Amortized Conditional Normalized Maximum Likelihood: Reliable Out-of-Distribution Uncertainty Estimation

Aurick Zhou, Sergey Levine
UC Berkeley

Uncertainty Estimation is Especially Important Under Distribution Shift

Machine learning models are often much less accurate with distribution shifted test data

Ideally, model should be able to identify inputs for which it’s predictions should not be trusted

Distribution shift hurts uncertainty estimation as well as accuracy, resulting in highly overconfident mistakes

Conditional Normalized Maximum Likelihood (CNML)

Intuition: want conservative uncertainty estimates for out-of-distribution test inputs where errors are more likely. If we can find models consistent with our training data that predict different labels on the test input, then we should have high uncertainty.

Given an input x, for each possible label y, find model that best fits the label together with the training data.

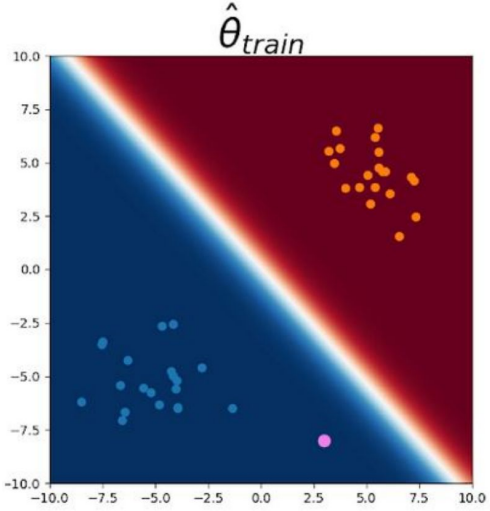
$$\hat{\theta}_y = \text{argmax}_\theta \log p_\theta(y|x) + \log p_\theta(D_{\text{train}})$$

Regret for a predicted distribution p and label y: $R(p, y) = \log p_{\hat{\theta}_y}(y|x) - \log p(y)$

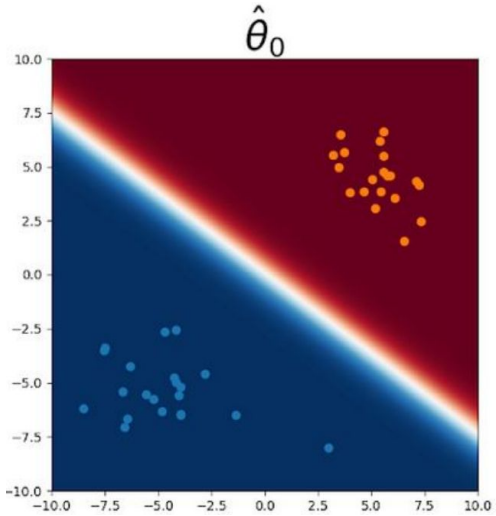
CNML minimizes worst-case regret: $p_{\text{CNML}} = \operatorname{argmin}_p \max_y R(p, y)$

$$p_{\text{CNML}}(y) = \frac{p_{\hat{\theta}_y}(y|x)}{\sum_{y'} p_{\hat{\theta}_{y'}}(y'|x)}$$

