{C}(\hat{Y})$

- **Que**: Draw typical $P(Y = 1)$ vs $\hat{Y}$
- **Que**: What should be $\hat{C}(\hat{Y})$ if you know $P(Y = 1)$ curve?
- **Ans**: $\hat{C}(\hat{Y}) = P\left(Y = \arg\max_k \hat{Y}\right)$. Can you draw it for binary case?
- BTW, Instead of modeling $\hat{C}(\hat{Y})$, we could model $\hat{C}(k) := P(Y = k)$ for all k



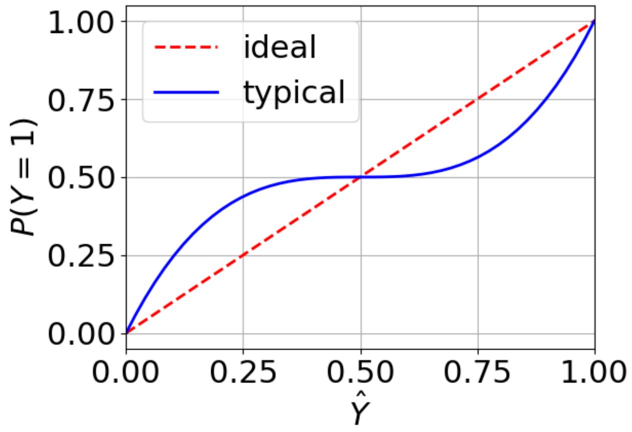
1. Histogram Binning Method

- to estimate $\hat{C}(\hat{Y})$
- Divide $\hat{Y} \in [0,1]$ into M bins
- Let $P(Y = 1) = \theta_m$ if $\hat{Y}$ falls in bin m
- To estimate θ_m, Loss $= \sum_m \sum_i 1\left(\hat{Y}_i \in \text{bin } m\right)(\theta_m - Y_i)^2$
- **Que**: What is the optimal θ_m?
- $\theta_m = \frac{\sum_i 1(\hat{Y}_i \in \text{bin } m) Y_i}{\sum_i 1(\hat{Y}_i \in \text{bin } m)}$



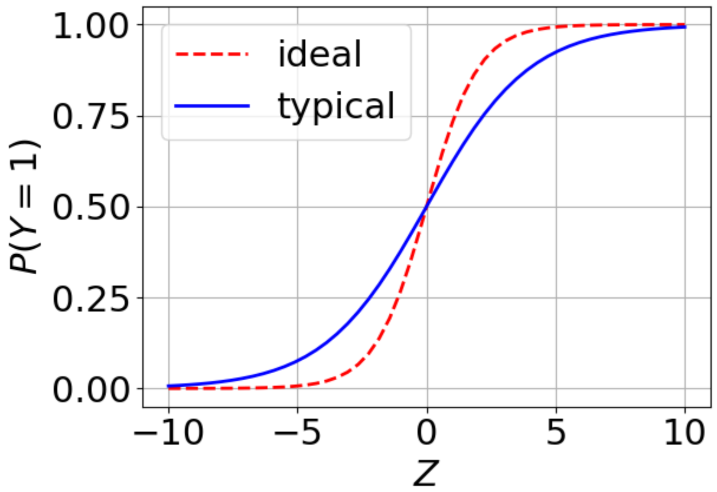
2. Isotonic Regression Method

- to estimate $\hat{C}(\hat{Y})$
- One can learn $P(Y = 1)$ as a function of $\hat{Y}$ using simple regression models
- Isotonic regression model is one such model



3. Platt Scaling Method

- to estimate $\hat{C}(Z)$ Z is logit
- **Que**: Draw typical $P(Y=1)$ vs Z
- Approximate this with a sigmoid as



