Think Your Deep Learning Model Works? Think Again!

Part 2

Tutorial at ECAI 2025, Bologna

Pietro Vischia
pietro.vischia@cern.ch
@pietrovischia

If you are reading this as a web page: have fun! If you are reading this as a PDF: please visit

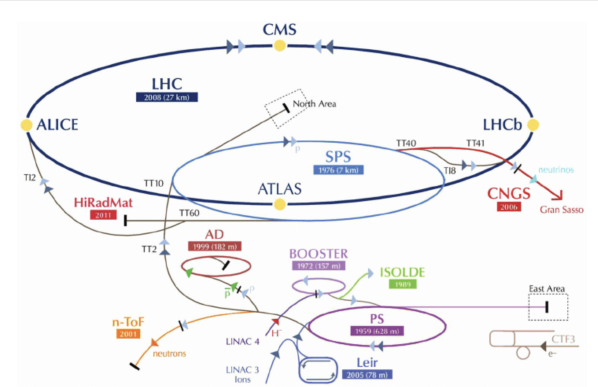
https://www.hep.uniovi.es/vischia/persistent/2025-10-26_TutorialECAI2025.html

to get the version with working animations

Disclaimer (who am I)

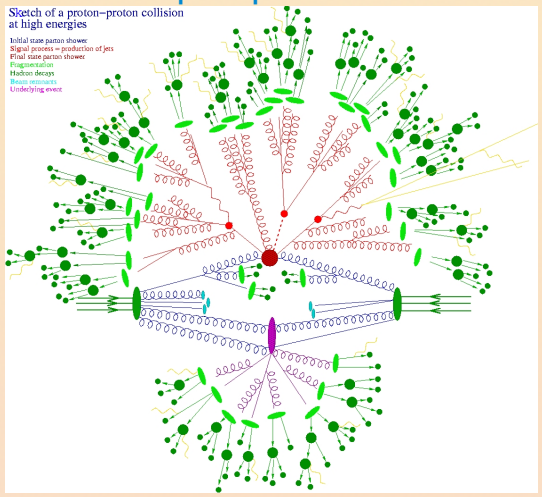
- Particle physicist (PhD 2016, IST Lisboa) now at Universidad de Oviedo and ICTEA
- Most of my career at the CMS Experiment at CERN
  - Specialized in statistics and machine learning applied to proton-proton collision data
  - We like frequentist properties
- In the past five years, I specialized in AI for experiment design
  - Check out the MODE Collaboration, https://mode-collaboration.github.io/
  - Lately focussing on neuromorphic computing (spiking neural networks) for nanophotonics readout of calorimeters

Complex accelerators



Existing CERN accelerator complex with Large Hadron Collider (LHC), Super Proton Synchrotron (SPS), Proton Synchrotron (PS), Antiproton Decelerator (AD), Low Energy Ion Ring (LEIR), Linear Accelerators (LINAC), CLIC Test Facility (CTF3), CERN to Gran Sasso (CNGS), Isotopes Separation on Line (ISOLDE), and neutrons Time of Flight (n-ToF).

Complex phenomena



Complex experiments and reconstruction

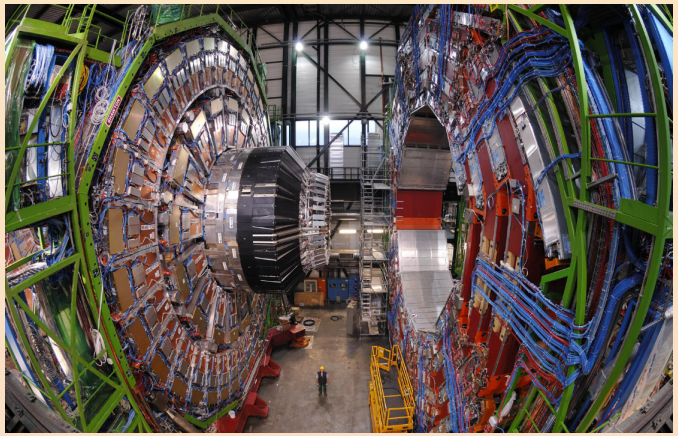
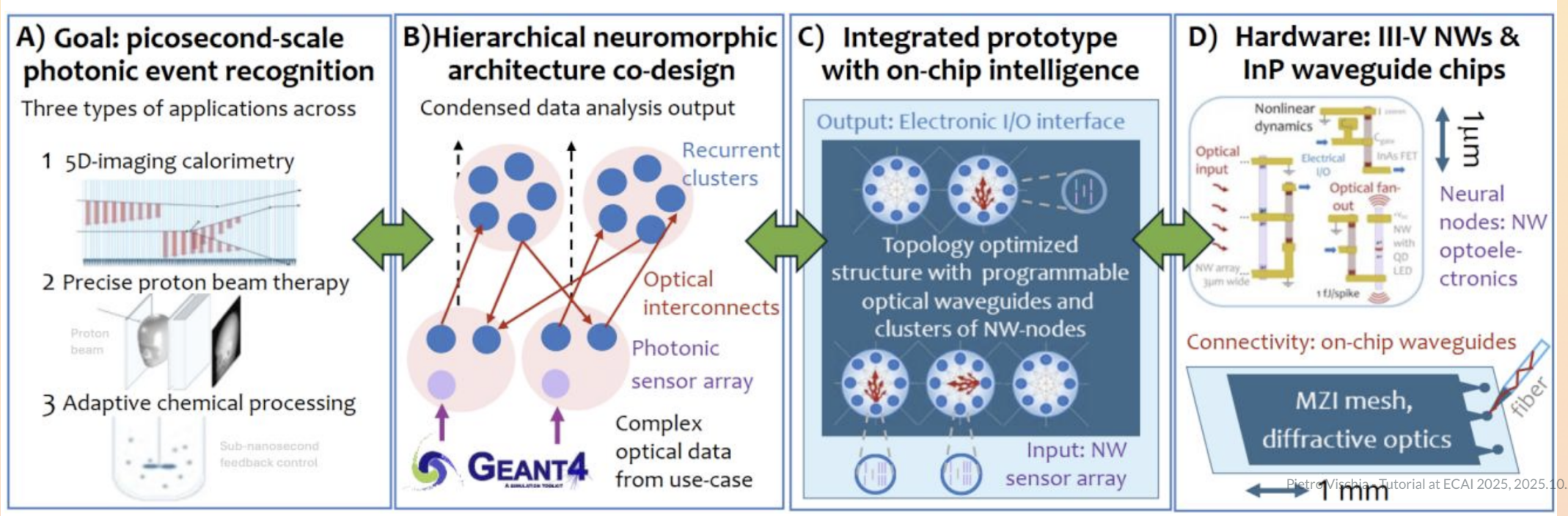


Image from 10.1103/PhysRevSTAB.16.054801 and from the CMS Collaboration

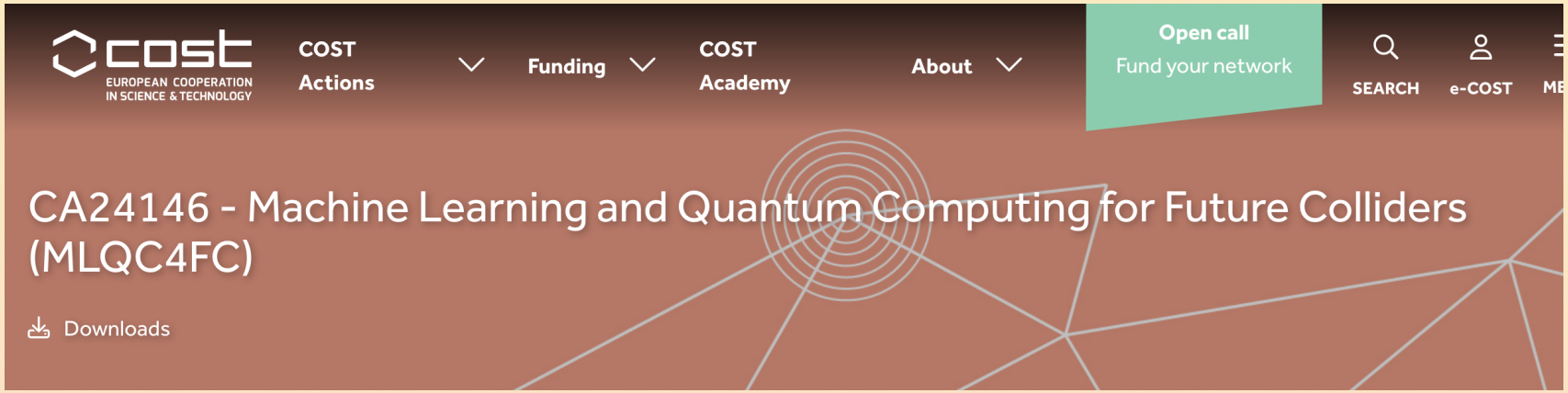
PHINDER Pathfinder Open (shameful advertisement part 1)

- Pathfinder Open 2025: Consortium just funded with 3.2 million euros
  - Will be hiring a PhD student on spiking neural networks and neuromorphic computing very soon: if you are interested, drop me a line!
- Picosecond-scale event processing with energy-efficient architectures
  - Particle physics detectors
  - Proton therapy detectors
  - Chemical process control

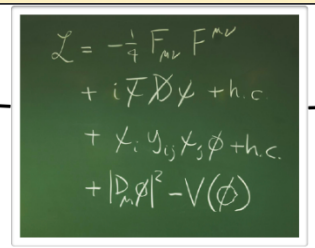


COST CA24146 (shameful advertisement part 2)

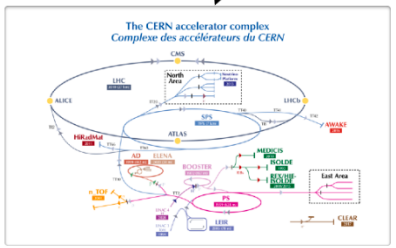
- WG1 coleader: Applications in Particle Physics
- Budget for travelling and organising events
- Anyone can join (just some reasonable loose acceptance criterion)



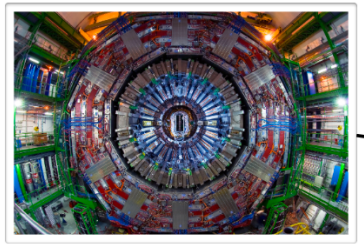
What do we do



First principle, quantum theoretical model

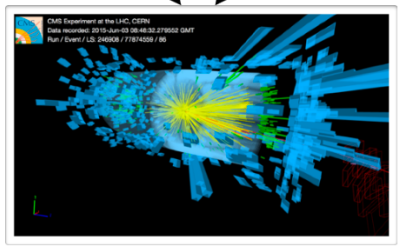


Colliders collecting 40 million events/s

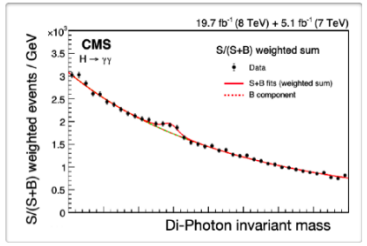


Detectors with 100 million readout channels

Massive theory-driven simulation codes



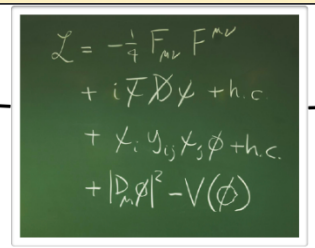
Complex reconstruction chain: from low-level readouts to high-level physics objects



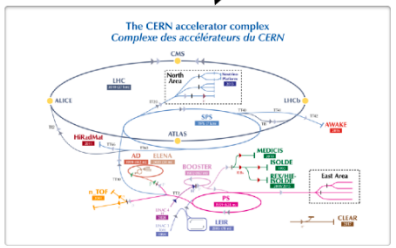
Sophisticated final statistical analysis

Illustration reedited from original by Gregor Kasieczka

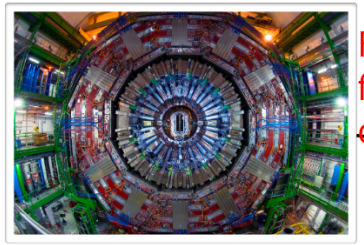
Where we can plug AI



Highlight symmetries, theorem proofing, model simplification

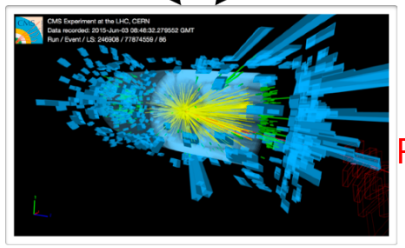


Online control, design of new accelerators

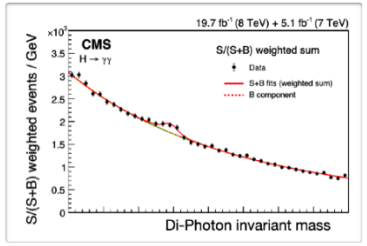


Fast AI-assisted inference for triggers, design of new detectors or experiments

AI-learned MonteCarlos surrogates, differentiable simulators



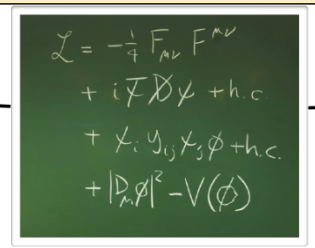
Reconstruction of physics objects, tagging of reconstructed objects



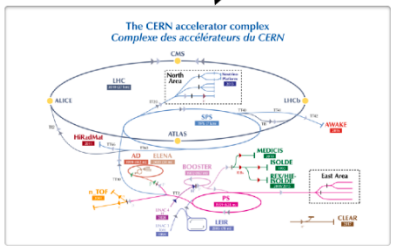
Unfolding, anomaly detection, signal extraction

Illustration reelaborated with some additions from original by Gregor Kasieczka

Where we can plug AI



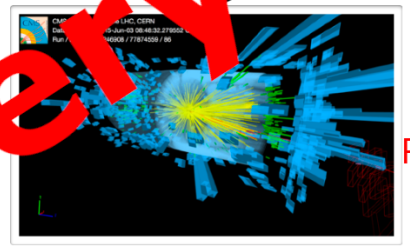
Highlight symmetries, theorem proofing, model simplification



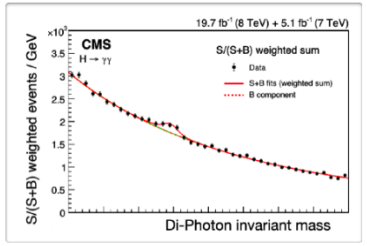
Online control, design of new accelerators

Fast AI-assisted inference for triggers, design of new detectors or experiments

AI-learned MonteCarlos surrogates, differentiable simulators



Reconstruction of physics objects, tagging of reconstructed objects



Unfolding, anomaly detection, signal extraction

Basically everywhere

Illustration reelaborated with some additions from original by Gregor Kasieczka

What is uncertainty?

- Aleatoric uncertainty: noise that is irreducible
  - "Statistical uncertainty", e.g. because of the stochasticity of a physical process
- Epistemic uncertainty: in the model itself
  - reduced by improving the model architecture or (sometimes) training on more data
- Out-of-distribution (OOD) uncertainty: response of the model to data that are significantly different from the training data set
  - Outliers
  - Sampled from a different distribution (but anomaly detection!)
  - Critical for model safety!

The program for this 1.5h

- Quantifying uncertainty
- Deployable techniques/algorithms
- Highlight the pitfalls and defects of these techniques
- Connect the various techniques (conformal prediction, calibration, regularization, and OOD) into a single pipeline

Why quantify uncertainty?

