Full Bayesian Significance Testing for Neural Networks in Traffic Forecasting: Appendix

Zehua Liu[1] , **Jingyuan Wang**[1,2,3*] , **Zimeng Li**[1] and **Yue He**[4]
[1]School of Computer Science and Engineering, Beihang University, Beijing, China
[2]School of Economics and Management, Beihang University, Beijing, China
[3]Key Laboratory of Data Intelligence and Management (Beihang University), Ministry of Industry and Information Technology, Beijing, China
[4]Department of Computer Science and Technology, Tsinghua University, Beijing, China
{liuzehua, jywang, zimengli}@buaa.edu.cn, heyuethu@mail.tsinghua.edu.cn

1 Code

https://github.com/liuzh-buaa/ST-nFBST.

2 Theoretical Derivation of Optimization with Uncertainties

In this section, we provide a detailed theoretical derivation for the optimization of a Bayesian Neural Network (BNN) in the context of heteroscedastic aleatoric uncertainty and epistemic uncertainty. We demonstrate the optimization function $\mathcal{L}$ through variational inference.

2.1 Optimization function $\mathcal{L}$

In the task of Bayesian Spatial-Temporal Modeling, we use a BNN, whose parameters θ follow a distribution rather than deterministic values, to represent the underlying function $f_0 : \mathbb{R}^{\tau_1 \times |\mathcal{V}|} \to \mathbb{R}^{\tau_2 \times |\mathcal{V}|}$. Given a dataset of n samples $\mathcal{D} = \{(\boldsymbol{X}_1, \boldsymbol{Y}_1), \ldots, (\boldsymbol{X}_n, \boldsymbol{Y}_n)\}$, For a pair of observations $(\boldsymbol{X}_i, \boldsymbol{Y}_i)$, the regression process can be modeled as corrupted with Gaussian random noise:

$$\boldsymbol{Y}_i \sim \mathcal{N}(f_0(\boldsymbol{X}_i), \sigma_0^2(\boldsymbol{X}_i)). \tag{1}$$

Moreover, we assume the observation noise can vary with inputs $\boldsymbol{X}_i$ at different time points, namely heteroscedastic. For convenience, we will abbreviate $\sigma_0(\boldsymbol{X}_i)$ as σ_0 if not causing any confusion.

Before optimization, a prior distribution is assigned to model parameters θ as an initial belief $\pi(\theta)$ according to experience. This belief is gradually adjusted to fit data $\mathcal{D}$ by using the Bayesian rule. The final belief is presented as the posterior distribution

$$P(\theta|\mathcal{D}) = \frac{\pi(\theta)P(\mathcal{D}|\theta)}{P(\mathcal{D})} = \frac{\pi(\theta)\prod_{i=1}^{n} P(\boldsymbol{Y}_i|\boldsymbol{X}_i, \theta)}{\int_\Theta \prod_{i=1}^{n} P(\boldsymbol{Y}_i|\boldsymbol{X}_i, \theta)\mathrm{d}\theta}. \tag{2}$$

However, it is intractable to solve the integral in Eq (2). Through variational inference, we use a tractable distribution to approximate the real but intractable posterior distribution. Formally, variational family $Q = \{q_\vartheta : \vartheta \in \Gamma\}$ is a predefined family of tractable distributions on model parameter space Θ, where ϑ is the parameter of variational distribution and Γ is the range of ϑ. The optimal variational distribution q_{ϑ^*} is chosen from Q such that

$$\vartheta^* = \arg\min_{\vartheta \in \Gamma} \mathrm{KL}(q_\vartheta(\theta) \| P(\theta|\mathcal{D})), \tag{3}$$

where KL divergence describes the "distance" between two distributions. According to its definition, the optimization objective function can be formulated as

$$\mathcal{L} = \mathrm{KL}(q_\vartheta(\theta) \| P(\theta|\mathcal{D})) = \int_\Theta q_\vartheta(\theta) \log \frac{q_\vartheta(\theta)}{P(\theta|\mathcal{D})} \mathrm{d}\theta. \tag{4}$$

