SAUC: Sparsity-Aware Uncertainty Calibration for Spatiotemporal Prediction with Graph Neural Networks

Dingyi Zhuang
Massachusetts Institute of Technology
Cambridge, USA
dingyi@mit.edu

Yuheng Bu
University of Florida
Gainesville, Florida, USA
buyuheng@ufl.edu

Guang Wang
Florida State University
Tallahassee, Florida, USA
guang@cs.fsu.edu

Shenhao Wang
University of Florida
Gainesville, Florida, USA
shenhaowang@ufl.edu

Jinhua Zhao
Massachusetts Institute of Technology
Cambridge, USA
jinhua@mit.edu

Abstract

Quantifying uncertainty is crucial for robust and reliable predictions. However, existing spatiotemporal deep learning mostly focuses on deterministic prediction, overlooking the inherent uncertainty in such prediction. Particularly, highly-granular spatiotemporal datasets are often sparse, posing extra challenges in prediction and uncertainty quantification. To address these issues, this paper introduces a novel post-hoc Sparsity-aware Uncertainty Calibration (SAUC) framework, which calibrates uncertainty in both zero and non-zero values. To develop SAUC, we firstly modify the state-of-the-art deterministic spatiotemporal Graph Neural Networks (ST-GNNs) to probabilistic ones in the pre-calibration phase. Then we calibrate the probabilistic ST-GNNs for zero and non-zero values using quantile approaches. Through extensive experiments, we demonstrate that SAUC can effectively fit the variance of sparse data and generalize across two real-world spatiotemporal datasets at various granularities. Specifically, our empirical experiments show a 20% reduction in calibration errors in zero entries on the sparse traffic accident and urban crime prediction. Overall, this work demonstrates the theoretical and empirical values of the SAUC framework, thus bridging a significant gap between uncertainty quantification and spatiotemporal prediction.

CCS Concepts

• **Information systems** → **Data mining**; • **Computing methodologies** → *Machine learning approaches.*

Keywords

Uncertainty Quantification, Calibration Methods, Spatiotemporal Sparse Data, Graph Neural Networks

SIGSPATIAL '24, October 29-November 1, 2024, Atlanta, GA, USA

ACM ISBN 979-8-4007-1107-7/24/10
https://doi.org/10.1145/3678717.3691241

ACM Reference Format:
Dingyi Zhuang, Yuheng Bu, Guang Wang, Shenhao Wang, and Jinhua Zhao. 2024. SAUC: Sparsity-Aware Uncertainty Calibration for Spatiotemporal Prediction with Graph Neural Networks. In *The 32nd ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL '24), October 29-November 1, 2024, Atlanta, GA, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3678717.3691241

1 Introduction

Spatiotemporal Graph Neural Networks (ST-GNNs) have been instrumental in leveraging spatiotemporal data for various applications, from weather forecasting to urban planning [10, 40, 44]. However, most of the models narrowly focus on deterministic predictions, which overlooks the inherent uncertainties associated with the spatiotemporal phenomena. For various safety-related tasks, such as traffic accidents and urban crime prediction, it is crucial to quantify their uncertainty, thus proactively preventing such phenomena from happening. Despite its importance, uncertainty quantification (UQ) remains relatively understudied in the spatiotemporal context, which can lead to erroneous predictions with severe social consequences [1, 8, 30].

Recent research has addressed UQ in spatiotemporal modeling [14, 36, 47], but often fails to account for the sparsity and asymmetrical distribution present in detailed spatiotemporal data. This sparsity, marked by an abundance of zeros, becomes essential when working with high-resolution data. Additionally, the need for granular forecasting in event precaution often leads to more diluted inputs, making sparsity unavoidable. Thus, effectively managing aleatoric (data) uncertainty is crucial in these forecasting tasks.

Prior ST-GNNs with basic UQ functionalities typically focus on assuming data distributions (e.g., Gaussian) and learning these distributions' parameters for spatiotemporal predictions [15, 36, 47]. A key limitation is their inadequate evaluation of UQ reliability using robust calibration metrics. Research by [10, 15, 37, 47] often emphasizes high accuracy and narrow prediction intervals as indicators of effective UQ. This approach overlooks the crucial relationship between data variance and prediction intervals (PI), which is vital for addressing aleatoric uncertainty. As highlighted by [17], most deep learning models are not fully calibrated for high UQ quality, leaving a significant research gap to be filled.

This paper introduces a Sparsity-Aware Uncertainty Calibration (SAUC) framework, calibrating model outputs to align with true

data distributions, which can be adapted to any ST-GNN model. Inspired by the zero-inflated model [2, 9], we partition zero and non-zero predictions and apply separate Quantile Regression (QR) models to ensure the discrepancy of prediction and true values lie within the PIs. This post-hoc approach, unlike uncertainty-aware spatiotemporal prediction models [29], offers flexibility in applying it to existing models without necessitating architectural changes or full model retraining. Consistent with prior studies on regression task calibration, we formulate a modified calibration metric tailored for PI-based calibration within asymmetric distributions. To investigate our model's generalizability, we adapt three prevalent numeric-output ST-GNN models with negative binomial (NB) distributions to accommodate sparse settings without compromising prediction accuracy. Tested on two real-world spatiotemporal datasets with different levels of sparsity, our SAUC demonstrates promising calibration results, thus assisting in reliable decision-making and risk assessment. The main contributions of this paper are threefold:

- We have developed a novel post-hoc uncertainty calibration framework tailored for predictions on sparse spatiotemporal data. This approach is designed to be compatible with current GNN models and is suitable for any probabilistic outputs.
- We focus on calibrating asymmetric distributions, like the NB distributions, using quantile regression. This approach deviates from conventional mean-variance UQ methods. Additionally, we have developed new calibration metrics based on prediction interval width.
- We conduct experiments on real-world datasets with different temporal resolutions to demonstrate promising calibration results of our SAUC, especially for zero values. We observe a roughly **20% reduction** in calibration errors relative to leading baseline models, underscoring its efficacy for safety-critical decision-making.

2 Related Work

2.1 Spatiotemporal Prediction

Spatiotemporal prediction through deep learning offers a powerful tool for various applications. Among them, GNNs have become prevalent in recent years [40–42, 44]. Most existing models focus on predicting the deterministic values without considering the associated uncertainty, which leaves a noticeable research gap concerning UQ. Such a limitation impacts the reliability of predictions, as the unaccounted uncertainty can lead to deviations from the true data variance.

Bayesian and the frequentist approaches are the two main existing methods for spatiotemporal UQ [36, 46]. While the Bayesian approach offers profound insights through techniques like Laplace approximation [38], Markov Chain Monte Carlo [27], and Variational Inference [16], it suffers from computational complexities and approximation inaccuracies. In contrast, the frequentist approach exploits Mean-Variance Estimation for computational efficiency and model flexibility, accommodating both parametric and non-parametric methods [36]. Recently, many frequentist approaches are proposed by modifying the last layers of the ST-GNN model outputs [10, 15, 29]. However, the application of these methods to sparse data is challenged by data dependencies and distribution constraints [11, 39]. For example, sparse data frequently deviate from models' mean-variance assumptions, favoring other right-skewed distributions like NB over Gaussian, and traditional metrics like mean squared errors can be skewed by sparse datasets' non-zero values.

The use of zero-inflated distributions with spatiotemporal neural networks has been introduced to handle zero instances and non-normal distribution in sparse data [47]. However, this approach has limitations in extreme scenarios and non-time-series contexts, impacting predictive accuracy and system robustness [28]. Therefore, the urgent need to quantify and calibrate uncertainty in the sparse component of spatiotemporal data is evident, yet remains largely unexplored.

2.2 Uncertainty Calibration

Although relatively understudied in the spatiotemporal context, UQ has been examined by many machine learning studies. Among all the UQ approaches, calibration is a post-hoc method that aims to match the predictive probabilities of a model with the true probability of outcomes, developing robust mechanisms for uncertainty validation [26].

Many techniques have been developed to calibrate pre-trained classifiers, mitigating the absence of inherent calibration in many uncertainty estimators [12, 19, 32]. Calibration for regression tasks, however, has received less attention [17, 34]. Post-hoc calibration methods for classifications, such as temperature scaling, Platt scaling, and isotonic regression, have successfully been adapted for regression [18]. Recent research [8] further demonstrated quantile regression-based calibration methods and their extensions. These post-hoc methods offer greater flexibility and wider applicability to various probabilistic outputs compared to in-training methods, without necessitating model retraining [14, 29].

In spatiotemporal prediction contexts, where accurate model outputs are important for applications like traffic accident forecasting and crime prediction [44], the significance of uncertainty calibration is often overlooked. Since spatial and temporal granularity of the data could vary, there could exist different levels of sparsity in the data [47]. It is essential to discern confidence for zero and non-zero segments due to their varied implications. In response, we propose the SAUC framework to manage asymmetric distributions and ensure variances of true targets align with estimated PIs [34].