$$P(Y=1) = \sigma(aZ+b)$$

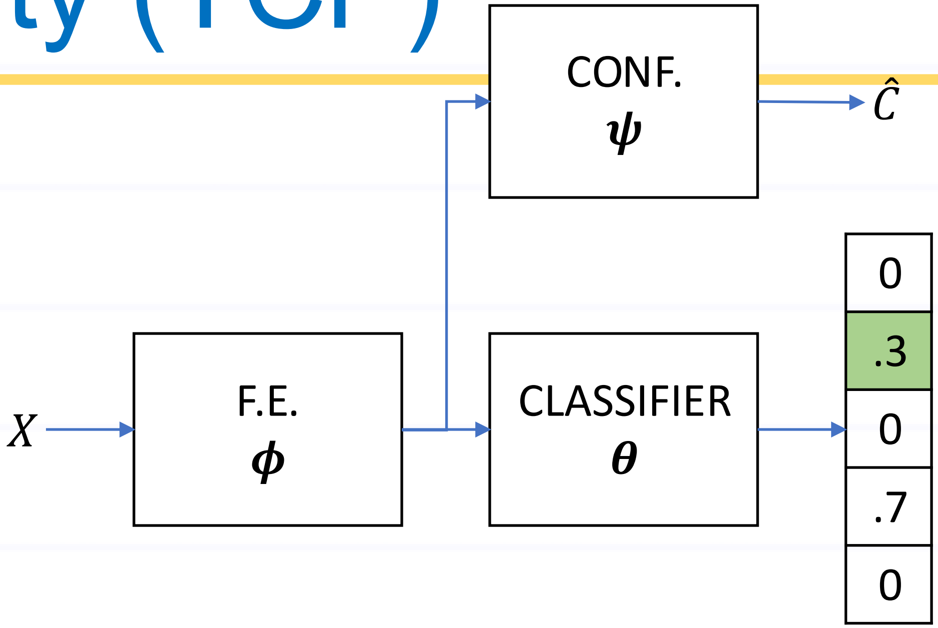
- If $b=0$, it is called Temperature Scaling method with $a = 1/T$

3. Platt Scaling Method

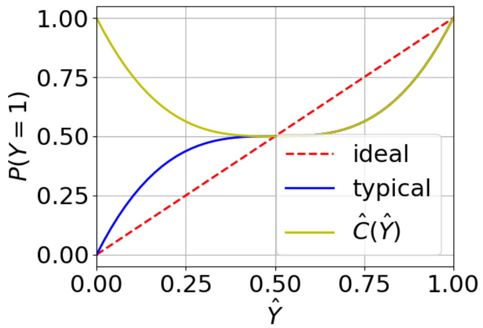
$$P(Y = 1) = \sigma(aZ + b)$$

- a and b are estimated using MLE over validation data
- **Que**: Could you write the loss function?
- Loss = $-\sum_i \delta_{y_i,1} \ln \sigma(aZ + b) + \delta_{y_i,0} \ln\big(1 - \sigma(aZ + b)\big)$

4. True Class Probability (TCP)



- $\hat{C}(X)$
- Train ψ as a regressor
- Loss $= \mathbb{E}\left[\left(C(X) - \hat{C}(X)\right)^2\right]$
- What should be the target?
- Use $C(X) = \hat{Y}[k]$; $k = Y$

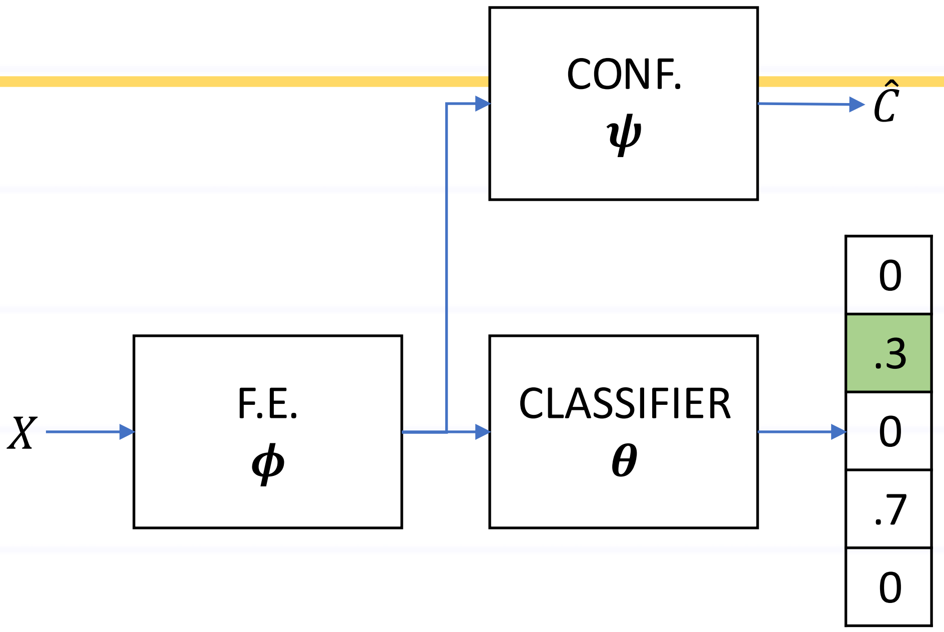


Corbiere, et al., "Addressing failure prediction by learning model confidence," NeurIPS 2019.

