Last Time:

Lecture 13

SP '26

Andrea Bajcsy

- uncertainty quantification I
  - ↳ epistemic vs. aleatoric
  - ↳ frequentist vs. Bayesian

This Time:

- uncertainty quantification II
  - ↳ practical techniques! ensembles, conformal prediction

Summary & UQ Methods

	Frequentism	Bayesianism
✓	data-driven, easy comp. MLE $\hat{\theta} = \arg\max_{\theta} p(y \mid x; \theta)$	prior dist "jump starts" learning, posterior models uncertainty over θ params $p(\theta \mid \mathcal{D}) = \frac{p(\theta)\prod p(y \mid x; \theta)}{p(\mathcal{D})}$
✗	misled by sampling noise, dataset size, etc.	computation usually very costly

frontiers: ⇒ beef this up! ex. ② below

⇒ approximate this! ex. ④ below

Practical Methods for UQ

Here are some key words | techniques you may come across when looking for UQ methods for deep neural networks.

TODAY!

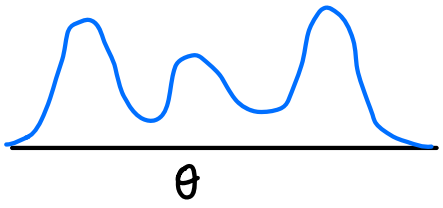
1) Deep Ensembles
   ↳ [Lakshminarayanan et al. "Simple & Scalable UQ ....". Neurips 2017]
2) Conformal Prediction (CP)
   ↳ [Angelopoulos & Bates. "Gentle Intro to CP." arXiv 2021]
3) Gaussian Processes
   ↳ [Rasmussen & Williams. "Gaussian Processes for ML." MIT Press 2006]
4) Laplace Approximation
   ↳ [Daxberger et. al. "Laplace Redux". NeurIPS 2024]
5) Monte Carlo Dropout
   ↳ [Gal & Ghahramani. "Dropout as Bayesian Approx." ICML 2016]

DEEP ENSEMBLES

Recall how ideally, we would have uncertainty $P(\theta|\mathcal{D})$ over space of solutions given dataset. BUT, this is hard! (recall, full Bayesian approach)

Recall how we ideally had:

① dist. over all weights: $P(\theta|\mathcal{D})$



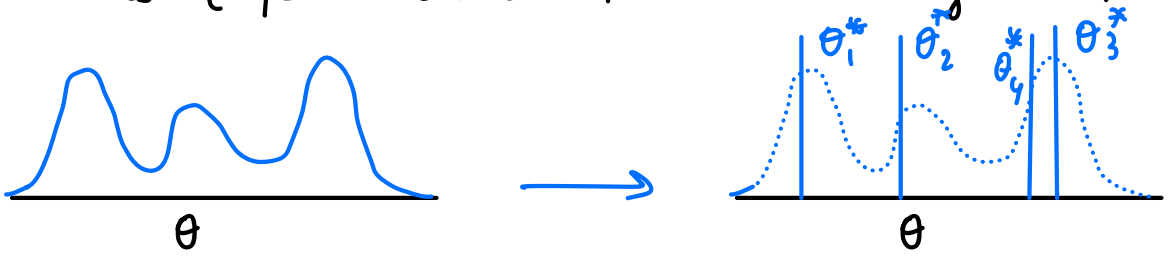
② predictions: $p(y|x,\mathcal{D}) = \int p(y|x,\theta) \cdot p(\theta|\mathcal{D})\, d\theta$

↑ INTRACTABLE INTEGRAL ex. $\theta \in \mathbb{R}^{7\text{BILLION}}$

Instead, deep ensembles approximate this empirically by replacing this complex integral w/ weighted sum of point estimates. Specifically:

① Train K independent neural networks via MLE.

This gives you $\theta_1^*, \theta_2^*, \ldots \theta_K^*$ different parameter values (b/c randomization + stochastic gradient descent.