3 Problem Formulation

3.1 ST-GNNs for Prediction

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$ where $\mathcal{V}$ represents the set of nodes (locations), $\mathcal{E}$ the edge set, and $\mathbf{A} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ the adjacency matrix describing the relationship between nodes. We denote the spatiotemporal dataset $\mathcal{X} \in \mathbb{R}^{|\mathcal{V}|\times t}$ where t is the number of time steps. The objective is to predict the target value $\mathcal{Y}_{1:|\mathcal{V}|,t:t+k}$ of future k time steps given all the past data up to time t, which is $\mathcal{X}_{1:|\mathcal{V}|,1:t}$. The full dataset is partitioned by timesteps into training, calibration (i.e., validation), and testing sets. The three data partitions are denoted as $\mathcal{X}_T$, $\mathcal{X}_S$, and $\mathcal{X}_U$, respectively. For example, the training set $\mathcal{X}_{1:|\mathcal{V}|,1:t}$ is denoted as $\mathcal{X}_T$. The associated target values are

represented as $\mathcal{Y}_T$, $\mathcal{Y}_S$, and $\mathcal{Y}_U$, with subscripts indicating the corresponding sets.

The objective of the ST-GNN models is to design model f_θ, parametrized by θ, which yields:

$$\hat{\mathcal{Y}}_d = f_\theta(\mathcal{X}_d; \mathcal{G}), \forall\, d \in \{T, S, U\}, \tag{1}$$

where $\hat{\mathcal{Y}}_d$ is the predicted target value, and the subscript represents the corresponding data set. Note that θ is fixed once f_θ is trained on the set T.

With emerging interests in quantifying the data and model uncertainty of ST-GNNs, researchers shift towards designing probabilistic GNNs to predict both mean values and PIs [5, 36]. Previous probabilistic ST-GNN studies often assume $\mathcal{X}$ and $\mathcal{Y}$ are independent and identically distributed random variables, i.e., denoted as X_i and Y_i to describe data distributions. The last layer of the neural networks architectures is modified to estimate the parameters of the assumed distributions[15, 47]. For this study, we use the NB distribution to discuss the sparse and discrete data distribution.

The NB distribution of each predicted data point is characterized by shape parameter μ and dispersion parameter α, and $\hat{\mu}_i \in \hat{M}_d$ denotes the mean value of the predicted distribution. Previous work has applied prediction layers with the corresponding likelihood loss after ST-GNN encoding to study the parameter sets of the predicted data points [15, 47]. The set of predicted μ and α corresponding to all elements of $\mathcal{Y}$ are denoted as $\hat{M}$ and $\hat{H}$. Therefore, Equation 1 can be rewritten in the format of NB distribution-based probabilistic ST-GNNs as:

$$(\hat{M}_d, \hat{H}_d) = f_\theta(\mathcal{X}_d; \mathcal{G}), \forall d \in \{T, S, U\}. \tag{2}$$

Subsequently, the objective function is adjusted to incorporate the relevant log-likelihood, as denoted in Equation 12.

It is important to note that the selection of the distribution is not the primary focus of this paper. Our goal is to present a general framework to calibrate probabilistic outputs for sparse spatiotemporal data, as long as the predicted distributions are provided. Implementations for other distributions are included in Section 5.7, but the NB distribution is used for illustration in this study.

3.2 Prediction Interval Calibration

Generally, for the frequentist approach, the spatiotemporal prediction models output the sets of location and shape parameters $\hat{M}_d, \hat{H}_d, \forall d \in \{T, S, U\}$ [17, 22, 33]. We construct the prediction intervals of the i-th data point as $\hat{\mathcal{I}}_i = [\hat{l}_i, \hat{u}_i]$ with $\hat{l}_i$ and $\hat{u}_i$ denoting the 5th- and 95th-percentiles, respectively. Notably, our discussion limits the prediction intervals to the 5% and 95% range. Based on [17], our target is to ensure a prediction model f_θ to be calibrated, i.e.,

$$\frac{\sum_{i=1}^{|U|} \mathbb{I}\{y_i \le \hat{F}_i^{-1}(p)\}}{|U|} \xrightarrow{|U|\to\infty} p,\ \forall p \in \{0.05, 0.95\}, \tag{3}$$

where p stands for the targeting probability values, $\mathbb{I}$ denotes the indicator function, and $\hat{F}_i(\cdot)$ is the CDF of the NB distributions parameterized by $\hat{\mu}_i$ and $\hat{\alpha}_i$. The primary goal is to ensure that, as the dataset size increases, the predicted CDF converges to the true one. Such an alignment is congruent with quantile definitions, suggesting, for instance, that 95% of the genuine data should fall beneath the 95% percentile. Rather than striving for a perfect match across $\forall p \in [0, 1]$, we mainly focus on calibrating the lower bound and upper bound.

Empirically, we leverage the calibration set S to calibrate the uncertainty in data. The empirical CDF for a given value $p \in \hat{F}_i(y_i)$ is computed [17, 23]:

$$\hat{\mathcal{P}}(p) = \frac{|\{y_i | \hat{F}_i(y_i) \le p, i = 1, \ldots, |S|\}|}{|S|}. \tag{4}$$

The calibration process entails mapping a calibration function C, such as isotonic regression or Platt scaling, to the point set $\{(p, \hat{\mathcal{P}}(p))\}_{i=1}^{|S|}$. Consequently, the composition $C \circ \hat{F}$ ensures that $\hat{\mathcal{P}}(p) = p$ within the calibration dataset. This composition can then be applied to the test set to enhance the reliability of predictions.

4 Sparsity-Aware Uncertainty Calibration

4.1 QR for Uncertainty Calibration

Unlike traditional regression techniques that target the conditional mean, QR predicts specific quantiles, which are essential for calibrating PIs. Given $\hat{M}_S$ and $\mathcal{Y}_S$ from the calibration set S, the linear QR model can be expressed as:

$$Q_{\mathcal{Y}_S | \hat{M}_S}(p) = \hat{M}_S^T \beta(p), \tag{5}$$

where $\beta(p)$ signifies the coefficient associated with quantile p. The goal here is to minimize the Pinball loss [8], which is defined as:

$$L_{\text{pinball}}(y, \hat{y}, p) = \begin{cases} (p-1) \cdot (y - \hat{y}), & \text{if } y < \hat{y} \\ p \cdot (y - \hat{y}), & \text{if } y \ge \hat{y} \end{cases}. \tag{6}$$

In our context, we focus on the cases of $p = 5\%$ and 95%, and thus the quantile regressions output calibrated PIs correspond to $[Q(0.05), Q(0.95)]$.

While the NB distribution is frequently used for estimating sparse data uncertainty in previous work [7, 15, 31, 47], it can still be prone to misspecification, particularly when assumptions about data characteristics are less precise. In such instances, non-parametric (not directly modeling the distribution parameters) and distribution-agnostic (not reliant on specific distribution assumptions) methods like QR emerge as robust alternatives, compared to a direct scaling or fitting of the dispersion parameter [6, 21]. QR's resilience to data assumptions helps mitigate potential errors in model specification, a critical advantage especially in handling datasets with heavy tails where significant risks or events are concentrated—features sometimes not fully captured by parametric models [13].

QR is also adept at addressing heteroscedasticity, where variance shifts with the response variable, a situation often oversimplified by a single dispersion parameter in parametric calibration methods [3, 45]. Its ability to handle sparse datasets and accommodate variance fluctuations, particularly in scenarios with numerous zeros, further underscores its utility. Enhanced with basis expansion techniques, QR can discern complex patterns across various quantiles, making it an ideal choice for scenarios where zero outcomes are prevalent and a nuanced understanding of data patterns and risks is crucial.

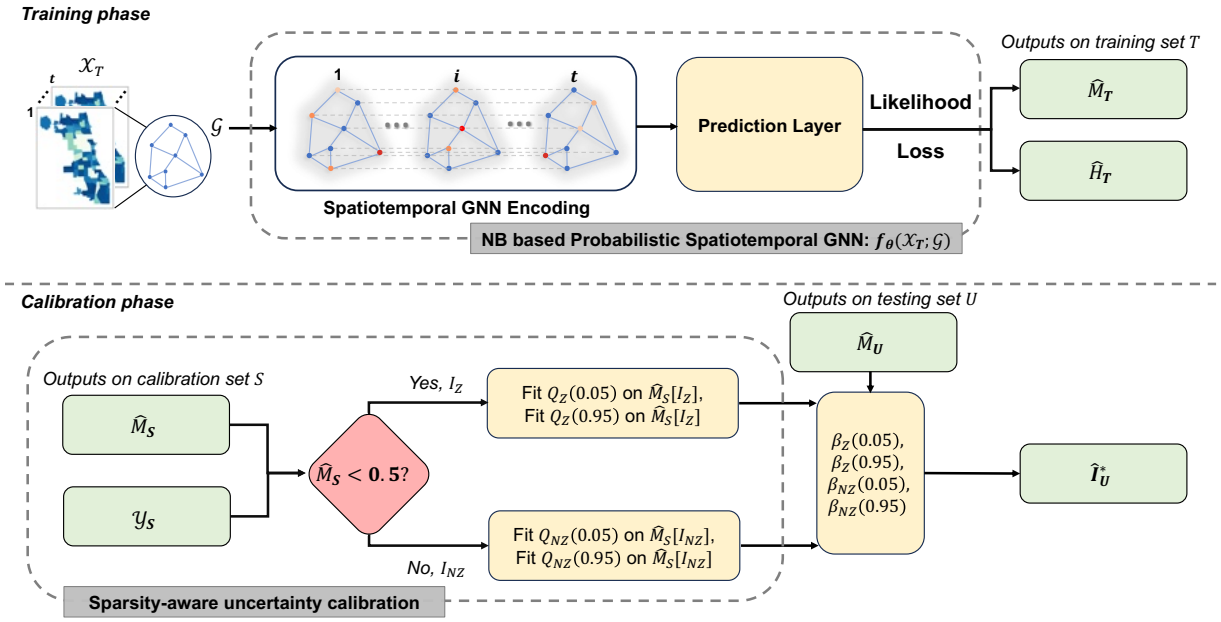


Figure 1: Experiments and modules of the SAUC framework. We modify existing ST-GNN models into probabilistic ones with NB distribution parameters as the outputs. Sparsity-aware uncertainty calibration is then conducted to obtain the quantile regression results and calibrated confidence interval for forecasting. These two steps can be easily and practically generalized to all existing ST-GNN models.

