

Limits of Probabilistic Safety Guarantees when Considering Human Uncertainty

Anonymous Author(s)

Affiliation

Address

email

Abstract: In robotic applications involving interaction with humans, such as autonomous driving, explicit safety guarantees are crucial in order to avoid potentially life-threatening accidents. Many data-driven methods have explored learning probabilistic bounds over human agents' trajectories (i.e. confidence tubes that contain trajectories with probability δ), which can then be used to guarantee safety with probability $1 - \delta$. However, almost all existing works consider $\delta \geq 0.001$. The purpose of this paper is to argue that (1) in safety-critical applications, it is necessary to provide safety guarantees with $\delta < 10^{-8}$, and (2) learning-based methods are ill-equipped to compute *accurate* confidence bounds at such low δ . Using two human driving datasets (*highD* and *inD*), as well as synthetically generated data, we show that current uncertainty models utilize inaccurate distributional assumptions in describing human behavior and/or require infeasible amounts of data to accurately learn confidence bounds for $\delta \leq 10^{-8}$. Collectively, these two issues results in highly unreliable confidence bounds, which can have dangerous implications if deployed on safety-critical systems. We conclude by suggesting that potential solutions incorporate prior knowledge of human interaction behavior, encoded through assume-guarantee contracts.

Keywords: Probabilistic Safety, Uncertainty Modeling, Safety-Critical Planning

1 Introduction

Autonomous robots will be increasingly deployed in more unstructured human environments (e.g. roads and malls) where they must safely carry out tasks in the presence of other human agents. The cost of failure is high in these environments, as safety violations can be life-threatening. At present, safety is often enforced by learning an uncertainty distribution or confidence bounds over the future trajectory of other agents, and designing a controller that is robust to such uncertainty [1]. Based on these learned trajectory distributions, probabilistic safety guarantees can be provided at a specified safety threshold δ over a given planning horizon (e.g. by enforcing chance constraints such that $\mathbb{P}(\text{collision}) \leq \delta$) [2, 3, 4, 5]. However, for such guarantees to hold, it is critical that we *accurately predict* the uncertainty over other agents' future trajectories with high probability $1 - \delta$.

Current works that provide probabilistic safety guarantees for autonomous navigation in uncertain, human environments consider safety thresholds in the range $\delta \geq 0.001$. While such guarantees are important, safety critical robotic applications require δ that are orders of magnitude smaller [6].

Suppose a robot/car is guaranteed safe with probability δ across every 10s planning horizon. Given a safety threshold $\delta \approx 0.001$, we could expect a safety violation every 3 hrs. For reference, based on NHTSA data [7], human drivers have an effective safety threshold $\delta \leq 10^{-7}$.

It is clear then that for safety-critical robotic applications, we must strive for extremely low safety thresholds, on the order $\delta \leq 10^{-8}$. However, this paper argues that current learning-based approaches that model human trajectory uncertainty (a) rely on highly inaccurate distribution assumptions, invalidating resulting safety guarantees, and/or (b) can not adequately extend to safety-critical situations. To illustrate this, we applied different uncertainty models (see Table 1) to datasets of human driving (the *highD* and *inD* datasets [8, 9]). We found that *even under extremely generous*

assumptions, learned models are highly inaccurate in capturing human behavior at low δ and often mispredict the probability of rare events by several orders of magnitude. Furthermore, we show that increasing dataset sizes will not adequately improve the accuracy of learned uncertainty models.

It is critical we emphasize two points regarding our results:

- We focus *not* on robust control algorithms, but rather on the learned uncertainties they leverage.
- We focus on in-distribution uncertainty, rather than out-of-distribution uncertainty. I.e., we highlight the fundamental inability of uncertainty models (see Table 1) to accurately capture distributions at very low values of δ , regardless of generalization ability or the learning algorithm used.

Our results highlight the danger in utilizing learned models of human uncertainty in safety-critical applications. Fundamental limitations prevent us from accurately learning the probability of rare trajectories with finite data, and using inaccurate confidence bounds can result in unexpected collisions. While this paper focuses on illustrating a significant problem (rather than providing a solution), we conclude by advocating for an alternate approach utilizing assume-guarantee contracts [10]; we believe these tools can address the limitations pointed out in this work by rigorously combining (a) learned patterns of behavior and (b) prior knowledge encoding human interaction rules.

Section II reviews the literature on human uncertainty modeling for safe motion planning. In Section III, we highlight issues with these models for safety-critical applications. In Section IV, we conclude by proposing handling human uncertainty through the lens of assume-guarantee contracts.

Uncertainty Model Class	Example Works	Min. Safety Threshold
Gaussian Process	[11, 2, 12, 13]	$\delta \geq 0.001$
Dynamics w/ Gaussian Noise	[14, 15, 16]	$\delta \geq 0.001$
Bayesian NN	[17, 18, 19]	$\delta \geq 0.05$
Noisy Rational Model	[3]	$\delta \geq 0.01$
Hidden Markov Model / Markov Chain	[20, 21]	$\delta \geq 0.01$
Quantile Regression	[5]	$\delta \geq 0.05$
Scenario Optimization	[22, 23, 24]	$\delta \geq 0.01$
Generative Models (e.g. GANs)	[25, 26, 27]	N/A

Table 1: Different model classes for capturing human trajectory uncertainty, used in safe planning algorithms to guarantee safety with probability $1 - \delta$. The right column shows the lowest safety threshold, δ , we could find used in the literature (in simulation or hardware experiments) for each model class. There is no entry for generative models, as these models have not yet been utilized to provide *explicit* safety guarantees during planning, though there is surely a trend in this direction.

2 Related Work

Most recent approaches for guaranteed safe navigation in proximity to humans or their cars approximate uncertainty in human trajectories as a random process (i.e. deviations from a nominal trajectory are drawn i.i.d. from a fixed or varying distribution). These uncertainty models help capture noise and the effects of latent variables (e.g. intention), and enable probabilistic safety guarantees in uncertain, dynamic environments. Most models fall into one or more of the following categories:

- **Gaussian Process (GP):** These approaches model other agents' trajectories as Gaussian processes, which treat trajectory uncertainty as a multivariate Gaussian [15, 28, 2, 12, 13]. There are several extensions, such as the IGP model [29] (which accounts for interaction between multiple agents), or others [30, 31]. However, they all treat uncertainty as a multivariate Gaussian.
- **Gaussian Noise with Dynamics Model:** These approaches use a dynamics model with additive Gaussian noise; noise can also be added in state observations. This induces a Gaussian distribution over other agents' future trajectory (or a situation where we can do moment-matching) [32, 16].
- **Quantile Regression:** This approach computes quantile bounds over the trajectories of other agents at a given confidence level, δ . This approach is beneficial in that it does not assume an uncertainty distribution over trajectories [33, 5].
- **Scenario Optimization:** This approach computes a predicted set over other agents' actions based on samples of previously observed scenarios [34]. This is a distribution-free approach (i.e. does

not assume a parametric uncertainty distribution) [35, 22, 23, 24]. [36, 37] do not use scenario optimization, but their work based on computing minimum support sets follows a similar flavor.