From the Bayesian rule Eq (2), Eq (4) can be simplified to

$$\begin{aligned}
\mathcal{L} &= \int_\Theta q_\vartheta(\theta) \log \frac{q_\vartheta(\theta)}{P(\theta|\mathcal{D})} \mathrm{d}\theta \\
&= \int_\Theta q_\vartheta(\theta) \log \left(q_\vartheta(\theta) \frac{P(\mathcal{D})}{P(\mathcal{D}|\theta)\pi(\theta)} \right) \mathrm{d}\theta \\
&= -\int_\Theta q_\vartheta(\theta) \log P(\mathcal{D}|\theta) \mathrm{d}\theta + \int_\Theta q_\vartheta(\theta) \log \frac{q_\vartheta(\theta)}{\pi(\theta)} \mathrm{d}\theta \\
&\quad + \int_\Theta q_\vartheta(\theta) \log p(\mathcal{D}) \mathrm{d}\theta \\
&= -\mathbb{E}\left[\log P(\mathcal{D}|\theta)\right] + \mathrm{KL}(q_\vartheta(\theta) \| \pi(\theta)) + \log P(\mathcal{D}).
\end{aligned} \tag{5}$$

The third term $\log P(\mathcal{D})$ is a constant once the data $\mathcal{D}$ is determined. In the following, we will present how to calculate the first two terms, respectively.

*Corresponding author

2.2 The first term of $\mathcal{L}$

Under the assumption of Eq (1), the log-likelihood of $\mathcal{D}$ is calculated as follows:

$$\begin{aligned}\log P(\mathcal{D}|\theta) &= \log \prod_{i=1}^{n} P(\boldsymbol{Y}_i|\boldsymbol{X}_i;\theta) \\ &= \sum_{i=1}^{n} \log P(\boldsymbol{Y}_i|\boldsymbol{X}_i;\theta) \\ &= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi}\sigma_0} \exp\left[-\frac{(\boldsymbol{Y}_i - f(\boldsymbol{X}_i;\theta))^2}{2\sigma_0^2}\right] \\ &= -\sum_{i=1}^{n} \log \sqrt{2\pi}\sigma_0(\boldsymbol{X}_i) \\ &\quad -\sum_{i=1}^{n} \frac{1}{2\sigma_0^2(\boldsymbol{X}_i)}(\boldsymbol{Y}_i - f(\boldsymbol{X}_i;\theta))^2.\end{aligned} \tag{6}$$

The second term of Eq (6) is similar to the Mean Squared Error (MSE) in the regression task within a scaling factor.

Using Monte Carlo integration, the first term of $\mathcal{L}$ can be calculated as follows:

$$-\mathbb{E}[\log P(\mathcal{D}|\theta)] \approx -\frac{1}{m}\sum_{j=1}^{m}[\log P(\mathcal{D}|\theta_j)], \tag{7}$$

where θ_j is obtained by sampling m times based on the variational distribution $q_\vartheta(\theta)$.

2.3 The second term of $\mathcal{L}$

In our experiment, we adopt popular diagonal Gaussian distributions as the prior distribution $\pi(\theta)$ and the variational distribution $q_\vartheta(\theta)$ of parameters θ. We first consider θ as a one-dimensional variable, that is:

$$\pi(\theta) \sim \mathcal{N}(\mu_\pi, \sigma_\pi^2) \tag{8}$$

and correspondingly

$$q_\vartheta(\theta) \sim \mathcal{N}(\mu, \sigma^2). \tag{9}$$

Then we get

$$\begin{aligned}\mathrm{KL}\left(q_\vartheta(\theta)\|\pi(\theta)\right) &= \int q_\vartheta(\theta)\log\frac{q_\vartheta(\theta)}{\pi(\theta)}\mathrm{d}\theta \\ &= \int q_\vartheta(\theta)\log q_\vartheta(\theta)\mathrm{d}\theta \\ &\quad -\int q_\vartheta(\theta)\log\pi(\theta)\mathrm{d}\theta \\ &= -\frac{1}{2}\left(1+\log(2\pi\sigma^2)\right) \\ &\quad +\frac{1}{2}\left(\log 2\pi\sigma_\pi^2 + \frac{\sigma^2+(\mu-\mu_\pi)^2}{\sigma_\pi^2}\right) \\ &= \log\frac{\sigma_\pi}{\sigma} + \frac{\sigma^2+(\mu-\mu_\pi)^2}{2\sigma_\pi^2} - \frac{1}{2}.\end{aligned} \tag{10}$$