4.2 SAUC Framework

Our SAUC framework is a two-step post-hoc calibration framework tailored for sparse data, as detailed in Figure 1. In the training phase, we trained the modified NB-based probabilistic ST-GNNs to obtain the model parameters θ (modifications detailed in Section 5.2). In the calibration phase, this training yields $\hat{M}_S, \hat{H}_S$ on the calibration set for uncertainty calibration. We partition the calibration set into N bins based on the granularity of data variance we are interested in. We set $N = 15$ by default, leaving its selection discussed in Section 5.6. Inspired by zero-inflated models, predictions in the calibration set S are bifurcated based on indices: $I_Z = \{\hat{M}_S < 0.5\}$ for prediction values nearing zero, and $I_{NZ} = \{\hat{M}_S \geq 0.5\}$ for non-zero predictions. The value of 0.5 is a natural threshold for rounding operations to either 0 or 1. Following this partitioning, two QR models, Q_Z and Q_{NZ}, are trained separately, calibrating quantiles $p = \{.05, .95\}$ to fine-tune the mean and PI predictions. The calibrated model is then applied to predict model outputs on the testing set. The pseudocode is provided in Algorithm 1, and its implementation can be found in the anonymous GitHub [1].

The SAUC framework focuses on calibrating PIs rather than modifying distribution parameters, aligning with the principles of previous post-hoc calibration methods like isotonic regression and Platt scaling [17, 25]. These methods are designed to refine a model's outputs without directly altering internal parameters such as the parameters μ and α of NB distributions. This strategy ensures a distinct separation between modeling and calibration, enhancing flexibility and broad applicability. It avoids complicating the calibration process and potentially impacting the core modeling objectives. Additionally, post-hoc methods have demonstrated practical effectiveness and computational efficiency in improving the probabilistic accuracy of predictions, without necessitating retraining or modifications to the original model.

Song et al. [33] argued that quantile regressions might not retain true moments for the targets with similar predicted moments because of the global averaging effects. Our SAUC framework addresses this limitation because it creates local segments based on

Algorithm 1 SAUC Framework

Require: $\hat{M}_S, \mathcal{Y}_S, N, \hat{M}_U, p \leftarrow \{.05, .95\}$
1: Define the bin thresholds $\mathcal{T}_S$ using percentiles of $\mathcal{Y}_S$.
2: Initialize U^*, L^* in the same size as $\hat{M}_U$.
3: **for** each bin i in $[1, N]$ **do**
4: Extract indices I of instances in bin i based on $\mathcal{T}_S$.
5: Split I to I_{NZ} for $\hat{M}_S \geq 0.5$ and I_Z for $\hat{M}_S < 0.5$.
6: **for** each set i in $\{I_{NZ}, I_Z\}$ **do**
7: **for** each value q in p **do**
8: Fit $Q(q)$ on $\hat{M}_S[i]$ and $\mathcal{Y}_S[i]$ based on Equation 5 and obtain $\beta(q)$.
9: **end for**
10: $L^*[i] \leftarrow \hat{M}_U^T \beta(.05)$, $U^*[i] \leftarrow \hat{M}_U^T \beta(.95)$
11: **end for**
12: **end for**
13: $\mathcal{I}^* \leftarrow [L^*, U^*]$.
14: **return** $\mathcal{I}^*$.

[1] https://github.com/AnonymousSAUC/SAUC

zero and non-zero values. The zero segment, potentially dominant, requires precision in calibration due to its binary implications in real-world applications. For the non-zero subset, the aim is to ensure that quantiles mirror actual event frequencies, resonating with the typical research focus on discerning relative risks or magnitudes of events.

4.3 Calibration Metrics for Asymmetric Distributions

4.3.1 Expected Normalized Calibration Error. While calibration errors are clearly established for classification tasks using Expected Calibration Error (ECE) [12, 19, 26], the definition is much less studied in the regression context, where prediction errors and intervals supersede accuracy and confidence. Based on Equation 3 and 4, we aim to calibrate the length of the prediction interval with the standard error, i.e.,

$$\left[\mathbb{E}[(\hat{\mu}_i - Y_i)^2 | \hat{\mathcal{I}}_i = \mathcal{I}_i]\right]^{1/2} = c|\mathcal{I}_i|, \ \forall i \in \{1, \ldots, |U|\}. \tag{7}$$

Note that $\mathcal{I}_i = [F_i^{-1}(.05), F_i^{-1}(.95)]$ is the realization of the predicted 5%-95% confidence interval and $|\mathcal{I}_i|$ denotes its width. Calibrating the model's standard errors with the 90% confidence interval width reveals a linear relationship with coefficient c, specifically for symmetric distributions like Gaussian. As an approximation, we continue to use the z-score of Gaussian distribution to compute the coefficient c. The width of the 5%-95% confidence interval is roughly $2 \times 1.645 = 3.29$ of the standard error, inversely proportional to c. Empirical results, illustrated in Figure 2, support a linear correlation between the ratio of the predicted interval $\hat{\mathcal{I}}$ and the estimated standard error of the distribution. In our experiments, this slope ranges from 0.3 to 0.4, aligning closely with our theoretical assumption. Consequently, this leads to a value of $c \approx \frac{1}{3.29} \approx 0.303$.

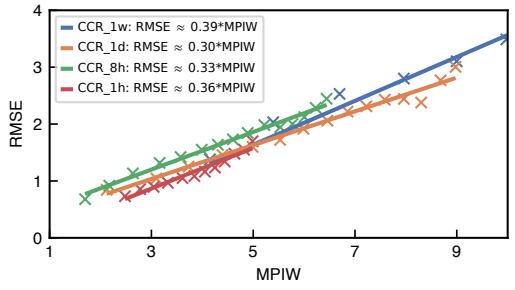


Figure 2: Explore the linear correlation between MPIW and RMSE using different cases of the CCR data.

Therefore, we could diagnose the calibration performance by comparing value differences on both sides of the equation. Similar to Algorithm 1, we partition the indices of test set samples into N evenly sized bins based on sorted predicted PI width, denoted as $\{B_j\}_{j=1}^{N}$. Each bin corresponds to a PI width that falls within the intervals: $[\min_{i \in B_j}\{|\mathcal{I}_i|\}, \max_{i \in B_j}\{|\mathcal{I}_i|\}]$. Note that the intervals are non-overlapping and their boundary values are increasing. Based on the discussion of Levi et al. [23] and Equation 7, for j-th bin, we define the Expected Normalized Calibration Error (ENCE) based on Root Mean Squared Error (RMSE) and Mean Prediction Interval Width (MPIW):

$$ENCE = \frac{1}{N}\sum_{j=1}^{N}\frac{|c \cdot MPIW(j) - RMSE(j)|}{c \cdot MPIW(j)}, \tag{8}$$

where the *RMSE* and *MPIW* are defined as:

$$RMSE(j) = \sqrt{\frac{1}{|B_j|}\sum_{i \in B_j}(\hat{\mu}_i - y_i)^2}, \tag{9}$$

$$MPIW(j) = \frac{1}{|B_j|}\sum_{i \in B_j}|\hat{\mathcal{I}}_i|. \tag{10}$$

The proposed ENCE normalizes calibration errors across bins, sorted and divided by $c \cdot MPIW$, and is particularly beneficial when comparing different temporal resolutions and output magnitudes across datasets. The $c \cdot MPIW$ and observed *RMSE* per bin should be approximately equal in a calibrated forecaster.

4.3.2 Coverage. In uncertainty calibration, coverage directly assesses the alignment between predicted uncertainty intervals and the true variability of the data. The coverage is calculated as how many points fall within the confidence interval, written as:

$$Coverage_{0.05-0.95} = \frac{1}{|U|}\sum_{i \in U} 1_{(y_i \geq \hat{F}_i^{-1}(0.05) \ \& \ y_i \leq \hat{F}_i^{-1}(0.95))}, \tag{11}$$

where $\hat{F}_i^{-1}(0.05)$ and $\hat{F}_i^{-1}(0.95)$ are the predicted lower and upper confidence interval bounds, respectively. The expected coverage should be 0.9 for an ideal calibration. A coverage value closer to the ideal value indicates that the predicted intervals encompass a large proportion of the observed data points, signifying that the uncertainty estimates are accurate and reliable.

Both ENCE and coverage evaluate probabilistic model performance but focus on different aspects: 1) Coverage is calculated at the point distribution level, while ENCE operates at the binning level, allowing detailed analysis across different ranges of predicted uncertainty; 2) Coverage measures the proportion of observed outcomes within PIs, assessing the model's ability to capture data variability directly. Meanwhile, ENCE evaluates the deviation between expected and actual calibration, providing a normalized error metric for accuracy and sharpness. As an extension of the ECE in previous calibration work [12, 17, 19, 32, 34], ENCE could adapt to distributions for sparse data like NB distributions while keeping the theoretical meaning of calibration error.