- **Noisy Rational (i.e. Boltzmann Rational) model:** This model treats the human as a rational actor who takes “noisily optimal” actions according to a distribution in the exponential family, shown in Equation (5). The uncertainty in the action is captured by this distribution, which relies on an accurate model of the human’s value function [38, 39, 1, 3, 40].
- **Generative Models (e.g. CVAE, GAN):** These models learn an implicit distribution over trajectories. They do not provide an explicit distribution, but can generate random trajectories that attempt to approximate the true distribution [25, 26, 27].
- **Hidden Markov Model (HMM) / Markov Chain:** These models differ in that they capture uncertainty over *discrete* sets of states/intentions (e.g. goal positions) – as opposed to capturing uncertainty over trajectories. Thus, the objective is to infer the other agents’ unobserved state/intention (from a discrete set) with very high certainty, $1 - \delta$ [41, 42, 43, 44, 45, 20, 21].
- **Uncertainty Quantifying (UQ) Neural Networks:** These approaches do not constitute a separate class of uncertainty models, but simply refer to methods that train a neural network to capture the distribution over other agents’ trajectories [46, 17, 19]. We list them separately due to their popularity. Most often these networks output a Gaussian distribution or mixture of Gaussians (e.g. Bayesian neural networks [47], deep ensembles [48], Monte-Carlo dropout [49]). These models can also quantify uncertainty over discrete states (i.e. infer the hidden state in HMMs) [50, 51].

Once a predicted trajectory and its uncertainty is learned, many mechanisms exist to guarantee safety (e.g. incorporating uncertainty into chance constraints). In this work, we are *not* focused on mechanisms for guaranteeing safety, but rather on the issue of learning/modeling trajectory uncertainty.

3 Issues with Uncertainty Models

This section illustrates the limitations of probabilistic models of uncertainty when considering human behavior. We show that prevalent classes of uncertainty (see Table 1) fail to capture human behavior at adequate safety thresholds ($\delta \leq 10^{-8}$), and exhibit significant errors when evaluated against real-world data. Since safe planning algorithms *assume their computed uncertainty distributions are accurate*, significant errors invalidate any claimed safety guarantees.

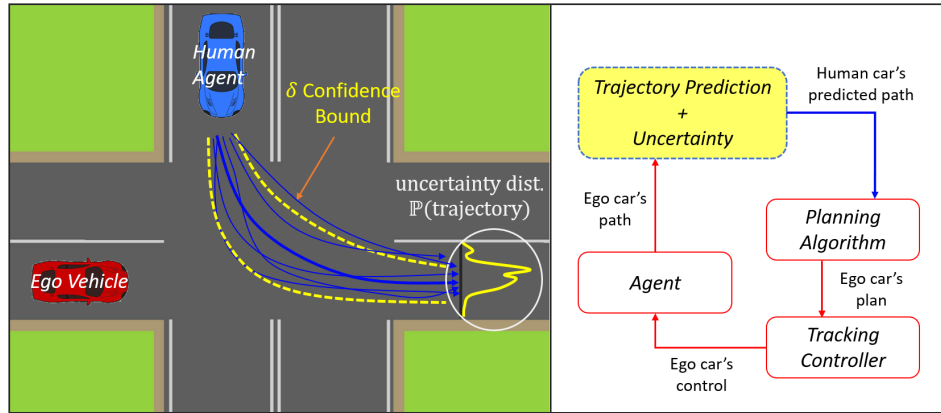


Figure 1: **(Left)** In this example the red car must take into account the blue car’s trajectory – and its uncertainty – in its plan to progress safely through the intersection. The dashed yellow curves denote the boundary of a tube that defines the δ confidence bound over trajectories. The white circle depicts a distribution over trajectories. The blue lines are example trajectories. **(Right)** Simplified illustration of different stages of the control pipeline. While every stage (prediction, planning, tracking) is crucial to guaranteeing safety, this paper focuses exclusively on the yellow box, *prediction*.

We highlight these limitations by testing prevalent modeling assumptions on real-world driving data from two driving datasets: (a) the highD dataset [8] capturing vehicles driving on German highways, and (b) the inD dataset [9] capturing vehicles driving within the city at an intersection. We extract all trajectories of length 10 seconds, $\tau_{[0,10]}$, in these datasets, and split the trajectories into a training set, $\tau_{[0,10]}^{(train)} \in \mathcal{D}^{train}$, and test set, $\tau_{[0,10]}^{(test)} \in \mathcal{D}^{test}$. For every test trajectory, $\tau_{[0,10]}^{(test)} \in \mathcal{D}^{test}$, we

collect all trajectories in the training set in equivalent scenarios,

$$\mathcal{M}(\tau^{(test)}) = \left\{ \tau_{[0:10]} \in \mathcal{D}^{train} \mid \|\tau_{[0:2]} - \tau_{[0:2]}^{(test)}\|_{\infty} < \epsilon \right\}. \quad (1)$$

We define *equivalent scenarios* as the set of trajectories that are ϵ -close ($\epsilon = 3\text{ft}$) over their first $2s$, such that $\mathcal{M}(\tau)$ denotes the set of equivalent scenarios to trajectory τ in the training set, \mathcal{D}^{test} . We chose a past observation horizon of $2s$ following [51], but found that the trends we observed did not change considerably when using $1s$ or $3s$ for the past observation horizon. For every test trajectory, $\tau^{(test)}$, we fit optimal parameters for our uncertainty model (e.g. Gaussian) to the equivalent training scenarios, $\mathcal{M}(\tau^{(test)})$, and observe where the test trajectory falls with respect to the computed distribution or bounds. By iterating through all trajectories in \mathcal{D}^{test} , we can compute statistics analyzing how well the test trajectories fit to models predicted from our training trajectories.

Intuition in equations: To clarify our method and make clear our assumptions, we outline our approach in terms of different distribution errors. Let us define an agent’s state x by its past history, $\tau_{[0:2]}$, and static environmental context, c , such that $x = (\tau_{[0:2]}, c)$. We then define its action, a , as its future trajectory $a = \tau_{[2:10]}$. Given that the future trajectory is drawn from some uncertain distribution, $a \sim \mathcal{A}(x)$, our goal is to learn a model $\hat{F}(x)$ that accurately approximates this distribution over trajectories, $\mathcal{A}(x)$, minimizing the following error,

$$L^{out} = \mathbb{E}[m(\mathcal{A}(x) \parallel \hat{F}(x))] , \quad (2)$$

where m defines some metric over probability distributions (e.g. total variation distance). The model \hat{F} is trained on data from our training set \mathcal{D}_{train} . Since we don’t have access to the true distribution, we can approximate the expectation in (2) using the test set, \mathcal{D}_{test} , yielding the error functions,

$$\begin{aligned} L^{unseen} &= \frac{1}{N_{unseen}} \sum_{x \in \mathcal{D}^{test}} [m(\hat{\mathcal{A}}(x) \parallel \hat{F}(x))] , \\ L^{seen} &= \frac{1}{N_{seen}} \sum_{x \in \mathcal{D}^{test} \cap \mathcal{D}^{train}} [m(\hat{\mathcal{A}}(x) \parallel \hat{F}(x))] , \end{aligned} \quad (3)$$

where $\hat{\mathcal{A}}(x)$ represents the approximation of $\mathcal{A}(x)$ based on \mathcal{D}_{test} , and N_{unseen}, N_{seen} are normalizing factors denoting the number of trajectories being considered. L^{unseen} can be interpreted as the test error, capturing how well the model \hat{F} captures the action distribution $\hat{\mathcal{A}}$ from states it may not have trained on. On the other hand, L^{seen} captures how well \hat{F} captures the action distribution, $\hat{\mathcal{A}}$, from states it *has* trained on. In general, the relationship between these errors follows,

$$L^{out} \underset{\substack{\geq \\ \text{distribution gap}}}{\geq} L^{unseen} \underset{\substack{\geq \\ \text{generalization gap}}}{\geq} L^{seen}. \quad (4)$$

In our analysis, we focus on L^{seen} . As this ignores any generalization or distribution gap, it benchmarks the *best potential performance* of each model class. The distribution gap quantifies how the change from the true trajectory distribution to the test distribution \mathcal{D}^{test} affects model accuracy. The generalization gap quantifies how out-of-distribution test examples affect the model accuracy.