Mathematically, you can think of it as mixture of Diracs:

$$P(\theta|\mathcal{D}) \approx \frac{1}{K} \sum_{k=1}^{K} \delta_{\theta_k^*}(\theta)$$

↑ dirac delta function where all prob. mass is on θ_i^*

② To form predictions with uncertainty, we do:

$$p(y|x,\mathcal{D}) = \frac{1}{K}\sum_{k=1}^{K} p(y|x;\theta_k^*)\cdot p(\theta_k^*|\mathcal{D})$$

↑ since we just have K Dirac deltas, this is a uniformly weighted mixture model

⚠ In practice, we represent the ensemble's predictions as a Gaussian whose mean + variance are mean + variance of the ensemble:

$$p(y|x,\mathcal{D}) \approx \mathcal{N}(\mu_{ens}(x), \sigma^2_{ens}(x))$$ ✓ e.g. for regression problems

where:

$$\mu_{ens}(x) = \frac{1}{K}\sum_{k=1}^{K} \mu_{\theta_k^*}(x)$$ ← get mean predictions from all K models

$$\sigma^2_{ens}(x) = \frac{1}{K}\sum_{k}\left(\sigma^2_{\theta_k^*} + \mu^2_{\theta_k^*}(x)\right) - \mu^2_{ens}(x)$$

↙ b/c $\text{var}(x) := \mathbb{E}[x^2] - \mathbb{E}[x]^2$

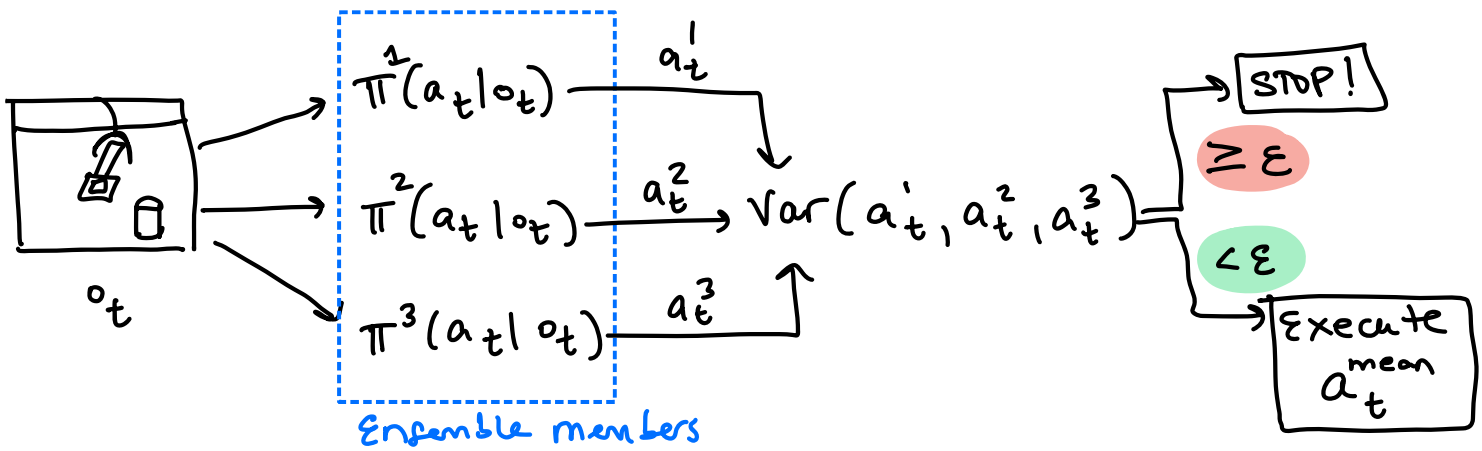
↑ variance of 1 ensemble member

↑ squared mean of 1 ensemble member

↑ mean of ensemble

What can we do with the ensemble?

↳ ex. if disagreement (i.e. variance $\sigma^2_{ens}(x)$) is too high then this means the model is not confident (i.e. maybe robot should stop, ask for help, etc.)