- Avoid catastrophic errors due to overconfidence in high-accuracy models
- Keeps the human in the loop: when uncertainty is large, abstain, request new data, or in general flag the outcome for human review
- Estimating uncertainty means estimating risk and therefore costs. In safety-critical settings, decisions are driven by costs, not accuracy only

Uncertainty estimates must be calibrated, robust, and computationally feasible

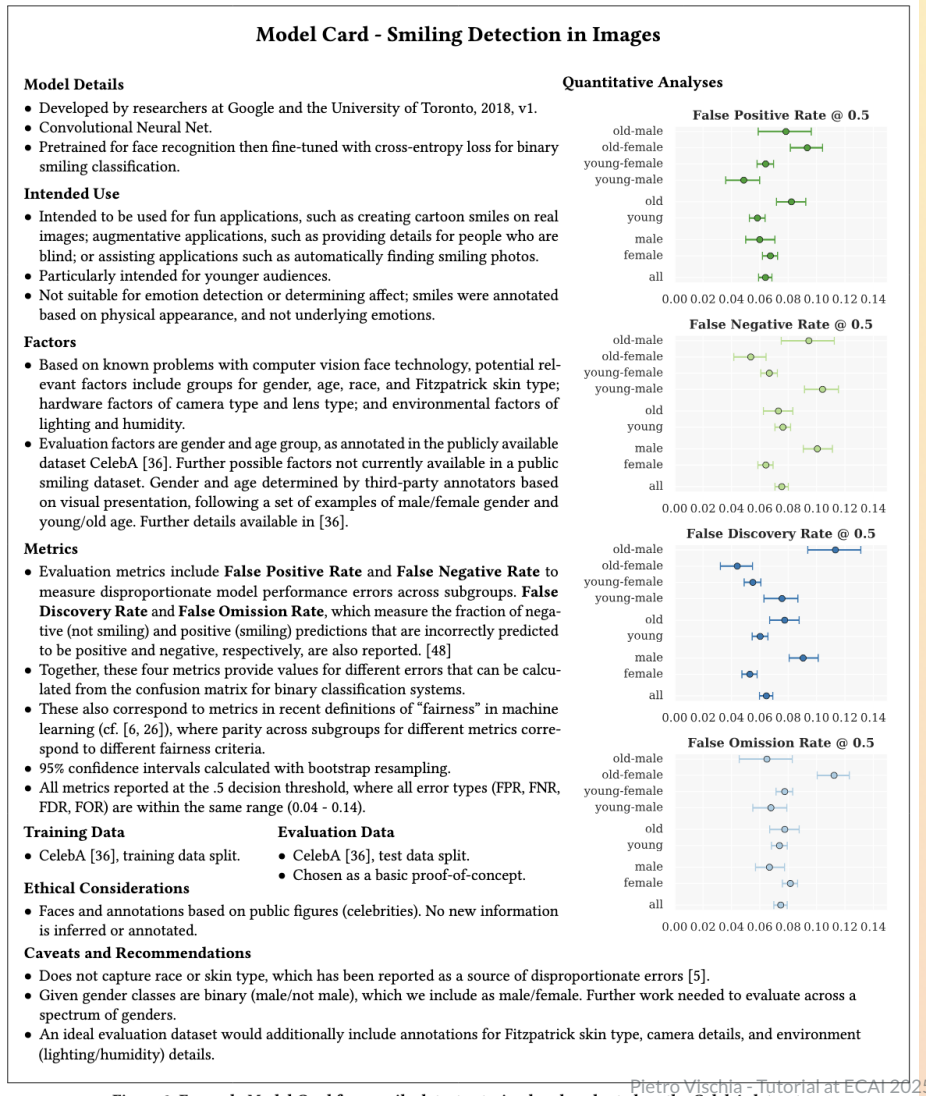
Further reading: Amodei et al, 2016: Concrete problems in AI Safety

What you will learn (hopefully)

- How to distinguish aleatoric and epistemic uncertainty and explain why the distinction matters.
- How to implement split conformal prediction and conformalized quantile regression with coverage guarantees
  - And why sometimes it's dangerous
- How regularization and inductive bias shape generalization and uncertainty
- How to filter out OOD data by OOD gating so that predictions are trusted only in-distribution

Communicate clearly about uncertainty

- End users should know when to trust the prediction
  - E.g. when to alert a nurse for human judgment
- A suitable pipeline should combine calibrated models, conformal intervals, and OOD gates to support these choices
- A first example are Model cards: short documents accompanying trained ML models, providing benchmarked evaluation in a variety of conditions
  - Across relevant groups (e.g. cultural, demographic, phenotipic) to the intended application domains



Model Card - Smiling Detection in Images

Model Details
- Developed by researchers at Google and the University of Toronto, 2018, v1.
- Convolutional Neural Net.
- Pretrained for face recognition then fine-tuned with cross-entropy loss for binary smiling classification.

Intended Use
- Intended to be used for fun applications, such as creating cartoon smiles on real images; augmentative applications, such as providing details for people who are blind; or assisting applications such as automatically finding smiling photos.
- Particularly intended for younger audiences.
- Not suitable for emotion detection or determining affect; smiles were annotated based on physical appearance, and not underlying emotions.

Factors
- Based on known problems with computer vision face technology, potential relevant factors include groups for gender, age, race, and Fitzpatrick skin type; hardware factors of camera type and lens type; and environmental factors of lighting and humidity.
- Evaluation factors are gender and age group, as annotated in the publicly available dataset CelebA [36]. Further possible factors not currently available in a public smiling dataset. Gender and age determined by third-party annotators based on visual presentation, following a set of examples of male/female gender and young/old age. Further details available in [36].

Metrics
- Evaluation metrics include **False Positive Rate** and **False Negative Rate** to measure disproportionate model performance errors across subgroups. **False Discovery Rate** and **False Omission Rate**, which measure the fraction of negative (not smiling) and positive (smiling) predictions that are incorrectly predicted to be positive and negative, respectively, are also reported. [48]
- Together, these four metrics provide values for different errors that can be calculated from the confusion matrix for binary classification systems.
- These also correspond to metrics in recent definitions of "fairness" in machine learning (cf. [6, 26]), where parity across subgroups for different metrics correspond to different fairness criteria.
- 95% confidence intervals calculated with bootstrap resampling.
- All metrics reported at the .5 decision threshold, where all error types (FPR, FNR, FDR, FOR) are within the same range (0.04 - 0.14).

Training Data
- CelebA [36], training data split.

Evaluation Data
- CelebA [36], test data split.
- Chosen as a basic proof-of-concept.

Ethical Considerations
- Faces and annotations based on public figures (celebrities). No new information is inferred or annotated.

Caveats and Recommendations
- Does not capture race or skin type, which has been reported as a source of disproportionate errors [5].
- Given gender classes are binary (male/not male), which we include as male/female. Further work needed to evaluate across a spectrum of genders.
- An ideal evaluation dataset would additionally include annotations for Fitzpatrick skin type, camera details, and environment (lighting/humidity) details.

Quantitative Analyses

Figure 2: Example Model Card for a smile detector trained and evaluated on the CelebA dataset.

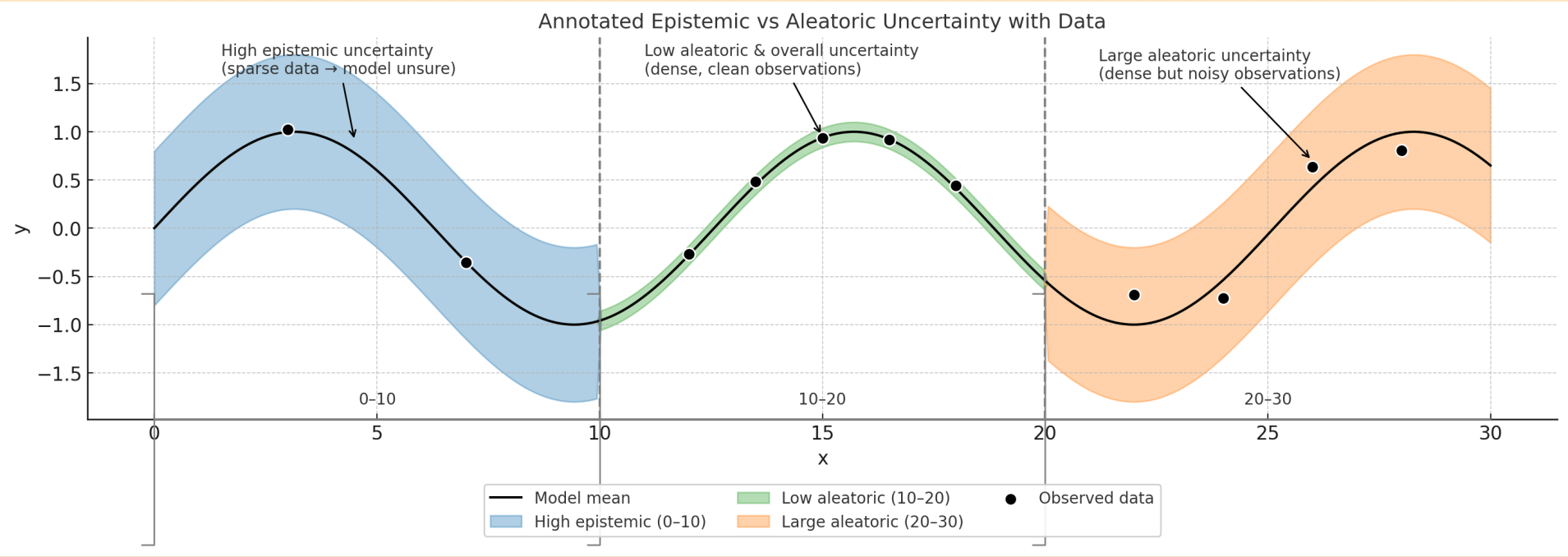
Figure from Mitchell et al., 2019

Outline

- Uncertainty foundations and calibration
- Conformal prediction and its main variants
- Regularization, inductive bias, and their interaction with uncertainty
- Out-of-Distribution (OOD) detection and the integration of all components into a pipeline

Aleatoric vs Epistemic uncertainty

- Aleatoric uncertainty: irreducible noise in the data (e.g. from a sensor)
  - Enlarging training dataset does not reduce it
  - Minimum achievable error (even with perfect dataset)
- Epistemic uncertainty: typically due to limited or nonrepresentative data
  - Enlarging training dataset reduces it
  - Requesting the model to extrapolate increases it
- Mixing the two leads to misguided interventions!!!



Do not mix aleatoric and epistemic uncertainties

- Earthquake load on a structure: S (peak ground acceleration) as intensity measure; R capacity to withstand a certain S.
  - R, S modelled as Poisson models
- $g(r, s, \epsilon) = ln(r) + \epsilon_1 - ln(s) + \epsilon_2$
  - E_1 are the errors in modelling the structure
  - E_2 are the errors due to the stochastic ground motion

- For a single earthquake, no distinction: then maybe failure prob is Poisson:

$$\widetilde{P}_{f,P_{\mathrm{sn}}} = 1 - \exp(-\mu_\nu \tilde{p}_\mathrm{f} t)$$

- Actually, failure events in time are not statistically independent

$$\widetilde{P}_\mathrm{f} = 1 - \int_{r,\varepsilon_1,\boldsymbol{\theta}} \exp\left[-\nu\Phi\left(-\frac{\ln r + \varepsilon_1 - \lambda_S}{\sqrt{\zeta_S^2 + \sigma_2^2}}\right) t\right] f_R(r) f_{E_1}(\varepsilon_1) f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \mathrm{d}r\, \mathrm{d}\varepsilon_1\, \mathrm{d}\boldsymbol{\theta}$$

  - Aleatory uncertainties in S and E_2 are renewed at each earthquake, all the others are common

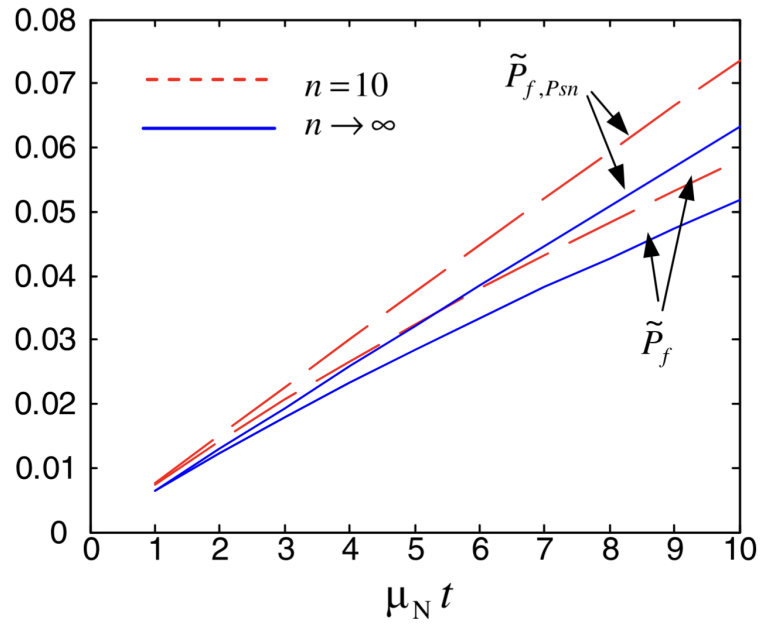


Fig. 5. Influence of non-ergodic uncertainties on time-variant reliability.

Example and plots from Kiureghian & Ditlevsen, 2009

Probabilistic predictions FTW

- A predictive distribution summarizes beliefs about future outcomes, not just a point estimate
  - In Bayesian terms, you want the full posterior
- Medical diagnosis: false positives (distress, further tests) preferrable to false negatives (patient dies due to being untreated)

Figure 1.25 An example of a loss matrix with elements L_{kj} for the cancer treatment problem. The rows correspond to the true class, whereas the columns correspond to the assignment of class made by our decision criterion.

$$\begin{array}{c} \\ \text{cancer} \\ \text{normal} \end{array} \begin{array}{c} \begin{array}{cc} \text{cancer} & \text{normal} \end{array} \\ \begin{pmatrix} 0 & 1000 \\ 1 & 0 \end{pmatrix} \end{array}$$

- Probabilistic predictions minimise risk
  - When loss matrix needs to be revisited, having the full posterior allows revising the decision criterion without having to retrain
- Probabilistic predictions allow us to determine a rejection criterion
  - e.g. minimise expected loss for a given fraction of rejected data points (connected to coverage, see later)
- Probabilistic predictions allow us to compensate for class priors
  - Incidence of disease in population dramatically shifts the posterior probability
- Probabilistic predictions allow us to combine models
  - E.g. break the problem in different subdiagnoses, then combine posteriors

Sharpness measures concentration, while calibration measures honesty of probabilities.

In practice, we prefer predictions that are both sharp and well calibrated.

Figure from C. Bishop, "Pattern Recognition and Machine Learning", 2006

(The Obligatory COVID-19 slide)

- Mortal disease
  - D: the patient is diseased (sick)
  - H: the patient is healthy
- Diagnostic test
  - $+$: the patient flags positive to the disease
  - $-$: the patient flags negative to the disease
- A very good test
  - $P(+|D) = 0.99$
  - $P(+|H) = 0.01$
- You take the test and you flag positive: do you have the disease?

$$P(D|+) = \frac{P(+|D)P(D)}{P(+)} = \frac{P(+|D)P(D)}{P(+|D)P(D)+P(+|H)P(H)}$$

- We need the incidence of the disease in the population, $P(D)$!
  - $P(D) = 0.001$ (very rare disease): then $P(D|+) = 0.0902$, which is fairly small
  - $P(D) = 0.01$ (only a factor 10 more likely): then $P(D|+) = 0.50$, which is pretty high
  - $P(D) = 0.1$: then $P(D|+) = 0.92$, almost certainty!