MADHAV

4. Normalized TCP

- $\hat{C}(X)$
- Loss $= \mathbb{E}\left[\left(C(X) - \hat{C}(X)\right)^2\right]$
- When number of classes is large, $\hat{Y}[k]$ gets smaller
- Use $C(X) = \frac{\hat{Y}[k]}{\max_{k'} \hat{Y}[k']}$; $k = Y$

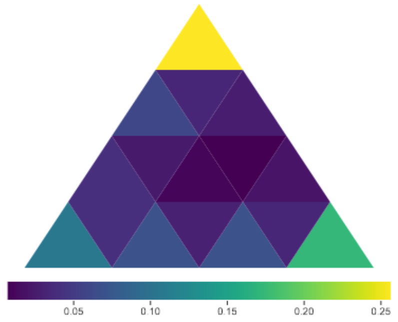


Corbiere, et al., "Addressing failure prediction by learning model confidence," NeurIPS 2019.

MADHAV

For multi-class classification

- Treat it as K one-vs-all problems
- Estimate $P(Y = k)$ for all k using $Z[k]$ or Z
- Hence, get $\hat{C}[k]$
- **Que**: Why not treat it as a full multi-class problem?
- Curse of dimensionality



MADHAV

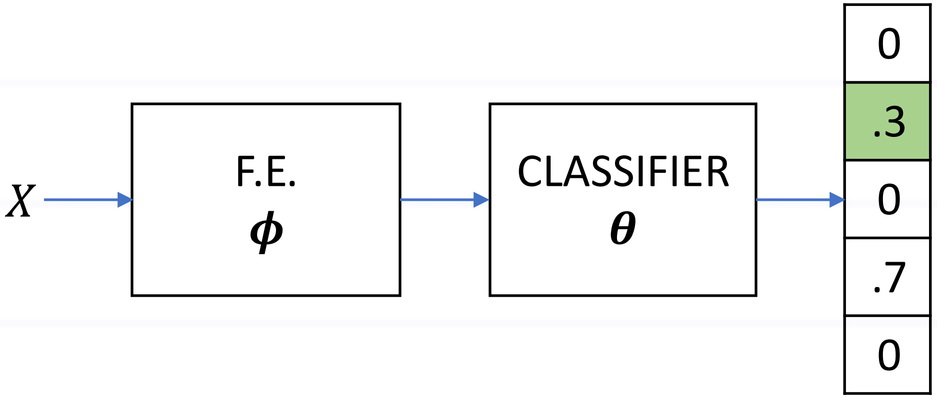
Bayesian Methods

Gal and Ghahramani, Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning, ICML 2016

Bayesian Neural Network

- Typical NN

$$\hat{p}(Y|X) = \hat{Y} = f_\theta\left(f_\phi(X)\right)$$



- In Bayesian NN, ϕ, θ are also random variables; so, we get

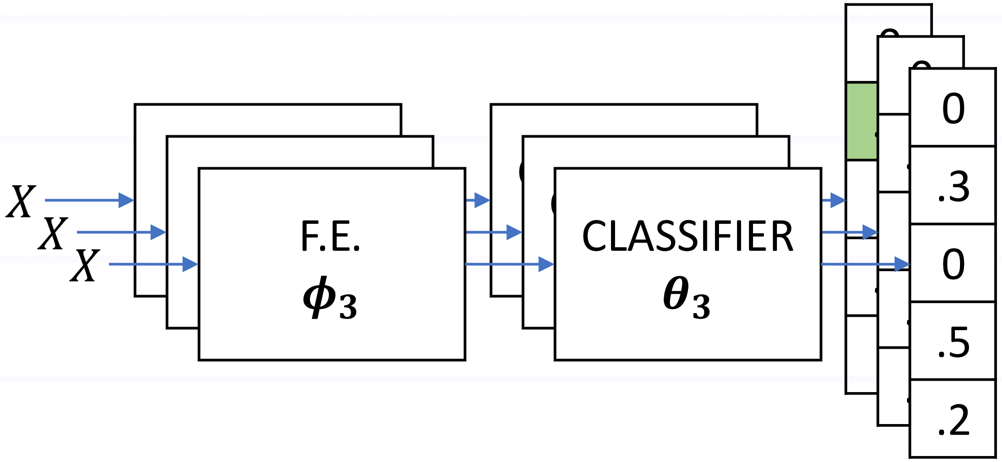
$$p(Y|X) = \iint p(Y|X,\theta,\phi)p(\theta,\phi)d\theta d\phi$$

