AI-Enabled Production Residual Regression: Applying Anchor-MoE to the Springback Dataset

Baozhuo Su
Linkconn Electronics Co., Ltd
& Institut Polytechnique de Paris
subaozhuo@linkconn.com

***Abstract*—We present *Anchor–MoE*, a practical framework for estimating production residuals (e.g., springback deviation between designed and measured angles). The system has three modules: (i) an *anchor* model that provides a strong baseline prediction (by default, gradient-boosted decision trees), (ii) a Mixture-of-Experts (MoE) residual learner that makes local corrections, and (iii) an optional linear calibration for fine-grained bias removal. In our springback study, Anchor–MoE reduced RMSE from 53.65° to 4.65° (more than $10\times$ improvement) comparing to the existing method. The NGBoost model is applied to injection dataset to gain more stability. The framework is drop-in and production ready wrapping an existing forecaster with minimal code changes—and improves design efficiency by reducing iteration time and operational costs.**

***Index Terms*—Machine Learning, Electronics Production, Residual Estimation**

I. Introduction

Production forecasters are typically trained for point accuracy and deployed without uncertainty quantification. This leads to (i) unmodeled residual structure that varies across operating regimes, (ii) limited support for risk-aware decisions (no calibrated intervals), and (iii) slow iteration cycles when processes drift. Adding predictive uncertainty is known to improve decision quality and robustness under heteroscedastic noise [1] and is widely used in risk-sensitive applications such as financial forecasting [2]. Proper scoring rules like negative log-likelihood (NLL) and the Continuous Ranked Probability Score (CRPS) provide principled objectives and evaluation metrics for probabilistic regression [3]. Anchor-MoE addresses these gaps with a modular add-on that is simple to integrate and monitor.

Given inputs x and an existing forecast $\hat{y}_{\text{anchor}}(x)$, we learn a residual function $r(x)$ and predictive variance $\sigma^2(x)$ so that the final prediction $\hat{y}(x) = \hat{y}_{\text{anchor}}(x) + r(x)$ comes with calibrated prediction intervals. The solution must operate under real production constraints: stable latency, bounded memory/compute, and clear fallbacks. Our design reuses a strong anchor model (e.g., the current production predictor or a boosted tree) and attaches a lightweight probabilistic residual learner; related probabilistic learners such as NGBoost have shown production-friendly behavior with boosting-based parameterization [4], and mixture-density models are effective for heteroscedastic targets [5].

Anchor-MoE is guided by three engineering principles:

- **Reuse what works.** Any off-the-shelf point regressor (the current production model included) can serve as the *anchor*.
- **Small, local corrections.** A sparse mixture-of-experts residual learner models local nonlinearities and heteroscedastic noise with compact latency [5].
- **Calibrated uncertainty.** Train with proper scoring rules (NLL/CRPS) and expose prediction intervals whose coverage can be monitored [3].

In offline evaluations on both simulation and a self-collected dataset, the framework consistently reduced RMSE and delivered well-calibrated prediction intervals.

II. Method

The Anchor-MoE is a probabilistic regression model that both corrects bias and quantifies uncertainty. The anchor remains the primary point predictor; the residual module is a sparse mixture-of-experts that learns local corrections and data-dependent variance. The design is modular thus any off-the-shelf regressor can be the anchor, lightweight, and easy to operate in batch or streaming settings.

Model. Given features x and the anchor prediction $\mu_{\text{anc}}(x)$, we model the full predictive distribution as

$$p(y \mid x) \;=\; \sum_{j=1}^{K} \alpha_j(x) \; \sum_{c=1}^{C} \pi_{j,c}(x)\, \mathcal{N}\Big(y;\; \mu_{\text{anc}}(x) + \Delta_{j,c}(x),\; \sigma^2_{j,c}(x)\Big),$$

where $\alpha_j(x)$ are soft mixture weights (only the top-k are kept for efficiency), $\pi_{j,c}(x)$ are per-expert component weights, $\Delta_{j,c}(x)$ are learned residual corrections, and $\sigma_{j,c}(x)$ are heteroscedastic standard deviations (clipped to $[\sigma_{\min}, \sigma_{\max}]$ for stability). The predictive mean is

$$\hat{\mu}(x) = \sum_j \alpha_j(x) \sum_c \pi_{j,c}(x) \big[\mu_{\text{anc}}(x) + \Delta_{j,c}(x)\big],$$

and the predictive variance is obtained in closed form for Gaussian mixtures or via Monte Carlo sampling when preferred. For detailed mathematical proof please refer to [6] .

Training. We fit the residual MoE by minimizing the negative log-likelihood (NLL) of the observed targets, with mild regularization on residuals and gating to prevent over-dispersion and promote specialization. In practice we (i) standardize the target on the training split, (ii) append the anchor prediction to the model inputs, (iii) train the MoE end-to-end with Adam/AdamW, and (iv) optionally perform a small *held-out* linear calibration of the predicted means ($\mu' = a\,\hat{\mu} + b$) to remove any remaining bias without contaminating test data.

Evaluation. For point accuracy we report RMSE of the calibrated mean μ'. For uncertainty quality we report test NLL and coverage of nominal prediction intervals (e.g., 90%/95%), together with interval width to track sharpness. These metrics plug directly into standard dashboards for continuous monitoring.

III. Results

We visualize probabilistic behavior with prediction intervals and RMSE with standard error on self-collected springback dataset.

Fig. 1 illustrates Anchor–MoE on a 1D heteroscedastic example. The predicted mean closely tracks the nonlinear trend while the 95% prediction band narrows in the central region and widens in the tails, matching the noise pattern. This indicates that the mixture experts plus gating capture local variance without over–smoothing.