Then, we consider θ follows a n-dimensional diagonal Gaussian distribution. That is,

$$\pi(\theta) \sim \mathcal{N}(\mu_\pi, \Sigma_\pi), \tag{11}$$

where

$$\mu_\pi = \begin{bmatrix}\mu_{\pi 1}\\ \mu_{\pi 2}\\ \cdots \\ \mu_{\pi n}\end{bmatrix}, \Sigma_\pi = \begin{bmatrix}\sigma_{\pi 1}^2 & 0 & \ldots & 0\\ 0 & \sigma_{\pi 2}^2 & \ldots & 0\\ \ldots & \ldots & \ldots & \ldots\\ 0 & 0 & \ldots & \sigma_{\pi n}^2\end{bmatrix}. \tag{12}$$

Correspondingly,

$$q_\vartheta(\theta) \sim \mathcal{N}(\mu, \Sigma), \tag{13}$$

where

$$\mu = \begin{bmatrix}\mu_1\\ \mu_2\\ \cdots \\ \mu_n\end{bmatrix}, \Sigma = \begin{bmatrix}\sigma_1^2 & 0 & \ldots & 0\\ 0 & \sigma_2^2 & \ldots & 0\\ \ldots & \ldots & \ldots & \ldots\\ 0 & 0 & \ldots & \sigma_n^2\end{bmatrix}. \tag{14}$$

Further, $q_\vartheta(\theta)$ can be denotes as

$$q_\vartheta(\theta) = \frac{1}{\sqrt{(2\pi)^n|\Sigma|}}\exp[-\frac{1}{2}(\theta-\mu)^T\Sigma^{-1}(\theta-\mu)], \tag{15}$$

and $\pi(\theta)$ can be denotes as

$$\pi(\theta) = \frac{1}{\sqrt{(2\pi)^n|\Sigma_\pi|}}\exp[-\frac{1}{2}(\theta-\mu_\pi)^T\Sigma_\pi^{-1}(\theta-\mu_\pi)]. \tag{16}$$

Then, we can get

$$\begin{aligned}\mathrm{KL}\left(q_\vartheta(\theta)\|\pi(\theta)\right) &= \int q_\vartheta(\theta)\log\frac{q_\vartheta(\theta)}{\pi(\theta)}\mathrm{d}\theta \\ &= \int q_\vartheta(\theta)\log q_\vartheta(\theta)\mathrm{d}\theta \\ &\quad -\int q_\vartheta(\theta)\log\pi(\theta)\mathrm{d}\theta \\ &= \log\frac{1}{\sqrt{(2\pi)^n|\Sigma|}} - \log\frac{1}{\sqrt{(2\pi)^n|\Sigma_\pi|}} \\ &\quad -\frac{1}{2}\int q_\vartheta(\theta)(\theta-\mu)^T\Sigma^{-1}(\theta-\mu)\mathrm{d}\theta \\ &\quad +\frac{1}{2}\int q_\vartheta(\theta)(\theta-\mu_\pi)^T\Sigma_\pi^{-1}(\theta-\mu_\pi)\mathrm{d}\theta \\ &= \frac{1}{2}\log\frac{|\Sigma_\pi|}{|\Sigma|} - \frac{1}{2}\mathbb{E}(\theta-\mu)^T\Sigma^{-1}(\theta-\mu) \\ &\quad +\frac{1}{2}\mathbb{E}(\theta-\mu_\pi)^T\Sigma_\pi^{-1}(\theta-\mu_\pi)\end{aligned} \tag{17}$$

According to Theorem 1, we can simplify Eq (17) as

$$\begin{aligned}\mathrm{KL}\left(q_{\vartheta}(\theta)\|\pi(\theta)\right) &= \frac{1}{2}\log\frac{|\Sigma_\pi|}{|\Sigma|} - \frac{1}{2}\operatorname{tr}(\Sigma^{-1}\Sigma) + \frac{1}{2}\operatorname{tr}(\Sigma_\pi^{-1}\Sigma) \\ &\quad - (\mu-\mu)^T\Sigma^{-1}(\mu-\mu) \\ &\quad - (\mu-\mu_\pi)^T\Sigma_\pi^{-1}(\mu-\mu_\pi) \\ &= \frac{1}{2}\log\frac{|\Sigma_\pi|}{|\Sigma|} - \frac{1}{2}n + \frac{1}{2}\operatorname{tr}\left(\Sigma_\pi^{-1}\Sigma\right) \\ &\quad + \frac{1}{2}(\mu-\mu_\pi)^T\Sigma_\pi^{-1}(\mu-\mu_\pi) \\ &= \sum_{i=1}^{n}\left(\log\frac{\sigma_{\pi i}}{\sigma_i} + \frac{\sigma_i^2 + (\mu_i-\mu_{\pi i})^2}{2\sigma_{\pi i}^2} - \frac{1}{2}\right)\end{aligned} \tag{18}$$

The result is consistent with Eq (10).