Monte Carlo Estimation

- $p(Y|X) = \iint p(Y|X,\theta,\phi)p(\theta,\phi)d\theta d\phi$
- $p(Y|X) = \mathbb{E}_{\theta,\phi\sim p(\theta,\phi)}[p(Y|X,\theta,\phi)]$
- $\hat{p}(Y|X) \approx \frac{1}{N}\sum_i p(Y|X,\theta_i,\phi_i)\,; \quad \theta_i,\phi_i \sim p(\theta,\phi)$

Monte Carlo Dropouts

- $\hat{p}(Y|X) \approx \frac{1}{N}\sum_{i=1}^{N} p(Y|X,\theta_i,\phi_i)\,;\ \theta_i,\phi_i \sim p(\theta,\phi)$
- $\theta,\phi \sim p(\theta,\phi)$ is approximated using dropout



Uncertainty

- $\mu = \frac{1}{N} \sum_i \hat{Y}_i$
- $\Sigma = \frac{1}{N} \sum_i (\hat{Y}_i - \mu)^{\intercal} (\hat{Y}_i - \mu)$
- Indicators of uncertainty:
  - Total variance $= \sum_{k,l} \Sigma_{kl}$
  - Entropy $= -\sum_k \mu[k] \ln \mu[k]$

Ensemble Method

- $\hat{p}(Y|X) \approx \frac{1}{N}\sum_i p(Y|X, \theta_i, \phi_i)\,; \quad \theta_i, \phi_i \sim p(\theta, \phi)$
- $\theta, \phi \sim p(\theta, \phi)$ is a model trained using stochastic optimization algorithms
- We train N different models and use them to estimate uncertainty

Input Perturbation Method

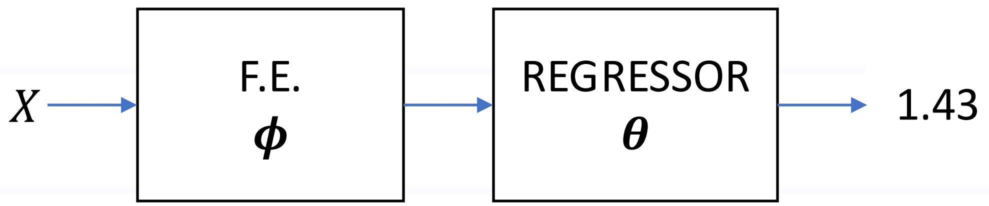
- $p(Y|X) = \int p(Y|\tilde{X})p(\tilde{X}|X)d\tilde{X}$
- $\hat{p}(Y|X) \approx \frac{1}{N}\sum_i p(Y|\tilde{X}_i)$; $\quad \tilde{X}_i = X + \epsilon_i$
- Here, ϵ_i is noise or systematic perturbation of input X

Regression

Kuleshov et al., Accurate Uncertainties for Deep Learning Using Calibrated Regression, ICML 2018

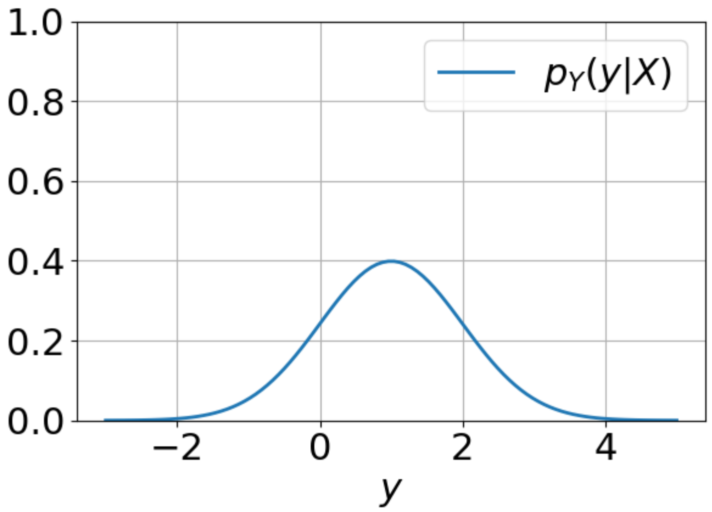
Regression

- Output $\hat{Y} \in \mathbb{R}^d$
- How do we define confidence?