(Loss function comes from inference)

- Decision-theoretic approach (C.P. Robert, "The Bayesian Choice")
  - $\mathcal{X}$: observation space
  - Θ: parameter space
  - $\mathcal{D}$: decision (action) space
- Statistical inference take a decision $d \in \mathcal{D}$ related to parameter $\theta \in \Theta$ based on observation $x \in \mathcal{X}$, under $f(x|\theta)$
  - Typically, d consists in estimating $h(\theta)$ accurately

$$U(\theta, d) = \mathbb{E}_{\theta,d}\Big[U(r)\Big]$$

(Loss function comes from inference)

- Loss function: $L(\theta, d) = -U(\theta, d)$
  - Represents intuitively the loss or error in which you incur when you make a bad decision (a bad estimation of the target function)
  - Lower bound at 0: avoids "infinite utility" paradoxes (St. Petersburg paradox, martingale-based stragegies)
- Generally impossible to uniformly minimize in d the loss for θ unknown
  - Need for a practical prescription to use the loss function as a comparison criterion in practice

(Frequentist loss, Bayesian loss)

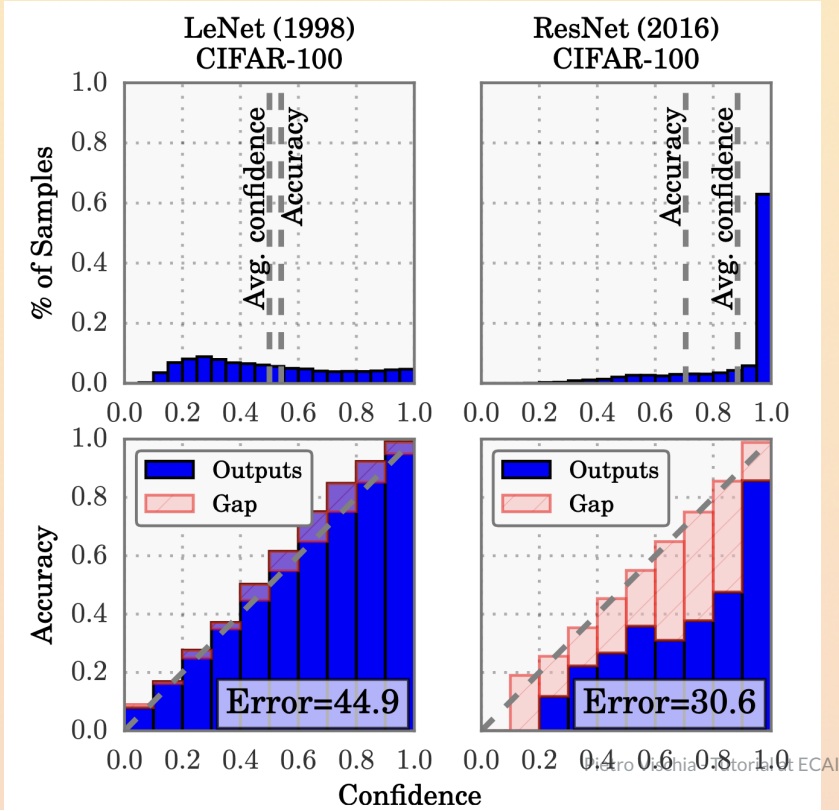
- Frequentist loss (risk) is integrated (averaged) on $\mathcal{X}$: $R(\theta, \delta) = \mathbb{E}_\theta \Big[L(\theta, \delta(x)) \Big]$
  - $\delta(\cdot)$ is an \textbf{estimator} of θ (e.g. MLE)
  - Compare estimators, find the best estimator based on long-run performance for all values of unknown θ
  - Issues: based on long run performance (not optimal for x_{obs}); repeatability of the experiment; no total ordering on the set of estimators
- Bayesian loss: is integrated on Θ: $\rho(\pi, d|x) = \mathbb{E}^\pi \Big[L(\theta, d)|x \Big]$
  - π is the prior distribution
  - Posterior expected loss averages the error over the posterior distribution of θ conditional on x_{obs}
  - Can use the conditionality because x_{obs} is known!
  - Can also integrate the frequentist risk; integrated risk $r(\pi, \delta) = \mathbb{E}^\pi \Big[R(\theta, \delta) \Big]$ averaged over θ according to π (total ordering)

Calibration: the problem

- A network should provide a calibrated confidence in addition to prediction
- Calibrated means the probability associated with the predicted label should reflect its ground truth correctness likelihood
  - **Calibration and accuracy are orthogonal!!!** Plus, calibration is intrinsically a frequentist concept
  - Miscalibration is further exacerbated by distribution shift (test data deviates from the training distribution due to environmental or acquisition changes)
- Networks in early 2000s were reasonably calibrated for binary classification (Niculescu-Mizil and Caruana, 2005)

- Reliability Diagram (visual inspection)
- Expected Calibration Error (summary statistic)
  - "Gap" in the plot

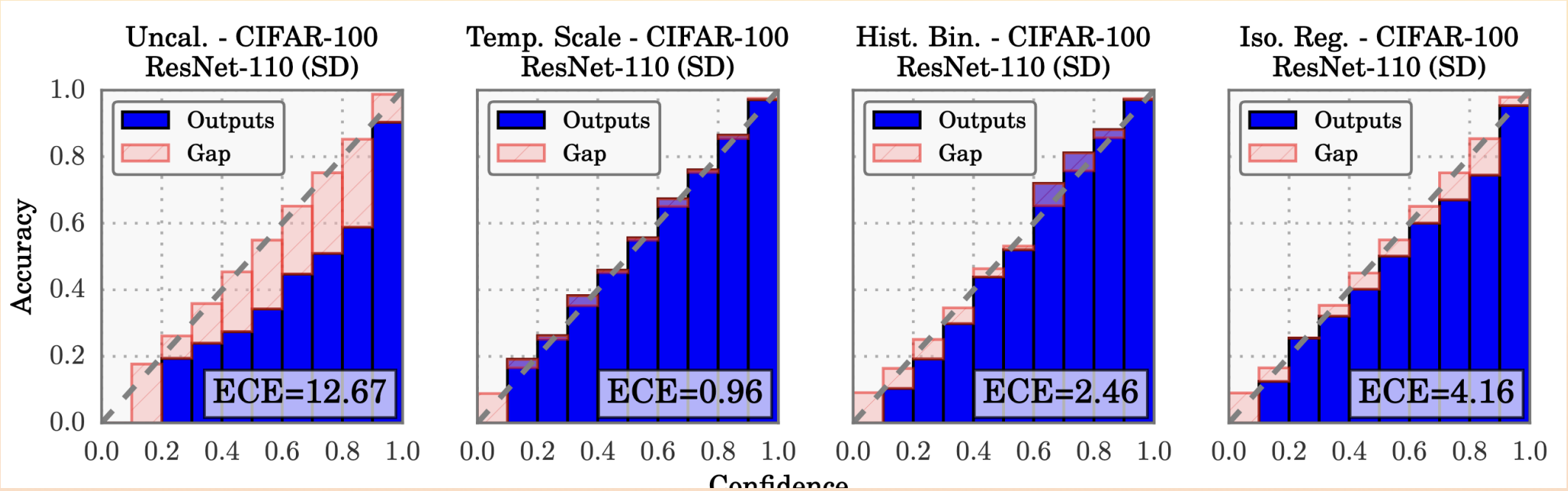
$$ECE = \sum_{m=1}^{M} \frac{|B_m|}{n} \left| acc(B_m) - conf(B_m) \right|$$



Plots from Guo et al. (2017)

Calibration: the solution

- Histogram binning: calculate calibrated prediction in each bin (on a holdout set), then assign it to predictions falling into that bin
- Isotonic regression: learn piecewise constant $f = argmin_f \sum_{i=1}^{n} (f(\hat{p}_i - y_i)^2$ (generalization of hist., where also boundaries are optimised jointly)
- Temperature scaling: logistic regression on the logits is trained to output calibrated probabilities. Changes confidence but not accuracy
- Other methods in reference below. All scale linearly with n validation samples



Plots from Guo et al. (2017)

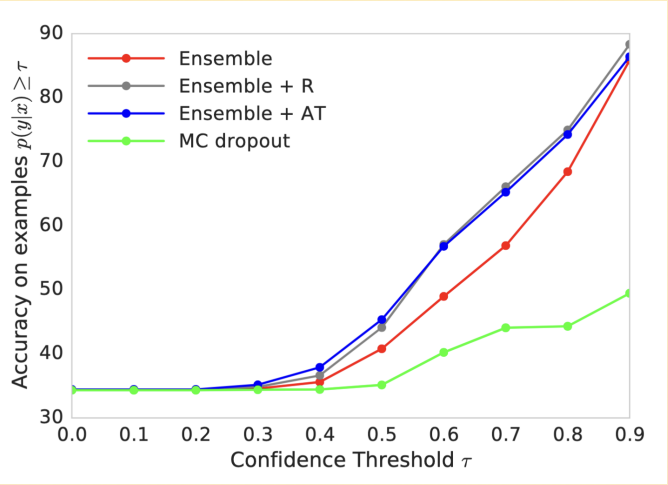
Uncertainty quantification

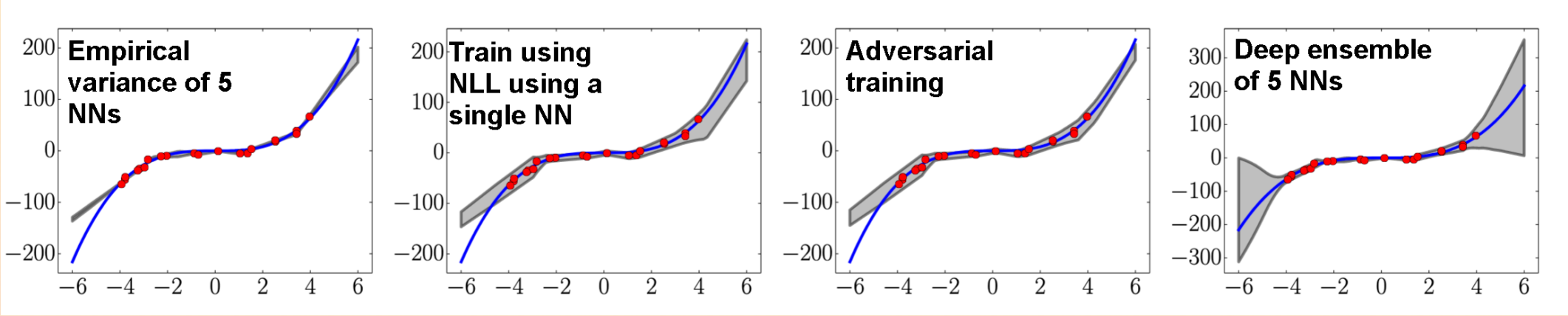
- Deep ensembles: average diverse models to approximate epistemic uncertainty and improve calibration
- SWAG: fit a Gaussian to the SGD trajectory to cheaply sample weights
- Laplace approximations: (often last-layer) provide a lightweight Bayesian head

- Choose by cost–fidelity trade-off and validate with decision metrics.

Deep Ensembles Lakshminarayanan et al., 2017

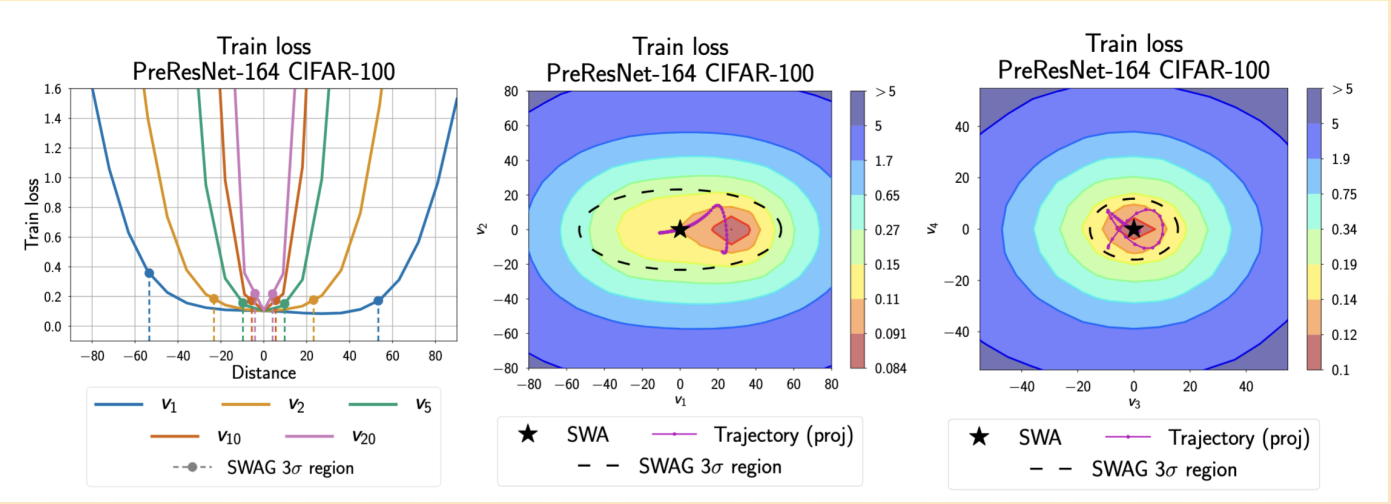
- Ensemble of models to take decision (mixture model of posterior distributions)
  - Randomization (same as in random forests) and bagging particularly efficient because of full parallelization
  - Can also do boosting, but less efficient (sequential fit)
- Randomization can be tricky with Neural Networks (multiple local optima)
  - Breiman (2001) proposes bagging and random feature selection
- Adversarial training to smooth predictive distributions
- Beats Monte Carlo Dropout ("use dropout for test sample")
- Quite robust and rather well calibrated



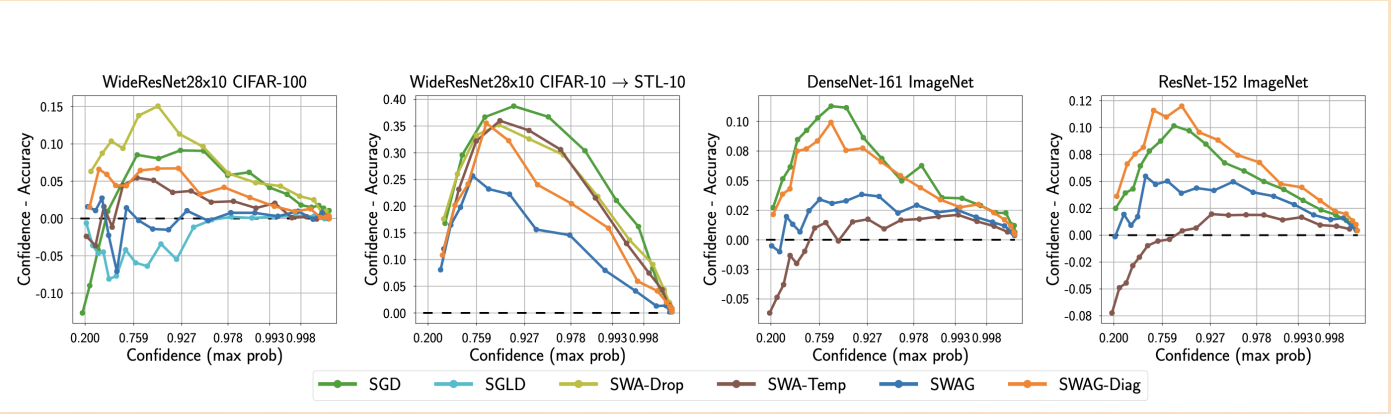


SWAG (SWA-Gaussian) Maddox et al., 2019

- Use information from the SGD trajectory to approximate posterior distribution of the NN weights
  - Good approximation of the posterior using a Gaussian distribution fitted to the first two moments of SGD iterates

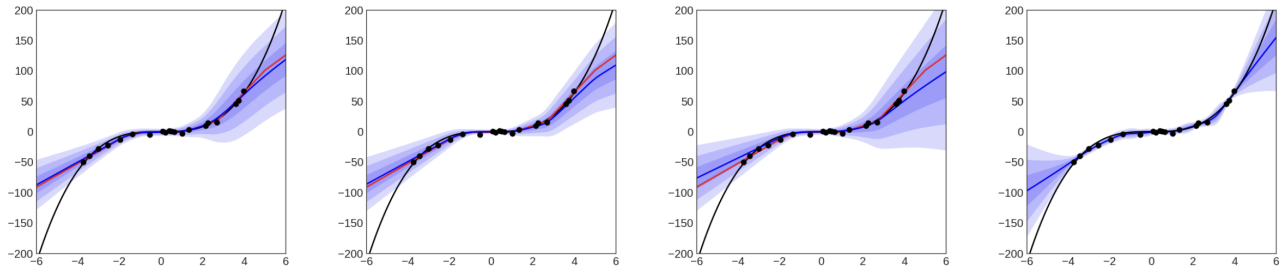


- Improves calibration when compared to SGD and simple SWA (Stochastic Weight Averaging)



Laplace Approximation Ritter et al., 2018

- Kronecker Factored (KFAC) Laplace approximation to the posterior of the trained network weights
  - Factorization of the Hessian w.r.t. parameters
  - **Can be calculated without retraining!!!** Last-layer Laplace provides a light Bayesian head for frozen feature extractors.
  - Approximates well posterior from 50000 HMC samples



(a) KF Laplace (b) Diagonal Laplace (c) Full Laplace (d) HMC

Figure 1: Toy regression uncertainty. Black dots are data points, the black line shows the noiseless function. The red line shows the deterministic prediction of the network, the blue line the mean output. Each shade of blue visualises one additional standard deviation. Best viewed on screen.