Accounting for multi-agent interactions: A valid criticism of simply looking at the distributional error $m(\hat{\mathcal{A}}(x) \parallel F(x))$ is that our distribution only depends on each individual agent’s state, and not the multi-agent context, which clearly influences the trajectory distribution (e.g. the blue car impacts the red car’s trajectory in Fig. 1). However, the influence of other agents is typically reflected through the endpoint of our trajectory, τ (e.g. the influence of an agent cutting into our lane is reflected in our rapid slowdown). Therefore, we can account for multi-agent interactions by assuming that an oracle gives us *the exact endpoint of every trajectory* $\tau_{[10]}$. While this is a generous assumption, it allows us to conservatively deal with the influence of other agents, under the assumption that the influence of other agents is reflected in the endpoint of our trajectory, $\tau_{[10]}$.

Summarizing, we consider (a) there is no distribution gap, (b) there is no generalization gap, and (c) we are given the end point of every trajectory (perfectly inferring the agent’s goal and the influence of other agents). Note that our objective is to illustrate limitations of learned probabilistic models, even under ideal conditions. Therefore, our strong assumptions (though unrealistic) help us reason about the best-case scenario for each model class, providing an upper-bound on performance.

If the models perform poorly under these extremely generous assumptions, we can not expect reasonable performance in realistic settings.

3.1 Gaussian Uncertainty Models

We start by analyzing the popular Gaussian uncertainty model, used in many UQ neural networks [19], Gaussian process models [2], and robust regression [31, 4]. These approaches model the data and its uncertainty with a Gaussian distribution (see top 3 rows in Table 1).

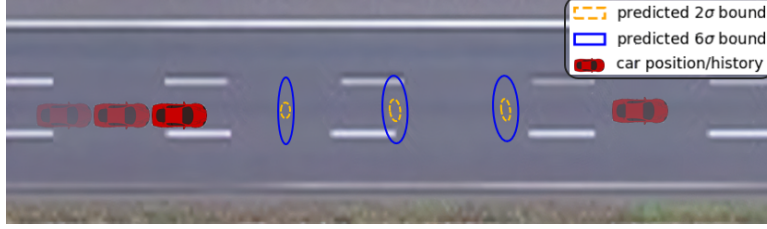


Figure 2: Plot of the 2σ and 6σ confidence bounds on the car’s future trajectory at 2, 4, and 6 seconds in the future, given that the position 8 seconds in the future is known. This is computed based on the data from similar trajectories (our trajectory is denoted by the red cars).

Using the procedure outlined at the beginning of Section 3, we compute the best-fit Gaussian distribution, \hat{F} , over the training trajectories \mathcal{D}^{train} , and observe how well it captures the in-distribution test trajectories in \mathcal{D}^{test} (i.e. how well it minimizes L^{seen}). Figure 3a shows the ratio of observed to expected violations in the test set at each safety threshold, δ . A violation is defined when the test trajectory lies outside the δ -uncertainty bound for a specified δ . If the data followed a perfect Gaussian distribution, each curve in Fig. 3a would track the dotted black line (i.e. keep a ratio near 1). While the Gaussian model might be valid for $\delta \geq 0.01$, it is highly inaccurate outside this range. Importantly it is overly risky for $\delta < 0.001$, posing a problem for safety-critical applications.

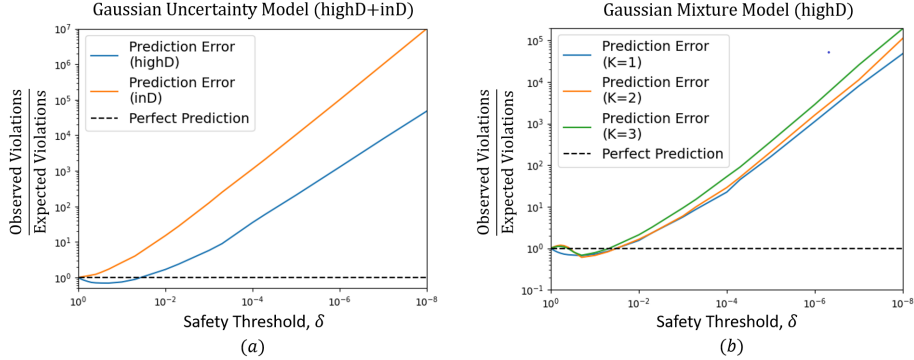


Figure 3: (a) Prediction error vs. safety threshold, δ , using a Gaussian uncertainty model on the highD and inD datasets. (b) Prediction error using a GMM with different number of mixtures, K , on the highD dataset. In both plots, the dashed black line represents a perfect prediction model.

In addition to the issue of inaccurate distributional assumptions, the confidence bounds that arise when dealing with $\delta \approx 10^{-8}$ become quite large, making planning around these bounds difficult or potentially infeasible. Figure 2 shows the 2σ and 6σ ellipsoids projecting the position of the car (red points) forward in time, based on the training dataset. Note that the 6σ ellipsoid (corresponding to $\delta \approx 10^{-9}$) encroaches on each lane, making it difficult for other cars to drive alongside it.

Note on Gaussian mixture models (GMM): One might point out that this problem could be alleviated by considering GMMs over a discrete set of goals (e.g. left versus right turn). For example, interacting Gaussian processes (IGP) leverage this tool to alleviate the freezing robot problem [29]. However, recall that our results assume that an oracle gives us the endpoint/goal of every trajectory; this should account for significant improvement that could be obtained from utilizing GMMs or goal inference. We confirm this by training GMMs on the same data with different number of mixtures ($K = 1, \dots, 4$), and noting that prediction performance on the test data did not improve (see Fig. 3b). Thus, our results illustrate the limitations of any Gaussian-based uncertainty model (IGP, GMM, etc.), by highlighting that human behavioral variation is inherently non-Gaussian.

To further emphasize fragility of the Gaussian model at low δ , we generated synthetic 2D data following 3 different, known distributions. We then examined how well the best fit Gaussian predicted violations at a given δ . Even with perfectly i.i.d. training/test data, the error at low δ was significant (off by an order-of-magnitude) for non-Gaussian distributions. See Appendix A for details.

3.2 Noisy Rational Model

The noisy rational model considers that humans behave approximately optimally with respect to some reward function. It has enabled significant progress in inverse reinforcement learning (IRL) by allowing researchers to learn reward functions from human data [39], and compute explicit uncertainty intervals over human agents' actions [3]. However, the noisy rational model adopts an underlying model of uncertainty in the exponential family, often simplifying to Gaussian:

$$\mathbb{P}(x_{t+1} | \beta) = \frac{e^{\beta Q_H(x_{t+1})}}{\sum_{\tilde{x}_{t+1}} e^{\beta Q_H(\tilde{x}_{t+1})}}. \quad (5)$$

In our driving scenario, the optimal model simplifies to the Gaussian distribution, since $Q_H = \|x_{t+1} - \hat{x}_{t+1}\|_\Sigma$ for some Σ (i.e. we want to best fit the data). As a result, the issues illustrated in Figures 2 and 3 are exactly faced by the noisy rational model (i.e. the shape of the underlying distribution does not match the assumed distribution). Thus, even in the best case – known goal, optimal data fit, no generalization gap – we conclude that these models are also ill-equipped to provide safety guarantees for safety-critical systems (e.g. $\delta < 10^{-8}$).