In conclusion, we have analyzed the meanings of different terms in the objective function Eq (5). The first term Eq (7) is related to data. The second term Eq (18) is only related to $\vartheta = (\mu, \sigma)$ further θ like a regularization term. The third term $\log P(\mathcal{D})$ is a constant once the data $\mathcal{D}$ is determined.

Theorem 1. *Given $\mathbf{x} \sim \mathcal{N}(\mu, \Sigma)$, where $\mathbf{x}$ is a n dimensional vector, that is $\mathbf{x} \in \mathbb{R}^{n\times 1}$. If we assume A is a $n \times n$ matrix, we have*

$$\mathbb{E}(\mathbf{x}^T A\mathbf{x}) = \operatorname{tr}(A\Sigma) + \mu^T A\mu. \tag{19}$$

Proof. The trace of a matrix (denoted as "tr") is defined as the sum of the principle diagonal elements of a matrix, which has the following properties:

$$\begin{aligned}&\operatorname{tr}(M_1M_2) = \operatorname{tr}(M_2M_1) \\ \Rightarrow\ &\operatorname{tr}(M_1M_2M_3) = \operatorname{tr}(M_2M_3M_1) = \operatorname{tr}(M_3M_1M_2),\end{aligned} \tag{20}$$

where M_1, M_2, M_3 are three compatible matrices. Therefore, we have

$$\mathbf{x}^T A\mathbf{x} = \operatorname{tr}(\mathbf{x}^T A\mathbf{x}) = \operatorname{tr}(A\mathbf{x}\mathbf{x}^T). \tag{21}$$

According to the properties of n-dimensional diagonal Gaussian distribution, we have

$$\begin{aligned}\Sigma &= \mathbb{E}\Big((\mathbf{x}-\mu)(\mathbf{x}-\mu)^T\Big) \\ &= \mathbb{E}\Big((\mathbf{x}-\mu)(\mathbf{x}^T-\mu^T)\Big) \\ &= \mathbb{E}\Big(\mathbf{x}\mathbf{x}^T - \mathbf{x}\mu^T - \mu\mathbf{x}^T + \mu\mu^T\Big) \\ &= \mathbb{E}(\mathbf{x}\mathbf{x}^T) - \mathbb{E}(\mathbf{x})\mu^T - \mu\mathbb{E}(\mathbf{x}^T) + \mu\mu^T \\ &= \mathbb{E}(\mathbf{x}\mathbf{x}^T) - \mu\mu^T.\end{aligned} \tag{22}$$

This is equivalent to

$$\mathbb{E}(\mathbf{x}\mathbf{x}^T) = \Sigma + \mu\mu^T. \tag{23}$$

Therefore,

$$\begin{aligned}\mathbb{E}(\mathbf{x}^T A\mathbf{x}) = \mathbb{E}\Big(\operatorname{tr}(A\mathbf{x}\mathbf{x}^T)\Big) &= \operatorname{tr}\Big(\mathbb{E}(A\mathbf{x}\mathbf{x}^T)\Big) \\ &= \operatorname{tr}\Big(A\mathbb{E}(\mathbf{x}\mathbf{x}^T)\Big) \\ &= \operatorname{tr}\Big(A(\Sigma + \mu\mu^T)\Big) \\ &= \operatorname{tr}(A\Sigma) + \operatorname{tr}(A\mu\mu^T).\end{aligned} \tag{24}$$

Then, based on Eq (21),

$$\mathbb{E}(\mathbf{x}^T A\mathbf{x}) = \operatorname{tr}(A\Sigma) + \mu^T A\mu. \tag{25}$$

□

3 Experimental Setup

In this section, we introduce a detailed experimental setup, including hardware settings and algorithm parameters. All the experiments are conducted on an Ubuntu machine equipped with two Intel(R) Xeon(R) Silver 4214R CPUs @ 2.40GHz with 12 physical cores, and four NVIDIA GeForce RTX 3090 GPUs, armed with 24GB of GDDR6X memory. We train the BNN of ST-*n*FBST using the PyTorch package (v1.7.1). The descriptions and settings of related hyper-parameters are listed as follows:

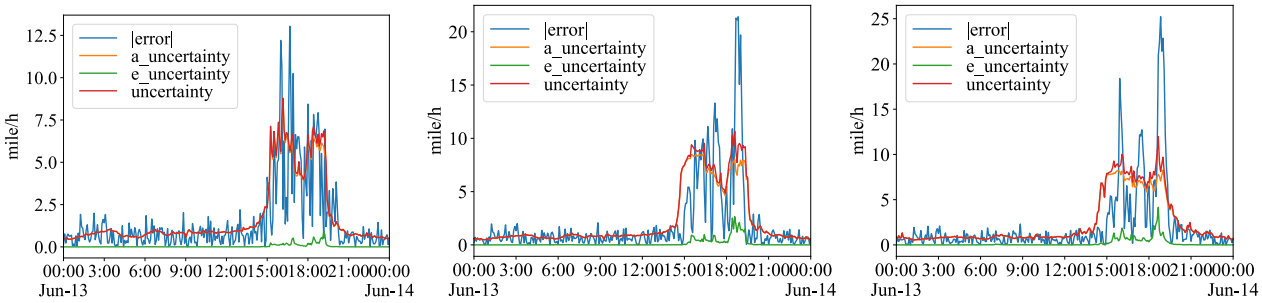
- lr. The initial learning rate of Adam optimizer. The default value is 0.005.
- lr_step_size. The period of learning rate decay. The default value is $[5, 20, 40, 70]$.
- lr_decay_ratio. The multiplicative factor of learning rate decay. The default value is 0.5.
- τ. The threshold to control the precision of $\hat{\sigma}_0$ obtained by the Sigma-Decoder. That is, $\hat{\sigma}_0 = \max(\hat{\sigma}_0, \tau)$. The default value is 0.01.
- μ_π. The mean of the prior distribution of θ in Eq (11). The default value is $\mathbf{0}$, which is a vector composed of 0.
- Σ_π. The (co)variance matrix of the prior distribution of θ in Eq (11). The default value is an identity matrix $\boldsymbol{I}$.
- max_diffusion_step. The max diffusion step defined in the diffusion convolution operation. The default value is 2.

Besides, as a core component of ST-*n*FBST, we need to sample through the Bayesian neural network and then approximate the posterior probability density by Kernel Density Estimation (KDE). The sample size we set is fifty. We use the Gaussian kernel function and choose the best bandwidth from $\{0.01, 0.05, 0.1, 1.0\}$ by GridSearchCV for each sampling through 5-fold cross-validation. The bandwidth here acts as a smoothing parameter, controlling the trade-off between bias and variance in the result. A large bandwidth leads to a very smooth (that is, high-bias) density distribution while a small bandwidth leads to an unsmooth (that is, high-variance) density distribution.

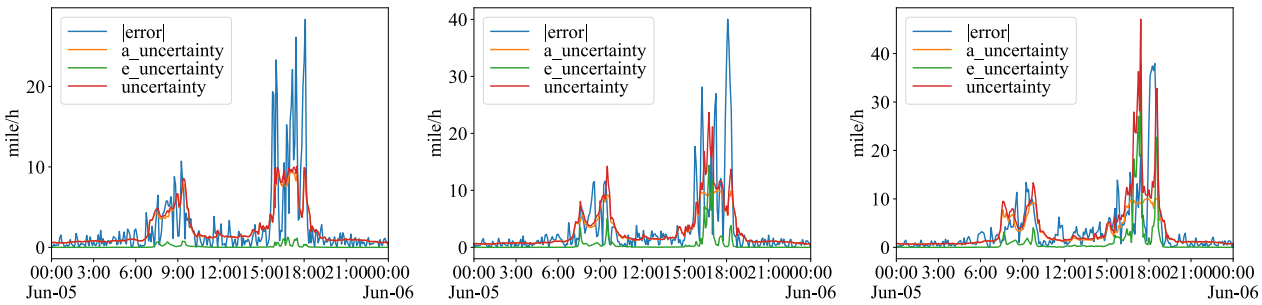
4 More Experimental Results on the PEMS-BAY Dataset

Traffic forecasting on the METR-LA (Los Angeles, which is known for its complicated traffic conditions) dataset is more challenging than that on the PEMS-BAY dataset. Thus we use METR-LA as the default dataset for analysis in the main paper, and more results on the PEMS-BAY dataset are provided here. The conclusions obtained from two datasets are consistent.