- Almost no prediction with absolute certainty

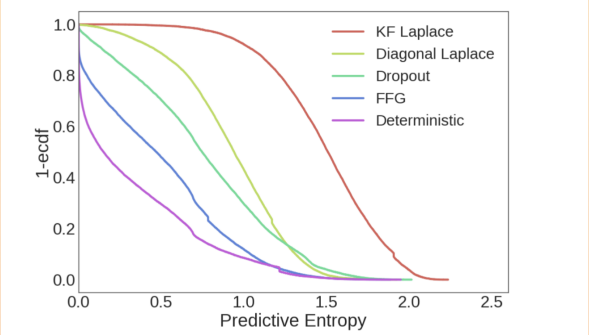
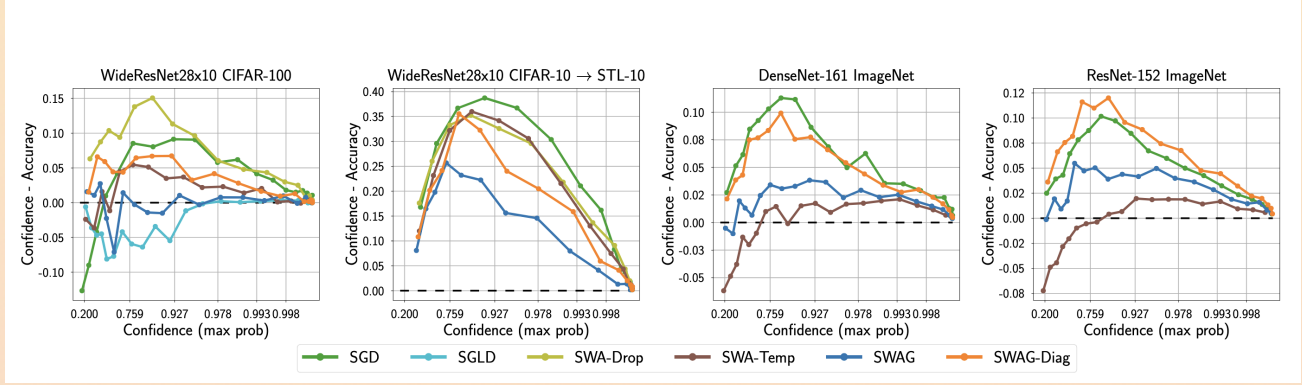


Figure 2: Predictive entropy on notMNIST ob-

- Inflates uncertainty for larger attacks



Conformal prediction: the idea behind it

- Conformal prediction (Vovk, Gammerman, Shafer (2005)) wraps any point predictor with finite-sample coverage guarantees
  - Converts calibration set scores into intervals or sets for new predictions.
  - Essential assumption is exchangeability of calibration and test examples
- Model-agnostic, therefore easy to retrofit onto existing systems.
- Transductive: make a prediction, retrain using this new prediction: do next prediction using calibration set of $n + 1$ data points

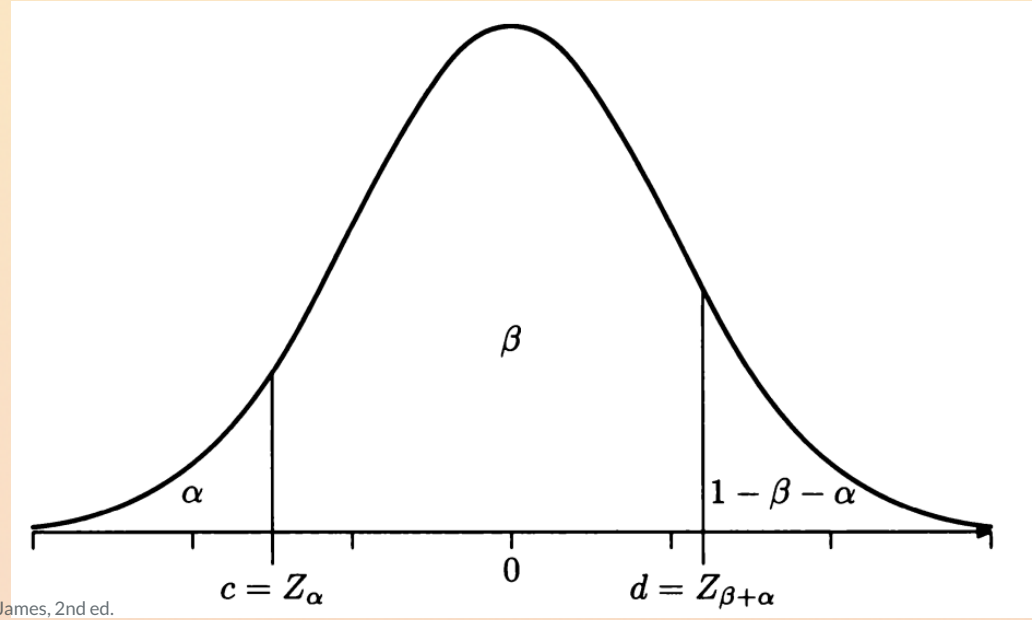
Figure 1: Prediction set examples on Imagenet. *We show three progressively more difficult examples of the class* `fox squirrel` *and the prediction sets (i.e., $C(X_{\text{test}})$) generated by conformal prediction.*

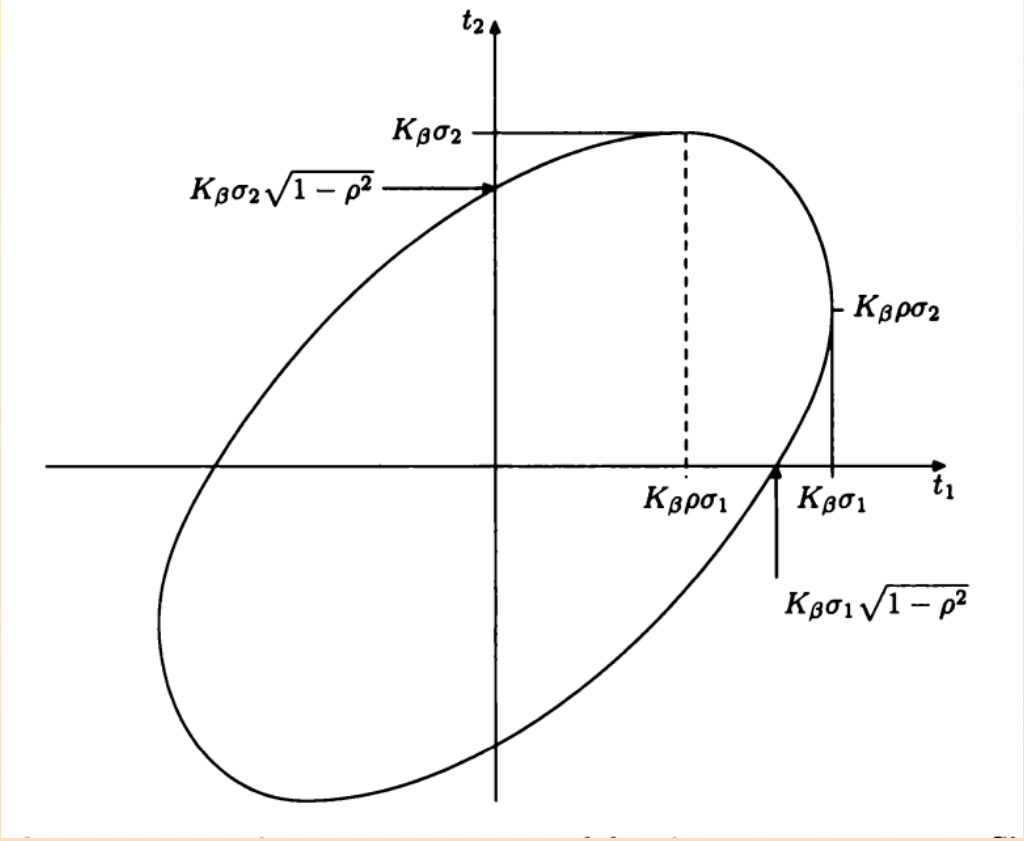
Figures from Angeloupoulos and Bates, 2021

A note on confidence intervals

- Probability content: solve $\beta = P(a \leq X \leq b) = \int_a^b f(X|\theta)dX$ for a and b
  - A method yielding interval with the desired β, has coverage

- Interpretation of fixed probability content
  $P\Big((\theta_{MLE} - \theta_{true})^2 \leq \sigma)\Big) = \beta$
  $P(-\sigma \leq \theta_{MLE} - \theta_{true} \leq \sigma) = \beta$
  $P(\theta_{MLE} - \sigma \leq \theta_{true} \leq \theta_{MLE} + \sigma) = \beta$





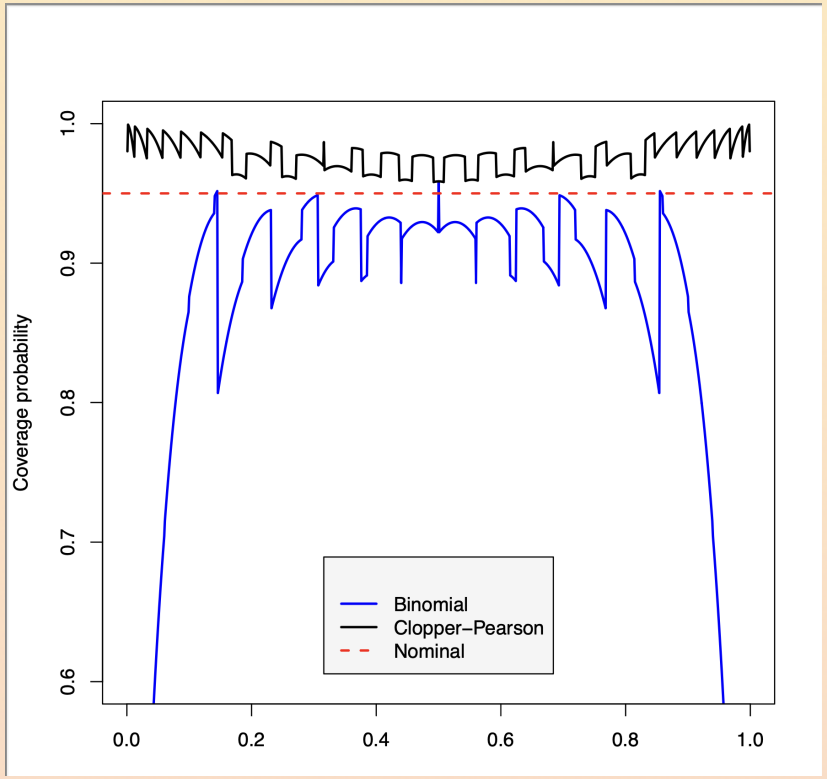
Plots from James, 2nd ed.

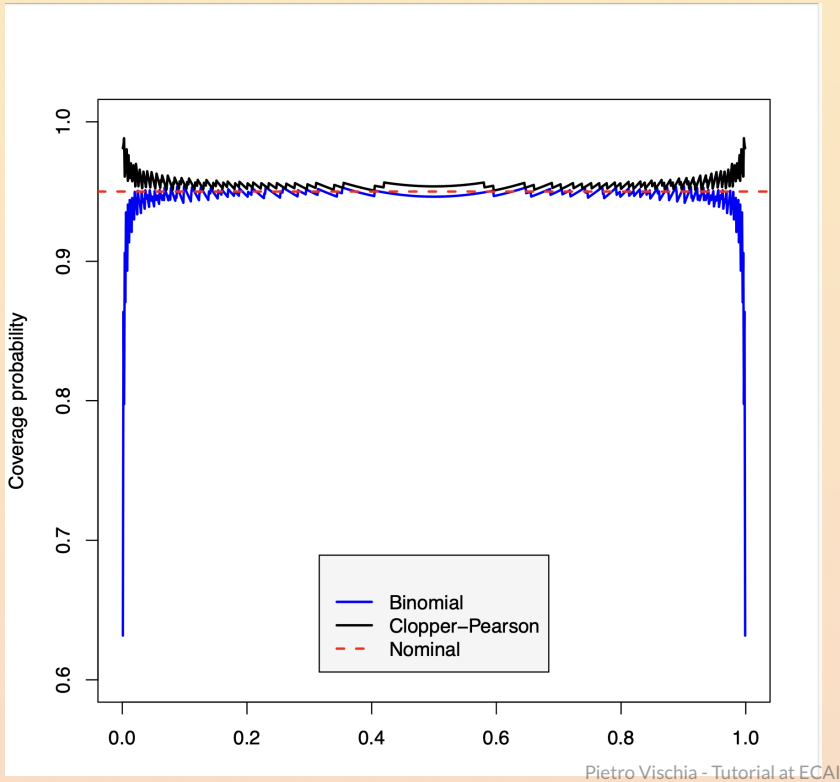
A note on coverage

- Operative definition of coverage probability
  - Fraction of times, over a set of (usually hypothetical) measurements, that the resulting interval covers the true value of the parameter
  - Obtain the sampling distribution of the confidence intervals using toy data
- Nominal coverage: the one you have built your method around
- Actual coverage: the one you calculate from the sampling distribution
  - Toy experiment: sample N times for a known value of θ_{true}
  - Compute interval for each experiment
  - Count fractions of intervals containing θ_{true}
- Nominal and actual coverage should agre if all assumptions of method are valid
  - Undercoverage: intervals smaller than proper ones
  - Overcoverage: intervals larger than proper ones

Coverage: the discrete Case

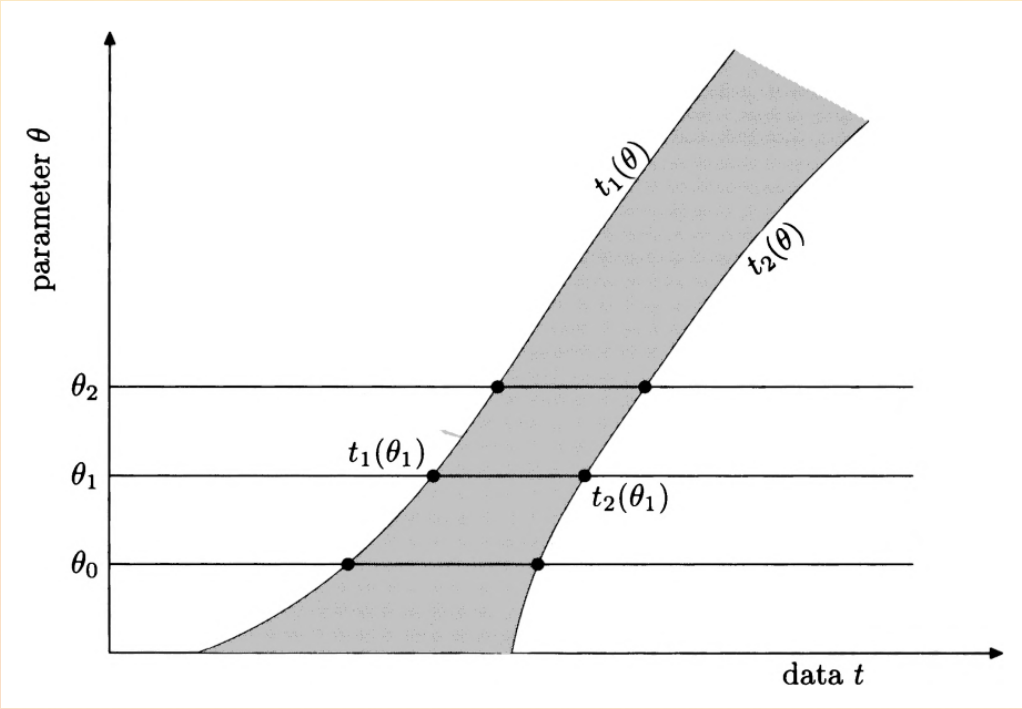
- Probability content $P(a \leq X \leq b) = \sum_a^b f(X|\theta)dX \leq \beta$
- Binomial: find (r_{low}, r_{high}) such that $\sum_{r=r_{low}}^{r=r_{high}} \binom{r}{N} p^r (1-p)^{N-r} \leq 1-\alpha$
  - Gaussian approximation: $p \pm Z_{1-\alpha/2}\sqrt{\frac{p(1-p)}{N}}$
  - Clopper Pearson: invert two single-tailed binomial tests

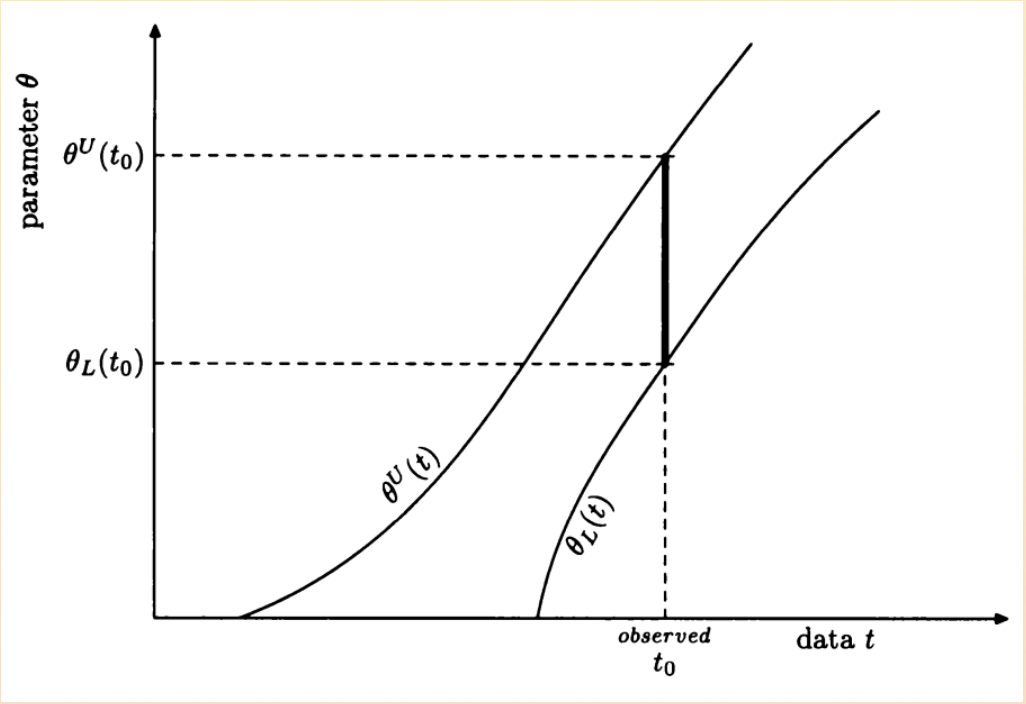




(The Neyman construction)