5 Experiments

5.1 Data Description

Two spatiotemporal datasets are used: (1) Chicago Traffic Crash Data (CTC)[2] sourced from 277 police beats between January 1, 2016 and January 1, 2023. The CTC data records show information about each traffic crash on city streets within the City of Chicago limits and under the jurisdiction of Chicago Police Department; (2) Chicago Crime Records (CCR)[3] derived from 77 census tracts spanning January 1, 2003 to January 1, 2023. This dataset reflects reported incidents of crime (with the exception of murders where

[2]https://data.cityofchicago.org/Transportation/Traffic-Crashes-Crashes/85ca-t3if
[3]https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-Present/ijzp-q8t2

data exists for each victim) that occurred in the City of Chicago. Despite both CTC and CCR data originating from the Chicago area, their disparate reporting sources lead to different spatial units: census tracts for CCR and police beats for CTC. Both datasets use the first 60% timesteps for training, 20% for calibration and validation, and 20% for testing. Full data descriptions can be found in Table 1.

Dataset	Resolution	Size	Sparsity	Mean	Max
CCR	1-hour	(77, 175321)	67%	0.3	59
	8-hour	(77, 21916)	18%	2.7	206
	1-day	(77, 7306)	4%	8	223
	1-week	(77, 1044)	< 0.1%	55.8	539
CTC	1-hour	(277, 61369)	96%	< 0.1%	5
	8-hour	(277, 7672)	76%	0.4	9
	1-day	(277, 2558)	47%	1.1	15
	1-week	(277, 366)	7%	7.3	45

Table 1: Characteristics of various datasets showing variation in sparsity at different temporal resolutions. Dataset sizes are represented as (spatial, temporal) dimension pairs. Sparse cases are marked grey.

The temporal resolutions of the datasets are varied to demonstrate the ubiquitous sparsity issue and its practical significance in spatiotemporal analysis. Four temporal resolutions, 1 hour, 8 hours, 1 day, and 1 week are created for CTC and CCR datasets. We designate the 1-hour and 8-hour cases of Crash and Crime datasets as sparse instances, due to a higher prevalence of zeros.

The histogram in Figure 3 shows traffic crash counts in all police beats across different temporal resolutions. Notably, higher resolutions show noticeable skewness and clustering of values at lower counts, illustrating the sporadic nature of traffic incidents.

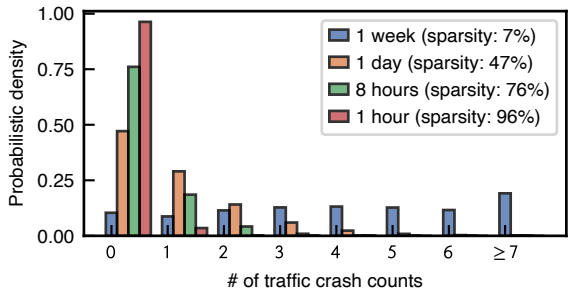


Figure 3: Histogram of number traffic crashes in the CTC data in different temporal resolutions

In the spatiotemporal data analysis, when we focus on more granular and responsive modeling, higher spatial or temporal resolutions are unavoidable. This observation emphasizes the importance of equipping predictive models with capabilities to adeptly manage sparse data, thereby more accurately mirroring real-world temporal patterns.

For adjacency matrices, we calculate geographical distances d_{ij} between the centroids of regions i and j, representing census tracts or police beats. This distance is then transformed into a similarity measure, $\mathbf{A}_{ij} = e^{-d_{ij}/0.1}$, where 0.1 is a scaling parameter, thus forming an adjacency matrix [42].

5.2 Modified Spatiotemporal Models

To demonstrate our model's ability to generalize, we modify three popular ST-GNN forecasting models: Spatio-temporal Graph Convolution Network (STGCN) [42], Graph WaveNet (GWN) [41], and Dynamic Spatial-Temporal Aware Graph Neural Network (DSTAGNN) [20]. These models, originally designed for numerical outputs, are adapted to output the location parameter μ and dispersion parameter α of NB distributions. We add two linear layers to the neural network structures to learn these parameters separately, renaming the models STGCN-NB, GWN-NB, and DSTAGNN-NB, respectively. While NB distributions are used as an example, the output parameters are free to be adjusted based on data assumptions. Lastly, we replace the mean square error loss with the likelihood loss defined in Equation 12:

$$\begin{aligned}\mathcal{L}(\mu,\alpha,y) = &-\left[y\cdot\log\left(\frac{\mu+\epsilon}{\mu+\alpha+2\epsilon}\right)\right]\\ &-\Gamma(y+\alpha+\epsilon)+\Gamma(y+1)+\Gamma(\alpha+\epsilon),\\ &-\alpha\cdot\log\left(\frac{\alpha+\epsilon}{\mu+\alpha+2\epsilon}\right)+\lambda\cdot||\alpha||^2\end{aligned} \tag{12}$$

where y is the target variable, μ and α are model outputs, Γ is the gamma function, λ is the regularization parameter, and ϵ is a small constant added to improve numerical stability. Equation 12 is derived from the NB likelihood controlled by μ and α. Similar likelihood loss could be derived when choosing other distributions.

The loss function ensures the predicted NB distribution aligns well with the true target values. It penalizes predictions where the mean of the distribution (μ) is far from the actual target values. Furthermore, it incorporates the shape of the distribution via the dispersion parameter (α) to capture the variability in the data. The loss function also includes a regularization term to prevent overfitting by discouraging overly large values of α.

All our experiments are implemented on a machine with Ubuntu 22.04, with Intel(R) Core(TM) i9-10980XE CPU @ 3.00GHz CPU, 128GB RAM, and NVIDIA GeForce RTX 4080 GPU.

Dataset	STGCN	GWN	DSTAGNN
CCR_1h	0.637 (0.703)	0.962 (0.766)	1.226 (1.117)
CCR_8h	2.386 (2.326)	2.052 (2.376)	2.272 (2.693)
CTC_1h	0.214 (0.220)	0.980 (0.220)	1.074 (0.871)
CTC_8h	0.633 (0.726)	0.924 (0.746)	1.289 (1.177)

Table 2: RMSE of original model outputs and the modified models (shown in the parenthesis) using sparse datasets.

As shown in Table 2, the model modification is successful without losing accuracy. In fact, the NB modification performs better in the sparse dataset as the NB distributions are more suitable for the nature of discrete data. Full implementation details and comparison of the modified ST-GNN models can be found in Appendix A.1.

Method / Dataset	Full observations: Before calibration	Histogram binning	Isotonic regression	Temp. scaling	Platt scaling	QR	SAUC	Zero-only targets: Before calibration	Histogram binning	Isotonic regression	Temp. scaling	Platt scaling	QR	SAUC
STGCN-NB Outputs														
CCR_1h	1.160	0.410	0.517	0.887	0.443	0.235	**0.198**	1.167	0.361	0.386	0.473	0.477	0.273	**0.267**
CCR_8h	1.389	0.344	0.417	0.385	0.373	0.392	**0.336**	1.065	0.568	0.578	0.427	0.162	0.175	**0.127**
CCR_1d	0.413	0.593	0.546	0.364	0.294	0.297	**0.192**	1.266	1.047	0.979	1.167	0.840	0.843	**0.738**
CCR_1w	0.855	0.784	0.773	0.765	0.579	0.272	**0.242**	0.354	0.362	0.332	0.287	0.345	0.155	**0.113**
CTC_1h	0.578	0.227	0.200	1.390	0.452	0.388	**0.165**	3.211	0.415	0.363	2.042	0.646	0.416	**0.078**
CTC_8h	0.323	0.320	0.255	0.570	0.297	0.241	**0.233**	0.330	0.247	0.276	0.551	0.230	0.196	**0.184**
CTC_1d	0.722	0.050	**0.011**	0.862	0.060	0.048	0.047	0.434	0.128	**0.027**	5.451	0.152	0.379	0.150
CTC_1w	2.279	1.227	1.527	0.914	2.265	0.378	**0.365**	1.823	2.384	2.009	1.160	1.343	0.767	**0.755**
GWN-NB Outputs														
CCR_1h	1.200	1.022	0.889	1.149	0.834	0.604	**0.493**	2.042	1.796	1.274	1.710	1.356	0.476	**0.375**
CCR_8h	0.670	0.636	0.612	0.696	0.655	0.597	**0.566**	1.398	0.389	0.492	1.179	0.478	0.308	**0.186**
CCR_1d	0.997	0.662	0.633	0.581	0.562	0.587	**0.504**	1.158	0.583	0.580	0.725	0.747	0.644	**0.563**
CCR_1w	0.858	0.938	0.935	**0.611**	0.931	0.857	0.829	0.335	0.218	0.219	**0.161**	0.219	0.250	0.172
CTC_1h	0.868	0.816	0.891	0.859	0.634	0.832	**0.290**	1.300	0.712	0.561	1.001	0.143	0.998	**0.139**
CTC_8h	0.931	0.416	0.388	1.381	0.169	0.226	**0.148**	2.213	0.720	0.659	1.995	0.421	0.776	**0.221**
CTC_1d	0.523	0.420	0.448	0.571	0.523	0.432	**0.139**	0.250	0.464	0.405	0.248	0.484	0.439	**0.210**
CTC_1w	0.408	0.115	**0.114**	0.259	0.236	0.241	0.239	0.466	0.479	**0.436**	0.457	0.451	0.476	0.450
DSTAGNN-NB Outputs														
CCR_1h	0.328	0.695	0.695	0.958	0.667	0.230	**0.107**	0.292	0.773	0.752	4.853	0.900	0.120	**0.102**
CCR_8h	0.895	0.941	0.940	0.604	0.928	0.443	**0.440**	0.811	0.905	0.901	0.419	0.930	0.572	**0.314**
CCR_1d	0.962	0.950	0.950	0.810	0.946	0.792	**0.709**	0.822	0.790	0.788	0.619	0.797	0.235	**0.171**
CCR_1w	0.999	1.000	1.000	0.987	1.000	0.977	**0.952**	0.260	0.140	0.140	**0.138**	0.139	0.150	0.149
CTC_1h	0.398	0.187	0.121	0.138	0.129	0.159	**0.119**	0.937	0.652	0.661	1.222	0.970	0.631	**0.229**
CTC_8h	1.462	0.666	0.669	0.473	0.767	0.505	**0.452**	0.694	0.575	0.608	0.586	0.950	0.612	**0.523**
CTC_1d	1.393	0.776	0.774	1.304	0.725	0.481	**0.402**	1.650	0.789	0.771	1.259	1.000	0.637	**0.503**
CTC_1w	0.890	0.876	0.874	**0.717**	0.840	0.824	0.833	0.980	0.757	0.743	**0.498**	0.827	0.679	0.550