3.3 Quantile Regression

Quantile regression is an appealing alternative as it does not require strong assumptions on the underlying uncertainty distribution [5]. It is only concerned with computing tubes such that $1 - \delta$ proportion of trajectories are within that tube and δ are outside. To demonstrate its performance, we revisit the driving datasets and compute quantile bounds for each trajectory in the test set, using the equivalent scenarios from our training set. These quantile bounds are approximated as the smallest convex tube containing $1 - \delta$ proportion of trajectories, which optimizes the expected mutual information between the state, x , and action, a [52].

Figure 4 shows the ratio of the observed to expected number of test trajectories outside each quantile at safety threshold δ . The quantile regression model performs much better than the Gaussian model, and is quite accurate for $\delta > 0.01$. However, performance rapidly deteriorates as δ decreases, making estimated confidence bounds meaningless, since they fail to predict violation probabilities.

This makes sense as obtaining accurate quantile bounds at the δ -confidence level relies on splitting the data: δ percent of points should be outside the quantile bound with the rest inside those bounds. However, little (if any) data is available outside the quantile bound for very low δ . Put differently, to *observe* a one-in-a-million event, we would need to see a million trajectories. To *reliably predict* those events, we would need many more trajectories.

Given the availability of increasingly large robotics datasets, it is important to ask if we could reach good accuracy at desired safety thresholds, δ , by using more data. To answer this, we define the smallest accurate safety threshold, δ_{min} , as follows,

$$\delta_{min} = \min \delta \quad \text{such that} \quad \left| \log \left(\frac{\text{expected}(\delta)}{\text{observed}(\delta)} \right) \right| \leq \varepsilon. \quad (6)$$

We set $\varepsilon = 0.5$, where ε represents the vertical distance between each curve in Fig 4 and the dotted black line. Thus, δ_{min} represents the smallest δ such that our computed quantile bounds are ε -accurate. Note that δ_{min} is computed with respect to a given set of data. Therefore, by varying the

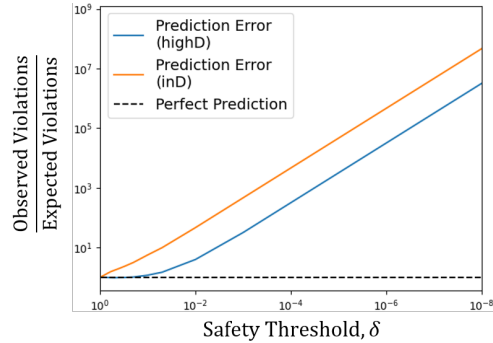


Figure 4: Prediction error vs. safety threshold, δ under computed quantile bounds on the highD and inD datasets. The dashed black line represents a perfect prediction model.

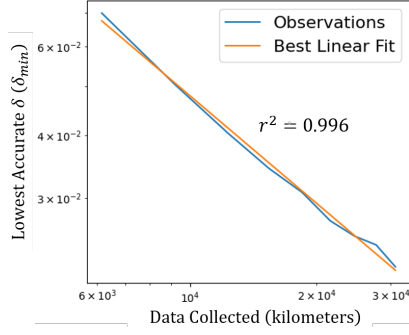


Figure 5: Smallest accurate δ versus amount of data collected (in km) – for the highD dataset. The trend is highly linear ($r^2 = 0.996$). The linear trend is also seen in results using the inD dataset.

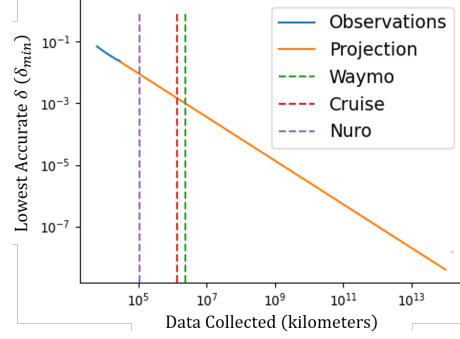


Figure 6: Projection showing the expected amount of data required (in km) to obtain an accurate safety threshold δ . The dashed lines show the number of kilometers driven in California in 2019 by Waymo, Cruise, and Nuro.

amount of data we provide for quantile regression, we can obtain Fig. 5, which captures how δ_{min} varies with the amount of data. Fig. 5 shows a surprisingly strong linear trend ($r^2 = 0.996$), which held across different datasets. This scaling is consistent with the lower bound on sample complexity derived in [53], suggesting that the quantile regression results shown are close to optimal (w.r.t. sample complexity). This lower bound is shown in Equation (7) and discussed further below.

While initially promising, if we project this linear trend down to $\delta_{min} \approx 10^{-8}$, we find that the amount of data required to reach safety-critical thresholds is far from feasible. Figure 6 shows that we would need more than 10 trillion kilometers of driving data to achieve accurate quantile bounds, just on certain highways and under our extremely generous assumptions. For reference, in 2018, approximately 5 trillion kilometers were driven in total across all cars/trucks in the U.S. [7].

Even assuming perfect generalization from one driving scenario to another, we would need almost a million times more driving data than the to-date cumulative driving kilometers of Waymo.

We conducted the same analysis on synthetic 2D data, and found the same trends seen in Fig. 4 (high accuracy until a given δ), and Fig. 5 (inverse relation $\delta_{min} \propto \frac{1}{N}$). See Appendix B for details.

Quantile Regression as a Fundamental Limitation: One might be tempted to conclude from Figure 6 that we should look for alternative methods (to quantile regression) that have better data efficiency (i.e. allow us to accurately infer confidence bounds at smaller δ with less data). Note that all methods attempting to provide confidence bounds over trajectories at a specified safety threshold δ (e.g. quantile regression) can be fundamentally viewed as classification problems; we must classify $1 - \delta$ trajectories within some learned bounds, with the rest outside those bounds. By viewing this as a classification problem, we can leverage results from VC-analysis that lower bound the data required, N , to reach a given prediction confidence [53]: To guarantee $Pr(\text{error}) \leq \delta$, we require

$$N(\delta, M) = \Omega\left(\frac{1}{\delta} \ln\left(\frac{1}{\delta}\right) + \frac{\text{VCdim}(M)}{\delta}\right) \quad (7)$$

where $\text{VCdim}(M)$ is the VC dimension of the utilized model M (see [53] for details/proof). The linear trend in Figures 5 and 6 (showing $N(\delta) \propto \frac{1}{\delta_{min}}$) fits very nicely with the lower bound (7), given that the second term dominates the first (i.e. we have large VC dimension). Note that if the first term dominated the second term, we would expect *worse* data scaling. This suggests that alternative methods will fail to provide confidence bounds with better data scaling than shown in Figure 6.

3.4 Other Uncertainty Models

Due to space constraints, we leave discussion of remaining models to Appendix C, including: (1) generative models (e.g. CVAE, GAN), (2) scenario optimization, and (3) hidden Markov models (HMM). However, below, we briefly describe the fundamental problems these models face:

- Scenario optimization – similar to quantile regression – requires far too much data to be feasible for small δ . Even with 40,000 trajectories in equivalent scenarios, we only reach $\delta \approx 10^{-4}$.

- Generative models *implicitly* learn the distribution $\mathcal{A}(x) = p(a|x)$. However, it has been shown – empirically and theoretically – that they can fail to learn the true distribution, even when their training objective nears optimality [54]. Furthermore, using state-of-the-art models [25, 26, 27], it would require *at least* a day to generate the trajectories necessary to certify safety with $\delta \approx 10^{-8}$.
- HMMs are distinct in that they learn probabilities over discrete states (e.g. goal positions). However, we show that even with a known observation function, $\mathbb{P}(\text{observation}|\text{state})$, it is highly unlikely to obtain sufficient confidence ($\delta \approx 10^{-8}$) of being in a given state (see Appendix C.3).