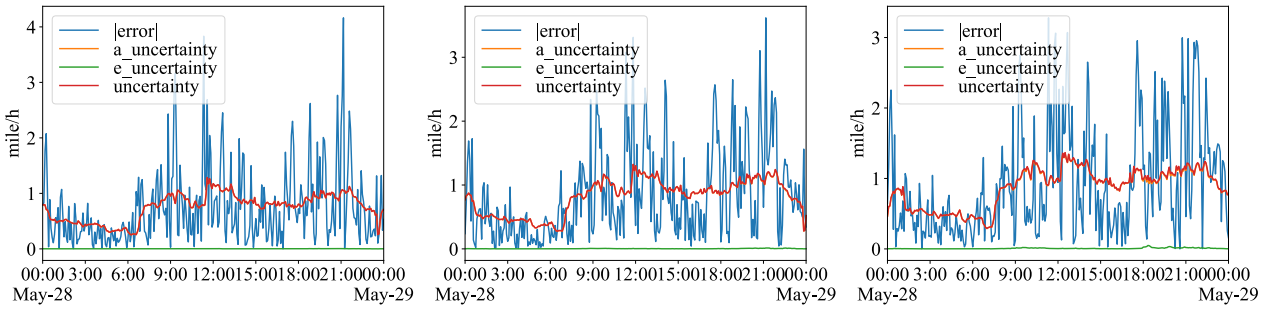
Figure 1 illustrates the trends of MAE and uncertainties under different prediction horizons. Sensor 3 and 6 are central sensors, while sensor 212 belongs to marginal sensors. We have the following observations:



(a) Predict 15min after for sensor 3 (b) Predict 30min after for sensor 3 (c) Predict 1h after for sensor 3



(d) Predict 15min after for sensor 6 (e) Predict 30min after for sensor 6 (f) Predict 1h after for sensor 6



(g) Predict 15min after for sensor 212 (h) Predict 30min after for sensor 212 (i) Predict 1h after for sensor 212

Figure 1: Comparison between uncertainties and |error| for different forecasting horizons on the PEMS-BAY dataset.

- The fluctuation of MAE closely aligns with the variations in uncertainty across diverse forecasting horizons. This implies that our method accurately captures uncertainties. Considering that we thoroughly leverage both uncertainties in the optimization process by incorporating them into the loss function, this further explains why our approach outperforms other state-of-the-art models.

- In most cases, the aleatoric uncertainty is much larger than the epistemic uncertainty, accounting for the majority of uncertainty. This implies that the traffic uncertainty is mainly data-related. Thanks to the powerful representation capability of neural networks, we can train models effectively to reduce epistemic uncertainty.

- There are some cases where error and uncertainties manifest inconsistently, specifically, with low error but high uncertainties. Moreover, epistemic uncertainty is high in such cases. This occurs when changes in road conditions at intersections, such as traffic congestion or unexpected accidents, result in a sharp decrease in the speed of other sensors. This implies that uncertainty, compared to error, is more effective in monitoring changes in road conditions. A more detailed case study is presented in Section 5.

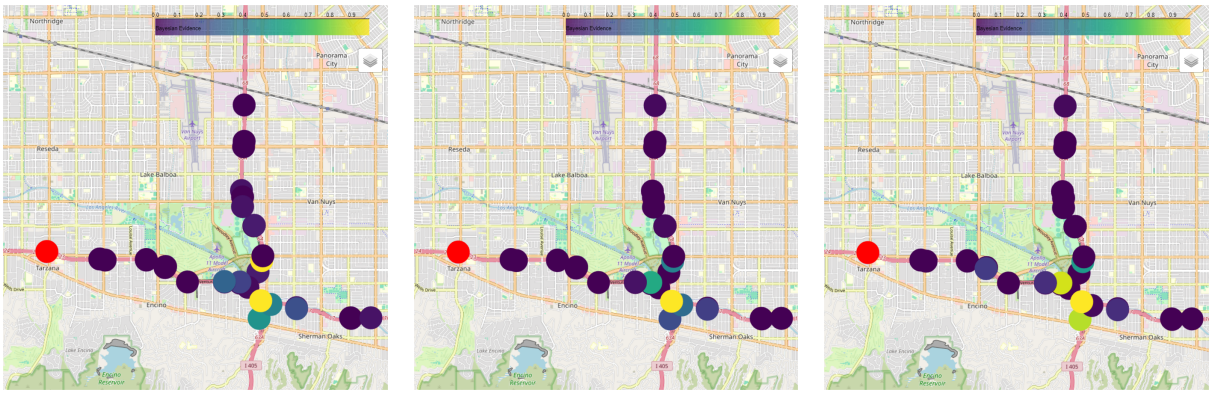
- As the forecasting horizon expands, both error and uncertainties of central sensors increase correspondingly, while those of marginal sensors remain relatively stable. However, the predictions of marginal sensors fluctuate significantly at different times. This aligns with the definition of two types of sensors. The marginal sensors exhibit fewer impacts from other sensors while utilizing less information from other sensors. However, the central sensors are more susceptible to the influence exerted by other sensors, rendering predictions more intricate.