- Unique solutions to finding confidence intervals are infinite
  - Let's suppose we have chosen a way
- Build horizontally: for each (hypothetical) value of θ, determine $t_1(\theta), t_2(\theta)$ such that $\int_{t_1}^{t_2} P(t|\theta)dt = \beta$
- Read vertically: from the observed value t_0, determine $[\theta_L, \theta^U]$ by intersection
- Intrinsically frequentist procedure



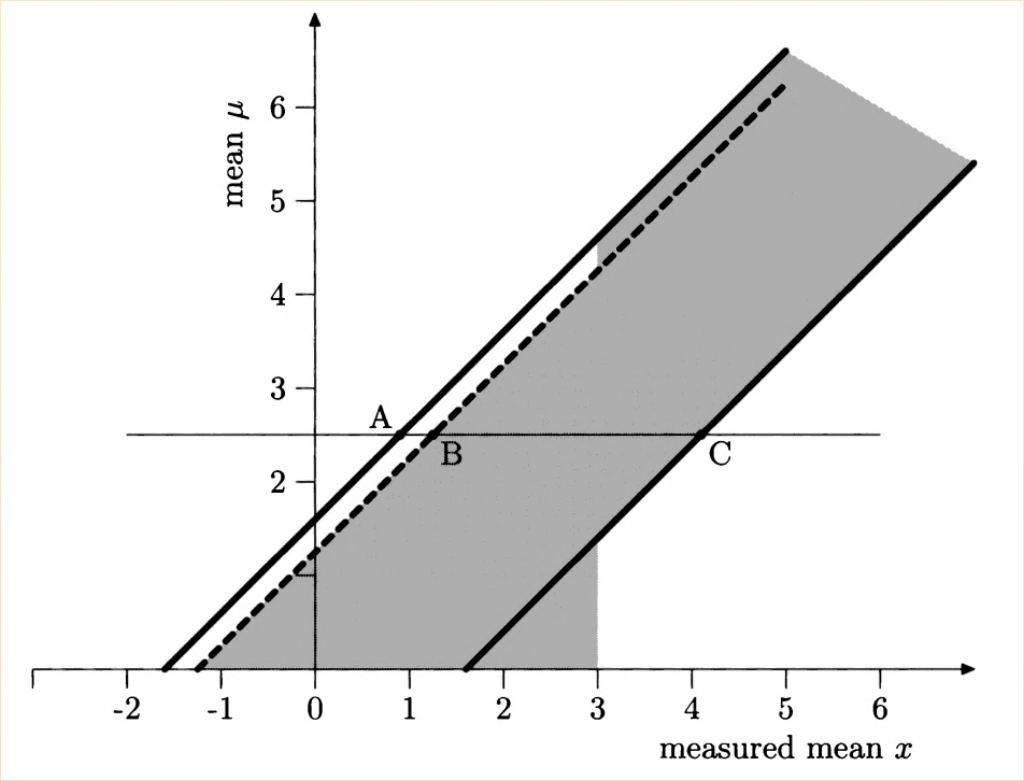


Figures from James, 2nd ed.

(Flip-flopping)

- Gaussian measurement (variance 1) of $\mu > 0$ (physical bound)
- Individual prescriptions are self-consistent
  - 90% central limit (solid lines)
  - 90% upper limit (single dashed line)

- Mixed choices (after looking at data) are problematic
- Unphysical values and empty intervals: choose 90% central interval, measure $x_{obs} = -2.0$
  - Interval empty, yet with the desired coverage

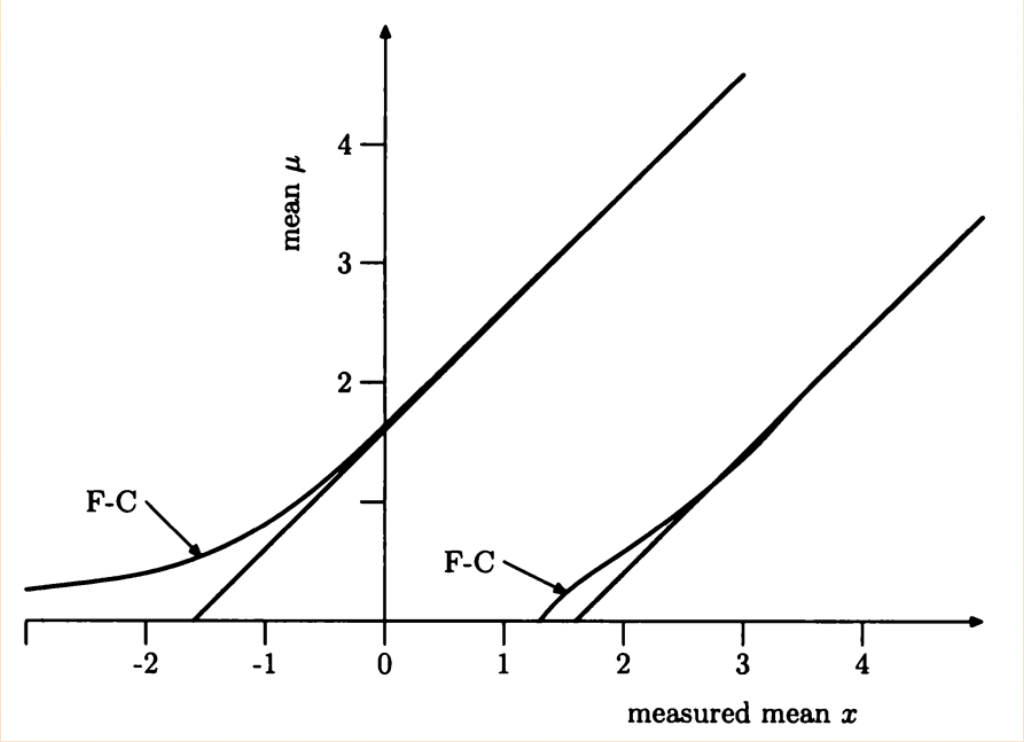


Figures from James, 2nd ed.

(The Feldman-Cousins Ordering Principle)

- Unified approach for determining interval for $\mu = \mu_0$
  - Include in order by largest $\ell(x) = \frac{P(x|\mu_0)}{P(x|\hat{\mu})}$
  - $\hat{\mu}$ value of μ which maximizes $P(x|\mu)$ within the physical region
  - $\hat{\mu}$ remains equal to zero for $\mu < 1.65$, yielding deviation w.r.t. central intervals

- Minimizes Type II error (likelihood ratio for simple test is the most powerful test)
- Solves the problem of empty intervals
- Avoids flip-flopping in choosing an ordering prescription

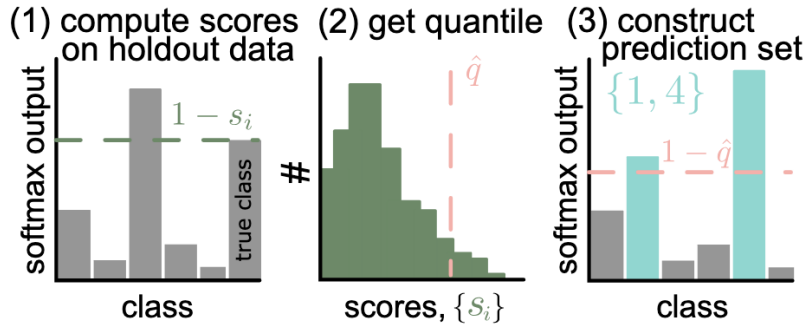


Split conformal workflow

- Fit a predictor $\hat{f}$ to training data
- Create a prediction set (a set of possible labels) using a small calibration set, such that:

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

- Now the probability that the prediction set contains the true label is almost exactly $1 - \alpha$ (marginal coverage, averaged over calibration and test points)
  - $\mathcal{C}$ is built by 1) Compute "nonconformity scores" on the calibration set using the trained model; 2) Take a quantile of those scores to determine how much to inflate predictions. 3) Output intervals for new points using that quantile as slack.

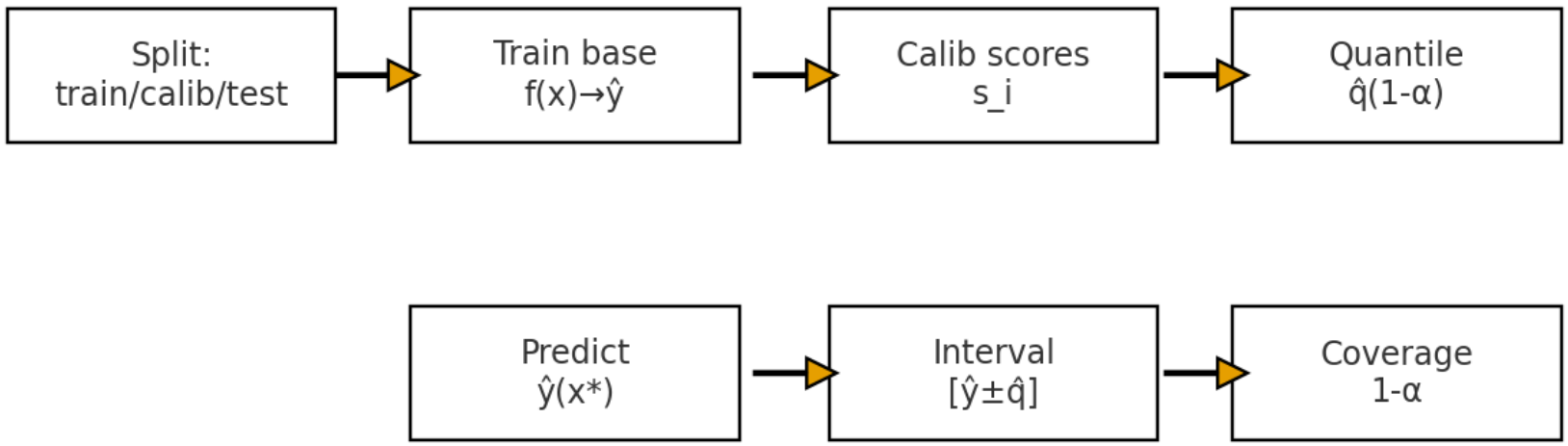



```python
# 1: get conformal scores. n = calib_Y.shape[0]
cal_smx = model(calib_X).softmax(dim=1).numpy()
cal_scores = 1-cal_smx[np.arange(n),cal_labels]
# 2: get adjusted quantile
q_level = np.ceil((n+1)*(1-alpha))/n
qhat = np.quantile(cal_scores, q_level, method='higher')
val_smx = model(val_X).softmax(dim=1).numpy()
prediction_sets = val_smx >= (1-qhat) # 3: form prediction sets
```

Figure 2: Illustration of conformal prediction *with matching Python code.*

Figures from Angeloupoulos and Bates, 2021

Conformal pipeline

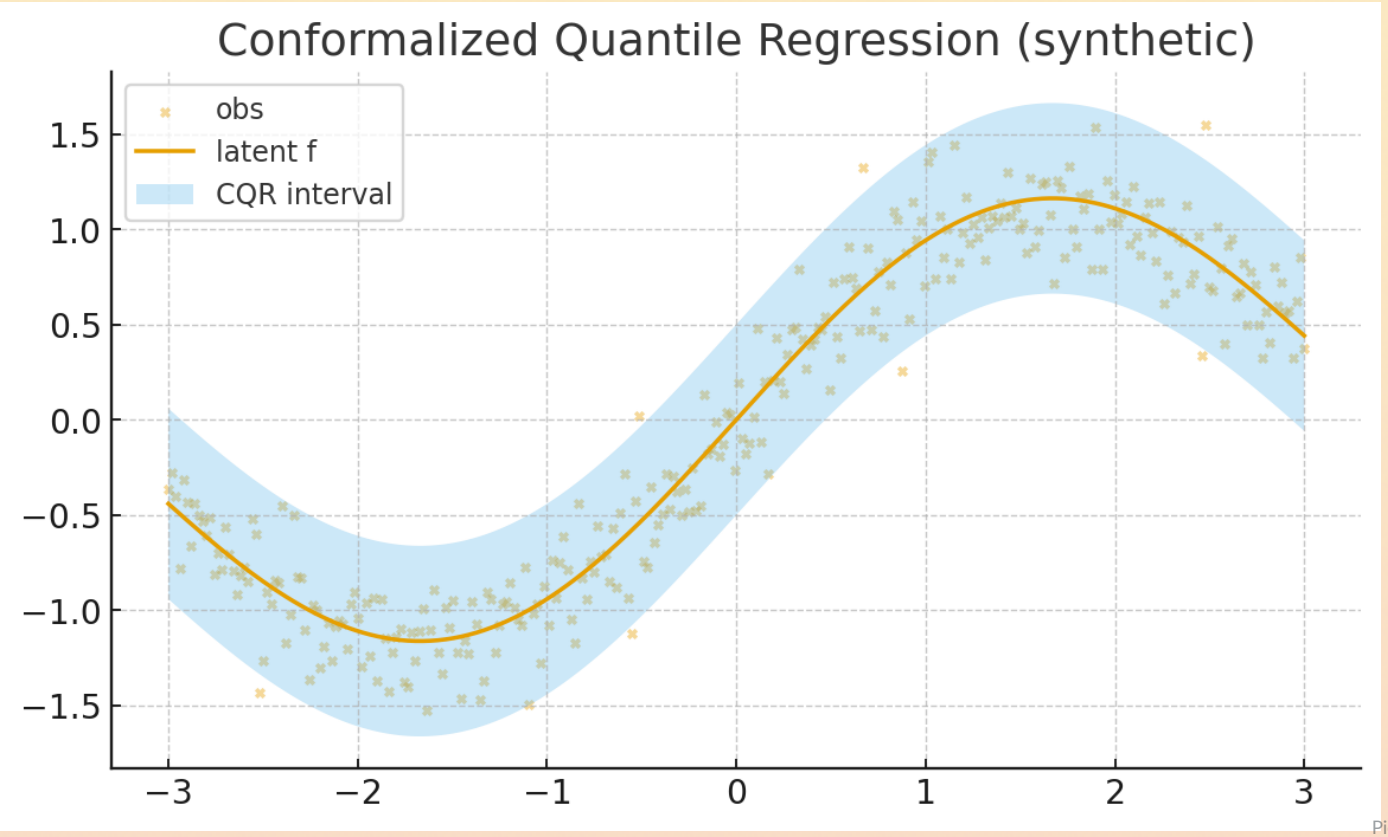


Model-agnostic; finite-sample under exchangeability.

Nonconformity scores for regression

- For symmetric intervals, a natural score is the absolute residual.
- Alternative scores exist for asymmetric losses or heavy-tailed noise.
- Score choice affects efficiency but not validity under exchangeability.

We will visualize the effect of different scores in the tutorial.



Conformal prediction have approximate coverage

- Marginal coverage at level $1 - \alpha$ over new examples
- No requirement of any parametric assumptions about the data
- Still, coverage not guaranteed!
  - Sometimes it undercovers!