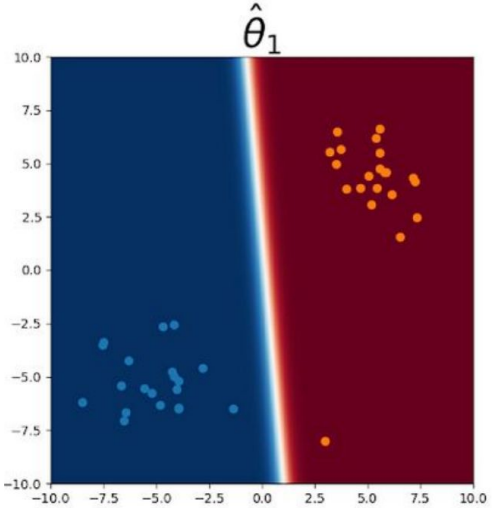
CNML for Logistic Regression



Optimal model prediction for the original training set, test input shown in pink

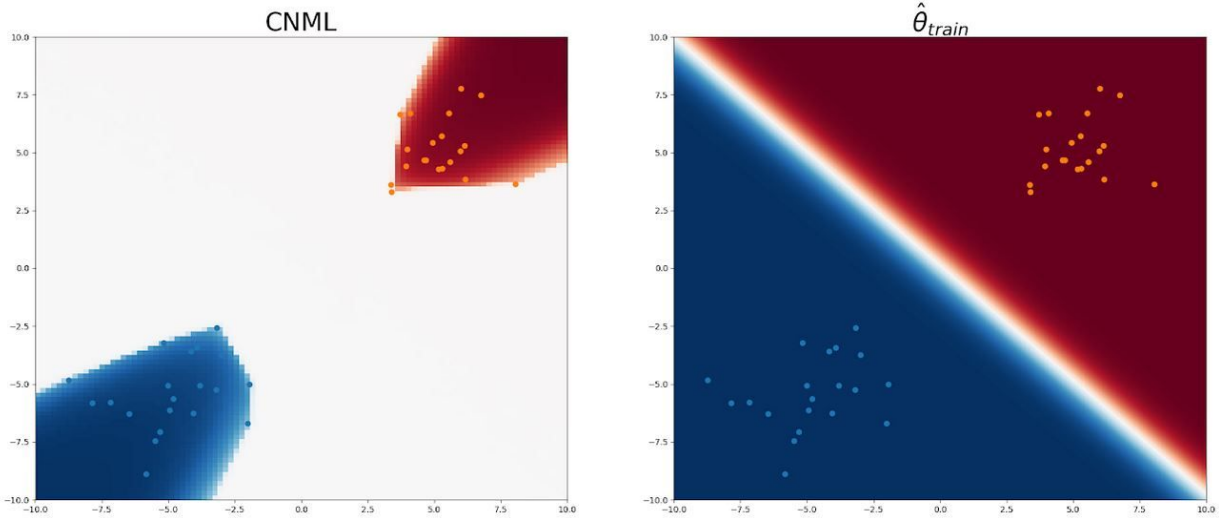


Optimal model predictions after assigning test input to the blue class



Optimal model predictions after assigning test input to the orange class

CNML for Logistic Regression



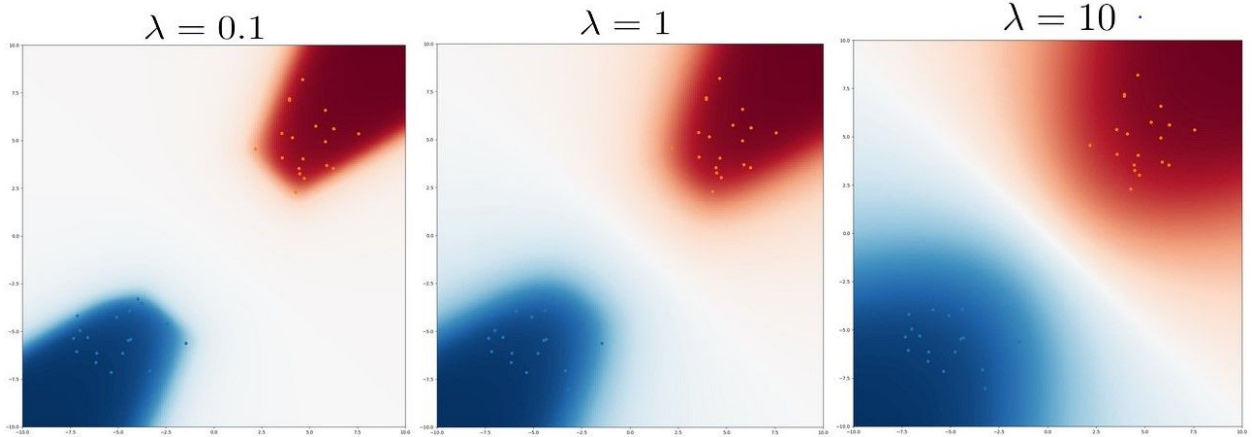
Heatmap of CNML predictions (left) vs training MLE (right): CNML predictions are very uncertain away from the training data, while the MLE extrapolates confidently.