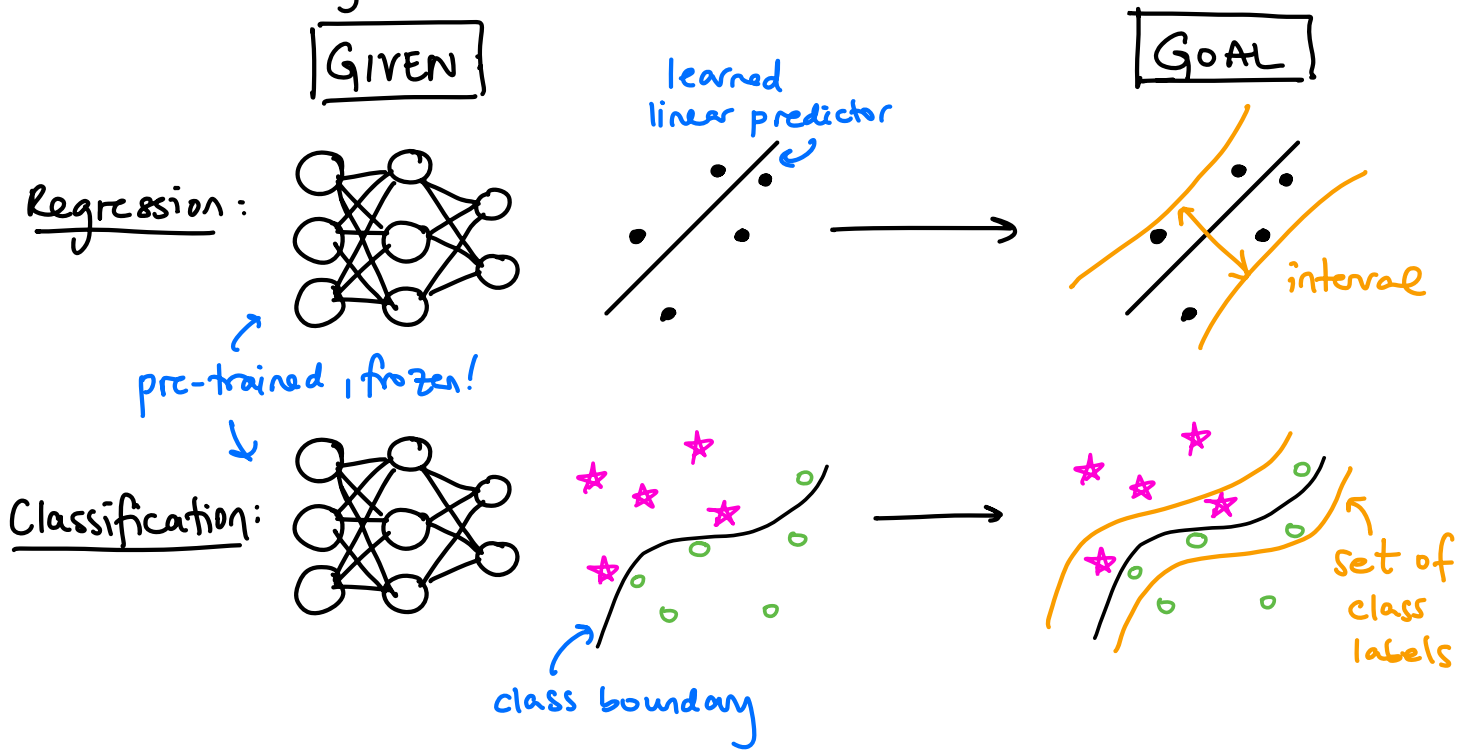
CONFORMAL PREDICTION

Ensembling has been very successful empirically. BUT, it requires you to train K separate "versions" of your predictive model. This is not possible when:

1. you download pre-trained model
2. you are training a foundation model
   - ex. it took ≈ 3 months to train GPT-4.... now imagine having to do that K times!

So, what kind of uncertainty quantification technique could we use in such a situation?

Conformal Prediction is a way to generate prediction sets for any (pre-trained) model.



Let's understand it via a image classification problem.

GOAL

Given:

→ a **calibration dataset** $\{(X_i, Y_i)\}_{i=1}^{N} \sim \mathbb{P}$ i.i.d
(fresh, unseen by model! ex. image $\in \mathbb{R}^d$, ex. class $\in Y = \{1, \dots K\}$)

→ a **model** $\hat{f}_y(x)$ which estimates $\mathbb{P}(Y|X)$
($\hat{f}$: inputs are x, outputs are over y; $\mathbb{P}$: true distribution!)

→ a **new input** X_{N+1} ← ex. new image

Predict a **SET** $C(X_{N+1}) \subseteq Y$ ← returns subset of label space

function C depends on the model $\hat{f}$ (ex. NN)

which contains the **TRUE CLASS** Y_{n+1} with high probability:

COVERAGE GUARUNTEE:

$$\mathbb{P}[Y_{n+1} \in C(X_{n+1})] \geq 1 - \alpha$$

(α: user-defined error rate)

(X_{n+1}: new image w/ UNKNOWN label!)

ex. $\alpha = 0.1$ → then prob. that label is in set is 90%.