Confidence

- Let us estimate the pdf of output, instead of point estimate. Let y be the output random variable.
- Now, true value Y can lie anywhere

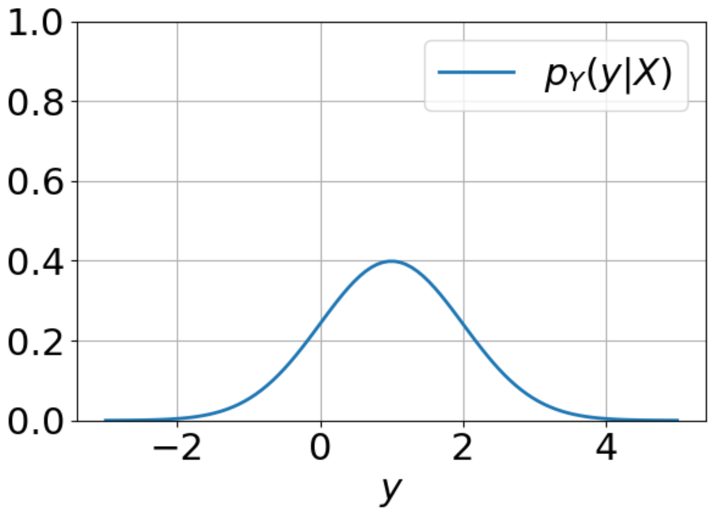


Confidence

- Example: Let y_2 be such that

$$\int_{-\infty}^{y_2} p(y|X)dy = 0.5$$

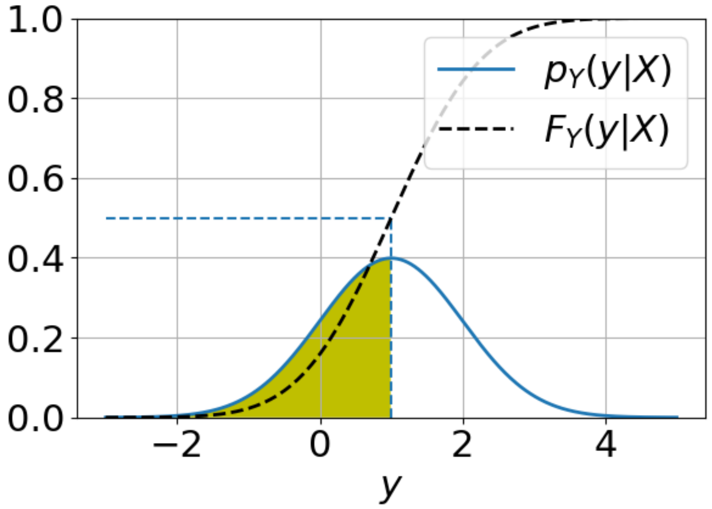
- Now, empirically $P(Y \leq y_2|X)$ should be 0.5
- This is true for only one X, how to generalize



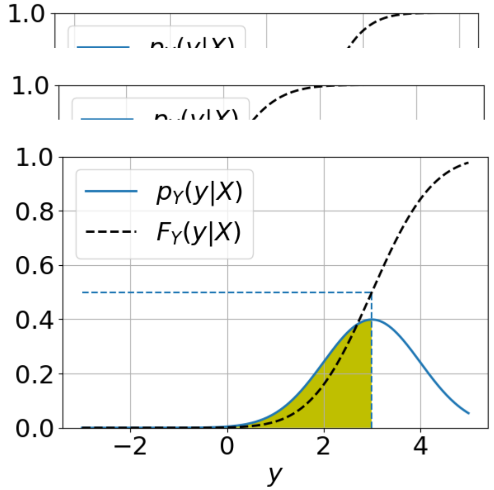
Confidence

- Let us define an interval in terms of probability mass
- Example: Let y_2 be a point such that

$$\int_{-\infty}^{y_2} p(y|X)dy = 0.5$$



- But we know that this is $F_Y(y_2|X) = 0.5$, where $F_Y(y_2)$ is the distribution function or CDF
- Thus, $y_2 = F_Y^{-1}(0.5|X)$
- y_2 is a function of X