Controlling conservativeness with regularization

CNML with highly expressive model classes might be able to fit too many labels, even close to the training data where we want to make confident predictions.

Solution: regularize! Instead of taking MLE for each label, take a MAP solution

$$\hat{\theta}_y = \underset{\theta}{\operatorname{argmax}} \log p_\theta(y|x) + \log p_\theta(D_{\text{train}}) + \log p(\theta)$$



Increasing weight decay regularization (higher λ) provides more confident extrapolation as we move away from the data.

CNML is infeasible in realistic scenarios

For each input and label, need to solve an expensive optimization problem over the entire training set

$$\hat{\theta}_y = \operatorname*{argmax}_{\theta} \log p_\theta(y|x) + \log p_\theta(D_{\text{train}}) + \log p(\theta)$$

Requires access to training data at test time, can be very slow to optimize

Amortized Conditional Normalized Maximum Likelihood (ACNML)

Idea: replace the training loss with a simple approximate Bayesian posterior density

$$\hat{\theta}_y = \arg\max_{\theta} \log p_\theta(y|x) + \underbrace{\log p_\theta(D_{\text{train}}) + \log p(\theta)}_{\text{equal to } \log p(\theta|D_{\text{train}})}$$

Learn approximate posterior density during training: $\log q(\theta) \approx \log p(\theta|D_{\text{train}})$

Solve simpler problem at test time: $\hat{\theta}_y = \operatorname{argmax}_{\theta} \log p_\theta(y|x) + \log q(\theta)$

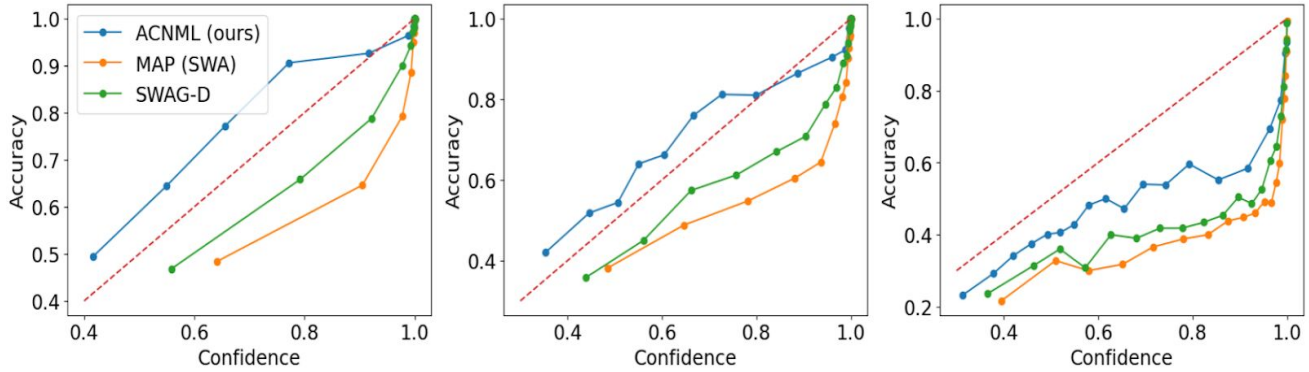
Algorithm 1 Amortized CNML (ACNML)

Input: Model class Θ, Training Data $(x_{1:n-1}, y_{1:n-1})$, Test Point: x_n, Classes (1, . . . , k)
Output: Predictive distribution $p(y|x_n)$
Training: Run approximate inference algorithm on training data $(x_{1:n-1}, y_{1:n-1})$ to get posterior density $q(\theta)$
for all possible labels $i \in (1, \ldots, k)$ **do**
  Compute $\hat{\theta}_i = \text{argmax}_\theta \log p_\theta(i|x_n) + \log q(\theta)$
end for
Return $p(y|x_n) = \frac{p_{\hat{\theta}_y}(y|x_n)}{\sum_{i=1}^{k} p_{\hat{\theta}_i}(i|x_n)}$