Figure 1: Prediction set examples on Imagenet. *We show three progressively more difficult examples of the class* `fox squirrel` *and the prediction sets (i.e., $C(X_{\text{test}})$) generated by conformal prediction.*

Desired Properties of Algo:

1. Exact coverage
2. small set size $|C(X)|$
3. "Adaptive" sets: small sets when easy; big sets when hard!

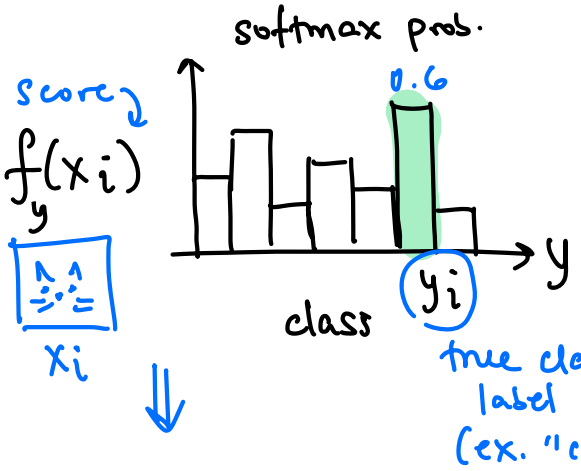
Ex.: naive way to get coverage:

$$C(X_{n+1}) = \begin{cases} Y & \text{w.p. } 1-\alpha \text{ (all labels)} \\ \emptyset & \text{w.p. } \alpha \text{ (no labels)} \end{cases}$$

⇒ get coverage but trivial/useless!

CONFORMAL PREDICTION ALGORITHM (Vovk et al., 2005)

① SCORE how poorly the model predicted the true class for all data in calibration set $\{(x_i, y_i)\}_{i=1}^{N} = \mathcal{D}_{calib}$



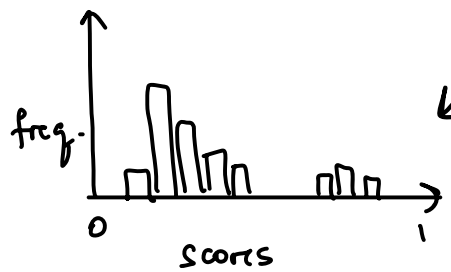
NON-CONFORMITY SCORE:

scalar ↳ $s_i := 1 - f_{y_i}(x_i)$

↳ high score ⇒ model is MORE WRONG

↳ low score ⇒ model is MORE RIGHT

$s_i = 1 - 0.6 = 0.4$



compute s_i for all $(x_i, y_i) \in \mathcal{D}_{calib}$
You get this empirical distribution over the non-conformity scores

② compute the $1-\alpha$ empirical quantile of nonconformity scores

Recall: 0.5-quantile is just the median — it's the value of a random variable such that 50% of the probability mass is "below" or to the "left" of this value.

0.9-quantile is value where 90% of probability mass is "below" or "to the left" of this value

Defn: p-quantile of distribution $\mathbb{P}$ is the value x such that:

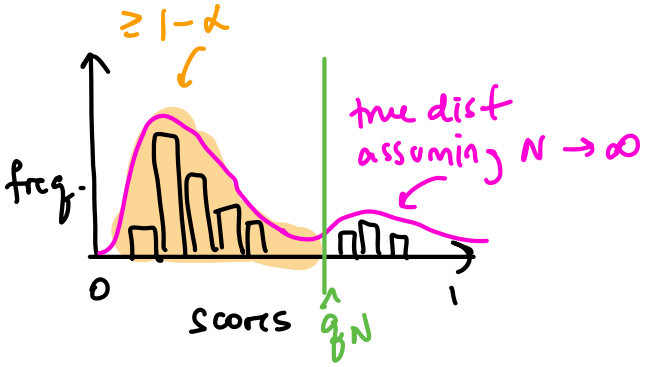
$$\mathbb{P}(X \leq x) \geq p.$$

Theoretically: find $\hat{q}_N$ s.t. $\mathbb{P}(s_{N+1} \leq \hat{q}_N) \geq 1-\alpha$ (future non-conformity score)

$\alpha = 0.1$ ↓

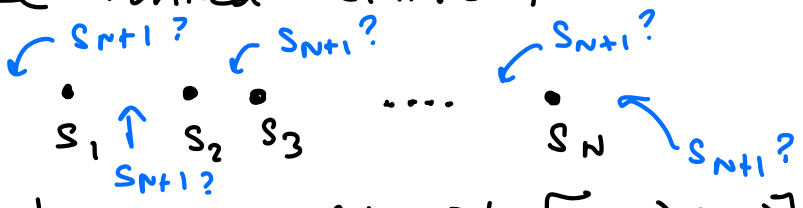
"find non-conformity score value large enough that only 10% of future values will ever exceed it (i.e. our error rate)"

⚠ CHALLENGE: theoretical vs. empirical quantile. If we had infinite data in our calibration set, then $\hat{q}_N$ would be perfect. But, we have finite data (ex. 100 pts.) — so, if we just took $(1-\alpha)$-quantile, we would pick 90^{th} smallest value out of $N=100$ pts, we would not account for the $N+1 = 101^{th}$ pt, and our coverage would only be $\frac{90}{101} \approx 89.1\% \not\geq 90\%$



Answer: do a finite sample adjustment, and instead of taking $(1-\alpha)$-quantile, we take $\lceil (1-\alpha)(n+1) \rceil$-quantile.