MADHAV

Confidence

- $y_2 = F_Y^{-1}(0.5|X)$
- Now, for a calibrated regression model, empirically

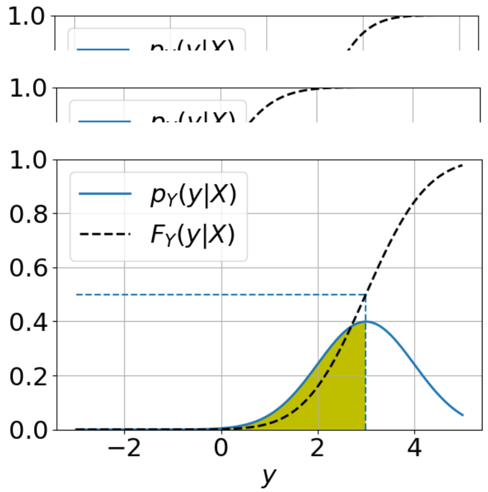
$$\frac{1}{N}\sum_{n}^{N} \mathbb{I}\{Y_i \leq F_Y^{-1}(0.5|X)\} =?$$

- $= 0.5$

Confidence

- In general, for a calibrated regression model,

$$\frac{1}{N}\sum_{n}^{N} \mathbb{I}\{Y_i \leq F_Y^{-1}(c|X)\} = c$$



Confidence

- Even more general, for a calibrated regression model,

$$\frac{1}{N}\sum_{n}^{N}\mathbb{I}\{F_Y^{-1}(c_2|X) \leq Y_i \leq F_Y^{-1}(c_2|X)\} = c_2 - c_1$$

- $(c_2 - c_1)$ is the confidence interval

Summary

- Confidence calibration is possible when we estimate the complete pdf $p(Y|X)$ instead of just a point estimate $\hat{Y}$
- We need to estimate CDF $F_Y(y|X)$ and inverse CDF $F_Y^{-1}(c|X)$ to get confidence intervals

Confidence

- It is common to assume output to be Gaussian

$$p(y|X) = \mathcal{N}(y; \mu(X), \sigma^2(x))$$

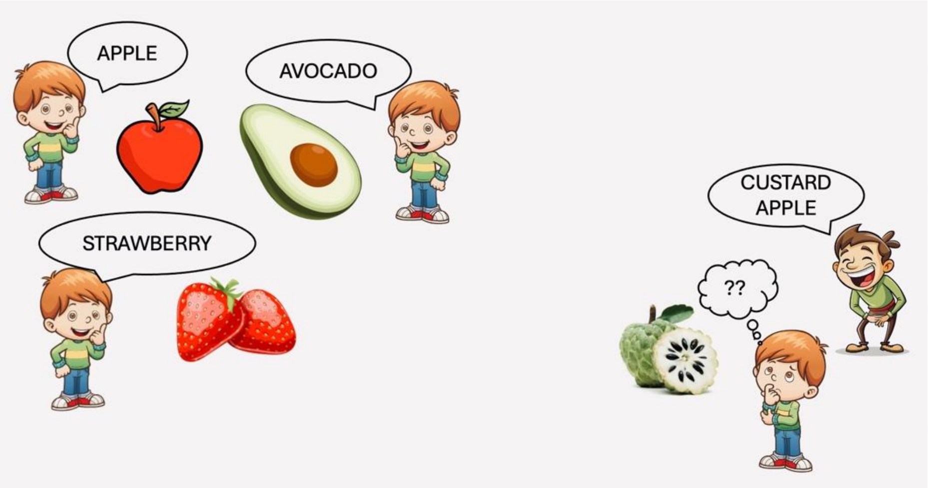
- $\mu(X), \sigma(x)$ are estimated using NNs

Sources of Uncertainty

- Kendall and Gal, What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?, NeurIPS 2017

- Hüllermeier and Waegeman, Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods, Machine Learning 2021