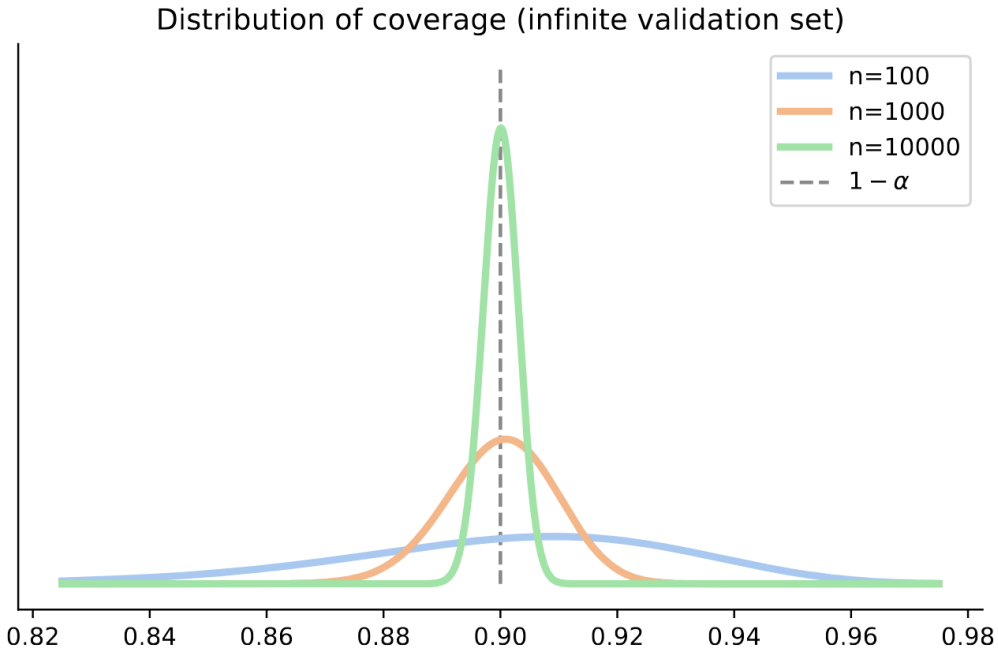


Figure 11: The distribution of coverage *with an infinite validation set is plotted for different values of* n *with* $\alpha = 0.1$*. The distribution converges to* $1 - \alpha$ *with rate* $\mathcal{O}\left(n^{-1/2}\right)$*.*

Figures from Angeloupoulos and Bates, 2021

Conformalized quantile regression (CQR) Romano et al., 2019

- Trains a model to predict lower and upper conditional quantiles and then calibrate slack (difference from target coverage)
- CQR retains marginal coverage while improving efficiency where quantiles are learnable

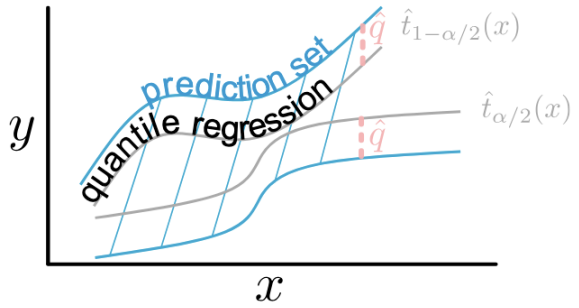
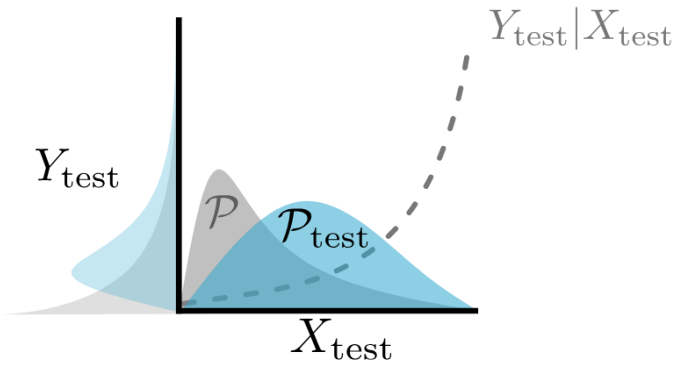


Figure 6: A visualization of the conformalized quantile regrssion algorithm *in Eq. (4). We adjust the quantiles by the constant $\hat{q}$, picked during the calibration step.*

Figures from Angeloupoulos and Bates, 2021

Conformal prediction and covariate shift

- Upweight conformal scores from calibration points that would be more likely under the new distribution, $w(x) = \frac{dP_{test}(x)}{dP/(x)}$
  - Weighted CP replaces the empirical quantile with a weighted quantile using density ratios.
  - Weights can be learned via a domain classifier and converted to ratios.



Imagine our calibration features $\{X_i\}_{i=1}^n$ are drawn independently from $\mathcal{P}$ but our test feature X_{test} is drawn from $\mathcal{P}_{\text{test}}$. Then, there has been a covariate shift, and the data are no longer i.i.d. This problem is common in the real world. For example,

Figures from Angeloupoulos and Bates, 2021

Jackknife+ for robustness

- Train models on fold complements and predicts both held-out points and the new input.
  - Aggregating bounds across folds reduces variance from a single split.
- Decide by coverage, training cost, evaluation cost

Method	Assumption-free theory	Typical empirical coverage
Naive (4)	No guarantee	$< 1 - \alpha$
Split conf. (holdout) (3)	$\geq 1 - \alpha$ coverage	$\approx 1 - \alpha$
Jackknife (7)	No guarantee	$\approx 1 - \alpha$, or $< 1 - \alpha$ if $\widehat{\mu}$ unstable
Jackknife+ (9)	$\geq 1 - 2\alpha$ coverage	$\approx 1 - \alpha$
Jackknife-minmax (10)	$\geq 1 - \alpha$ coverage	$> 1 - \alpha$
Full conformal (15)	$\geq 1 - \alpha$ coverage	$\approx 1 - \alpha$, or $> 1 - \alpha$ if $\widehat{\mu}$ overfits
K-fold CV+ (11)	$\geq 1 - 2\alpha$ coverage	$\gtrsim 1 - \alpha$
K-fold cross-conf. (12)	$\geq 1 - 2\alpha$ coverage	$\gtrsim 1 - \alpha$

Method	Model training cost	Model evaluation cost
Naive (4)	1	$n + n_{\text{test}}$
Split conf. (holdout) (3)	1	"
Jackknife (7)	n	"
Jackknife+ (9)	n	$n_{\text{test}} \cdot n$
Jackknife-minmax (10)	n	"
K-fold CV+ (11)	K	$n + n_{\text{test}} \cdot K$
K-fold cross-conf. (12)	K	"
Full conformal (15)	$n_{\text{test}} \cdot n_{\text{grid}}$	$n_{\text{test}} \cdot n_{\text{grid}} \cdot n$

Tables from Barber et al. (2019)

Pitfalls with conformal prediction

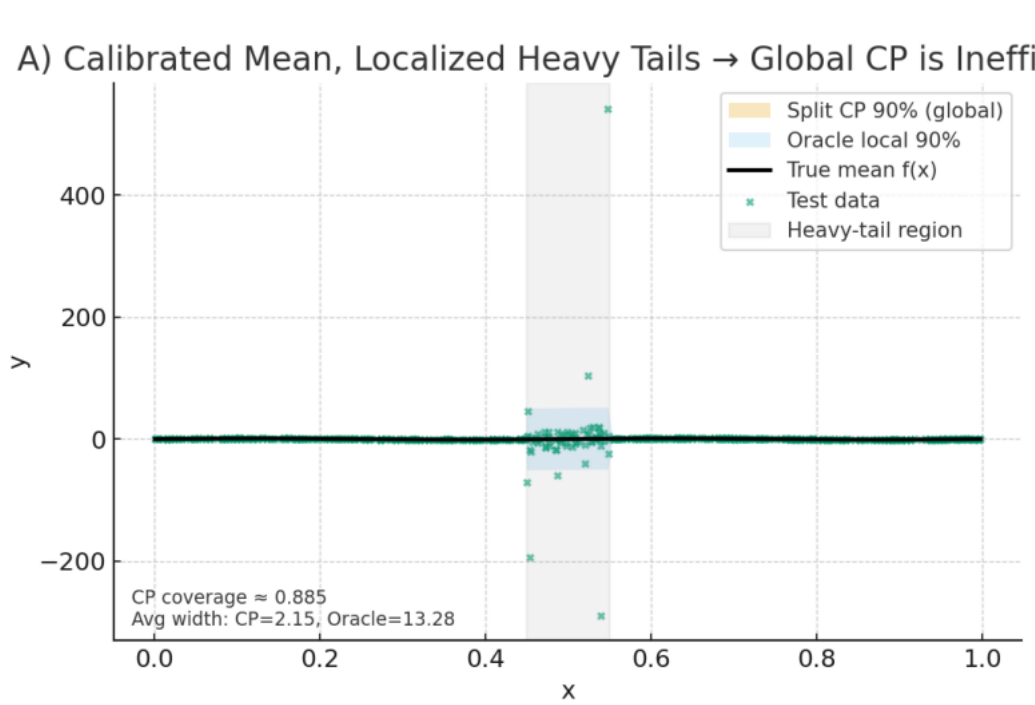
- Calibrating on leaked or preprocessed data invalidates coverage.
- Coverage is only marginal
- Results are in terms of covering sets: for binary classification, it doesn't work!
- Re-using the same calibration set for model selection biases intervals
- Ignoring covariate shift leads to optimistic coverage estimates

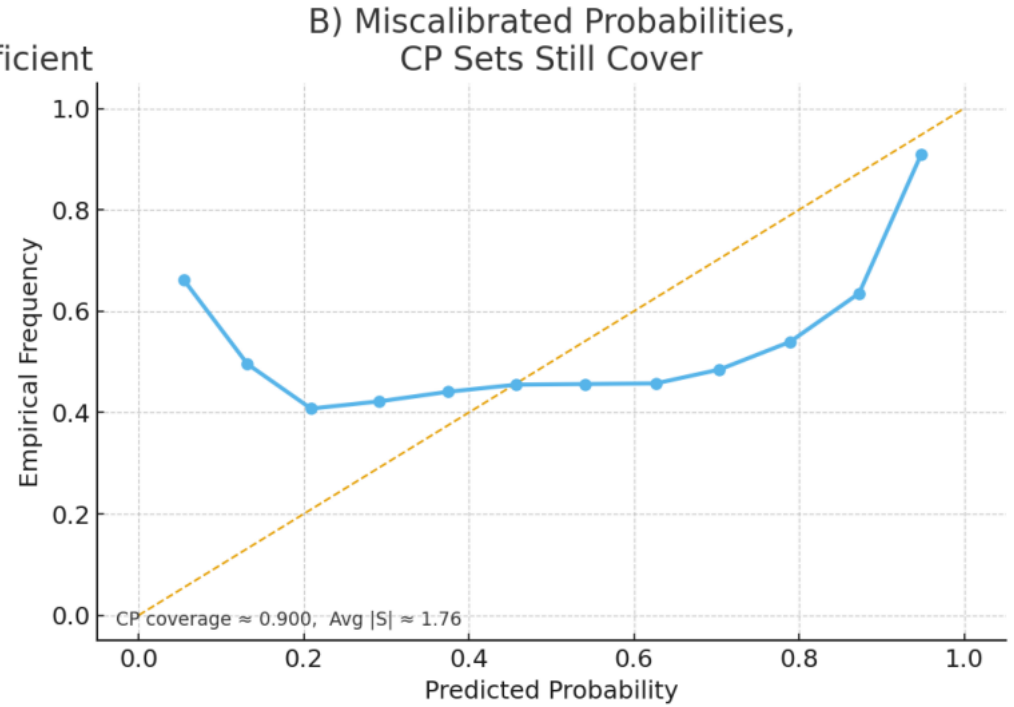
- Never believe individuals on the internet who are too enthusiastic about conformal prediction
  - Particularly if they happen to make a living by selling courses on it

Conformal prediction vs Calibration

- Conformal prediction gives coverage for sets
- Probability calibration targets probabilities.

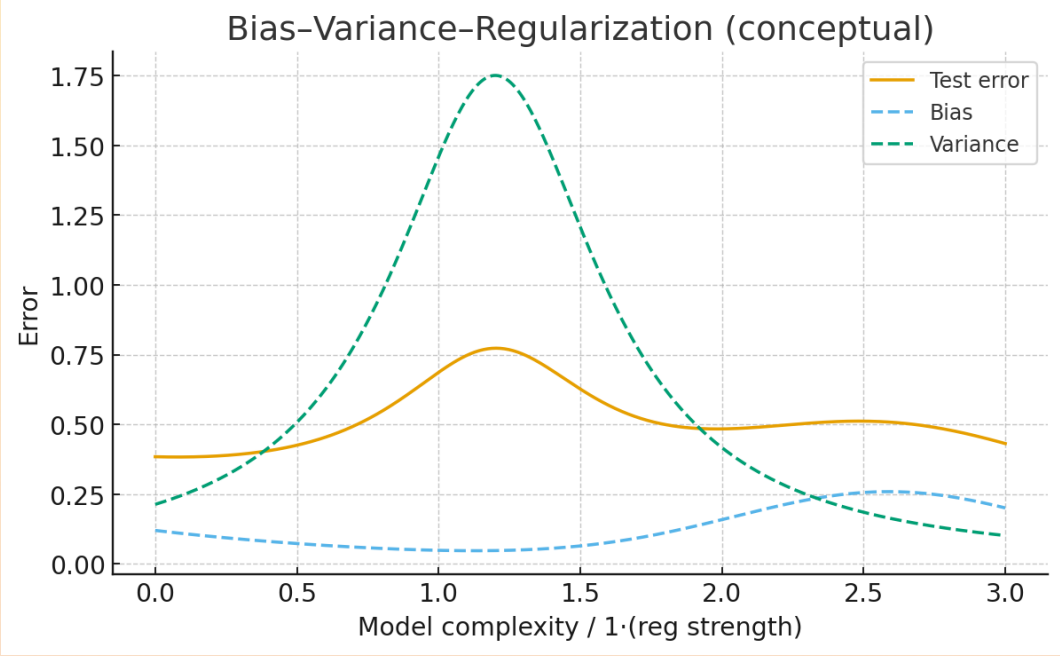
- A model can be calibrated yet yield inefficient intervals if residuals are heavy-tailed.
- Conversely, conformal prediction coverage can hold while probabilities are miscalibrated.





Why regularization matters for uncertainty

- Regularization shapes the hypothesis space and the variance of predictions
  - The amount a model overfits or underfits directly affects how confident it should be
- Overly flexible models can become overconfident.
- Regularization helps the model reduce its confidence when data don't support a strong decision
- Too much regularization increases bias and shrinks too much the variance
- Nevertheless, post-hoc calibration is still recommended after training.



Explicit regularisers

- Weight decay penalizes large parameters and often improves calibration.
- Dropout averages sub-networks and can reduce variance but may harm calibration if misused.
- Label smoothing (see figure) prevents extreme probabilities and can improve robustness to shift.
- Loss function penalties: add a penalty term to the cross section (e.g. penalizing models that do not satisfy the required properties. Example to read: Karpatne, 2017
- In all cases, there is a hyperparameter that we can tune using validation objectives

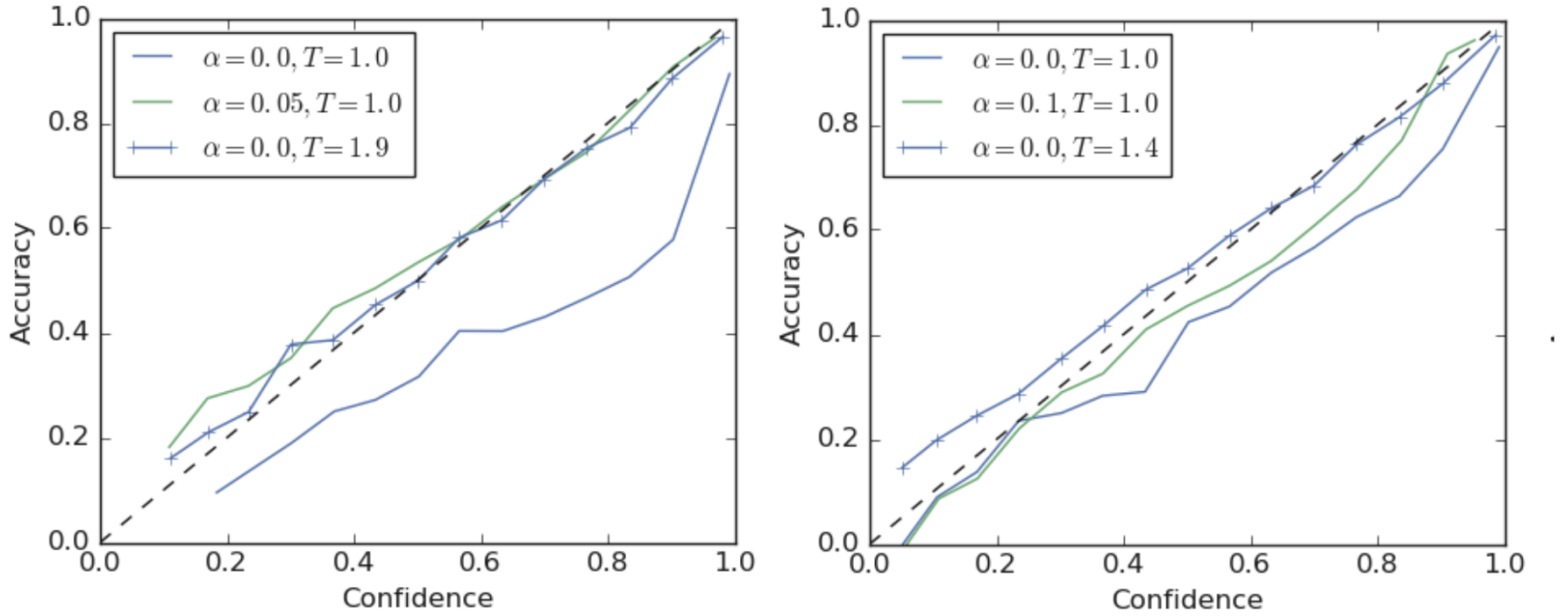


Figure 2: Reliability diagram of ResNet-56/CIFAR-100 (left) and Inception-v4/ImageNet (right).