Table 3: ENCE of the calibration methods. Bold fonts mark the best and underlines denote the second-best calibration results.

5.3 Baseline Calibration Methods

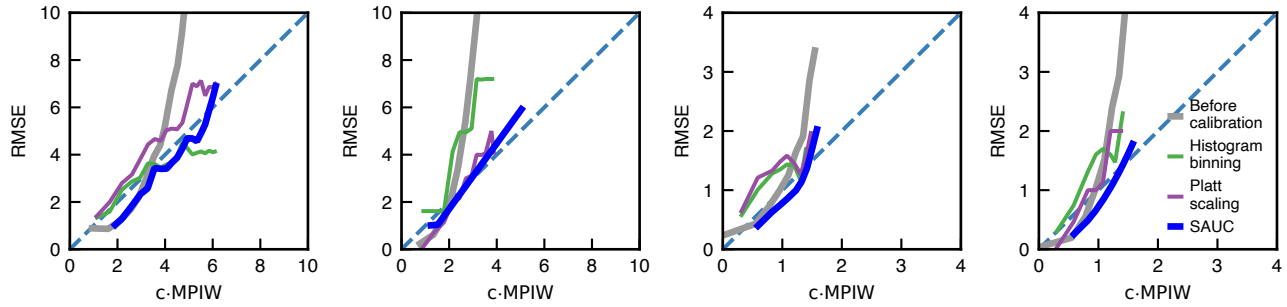
We compare the SAUC framework with existing post-hoc calibration methods proven to be effective in the regression tasks: (1) *Isotonic regression* [25], which ensures non-decreasing predictive probabilities, assuming higher model accuracy with increased predicted probabilities; (2) *Temperature scaling* [18], which scales the model's output using a learned parameter, whose efficacy for regression models depends on the output's monotonicity in relation to confidence; (3) *Histogram binning* [43], a model partitioning predicted values into bins, calibrating each independently according to observed frequencies, yet might lack flexibility for diverse prediction intervals; (4) *Platt scaling* [17], which applies a scaling transformation to the predicted values with the function derived from minimizing observed and predicted value discrepancies, providing linear adaptability but potentially inadequate for complex uncertainty patterns; (5) *Quantile regression* [8], see Section 4.1.

5.4 Calibration Results Comparison

5.4.1 ENCE Evaluation. Table 3 presents the calibration performance of baseline models and our proposed framework in different cases, measured by ENCE. The term "zero-only targets" denotes ENCE computation solely on true zero target values.

Table 3 highlights the superior performance of the SAUC framework across most datasets, with bold type indicating the best results and underlines denoting the second-best. Our findings reveal that the SAUC framework typically outperforms other models, though it might fall short in aggregated cases, such as those with a 1-week resolution. Notably, SAUC demonstrates approximately an overall 23% reduction in ENCE compared to the second-best model with full observations and a similar 20% reduction in sparsity entries. This reduction is assessed by computing the average percentage improvement relative to the second-best model across all scenarios. The separation of zero and non-zero data during calibration proved particularly beneficial, effectively addressing the issues of heteroscedasticity and zero inflation. However, some calibration methods may worsen errors due to overfitting, particularly evident in temperature scaling and isotonic regression.

Histogram binning and isotonic regression exhibit similar results among baseline models, facilitated by the non-decreasing nature of Equation 4. Temperature scaling performs well in coarse temporal resolutions, while Platt scaling excels in sparse scenarios. QR consistently ranks second-best, with our SAUC framework closely matching QR in aggregated non-sparse cases and outperforming notably in sparse scenarios.



(a) Full observations, CCR_8h (b) Zero-only targets, CCR_8h (c) Full observations, CCR_1h (d) Zero-only targets, CCR_1h

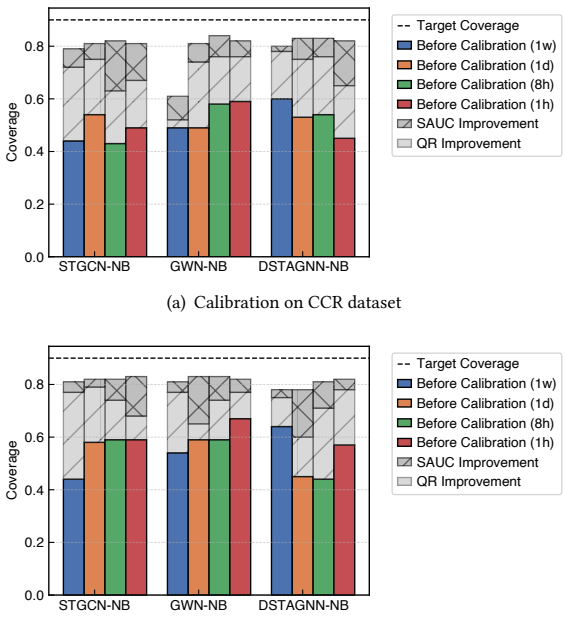
Figure 4: Reliability diagrams of different calibration methods applied to STGCN-NB outputs on different components of CCR_8h and CCR_1h data. Results closer to the diagonal dashed line are considered better.

5.4.2 Reliability Diagram Evaluation. Utilizing Equations 7 and 8, we refine the conventional reliability diagram to evaluate the calibration efficacy of a model [18]. This involves partitioning predictions into bins based on $\mathcal{I}$, after which we compute the RMSE and MPIW employing the calibrated μ^* and $|\mathcal{I}^*|$ for each respective bin.

Figure 4 presents a comparative reliability diagram of STGCN-NB outputs prior to calibration, alongside Isotonic regression, Platt scaling, and our SAUC framework, each applied to different segments of the CCR_8h dataset. We selectively highlight representative non-parametric and parametric baseline calibration approaches based on their performance. The diagonal dashed line symbolizes the calibration ideal: a closer alignment to this line indicates superior calibration.

A closer examination of Figure 4(a) reveals that both baseline models and our SAUC framework adeptly calibrate the model outputs across all observations. Yet, focusing solely on the reliability diagram of our calibrated outputs for ground-truth zero values, as depicted in Figure 4(b), there is a noticeable deviation from the ideal line. Conversely, our SAUC framework demonstrates commendable alignment with the ideal when authentic data are zeros. This observation remains consistent for the CCR_1h dataset, as seen in Figures 4(c) and 4(d). By contrasting Figures 4(a) and 4(c), it becomes evident that the calibration performance diminishes in scenarios with finer resolutions. Nevertheless, our SAUC framework retains commendable efficacy, particularly when predicting zero entries, attributed to our distinctive calibration approach coupled with QR's adeptness at managing heteroscedasticity, especially when predictions on zeros exhibit considerable variability. Contemporary probabilistic spatiotemporal prediction models predominantly prioritize refining MPIWs for enhanced accuracy, often unintentionally neglecting practical safety considerations. Our calibration strategy offers outputs with reinforced reliability.

5.4.3 Coverage Evaluation. We further evaluate the calibration performance through coverage, defined in Equation 11. The calibration improvements achieved using the SAUC framework and QR are illustrated in Figure 5, applied to various pre-calibrated models across different resolutions of the CCR and CTC datasets. The QR improvement in Figure 5 signifies the enhancement in coverage by applying QR to the pre-calibrated model, while SAUC improvement denotes the further enhancement over QR. Across all models and resolutions, calibration markedly enhances coverage. Initially, coverage levels were consistently below the target (0.9), exhibiting notable variations depending on the temporal resolution. Following calibration, all models and datasets achieve a similar level of coverage, closer to the target. Particularly on the sparser datasets (resolutions 1h and 8h), the improvements of SAUC compared with QR are generally more pronounced, notably for model performance on the CCR datasets. These findings underscore the efficacy of the SAUC framework in enhancing model calibration and ensuring improved prediction reliability across diverse temporal resolutions and models, particularly when handling sparser data inputs.