Note on UQ Neural Networks: We have not discussed UQ neural networks, because neural networks do not compose a distinct class of uncertainty models. Instead, they only provide a functional representation of the uncertainty in a given class (e.g. UQ neural networks typically output a Gaussian distribution or quantile bounds). The results in this section highlight *best-case* performance bounds for each class of model uncertainties, given optimal fit to the data. Therefore, any attempt to use neural networks to parameterize the model uncertainty will only yield worse performance.

Code for all results in the paper and Appendix can be found in the supplementary materials.

4 Conclusion / Future Work

Our main message is that *even under extremely generous assumptions*, models of human uncertainty are unable to extend safety guarantees to confidence levels, e.g. $\delta < 10^{-8}$, needed for safety-critical applications. Learned uncertainty distributions become highly inaccurate at low δ , producing unreliable confidence bounds, which could have dangerous consequences.

Significant inaccuracies in learned uncertainty distributions or bounds undermine any claimed guarantees of safety. Any safety thresholds, δ , lose their meaning.

There is a fundamental limitation to modeling human uncertainty purely as a random process. Data-driven methods (i.e. machine learning) are designed to capture prominent patterns in data and predict *likely* events; they are not suited to predict rare events. Intuitively, we need at least a million samples to *observe* a one-in-a-million event, and we need many more samples to *reliably predict* those events (even without considering generalizability). While it is theoretically possible that huge datasets could eventually enable accurate prediction of rare events, we have shown that such amounts of data are far from feasible in the near future (even ignoring generalization issues and computational cost).

Human behavior uncertainty vs. sensor-based uncertainties: Even if a system must be certified safe with $\delta = 10^{-8}$, it is uncommon to require any single module to have a failure probability less than 10^{-8} . Instead, redundancy with multiple (less reliable but independent) modules can help certify system safety. For example, in a robotic system, two different modules (one using LIDAR and one using stereo cameras) might predict an obstacle’s current position, each with confidence $1 - 10^{-4}$. Then the overall system can reason about the obstacle’s current position with confidence $1 - 10^{-8}$. However, the key is that *the modules must be independent*. While this may be a fair assumption for sensing uncertainty, it is not fair for human behavior prediction. No sensor exists to measure a human agent’s *future* position, and it is unclear how to enforce independence of multiple predictive modules. However, this is an interesting avenue of research: to simultaneously learn multiple predictors [55], *while enforcing and verifying independence between their predictions*.

Future Work: A promising solution to guarantee safety at low δ is to utilize prior knowledge about human behavior; in particular, humans obey interaction rules (e.g. signaling intent) [56], which bound uncertainty in useful ways. These rules can be encoded through *assume-guarantee contracts*.

Definition 1. [10] *An assume-guarantee contract for an agent is a 2-tuple $(\mathcal{A}, \mathcal{G})$ where,*

- \mathcal{A} is a set of behavioral constraints that the agent assumes its environment to have.
- \mathcal{G} is a set of behavioral constraints that it must obey, as long as its environment satisfies \mathcal{A} .

A contract might encode that an agent cannot mislead others about its intention, assuming that others do not mislead it. For example, on the highway, an agent cannot first pretend to yield to a merging vehicle (i.e. slow down), before speeding up to hit that vehicle. We thus propose trading one challenge for another: rather than learning uncertainty bounds that agents obey with probability $1 - \delta$, we instead aim to learn/design contracts (i.e. behavioral constraints) they must surely obey.

We believe such a framework is necessary to combine *learned components that predict expected actions* and *prior knowledge restricting the danger of rare events* in a rigorous, interpretable manner.

References

- [1] J. Fisac, A. Bajcsy, S. Herbert, D. Fridovich-Keil, S. Wang, C. Tomlin, and A. Dragan. Probabilistically Safe Robot Planning with Confidence-Based Human Predictions. 2018.
- [2] G. S. Aoude, B. D. Luders, J. M. Joseph, N. Roy, and J. P. How. Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns. *Autonomous Robots*, 35(1):51–76, 2013.
- [3] D. Fridovich-Keil, A. Bajcsy, J. F. Fisac, S. L. Herbert, S. Wang, A. D. Dragan, and C. J. Tomlin. Confidence-aware motion prediction for real-time collision avoidance. *IJRR*, 2020.
- [4] Y. K. Nakka, A. Liu, G. Shi, A. Anandkumar, Y. Yue, and S.-J. Chung. Chance-Constrained Trajectory Optimization for Safe Exploration and Learning of Nonlinear Systems. *arXiv*, 2020.
- [5] D. D. Fan, A. Akbar Agha-mohammadi, and E. A. Theodorou. Deep learning tubes for tube mpc, 2020.
- [6] S. Shalev-Shwartz, S. Shammah, and A. Shashua. On a Formal Model of Safe and Scalable Self-driving Cars. *arXiv*, 2017.
- [7] National highway traffic safety administration. traffic safety facts annual report, 2019.
- [8] R. Krajewski, J. Bock, L. Kloecker, and L. Eckstein. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *International Conference on Intelligent Transportation Systems (ITSC)*, 2018.
- [9] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater, and L. Eckstein. The ind dataset: A drone dataset of naturalistic road user trajectories at german intersections. 2019.
- [10] T. Phan-Minh, K. X. Cai, and R. M. Murray. Towards assume-guarantee profiles for autonomous vehicles, 2019.
- [11] J. F. Fisac, E. Bronstein, E. Stefansson, D. Sadigh, S. S. Sastry, and A. D. Dragan. Hierarchical game-theoretic planning for autonomous vehicles. In *IEEE ICRA*, 2019.
- [12] A. Hakobyan and I. Yang. Learning-Based Distributionally Robust Motion Control with Gaussian Processes. *arXiv*, 2020.
- [13] R. Cheng, M. J. Khojasteh, A. D. Ames, and J. W. Burdick. Safe Multi-Agent Interaction through Robust Control Barrier Functions with Learned Uncertainties. *arXiv*, 2020.
- [14] W. Xu, J. Pan, J. Wei, and J. M. Dolan. Motion planning under uncertainty for on-road autonomous driving. In *IEEE ICRA*, 2014.
- [15] D. Sadigh and A. Kapoor. Safe control under uncertainty with Probabilistic Signal Temporal Logic. In *Robotics: Science and Systems*, volume 12, 2016.
- [16] M. Forghani, J. M. McNew, D. Hoehener, and D. Del Vecchio. Design of driver-assist systems under probabilistic safety specifications near stop signs. *IEEE T-ASE*, 13(1):43–53, 2016.
- [17] G. Kahn, A. Villafior, V. Pong, P. Abbeel, and S. Levine. Uncertainty-Aware Reinforcement Learning for Collision Avoidance. 2017.
- [18] D. D. Fan, J. Nguyen, R. Thakker, N. Alatur, A.-a. Agha-mohammadi, and E. A. Theodorou. Bayesian Learning-Based Adaptive Control for Safety Critical Systems. *arXiv*, 2019.
- [19] R. Michelmore, M. Wicker, L. Laurenti, L. Cardelli, Y. Gal, and M. Kwiatkowska. Uncertainty Quantification with Statistical Guarantees in End-to-End Autonomous Driving Control. 2019.
- [20] D. Sadigh, K. Driggs-Campbell, A. Puggelli, W. Li, V. Shia, R. Bajcsy, A. L. Sangiovanni-Vincentelli, S. S. Sastry, and S. A. Seshia. Data-driven probabilistic modeling and verification of human driver behavior. In *AAAI Spring Symposium*, pages 56–61, 2014.
- [21] W. Liu, S. W. Kim, S. Pendleton, and M. H. Ang. Situation-aware decision making for autonomous driving on urban road using online POMDP. In *IEEE IV*, 2015.
- [22] G. Cesari, G. Schildbach, A. Carvalho, and F. Borrelli. Scenario model predictive control for lane change assistance and autonomous driving on highways. *IEEE ITSM*, 9(3):23–35, 2017.
- [23] Y. Chen, S. Dathathri, T. Phan-Minh, and R. M. Murray. Counter-example Guided Learning of Bounds on Environment Behavior. *arXiv*, 2020.
- [24] H. Sartipizadeh and B. Açıkmeşe. Approximate convex hull based scenario truncation for chance constrained trajectory optimization. *Automatica*, 112, 2020.
- [25] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi. Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks. In *IEEE CVPR*, 2018.
- [26] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese. SoPhie: An attentive GAN for predicting paths compliant to social and physical constraints. In *IEEE CVPR*, 2019.
- [27] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone. Trajectron++: Multi-Agent Generative Trajectory Forecasting With Heterogeneous Data for Control. *arXiv*, 2020.