Srivastava et al., 2014, figure from Müller et al., 2019

Implicit regularisers

- Data augmentation: replicate each data point by e.g. generating its transformed version under the symmetry in exam. Example to read: Chen, 2020
  - Encodes invariances and typically helps accuracy and calibration
- Architectural design: tweak the architecture of the model to make it satisfy some symmetries. For instance, CNNs for rotations and traslations. Works with approximate and exact symmetries.
  - Embed inductive biases such as equivariance or attention locality
- Optimization schedulers and early stopping act as implicit regularizers
- In overparameterized models, often these provide stronger regularization than explicit penalty terms
- In regulated domains, penalty terms often used to introduce the effect of regularization

Bias: when it's inductive, it's a blessing

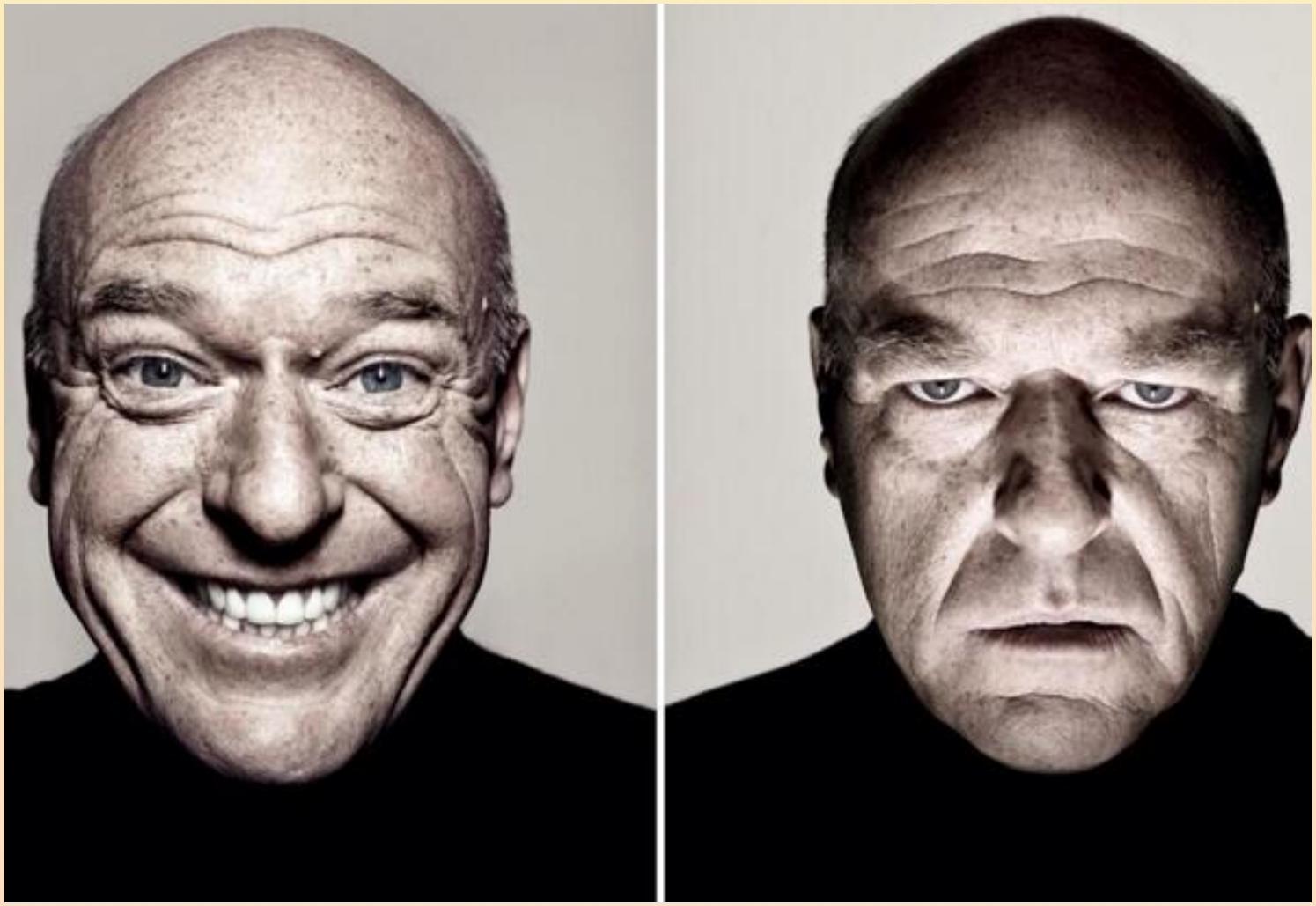
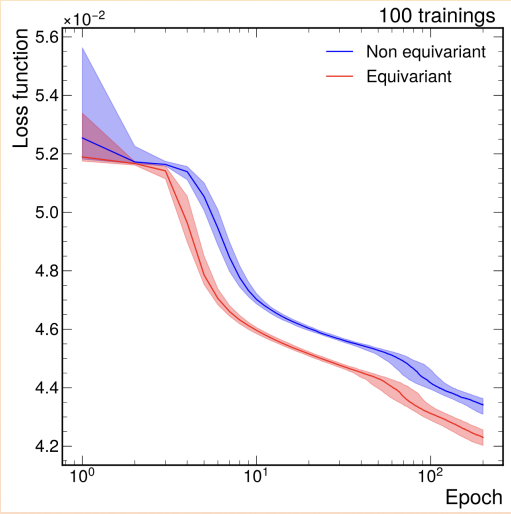


Image by Dean Norris himself

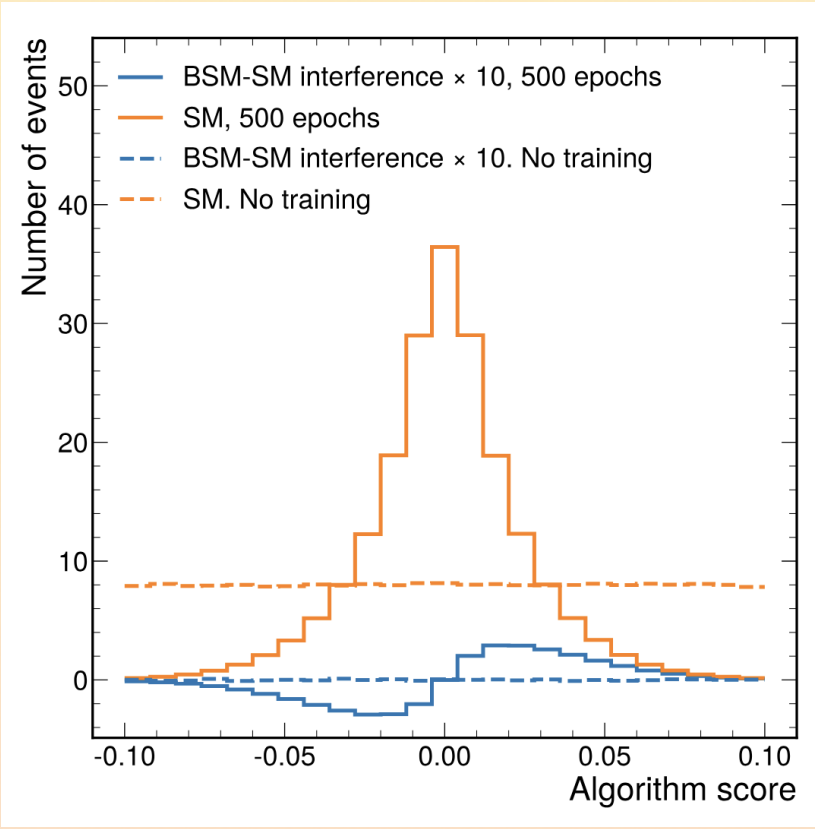
Inductive bias for CP S. Sanchez, M. Kolosova, C. Ramón, P.V. Phys. Rev. D 110, 096023

- Most general equivariant function under CP:
  $f(event) = g(event) - g(CP(event))$
- Parameterize g using a neural network, train f to minimize a loss function
  - After training score is CP-odd (even) for CP-odd (even) processes
  - Any SM-like mismodelling/background will be symmetric by construction!
  - Constructive/destructive interference pattern for positive/negative values
- Injected information results in **40-300% less iterations** needed to achieve the same loss value!!!



From , Phys. Rev. D 110, 096023 (2024)

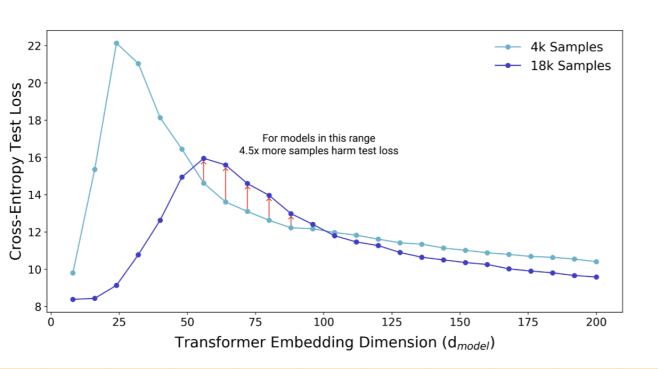
- Equivariance respected at all stages of training
  - The observable is robust even before training convergence



Double descent, bias, and coverage degradation

- As capacity grows, models can interpolate and still generalize via implicit bias.
  - Sometimes, more data $\longrightarrow$ worse test loss
- Measured test error may descend again beyond the interpolation threshold.
- Calibration and coverage can degrade if capacity lacks guiding priors.

We evaluate uncertainty metrics across the capacity sweep.



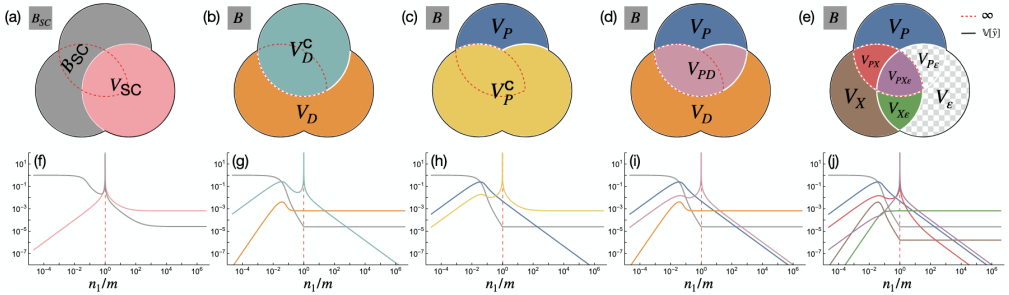


Figure 1: (a-e) The different bias-variance decompositions described in Sec. 4. (f-j) Corresponding theoretical predictions of Thm. 1 for $\gamma = 0$, $\phi = 1/16$ and $\sigma = \tanh$ with SNR $= 100$ as the model capacity varies across the interpolation threshold (dashed red). (a,f) The semi-classical decomposition of [21, 23] has a nonmonotonic and divergent bias term, conflicting with standard definitions of the bias. (b,g) The decomposition of [25] utilizing the law of total variance interprets the diverging term V_D^{C} as "variance due to optimization". (c,h) An alternative application of the law of total variance suggests the opposite, *i.e.* the diverging term V_P^{C} comes from "variance due to sampling". (d,i) A bivariate symmetric decomposition of the variance resolves this ambiguity and shows that the diverging term is actually V_{PD}, *i.e.* "the variance explained by the parameters and data together beyond what they explain individually." (e,j) A trivariate symmetric decomposition reveals that the divergence comes from two terms, V_{PX} and $V_{PX\varepsilon}$ (outlined in dashed red), and shows that label noise exacerbates but does not cause double descent. Since $V_\varepsilon = V_{P\varepsilon} = 0$, they are not shown in (j).

Figures from Nakkiran et al., 2020 and Adlam, Pennington, 2020

OOD Detection

- Inputs far from training support undermine assumptions and invalidate guarantees
- OOD gates protect the predictor by abstaining or escalating on unfamiliar inputs
  - They trigger data collection loops that shrink epistemic uncertainty over time.
  - They must be tuned on proxy validation sets representative of deployment.

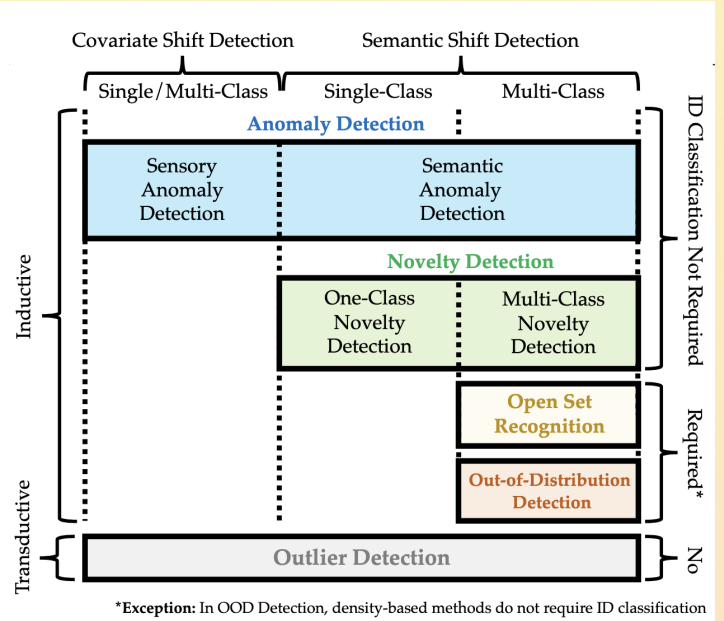
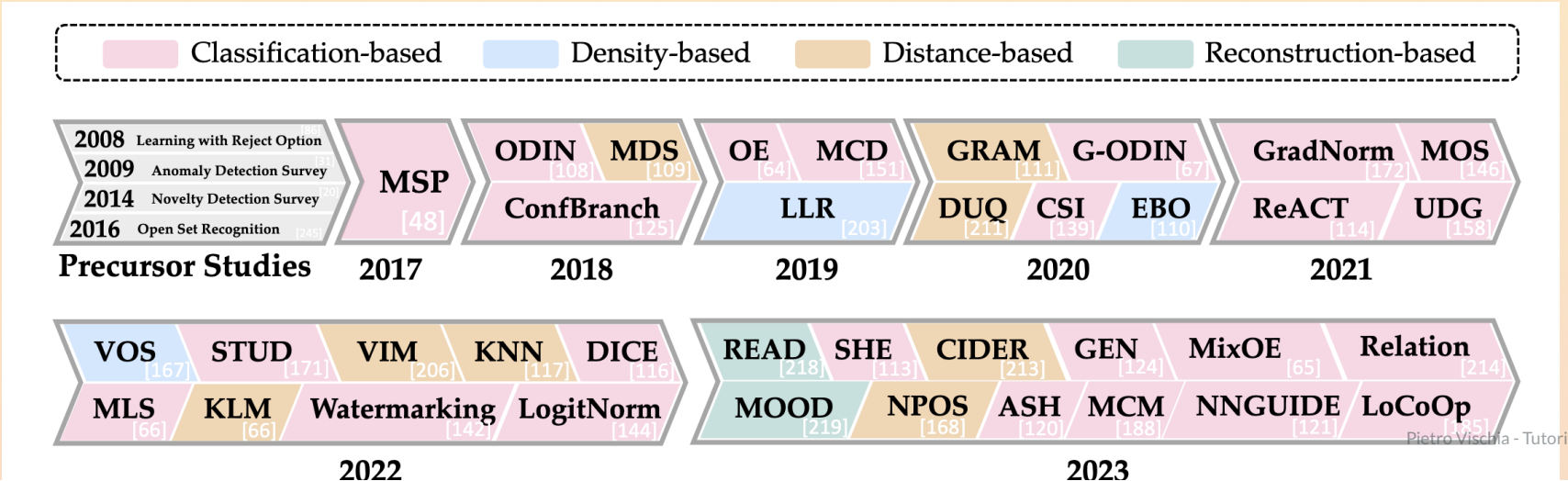