5 Case Study

In this section, we provide a case study to show the conclusion obtained by significance testing spatially about the influence of intersections. Furthermore, we analyze the reasons why error is inconsistent with uncertainty in some cases.

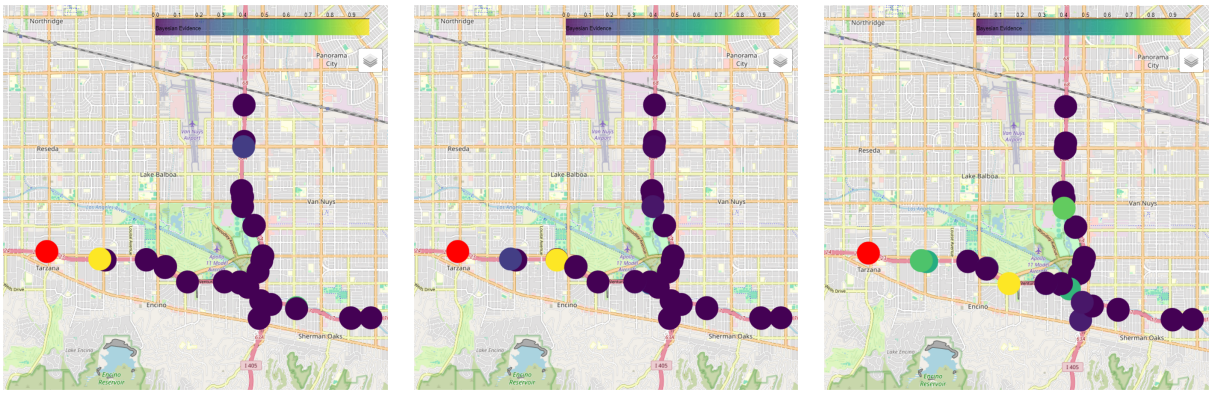
Figure 2,3,4,5 show the testing results for different sensors under various error and uncertainty levels. They all represent the testing results for 15 minutes forecasting horizon, representing short-term forecasting. Figure 2 and 4 correctly identify that sensors near intersections are insignificant, hence the low error and uncertainty. On the other hand, the high error and uncertainty in Figure 3 and 5 is attributed to their determination that surrounding sensors are insignificant, but the sensors at intersections are significant. By comparing them, it can be observed that short-term traffic forecasting is more concerned with the road conditions where their own sensors are located, but do not pay attention to the situation at intersections. The conclusion is verified by the real dataset. For example, the true speed in Figure 5c is 58.63 miles/h, while the prediction is 19.97 miles/h. The prediction is significantly lower than the true speed. This discrepancy arises from the fact that, within a very short time frame in the recent past, although the true speeds at the intersections are indeed slow, the speeds of sensors on the same road as sensor 149 remain high. However, the testing results indicate that information from the intersection is utilized at this time, leading to high errors and increased uncertainty.

Subsequently, we conduct the significance testing for the same sensor at the same moment as Figure 3c, with the forecasting horizon changed to 1h, representing long-term forecasting. As depicted in Figure 6, the error significantly decreased, but uncertainty remained high. By observing the speeds of different sensors, we found that those of sensors on the same road as the interest sensor are high, while the speeds at intersections are considerably slow. We conclude that the changes at intersections take time to propagate to other road segments. Therefore, short-term forecasting should not focus on them, but they have a significant impact on long-term forecasting. Due to a sharp transition from high to low prediction speed and the testing of a significant impact at the intersection, the uncertainty of predictions in Figure 6 remains high. This also confirms that uncertainty, compared to error, is more effective in monitoring changes in road conditions.