(a) Calibration on CCR dataset

(b) Calibration on CTC dataset

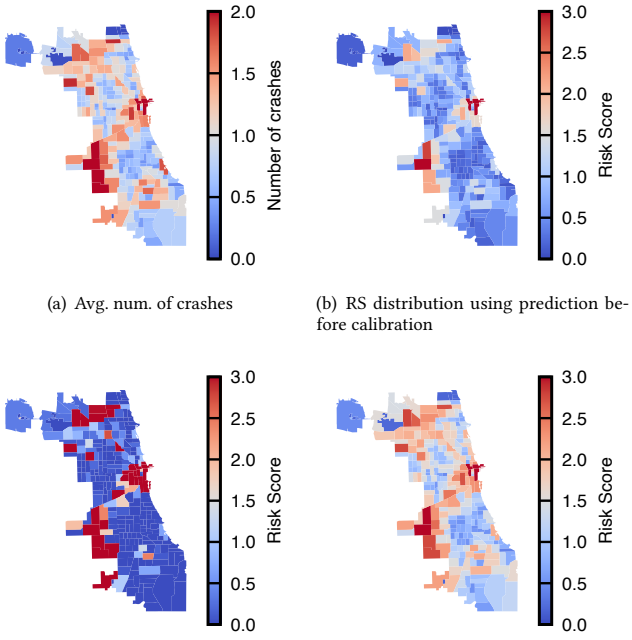
Figure 5: Coverage improvements by applying SAUC framework and QR on different data sets and models.

5.5 Risk Analysis

To evaluate the spatial effects of spatiotemporal prediction after SAUC calibration, we introduce Risk Score (RS), an enhancement of the traditional metric defined as the product of *event probability* and *potential loss magnitude* [24, 35]:

$$RS = \hat{\mu} \times |\hat{I}|. \tag{13}$$

This metric integrates anticipated risk ($\hat{\mu}$) and prediction uncertainty ($|\hat{I}|$), attributing higher RS to regions with both high incident frequency and uncertainty. This RS metric assigns higher risk to predictions that are characterized by both large predicted mean values and high uncertainty.



(a) Avg. num. of crashes

(b) RS distribution using prediction before calibration

(c) RS distribution using calibrated prediction via Platt scaling

(d) RS distribution using using calibrated prediction via SAUC

Figure 6: Traffic crash accident distributions and calibration using CTC_8h data. All the data are averaged over the temporal dimension.

Using the CTC_8h dataset and outputs from STGCN-NB, we demonstrate the importance of SAUC for evaluating the risks of traffic accidents. Figure 6 compares the average number of crashes and average RS values over time for each region, pre and post-calibration. Prior to calibration, as depicted in Figure 6(b), the RS in regions, especially Chicago's northern and southern sectors, did not provide actionable insights, given their frequent traffic incidents reported by Figure 6(a). This is due to the tight PIs from pre-calibration method outputs lower the RS. As shown in Figure 6(c), RS distribution from Platt scaling calibrated predictions could highlight certain risky areas, but still miss the major crash regions in southern Chicago. In contrast, SAUC-calibrated predictions, as seen in Figure 6(d), unveil coherent spatial distribution patterns highly consistent with Figure 6(a), demonstrating the effectiveness for urban safety monitoring.

Rather than just yielding exact numerical predictions with tight confidence intervals, it is imperative that models convey equivalent severity levels when integrating uncertainty. Such an approach ensures reliable outputs in crime and accident predictions. Notably, the reliable estimation of zero or non-zero incident counts is pivotal in risk management. Existing probabilistic prediction studies focus on exact values and restricted confidence intervals, sometimes neglecting the data variability, especially around zero incidences.

5.6 Sensitivity Analysis

The number of bins N in both the SAUC framework and ENCE discussion is important. We choose the bin number from the range $N = [1, 5, 10, 15, 20, 25, 30]$ and discuss the ENCE change with respect to the bin number. We conduct the sensitivity analysis using the four cases in the crime dataset using STGCN-NB model outputs.

Dataset/Bin	1	5	10	15	20	25
Full Observations						
CCR_1h	0.513	0.302	0.299	0.198	0.249	0.292
CCR_8h	0.096	0.113	0.185	0.336	0.123	0.115
CCR_1d	0.093	0.167	0.185	0.192	0.189	0.207
CCR_1w	0.074	0.216	0.323	0.242	0.254	0.272
Zero-only Observations						
CCR_1h	0.650	0.689	0.604	0.267	0.229	0.501
CCR_8h	0.642	0.583	0.504	0.127	0.172	0.345
CCR_1d	0.641	1.181	1.096	0.738	1.025	0.958
CCR_1w	1.445	0.950	0.361	0.113	0.363	0.567

Table 4: ENCE on the full observations and zero-only observations with respect to different bin numbers.

From Table 4, it is evident that the number of bins significantly impacts the SAUC framework and ENCE performance. Both very small and very large bin numbers may result in suboptimal ENCE values. Intuitively, with only one bin, all data points are grouped together, making it difficult to differentiate levels of variance as the comparison involves the overall data variability and the average MPIW. Conversely, having too many bins means each bin contains few data points, leading to fluctuations due to the small sample size. Therefore, an optimal bin number range between 10 and 20 is preferable for achieving reliable ENCE.

5.7 Discussion on Distribution Assumptions

A common implementation question arises: is the NB distribution flexible enough to capture the response distributions? The focus of this work lies in uncertainty calibration methods for sparse data, where specific selections of distributions are less discussed. Nevertheless, it is imperative to examine whether the presented distributions offer sufficient flexibility.

In comparison to the NB distribution, we have also implemented Poisson and Gaussian distributions as estimated distributions for our model. Models' modification remains unchanged following Section 3.1, with Equation 2 replaced by estimations of corresponding

parameters for Poisson and Gaussian distributions. The last layer is modified to output the distribution parameters of the Poisson distributions exclusively. For the Poisson distribution, only one parameter λ is outputted, and the loss function can be similarly defined as:

$$\mathcal{L}(\hat{\lambda}, y) = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{\lambda}_{s,i} - y_i \cdot \log(\lambda_{s,i})\right), \tag{14}$$

where $\hat{\lambda}_s = \hat{\lambda} + \epsilon$, with $\hat{\lambda}$ representing the predicted rate parameters, target representing the target counts, and ϵ a small constant (e.g., 10^{-10}) added for numerical stability. Equation 14 is derived from the Poisson likelihood function, ensuring robust and stable learning for our spatiotemporal prediction models.

Similarly, for Gaussian distribution, we estimate the location and scale parameters μ and σ with the loss function:

$$\mathcal{L}(\hat{\mu}, \hat{\sigma}, y) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{2}\log(2\pi\hat{\sigma}_i^2) + \frac{(y_i - \hat{\mu}_i)^2}{2\hat{\sigma}_i^2}\right) + \lambda||\hat{\sigma}||_2, \tag{15}$$

where penalization for the variance is also added, similar to Equation 12. We discuss the prediction RMSE and ENCE performance of STGCN, STGCN-NB, STGCN-Poisson, and STGCN-Gaussian, as shown in Table 5.

Dataset	Metric	Original	NB	Poisson	Gaussian
CCR_1h	RMSE	0.637	0.703	0.581	0.741
	ENCE	/	1.160	1.692	3.292
CCR_8h	RMSE	2.386	2.326	2.419	2.303
	ENCE	/	1.389	0.913	30.286
CCR_1d	RMSE	5.167	4.286	6.296	6.701
	ENCE	/	0.413	0.999	67.267
CCR_1w	RMSE	23.333	20.242	23.907	31.044
	ENCE	/	0.855	0.100	168.889
CTC_1h	RMSE	0.214	0.220	0.282	0.321
	ENCE	/	0.578	3.079	0.962
CTC_8h	RMSE	0.633	0.726	0.962	0.972
	ENCE	/	0.323	0.960	2.320
CTC_1d	RMSE	1.148	1.786	1.138	1.130
	ENCE	/	0.722	1.509	3.351
CTC_1w	RMSE	3.317	3.929	3.450	3.321
	ENCE	/	2.279	0.145	5.867

Table 5: Comparison of RMSE and ENCE for the original model and its NB, Poisson, and Gaussian modification.

Based on Table 5, it is clear that the Poisson and Gaussian modifications exhibit similar performance to the NB distribution in terms of RMSE based on the experimental results. However, NB demonstrates better ENCE when capturing sparse data cases compared with the other two distributions, thus it is selected as the distribution predominantly discussed in this paper. More dedicated data distributions' assumptions can be explored, as demonstrated by the work of [4, 15, 47, 48].

5.8 Complexity Discussion

From Equation 5, we know that the complexity of the SAUC framework depends on the size of the data. Since our method is a post-hoc method, the added complexity is easier to address in terms of runtime. Same as the settings in sensitivity analysis, we visualize the model runtime on different bin numbers and data resolutions by running STGCN-NB, shown in Figure 7. It is clear that the selection of the bin number does not influence the complexity and runtime. Moreover, by reducing the resolution from 1 hour to 8 hours, 1 day, and 1 week, the runtime decreases nearly exponentially with the change in the temporal resolutions of the data.

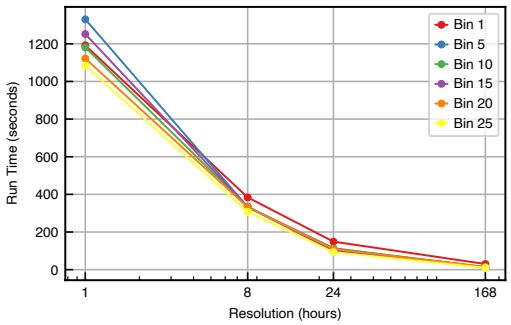


Figure 7: Runtime of the calibration methods on the full observations with different bin numbers and data resolutions of CCR dataset. The x-axis scale is logarithmic.