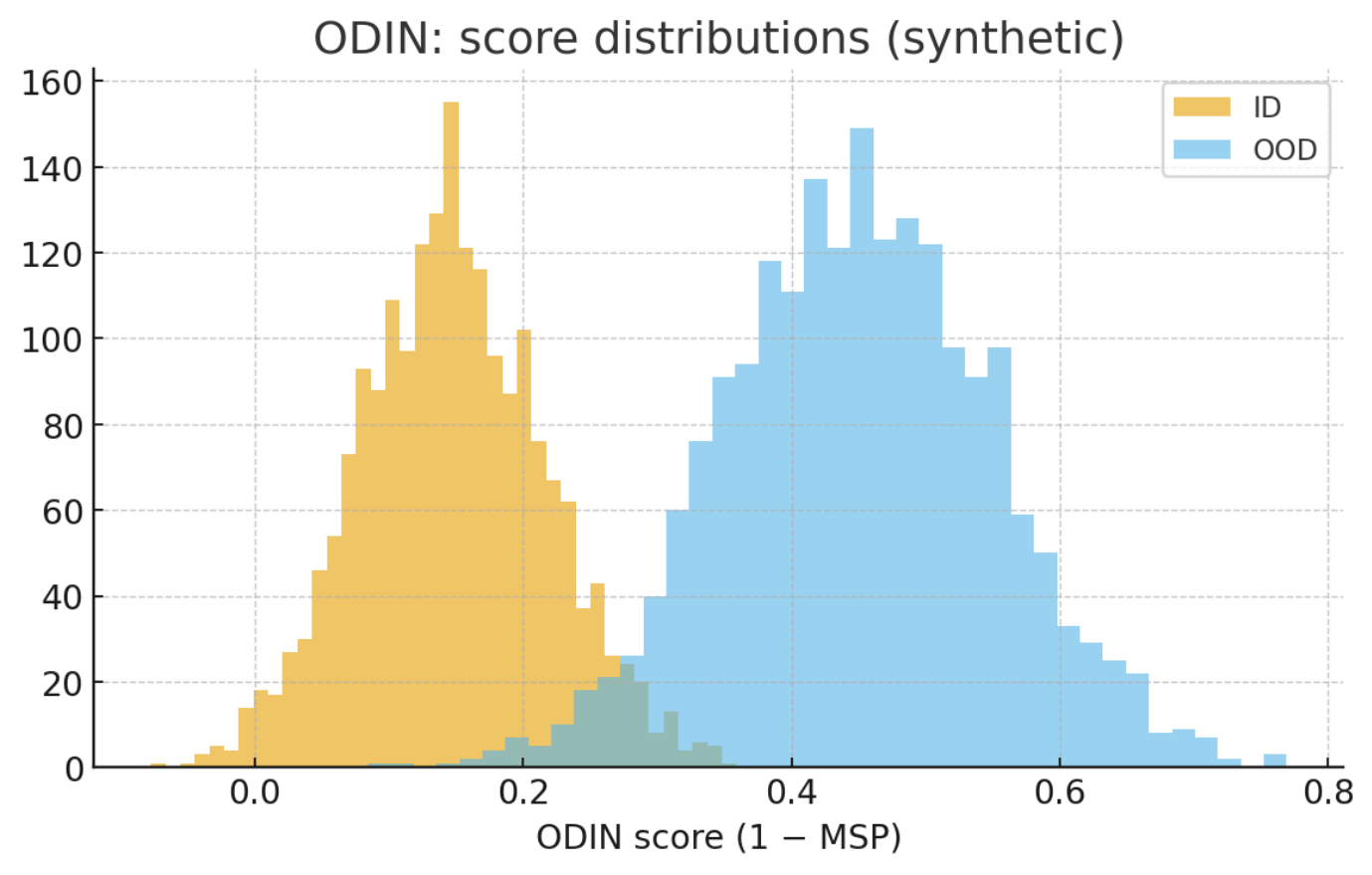
Fig. 1 Taxonomy of generalized OOD detection framework,



Diagrams from Yang et al., 2021

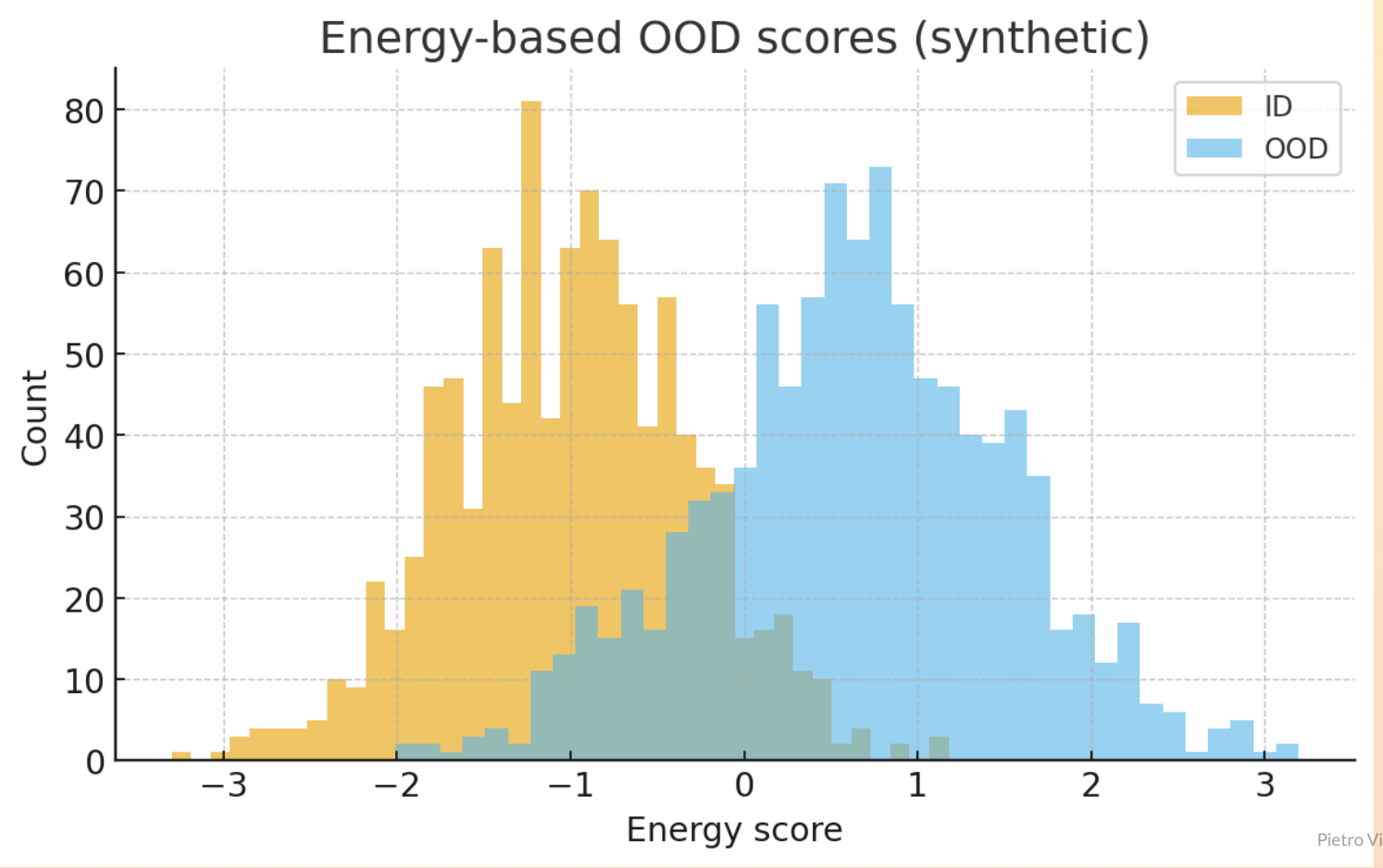
ODIN: detect OOD on pretrained network Liang et al., 2018

- Apply temperature scaling to logits and apply tiny input perturbations
- This amplifies separation between in- and out-of-distribution scores.
- It requires no retraining and works with any trained classifier.



Energy-based OOD detection Liu et al., 2020

- The energy score equals negative log-sum-exp of logits and correlates with confidence
- Energy-based thresholds often outperform max-softmax probabilities for OOD
- Thresholds must be validated on a proxy set and rechecked after recalibration.



OOD and conformal prediction

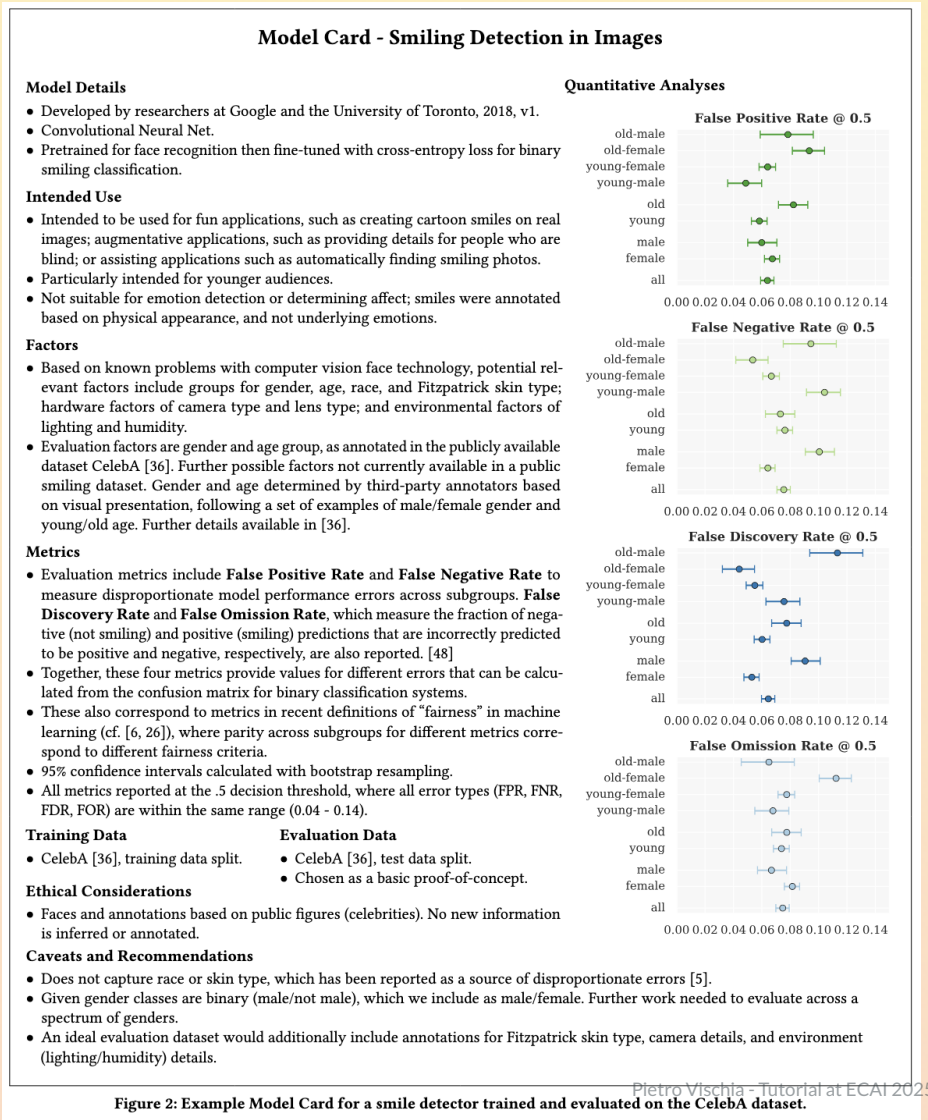
- Conformal prediction assumes exchangeability and does not guarantee coverage on extreme OOD inputs.
- An OOD gate helps keeping conformal prediction in its regime of validity
  - Filtering improbable inputs.
- Conformal prediciton intervals widen near distribution edges, signaling caution to users.
- Clear advantages in coupling OOD gates and conformal prediction

Good OOD practices

- Use realistic corruptions or near-OOD samples for validation, not just far-OOD datasets.
- Re-evaluate thresholds after every model update and recalibration.
- Log the distribution of scores to catch drift early.
- Document fallback behaviors to prevent silent failures

Auditing and governance

- Record datasets, splits, thresholds, and calibration procedures used in the release notes
  - Back to the model cards we saw at the beginning
- Track drift detection and incident response.
- Design dashboards that expose uncertainty and OOD statistics to operators.



Model Card - Smiling Detection in Images

Model Details
- Developed by researchers at Google and the University of Toronto, 2018, v1.
- Convolutional Neural Net.
- Pretrained for face recognition then fine-tuned with cross-entropy loss for binary smiling classification.

Intended Use
- Intended to be used for fun applications, such as creating cartoon smiles on real images; augmentative applications, such as providing details for people who are blind; or assisting applications such as automatically finding smiling photos.
- Particularly intended for younger audiences.
- Not suitable for emotion detection or determining affect; smiles were annotated based on physical appearance, and not underlying emotions.

Factors
- Based on known problems with computer vision face technology, potential relevant factors include groups for gender, age, race, and Fitzpatrick skin type; hardware factors of camera type and lens type; and environmental factors of lighting and humidity.
- Evaluation factors are gender and age group, as annotated in the publicly available dataset CelebA [36]. Further possible factors not currently available in a public smiling dataset. Gender and age determined by third-party annotators based on visual presentation, following a set of examples of male/female gender and young/old age. Further details available in [36].

Metrics
- Evaluation metrics include **False Positive Rate** and **False Negative Rate** to measure disproportionate model performance errors across subgroups. **False Discovery Rate** and **False Omission Rate**, which measure the fraction of negative (not smiling) and positive (smiling) predictions that are incorrectly predicted to be positive and negative, respectively, are also reported. [48]
- Together, these four metrics provide values for different errors that can be calculated from the confusion matrix for binary classification systems.
- These also correspond to metrics in recent definitions of "fairness" in machine learning (cf. [6, 26]), where parity across subgroups for different metrics correspond to different fairness criteria.
- 95% confidence intervals calculated with bootstrap resampling.
- All metrics reported at the .5 decision threshold, where all error types (FPR, FNR, FDR, FOR) are within the same range (0.04 - 0.14).

Training Data
- CelebA [36], training data split.

Evaluation Data
- CelebA [36], test data split.
- Chosen as a basic proof-of-concept.

Ethical Considerations
- Faces and annotations based on public figures (celebrities). No new information is inferred or annotated.

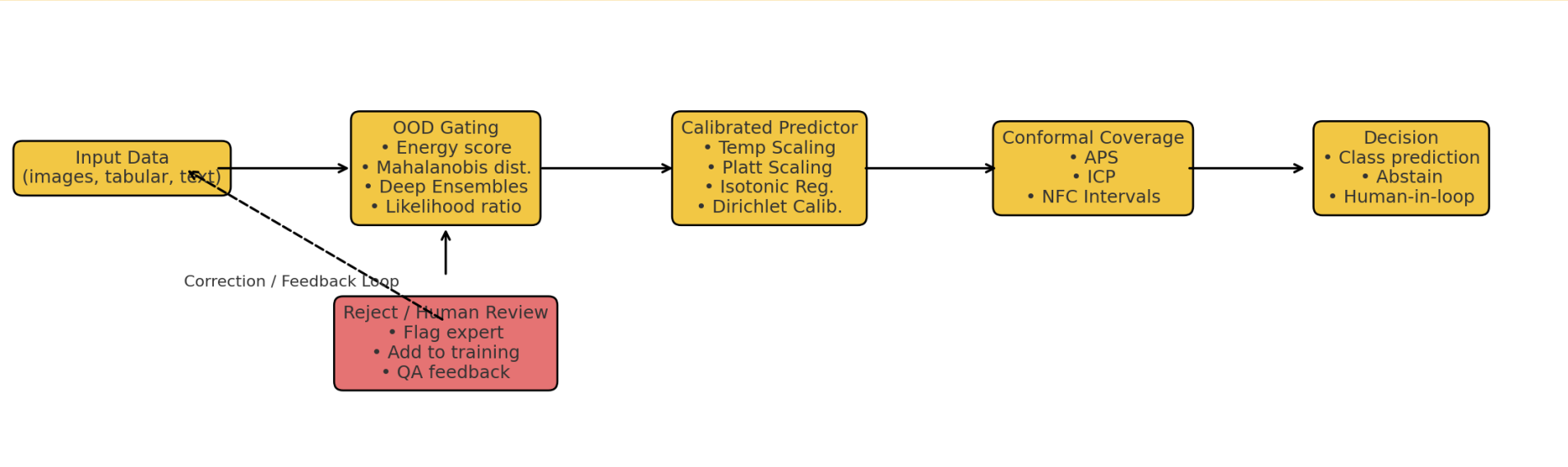
Quantitative Analyses

Caveats and Recommendations
- Does not capture race or skin type, which has been reported as a source of disproportionate errors [5].
- Given gender classes are binary (male/not male), which we include as male/female. Further work needed to evaluate across a spectrum of genders.
- An ideal evaluation dataset would additionally include annotations for Fitzpatrick skin type, camera details, and environment (lighting/humidity) details.

Figure 2: Example Model Card for a smile detector trained and evaluated on the CelebA dataset.

Figure from Mitchell et al., 2019

An integrated pipeline



Further references

- Dedicated workshop (COST Action "COMETA"): https://indico.cern.ch/event/1487660/
- Structured prediction: https://papers.nips.cc/paper_files/paper/2015/hash/52d2752b150f9c35ccb6869cbf074e48-Abstract.html
- Bayesian Learning: Radford Neal, Bayesian Learning for Neural Networks
- Video seminar by Gael Varouquaux

Feedback Welcome

or click here: https://share.google/kJlNe22oTtXzNnv0K

That's all!

Hands on: https://github.com/vischia/ecai2025/