↳ intuition: by multiplying by $(n+1)$ we are accounting for how s_{N+1} may be ranked relative to existing calibration pts.



In our 100 data pt. example, we would pick $\lceil (1-\alpha)(n+1) \rceil = \lceil 0.9 * 101 \rceil = 91$ st smallest value out of 100 pts and our coverage is $\frac{91}{101} \approx 90.09\% \geq 90\%$

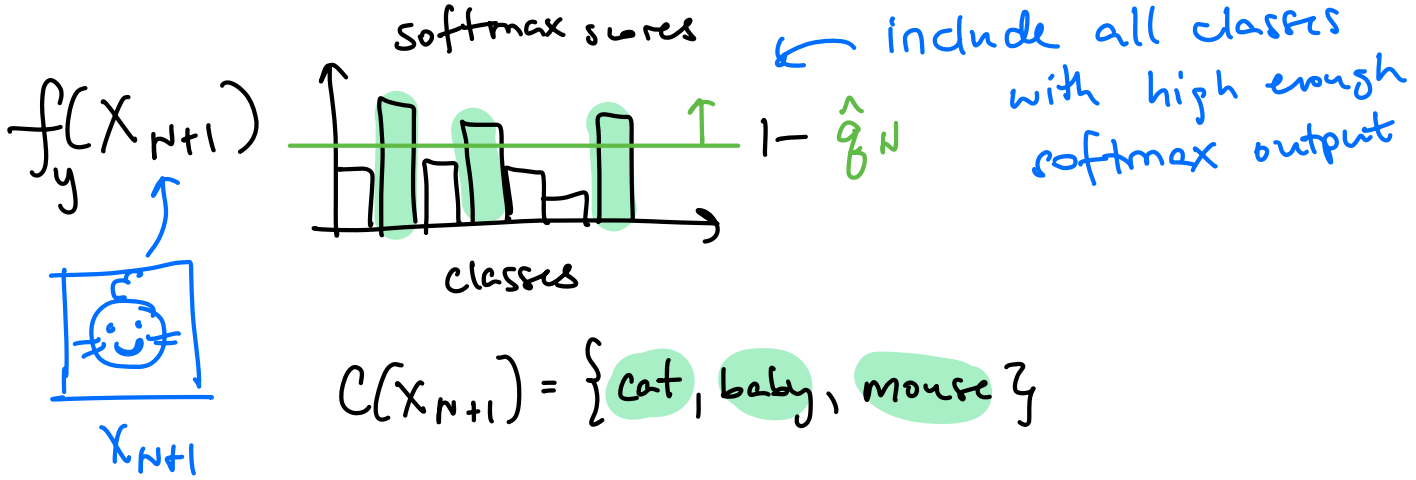
⇒ see proof for why this works in Appendix D of "A Gentle Intro to Conformal Prediction" and Prof. Ryan Tibshirani's notes (on class website)

⚠ only works b/c data (X_i, Y_i) is i.i.d.! (or, in general, exchangeable)

③ Form prediction sets using the $\hat{q}_N$ we constructed before (with finite sample adjustment):

NOTE: you may see: $s(X_{N+1}, y) \leq \hat{q}$ ↓ $1 - \hat{f}(X_{N+1}) \leq \hat{q}$ ↓ $\hat{f}(X_{N+1}) \geq 1 - \hat{q}$

$$C(X_{N+1}) = \{ y : f_y(X_{N+1}) \geq 1 - \hat{q}_N \}$$



← include all classes with high enough softmax output

$$C(X_{N+1}) = \{\text{cat}, \text{baby}, \text{mouse}\}$$

Theorem (Informal): For any model, dataset, α, N, you achieve the coverage guarantee:

$$1 - \alpha \leq \mathbb{P}(Y_{N+1} \in C(X_{N+1})) \leq 1 - \alpha + \frac{1}{N+1}$$

ex. if $\alpha = 0.1$ and $N = 100$, then the size of the prediction set is not too conservative except for a factor of $\frac{1}{N+1}$. So $0.9\% \leq \mathbb{P}(\ldots) \leq 0.91\%$.

Some Things to Consider:

① Exchangeable or i.i.d dataset is often NOT true in sequential decision-making domain!

② The design of non-conformity score REALLY matters! Key design question in CP

③ Think carefully about how to construct calibration data.