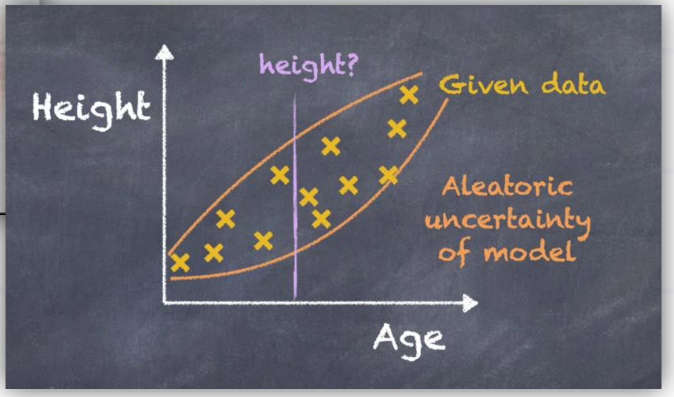
Calibration is not enough



Two kinds of uncertainty:

1. **Model** is limited, not trained on this data or class. **Epistemic Uncertainty**

2. **Data** ambiguous, even if model has been trained on similar data. **Aleatoric Uncertainty**



Should we distinguish?

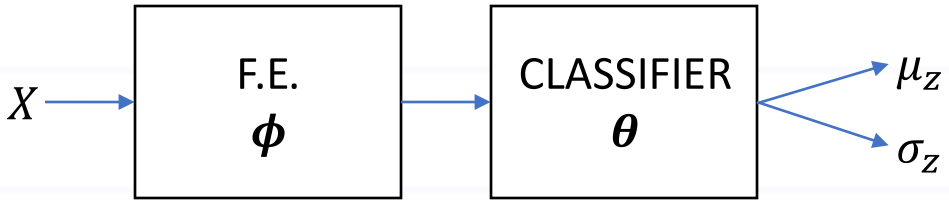
- Epistemic Uncertainty
  - tells about out of domain data (new, unseen inputs) (useful for model adaptation and active learning)
  - tells about anomalies and outliers
  - tells about unseen classes (novel class discovery)
- Aleatoric Uncertainty
  - tells about difficult data which needs manual intervention. More training won’t help.

Can we distinguish?

Yes

- Bayesian NN
- Evidential learning [Sensoy et al., Evidential Deep Learning to Quantify Classification Uncertainty, NeurIPS 2018]

Consider the Z space (logits)



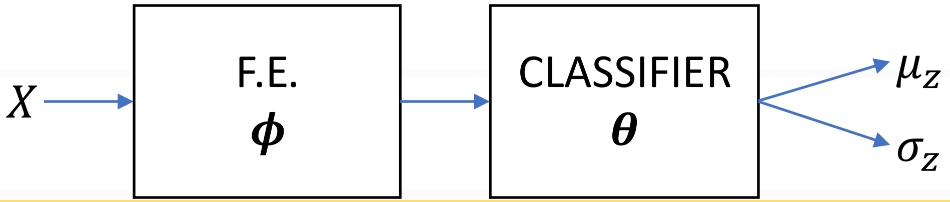
- Assume Gaussian output

$$Z \sim \mathcal{N}(\mu_z, \sigma_z^2); \mu_z \in \mathbb{R}^K, \sigma_z \in \mathbb{R}^K$$

where $\mu_z, \sigma_z = f_\theta\left(f_\phi(X)\right)$

- Output is as usual $\hat{Y} = \mathrm{softmax}(Z)$ for classification and $\hat{Y} = Z$ for regression

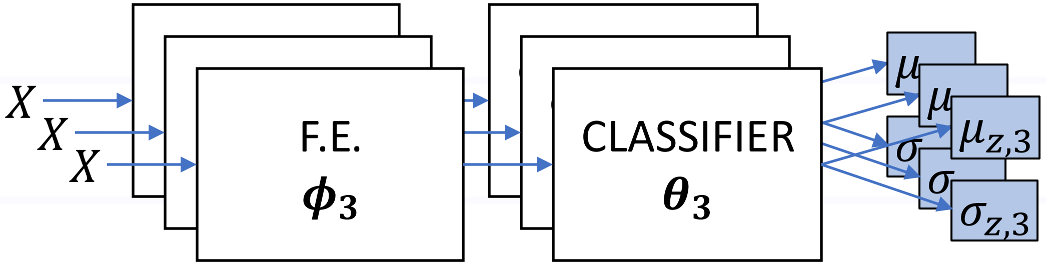
Uncertainty



- σ_Z quantifies the uncertainty in Z, but what kind of uncertainty?
- Let us perturb the model parameters
- $\mu_{Z,i}, \sigma_{Z,i} = f_{\theta_i}\left(f_{\phi_i}(X)\right)$