6 Conclusion

Current spatiotemporal GNN models mainly yield deterministic predictions, overlooking data uncertainties. Given the sparsity and asymmetry in high-resolution spatiotemporal data, quantifying uncertainty is challenging. Previous ST-GNN models covert the deterministic models to probabilistic ones by changing the output layers, without proper discussions in the reasonableness of learned distributions and UQ. We introduce the SAUC framework, which calibrates uncertainties for both zero and non-zero values using the zero-inflated nature of these datasets. As a post-hoc method, SAUC is available to be adapted to any existing ST-GNN model. We further introduce new calibration metrics suited for asymmetric distributions. Tests on real-world datasets show SAUC reduces calibration errors by 20% for zero-only targets. SAUC enhances GNN models and aids risk assessment in sparse datasets where accurate predictions are critical due to safety concerns.

Acknowledgments

We extend our gratitude to the reviewers for their time and effort in reviewing and refining our manuscript.

This material is based upon work supported by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Vehicle Technology Program Award Number DE-EE0009211.

Shenhao Wang and Yuheng Bu acknowledge the support from the Research Opportunity Seed Fund (ROSF) 2023 at the University of Florida. Guang Wang is partially supported by the National Science Foundation under Grant No. 2411152

References

[1] H Abdo, Jean-Marie Flaus, and F Masse. 2017. Uncertainty quantification in risk assessment-representation, propagation and treatment approaches: application to atmospheric dispersion modeling. *Journal of Loss Prevention in the Process Industries* 49 (2017), 551–571.

[2] Deepak K Agarwal, Alan E Gelfand, and Steven Citron-Pousty. 2002. Zero-inflated models with application to spatial count data. *Environmental and Ecological statistics* 9 (2002), 341–355.

[3] Francisco Antunes, Aidan O'Sullivan, Filipe Rodrigues, and Francisco Pereira. 2017. A review of heteroscedasticity treatment with Gaussian processes and quantile regression meta-models. *Seeing Cities Through Big Data: Research, Methods and Applications in Urban Informatics* (2017), 141–160.

[4] Loris Bazzani, Hugo Larochelle, and Lorenzo Torresani. 2016. Recurrent mixture density network for spatiotemporal visual attention. *arXiv preprint arXiv:1603.08199* (2016).

[5] Wendong Bi, Lun Du, Qiang Fu, Yanlin Wang, Shi Han, and Dongmei Zhang. 2023. MM-GNN: Mix-Moment Graph Neural Network towards Modeling Neighborhood Feature Distribution. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*. 132–140.

[6] C Bishop. 2006. Pattern recognition and machine learning. *Springer google schola* 2 (2006), 531–537.

[7] Xu Chen and Surya T Tokdar. 2021. Joint quantile regression for spatial data. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 83, 4 (2021), 826–852.

[8] Youngseog Chung, Willie Neiswanger, Ian Char, and Jeff Schneider. 2021. Beyond pinball loss: Quantile methods for calibrated uncertainty quantification. *Advances in Neural Information Processing Systems* 34 (2021), 10971–10984.

[9] Marcus VM Fernandes, Alexandra M Schmidt, and Helio S Migon. 2009. Modelling zero-inflated spatio-temporal processes. *Statistical Modelling* 9, 1 (2009), 3–25.

[10] Xiaowei Gao, Huanfa Chen, and James Haworth. 2023. A spatiotemporal analysis of the impact of lockdown and coronavirus on London's bicycle hire scheme: from response to recovery to a new normal. *Geo-spatial Information Science* (2023), 1–21.

[11] Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, et al. 2021. A survey of uncertainty in deep neural networks. *arXiv preprint arXiv:2107.03342* (2021).

[12] Sebastian Gruber and Florian Buettner. 2022. Better uncertainty calibration via proper scores for classification and beyond. *Advances in Neural Information Processing Systems* 35 (2022), 8618–8632.

[13] Lingxin Hao and Daniel Q Naiman. 2007. *Quantile regression*. Number 149. Sage.

[14] Kexin Huang, Ying Jin, Emmanuel Candes, and Jure Leskovec. 2023. Uncertainty quantification over graph with conformalized graph neural networks. *NeurIPS* (2023).

[15] Xinke Jiang, Dingyi Zhuang, Xianghui Zhang, Hao Chen, Jiayuan Luo, and Xiaowei Gao. 2023. Uncertainty Quantification via Spatial-Temporal Tweedie Model for Zero-inflated and Long-tail Travel Demand Prediction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (, Birmingham, United Kingdom,) *(CIKM '23)*. Association for Computing Machinery, New York, NY, USA, 3983–3987. https://doi.org/10.1145/3583780.3615215

[16] Abbas Khosravi, Saeid Nahavandi, Doug Creighton, and Amir F Atiya. 2010. Lower upper bound estimation method for construction of neural network-based prediction intervals. *IEEE transactions on neural networks* 22, 3 (2010), 337–346.

[17] Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. 2018. Accurate uncertainties for deep learning using calibrated regression. In *International conference on machine learning*. PMLR, 2796–2804.

[18] Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. 2019. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. *Advances in neural information processing systems* 32 (2019).

[19] Ananya Kumar, Percy S Liang, and Tengyu Ma. 2019. Verified uncertainty calibration. *Advances in Neural Information Processing Systems* 32 (2019).

[20] Shiyong Lan, Yitong Ma, Weikang Huang, Wenwu Wang, Hongyu Yang, and Pyang Li. 2022. Dstagnn: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting. In *International conference on machine learning*. PMLR, 11906–11917.

[21] Max-Heinrich Laves, Sontje Ihler, Jacob F Fast, Lüder A Kahrs, and Tobias Ortmaier. 2020. Well-calibrated regression uncertainty in medical imaging with deep learning. In *Medical Imaging with Deep Learning*. PMLR, 393–412.

[22] Dan Levi, Liran Gispan, Niv Giladi, and Ethan Fetaya. 2020. Evaluating and Calibrating Uncertainty Prediction in Regression Tasks. http://arxiv.org/abs/1905.11659 arXiv:1905.11659 [cs, stat].

[23] Dan Levi, Liran Gispan, Niv Giladi, and Ethan Fetaya. 2022. Evaluating and calibrating uncertainty prediction in regression tasks. *Sensors* 22, 15 (2022), 5540.

[24] Kwok-suen Ng, Wing-tat Hung, and Wing-gun Wong. 2002. An algorithm for assessing the risk of traffic accident. *Journal of safety research* 33, 3 (2002), 387–410.

[25] Alexandru Niculescu-Mizil and Rich Caruana. 2005. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*. 625–632.

[26] Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. 2019. Measuring Calibration in Deep Learning.. In *CVPR workshops*, Vol. 2.

[27] Tim Pearce, Alexandra Brintrup, Mohamed Zaki, and Andy Neely. 2018. High-quality prediction intervals for deep learning: A distribution-free, ensembled approach. In *International conference on machine learning*. PMLR, 4075–4084.

[28] Weizhu Qian, Dalin Zhang, Yan Zhao, Kai Zheng, and JQ James. 2023. Uncertainty quantification for traffic forecasting: A unified approach. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 992–1004.

[29] Moshiur Rahman. 2023. Uncertainty-Aware Traffic Prediction using Attention-based Deep Hybrid Network with Bayesian Inference. *International Journal of Advanced Computer Science and Applications* 14, 6 (2023).

[30] Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence* 1, 5 (2019), 206–215.

[31] S Sankararaman and S Mahadevan. 2013. Distribution type uncertainty due to sparse and imprecise data. *Mechanical Systems and Signal Processing* 37, 1-2 (2013), 182–198.

[32] Maohao Shen, Yuheng Bu, Prasanna Sattigeri, Soumya Ghosh, Subhro Das, and Gregory Wornell. 2023. Post-hoc Uncertainty Learning Using a Dirichlet Meta-Model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 9772–9781.

[33] Hao Song, Tom Diethe, Meelis Kull, and Peter Flach. 2019. Distribution calibration for regression. In *International Conference on Machine Learning*. PMLR, 5897–5906.

[34] Jayaraman J Thiagarajan, Bindya Venkatesh, Prasanna Sattigeri, and Peer-Timo Bremer. 2020. Building calibrated deep models via uncertainty matching with auxiliary interval predictors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 6005–6012.

[35] David Vose. 2008. *Risk analysis: a quantitative guide*. John Wiley & Sons.

[36] Qingyi Wang, Shenhao Wang, Dingyi Zhuang, Haris Koutsopoulos, and Jinhua Zhao. 2023. Uncertainty Quantification of Spatiotemporal Travel Demand with Probabilistic Graph Neural Networks. *arXiv preprint arXiv:2303.04040* (2023).

[37] Dongxia Wu, Liyao Gao, Matteo Chinazzi, Xinyue Xiong, Alessandro Vespignani, Yi-An Ma, and Rose Yu. 2021. Quantifying uncertainty in deep spatiotemporal forecasting. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1841–1851.

[38] Ying Wu and JQ James. 2021. A bayesian learning network for traffic speed forecasting with uncertainty quantification. In *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–7.

[39] Ying Wu, Yongchao Ye, Adnan Zeb, James JQ Yu, and Zheng Wang. 2023. Adaptive Modeling of Uncertainties for Traffic Forecasting. *arXiv preprint arXiv:2303.09273* (2023).