- [28] D. Ellis, E. Sommerlade, and I. Reid. Modelling pedestrian trajectory patterns with gaussian processes. In *IEEE ICCV Workshops*, pages 1229–1234, 2009.
- [29] P. Trautman, J. Ma, R. M. Murray, and A. Krause. Robot navigation in dense human crowds: Statistical models and experimental studies of human-robot cooperation. *IJRR*, 2015.
- [30] S. Ferguson, B. Luders, R. C. Grande, and J. P. How. Real-time predictive modeling and robust avoidance of pedestrians with uncertain, changing intentions. In *STAR*, 2015.
- [31] A. Liu, G. Shi, S.-J. Chung, A. Anandkumar, and Y. Yue. Robust Regression for Safe Exploration in Control. *arXiv*, 2019.
- [32] A. Gray, Y. Gao, T. Lin, J. K. Hedrick, and F. Borrelli. Stochastic predictive control for semi-autonomous vehicles with an uncertain driver model. In *IEEE ITSC*, pages 2329–2334, 2013.
- [33] N. Tagasovska and D. Lopez-Paz. Single-Model Uncertainties for Deep Learning. In *Neural Information Processing Systems*, 2018.
- [34] M. C. Campi and S. Garatti. Wait-and-judge scenario optimization. *Mathematical Programming*, 2018.
- [35] A. Carvalho, S. Lefèvre, G. Schildbach, J. Kong, and F. Borrelli. Automated driving: The role of forecasts and uncertainty - A control perspective. In *European Journal of Control*, volume 24, pages 14–32, 2015.
- [36] K. Driggs-Campbell, V. Govindarajan, and R. Bajcsy. Integrating Intuitive Driver Models in Autonomous Planning for Interactive Maneuvers. *IEEE T-ITS*, 18(12):3461–3472, 2017.
- [37] K. Driggs-Campbell, R. Dong, and R. Bajcsy. Robust, informative human-in-the-loop predictions via empirical reachable sets. *IEEE T-IV*, 3(3):300–309, 2018.
- [38] N. Li, D. Oyler, M. Zhang, Y. Yildiz, A. Girard, and I. Kolmanovsky. Hierarchical reasoning game theory based approach for evaluation and testing of autonomous vehicle control systems. In *IEEE Conference on Decision and Control*, 2016.
- [39] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan. Planning for autonomous cars that leverage effects on human actions. In *Robotics: Science and Systems*, 2016.
- [40] M. Kwon, E. Biyik, A. Talati, K. Bhasin, D. P. Losey, and D. Sadigh. When Humans Aren’t Optimal: Robots that Collaborate with Risk-Aware Humans. *arXiv*, 2020.
- [41] R. Kelley, A. Tavakkoli, C. King, M. Nicolescu, M. Nicolescu, and G. Bebis. Understanding human intentions via Hidden Markov Models in autonomous mobile robots. In *HRI*, 2008.
- [42] C. L. McGhan, A. Nasir, and E. M. Atkins. Human intent prediction using Markov decision processes. *Journal of Aerospace Information Systems*, 2015.
- [43] T. Bandyopadhyay, K. S. Won, E. Frazzoli, D. Hsu, W. S. Lee, and D. Rus. Intention-aware motion planning. In *Springer Tracts in Advanced Robotics*, 2013.
- [44] C. P. Lam and S. S. Sastry. A POMDP framework for human-in-the-loop system. In *IEEE Conference on Decision and Control*, 2014.
- [45] D. Tran, W. Sheng, L. Liu, and M. Liu. A Hidden Markov Model based driver intention prediction system. In *IEEE-CYBER*, 2015.
- [46] T. Gindele, S. Brechtel, and R. Dillmann. A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments. In *IEEE ITSC*, 2010.
- [47] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra. Weight uncertainty in neural networks, 2015.
- [48] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles, 2016.
- [49] Y. Gal and Z. Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, 2016.
- [50] Y. Hu, W. Zhan, and M. Tomizuka. Probabilistic Prediction of Vehicle Semantic Intention and Motion. In *IEEE Intelligent Vehicles Symposium*, volume 2018-June, pages 307–313, 2018.
- [51] W. Ding, J. Chen, and S. Shen. Predicting vehicle behaviors over an extended horizon using behavior interaction network. In *IEEE ICRA*, 2019.
- [52] K. Pelckmans, J. D. Brabanter, J. A. K. Suykens, and B. De Moor. Support and Quantile Tubes. *arXiv*, 2008.
- [53] A. Ehrenfeucht, D. Haussler, M. Kearns, and L. Valiant. A general lower bound on the number of examples needed for learning. *Information and Computation*, 1989.
- [54] S. Arora, A. Risteski, and Y. Zhang. Do GANs learn the distribution? some theory and empirics. In *International Conference on Learning Representations*, 2018.
- [55] A. Filos, P. Tigas, R. McAllister, N. Rhinehart, S. Levine, and Y. Gal. Can Autonomous Vehicles Identify, Recover From, and Adapt to Distribution Shifts? *arXiv*, 2020.
- [56] M. E. Bratman. Shared Cooperative Activity. *The Philosophical Review*, 1992.

A Fragility of Gaussian Uncertainty Model (Synthetic Data)

We further tested the Gaussian uncertainty model on a synthetic 2D data set, using the same process detailed in Section 3.2. Each 2D data point is analogous to a trajectory, $a = \tau_{[2:10]} \sim \mathcal{A}$, in the highD/inD driving dataset. Therefore, the goal is still to learn the model \hat{F} that best matches the data distribution \mathcal{A} , minimizing $m(\mathcal{A}||\hat{F})$. However, using synthetic data allows us to test the accuracy of the uncertainty model with respect to a *known* underlying probability distribution, \mathcal{A} .

We randomly generated 10,000 2D points for training data (further increasing the amount of training data did not improve performance) from 3 different distributions: (a) perfect Gaussian, (b) Gaussian with uniform noise (magnitude of noise was 30% of the data range), and (c) Gaussian with symmetric non-uniform noise (also 30% magnitude). For each of these training datasets, we computed the Gaussian uncertainty model that best fit the data. We then generated 10,000,000 2D points for our test data *following the exact same distribution as the training data*, and observed how well our computed Gaussian uncertainty model captured the test data.