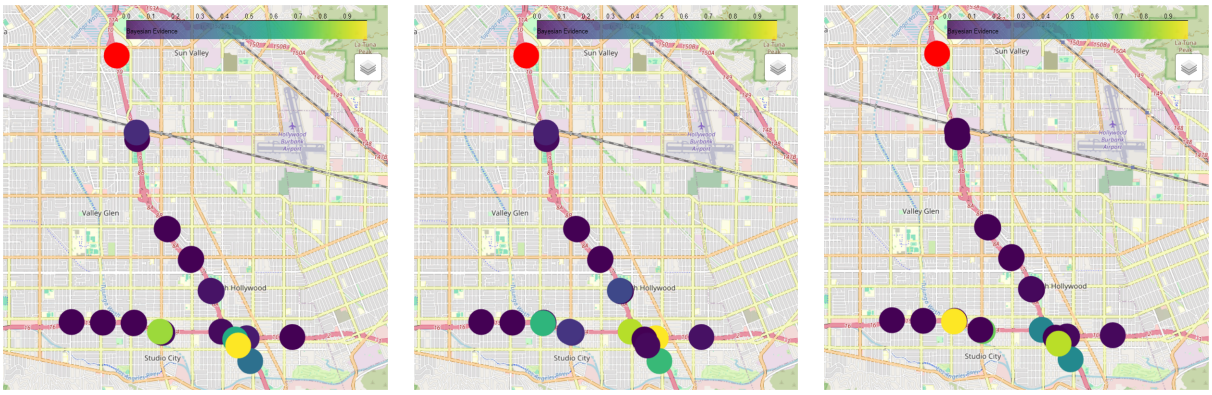
(a) (b) (c)

Figure 2: Spatial testing results for sensor 181 (highlighted in red) under low uncertainty and low error at different times.



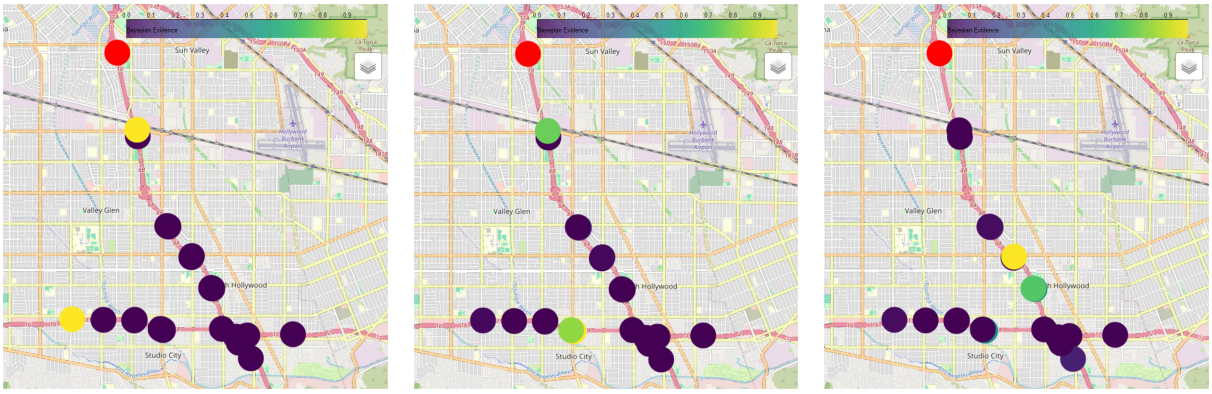
(a) (b) (c)

Figure 3: Spatial testing results for sensor 181 (highlighted in red) under high uncertainty and high error at different times.



(a) (b) (c)

Figure 4: Spatial testing results for sensor 149 (highlighted in red) under low uncertainty and low error at different times.



(a) (b) (c)

Figure 5: Spatial testing results for sensor 149 (highlighted in red) under high uncertainty and high error at different times.

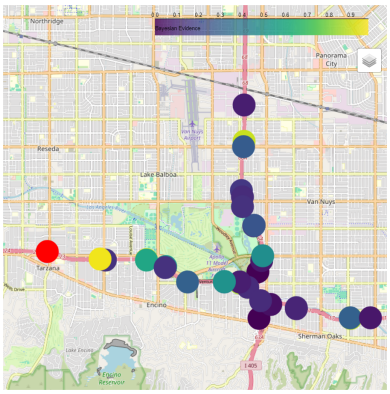


Figure 6: Spatial testing results for sensor 181 under low error but high uncertainty level for 1h forecasting horizon.