[40] Yuankai Wu, Dingyi Zhuang, Aurelie Labbe, and Lijun Sun. 2021. Inductive graph neural networks for spatiotemporal kriging. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 4478–4485.

[41] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. 2019. Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121* (2019).

[42] Bing Yu, Haoteng Yin, and Zhanxing Zhu. 2017. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875* (2017).

[43] Bianca Zadrozny and Charles Elkan. 2001. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Icml*, Vol. 1. 609–616.

[44] Xiangyu Zhao, Wenqi Fan, Hui Liu, and Jiliang Tang. 2022. Multi-type urban crime prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 4388–4396.

[45] Yao Zheng, Qianqian Zhu, Guodong Li, and Zhijie Xiao. 2018. Hybrid quantile regression estimation for time series models with conditional heteroscedasticity. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 80, 5 (2018), 975–993.

[46] Zhengyang Zhou, Yang Wang, Xike Xie, Lei Qiao, and Yuantao Li. 2021. STUaNet: Understanding uncertainty in spatiotemporal collective human mobility. In *Proceedings of the Web Conference 2021*. 1868–1879.

[47] Dingyi Zhuang, Shenhao Wang, Haris Koutsopoulos, and Jinhua Zhao. 2022. Uncertainty quantification of sparse travel demand prediction with spatial-temporal graph neural networks. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4639–4647.

[48] Simiao Zuo, Haoming Jiang, Zichong Li, Tuo Zhao, and Hongyuan Zha. 2020. Transformer hawkes process. In *International conference on machine learning*. PMLR, 11692–11702.

A Appendix

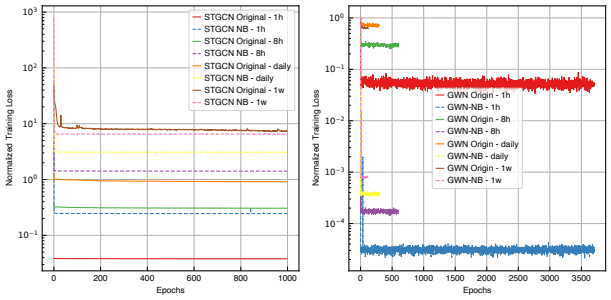
A.1 Modified Model Implementation Details

To implement the modified GNNs, we use the Github repositories for STGCN [4], GWN [5], and DSTAGNN [6] respectively. We basically keep all the hyper-parameters and important parameters as the default from the original repositories, which can be referred to in our Github repository [7]. Notice that we keep the same model parameters in both the original models and the modified models, which can be found in Table 6

Parameters/Models	STGCN	GWN	DSTAGNN
Input window length	12	12	12
Output window length	12	12	12
Learning rate	0.001	0.001	0.001
Batch size	24	24	24
Output layer	Linear	Conv2d	Linear
Training epochs	1000	100	500
Early stop	No	Yes	Yes

Table 6: Parameters of original and modified GNN models.

Training loss on CCR datasets can be found in Figure 8. All models are converged properly.



(a) Training Loss of STGCN on CCR dataset (b) Training Loss of GWN on CCR dataset

Figure 8: Training loss of STGCN, GWN, and their modifications on the CCR dataset.

The full implementation results can be found in Figure 7. It is clear that the model's performance in terms of numerical accuracy is close to the results using models before modification. In fact, the NB modification shows better performance in the sparse dataset as the NB distributions are more suitable for the nature of discrete data.

A.2 Baseline Model Implementation

We list the introductions and pseudo codes to implement the baseline models.

[4]https://github.com/FelixOpolka/STGCN-PyTorch
[5]https://github.com/nnzhan/Graph-WaveNet
[6]https://github.com/SYLan2019/DSTAGNN
[7]https://github.com/AnonymousSAUC/SAUC

Dataset	Error Metric	STGCN	STGCN-NB	GWN	GWN-NB	DSTAGNN	DSTAGNN-NB
CCR_1h	MAE	0.429	0.317	0.848	0.334	0.982	0.901
	RMSE	0.637	0.703	0.962	0.766	1.226	1.117
CCR_8h	MAE	1.499	1.452	1.395	1.478	1.475	1.198
	RMSE	2.386	2.326	2.052	2.376	2.272	2.693
CCR_daily	MAE	2.951	2.719	2.521	2.681	2.500	2.421
	RMSE	5.167	4.286	3.784	9.703	3.848	3.368
CCR_weekly	MAE	14.146	12.505	9.938	14.516	10.389	17.341
	RMSE	23.333	20.242	16.103	23.231	16.578	21.382
CTC_1h	MAE	0.084	0.044	0.960	0.044	1.049	0.832
	RMSE	0.214	0.220	0.980	0.220	1.074	0.871
CTC_8h	MAE	0.453	0.346	0.810	0.352	1.113	0.995
	RMSE	0.633	0.726	0.924	0.746	1.289	1.177
CTC_daily	MAE	0.861	1.433	0.895	1.019	1.299	0.986
	RMSE	1.148	1.786	1.154	1.550	1.596	1.353
CTC_weekly	MAE	2.474	3.618	2.423	3.279	2.502	2.956
	RMSE	3.317	3.929	3.178	3.689	3.256	3.876

Table 7: Comparisons between modified models and the original models.

A.2.1 Temperature Scaling. Temperature scaling is a parametric method that modifies the outputs of a regression model. This modification is controlled by a single parameter, the temperature T. The algorithms can be formulated as follows:

Algorithm 2 Temperature Scaling

1: Train a model on a training dataset, yielding point predictions $f(x)$.
2: Optimize the temperature T on the validation set by minimizing the mean squared error between the predicted and true values:

$$T = \arg\min_T \frac{1}{n}\sum_{i=1}^{n}(f(x_i) - y_i)^2$$

3: **for** each new input x **do**
4: Compute the model's prediction $f(x)$.
5: Apply temperature scaling with the learned T to the prediction $f(x)$, yielding scaled prediction:

$$g(f(x)) = \frac{f(x)}{T}$$

6: **end for**

Temperature scaling is an efficient and straightforward method for calibration that requires optimization of only a single parameter. By scaling the outputs, it can yield calibrated predictions without modifying the rank ordering of the model's predictions.

A.2.2 Isotonic Regression. Isotonic regression is a non-parametric method utilized for the calibration of a predictive model's outputs. In the context of regression, this method operates as shown below:

Isotonic regression makes no assumptions about the form of the function connecting the model's predictions to calibrated predictions. This flexibility allows it to fit complex, non-linear mappings, which can provide improved calibration performance when the model's outputs are not well-modeled by a simple function.

A.2.3 Histogram Binning. Histogram binning can also be applied in a regression setting to calibrate predictions. The idea is to split the range of your model's outputs into several bins, and then adjust

Algorithm 3 Isotonic Regression for Regression

Train a model on a training dataset, providing point predictions $f(x)$.
Fit an isotonic regression model on the validation set:
  Sort the predictions $f(x_i)$ and corresponding true values y_i in ascending order of $f(x_i)$.
  Solve the isotonic regression problem to find a non-decreasing sequence $g(f(x))$ that minimizes:

$$\min \sum_{i=1}^{n} (g(f(x_i)) - y_i)^2$$

for each new input x **do**
  Compute the model's prediction $f(x)$.
  Use the fitted isotonic regression model to convert the prediction $f(x)$ to a calibrated prediction: $g(f(x))$
end for

the predictions within each bin to match the average actual outcome within that bin. The specific algorithm is:

Algorithm 4 Histogram Binning

Train a model on a training dataset, yielding predictions $f(x)$.
Determine the bins for the predictions on the validation set.
for each bin B_i **do**
  Compute the average true value $\bar{y}_i$ for all examples in the bin:

$$\bar{y}_i = \frac{1}{|B_i|} \sum_{x_j \in B_i} y_j$$

  Adjust the prediction for each example in the bin to the average true value $\bar{y}_i$.
end for
for each new input x **do**
  Compute the model's prediction $f(x)$.
  Find the bin B_i that $f(x)$ falls into and adjust $f(x)$ to the average true value for that bin:

$$g(f(x)) = \bar{y}_i$$

end for

This method assumes that predictions within each bin are uniformly mis-calibrated. It is less flexible than isotonic regression, but it is simpler and less prone to overfitting, especially when the number of bins is small.

A.2.4 Platt Scaling. In a regression setting, Platt Scaling can be interpreted as applying a sigmoid function transformation to the model's outputs. The output is then considered as the mean of a Bernoulli distribution. While this might be useful in some contexts, it is not generally applicable to regression problems but can be useful in sparse datasets. A linear transformation is fitted to the model's predictions to minimize the mean squared error on the validation set. The parameters of this transformation are learned from the data, making this an instance of post-hoc calibration. Note that this adaptation might not be suitable for all regression tasks, especially those where the target variable has a non-linear relationship with the features.

Algorithm 5 Platt Scaling

Train a model on a training dataset, yielding predictions $f(x)$.
Minimize the mean squared error on the validation set between $g(f(x))$ and the true outcomes, where $g(f(x)) = af(x) + b$ is a linear transformation of the model's predictions: $a, b = \arg\min_{a,b} \frac{1}{n} \sum_{i=1}^{n} (af(x_i) + b - y_i)^2$
for each new input x **do**
  Compute the model's prediction $f(x)$.
  Apply the learned linear transformation to $f(x)$ to get the calibrated prediction: $g(f(x)) = af(x) + b$
end for