Uncertainty

- Que: What is $\mathbb{E}[Z]$?



$$\mathbb{E}[Z] = \iint Zp(Z|X,w)p(w)dw\, dZ \;; \quad w = \{\phi, \theta\}$$

$$= \int \mu_{z,w}\; p(w)dw$$

$$\approx \frac{1}{N}\sum_{i=1}^{N} \mu_{z,i}$$

Uncertainty

- Que: What is $\mathrm{covar}[Z]$ or $\mathbb{E}[(Z - \mathbb{E}[Z])^\intercal (Z - \mathbb{E}[Z])]$?

$$\mathbb{E}[Z^\intercal Z] = \iint Z^\intercal Z p(Z|X, w) p(w) dw\, dZ$$

$$= \int \left(\sigma_{z,w}^2 I + \mu_{z,w}^\intercal \mu_{z,w}\right) p(w) dw$$

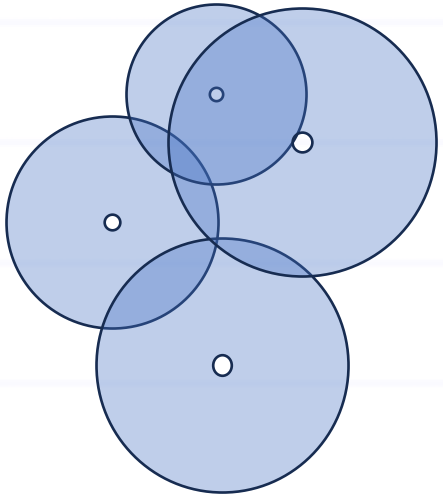
$$\approx \frac{1}{N} \sum_{i=1}^{N} \left(\sigma_{z,i}^2 I + \mu_{z,i}^\intercal \mu_{z,i}\right)$$

MADHAV

Uncertainty

- $\text{covar}[Z] \approx \frac{1}{N}\sum_{i=1}^{N}\left(\sigma_{z,i}^{2} I + \mu_{z,i}^{\intercal}\mu_{z,i}\right) - \mathbb{E}[Z]^{2}$
- $= \underbrace{\frac{1}{N}\sum_{i}\sigma_{z,i}^{2} I}_{\text{Aleatoric Uncertainty}} + \underbrace{\frac{1}{N}\sum_{i}\mu_{z,i}^{\intercal}\mu_{z,i} - \left(\frac{1}{N}\sum_{i=1}^{N}\mu_{z,i}\right)^{2}}_{\text{Epistemic Uncertainty}}$

PICTORIAL UNDERSTANDING



Quick Introductions

Evidential Learning

- Treat μ_z, σ_z also as random variables
- $\mathrm{var}[\mu_z]$ gives epistemic uncertainty
- $\mathbb{E}[\sigma_z^2]$ gives aleatoric uncertainty

Conformal Prediction

- Random samples from an unknown distribution are given
- What is the probability that the next sample falls between the 3rd and 4th samples on the line?
- Hint: draw the CDF
- This gives us the confidence intervals

Further Reading

- Lakshminarayanan et al., “*Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles*,” NeurIPS, 2017
- Seitzer, et al., “*On the Pitfalls of Heteroscedastic Uncertainty Estimation with Probabilistic Neural Networks*,” ICLR 2022
- Sensoy et al., “*Evidential Deep Learning to Quantify Classification Uncertainty*”, NeurIPS 2018
- Angelopoulos and Bates, “*A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification*”, 2022
- Ryan Tibshirani, “*Conformal Prediction*”, Advanced Topics in Statistical Learning, Spring 2023

Further Reading

- Nagarathna, Thishyan, Chaganti, Arora, "*ASR Confidence Estimation using True Class Lexical Similarity Score*", Interspeech 2025
- Nagarathna, Thishyan and Arora, "*TeLeS: Temporal Lexeme Similarity Score to Estimate Confidence in End-to-End ASR*", IEEE TASLP 2024
- Saxena and Arora, "*Interactive Singing Melody Extraction Based on Active Adaptation*", IEEE TASLP 2024

Questions?

Next: Part 2