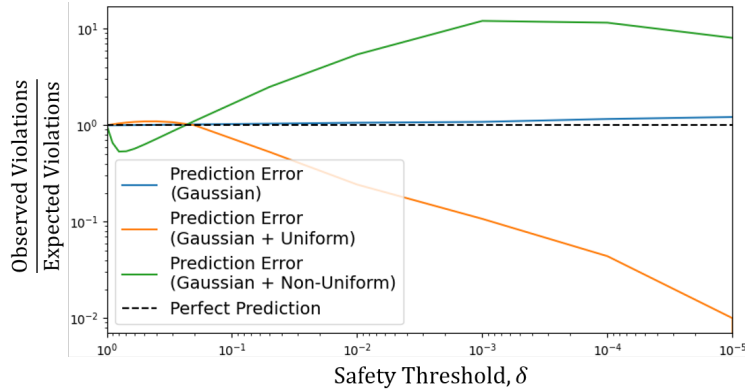


Figure 7: Prediction error vs. safety threshold, δ , using a Gaussian uncertainty model on synthetic 2D data generated from 3 different distributions. The dashed black line represents a perfect prediction model. Significant prediction error arises when the underlying data distribution is non-Gaussian.

Figure 7 shows that the learned uncertainty model performed very well when the underlying data distribution was Gaussian (blue curve). However, it performed poorly (off by an order-of-magnitude) at low δ when the underlying distribution was non-Gaussian. When the underlying distribution was Gaussian with added uniform noise (orange curve), the observed violations were much lower than the expected violations (i.e. the model was conservative). This is good for safety, but would clearly lead to overly conservative behavior, especially since the model is off by orders of magnitude.

However, more concerning is the case when the underlying distribution is Gaussian with *non-uniform* noise (green curve). In this case, the observed violations were much higher than the expected violations (greater by an order of magnitude), posing a clear risk for safety-critical applications. This reinforces our results in Section 3.2 by illustrating that significant prediction error inevitably arises, regardless of the amount of training data, when the underlying data distribution is non-Gaussian.

B Quantile Regression (Synthetic Data)

We repeated the quantile regression experiments from Section 3.3, using synthetic 2D data rather than real-world driving data. This allowed us to observe how well the uncertainty model performed under ideal conditions when the training/testing data were perfectly i.i.d. We randomly generated 1,000,000 2D training data points (analogous to 1,000,000 trajectories) following a Gaussian distribution, and computed δ -quantile bounds following the same procedure described in Section 3.3 (i.e. computing the smallest convex set containing $1 - \delta$ points). We then generated 10,000,000 2D test data points *following the exact same distribution as the training data*, and observed how well our computed quantile bounds captured the test data.

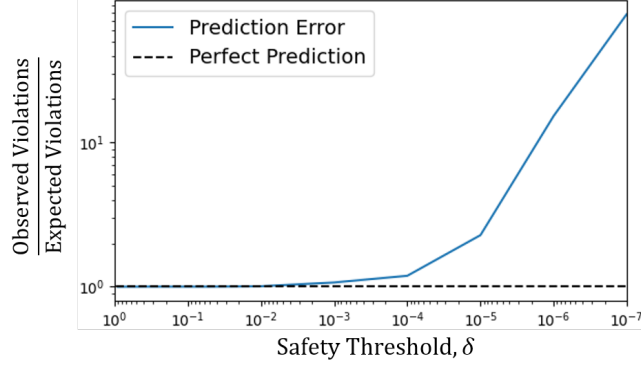


Figure 8: Prediction error vs. safety threshold, δ under computed quantile bounds on synthetic 2D data. The dashed black line represents a perfect prediction model.

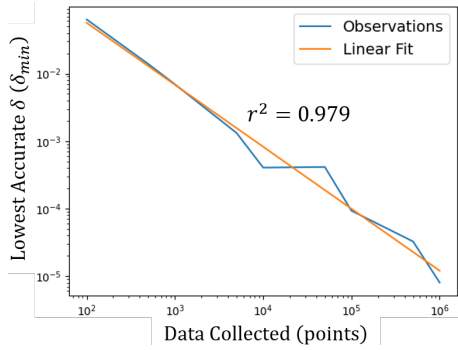


Figure 9: Smallest accurate δ versus amount of data using synthetic 2D data. The trend is highly linear ($r^2 = 0.979$).

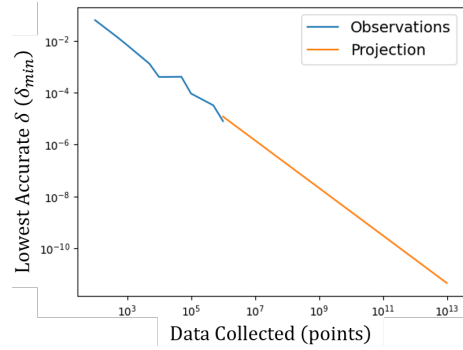


Figure 10: Projection showing the expected amount of data required to obtain an accurate safety threshold δ .

Figure 8 shows the prediction error (i.e. ratio between expected and observed proportion of trajectories outside each quantile) versus the safety threshold δ . The quantile regression model performed very well up to $\delta \geq 0.001$. However, performance rapidly deteriorated as δ decreased, meaning the model failed to accurately predict violation probabilities at those safety thresholds. This is consistent with our results in Section 3.3.

Using the synthetic data, we computed the smallest accurate safety threshold, δ_{min} , as a function of the amount of training data, N . This threshold δ_{min} was defined as follows,

$$\delta_{min} = \min \delta \quad \text{such that} \quad \left| \log \left(\frac{\text{expected}(\delta)}{\text{observed}(\delta)} \right) \right| \leq \varepsilon. \quad (8)$$

where we set $\varepsilon = 0.5$, which represents the vertical distance between the blue curve in Fig 8 and the dotted black line. Therefore, δ_{min} represents the smallest δ such that our computed quantile bounds are ε -accurate (as described in Section 3.3). Figure 9 shows the same inverse linear trend ($\delta_{min} \propto \frac{1}{N}$) on the synthetic data that was seen with the real driving data. Figure 10 shows the extrapolation of this trend towards lower δ_{min} . This result reinforces the point made in Section 3.3 that quantile regression can be very accurate for larger δ , but it may not be feasible to collect enough data to reach safety thresholds $\delta_{min} \leq 10^{-8}$.

C Other Uncertainty Models

C.1 Generative Models

Generative models have garnered significant interest in trajectory prediction for their ability to implicitly learn the distribution $\mathcal{A}(x) = p(a|x)$. However, there are two significant issues with these

approaches, the first of which is the time required to utilize these models in safety-critical situations. For example at best, a single prediction takes $\approx 0.05s$ with Social-GAN [25], or $\approx 0.001s$ with Trajectron [27]. In order to guarantee safety with probability $\delta = 0.01$, we would need to generate 100 trajectories taking $> 0.1s$. To guarantee safety with probability $\delta = 10^{-8}$, we would need to generate 10^8 trajectories taking $> 100,000s$ (> 1 day), which is not suitable for real-time operation. While computational cost will surely decrease over time, it is unclear whether this modeling approach will be feasible in the near future.

More importantly, there are no guarantees that the uncertainty distribution implicitly captured by generative models provides any reasonable approximation to the true uncertainty distribution. It has been shown – both empirically and theoretically – that GANs can fail to learn the true distribution (suffering from “mode collapse”), even when their training objective nears optimality [54]. Furthermore, the theoretical data efficiency bound (7) suggests that the implicit distribution learned by such models will be inaccurate (at the safety thresholds we are considering) without currently infeasible amounts of data.