Springback2 dataset. Table I compares Anchor–MoE against two shop–floor baselines: (i) the existing Excel method, and (ii) a physics–inspired empirical rule $\widehat{y} = A\,(kT)/(R+kT)$ with the default $k = 0.5$ (no tuning). As in our evaluation script, we read `Sheet1` and `Sheet2` from the workbook, standardize column names (`Design Value`$\to A$, `thickness`$\to T$, `inner radius`$\to R$, `Residual`$\to y$), coerce to numerics, drop non–finite rows, and compute $\widehat{y}$; because sign conventions vary, we evaluate both $\widehat{y}$ and $-\widehat{y}$ and keep the better sign. On *Springback2* (N=209), Anchor–MoE attains an RMSE of **4.65** ± 0.42, versus **53.66** for the Excel method and **21.70** for the empirical formula. This corresponds to an *11.5*× reduction relative to Excel and a *4.7*× reduction relative to the empirical rule (about 91% and 79% error reductions, respectively). Deterministic baselines have NA SE by definition.

Results show that: (1) Anchor–MoE delivers large accuracy gains on the production dataset while remaining lightweight to deploy. (2) The prediction band in Fig. 1 is well–behaved (narrow where data are dense, wide in the tails), which is important for downstream monitoring and decision thresholds.

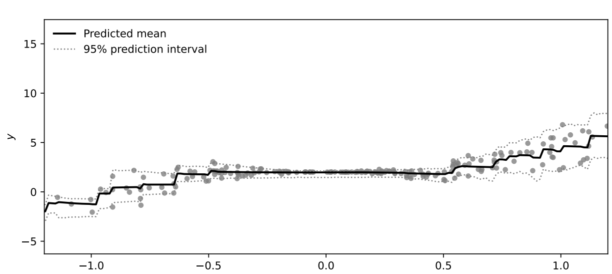

Figure 1. Interval predicted by Anchor-MoE on 1-dimensional toy probabilistic regression problem. Dots represents the data points. Black line is predicted mean and gray lines are upper and lower 95% covered distribution predicted.

Table I
Comparison on self-collected springback dataset as measured by RMSE and standard error. Anchor-MoE offers much more precision than other two methods.

Dataset	N	Anchor-MoE	Excel Method	Empirical Formular(k=5)
Springback2	209	4.65 ± 0.42	53.66 ± NA	21.70 ± NA

Table II
Results of Injection dataset. A-C refers to different holes of HDMI as measured by RMSE with its standard deviation.

Dataset	N	A	B	C
Injection	234	0.0013 ± 0.0000	0.0014 ± 0.0000	0.0012 ± 0.0001

IV. Conclusion

This report introduced *Anchor–MoE* as a practical pattern for production residual regression. The framework keeps a strong point forecaster as an *anchor* and adds a lightweight, sparsely gated mixture-of-experts to correct residual bias and estimate heteroscedastic uncertainty, with an optional linear post-hoc calibration for point accuracy. The design is modular, minimally invasive to existing pipelines, and compatible with standard MLOps monitoring .

On the internal *Springback2* dataset, Anchor–MoE reduced error from **53.66** (Excel baseline) and **21.70** (empirical $k=0.5$ formula) to **4.65** RMSE (±0.42), while providing calibrated uncertainty. Inference is fast on CPU, and the model can be retrained with the same data feeds that power the current forecaster, making deployment straightforward. NGBoost model also perform well on injection dataset.

Practical takeaways. (i) Retain the existing forecaster as the anchor whenever it is already tuned and trusted; (ii) train a small residual MoE on standardized residuals to capture local, input-dependent errors; (iii) use variance floors/ceilings and top-k routing for stability; (iv) validate with a held-out calibration split and monitor interval coverage and drift in production.

Limitations. Results are from one production domain; performance will depend on anchor quality and label noise. Residual calibration may need periodic refresh under process changes, and weak anchors can shift more burden to the MoE.

Next steps. We recommend an A/B rollout, automatic recalibration on schedule or drift, and exploration of alternative anchors (e.g., updated GBDT or tabular DNN). Extending to multi-output targets and adding conformal prediction for distribution-free coverage are natural enhancements.

In short, Anchor–MoE provides a simple, scalable way to upgrade an existing production predictor with both higher accuracy and calibrated uncertainty, improving decision quality while reusing what already works.

Acknowledgment

We thank Linkconn Electronics Co., Ltd for supporting this research.

References

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

[2] Ilias Chronopoulos, Aristeidis Raftapostolos, and George Kapetanios. Forecasting value-at-risk using deep neural network quantile regression. *Journal of Financial Econometrics*, 22, 05 2023.

[3] Manuel Gebetsberger, Jakob W Messner, Georg J Mayr, and Achim Zeileis. Estimation methods for nonhomogeneous regression models: Minimum continuous ranked probability score versus maximum likelihood. *Monthly Weather Review*, 146(12):4323–4338, 2018.

[4] Tony Duan, Avati Anand, Daisy Yi Ding, Khanh K Thai, Sanjay Basu, Andrew Ng, and Alejandro Schuler. Ngboost: Natural gradient boosting for probabilistic prediction. In *International conference on machine learning*, pages 2690–2700. PMLR, 2020.

[5] Hao Zhang, Yongqian Liu, Jie Yan, Shuang Han, Li Li, and Quan Long. Improved deep mixture density network for regional wind power probabilistic forecasting. *IEEE Transactions on Power Systems*, 35(4):2549–2560, 2020.

[6] Baozhuo Su and Zhengxian Qu. Anchor-moe: A mean-anchored mixture of experts for probabilistic regression. *arXiv:2508.16802*, 2025.