ACNML provides conservative uncertainty estimates

Evaluate on corrupted image datasets to simulate distribution shift

Want predictions to be calibrated even with shift: confidence = accuracy

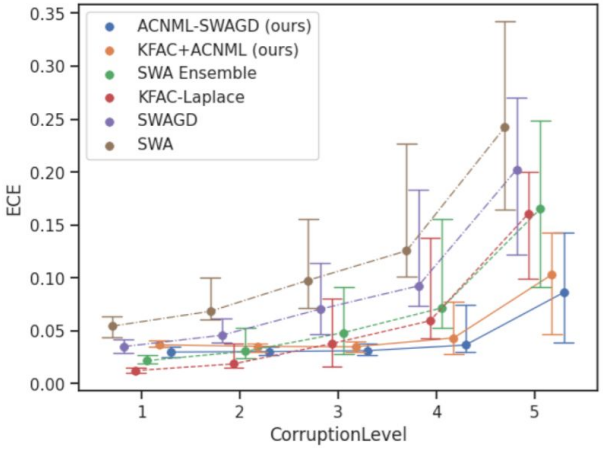


(a) CIFAR10 Test (b) CIFAR10-C Corruption Level 3 (c) CIFAR10-C Corruption Level 5

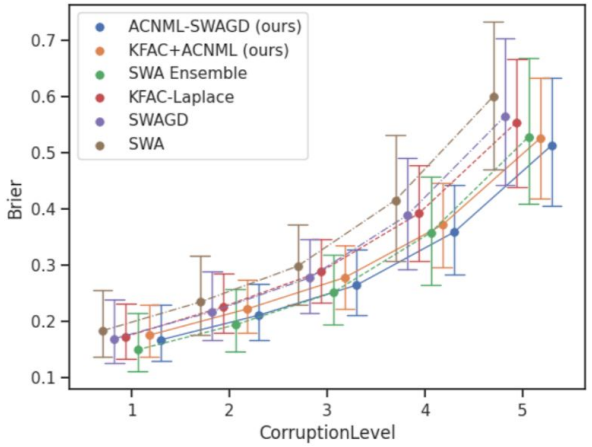
Reliability diagrams visualizing model confidences vs accuracy show ACNML gives less confident predictions overall, resulting in much improved calibration on more severe distribution shifts.

Empirical Results

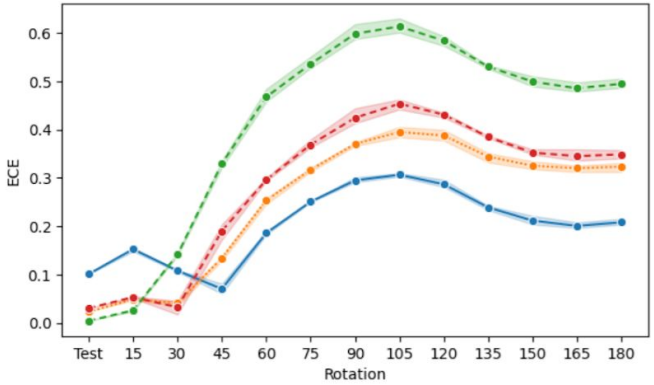
ACNML outperforms Bayesian methods and deep ensembles on severe distribution shifts in terms of expected calibration error as well in Brier score.



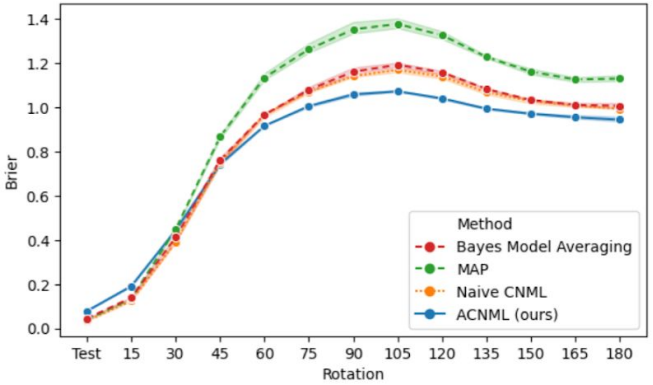
(a) CIFAR10C VGG16 ECEs (lower is better)



(b) CIFAR10C VGG16 Brier Scores (lower is better)



(a) Rotated MNIST ECEs (lower is better)



(b) Rotated MNIST Brier Scores (lower is better)