C.2 Scenario Optimization Model

Scenario Optimization is an appealing approach because (like quantile regression) it does not assume an underlying distribution over the data [24]. It relies only on the assumption that the data is drawn i.i.d. from some fixed (unknown) distribution. Therefore, we can obtain a high-confidence bound on the probability that a new trajectory is inside or outside a computed tube, *without strong assumptions on the underlying distribution*.

With this approach, the safety threshold, δ , is a direct function of the amount of observed data [34]; in other words $\delta = \delta(N)$, where N is the number of training trajectories or “samples”. Therefore, we cannot set arbitrarily small confidence levels (e.g. $\delta = 10^{-8}$). While this prevents users from applying the approach inappropriately, it requires very large amounts of data to get to low enough confidence levels for safety-critical applications. For example with 40,000 trajectories, we were able to reach $\delta \approx 10^{-4}$ (after this point, computational feasibility became an issue). This suggests it is not feasible to reach desired δ levels given realistic datasets.

Using the highD dataset, and treating the trajectories in the training set as observed samples, we obtained high-confidence bounds (computed as the convex hull of the training trajectories) such that new trajectories should lie within those bounds with probability at least $1 - \delta$. For example, Figure 11 shows the predicted confidence bounds for two representative driving instances; in the left plot, the goal position is *not* given but in the right plot, the goal position *is* given. The scenario optimization approach predicts that the future trajectory of each car should fall within the blue confidence bounds at 2, 4, 6 seconds in the future with 98.5% (left plot) or 95.1% (right plot) probability.

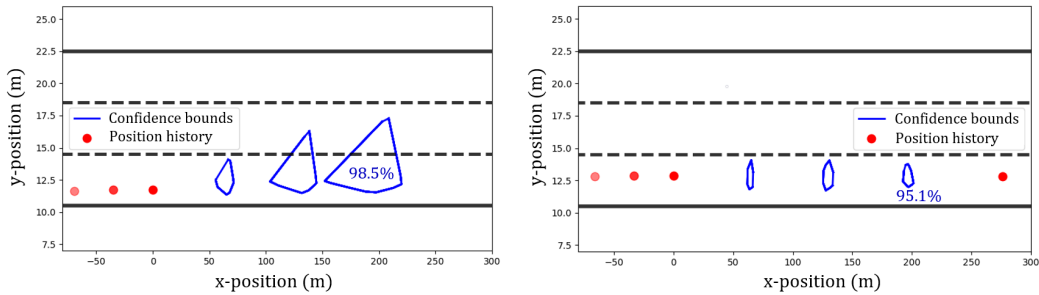


Figure 11: Plot of confidence bounds over the car’s future trajectory. The car’s positional history is shown by the red circles, and training data is taken from equivalent scenarios in the highD dataset. (Left) The goal position of the car is *not* known. We compute a 98.5% probability that a new trajectory falls within the blue confidence bounds at 2, 4, and 6 seconds in the future. (Right) The goal position of the car *is* known. We compute a 95.1% probability that a new trajectory falls within the blue confidence bounds at 2, 4, and 6 seconds in the future.

To test the accuracy of the computed confidence bounds, we examined how often trajectories in the test set actually remained within those bounds in the highD dataset. The ratio of observed violations to expected violations was smaller than expected (i.e. the method was conservative),

518 which is reassuring for safety. Specifically, the observed vs. expected percentage of violations was
 519 approximately 5% vs. 14%.

520 However, the safety threshold $\delta(N)$ was always large ($\delta \in [0.02, 0.41]$) and unable to be arbitrarily
 521 defined, which makes the scenario optimization approach currently inapplicable to many safety-
 522 critical applications. This is consistent with our conclusion in Section 3.3, that much more data is
 523 necessary to obtain reliable, probabilistic bounds.

524 C.3 Hidden Markov Models

525 Rather than reasoning about uncertainty only over trajectories, many methods in the POMDP liter-
 526 ature reason about uncertainty over discrete intentions. Most often, these discrete intentions denote
 527 different goal positions for the agent, but they could also denote different operational modes (e.g.
 528 left vs. right turn). Hidden Markov models enable us to compute an agent’s most likely intention,
 529 which proves useful in solving many challenging problems. However, when guaranteeing safety
 530 with safety threshold δ , intention must be correctly inferred with probability $1 - \delta$. Issues arise
 531 when the intention must be inferred with very high confidence $\delta \leq 10^{-8}$.

532 We demonstrate this on a 1D toy problem with synthetic data. We generated 1000 i.i.d. data points
 533 from two distinct distributions (mode 1 and mode 2), and computed the best fit Gaussian for each
 534 of these distributions. Note that our results did not change when increasing the amount of data. We
 535 then computed the intervals in which a new point would have to lie in order for us to classify it in
 536 either mode 1 or mode 2 with $1 - \delta$ confidence. This was done by applying Bayes rule, assuming a
 537 uniform prior over the modes,

$$\mathbb{P}(\text{mode} | x) = \frac{\mathbb{P}(x | \text{mode}) \mathbb{P}(\text{mode})}{\mathbb{P}(x)}. \quad (9)$$

538 Figure 12 shows these intervals when the points were generated from (a) a Gaussian distribution,
 539 and (b) a uniform distribution. The interval covered by the gray line denotes the interval in which
 540 we can *not* classify (with δ confidence) a point’s mode. We note that the gray line extends across
 541 a significant portion of the data range, but is reasonable when the underlying distribution of points
 542 in each mode is *perfectly Gaussian*. However, when the generated data is uniformly random, the
 543 uncertainty interval stretches across the entire range of data. This suggests that inferring intention
 544 or hidden “modes” under uncertainty will often be infeasible when considering very low safety
 545 thresholds, δ , especially since we have shown that human behavioral variation is non-Gaussian.
 546 Furthermore, we cannot compensate for this non-Gaussian variation as we do not have accurate
 547 knowledge of the true distribution.

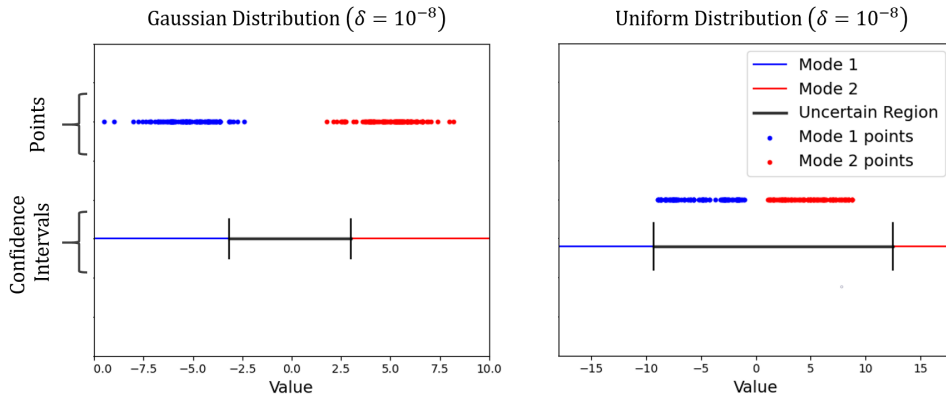


Figure 12: Synthetic data is generated from two different modes: (*mode 1 – blue, mode 2 – red*). The confidence intervals below denote where a point would have to lie in order to classify it, with confidence $\delta = 10^{-8}$, as coming from either mode 1 or mode 2. For example, if a new point falls in the interval covered by the blue bar, it can be classified as coming from mode 1 with confidence $\delta \leq 10^{-8}$. If it falls anywhere in the gray interval, we cannot conclude its mode (assuming a uniform prior). (a) Data for each mode is generated from a Gaussian distribution. (b) Data for each mode is generated from a uniform distribution.